

IBM Security Directory Integrator
Version 7.2.0.1

Erste Schritte



IBM Security Directory Integrator
Version 7.2.0.1

Erste Schritte



Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die allgemeinen Informationen in „Bemerkungen“ auf Seite 113 gelesen werden.

Impressum

Diese Ausgabe bezieht sich auf Version 7.2.0.1 des Lizenzprogramms *IBM Security Directory Integrator (5724-K74)* sowie auf alle nachfolgenden Releases und Modifikationsstufen, bis dieser Hinweis in einer Neuauflage geändert wird.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs *IBM Security Director Integrator, Version 7.2.0.1, Getting Started Guide*, IBM Form GI11-9325-03, herausgegeben von International Business Machines Corporation, USA

© Copyright International Business Machines Corporation 2003, 2014

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:
TSC Germany
Kst. 2877
Juni 2014

Inhaltsverzeichnis

Abbildungsverzeichnis	v	Ausführungsfehler entschlüsseln	66
Informationen zu dieser Veröffentlichung	vii	Kapitel 3. Ereignisgesteuerte Integration.	71
Zugriff auf Veröffentlichungen und Terminologie	vii	Zeitplanung für Fertigungslinien	72
Eingabehilfen	ix	Fertigungslinien für Serviceanforderung	74
Technische Schulung	ix	Kapitel 4. Abschottung Ihrer Integrationslösungen	83
Informationen zur Unterstützung	ix	Lesbarkeit, Wiederverwendung und Konfigurierbarkeit	83
Erklärung zu geeigneten Sicherheitsvorkehrungen	x	Protokollierung und Prüfung	85
Kapitel 1. Einführung	1	Konnektivitätsprobleme	86
Vereinfachen und Lösen von Problemen	3	Verfügbarkeit der Fertigungslinie	86
Kernel/Komponentenarchitektur	4	Skalieren und Leistung	88
Datenmodell 'Eintrag-Attribut-Wert'	5	Überwachung	90
Datenflüsse sind Fertigungslinien	7	Der Fertigungsliniendebugger	90
Erste Schritte	8	Anhang. EasyETL-Leitfaden	91
Kapitel 2. IBM Security Directory Integrator - Erste Schritte	11	Projekt erstellen	94
Erste Fertigungslinie erstellen	15	Änderungen erkennen	105
Fertigungslinie ausführen.	29	Bemerkungen	113
Nullverhalten: Umgang mit fehlenden Attributen/Werten	32	Index	117
Debug für Ihre Fertigungslinie ausführen	39		
Daten in einer sequenziellen Quelle suchen.	47		
Modus 'Lookup' verwenden	58		
Vererbung	63		
Regeln für die Suche mit 'Lookup' sind Verknüpfungsbedingungen	65		

Abbildungsverzeichnis

1. Das Datenmodell 'Eintrag-Attribut-Wert'	6	41. Debug für Ihre Fertigungslinie ausführen	39
2. Datenfluss durch eine Fertigungslinie	7	42. Der Data Stepper der Fertigungslinie	40
3. Lernprogramm szenario	8	43. Schrittweises Durchgehen der Fertigungslinien- ausführung	41
4. Konfigurationseditor starten	11	44. Ausführung bis zum Connector 'Write_XM- L_File'	42
5. Arbeitsbereich auswählen	11	45. Erweiterter Debugmodus	43
6. Eingangsanzeige des Konfigurationseditors	12	46. Schaltflächen des Debuggers	44
7. Neues Projekt benennen	13	47. Unterbrechungspunkt festlegen	45
8. Hauptanzeige des Konfigurationseditors	14	48. Unterbrechungspunkt im Script festlegen	46
9. Diagramm eines vereinfachten Szenarios mit nur zwei Datenquellen	15	49. Befehlszeile für JavaScript-Auswertung	47
10. Dialogfenster 'Neue Fertigungslinie'	16	50. Das Flussdiagramm des Szenarios	48
11. Leerer Fertigungslinieneditor	17	51. 'FullName' in die Eingabezuordnung Ihres Connectors im Iteratormodus ziehen	48
12. Neue Komponente hinzufügen	18	52. Zuordnung für 'FullName' bearbeiten	49
13. Komponente auswählen	18	53. Connectorschleife ziehen	50
14. Connector umbenennen und zugehörigen Mo- dus ändern	19	54. Konfiguration der Connectorschleife	50
15. Konfigurationsanzeige für den Dateiconnector	20	55. Connectorschleife - Erweiterte Einstellungen	51
16. Parser unter 'Neues Objekt einfügen' auswäh- len	21	56. Hierarchische Attribute	51
17. Kontextmenüauswahl 'Daten durchsuchen'	22	57. Aus Abschnitt 'Schema' in Attributzuordnung ziehen	52
18. Der Datenbrowser	23	58. Bedingungseditor für IF-Verzweigung	52
19. Schema interaktiv durch Durchsuchen von dy- namischen Daten ermitteln	23	59. Script für Hook 'End of Data' erstellen	53
20. Fertigungslinie, für die der Connector im Itera- tormodus ausgewählt ist	24	60. Komponentenliste im Abschnitt 'Datenfluss' der Fertigungslinie	54
21. Schaltfläche 'Komponente hinzufügen'	25	61. Script einer Bedingung für die IF-Verzweigung erstellen	55
22. Fertigungslinie, für die zwei Connector ausge- wählt sind	26	62. Vollständige Fertigungslinie mit FOR-EACH- Schleife	56
23. Attribute in die Ausgabezuordnung ziehen	27	63. Protokollausgabe mit statistischen Daten zur IF-Verzweigung	57
24. Attributzuordnungsregel umbenennen	27	64. XML-Ausgabe mit dem Attribut 'telephoneNo'	58
25. Attribut 'FullName' der Ausgabezuordnung hinzufügen	28	65. Kopierfunktion der Fertigungslinie	59
26. Zuordnung bearbeiten	28	66. Fertigungslinie in Ihr Projekt kopieren	60
27. Die Schaltfläche 'Ausführen'	29	67. Fertigungslinie 'CreatePhoneDB.assemblyline' ausführen	60
28. Protokollausgabe der Fertigungslinienausfüh- rung	29	68. Protokollausgabe der Fertigungslinie 'Create- PhoneDB'	61
29. Schaltflächenleiste für das Protokollausgabe- fenster	30	69. Connector in 'Ressourcen' ziehen	62
30. Daten durchsuchen, die von einem Ausgabe- connector erstellt wurden	30	70. Neue Ressource in Ihre Fertigungslinie ziehen	62
31. XML-Ergebnis durchsuchen	31	71. Vererbung für die Registerkarte 'Hooks' festle- gen	63
32. Option 'Nullverhalten' für Konfiguration auf Fertigungslinienebene	32	72. Vererbung für eine Zuordnungsregel wieder- herstellen	64
33. Konfigurationsdialog 'Nullverhalten'	33	73. Modus ändern, Attribute erkennen und zuord- nen	64
34. Ergebnis der Einstellungen für Nullverhalten in der XML-Ausgabe	33	74. Eine einfache Verknüpfungsbedingung	66
35. IF-Verzweigungskomponente auswählen	34	75. Fehlernachricht in der Protokollausgabe	66
36. Bedingungen für die IF-Verzweigung bearbei- ten	35	76. Teil des Flussdiagramms für den Suchmodus 'Lookup'	68
37. Einfache Bedingungen zur IF-Verzweigung hinzufügen	36	77. Erste Lernprogrammübung abgeschlossen	69
38. Ihre erste vollständige Fertigungslinie	37	78. IBM Security Directory Integrator Scheduler	73
39. Nullverhalten für die Fertigungslinie zurück- setzen	38	79. Attributzuordnungsanzeige von 'HTTP_Server- _Connector'	74
40. Protokollausgabe mit Ihren Nachrichten und dem Speicherauszug des Eintrags	38	80. Attributelement zur Eingabezuordnung hinzu- fügen	75

81. Platzhalterelement für die Zuordnung	76	95. Ausgabeattribut umbenennen	99
82. Als Attribute zurückgegebene TCP- und HT-TP-Headereigenschaften	77	96. Ein Datensatz wurde gelesen und erfasst	100
83. Funktionskomponente der Fertigungslinie (AssemblyLine Function Component) ziehen	78	97. EasyETL-Fertigungslinie abgeschlossen	100
84. Speicherauszug für Work-Entry, gefolgt von statistischen Daten zur Fertigungslinie (dieser Screenshot ist in Englisch).	79	98. Konvertierung aktivieren.	101
85. Abgeschlossene Fertigungslinie 'TINA_WebServer'	80	99. Konvertierungsscript anzeigen	102
86. Einfache Web Interface für Ihre Lösung	80	100. Ausdruck auswerten	102
87. Begrüßungsseite	92	101. Ausgabedatenerfassung mit berechnetem Attribut 'FullName'	103
88. Abbildung 2. EasyETL-Workbench	93	102. XML-Ausgabe	104
89. Schaltfläche 'Neues Projekt'	94	103. Deltakonfiguration	105
90. Einfacher Fertigungslinieneditor	94	104. Alle Einträge sind unverändert und werden übersprungen	107
91. Quelleninformationen auswählen	95	105. Verknüpfungsbedingungen auswählen	108
92. Parameter für den Dateipfad konfigurieren	96	106. Befehlszeilenressourcen für die Ausführung des ETL-Jobs erstellen.	109
93. Verbindung testen und Schema erkennen	97	107. ETL-Job mit voller Geschwindigkeit ausführen	110
94. Eingabequelle konfiguriert	98	108. Verknüpfungsbedingungen für einen Eingabecorridor definieren	111

Informationen zu dieser Veröffentlichung

Die Veröffentlichung 'IBM® Security Directory Integrator - Erste Schritte' enthält Beschreibungen der grundlegenden Konzepte von Security Directory Integrator und Hintergrundinformationen zum Design effektiver Datenintegrationslösungen.

Dieses Dokument enthält Informationen zu Konzepten von IBM Security Directory Integrator sowie Beispiele, die Sie bei den ersten Schritten mit dem Produkt unterstützen.

Zugriff auf Veröffentlichungen und Terminologie

Lesen Sie die Beschreibungen der IBM Security Directory Integrator Version 7.2.0.1-Bibliothek und der Referenzliteratur, auf die Sie online zugreifen können.

In diesem Abschnitt finden Sie die folgenden Informationen:

- Eine Liste der Veröffentlichungen in der „IBM Security Directory Integrator-Bibliothek“.
- Links zu „Onlineveröffentlichungen“ auf Seite viii.
- Einen Link zur „IBM Terminologiewebsite“ auf Seite ix.

IBM Security Directory Integrator-Bibliothek

Die folgenden Dokumente sind in der IBM Security Directory Integrator-Bibliothek verfügbar:

- *IBM Security Directory Integrator Version 7.2.0.1 Federated Directory Server Administration Guide*
Diese Veröffentlichung enthält Informationen zum Entwerfen, Implementieren und Verwalten von Datenintegrationslösungen über die Federated Directory Server-Konsole. Sie enthält ferner Informationen zur Verwendung des Protokolls 'System for Cross-Domain Identity Management' (SCIM) und der Schnittstelle für das Identitätsmanagement.
- *IBM Security Directory Integrator Version 7.2.0.1 Erste Schritte*
Diese Veröffentlichung enthält ein kurzes Lernprogramm und eine Einführung in IBM Security Directory Integrator. Sie umfasst Beispiele zur Erstellung der benötigten Interaktionsabläufe und praktische Übungen, mit deren Hilfe Sie sich mit IBM Security Directory Integrator vertraut machen können.
- *IBM Security Directory Integrator Version 7.2.0.1 Benutzerhandbuch*
Diese Veröffentlichung enthält Informationen zur Verwendung von IBM Security Directory Integrator. Darüber hinaus enthält sie Anweisungen zum Entwurf von Lösungen mithilfe des Security Directory Integrator-Entwicklertools (des Konfigurationseditors) sowie zur Ausführung vordefinierter Lösungen über die Befehlszeile. Sie bietet außerdem Informationen zu den verwendeten Schnittstellen und Konzepten sowie zur Erstellung von Fertigungslinien.
- *IBM Security Directory Integrator Version 7.2.0.1 Installations- und Administratorhandbuch*
Diese Veröffentlichung enthält vollständige Informationen zur Installation, Migration von einer Vorversion, Konfiguration der Protokollierungsfunktion und zum Sicherheitsmodell, das der fernen Server-API von IBM Security Directory

Integrator zugrunde liegt. Darüber hinaus enthält sie Informationen dazu, wie Lösungen implementiert und verwaltet werden.

- *IBM Security Directory Integrator Version 7.2.0.1 Reference Guide*

Diese Veröffentlichung enthält detaillierte Informationen zu den einzelnen Komponenten von IBM Security Directory Integrator (Connector, Funktionskomponenten, Parser, Objekte usw.), die die Bausteine der Fertigungslinie darstellen.

- *IBM Security Directory Integrator Version 7.2.0.1 Problem Determination Guide*

Diese Veröffentlichung enthält Informationen zu den Tools, Ressourcen und Verfahren in IBM Security Directory Integrator, die Sie bei der Identifikation und Lösung von Problemen unterstützen können.

- *IBM Security Directory Integrator Version 7.2.0.1 Message Guide*

Diese Veröffentlichung enthält eine Liste aller Informationsnachrichten, Warnungen und Fehlernachrichten zu IBM Security Directory Integrator.

- *IBM Security Directory Integrator Version 7.2.0.1 Password Synchronization Plug-ins Guide*

Diese Veröffentlichung enthält umfassende Informationen zur Installation und Konfiguration der folgenden fünf IBM Plug-ins zur Kennwortsynchronisation: Windows Password Synchronizer, Sun Directory Server Password Synchronizer, IBM Security Directory Server Password Synchronizer, Domino Password Synchronizer und Password Synchronizer für UNIX und Linux. Darüber hinaus sind Konfigurationsanweisungen für den LDAP- und den JMS-Kennwortspeicher enthalten.

- *IBM Security Directory Integrator Version 7.2.0.1 Releaseinformationen*

Diese Veröffentlichung enthält Beschreibungen neuer Funktionen sowie kurzfristig aktualisierte Informationen zu IBM Security Directory Integrator, die nicht mehr in die Dokumentation aufgenommen werden konnten.

Onlineveröffentlichungen

IBM stellt Produktveröffentlichungen, wenn das Produkt freigegeben wird und wenn die Veröffentlichungen aktualisiert werden, an den folgenden Positionen bereit:

IBM Security Directory Integrator-Bibliothek

Auf der Website mit der Produktdokumentation (<http://www-01.ibm.com/support/knowledgecenter/SSCQGF/welcome>) wird die Einführungsseite angezeigt und Sie finden Informationen über die Navigation in dieser Bibliothek.

IBM Security Systems Documentation Central

IBM Security Systems Documentation Central bietet eine alphabetische Liste aller IBM Security Systems-Produktbibliotheken sowie Links zur Online-dokumentation für bestimmte Versionen der einzelnen Produkte.

IBM Publications Center

Die Website mit dem IBM Publications Center (<http://www-05.ibm.com/e-business/linkweb/publications/servlet/pbi.wss>) bietet angepasste Suchfunktionen, mit denen Sie alle benötigten IBM Veröffentlichungen finden.

Zugehörige Informationen

Referenzinformationen zu IBM Security Directory Integrator finden Sie hier:

- IBM Security Directory Integrator verwendet den JNDI-Client von Oracle. Informationen zum JNDI-Client finden Sie in der Veröffentlichung *Java Naming and Directory Interface™ Specification* unter <http://download.oracle.com/javase/7/docs/technotes/guides/jndi/index.html>.
- Informationen, die Sie bei der Beantwortung von Fragen zu IBM Security Directory Integrator unterstützen, finden Sie unter https://www-947.ibm.com/support/entry/myportal/over-accesspubsview/software/security_systems/tivoli_directory_integrator.

IBM Terminologiewebsite

Die IBM Terminologiewebsite enthält die konsolidierte Terminologie aus Produktbibliotheken an einer einzigen Position. Sie finden die Terminologiewebsite unter der folgenden Adresse: <http://www.ibm.com/software/globalization/terminology>.

Eingabehilfen

Funktionen zur behindertengerechten Bedienung (Eingabehilfefunktionen) unterstützen Benutzer mit körperlichen Behinderungen, wie z. B. eingeschränkter Bewegungsfähigkeit oder Sehkraft, beim erfolgreichen Einsatz von Softwareprodukten. Dieses Produkt unterstützt behindertengerechte Tools, die die Elemente der Benutzerschnittstelle ansagen und die Navigation in dieser Schnittstelle vereinfachen. Sie können alle Funktionen der grafischen Benutzerschnittstelle anstatt mit der Maus auch über die Tastatur aufrufen.

Weitere Informationen finden Sie im Anhang zu den Eingabehilfen in der Veröffentlichung *Directory Integrator - Konfiguration*.

Technische Schulung

Informationen zu technischen Schulungen finden Sie auf der Website für IBM Education unter der Adresse <http://www.ibm.com/software/tivoli/education>.

Informationen zur Unterstützung

IBM Support bietet Unterstützung bei Codeproblemen und kurzen Routinefragen zu Installation oder Verwendung. Sie können direkt über <http://www.ibm.com/software/support/probsub.html> auf die IBM Software Support-Website zugreifen.

Fehlerbehebung bietet detaillierte Informationen zu folgenden Themen:

- Welche Informationen vor der Kontaktaufnahme mit IBM Support zusammengestellt werden sollten.
- Die verschiedenen Methoden zur Kontaktaufnahme mit IBM Support.
- Die Verwendung von IBM Support Assistant.
- Anweisungen und Ressourcen zur Fehlerbestimmung, um das Problem einzugrenzen und zu beheben.

Erklärung zu geeigneten Sicherheitsvorkehrungen

Zur Sicherheit von IT-Systemen gehört der Schutz von Systemen und Informationen in Form von Vorbeugung, Erkennung und Reaktion auf unbefugten Zugriff innerhalb des Unternehmens und von außen. Unbefugter Zugriff kann dazu führen, dass Informationen geändert, gelöscht, veruntreut oder missbräuchlich verwendet werden. Ebenso können Ihre Systeme beschädigt oder missbräuchlich verwendet werden, einschließlich zum Zweck von Attacken. Kein IT-System oder Produkt kann umfassend als sicher betrachtet werden. Kein einzelnes Produkt, kein einzelner Service und keine einzelne Sicherheitsmaßnahme können eine unbefugte Verwendung oder einen unbefugten Zugriff mit vollständiger Wirksamkeit verhindern. IBM Systeme, Produkte und Services werden als Teil eines umfassenden Sicherheitskonzepts entwickelt, sodass die Einbeziehung zusätzlicher Betriebsprozesse erforderlich ist. IBM übernimmt keine Gewähr dafür, dass Systeme, Produkte oder Services vor zerstörerischen oder unzulässigen Handlungen Dritter geschützt sind oder dass Systeme, Produkte oder Services Ihr Unternehmen vor zerstörerischen oder unzulässigen Handlungen Dritter schützen.

Kapitel 1. Einführung

Das vorliegende Handbuch bietet eine einfache Einführung in ein einfaches System. Das Wort *einfach* ist hier ganz wörtlich zu nehmen, denn der beste Weg zum Begreifen eines komplexen Problems besteht darin, es in einfachere Teile aufzugliedern, die leichter zu verstehen sind, und sich diese schrittweise zu erarbeiten. Teile und herrsche! Diese Vorgehensweise verwenden Sie intuitiv, um alltägliche Problemstellungen zu lösen, und sie lässt sich auch anwenden, um den Informationsaustausch innerhalb eines Büros, eines Unternehmens oder weltweit zu realisieren.

Wenn Sie schnell mit dem Extrahieren der Daten aus Dateien, Verzeichnissen, Datenbanken oder von Lotus Notes beginnen wollen und diese Daten an eine andere Stelle übertragen möchten, dann können Sie direkt mit dem Anhang „EasyETL-Leitfaden“, auf Seite 91 fortfahren. Mit dieser Funktion können Sie das Potenzial von IBM Security Directory Integrator nutzen, ohne sich vorher Wissen über die zentralen Bestandteile aneignen zu müssen. Sie brauchen nur Ihre Quelle und Ihr Ziel auswählen, auf 'Ausführen' klicken und schon kommt es zum gewünschten Datenfluss. Wenn Sie hingegen eine größere Kontrolle darüber erlangen möchten, wie Daten gelesen, gefiltert, aufbereitet, konvertiert und versetzt werden, dann lesen Sie hier weiter. Sie können „EasyETL-Leitfaden“, auf Seite 91 immer noch im Anschluss lesen.

IBM Security Directory Integrator¹ wurde ausgehend von der Voraussetzung entworfen und entwickelt, dass selbst die komplexesten Integrationsprobleme in drei grundlegende Teile zerlegt werden können:

- Die Systeme, die bei der Datenübertragung verwendet werden, auch als *Datenquellen* bezeichnet.
- Die *Datenflüsse* zwischen diesen Systemen.
- Die *Ereignisse*, die die Datenflüsse auslösen.

Mit IBM Security Directory Integrator setzen Sie dieses atomare Verständnis des Integrationsproblems direkt in eine Lösung um, die Sie schrittweise entwickeln und bei der Sie die einzelnen Datenflüsse nacheinander in das Gesamtkonzept integrieren, wobei Sie das Ergebnis kontinuierlich anhand entsprechender Rückmeldungen und Überprüfungsoperationen verifizieren können. Durch diesen Ansatz kann der Aufwand für Integrationsprojekte leichter geschätzt und geplant werden. Dieser Aufwand kann in vielen Fällen auf eine reine Bestandsaufnahme der zu implementierenden Datenflüsse und das Ermitteln der hierbei anfallenden Kosten reduziert werden. Wenn Sie eine Task in nachvollziehbaren Schritten ausführen, können Sie den Aktionären regelmäßig den Fortschritt veranschaulichen.

IBM Security Directory Integrator beschleunigt desweiteren die Entwicklung durch eine Überbrückung der technischen Unterschiede zwischen Ihren Datenquellen, wodurch Sie mehr Zeit haben, um sich auf die Geschäftsanforderungen zu konzentrieren.

Durch Nutzung der Leistungsfähigkeit von Eclipse ist die IBM Security Directory Integrator-Entwicklungsumgebung umfassend und erweiterbar. Aus Integrations-

1. Lassen Sie sich nicht durch die Bezeichnung (Directory - Verzeichnis) beirren: IBM Security Directory Integrator ist nicht auf die Arbeit mit Verzeichnissen begrenzt und unterstützt alle großen Datenspeicher, den Transport von Daten, Protokolle und Anwendungsprogrammierschnittstellen, wobei natürlich auch LDAP-Verzeichnisse eingeschlossen sind.

projekten werden Bibliotheken mit Komponenten und Geschäftslogik erstellt, die schnell wiederverwendet werden können, um neue Problemstellungen zu lösen. Dadurch können Teams unternehmensweit IBM Security Directory Integrator-Ressourcen gemeinsam nutzen. Auf diese Weise können separat entwickelte Projekte (sogar Einzellösungen) direkt in eine einheitliche Gesamtstruktur integriert werden.

Das vorliegende Dokument bietet eine Einführung zu der oben beschriebenen Methodik des 'Vereinfachens und Lösens'. Sie unternehmen erste Schritte und lernen die elegante Schlichtheit des IBM Security Directory Integrator-Toolsets kennen. Dabei handelt es sich insbesondere um diese beiden Programme:

- Die Entwicklungsumgebung, als *Konfigurationseditor* (Configuration Editor, CE) bezeichnet.
- Die Laufzeitsteuerkomponente, die einfach als *Server* bezeichnet wird.

Sie bauen Ihre IBM Security Directory Integrator-Lösungen mit dem Konfigurationseditor auf, während ein oder mehrere Server als Basis für die Ausführung verwendet werden. Diese Programme arbeiten zusammen, wodurch der Benutzer den Eindruck eines reibungslosen Ablaufs gewinnt und sogar die Möglichkeit hat, über Plattformen hinweg zu arbeiten. Zum Beispiel können Sie auf Ihrem Laptop entwickeln, während Sie Lösungen, die über Fernzugriff auf einem Mainframe-Computer ausgeführt werden, testen und ein Debug für sie ausführen.

Scripterstellung in JavaScript

Wie oben erwähnt, können Sie mit IBM Security Directory Integrator sehr schnell Integrationslösungen aufbauen. Um die integrierte automatisierte Funktionalität durch Ihre eigenen angepassten Verarbeitungs- und Datenflussabläufe zu erweitern, müssen Sie Scriptfragmente schreiben.

Für die Scripterstellung wird JavaScript verwendet und IBM Security Directory Integrator stellt die Komponente IBM JSEngine bereit, mit der eine schnelle, zuverlässige Scriptingumgebung zur Verfügung steht. Daher müssen Sie die zentralen Bestandteile der Sprache JavaScript verstehen. Es gibt mehrere gute Veröffentlichungen in gedruckter Form und online, um sich in JavaScript einzuarbeiten. Suchen Sie in den IBM Security Directory Integrator-Newsgruppen und auf den entsprechenden Webseiten nach Empfehlungen und Links.

Weitere Informationen zur Scripterstellung in IBM Security Directory Integrator finden Sie im *Directory Integrator - Konfiguration*.

IBM Security Directory Integrator installieren

IBM Security Directory Integrator kann innerhalb weniger Minuten installiert und sofort zum Erstellen, Testen und Implementieren von Lösungen eingesetzt werden. Das Produkt kann auf einer Vielzahl verschiedener Plattformen eingesetzt werden; hierzu gehören beispielsweise Microsoft Windows, IBM AIX, IBM System z sowie eine Reihe verschiedener UNIX- und Linux-Umgebungen.

Es gibt drei wichtige Pfade beim Installieren von IBM Security Directory Integrator und das Installationsprogramm fordert sie auf, die ersten beiden anzugeben:

1. Das *Installationsverzeichnis*, in dem die Programmdateien sowie die Stapeldateien oder Scripte enthalten sind, die zum Starten der verschiedenen Tools verwendet werden.
2. Das *Lösungsverzeichnis* (Solution Directory, 'SolDir'). Das Lösungsverzeichnis ist während der Ausführung von IBM Security Directory Integrator der aktuelle

Ordner. Sie werden bemerken, dass die für die Initialisierung der Entwicklungsumgebung 'Konfigurationseditor' (ibmditk) und des Servers (ibmdisrv) erforderlichen Stapeldateien und Scripts beide mit einem Befehl starten, in dem Sie aufgefordert werden, zum Lösungsverzeichnis zu wechseln. Dadurch werden alle relativen Pfade, die in Ihrer Lösung verwendet werden, von Ihrem Lösungsverzeichnis aus angezeigt.

3. Der Ordner *Arbeitsbereich*. Hier werden Ihre Projekt- und Ressourcendateien² gespeichert. Dieser Ordner nimmt in Ihrem Lösungsverzeichnis standardmäßig den Namen "workspace" (Arbeitsbereich) an.

Weitere Informationen zur Installation von IBM Security Directory Integrator finden Sie in den *IBM Security Directory Integrator Installationsanweisungen* in der Veröffentlichung *Installation und Verwaltung*.

Lernprogrammdateien installieren

Die Lernprogrammübungen in diesem Buch erfordern als Unterstützung Datendateien, die sich im Unterordner `examples/Tutorial` des IBM Security Directory Integrator-Installationsverzeichnisses befinden. Beispiel: Eine Standardinstallation von Windows würde diese Dateien im folgenden Verzeichnis speichern:

```
C:\Programme\IBM\TDI\V7.2\examples\Tutorial
```

Das Verzeichnis 'Tutorial' sollte folgende Dateien enthalten:

- `CreatePhoneDB.assemblyline`
- `index.html`
- `OtherPage.html`
- `People.csv`
- `PhoneNumbers.xml`
- `readme.txt`
- `Return web page.script`

Anmerkung: Wie im vorhergehenden Abschnitt erwähnt, werden Sie vom Installationsprogramm aufgefordert, die Speicherposition Ihres Lösungsverzeichnisses anzugeben. Dort werden Ihre Projekt- und Ressourcendateien gespeichert. In der Regel handelt es sich um einen Unterordner mit dem Namen `My Documents\TDI` in Ihrem Ausgangsverzeichnis.

Kopieren Sie den Ordner `Tutorial` in Ihr Lösungsverzeichnis, um von den Konfigurationseditortools aus schneller auf ihn zugreifen zu können.

Vereinfachen und Lösen von Problemen

Dieser Abschnitt unterstützt Sie dabei, den richtigen Anfang für den Entwurf einer Datenintegrationslösung zu finden. Obwohl die Entwurfsstrategie inkrementell ist, eignet sie sich für jede Größe von Integrations- und Systemimplementierungsprojekten, einschließlich großer Projekte.

Verwenden Sie eine Strategie wie die vorliegende, um eine schrittweise Datenintegrationslösung mithilfe von IBM Security Directory Integrator zu entwerfen:

- Reduzieren Sie die Komplexität eines Problems, indem Sie es in kleinere Teile aufspalten, die dann einfacher bewältigt werden können.

2. Diese Begriffe werden in Kapitel 2, „IBM Security Directory Integrator - Erste Schritte“, auf Seite 11 erläutert.

- Beginnen Sie mit einem Teilbereich der Gesamtlösung. Vorzugsweise sollte dies ein Teil sein, der in ein bis zwei Wochen bewältigt werden kann.
- Beginnen Sie mit einem Teilbereich der Gesamtlösung, der eigenständig funktionieren kann.

Handhabung komplexer Projekte

Die beste Strategie für die Durchführung großer Integrations- und Systemimplementierungsprojekte ist es, die Komplexität eines Problems durch seine Aufspaltung in kleinere Teile zu reduzieren, die dann einfacher bewältigt werden können. Sobald dies erledigt ist, beginnen Sie mit der Arbeit an einem Teilbereich der Gesamtlösung. Vorzugsweise sollte dies ein Teil sein, der eigenständig implementiert werden kann. Auf diese Weise erbringt sie bereits einen Investitionsertrag, während Sie noch mit der Lösung der anderen Teilbereiche der Gesamtlösung beschäftigt sind.

Wenn Sie den Teilbereich isoliert haben, den Sie als Erstes bearbeiten wollen, sollten Sie diesen noch weiter vereinfachen, indem Sie sich auf die Basiseinheiten der Kommunikation konzentrieren: auf die Datenflüsse selbst. Sie können nun mit der Implementierung beginnen. Die Entwicklung von Integrationslösungen erfolgt unter Verwendung des IBM Security Directory Integrator-Konfigurationseditors (Configuration Editor, CE) in einer Reihe von Zyklen, die aus Versuchs-, Test- und Optimierungsphasen bestehen. Auf diese Weise wird der Entwicklungsprozess in mehreren Wiederholungs- und Versuchsläufen durchgeführt. Dadurch können Sie nicht nur Ihre Kenntnisse der Systeminstallation vertiefen, sondern auch parallel zu Ihrem wachsenden Verständnis der gegebenen Problemstellung und ihrer Auswirkungen auf die vorhandene Infrastruktur Ihre Integrationslösung immer weiter entwickeln.

Zugehörige Themen

In den folgenden Themen finden Sie Erläuterungen dazu, wie IBM Security Directory Integrator es Ihnen ermöglicht, Daten mit Fertigungslinien umzusetzen.

- „Kernel/Komponentenarchitektur“
- „Datenmodell 'Eintrag-Attribut-Wert'“ auf Seite 5
- „Datenflüsse sind Fertigungslinien“ auf Seite 7

Kernel/Komponentenarchitektur

Ein grundlegendes Qualitätsmerkmal von IBM Security Directory Integrator ist das Kernel/Komponentendesign.

Der Begriff *Kernel* bezieht sich hier auf das RID-Framework (Rapid Integration Development, RID), das Ihnen einen schnellen Aufbau Ihrer Integrationslösungen ermöglicht und automatisierte Ausführungslogik bereitstellt, um diese auszuführen. Funktionen, die Sie üblicherweise manuell codieren müssten (und die deshalb häufig vernachlässigt werden), wie Protokoll-/Tracemodule, Verbindungswiederherstellung, Änderungserkennung, Fehlerbehandlung und eine externe Verwaltungs-API, stehen selbst für den einfachsten Datenfluss sofort zur Verfügung.

Zusätzlich zu dieser generischen Kernelfunktionalität bietet IBM Security Directory Integrator eine Reihe von datenquellenspezifischen Komponenten: Unterstützungsobjekte, die die technischen Details bei der Interaktion mit Ihren Datenquellen abstrahieren und damit vereinfachen. Die beiden Komponententypen, die Sie am häufigsten verwenden werden, sind *Connector* und *Parser*.

Connector bieten Konnektivität zu einer Vielfalt von Datenquellen sowie eine inhärente Handhabung von strukturierten Daten, die nicht von der zugrunde liegenden Organisation der Daten abhängig ist. Einige Connector dienen auch als Ereignishandler. Sie stellen z. B. Bindungen zu IP-Ports her und warten auf eingehende Verbindungsanforderungen oder sie sind empfangsbereit für Änderungen, die in Verzeichnissen, Datenbanken oder Dateien auftreten.

Parser werden andererseits dazu verwendet, um unstrukturierte Daten zu handhaben, das heißt, Byteströme, wie sie in Dateien, POP3/SMTP-E-Mails, Nachrichten einer Nachrichtenwarteschlange (MQ) und Daten enthalten sind, die über IP-Ports hinweg fließen.

IBM Security Directory Integrator bietet eine erweiterbare Bibliothek mit Connectors und Parsern, die jeweils für die Arbeit mit einem bestimmten System, Service, Transport, Format oder einer bestimmten API entworfen wurden. Die Austauschbarkeit der IBM Security Directory Integrator-Komponenten untereinander ermöglicht es Ihnen, eine Lösung zu erstellen, die auf Testdaten wie z. B. Textdateien basiert und in der Sie anschließend die verwendeten Connectors einfach austauschen können, um Ihre Lösung zur Überprüfung und Implementierung auf aktuelle Daten zu verweisen.

Darüber hinaus sind IBM Security Directory Integrator-Komponenten unkompliziert in der Verwendung und einfach zu erstellen und zu erweitern. Sie können Ihre Bibliothek so erweitern, dass sie angepasste Datenquellen und Services bearbeitet. Dazu laden Sie neue Komponenten von einer Community-Website herunter, schreiben eigene Komponenten in Java™ oder erstellen und testen sie mithilfe von Scripts direkt im Konfigurationseditor.

Datenmodell 'Eintrag-Attribut-Wert'

Es gibt von System zu System große Unterschiede, wie Daten organisiert und gespeichert werden:

- In Datenbanken werden Informationen in Zeilen gespeichert, die in der Regel eine feste Anzahl von Spalten umfassen, von denen jede einen einzigen Wert pro Datensatz aufweist.
- Verzeichnisse verwalten objektorientierte Einträge, die eine unterschiedliche Anzahl von Attributen enthalten können. Diese enthalten wiederum null, einen oder mehrere Werte³.
- Lotus Domino-Datenbanken enthalten Dokumente, die aus Feldern bestehen, die jeweils als einwertiges oder mehrwertiges Feld definiert werden können.
- Andere Systeme wiederum stellen Ihren Dateninhalt als Knoten, Objekte, Dateien, formatierte Byteströme oder Schlüssel-Wert-Sätze dar.

Damit eine Kommunikation für alle Teilnehmer aussagekräftig ist, müssen die Datenformate miteinander kompatibel sein oder sie müssen umgesetzt werden, sodass sie für jedes beteiligte System geeignet sind. Dies wird als *Datenmarshaling* bezeichnet und ist oft die erste Hürde, die ein Integrationsspezialist zu nehmen hat und die schnell eine beträchtliche Menge von Projektressourcen verbrauchen kann, bis sie überwunden ist. IBM Security Directory Integrator-Connector bearbeiten dies für sie, indem sie quellenspezifische Typen automatisch in eine konsistente, kanonische Darstellung konvertieren. Einzelne Datenwerte werden in relevante Java-Objekte umgesetzt, bei denen vergleichbare native Typen in gleicher Weise dargestellt

3. Sie besitzen beispielsweise mehrere E-Mail-Adressen, die alle in dem mehrere Werte aufweisenden Attribut 'mail' im Mitarbeiterverzeichnis Ihres Unternehmens gespeichert werden können

werden. Beispielsweise werden Zeilen, die aus Dateien, LDAP-Zeichenfolgeattributen, Domino-Textfeldern und CHAR- und VARCHAR-Spalten von Managementsystemen für relationale Datenbanken eingelesen werden, alle von ihren jeweiligen Connectors in `java.lang.String` konvertiert.

Diese mit Datenmarshaling bearbeiteten Werte werden anschließend in Attributen akkumuliert. Dies sind spezialisierte Java-Objekte, die von IBM Security Directory Integrator definiert werden. Wie oben bereits erwähnt, lassen einige Quellen nur einen einzigen Wert pro Spalte oder Feld zu, während andere zulassen, dass unter demselben Attributnamen mehrere Werte gespeichert werden. Das IBM Security Directory Integrator-Attribut unterstützt sowohl einwertige als auch mehrwertige Implementierungen und kann sogar, falls erforderlich, keine Werte enthalten, beispielsweise wenn eine Spalte, die Nullwerte enthalten darf, in einer Datenbank dargestellt wird.

Alle Attribute, die eine einzelne Dateneinheit bilden (d. h. eine Datei, eine Nachricht, ein Dokument usw.) werden in einem weiteren IBM Security Directory Integrator-Objekt gesammelt, das als *Eintrag* bezeichnet wird. Ein Eintrag kann eine beliebige Anzahl von Attributen oder kein Attribut enthalten.

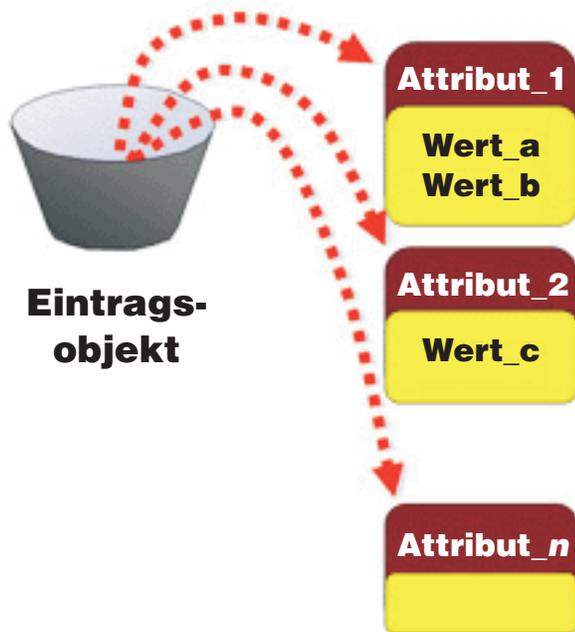


Abbildung 1. Das Datenmodell 'Eintrag-Attribut-Wert'

Jeder Datenfluss weist einen primären 'Eintragsbucket' auf, der als zugehöriger 'Work-Eintrag' bezeichnet wird. Wenn ein Connector Daten einliest, erstellt er Attribute und versetzt diese in den Work-Eintrag. Alle für die Ausgabe konfigurierten Connector verwenden Attribute, die im Work-Eintrag vorgefunden werden, um Änderungen an Zielsystemen auszuführen.

Dieses zweiphasige Verfahren bietet eine nahezu unbegrenzte Flexibilität bei der Übertragung, Transformation, Filterung und Aufbereitung von Daten. Es bedeutet auch, dass Sie Ihren Datenfluss anfänglich vollständig mit Eingabecorridor erstellen und die Daten anschließend beim Lesen und Bearbeiten mit dem Konfigurationseditor interaktiv untersuchen können, bevor Sie Verbindungen zu Ausgabesystemen überhaupt in Betracht ziehen müssen.

Später werden Sie sehen, dass der IBM Security Directory Integrator-Eintrag komplexe, hierarchisch aufgebaute Daten ebenso leicht bearbeitet wie unstrukturierte Schemata.

Datenflüsse sind Fertigungslinien

Jeder Datenfluss in Ihrer Lösung ist als IBM Security Directory Integrator *Fertigungslinie* implementiert. In dieser und anderen Dokumentationen wird für 'Fertigungslinie' auch die Abkürzung 'FL' verwendet.

Fertigungslinien sind geordnete Listen mit Komponenten, die einen einzigen, kontinuierlichen Pfad von Eingabequellen zu Zielen bilden. Durch den vom Kernel bereitgestellten, integrierten Funktionsablauf werden die Komponenten miteinander verbunden und die im Work-Eintrag enthaltenen Daten werden von einer Komponente zur nächsten weitergeleitet.

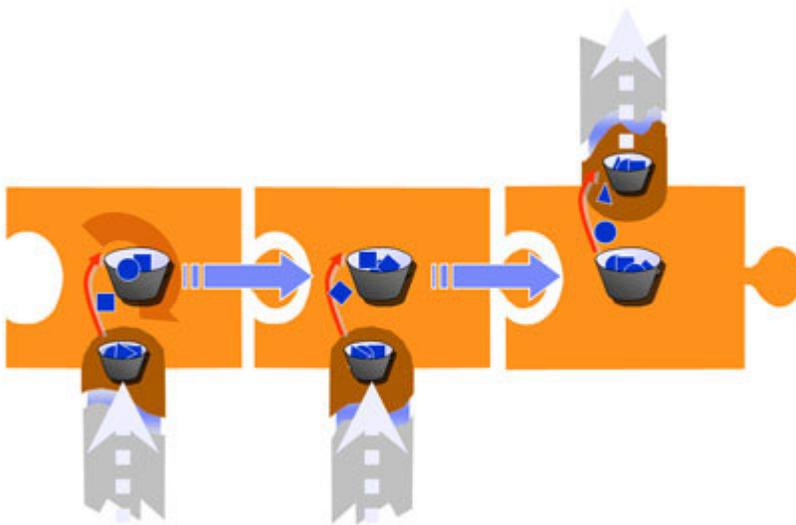


Abbildung 2. Datenfluss durch eine Fertigungslinie

Viel besser als die ausführlichste Erläuterung kann eine Abbildung den Sachverhalt vermitteln. Dies trifft auch für das oben stehende Diagramm zu. Die drei Puzzleteile stellen Connector dar, die miteinander verbunden sind und so eine Fertigungslinie bilden. Der dunkler abgebildete 'Stamm' jedes Puzzleteils kennzeichnet den datenquellenspezifischen Teil des Connectors, das heißt, die *Schnittstelle* zu dem verbundenen System, auch *Connectorschnittstelle* genannt (Connector Interface, CI). Der heller abgebildete restliche Teil jedes Puzzleteils ist ein Abbild der generischen Funktionalität des Kernels, der bewirkt, dass alle Komponenten auf gleiche und vorhersehbare Weise arbeiten, wodurch sie miteinander verknüpft werden können und automatisierte Muster von Funktionsabläufen mit Steuerungspunkten für die Anpassung bereitgestellt werden⁴.

Diese Abbildung veranschaulicht einige weitere wichtige Konzepte. Beispielsweise ist zusätzlich zu dem oben abgebildeten Work-Eintrag, der von Komponente zu Komponente durch die Fertigungslinie weitergeleitet wird, ein zusätzlicher Java-"Bucket" vorhanden, der in jede der Connectorschnittstellen verschachtelt ist. Loka-

4. Wie Sie sehen, spiegelt jede Fertigungslinienkomponente die Kernel-/Komponentenarchitektur von IBM Security Directory Integrator wider. Wenn Sie eine eigene Komponente erstellen wollen, müssen Sie nur die zugehörige Schnittstelle implementieren. Der Fertigungslinienwrapper und die vielfältigen Möglichkeiten der integrierten Funktionalität der Fertigungslinie stehen automatisch zur Verfügung, ein Service des IBM Security Directory Integrator-Kernels.

le Eintragsobjekte werden zum Zwischenspeichern von Daten während Lese- und Schreiboperationen verwendet, die von der jeweiligen Connectorschnittstelle ausgeführt werden. Ein solches Eintragsobjekt wird als *Conn-Eintrag* der Connectorschnittstelle bezeichnet.

Beachten Sie nun die geschwungenen Pfeile, die den Datenfluss zwischen den verschiedenen Conn-Einträgen und dem Work-Eintrag der Fertigungslinie darstellen. Dies sind *Attributzuordnungen* und jede von ihnen stellt einen Satz von Regeln für die Übertragung von Daten und die Konvertierung auf dem zugehörigen Weg in oder aus der Fertigungslinie dar. Die Zuordnungen, die Daten aus dem Conn-Eintrag in den Work-Eintrag befördern, werden als *Eingabezuordnungen* bezeichnet, da sie festlegen, welche Daten an die Fertigungslinie weitergegeben werden. Der Pfeil im Puzzleteil ganz rechts, das Daten zeigt, die sich in die andere Richtung bewegen - aus dem Work-Eintrag in den Conn-Eintrag - wird als *Ausgabezuordnung* bezeichnet.

Daraus, dass es zu einem Zeitpunkt immer nur einen Work-Eintrag gibt, können Sie schließen, dass Fertigungslinien Elemente nacheinander verarbeiten. Beispiel: Eine einzelne Datenbankzeile, einen einzelnen Verzeichniseintrag, eine einzelne Nachricht einer Nachrichtenwarteschlange (MQ) usw. Dies ist ein weiterer wichtiger Aspekt von IBM Security Directory Integrator. Obwohl eine Fertigungslinie hundert oder sogar tausend Einträge pro Sekunde zyklisch verarbeiten kann⁵ ist dies ein wichtiger Aspekt, den Sie beim Entwerfen Ihrer Lösung berücksichtigen müssen. Natürlich ist es möglich, Arbeit auf mehrere Fertigungslinien zu verteilen. Informationen dazu und weitere Verfahren für eine Optimierung der Fertigungslinienleistung finden Sie in anderen IBM Security Directory Integrator-Dokumentationen.

Erste Schritte

Beginnen Sie ein Integrationsprojekt am besten damit, das bestehende Problem als Diagramm darzustellen.

Nehmen Sie einen Bleistift und ein Stück Papier zur Hand und stellen Sie die gewünschten Datenflüsse in groben Zügen dar. Diese Vorgehensweise erleichtert nicht nur die Visualisierung der auszuführenden Arbeiten, sondern sie dient auch als Entwurf für die Implementierung dieser Datenflüsse in IBM Security Directory Integrator.

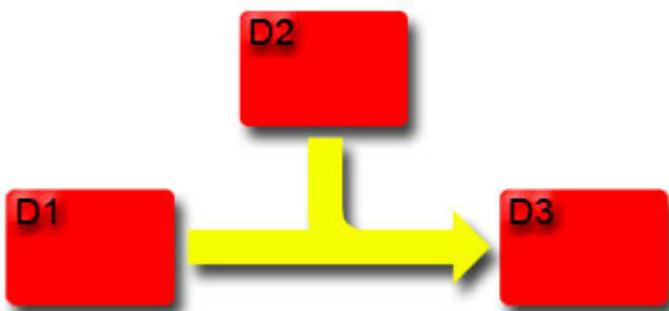


Abbildung 3. Lernprogrammszenario

5. Die Leistung hängt vom Design und der Komplexität der Fertigungslinie und der Konfiguration des Systems ab, das den Server ausführt.

Der erste Schritt bei der Erstellung einer IBM Security Directory Integrator-Lösung besteht darin, Datenflüsse zwischen Datenquellen in Fertigungslinien umzusetzen, die aus Connector bestehen. Der Leitsatz von IBM Security Directory Integrator 'Vereinfachen und Lösen' gibt vor, dass Ihre Lösung aus einzelnen, aufeinander aufbauenden Schritten entwickelt und der Beginn so einfach wie möglich sein sollte.

Sehen Sie sich zur besseren Veranschaulichung das Beispielszenario an, das Sie für Ihre erste Fertigungslinie verwenden. An dieser Integrationstask sind drei Datenquellen beteiligt, die mit D1, D2 und D3 bezeichnet sind. Als gewünschte Lösung wird angestrebt, den Inhalt von D1 auf D3 zu migrieren und diese Daten durch Daten zu erweitern, die aus D2 stammen. Wenn Sie diese Anforderung in eine Fertigungslinie umsetzen, erhalten Sie drei Connector, einen pro Datenquelle:

1. Der erste Connector, der D1 *iteriert* und diese Daten dem Datenfluss zuführt.
2. Auf diesen Connector folgt ein zweiter, der zugehörige Datensätze in D2 *sucht* und diese Werte mit den aus D1 stammenden Werten zusammenführt.
3. Schließlich ein dritter Connector, der so konfiguriert ist, dass er diese erweiterten Datensätze zu D3 *hinzufügt*.

Anstatt das gesamte Problem sofort in Angriff zu nehmen, ermöglicht IBM Security Directory Integrator es Ihnen, die Tasks zu vereinfachen, indem Sie mit lediglich zwei Connector beginnen: Mit einem Connector, der den Inhalt von D1 in die Fertigungslinie einliest und mit einem weiteren, der diese Werte in D3 schreibt. Sobald diese Minimallösung einer Fertigungslinie ordnungsgemäß funktioniert, kann sie anschließend mit dem Connector zu D2 erweitert werden, um zusätzliche Attribute einzufügen. Exakt auf diese Weise werden Sie Ihre erste IBM Security Directory Integrator-Lösung erstellen. Der verbleibende Teil dieses Handbuchs umfasst die Schritte, mit denen Sie durch diesen Prozess geführt werden.

Kapitel 2. IBM Security Directory Integrator - Erste Schritte

Dieser Abschnitt bietet Informationen, die für das Verständnis der Grundlagen von IBM Security Directory Integrator nützlich sind, sowie eine Reihe von Lernprogrammübungen, mit denen Sie praktische Erfahrungen mit der Entwicklungsumgebung sammeln können.

Der erste Schritt zum Kennenlernen des Produkts ist das Starten der IBM Security Directory Integrator-Entwicklungstools, die auch als Konfigurationseditor bezeichnet werden (Configuration Editor, CE).

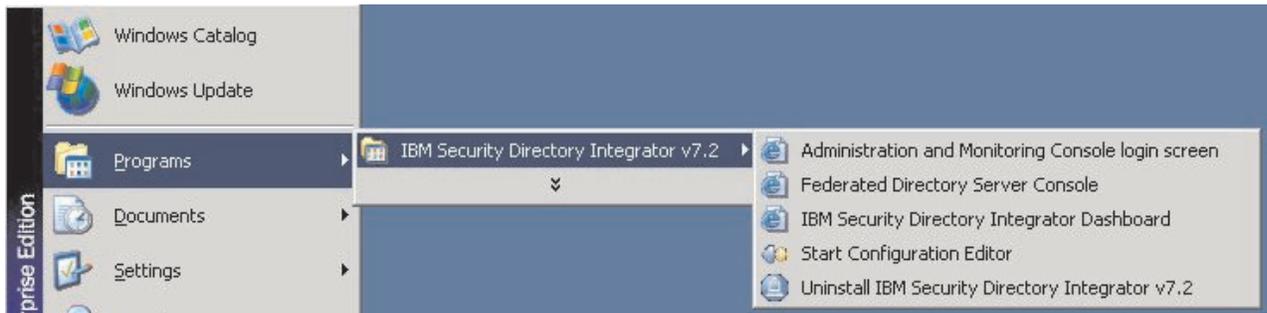


Abbildung 4. Konfigurationseditor starten

Wenn Sie den Konfigurationseditor zum ersten Mal aufrufen, wird Ihnen dieses Dialogfenster angezeigt, in dem Sie Ihren Arbeitsbereich angeben.

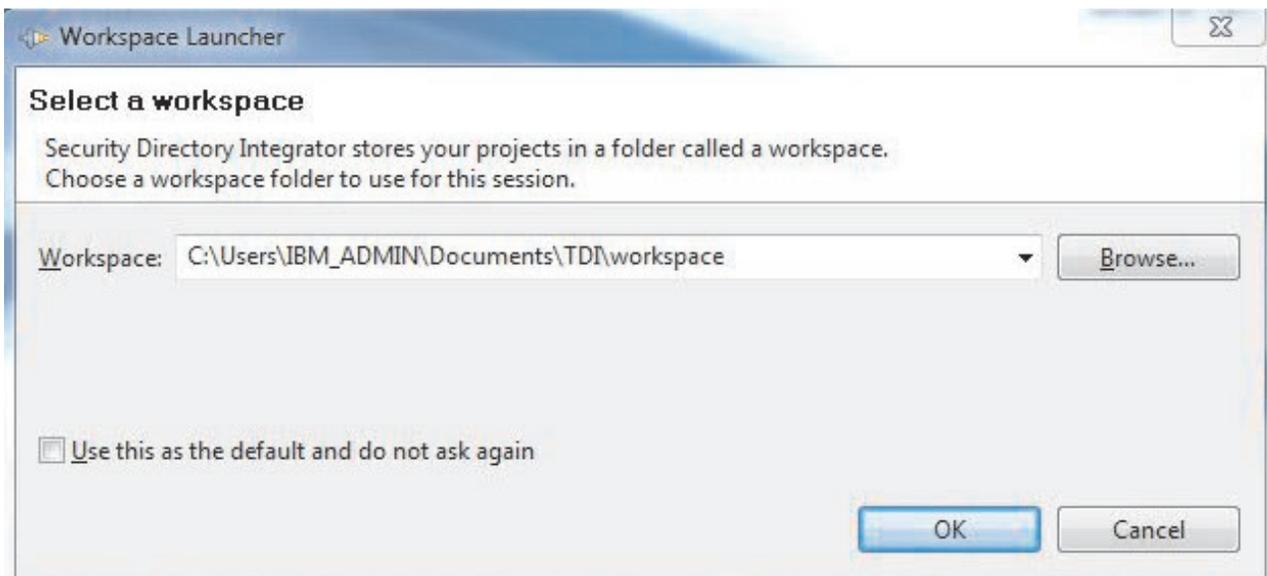


Abbildung 5. Arbeitsbereich auswählen

Ihr Arbeitsbereich ist der Bereich, in dem der Konfigurationseditor Ihre Projektdateien einschließlich Komponenten und Fertigungslinien speichert. Er befindet sich in der Regel in Ihrem Lösungsverzeichnis.

