



ENOVIA SmarTeam

SmarTeam – Foundation Administration Guide

© Dassault Systèmes, 2008, 2010. All rights reserved.

CATIA, ENOVIA, SMARTEAM and the 3DS logo are registered trademarks of Dassault Systèmes or its subsidiaries in the US and/or other countries.

PROPRIETARY RIGHTS NOTICE: This documentation is the property of Dassault Systèmes. This documentation shall be treated as confidential information and may only be used by employees or contractors of the Customer in accordance with the terms of the End-User License Agreement accepted by Customer.

Any use of the Licensed Program contained in this media or accompanying it, is subject to the terms of the End User License Agreement accepted by Customer. The Licensed Program is protected by international copyright laws and international treaties. Unauthorized use, reproduction and/or distribution of any of the Licensed Program, or any part thereof, may result in severe civil and/or criminal penalties, and will be prosecuted to the maximum extent possible under the law. Company names and product names mentioned herein are the property of their respective owners and certain portions of the Licensed Program contain elements subject to copyright owned by these entities. See the Documentation CD provided with the Licensed Program for details and/or additional terms and conditions relating to these entities.

Part Number: FDN-A1-200410

Contents

Chapter 1: Introduction	1
Overview	1
SmarTeam Vaults	1
SmarTeam – Workflow	2
SmarTeam Full Text Search	2
SmarTeam Report Connector	2
SmarTeam System Configuration Editor	2
Related Documentation	2
Internet Site	3
Chapter 2: SmarTeam Vault Concepts	4
Vault Security	4
SmarTeam Vault	4
Vault Maintenance	5
Vault Replication Concepts	6
File Access Permissions	6
Licensing	6
File Security	6
Accessing Remote Files Locally	7
Vault Configuration	7
Vault Replication and Efficient Vault Replication	8
File Naming Convention	9
Replication on Demand	9
Vault Replication Independent of Database Replication	10
Selective Vault Replication	10
Setting Up Vault Replication with RDS™	10
Lifecycle API	11
Chapter 3: Setting Up a Vault Site	12
Vault Server Setup Checklist	12
Prerequisites	12
Installation Verification	12
Preparing the Vault Server Environment	12
Setting Up a Vault Server	13
Configuring the Vault Server	13
Enabling the Vault Server	21
Testing the Vault Server	23

Chapter 4: Multi-Vault Environment	25
SmarTeam – Multi-Vault Replication Setup Checklist	25
Configuration of Vaults for a Site	25
Vault Replication Mechanism (in DMD)	26
Add Vault Replication Mechanism	26
Vault Server Setup	26
Setting Up Vault Sites	26
Vault Groups Replication Setup	34
Chapter 5: Vaults Security	41
HTTP and HTTPS	41
SSL	41
Chapter 6: SmarTeam – Workflow	43
SmarTeam – Workflow	43
Setting Up the SmarTeam – Workflow Server	45
Preparing the Flow Server Environment	46
Installing the Flow Service	46
Chapter 7: Full Text Search	48
SmarTeam Full Text Search Environment	48
SmarTeam Full Text Search Keys	48
SmarTeam Full Text Search Components	49
Recommended Configurations	49
Microsoft Indexing Service File Formats Support	51
Installing Full Text Search	52
Indexing Other File Types in Microsoft Indexing Service	55
Microsoft Indexing Service Performance Tuning	56
Rescanning FTS Directories After Adding an IFilter	60
Changing the Default Settings of SmarTeam Core Services	60
Microsoft Installer	61
Chapter 8: SmarTeam Report Connector	62
What is the Purpose of SmarTeam Report Connector?	62
Data Flow	62
Data Structure	63
Setup Prerequisites	63
Software and Hardware Requirements	63
Installation of the SmarTeam Report Connector Utility	64
Security	64
Licensing	64
Implementation	64
Desktop Applications	64
Web Applications	65
In-house Development Applications	65
Data Retrieval Methods	65

Implementation Recommendations	65
Typical Consumers	66
Connecting to External Report Generators	67
Defining a Default Database Connection	67
Connecting to Crystal Reports	68
Connecting to Visual Basic	75
Connecting to Excel	80
SQL-92 Format	84
Chapter 9: SmarTeam System Configuration Editor	88
Introduction	88
Overview	88
Upgrading from Previous SmarTeam Versions	88
Terminology	89
Configuration Key (Element)	89
Configuration Set (Header)	89
Configuration Files	89
Configuration Key Hierarchy	93
Using the System Configuration Editor	96
Accessing the System Configuration Editor	96
System Configuration Main Page	96
View Types	96
Implementation	99
Key Types	99
Connecting SmarTeam Applications to System Configuration Service through a Firewall	100
Mapping Pre-V5R13 Repositories	100
Transporting System Configuration	102
Adding Complex Keys to the System Configuration Service	104
Adding a New Key to the System Configuration	107
Writing to the System Configuration Service using SmarTeam API (session smConfig)	107
Defining the Default Connection for the SmarTeam Database for Multiple Clients	108
Using Individual Configurations for Different Users	108
Manually Editing System Configuration XML Files	109
System Configuration Service	109
Configuration Schema	109
Configuring the System Configuration Service to Work with the Windows Protocol	109
Configuring Core Services on a Multi-Network Card Machine	111
Configuring IIS 64bit to Work with an Application Running at 32bit	112
XML Troubleshooting	112
Configuring Session Management	113

Chapter 10: Vault Redundancy Core and Flow Setup on MSCS 2003	114
Clusters	114
What is a Computer Cluster?	114
What is Microsoft Cluster Server?	114
Setting up SmarTeam Vault Server on MSCS	114
Setting up SmarTeam Core Services on MSCS	117
Setting up SmarTeam Workflow Services on MSCS	119
 Chapter 11: System Configuration Keys	 121

Chapter 1: Introduction

Overview

This guide outlines the various administrative procedures required to successfully set up and maintain [SmarTeam Vaults](#), [SmarTeam – Workflow](#), [SmarTeam Full Text Search](#), [SmarTeam Report Connector](#) and [SmarTeam System Configuration Editor](#) in a corporate environment.

Note: All the documentation mentioned in this document, unless specified otherwise, is available on the SmarTeam Documentation CD.

SmarTeam Vaults

Maintaining security and control over documents is of the utmost importance to any organization. SmarTeam provides a secure electronic vault for this purpose. This electronic vault ensures that only users with access permission can access an object. The system is secured at all access levels.

Traditionally, a document manager performed the role of maintaining security. Files were stored in a secured library and an employee would copy out a file, work with it and then copy the file back in. The document manager would supervise the flow of files, ensuring that an employee had authorization to access a file, and that no two people would work with the same file simultaneously.

SmarTeam provides a Vault Server Setup utility (for Windows environment) that enables you to define vaults, implement the Vault Server process and set up security modes for the vaults.

SmarTeam users can work on a remote Web server that is connected to a site, such as Site A, and access a vault server at a different site, such as Site B, which is local to the user. In this way data is saved on the local vault.

In addition to this, SmarTeam provides a logging mechanism, which displays SmarTeam activities including vault activities.

Note: This guide is based on using a local vault at every site, which is what SmarTeam recommends. It is also possible to work with remote vaults, however this could result in reduced performance and is not the preferred method of work.

SmarTeam – Workflow

The **Flow Server** controls the flow of Processes between users in the SmarTeam - Workflow environment. The Flow Server constantly monitors the database at predefined time intervals and checks whether any Process is ready to be moved to the next user (node). The Flow Server utilities enable you to set up and run the Flow Server for your network.

SmarTeam Full Text Search

SmarTeam Full Text Search enables you to perform complex textual searches on data, such as within the vaults and on textual metadata stored inside the database without knowing where that data is stored. Complex textual searches include Boolean expressions (**AND**, **OR**) as well as phonetic operators, such as **sounds like**.

SmarTeam Report Connector

SmarTeam Report Connector enables administrators to securely retrieve information from SmarTeam and generate customized reports using standard third-party report generators, such as Crystal Reports®, Microsoft® Visual Basic (VB), and Microsoft® Office Excel. The report tool connects to the SmarTeam database through a designated OLE DB provider.

SmarTeam System Configuration Editor

The System Configuration Editor provides a centralized mechanism that contains all configuration-related information for all SmarTeam applications. The system configuration service has multiple levels of configuration allowing easy manageability and security across sites, machines, applications, databases and users from anywhere in the organization.

Related Documentation

The following documents are referred to in this guide. All the documents are available on the SmarTeam Documentation CD unless specified otherwise.

Name of Document	Remarks
Introduction to SmarTeam Installation	It is recommended that you read this document thoroughly and plan your topology prior to installing your SmarTeam configuration or products.
SmarTeam Procedure for Upgrading	Details the upgrade procedure if you are upgrading from a previous version of SmarTeam - Foundation.
Hardware and Software Requirements	Details the hardware and software required for a successful installation.

Name of Document	Remarks
SmarTeam – Foundation Installation Guide	Details about installing SmarTeam Core Services
SmarTeam – Editor Installation Guide	Details about the SmarTeam – Editor installation process
SmarTeam - Multi-site Administration Guide	Details the various administration procedures to successfully set up, customize and maintain a SmarTeam - Multi-site system in a corporate environment.

IMPORTANT!

For details of the minimum Hardware and Software requirements, see the ENOVIA SmarTeam Hardware and Software Requirements Guide.

Internet Site

You are highly recommended to frequently visit our website for the latest updates and plug-in products, including the latest Service Packs, Program Directory (Release Notes), Hotfixes and Technical Support at <http://www.3ds.com/support/>.

In addition, you will also be able to view any installation known issues.

Chapter 2: SmarTeam Vault Concepts

Vault Security

The SmarTeam Vaults security is handled as follows:

- Domain users do not have read/write permission to the Vaults folders
- Dedicated users who run the vault service have read/write permission to the Vaults folders
- Using a User command that transfer files via Vault service
- The Vault service uses the http or https protocols
- The Vault service requests are handled with encrypted messages (time-based ticket) for each transfer

SmarTeam Vault

The electronic vault is divided into three sections. Each section reflects a different state of an object. SmarTeam provides a framework for defining three default directories where files are stored based on state only. You can change these default directories as described in [Vault Maintenance](#).

■ Checked In

The purpose of the **Checked In** vault is to provide a secured environment for objects that are in a dynamic development stage. Objects may be checked out, modified and checked in again by users with the correct access permissions. Objects are placed in this vault using the **Check In** revision option. Files can be checked out of this vault using the **Check Out** revision option. The default directory for this vault is: ..\SmarTeam\Vaults\Checked In.

The Default file size permitted for **Check In** and **Check Out** is 1GB. This value is sufficiently large for most implementations.

Although the file size can be set up to 2⁶³ bytes, it is recommended that you do not customize the file size more than the actual required size due to the risk of denial-of-service attack (DoS attack).

If you attempt to Check In or Check Out files larger than the configured size, an error message will be displayed.

To increase the maximum file size that may be checked in:

- 1 Open the configuration file SmarTeam.Std.Vault.Server.Host.exe.config located in the SmarTeam\Bin directory.

This file contains many instances of the parameter **<maxReceivedMessageSize>**.

- 2 Change the value of the parameter only where the parameter appears with the two binding names listed below:

VaultHttpStreamedBinding VaultHttpStreamedBindingSsl

- 3 Important: Do not modify any other parameters other than the ones specified.

To increase the maximum file size that may be checked out:

- 1 Open the configuration file smarteam.std.vaultClient.config.xml.
- 2 Update the value for every instance of the parameters **<MaxBufferSize>** and **<MaxReceivedMessageSize>**.

■ Released

The purpose of the **Released** vault is to provide a secured environment for objects that have been approved, protecting them from being erroneously modified. This vault might have stricter security measures, meaning fewer users will be able to access the objects in this vault. Objects are placed in this vault by activating the **Release** operation. The **New Release** operation allows you to create a new revision of the document and modify it. This option may be restricted only to high-level users. The default directory for this vault is `..\SmarTeam\Vaults\Released`.

■ Obsolete

The purpose of this vault is to provide a secured environment for objects that are no longer in use. Objects are placed in this vault using the **Obsolete** operation, and these objects can be accessed by SmarTeam – Editor users but cannot be changed. The default directory for this vault is `..\SmarTeam\Vaults\Obsolete`.

Lifecycle operations are described in detail in SmarTeam – Editor Online Help. In the *Revisions* page you can view the lifecycle of an object. This refers to the different revision states of an object. Each time you create a new revision, it is displayed on a new line on this page.

Vault Maintenance

SmarTeam enables you to define different vaults for different types of files. Furthermore, you can define separate vaults for individual projects (or groups of projects). For example, in the **Checked In** vault, you can define that all Word documents are saved in a directory called `..\SmarTeam\Checked In\Word`, all AutoCAD files are saved in a directory called `..\SmarTeam\Checked In\Autocad` and all other files remain in `..\SmarTeam\Vaults\Checked In`.

Note: The system administrator generally performs this operation.

The **Vault Maintenance** option enables you to:

- Specify a new directory in which a file type from a vault is stored. For example, you may specify a new directory path for all Word documents located in the **Released** vault.
- Delete a vault definition.
- Modify the path of a vault definition.

Vault Replication Concepts

Vault Replication concepts relate to converting an existing legacy Vault Configuration into one that can participate in a SmarTeam – Multi-site system or just a Multi-Vault environment with one central database. This process includes:

- Creating a Vault Configuration for a Site
- Setting up a Vault Replication

For additional information, see [Multi-Vault Environment](#).

File Access Permissions

In a Vault Replication environment, the remote Vault Server copies the required files to the local mirror vault. In this case, the SmartVault Server user must be a global user with full permissions to access to all vault directories, where files are hosted as well as any remote vaults.

Licensing

The Vault Server Setup administration utility must have access to a SmarTeam – Multi-site (MUS) license or SmarTeam - Multi-Vault (MUV) license. When using a Multi-site environment the MUS license covers the Vault license. When only working with Multi-Vault an MUV license is required.

All SmarTeam licenses are located on the License Use Management (LUM) server at the Primary Site.

IMPORTANT! While Database Multi-site tools must be run from the Primary Site, the Vault Replication Setup tool must be run from every site involved in Vault Replication. Since the licenses are only located on the Primary Site, Vault setup must have access to these licenses from the site on which it is running.

To set up Vault Server access between sites:

- 1 Open a firewall rule between the sites.
- 2 Implement a VPN between the two sites.

Access from the Vault Server machine to the Primary Site must be established temporarily for configuration purposes. After the Vault Server Setup utility closes, there is no need for the Vault Server service to have access to the LUM server.

File Security

Unlike conventional file copy operations, the file replication server compresses files prior to transferring and decompresses them after they arrive in the corresponding mirror vault. The transaction between sites can be encrypted using conventional encryption methods, to ensure security.

Accessing Remote Files Locally

SmarTeam – Multi-Vault Replication enables the SmarTeam – Editor user to send and retrieve files from a local vault using a local Vault Server, regardless of the user's location. The file's local entity is retrieved from a local vault. This is possible because files that originate in remote vaults are copied to a mirrored vault on all sites.

Because all files exist on all sites, roaming users can access files from any site. If one site merges with another, all users from the merged site can log in to the database from the new site and continue working.

Vault Configuration

The following components exist in a Vault Configuration:

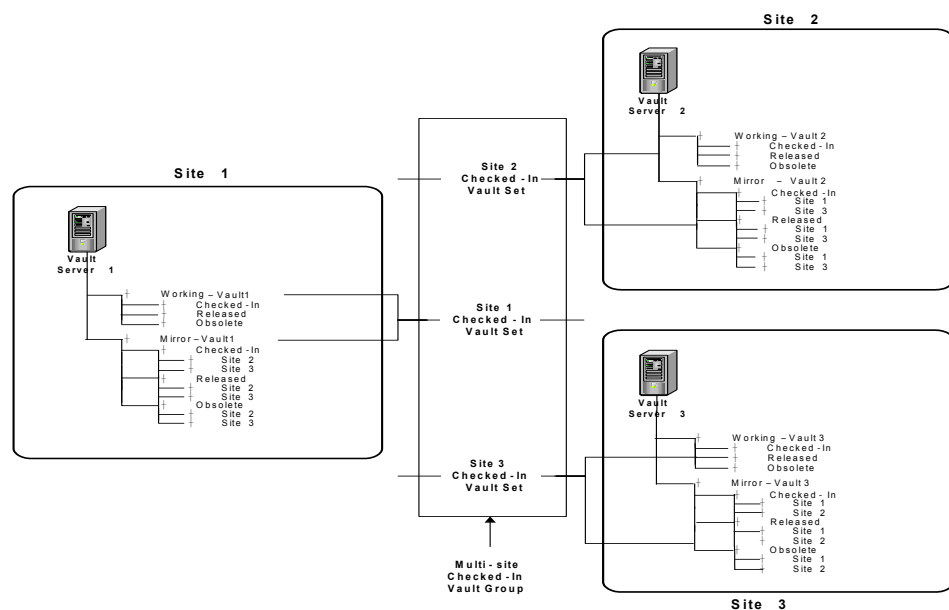
- **Site:** Identifies the current site to differentiate it from the other sites in the SmarTeam – Multi-site environment
- **Vault Server (Node):** A computer in the site, which contains vaults for the site
- **Working Shared Directory:** Shared directory on the Vault Server, which contains working vaults
- **Working Vaults:** Subdirectories of the Working Shared Directory, which contain working files: \Checkin, \Released, \Obsolete.
- **Mirror Shared Directory:** Shared directory on the Vault Server, which contains mirror vaults
- **Mirror Vaults:** Subdirectories of the Mirror Shared Directory: \Mirror_Checkin, \Mirror_Released, \Mirror_Obsolete. These directories do not contain files; they organize the mirror vaults into operational types. The actual replicated files are stored in subdirectories of these directories according to source sites.
- **Mirror Vault Site Subdirectory:** Subdirectories of the Mirror Vaults contain replicated files from other sites. The subdirectory names should be the same as the corresponding working vault directory names with the <MySite> subdirectory, for example: \Checkin, \Released, \Obsolete for all sites.

Vault Replication and Efficient Vault Replication

After the vault configuration is created, you must perform the Vault Replication setup procedure. This procedure basically organizes your vaults into operational groups that span the sites.

The organization of vaults into groups is used by an application to quickly and efficiently locate a mirror directory on the local machine containing a copy of a desired file. The original copy of the file is on a remote machine.

The groups normally correspond to the Check In, Released and Obsolete operations. For example, the Check In group contains all vaults from all sites that relate to the Check In operation, including both working and mirror vaults.



Vault configuration is designed to produce a very efficient Vault Replication for multiple-vault systems.

If a SmarTeam – Multi-Vault system contains many vaults to be replicated, it is not efficient to define a separate replication job for each vault, since each job creates its own thread. Instead, the replication system should replicate many vaults in one job. This arrangement significantly reduces the number of jobs required in complex systems.

The goal is to replicate all vaults of a working shared directory to all vaults of a mirror shared directory in one recursive job. To accomplish this, the vault structure in the mirror root directory must be identical to the vault structure in the source root directory.

File Naming Convention

To avoid conflicting filenames within vaults from different sites, it is recommended to provide a unique filename when performing the first Check In lifecycle operation for a file.

Note: By default, any file retains its original name when performing the first Check In lifecycle operation. Therefore, to successfully change the filename during the first Check In operation, perform the procedure described in the File Naming Configuration section of the SmarTeam – Multi-site Administration Guide.

For each lifecycle operation on a file already located in a vault, a unique filename is assigned to it by default.

Replication on Demand

Sometimes it is necessary to view a file owned by a remote site immediately without waiting for its scheduled replication.

You can use either of these options:

- The Remote Copy Feature is the default SmarTeam option. It is accessed by using a standard Windows XCOPY operation that copies the files from a remote site. A disadvantage of this method is that it functions relatively slowly and does not permeate firewalls.

OR

- RepliWeb-based functionality enables fast copying of files through firewalls, which use script hooks to attach the SmarTeam stdCopyFileFromRemoteSite.ebs script to the After - On File Exists On Local Mirror event (the Before setting is also possible). The stdCopyFileFromRemoteSite.ebs script is located in the SmarTeam\SDK\Samples\MultiSite\BasicScripts directory after installing the SmarTeam – Editor SDK component. The stdCopyFileFromRemoteSite.ebs script uses the rw_fetch_file.dll file to create a RepliJob, which in turn calls the script to replicate the required file. Each time you need to view a file that has not been replicated from a remote site, the file is copied to the mirror directory on the local site.

Notes:

- This procedure requires RepliWeb® Console to be installed on the client computer. For details, see the RepliWeb Installation and Configuration Guide.
- When using RepliWeb® it is necessary to customize the script provided by SmarTeam.
- The script (stdCopyFileFromRemoteSite) calls the RepliWeb® DLL function from the rw_fetch_file.dll with parameters for the source, destination and user authentication parameters.

- RepliWeb® is firewall-friendly application, does not affect system operation and for the end user, the operation is executed transparently.

Vault Replication Independent of Database Replication

SmarTeam – Multi-Vault also provides Vault Replication capabilities, which is completely independent from database replication and can be implemented separately.

Site is also used with Vault Replication, but may not correlate with the database site definition. Moreover, there can be more vault sites than database sites, depending on the customer's environment and requirements.

Selective Vault Replication

As opposed to database replication (see SmarTeam – Multi-site Administration Guide), you can set up Selective Vault Replication by selecting which file does and does not replicate. Select one of the following options for Selective Replication:

- Create special vaults that are responsible for saving files, which are not to be replicated. You must save them in the Vault Replication framework, but you do not have to set up RDS jobs for replication in respective mirrors.

OR

- You can maintain the same vaults, but you must manually configure RDS jobs to replicate a selected set of files. This list of files needs to be provided by the SmarTeam application using scripts.

Setting Up Vault Replication with RDS™

When purchasing a SmarTeam – Multi-Vault license or SmarTeam - Multi-site license, customers receive an RDS™ (RepliWeb® Deployment Suite) package.

For each designated work vault there should be a parallel vault on all sites. To enable users to access files originated on remote sites, each work vault from any site should have a corresponding mirror vault on all other sites.

After mapping all mirrored vaults in the Vault Server setup utility that correspond with their work vaults, a user can issue a mapping report that is used by the administrator to perform the equivalent mapping in the RDS™ utility. RDS™ is responsible for the actual replication of files between the vaults in the various sites.

All work vaults should have mirrored vaults. If a customer does not want files created for a specific project on a certain site to be accessed from other sites, the following requirements should be met:

- A designated vault should exist for that project
- The designated vault should not be replicated, although a mirror vault is assigned to it

A typical user environment may include two sites, where each site is working on a different project. There may be collaborative work where users from one site are working on objects that belong to projects originated on the other site. In some situations, for example, CAD files and office files

need to be located in different vaults. For quick and easy access to information, replicated databases are also implemented. In this scenario, all vaults should have corresponding mirror vaults on all sites and each site should have its own local database server and a local Vault Server.

Time difference between sites, availability of required information, the amount of data that needs to be transferred and the available bandwidth all need to be measured to establish the correct time interval between replications.

Note: Vault replication can be applied with the RepliWeb R-1 product as well.

Lifecycle API

The access points where an implementer can customize a lifecycle process are:

Customization related to File Operation that should be assigned in **Before – Life Cycle Stage 2** for Hooks:

- Check In
- Release
- Obsolete

Customization related to File Operation that should be assigned in **After – Life Cycle Stage 2** for Hooks:

- Check Out
- New Release
- Copy File

If a script was previously hooked to a different hook, you must update the scrip according to Life Cycle Stage 2 methodology, see Client-Side Hooks for Client-Based Applications.pdf on the SmarTeam Documentation CD.

