


# **Sterling Call Center and Sterling Store: Implementation Guide**

Release 9.0

*Last updated in HF3*

August 2010



# Copyright Notice

Copyright © 2005 - 2010

Sterling Commerce, Inc.

ALL RIGHTS RESERVED

RESTRICTED RIGHTS

## STERLING COMMERCE SOFTWARE

\*\*\*TRADE SECRET NOTICE\*\*\*

THE STERLING COMMERCE SOFTWARE DESCRIBED BY THIS DOCUMENTATION ("STERLING COMMERCE SOFTWARE") IS THE CONFIDENTIAL AND TRADE SECRET PROPERTY OF STERLING COMMERCE, INC., ITS AFFILIATED COMPANIES OR ITS OR THEIR LICENSORS, AND IS PROVIDED UNDER THE TERMS OF A LICENSE AGREEMENT. NO DUPLICATION OR DISCLOSURE WITHOUT PRIOR WRITTEN PERMISSION. RESTRICTED RIGHTS.

This documentation, the Sterling Commerce Software it describes, and the information and know-how they contain constitute the proprietary, confidential and valuable trade secret information of Sterling Commerce, Inc., its affiliated companies or its or their licensors, and may not be used for any unauthorized purpose, or disclosed to others without the prior written permission of the applicable Sterling Commerce entity. This documentation and the Sterling Commerce Software that it describes have been provided pursuant to a license agreement that contains prohibitions against and/or restrictions on their copying, modification, and use. Duplication, in whole or in part, if and when permitted, shall bear this notice and the Sterling Commerce, Inc. copyright notice.

U.S. GOVERNMENT RESTRICTED RIGHTS. This documentation and the Sterling Commerce Software it describes are "commercial items" as defined in 48 C.F.R. 2.101. As and when provided to any agency or instrumentality of the U.S. Government or to a U.S. Government prime contractor or a subcontractor at any tier ("Government Licensee"), the terms and conditions of the customary Sterling Commerce commercial license agreement are imposed on Government Licensees per 48 C.F.R. 12.212 or 227.7202 through 227.7202-4, as applicable, or through 48 C.F.R. 52.244-6.

These terms of use shall be governed by the laws of the State of Ohio, USA, without regard to its conflict of laws provisions. If you are accessing the Sterling Commerce Software under an executed agreement, then nothing in these terms and conditions supersedes or modifies the executed agreement.

---

Sterling Commerce, Inc.  
4600 Lakehurst Court  
Dublin, Ohio 43016-2000

Copyright © 2005 - 2010

## **WARRANTY DISCLAIMER**

This documentation and the Sterling Commerce Software which it describes are licensed either "AS IS" or with a limited warranty, as set forth in the Sterling Commerce license agreement. Other than any limited warranties provided, NO OTHER WARRANTY IS EXPRESSED AND NONE SHALL BE IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE OR FOR A PARTICULAR PURPOSE. Without limitation to the foregoing, Sterling Commerce does not warrant or represent that use of this software will ensure compliance with the Federal Trade Commission's (FTC) Mail or Telephone Order Merchandise Trade Regulation Rule (the "Rule"), with the U.S. Department of Transportation's (DOT) Hazardous Materials Regulations (HMR) found in Title 49 of the Code of Federal Regulations (49 CFR), or any other legal requirements, and users of this software should consult independent legal counsel and technical support to ensure compliance with the Rule, with the HRM, and other legal requirements. The applicable Sterling Commerce entity reserves the right to revise this publication from time to time and to make changes in the content hereof without the obligation to notify any person or entity of such revisions or changes.

## **Third-Party Materials**

### **Third-Party Links**

The Sterling Commerce Software may include links or references to internet web sites and resources operated or offered by third parties. Such links are provided to facilitate your acquisition of third party software products or services which may enable or otherwise enhance your use of the Sterling Commerce Software. Sterling Commerce makes no representations, warranties, or other commitments whatsoever about any such web sites or third party resources. You are solely responsible for any transaction, download, installation and use of any software product or services made available via such third party web sites or resources, and for compliance with any and all terms applicable to such third party software products or services. Accordingly, Sterling Commerce is not responsible for the availability of such third party web sites or resources, and is not responsible for any use of, or reliance on, any such content, products, services, or other materials on, or available through, such web sites or resources.

### **Third-Party Software**

The Sterling Commerce Software may be distributed with or otherwise rely upon other Sterling Commerce software which may be distributed on the same storage media with products, ("Third Party Software") offered by third parties ("Third Party Licensors"). Third Party Software information for such additional components of the Sterling Commerce Software is located at: <INSTALL\_DIR>/README\_COM\_PCA.htm.



# Contents

---

## Preface

Intended Audience .....	xxix
Structure .....	xxix
Documentation .....	xxx
Conventions .....	xxxiii

## 1 Introduction

## 2 Process Types

2.1	Order Fulfillment .....	3
2.1.1	Create .....	4
2.1.2	Schedule .....	4
2.1.3	Release .....	5
2.1.4	Ship .....	5
2.1.5	Return .....	6
2.1.6	Close .....	6
2.1.7	Purge .....	7
2.2	Outbound Shipment .....	7
2.2.1	Create .....	7
2.2.2	Backroom Pick .....	7
2.2.3	Confirm .....	8
2.2.4	Create Invoice .....	8
2.2.5	Close .....	8
2.3	Reverse Logistics .....	8

2.3.1	Create .....	9
2.3.2	Authorize .....	9
2.3.3	Schedule.....	9
2.3.4	Receive.....	9
2.3.5	Close .....	10
2.3.6	Purge .....	10

### **3 Extending and Customizing the Application**

3.1	Order Entry Screens Customization.....	11
3.2	Add Lines To Order Screens Customization .....	13
3.3	Extending Actions.....	13
3.4	Shared Tasks.....	14
3.4.1	Address Panel Shared Task .....	14
3.4.2	Order Notes Shared Task.....	15
3.4.3	Service Appointment Calendar Shared Task .....	15
3.4.4	Add Payment Method Shared Task.....	16
3.4.5	Order Summary Shared Task .....	17
3.4.6	Return Search Shared Task .....	18
3.4.7	Cancel Order Shared Task .....	18
3.4.8	Create Alert Shared Task.....	19
3.4.9	Alert List Shared Task .....	19
3.4.10	Message Dialog Box Shared Task.....	19
3.4.11	Item Details Shared Task .....	20
3.4.12	Add Modify Charges Shared Task.....	22
3.4.13	Customer Details Shared Task .....	23
3.4.14	Advanced Item Search Shared Task.....	27
3.4.15	Price Match Shared Task .....	29
3.4.16	Override Item Price Shared Task .....	30
3.5	User Interface Customization .....	30
3.5.1	Special Themes.....	30
3.5.1.1	Mandatory Fields .....	31
3.5.1.2	Smileys.....	31
3.5.2	Display of Day Panels in the Service Appointment Calendar.....	31
3.5.3	User Interface Validation of Phone Number and E-Mail Address.....	32
3.6	Address Panel Customization .....	33

3.7	Opening New Related Tasks in the Order Editor .....	33
3.8	Opening the Screens of the Sterling Call Center and Sterling Store Application in an Editor .....	34
3.8.1	Opening the Order Summary Screen in an Editor .....	34
3.8.2	Opening the Return Search Screen in an Editor .....	35
3.8.3	Opening the Cancel Order Screen in an Editor .....	35
3.8.4	Opening the Create Alert Screen in an Editor .....	36
3.8.5	Opening the Alert List Screen in an Editor .....	36
3.8.6	Opening the Alert Details Screen in an Editor .....	37
3.8.7	Opening the Item Details Screen in an Editor .....	37
3.8.8	Opening the Customer Details Screen in an Editor .....	39
3.8.9	Opening the Return Order Summary Screen in an Editor.....	44
3.8.10	Opening the Advanced Item Search Screen in an Editor.....	45
3.9	Opening the Price Match Wizard in an Editor .....	45

## **4 Implementing the Sterling Call Center Features**

4.1	Item Inquiry.....	47
4.1.1	Searching and Viewing the Details of an Item .....	47
4.1.1.1	Solution .....	47
4.1.1.2	End-User Impact .....	51
4.1.1.3	Implementation .....	51
4.1.1.4	Reference Implementation.....	53
4.1.2	Handling Styles for an Item .....	53
4.1.2.1	Solution .....	54
4.1.2.2	End-User Impact .....	56
4.1.2.3	Implementation .....	58
4.1.2.4	Reference Implementation.....	61
4.2	Configured Items .....	62
4.2.1	Solution .....	62
4.2.2	End-User Impact .....	63
4.2.3	Implementation .....	63
4.2.4	Reference Implementation.....	64
4.3	Alternate Item Identifier .....	64
4.3.1	Solution .....	64
4.3.2	End-User Impact .....	65

4.3.3	Implementation .....	65
4.3.4	Reference Implementation.....	66
4.4	Item Display Options Configuration.....	66
4.4.1	Solution .....	66
4.4.2	End-User Impact .....	66
4.4.3	Implementation .....	67
4.4.4	Reference Implementation.....	70
4.5	Alternate Store Pickup Locations Search .....	70
4.5.1	Solution .....	70
4.5.2	End-User Impact .....	71
4.5.3	Implementation .....	71
4.5.4	Reference Implementation.....	72
4.6	Order Capture.....	72
4.6.1	Order Creation .....	73
4.6.1.1	Solution .....	73
4.6.1.2	End-User Impact .....	95
4.6.1.3	Implementation .....	96
4.6.1.4	Reference Implementation.....	97
4.6.2	Capturing Orders through Other Channels .....	98
4.6.2.1	Solution .....	98
4.6.2.2	End-User Impact .....	102
4.6.2.3	Implementation .....	102
4.6.2.4	Reference Implementation.....	103
4.6.3	Order Delay.....	103
4.6.3.1	Solution .....	103
4.6.3.2	End-User Impact .....	103
4.6.3.3	Implementation .....	103
4.6.3.4	Reference Implementation.....	103
4.6.4	Order Validation .....	104
4.6.4.1	Duplicate Order Validation .....	104
4.6.4.2	Address Verification .....	109
4.6.4.3	Fraud Check .....	116
4.6.4.4	Item Validation .....	120
4.6.4.5	Credit Limit Exceeded .....	123
4.7	Order Inquiry.....	124



4.7.1	Solution .....	124
4.7.2	End-User Impact .....	128
4.7.3	Implementation .....	128
4.7.4	Reference Implementation.....	129
4.8	Order Monitoring.....	129
4.8.1	Federal Trade Commission Compliance.....	130
4.8.1.1	Solution .....	131
4.8.1.2	End-User Impact .....	137
4.8.1.3	Implementation .....	137
4.8.2	Shipment Notification .....	147
4.8.2.1	Solution .....	147
4.8.2.2	End-User Impact .....	149
4.8.2.3	Implementation .....	149
4.8.3	Order Cancellation Notification .....	150
4.8.3.1	Solution .....	150
4.8.3.2	End-User Impact .....	152
4.8.3.3	Implementation .....	153
4.9	Order Maintenance .....	153
4.9.1	Apply Hold .....	155
4.9.1.1	Solution .....	155
4.9.1.2	End-User Impact .....	155
4.9.1.3	Implementation .....	155
4.9.1.4	Reference Implementation.....	156
4.9.2	Remove Holds .....	156
4.9.2.1	Solution .....	156
4.9.2.2	End-User Impact .....	157
4.9.2.3	Implementation .....	157
4.9.2.4	Reference Implementation.....	157
4.9.3	Change Payment Method.....	157
4.9.3.1	Solution .....	158
4.9.3.2	End-User Impact .....	168
4.9.3.3	Implementation .....	168
4.9.3.4	Reference Implementation.....	169
4.9.4	Add Line .....	170
4.9.4.1	Solution .....	170

4.9.4.2	End-User Impact .....	172
4.9.4.3	Implementation .....	173
4.9.4.4	Reference Implementation.....	174
4.9.5	Add Multiple Items to an Order .....	174
4.9.5.1	Solution .....	174
4.9.5.2	End-User Impact .....	176
4.9.5.3	Implementation .....	177
4.9.5.4	Reference Implementation.....	178
4.9.6	Change Gift Options.....	179
4.9.6.1	Solution .....	179
4.9.6.2	End-User Impact .....	180
4.9.6.3	Implementation .....	180
4.9.6.4	Reference Implementation.....	181
4.9.7	Customer Appeasement .....	181
4.9.7.1	Solution .....	182
4.9.7.2	End-User Impact .....	182
4.9.7.3	Implementation .....	183
4.9.7.4	Reference Implementation.....	183
4.9.8	Price Match .....	184
4.9.8.1	Solution .....	184
4.9.8.2	End-User Impact .....	189
4.9.8.3	Implementation .....	189
4.9.8.4	Reference Implementation.....	190
4.9.9	Competitor Search .....	191
4.9.9.1	Solution .....	191
4.9.9.2	End-User Impact .....	192
4.9.9.3	Implementation .....	192
4.9.9.4	Reference Implementation.....	192
4.9.10	Price Match Search .....	193
4.9.10.1	Solution .....	193
4.9.10.2	End-User Impact .....	194
4.9.10.3	Implementation .....	194
4.9.10.4	Reference Implementation.....	194
4.9.11	Create or Modify a Competitor Record .....	194
4.9.11.1	Solution .....	194

4.9.11.2	End-User Impact .....	195
4.9.11.3	Implementation .....	195
4.9.11.4	Reference Implementation.....	195
4.9.12	Create or Modify a Price Match Record .....	195
4.9.12.1	Solution .....	196
4.9.12.2	End-User Impact .....	197
4.9.12.3	Implementation .....	197
4.9.12.4	Reference Implementation.....	197
4.9.13	Reship an Item .....	197
4.9.13.1	Solution .....	197
4.9.13.2	End-User Impact .....	199
4.9.13.3	Implementation .....	199
4.9.13.4	Reference Implementation.....	200
4.9.14	Add or Modify Charges .....	200
4.9.14.1	Solution .....	200
4.9.14.2	End-User Impact .....	201
4.9.14.3	Implementation .....	201
4.9.14.4	Reference Implementation.....	202
4.9.15	Add Coupon.....	203
4.9.15.1	Solution .....	203
4.9.15.2	End-User Impact .....	203
4.9.15.3	Implementation .....	203
4.9.15.4	Reference Implementation.....	204
4.9.16	Cancel Order .....	204
4.9.16.1	Solution .....	204
4.9.16.2	End-User Impact .....	207
4.9.16.3	Implementation .....	207
4.9.16.4	Reference Implementation.....	208
4.9.17	Track an Item .....	209
4.9.17.1	Solution .....	209
4.9.17.2	End-User Impact .....	211
4.9.17.3	Implementation .....	212
4.9.17.4	Reference Implementation.....	212
4.9.18	Order Notes.....	212
4.9.18.1	Solution .....	212

4.9.18.2	End-User Impact .....	214
4.9.18.3	Implementation .....	215
4.9.18.4	Reference Implementation.....	216
4.9.19	Change Fulfillment Options .....	218
4.9.19.1	Solution .....	218
4.9.19.2	End-User Impact .....	222
4.9.19.3	Implementation .....	223
4.9.19.4	Reference Implementation.....	223
4.9.20	Fulfillment Summary.....	224
4.9.20.1	Solution .....	224
4.9.20.2	End-User Impact .....	230
4.9.20.3	Implementation .....	230
4.9.20.4	Reference Implementation.....	231
4.9.21	Delivery and Service Grouping .....	231
4.9.21.1	Solution .....	231
4.9.21.2	End-User Impact .....	232
4.9.21.3	Implementation .....	232
4.9.21.4	Reference Implementation.....	233
4.9.22	Change Service Instructions .....	233
4.9.22.1	Solution .....	233
4.9.22.2	End-User Impact .....	233
4.9.22.3	Implementation .....	233
4.9.22.4	Reference Implementation.....	234
4.9.23	Change Service Appointment .....	234
4.9.23.1	Solution .....	234
4.9.23.2	End-User Impact .....	236
4.9.23.3	Implementation .....	236
4.9.23.4	Reference Implementation.....	237
4.9.24	Service Fulfillment Summary .....	237
4.9.24.1	Solution .....	237
4.9.24.2	End-User Impact .....	238
4.9.24.3	Implementation .....	239
4.9.24.4	Reference Implementation.....	239
4.9.25	Reservations.....	239
4.9.25.1	Solution .....	239

4.9.25.2	Implementation .....	240
4.9.25.3	End-User Impact .....	240
4.9.25.4	Reference Implementation.....	241
4.9.26	Increase Order Line Quantity .....	241
4.9.26.1	Solution .....	241
4.9.26.2	End-User Impact .....	243
4.9.26.3	Implementation .....	243
4.9.26.4	Reference Implementation.....	243
4.9.27	Schedule and Release an Order .....	243
4.9.27.1	Solution .....	244
4.9.27.2	End-User Impact .....	244
4.9.27.3	Implementation .....	244
4.9.27.4	Reference Implementation.....	244
4.9.28	View Procurement Orders .....	245
4.9.28.1	Solution .....	245
4.9.28.2	End-User Impact .....	245
4.9.28.3	Implementation .....	245
4.9.28.4	Reference Implementation.....	245
4.9.29	Change Order Address .....	245
4.9.29.1	Solution .....	246
4.9.29.2	End-User Impact .....	247
4.9.29.3	Implementation .....	247
4.9.29.4	Reference Implementation.....	247
4.9.30	Launch Application Console.....	248
4.9.30.1	Solution .....	248
4.9.30.2	End-User Impact .....	248
4.9.30.3	Implementation .....	248
4.9.30.4	Reference Implementation.....	249
4.9.31	Change an Item's Style .....	249
4.9.31.1	Solution .....	250
4.9.31.2	End-User Impact .....	250
4.9.31.3	Implementation .....	250
4.9.31.4	Reference Implementation.....	250
4.9.32	Reconfigure Bundle Items .....	251
4.9.32.1	Solution .....	251

4.9.32.2	End-User Impact .....	252
4.9.32.3	Implementation .....	252
4.9.32.4	Reference Implementation.....	253
4.9.33	Change Tax Options .....	253
4.9.33.1	Solution .....	253
4.9.33.2	End-User Impact .....	253
4.9.33.3	Implementation .....	253
4.9.33.4	Reference Implementation.....	254
4.9.34	Copy Order.....	254
4.9.34.1	Solution .....	254
4.9.34.2	End-User Impact .....	254
4.9.34.3	Implementation .....	254
4.9.34.4	Reference Implementation.....	255
4.9.35	E-Mail Order Information.....	255
4.9.35.1	Solution .....	255
4.9.35.2	End-User Impact .....	258
4.9.35.3	Implementation .....	258
4.9.35.4	Reference Implementation.....	258
4.9.36	Transfer Draft Order to Sterling Web.....	258
4.9.36.1	Solution .....	259
4.9.36.2	End-User Impact .....	260
4.9.36.3	Implementation .....	260
4.9.36.4	Reference Implementation.....	260
4.10	Payment Processing.....	260
4.10.1	Solution .....	261
4.10.1.1	Credit Card.....	263
4.10.1.2	Stored Value Card .....	264
4.10.1.3	Pre-paid.....	265
4.10.1.4	Check .....	265
4.10.1.5	Customer Account .....	266
4.10.1.6	Refund Check .....	266
4.10.2	End-User Impact .....	267
4.10.3	Implementation .....	267
4.10.3.1	Credit Card Implementation.....	269
4.10.3.2	Stored Value Card Implementation .....	272

4.10.3.3	Return Check Implementation.....	274
4.10.3.4	Payment Processing Error Handling.....	276
4.10.3.5	Strike Limit Configuration.....	279
4.10.4	Reference Implementation.....	279
4.11	Payment Inquiry .....	282
4.11.1	Solution .....	282
4.11.2	End-User Impact.....	283
4.11.3	Implementation .....	283
4.11.4	Reference Implementation.....	283
4.12	Real-Time Credit Card Authorization .....	283
4.12.1	Solution .....	283
4.12.2	End-User Impact.....	285
4.12.3	Implementation .....	286
4.12.4	Reference Implementation.....	286
4.13	Return Order Capture .....	286
4.13.1	Solution .....	287
4.13.1.1	Return Line Selection.....	287
4.13.1.2	Identifying the Person Returning an Item. ....	288
4.13.1.3	Adding Lines to a Return .....	288
4.13.1.4	Validating Lines and Addresses .....	288
4.13.1.5	Entering Return Reason .....	288
4.13.1.6	Return Reason Rule .....	289
4.13.1.7	Overriding the Return Policy .....	289
4.13.1.8	Customer Identification Required.....	289
4.13.1.9	Is Customer Info Present Rule.....	289
4.13.1.10	Modifying an Exchange .....	290
4.13.1.11	Default Exchange Type Rules .....	290
4.13.1.12	Needs Appointments Rule.....	290
4.13.1.13	Address Info Needed Rule .....	290
4.13.1.14	Create Reservation Rule.....	290
4.13.1.15	Exchange Fulfillment Options .....	290
4.13.1.16	Needs Appointment Rule .....	291
4.13.1.17	Is Customer Info Present Rule.....	291
4.13.1.18	Address Info Needed Rule .....	291
4.13.1.19	Create Reservation Rule.....	291

4.13.1.20	Fulfillment Summary.....	291
4.13.1.21	Select Appointment .....	291
4.13.1.22	Payment Summary .....	292
4.13.2	End-User Impact .....	292
4.13.2.1	Line Selection .....	292
4.13.2.2	Exchange Fulfillment Options .....	292
4.13.3	Implementation .....	292
4.13.3.1	Line Selection .....	293
4.13.3.2	Return Reason .....	293
4.13.3.3	Modify an Exchange .....	293
4.13.3.4	Exchange Fulfillment Options .....	293
4.13.3.5	Fulfillment Summary.....	294
4.13.3.6	Status Changes to a Return or Exchange Order .....	294
4.13.3.7	Select Appointment .....	295
4.13.4	Reference Implementation.....	295
4.13.4.1	Return Reason .....	295
4.13.4.2	Exchange Fulfillment Options .....	296
4.13.4.3	Select Appointment .....	296
4.14	Return Order Inquiry .....	296
4.14.1	Solution .....	296
4.14.2	End-User Impact .....	297
4.14.3	Implementation .....	297
4.14.4	Reference Implementation.....	297
4.15	Return Order Maintenance.....	298
4.15.1	Return Line Selection .....	298
4.15.1.1	Solution .....	298
4.15.1.2	End User Impact .....	303
4.15.1.3	Implementation .....	303
4.15.1.4	Reference Implementation.....	303
4.15.2	Entering Return Reason.....	303
4.15.2.1	Solution .....	303
4.15.2.2	End User Impact .....	305
4.15.2.3	Implementation .....	306
4.15.2.4	Reference Implementation.....	306
4.15.3	Modifying an Exchange .....	306



4.15.3.1	Solution .....	306
4.15.3.2	End User Impact .....	307
4.15.3.3	Implementation .....	307
4.15.3.4	Reference Implementation.....	307
4.15.4	Exchange Fulfillment Options .....	307
4.15.5	Fulfillment Summary .....	308
4.15.6	Select Appointment .....	308
4.15.7	Payment Summary.....	308
4.15.7.1	Solution .....	308
4.15.7.2	End User Impact .....	310
4.15.7.3	Implementation .....	310
4.15.7.4	Reference Implementation.....	310
4.15.8	Create Exchange Order .....	310
4.15.8.1	Solution .....	310
4.15.8.2	End User Impact .....	311
4.15.8.3	Implementation .....	311
4.15.8.4	Reference Implementation.....	311
4.15.9	Report Extra Items.....	311
4.15.9.1	Solution .....	312
4.15.9.2	End-User Impact.....	312
4.15.9.3	Implementation .....	313
4.15.9.4	Reference Implementation.....	313
4.15.10	Report Wrong Items .....	313
4.15.10.1	Solution .....	314
4.15.10.2	End-User Impact.....	315
4.15.10.3	Implementation .....	315
4.15.10.4	Reference Implementation.....	315
4.15.11	Report Unexpected Items.....	316
4.15.11.1	Solution .....	316
4.15.11.2	End-User Impact.....	319
4.15.11.3	Implementation .....	319
4.15.11.4	Reference Implementation.....	320
4.15.12	Issue Refund Now .....	320
4.15.12.1	Solution .....	320
4.15.12.2	End-User Impact.....	321

4.15.12.3	Implementation .....	321
4.15.12.4	Reference Implementation.....	321
4.15.13	Change Return Address .....	321
4.15.13.1	Solution .....	321
4.15.13.2	End-User Impact .....	322
4.15.13.3	Implementation .....	322
4.15.13.4	Reference Implementation.....	322
4.15.14	Change Service Appointment .....	322
4.15.14.1	Solution .....	323
4.15.14.2	End-User Impact .....	324
4.15.14.3	Implementation .....	324
4.15.14.4	Reference Implementation.....	324
4.15.15	Change Return Method.....	325
4.15.15.1	Solution .....	325
4.15.15.2	End User Impact .....	326
4.15.15.3	Implementation .....	327
4.15.15.4	Reference Implementation.....	327
4.15.16	Cancel Return Order .....	327
4.15.16.1	Solution .....	328
4.15.16.2	End-User Impact .....	329
4.15.16.3	Implementation .....	329
4.15.16.4	Reference Implementation.....	329
4.15.17	Payment Confirmation.....	330
4.15.17.1	Solution .....	330
4.15.17.2	End User Impact .....	330
4.15.17.3	Implementation .....	330
4.15.17.4	Reference Implementation.....	331
4.16	Customer Creation .....	331
4.16.1	Create Consumer Customer .....	331
4.16.1.1	Solution .....	331
4.16.1.2	End-User Impact .....	334
4.16.1.3	Implementation .....	334
4.16.1.4	Reference Implementation.....	335
4.16.2	Create Business Customer .....	335
4.16.2.1	Solution .....	335

4.16.2.2	End-User Impact .....	338
4.16.2.3	Implementation .....	339
4.16.2.4	Reference Implementation.....	339
4.17	Customer Details .....	339
4.17.1	Solution .....	339
4.17.2	End-User Impact.....	342
4.17.3	Implementation .....	342
4.17.4	Reference Implementation.....	342
4.18	Customer Inquiry .....	342
4.18.1	Solution .....	342
4.18.2	End-User Impact.....	343
4.18.3	Implementation .....	344
4.18.4	Reference Implementation.....	344
4.19	Customer Maintenance .....	344
4.19.1	Manage Contacts.....	344
4.19.1.1	Solution .....	344
4.19.1.2	End-User Impact.....	348
4.19.1.3	Implementation .....	348
4.19.1.4	Reference Implementation.....	348
4.19.2	Manage Business Customer Information .....	348
4.19.2.1	Solution .....	348
4.19.2.2	End-User Impact.....	349
4.19.2.3	Implementation .....	349
4.19.2.4	Reference Implementation.....	349
4.19.3	Manage Customer Addresses .....	349
4.19.3.1	Solution .....	349
4.19.3.2	End-User Impact.....	350
4.19.3.3	Implementation .....	350
4.19.3.4	Reference Implementation.....	350
4.19.4	Manage Payment Methods.....	350
4.19.4.1	Solution .....	350
4.19.4.2	End-User Impact.....	351
4.19.4.3	Implementation .....	351
4.19.4.4	Reference Implementation.....	352
4.19.5	Reset User Passwords.....	352

4.19.5.1	Solution .....	352
4.19.5.2	End-User Impact .....	353
4.19.5.3	Implementation .....	353
4.19.5.4	Reference Implementation.....	353
4.19.6	Customer Team Assignments .....	354
4.19.6.1	Solution .....	354
4.19.6.2	End-User Impact .....	354
4.19.6.3	Implementation .....	354
4.19.6.4	Reference Implementation.....	355
4.19.7	Customer User Assignments .....	355
4.19.7.1	Solution .....	355
4.19.7.2	End-User Impact .....	355
4.19.7.3	Implementation .....	356
4.19.7.4	Reference Implementation.....	356
4.19.8	Customer Self Assignments .....	356
4.19.8.1	Solution .....	356
4.19.8.2	End-User Impact .....	357
4.19.8.3	Implementation .....	357
4.19.8.4	Reference Implementation.....	357
4.19.9	Customer Notes .....	357
4.19.9.1	Solution .....	357
4.19.9.2	End-User Impact .....	359
4.19.9.3	Implementation .....	359
4.19.9.4	Reference Implementation.....	359
4.20	User Creation .....	359
4.20.1	Create a User .....	359
4.20.1.1	Solution .....	359
4.20.1.2	End-User Impact .....	361
4.20.1.3	Implementation .....	361
4.20.1.4	Reference Implementation.....	361
4.20.2	Modify a User .....	361
4.20.2.1	Solution .....	362
4.20.2.2	End-User Impact .....	362
4.20.2.3	Implementation .....	362
4.20.2.4	Reference Implementation.....	363

4.21	User Team Creation.....	363
4.21.1	Create a User Team.....	363
4.21.1.1	Solution.....	363
4.21.1.2	End-User Impact.....	364
4.21.1.3	Implementation.....	365
4.21.1.4	Reference Implementation.....	365
4.21.2	Manage a User Team.....	365
4.21.2.1	Solution.....	365
4.21.2.2	End-User Impact.....	366
4.21.2.3	Implementation.....	366
4.21.2.4	Reference Implementation.....	366
4.22	Pagination.....	366
4.22.1	Solution.....	367
4.22.2	End-User Impact.....	367
4.22.3	Implementation.....	368
4.22.4	Reference Implementation.....	368
4.23	Alert and Queue Management.....	369
4.23.1	Solution.....	369
4.23.2	End-User Impact.....	371
4.23.3	Implementation.....	371
4.23.4	Reference Implementation.....	373
4.24	Announcements.....	373
4.24.1	Solution.....	374
4.24.2	End-User Impact.....	375
4.24.3	Implementation.....	376
4.24.4	Reference Implementation.....	376
4.25	User Preferences.....	377
4.25.1	Solution.....	377
4.25.2	End-User Impact.....	378
4.25.3	Implementation.....	378
4.25.4	Reference Implementation.....	378
4.26	Viewing the Documentation Library.....	379
4.26.1	Solution.....	379
4.26.2	End-User Impact.....	379
4.26.3	Implementation.....	379

4.26.4	Reference Implementation.....	379
4.27	Deploying the Sterling Call Center Clients in Phases .....	380
4.27.1	Solution .....	380
4.27.2	End-User Impact .....	382
4.27.3	Implementation .....	383
4.27.4	Reference Implementation.....	383
4.28	Helpful Text Messages .....	384
4.28.1	Solution .....	384
4.28.2	End-User Impact .....	385
4.28.3	Implementation .....	385
4.28.4	Reference Implementation.....	386
4.29	Integrating with Business Center.....	386
4.29.1	Solution .....	386
4.29.2	End-User Impact .....	387
4.29.3	Implementation .....	387
4.29.4	Reference Implementation.....	387

## 5 Implementing the Sterling Store Features

5.1	Item Inquiry.....	389
5.2	Configured Items .....	389
5.3	Alternate Item Identifier.....	390
5.4	Item Display Options Configuration.....	390
5.5	Alternate Store Pickup Locations Search .....	390
5.6	Order Capture.....	390
5.7	Order Inquiry.....	390
5.8	Order Monitoring .....	391
5.9	Order Maintenance .....	391
5.9.1	Apply Hold .....	392
5.9.2	Resolve Holds .....	392
5.9.3	Change Payment Method .....	393
5.9.4	Add Multiple Items to an Order .....	393
5.9.5	Change Gift Options.....	393
5.9.6	Customer Appeasement .....	393
5.9.7	Open Box Items .....	393
5.9.7.1	Solution .....	393

5.9.7.2	End-User Impact .....	394
5.9.7.3	Implementation .....	394
5.9.7.4	Reference Implementation.....	395
5.9.8	Add Coupon.....	395
5.9.9	Cancel Order .....	395
5.9.10	Track an Item .....	396
5.9.11	Order Notes.....	397
5.9.12	Change Fulfillment Options .....	397
5.9.13	Fulfillment Summary .....	397
5.9.14	Delivery and Service Grouping .....	398
5.9.15	Change Service Instructions .....	398
5.9.16	Change Service Appointments.....	398
5.9.17	Reservations .....	398
5.9.18	Price Match .....	398
5.9.19	Competitor Search .....	399
5.9.20	Price Match Search .....	399
5.9.21	Create or Modify a Competitor Record .....	399
5.9.22	Create or Modify a Price Match Record .....	399
5.9.23	Increase Order Line Quantity .....	399
5.9.24	Schedule and Release an Order .....	400
5.9.25	View Procurement Orders .....	400
5.9.26	Change Order Address .....	400
5.9.27	Launch Application Console.....	400
5.9.28	Change an Item's Style .....	400
5.9.29	Reconfigure Bundle Items .....	401
5.9.30	Change Tax Options .....	401
5.9.31	Copy Order .....	401
5.9.32	E-Mail Order Information.....	401
5.9.33	Transfer Draft Order to Web .....	402
5.10	Payment Processing .....	402
5.11	Payment Inquiry .....	402
5.12	Return Order Capture .....	402
5.12.1	Create Return .....	403
5.12.2	Return Reason .....	403
5.12.2.1	All Items Are In Hand Rule .....	404

5.12.3	Fulfillment Summary.....	404
5.12.4	Payment Processing.....	405
5.13	Return Order Inquiry .....	405
5.14	Return Order Maintenance.....	405
5.14.1	Report Extra Items .....	406
5.14.2	Report Wrong Items .....	406
5.14.3	Report Unexpected Items .....	406
5.14.4	Issue Refund Now .....	406
5.14.5	Change Return Address .....	406
5.14.6	Cancel Return Order .....	407
5.15	Customer Creation .....	407
5.15.1	Create Consumer Customer .....	407
5.15.2	Create Business Customer .....	407
5.16	Customer Details.....	407
5.17	Customer Inquiry .....	408
5.18	Customer Maintenance.....	408
5.18.1	Manage Contacts.....	408
5.18.2	Manage Business Customer Information .....	408
5.18.3	Manage Customer Addresses .....	409
5.18.4	Manage Payment Methods .....	409
5.18.5	Reset User Passwords .....	409
5.18.6	Customer Team Assignments .....	409
5.18.7	Customer User Assignments .....	409
5.18.8	Customer Self Assignments .....	410
5.18.9	Customer Notes .....	410
5.19	User Creation .....	410
5.20	User Team Creation.....	410
5.21	Pagination .....	411
5.22	Alert and Queue Management .....	411
5.22.1	Solution .....	411
5.22.2	End-User Impact .....	412
5.22.3	Implementation .....	412
5.22.4	Reference Implementation.....	412
5.23	Announcements .....	412
5.24	User Preferences .....	412



5.25	Viewing the Documentation Library .....	413
5.26	Deploying Sterling Store Clients in Phases .....	413
5.27	Helpful Text Messages .....	413
5.28	Order Print Tasks .....	413
5.28.1	Create Order and Order Summary Prints .....	414
5.28.1.1	Solution .....	414
5.28.1.2	End-User Impact .....	414
5.28.1.3	Implementation .....	414
5.28.1.4	Reference Implementation.....	415
5.28.2	Change Order Print.....	415
5.28.2.1	Solution .....	415
5.28.2.2	End-User Impact .....	415
5.28.2.3	Implementation .....	415
5.28.2.4	Reference Implementation.....	416
5.28.3	Return or Exchange Order Prints .....	416
5.28.3.1	Solution .....	416
5.28.3.2	End-User Impact .....	416
5.28.3.3	Implementation .....	417
5.28.3.4	Reference Implementation.....	417
5.29	User Security.....	417
5.29.1	Solution .....	417
5.29.2	End-User Impact .....	418
5.29.3	Implementation .....	418
5.29.4	Reference Implementation.....	418
5.30	In-Store Pick-up Tasks.....	419
5.30.1	Shipment Inquiry .....	419
5.30.1.1	Solution .....	419
5.30.1.2	End-User Impact .....	420
5.30.1.3	Implementation .....	420
5.30.1.4	Reference Implementation.....	420
5.30.2	Print Pick Ticket .....	420
5.30.2.1	Solution .....	420
5.30.2.2	End-User Impact .....	421
5.30.2.3	Implementation .....	421
5.30.2.4	Reference Implementation.....	422

5.30.3	Backroom Pick .....	422
5.30.3.1	Solution .....	422
5.30.3.2	End-User Impact .....	425
5.30.3.3	Implementation .....	425
5.30.3.4	Reference Implementation.....	426
5.30.4	Undo Backroom Pick .....	426
5.30.4.1	Solution .....	426
5.30.4.2	End-User Impact .....	427
5.30.4.3	Implementation .....	427
5.30.4.4	Reference Implementation.....	427
5.30.5	Customer Pick.....	427
5.30.5.1	Solution .....	427
5.30.5.2	End-User Impact .....	430
5.30.5.3	Implementation .....	430
5.30.5.4	Reference Implementation.....	431
5.31	Integrating with Business Center .....	431
5.31.1	Solution .....	432
5.31.2	End-User Impact .....	432
5.31.3	Implementation .....	432
5.31.4	Reference Implementation.....	433

## 6 Summary of Components

6.1	Database Extensions for Sterling Call Center and Sterling Store.....	435
6.2	APIs and User Exits .....	443
6.3	Services.....	446
6.3.1	Order Maintenance Services.....	447
6.3.2	Order Pricing Services.....	447
6.3.3	Customer Services .....	448
6.3.4	Order Capture Services .....	448
6.3.5	Payment Processing Services .....	449
6.3.6	Order Printing Services .....	449
6.3.7	FTC Compliance Services.....	450
6.4	Monitor Events.....	451
6.5	XSL Files .....	451
6.6	Hold Types .....	453

6.7	Transactions.....	453
6.8	Time-Triggered Purge Transactions.....	454
6.8.1	Price Match Purge .....	454
6.8.2	Price Match History Purge.....	456
6.9	Events .....	458
6.9.1	Event Call Back Handlers.....	458
6.10	Agents.....	459

## **7 Performance Tuning**

7.1	Performance Recommendations .....	461
7.2	Enabling Reference Data Caching .....	462
7.3	Cached Data.....	462

## **A Localizing the Application**

A.1	User Interface Themes.....	465
A.2	Literals .....	465
A.2.1	Resource Bundles.....	466
A.3	Localizing Address and Store Node Information .....	468
A.4	Factory Setup Data .....	469

## **Index**



# Preface

---

This manual explains the Sterling Call Center™ and Sterling Store™ applications and the Reference Implementation provided along with them. This manual also contains information about the various processes and solutions offered by Sterling Call Center and Sterling Store.

## Intended Audience

This manual is intended to provide assistance to individuals who need to configure Sterling Call Center and Sterling Store and implement the features relevant to their business needs.

## Structure

This manual contains the following sections:

### **Chapter 1, "Introduction"**

This chapter explains the customer order management business challenges and how Sterling Call Center and Sterling Store assists you in solving them.

### **Chapter 2, "Process Types"**

This chapter explains the process types defined for Sterling Call Center and Sterling Store.

### **Chapter 3, "Extending and Customizing the Application"**

This chapter explains the process of extending and customizing Sterling Call Center and Sterling Store.

### **Chapter 4, "Implementing the Sterling Call Center Features"**

This chapter discusses how to implement the Sterling Call Center features.

### **Chapter 5, "Implementing the Sterling Store Features"**

This chapter discusses how to implement the Sterling Store features.

### **Chapter 6, "Summary of Components"**

This chapter discusses the new components available for the existing Selling and Fulfillment Foundation customers.

### **Chapter 7, "Performance Tuning"**

This chapter explains the steps you need to take to fine-tune Sterling Call Center and Sterling Store for optimal performance.

### **Appendix A, "Localizing the Application"**

This appendix explains the process of localization in Sterling Call Center and Sterling Store.

## **Documentation**

For more information about the Sterling Call Center™ and Sterling Store™ components, see the following manuals:

- *Sterling Call Center and Sterling Store: Release Notes*
- *Sterling Call Center and Sterling Store: Concepts*
- *Sterling Selling and Fulfillment Suite™: Applications Installation Guide*
- *Sterling Call Center and Sterling Store: Deployment Guide*
- *Sterling Selling and Fulfillment Suite™: Applications Reference Implementation Guide*
- *Sterling Call Center and Sterling Store: Implementation Guide*
- *Sterling Selling and Fulfillment Suite™: Applications Configuration Guide*
- *Sterling Call Center and Sterling Store: User Guide*
- *Sterling Call Center and Sterling Store: Upgrade Guide*
- *Sterling Call Center and Sterling Store: Business Intelligence Guide*

- *Sterling Call Center and Sterling Store: Javadocs*

For more information about the Selling and Fulfillment Foundation components, see the following manuals:

- *Selling and Fulfillment Foundation: Release Notes*
- *Selling and Fulfillment Foundation: Installation Guide*
- *Selling and Fulfillment Foundation: Upgrade Guide*
- *Selling and Fulfillment Foundation: Configuration Deployment Tool Guide*
- *Selling and Fulfillment Foundation: Performance Management Guide*
- *Selling and Fulfillment Foundation: High Availability Guide*
- *Selling and Fulfillment Foundation: System Management Guide*
- *Selling and Fulfillment Foundation: Localization Guide*
- *Selling and Fulfillment Foundation: Customization Basics Guide*
- *Selling and Fulfillment Foundation: Customizing APIs Guide*
- *Selling and Fulfillment Foundation: Customizing Console JSP Interface for End User Guide*
- *Selling and Fulfillment Foundation: Customizing the RCP Interface Guide*
- *Selling and Fulfillment Foundation: Customizing User Interfaces for Mobile Devices Guide*
- *Selling and Fulfillment Foundation: Customizing Web UI Framework Guide*
- *Selling and Fulfillment Foundation: Customizing Swing Interface Guide*
- *Selling and Fulfillment Foundation: Extending the Condition Builder Guide*
- *Selling and Fulfillment Foundation: Extending the Database Guide*
- *Selling and Fulfillment Foundation: Extending Transactions Guide*
- *Selling and Fulfillment Foundation: Using Sterling RCP Extensibility Tool Guide*
- *Selling and Fulfillment Foundation: Integration Guide*

- *Selling and Fulfillment Foundation: Product Concepts Guide*
- *Sterling Warehouse Management™ System: Concepts Guide*
- *Selling and Fulfillment Foundation: Application Platform Configuration Guide*
- *Sterling Distributed Order Management™: Configuration Guide*
- *Sterling Supply Collaboration: Configuration Guide*
- *Sterling Global Inventory Visibility™: Configuration Guide*
- *Catalog Management™: Configuration Guide*
- *Sterling Logistics Management: Configuration Guide*
- *Sterling Reverse Logistics™: Configuration Guide*
- *Sterling Warehouse Management System: Configuration Guide*
- *Selling and Fulfillment Foundation: Application Platform User Guide*
- *Sterling Distributed Order Management: User Guide*
- *Sterling Supply Collaboration: User Guide*
- *Sterling Global Inventory Visibility: User Guide*
- *Sterling Logistics Management: User Guide*
- *Sterling Reverse Logistics: User Guide*
- *Sterling Warehouse Management System: User Guide*
- *Selling and Fulfillment Foundation: Mobile Application User Guide*
- *Selling and Fulfillment Foundation: Business Intelligence Guide*
- *Selling and Fulfillment Foundation: Javadocs*
- *Sterling Selling and Fulfillment Suite™: Glossary*
- *Parcel Carrier: Adapter Guide*
- *Selling and Fulfillment Foundation: Multitenant Enterprise Guide*
- *Selling and Fulfillment Foundation: Password Policy Management Guide*
- *Selling and Fulfillment Foundation: Properties Guide*
- *Catalog Management: Concepts Guide*



- *Selling and Fulfillment Foundation: Pricing Concepts Guide*
- *Selling and Fulfillment Foundation: Setting Up Quotes in Distributed Order Management*
- *Sterling Sensitive Data Capture Server, Release 1.0: Configuration Guide*
- *Sterling Sensitive Data Capture Server, Release 1.0: PA-DSS Implementation Guide*
- *Selling and Fulfillment Foundation: Secure Deployment Guide*
- *Business Center: Item Administration Guide*
- *Business Center: Pricing Administration Guide*
- *Business Center: Customization Guide*
- *Business Center: Localization Guide*
- *Sterling Field Sales: Deployment Guide*
- *Sterling Field Sales: Implementation Guide*
- *Sterling Field Sales: Localization Guide*
- *Sterling Field Sales: User Guide*
- *Sterling Field Sales: Customization Guide*

For a description of the various documents in the Sterling Call Center and Sterling Store documentation set, see the Sterling Call Center and Sterling Store documentation home page at:

<INSTALL\_DIR>/documentation/YCD\_doc\_home.html

<INSTALL\_DIR> is the directory where Sterling Call Center and Sterling Store and the Selling and Fulfillment Foundation are installed.

## Conventions

The following conventions may be used in this manual:

Convention	Meaning
. . .	Ellipsis represents information that has been omitted.
< >	Angle brackets indicate user-supplied input.

Convention	Meaning
mono-spaced text	Mono-spaced text indicates a file name, directory path, attribute name, or an inline code example or command.
/ or \	Slashes and backslashes are file separators for Windows, UNIX, and Linux operating systems. The file separator for the Windows operating system is "\" and the file separator for UNIX and Linux systems is "/". The UNIX convention is used unless otherwise mentioned.
<INSTALL_DIR>	User-supplied location of the Selling and Fulfillment Foundation installation directory and Sterling Call Center and Sterling Store installation directory. This is only applicable for Release 8.0.
<ANALYTICS_HOME>	User-supplied location of the Sterling Call Center and Sterling Store: Business Intelligence installation directory. <b>Note:</b> This convention is used only in the <i>Sterling Call Center and Sterling Store: Business Intelligence Guide</i> .
<COGNOS_HOME>	User-supplied location of the Cognos installation directory. <b>Note:</b> This convention is used only in the <i>Sterling Call Center and Sterling Store: Business Intelligence Guide</i> .

**Notes:** The Sterling Call Center and Sterling Store documentation set uses the following conventions in the context of the product name:

- Sterling Customer Order Management PCA is used for Release 7.5 and earlier.
- Sterling Call Center and Sterling Store is used for Release 8.0.

The Selling and Fulfillment Foundation documentation set uses the following conventions in the context of the product name:

- Yantra is used for Release 7.7 and earlier.
- Sterling Supply Chain Applications is used for Releases 7.9 and 7.11.
- Sterling Multi-Channel Fulfillment Solution is used for Releases 8.0 and 8.2.
- Selling and Fulfillment Foundation for Release 8.5.

# Introduction

---

Sterling Call Center and Sterling Store provides solutions to the challenges found in typical customer order management business scenarios.

Sterling Call Center and Sterling Store presents multi-channel order management solutions by providing:

- A user interface to handle call center order modifications.
- A user interface to handle store walk-in order modifications.



## Process Types

---

An order goes through a wide range of statuses throughout its fulfillment cycle. An order is processed by transactions that perform modifications to the order and its related entities such as shipments, invoices, and returns. These transactions also determine how an order moves from one status to the next. Additionally, an order that is in a particular status can go through a condition to determine which transaction should process it next.

An order's flow throughout its fulfillment cycle is represented graphically by a pipeline. The pipeline determines the statuses that an order can be in, the transactions that process it, and the conditions it must go through in order to be fulfilled. As part of the reference implementation, a pipeline condition is set up, utilizing the application's pipelines for the following process types:

- [Order Fulfillment](#)
- [Outbound Shipment](#)
- [Reverse Logistics](#)

For more information about defining process type pipelines, see the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

### 2.1 Order Fulfillment

The Sterling Call Center and Sterling Store Sales Order Fulfillment pipeline is used to process orders of the Sales Order document type within Sterling Call Center and Sterling Store.

Orders in this pipeline go through the following stages:

- [Create](#)
- [Schedule](#)
- [Release](#)
- [Ship](#)
- [Return](#)
- [Close](#)
- [Purge](#)

### 2.1.1 Create

The first step in fulfilling an order is capturing it. Orders can be captured by using either the Create Order task of the Sterling Call Center and Sterling Store application, or from a Web portal or an external order entry system. The data of the order is passed to the createOrder API and processed through the Create Draft Order and Create Order transactions. Both transactions have the same end results because the order is moved to the Created, Reserved, or Back Ordered status before it is picked up by the Schedule transaction. However, using the Create Draft Order transaction places the order into either of the two intermediate statuses, Draft Order Created or Draft Order Reserved, where most of the order's attributes can still be modified.

After the order is ready to be processed, it can be confirmed through the Confirm Draft Order transaction.

The next step in fulfilling the order is scheduling it for delivery.

### 2.1.2 Schedule

The scheduling process schedules an order so that it can be delivered to the customer. Important shipment attributes such as delivery dates and ship nodes are determined at this point.

The Schedule transaction attempts to process orders that are in the Reserved, Back Ordered, Unscheduled, and Created statuses. If the orders that are in the Back Ordered or Reserved status cannot be processed by the `Schedule` transaction, they wait for a predefined time interval before being reprocessed. For more information about

transactions, see the *Sterling Distributed Order Management: Configuration Guide*.

If the orders that are in the Created status cannot be scheduled because of insufficient inventory, they are moved to the Back Ordered status and wait to be reprocessed.

If the Schedule transaction is successful, but the specified ship node on the order line does not have enough inventory for the shipment, and needs to procure its inventory from another node, the creation of a chained order is required. The order is moved to the Awaiting Chained Order Creation status, where after being processed by the Chained Order Create transaction, it waits to be processed for shipment.

If the ship node is a drop-ship node, the next step for the order is creating a chained order. If chained order creation is not required, the order is moved to the Scheduled status.

### 2.1.3 Release

After Sterling Call Center and Sterling Store has identified one or more ship nodes that can be utilized to ship the ordered merchandise, the corresponding nodes are notified with all the relevant information necessary to send the shipment. This is the Release process.

The Release transaction takes the orders that are in the Scheduled status and sends the notification to the nodes. Occasionally, the Schedule transaction's inventory picture and the actual inventory picture in the nodes can get out of synch. When this happens, the Release transaction sends a notification to a node asking for inventory that does not actually exist. The node then notifies Sterling Call Center and Sterling Store that it cannot fulfill the order because of insufficient inventory, and the order gets backordered from the node. The order is then moved to the Back Ordered From Node status, and waits to be reprocessed by the Schedule transaction at a later time.

If the node has successfully received the release and has the necessary inventory, the order can be shipped.

### 2.1.4 Ship

Shipment-related transactions use the information included in the release sent to the warehouse to plan for shipments as efficiently as possible.

When several shipments are going to the same location, and can be grouped within the same loads, they can be consolidated together to minimize the cost while still meeting the promised delivery dates. If shipment consolidation is required, shipments must be consolidated together before the actual shipment, as represented in the pipeline by the Consolidate To Shipment transaction.

Orders can be included in or removed from existing shipments before the actual shipment occurs using the Include Order In Shipment and Remove Order From Shipment transactions respectively.

At this point, an order can be considered complete, and, over time, is closed and purged from the system. However, if the customer decides to return the received merchandise, the order's life cycle continues.

### 2.1.5 Return

After a customer receives a shipment, they may find that there is something wrong with the merchandise. For example, some items in the order may be the wrong color, or may have been damaged in transit. The Include In Return transaction creates a return order that must be shipped back to a warehouse to be examined.

After a return order is created, it must be received and processed by the warehouse. The return order's life cycle is handled by the Reverse Logistics pipeline. Based on the status of the return order, the Sterling Call Center and Sterling Store Sales Order Fulfillment pipeline updates the status of the order.

Therefore, when a return order is received, the order moves to the Return Received status through the Receive Return listener transaction.

Similarly, if the return order is cancelled in the Reverse Logistics pipeline, the order moves back to the Shipped status through the Remove From Return listener transaction.

After the return order is received, if it is unreceived and sent back to the customer, the return order is placed back in the Return Created status through the Unreceive Return listener transaction.

### 2.1.6 Close

After an order moves to the Shipped, Cancelled, or Return Received status, the order can be closed by the Close Order transaction. By



default, closed orders cannot be searched for in the Application Console unless it is explicitly specified in the Search screen.

### 2.1.7 Purge

Purging is the process where the records pertaining to an order are moved from the regular database tables to the history tables. If orders have been in a final status such as Shipped, Return Received or Cancelled for a specific time, they are purged. History orders can be reopened at any time, but by default, cannot be searched for in the Application Console unless explicitly specified in the Search screen.

## 2.2 Outbound Shipment

The Sterling Call Center and Sterling Store Outbound Shipment pipeline is used to process the outbound shipments within Sterling Call Center and Sterling Store.

Shipments in this pipeline go through the following stages:

- [Create](#)
- [Backroom Pick](#)
- [Confirm](#)
- [Create Invoice](#)
- [Close](#)

### 2.2.1 Create

The outbound shipment process starts when an order line is released, and a shipment is created for the released order line. The Create Shipment transaction is used to create a shipment for the order lines that have been released.

### 2.2.2 Backroom Pick

After a shipment is created, the Process Shipment for Backroom Pick transaction picks up the shipment and moves it to either the Ready For Backroom Pick status or the Ready For Customer status depending on whether the store is configured for backroom pick. If the shipment line is in the Ready For Backroom Pick status, it is processed by the Store

Backroom Pick transaction and moved to the Ready For Customer status. The backroom pick process for a shipment line can be undone using the Undo Backroom Pick transaction, which moves the shipment line to the Ready For Backroom Pick status.

### 2.2.3 Confirm

After a shipment moves to the Ready For Customer status, the Confirm Shipment transaction processes the shipment and moves it to the Shipment Shipped status.

### 2.2.4 Create Invoice

After a shipment is shipped to a customer, an invoice must be created for the shipment. This is achieved by using the Create Shipment Invoice transaction, which processes the shipments that are in the Shipment Shipped status. After the invoice is created, the shipment is moved to the Shipment Invoiced status.

### 2.2.5 Close

After a shipment is moved to the Shipment Invoiced or the Shipment Cancelled status, it can be closed using the Close Shipment transaction.

## 2.3 Reverse Logistics

The Sterling Call Center and Sterling Store Reverse Logistics pipeline is used to process the return orders within Sterling Call Center and Sterling Store.

The return orders in this pipeline go through the following stages:

- [Create](#)
- [Authorize](#)
- [Schedule](#)
- [Receive](#)
- [Close](#)
- [Purge](#)

### 2.3.1 Create

The first step in fulfilling a return order is capturing it. Return orders can be captured either by using the Create Return task of the Sterling Call Center and Sterling Store application, or from a Web portal, or an external order entry system. The data of the return order is passed to the createOrder API and processed through the Create Draft Return and Create Return transactions. Both the transactions move the return order to the Created status before the return order is picked up by the Schedule transaction.

### 2.3.2 Authorize

After a return order is created, the Authorize Return transaction picks up the return order and moves it to either the Authorized status or the Not Authorized status. If the return order is moved to the Not Authorized status, it is purged using the Purge Return transaction. If the return order is moved to the Authorized status, it can be scheduled.

### 2.3.3 Schedule

After a return order is authorized, the Schedule transaction picks it up and moves it to the Released status. This is when the return order is released to the receiving node.

### 2.3.4 Receive

Receiving refers to the process of receiving returned items. Depending on the return fulfillment method, a return order that is in the Released status is processed by the Include Order in Shipment transaction, the Deliver Order transaction, or the Receive Return transaction.

If the return order is picked up by the Include Order in Shipment transaction, the order is moved to the Included in Shipment status, after which it is picked up by the Ship Shipment transaction and is passed through the Return Required? condition. If the value of the Return Required? condition is "true", the return order is moved to the Awaiting Invoice status. If the value of the Return Required? condition is "false", the return order is moved to the Shipped status, after which it is processed by the Deliver Order transaction, which moves the order to the Order Picked Up status. Finally, the return order is picked up by the

Receive Return transaction, which moves it to either the Received status or the Received As Components status.

### 2.3.5 Close

After a return order is received in the return node, it is processed by the Close Receipt transaction. This transaction moves the return order to the Receipt Closed status, after which the return order is invoiced, and finally closed, using the Close Return transaction.

---

---

**Note:** A return order that is in the Received As Components status can be closed using the Close Return transaction.

---

---

### 2.3.6 Purge

Purging is the process where the records pertaining to a return order are moved from the regular database tables to the history tables. After a return order has been in the Return Invoiced status or the Received As Components status for a specific time, it is purged.

## Extending and Customizing the Application

---

The user interfaces within Sterling Call Center and Sterling Store can be customized to meet your specific business needs. Tools are provided to help you make customizations for the following:

- [Order Entry Screens Customization](#)
- [Add Lines To Order Screens Customization](#)
- [Extending Actions](#)
- [Shared Tasks](#)
- [User Interface Customization](#)
- [Address Panel Customization](#)
- [Opening New Related Tasks in the Order Editor](#)
- [Opening the Screens of the Sterling Call Center and Sterling Store Application in an Editor](#)
- [Opening the Price Match Wizard in an Editor](#)

### 3.1 Order Entry Screens Customization

This section explains how to:

- Replace the existing order entry screens with customized screens.
- Customize the create order navigation panel.

### Replacement of the Existing Order Entry Screens with Customized Screens

You can replace the following order entry screens with customized screens:

- Customer Identification: Shipping Address screen—This screen creates or updates an order with the shipping address by calling the createOrder or changeOrder API respectively.
- Customer Identification: Billing Address screen—This screen updates an order with the billing address by calling the changeOrder API.
- Order Line Entry screen—This screen creates or updates an order with new order lines by calling the createOrder or changeOrder API respectively.
- Payment Confirmation: Simple Payment Confirmation screen—This screen updates the order with new payment methods and the amount to be charged against each payment method. You can confirm the order by calling the confirmDraftOrder API.
- Payment Confirmation: Return and Exchange Payment Confirmation screen—This screen updates an order with new payment methods and the amounts to be charged against each payment method for return and exchange orders. You can confirm the exchange order by calling the confirmDraftOrder API.

### Navigation Panel (Create Order) Customization

Sterling Call Center and Sterling Store enables you to customize the Create Order Navigation panel in an extension plug-in. The extension plug-in must register a new metadata XML with the Sterling Call Center and Sterling Store plug-in.

#### Metadata XML Structure

The metadata XML must adhere to the following structure:

```
<?xml version="1.0" encoding="UTF-8"?>
<BreadCrumbs>
  <WizardEntity Sequence="" id="" Impl="" Category=""/>
</BreadCrumbs>
```

You can register the metadata XML during the start up of the extension plug-in by invoking the following static method:

```
YCDOrderEntryUtils.registerNavigationPanelMetaData(Element
navigationPanelMetaDataToRegister)
```

**Note:** The Category specified in the metadata XML is a bundle key, the value of which is displayed as a label for the bread crumb. If this bundle key value comprises of multiple words, it is recommended to use new line characters in the bundle key value in order to avoid the label from being truncated or blurred.

## 3.2 Add Lines To Order Screens Customization

This section explains how to customize the add lines to order navigation panel.

### Navigation Panel (Add Lines To Order) Customization

Sterling Call Center and Sterling Store enables you to customize the Add Lines To Order Navigation panel in an extension plug-in. The extension plug-in must register a new metadata XML with the Sterling Call Center and Sterling Store plug-in.

You can register the metadata XML during the start up of the extension plug-in by invoking the following static method:

```
YCDOrderEntryUtils.registerNavigationPanelMetaDataForAddLinesT
ask(ElementnavigationPanelMetaDataToRegister)
```

The metadata XML must adhere to the structure specified in [Metadata XML Structure](#).

## 3.3 Extending Actions

In order to extend the actions in the Order Search, Item Search and Appointment screens, in Sterling Call Center and Sterling Store, the Rich Client Platform provides the following action classes:

- Order Search screen

```
com.yantra.pca.ycd.rcp.tasks.orderSearch.actions.YCDViewOrd
erDetailsAction
```

- Item Search screen

```
com.yantra.pca.ycd.rcp.tasks.itemSearch.actions.YCDItemView
Details
```

- Appointment screen

```
com.yantra.pca.ycd.rcp.tasks.common.delivery.actions.YCDApp
ointmentPageRefreshCalendarAction
```

On invoking the above action, the calendar on the Appointment screen is reloaded. This action considers the input namespace "WorkOrderAppointmentOptionsInput" when fetching the capacity information for the Appointment Calendar.

These action classes are not displayed on the Rich Client Platform Extensibility Tool. For more information about customizing screens, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

## 3.4 Shared Tasks

This section explains the various shared tasks that are registered with Sterling Call Center and Sterling Store. For more information about shared tasks, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

### 3.4.1 Address Panel Shared Task

Sterling Call Center and Sterling Store provide the Address Verification screen as a shared task. This shared task is used to open the Address Verification screen as a pop-up window or dialog box in any user interface (UI) that references Sterling Call Center and Sterling Store.

The following is the input XML for the Address Panel task:

```
<PersonInfo AVSMode="" AddressID="" AddressLine1="" AddressLine2=""
AddressLine3="" AddressLine4="" AddressLine5="" AddressLine6=""
IsCommercialAddress="" AlternateEmailID="" Beeper="" City="" Company=""
Country="" DayFaxNo="" DayPhone="" Department="" EnterpriseCode="" EMailID=""
EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl="" JobTitle="" LastName=""
Latitude="" Longitude="" MiddleName="" MobilePhone="" OtherPhone="" PersonID=""
PersonInfoKey="" State="" Suffix="" Title="" ZipCode="" />
```

The following is the output XML for the Address Panel task:

```
<PersonInfo AVSMode="" AddressID="" AddressLine1="" AddressLine2=""
AddressLine3="" AddressLine4="" AddressLine5="" AddressLine6=""
```



```
IsCommercialAddress="" AlternateEmailID="" Beeper="" City="" Company=""
Country="" DayFaxNo="" DayPhone="" Department="" EMailID="" EveningFaxNo=""
EveningPhone="" FirstName="" HttpUrl="" JobTitle="" LastName="" Latitude=""
Longitude="" MiddleName="" MobilePhone="" OtherPhone="" PersonID=""
PersonInfoKey="" State="" Suffix="" Title="" ZipCode="" />
```

The name space used for the output XML is "PersonInfo".

You can define the titles as common codes in the Channel Applications Manager by selecting Define Customers > Define Titles. The titles must be defined at the enterprise level. The getCommonCodeList API is called to retrieve the defined titles and populate the Title drop-down list. If titles are not defined, the Title drop-down list is not displayed.

### 3.4.2 Order Notes Shared Task

Sterling Call Center and Sterling Store provide the Order Notes screen as a shared task. This shared task is used to open the Order Notes screen as a pop-up window or dialog box in any UI that references Sterling Call Center and Sterling Store.

The following is the input XML for the Order Notes task:

```
<Order OrderHeaderKey="" Status="" />
```

The following is the output XML for the Order Notes task:

```
<Notes>
  <Note ContactReference="" ContactType="" ContactUser=""
    NoteText="" Priority="" ReasonCode="" />
</Notes>
```

The name space used for the output XML is "Note".

### 3.4.3 Service Appointment Calendar Shared Task

Sterling Call Center and Sterling Store provide the Service Appointment Calendar screen as a shared task. This shared task is used to view the calendar or select an appointment from the calendar.

The following is the input XML for the Service Appointment Calendar task:

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Change Delivery Appointment Shared Task Input XML-->
<WorkOrder EnterpriseCode="" Timezone="" WorkOrderKey=""
```

```

PreferredResourcePoolId="" PreferredServiceResourceId="" CurrentApptSeq="">
  <WorkOrderAppointments>
    <WorkOrderAppointment PromisedApptStartDate="" PromisedApptEndDate=""
      ApptSeq="" />
  </WorkOrderAppointments>
  <SuggestedAppointment ApptStartTimeStamp="" ApptEndTimeStamp="" />
</WorkOrder>

```

The following is the output XML for the Service Appointment Calendar task:

```

<?xml version="1.0" encoding="UTF-8"?>
<!--Change Delivery Appointment Shared Task Output XML-->
<SuggestedAppointment ApptEndTimeStamp="" ApptStartTimeStamp="" />

```

The name space used for the output XML is "CurrentAppointment".

### 3.4.4 Add Payment Method Shared Task

Sterling Call Center and Sterling Store provide the Add Payment Method screen as a shared task. This shared task is used to open the Add Payment Method screen as a pop-up window in any UI that references Sterling Call Center and Sterling Store.

Following is the input XML for the Add Payment Method task:

```

<PaymentMethod Currency="" OrganizationCode="">
  <OrderBillToAddress AddressLine1="" AddressLine2="" AddressLine3=""
    AddressLine4="" AddressLine5="" AddressLine6=""
    AlternateEmailID="" Beeper="" City="" Company="" Country=""
    DayFaxNo="" DayPhone="" Department="" EMailID=""
    EveningFaxNo="" EveningPhone="" FirstName="" JobTitle=""
    LastName="" MiddleName="" MobilePhone="" OtherPhone=""
    PersonID="" PersonInfoKey="" State="" Suffix="" Title=""
    ZipCode="" />
</PaymentMethod>

```

Following is the output XML for the Add Payment Method task:

```

<Event Action="">
  <PaymentMethod DisplayPaymentReferencel="" PaymentType=""
    MaxChargeLimit="" PaymentReferencel="" PaymentReference2=""
    PaymentReference3="" SaveToCustomer="" NewPaymentMethod=""
    CreditCardExpMonth="" CreditCardExpYear="" CreditCardName=""
    CreditCardNo="" CreditCardType="" FirstName="" LastName=""

```

```

MiddleName="" UseDefaultAddress="" CreditCardExpDate=""
AccountLimit="" AccountLimitCurrency="" CustomerAccountNo=""
CustomerPONo="" DisplayCustomerAccountNo="" SvcNo=""
TempPaymentReferencel="" UnlimitedCharges="" DisplayCreditCardNo="">
  <PersonInfoBillTo AddressLine1="" AddressLine2="" AddressLine3=""
    AddressLine4="" AddressLine5="" AddressLine6="" AlternateEmailID=""
    Beeper="" City="" Company="" Country="" Createprogid="" Createts=""
    Createuserid="" DayFaxNo="" DayPhone="" Department="" EMailID=""
    ErrorTxt="" EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl=""
    IsAddressVerified="" JobTitle="" LastName="" Lockid="" MiddleName=""
    MobilePhone="" Modifyprogid="" Modifyts="" Modifyuserid=""
    OtherPhone="" PersonID="" PersonInfoKey="" PreferredShipAddress=""
    State="" Suffix="" Title="" UseCount="" VerificationStatus=""
    ZipCode="" isHistory=""/>
</PaymentMethod>
</Event>

```

### 3.4.5 Order Summary Shared Task

Sterling Call Center and Sterling Store provide the Order Summary wizard as a shared task. This shared task is used to open the Order Summary screen as a pop-up window in any UI that references Sterling Call Center and Sterling Store.

Following is the input XML for the Order Summary task:

```

<Order BillToID="" BuyerOrganizationCode="" CustomerFirstName=""
  CustomerLastName="" DocumentType="" EnterpriseCode="" OrderHeaderKey=""
  OrderNo="" SellerOrganizationCode="" isHistory="">
  <PriceInfo Currency=""/>
</Order>

```

The Order Summary task returns null as the output.

---



---

**Note:** Ensure that the OrderHeaderKey parameter is present in the input XML.

---



---

The Sterling Call Center and Sterling Store applications enable you to open the Order Summary screen in an editor by using a utility method that is explained in [Section 3.8.1, "Opening the Order Summary Screen in an Editor"](#).

### 3.4.6 Return Search Shared Task

Sterling Call Center and Sterling Store provide the Return Search wizard as a shared task. This shared task is used to open the Return Search screen as a pop-up window in any UI that references Sterling Call Center and Sterling Store.

Following is the input XML for the Return Search task:

```
<ReturnOrder DocumentType="" />
```

The Return Search task returns null as the output.

The Sterling Call Center and Sterling Store applications enable you to open the Return Search screen in an editor by using a utility method that is explained in [Section 3.8.2, "Opening the Return Search Screen in an Editor"](#).

### 3.4.7 Cancel Order Shared Task

Sterling Call Center and Sterling Store provide the Cancel Order wizard as a shared task. This shared task is used to open the Cancel Order screen as a pop-up window in any UI that references Sterling Call Center and Sterling Store.

Following is the input XML for the Cancel Order task:

```
<Order BillToID="" BuyerOrganizationCode="" ChangeTaskName=""
  CustomerFirstName="" CustomerLastName="" DocumentType="" EnterpriseCode=""
  OrderHeaderKey="" OrderNo="" SellerOrganizationCode="" isHistory="">
  <PriceInfo Currency="" />
</Order>
```

The Cancel Order task returns null as the output.

**Note:** Ensure that the OrderHeaderKey parameter is present in the input XML.

The Sterling Call Center and Sterling Store applications enable you to open the Cancel Order screen in an editor by using a utility method that is explained in [Section 3.8.3, "Opening the Cancel Order Screen in an Editor"](#).

### 3.4.8 Create Alert Shared Task

Sterling Call Center and Sterling Store provide the Create Alert screen as a shared task. This shared task is used to open the Create Alert screen as a pop-up window in any UI that references Sterling Call Center and Sterling Store.

Following is the input XML for the Create Alert task:

```
<Inbox AlertCreate="" Title=""/>
```

The Create Alert task returns null as the output.

The Sterling Call Center and Sterling Store applications enable you to open the Create Alert screen in an editor by using a utility method that is explained in [Section 3.8.4, "Opening the Create Alert Screen in an Editor"](#).

### 3.4.9 Alert List Shared Task

Sterling Call Center and Sterling Store provide the Alert List screen as a shared task. This shared task is used to open the Alert List screen as a pop-up window in any UI that references Sterling Call Center and Sterling Store.

Following is the input XML for the Alert List task:

```
<Inbox Status="" StatusQryType="" SubscribedQueues=""/>
```

---

**Note:** If a null input is passed to the Alert List task, this task opens a blank Alert List screen.

---

The Alert List task returns null as the output.

The Sterling Call Center and Sterling Store applications enable you to open the Alert List screen in an editor by using a utility method that is explained in [Section 3.8.5, "Opening the Alert List Screen in an Editor"](#).

### 3.4.10 Message Dialog Box Shared Task

Sterling Call Center and Sterling Store provide the message dialog box as a shared task. This shared task is used to display error messages,

warning messages, confirmation messages, and messages with appropriate buttons.

The following is the input XML for the message dialog box:

```
<OptionDialog TitleKey="" DescriptionKey="" DefaultOption="" DialogImage="">
  <OptionTextList>
    <OptionText OptionTextKey="" />
  </OptionTextList>
</OptionDialog>
```

The following is the output XML for the message dialog box:

```
<SelectedOption SelectedOptionKey="" SelectedOptionIndex="" />
```

### 3.4.11 Item Details Shared Task

Sterling Call Center and Sterling Store provide the Item Details wizard as a shared task. This shared task is used to open the Item Details screen as a pop-up window in any UI that references Sterling Call Center and Sterling Store.

The following is the input XML template for the Item Details task:

```
<Item AuthenticationKey="" CallingOrganizationCode="" CanUseAsServiceTool=""
  Currency="" DemandType="" DisplayLocalizedFieldInLocale=""
  ExcludeChildItemsOfModelItems="" ExcludeItemsInPriceList=""
  ExecuteQueries="" FulfillmentType="" GetAvailabilityFromCache=""
  GetUnpublishedItems="" GlobalItemID="" IgnoreInvalidItems=""
  IgnoreIsSoldSeparately="" IgnoreOrdering=""
  InheritAttributesFromClassification="" IsForOrdering="" ItemGroupCode=""
  ItemID="" MaximumRecords="" OrganizationCode="" PriceProgramKey=""
  PriceProgramName="" PricingDate="" PurposeForDisplayClassification=""
  ShowAttributesWithDifferentValuesOnly="" ShowFinalSupersededItemOnly=""
  ShowKeyItemAttributesOnly="" StartItemKey="" SubCatalogOrganizationCode=""
  UnitOfMeasure="" UserOrganizationCode="">
  <BarCode BarCodeData="" BarCodeType="">
    <ContextualInfo OrganizationCode="" />
  </BarCode>
  <CustomerInformation CustomerID="" CustomerLevel="" CustomerType=""
    Region="" RelationshipType="" Vertical="" />
  <ShipToAddress AddressLine1="" AddressLine2="" AddressLine3=""
    AddressLine4="" AddressLine5="" AddressLine6="" AlternateEmailID=""
    Beeper="" City="" Company="" Country="" DayFaxNo="" DayPhone=""
    Department="" EMailID="" ErrorTxt="" EveningFaxNo="" EveningPhone=""
    FirstName="" HttpUrl="" IsCommercialAddress="" JobTitle="" LastName=""
```

```

        MiddleName="" MobilePhone="" OtherPhone="" PersonID="" PersonInfoKey=""
        PreferredShipAddress="" State="" Suffix="" TaxGeoCode="" Title=""
        UseCount="" VerificationStatus="" ZipCode=""/>
<ShipNodes>
    <ShipNode Node=""/>
</ShipNodes>
<ItemAssociationTypeList>
    <ItemAssociationType Type=""/>
</ItemAssociationTypeList>
<PrimaryInformation AssumeInfiniteInventory="" ColorCode=""
    ConfiguredModelKey="" CostCurrency="" CountryOfOrigin=""
    DefaultProductClass="" Description="" IsAirShippingAllowed=""
    IsConfigurable="" IsDeliveryAllowed="" IsForwardingAllowed=""
    IsFreezerRequired="" IsHazmat="" IsModelItem="" IsParcelShippingAllowed=""
    IsPickupAllowed="" IsPreConfigured="" IsProcurementAllowed=""
    IsReturnService="" IsReturnable="" IsShippingAllowed=""
    IsStandaloneService="" IsSubOnOrderAllowed="" ItemType="" KitCode=""
    ManufacturerItem="" ManufacturerItemDesc="" ManufacturerName=""
    MasterCatalogID="" MaxOrderQuantity="" MinOrderQuantity=""
    MinimumCapacityQuantity="" ModelItemUnitOfMeasure="" NumSecondarySerials=""
    PricingQuantityConvFactor="" PricingQuantityStrategy="" PricingUOM=""
    PricingUOMStrategy="" ProductLine="" RequiresProdAssociation=""
    ServiceTypeID="" ShipmentConsolidationWindow="" ShortDescription=""
    SizeCode="" Status="" TaxableFlag="" UnitCost="" UnitHeight=""
    UnitHeightUOM="" UnitLength="" UnitLengthUOM="" UnitWeight=""
    UnitWeightUOM="" UnitWidth="" UnitWidthUOM=""/>
<InventoryParameters ATPRule="" AdvanceNotificationTime=""
    CapacityQuantityStrategy="" DefaultExpirationDays=""
    FutureSafetyFactorPercentage="" InventoryMonitorRule="" IsFifoTracked=""
    IsItemBasedAllocationAllowed="" IsSerialTracked="" LeadTime=""
    MaximumNotificationTime="" MinNotificationTime=""
    NodeLevelInventoryMonitorRule="" OnhandSafetyFactorPercentage=""
    OnhandSafetyFactorQuantity="" ProcessingTime="" SerializedFlag=""
    TagControlFlag="" TimeSensitive="" UseUnplannedInventory=""/>
<ClassificationCodes CommodityCode="" ECCNNo="" HarmonizedCode=""
    HazmatClass="" Model="" NAICSCode="" NMFCClass="" NMFCCode=""
    OperationalConfigurationComplete="" PickingType="" PostingClassification=""
    Schedule_B_Code="" StorageType="" TaxProductCode="" UNSPSC=""/>
<ItemAliasList>
    <ItemAlias AliasName="" AliasValue=""/>
</ItemAliasList>
<ItemExclusionList>
    <ItemExclusion ExclusionCode=""/>
</ItemExclusionList>
<AdditionalAttributeList>

```

```

        <AdditionalAttribute DoubleValue="" IntegerValue="" Name="" Value=""/>
    </AdditionalAttributeList>
    <CategoryFilter CategoryPath="" OrganizationCode=""/>
    <Components>
        <Component ComponentItemID="" ComponentItemKey=""
            ComponentOrganizationCode="" ComponentUnitOfMeasure=""/>
    </Components>
    <LanguageDescriptionList>
        <LanguageDescription LocaleCode=""/>
    </LanguageDescriptionList>
    <InventoryTagAttributes BatchNo="" ItemTagKey="" LotAttribute1=""
        LotAttribute2="" LotAttribute3="" LotKeyReference="" LotNumber=""
        ManufacturingDate="" RevisionNo=""/>
    <ItemAttributeGroupTypeList>
        <ItemAttributeGroupType ClassificationPurposeCode=""
            ItemAttributeGroupType=""/>
    </ItemAttributeGroupTypeList>
    <OrderBy>
        <Attribute Desc="" Name=""/>
    </OrderBy>
</Item>

```

The Item Details task returns null as the output.

The Sterling Call Center and Sterling Store applications enable you to open the Item Details screen in an editor by using a utility method that is explained in [Section 3.8.7, "Opening the Item Details Screen in an Editor"](#).

### 3.4.12 Add Modify Charges Shared Task

Sterling Call Center and Sterling Store provide the Add Modify Charges wizard as a shared task. This shared task is used to open the Add Modify Charges screen as a pop-up window in any UI that references Sterling Call Center and Sterling Store.

The following is the input XML template for the Add Modify Charges task:

```

<Order ApplicationCode="" ApplicationVersion=""
    DisplayTransactionPrice="" DocumentType="" EnterpriseCode=""
    LastOrderLineKey="" LocalShipNode="" MaximumRecords="" OrderHeaderKey=""
    OrderNo="" SortingType="" StatusModificationType="">
    <Modifications>
        <Modification ModificationType=""/>
    </Modifications>
</Order>

```



```

</Modifications>
<SpecificOrderLines>
  <OrderLine GetBundleComponents="" OrderLineKey="" PrimeLineNo=""
    SubLineNo="">
    <Modifications>
      <Modification ModificationType=""/>
    </Modifications>
  </OrderLine>
</SpecificOrderLines>
<OrderLines>
  <OrderLine>
    <Modifications>
      <Modification ModificationType=""/>
    </Modifications>
  </OrderLine>
</OrderLines>
</Order>

```

The Add Modify Charges task returns null as the output.

### 3.4.13 Customer Details Shared Task

Sterling Call Center and Sterling Store provide the Customer wizard as a shared task. This shared task is used to open the Customer Details screen as a pop-up window in any UI that references Sterling Call Center and Sterling Store. It can be used for both consumer customers as well as business customers.

The following is the input XML template for the Customer Details task:

```

<Customer CustomerKey="" BuyerOrganizationCode="" CallingOrganizationCode=""
  CanConsumeSupplementalCapacity="" CommitmentTime=""
  CustomerAssignedVendorID="" CustomerClassificationCode="" CustomerID=""
  CustomerLevel="" CustomerType="" ExternalCustomerID="" FuncAckTime=""
  OrganizationCode="" ParentCustomerKey="" RegisteredDate=""
  RelationshipType="" RootCustomerKey="" SendsAsn="" SendsCommitment=""
  SendsFuncAck="" SlotPreferenceType="" Status=""
  Vertical="">
  <ParentCustomer CustomerID="" OrganizationCode=""/>
  <BuyerOrganization AccountWithHub="" AuthorityType=""
    BillingAddressKey="" CatalogOrganizationCode="" ContactAddressKey=""
    CorporateAddressKey="" CreatorOrganizationKey="" DunsNumber=""
    InventoryOrganizationCode="" InventoryPublished="" IsBuyer=""
    IsHubOrganization="" IsLegalEntity="" IsNode="" IsSourcingKept=""
    IssuingAuthority="" ItemXrefRule="" LocaleCode="" OrganizationCode=""

```

```

OrganizationKey="" OrganizationName="" ParentOrganizationCode=""
PaymentProcessingReqd="" PrimaryEnterpriseKey="" PrimarySicCode=""
PrimaryUrl="" TaxExemptFlag="" TaxExemptionCertificate=""
TaxJurisdiction="" TaxpayerId="" XrefAliasType=""
XrefOrganizationCode="">
<Node AcceptanceRequired="" ActivateFlag=""
    AdvanceNotificationTime="" AgentCriteriaGroup="" CurrentBolNumber=""
    DcmIntegrationRealTime="" DefaultDeclaredValue=""
    ExportLicenseExpDate="" ExportLicenseNo="" GLN=""
    GenerateBolNumber="" IdentifiedByParentAs=""
    InheritConfigFromShipNode="" InterfaceSubType="" InterfaceType=""
    InventoryTracked="" Inventorytype="" IsItemBasedAllocationAllowed=""
    Latitude="" Longitude="" MaintainInventoryCost=""
    MaxDaysToScheduleBefore="" MinNotificationTime="" OwnerType=""
    PicklistType="" ProcureToShipAllowed="" ReceiptProcessingTime=""
    ReceivingCalendarKey="" ReceivingNode="" RequiresChangeRequest=""
    ReturnCenterFlag="" ReturnsNode="" SerialTracking="" ShipNode=""
    ShipNodeClass="" ShipnodeType="" ShippingCalendarKey=""
    ShippingNode="" SupplierKey="" ThreePlNode="" TimeDiff=""
    UseEndOfShiftTime="" XDocNode=""/>
<CorporatePersonInfo AddressID="" AddressLine1=""
    AddressLine2="" AddressLine3="" AddressLine4="" AddressLine5=""
    AddressLine6="" AlternateEmailID="" Beeper="" City="" Company=""
    Country="" DayFaxNo="" DayPhone="" Department="" EMailID=""
    ErrorTxt="" EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl=""
    JobTitle="" LastName="" Latitude="" Longitude="" MiddleName=""
    MobilePhone="" OtherPhone="" PersonID="" PersonInfoKey=""
    PreferredShipAddress="" State="" Suffix="" Title="" UseCount=""
    VerificationStatus="" ZipCode=""/>
<ContactPersonInfo AddressID="" AddressLine1="" AddressLine2=""
    AddressLine3="" AddressLine4="" AddressLine5="" AddressLine6=""
    AlternateEmailID="" Beeper="" City="" Company="" Country=""
    DayFaxNo="" DayPhone="" Department="" EMailID="" ErrorTxt=""
    EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl="" JobTitle=""
    LastName="" Latitude="" Longitude="" MiddleName="" MobilePhone=""
    OtherPhone="" PersonID="" PersonInfoKey="" PreferredShipAddress=""
    State="" Suffix="" Title="" UseCount="" VerificationStatus=""
    ZipCode=""/>
<BillingPersonInfo AddressID="" AddressLine1="" AddressLine2=""
    AddressLine3="" AddressLine4="" AddressLine5="" AddressLine6=""
    AlternateEmailID="" Beeper="" City="" Company="" Country=""
    DayFaxNo="" DayPhone="" Department="" EMailID="" ErrorTxt=""
    EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl="" JobTitle=""
    LastName="" Latitude="" Longitude="" MiddleName="" MobilePhone=""
    OtherPhone="" PersonID="" PersonInfoKey="" PreferredShipAddress=""

```

```

        State="" Suffix="" Title="" UseCount="" VerificationStatus=""
        ZipCode=""/>
</BuyerOrganization>
<Consumer BillingAddressKey="">
    <BillingPersonInfo AddressID="" AddressLine1="" AddressLine2=""
        AddressLine3="" AddressLine4="" AddressLine5="" AddressLine6=""
        AlternateEmailID="" Beeper="" City="" Company="" Country=""
        DayFaxNo="" DayPhone="" Department="" EMailID="" ErrorTxt=""
        EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl="" JobTitle=""
        LastName="" Latitude="" Longitude="" MiddleName="" MobilePhone=""
        OtherPhone="" PersonID="" PersonInfoKey="" PreferredShipAddress=""
        State="" Suffix="" Title="" UseCount="" VerificationStatus=""
        ZipCode=""/>
</Consumer>
<CustomerSlotPreferenceList>
    <CustomerSlotPreference EndTime="" FridayPreferred=""
        MondayPreferred="" SaturdayPreferred="" StartTime=""
        SundayPreferred="" ThursdayPreferred="" TuesdayPreferred=""
        WednesdayPreferred=""/>
</CustomerSlotPreferenceList>
<CustomerSchedulingPreferences CancelOrderOnBackorder="Y"
    IsDropShipAllowed="Y" IsLineShipComplete="Y" IsLineShipSingleNode="Y"
    IsShipComplete="Y" IsShipSingleNode="Y" IsSubstitutionAllowed="Y"
    OptimizationType=""/>
<CustomerContactList>
    <CustomerContact ApproverProxyUserId="" ApproverUserId="" Company=""
        CustomerContactID="" DateOfBirth="" DayFaxNo="" DayPhone=""
        Department="" EmailID="" EveningFaxNo="" EveningPhone=""
        FirstName="" JobTitle="" LastName="" MiddleName="" MobilePhone=""
        SpendingLimit="" SpendingLimitCurrency="" SpouseDateOfBirth=""
        Status="" Title="" UserID="" WeddingAnniversaryDate="">
<CustomerAdditionalAddressList>
    <CustomerAdditionalAddress AddressType=""
        CustomerAdditionalAddressID="" IsBillTo="" IsDefaultBillTo=""
        IsDefaultShipTo="" IsDefaultSoldTo="" IsShipTo="" IsSoldTo="">
    <PersonInfo AddressID="" AddressLine1="" AddressLine2=""
        AddressLine3="" AddressLine4="" AddressLine5="" AddressLine6=""
        AlternateEmailID="" Beeper="" City="" Company="" Country=""
        DayFaxNo="" DayPhone="" Department="" EMailID="" ErrorTxt=""
        EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl="" JobTitle=""
        LastName="" Latitude="" Longitude="" MiddleName="" MobilePhone=""
        OtherPhone="" PersonID="" PersonInfoKey="" PreferredShipAddress=""
        State="" Suffix="" Title="" UseCount="" VerificationStatus=""
        ZipCode=""/> </CustomerAdditionalAddress>
</CustomerAdditionalAddressList>

```