Wenn Sie die gewünschte Speicherposition für Ihren Arbeitsbereich ausgewählt haben, klicken Sie auf die Schaltfläche **OK**. Nun wird die Eingangsanzeige geöffnet.

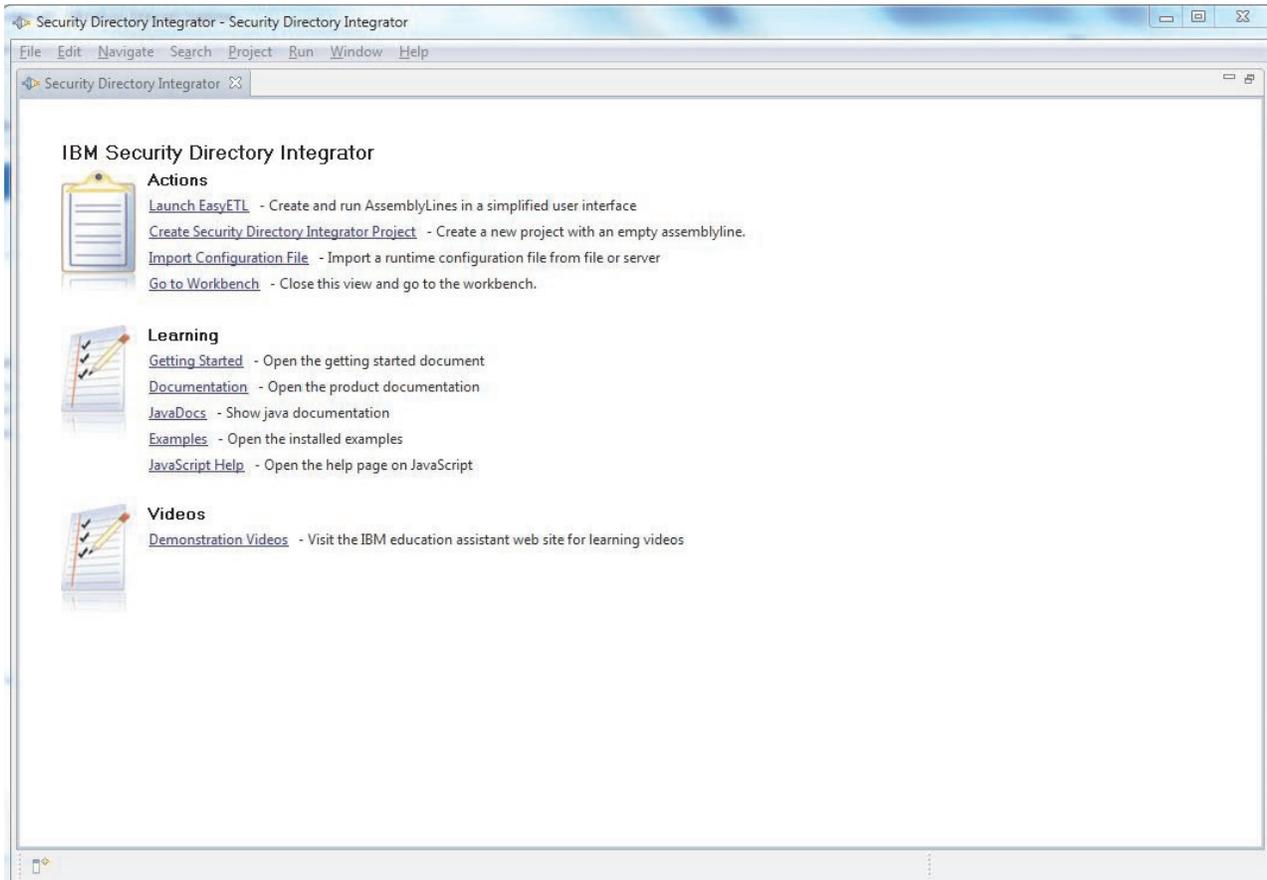


Abbildung 6. Eingangsanzeige des Konfigurationseditors

In der Eingangsanzeige werden eine Reihe von Links für den Schnelleinstieg bereitgestellt⁶.

Wenn Sie Integrationslösungen mit IBM Security Directory Integrator erstellen, testen oder modifizieren, arbeiten Sie innerhalb eines Projekts. Projekte sind Sammlungen von Fertigungslinien und ihren einzelnen Komponenten und jedes Projekt wird in einem eigenen zugehörigen Unterordner in Ihrem Arbeitsbereich angezeigt. Die Fertigungslinien und Komponenten, aus denen ein Projekt besteht, werden als einzelne Dateien gespeichert, die sich wiederum in Unterverzeichnissen des Projektordners befinden.

Wählen Sie auf dieser Seite den zweiten Link von oben⁷ (*Security Directory Integrator-Projekt erstellen*) aus, um Ihr erstes Projekt einzurichten. Sie müssen Ihrem neuen Projekt nun einen Namen geben. Nennen Sie es 'Tutorial' (Lernprogramm) und klicken Sie auf **Fertigstellen**.

6. Sie können jederzeit zu dieser Anzeige zurückkehren, indem Sie **Hilfe > Willkommen!** im Hauptmenü auswählen.

7. Der oberste Link 'Launch EasyETL' öffnet eine vereinfachte Workbench und wird im Anhang „EasyETL-Leitfaden“, auf Seite 91 beschrieben.

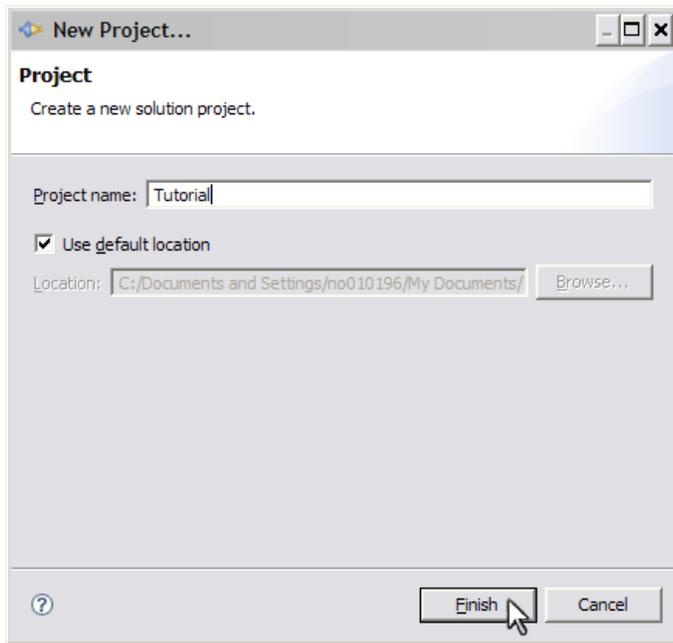


Abbildung 7. Neues Projekt benennen

Nun wird der Hauptarbeitsbereich für die Entwicklung angezeigt. Die Größe aller dieser Anzeigen kann verändert werden und Sie können den Aufbau des Bildschirms selbst bestimmen. Hier wird die IBM Security Directory Integrator-Standardperspektive angezeigt⁸.

8. Eine *Perspektive* ist einfach eine Organisation der Entwicklungsumgebungsanzeigen. Wenn Sie Änderungen am Layout vorgenommen haben und zur IBM Security Directory Integrator-Standardperspektive zurückkehren wollen, klicken Sie einfach auf **Fenster** im Menü ganz oben und wählen Sie die Option zum Zurücksetzen der Perspektive aus.

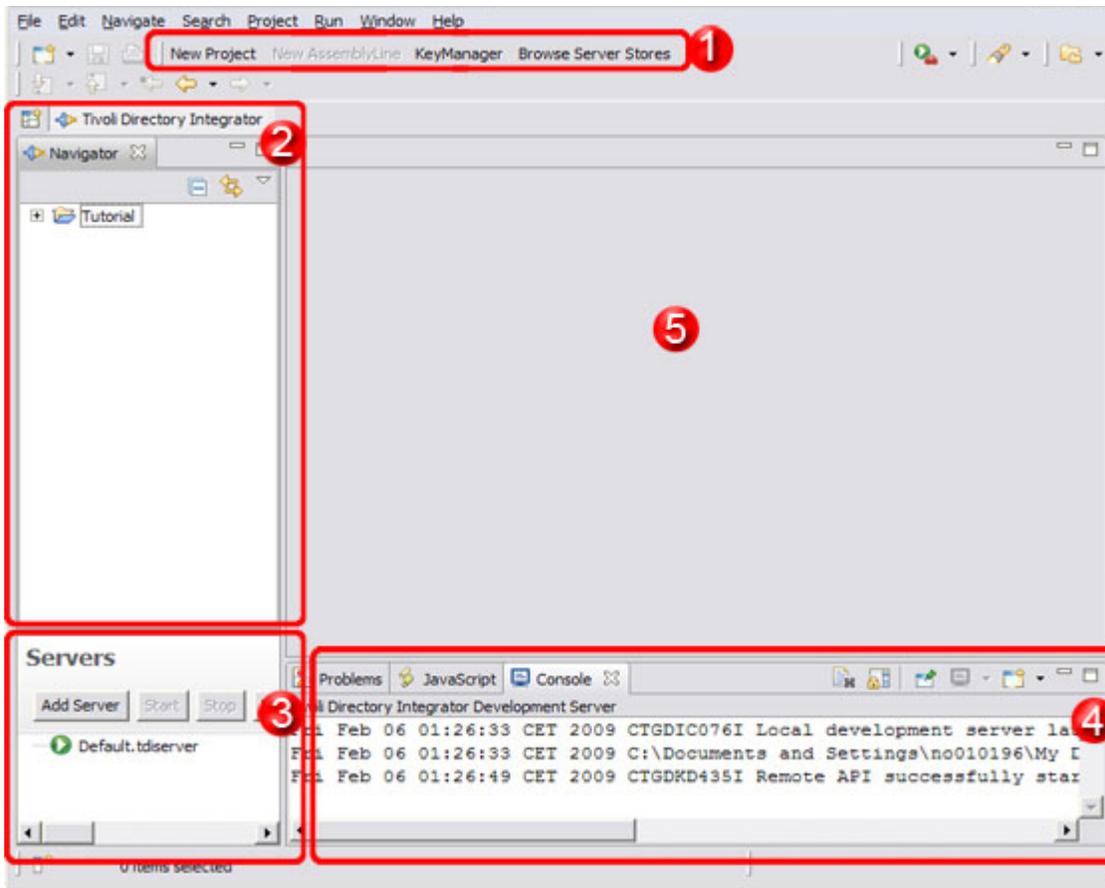


Abbildung 8. Hauptanzeige des Konfigurationseditors

Anmerkung: Die einzige Perspektive, die Sie verwenden müssen, ist **Security Directory Integrator** oder **Easy ETL**. Wenn der Konfigurationseditor nicht das erwartete Erscheinungsbild aufweist oder nicht die erwarteten Fensterbereiche enthält, führen Sie die folgenden Schritte aus:

- Wählen Sie **Fenster > Perspektive öffnen > Security Directory Integrator** aus. Mit dieser Option wird die Security Directory Integrator-Perspektive ausgewählt.
- Bewirkt dies keine Änderung, wählen Sie **Fenster > Perspektive zurücksetzen...** aus. Mit diesem Befehl wird sichergestellt, dass alle Standardfensterbereiche wie gewünscht positioniert sind.

Dies ist die Hauptanzeige, in der Sie sich bei der Arbeit mit IBM Security Directory Integrator am meisten bewegen werden. Sehen Sie sich zunächst die im oben stehenden Screenshot markierten und durchnummerierten Bereiche an,⁹ ohne die hier vorhandenen Navigationselemente im Einzelnen zu betrachten:

1. In der Mitte der Hauptzeile mit den Schaltflächen befindet sich eine Reihe von Direktaufufen zum Erstellen neuer Projekte, oder, falls ein Projekt im Navigator bereits ausgewählt ist, zum Erstellen neuer Fertigungslinien in diesem Projekt. Es ist auch eine Schaltfläche vorhanden, mit der das Schlüsselmanagertool für die Arbeit mit Zertifikatsschlüsselspeichern und -truststores gestartet werden kann, sowie die Schaltfläche **Serverspeicher durchsuchen**, mit der ver-

9. Wie bei den meisten Eclipse-basierten Anwendungen gibt es jeweils verschiedene Möglichkeiten, um ein und dieselbe Operation auszuführen. Im *Directory Integrator - Konfiguration* werden die verschiedenen Optionen und Anzeigen beschrieben, die zur Verfügung stehen.

schiedene Eigenschafteneinstellungen von dem IBM Security Directory Integrator-Server abgerufen werden, der diesem Projekt zugeordnet ist.

2. Dies ist die Anzeige *Navigator*, die eine Baumstrukturansicht Ihrer Entwicklungsressourcen bietet. Ihr neues Projekt 'Tutorial1' sollte hier angezeigt werden.
3. Die Anzeige 'Server' zeigt den Status aller konfigurierten Server an. Das Pfeilsymbol neben 'Default.tdiserver' zeigt an, dass dieser Server für Sie gestartet wurde. Diese Anzeige bietet auch Schaltflächen zum Definieren neuer Server, zum Starten und Stoppen Ihrer Server sowie zum Aktualisieren der Liste und Anzeigen eines Serverprotokolls.¹⁰

Beachten Sie, dass beim Starten einer Fertigungslinie sowohl die Konfigurationsinstanz¹¹ als auch die Fertigungslinie in dieser Anzeige angezeigt werden.

4. Sie sehen hier eine Reihe von Registerkarten, wobei die zurzeit ausgewählte Registerkarte die Konsolenausgabe Ihres Servers anzeigt. Sie können den hier angezeigten Nachrichten entnehmen, dass Ihr Server aktiv ist und dass die zugehörige Anwendungsprogrammierschnittstelle initialisiert wurde und für die Verwendung bereit ist.
5. Fertigungslinien und Komponenten, die Sie erstellen und öffnen, werden in dem grauen Bereich dieses Screenshots als Anzeigen des *Editors* angezeigt. Für jeden Ressourcentyp (Connector, Parser, Fertigungslinie usw.) steht ein eigener, speziell für diesen Typ bestimmter Editor zur Verfügung.

Erste Fertigungslinie erstellen

Indem Sie zu dem in der Einführung dargestellten Beispielszenario zurückkehren, erstellen Sie nun eine Fertigungslinie, die Informationen von D1 auf D3 migriert, wobei zunächst das Einfügen von Daten aus D2 ignoriert wird.



Abbildung 9. Diagramm eines vereinfachten Szenarios mit nur zwei Datenquellen

Der Ordner 'Tutorials' (den Sie aus dem Verzeichnis *SDI-Installationsverzeichnis/examples* in das Lösungsverzeichnis kopiert haben) enthält die Datei *People.csv*:

```
First;Last;Title  
Bill;Sanderman;Chief Scientist  
Mick;Kamerun;CEO  
Jill;Vox;CTO
```

10. Sollte Ihr Server nicht ordnungsgemäß gestartet worden sein, öffnen Sie 'TDI-Server' und klicken Sie doppelt auf 'Default.tdiserver'. Hierdurch wird das zugeordnete Serverdokument geöffnet. Stellen Sie sicher, dass die Installations- und die Lösungsverzeichniseinstellungen korrekt sind und klicken Sie auf die Option **Lösungsverzeichnis erstellen**, die sich in dieser Anzeige oben befindet. Wenn das Problem dadurch nicht behoben wird, müssen Sie Unterstützung anfordern.

11. Wenn der IBM Security Directory Integrator-Server eine Konfiguration lädt, erstellt er eine *Konfigurationsinstanz*, die die Fertigungslinien dieses Projekts kapselt und ermöglicht, dass diese in einer eigenen enthaltenen Umgebung ausgeführt werden. Dies bedeutet, dass Sie dieselbe Konfiguration mehrere Male auf demselben Server laden können. Dieses Vorgehen führt zu voneinander getrennten Konfigurationsinstanzen, die alle denselben Satz von Fertigungslinien enthalten, ohne dass diese sich gegenseitig behindern.

Roger
Gregory;Highpeak;VP Product Development
Ernie;Hazzle;Chief Evangelist
Peter;Belamy;Business Support Manager

An der oben stehenden Auflistung sehen Sie, dass das CSV-Format (*character separated value*) verwendet wurde. Diese Datei stellt die D1-Eingabedatenquelle dar. Ihre Fertigungslinie extrahiert diese Daten und überträgt sie in ein XML-Dokument, das Ihr D3-Ausgabeziel sein wird.

Klicken Sie auf **Neue Fertigungslinie** in der Symbolleiste ganz oben und nennen Sie die neue Fertigungslinie 'CSV2XML'.



Abbildung 10. Dialogfenster 'Neue Fertigungslinie'

Klicken Sie nun auf die Schaltfläche **Fertigstellen**, um die Fertigungslinie auf einer Registerkarte des Fertigungslinieneditors zu öffnen.

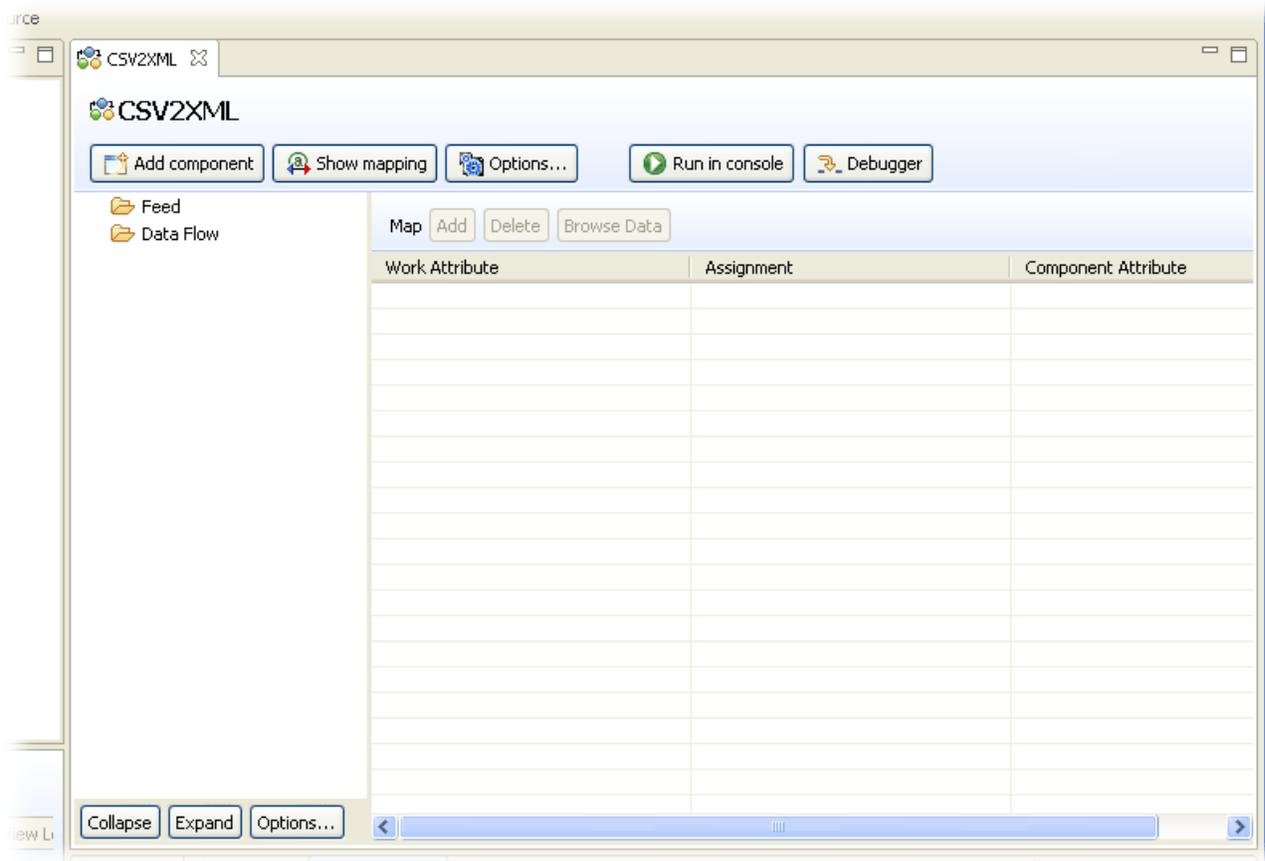


Abbildung 11. Leerer Fertigungslinieneditor

Im linken Teil des Fertigungslinieneditors ist die Liste der Komponenten enthalten, aus denen diese Fertigungslinie besteht. Zurzeit ist er bis auf die Abschnittsnamen *Zuführung* (Feed) und *Datenfluss* (Data Flow) leer. Im rechten Bereich werden alle Attribute angezeigt, die bei der Eingabe- und Ausgabezuordnung der Fertigungslinie verwendet werden.

Betrachten Sie zum Verständnis dieses Fertigungslinienabschnitts die Aufgabe, die diese Fertigungslinie ausführen soll: *Für jede Zeile in der CSV-Datei einen neuen Knoten im XML-Dokument erstellen*. Dieses Schleifenverhalten wird automatisch durch den IBM Security Directory Integrator-Kernel bereitgestellt, der die im Fertigungslinienabschnitt *Datenfluss* aufgelisteten Komponenten ausführt, solange Eingabedaten vorhanden sind, die von Connector im Abschnitt *Zuführung* stammen¹²

Diese Funktion benutzen Sie nun, indem Sie dem Abschnitt 'Zuführung' (Feed) einen Connector hinzuzufügen, der die genannte CSV-Eingabedatei einliest. Führen Sie dies aus, indem Sie mit der rechten Maustaste auf den Abschnittsordner *Zuführung* (Feed) klicken und **Komponente hinzufügen...** auswählen.

12. Beachten Sie, dass jeweils nur ein Connector im Abschnitt *Zuführung* (Feed) Daten an die Fertigungslinie übermittelt. Wenn Sie hier mehr als einen Connector im Iteratormodus platzieren, wird zunächst der Connector ganz oben geleert, bevor der nächste Connector in der Reihenfolge beginnt, aus der zugehörigen Quelle zu lesen.

Dieses Dialogfenster bietet eine Reihe von Optionen zum Suchen und Auswählen der gewünschten Komponente:

1. Geben Sie zunächst einen Teil des Namens der Komponente in das Textfeld ein. Daraufhin wird die Auswahlliste auf der rechten Seite entsprechend gefiltert. Geben Sie in diesem Beispiel "Datei" ein.
2. Sie können die Auswahlliste optional einschränken, sodass nur ein einziger Komponententyp eingeschlossen wird - z. B. nur Connector, Parser, Scripts usw.
3. Suchen Sie in dieser Liste nach der gewünschten Komponente und wählen Sie sie aus. In unserem Beispiel ist dies 'Dateiconnector'.

Der neue Connector erhält automatisch den Namen 'Dateisystemconnector'. Ändern Sie den Namen in 'Read_CSV_File', damit der Name im Kontext Ihrer Lösung eine größere Bedeutung hat¹³ und wählen Sie anschließend die Option **Iterator** in der Dropdown-Liste **Modus** aus.

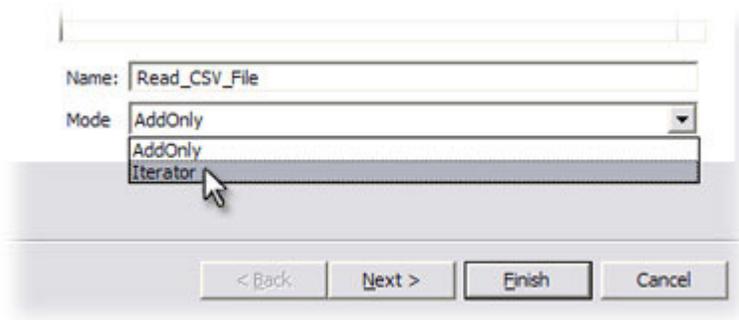


Abbildung 14. Connector umbenennen und zugehörigen Modus ändern

Durch die Moduseinstellung eines Connector wird die integrierte Ausführungslogik der Fertigungslinie darüber informiert, welche Rolle diese Komponente im Datenfluss spielt. Der Iteratormodus bewirkt das Verhalten *FOR-EACH*, das erforderlich ist, um die Daten aus der CSV-Datei nacheinander an die Komponenten zu liefern, die Sie dem Abschnitt *Datenfluss* hinzufügen.

Klicken Sie nun auf die Schaltfläche **Weiter >**, um in der Konfigurationsanzeige für den ausgewählten Connector fortzufahren.

13. Sie können Connector mit frei gewählten Namen bezeichnen. Es empfiehlt sich jedoch, dass Sie einen Namen wählen, mit dem Sie auch eine Scriptvariable bezeichnen würden: Beginnen Sie mit einem Buchstaben, auf den eine beliebige Anzahl von Buchstaben, Ziffern und Unterstrichszeichen folgt. Der Grund dafür ist, dass alle Fertigungslinienkomponenten automatisch als Scriptvariablen registriert werden, wodurch es später einfacher ist, neu zu konfigurieren und sie von Ihrem Scriptcode aus auszuführen.

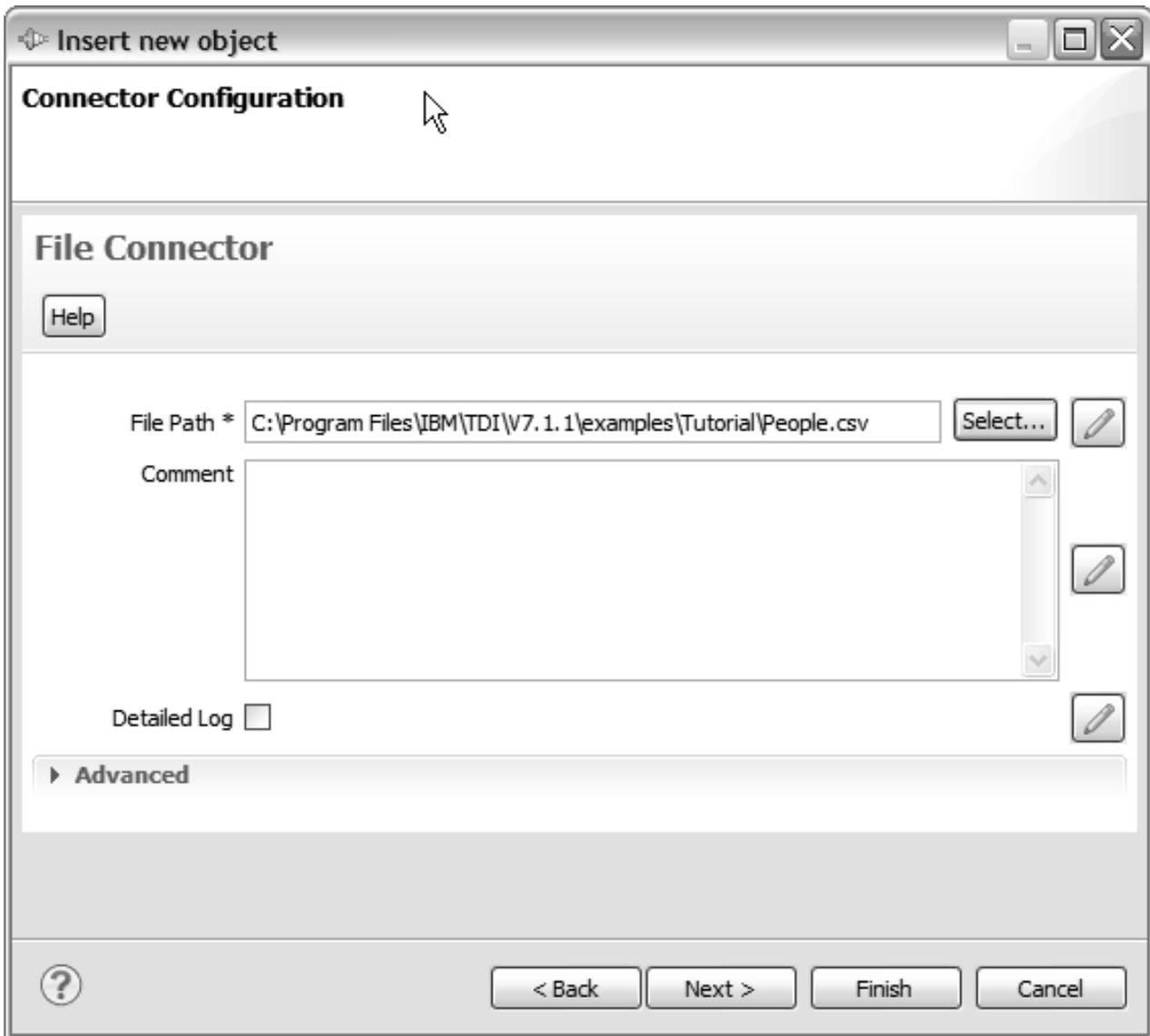


Abbildung 15. Konfigurationsanzeige für den Dateiconnector

Jede Komponente stellt einen eigenen zugehörigen Satz von Konfigurationsparametern bereit. Die nun angezeigten Parameter gehören zum Dateiconnector, der lediglich einen erforderlichen Parameter aufweist: **Dateipfad**. Geben Sie den Pfad zur Datei `People.csv` ein - entweder den vollständigen Pfad oder den relativen Pfad von Ihrem Lösungsverzeichnis aus wie im Screenshot oben angezeigt¹⁴ - oder klicken Sie auf die Schaltfläche **Auswählen**, um einen Dateibrowser aufzurufen, mit dem Sie nach dieser Datei suchen können.

Da eine formatierte Textdatei ein Bytestrom und keine strukturierte Datenquelle wie eine Datenbank oder ein Verzeichnis ist, müssen Sie einen Parser konfigurieren, um beim Lesen des Bytestroms das Format zu interpretieren. IBM Security Directory Integrator bietet eine leistungsfähige und vielseitige Datenbrowserfunktion zum interaktiven Testen Ihrer Connector-/Parserauswahl und -konfiguration.

14. Durch dieses Verfahren kann Ihre Lösung leichter versetzt und gemeinsam genutzt werden, da Sie lediglich das gewünschte Lösungsverzeichnis angeben müssen und alle relativen Pfade ohne Änderung funktionieren.

Bevor zu diesem Punkt übergegangen wird, müssen Sie zunächst diesen Assistenten beenden, indem Sie erneut auf **Weiter >** klicken und mit der **Parserkonfiguration** fortfahren. Hier klicken Sie auf **CSV-Parser**, um diesen auszuwählen.

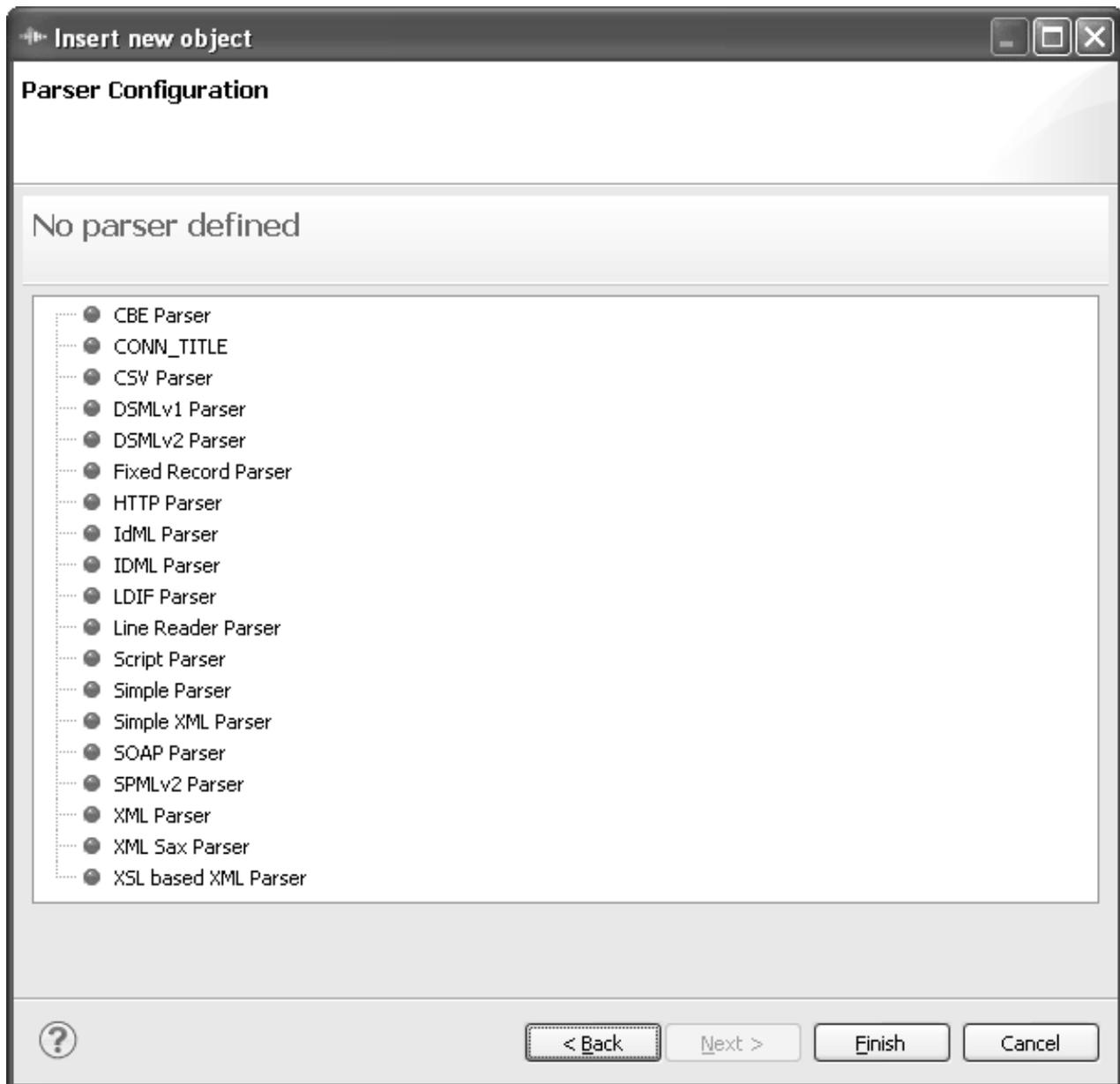


Abbildung 16. Parser unter 'Neues Objekt einfügen' auswählen

Sobald Sie den CSV-Parser ausgewählt haben, klicken Sie auf **Fertigstellen**, um den Assistenten zu schließen. Nun wird die Konfigurationsanzeige für den Parser angezeigt. Da die Standardeinstellungen nicht geändert werden müssen, klicken Sie einfach erneut auf 'Fertigstellen', um den Assistenten abzuschließen.

Im nächsten Schritt *ermittelt* Ihr Connector das Schema Ihrer Eingabequelle, um diese Werte Ihrer Fertigungslinie zuzuordnen. Hierbei ist der Einsatz des Datenbrowsers von Nutzen.¹⁵ Beginnen Sie mit dem Datenbrowser, indem Sie mit der rechten Maustaste auf Ihren neuen Connector im Iteratormodus in der Baumstruktur mit den Fertigungslinienkomponenten klicken und **Daten durchsuchen** im Kontextmenü auswählen.

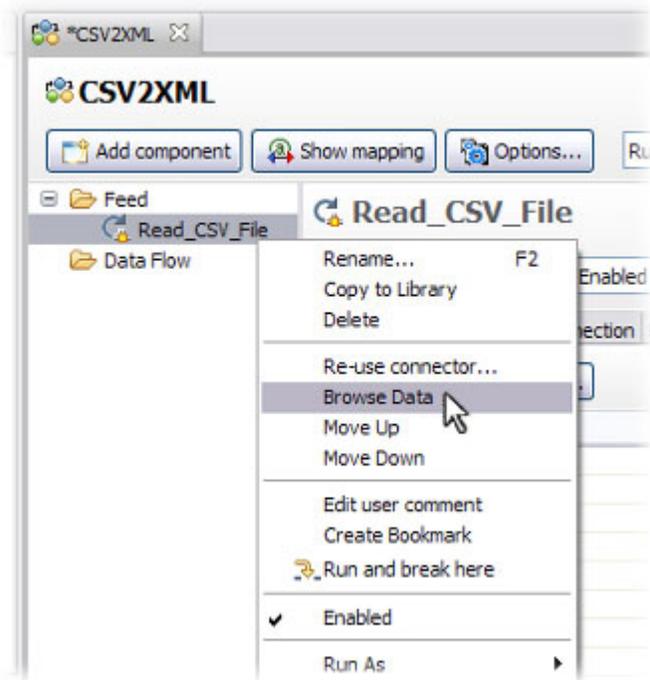


Abbildung 17. Kontextmenüauswahl 'Daten durchsuchen'

Dadurch wird der Datenbrowser als neue Editorregisterkarte geöffnet.

15. Da Sie wissen, dass die Datei das CSV-Format aufweist, ist die schnellste Methode, wenn Sie einfach auf die Schaltfläche **Verbindung herstellen** und auf **Nächster** im Schemabereich des Connectors im Iteratormodus klicken. Anschließend ziehen Sie die ermittelten Attribute wie gewünscht in die Eingabezuordnung. Der Datenbrowser ist dann von Nutzen, wenn Sie das Format nicht kennen. Probieren Sie es auf jeden Fall auch mit dem Datenbrowser.

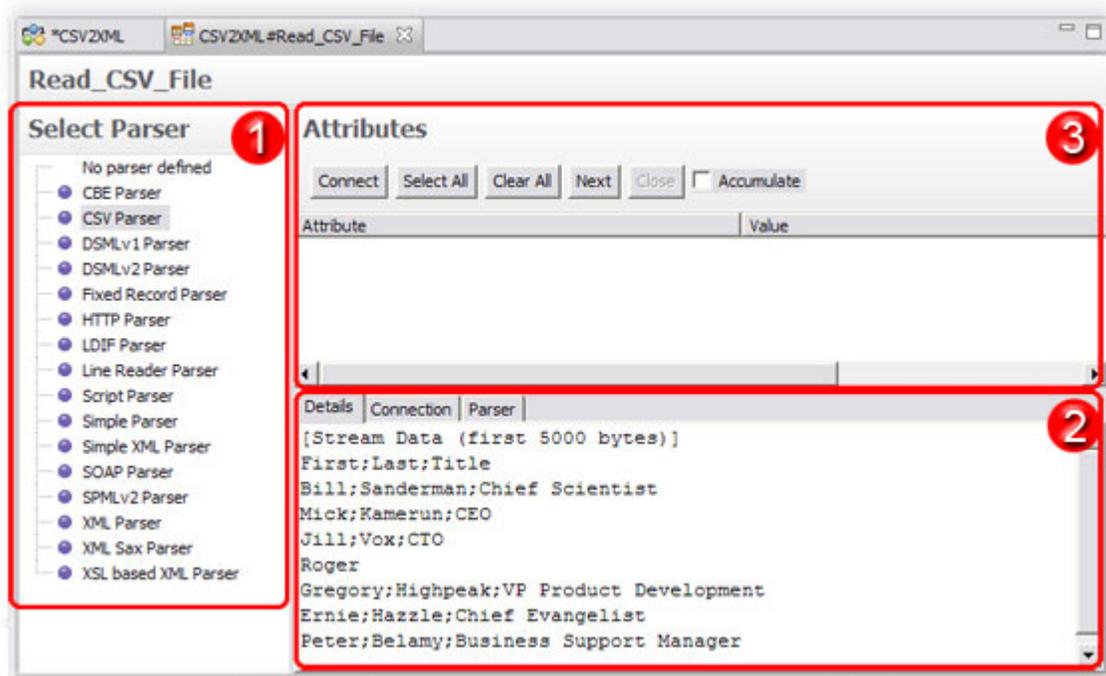


Abbildung 18. Der Datenbrowser

In dem mit '1' gekennzeichneten Bereich im oben stehenden Screenshot wählen Sie den gewünschten Parser aus bzw. ändern ihn. Der Bereich 2 bietet die Registerkarte 'Details', in dem der unformatierte Bytestrom angezeigt wird, der syntaktisch analysiert werden soll. Es ist auch eine Registerkarte vorhanden, auf der die Verbindungsparameter des Connector geändert werden, sowie eine Registerkarte zum Konfigurieren des ausgewählten Parsers.

Im letzten Abschnitt dieses Dialogfensters (#3 im Screenshot) wird die Verbindung zur Datenquelle hergestellt und ermittelt, welche Attribute zur Verfügung stehen. Um die Ermittlung jetzt auszuführen, klicken Sie zunächst auf die Schaltfläche **Verbindung herstellen** und anschließend auf die Schaltfläche **Nächster**.

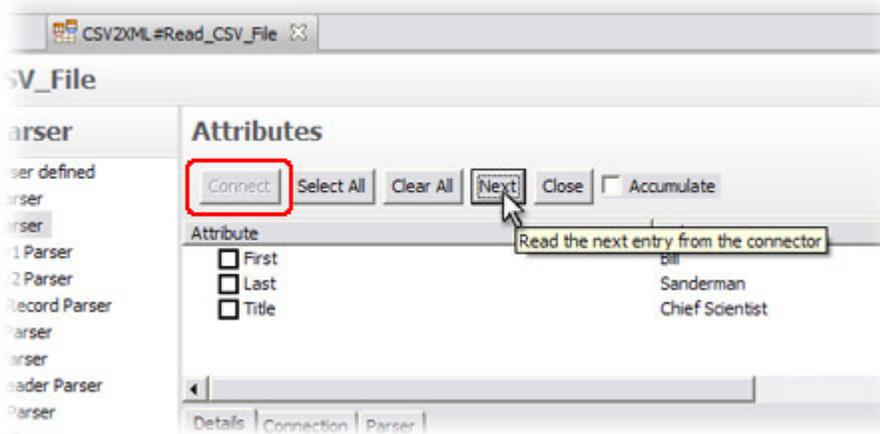


Abbildung 19. Schema interaktiv durch Durchsuchen von dynamischen Daten ermitteln

Sie haben nun das *Schema* dieser Datei ermittelt. Wählen Sie die Attribute aus, die Sie zuordnen wollen (in diesem Fall sind es alle), indem Sie das jeweilige Kontrollkästchen neben dem entsprechenden Attribut auswählen oder indem Sie die Schaltfläche **Alles auswählen** verwenden.

Verwenden Sie den Direktaufruf 'Strg-W', um die Registerkarte für den Datenbrowser zu schließen, oder klicken Sie einfach auf das Symbol zum Schließen (X) am rechten Rand des Registerkartenreiters. Kehren Sie zum Fertigungslinieneditor zurück, in dem Ihre Fertigungslinie so angezeigt werden sollte wie im unten stehenden Screenshot.¹⁶

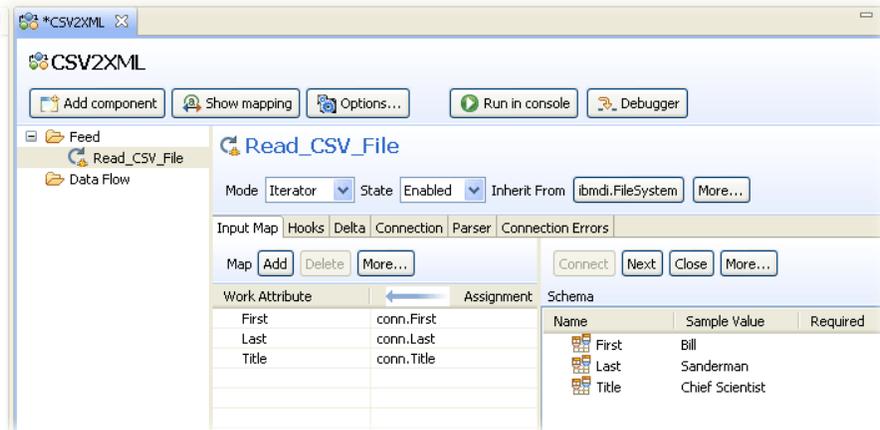


Abbildung 20. Fertigungslinie, für die der Connector im Iteratormodus ausgewählt ist

Details zur ausgewählten Komponente werden auf der rechten Seite der Komponentenliste der Fertigungslinie angezeigt, einschließlich der drei Zuordnungsregeln, die Sie gerade in der Eingabezuordnung festgelegt haben. Jedes Element der Attributzuordnung weist eine **Zuordnung** auf. Dies ist ein Scriptfragment, das bewertet wird, um den Wert (oder die Werte) des Zielattributs festzulegen.

Sehen Sie sich diese *Zuordnungen* näher an, bevor Sie fortfahren: Sie wissen aus dem Abschnitt „Datenmodell 'Eintrag-Attribut-Wert'“ auf Seite 5, dass die Fertigungslinie über einen global verfügbaren *Work-Eintrag* verfügt, der alle Daten überträgt, die durch die Fertigungslinie transportiert werden. Auf dieses Objekt wird mithilfe der im Voraus registrierten Scriptvariable *work* in Scriptcode verwiesen. Zusätzlich besitzen die Schnittstellen der Connector jeweils einen eigenen zugehörigen *Conn-Eintrag*, der als Cache für Lese- und Schreiboperationen verwendet wird. Auf dieses komponentenspezifische Objekt wird vom Script aus über die vorab registrierte Variable *conn* zugegriffen.¹⁷ Betrachten Sie zum besseren Verständnis die erste Zuordnungsregel. Sie erstellt im Work-Eintrag ein Attribut mit dem Namen 'First' (Erster). Der zugehörige Wert ist aus der folgenden Zuordnung abgeleitet:

```
conn.First
```

16. Sollte sich Ihr Connector im Abschnitt *Datenfluss* befinden, ziehen Sie ihn einfach nach oben in den Abschnitt *Zuführung*. Sollte der Modus 'Iterator' nicht eingestellt sein, klicken Sie mit der rechten Maustaste auf den Connector, wählen Sie **Modus** aus und wählen Sie anschließend **Iterator** aus.

17. Die Variable *conn* ist nur für begrenzte Zeiträume verwendbar, wie in den 'Hook Flow Diagrams' von IBM Security Directory Integrator dargestellt. Außerhalb dieses Bereichs kann auch auf die Variable zugegriffen werden, indem eine Komponente nach dem zugehörigen Conn-Eintrag abgefragt wird.

Diese abgekürzte Schreibweise verweist auf das Attribut mit dem Namen 'First', das gerade in den Eintrag conn eingelesen wurde, und die zugehörigen Werte werden verwendet, um das neue Work-Eintrag-Attribut zu füllen. Ein vergleichbares Zuordnungsscript wäre:

```
return conn.getAttribute("First");18
```

Zurück zur Übung: Sie müssen nun Ihren Ausgabeconnector hinzufügen, um das XML-Zieldokument (Datenquelle D2) zu erstellen. Verwenden Sie dieses Mal die Schaltfläche **Komponente hinzufügen**, die sich oben in der Fertigungslinienkomponentenanzeige befindet.

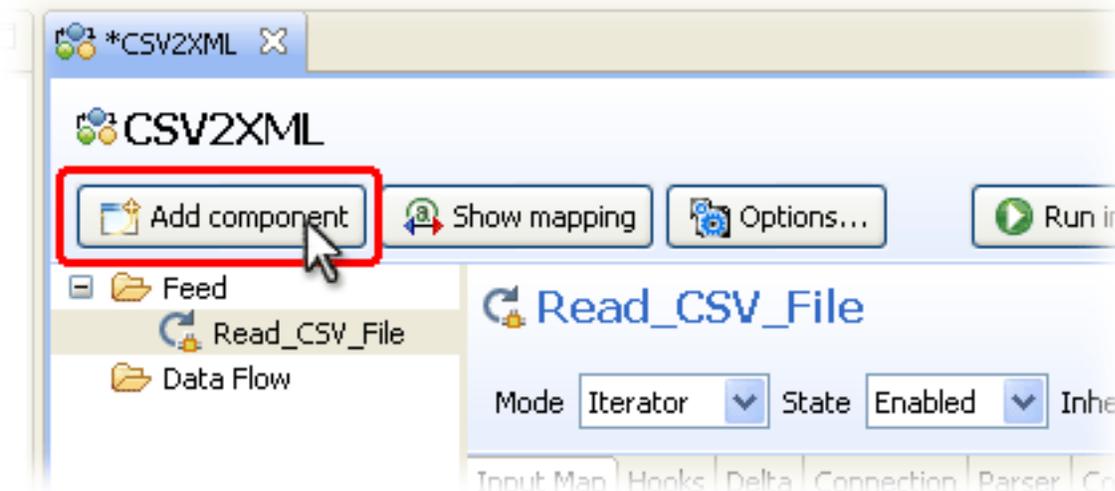


Abbildung 21. Schaltfläche 'Komponente hinzufügen'

Wählen Sie erneut den Dateiconnector aus und benennen Sie ihn in 'Write_XML_File' um. Behalten Sie als Modus die Einstellung **AddOnly** bei und klicken Sie anschließend auf **Weiter >**.

Legen Sie in der Konfigurationsanzeige für den Connector den Parameter **Dateipfad** fest, um in die Datei mit dem Namen `Output.xml` zu schreiben, die sich im Lernprogrammordner befindet. Wählen Sie anschließend in der nächsten Assistentenanzeige 'XML-Parser' aus. Sie können nun auf **Fertigstellen** klicken, da Sie die XML-Parserkonfiguration nicht ändern müssen. Beachten Sie, dass Sie im Fall Ihres Ausgabeconnectors keine Schemaerkennung verwenden können, da keine Datei `Output.xml` vorhanden ist, um die Erkennung durchzuführen.

18. Für Benutzer, die mit Version 6.x und früher vertraut sind: Sie können auch die Syntax vor Version 7.0 verwenden:

```
ret.value = conn.getAttribute("First")
```

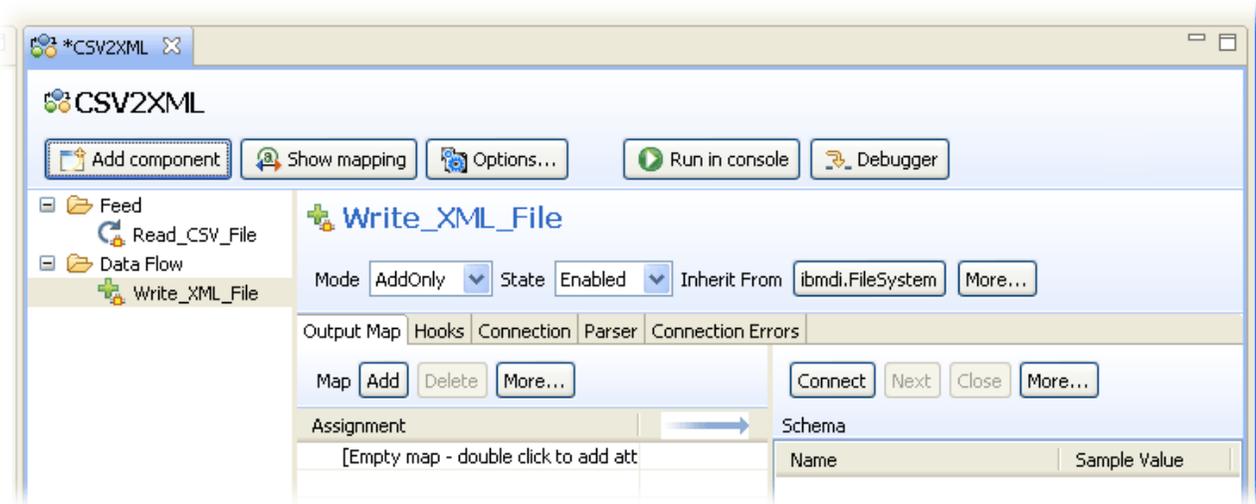


Abbildung 22. Fertigungslinie, für die zwei Connector ausgewählt sind

Möglicherweise haben Sie beim Auswählen einer Komponente bemerkt, dass die zugehörigen Details auf der rechten Seite der Editoranzeige angezeigt werden. Wenn Sie den Ordner 'Zuführung' oder den Ordner 'Datenfluss' auswählen, wird Ihnen eine Übersicht aller **Attributzuordnungen** für diese Fertigungslinie angezeigt. Diese Anzeige eignet sich gut, um Ihre Eingabeattribute in die Ausgabezuordnung Ihres letzten Connectors zu kopieren. Rufen Sie diese Anzeige jetzt durch Klicken auf 'Zuführung' oder 'Datenfluss' auf.