Chapter 3: Setting Up a Vault Site

Vault Server Setup Checklist

You must complete all the stages in this checklist to successfully setup and enable a SmarTeam Vault Server.

*Requirement: M = Mandatory, O = Optional

	Item	M/O*	Reference
Stage 1: Pre-Setup			
<input type="checkbox"/>	Verify that your Hardware & Software meet the requirements	M	SmarTeam Hardware and Software Requirements Guide
<input type="checkbox"/>	Verify SmarTeam – Foundation is installed and configured properly at each site	M	SmarTeam – Foundation Installation Guide
<input type="checkbox"/>	Verify Vault Server is installed as part of SmarTeam – Foundation installation or on a separate machine	M	SmarTeam – Foundation Installation Guide
<input type="checkbox"/>	Assign permissions to the Vault Server user	M	Preparing the Vault Server Environment
<input type="checkbox"/>	Check for any additional prerequisites on the SmarTeam Web Site	M	Release Notes of latest service pack in the release or SmarTeam Support Site
Stage 2: Setup			
<input type="checkbox"/>	Set up SmarTeam Vault Server	M	Setting Up a Vault Server
<input type="checkbox"/>	Test SmarTeam Vault Server	M	Testing the Vault Server

Prerequisites

Installation Verification

Before setting up a SmarTeam vault site, verify the following is installed on your system:

- Vault Server is installed as part of SmarTeam – Foundation installation or on a separate machine

For more information, see the SmarTeam – Foundation Installation Guide.

Preparing the Vault Server Environment

To assign permissions to the Vault Server user:

Note: This procedure is performed after installing the Vault Server utilities.

- 1 From **Start**, select **Programs > Administrative Tools > User Manager** to view the User Manager window.
- 2 From the **Policies** menu, select **User Rights** to view the User Rights Policy window.
- 3 Enable **Show Advanced User Rights**.
- 4 Next to the **Right** field, click on the dropdown list and select **Log on as a Service**. Click **OK**.
The **Grant To** list box shows the User/Group that receives the selected user rights. The name of the Vault Server user (or the Group that the user belongs to) appears.
- 5 If the appropriate User/Group is not displayed in the Grant To list box, click **Add** to display a list of Users/Groups. Select the Vault Server user and click **OK**. The Vault Server username appears in the **Grant To** list box.
- 6 Next to the **Right** field, click on the dropdown list and repeat the previous steps for **Act as part of the operating system** and **Increase Quotas**.

The system administrator can create a Vault Server log file during runtime for listing all vault activities for review by the system administrator. (By default a log file is not created.)

Note: A Vault Server user must have full permissions to all vault directories, where files are hosted.

Setting Up a Vault Server

A Foundation server must be installed at every site where a Vault Server exists. There can be more than one Vault Server per foundation server, which do not have to be hosted on the same machine.

IMPORTANT! This section is only applicable for sites licensed for using the SmarTeam Vault Server.

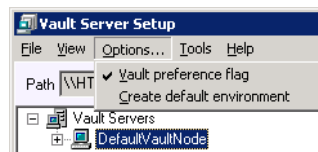
Configuring and enabling the vaults is done by the administrator in the following stages:

- [To configure a default environment](#): is done from the Vault Server Setup window via the Admin Console to define the vaults.
- OR**
- [To configure vaults manually](#): is done manually from the Vault Server Setup window via the Admin Console to define the vaults.
- [Enabling the Vault Server](#) is done from the SmarTeam – Editor so you can work with vaults once they are configured.

Configuring the Vault Server

To configure a default environment:

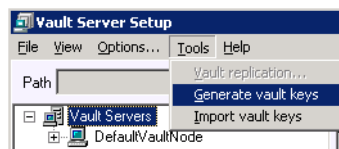
- 1 From the Admin Console, log in as an administrator.
- 2 Select Vault Management > Vault Setup.
The Vault Server Setup window appears.



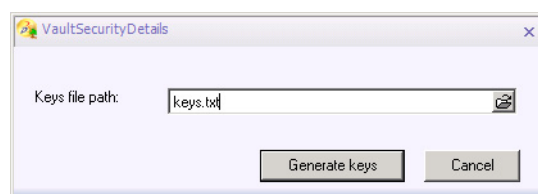
- 3 Select Options > Create default environment.
A confirmation dialog box appears.



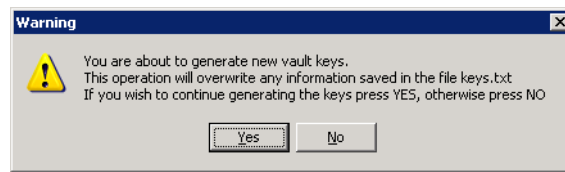
- 4 Click **OK**.
- 5 From the Vault Server Setup window, select Tools > Generate vault keys.



- 6 In the VaultSecurityDetails dialog box, type the vault keys path and filename.
The default path is: C:\Program Files\SmartTeam\Bin

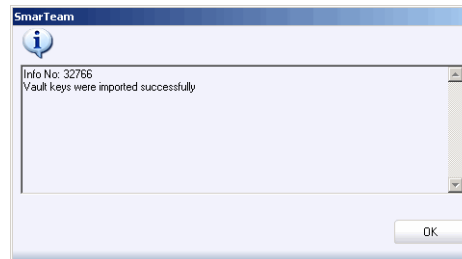


- 7 Click **Generate keys**.
A dialog box appears confirming you want to continue the process of generating the keys.



- 8 Click **Yes** to continue.

A dialog box appears confirming the vault keys were generated successfully.

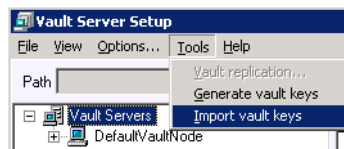


- 9 Click **OK**.

- 10 Run **SmarTeam.Std.Vault.ImportKeyTool.exe** from the command line on each Vault Server to set up additional servers using the following syntax:

**SmarTeam.Std.Vault.ImportKeyTool.exe /U:<UserName> /P:<Password>
/R:<DatabaseReplicaId> <KeyFile>**

- 11 From the Vault Server Setup window, select Tools > Import vault keys.

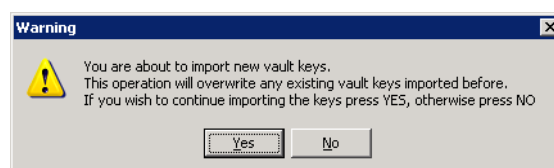


- 12 In the VaultSecurityDetails dialog box, type the same filename you used in [Step 6](#).



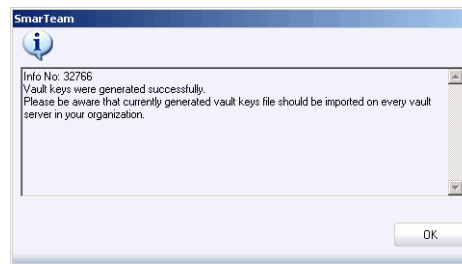
- 13 Click **Import keys**.

A dialog box appears confirming you want to continue the process of importing the keys.



14 Click **Yes**.

The following confirmation dialog appears.



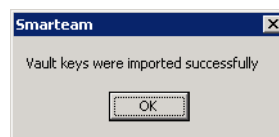
Note: If there is a multi-vault configuration on your environment, you should generate the vault keys from only one vault (see [Step 5](#)), but the import should be made from all the vaults on your environment (see [Step 11](#)).

Note: When a vault machine domain membership is changed or a user account under which the vault service runs is changed, do the following:

- Delete ..\Vault\PersistencyStorage folder, such as C:\Documents and Settings\All Users\Application.
- Repeat [Step 11](#) to import the key.

15 Click **OK** to continue.

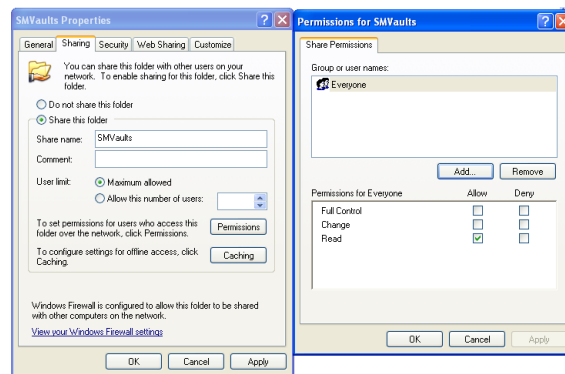
A dialog box appears confirming the vault keys were imported successfully.

**16** From the Vault Server Setup window, select File > Save.

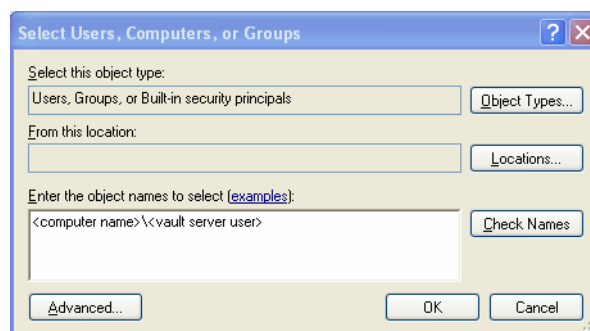
Note: Although this step is optional, because it will occur before the vault setup closes down if it is not performed now, SmarTeam recommends you save any updates now.

17 Exit the Vault Server Setup window.**18** Exit the Admin Console.**To configure vaults manually:**

- 1** Create the following directories on a local machine:
 - a** C:\SmVaults
 - b** C:\SmVaults\Checked In
 - c** C:\SmVaults\Released
 - d** C:\SmVaults\Obsolete
- 2** Right click **SMVaults** > **Properties**.
- 3** From SMVaults Properties, select **Sharing** > **Share this folder** > **Permissions** to enable sharing in C:\SmVaults.



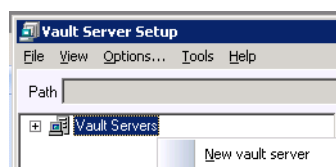
- 4 In Permissions for SMVaults, select **Add** to include the Vault Server user and then click **OK**.



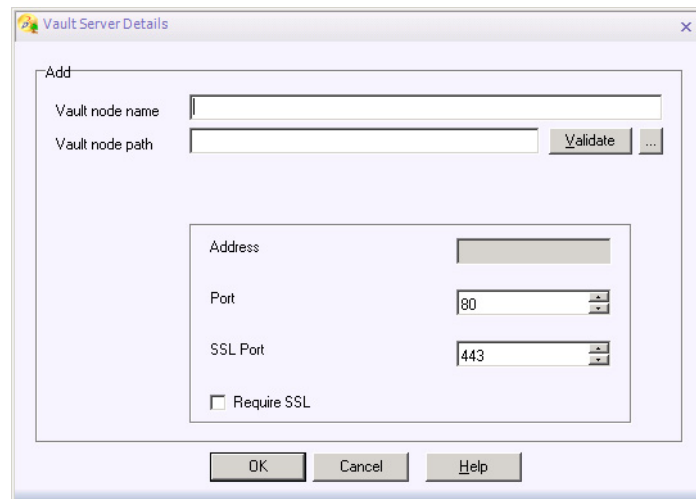
Note: This step is optional, because all folders do not have to be shared unless the Vault Server and directories are located on a different computer than the one from which you are running the vault setup.

- 5 From the Admin Console, log in as an administrator.
- 6 Select Vault Management > Vault Setup.
The Vault Server Setup window appears.

Note: The default SmarTeam Vault Server structure is customizable.



- 7 Select **New Vault Server**.
The Vault Server Details window appears.



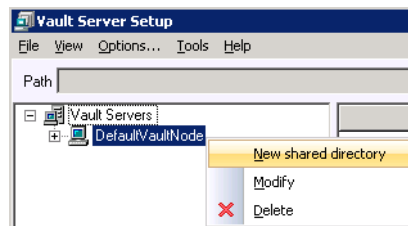
The Vault Server Details window contains the following fields:

Field	Description
Vault node name	Type a name for the Vault Server.
Vault node path	Type the path in the network to the location where the Vault Server is installed. If applicable, click Browse to display a standard file selection window.
Validate	Either type or browse to the machine name in the Vault node path field, and click Validate to verify the Vault node path exists in the network. If it is found, the Vault node path is marked with an underscore. If not, an error message appears.
Address	Vault Server address.
Port	The default value is 80 for http communication with the vault. This value can be changed as required.
SSL Port	The default value is 443 for https/SSL communication with the vault. This value can be changed as required.
Require SSL	Select this field if you want to work in secure SSL mode.

8 Click **OK**.

Note: The default values of 80 and 443 are optional values and can be modified. The administrator can select any ports as long as the port and the SSL port are different. For more information, see [Chapter 5, Vaults Security](#).

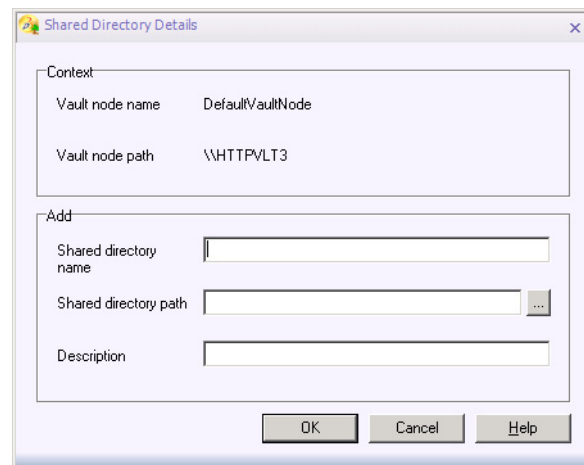
9 Right click on the machine name and select **New shared directory** to create a shared directory on each of the servers (nodes) that contains the Vault directories.



Note: This step is optional. The directory does not have to be shared and the name of the directory does not have to be the same name on every server.

- 10 Browse to the shared vault folder you created in [Step 2](#).

The Shared Directory Details window appears.



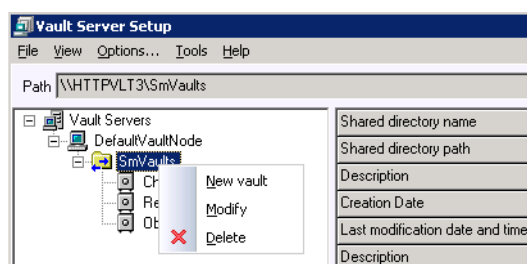
The Shared Directory Details window contains the following fields:

Field	Description
Name	Type a name for the shared directory.
Shared directory	Type the directory name in the Vault Server that will serve as a shared directory. If necessary, click Browse to display a standard file selection window.
Description	Type descriptive text (optional).

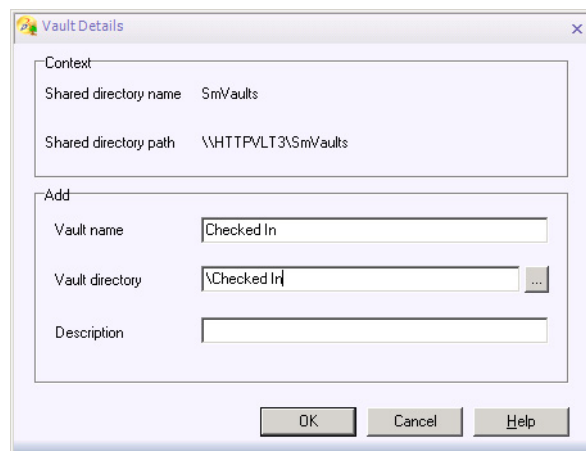
- 11 Complete the fields in the Shared Directory Details window and click **OK**.

The information you typed in these fields can be modified or deleted.

- 12 Right click on the shared directory and select **New vault**.



- 13** Browse to the relevant vault directory.
The Vault Details window appears.



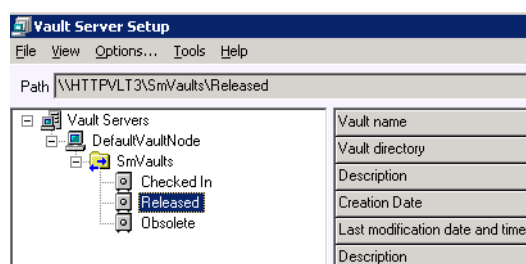
The Vault Details window contains the following fields:

Field	Description
Name	Type a name for the vault.
Vault directory	Type the network path for the vault location. If necessary, click Browse to display a standard file selection window.
Description	Type descriptive text (optional).

- 14** Complete the fields in the Vault Details window for the Checked In directory path. Repeat this process for Released and Obsolete directories.

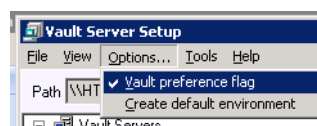
Note: The location of the vault directories can be modified.

The final configuration of the Vault Server Setup window appears.



- 15** Follow [Step 5](#) to [Step 15](#) to generate and import vault keys.
16 From the Vault Server Setup window, select File > Save to save the updates.

Note: It is possible to enable the Vault Server by selecting Option > Vault preference flag.-



- 17 Exit the Vault Server Setup window.
- 18 Exit the Admin Console.

Enabling the Vault Server

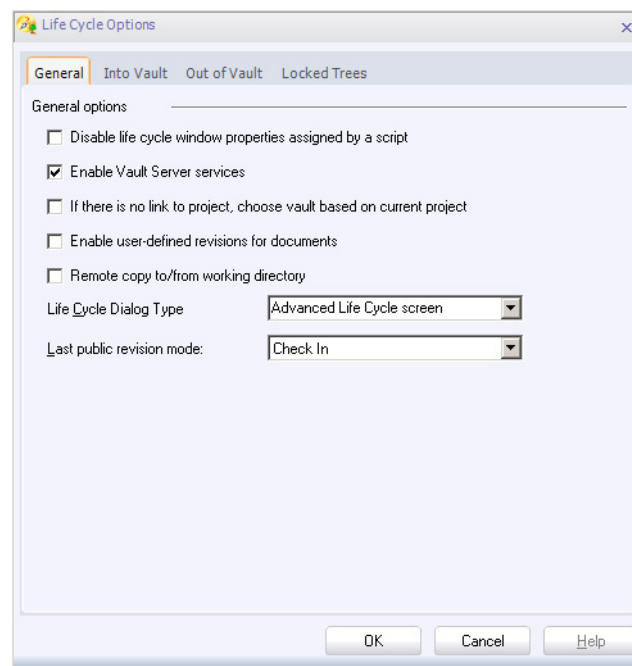
This procedure is done from SmarTeam – Editor.

Notes:

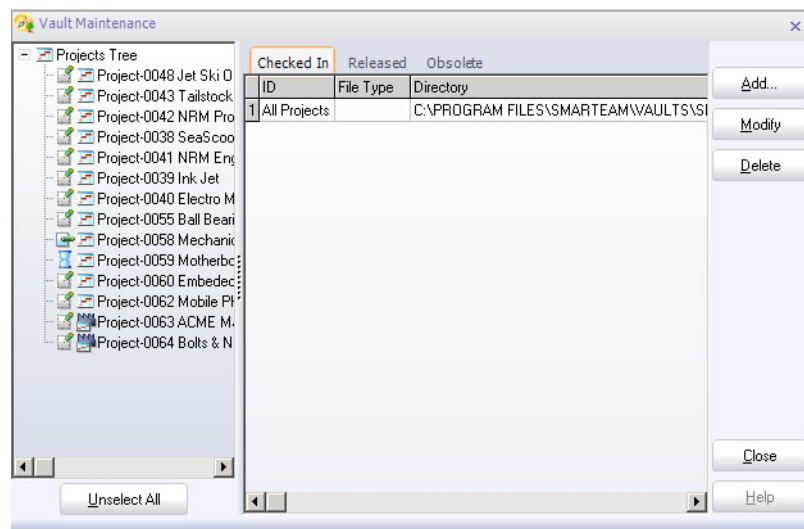
- a If you already enabled the Vault Server using the Vault preference flag in the [To configure vaults manually](#) procedure, skip to [Step 5](#) in this procedure.
- b When a vault is defined per project only, there is no default vault for "All projects" in the Vault Maintenance utility, a document that saved in the context of the parent object that has no link to a project will be placed in the vault defined for the parent object.

To enable the Vault Server:

- 1 From SmarTeam – Editor, log in as an administrator.
- 2 From the SmarTeam – Editor main window, select **Tools > Administrator Options**.
- 3 From the Administrator Options window, select **Lifecycle Options**.
- 4 From the Lifecycle Options window, select **Enable Vault Server services**.



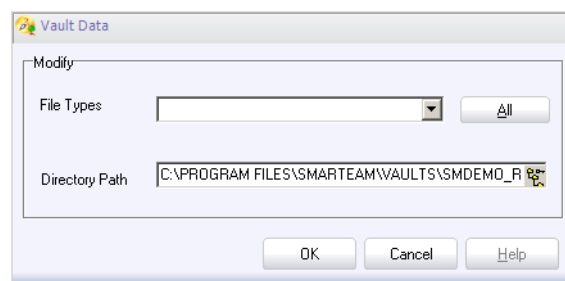
- 5 From the SmarTeam – Editor main window, select **Tools > Vault Maintenance**. The Vault Maintenance window appears.



- 6 Select one or several projects from the Projects Tree in the left pane to define directories for the files belonging to these projects.

Note: To define vaults for all projects simultaneously, select **Projects Tree**. To deselect all projects, click on the **Unselect All** button.

- 7 Click one of the tabs at the top of the window – **Checked In**, **Released** or **Obsolete** – to select a vault. You can now define directories for the files located in the selected vault.
- 8 To define a new vault for a specific file type, click **Add**.
The **Vault Data** window appears.



- 9 In the File Type field, click on the dropdown list to view a list of file types and select one.
- 10 In the **Directory** field, type a directory path to specify where the selected file type will be placed. If necessary, click on the **Browse** button to the right of this field to select a directory from a standard Windows selection window.
- 11 Click **OK** to save your changes and close the window or click **Cancel** to close the window without making any changes.
The path definition is added to the Vault Maintenance window.
- 12 To define a directory into which all other file types will be placed, in the **Vault Data** window, click **All**. If necessary, click **Browse** to select a directory from a standard Windows selection window.
- 13 To redefine an existing directory or file type, select the existing entry and then click **Modify**.
The **Vault Data** window enables you to redefine the applicable entry.

- 14 Click **OK** to save your changes and close the Vault Maintenance window.

Testing the Vault Server

Running the Vault Tester utility on the client computer enables the administrator to check that the Vault Server is working properly and that files are properly transferred.

The following occurs as a result of running the Vault Tester utility:

- The test file moves from the first vault to a second vault.
- The test file is copied to the desktop of the Vault Tester.

By default, the Vault Tester runs this cycle of events five times, but you can determine the number of cycles to run. You can also determine the exact delay between operation within a cycle and between cycles.

To define Vault Tester parameters:

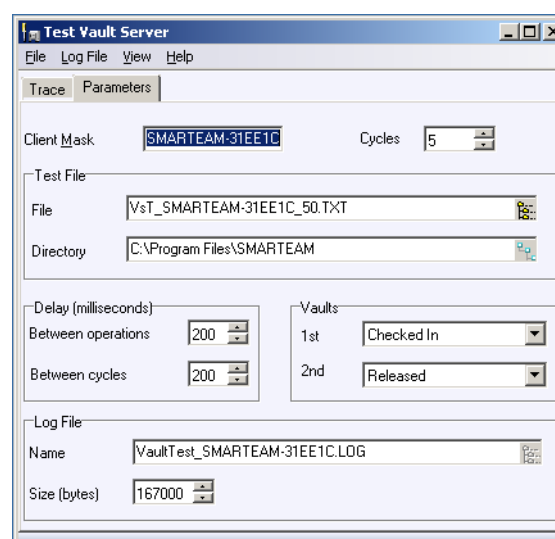
- 1 From Start, select Programs > SmarTeam > Administrative Tools > Admin Console > Vault Management > Vault Tester.