```

<CustomerPaymentMethodList>
  <CustomerPaymentMethod AccountLimit="" AccountLimitCurrency=""
    CreditCardExpDate="" CreditCardName="" CreditCardNo=""
    CreditCardType="" CustomerAccountNo="" DisplayCreditCardNo=""
    DisplayCustomerAccountNo="" DisplayPaymentReferencel="" FirstName=""
    IsDefaultMethod="" LastName="" MiddleName="" PaymentReferencel=""
    PaymentReference2="" PaymentReference3="" PaymentType="">
    <PersonInfoBillTo AddressID="" AddressLine1="" AddressLine2=""
      AddressLine3="" AddressLine4="" AddressLine5="" AddressLine6=""
      AlternateEmailID="" Beeper="" City="" Company="" Country=""
      DayFaxNo="" DayPhone="" Department="" EMailID="" ErrorTxt=""
      EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl="" JobTitle=""
      LastName="" Latitude="" Longitude="" MiddleName="" MobilePhone=""
      OtherPhone="" PersonID="" PersonInfoKey="" PreferredShipAddress=""
      State="" Suffix="" Title="" UseCount="" VerificationStatus=""
      ZipCode=""/>
  </CustomerPaymentMethod>
</CustomerPaymentMethodList>
<Note AuditTransactionID="" ContactReference="" ContactTime=""
  ContactType="" ContactUser="" CustomerSatIndicator="" NoteText=""
  Operation="" Priority="" ReasonCode="" SequenceNo="" Tranid=""
  VisibleToAll=""/>
</CustomerContact>
</CustomerContactList>
<CustomerAdditionalAddressList>
  <CustomerAdditionalAddress AddressType=""
    CustomerAdditionalAddressID="" IsBillTo="" IsDefaultBillTo=""
    IsDefaultShipTo="" IsDefaultSoldTo="" IsShipTo="" IsSoldTo="">
    <PersonInfo AddressID="" AddressLine1="" AddressLine2="" AddressLine3=""
      AddressLine4="" AddressLine5="" AddressLine6="" AlternateEmailID=""
      Beeper="" City="" Company="" Country="" DayFaxNo="" DayPhone=""
      Department="" EMailID="" ErrorTxt="" EveningFaxNo="" EveningPhone=""
      FirstName="" HttpUrl="" JobTitle="" LastName="" Latitude="" Longitude=""
      MiddleName="" MobilePhone="" OtherPhone="" PersonID="" PersonInfoKey=""
      PreferredShipAddress="" State="" Suffix="" Title="" UseCount=""
      VerificationStatus="" ZipCode=""/>
  </CustomerAdditionalAddress>
</CustomerAdditionalAddressList>
<CustomerPaymentMethodList>
  <CustomerPaymentMethod AccountLimit="" AccountLimitCurrency=""
    CreditCardExpDate="" CreditCardName="" CreditCardNo="" CreditCardType=""
    CustomerAccountNo="" DisplayCreditCardNo="" DisplayCustomerAccountNo=""
    DisplayPaymentReferencel="" FirstName="" LastName="" MiddleName=""
    PaymentReferencel="" PaymentReference2="" PaymentReference3=""
    PaymentType="">

```

```

    <PersonInfoBillTo AddressID="" AddressLine1=""
      AddressLine2="" AddressLine3="" AddressLine4="" AddressLine5=""
      AddressLine6="" AlternateEmailID="" Beeper="" City="" Company=""
      Country="" DayFaxNo="" DayPhone="" Department="" EMailID=""
      ErrorTxt="" EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl=""
      JobTitle="" LastName="" Latitude="" Longitude="" MiddleName=""
      MobilePhone="" OtherPhone="" PersonID="" PersonInfoKey=""
      PreferredShipAddress="" State="" Suffix="" Title="" UseCount=""
      VerificationStatus="" ZipCode=""/>
  </CustomerPaymentMethod>
</CustomerPaymentMethodList>
<AnswerSets>
  <AnswerSet AnswerSetMasterKey="" PrimaryTableName=""
    PrimaryTableRefKey="" QuestionType="">
    <Answers>
      <Answer AnswerMasterKey="" AnswerSetMasterKey="" AnswerValue=""
        QuestionID=""/>
    </Answers>
  </AnswerSet>
</AnswerSets>
<NoteList>
  <Note AuditTransactionId="" ContactReference="" ContactTime=""
    ContactType="" ContactUser="" CustomerSatIndicator="" NoteText=""
    Priority="" ReasonCode="" SequenceNo="" Tranid="" VisibleToAll=""/>
</NoteList>
  <CustomerCurrencyList Reset="">
    <CustomerCurrency Currency="" CustomerCurrencyKey=""
      IsDefaultCurrency="" Operation=""/>
  </CustomerCurrencyList>
</Customer>

```

The Customer Details task returns null as the output.

The Sterling Call Center and Sterling Store applications enable you to open the Customer Details screen in an editor by using a utility method that is explained in [Section 3.8.8, "Opening the Customer Details Screen in an Editor"](#).

### 3.4.14 Advanced Item Search Shared Task

The Sterling Call Center and Sterling Store provides the Advanced Item Search Wizard as a shared task. This shared task is used to open the Advanced Item Search Wizard as a pop-up window or dialog box in any user interface (UI) that references Sterling Call Center and Sterling

Store. On selecting an Item from the list and click of Select button, the item info is returned as output of the shared task. On double click of the selected item in the list screen; another pop-up screen showing the item details opens up. On click of Select button on the item details pop-up, the item info is returned as output of the shared task.

The following is the input XML template for the Item Search shared task:

```
<Item CallingOrganizationCode="" EnterpriseCode=""
ItemGroupCode=""
ItemID="" ItemIDQryType="">
<PrimaryInformation Description="" DescriptionQryType=""
ManufacturerName="" ManufacturerNameQryType=""/>
</Item>
```

The following is the output XML template for the Item Search shared task:

```
<Item CanUseAsServiceTool="" Currency="" GlobalItemID=""
InheritAttributesFromClassification="" ItemGroupCode=""
ItemID="" ItemKey=""
OrganizationCode="" UnitOfMeasure="" YCDPaginationPageNumber="">
<PrimaryInformation AssumeInfiniteInventory=""
CapacityPerOrderedQty="" CapacityQuantityStrategy=""
ComputedUnitCost="" CostCurrency="" CountryOfOrigin=""
CreditWOREceipt="" DefaultProductClass=""
Description=""
ExtendedDescription=""
FixedCapacityQtyPerLine="" FixedPricingQtyPerLine=""
InvoiceBasedOnActuals="" InvolvesSegmentChange=""
IsAirShippingAllowed="" IsDeliveryAllowed=""
IsFreezerRequired="" IsHazmat="" IsParcelShippingAllowed=""
IsPickupAllowed="" IsReturnService="" IsReturnable=""
IsShippingAllowed="" IsStandaloneService="" ItemType=""
KitCode="" ManufacturerItem="" ManufacturerItemDesc=""
ManufacturerName="" MasterCatalogID="" MaxOrderQuantity=""
MinOrderQuantity="" MinimumCapacityQuantity=""
OrderingQuantityStrategy="" PricingQuantityConvFactor=""
PricingQuantityStrategy="" PricingUOM="" PricingUOMStrategy=""
PrimaryEnterpriseCode="" PrimarySupplier="" ProductLine=""
RequiresProdAssociation="" ReturnWindow="" RunQuantity=""
SerializedFlag="" ServiceTypeID=""
ShortDescription=""
Status="" TaxableFlag="" UnitCost="" UnitHeight=""
UnitHeightUOM="" UnitLength="" UnitLengthUOM=""
UnitVolume="" UnitVolumeUOM="" UnitWeight=""
```

```

        UnitWeightUOM="" UnitWidth="" UnitWidthUOM=""/>
    <InventoryParameters ATPRule="" AdvanceNotificationTime=""
        DefaultExpirationDays="" InventoryMonitorRule=""
        IsFifoTracked="" IsSerialTracked="" LeadTime=""
        MinNotificationTime="" NodeLevelInventoryMonitorRule=""
        ProcessingTime="" TagControlFlag="" TagReqAtReceipt=""
        TagReqAtShipment="" TimeSensitive=""/>
    <ClassificationCodes CommodityCode="" ECCNNo="" HarmonizedCode=""
        HazmatClass="" NAICSCode="" NMFCClass="" NMFCCode=""
        OperationalConfigurationComplete="" PickingType=""
        PostingClassification="" Schedule_B_Code="" StorageType=""
        TaxProductCode="" UNSPSC="" VelocityCode=""/>
    <ItemAliasList>
        <ItemAlias AliasName="" AliasValue=""/>
    </ItemAliasList>
    <ItemExclusionList/>
    <AdditionalAttributeList>
        <AdditionalAttribute Name="" Value=""/>
    </AdditionalAttributeList>
    <LanguageDescriptionList/>
    <ItemAttributeGroupTypeList/>
    <ComputedPrice ListPrice="" RetailPrice="" UnitPrice="">
        <QuantityRangePriceList/>
    </ComputedPrice>
</Item>

```

### 3.4.15 Price Match Shared Task

Sterling Call Center and Sterling Store provides the Price Match Wizard as a shared task. This shared task is used to open the Price Match Wizard as a pop-up window or dialog box in any UI that references Sterling Call Center and Sterling Store.

The following is the input XML template for the Override Price task:

```

<Order CustomerFirstName="" CustomerLastName="" DocumentType=""
EnterpriseCode="" OrderHeaderKey=""
    OrderNo="" SellerOrganizationCode="" isHistory="" IsOrderEntry=""
DraftOrderFlag="">
    <PriceInfo Currency=""/>
</Order>

```

OrderHeaderKey, DocumentType, and EnterpriseCode are mandatory parameters and must be passed in the input.

To use this shared task for Draft Orders, pass `DraftOrderFlag="Y"` in the input.

The Price Match task returns null as the output.

### 3.4.16 Override Item Price Shared Task

Sterling Call Center and Sterling Store provides the Override Price for Item screen as a shared task. This shared task is used to open the Override Price as a pop-up window or dialog box in any user interface (UI) that references Sterling Call Center and Sterling Store.

The following is the input XML template for the Override Price task:

```
<Order EnterpriseCode="">
  <OrderLine>
    <LinePriceInfo Currency="" TransactionUnitPrice="" UnitPrice=""/>
  </OrderLine>
</Order>
```

The following is the output XML template for the Override Price task:

```
<Order ModificationReasonCode="">
  <OrderLine>
    <LinePriceInfo IsPriceLocked="" UnitPrice=""/>
    <Notes>
      <Note ContactReference="" ContactType="" ContactUser=""
        NoteText="" Priority="" ReasonCode="" VisibleToAll=""/>
    </Notes>
  </OrderLine>
</Order>
```

## 3.5 User Interface Customization

This section explains the user interface customizations supported by Sterling Call Center and Sterling Store.

### 3.5.1 Special Themes

This section describes the special themes needed when implementing Sterling Call Center and Sterling Store.



### 3.5.1.1 Mandatory Fields

In Sterling Call Center and Sterling Store, certain screens have mandatory fields. You can customize the display of mandatory fields by creating a custom theme and modifying the entry in your custom theme. For more information about user interface themes, see [Section A.1, "User Interface Themes"](#).

### 3.5.1.2 Smileys

Sterling Call Center and Sterling Store uses emoticons. Each of these emoticons are configured by defining a theme starting with a "Smiley" string. For instance, you can display the 😞 (sad) emoticon by using the "SmileySad" theme.

## 3.5.2 Display of Day Panels in the Service Appointment Calendar

You can use the post command on the `getWorkOrderAppointmentOptions` API to override the default view of the day panels. You can modify the output template of the `getWorkOrderAppointmentOptions` API by adding the following elements at the root level:

```
<DateDisplays>
  <DateDisplay Date="<date>" ThemeName="<theme name>" StringDisplay="<string to
display>" />
</DateDisplays>
```

The value of the `ThemeName` attribute should be a valid theme name as found in the theme files being used in the client application. This theme will be applied to the day panels that appear in the calendar. This enables you to change the display of the day panel, including the background color. The value of the `StringDisplay` attribute will be displayed as it is inside each day panel. This enables you to display any string inside each day panel.

For more information about customizing an output template, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

### 3.5.3 User Interface Validation of Phone Number and E-Mail Address

Sterling Call Center and Sterling Store validates e-mail address and phone numbers on the address capture screens. These fields are validated in the plug-in using the YRCDataFormatter extension point. For more information about custom data formatting, refer to the *Selling and Fulfillment Foundation: Customizing the Rich Client Platform Interface Guide*.

---

---

**Note:** Sterling Call Center and Sterling Store does not validate the phone numbers and e-mail address on any of the search screens.

---

---

Sterling Call Center and Sterling Store checks that each phone number and e-mail address entered is valid.

The phone number validation logic verifies that there are either ten or eleven digits entered in the field.

Once a phone number has been validated, Sterling Call Center and Sterling Store removes all non-digit characters from the phone number field and formats it: (###) - ### - ####.

---

---

**Note:** Sterling Call Center and Sterling Store will only validates phone numbers located in the United States. The validation logic can be extended in the extension behavior class to work with other countries, if required. For more information about creating extension behavior, refer to the *Selling and Fulfillment Foundation: Using the Sterling Rich Client Platform Extensibility Tool Guide*

---

---

Sterling Call Center and Sterling Store validates e-mail addresses using the following criteria:

- The first character must be an alphabet.
- The symbol @ must be present.
- There must be at least one character before @.
- There must be at least one character between at and the period.

- There must be between 2-4 characters after the period.
- There must not be spaces in the e-mail address.
- There must be a semi-colon (;) between addressess.

If any of these validations fail, Sterling Call Center and Sterling Store displays an error message to notify the user of the incorrect field.

## 3.6 Address Panel Customization

Sterling Call Center and Sterling Store uses a common address panel when displaying the addresses in the following screens.

- Un-verified and verified addresses on the Alert Details screen for Alerts of type 'Address Verification'
- Ship To Address in Change Order Address
- Pickup Address on Return Fulfillment Summary
- Ship to address on Shipment Details
- Bill To address on Customer Pick
- Store Address on Line Summary
- Bill To Address and Ship To Address on Order Summary
- Bill To address on Invoice Details
- Store Address on Procurement Order Summary
- Bill To Address and Ship To Address on Return Order Summary
- Ship To address on Track an Item

If you customize the panel on any of these screens, the customizations will take effect in all the screens. For example, if you hide a control on the bill to address panel of the Order Summary screen, the control will also be hidden in the other screens displaying address.

## 3.7 Opening New Related Tasks in the Order Editor

Sterling Call Center and Sterling Store provides an Order Editor in which you can open new related tasks by specifying the following:

- Editor ID

```
com.yantra.pca.ycd.rcp.editors.YCDOrderEditor
```

- Input

```
<Order CustomerFirstName=" " CustomerLastName=" "
DocumentType=" " EnterpriseCode=" " OrderHeaderKey=" " OrderNo=" "
SellerOrganizationCode=" " isHistory=" ">
  <PriceInfo Currency=" " />
</Order>
```

For more information about using editor, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

## 3.8 Opening the Screens of the Sterling Call Center and Sterling Store Application in an Editor

This section explains the procedure that you must follow to open the screens of the Sterling Call Center and Sterling Store application in an editor.

**Note:** It is recommended that you use the utility methods that are provided by the Sterling Call Center and Sterling Store application to open the screens in an editor.

### 3.8.1 Opening the Order Summary Screen in an Editor

You can open the Order Summary screen in an editor by using the `YCDEXTENSION_UTILS.launchTaskInEditor(String taskId, Element input)`; utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDEXTENSION_UTILS.launchTaskInEditor("YCD_TASK_VIEW_ORDER_SUMMARY", input);
```

Use the following input XML to open the Order Summary screen in an editor:

```
<Order BillToID=" " BuyerOrganizationCode=" " CustomerFirstName=" "
CustomerLastName=" " DocumentType=" " EnterpriseCode=" " OrderHeaderKey=" "
OrderNo=" " SellerOrganizationCode=" " isHistory=" ">
  <PriceInfo Currency=" " />
</Order>
```

```
</Order>
```

**Note:** Validations are not performed in the Order Summary screen to verify whether the order is a Draft Order.

Ensure that the OrderHeaderKey parameter is present in the input XML.

### 3.8.2 Opening the Return Search Screen in an Editor

You can open the Return Search screen in an editor by using the `YCDExtensionUtils.launchTaskInEditor(String taskId, Element input);` utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDExtensionUtils.launchTaskInEditor("YCD_TASK_RETURN_SEARCH",
input);
```

Use the following input XML to open the Return Search screen in an editor:

```
<Order BillToID="" BillToIDQryType="" CardNo="" CustomerEMailID=""
  CustomerEMailIDQryType="" CustomerFirstName="" CustomerFirstNameQryType=""
  CustomerLastName="" CustomerLastNameQryType="" CustomerPhoneNo=""
  CustomerPhoneNoQryType="" CustomerZipCode="" DraftOrderFlag=""
  EnterpriseCode="" FromOrderDate="" OrderNo="" OrderNoQryType=""
  ShowOnlyStoreOrders="" ToOrderDate="">
  <OrderLine DerivedFromOrderHeaderKeyQryType="" />
</OrderLines>
<OrderLine>
  <DerivedFromOrder OrderNo="" OrderNoQryType="" />
</OrderLine>
</OrderLines>
<OrderBy>
  <Attribute Desc="" Name="" />
</OrderBy>
</Order>
```

### 3.8.3 Opening the Cancel Order Screen in an Editor

You can open the Cancel Order screen in an editor by using the `YCDExtensionUtils.launchTaskInEditor(String taskId, Element input);`

utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDExtensionUtils.launchTaskInEditor("YCD_TASK_ORDER_CANCEL",
input);
```

Use the following input XML to open the Cancel Order screen in an editor:

```
<Order BillToID="" BuyerOrganizationCode="" ChangeTaskName=""
  CustomerFirstName="" CustomerLastName="" DocumentType="" EnterpriseCode=""
  OrderHeaderKey="" OrderNo="" SellerOrganizationCode="" isHistory="">
  <PriceInfo Currency=""/>
</Order>
```

**Note:** Ensure that the OrderHeaderKey parameter is present in the input XML.

### 3.8.4 Opening the Create Alert Screen in an Editor

You can open the Create Alert screen in an editor by using the YCDExtensionUtils.launchTaskInEditor(String taskId, Element input); utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDExtensionUtils.launchTaskInEditor("YCD_CREATE_ALERT", input);
```

Use the following input XML to open the Create Alert screen in an editor:

```
<Inbox AssignedToUserId="" DetailDescription="" ExceptionType="" FollowupDate=""
OrderNo="" Priority="" QueueKey="" SalesOrderNo=""/>
```

### 3.8.5 Opening the Alert List Screen in an Editor

You can open the Alert List screen in an editor by using the YCDExtensionUtils.launchTaskInEditor(String taskId, Element input); utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDExtensionUtils.launchTaskInEditor("YCD_SEARCH_ALERT", input);
```

Use the following input XML to open the Alert List screen in an editor:

```
<Input>
  <Input>
    <Inbox ConsolidationCount="" ConsolidationCountQryType=""
```

```

        ExceptionType="" FromFollowupDate="" FromGeneratedOn="" OrderNo=""
        OrderNoQryType="" Priority="" PriorityQryType="" QueueKey=""
        ShipnodeKey="" SubscribedQueues="" ToFollowupDate=""
        ToGeneratedOn=""> <User Loginid="" LoginidQryType=""/>
    </Inbox>
</Input>
</Input>

```

### 3.8.6 Opening the Alert Details Screen in an Editor

You can open the Alert Details screen in an editor by using the `YCDExtensionUtils.launchTaskInEditor(String taskId, Element input);` utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDExtensionUtils.launchTaskInEditor("YCD_ALERT_DETAIL", input);
```

Use the following input XML to open the Alert List screen in an editor:

```
<Inbox InboxKey="" Title=""/>
```

### 3.8.7 Opening the Item Details Screen in an Editor

You can open the Item Details screen in an editor by using the `YCDExtensionUtils.launchTaskInEditor(String taskId, Element input);` utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDExtensionUtils.launchTaskInEditor("YCD_TASK_ITEM_DETAILS",
input);
```

The input element in the utility method should not be null and should use the following input XML. However, if the input element is null, an error message is displayed indicating you to enter a valid input element.

```

<Item AuthenticationKey="" CallingOrganizationCode=""
    CanUseAsServiceTool="" Currency="" DemandType=""
    DisplayLocalizedFieldInLocale="" ExcludeChildItemsOfModelItems=""
    ExcludeItemsInPriceList="" ExecuteQueries="" FulfillmentType=""
    GetAvailabilityFromCache="" GetUnpublishedItems="" GlobalItemID=""
    IgnoreInvalidItems="" IgnoreIsSoldSeparately="" IgnoreOrdering=""
    InheritAttributesFromClassification="" IsForOrdering=""
    ItemGroupCode="" ItemID="" MaximumRecords="" OrganizationCode=""
    PriceProgramKey="" PriceProgramName="" PricingDate=""
    PurposeForDisplayClassification=""
    ShowAttributesWithDifferentValuesOnly=""

```

```

ShowFinalSupersededItemOnly="" ShowKeyItemAttributesOnly=""
StartItemKey="" SubCatalogOrganizationCode="" UnitOfMeasure=""
UserOrganizationCode="">
  <Barcode BarcodeData="" BarcodeType="">
    <ContextualInfo OrganizationCode=""/>
  </Barcode>
  <CustomerInformation CustomerID="" CustomerLevel="" CustomerType=""
    Region="" RelationshipType="" Vertical=""/>
  <ShipToAddress AddressLine1="" AddressLine2="" AddressLine3=""
    AddressLine4="" AddressLine5="" AddressLine6=""
    AlternateEmailID="" Beeper="" City="" Company="" Country=""
    DayFaxNo="" DayPhone="" Department="" EMailID="" ErrorTxt=""
    EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl=""
    IsCommercialAddress="" JobTitle="" LastName="" MiddleName=""
    MobilePhone="" OtherPhone="" PersonID="" PersonInfoKey=""
    PreferredShipAddress="" State="" Suffix="" TaxGeoCode=""
    Title="" UseCount="" VerificationStatus="" ZipCode=""/>
  <ShipNodes>
    <ShipNode Node=""/>
  </ShipNodes>
  <ItemAssociationTypeList>
    <ItemAssociationType Type=""/>
  </ItemAssociationTypeList>
  <PrimaryInformation AssumeInfiniteInventory="" ColorCode=""
    ConfiguredModelKey="" CostCurrency="" CountryOfOrigin=""
    DefaultProductClass="" Description="" IsAirShippingAllowed=""
    IsConfigurable="" IsDeliveryAllowed="" IsForwardingAllowed=""
    IsFreezerRequired="" IsHazmat="" IsModelItem=""
    IsParcelShippingAllowed="" IsPickupAllowed="" IsPreConfigured=""
    IsProcurementAllowed="" IsReturnService="" IsReturnable=""
    IsShippingAllowed="" IsStandaloneService=""
    IsSubOnOrderAllowed="" ItemType="" KitCode=""
    ManufacturerItem="" ManufacturerItemDesc="" ManufacturerName=""
    MasterCatalogID="" MaxOrderQuantity="" MinOrderQuantity=""
    MinimumCapacityQuantity="" ModelItemUnitOfMeasure=""
    NumSecondarySerials="" PricingQuantityConvFactor=""
    PricingQuantityStrategy="" PricingUOM="" PricingUOMStrategy=""
    ProductLine="" RequiresProdAssociation="" ServiceTypeID=""
    ShipmentConsolidationWindow="" ShortDescription="" SizeCode=""
    Status="" TaxableFlag="" UnitCost="" UnitHeight=""
    UnitHeightUOM="" UnitLength="" UnitLengthUOM="" UnitWeight=""
    UnitWeightUOM="" UnitWidth="" UnitWidthUOM=""/>
  <InventoryParameters ATPRule="" AdvanceNotificationTime=""
    CapacityQuantityStrategy="" DefaultExpirationDays=""
    FutureSafetyFactorPercentage="" InventoryMonitorRule=""

```



```

        IsFifoTracked="" IsItemBasedAllocationAllowed=""
        IsSerialTracked="" LeadTime="" MaximumNotificationTime=""
        MinNotificationTime="" NodeLevelInventoryMonitorRule=""
        OnhandSafetyFactorPercentage="" OnhandSafetyFactorQuantity=""
        ProcessingTime="" SerializedFlag="" TagControlFlag=""
        TimeSensitive="" UseUnplannedInventory=""/>
<ClassificationCodes CommodityCode="" ECCNNo="" HarmonizedCode=""
        HazmatClass="" Model="" NAICSCode="" NMFCClass="" NMFCCode=""
        OperationalConfigurationComplete="" PickingType=""
        PostingClassification="" Schedule_B_Code="" StorageType=""
        TaxProductCode="" UNSPSC=""/>
<ItemAliasList>
    <ItemAlias AliasName="" AliasValue=""/>
</ItemAliasList>
<ItemExclusionList>
    <ItemExclusion ExclusionCode=""/>
</ItemExclusionList>
<AdditionalAttributeList>
    <AdditionalAttribute DoubleValue="" IntegerValue="" Name="" Value=""/>
</AdditionalAttributeList>
<CategoryFilter CategoryPath="" OrganizationCode=""/>
<Components>
    <Component ComponentItemID="" ComponentItemKey=""
        ComponentOrganizationCode="" ComponentUnitOfMeasure=""/>
</Components>
<LanguageDescriptionList>
    <LanguageDescription LocaleCode=""/>
</LanguageDescriptionList>
<InventoryTagAttributes BatchNo="" ItemTagKey="" LotAttribute1=""
    LotAttribute2="" LotAttribute3="" LotKeyReference=""
    LotNumber="" ManufacturingDate="" RevisionNo=""/>
<ItemAttributeGroupTypeList>
    <ItemAttributeGroupType ClassificationPurposeCode=""
ItemAttributeGroupType=""/>
</ItemAttributeGroupTypeList>
<OrderBy>
    <Attribute Desc="" Name=""/>
</OrderBy>
</Item>

```

### 3.8.8 Opening the Customer Details Screen in an Editor

You can open the Customer Details screen in an editor by using the `YCDExtensionUtils.launchTaskInEditor(String taskId, Element input);`

utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDExtensionUtils.launchTaskInEditor("YCD_CUSTOMER_DETAILS",
input);
```

The utility method can be used for both consumer customers and business customers.

The input element in the utility method should not be null and should use the following input XML. However, if the input element is null, an error message is displayed indicating you to enter a valid input element.

```
<Customer CustomerKey="" BuyerOrganizationCode="" CallingOrganizationCode=""
  CanConsumeSupplementalCapacity="" CommitmentTime=""
  CustomerAssignedVendorID="" CustomerClassificationCode=""
  CustomerID="" CustomerLevel="" CustomerType="" ExternalCustomerID=""
  FuncAckTime="" OrganizationCode="" ParentCustomerKey=""
  RegisteredDate="" RelationshipType="" RootCustomerKey="" SendsAsn=""
  SendsCommitment="" SendsFuncAck="" SlotPreferenceType="" Status=""
Vertical="">
  <ParentCustomer CustomerID="" OrganizationCode=""/>
  <BuyerOrganization AccountWithHub="" AuthorityType=""
    BillingAddressKey="" CatalogOrganizationCode=""
    ContactAddressKey="" CorporateAddressKey=""
    CreatorOrganizationKey="" DunsNumber=""
    InventoryOrganizationCode="" InventoryPublished="" IsBuyer=""
    IsHubOrganization="" IsLegalEntity="" IsNode=""
    IsSourcingKept="" IssuingAuthority="" ItemXrefRule=""
    LocaleCode="" OrganizationCode="" OrganizationKey=""
    OrganizationName="" ParentOrganizationCode=""
    PaymentProcessingReqd="" PrimaryEnterpriseKey=""
    PrimarySicCode="" PrimaryUrl="" TaxExemptFlag=""
    TaxExemptionCertificate="" TaxJurisdiction="" TaxpayerId=""
    XrefAliasType="" XrefOrganizationCode="">
    <Node AcceptanceRequired="" ActivateFlag=""
      AdvanceNotificationTime="" AgentCriteriaGroup=""
      CurrentBolNumber="" DcmIntegrationRealTime=""
      DefaultDeclaredValue="" ExportLicenseExpDate=""
      ExportLicenseNo="" GLN="" GenerateBolNumber=""
      IdentifiedByParentAs="" InheritConfigFromShipNode=""
      InterfaceSubType="" InterfaceType="" InventoryTracked=""
      Inventorytype="" IsItemBasedAllocationAllowed="" Latitude=""
      Longitude="" MaintainInventoryCost=""
      MaxDaysToScheduleBefore="" MinNotificationTime=""
      OwnerType="" PicklistType="" ProcureToShipAllowed=""
```

```

ReceiptProcessingTime="" ReceivingCalendarKey=""
ReceivingNode="" RequiresChangeRequest=""
ReturnCenterFlag="" ReturnsNode="" SerialTracking=""
ShipNode="" ShipNodeClass="" ShipNodeType=""
ShippingCalendarKey="" ShippingNode="" SupplierKey=""
ThreePlNode="" TimeDiff="" UseEndOfShiftTime="" XDocNode=""/>
<CorporatePersonInfo AddressID="" AddressLine1=""
AddressLine2="" AddressLine3="" AddressLine4=""
AddressLine5="" AddressLine6="" AlternateEmailID=""
Beeper="" City="" Company="" Country="" DayFaxNo=""
DayPhone="" Department="" EMailID="" ErrorTxt=""
EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl=""
JobTitle="" LastName="" Latitude="" Longitude=""
MiddleName="" MobilePhone="" OtherPhone="" PersonID=""
PersonInfoKey="" PreferredShipAddress="" State="" Suffix=""
Title="" UseCount="" VerificationStatus="" ZipCode=""/>
<ContactPersonInfo AddressID="" AddressLine1="" AddressLine2=""
AddressLine3="" AddressLine4="" AddressLine5=""
AddressLine6="" AlternateEmailID="" Beeper="" City=""
Company="" Country="" DayFaxNo="" DayPhone="" Department=""
EMailID="" ErrorTxt="" EveningFaxNo="" EveningPhone=""
FirstName="" HttpUrl="" JobTitle="" LastName="" Latitude=""
Longitude="" MiddleName="" MobilePhone="" OtherPhone=""
PersonID="" PersonInfoKey="" PreferredShipAddress=""
State="" Suffix="" Title="" UseCount=""
VerificationStatus="" ZipCode=""/>
<BillingPersonInfo AddressID="" AddressLine1="" AddressLine2=""
AddressLine3="" AddressLine4="" AddressLine5=""
AddressLine6="" AlternateEmailID="" Beeper="" City=""
Company="" Country="" DayFaxNo="" DayPhone="" Department=""
EMailID="" ErrorTxt="" EveningFaxNo="" EveningPhone=""
FirstName="" HttpUrl="" JobTitle="" LastName="" Latitude=""
Longitude="" MiddleName="" MobilePhone="" OtherPhone=""
PersonID="" PersonInfoKey="" PreferredShipAddress=""
State="" Suffix="" Title="" UseCount=""
VerificationStatus="" ZipCode=""/>
</BuyerOrganization>
<Consumer BillingAddressKey="">
<BillingPersonInfo AddressID="" AddressLine1="" AddressLine2=""
AddressLine3="" AddressLine4="" AddressLine5=""
AddressLine6="" AlternateEmailID="" Beeper="" City=""
Company="" Country="" DayFaxNo="" DayPhone="" Department=""
EMailID="" ErrorTxt="" EveningFaxNo="" EveningPhone=""
FirstName="" HttpUrl="" JobTitle="" LastName="" Latitude=""
Longitude="" MiddleName="" MobilePhone="" OtherPhone=""

```

```

        PersonID="" PersonInfoKey="" PreferredShipAddress=""
        State="" Suffix="" Title="" UseCount=""
        VerificationStatus="" ZipCode=""/>
    </Consumer>
    <CustomerSlotPreferenceList>
        <CustomerSlotPreference EndTime="" FridayPreferred=""
            MondayPreferred="" SaturdayPreferred="" StartTime=""
            SundayPreferred="" ThursdayPreferred="" TuesdayPreferred=""
WednesdayPreferred=""/>
    </CustomerSlotPreferenceList>
    <CustomerSchedulingPreferences CancelOrderOnBackorder="Y"
        IsDropShipAllowed="Y" IsLineShipComplete="Y"
        IsLineShipSingleNode="Y" IsShipComplete="Y" IsShipSingleNode="Y"
        IsSubstitutionAllowed="Y" OptimizationType=""/>
    <CustomerContactList>
        <CustomerContact ApproverProxyUserId="" ApproverUserId=""
            Company="" CustomerContactID="" DateOfBirth="" DayFaxNo=""
            DayPhone="" Department="" EmailID="" EveningFaxNo=""
            EveningPhone="" FirstName="" JobTitle="" LastName=""
            MiddleName="" MobilePhone="" SpendingLimit=""
            SpendingLimitCurrency="" SpouseDateOfBirth="" Status=""
            Title="" UserID="" WeddingAnniversaryDate="">
        <CustomerAdditionalAddressList>
            <CustomerAdditionalAddress AddressType=""
                CustomerAdditionalAddressID="" IsBillTo=""
                IsDefaultBillTo="" IsDefaultShipTo=""
                IsDefaultSoldTo="" IsShipTo="" IsSoldTo="">
            <PersonInfo AddressID="" AddressLine1=""
                AddressLine2="" AddressLine3="" AddressLine4=""
                AddressLine5="" AddressLine6=""
                AlternateEmailID="" Beeper="" City="" Company=""
                Country="" DayFaxNo="" DayPhone="" Department=""
                EmailID="" ErrorTxt="" EveningFaxNo=""
                EveningPhone="" FirstName="" HttpUrl=""
                JobTitle="" LastName="" Latitude="" Longitude=""
                MiddleName="" MobilePhone="" OtherPhone=""
                PersonID="" PersonInfoKey=""
                PreferredShipAddress="" State="" Suffix=""
                Title="" UseCount="" VerificationStatus="" ZipCode=""/>
            </CustomerAdditionalAddress>
        </CustomerAdditionalAddressList>
    <CustomerPaymentMethodList>
        <CustomerPaymentMethod AccountLimit=""
            AccountLimitCurrency="" CreditCardExpDate=""
            CreditCardName="" CreditCardNo="" CreditCardType=""

```

```

CustomerAccountNo="" DisplayCreditCardNo=""
DisplayCustomerAccountNo=""
DisplayPaymentReference1="" FirstName=""
IsDefaultMethod="" LastName="" MiddleName=""
PaymentReference1="" PaymentReference2=""
PaymentReference3="" PaymentType="">
<PersonInfoBillTo AddressID="" AddressLine1=""
AddressLine2="" AddressLine3="" AddressLine4=""
AddressLine5="" AddressLine6=""
AlternateEmailID="" Beeper="" City="" Company=""
Country="" DayFaxNo="" DayPhone="" Department=""
EMailID="" ErrorTxt="" EveningFaxNo=""
EveningPhone="" FirstName="" HttpUrl=""
JobTitle="" LastName="" Latitude="" Longitude=""
MiddleName="" MobilePhone="" OtherPhone=""
PersonID="" PersonInfoKey=""
PreferredShipAddress="" State="" Suffix=""
Title="" UseCount="" VerificationStatus="" ZipCode=""/>
</CustomerPaymentMethod>
</CustomerPaymentMethodList>
<Note AuditTransactionID="" ContactReference=""
ContactTime="" ContactType="" ContactUser=""
CustomerSatIndicator="" NoteText="" Operation=""
Priority="" ReasonCode="" SequenceNo="" Tranid=""
VisibleToAll=""/>
</CustomerContact>
</CustomerContactList>
<CustomerAdditionalAddressList>
<CustomerAdditionalAddress AddressType=""
CustomerAdditionalAddressID="" IsBillTo=""
IsDefaultBillTo="" IsDefaultShipTo="" IsDefaultSoldTo=""
IsShipTo="" IsSoldTo="">
<PersonInfo AddressID="" AddressLine1="" AddressLine2=""
AddressLine3="" AddressLine4="" AddressLine5=""
AddressLine6="" AlternateEmailID="" Beeper="" City=""
Company="" Country="" DayFaxNo="" DayPhone=""
Department="" EMailID="" ErrorTxt="" EveningFaxNo=""
EveningPhone="" FirstName="" HttpUrl="" JobTitle=""
LastName="" Latitude="" Longitude="" MiddleName=""
MobilePhone="" OtherPhone="" PersonID=""
PersonInfoKey="" PreferredShipAddress="" State=""
Suffix="" Title="" UseCount="" VerificationStatus=""
ZipCode=""/>
</CustomerAdditionalAddress>
</CustomerAdditionalAddressList>

```

```

<CustomerPaymentMethodList>
  <CustomerPaymentMethod AccountLimit="" AccountLimitCurrency=""
    CreditCardExpDate="" CreditCardName="" CreditCardNo=""
    CreditCardType="" CustomerAccountNo=""
    DisplayCreditCardNo="" DisplayCustomerAccountNo=""
    DisplayPaymentReferencel="" FirstName="" LastName=""
    MiddleName="" PaymentReferencel="" PaymentReference2=""
    PaymentReference3="" PaymentType="">
    <PersonInfoBillTo AddressID="" AddressLine1=""
      AddressLine2="" AddressLine3="" AddressLine4=""
      AddressLine5="" AddressLine6="" AlternateEmailID=""
      Beeper="" City="" Company="" Country="" DayFaxNo=""
      DayPhone="" Department="" EMailID="" ErrorTxt=""
      EveningFaxNo="" EveningPhone="" FirstName="" HttpUrl=""
      JobTitle="" LastName="" Latitude="" Longitude=""
      MiddleName="" MobilePhone="" OtherPhone="" PersonID=""
      PersonInfoKey="" PreferredShipAddress="" State=""
      Suffix="" Title="" UseCount="" VerificationStatus=""
ZipCode=""/>
  </CustomerPaymentMethod>
</CustomerPaymentMethodList>
<AnswerSets>
  <AnswerSet AnswerSetMasterKey="" PrimaryTableName=""
    PrimaryTableRefKey="" QuestionType="">
    <Answers>
      <Answer AnswerMasterKey="" AnswerSetMasterKey=""
        AnswerValue="" QuestionID=""/>
    </Answers>
  </AnswerSet>
</AnswerSets>
<NoteList>
  <Note AuditTransactionId="" ContactReference="" ContactTime=""

```

### 3.8.9 Opening the Return Order Summary Screen in an Editor

You can open the Return Order Summary screen in an editor by using the `YCDExtensionUtils.launchTaskInEditor(String taskId, Element input);` utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDExtensionUtils.launchTaskInEditor("YCD_TASK_VIEW_RETURN_ORDE
R_SUMMARY", input);
```

The input element in the utility method should not be null and should use the following input XML. However, if the input element is null, an error message is displayed indicating you to enter a valid input element.

```
<Order DraftOrderFlag="" CustomerFirstName="" CustomerLastName=""
      DocumentType="" EnterpriseCode="" OrderHeaderKey="" OrderNo=""
      SellerOrganizationCode="" isHistory="" BuyerUserId="" CustomerContactID=""
      BillToID="" BuyerOrganizationCode="" > PriceInfo Currency=""
      TotalAmount=""/>
<ExchangeOrders>
  <ExchangeOrder OrderHeaderKey=""/>
</ExchangeOrders>
</Order>
```

The EnterpriseCode, OrderHeaderKey, DocumentType, and OrderNo parameters are mandatory and must be passed in the input.

If the DocumentType parameter is not passed in the input, the Sterling Call Center and Sterling Store applications will pass the Default Return Order Document Type value in the input.

### 3.8.10 Opening the Advanced Item Search Screen in an Editor

You can open the Advanced Item Search screen in an editor by using the YCDExtensionUtils.launchTaskInEditor(String taskId, Element input); utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDExtensionUtils.launchTaskInEditor("YCD_TASK_ITEM_SEARCH",
input);
```

The following is the input XML template:

```
<Item CallingOrganizationCode="" EnterpriseCode="" ItemGroupCode=""
      ItemID="" ItemIDQryType="">
  <PrimaryInformation Description="" DescriptionQryType=""
      ManufacturerName="" ManufacturerNameQryType=""/>
</Item>
```

## 3.9 Opening the Price Match Wizard in an Editor

You can open the Price Match Wizard in an editor by using the YCDExtensionUtils.launchTaskInEditor(String taskId, Element input);

utility method. For example, you can use the following utility method from the extension behavior class:

```
YCDExtensionUtils.launchTaskInEditor("YCD_PRICE_MATCH_AN_ITEM",  
input);
```

Input element should not be null and should follow the following template:

```
<Order CustomerFirstName="" CustomerLastName="" DocumentType=""  
      EnterpriseCode="" OrderHeaderKey=""  
      OrderNo="" SellerOrganizationCode="" isHistory="" IsOrderEntry=""  
DraftOrderFlag="">  
  <PriceInfo Currency=""/>  
</Order>
```

OrderHeaderKey, DocumentType, and EnterpriseCode are mandatory parameters and must be passed in the input.

To use this utility for Draft Orders, pass DraftOrderFlag="Y" in the input.



# Implementing the Sterling Call Center Features

---

This chapter describes the call center features supported by Sterling Call Center and Sterling Store.

## 4.1 Item Inquiry

A user may want to search for product items or service items and view the details pertaining to them, such as fulfillment methods, availability, promotions, associations, and price. An enterprise may want to model items based on distinct attributes such as style, size and color, and enable users to search and view the details of the stylized items.

### 4.1.1 Searching and Viewing the Details of an Item

Call center representatives may frequently receive questions from customers about specific items, including their fulfillment methods, specifications, availability, or price.

#### 4.1.1.1 Solution

The Item Inquiry task enables users to search for product and service items and view their details. Based on the search criteria entered, a list of items is displayed. The user can view additional information such as fulfillment methods, availability, price, promotions, and specifications for each item.

---

---

**Note:** Users can also search for and view the details of items whose effective end date has expired.

---

---

Sterling Call Center and Sterling Store provides two ways to search for items:

- Quick Access—The user can enter the item ID, description and manufacturer's name and search for the item.
- Advanced Item Search—The user can enter more restrictive search criteria to search for the item.

Sterling Call Center and Sterling Store enable you to specify the type of search to be displayed in the Advanced Item Search screen. Users can search either for all the items present in the database or search only for the items that are currently being sold (items that are present in the current active search index). For more information about configuring the item search options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

---

---

**Note:** When the user searches for provided services from the current active search index, Sterling Call Center and Sterling Store will search and list the Provided Service item only if the item has been assigned to a category. For more information about assigning items to a category, see the *Business Center: Item Administration Guide*.

---

---

If the user has access to multiple enterprises, the Enterprise drop-down list is displayed in the Item Search Criteria panel of the Item Search screen. The `getOrganizationList` API is called to retrieve the list of enterprises. If the search criteria results in a single record, the Item Details screen is displayed. If the search criteria results in multiple records being retrieved, the Search and List screen is displayed.

This section explains the Item Inquiry process and the solution offered by Sterling Call Center and Sterling Store.

### Item Information Retrieval

When a user scans item ID using the barcode scanner, the `translateBarCode` API is used to translate and validate the barcode of the

item. If alternate item identifiers are entered or scanned, the `translateBarCode` API is called to retrieve the original item information for the scanned alternate item identifiers. However, the `translateBarCode` API is called only if the Enable Scanning Of Items Using Bar Code rule is selected in Configure Item Search Options of the Channel Applications Manager. For more information about alternate item identifiers, see [Section 4.3, "Alternate Item Identifier"](#). The `getCompleteItemList` API retrieves a list of items based on the input criteria. This item list includes additional information such as UOMs, fulfillment methods, pricing, and promotions.

The `getCompleteItemList` API supports case-insensitive searches for system defined columns. For more information about enabling APIs to support case-insensitive searches for search tasks in the Sterling Call Center and Sterling Store applications, refer to the *Selling and Fulfillment Foundation: Extending the Database Guide*.

The user can search for products, services, or both products and services in the Item Search screen. When searching for services, or both products and services, the results display both independent and non-independent services. Non-independent services cannot be added to an order from the Item Inquiry screens.

---



---

**Note:** If the user searches by item ID, item entitlement rules will be applied, if configured, when the `getCompleteItemList` API is called. For more information about configuring the item search options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

---



---

The `getItemUOMList` is used to get the list of UOMs for the item.

Sterling Call Center and Sterling Store provides a configuration to display Delivery and Shipping as separate fulfillment methods. It also provides a configuration to never show the Pickup fulfillment method if the enterprise does not support Pickup. These configurations are verified when displaying the fulfillment methods in the Preview panel of the Item Search screen and the Item Details panel of the Item Details screen.

The `YFSGetPromotionsForItemList` user exit is called to retrieve and display the promotion details on the Item Details screen.

### Item Availability Retrieval

The findInventory API is called to display the availability of product items and the capacity of service items.

You can disable availability checks in the Item Search screen to avoid frequent inventory requests on your inventory system.

You can inquire about an item's availability in the Item Detail screen even when the default availability check is disabled. The display of the Country drop-down list on this availability panel is based on the rule to allow country entry. If you have configured region schemas for displaying states, the availability panel displays a drop-down list of states.

### Displaying Item Images

You can display the item images along with the item details in Sterling Call Center and Sterling Store. These images can reside on different servers and can be retrieved when needed. You can configure the servers to fetch the item images, and the protocol to be used when fetching the item images. The Item Search screen requires an image to be 32 X 32 pixels in size in order to be configured on the image server. The Item Details screen requires an image that is 180 X 180 pixels in size in order to be configured on the image server.

### Displaying the URL of a Product Web Page for an Item

Sterling Call Center and Sterling Store enable you to display the URL of the product Web page pertaining to an item in the Item Details screen. Configure the Display of a Product Web Page for an Item rule in the Channel Applications Manager by selecting the asset type to be used to display the URL. The getCompleteItemList API is called to retrieve the list of assets that are associated with the item. For more information about configuring the display of a product Web page for an item, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### Displaying the Key Attributes of an Item

Sterling Call Center and Sterling Store enable you to display key attributes pertaining to an item in the Item Details screen. When defining an attribute, you must specify whether the attribute is a key attribute. For more information about key attributes, refer to the *Business Center: Item Administration Guide*.

## Pagination

Sterling Call Center and Sterling Store supports smart retrieval of record sets in the Item Search screen.

For more information about pagination, see [Section 4.22, "Pagination"](#).

### 4.1.1.2 End-User Impact

This section lists the end-user impact of this feature:

- In the Advanced Item Search screen, either the Search Through All Items panel or the Search Items Currently Being Sold panel is displayed based on the Item Search Options configuration.
- The user will see item images if the image server is configured.
- The user will see the View Product Web Page hyperlink in the Item Details screen based on the configuration.
- In the Item Search screen and Item Details screen, the user will see Shipping and Delivery as separate fulfillment methods based on the configuration.
- In the Item Search screen and Item Details screen, the user will see pickup as a fulfillment method based on the configuration.
- The user will see the item availability information based on the check for availability configuration rule.
- If the Enable Scanning Of Items Using Barcode rule is selected, users can scan and enter the alternate item identifiers into all the item entry fields throughout the application.
- When checking for availability, the user will see the country and states drop-down list based on the configuration.
- The user will see the Key Attributes panel based on the Is Key Attribute configuration rule.

### 4.1.1.3 Implementation

This section explains the steps to implement for this feature.

- Sterling Call Center and Sterling Store allow you to configure the type of search to be displayed in the Advanced Item Search screen. For more information about configuring the item search options, see the

*Sterling Selling and Fulfillment Suite: Applications Configuration Guide.*

- Sterling Call Center and Sterling Store allow you to configure the store pickup rules. For more information about configuring the store pickup rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure availability checks and reservation options. For more information about configuring availability checks and reservation options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the server to fetch the images.

Define entries such as name, protocol, base URL, in the `locations.ycfg` file. The Item Search screens within Sterling Call Center and Sterling Store are designed for small item images that are 32 X 32 pixels in size and large item images that are 180 X 180 pixels in size. It is strongly recommended that your images conform to these sizes for ideal display within the screens. Also, ensure that the item images are stored as a `.jpg` or `.gif` image in order to be displayed in the Item Details screen. For more information about configuring the connection settings to fetch images from the server, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

- You can configure the Sterling Call Center and Sterling Store application to display the URL of a product Web page pertaining to an item in the Item Details screen. For more information about configuring the display of a product Web page for an item, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- You can configure the Sterling Call Center and Sterling Store application to enable users to add or search for items using the alternate item identifiers. For more information about scanning of items using bar codes, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- You can define an item attribute as a key attribute. For more information about defining key attributes, refer to the *Business Center: Item Administration Guide*.

#### 4.1.1.4 Reference Implementation

This section lists the default values provided for this feature.

- As part of reference implementation, Sterling Call Center and Sterling Store enable users to search for all the items that are available in the database.
- The YFSGetPromotionsForItemList implementation accepts all the items provided as part of the reference implementation. For more information about the items provided as the reference implementation, see the *Sterling Selling and Fulfillment Suite: Applications Reference Implementation Guide*.
- The default YFSGetPromotionsForItemListUE user exit implementation accepts the following promotions:
  - The Prevent Availability Check When Searching for Items rule is defaulted to "Y".
  - The Allow Country to be Entered for screens that check Item Availability rule is defaulted to "Y".
- By default, the Enable Scanning Of Items Using Barcode rule is selected in Configure Item Search Options of the Channel Applications Manager.
- By default, the following asset types are provided:
  - Audio
  - Video
  - Text
  - Image

#### 4.1.2 Handling Styles for an Item

Some retailers sell products that come in different variations, such as different styles, sizes, and colors, for example, a T-shirt may be sold in three different sizes, 10 different colors, and "v-neck" and "regular neck" styles.

### 4.1.2.1 Solution

Sterling Call Center and Sterling Store enable an enterprise to model items based on style, size, and color. It also enables users to search and view the details of stylized items.

This section explains the Item Inquiry process for stylized items and solution offered by Sterling Call Center and Sterling Store.

### Configuring the Product Catalog

Sterling Call Center and Sterling Store provides the configuration to define an item as a model item in the Product Catalog. If an item is marked as a model item, you can define different styles. Stylized items are those items that share common attributes with the model item, for example, if the model item is a Short-sleeved Polo shirt, possible stylized items will be Small Red Short-sleeved Polo Shirt and Medium Red Short-sleeved Polo Shirt.

You must define a Classification with the Classification Purpose as Distinct Attributes in order to be able to define the set of attributes that identify a stylized item.

When defining stylized items for a model item, Sterling Call Center and Sterling Store assume that the fulfillment method and associations of the stylized items are the same as those of the model item. If you associate a stylized item with an order line containing a related item, Sterling Call Center and Sterling Store assumes that the related item applies to all the stylized items defined for the corresponding model item.

Sterling Call Center and Sterling Store assumes that model items cannot be marked as configured items. Also, it is assumed that configured items do not contain model items as components.

### Item-Style Information Retrieval

Sterling Call Center and Sterling Store provides the functionality to search for model items and stylized items in the Item Search screen.

Sterling Call Center and Sterling Store provides the Item Style pop-up window which displays information about the availability and various styles for a model item. The `getCompleteItemList` API is called to retrieve the distinct attributes for a model item.



The `getItemUOMList` API is used to get the list of UOMs for the item. The `YCDOverrideDeliveryMethodUE` user exit is called to determine the list of available fulfillment methods for the item.

Delivery and Shipping are displayed as separate fulfillment methods based on the configuration for Order Entry Rules.

You can also configure the Sterling Call Center and Sterling Store user interface to never show the Pickup fulfillment method if the enterprise does not support the Pickup fulfillment method based on the configuration for Store Pickup Rules.

### **Style Items pop-up window**

Sterling Call Center and Sterling Store provides a Style Items pop-up window that the CSRs can use to browse and select different combinations of style items. The item styles can be displayed as buttons or as lists, based on the configuration rule, for displaying the stylized items. For more information about configuring the display of Model items, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. If the item style pop-up window is configured to display styles as buttons, the images for the buttons should be defined on an image server.

You can configure the order in which attribute values are displayed on the item style pop-up screen. For example, if you have configured Size as a distinct attribute for a stylized item and this attribute can take the values L, M, S and XL, you can configure that the values appear in the order S, M, L and XL on the style item pop-up.

If a distinct attribute value is not specified for some stylized items but is specified for all other stylized items, Sterling Call Center and Sterling Store displays a literal "None" for the stylized item for which the distinct attribute value is not defined. The display of this literal is bundle controlled. For example, if you have configured Size and Color as distinct attributes and have forgotten to specify the Color for one of the stylized items, the literal "None" will be displayed on the style item pop-up screen for the item.

### **Item Availability Retrieval**

Sterling Call Center and Sterling Store provides the functionality to retrieve availability information for a stylized item. A panel is provided to check for availability if a Ship To address is not entered. The display of

the Country drop-down list on this availability panel is based on the rule to allow Country entry. If you have configured region schemas to display states, the availability panel displays a drop-down list of states. For more information about configuring region schemas, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

Sterling Call Center and Sterling Store enable you to perform either real-time availability checks for all child items or for child items in a cached inventory. If you have configured the Styles pop-up window to perform real-time availability checks, the `getFulfillmentOptionsForLines` API is called to retrieve the list of all child items. However, if you have configured the Styles pop-up window to check the availability of child items in a cached inventory, the `getCompleteItemList` API is called to retrieve the list of all child items for a particular distribution group. Also, in the latter scenario, the Fulfillment Options panel will not be displayed in the Styles pop-up window.

For more information about configuring the availability checks, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

For more information about creating a distribution group, see the *Sterling Global Inventory Visibility: Configuration Guide*.

### **Displaying Item Images and Style, Size, and Color Images**

You can display item images along with the item details in the Item Styles pop-up window. These images may reside on different server and can be retrieved when needed.

You can configure the server to fetch the item images and the protocol to be used when fetching the item images. Additionally, you can also configure to display images for style, size, and color in the Item Styles pop-up window, if item styles are displayed as buttons.

#### **4.1.2.2 End-User Impact**

This section lists the end-user impact for this feature.

- When searching for items, if the user enters a complete or partial Model Item ID in the search criteria of the Item Search screen, the corresponding Model item will be displayed in the search results.
- The Fulfillment Options panel will not be displayed in the Style Item pop-up window if the Sterling Call Center and Sterling Store

applications are configured to perform availability checks for child items in a cached inventory.

- In the Style Items pop-up window, the user will see Shipping and Delivery as separate fulfillment methods based on the configuration for Order Entry Rules.
- In the Style Items pop-up window, the user will see pickup as a fulfillment method based on the configuration for Store Pickup Rules.
- When checking for availability in the Style Items pop-up window, the user will see Country and States drop-down list based on the configuration for country display and region schemas.
- In the Style Items pop-up window, the user will see item styles as buttons or as lists based on the configuration rule for display of style items.
- In the Style Items pop-up window, if images are configured for Distinct Attributes, the buttons display the configured images. If images are not configured, the buttons display the attribute values.
- In the Style Items pop-up window, the user will see a message “No styles are defined for this item” if no child items are associated with the Model Item.
- In the Style Items pop-up window, the user will see a message “Styles are not configured for this item” if the Model item is not associated with any classification definition with the purpose code “Distinct Attributes”.
- In the Style Items pop-up window, the user will see a message “There is a configuration error with the setup for the Style Items, because of which we cannot display the options.” if the same attribute values are defined for two or more item stylized items . For example, if you have configured Size and Color as distinct attributes for items Item1, Item2, and Item3, and the attribute values are as below, the user will see the error message because Item1 and Item2 have the same attribute values for Size and Color.

<sup>51</sup>  
**Table 4–1 Sample Stylized Items**

Item ID	Color	Size
Item 1	Red	S1

**Table 4–1 Sample Stylized Items**

Item ID	Color	Size
Item 2	Red	S1
Item 3	Red	S2

- If you have configured model items as components of a bundle item, while adding these bundle items item to the order, the error message "Cannot include a bundle item with a model item as a component" is displayed on the Add Items screen.

#### 4.1.2.3 Implementation

This section explains the steps to implement for this feature.

- Define the Model items and Child items in the Product Catalog. Define attributes that uniquely define the Child items. The fulfillment methods and associations defined for the Child item should be the same as those defined for the model item. For more information about defining Model items and child items, refer to the *Catalog Management: Configuration Guide*.
- Define a Classification with the Classification Purpose as Distinct Attributes.

Define the attributes for an item and associate these attributes with the classifications pertaining to the stylized items. Further, assign values to the attributes defined. When assigning values, ensure that the combination of values for the child items within a model item are unique.

For more information about defining classifications and attributes, refer to the *Business Center: Item Administration Guide*.

**Note:** When defining the attributes for a classification pertaining to a stylized item, ensure that all the attributes belong to the same attribute group.

- Add entries in the resource bundle and the possible values that can be returned for the distinct attributes of a stylized item. Configure the order in which attribute values are displayed on the Style Items pop-up screen by specifying the appropriate value for the resource

bundle entries. For example, if you have configured Size as a distinct attribute for an item and this attribute takes the values L, M, S and XL, you can configure that the values appear in the order S, M, L and XL, by providing bundle keys for these attributes as below. Note that the alphabetical order of the bundle keys correspond to the order in which the attributes are to be shown.

- Size\_S1\_Small=S
- Size\_S2\_Medium=M
- Size\_S3\_Large=L
- Size\_S4\_Extra\_Large=XL

For more information about resource bundles, see [Section A.2.1, "Resource Bundles"](#).

- You can also configure the order in which the attribute values are displayed on the Style Items pop-up screen by using sequence numbers.

For a given item attribute, configure the sequence numbers for the distinct attribute values in the output of the `getCompleteItemList` API. The attribute values are sorted in the ascending order of the sequence numbers. The bundle keys for the item attribute values are displayed in the sorted order.

The following rules must be applied when configuring the sequence numbers:

- The sequence numbers must be set with the below attribute:  
`AttributeValueSeqNo="<sequence number>"`
- The sequence numbers must be integers.
- The sequence numbers must be unique for the attribute values.
- The sequence numbers must be consistent for the attribute values in the child item elements in the output of the `getCompleteItemList` API.

For the above example, if you have configured the sequence numbers as below;

**Table 4–2 Sample Sequence Numbers**

Attribute Value	Sequence Numbers	Bundle Entry
Size_S1_Small	4	S
Size_S2_Medium	3	M
Size_S3_Large	2	L
Size_S4_Extra_Large	1	XL

the attribute values will be displayed in the following order:

- Size\_S4\_Extra\_Large=XL (1)
  - Size\_S3\_Large=L (2)
  - Size\_S2\_Medium=M (3)
  - Size\_S1\_Small=S (4)
- Sterling Call Center and Sterling Store allows you to configure the server to fetch the item images.  
 Define entries such as name, protocol, and base URL in the `locations.ycfg` file. The Style Size pop-up window is designed for medium-sized images that are 64 X 64 pixels in size. It is strongly recommended that your images conform to this size for ideal display within the screens. For more information about configuring connection settings to fetch images from the server, refer to the *Selling and Fulfillment Foundation: Customization Guide*.
  - To display item styles as buttons or as lists, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
  - To configure images for item style, size, and color, define a new config in the `locations.ycfg` file with the Name set to "STYLE\_ITEM" (see the following example below)

```
<Config Name="STYLE_ITEM"
  Protocol = "http"
  BaseUrl = "127.0.0.1"
  PortNumber = "80"
  ApiUrl = "/COMImages/$param1$.jpg"
  DefaultApiUrl = "/COMImages/404.jpg">
</Config>
```

The names of the images on the server should adhere to the following convention:

`<Attribute Name>_<Attribute Value>`

Also, ensure that the images are stored as a .jpg or .gif image in order to be displayed in the Style Items pop-up window.

For example, if you have configured Color as a distinct attribute and this attribute can take the value Red, the corresponding image on the server should be Color\_Red.jpg. For more information about configuring the connection settings to fetch images from the server, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

- You can configure the Sterling Call Center and Sterling Store applications to perform either real-time availability checks for all child items or for child items in a cached inventory. For more information about configuring the availability checks, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- You can configure whether Delivery and Shipping are to be shown as separate fulfillment methods in the Item Style pop-up window. For more information about configuring the display of model items, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the store pick up rules. For more information about configuring the store pick up rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.1.2.4 Reference Implementation

This section describes the default values provided for this feature.

- The Classification Purpose for Distinct Attributes is bound to the Itemtype attribute. The Classification Values have the following hierarchy:
  - Style Item > Mens Apparel > T-Shirt. The Distinct Attribute Color is defined for Mens Apparel. These attributes are defined for Matrix Incorporated and Heavy-Lift.
  - Style Item > Electronics Variations > Refrigerator. The Distinct Attributes Color, Orientation, and Handle are defined for

Electronics Variations. These attributes are defined for Matrix Incorporated.

- By default, the Sterling Call Center and Sterling Store applications are configured to perform real-time availability checks on all child items.
- The item style display rule is configured to display the button view for all the organizations.

## 4.2 Configured Items

Some retail environments may fulfill orders comprising of configured items. A configured item is a group of items which are related such as a computer which comprises of the processor, keyboard, mouse, and monitor. Configured items are also referred to as bundles.

### 4.2.1 Solution

Sterling Call Center and Sterling Store allows you to configure the bundle fulfillment mode. That is, whether bundle components should be shipped together or independently. For more information about configuring bundle fulfillment mode, refer to the *Catalog Management: Configuration Guide*.

Bundles are supported in the following flows:

- [Order Creation](#)
- [Add Multiple Items to an Order](#)
- [Change Fulfillment Options](#)
- [Change Gift Options](#)
- [Price Match](#)
- [Reship an Item](#)
- [Cancel Order](#)
- [Change Service Instructions](#)
- [Change Service Appointment](#)
- [Return Line Selection](#)

Sterling Call Center and Sterling Store allows you to configure if bundle components should be displayed on the following screens:



- Order Summary
- Shipment Summary
- Return Summary

If the `IsConfigurable` value of an item is "Y", users can configure the item using the Product Configurator. For more information about the Product Configurator, see [Product Configurator](#).

Sterling Call Center and Sterling Store assumes that Configured Items do not contain Model Items as components.

---



---

**Note:** If you have configured model items as components of a bundle item, while adding these model items as components of a bundle item to the order, the error message "Cannot include a bundle item with a model item as a component" is displayed.

---



---

## 4.2.2 End-User Impact

The user will be able to see the bundle parent and components, if the bundle item is configured as ship independent. The user will only see the bundle parent if the bundle item is configured as ship together.

Based on the configuration for display of bundle components, the user will be able to see the bundle components in the following screens:

- Order Summary
- Shipment Summary
- Return Summary

## 4.2.3 Implementation

This section explains the configurations for bundles.

- Sterling Call Center and Sterling Store allows you to configure bundle items as ship together or ship independent. For more information about configuring bundle items as ship together or ship independent, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the display of bundle components on the Order Summary, Return

Summary and Shipment Summary screens. For more information about configuring the display of bundle components, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 4.2.4 Reference Implementation

This section explains the reference implementation provided as part of bundles.

- The Display Component Items For Bundles In Summary Screens rule is set to not display bundle component items.

## 4.3 Alternate Item Identifier

In some retail environments, the enterprise may use a different item identifier than what has been configured in the enterprise's global catalog. These enterprises use this alternate item identifier to scan and search for product or service items in the user interface. The alternate item identifiers are supported for product and service items.

### 4.3.1 Solution

Sterling Call Center and Sterling Store allows you to configure the item alias for alternate item identifiers. For more information about configuring item aliases for alternate item identifiers, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

The user can scan or enter the alternate item identifiers in the Sterling Call Center and Sterling Store application.

If the Enable Scanning Of Items Using Barcode rule is selected in Configure Item Search Options of the Channel Applications Manager, the `translateBarCode` API is called in the following screens to retrieve the original item information for the scanned alternate item identifiers:

- Item Inquiry screen
- Create Order—Add Items screen
- Backroom Pick screen and Customer Pick screen

For more information about the item information retrieval, see [Section 4.1.1, "Searching and Viewing the Details of an Item"](#).

The `getCompleteItemList` API is called on the Large Order Entry Screen to retrieve the original item information for the scanned alternate item identifiers.

**Note:** Barcode translation takes place only when you press the TAB key after scanning the item barcode. The scanner must be configured to send out a TAB message immediately after scanning the barcode. This configuration may change based on the device.

Sterling Call Center and Sterling Store also provides the ability to store an alternate item identifier on an order line.

### 4.3.2 End-User Impact

You can scan the alternate item identifiers into all item entry fields throughout the user interface.

### 4.3.3 Implementation

If you want to use the alternate item ID instead of the original item ID, configure the:

- Store Alternate Item IDs on Order Lines rule.
- Rule to set the item alias for alternate item identifiers.

For more information about configuring the rules to display the alternate item identifier, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- Barcode translation for an item alias configured for an alternate item identifier. For more information about defining barcodes, refer to the *Sterling Warehouse Management System: Configuration Guide*.

---

**Note:** To display the item alias on all screens, you must set the display attributes for the item. To set the display attributes, configure the item attributes and modify the resource bundle key. For more information about configuring item display attributes, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

---

### 4.3.4 Reference Implementation

The reference implementation items are associated with the alternate item identifiers. The following implementations are defined for Matrix Incorporated and Heavy-Lift:

- When configuring an item, you can specify the alternate item identifier in the Item Alias field.
- If you configure the Alternate Item ID rule, the alternate item identifier gets stamped on the order line.

## 4.4 Item Display Options Configuration

You may want to display certain item attributes to different enterprises as the description in Sterling Call Center and Sterling Store user interface. For example, one retailer may consider the item attributes 'ItemID' and 'Item Description' to be the most important attributes, while a different retailer may consider 'ItemID' and 'Manufacturer' to be the most important attributes to display in the user interface.

### 4.4.1 Solution


Sterling Call Center and Sterling Store enables you to configure which item attributes are displayed throughout the user interface. This is done by first configuring the item attributes associated with the item and then editing the resource bundle key.

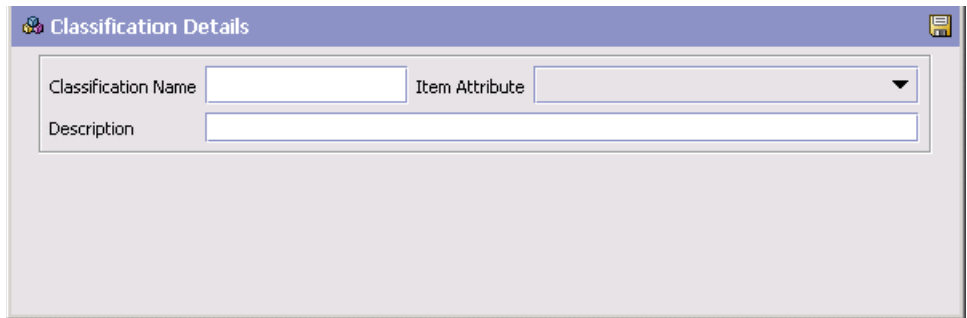
### 4.4.2 End-User Impact

The end user sees the primary item attributes you configure, in the format you specify, throughout the Sterling Call Center and Sterling Store user interface.

### 4.4.3 Implementation


To configure which item attributes are available for display:


1. Create a new classification and associate it with the classification purpose "Display Attributes". For more information about defining classifications, refer to the *Catalog Management: Configuration Guide*.
  - a. From the Channel Applications Manager, select Manage Products and Categories.
  - b. Under Manage Products and Categories, select Define Types of Product Classification. The Classification Definitions tab is displayed.
  - c. Choose . The Classification Details pop-up window is displayed.

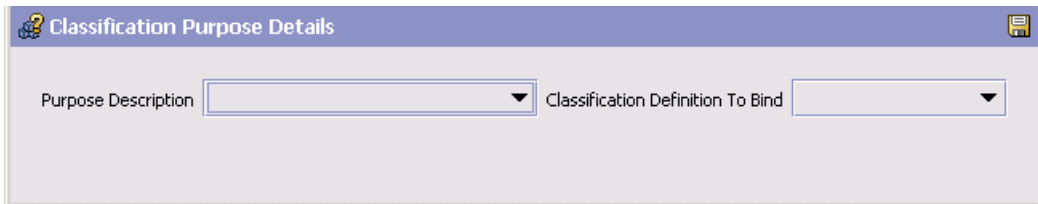


- d. In Classification Name, enter the name of the classification.
- e. From the Item Attribute drop-down list, select the item attribute you want to associate with the classification.

**Note:** The item attributes in the drop-down may display with underscores or without spacing. Disregard this.

- f. In Description, enter a brief description of the classification.
- g. Choose .
- h. From Classification Definition window, select the Classification Purposes tab.


- i. Choose . The Classification Purpose Details pop-up window is displayed.



The image shows a dialog box titled "Classification Purpose Details". It has a light blue header bar with a question mark icon on the left and a save icon on the right. The main area is white and contains two drop-down menus. The first menu is labeled "Purpose Description" and the second is labeled "Classification Definition To Bind". Both menus have a small downward arrow on the right side of their text boxes.

- j. Choose the "Item Display Attributes" classification purpose from the Purpose Description drop-down menu.

**Note:** You cannot assign more than one classification definition to a classification purpose.

- k. Choose the classification definition that you want to bind a classification purpose to from the Classification Definition To Bind drop-down menu.
- l. Choose .

- m. Define a Classification Hierarchy for the new classification and associate it with the item attribute groups and attribute values. For more information about defining classifications, refer to the *Catalog Management: Configuration Guide*.

If you have configured item aliases for alternate item identifiers, ensure that the item alias attribute is included in the classification hierarchy details. For more information about alternate item identifiers, see [Section 4.3, "Alternate Item Identifier"](#).

2. Modify the resource bundle.

Sterling Call Center and Sterling Store uses resource bundle keys to display item information on all screens. The resource bundle keys define the formatting and sequencing of the item attributes you choose when configuring the classification values.

To allow new or modified item attributes to display on a screen in Sterling Call Center and Sterling Store, you must edit the ItemDisplay bundle keys.

To edit the ItemDisplay bundle keys:

- a. From the Sterling Call Center and Sterling Store root directory > Plugins, select the `plugins\com.yantra.pca.ycd.rcp_1.0.0` folder.
- b. Select the `com.yantra.pca.ycd_format_bundle.properties` file located in the `resources.jar` file.
- c. In the `com.yantra.pca.ycd_format_bundle.properties` file, search for the following bundle keys:
  - `ItemDisplayConfigurationForLineDisplay`
  - `ItemDisplayConfigurationForLineDisplayNoAttributeList`
  - `ItemShortDescDisplayConfiguration`
  - `ItemShortDescDisplayConfigurationNoAttributeList`
  - `ItemIDDisplayConfiguration`
  - `ItemIDDisplayConfigurationNoAttributeList`

You can modify the value that appears to the right side of the equal sign ("=") in these bundle entries. The right side can contain any formatting characters, such as dashes, semicolons, and so forth. The right side can also contain place holders for the item attributes that you have made available through the configuration in the following format: {x}, where 'x' is an integer representing the index of the item attribute you want to display.

For example, a classification has the following item attributes:

Display Attributes		
Item Attribute	Description	Is Key Attribute
ShortDescription	ShortDescription	<input type="checkbox"/>
ColorCode	ColorCode	<input type="checkbox"/>
UnitOfMeasure	UnitOfMeasure	<input type="checkbox"/>
ItemID	ItemID	<input type="checkbox"/>

The place holders are available in the same sequence as the display attributes. In this example, the following place holders are available for use in the resource bundle:

```
# {0}-ShortDescription  
  {1}-ColorCode  
  {2}-UnitOfMeasure  
  {3}-ItemID
```

The resource bundle can be changed from this:

```
ItemIDDisplayConfiguration={3},{2},{0}
```

which displays the item ID, the unit of measure, and the short description in the Sterling Call Center and Sterling Store user interface, to this:

```
ItemIDDisplayConfiguration={3},{0},{2}
```

which displays the item ID, the short description, and the unit of measure in the Sterling Call Center and Sterling Store user interface.

### 4.4.4 Reference Implementation

None.

## 4.5 Alternate Store Pickup Locations Search

While creating an order, the customer may want to know the stores at which the items are available. In such situations, call center and store representatives may need to retrieve the availability information for the requested items.

### 4.5.1 Solution

Sterling Call Center and Sterling Store provides a way to search across stores to determine which ones have items available for pickup.

This section explains the Alternate Store Pickup Locations Search task and the solution offered by Sterling Call Center and Sterling Store.

The Alternate Stores screen displays a list of items and nearby stores where the item is available. You can configure which stores should be displayed in this screen. The `getSurroundingNodeList` API is called to



retrieve the list of surrounding nodes. The findInventory API is called to retrieve the inventory availability information.

### Store Pickup Rules

You can configure the following store pick rules which are used to retrieve results for the Alternate Store Pickup Locations Search.

- Distance To Consider When Finding Nearby Store
- Distance To Consider When Finding More Store
- Distance UOM
- Standard Search Fulfillment Type
- Extended Search Fulfillment Type

For more information configuring these rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### Availability Information Retrieval

The display of the country drop-down list on the availability panel is based on the rule to allow country entry. If you have configured region schemas for display of states, the availability panel displays a drop-down list for states. For more information about configuring region schemas, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

## 4.5.2 End-User Impact

Based on the configuration to display country and state drop down lists, the end-user will see country and state drop down lists in the Alternate Stores screen.

## 4.5.3 Implementation

This section explains the configurations for the alternate store pickup locations search task.

- Sterling Call Center and Sterling Store allows you to configure Store Pickup rules. For more information about configuring the store pickup rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- Sterling Call Center and Sterling Store allows you to configure the display of country and state drop down lists. For more information about configuring the display of country and state drop down lists, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 4.5.4 Reference Implementation

This section explains the reference implementation provided as part of alternate store pickup locations search task.

- The default value for 'Distance Radius to Consider When Finding Nearby Stores' is 25.
- The default value for 'Distance Radius to Consider When Finding More Stores' is 50.
- The default value for 'Distance UOM' is Mile.
- The default value for 'Standard Search Fulfillment Type' is Fulfillment Type for Product Sourcing.
- The default value for 'Extended Search Fulfillment Type' is Fulfillment Type for Product Sourcing.

## 4.6 Order Capture

For orders to be fulfilled within Sterling Call Center and Sterling Store, the relevant data must be captured from the Sterling Call Center and Sterling Store user interface, a Web portal, or other external order entry systems. This involves accurately storing all the information relevant to an order internally, and after that is done, ensuring that the information captured is valid, and can be processed.

This section describes the following:

- [Order Creation](#)
- [Capturing Orders through Other Channels](#)
- [Order Delay](#)
- [Order Validation](#)

## 4.6.1 Order Creation

Sterling Call Center and Sterling Store receives the order-related data entered through customer order channels such as a retailer Web site, and stores them internally for them to be processed. Call center agents and store representatives can also create orders.

### 4.6.1.1 Solution

Using the Create Order task, users who have access to multiple enterprises, can select the enterprise for which an order has to be created. It also enables the user to identify a customer and create an order for that customer. The user can enter multiple order lines for an order, modify the fulfillment options and appointments, and confirm the payment information as requested by the customer. Sterling Call Center and Sterling Store provides a set of rules for the create order task. You can use these rules to configure the order creation task to meet your specific requirements.

Using the Customer Screen Sequence and Line Entry Sequence rules, you can switch the order of the Customer Identification and Add Line steps. For more information about Configuring the Customer Screen Sequence and Line Entry Sequence rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

Validations are carried out for items and the quantity of items in an order in the Customer Identification screen and the Add Items screen. For more information about validating items, see [Section 4.6.4.4, "Item Validation"](#).

If you create an order for a contact who does not have a User ID, that contact's payment methods will not be visible when you reach the payment confirmation screen. For more information about creating a contact, see [Section 4.16.1, "Create Consumer Customer"](#).

In the Sterling Call Center and Sterling Store, you cannot create an order for an anonymous customer whose record was created in the Sterling Multi-Channel Selling Solution. You must create the customer in Sterling Call Center and Sterling Store.

When creating an order for a customer or contact, the status of the customer or contact must not be Inactive. If the status is Inactive, an error message is displayed.

The steps involved in the order creation process in Sterling Call Center and Sterling Store are:

1. [Identifying a Customer](#)
2. [Adding Items to an Order](#)
3. [Opening an Existing Order from the Large Order Item Entry Screen](#)
4. [Changing the Fulfillment Options](#)
5. [Viewing Pickup and Shipping Fulfillment Summary](#)
6. [Changing Service Appointments](#)
7. [Viewing the Service Fulfillment Summary](#)
8. [Viewing and Confirming Payment Details](#)
9. [Entering a Name for a Draft Order](#)

### **Identifying a Customer**

This section explains the customer identification step in the create order task, and the solution offered by Sterling Call Center and Sterling Store.

- [Identifying or Searching for a Customer](#)
- [Skipping the Customer Identification step](#)

If a customer contact ID is entered in the order header, the YCD\_SetOrderDefaults\_2.0 or YCD\_SetOrderDefaultsOnChange\_2.0 services are invoked as the beforeCreateOrderUE or beforeChangeOrderUE user exits, respectively, that are called by the createOrder API or changeOrder API. The getCustomerContactList API is called to populate the order with the first name, last name, e-mail address, and day phone number of the customer contact. If a customer contact ID is not entered in the order header, the billing address on the order is used to populate the first name, last name, e-mail address, and day phone number fields. The postal code in the order is always retrieved from the customer's Bill To address.

If a customer or contact does not have default addresses or payment methods defined, the default addresses or payment methods of the closest customer in the customer hierarchy are used.

## Identifying or Searching for a Customer

Sterling Call Center and Sterling Store provides the functionality to search for a customer based on three options:

- [Advanced Customer Search](#)
- [Search Based on Address](#)
- [Basic Customer Search](#)

### Advanced Customer Search

The Advanced Customer Search screen is recommended if the user wants to place orders for the customers maintained by Sterling Call Center and Sterling Store. This screen provides a full set of features to search against your database. Using the Advanced Customer Search option in Order Entry rules enables you to use of the more detailed information displayed by the Advanced Customer Search screen as well as associating additional addresses and payment methods back to the customer. Use the Advanced Customer Search screen if you want to access the customers created in the Sterling Call Center and Sterling Store.

When searching a customer master database, you can enable the search for business customers by configuring the Business Customers Enabled rule. You can also configure the default customer type when searching for a customer. For more information about configuring the display of the Customer Identification screen, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

A user can add a new contact for a business customer in the Customer Identification screen. When the manageCustomer API is called at the end of the corresponding process, the new contact is saved against the selected customer record. The getCustomerList API is called to update the details of the customer after the contact is added.

A user can enter a new address or modify an existing address in the Customer Identification screen. For more information about the Address Search screen, see [Section 4.6.4.2, "Address Verification"](#).

When creating a customer from the Advanced Customer Search screen, the Create Customer wizard is loaded in a dialog box. [Section 4.16, "Customer Creation"](#) contains more information about this task.

When a new address is added or modified using the Advanced Customer Search screen, the new address is saved against the selected customer record or contact record through a `manageCustomer` API call.

When creating an order for a business customer from the Advanced Customer Search screen, a user can allow tax exemptions on the order by configuring the Change Tax Options for a business customer rule. If only one currency is defined for the customer, and this is not the same as the currency defined for the enterprise organization, the customer's currency is applied to the order through the `createOrder` API or the `changeOrder` API call.

The Advanced Customer Search screen contains validations to ensure that the end user enters enough information for a search. If this screen is extended to contain additional fields, these validations are not performed. This enables you to create validations based on the fields that you added. For more information about customizing screens, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

### Search Based on Address

If you do not maintain a customer database, you can choose to display Address Search screen which allows a user to search among the existing addresses in the system.

In a multischema setup, the addresses may be present in multiple transaction schemas. The `getOrganizationHierarchy` API is called to retrieve the colony to which the enterprise belongs. The `getPersonInfoList` API is called to retrieve the addresses based on the search criteria entered and the colony to which the enterprise belongs.

The `getPersonInfoList` API is called to retrieve the addresses based on the search criteria entered. The `getConsumerDetails` API is used to retrieve the details of the consumer associated with the address. If a consumer corresponding to the address is not found, and if corporate users are enabled for the corresponding enterprise, the system attempts to find a corporate user associated with the address.

If the Customer Identification screen is the first screen in the create order flow, this screen calls the `createOrder` API. Otherwise, the `changeOrder` API is called to save the address on the order.

The user can enter a new address or modify an existing address in the Address Search screen. For more information about the Address Search screen, see [Section 4.6.4.2, "Address Verification"](#).

The Address Entry screen performs the necessary validations to ensure that the end user enters enough information for a search. If this screen is extended to contain additional fields, these validations are not performed. This enables you to create validations based on the fields that you added. For more information about customizing screens, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

### Basic Customer Search

The Basic Customer Search screen is not recommended for new installations. It is only provided for backward compatibility.

Sterling Call Center and Sterling Store provides a configuration to enable searches based on customer types. You can also configure the default customer type to be used when searching for a customer. For more information about configuring the customer types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

The `getCustomerList` API is used to retrieve the details of the business customer or consumer customer. If the Customer Identification screen is the first screen in the create order flow, this screen calls the `createOrder` API. Otherwise, the `changeOrder` API is called to save the address on the order.

The user can enter a new address or modify an existing address in Customer Identification screen. For more information about the Address Search screen, see [Section 4.6.4.2, "Address Verification"](#).

The Basic Customer Search screen contains validations to ensure that the end user enters enough information for a search. If these screens are extended to contain additional fields, these validations are not performed. This enables you to create validations based on the fields that you added. For more information about customizing screens, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

### Skipping the Customer Identification step

The user has the option of skipping the customer identification step.

- If the user is using the Default Address Entry/Search screen, the user will be prompted to enter the billing address before the completion of the order capture.
- If the user is using one of the Customer Search screens, the Customer Search screen is displayed again before the order is captured. At this point, the user can either search for or create a

customer for use on the order, or can enter a billing address. In such a situation, no customer record will be associated to the order.

If the user skips the customer identification step, the items and the quantity of the items in the order are validated in the Customer Identification screen after the user adds the items to the order. The `validateItemForOrdering` API is called to validate the items in the order. If the items in the order fail validation, the Add Items screen is displayed. For more information about item validation, see [Section 4.6.4.4, "Item Validation"](#).

### Adding Items to an Order

This section explains the item entry step in the create order task, and the solution offered by Sterling Call Center and Sterling Store. Adding items to an order comprises the following tasks:

- [Alternate Views for the Add Items Screen](#)
- [Identifying Items](#)
- [Display of Columns in the Add Items screen](#)
- [Configuring the Focus of the Cursor in the Add Items or the Large Order Entry Screen](#)
- [Adding Or Deleting Lines From An Order](#)
- [Customizing the Way Items are Added to an Order](#)
- [Alternate Item Identifiers](#)
- [Item Entitlement](#)
- [Adding Stylized Items or Modifying an Item's Style](#)
- [Product Configurator](#)
- [Adding Related Items or Replacing Items](#)
- [Viewing Updated Price Details and Price Breakdown](#)
- [Displaying the Fulfillment Methods](#)
- [Checking for Availability](#)
- [Disabling Pickup from Store for all Items](#)



### Alternate Views for the Add Items Screen

Sterling Call Center and Sterling Store provides the functionality to search for items based on two options:

- Large Order Item Entry (with a table that is designed to handle a large number of items)
- Add items (with panels for each order line)

Sterling Call Center and Sterling Store provides a Large Order Item Entry screen that is designed to handle a large number of items. You can choose to either display this screen always, or when the number of order lines exceeds a preconfigured number. You can also opt to never display this screen. If your enterprise fulfills orders with a large number of order lines, it is recommended to use the Large Order Item Entry screen. For more information about configuring the display of Large Order Item Entry screen, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. If you choose to display the Large Order Item Entry screen, it is displayed when performing the following tasks:

- Create Order
- Add Lines to Order
- Change Item Style
- Return Entry: Add Exchange items

### Identifying Items

If you have configured to display the Add Items screen with panels, when the user scans the item ID using the barcode scanner, the `translateBarCode` API is used to translate and validate the barcode of the item. If alternate item identifiers are entered or scanned, the `translateBarCode` API is called to retrieve the original item information for the scanned alternate item identifiers. For more information about alternate item identifiers, see [Section 4.3, "Alternate Item Identifier"](#). However, the `translateBarCode` API is called only if the Enable Scanning Of Items Using Barcode rule is selected in Configure Item Search Options of the Channel Applications Manager. For more information about the item information retrieval, see [Section 4.1.1, "Searching and Viewing the Details of an Item"](#). The `getCompleteItemList` API is called to retrieve descriptions, price, accessories, and units of measure for the item.

If you have configured to display the Large Order Entry screen, when the user scans item ID using the barcode scanner, the `getCompleteItemList`

API is used to translate and validate the barcode of the item. If alternate item identifiers are entered or scanned, the `getCompleteItemList` API is called to retrieve the original item information for the scanned alternate item identifiers. For more information about alternate item identifiers, see [Section 4.3, "Alternate Item Identifier"](#). The `getCompleteItemList` API is used to retrieve descriptions, price, accessories, and units of measure for the item.

---

---

**Note:** When searching for an item, if item entitlement has been configured for the customer, the `getCompleteItemList` API returns only items that have been configured for that customer.

---

---

The `getCompleteItemList` API is called to retrieve the details of the item, and the item is added to the order.

When invoking the `getCompleteItemList` API, in addition to retrieving the details of the original item, Sterling Call Center and Sterling Store also retrieves the details of the associated items that have been configured to substitute the original item. If the original item meets all of the following conditions, the user is prompted to add the item configured to substitute the original item:

- The item's effective end date has expired.
- The item has an associated item, whose Association Type is configured as Supersession.

---

---

**Note:** Sterling Call Center and Sterling Store does not take into consideration the associated quantity for an associated item whose association type is configured as Supersession.

---

---

- The Item Substitution On Order Allowed rule is set to "No" in the Applications Manager. For more information about defining a product item's sourcing and transportation, refer to the *Catalog Management: Configuration Guide*.

In an order, a user cannot add an item whose effective start date and effective end date are either before the current date or after the current date. For example, if the current date is 05/26/2009, and a user

attempts to add an item whose start date is 01/01/2009 and end date is 04/30/2009, Sterling Call Center and Sterling Store displays an error message.

Sterling Call Center and Sterling Store does not allow users to add items that are always tag-controlled to an order.

### **Display of Columns in the Add Items screen**

You can configure whether line quantity, UOM, line price and line total columns need to be displayed on the Add Items screens. For more information about configuring the columns to be displayed in the Add Items screen, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### **Configuring the Focus of the Cursor in the Add Items or the Large Order Entry Screen**

You can configure the focus of the cursor after adding an item to an order. Usually, for an enterprise for which multiple quantities are present in a single order line (mostly Business-to-Business), the practice is to enter the quantity after the item is added to the order. However, for an enterprise for which a single quantity is present in a single order line (mostly Business-to-Consumer), it is recommended that the focus is placed on the next Item ID entry field instead of the Quantity field. For more information about configuring the focus of the cursor, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

If a user adds multiple items from the Item Search screen, the focus of the cursor shifts to the next Item ID field and the default value displayed in the Quantity field is 1.

If a user adds an item from the Related Items pop-up window or panel, depending on the configuration, the focus of the cursor shifts to the next Item ID field. However, the quantity of the related item that is displayed is derived by multiplying the Associated Qty with the quantity of the parent item. If the user has not specified any quantity for the parent item, the value of the parent item's quantity is taken as 1, and subsequently, the quantity of the related item that is displayed is derived by multiplying the Associated Qty with 1.

If a user adds an item from the Style Items pop-up window, the focus of the cursor shifts to the next Item ID field and the default value displayed in the Quantity field is 1.

If a user is adding a single item from the Item Search screen, the behavior of the cursor and the quantity of the item that is displayed depends on the way the cursor's focus is configured.

- If the focus of the cursor is set to the next Item ID field, the cursor shifts to the next Item ID field and the default value displayed in the Quantity field is 1.
- If the focus of the cursor is set to the next editable field (Quantity), the cursor shifts to the next editable field and no default value is displayed in the Quantity field.

### **Adding Or Deleting Lines From An Order**

If the add items screen is the first screen in the create order flow, this screen calls the createOrder API. Otherwise, the changeOrder API is called to add lines to the order. The changeOrder API is also called to delete existing lines from the order.

**Note:** When a user deletes an order line in the Add Items screen, the changeOrder API is called to delete the order line from the order. However, in the Large Order Item Entry screen, the changeOrder API is called to delete the order line only when the user navigates to the next screen.

Sterling Call Center and Sterling Store assumes that the status modification rules allow addition of lines to cancelled orders. If the status modification rules do not allow addition of lines to cancelled orders, the user will not be able to add new lines to the order and remove all existing lines at the same time.

### **Customizing the Way Items are Added to an Order**

Sterling Call Center application provides a way to add items to an order using its extensibility feature. This feature can be used by users to customize the application to support the task of adding lines to an order from a customer's wish list.

---

**Note:** The items that are being added to an order should be valid items. The Add To Order utility method (`YCDExtensionUtils.addToOrder (Element ItemList);`) does not validate the items. Also, Delivery Service items cannot be added to an order using the extensibility feature.

---

- **Adding Standalone Items using Extensibility**

The `YCDExtensionUtils.addToOrder (Element ItemList);` utility method can be used to add a list of standalone items to both the Add Items screen and the Advanced Add Items screen. The `ItemList` element should have the following attributes:

```
<ItemList CallingOrganizationCode=" " >
  <Item ItemID=" " UnitOfMeasure=" " ItemGroupCode=""/>
</ItemList>
```

For the list of the input XML attributes, see [Table 4–3](#)

- **Adding Stylized Items to the Advanced Add Items Screen using Extensibility**

The `YCDExtensionUtils.addToOrder (Element ItemList);` utility method can be used to add stylized items to the Advanced Add Items screen using extensibility. The `ItemList` element should have the following attributes:

```
<ItemList CallingOrganizationCode=" " >
  <Item ItemID=" " UnitOfMeasure=" " ItemGroupCode=""/>
</ItemList>
```

Here, `ItemID` indicates the Item ID of the model item.

When adding a stylized item to the Advanced Add Items screen, the system displays the Style Size pop-up window in which you can enter the details of the item. If you are adding multiple model items, the system will not display the Style Size pop-up window. Click the Choose Style hyperlink in the Item panel to select the item style.

- **Adding Stylized Items to the Add Items Screen using Extensibility**

The `YCDExtensionUtils.addToOrder (Element ItemList);` utility method can be used to add stylized items to the Add Items screen using extensibility. The `ItemList` element should have the following attributes:

```
<ItemList CallingOrganizationCode=" " >
  <Item ItemID=" " UnitOfMeasure=" " ItemGroupCode="">
    <PrimaryInformation IsModelItem="" />
  </Item>
</ItemList>
```

Here, `ItemID` indicates the Item ID of the model item. When adding a stylized item to the Add Items screen, the system displays the Style Size pop-up window in which you can enter the details of the item. If you are adding multiple model items, the system will not display the Style Size pop-up window. Click the Choose Style hyperlink in the Item panel to select the item style.

- **Adding Related Items using Extensibility**

Related items can be added to the Advanced Add Items screen using the `YCDExtensionUtils.addToOrder (Element ItemList);` utility method. The `ItemList` element should have the following attributes:

```
<ItemList CallingOrganizationCode=" " ParentUniqueKey="">
  <Item ItemID=" " UnitOfMeasure=" " AssociatedQuantity=""
    ItemGroupCode=" " AssociationType="" />
</ItemList>
```

For the list of the input XML attributes, see [Table 4–3](#).

---

**Note:** If the ParentUniqueKey is not passed, the items will be added as standalone items. Also, the Association Type is mandatory to form the correct relationship between the PROD parent items and the PROD child items.

---



---

**Note:** When adding items to an order, the items can be added either as a list of stand alone order lines or a list of related items to an existing parent order line.

---



---

**Note:** This feature is currently supported only for the Advanced Add Items screen and not for the Add Items screen.

---

- **Adding Multiple Stylized Items from the Style Size Pop-up Window using Extensibility**

Sterling Call Center and Sterling Store provides a way to add multiple stylized items to an order without closing the Style Size pop-up window.

You can add a new control, for example, a button, that enables you to add the items. When you click this control, you need to get the "SelectedItem" model from the screen and form an input element that is similar to that provided in the following example:

```
<ItemList CallingOrganizationCode=" ">
  <Item Currency="USD" ItemGroupCode="PROD"
    ItemID="100700" OrganizationCode="Matrix-R"
    Quantity="1.0" UnitOfMeasure="EACH">
  </Item>
</ItemList>
```

Here, the Item element is the "SelectedItem" model.

The `YCDEExtensionUtils.addToOrder (Element ItemList);` utility method can be invoked by using the ItemList element given in the previous example.

**Note:** You should provide the complete information pertaining to the items that are being passed. If you do not provide the complete information, the system displays the Style Size pop-up window again.

- Input XML for Adding Items using Extensibility**

The input XML for adding items using the extensibility feature is as follows:

```
<ItemList CallingOrganizationCode=" " ParentUniqueKey="">
  <Item ItemID=" " UnitOfMeasure=" " AssociatedQuantity=" "
    ItemGroupCode=" " AssociationType=" " >
    <PrimaryInformation IsModelItem=" " />
  </Item>
</ItemList>
```

**Table 4–3** *Input XML for Adding Items using Extensibility*

Attribute	Description
CallingOrganizationCode	The CallingOrganizationCode at the ItemList level that must match the EnterpriseCode of the order.
ParentUniqueKey	TransactionallineID of the item that should be the parent item of the ItemList Element.
ItemID	Item ID
UnitofMeasure	Unit of measure
AssociatedQuantity	The quantity to be associated with the parent item. This is necessary if the item is added as a child line.
ItemGroupCode	Item Group Code
AssociationType	The association that is to be set up between the items, for example, cross sell, up sell, and so on.
IsModelItem	The information required when adding the model items to the Add Items screen.



---

**Note:** To add a stylized item from the Style Size pop-up window, the item element that is passed should be the `SelectedItem` model in the Style Size pop-up window.

---

### Alternate Item Identifiers

Sterling Call Center and Sterling Store allows you to configure the item alias for alternate item identifiers (IDs). For more information about alternate item identifiers, see [Section 4.3, "Alternate Item Identifier"](#).

### Item Entitlement

If a user keys in an Item ID directly into the Item field, this Item ID will be validated against the item entitlement in the Sterling Multi-Channel Selling Solution based on the customer's product entitlement from the order. If the customer information is not present in the order, the anonymous customer will be used.

If a customer does not have entitlement to an item but the user does, the user can see the item, but cannot add it to an order. An attempt to add the item to an order will cause an Invalid Item error.

### Adding Stylized Items or Modifying an Item's Style

A user can add a stylized item or modify an item's style based on the user permissions and order status modification rules. For more information about changing an item's style, see [Section 4.9.31, "Change an Item's Style"](#).

### Product Configurator

A user can access the Product Configurator from within the Sterling Call Center and Sterling Store applications. Modify the `locations.ycfg` file to enable the user to access the Product Configurator. For more information about modifying the `locations.ycfg` file, see the *Sterling Call Center and Sterling Store: Deployment Guide*.

If the `IsConfigurable` value of an item is "Y", a Configure button is displayed in the Add Items screen. On clicking this button, a pop-up window, which contains a browser that launches the Product Configurator installed in the Business Center application, is displayed.

After an item has been configured using the Product Configurator, an XML file that contains the updated item configuration is returned to the Add

Items screen when the user clicks the Apply button. This information is then passed to the changeOrder API. Note that until the changeOrder API is called, the price and availability information pertaining to the configured item is not displayed in the Add Items screen.

If the IsPreconfigured value of the item is also "Y", users do not have to manually configure the item. The item will automatically select the predefined components, and add them to the order when the changeOrder API is called.

If either the configurable item has not been configured or if it contains an invalid component, the validateItemsForOrdering API will validate the item and display suitable error messages if the item contains an invalid configuration.


When reconfiguring a configurable item, a REPLACE action is set on the parent order line during the changeOrder API call. This will prevent loss of the reservations that exist on the components that have not been modified.

For more information about reconfiguring configurable items, see the *Business Center: Item Administration Guide*.

### **Adding Bundle Items**

If a user enters a Bundle Parent Item ID when adding items to an order, all the component items will be added as different order lines in the order along with the bundle parent item, and the bundle will appear as a single item in the Add Item screen and the Large Order Item Entry screen.

A user can view the components of the bundle item using one of the following options:

- In the Add Items screen, clicks the  icon.
- In the Large Order Item Entry screen, clicks the View Components hyperlink.

For more information about bundles, see [Section 4.2, "Configured Items"](#).

### **Adding Related Items or Replacing Items**

Sterling Call Center and Sterling Store provides the capability to display the associated items for an order line and add the associated items or replace the original items with the associated items.

You can configure whether the related items should be displayed in a pop-up window or a panel at the bottom of the screen when a user views the associated items for an item on the order. For more information about configuring the display of related items, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. If you have configured to display the Large Order Entry screen, Sterling Commerce recommends that you configure to display the Related Items screen as a pop-up window.

If the screen resolution of your system is set to 800 X 600, it is recommended to configure the related items to display as a pop-up window in the Add Items screen.

You can configure which associated items should be displayed in the Related Items pop-up window or panel. For more information about configuring which items should be displayed, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. The `getAssociationRelationshipListAPI` is called to retrieve the associated items that can be displayed in the Related Items panel.

You can configure whether an associated item should be added as a related item or the original item should be replaced by the associated item when a user adds an associated item to an order line. For more information about configuring whether an associated item should be added to an order line or replace the original item, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

Sterling Call Center and Sterling Store allows you to configure associations for items. For more information about associations, refer to the *Catalog Management: Configuration Guide*.

You can define the line relationship types to be used when linking two related lines together. The relationship types defined should support sorting. This ensures that the order lines displayed in Sterling Call Center and Sterling Store are ordered based on the relationship type. For more information about line relationship types, refer to the *Sterling Distributed Order Management: Configuration Guide*.

After the associations and line relationship types are defined, you can specify the line relationship type to be used when related product items are added to an order. For more information about specifying relationship types for associated product items, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. The `getAssociationRelationshipListAPI` is called to retrieve information.

You can specify whether inventory checks should be performed for associated items. For more information about configuring inventory checks for associated items, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. If inventory for associated item is not available and the inventory check required rule is enabled, the associated item is not displayed on the UI. The `getAssociationRelationshipListAPI` is called to retrieve information about which associated items need to be displayed in the related items panel.

When the related items are added to the order, the fulfillment methods for the related items are copied from that of the parent item. If the related item does not support the fulfillment method, the fulfillment option of the related item is set to the default fulfillment method. When you select the parent item to modify the fulfillment option, the related lines associated with it are also selected for modification.

Sterling Call Center and Sterling Store provides a configuration to specify the line relationship type to be used when related service items are added to the order. For more information about specifying relationship types for associated service items, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### **Viewing Updated Price Details and Price Breakdown**

If the addition or modification of an order line results in charges or discounts being applied to the order or order lines, or a change in price details, Sterling Call Center and Sterling Store provides an Update Order button to enable users to view the payment impact before navigating to the next screen. The user can view the updated order total and order line total along with the charges and discounts applied to the order after the user updates the order by clicking the Update Order button. For example, if a pricing rule states that for every 10 quantity of Item1 ordered, the customer can get a \$100 discount, the user will be able to view the discount after the user clicks the Update Order button. However, the Update Order button is displayed only if the user has the necessary permission to use it.

If permission has not been turned on for the Update Order button, Sterling Call Center and Sterling Store does not display the price breakdown of an order line, and the extended price is displayed as the line total.

The user can click the Subtotal value hyperlink to view the price breakdown of the order in the Order Pricing Summary pop-up window. If

there are any changes made to the order, the `changeOrder` API is called to update the order. The `getCompleteOrderDetails` API is used to display the price breakdown of the order. The user can also view the details of the charges and discounts applied to the order by clicking the charges and discounts value hyperlinks from the Add Items screen and the Order Pricing Summary pop-up window. The `getChargeNameList` API is called to display the charge names associated with the charges and discounts applied to the order, and the `getCompleteOrderDetails` API is used to display the charges and discounts.

The `changeOrder` API or `createOrder` API is called to refresh the Add Items screen after the user clicks the Update Order button. The `YFSOrderRepricingUE` user exit can be called as part of the `changeOrder` API or `createOrder` API call to update the appropriate prices, charges, and discounts for the order.

When an existing draft order is opened, repricing is done on the order.

### **Displaying the Fulfillment Methods**

The `getCompleteItemList` API is called to retrieve the fulfillment methods that are applicable to the item. Based on the configuration, Delivery and Shipping may be displayed as separate fulfillment methods. You can also configure the default fulfillment method. For more information about configuring Delivery and Shipping as separate fulfillment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

The `YCDOVERRIDEDELIVERYMETHODUE` user exit is used to override the fulfillment methods defined in the catalog. This user exit provides the ability to override the logic of enabling a fulfillment method. The default implementation is not provided, and you can use your own implementation for this user exit.

### **Checking for Availability**

Sterling Call Center and Sterling Store provides a panel to check for availability if the Ship To address is not present in the order. The display of the Country drop-down list on this availability panel is based on the rule to allow country entry. If you have configured region schemas for displaying states, the availability panel displays a drop-down list for states. For more information about country display and configuring region schemas, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

The `getFulfillmentOptionsForLines` API is used to determine the availability for product items, bundle items, and related items. The `findInventory` API is used to determine whether a location is serviceable for a Provided Service Item.

A user can disable the availability checks when the inventory is maintained externally because the user is aware of the inventory availability details. This aids in better performance and pace in capturing the order.

If the Prevent Initial Availability Checks During Order Entry and Order Modifications rule is disabled or set to "N", the availability check will not be performed, and the calls to `getFulfillmentOptionsForLines`, `findInventory`, or its derivation APIs will be eliminated on the Line Entry screen. For more information about configuring availability checks and reservations, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

When related items are configured to be displayed in a panel, the availability information of the related items is not updated when the quantity is changed for the parent item. It is updated only when the focus is reset to the parent order line.

### **Disabling Pickup from Store for all Items**

You can configure the user interface to never show the Pickup fulfillment method if the enterprise does not support that fulfillment method. For more information about configuring store pickup rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. If you have configured the UI to never display the pickup fulfillment method, Sterling Call Center and Sterling Store assumes that you will not set the default fulfillment method for order lines as pickup.

### **Opening an Existing Order from the Large Order Item Entry Screen**

Sterling Call Center and Sterling Store provides a way to open an existing order from the Large Order Item Entry screen by using the extensibility feature. A new `refreshAddItem(String orderHeaderKey)` method is exposed in the `com.yantra.pca.ycd.rcp.exposed.YCDExtensionUtils` utility method. Using the order header key as the input, this method opens the corresponding order in the Large Order Item Entry screen. You can invoke this method from the extension behavior class of the Large Order

Item Entry screen as `YCDExtensionUtils.refreshAddItem(String orderHeaderKey)`

If an order is already open in the client, and if you pass the order header key corresponding to this order, this method transfers the focus to the open order. In addition, this action does not refresh the open order.

---

**Note:** If you need to refresh the Large Order Item Entry screen, use the REFRESH action provided by the Rich Client Platform Framework.

---

## Changing the Fulfillment Options

When a customer requests a change in the fulfillment options for an item, you can modify the fulfillment options as requested, for example, a customer may request the delivery of ordered items, but later decide to pick up the items from a store. In such situations, the user can modify the fulfillment options for the items or a provided service line. For more information about changing the fulfillment options, see [Section 4.9.19, "Change Fulfillment Options"](#).

## Viewing Pickup and Shipping Fulfillment Summary

The Fulfillment Summary screen provides the complete scenario of how the pickup and ship order lines are fulfilled. The items are reserved based on the configuration. For more information about fulfillment summary, see [Section 4.9.20, "Fulfillment Summary"](#).

## Changing Service Appointments

The user can change the delivery appointment for the customer, if applicable. The items are reserved based on the configuration. For more information about service appointments, see [Section 4.9.23, "Change Service Appointment"](#).

## Viewing the Service Fulfillment Summary

After adding items to the order using the `changeOrder` API, if you choose the Delivery option, the `generateWorkOrder` API is used to create the delivery lines and generate work orders. For more information about service fulfillment summary, see [Section 4.9.24, "Service Fulfillment Summary"](#).



### Viewing and Confirming Payment Details

The Payment Confirmation screen displays the summary of the payments that have to be made for an order. For more information about adding or modifying the payment methods in an order, see [Section 4.9.3, "Change Payment Method"](#).

Sterling Call Center and Sterling Store can be used to carry out critical payment-related processes during order management processing. It also enables you to integrate with external payment processing systems such as CyberSource® or Chase™ Paymentech. For more information about integrating with external systems for payment handling, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

Sterling Call Center and Sterling Store provides the functionality to process real-time credit card authorizations when confirming draft orders. For more information about real-time credit card authorizations, see [Section 4.12, "Real-Time Credit Card Authorization"](#).

If an exchange order is opened from the sales order flow, the exchange order cannot be confirmed if the return order is a draft order. The `getCompleteOrderDetails` API is called to verify whether the return order associated with the exchange order has been confirmed. If the return order has not been confirmed, an appropriate message is displayed in the Confirm Payment screen.

### Entering a Name for a Draft Order

Sterling Call Center and Sterling Store enables a user to enter a name for a draft order during the process of creating an order. You can also configure the default draft order name that is displayed in the Confirm pop-up window that appears when a user attempts to close the Order Entry screen before completing the order creation process. For more information about configuring the default draft order name, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. (The `changeOrder` API is called to save the Draft Order Name on the order header.)



#### 4.6.1.2 End-User Impact

Depending on how you configure the order entry rules, the end users will see different behavior in the user interface:

- The user will see either the Customer Identification screen or the Add Item screen as the first screen in the order creation flow based on the Customer Screen Sequence and Line Entry Sequence rule.
- The user will see the Advanced Customer Search screen, the Address Search screen, or the Basic Customer Search screen based on the configuration.
- The user can search for customers based on the customer type if the configuration for Customer Type Search is enabled, and the default Customer Type during search is Business.
- The user can enter tax options for a business customer if the Change Tax Options for a business customer rule is enabled.
- The user will see either the Item Entry screen or the Large Order Item Entry screen based on the configuration.
- The user will see Shipping and Delivery as separate fulfillment methods based on the configuration.
- Based on the configuration, the user will see either the Basic Fulfillment Summary screen or the Advanced Fulfillment Summary screen. In the Advanced Fulfillment Summary screen, if a scheduling rule to display an additional shipping option is configured, the user will be able to see an additional option to schedule the order based on the scheduling rule configured.
- The user will see only the set of columns that you configure on the Add Line screen.
- The user will see the related items as a panel in the Add Line screen or as a pop-up window based on the configuration.
- The user will see item associations that are configured as Show in UI. The user will see the availability for the associated items based on the inventory check for associated items. Based on the configuration, when an associated item is added to an order, the associated item either replaces the original item or is added as a related item.
- When checking for availability, the user will see Country and States drop-down list based on the configuration.

#### 4.6.1.3 Implementation

This section explains the configurations for the Create Order task.

- Sterling Call Center and Sterling Store allows you to configure the order entry rules. For more information about configuring order entry rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. If the enterprise has been configured to not display unit prices in the UI, it is recommended that permissions to use the Update Order button be denied.
- Sterling Call Center and Sterling Store allows you to configure the display of model items. For more information about configuring the display of model items, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure external payment options. For more information about configuring the external payment options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- To add relationships between lines, you need to configure the association types to be displayed on the user interface. For more information about association types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. You should also configure a Relationship Type to be used with the Association Type. Ensure that the Consider for Sorting box is selected when defining line relationship types. For more information about configuring relationship types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the store pickup rules. For more information about configuring the store pickup rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure availability checks and reservation options. For more information configuring availability checks and reservation options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the order statuses for which addition of new order lines is allowed. Ensure that the status modification rules allow addition of lines to cancelled

order. For more information about configuring order statuses, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- Implement the YCDOVERRIDEDELIVERYMETHODUE user exit if the delivery method defined in the catalog has to be overridden.
- This task is permission-controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.
- To allow the user to view the updated price details and price breakdown on the Add Items screen before navigating to the next screen, the user must have resource permissions to see the Update Order button.
- Notes entered for this task are saved on the order with the YCD\_ORDER\_ENTRY note type.
- Enable notes for all status modifications of an order. This is how Sterling Call Center and Sterling Store records status changes. For more information about enabling notes, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.6.1.4 Reference Implementation

This section explains the reference implementation provided for the create order task.

- The Customer Identification step is configured to be the first step in the create order task.
- By default, the Advanced Customer Search screen is displayed for customer identification.
- By default, the customer type search is enabled, and the default customer type during search is Business.
- By default, users are not allowed to enter tax options for a business customer.
- By default, the Add Items screen is displayed for item entry.
- By default, Shipping and Delivery are displayed as separate fulfillment options, and the default fulfillment method is Delivery.
- By default, the Basic Fulfillment Summary screen is displayed.
- By default, the following columns are displayed in the line entry step:

- Unit of Measure
  - Line Quantity
  - Line Price
  - Line Totals
- By default, related items are displayed as a pop-up window.
- The Allow Country to be Entered for screens that check Item Availability rule is defaulted to "Y".
- As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to create an order to all user groups.
- As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to the hub users to access the Update Order button.

### 4.6.2 Capturing Orders through Other Channels

Sometimes an enterprise may have orders captured through other channels. If this order is to be processed in Sterling Call Center and Sterling Store, additional fields should be passed to the createOrder API.

#### 4.6.2.1 Solution

The information that can be passed on to Sterling Call Center and Sterling Store to process an order can be divided into the following groups:

- [Address Information](#)
- [Customer Information](#)
- [Item and Line Information](#)
- [Pricing and Payment Information](#)
- [Fulfillment Information](#)
- [Additional Order Information](#)

## Address Information

The following address information fields can be passed on to Sterling Call Center and Sterling Store:

**Table 4–4 Address Information fields**

Field	Description
Ship To Address	The address to which the order is being shipped.
Bill To Address	The address to which the bill for the order is sent.
Is Address Verified Flag	This flag indicates whether the address has been verified by an external address verification system. For more information about address verification, see <a href="#">Section 4.6.4.2, "Address Verification"</a> .
All Addresses Verified Flag	This flag indicates whether all addresses on the order have been verified by an external address verification system. For more information about address verification, see <a href="#">Section 4.6.4.2, "Address Verification"</a> .

## Customer Information

The following customer information fields can be passed at the <Order> level on to Sterling Call Center and Sterling Store by calling the createOrder API or changeOrder API:

**Table 4–5 Customer Information fields**

Field	Description
Customer ID	The identifier of the customer on the order. This field maps to the Bill To ID field in the database.
Customer Contact ID	The identifier of the customer contact on the order. When the order header contains a customer contact ID, any changes to contact information, such as payment methods and addresses, are saved to the applicable contact.
Special Care Flag	This flag indicates whether this is a special care customer.
IP Address	The IP address of the computer from which the order was placed. This field is referred to as SourceIPAddress in the input XML to the APIs.

**Table 4–5 Customer Information fields**

Field	Description
Customer Phone Number	The daytime phone number of the customer who placed the order. This attribute can be passed at the order header level, and if not, it is defaulted to the daytime phone number of the customer contact. If a customer contact daytime phone number is not available, it is defaulted to the daytime phone number of the bill-to address, if available. This field is referred to as CustomerPhoneNo in the input XML to the APIs.
Customer First Name	The first name of the customer who placed the order. This attribute can be passed at the order header level, and if not, it is defaulted to the first name of the customer contact. If the first name of a customer contact is not available, it is defaulted to the customer first name of the bill-to address, if available. This field is referred to as CustomerFirstName in the input XML to the APIs.
Customer Last Name	The last name of the customer who placed the order. This attribute can be passed at the order header level, and if not, it is defaulted to the last name of the customer contact. If the last name of a customer contact is not available, it is defaulted to the customer last name of the bill-to address, if available. This field is referred to as CustomerLastName in the input XML to the APIs.
Customer E-Mail ID	The e-mail ID of the customer who placed the order. This attribute can be passed at the order header level, and if not, it is defaulted to the e-mail ID of the customer contact. If the e-mail ID of a customer contact is not available, it is defaulted to the e-mail ID of the bill-to address, if available. This field is referred to as CustomerEMailID in the input XML to the APIs.

## Item and Line Information

The following item and line information fields can be passed on to Sterling Call Center and Sterling Store:

**Table 4–6 Item and Line Information fields**

Field	Description
Item ID	The identifier of the item on an order line.
Quantity	The quantity of the item ordered on an order line.
Item Promised Date	The shipping date displayed by the front-end system. Typically, retailer web sites display next to an item 'usually ships within 24 hours', or 'usually ships within 1-3 business days'. That projected date is the Item Promised Date.
Line Dependencies and Relationships	The relationships, if any, that exist between certain order lines. For example, an order may require all of its order lines to be shipped together. That relationship would be defined here.

## Pricing and Payment Information

The following pricing and payment information fields can be passed on to Sterling Call Center and Sterling Store:

**Table 4–7 Pricing and Payment Information fields**

Field	Description
Payment Method	The payment method used on the order.
Price	The price for an item.
Charges	The amount of charges applied to the order.
Taxes	The tax amount applied to the order.
List of Promotions Applied to the Order	The list of possible promotions that can be used on the order. For example, free shipping on orders above \$20, or a 10% discount on a specific category of goods.

## Fulfillment Information

The following fulfillment information fields can be passed on to Sterling Call Center and Sterling Store:

**Table 4–8 Fulfillment Information fields**

Field	Description
Carrier Service Level	The type of service used to ship the order. For example, a customer could want a next day air delivery, or a standard delivery that would take 5 business days, depending on how much they are willing to spend. This attribute helps Sterling Call Center and Sterling Store select a carrier for the order.
Fulfillment Option	The fulfillment mode of the order: <ul style="list-style-type: none"> <li>• Ship To Home</li> <li>• Delivery</li> <li>• Store Pick-up</li> </ul>

## Additional Order Information

The following additional order information fields can be passed on to Sterling Call Center and Sterling Store:

**Table 4–9 Additional Order Information fields**

Field	Description
Order Date	The date on which the order was initially placed.
Channel	The channel through which the order was entered. For example, this could be a web portal, a mail order, or a phone call.
Enterprise Code	The enterprise code on the order.

### 4.6.2.2 End-User Impact

None.

### 4.6.2.3 Implementation

None.



#### 4.6.2.4 Reference Implementation

None.

### 4.6.3 Order Delay

A customer may want to modify an order after it has been captured, for example to add quantity to a line, or change a ship-to address. Once an order is scheduled, most of its attributes can no longer be changed. Therefore, it is necessary to delay the scheduling of an order for a certain amount of time in case a customer wants to make a change to their order.

#### 4.6.3.1 Solution

To solve this business requirement, an order should be prevented from being scheduled until a certain number of hours has elapsed after order capture. In that time period, the order can be modified. Therefore, all orders should be delayed at order capture time to allow for these last minute modifications.

The length of the delay depends on your specific business practices.

#### 4.6.3.2 End-User Impact

An order is modifiable by a customer service representative for a certain amount of time, during which the order is not picked up by the scheduling agent.

#### 4.6.3.3 Implementation

When calling the createOrder API to initially capture an order from a front-end system into Sterling Call Center and Sterling Store, set the `EarliestScheduleDate` attribute in the input XML to a date that corresponds to the current time + the number of hours that the order should be delayed for.

For more information about the createOrder API, refer to the *Selling and Fulfillment Foundation: Javadocs*.

#### 4.6.3.4 Reference Implementation

None.

### 4.6.4 Order Validation

When an order is captured by Sterling Call Center and Sterling Store, it needs to be verified as a valid order that can be paid for, sent to the correct address, and is the order that the customer meant to enter in the front-end system.

The order validation process includes:

- [Duplicate Order Validation](#)
- [Address Verification](#)
- [Fraud Check](#)
- [Item Validation](#)
- [Credit Limit Exceeded](#)

#### 4.6.4.1 Duplicate Order Validation

When placing an order on the front-end system, a customer may inadvertently click the Submit button twice, thereby creating two duplicate orders. To avoid this scenario, every order that is captured must be checked against orders with similar attributes.

#### 4.6.4.1.1 Solution

When a draft order is confirmed, or an order is created, the YCD\_DUPLICATE\_ORDER hold is applied to the order. The hold prevents any transaction from processing the order, and is processed by the Duplicate Order agent, which is derived from the Process Order Hold Type base transaction.

Details for the Duplicate Order agent are described below:

#### Attributes

**Table 4–10 Duplicate Order Agent Attributes**

Attribute	Value
Base Transaction ID	PROCESS_ORDER_HOLD_TYPE
Base Document Type	0001 (Sales Order)
Base Process Type	ORDER_FULFILLMENT (Order Fulfillment)
Abstract Transaction	Yes
APIs Called	checkDuplicateOrder

#### Criteria Parameters

**Table 4–11 Duplicate Order Agent Criteria Parameters**

Parameter	Description
Action	This field is used internally by Sterling Call Center and Sterling Store. The only valid value is Get. Please do not modify this field.
Number of Records to Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

**Table 4–11 Duplicate Order Agent Criteria Parameters**

Parameter	Description
CollectPendingJobs	<p>If this parameter is set to “N”, the agent does not collect information on the pending jobs for this time-triggered transaction. This pending job information is used to monitor the agent in the System Management Console.</p> <p>By default, CollectPendingJobs is set to Y. It can be helpful to set it to N if one particular time-triggered transaction is performing a significant amount of getPendingJobs queries, and the overhead cost is too high.</p>
Next Task Queue Interval	The number of hours to wait before attempting to check for duplicate orders again. If not set, the default value is 5 hours from the current time.

**Statistics Tracked**

None.

**Pending Job Count**

The number of orders that need to be checked for duplicates.

**Events Raised**

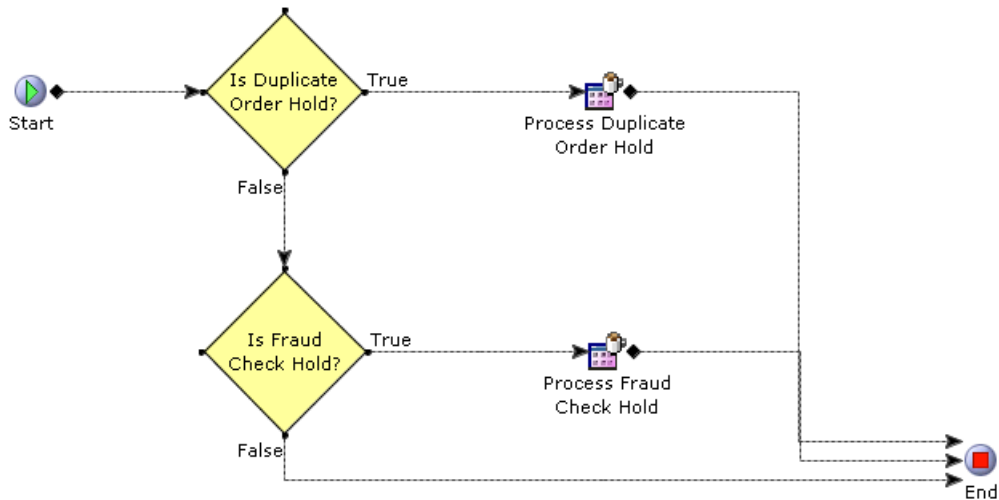
The following events are raised by this time-triggered transaction:

**Table 4–12 Events Raised by the Duplicate Order Agent**

Transaction/Event	Key Data	Data Published	Template Support?
ON_DUPLICATE_ORDER	OrderHeaderKey, OrderNo	VERIFY_ADDRESS.0001.VERIFY_ADDRESS_FAILED.0001.xml	Yes
ON_HOLD_TYPE_STATUS_CHANGE	modifyOrder_dbd.txt	PROCESS_ORDER_HOLD_TYPE.ON_HOLD_TYPE_STATUS_CHANGE.xml	Yes
ON_SUCCESS	modifyOrder_dbd.txt	PROCESS_ORDER_HOLD_TYPE.ON_SUCCESS.xml	Yes

The Duplicate Order agent calls the YFSProcessOrderHoldTypeUE user exit, which is implemented as the YCD\_ProcessCustOrdMgmtHolds\_1.0 service as illustrated in [Figure 4–1](#).

**Figure 4–1 The YCD\_ProcessCustOrdMgmtHolds\_1.0 Service**

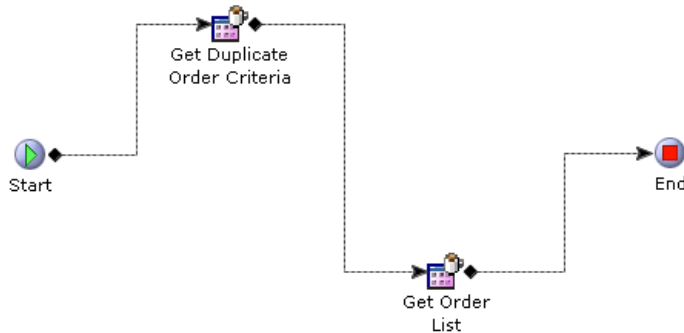


In this case, the processDuplicateOrderHold custom API is called, which in turn calls the checkDuplicateOrder API, which finally calls the YCDProcessDuplicateOrderCheckUE user exit.

The user exit takes the following file as its input XML:

```
<INSTALL_DIR>/repository/xapi/template/merged/userexit/process
DuplicateOrderCheck.0001.xml
```

The XML is then passed on to the YCD\_ProcessDuplicateOrderCheck\_1.0 service as illustrated in [Figure 4–2](#).

*Figure 4–2 The YCD\_ProcessDuplicateOrderCheck\_1.0 Service*

In this service, the `getOrderList` API is called using the output of the `getDuplicateOrderCriteria` custom API as input. If more than one record is returned by the `getOrderList` API call, then the order is a potential duplicate.

By default, the criteria for a duplicate order are the following:

- Bill To ID: if the identifier of the customer on the orders is identical.
- Total Amount: if the total amount on the orders are within \$5 of each other.
- Order Date: if the timestamp on the orders are within 20 seconds of each other.
- Enterprise Code: if the enterprise code on the orders is identical.
- IP Address: if the IP address of the customer on the orders is identical.

In that case, the `ON_DUPLICATE_ORDER` event is raised by the Duplicate Order agent, which invokes the `YCD_DuplicateOrderAlert_1.0` service through the `YCDOnDuplicateOrder` action. The `YCD_DuplicateOrderAlert_1.0` service sends an alert to the Duplicate Order queue, which is monitored by default by the user.

#### 4.6.4.1.2 End-User Impact

The user is notified through an alert of possible duplicate orders. They must either cancel the order if it is a duplicate, or remove the Duplicate Order hold if it is a valid order.

#### 4.6.4.1.3 Implementation

To use this feature, you should ensure that the YCD\_DUPLICATE\_ORDER hold type is implemented for the enterprise that performs your duplicate order checks. Because the hold type provided by Sterling Call Center and Sterling Store is for the DEFAULT Enterprise, your Enterprise should either inherit its configuration from DEFAULT, or have a copy of this hold type implemented.

In the YCD\_DuplicateOrderAlert\_1.0 service, the alert sent to the Duplicate Order queue uses the following XSL by default:

```
<INSTALL_DIR>/repository/xapi/template/merged/exception_console/YCD_DuplicateOrderAlert.xsl.sample
```

This XSL should be copied to a file named YCD\_DuplicateOrderAlert.xsl, customized as needed, and its path and name should be specified in the Template Name field of the Alert properties.

You can save it to another directory, but using the standard directory structure supplied by the Selling and Fulfillment Foundation helps ensure consistency.

You can customize all services and user exits to adapt to changing business needs. You can also customize the getDuplicateOrderCriteria custom API to return the exact set of criteria that you use to determine whether an order is duplicate.

#### 4.6.4.1.4 Reference Implementation

None.

#### 4.6.4.2 Address Verification

When a customer defines or changes an address, it needs to be verified as being a valid address. You may want to verify addresses on the order for any of the following reasons:

- To ensure that the order is shipped to an address that actually exists and that the user has entered it correctly
- To ensure that the ship-to address is not a known fraudulent address
- To ensure that the address is stored in Sterling Call Center and Sterling Store as a standardized address that all carrier services can understand and process

Sterling Call Center and Sterling Store has an understanding of when such validations should occur.

### 4.6.4.2.1 Solution

Sterling Call Center and Sterling Store provides a user interface to capture an address for the order.

There may be various scenarios like order capture or change of order address which mandate address verification. Sterling Call Center and Sterling Store provides an address verification agent to verify addresses on the order.

### Address Capture

Sterling Call Center and Sterling Store provides a user interface to capture an address pertaining to an order. This screen can be launched as a Shared Task. For more information about shared tasks, see [Section 3.4, "Shared Tasks"](#). When an address is either entered or modified, the `verifyAddress` API is called to verify the address.

A user can derive the City and State names using the Country and Postal Code fields, if the corresponding enterprise has defined a region schema in the Applications Manager. For more information about defining a region schema, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

The `getRegionList` API is called to retrieve the City and State names. The region schema that is called corresponds to the enterprise to which the user belongs. If multiple values are retrieved for the City and State names, the first value returned by the `getRegionList` API is displayed. For more information about deriving the City and State names using the Country and Postal Code fields, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

Users have the option of overriding the validation if they feel that the address that is provided is correct despite an address verification failure. If address validation fails and the Address Verification System (AVS) returns multiple addresses, the user can select an address from the list, or select none and override the validation.

### UI Verification of Phone Number and E-Mail Address

Sterling Call Center and Sterling Store validates e-mail address and phone numbers on the address capture screens. For more information



about validating the phone number and e-mail address, see [Section 3.5.3, "User Interface Validation of Phone Number and E-Mail Address"](#).

## Address Verification

When the order is initially captured by Sterling Call Center and Sterling Store using the createOrder API, the customer's ship-to and bill-to addresses should be validated. Address validation, however, could have already been completed by the front-end system for some or all of the addresses on the order. For example, the bill-to address is often validated by the credit card authorization process. The ship-to address could also have been validated up front.

Sterling Call Center and Sterling Store keeps track of which addresses have already been verified. Each address record maintains the `IsAddressVerified` flag with the value Y if the address has been recognized as being valid. If all the addresses on the order are valid, the `AllAddressesVerified` flag is set to Y, at the order header level.

During order capture, if the `AllAddressesVerified` flag is passed as "Y" in the input XML of the createOrder API, Sterling Call Center and Sterling Store does not attempt to verify any addresses on the order unless one of them is changed.

If the `AllAddressesVerified` flag is passed as N in the input XML of the createOrder API, Sterling Call Center and Sterling Store places the order on the address verification hold. The hold type applied to orders that have failed address verification is `YCD_VERIFY_ADDRESS`. By default, this hold type is defined for the DEFAULT Enterprise. If your implementation defines a different Enterprise to perform the address verification, you should either ensure that your configuration is inherited from the DEFAULT Enterprise, or that you create a copy of the `YCD_VERIFY_ADDRESS` hold type for your Enterprise. Orders on the address verification hold cannot be scheduled or released.

The Address Verification agent processes the address verification hold by trying to validate all addresses with `IsAddressVerified` set to N.

End users are notified of the addresses that have failed verification through alerts. When all the addresses on an order are verified, the associated alerts are closed automatically.

### Address Verification User Exit

The user exit allows Sterling Call Center and Sterling Store to interact with the AVS and modify the order for which the AVS was called appropriately.

The YCDVerifyAddressWithAVSUE user exit is called from the verifyAddress API. The user exit returns an XML output, which is same as the verifyAddress API output. For more information about the YCDVerifyAddressWithAVSUE user exit, refer to the *Selling and Fulfillment Foundation: Javadocs*.

The value of the AVSReturnCode attribute returned by the user exit can have the following values:

- VERIFIED if the address is valid.
- FAILED if the address is not valid.
- AVS\_DOWN if the AVS system could not be connected with.

### Address Verification Agent

The Address Verification agent picks up orders that are placed on the address verification hold. For each address in an order that needs to be validated, the agent calls the verifyAddress API, which in turn calls the YCDVerifyAddressWithAVSUE user exit to interact with the AVS.

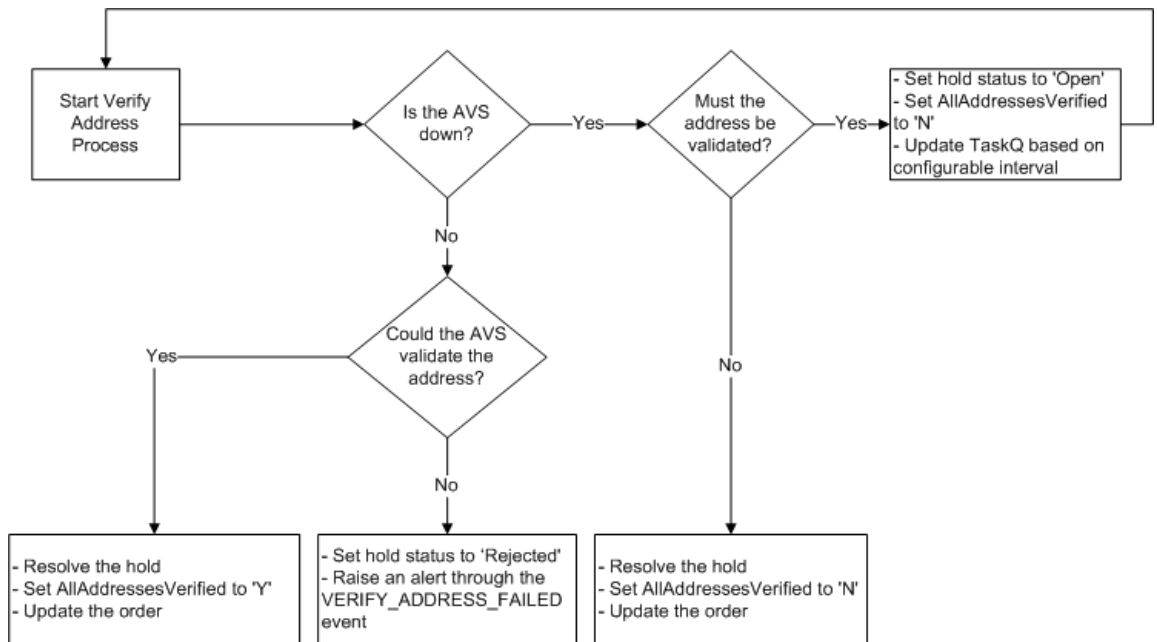
If the AVS confirms that all addresses are valid, the agent removes the hold on the order. If the address is invalid, the agent rejects the hold.

When an address is not valid, the AVS returns multiple possible close matches if it can. The agent handles this situation in the same way as if the AVS had indicated an invalid address: it rejects the hold on the order and raises an alert. The agent does not reprocess the order unless an address is changed manually.

In the event that the AVS is not online, the agent can be configured to either remove the hold from the order and process it, or reject it, to be reprocessed by the agent at a later time.

The time interval between the agent rejecting a hold and reprocessing that hold is configurable.

The flow chart in [Figure 4–3](#) illustrates the behavior of the Address Verification agent:

**Figure 4–3 The Address Verification Agent**

The details for the address verification agent are as follows:

### Attributes

**Table 4–13 Address Verification Agent Attributes**

Attribute	Value
Base Transaction ID	PROCESS_ORDER_HOLD_TYPE
Base Document Type	0001 (Sales Order)
Base Process Type	ORDER_FULFILLMENT (Order Fulfillment)
Abstract Transaction	No
APIs Called	verifyAddress

## Criteria Parameters

**Table 4–14 Address Verification Agent Criteria Parameters**

Parameter	Description
Action	This field is used internally by Sterling Call Center and Sterling Store. The only valid value is Get. Please do not modify this field.
Number of Records to Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
CollectPendingJobs	<p>If this parameter is set to N, the agent does not collect information on the pending jobs for this time-triggered transaction. This pending job information is used to monitor the agent in the System Management Console.</p> <p>By default, CollectPendingJobs is set to Y. It can be helpful to set it to N if one particular time-triggered transaction is performing a significant amount of getPendingJobs queries, and the overhead cost is too high.</p>
Next Task Queue Interval	The number of hours to wait before attempting to process an address again. This value is used in conjunction with the MUST_VERIFY_ADDRESS rule. If not set, the default value is 5 hours from the current time.

## Statistics Tracked

None.

## Pending Job Count

The number of orders that need to have addresses verified.

## Events Raised

The following events are raised by this time-triggered transaction:

**Table 4–15 Events Raised by the Address Verification Agent**

Transaction/Event	Key Data	Data Published	Template Support?
VERIFY_ADDRESS_FAILED	OrderHeaderKey, OrderNo	VERIFY_ADDRESS.0001 .VERIFY_ADDRESS_FAILED.001.xml	Yes

### 4.6.4.2.2 End-User Impact

The user will be notified using alerts if the address verification agent is configured to raise alerts.

### 4.6.4.2.3 Implementation

To use the address verification feature, the following steps need to be completed:

- Implement the YCDVerifyAddressWithAVSUE user exit to verify the address with AVS.
- Configure the address verification agent:

To configure the behavior of the agent when the AVS is not online, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

**Warning:** The Sterling Call Center and Sterling Store Rules home page is not permission controlled. Be sure to disable access to this page when Sterling Call Center and Sterling Store is deployed into a production environment.

To configure the amount of time that the agent should wait before attempting to reprocess an address that could not be verified can be configured in the Applications Manager. For more information about configuring agent criteria parameters, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

- If your configuration does not inherit from the DEFAULT Enterprise, create a copy of the YCD\_VERIFY\_ADDRESS hold type for your Enterprise.
- To validate the address and phone numbers entered on the address capture user interface, extend the YRCDataFormatter extension point.

### 4.6.4.2.4 Reference Implementation

Sterling Call Center and Sterling Store provides reference implementation for the YCDVerifyAddressWithAVSUE user exit. If you pass the value "AMES POND DR" to the PersonInfo/@AddressLine1 attribute, the user exit returns a set of three matches for the address.

### 4.6.4.3 Fraud Check

Your business may maintain a list of known fraudulent IP addresses, or a list of customers that have placed fraudulent orders in the past. Orders should be validated by specific fraud checks at order capture time, and fraud analysts should be notified of the failures.

#### 4.6.4.3.1 Solution

The fraud check mechanisms are very specific to business needs and practices. Because of this, Sterling Call Center and Sterling Store provides a general framework for fraud checks.

All orders are placed on the YCD\_FRAUD\_CHECK hold upon order creation or draft order confirmation. This hold prevents any transaction from processing the order, and is processed by the Fraud Check agent once the Duplicate Order hold has been resolved. For more information about duplicating an order validation, see [Section 4.6.4.1, "Duplicate Order Validation"](#). The Fraud check agent is derived from the Process Order Hold Type base transaction.

The details of the Fraud Check agent are as follows:

### Attributes

**Table 4–16 Fraud Check Agent Attributes**

Attribute	Value
Base Transaction ID	PROCESS_ORDER_HOLD_TYPE
Base Document Type	0001 (Sales Order)
Base Process Type	ORDER_FULFILLMENT (Order Fulfillment)
Abstract Transaction	Yes
APIs Called	checkFraudOnOrder

### Criteria Parameters

**Table 4–17 Fraud Check Agent Criteria Parameters**

Parameter	Description
Action	This field is used internally by Sterling Call Center and Sterling Store. The only valid value is Get. Do not modify this field.
Number of Records to Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
CollectPendingJobs	<p>If this parameter is set to <i>N</i>, the agent will not collect information on the pending jobs for this time-triggered transaction. This pending job information is used for monitoring the agent in the System Management Console.</p> <p>By default, <i>CollectPendingJobs</i> is set to <i>Y</i>. It can be helpful to set it to <i>N</i> if one particular time-triggered transaction is performing a significant amount of <i>getPendingJobs</i> queries, and the overhead cost is too high.</p>
Next Task Queue Interval	The number of hours to wait before attempting to process an order for fraud check. If not set, the default value is 5 hours from the current time.

Statistics Tracked

None.

Pending Job Count

The number of orders that need to be checked for fraud.

Events Raised

The following events are raised by this time-triggered transaction:

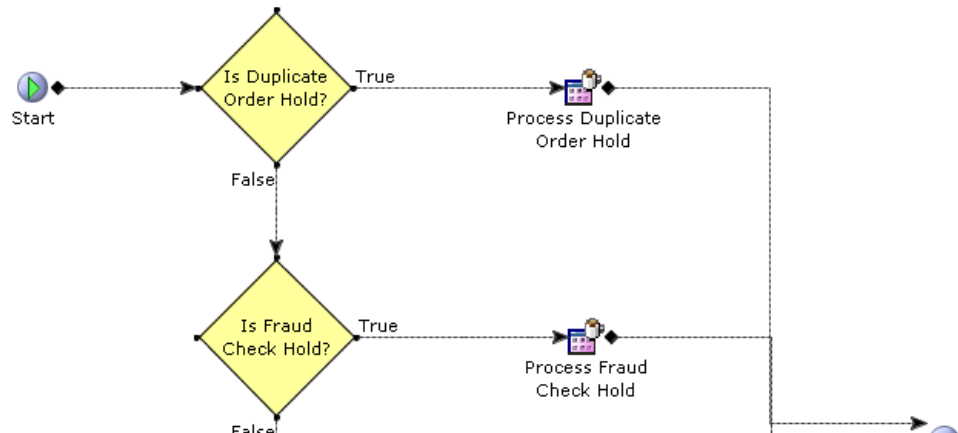
Table 4–18 Events Raised by the Fraud Check Agent

Transaction/Event	Key Data	Data Published	Template Support?
ON_FRAUDULENT_ORDER	OrderHeaderKey, OrderNo	FRAUD_CHECK.0001.ON_FRAUDULENT_ORDER.0001.xml	Yes
ON_HOLD_TYPE_STATUS_CHANGE	modifyOrder_dbd.txt	PROCESS_ORDER_HOLD_TYPE.ON_HOLD_TYPE_STATUS_CHANGE.xml	Yes
ON_SUCCESS	modifyOrder_dbd.txt	PROCESS_ORDER_HOLD_TYPE.ON_SUCCESS.xml	Yes

The Fraud Check agent calls the YFSProcessOrderHoldTypeUE user exit, which is implemented as the YCD\_ProcessCustOrdMgmtHolds\_1.0 service as illustrated in [Figure 4–4](#).



**Figure 4–4 The YCD\_ProcessCustOrdMgmtHolds\_1.0 Service**



In this case, the processFraudCheckHold custom API is called, which in turn calls the checkFraudOnOrder API, which finally calls the YCDProcessFraudCheckOnOrderUE user exit.

The user exit takes the following file as its input XML:

```
<INSTALL_DIR>/repository/xapi/template/merged/userexit/process
OrderFraudCheck.0001.xml
```

If the user exit returns a message to the agent that indicates possible fraud, the ON\_FRAUDULENT\_ORDER event is raised, and the YCDFraudCheckAlert\_1.0 service is invoked through the YCDOnFraudulentOrder action.

The YCD\_FraudCheckAlert\_1.0 service sends an alert to the Fraud Queue queue, which is monitored by the Fraud Analyst user.

If the user exit returns a message indicating that the order is not fraudulent, the agent will automatically resolve the hold. The processFraudCheckHold API, on the other hand, will not.

#### 4.6.4.3.2 End-User Impact

When an order fails the fraud check, an alert is sent to the queue monitored by the Fraud Analysts. After analyzing the order, the Fraud Analyst either cancels it if it turns out to be fraudulent, or removes the hold so that it can be processed normally.

#### 4.6.4.3.3 Implementation

To use this feature, you should ensure that the YCD\_FRAUD\_CHECK hold type is implemented for the Enterprise that performs the fraud checks. Since the hold type provided by Sterling Call Center and Sterling Store is for the DEFAULT Enterprise, your Enterprise should either inherit its configuration from the DEFAULT Enterprise, or have a copy of this hold type implemented.

In the YCD\_FraudCheckAlert\_1.0 service, the alert sent to the Fraud Check queue uses the following XSL by default:

```
<INSTALL_DIR>/repository/xapi/template/merged/exception_console/YCD_FraudCheckAlert.xsl.sample
```

This XSL should be copied to a file named YCD\_FraudCheckAlert.xsl, customized as needed, and its path and name should be specified in the Template Name field of the Alert properties.

You can save it to another directory, but using the standard directory structure supplied by the Selling and Fulfillment Foundation helps ensure consistency.

All of these services can be customized as needed to fit your business needs, as well as the user exits.

The YCDProcessFraudCheckOnOrderUE user exit should return the response code of the fraud check in the FraudCheckResponseCode attribute, at the Order element level. This attribute value should be 'FAILED' if the fraud check failed, and 'SUCCESS' if the fraud check succeeded. Additionally, a FraudCheckResponseMessage element should be returned under the Order element. For more information about the YCDProcessFraudCheckOnOrderUE user exit, refer to the *Sterling Call Center and Sterling Store: Javadocs*.

#### 4.6.4.3.4 Reference Implementation

None.

#### 4.6.4.4 Item Validation

After items are added to an order, validations must be performed to verify whether the items or the quantity of the items in the order are valid. For example, an enterprise may want to sell some items only in

bulk quantities. In such a situation, the enterprise may want to set a minimum order quantity for those items.

#### 4.6.4.4.1 Solution

In the Add Items screen, Sterling Call Center and Sterling Store validates whether an item or the quantity of an item is valid for an order. Validations are also performed in the Customer Identification screen to verify whether a customer is entitled to an item. The validations are performed only if the following rules are configured in the Applications Manager:

- Enable Extended Item Validation

For more information about defining installation rules, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

- Validate Item

For more information about configuring a document's order validation, refer to the *Sterling Distributed Order Management: Configuration Guide*.

- Validate Item During Order Creation/Modification

For more information about defining a process type's primary information, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

The items in the order are validated based on the following criteria:

- Minimum Order Quantity and Maximum Order Quantity—The quantity ordered for the item must not be less than the minimum order quantity and not more than the maximum order quantity that is configured for the item.

---

**Note:** A user can override the quantity validation for an item. If the user overrides the quantity validation for an item, the quantity validation is overridden for all the order lines containing the item, and a note will be automatically generated on the order using the YCD\_QUANTITY\_VALIDATION\_OVERRIDDEN note type.

---

- Item Cannot be Sold Separately—The item must be configured in such a way that it can be sold separately.
- Item Entitlement—The customer to whom the order belongs must have entitlements to the item.
- Item Status—The status of the item must be "Published".
- Item's Effective Date—The effective end date of the item must not be earlier than the current date.

The `validateItemForOrdering` API is called to validate the items as follows:

- When the Add Items screen is launched if the order is an existing draft order.
- When the user clicks the Next button in the Add Items screen.
- When the User clicks the Update Order button in the Add Items screen.
- When the user clicks the Next button in the Customer Identification screen after adding the items to the order.

If validation fails for any of the items, an error message is displayed to notify the user of the invalid item in the order. If the validation fails because the item is not valid and the minimum or maximum quantity criteria is not met, the error message pertaining to the invalid item is displayed.

#### 4.6.4.4.2 End-User Impact

Validations will be performed to verify whether the items and the quantity of items in an order are valid or not only if the item validation rules are configured in the Applications Manager.

#### 4.6.4.4.3 Implementation

To use this feature, the item validation rules must be configured in the Applications Manager.

Notes, which are automatically generated on the order when the quantity validation is overridden, are saved in the order using the YCD\_QUANTITY\_VALIDATION\_OVERRIDDEN note type.

#### 4.6.4.4.4 Reference Implementation

None.

#### 4.6.4.5 Credit Limit Exceeded

When a customer exceeds the credit limit on the customer account used to place an order, the order is placed on hold. The hold prevents the order from processing the following transactions:

- Accept Reservation
- Close Order
- Confirm Draft Order
- Release Order
- Schedule Order
- Payment Collection
- Payment Execution

Refer to the *Sterling Distributed Order Management: Configuration Guide* for more information about removing the hold.

#### 4.6.4.5.1 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides an event handler for customer modifications performed on account limit change, which is handled by CUSTOMER\_DEFINITION.ON\_ACCOUNT\_LIMIT\_CHANGE. When the account limit is changed, all orders that use the account number are updated, and the system automatically removes the hold from the order if the credit limit has been increased sufficiently to confirm the order.

As part of reference implementation, Sterling Call Center and Sterling Store provides a hold type that implements the

USE\_HOLD\_TYPE\_CUSTOMER\_ACCOUNT rule provided by Selling and Fulfillment Foundation, which automatically places the order on hold if the customer exceeds his customer account limit.

## 4.7 Order Inquiry

Customers often enquire about information on orders they have placed. In such situations, Sterling Call Center and Sterling Store needs to respond to order inquiries by publishing information about the orders captured internally to the external systems.

### 4.7.1 Solution

Sterling Call Center and Sterling Store provides user interface tasks which enable users to search for orders. It also enables users to view the summary of an order. Additionally, users can view the summary of individual order lines.

#### Order Search

This section explains the Order Search task and the solution offered by Sterling Call Center and Sterling Store.

Sterling Call Center and Sterling Store provides two ways to search for orders:

- Quick Access—The user can enter the order number, customer's phone number or customer's e-mail ID and search for the order.
- Advanced Order Search—The user can enter more restrictive search criteria to search for the order.

If a user has access to multiple enterprises, the Enterprise drop-down list is displayed in the Order Search Criteria panel of the Advanced Order Search screen. The `getOrganizationList` API is called to retrieve the list of enterprises. If the search criteria results in a single record, the Order Summary screen is displayed. If the search criteria results in multiple records being retrieved, the Search and List screen is displayed.

The search results can be sorted in ascending or descending order based on following Order By options:

- Order Number
- Order Date

- Customer Last Name

After an order has been captured, the detailed information about the order, shipments, containers, and the return orders associated with that order can be retrieved.

Sterling Call Center and Sterling Store user interface (UI) allows users to search and view the orders that are purged into their corresponding history tables. When these purged orders are accessed through the UI, they are restored to the non-history tables by calling the `restoreOrder` API.

Sterling Call Center and Sterling Store provides the `getCompleteOrderDetails` API, which retrieves the list of orders that match the search criteria. For more information about the `getCompleteOrderDetails` API, refer to the *Sterling Call Center and Sterling Store: Javadocs*.

The `getCompleteOrderDetails` API can return the `OverallStatus` attribute for orders and order lines. This attribute's value is based on the `YFSGetOverallStatusUE` user exit. For more information about APIs and user exits, refer to the *Selling and Fulfillment Foundation: Javadocs*.

The order status generated by the Sterling Call Center and Sterling Store application may be different than the order status that you want to display to customers. For example, an order may be in the Scheduled status, but you may want to display the status as In Progress to customers who access the Sterling Multi-Channel Selling Solution to inquire about the status of their order. You can implement the `YFSGetOverallStatusUE` user exit to return the `OverallStatus` attribute, which is used to display the order status in the Order Summary screen and the Order Line Summary screen of Sterling Call Center and Sterling Store. For more information about the `YFSGetOverallStatusUE` user exit, refer to the corresponding section in the *Sterling Call Center and Sterling Store: Javadocs*.

## Order Summary

Sterling Call Center and Sterling Store enables users to view the summary of a particular order. Users can view order information such as the order number, date when the order was created, status of the order, total amount, shipping address, billing address, and individual order lines.

The `getCompleteOrderDetails` API is called to display the Order Summary screen. This API returns a `SupportLevel` attribute with the following values:

- **Minimum**—If this value is returned, the Order Summary screen is displayed with most of the tasks disabled. For more information about the Order Summary screen for incompatible orders, see [Order Summary Screen for Incompatible Orders](#).
- **Medium**—If this value is returned, the Order Summary screen will be displayed to the older clients, but with fewer order modification tasks.
- **Full**—If this value is returned, the Order Summary screen is displayed with all the order modification tasks.

Also, the `getExceptionListForOrder` API is called to retrieve the number of open alerts on the order.

If an order is being modified by a user through a different channel, for example, Sterling Web, and the order contains unsaved changes, when this order is opened simultaneously by a Sterling Call Center or Sterling Store user, a validation is performed when the order is retrieved using the `getCompleteOrderDetails` API in order to verify whether the order contains unsaved changes. If the order contains unsaved changes, the Sterling Call Center or Sterling Store user will be prompted to revert the unsaved changes made to the order or close the order without performing any modification. If the order does not contain any unsaved changes, the Order Summary screen is displayed, and the Sterling Call Center and Sterling Store user can make the necessary changes.

Similarly, if a Sterling Call Center or Sterling Store user is performing a modification to an order, and the order is simultaneously opened by a Sterling Web user for order modifications, when the `changeOrder` API is called, an error is thrown, indicating that there are unsaved changes in the order. A message is also displayed in the Sterling Call Center and Sterling Store applications indicating that the order contains changes made through a different channel. The Sterling Call Center or Sterling Store user is automatically taken to the Order Summary screen and is prompted with an option to either revert the unsaved changes, or close the Order Summary screen.

Sterling Call Center and Sterling Store assumes that the `Create Shipments For Products Being Delivered In Addition To Work Order` check box is selected in the Applications Manager, to view order lines that have



Delivery as a fulfillment method and are already delivered. This is defined as part of the process type's primary information. For more information about defining a process type's primary information, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

## Order Pricing Summary

Sterling Call Center and Sterling Store enables users to view the price breakdown of an order. In the Order Pricing Summary pop-up window, users can view the details of the prices on the order, such as the order line prices, the line-level price adjustments, the order total, and the order-level price adjustments. The `getCompleteOrderDetails` API is called to display the Order Pricing Summary pop-up window.

## Order Summary Screen for Incompatible Orders

An Order Summary screen for incompatible orders is displayed if the value of the `SupportLevel` attribute of the `getCompleteOrderDetails` API is `Minimum`. This screen enables the user to view the details of the order, but does not allow the user to perform any of the order modification tasks, except order cancellation. A message is displayed on the Order Summary screen indicating that the screen has been modified because it is not compatible with the client version being used.

The user is provided with an option to cancel the order. The `CANCEL_REASON` common code is used to display the list of cancellation reasons. The `changeOrder` API is called to cancel the order and a note is automatically added to the order along with the cancellation reason selected by the user.

## Line Summary

Users can view the summary of individual order lines. The Line Summary screen displays the primary details of the order line and the shipping address. If the user has included a message for a gift item, the gift message is displayed in the Primary Details panel of the Line Summary screen. The `getOrderLineDetails` API is called to display the Line Summary screen. Users can also view the pricing information and important events or notes specified for the order line. The `getOrderAuditDetails` API is called to display the important events specified for the order line.

If the Display of URLs for Items rule in the Channel Applications Manager is configured to display the URL of a product Web page pertaining to an item, a View Product Web Page hyperlink is displayed in the Primary Details panel of the Line Summary screen. The `getCompleteOrderDetails` API is called to retrieve the list of assets that are associated with the item. For more information about configuring the display of a product Web page for an item, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### Pagination

Sterling Call Center and Sterling Store supports smart retrieval of record sets in the Order Search screen. You can configure if history records need to be automatically fetched while retrieving records. For more information about configuring pagination rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

For more information about pagination, see [Section 4.22, "Pagination"](#).

## 4.7.2 End-User Impact

This section explains the end-user impact for the Order Inquiry task.

- Based on the configuration to search for history orders, the user will see history orders or recent orders in the search results.
- In Order Search and Return Order Search tasks, the user sees the Order By pull-down window, which enables a user to sort the data by Order Number, Order Date, or Customer Last Name.
- If an order is incompatible with a particular version of the application, an Order Summary screen for incompatible orders is displayed.
- If the `YFSGetOverallStatusUE` user exit is implemented, and if this user exit returns the `OverallStatus` attribute, the user will be able to view the Display Status field in the Order Summary screen and the Order Line Summary screen.
- The user will see the View Product Web Page hyperlink in the Line Summary screen based on the configuration.

## 4.7.3 Implementation

This section explains the configurations for the order inquiry task.

- Implement the `YFSGetOverallStatusUE` user exit to return the `OverallStatus` attribute for orders and order lines.
- Define the qualifiers for an order to indicate the compatibility of an order with a particular version of the application. For more information about defining the qualifiers, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
- Configure the list of cancellation reasons to be displayed while cancelling the order. For more information about configuring the cancellation reasons, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Configure the Sterling Call Center and Sterling Store application to display the URL of a product Web page pertaining to an item, in the Line Summary screen. For more information about configuring the display of a product Web page for an item, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission-controlled. Sterling Call Center and Sterling Store allows you to assign permissions to the corresponding user groups for this task.

#### 4.7.4 Reference Implementation

This section explains the reference implementation provided as part of the Order Inquiry task.

- By default, the following asset types are provided:
  - Audio
  - Video
  - Text
  - Image
- As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to search for orders to all users.

### 4.8 Order Monitoring

After the orders have been captured, validated, paid for, and their processing has begun, they need to be monitored to ensure that

customers are notified when the order is about to ship, or whether the promised delivery dates are going to be met.

Currently, Sterling Call Center and Sterling Store supports the following aspects of order monitoring:

- [Federal Trade Commission Compliance](#)
- [Shipment Notification](#)
- [Order Cancellation Notification](#)

### 4.8.1 Federal Trade Commission Compliance

The United States Federal Trade Commission (FTC) defines rules that retailers must follow when dealing with customer orders. One of these rules deals with mail or telephone order merchandise.

This rule states that:

- A notification must be sent to a customer if the promised ship date cannot be met and that a new promised ship date will be given to the customer.
- If the customer does not respond to the notification that is sent the first time the merchandise is delayed, the retailer can assume that the customer accepts the delay.
- If the merchandise is delayed more than once and the customer does not respond to these notifications, the retailer must cancel the order.
- If the retailer is unable to give a promise date and the customer can accept an indefinite delay, notifications do not have to be sent to the customer notifying them of delays.

In retail scenarios, ship dates should be calculated based on business advertising. For example, if a Web site states that items that are on-hand are shipped within ten days, the promised ship date should be ten days after the date of the order capture. If a time frame is not advertised by the retailer, the FTC stipulates that an order must be shipped within 30 days of the order capture.

If it becomes apparent that the ship date will not be met, a second ship date must be calculated. This date cannot be more than 30 days after the first ship date. A notification should be sent to the customer, informing them about the shipment failure, along with the next ship

date. If, for any reason, the notice cannot be received by the customer, the order is expected to be cancelled by the end of the day corresponding to the first ship date.

A second ship date has been calculated, and if, before that date, it becomes apparent that the order will not be completely delivered on time, a notification should be sent to the customer, stating that the ship date is unknown, and that the order should be cancelled on the second ship date.

---



---

**Note:** When a customer has an e-mail set on the order, notification e-mails will be sent directly to them. In all cases, an alert will be raised. This alert can be resolved when a response is received from the customer or automatically, if the alert is raised for the first delay.

---



---

#### 4.8.1.1 Solution

Sterling Call Center and Sterling Store monitors orders from end-to-end to assist you in abiding by this FTC regulation. To do this, five dates are used to determine whether a delay occurred, if yes, the type of delay, whether a notification is required, and finally, notify the customer accordingly. The dates are:

- Expected ship date
- First promised date
- Current promised date
- Next promised date
- Cancel date

#### Calculation of Dates

##### First Promised Date

When an order is created, the first promised date should be set by the application creating the order. This can be done by passing the OrderDate element of the order line, with the DateTypeId as YCD\_FTC\_FIRST\_PROMISE\_DATE, and the ActualDate, that is, the date the merchandise will ship on. If the first promised date is not passed, Sterling Call Center and Sterling Store will default this date to OrderDate + a fixed number of days that can be configured. When orders are

created within Sterling Call Center and Sterling Store, the first promised date is set to the reservation date (see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide* for information about configuring FTC Compliance). If reservations are not turned on in Sterling Call Center and Sterling Store, the same logic that is used to determine the reservation date will be used to determine the first promised date. This date will remain untouched unless it is reset through the UI. If the order is changed in a way that results in the ship date being changed, this date can be updated to the new first promised date. This will be determined by the application making the change. When the first promised date is passed in the input to the createOrder API or changeOrder API, all the dates will be reset to reflect the new promised date:

```
<OrderDates>
  <OrderDate ActualDate="20050807"
    DateTypeId="YCD_FIRST_PROMISED_DATE"/>
</OrderDates>
```

**Note:** If a line has multiple expected ship dates, the highest ship date will be used. This means that the customer is being told that the entire quantity on this line will be shipped by this date. Some quantity may be shipped earlier, but the shipment will be completed by this date.

### Current Promised Date

The current promised date will be set to the first promised date whenever the first promised date is passed as input to the createOrder API or changeOrder API. When a delay is accepted, the current promise date is set to the next promised date given to the customer.

---

---

**Note:** This is handled by one of the order monitors shipped with Sterling Call Center and Sterling Store and the Alert Details screens in Sterling Call Center and Sterling Store. If there are any custom screens that will resolve FTC notifications or accept delays, they will have to set the FTC promised date on the order line. To accomplish this, pass the OrderDate Element in the input to changeOrder API, with the DateTypeId attribute set to YCD\_FTC\_PROMISED\_DATE.

---

---

### Next Promised Date

The next promised date is always handled through back-end logic and should never be set by the UI. When the order monitor runs, if a delay is found, the next promised date will be set to the maximum ship date for the corresponding line. In case of an indefinite delay, the next promised date will be set to a high date.

### Cancel Date

The cancel date is always handled through back-end logic and should never be set by the UI. When the order monitor runs, if it is determined that the order has to be cancelled because action is not taken by the customer, the cancel date will be set. If the delay is greater than 30 days or if it is an indefinite delay, Sterling Call Center and Sterling Store will set the cancel date to the current promise date + the fixed number of days. This value can be configured in the Configuring FTC Compliance screen in the Channel Applications Manager (see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide* for information about configuring FTC Compliance). In cases where the delay is less than 30 days, and at least one delay has already been sent to the customer, the cancel date will be set to the current promise date + the process buffer.

## Notifications

### Backorder/Indefinite Notifications

When a quantity on an order line is backordered, an indefinite notification is sent to the customer, informing them of the backorder. Because the inventory picture changes often, no notification will be sent to the customer for more than the configured time before the promised

date. This allows time for rescheduling so that the order will not be delayed.

If the order is backordered from node, a notification will not be sent. Instead, Sterling Call Center and Sterling Store will wait for the scheduler to run again in order to determine either another node or whether it needs to back order, in which case, Sterling Call Center and Sterling Store will send the indefinite notification to the customer.

The monitor rules used to catch these scenarios are:

IF Order

meets condition Indefinite FTC Notification Is Needed? and has not reached YCD unreachable Milestone 48 elapsed hours before Actual Current FTC Promise Date  
THEN

raise Create Indefinite FTC Delay Email.

IF Order

meets condition Indefinite FTC Notification Is Needed? and has not reached YCD unreachable Milestone 120 elapsed hours before Actual Current FTC Promise Date  
THEN

raise Create Indefinite FTC Delay Alert.

---

---

**Note:** In compliance with this FTC regulation, if the customer accepts an indefinite delay, they will not be notified again until the shipment is shipped.

---

---

### First Delay Notifications

When a quantity on an order line is delayed beyond the process buffer and within 30 days of the promised date, and where the promised date has not already been delayed, a first delay notification will be sent to the customer. The notification is sent a configured number of hours before the promised date. By default, this is 48 hours for orders with e-mail addresses and 120 hours for orders without e-mail addresses. This is achieved by defining the monitoring rules that raise events when a line does not have its entire quantity shipped or cancelled. The monitor rules that are used to handle these scenarios are described in the next paragraph. These events are consolidated at the order level. Therefore, the customer is notified only once for all the lines on the same order with the same promised date and same type of delay. Notes will also be added to each order line, stating that the customer has been notified by e-mail,



or that the customer should be notified by a CSR about the order not being shipped on time.

Each line, with the applicable quantity, is included in the notice to the customer, along with a second promised delivery date. The notice informs the customer that the order is not likely to be shipped by the first promised date. It will also state that if the customer does not reply, it will be assumed that the delay is acceptable and therefore approved.

IF Order

meets condition First FTC Notification Is Needed? and has not reached Order line completely shipped or cancelled 48 elapsed hours before Actual First FTC Promise Date

THEN

raise Send First FTC Delay Email.

IF Order

meets condition First FTC Notification Is Needed? and has not reached Order line completely shipped or cancelled 120 elapsed hours before Actual First FTC Promise Date

THEN

raise Create First FTC Delay Alert.

## Next Delay Notifications

As with the first FTC notification, the next delay notification is defaulted to send notifications if the lines have not been completely shipped or cancelled 48 hours before the promised date when the customer has an e-mail address, and 120 hours before the promised date if they do not. The next delay notifications are the same as the first delay notifications with the following exceptions:

- The notification states that if the customer does not explicitly request the order to remain open, it will be cancelled by the cancel date.
- The monitor rule will retrigger to catch the additional delays.

In the event that the customer requests to keep the order open and accepts the delay, and if there are further delays, the customer will continue to be notified of these delays.

The monitor rules that handle these scenarios are:

IF Order

meets condition Next FTC Notification Is Needed? and has not reached Order line completely shipped or cancelled 48 elapsed hours before Actual Current FTC Promise Date

```
THEN
    raise Send FTC Delay Email.