Sie sehen hier eine Liste der Attribute (insgesamt drei), die vom Connector im Iteratormodus in Ihre Fertigungslinie übertragen werden. Wählen Sie diese Eingabezuordnungsattribute aus¹⁹ und ziehen Sie sie herunter in die Ausgabezuordnung des Connectors 'Write_XML_File', wodurch der Datenfluss abgeschlossen wird.

19. Mit der Steuerungstaste und Klicken können Sie mehrere auswählen oder Sie können mit der Umschalttaste und Klicken einen Bereich auswählen.

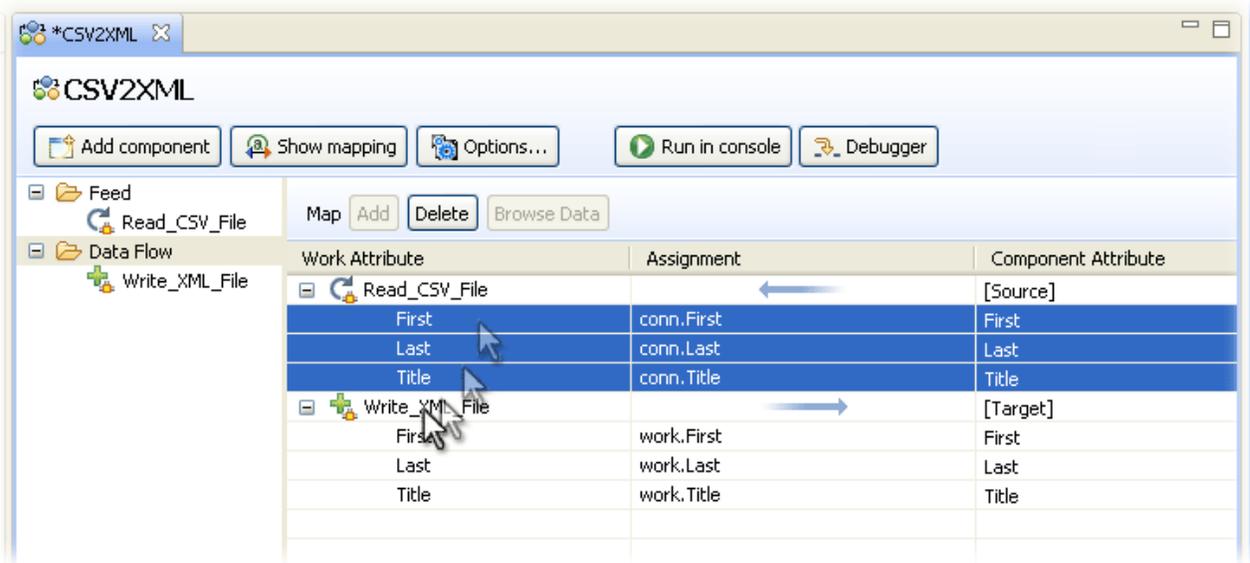


Abbildung 23. Attribute in die Ausgabezuordnung ziehen

Beachten Sie, wie die Zuordnung automatisch aus dem Eingabe- in das Ausgabeformat konvertiert wird. Beispielsweise erstellt das erste Zuordnungselement in der Eingabezuordnung Ihres Connectors 'Read_CSV_File' ein Attribut im Work-Eintrag mit der Bezeichnung 'First', das alle in conn.First (d. h., in dem Attribut mit der Bezeichnung 'First', das in den Conn-Eintrag eingelesen wurde) gefundenen Werte enthält. Wenn Sie diese Eingabezuordnungsregel in eine Ausgabezuordnungsregel ziehen, wird die zugehörige Zuordnung geändert, sodass der Wert nun aus dem Work-Eintrag stammt. Außerdem erstellt die Zuordnungsregel ein Zielattribut im Cache des Connectors (den Conn-Eintrag).

Wenn Sie die Quelle für eine Zuordnungsregel ändern wollen, müssen Sie die Zuordnung bearbeiten. Wenn Sie den Namen des Attributs ändern wollen, das zugeordnet wird, klicken Sie einfach mit der rechten Maustaste und benennen Sie es um. Führen Sie diese Schritte jetzt für die ersten beiden Ausgabezuordnungsregeln aus.

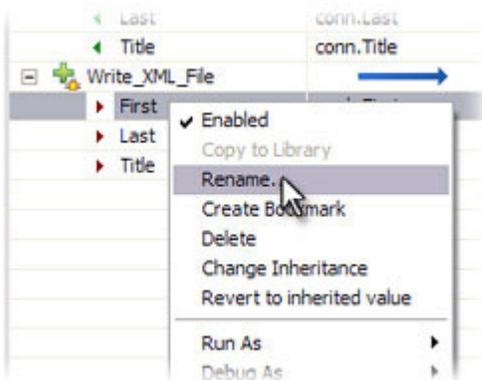


Abbildung 24. Attributzuordnungsregel umbenennen

Ändern Sie 'First' in 'FirstName' und 'Last' in 'LastName'.

Fügen Sie dieser Ausgabezuordnung nun ein neues Zuordnungselement hinzu, indem Sie mit der rechten Maustaste auf die Ausgabezuordnung 'Write_XML_File' selbst klicken und **Attribut hinzufügen** auswählen.

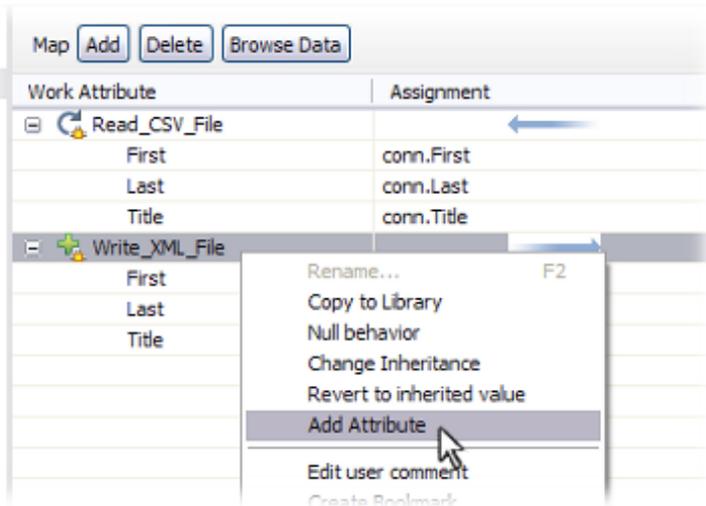


Abbildung 25. Attribut 'FullName' der Ausgabezuordnung hinzufügen

Nennen Sie das Ziel dieser neuen Zuordnungsregel 'FullName', klicken Sie auf **OK** und rufen Sie dann durch Doppelklicken die zugehörige Zuordnung zur Bearbeitung auf. Daraufhin wird die Scripteditoranzeige geöffnet und ein Standardzuordnungsscript wird angezeigt: `work.FullName`. Natürlich ist im Work-Eintrag kein Attribut 'FullName' enthalten, diese Zuordnung kann also keine Werte festlegen. Sie müssen diesen Wert stattdessen *berechnen*, indem Sie das Script so ändern, dass es die Attribute 'First' und 'Last' verknüpft, wobei zwischen diesen Werten ein einzelnes Leerzeichen stehen bleibt:

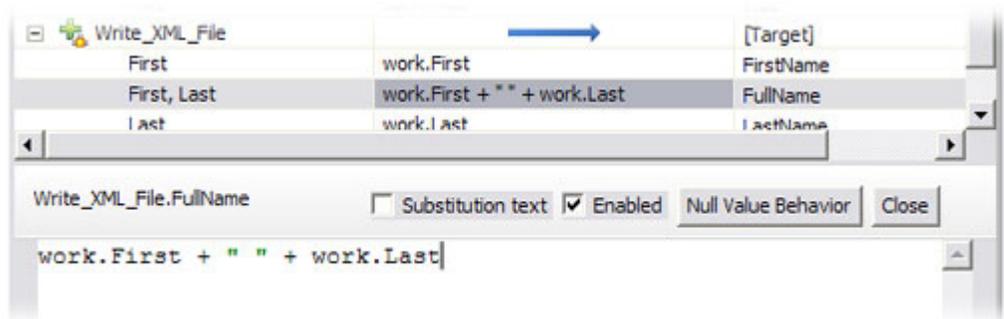


Abbildung 26. Zuordnung bearbeiten

Das Script sollte wie folgt aussehen:

```
work.First + " " + work.Last
```

Beachten Sie, dass bei einzeiligen Zuordnungsscripts für die Attributzuordnung wie dem vorliegenden kein abschließendes Semikolon erforderlich ist.²⁰

20. Die Syntax vor Version 7.0 wird auch unterstützt. Attributzuordnungsscripts können also nach wie vor mit `"ret.value ="` beginnen.

Klicken Sie nach dem Abschließen aller erforderlichen Arbeitsschritte auf die Schaltfläche **Schließen**, die sich in der Scripteditoranzeige auf der rechten Seite oben befindet.

Fertigungslinie ausführen

Ihre Fertigungslinie muss nun getestet werden.

Klicken Sie dazu auf die Schaltfläche **Ausführen**, die sich oben im Fertigungslinieneditor befindet.

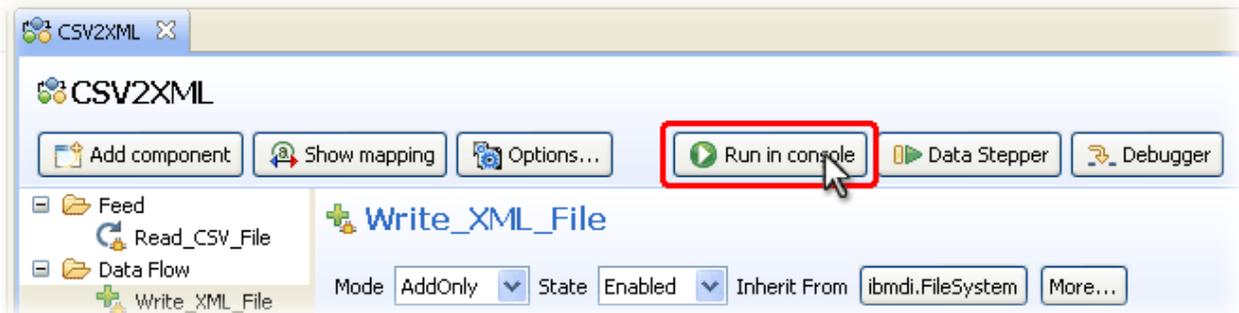


Abbildung 27. Die Schaltfläche 'Ausführen'

Nun wird eine neue Registerkarte mit einem *Ausführungsfenster* geöffnet, das die aus Ihrer Fertigungslinie stammende Protokollausgabe anzeigt.

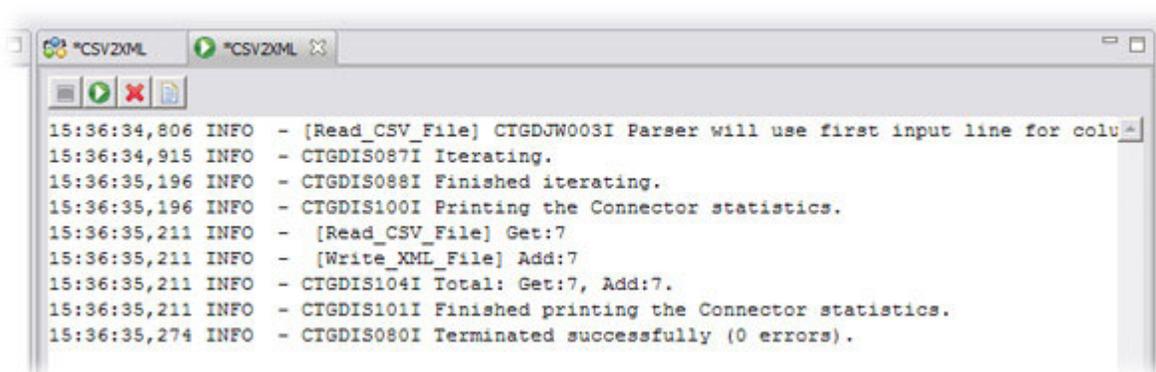


Abbildung 28. Protokollausgabe der Fertigungslinienausführung

Tatsächlich hat der Konfigurationseditor auf Grundlage Ihrer Fertigungslinie mit allen zugehörigen Komponenten eine *Konfiguration* (Config) exportiert, d. h. ein XML-Dokument, das die einem IBM Security Directory Integrator-Server zugeordnete Arbeit definiert. Er hat dann diese Konfiguration über eine Pipe an den Server übertragen und ihn angewiesen, Ihre Fertigungslinie auszuführen und alle Protokollausgaben für die Anzeige auf dem Bildschirm zu erfassen.

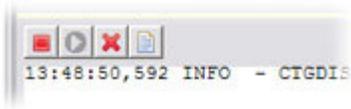


Abbildung 29. Schaltflächenleiste für das Protokollausgabefenster

Das Ausführungsfenster bietet auch eine Schaltflächenleiste mit Optionen zum Stoppen der Fertigungslinie, zum erneuten Starten, falls sie gestoppt wurde, und zum Löschen des Protokollinhalts. Die Schaltfläche ganz rechts öffnet die aktuelle Protokollausgabe in einem externen Editor.

Die Protokollausgabe einer Fertigungslinie endet mit statistischen Daten zu allen beteiligten Komponenten; in Ihrem Fall handelt es sich um lediglich zwei Connector. Aus den oben stehenden Informationen wird deutlich, dass aus der CSV-Datei sieben Einträge gelesen wurden und dass sieben Knoten in das XML-Dokument geschrieben wurden.

Sie können Ihre Ausgabedatei auf der Platte lokalisieren und sie in einem Browserfenster öffnen, um Ihre Ergebnisse zu bestätigen. Sie können auch den Datenbrowser verwenden, indem Sie mit der rechten Maustaste auf Ihren Ausgabeconnector (Write_XML_File) klicken und **Daten durchsuchen** auswählen.

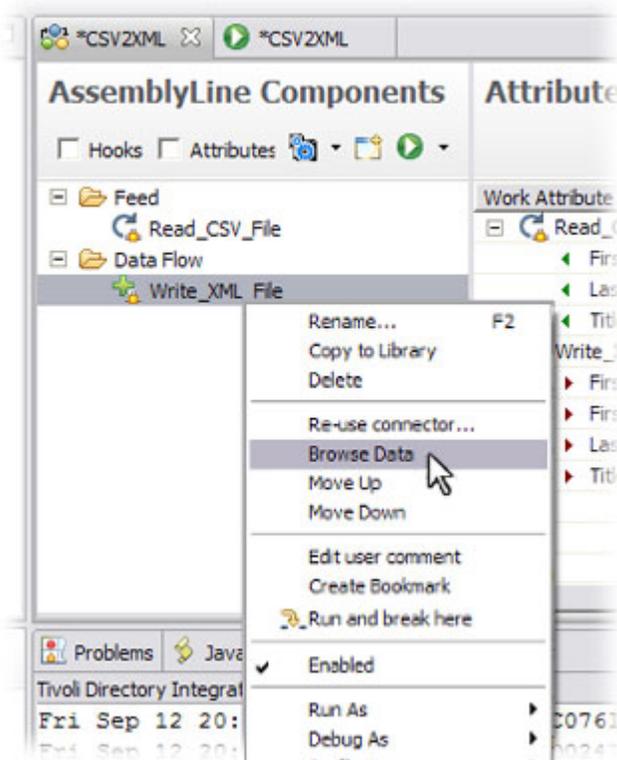


Abbildung 30. Daten durchsuchen, die von einem Ausgabeconnector erstellt wurden

Dadurch wird der Datenbrowser für den ausgewählten Connector aufgerufen, der konfiguriert und für die Verwendung bereit ist. Klicken Sie auf die Schaltfläche **Verbindung herstellen** und auf **Nächster**, um die Ausgabedaten zu lesen und anzuzeigen.

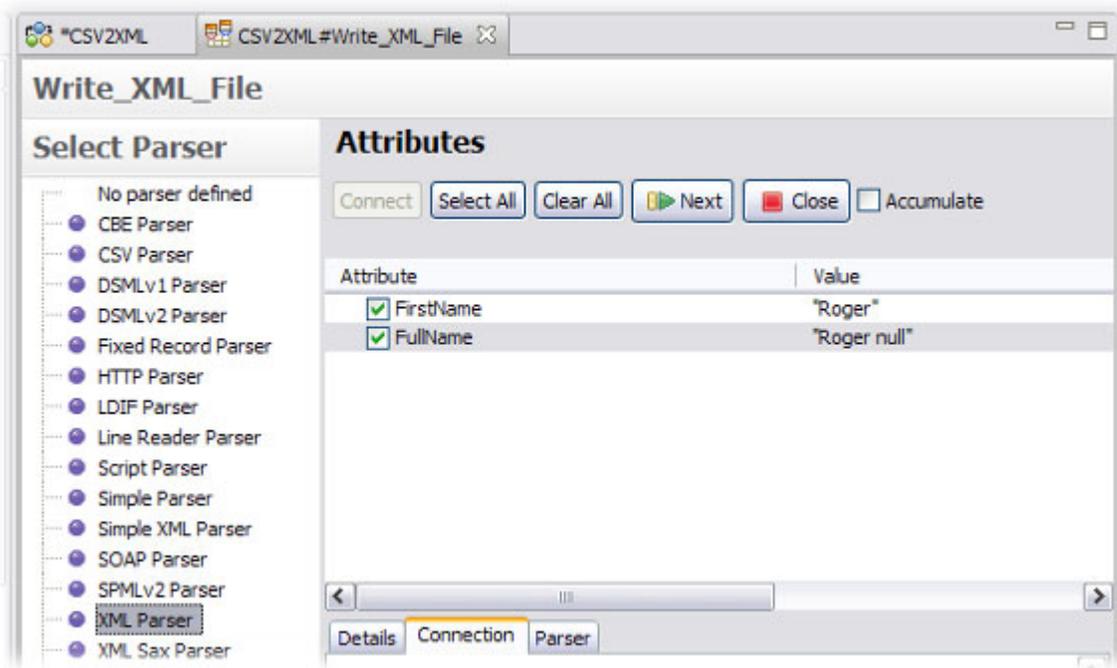


Abbildung 31. XML-Ergebnis durchsuchen

Die XML-Ausgabe sollte eine genaue Darstellung der Eingabewerte sein, plus Ihrer Zuordnungslogik. Das Ergebnis ist bis auf den vierten Eintrag (Roger) zufriedenstellend. Beim dritten Eintrag fehlen 'LastName' und 'Title' und es wurde der folgende Wert für 'FullName' berechnet: Roger null.

Wenn Sie die Eingabedaten näher untersuchen, stellen Sie fest, dass eine CSV-Zeile unvollständig ist:

```

First;Last;Title
Bill;Sanderman;Chief Scientist
Mick;Kamerun;CEO
Jill;Vox;CTO
Roger
Gregory;Highpeak;VP Product Development
Ernie;Hazzle;Chief Evangelist
Peter;Belamy;Business Support Manager

```

Fehlende und ungültige Eingabedaten sind ein verbreitetes Phänomen. Sie müssen Ihre Lösung so vorbereiten, dass Sie diese Fälle herausfiltert oder während der Verarbeitung korrigiert.

Nullverhalten: Umgang mit fehlenden Attributen/Werten

Bei fehlenden Werten können Sie entweder die integrierte Nullverhaltenfunktion der Attributzuordnungen verwenden oder Sie können sie selbst erkennen und bearbeiten.

Zunächst muss das Nullverhalten konfiguriert werden. Klicken Sie dazu mit der rechten Maustaste auf den Connector 'Read_CSV_File' in der Attributzuordnungsanzeige (nicht in der Baumstrukturansicht für Fertigungslinienkomponenten) und wählen Sie **Nullverhalten** im Kontextmenü aus.²¹

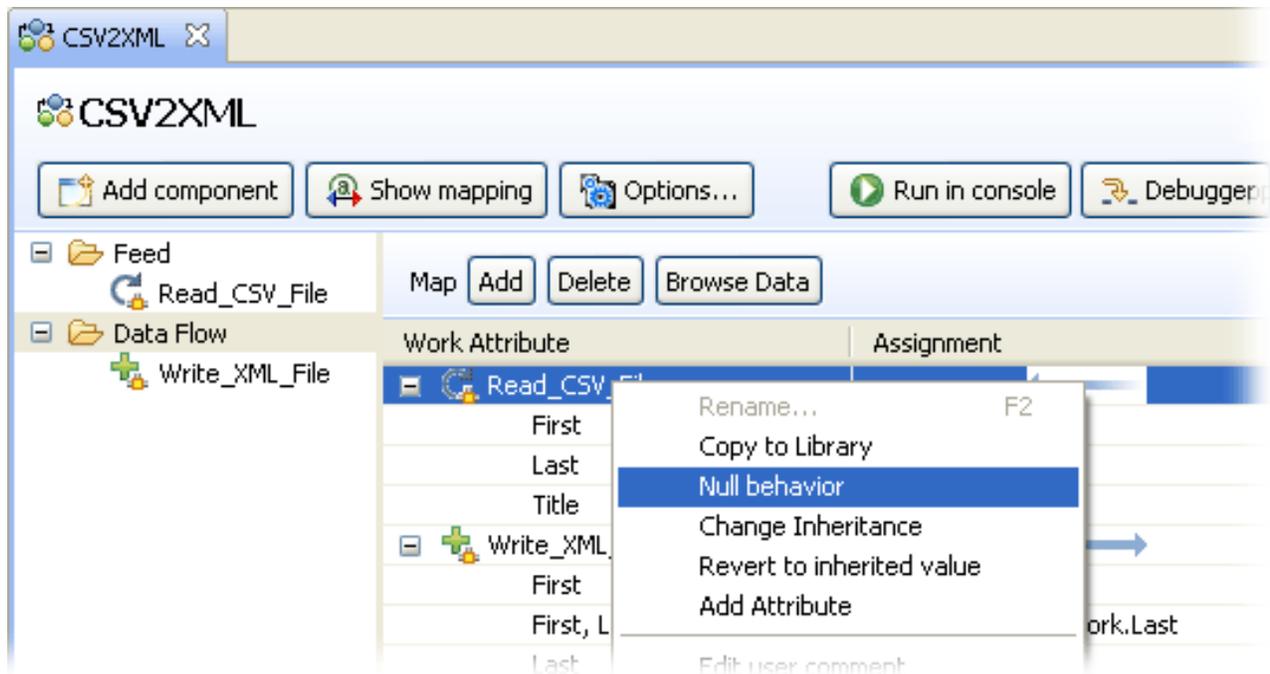


Abbildung 32. Option 'Nullverhalten' für Konfiguration auf Fertigungslinienebene

Hierdurch wird das Dialogfenster 'Nullverhalten' aufgerufen, in dem Sie konfigurieren können, wie *NULL* definiert wird - dies kann je nach Typ der zu lesenden Quelle variieren - und wie diese Situation verarbeitet werden soll. Standardmäßig bedeutet *NULL*, dass ein Attribut fehlt. Gemäß Standardhandhabung wird dieses Attribut aus der Zuordnungsoperation entfernt. Als Endergebnis ist im Empfangseintrag kein Attribut mit dem angegebenen Namen vorhanden.

Verwenden Sie im Dialogfenster 'Nullverhalten' die Optionsfelder auf der rechten Seite, um *NULL* als leeren Zeichenfolgewert zu definieren, und verwenden Sie anschließend die Optionsfelder auf der linken Seite, um anzugeben, dass in diesem Fall der Standardwert *"* missing *"* zurückgegeben wird.

21. Oder Sie können den Connector selbst in der Komponentenliste der Fertigungslinie auswählen: Klicken Sie auf die Schaltfläche **Mehr...** in der Eingabezuordnung oben und wählen Sie dort 'Nullverhalten' aus.

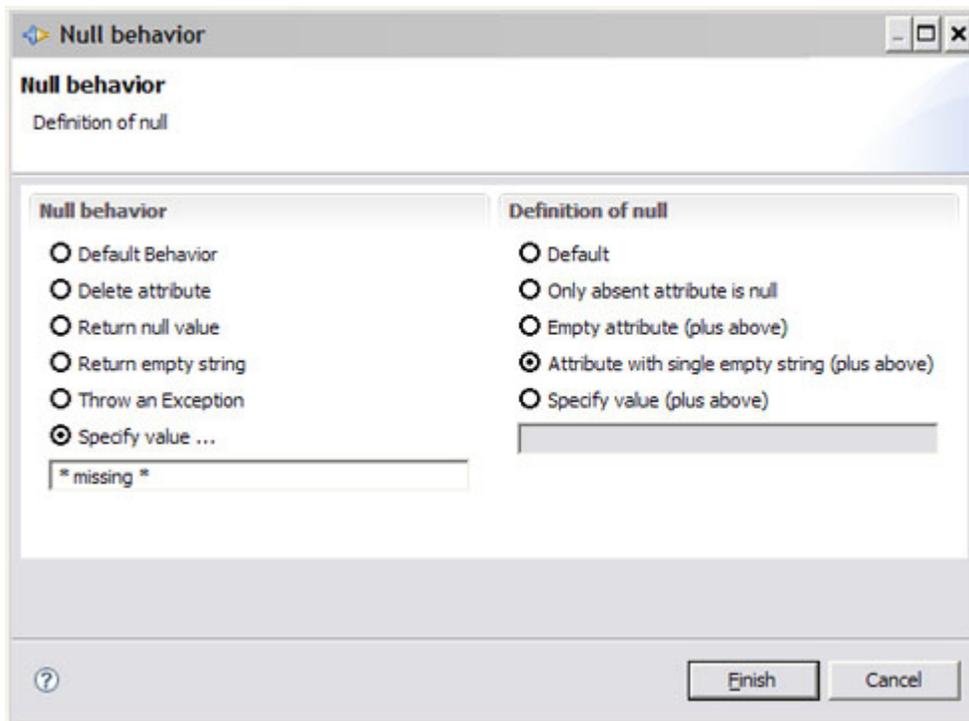


Abbildung 33. Konfigurationsdialog 'Nullverhalten'

Führen Sie Ihre Fertigungslinie erneut aus und aktualisieren Sie das Browserfenster, das den Inhalt der Datei Output.xml anzeigt. Der Eintrag für 'Roger' sollte nun den Sonderwert NULL ("* missing *") für 'Title' und 'LastName' aufweisen.

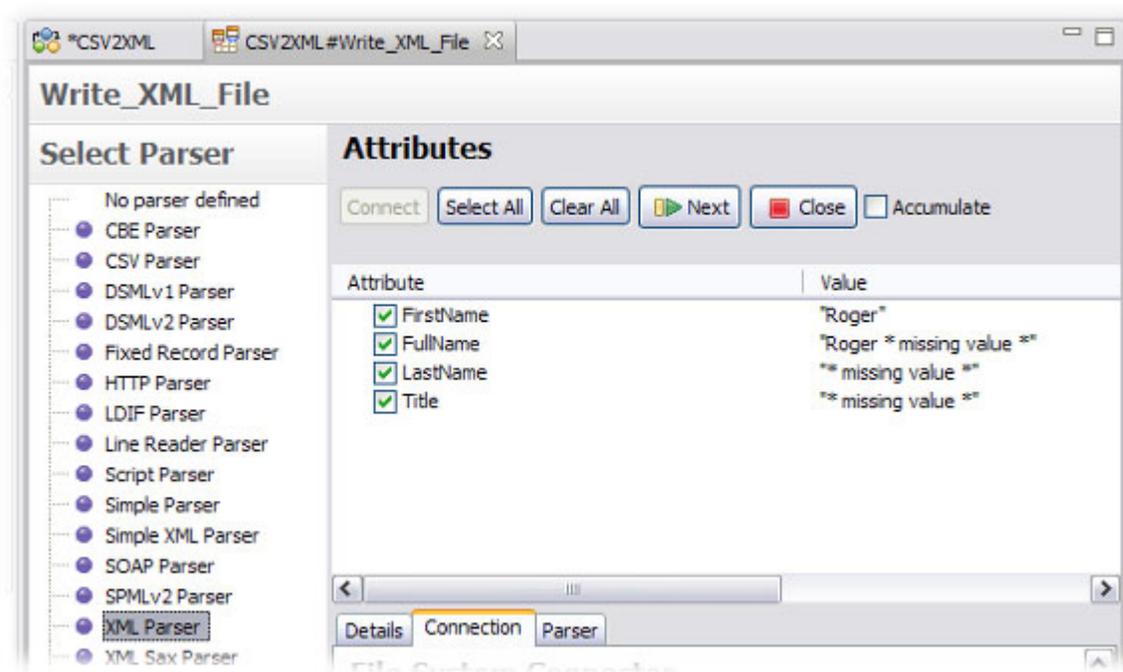


Abbildung 34. Ergebnis der Einstellungen für Nullverhalten in der XML-Ausgabe

Dies sieht etwas besser aus, zumindest handelt es sich um eine bewusste Auswahl. Manchmal ist ein Eintrag allerdings zu unvollständig, als dass eine Fortsetzung der Verarbeitung möglich wäre. In unserem Szenario benötigen Sie mindestens Werte für 'FirstName' und 'LastName', um 'FullName' zu berechnen. Deshalb fügen Sie Ihrer Fertigungslinie nun Filterlogik hinzu, um sicherzustellen, dass alle Einträge diese Anforderung erfüllen.

Klicken Sie zu Beginn erneut auf die Schaltfläche zum Einfügen einer Komponente und wählen Sie anschließend die Option **Steuerungs-/Datenflusskomponenten** aus den Optionsfeldern auf der linken Seite aus. Wählen Sie anschließend 'IF' in der Liste aus, nennen Sie sie 'Incomplete data' (Unvollständige Daten)²². und klicken Sie auf **Fertigstellen**.

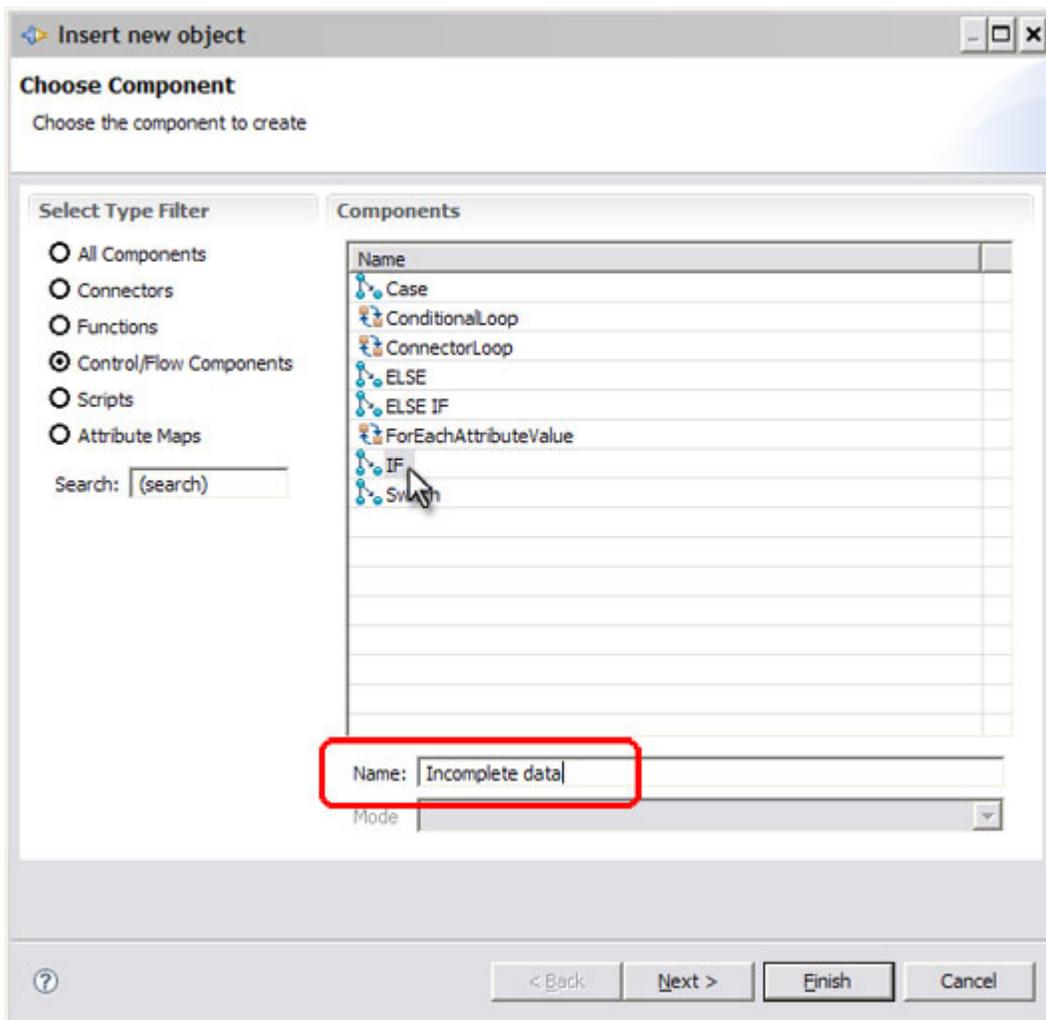


Abbildung 35. IF-Verzweigungskomponente auswählen

22. Bereits vorher wurde erwähnt, dass Sie Connector so benennen, wie Sie eine Scriptvariable benennen würden. Dies trifft auch auf Funktionskomponenten zu. Dies ist jedoch für Attributzuordnungskomponenten, Verzweigungen, Schleifen und Scripts weniger wichtig, da auf Sie selten direkt vom Script aus zugegriffen wird. In diesem Handbuch wurde größere Wichtigkeit darauf gelegt, dass das Lesen der Fertigungslinie einfach ist.

Ziehen Sie diese IF-Verzweigung nun nach oben, sodass sie vor Ihrem Ausgabeconnector steht. Im Editorbereich der IF-Verzweigung können Sie Ihre Bedingungen hinzufügen.

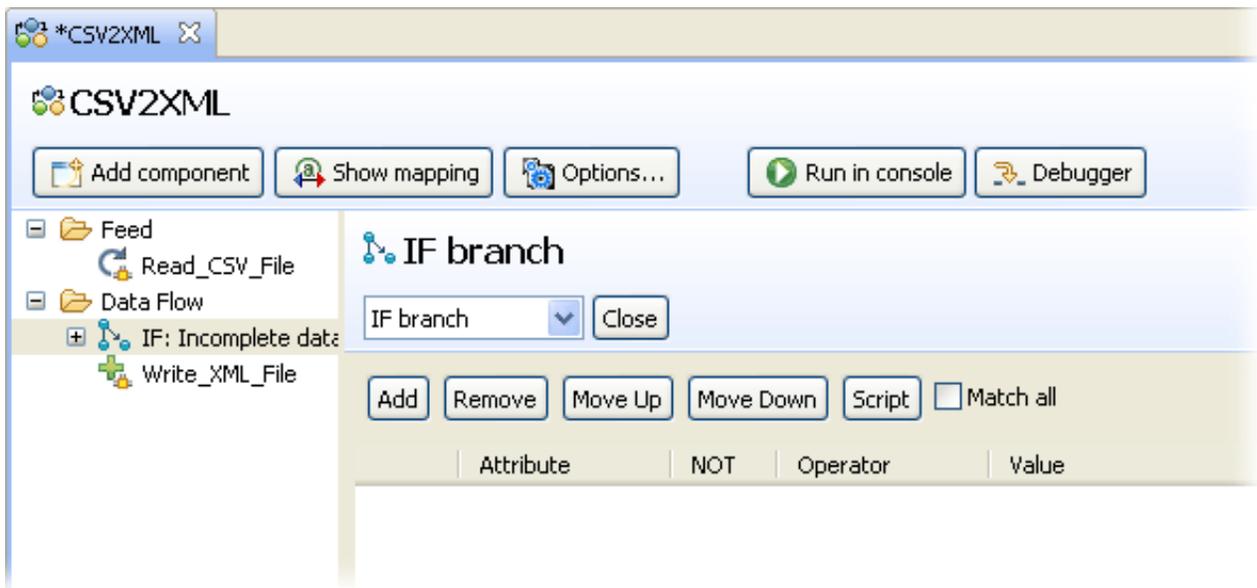


Abbildung 36. Bedingungen für die IF-Verzweigung bearbeiten

Sie haben hier die Möglichkeit, einfache Bedingungen hinzuzufügen oder ein Scriptfragment zu schreiben - oder beides zu tun. Beachten Sie das Kontrollkästchen **Übereinstimmung mit allen**, mit dem festgelegt wird, ob Ihre Bedingungen (einfache und scriptbasierte) mit einem zwischen ihnen stehenden implizierten *ODER* (Kontrollkästchen nicht ausgewählt) oder mit einem implizierten *UND* (Kontrollkästchen ausgewählt) ausgewertet werden.

Fügen Sie eine einfache Bedingung durch Klicken auf die Schaltfläche **Hinzufügen** hinzu. Wählen Sie anschließend das Attribut 'First' aus der Dropdown-Liste ganz links aus und wählen Sie anschließend den Operator 'has value(s)' (hat Werte(e)) aus. Verneinen Sie diese Bedingung, indem Sie den Spaltenwert **NICHT** umschalten. Im Feld ganz rechts muss bei Verwendung des Operators 'has value(s)' (hat Werte(e)) nichts angegeben werden. Fügen Sie nun auch dem Attribut mit dem Namen 'Last' eine ähnliche Bedingung *Weist keine Werte auf* hinzu. Stellen Sie abschließend sicher, dass das Kontrollkästchen **Übereinstimmung mit allen** nicht ausgewählt ist - dies würde *Beliebige Übereinstimmung* implizieren -, sodass diese Verzweigung ausgelöst wird, da für beide Attribute die Werte fehlen.

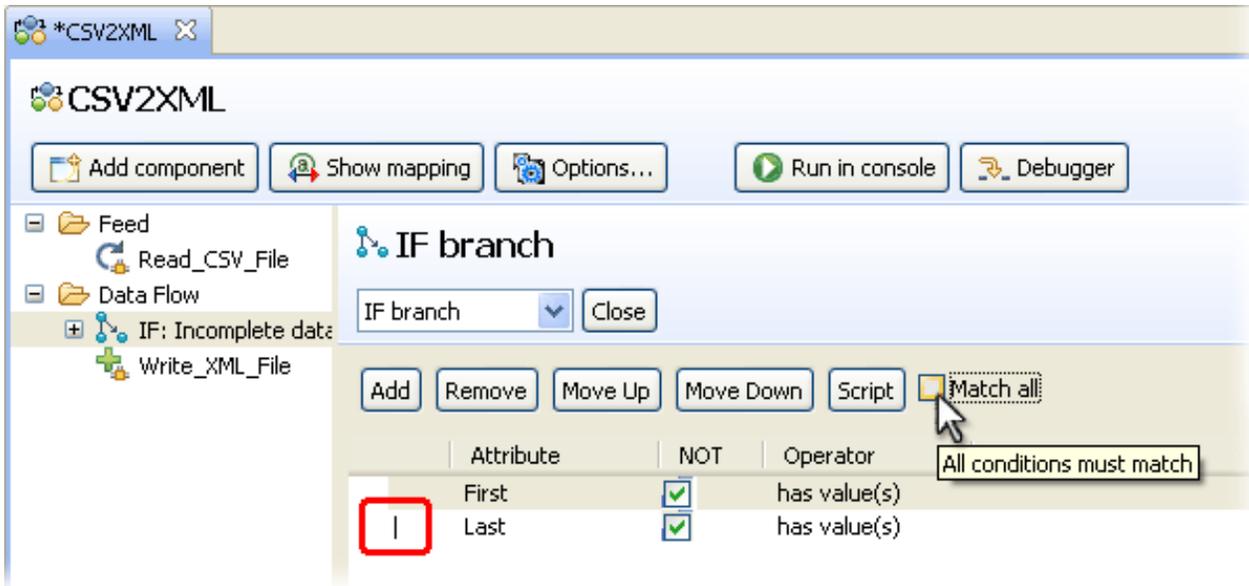


Abbildung 37. Einfache Bedingungen zur IF-Verzweigung hinzufügen

Eine IF-Verzweigung lenkt den Ausführungsablauf der Fertigungslinie in eine andere Richtung, wenn die angegebenen Bedingungen als *wahr* ausgewertet werden. In Ihrem Fall wird die Verarbeitung für die Komponenten unter der Verzweigung fortgesetzt, wenn dem Attribut 'First' oder dem Attribut 'Last' keine Werte zugeordnet sind oder wenn eines von ihnen sogar nicht im Work-Eintrag vorhanden ist. Mit anderen Worten, der Operator 'has value(s)' (hat Werte(e)) schließt auch die Prüfung auf 'exists' (vorhanden) mit ein.

Erweitern Sie nun die IF-Verzweigung und klicken Sie doppelt auf den Platzhalter, der unter ihr angezeigt wird, um dort eine Komponente einzufügen. Klicken Sie im Dialogfenster **Komponente auswählen** auf das Optionsfeld für **Scripts**, wählen Sie das mit dem Namen 'Script' aus und klicken Sie auf **Fertigstellen**. Benennen Sie diese Scriptkomponente (Script Component, SC) in 'Write to Log' (In Protokoll schreiben) um und geben Sie anschließend das folgende JavaScript-Fragment in die Komponente ein:²³

```
task.logmsg("*** Skipping incomplete entry");
```

Die hier verwendete Variable *task* verweist auf die Fertigungslinie selbst und bietet Ihnen eine Anzahl nützlicher Funktionen wie z. B. die Methode `logmsg()`. Dieser scriptbasierte Aufruf bewirkt, dass die angegebene Textnachricht in Ihre Protokollausgabe geschrieben wird.

23. Der Scripteditor bietet eine Funktion zur *Codevervollständigung*, die anzeigt, welche Optionen Ihnen zur Verfügung stehen. Geben Sie zum Beispiel "tas" in den Scripteditor ein und drücken Sie anschließend die Tastenkombination Strg + Leertaste, um die Dropdown-Liste der Codevervollständigung zu öffnen, die als einzige Option `task` angeben sollte. Wenn Sie die Eingabetaste drücken, wird diese Option ausgewählt und in Ihr Script eingegeben. Drücken Sie nun die Dezimalpunktstaste (.), sodass Ihr Script "task." lautet. Warten Sie einen Augenblick. Eine neue Dropdown-Liste für die Codevervollständigung wird angezeigt, nun mit einer Liste aller Methoden und Eigenschaften, auf die Sie im Objekt `task` zugreifen können.

Schließen Sie auch den aktuellen Inhalt des Work-Eintrags ein, damit Ihre Protokollausgabe noch mehr Informationen enthält. Führen Sie dies aus, indem Sie mit der rechten Maustaste auf die IF-Verzweigung klicken, die Option **Komponente hinzufügen...** auswählen und erneut das Optionsfeld für **Scripts** auswählen. Wählen Sie nun die vordefinierte Scriptkomponente mit dem Namen 'SpeicherauszugfürWork_Eintrag' aus.

Abschließend müssen Sie die Fertigungslinie anweisen, den aktuellen Zyklus an diesem Punkt zu stoppen und die Steuerung an den Connector im Iteratormodus zurückzugeben, sodass er die nächste CSV-Zeile einlesen kann und dabei den aktuellen Eintrag effektiv aus der XML-Ausgabe filtert. Dieses Verhalten müssen Sie selbst angeben, andernfalls fährt die Steuerung mit der ersten Komponente nach der IF-Verzweigung fort. Sie müssen also erneut mit der rechten Maustaste auf Ihre IF-Verzweigung und anschließend auf die Option **Komponente hinzufügen...** zum Einfügen einer Komponente klicken. Wählen Sie **Scripts** aus und wählen Sie anschließend die Scriptkomponente mit dem Namen 'Exitdatenfluss' aus. Ihre Fertigungslinie sollte nun wie folgt aussehen:

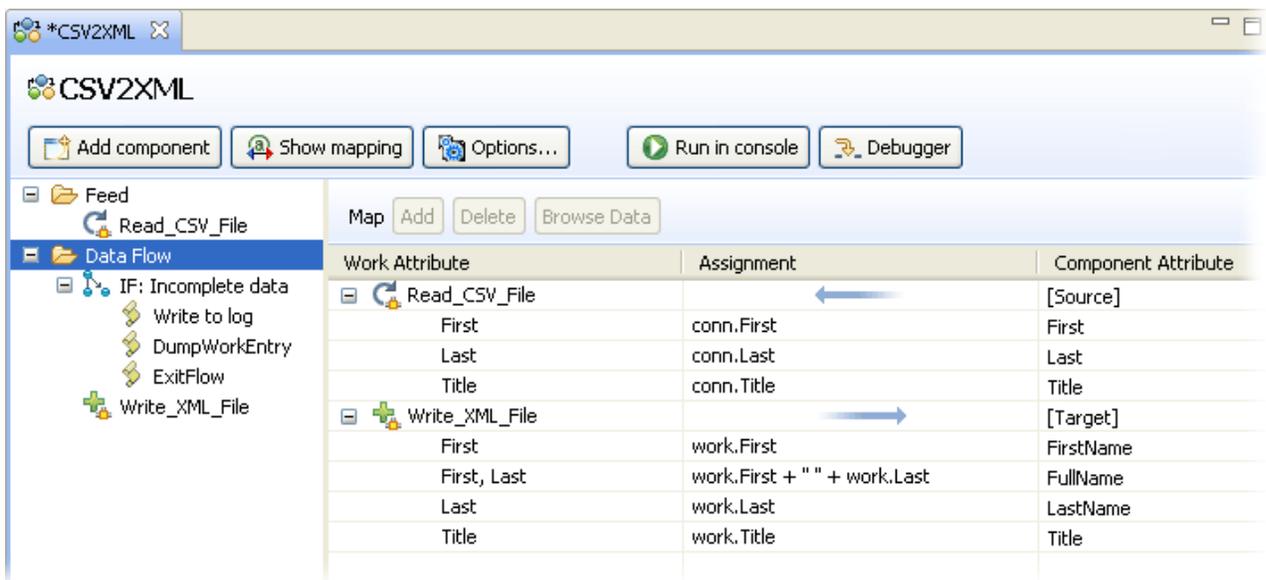


Abbildung 38. Ihre erste vollständige Fertigungslinie

Bevor Sie Ihre Fertigungslinie erneut ausführen, müssen Sie noch einmal das Dialogfenster 'Nullverhalten' öffnen und die Standardauswahl für Definition und Verhalten wiederherstellen - andernfalls werden Ihre IF-Verzweigungsbedingungen niemals als *wahr* (true) ausgewertet.

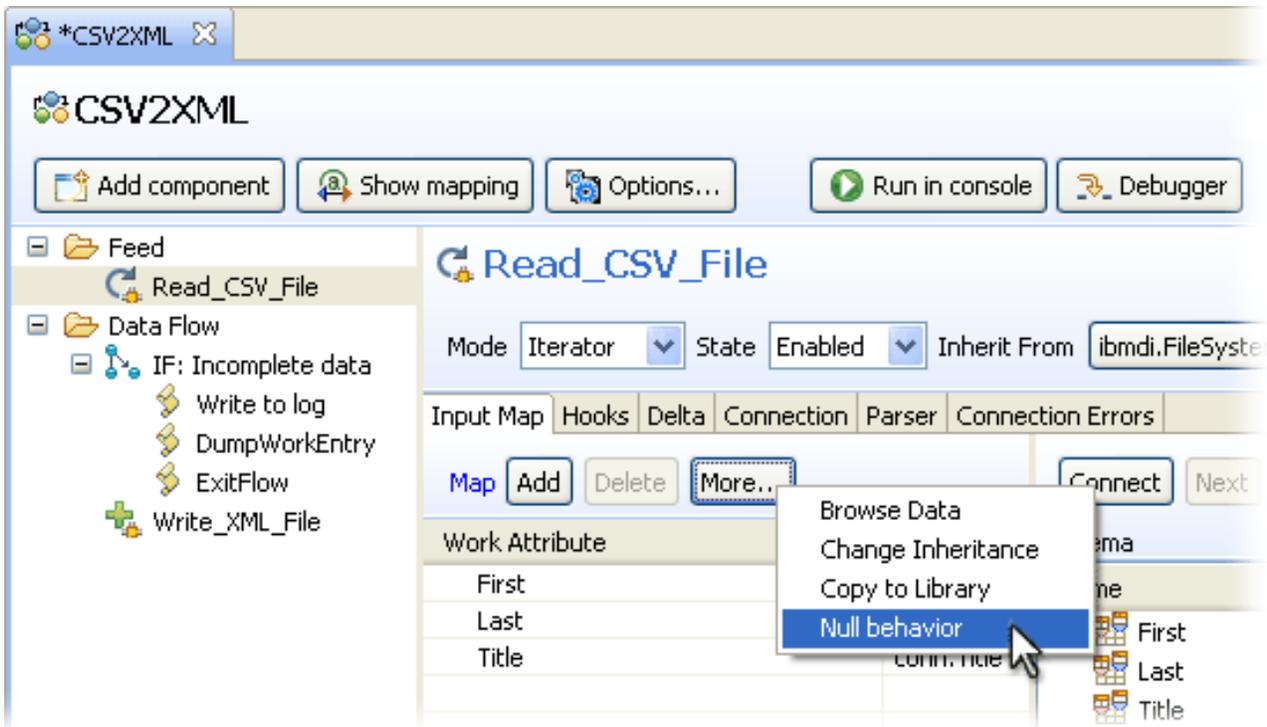


Abbildung 39. Nullverhalten für die Fertigungslinie zurücksetzen

Wenn Sie Ihre Fertigungslinie nun erneut ausführen, wird Ihre Nachricht gefolgt vom Speicherauszug des Eintrags angezeigt:

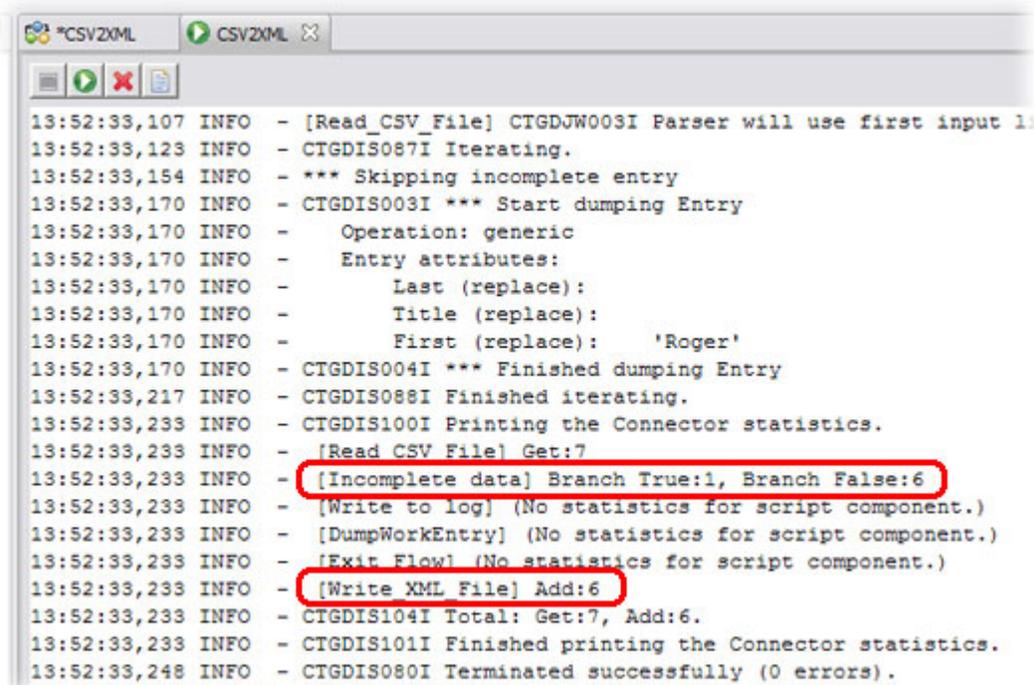


Abbildung 40. Protokollausgabe mit Ihren Nachrichten und dem Speicherauszug des Eintrags

An den statistischen Daten sehen Sie, dass Ihre Verzweigung 'Daten unvollständig' einmal 'true' (wahr) war und als Ergebnis davon Ihrer XML-Ausgabe lediglich sechs Knoten hinzugefügt wurden. Wenn Sie das Datenbrowserfenster erneut aktualisieren, sehen Sie, dass 'Roger' tatsächlich nicht mehr vorhanden ist.

Debug für Ihre Fertigungslinie ausführen

Eine der leistungsfähigsten Komponenten von IBM Security Directory Integrator ist der zugehörige integrierte Fertigungsliniendebugger, der es möglich macht, dass Sie die Ausführung Ihrer Fertigungslinie schrittweise durchgehen, Daten anzeigen und diese sogar an Ort und Stelle modifizieren können.