The Vault Tester main window appears.

Note: Only an administrator can run this utility.



- The **Trace** tab displays a log of the operations that take place when the Vault Tester is executed.
 - The **Parameters** tab enables you to define the parameters of the test.
- 2 Click the **Parameters** tab and define the test parameters.



You can use the default values or define new values.

Field	Description
Client mask	The Client Mask serves as an identity name of the Vault Tester to communicate with the Vault Server. By default, this field displays the computer name where the Vault Tester is located, although this value can be changed.
Cycles	Displays the number of cycles to be performed.
Test file/ Directory	Displays the name of the test file and the directory in which it is located. You can select a different test file and browse to a different location.
Delay between operations	Displays in milliseconds the amount of delay between operations (within a cycle).
Delay between cycles	Displays in milliseconds the delay between cycles.
Vaults 1st	During runtime, the test file is copied from the desktop (of the Vault Tester) to the vault defined in the Vaults 1st field. From the dropdown list, select a SmarTeam vault.
Vaults 2nd	During runtime, the test file is moved from Vaults 1st to Vaults 2nd. From the dropdown list, select a SmarTeam vault.
Log file name	During runtime, the log file records each event that takes place. It is identical to the information displayed in the Trace tab. Note: By default a log file is not created. If you want to create a log file, select On from the Log File menu.
Log file size	Displays the limit of the size of the log file that is created during runtime.

To run the Vault Tester utility:

- 1 From Start, run the **VaultTester.exe** command from the command line or from the **File** menu and select **Start Working**.
The Vault Tester utility automatically creates a test file and runs it according to the parameters previously defined.
- 2 Click the **Trace** tab to view a log of the events as they occur.

Chapter 4: Multi-Vault Environment

SmarTeam – Multi-Vault Replication Setup Checklist

This checklist provides a detailed list of all the steps that need to be performed and the order in which they should be performed to successfully setup SmarTeam – Multi-Vault Replication.

*Requirement: M = Mandatory, O = Optional

	Item	M/O*	Reference
Stage 1: Pre-Setup			
<input type="checkbox"/>	Verify that your Hardware & Software meet the requirements	M	SmarTeam Hardware and Software Requirements Guide
<input type="checkbox"/>	Verify SmarTeam – Foundation is installed and configured properly	M	SmarTeam – Foundation Installation Guide
<input type="checkbox"/>	Install Multi-site Admin (includes the required components and administrative tools to configure the distributed environment on the SmarTeam – Multi-site server)	M	SmarTeam – Multi-site Installation Guide
<input type="checkbox"/>	Assign permissions to the Vault Server user	M	Preparing the Vault Server Environment
<input type="checkbox"/>	Check for any additional prerequisites on the SmarTeam Web Site	M	Release Notes of latest service pack in the release or SmarTeam Support Site
Stage 2: Setup			
<input type="checkbox"/>	Set up the Data Model Designer (DMD) for Vault Replication	M	Vault Replication Mechanism (in DMD)
<input type="checkbox"/>	Set up Vault Sites	M	Setting Up Vault Sites
<input type="checkbox"/>	Configure Vaults for a Site	M	Configuration of Vaults for a Site
<input type="checkbox"/>	Set up Vault Groups Replication	M	Vault Groups Replication Setup
<input type="checkbox"/>	Run SmarTeam Data Model Designer (DMD) for Vault Replication	M	SmarTeam – Editor Online Help
<input type="checkbox"/>	Install and configure RepliWeb or DFS	M	RepliWeb Installation and Configuration Guide OR DFS Installation and Configuration Guide

Configuration of Vaults for a Site

The following steps required to configure vaults for a site are detailed in the following sections:

- 1 [Add Vault Replication Mechanism](#)
- 2 [Setting Up the Vault Server](#)

- 3 [Creating a New Vault Configuration Site](#)
- 4 [Creating a Site in an Existing Vault Configuration](#)
- 5 [Creating a Vault Server](#)
- 6 [Creating a Working Shared Directory](#)
- 7 [Creating Working Vaults](#)
- 8 [Creating a Mirror Shared Directory](#)
- 9 [Creating Mirror Vaults](#)

Vault Replication Mechanism (in DMD)

Add Vault Replication Mechanism

Before configuring Vault Replication on your SmarTeam – Multi-site system, you need to run the SmarTeam – Multi-site Data Model Wizard to add the Vault Replication mechanism to the Primary SmarTeam database, if it is not already installed. For details, see the Adding Replication Mechanisms section in the SmarTeam - Multi-site Administration Guide.

Vault Server Setup

Setting up Vault Replication in a SmarTeam – Multi-Vault system consists of three stages:

- 1 Vault Configuration: For each site, defining the Vault Servers and vaults that participate in the Vault Replication. (See [Setting Up Vault Sites](#).)
- 2 Vault Replication: Defining the logical organization of the vault configuration into operational vault groups corresponding to the Check In, Obsolete and Released lifecycle states. (See [Vault Groups Replication Setup](#).)
- 3 Actual Copying of Files: The actual replication or copying of files from a working to a mirror directory is not supported by SmarTeam software. Rather, the user can choose from several available software options to accomplish this task. For a description about how to install and configure the RepliWeb Deployment Suite® (RDS) software for Vault Replication in the SmarTeam – Multi-site environment, see RepliWeb Installation & Configuration Guide.

Setting Up Vault Sites

Setting Up the Vault Server

To set up the Vault Server:

On the Vault Server machine, launch the Vault Server Setup as follows:

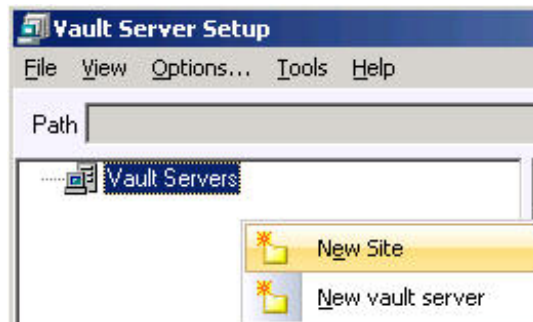
- Start > Programs > [SmarTeam menu] > Admin Console > Vault Server Setup

Creating a New Vault Configuration Site

To create a new vault configuration site:

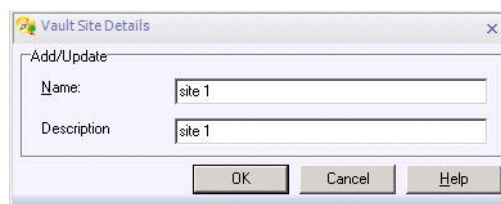
Note: This section applies if you are creating a new vault configuration. If you want to convert an existing vault to participate in a SmarTeam – Multi-site system, see [Creating a Site in an Existing Vault Configuration](#).

- 1 In the Vault Server Setup window, right click **Vault Servers** and select **New Site**.



The Vault Site Details dialog box appears with the following options:

- Vault Site name: Type a name for the vault site
- Site Description: Type description text



Note: The names for sites, vaults, and directory names must be unique for all sites.

- 2 Click **OK**.

The new site appears in the Vault Server Setup window.

- 3 Continue to [Creating a Vault Server](#).

Creating a Site in an Existing Vault Configuration

It is easy to convert an existing isolated vault configuration into one that can participate in a SmarTeam – Multi-site system.

The components of a vault configuration for an isolated site are:

- Vault Server (Node): Computer that contains the vaults
- Shared Directory: Shared directory on Vault Server that contains the working vaults
- Vaults: Subdirectories of the Shared Directory that contains \Check In, \Released, \Obsolete working files

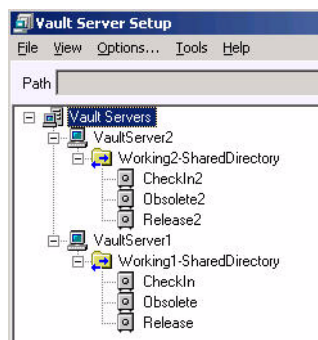
To convert a vault configuration into a configuration supported by SmarTeam – Multi-site:

- 1 See [Creating a site that contains a vault configuration](#).
- 2 See [Linking the Vault Server to the Site](#).
- 3 After performing these steps, see [Creating a Mirror Shared Directory](#).

Creating a site that contains a vault configuration

To create a site that contains a vault configuration:

- 1 From the Vault Server Setup window, which shows an existing vault configuration, right click **Vault Servers** and select **New Site**.



- 2 From the Vault Site Details dialog box complete the following fields:

- Vault Site Name: Type a name for the vault site
- Site Description: Type description text

Note: The names for sites, vaults, and directory names must be unique for all sites.

- 3 In the appropriate field, type the data and click **OK**.

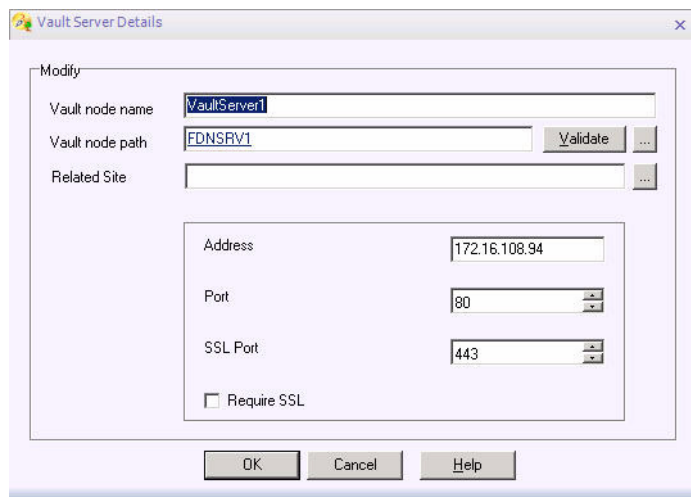
The new site, Site 1, appears on the Vault Server Setup window.

Linking the Vault Server to the Site

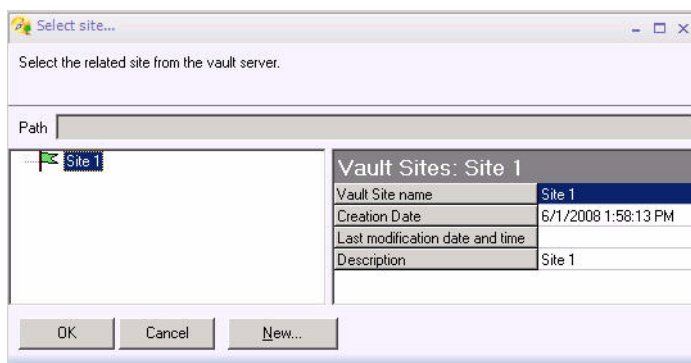
To link the Vault Server to a specific site:

- 1 In the Vault Server Setup window, right click **Vault Server1** and select **Modify** to link Vault Server1 with Site 1.

The Vault Server Details window appears showing the details of the existing Vault Server site.



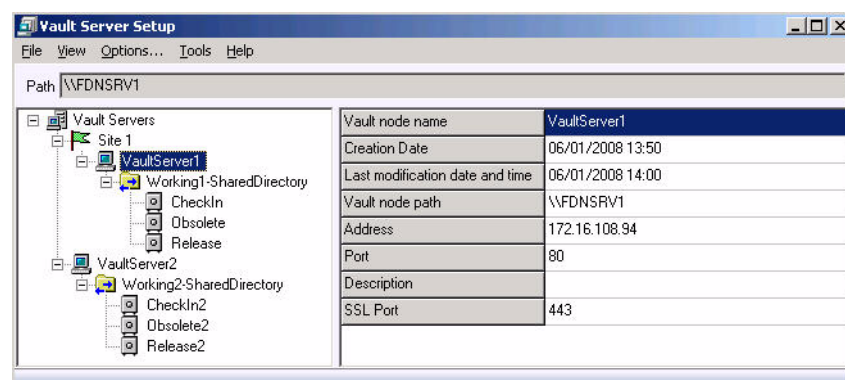
- 2 Click the Browse button for the Related Site field.
- 3 From the Select Site window, click **Site 1** and click **OK**.



Site 1 appears in the Related Site field of the Vault Server Details window.

- 4 Click **OK**.

Site 1 appears on the Vault Server Setup window in the same tree as VaultServer1.



- 5 Repeat this procedure to define **Site 2** and link it with **VaultServer2**.
- 6 Continue to [Creating a Mirror Shared Directory](#).

Alternative Method

The following alternative method also creates a new site in an existing vault configuration:

- 1 In the Vault Server Setup window, right click **VaultServer1** and select **Modify**.
- 2 From the Vault Server Details window, click the Browse button of the Related Site field.
- 3 From the Select Site window, click **New Site**.
- 4 From the Vault Site Details dialog box, complete the fields and click **OK**.
- 5 From the Select Site window, select **New Site** and click **OK**.
- 6 From the Vault Server Setup window, click **OK**.
- 7 Continue to [Creating a Mirror Shared Directory](#).

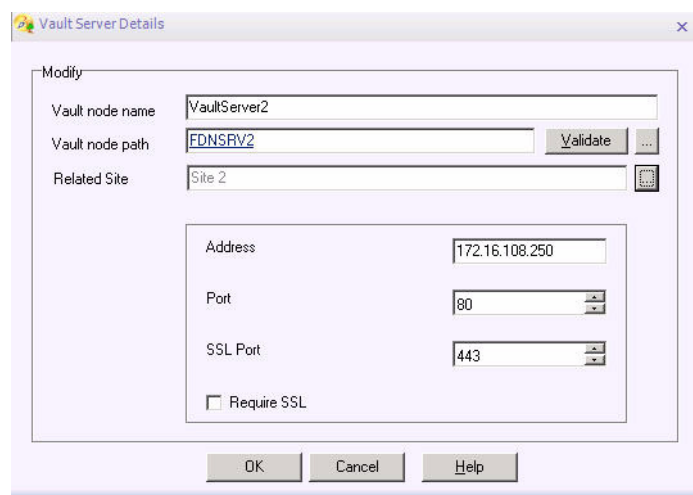
Creating a Vault Server

To create a Vault Server for the site you have created:

- 1 From the Vault Server Setup window, right click on the site you previously created, (see [Creating a site that contains a vault configuration](#)). From the popup menu, select New Vault Server.

The Vault Server Details dialog box appears with the following options:

- Vault Node Name: Type a name for the Vault Server.
 - Vault Node Path: Type the path in the network to the location where the Vault Server is installed. If applicable, click on the browse button to display a standard file selection window.
 - Related Site: The site in which the Vault Server is defined (appears automatically).
 - Vault Server protocol: Select the protocol for client/server communication.
- 2 Click **Validate** to verify the connection.



- 3 Complete the fields and click **OK**.

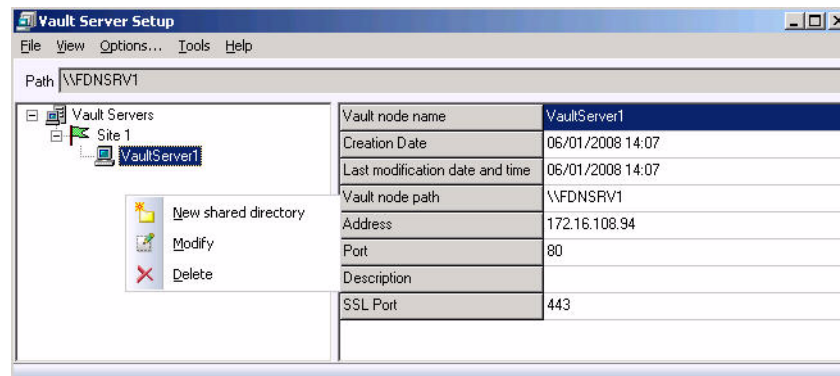
Note: If you receive a message concerning the vault configuration at a remote site, follow the instructions in the message.

The new Vault Server appears in the Vault Server Setup window under the site.

Creating a Working Shared Directory

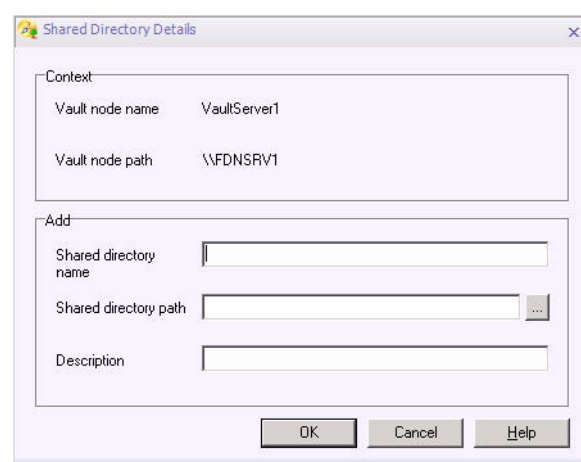
To create a Working Shared Directory in the Vault Server you have created:

- 1 Create an actual directory in the vault node you defined in [Creating a Vault Server](#). Name the directory, for example: <Vault node path>\SharedDir and assign it shared privileges.
- 2 From the Vault Server Setup window, right click on the Vault Server you created and select **New Shared Directory** from the popup menu.



The Shared Directory Details dialog box appears with the following options:

- Vault Node Name: Vault Server name.
- Vault Node Path: Vault Server path.
- Shared Directory Name: Type a name for the shared directory, such as Working1-SharedDirectory.
- Shared Directory Path: In the Vault Node field that you defined in [Step 1](#), type the shared directory name, such as <Vault node path>\SharedDir. If necessary, click the browse button to display a standard file selection window.
- Description: Type descriptive text (optional).



- 3 Complete the fields and click **OK**.

The new Shared Directory appears in the Vault Server Setup window under the Vault Server.

Creating Working Vaults

To create a working vault in the created working Shared Directory:

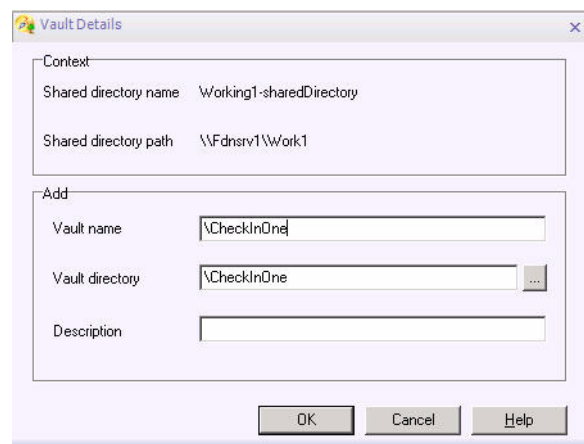
- 1 From the Vault Server Setup window, right click the Shared Directory you previously created (see [Creating a Working Shared Directory](#)), and select **New Vault** from the popup menu.

The Vault Details dialog box appears with the following options:

- Shared Directory Name
- Shared Directory Path
- Vault Name: Vault names in \Check In, \Released, and \Obsolete working files
- Vault Directory: Type the network path for the vault location relative to the Working Shared Directory: one from \Check In, \Released or \Obsolete. If applicable, click **Browse** to display a standard file selection window.

Note: The vault directories do not have to be created ahead of time. You can create them through the Vault Server Setup window.

- Description: Type descriptive text (optional).



- 2 Complete the fields and click **OK**.
 - 3 Repeat this procedure for all three working vaults: Check In, Released, and Obsolete.
- The new Vaults appear in the Vault Server Setup window under the Shared Directory.

Creating a Mirror Shared Directory

To create a Mirror Shared Directory in the Vault Server you have created:

- Repeat the steps in [Creating Working Vaults](#), except for the following fields:
 - Shared Directory Name: Type a name for the shared directory, such as Mirror1-SharedDirectory
 - Shared Directory Path: Type the directory name in the Vault Server that serves as a mirror shared directory, such as <Vault node path>\Mirror_SharedDir.

If necessary, click the browse button to display a standard file selection window.

Creating Mirror Vaults

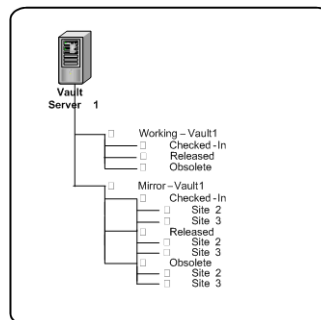
To create mirror vaults in the created Mirror Shared Directory:

- 1 Repeat the steps in [Creating Working Vaults](#), except for the following fields:
 - Vault Name: Example: Mirror1_Checkin, Mirror1_Released, and Mirror1_Obsolete
 - Vault Directory: Type the network path for the vault location relative to the Mirror Shared Directory, such as \Check In, \Released, or \Obsolete.

If applicable, click the browse button to display a standard file selection window.

Note: The vault directory for the Mirror vaults must have the same names as the Working vaults. It is necessary to add the <MySite> subdirectory under the vaults in the mirror shared directories because the recursive replication defined in this document does not automatically create the required subdirectories under the mirror vault.

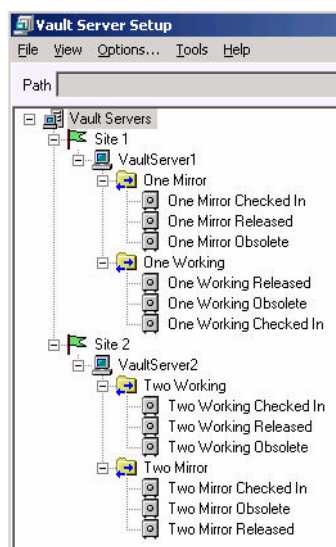
- 2 Repeat for all of the following mirror vaults as required: Check In, Released, and Obsolete. The following image shows the required hierarchy for Mirror – Vault1.



Vault Configuration Example

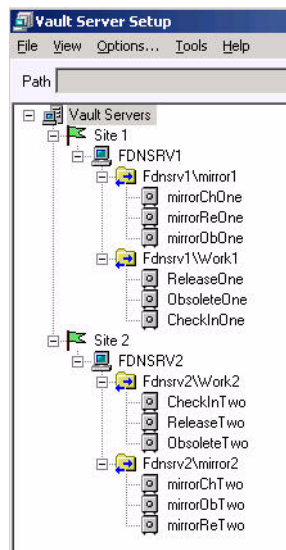
Vault Configuration Logical View

The following figure shows a typical example of a vault configuration for two sites, where the logical names of the entities are displayed:



Vault Configuration Physical View

The following figure shows a typical example of a vault configuration for two sites, where the physical names of the entities are displayed:



Vault Groups Replication Setup

In this section, configure the operational groups of vaults.

The general steps are:

- 1 Create a Group for each Lifecycle state
- 2 Designate the vaults for each Group

Setting Up the Vault Server

To set up a Vault Server:

- 1 On the Vault Server machine, launch the Vault Server Setup as follows:
 - Start > Programs > [SmarTeam menu] > Admin Console > Vault Server Setup.

OR

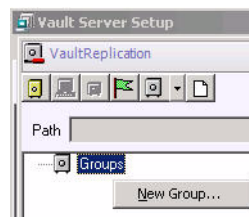
- Go to <SmarTeam Home>\Bin directory and double click on VaultSetup.exe
- 2 From the Vault Server Setup window, select Tools > Vault Replication.

The Vault Replication dialog box appears.