IF Order
    meets condition Next FTC Notification Is Needed? and has not reached Order
line completely shipped or cancelled 120 elapsed hours before Actual Current FTC
Promise Date
THEN
    raise Create FTC Delay Alert.
```

### Automatic Acceptance

When first delay notifications are sent, if the customer does not respond by the promised date, the delay is automatically accepted and the open alert is automatically closed. This is achieved by defining a monitor rule that runs on the promised date if the line has not been completely shipped or cancelled. These events will be consolidated at the order level. No notification will be sent to the customer. However, a note will be logged on the order line, stating that it was automatically accepted because the customer did not respond.

The monitor rule that is used to handle this scenario is:

```
IF Order
    has not reached Order line completely shipped or cancelled within 1 elapsed
hours of Actual First FTC Promise Date
THEN
    raise Auto Resolve FTC Alerts.
```

---

---

**Note:** If the promised date given to the customer is the same day the order is taken, and if, later in the day, it is found that it will not ship on that day, a notification will be sent and the delay will be automatically accepted at the same time. However, this complies with the FTC regulations, and if the customer does not agree to the delay, they can cancel their order any time.

---

---

### Auto Cancellation Notice

When a next delay notification or indefinite delay notification is sent, and the customer does not respond by the cancel date, the order lines will be cancelled and the corresponding alert will be closed. This is achieved by

defining a monitor rule that will run on the cancel date if the line has not been completely shipped or has been cancelled. These events will be consolidated at the order level. The customer will be sent a notice that the order lines were cancelled, and notes will be put on the order line, stating that the lines were cancelled because the customer did not respond.

The monitor rule that is used to handle this scenario is:

```
IF Order
  has not reached Order line completely shipped or cancelled within 1 elapsed
hours of Actual YCD FTC CCANCEL DATE
THEN
  raise FTC Cancel Lines.
```

#### 4.8.1.2 End-User Impact

Alerts will be raised whenever a customer needs to be notified, regardless of whether an e-mail was sent to the customer or not. If an e-mail was sent, the alert description will state that the e-mail was sent.

In the event that the customer has not provided an e-mail address, the user should contact the customer, either by phone or mail, and inform them of the problem with the order. The user should not resolve the alert until the customer has responded to the notification with either acceptance or cancellation.

#### 4.8.1.3 Implementation

The services called by Sterling Call Center and Sterling Store to send notifications to customers can be customized as needed.

To run the monitor for Federal Trade Commission compliance, use the order monitor template that is located in <INSTALL\_DIR>\template\monitor. If you have not previously extended this template, rename the sample file ORDER\_MONITOR\_EX.001.xml.sample as ORDER\_MONITOR\_EX.001.xml. If you have already extended this template, include the contents of the sample file in your extended template.

To use this feature, you must enable the Synchronize Dates Between Master Order Dates And Dates On Order Line And Schedules rule in the Applications Manager. For more information about defining transaction rules, refer to the *Sterling Distributed Order Management: Configuration*

*Guide.* During the fulfillment process, the order date is set to the maximum ship date. When the Order Monitor runs, the maximum ship date is compared with the promised date to determine if a delay has occurred. If a delay has occurred, the order date is updated based on the stipulations laid down by the Federal Trade Commission.

Notes entered for this task are saved in the order using the YCD\_FTC\_NOTIFICATION note type.

## Services

The following services are present in the Applications Manager in the Application Platform Module under the Sales Order document type:

- YCD\_FTC\_AUTO\_RESOLVE – This service is used to resolve the FTC notification alert, and update the dates on the order lines being delayed. It is invoked by the YCD\_FTC\_Auto\_Resolve monitor event. This service invokes the FTCHandler custom API with the arguments listed in [Table 4–19](#). For details about how this custom API can be customized, see the corresponding section. [Figure 4–5](#) illustrates this service.

**Table 4–19** YFC\_FTC\_AUTO\_RESOLVE arguments for FTCHandler

Fields	Description
FTCDelayType	AutoResolve
NoteText	YCD_FTC_AUTO_RESOLVE_NOTE_TEXT

- YCD\_FTC\_CANCEL\_ORDER\_LINES – This service is used to cancel the order lines if the customer has not responded to the notifications and resolved the FTC notification alert. It is invoked by the YCD\_FTC\_Cancel\_Lines monitor event. This service invokes the FTCHandler custom API with the arguments listed in [Table 4–20](#). For details about how this custom API can be customized see the corresponding section. The only difference between this service and the YCD\_FTC\_AUTO\_RESOLVE service are the arguments passed to the FTCHandler. [Figure 4–5](#) illustrates this service.

**Table 4–20 YCD\_FTC\_CANCEL\_ORDER\_LINES arguments for FTCHandler**

Fields	Description
FTCDelayType	CancelLines
NoteText	YCD_FTC_CANCEL_ORDER_LINES_NOTE_TEXT

**Figure 4–5 The YCD\_FTC\_AUTO\_RESOLVE and YCD\_FTC\_CANCEL\_ORDER\_LINES Services.**

- YCD\_FTC\_DELAY\_ALERT – This service is used to update the order dates and send a notification to the customer. It is invoked by the YCD\_FTC\_Delay\_Alert monitor event. This service invokes the FTCHandler custom API with the arguments listed in [Table 4–21](#). For details about how this custom API can be customized, see the corresponding section. [Figure 4–6](#) illustrates this service.

**Table 4–21 YCD\_FTC\_DELAY\_ALERT arguments for FTCHandler**

Fields	Description
FTCDelayType	NextFTCDelay
NoteText	FTC_Delay_Alert_Note_Text

- YCD\_FTC\_FIRST\_DELAY\_ALERT – This service is used to update the order dates and send a notification to the customer. It is invoked by the YCD\_FTC\_First\_Delay\_Alert monitor event. This service invokes the FTCHandler custom API with the arguments listed in [Table 4–22](#). For details about how this custom API can be customized, see the corresponding section. The only difference between this service and the YCD\_FTC\_DELAY\_ALERT service are the arguments passed to the FTCHandler custom API. [Figure 4–6](#) illustrates this service.

**Table 4–22 YCD\_FTC\_FIRST\_DELAY\_ALERT arguments for FTCHandler**

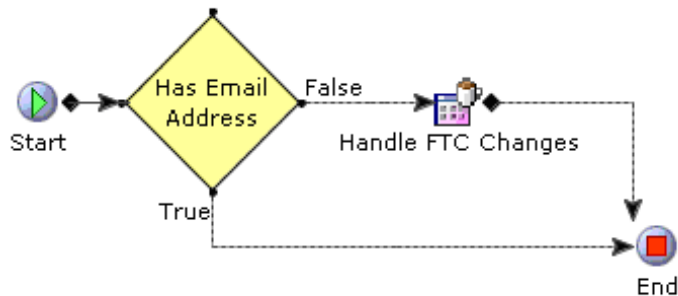
Fields	Description
FTCDelayType	FirstFTCDelay
NoteText	FTC_First_Delay_Alert_Note_Text

- **YCD\_FTC\_INDEFINITE\_DELAY\_ALERT** – This service is used to update the order dates and send a notification to the customer. It is invoked by the YCD\_FTC\_Indefinite\_Delay\_Alert monitor event. This service invokes the FTCHandler custom API with the arguments listed in [Table 4–23](#). For details about how this custom API can be customized, see the corresponding section. The only difference between this service and the YCD\_FTC\_DELAY\_ALERT service are the arguments passed to the FTCHandler custom API. [Figure 4–6](#) illustrates this service.

**Table 4–23** *YCD\_FTC\_INDEFINITE\_DELAY\_ALERT arguments for FTCHandler*

Fields	Description
FTCDelayType	IndefiniteFTCDelay
NoteText	FTC_Indefinite_Delay_Alert_Note_Text

**Figure 4–6** *YCD\_FTC\_DELAY\_ALERT, YCD\_FTC\_FIRST\_DELAY\_ALERT, and YCD\_FTC\_INDEFINITE\_DELAY\_ALERT Services.*



- **YCD\_FTC\_DELAY\_EMAIL** – This service is used to update the order dates and send a notification to the customer. It is invoked by the YCD\_FTC\_Delay\_Email monitor event. This service invokes the FTCHandler custom API with the arguments listed in [Table 4–24](#). For details about how this custom API can be customized, see the corresponding section. [Figure 4–7](#) illustrates this service.

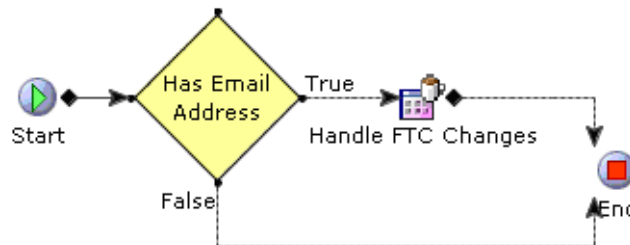
**Table 4–24 YCD\_FTC\_DELAY\_EMAIL arguments for FTCHandler**

Fields	Description
FTCDelayType	FirstFTCDelay
NoteText	FTC_First_Delay_Email_Alert_Note_Text

- YCD\_FTC\_INDEFINITE\_DELAY\_EMAIL – This service is used to update the order dates and send a notification to the customer. It is invoked by the YCD\_FTC\_Indefinite\_Delay\_Email monitor event. This service invokes the FTCHandler custom API with the arguments listed in [Table 4–25](#). For details about how this custom API can be customized, see the corresponding section. The only difference between this service and the YCD\_FTC\_DELAY\_EMAIL service are the arguments passed to the FTCHandler custom API. [Figure 4–7](#) illustrates this service.

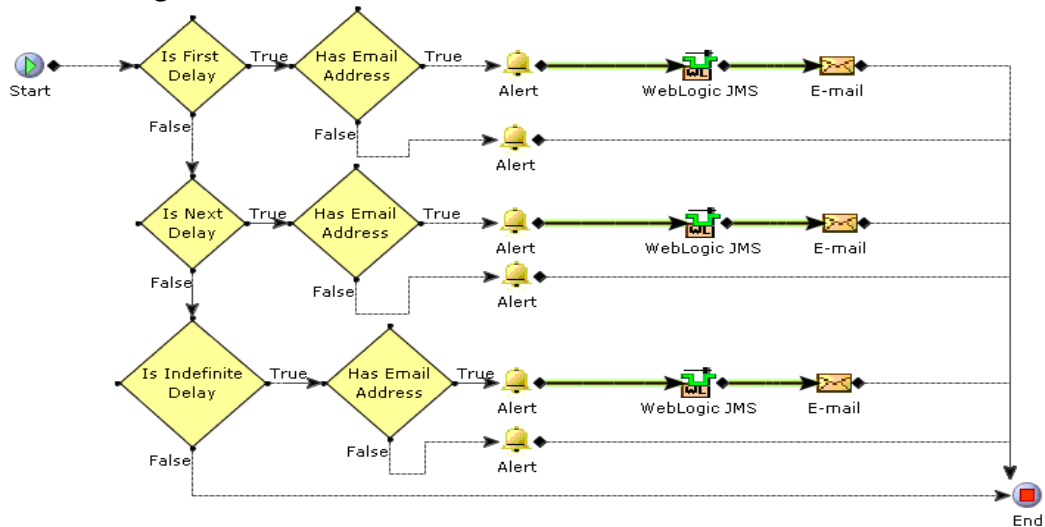
**Table 4–25 YCD\_FTC\_INDEFINITE\_DELAY\_EMAIL arguments for FTCHandler**

Fields	Description
FTCDelayType	IndefiniteFTCDelay
NoteText	FTC_Indefinite_Delay_Email_Alert_Note_Text

**Figure 4–7 YCD\_FTC\_DELAY\_EMAIL, YCD\_FTC\_FIRST\_DELAY\_EMAIL, and YCD\_FTC\_INDEFINITE\_DELAY\_EMAIL Services.**

- YCD\_FTC\_SEND\_NOTIFICATION – This service determines what type of notification is needed, and determines what the notification should state. This service is invoked by the custom API FTCHandler. This is used in all the services described until now. [Figure 4–8](#) illustrates this service.

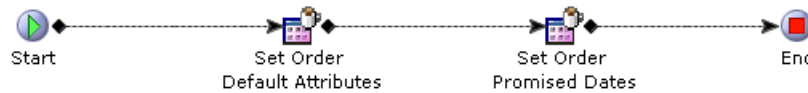
Figure 4–8 YCD\_FTC\_SEND\_NOTIFICATION Service.



- YCD\_SetOrderDefaults\_2.0 and YCD\_SetOrderDefaultsOnChange\_2.0—These two services are invoked as the beforeCreateOrderUE and beforeChangeOrderUE user exits, respectively, that are called by the createOrder API and changeOrder API. For more information about user exits and APIs, refer to the *Selling and Fulfillment Foundation: Javadocs*. These services are used, in part, to set the promised delivery dates. Under the Arguments tab of the Set Order Promised Dates component in both the services, configure the advertised promised date that is used to calculate the first promised delivery date and how many days after the first promised date the second promised date is set to by default (this value cannot exceed 30). [Figure 4–9](#) illustrates this service.



**Figure 4–9 The YCD\_SetOrderDefaults\_2.0 and YCD\_SetOrderDefaultsOnChange\_2.0 Services**



Depending on the service you are configuring, you may have to configure the Alert or E-mail sender components.

### FTCHandler Custom API

This custom API contains important parts of the logic for the FTC compliance feature. This custom API determines what type of delay is being sent, what the note text should contain, and what service should be used to send notifications to a customer by the three arguments. These three arguments are described here.

- **DelayType** – This argument tells the custom API about what it needs to do to the order. It can resolve alerts by accepting or cancelling the lines, and get all the lines on the order with the same delay type and promised date, and invoke the notification service with that list of lines. The possible values for this argument are:
  - IndefiniteFTCDelay
  - NextFTCDelay
  - FirstFTCDelay
  - AutoResolve
  - CancellLines
- **NoteText** – This argument will set the bundle key that will be used to set the note text on the order lines when they are changed.
- **NotificationService** – This argument will specify the name of the service that will run when it is determined that a notification needs to be sent. If this argument is not passed, it will be defaulted to YCD\_FTC\_SEND\_NOTIFICATION.

This service will also close the FTC Notification alerts for the corresponding lines if the DelayType argument is passed as AutoResolve or CancellLines.

### Monitor Rules

The monitor rules can be customized to notify the customer of delays, either earlier than or later than what is configured by default. By default, all the notifications are sent to the customer, at the most, 120 hours before the date, if they have not provided an e-mail address. If they have provided an e-mail address, the notification is sent, at the most, 48 hours before the date. This can be changed by going to the corresponding rule and changing the time component to suit your business requirements better. For more information about modifying the monitor rules, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

### Configuring the Alert Component

If the service raises an alert, the alert component must be configured accordingly. In particular, set the Alert Queue Name field to the queue where the alert must be sent, which, by default, is set to the Customer Notification queue. Additionally, specify the XSL used for the template of the alert, in the Template Name field.

For more information about configuring the XSL templates, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

For more information about component nodes, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

### Configuring the E-mail Component

If the service sends an e-mail notification, the e-mail component must be configured accordingly. In particular, set the E-mail Server field to the ID of the mail server user to send the notifications. Additionally, specify the XSL used for the template of the e-mail, in the Template Name field.

For more information about configuring the XSL templates, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

For more information about the component nodes, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

### Configuring Templates

The XSL components are used to convert the output XMLs into HTML messages that are sent to the customers through the e-mail components of the services. By default, the names of all the XSL files are provided by

Sterling Call Center and Sterling Store with a .sample extension. Copy these to create a new XSL file without the .sample extension, customize your XSL files as needed, and update the Template Name field of the E-mail Sender components of the relevant services.

The following XSLs are used as templates of the notification e-mail components in the default implementation of this feature:

- `<INSTALL_DIR>/repository/xapi/template/merged/email/ycd/YCD_FTC_First_Delay_notification.xsl.sample`: This XSL formats the HTML message that is sent by e-mail to the customer to inform them that the order has been delayed for the first time.
- `<INSTALL_DIR>/repository/xapi/template/merged/email/ycd/YCD_FTC_Delay_Notification.xsl.sample`: This XSL formats the HTML message that is sent by e-mail to the customer to inform them that the order has been delayed again.
- `<INSTALL_DIR>/repository/xapi/template/merged/email/ycd/YCD_FTC_Indefinite_Notification.xsl.sample`: This XSL formats the HTML message that is sent by e-mail to the customer to inform them that the order will be delayed and a new date cannot be determined.
- `<INSTALL_DIR>/repository/xapi/template/merged/exception_console/YCD_CancelOrderNotification.xsl.sample`: This XSL formats the HTML message that is sent to the user so that they can inform the customer of the auto cancellation alert if they do not have an e-mail address.

You can save it to another directory, but using the standard directory structure supplied by the Selling and Fulfillment Foundation helps ensure consistency.


### Implementation at the Enterprise Level

By default, the FTC feature is configured at the hub level. If you want this configuration to be at the enterprise level, you have two options:

- Copy the existing pipeline and monitor rules to the enterprise level.
- Create your own pipeline and monitor rules from scratch.

### Copying the Existing Hub-Level Pipeline to the Enterprise Level

Following is the recommended process for moving the configuration to the enterprise level:

1. From the Channel Applications Manager, load the configuration for the enterprise the pipeline is to be copied to.
2. Select Configure Business Process Models and Monitoring Rules.
3. Under Configure Business Process Models and Monitoring Rules, select Configure Order Fulfillment Process. If the dependent task list pop-up window is displayed, you can click the Ignore Lock icon and enter the configuration.
4. In the Pipelines tab, click the Sterling Call Center and Sterling Store Sales Order Fulfillment pipeline.
5. Click . The process modeling dialog box is displayed.
6. Enter the new pipeline ID and click OK.

**Note:** Order monitors and monitor events are automatically copied along with the pipeline.

### Creating a pipeline

This process involves creating each monitor rule and monitor event manually. For this reason, it is recommended that you copy the existing pipeline. You can always change the pipeline after it is copied. If you still want to create your own pipeline, follow these steps:

1. From the Channel Applications Manager, load the configuration for the enterprise the pipeline is to be copied to.
2. Select Configure Business Process Models and Monitoring Rules.
3. Under Configure Business Process Models and Monitoring Rules, select Configure Order Fulfillment Process. If the dependent task list pop-up window is displayed, click the Ignore Lock icon and enter the configuration.
4. In the Pipelines tab, click the Sterling Call Center and Sterling Store Sales Order Fulfillment Pipeline.
5. Select the pipeline that you created.

6. Create nine monitor rules that are configured in exactly the same manner as those configured at the hub level. For more information about how to create monitor rules, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
7. From the Channel Applications Manager, select Configure Business Process Models and Monitoring Rules.
8. Under Configure Business Process Models and Monitoring Rules, select Define Order Monitoring Events. If the dependent task list pop-up window is displayed, click the Ignore Lock icon and enter the configuration.
9. Create nine monitor events that are configured in exactly the same manner as those configured at the hub level. For more information about how to create monitor events, refer to the *Sterling Distributed Order Management: Configuration Guide*. The events should point to the same services that the hub-level monitor events point to. If you do not want to use the default services, make copies of the default services and point the monitor events to these services.

## 4.8.2 Shipment Notification

When an order ships out of a warehouse, it is important that the customers are notified of the shipment as a confirmation that everything is going well with the order.

### 4.8.2.1 Solution

Sterling Call Center and Sterling Store provides a shipment notification agent that sends an e-mail to the customer containing all shipments for a given day. For each shipment entry in the YFS\_SHIPMENT table, the NOTIFICATION\_SENT flag is maintained, which indicates whether the notification for that shipment has been sent to the customer.

One consolidated notification is sent for all shipments in a given day that the customer has not been notified. The shipment notification does not contain line item details if the shipment is not packed in a container.

To achieve this, the Shipment Notification time-triggered transaction is defined at the Sales Order document type level. The transaction reads task queue entries, and if some shipments qualify for notification, the SEND\_NOTIFICATION event is raised, which calls the

YCD\_ShipmentNotificationEmail\_1.0 service, that is responsible for sending an e-mail to the customer.

The details of the shipment notification agent are as follows:

### Attributes

**Table 4–26 Shipment Notification Agent Verification Agent Attributes**

Attribute	Value
Base Transaction ID	SHIPMENT_NOTIFICATION
Base Document Type	0001 (Sales Order)
Base Process Type	ORDER_FULFILLMENT (Sales Order)
Abstract Transaction	None
APIs Called	None

### Criteria Parameters

**Table 4–27 Shipment Notification Agent Criteria Parameters**

Parameter	Description
Action	This field is used internally by Sterling Call Center and Sterling Store. The only valid value is Get. Do not modify this field.
Number of Records to Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
CollectPendingJobs	<p>If this parameter is set to "N", the agent does not collect information on the pending jobs for this time-triggered transaction. This pending job information is used to monitor the agent in the System Management Console.</p> <p>By default, CollectPendingJobs is set to "Y". It can be set to "N" if one particular time-triggered transaction is performing a significant amount of getPendingJobs queries, and the overhead cost is too high.</p>

**Statistics Tracked**

None.

**Pending Job Count**

The number of orders for which a shipment notification has to be sent.

**Events Raised**

The following events are raised by this time-triggered transaction:

**Table 4–28 Events Raised by the Shipment Notification Agent**

Transaction/Event	Key Data	Data Published	Template Support?
SEND_NOTIFICATION	order_dbd.txt	SHIPMENT_NOTIFICATION.SEND_NOTIFICATION.0001.xml	Yes

**4.8.2.2 End-User Impact**

None.

**4.8.2.3 Implementation**

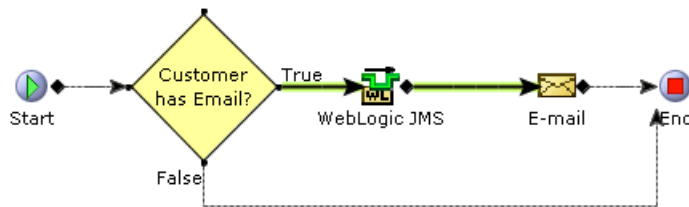
When the Shipment Notification agent raises the SEND\_NOTIFICATION event, the XML used as input to the YCD\_ShipmentNotificationEmail\_1.0 service is in:

```
<template>/event/SHIPMENT_NOTIFICATION.SEND_NOTIFICATION.0001.xml
```

**Note:** Sterling Commerce recommends that the Shipment Notification agent be run once a day, at the end of the day.

You can customize the YCD\_ShipmentNotificationEmail\_1.0 service as needed. By default, this service checks whether the customer has an e-mail address, and if that is the case sends the e-mail using the E-mail Sender component. [Figure 4–10](#) illustrates YCD\_ShipmentNotificationEmail\_1.0 service.

**Figure 4–10 The YCD\_ShipmentNotificationEmail\_1.0**



You must specify the e-mail server and e-mail server listener port in the properties of the E-mail Sender component. In the Body Template field, specify XSL used to generate the HTML message that is sent to the customer. By default, the XSL used is in the following file:

```
<INSTALL_DIR>/repository/xapi/template/merged/email/YCD_ShipmentNotification.xsl.sample
```

Copy this XSL file, remove the .sample extension, customize it as needed, and specify it in the Body Template field of the E-mail Sender component.

You can save it to another directory, but using the standard directory structure supplied by the Selling and Fulfillment Foundation helps ensure consistency.

For more information about service builder nodes and parameters, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

### 4.8.3 Order Cancellation Notification

You may cancel the orders for various reasons: when a customer does not want the order that is placed, by mistake, a duplicate is created, or the payment methods on the order turn out to be fraudulent. As soon as you cancel the order, make sure you send a notification to the customer.

#### 4.8.3.1 Solution

Sterling Call Center and Sterling Store provides a service that handles e-mail notification when an order is successfully cancelled.

When an order is cancelled through the changeOrder API, the ON\_CANCEL event is raised, which invokes the

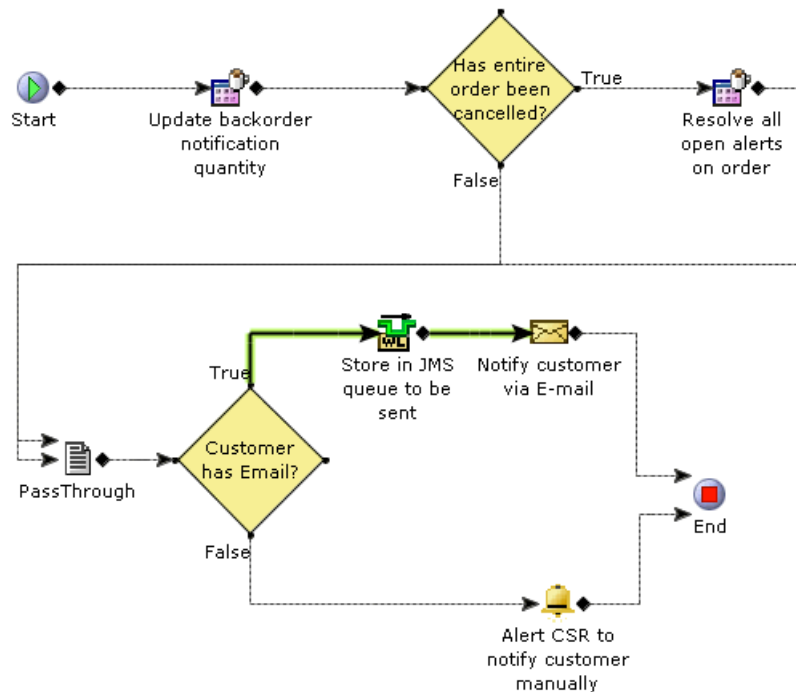


YCD\_CancelOrderNotification\_1.0 service through the YCDOnCancel action.

Before sending the notification, the service attempts to close all open alerts on the order if all order lines in that order have been cancelled. A notification is sent to the customer if they have an e-mail address defined. Otherwise, an alert is sent to the user to indicate that the user must inform the customer about the order cancellation.

Orders that are automatically cancelled when promised delivery dates are not met can be notified to the customer using this feature. For more information about order cancellation due to passed promised delivery dates, see [Section 4.8.1, "Federal Trade Commission Compliance"](#). The message sent to the customer depends on the reason, either an automatic cancellation or a regular cancellation. [Figure 4–11](#) illustrates the YCD\_CancelOrderNotification\_1.0 service.

Figure 4–11 YCD\_CancelOrderNotification\_1.0 Service



This service calls the `getOrderDetails` API to retrieve the relevant order information, and passes it on to an e-mail service that will send the information to the customer, in html format, converted by an XSL.

Although the solution above is provided with Sterling Call Center and Sterling Store, you may need to enhance the service to account for the e-mail service being potentially down. The suggested solution is instead if calling the `getOrderDetails` API, pass on the output XML of the `ON_SUCCESS` event to a queue of your choice (for example, a JMS queue), and check to see if the e-mail server is up. If it is down, leave the notification in the queue, and try to reprocess it at a later time. If it is up, call the XSL to convert the XML in the queue to HTML, and send the e-mail to the customer.

#### 4.8.3.2 End-User Impact

If an ordered is cancelled and the customer does not have an e-mail address defined, an alert is sent to the `Customer Notification` queue.

By default, the user monitors this queue and informs the customer about cancellation.

#### 4.8.3.3 Implementation

Unless you do not want to use this feature, ensure that the `ON_CANCEL` event is active in the Change Order transaction, and that the `YCDOnCancel` action invokes the `YCD_CancelOrderNotification_1.0` service.

By default, this feature is turned off because the `ON_SUCCESS` event is disabled. If you want to use this feature, enable the `ON_SUCCESS` event for the Confirm Draft Order and Create Order transactions in the Applications Manager, for the Consumer Direct Order Fulfillment document type.

The template used by the E-mail Sender component of the service to format the output XML of the `getOrderDetails` API into an HTML that is e-mailed to the customer is found in the following file:

```
<INSTALL_DIR>/repository/xapi/template/merged/email/YCD_CancelOrderNotification.xsl.sample.
```

Copy this file, name it `CancelOrderNotification.xsl`, customize it as needed, and specify the XSL used to generate the HTML message in the Body Template field of the service's E-mail Sender component.

You can save it to another directory, but using the standard directory structure supplied by the Selling and Fulfillment Foundation helps ensure consistency.

## 4.9 Order Maintenance

The user can modify orders that are created and captured in Sterling Call Center and Sterling Store. For example, customers may call the user to change the payment type or bill-to address on their order, or to cancel the order because they are dissatisfied with a late shipment. Therefore, it is very important to maintain orders.

Sterling Call Center and Sterling Store provides the following order maintenance features:

- [Apply Hold](#)
- [Remove Holds](#)

- [Change Payment Method](#)
- [Add Line](#)
- [Add Multiple Items to an Order](#)
- [Change Gift Options](#)
- [Customer Appeasement](#)
- [Price Match](#)
- [Competitor Search](#)
- [Price Match Search](#)
- [Create or Modify a Competitor Record](#)
- [Create or Modify a Price Match Record](#)
- [Reship an Item](#)
- [Add or Modify Charges](#)
- [Add Coupon](#)
- [Cancel Order](#)
- [Track an Item](#)
- [Order Notes](#)
- [Change Service Instructions](#)
- [Change Service Appointment](#)
- [Change Fulfillment Options](#)
- [Fulfillment Summary](#)
- [Reservations](#)
- [Increase Order Line Quantity](#)
- [Schedule and Release an Order](#)
- [View Procurement Orders](#)
- [Change Order Address](#)
- [Launch Application Console](#)
- [Change an Item's Style](#)

- [Change Tax Options](#)
- [Copy Order](#)
- [E-Mail Order Information](#)
- [Transfer Draft Order to Sterling Web](#)

## 4.9.1 Apply Hold

Users may sometimes want to manually place an order on hold for the purpose of verification, for example, a user may want to place an order on a Fraud Check hold after creating the order.

### 4.9.1.1 Solution

Sterling Call Center and Sterling Store enables users to place an order on hold after order creation. Based on user permissions and status modification rules, users can apply holds on the entire order or specific order lines.

The `getCompleteOrderDetails` API is used to display the Apply Hold screen and the `changeOrder` API is called when the user applies a hold.

### 4.9.1.2 End-User Impact

A user can place a hold on an order or order line only if the user belongs to a user group that has the permission to apply holds. Also, users can select only those hold types for which they have permission.

If status modification rules are configured such that users are not allowed to apply holds on an order or order line, an error message is displayed.

### 4.9.1.3 Implementation

This section explains the configurations for the apply holds task.

- To grant permission for users to apply holds on an order or order line, configure the hold type appropriately:
  - Any user can apply a hold
  - Users belonging to a particular user group can apply holds

For more information about configuring holds, refer to the *Sterling Distributed Order Management: Configuration Guide*.

- Notes entered for this task are saved on the order with YCD\_HOLD\_INFO note type.
- This task is permission controlled. You can assign permissions to user groups for this task.

### 4.9.1.4 Reference Implementation

This section explains the reference implementation provided for the apply hold task.

- Sterling Call Center and Sterling Store provides permissions to apply hold to CSR Leads and store managers.
- By default, the following order level hold types are provided for the apply hold task:
  - Credit Limit Reached
  - Duplicate Order
  - Fraud Check
  - Address Verification
- Sterling Call Center and Sterling Store provides permissions to apply the hold types provided, to all user groups.

## 4.9.2 Remove Holds

In some retail environments, holds are automatically applied to an order during order modifications. Call center and store representatives may need to manually remove these holds applied on orders and order lines.

### 4.9.2.1 Solution

Sterling Call Center and Sterling Store enables the users to remove the holds applied to an entire order or to just certain order lines.

It also provides the ability to control which users can remove which types of holds. A user cannot manually put an order on hold within the Sterling Call Center and Sterling Store user interface.

The `getCompleteOrderDetails` API is called to retrieve the details of the hold. The `changeOrder` API is called to update the status of the hold, after the hold is removed.

### 4.9.2.2 End-User Impact

A user can remove holds placed on an order or order line if that user belongs to a user group that has permissions to remove holds. Once the hold on the order is removed, the user can modify the order or order line.

### 4.9.2.3 Implementation

This section explains the configurations for the remove holds task.

- To remove holds placed on an order or order line, configure the following rules to grant permissions:
  - Any User Can process this Hold
  - Users Belonging To The Following User Groups Can Process This Hold

For more information about hold types, refer to the *Sterling Distributed Order Management: Configuration Guide*.

- Notes entered for this task are saved on the order with YCD\_HOLD\_INFO note type.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

### 4.9.2.4 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to remove holds to all user groups.

## 4.9.3 Change Payment Method

At times, a customer may want to add a new payment method or modify an existing payment method when creating or modifying an order.

---

**Note:** When a credit card payment method is selected by the user, you must validate the credit card.

---



---

**Note:** When the order header contains a CustomerContactID, the manageCustomer API saves the payment methods to the applicable customer contact.

---

#### 4.9.3.1 Solution

The Sterling Call Center and Sterling Store application supports different types of payment methods. Users can add or modify the payment methods using the Change Payment Method task. The application supports credit card validation and enables you to configure the way credit card names are displayed.

#### **Adding or Modifying a Payment Method During Order Creation or Order Modification**

When an order is modified or created, for example, if the order is cancelled or if new order lines are added, a Confirm Payment screen is displayed. Users can view the details of the existing payment methods pertaining to an order. Based on the PAYMENT\_METHOD status modification rule, users can add or modify payment methods. For more information about configuring the status modification rule, refer to the *Sterling Distributed Order Management: Configuration Guide*. For more information about adding a new payment method, see [Adding a Payment Method](#). For more information about modifying an existing payment method, see [Modifying the attributes of a Payment Method](#).

When creating an order in the context of a customer, the payment methods that were previously saved against the customer are displayed in the Confirm Payment screen. If a default payment method was chosen by the customer earlier, this payment method is populated and is charged with the order amount. However, if the customer wants to use another payment method, the default payment method must be unselected in the Confirm Payment screen. For more information about creating an order in the context of a customer, see [Customer Context Ordering](#).

#### **Modifying Payment Methods on an Order**

Sterling Call Center and Sterling Store provides the functionality to modify a payment method of an order by using one of the following screens:

- [Change Payment Methods screen](#) (payment methods are displayed in a table)
- [Enhanced Change Payment Methods Screen](#) (payment methods are displayed in panels)



Based on the configuration, either the Change Payment Methods screen or the enhanced Change Payment Methods screen is displayed. It is recommended that you use the enhanced Change Payment Method screen. The PAYMENT\_METHOD status modification rule is called in both the screens to determine whether the payment method of an order can be changed.

### Change Payment Methods screen

In the Change Payment Methods screen, the payment methods are displayed in a table. The getCompleteOrderDetails API is called to retrieve the payment methods on an order. The Details panel displays the payment types, the amount charged on the order, the amount refunded, the maximum charged amount on the payment type, and the action to be performed on the payment.

A user can activate a payment method by selecting Activate from the Actions drop-down list. To suspend an active payment method from further transactions, users can select Suspended For Charge from the Actions drop-down list. Users can also suspend a payment method for charge and refund by selecting Suspended: Charge and Refund from the Actions drop-down list.

---

**Note:** If a payment method has already been processed, it has to be manually suspended to prevent it from being charged.

---

Users can modify the attributes of an existing payment method by clicking on the payment method they want to modify. An Edit Payment Method pop-up window is displayed. For more information about modifying an existing payment method, see [Modifying the attributes of a Payment Method](#). Users can also add new payment methods by clicking the Add Payment Method hyperlink. For more information about adding a new payment method, see [Adding a Payment Method](#). The changeOrder API is called to add or modify a payment method.

---

---

**Note:** Users cannot add notes in the Change Payment Methods screen.

---

---

### Enhanced Change Payment Methods Screen

Sterling Call Center and Sterling Store provides an enhanced Change Payment Methods screen, which enables users to change the payment methods on an order. The `getCompleteOrderDetails` API is called to retrieve the payment methods, order total, and bill-to address pertaining to an order. Each payment method is displayed in a panel. The payment methods are displayed in the order of the charge sequence and the available funds, if applicable. Payment details, payment activity, maximum charged amount on the payment method, and payment transactions, if any, pertaining to each payment method is displayed.

Users can modify the attributes of some payment methods by clicking the Edit button corresponding to that payment method. For more information about modifying an existing payment method, see [Modifying the attributes of a Payment Method](#). After modifying the attributes of a payment method, the Adjust Charges pop-up window is displayed and the charges on the applicable payment methods are adjusted accordingly.

The Adjust Charges pop-up window can also be displayed by clicking the Adjust Charges button in the enhanced Change Payment Methods screen. All the payment methods are displayed in the Adjust Charges pop-up window and the charges on applicable payment methods are adjusted accordingly.

Users can remove the payment methods that are yet to be processed using the Remove button corresponding to that payment method. If a payment method has already been processed, the Revoke button is displayed instead of the Remove button. Users can click the Revoke button to suspend any further transactions to be performed on the payment method. If a payment method is revoked, users can click the Restore button to enable transactions to be performed on the payment method.

Users can also add a new payment method by clicking the Add Payment Method hyperlink. After adding a payment method, the Adjust Charges pop-up window is displayed and the charges on the applicable payment methods are adjusted accordingly. For more information about adding a

new payment method, see [Adding a Payment Method](#). The `changeOrder` API is called to add or modify a payment method.

In the enhanced Change Payment Methods screen, users can also add notes pertaining to the Change Payment Method task. The `changeOrder` API is called to add the notes.

## Adding a Payment Method

When adding a payment method, the following are taken into consideration:

- A user can add a payment method based on the user permissions and order status modification rules.
- If an order is created in the context of a customer and if the Customer Payment Method rule is configured for the customer's enterprise to allow the customer payment method information to be stored and used through the application, the user will be able to save the payment method against the customer or contact. For more information about configuring the customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. For more information about creating an order in the context of a customer, see [Customer Context Ordering](#).
- If a stored value card payment method is used, the stored value card is validated. For more information about validating a stored value card, see.
- If a credit card payment method is used, the credit card is validated. For more information about validating a credit card, see [Credit Card Validation Using Sterling Call Center and Sterling Store](#).

## Modifying the attributes of a Payment Method

When modifying the attributes of a payment method, the following are taken into consideration:

- A user can modify the payment method based on the user permissions and order status modification rules.
- If the following conditions are satisfied, users can modify the attributes pertaining to a payment method that is described in [Table 4–29](#):
  - The payment method has been saved to the order.

- The payment processing agents are yet to process the order.
- There are no existing authorizations on any other payment methods.

**Table 4–29 Attributes that can be modified for a Payment Method**

Payment Method	Attributes
Credit Card	Billing Address and Credit Card Expiration Date
Stored Value Card	Payment Reference1, Payment Reference2, and Payment Reference3
Pre-Paid	Max Charge Limit
Other	Max Charge Limit
Check	None
Refund Check	None
Customer Account	None

### Customer Context Ordering

Sterling Call Center and Sterling Store provides a configurable option to save a payment method against a customer in the Add Payment Method pop-up window provided the order is created in the context of a customer, and the Customer Payment Method rule is configured for the customer's enterprise to allow the customer payment method information to be stored and used through the application. For more information about configuring the customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

The manageCustomer API is called to save the payment method against the customer or contact.

**Note:** The address entered in the credit card payment method will not be saved against the customer.

### Handling Stored Value Cards

Sterling Call Center and Sterling Store enable you to validate a stored value card (SVC) payment method using the Sterling Sensitive Data Capture Server (SSDCS).

### Stored Value Card Validation

Sterling Call Center and Sterling Store enable you to validate the SVC payment method using SSDCS in order to comply with the Payment Application Data Security Standard (PA-DSS). For more information about configuring SSDCS, refer to the *Sterling Sensitive Data Capture Server, Release 1.0: Configuration Guide*.

When a user selects the SVC payment method, Sterling Call Center and Sterling Store applications will verify whether the value of the `yfs.ssdcs.tokenize.svc` property is Y, if the value is Y, the `createAccessToken` API is called to generate an authentication token which in turn is passed to SSDCS. A browser, which is embedded in the Stored Value Card # field is opened by SSDCS with a single entry field for the SVC number. During installation, you can customize the entry field provided by the browser to suit your preferences.

When the user enters the SVC number, SSDCS validates the SVC number and returns a display number, generated SVC token, and information pertaining to validation. If validation fails, an appropriate error message is displayed, indicating that the SVC number is invalid.

For more information about SVC validation using SSDCS, refer to the *Sterling Sensitive Data Capture Server, Release 1.0: Configuration Guide*.

### Handling Credit Cards

Sterling Call Center and Sterling Store enable you to validate the credit card payment method. Based on the configuration, the credit card payment method can be validated through Sterling Call Center, Sterling Store, or SSDCS.

Sterling Call Center and Sterling Store also enable you to configure the way credit card names are displayed.

### Credit Card Validation Using Sterling Call Center and Sterling Store

All credit cards follow a pattern in their number depending on their type. For example, the first two digits of all MasterCard credit cards are between 51 and 55, and the total length of a MasterCard card number is 16. An American Express card number has either 34 or 37 as its first two digits, and has a total length of 15.

Additionally, all numbers of a credit card follow a pattern, which can be validated with Luhn's algorithm. However, this algorithm only ensures that the given credit card number is a valid number. It does not ensure that it exists in any bank database.

It is useful to run this validation every time a new credit card number is added to save payment processing validations, which will fail every time against an invalid number.

Sterling Call Center and Sterling Store provides functionality to process real-time credit card authorizations while confirming draft orders. For more information about real-time credit card authorizations, see [Section 4.12, "Real-Time Credit Card Authorization"](#).

### **Credit Card Validation Using the Sterling Sensitive Data Capture Server**

Sterling Call Center and Sterling Store enable you to validate a credit card payment method using SSDCS in order to comply with the PA-DSS. For more information about configuring SSDCS, refer to the *Sterling Sensitive Data Capture Server, Release 1.0: Configuration Guide*.

During order creation, when a user is adding or modifying a credit card payment method, the Sterling Call Center and Sterling Store applications will verify whether the value of the `yfs.ssdcs.tokenize.cc` property is Y, if the value is Y, the `createAccessToken` API is called to generate an authentication token which in turn is passed to SSDCS. A browser, which is embedded in the Credit Card # field is opened by SSDCS with a single entry field for the credit card number. During installation, you can customize the entry field provided by the embedded browser to suit your preferences.

The user first selects a credit card type from the Credit Card Type drop-down list. The user then enters the credit card number in the Credit Card # field. When the user tabs out of the embedded browser, the credit card number is automatically passed to SSDCS, which in turn validates the credit card. If validation succeeds, it returns a generated credit card token, a display number, and a credit card type to the Sterling Call Center and Sterling Store applications. The display number is displayed instead of the credit card number in the Credit Card # field. If the credit card type returned by SSDCS is not the credit card type that was selected by the user, the Credit Card Type field will be updated to reflect the credit card type returned by SSDCS. If the credit card validation fails, an appropriate error message is displayed, indicating that the credit card

number that has been entered is invalid. The error message can be customized through SSDCS. Also, the error message will be subjected to run through the localization bundle.

For more information about credit card validation using SSDCS, refer to the *Sterling Sensitive Data Capture Server, Release 1.0: Configuration Guide*

### Displaying Credit Card Name

Sterling Call Center and Sterling Store enables you to configure whether a text field or multiple text fields should be used for entering credit card name. For more information about displaying credit card name, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

---

---

**Note:** Sterling Call Center and Sterling Store does not allow you to retrieve sensitive payment information such as credit card number, SVC number, customer's account number.

---

---



---

---

**Note:** Ensure that the Display Credit Card Name rule is same for an enterprise for which you are creating an order, and for the customers defined for an enterprise.

---

---

The credit card number is validated as follows:

- [Prefix and Length Validation](#)
- [Luhn's Algorithm](#)

### Prefix and Length Validation

[Table 4–30](#) specifies the exact numbers to prefix for a card number of a specific card type, and how many digits the card number must contain. For example, American Express cards all start with either 34 or 37 as their first digits, and contain 15 digits.

---

---

**Note:** The JCB credit card type has two entries in [Table 4–30](#). This is due to the fact that a JCB card can follow the prefix system listed in the table.

---

---

**Table 4–30 Credit Card Types**

Credit Card Type	Prefix	Length
American Express	34, or 37	15
MasterCard	51 through 55	16
Visa	4	13 or 16
Diners Club and Carte Blanche	36, 38, or 300 through 305	14
Discover	6011	16
JCB	2123 or 1800	15
JCB	3	16

When a credit card type is selected and a credit card number is entered in the Application Console, the system validates the number prefixed and the length. If both are valid, the system executes the Luhn's algorithm.

### Luhn's Algorithm

Luhn's algorithm determines whether or not a credit card number is valid. For a given credit card number:

1. Double the value of every other digit from right to left, beginning with the second to last digit.
2. Add the digits of the results of [Step 1](#) to the remaining digits in the credit card number.
3. If the result mod 10 is equal to 0, the number is valid. If the result mod 10 is not equal to 0, the validation fails.

### Example

A credit card of type Visa is added with the number 4624 7482 3324 9080.

This credit card number starts with 4, as it should be in all Visa cards. It also has 16 digits, which is valid for a Visa card. Now, execute Luhn's algorithm:

- a. Double the value of every other digit from right to left, beginning with the second to last digit.

**4624 7482 3324 9080:**



- $8 * 2 = 16$
  - $9 * 2 = 18$
  - $2 * 2 = 4$
  - $3 * 2 = 6$
  - $8 * 2 = 16$
  - $7 * 2 = 14$
  - $2 * 2 = 4$
  - $4 * 2 = 8$
- b. Add the digits of the results of the previous step to the remaining digits in the credit card number.
- The digits of the result of [Step a](#) amount to the following:
  - $1 + 6 + 1 + 8 + 4 + 6 + 1 + 6 + 1 + 4 + 4 + 8 = 50$
  - The remaining digits in the credit card number amount to the following:
  - $6 + 4 + 4 + 2 + 3 + 4 + 0 + 0 = 23$
  - The sum of the two sub results is:
  - $23 + 50 = 73$
- c. If the result mod 10 is equal to 0, the number is valid. If the result mod 10 is not equal to 0, the validation fails.

$73 \text{ mod } 10$  is 3, therefore the card number is not valid.

A Visa credit card of number 4624 7482 3324 9780, on the other hand, passes the validation.

---

**Note:** You can override the validation for the credit card types listed in [Table 4–30](#) by implementing YCDValidateCreditCardInfoUE user exit. If you are configuring new credit card types, the validation for the new credit card types should also be implemented by the YCDValidateCreditCardInfoUE user exit.

---

#### 4.9.3.2 End-User Impact

- Based on the configuration, a user will see either the Change Payment Methods screen or the enhanced Change Payment Methods screen.
- After a user enters the credit card number and clicks the Save button, the system validates the credit card number.

#### 4.9.3.3 Implementation

This section explains the configurations for the Change Payment Method task.

- You can configure the Sterling Call Center and Sterling Store application to display either the old Change Payment Methods screen or the enhanced Change Payment Methods screen.
- Sterling Call Center and Sterling Store enable you to configure the order statuses to allow users to add or modify payment methods. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store enable you to validate a credit card payment method using SSDCS. For more information about configuring SSDCS to validate credit cards, refer to the *Sterling Sensitive Data Capture Server, Release 1.0: Configuration Guide*.
- Sterling Call Center and Sterling Store enable you to validate an SVC payment method using SSDCS. For more information about configuring SSDCS to validate an SVC, refer to the *Sterling Sensitive Data Capture Server, Release 1.0: Configuration Guide*.
- You can customize the credit card number field and the SVC number field in the browser provided by SSDCS. The browser contains a single entry field that is used for the credit card number or SVC number entered by users. You can define a `css` file to customize the appearance of the entry field and pass the `css` file as a value to the `yfs.properties_ycd_ext` property file as follows:

```
ycd.ssdcs.<theme>.cssurl=/css/<theme>_sccs_ssdcs.css
```

For example, `ycd.ssdcs.ice.cssurl=/css/ice_sccs_ssdcs.css`

- Sterling Call Center and Sterling Store allow you to configure different credit card types. For more information about configuring different

credit card types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- If you have configured a credit card type other than the types listed in [Table 4–30](#), implement the YCDValidateCreditCardInfoUE user exit to validate this credit card type. Implement the YCDValidateCreditCardInfoUE user exit to override the validation for the credit card types listed in [Table 4–30](#).
- Notes entered in the enhanced Change Payment Method screen are saved on the order with the YCD\_PAYMENT\_CHANGE note type.
- Configure the Customer Payment Method rule for a customer's enterprise if you want to save the payment method against a customer when adding a new payment method. This functionality is available only when an order is created in the context of a customer. If the order is not created in the context of a customer, users will not be able to save the payment method against the customer or contact. For more information about configuring the customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.9.3.4 Reference Implementation

This section explains the reference implementations provided for the Change Payment Method task:

- By default, the enhanced Change Payment Methods screen is displayed.
- By default, Sterling Call Center and Sterling Store provides permissions to all users to change the payment methods on an order.
- By default, the Allow Customer Payment Method Information To Be Stored And Used Through The User Interface check box is selected to enable users to save the payment methods against the customer or contact. For more information about configuring the customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Following are the default payment types that are configured:
  - Credit Card
  - Pre-paid
  - Check

- Refund Check
  - Customer Account
  - Stored Value Card
  - Other Payment Method
- Sterling Call Center and Sterling Store supports the following pre-defined credit card types:
  - American Express
  - Master Card
  - Visa
  - Diners Club
  - Carte Blanche
  - Discover
  - JCB
- Sterling Call Center and Sterling Store provides a reference implementation of YCDValidateCreditCardInfoUE user exit, which fails the validation for the following credit card numbers: 7890123456789012, 8901234567890123, and 9012345678901234. This user exit does not validate the credit card information if the credit card number is 8091234567890123. For all other credit card numbers, the validation succeeds.

### 4.9.4 Add Line

After placing an order, a customer may want to add new product or service items to the order, if the customer has forgotten to place the order for product or service items earlier, or if it was unavailable at the time of placing the initial order.

#### 4.9.4.1 Solution

Sterling Call Center provides an add line task which the user can use to add order lines to the existing order. This section explains the complete process of adding a new order line to an existing order.

## Validating Whether A New Item Can Be Added

Sterling Call Center and Sterling Store allows you to validate whether the user can add a new order line to an existing order based on the following criteria:

- Order Age—You can configure the number of days after an order is created to allow the addition of new order lines to an existing order.
- Order Modification rule—You can configure the order statuses for which the add line action is allowed.

## Searching For An Item To Be Added

The `getCompleteItemList` API is called that retrieves a list of items based on the input criteria. The list of items includes additional information such as pricing, availability and promotions.

Sterling Call Center and Sterling Store does not allow users to add items which are always tag controlled to an order.

## Choosing The Fulfillment Option And Quantity

If an item is available that can be added to an order line, choose the required quantity and the delivery option. The `getFulfillmentOptionsForLines` API is used to obtain the available fulfillment options, which include delivery, shipping, and pickup options for a set of new or existing order lines. The customer can also view details about the nearby stores from where the customer can pick the item.

The `getSurroundingNodeList` API is called that provides a list of stores within a specified radius, from where the item can be picked up. To configure the radius for stores from where the items can be picked up, the following rules are considered:

- Distance To Consider rule—You can configure the distance from the current location for stores from where the customer can pick the items.
- UOM To Consider rule—You can configure the UOM for the distance of stores from the current location.

If the user enters the delivery address and selects the 'ship to' carrier service code, the `getFulfillmentOptionsForLines` API is used to determine

the availability of 'delivery address' and 'carrier service' the user has input.

**Note:** If the `getFulfillmentOptionsForLines` API determines that both delivery and shipping are allowed, the fulfillment method is given the preference and the UI displays the fulfillment method as "Available to Deliver".

To display the carrier service information, the `Y_ANY` carrier organization must be implemented with the associated carrier services.

### Choosing Accessories

The `getCompleteItemList` API is used to determine the accessories for an item. The user can select fulfillment options and the quantity for each of the selected accessories.

### Choosing A Delivery Appointment

After adding the items to an order using the `changeOrder` API, the `generateWorkOrder` API is used to create any delivery lines (if the chosen delivery option is 'Delivery'), and generate work orders for the delivery lines.

The user can change appointment for any generated work order, if necessary.

### Configuring Reasons For Overriding Item Price

The item prices can be overridden during the Add Line process using the `YCD_PRICEOVERRIDE` common code. You can configure reasons such as substitution, price adjustment, and price match to allow override.

#### 4.9.4.2 End-User Impact

The user will not be able to add new order lines to an existing order after the configured amount of days are passed. Addition of order lines to an order also depends on the status of the order. Addition of order lines is done through the status modification rules. The modification types performed for the Add Line process include:

- Add Line

- Associate Delivery Line With Product Line
- Change Appointment

#### 4.9.4.3 Implementation

This section describes how to configure the Add Line rule:

1. Log in to the Channel Applications Manager. For more information about logging in to the application, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
2. From the Channel Applications Manager, select Configure Order Administration > Advanced Configurations > Add Line Rules.
3. In the Maximum Number Of Days After Order To Allow Adding An Order Line field, enter the number of days allowed to add an order line.
4. From the Applications Manager, select Configure Order Administration > Configure Order Modification Rules > Add Line > Order.
5. Select the order statuses for which the add line action is allowed.
6. From the Channel Applications Manager, select Order Promising > Store Pickup Rules.
  - Check The Enterprise Allows Pickup From Store Fulfillment box to indicate that an item is available for pick up.
  - In Distance To Consider When Finding Nearby Stores field, enter the appropriate distance.
  - In Distance UOM, enter the appropriate UOM for the distance.
7. Click Save.

To define the Y\_ANY carrier service.

1. From the Channel Applications Manager, select Configure Outbound Logistics > Define Carriers > Roles and Participation > Carrier Attributes > Parcel Services.
2. Define the Y\_ANY carrier service. For more information about defining carrier services, refer to the *Sterling Logistics Management: Configuration Guide*.

Notes entered for this task are saved on the order with YCD\_NEW\_ITEM\_INFO note type.

### 4.9.4.4 Reference Implementation

This section explains the default values that are provided to configure rules to add an order line.

- The maximum number of days allowed after an order is created to allow addition of new order lines to an existing order is 30.
- If pickup from a store is allowed for the items of the order, the distance radius to consider when finding nearby stores is 25 miles.

The following reasons are configured in the YCD\_PRICEOVERRIDE common code:

- Substitution
- Price Adjustment
- Price Match

All rules for the Add Line task are defined for all the organizations.

As part of the reference implementation provided by Sterling Call Center and Sterling Store, the Order Repricing user exit is enabled when additional items are added to the Order.

## 4.9.5 Add Multiple Items to an Order

After placing an order, a customer may want to add more of an item, new items or service items to the order.

### 4.9.5.1 Solution

Sterling Call Center and Sterling Store provide a task to add multiple items to an order. This section explains the solution offered by Sterling Call Center and Sterling Store for the process of adding multiple items to an order.

### Validating Whether New Items Can Be Added

Sterling Call Center and Sterling Store validates if new order lines can be added to an existing order based on the following configurations:

- Order Age—The number of days after an order is created to allow the addition of new order lines to an existing order.
- Order Modification rule—The order statuses for which permit addition of lines to an order.



## Adding Items to the Order

This section explains the item entry step in the add multiple items task and the solution offered by Sterling Call Center and Sterling Store.

The user cannot change any of the existing order lines. For more information about adding new order lines, see [Adding Items to an Order](#).

## Adding or Deleting Lines from an Order

The changeOrder API is called to add only new lines to the order. The changeOrder API is also called to delete existing lines from the order.

Sterling Call Center and Sterling Store assumes that the status modification rules allow addition of lines to cancelled orders. If the status modification rules do not allow addition of lines to cancelled orders, the user will not be able to add new lines to the order and remove all existing lines at the same time.

## Validating Items in an Order

Sterling Call Center and Sterling Store validates the new items that are added to an order. If a newly added item fails validation, an error message is displayed. For more information about validating items in an order, see [Section 4.6.4.4, "Item Validation"](#).

## Changing Fulfillment Options

The user can modify the fulfillment options for the items added. For more information about changing fulfillment options, see [Section 4.9.19, "Change Fulfillment Options"](#).

## Viewing Service Fulfillment Summary

After adding items to the order using the changeOrder API, if you choose the Delivery option, the generateWorkOrder API is used to create the delivery lines and generate work orders. For more information about service fulfillment summary, see [Section 4.9.24, "Service Fulfillment Summary"](#).

## Changing Service Appointments

The user can change the delivery appointment for the customer, if applicable. The items are reserved based on the configuration. For more information about service appointments, see [Section 4.9.23, "Change Service Appointment"](#).

### Viewing Pick up and Shipping Fulfillment Summary

The Fulfillment Summary screen provides the complete scenario of how the order is configured and fulfilled. The items are reserved based on the configuration. For more information about fulfillment summary, see [Section 4.9.20, "Fulfillment Summary"](#).

### Viewing and Confirming Payment Details

Due to the addition of items, the payment information changes, and the system displays the updated payment details. To make additional payments, the user must select the payment method.

Sterling Call Center and Sterling Store can be used to carry out the critical payment related processes during order management processing and enables you to integrate with external payment processing systems such as CyberSource or Chase Paymentech. For more information about integrating with external systems for payment handling, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.9.5.2 End-User Impact

When the user adds related items to an order, the fulfillment method is defaulted to the fulfillment method of the parent item.

The user will not be shown certain columns in the order line entry screen if the user has configured not to display.

The user will be able to add bundle items to the order.

Depending on how you configure the order entry rules, the end users will see different behavior in the user interface:

- The user will see either the Item Entry screen or Large Order Item Entry screen based on the configuration.
- The user will see Shipping and Delivery as separate fulfillment methods based on the configuration to display Shipping and Delivery as separate fulfillment methods.
- The user will only see the set of columns that you configure on the Add Line screen.
- The user will see related items as a panel on the Add Line screen or as a pop-up window based on the configuration.

- The user will see item associations that are configured as Show in UI. The user will see the availability for the associated items based on the inventory check for associated items. Based on the configuration, when an associated item is added to an order, the associated item either replaces the original item or is added as a related item.

#### 4.9.5.3 Implementation

This section explains the configurations for the add lines to order task.

- Sterling Call Center and Sterling Store allows you to configure the Order Age beyond which new lines cannot be added. For more information about configuring the add line rule, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the order statuses for which addition of new order lines is allowed. Ensure that the status modification rules allow addition of lines to cancelled order. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the store pick up rules. For more information about configuring the store pick up rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure rules for adding lines to the order. For more information about configuring order entry rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the display of model items. For more information about configuring the display of model items, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure external payment options. For more information about configuring external payment options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- To add relationships between lines, you need to configure the association types to be displayed on the user interface. You also need to configure a Relationship Type to be used with the association type.

For more information about configuring relationship types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- Sterling Call Center and Sterling Store allows you to configure availability checks and reservation options. For more information about configuring availability checks and reservation options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission-controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.
- Notes entered for this task are saved in the order using the YCD\_MULTI\_ADD\_LINES\_INFO note type.

#### 4.9.5.4 Reference Implementation

This section explains the reference implementation provided for the add lines to order task.

- By default, the Maximum number of days after order date to disallow adding an order line rule is set to 30 days.
- By default, the Add Items screen is displayed for item entry.
- By default, Shipping and Delivery are displayed as separate fulfillment options and the default fulfillment method is Delivery.
- By default, the following columns are displayed in the line entry step:
  - Unit of Measure
  - Line Quantity
  - Line Price
  - Line Totals
- By default, related items are displayed as a pop-up window.
- As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to add lines to an order to all user groups.

## 4.9.6 Change Gift Options

A customer may want to place an order for some gift items. In such situations, you can configure the available gift options to ensure that the selected items are marked as gift items.

### 4.9.6.1 Solution

Sterling Call Center and Sterling Store user interface provides two screens to change gift options.

To change gift option for pick up and shipping lines, the Fulfillment Summary screen is used. For more information about changing gift options for pick up and shipping lines, see [Marking Items as Gifts](#). To change gift options of delivery lines, the Delivery Fulfillment Summary screen is provided. Based on the Available Gift Options Rule in the Channel Applications Manager, users can mark either the entire work order or individual work order lines as a gift. For more information about changing gift options for delivery lines, see [Marking Delivery Lines As Gift Items](#).

You can configure which of the fulfillment methods allow gift options to be recorded in the user interface.

The Sterling Call Center and Sterling Store application also enables users to include a gift message for the item marked as a gift. Gift messages are stored as instructions with a GIFT instruction type. Additionally, if the Allow Gift Wrapping Of Items check box is selected in Configure Available Gift Options of the Channel Applications Manager, the application will enable a user to gift wrap an item marked as a gift. For more information about configuring the available gift options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

The `getCompleteOrderDetails` API is called to get details of the order lines marked for a gift recipient. The `changeOrder` API is called to record the gift option on the order.

### Handling Bundle Items

If a bundle item is configured as ship independent, gift options can be changed for all the corresponding component items. Users can gift wrap only those component items that support gift wrapping. If a bundle item is configured as ship together, gift options can be changed only for the bundle parent. Users can gift wrap the bundle parent only if it supports

gift wrapping. For more information about bundles, see [Section 4.2, "Configured Items"](#).

#### 4.9.6.2 End-User Impact

This section explains the end-user impact for the Change Gift Options task:

- Based on the Configuring Available Gift Options rule, the user can enter gift options for product items, bundle items which are being shipped, picked up, or being delivered.
- The Gift Wrap Item check box will be displayed to the user if both the following conditions are satisfied:
  - The ship node is configured to provide gift wrapping services. For more information about configuring the attributes of a node, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
  - The Configuring Available Gift Options rule in the Channel Applications Manager is configured to allow gift wrapping of an item.

#### 4.9.6.3 Implementation

This section explains the configurations for the change gift options task.

- You can configure the gift options for the following items:
  - Items being Picked Up
  - Items being Shipped
  - Items being Delivered

You can also configure the gift options to allow users to select either the entire work order or individual work order lines as a gift.

For more information about configuring the available gift options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- You can configure a ship node to provide gift wrapping services. For more information about configuring the attributes of a node, refer to

the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

- You can configure the Sterling Call Center and Sterling Store application to enable users to gift wrap an item. For more information about configuring the available gift options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the order statuses for which changing gift options is allowed. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- If you want to modify permissions for this task, you must also modify permissions for the following actions:
  - Mark Shipping Lines As a Gift actions
  - Enter Gift Recipient Info
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### 4.9.6.4 Reference Implementation

This section explains the reference implementation provided as part of the change gift options task.

- The Items being Picked Up, Items being Shipped, and Items being Delivered rules are set to "Y" for all the organizations.
- By default, the Sterling Call Center and Sterling Store is configured to enable users to select an entire work order as a gift.
- By default, the Sterling Call Center and Sterling Store application does not allow users to gift wrap an item marked as a gift.
- As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to change the gift options to all user groups.

### 4.9.7 Customer Appeasement

Sometimes you may have to appease the customer when a customer reports of not being satisfied with any of the services provided. For instance, a customer may report problems faced when placing an order

online or report of receiving a damaged item. In such situations, you need to perform the appeasement task to make amends for the bad experiences.

### 4.9.7.1 Solution

This section explains the customer appeasement process and the solution offered by Sterling Call Center and Sterling Store.

#### Selecting Appeasement Reasons

The customers may call a user to report various reasons for which a user needs to perform an appeasement action on the customer's order. The appeasement may apply to the entire order or only some of the items in the order. The `getCommonCodeList` API is invoked, which retrieves all common codes that match the input criteria. The reasons for appeasement are configured using the `YCD_APPEASEMENT_RSN` common code. You can configure reasons such as damaged item, bad phone experience and so forth to enable the appeasement action.

#### Selecting An Appeasement Offer

The `YCDGetAppeasementOffersUE` user exit returns the appeasement offers. This user exit provides an option to plug in custom logic to retrieve appeasement offers. The offers provided depend on the selected appeasement reason and order total details. For example, you can provide 10% discount on the current order. Using the `YCDGetAppeasementOffersUE` user exit, you can implement different appeasement offers, if applicable.

If the appeasement applies to the current order, the `recordInvoiceCreation` API is called which creates a `CREDIT_MEMO` invoice for the appeasement. If the selected appeasement offer applies for future orders, the `YCDSendFutureOrderCustomerAppeasementUE` user exit is invoked. This user exit implementation has the responsibility (like informing the promotion engine) to apply the appeasement on future orders of the same customer.

### 4.9.7.2 End-User Impact

None.



### 4.9.7.3 Implementation

This section explains the configurations for this feature.

- You can configure to provide a list of reason codes that are available during the customer appeasement process. For more information about defining appeasement reasons, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Notes entered for this task are saved on the order with YCD\_CUSTOMER\_APEASE note type.

### 4.9.7.4 Reference Implementation

This section explains the various factory setups that need to be configured:

1. Create a new charge category called CUSTOMER\_APEASEMENT and describe it as Customer Appeasement. This is billable.
2. Create a new charge name called CUSTOMER\_APEASEMENT and describe it as Customer Appeasement.
3. Configure the following appeasement reasons as common codes:
  - Late Shipment
  - Damaged Items
  - Bad Delivery Experience
  - Bad Online Experience
  - Bad Phone Experience
4. The default implementation of the YCDGetAppeasementOffersUE user exit returns the following offers:
  - FLAT\_AMOUNT\_ORDER (if the offer provides a flat amount discount. The discount is applied only at the order header level)
    - A flat discount of \$10 is returned.
  - PERCENT\_ORDER (if the offer provides a percentage discount on the current order. The discount is applied at both order and order line level).
    - A discount of 10% is returned on the order.

- PERCENTAGE\_FUTURE\_ORDER (if the offer provides a percentage discount on the future order).
    - A discount of 5% for the future order is returned.
  - VARIABLE\_AMOUNT\_ORDER (if the offer allows the user to enter the discount amount. The discount is applied only at the order header level).
    - This is a flat discount and the amount is entered by the user.
5. The default implementation of the YCDSendFutureOrderCustomerAppeasementUE user exit does not return any offer because it is the implementation of the logic to send the future order appeasement offer to the promotion engine.

This set up is provided for all the organizations.

### 4.9.8 Price Match

A customer may sometimes find that an item is available for a cheaper price at a different retailer. In such situations, the call center or store associate can investigate the customer's claims and perform the necessary price match action.

#### 4.9.8.1 Solution

Sterling Call Center and Sterling Store provides user interfaces for maintaining a list of price match details, competitor details, and for recording price match information against an order line. Users can perform the price match task, all the while ensuring the user acts in accordance with the price match policies.

This section explains the Price Match process and the solution offered by Sterling Call Center and Sterling Store.

#### Validating If An Item Can Be Price Matched

Sterling Call Center and Sterling Store provides the ability to determine which items can be price matched. In the Price Match Line Selection screen, the YCDCanPriceMatchBePerformedUE user exit is called to determine which items can be price matched. By default, the logic provided in the default implementation user exit considers the price match window rule. This rule specifies the exact time period beyond which the user cannot perform a price match. For example, you can use

this rule if you have a 15 day guaranteed price period. You can, however, provide a different validation logic by creating your own implementation for this user exit. However, even after the YCDCanPriceMatchBePerformedUE user exit determines that a price match cannot be performed, the user can still perform the price match if the user has permissions.

### **Handling Bundle Items**

The Price Match Line Selection screen does not allow users to price match bundle component items. For more information about bundles, see [Section 4.2, "Configured Items"](#).

### **Searching For A Competitor**

When the customer reports that a competitor is selling an item at a cheaper price, the call center or store representative can search whether the competitor already exists in the database. If the competitor does not exist and if the user has the permissions to create a competitor, the user can create the competitor and proceed with the price match. If the user does not have the permissions to create a competitor, price match cannot be performed.

The getCompetitorList API is called to get the list of competitors and the manageCompetitor API is called to create a competitor. For more information about creating a competitor, see [Section 4.9.11, "Create or Modify a Competitor Record"](#).

### **Checking For Prior Price Matches**

When the user selects the item to perform a price match and then selects the competitor, the system checks for prior price matches performed for that item. If prior price matches are found, the user can choose from a list of prior-approved price matches. This allows the user to apply the price match quickly without having the user to waste time in investigating the price match. The getPriceMatchList API is called to get the list of prior price matches.

### **Price Match Worksheet Calculation**

If the user cannot find a prior-approved price match or if the user wants to manually enter the field values, the user can enter the competitor's price, charges and discounts and modify the existing shipping charges and discounts in the Price Match Worksheet. Discounts can be modified

at the order header level and line level. For more information about adding discounts, see [Section 4.9.14, "Add or Modify Charges"](#).

If a customer requests for a price match that is less than a previously rejected price match, the price match is rejected. If a customer requests for a price match that is higher than a previously approved price match, the price match is approved.

The Refund Amount Per Unit From Price Match is calculated by applying the price match percentage defined for the competitor to the difference between the competitor total price and existing total price.

### **Defaulting Specific Charges For A Price Match**

An enterprise can configure which of the order line charges needs to be taken into consideration as shipping charge when performing a price match. Depending on the configuration, the specific order line charge is defaulted on the Price Match Worksheet.

### **Checking For "No Hassle" Criteria**

If none of the prior price matches can be used to approve or disapprove the customer's requested price match, the system will perform a "No Hassle" price match check to verify if the price match can be given to the customer without a CSR or store representative having to investigate the customer's price match claims manually.

The `checkNoHassleCriteria` API is used to validate the price match.

Sterling Call Center and Sterling Store allows you to configure the following "No Hassle" rule:

- **Maximum No Hassle Unit Price Difference**—According to this rule, the price match action is allowed for items whose price difference falls within the configured range.
- **Maximum No Hassle Unit Price Difference Percentage**—This is the maximum percentage price difference rule that allows to price match an item. The maximum percentage difference between the current item's price and the competitor's price can be configured to disallow the price match action if this difference is exceeded.
- **Maximum No Hassle Line Quantity**—The maximum ordered item quantity should not exceed a certain limit if price match is to be allowed. When a user performs the "No Hassle" price check, the system compares the price match information against the rules that

have been configured using the `checkNoHasslePriceMatch` API. By default, this API considers only the rules mentioned above. You can implement different "No Hassle" logic by implementing the `checkNoHasslePriceMatchUE` user exit.

If the system determines that the requested price match falls within the "No Hassle" criteria, it allows the user to apply the price match to the order.

If the requested price match does not fall within the "No Hassle" criteria, the user should either manually investigate the price match by viewing the competitor's Web site or sales flyers, or can create a follow-up alert for this price match request that is to be investigated later. In the latter scenario, the user should inform the customers that they will be contacted later about the price match request.

### **Overriding Disallowed Price Matches**

If, despite a price match not being allowed because of non-compliance with the "No Hassle" criteria, users can still perform the price match if they have the necessary permissions. They can also specify an expiration date for the price match.

### **Recording A Price Match On An Order Line**

When a user performs a new price match, the system saves the new price match information. Whether an absolute price match or a possible price match, the `changeOrder` API is called to apply the price match. The Price Match Recorded As enterprise rule is used to determine the way in which to apply the price match. The user can record price match as a charge in the unit price, or apply with the charge category as discount. The charge name is according to the Price Match Charge Name rule.

### **Creation Of Alerts**

If a price match is not allowed because of non-compliance with "No Hassle" criteria, the price match is placed in `Pending` status, and an alert is raised. The `managePriceMatch` API is called to create a price match record. The `createException` API is used to create an alert of the type `YCD_PRICE_MATCH`. The alert is assigned to the `YCD_PRICE_MATCH` queue. For more information about the creation of alerts, see [Section 4.23, "Alert and Queue Management"](#).

## Approval Of Pending Price Matches

Sterling Call Center and Sterling Store provides two ways in which to approve the pending price matches:

- **Alert Details screen**—The user can approve the price match records that are in `Pending` status from the Alert Details screen. For more information about searching for alerts and viewing alert details, see [Section 4.23, "Alert and Queue Management"](#).
- **Price Match Details screen**—The user can approve the price match records that are in `Pending` status from the Price Match Details screen. For more information about viewing price match details, see [Section 4.9.12, "Create or Modify a Price Match Record"](#).

A user with the necessary permissions to search for alerts can view the alerts generated for price matches that are in `Pending` status. The Alert Details screen displays the list of order lines on which the price match was attempted, by calling the `getOrderLinesForPriceMatch` API. The user can either approve or deny the price match.

If the price match is approved, the `managePriceMatch` API is called to change the price match status and the `changeOrder` API is called to apply the price match on those order lines on which price match is possible.

If the price match is denied, the `managePriceMatch` API is called to change the price match status, and the order lines are not updated.

## Viewing and Confirming Payment Details

The Payment Confirmation screen displays the summary of the payment impact on the order as a result of the price match. If payment is configured to be handled outside Sterling Call Center and Sterling Store, the Payment Confirmation screen is not displayed.

## Purging Of Price Match Records

In order to minimize the number of unused price match records in the database, Sterling Call Center and Sterling Store provides a price match purge agent. If the number of days that have elapsed after the price match expiry date is greater than the configured number of retention days, the agent moves the price match records to the price match history table. If a price match record is still present in the `YFS_ORDER_LINE_PRICE_MATCH` table, the price match record cannot be purged. In such situations, the price match record will be purged only

after the record is purged from the YFS\_ORDER\_LINE\_PRICE\_MATCH table.

For more information about configuring the agent criteria details, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

The price match purge retention days can be configured using the Channel Applications Manager.

### **Purging Of Price Match History Records**

Sterling Call Center and Sterling Store also provides the price match history purge agent that purges the price match history records from the YFS\_PRICE\_MATCH\_H table.

For more information about configuring the agent criteria details, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

The price match history purge retention days can be configured using the Channel Applications Manager.

### **4.9.8.2 End-User Impact**

Based on the rules to check the "No Hassle" criteria, the user will be able to either allow or disallow price matches.

### **4.9.8.3 Implementation**

This section explains the configurations for the price match task.

- Sterling Call Center and Sterling Store allows you to configure the criteria that determines when a price match can be performed on an item in an order. For more information about configuring price match rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to specify the retention days for price match records and price match history records through purge rules. For more information about configuring the price match purge retention days and price match history purge retention days, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- Implement the YCDCanPriceMatchBePerformedUE user exit to verify if an item can be price matched.
- Implement the checkNoHasslePriceMatchUE user exit to validate if the price match satisfies the "No Hassle" criteria rule.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.
- Sterling Call Center and Sterling Store allows you to provide permissions to users to override and perform price matches for items that do not satisfy the price match window rule.
- Sterling Call Center and Sterling Store allows you to provide the necessary permissions to users to override and perform price matches for items that do not satisfy the "No Hassle" criteria rules.
- Sterling Call Center and Sterling Store allows you to provide the necessary permissions to specific users to create competitors.
- If you are providing the necessary permissions to perform a price match, you should also provide permissions for the following tasks:
  - Create Price Match
  - Search Competitor
  - Search Price Match
  - Search for Alerts
- Notes entered for this task are saved in the order using the YCD\_PRICE\_MATCH note type.

#### 4.9.8.4 Reference Implementation

This section lists the default values that are provided as a part of the reference implementation to configure the rules to perform price match for an item.

- The Maximum Number of Days after Order Line Was Created to Allow Price Match rule is set to 14 days.
- The Maximum No Hassle Unit Price Difference rule is set to \$100.
- The Maximum No Hassle Unit Price Difference (%) rule is set to 25%.
- The Maximum No Hassle Line Quantity rule is set to 10.



- The Default Number of Days for Price Matches to Expire rule is set to 30 days.
- The Price Match Percentage Applied rule is set to 100%.
- The Charge Category Representing Shipping Charges To Default On The Price Match WorkSheet rule is not set by default.
- The Price Match Recorded As rule is set to Change In Order Line's Unit Price.
- Sterling Call Center and Sterling Store provides permissions to record price match to all the user groups as part of reference implementation.
- Sterling Call Center and Sterling Store provides permissions to override the price match window rule to the CSR lead user group and store managers as part of reference implementation.
- Sterling Call Center and Sterling Store provides the necessary permissions to override the "No Hassle" criteria rules to CSR lead user group and store managers as part of reference implementation.

All the rules for price match are defined for all the organizations.

## 4.9.9 Competitor Search

The call center and store representatives may have to search for existing competitors when performing a price match. Also, a retailer may want to maintain a list of valid competitors for whom a price match can be performed.

### 4.9.9.1 Solution

Sterling Call Center and Sterling Store provides a user interface that enables users to search, and edit competitor records. If a user has access to multiple enterprises, the Enterprise drop-down list is displayed in the Competitor Search Criteria panel of the Competitor Search screen. The `getOrganizationList` API is called to retrieve the list of enterprises. When performing a price match, the user can verify if the competitor already exists in the database. Competitors can be defined for each enterprise.

The `getCompetitorList` API is called to retrieve a list of existing competitors.

If competitor records exist on an external database, the `getCompetitorListUE` user exit is called to retrieve a list of existing competitors.

Sterling Call Center and Sterling Store provides the following common codes for competitor statuses. The `getCommonCodeList` API is called to retrieve the descriptions for the competitor statuses.

- **Approved**—This status is used for approved competitors. Price matches can be performed for these competitors.
- **Pending**—This status is used for competitors for whom approval is pending. Price matches can be performed for these competitors only after they are approved.
- **Rejected**—This status is used for rejected competitors. Price matches cannot be performed for these competitors.

### Pagination

Sterling Call Center and Sterling Store supports the smart retrieval of record sets in the Competitor Search screen.

For more information about pagination, see [Section 4.22, "Pagination"](#).

#### 4.9.9.2 End-User Impact

None.

#### 4.9.9.3 Implementation

This section lists the configurations for the competitor search task:

- To retrieve competitor records from external systems, implement the `getCompetitorListUE` user exit.
- The Competitor Search task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### 4.9.9.4 Reference Implementation

Sterling Call Center and Sterling Store provides the necessary permissions to search for competitors to all the user groups as part of reference implementation.

## 4.9.10 Price Match Search

The call center and store representatives may have to search for existing price matches when performing a price match.

### 4.9.10.1 Solution

Sterling Call Center and Sterling Store provides a user interface which enables users to search and edit the existing price match records. If a user has access to multiple enterprises, the Enterprise drop-down list is displayed in the Price Match Search Criteria panel of the Price Match Search screen. The `getOrganizationList` API is called to retrieve the list of enterprises. During the price match process, a user may want to search for an existing price match. Based on the search criteria entered, a list of price matches is displayed. Based on the configuration, alternate item identifiers are used to display the items in the price match search results.

The `getPriceMatchList` API is called to retrieve a list of existing price matches.

If the price match records exist on an external database, the `getPriceMatchListUE` user exit is called to retrieve the list of existing price matches.

---

---

**Note:** If the price match records exist on an external database, Sterling Commerce assumes that the competitor records also exist on an external database and the `getCompetitorListUE` user exit is called to retrieve a list of existing competitors.

---

---

Sterling Call Center and Sterling Store provides the following common codes for price match statuses. The `getCommonCodeList` API is called to retrieve the descriptions for the price match statuses.

- **Approved**—This status is used for approved price matches. These price matches can be applied to order lines.
- **Pending**—This status is used for price matches for which approval is pending. These price matches should be approved before they can be applied to order lines.
- **Rejected**—This status is used for rejected price matches. These price matches cannot be applied to order lines.

### Pagination

Sterling Call Center and Sterling Store supports the smart retrieval of record sets in the Price Match Search screen.

For more information about pagination, see [Section 4.22, "Pagination"](#).

#### 4.9.10.2 End-User Impact

None.

#### 4.9.10.3 Implementation

This section lists the configurations for the price match search task.

- Implement the getPriceMatchListUE user exit to retrieve price match records from external systems.
- This price match search task is permission controlled. Sterling Call Center and Sterling Store allows you to assign the necessary permissions to user groups for this task.

#### 4.9.10.4 Reference Implementation

Sterling Call Center and Sterling Store provides the necessary permissions to search for price matches to all the user groups as part of reference implementation.

### 4.9.11 Create or Modify a Competitor Record

If a competitor record does not already exist in the database, the call center and store representatives may need to create a new competitor, when performing a price match. CSR leads and Store leads may want to modify an existing competitor record, for example, a CSR lead may want to change the status of a competitor from Pending to Approved.

#### 4.9.11.1 Solution

Sterling Call Center and Sterling Store provides a user interface task to create a new competitor if that competitor does not already exist in the database. It also allows you to modify an existing competitor record. Users who have access to multiple enterprises can select the enterprise for which a competitor record has to be created.

The getCompetitorDetails API is called to retrieve details about a competitor. The manageCompetitor API is called to create a new competitor or update an existing competitor.

If the competitor records exist on an external database, the getCompetitorDetailsUE user exit is called to retrieve the details of existing competitors. The manageCompetitorUE user exit is called to create a new competitor or update an existing competitor.

Sterling Call Center and Sterling Store provides the following common codes for competitor statuses. The getCommonCodeList API is called to retrieve the descriptions for the competitor statuses.

- Approved
- Pending
- Rejected

#### **4.9.11.2 End-User Impact**

None.

#### **4.9.11.3 Implementation**

The create or modify competitor record task is permission-controlled. Sterling Call Center and Sterling Store allows you to assign the necessary permissions to user groups for this task.

When creating a new competitor, the value of the price match percentage for the competitor will be defaulted to the value of the Price Match Percentage Applied rule defined in the Channel Applications Manager. For more information about configuring the Price Match Percentage Applied rule, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### **4.9.11.4 Reference Implementation**

Sterling Call Center and Sterling Store provides the necessary permissions to create or modify a competitor record to the CSR lead user group and store managers as part of reference implementation.

### **4.9.12 Create or Modify a Price Match Record**

If a price match does not exist for a selected item and competitor, the call center and store representatives may need to create a new price

match record. CSR leads and Store leads may want to modify an existing price match record, for example, a CSR lead may want to change the status of a price match from Pending to Approved.

### **4.9.12.1 Solution**

Sterling Call Center and Sterling Store provides a user interface task to create a new price match, if the price match does not already exist in the database. It also allows users to modify the existing price match records. Users who have access to multiple enterprises can select the enterprise for which a price match record has to be created.

This section explains the process of creating and modifying price match records and the solution offered by Sterling Call Center and Sterling Store.

### **Adding Items To A Price Match**

When selecting items for price match, a user can search for items based on Item ID or Alternate Item ID. The `getCompleteItemList` API is called to retrieve details of the item.

### **Retrieving Details Of The Price Match**

The `getPriceMatchDetails` API is called to display the details of an existing price match. The `getPriceMatchDetailsUE` user exit is invoked if the price match records exist on an external database. The `getOrderLinesForPriceMatch` API is called to retrieve the order lines related to the price match.

### **Creating A Price Match**

The `managePriceMatch` API is called to create a price match record. The `managePriceMatchUE` user exit is invoked if the price match records exist on an external database.

### **Modifying A Price Match**

The `managePriceMatch` API is called to modify a price match. The `managePriceMatchUE` user exit is invoked if the price match records exist on an external database. When a price match is approved, the `changeOrder` API is called to apply the price match to the corresponding order lines. The Price Match Recorded As enterprise rule is called to determine the way in which to apply the price match. The user can

record price match as a charge in the unit price, or apply with the charge category as discount. The charge name is according to the Price Match Charge Name rule.

Sterling Call Center and Sterling Store provides the following common codes for price match statuses. The `getCommonCodeList` API is called to retrieve the following descriptions for the price match statuses:

- Approved
- Pending
- Rejected

#### **4.9.12.2 End-User Impact**

None.

#### **4.9.12.3 Implementation**

The create or modify a price match record task is permission controlled. Sterling Call Center and Sterling Store allows you to assign the necessary permissions to user groups for this task.

#### **4.9.12.4 Reference Implementation**

Sterling Call Center and Sterling Store provides the necessary permissions to create or modify a price match record to the CSR lead user group and store managers as part of reference implementation.

### **4.9.13 Reship an Item**

A customer may report that the item ordered is either not received or damaged in transit. In such situations, the item may need to be reshipped or the customer has to be refunded for the shipment.

#### **4.9.13.1 Solution**

A user would typically reship a shipment when the system indicates that an item has been shipped but the customer claims of not having received it. This section explains the complete Reship process.

#### **Tracking An Item**

Reship of an item is possible if it is in the `Shipped` status. If the item is in `Delivered` status, it cannot be reshipped. This is because an order line

can be reshipped only if a shipment has been created for it; when a delivery line is delivered without a shipment, it cannot logically be reshipped. Therefore, the user must first track the item and investigate the current status of the item to determine whether reship is possible or not.

### Configuring Reasons For Reship

The reasons for reshipping an item are configured using RESHIP\_REASON common code. You can configure reship reasons such as missing items, missing container, wrong item and so forth to allow performing reship action.

### Validating that an Item can be Reshipped

Before reshipping an item, you must ensure that the following conditions are met:

- The order line must have enough available quantity that has not already been reshipped.
- The reship of an item is allowed only after certain number of elapsed days. This is to ensure that sufficient time is allowed for the item to reach the customer. For example, we can configure the MINIMUM\_RESHIP\_WINDOW as 5 days to ensure that a reship is possible only after 5 days after placing an order.

The validity of the item to be reshipped is checked using the validateReship API. The validateReship API validates that the order lines passed as input are valid for being reshipped. You may implement other validations using the validateReshipUE user exit, if applicable.

If the item is not valid for reshipping, but needs to be reshipped, the user can override the reship validation, if the users have permissions to do so, or create a follow-up alert for their supervisor.

### Handling Bundle Items

You can reship bundle parent items and bundle component items. Reshipping a Bundle parent item will result in reship of all component items. For more information about bundles, see [Section 4.2, "Configured Items"](#).



## Reshipping and Refunding an Item

After validating, the user can either reship or refund the items. The reship or refund options are provided based on the availability of the items.

If the customer chooses the reship option, the reshipOrderLines API is called which creates new order lines to fulfill reshipment of order lines passed in the input. Attributes on the new order lines created by the reshipOrderLines API can be controlled by the ReshipOrderLines document template for the order's document type, as well as by the YFSreshipOrderLinesUE user exit. Additional validations may be done by this user exit. This user exit may cause the API to return different reship message.

Sometimes, the ordered item may be out of stock or the customer might not want the item. In such situations, the user would have to refund the items. If refund is involved, a return is created for the order line with line type as 'Credit'.

The user can enter notes when reshipping an item. The changeOrder API is called to save the notes in the order.

---

**Note:** The order lines are not reserved when you reship an item.

---

### 4.9.13.2 End-User Impact

None.

### 4.9.13.3 Implementation

This section explains how to configure the reasons for reshipping an item and how to configure the minimum reship window days.

1. Log in to the Channel Applications Manager. For more information about logging in to the application, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
2. From the Channel Applications Manager, select Configure Order Administration > Reship Reasons.
3. Enter appropriate reasons for reshipping the item.

4. From the Channel Applications Manager, select Configure Order Administration > Advanced Configurations > Configure Transaction Specific Rules.
5. In the Reship Order panel, enter the appropriate number of days in the Minimum Reship Window field.

Notes entered for this task are saved on the order with YCD\_RESHIP\_INFO note type.

### 4.9.13.4 Reference Implementation

This section explains the reference implementation configured to allow reship of an item for all organizations.

1. The minimum reship window period is set to five days which specifies the minimum number of days to be elapsed after which reship is allowed.
2. The following reasons are configured in the RESHIP\_REASON common code:
  - Missing Item
  - Missing Container
  - Wrong Item
  - Cargo Loss
  - Damaged or Defective Item

## 4.9.14 Add or Modify Charges

Call center and store representatives may have to add a new charge or modify an existing charge for an order that has been placed. For example, if a customer has been given two appeasements for the same issue, this task can be used to recover the excess refunded amount.

### 4.9.14.1 Solution

Sterling Call Center and Sterling Store provides a user interface that allows users to add or modify charges at both the header level and the line level.

The getChargeCategoryList API is called to retrieve the list of charge categories. The getChargeNameList API is called to retrieve the list of

charge names. The user can add or modify charges at the header level and line level, based on permissions.

The `getCompleteOrderDetails` API is called to display the list of charges and taxes that are already applied on the order. The `changeOrder` API is called to apply the charge modifications. However, if the charges that are applied on the order are based on a pricing rule, users cannot modify the charges.

### Viewing and Confirming Payment Details

The Payment Confirmation screen displays the summary of the payment impact on the order as a result of adding or modifying the charges. If payment is configured to be handled outside Sterling Call Center and Sterling Store, the Payment Confirmation screen is not displayed. For more information about confirming the payment details, see [Section 4.9.3, "Change Payment Method"](#).

#### 4.9.14.2 End-User Impact

None.

#### 4.9.14.3 Implementation

This section lists the configurations for this task.

- Sterling Call Center and Sterling Store allows you to configure the order statuses for which the addition of a new charge or the modification of an existing charge is allowed. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to define charge names, charge categories, and tax names for an enterprise. For more details about charge names, charge categories, and tax names, refer to the *Sterling Distributed Order Management: Configuration Guide*.
- If you do not use the reference implementation provided by Sterling Call Center and Sterling Store, you can implement the `YFSOrderRepricingUE` user exit to handle repricing during order modification. This implementation should take into consideration the changes in charges and taxes. For example, you may have to extend the Add Modify Charges user interface to pass additional information to the `changeOrder` API in order to indicate that the charges and taxes were overridden through the UI. For more information about

the YFSOrderRepricingUE user exit and the changeOrder API, refer to the *Sterling Call Center and Sterling Store: Javadocs*.

- The add or modify charges task is permission controlled. Sterling Call Center and Sterling Store allows you to assign the necessary permissions to user groups for this task.
- Notes entered for this task are saved in the order with the YCD\_ADD\_MODIFY\_CHARGES note type.

#### 4.9.14.4 Reference Implementation

This section explains the reference implementation provided as part of reference implementation for the add or modify charges task.

- The following charge categories are provided as part of the reference implementation:
  - Customer Appeasement
  - Discount
  - Handling
  - Price Match
  - Shipping
  - Personalization
- The following charge names are provided as part of the reference implementation:
  - Customer Appeasement
  - Discount
  - Handling
  - Price Match
  - Shipping
  - Personalization
- Sterling Call Center and Sterling Store provides the necessary permissions to add or modify charges to the CSR lead user group and store managers as part of reference implementation.

## 4.9.15 Add Coupon

After placing an order, a customer may want to add a coupon or promotion code to the order.

### 4.9.15.1 Solution

The Sterling Multi-Channel Selling Solution integration provides a user interface task that enables a user to add a coupon or a promotion code to an order. The YCDPromotionValidationUE user exit is used to verify whether a coupon or a promotion code is valid for a customer or not. If the YCDPromotionValidationUE user exit is not implemented, the validateCoupon API is called to validate the coupon or promotion code.

### Applying the Promotion to the Order

If the YCDPromotionValidationUE user exit or the validateCoupon API confirms that a coupon or a promotion code is valid, the changeOrder API is called to apply the new coupon or promotion code to the order. The repricing logic is executed whenever the Reprice Order flag is set to "Y" for the Change Promotion modification type.

### 4.9.15.2 End-User Impact

None.

### 4.9.15.3 Implementation

This section explains the configuration settings for adding a coupon or a promotion code to an order.

1. Log in to the Channel Applications Manager. For more information about logging in to the Channel Applications Manager, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
2. In the Applications Manager, set the StatusModificationTypeKey to "PROMO\_AWARD", by performing the following steps:
  - a. From the Applications Manager menu bar, select Distributed Order Management > Document Specific > Sales Order > Fulfillment > Order Modification > Modifications Impacting Pricing.
  - b. Click Details. In the Modification Type List screen, add the Change Promotion and Remove Promotion modification types to the list of modifications that impact pricing.

Notes entered for this task are saved in the order using the YCD\_NEW\_PROMOTION note type.

### 4.9.15.4 Reference Implementation

This section explains the reference implementation provided for the Add Coupon task.

- As part of reference implementation, Sterling Call Center and Sterling Store provides coupons. For more information about coupons provided as part of reference implementation, see the *Sterling Selling and Fulfillment Suite: Applications Reference Implementation Guide*.
- All rules for adding a coupon or promotion code are configured for all the pricing organizations.

## 4.9.16 Cancel Order

A customer may want to cancel an entire order, or just certain product or service items of the order.

### 4.9.16.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which enables users to cancel the entire order or only certain product or service items of the order.

This section explains the Cancel Order process and the solution offered by Sterling Call Center and Sterling Store.

### Selecting The Cancel Quantity

The customers can choose to cancel the entire order, if all ordered items need to be cancelled, or cancel a part of the order if only some items of the order need to be cancelled.

### Cancelling Related Lines

When the user selects a parent item that has related product or service lines for cancellation, the related lines are automatically selected for cancellation.

## Canceling a Bundle Item

The user can only cancel a bundle parent item. When a bundle parent item is cancelled, all the components of the bundle are cancelled automatically.

## Configuring Reasons For Cancellation

The reasons for cancelling an item can be configured in the CANCEL\_REASON common code. You can configure reasons such as change of mind, cheaper price found, and so forth to allow the Cancel action.

## Canceling Pick up Lines After Release

If the Allow Cancellation of Pickup Lines after Release rule is enabled, the user can cancel the order lines even after the order lines with the pickup fulfillment method are moved to the Released status. This is achieved by calling the following APIs:

- getCompleteOrderDetails API—This API retrieves the sales order details.
- changeOrder API—This API cancels the lines on the order and release, if associated with the order line.
- changeShipment API—This API cancels the shipment lines associated with the order line.

## Canceling Delivery Lines After Release

If the Allow Cancellation of Delivery Lines after Release rule is enabled, the user can cancel the order lines that have the Delivery fulfillment method and are moved to the Released status, but are not shipped. This is achieved by calling the following APIs:

- getCompleteOrderDetails API—This API retrieves the sales order details.
- changeOrder API—This API cancels the lines on the order and release associated with the order line.
- changeShipment API—This API cancels the shipment lines associated with the order line.

In order to cancel delivery lines after release, shipments must be created for product lines with fulfillment method as delivery.

## Requesting For A Stop Delivery After Lines Have Been Shipped

The Cancel Order screen in Sterling Call Center and Sterling Store can be configured to allow users to request a stop delivery for order lines in Shipped status.

If an order line has already been shipped, you can either enable the lines for requesting a stop delivery or disable the lines so they cannot be cancelled if you do not support a stop delivery process.

This can be achieved by configuring the order status modification rules appropriately. If you wish to allow a stop delivery to be requested on shipped lines, configure the stop delivery modification type to be allowed for the Shipped status. If you do not support stop delivery requests, you should disable this modification type.

Order lines can either be cancelled or requested for stop delivery (not both). Therefore, you must configure the status modification rules for cancellation and stop delivery to be mutually exclusive; any status that is allowed for cancellation cannot be allowed for stop delivery and vice-versa. For more information about configuring status modification rules, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

This is achieved by calling the following APIs:

- `getCompleteOrderDetails` API—This API returns the quantity which is available for cancellation and stop delivery.
- `manageStopDeliveryRequest` API—This API manages the request for the stop delivery.

The `createException` API is used to create an alert of type `YCD_STOP_DELIVERY` whenever a stop delivery request is placed. The alert is assigned to the `YCD_STOP_DELIVERY` queue. For more information about creation of alerts, see [Section 4.23, "Alert and Queue Management"](#).

In order for stop delivery requests to work, shipments must be created for product lines with fulfillment method as delivery.



#### 4.9.16.2 End-User Impact

This section explains the end-user impact for the cancel order task.

- When the user selects a line for cancellation that has related lines, the related lines are automatically selected for cancellation.
- Based on the rules configured for cancellation of released lines, the end users can cancel delivery and pickup order lines that are above the Released status.
- Based on the status modification rules for an order, the end users can place stop delivery requests for shipped lines.
- The Payment Confirmation screen will display the list of all stop delivery requests placed on the order. Since order lines requested for stop delivery are not yet cancelled from the order, the user will not see any change in the order total on the Payment Confirmation screen. If these order lines are successfully stopped from being delivered to the customer, the order total will be adjusted at that time. Refunds for the stop delivery will only be made after the item is received.

#### 4.9.16.3 Implementation

This section explains the configurations for the cancel order task.

- To cancel delivery lines after they have been released or to allow requests for stop delivery after lines have been shipped, ensure that that Sales Order Fulfillment pipeline is configured to create shipments for product lines with fulfillment method as delivery. For more information about the sales order fulfillment pipeline, refer to the *Sterling Distributed Order Management: Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to define reason codes that are available during the cancellation process. For more information about defining cancellation reasons, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the order statuses for which cancellation of order lines is allowed. If you allow cancellation of order lines for a particular status, ensure that you allow cancellation of work order lines for the corresponding status. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- When a stop delivery request is placed, the YCD\_StopDeliveryRequest\_Alert\_8.0 service generates an alert to notify that a stop delivery request is placed. This alert is sent to the Stop Delivery Request queue. When a return is received for the item for which stop delivery is requested, the YCD\_StopDeliveryRequest\_UpdateQty\_8.0 service marks the stop delivery request as successful.
- Notes entered for this task are saved on the order with YCD\_CANCEL\_INFO note type.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

### 4.9.16.4 Reference Implementation

This section explains the reference implementation provided as part of the Cancel Order task.

- The following reasons are configured in the "CANCEL\_REASON" common code:
  - Change of Mind
  - Cheaper Price Found
  - Late/Failed Delivery
  - Unacceptable Delivery Time Promised
  - Product No Longer Available
  - Fraudulent Order
  - Address Undeliverable
  - Duplicate Order
  - Other
- As part of the reference implementation provided by Sterling Call Center and Sterling Store, the Order Repricing user exit is enabled when cancelling quantity on the Order.
- As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to cancel an order to all user groups.

## 4.9.17 Track an Item

The customers may sometimes need to know the status of the orders. For example, if a customer does not receive items that were ordered, then the customer can call the CSR to enquire about the shipment status.

### 4.9.17.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which enables users to track orders.

The `getCompleteShipmentDetails` API is called to display the details of each order line such as status, work order-related information, shipments, containers. This API invokes the `YCDgetTrackingNumberURL` user exit to retrieve the tracking URL for a container. This user exit is implemented to retrieve the Tracking Number URL for the given Tracking Number, Carrier, Carrier Service, and `ScacAndService`. The URL is displayed on the Order Summary and Shipment Tracking screens. When the user clicks this hyperlink, the appropriate carrier's website is displayed where the user can view the container's tracking information. Users can also view the shipment details of the order by clicking the View Shipment hyperlink. The `getSortedShipmentDetails` API is called to retrieve the shipment details.

Sometimes, an enterprise may want to view the activities pertaining to a container, for example, a truck delivering goods to a certain region has broken down. If activities are specified for a container, the View Container Activities hyperlink is displayed in the Shipment Inquiry screen. On clicking this hyperlink, the Container Activities pop-up window is displayed. A user can view information such as the activity code, timestamp, additional information pertaining to the activity, and the user who created the activity.

If an order is on hold, the user can view and resolve holds. For more information about resolving holds, see [Section 4.9.2, "Remove Holds"](#).

Refer [Table 4–31](#) for different order statuses and the corresponding screen display.

**Table 4–31 Order Status and Screen Display**

Order Status	Screen Display
Scheduled/Released	<ul style="list-style-type: none"> <li>• If the fulfillment method is 'Shipping', the user can view the estimated ship date, estimated delivery date, item ID and description, and the item quantity.</li> <li>• If the fulfillment method is 'Pickup', the user can view the estimated pickup date, item ID and description, and the item quantity.</li> <li>• If the fulfillment method is 'Delivery' and if the order has associated work orders, the user can view the workorder information, otherwise the user can view the estimated ship date, estimated delivery date, item ID and description, and the item quantity.</li> </ul>
Created/Reserved/ Back Ordered	<ul style="list-style-type: none"> <li>• If the fulfillment method is 'Shipping', the user can view the estimated ship date, estimated delivery date, item ID and description, and the item quantity.</li> <li>• If the fulfillment method is 'Pickup', the user can view the estimated pickup date, item ID, description, and the item quantity.</li> <li>• If the fulfillment method is 'Delivery', the user can view the estimated delivery date, item ID and description, and the item quantity.</li> </ul>
Cancelled	If the fulfillment method is 'Shipping', 'Delivery', or 'Pickup', the user can view the cancel date and cancel quantity.

**Table 4–31 Order Status and Screen Display**

Order Status	Screen Display
Included in shipment/Shipped	<ul style="list-style-type: none"> <li>• If the items are Included in Shipment, the user can view the 'estimated to ship on' date, the estimated delivery date, and the item quantity. The user can view the shipped-on date, and the estimated delivery date. The View shipment, Reship shipment, and the reship container hyperlinks are available for a single order.</li> <li>• If the fulfillment method is 'Pickup', the shipment information is recorded. The user can view the 'picked up on' date, and the pickup quantity, and the store name.</li> </ul>
Work Order Created	<p>If an appointment is configured, the user can view the recent, open delivery appointment dates.</p> <p>The user can also view the item's identifier, quantity, and the unit price information.</p>
Work Order Completed	<p>The user can view the date on which the items were delivered to the customer, completed appointment date, completed delivery date and time, and the item quantity.</p> <p>The user can also view the item's identifier, quantity, and the unit price information.</p>
Work Order Cancelled	If a work order is in cancelled status, the user will not be able to view any information.
Stop Delivery Request Placed	If a stop delivery request is placed, the user will see an icon indicating the stop delivery request.

If an order has associated work orders, based on the Process Type Details configuration options, the user can view the shipment and container information.

#### **4.9.17.2 End-User Impact**

None.

### 4.9.17.3 Implementation

This section explains the configurations for the track an item task.

- Implement the YCDgetTrackingNumberURL user exit to retrieve the tracking URL for a container from an external system.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

### 4.9.17.4 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to track an item to all user groups.

## 4.9.18 Order Notes

A user may have to enter notes to an order or order line, describing the actions performed against that order or order line, for future reference.

### 4.9.18.1 Solution

Order notes can be either system-generated or entered by a user. Users can add notes at either the order header level or order line level. Additionally, a user can track important events and changes that have occurred on an order by viewing previously logged order notes.

Sterling Call Center and Sterling Store provides the following features:

- An Order Notes screen, where a user can view and add notes to an order.
- A Notes panel, where a user can add notes. The Notes panel is displayed under each order-related task and in the Order Notes screen.
- A mechanism to automatically log notes when important events occur in an order's life cycle.

### Order Notes Screen

Sterling Call Center and Sterling Store provides an Order Notes screen, which can be launched from the Order Summary screen and the View/Add Notes hyperlink in each order-related task.

Users can limit the notes that are displayed in the Order Notes screen based on the following criteria:

- Note Type
- Contact Method
- Priority
- System-Generated or Manually Entered
- Created on Order or Return Order

The `getCommonCodeList` API is called to retrieve the note types and contact methods for the order. The `getCompleteOrderDetails` API is called to retrieve the notes that match the filter criteria.

Users can also add notes through the Order Notes screen. For more information about adding notes, see ["Adding Notes Through the Order Notes Screen"](#).

The Order Notes screen can be accessed from an extension plug-in as a shared task. For more information about the Order Notes shared task, see [Section 3.4.2, "Order Notes Shared Task"](#).

### **Adding Notes Through the Order Notes Screen**

Sterling Call Center and Sterling Store provides a Notes panel in the Order Notes screen to enable users to add notes.

The `getCommonCodeList` API is called to retrieve the note types and the contact methods for the order. The `changeOrder` API is called to add notes to the order.

### **Adding Notes Through Order-Related Tasks**

Sterling Call Center and Sterling Store provides a Notes panel under each order-related task to enable users to add notes to the order while performing the corresponding task. Based on the related task, the corresponding notes are displayed by default in the Notes panel.

The `getCommonCodeList` API is called to retrieve the contact methods for the order. When the `changeOrder` API is called, the notes added to the order are saved along with a note type specific to the related task.

## System-Generated Notes

You can configure the system to log notes automatically when the following events occur:

- Chained Order Created—The system generates a note when a chained order is created.
- FTC Notification—The system generates a note if the FTC monitors make a change in an order or send a notification.
- Hold Created—The system generates a note when a hold is created.

**Note:** System-generated notes will not be logged if a hold, irrespective of whether it is added to an order or order line, is applied based on the hold type configuration. For example, if the Fraud Check hold type is configured to be automatically applied at the time of order creation, a note will not be logged when the order is created. A note will be logged only if the Fraud Check hold is applied by calling the changeOrder API.

- Hold Resolved—The system generates a note when a hold is resolved.
- Hold Rejected—The system generates a note when a hold is rejected.
- Back Ordered—The system generates a note if an order is back ordered while scheduling or releasing.
- On Shipment Confirmation—The system generates a note when a shipment is confirmed.
- Appointment Execution Failure—The system generates a note if an appointment fails.
- Appointment Execution Success—The system generates a note if an appointment is completed.
- Alert Created—The system generates a note when an alert is created for an order.

### 4.9.18.2 End-User Impact

The notes that have been configured to be logged automatically will be visible to the end users in the Order Notes screen.



### 4.9.18.3 Implementation

This section explains the configurations for the order notes task.

- Sterling Call Center and Sterling Store allows you to define note types and configure automatic note logging. For more information about defining note types and configuring note logging, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. This configuration does not provide an option to configure system-generated notes on chained order create event and FTC Notification event.

**Note:** You can use the sample XSL files listed in [Table 4–32](#) or you can use your own XSL files while selecting the XSL transformation file for an event in the Order Notes Wizard.

By default, the system-generated notes on chained order created event is enabled. To disable system generated notes on this event, do not invoke the YCD\_Log\_Note\_On\_Chained\_Order\_Created\_2.0 service.

System generated notes on FTC Notification events cannot be disabled.

- Sterling Call Center and Sterling Store provides sample XSL files to generate the note texts of the notes being logged for each event. [Table 4–32](#) lists the events and the sample XSL files used to log notes on the corresponding events.

**Table 4–32 Sample XSL Files**

Event	Sample XSL File
Chained Order Created	YCD_Log_Note_On_Chained_Order_Created_2.0.xsl.sample
Hold Created	YCD_Log_Note_On_Hold_Status_Change_8.0.xsl.sample
Hold Resolved	YCD_Log_Note_On_Hold_Status_Change_8.0.xsl.sample
Hold Rejected	YCD_Log_Note_On_Hold_Status_Change_8.0.xsl.sample

**Table 4–32 Sample XSL Files**

Event	Sample XSL File
Back Ordered	YCD_Log_Note_On_Back_Order_8.0.xsl.sample
On Shipment Confirmation	YCD_Log_Note_On_On_Shipment_Confirmation_8.0.xsl.sample
Appointment Execution Failure	YCD_Log_Note_On_On_Appointment_Failure_8.0.xsl.sample
Appointment Execution Success	YCD_Log_Note_On_On_Appointment_Completion_8.0.xsl.sample
Alert Created	YCD_Log_Note_On_Create_Exception_8.0.xsl.sample

- Sterling Call Center and Sterling Store logs notes with a unique note type for each task. The information regarding note types used in the Sterling Call Center and Sterling Store task flows are given in the corresponding task sections.
- Users can select a customer contact type while entering a note. For more information about configuring contact types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.9.18.4 Reference Implementation

This section explains the default Note Types and the Contact Types that are configured.

1. The Note Types are configured for system-generated notes and task-based notes.
  - The following are the Note Types configured for system-generated notes:
    - Chained Order Information
    - FTC Notification Information
    - Shipment Confirmation Information
    - Hold Information
    - Backorder Information
    - Delivery Appointment Change Information

- Create Alert Information
- The following are the Note Types configured for task-based notes:
  - Called Customer
  - Wrong Item Received
  - Customer Called
  - Schedule and Release Information
  - Issue Refund Now Information
  - Increase Line Quantity Information
  - Return Order Cancellation Information
  - Change Return Address
  - Change Return Method
  - Change Service Appointment
  - Called Issuing Bank
  - Marketing Opt-In
  - Cleared Held Order
  - Return Policy Overridden
  - Backorder Information
  - Hold Information
  - New Item Information
  - Order Entry
  - Fulfillment Options Change Information
  - Delivery Appointment Change Information
  - Delivery Instructions Change Information
  - Order Cancellation Information
  - Return Information
  - Order Address Change Information
  - Reship Information
  - Price Match Information

- Customer Appeasement Information
  - New Promotion Information
  - Payment Change Information
  - Customer Verification Information
  - Add Modify Charges
  - Undo-Pick Information
  - Backroom-Pick Shortage Information
  - Create Alert Information
  - Add Lines to Order Information
  - Tax Exempt Information
  - Quantity Validation Overridden
2. As part of the reference implementation, the following Contact Types are configured:
- Phone
  - E-mail

### 4.9.19 Change Fulfillment Options

When a customer requests a change in the fulfillment options for an item, the user can modify the fulfillment options as requested. For example, a customer may request the delivery of ordered items, but later decides to pick up the items from a store. In such situations, the user can modify the fulfillment options for the items or a provided service line.

---

---

**Note:** In the Change Fulfillment Options screen, all order lines that are associated with drop ship lines are disabled.

---

---

#### 4.9.19.1 Solution

This section explains how to modify the fulfillment options.

## Changing The Fulfillment Option For An Order Or An Order Line

The user can change the fulfillment options and the address for an order. The fulfillment options that are available are shipment, delivery, and pickup.

---

**Note:** When the order header contains a CustomerContactID, the manageCustomer API saves the addresses to the applicable customer contact.

---

If the rule to enable stores is configured, the items are available for pickup from a store. The options available for pickup are based on the option selected by the user. The getFulfillmentOptionsForLines API returns all fulfillment options that are applicable for the order line.

- If the selected option is "Ship", the items are shipped to the specified address.
- If the selected option is "Delivery", the items are delivered to the customer's doorstep.
- If the selected option is "Pickup", the user has a choice to choose to pick up from the current store or to select another store by entering the country and postal code details.

The getFulfillmentOptionsForLines API is used to determine the availability of the 'delivery address' and 'carrier service' that the user has supplied.

The generateWorkOrder API checks for order lines marked for delivery and determines if it has a delivery service associated with it, and if not, the generateWorkOrder API creates one and then creates a work order.

The YCDOVERRIDEDELIVERYMETHODUE user exit is used to override fulfillment methods. This user exit provides the ability to override the logic of enabling a fulfillment method. The default implementation is not provided and you can use your own implementation for this user exit.

If the UsedForOrdering attribute is set to "Y", the information about all the carrier services is displayed.

The order dates for each order line displays based on the fulfillment methods of the order line.

- Pickup—The value of the ReqShipDate attribute from the OrderLine element displays.
- Ship—The order dates display based on:
  - The date range of the CommittedDate attribute values within the OrderDates element displays as follows:
    - The minimum date is considered from the OrderDate element having DateTypeId attribute value as MIN\_DELIVERY.
    - The maximum date is considered from the OrderDate element having DateTypeId attribute value as MAX\_DELIVERY.
  - If the minimum date and maximum date for the CommittedDate attribute value are blank, the date range of the ExpectedDeliveryDate attribute value within Details element in the OrderStatus element displays.
- Delivery—The value of the PromisedApptStartDate attribute displays, if there is a work order associated with the orderline. Otherwise, the order dates do not display.

### Disabling Availability Checks

The user can disable availability checks when inventory is maintained externally as the user is aware of the inventory availability details.

### Disabling Fulfillment Methods

A user can select only the supported fulfillment methods for order lines. The fulfillment methods that are not supported are disabled. For instance, if an item does not support the "Pickup" fulfillment method, upon selecting the order line, the pick up option is disabled. Based on the configuration, Delivery and Shipping are displayed as separate fulfillment methods.

### Disabling Pickup From Store for All Items

You can configure the user interface to never show the pickup fulfillment method if the enterprise does not support that fulfillment method.

### Validating Fulfillment Options

A user may skip the customer identification details when creating an order and proceeding to the next screen. When the Change Fulfillment

Options screen is displayed, if the user moves to the next screen, the system validates that the fulfillment method is added to every order line. The user must enter the Ship To Address, if the fulfillment method is "Shipping", the Deliver To Address if the fulfillment method is "Delivery". If the fulfillment method is "Pickup", the Pick up Store must be entered.

### **Selection of Related Lines**

When the user wants to modify the fulfillment option for the parent item, the related lines associated with it are also selected for modification.

### **Selection of Bundle Lines**

When the user wants to modify the fulfillment option for the bundle item, if the bundle item is configured as ship independent, you can also select the components lines for modification. If the bundle item is configured as ship together, only the bundle parent item is displayed for modification and the fulfillment method of the component lines will be changed accordingly. For more information about bundles, see [Section 4.2, "Configured Items"](#).

### **Changing Service Instructions And Appointments**

A user can modify the service instructions. A user can also take appointments for deliveries for a customer by selecting the available time slot and date from the calendar. The generateWorkOrder API considers the new work orders and existing work orders, if any. If the work orders are in the Failed or Cancelled status, or if an appointment is not taken, the generateWorkOrder API returns a suggested appointment. For more information about changing service appointments, see [Section 4.9.23, "Change Service Appointment"](#).

### **Changing Pickup Lines**

A user can modify pickup lines even if they are in the Released status, and until the items are picked up and are not in the Shipped status. The user can change the ship node and fulfillment method for the pick up line. The changeRelease API is called to backorder an order line that is in the Released status.

### **Changing Delivery Lines**

A user can modify delivery lines even if they are in the Released status and are not in the Shipped status. The user can change the ship node

and the fulfillment method for a delivery line if the "Allow Modification of Delivery Lines after Release" option is enabled. The changeRelease API is called to backorder an order line that is in the Released status.

### Payment Confirmation Details

Due to the change in fulfillment options, if the payment information changes, the system displays the updated payment details. If the customer wants to make additional payments, the user must select the payment method.

### Customer Context Ordering

When the order is in the context of a customer, all addresses available to the customer/contact will be provided in the drop-down list in the Change Fulfillment Options screen. These addresses will also appear in the create new and edit address pop-up screens to serve as a starting point for a new address.

---

---

**Note:** When the order is in the context of a customer, all new addresses added to the order will automatically be saved against the customer record or contact record.

---

---

#### 4.9.19.2 End-User Impact

This section explains the impact on the user.

#### Disabling Availability Checks

The user is not shown the availability details in any store on the Add Multiple Lines, Order Entry, Change Fulfillment Options, Change Order Addresses, and Item Inquiry screens.

#### Changing Pickup Lines

The user is allowed to modify the order lines that have the pickup fulfillment method even after the status is changed to Released.

#### Changing Delivery Lines

The user is allowed to modify the order lines that have the delivery fulfillment method even after the status is changed to Released.



### 4.9.19.3 Implementation

This section details the rules to be configured to change the fulfillment options.

#### Disabling Availability Checks

If the Prevent Initial Availability Checks During Order Entry and Order Modifications rule is enabled, the inventory availability check is not performed and the calls to the `getFulfillmentOptionsForLines`, `findInventory`, or any derivation APIs are eliminated. This aids in better performance and pace when capturing the order.

#### Changing Pickup Lines

To modify the pickup lines that are in the `Released` status or above, check the Allow Modification of Pickup Dates and Store for Pickup Lines after Release box for the Enterprise.

#### Changing Delivery Lines

To modify the delivery lines that are in the `Released` status or later, check the Allow Modification of Fulfillment Options for Delivery Lines After Release box for the Enterprise.

Notes entered for this task are saved on the order with `YCD_DELIVERY_OPTIONS` note type.

### 4.9.19.4 Reference Implementation

This section explains the reference implementation provided for the following tasks:

#### Disabling Availability Checks

The Prevent Initial Availability Checks During Order Entry and Order Modifications rule is defaulted to ensure that availability checks are performed.

#### Changing Pickup Lines

The Allow Modification of Pickup Dates and Store for Pickup Lines after Release rule is defaulted to allow modification of lines after release.

### Changing Delivery Lines

The Allow Modification of Fulfillment Options for Delivery Lines after Release rule is defaulted to allow modification of lines after release.

The Allow Cancellation of Delivery Lines after Release rule is also defaulted to allow modification of lines after release.

## 4.9.20 Fulfillment Summary

Sometimes Sterling Call Center and Sterling Store users may have to view the fulfillment summary of an order to get a complete picture of the way in which the order is being fulfilled. In a B2B scenario, customers may place orders for large quantities of items, and this may result in the items being available on multiple dates. In such a situation, Sterling Call Center and Sterling Store users may require a better visibility into the way the orders are fulfilled.

### 4.9.20.1 Solution

The Fulfillment Summary screen provides information about the way in which an order is configured and fulfilled. The order lines are grouped based on the fulfillment method. Within each fulfillment method, the order lines can be grouped as follows:

- Shipment and Ship To Address (for Shipping fulfillment method)
- Ship Node (for Pickup fulfillment method)

The Fulfillment Summary screen provides visibility to order information, line items, quantity, price, fulfillment time, fulfillment date, available date, and delivery date. The Fulfillment Summary screen also provides the ability to:

- Set the order lines as gift items
- Select the replacement nodes for pickup
- Modify the fulfillment options of the order lines
- Cancel lines

### getOrderFulfillmentDetails API

This API calls the getCompleteOrderDetails API, getPossibleSchedules API, and the getCarrierServiceOptionsForOrdering API. It returns information such as availability of items, level of service, and so forth,

that are added to the order line. This information is used to display the Fulfillment Summary screen.

### **Viewing the Fulfillment Summary Screen**

Sterling Call Center and Sterling Store enables users to view the fulfillment summary in the following screens:

- Basic Fulfillment Summary Screen
- Advanced Fulfillment Summary Screen

The Basic Fulfillment Summary screen is recommended for enterprises fulfilling orders in business-to-customer (B2C) scenarios.

The Advanced Fulfillment Summary screen is recommended for enterprises fulfilling orders in business-to-business (B2B) scenarios.

The following features are available in the Basic Fulfillment Summary screen and the Advanced Fulfillment Summary screen:

- [Availability of Items](#)
- [Changing the Carrier Service for Shipping](#)
- [Changing the Pickup Date of the Item](#)
- [Changing Fulfillment Method](#)
- [Handling Related Items](#)
- [Handling Bundle Items](#)
- [Cancelling Items](#)
- [Reserving Items](#)
- [Marking Items as Gifts](#)
- [Changing the Pickup Location](#)
- [Overriding Availability](#)

In addition, the following features are available in the Advanced Fulfillment Summary screen:

- [Selecting the Optimization Type to be Used for the Order](#)
- [Changing Requested Delivery Date for Order Line](#)
- [Combining or Splitting an Order Line](#)

### Availability of Items

The item availability is displayed depending on the date. The `getPossibleSchedule` API returns the date, which determines whether or not the items are available.

### Changing the Carrier Service for Shipping

The `getCarrierServiceOptionsForOrdering` API provides a list of carrier service options and their estimated dates for items that have fulfillment method as "Shipping". By implementing the `GetCarrierServiceOptionsForOrderingUE` user exit, you can obtain the cost of each carrier service.

**Note:** Users can change the carrier service of an order line based on the order status modification rules.

### Changing the Pickup Date of the Item

The user can change the pickup date for an item by selecting the preferred date from the calendar. Only the dates on or after the `ProductShipDate` are enabled on this calendar.

### Changing Fulfillment Method

The fulfillment method can be changed for an item if the item supports the new fulfillment method. This is possible by choosing the action you want to change from "Shipping" to "Change to Pick up" or from "Change to Pickup" to "Shipping".

### Handling Related Items

The related items are selected automatically when the parent item and the related lines have the same fulfillment method.

### Handling Bundle Items

If the bundle item is configured as ship independent, gift options can be changed for all the component items. If the bundle item is configured as ship together, then if the gift option for bundle parent is changed, the gift option will be changed for all the components items. For more information about bundles, see [Section 4.2, "Configured Items"](#).

The user can only cancel a bundle parent item. When a bundle parent item is cancelled, all the components of the bundle are cancelled automatically.

### **Cancelling Items**

Items that are in a status that supports cancellation can be cancelled by selecting the Cancel action.

### **Reserving Items**

Items can be reserved upon confirmation or when navigating to the next screen.

### **Marking Items as Gifts**

The user can mark selected items as gift items by clicking the Gift Options button. Users can include gift messages for the gift items.

You can also configure the gift options to enable users to gift wrap an item marked as a gift. For more information about configuring the available gift options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### **Changing the Pickup Location**

The pickup location can be changed by clicking the Pick Locations button for the selected order lines. This button is displayed in the Pickup panel.

### **Overriding Availability**

The inventory availability can be overridden only in the Pickup panel, when logged in at the store from where the user wants to pickup inventory. Overriding of availability is possible only if the user logs in at the current store from where you perform pickup, and belongs to a user group that has permission to override inventory availability.

The order dates for each order line displays based on the fulfillment methods of the order line.

- Pickup—The value of the ReqShipDate attribute from the OrderLine element displays.
- Ship—The order dates display based on:
  - The date range of the CommittedDate attribute values within the OrderDates element displays as follows:

- The minimum date is considered from the OrderDate element having DateTypeId attribute value as MIN\_DELIVERY.
  - The maximum date is considered from the OrderDate element having DateTypeId attribute value as MAX\_DELIVERY.
  - If the minimum date and maximum date for the CommittedDate attribute value are blank, the date range of the ExpectedDeliveryDate attribute value within Details element in the OrderStatus element displays.
- Delivery—The value of the PromisedApptStartDate attribute displays, if there is a work order associated with the orderline. Otherwise, the order dates do not display.

### Selecting the Optimization Type to be Used for the Order

Users can select the optimization type to be used to fulfill the order based on the following options:

- Optimization of order scheduling based on the number of shipments, followed by the ship node priority, and the date. If the user selects this option, the items in the order will be shipped in as few shipments as possible. Therefore, even if some items are available earlier, they will not be shipped until all the items are available. This will reduce the shipping costs.
- Optimization of order scheduling based on the scheduling rule configured in the Channel Applications Manager. This option is displayed only if you configure the Scheduling Rule For Additional Shipping Option rule in the Channel Applications Manager. For more information, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

The description of this option is defined as a bundle entry in the `com.yantra.pca.ycd_bundle.properties` file, and the key for this bundle entry is the scheduling rule.

- Optimization of order scheduling based on the date, followed by the number of shipments, and the ship node priority. If the user selects this option, the items will be shipped as and when they become available. This may result in higher shipping costs.

---

---

**Note:** The shipping options will be displayed based on the status modification rules. If the status modification rules do not allow changing of shipping options, the shipping options will not be displayed in the Advanced Fulfillment Summary screen.

---

---

One of the options is selected by default based on the optimization type that is saved in the order. If the user changes the optimization type, the changeOrder API is called to save the updated information.

### Changing Requested Delivery Date for Order Line

Users can change the requested delivery date pertaining to an order line by clicking the Change Line Options button. The changeOrder API is called to save the new requested delivery date after the user clicks either Save and Refresh or Next.

---



---

**Note:** Users can change the requested delivery date pertaining to an order line based on the order status modification rules.

---

---

### Combining or Splitting an Order Line

If an order line is included in more than one shipment, users can combine the order line by clicking the  icon. When this icon is clicked, the FillQty attribute in the order line is set to the value of the OrderedQty attribute. Users can also split an order line into multiple shipments by clicking the  icon. When this icon is clicked, the FillQty attribute in the order line is set to a blank value. This feature enables a user to have better control on the way items are shipped at the line level. For example, if a customer wants the entire quantity of an item to be shipped together at the same time, the user can combine the order lines, and if the customer wants some quantity of an item to be shipped on a particular date and the remainder of the item on a different date, the user can split the order line. This feature also enables a user to use a scheduling rule defined by the enterprise to optimize order scheduling.

---

**Note:** The number of shipments displayed in the Advanced Fulfillment Summary screen is based on the attributes defined in the Release Consolidation template in the Applications Manager.

---

#### 4.9.20.2 End-User Impact

This section explains the impact on the user.

##### Alternate Views of the Fulfillment Summary Screen

Based on how you configure the order entry rules, end users will see either the Basic Fulfillment Summary screen or the Advanced Fulfillment Summary screen in the order creation task flow.

##### Handling Related Items

When the user selects the parent item, the related lines having the same fulfillment method as the parent item are also selected.

##### Handling Bundle Items

When the user wants to modify the fulfillment option for the bundle item, if the bundle item is configured as ship independent, the user can also select the components lines for modification. If the bundle item is configured as ship together, the user can only select the bundle parent item for modification and the fulfillment method of the component lines will be changed automatically. For more information about bundles, see [Section 4.2, "Configured Items"](#).

#### 4.9.20.3 Implementation

This section explains the configurations for viewing the fulfillment summary:

- Sterling Call Center and Sterling Store enables you to configure the Fulfillment Summary screen to be displayed in the order creation task flow. Sterling Call Center and Sterling Store also enables you to configure a default scheduling rule to be used to display an additional shipping option in the Advanced Fulfillment Summary screen. For more information, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.



- You can enable users to record the committed dates when the user selects the level of service. For more information, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.9.20.4 Reference Implementation

This section explains the reference implementation provided for viewing the fulfillment summary.

- The "Compute Expected Dates When Requested Dates On The Pickup Order Lines Are Changed" rule is set to "Y" for all the organizations.
- By default, the Basic Fulfillment Summary screen is displayed in the order creation task flow for all the organizations.
- CORP-SCH and SYSTEM scheduling rules are provided for all the organizations.
- The OptimizationType attribute for the CORP-SCH scheduling rule is set to "03".
- By default, permission is provided for modifying the level of service and the requested delivery date of an order line.

### 4.9.21 Delivery and Service Grouping

When a complex order pertaining to numerous products, provided services, and delivery services exists, the task of scheduling appointments for these orders in the appointment scheduling screens of the Sterling Call Center and Sterling Store application becomes difficult. Users can group the order lines into smaller groups to ensure that appointments are available for the various services.

#### 4.9.21.1 Solution

The Delivery and Service Grouping screen provides a summary of all the work orders and appointments related to an order. This screen enables users to select products and services, and assign them to a group before choosing an appointment. Users can create new groups, merge multiple groups, and add or remove lines from the existing groups. Sterling Call Center and Sterling Store enables you to assign permissions to users to access the Delivery and Service Grouping task on the order creation and order modification tasks. Users with permissions can access the Delivery and Service Grouping task from the Change Fulfillment Options, Service Fulfillment Summary, View/Change Service Instructions, and Change

Service Appointment screens. The Delivery and Service Grouping task does not consider the inventory of product items or capacity of service items when grouping the order lines.

The getOrderFulfillmentDetails API is called to retrieve the list of work orders sorted based on the ShipToKey attribute. This API does not provide any out-of-the-box sequencing logic on work orders. However, you can use the post-command of this API for your custom sequencing logic. The manageServiceGrouping API is a wrapper API that internally calls the changeOrder API, modifyWorkOrder API, and the createWorkOrder API. The changeOrder API is called to create new associations between the product and delivery service lines, create additional delivery service line, and remove the old associations between the product and delivery service lines. The modifyWorkOrder API is called to cancel or remove an orderline from an existing work order, and to associate a product or provide service line to an existing work order. The createWorkOrder API is called to create the order.

---

---

**Note:** If the Delivery Allowed flag for an item, say, Item A, is set to Only If Something Else Is Being Delivered, and if this item is part of an order that contains another item, say, Item B, for which the Delivery Allowed flag is set to Yes, users cannot create different groups for these two items. If a user tries to create different groups, an error is displayed, indicating that some of the order lines (in this context, Item A) are not grouped with a valid delivery line.

---

---

### 4.9.21.2 End-User Impact

This section explains the end-user impact for the Delivery and Service Grouping task:

- The user will see the Perform Advanced Service Grouping check box in the Change Fulfillment Options screen based on user permissions.
- The user will see the Advanced Service Grouping button in the Service Fulfillment Summary, View/Change Service Instructions, and Change Service Appointment screens based on user permissions.

### 4.9.21.3 Implementation

The Delivery and Service Grouping task is permission controlled. Sterling Call Center and Sterling Store allows you to assign the necessary

permissions to user groups for this task. For more information about administering user group permissions, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

#### **4.9.21.4 Reference Implementation**

As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to all users to perform the Delivery and Service Grouping task.

### **4.9.22 Change Service Instructions**

Customers may request to change instructions for orders that are associated with delivery lines or service lines.

#### **4.9.22.1 Solution**

Sterling Call Center and Sterling Store provides a user interface task which enables users to change service instructions.

At the time of delivery, if a customer requests to change the service instruction for a product or service item, the user can modify the service instructions as requested. For each service line, the user can modify the service instructions in the text box that has been provided. Sterling Call Center and Sterling Store allows the users to modify instructions for the deliveries or services that are not complete. For more information about viewing service fulfillment summary, see [Section 4.9.24, "Service Fulfillment Summary"](#).

#### **4.9.22.2 End-User Impact**

None.

#### **4.9.22.3 Implementation**

This section explains the configurations for the change service instructions task.

- Notes entered for this task are saved on the order with YCD\_DELIVERY\_INSTR note type.
- Sterling Call Center and Sterling Store allows you to configure the order statuses for which changing service instructions is allowed. For more information about configuring the order statuses for

modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### 4.9.22.4 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to change service instructions to an order to all user groups.

### 4.9.23 Change Service Appointment

When a customer places an order for provided services or delivery services, an appointment needs to be taken with the corresponding retailer to agree on a time frame for the services to be fulfilled. Additionally, customers may need to provide the retailer with specific instructions to ensure that the service fulfillment is successful. For example, the customer may request the retailer to deliver an item at the back door.

After placing an order, the customers may request to make changes to an appointment or service instructions.

#### 4.9.23.1 Solution

Sterling Call Center and Sterling Store provides configurable appointment screens, which include Service Fulfillment Summary screen and Change Service Appointment screen that allow users to create or modify service appointments and enter service instructions.

Sterling Call Center and Sterling Store does not support scheduling multiple appointments for a delivery or service line, while creating an order. The user can only take one appointment per line at a time.

If the customer wants more appointments to be taken, perform one of the following:

- The existing appointment must to be marked completed/failed and a new appointment must be taken.
- A new line must be added and a new appointment must be taken for the newly added line.

## Service Fulfillment Summary Screen

The Service Fulfillment Summary screen provides visibility to service appointments pertaining to an order. For more information about service fulfillment summary, see [Section 4.9.24, "Service Fulfillment Summary"](#).

## Change Service Appointment Screen

A calendar is displayed from which the user can select the appropriate appointment for the customer. When the customer chooses the time slot for delivery, the calendar displays information such as working or non-working days, capacity available days, and so forth. Sterling Call Center and Sterling Store provides multiple ways to take appointments. The user can take appointments by selecting the time slot and viewing the days available for the selected time slot. Alternatively, the user can select a day in the calendar and view the slots available on that day.

You can configure the number of weeks to display on the calendar at a time. By default, a complete month is displayed. To improve the performance, reduce the number of weeks to display to two so that only two weeks of data needs to be returned by the `getWorkOrderAppointmentOptions` API.

Users can look beyond the initial weeks displayed on the calendar in case there are no acceptable appointments within the initial time frame. By default, the appointment calendar allows the user to look out infinitely into the future for appointments. However, your capacity system may not have visibility beyond a certain time frame. You can configure the maximum number of days to allow users to look into the future. After the maximum number of days is reached in the appointment screen, the user will not be able to proceed to the next time period. When an appointment is saved, the `reserveOrder` API is called. This API calls the `changeOrder` API to make a reservation.

If the reservation succeeds, the `modifyWorkOrder` API is called to save the appointment on the work order. If the reservation fails due to that same inventory being consumed by another order, the Service Appointment screen refreshes and requests the user to take a new appointment based on the new inventory picture.

## Modifying Delivery Lines After Release

If the Allow Modification of Delivery Lines After Release rule is enabled, the user can change the service appointments of order lines that have the "Delivery" fulfillment method and are not in the `Shipped` status.

## Handling Bundle Items

If the bundle item is configured as ship independent, the user can change the service appointment for all the component items. If the bundle item is configured as ship together, the user can only change the service appointment for the bundle parent item. For more information about bundles, see [Section 4.2, "Configured Items"](#).

### 4.9.23.2 End-User Impact

The configurable rules for the Change Service Appointment screen determines what information to display. As explained in the [Change Service Appointment Screen](#) section, you can configure the default calendar view and the number of weeks to display.

### 4.9.23.3 Implementation

This section explains configurations for the change service appointments task.

- To change the service appointment calendar view, configure the following rules:
  - Capacity Information
  - Calendar Display
  - Default Appointment Calendar View
  - Defaulting Option

For more information about configuring the service appointment calendar view, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- Notes entered for this task are saved on the order with YCD\_DELIVERY\_APPT note type.
- Sterling Call Center and Sterling Store allows you to configure the order statuses for which changing service appointments is allowed. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the order statuses for which cancellation or modification of fulfillment options for delivery and pickup order lines that are beyond the

released status is allowed. For more information about configuring the special order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### 4.9.23.4 Reference Implementation

The instruction type provided and the note reason configured for changing a delivery appointment are:

- An instruction type called 'DELIVERY' is provided for saving service appointment instructions.
- The Calendar Display is set to "Monthly View"
- The Default Appointment Calendar View is set to "Select Slot First, Then Choose the Day".
- As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to change service appointments on an order to all user groups.

### 4.9.24 Service Fulfillment Summary

The Service Fulfillment Summary screen provides visibility to service appointments pertaining to an order.

#### 4.9.24.1 Solution

This section explains the functionality of the Service Fulfillment Summary screen.

##### **getOrderFulfillmentDetails API**

This API calls the `getCompleteOrderDetails`, `getPossibleSchedules`, and `getCarrierServiceOptionsForOrdering` APIs. It returns information pertaining to the work orders associated with delivery and service lines. This information is used to display the Delivery Fulfillment Summary screen.

This screen also provides the ability to:

- Mark delivery lines as gift items
- Change service instructions for work orders

### Marking Delivery Lines As Gift Items

The Sterling Call Center and Sterling Store allows you to select items being delivered as gifts based on status modification rules. You can configure the Sterling Call Center and Sterling Store application to enable users to mark either the entire work order or individual work order lines as gifts. For more information about configuring the gift options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

Users can include a gift message for an item marked as a gift. Additionally, you can also configure the gift options to enable users to gift wrap an item marked as a gift.

The getOrderFulfillmentDetails API is called to retrieve the details of the order lines marked for a gift recipient. The changeOrder API is called to modify the gift options for the lines.

### Change Service Instructions For The Work Order

The Sterling Call Center and Sterling Store allows you to change service instructions if the work order status modification rules permit. For more information about changing service instructions, see [Section 4.9.22, "Change Service Instructions"](#).

The getOrderFulfillmentDetails API is called to retrieve details of the work orders. The modifyWorkOrder API is called to modify the service instructions on the work order.

### Handling Bundle Items

If the bundle item is configured as ship independent, the user can change the service instructions for all the component items. If the bundle item is configured as ship together, the user can only change the service instructions for the bundle parent item. For more information about bundles, see [Section 4.2, "Configured Items"](#).

#### 4.9.24.2 End-User Impact

None.



### 4.9.24.3 Implementation

This section explains the configurations for the service fulfillment summary screen.

- Sterling Call Center and Sterling Store allows you to configure the order statuses for which changing gift options is allowed. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the order statuses for which changing service instructions is allowed. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the available gift options. For more information about configuring the available gift options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 4.9.24.4 Reference Implementation

None.

## 4.9.25 Reservations

The reservation of items is performed to confirm availability and resolve issues when handling multiple customers simultaneously. For example, if two CSRs are assisting two different customers for the same item, which is the only available item, and if the item is not reserved, then both CSRs might sell the item by checking the availability status. In such situations, reservations are essential.

### 4.9.25.1 Solution

This section describes the functionality of reservations.

#### Creating Reservations

Reservations can be created only if the Reserve Items During Order Entry and Order Modifications rule is enabled.

To reserve items on the Fulfillment Summary screen for order lines with fulfillment method as "Pickup" or "Ship", the reservation is created by calling the changeOrder API.

To reserve items on the Delivery Appointment screen for order lines with fulfillment method as "Delivery", the reservation is created by calling the reserveOrder API.

### Handling Bundle Items

While reserving a bundle item, if the reservation of a component item fails, the bundle parent item will not be reserved. For more information about bundles, see [Section 4.2, "Configured Items"](#).

#### 4.9.25.2 Implementation

This section explains the rules to configure item reservations.

To reserve items:

1. Log in to the Channel Applications Manager.  
For more information about logging in to the Channel Applications Manager, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
2. From the Channel Applications Manager, select Configure Order Administration > Advanced Configurations > Configure Availability Check and Reservation Options. The Configure Availability Check and Reservation Options screen is displayed.
3. Configure the Reserve Items During Order Entry and Order Modifications rule. The default value is "N".

---

---

**Note:** If the Reserve Items During Order Entry and Order Modifications rule is enabled, and in the Fulfillment Summary Screen, if the system fails to reserve some items, the user cannot go to the next screen.

---

---

#### 4.9.25.3 End-User Impact

None.

#### 4.9.25.4 Reference Implementation

By default, the Reserve Items During Order Entry and Order Modifications rule is set to "N" for all the organizations.

### 4.9.26 Increase Order Line Quantity

When a customer wants to increase the quantity of one or more items on an order that is already placed, the call center or store representatives can increase the order line quantity as requested. Line quantity for provided services cannot be increased.

#### 4.9.26.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which enables users to increase the order line quantity.

This section explains Increase Order Line Quantity process and the solution offered by Sterling Call Center and Sterling Store.

#### Validating Whether The Order Line Quantity Can Be Increased

Sterling Call Center and Sterling Store validates if the order line quantity can be increased based on the order modification rules. The order modification rule verifies if the order status permits addition of lines to an order.

The `getCompleteOrderDetails` API is called to retrieve the details of the order. The `findInventory` API is called to retrieve the capacity information for service items. The `getFulfillmentOptionsForLines` API is called to retrieve the inventory information for product items.

The `getFulfillmentOptionsForLines` API is called with `ReqEndDate` as an input to verify whether the additional item quantity can be fulfilled along with the existing order quantity. The `ReqEndDate` is the farthest available `ExpectedShipmentDate` taken from the `Details` element under the order status of the order line. The `ShipNode` is passed from the `OrderStatus` element having the `Details` element from where the `ExpectedShipmentDate` is taken.

If the `getFulfillmentOptionsForLines` API returns the `HasAnyUnavailableQty` attribute value as "N", the adjustment can be fulfilled and the additional item quantity can be added to the order. If the

value of the `HasAnyUnavailableQty` attribute is set to "Y", the additional item quantity requested by the customer cannot be added to the order. In such situations, a warning icon is displayed on the screen along with the available item quantity that can be added to the order.

If the `getFulfillmentOptionsForLines` API returns a different delivery date for the additional item quantity that is not the same as the original delivery date, an appropriate message is displayed on the screen.

---

---

**Note:** If the service line is a related line and you increase the quantity of the parent item, based on the configuration, the quantity of the service line automatically increases. For more information about configuring the provided services, refer to the *Catalog Management: Configuration Guide*.

---

---

---

---

**Note:** When an order line is a part of a work order that has been partially delivered, users cannot increase the order line quantity.

---

---

### Handling Bundle Items

While increasing the line quantity of a bundle item, the quantity of the component items will automatically be increased. For more information about bundles, see [Section 4.2, "Configured Items"](#).

### Validating Whether The Items Are Present In The Store

The user needs to verify if the requested additional item quantity is present in the store. If the minimum order line status is in the `Released` status or exceeds this status, a warning message displays that prompts the user to manually check whether the items are present in the store.

### Viewing Service Fulfillment Summary

The user can view the service fulfillment summary and modify appointments for the items added. The items are reserved based on the configuration. For more information about service fulfillment summary, see [Section 4.9.24, "Service Fulfillment Summary"](#). For more information about service appointments, see [Section 4.9.23, "Change Service Appointment"](#).

## Viewing And Confirming Payment Details

Due to the increase in line quantity, the payment information changes, and the system displays the updated payment details. To make additional payments, the user must select the payment method.

Sterling Call Center and Sterling Store can be used to carry out the critical payment related processes during order management processing and enables you to integrate with external payment processing systems such as CyberSource or Chase Paymentech.

### 4.9.26.2 End-User Impact

None.

### 4.9.26.3 Implementation

This section explains the configurations for the increase order line quantity task.

- Sterling Call Center and Sterling Store allows you to configure the order statuses for which increasing order line quantity is allowed. For more information about configuring the order modification statuses, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

### 4.9.26.4 Reference Implementation

This section explains the reference implementation provided as part of the increase order line quantity task.

- As part of the reference implementation provided by Sterling Call Center and Sterling Store, order repricing logic is enabled when the line quantity is increased on the Order.
- Sterling Call Center and Sterling Store provides permissions to increase the order line quantity to all users as part of reference implementation.

## 4.9.27 Schedule and Release an Order

After an order is created, it has to be scheduled so that it can be delivered to the customer. Important shipment attributes, such as

delivery dates and ship nodes, are determined at this point. After Sterling Call Center and Sterling Store determines one or more ship nodes that can be utilized to ship the ordered merchandise, those nodes are notified with all the information that is necessary to send the shipment. In some cases, it may be necessary to execute the schedule and release process from Sterling Call Center and Sterling Store user interface.

### **4.9.27.1 Solution**

Sterling Call Center and Sterling Store provides a user interface that allows users to schedule and release an order. This task enables users to demonstrate the flow of an order that is created and fulfilled in Sterling Call Center and Sterling Store without having to go to the Selling and Fulfillment Foundation console.

The order is scheduled by calling the `scheduleOrder` API. It is released by calling the `releaseOrder` API. After the order is successfully scheduled and/or released, the details of all the order lines are displayed by calling the `getOrderLineStatusList` API and `getCompleteOrderDetails` API. If there is a problem with scheduling an order, the `getPossibleSchedules` API is called to display the schedule failure reasons.

Sterling Call Center and Sterling Store provides a list of common codes to display reasons for the failures that may occur during scheduling and releasing. These common codes are defined in the `YCD_YFS_BASE_COMMON_CODE` table and can be identified by the `YCD_SCH_FAILURE_RSN` code type.

### **4.9.27.2 End-User Impact**

None.

### **4.9.27.3 Implementation**

The schedule and release order task is permission controlled. Sterling Call Center and Sterling Store allows you to assign the necessary permissions to user groups for this task.

### **4.9.27.4 Reference Implementation**

Sterling Call Center and Sterling Store provides the necessary permissions to schedule and release an order to hub users as part of reference implementation.

## 4.9.28 View Procurement Orders

Call center and store representatives may need to view procurement and transfer orders that have been created for a sales order to get a complete picture of how the sales order is being fulfilled. If the ship node defined for an order line does not have enough inventory to fulfill the order, the required inventory can be acquired from a procurement node. In this scenario, the ship node defined for the order line receives inventory from the procurement node to fulfill the order.

### 4.9.28.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which allows users to view procurement orders (Purchase Orders and Transfer Orders) that are created for a sales order.

The getOrderLineList API is called to retrieve a list of procurement orders for the sales order. The getCompleteOrderDetails API is called to display the details of the procurement order.

### 4.9.28.2 End-User Impact

None.

### 4.9.28.3 Implementation

This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

### 4.9.28.4 Reference Implementation

Sterling Call Center and Sterling Store provides permissions to view procurement orders to hub users as part of reference implementation.

## 4.9.29 Change Order Address

A customer may want to change or modify the addresses on an order after placing the order. In such situations, call center and store representatives can modify the addresses as requested.

---

**Note:** When the order header contains a CustomerContactID, the manageCustomer API saves the addresses to the applicable customer contact.

---

### 4.9.29.1 Solution

This section explains the process of changing the order address and the solution offered by Sterling Call Center and Sterling Store.

#### Disabling Availability Checks

You can disable availability checks when inventory is maintained externally. If the Prevent Availability Checks During Order Entry and Order Modifications rule is enabled or set to "Y", the inventory availability check is not performed and the calls to the `getPossibleSchedules`, `findInventory`, or any derivation APIs are eliminated. This aids in better performance and pace in capturing the order.

#### Validating the Address Information

When the user has to enter an order address for a new customer, or modify an existing address, the validity of the address needs to be verified.

The validity of the address needs to be verified when the address of a new customer is to be entered or when the existing address of a customer needs to be modified. For more information about address verification, see [Section 4.6.4.2, "Address Verification"](#).

#### Modifying Delivery Lines After Release

If the Allow Modification of Delivery Lines After Release rule is enabled, the user can change the shipping address of order lines that have the Delivery fulfillment method and are not in the `Shipped` status.

#### Viewing Service Fulfillment Summary

After changing the address on the order using the `changeOrder` API, the `generateWorkOrder` API is used to create the delivery lines and generate work orders. The user can change the delivery appointment for the customer, if applicable. The items are reserved based on the configuration. For more information about service fulfillment summary, see [Section 4.9.24, "Service Fulfillment Summary"](#). For more information about service appointments, see [Section 4.9.23, "Change Service Appointment"](#).



---

---

**Note:** The order lines with fulfillment method as "Shipping" are not reserved when the user changes the shipping address.

---

---

#### 4.9.29.2 End-User Impact

None.

#### 4.9.29.3 Implementation

This section explains the configurations for the change order address task.

- Sterling Call Center and Sterling Store allows you to configure the order statuses for which changing the address is allowed. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure availability checks and reservation options. For more information about configuring availability checks and reservation options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.
- Notes entered for this task are saved on the order with YCD\_SHIPPING\_ADDRESS note type.

#### 4.9.29.4 Reference Implementation

This section explains the reference implementation provided as part of change order address task.

- By default, the Prevent Availability Checks During Order Entry and Order Modifications rule is set to "N" for all the organizations.
- Sterling Call Center and Sterling Store provides permissions to change the order address to all users as part of reference implementation.

### 4.9.30 Launch Application Console

In some implementations, call center and store representatives may need to access screens and functionality from the Application Console. For example, call center and store representatives may need to access the detailed order audit screens available in the Application Console.

#### 4.9.30.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which allows users to launch the Application Console from within the Sterling Call Center and Sterling Store user interface.

Sterling Call Center and Sterling Store allows you to configure the URL to be used when launching the Application Console from the Sterling Call Center and Sterling Store user interface so that you can configure a specific screen or set of screens that will be opened.

**Note:** Since Application Console are not supported on Linux, you cannot launch the Application Console from within the Sterling Call Center and Sterling Store user interface for Linux based installations.

#### 4.9.30.2 End-User Impact

None.

#### 4.9.30.3 Implementation

This section explains the configurations for this task.

- To launch the Application Console from the Sterling Call Center and Sterling Store user interface, you must configure the Config element "COM.Console" in the `locations.ycfg` file. For information about editing the `locations.ycfg` file, refer to the *Sterling Call Center and Sterling Store: Installation Guide*.

For information about configuring settings to allow logging into the browser, refer to the *Selling and Fulfillment Foundation: Installation Guide*.

- If the client machine's network connection is configured to use a proxy server and port:

- Ensure that the proxy server and port are specified in connection settings for the client machine's default browser.
- Ensure that the BaseUrl and PortNumber are set in the `locations.ycfg` file.

For example if the client machine's network connection is configured to use a proxy server "Proxy" and Port "8080", you should set the following in the `locations.ycfg` file.

```
BaseUrl = "Proxy"
PortNumber = "8080"
```

For more information about editing the `locations.ycfg` file, refer to the *Sterling Call Center and Sterling Store: Installation Guide*.

- Sterling Call Center and Sterling Store allows you to configure the URL to be used when launching the Application Console from the Sterling Call Center and Sterling Store user interface. For more information about configuring the URL, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

Sterling Call Center and Sterling Store allows you to configure URLs for each enterprise in your organization. If the user belongs to multiple enterprises, the URL corresponding to the user's primary enterprise is used.

- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### 4.9.30.4 Reference Implementation

This section explains the default values provided for this feature.

- The rule to configure the URL to be launched is set to the Application Console URL for all the organizations.
- Sterling Call Center and Sterling Store provides permissions to launch the Application Console to hub users as part of reference implementation.

### 4.9.31 Change an Item's Style

After placing an order for a stylized item, a customer may want to change the attributes of that item.

#### 4.9.31.1 Solution

Sterling Call Center and Sterling Store provides the Change Items Style functionality which enables the user to change an item's style even after the item is added to the order.

The `getCompleteItemList` API is called to display the different styles for an item. On entering the partial Item ID of a model item or a stylized item, a pop-up window appears, in which the user can select the corresponding item by selecting the attributes that are used to model the item. For more information about handling styles for an item, see [Section 4.1.2, "Handling Styles for an Item"](#).

When an item's style is changed:

- The original item is cancelled and a new item is added.

<b>Note:</b> The user will be able to change an item's style only if status modification rules permit the cancellation and addition of lines.
---

- If the order line containing the original item is associated with a related item, the same association applies to the order line containing the newly added item.

When the attributes of a stylized item are changed, the `changeOrder` API is called to cancel the existing order line, and the `changeOrder` API is called to add a new line.

#### 4.9.31.2 End-User Impact

None.

#### 4.9.31.3 Implementation

The change an item's style task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### 4.9.31.4 Reference Implementation

Sterling Call Center and Sterling Store provides the necessary permissions to change an item's style to all the user groups as part of reference implementation.

## 4.9.32 Reconfigure Bundle Items

Sterling Call Center and Sterling Store representatives may have to reconfigure bundle items. For example, a customer may place an order for a computer that was originally configured with a slower, inexpensive processor, but may later ask the CSR to reconfigure the computer to have a faster, more expensive processor.

### 4.9.32.1 Solution

Sterling Call Center and Sterling Store enable users to reconfigure bundle items.

This section explains the Reconfigure Item process, and the solution offered by Sterling Call Center and Sterling Store.

#### Reconfiguring an Item

The `getCompleteOrderDetails` API is called to display the details of the order lines. When the Reconfigure hyperlink is clicked, the Product Configurator is launched from within Sterling Call Center and Sterling Store. For more information about launching the Product Configurator from within Sterling Call Center and Sterling Store, refer to the *Sterling Call Center and Sterling Store: Installation Guide*. For more information about the Product Configurator, see [Product Configurator](#).

After the item is reconfigured, the `changeOrder` API is called to update the order.

#### Changing the Fulfillment Method of an Item

A user can change the fulfillment method of an item after reconfiguring the item. For more information about changing the fulfillment method, see [Section 4.9.19, "Change Fulfillment Options"](#).

#### Viewing the Pick Up and Shipping Fulfillment Summary

After modifying the fulfillment method of an item, a user can view the details of the fulfillment summary in the Fulfillment Summary screen. The items are reserved based on the configuration. For more information about fulfillment summary, see [Section 4.9.20, "Fulfillment Summary"](#).

### Viewing the Service Fulfillment Summary

After adding items to an order using the `changeOrder` API, if an user selects the Delivery option, the `generateWorkOrder` API is used to create the delivery lines and generate work orders. The user can change the delivery appointment for the customer, if applicable. Items are reserved based on the configuration. For more information about service fulfillment summary, see [Section 4.9.24, "Service Fulfillment Summary"](#). For more information about service appointments, see [Section 4.9.23, "Change Service Appointment"](#).

### Confirming the Payment Details

If the order total changes because of the changes to the configuration of an item, or changes to the fulfillment options, the system displays the updated payment details. If a customer wants to make additional payments, the user must select the corresponding payment method. For more information about confirming the payment details, see [Section 4.9.3, "Change Payment Method"](#).

#### 4.9.32.2 End-User Impact

None.

#### 4.9.32.3 Implementation

This section explains the configurations for this task.

- Sterling Call Center and Sterling Store allow you to configure external payment options. For more information about configuring external payment options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allow users to reconfigure bundle items only if the change bundle definition status modification rule is enabled. For more information about configuring the order modification rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission-controlled. Sterling Call Center and Sterling Store allow you to assign permissions to user groups for this task.

#### 4.9.32.4 Reference Implementation

Sterling Call Center and Sterling Store provide the necessary permissions to launch the Reconfigure Items task to all users as part of reference implementation.

### 4.9.33 Change Tax Options

A business customer may want to request tax exemptions on an order after placing the order. A CSR can allow tax exemptions on an order for a business customer.

#### 4.9.33.1 Solution

Sterling Call Center and Sterling Store provides a user interface that allows a user to indicate whether an order should be exempt from tax, based on the permissions provided to the user. The `getCompleteOrderDetails` API is called to retrieve the tax options for the order, and the `changeOrder` API is called to save the tax options on the order.

#### 4.9.33.2 End-User Impact

None.

#### 4.9.33.3 Implementation

This section explains the configurations for the Change Tax Options task.

- Sterling Call Center and Sterling Store allows you to configure the order statuses for which a CSR can allow tax exemptions. Ensure that the status modification rules allow the task of changing tax options for an order. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Depending on whether taxes are applied at the header level or the line level, implement either the `YFSRecalculateHeaderTaxUE` user exit or the `YFSRecalculateLineTaxUE` user exit to recalculate taxes after the tax exemption code is entered.
- This task is permission-controlled. Sterling Call Center and Sterling Store allows you to assign permissions to the corresponding user groups for this task.

- Notes entered for this task are saved in the order using the YCD\_TAX\_OPTIONS note type.

### 4.9.33.4 Reference Implementation

The Sterling Call Center and Sterling Store application provides permissions to CSR Leads and store managers as part of the reference implementation.

## 4.9.34 Copy Order

When creating an order or viewing an order that was placed earlier, a customer may decide to use that order to create new orders in the future, where applicable. For example, a customer may place an order for a particular list of items every month, with the list varying only slightly from month to month.

### 4.9.34.1 Solution

Sterling Call Center and Sterling Store provides the Copy Draft Order related task, which enables a user to copy an existing draft order, and the Order Again related task to create a new draft order from a confirmed order, based on the permissions provided to the user.

The copyOrder API is called to copy an existing draft order or confirmed order, including all the attributes defined in the Copy Order template in the Applications Manager. For more information about defining a process type's templates, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

**Note:** When a user copies an order, the EnterpriseCode of the existing order will be retained in the copied order.

### 4.9.34.2 End-User Impact

None.

### 4.9.34.3 Implementation

This section explains the configurations for the Copy Order task:



- You should define the order and order line attributes that should be copied when copying an order in the Copy Order template in the Applications Manager.
- This task is permission-controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for the two related tasks.

#### 4.9.34.4 Reference Implementation

This section explains the reference implementation provided as part of the Copy Order task:

- Sterling Call Center and Sterling Store provides a Copy Order template that can be modified depending on the way this functionality is implemented.
- Sterling Call Center and Sterling Store provides permissions for the Copy Draft Order and Order Again tasks to all user groups provided by Sterling Call Center and Sterling Store.

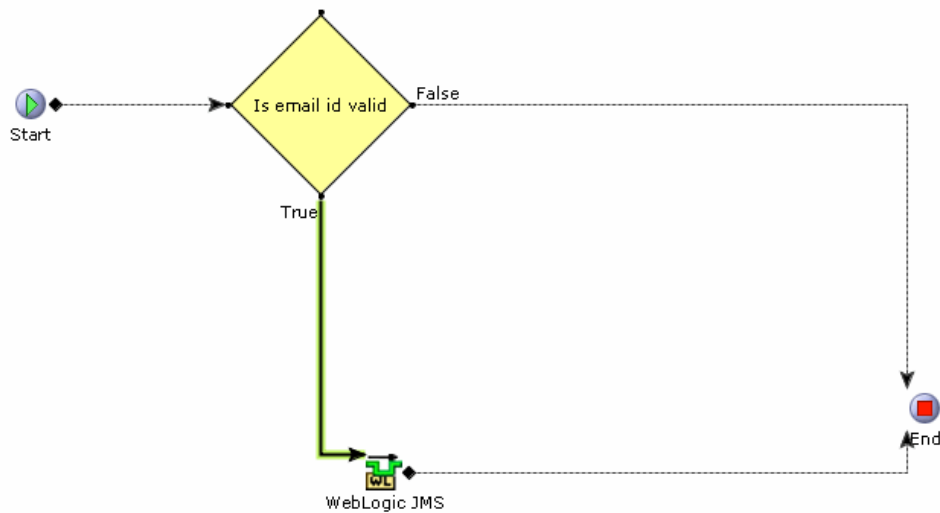
### 4.9.35 E-Mail Order Information

A customer may request a CSR to send information pertaining to an order through e-mail. A customer may, for example, want to review the items that have been included in the order.

#### 4.9.35.1 Solution

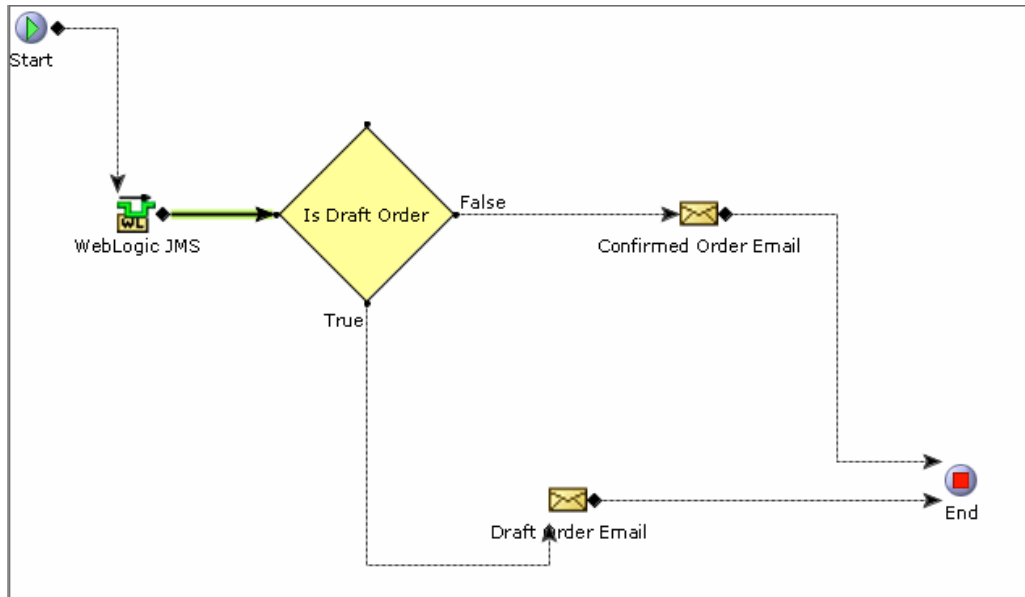
Sterling Call Center and Sterling Store provides the Send Order E-mail related task to enable users to send information pertaining to a draft order or a confirmed order to the e-mail address of the customer to whom the order belongs. Users can send the order information to a single e-mail address or multiple e-mail addresses.

The YCDEmailOrderUE user exit is called to send the order information to the customer through e-mail. The YCDEmailOrderUE user exit uses the YCD\_Populate\_Order\_Email\_8.5 service and the YCD\_Send\_Order\_Email\_8.5 service illustrated in [Figure 4–12](#) and [Figure 4–13](#) to send the order information through e-mail.

*Figure 4–12 YCD\_Populate\_Order\_Email\_8.5 Service*

The YCDEmailOrderUE user exit calls the `getCompleteOrderDetails` API to retrieve the relevant information required for sending the e-mail to the customer. The information is passed to the YCD\_Populate\_Order\_Email\_8.5 service and the e-mail is sent to the customer using the YCD\_Send\_Order\_Email\_8.5 service, as illustrated in [Figure 4–13](#).

Figure 4–13 YCD\_Send\_Order\_Email\_8.5 Service



By default, Sterling Call Center and Sterling Store provide the following e-mail templates:

- YCD\_Draft\_Email\_Order.xml.sample
- YCD\_Confirmed\_Email\_Order.xml.sample

The XSL's must be renamed to YCD\_Draft\_Email\_Order.xml and YCD\_Confirmed\_Email\_Order.xml respectively, customized as needed, and their path and names should be specified in the Body Template of the E-mail Sender properties.

### Validation of E-Mail Addresses

Sterling Call Center and Sterling Store validates the e-mail addresses to which a user has to send order information. The e-mail addresses are validated based on the following criteria:

- The first character must be an alphabet.
- The symbol @ must be present.
- There must be at least one character before at.
- There must be at least one character between at and the period.

- There must be between 2-4 characters after the period.
- There must not be spaces in the e-mail address.
- There must be a semi-colon (;) between addressess.

If any of these criteria fail, Sterling Call Center and Sterling Store displays an error message to notify the user of the incorrect criterion.

### **4.9.35.2 End-User Impact**

None.

### **4.9.35.3 Implementation**

This section explains the configurations for the Send Order E-mail task:

- To use this feature, you should implement the YCDEmailOrderUE user exit.
- The Send Order E-mail related task is permission-controlled. Sterling Call Center and Sterling Store allows you to assign permissions to the corresponding user groups for this task.

### **4.9.35.4 Reference Implementation**

Sterling Call Center and Sterling Store provides permissions to all the user groups as part of the reference implementation.

## **4.9.36 Transfer Draft Order to Sterling Web**

Draft orders can be created either by a customer service representative through the Sterling Call Center and Sterling Store applications, or by a customer through the Sterling Web application. Draft orders created in Sterling Web are accessible through Sterling Call Center and Sterling Store. However, draft orders that are opened through Sterling Call Center and Sterling Store are not accessible through Sterling Web. In such situations, CSRs may want to transfer the draft order from Sterling Call Center and Sterling Store to Sterling Web in order to enable customers to access the draft orders.

#### 4.9.36.1 Solution

Sterling Call Center and Sterling Store enable users to transfer the draft orders, which have been created or opened in the Sterling Call Center and Sterling Store applications, to the Sterling Web application.

When a CSR or a customer creates a draft order through Sterling Call Center, Sterling Store, or Sterling Web, and if the draft order is opened in Sterling Call Center or Sterling Store, the value of the ChannelID is set to Call Center or Store respectively. This prevents the order from being opened through different channels. For example, if a customer has created a draft order in Sterling Web, and this draft order is opened by a CSR through the Sterling Call Center application, the ChannelID on that order is automatically changed from Web to Call Center. However, when the order is closed, the ChannelID does not revert to Web. If the customer requests the ChannelID to be changed to Web, the CSR can change the ChannelID using the Transfer Order to Web task.

However, note that not all draft orders can be transferred from Sterling Call Center and Sterling Store to Sterling Web. When changes are made to a draft order, a tag is applied to the draft order. For example, if the price on an order is overridden, the IsPriceLocked flag is set to Y. An order tag can be applied to the draft order indicating that there is at least one line in the draft order with an overridden price. In such a scenario, the draft order cannot be modified from the current version of Sterling Web, and should therefore not be transferred from Sterling Call Center and Sterling Store to Sterling Web. Sterling Call Center and Sterling Store also enable you to specify the version of the Sterling Web application that you want to be associated to Sterling Call Center and Sterling Store. You can also define tags to specify the support levels for the applications.

---

**Note:** If you do not specify the version of the Sterling Web application, the Transfer Order to Web task will not be displayed to users.

---

When the draft order is opened in Sterling Call Center and Sterling Store, and if the ChannelID on the order is set to Web, the changeOrder API is called to change the ChannelID to Call Center or Store. If a CSR wants to transfer a draft order from Sterling Call Center and Sterling Store to Sterling Web, the getCompleteOrderDetails API is called to check the support level of the corresponding draft order for the version of the

Sterling Web application to which the draft order has to be sent, and any tags that are applied to the draft order. Based on these values, the Sterling Call Center and Sterling Store applications will determine whether the draft order can be sent to Sterling Web. If the draft order can be sent to Sterling Web, the ChannelID is changed to Web, and the Sterling Call Center and Sterling Store applications prompt the CSR to enter a name for the draft order. Otherwise, the ChannelID on the draft order will not be changed. An appropriate message is displayed, indicating that the draft order cannot be sent to Sterling Web because it is incompatible with the version of the Sterling Web application used by the customer.

### 4.9.36.2 End-User Impact

The Transfer Order to Web task will be displayed only if the version of the Sterling Web application is specified in the Channel Applications Manager.

### 4.9.36.3 Implementation

You can define the version of the Sterling Web application that is associated to Sterling Call Center and Sterling Store. For more information about specifying the version of Sterling Web, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

You can also define tags to specify the support levels for the Sterling Call Center and Sterling Store applications. For more information about specifying the order tags for a sales order, refer to the *Channel Applications Manager: Configuration Guide*.

### 4.9.36.4 Reference Implementation

None.

## 4.10 Payment Processing

In a customer order scenario, each order has to be paid for whenever it is placed. The user can use different payment types to pay for an order. To process these payment types, different requests need to be made to payment servers. For example, a credit card needs to be validated as not being fraudulent, having sufficient funds, and charged. However, for a stored value card (SVC), validations need to be done to see whether it has sufficient funds and is charged.

The commonly used payment types need to be made available to the customers placing orders. The payment types need to be validated and charged by external payment validation servers.

If your enterprise implements its own or a third-party payment system, the Payment Summary screen is not displayed. This option is set through the external payment rule, YCD\_PAYMENT\_ENTERED\_EXTERNALLY.

### 4.10.1 Solution

An unlimited number of payment methods can be used on an order. The payment methods are based on the following payment types defined in Sterling Call Center and Sterling Store:

- [Credit Card](#)
- [Stored Value Card](#)
- [Pre-paid](#)
- [Check](#)
- [Customer Account](#)
- [Refund Check](#)

Through user exit implementations, Sterling Call Center and Sterling Store interacts with the payment servers to authorize and charge the payment methods, if applicable. It is assumed that the front-end system attempts to validate and charge the payment methods up front, and Sterling Call Center and Sterling Store attempts to validate any unauthorized or uncharged amount at the time of order capture, or when a payment method is modified on the order.

Sterling Call Center and Sterling Store provides the capability to put an order on hold if the spending limit of the customer is breached. The hold type that is applied to the order is determined by the Hold Type To Be Applied When Order Needs Approval rule. To use this functionality, you must enable the Hold Type To Be Applied When Order Needs Approval rule in the Applications Manager. For more information about defining transaction rules, refer to the *Sterling Distributed Order Management: Configuration Guide*.

When any payment method validations or charges fail, the system invokes configurable services to send alerts to queues that are monitored by user groups.

Based on the encryption rules configured as part of the Installation Rules provided by Selling and Fulfillment Foundation, some of the fields of the different payment methods can be encrypted. These encryption rules are read by the client and the encrypted payment attributes are hidden from the user. For more information about Installation Rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

**Table 4–33 Attributes that may be encrypted**

<b>Payment Method</b>	<b>Rule—Encrypt Primary Payment Attributes for the following Payment Type Groups</b>	<b>Attributes</b>	<b>UI Behavior</b>
Credit Card	Credit Card	Credit Card #	Display Credit Card #.
Credit Card	Encrypt Credit Card Name and Credit Card Expiration Date	Credit Card Name, Expiration Date	The Credit Card Name is displayed as a set of asterisks.  The Expiration Date fields specified, i.e. the Month and Year, are also displayed as a set of asterisks.
Stored Value Card	Stored Value Card	SVC #, Payment Reference 1	Display SVC #, Display Payment Reference 1.
Customer Account	Customer Account	Customer Account #	Display Customer Account #.
Check/PrePaid/Other Payment	Other	Payment Reference 1	Display Payment Reference 1.

### Payment Method Display Sequence

The payment type's charge sequence is used when adding payment methods to the order. If the charge sequence of two payment methods is the same, the following logic is used to display the payment methods in the Confirm Payment screen:

- Payment methods of the OTHER payment type group are displayed first.



- Payment methods of the STORED VALUE CARD payment type group are displayed after the payment methods of the OTHER payment type group.
- Payment methods of the CREDIT CARD and CUSTOMER ACCOUNT payment type groups are displayed after the payment methods of the STORED VALUE CARD payment type group.
- In case of multiple stored value cards, based on the funds available on the payment method, the stored value cards are sorted in ascending order (from the least amount available to the highest).

### **Logic Used to Apply Charge Amounts**

The charge amount will be automatically added to the payment methods of the OTHER and STORED VALUE CARD payment type groups based on the MaxChargeLimit and FundsAvailable attributes defined for the groups respectively.

If only one credit card or customer account exists on the order along with the payment methods of the OTHER payment type group or STORED VALUE CARD payment type group or both the groups, the charge amount will be automatically added to the payment methods of the OTHER and STORED VALUE CARD payment type groups, and the remainder of the order total will be applied as the charge amount on the credit card or customer account.

During order modification, only the additional amount of the order will be required for charge amount entry. The existing stored value card, credit card, and customer accounts will be visible for assigning additional charge amounts, if applicable. The MaxChargeLimit will be increased on these payment methods, but no real-time authorization will occur. The CSR can also add additional payment methods, if required.

#### **4.10.1.1 Credit Card**

Credit cards always need to be checked for fraudulent activity and funds availability by a payment system. It is assumed that the front-end systems attempt to check the credit cards up front for fraud, and attempt to authorize them. If the validation and authorization are successful, the order is captured with the authorization details. However, if the payment verification system is down, or if a payment type is changed on the order, payment verification may need to be completed by Sterling Call Center and Sterling Store payment processing agents.

When an order is captured, the request collection agent creates an authorization and charge request for any unauthorized amount. The requestCollection API is called and creates an authorization or charge request, which is picked up by the executeCollection API. The executeCollection API in turn calls the YFSCollectionCreditCardUE user exit, which communicates with the external payment systems.

If the credit card is authorized, the order is processed normally. If the user exit returns a response code indicating that something is wrong with the credit card, the order is not processed, and an alert is sent to the appropriate queue depending on why the credit card failed the authorization.

For more information about implementing the credit card payment method, see [Section 4.10.3.1, "Credit Card Implementation"](#).

Sterling Call Center and Sterling Store provides functionality to process real-time credit card authorizations while confirming draft orders. For more information about real-time credit card authorizations, see [Section 4.12, "Real-Time Credit Card Authorization"](#).

### **4.10.1.2 Stored Value Card**

Unlike credit cards, a stored value card (SVC) is always assumed not to be fraudulent, since it is issued by the store itself. However, SVCs need to be checked for available funds. Similar to credit cards, this is assumed to be completed by the front-end system, but any SVC payment method not validated for funds is processed by Sterling Call Center and Sterling Store.

When SVC payment types are created, ensure that the "Charge Up To Available" option is selected.

When an order is captured, the Request Collection agent creates a charge request for any uncharged amount, which is picked up by the Payment Execution agent. The Request Collection agent calls the payment server to get the available funds using the YFSGetFundsAvailableUE user exit. The default implementation for this user exit is not provided.

The executeCollection API calls the YFSCollectionStoredValueCardUE user exit for the payment collection.

If the SVC is successfully charged, the order is processed normally. If the user exit returns a response code indicating that the SVC could not be

processed, either because of unavailable funds or due to the payment system being unavailable, the order is not processed, and an alert is sent to the appropriate queue.

Sterling Call Center and Sterling Store assumes that the FundsAvailable on an SVC are adjusted as soon as the authorization call is made on the payment method. Therefore, in order modification scenarios, only the additional amount required is considered when comparing against the FundsAvailable attribute.

For more information implementing the stored value card method, see [Section 4.10.3.2, "Stored Value Card Implementation"](#).

#### **4.10.1.3 Pre-paid**

An entire order or a part of the an order may be charged before the order is captured. For example, a customer may place an order in a store, which can be picked up from there at a later time, at another store, or shipped to their home. Full or partial payment can be made for that order, and the store operator can create that order online providing all payment information.

The amount that has already been paid at the time of order capture falls under the Pre-paid payment method.

When adding the pre-paid payment method to an order, the following fields are captured by Sterling Call Center and Sterling Store:

- Transaction Number (corresponds to Payment Reference #1)
- Register Number (corresponds to Payment Reference #2)
- Store Tender Type (corresponds to Payment Reference #3)

#### **4.10.1.4 Check**

An entire order or a part of the order can be paid using a check. Sterling Call Center and Sterling Store assumes that a check payment method is pre-charged and pre-approved. The check amount should be specified, and is marked as the collected amount.

When adding the check payment method to an order, the following fields are captured in Sterling Call Center and Sterling Store:

- Check Account Number (corresponds to Payment Reference #1)
- Check Amount (corresponds to Payment Reference #2)

- Check Reference (corresponds to Payment Reference #3)

### 4.10.1.5 Customer Account

An entire order or a part of an order can be paid for by using a customer account payment type. The customer account payment type is a payment type where a credit amount is assigned to a customer by the corresponding enterprise. Users can specify the account limit for a customer account payment type.

When an order is created or modified, the customer account payment type is validated to check for sufficient funds. If a customer account payment type has insufficient funds to complete a transaction, a warning message is displayed, indicating that the funds available in the account are insufficient to fulfill the order, and that the order may be put on hold until sufficient funds are provided. The `getCustomerDetails` API is called to retrieve the account limit for a customer account payment type. For more information about configuring the payment rules, refer to the *Sterling Distributed Order Management: Configuration Guide*.

When adding a customer account payment type for a customer, information pertaining to the following fields is captured in the Sterling Call Center and Sterling Store application:

- Customer Account Number
- Currency
- Account Limit

### 4.10.1.6 Refund Check

When a check or small amount of money need to be refunded to the customer, a refund check can be issued and mailed to them.

The request collection agent creates a request for the refund amount. A refund check is always assumed to be valid, and there is no authorization mechanism. The charge request is picked up by the `executeCollection` API, which calls the `YFSCollectionOthersUE` user exit to interact with the payment system.

For more information about implementing the refund check payment method, see [Section 4.10.3.3, "Return Check Implementation"](#).

---

---

**Note:** Although the Refund Check payment type is available on a sales order, it must only be used in the context of a return.

---

---

## 4.10.2 End-User Impact

The payment processing of Sterling Call Center and Sterling Store affects users of the Application Console in the following ways:

- Certain user groups can view alerts raised by failed payment validations or charges. The users who can view the alert details depends on the alert services configuration. The end users resolve alerts by calling the customers and replacing the payment method on the order. For more information about implementing alert services, see [Section 4.10.3.4, "Payment Processing Error Handling"](#).
- When modifying orders, the users are able to select the available payment types as payment methods on the orders.
- The strike limit is displayed on the order, and if that limit exceeds, an alert is raised. For more information about strike limit configuration, see [Section 4.10.3.5, "Strike Limit Configuration"](#).

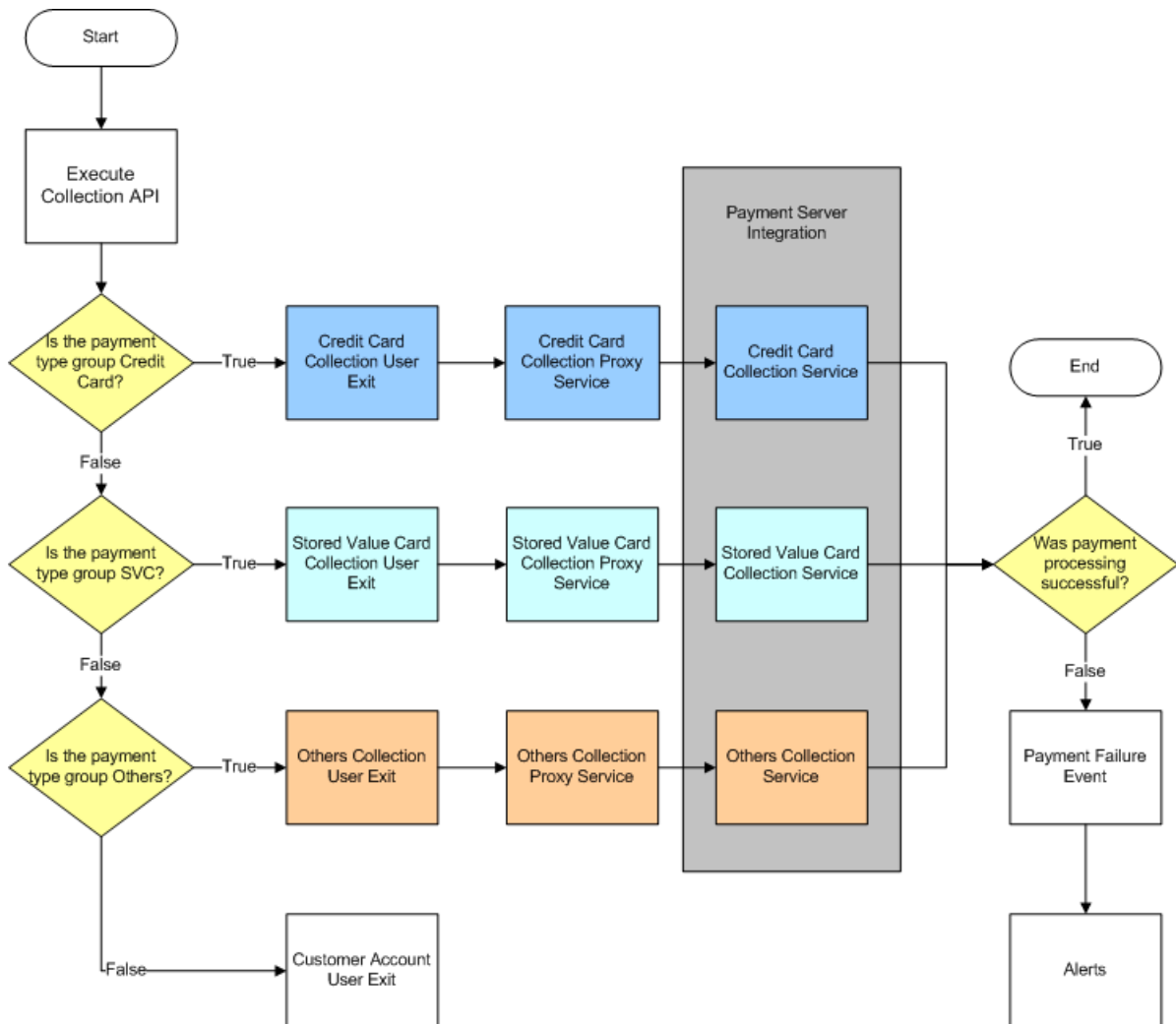
## 4.10.3 Implementation

All payment processing goes through the `executeCollection` API. This API calls the appropriate user exits depending on the payment method being used. The those user exits call the system defined payment services. These services are pointers to the customizable payment service implementations, which call the external payment server to perform authorizations and charges.

Depending on the response codes returned by the payment server, the user exits raise an event which invokes services responsible for sending alerts to the appropriate queues if certain user groups need to be notified of the outcome of the payment verification attempt. For more information about error handling in payment processing, see [Section 4.10.3.4, "Payment Processing Error Handling"](#).

[Figure 4–14](#) depicts the flow of payment processing in Sterling Call Center and Sterling Store.

*Figure 4–14 Payment Processing*



The following topics are covered in this section:

- [Credit Card Implementation](#)
- [Stored Value Card Implementation](#)

- [Return Check Implementation](#)
- [Payment Processing Error Handling](#)
- [Alert Services](#)
- [Strike Limit Configuration](#)

#### 4.10.3.1 Credit Card Implementation


In order for a credit card to be validated as a payment method, you must implement the YFSCollectionCreditCardUE user exit. It takes its input from the executeCollection API. From that input, the following XML is created, and passed to a java class which in turn calls the YCD\_ExecuteCollectionCreditCard\_Proxy\_1.0 service, a pointer to the YCD\_ExecuteCollectionCreditCard\_1.0 service which is responsible for integrating with the external payment system:

```
<Payment AuthorizationId="" BillToAddressLine1="" BillToCity="" BillToCountry=""
BillToDayPhone="" BillToEmailId="" BillToFirstName="" BillToId="" BillToKey=""
BillToLastName="" BillToState="" BillToZipCode="" bPreviouslyInvoked=""
ChargeTransactionKey="" ChargeType="" CreditCardExpirationDate=""
CreditCardName="" CreditCardNo="" CreditCardType="" Currency=""
CustomerAccountNo="" CustomerPONo="" DocumentType="" EnterpriseCode=""
MerchantId="" OrderHeaderKey="" OrderNo="" PaymentReference1=""
PaymentReference2="" PaymentReference3="" PaymentType="" RequestAmount=""
ShipToAddressLine1="" ShipToCity="" ShipToCountry="" ShipToDayPhone=""
ShipToEmailId="" ShipToFirstName="" ShipToId="" ShipToKey="" ShipToLastName=""
ShipToState="" ShipToZipCode="" SvcNo="">
```

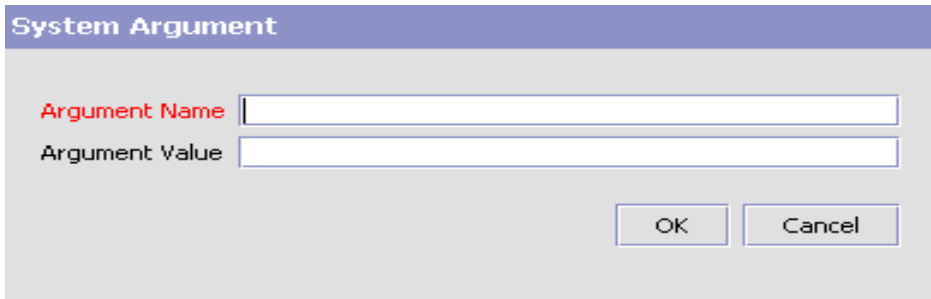
You should create your own custom service to handle the integration with the external payment system, and modify the YCD\_ExecuteCollectionCreditCard\_Proxy\_1.0 service to point to your service.

To modify YCD\_ExecuteCollectionCreditCard\_Proxy\_1.0 service to use your own custom service, perform following steps:

1. Launch the Applications Manager.
2. From Distributed Order Management > Document Specific > Sales Order > Fulfillment > Fulfillment Process Model > Service Repository > Payment Processing.
3. From the application rules side panel, select YCD\_ExecuteCollectionCreditCard\_Proxy\_1.0 service.

4. Select the Execute Collection Check API component.
5. Select the Arguments tab.
6. Click . The Service Argument window is displayed as shown in [Figure 4–15](#).
7. In Argument Name, enter ServiceName.
8. In Argument Value, enter the name of your own custom service.

**Figure 4–15** *System Argument Window for API Component*



The image shows a dialog box titled "System Argument". It has two text input fields. The first field is labeled "Argument Name" in red text. The second field is labeled "Argument Value" in black text. At the bottom right of the dialog box, there are two buttons: "OK" and "Cancel".

**Note:** Sterling Call Center and Sterling Store reference implementation provides a demo service, YCD\_ExecuteCollectionCreditCard\_1.0 service, to integrate with the external payment system.

The expected output of the YCD\_ExecuteCollectionCreditCard\_Proxy\_1.0 service is in the following XML format:

```
<Payment ResponseCode="" AsynchRequestProcess="" AuthAVS="" AuthCode=""
AuthorizationAmount="" AuthorizationExpirationDate="" AuthorizationId=""
AuthReturnCode="" AuthReturnFlag="" AuthReturnMessage="" AuthTime=""
BPreviousInvocationSuccessful="" CollectionDate="" DisplayPaymentReferencel=""
HoldOrderAndRaiseEvent="" HoldReason="" InternalReturnCode=""
InternalReturnFlag="" InternalReturnMessage="" PaymentReferencel=""
PaymentReference2="" PaymentReference3="" RequestID="" RetryFlag=""
SCVVAuthCode="" SuspendPayment="" TranAmount="" TranRequestTime=""
TranReturnCode="" TranReturnFlag="" TranReturnMessage="" TranType="" />
```



The value of the `ResponseCode` attribute returned by the user exit determines how to handle the order, if the payment method fails the validation:

If the payment processing fails for any reason, the `COLLECTION_FAILED` is raised by the `YFSCollectionCreditCardUE` user exit. The event calls the `YCDOnCollectionFailure` action, which invokes the `YCD_ProcessCollectionFailure_1.0` service, responsible for calling the appropriate alert service depending on the `ResponseCode`.

- **APPROVED**—This response code returns if the authorization or charge was successful.
- **HARD\_DECLINE**—This response code returns if the authorization or charge fails because the card is invalid. For example, a lost or stolen credit card.

In this case, the order's strike count is increased by 1. The strike count represents how many times an order has had its payment declined with a `HARD_DECLINE` response code. The maximum allowed number of strikes is configurable. For more information about strike limit configuration, see [Section 4.10.3.5, "Strike Limit Configuration"](#).

- If the strike count is higher than the maximum allowed number of strikes, the payment method is marked not to retry authorization, suspended, updated with the failure details, and the `YCD_PaymentAuthRetryLimitAlert_1.0` alert service invoked. The alert is sent to the Payment Auth Retry Limit Reached queue, which is monitored by the Fraud Analyst Group users.
- If the strike count is lower than the maximum allowed number of strikes, the payment method is marked not to retry authorization, suspended, updated with the failure details, and the `YCD_PaymentHardDeclinedAlert_1.0` alert service is invoked. The alert is sent to the Payment Hard Declined queue, which is monitored by the Fraud Analyst Group users.
- **SOFT\_DECLINE** and **BANK\_HOLD**—This response code is returned if the authorization or charge fails because the card was declined. For example, an invalid CVV number or insufficient funds.

The payment method is marked not to retry authorization, suspended, updated with the failure details, and the `YCD_PaymentDeclinedAlert_1.0` alert service invoked. The alert is

sent to the Payment Declined queue, which is monitored by the CSR Group users.

- **SERVICE\_UNAVAILABLE**—This response code returns if the credit card authentication service could not be reached.

The payment method is marked as retry charge, is not suspended, not updated with the failure details, and the YCD\_PaymentServiceUnavailableAlert\_1.0 alert service is invoked. The alert is sent to the Payment Service Unavailable queue, which is monitored by the System Group users.

For more information about error handling in payment processing, see [Section 4.10.3.4, "Payment Processing Error Handling"](#).


### 4.10.3.2 Stored Value Card Implementation

The Stored Value Cards do not need to be authorized. It needs to be only charged. In order for stored value cards to be charged, you must implement the YFSCollectionStoredValueCardUE user exit. This user exit takes its input from the executeCollection API. From that input, the following XML is created, and passed to a java class which in turn calls the YCD\_ExecuteCollectionSVC\_1.0\_Proxy service, a pointer to the YCD\_ExecuteCollectionSVC\_1.0 service responsible for integrating with the external payment system:

```
<Payment AuthorizationId="" BillToAddressLine1=""
BillToCity="" BillToCountry="" BillToDayPhone=""
BillToEmailId="" BillToFirstName="" BillToId="" BillToKey=""
BillToLastName="" BillToState="" BillToZipCode=""
bPreviouslyInvoked="" ChargeTransactionKey="" ChargeType=""
CreditCardExpirationDate="" CreditCardName="" CreditCardNo=""
CreditCardType="" Currency="" CustomerAccountNo=""
CustomerPONo="" DocumentType="" EnterpriseCode=""
MerchantId="" OrderHeaderKey="" OrderNo=""
PaymentReference1="" PaymentReference2="" PaymentReference3=""
PaymentType="" RequestAmount="" ShipToAddressLine1=""
ShipToCity="" ShipToCountry="" ShipToDayPhone=""
ShipToEmailId="" ShipToFirstName="" ShipToId="" ShipToKey=""
ShipToLastName="" ShipToState="" ShipToZipCode="" SvcNo="">
```

You should create your own custom service to handle the integration with the external payment system, and modify the YCD\_ExecuteCollectionSVC\_Proxy\_1.0 service to point to your service.

To modify YCD\_ExecuteCollectionSVC\_Proxy\_1.0 service to use your own custom service, perform following steps:

1. Launch the Applications Manager.
2. From Distributed Order Management > Document Specific > Sales Order > Fulfillment > Fulfillment Process Model > Service Repository > Payment Processing
3. From the application rules side panel, select YCD\_ExecuteCollectionSVC\_Proxy\_1.0 service.
4. Select the Execute Collection Check API component.
5. Select the Arguments tab.
6. Click . The Service Argument window is displayed as shown in [Figure 4–15](#).
7. In Argument Name, enter ServiceName.
8. In Argument Value, enter the name of your own custom service.

**Note:** Sterling Call Center and Sterling Store reference implementation provides a demo service, YCD\_ExecuteCollectionSVC\_1.0 service, to integrate with the external payment system.

The expected output of the user exit is in the following XML format:

```
<Payment ResponseCode="" AsynchRequestProcess="" AuthAVS=""
AuthCode="" AuthorizationAmount="" AuthorizationId=""
AuthReturnCode="" AuthReturnFlag="" AuthReturnMessage=""
DisplayPaymentReferencel="" DisplaySvcNo=""
HoldOrderAndRaiseEvent="" HoldReason="" PaymentReferencel=""
PaymentReference2="" PaymentReference3="" RequestID=""
RetryFlag="" SuspendPayment="" SvcNo="" TranAmount=""
TranRequestTime="" TranReturnCode="" TranReturnFlag=""
TranReturnMessage="" TranType="" />
```

The value of the `ResponseCode` attribute returned by the user exit determines how the order is handled if the payment method fails the validation:

- **APPROVED**—This response code is returned if the charge was successful.
- **DECLINED**—This response code is returned if the charge failed.  
The payment method is marked not to retry charge, suspended, updated with the failure details, and the `YCD_PaymentDeclinedAlert_1.0` alert service invoked. The alert is sent to the `Payment Declined` queue, which is monitored by the `CSR Group` users.
- **SERVICE\_UNAVAILABLE**—This response code is returned if the authentication service could not be reached.  
The payment method is marked as retry charge, is not suspended, not updated with the failure details, and the `YCD_PaymentServiceUnavailableAlert_1.0` alert service invoked. The alert is sent to the `Payment Service unavailable` queue, which is monitored by the `System Group` users.

For more information about error handling in payment processing, see [Section 4.10.3.4, "Payment Processing Error Handling"](#).

### 4.10.3.3 Return Check Implementation

In order for a charge to be issued for the refund check, you must implement the `YFSCollectionOthersUE` user exit.

The `executeCollection` API passes its output to the user exit. From that input, the following XML is created, and passed to the `YCD_ExecuteCollectionRefundCheck_Proxy_1.0` proxy service, which is the pointer to the `YCD_ExecuteCollectionRefundCheck_1.0` service responsible for integrating with the external payment system:

```
<Payment AuthorizationId="" BillToAddressLine1=""
BillToCity="" BillToCountry="" BillToDayPhone=""
BillToEmailId="" BillToFirstName="" BillToId="" BillToKey=""
BillToLastName="" BillToState="" BillToZipCode=""
bPreviouslyInvoked="" ChargeTransactionKey="" ChargeType=""
CreditCardExpirationDate="" CreditCardName="" CreditCardNo=""
CreditCardType="" Currency="" CustomerAccountNo=""
```


```

CustomerPONo="" DocumentType="" EnterpriseCode=""
MerchantId="" OrderHeaderKey="" OrderNo=""
PaymentReference1="" PaymentReference2="" PaymentReference3=""
PaymentType="" RequestAmount="" ShipToAddressLine1=""
ShipToCity="" ShipToCountry="" ShipToDayPhone=""
ShipToEmailId="" ShipToFirstName="" ShipToId="" ShipTokey=""
ShipToLastName="" ShipToState="" ShipToZipCode="" SvcNo="">

```

You must create your own custom service to integrate with the external payment system, and modify the YCD\_ExecuteCollectionRefundCheck\_Proxy\_1.0 service to point to your service.

To modify YCD\_ExecuteCollectionRefundCheck\_Proxy\_1.0 service to use your own custom service, perform following steps:

1. Launch the Applications Manager.
2. From Distributed Order Management > Document Specific > Sales Order > Fulfillment > Fulfillment Process Model > Service Repository > Payment Processing
3. From the application rules side panel, select YCD\_ExecuteCollectionRefundCheck\_Proxy\_1.0 service.
4. Select the Execute Collection Check API component.
5. Select the Arguments tab.
6. Click . The Service Argument window is displayed as shown in [Figure 4–15](#).
7. In Argument Name, enter ServiceName.
8. In Argument Value, enter the name of your own custom service.

---

**Note:** Sterling Call Center and Sterling Store reference implementation provides a demo service YCD\_ExecuteCollectionRefundCheck\_1.0 service to integrate with the external payment system.

---

The expected output of the YCD\_ExecuteCollectionRefundCheck\_Proxy\_1.0 service is in the following XML format:

```
<Payment AuthCode=" " AuthorizationAmount=" " AuthorizationId=" "
AuthReturnCode=" " AuthReturnFlag=" " AuthReturnMessage=" "
DisplayPaymentReferencel=" " HoldOrderAndRaiseEvent=" "
HoldReason=" " RequestID=" " RetryFlag=" " SuspendPayment=" "
TranAmount=" " PaymentReferencel=" " PaymentReference2=" "
PaymentReference3=" " TranReturnCode=" " TranReturnFlag=" "
TranReturnMessage=" " TranType=" " />
```

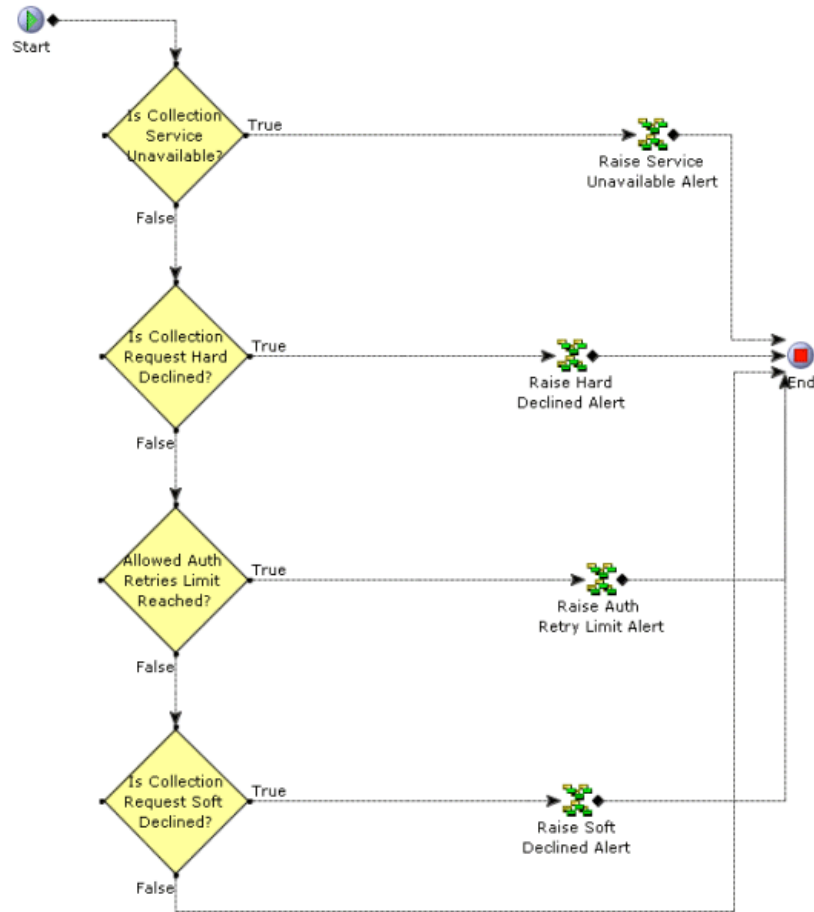
The user exit should return the `PaymentReferencel`, `PaymentReference2` and `PaymentReference3` as Check Account Number, Check Amount, and Check Reference, respectively.

### 4.10.3.4 Payment Processing Error Handling

When user exits receive response codes from the Payment Server, they raise the `COLLECTION_FAILED` event if the collection fails for any reason. That event calls the `YCDOnCollectionFailure` action, which invokes the `YCD_ProcessFailure_1.0` service.

This service examines the response code and calls one of the defined alert services to send notification to the appropriate queue. [Figure 4–16](#) illustrates the Process Collection Failure service.

**Figure 4–16** The *YCD\_ProcessCollectionFailure\_1.0* Service



All services can be accessed through the Applications Manager in the Application Platform module at the Sales Order document type level, in the Payment Processing branch service definitions tree.

The XML passed by the COLLECTION\_FAILED event to the YCD\_ProcessCollectionFailure\_1.0 service is found in the following file:

```
<INSTALL_DIR>/documentation/api_javadocs/XML/YFS_PAYMENT_EXECUTION.CHARGE_FAILED_XML.xml
```

For an example of how an error in payment processing is handled in Sterling Call Center and Sterling Store, see [Example: HARD\\_DECLINE response code on credit card collection attempt](#).

For a list of the alert services defined by default in Sterling Call Center and Sterling Store, see [Section 4.10.3.4.1, "Alert Services"](#).

### **Example: HARD\_DECLINE response code on credit card collection attempt**

If the Payment Server returns the HARD\_DECLINE response code to the YFSCollectionCreditCardUE user exit on an order that hasn't been declined before, it raises the COLLECTION\_FAILED event, which invokes the YCD\_ProcessCollectionFailure\_1.0 service through the YCDOnCollectionFailure action.

This service evaluates the XML output of the COLLECTION\_FAILED event, whether or not the payment service is available (which is false). Then, the service evaluates whether the collection service is hard declined (which is true).

Therefore, the YCD\_PaymentHardDeclinedAlert\_1.0 service invokes, which sends an alert to the Payment Hard Declined queue, which is typically monitored by the user as illustrated in [Figure 4–17](#).

**Figure 4–17 The YCD\_PaymentHardDeclinedAlert\_1.0 Service**



### **4.10.3.4.1 Alert Services**

By default, the following services are defined in Sterling Call Center and Sterling Store to send alerts in case of payment failure:

- YCD\_PaymentHardDeclinedAlert\_1.0
- YCD\_PaymentServiceUnavailableAlert\_1.0
- YCD\_PaymentAuthRetryLimitAlert\_1.0
- YCD\_PaymentDeclinedAlert\_1.0



If you need to change the queue to which payment alerts are sent, the user can edit the Alert Queue Name field on the General tab in the alert properties for the appropriate service.

#### 4.10.3.5 Strike Limit Configuration

A strike is given to an order when a credit card is declined by the payment server with `HARD_DECLINE` as a response code. An order that has had several credit cards declined because they were stolen or invalid can be considered to be fraudulent. You may want to build a mechanism that cancels the order automatically in that case.

You can define how many strikes an order can receive before an alert is sent to the fraud analyst. For more information about configuring the strike limit, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 4.10.4 Reference Implementation

This section explains the reference implementation provided as part of the Sterling Call Center and Sterling Store application's capability to put an order, which is pending approval, on hold.

- An implementation of the `YCD_ORDER_APPROVAL` hold type that prevents an order from processing the following transactions:
  - Accept Reservation
  - Close Order
  - Release Order
  - Schedule Order
  - Payment Collection
  - Payment Execution
- By default, the `YCD_ORDER_APPROVAL` hold type is defined as the value of the Hold To Be Applied When Order Needs Approval rule defined in the Applications Manager.
- An implementation of the `CHANGE_ORDER.ON_HOLD_TYPE_STATUS_CHANGE` event that generates e-mails for the approver of the order and the customer to whom the order belongs. If the e-mail cannot be sent because the approver of the order does not have an e-mail address, an alert will

be generated to inform the CSR to contact the approver of the order. The alert sent to the Order Approval queue uses the following XSL by default:

```
<INSTALL_DIR>\repository\xapi\template\merged\exception_console\ycd\YCD_OrderApprovalAlert.xsl.sample
```

This XSL should be copied to a file named `YCD_OrderApprovalAlert.xsl`, customized as needed, and its path and name should be specified in the Body Template field of the Alert properties.

The `CHANGE_ORDER.ON_HOLD_TYPE_STATUS_CHANGE` event is raised when the `YCD_ORDER_APPROVAL` hold type is applied to the order through the `CHANGE_ORDER` transaction. The `CHANGE_ORDER.ON_HOLD_TYPE_STATUS_CHANGE` event invokes a service that generates an e-mail, which is sent to the approver of the order and the customer who placed the order, or an alert to indicate that the order is on hold pending approval.

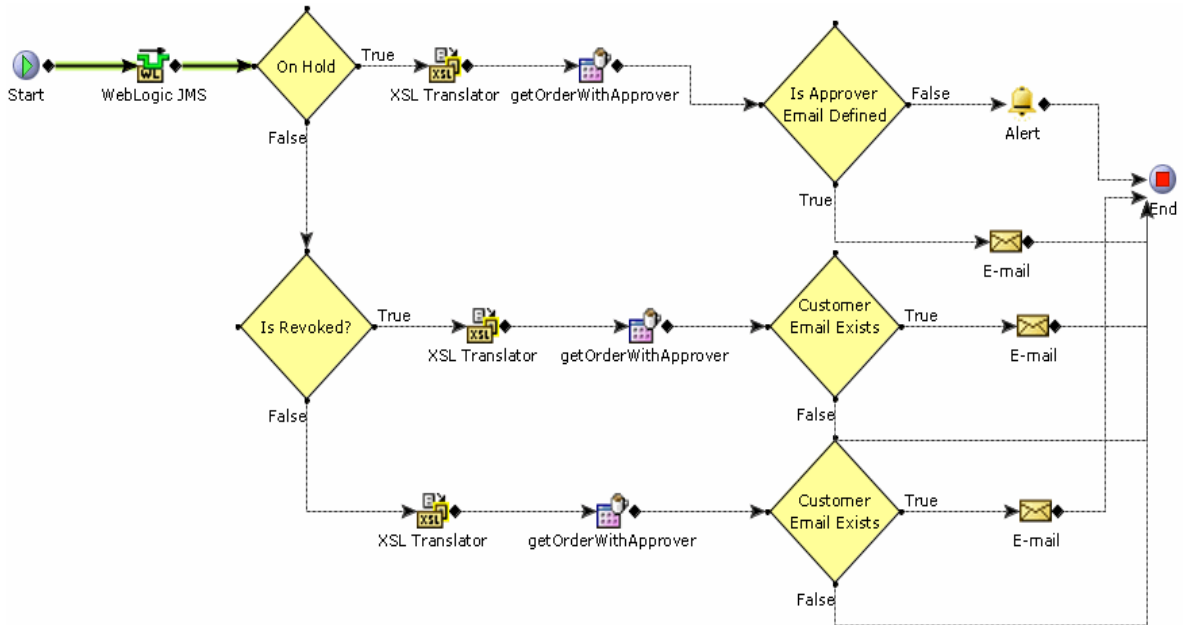
By default, Sterling Call Center and Sterling Store provide the following e-mail templates:

- `YCD_OrderApprovalEmail.xsl.sample`
- `YCD_OrderApprovalAcceptedEmail.xsl.sample`
- `YCD_OrderApprovalRejectedEmail.xsl.sample`

The XSL's must be renamed to `YCD_OrderApprovalEmail.xsl`, `YCD_OrderApprovalAcceptedEmail.xsl`, and `YCD_OrderApprovalRejectedEmail.xsl` respectively, customized as needed, and their path and names should be specified in the Body Template of the E-mail Sender properties.

A sample implementation of the service that is used to generate an e-mail or an alert when an order is on hold pending approval is illustrated in [Figure 4–18](#).

Figure 4–18 The YCD\_Order\_Approval\_Email\_8.5 Service



The custom API checks to verify whether a proxy approver exists for the approver of the order. If a proxy approver exists for the approver of the order, the `getCustomerContactList` API is called to fetch the details of the customer contact who has been set as the approver of the order. The service passes through a condition that checks whether the approver of the order has an e-mail address.

- If the approver of the order has an e-mail address, an e-mail is sent to both the approver of the order and the customer to whom the order belongs.
- If the approver of the order does not have an e-mail address, an alert is generated to indicate that the order is on hold pending approval and to inform the CSR to contact the approver of the order.

If the approver of the order has an e-mail address, e-mails will be sent in the following situations:

- After a customer places an order, if the order is put on hold pending approval, an e-mail will be sent to the approver of the

order or the proxy approver and the customer to whom the order belongs to inform them that the order has been put on hold pending approval.

- When the approver approves or rejects the hold, an e-mail will be sent to the approver of the order and the customer to whom the order belongs to inform them that the order has been either approved or rejected.
- If, after the approver of the order approves the hold, the order is put on hold pending approval from a second approver, an e-mail will be sent to the new approver and the customer to whom the order belongs to inform them that the order is on hold pending approval.

## 4.11 Payment Inquiry

Sometimes, customers may want to inquire about the payment details pertaining to an order. For example, a customer may want to inquire about the payment methods or refunds pertaining to an order.

### 4.11.1 Solution

The Payment Inquiry task enables users to view the payment information pertaining to an order, such as the order total, refunds, and pending charges. The `processOrderCollection` API is called to retrieve the payment details.

The user can also view the invoice details from the Payment Inquiry screen. The `getCompleteOrderInvoiceDetails` API is called to display the Invoice Details screen.

The payment status generated by the Sterling Call Center and Sterling Store application may be different than the payment status that you want to display to customers. You can implement the `YFSGetOverallStatusUE` user exit to return the `PaymentStatus` attribute, which is used to display the payment status in the Payment Inquiry screen of Sterling Call Center and Sterling Store. For more information about the `YFSGetOverallStatusUE` user exit, see the corresponding section in the *Sterling Call Center and Sterling Store: Javadocs*.

### 4.11.2 End-User Impact

If the YFSGetOverallStatusUE user exit is implemented, and if this user exit returns the `PaymentStatus` attribute, the user will be able to view the Payment Status field in the Payment Inquiry screen.

### 4.11.3 Implementation

This section explains the configurations for the Payment Inquiry task.

- Implement the YFSGetOverallStatusUE user exit to return the `PaymentStatus` attribute.
- This task is permission-controlled. You can assign the necessary permissions to the corresponding user groups.

### 4.11.4 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides the permission to view payment information to all the users.

## 4.12 Real-Time Credit Card Authorization

When confirming a draft order, you may want to process real-time credit card authorizations instead of batch processing orders. For instance, if an authorization fails due to an invalid card number, expired date, or invalid billing address, the CSR can inform the customer immediately and correct the errors. The CSR can also suggest that the customer use a different card. In order to authorize credit card in real-time, the CVV number must be captured.

### 4.12.1 Solution

Sterling Call Center and Sterling Store provides the functionality to process real-time credit card authorizations when confirming a draft order. This section explains the real-time credit card authorization process and the solution offered by Sterling Call Center and Sterling Store.

### Authorizing Credit Cards

Authorizing a credit card involves validating whether a credit card is valid and can be used to pay for an amount of money requested after adding it in the user interface. The logic to authorize a credit card is implemented as the last step in the order entry process, that is, while confirming a draft order in the Confirm Payment screen.

The `executePaymentTransactions` API is called to authorize the payment method. If the authorization succeeds, the `changeOrder` API is called to save the authorization on the order. If the authorization fails, the CSR can proceed by selecting the Override and Continue check box. If this check box is selected, the payment information and the failed authorization details are saved on confirmation of the order.

If multiple credit cards are added to a single order, the CSR must enter the charge amount on each card. The `executePaymentTransactions` API is called to authorize each credit card. If the authorization succeeds for all the payment methods, the `changeOrder` API is called to save the authorizations on the order.

Using the Advanced Credit Card Rules, you can configure Advanced Credit Card Rules to authorize a credit card before saving it to an order. If you have configured any payment rules that do not require authorization, do not enable this rule. For more information about configuring advanced credit card rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. If this rule is enabled, the CVV number is mandatory for all payment methods of CREDIT CARD payment type group. Real-time credit card authorizations are not available for any modification tasks.

### Handling Authorization Failures

If the authorization for any payment method fails, an appropriate error message is displayed. The payment method is displayed in red, highlighting the failed payment method. The CSR will then be prompted to edit the payment methods.

When an order has two payment methods of CREDIT CARD payment type group, it is possible that the first credit card is successfully authorized, but not the second card. In such a situation, if the CSR attempts to add the amount from the failed authorization to the credit card that has already been authorized, the additional amount is not authorized in real time, but will be picked up by the payment processing agents.

## Handling CVV Numbers

When confirming a draft order, the CSR must enter the customer's payment information and CVV number in the Add Payment Method screen. If the payment method is already defined, but not authorized, and if real-time credit card authorization is enabled, the CSR can select the corresponding payment method. The CSR will then be prompted to enter only the CVV number. The CVV number corresponds to the `PaymentReference1` attribute. This attribute is passed to the `executePaymentTransaction` API, and the CVV number is validated. The CVV number will not be saved on an order.

In Sterling Call Center, Release 9.0, and Sterling Store, Release 9.0, a `SecureAuthenticationCode` attribute, which corresponds to the CVV number entered by the user, is introduced. If you have configured SSDCS, the CVV number will correspond to the `SecureAuthenticationCode` attribute instead of the `PaymentReference1` attribute. This ensures that the CVV number is not saved on the order or in any log files.

To hide the CVV number from client-side log files, disable the log functionality, which the CSR will be using, from the client applications. This can be accomplished by not defining a log location in the `com.ini` file.

It is recommended that the RCP client components are stored in protected file folders in order to prevent the program or property file from being tampered. As further assurance, use a one-way hash, for example, SHA-256, to validate the programs and property files during startup. Additionally, a filter can be provided to the log files to remove sensitive data from the traces generated by the Sterling Call Center and Sterling Store clients. This can be done using the `logfilter.properties_ycd_ext` file provided by the Sterling Call Center and Sterling Store applications. Out-of-the-box, the `SecureAuthenticationCode`, `CreditCardNo`, `Password`, and `SVCNo` attributes are filtered to remove any sensitive data.

### 4.12.2 End-User Impact

If real-time authorization is enabled, the CVV code is mandatory for credit cards in the Confirm Payment screen.

### 4.12.3 Implementation

This section explains the configurations for real-time credit card authorization.

- Sterling Call Center and Sterling Store allows you to configure advanced credit card rules to process real-time credit card authorizations when confirming a draft order. For more information about advanced credit card rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Implement the YFSCollectionCreditCardUE user exit to handle both encrypted and unencrypted credit card numbers.
- To facilitate the return of user-friendly error messages whenever an error occurs, implement the YFSCollectionCreditCardUE user exit in such a way that it returns a localizable message in the internalReturnMessage attribute of the output. In order to localize these messages, add them to the extension resource bundle that will be picked up by the application. For more information about localizing resource bundles, refer to the *Selling and Fulfillment Foundation: Localization Guide*.

---

**Note:** If the payment system is unavailable, or if an error occurs, the YFSCollectionCreditCardUE user exit does not return any message. The application then uses the default error message bundle entry Authorization\_Failed\_No\_Message\_Was\_Returned to display an error message on the screen.

---

### 4.12.4 Reference Implementation

None.

## 4.13 Return Order Capture

Users need to enter return orders with maximum flexibility. They might need to create a return from several sales orders, allow a customer to return an item without a receipt, note return reasons, create exchanges, and set up appointments for pickup.



### 4.13.1 Solution

Sterling Call Center and Sterling Store provides a returns interface that enables CSRs to manage all aspects of a return process. Users who have access to multiple enterprises can select the enterprise for which a return order has to be created. A CSR can perform the following using the Create Return task:

- Create a return from single or multiple sales orders.
- Create a return for a customer who has no order in the system or has lost a receipt (if the enterprise allows this).
- Create a return for a customer who has received a wrong item or an item that was not ordered.
- Enter a variety of return reasons.
- Avoid the high shipping and handling costs of customer returns by exercising configurable appeasement and price-matching options.
- Create exchanges.
- Change an item's pickup location (if it is different from the original delivery address).
- Allow a customer to keep an item if returning it is not cost-effective for the company.
- Select return appointments and consolidate them with exchange appointments, ensuring that the item can be uninstalled and a new item can be installed in the same appointment.
- Create a return based on the container, because a customer may want to return an unopened or damaged package or shipment container.

This section contains a description of the solutions that the Sterling Call Center and Sterling Store provides for creating return orders.

#### 4.13.1.1 Return Line Selection

Sterling Call Center and Sterling Store provides a variety of ways in which a user can search for a line to return or exchange:

- By Order - Users can locate an existing order and find lines to return.
- By Container - Users can search for an order by entering the shipping container tracking number.

- By Items Without an Order (as long as the enterprise allows this) - Users can allow customers to return items not associated with an order.
- By Extra Item - Users can create a return for an item that a customer received by mistake.

For more information about return line selection, see [Section 4.15.1, "Return Line Selection"](#).

### **4.13.1.2 Identifying the Person Returning an Item.**

Sterling Call Center and Sterling Store enables a user to indicate whether an item is being returned by the customer who originally purchased it or by the recipient who received the item as a gift. If the item is being returned by the gift recipient, the application prompts the user to enter the customer identification information. For more information about customer identification, see [Identifying a Customer](#).

### **4.13.1.3 Adding Lines to a Return**

Sterling Call Center and Sterling Store provides a variety of methods to add lines to a return order. For more information, see [Section 4.15.1, "Return Line Selection"](#).

### **4.13.1.4 Validating Lines and Addresses**

After lines have been selected for return and added to the return order, Sterling Call Center and Sterling Store validates the lines and addresses before displaying the Return reason screen. For more information, see [Section 4.15.1, "Return Line Selection"](#).

### **4.13.1.5 Entering Return Reason**

Sterling Call Center and Sterling Store provides a Return Reason screen that enables users to record information such as the reason for the return and whether items are being returned or exchanged. Users may want to add a service line or perform return avoidance measures such as price match and appeasement. They may also want to override the return policy for that item. For more information, see [Section 4.15.2, "Entering Return Reason"](#).

#### 4.13.1.6 Return Reason Rule

A return reason rule for returns can be configured in the common code. For more information about defining return reasons, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.13.1.7 Overriding the Return Policy

For each line that is selected for return, Sterling Call Center and Sterling Store displays the return policy. Users can override this return policy. This is permission controlled. For more information, see [Section 4.15.2, "Entering Return Reason"](#).

#### 4.13.1.8 Customer Identification Required

When creating a return, customer information for the return may or may not be available. Typically, when a return is created for an existing sales order, the corresponding customer information is picked up from the sales order and used for the return order. However, if a return is created for Items Without An Order or Extra Items, the customer information should be entered by the end user.

If the customer identification step is required in the return creation process, you can configure the Customer Identification screen to be displayed either before or after the Add Items to Exchange screen.

For more information about configuring the Customer Identification screen, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.13.1.9 Is Customer Info Present Rule

This rule determines whether the return order header contains all the customer information, and if not, it displays the appropriate customer search screens:

- Customer Identification
- Customer Search
- Advanced Customer Search

On a return or an exchange order, the header address is copied to every line that does not already have an address.

### 4.13.1.10 Modifying an Exchange

The Sterling Call Center enables users to modify exchange orders, specify the change quantity and Unit of Measure (UOM) on exchange lines, override exchange prices, and select fulfillment options. For more information, see [Section 4.15.3, "Modifying an Exchange"](#).

### 4.13.1.11 Default Exchange Type Rules

Two rules exist for the default exchange type determine permission control. For more information about the Default Exchange Type Rule for Permission Control and the Default Exchange Type Rule for Change, see [Section 4.15.3, "Modifying an Exchange"](#).

### 4.13.1.12 Needs Appointments Rule

This rule looks at the exchange order and searches for lines that require a work order to be created for them. This rule looks at a model for exchange work order and a flag in the wizard for return work order. If either one has a work order, the appointment flow is run.

### 4.13.1.13 Address Info Needed Rule

This rule copies the address from the return to the exchange order when the address is already available for the return order. It also copies the address to any line on the return order or exchange order that does not have an address.

### 4.13.1.14 Create Reservation Rule

If a user specifies an exchange, this rule reserves the item so that the inventory is not consumed by another customer. See [Section 4.9.25, "Reservations"](#) for more information.

### 4.13.1.15 Exchange Fulfillment Options

The Sterling Call Center and Sterling Store provides the Exchange Fulfillment Options that enable users to change the fulfillment method, the pickup location, and the shipping and delivery addresses on exchange orders. For more information, see [Section 4.15.4, "Exchange Fulfillment Options"](#)

#### **4.13.1.16 Needs Appointment Rule**

The Sterling Call Center and Sterling Store provides an appointment rule to determine when work orders are necessary for return orders. This rule looks at a model for exchange work orders and a flag in the wizard for return work orders. If either one has a work order, the appointment flow is run.

#### **4.13.1.17 Is Customer Info Present Rule**

This rule determines whether enough customer information is present to complete a return order, and if not, it displays the appropriate customer search screens:

- Customer Identification
- Customer Search
- Advanced Customer Search

#### **4.13.1.18 Address Info Needed Rule**

This rule copies address information to return and exchange orders. It also copies the address to any line on the return or exchange order that does not have an address.

#### **4.13.1.19 Create Reservation Rule**

Sterling Call Center and Sterling Store provides a rule that reserves items so that the inventory is not consumed by another customer.

#### **4.13.1.20 Fulfillment Summary**

Sterling Call Center and Sterling Store enables users to specify and view the shipping, pickup, and delivery methods for return and exchange orders. For more information, see [Section 4.15.5, "Fulfillment Summary"](#).

#### **4.13.1.21 Select Appointment**

Sterling Call Center and Sterling Store provides users with the ability to manage and view return order appointments. For more information, see [Section 4.15.6, "Select Appointment"](#).

### 4.13.1.22 Payment Summary

Users can view and change payment details on the Payment Summary screen. For more information, see [Section 4.15.7, "Payment Summary"](#).

## 4.13.2 End-User Impact

This section explains the end-user impact for return order capture.

### 4.13.2.1 Line Selection

If the Reship rule is set, an icon signifying reshipped lines is displayed. If the Reship rule is not set, a check box is displayed at the top of the screen which, if selected, displays reship details.

If the rule that allows the return of items without an order is set, the Items Without an Order tab is displayed and customers can return an item without an order. If the rule is not set, the Items Without an Order tab is not displayed and customers can return an item only if they have a receipt or the item can be found on an order.

### 4.13.2.2 Exchange Fulfillment Options

- Disabling Availability Checks - The user is not shown the availability details in any store on the Add Multiple Lines, Order Entry, Change Fulfillment Options, Change Order Addresses, and Item Inquiry screens.
- Changing Pickup Lines - The user is allowed to modify the order lines that have the pickup fulfillment method even after the status is changed to Released.
- Changing Delivery Lines - The user is allowed to modify the order lines that have the delivery fulfillment method even after the status is changed to Released.

## 4.13.3 Implementation

This section explains implementation considerations for return order capture.

Return entry is a permission-controlled task. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### 4.13.3.1 Line Selection

To display an icon identifying reshipped lines, turn on the Reship configuration. If the Reship rule is turned off and reship lines exist, a check box displays reshipped details. For more information about configuring the Reship rule, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

If your enterprise does not wish to allow returns for items not associated with an order, turn off the rule that allows returning items without an order. In this case, the Items Without an Order tab would not be displayed. For more information about configuring the return entry rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.13.3.2 Return Reason

The rule for the Override Return Policy note type notes pop-up window is configurable.

The reasons for returning an item can be configured in the RETURN\_REASON common code.

For more information about defining return reasons, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.13.3.3 Modify an Exchange

You can configure two exchange type rules for the Add Exchange Items screen that define the enterprise-specific value for the default exchange type and that specify user access to this feature. For more information about defining exchange types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.13.3.4 Exchange Fulfillment Options

This section details the rules to be configured to change the fulfillment options.

- **Disabling Availability Checks** - If the Prevent Initial Availability Checks During Order Entry and Order Modifications rule is enabled, the inventory availability check is not performed and the calls to the getFulfillmentOptionsForLines, findInventory, or any derivation APIs are eliminated. This aids in better performance and pace when capturing the order. For more information about availability checks

and reservation options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- **Changing Pickup Lines** - To modify the pickup lines that are in the Released status or above, check the Allow Modification of Pickup Dates and Store for Pickup Lines after Release box for the Enterprise. For more information, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

Changing Delivery Lines - To modify the delivery lines that are in the Released status or later, check the Allow Modification of Fulfillment Options for Delivery Lines After Release box for the Enterprise.

Notes entered for this task are saved on the order with YCD\_DELIVERY\_OPTIONS note type. For more information, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 4.13.3.5 Fulfillment Summary

For returns, the UI is painted with a summary panel for each delivery method. Delivery methods are suggested by Sterling Call Center and Sterling Store and can be customized through the YCDgetReturnMethodUE. The delivery methods are:

- Return Pickup
- Return Shipping
- Items can be kept

The YCDgetReturnMethodUE applies a standard user exit template and passes the result to the user exit. The default implementation is not provided for this user exit. If the user selects the Ship To Return Center return method, the RETURN\_FULFILLMENT fulfillment type is used to determine the correct return center to receive the item.

### 4.13.3.6 Status Changes to a Return or Exchange Order

Enable notes for all status modifications of a return order or an exchange order. This is how Sterling Call Center and Sterling Store records status changes. For more information about enabling notes for all status modifications, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.



#### 4.13.3.7 Select Appointment

This section explains the configurations for the change service appointment task.

- To change the service appointment calendar view, configure the following rules:
  - Calendar Display
  - Default Appointment Calendar View
  - Capacity Information

For more information about configuring the service appointment calendar view, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- Sterling Call Center and Sterling Store allows you to configure the order statuses for which changing service appointment is allowed. For more information about configuring the order modification statuses, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 4.13.4 Reference Implementation

This section explains reference implementation information for return order capture.

#### 4.13.4.1 Return Reason

The reference implementation ships the following return reasons:

- Cargo Damaged
- Change of Mind
- Damaged Items
- Loose Parts
- Lower Price Found
- Missing Parts
- Other

For more information about defining return reasons, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 4.13.4.2 Exchange Fulfillment Options

This section explains the reference implementation provided for the following tasks:

- Disabling Availability Checks - The Prevent Initial Availability Checks During Order Entry and Order Modifications rule is defaulted to ensure that availability checks are performed.
- Changing Pickup Lines - The Allow Modification of Pickup Dates and Store for Pickup Lines after Release rule is defaulted to allow modification of lines after release.
- Changing Delivery Lines - The Allow Modification of Fulfillment Options for Delivery Lines after Release rule is defaulted to allow modification of lines after release.

The Allow Cancellation of Delivery Lines after Release rule is also defaulted to allow modification of lines after release.

### 4.13.4.3 Select Appointment

The section explains the reference implementation provided for the change service appointment task.

- An instruction type called 'DELIVERY' is provided for saving service appointment instructions.
- Sterling Call Center and Sterling Store provides permissions to change service appointments to all users as part of reference implementation.

## 4.14 Return Order Inquiry

Sometimes customers may inquire information about return orders. For example, a customer may want to cancel a return or change a pickup appointment on the return order. In such situations, call center or store representatives need to search for the appropriate return order and then perform the necessary actions.

### 4.14.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which enables users to search for return orders.

If a user has access to multiple enterprises, the Enterprise drop-down list is displayed in the Return Order Search Criteria panel of the Return Search screen. The `getOrganizationList` API is called to retrieve the list of enterprises. The `getOrderList` API is called to retrieve the list of orders that match the search criteria. The `getCompleteOrderDetails` API is called to display the details of the return order. The `getExceptionListForOrder` API is called to retrieve the list of alerts on the return order. For more information about alerts, see [Section 4.23, "Alert and Queue Management"](#).

The search results can be sorted in ascending or descending order based on following Order By options:

- Return Order Number
- Return Order Date
- Customer Last Name

### Pagination

Sterling Call Center and Sterling Store supports smart retrieval of record sets in the Return Search screen. You can configure if history records need to be automatically fetched while retrieving records. For more information about configuring pagination rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

For more information about pagination, see [Section 4.22, "Pagination"](#).

## 4.14.2 End-User Impact

Based on the configuration to search for history orders, the end user will see history orders or recent orders in the search results.

## 4.14.3 Implementation

This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

## 4.14.4 Reference Implementation

Sterling Call Center and Sterling Store provides permissions to search for return orders to all users as part of reference implementation.

## 4.15 Return Order Maintenance

Sterling Call Center and Sterling Store provides the following return order maintenance features:

- [Return Line Selection](#)
- [Selecting Lines to Return](#)
- [Entering Return Reason](#)
- [Modifying an Exchange](#)
- [Exchange Fulfillment Options](#)
- [Fulfillment Summary](#)
- [Select Appointment](#)
- [Payment Summary](#)
- [Create Exchange Order](#)
- [Report Extra Items](#)
- [Report Wrong Items](#)
- [Report Unexpected Items](#)
- [Issue Refund Now](#)
- [Change Return Address](#)
- [Change Service Appointment](#)
- [Change Return Method](#)
- [Cancel Return Order](#)
- [Payment Confirmation](#)

### 4.15.1 Return Line Selection

Users need to be able to select one or more lines of items to add to a return order from Order Summary screen or from the Task Panel.

#### 4.15.1.1 Solution

Users can create returns using the following criteria:

- [By Order](#)

- [By Container](#)
- [By Items Without an Order](#) (as long as the enterprise allows this)
- [By Extra Item](#)

### **By Order**

The most common way to create a return order is from a previously placed order. After an order is chosen, either the entire order or select lines from the order can be designated for return. A user can also include lines from multiple orders on the same return.

By selecting the Order tab on the Create Return screen, a user can either enter an order number or search for orders by using the Order Search functionality described in [Section 4.7, "Order Inquiry"](#).

When a user clicks the Order tab and searches for the order or enters the order number, a call is made to the `getOrderLineswithTransactionalQuantity` API to get the order information. When the order is added, the available quantity for return shows only the quantity that can be returned, depending upon the modification rules and the line status of the quantity.

Lines that are not in a returnable status cannot be selected by the user for return. This occurs when the order status shows that the item has not been shipped or has already been returned.

Notes entered for this task are saved on the order with YCD\_RETURN\_INFO note type.

### **By Container**

A user may need to create a return order based on shipping containers. This requires searching for the container's tracking number. Searching for orders by entering the tracking number enables users to find orders and lines to return.

In the Tracking # tab, when the user enters the tracking number and the Add button, a call is made to `getShipmentContainerList`. If the tracking number matches with an existing tracking number, all the orders (complete or partial) that are part of the searched container are added to the return creation in individual panels, with each line showing only the quantities that are available for return.

For each order, the `getOrderLinesWithTransactionalQuantity` API is called. Each order is displayed in its own collapsible panel. When all the orders have been retrieved, a check takes place for the lines that were shipped in the container. This sets the quantity based on the quantity in the container.

### **By Items Without an Order**

Items that are not associated with an order can also be returned, unless the enterprise has turned off the rule that allows returns without an order. In that case, the Items Without an Order tab will not be displayed.

By selecting the By Items Without an Order tab, a user can create a return for an item that is not associated with an order. When a user scans or enters the Item ID, the `getCompleteItemList` API is called to get the item information and verify whether the ID is valid.

When `getCompleteItemList` returns with only one result, a collapsible panel is created, showing the items not associated with an order. A Remove Line hyperlink enables users to remove the item if they entered it by mistake. If `getCompleteItemList` returns with no result or with multiple results, an error stating that the order could not be identified is displayed.

### **By Extra Item**

On the Extra Items tab, a user can create a return for an item that a customer received by mistake. Because these extra items were not purchased, refunds are not issued.

When a user selects the Extra Item tab and either scans or enters the Item ID, the `getCompleteItemList` API is called to get the item information and verify whether the ID is valid. When `getCompleteItemList` returns, it puts the item in a collapsible panel that shows extra items to return. The user can also remove the line if the ID that was entered is incorrect.

If no results or multiple results are returned, an error is displayed.

Sterling Call Center and Sterling Store provides a line type `YCD_EXTRANEIOUS` which is used as the line type when creating an order line for extra items.

## Selecting Lines to Return

After a list containing one or more lines is identified for return, the lines that are added to the return order must have the check box selected to be returned. Blind and extra items displayed will always be returned unless they are removed from the UI.

## Selecting by Container

If the order has at least one container, a link is shown at the order level which, when clicked, calls `getCompleteShipmentDetails` and opens a container selection pop-up window. This window displays all the containers on the order along with the items in each container. When a user selects one or more containers and clicks Confirm, all the lines in the container are automatically selected and the quantities increased according to the amount in each container. This creates new, collapsible panels for each order if the container has lines from multiple orders.

## Stop Delivery Request

If a line has a stop delivery request placed on it, an icon indicates that the status of this request is open until it is completed, at which point the icon is not displayed and the returnable quantity is reduced.

## Open Box

If the line on the original sales order was an open box, the open box icon is displayed only to provide information to the CSR. If a customer requests an exchange for an open box, the exchange item will be a regular (non-open box) item.

## Gift Lines

If the line on the original sales order was a gift, the gift icon is displayed only for CSR informational purposes.

## Bundles

Only entire bundles are returnable, so a return activity can take place only in the context of the entire bundle. For this reason, only parent lines are displayed in the bundle. For more information about bundles, see [Section 4.2, "Configured Items"](#).

### Reship Lines

If the reship configuration rule is turned on, which is the default, a column displays an icon identifying the reshipped lines. If the reship rule is turned off and reship lines exist, a check box is displayed for those lines. Selecting the check box displays reship details.

Reshipped lines are available for return, but the price should be \$0.00 in cases where customers receive items they have asked to be reshipped.

To ensure that refunds are not issued for extra items, define pricing user exits with the line type marked as extra and the price set to \$0.00. The line type is configurable using the rule for extraneous line types.

### Collecting and Validating Return Order Information

After a user selects items for return, the system must collect and validate the necessary information to create a return order.

When the user clicks the Next button after line selection, a validation occurs to verify whether there is at least one line on the screen with a check box selected. If yes, input is constructed for the createOrder API, but it is not called until the return policy is checked and the user has entered a return reason, as described in [Section 4.15.2, "Entering Return Reason"](#).

The shipTo and billTo addresses on the sales order lines are copied to the new return line. Also, lines are always copied (except for gift lines). If all the header addresses and customer information are the same, this is copied as well. If two header addresses or customer information differ from one another, no copy takes place. This information is captured later in the flow.

No address is populated on the return order if the item is being returned by a gift recipient.

When these processes are complete, a model is set so that the Return Reason screen can use it. This model is read, modified, and saved to ensure that the correct information is used.

Missing addresses are determined later in the flow.



#### 4.15.1.2 End User Impact

If the Reship rule is set, an icon signifying reshipped lines is displayed. If the Reship rule is not set, a check box is displayed at the top of the screen which, if selected, displays reship details.

If the rule that allows the return of items without an order is set, the Items Without an Order tab is displayed and customers can return an item without an order. If the rule is not set, the Items Without an Order tab is not displayed and customers can return an item only if they have a receipt or if the item can be found on an order.

#### 4.15.1.3 Implementation

To display an icon identifying reshipped lines, turn on the Reship configuration. If the Reship rule is turned off and reship lines exist, a check box displays the reshipped details. For more information about configuring the Reship rule, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

If your enterprise does not want to allow returns for items not associated with an order, turn off the rule that allows the return of items without an order. In this case, the Items Without an Order tab will not be displayed. For more information, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.15.1.4 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides pricing for items in the master data of the catalog.

### 4.15.2 Entering Return Reason

Users need to record information such as the reason for return and whether items are being exchanged. They may want to add a service line or perform return avoidance through price match and appeasement.

#### 4.15.2.1 Solution

After all the lines that have to be put on the order have been selected, the Return Reason screen enables users to perform the actions described in this section.

### Selecting a Return Reason

Users must select a return reason for an item to be returned or exchanged. The Apply to All Items button applies the return reason to the entire return order.

For each line that is selected for return on the Return Reason screen, text is displayed at the top of the screen showing the return policy for that item. If the item cannot be returned, a user can optionally select a link to remove that line from the return order. Alternatively, the user can override the return policy by clicking the Override Return Policy button. If the user does not click the Override Return Policy button, the line will not be added to the return, but will be displayed when the user comes back to the screen.

On the Return Reason screen, there may be some lines that can be returned along with lines that cannot be returned. A panel displays the lines that cannot be returned. This information is maintained throughout the order return process and is always displayed on the Return Reason screen as long as the same instance of the wizard exists and the user has not closed the wizard. If the user comes back to this screen after going forward in the flow, the same information will still be here, unless it has been overridden.

### Return Policy

When overriding the return policy is allowed, it is determined by the `getReturnPolicy` API and the `YCDGetReturnPolicyUE`. If the item cannot be returned, the user can override the return policy by clicking the Override Return Policy button.

An Override Return Policy notes pop-up window requires the user to enter a reason for the override before continuing. The rule for overriding return policy note type is configurable.

### Creating Exchanges

The Return Reason screen has two check boxes for exchanges:

- **Exchange Returned Items** (check box at the top of the screen:)

When a user selects this check box, it signifies that an exchange order will be created when the user clicks Next. If none of the exchanges is for the same item, the exchange order is created with

no lines. Exchange lines can be specified on the next screen.

- **Exchange for the same item** (check box in the panel on the right:)

When the user selects this check box, the Exchange Returned Items check box at the top of the screen is also automatically selected, and an exchange order is populated automatically with details of the item that has been selected for exchange. These details include the quantity and price for an even exchange on the item. To exchange all the items on the Return Reason screen for the same item, click the Apply to All Items button.

### Add Return Service

The Add Return Service link creates a pop-up window (called by the getCompleteItemList API) with return services available for the selected item, if applicable. If the service is configured to specify the quantity at the service line, the quantity field is editable. An added return service can be associated to only one product line.

### Price Match This Item

The link for Price Match This Item launches the price match flow as a sub-task. If a price match is performed on a line in the Return Reason screen, that line is removed from the screen and is no longer considered a part of the return order. For more information about the price match process and the solution offered by Sterling Call Center and Sterling Store, see [Section 4.9.8, "Price Match"](#).

### Create Appeasement

The Create Appeasement link launches the appeasement flow as a sub-task. If a customer appeasement is performed on a line in the Return Reason screen, that line is removed from the screen and is no longer considered a part of the return order. For more information about the customer appeasement process and the solution offered by Sterling Call Center and Sterling Store, see [Section 4.9.7, "Customer Appeasement"](#).

### 4.15.2.2 End User Impact

None.

### 4.15.2.3 Implementation

The rule for the Override Return Policy note type pop-up window is configurable.

The reasons for returning an item can be configured in the RETURN\_REASON common code.

For more information about defining return reasons, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 4.15.2.4 Reference Implementation

You can configure reasons such as:

- Cargo Damaged
- Change of Mind
- Damaged Items
- Loose Parts
- Lower Price Found
- Missing Parts
- Other

For more information about defining return reasons, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

## 4.15.3 Modifying an Exchange

Users need to specify many types of exchange details.

### 4.15.3.1 Solution

After the user enters a return reason in the Return Reason screen and specifies an exchange, the Add Exchange Items screen is displayed.

In this screen, users can specify the change quantity and Unit of Measure (UOM) on exchange lines, override the price on an exchange line, select the fulfillment option for the exchange line, and change the exchange type.

All other aspects of creating an exchange order are the same as in order capture, which is described in detail in [Section 4.6, "Order Capture"](#).

### Default Exchange Type Rule for Permission Control

Permission control can be configured for the default exchange type, which restricts unauthorized users from changing the default exchange type. The rule is enterprise-specific and has three values:

- Regular
- Advanced
- Prepaid

### Default Exchange Type Rule for Change

An additional rule enables your organization to decide whether you want to allow the default exchange type value to be changed at all. If this rule is disabled, only the link is hidden; the exchange type description is still displayed.

#### 4.15.3.2 End User Impact

None.

#### 4.15.3.3 Implementation

You can configure two exchange type rules for the Add Exchange Items screen that define the enterprise-specific value for the default exchange type and that specify user access to this feature. For more information about configuring the exchange type rules, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.15.3.4 Reference Implementation

None.

## 4.15.4 Exchange Fulfillment Options

Users need to be able to view and change fulfillment details on a return order.

The Exchange Fulfillment Options screen enables a user to change the fulfillment method on an exchange order, change the shipping/delivery address on the exchange lines, and change the pickup location on the exchange lines.

For more information about the Exchange Fulfillment Options screen, see [Section 4.9.19, "Change Fulfillment Options"](#).

### 4.15.5 Fulfillment Summary

Users need to specify and view the shipping, pickup, and delivery methods for return and exchange lines.

See [Section 4.15.15, "Change Return Method"](#) for a complete description of the Fulfillment Summary.

### 4.15.6 Select Appointment

Users need to be able to manage and view return-order appointments.

On the Appointment screen, users can view return lines on the exchange or return order and see when they will be delivered. In addition, users can change appointments and add delivery instructions.

Consolidation occurs automatically.

The Select Appointment screen in the return and exchange process is exactly the same as it is in order capture. For more information, see [Section 4.15.14, "Change Service Appointment"](#).

### 4.15.7 Payment Summary

Users need to view and change payment details for return orders and exchange orders.

#### 4.15.7.1 Solution

On the Payment Summary screen, a user can view payment details for returns and exchanges and view header, line charges, and taxes for returns and exchanges, as well as add a note to the return.

In the top left corner of the Payment Summary screen, the Overall Payment Details are displayed. This panel includes Credits from Returns, New Charges from Exchanges, if any, and the Net Balance of the customer refund amount or the amount due from the customer.

If your enterprise implements its own or a third-party payment system, the Sterling Call Center and Sterling Store Payment Summary screen is

not displayed. This option is set through a rule to enter payments externally.

### **Return Panel**

If there is no exchange order, only the Return Order Totals panel on the lower left lists Refunded Charges, Taxes, and Fees for items being returned. For Sterling Call Center and Sterling Store, agents perform these tasks in the background. The payment process and how customers get refunds is not handled by the Sterling Call Center and Sterling Store UI, except to display the amount of money the customer will be refunded.

Through the Return Order Totals panel, users can access pop-up windows and screens that enable them to add or modify return fees, charge categories, charge names, charge amounts, and notes, if necessary. These changes are reflected in the totals on the Overall Payment Details panel in the Payment Summary screen.

### **Exchange Panel**

If there is an exchange, the Exchange Order Totals panel on the lower right lists charges, taxes, and discounts for items being exchanged. Through the Exchange panel, users can access pop-up windows and screens that enable them to add or modify exchange fees, add charges, charge names, charge amounts, and notes, if necessary. These changes are reflected in the totals on the Overall Payment Details panel in the Payment Summary screen.

### **Payment Methods Panel**

Users can enter a payment method in the Payment Methods panel in the top right corner of the Payment Summary screen. Through the Payment Methods panel, users can also supply Payment Type information and the amount to be paid, as well as apply coupons and promotions.

For information about typical payment processing, see [Section 4.10, "Payment Processing"](#) contains a complete description of the entire payment processing interface.

Sterling Call Center and Sterling Store provides functionality to process real-time credit card authorizations while confirming draft orders. For more information about real-time credit card authorizations, see [Section 4.12, "Real-Time Credit Card Authorization"](#).

### **4.15.7.2 End User Impact**

None.

### **4.15.7.3 Implementation**

None.

### **4.15.7.4 Reference Implementation**

None.

## **4.15.8 Create Exchange Order**

Customer may want an exchange item for the items being returned. In such situations call center and store representatives may need to create return orders as requested by the customer.

### **4.15.8.1 Solution**

Sterling Call Center and Sterling Store provides a user interface task which enables users to create exchange order for the return order.

This section explains the Create Exchange process and the solution offered by Sterling Call Center and Sterling Store.

### **Adding Items To The Exchange Order**

The Add Items screen enables users to add lines to the exchange order. For more information about adding items to the exchange order, see [Section 4.15.3, "Modifying an Exchange"](#).

### **Exchange Fulfillment Options**

On this screen, users can view and modify the fulfillment options for the exchange order. For more information about exchange fulfillment options, see [Section 4.15.4, "Exchange Fulfillment Options"](#).

### **Fulfillment Summary**

The Fulfillment Summary screen provides the complete scenario of how the exchange order is configured and fulfilled. For more information about fulfillment summary, see [Section 4.15.5, "Fulfillment Summary"](#).



### Select Appointment

The Select Appointment screen enables users to create and modify appointments. In addition, it also allows users to enter and modify service instructions. For more information about selecting appointments, see [Section 4.15.6, "Select Appointment"](#).

### Payment Confirmation

The Payment Confirmation screen displays the summary of the payment impact on the order as a result of adding exchange items. For more information about viewing and confirming payment details, see [Section 4.15.17, "Payment Confirmation"](#).

#### 4.15.8.2 End User Impact

If an exchange order is already created for a return order and the status of the exchange order is `Created`, the user cannot create another exchange order for the corresponding return order. If the exchange order is in the `Draft` status, the Add Items screen is displayed, which enables users to add items to the exchange order.

#### 4.15.8.3 Implementation

This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### 4.15.8.4 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to create exchange order to all user groups.

### 4.15.9 Report Extra Items

Sometimes a customer may report that the he has received extra items in their shipment by mistake. For example, if a customer ordered a single shirt and has received two shirts instead. In such situations, the extra items can either be returned back to the enterprise or the CSR can allow the customer to keep the extra items depending on the value of the items.

### 4.15.9.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which enables users to report extra items.

This section explains the Report Extra Items process and the solution offered by Sterling Call Center and Sterling Store.

#### Identifying Extra Items And Creating A Return

The Extra Item Identification screen is provided for users to enter data about the extra item. In some instances, the customer and CSR may not be able to identify the specific item that was given to the customer. For example, the extra item may be a blue shirt that the customer is not able to describe well enough for the CSR to find the correct item ID. In this scenario, you can configure the system to allow the CSR to enter an "unknown" item in this screen. Since the application requires you to use a valid item ID, you can configure a temporary item ID to be saved on the return. In this case, the CSR can manually enter a description of the item in this screen. When the item is actually received at the warehouse, the correct item ID can be substituted on the return.

The `getCompleteItemList` is called to validate the item ID, when a product or service item ID or alternate item ID is entered or scanned. The `createOrder` API is called to create a return for the extra items. Sterling Call Center and Sterling Store provides a line type `YCD_EXTRANEIOUS` which is used as the line type when creating an order line for extra items.

#### Displaying Fulfillment Summary For Return Items

The Fulfillment Summary screen displays the suggest return method for the extra items reported. For more information about fulfillment summary, see [Section 4.15.5, "Fulfillment Summary"](#).

#### Creating Service Appointment For Return Items

If the user chooses to have the extra item picked up, the Return Service Appointment screen is displayed. For more information about creating service appointments, see [Section 4.15.14, "Change Service Appointment"](#).

### 4.15.9.2 End-User Impact

None.

#### 4.15.9.3 Implementation

This section explains the configurations for the return extra item task.

- Sterling Call Center and Sterling Store allows you to configure the line type for extraneous items and wrongly shipped items. For more information about configuring the line types for extra items, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure rules to allow unknown item returns and the item ID to be used for unknown item returns. For more information about allowing unknown item returns and configuring the item ID for unknown item returns, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### 4.15.9.4 Reference Implementation

This section explains the reference implementation provided as part of the report extra item task.

- Unknown item returns are allowed for Matrix Incorporated, Matrix Retail, and Matrix Business.
- The item ID for a Unknown item is WRONGITEM for Matrix Incorporated, Matrix Retail, and Matrix Business.
- The line type for the return item is set to YCD\_EXTRANEIOUS\_LINE\_TYPE for Matrix Incorporated, Matrix Retail, and Matrix Business.
- As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to report extra items to all user groups.

### 4.15.10 Report Wrong Items

Sometimes a customer may report that he received different items than what he ordered. For example, the customer may have ordered a black shirt, but received a blue shirt instead. In such situations you may need to reship the items that were ordered and also create a return for the wrong items that were delivered.

### 4.15.10.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which enables users to report wrong items.

This section explains the Report Wrong Items process and the solution offered by Sterling Call Center and Sterling Store.

### Reshipping Items

When a customer reports that he has received wrong items, the originally ordered items need to be reshipped. The `validateReship` API is called to verify if the item can be reshipped. If the items cannot be reshipped, the user can override reship validation only if the user is a supervisor. For more information about reshipping an item, see [Section 4.9.13, "Reship an Item"](#).

### Refunding The Amount For Unavailable Items

Sometimes, the ordered items may be out of stock, or the customer might not want the items. In such situations, the CSR may need to refund the amount for the originally ordered items. The Payment Confirmation screen displays the summary of the payment impact on the order as a result of the refund. For more information about payment summary, see [Section 4.15.17, "Payment Confirmation"](#).

### Identifying Wrong Items And Creating A Return

The Wrong Item Identification screen is provided for users to enter data about the wrong item. In some instances, the customer and CSR may not be able to identify the specific item that was given to the customer. For example, the wrong item may be a blue shirt that the customer is not able to describe well enough for the CSR to find the correct item ID. In this scenario, you can configure the system to allow the CSR to enter an "unknown" item in this screen. Since the application requires you to use a valid item ID, you can configure a temporary item ID to be saved on the return. In this case, the CSR can manually enter a description of the item in this screen. When the item is actually received at the warehouse, the correct item ID can be substituted on the return. The `createOrder` API is called to create a return for the wrong items.

## Modifying The Fulfillment Method Of The Return Items

The customer can choose to modify the fulfillment method of the return items. The changeOrder API is called to modify the fulfillment option for the wrong item. For more information about fulfillment summary, see [Section 4.15.5, "Fulfillment Summary"](#).

### 4.15.10.2 End-User Impact

None.

### 4.15.10.3 Implementation

This section explains the configurations for the return wrong item task.

- Sterling Call Center and Sterling Store allows you to configure the line type for extraneous items and wrongly shipped items. For more information about configuring the line types for wrong items, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure rules to allow unknown item returns and the item ID to be used for unknown item returns. For more information about allowing unknown item returns and configuring the item ID for unknown item returns, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

### 4.15.10.4 Reference Implementation

This section explains the reference implementation provided as part of the report wrong item task.

- Unknown item returns are allowed for Matrix Incorporated, Matrix Retail, and Matrix Business.
- The item ID for a Unknown item is WRONGITEM for Matrix Incorporated, Matrix Retail, and Matrix Business.
- The line type for the return item is set to YCD\_WRONGLY\_SHIPPED\_LINE\_TYPE for Matrix Incorporated, Matrix Retail, and Matrix Business.

- As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to report wrong items to all user groups.

### 4.15.11 Report Unexpected Items

Sometimes a customer may report that he did not order for any items and has received some items. For example, the shipping carrier may have dropped the shipment off at the wrong house. In such situations you may need to create a customer and then create a return for the wrongly shipped items.

#### 4.15.11.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which enables users to report unexpected items.

This section explains the Report Unexpected Items process and the solution offered by Sterling Call Center and Sterling Store.

#### Identifying Customers

When a customer reports that he had not ordered for any items, but has received some items, you need to search if the customer already exists. If the customer does not exist, you may need to create a new customer.

#### Identifying or Searching for a Customer

Sterling Call Center and Sterling Store provides the functionality to search for a customer, based on three options:

- [Advanced Customer Search](#)
- [Search based on address](#)
- [Basic Customer Search](#)

#### Advanced Customer Search

This screen is recommended if the user wants to report unexpected items for customers maintained by Sterling Call Center and Sterling Store. This screen provides a full set of features to search against your database. The Advanced Customer Search option enables you to use additional addresses and payment methods that have been associated to the customer. You must use the Advanced Customer Search if you want to access the customers created in the Sterling Call Center and Sterling

Store. All customer information is obtained through a `getCustomerList` API call.

When searching a customer master database, you can enable the search on business customers by configuring the Business Customers Enabled rule. You can also configure the default customer type when searching for a customer. For more information about configuring the display of customer identification screen, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

The user can enter a new address or modify an existing address in this screen. For more information about the address capture screen, see [Section 4.6.4.2, "Address Verification"](#).

When creating a customer from the Advanced Customer Search screen, the create customer wizard is loaded in a pop-up window. [Section 4.16, "Customer Creation"](#) contains more information about this task.

When a new address is added or modified using the Advanced Customer Search screen, the new address is saved against the selected customer record through a `manageCustomer` API call.

The Advanced Customer Search screen contains validations to ensure that the end user enters enough information for a search. If this screen is extended to contain additional fields, these validations are not performed. This enables you to create validations based on the fields that you added. For more information about customizing screens, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

### **Search based on address**

If you do not maintain a customer database, you can choose to display this screen which allows the user to search among the existing addresses in the system.

The `getPersonInfoList` API is called to retrieve the addresses for the search criteria. The `getConsumerDetails` API is used to retrieve the details of the consumer associated with the address. If a consumer corresponding to the address is not found, and if corporate users are enabled for the enterprise, the system attempts to find a corporate user associated with the address.

The user can enter a new address or modify an existing address in this screen. For more information about the address capture screen, see [Section 4.6.4.2, "Address Verification"](#).

The Address Entry screen contains validations to ensure that the end user enters enough information for a search. If this screen is extended to contain additional fields, these validations are not performed. This enables you to create validations based on the fields that you added. For more information about customizing screens, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

### Basic Customer Search

This screen is not recommended for new installations and is only provided for backward compatibility.

Sterling Call Center and Sterling Store provides a configuration to enable searches based on customer types. You can also configure the default customer type to be used when searching for a customer. For more information about configuring the customer types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

The `getCustomerList` API is used to retrieve the details of the business customer or consumer customer.

The user can enter a new address or modify an existing address in this screen. For more information about the address capture screen, see [Section 4.6.4.2, "Address Verification"](#).

The Basic Customer Search screen contains validations to ensure that the end user enters enough information for a search. If these screens are extended to contain additional fields, these validations are not performed. This enables you to create validations based on the fields that you added. For more information about customizing screens, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

### Identifying Unexpected Items And Creating A Return

The Unexpected Item Identification screen is provided for users to enter data about the unexpected item. In some instances, the customer and CSR may not be able to identify the specific item that was given to the customer. For example, the item may be a blue shirt that the customer is not able to describe well enough for the CSR to find the correct item ID. In this scenario, you can configure the system to allow the CSR to enter an "unknown" item in this screen. Since the application requires you to use a valid item ID, you can configure a temporary item ID to be saved on the return. In this case, the CSR can manually enter a description of



the item in this screen. When the item is actually received at the warehouse, the correct item ID can be substituted on the return.

The `getCompleteItemList` is called to validate the item ID, when a product or service item ID or alternate item ID is entered or scanned. The `createOrder` API is called to create a return for the unexpected items. Sterling Call Center and Sterling Store provides a line type `YCD_EXTRANEIOUS` which is used as the line type when creating an order line for extra items.

### Displaying Fulfillment Summary For Return Items

The Fulfillment Summary screen displays the suggested return method for the unexpected items reported. For more information about fulfillment summary, see [Section 4.15.5, "Fulfillment Summary"](#).

### Creating Service Appointment For Return Items

If the user chooses to have the unexpected item picked up, the Return Service Appointment screen is displayed. For more information about creating service appointments, see [Section 4.15.14, "Change Service Appointment"](#)

#### 4.15.11.2 End-User Impact

- The user will see either the Advanced Customer Search screen, Address Search screen or Basic Customer Search screen based on the configuration.
- Users can search for customers based on the customer type, if the configuration for customer type search is enabled and the default customer type during search is Business.

#### 4.15.11.3 Implementation

This section explains the configurations for the return unexpected item task.

- Sterling Call Center and Sterling Store allows you to configure rules to allow unknown item returns and the item ID to be used for unknown item returns. For more information about allowing unknown item returns and configuring the item ID for unknown item returns, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- Sterling Call Center and Sterling Store allows you to configure the type of the customer search screen that needs to be displayed. For more information about configuring the customer search screen, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

### 4.15.11.4 Reference Implementation

This section explains the reference implementation provided as part of the report wrong item task.

- Unknown item returns are allowed for Matrix Incorporated, Matrix Retail, and Matrix Business.
- The item ID for a Unknown item is WRONGITEM for Matrix Incorporated, Matrix Retail, and Matrix Business.
- The line type for the return item is set to YCD\_EXTRANEIOUS\_LINE\_TYPE for Matrix Incorporated, Matrix Retail, and Matrix Business.
- As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to report unexpected items to all user groups.

## 4.15.12 Issue Refund Now

A customer may want the refund to be issued immediately. For example, if the return method of an item is pick up and no representatives from the enterprise arrived to pick up the item, the customer may call the call center or store and demand for an immediate refund. In such situations, the call center and store representatives may need to issue a refund for the items immediately.

### 4.15.12.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which enables users to issue refund to the customer immediately.

The `getCompleteOrderDetails` API is called to display the details of all order lines of the return order. When the user selects the lines for refund and clicks Next, the `createOrderInvoice` API is called to create the invoice

for the selected items. The output of the createOrderInvoice API is used to display the header charges, tax breakup, line details and line charges in the Invoice Details screen.

The processOrderPayments API is called to authorize the payment type and refund for the selected items. If the return order line is associated with a sales order, this API processes payments against the corresponding sales order. If the return order line is not associated with a sales order, this API processes payments against the return order.

The getChargeNameList API is called to display the names of the charges and the getChargeCategoryList API is called to display the charge category if the charge name is not displayed.

#### **4.15.12.2 End-User Impact**

None.

#### **4.15.12.3 Implementation**

This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### **4.15.12.4 Reference Implementation**

Sterling Call Center and Sterling Store provides permissions to issue refund for return orders to all users as part of reference implementation.

### **4.15.13 Change Return Address**

A customer may want to change or modify the addresses on a return order. In such situations, call center or store representatives can modify the addresses as requested.

---

---

**Note:** When the order header contains a CustomerContactID, the manageCustomer API saves the addresses to the applicable customer contact.

---

---

#### **4.15.13.1 Solution**

Sterling Call Center and Sterling Store provides a user interface task which enables users to change the address on a return order.

When the user has to enter an order address for a new customer, or modify an existing address, the validity of the address needs to be verified. For more information about address verification, see [Section 4.6.4.2, "Address Verification"](#).

The changeOrder API is called to change the address. If the return method is pick up, the generateWorkOrder API is called to take a new appointment.

### 4.15.13.2 End-User Impact

None.

### 4.15.13.3 Implementation

This section explains the configurations for the change return address task.

- Sterling Call Center and Sterling Store allows you to configure the return order statuses for which changing return address is allowed. For more information about configuring the return order modification statuses, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

### 4.15.13.4 Reference Implementation

Sterling Call Center and Sterling Store provides permissions to change return address to all users as part of reference implementation.

## 4.15.14 Change Service Appointment

A customer may want to change the appointment for a return order and an exchange order, if applicable. An appointment needs to be taken with the corresponding enterprise to agree on a time frame for the services to be fulfilled. Additionally, customers may need to provide the enterprise with specific instructions to ensure that the service fulfillment is successful, for example, the customer may request the retailer to deliver the exchange item at the back door.

#### 4.15.14.1 Solution

Sterling Call Center and Sterling Store provides configurable appointment screens, which include the Service Appointment screen and Service Fulfillment Summary screen that allow users modify service appointments and enter service instructions. For more information about configuring the service appointment calendar, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### Service Fulfillment Summary Screen

The Service Fulfillment Summary screen provides visibility to service appointments pertaining to a return order and exchange order, if applicable. The `getOrderFulfillmentDetails` API is called to display a list of appointments.

The gift options hyperlink is displayed only for the exchange orders. When you select an item as a gift item, the `changeOrder` API is called to modify the gift option.

#### Service Appointment Screen

A calendar is displayed from which the user can select the appropriate appointment for the customer.

You can also configure the number of weeks to display on the calendar at a time. By default, a complete month is displayed. To improve the performance, reduce the number of weeks to display to two so that only two weeks of data needs to be returned by the `getWorkOrderAppointmentOptions` API.

The `getOrderFulfillmentDetails` API is called to display the details of the appointment and the order lines included in the appointment.

Users can look beyond the initial weeks displayed on the calendar in case there are no acceptable appointments within the initial timeframe. By default, the appointment calendar allows the user to look out infinitely into the future for appointments. However, your capacity system may not have visibility beyond a certain timeframe. You can configure the maximum number of days to allow users to look into the future. After the maximum number of days is reached in the appointment screen, the user will not be able to proceed to the next time period.

When an appointment is saved, the `reserveOrder` API is called for exchange order lines if there are any on this appointment. This API calls the `changeOrder` API to make a reservation. If the reservation succeeds,

the modifyWorkOrder API is called to save the appointment on the work order.

The return order lines are displayed in the return order lines panel and exchange order lines are displayed in the exchange order lines panel.

### 4.15.14.2 End-User Impact

None.

### 4.15.14.3 Implementation

This section explains the configurations for the change service appointment task.

- To change the service appointment calendar view, configure the following rules:
  - Calendar Display
  - Default Appointment Calendar View
  - Capacity Information

For more information about configuring the service appointment calendar view, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- Sterling Call Center and Sterling Store allows you to configure the return order statuses for which changing service appointment is allowed. For more information about configuring the return order modification statuses, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

### 4.15.14.4 Reference Implementation

The section explains the reference implementation provided for the change service appointment task.

- An instruction type called 'DELIVERY' is provided for saving service appointment instructions.

- Sterling Call Center and Sterling Store provides permissions to change service appointments and instructions to all users as part of reference implementation.

## 4.15.15 Change Return Method

Sometimes a customer may want to change the return method of the return items or fulfillment method for the exchange items. For example, a customer may have agreed to ship the item back to a return center when creating a return, but may call back later to schedule a return pickup appointment.

### 4.15.15.1 Solution

Sterling Call Center and Sterling Store provides a user interface task which enables users to change the return method of an item.

This section explains the Change Return Method process and the solution offered by Sterling Call Center and Sterling Store.

The getOrderFulfillmentDetails API is called to retrieve the fulfillment details of the return items. The getReturnMethod API is called to retrieve the return method of all the items.

Delivery methods are suggested by Sterling Call Center and Sterling Store and can be customized through the YCDGetReturnMethodUE user exit. The delivery methods are:

- Return Pickup—The items being returned will be picked up at the customer's address.
- Return Shipping—The customer will ship the items back to the return center.
- Items can be kept—The customer can keep the items because the cost of returning items is higher than the value of the item.

When a return line is marked as Items can be kept, the line type is changed to Credit. When an extraneous line is marked as Items can be kept, the line type is changed to ExtraLine\_Credit.

If there are any exchange items, the getOrderFulfillmentDetails API is called again to retrieve the fulfillment details of the exchange items.

After the user's first visit to the Fulfillment Summary screen, if he makes changes to the Return Order and revisits the Fulfillment Summary

screen, it may not contain changes made during the original visit because the Fulfillment Summary screen always chooses options based on the configuration that is set by the enterprise for the items.

For example, originally, a pickup appointment may have been scheduled for three items being returned (two expensive items and one inexpensive item). The customer changes his/her mind and decides to return only the inexpensive item instead. On the second visit to the Fulfillment Summary screen, the customer should be instructed to keep the item instead. To help users remember this, a warning message is displayed on the second and subsequent visits to the Fulfillment Summary screen reminding them to review the information carefully on this screen and ensure that options they set previously are still in effect. They can reset these options, if necessary.

The current delivery method, the delivery methods that are allowed, and the user's privileges decide which return methods are displayed, as determined by the `IsSchedulePickupAllowed`, `IsReturnReturnNodeFound`, and `IsReceiptExpected` attributes. If `IsReceiptExpected` is returned as "N" from the `getReturnMethod` API, then the Customer Can Keep button is displayed, provided the user has privileges.

When the user modifies the return method and clicks Next, the `changeOrder` API is called to update the return method.

The `generateWorkOrder` API checks for order lines marked for delivery and determines if it has a delivery service associated with it, and if not, the `generateWorkOrder` API creates one and then creates a work order. If several work orders are required, `generateWorkOrder` tries to consolidate them.

### Handling Bundle Items

Bundles on exchange lines are handled so that if a parent is selected, all the children are also selected. For more information about bundles, see [Section 4.2, "Configured Items"](#).

#### 4.15.15.2 End User Impact

The Customer Can Keep button is displayed only if the user has permissions to Override Return Method To Allow Customer To Keep Items.



### 4.15.15.3 Implementation

This section explains the configurations for the Change Return Method task.

- The YCDGetReturnMethodUE applies a standard user exit template and passes the result to the user exit. The default implementation is not provided for this user exit.
- If the user selects the Ship To Return Center return method, the RETURN\_FULFILLMENT fulfillment type is used to determine the correct return center to receive the item. Ensure that you configure the sourcing rules to select the appropriate return center node for RETURN\_FULFILLMENT fulfillment type. For more information about configuring sourcing rules, refer to the *Sterling Distributed Order Management: Configuration Guide*. The return center node will be displayed on the Return Fulfillment Summary screen so the CSR can tell the customer the correct address to ship the items back to the return center.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

### 4.15.15.4 Reference Implementation

This section explains the reference implementation provided as part of change return method task.

- Sterling Call Center and Sterling Store provides access to the Customer Can Keep button to all users as part of reference implementation
- Sterling Call Center and Sterling Store provides permissions to change the return method to all users as part of reference implementation.

## 4.15.16 Cancel Return Order

A customer may want to cancel some of the items or the entire return order. The customer may also want to cancel the exchange order corresponding to the return order.