Bewegen Sie sich schrittweise durch Ihre erste Fertigungslinie, indem Sie auf die Schaltfläche **Debugsitzung starten** klicken.

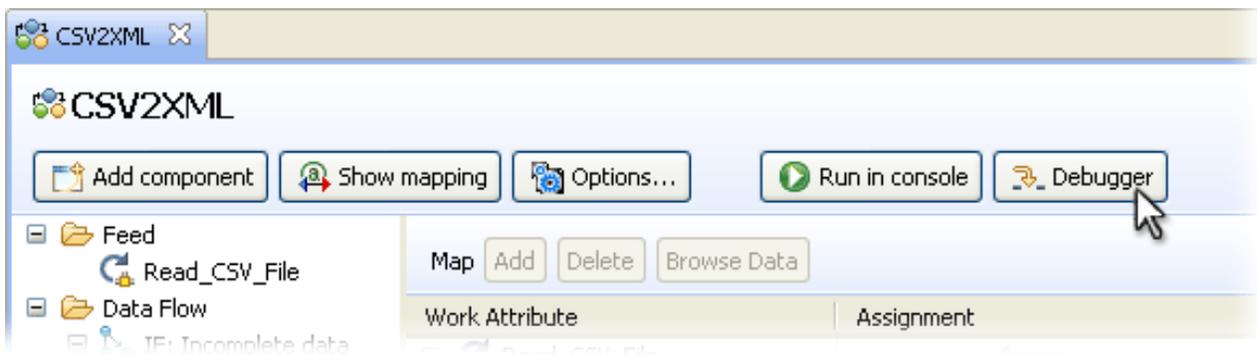


Abbildung 41. Debug für Ihre Fertigungslinie ausführen

Statt des Fensters mit der Standardausführungskonsole wird Ihnen das Debugfenster angezeigt.

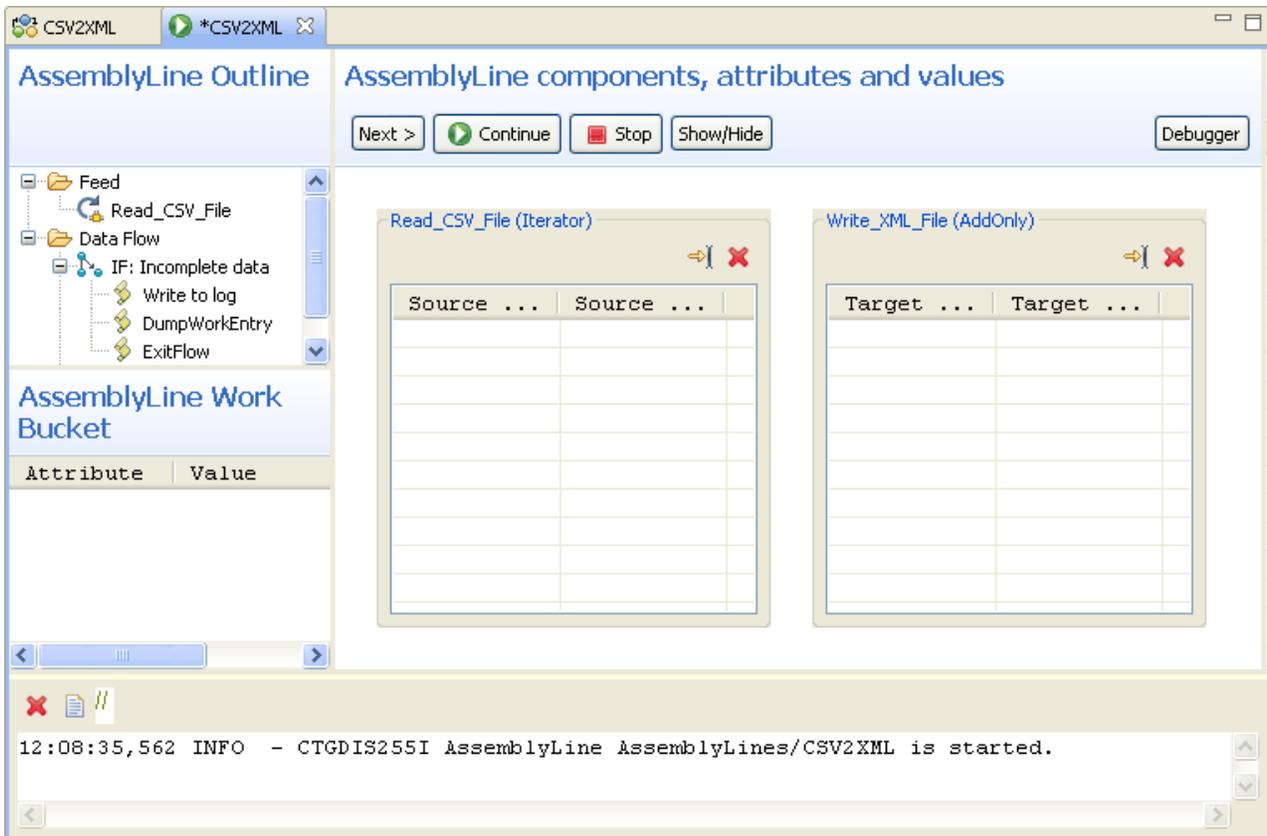


Abbildung 42. Der Data Stepper der Fertigungslinie

Der Fertigungsliniendebugger bietet zwei Modi: Den Data Stepper, der einfache und unkomplizierte Testfunktionen bietet, und den erweiterten Fertigungsliniendebugger, mit dem Sie detailliertere Tests durchführen können wie zum Beispiel das schrittweise Durchgehen von Scripts, die interaktive Arbeit mit Java-Bibliotheken und das Ändern von Daten in der Datei.

Der Data Stepper ist ein nützliches Tool, um die Ausführung Ihrer Fertigungslinienconnectors schrittweise durchzugehen und das Lesen, Schreiben und Konvertieren der Daten anzuzeigen. Diese Anzeige ist in drei Hauptbereiche unterteilt:

- Der **Fertigungslinienentwurf** zeigt Ihre Fertigungslinie mit Hervorhebungen an den Stellen an, an denen die Ausführung stoppt.
- Der **Work-Bucket der Fertigungslinie** zeigt alle Attribute an, die dem Arbeitseintrag zugeordnet sind, d. h., die Attribute, die in den Eingabezuordnungs- oder Attributzuordnungs-komponenten gefunden wurden.
- Der Bereich mit **Fertigungslinienkomponenten, Attribute** und **Werte**, in dem Ihnen eine Zeile mit Schaltflächen zur Steuerung Ihrer Debugsitzung und eine Gruppe von Anzeigegittern für alle Connectors in der Fertigungslinie zur Verfügung stehen.

Unten wird ein Fenster mit der Konsolenausgabe angezeigt, das dieselben Informationen enthält, die Sie bei der Ausführung Ihrer Fertigungslinie über die Schaltfläche 'Ausführen' erhalten würden.

Zu diesem Zeitpunkt wurde Ihre Fertigungslinie zum Testserver gesendet, und ist startbereit. Die Ausführung erfolgt auf Ihren Befehl hin. Verwenden Sie die Schaltfläche 'Weiter', um mit dem schrittweisen Durchgehen zu beginnen.

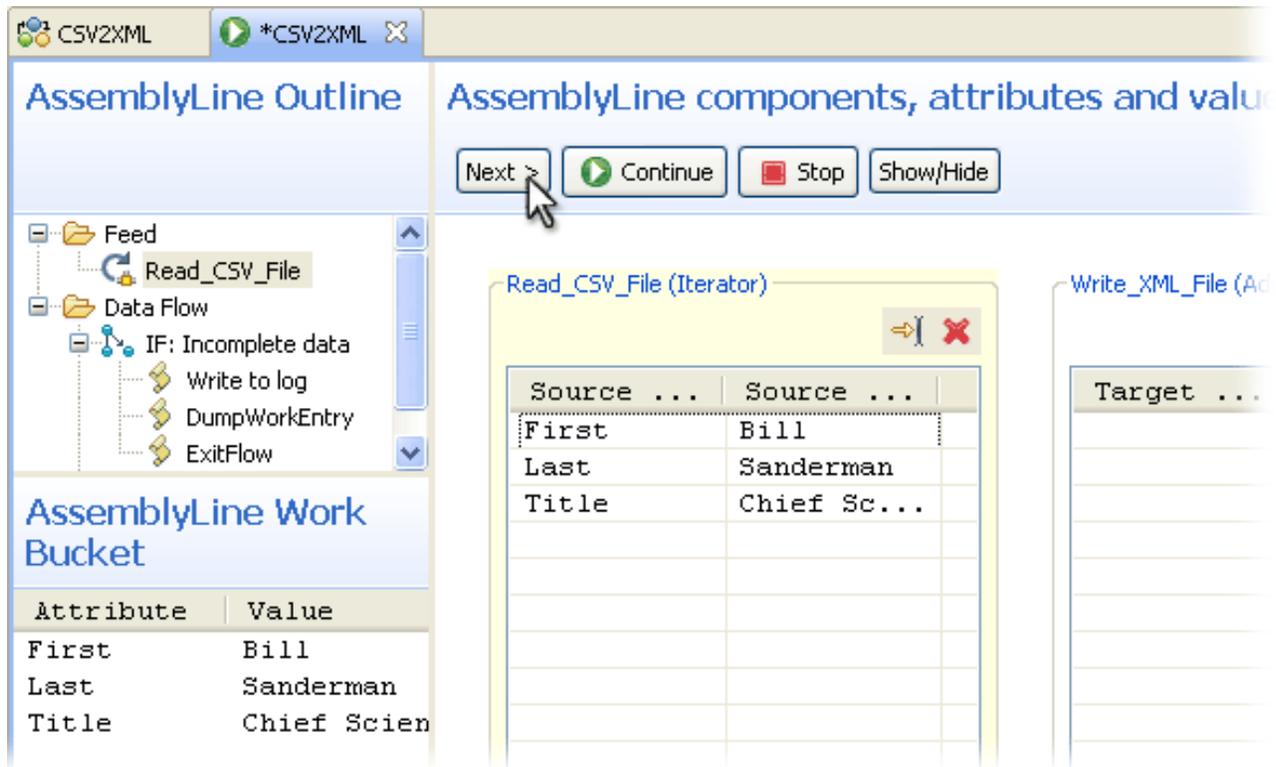


Abbildung 43. Schrittweises Durchgehen der Fertigungslinienausführung

In der Anzeige können Sie drei Dinge beobachten: Der **Fertigungsliniementwurf** zeigt, dass der Connector "Read_CSV_File" momentan aktiv ist. Der **Work-Bucket der Fertigungslinie** zeigt die Attribute an, die gerade von diesem Connector gelesen wurden. Das Datenanzeigegitter für diesen Connector wird ebenfalls mit diesen Attributen gefüllt. Immer wenn Sie auf die Schaltfläche **Weiter** klicken, wird der nächste Schritt ausgeführt und die Informationsanzeigen werden aktualisiert.

Sie können ferner die Schaltfläche **Ausführung bis zu diesem Punkt** oben im Datengitter für einen Connector verwenden, um zum angegebenen Punkt zu springen. Führen Sie diese Aktion jetzt für den Connector 'Write_XML_File' aus.

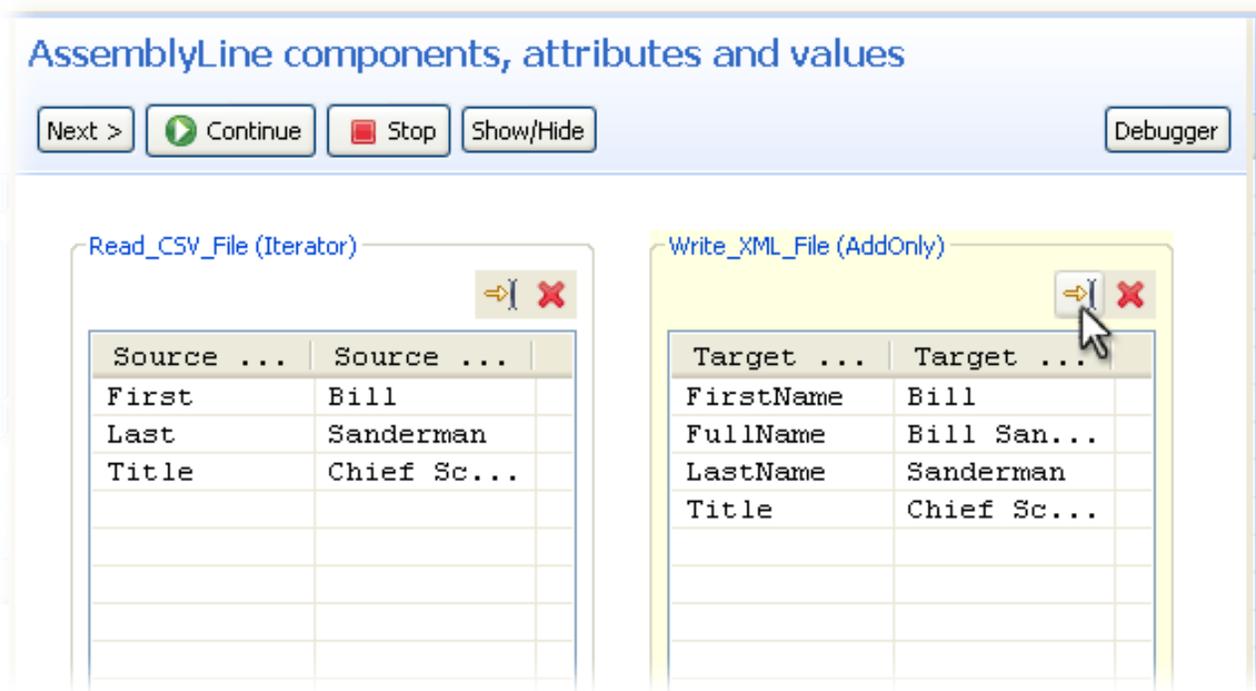


Abbildung 44. Ausführung bis zum Connector 'Write_XML_File'

Das Datengitter zeigt nun die Attribute in der Ausgabezuordnung dieses Connectors zusammen mit den zugehörigen Werten an - einschließlich des berechneten Wertes für 'FullName'.

Betrachten Sie die Schaltflächen in der Symbolleiste des Data Steppers an, um festzustellen, welche Möglichkeiten Sie jetzt haben.

- **Weiter** > Die Verarbeitung erfolgt bis zur nächsten Komponente und alle Datenanzeigebereiche werden aktualisiert.
- **Fortsetzen** Die Ausführung der Fertigungslinie erfolgt bis zur Fertigstellung.
- **Stoppen** Die Ausführung wird sofort beendet.
- **Ein-/ausblenden** Sie können entscheiden, welche Connector datengitter angezeigt werden sollen.
- **Debugger** Es erfolgt ein Wechsel in den vollen Debugmodus, in dem Sie den Script-Code schrittweise durchgehen, Unterbrechungspunkte setzen, beliebige Attribute und Script-Variablen anzeigen und ändern und JavaScript-Befehle im Kontext Ihrer Fertigungslinie interaktiv ausführen können.

Obwohl der Data Stepper eine große Anzahl an Informationen darüber bietet, wie Ihre Fertigungslinie ausgeführt wird, benötigen Sie in einigen Fällen die zusätzliche Leistung des erweiterten Debuggers. Denken Sie daran, dass Sie während einer Debugsitzung zwischen dem Data Stepper und dem erweiterten Modus so oft Sie möchten hin- und herschalten können. Versuchen Sie dies nun, indem Sie auf die Schaltfläche **Debugger** ganz rechts in der Zeile der Schaltflächen des Data Steppers klicken.

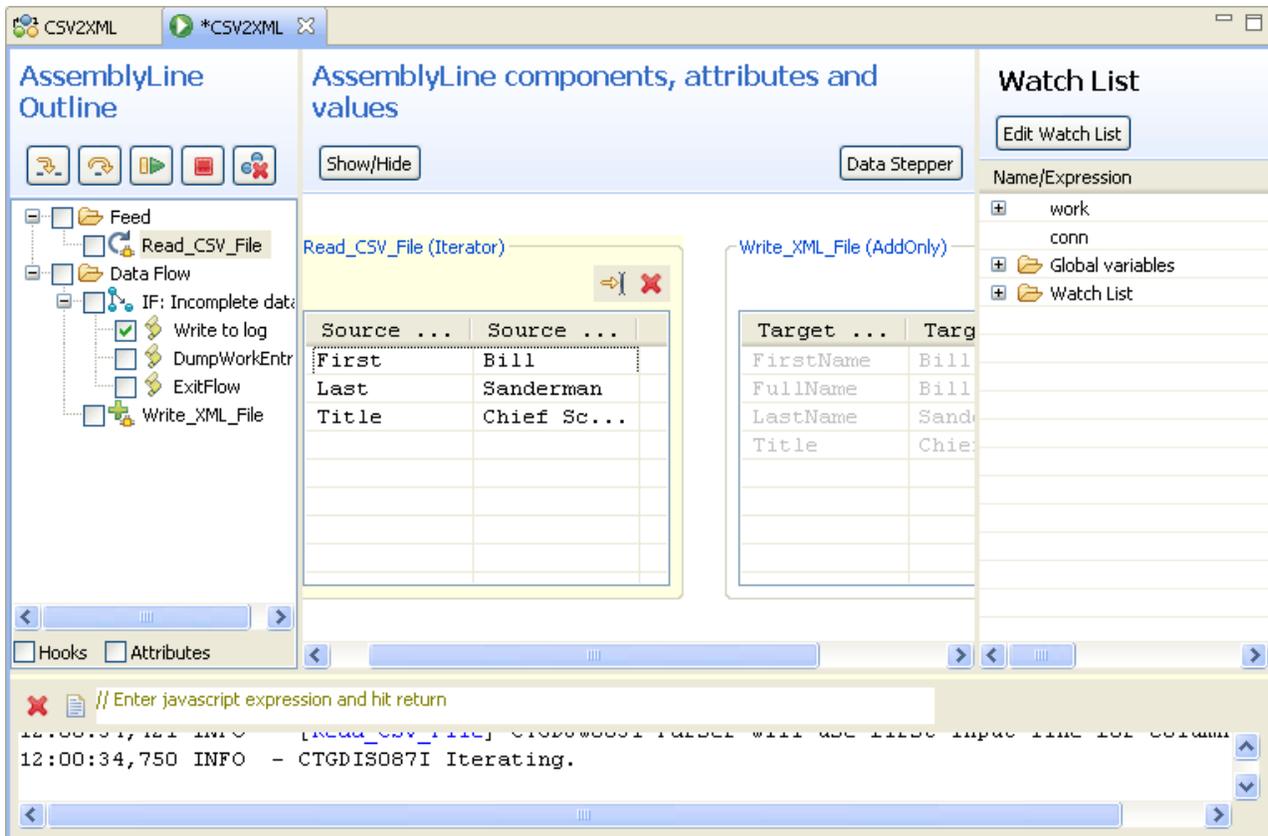


Abbildung 45. Erweiterter Debugmodus

Wenn Sie umschalten, wird die Anzeige neu erstellt, um neue Steuerelemente für den **Fertigungsliniendesign** zur Verfügung zu stellen, und der **Work-Bucket der Fertigungslinie** wird durch die **Überwachungsliste** rechts im Fenster ersetzt. Die **Überwachungsliste** zeigt die Standardattribute *Buckets* an: 'work' und 'conn'. Ferner gibt es einen Ordner mit dem Namen 'Globale Variablen', der beim Öffnen alle Variablen anzeigt, die für Ihre Fertigungslinie definiert wurden. Dazu gehören auch die integrierten Variablen wie **work** und **system** sowie alle von Ihnen im Script-Code definierten Variablen. Der letzte Überwachungsordner steht Ihnen zu Ihrer Verwendung zur Verfügung. Sie können Variablen oder ganze JavaScript-Ausdrücke, die Sie überwachen möchten, mithilfe der Schaltfläche **Überwachungsliste editieren** oben in dieser Anzeige hinzufügen.

Sehen Sie sich jetzt den **Fertigungsliniendesign** an. Die Felder neben den Komponenten in dieser Baumstrukturansicht werden als Unterbrechungspunkte bezeichnet und Sie können IBM Security Directory Integrator während der Ausführung bei einer beliebigen Komponente stoppen lassen, indem Sie auf die betreffenden Komponenten klicken. Sie können ferner mit der rechten Maustaste auf einen beliebigen Knoten klicken und **Ausführen und hier unterbrechen** auswählen, um die Ausführung der Fertigungslinie bis zu diesem Punkt auszuführen und dort zu unterbrechen. Die Symbolleiste oberhalb des Entwurfs enthält einige der Steuerelemente, die Ihnen auch im Data Stepper zur Verfügung stehen. Hinzu kommen eine Reihe neuer Steuerelemente:



Abbildung 46. Schaltflächen des Debuggers

Die Schaltfläche sind von links beginnend folgende:

- **Step Into** Sie können eine Step-Into-Operation in Attributzuordnungen, Scripts und sogar in die zugrunde liegenden Workflows der Fertigungslinie und ihrer Komponenten ausführen. Diese Punkte in integrierten Flüssen werden als "Hooks" (Haken) bezeichnet und später in einer Übung erläutert.
- **Step Over** Diese Option entspricht der Schaltfläche **Weiter >**, die Sie aus Data Stepper kennen. Im Debugger gilt diese Option auch für Funktionsaufrufe eines Scripts.
- **Fortsetzen** Die Fertigungslinie wird bis zum Ende ausgeführt (so wie in Data Stepper) oder bis ein Unterbrechungspunkt erreicht wird.
- **Stoppen** Die Fertigungslinie wird angehalten (so wie in Data Stepper).
- **Alle Unterbrechungspunkte entfernen** Alle Unterbrechungspunkte, die Sie für Ihre Fertigungslinie festgelegt haben, werden entfernt.

Um ein Gefühl für die Funktionsweise von Unterbrechungspunkten zu bekommen, versuchen Sie bitte, einen Unterbrechungspunkt für das Script 'Write to log' festzulegen, indem Sie in das Feld neben dieser Komponente klicken.

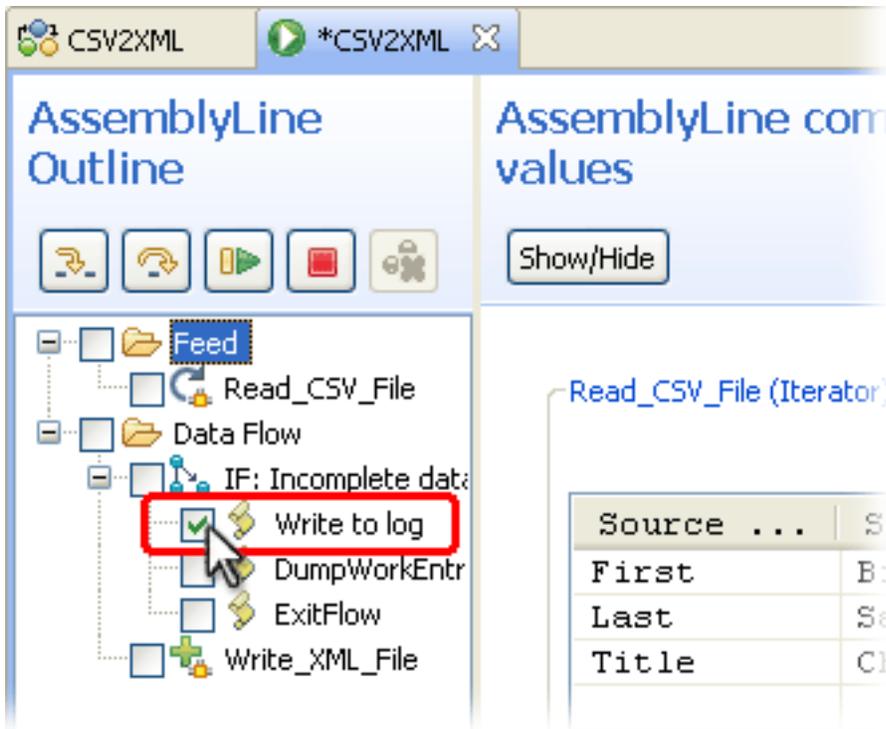


Abbildung 47. Unterbrechungspunkt festlegen

Klicken Sie jetzt auf die Schaltfläche **Fortsetzen**, damit Ihre Fertigungslinie ausgeführt wird, bis die IF-Verzweigung wahr (*true*) ist und Sie über das Script 'Write to log' die Steuerung erhalten. IBM Security Directory Integrator öffnet darüber hinaus einen Scriptbereich, in dem Sie den Code schrittweise durchgehen können. Sie können Unterbrechungspunkte in jeder beliebigen Zeile des Scripts festlegen, indem Sie doppelt auf den linken Rand dieser Zeile klicken.



Abbildung 48. Unterbrechungspunkt im Script festlegen

Desweiteren können Sie auf jeden Knoten in der Komponentenliste doppelt klicken, um die Debuganzeige aufzurufen. Wie Sie in der Abbildung oben sehen können, gibt es eine Schaltfläche mit dem Namen **Unterbrechungspunktbedingung**. Sie können diese Schaltfläche verwenden, um einen JavaScript-Ausdruck zu definieren, der entweder mit *true* oder *false* ausgewertet werden muss, und der bestimmt, ob ein Unterbrechungspunkt aktiv ist oder nicht. Der Unterbrechungspunkt oben könnte zum Beispiel als aktiv definiert werden, wenn

```
work.First.startsWith("R")
```

oder

```
mycounter > 1000
```

Dies ist sehr hilfreich bei Debugproblemen, die nur in einigen Details der Eingabedatei auftreten.

Falls Sie aus irgendeinem Grund zum vorherigen Schritt zurückkehren müssen, können Sie Ihre Debugsitzung einfach stoppen und erneut starten. Sie können auch zum Data Stepper umschalten, indem Sie auf die Schaltfläche 'Data Stepper' klicken.

Bevor Sie den erweiterten Debugger verlassen, sollten Sie sich jedoch noch eine interessante Funktion anschauen: **Befehlszeile für JavaScript-Auswertung**.

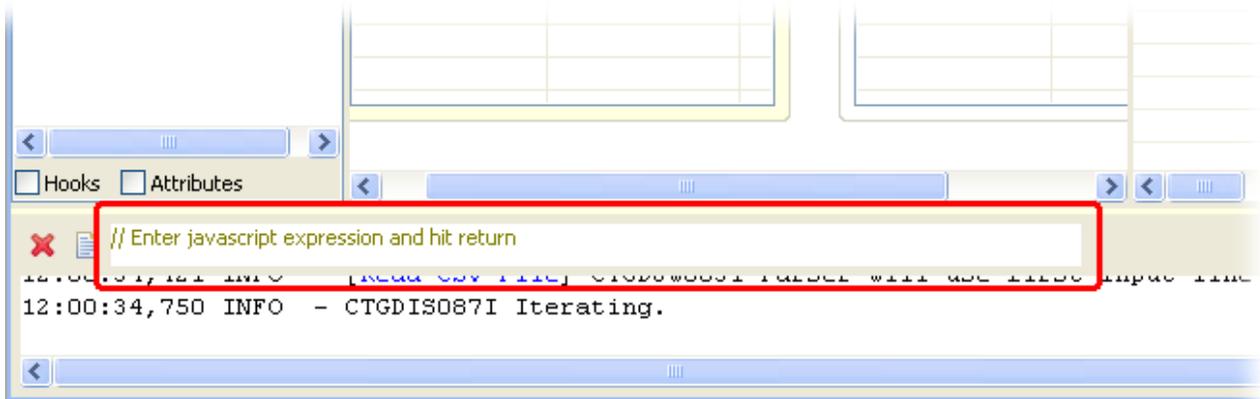


Abbildung 49. Befehlszeile für JavaScript-Auswertung

Dieses Eingabefeld oberhalb des Protokollausgabebereichs ermöglicht Ihnen die Ausführung eines beliebigen Scriptfragments im Kontext der Ausführung Ihrer Fertigungslinie. Versuchen Sie dies nun, indem Sie den folgenden Befehl eingeben und dann die Eingabetaste drücken:

```
task.dumpEntry(work)
```

Daraufhin wird der Inhalt des Arbeitseintrags im Fenster der Protokollausgabe angezeigt. Geben Sie nun Folgendes ein:

```
i = 42
```

Die folgende Nachricht wird im Protokoll angezeigt: `i=42 >> 42.0`

Diese Nachricht besagt, dass Sie eine neue Variable ('i') mit dem Wert 42 definiert haben. Der Ausdruck selbst wird (wie alle Scriptanweisungen) mit dem Wert der Zuordnung ausgewertet. Sie können die Werte von bereits in Ihrer Fertigungslinie definierten Variablen und Attributen ändern. Beispiel:

```
work.First="Rudy"
```

Nach der Ausführung dieser Zeile, lautet der Wert des Attributes 'First' folglich 'Rudy'. Die Möglichkeit der Änderung von Daten an Ort und Stelle heißt, dass Sie sicherstellen können, dass Ihre Fertigungslinie eine Step-Into-Operation für alle Verzweigungslogiken durchführt, sodass Sie Ihre Lösung vollständig testen können.

Es wird dringend empfohlen, dass Sie sich ausreichend Zeit nehmen, um mit dem Data Stepper und dem Fertigungsliniendebugger vertraut zu werden. Dadurch erhalten Sie nicht nur wichtige Einblicke in die Arbeitsweise Ihrer Fertigungslinie, einschließlich aller integrierten Workflows, die durch den IBM Security Directory Integrator-Server-Kernel bereitgestellt werden, sondern Sie werden auch bei der Bewertung Ihrer eigenen Implementierung und Ihrer Annahmen zu Ihren Daten unterstützt.

Daten in einer sequenziellen Quelle suchen

Als nächsten Schritt im Lernprogrammsszenario wird nun eine Suche in D2 hinzugefügt.

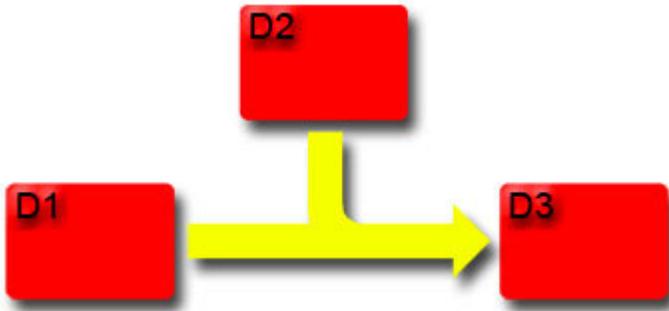


Abbildung 50. Das Flussdiagramm des Szenarios

Das Verzeichnis 'Tutorial' enthält eine Datei mit dem Namen PhoneNumbers.xml, die als Ihre Datenquelle 'D2' dienen soll. Diese Datei enthält eine Reihe von XML-Einträgen, von denen jeder zwei Attribute aufweist: 'User' und 'telephoneNo'.

Sie sollen nun 'telephoneNo' als Teil der Daten einschließen, die in das XML-Ausgabedokument geschrieben werden. Da Sie auf diese Textdatei nicht beliebig zugreifen können, um eine Suche durchzuführen, wie dies in einer Datenbank oder einem Verzeichnis möglich ist, wird die korrekte Telefonnummer für jeden CSV-Eintrag gesucht, indem die Datei in einer Schleife durchlaufen wird und 'User' mit 'FullName' verglichen wird, das aus dem aktuellen CSV-Eintrag stammt.

'FullName' wird jedoch zunächst in der Ausgabezuordnung des Connector 'Write_XML_File' berechnet, d. h. es ist zum Ausführen des Vergleichs zu spät. Dies bedeutet, dass Sie dieses berechnete Attribut aus der Ausgabezuordnung von 'Write_XML_File' in die Eingabezuordnung von 'Read_CSV_File' verschieben müssen. Dies führen Sie aus, indem Sie zuerst das Attributzuordnungselement aus der einen Zuordnung nach oben in die andere Zuordnung ziehen.

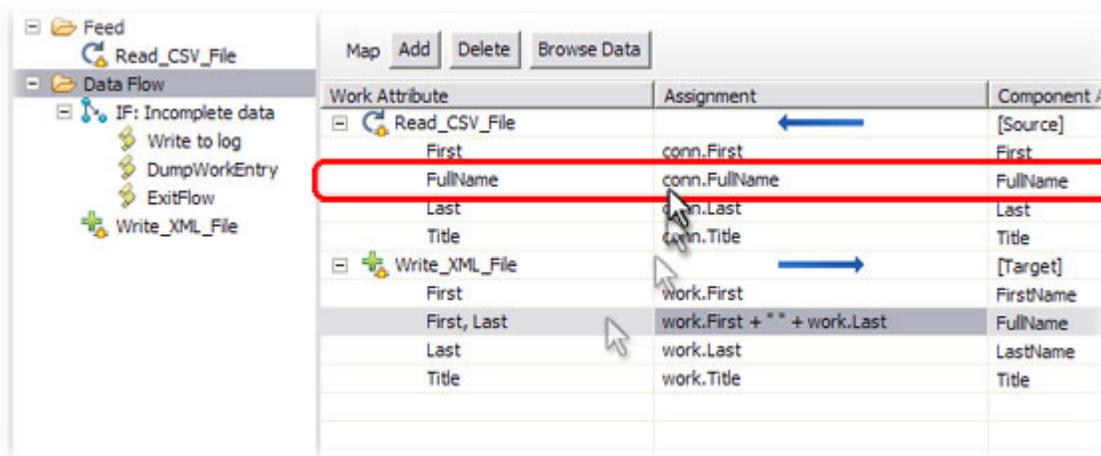


Abbildung 51. 'FullName' in die Eingabezuordnung Ihres Connectors im Iteratormodus ziehen

Nun ist in beiden Zuordnungen ein Zuordnungselement 'FullName' enthalten. Sie müssen die Zuordnung der Eingabezuordnung anpassen, da Sie nun eine Zuordnung von einem Conn-Eintrag zum Work-Eintrag durchführen, anstatt andersherum, wie es bei einer Ausgabezuordnung der Fall wäre. Führen Sie dies aus, indem Sie doppelt auf 'FullName' klicken, dass sich unter 'Read_CSV_File' befindet, und die Zuordnung wie folgt ändern:

conn.First + " " + conn.Last

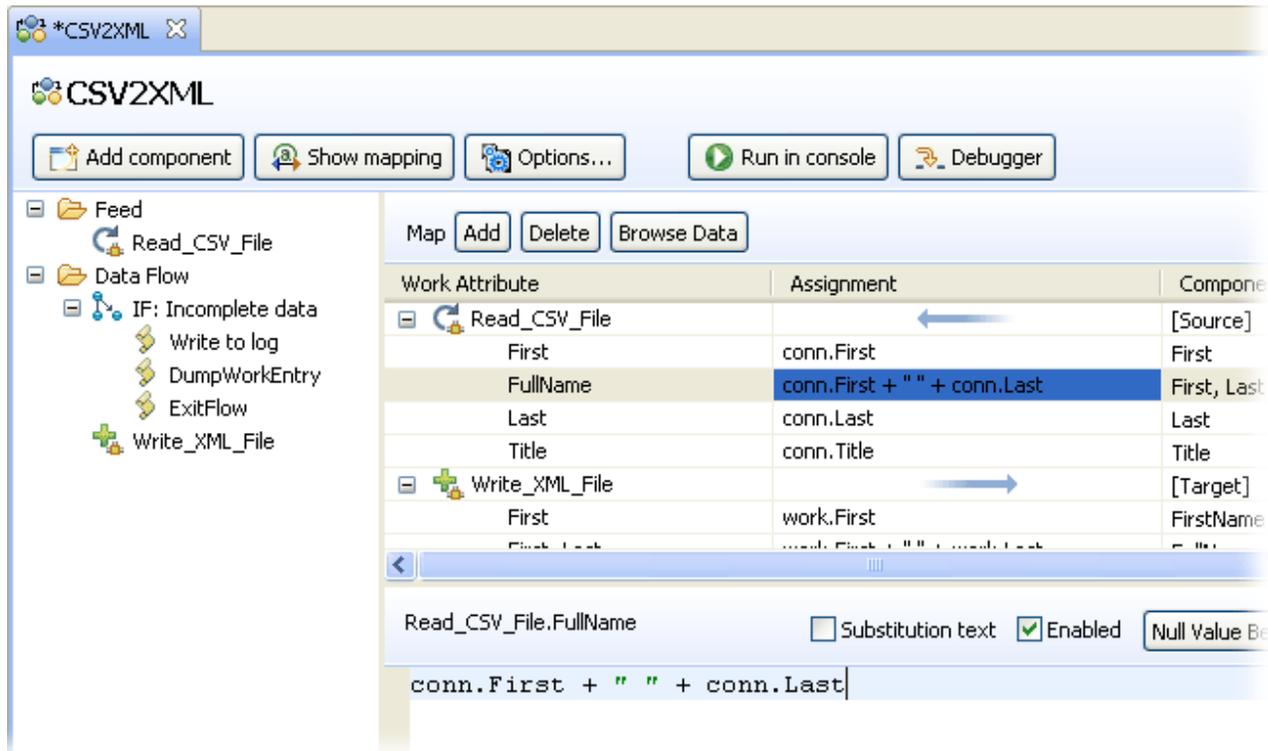


Abbildung 52. Zuordnung für 'FullName' bearbeiten

Sie können auch das ursprüngliche Ausgabezuordnungselement 'FullName' bearbeiten, sodass die zugehörige Zuordnung einfach `work.FullName` lautet, da dieses Attribut dank Ihrer modifizierten Eingabezuordnung nun im Work-Eintrag zur Verfügung stehen wird.

Führen Sie die Fertigungslinie nun erneut aus und überprüfen Sie die Datei `Output.xml`, um sicherzustellen, dass sie ohne Änderungen ist. Wenn Sie dies bestätigt haben, verwenden Sie eine *Schleifenkomponente*, um die Datei `PhoneNumbers.xml` durchzulesen und nach den Nummern der einzelnen Benutzer zu suchen.²⁴

Fügen Sie zunächst dem Abschnitt **Datenfluss** eine neue Komponente hinzu. Wählen Sie diesmal die Komponente mit der Bezeichnung 'Connectorschleife' (*ConnectorLoop*) aus und nennen Sie sie 'TelephoneNumber'. Eine Connectorschleife ist eine Schleifenkomponente, die zum Lesen von Informationen aus einer Datenquelle einen Connector verwendet und anschließend alle Komponenten, die unter ihm zugeordnet sind, pro Eintrag, der von diesem Connector zurückgegeben wird, einmal zyklisch verarbeitet. Dies ähnelt dem Verhalten *FOR-EACH* eines Connectors im Iteratormodus im Abschnitt 'Zuführung', der Komponenten im Abschnitt **Datenfluss** für jeden gelesenen Eintrag zyklisch verarbeitet.

24. Es sind drei Typen von Schleifenkomponenten vorhanden: 1) Die *Connectorschleife*, mit der Sie Daten zyklisch verarbeiten können, die von einem Connector im Modus 'Iterator' oder 'Lookup' zurückgegeben werden. Dies ist der Schleifentyp, den Sie in dieser Übung verwenden. 2) Die 'Schleife für Attributwert' (*ForEachAttributeValueLoop*), mit der Sie die Werte eines mehrwertigen Attributs, wie sie in Systemen wie Lotus Notes und LDAP-Verzeichnissen vorhanden sind, leicht in einer Schleife durchlaufen können. 3) Die 'Bedingte Schleife' (*ConditionalLoop*), die einfache und scriptbasierte Bedingungen verwendet (wie sie auch von Verzweigungen verwendet werden), um die zyklische Verarbeitung zu steuern.

Ziehen Sie Ihre neue Connectorschleife 'TelephoneNumber' zwischen die IF-Verzweigung und den Connector 'Write_XML_File'. Stellen Sie sicher, dass sie nicht *in* die IF-Verzweigung gelangt.

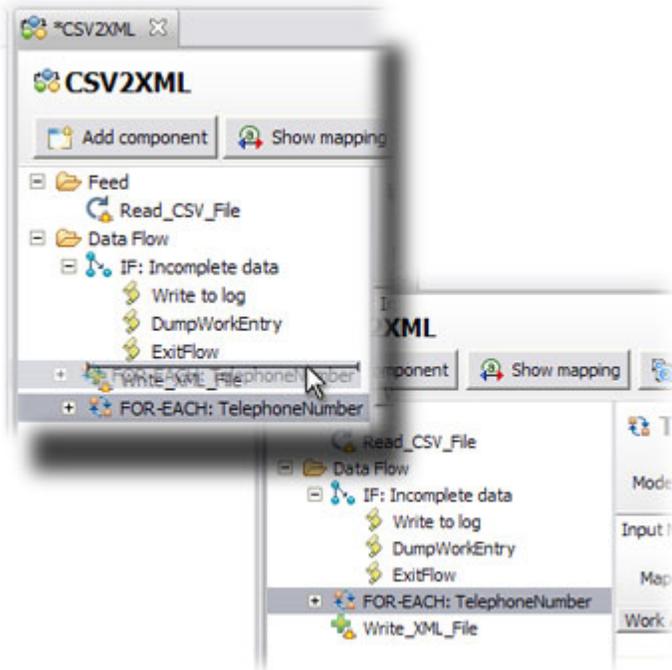


Abbildung 53. Connectorschleife ziehen

Wählen Sie sie nun aus, um den zugehörigen Editor zu öffnen, der einem Connectoreditor ähnelt.

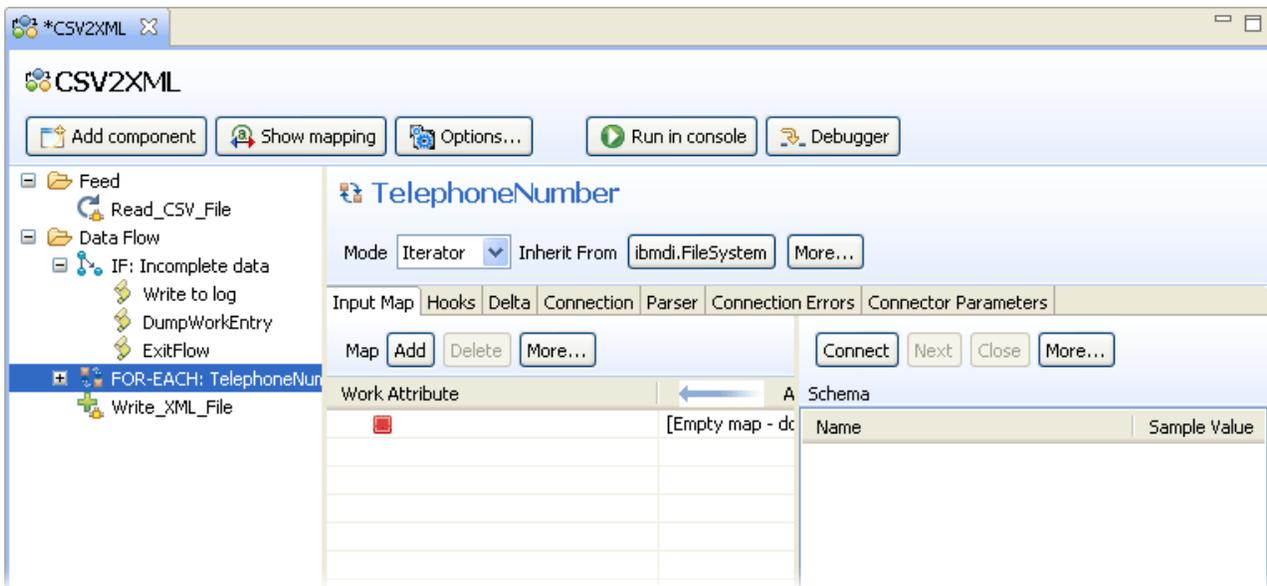


Abbildung 54. Konfiguration der Connectorschleife

Die wichtigsten Unterschiede bestehen darin, dass die Dropdown-Liste **Modus** immer nur die Optionen **Iterator** und **Lookup** enthalten wird. Darüber hinaus ist die Schaltfläche **Mehr...** vorhanden, die Optionen zum Begrenzen der Einträge, die zyklisch verarbeitet werden, sowie den Dropdown-Parameter **Initialisieren** mit folgenden drei Auswahlmöglichkeiten bietet:

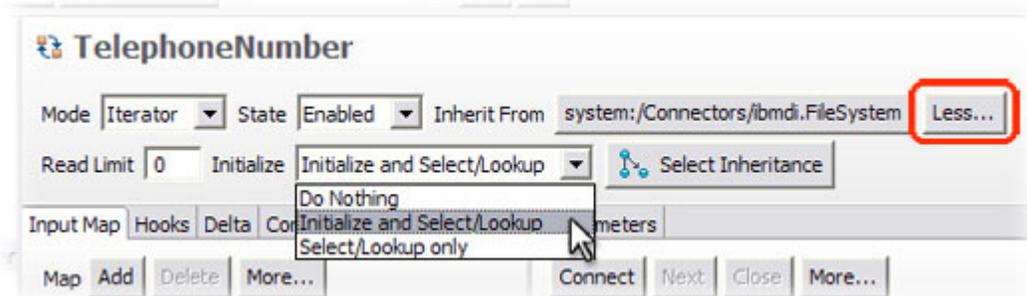


Abbildung 55. Connectorschleife - Erweiterte Einstellungen

- **Keine Aktion ausführen**, dies bedeutet, dass der zugehörige integrierte Connector in keiner Weise initialisiert wird, wenn diese Schleife während der Fertigungslinienverarbeitung erreicht wird.
- **Initialisierung und Auswahl/Suche**, um anzuweisen, dass der Connector stets initialisiert wird, wenn die Schleife mit der zyklischen Verarbeitung beginnt. Verwenden Sie diese Option, da Ihre Connectorschleife aus einer Datei lesen wird, und Sie möchten, dass die Schleife jedes Mal am Anfang startet.
- **Nur Auswahl/Suche**, diese Auswahlmöglichkeit ist nützlich, wenn Ihre Connectorschleife auf eine Datenbank, ein Verzeichnis oder eine andere Datenquelle mit wahlfreiem Zugriff verweist. In diesem Fall ist es nicht erforderlich, die Verbindung jedes Mal neu zu initialisieren. Es muss nur die Suche erneut abgesetzt werden. Im Fall des Iteratormodus ist dies eine Operation *Auswählen* und im Suchmodus eine Operation *Suchen*.

Konfigurieren Sie den Schleifenconnector (der standardmäßig den Typ 'FileSystem' aufweist), damit er die Datei PhoneNumber.xml liest, und wählen Sie anschließend den 'XML-Parser' aus. Rufen Sie nun die Registerkarte 'Attributzuordnung' auf, um Attribute zu erkennen.

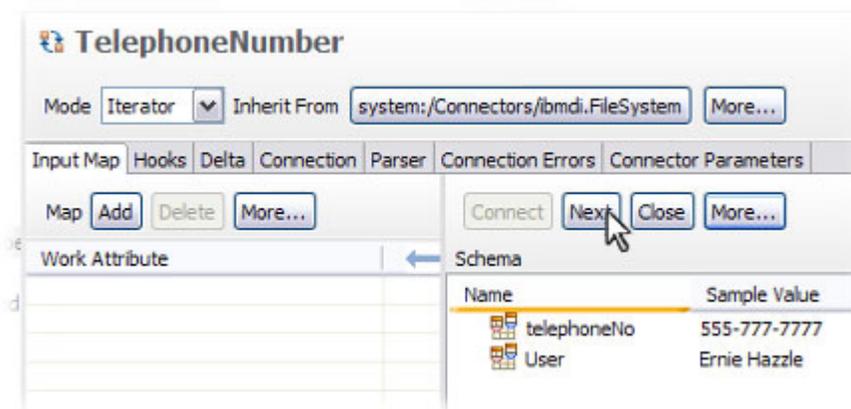


Abbildung 56. Hierarchische Attribute

Sie nehmen Ihre Zuordnung auf Attributebene vor, indem Sie 'User' und 'telephoneNo' im Eingabeschema auswählen und sie in die Eingabezuordnung ziehen.

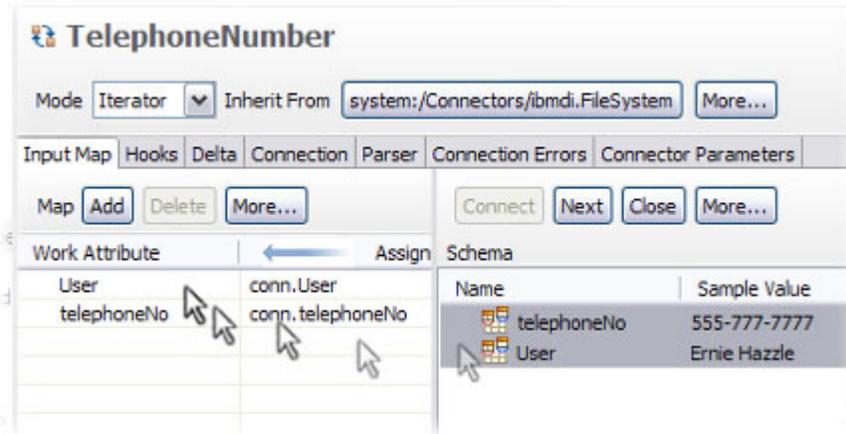


Abbildung 57. Aus Abschnitt 'Schema' in Attributzuordnung ziehen

Dadurch erhalten Sie eine Zuordnungsregel für ein Attribut mit dem Namen "User" und eine Zuordnungsregel für "telephoneNo".

Sie können den Schleifenconnectoreditor nun schließen und anschließend unter ihm eine IF-Verzweigung hinzufügen, indem Sie mit der rechten Maustaste auf die Connectorschleife klicken und die Option **Komponente hinzufügen...** auswählen. Nennen Sie diese IF-Verzweigung 'Matching Name Found_1' (Übereinstimmender Name gefunden). Fügen Sie nun eine einfache Bedingung hinzu, die überprüft, ob Folgendes zutrifft: 'User' ist gleich '\$FullName'.²⁵

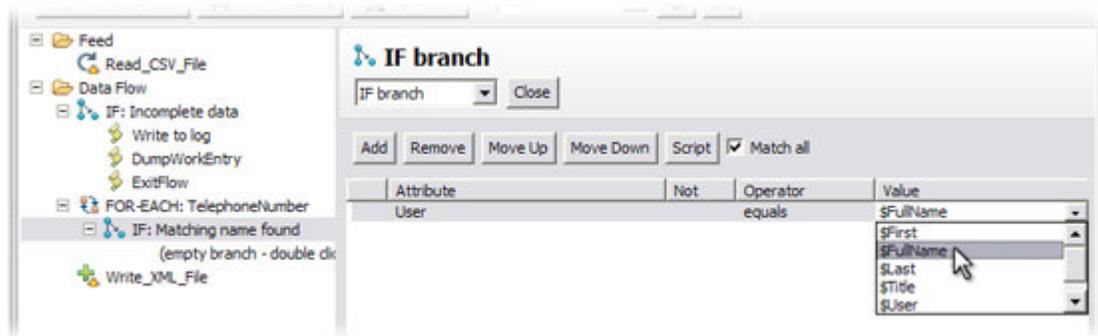


Abbildung 58. Bedingungseditor für IF-Verzweigung

Wenn eine Übereinstimmung gefunden wird, soll die Connectorschleife mit den ordnungsgemäßen Werten in den Attributen 'User' und 'telephoneNo' beendet werden. Damit dies geschieht, fügen Sie eine Scriptkomponente hinzu, die Sie 'Exit loop' (Schleife beenden) nennen und schreiben Sie das folgende Script hinein:

```
system.exitBranch("loop");
```

25. Das Dollarzeichen ist ein Sonderzeichen, das hier anzeigt, dass 'FullName' nicht eine Literalzeichenfolge ist, zu der eine Übereinstimmung gesucht wird, sondern der Wert eines Attributs, das im Work-Eintrag vorhanden ist.

Was geschieht, wenn die Connectorschleife das Ende der Datei PhoneNumbers.xml erreicht, ohne eine Übereinstimmung zu finden? Die Attribute 'User' und 'telephoneNo' enthalten die Werte, die aus dem letzten Eintrag in der Datei gelesen wurden; es hilft also nicht, eine Suche nach leeren Attributen zu starten. Es muss ein anderer Weg gefunden werden, um eine fehlgeschlagene Übereinstimmung zu erkennen.

Die Lösung: Sie verwenden eine Scriptvariable als Markierung, durch die angegeben wird, dass eine Übereinstimmung gefunden wurde. Führen Sie dies nun aus, indem Sie eine Scriptkomponente, die Sie 'foundUser' (Benutzer gefunden) nennen, in die IF-Verzweigung einfügen, sodass sie sich direkt vor der Scriptkomponente 'Exit loop' befindet. Die Scriptkomponente sollte das folgende Scriptfragment enthalten:

```
foundUser = true;
```

Um anzugeben, dass das Ende der Eingabedatei erreicht ist, ohne dass eine Übereinstimmung gefunden wurde, wählen Sie einfach Ihre Connectorschleife aus und öffnen Sie die Registerkarte 'Hooks'.