Creating a Group

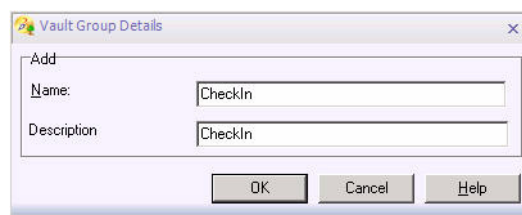
To create a new group:

- 1 From the Vault Replication window, right click **Groups** and select **New Group**.



The Vault Group Details dialog box appears showing the following options:

- Vault Group Name: Type the vault group name, such as Check In, Released or Obsolete.
- Vault Group Description: Type a text description.



- 2 Complete the Vault Group details and click **OK**.

The new vault group, Check In, appears in the Vault Replication window.

- 3 Repeat the same procedure for all Vault Groups: Check In, Released and Obsolete.

Creating a Vault Set

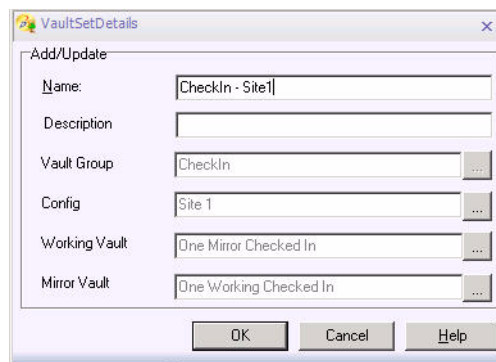
A Vault Set is the set of vaults in a site that are associated with a specific operation. For example, at Site1, all the working and mirror vaults associated with Check In form a Vault Set. A good name for this Vault Set is Check In-Site1, which shows the operation and the site for the Vault Set.

To associate the Vault Sets with the group of that operation:

- 1 In the left area of the Vault Replication window, select a Group for which the Vault Set is to be created and right click.

The Vault Set Details dialog box appears, with the following options:

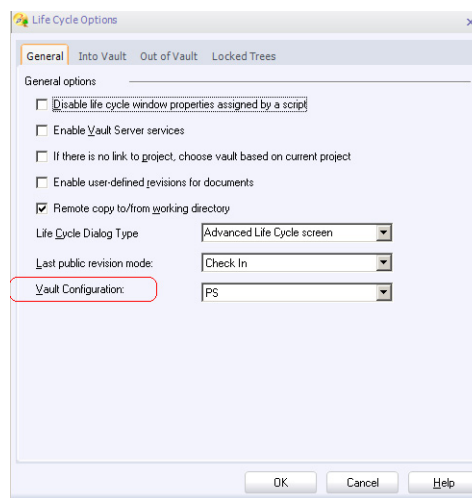
- VaultSet Name: Type a VaultSet name, such as Check In, Released or Obsolete.
- Vault Set Description: Type text description (optional).
- Vault Group: Vault group to which this Vault Set is associated (automatically completed).
- Config: Site to which the Vault Set belongs.
- Working Vault: Name of the working vault at this site for this operation. If applicable, click on the browse button to display a standard file selection window.
- Mirror Vault Name: Name of the mirror vault at this site for this operation. If applicable, click on the browse button to display a standard file selection window.



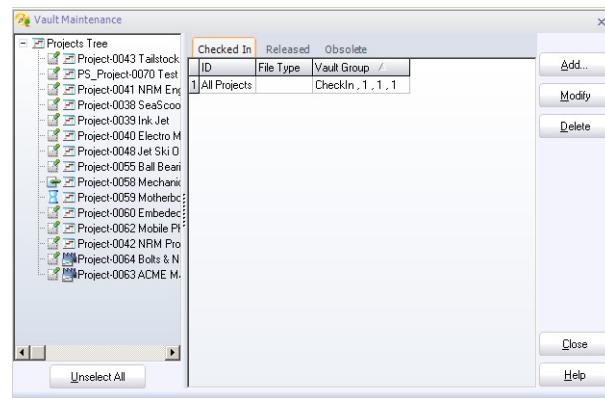
- 2 Complete the fields and click **OK**.
- 3 Repeat for all Vault Sets in all sites for this group.
- 4 Repeat for all Groups, such as Check In, Released and Obsolete.

Post Configuration Tasks

- 1 After completing the Vault Configuration, the user needs to specify the Vault Configuration to be used by each site. Run SmarTeam – Editor on each site and connect to the site database. Go to Tools > Administrator Options > Lifecycle Options. From the General tab, select the Vault Configuration for that site.



- 2 On the Primary Site in SmarTeam – Editor go to Tools > Vault Maintenance to define CheckedIn, Released and Obsolete Vault Groups for projects. This will then be replicated to the other sites.



Vault Replication Report

Providing Mapping Information to Replication Software

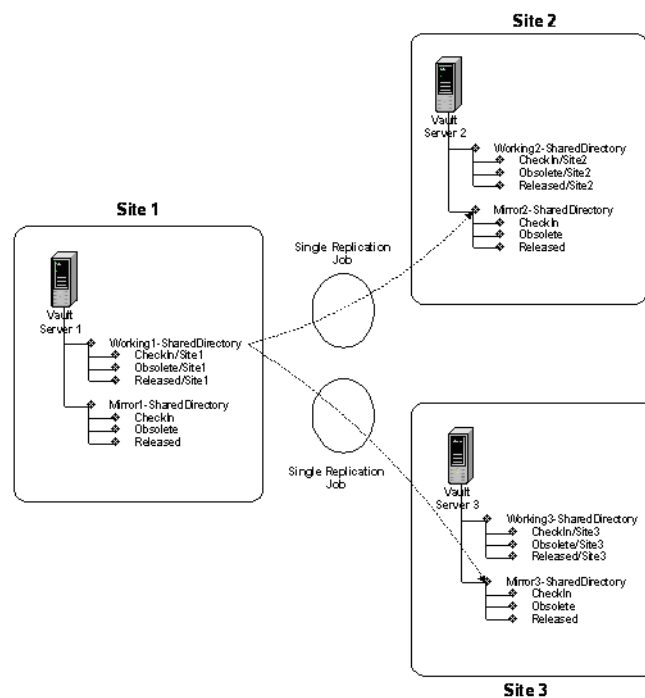
The Replication software used to support Vault Replication for a specific SmarTeam – Multi-site system, such as RDS, has to be provided with instructions to specify the source and destination directories in accordance with the vault configuration for that system.

Generate these instructions automatically by pressing the report button on the Vault Replication window after the Vault Replication procedure is performed.

For the vault configuration, the report has the following format:

```
[Site 1]
"\\Fdnsrv1\mirror1\mirrorchone" "\\Fdnsrv2\mirror2\mirrorchtwo\Site 1"
"\\Fdnsrv1\mirror1\mirrorreone" "\\Fdnsrv2\mirror2\mirrorretwo\Site 1"
"\\Fdnsrv1\work1\obsoleteone" "\\Fdnsrv2\mirror2\mirrorobtwo\Site 1"
[Site 2]
"\\Fdnsrv2\work2\checkintwo" "\\Fdnsrv1\work1\checkinone\Site 2"
"\\Fdnsrv2\work2\releasetwo" "\\Fdnsrv1\work1\releaseone\Site 2"
"\\Fdnsrv2\work2\obsoleteTwo" "\\Fdnsrv1\mirror1\mirrorobone\Site 2"
```

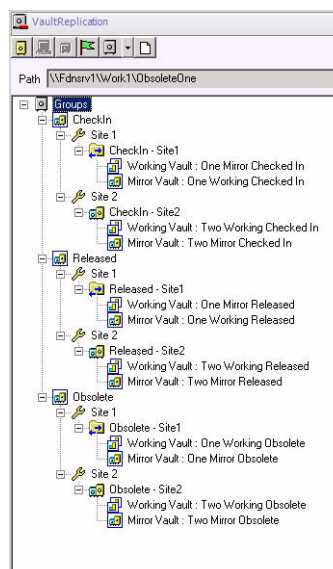
The following figure shows the subdirectories graphically and also indicates the file replication activity for the Check In operational group.



Vault Replication Example

Vault Replication Logical View

The following figure shows a typical example of a Vault Replication for two sites, where the logical names of the entities appear:



Replication Operations

The actual replication or copying of files from working to mirror directory is not supported by SmarTeam software. Rather, the user can choose from several available software options to accomplish this task. For a description about how to install and configure the RepliWeb Deployment

Suite® (RDS) software for Vault Replication in the SmarTeam – Multi-site environment, see the RepliWeb Installation & Configuration Guide. This document contains useful information regarding any software you use for replication operations.

When setting up a Replication Job for each working/mirror directory pair, you must differentiate between RedLine file and all other files (non-Redline files). In addition, you must define two types of jobs:

- [Non-RedLine Files](#)
- [RedLine Files](#)

Non-RedLine Files

Non-Redline files are replicated from working to mirror directories as an exact copy, so that at the end of the replication, the mirror directory is an exact copy of the working directory. In RDS this is saved as **Mirror source to target** under the Logic tab.

RedLine Files

RedLine files need to be treated separately because a RedLine file can be updated by SmarTeam directly in a mirror directory. Therefore it is possible that the latest version of the RedLine file is in the mirror directory but not in the source directory, or in mirror directories at other sites. When this occurs, the latest version of the RedLine file must be copied to the source directory and, from there, replicated to the mirror directories at other sites.

To handle RedLine files, do the following:

- When the RedLine file is saved to the local directory, working or mirror, a script is activated by the SmarTeam script hook, **After - On Saving RedLine**. This script is not provided by SmarTeam, but it can be easily written.
- The script performs two tasks in the following sequence:
 - a If the RedLine file was saved to the local mirror directory, the script copies the RedLine file from local mirror directory to the remote working directory corresponding to the local mirror directory.
 - b Whether the RedLine file is saved to a local, working or mirror directory, the script creates a trigger file in the local Vault Server shared directory that initiates the RDS replication of all RedLine files. This replication transfers the new RedLine file from the source working directory to the mirror directories at other sites.

For [Step b](#), you must define a separate RedLine Replication Job for each working/mirror directory pair in the Vault Replication. To ensure that later versions are not overwritten in the mirror directory, use logic similar to the RDS **Backup source** logic.

Deleting Files During Vault Replication

Select **Delete** from the SmarTeam – Editor menu to delete all relevant files during Vault Replication. A file can also be deleted from a vault directory when moved to the Release or Obsolete vault during a Release and/or Obsolete lifecycle operation.

This does not interfere with SmarTeam – Editor's normal operation. In addition, there is no possibility of inadvertently creating an identical file, since the files are given names using the object's unique ID.

Cleaning Up Mirror Vaults

Since RepliWeb® is used in backup mode, files are only added to mirror directories and files that are deleted at a local file are not deleted at the mirror vault, so that files accumulate at the mirror vault.

The mirror vaults can be periodically purged by a dedicated RepliWeb® job.

Chapter 5: Vaults Security

To ensure the vaults are configured securely the following setup requirements are addressed:

- Vault keys must be generated and imported (see [To configure a default environment:: Step 6 - Step 15](#))
- [HTTP and HTTPS](#)
- [SSL](#)

HTTP and HTTPS

Vaults can work with either http and https, which is more secure.

Two ports are configured during setup. (See [To configure vaults manually:: Step 7.](#))

- 80 is the default port number for http
- 443 is the default port for https

These ports are preferable because they are standard ports for http and https. If a user wants to use different ports, they can be changed in this dialog box. (See [To configure vaults manually:: Step 7.](#))

If the **Require SSL** checkbox is enabled, any communication with the Vault Server must be via https only, or an error occurs.

SSL

To configure a Vault Server with https / SSL:

- 1** A user buys or creates an SSL certificate and registers the certificate on a port used for SSL by using httpcfg.exe (Microsoft's utility).
This standard procedure is explained by Microsoft® in the following articles:
 - <http://msdn.microsoft.com/msdnmag/issues/06/08/securitybriefs/default.aspx>
 - <http://msdn2.microsoft.com/en-us/library/ms733768.aspx>
- 2** Import Certificate into vault machine.
 - a** From Start > Run, execute the MMC command.
 - b** From the File menu, click **Add/Remove Snap-in**.
 - c** Select the Snap-in you want to add from the available list.
 - d** Expand Certificates > Personal folder. Right click on All Tasks > Import.
 - e** Browse to <certificates.pfx> file and then click **OK**.

- f** Select File > Open. You may be prompted to type a password.
- 3** Open httpcfg.exe.
For information about the parameters of httpcfg.exe, see the Microsoft articles:
 - <http://msdn.microsoft.com/msdnmag/issues/06/08/securitybriefs/default.aspx>
 - <http://msdn2.microsoft.com/en-us/library/ms733768.aspx>
- 4** To enable clients to communicate with a vault over https, an administrator must change **smarteam.std.vaultClient.config.xml** configuration setting from Default to Domain on the Domain level of the Foundation machine by doing the following:
 - a** From the System Configuration Editor, open the smarteam.std.vaultClient.config.xml.
 - b** Select **Edit** and replace all the occurrences of following http attribute values:
`<SecurityMode>None</SecurityMode>`
with the following https attribute values:
`<SecurityMode>Transport</SecurityMode>`
 - c** Save the file.
- 5** Restart the Vault Server.
- 6** Run IISRESET on SmarTeam – Web Editor.
- 7** Reopen SmarTeam – Editor to ensure all the cache is reloaded.

Note: The SecurityMode attribute values are changed from http to https for all clients using this Foundation machine because the configuration is done on the Domain level.

For administrators that need more information, see

<http://msdn2.microsoft.com/en-us/library/system.servicemodel.basichttpsecuritymode.aspx>.

Chapter 6: SmarTeam – Workflow

After the SmarTeam – Foundation software has been installed you must perform post-installation tasks as described in this chapter.

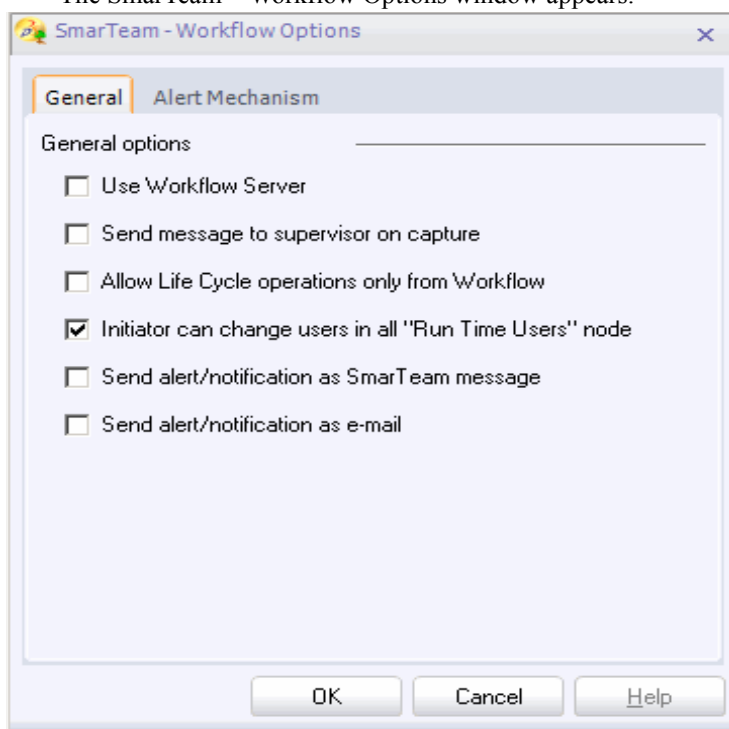
SmarTeam – Workflow

If you selected Workflow components in the select components window, you have installed SmarTeam – Workflow. You must now configure your computer to work with this product successfully.

DEMO INSTALLATION - Configure the Workflow server, through the Workflow server setup utility.

To define SmarTeam – Workflow options:

- 1 From the SmarTeam – Editor main menu, select **Tools > Administrator Options...** to display the Administrator Options window.
- 2 In the Workflow section, click **SmarTeam – Workflow Options**.
The SmarTeam – Workflow Options window appears.



The following options are available from the SmarTeam – Workflow Options window.

■ Use SmarTeam – Workflow Server

When selected, the SmarTeam – Workflow Server appears.

■ Send message to supervisor on capture

When selected, a message is generated and sent to a supervisor during a capture operation.

■ Allow Lifecycle operations only from SmarTeam – Workflow

When selected, Lifecycle operations are only available from within SmarTeam – Workflow.

When cleared, Lifecycle operations are available from both SmarTeam – Workflow and SmarTeam – Editor.

■ Initiator can change users in all "Run Time Users" node

This option determines whether the initiator can set users on all **Run time user nodes** when the user initiates a process. Select this option if the current node is directly connected to the node for setting users. This option enables the initiator to set users also in nodes that are not directly connected to the Start node.

■ Send alert/notification as SmarTeam message

When selected, workflow alerts and notifications are as a SmarTeam message that appears in your SmarTeam user inbox.

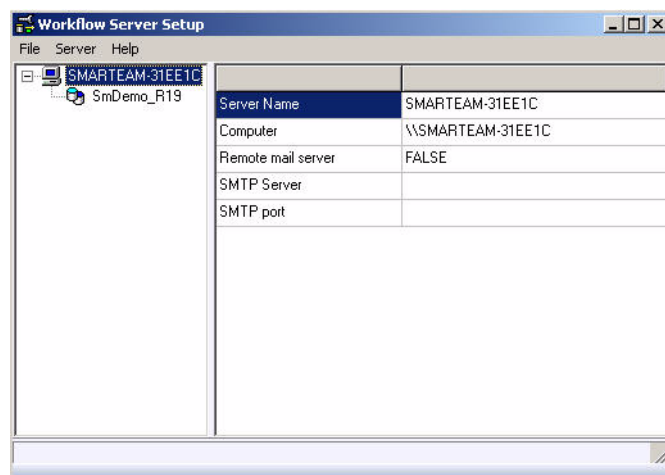
■ Send alert/notification as email

When selected, workflow alerts and notifications are sent as an email that appears in your user mailbox.

- 3** In the SmarTeam – Workflow Options window, select the applicable checkbox for the option you want to select.
- 4** Click **OK** to save your changes and close the SmarTeam – Workflow Options window.
- 5** In the Administrator Options window, click **Close** to exit.

Setting Up the SmarTeam – Workflow Server

The Flow Server utilities enable you to set up and run the Flow Server for your network. After the Flow Server utilities are installed, the Flow Service is automatically installed on the computer designated as the server, and it runs transparently in the background. From the Control Panel, you can open the standard Services window, which lists the services currently installed. After installation, the Flow Service is displayed among the other services.



Note: Keep in mind that special permissions must be defined for the Flow Service, after the installation process (using standard Windows procedures).

The Flow Server controls the flow of processes between users in the SmarTeam – Workflow environment. The Flow Server constantly monitors the database at predefined time intervals and checks whether any process is ready to be moved to the next user (node). The administrator can determine the following parameters:

- The databases that the Flow Server monitors.
- The time intervals that the Flow Server checks a specific database.
- The number of work threads used by the Flow Server for a specific database; the more work threads assigned, the faster the processing speed.
- The services upon which the Flow Service is dependent. These services run transparently in the background in conjunction with the Flow Service.

Preparing the Flow Server Environment

The following utilities are needed to install and maintain the Flow Server. Some of these files work automatically in the background; others must be run so that you can define specific parameters. These utilities (.exe files) are installed in the SmarTeam/Bin directory by default.

Utility	Purpose
SetupWorkflow Service.exe	Runs the Service Setup utility, which enables you to define the following parameters: <ul style="list-style-type: none"> • Services that are dependent on the Flow Service • Username and password for the Service
FlowServerSetup.exe	Runs the Server Setup utility, which enables you to define: <ul style="list-style-type: none"> • Databases monitored by the Flow Server • Time intervals (audit time) during which the Flow Server monitors a specific database • Number of work threads used by the Flow Server to process the information
WorkflowCapsule.exe	The operating system runs this utility when the Service is started.
WorkflowServer.exe	This utility is invoked by the WorkflowCapsule utility. It implements the audit process and connects the threads to the database. This utility actually implements the Flow Server operation.

To ensure a smooth installation of the Flow Server utilities, check that a user account is defined, which serves as the Flow Service user.

Installing the Flow Service

The Flow Service must be installed on the same computer that is designated as the Flow Server. When you install the Flow Service, you can determine the following:

- Services upon which the Flow Service is dependent, such as those services that run in conjunction with the Flow Service
- Username and password for the Service

To Install the Flow Service:

- 1 Double-click on the SetupWorkflowService.exe.
 - Available list box lists the services that are currently installed on your server
 - Selected list box lists the services that must be invoked before the Flow Service
 - Username and Password refer to the Flow Service user
- 2 Select the services that enable access to the databases that the Flow Server monitors and move these services into the Selected list box.
- 3 In the Account Name field, type the user account for the Flow Service. Verify the domain name is included when necessary. In the Password field, type the password of the user.

Use the following syntax: <domain>\<accountname> for the account name. If the account is a local account, use the following syntax: \\<computer>\<accountname>.

Note: The assigned user must be a member of the Flow Administrators group.

- 4** Click **Install** to install the Flow Service.

Chapter 7: Full Text Search

This option enables you to perform complex textual searches on data without knowing where that data is stored. Full Text Search enables you to perform searches on files within the vaults and on textual metadata stored in the database. Complex textual searches include Boolean expressions, such as **AND**, **OR**, and also the use of phonetic phrases such as **sounds like**.

SmarTeam Full Text Search Environment

SmarTeam Full Text Search uses the following technologies:

- **Microsoft Indexing Service:** Supports Windows Server 2003 and above and includes Microsoft Index Service, which provides Full Text Search and indexing of files. The service provides support for many popular file formats and can be extended to support additional formats. Extensions for useful formats, such as Adobe Acrobat and AutoCAD drawings.

For more information about the Microsoft Indexing Service, see:

<http://msdn2.microsoft.com/en-us/library/ms689646.aspx>

- **Full Text Search:** Performs Full Text Search queries on data stored in the database.

The SmarTeam Full Text Search capability uses the Microsoft Full Text Search API. There might be different result sets when using the SmarTeam Full Text Search or the Microsoft® Full Text Search.

For example: When a user types TEXT*, SmarTeam translates it to: {prop name=All}TEXT*{/prop}

The detailed technical explanation can be found in the Microsoft technical notes, at:

[http://msdn.microsoft.com/en-us/library/ms690492\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/ms690492(VS.85).aspx)

Note: When you perform a Full Text Search operation, you can search for exact phrases or use wild cards (e.g., *, ?) in the search.

SmarTeam Full Text Search Keys

These keys are available for configuring the Full Text Search feature:

- **SmartFTSMaxResultCount:** Specifies the maximum number of objects that can be viewed in the result list. If the number of objects to be displayed is greater than this value, only the first <SmartFTSMaxResultCount> objects are displayed. SmarTeam recommends that this number not exceed 100. If a search returns a larger number of results, the search should be refined.

- **SmartFTSDefaultOperator**: Specifies the operand to be used between two words in a Search operation that have no Boolean operand between them (**and** / **or**). For example, if this key is set to **and**, the **Cat Dog Mouse** string is read as **Cat and Dog and Mouse**. Possible values: **and**, **or**.

SmarTeam Full Text Search Components

SmarTeam Full Text Search components consist of:

- SmarTeam Full Text Search Server
- SmarTeam Full Text Search Client

SmarTeam Full Text Search Server

SmarTeam Full Text Search server is a web service that provides a front end for the file indexing service. Currently it is implemented on top of Microsoft's Indexing Service.

SmarTeam Full Text Search Server searches using the indexing service in the files that reside in the SmarTeam vault. Results are returned to the SmarTeam Full Text Search Client.