### **4.15.16.1 Solution**

Sterling Call Center and Sterling Store provides a user interface task which enables users to cancel some of the items or all the items included in the return order and exchange order.

The `getCompleteOrderDetails` API is called to display the details of the order. The `changeOrder` API is called to cancel the item(s).

### **Selecting The Cancel Quantity**

The users can cancel the entire order, or cancel a part of the order as requested by the customer.

### **Cancelling Related Lines**

When the user selects a parent item that has related product or service lines for cancellation, the related lines are automatically selected for cancellation.

### **Cancellation Reasons For Return**

Sterling Call Center and Sterling Store provides a list of common codes to display reasons for cancelling a return item. You can define reasons such as 'change of mind' to allow the Cancel action.

### **Cancellation Of Exchange Order Corresponding To The Return Order**

If the return that is being cancelled has an associated exchange order, Sterling Call Center and Sterling Store enables users to cancel some of the items or all the items in the exchange order. For more information about canceling an exchange order, see [Section 4.9.16, "Cancel Order"](#).

### **Viewing and Confirming Payment Details**

The Payment Confirmation screen displays the summary of the payment impact on the order as a result of cancelling items. For more information

about viewing and confirming payment details, see [Section 4.15.17, "Payment Confirmation"](#).

---

---

**Note:** If the user attempts to cancel a line for which an invoice has been created, the Sterling Call Center and Sterling Store application displays an error, and does not allow the user to cancel the line.

---

---

#### 4.15.16.2 End-User Impact

None.

#### 4.15.16.3 Implementation

This section explains the configurations for the cancel return order task.

- Sterling Call Center and Sterling Store allows you to define reason codes that are available during the cancellation process. For more information about defining cancellation reasons, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store allows you to configure the return order statuses for which cancellation of return order lines is allowed. If you allow cancellation of return order lines for a particular status, ensure that you allow cancellation of work order lines for the corresponding status. For more information about configuring the order statuses for modification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign permissions to user groups for this task.

#### 4.15.16.4 Reference Implementation

This section explains the reference implementation provided as part of cancel return order task.

- The following reasons are configured in the "CANCEL\_REASON" common code for returns:
  - Change of Mind
  - Others

- Sterling Call Center and Sterling Store provides permissions to cancel return order to all users as part of reference implementation.

### 4.15.17 Payment Confirmation

Users need to view and change payment details for return orders and exchange orders.

**Note:** When the order header contains a CustomerContactID, the manageCustomer API saves the payment methods to the applicable customer contact.

#### 4.15.17.1 Solution

On the Payment Summary screen, the user can view payment details for returns and exchanges, view header, line charges, and taxes for returns and exchanges, as well as add a note to the return.

The getCompleteOrderDetails API is called to retrieve the details of the return order and exchange order, if applicable. The getChargeCategoryList API is called to retrieve the list of charge categories. The getChargeNameList API is called to retrieve the list of charge names. The getChargeCategoryList API and getChargeNameList API are called again to retrieve list of charge categories and the list of charge names respectively for the exchange order, if an exchange order exists.

If there is a payment impact, the changeOrder API is called to confirm the payment.

#### 4.15.17.2 End User Impact

If the Exchange Type attribute is set to Regular or Advanced, the user will see a consolidated summary for return orders and exchange orders. If the Exchange Type attribute is set to Advanced Prepaid, the user will see separate summary panels for return orders and exchange orders.

#### 4.15.17.3 Implementation

You should assign permissions to users for the following tasks:

- Add Coupon
- Add/ Modify Charges

- Add Payment Method

#### 4.15.17.4 Reference Implementation

None.

## 4.16 Customer Creation

Sterling Call Center and Sterling Store enables you to create two types of customers:

- Consumer Customers
- Business Customers

---

---

**Note:** The customer creation feature launches a single wizard that displays different screens depending on the customer type.

---

---

### 4.16.1 Create Consumer Customer

Call center and store representatives must be able to create customer records containing contact and payment information.

---

---

**Note:** Payment information can only be entered if the Customer Payment Method rule is configured to allow customer payment method information to be stored and used through the application.

---

---

#### 4.16.1.1 Solution

Sterling Call Center and Sterling Store provides a user interface feature that enables CSRs to create and manage consumer customers. This

section explains the Create Consumer task and the solution offered by Sterling Call Center and Sterling Store.

---

---

**Note:** When a CSR logs in to Sterling Call Center and Sterling Store, the organizations to which the CSR has access are automatically retrieved.

If the CSR has access to only one organization or all the organizations point to the same customer master organization, the Create Consumer screen is immediately displayed at customer creation. However, if the CSR has access to multiple enterprises and the enterprises are associated to more than one customer master organization, the CSR must first select an enterprise before creating a customer for the customer master organization of the selected enterprise.

---

---

### Create a Contact

A consumer customer consists of a single contact. The only information required for a consumer customer is a consumer contact record, which serves as a unique identifier. The First Name and Last Name of a customer are mandatory.

The Create Consumer screen contains a Login field in which you can enter a User ID.

---

---

**Note:** Login cannot be changed after it has been entered and saved.

---

---

When a user ID is entered:

- The createUserHierarchy API creates a user record containing the customer's information, such as the customer's user ID, locale, password, e-mail address, and contact information.

---

**Note:** The Selling and Fulfillment Foundation's IS\_GLOBAL\_USER rule determines if the UserID on the user record is globally unique or unique to only the customer enterprise.

If the UserID is globally unique, the DisplayUserID, the LoginID stored on the user record, and the UserID stored on the contact record are the same. If the UserID is specific to the customer enterprise, the DisplayUserID is collected on the screen, and an underscore and the organization suffix are appended to the LoginID stored on the user record and the UserID stored on the contact record to make them globally unique.

---

- The GeneratePassword attribute in the createUserHierarchy API is set to Y to generate a web user password for the customer. The createUserHierarchy API invokes the RESET\_PASSWORD.ON\_SUCCESS, which triggers a service call to YCD\_Send\_Generated\_Password\_8.5. This causes an e-mail, containing the web user's login and password, to be sent to the newly created customer. This functionality is activated through Sterling Call Center and Sterling Store's reference implementation.

---

**Note:** If a user ID is provided, the e-mail address is mandatory in order to send an e-mail containing the web user's password to the customer. However, if a user ID is not provided, the e-mail address is not mandatory, and it will be saved to only the customer's contact record.

---

If a user record already exists, the manageUserHierarchy API is called.

The YCD\_DEFAULT\_CONSUMER\_USER\_ROLE rule is provided as part of the Sterling Call Center and Sterling Store factory setup. If a value has been defined for this rule, the defined user role is automatically assigned

to the new user record, providing the user with the required permissions to access the Web channel, if required.

The `manageCustomer` API is called to save the customer record. If the user creating the customer requires permission to access customers, the `manageCustomerAssignment` API is called to assign the customer to the user and his team.

### Add a Payment Method

If the Customer Payment Method rule is configured for a customer's enterprise to allow the customer payment method information to be stored and used through the application, a user can add payment methods to the customer's account. For more information about configuring customer payment methods, see . The `manageCustomer` API is called to assign the payment methods. These payment methods can be used by the customer when placing orders. The default payment method will be the default option selected on the orders created for the customer.

#### 4.16.1.2 End-User Impact

If the Configure Payment Method rule is configured for a customer's enterprise to allow the customer payment method information to be stored and used through the application, the user will see the Add Payment Method hyperlink. For more information about configuring customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.16.1.3 Implementation

If a user wants to add payment methods to a customer record, the Configure Payment Method rule should be configured for the customer's enterprise to allow the customer payment method information to be stored and used through the application, and all the payment methods must be previously defined in the Channel Applications Manager for the organization of the user creating the customer. For more information about configuring the customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. For more information about adding payment methods, refer to the *Sterling Distributed Order Management: Configuration Guide*.

This task is permission controlled. You can launch this task only if you belong to a user group that is assigned the necessary permissions.



#### 4.16.1.4 Reference Implementation

Consumer customers are provided as part of reference implementation for all the organizations. For more information about the customers for each organization, see the *Sterling Selling and Fulfillment Suite: Applications Reference Implementation Guide*.

### 4.16.2 Create Business Customer

Call center and store representatives must be able to create business customer records containing contact and payment information.

---

**Note:** Payment information can only be entered if the Customer Payment Method rule is configured to allow customer payment method information to be stored and used through the application.

---

#### 4.16.2.1 Solution

Sterling Call Center and Sterling Store provides a user interface feature that enables users to create and manage business customers. This section explains the Create Business Customer task and the solution offered by Sterling Call Center and Sterling Store.

---

**Note:** When a CSR logs in to Sterling Call Center and Sterling Store, the organizations to which the CSR has access are automatically retrieved.

---

If the CSR has access to only one organization or all the organizations point to the same customer master organization, the Customer Entry screen is immediately displayed at customer creation. However, if the CSR has access to multiple enterprises and the enterprises are associated to more than one customer master organization, the CSR must first select an enterprise before creating a business customer for the customer master organization of the selected enterprise.

---

#### Create a Customer

A business customer consists of a Buyer Organization and any number of contacts. The addresses and payment methods can be defined for the

customer or for specific contacts. The unique identifier of a business customer is the Customer ID.

---

---

**Note:** The Enterprise Organization suffix defined in the Channel Applications Manager will automatically be appended to the Customer ID, creating the Customer Organization record.

---

---

The minimum information required to create a Business Organization is the business customer's ID and name, and a single contact record. The Create Business Customer task enables you to enter this information and other customer and organization-level attributes, as well as the default addresses and contact information pertaining to the customer.

---

---

**Note:** You must define at least one customer contact for a business customer. The First Name and Last Name of the customer contact are mandatory.

---

---

The Contact Entry screen contains a Login field in which you can enter a user ID.

---

---

**Note:** Login cannot be changed after it has been entered and saved.

---

---

When a user ID is entered:

- The createUserHierarchy API creates a user record containing the customer's information, such as the customer's user ID, locale, password, e-mail address, and contact information.

---

**Note:** The Selling and Fulfillment Foundation's IS\_GLOBAL\_USER rule determines if the UserID on the user record is globally unique or unique to only the customer enterprise.

If the UserID is globally unique, the DisplayUserID, the LoginID stored on the user record, and the UserID stored on the contact record are the same. If the UserID is specific to the customer enterprise, the DisplayUserID is collected on the screen, and an underscore and the organization suffix are appended to the LoginID stored on the user record and the UserID stored on the contact record to make them globally unique.

---

- The GeneratePassword attribute in the createUserHierarchy API is set to Y to generate a web user password for the customer. The createUserHierarchy API invokes the RESET\_PASSWORD.ON\_SUCCESS event, which triggers a service call to YCD\_Send\_Generated\_Password\_8.5. This causes an e-mail, containing the web user's login and password, to be sent to the newly created customer. This functionality is activated through Sterling Call Center and Sterling Store's reference implementation.

---

**Note:** If a user ID is provided, the e-mail address is mandatory in order to send an e-mail containing the user's password to the customer. However, if a user ID is not provided, the e-mail address is not mandatory, and it will be saved to only the customer's contact record.

---

If a user record already exists, the manageUserHierarchy API is called.

You can assign user roles to the new user record by selecting the appropriate check boxes in the Customer Account panel. The getUserGroupList API populates the list of available user roles based on

the organization and the "EXTERNAL" group type. The "EXTERNAL" group type limits the list of user groups to those groups exposed for web use.

The `getCurrencyList` API is called to populate the Default Currency drop-down list. As part of the customer creation process, you can define only one currency for the customer. You can define additional currencies and set a default currency at a later time using the Manage Customer Information task.

The `getCommonCodeList` API is called to populate the drop-down options for Status, Relationship Type, Customer Level, and Vertical. This data is cached, as described in [Section 7.3, "Cached Data"](#).

All addresses created in the customer entry screen are created at the customer level, and are accessible by all contacts of the customer record. All addresses created in the contact screen are created specifically for that contact. The `manageCustomer` API is called to save the customer record. If the user requires permission to access customers, the `manageCustomerAssignment` API is called to assign the customer to the user and the user's team.

### Add a Payment Method

If the Customer Payment Method rule is configured for a customer's enterprise to allow the customer payment method information to be stored and used through the application, a user can add payment methods to the customer's account. For more information about configuring the customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. The `manageCustomer` API is called to assign the payment methods. These payment methods can be used by the customer when placing orders. The default payment method will be the default option selected in the orders created for the customer.

#### 4.16.2.2 End-User Impact

If the Customer Payment Method rule is configured for a customer's enterprise to allow the customer payment method information to be stored and used through the application, the user will see the Add Payment Method hyperlink. For more information about configuring the customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.16.2.3 Implementation

If the user wants to add payment methods to a customer record, the Customer Payment Method rule should be configured for the customer's enterprise to allow the customer payment method information to be stored and used through the application, and all the payment methods must be previously defined in the Channel Applications Manager for the organization of the user creating the customer. For more information about configuring the customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. For more information about adding payment methods, refer to the *Sterling Distributed Order Management: Configuration Guide*.

This task is permission controlled. You can launch this task only if you belong to a user group that is assigned the necessary permissions.

#### 4.16.2.4 Reference Implementation

Business customers are provided as part of reference implementation for all the organizations. For more information about the customers for each organization, see the *Sterling Selling and Fulfillment Suite: Applications Reference Implementation Guide*.

## 4.17 Customer Details

Users need the ability to view customer details for both business and consumer customers.

### 4.17.1 Solution

Sterling Call Center and Sterling Store provides the ability to view customer details for both business and consumer customers.

The `getCustomerDetails` API is called to display customer details screens. The `getCustomerDetails` API launches a single wizard that displays different screens depending on the customer type. For a consumer customer, the Consumer Details screen is displayed. For a business customer, the Business Customer Details screen is displayed.

#### Consumer Details Screen

The Consumer Details screen displays the name, status, contact information, Bill To address, and Ship To address of the consumer customer. It also contains five tabs: Draft Orders, Order History, Return

History, Addresses, and Payment Methods. In addition, an Important Notes panel is displayed at the bottom of the Consumer Details screen.

The `getCustomerDetails` API is called to populate the Consumer Details screen with the name, status, contact information, Bill To address, and Ship To address of the consumer customer, as well as information in the Addresses and Payment Methods tabs. In addition, it populates the Important Notes panel with notes pertaining to the consumer customer. When you click the View/Add Notes hyperlink in the Important Notes panel, the `getCommonCodeList` API is called to retrieve the note types and contact methods for the customer.

The `getOrderList` API is called to populate the Draft Orders, Order History, and Return History tabs with the customer's orders.

---

**Note:** The Draft Orders, Order History, and Return History tabs support pagination.

---

The Order History tab provides search capabilities, enabling the user to search for orders by hold type, status, or both hold type and status. Searching by hold type displays orders that have a specific hold applied to the orders. When searching by status:

- **Open** displays all orders.
- **Backordered** displays orders that have a status of backordered or extended (1500).
- **Shipped** displays orders that have a status of shipped or greater than shipped (3700).

---

**Note:** Extended statuses are also considered when searching by status.

---

The Order History tab also provides the ability to toggle between recent orders and orders in the history table.

### Business Customer Details Screen

The Business Customer Details screen displays the name, ID, status, Bill To address, Ship To address, and Sold To address of the business customer. The customer's name is a link to the customer's Web site, if available, which will open in the default browser. The Business Customer

Details screen also contains seven tabs: Contacts, Draft Orders, Order History, Return History, Addresses, Payment Methods, and Child Customers. An Important Notes panel is displayed at the bottom of the Business Customer Details screen.

The `getCustomerDetails` API is called to populate the Business Customer Details screen with the name, ID, status, Bill To address, Ship To address, and Sold To address of the business customer, as well as information in the Addresses and Payment Methods tabs. In addition, the `getCustomerDetails` API populates the list of contacts in the Contacts tab. In the Contacts tab, information about the selected contact is displayed in the panel on the right of the screen. When you double-click a contact, the `getCustomerDetails` API, `getUserGroupList` API, `getUserHierarchy` API, and `getLocaleList` API are called to populate the Contact Details screen for the selected contact. Finally, the `getCustomerDetails` API populates the Important Notes panel with notes pertaining to the business customer. When you click the View/Add Notes hyperlink in the Important Notes panel, the `getCommonCodeList` API is called to retrieve the note types and contact methods for the customer.

---

---

**Note:** If a customer or contact does not have default addresses or payment methods defined, the default addresses or payment methods of the closest customer in the customer hierarchy are used.

---

---

The `getOrderList` API is called to populate the Draft Orders, Order History, and Return History tabs with the customer's orders.

---

---

**Note:** The Draft Orders, Order History, and Return History tabs support pagination.

---

---

The Order History tab provides search capabilities, enabling the user to search for orders by hold type, status, or both hold type and status. Searching by hold type displays orders that have a specific hold type applied to the orders. When searching by status:

- **Open** displays all orders.
- **Backordered** displays orders that have a status of backordered or extended (1500).

- **Shipped** displays orders that have a status of shipped or greater than shipped (3700).

**Note:** Extended statuses are also considered when searching by status.

The Order History tab also provides the ability to toggle between recent orders and orders in the history table.

When the Child Customers tab is selected, the `getCustomerList` API is called to retrieve a list of the child customers, if any, of the current customer. In addition, if a parent customer exists for the current customer, a link to the parent customer's Business Customer Details screen is displayed. When you double-click a child customer or the Parent Customer link, the `getCustomerDetails` API is called to display the Business Customer Details screen for the applicable child customer or parent customer.

### 4.17.2 End-User Impact

The end user can search for consumer customers or business customers against the customer database already in the database.

### 4.17.3 Implementation

None.

### 4.17.4 Reference Implementation

None.

## 4.18 Customer Inquiry

Users need the ability to perform customer inquiry for both business and consumer customers.

### 4.18.1 Solution

Sterling Call Center and Sterling Store provides the ability to search and view two types of customers:



- Consumer Customers
- Business Customers

Sterling Call Center and Sterling Store provides two ways to search for customers:

- Quick Access—The user can select the customer type and enter the customer's e-mail ID, user ID, day phone number, and the business name in case of a business customer and search for the customer.
- Advanced Customer Search—The user can enter more restrictive search criteria to search for the customer.

If a user has access to multiple enterprises, the Enterprise drop-down list is displayed in the Customer Search Criteria panel of the Advanced Customer Search screen. The `getOrganizationList` API is called to retrieve the list of enterprises. If the search results in a single record, the Customer Details screen is displayed. If the search criteria results in multiple records being retrieved, the Search and List screen is displayed.

The `getCustomerList` API is called to retrieve the customer records. The `getCustomerList` API supports case-insensitive searches for system defined columns. For more information about enabling the `getCustomerList` API to support case-insensitive searches for search tasks in the Sterling Call Center and Sterling Store applications, refer to the *Selling and Fulfillment Foundation: Extending the Database Guide*. For more information about the `getCustomerList` API, refer to the *Sterling Call Center and Sterling Store: Javadocs*.

## Pagination

Sterling Call Center and Sterling Store supports smart retrieval of record sets in the Customer Search screen.

For more information about pagination, see [Section 4.22, "Pagination"](#).

---



---

**Note:** If searching by address for business customers is enabled, special performance considerations need to be evaluated to ensure that this type of query is fast enough.

---



---

## 4.18.2 End-User Impact

The end user can search for Consumer Customers or Business Customers against the customer database or addresses already in the database.

### 4.18.3 Implementation

Sterling Call Center and Sterling Store allows you to configure search for business customers and consumer customers. For more information about configuring customer search options, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 4.18.4 Reference Implementation

Sterling Call Center and Sterling Store provides customers as part of reference implementation for all the organizations. For more information about these customers, see the *Sterling Selling and Fulfillment Suite: Applications Reference Implementation Guide*.

## 4.19 Customer Maintenance

Sterling Call Center and Sterling Store provides the ability to manage a customer's business information, contacts, addresses, and payment methods.

### 4.19.1 Manage Contacts

CSRs must be able to manage customer records from Sterling Call Center and Sterling Store.

#### 4.19.1.1 Solution

Sterling Call Center and Sterling Store enables you to manage the contacts of a consumer or business customer. You can add contacts, modify contact records, modify contact addresses, manage contact payment methods, and delete contacts that are no longer active.

A consumer customer can have only one contact record associated to that customer. For a business customer, multiple contact records can be available for modification, creation, or deletion.

The `getCustomerDetails` API is called to display the list of customer contact records, and the `manageCustomer` API is called to modify the contact details.

## Consumer Customer

The Manage Contact Information task, which is accessible from the Consumer Details screen, enables a CSR to modify the contact information pertaining to a consumer customer. When the CSR selects this task, the `getCustomerDetails` API is called.

---

**Note:** First Name, Last Name, and Status are mandatory fields in the Manage Contact Information screen. Also, if a login ID is entered in the Login field, E-mail Address and Locale are mandatory fields.

---

Updates to the user record are exposed and saved with the `createUserHierarchy` API or `modifyUserHierarchy` API. If the `createUserHierarchy` API is called, the `GeneratePassword` attribute is set to Y to generate a web user password for the customer. The `createUserHierarchy` API invokes the `RESET_PASSWORD.ON_SUCCESS`, which triggers a service call to `YCD_Send_Generated_Password_8.5`. This causes an e-mail, containing the web user's login and password, to be sent to the newly created customer. This functionality is activated through Sterling Call Center and Sterling Store's reference implementation.

For information about the `IS_GLOBAL_USER` rule, which determines if the user ID on the user record is globally unique or unique to only the customer enterprise, see [Section 4.16.1, "Create Consumer Customer"](#).

The `manageCustomer` API is called to save the modified contact details.

## Business Customer

Multiple contacts can be defined for a business customer. The tasks that are available to manage contacts can be accessed from the Business Customer Details screen, and all the business customer-related screens.

When the View Contact Details hyperlink in the Business Customer Details screen is clicked, the details pertaining to the selected contact are displayed.

---

**Note:** After a contact is created or modified, if the customer associated to that contact has an editor open, the editor will be closed and the Customer Details screen will be reopened so that the update is immediately visible.

---

- **Add Contacts**

The Create Customer Contact task enables a CSR to add a new contact. In the Create Customer Contact screen, the CSR can enter contact information such as first name, last name, e-mail address, login ID, locale, status, phone numbers, user roles, contact spending limit, and approvers.

---

---

**Note:** First Name, Last Name, and Status are mandatory fields in the Create Customer Contact screen. Also, if a login ID is entered in the Login field, E-mail Address and Locale are mandatory fields.

---

---

Updates to the user record are exposed and saved with the createUserHierarchy API or modifyUserHierarchy API. If the createUserHierarchy API is called, the GeneratePassword attribute is set to Y to generate a web user password for the customer. The createUserHierarchy API invokes the RESET\_PASSWORD.ON\_SUCCESS, which triggers a service call to YCD\_Send\_Generated\_Password\_8.5. This causes an e-mail, containing the web user's login and password, to be sent to the newly created customer. This functionality is activated through Sterling Call Center and Sterling Store's reference implementation.

For information about the IS\_GLOBAL\_USER rule, which determines if the user ID on the user record is globally unique or unique to only the customer enterprise, see [Section 4.16.2, "Create Business Customer"](#).

The manageCustomer API is called to save the contact details.

- **Modify Contacts**

The Manage Customer Contact task enables a CSR to modify the details of a customer contact.

From the Contact Details screen, which can be accessed by selecting a contact and clicking the View Contact Details hyperlink in the Business Customer Details screen, a CSR can select the Manage Customer Contact task. The fields in the Manage Customer Contact screen are identical to the fields in the Create Customer Contact

screen, except the fields in the Manage Customer Contact screen are populated with data.

**Note:** First Name, Last Name, and Status are mandatory fields in the Manage Customer Contact screen. If a login ID is entered in the Login field, E-mail Address and Locale are mandatory fields. If a login ID was previously entered, a user record already exists and the Login field is disabled.

For information about the IS\_GLOBAL\_USER rule, which determines if the user ID on the user record is globally unique or unique to only the customer enterprise, see [Section 4.16.2, "Create Business Customer"](#).

The getCustomerDetails API is called to retrieve the customer contact record. Based on the contact's CustomerContactID, the CustomerContact element is extracted from the input. The getUserGroupList API is called to populate the User Roles panel with the available user groups that have a user type of external. The getUserHierarchy API is called to retrieve the customer's locale values, and the getLocaleList API is called to populate the Locale drop-down list.

Updates to the user record are exposed and saved with the createUserHierarchy API or modifyUserHierarchy API. If the createUserHierarchy API is called, the GeneratePassword attribute is set to Y to generate a web user password for the customer. The createUserHierarchy API invokes the RESET\_PASSWORD.ON\_SUCCESS, which triggers a service call to YCD\_Send\_Generated\_Password\_8.5. This causes an e-mail, containing the web user's login and password, to be sent to the newly created customer. This functionality is activated through Sterling Call Center and Sterling Store's reference implementation.

The manageCustomer API is called to save the modified contact details.

- **Delete Contacts**

The Delete Customer Contact task, which can be accessed from the Contact Details screen, enables a CSR to delete a contact. A pop-up window is displayed, which prompts the CSR to confirm the deletion. When the CSR confirms the deletion, the contact is removed from the

Business Customer Details screen. The manageCustomer API is called to save the changes.

For information about managing customer addresses, see [Section 4.19.3, "Manage Customer Addresses"](#). For information about managing customer payment methods, see [Section 4.19.4, "Manage Payment Methods"](#).

### 4.19.1.2 End-User Impact

These tasks are permission controlled. You can launch these tasks only if you belong to a user group that is assigned the necessary permissions.

### 4.19.1.3 Implementation

These tasks are permission controlled. If you do not have access to the customer details record, these tasks are not available, regardless of resource permissions.

### 4.19.1.4 Reference Implementation

None.

## 4.19.2 Manage Business Customer Information

Users must be able to manage business customer information from Sterling Call Center and Sterling Store.

### 4.19.2.1 Solution

Sterling Call Center and Sterling Store enables you to manage business customer information, such as a customer's business name, Web site, status, shipping preferences, and tax information. In addition, you can manage business customer classifications, such as the customer's relationship type and customer level. You can access the Manage Customer Information task from the Business Customer Details screen.

---

---

**Note:** In the Manage Customer Information screen, the Customer ID field is disabled. In addition, Business Name and Default Currency are mandatory fields.

---

---

The `getCustomerDetails` API is called to populate the fields in the Manage Customer Information screen. The `getCommonCodeList` API is called to populate the drop-down options for Status, Relationship Type, Customer Level, and Vertical. The `getCurrencyList` API is called to retrieve the list of currencies. The `manageCustomer` API is called to save the changes to the customer record.

#### **4.19.2.2 End-User Impact**

This task is permission controlled. You can launch this task only if you belong to a particular user group that is assigned permissions.

#### **4.19.2.3 Implementation**

This task is permission controlled. If the user does not have access to the customer details record, this task is not available, regardless of the user's resource permission.

#### **4.19.2.4 Reference Implementation**

None.

### **4.19.3 Manage Customer Addresses**

Users may need to define multiple addresses for a customer.

#### **4.19.3.1 Solution**

Sterling Call Center and Sterling Store enables users to define addresses for a customer. Users can define addresses either at the contact level or at the customer level, and provide a description for the addresses. A user can also define default Ship To, Bill-To and Sold To addresses to be used during ordering. If a contact has no defaults selected, then the customer default addresses will be used.

You can define and manage addresses for business contacts from any of the business customer contact screens.

The `getCustomerDetails` API is called to display the list of customer addresses. The `manageCustomer` API is called to modify the customer addresses.

#### **4.19.3.2 End-User Impact**

This task is permission controlled. Users can launch this task only if the user belongs to a particular user group that is assigned permissions.

#### **4.19.3.3 Implementation**

This task is permission controlled. If the user does not have access to the customer details record, this task is not available, regardless of the user's resource permission.

#### **4.19.3.4 Reference Implementation**

None.

### **4.19.4 Manage Payment Methods**

There may be situations when users have to define payment methods for customers so that the customer can choose these payment methods while placing an order without having to enter the payment details again. Payment methods can be defined only if the Customer Payment Method rule is configured for the customer's enterprise to allow the customer payment method information to be stored and used through the application. If the rule is not configured, the Manage Customer Payment Methods hyperlink will not be displayed. For more information about configuring customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### **4.19.4.1 Solution**

The Sterling Call Center and Sterling Store enables you to define payment methods for a customer. You can define payment methods at the contact level or at the customer level.

You can define and manage payment methods for business contacts from any of the business customer contact screens.



You can modify the attributes of a payment method.

---

---

**Note:** Attributes that are unique identifiers are displayed but cannot be modified. Encrypted attributes are not displayed.

For the Customer Account payment method, Customer Account # is a unique identifier. For the Credit Card payment method, Credit Card # is a unique identifier. For the other payment methods, the payment reference numbers are unique identifiers.

---

---

The `getCustomerDetails` API is called to display the list of payment types available for the customer. The `manageCustomer` API is called to create or modify the payment method.

#### 4.19.4.2 End-User Impact

This task is permission controlled. You can launch this task only if you belong to a particular user group that is assigned permissions.

---

---

**Note:** Billing address will not be directly tied to the credit card records stored against the customer. In order to relate them, this needs to be defined during ordering.

Available payment types created against customer records are not limited.

---

---

If the Customer Payment Method rule is configured for a customer's enterprise to allow the customer payment method information to be stored and used through the application, the user will see the Manage Customer Payment Methods hyperlink. For more information about configuring customer payment methods, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

#### 4.19.4.3 Implementation

If the user wants to add payment methods to a customer record, the Customer Payment Method rule should be configured for the customer's enterprise to allow the customer payment method information to be stored and used through the application, and all payment methods must

be previously defined in the Channel Applications Manager for the Organization of the user creating the customer. For more information about adding payment methods, see *Sterling Distributed Order Management: Configuration Guide*.

If the additional attributes of a credit card are encrypted, such as the name on the card or the expiration date, they will not be visible from the Payment Method display panel.

This task is permission controlled. If the user does not have access to the customer details record, this task is not available, regardless of the user's resource permission.

#### **4.19.4.4 Reference Implementation**

None.

### **4.19.5 Reset User Passwords**

Sometimes, customers may forget their user passwords and, consequently, may be unable to access their Web user accounts. CSRs must be able to reset a user's password from Sterling Call Center and Sterling Store.

#### **4.19.5.1 Solution**

Sterling Call Center and Sterling Store enables a CSR to reset a user's password, if a user ID is associated to the corresponding user's record. This section explains the Reset User Password feature, and the solution offered by Sterling Call Center and Sterling Store.

For a consumer customer, the Reset User Password feature can be accessed from the Consumer Details screen. For a business customer contact, the Reset User Password feature can be accessed from the Contact Details screen. When a CSR selects the Reset User Password task and confirms that the user wants to have the password reset, the requestPasswordReset API is called, which performs the following tasks:

- Generates a RequestId, which can later be used by the changePassword API to change the user's password to a temporary password to allow the user to log in to the Web channel and manually change their password.

- Invokes the RESET\_PASSWORD.ON\_REQUEST event, which provides a location from which an e-mail can be sent. This event outputs a UserID, ResetType, and RequestId. With this information, the YCD\_Send\_Reset\_Password\_8.5 sample service generates an e-mail containing a link to a URL with the RequestId that can be used to modify the user password. The URL provided in this sample e-mail must be changed to link to a valid URL for a password reset Web page that must exist in your Web application. This Web page must be accessible by the customer so that the customer can finish the password reset process.

"Email" is the default reset type value for the requestPasswordReset API call.

#### **4.19.5.2 End-User Impact**

This task is permission controlled. You can launch this task only if you belong to a user group that is assigned the necessary permissions.

The user receives an e-mail containing the new password. In addition, the e-mail states that the user should change the password the next time the user logs in to the Web site.

#### **4.19.5.3 Implementation**

This task is permission controlled. If you do not have access to the corresponding customer details record, this task is not available, regardless of the resource permissions provided.

You should modify the sample e-mail template to match your needs. This includes changing the URL that the customers will use to finish the reset password process. The URL must link to a valid Web page that is accessible by the customer.

#### **4.19.5.4 Reference Implementation**

As part of reference implementation, Sterling Call Center and Sterling Store provides a simple password policy, which is configured for the enterprise, that allows passwords to be reset when the reset type is "Email".

The RESET\_PASSWORD.ON\_REQUEST event is activated through the reference implementation provided by the Sterling Call Center and Sterling Store application.

**Note:** In the reference implementation provided by Sterling Call Center and Sterling Store, customer passwords are never exposed to users of the Sterling Call Center and Sterling Store. Sterling Call Center and Sterling Store assumes that a Web site exists in which the customer can reset the password after the request has been made.

### 4.19.6 Customer Team Assignments

Sterling Call Center and Sterling Store provides the ability to assign customers to a team, once the customer is created. A supervisor can assign a customer from his team to any of his immediate sub teams.

#### 4.19.6.1 Solution

The getTeamList API is called to retrieve a list of sub teams assigned to the supervisor. The getCustomerAssignmentList API is called to retrieve a list of the teams already assigned to the customer that are part of the list of sub teams with which the supervisor is working. The manageCustomerAssignment API is called to assign the customer to the selected team.

#### 4.19.6.2 End-User Impact

None.

#### 4.19.6.3 Implementation

The teams (data security groups) must be defined in the Channel Applications Manager and users must be associated to the teams.

The only purpose of this task is to further distribute a customer to sub teams. Therefore, the supervisor's team should also have sub teams defined in the Channel Applications Manager. For more information about creating teams and data security groups, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

#### 4.19.6.4 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides the YCD\_Update\_Customer\_Assignment\_8.0 service. This service calls the updateCustomerAssignmentsLookup API when the On\_Assignments\_Change event is raised for a Customer\_Definition transaction.

---

**Note:** This task will not be accessible if the user is not a supervisor and the user's team does not require Customer Assignments, even if the user has permission to access this related task.

---

### 4.19.7 Customer User Assignments

Supervisors need the ability to assign a customer to selected users on his team.

#### 4.19.7.1 Solution

Sterling Call Center and Sterling Store provides the ability to assign customers to a user, once the user is created and assigned to a team. A supervisor can search for users from his team. The getUserList API is called to retrieve a list of users assigned to the supervisor's team, to limit the results. After a user is selected, the Customer Assignment screen is displayed. This screen provides the supervisor with a view into the customers that a selected user already has access to, and provides the ability to perform a search for additional customers that the supervisor can assign to the user.

#### 4.19.7.2 End-User Impact

If the user is not a supervisor or the user is a member of a team that does not require customer assignments, or the rule to manually assign customers to users is disabled, the related task will not be displayed, even if the user has access permissions.

### 4.19.7.3 Implementation

The teams (data security groups) need to be defined in the Channel Applications Manager, users must be associated to the teams, and the user must be a supervisor.

The purpose of this task is to assign customers to users. Therefore, to access this task the supervisor must be on a team that requires customer assignments and the rule to manually assign customers to users must be enabled.

To define customer access mode, call the `manageTeam` API to set the `customerAccessMode` to 02, which is the Requires Customer Assignment permission.

For more information about creating teams/data security groups, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

### 4.19.7.4 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides the `YCD_Update_Customer_Assignment_8.0` service. This service calls the `updateCustomerAssignmentsLookup` API when the `On_Assignments_Change` event is raised for a `Customer_Definition` transaction.

## 4.19.8 Customer Self Assignments

Users sometimes may need to assign themselves to customers.

### 4.19.8.1 Solution

The Sterling Call Center and Sterling Store provides the ability for users to assign customers to themselves. When a user searches for customers, the `getUserHierarchy` API is called to retrieve the maximum number of customers assignments for the user. The `getCustomerAssignmentList` API is called to retrieve a list of customers. A user can search among customers that are already assigned to his team and assign one to himself. If a customer is selected, the `manageCustomerAssignments` API is called to assign the customer to the user.

#### 4.19.8.2 End-User Impact

If the user is on a team that does not require customer assignment and manual assignment of customers to users is not required, then even if the user has permission to view this task, the related task will not be displayed.

#### 4.19.8.3 Implementation

The teams (data security groups) need to be defined in the Channel Applications Manager, and users must be associated to the teams.

The purpose of this task is to assign customers to users. Therefore, to access this task the user must be on a team that requires customer assignments and the rule to manually assign customers to users must be enabled.

To define customer access mode, call the manageTeam API to set the customerAccessMode to 02, which is the Requires Customer Assignment permission.

For more information about creating teams/data security groups, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

#### 4.19.8.4 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides the YCD\_Update\_Customer\_Assignment\_8.0 service. This service calls the updateCustomerAssignmentsLookup API when the On\_Assignments\_Change event is raised for a Customer\_Definition transaction.

### 4.19.9 Customer Notes

A user may have to enter notes about a customer, such as when the customer can be contacted if queries are raised about an order.

#### 4.19.9.1 Solution

Customer notes can be either system-generated or entered by a user. Users can add notes from the details screen of the customer or contact. Additionally, a user can track important customer-related information by viewing previously logged customer notes.

Sterling Call Center and Sterling Store provides the following features:

- A Customer Notes screen, where a user can view and add notes about a customer.
- An Important Notes panel, where a user can add notes. The Important Notes panel is displayed at the bottom of the Customer Details screen.

### Customer Notes Screen

Sterling Call Center and Sterling Store provides a Customer Notes screen, which can be launched from the customer details screen.

Users can limit the notes that are displayed in the Customer Notes screen based on the following criteria:

- Note Type
- Contact Method
- Priority
- System-Generated or Manually Entered
- Internal Notes

The `getCommonCodeList` API is called to retrieve the note types and contact methods for the customer. The `getCustomerDetails` API is called to retrieve the customer notes, which are filtered on the client. The `manageCustomer` API is called to save the notes.

### Adding Notes to Customer Details

Sterling Call Center and Sterling Store provides a Customer Notes screen to enable users to add notes.

The `getCommonCodeList` API is called to retrieve the note types and contact methods for the customer. The Sterling Call Center and Sterling Store application also enables users to mark the note as an internal note.

### Customer Contact Notes

Users can enter notes about a customer contact. This feature can be accessed from the Contact Details screen. The Customer Notes screen for contacts is identical to the Customer Notes screen for customers.



#### 4.19.9.2 End-User Impact

The notes that have been configured to be logged automatically will be visible to the end users in the Customer Notes screen.

#### 4.19.9.3 Implementation

None.

#### 4.19.9.4 Reference Implementation

The Customer Notes screen uses the same Note Types and Contact Types used by order notes. For information about attributes for customer notes, see the information provided under Reference Implementation in [Section 4.9.18, "Order Notes"](#).

## 4.20 User Creation

Sterling Call Center and Sterling Store representatives may need to create and modify users. For example, a CSR Lead may have to create a new CSR user.

---

**Note:** A user is an employee of the enterprise organization who can log in to the Sterling applications.

---

### 4.20.1 Create a User

Call Center and Store representatives may have to create a user.

#### 4.20.1.1 Solution

Sterling Call Center and Sterling Store provides a user interface task that enables CSRs to create users. This section explains the wizard-based Create User task and the solution offered by Sterling Call Center and Sterling Store.

### Privileges Required

CSRs who can access the Create User task have the necessary privileges to create users for the organizations to which they have access.

**Note:** When assigning user permissions to new users, the CSR can assign permissions to only those groups to which the CSR is assigned. This means that a CSR cannot grant permission to any resource that the CSR does not have access to.

### Create a User Record

To create a user, the CSR must populate the mandatory Organization, Login, Name, E-mail Address, Menu Group, Theme, and Locale fields in the User Entry screen. Initially, the `getOrganizationList`, `getMenuList`, `getThemeList`, and `getLocaleList` APIs are called. When the user's Organization is selected, the `getTeamList` API is called and the Department drop-down list is populated from the selected organization's defined departments. The `getTeamList` API populates the Team drop-down list with all the teams configured for the user's organization and the user's customer master organization.

Additional information that can be entered in this screen includes whether the user is classified as an active user, whether the user is classified as a supervisor, the maximum number of customers who can be assigned to the user, the user's contact address, the user's billing address, and the user's assignment to user roles which defines the user's access to Sterling applications.

After the information is entered and the CSR clicks the Next button, the `createUserHierarchy` API is executed and the corresponding user is created. This user is added to the wizard model so that if the CSR returns to this screen later, the `modifyUserHierarchy` API is called instead of the `createUserHierarchy` API.

When a user record is created, the `GeneratePassword` attribute in the `createUserHierarchy` API is set to Y to generate a web user password for the customer. The `createUserHierarchy` API invokes the `RESET_PASSWORD.ON_SUCCESS` event, which triggers a service call to `YCD_Send_Generated_Password_8.5`. This causes an e-mail, containing the web user's login and password, to be sent to the newly created

customer. This functionality is activated through Sterling Call Center and Sterling Store's reference implementation.

### **Assign Queues**

The second screen in the wizard is the Queue Selection screen, which allows the CSR to define the queues to which the new user will be assigned in order to receive alerts. The left panel lists the queues that are available to the user, and the right panel lists the queues that are assigned to the user.

A `getQueueList` API is called to retrieve the available queues from the user's organization and the default organization, and the results are placed in the Available Queues panel. For a newly created user, the Assigned Queues panel will be empty. For an existing user, the queue subscriptions will be pulled from the `getUserHierarchy` API, and will be manually removed from the output of the `getQueueList` API.

When the CSR clicks the Confirm button, the `modifyUserHierarchy` API is called to save the queue subscription results to the user's record. The `modifyUserHierarchy` must be called with the Reset flag to ensure that the queues that have been removed have actually been removed. The `modifyUserHierarchy` API then takes the source model for the Assigned Queues table and saves those queues against the user.

#### **4.20.1.2 End-User Impact**

None.

#### **4.20.1.3 Implementation**

This task is permission controlled. You can launch this task only if you belong to a user group that is assigned the necessary permissions.

#### **4.20.1.4 Reference Implementation**

The existing users provided by the reference implementation (that is, CSR, CSR Lead, Store Manager, Store Associate) will be visible for the organizations they are created for through the reference implementation.

## **4.20.2 Modify a User**

Sterling Call Center and Sterling Store representatives may have to modify a user's information. For example, a CSR may need to be

assigned to an additional queue to start working on different types of alerts.

### 4.20.2.1 Solution

Sterling Call Center and Sterling Store provides a user interface task that enables CSRs to modify other users. The task is similar to the Create User task, except that the CSR must first search for the user whose information needs to be modified.

This section explains the wizard-based Modify User task and the solution offered by Sterling Call Center and Sterling Store.

### Privileges Required

CSR users who can access the Create User task have the required privileges to modify users for the organizations to which they have access.

### Search for the User

The first screen in this wizard-based task is a search screen, which enables the CSR to search for the user whose details have to be modified. The search criteria are Organization, User ID, Name, and Status. The results are displayed in the User Search Results panel.

### Modify the User Information

After the user is selected, the following screens are almost identical to the screens described in [Section 4.20.1, "Create a User"](#).

The Modify User task opens a similar screen to the Create User task, but the `getUserHierarchy` API is called, which populates the screen with information pertaining to the selected user. Additionally, the Queue Subscription screen is populated with the user's previously assigned queues.

### 4.20.2.2 End-User Impact

None.

### 4.20.2.3 Implementation

See [Section 4.20.1.3, "Implementation"](#).

#### 4.20.2.4 Reference Implementation

None.

## 4.21 User Team Creation

Sterling Call Center and Sterling Store enables you to create and manage user teams. Teams are created to assign CSR users to a set of customers. This assignment enables customers to maintain a relationship with specific CSRs so that customers receive better customer service.

---

---

**Note:** A user team is a collection of users who have common data and user interface access requirements. User teams can have access to specific document types, enterprises, ship nodes, customers, and screens within the user interfaces.

---

---

### 4.21.1 Create a User Team

Sterling Call Center and Sterling Store representatives may have to create a user team, and assign customers to that team.

#### 4.21.1.1 Solution

Sterling Call Center and Sterling Store enables users to create a user team and assign customers to that team. This section explains the Create User Team task and the solution offered by Sterling Call Center and Sterling Store.

#### Create a User Team

To create a user team, the user must populate the mandatory fields in the Team Information panel of the Create User Team task. These include the Team ID, Description, Parent Team, and Organization fields, and the selection of the Customer Access Mode.

---

---

**Note:** The Organization field is displayed only if the user has access to more than one customer master organization. When the Organization field is displayed, the `getOrganizationList` API is called to populate the list of organizations in the Organization drop-down list.

---

---

The user team inherits most of its permissions from a pre-existing parent team. The `getTeamList` API is called to populate the list of parent teams in the Parent Team drop-down list, and the user selects a parent team from this list. The parent team determines which enterprises and document types that the user will have access to when assigned to this team. Typically, the initial parent team will be created in the Channel Applications Manager first.

When selecting the Customer Access Mode:

- If Assigned Customers is selected, the ability to assign specific customers to the user team is provided in this screen.
- If User To Customer Assignment Required is selected, the ability to assign customers to the user team is provided in this screen. It is expected that customers assigned to the team will be assigned to specific users on the team later.
- If All Customers is selected, the portion of the screen that enables you to assign customers to the team will not be visible, and the team will automatically have access to all the customers.

### Assign Customers to the Team

To assign customers to a user team, the user enters the corresponding customer search criteria in the Find Existing Customer panel. The `getCustomerList` API is called to retrieve the customer records, and the results are displayed in the Customer Search Results panel. The user can then select customers to assign to the user team. For more information about the `getCustomerList` API, refer to the *Sterling Call Center and Sterling Store: Javadocs*.

### Save the Team

The Assigned Customers panel is not populated initially. After the user completes and confirms the entries in the Create User Team screen, the `manageTeam` API saves the user team record and generates a `TeamKey`. If the user has assigned customers to the user team, the `manageCustomerAssignment` API is called, which saves the customer assignments against the newly created user team.

#### 4.21.1.2 End-User Impact

None.

#### 4.21.1.3 Implementation

This task is permission controlled. You can launch this task only if you belong to a user group that is assigned the necessary permissions.

#### 4.21.1.4 Reference Implementation

None.

### 4.21.2 Manage a User Team

Sterling Call Center and Sterling Store representatives may have to manage user teams.

#### 4.21.2.1 Solution

Sterling Call Center and Sterling Store provides a user interface task that enables users to manage user teams. This task is similar to the Create User Team task, except that the user must first search for the team.

This section explains the wizard-based Manage User Team task and the solution offered by Sterling Call Center and Sterling Store.

#### Search for the User Team

The first screen in this wizard-based task is a search screen that enables the user to search for an existing user team. The team search criteria are Team ID, Description, Parent Team, and Organization. The search is performed using the `getTeamList` API call.

---

**Note:** The Organization field is displayed only if the user has access to more than one customer master organization.

---

If only one search result is returned, the user is taken directly to the next screen, where the user can make changes to the user team. If multiple results are returned, the user must first select a team from the list before being taken to the next screen. The `getTeamList` API retrieves and displays the Team ID, Description, Parent Team, and Customer Access Mode for the selected team. The `getCustomerAssignmentList` API retrieves and displays all the customers assigned to the team.

### Manage the User Team

After the team is selected, the next screen in this wizard-based task enables the user to make modifications to the team details.

The Manage User Team screen is almost identical to the Create User Team screen, with the following differences:

- The Team ID field is disabled.

**Note:** This field is enabled for editing only during team creation.

- The Organization field is not displayed.
- The screen is populated with team and customer assignment information as a result of the `getTeamList` API and the `getCustomerAssignmentList` API being called.
- When the user clicks the Confirm button, the `manageTeam` API is called with a `TeamKey`, which updates the existing user team record instead of creating a new user team record.

#### 4.21.2.2 End-User Impact

None.

#### 4.21.2.3 Implementation

See [Section 4.21.1.3, "Implementation"](#).

#### 4.21.2.4 Reference Implementation

None.

## 4.22 Pagination

When users perform inquiries, they need the information to be retrieved in a minimum amount of time and to be displayed as efficiently as possible.



### 4.22.1 Solution

Sterling Call Center and Sterling Store supports smart retrieval of record sets in all of its search and list screens. These include:

- Order search
- Price Match search
- Competitor search
- Alert search
- Customer search
- Item search
- Return order search
- Advanced shipment search

For large record sets, smart retrieval enables a CSR to display the records as a full, complete set. Pagination is performed by using the YRCPaginationData object, which transforms an API call into a getPage call, and retrieves the first page of records for display. If more records that fit the criteria exist, the CSR can click the Get More Records link to retrieve them, one list of records at a time. Each time this is done, records are appended to the existing record list, until the complete set of records is displayed. The most recent set of records returned is highlighted. In addition, in the Order Search and Return Order Search tasks, an Order By pull-down window allows for sorting returned data by Order Number, Order Date, or Customer Last Name.

The Order Search and Return Order Search tasks also enable the CSR to view history orders. This is performed through a link on the display, which toggles between displaying recent orders and history orders. This requires two different YRCPaginationData objects, as described in the *Selling and Fulfillment Foundation: Customization Guide*.

### 4.22.2 End-User Impact

Record display screens have different links, depending on how record handling is configured. The CSR might see the following links at the bottom of the screen:

- Total Available link - can be configured to allow users to request the total number of records to display, to always display the total number

of records, or to never display the total number of records. If configured so that the user can request the total number of records, as the CSR repeatedly clicks on the Get More Records link, each successive set of records is displayed until none are left.

- View History Orders/View Recent Orders - this link is applicable only to Order Search and Return Order Search tasks. If configured, this link toggles between automatically displaying the total number of history records available for the search criteria and displaying recent order records that fit the search criteria.

### 4.22.3 Implementation

Pagination uses the following configuration options:

- When a CSR performs searches, the number of records retrieved is specified by the following properties in the `yfs_propertierties_ysc_ext` file:
  - `yfs.rcp.ui.PageSize` - number of records to retrieve
  - `yfs.ui.queryTimeout` - number of seconds for each SQL statement to execute before console timeout

The default values for these properties are extensible through the `YRCPaginationData` object, as described in the *Selling and Fulfillment Foundation: Customization Guide*.

- Configure pagination rules as described in the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

**Note:** Sterling Call Center and Sterling Store recommends that you do not configure pagination to automatically display the total number of records. This count can affect the performance of the application server as well as all of the clients.

### 4.22.4 Reference Implementation

None.

## 4.23 Alert and Queue Management

The Alert Management task provides the ability for the user to create, resolve, and assign alerts. Alerts are generated automatically when the system raises certain events happen, or they may be manually created by the users. For example, when a payment authorization fails for an order, the system generates an alert. The user can then call the customer and request for a different payment method. Alternatively, the user can create an alert for the supervisor when the customer requests a price match that exceeds the 'No Hassle' price match check. Sterling Call Center and Sterling Store provides the comprehensive visibility to alerts assigned to users, move alerts to a different queue, and so on. Alerts are also referred to as exceptions.

### 4.23.1 Solution

Sterling Call Center and Sterling Store provides features that enable you to create appropriate types of alerts and process them in a timely and efficient manner.

You can configure whether to notify user whenever an alert is assigned to the current user or notify users whenever an alert is added to a queue the user is subscribed. The `getExceptionListForOrder` API is used to poll for new alerts in the Sterling Call Center and Sterling Store client.

If you have configured to notify users whenever an alert is added to a queue the user is subscribed, alert notifications will notify a user when an alert is created in the user's queues in the last polling time. The user will not be notified of alerts moved to the user's queues during that time.

If you have configured to notify user whenever an alert is assigned to the current user, the user will be notified of alerts assigned to them since the last polling time.

#### Alert Creation

Sterling Call Center and Sterling Store enables users to create alerts in the system.

Users who have access to multiple enterprises can create alerts for a specific enterprise by selecting the corresponding enterprise from the Enterprise drop-down list in the Create Alert screen.

Using the Alert Type configuration, you can define the types of alerts that can be raised in a system. Users can then select the appropriate alert

type from the Alert Type drop-down list in the Create Alert screen. The `getExceptionTypeList` API is called to retrieve the list of alert types. Users can also assign a priority for an alert.

Sterling Call Center and Sterling Store enable you to display a default description for an alert type that is selected. For more information about displaying the default description for an alert type, see [Section 4.23.3, "Implementation"](#).

Queues can be used to group related alerts. The Queue drop-down list will display the list of queues available for the DEFAULT enterprise and the enterprise selected by the user from the Enterprise drop-down list. The `getQueueList` API is called to retrieve the list of queues.

Alerts can also be assigned to internal users during alert creation. If a user is specified while creating the alert, the user will be validated using the `getUserList` API.

The `createException` API is called to create the alert.


### Designing Custom Alert Resolution Panels

Sterling Call Center and Sterling Store provides visibility to alerts and resolve them using the Alert Details screen. This screen is unique and displays different alert resolution panels in the screen depending on the alert type that is being viewed. For example, if you are viewing a duplicate order alert (an alert raised when the system detects that two or more orders have been entered in the system that may be duplicates because they have similar attributes), the Alert Detail screen lists the orders that are potential duplicates so that the users can view the order details and cancel appropriate orders.

Sterling Call Center and Sterling Store provides pre-defined alert types. Additionally, you can define new resolution panels for your own custom alert types.

### Alert Notification

Sterling Call Center and Sterling Store notifies a user about the latest Priority 1 alerts that are open. The user is notified of the alerts in one of the following ways:

- The  (alerts) icon is displayed on the CSR message panel, which also displays:
  - Open alerts that are assigned to the current user.

- Priority 1 alerts that are added to the queues to which the current user is subscribed.

The `getExceptionListForOrder` API is called to retrieve all open and priority 1 alerts, consolidate them, and display the number of open alerts.

---

**Note:** If there are no priority 1 alerts assigned to the queue to which the user is subscribed, the total number of follow-up alerts are displayed.

---

- The Alert Notification panel is displayed at the specified time interval based on the configuration. This is indicated by a beep. The `getExceptionListForOrder` API is called to retrieve the latest open and priority 1 alerts, consolidate them, and display the number of open alerts.

Based on user permissions, the Do Not Notify check box is either displayed or hidden in the Alert Notification panel.

## 4.23.2 End-User Impact

The users can view popup notifications of new alerts based on the configuration. Only users with necessary permissions can turn off further alert notifications.

## 4.23.3 Implementation

- You can define alert types as needed. When configuring your alert type, in the alert type configuration, you can specify a java class that is an instance of the SWT composite class (`org.eclipse.swt.widgets.Composite`) as the resolution form. You must create this composite and include components that helps the users to resolve alerts for that specific alert type. You can call APIs within the composite if you require additional information in that panel. If you do not want to design a new composite for any of the custom alert types, you can leave the Resolution Form field blank in the alert type configuration. If you do not enter any information in this field, the default resolution panel are used. The default resolution panel displays the data from the Alert Details column of the alert table (YFS\_INBOX).

- In the Details panel, you can display a default description for an alert type that is selected by performing the following steps:
  1. Create a bundle entry for the alert type in the `com.yantra.pca.ycd_bundle.properties` bundle file present in the `<INSTALL_DIR>` folder.
  2. The key of the bundle entry must be of the format, `<Alert Type>_DEFAULT_DESC`. For example, if the alert type is `YCD_VERIFY_ADDRESS`, the bundle key must be named as `YCD_VERIFY_ADDRESS_DEFAULT_DESC`.
  3. Enter a value for the bundle key that has been created. The value is the description that will automatically be populated in the Details panel of the Create Alert screen, when the alert type is selected. However, if you have not entered any value for the bundle key, the Details panel will not display any description.
- You can group similar types of alerts into queues and assign them to users. For more information about assigning alerts and queues, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Configure the following rules to notify users about the alerts raised:
  - Notify Users Whenever an Alert is Assigned to the Current User
  - Notify Users Whenever an Alert is Added to a Queue The User is Subscribed
  - Never Notify Users About Alerts
  - Poll for New Alerts Every `<number_of_minutes>` Minutes

For more information about alert notification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

- You can provide permissions to users to enable them to turn off further alert notifications by selecting the Do Not Notify check box in the Alert Notification panel, if necessary. For more information about administering user group permissions, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
- Notes entered for this task are saved with `YCD_CREATE_ALERT` note type.

#### 4.23.4 Reference Implementation

By default, the following alert types are provided:

- YCD\_VERIFY\_ADDRESS
- YCD\_PAYMENT\_FAILURE
- YCD\_DUPLICATE\_ORDER
- YCD\_ORDER\_CANCEL\_NOTICE
- YCD\_PRICE\_MATCH
- YCD\_RESHIP
- YCD\_STOP\_DELIVERY

The resolution types provided for the following alert types are:

- YCD\_VERIFY\_ADDRESS
- YCD\_PAYMENT\_FAILURE
- YCD\_DUPLICATE\_ORDER
- YCD\_ORDER\_CANCEL\_NOTICE

Alerts types, queues, and queue subscriptions are defined for all the organizations.

By default, the following note type is provided for all the organizations:

- YCD\_CREATE\_ALERT

By default, the Do Not Notify check box in the Alert Notification panel is displayed to all users.

#### 4.24 Announcements

CSRs may sometimes want to provide useful information or timely messages to other CSRs, for example, to convey the information that the delivery of goods to a certain region is delayed because of bad weather conditions. In such situations, users may want to relay the information to other users.

## 4.24.1 Solution

Sterling Call Center and Sterling Store provides a feature that enables users to create an announcement to post messages or useful information to other users. Users can view the announcements in the Announcement panel provided in the Quick Access screen. Users will also be notified when a new announcement is created.

### Create Announcement

Users who belong to a user group that has the necessary permissions can create announcements. The announcements that are created are stored as alerts. When an announcement is created, the alert type that is saved on the announcement is determined by the alert type used in the Configuring Announcements rule.


An announcement can be created for all the users or specific users, using the concept of queues. When creating an announcement, the user must select the queue for which the announcement has to be created. The list of available queues that a user can select from the Send To drop-down list while creating an announcement is determined through the Queue Group specified on the queue. All the available queues are assumed to have the same priority. In order to display the queue in the Send To drop-down list, the Queue Group must be set to "Announcement Queue Group" and the priority must be set to 1. All users who are subscribed to the queue will be able to view the announcement.

When an announcement is created, the expiration days for an announcement is set to the difference between the Expiration Date and the date when the announcement is created. If this difference is lesser than or equal to 0, the value of the expiration days will be set to 1. If the difference is greater than 0, the value of the expiration days will be set to the difference.

The createException API is called to create the announcement.

### Announcement Notification

The Sterling Call Center and Sterling Store application notifies users about the new announcements that are available. A user can view an announcement only if the announcement has been created for a queue to which the user is subscribed.

An  (announcements) icon is displayed on the CSR Message panel, and the number adjacent to the icon is periodically refreshed based on the



polling interval specified in the Channel Applications Manager. The Announcements hyperlink displays the number of latest announcements available since the last polling interval.

The `getExceptionList` API is called to retrieve the latest announcements available since the last polling interval.

### Viewing and Closing Announcements

Users can view the list of announcements created in the Announcements panel of the Quick Access screen. The `getExceptionList` API is called to retrieve the list of available announcements that are created in the queues to which the user is subscribed. Announcements that have expired since the last polling interval will not be removed from the Announcements panel until the user restarts the application.

Users can view the Announcement Details screen by clicking the View Details hyperlink on the corresponding announcement. The `getExceptionDetails` API is called to display the Announcement Details screen. Additionally, users who belong to a particular user group that has the necessary permissions can cancel an announcement from the Announcement Details screen.

The `resolveException` API is called to cancel the announcement. Announcements that are cancelled will be removed from the Announcements panel when the next polling occurs, or when the user restarts the application.

### Purging Announcements

When an announcement is created, it is stored in the `YFS_INBOX` table. Since the entries in the `YFS_INBOX` table are purged by the `INBOXPRG` purge agent, announcements will also be purged when the `INBOXPRG` agent is executed. The purge criteria for announcements depends on the configurations defined for `INBOXPRG` purge agent. For more information about `INBOXPRG` purge agent, see *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

## 4.24.2 End-User Impact

A user can create, view, and cancel announcements if the user belongs to a user group that has the corresponding permissions.

### 4.24.3 Implementation

This section explains the configurations for the Announcements task.

- You can define the queues for which an announcement has to be created. While defining queues, ensure that the Queue Group is set to "Announcement Queue Group" and the priority of the queue is set to 1. For more information about defining queues, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
- Sterling Call Center and Sterling Store enables you to configure the alert type that will be saved on the announcements when they are created. For more information about configuring alert types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store enables you to configure the polling interval for announcements. For more information about configuring the polling interval, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Configure the INBOXPRG purge agent to purge the announcements.
- The Announcements task is permission controlled. You can assign permissions to user groups for creating, viewing, and cancelling announcements.

### 4.24.4 Reference Implementation

This section explains the reference implementation provided for the Announcement task.

- By default, CSR Leads and Store Managers are provided permissions to create announcements.
- By default, permission to view the announcements are provided to all the users.
- By default, a queue with the YCD\_ANNOUNCEMENT\_QUEUE queue ID, and for which an announcement has to be created, is provided.
- By default, an Announcement alert type is provided that will be used while creating an announcement.

## 4.25 User Preferences

When viewing the Sterling Call Center and Sterling Store application, the user may want to personalize the appearance of the screen. A novice user may also wish to view additional help messages while performing a task.

### 4.25.1 Solution

Sterling Call Center and Sterling Store provides a user interface task to set user preferences. The user can set preferences such as display of alert notifications, additional help messages, keyboard shortcuts, customer message panel, CSR message/customer message panel, position of scratch pad, and the position of related tasks. The user can also set preferences to remember pagination preference, and to display the scratch pad soon after logging in to the application.

The `getProperty` API is called to retrieve the preferences of the user when the user preferences screen is opened. The following properties are updated:

- `YCDAlertNotification`—To display alert notifications
- `YCDAddtnlHlpMsgsIcons`—To display additional help messages and icon legends
- `YCDRelatedTaskShortcut`—To display keyboard shortcuts
- `YCDCustomerCSRMessages`—To display collapsible customer message panel
- `YCDSavePaginationSort`—To remember the pagination preferences
- `YCDRelatedTaskPosition`—To position the related task panel
- `YCDCustomrCSRMsgPnlPsn`—To position the CSR message panel
- `YCDScratchPadDisplay`—To display the scratch pad in the Related Tasks panel or in a pop-up window
- `YCDScratchPadEnabled`—To display the scratch pad after logging in to the application

The `manageProperty` API is called to update the properties.

## 4.25.2 End-User Impact

This section explains the end-user impact for setting the user preferences.

- Based on the preferences set, the user will see alert notifications, additional help messages, keyboard shortcuts.
- The related tasks, CSR message/customer message panels, and scratch pad are positioned based on the user preferences.
- Based on the preferences set, the user will be notified when new records are inserted in an already sorted or filtered table.
- Based on the preference set, the user will view the scratch pad soon after logging in to the application.

## 4.25.3 Implementation

None.

## 4.25.4 Reference Implementation

This section explains the reference implementation provided as part of the setting user preferences task.

- By default, the Alert Notification panel is displayed.
- By default, the Additional Helpful Text Messages are displayed.
- By default, the keyboard shortcuts for the related tasks are displayed.
- By default, the Customer Message panel is displayed.
- By default, the position of CSR Message/Customer Message panel is Top.
- By default, the Pagination Preference is not saved.
- By default, the Scratch Pad is displayed within the Related Task panel.
- By default, the Scratch Pad is displayed soon after a user logs in to the application.
- By default, the position of the Related Tasks panel is Left.

## 4.26 Viewing the Documentation Library

When using the Sterling Call Center and Sterling Store application, users may want to access the product documentation library to get information about the Sterling Call Center and Sterling Store application from the Sterling Call Center and Sterling Store documents.

### 4.26.1 Solution

Sterling Call Center and Sterling Store enables users to view the product documentation library to get additional information about the Sterling Call Center and Sterling Store application. The documentation library can be hosted either by Sterling Commerce or locally by your enterprise. During installation of Selling and Fulfillment Foundation, you can select whether you want to enable an Online Documentation Library or a Local Documentation Library. You can also change this selection after installation. For more information about selecting the type of documentation library, refer to the *Selling and Fulfillment Foundation: Installation Guide*.

Users can access the documentation library by selecting the Documentation Library option from the Help Menu.

### 4.26.2 End-User Impact

A user can access the documentation library if the user belongs to a user group that has the permission to view the product documentation.

### 4.26.3 Implementation

This task is permission controlled. Sterling Call Center and Sterling Store enables you to assign permissions to user groups for this task.

### 4.26.4 Reference Implementation

As part of reference implementation, Sterling Call Center and Sterling Store provides permissions to all the user groups to view the product documentation.

## 4.27 Deploying the Sterling Call Center Clients in Phases

As new versions of Sterling Call Center and Sterling Store introduce new business processes and capabilities, some enterprises may want to deploy the new versions of the application to only specific users, and retain the earlier version of the application for other users. Users accessing an earlier version of the application should be allowed to access only those business processes and capabilities pertaining to the version the user is currently using. For example, an enterprise may want to display the Basic Fulfillment Summary Screen to users who are using an earlier version of the application and display the Advanced Fulfillment Summary Screen to users who are using the new version of the application. Also, by deploying the new version of the application to only specific users, the enterprise will have more time to train existing users on the new functionalities.

### 4.27.1 Solution

Sterling Call Center and Sterling Store enables enterprises to deploy the application in a phased manner. When a new version of the application is released, the application server must be updated with that version of the application. The new version of the application client can be made available for specific users, while the earlier version of the application client can be used by the other users. The functionality that is available to the users is based on the version of the application deployed on the clients.

To ensure that multiple versions of the client can access the application server, perform the following tasks:

- If you want to access the application server using Release 7.4, 7.5, or 8.0 of the client, modify the `locations.ycfg` file of the client accordingly. For example, if you are using Release 7.5 of the client to access an application server that has Selling and Fulfillment Foundation, Release 9.0, installed, modify the `locations.ycfg` file of the Release 7.5 client to make it compatible with Selling and Fulfillment Foundation, Release 9.0. For more information about modifying the `locations.ycfg` file, see the *Sterling Call Center and Sterling Store: Deployment Guide*.

- If you are trying to access the application server using Release 7.4, 7.5, 8.0, or 8.5 of the client, copy the `commands_<COMMANDS_VERSION>.ycml` file to the `<INSTALL_DIR>\repository\xapi\template\merged\com.yantra.yfcrpc\commands\<module>` directory.

Here, `<COMMANDS_VERSION>` refers to the version of the client application that is specified in the `client.properties` file of the client application, and `<module>` refers to either YFSSYS00011 (Sterling Call Center) or YFSSYS00006 (Sterling Store).

Re-create the `resources.jar`. Rebuild the Selling and Fulfillment Foundation Enterprise ARchive (EAR) package and redeploy it. For more information about re-creating the `resources.jar` and EAR, see the *Selling and Fulfillment Foundation: Installation Guide*.

### Version-Specific Business Rules

Sterling Call Center and Sterling Store enables you to configure different values for specific rules depending on the version of the application accessing the rule. The following rules can be configured in different ways based on the application's version:

- Configuring the way the Fulfillment Summary screen is displayed. For more information about configuring the display of the Fulfillment Summary screen, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Changing the tax options for a business customer. For more information about changing the tax options for a business customer, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Scheduling the rule for the Additional Shipping option. For more information about selecting the scheduling rule, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

For more information about configuring different values for a rule. See the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### Compatibility of Order with Different Application Client Versions

If the application client is deployed in phases, two different versions of the application client may simultaneously access the application server. Therefore, orders created in the new version of the application client might have features on the order that make it unusable in the earlier version of the application client. To prevent order modifications in an earlier version of the application that is incompatible with the order, you can define a qualifier for the order. A qualifier can be defined over a range of client application versions. The qualifier determines the compatibility level of the order with a given version of the application. For more information about defining qualifiers for an order, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

Based on the qualifier defined for an order, the `getCompleteOrderDetails` API returns a `SupportLevel` attribute that indicates whether the order is compatible with the version of the application accessing the order. For more information about the display of orders based on compatibility levels, see [Order Summary](#). For more information about the `getCompleteOrderDetails` API, refer to the *Sterling Call Center and Sterling Store: Javadocs*.

### Version-Specific Common Codes

Sterling Call Center and Sterling Store enables you to define common codes that are specific to a particular version of the application client. An enterprise may introduce a new common code that is applicable to only a particular version. In order to implement this, you can define qualifiers for the common codes. For more information about defining qualifiers for the common codes, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*. For more information about configuring the values of the common codes based on the version of the application, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. The `CANCEL_REASON` common code type can have values that are specific to a version of the application.

## 4.27.2 End-User Impact

This section explains the end-user impact if the Sterling Call Center and Sterling Store application is deployed in phases.



- If different values have been defined for a specific rule in different versions, users will see different types of behavior in the application depending on the version of the application the user is using.
- Based on the compatibility level of an order with a particular version of the application client, the appropriate Order Summary screen is displayed. For more information about the different compatibility levels of an order, see [Order Summary](#).
- Based on the common code values that are configured for a particular version of the application, appropriate common codes are displayed depending on the version of the application being used.

### 4.27.3 Implementation

This section explains the configuration for deploying the Sterling Call Center and Sterling Store application in phases.

- Sterling Call Center and Sterling Store enables you to define qualifiers for a rule. For more information about defining qualifiers for a rule, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*. For more information about configuring different values for a rule, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Sterling Call Center and Sterling Store enables you to define qualifiers for an order to indicate the compatibility of an order with a particular version of the application. For more information about defining qualifiers, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
- Sterling Call Center and Sterling Store enables you to define qualifiers for the common codes. For more information about defining qualifiers for the common codes, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*. For more information about configuring different values for the common codes, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 4.27.4 Reference Implementation

None.

## 4.28 Helpful Text Messages

When using the Sterling Call Center and Sterling Store application, a novice user may want to view additional help messages which guide the user to perform a task.

### 4.28.1 Solution

Sterling Call Center and Sterling Store provides a help assistant screen for each task. The Sterling Call Center and Sterling Store Assistant is launched when a screen is loaded and it displays a hyperlink for each task that can be performed on the corresponding screen. On clicking a hyperlink, detailed instructions for the selected task is displayed. Help Assistant is available for the following screens:

- Create Order: Select Customer
- Add Lines to Order
- Change Fulfillment Options
- Pickup Select Store Pop-up
- Take Appointment
- Take Appointment: Resolve Line Problems
- Delivery Fulfillment Summary
- Ship/Pickup Fulfillment Summary
- Price Match: Price Match Details
- Create Return: Select Lines
- Create Return: Return Reason
- Create Return: Fulfillment Summary
- Create Return: Take Appointment
- Create Return: Change Appointment

The content for helpful text messages may reside on a different server and can be retrieved when needed. You can configure the server URL where the content for helpful text messages reside. A `taskhelp.xml` is provided which contains the help content for each screen.

The `taskhelp.xml` must adhere to the following structure:

```

<Screens>
<Screen ID=" ">
<HelpTasks>
<HelpTask TaskID=" " BundleKey=" " Url=" "/>
</HelpTasks>
</Screen>
</Screens>

```

Sterling Call Center and Sterling Store also allows you to add more questions to a screen for which content is already provided. You can also customize and localize the titles that appear on the Help Assistant screen.

Sterling Call Center and Sterling Store also allows you to add content for new screens.

The user can configure whether the help assistant screen should be displayed when the screens are launched. For more details on user preferences, see [Section 4.25, "User Preferences"](#).

## 4.28.2 End-User Impact

Based on user preferences, the user will see additional helpful text messages.

## 4.28.3 Implementation

This section explains the configurations for helpful text messages.

- Sterling Call Center and Sterling Store allows you to configure the server URL from where the content for the Help Assistant screens is retrieved. To configure the URL, set the value for the `yfs.ycd.help.baseurl` property in the `<INSTALL_DIR>/properties/customer_overrides.properties` file.
- Sterling Call Center and Sterling Store allows you to add content for existing screens.

### To add more questions to an existing screen:

- a. Create a new folder where the customized file will be located.
- b. Copy the `reference\help\taskhelp.xml` file to the new folder created in [Step a](#).

- c. Modify the `taskhelp.xml` file to include additional `HelpTask` elements under the screen for which more helpful messages are added.
- d. In the `plugin.xml` file, implement the extension point `com.yantra.pca.ycd.rcp.YCDTaskHelpContent`. Create an extension `HelpFileName`. In the `Name` field, specify the relative path to the new file created in [Step b](#).
- Sterling Call Center and Sterling Store allows you to customize and localize the titles that appear on the Help Assistant screen. For more information about customizing the titles that appear on the Help Assistant screens, see [Section A.2.1, "Resource Bundles"](#).

### 4.28.4 Reference Implementation

None.

## 4.29 Integrating with Business Center

In some implementations, call center and store representatives may have to access screens and functionality from the Business Center application. For example, call center and store representatives may have to configure pricing rules in Business Center.

### 4.29.1 Solution

Sterling Call Center and Sterling Store provides a user interface task that enables users to launch Business Center from within Sterling Call Center and Sterling Store. When the task is selected, the Business Center application is launched in an embedded Web browser. The location of the application server running the Business Center must be defined in the `locations.ycfg` file. For more information about the `locations.ycfg` file, refer to the *Sterling Call Center and Sterling Store: Installation Guide*.

Sterling Call Center and Sterling Store provides a single sign-on capability to Business Center, enabling the user to access both the Item Administration and Pricing Administration applications. To log in, the user must belong to the following:

- A user group that has permission to launch Business Center from within Sterling Call Center and Sterling Store. This user interface task is permission controlled.
- A user group that has permission to the Business Center application and its screens.
- A menu group that can access both the Sterling Call Center and Sterling Store and Sterling Business Center.

For information about using Business Center, refer to the *Sterling Business Center Pricing Administration Guide* and the *Sterling Business Center Item Administration Guide*.

### 4.29.2 End-User Impact

Users will not be able to access Business Center unless they belong to the menu group and user groups described in [Section 4.29.1, "Solution"](#).

### 4.29.3 Implementation

This section explains the configurations for the Business Center task.

- Business Center can be deployed on an application server that is different from the Sterling Call Center and Sterling Store server. To enable the launch of a Web browser for Business Center, the location of the Business Center application server must be defined in the `locations.ycfg` file. For more information about the `locations.ycfg` file, refer to the *Sterling Call Center and Sterling Store: Installation Guide*.
- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign the necessary permissions to the corresponding user groups for performing this task.

### 4.29.4 Reference Implementation

None.



# Implementing the Sterling Store Features

---

This chapter describes the store features supported by Sterling Store.

## 5.1 Item Inquiry

A user may want to search for product items or service items and also view the details for them such as fulfillment methods, availability, promotions, associations, price. An enterprise may want to model items based on distinct attributes such as style, size and color and enable users to search and view details of stylized items. For more information about item inquiry, see [Section 4.1, "Item Inquiry"](#).

## 5.2 Configured Items

In some retail environments may fulfill orders comprising of configured items. A configured item is a group of items which are related such as a computer which comprises of the processor, keyboard, mouse, and monitor. Configured items are also referred to as bundles. For more information about configured items, see [Section 4.2, "Configured Items"](#).

Bundles are supported in the following flows:

- [Shipment Inquiry](#)
- [Backroom Pick](#)
- [Undo Backroom Pick](#)
- [Customer Pick](#)

## 5.3 Alternate Item Identifier

The alternate item identifier can be used to scan and search for product or service items in the store application. For more information about alternate item identifiers, see [Section 4.3, "Alternate Item Identifier"](#).

The Order Print tasks provided by Sterling Store do not display alternate item identifiers, even if you have configured alternate item identifiers. For more information about order prints, see [Section 5.28, "Order Print Tasks"](#).

## 5.4 Item Display Options Configuration

Sterling Call Center and Sterling Store enables you to configure which item attributes need to be displayed throughout the user interface. For more information about configuring item display options, see [Section 4.4, "Item Display Options Configuration"](#).

## 5.5 Alternate Store Pickup Locations Search

Sterling Call Center and Sterling Store provides an ability to search for availability of items for pickup across stores. For more information about alternate stores, see [Section 4.5, "Alternate Store Pickup Locations Search"](#).

## 5.6 Order Capture

Data must be captured for fulfilling an order in a store. The order capture process includes order creation and order validation. For more information about order capture, see [Section 4.6, "Order Capture"](#).

### **Adding Open Box Items**

Sterling Store provides the ability to add open box items to an order and to view the open box items on existing orders. For more information about open box items, see [Section 5.9.7, "Open Box Items"](#).

## 5.7 Order Inquiry

A customer may walk into a store to inquire about any order placed in the store. In such situations, Sterling Call Center and Sterling Store



needs to respond to order inquiries by publishing information about the orders captured to the external systems. The order inquiry process and implementation supported in a store is similar to the order inquiry process in a call center.

The order inquiry process in the Sterling Store scenario enables you to search for draft orders and orders that are created in a store. For more information about order inquiry, see [Section 4.7, "Order Inquiry"](#).

## 5.8 Order Monitoring

Once orders are captured, they need to be monitored so that the customers are notified about the order status. For more information about order monitoring, see [Section 4.8, "Order Monitoring"](#).

## 5.9 Order Maintenance

You can modify orders that are created and captured in Sterling Call Center and Sterling Store. The following order maintenance features are supported in the Sterling Store scenario:

- [Apply Hold](#)
- [Resolve Holds](#)
- [Change Payment Method](#)
- [Add Multiple Items to an Order](#)
- [Change Gift Options](#)
- [Customer Appeasement](#)
- [Open Box Items](#)
- [Add Coupon](#)
- [Cancel Order](#)
- [Track an Item](#)
- [Return Order Capture](#)
- [Order Notes](#)
- [Change Fulfillment Options](#)
- [Change Service Instructions](#)

- [Change Service Appointments](#)
- [Fulfillment Summary](#)
- [Reservations](#)
- [Price Match](#)
- [Increase Order Line Quantity](#)
- [Schedule and Release an Order](#)
- [View Procurement Orders](#)
- [Change Order Address](#)
- [Launch Application Console](#)
- [Change an Item's Style](#)
- [Change Tax Options](#)
- [Copy Order](#)
- [E-Mail Order Information](#)
- [Transfer Draft Order to Web](#)

### 5.9.1 Apply Hold

Sterling Call Center and Sterling Store enables users to apply holds on an order after order creation. Based on user permissions and status modification rules, users can apply holds on an entire order or a specific order line. For more information about applying holds, see [Section 4.9.1, "Apply Hold"](#).

### 5.9.2 Resolve Holds

Sterling Call Center and Sterling Store provides visibility to holds applied to an entire order or to just certain order lines, and enables you to remove them. An order can be placed on hold with various hold types, either automatically or manually. For more information about removing holds, see [Section 4.9.2, "Remove Holds"](#).

### 5.9.3 Change Payment Method

Different types of payment are supported in the Sterling Store scenario. For more information about the process and implementation of changing a payment method, see [Section 4.9.3, "Change Payment Method"](#).

### 5.9.4 Add Multiple Items to an Order

Once an order is created, the customer may want to add a few more product or service items to that order. Sterling Store enables you to add multiple items to an order. For more information about adding multiple items, see [Section 4.9.5, "Add Multiple Items to an Order"](#).

### 5.9.5 Change Gift Options

You can identify certain items as gifts in an order. You can configure which of the fulfillment methods allow gift options to be recorded in the user interface. For more information about changing gift options, see [Section 4.9.6, "Change Gift Options"](#).

### 5.9.6 Customer Appeasement

When a customer is not satisfied or has had a bad experience with any of the services that were provided, you can present them with the option to appease them. For example, a customer may walk in to a store to report having received a damaged item. In such situations, you need to perform an appeasement task to make amends for the bad experience. For more information about customer appeasement, see [Section 4.9.7, "Customer Appeasement"](#).

### 5.9.7 Open Box Items

Open box items are display items that are returned with minute defects and are sold at a discounted rate. You can add the open box items to an order.

#### 5.9.7.1 Solution

Sterling Call Center and Sterling Store provides the ability to add open box items to an order and to view the open box items on existing orders.

Typically, open box items are assigned a unique identifier to differentiate them from the other items. In Sterling Call Center and Sterling Store, the

unique identifiers for the available open box items are stored as item tag attributes. For more information about defining a product item's inventory information, refer to the *Catalog Management: Configuration Guide*.

Open box items must be assigned a different product class to ensure that they do not appear in normal item availability inquiries and are not promised to orders for regular items.

Open box identifiers can be entered in the user interface in the create order and add line screens. The `getCompleteOrderDetails` API retrieves the sales order details that contain the `OrderLineInvAttRequest` element as an output template. The user interface uses the `getTagListForOrdering` API to validate the open box ID and get the corresponding item information for that open box ID such as the Item ID, Unit of Measure, Product Class, and Price Details. The `createOrder` or `changeOrder` APIs are called with open box unique identifiers and passed to the `OrderLineInvAttRequest` element as input.

### 5.9.7.2 End-User Impact

None.

### 5.9.7.3 Implementation

This section explains the configuration needed to add an open box item to an order.

1. Log in to the Channel Applications Manager. For more information about logging in to the application, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
2. In the Applications Manager, set the following rules:
  - Allow Addition of Open Box Items to Orders—Enable this rule to allow open box items to be added to orders.
  - Tag Identifier for Open Box ID—Select the appropriate item tag attribute that is used to track the unique open box identifier from the drop-down list.

---

**Note:** In the user role configuration, you can grant or revoke permission to the "Add Open Box" resource if you want to control the capability to add open box items at a user level.

---

- **Product Class for Open Box Items**—Select the appropriate product class that represents open box items from the drop-down list. This product class is used to differentiate open box items from regular availability checks and order promising.
- **Fulfillment Methods Supported for Open Box Items**—Select the appropriate fulfillment method to use for open box items. The following types of fulfillment method are supported for open box items:
  - Pickup
  - Shipping
  - Delivery

#### 5.9.7.4 Reference Implementation

Open box items are provided as part of reference implementation. For more information about open box items, see the *Sterling Selling and Fulfillment Suite: Applications Reference Implementation Guide*.

### 5.9.8 Add Coupon

This feature allows a user to add coupons or promotion codes for an order that has been placed. For more information about adding coupons to orders, see [Section 4.9.15, "Add Coupon"](#).

### 5.9.9 Cancel Order

A customer may want to cancel an entire order, or just certain product or service items on the order. The items on an order can be cancelled as long as they have not been shipped. For example, you can cancel items that are included in the shipment but still present in the store.

In the Sterling Store scenario, depending on the store enterprise configurations, a store representative can cancel the order lines that are not shipped from a local store, even if permissions are not assigned. The

getCompleteOrderDetails API determines whether or not an order line can be cancelled. If the cancellation request has some items beyond Released status, the store representative must first validate if the items can be cancelled. The order lines can be cancelled even after the order lines with the pick up fulfillment method are moved to the Released status. The changeOrder API is called to cancel the lines on the order and release, if associated with the order line. The changeShipment API is called to cancel the shipment lines associated with the order line. This API is called only when the shipment line is associated with an order line.

As a part of Cancel Order task implementation:

1. Log in to the Channel Applications Manager.  
For more information about logging in to the Channel Applications Manager, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
2. From the Channel Applications Manager, select Configure Order Administration > Configure Order Modification Rules. The Modification Rules: Order Fulfillment screen is displayed.
3. In Group By drop-down list, select Modification Type.
4. From Order Fulfillment > Add Note > Line, right-click Sent To Store.
5. Select Allow Modification.
6. From Order Fulfillment > Add Note > Order, right-click Sent To Store.
7. Select Allow Modification.

For more information about cancelling an order, see [Section 4.9.16, "Cancel Order"](#).

### 5.9.10 Track an Item

The customers may sometimes need to know the status of the orders. For example, if a customer does not receive items that were ordered, then the customer can call the CSR to enquire about the shipment status. For more information about tracking an item, see [Section 4.9.17, "Track an Item"](#).

### 5.9.11 Order Notes

A user may need to enter notes to an order or order line, describing the actions performed against that order or order line, for future reference.

As part of the reference implementation, the following Contact Types are configured:

- Phone
- E-mail
- In Person

For more information about adding notes to an order, see [Section 4.9.18, "Order Notes"](#).

### 5.9.12 Change Fulfillment Options

You can use this task to change fulfillment options for orders that are associated with the delivery lines.

The `getShipmentList` API returns a list of shipments associated with the order for a particular node.

If an item has to be delivered or picked up from a local store for which the order status is greater than `Released` and not shipped, the item is most likely to be present in the store. In such situations, the store representative can verify whether the item is present in the store and modify the fulfillment options as requested by the customer by using the `getCompleteOrderDetails` API.

For more information about changing the fulfillment options, see [Section 4.9.19, "Change Fulfillment Options"](#).

---

---

**Note:** You can configure the user interface to never show the pick up fulfillment method if the enterprise does not support that fulfillment method.

---

---

### 5.9.13 Fulfillment Summary

Sometimes, Sterling Call Center and Sterling Store users may have to view the fulfillment summary of an order to get a complete picture of the way the order is being fulfilled. In a B2B scenario, customers may place orders for large quantities of items, and this may result in the items

being available on multiple dates. In such a situation, Sterling Call Center and Sterling Store users may require a better visibility into the way the orders are fulfilled.

For more information about viewing the fulfillment summary, see [Section 4.9.20, "Fulfillment Summary"](#).

### 5.9.14 Delivery and Service Grouping

When a complex order pertaining to numerous products, provided services, and delivery services exists, users can group the order lines into smaller groups to ensure that appointments are available for the various services. For more information about grouping order lines, see [Section 4.9.21, "Delivery and Service Grouping"](#).

### 5.9.15 Change Service Instructions

At the time of delivery, if a customer requests to change the service instructions for a product or service item, you can modify the service instructions as requested. For more information about changing service instructions, see [Section 4.9.22, "Change Service Instructions"](#).

### 5.9.16 Change Service Appointments

When a customer wants to change the date and time of delivery of the ordered items, you can modify the service appointments. For more information about changing service appointments, see [Section 4.9.23, "Change Service Appointment"](#).

### 5.9.17 Reservations

The reservation of items is performed to confirm availability and resolve issues when handling multiple customers simultaneously. For more information about reservations, see [Section 4.9.25, "Reservations"](#).

### 5.9.18 Price Match

A customer may sometimes find that an item is available at a cheaper price from a different retailer. In such situations, the store representative can investigate the customer's claims and perform the necessary price match action. For more information about price matching an item, see [Section 4.9.8, "Price Match"](#).



### 5.9.19 Competitor Search

Sterling Call Center and Sterling Store provides the ability the search and view competitor records. While performing a price match, you can search if the competitor already exists in the database. For more information about searching for a competitor, see [Section 4.9.9, "Competitor Search"](#).

### 5.9.20 Price Match Search

Sterling Call Center and Sterling Store provides the ability the search and view existing price match records. During the price match process, a user may want to search for an existing price match. Based on the search criteria, a list of price matches is displayed. For more information about searching for a price match, see [Section 4.9.10, "Price Match Search"](#).

### 5.9.21 Create or Modify a Competitor Record

Sterling Call Center and Sterling Store provides the functionality to create a new competitor, if the competitor does not already exist in the database, and also allows you to modify an existing competitor record. For more information about creating or modifying a competitor record, see [Section 4.9.11, "Create or Modify a Competitor Record"](#).

### 5.9.22 Create or Modify a Price Match Record

Sterling Call Center and Sterling Store provides the functionality to create a new price match, if the price match does not already exist in the database, and also allows you to modify an existing price match record. For more information about creating or modifying a price match record, see [Section 4.9.12, "Create or Modify a Price Match Record"](#).

### 5.9.23 Increase Order Line Quantity

When a customer wants to increase the quantity of one or more items on an order that is already placed, the user can increase the order line quantity as requested. For more information about increasing an order line quantity, see [Section 4.9.26, "Increase Order Line Quantity"](#).

In the Sterling Store application, while increasing the order line quantity, if the order line is in the Released status, a warning message which

prompts the user to manually check whether the items are present in the store is displayed.

### 5.9.24 Schedule and Release an Order

Sterling Call Center and Sterling Store enables a user to schedule and release an order. For more information about scheduling and releasing an order, see [Section 4.9.27, "Schedule and Release an Order"](#).

### 5.9.25 View Procurement Orders

Sterling Call Center and Sterling Store enables a user to view procurement orders (Purchase Orders and Transfer Orders) that are created for a sales order. For more information about viewing procurement orders, see [Section 4.9.28, "View Procurement Orders"](#).

### 5.9.26 Change Order Address

A customer may want to change or modify the addresses on an order after placing the order. In such situations, you can modify the addresses as requested by the customer. For more information about changing the order address, see [Section 4.9.29, "Change Order Address"](#).

### 5.9.27 Launch Application Console

In some implementations, call center and store representatives may need to access screens and functionality from the Application Console. For example, call center and store representatives may need to access the detailed order audit screens available in the Application Console.

For more information about launching the Application Console, see [Section 4.9.30, "Launch Application Console"](#).

### 5.9.28 Change an Item's Style

After placing an order for a stylized item, a customer may want to change the attributes of the item. Sterling Call Center and Sterling Store enables you to change an item's style once the item is added to the order. For more information about changing an item's style, see [Section 4.9.31, "Change an Item's Style"](#).

### 5.9.29 Reconfigure Bundle Items

Sterling Call Center and Sterling Store enable users to reconfigure bundle items. For example, if the customer has ordered a computer and wants to change the configuration of the processor, this task enables users to change the necessary configurations. For more information about reconfiguring bundle items, see [Section 4.9.32, "Reconfigure Bundle Items"](#).

### 5.9.30 Change Tax Options

A business customer may want to request tax exemptions on an order after placing the order. For example, after placing an order, if a business customer decides to give away the items ordered to a charity organization, Sterling Call Center and Sterling Store enables the store representative to allow tax exemptions on the order. For more information about changing tax options, see [Section 4.9.33, "Change Tax Options"](#).

### 5.9.31 Copy Order

When creating an order or viewing an order that was placed earlier, a customer may decide to use that order to create new orders in the future, where applicable. For example, a customer may place an order for a particular list of items every month, with the list varying only slightly from month to month. Sterling Call Center and Sterling Store provides the Copy Draft Order related task, which enables a user to copy an existing draft order, and the Order Again related task to create a new draft order from a confirmed order, based on the permissions provided to the user. For more information about copying orders, see [Section 4.9.34, "Copy Order"](#).

### 5.9.32 E-Mail Order Information

At times, a customer may request a store representative to send information about an order by e-mail. For example, a customer may want to review the items that have been included in the order. Sterling Call Center and Sterling Store provides the Send Order E-mail related task to enable users to send information pertaining to an order to the e-mail address of the customer. For more information about e-mailing order information, see [Section 4.9.35, "E-Mail Order Information"](#).

### 5.9.33 Transfer Draft Order to Web

Draft orders can be created either by a customer service representative through the Sterling Call Center and Sterling Store applications, or by a customer through the Sterling Web application. Draft orders created in Sterling Web are accessible through Sterling Call Center and Sterling Store. However, draft orders that are opened through Sterling Call Center and Sterling Store are not accessible through Sterling Web.

Sterling Call Center and Sterling Store enable users to transfer the draft orders, which have been created or opened in the Sterling Call Center and Sterling Store applications, to the Sterling Web application. For more information about transferring draft orders to Web, see [Section 4.9.36, "Transfer Draft Order to Sterling Web"](#).

## 5.10 Payment Processing

In the Sterling Store scenario, each order has to be paid when it is placed. The customer can use different types of payment to pay for an order. For more information about payment processing and implementation, see [Section 4.10, "Payment Processing"](#).

## 5.11 Payment Inquiry

Sterling Call Center and Sterling Store enables users to view payment information pertaining to an order, such as order total, refunds, and pending charges. For more information about payment inquiry, see [Section 4.11, "Payment Inquiry"](#).

## 5.12 Return Order Capture

Sterling Store implements the same Return Order functionality that is in Sterling Call Center, as described in [Section 4.13, "Return Order Capture"](#). This section contains the few exceptions that apply to Sterling Store for creating returns, entering return reasons, and viewing or changing the Fulfillment Summary and Payment Summary.

Maintaining returns is described in [Section 5.14, "Return Order Maintenance"](#). This includes tasks such as reporting extra items, wrong items, and unexpected items, issuing immediate refunds, changing return addresses, and canceling return orders.

### 5.12.1 Create Return

This section describes only the Sterling Store functionality for creating a return. For more information about creating returns, see [Section 4.13, "Return Order Capture"](#).

To scan an item for return, a user selects the By Order tab, which shows an Item ID field. When the user scans the item, the Item ID is displayed in the Item ID field. If the item scanned is not on the order, an error message is displayed indicating that the Item ID was not found.

A scanned item is automatically marked in hand when it is scanned. Returns without a receipt and returns by Extra Item (items that the customer received by mistake) are also assumed to be in hand by the system. Certain aspects of the return process, such as return methods and appointments, are unnecessary if an item is in hand.

If an item is not scanned, the return panel contains a check box for "Item is in Hand" and a check box for "All Selected Items Are Present."

When scanning items, the last scanned item is always visible on the screen.

Sterling Store provides the capability to record items that were in hand when the return was created.

### 5.12.2 Return Reason

This section describes only the Sterling Store functionality for the Return Reason screen. For more information about entering the Return Reason screen, see [Section 4.15.2, "Entering Return Reason"](#).

For an item that was marked in hand on the Line Selection screen, the Return Reason screen contains a drop-down for the Return Disposition Code, which enables the user to specify the condition of the item when it was returned. If the user does not have any items in hand, he will not be able to enter the Disposition Code. This code is required only for an item in hand to continue the return process flow. The value for this field is stored in YFS\_ORDER\_LINE, as described in [Table 6–2, "YFS\\_ORDER\\_LINE Extensions"](#).

As a part of the reference implementation, the following Return Disposition Codes are configured for the Create Store Return task:

- Sent For Repair

- Scrapped
- Restocked

To apply the same Disposition Code to all items in hand that are being returned, click the Apply to All Items button on the Return Reason Screen. For more information about returns and exchanges, see [Section 4.13, "Return Order Capture"](#).

### 5.12.2.1 All Items Are In Hand Rule

After the Address Info Needed rule described in [Section 4.15.4, "Exchange Fulfillment Options"](#) is run, the "All Items Are in Hand" rule will be run. This rule checks whether all the lines on the order are marked as in hand or if there is an exchange order. If either is true, the flow continues to the Fulfillment Summary screen.

If no exchange order is created and every item in the return order is in hand, the flow goes to the Payment screen because no lines are being returned by shipping or being picked up, and no reservations or appointments are needed.

If no exchange order is created on the Return Reason screen or if there is a return order with any line that is not in hand, the returns flow proceeds to the Fulfillment Summary screen.

If there is no exchange order, the flow proceeds to the Fulfillment Summary screen.

## 5.12.3 Fulfillment Summary

This section describes only the Sterling Store functionality for the Fulfillment Summary screen. For more information about entering the Fulfillment Summary screen, see [Section 4.15.5, "Fulfillment Summary"](#).

When this screen is displayed, a call to the `getReturnMethod` API is made, which determines whether the line needs to be displayed in a pickup panel, a shipping panel, or whether the customer can keep the item.

The same process collects lines that are already received or are in hand, and displays them in their own panel, so that the user can see their status. This status cannot be changed.

### 5.12.4 Payment Processing

This section describes only the Sterling Store functionality for the Payment Summary screen. For more information about entering the Payment Summary screen, see [Section 4.15.7, "Payment Summary"](#).

Instead of calling the `confirmOrder` API to process payments, Sterling Store calls a user exit called `processReturnCompletionUE`. This user exit calls a list of APIs based on the return or exchange lines. The default implementation of these APIs tries to push the order through the pipeline as far possible, receiving the items that are in hand and setting the items that are to be picked up or shipped to `Released` status. When items that are not in hand are received at the warehouse, a receipt is started so that the customer receives a refund.

## 5.13 Return Order Inquiry

Sometimes customers may inquire information about return orders. For example, a customer may want to change the return address or appointment on the return order. In such situations, call center or store representatives need to search for the appropriate return order and then perform the necessary actions. For more information about return order inquiry, see [Section 4.14, "Return Order Inquiry"](#).

In Sterling Store, the `SellerOrganizationCode` attribute is passed to the `getOrderList` API.

## 5.14 Return Order Maintenance

Sterling Call Center and Sterling Store provides the following return order maintenance features:

- [Report Extra Items](#)
- [Report Wrong Items](#)
- [Report Unexpected Items](#)
- [Issue Refund Now](#)
- [Change Return Address](#)
- [Cancel Return Order](#)

### 5.14.1 Report Extra Items

Sometimes a customer may report that the customer has received extra items. For example, if a customer ordered for a TV and has received 2 TV's instead of 1. In such situations, the extra items can either be returned back or you can allow the customer to keep the extra items. For more information about reporting extra items, see [Section 4.15.9, "Report Extra Items"](#).

### 5.14.2 Report Wrong Items

Sometimes a customer may report that the customer did not receive the items that were ordered. In such situations you may need to reship the items that were ordered and also create a return for the wrong items that were delivered. For example, if a customer ordered for TV and has received a DVD player instead of the TV, you need to create a return for the DVD player and reship the TV. For more information about reporting wrong items, see [Section 4.15.10, "Report Wrong Items"](#).

### 5.14.3 Report Unexpected Items

Sometimes a customer may report that he did not order for any items and has received some items. In such situations you may need to create a customer and then create a return for the wrongly shipped items. For more information about reporting unexpected items, see [Section 4.15.11, "Report Unexpected Items"](#).

### 5.14.4 Issue Refund Now

A customer may call to inquire about why he has not been refunded for the items which have already been returned. In such situations, you may need to issue a refund for the items immediately. For more information about issuing a refund, see [Section 4.15.12, "Issue Refund Now"](#).

### 5.14.5 Change Return Address

A customer may want to change or modify the addresses on a return order. In such situations, you can modify the addresses as requested. For more information about changing the return address, see [Section 4.15.13, "Change Return Address"](#).



### 5.14.6 Cancel Return Order

A customer may want to cancel some of the items or the entire return order. The customer may also want to cancel the exchange order corresponding to the return order. For more information about canceling a return order, see [Section 4.15.16, "Cancel Return Order"](#).

## 5.15 Customer Creation

Sterling Call Center and Sterling Store enables you to create two types of customers:

- Consumer Customers
- Business Customers

---

---

**Note:** The customer creation feature launches a single wizard that displays different screens depending on the customer type.

---

---

### 5.15.1 Create Consumer Customer

A Consumer Customer consists of a single contact. The only information required for a consumer customer is a consumer contact record, which serves as a unique identifier for the customer. For more information about creating a consumer customer, see [Section 4.16.1, "Create Consumer Customer"](#).

### 5.15.2 Create Business Customer

A Business Customer consists of a Buyer Organization and any number of contacts. The addresses and payment methods can be defined for all contacts, or for specific contacts, depending on where they are created. The unique identifier of a Business Customer is the Customer ID. For more information about creating a business customer, see [Section 4.16.2, "Create Business Customer"](#).

## 5.16 Customer Details

Sterling Call Center and Sterling Store enables you to view customer details for both business and consumer customers. For more information about viewing customer details, see [Section 4.17, "Customer Details"](#).

## 5.17 Customer Inquiry

Sterling Call Center and Sterling Store provides the ability to search and view two types of customers:

- Consumer Customers
- Business Customers

For more information about customer inquiry, see [Section 4.18, "Customer Inquiry"](#).

## 5.18 Customer Maintenance

Sterling Call Center and Sterling Store provides the ability to manage a customer's contacts, addresses, and payment methods.

### 5.18.1 Manage Contacts

Sterling Call Center and Sterling Store enables you to manage the contacts of a consumer or business customer. You can add contacts, modify contact records, modify contact addresses, manage contact payment methods, and delete contacts that are no longer active.

For a consumer customer, Sterling Call Center and Sterling Store prompts you to modify the single contact that is associated to the consumer. A consumer customer can have only one contact record associated to that customer. For a business customer, multiple contact records can be available for modification, creation, or deletion. For more information about managing contacts, see [Section 4.19.1, "Manage Contacts"](#).

### 5.18.2 Manage Business Customer Information

Sterling Call Center and Sterling Store enables you to manage business customer information, such as a customer's business name, Web site, status, shipping preferences, and tax information. In addition, you can manage business customer classifications, such as the customer's relationship type and customer level. For more information about managing business customer information, see [Section 4.19.2, "Manage Business Customer Information"](#).

### 5.18.3 Manage Customer Addresses

Sterling Call Center and Sterling Store enables you to define multiple addresses for a customer. You can define addresses at the contact level or at the customer level. You can also define default Ship To, Bill-To and Sold To addresses to be used during ordering. If a contact has no defaults selected, then the customer default addresses will be used. For more information about managing customer addresses, see [Section 4.19.3, "Manage Customer Addresses"](#).

### 5.18.4 Manage Payment Methods

Sterling Call Center and Sterling Store enables you to define multiple payment methods for a customer. You can define payment methods at the contact level or at the customer level. You can also define a default payment method at the contact level or customer level. For more information about managing payment methods, see [Section 4.19.4, "Manage Payment Methods"](#).

### 5.18.5 Reset User Passwords

Sterling Call Center and Sterling Store enables you to reset a user's password, if a user ID is associated to the user's record. For more information about resetting a user's password, see [Section 4.19.5, "Reset User Passwords"](#).

### 5.18.6 Customer Team Assignments

Sterling Call Center and Sterling Store provides the ability to assign customers to a team once the customer is created. A supervisor can assign a customer from his team, to any of his immediate sub teams. For more information about assigning a customer to a team, see [Section 4.19.6, "Customer Team Assignments"](#).

### 5.18.7 Customer User Assignments

Sterling Call Center and Sterling Store provides the ability to assign customers to a user, once the user is created and assigned to a team. A supervisor can search for users from his team and assign the customer to the selected user. For more information about assigning a customer to a user, see [Section 4.19.7, "Customer User Assignments"](#).

### 5.18.8 Customer Self Assignments

A user with permissions can assign himself to customers that is available to his/her team. Users can assign customers to themselves up to a maximum limit and also remove customers that they are no longer working with. For more information about assigning a customer, see [Section 4.19.8, "Customer Self Assignments"](#).

### 5.18.9 Customer Notes

A user may have to enter notes about a customer, such as when the customer can be contacted if queries are raised about an order. For more information about customer notes, see [Section 4.19.9, "Customer Notes"](#).

## 5.19 User Creation

Sterling Call Center and Sterling Store enables you to create and modify users. For example, a store representative may have to create a new store representative user.

---

---

**Note:** A user is an employee of the enterprise organization who can log in to the Sterling applications.

---

---

For more information about creating and managing users, see [Section 4.20, "User Creation"](#).

## 5.20 User Team Creation

Sterling Call Center and Sterling Store enables you to create and manage user teams. Teams are created to assign store representatives to a set of customers. This assignment enables customers to maintain a relationship

with specific store representatives so that customers receive better customer service.

**Note:** A user team is a collection of users who have common data and user interface access requirements. User teams can have access to specific document types, enterprises, ship nodes, customers, and screens within the user interfaces.

For more information about creating and managing user teams, see [Section 4.21, "User Team Creation"](#).

## 5.21 Pagination

When users perform inquiries, they need the information to be retrieved in a minimum amount of time and to be displayed as efficiently as possible. For more information about pagination, see [Section 4.22, "Pagination"](#).

## 5.22 Alert and Queue Management

The alert management task enables the user to create, assign, and resolve alerts. The new alerts that are raised can be notified to the user based on certain configurations, such as poll time. Sterling Call Center and Sterling Store provides a comprehensive visibility to the alerts assigned to users, alerts moved to a different queue, and so on.

If your enterprise has configured queues common to various stores, you may want store users to look at alerts intended only for his store.

### 5.22.1 Solution

Sterling Store provides a way to filter alerts based on the user's current store based on a configuration. If the rule is enabled the store user will be able to see alerts assigned only to his store. If this rule is enabled, the ShipnodeKey is passed to the APIs when retrieving alerts in the following screens:

- Alert Search
- View My Alerts and Queues

- Alert Notification Panel

For more information about alert and queue management, see [Section 4.23, "Alert and Queue Management"](#).

### 5.22.2 End-User Impact

None.

### 5.22.3 Implementation

Sterling Store allows you to configure user alert notification. For more information about configuring user alert notification, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.

### 5.22.4 Reference Implementation

None.

## 5.23 Announcements

Sterling Call Center and Sterling Store enables users to post messages or other useful information pertaining to users. Users who belong to a user group that has the necessary permission can create, view, and cancel announcements. Announcements created in Sterling Store by a store user for a particular store will have the ship node of the store stamped on the announcement. However, announcements created in Sterling Call Center will not have the ship node of the store stamped on the announcement, and therefore, will not be displayed in Sterling Store. For more information about announcements, see [Section 4.24, "Announcements"](#).

## 5.24 User Preferences

Sterling Call Center and Sterling Store enables you to set user preferences such as the display of alert notifications, additional help messages, keyboard shortcuts, customer message panel, position of related tasks, CSR message/customer message panel, and scratch pad. For more information about user preferences, see [Section 4.25, "User Preferences"](#).

---

**Note:** If the `-DhideStoreCustomerMessage` value in the `som.ini` file is set to `true`, the Customer Message panel is not displayed irrespective of whether the Hide customer message panel check box in the User Preferences is selected or not. For more information about the `som.ini` file, refer to the *Sterling Call Center and Sterling Store: Installation Guide*.

---

## 5.25 Viewing the Documentation Library

Sterling Call Center and Sterling Store enables users to view the product documentation library to get additional information about the Sterling Call Center and Sterling Store application. For more information about the product documentation, see [Section 4.26, "Viewing the Documentation Library"](#).

## 5.26 Deploying Sterling Store Clients in Phases

Sterling Call Center and Sterling Store enables enterprises to deploy the application in a phased manner. The functionality available to the users is based on the version of the application deployed on the clients. For more information about deploying the Sterling Store application in phases, see [Section 4.27, "Deploying the Sterling Call Center Clients in Phases"](#).

## 5.27 Helpful Text Messages

When using the Sterling Call Center and Sterling Store applications, a novice user may want to view additional help messages which guide the user to perform a task. For more information about helpful text messages, see [Section 4.28, "Helpful Text Messages"](#).

## 5.28 Order Print Tasks

A customer may visit a store to either place an order or make some modifications to an existing order that may result in some payment changes. In such situations, you must issue an order slip of the transaction to the customer.

Printers are configured as devices. For more information about defining devices, refer to the *Sterling Warehouse Management System: Configuration Guide*.

### 5.28.1 Create Order and Order Summary Prints

When you confirm a draft order, you must issue the customer a Order Sales Slip that has a snapshot of the order. Typically, the snapshot displays order lines and payment details.

The Order Sales Slip is also printed from the Order Summary screen when the customer needs the summary of the order.

#### 5.28.1.1 Solution

On confirming the draft order, the system invokes the print service component. Whenever a customer requests for an order slip, you can click the Print Order Summary related task to print the Order Sales Slip.

#### 5.28.1.2 End-User Impact

None.

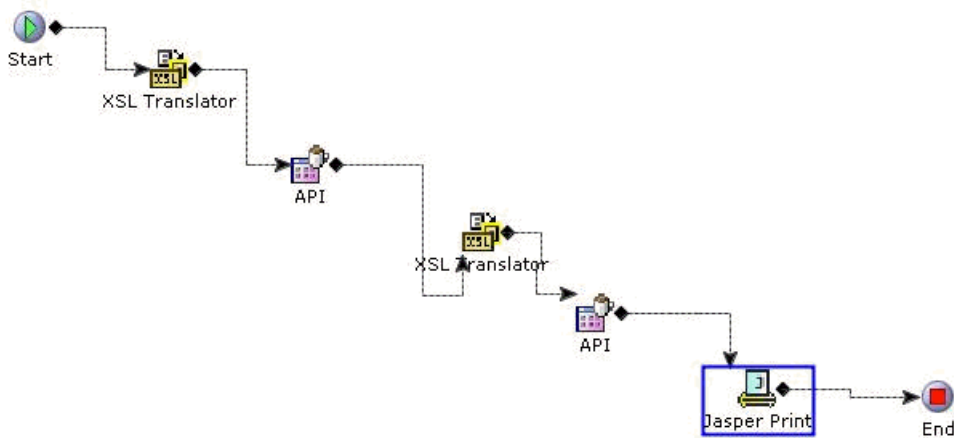
#### 5.28.1.3 Implementation

The following service can be found in the COM Print transaction of the Applications Manager:

YCD\_PrintOrderSummary\_3.0 service—This service is used to print the Sales Order Slip. It is invoked by the callPrintOrderSummaryService event of the COM transaction. [Figure 5–1](#) illustrates this service.



*Figure 5–1 YCD\_PrintOrderSummary\_3.0 service*



#### 5.28.1.4 Reference Implementation

None.

## 5.28.2 Change Order Print

When you make modifications such as order cancellation, adding a coupon, increase in the order line quantity, changes in the fulfillment options, and so forth to an existing order, the Change Order Slip is generated.

### 5.28.2.1 Solution

After performing the order modifications, the print service component is invoked.

### 5.28.2.2 End-User Impact

None.

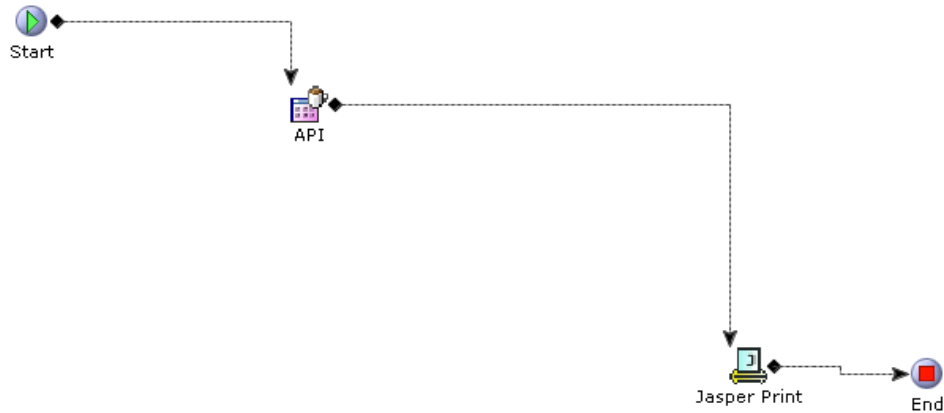
### 5.28.2.3 Implementation

The following service can be found in the COM Print transaction of the Applications Manager:

YCD\_PrintChangeOrderSummaryPrintReport\_3.0 service—This service is used to print the Change Order Slip. It is invoked by the

callPrintChangeOrderService event of the Sterling Call Center and Sterling Store transaction. [Figure 5–2](#) illustrates this service.

**Figure 5–2** *YCD\_PrintChangeOrderSummaryPrintReport\_3.0*



#### 5.28.2.4 Reference Implementation

None.

### 5.28.3 Return or Exchange Order Prints

When a customer returns items to a store, a Return Order Slip must be printed. If the customer has requested for exchange of items, the Exchange Order Slip provides information about the exchange order.

#### 5.28.3.1 Solution

When a customer returns items to a store or exchanges any item, the print service component is invoked.

#### 5.28.3.2 End-User Impact

None.

### 5.28.3.3 Implementation

The following services can be found in the Sterling Call Center and Sterling Store Print transaction of the Applications Manager:

YCD\_PrintReturnOrderSummary\_3.0 service—This service is used to print the Return Order Slip. It is called by the callPrintReturnOrderSummaryService event of the COM transaction. [Figure 5–1](#) illustrates this service.

YCD\_PrintExchangeOrderSummary\_3.0 service—This service is used to print the Exchange Order Slip. It is called by the callPrintExchangeOrderSummaryService event of the COM transaction. [Figure 5–1](#) illustrates this service.

### 5.28.3.4 Reference Implementation

None.

## 5.29 User Security

User Security enables you to ensure that users access only the required information for carrying out their tasks. This requires you to set up user groups and associate users to the user groups. The Channel Applications Manager enables you to create users and user groups and assign permissions.

### 5.29.1 Solution

A user has the privilege to change the password or the user group of another user.

#### Changing a Password

A user can change a password in the Change Password screen. The modifyUserHierarchy API is invoked to change the password. [Table 5–1](#) describes the various validations that the system performs before changing the password.

For more information about the modifyUserHierarchy API, refer to the *Selling and Fulfillment Foundation: Javadocs*.

**Table 5–1** *Change Password Validations*

Validation	Result
If the old password and the new password match	An error message is displayed.
If the new password and the confirmed password do not match	An error message is displayed.
If the old password is incorrect	An error message is displayed.
If the old password field is blank	An error message is displayed.

### Managing Users

A store administrator can either change the credentials of a user or transfer a user to another store. To get a list of all users, the `getUserList` API is invoked. To get the list of all user groups, the `getUserGroupList` API is called. The `modifyUserHierarchy` API is invoked, to apply the changes made to user credentials.

To transfer a store user from one store to another, the `getOrganizationHierarchy` API is called which validates the store to which the user was transferred. However, the store administrators cannot choose to transfer themselves to another store.

## 5.29.2 End-User Impact

None.

## 5.29.3 Implementation

None.

## 5.29.4 Reference Implementation

None.

## 5.30 In-Store Pick-up Tasks

Many retailers want to provide customers the option of ordering items online and picking them up at the store sometimes even on the same day. Sterling Store provides functionality to achieve this with the ability to search for shipments, print pick tickets, perform backroom pick and customer pick. It also enables users to undo a backroom pick.

### 5.30.1 Shipment Inquiry

A store associate may need to search for shipments for various tasks such as printing a pick ticket, recording a backroom pick, customer pick or undo a backroom pick.

#### 5.30.1.1 Solution

Sterling Store provides a user interface task which enables users to search for shipments which need to be picked up by the customer from the current store. This section explains the Shipment Search process and the solution offered by Sterling Store.

#### Searching for Shipments

The store associate can search for shipments based on certain search criteria. The `getShipmentList` API is used to retrieve the list of all the shipments that match the search criteria.

Along with the list of shipments that match the search criteria, the store associate can view other details related to the shipments.

#### Shipment Details

The store associate can view shipment details, such as items in the shipment, quantity of items, and unit price of the item. The `getSortedShipmentDetails` API is called to retrieve the shipment details. If activities are specified for a container, the [View Container Activities](#) hyperlink is displayed in the Shipment Details screen. On clicking this hyperlink, the Container Activities pop-up window is displayed. A user can view information such as the activity code, timestamp, additional information pertaining to the activity, and the user who created the activity.

### Handling Bundle Items

If the Display Bundle Components on Order Summary Screen rule is enabled, the bundle component items are displayed on the Shipment Details screen else the bundle parent item is displayed.

#### 5.30.1.2 End-User Impact

None.

#### 5.30.1.3 Implementation

This section explains the configurations for the shipment inquiry task.

- Sterling Store allows you to configure the Display Bundle Components on Order Summary Screen rule to display bundle component items on the Shipment Details screen. For more information about configuring the display of bundle components, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- This task is permission controlled. The Sterling Store allows you to assign permissions to user groups for this task.

#### 5.30.1.4 Reference Implementation

As part of reference implementation, Sterling Store provides permissions to search shipments to all user groups.

### 5.30.2 Print Pick Ticket

Employees at the store need to know which items to move to the customer pickup area before the customer arrives at the store. Pick ticket is a sheet containing the list of items that need to be moved from the backroom pick location to the customer pick location.

#### 5.30.2.1 Solution

Sterling Store provides a user interface task which enables users to print pick ticket. This section explains the Print Pick Ticket process and the solution offered by Sterling Store.

The `getShipmentList` API is called to retrieve the list of shipments with status less than `Shipment Shipped` and for which pick ticket is not printed.

The YCD\_PrintPickTicket\_8.0 service is used to print the pick ticket of the selected shipment on the user interface.

---

---

**Note:** The Shipment Details screen has a related task for printing a pick ticket just in case a pick ticket for a shipment needs to be reprinted.

---

---

### 5.30.2.2 End-User Impact

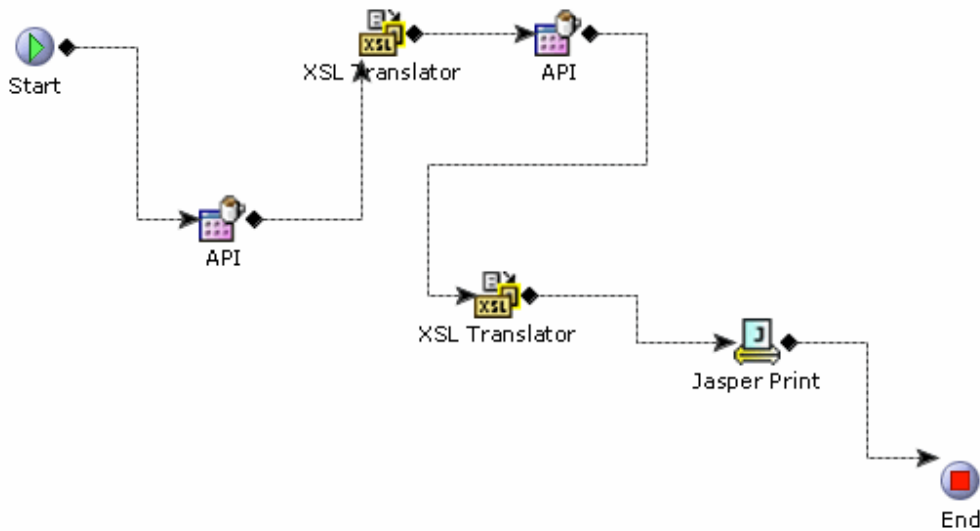
None.

### 5.30.2.3 Implementation

This section explains the configurations for the print pick ticket task.

- You need to configure the printer to print the Pick Ticket. Printers are configured as devices. For more information about defining devices, refer to the *Sterling Warehouse Management System: Configuration Guide*.
- You need to associate the configured printer with PRINT\_PICK\_TICKET print document. For more information about defining print documents, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.
- The YCD\_PrintPickTicket\_8.0 service can be found in the COM Print transaction of the Applications Manager. This service is invoked to print the pick ticket. It is invoked by the callPrintPickTicketService event of the COM Print transaction. [Figure 5–3](#) illustrates this service.

*Figure 5–3 YCD\_PrintPickTicket\_8.0 service*



- This task is permission controlled. Sterling Store allows you to assign permissions to user groups for this task.

#### 5.30.2.4 Reference Implementation

As part of reference implementation, the Sterling Store provides permissions to print pick ticket to all user groups.

### 5.30.3 Backroom Pick

The movement of items from backroom to the customer service area needs to be tracked in a store. Based on the store configuration, Sterling Store supports the recording of backroom picks for a store.

#### 5.30.3.1 Solution

Sterling Store provides a user interface task which enables users to record a backroom pick. This section explains the Backroom Pick process and the solution offered by Sterling Store.



## Recording Backroom Pick

Sterling Store provides the following extended statuses of Shipment Created status:

- Ready For Customer
- Ready For Backroom Pick

Sterling Commerce strongly recommends you to not delete these statuses.

Sterling Store provides the Store Backroom Pick (BaseTransactionKey="Change\_Shipment\_Status") transaction with Pickup Status as Ready For Backroom Pick and the drop status as Ready For Customer.

The Process Shipment for Backroom Pick (BaseTransactionKey="Change\_Shipment\_Status") transaction is introduced to change the shipment status from Shipment Created to either Ready For Backroom Pick status or Ready For Customer status based on the Is Backroom Pick Required rule.

Store associates can scan the items to be picked or enter the Item ID in the screen provided. If the Enable Scanning Of Items Using Barcode rule is selected in Configure Item Search Options of the Channel Applications Manager, the translateBarCode API is called to translate the barcode entered. The Item ID entered is validated to ensure whether the item belongs to the shipment. The quantity being picked is validated against the quantity available to pick.

During the recording of a backroom pick, the changeShipment API is called to save the picked quantity in the BackroomPickedQuantity attribute of the shipment line. When the shipment status needs to be changed from Ready for Backroom Pick to Ready for Customer, the changeShipmentStatus API is called with transaction as Store Backroom Pick. The YCDUpdateLocationInventoryUE user exit is called to update the inventory when moving inventory from the backroom pick location to the customer pick location.

## Shortage Resolution

If there is a shortage due to inadequate inventory, the resolution for the shortage is reported as either "Inventory Shortage" or "Will Pick Later". The [Table 5–2](#) provides the list of shortage resolutions and the corresponding system behavior.

**Table 5–2 Shortage Resolution and Actions**

Shortage Resolution	System Behavior
Inventory Shortage	Items are backordered from the shipment by calling the changeShipment API with the value of the BackOrderRemovedQuantity flag set to “Y”.
Will Pick Later	No Impact on shipment status. Only the recorded quantity is updated as BackroomPickedQuantity on the corresponding shipment line. It is expected that the store associate will pick up the remaining quantity at a later time.

### Handling Bundle Items

On the Backroom Pick screens, all the bundle components are displayed. If there is a shortage for any component of the bundle, an action is taken as follows:

- Inventory Shortage—Appropriate quantity of the bundle component item is backordered.
- Will Pick Later—Shipment is updated with the picked quantity of the bundle components.

If there is shortage for some of the bundle components in such a way that the configured bundle component ratio is broken

- If the bundle item is configured as ship independent:
  - If the shortage resolution is Inventory Shortage, the shortage quantity of the bundle component item is backordered and the picked quantity is updated as BackroomPickRequired on the corresponding shipment line.
  - If the shortage resolution is Will Pick Later, the picked quantity is updated as BackroomPickRequired on the corresponding shipment line.
- If the bundle item is configured as ship together:
  - If the shortage resolution is Inventory Shortage, the entire bundle item is backordered.
  - If the shortage resolution is Will Pick Later, the picked quantity of the components is updated as BackroomPickRequired on the corresponding shipment lines.

### 5.30.3.2 End-User Impact

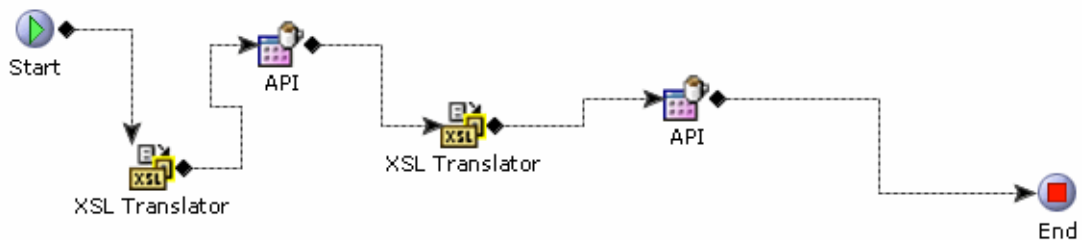
None.

### 5.30.3.3 Implementation

This section explains the various implementation steps required for recording backroom pick.

- Sterling Store allows you to configure the Is Backroom Pick Required rule. For more information about configuring the Is Backroom Pick Required rule, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Process Modeling—Invoke the Change Shipment Status to Backroom Pick action on the ON\_SUCCESS event of Create Shipment transaction. This action in turn invokes the YCD\_Change\_Status\_Backroom\_Pick\_8.0 service. This service calls the getRuleDetails API to retrieve rule value of Is Backroom Pick Required rule. If the rule value is set to “Y”, the Process Shipment for Backroom Pick (BaseTransactionKey="Change\_Shipment\_Status") transaction changes the status of the shipment to Ready For Backroom Pick. If the rule value is set to “N”, the transaction changes the status of the shipment to Ready For Customer. [Figure 5–4](#) illustrates this service.

**Figure 5–4** YCD\_Change\_Status\_Backroom\_Pick\_8.0 service



- Implement the YCDUpdateLocationInventoryUE user exit to update the inventory stored in an external system after the backroom pick process.

- Notes entered for this task are saved on the order with YCD\_BACKROOM\_PICK\_SHORTAGE note type.
- This task is permission controlled. Sterling Store allows you to assign permissions to user groups for this task.

### 5.30.3.4 Reference Implementation

This section explains the reference implementation provided as part of the backroom pick task.

- The ON\_SUCCESS event of Create Shipment transaction invokes the Change Shipment Status to Backroom Pick action for Matrix Retail.
- The Is Backroom Pick Required rule is set to "Y" for Matrx\_Store\_1 store and "N" for Matrx\_Store 2 store and Matrx\_Store\_2 store.
- Sterling Store provides permissions to record a backroom pick to all user groups.

## 5.30.4 Undo Backroom Pick

After performing the backroom pick, the customer may decide not to pick up the order. The store associate needs to undo the backroom pick process so that the items can be moved back.

### 5.30.4.1 Solution

Sterling Store provides a user interface task which enables users to undo backroom pick. This section explains the Undo Backroom Pick process and the solution offered by Sterling Store.

The Undo Store Backroom Pick (BaseTransactionKey="Change\_Shipment\_Status") transaction is introduced between the Ready For Customer and Ready For Backroom Pick statuses.

The undoBackRoomPick API is called to perform undo backroom pick. This API changes the status of the shipment from Ready for Customer to Ready for Backroom Pick.

### Handling Bundle Items

Bundle parent and component items are displayed on the Undo Backroom Pick screen but the table rows corresponding to the parent items are disabled.

#### 5.30.4.2 End-User Impact

None.

#### 5.30.4.3 Implementation

This section explains the configurations for the undo backroom pick task.

- Notes entered for this task are saved on the order with YCD\_UNDO\_PICK note type.
- This task is permission controlled. Sterling Store allows you to assign permissions to user groups for this task.

#### 5.30.4.4 Reference Implementation

As part of reference implementation, Sterling Store provides permissions to undo backroom pick to all user groups.

### 5.30.5 Customer Pick

When a customer walks into a store to pick up the ordered items, Sterling Store enables you to record customer pick.

#### 5.30.5.1 Solution

Sterling Store provides a user interface task which enables users to record a customer pick. This section explains the Customer Pick process and the solution offered by Sterling Store.

#### Searching for a Shipment

The Advanced Shipment Search or Customer Pick task can be used to search the shipment for which the customer pick needs to be performed. The getShipmentList API is called to search the shipment based on the search criteria.

### Verification of the Customer

The `getSortedShipmentDetails` API is called to retrieve the details that must be displayed on the various screens pertaining to the Customer Pick flow.

The store associate must first validate whether the person who walked into the store is actually the intended recipient of the item.

Credit card verification type is built into the application. You can configure your own verification types such as license verification, address verification. For more information about configuring verification types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*. In case of credit card verification, the last 4 digits of the credit card number are validated against the last 4 digits of credit card number of the order for which customer pick is being performed.

All the verification types are stored as common codes for the enterprise with the code type as "YCD\_CUST\_VERFN\_TYP". The `getCommonCodeList` API is called to retrieve the verification types. The verification types displayed on the Customer Verification screen are sorted in the alphabetical order.

Whenever a customer is verified during the customer pick process, a note is logged by calling the `changeOrder` API.

### Identifying Items for Customer Pick

If the `Enable Scanning Of Items Using Barcode` rule is selected in `Configure Item Search Options` of the `Channel Applications Manager`, store associates can scan the items to be picked or enter the Item ID in the field provided. The `translateBarCode` API is called to translate the barcode that is entered. The Item ID entered is validated to ensure whether the item belongs to the shipment. The quantity being picked is validated against the quantity available to pick.

### Resolving Shortages

If there are shortages of items, the shortage resolution screen displays the shortage resolutions. The [Table 5–3](#) provides the list of shortage resolutions and the corresponding system behavior.

**Table 5–3 Shortage Resolution and Actions**

Shortage Resolution	System Behavior
Inventory Shortage	Items are backordered from the shipment by calling the changeShipment API with the value of the BackOrderRemovedQuantity flag set to "Y".
Cancel Remaining	Items are cancelled from the shipment by calling the changeShipment API with the value of the CancelRemovedQuantity flag set to "Y".
Customer Preference	No Impact on shipment status. Only the recorded quantity is updated on the corresponding shipment line and a new shipment is created by calling the splitShipment API. It is assumed that the customer will pickup the remaining quantity later.

### Recording Customer Pick

At the end of the Customer Pick process, following API's are called:

- recordCustomerPick
- raiseEvent

The recordCustomerPick API does the following:

- Calls the changeOrder API to save the note for customer verification.
- Calls various API's to take appropriate actions for shortage resolution as mentioned in [Table 5–3](#).
- Calls YCDUpdateLocationInventoryUE user exit to update the location inventory.

The raiseEvent API invokes handler for the callPrintCustomerPickAknwlgmntService event of the COM Print transaction.

### Handling Bundle Items

On the Customer Pick screens, only the bundle parent item is displayed. If there is a shortage for any bundle parent quantity, an action is taken as follows:

- Inventory Shortage—shortage quantity of the bundle component item is backordered.

- Will Pick Later—Shipment is split and new shipment is created for the shortage quantity of the bundle item.
- Cancel—Shortage quantity is removed from the shipment and order.

### 5.30.5.2 End-User Impact

None.

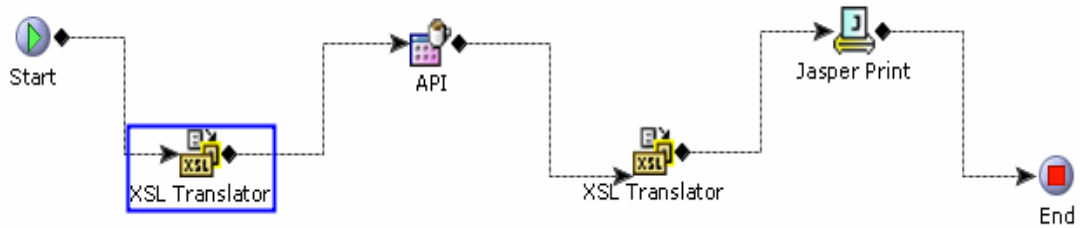
### 5.30.5.3 Implementation

This section explains the configurations for the customer pick task.

- Sterling Store allows you to configure the verification types to be used for customer verification step during Customer Pick. For more information about configuring verification types, see the *Sterling Selling and Fulfillment Suite: Applications Configuration Guide*.
- Notes for customer verification are saved on the order with YCD\_CUSTOMER\_VERIFICATION note type.
- Process Modeling—Invoke the Print Customer Pick Acknowledgement action on the callPrintCustomerPickAcknwlgmntService event of the COM Print transaction. This action in turn invokes the YCD\_PrintCustomerPickAcknowledgement\_8.0 service. This service calls the getSortedShipmentDetails API to retrieve the details of the picked items. This service also calls the getPrinter API to find a printer to print the acknowledgement, and the getCurrencyList API to display the currency details on the acknowledgement. [Figure 5–5](#) illustrates this service.



*Figure 5–5 YCD\_PrintCustomerPickAcknowledgement\_8.0 service*



- This task is permission controlled. Sterling Store allows you to assign permissions to user groups for this task.

#### 5.30.5.4 Reference Implementation

This section explains the reference implementation for the customer pick task.

- Sterling Store provides the following verification types for Matrix Incorporated, Matrix Retail, Matrix Business, and Heavy-Lift:
  - Address verification
  - License verification
  - Passport verification
  - Other Verification
- Sterling Store provides permissions to record customer pick to all user groups.

## 5.31 Integrating with Business Center

In some implementations, call center and store representatives may have to access screens and functionality from the Business Center application. For example, call center and store representatives may have to configure pricing rules in Business Center.

### 5.31.1 Solution

Sterling Call Center and Sterling Store provides a user interface task that enables users to launch Business Center from within Sterling Call Center and Sterling Store. When the task is selected, the Business Center application is launched in an embedded Web browser. The location of the application server running the Business Center must be defined in the `locations.ycfg` file. For more information about the `locations.ycfg` file, refer to the *Sterling Call Center and Sterling Store: Installation Guide*.

Sterling Call Center and Sterling Store provides a single sign-on capability to Business Center, enabling the user to access both the Item Administration and Pricing Administration applications. To log in, the user must belong to the following:

- A user group that has permission to launch Business Center from within Sterling Call Center and Sterling Store. This user interface task is permission controlled.
- A user group that has permission to the Business Center application and its screens.
- A menu group that can access both the Sterling Call Center and Sterling Store and Sterling Business Center.

For information about using Business Center, refer to the *Sterling Business Center Pricing Administration Guide* and the *Sterling Business Center Item Administration Guide*.

### 5.31.2 End-User Impact

Users will not be able to access Business Center unless they belong to the menu group and user groups described in [Section 5.31.1, "Solution"](#).

### 5.31.3 Implementation

This section explains the configurations for the Business Center task.

- Business Center can be deployed on an application server that is different from the Sterling Call Center and Sterling Store server. To enable the launch of a Web browser for Business Center, the location of the Business Center application server must be defined in the `locations.ycfg` file. For more information about the

`locations.ycfg` file, refer to the *Sterling Call Center and Sterling Store: Installation Guide*.

- This task is permission controlled. Sterling Call Center and Sterling Store allows you to assign the necessary permissions to the corresponding user groups for performing this task.

### 5.31.4 Reference Implementation

None.



## Summary of Components

---

This chapter explains the various components that are included in Sterling Call Center and Sterling Store. It is assumed that the user is familiar with the Selling and Fulfillment Foundation and understands which components need to be added or modified.

This chapter explains the following components:

- [Database Extensions for Sterling Call Center and Sterling Store](#)
- [APIs and User Exits](#)
- [Services](#)
- [Monitor Events](#)
- [XSL Files](#)
- [Hold Types](#)
- [Transactions](#)
- [Time-Triggered Purge Transactions](#)
- [Events](#)
- [Agents](#)

### 6.1 Database Extensions for Sterling Call Center and Sterling Store

To implement Sterling Call Center and Sterling Store, the following tables have been extended:

- [YFS\\_ORDER\\_HEADER](#)
- [YFS\\_ORDER\\_LINE](#)

- YFS\_SHIPMENT
- YFS\_PRICE\_MATCH
- YFS\_PRICE\_MATCH\_H
- YFS\_COMPETITOR
- YFS\_ORDER\_LINE\_PRICE\_MATCH
- YFS\_ASSC\_RELATIONSHIP
- YFS\_EVENT\_AUDITOR

## YFS\_ORDER\_HEADER

*Table 6–1 YFS\_ORDER\_HEADER Extensions*

Column Name	Data Type	Description
ALL_ADDRESSES_VERIFIED	Char (1)	This flag determines whether or not the addresses in this order have been verified by an address verification program.
COMPL_GIFT_BOX_QTY	Number (3)	The number of complimentary gift boxes to be given with the order.
NO_OF_AUTH_STRIKES	Number (1)	Indicates the number of failed credit card authorizations with HARD_DECLINE as reason code for the order.
SOURCE_IP_ADDRESS	Varchar2 (20)	The IP address of the system from which the customer placed order.
CUSTOMER_FIRST_NAME	Varchar2 (40)	The first name of the customer who placed the order.
CUSTOMER_LAST_NAME	Varchar2 (40)	The last name of the customer who placed the order.

*Table 6–1 YFS\_ORDER\_HEADER Extensions*

Column Name	Data Type	Description
CUSTOMER_PHONE_NO	Char (40)	The daytime phone number of the customer who placed the order.
CUSTOMER_ZIP_CODE	Varchar2 (35)	The address postal code of the customer who placed the order.

**YFS\_ORDER\_LINE***Table 6–2 YFS\_ORDER\_LINE Extensions*

Column Name	Data Type	Description
BACKORDER_NOTIFICATION_QTY	Number (14,4)	Indicates the backordered quantity for which the last customer notification was sent.
IS_PRICE_MATCHED	Char (1)	This flag determines whether the price match was performed for the item.
ITEM_IS_IN_HAND	Char (1)	Specifies whether the item was in hand when it was returned.
IS_PICK_UP_NOW	Char (1)	This flag determines whether price match was performed for the item.
DISPOSITION_CODE	Char (40)	The disposition code for the return line.

**YFS\_SHIPMENT***Table 6–3 YFS\_SHIPMENT Extensions*

Column Name	Data Type	Description
NOTIFICATION_SENT	Char (1)	This flag determines whether or not the customer has been informed about this shipment.
PICKTICKET_PRINTED	Char (1)	This flag indicates whether or not the pick ticket has been printed.
BACKROOM_PICK_REQ UIRED	Char (1)	This flag indicates whether or not backroom pick is required.

**YFS\_SHIPMENT\_LINE***Table 6–4 YFS\_SHIPMENT\_LINE Extensions*

Column Name	Data Type	Description
BACKROOM_PICKED_QTY	Number (14,4)	Indicates the backroom picked quantity for the shipment line.

**YFS\_PRICE\_MATCH***Table 6–5 YFS\_PRICE\_MATCH Extensions*

Column Name	Data Type	Description
PRICE_MATCH_KEY	Char (24)	The primary key for the YFS_PRICE_MATCH table.
STATUS	Varchar2 (100)	Indicates the status of the price match record.
PRICE_MATCH_DESC RIPTION	Varchar2 (2000)	The description of the price match.
ENTERPRISE_CODE	Char (24)	Enterprise for which the price match is created.
COMPETITOR_KEY	Char (24)	Indicates the foreign key to the YFS_COMPETITOR table.



**Table 6–5 YFS\_PRICE\_MATCH Extensions**

Column Name	Data Type	Description
COMPETITOR_UNIT_PRICE	Number (15,2)	Indicates the competitor's unit price.
COMPETITOR_DISCOUNTS	Number (15,2)	Indicates the competitor's discount.

Sterling Call Center and Sterling Store supports data security for the price match entity based on the ENTERPRISE\_CODE column.

## YFS\_PRICE\_MATCH\_H

**Table 6–6 YFS\_PRICE\_MATCH\_H Extensions**

Column Name	Data Type	Description
PRICE_MATCH_KEY	Char (24)	The primary key for the YFS_PRICE_MATCH table.
STATUS	Varchar2 (100)	Indicates the status of the price match record.
PRICE_MATCH_DESCRIPTION	Varchar2 (2000)	The description of the price match.
ENTERPRISE_CODE	Char (24)	Enterprise for which the price match is created.
COMPETITOR_KEY	Char (24)	Indicates the foreign key to the YFS_COMPETITOR table.
COMPETITOR_UNIT_PRICE	Number (15,2)	Indicates the competitor's unit price.
COMPETITOR_DISCOUNTS	Number (15,2)	Indicates the competitor's discount.

Sterling Call Center and Sterling Store supports data security for the price match history entity based on the ENTERPRISE\_CODE column.

**YFS\_COMPETITOR***Table 6–7 YFS\_COMPETITOR Extensions*

Column Name	Data Type	Description
COMPETITOR_KEY	Char (24)	The primary key for the YFS_COMPETITOR table.
COMPETITOR_NAME	Varchar (40)	The competitor's name who offered a lower price of the item.
COMPETITOR_DESC	Varchar2 (2000)	The description of the competitor.
ENTERPRISE_CODE	Char (24)	Enterprise the competitor belongs to.
WEB_ADDRESS	Varchar2 (255)	Indicates the web address of the competitor.
STATUS	Char (40)	Indicates the status of the competitor.
PERCENT_PRICE_MATCH	Number (5,2)	Indicates the price match percentage.
CREATETS	sysdate	Indicates the timestamp when the record was created.
MODIFYTS	sysdate	Indicates the timestamp when the record was last modified.
CREATEUSERID	Varchar2 (40)	Indicates the user who created the record.
MODIFYUSERID	Varchar2 (40)	Indicates the user who modified the record.
CREATEPROGID	Varchar2 (40)	Indicates the program that created the record.
MODIFYPROGID	Varchar2 (40)	Indicates the program that modified the record.
LOCKID	Number (5)	Indicates the integer used to track and prevent concurrent modifications.

Sterling Call Center and Sterling Store supports data security for the competitor entity based on the ENTERPRISE\_CODE column.

**YFS\_ORDER\_LINE\_PRICE\_MATCH****Table 6–8 YFS\_ORDER\_LINE\_PRICE\_MATCH Extensions**

Column Name	Data Type	Description
ORDER_LINE_PRICE_MATCH_KEY	Char (24)	The primary key for the YFS_ORDER_LINE_PRICE_MATCH table.
ORDER_LINE_KEY	Char (24)	The primary key of the YFS_ORDER_LINE table which relates the price match to an order line.
ORDER_HEADER_KEY	Char (24)	The primary key of the YFS_ORDER_HEADER table which relates the price match to an order.
PRICE_MATCH_KEY	Char (24)	The primary key of the YFS_PRICE_MATCH table which relates the price match to an order line.
EXISTING_UNIT_PRICE	Number (15,2)	This is the unit price of an item used for price match.
EXISTING_DISCOUNTS	Number (15,2)	This is the discounts for an item used for price match.
EXISTING_CHARGES	Number (15,2)	This is the charges for an item used for price match.
COMPETING_CHARGES	Number (15,2)	Indicates the competitor's charges.

**YFS\_ASSC\_RELATIONSHIP****Table 6–9 YFS\_ASSC\_RELATIONSHIP Extensions**

Column Name	Data Type	Description
ASSC_RELATIONSHIP_KEY	Char (24)	The primary key for the YFS_ASSC_RELATIONSHIP table.
ENTERPRISE_CODE	Char (24)	Enterprise for which the association types are configured to be displayed.
ASSOCIATION_TYPE	Varchar2 (20)	The association type to be displayed.

**Table 6–9 YFS\_ASSC\_RELATIONSHIP Extensions**

Column Name	Data Type	Description
RELATIONSHIP_TYPE	Varchar (20)	The relationship type indicates the relationship between the order lines.
SHOW_IN_UI	Char (1)	This attribute indicates whether or not the association type has to be displayed.
CREATETS	sysdate	Indicates the timestamp when the record was created.
MODIFYTS	sysdate	Indicates the timestamp when the record was last modified.
CREATEUSERID	Varchar2 (40)	Indicates the user who created the record.
MODIFYUSERID	Varchar2 (40)	Indicates the user who modified the record.
CREATEPROGID	Varchar2 (40)	Indicates the program that created the record.
MODIFYPROGID	Varchar2 (40)	Indicates the program that modified the record.
LOCKID	Number (5)	Indicates the integer used to track and prevent concurrent modifications.
INVENTORY_CHECK_REQUIRED	Char (1)	This attribute indicates whether or not the inventory check has to be performed.
ASSC_FUNCTION	Varchar2 (40)	This indicates whether the associated item will replace the original item, or will be added as its related item.

## YFS\_EVENT\_AUDITOR

*Table 6–10*

Column Name	Data Type	Description
EVENT_AUDITOR_KEY	Char (24)	The primary key for the YFS_EVENT_AUDITOR table.
ORGANIZATION_CODE	Char (24)	Organization for which the event auditor record is created.
EVENT_AUDITOR_TYPE	Char (24)	Indicates the event auditor type. The available options are: <ul style="list-style-type: none"> <li>• TRANSACTION_EVENT</li> <li>• HOLD_ADDED_EVENT</li> <li>• HOLD_RESOLVED_EVENT</li> <li>• HOLD_REJECTED_EVENT</li> <li>• APPLICATION_EVENT</li> </ul>
TRANID	Varchar2 (40)	Indicates the transaction ID the auditor record corresponds to for a transaction event.
EVENTID	Varchar2 (40)	Indicates the event ID the auditor event corresponds to for a hold event.
HOLD_TYPE	Char (20)	Indicates the hold type the auditor event corresponds to for a transaction event.
NOTE_TYPE	Varchar2 (40)	Indicates the reason code for the note that has been logged.
TRANSFORMATION_FILE	Varchar2 (100)	Indicates the XSL file used to transform the XML data to notes text.
ACTIVE	Char(1)	Indicates whether or not the auditor event is active.

## 6.2 APIs and User Exits

Table 6–11 and Table 6–12 provide the list of APIs and user exits introduced in Sterling Call Center and Sterling Store, and the features that are affected.

**Table 6–11 APIs**

API	Features Affected
getAssociationRelationshipList	Order Creation
verifyAddress	Address Verification
getCompleteOrderDetails	Cancel Order, Change Fulfillment Options, Increase Order Line Quantity, Order Creation, Add Multiple Items to an Order
getCompleteOrderInvoiceDetails	Payment Inquiry
getCompleteItemList	Order Creation, Item Inquiry, Add Line
checkDuplicateOrder	Duplicate Order Validation
checkFraudOnOrder	Fraud Check
getReturnMethod	Return Order Capture, Change Return Method
getOrderLinesWithTransactionQuantity	Return Order Capture
applyPriceMatch	Price Match
checkNoHasslePriceMatch	Price Match
checkOrderPriceMatch	Price Match
getPriceMatchCompetitors	Price Match
managePriceMatch	Price Match
getCompetitorList	Price Match
manageCompetitor	Price Match
getCompetitorDetails	Price Match
getPriceMatchList	Price Match
managePriceMatch	Price Match
getPriceMatchDetails	Price Match
getOrderLinesForPriceMatch	Price Match
listCarrierService	Change Fulfillment Options

**Table 6–11 APIs**

API	Features Affected
manageServiceGrouping	Delivery and Service Grouping
getOrderFulfillmentDetails	Fulfillment Summary, Change Service Appointment
reserveOrder	Change Service Appointment
getEventAuditorList	Order Notes
getUserGroupList	Managing Users
getCompleteShipmentDetails	Track an Item
getSortedShipmentDetails	Shipment Inquiry, Customer Pick, Track an Item
undoBackRoomPick	Undo Backroom Pick
recordCustomerPick	Customer Pick

**Table 6–12 User Exits**

User Exit	Features Affected
UpdateLocationInventoryUE	Backroom Pick, Customer Pick
YCDPromotionValidationUE	Add Coupon
YCDVerifyAddressWithAVSUE	Address Verification
YCDProcessReturnCompletionUE	Return Order Capture
YCDGetAppeasementOffersUE	Customer Appeasement
YCDSendFutureOrderCustomerAppeasementUE	Customer Appeasement
YCDProcessDuplicateOrderCheckUE	Duplicate Order Validation
YCDProcessOrderFraudCheckUE	Fraud Check
YCDGetReturnPolicyUE	Return Order Capture, Change Return Method
YCDGetReturnMethodUE	Return Order Capture
YCDCanPriceMatchBePerformedUE	Price Match

**Table 6–12 User Exits**

User Exit	Features Affected
YCDCheckNoHasslePriceMatchUE	<a href="#">Price Match</a>
YCDCheckOrderPriceMatchUE	<a href="#">Price Match</a>
YCDGetPriceMatchCompetitorsUE	<a href="#">Price Match</a>
YCDManagePriceMatchUE	<a href="#">Price Match</a>
YCDgetCompetitorListUE	<a href="#">Price Match</a>
YCDGetCompetitorDetailsUE	<a href="#">Price Match</a>
YCDGetPriceMatchListUE	<a href="#">Price Match</a>
YCDGetPriceMatchDetailsUE	<a href="#">Price Match</a>
YCDGetTrackingNumberURLUE	<a href="#">Track an Item</a>
YCDOverrideDeliveryMethodUE	<a href="#">Order Creation, Change Fulfillment Options</a>
YCDValidateCreditCardInfoUE	<a href="#">Change Payment Method</a>
YCDEmailOrderUE	<a href="#">E-Mail Order Information</a>

For more information about these APIs and user exits, see the *Sterling Call Center and Sterling Store: Javadocs*.

## 6.3 Services

The following services are introduced in Sterling Call Center and Sterling Store, and can be found in the service definitions of the `Sales Order` document type:

- [Order Maintenance Services](#)
- [Order Pricing Services](#)
- [Customer Services](#)
- [Order Capture Services](#)
- [Payment Processing Services](#)
- [Order Printing Services](#)



- [FTC Compliance Services](#)

### 6.3.1 Order Maintenance Services

*Table 6–13 Order Maintenance Services*

Service	Feature Affected
YCD_ResolveAddressHold_1.0	<a href="#">Address Verification</a>
YCD_ResolveOrderAlerts_1.0	<a href="#">Address Verification</a>
YCD_SetOrderDefaultsOnChange_2.0	<a href="#">Federal Trade Commission Compliance, Identifying a Customer</a>
YCD_UpdateBackorderOnCancel_1.0	<a href="#">Federal Trade Commission Compliance</a>
YCD_StopDeliveryRequest_UpdateQty_8.0	<a href="#">Cancel Order</a>
YCD_StopDeliveryRequest_Alert_8.0	<a href="#">Cancel Order</a>
YCD_LogNotesForOrderHistory_8.0	<a href="#">Order Notes</a>
YCD_Change_Status_Backroom_Pick_8.0	<a href="#">In-Store Pick-up Tasks</a>
YCD_Populate_Order_Email_8.5	<a href="#">E-Mail Order Information</a>
YCD_Send_Order_Email_8.5	<a href="#">E-Mail Order Information</a>
YCD_Trigger_Customer_Limit_Adjustment_8.5	<a href="#">Credit Limit Exceeded</a>

### 6.3.2 Order Pricing Services

The following customer notification services are introduced in the Sterling Multi-Channel Selling Solution Integration implementation.

**Table 6–14 Order Pricing Services**

Service	Feature Affected
YCD_Coupon_Validation_UE_8.0	This feature enables a user to add coupons or promotion codes for an order that has been placed. The Sterling Multi-Channel Selling Solution integration provides an implementation of this feature.

### 6.3.3 Customer Services

The following customer services are introduced in Sterling Call Center and Sterling Store.

**Table 6–15 Customer Services**

Service	Feature Affected
YCD_CancelOrderNotification_1.0	<a href="#">Federal Trade Commission Compliance</a>
YCD_ShipmentNotificationEmail_1.0	<a href="#">Shipment Notification</a>
YCD_Update_Customer_Assignment_8.0	Team and User Assignment

### 6.3.4 Order Capture Services

The following order capture services are introduced in Sterling Call Center and Sterling Store.

**Table 6–16 Order Capture Services**

Service	Feature Affected
YCD_DuplicateOrderAlert_1.0	<a href="#">Duplicate Order Validation</a>
YCD_ProcessCustOrdMgmtHolds_1.0	<a href="#">Duplicate Order Validation</a>
YCD_ProcessDuplicateOrderCheck_1.0	<a href="#">Duplicate Order Validation</a>

**Table 6–16 Order Capture Services**

Service	Feature Affected
YCD_SetOrderDefaults_2.0	Federal Trade Commission Compliance, Identifying a Customer
YCD_VerifyAddressFailedAlert_1.0	Address Verification

### 6.3.5 Payment Processing Services

The following payment processing services are introduced in Sterling Call Center and Sterling Store.

**Table 6–17 Payment Processing Services**

Service	Feature Affected
YCD_ExecuteCollectionCreditCard_Proxy_1.0	Payment Processing
YCD_ExecuteCollectionRefundCheck_Proxy_1.0	Payment Processing
YCD_ExecuteCollectionSVC_Proxy_1.0	Payment Processing
YCD_PaymentAuthRetryLimitAlert_1.0	Payment Processing
YCD_PaymentDeclinedAlert_1.0	Payment Processing
YCD_PaymentHardDeclinedAlert_1.0	Payment Processing
YCD_PaymentServiceUnavailableAlert_1	Payment Processing
YCD_ProcessCollectionFailure_1	Payment Processing

### 6.3.6 Order Printing Services

The following order printing services are introduced in Sterling Call Center and Sterling Store.

**Table 6–18 Order Printing Services**

Service	Feature Affected
YCD_PrintOrderSummary_3.0	Create Order and Order Summary Prints
YCD_PrintReturnOrderSummary_3.0	Return or Exchange Order Prints

**Table 6–18 Order Printing Services**

Service	Feature Affected
YCD_PrintExchangeOrderSummary_3.0	Return or Exchange Order Prints
YCD_PrintChangeOrderSummaryPrintReport_3.0	Change Order Print
YCD_PrintPickTicket_8.0	Print Pick Ticket

### 6.3.7 FTC Compliance Services

The following FTC compliance services are introduced in Sterling Call Center and Sterling Store.

**Table 6–19 FTC Compliance Services**

Service	Feature Affected
YCD_FTC_AUTO_RESOLVE	Federal Trade Commission Compliance
YCD_FTC_CANCEL_ORDER_LINES	Federal Trade Commission Compliance
YCD_FTC_DELAY_ALERT	Federal Trade Commission Compliance
YCD_FTC_DELAY_EMAIL	Federal Trade Commission Compliance
YCD_FTC_FIRST_DELAY_ALERT	Federal Trade Commission Compliance
YCD_FTC_FIRST_DELAY_EMAIL	Federal Trade Commission Compliance
YCD_FTC_INDEFINITE_DELAY_ALERT	Federal Trade Commission Compliance
YCD_FTC_INDEFINITE_DELAY_EMAIL	Federal Trade Commission Compliance
YCD_FTC_SEND_NOTIFICATION	Federal Trade Commission Compliance

## 6.4 Monitor Events

The following monitor events are introduced in Sterling Call Center and Sterling Store.

**Table 6–20 Monitor Events**

Monitor Event	Feature Affected
YCD_FTC_Auto_Resolve	Federal Trade Commission Compliance
YCD_FTC_Cancel_Lines	Federal Trade Commission Compliance
YCD_FTC_Delay_Alert	Federal Trade Commission Compliance
YCD_FTC_Delay_Email	Federal Trade Commission Compliance
YCD_FTC_First_Delay_Alert	Federal Trade Commission Compliance
YCD_FTC_First_Delay_Email	Federal Trade Commission Compliance
YCD_FTC_Indefinite_Delay_Alert	Federal Trade Commission Compliance
YCD_FTC_Indefinite_Delay_Email	Federal Trade Commission Compliance

## 6.5 XSL Files

The following sample XSL translation files have been provided by Sterling Call Center and Sterling Store to support the customization and localization of information.

**Table 6–21 XSL Translators**

XSL File	Feature Affected
<INSTALL_DIR>/repository/xapi/template/merged/exception_console/YCD_FTC_Alert_Template_5.0.xsl.sample	Federal Trade Commission Compliance
<INSTALL_DIR>/repository/xapi/template/merged/exception_console/YCD_FTC_Email_Alert_Template_5.0.xsl.sample	Federal Trade Commission Compliance

**Table 6–21 XSL Translators**

<b>XSL File</b>	<b>Feature Affected</b>
<INSTALL_DIR>/repository/xapi/template/merged/service/ycd/YCD_Log_Note_On_Hold_Status_Change_8.0.xsl.sample	<a href="#">Order Notes</a>
<INSTALL_DIR>/repository/xapi/template/merged/service/ycd/YCD_Log_Note_On_Back_Order_8.0.xsl.sample	<a href="#">Order Notes</a>
<INSTALL_DIR>/repository/xapi/template/merged/service/ycd/YCD_Log_Note_On_Confirm_Shipment_8.0.xsl.sample	<a href="#">Order Notes</a>
<INSTALL_DIR>/repository/xapi/template/merged/service/ycd/YCD_Log_Note_On_Appointment_Completion_8.0.xsl.sample	<a href="#">Order Notes</a>
<INSTALL_DIR>/repository/xapi/template/merged/service/ycd/YCD_Log_Note_On_Appointment_Failure_8.0.xsl.sample	<a href="#">Order Notes</a>
<INSTALL_DIR>/repository/xapi/template/merged/service/ycd/YCD_Log_Note_For_Order_BackOrdered_2.0.xsl.sample	<a href="#">Order Notes</a>
<INSTALL_DIR>/repository/xapi/template/merged/service/ycd/YCD_Log_Note_On_Chained_Order_Created_2.0.xsl.sample	<a href="#">Order Notes</a>
<INSTALL_DIR>/repository/xapi/template/merged/service/ycd/YCD_Get_Backroom_Flag_8.0.xsl.sample	<a href="#">Backroom Pick</a>
<INSTALL_DIR>/repository/xapi/template/merged/service/ycd/YCD_Set_Backroom_Flag_8.0.xsl.sample	<a href="#">Backroom Pick</a>

## 6.6 Hold Types

The following hold types are introduced in Sterling Call Center and Sterling Store.

**Table 6–22** *Hold Types*

Hold Type	Feature Affected
YCD_DUPLICATE_ORDER	Duplicate Order Validation
YCD_VERIFY_ADDRESS	Address Verification
YCD_FRAUD_CHECK	Fraud Check
YCD_CREDIT_LIMIT	Credit Limit Exceeded

## 6.7 Transactions

The following transactions are introduced in Sterling Call Center and Sterling Store.

**Table 6–23** *Transactions*

Transaction Name	Feature Affected	Events
Duplicate Order	Duplicate Order Validation	On Duplicate Order
Address Verification	Address Verification	Address Verification Failed
Shipment Notification	Shipment Notification	Send Notification
COM PRINT	Create Order and Order Summary Prints	Print Order Summary
COM PRINT	Return or Exchange Order Prints	Print Return Order Summary
COM PRINT	Return or Exchange Order Prints	Print Exchange Order Summary
COM PRINT	Change Order Print	Print Change Order
COM PRINT	Print Pick Ticket	Print PickTicket

*Table 6–23 Transactions*

Transaction Name	Feature Affected	Events
COM PRINT	Customer Pick	Customer Pick Acknowledgement
YCD_UNDO_BACKROOM_PICK	Backroom Pick	None
YCD_BACKROOM_PICK	Backroom Pick	None

## 6.8 Time-Triggered Purge Transactions

There are several transactions that you can use to purge your database tables at specific time intervals. Purge transactions determine when a table should be purged by determining the current date and subtracting the retention days specified by the purge. If the timestamp on the table is less than or equal to (current day - retention days) the table is purged.

Sterling Call Center and Sterling Store provides the following time-triggered purge transactions:

- [Price Match Purge](#)
- [Price Match History Purge](#)

### 6.8.1 Price Match Purge

This purge removes price match data from the system. If the expiry date of a price match record is less than or equal to (current day - retention days), the agent moves the price match records to the price match history table. Sterling Call Center and Sterling Store provides the YCDPRICEMATCHPRG purge agent which purges the price match records from the YFS\_PRICE\_MATCH table. If a price match record is still present in the YFS\_ORDER\_LINE\_PRICE\_MATCH table, the price match record cannot be purged. In such a situation, the price match record will be purged only after the record is purged from the YFS\_ORDER\_LINE\_PRICE\_MATCH table by running the Order Purge time-triggered transaction. For more information about Order Purge, refer to the *Sterling Distributed Order Management: Configuration Guide*.



## Attributes

The following are the attributes for this time-triggered transaction:

**Table 6–24 Price Match Purge Attributes**

Attribute	Value
Base Transaction ID	YCDPRICEMATCHPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs called	None
User Exits Called	YFSBeforePurgeUE

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table 6–25 Price Match Purge Criteria Parameters**

Criteria Parameters	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Purge Code	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that are moved to history tables without actually moving them.</li> </ul>
Enterprise Code	Optional. The organization for which the Alert Purge needs to be run. If not passed, then all enterprises are monitored.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.

*Table 6–25 Price Match Purge Criteria Parameters*

Criteria Parameters	Description
CollectPendingJobs	If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
ColonyID	Required in a multi schema deployment when the YFS_PRICE_MATCH table may exist in multiple schemas. Runs the agent for the colony.

### Statistics Tracked

The following statistics are tracked for this transaction:

*Table 6–26 Price Match Purge Statistics*

Statistic Name	Description
NumPriceMatchPurged	Number of price match records purged.

## 6.8.2 Price Match History Purge

This purge removes price match history data from the system. Sterling Call Center and Sterling Store provides the YCDPRICEMATCHHISTPRG purge agent which purges the price match history records from the YFS\_PRICE\_MATCH\_H table.

### Attributes

The following are the attributes for this time-triggered transaction:

*Table 6–27 Price Match History Purge Attributes*

Attribute	Value
Base Transaction ID	YCDPRICEMATCHHISTPRG
Base Document Type	General
Base Process Type	General
Abstract Transaction	No
APIs called	None
User Exits Called	YFSBeforePurgeUE

## Criteria Parameters

The following are the criteria parameters for this transaction:

**Table 6–28 Price Match History Purge Criteria Parameters**

Criteria Parameters	Description
Action	Required. Triggers the transaction. If left blank, it defaults to Get, the only valid value.
Purge Code	Required. Cannot be modified. Used for internal calculations, such as determining retention days. Corresponds with the PurgeCode used in Business Rules Purge Criteria.
Live	Optional. Mode in which to run. Valid values are: <ul style="list-style-type: none"> <li>Y - Default value. Moves qualifying records from the regular tables listed under Tables Purged to the corresponding history tables.</li> <li>N - Test mode. Determines the rows that are moved to history tables without actually moving them.</li> </ul>
Enterprise Code	Optional. The organization for which the Alert Purge needs to be run. If not passed, then all enterprises are monitored.
Number of Records To Buffer	Optional. Number of records to retrieve and process at one time. If left blank or specified as 0 (zero), it defaults to 5000.
CollectPendingJobs	If this parameter is set to N, the agent does not collect information on the pending jobs for this monitor. This pending job information is used for monitoring the monitor in the System Management Console.
ColonyID	Required in a multi schema deployment when the YFS_PRICE_MATCH_H table may exist in multiple schemas. Runs the agent for the colony.

## Statistics Tracked

The following statistics are tracked for this transaction:

**Table 6–29 Price Match History Purge Statistics**

Statistic Name	Description
NumPriceMatchHistory Purged	Number of price match history records purged.

## 6.9 Events

The following transaction events are implemented in Sterling Call Center and Sterling Store.

**Table 6–30 Events**

Transaction	Event
Change Order	On Cancel
Change Order	On Success
Change Order Release	On Backorder
Change Order Release	On Cancel
Confirm Draft Order	On Success
Create Order	On Success
Release Order	On Cancel
Release Order	On Backorder
Schedule Order	On Backorder
Return Receipt	On Success
Create Shipment	On Success

### 6.9.1 Event Call Back Handlers

The following pre-event call back handlers are implemented in Sterling Call Center and Sterling Store.

**Table 6–31 Pre-event Call Back Handlers**

Transaction	Event	Feature Affected
Confirm Shipment	On Success	<a href="#">Order Notes</a>
Schedule Order	On Backorder	<a href="#">Order Notes</a>

**Table 6–31 Pre-event Call Back Handlers**

Transaction	Event	Feature Affected
Release Order	On Backorder	<a href="#">Order Notes</a>
Confirm Work Order	On Work Order Completion	<a href="#">Order Notes</a>
Confirm Work Order	On Work Order Execution Failure	<a href="#">Order Notes</a>

For more information about pre-event call back handlers, refer to the *Selling and Fulfillment Foundation: Application Platform Configuration Guide*.

## 6.10 Agents

The following agents are implemented in Sterling Call Center and Sterling Store.

**Table 6–32 Agents**

Agent	Feature Affected
YCDPRICEMATCHPRG	<a href="#">Purging Of Price Match Records</a>
YCDPRICEMATCHHISTPRG	<a href="#">Purging Of Price Match History Records</a>



## Performance Tuning

---

This chapter explains the tasks you must perform to fine-tune Sterling Call Center and Sterling Store for optimal performance.

### 7.1 Performance Recommendations

This section, specifically, [Table 7–1](#), lists the performance recommendations for the Sterling Call Center and Sterling Store applications.

**Table 7–1 Performance Recommendations**

Feature	Recommendation
Managing the database	Create indexes for the UPPER (LAST_NAME), UPPER (FIRST_NAME), and DAY_PHONE columns of the YFS_PERSON_INFO table.
Searching for the Manufacturer's Name in the Quick Access screen.	Create indexes for the MANUFACTURER_NAME column of the YFS_ITEM table. Also create indexes for all the item attributes for which searches are enabled.
Defaulting the City and State fields based on the postal code, on the address panel	Create indexes for the YFS_REGION_DETAIL column of the FROM_ZIP and TO_ZIP tables.
Retrieving orders that contain work orders	Create indexes for the CURRENT_WORK_ORDER_KEY column of the YFS_ORDER_LINE table.

**Table 7–1 Performance Recommendations**

Feature	Recommendation
Searching for customers	Create indexes for the PARENT_CUSTOMER_KEY, ORGANIZATION_CODE, and BUYER_ORGANIZATION_CODE columns of the YFS_CUSTOMER table.
Managing customer payment methods	Create indexes for the CUSTOMER_CONTACT_KEY column of the YFS_CUSTOMER_PAYMENT_METHOD table.
Assigning a customer to selected users	Create indexes for the TEAM_ID and ORGANIZATION_CODE columns of the YFS_CUSTOMER_ASSIGNMENT_LOOKUP table.
Searching for alerts	Create indexes for the ENTERPRISE_KEY, TEAM_CODE, BILL_TO_ID, EXCEPTION_TYPE_SHADOW, ACTIVE_FLAG, SHIPMENT_KEY columns of the YFS_INBOX table.

For more information about custom indexes, refer to the *Selling and Fulfillment Foundation: Performance Management Guide*.

## 7.2 Enabling Reference Data Caching

The caching feature is enabled using the `yfs.dbcache.classes` parameter in the `yfs.properties` file. Sterling Commerce recommends that you add `com.yantra.yfs.dbclasses.YFS_Assc_RelationshipDBCACHEHome` and `com.yantra.yfs.dbclasses.YFS_Event_Auditor_DBCACHEHome` to the `yfs.dbcache.classes` parameter value. For more information about enabling reference data caching, refer to the *Selling and Fulfillment Foundation: Performance Management Guide*.

## 7.3 Cached Data

Sterling Call Center and Sterling Store caches the results of APIs that are called repeatedly in an application. The cached data is stored (in the client's memory) by the Enterprise, if it is Enterprise-level data.

Sterling Call Center and Sterling Store caches the following data:



- All rules
- The results of the `getCurrencyList` API call, which populates the Currency and Default Currency drop-down lists with defined currencies
- The results of the `getLocalList` API call, which populates the Locale drop-down list with defined locales
- The results of the `getPaymentTypeList` API call, which populates the Payment Type drop-down list with defined payment types
- Common code types

The values for the following common code types are cached:

- **RELATIONSHIP\_TYPE**  
The `getCommonCodeList` API is called to populate the Relationship Type drop-down list with defined values.
- **CUSTOMER\_LEVEL**  
The `getCommonCodeList` API is called to populate the Customer Level drop-down list with defined values.
- **VERTICAL**  
The `getCommonCodeList` API is called to populate the Vertical drop-down list with defined values.
- **CUSTOMER\_STATUS**  
The `getCommonCodeList` API is called to populate the Status drop-down list with defined values.
- **COUNTRY**  
The `getCommonCodeList` API is called to populate the Country drop-down list with defined values. When a country is selected from the Country drop-down list, the `getRegionList` API is called to populate the State drop-down list with defined values.

When the APIs are initially called, Sterling Call Center and Sterling Store stores the results of the API calls in cache memory. When the data for the API calls is requested again, Sterling Call Center and Sterling Store applications retrieve the information from cache memory instead of making repeated API calls. The cached data is persisted until the user logs out or closes the application.

For information about enabling reference data caching, see [Section 7.2, "Enabling Reference Data Caching"](#).

# Localizing the Application

---

This chapter explains how to localize Sterling Call Center and Sterling Store.

## A.1 User Interface Themes

The user interface theme files specify the screen colors, display fonts, and images to use. The display fonts are dependent on the languages that need to be supported. However, some fonts may not support all languages. When setting up a theme, choose a font that displays the specific language you need or choose a font that displays for a particular locale. For example, when setting up a Japanese locale, customize the theme to use a font that displays Japanese characters such as Hiragana.

To localize Sterling Call Center and Sterling Store user interface, save `com.yantra.pca.ycd_sapphire.ythm`, `com.yantra.pca.ycd_earth.ythm`, `com.yantra.pca.ycd_ice.ythm`, and `com.yantra.pca.ycd_jade.ythm` files that are located in the `<INSTALL_DIR>/rcp/COM/reference` folder to `RCP_EXTN_FOLDER`.

For more information about localizing a theme, refer to the *Selling and Fulfillment Foundation: Localization Guide*.

## A.2 Literals

Sterling Call Center and Sterling Store uses resource bundles that contain literals or text displayed on the screens. Sterling Call Center and Sterling Store enables you to customize and localize resource bundles based on the user's locale. In addition, literals used in the customized screens have their own resource bundles, and are considered during the localization process.

For more information about literals, refer to the *Selling and Fulfillment Foundation: Localization Guide*.

If you have customized Sterling Call Center and Sterling Store applications, you must localize those customizations also. For more information about customizing the Rich Client Platform Interface, refer to the *Selling and Fulfillment Foundation: Customization Guide*.

**Note:** Literals cannot be localized in the following places:

- Condition Builder
- Order/Shipment Monitor
- Hard-coded literals in APIs

## A.2.1 Resource Bundles

Sterling Call Center and Sterling Store always releases complete resource bundles in the `com.yantra.pca.ycd_bundle.properties` file with the localized versions of Sterling Call Center and Sterling Store. Incremental updates are not provided. If you localize Sterling Call Center and Sterling Store, it is your responsibility (or that of your third-party localization company) to compare and validate the differences between the resource bundles shipped with the product to those you have localized.

**Note:** The client resource bundle files have some entries having '&' in the value. These values are used for mnemonic key support in the application. When localizing, ensure that all the literals on any screen have a unique mnemonic key.

The Selling and Fulfillment Foundation Sterling Call Center and Sterling Store application uses the resource bundle files to define literals for the user interface.

To localize the client application resource bundles:

1. Save the:

- a. `com.yantra.pca.ycd_bundle.properties` bundles as `com.yantra.pca.ycd_bundle_[language]_[country]_[variant].properties` located in the `<INSTALL_DIR>/resources` folder.

For example, in the case of French customers, `com.yantra.pca.ycd_bundle.properties` should be saved as `com.yantra.pca.ycd_bundle_fr_FR.properties`, where *fr* indicates the language, that is French, *FR* indicates the country, France, and assuming that France does not have any variants.

- b. `com.yantra.pca.ycd_Message_bundle.properties` as `com.yantra.pca.ycd_Message_bundle_[language]_[country]_[variant].properties` located in the `<INSTALL_DIR>/resources` folder.
- c. `com.yantra.pca.ycd_format_bundle.properties` as `com.yantra.pca.ycd_format_bundle_[language]_[country]_[variant].properties` located in the `<INSTALL_DIR>/resources` folder.
- d. `com.yantra.pca.ycd.rcp_1.0.0_plugin.properties` as `com.yantra.pca.ycd.rcp_1.0.0_plugin_[language]_[country]_[variant].properties` located in the `<INSTALL_DIR>/resources` folder.

To localize the titles that appear on the Help Assistant Screen:

1. Save the:

- a. `com.yantra.pca.ycd_HelpMessages_bundle.properties` as `com.yantra.pca.ycd_HelpMessages_bundle_[language]_[country]_[variant].properties` located in the `<INSTALL_DIR>/resources`.

Here, language is the language to which you want to localize, country is the Country Code, and variant is the Time Zone or the Variant Component.

---

**Note:** After saving the bundle files as mentioned above, you must create Sterling Call Center and Sterling Store client applications. For more information about creating a client application, refer to the *Sterling Call Center and Sterling Store: Installation Guide*.

---

For more information about resource bundles, refer to the *Selling and Fulfillment Foundation: Localization Guide*.

To localize the resource bundles server application:

1. Copy the:
  - a. `<INSTALL_DIR>/resources/ycdbundle.properties` file and save it as `<INSTALL_DIR>/resources/ycdbundle_<language>_<country>.properties`.
2. Re-create the Selling and Fulfillment Foundation EAR package.

### A.3 Localizing Address and Store Node Information

When the Address and Store Node information are provided, a bundle key is also provided for all single line displays.

- For a Store Node:

The standard display is formatted as per the `Common_Binding_Store_Node` key in `com.yantra.pca.ycd_bundles.properties`. For example, a bundle record looks like:

```
Common_Binding_Store_Node={8}, {9} ({1})
```

The `Common_Binding_Store_Node` key is passed to the following method in `YRCPlatformUI`:

`getFormattedString (String key, String[] data)` along with the `String[]` array whose contents should match the place holders defined in the `Common_Binding_Store_Node` key.

- For Address Records:

The standard display is formatted as per the `Common_Binding_Address` key in `com.yantra.pca.ycd_bundles.properties`. For example, a bundle record looks like:

```
Common_Binding_Address={0} - {6}, {7}
```

The `Common_Binding_Address` key is passed to the following method in `YRCPlatformUI`:

`getFormattedString` (String key, String[] data) along with the String[] array whose contents should match the place holders defined in the `Common_Binding_Address` key.

## A.4 Factory Setup Data

Besides storing your transactional data, the database also stores configuration data, such as error codes and item descriptions of various attributes. This means that the database may need to store values in a language-specific format. If these database literals are not localized, screen literals display inconsistently, with some displaying in the localized language and others displaying in English.

The following files are available in the localization kit:

- `ycdriliterals2translate.properties`—This file contains all the Sterling Call Center and Sterling Store RI literals that need to be localized.
- `ycdfcaddinliterals2translate.properties`—This file contains all the Sterling Call Center and Sterling Store factory setup Addin literals that need to be localized.
- `ycdfcpcaliterals2translate.properties`—This file contains all the Sterling Call Center and Sterling Store factory setup literals except the Addin literals that need to be localized.

To localize these files:

1. Localize and merge the `ycdfcpcaliterals2translate.properties` file and the `ycdfcaddinliterals2translate.properties` file.
2. Rename the merged file as  
`en_US_ycdfclocalizedstrings_<language>_<country>.properties`.
3. Rename the `ycdriliterals2translate.properties` file to  
`en_US_ycdrilocalizedstrings_<language>_<country>.properties`.

For example, in the case of French customers, `ycdriliterals2translate.properties` should be renamed to `en_US_ycdrilocalizedstrings_fr_FR.properties`, where *fr* indicates the language, that is French, and *FR* indicates the country, France.

4. Copy the following files to the  
    <INSTALL\_DIR>\installed\_data\smcfs\components\complete\_installation\factorysetup\XMLS\<language>\_<country> folder.
  - en\_US\_ycdrilocalizedstrings\_<language>\_<country>.properties
  - en\_US\_ycdfclocalizedstrings\_<language>\_<country>.properties

For more information about loading the language pack translations, refer to the *Sterling Selling and Fulfillment Suite: Applications Installation Guide*.



## A

---

- Add Items screen
  - viewing price breakdown, 90
  - viewing unit price details, 90
- add multiple items to order
  - validating new items, 175
- additional order information
  - field capture, 102
- address information
  - field capture, 99
- address verification, 109
  - address verification agent, 112
  - end-user impact, 115
  - implementation, 115
  - introduction, 109
  - solution, 110
- address verification agent, 112
- Address Verified Flag field, 99
- AllAddressesVerified flag, 111
- Alternate Item Identifier, 64
- American Express credit card, 170
- APPROVED response code, 271

## B

---

- bank hold payment failure, 271
- BANK\_HOLD response code, 271
- bill to ID, 108

## C

---

- Carrier Service Level field, 102

- Carte Blanche credit card, 170
- Changing Delivery Lines, 221
- Changing Fulfillment Options, 218
- Changing Order Address, 245
- Changing Pickup Lines, 221
- changing service appointments, 234
- Channel field, 102
- check
  - account number, 276
  - amount, 276
  - payment method, 265
  - reference, 276
- check account number, 265
- check amount, 265
- check reference, 266
- COLLECTION\_FAILED event, 276, 277
- credit card, 163
  - implementation, 269
  - payment method, 263
- credit card validation
  - end-user impact, 168
  - implementation, 168
  - introduction, 163
- customer ID. See bill to ID
- customer information
  - field capture, 99

## D

---

- data
  - overview, 469
- declined payment failure, 274
- DECLINED response code, 274

- Diners Club credit card, 170
- Disabling Availability Checks, 220
- Disabling Fulfillment Methods, 220
- Disabling Pick up from Store for all Items, 220
- Discover credit card, 170
- draft orders
  - entering names, 94
- duplicate order validation, 104
  - bill to ID, 108
  - end-user impact, 108
  - enterprise code, 108
  - implementation, 109
  - introduction, 104
  - IP address, 108
  - order date, 108
  - solution, 105
  - total amount, 108

## E

---

- e-mail notification, 150
- enterprise code, 108
- Enterprise Code field, 102
- environment variable
  - YANTRA\_HOME, xxxiv
- executeCollection API, 264, 267, 269, 272, 274

## F

---

- field capture
  - additional order information, 102
  - address information, 99
  - customer information, 99
  - fulfillment information, 102
  - item and line information, 101
  - pricing and payment information, 101
- fraud check, 116
  - end-user impact, 119
  - implementation, 120
  - introduction, 116
  - solution, 116
- FTC Compliance
  - promised delivery dates, 131
  - solution, 131
- FTC compliance, 130

- fulfillment information
  - field capture, 102
- Fulfillment Option field, 102
- Fulfillment Summary, 224
  - viewing, 225

## H

---

- hard decline payment failure, 271, 279
- HARD\_DECLINE response code, 271, 279
- holds
  - resolving, 156

## I

---

- Increasing Order Line Quantity, 241
- IP address, 108
- IP Address field, 99
- IsAddressVerified flag, 111
- item information
  - field capture, 101
- Item Promised Date field, 101

## J

---

- JCB credit card, 170

## L

---

- line information
  - field capture, 101
- List of Promotions Applied to the Order field, 101
- Luhn's algorithm, 166

## M

---

- Master Card, 170

## N

---

- NOTIFICATION\_SENT flag, 147

## O

---

- ON\_CANCEL event, 150
- order cancellation notification
  - end-user impact, 152
  - implementation, 153
  - introduction, 150
  - solution, 150
- order creation, 73
- order date, 108
- order delay, 103
  - end-user impact, 103
  - implementation, 103
  - introduction, 103
  - solution, 103
- order inquiry, 124
- order validation, 104
  - address verification, 109
  - duplicate order validation, 104
  - fraud check, 116
  - verifying items, 120
- Orders
  - e-mailing details to customer, 255
- orders
  - adding coupons, 203
  - adding related items, 88
  - adding upsell items, 88
  - copying, 254, 401
  - e-mailing details to customer, 401

## P

---

- Payment Auth Retry Limit Reached queue, 271
- Payment Declined queue, 272, 274
- Payment Hard Declined queue, 271
- payment information
  - field capture, 101
- payment processing
  - end-user impact, 267
  - implementation, 267
  - introduction, 260
  - solution, 261
- payment processing error handling, 276
- payment service unavailable, 272, 274
- Payment Service Unavailable queue, 272, 274

- Pre-paid payment method, 265
- pricing information
  - field capture, 101
- promised delivery dates, 131

## R

---

- refund check
  - payment method, 266
- register number, 265
- requestCollection API, 264
- Reservations, 239
- return check
  - implementation, 274

## S

---

- SEND\_NOTIFICATION event, 147
- service appointments
  - modifying delivery lines after release, 235
- service unavailable payment failure, 272, 274
- SERVICE\_UNAVAILABLE response code, 272, 274
- shipment alert consolidation, 147
- shipment notification, 147
  - end-user impact, 149
  - implementation, 149
  - introduction, 147
  - solution, 147
- Shipment Notification agent, 148
- soft declined payment failure, 271
- SOFT\_DECLINED response code, 271
- Special Care Flag field, 99
- store tender type, 265
- stored value card
  - implementation, 272
  - payment method, 264
- strike limit configuration, 279
- System-generated notes, 214

## T

---

- total amount, 108
- transaction number, 265

## V

validateCoupon API, 203  
Visa credit card, 170

YFSGetFundsAvailableUE user exit, 264

## Y

YANTRA\_HOME, xxxiv  
YCDEmailOrderUE user exit, 255  
YCDOnCollectionFailure action, 276  
YCDPromotionValidationUE user exit, 203  
YCD\_CancelOrderNotification\_1.0 service, 151  
YCD\_ExecuteCollectionCreditCard\_Proxy\_1.0  
service, 269, 270  
YCD\_ExecuteCollectionCreditCard\_1.0  
service, 269  
YCD\_ExecuteCollectionRefundCheck\_Proxy\_1.0  
service, 274, 275, 276  
YCD\_ExecuteCollectionRefundCheck\_1.0  
service, 274, 275  
YCD\_ExecuteCollectionSVC\_Proxy\_1.0  
service, 272, 273  
YCD\_ExecuteCollectionSVC\_1.0 service, 272,  
273  
YCD\_OrderConfirmationEmail\_1.0 service, 153  
YCD\_PaymentAuthRetryLimitAlert\_1.0  
service, 271  
YCD\_PaymentDeclinedAlert\_1.0 service, 271,  
274  
YCD\_PaymentHardDeclinedAlert\_1.0 service, 271  
YCD\_PaymentServiceUnavailableAlert\_1.0  
service, 272, 274  
YCD\_Populate\_Order\_Email\_8.5 service, 255  
YCD\_ProcessCollectionFailure\_1.0 service, 277  
YCD\_ProcessFailure\_1.0 service, 276  
YCD\_Send\_Order\_Email\_8.5 service, 255  
YCD\_SetOrderDefaultsOnChange\_2.0  
service, 142  
YCD\_SetOrderDefaults\_2.0 service, 142  
YCD\_ShipmentNotificationEmail\_1.0 service, 148,  
149  
YFSCollectionCreditCardUE user exit, 264, 269  
YFSCollectionOthersUE user exit, 274  
YFSCollectionStoredValueCardUE user exit, 264,  
272