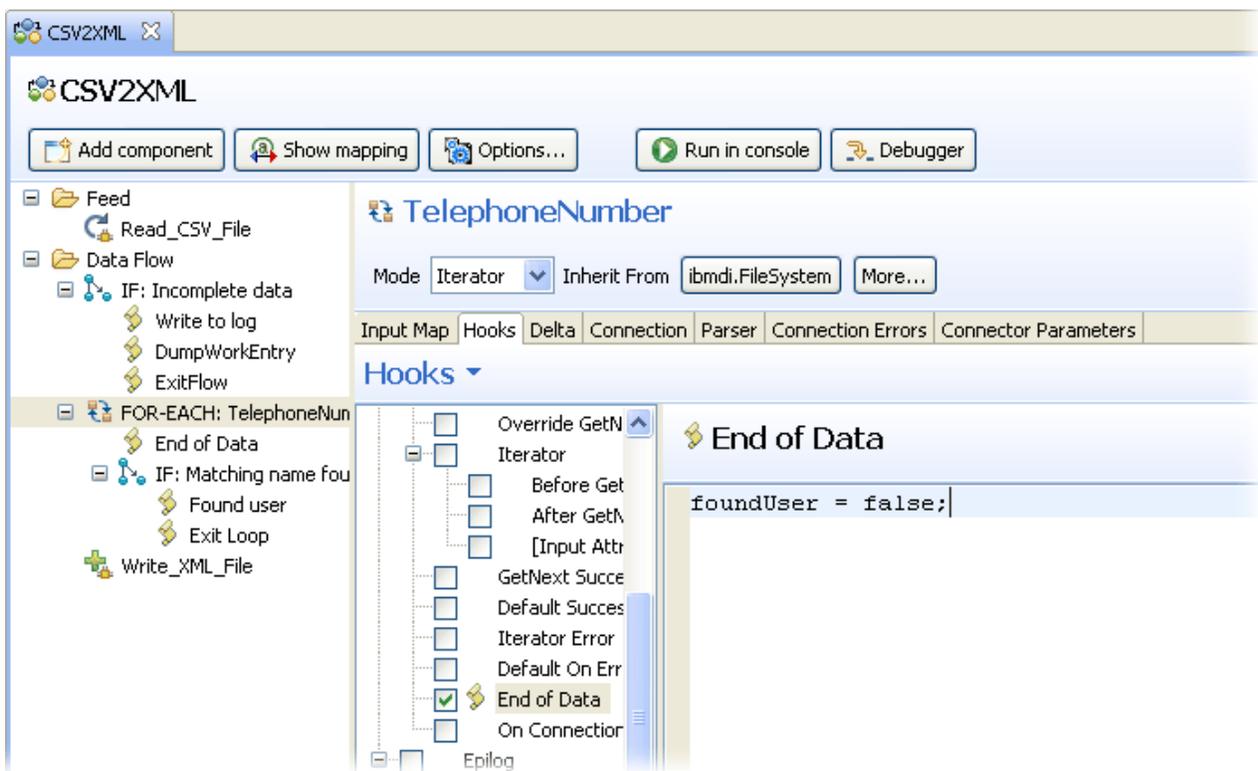


Abbildung 59. Script für Hook 'End of Data' erstellen

Wählen Sie den Hook mit dem Namen 'End of Data' aus und geben Sie dieses Script ein.

```
foundUser = false;
```

Der Hook 'End of Data' wird nur erreicht, wenn der Connector versucht, über den letzten Eintrag in der zugehörigen verbundenen Quelle hinaus zu lesen. In diesem Fall wurde keine Übereinstimmung gefunden.

Ihre Fertigungslinie sollte nun folgende Komponenten aufweisen:

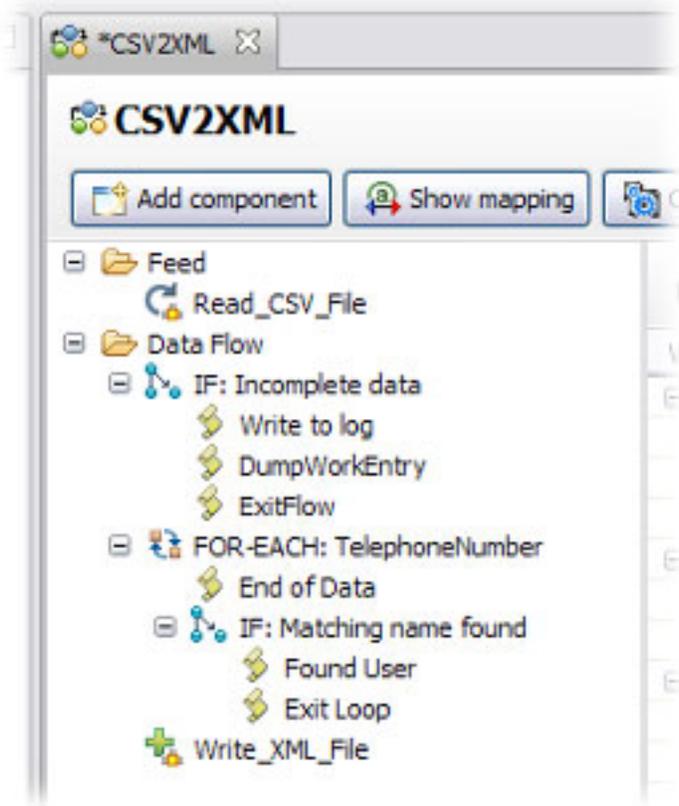


Abbildung 60. Komponentenliste im Abschnitt 'Datenfluss' der Fertigungslinie

Sie sollten nun anhand einer Überprüfung Ihrer Scriptvariable in der Lage sein festzustellen, ob die Suche erfolgreich war. Dies ist wichtig, da Sie jedes Mal, wenn keine Übereinstimmung gefunden wird, auch einen Standardwert für das Attribut 'telephoneNo' festlegen müssen, da andernfalls der letzte Wert dennoch von der Connectorschleife eingelesen wird.

Fügen Sie deshalb direkt auf die Connectorschleife folgend eine weitere IF-Verzweigung ein und nennen Sie sie 'Not found user' (Benutzer nicht gefunden). Klicken Sie auf die Schaltfläche **Script** im Detailbereich der **IF-Verzweigung** und geben Sie dieses Script ein, um den Wert Ihrer Scriptvariable zu überprüfen:

```
! foundUser
```

Das Ausrufezeichen negiert den Wert von 'foundUser', sodass diese Verzweigungsbedingung als *wahr* (true) ausgewertet wird, wenn 'foundUser' im Hook 'End of Data' Ihrer Connectorschleife auf *falsch* (false) gesetzt wurde.

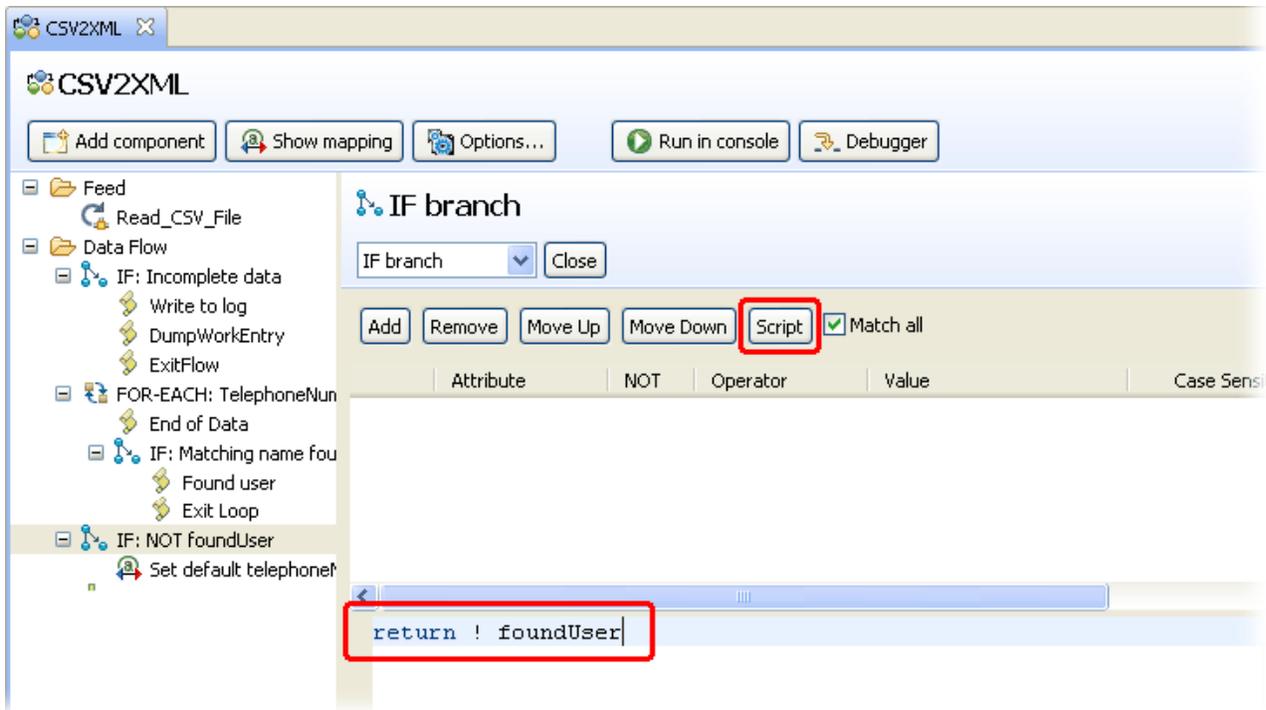


Abbildung 61. Script einer Bedingung für die IF-Verzweigung erstellen

Fügen Sie unter der Verzweigung eine neue Komponente mit dem Typ 'Attributzuordnung' ein. Nennen Sie diese Attributzuordnungskomponente 'Set default telephoneNo', um ihre Funktion im Kontext Ihrer Fertigungslinie deutlich zu machen. Erstellen Sie nun mit der Schaltfläche **Attribut hinzufügen**, ein einzelnes Attribut mit dem Namen 'telephoneNo' - dies ist derselbe Name, der von Ihrer Connector-schleife zurückgegeben wird. Klicken Sie doppelt auf dieses Attribut, um das Zuordnungsscript festzulegen:

```
"N/A"
```

Dies bedeutet, dass bei jeder Person, die aus der CSV-Eingabe eingelesen und in der Datei 'PhoneNumbers.xml' nicht gefunden wurde, für 'telephoneNo' der Wert "N/A" festgelegt wird.²⁶

Schließen Sie dieses neue Attribut 'telephoneNo' zuletzt in die Ausgabezuordnung von 'Write_XML_File' ein, indem Sie es dorthin ziehen und sicherstellen, dass die Zuordnung wie folgt lautet:

```
work.telephoneNo
```

Ihre Fertigungslinie sollte nun wie folgt aussehen:

26. Wenn Sie möchten, dass diese Benutzer kein Attribut 'telephoneNo' aufweisen, verwenden Sie einfach eine Zuordnung, die keinen Wert zurückgibt. Hierzu wird der Sonderwert 'null' zurückgegeben:

```
null
```

Dadurch entfernt das Standardnullverhalten das Attribut aus dem Work-Eintrag. Dadurch erreicht es nicht die Ausgabezuordnung Ihres Connectors 'Write_XML_File' und wird daher nicht im entsprechenden XML-Dokument angezeigt.

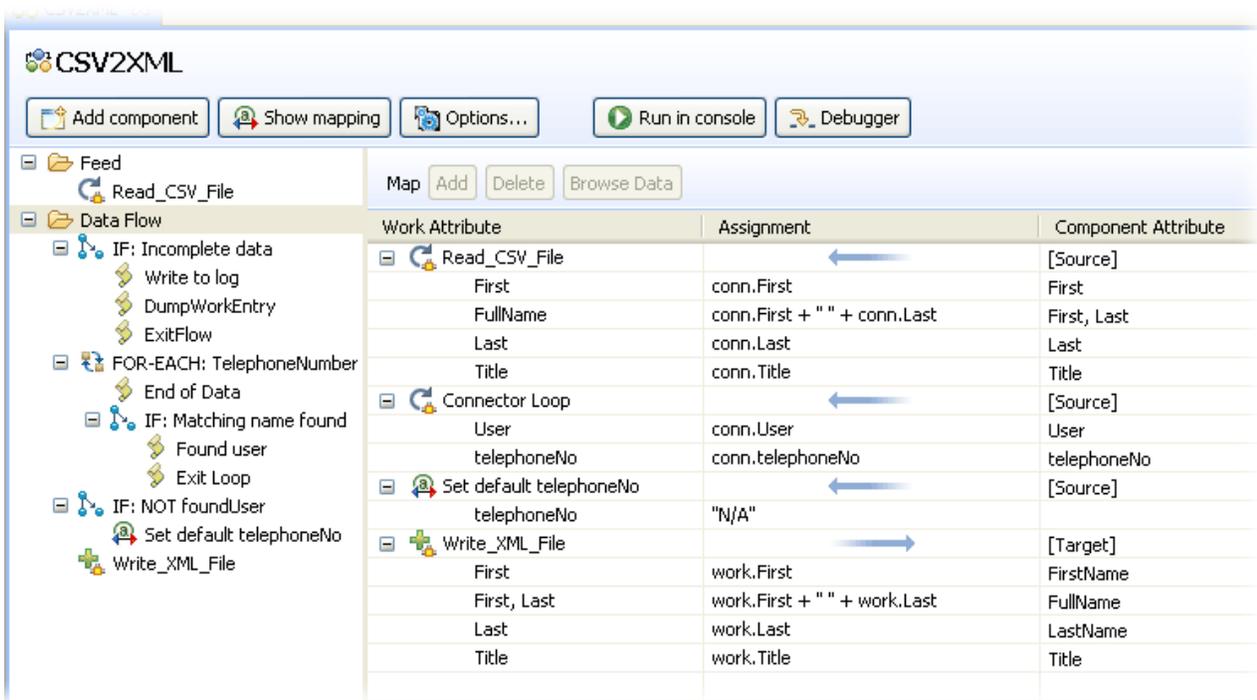


Abbildung 62. Vollständige Fertigungslinie mit FOR-EACH-Schleife

Führen Sie Ihre Fertigungslinie erneut aus und untersuchen Sie die Protokollausgabe. Ihre Verzweigung 'Not found user' sollte zweimal als *wahr* (true) und für die übrigen vier Einträge als *falsch* (falsch) ausgewertet worden sein.

```
16:35:27,281 INFO - CTGDIS003I *** Start dumping Entry
16:35:27,281 INFO - Operation: generic
16:35:27,281 INFO - Entry attributes:
16:35:27,281 INFO - First (replace): 'Roger'
16:35:27,281 INFO - CTGDIS004I *** Finished dumping Entry
16:35:27,328 INFO - CTGDIS088I Finished iterating.
16:35:27,343 INFO - CTGDIS100I Printing the Connector statistics.
16:35:27,343 INFO - [Read_CSV_File] Get:7
16:35:27,343 INFO - [Incomplete data] Branch True:1, Branch False:6
16:35:27,343 INFO - [Write to log] Calls: 1
16:35:27,359 INFO - [DumpWorkEntry] Calls: 1
16:35:27,359 INFO - [ExitFlow] Calls: 1
16:35:27,359 INFO - [TelephoneNumber] Start Loop:6, Loop Cycles:0
16:35:27,359 INFO - [Matching name found] Branch True:0, Branch False:0
16:35:27,359 INFO - [Found User] Calls: 0
16:35:27,359 INFO - [Exit Loop] Calls: 0
16:35:27,359 INFO - [Write_XML_File] Add:6
16:35:27,359 INFO - CTGDIS104I Total: Get:7, Add:6.
16:35:27,359 INFO - CTGDIS101I Finished printing the Connector statistics.
16:35:27,359 INFO - CTGDIS080I Terminated successfully (0 errors).
```

Abbildung 63. Protokollausgabe mit statistischen Daten zur IF-Verzweigung

Beachten Sie, dass einige Komponentennamen in den statistischen Fertigungsliniendaten der Protokollausgabe markiert sind (blau). Wenn Sie bei gedrückter Steuertaste mit der linken Maustaste auf einen der Namen klicken, wird die ausgewählte Komponente im Fertigungslinieneditor geöffnet.

Daraufhin sollte Ihre XML-Ausgabe wie folgt aussehen:

```
-<DocRoot>
- <Entry>
  <FirstName>Bill</FirstName>
  <Title>Chief Scientist</Title>
  <LastName>Sanderman</LastName>
  <telephoneNo>N/A</telephoneNo>
  <FullName>Bill Sanderman</FullName>
</Entry>
- <Entry>
  <FirstName>Mick</FirstName>
  <Title>CEO</Title>
  <LastName>Kamerun</LastName>
  <telephoneNo>555-666-6666</telephoneNo>
  <FullName>Mick Kamerun</FullName>
</Entry>
- <Entry>
  <FirstName>Jill</FirstName>
  <Title>CTO</Title>
```

Abbildung 64. XML-Ausgabe mit dem Attribut 'telephoneNo'

So weit, so gut. Nun wird der Suchmodus *Lookup* zum Erstellen der Verknüpfung verwendet.

Modus 'Lookup' verwenden

Anstatt Ihre aktive Fertigungslinie zu modifizieren, wobei Ihnen möglicherweise ein Fehler unterläuft, machen Sie eine Kopie der Fertigungslinie und ändern statt des Original die Kopie. Klicken Sie dazu mit der rechten Maustaste auf die Fertigungslinie 'CSV2XML' und wählen Sie **Kopieren** aus.

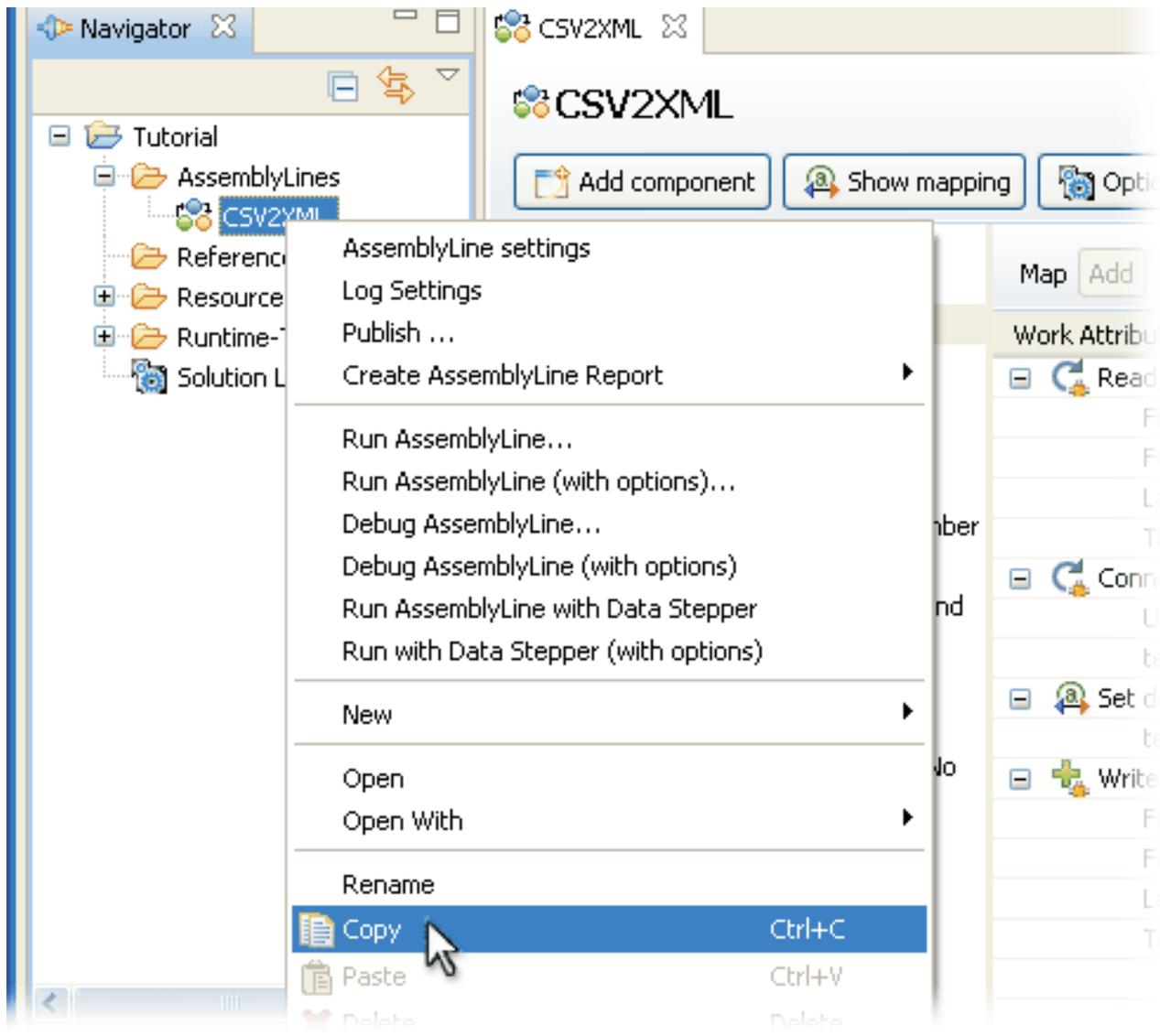


Abbildung 65. Kopierfunktion der Fertigungslinie

Klicken Sie nun mit der rechten Maustaste auf den Ordner 'Fertigungslinien' im Navigator und wählen Sie die Option **Einfügen** aus. Nennen Sie diese neue Fertigungslinie 'CSV2XML_LookupMode' und klicken Sie anschließend doppelt auf sie, um den Fertigungslinieneditor zu öffnen.

Sie können nun die Connectorschleife einschließlich aller untergeordneten Komponenten entfernen. Wählen Sie sie einfach aus und drücken Sie auf die Taste zum **Löschen**. An die freigewordene Position ziehen Sie einen JDBC-Connector, den Sie aus einer anderen Fertigungslinie kopieren.

Doch zunächst müssen Sie die Datenbanktabelle erstellen, aus der dieser Connector lesen soll oder, besser gesagt, Sie müssen eine vordefinierte Fertigungslinie ausführen, die die Tabelle für Sie erstellt. Navigieren Sie dazu mit einem Dateibrowser zu dem Ordner 'Tutorial' und suchen Sie nach der Datei mit dem Namen Create-PhoneDB.assemblyline. Ziehen Sie sie in das Konfigurationseditorfenster in den Ordner 'Fertigungslinien' in der Navigatoranzeige.

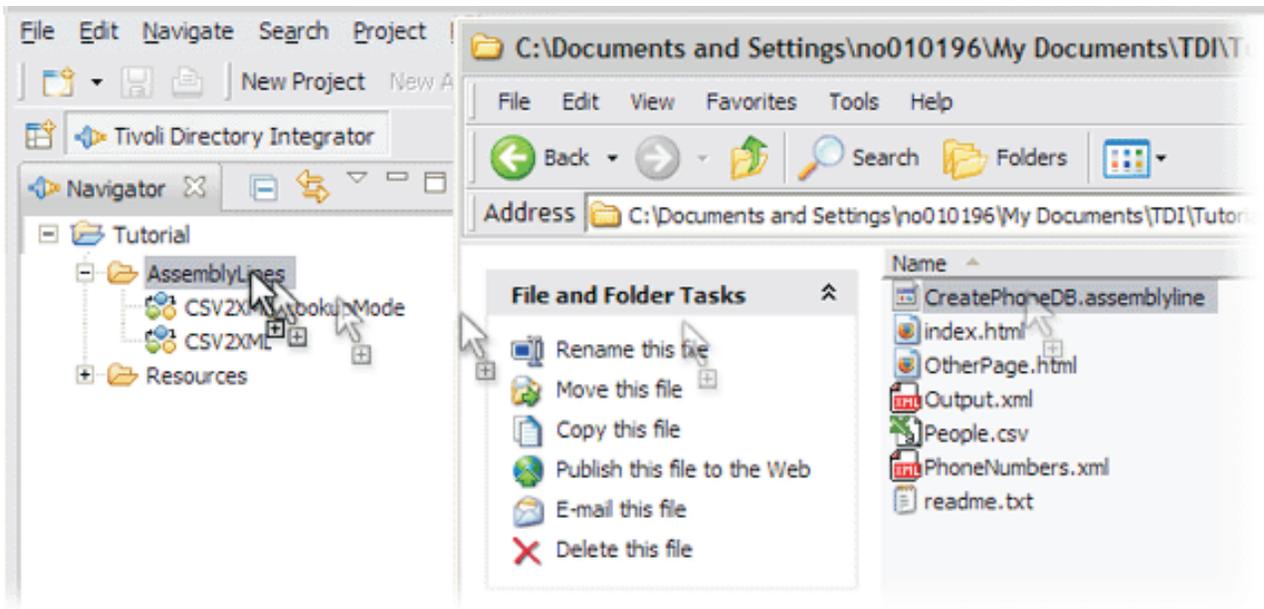


Abbildung 66. Fertigungslinie in Ihr Projekt kopieren

Die Fertigungslinie wird in Ihr Projekt importiert. Klicken Sie nun auf das Projekt und wählen Sie **Fertigungslinie ausführen...** (Run AssemblyLine...) aus.

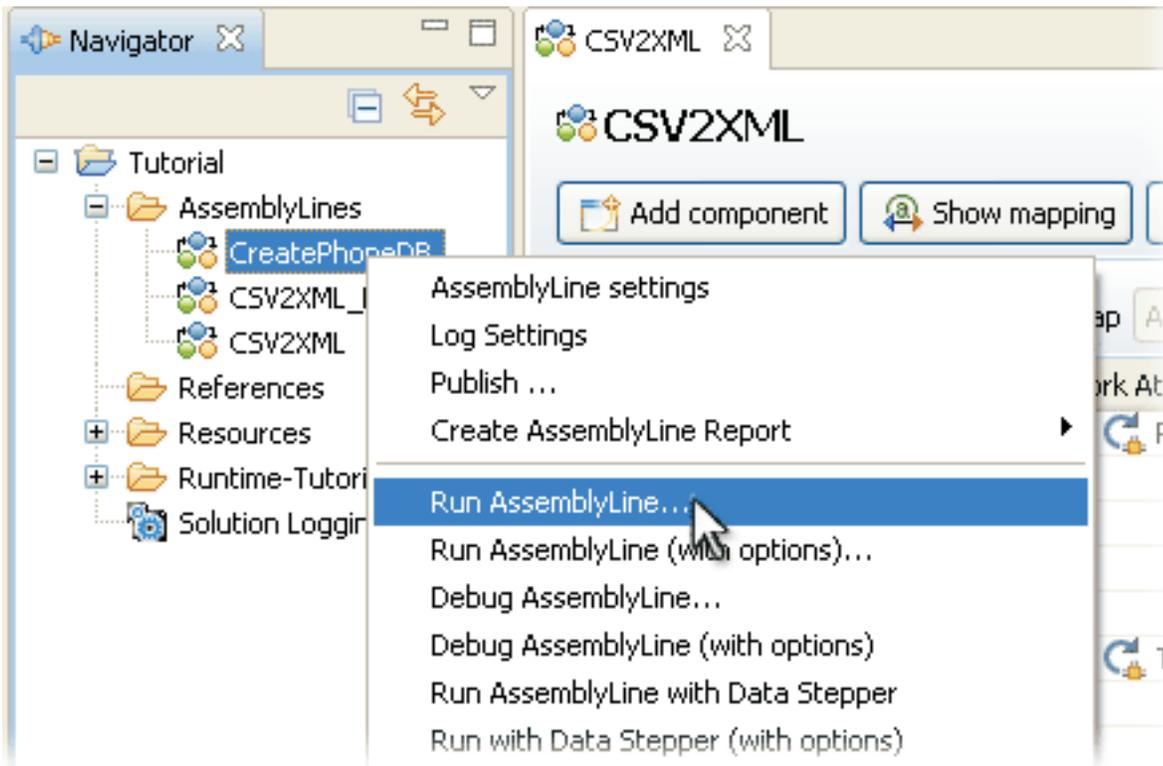
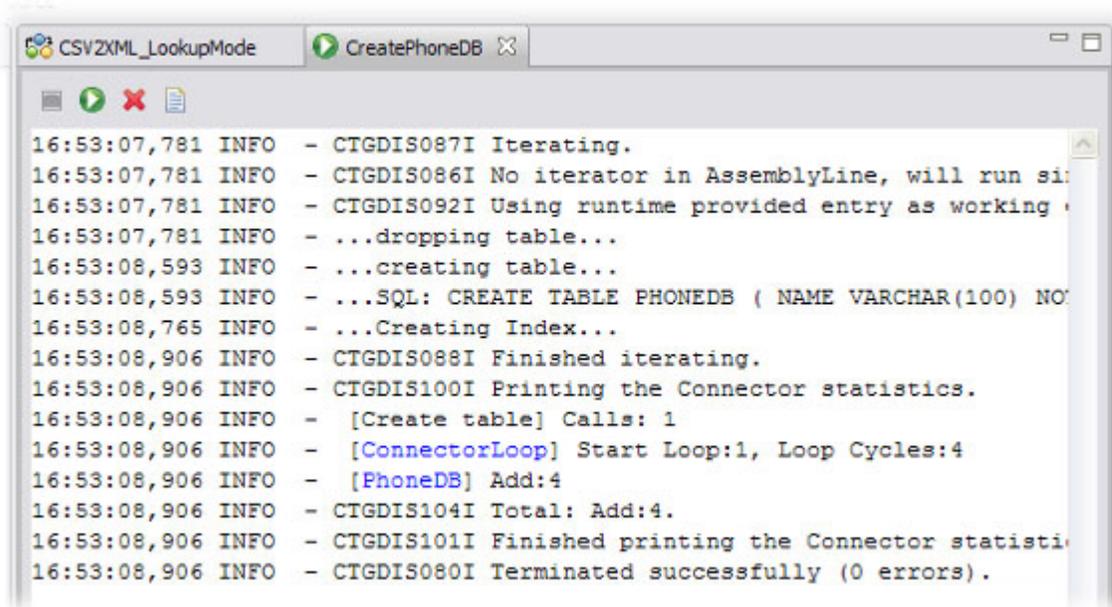


Abbildung 67. Fertigungslinie 'CreatePhoneDB.assemblyline' ausführen

Diese Fertigungslinie erstellt zunächst eine Derby-Datenbank²⁷ mit dem Namen 'TutorialDB' (Lernprogrammdatenbank) unter Ihrem Lösungsverzeichnis und konfiguriert dann die Tabelle 'PhoneDB'. Sie läuft anschließend in einer Schleife durch PhoneNumbers.xml und lädt diese Informationen in die neue Tabelle.²⁸

Bei Erfolg sollte die Protokollausgabe wie folgt aussehen:



```
16:53:07,781 INFO - CTGDIS087I Iterating.
16:53:07,781 INFO - CTGDIS086I No iterator in AssemblyLine, will run si
16:53:07,781 INFO - CTGDIS092I Using runtime provided entry as working
16:53:07,781 INFO - ...dropping table...
16:53:08,593 INFO - ...creating table...
16:53:08,593 INFO - ...SQL: CREATE TABLE PHONEDB ( NAME VARCHAR(100) NO
16:53:08,765 INFO - ...Creating Index...
16:53:08,906 INFO - CTGDIS088I Finished iterating.
16:53:08,906 INFO - CTGDIS100I Printing the Connector statistics.
16:53:08,906 INFO - [Create table] Calls: 1
16:53:08,906 INFO - [ConnectorLoop] Start Loop:1, Loop Cycles:4
16:53:08,906 INFO - [PhoneDB] Add:4
16:53:08,906 INFO - CTGDIS104I Total: Add:4.
16:53:08,906 INFO - CTGDIS101I Finished printing the Connector statisti
16:53:08,906 INFO - CTGDIS080I Terminated successfully (0 errors).
```

Abbildung 68. Protokollausgabe der Fertigungslinie 'CreatePhoneDB'

Die Fertigungslinie 'CreatePhoneDB' weist einen JDBC-Connector auf, der bereits konfiguriert und betriebsfähig ist. Sie kopieren diese Komponente in Ihre Projektressourcenbibliothek (genauer gesagt, in den Ordner "Connector") und verwenden sie dann in Ihrer Fertigungslinie wieder.

Öffnen Sie die Fertigungslinie 'CreatePhoneDB', klicken Sie auf den Connector mit dem Namen 'PhoneDB' und ziehen Sie ihn in den Ordner 'Connector', der sich unter 'Ressourcen' in der Baumstrukturansicht des Navigators befindet.

27. Apache Derby ist eine relationale Open-Source-Datenbank, die im Produktpaket von IBM Security Directory Integrator enthalten ist.

28. Obwohl die Fertigungslinie in diesem Handbuch nicht behandelt wird, ist dies ein gutes Beispiel für erweiterte Scripterstellungsverfahren, die verwendet werden, um die datenquellenspezifische Funktionalität zu nutzen, die in den meisten Connectorschnittstellen vorhanden ist. Untersuchen Sie diese Fertigungslinie und erkunden Sie sie spielerisch.

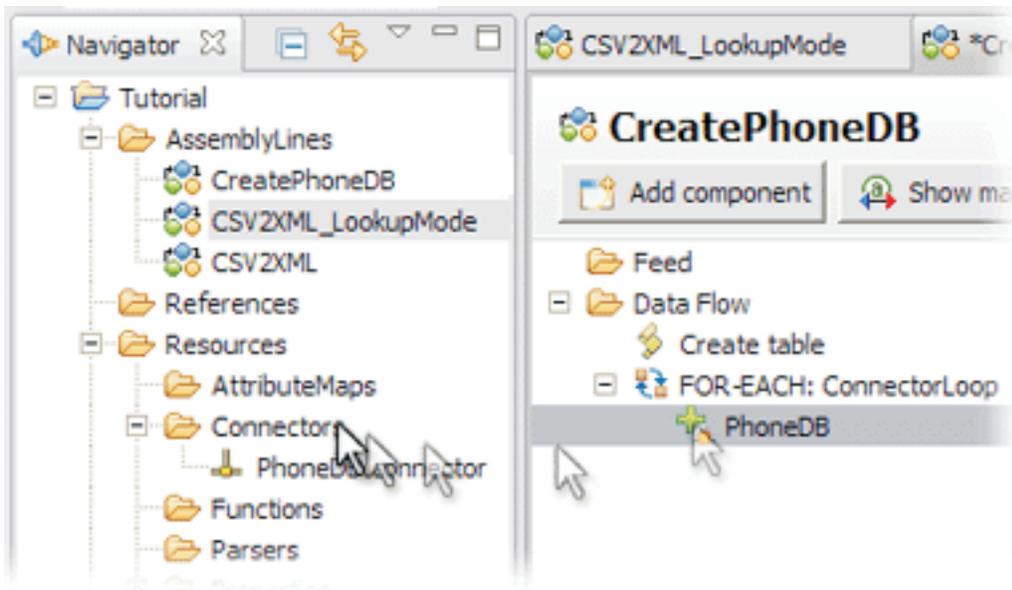


Abbildung 69. Connector in 'Ressourcen' ziehen

Schließen Sie 'CreatePhoneDB', sodass wieder Ihre eigene Fertigungslinie angezeigt wird. Ziehen Sie anschließend die neue Ressource 'PhoneDBConnector' an die Position, die zuvor von der Connectorschleife eingenommen wurde.

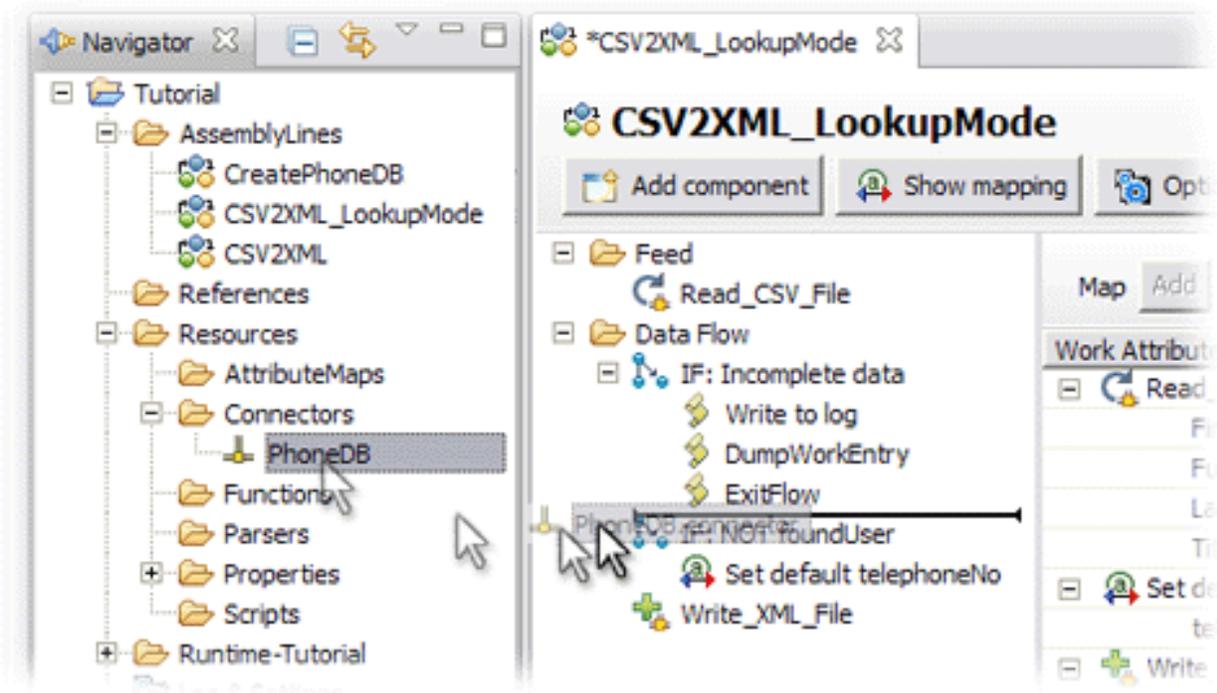


Abbildung 70. Neue Ressource in Ihre Fertigungslinie ziehen

Vererbung

Ist Ihnen aufgefallen, dass dieser Connector in Ihrer Fertigungslinie in blau angezeigt wird? Er wird blau angezeigt, weil zurzeit ein Vererbungsvorgang zwischen dem Connector und Ihrer Ressourcenbibliothek stattfindet. Dies bedeutet, dass der Connector Konfigurationseinstellungen zur Laufzeit dynamisch von dem Connector abrufen, den Sie mit der Maus gezogen haben. Diese Vererbungsfunktion erleichtert die Wiederverwendung von Ressourcen wie konfigurierten Komponenten und scriptgesteuerter Logik über mehrere Fertigungslinien hinweg.

Sie können das übergeordnete Element einer Komponente mit der Schaltfläche **Übernehmen von** ändern, die sich oben in der zugehörigen Editoranzeige befindet. Die Registerkarten der Komponente wie Konfiguration, Delta, Eingabe-/Ausgabebezuordnung und Hooks bieten ebenfalls eine Vererbungsoption, mit der sie ein anderes übergeordnetes Element festlegen können, als es die Komponente aufweist.

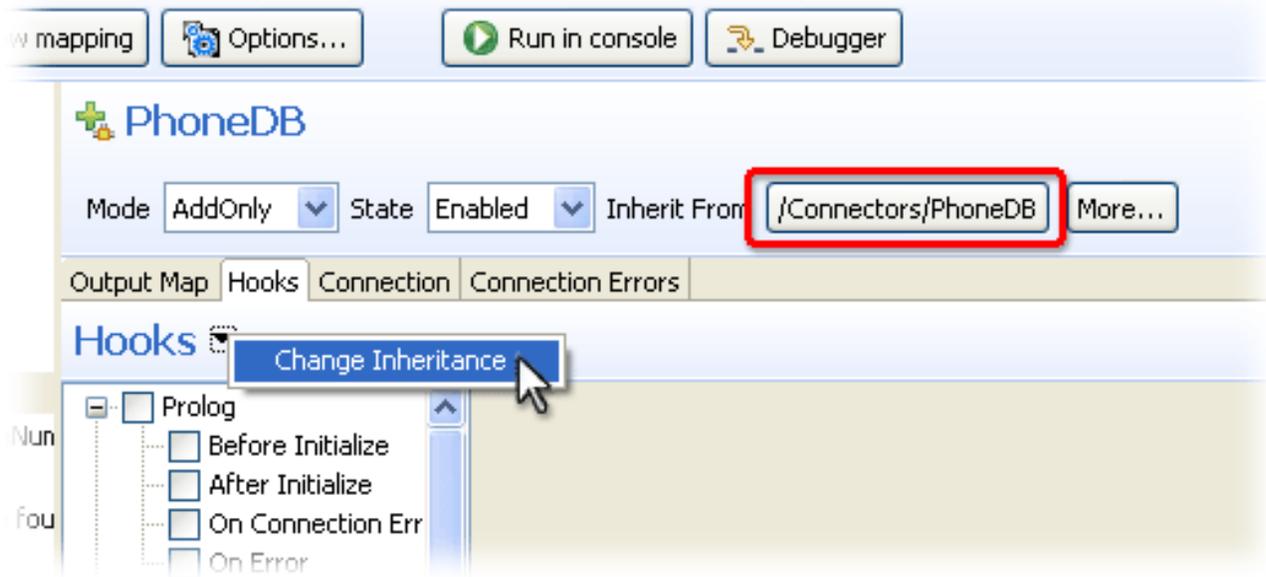


Abbildung 71. Vererbung für die Registerkarte 'Hooks' festlegen

Übernommene Werte werden in blauer Schrift angezeigt. Wenn Sie die Farbe ändern, wird die Vererbung unterbrochen. Die Vererbung wird wiederhergestellt, wenn die Option **Auf übernommenen Wert zurücksetzen** in den Kontextmenüs der Attributzuordnungsregeln und der Hooks ausgewählt wird.

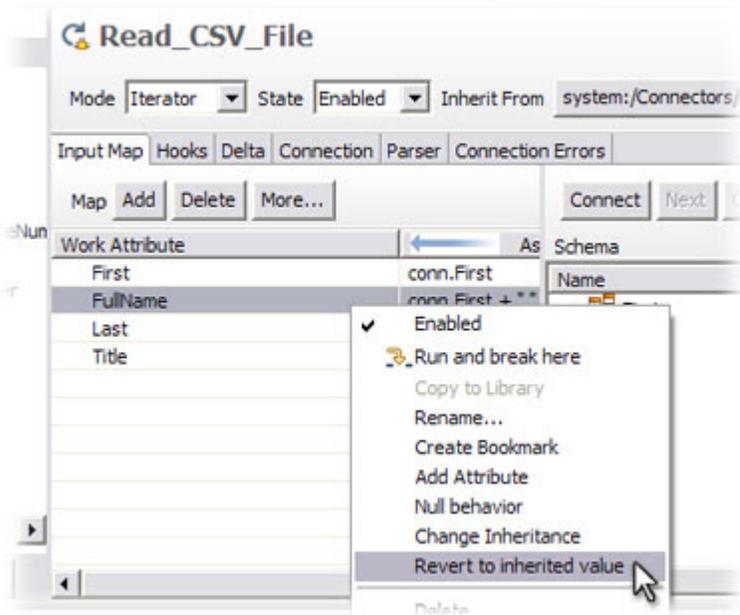


Abbildung 72. Vererbung für eine Zuordnungsregel wiederherstellen

Kehren Sie wieder zur Übung zurück und ändern Sie zuerst die Einstellung für den **Modus** des neuen Connectors "PhoneDB" von **AddOnly** in **Lookup**.

Da die Attributzuordnung ursprünglich eine dem vorhergehenden Modus zugeordnete *Ausgabebezuordnung* war, müssen Sie als nächsten Schritt das Eingabeschema ermitteln, indem Sie auf die Schaltflächen **Verbindung herstellen** (Connect) und **Nächster** (Next) klicken, die sich oberhalb des Connectorschemas befinden. Im dritten und letzten Schritt ziehen Sie das Attribut 'PHONE' aus 'Schema' in die Eingabebezuordnung (Input Map), wodurch Sie eine einfache Zuordnung für diesen Wert erhalten.

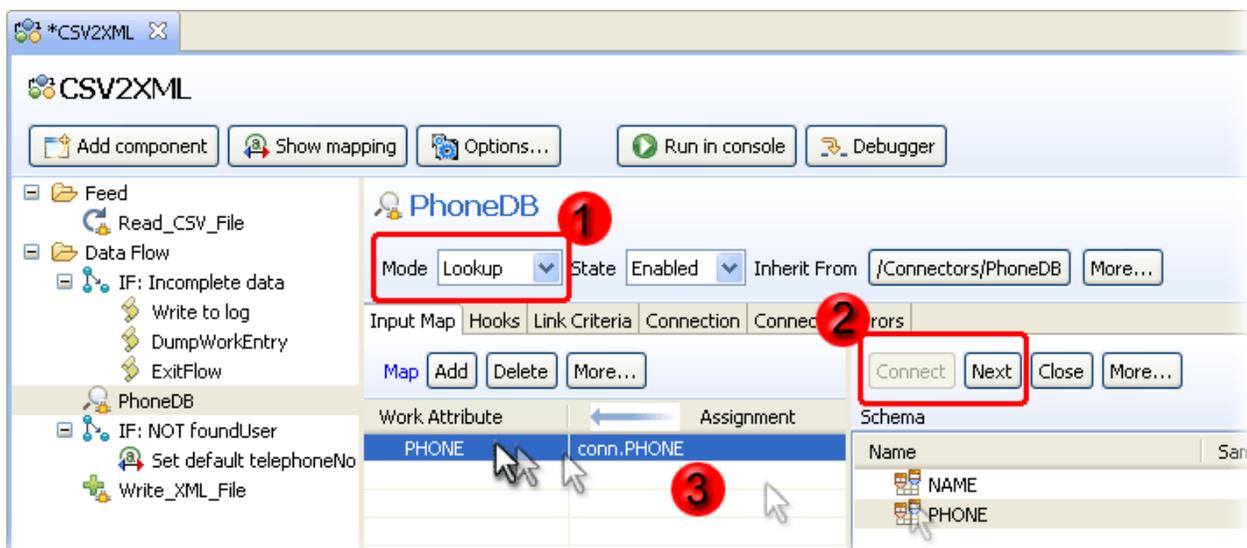


Abbildung 73. Modus ändern, Attribute erkennen und zuordnen

Sobald Sie die Eingabezuordnungsregel erstellt haben, ist es wichtig, dass Sie sie von 'PHONE' in 'telephoneNo' umbenennen, sodass sie mit der Ausgabezuordnung Ihres Connectors 'Write_XML_File' übereinstimmt.

Vielleicht ist Ihnen aufgefallen, dass Sie 'NAME' nicht zugeordnet haben. Dies ist nicht erforderlich, da Sie über den Namen des Benutzers bereits verfügen. Dieses Feld wird stattdessen bei der Definition der Suchregel verwendet.

Regeln für die Suche mit 'Lookup' sind Verknüpfungsbedingungen

Als Sie den Suchmodus *Lookup* ausgewählt haben, wurde im Connectoreditor eine neue Registerkarte mit der Bezeichnung 'Verknüpfungsbedingungen' angezeigt. Verknüpfungsbedingungen werden verwendet, um die Abgleichsregeln für die Suche zu definieren.

Sie können Abgleichsregeln als einfache Verknüpfungsbedingungen definieren, indem Sie die Dropdown-Listen verwenden, neue Verknüpfungsbedingungen wie benötigt hinzufügen und das Kontrollkästchen **Beliebige Übereinstimmung** (Match Any) auswählen, wie Sie es für Bedingungen tun würden. Alternativ können Sie das Kontrollkästchen **Bedingungen mit angepasstem Script erstellen** (Build criteria with custom script) auswählen und stattdessen ein Scriptfragment schreiben, das die aktuelle Suchregel berechnet, wie dieses Beispiel eines LDAP-Suchfilters zeigt:

```
"(cn=" + work.FullName + ")"
```

Beachten Sie, dass Ihre Lösung dadurch stärker an die zu durchsuchende Datenquelle gebunden wird, da Sie die tatsächliche Syntax schreiben müssen, die vom verbundenen System erwartet wird. In Ihrem Fall würde dies bedeuten, dass eine Klausel WHERE erstellt werden muss (ohne das Schlüsselwort 'WHERE' selbst).

Im Gegensatz dazu werden einfache Verknüpfungsbedingungen vom Connector für Sie in native Suchsyntax umgesetzt, sodass Sie die Connectorschnittstelle wechseln können, ohne Ihre Verknüpfungsbedingungen erneut zu erstellen.

Einfache Verknüpfungsbedingungen sehen Bedingungen ähnlich. Die erste Dropdown-Liste wird mit dem Schema gefüllt, das Sie erkannt haben, und die zweite zeigt an, welche Work-Eintrag-Attribute zu diesem Zeitpunkt in Ihrer Fertigungslinie verfügbar sind. Genauso wie bei Bedingungen wird auch hier das Dollarzeichen verwendet, um anzuzeigen, dass der Wert des benannten Attributs zur Laufzeit ersetzt werden sollte, um den Suchfilter zu erstellen.

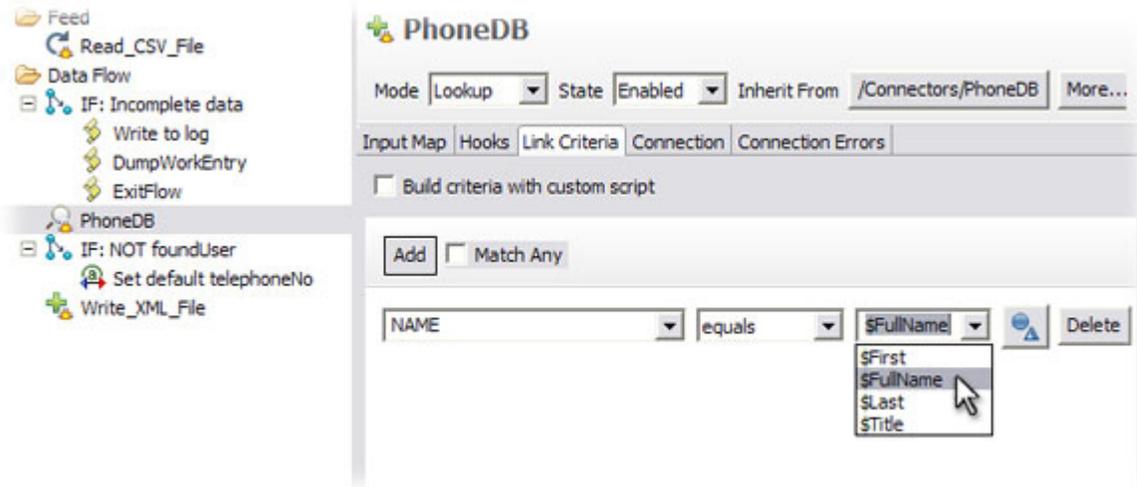


Abbildung 74. Eine einfache Verknüpfungsbedingung

Vergessen Sie nicht, Ihre Arbeit zu speichern, z. B. durch Drücken der Tastenkombination Strg-S.²⁹ Führen Sie dies regelmäßig durch, damit Sie Ihre Arbeiten nicht verlieren.

Sie müssen nun Ihre Fertigungslinie erneut **Ausführen**.

Ausführungsfehler entschlüsseln

Bearbeiten Sie Fehler in Ruhe. Der Stackauszug, der in der Protokollausgabe angezeigt wird, gibt an, dass ein Fehler aufgetreten ist. Blättern Sie zum Anfang des allerersten Stackauszugs. Hier werden Informationen dazu angezeigt, *wo* das Problem aufgetreten ist und durch *was* es verursacht wurde.