SmarTeam Full Text Search Client

SmarTeam Full Text Search Client provides the ability to perform the Full Text Search queries. When a query is issued it queries the files in the SmarTeam vault using the SmarTeam Full Text Search Server and submits a query to the underlying database to search the stored data.

Recommended Configurations

SmarTeam provides the following installation configurations for installing FTS within the SmarTeam environment:

- FTS Configured with Three Computers (Recommended)
- FTS Configured with Four Computers
- FTS Configured with Three Computers
- FTS Configured with Two Computers
- FTS Configured with a Single Computer

FTS Configured with Three Computers (Recommended)

This configuration is recommended for optimal results.

Note: In cases of low system performance, it is recommended to install the Full Text Search Server and the Index Server on separate machines.

Computer With	Software	Operating System (OS)
Database Server	Oracle / SQL	Windows Server 2003 and above
Vault Server	SmarTeam Vault Server	Windows Server 2003 and above
Full Text Search Server, Index Server	Full Text Search Server, IIS 6.0 and above, Microsoft Indexing Service	Windows Server 2003 and above

FTS Configured with Four Computers

Computer with	Software	Operating System (OS)
Database Server	Oracle / SQL	Windows Server 2003 and above
Vault Server	SmarTeam Vault Server	Windows Server 2003 and above
Full Text Search Server	Full Text Search Server, IIS 6.0 and above	Windows Server 2003 and above
Index Server	Microsoft Indexing Service	Windows Server 2003 and above

FTS Configured with Three Computers

Computer with	Software	Operating System (OS)
Database Server	Oracle / SQL	Windows Server 2003 and above
Vault Server, Full Text Search Server	SmarTeam Vault Server, Full Text Search Server	Windows Server 2003 and above
Index Server	Microsoft Indexing Service	Windows Server 2003 and above

FTS Configured with Two Computers

Computer with	Software	Operating System (OS)
Database Server	Oracle / SQL	Windows Server 2003 and above
Vault Server, Full Text Search Server, Index Server	SmarTeam Vault Server, Full Text Search Server, Microsoft Indexing Service	Windows Server 2003 and above

FTS Configured with a Single Computer

Computer with	Software	Operating System (OS)
Database Server, Vault Server, Full Text Search Server, Index Server	Oracle / SQL, SmarTeam Vault Server, Full Text Search Server, Microsoft Indexing Service	Windows Server 2003 and above

Microsoft Indexing Service File Formats Support

The Microsoft Indexing Service support, by default, the indexing of the following file formats:

- All Office Files (Word, Excel, PowerPoint)
- Text Files
- HTML Files

Microsoft Indexing Service can be extended to support more file formats by using 3rd party components. These 3rd party components implement an Indexing Service extension interface called IFilter.

Note: IFilters are available as shareware or may require a license from third-party vendors. ENOVIA SmarTeam assumes absolutely no liability whatsoever regarding use of these third-party applications.

For more information on the IFilter interface, see the Microsoft Web site:

[http://msdn2.microsoft.com/en-us/library/ms691105\(VS.85\).aspx](http://msdn2.microsoft.com/en-us/library/ms691105(VS.85).aspx)

Useful IFilters

■ Adobe Acrobat – IFilter

A PDF IFilter enables the indexing of PDF files.

This IFilter can be downloaded from the Adobe Web site at www.adobe.com.

After accessing the site, click on the Search link and perform a search for IFilter to locate the file to download.

■ CAD & Company – IFilter

A DWG IFilter enables a text search of DWG drawings.

This IFilter can be downloaded from the following sites:

<http://www.cad-company.nl/ifilter/>

<http://www.autodesk.com/partnerproducts>

After accessing the site, click on the Search Engine link and perform a search for DWG IFilter to locate the file to download.

IMPORTANT! After installing an IFilter, refer to Rescanning FTS Directories After Adding an IFilter for instructions on how to rescan indexed directories.

Installing Full Text Search

This section contains the following installation instructions:

- Full Text Search Server – Post Installation
- Full Text Search Client – Post Installation
- Indexing Other File Types in the Microsoft Indexing Service
- Microsoft Indexing Service Performance Tuning

Full Text Search Server – Post Installation

After completing the Smart Full Text Search Server software installation process, you must perform the following post installation tasks:

- Defining a Full Text Search Catalog
- Updating the Database for Full Text Searching
 - Oracle
 - Microsoft SQL Server

Defining a Full Text Search Catalog

To define a Full Text Search Catalog:

- 1 From the Start button, select **Settings > Control Panel > Administrative Tools > Computer Management**.
The Computer Management window appears.
- 2 In the *Computer Management* window, select and select **Services and Applications**. Select **Indexing Service**.
- 3 From the menu, select **Action > New > Catalog**.
The Add Catalog window appears.

- 4** In the Add Catalog window, complete the following fields:

Name	Type SmartFTS for the Catalog name.
Location	Type the location path or click Browse to locate a directory for saving the SmartFTS Catalog via the Browse for Folder window. This folder holds the database of indexed data. Note: Prior to selecting a folder for the catalog, create a directory for it. The newly-created SmartFTS Catalog includes all indexed data retrieved from searched directories.

- 5** Click **OK** to save the new SmartFTS Catalog in the selected location.
SmartFTS Catalog name appears in the Computer Management Tree.
- 6** In the Computer Management Tree, select **SmartFTS Catalog** to view its contents in the right pane.
- 7** Right click **Directories** to view a popup menu. From the popup menu, select **New > Directory**. The Add Directory window appears.
- 8** In the Add Directory window, complete the following fields:

Path	Type the location path to the SmarTeam vault directory or click Browse to locate the vault directory via the Browse for Folder window (Alias (UNC)). When the SmarTeam vault directory is located on a different computer, type the full location path.
Account Information	The Username and Password fields are only required when accessing a UNC folder that requires specific access.
Include in Index	Yes: Includes this directory for a text search via the Indexing Service. No: This directory is not included for a text search via the Indexing Service.

- 9** In the Add Directory window, click **OK** to save your entries and add the directory to the SmartFTS Indexing Service.
- 10** In the Computer Management window, click **Close** to exit.

Note: The user running the Microsoft Indexing Service must be a member of the Vault Administrators Group (SmVaultAdmins). By default the user running the Microsoft Indexing Service is **SYSTEM**.

- 11** Proceed to [Updating the Database for Full Text Searching](#).

Updating the Database for Full Text Searching

To update the SmarTeam Database to enable Full Text Search, the SmarTeam Database Administrator needs to use the Full Text Search SQL Script Generator, which helps prepare all necessary SQL statements to enable Full Text Search of the database.

To update the SmarTeam Database to function with Smart Full Text Search software:

- 1 From **Start**, select **Programs > SmarTeam > Administrative Tools > Full Text Search SQL Script Generator**.
The Available Databases window appears.
- 2 In the Available Databases window, select the required database then click **OK** to continue.
- 3 In the SmarTeam User Login [<Database Name>] window, type your username and password then click **OK** to continue.
The SmarTeam Full Text Search SQL Script Generator Welcome window appears.
- 4 After reading the information in the Welcome window, click **Next** to continue to the next Choosing Fields for Full Text Indexing window.
- 5 In the Choosing Fields for Full Text Indexing window, select the Class for adding Full Text Search capabilities.
 - In the left pane, select a Class to view available attributes in the right pane.
 - Attributes displayed are only of Char and Memo type, because these are the only attributes types that can be full text indexed.
 - Attributes displayed as disabled and marked with the [Inherited] tag are attributes inherited from a parent class or super class.
 - In the left pane, a green checkmark is displayed next to a Class name when attributes other than default attributes are selected.
 - In the right pane, click **Select All** to select all attributes displayed for a selected Class.
 - In the right pane, click **Clear** to deselect all selected attributes other than the default attributes.
 - After selecting additional attributes, click **Next** to generate the script and proceed to the next Generated Script window.
- 6 The Generated Script window displays a script generated according to your selections in the previous window. An SQL script is generated according to the database type (SQL Server or Oracle) for the DBA to save and run on the database server.
 - Click **Copy** to save the script to the clipboard for inserting in a specific file.
 - Click **Save** to view a standard Windows Save As window and save the script in SQL Script (*.sql) format.
- 7 After saving or copying the file, click **Next** to proceed to the next window.
- 8 In the Generated Script Complete window, click **Finish** to exit the utility.
- 9 Run the SQL script file to create an index with the SmarTeam database. (The SQL scripts should be run in SQL Plus in Oracle and in Query Analyzer in SQL Server by the owner of the database.) Add the type of user, such as owner of the database that runs the script.
- 10 The FTS script generator provides the script 'as is'. If there are locks in the database that prevent running the script, you must fix the script locally. If it is necessary to delete the indexes, launch SmartDBRepairing.exe and run **Repair Indexes**.

Prior to running the generated script on an SQL Server database, see [Updating an SQL Server Database for Full Text Searching](#).

Updating the Full Text Search indexes on the database is a costly operation and cannot occur in real-time, since it can cause serious decreases to performance. It is recommended that you create a scheduled job to update the indexes every 5-10 min.

Updating an Oracle Database for Full Text Searching

After running the generated SQL script, run the following script on the database to create a scheduled job to update the Full Text Search indexes every 5-10 minutes. This job is required to ensure the Full Text Search indexing is up-to-date with the data in the database.

To run the script:

- 1 Type SQL Plus as a CTXSYS Oracle user.
- 2 Run the `ctx_schedule.sql` script from <SmarTeam Home Directory>\Utilities\SmartFTS\Server\Utils\SQL\Oracle.

This script creates a job for updating the Full Text Search Indexes as well as a stored procedure for starting and stopping the job.

- 3 Type SQL Plus as SmarTeam user.
- 4 Run the `gen_fts_schedule_start.sql` script from <SmarTeam Home Directory>\Utilities\SmartFTS\Server\Utils\SQL\Oracle\.

This script starts the job. To stop the job run the `gen_fts_schedule_stop.sql` script.

Add the type of user that runs the script: owner of the database.

Updating an SQL Server Database for Full Text Searching

Before running the generated script, you must create a full text catalog on the database.

To create a full text catalog:

- 1 Type the SQL Server Enterprise Manager.
- 2 Select the desired server and database.
- 3 Right click on **Full Text Catalogs** and select **New Full-Text Catalog**.
- 4 Type **Cat_SmarTeam**.
- 5 Click on the **Schedules** tab and click **New Catalog Schedule**.
- 6 Assign a name to the schedule.
- 7 In **Job Type**, select **Incremental Population**.
- 8 In **Schedule Frequency**, select **Recurring** and click on **Change**.
- 9 From **Occurs**, select **Daily**.
- 10 From **Daily Frequency**, select **Occurs Every 10 minutes**.
- 11 Select a **Starting at** time from the scroll box.
- 12 Select a **Ending at** time from the scroll box.
- 13 Click **OK** to close the Change frequency window.
- 14 Click **OK** to create the catalog.
- 15 Now run the generated script using the Query Analyzer.

Indexing Other File Types in Microsoft Indexing Service

To index file types in Microsoft Indexing Service:

- 1 From the **Start** button, select **Settings > Control Panel > Administrative Tools > Computer Management** to view the Computer Management window.
- 2 In the Computer Management window, select and select **Services and Applications**.
- 3 Select **Indexing Service** and then select **SmartFTS**.
- 4 Right click **SmartFTS** and from the popup menu, select **Properties**.
The SmartFTS Properties window appears.
 - In the SmartFTS Properties window, click on the **Generation** tab to view the **Generation** page.
 - **Inherit above settings from Service**: Deselect this option.
 - **Index files with unknown extensions**: Select this option.
 - Click **OK** to save your changes and return to the Computer Management window.
- 5 In the Computer Management window, click **Close** to exit.

Microsoft Indexing Service Performance Tuning

System performance can be optimized for best performance using the following Full Text Search service:

- 1 From **Start**, select **Settings > Control Panel > Administrative Tools > Computer Management**.
The Computer Management window appears.
- 2 In the Computer Management window, select and select **Services and Applications**.
 - Right click on **Indexing Service** and from the popup menu, select **Stop** to stop the Indexing Service.
 - From the popup menu select **All Tasks > Tune Performance**.
The Indexing Service Usage window appears.
In the Indexing Search Usage window, select one of the following options to optimize system performance:
 - Used often**
Specifies that the Indexing Service is used often enough on this computer to require better than average performance.
 - Used occasionally**
Specifies that the Indexing Service is *not* used often enough on this computer to require better than average performance.
 - Never used**
Specifies that the Indexing Service is never used on this computer and that it will be turned off.
 - Customize**
Select this option and then click **Customize...** to set your own personalized performance tuning settings.

- 3 Right click on **Indexing Service** and from the popup menu select **Start** to restart the Indexing Service.

Post Installation – Client

After completing the Smart Full Text Search Client software installation process, proceed to [Attaching the Full Text Search Script](#).

Attaching the Full Text Search Script

To attach the Full Text Search Script:

- 1 From **Start**, select **Programs > SmarTeam > Administrative Tools > Admin Console > SmartBasic Script Maintenance** to launch the Script Maintenance utility.
- 2 In the SmarTeam User Login [<Database Name>] window, type your username and password and then click **OK** to continue.
The Script Maintenance window appears.
- 3 In the right pane of the Script Maintenance window, click on the **User Defined** tab to view the **User Defined** window.

Operation

Click in an empty cell and add the **Full Text Search** operation name.

Function Name

Double click in the cell corresponding to the new **Full Text Search** operation to view the Script Browser window.

In the Script Browser window:

File Name

Double click on **SmartFTS-1.0.BS** to view the script in the **Functions** pane and **Source** pane.

Click **OK** to exit the Script Browser window and save the script in **Script Maintenance > User Defined** tab window.

- 4 In the Script Maintenance window, select **File > Exit** to save your entries and exit the Script Maintenance utility.

For further information about working with the Script Maintenance utility, see the SmarTeam – Editor Administration Guide.

- 5 Proceed to [Updating the Menu Editor](#).

Updating the Menu Editor

After successfully adding the scripts in the previous section, you now need to add the **Full Text Search** menu and icon to the SmarTeam menu and toolbar via the SmarTeam Menu Editor utility.

The Menu Editor utility enables the system administrator to customize SmarTeam Menu or toolbar profiles by adding and removing menus, sub-menus, commands or (image) buttons. Changed or new menu profiles are saved in the SmarTeam Database and are accessible to the SmarTeam Menu system.

IMPORTANT! Any changes made in the Menu Editor will only take effect in SmarTeam after saving your changes and exiting the Menu Editor. Whenever changes are made to a System Profile, the System Profile name appears in red until all changes have been saved.

To update the Menu Editor utility for a selected menu profile:

- 1 From **Start**, select **Programs > SmarTeam > Administrative Tools > Admin Console > Menu Editor**.

OR

From the SmarTeam main menu select **Tools > Admin Console > Menu Editor**.

The SmarTeam login window appears.

- 2 Type your Username and Password if necessary and then click **OK** to save your entries and close the window.
- 3 In the **Editor Tree** pane, select **Menu Commands > Default**.
- 4 Right click on **User Defined Commands** and from the popup menu, select **New User Defined Command** to view the New User Defined Command window.
- 5 In the New User Defined Command window, add the new **Full Text Search** command by completing the following fields:
 - **Caption** field: Type the new **Full Text Search. Internal name** command name. This field is accessed automatically when you complete the **Caption** field.
 - **User Script** : Click on **Browse** to view the Select User Defined Script window. Select **Full Text Search** and then click **OK** to save your selection.
- 6 Click **OK** to save your entries and add the new command to the **User Defined Commands** menu tree.
After adding the new command, the new menu appears on the **User Defined Commands** menu tree.
- 7 In the **Editor Tree** pane, select **Menu Profiles > System Profiles > System > Default > Pull Down Menus > SmarTeam Main Menu > Actions** and then select **User Defined Tools**.
- 8 Right click on **User Defined Tools** and from the popup menu select **New Menu Item** to view the New Menu Item window.
- 9 In the New Menu Item window, add **Full Text Search** by completing the following fields:
 - **Menu item type** field: Click on the dropdown list and select **Command Item** (if not already selected).
 - **Caption**: Type the new menu name, **Full Text Search**.
 - **Internal name**: This field is accessed automatically when you complete the **Caption** field.
 - **Command**: Click on **Browse** to view the Select Command window. Scroll down to locate **User Defined Commands** and then select **Full Text Search**. Click **OK** to exit the Select Command window and save your selection.
 - **Visible**: Select this option to verify this menu is visible to users.
 - **Customizable**: Select this option to enable a defined user to customize this menu.

Click **OK** to save your entries and add the new menu to the **User Defined Tools** menu tree.
After adding the two new menu items, the new menu appears on the **User Defined Tools** menu tree.
- 10 In the **Editor Tree** pane, select **Menu Commands > Default > User Defined Commands** and then select **Full Text Search**.
- 11 Right click on **User Defined Tools** and from the popup menu, select **Edit** to view the Full Text Search Properties window. Click on the **Icon** tab to view the **Icon** page.

- 12** In the **Item** page, click **Select Icon** to view a standard Windows Open window, displaying the contents of the <SmarTeam>\Icons folder.

- In the **Files of type** field, click on the dropdown list and select Bitmaps (*.bmp).
- In the main window, select FullTextSearch.bmp.
- Click **Open** to add the **Full Text Search** icon to the **SmarTeam Standard** toolbar.
The icon name FullTextSearch.bmp appears on the **Icon** page.
- Click **OK** to save your entry.

Changes made in the Menu Editor only take effect in SmarTeam after you save your changes and exit the Menu Editor.

Whenever changes are made to a System Profile, the System Profile name appears in red until all changes have been saved.

- 13** After making changes to a System Profile, save your changes:

- Select the **System Profile** folder in which you want to save changes
- Right click on the folder and from the popup menu, select **Save**.

OR

- From the **Menu Editor** main menu select **File > Save Profile**.

- 14** From the Menu Editor main menu, select **File > Save All Profiles** to save changes to more than one System Profile at the same time.

After saving a changed System Profile(s), the profile's name reverts to black text, indicating that all changes have been saved.

Note: If you attempt to exit the Menu Profile utility without saving any changes made to a System Profile, a SmarTeam warning window appears, prompting you to save your changes.

- 15** In the SmarTeam warning window, click **Yes** to save any changes made to the System Profiles or click **No** to abort any changes made and exit the Menu Editor utility.
- 16** Click **Cancel** to return to the Menu Editor utility without exiting and without saving any changes made to the System Profiles.

Configuring SmarTeam Full Text Search

Note: **IndexingServer > Machine Name** and **Catalog** settings are only relevant when a search on files within the vault is required.

To configure SmarTeam Full Text Search Client configuration settings:

- 1** Open the SmarTeam System Configuration Editor.
- 2** Search for the **SmartFTS. IndexingServer** key.
The configuration set and key name appear.
- 3** Click on the key name. The values of the key for various override levels appear. (If a value does not exist at the desired override level, you can add it by clicking **Add Value**.)
- 4** Click on the value at the desired override level, insert the name of the SmarTeam Full Text Search Server in the value field and click **Save Changes**.

- 5 Similarly, search for the **SmartFTS.MachineName** key and set the value to Microsoft Indexing Service Machine Name, which is the name of the machine on which the Microsoft Indexing Service is installed (usually it has the same value as the IndexingServer key).
- 6 Similarly, search for the **SmartFTS.Catalog** key and set the value to the name of the indexing catalog as it appears in the Microsoft Indexing Service.
- 7 Similarly, search for the **SmartFTS.DisplayedFields** key and set the value to **TDM_ID;CLASS_ID;REVISION;TDM_DESCRIPTION**. This value holds the name of the attributes that appear in the results window. The values should be separated by a semi-colon (;).
- 8 Similarly, search for the **SmartFTSMaxResultCount** key and set the maximum objects to be viewed as the result list. If the number of results set is higher than the maximum results SmarTeam can display, the maximum result count is the system's limit. We recommend that you do not input a number greater than 100. If a search returns a higher number of results, the user should refine the search.
- 9 Similarly, search for the **SmartFTSDefaultOperator** key. Set the FTS default operator. Available values are **AND** / **OR**.
- 10 Similarly, search for the **smartFTSCheckFiles** key and **smartFTSCheckMetaData** key to set the default check in method to Files only / DB only / DB and Files. For both keys the available values are **TRUE** / **FALSE**.

Rescanning FTS Directories After Adding an IFilter

After defining an IFilter, you will need to rescan the indexed directories.

To rescan the indexed directories:

- 1 From **Start**, select **Settings > Control Panel > Administrative Tools > Computer Management** to view the Computer Management window.
- 2 In the Computer Management window, select **Services and Applications**.
 - Select **Indexing Service**.
 - Select **SmartFTS > Directories** and then select **Directories** to view the **Full Text Search** indexed directory in the right pane.
 - In the right pane, right click on the displayed directory and then select **All Tasks > Rescuing (Incremental)** to start an incremental scan of the selected indexed directory.

Note: Starting a full scan results in rescanning all the documents in a catalog's scope, which could take a long time. Queries continue to return documents, but query speed could be affected during scanning and indexing.

The Incremental Rescan message window appears, prompting you to confirm the rescan operation for the selected indexed directory.

- 3 Click **Yes** to continue with the rescan operation.

OR

- Click **No** to abort the rescan operation.
- 4 In the Computer Management window, click on **Close** to exit.

Changing the Default Settings of SmarTeam Core Services

To change the default settings of Core Services, e.g., changing the port number, open the **SystemConfigurationRemoting.config** file, which is located in the SmarTeam Bin directory on the machine on which the System Configuration Service is installed.

Microsoft Installer

After installing any SmarTeam product, do not remove or rename any file or directory.

The Microsoft Installer may appear when you launch a SmarTeam application if a directory or file has been deleted, changed or renamed. To prevent this, do the following:

- 1** Open the computer's Event Viewer.
- 2** Search for information or an error event related to the Installer.

For example, a possible cause could be the deletion of the UpdatedScripts folder under the script directory.
- 3** After finding the cause, take the required action. For example, restore a modified file name to its original name, or restore the deleted file.

Chapter 8: SmarTeam Report Connector

What is the Purpose of SmarTeam Report Connector?

SmarTeam Report Connector enables administrators to securely retrieve information from SmarTeam and generate customized reports using standard third-party report generators, such as Crystal Reports®, Microsoft® Visual Basic (VB), and Microsoft® Office Excel. The report tool connects to the SmarTeam database through a designated OLE DB provider.

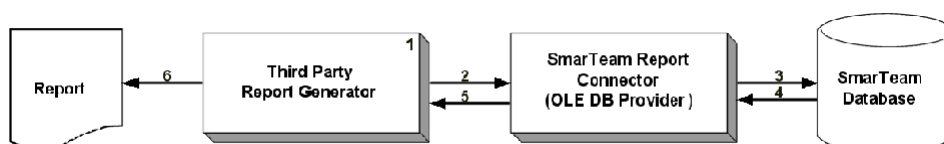
Administrators can optimize the value of SmarTeam data by:

- Monitoring, tracking and analyzing trends in products, projects, processes and tasks
- Producing high quality reports and charts in many common file formats, such as DOC, XLS, PDF, and HTM
- Distributing dynamic reports over the Web in real time
- Producing cross-system reports that integrate information from SmarTeam and other systems, such as ERP

Data Flow

Administrators can securely retrieve information from SmarTeam and generate tailored reports about various activities using customized report templates via standard third-party report generators. The report generator connects to the SmarTeam database through a SmarTeam Report Connector, which is a designated Microsoft OLE DB provider that facilitates access to different data sources. This OLE DB provider is version specific and not backward compatible with the previous Report Connector. OLE DB uses Structured Query Language (SQL)-92 to bridge between a variety of report generators and the SmarTeam database.

Note: Standard SQL-92 is partially supported by the SmarTeam Report Connector. See [SQL-92 Format](#) for the supported SQL-92 script format and syntax.



- 1 An administrator sends a command to the report generator to produce a report.
- 2 The report generator sends the request to the SmarTeam OLE DB provider (Report Connector).

- 3 The SmarTeam OLE DB provider (Report Connector) sends the request to the SmarTeam database.
- 4 Data streams from the SmarTeam database via a SmarTeam API to the SmarTeam OLE DB provider (Report Connector), which converts the data into readable strings that the report generator recognizes.
- 5 Data is sent from the SmarTeam OLE DB provider (Report Connector) over SQL-92 to the report generator.
- 6 The report generator produces a report at the request of the administrator.

Note: For effective usage, a ratio of 1 report generator to every 5 SmarTeam users is recommended.

Data Structure

Data within SmarTeam is divided into the following schemas, and is accessed from tables in the application:

- **Classes:** Collections of related SmarTeam metadata, presented in table format. Classes names are internal names, which are defined in the Data Model Designer (DMD). A classes schema contains the following types of classes:
 - **SmarTeam Classes:** Include all available SmarTeam classes, such as Users, Projects, Items and Folders
 - **Link Classes:** Include relations between SmarTeam classes, such as shown in a Documents Tree, presented in table format
 - **Lookup Classes:** Include a unique set of values referenced by SmarTeam classes attributes, such as State, and Phase, presented in table format
- **Queries:** Information defined by SmarTeam saved queries, which contains a user's saved searches and public saved searches.

Note: Attribute names are assigned a prefix.

After selecting **Tables** in the application (for example, the Database Expert of the report generator), a lists of classes, links and lookups appear in table format. All SmarTeam data (such as saved queries in SmarTeam) are accessible by administrators according to the same security requirements defined within SmarTeam.

Setup Prerequisites

Software and Hardware Requirements

The SmarTeam Report Connector software and hardware requirements are based on the SmarTeam – Editor Requirements. See the SmarTeam Hardware and Software Requirements documentation for details.

To use the SmarTeam Report Connector, one of the following standard report generator applications must be installed on your system:

- Crystal Reports® (v.X and v.XI)
- Microsoft® Visual Basic (VB v.6 and Visual Studio 2003)
- Microsoft® Office Excel (Microsoft® Office Excel 2000 and Microsoft® Office Excel 2003)

Note: Each major version of SmarTeam supports the current version of these report generators as well as the previous version.

Installation of the SmarTeam Report Connector Utility

SmarTeam Report Connector is installed automatically as part of the SmarTeam – Editor installation process.

Security

SmarTeam Report Connector uses the SmarTeam authentication and authorization mechanism via session management, eliminating the need to develop a parallel set of authorizations and providing a convenient and familiar working environment for administrators.

Licensing

Licensing for the SmarTeam Report Connector is configured automatically via SmarTeam – Editor

Note: If SmarTeam – Editor is not installed, licensing is configured automatically via SmarTeam – Web Editor.

Implementation

There are various ways of implementing the SmarTeam Report Connector within a company. The options are:

- [Desktop Applications](#): Administrators can create a customized, predefined report template, which includes the required data fields needed by end-user clients. Based on the report template, administrators can save report data in common file formats enabling users to access it and continue their analysis using other application capabilities, such as report generators, development tools and Microsoft® Office Excel.
- [Web Applications](#): Report templates can be uploaded to a report generator server to enable report viewing by end-user clients.
- [In-house Development Applications](#): In-house applications can be easily developed to access the SmarTeam database and generate sophisticated SmarTeam reports.

Desktop Applications

A company purchases a report generator to produce detailed reports about various activities.

The system administrator generates report templates according to the company's needs based on preselected fields from within the SmarTeam database objects. The administrator places the report template files in a public location where users that have access to SmarTeam's database and a report generator can run the reports templates and retrieve the required data.

The reports are executed by users in their native format or via common file type formats, such as XLS and PDF. Users run these predefined reports using the appropriate tool.

When using native files, the report tool connects to the SmarTeam database objects through the SmarTeam Report Connector and a refreshed report is generated based on the predefined template.

Reports can be saved under a file-managed class, enabling users to view reports with the SmarTeam viewer.

See [Connecting to External Report Generators](#) for specific details on connecting to external report generators.

Web Applications

A company purchases a Web server report generator to enable each user to access uploaded reports via a Web browser. In this case, no installation is needed for running the uploaded report.

The administrator can create report templates similar to the desktop application templates, but these report templates are saved to the report generator Web server. After uploading, the administrator can choose to run the report at a given time and also use a push mechanism in order to provide the report to the user. The end-user can execute the report via a Web browser at any given time and in a variety of formats depending on the specific brand of Web server used.

In-house Development Applications

A company has developed an application that extracts data from the SmarTeam Report Connector and produces an EXE file of the developed project. After SmarTeam is installed on the user's machine the user can run the report.

Note: In this case, data is read-only, which means that data cannot be written back to SmarTeam.

Data Retrieval Methods

There are two methods of retrieving data from SmarTeam:

- **Predefined:** Create a predefined template (in the report generator application) by defining links between classes and lookups
- **Query:** Run a saved query, which is located within SmarTeam

Implementation Recommendations

When presenting reports in SmarTeam, these steps should be followed:

- Create a class in SmarTeam named **Reports**. This class should be file managed.
- Add a new instance to this class and attach the report to it. PDF, XLS, TXT formats are supported for report output, and can be viewed using the SmarTeam viewer. (This step is optional.)

For example, to view information in Crystal Reports, you need to implement a container for the Crystal Reports ActiveX and select the OCX viewer.

Typical Consumers

This section presents some typical consumers of SmarTeam Report Connector and demonstrates how they can use it to achieve their needs.

For example, a factory has various departments, such as engineering, project management and purchasing. These departments need different types of reports.

Table 1: Sample Report Requirements

Role/Report Type	Summary Report	Detailed Report	Offline Report	Real-time Online
Engineering Group Manager	Yes	No	Yes	No
Engineering Group	No	Yes	No	Yes
Project Manager	Yes	No	Yes	Yes
Sales/Purchasing	Yes	No	Yes	No

The following conclusions can be drawn from reading the Sample Report Requirements table:

- The Engineering Group Manager does not need a detailed report or a real-time report. The information provided by a summary report is sufficient to allow the manager to analyze the current situation and react accordingly.
- The Engineering Group requires a detailed report and a real-time report to analyze the current situation.
- The project manager needs a summary report, an offline report and also a real-time report.
- The sales department needs a summary report and an offline report to see what parts exist in the system.

SmarTeam Report Connector can provide customized reports for each role.

The following user scenarios include examples of reports that might be needed in the factory:

- Ms. F is working in a design office on a machine renewal project using SmarTeam. She has been asked to specify which documents are linked to released parts. The parts are still actually based on Time Effectivity, which means they are less than a year old. Based on this information, Ms. F must produce a Bill of Materials (BOM) report for each requested part and highlight it.
Ms. F generates an Excel report and sends it to the parts manufacturer.
- Mr. H needs to know the engineers' workload. To do this, he runs a report based on the current ECO processes running in SmarTeam and checks which employees have the most processes running as well as the duration of the processes. Based on this information, Mr. H can divide the workload evenly.
- Mr. D prints a report for the customer showing all the spare parts of a machine including price and status.
- Ms. K prints all the processes that have been captured by a user that are behind schedule.
- Mr. M needs a report that provides a historical project status including details per project and the trends of progress in the overall projects. To produce this, two data sources can be combined to see the current status in one data source compared to the historical data in another data source.

Using SmarTeam Report Connector, these employees can produce the relevant reports quickly and efficiently, reducing risks of errors that affect data management.

Connecting to External Report Generators

SmarTeam Report Connector connects to the following external report generators supported by SmarTeam:

- Crystal Reports: See [Connecting to Crystal Reports](#)
- Visual Basic: See [Connecting to Visual Basic](#)
- Excel: See [Connecting to Excel](#)

IMPORTANT! Before you initially connect to a report generator, you must first connect to SmarTeam. See [Defining a Default Database Connection](#) for additional information.

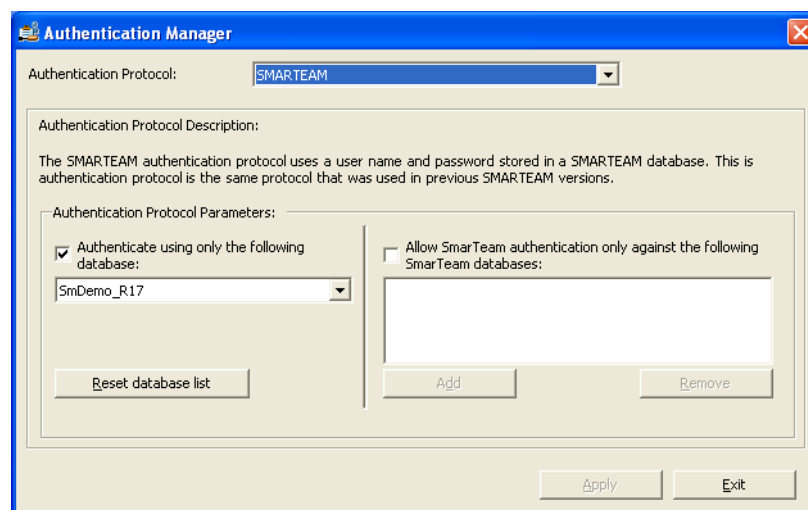
Note: The security used to connect to these report generators is SmarTeam security. Therefore, the information you see in these screens depends on the security rights assigned to you within SmarTeam.

Defining a Default Database Connection

Before you initially connect to any external report generators, you must first connect to the SmarTeam database as an authenticated user.

- 1 Navigate to Programs > SmarTeam > Administrative Tools > Authentication Manager.
- 2 From the Authentication Manager window, do the following:
 - a Enable **Authenticate using only the following database**.
 - b Select the appropriate database from the dropdown list.
 - c Click **Apply**.
 - d Click **Exit**.

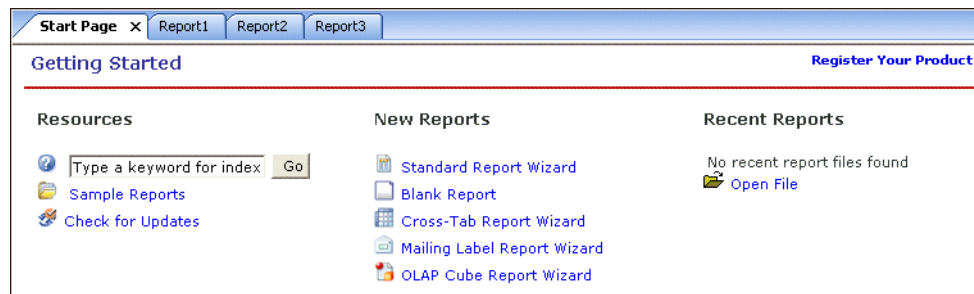
You are now an authenticated SmarTeam user.



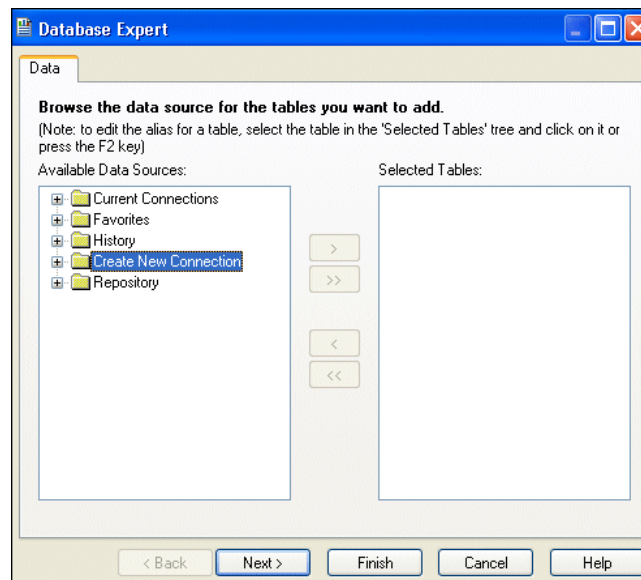
Connecting to Crystal Reports

To connect to Crystal Reports® XI:

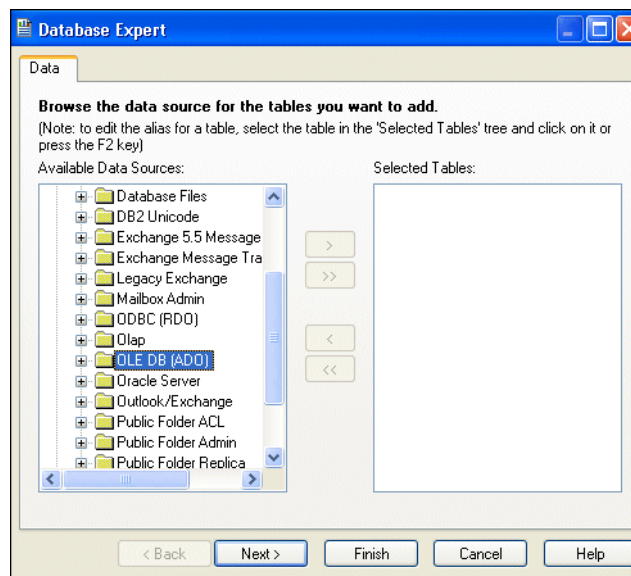
- 1 Open the Crystal Reports® XI application. From the Getting Started window, select **Standard Report Wizard**.



- 2 From the Data tab, select **Create New Connection**.

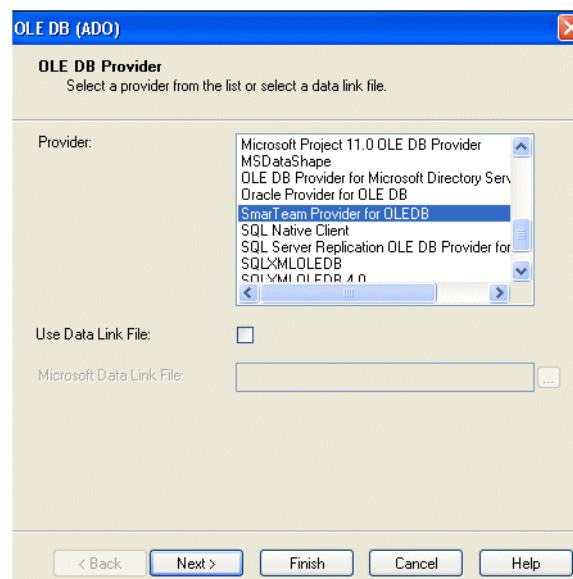


- 3 From the list of connections available, select **OLE DB (ADO)**.



4 Do one of the following:

- From the OLE DB Provider window, select **SmarTeam Provider for OLE DB** and click **Next**.



Continue to [Step 5](#).

OR

- From the OLE DB Provider window, select **Use Data Link File** and navigate to the data link file containing the connection information. This method can be helpful when saving reports, because Crystal Reports does not save password information.

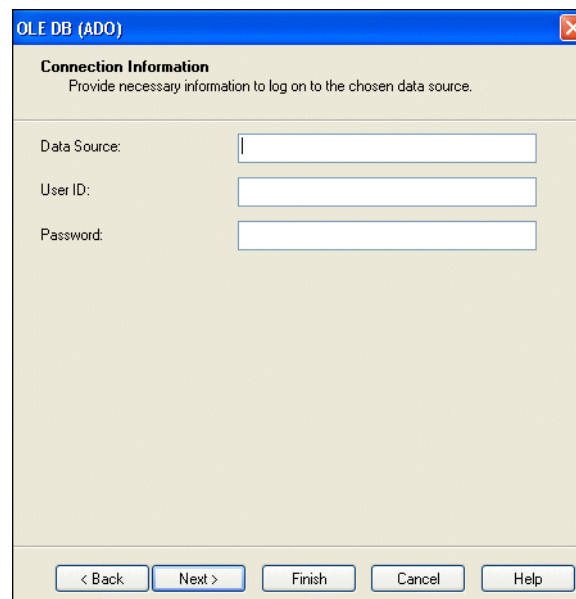
For information about creating a data link file, see [Data Link File Creation](#).

Continue to [Step 6](#).

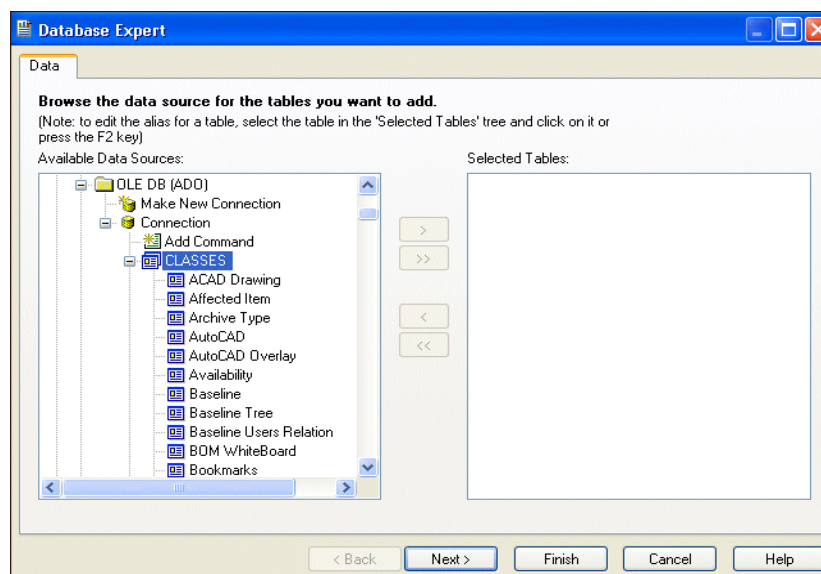
5 In the Connection Information window:

- If you are connected to one database (default), type your **User ID** and **Password** (as required), and click **Finish**.
- If you are connected to two databases and want to generate reports for the non-default database, type the **database ID** [case sensitive] for the non-default database in the Data Source field, type your **User ID** and **Password** (as required), and then click **Finish**.

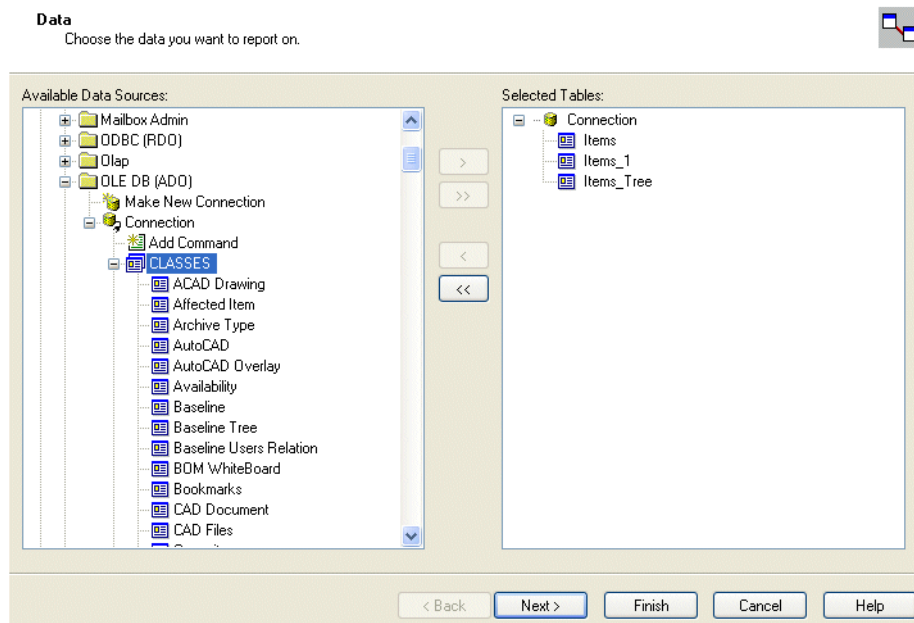
Note: To access your database ID, navigate to Programs > SmarTeam > Administrative Tools > Database Connection Manager. Double click on a required database and copy the ID.



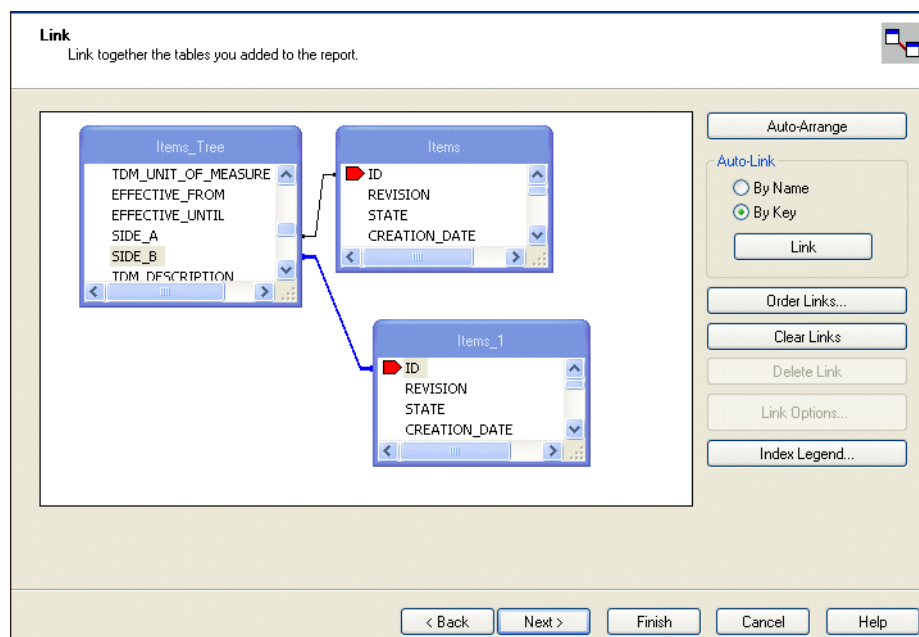
- 6 From the Available Data Sources, double click **CLASSES** or **QUERIES** to expand the list of available tables for both categories, which you can include in your report.



- 7 Double click each applicable table (listed under CLASSES in this example). The selected tables move to the Selected Tables area of the Database Expert window. Click **Next**.



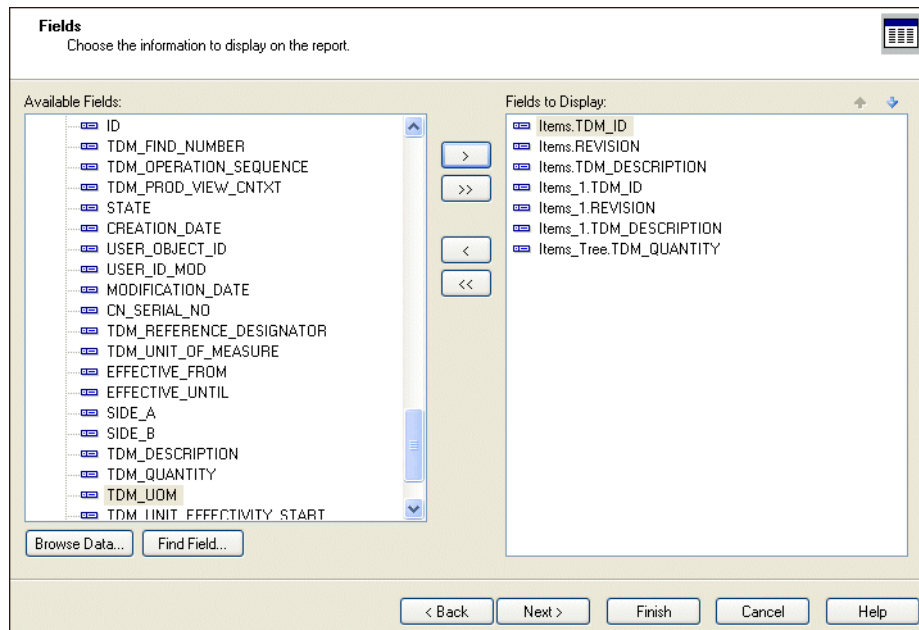
- 8 From the Link window, verify **By Key** is selected. If required, you can create additional links between the matched records of one table with the corresponding records of another table. Click **Next**.