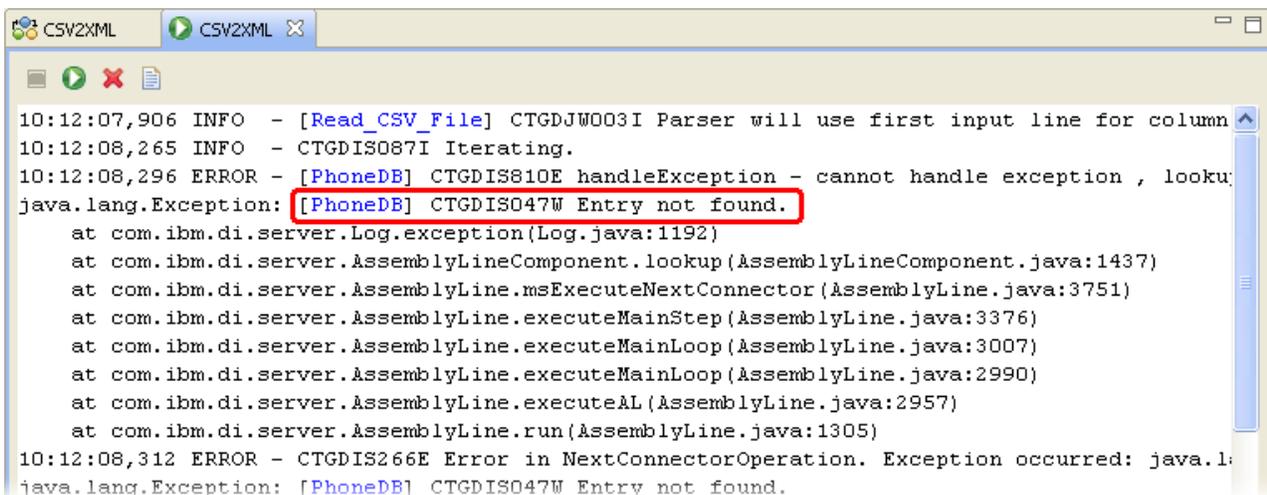


Abbildung 75. Fehlernachricht in der Protokollausgabe

29. Wenn Sie Fertigungslinien oder Ressourcen gelöscht haben und diese wiederherstellen wollen, klicken Sie mit der rechten Maustaste auf Ihr Projekt in der Navigatoranzeige und wählen Sie die Option für die Wiederherstellung aus dem lokalen Protokoll aus. Daraufhin wird eine Liste mit Ressourcenversionen angezeigt, mit denen Sie die Wiederherstellung durchführen können. Sie müssen natürlich zunächst etwas speichern, bevor es in Ihrem Protokoll angezeigt werden kann.

Der Komponentename steht in eckigen Klammern ('PhoneDB'). Darauf folgt die Fehlerbeschreibung, es sei kein Eintrag gefunden worden; die Suche ist also fehlgeschlagen.

Wenn ein Connector mit dem Modus *Lookup* konfiguriert ist, erwartet das System, dass bei Ausführen einer Suche nur eine einzige übereinstimmende Aufzeichnung gefunden wird. Wenn keine Aufzeichnung gefunden wird oder wenn mehrere Aufzeichnungen mit den Verknüpfungsbedingungen übereinstimmen, gelangen Sie zu speziellen *Hooks*, die zumindest aktiviert sein müssen, damit ein Stoppen der Fertigungslinie verhindert wird. Dieses Verhalten ist in den Datenflussdiagrammen deutlich zu sehen, die in der Veröffentlichung *Referenzinformationen* enthalten sind. Hier ein Auszug aus der Seite, in der der Suchmodus *Lookup* detailliert erläutert wird:

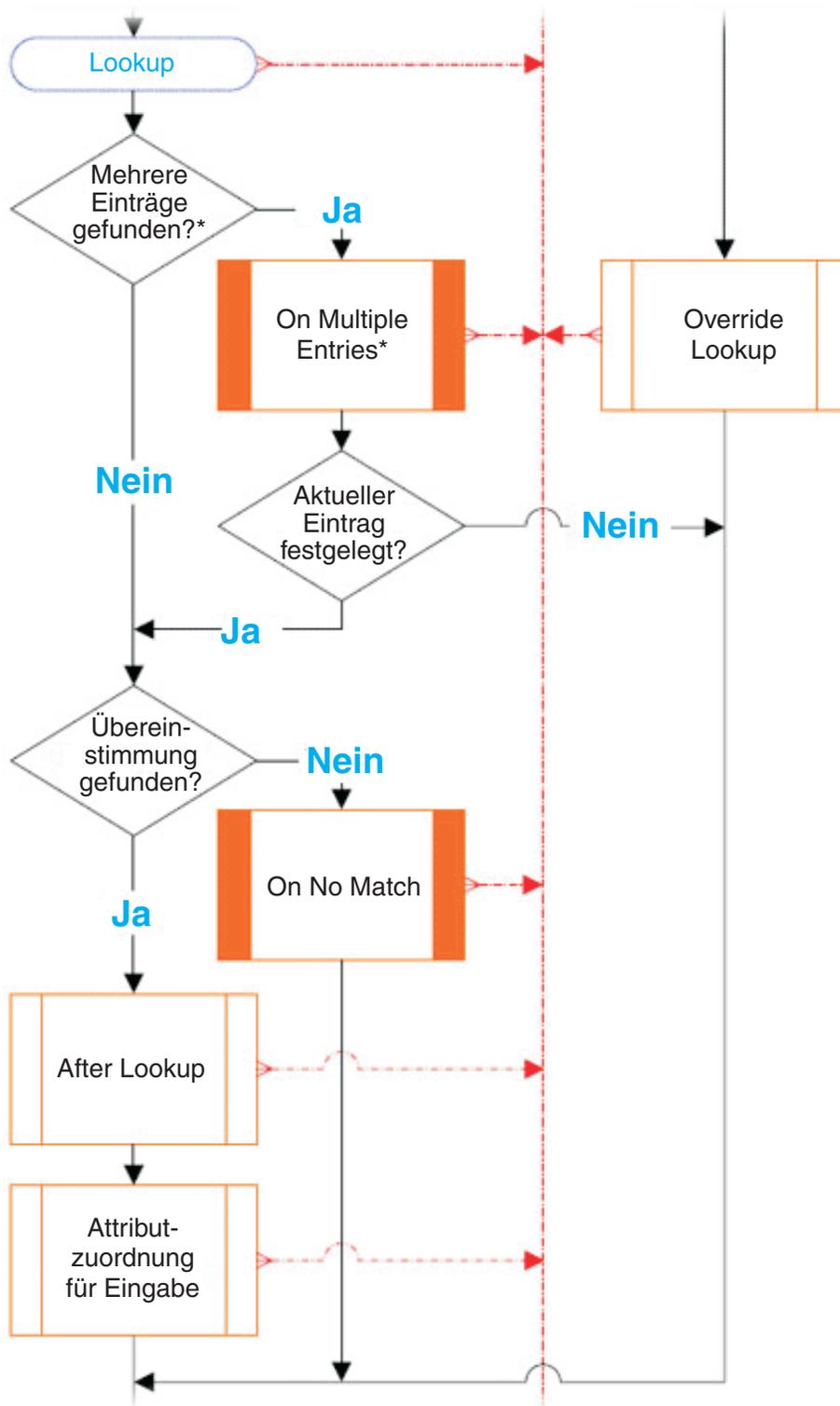


Abbildung 76. Teil des Flussdiagramms für den Suchmodus 'Lookup'

Sie können sich dieses Verhalten zunutze machen, um Ihre Markierungsvariable `foundUser` festzulegen. Klicken Sie mit der rechten Maustaste auf den Connector 'PhoneDB' und wählen Sie **Hooks** aus, um den Editor für Hooks zu öffnen. Wählen Sie den Hook 'On No Match' aus und geben Sie den Scriptcode ein, um `foundUser` auf *falsch* (false) zu setzen.

```
foundUser = false;
```

Wählen Sie nun 'After Lookup' aus und geben Sie dieses zusätzliche Hookscript ein:³⁰

```
foundUser = true;
```

Ihre Fertigungslinie sollte nun so aussehen wie die Fertigungslinie in diesem Screenshot:

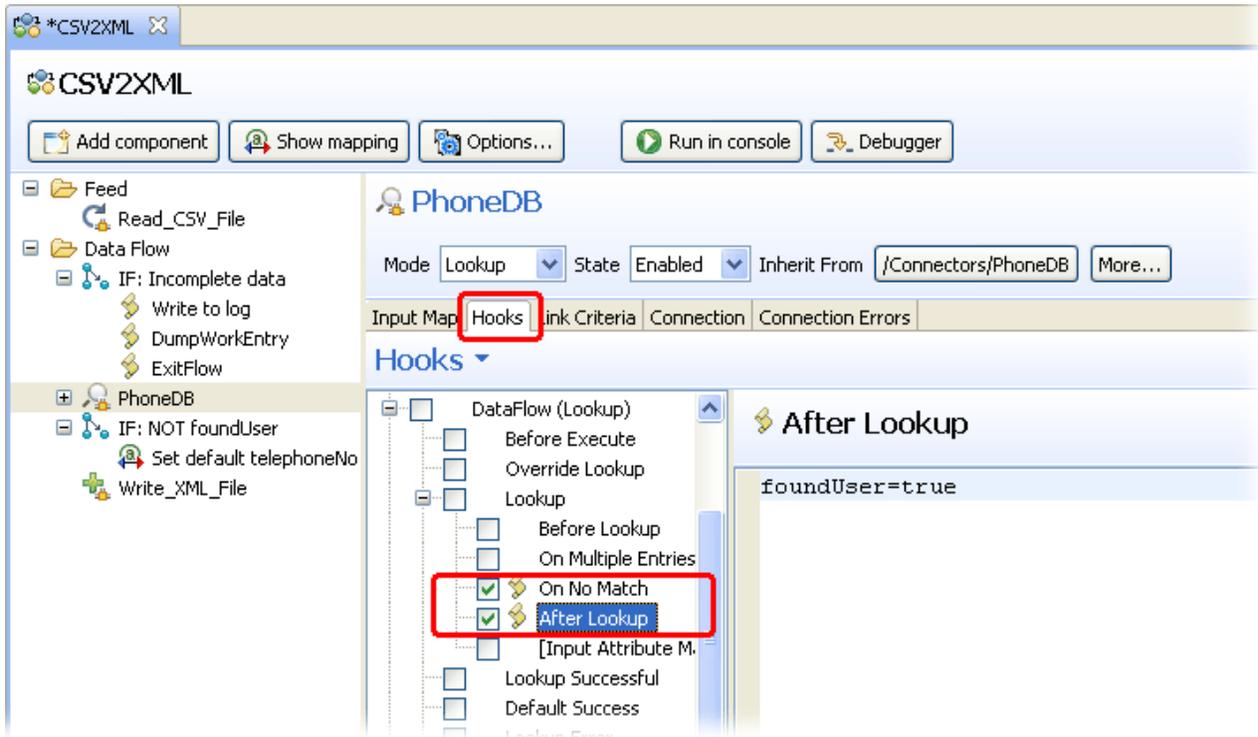


Abbildung 77. Erste Lernprogrammübung abgeschlossen

Sie können diese Fertigungslinie nun erneut **ausführen** und diesmal sollte sie ohne Fehler beendet werden. Die Datei `Output.xml` sollte mit den Ergebnissen identisch sein, die Sie von Ihrer schleifenbasierten Fertigungslinie erhalten haben.

Herzlichen Glückwunsch! Sie haben Ihre erste IBM Security Directory Integrator-Lernprogrammübung abgeschlossen. Im Folgenden wird erklärt, wie Echtzeitergebnisse zum Auslösen von Fertigungslinien verwendet werden.

30. Wie Sie im Flussdiagramm sehen können, wird der Hook 'After Lookup' nur ausgeführt, wenn das Ergebnis der Suche eine einzige Übereinstimmung ist.

Kapitel 3. Ereignisgesteuerte Integration

Bisher haben Sie Ihre Fertigungslinien als Stapelprozesse ausgeführt und sie jedes Mal, wenn Sie Daten bewegen wollten, manuell gestartet. Sie können Fertigungslinien auch von der Befehlszeile aus ausführen, indem Sie den IBM Security Directory Integrator-Server wie folgt aufrufen³¹:

```
ibmdisrv -c examples/Tutorial/Tutorial1.xml -r CSVtoXML
```

Auf diese Weise ist es eine leichte Aufgabe, Planungstools zu verwenden (beispielsweise *crontab*), um Operationen der Fertigungslinie zu planen oder um sie leicht von externen Anwendungen aus zu starten.

IBM Security Directory Integrator bietet eine Reihe von Funktionen, mithilfe derer Ihre Fertigungslinien ereignissensitiv werden, wodurch Ihre Lösungen ein weites Spektrum von Echtzeittriggern handhaben und auf sie reagieren können.

Beispiele für diese Trigger:

- Protokollanfragen, die über einen IP-Port eingehen, wie REST-Aufrufe, SNMP- und Web-Services.
- Neue Nachrichten, die in einer Warteschlange angezeigt werden.
- E-Mails, die in einem Maileingang eintreffen.
- Änderungen an Daten, z. B. in Dateien, Datenbanken, Verzeichnissen und Notes-Datenbanken.
- Fertigungslinienoperationen, die auf dem Scheduler oder dem Zeitgeber basieren

Diese Liste ist nicht vollständig und Sie finden eine Anleitung und Anregungen in weiteren IBM Security Directory Integrator-Dokumentationen, auf den Community-Websites und in Newsgroups.

Es gibt verschiedene Möglichkeiten, diese Ereignisse in Ihrer Lösung zu handhaben:

Connector im Iteratormodus

Einige Connector ermöglichen es Ihnen, Zeitlimitparameter für den Iteratormodus zu konfigurieren. Ein Beispiel dafür ist der Dateisystemconnector, der so konfiguriert werden kann, dass er eine Datei bis zum Dateiende liest und dann auf neue Informationen wartet - ein so genannter Tail-Lesevorgang.

Andere Connector, wie beispielsweise Connector für die RDBS-Änderungserkennung und LDAP-Änderungsprotokolle, haben eine ähnliche Funktionsweise. Die Connector ermöglichen die Erstellung von Fertigungslinien, die ohne Unterbrechung ausgeführt werden und darauf warten, dass im verbundenen System neue Änderungen auftreten.

Darüber hinaus steht der Zeitgeberconnector zur Verfügung, der im Iteratormodus ausgeführt wird und so konfiguriert werden kann, dass er in zeitlich gesteuerten Intervallen, die durch einen Planungsparameter festgelegt werden, die Fertigungslinie ausführt. Sie werden diesen in Kürze testen.

31. Der IBM Security Directory Integrator-Server stellt einen Verwendungshinweis zur Verfügung, wenn er ohne Befehlszeilenargumente aufgerufen wird: `ibmdisrv`

IBM Security Directory Integrator beinhaltet eine Webadministrationkonsole, die zur Standardinstallation gehört. Diese browserbasierte Anwendung wird Administration and Monitoring Console (AMC) genannt und Sie können mit ihr den Zustand Ihrer Fertigungslinien überwachen, Konfigurationen sofort auf aktive Server laden, Fertigungslinien starten und stoppen und das Fehler- und das Antwortverhalten konfigurieren, sodass Ihre Integrationslösungen hoch verfügbar bleiben. Sie kann auch verwendet werden, um Zeitpläne für die Ausführung Ihrer Fertigungslinien zu erstellen. Die Behandlung des Webverwaltungstools würde jedoch den Rahmen dieses Handbuchs sprengen.

Anmerkung: Die AMC-Funktion ist veraltet und wird in einer zukünftigen Version von IBM Security Directory Integrator entfernt.

Connector im Servermodus

Einige spezialisierte Connector, wie beispielsweise der HTTP-Serverconnector und der LDAP-Serverconnector, ermöglichen es Ihnen, Lösungen zu entwickeln, die eingehende Anforderungen von externen Clients verarbeiten, angeforderte Aktionen ausführen und entsprechende Antworten geben. In der nächsten Übung verwenden Sie den HTTP-Serverconnector.

Benachrichtigungen und Merkmale

IBM Security Directory Integrator enthält Komponenten, die IBM Security Directory Integrator-Benachrichtigungsereignisse empfangen können, sowie Komponenten (und Scriptaufrufe) zum Senden dieser Ereignisse - sogar zwischen individuellen IBM Security Directory Integrator-Servern, die auf verschiedenen Plattformen ausgeführt werden.

In diesem Handbuch werden die Zeitplanung für Fertigungslinien und der Connectormodus 'Server' noch näher betrachtet.

Zeitplanung für Fertigungslinien

Über die Benutzerschnittstelle des Konfigurationseditors können Sie einen Scheduler für die Ausführung einer Fertigungslinie zu bestimmten Zeitpunkten erstellen. So können Sie beispielsweise einen Scheduler erstellen, mit dem eine Ausführung täglich um 3:05 Uhr geplant wird oder eine Ausführung jeden Samstag um 7:00 Uhr. Auch die Erstellung komplexerer Zeitpläne ist möglich.

Zum Erstellen eines Schedulers klicken Sie im Konfigurationseditor auf **Datei > Neu > Scheduler**. Alternativ dazu können Sie auch mit der rechten Maustaste auf die Fertigungslinie klicken und **Zeitplan erstellen** auswählen.

Wenn der IBM Security Directory Integrator-Server eine Konfigurationsdatei lädt, werden die aktivierten Scheduler, die in der Konfigurationsdatei enthalten sind, automatisch gestartet. Der Befehl zum Laden einer Konfigurationsdatei lautet `ibmdisrv -c myconfig.xml -d`. Dabei ist `myconfig.xml` die exportierte Konfigurationsdatei. Das Stoppen der Konfigurationsinstanz bewirkt auch das Stoppen aller zugehörigen Scheduler.

Scheduler

Type
 Keepalive Timer Enabled

AssemblyLine/Sequence *

Do not start if AL is already running

Start Times

Month
 Every Specific

Day
 Every Day in Month Day In Week

Hour
 Every Specific

Minutes after the hour

Seconds after the full minute

▶ **Advanced**

Abbildung 78. IBM Security Directory Integrator Scheduler

Details zum IBM Security Directory Integrator Scheduler sind im entsprechenden Abschnitt der Veröffentlichung *Referenzinformationen* enthalten.

Fertigungslinien für Serviceanforderung

In dieser letzten Übung erstellen Sie eine Web-Server-Fertigungslinie, um eine sehr einfache Benutzerschnittstelle zum Starten Ihrer Fertigungslinie 'CSV2XML_LookupMode' bereitzustellen, mit anderen Worten, einen HTTP-Service zum Initiieren von Datentransfers.

Erstellen Sie zunächst eine neue Fertigungslinie und nennen Sie sie 'TINA_WebServer'.³² Fügen Sie anschließend eine neue Komponente ein, indem Sie 'HTTP_Server_Connector' auswählen und auf **Fertigstellen** klicken, um den Assistenten zu beenden. Öffnen Sie nun die zugehörige Registerkarte **Eingabezuordnung**.

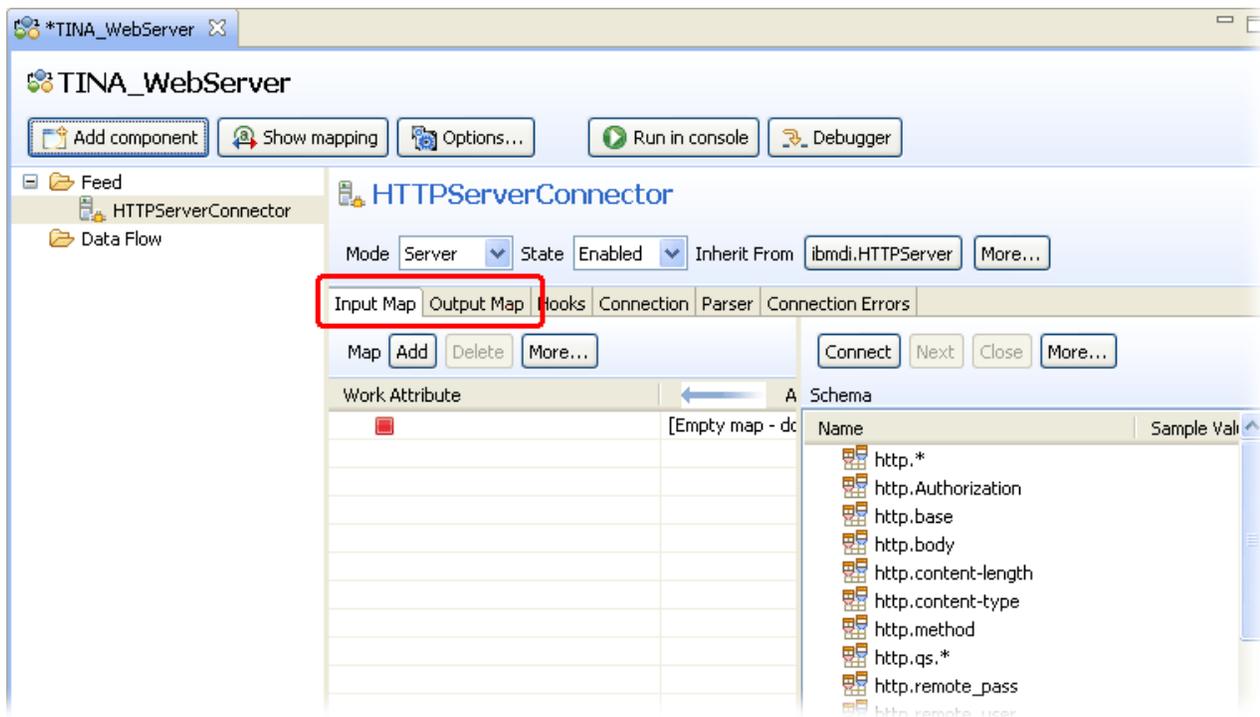


Abbildung 79. Attributzuordnungsanzeige von 'HTTP_Server_Connector'

Server-Connectors sind eine Ergänzung zu Funktionskomponenten. Während Funktionskomponenten Serviceanforderungen erstellen, stellt ein Server-Connector einen Service bereit und ist die Basis für die Ausführung eines Service. Dies führt dazu, dass die Eingabezuordnung für einen Server-Connector verwendet wird, um Attribute zu empfangen, die von dem Client stammen, der eine Anforderung vornimmt, während die Ausgabezuordnung die Methode für eine Antwort bereitstellt. Sie sehen auch im rechten Teil der Attributzuordnungsanzeige, dass das Eingabeschema bereits vorhanden ist. Dasselbe trifft auf das Ausgabeschema zu. Server-Connector stellen diese Informationen bereit, um Sie bei Ihrer Zuordnung zu unterstützen. Beachten Sie jedoch, dass einige dieser Schemaattribute Platzhalterzeichen wie 'http.qs.*' enthalten. Dies sind entwurfszeitrelevante Informationen, die Ihnen mitteilen, dass eine beliebige Anzahl von eingehenden Attributen zu erwarten sind, deren Namen mit 'http.qs.' beginnen.³³

32. TINA steht hier für "TINA ist nicht Apache".

33. Diese bestimmte Attributgruppe (http.qs.*) überträgt alle Abfragezeichenfolgeparameter, die vom Client an den HTTP-Aufruf übergeben werden.

Die Dropdown-Liste **Modus** für alle Server-Connector bietet Server- und Iteratormodus. Mit einem zusätzlichen Modus (Response), der hier nicht angezeigt ist, sind es insgesamt drei Modi. Der Server-Connector schaltet während seiner Operation zu verschiedenen Zeitpunkten zwischen diesen Modi um:

1. Ein Server-Connector beginnt zunächst im Modus *Server*, stellt wie ein IP-Port eine Verbindung zu einer Ressource her und wartet auf eingehende Clientverbindungen.
2. Sobald eine Verbindung hergestellt ist, wechselt der Connector in den Modus *Iterator*, um basierend auf der Eingabezuordnung Clientdaten abzurufen, die er an die Datenflusskomponenten übergibt.
3. Wenn die *Datenflusskomponenten* die Ausführung abgeschlossen haben, wechselt der Connector schließlich in den Modus 'Response' (Antwort) und verwendet die Ausgabezuordnung, um eine Antwort an den Client zu erstellen.

Dies wird automatisch ausgeführt, es ist keine Benutzeraktion erforderlich. Wenn Sie jedoch Scripts für Hooks erstellen, werden Sie bemerken, dass drei verschiedene Arten von Hooks vorhanden sind: *Server*, *Iterator* und *Response*.

Fahren Sie mit der Lernprogrammübung fort, indem Sie der Eingabezuordnung ein Attributelement hinzufügen.

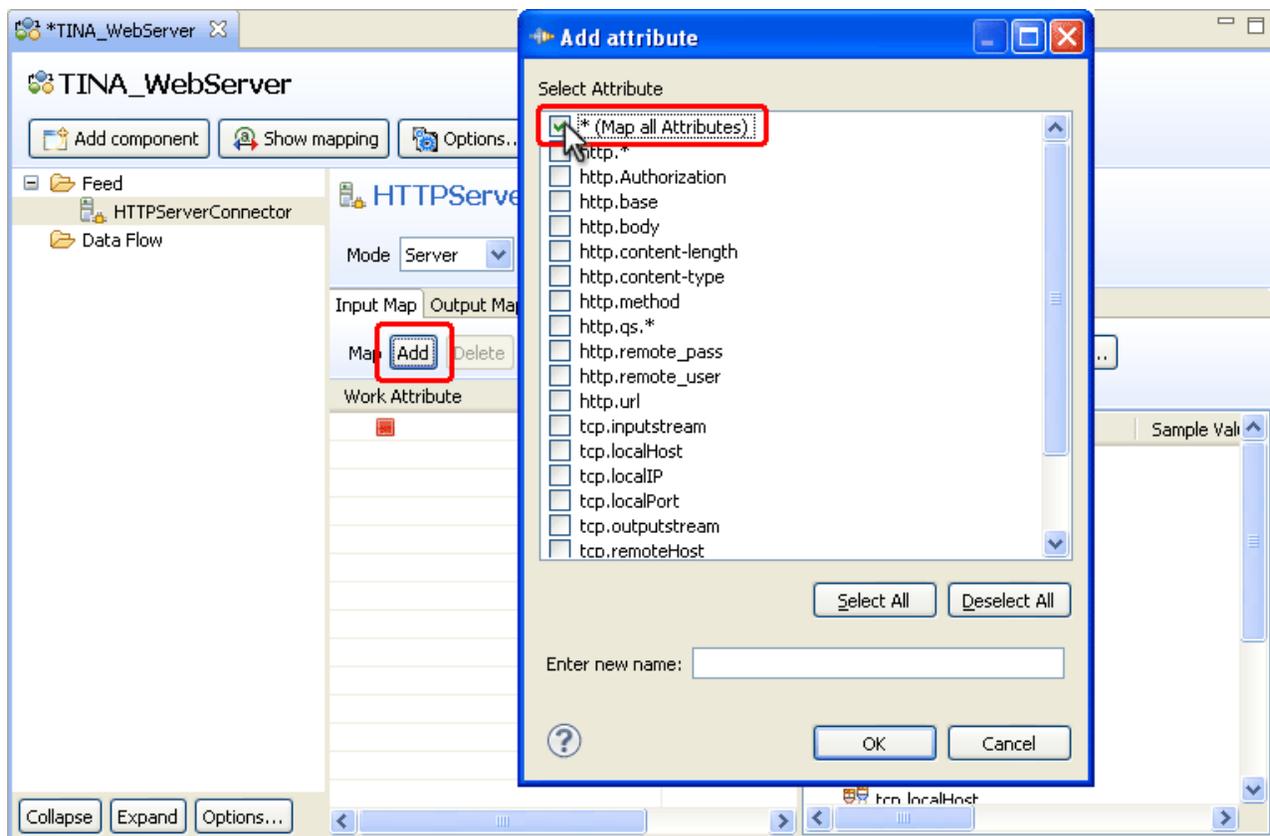


Abbildung 80. Attributelement zur Eingabezuordnung hinzufügen

Wählen Sie einfach die oberste Option im angezeigten Dialogfenster aus. Sie hätten im Eingabefeld für einen neuen Namens auch den Stern (*) eingeben können. Dies ist die besondere Platzhalterzeichenzuordnungsregel, die bewirkt, dass IBM Security Directory Integrator in allen Attributen, die vom Connector gelesen werden, eine Zuordnung vornimmt.

Ihre Eingabezuordnung sieht wie folgt aus:

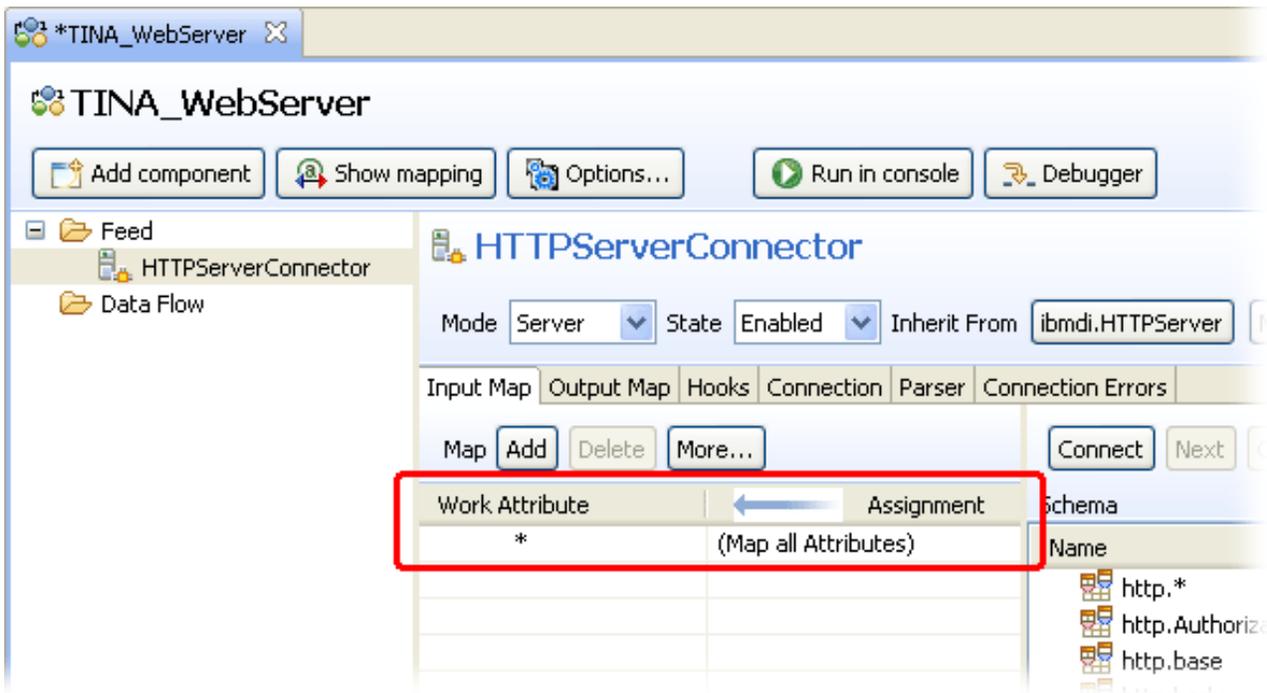


Abbildung 81. Platzhalterelement für die Zuordnung

Fügen Sie auch der Ausgabezuordnung ein Platzhalterelement für die Zuordnung hinzu, um sicherzustellen, dass alle Attribute, die im Work-Eintrag für die Antwortnachricht festgelegt sind, wiederum dem Client zugeordnet werden.

Um diese Komponente zu testen, platzieren Sie die Scriptkomponente 'Speicherausgang für Work_Eintrag' in den Abschnitt 'Datenfluss' und klicken Sie anschließend auf **Ausführen**, um die Fertigungslinie auszuführen. Die Protokollausgabe sollte eine Nachricht mit dem Inhalt anzeigen, dass Ihr HTTP-Server-Connector an Port 80 empfangsbereit für HTTP-Verbindungen ist.³⁴ Dies bedeutet, dass Ihre Fertigungslinie darauf wartet, dass Clients Verbindungen herstellen. Sie können dies nun ausführen, indem Sie einen Browser öffnen und zu der folgenden URL navigieren:

`http://localhost:80`

Sehen Sie sich nun Ihre Protokollausgabe an. Sie zeigt eine Anzahl von TCP- und HTTP-Headereigenschaften an, die als Attribute zurückgegeben wurden.

34. Sollte dieser Port bereits verwendet werden, öffnen Sie einfach die Konfigurationsanzeige für 'HTTP_Server_Connector' und wählen Sie einen anderen Port aus.

```

12:54:38,165 INFO - [HTTPServerConnector] CTGDIS498I Using provided parameter
12:54:38,181 INFO - CTGDIS003I *** Start dumping Entry
12:54:38,197 INFO - Operation: generic
12:54:38,197 INFO - Entry attributes:
12:54:38,197 INFO - tcp.localPort (replace): '80'
12:54:38,197 INFO - http.Cookie (replace): 'JSESSIONID=0000MzTM5PekFp'
12:54:38,197 INFO - tcp.outputstream (replace): 'java.net.SocketOutput
12:54:38,212 INFO - http.Accept-Language (replace): 'en-us,en;q=0.5'
12:54:38,212 INFO - tcp.localHost (replace): 'localhost'
12:54:38,212 INFO - http.Accept-Charset (replace): 'ISO-8859-1,utf-8;
12:54:38,212 INFO - tcp.inputstream (replace): 'java.net.SocketInputS
12:54:38,212 INFO - http.base (replace): '/'
12:54:38,212 INFO - http.Accept (replace): 'text/html,application/xht
12:54:38,212 INFO - tcp.localIP (replace): '127.0.0.1'
12:54:38,212 INFO - http.Accept-Encoding (replace): 'gzip,deflate'
12:54:38,212 INFO - tcp.remotePort (replace): '1993'
12:54:38,212 INFO - http.User-Agent (replace): 'Mozilla/5.0 (Windows;
12:54:38,212 INFO - tcp.remoteHost (replace): 'localhost'
12:54:38,212 INFO - http.method (replace): 'GET'
12:54:38,212 INFO - http.Host (replace): 'localhost:80'
12:54:38,212 INFO - http.Connection (replace): 'keep-alive'
12:54:38,212 INFO - http.url (replace): '/'
12:54:38,228 INFO - http.Keep-Alive (replace): '300'
12:54:38,228 INFO - tcp.remoteIP (replace): '127.0.0.1'
12:54:38,228 INFO - tcp.socket (replace): 'Socket[addr=localhost/127
12:54:38,228 INFO - CTGDIS004I *** Finished dumping Entry
12:54:38,322 INFO - CTGDIS003I *** Start dumping Entry

```

Abbildung 82. Als Attribute zurückgegebene TCP- und HTTP-Headereigenschaften

Das einzige für Sie in dieser Übung interessante Attribut ist 'http.base', das den Teil der URL enthält, der nach dem Host und dem Socket angezeigt wird. Insbesondere werden Sie danach suchen, weil dort der Text 'RunAL' (Fertigungslinie ausführen) enthalten sein muss.

Um zu überprüfen, ob dieser Text vorhanden ist, fügen Sie dem Abschnitt 'Datenfluss' Ihrer Fertigungslinie eine IF-Verzweigung hinzu. Nennen Sie sie 'RunAL detected' (Fertigungslinie erkannt) und legen Sie als Bedingung Folgendes fest: `http.base contains (enthält) 'RunAL'`.

Wenn diese Verzweigungsbedingung als *wahr* ausgewertet wird, müssen Sie Ihre Fertigungslinie 'CSV2XML_LookupMode' starten. Dazu verwenden Sie die Funktionskomponente der Fertigungslinie (AssemblyLine Function Component) aus Ihrer Fertigungslinie 'Scheduler' wieder, indem Sie sie zuerst in **Ressourcen > Funktionen** in der Navigatoranzeige und anschließend zurück in diese Fertigungslinie ziehen und sie in der neuen IF-Verzweigung übergeben.

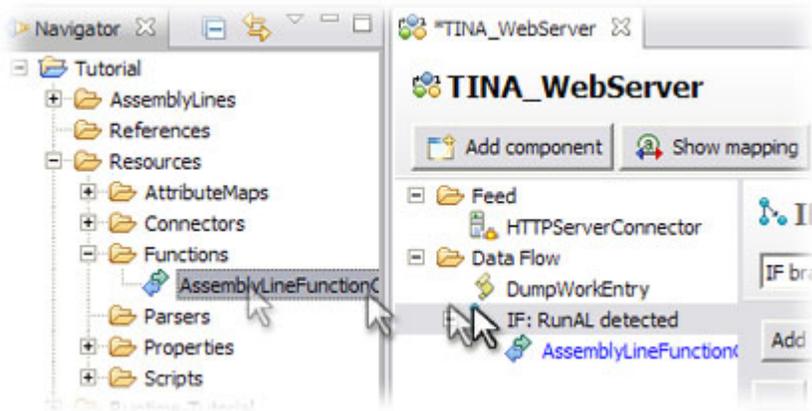


Abbildung 83. Funktionskomponente der Fertigungslinie (AssemblyLine Function Component) ziehen

Führen Sie nun Ihre Fertigungslinie erneut aus und geben Sie folgenden Text in das Adressfeld Ihres Browsers ein:

`http://localhost/RunAL`

Nun wird der Speicherauszug für Work-Eintrag in der Protokollausgabe angezeigt, gefolgt von den statistischen Daten zur aufgerufenen Fertigungslinie.

```

TINA_WebServer x
- http.Base (replace): /RunAL
- http.Accept (replace): 'text/html,application/xhtml+xml,application/xml;q=0.9,
- tcp.localIP (replace): '127.0.0.1'
- http.Accept-Encoding (replace): 'gzip,deflate'
- tcp.remotePort (replace): '2132'
- http.User-Agent (replace): 'Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv
- tcp.remoteHost (replace): 'localhost'
- http.method (replace): 'GET'
- http.Host (replace): 'localhost'
- http.Connection (replace): 'keep-alive'
- http.url (replace): '/RunAL'
- http.Keep-Alive (replace): '300'
- tcp.remoteIP (replace): '127.0.0.1'
- tcp.socket (replace): 'Socket[addr=localhost/127.0.0.1,port=2132,localport=80
- CTGDIS004I *** Finished dumping Entry
- [AssemblyLineFunctionComponent] CTGDIS255I AssemblyLine AssemblyLines/CSV2XML_LookupMo
- [AssemblyLineFunctionComponent] [Read_CSV_File] CTGDJW003I Parser will use first input
- [AssemblyLineFunctionComponent] CTGDIS087I Iterating.
- [AssemblyLineFunctionComponent] *** Skipping incomplete entry
- [AssemblyLineFunctionComponent] CTGDIS003I *** Start dumping Entry
- [AssemblyLineFunctionComponent] Operation: generic
- [AssemblyLineFunctionComponent] Entry attributes:
- [AssemblyLineFunctionComponent] Last (replace):
- [AssemblyLineFunctionComponent] Title (replace):
- [AssemblyLineFunctionComponent] First (replace): 'Roger'
- [AssemblyLineFunctionComponent] FullName (replace): 'Roger '
- [AssemblyLineFunctionComponent] CTGDIS004I *** Finished dumping Entry
- [AssemblyLineFunctionComponent] CTGDIS088I Finished iterating.
- [AssemblyLineFunctionComponent] CTGDIS100I Printing the Connector statistics.
- [AssemblyLineFunctionComponent] [Read_CSV_File] Get:7
- [AssemblyLineFunctionComponent] [Incomplete data] Branch True:1, Branch False:6
- [AssemblyLineFunctionComponent] [Write to log] (No statistics for script component.)
- [AssemblyLineFunctionComponent] [DumpWorkEntry] (No statistics for script component.)
- [AssemblyLineFunctionComponent] [Exit Flow] (No statistics for script component.)

```

Abbildung 84. Speicherauszug für Work-Entry, gefolgt von statistischen Daten zur Fertigungslinie (dieser Screenshot ist in Englisch)

Ihr Service ist betriebsfähig, aber die Übung ist hier noch nicht zu Ende. Zuerst verbessern und vereinfachen Sie sie, indem Sie Ihre Web-Server-Fertigungslinie anweisen, einige HTML-Seiten zurückzugeben. Dazu müssen in der Regel eine Reihe von Scripts erstellt werden. Da Ihnen jedoch weitere Lernprogrammdateien mit entsprechendem Inhalt zur Verfügung stehen, können Sie Scripts einfach ziehen und übergeben.

Inaktivieren Sie zunächst das Script für 'SpeicherauszugfürWork_Eintrag', um die Protokollausgabe zu minimieren. Fügen Sie direkt nach der IF-Verzweigung 'RunAL detected' eine ELSE-Verzweigung ein. Nennen Sie sie 'Return web page'. Suchen Sie nun mit einem Dateibrowser im Lernprogrammverzeichnis nach der Scriptdatei mit dem Namen 'Return web page.script' und ziehen Sie sie in **Ressourcen > Scripts**. Von dort aus können Sie sie in Ihre Fertigungslinie ziehen und Sie in der ELSE-Verzweigung übergeben. Ihre Fertigungslinie sollte nun wie folgt aussehen:

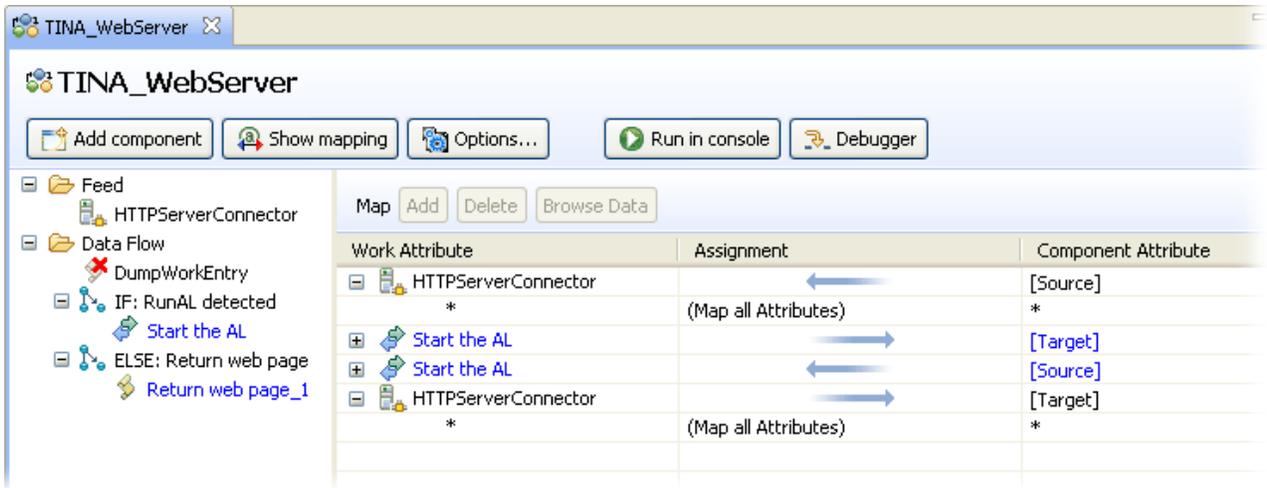


Abbildung 85. Abgeschlossene Fertigungslinie 'TINA_WebServer'

Führen Sie die Fertigungslinie erneut aus. Wenn Sie nun `http://localhost` aufrufen, sollte folgende Webseite angezeigt werden:

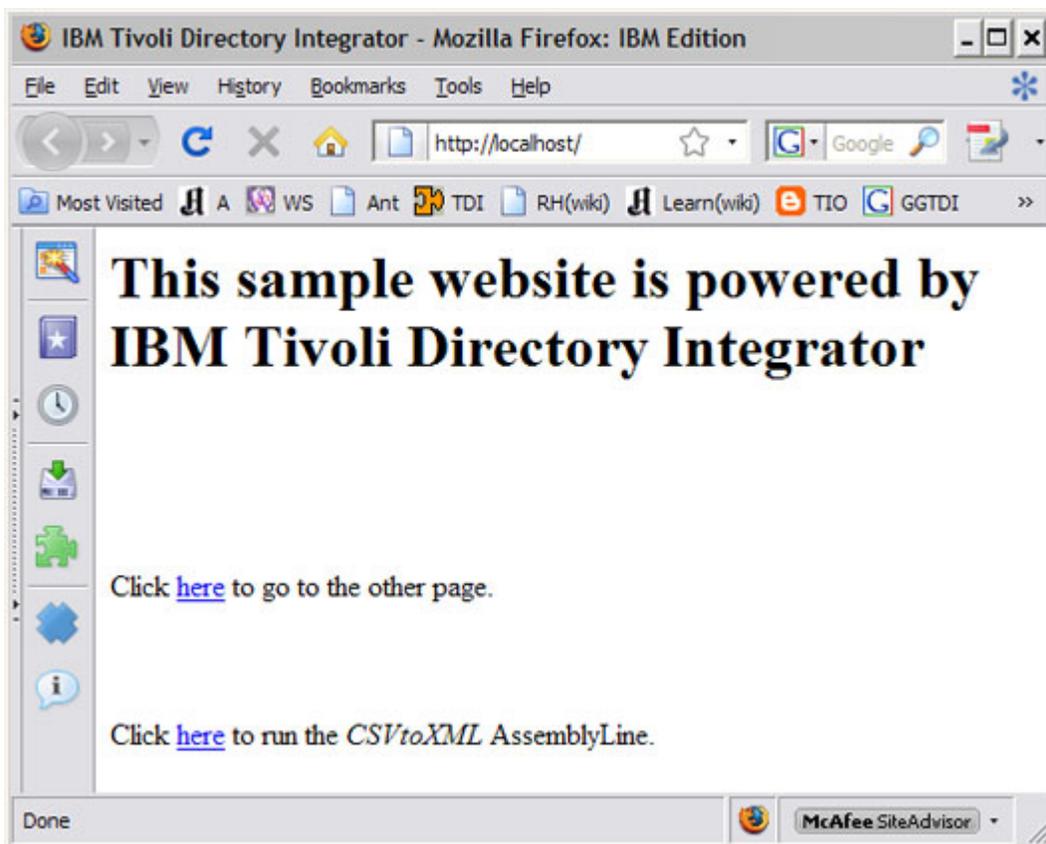


Abbildung 86. Einfache Web Interface für Ihre Lösung

Mit dem Link oben können Sie zu `OtherPage.html` navigieren, während mit dem Link unten der erforderliche Text 'RunAL' in der Datei 'http.base' gesendet wird, um Ihre Fertigungslinie zu starten.

Damit ist der Teil mit den praktischen Übungen in diesem Handbuch abgeschlossen.

Kapitel 4. Abschottung Ihrer Integrationslösungen

Wie Sie gesehen haben, ist die Erstellung von Fertigungslinien eine schnell erledigte Aufgabe. Eine abgeschlossene Fertigungslinienausführung bedeutet jedoch nicht, dass Ihre Lösung bereit ist für Spitzenzeiten. Sogar einfache Integrationstasks erfordern ein Minimum an Vorausdenken und Ergebnisanalyse.

Über folgende relevante Fragen sollten Sie sich Gedanken machen:

- Werden alle Datenquellen wie erwartet verarbeitet? Wie können Sie dies bestätigen?
- Sind Unregelmäßigkeiten im Dateninhalt und bei der Datenqualität festgestellt worden? Werden Sie bearbeitet?
- Hat die Verarbeitung Auswirkung auf andere Datensätze, Systeme oder Fertigungslinien? Inwiefern?
- Muss die Integration protokolliert werden?
- Wer wird Ihre Lösung implementieren? Wer wird Sie verwenden und von wem und wie wird sie verwaltet?

Bei Fertigungslinien mit langer Laufzeit, wie solchen, die für Synchronisationen und zum Versorgen von Services eingesetzt werden, müssen Sie zusätzliche Überlegungen wie Verfügbarkeit und Fehlertoleranz, Leistung, Skalierbarkeit und Sicherheit in Betracht ziehen.

Das Ziel dieses letzten Abschnitts ist es, Ihnen diese Fragen vor Augen zu führen sowie eine Reihe von IBM Security Directory Integrator-Funktionen und Verfahren aufzuzeigen, die Sie zur Bearbeitung dieser Fragen verwenden können.

Anmerkung: Falls Sie zum ersten Mal mit IBM Security Directory Integrator arbeiten, sollten Sie sich keine Gedanken machen, wenn Ihnen dieser Abschnitt kompliziert und schwer nachvollziehbar erscheint. Lesen Sie diese Abschnitte einfach erneut, wenn Sie etwas mehr Erfahrung haben und mit dem System vertrauter geworden sind.

Lesbarkeit, Wiederverwendung und Konfigurierbarkeit

Jegliche Entwicklungsarbeit erfordert Fehlerbehebung, Pflege und Erweiterung. IBM Security Directory Integrator-Lösungen bilden keine Ausnahme.

Sie können dies vereinfachen, indem Sie einige grundlegende Richtlinien befolgen.

1. Vergessen Sie während des Schreibens Ihrer Fertigungslinie nicht, dass Ihre Fertigungslinie von anderen verstanden, verwendet und verwaltet werden muss. Dies bedeutet, dass Fertigungslinien so kurz wie möglich sein und Komponenten klare und beschreibende Namen erhalten sollten. Nichtprogrammierer können Logik, die mittels Verzweigungen und Schleifen in die Fertigungslinie implementiert ist, einfacher lesen und für sie ein Debug ausführen, als Scripts, die in Hooks 'verborgen' sind oder in Scriptkomponenten gepackt wurden.
2. Als logische Folge einer *kurzen Fertigungslinie* sollten Scriptfragmente ebenfalls kurz gehalten werden. Schreiben Sie keine kompakten Codeblöcke, sondern teilen Sie den Code in kleinere Einheiten auf und platzieren Sie ihn sogar in se-

parate Scriptkomponenten, um die Lesbarkeit und die Debugfähigkeit zu erhöhen. Es ist dann möglich, eine Scriptkomponente zu inaktivieren, um Code zu überspringen.

Eine andere Möglichkeit, die Lesbarkeit zu erhöhen und die Codeduplizierung zu verhindern, bietet die Verwendung der Vererbung von Scriptkomponenten (im Ordner 'Scripts' in der Baumstrukturansicht 'Navigator') und die Definition von Funktionen für allgemeine Tasks. Eine Fertigungslinie wird im Kontext ihrer zugehörigen Scriptsteuerkomponente ausgeführt, damit alle an einer Stelle deklarierten Variablen und Funktionen in der gesamten Fertigungslinie zur Verfügung stehen. Die übliche Position, um Variablen und Funktionen zu definieren, ist in den Prologhooks der Fertigungslinie oder in Scripts, die als "Zusätzliche Prologe" definiert wurden.³⁵

3. Ziehen Sie die Lesbarkeit der Eleganz vor, wenn es um Ihre Algorithmen geht. Vergessen Sie auch hier nicht, dass Ihnen Ihre eigene Arbeit sechs Monate später möglicherweise so fremd wie die Arbeit eines anderen ist. Durch Rücksichtnahme auf Ihre Kollegen erweisen Sie sich selbst einen Dienst.
4. Beachten Sie, dass möglicherweise Mitarbeiter ohne Konfigurationseditor-Know-how Einstellungen modifizieren und Ihre Fertigungslinien ausführen müssen. Fertigungslinien können ohne großen Aufwand von der Befehlszeile aus gestartet werden:

```
ibmdisrv -c myConfig.xml -r myAssemblyLine
```

Dies bedeutet, dass Sie Scripts oder Stapeldateien vorbereiten können, um dies zu vereinfachen.

5. Vereinfachen Sie das Rekonfigurieren Ihrer Fertigungslinien, indem Sie Parametereinstellungen durch die Verwendung von Eigenschaften auslagern. Eigenschaften sind Schlüssel-Wert-Paare, die in Dateien oder Datenbanken gespeichert werden können und die es ermöglichen, dass Ihre Lösung von außerhalb des Konfigurationseditors rekonfiguriert werden kann. Sie binden Eigenschaften an Komponentenparameter, indem Sie auf eine Parameterbezeichnung klicken und auf die Option **Eigenschaft hinzufügen** klicken.

Eigenschaften können über Ihre Scripts mithilfe der Aufrufe `system.getTDIProperty()` und `system.setTDIProperty()` auch abgefragt und modifiziert werden. Dies ermöglicht es Ihnen, auch angepasste Logik über externe Eigenschafteneinstellungen ohne großen Aufwand umschaltbar zu machen.

Eigenschaften können darüber hinaus in einem aktiven Server geändert werden, indem das Befehlszeilendienstprogramm `bin/tdisrvctl` verwendet wird, über das Sie Fertigungslinien auch starten und stoppen, Status abfragen und Konfigurationen (erneut) laden können - ohne den Server zu stoppen.

6. Wie bereits oben erwähnt, vereinfacht die Verwendung von relativen Pfaden für Dateien das Versetzen Ihrer Lösung in eine neue Installation. Es wird empfohlen, dass Sie ihre Pfade relativ zu dem Verzeichnis angeben, aus dem die XML-Konfigurationsdatei geladen wird. Auf dies kann über die Eigenschaft `{config.$directory}` zugegriffen werden, die anschließend verwendet werden kann, um bestimmte Pfadparameter mithilfe der Option **Text mit Substitution** anzugeben. Beispiel:

```
{config.$directory}/html
```

35. Zusätzliche Prologe werden ausgeführt, bevor ein eigener Prologhook der Fertigungslinie aufgerufen wird. Diese werden in der Anzeige 'Einstellungen' der Fertigungslinie ausgewählt. Sie rufen diese Anzeige auf, indem Sie mit der rechten Maustaste auf eine Fertigungslinie klicken und die Option **Einstellungen für Fertigungslinie** auswählen.