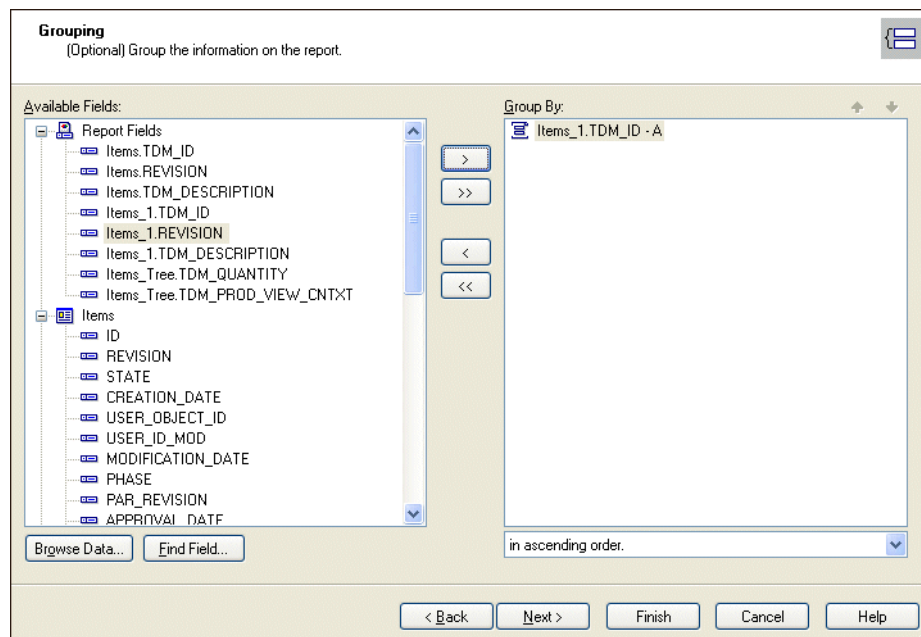
This example illustrates how to define the link between classes. The classes provide information about one level of a hierarchical structure of a model in which there is a parent item with children items. In addition, details about each specific link are available, such as quantity, unit of measure (UOM), and effectivity. This structure is known as a Part List report.

The Items class provides details about the parent item in Bill of Materials (BOM). The Items_1 class provides details about the children items in BOM. The Links show the connection between the ID field for Items class with Side_A field for Items_Tree class and between the ID field for Items_1 class with Side_B field for Items_Tree class.

- 9 From the Crystal_Fields window, double click the required fields for the Part List report. The fields appear in the Fields to Display area. Click **Next**.

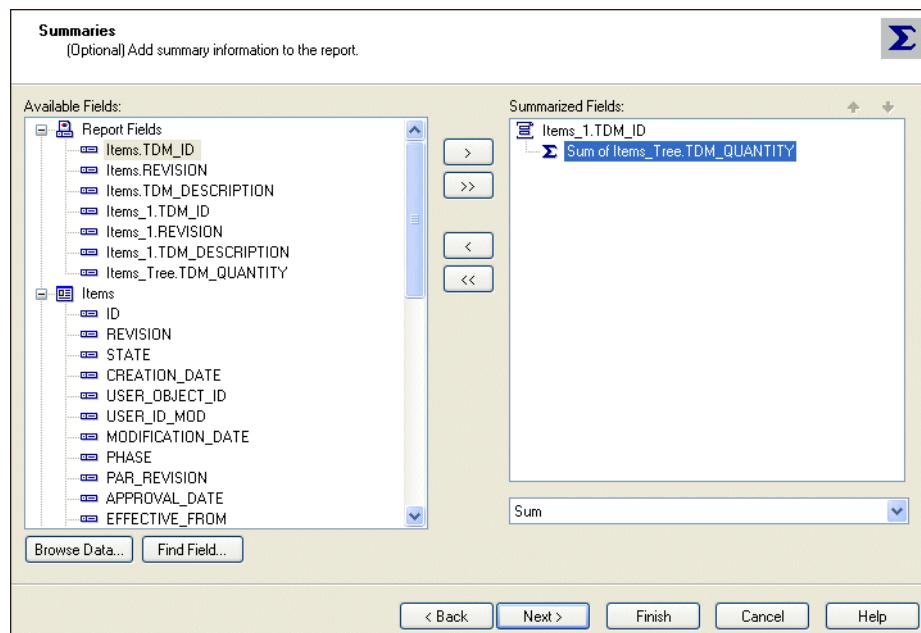


- 10 From the Crystal_Grouping window, double click a Report Field, such as Items_1.TDM_ID. The selected Report Field moves to the Group By area of the window. From the dropdown list, select **in ascending order** and click **Next**.
The Item IDs of the child items associated with this Report Field are presented in ascending order in the generated report.



- 11 From the Crystal_Summaries window, click **Next**.

Sum of Items_Tree.TDM_QUANTITY appears in the Summarized Fields area of the Crystal_Summaries window.



- 12 From the Crystal_Summaries window, delete **Sum of Items_Tree.TDM_QUANTITY** and click **Next**.

- 13 To add conditions to the report, you must assign values to specific fields. From the Crystal_RecordSelection window, select the required fields and the specific conditions from the dropdown lists. Click **Finish**.

Record Selection
(Optional) Select a subset of information to display.

Available Fields:

- Items_Tree
 - ID
 - TDM_FIND_NUMBER
 - TDM_OPERATION_SEQUENCE
 - TDM_PROD_VIEW_CNTXT
 - STATE
 - CREATION_DATE
 - USER_OBJECT_ID
 - USER_ID_MOD
 - MODIFICATION_DATE
 - CN_SERIAL_NO
 - TDM_REFERENCE_DESIGNATOR
 - TDM_UNIT_OF_MEASURE
 - EFFECTIVE_FROM
 - EFFECTIVE_UNTIL
 - SIDE_A
 - SIDE_B
 - TDM_DESCRIPTION
 - TDM_QUANTITY
 - TDM_UOM
 - TDM_UNIT_EFFECTIVITY_START

Filter Fields:

- Items.REVISION
- Items.TDM_ID
- Items_Tree.TDM_PROD_VIEW_CNTXT

is less than

0

< Back Next > Finish Cancel Help

14 The Report appears. The following is an example of a generated report.

Part List Single Level						
Parent ID	Parent Rev	Parent Desc	Child ID	Child Rev	Child Desc	Qty
Jet1185-L	A	Four-stroke BE-C co	BL-44-090-88S	A	Extensible Climbing	1
Jet1185-L	A	Four-stroke BE-C co	FP-98-098-78F	A	External panel prote	1
Jet1185-L	A	Four-stroke BE-C co	Jet109-887-99T	A	Water jet pump with	1
Jet1185-L	A	Four-stroke BE-C co	JetBE-4444-C	A	Mechanical extensio	1
Jet1185-L	A	Four-stroke BE-C co	M109-887-99	A	Two stroke/Twin cyli	1
Jet1185-L	A	Four-stroke BE-C co	R109-99-88	A	Handlebar for setting	1
Jet1185-L	A	Four-stroke BE-C co	RS-99-098-7T	A	Reinforcement to sta	1
Jet1185-L	A	Four-stroke BE-C co	Seat407-99-02	A	Seat 407	1

Data Link File Creation

To save user information, connection information and password, you can use a data link file with a UDL extension, which enables you to save time each time you are prompted for this information.

To create a data link file:

- 1 From Internet Explorer navigate to the location where you want to save the data link file.
- 2 Right click and select New > Text Document.
- 3 Rename the document by adding the **.udl** extension to the end of the document name.
- 4 Click **OK** if you are prompted to change the file type.

Note: If you are unable to change the file type, navigate to Tools > Folder Options > View and deselect **Hide extensions for known file types**.

The data link file appears.



To configure the data link file:

- 1 Double click the data link file.
 - 2 From the Data Link Properties window, select Provider > **SmarTeam Provider for OLE DB**.
 - 3 Click **Next**.
 - 4 From the Connection tab, complete the Data Source field as required.
 - 5 Type your username and password.
 - 6 Deselect **Blank Password** and select **Allow Saving Password**.
- Note:** If you want to test your connection, click **Test Connection**.
- 7 Click **OK**.

The Link Properties window closes and your connection information is saved.

Note: Your username and password are saved in the data link file as clear text and can be accessed by any authorized user.

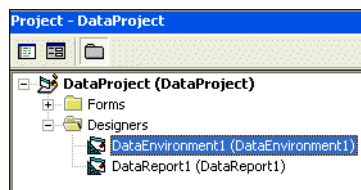
Connecting to Visual Basic

To connect to Microsoft® Visual Basic 6.0:

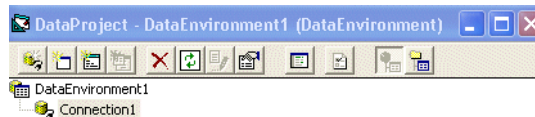
- 1 Navigate to Microsoft® Visual Studio 6.0 > Microsoft® Visual Basic 6.0.
From the New Project window, open a **New DataProject**.



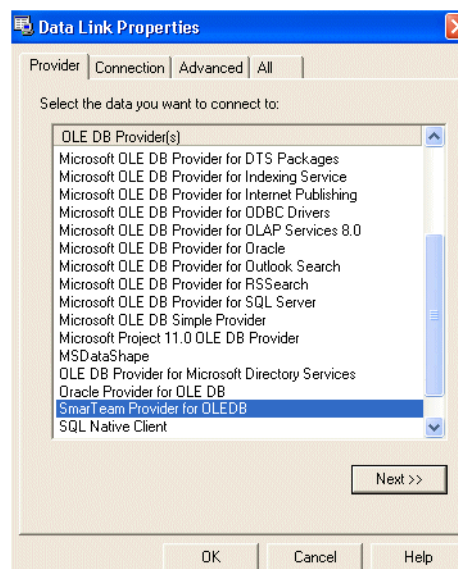
- 2 Double click a **Data Environment**, such as Data Environment 1.



- 3 Right click **Connection1** and select **Properties**.



- 4 From the Provider tab, select **SmarTeam Provider for OLEDB** from the list of OLE DB Providers and click **Next**.



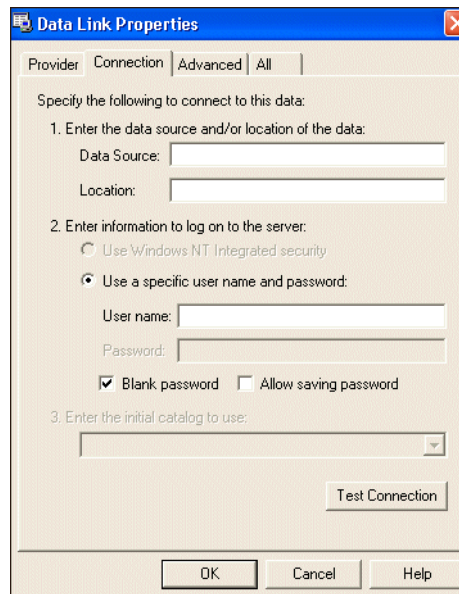
- 5 From the Connection tab:

- If you are connected to one database (default), select **Use a specific user name and password**, type your **User name** and **Password** as indicated, and then click **OK** to log on to SmarTeam.

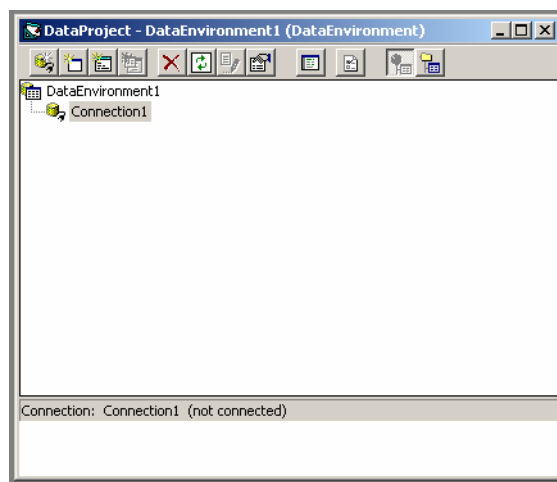
Note: It is recommended that you select **Allow saving password**.

- If you are connected to two databases and want to generate reports for the non-default database, type the **database ID** [case sensitive] for the non-default database in the Data Source field, select **Use a specific user name and password**, type your **User name** and **Password** as indicated, and then click **OK** to log on to the server.

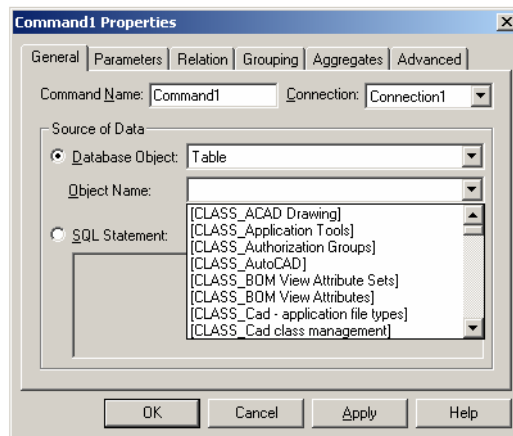
Note: To access your database ID, navigate to Programs > SmarTeam > Administrative Tools > Database Connection Manager. Double click on a required database and copy the ID.



- 6 From the Data Environment window, right click Connection1 again and select **Add Command** to add a new command.



- 7 Right click the new command and select **Properties**.
- 8 From the Database Object dropdown list, select **Table**.

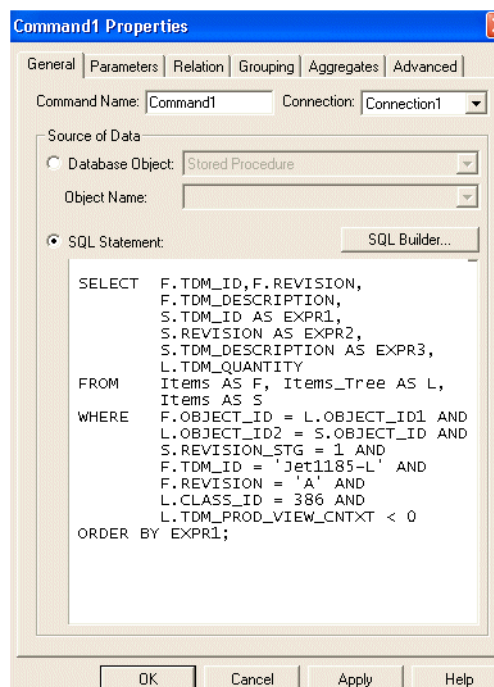


OR

Select **SQL Statement**, type the required SQL statement in the available field and then click **OK**.

If you want to copy and paste the SQL statement from this example in the available field, it is provided here:

```
SELECT F.TDM_ID,F.REVISION,F.TDM_DESCRIPTION, S.TDM_ID AS EXPR1,
S.REVISION AS EXPR2, S.TDM_DESCRIPTION AS EXPR3,L.TDM_QUANTITY
FROM Items AS F, "Items Tree" AS L, Items AS S WHERE F.OBJECT_ID =
L.OBJECT_ID1 AND L.OBJECT_ID2 = S.OBJECT_ID AND S.REVISION_STG = 1 AND
F.TDM_ID = 'Jet1185-L' AND F.REVISION = 'A' AND L.CLASS_ID = 386 AND
L.TDM_PROD_VIEW_CNTXT < 0 ORDER BY EXPR1;
```



9 Right click DataReport1 > View Object.

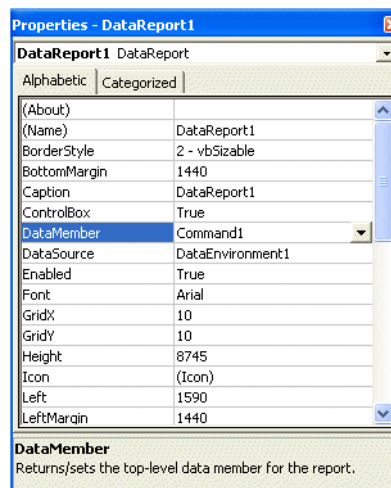
10 Drag and drop the required fields from DataEnvironment1 to DataReport1 as shown below.

Part List Single Level						
Parent ID	Parent	Parent Desc	Child ID	Child Rev	Child Desc	Qty
TDM_ID	REVISIO	TDM_DESCRIPTION	EXPR1	EXPR2	EXPR3	TDM_QUA

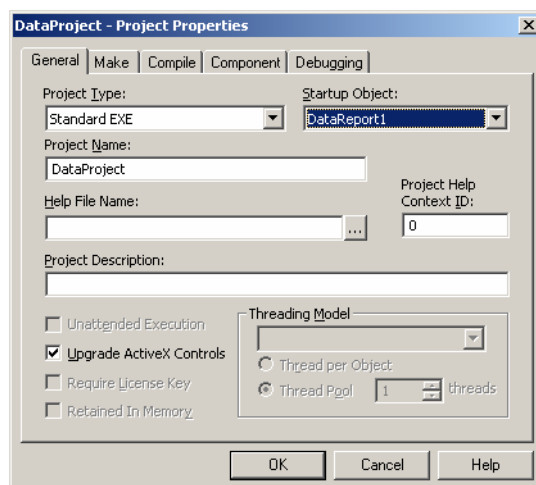
- 11** Click **DataReport1**. From Properties-DataReport1, select the following from the dropdown lists:

DataSource = DataEnvironment1

DataMember = Command1



- 12** Select DataProject > Project Properties. From the Startup Object field, select DataReport1.



- 13** Click **Start** to run the report. The following is an example of a generated report.

Part List Single Level

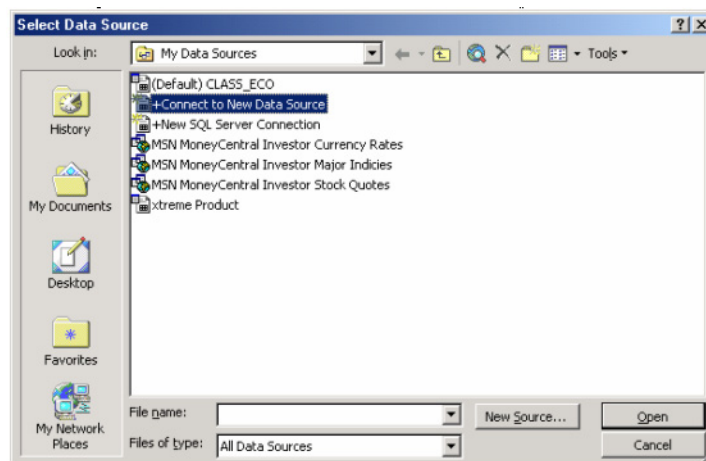
Parent ID	Parent Rev	Parent Desc	Child ID	Child Rev	Child Desc	Qty
Jet1185-L	A	Four-stroke BE-C	BL-44-090-88S	A	Extensible Climbing	1
Jet1185-L	A	Four-stroke BE-C	FP-98-098-78F	A	External panel	1
Jet1185-L	A	Four-stroke BE-C	Jet109-887-99T	A	Water jet pump with	1
Jet1185-L	A	Four-stroke BE-C	JetBE-4444-C	A	Mechanical	1
Jet1185-L	A	Four-stroke BE-C	M109-887-99	A	Two stroke/Twin	1
Jet1185-L	A	Four-stroke BE-C	R109-99-8S	A	Handlebar for setting	1
Jet1185-L	A	Four-stroke BE-C	RS-99-098-7T	A	Reinforcement to	1
Jet1185-L	A	Four-stroke BE-C	Seat407-99-02	A	Seat 407	1

Connecting to Excel

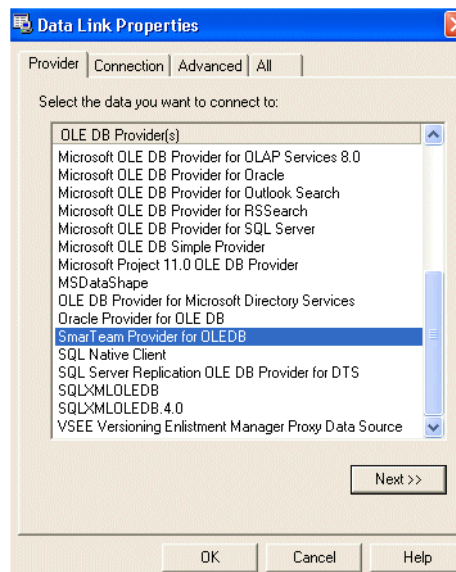
Note: When working in a .NET 2.0 environment, the SmarTeam – Office Integration cannot be activated unless you install Microsoft® Update "KB907417". To obtain this update, see: <http://www.microsoft.com/downloads/details.aspx?familyid=1B0BFB35-C252-43CC-8A2A-6A64D6AC4670&displaylang=en>.

To connect to Microsoft® Office Excel 2003:

- 1 Navigate to Microsoft® Office > Microsoft® Office Excel 2003.
Open a new Sheet.
- 2 From the Toolbar, select Data > Import External Data > Import Data.
- 3 From the Select Data Source window, select **+Connect to New Data Source**.



- 4 From the Data Connection Wizard window, select **Other/advanced** and click **Next**.
- 5 From the Data Link Properties window, select SmarTeam Provider for OLEDB from the OLE DB Provider list and click **Next**.



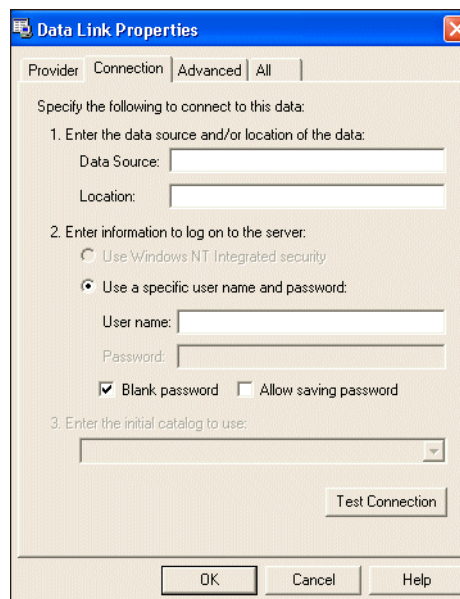
6 From the Connection tab of the Data Link Properties window:

- If you are connected to one database (default), select **Use a specific user name and password**, type your **User name** and **Password** as indicated, and then click **OK** to log on to SmarTeam.