7. Gehen Sie wie bereits zuvor erwähnt insbesondere beim Erstellen von Lösungen, die von anderen Benutzern implementiert und ausgeführt werden, nicht davon aus, dass diese Benutzer über IBM Security Directory Integrator-Know-how verfügen. Stellen Sie für den Start Ihrer Fertigungslinie Stapeldateien/Scripts bereit, einschließlich von Fertigungslinien zum Testen und Überprüfen. Diese könnten z. B. einfach eine Verbindung zu Datenquellen herstellen und zurückberichten, ob Erfolg oder ein Fehler eingetreten ist. Beachten Sie, dass Sie die Servermethode (`main.logmsg()`) verwenden müssen, um Nachrichten in die Konsolbefehlszeile zu schreiben, damit Ihre Stapeldateien/Scripts Statusinformationen zurückgeben. Verwenden Sie hierzu nicht die Fertigungslinienversion `task.logmsg()`, die Sie in den Lernprogrammübungen verwendet haben. Diese würde Ihre Nachricht lediglich an das Protokoll senden.

Dies sind einfach einige Hinweise. Weitere Informationen finden Sie in anderen IBM Security Directory Integrator-Dokumentationen und in den Newsgroups.

Protokollierung und Prüfung

IBM Security Directory Integrator verwendet *log4j*, um eine flexible Protokollverwaltung bereitzustellen. Sie können aus einer Anzahl standardmäßiger *Appender* wählen, zu denen 'syslog' für Unix, 'eventlog' für Windows, tägliche Dateien und Umlaufprotokolle gehören. Neue Appender können erstellt oder heruntergeladen und ebenfalls verwendet werden.

Standardmäßig ist nur eine minimale Serverprotokollierung aktiviert. Sie sollten als absolutes Minimum die Protokollierung für Ihre Fertigungslinien mithilfe des Appender 'FileRoller' definieren, der in den Unterordner 'logs' Ihres Lösungsverzeichnisses schreibt, und der Protokolldatei denselben Namen wie die Fertigungslinie geben. Für Ihre Fertigungslinie 'CSV2XML' würden Sie also einen Satz von Umlaufprotokolldateien basierend auf dem folgenden Dateipfad definieren: `logs/CSV2XML.log`.

Beachten Sie, dass Sie mit der Methode `logmsg()` die Protokollebene für Ihre Nachricht optional definieren können, indem Sie eines der folgenden Schlüsselwörter direkt vor Ihrer Protokollnachricht als erstes Argument übergeben: DEBUG, INFO, WARN, ERROR oder FATAL. Protokollebenen sind einschließend, WARN enthält also auch ERROR und FATAL; die Ebene DEBUG bedeutet, dass alle Nachrichten aller Ebenen protokolliert werden. Beispielsweise wird eine Nachricht wie `task.logmsg("DEBUG", "Updated: " + conn);`

nur von Appender ausgegeben, die für die Protokollierung auf der Ebene DEBUG festgelegt sind.

Sie können Ihrer Lösung Prüfnachrichten hinzufügen, die außerhalb eines aktiven Servers aktiviert bzw. inaktiviert werden können, indem Aufrufe an `task.logmsg()` mit einer IF-Anweisung als Präfix versehen werden, die den Wert einer Eigenschaft überprüft. Beispiel: Dieses Scriptfragment kann in einem Hook 'DataFlow - Update Successful' angezeigt werden:

```
if (system.getTDIPProperty("MyProps","audit").equalsIgnoreCase("true"))
    task.logmsg("DEBUG", "Updated the following data: " + conn);
```

Wenn Sie das Befehlszeilendienstprogramm `tdisrvctl` zum Ändern des Werts der Eigenschaft "audit" im Eigenschaftsspeicher "MyProps" verwenden, können Sie diesen Typ von Prüfnachrichten für einen aktiven Server dynamisch ein- bzw. ausschalten.

Im Allgemeinen ist es besser, zu viele als zu wenige Informationen zu protokollieren. Natürlich sollten Sie auch nicht die Protokollausgabe überlaufen lassen. Es kann sich als schwierig erweisen, eine überfüllte Protokollausgabe nach wichtigen Nachrichten zu durchsuchen.

Konnektivitätsprobleme

Konnektivitätsprobleme können im Allgemeinen in zwei Kategorien eingeteilt werden: Initialisierungsfehler und unterbrochene Verbindungen.

Standardmäßig rufen alle Komponenten zu Beginn des Fertigungslinienbetriebs während der Initialisierungsphase Verbindungen auf. Wenn aus einem Grund eine Verbindung zu diesem Zeitpunkt fehlschlägt, wird der Hook 'Prolog – On Connection Error' aufgerufen. Dadurch verfügen Sie über einen Scriptcontainer, von dem aus Sie Alerts senden oder sogar Parametereinstellungen ändern und versuchen können, erneut eine Verbindung herzustellen. Entsprechend verwenden Sie beim Auftreten von Verbindungsfehlern während des Fertigungslinienzyklus den Hook 'Datenfluss – On Connection Lost', um diese Situation zu handhaben.

Zusätzlich zu dieser benutzerdefinierten Handhabung bieten Connector und Funktionskomponenten auch die integrierte Funktion zur Verbindungswiederherstellung (*reconnect*) über die Registerkarte **Verbindungsfehler**. Hier können Sie die Komponente anweisen, eine unterbrochene Verbindung wiederherzustellen oder, im Falle von Initialisierungsproblemen, mit dem Versuch fortzufahren, die Verbindung herzustellen.

Falls es sich um Initialisierungsfehler handelt, ist es nicht das übliche Verfahren, die Verbindungswiederherstellung zu aktivieren, es sei denn, Sie haben wiederholt auftretende Probleme wie sporadisch auftretende Zeitlimitüberschreitungen während der SSL-Vereinbarung oder ähnliche sporadisch auftretende Verbindungsprobleme.

Es empfiehlt sich jedoch, die Funktion **Verbindung bei Unterbrechung automatisch wiederherstellen** zu aktivieren. Dadurch kann Ihre Komponente die zugehörige Verbindung wieder herstellen und fortfahren, als sei nichts geschehen. Beachten Sie, dass im Falle von Connector im Modus *Iterator* das Wiederherstellen einer Verbindung auch bedeutet, dass der 'Cursor' für die Iteration möglicherweise zurückgesetzt wird, sodass der Zyklus erneut mit dem ersten Eintrag in der Ergebnismenge beginnt. Dies stellt für zustandsbestimmte Iteratoren wie Änderungserkennungconnector kein Problem dar, da diese Komponenten automatisch Statusinformationen verwenden, um an dem Punkt fortzufahren, wo sie aufgehört haben.

Verfügbarkeit der Fertigungslinie

Eine Verbesserung der Verfügbarkeit der Fertigungslinie umfasst zwei Dinge: 1) Alles zu tun, um zu verhindern, dass Fertigungslinien gestoppt werden und 2), gestoppte Fertigungslinien so schnell wie möglich erneut zu starten. Bei Szenarien wie Migrationen mit langer Laufzeit und Synchronisationen ist es beim Neustart von Fertigungslinien möglicherweise auch erforderlich, dass eine Fortsetzung vom Fehlerpunkt aus erfolgt.

Eine Fertigungslinie stoppt, wenn eine nicht behandelte Ausnahmebedingung auftritt. Sie können sich gegen diese Möglichkeit schützen, indem Sie alle Fehler behandeln. Dies bedeutet in IBM Security Directory Integrator, dass mindestens der

Hook 'Default On Error' aller Connector und Funktionskomponenten aktiviert wird. Indem Sie Fehlerhooks aktivieren, weisen Sie den Server an, trotz einer Ausnahmebedingung fortzufahren.

Wie bei den Lernprogrammübungen beschrieben, wird Ihnen bei Stoppen der Fertigungslinie aufgrund eines Fehlers ein Stack-Trace angezeigt. Dem Stack-Trace gehen Informationen zur Position und zur Ursache des Fehlers voraus. Wenn Sie durch Aktivieren von Fehlerhooks verhindern, dass die Fertigungslinie stoppt, müssen Sie den Fehlerstatus berichten, indem Sie spezielle Objekte verwenden, die für Ihre Scripts zur Verfügung stehen.

```
task.logmsg("ERROR", "[" + thisConnector.getName() + "] - " +
            error);
```

Das oben stehende Beispielscript verwendet zwei solcher Objekte: `thisConnector`, das stets auf die Komponente verweist, mit der das zurzeit ausgeführte Script verknüpft ist, und die vordefinierte Variable mit dem Namen `error`. Das Objekt `error` ist ein Eintrag wie `work` und `conn` und es enthält Attribute wie `'status'`, `'connectorname'`³⁶ und `'message'`, plus weitere relevante Details zu kürzlich aufgetretenen Fehlersituationen. Da es sich um ein Eintragsobjekt handelt, können Sie zum Anzeigen des zugehörigen Inhalts `task.dumpEntry(error)` oder auch direkte Verweise auf Attributnamen wie zum Beispiel `work.message` verwenden oder einfach `error` an eine Zeichenfolgenachricht wie im obenstehenden Beispiel anhängen, da sich alle Eintragsobjekte bei Bedarf selbst in Zeichenfolgedarstellungen konvertieren können.

Zum Handhaben von Ausnahmebedingungen, die in Scripts wie in der Attributzuordnung sowie in Hooks und Scriptkomponenten auftreten, schließen Sie Ihren Code in 'try-catch'-Blöcke ein. Dadurch können Sie Ausnahmebedingungen abfangen und sie selbst bearbeiten:

```
try {
    res = myLib.callToSomeFunctionThatMightFail();
} catch (excptn) {
    task.logmsg("Call failed with error: " + excptn);
}
```

Zusätzlich zur Behandlung der Fehler sollten Sie die Funktion zum automatischen Wiederherstellen einer Verbindung bei Unterbrechung verwenden, die im vorhergehenden Abschnitt beschrieben wurde. Diese verhindert, dass Ihre Fertigungslinie aufgrund von vorübergehenden Konnektivitätsproblemen fehlschlägt, wie z. B. dem Überschreiten des zulässigen Zeitlimits durch eine Datenquelle oder die Firewall.

Wenn Ihre Fertigungslinie aufgrund einer Ursache vorzeitig stoppt, muss der nächste Schritt sein, sie erneut zu starten. Eine Methode wäre, das Webverwaltungstool zu verwenden, um das Fehler- und das Antwortverhalten zu definieren.

Eine weitere, sogar ergänzende Methode besteht darin, eine 'Starter'-Fertigungslinie zu erstellen und die Funktionskomponente der Fertigungslinie zu nutzen, die Sie in den Lernprogrammübungen des letzten Abschnitts verwendet haben. Indem Sie diese Funktionskomponente der Fertigungslinie in eine endlose bedingte Schleife platzieren - mit anderen Worten, mit einer Bedingung ausstatten, die stets `true`

36. Wie Sie möglicherweise festgestellt haben, ist das Wort 'Connector' manchmal synonym zu 'Komponente', genauso wie Variablen wie `thisConnector` sich auch auf Funktionskomponenten, Scriptkomponenten und Attributzuordnungskomponenten beziehen können. Dasselbe trifft auf das Attribut `'connectorname'` des Objekts `error` zu, das auch den Namen einer Komponente beliebigen Typs aufweisen kann.

(wahr) ist – und anschließend die Funktionskomponente der Fertigungslinie so konfigurieren, dass sie die gewünschte Servicefertigungslinie aufruft und wartet, bis diese abschließt, stellen Sie sicher, dass die endlose Schleife Ihren Datenfluss einfach erneut startet, wenn die Steuerung an die 'LauncherAL' ('Starter'-Fertigungslinie) zurückgegeben wird.

Wenn Sie das oben beschriebene Verfahren mit der Schleife zum erneuten Starten auf eine Synchronisationsfertigungslinie anwenden, die auf einem der Änderungs-erkennungconnector (oder der Delta Engine, beide werden an anderer Stelle beschrieben) basiert, fährt die Fertigungslinie automatisch von dem Punkt aus fort, an dem sie fehlgeschlagen ist. Falls nicht, dann lastet die Statushandhabung auf Ihnen. Es ist ein übliches Verfahren, den Systemspeicher zu verwenden, um Statusinformationen wie Zeitmarken oder andere Schlüsselwerte für sortierte Ergebnismengen auf Platte zu speichern. Bei jedem Initialisieren der Fertigungslinie werden diese Statusinformationen dann auf den Connector im Iteratormodus angewendet, um die Verarbeitung nach dem zuvor behandelten Eintrag unverzüglich wiederaufzunehmen.

Wenn die Iterationsdatenquelle für die Fertigungslinie keine sortierten Rückgaben unterstützt, muss die Iteration möglicherweise von Beginn an gestartet werden. Beachten Sie in diesem Fall, dass der Aktualisierungsmodus des Connectors die Funktion **Änderungen berechnen** bietet, die von der Ausgabe zugeordnete Attribute (Output Mapped Attributes) mit den zurzeit im Zielsystem gefundenen vergleicht und die Modify-Operation überspringt, wenn keine Unterschiede festgestellt werden.

Sie können einen Single Point of Failure vermeiden, indem Sie für mehrere IBM Security Directory Integrator-Server ein sicheres Transportprotokoll wie z. B. IBM MQ einführen. Auf diese Weise kann eine beliebige Anzahl von Servern (und Fertigungslinien) verwendet werden, um die Verarbeitung durch Platzieren von Daten und sogar Verarbeitungsanweisungen in die Warteschlange zu starten. Auf der Empfangsseite nehmen mehrere Server/Fertigungslinien diese in der Reihenfolge des Eingangs auf und führen die angeforderten Arbeiten aus. Dies führt nicht nur zu einer zuverlässigeren Lösung, sondern es ermöglicht auch ein einfacheres Skalieren durch Hinzufügen von Sender- und Empfängerfertigungslinien.

Hier wurde ein umfangreiches Thema in Kürze behandelt. Das Ziel war allerdings auch eher die Inspiration anzuregen, als einen bestimmten Ansatz vorzuschreiben. Es empfiehlt sich, nach detaillierteren Empfehlungen und Beispielen auf Community-Websites und in Diskussionsgruppen zu suchen.

Skalieren und Leistung

Dies ist wieder ein Thema, das eine längere Beschreibung als die hier vorliegende verdienen würde. Einige Punkte sollen allerdings erwähnt werden, wenn auch eher allgemein.

Der ausschlaggebende Faktor für die Verarbeitungsgeschwindigkeit Ihrer Datenflüsse ist die Ein-/Ausgabe. Die Zeit, die zum Abrufen von Daten aus Quellen oder zum Ausgeben von Änderungen an Ziele verwendet wird, beeinflusst stark die Verarbeitungszeit innerhalb Ihrer Fertigungslinie. Darüber hinaus kann das Aushandeln von Verbindungen sehr kostenintensiv sein, insbesondere wenn für die Sicherheit das Handshakeverfahren einbezogen wird. Wenn Sie diese Punkte beim Entwurf und der Erstellung Ihrer Fertigungslinien im Auge behalten, wird dies direkten Einfluss auf deren Leistung haben.

Stellen Sie sich beispielsweise eine servermodusbasierte Fertigungslinie vor, die eingehende Protokollanforderungen von Clients empfängt, wie z. B. von der HTTP-Server-Lösung, die Sie in der letzten Übung erstellt haben. Jede empfangene Anfrage bewirkt, dass eine Servicefertigungslinie gestartet wird, um die tatsächliche Arbeit auszuführen. Wenn diese aufgerufene Fertigungslinie Komponenten initialisieren muss, so wird die Umschaltung für jede Anforderung mindestens soviel Zeit in Anspruch nehmen, wie die Gesamtzeit zum Herstellen aller Verbindungen.

Es gibt folgende drei Möglichkeiten, um diese Situation zu entschärfen:

- Lassen Sie die Hauptfertigungslinie im Servermodus die tatsächliche Verarbeitung ausführen, anstatt diese Aufgabe zu vergeben. Dadurch ist es nicht mehr erforderlich, dass Verbindungen für jede Anforderung hergestellt und wieder unterbrochen werden.
- Entwerfen Sie die Servicefertigungslinie(n) so, dass Sie im Modus *Manuell (Umlaufmodus)* aufgerufen werden kann/können. 'Manuell (Umlaufmodus)' bewirkt, dass die Servicefertigungslinie initialisiert wird, wenn die Funktionskomponente der Fertigungslinie initialisiert wird. Darüber hinaus bewirkt die Funktionskomponente der Fertigungslinie, dass die Servicefertigungslinie pro Aufruf nur einen Zyklus ausführt. Auf diese Weise funktioniert die Servicefertigungslinie wie eine Komponente der aufrufenden Fertigungslinie und muss so aufgebaut werden, dass dieses Verhalten berücksichtigt wird.
- Verwenden Sie globales Connector-Pooling. Diese im Handbuch *Referenzinformationen* beschriebene Funktion ermöglicht es Ihnen, einen Pool mit Connector zu definieren, die beim Starten des Servers initialisiert werden und von mehreren Fertigungslinien wie benötigt gemeinsam genutzt werden können.

Eine weitere Möglichkeit zum Verbessern der Leistung ist das Verteilen von verarbeitungsintensiven Tasks auf mehrere simultan laufende Fertigungslinien. Nehmen Sie als Beispiel eine Migrationstask, bei der nicht eine einzige Fertigungslinie mit dem gesamten Quellendatensatz arbeitet, sondern Sie stattdessen mehrere Fertigungslinien starten, die jeweils eine Untergruppe handhaben. Da Sie Initialisierungsparameter beim Aufrufen an eine Fertigungslinie übergeben können, kann eine einzige Fertigungslinie entwickelt werden, die mehrere Male mit einem Filterparameter zum Steuern des Datenbereichs gestartet wird, den die Instanz verarbeiten soll.

Als weitere Möglichkeit können Sie auch einen Nachrichtenbus in Ihre Lösung integrieren, wie im vorherigen Abschnitt beschrieben. Dieser Ansatz wurde mit großem Erfolg bei einigen der größten Kunden von IBM eingesetzt.

Falls die Verarbeitungsgeschwindigkeit durch instabile Netzverbindungen oder Systeme mit niedriger Verfügbarkeit behindert wird, können Sie zusätzliche Fertigungslinien als Hintergrundtasks implementieren, die schlecht zu erreichende Daten mit lokalen Hochleistungsspeichern synchronisieren. Insbesondere beim Implementieren von Echtzeitservices kann dieses Verfahren sicherstellen, dass für Clientanforderungen eine zufriedenstellende Antwortzeit erreicht wird.

Überwachung

Ohne weitere Modifizierung werden Ihre IBM Security Directory Integrator-Lösungen mit dem Webverwaltungstool schnell und einfach verwaltet. Dieses Integrated Solutions Console-Plug-in (ISC) ist eine Anwendung des Anwendungsservers, die eine beliebige Anzahl von Lösungen überwachen kann, die auf einer beliebigen Anzahl von Servern in Ihrer Infrastruktur ausgeführt werden. Zusätzlich zu sofort einsatzfähigen Funktionen zum Anzeigen von statistischen Daten zu Fertigungslinien und zum Anzeigen von Protokollen und Start-/Stopzeiten ermöglicht das Webverwaltungstool, die Konsole für Status und Störungen anzupassen sowie Zeitpläne und Fehler-/Antwortverhalten zu definieren, um Ihre Fertigungslinien in aktivem Zustand zu halten.

Sie können Ihre IBM Security Directory Integrator-Lösungen auch so konfigurieren, dass sie Statusereignisse senden, zum Beispiel als SNMP-Traps oder mithilfe des angepassten Ereignisformats des Systems. Das System ist für die JMX-Verwaltung und -überwachung, z. B. mithilfe von ITM, fertig konfiguriert.

Der Fertigungsliniendebugger

Das Folgende wurde zwar bereits erwähnt, aber es lohnt sich, dies zu wiederholen: Die Zeit, die Sie einsetzen, um mit dem Fertigungsliniendebugger vertraut zu werden, zahlt sich durch eine beschleunigte Lösungsentwicklung, Fehlerbehebung und Implementierung zehnfach aus.

Anhang. EasyETL-Leitfaden

'ETL' steht für 'Extract, Transform and Load' (Extrahieren, Konvertieren und Laden) und dient dazu, Daten von einer Quelle abzurufen, diese wie erforderlich zu ändern und dann an einer anderen Stelle abzulegen. EasyETL ist eine Funktion von IBM Security Directory Integrator, mit der Sie dies schnell und interaktiv mit nur wenigen Tastatureingaben erledigen können.

Allgemeine Beispiele für ETL sind:

- Exportieren von Datenbanksätzen, Notes-Dokumenten, Verzeichniseinträgen oder sogar von eingehenden Mails oder MQ-Nachrichten in eine Datei.
- Laden von Daten von einer Datei in ein System oder einen Datenspeicher.
- Migrieren von Daten direkt von einem System in ein anderes oder bei Software- oder Schemaaktualisierungen zwischen verschiedenen Versionen desselben Systems.

Mit IBM Security Directory Integrator EasyETL können Sie diese und andere Aufgaben in wenigen intuitiv zu erfassenden Schritten erledigen, sodass Sie Lösungen erhalten, die sowohl für einen einmaligen Datenfluss als auch für wichtige und kritische Datenflüsse in Ihrer Infrastruktur geeignet sind.

Der erste Schritt bei der Erstellung einer neuen EasyETL-Task besteht darin, Ihre Eingabequelle und die Attribute auszuwählen, die Sie übertragen möchten. Bereits zu diesem Zeitpunkt ermöglicht Ihnen IBM Security Directory Integrator die Ausführung Ihres EasyETL-Jobs, um die gelesenen Daten in Ihrem Kopierpuffer zu erfassen und später einzufügen. Wenn die Daten konvertiert oder berechnet werden müssen, können Sie mit EasyETL eine Konvertierung hinzufügen, den ETL-Job erneut ausführen und die konvertierten Daten kopieren bzw. einfügen. Sie können auch ein Ausgabeziel auswählen, und Ihren EasyETL-Job die Daten direkt dorthin schreiben lassen.

Sobald Ihre EasyETL-Lösung wie gewünscht funktioniert, kann IBM Security Directory Integrator die Befehlszeilenressourcen (Stapeldatei oder Scripts) zum Starten und Planen der Integrationstask generieren. Schließlich nutzt EasyETL die Änderungserkennungsfunktionen von IBM Security Directory Integrator, um Ihren ETL-Job schnell in eine Datensynchronisationstask umzuwandeln.

Anmerkung: Die gute Nachricht für IBM Security Directory Integrator-Benutzer ist, dass jede EasyETL-Lösung ein IBM Security Directory Integrator-Projekt mit einer einzigen Fertigungslinie ist, das in einer voll ausgestatteten Entwicklungsumgebung geöffnet werden kann. Wenn Sie es ändern, ist es jedoch nicht mehr als EasyETL verfügbar.

EasyETL verwenden

Starten Sie den IBM Security Directory Integrator-Konfigurationseditor und wählen Sie den zu verwendenden Arbeitsbereich aus. Wenn IBM Security Directory Integrator das erste Mal gestartet wird, wird die Begrüßungsseite angezeigt³⁷.

37. Sie können jederzeit zur Begrüßungsseite zurückkehren, indem Sie **Hilfe > Willkommen!** im Hauptmenü auswählen.

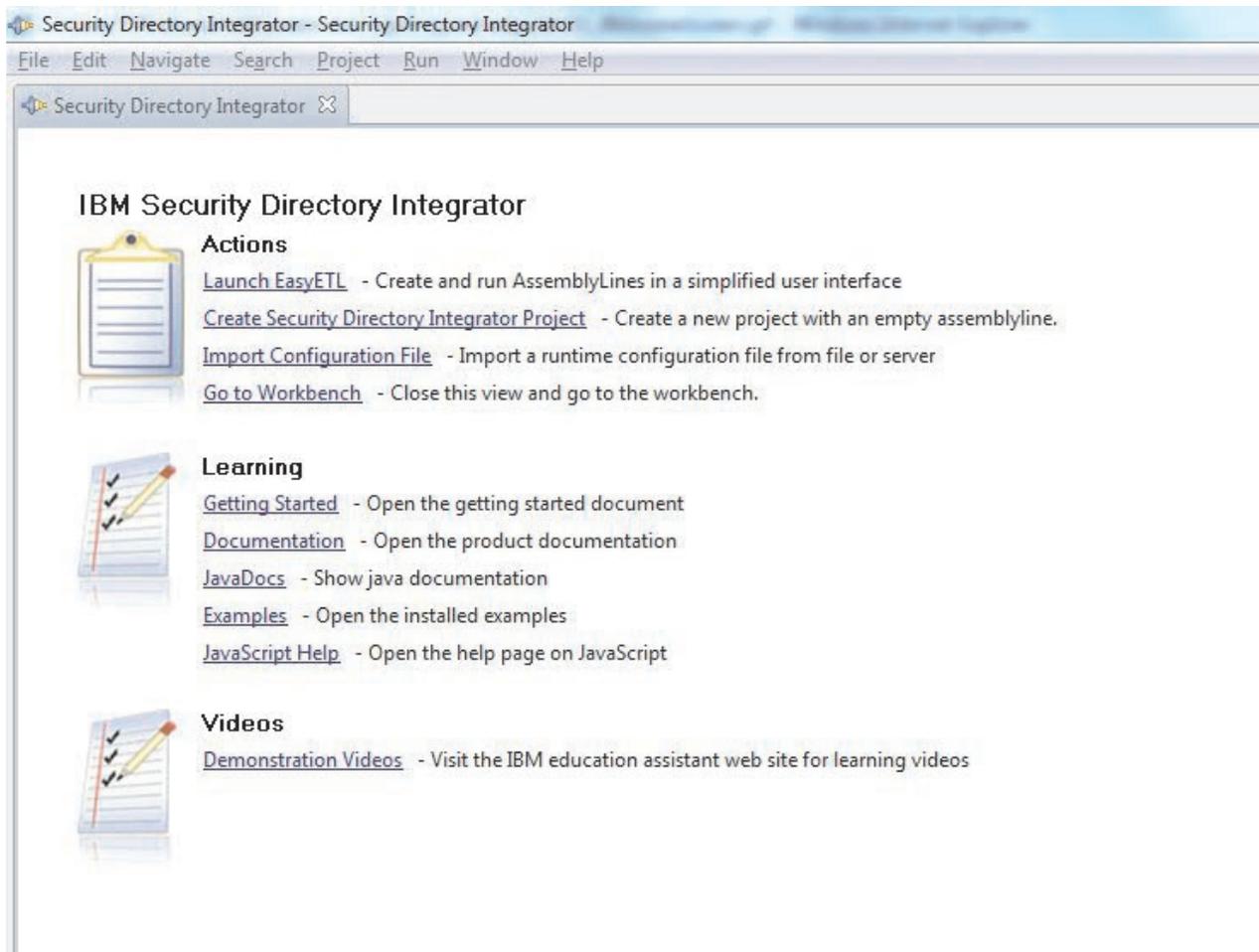


Abbildung 87. Begrüßungsseite

Wie im Screenshot oben dargestellt, öffnet der oberste Link hier die EasyETL-Workbench³⁸. Klicken Sie jetzt auf diesen Link, um die Security Directory Integrator-EasyETL-Workbench zu öffnen.

38. EasyETL ist eine Security Directory Integrator-Perspektive, und Sie können zwischen den Perspektiven mit der folgenden Menüauswahl hin- und herschalten: **Fenster > Perspektive öffnen > Andere....** Falls Sie Änderungen an einer Perspektive vorgenommen haben und diese auf die Standardeinstellungen zurücksetzen möchten, wählen Sie **Fenster > Perspektive zurücksetzen** aus.

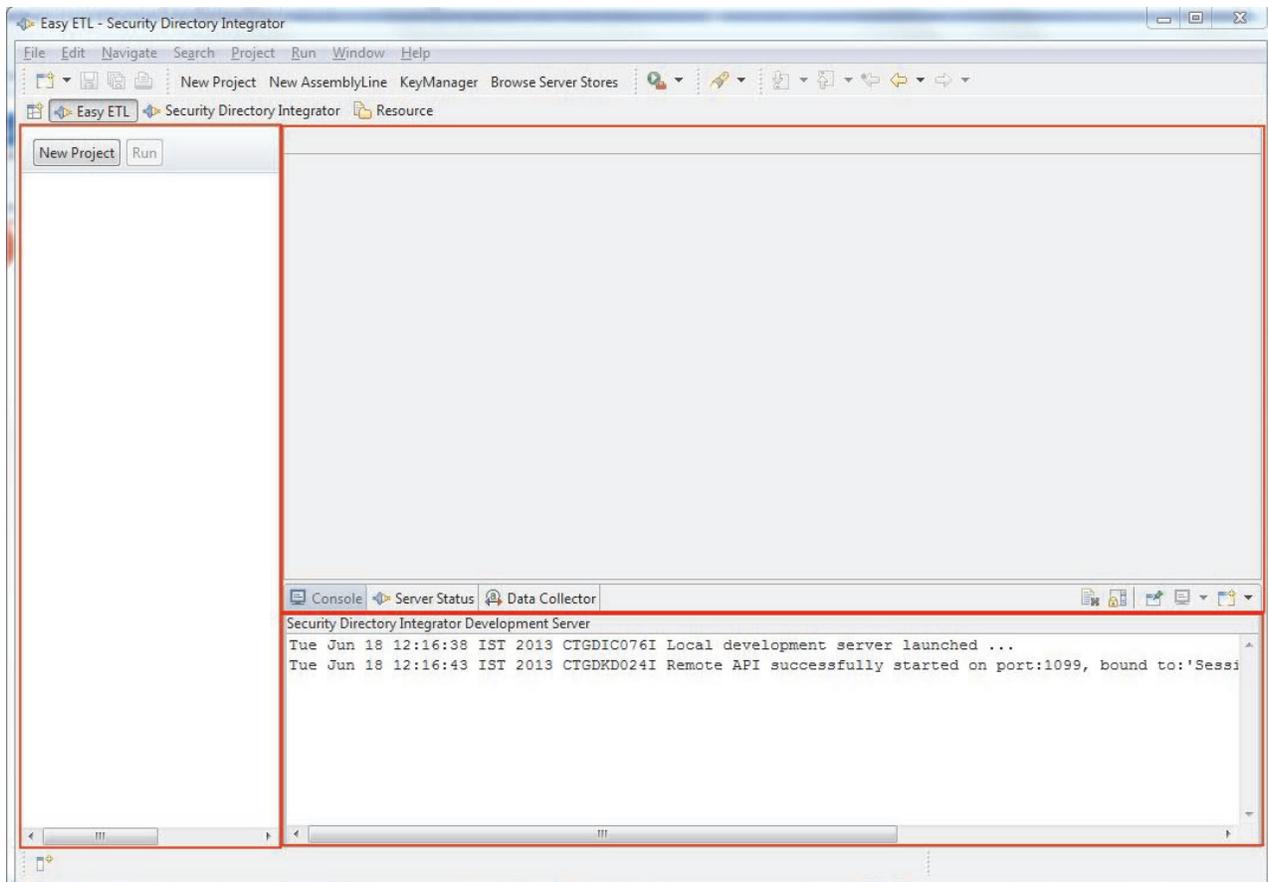


Abbildung 88. Abbildung 2. EasyETL-Workbench

Die EasyETL-Workbench zeigt drei Dinge an:

- Den Projektnavigator, der Ihre ETL-Jobs auflistet. Sie können mit der rechten Maustaste auf ein beliebiges Projekt klicken, um die Jobs auszuführen oder um Befehlszeilenressourcen zum Starten zu erstellen.
- Einen einfachen Fertigungslinieneditor³⁹ für jedes offene Projekt.
- Verschiedene Ansichten wie Registerkarten im unteren Teil der Anzeige. Standardmäßig erhalten Sie drei Ansichten:
 - Konsolenausgabe des IBM Security Directory Integrator-Servers.
 - Serverstatusansicht, in der Sie sowohl den Server als auch aktive EasyETL-Projekte überwachen können.
 - Datenkollektor, in dem die Ergebnisdaten für jeden Zyklus in einer tabellarischen Liste angezeigt werden.

Sie finden beim Lesen des Dokuments weitere Informationen zu jedem Abschnitt.

39. Eine Fertigungslinie ist die Implementierung eines Datenflusses in IBM Security Directory Integrator. Wenn Sie also ein EasyETL-Projekt erstellen oder öffnen, dann wird die zugrunde liegende Fertigungslinie im Editor angezeigt.

Beachten Sie auch, dass der Begriff 'Fertigungslinie' in dieser oder anderen Veröffentlichungen zu IBM Security Directory Integrator mit AL (AssemblyLine) abgekürzt wird.

Projekt erstellen

Erstellen Sie ein neues EasyETL-Projekt, indem Sie auf die Schaltfläche **Neues Projekt** oben im Projektnavigator klicken.

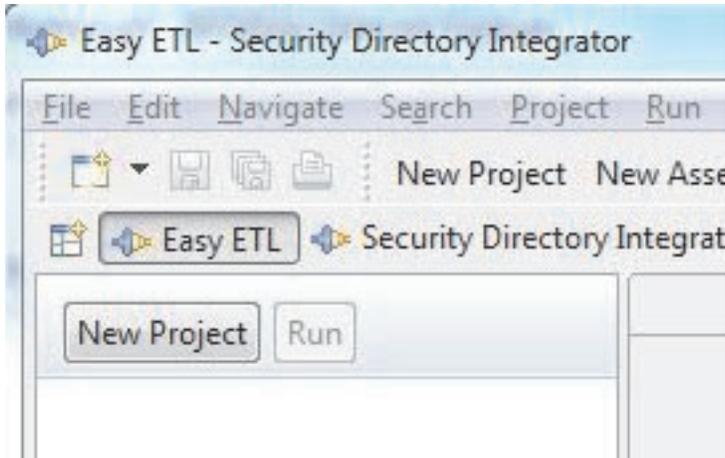


Abbildung 89. Schaltfläche 'Neues Projekt'

Nennen Sie dieses Projekt 'CSVtoXML' und klicken Sie auf **Fertigstellen**. Ihr neues Projekt wird im einfachen Fertigungslinieneditor geöffnet.

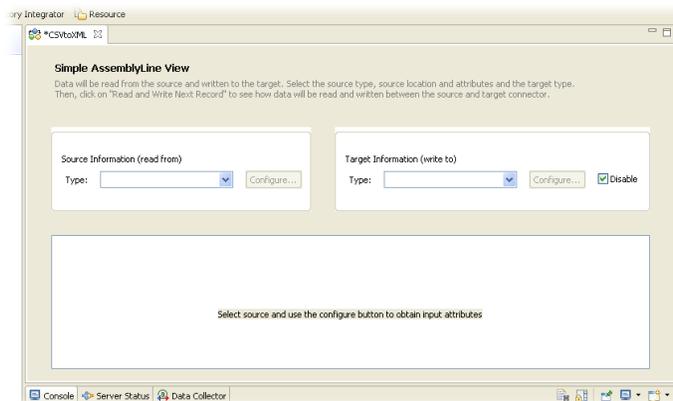


Abbildung 90. Einfacher Fertigungslinieneditor

Der Editor bietet zwei Dropdown-Menüs: eines für die Auswahl der Eingabequelle und eines für die Auswahl des Ziels. Der Bereich unten ist leer (abgesehen vom Hilfetext), bis Sie eine Quelle ausgewählt haben.

Die Eingabe für Ihre ETL-Fertigungslinie konfigurieren

Konfigurieren Sie die Eingabe für Ihre EasyETL-Fertigungslinie, indem Sie auf das Dropdown-Menü für die Quelleninformationen klicken und die Option **Dateiconnector** auswählen.

Simple AssemblyLine View

Data will be read from the source and written to the target. Select the source. Then, click on "Read and Write Next Record" to see how data will be read and

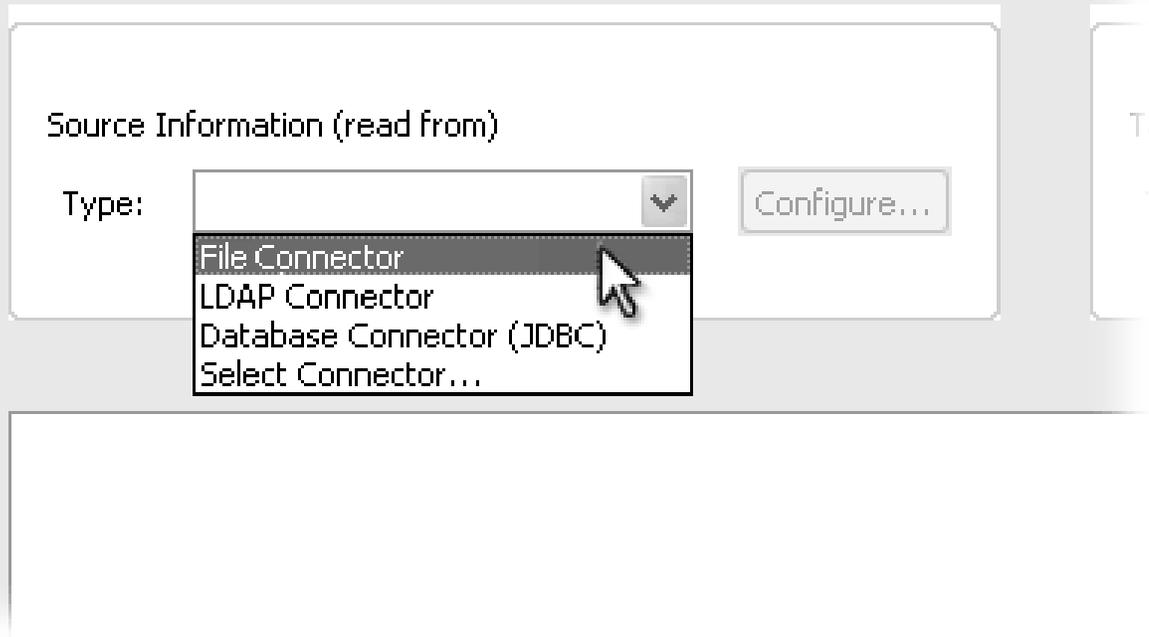


Abbildung 91. Quelleninformationen auswählen

Ihnen wird der Konfigurationsdialog für diesen Connector angezeigt.

Verweisen Sie mit dem Parameter Dateipfad auf die Datei 'People.csv' in folgendem Verzeichnis:

`TDI_HOME/examples/Tutorial`

Dabei wird `TDI_HOME` durch das IBM Security Directory Integrator-Installationsverzeichnis auf der jeweiligen Maschine ersetzt⁴⁰.

40. IBM Security Directory Integrator unterstützt sowohl den normalen als auch den umgekehrten Schrägstrich als Pfadtrennzeichen, wenn die Ausführung unter Windows erfolgt. Ihre IBM Security Directory Integrator-Lösungen weisen eine bessere Portierbarkeit zwischen Windows und allen anderen Plattformen, auf denen IBM Security Directory Integrator ausgeführt wird, auf, wenn Sie den normalen Schrägstrich verwenden.

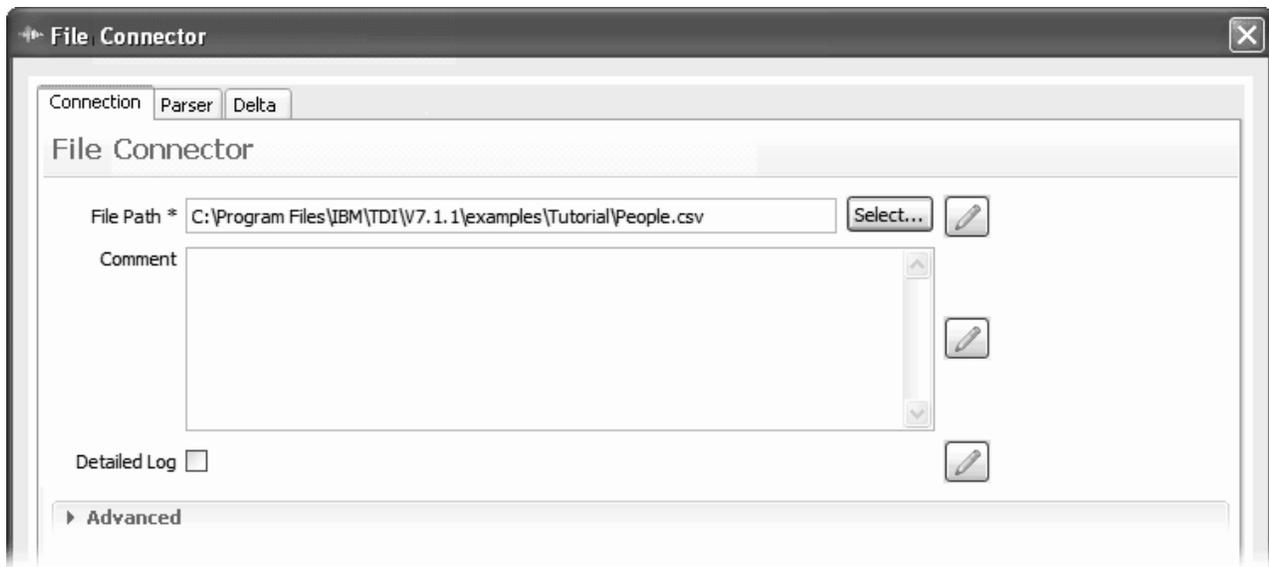


Abbildung 92. Parameter für den Dateipfad konfigurieren

Klicken Sie jetzt auf die Registerkarte mit der Beschriftung **Parser** und wählen Sie **CSV-Parser** aus, wobei Sie die Standardkonfigurationsparameter unverändert beibehalten. Klicken Sie schließlich auf die Schaltfläche **Verbindung herstellen** unten im Dialogfeld, um die Verbindung zu testen und verfügbare Attribute zu erkennen.

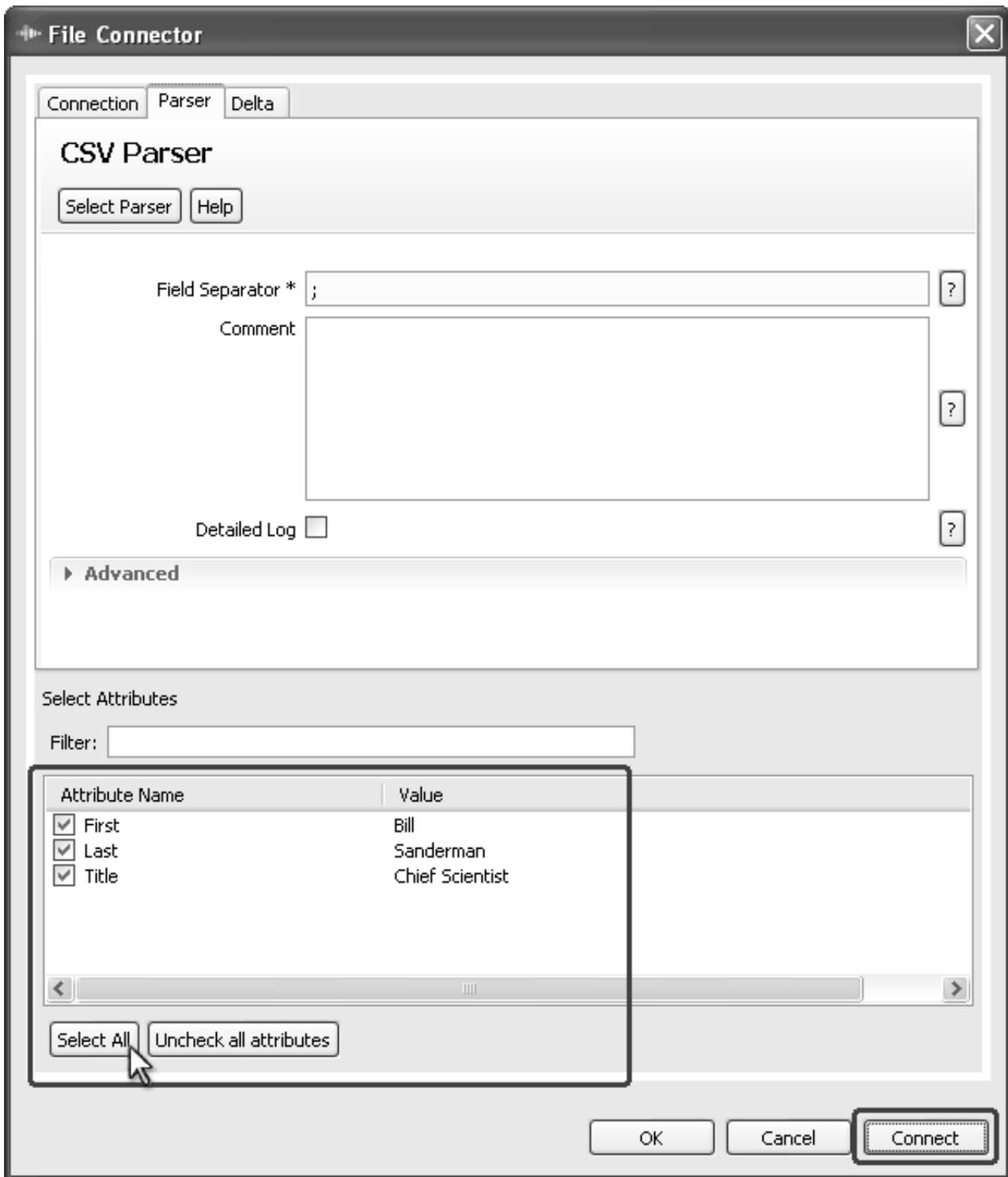


Abbildung 93. Verbindung testen und Schema erkennen

Das Schema für das verbundene System wird angezeigt und Sie können an dieser Stelle auswählen, welches Sie in Ihrem Datenfluss verwenden möchten. Verwenden Sie für dieses Beispiel die Schaltfläche **Alles auswählen** und klicken Sie dann auf **OK**, um den Konfigurationsdialog zu schließen.

Zurück in der EasyETL-Arbeitsumgebung sehen Sie, dass die untere Hälfte des einfachen Fertigungslinieneditors sich verändert hat und nun die konfigurierte Eingabequelle darstellt. IBM Security Directory Integrator bietet jetzt eine Schaltfläche, mit der Sie diese Informationen schrittweise Datensatz für Datensatz durchgehen können, sowie Schaltflächen zur Ausführung der ETL-Task bis zur Fertigstellung und zum Stoppen der Ausführung.

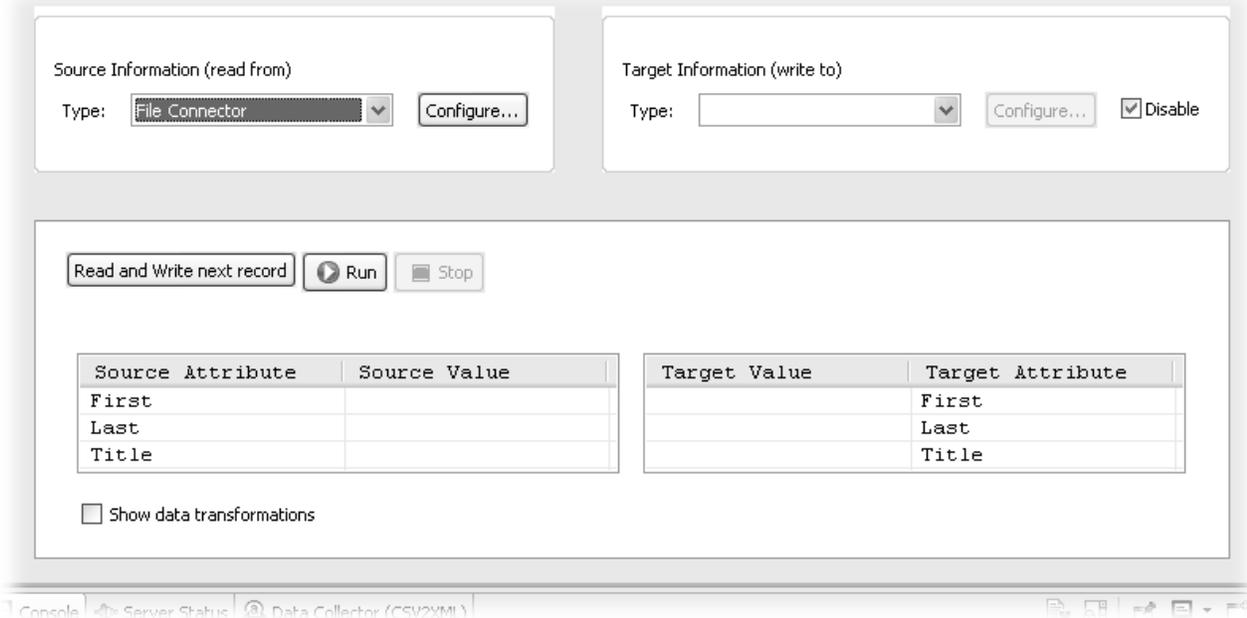


Abbildung 94. Eingabequelle konfiguriert

Unter diesen Schaltflächen befinden sich zwei Gitterfelder, die als *Datenviewer* bezeichnet werden. Die Datenviewer links zeigen Ihre Eingabeattribute an. Die im vorherigen Schritt zum Lesen ausgewählten Eingabeattribute werden in Fettschrift und am Anfang der Aufzählung dargestellt. Diese Attribute werden als *Eingabebezuordnung* bezeichnet. Weiter unten folgen alle nicht ausgewählten Attribute, die grau dargestellt werden. Sie können diese Attribute in die Eingabebezuordnung integrieren, indem Sie doppelt auf sie klicken. Auf gleiche Weise können Sie Attribute aus der Zuordnung entfernen, indem Sie doppelt auf sie klicken oder sie löschen⁴¹.

Das Feld rechts ist der Ausgabeviewer, der die zu schreibende Gruppe der Attribute anzeigt. In IBM Security Directory Integrator wird dies als *Ausgabebezuordnung* bezeichnet und stimmt standardmäßig mit der von Ihnen ausgewählten Liste für die Eingabebezuordnung überein. Beachten Sie, dass Sie den Namen eines Ausgabeattributs ändern können, indem Sie in der rechten Spalte klicken und dann den Wert bearbeiten. Verwenden Sie diese Technik, um das Attribut mit dem Namen 'Title' in 'JobTitle' umzubenennen.

41. Sie können auch mit der rechten Maustaste klicken und die Optionen **Attribut hinzufügen** oder **Attribut entfernen** auswählen.

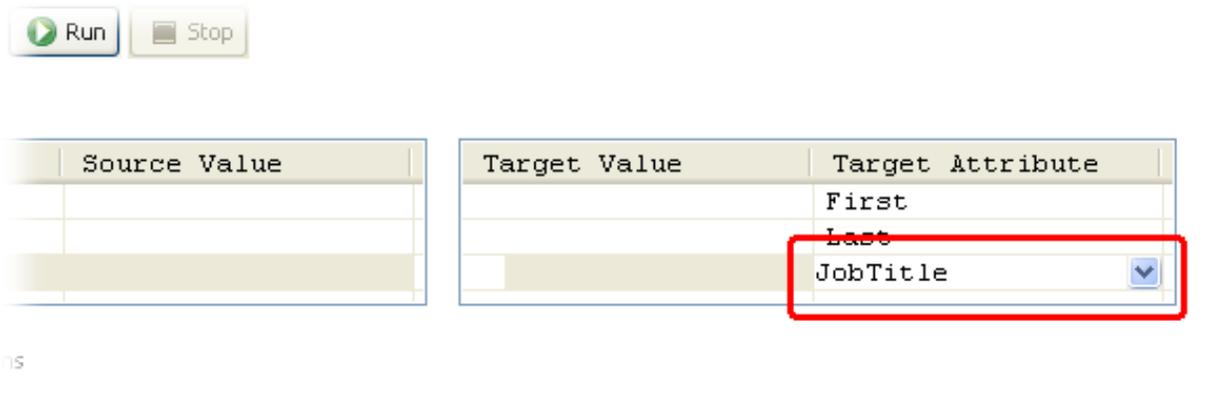


Abbildung 95. Ausgabeattribut umbenennen

Ihr EasyETL-Projekt ist jetzt für den ersten Test bereit.

EasyETL-Fertigungslinie ausführen

Wählen Sie die Ansicht **Datenkollektor** unten in der Anzeige aus und klicken Sie dann auf die Schaltfläche **Nächsten Datensatz lesen und schreiben**. Daraufhin passiert Folgendes.

1. Es kommt zu einer Verzögerung, da Ihre EasyETL-Fertigungslinie auf den aktiven Server übertragen und dort gestartet wird.
2. Der erste Datensatz wird gelesen und von Ihrer CSV-Eingabequelle geparkt. Die Daten werden sowohl im Eingabe- als auch im Ausgabegitter angezeigt.
3. Die von Ihnen für die Ausgabe ausgewählten Attribute werden geschrieben und in der Ansicht 'Datenkollektor' erfasst⁴².

Sie können Ihre ETL-Projekte immer noch testen und die Daten durch die Fertigungslinie fließen sehen, obwohl Sie noch kein Ausgabeziel ausgewählt haben.

42. Die Schaltfläche ganz links im oberen Teil der Ansicht **Datenkollektor** öffnet einen Konfigurationsdialog, in dem Sie die **Puffergröße des Datenkollektors** vergrößern können. Beachten Sie, dass Sie möglicherweise den für IBM Security Directory Integrator verfügbaren Speicherplatz erhöhen müssen, wenn Sie planen, große Datenmengen zu erfassen. Tun Sie dies, indem Sie die Stapeldatei bzw. das Script 'ibmditk' im IBM Security Directory Integrator-Installationsordner suchen und in einem Editor öffnen. Relativ weit unten befindet sich die Zeile, die 'miadmin' startet, und Sie können folgenden Text nach der Option `-vmargs` einfügen: `-Xmx512M`. In diesem Beispiel wird die Größe des IBM Security Directory Integrator-Datenspeichers auf 512 MB erhöht.

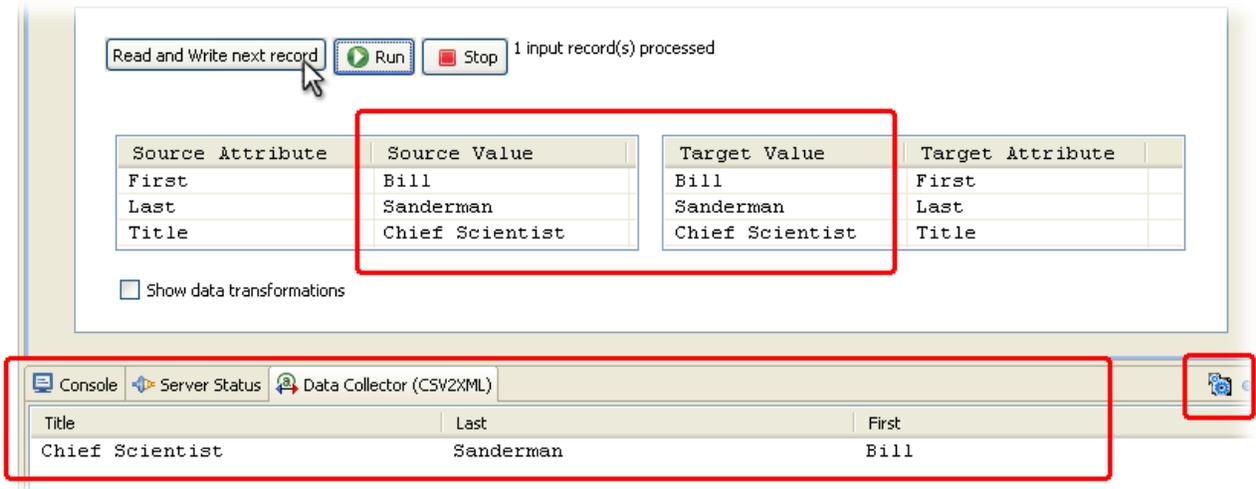


Abbildung 96. Ein Datensatz wurde gelesen und erfasst

Jedes Mal, wenn Sie auf die Schaltfläche **Nächsten Datensatz lesen und schreiben** klicken, wird ein weiterer Datensatz gelesen, angezeigt und erfasst. Wenn Sie jetzt auf die Schaltfläche 'Ausführen' klicken, wird Ihr ETL-Job bis zur Fertigstellung ausgeführt und dieser Fertigungslinienbericht wird angezeigt.

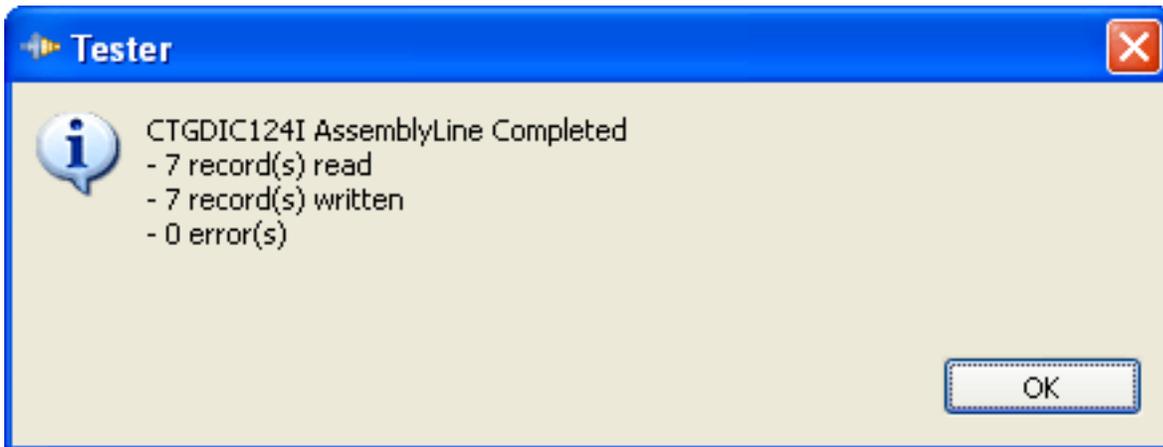


Abbildung 97. EasyETL-Fertigungslinie abgeschlossen

Wie im Dialog oben dargestellt, werden tatsächlich keine Datensätze irgendwo geschrieben. Der **Data Collector** bietet jedoch noch immer ein übersichtliches grafisch orientiertes Feedback zu den Informationen, die extrahiert und übertragen werden.

Desweiteren können Sie Zeilen in erfassten Daten auswählen und diese Daten in eine Datei oder in ein anderes Ziel kopieren oder einfügen.⁴³

43. Daten werden im CSV-Format kopiert, um den Import von Werten in Arbeitsblätter und Berichtstabellen zu vereinfachen. CSV ist die Abkürzung für Character Separated Value (durch Trennzeichen getrennter Wert) und das in IBM Security Directory Integrator verwendete Trennzeichen ist das Semikolon (;).

Konvertierungen

Bis jetzt sind Ihre Ausgabewerte mit den Eingaben identisch. Es gibt jedoch Fälle, in denen Sie diese Werte ändern oder auf Grundlage der gelesenen Daten neu berechnen möchten. Dies können Sie in IBM Security Directory Integrator EasyETL durch das Schreiben von Konvertierungen in JavaScript erreichen.

Um mit Konvertierungen zu arbeiten, müssen Sie diese durch Auswahl des Kontrollkästchens **Datenkonvertierungen anzeigen** aktivieren.

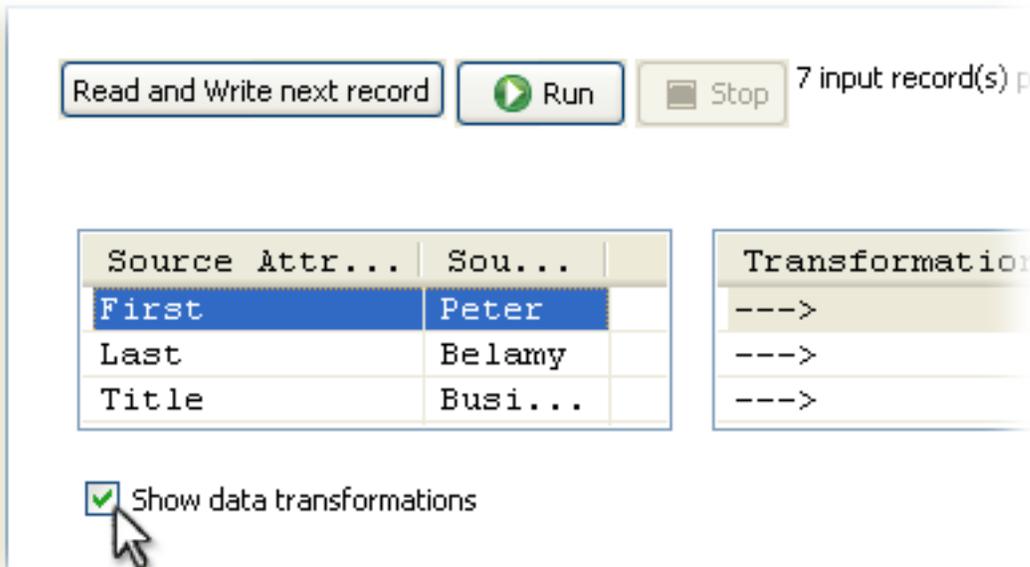


Abbildung 98. Konvertierung aktivieren

Durch diese Auswahl wird ein neues Gitterfeld zwischen Datenviewern angezeigt: der Konvertierungsvierer. Hier sehen Sie Pfeile, die anzeigen, dass alle drei Ausgabeattribute ihre Werte direkt von den Eingabeattributen erhalten - mit anderen Worten, es fand keine Konvertierung statt. Sie werden jetzt ein neues Ausgabeattribut definieren und dann das Konvertierungsscript hinzufügen, um den Wert zu berechnen. Führen Sie diese Aufgabe nun aus, indem Sie mit der rechten Maustaste in der Ausgabezuordnung klicken, 'Attribut hinzufügen' auswählen und dieses Attribut 'FullName' nennen. Klicken Sie jetzt doppelt auf die Konvertierung links vom neuen Ausgabeattribut und geben Sie das Scriptfragment ein:⁴⁴ `return First + " " + Last`

44. Denken Sie daran, dass Sie die Tasten Strg + Leerzeichen drücken können, um eine Liste mit Vorschlägen für die Eingabe zu erhalten. Diese Liste umfasst einige spezielle Objekte wie 'task' und 'main' und listet unten die Attribute auf, die von Ihrer Eingabequelle gelesen wurden.

Beachten Sie, dass die hier gezeigte Schreibweise für den Zugriff auf die Eingabeattribute nur in EasyETL funktioniert. In der IBM Security Directory Integrator-Workbench mit vollem Funktionsumfang müssen Sie einem Attributnamen ein Eingabeobjekt als Präfix vorausstellen, dass das Attribut sozusagen trägt - zum Beispiel den Work-Eintrag (work). Im Ergebnis sieht das obige Beispiel wie folgt aus:

```
return work.First + " " + work.Last
```

Der Work-Eintrag und andere IBM Security Directory Integrator-Konzepte werden im ersten Abschnitt dieses Dokuments erläutert.

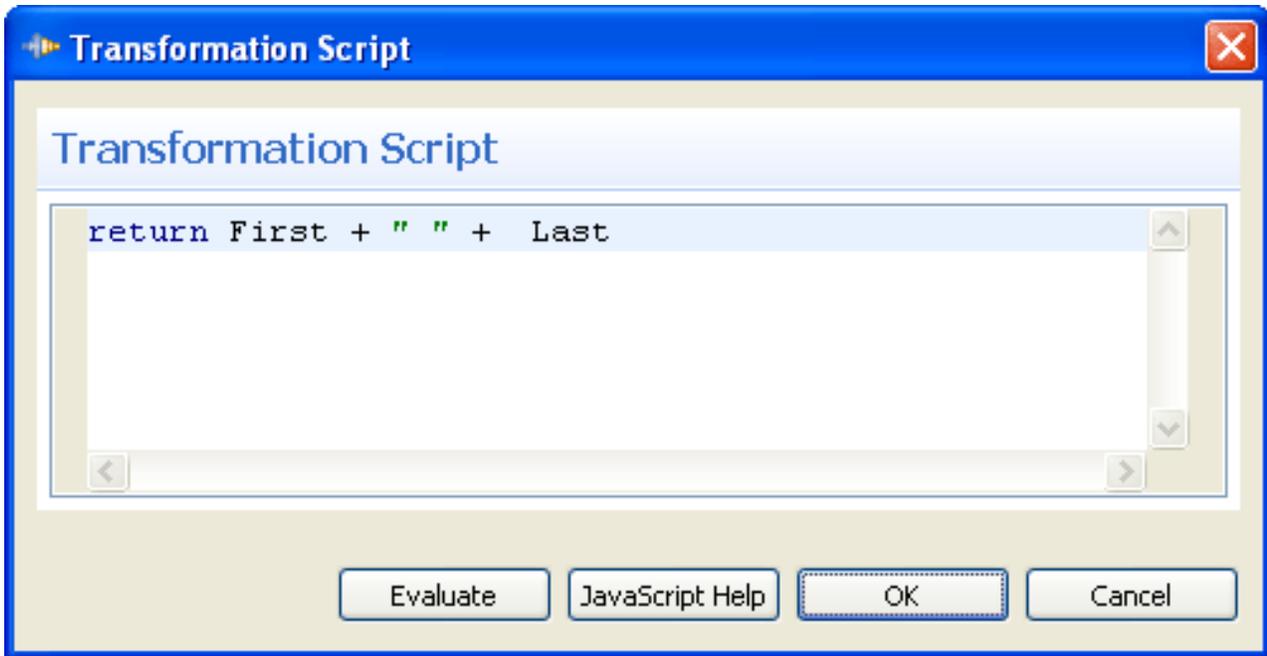


Abbildung 99. Konvertierungsscript anzeigen

Im Dialogfeld des Konvertierungsscripteditors befindet sich die Schaltfläche **Auswerten** zum Testen des Scripts sowie eine Schaltfläche für die Anzeige von Tipps und Beispielen zu JavaScript.

Klicken Sie jetzt auf die Schaltfläche **Auswerten**, um zu erfahren, wie Ihre Konvertierung funktioniert.

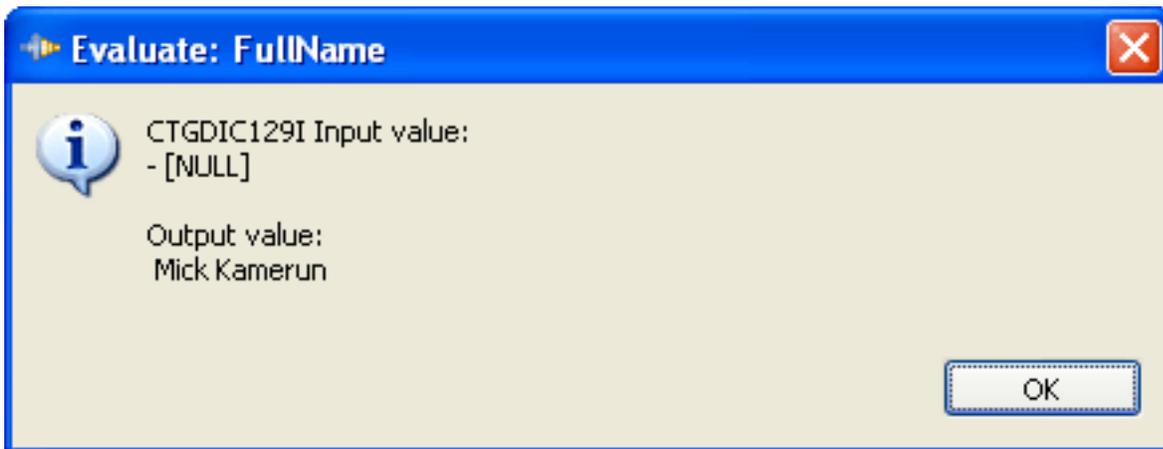
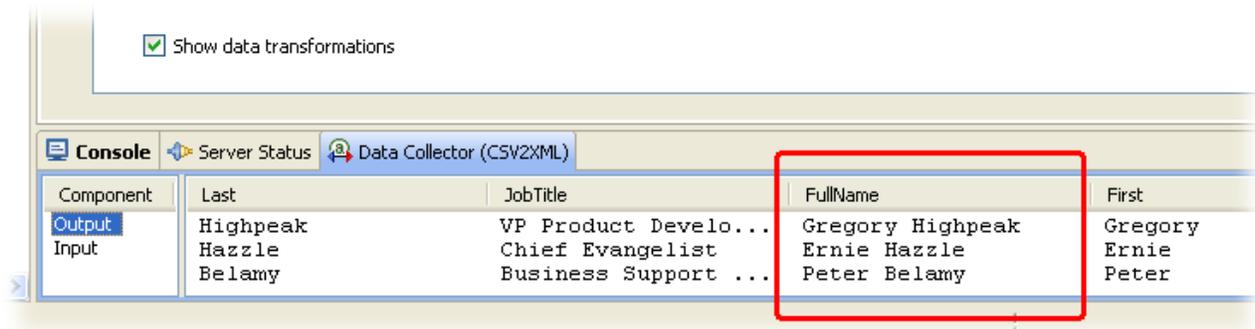


Abbildung 100. Ausdruck auswerten

Der angezeigte Ausgabewert wurde mithilfe der Daten berechnet, die Sie bei der Ausführung Ihrer Fertigungslinie erfasst haben. Schließen Sie das Dialogfenster mit den Auswertungsergebnissen, indem Sie auf **OK** klicken.

Klicken Sie jetzt auf **OK**, um dieses Script zu akzeptieren und schließen Sie den Konvertierungsscripteditor. Führen Sie Ihre EasyETL-Fertigungslinie erneut aus, indem Sie auf **Ausführen** klicken, und anschließend die erfassten Einträge anzeigen. Beachten Sie, dass der **Datenkollektor** Ihnen jetzt zwei Komponentendatenerfas-

sungen zur Auswahl anzeigt: *Ausgabe* und *Eingabe*. Wählen Sie **Ausgabe** aus und sehen Sie sich an, wie Ihr Konvertierungsscript den Wert 'FullName' für jeden Eintrag generiert hat.



The screenshot shows a console window titled 'Data Collector (CSV2XML)'. At the top, there is a checkbox labeled 'Show data transformations' which is checked. Below this, there is a table with the following data:

Component	Last	JobTitle	FullName	First
Output	Highpeak	VP Product Develo...	Gregory Highpeak	Gregory
Input	Hazzle	Chief Evangelist	Ernie Hazzle	Ernie
	Belamy	Business Support ...	Peter Belamy	Peter

A red rectangular box highlights the 'FullName' column in the table.

Abbildung 101. Ausgabedatenerfassung mit berechnetem Attribut 'FullName'

Jetzt wissen Sie, wie Sie eine Eingabequelle definieren, die Attribute zur Extraktion auswählen und wie Sie diese Daten konvertieren, damit Sie Ihre Ausgabeanforderungen erfüllen. Der nächste Schritt besteht darin, ein Ausgabeziel auszuwählen und die Daten dorthin zu übertragen.

Ausgabeziel auswählen

Wählen Sie den **Dateiconnector** für Ihr Ausgabeziel aus, indem Sie das Dropdown-Menü oberhalb der Ausgabeansicht verwenden und das Schreiben in die Datei mit dem Namen 'Output.xml' definieren.⁴⁵ Wählen Sie **XML-Parser** aus und klicken Sie auf **OK**. Beachten Sie, dass Sie auch die Schaltfläche Verbindung herstellen verwenden können, um sicherzustellen, dass der von Ihnen eingegebene Dateipfad gültig ist. Es können jedoch erst dann Daten erkannt werden, wenn Ihr Connector auf eine vorhandene XML-Datei verweist.

Sobald die Ausgabe konfiguriert ist, führen Sie Ihre ETL-Fertigungslinie erneut aus. Sobald die Ausführung beendet ist, können Sie die Ausgabedatei öffnen und die Ergebnisse überprüfen.

45. Wenn Sie keinen vollständigen Pfad angeben, sondern einen relativen Pfad, dann beginnt IBM Security Directory Integrator diesen Pfad beim Lösungsverzeichnis, das Sie während der Installation angegeben haben.

```

- <DocRoot>
  - <Entry>
    <First>Bill</First>
    <Last>Sanderman</Last>
    <Title>Chief Scientist</Title>
  </Entry>
  - <Entry>
    <First>Mick</First>
    <Last>Kamerun</Last>
    <Title>CEO</Title>
  </Entry>
  - <Entry>
    <First>Jill</First>
    <Last>Vox</Last>
    <Title>CTO</Title>
  </Entry>
  - <Entry>
    <First>Roger</First>
  </Entry>
  - <Entry>
    <First>Gregory</First>
    <Last>Highpeak</Last>
    <Title>VP Product Development</Title>
  </Entry>
  - <Entry>
    <First>Ernie</First>
    <Last>Hazzle</Last>
    <Title>Chief Evangelist</Title>
  </Entry>
  - <Entry>
    <First>Peter</First>
    <Last>Belamy</Last>
    <Title>Business Support Manager</Title>
  </Entry>
</DocRoot>

```

Abbildung 102. XML-Ausgabe

Ihr Ausgabeziel könnte ebenfalls eine Datenbanktabelle sein, so wie Ihre Eingabe auch von einer Lotus Notes-Anwendung oder einem LDAP-Verzeichnis stammen könnte. Die Schritte, die Sie zur Erstellung dieses ETL-Beispieljobs ausgeführt haben, sind unabhängig von den Systemen oder Datenspeichern, mit denen Sie arbeiten.

Änderungen erkennen

IBM Security Directory Integrator bietet eine Reihe von Komponenten zur Erkennung von Änderungen an den Eingabedaten. Zusätzlich zu den angebotenen Änderungserkennungsconnectors haben Sie ferner die Möglichkeit, die *Delta Engine* für Ihre Eingabequelle zu aktivieren.

Die Delta Engine nimmt Momentaufnahmen der Daten beim Lesen auf und vergleicht diese mit den Momentaufnahmen, die während der vorherigen Ausführung aufgenommen wurden, um die Änderungen festzustellen. Diejenigen Einträge, die nicht geändert wurden, werden übersprungen. Nur die geänderten Einträge werden für die Verarbeitung in Ihrer EasyETL-Fertigungslinie abgerufen.

Klicken Sie für Ihre Eingabequelle auf die Schaltfläche **Konfigurieren** und wählen Sie anschließend die Registerkarte **Delta** aus.

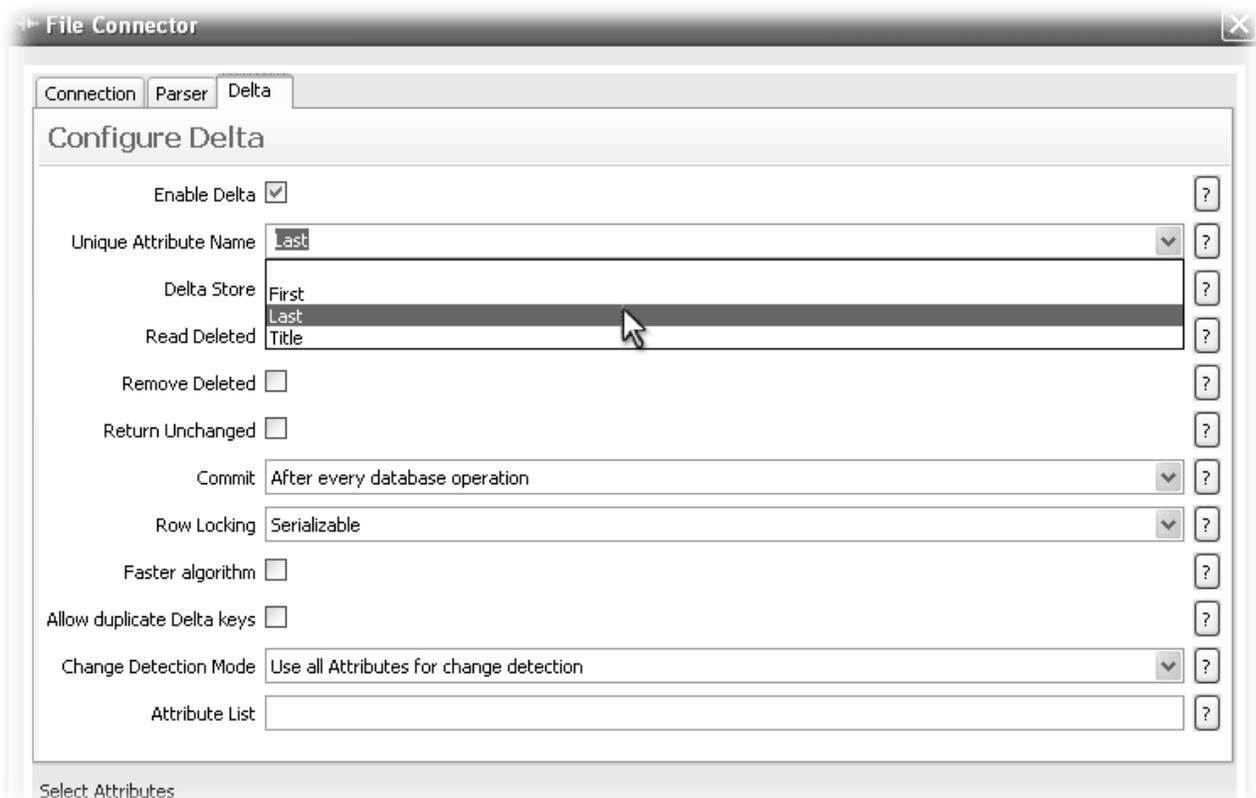


Abbildung 103. Deltakonfiguration

Sie müssen zuerst die Delta Engine aktivieren, indem Sie das Kontrollkästchen oben in der Konfigurationsanzeige auswählen. Verwenden Sie dann das Drop-down-Menü, um 'First' für **Eindeutiger Attributname** auszuwählen⁴⁶.

Es gibt mehrere andere verfügbare Parameter, die zum Teil besser für die Arbeit mit der IBM Security Directory Integrator-Standardworkbench als für die Arbeit in EasyETL geeignet sind. So kann eine EasyETL-Fertigungslinie zwar neue und geänderte Einträge erkennen und übertragen, sie kann jedoch weder eine Zeile aus einer Datenbank noch einen Eintrag aus einem Verzeichnis entfernen. Die Fertigungslinie wird diese Informationen jedoch mit dem LDIF-Parser in ein Ausgabeziel wie zum Beispiel einen Dateiconnector schreiben. LDIF-Dateien können Änderungsoperationen enthalten und einige Systeme unterstützen den LDIF-Import.

Hier können Sie mehr über die vollständigen Delta-Handling-Komponenten von IBM Security Directory Integrator erfahren:

http://www.tdi-users.org/twiki/pub/Integrator/HowTo/HowTo_SyncData_6.1.1070523.pdf

Möglicherweise möchten Sie den Parameter **Festschreiben** ändern. Dieser Parameter steuert die Festschreibung neuer und geänderter Momentaufnahmen in der IBM Security Directory Integrator-Systemspeicherdatenbank. Standardmäßig ist dieser Parameter mit 'Nach jeder Datenbankoperation' definiert und tritt so während des Lesens in Erscheinung.

Wenn Sie jedoch sicherstellen möchten, dass eine Änderung erfolgreich übertragen wurde, bevor Sie die Momentaufnahme festschreiben, legen Sie im Dropdown-Menü stattdessen 'Am Ende des FL-Zyklus' fest, sodass die Festschreibung nach der Aktualisierung des Ausgabeziels erfolgt.

Die Delta Engine benötigt für Ihre Arbeit eine Gruppe von Ausgangsmomentaufnahmen. Sie erstellen diese Momentaufnahmen, wenn Sie Ihren ETL-Job das erste Mal ausführen, nachdem die Delta Engine aktiviert wurde. Nachdem der Job beendet wurde, werden Sie bemerken, dass der Popup-Bericht doppelt so viele Schreibvorgänge aufweist. Dies liegt daran, dass IBM Security Directory Integrator auch die Momentaufnahmen zählt, die in den Systempeicher geschrieben werden, sodass Sie für jeden verarbeiteten Eintrag zwei Schreibvorgänge erhalten.

Versuchen Sie, Ihre EasyETL-Fertigungslinie erneut auszuführen. Sie werden feststellen, dass dieses Mal keine Einträge geschrieben werden. Die Delta Engine erkennt, dass die Eingabedatensätze alle unverändert sind und überspringt sie.

46. Wie Sie sich vielleicht schon denken können, verwendet die Delta Engine eines Ihrer Eingabeattribute, um die Momentaufnahmen eindeutig zu kennzeichnen. Wenn es keinen eindeutigen Wert in den Eingabedaten gibt, können Sie mehrere Attribute angeben, die miteinander verkettet werden, und so die ID der Momentaufnahme bilden. Sie können dies erreichen, indem Sie die Namen mehrerer Attribute eingeben und jeweils ein Pluszeichen (+) dazwischen setzen. Zum Beispiel: First + Last

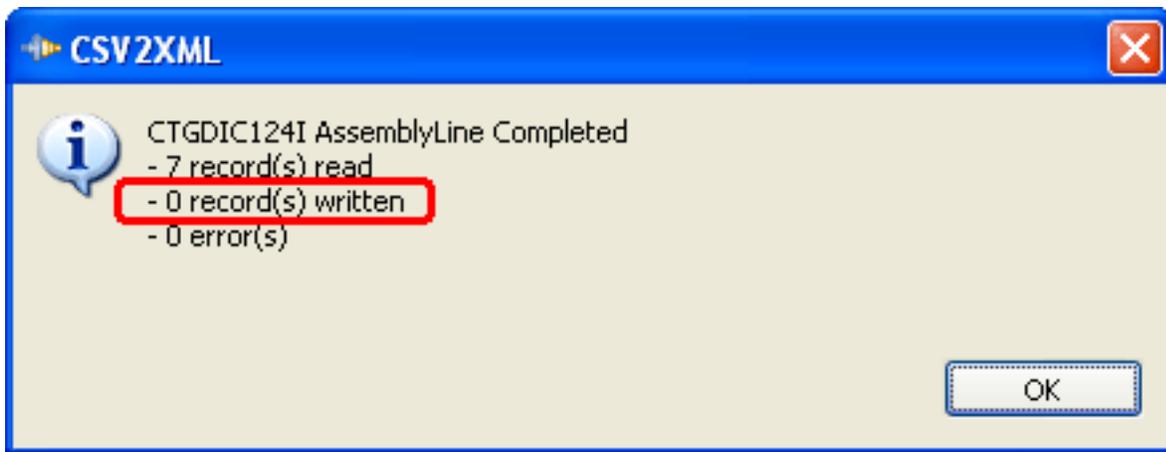


Abbildung 104. Alle Einträge sind unverändert und werden übersprungen

Als letzten Test öffnen Sie die CSV-Eingabedatei und ändern einen beliebigen Feldwert - außer den Wert für 'Last'⁴⁷. Speichern Sie die Änderungen und führen Sie Ihren ETL-Job erneut aus. Sie werden feststellen, dass nur geänderte Einträge verarbeitet werden.

Ausgabeziel für Aktualisierung konfigurieren

Die aktuelle Konfiguration funktioniert bestens für die Ausgabe in eine Datei. Wenn Sie diese Änderungen jedoch in ein Verzeichnis, RDBMS oder ähnliche Datenspeicher übertragen möchten, dann werden Sie sowohl neue Daten hinzufügen als auch vorhandene Datensätze aktualisieren wollen. Damit Ihr EasyETL-Job diese Aufgabe erfüllt, müssen Sie zuerst auswählen, welches Ausgabeattribut Sie als Kriterium für die Lokalisierung des zu ändernden Datensatzes verwenden wollen.

Diese Auswahl erfolgt durch Klicken mit der rechten Maustaste auf das gewünschte Ausgabeattribut und die Auswahl der Option **Als Verknüpfungsbedingung verwenden**.

47. Dies ist das Attribut, das für die Identifizierung von Momentaufnahmen verwendet wird und Änderungen an diesem Wert würden dazu führen, dass der Eintrag für die Delta Engine als *neuer* Datensatz erscheint.

record(s) processed

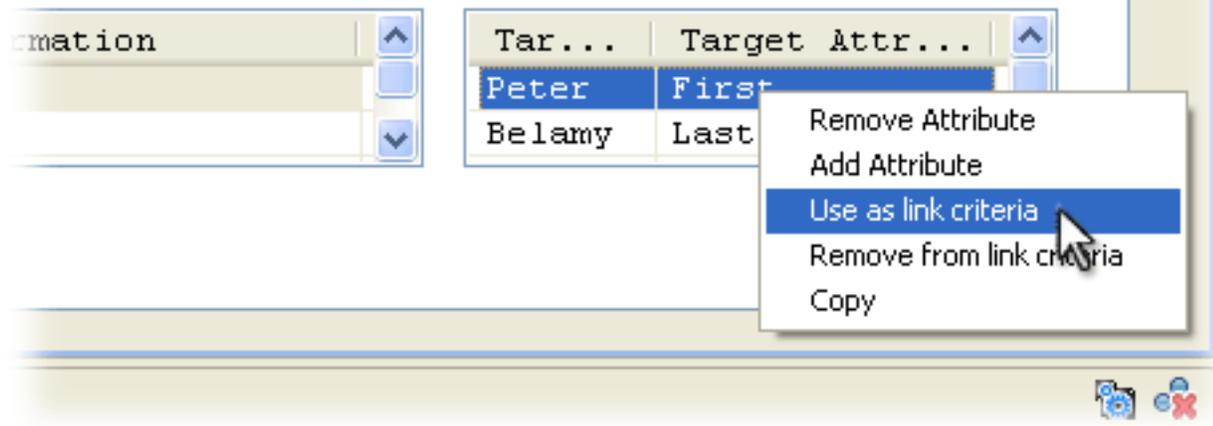


Abbildung 105. Verknüpfungsbedingungen auswählen

Wenn der Ausgabeconnector jetzt in das Ziel schreibt, sucht er zuerst mithilfe des angegebenen Attributs für die Verknüpfungsbedingungen nach einem Datensatz. Falls keine Übereinstimmung gefunden wird, wird ein neuer Eintrag hinzugefügt. Falls eine Übereinstimmung gefunden wird, wird der betreffende Datensatz aktualisiert.

Das Konzept ist ganz einfach: Ihr ETL-Job wurde jetzt so konfiguriert, dass er eine kontinuierliche Synchronisation zwischen Ihrer Eingabequelle und dem Ausgabeziel zur Verfügung stellt.

Befehlszeilenressource zur Ausführung und Planung Ihres ETL-Jobs

Sobald Ihre ETL-Fertigungslinie für die Implementierung bereit ist, können Sie im Navigator mit der rechten Maustaste auf das Projekt klicken und die Option **Dateien erstellen, die für die Ausführung dieses Projekts bzw. dieser Fertigungslinien in der Befehlszeile erforderlich sind.** auswählen.

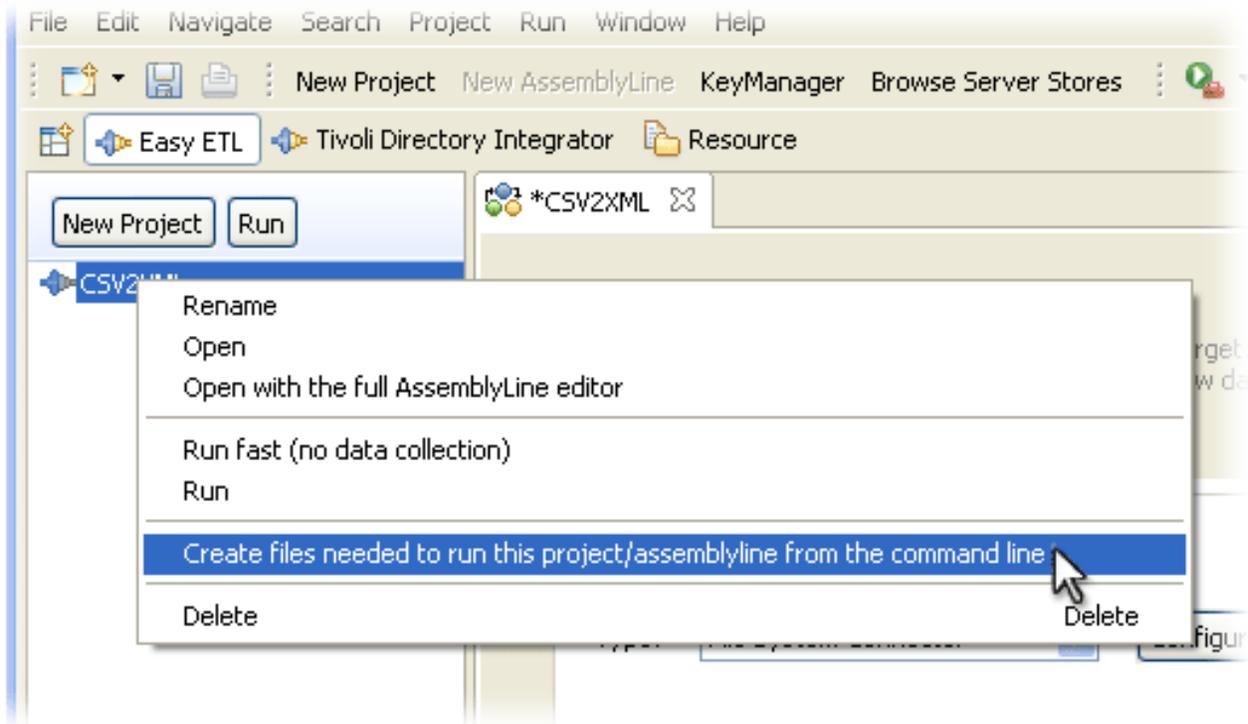


Abbildung 106. Befehlszeilenressourcen für die Ausführung des ETL-Jobs erstellen

Das Dialogfenster 'Dateien exportieren' wird mit Informationen darüber angezeigt, wohin diese Script-/Stapeldatei geschrieben werden soll. Beachten Sie, dass derselbe Name wie für das Projekt verwendet wird, sodass bei der Ausführung dieser Übung des Lernprogramms unter Windows der Name 'CSV2XML.bat' verwendet wird. Die Ausführung Ihres EasyETL-Projekts von der Befehlszeile bietet die maximale Leistung für Ihre Lösung.

Ferner wird eine XML-Datei an derselben Position erstellt. Diese Datei wird als IBM Security Directory Integrator-Konfigurationsdatei (*Config*) bezeichnet. Sie enthält die Details Ihrer EasyETL-Fertigungslinie, die der IBM Security Directory Integrator-Server für die Ausführung benötigt. Wenn Sie das generierte Script in einem Texteditor öffnen, sehen Sie den Einzeiler, der für das Starten des IBM Security Directory Integrator-Servers erforderlich ist, verweisen damit auf eine Konfigurationsdatei (*Config*) und geben dann die Fertigungslinie an, die ausgeführt werden soll. Sie müssen lediglich eine geplante Task oder einen CRON-Job konfigurieren, um dieses Script regelmäßig aufzurufen, damit Ihre Synchronisations-/Migrationsservices aktuell bleiben.

Weitere Optionen

Hochgeschwindigkeits-ETL

Obwohl der Datenkollektor ein leistungsfähiges Tool ist, wird Ihre ETL-Fertigungslinie langsamer ausgeführt, wenn die Datenerfassung und -darstellung am Bildschirm erfolgt. Wenn Sie möchten, dass Ihre ETL-Fertigungslinie so schnell wie möglich ausgeführt wird, dann können Sie entweder das Projekt auswählen und auf die Schaltfläche 'Ausführen' oben im Navigator klicken oder Sie können mit der rechten Maustaste auf das Projekt klicken und die Option **Schnellausführung (keine Datenerfassung)**

auswählen.

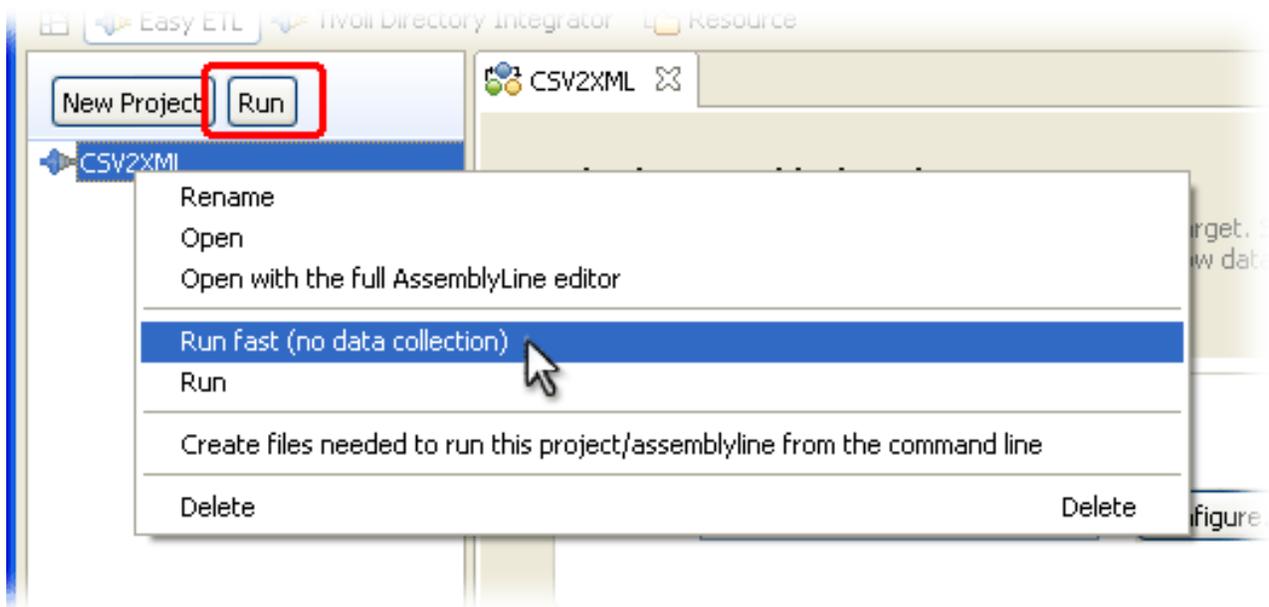


Abbildung 107. ETL-Job mit voller Geschwindigkeit ausführen

Beide Möglichkeiten führen dazu, dass eine Konsolanzeige angezeigt wird, in der Protokollnachrichten Ihrer Fertigungslinie erscheinen, während Ihre Fertigungslinie mit voller Geschwindigkeit ausgeführt wird.

Beachten Sie, dass die Option **Ausführen** im Projektkontextmenü den ETL-Job mit Datenerfassung ausführt.

Eingabedateien filtern

Eine weitere leistungsfähige Komponente ist die Möglichkeit, den Inhalt Ihrer Eingabedateien zu steuern. Diese Option ist immer dann verfügbar, wenn Ihre Eingabequelle eine Datenbank oder ein Verzeichnis ist.

Wählen Sie zum Beispiel 'LDAP-Connector' als Eingabe aus und sehen Sie sich den Konfigurationsdialog für diese Komponente an. Neben dem Parameter **Suchfilter** befindet sich eine Schaltfläche mit drei Punkten (...). Diese Schaltfläche öffnet den Editor für Verknüpfungsbedingungen, in dem Sie Suchregeln definieren können, die bei der Erstellung der Ergebnismenge angewendet werden, die dieser Connector lesen soll.

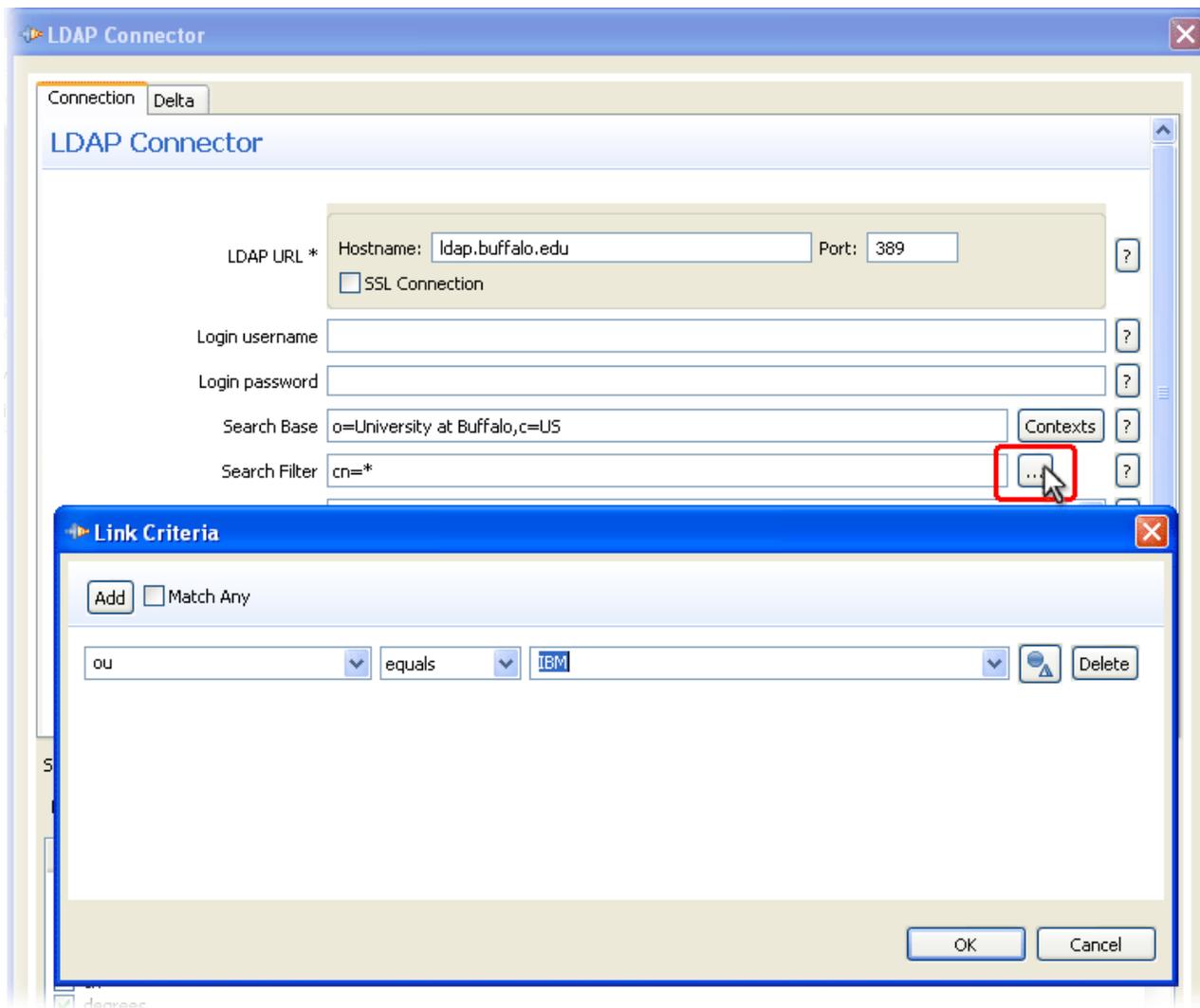


Abbildung 108. Verknüpfungsbedingungen für einen Eingabconnector definieren

Dieselbe Funktion ist für die Datenbank- und JDBC-Connectors verfügbar. Auch hier finden Sie die Schaltfläche mit den drei Punkten (...) neben dem Parameter **Auswählen**.

Obwohl Sie die LDAP-Suchsyntax selbst direkt in den Suchparameter eingeben können, müssen Sie die Syntax für die LDAP-Suchfilter oder die JDBC-Auswahanweisungen (SELECT) kennen. Es ist häufig einfacher, die gewünschte Auswahl durch Verknüpfungsbedingungen auszudrücken und den Connector die zugrunde liegende Syntax bearbeiten zu lassen.

Ihre EasyETL-Fertigungslinie auf die nächste Ebene stufen

Wenn Sie Ihr ETL-Projekt im voll ausgestatteten IBM Security Directory Integrator-Fertigungslinieneditor öffnen, können Sie angepasste Protokollierung und Überwachung, Fehlerbehandlung, Funktionsübernahmelogik, automatische Wiederverbindung, Datenerweiterung (Joins) und viele weitere Elemente zu Ihrer Migrations- oder Synchronisationslösung hinzufügen. Sie können dies erreichen, indem Sie mit der rechten Maustaste auf ein Projekt klicken und die Option **Im Fertigungslinieneditor mit vollem Funktions-**

umfang öffnen auswählen. Sie arbeiten dann weiterhin in der EasyETL-Workbench, können aber auf weitere verfügbare Funktionen für Ihre Fertigungslinie zugreifen.

Wenn Ihnen diese Umgebung gefällt, können Sie auch in die Perspektive 'Security Directory Integrator' wechseln (**Fenster > Perspektive öffnen > Security Directory Integrator**) und in der vollständig ausgestatteten IBM Security Directory Integrator-Workbench arbeiten. Es empfiehlt sich dann jedoch, nachdem Sie EasyETL jetzt beherrschen, erneut mit Abschnitt 1 zu beginnen und das gesamte Leistungsspektrum von IBM Security Directory Integrator im Detail kennen zu lernen.

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieser Dokumentation ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

IBM Director of Licensing
IBM Europe, Middle East & Africa
Tour Descartes
2, avenue Gambetta
92066 Paris La Defense
France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in diesen Informationen nicht ausgeschlossen werden. Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuausgabe veröffentlicht. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Corporation
2Z4A/101
11400 Burnet Road
Austin, TX 78758 U.S.A.

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Dokument aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedingungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer kontrollierten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können davon abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Aussagen über Pläne und Absichten von IBM unterliegen Änderungen oder können zurückgenommen werden und repräsentieren nur die Ziele von IBM.

Alle von IBM angegebenen Preise sind empfohlene Richtpreise und können jederzeit ohne weitere Mitteilung geändert werden. Händlerpreise können u. U. von den hier genannten Preisen abweichen.

Diese Veröffentlichung dient nur zu Planungszwecken. Die in dieser Veröffentlichung enthaltenen Informationen können geändert werden, bevor die beschriebenen Produkte verfügbar sind.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind und Programmier Techniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Beispielpprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für

die diese Beispielprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit IBM Anwendungsprogrammierschnittstellen konform sind.

Kopien oder Teile der Beispielprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

© (Name Ihrer Firma) (Jahr). Teile des vorliegenden Codes wurden aus Beispielprogrammen der IBM Corporation abgeleitet. © Copyright IBM Corp. _Jahr/Jahre angeben_. Alle Rechte vorbehalten.

Marken

IBM, das IBM Logo und ibm.com sind Marken oder eingetragene Marken der International Business Machines Corp. in den USA und/oder anderen Ländern. Weitere Produkt- und Servicenamen können Marken von IBM oder anderen Unternehmen sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite "Copyright and trademark information" unter www.ibm.com/legal/copytrade.shtml.

Adobe, Acrobat, PostScript und alle auf Adobe basierenden Marken sind Marken oder eingetragene Marken der Adobe Systems Incorporated in den USA und/oder anderen Ländern.

IT Infrastructure Library ist eine eingetragene Marke der Central Computer and Telecommunications Agency. Die Central Computer and Telecommunications Agency ist nunmehr in das Office of Government Commerce eingegliedert worden.

Intel, das Intel-Logo, Intel Inside, das Intel Inside-Logo, Intel Centrino, das Intel Centrino-Logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium und Pentium sind Marken oder eingetragene Marken der Intel Corporation oder ihrer Tochtergesellschaften in den USA und anderen Ländern.

Linux ist eine Marke von Linus Torvalds in den USA und/oder anderen Ländern.

Microsoft, Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

ITIL ist eine eingetragene Marke, eine eingetragene Gemeinschaftsmarke des Office of Government Commerce und eine eingetragene Marke, die beim US Patent and Trademark Office registriert ist.

UNIX ist eine eingetragene Marke von The Open Group in den USA und/oder anderen Ländern.



Java und alle auf Java basierenden Marken und Logos sind Marken oder eingetragene Marken der Oracle Corporation und/oder ihrer verbundenen Unternehmen.

Cell Broadband Engine wird unter Lizenz verwendet und ist eine Marke der Sony Computer Entertainment, Inc. in den USA und/oder anderen Ländern.

Linear Tape-Open, LTO, das LTO-Logo, Ultrium und das Ultrium-Logo sind Marken von HP, der IBM Corp. und von Quantum in den USA und/oder anderen Ländern.

Index

A

Absicherung 83
Anwendungsmanagementkonsole 71
Appender 85
Attribut 5
Attributzuordnung 7, 15
Ausbildung ix
Ausführen 29
Ausführungsfehler 66
Ausgabebezuordnung 7
Ausnahmebedingungen 86

B

Befehlszeile 71
Behandlung von Ereignissen 74

C

Conn 7

D

Daten 5
Daten suchen 47
Datenfluss 7
Debug 39
Debugger 90

E

Eingabehilfen ix
Eingabebezuordnung 7
Eintrag 5
Ereignis 71
Ereignisse 71
Erste Schritte 8

F

Fehlende Daten 32
Fehlerbehebung ix
Fehlertoleranz 83
Fertigungslinie 7, 15, 29

H

Hinzufügen 8

I

IBM
 Software Support ix
 Support Assistant ix
IBM Security Directory Integrator-Projek-
te 1
Initialisierungsfehler 86

Iterieren 8

J

JavaScript 1

K

Konnektivität 86
Konsole 72

L

Leistung 88
Lernprogrammdateien 1
Lesbarkeit 83
log4j 85
Lookup 8
Lösungen, die IBM Security Directory In-
tegrator verwenden 3
Lösungen entwerfen 1, 3

M

Mit sehr einfachen Lösungen begin-
nen 3
Modus 'Lookup' 58, 65

N

Nullverhalten 32
Nullwerte 32

P

Problembestimmung ix
Protokollausgabe 29
Protokollierung 85

R

Reagieren 71

S

Scheduler 72
Schulung ix
Sicherheit 83
Skalieren 88
Stepper 39
Suche 65

T

TINA 74
Trigger 71

U

Übereinstimmung 65
Unterbrochene Verbindung 86

V

Vererbung 63
Verfügbarkeit 83, 86
Verknüpfen 47
Verknüpfungsbedingungen 65
Verwaltung 83

W

Web-Server 74
Wert 5
Wiederverwendung 83

Z

Zeitplanung 72



GI11-3250-03