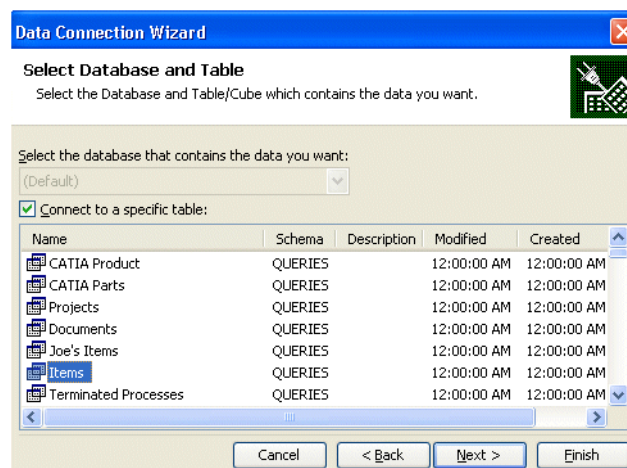
Note: It is recommended that you select **Allow saving password**.

- If you are connected to two databases and want to generate reports for the non-default database, type the **database ID** [case sensitive] for the non-default database in the Data Source field, select **Use a specific user name and password**, type your **User name** and **Password** as indicated, and then click **OK** to log on to the server.

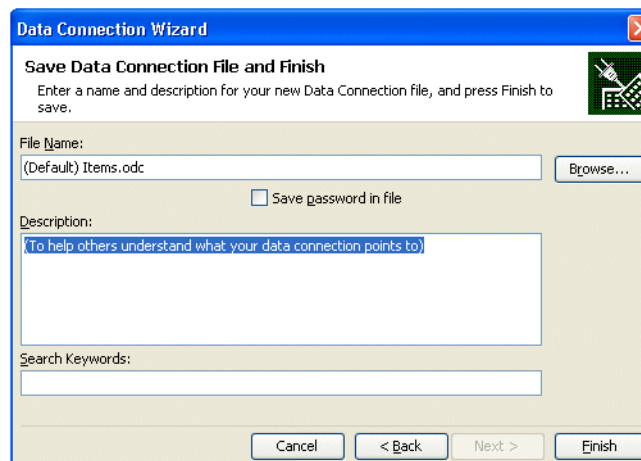
Note: To access your database ID, navigate to Programs > SmarTeam > Administrative Tools > Database Connection Manager. Double click on a required database and copy the ID.



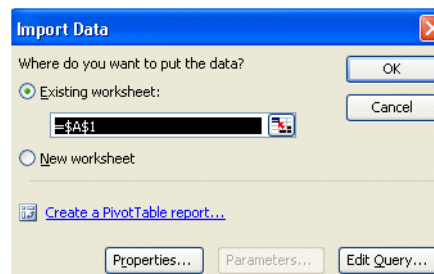
- 7 From the Select Database and Table window of the Data Connection Wizard select any table and click **Next**.



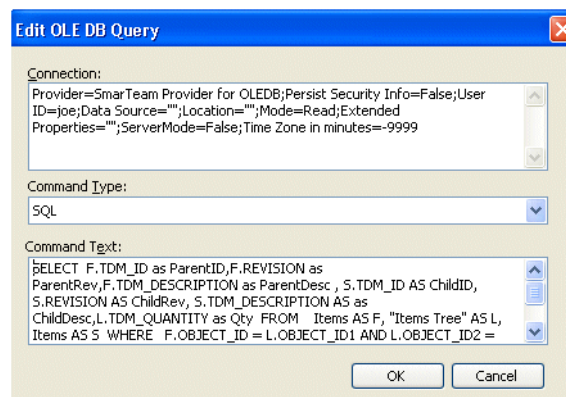
- 8 From the Save Data Connection File and Finish window of the Data Connection Wizard click **Finish**.



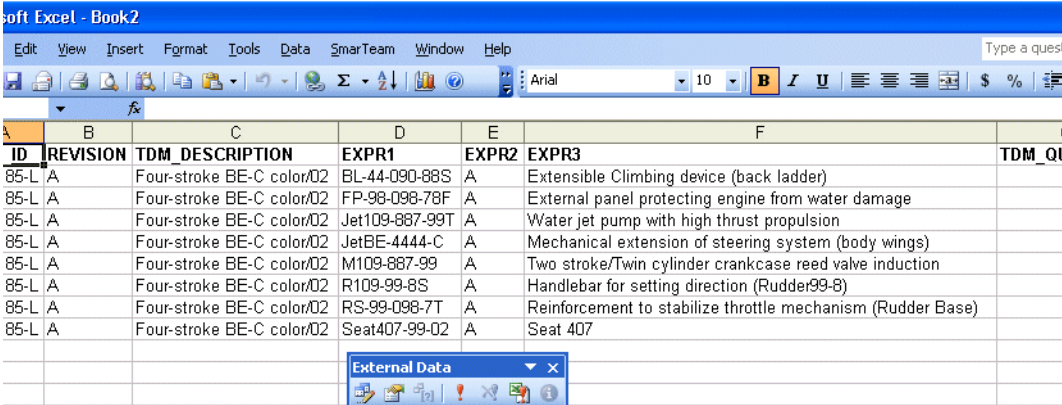
- 9 From the Import Data window, click Edit Query.



- 10 From the Edit OLE DB Query, select **SQL** from the Command Type dropdown list. Type the SQL statement in the Command Text field and click **OK**.



- 11 From the Import Data window, click **OK**.
- 12 The report appears. The following is an example of a generated report.



ID	REVISION	TDM_DESCRIPTION	EXPR1	EXPR2	EXPR3	TDM_Q1
85-L	A	Four-stroke BE-C color/02	BL-44-090-88S	A	Extensible Climbing device (back ladder)	
85-L	A	Four-stroke BE-C color/02	FP-98-098-78F	A	External panel protecting engine from water damage	
85-L	A	Four-stroke BE-C color/02	Jet109-887-99T	A	Water jet pump with high thrust propulsion	
85-L	A	Four-stroke BE-C color/02	JetBE-4444-C	A	Mechanical extension of steering system (body wings)	
85-L	A	Four-stroke BE-C color/02	M109-887-99	A	Two stroke/Twin cylinder crankcase reed valve induction	
85-L	A	Four-stroke BE-C color/02	R109-99-8S	A	Handlebar for setting direction (Rudder99-8)	
85-L	A	Four-stroke BE-C color/02	RS-99-098-7T	A	Reinforcement to stabilize throttle mechanism (Rudder Base)	
85-L	A	Four-stroke BE-C color/02	Seat407-99-02	A	Seat 407	

SQL-92 Format

This is the Structured Query Language (SQL)-92 script format and syntax that is supported by the SmarTeam Report Connector. SQL-92 is used to retrieve data, such as query definitions, from database management systems. For additional information about using SQL-92, see [Data Flow](#).

select_statement:

```
SELECT [DISTINCT] [TOP count]
      * | attribute_expression [, ...]
FROM from_item
[WHERE conditions]
[GROUP BY attribute [, ...]]
[HAVING conditions]
[UNION [ALL]] | INTERSECT | MINUS select_statement]
[ORDER BY attribute [ASC | DESC] [, ...]]
```

alias:

name

count:

number

attribute_expression:

```
<attribute | const | func([arguments]) | expression> [[AS]
alias]
```

```

arguments:
[attribute | const | func([arguments]) | expression] [, ...]

expression:
<attribute | const | func([arguments])> expression_operator
<attribute | const | func([arguments])>

expression_operator:
<||> | + | - | * | /

const:
[string | number]

from_item:
from_object [join_type from_item
              [ON join_conditions | USING (join_attribute[,
...])]]

from_object:
table | (select_statement) [[AS] alias]

table:
name

attribute:
[table.]name

join_type:
, | [[LEFT | RIGHT] OUTER | INNER] JOIN

```

join_conditions:

[() join_condition [logical_operator join_conditions] ()]

join_condition:

[NOT] attribute = attribute

logical_operator:

AND | OR

join_attribute:

name | table [[AS] alias] ** when table is a link class only

conditions:

[() condition [logical_operator conditions] ()]

condition:

attribute_expression operator [value]

operator:

< | > | = | <= | >= | <> | != | IS [NOT] NULL | LIKE | IN

value:

attribute_expression | func(arguments) | [() number [, ...] ()]
| [() string [, ...] ()]

string:

['] name ['] [|| string]

func:

group_func | format_func

group_func:

MIN | MAX | COUNT | SUM | AVG

format_func:

** detailed below

name:

\w+

number:

[+|-]\d+

Chapter 9: SmarTeam System Configuration Editor

Introduction

Overview

The System Configuration Editor provides a centralized mechanism that contains configuration-related information for all SmarTeam applications.

The system configuration service has multiple levels of configuration allowing easier manageability and security across sites, machines, applications, databases and users from anywhere in the organization.

The administration of the System Configuration Editor is performed using a Web-based application.

This chapter provides a description of the functionality and administration of the SmarTeam System Configuration Editor.

Upgrading from Previous SmarTeam Versions

If you are upgrading from a previous version of SmarTeam, you need to run the SmarTeam System Configuration Migration Wizard. For more details, see the SmarTeam Procedure for Upgrading to V5Rxx.

Migrating from V5R10 and V5R11

When upgrading from V5R10 and V5R11, all keys are migrated directly to the new System Configuration System.

Migrating from V5R12 and V5R13

There is a conceptual difference between SmarTeam R12 and R13 (and above) regarding the System Configuration Service:

- In R12, the System Configuration Service was not integrated into the entire SmarTeam system; it was only used by the Session Management service, NLS and Help sub-systems.
- From R13, the System Configuration was fully integrated in SmarTeam. All configurations that were previously saved in the database, .INI files and registry were merged into the System Configuration Service.

Terminology

Configuration Key (Element)

Configuration keys define the parameters in the SmarTeam applications.

Configuration Set (Header)

A configuration set is a collection of configuration keys. The configuration set represents the transactional unit for the System Configuration Service. Each configuration set is defined by a unique schema.

Note: In R10, configuration set is known as header.

In general, a configuration set represents a set of configuration elements that are logically grouped together. For example, the **smarteam.std.lifeCycle.config** configuration set holds keys related to the configuration elements of the lifecycle.

Configuration Files

A configuration file is a physical collection of keys on the computer's disk. Each configuration file contains keys, not necessarily related to its purpose. For example, keys related to lifecycle can appear in the configuration files **smarteam.std.legacyPreferences** and in **smarteam.std.lifeCycle**. Each configuration file has a unique name.

The following configuration files currently appear in the SmarTeam system:

Name	Description
smarteam.std.legacyPreferences	Contains configuration keys that was migrated from SmarTeam .INI files, Registry and Database
smarteam.std.clientLibraries	Contains the entire configuration for the client libraries sub-system.
smarteam.std.configClient	Contains configuration keys related to the System Configuration Client
smarteam.std.configurationManagement	Contains configuration keys related to the SmarTeam Configuration Management sub-system
smarteam.std.dynamicTypeMappings	Contains the mapping between user-defined types and their strong name types. This file is related to SmarTeam – Web Editor.
smarteam.std.embeddedScripts	Contains configuration keys related to the Embedded Scripts Service
smarteam.std.externalApplications	Contains configuration keys related to the add-ons of SmarTeam – Web Editor.
smarteam.std.FavoriteSearches	Contains configuration keys related to the favorite searches in SmarTeam – Web Editor.
smarteam.std.FileStorageManager	Contains configuration keys related to the File Storage Manager sub system.

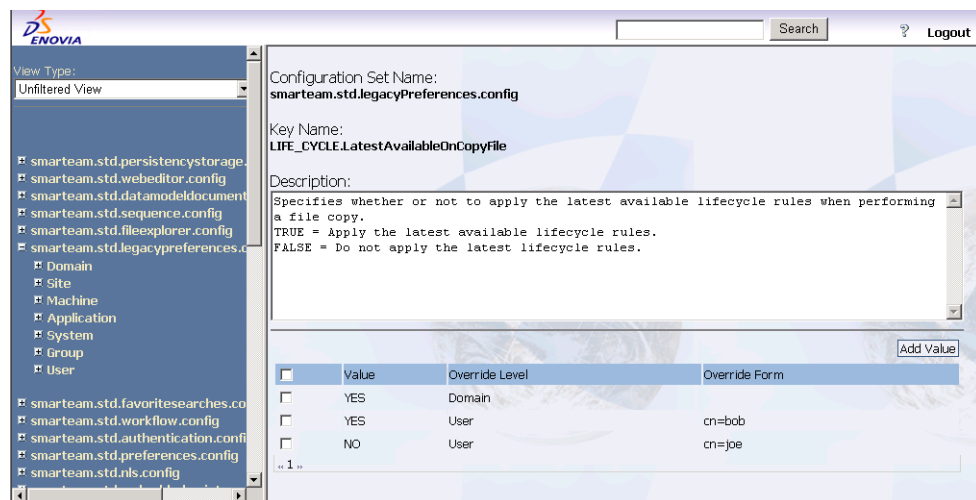
The System Configuration Level is a logical collection of System Configuration keys. These keys become the default keys for an end-user if they are not overridden on the lower levels.

The keys are not linked functionally, but are simply used to set up the defaults for the lower levels.

The System Configuration Service supports multiple configuration override levels to allow the definition of complex configuration combinations. The concept behind the configuration levels is the ability to allow information to be added or changed in the configuration according to the appropriate level.

In each level, values can be assigned to keys that either have or do not have different values in the previous levels.

If a key has different values in two different override levels, the value belonging to the lower level is used. A sample presenting override levels is shown:



The following table presents the available configuration override levels:

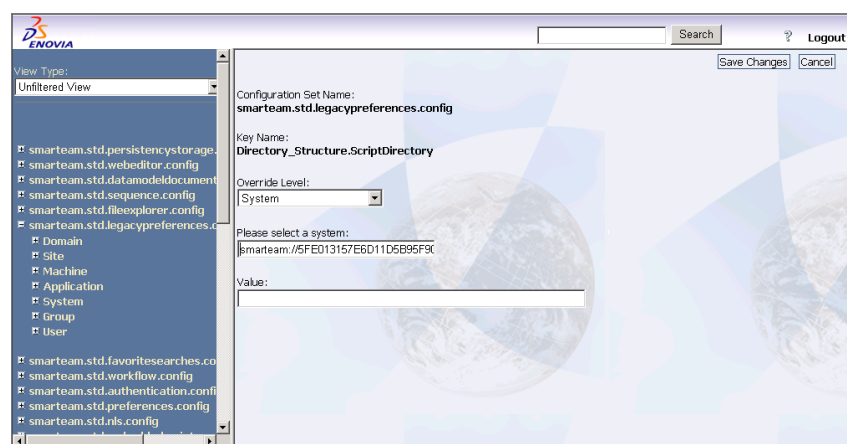
Configuration Override Level	Hierarchy	Description	Location
Default	0	The default values are used for all levels. If a value is not defined on the lower level, the default value is taken. Default values are read-only.	<SmarTeam>\ConfigurationSettings\Default
Domain	1	The Domain level is the topmost level. This level defines the default configuration for all applications, machines, system and users. Note: There can be several systems in the domain.	<SmarTeam>\ConfigurationSettings\Domain

System (Database)	2	The System level controls the configuration per system ID. A system ID represents any system to which SmarTeam applications can connect, such as a relational database. For example, a system ID can comprise a database replication ID.	<SmarTeam>\ConfigurationSettings\<systemid>
Site	3	The Site level is used to allow different configurations for different sites.	<SmarTeam>\ConfigurationSettings\Site
Machine	4	The Machine level controls the configuration per machine, which is identified by machine name.	<SmarTeam>\ConfigurationSettings\<machine-name>
Application	5	The Application level controls the configuration per application, which is identified by name. For example, SmarTeam – Editor, SmarTeam – Web Editor, SmarTeam – Community Workspace.	<SmarTeam>\ConfigurationSettings\<application-name>
User	6	The User level controls the configuration per user, which is identified by a unique text string, such as joe. Note: By default, this mostly concerns Visual Settings	<SmarTeam>\ConfigurationSettings\<user-strong-name>

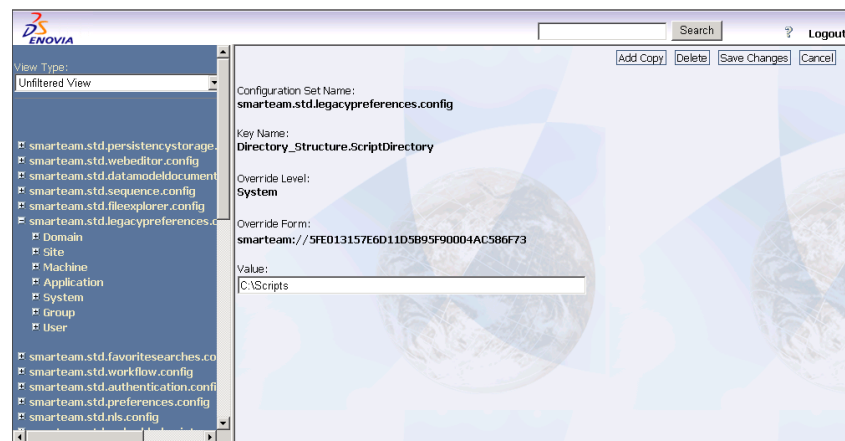
Example

The following example explains how the override levels work.

A new key is added on the system level:

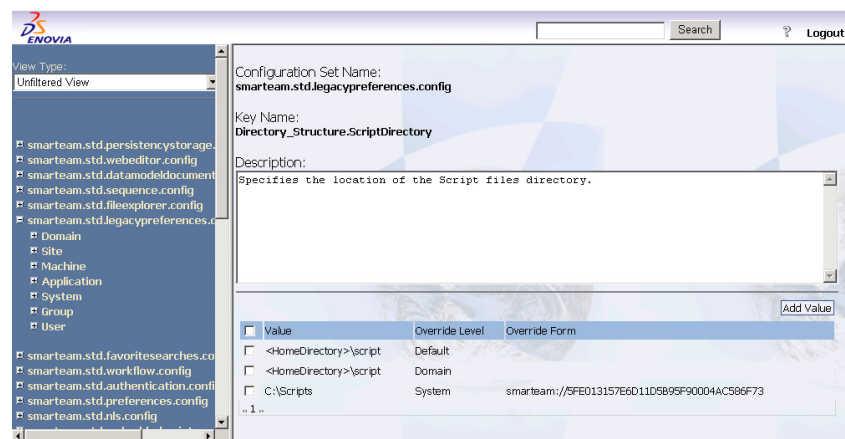


This definition is shown in the Override Form field for this key.



As shown, a user connected to the database with 5FE013157E6D11D5B95F90004AC586F73 DB ID is using the C:\Scripts directory .

For users connected to any other database, the Home Directory\Script directory is used.



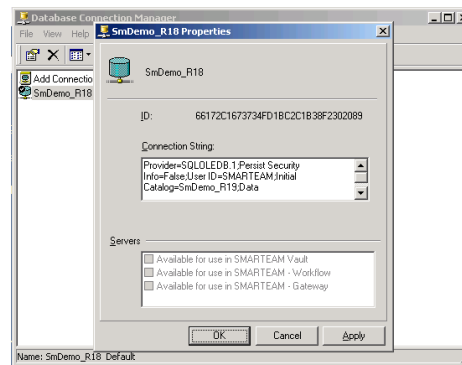
To obtain the database ID:

- Run the Database Connection Manager and search for the **smarteam.std.legacyPreferences.config** entry in the System Configuration Editor.

OR

- Run the Database Connection Manager, select the database and right-click on **Properties**.

The following window image appears:



The Database ID is shown at the top of the window.

Note: When working in the SmarTeam - Multi-site environment, the script directory is defined on the Site level instead of System level, so DB_REPLICID is used instead of DATABASE_ID.

Configuration Key Hierarchy

Configuration Key Summary

The Configuration Key Summary is a logical collection of the final settings of all keys that are applied for a specific user.

Example: In this table, Key 1 is assigned values at the Domain, System and User levels.

	Key 1	Key 2	Key 3	Level
	1	1	1	Domain Level
	2		2	System Level
	3			User Level
TOTAL	3	1	2	

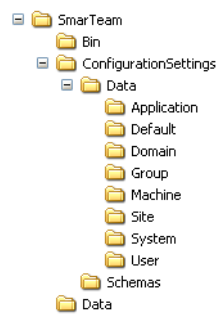
The final value for Key1 is 3 as the override level for Users takes precedence.

The final value for Key2 is 1 as only the Domain level exists has a value for this key.

The final value for Key3 is 2 as the override level for Systems takes precedence.

System Configuration Override Levels

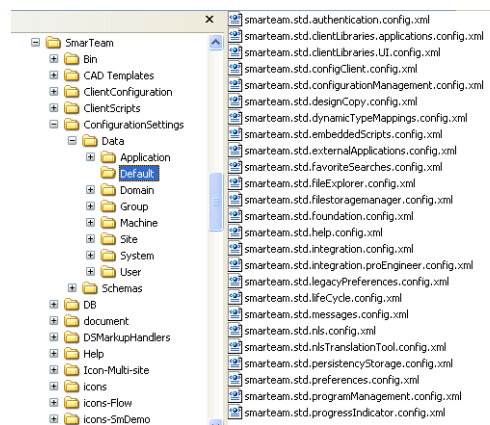
The following diagram shows a representation of the override levels contained in the System Configuration Service. The files are located in <SmarTeam Home Directory>\ConfigurationSettings\Data as shown:



Default Level

The settings defined at the Default level are located in the Default folder.

The differences between settings for each version are stored in the appropriate folder in the Default folder as shown:



Domain Level

The settings defined at the Domain level are stored in the Domain folder.

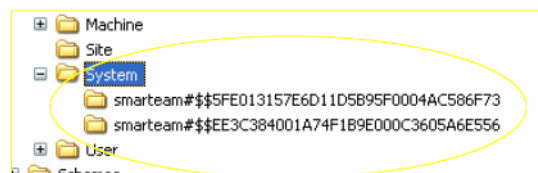
System Level

The settings defined at the System level are stored in the System folder. For each database, a sub-folder is created. The name of the sub-folder is in the following format:

smarteam#\$\$<databaseID>

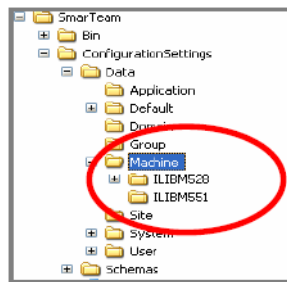
For example: smarteam#\$\$5FE013157E6D11D5B95F0004AC586F73

A sample System folder is shown:



Machine Level

The settings defined at the Machine level are stored in the Machine folder. For each machine, a sub-folder is created in which the modified xml files are stored. A sample Machine folder is shown:



User Level

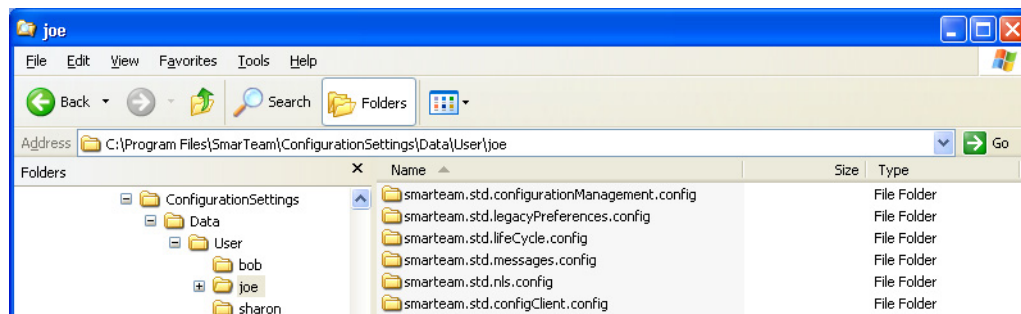
The settings defined at the User level are stored in the User folder. For each user for whom specific settings are defined, a new sub-folder is created. The name of the sub-folder is in the following format:

<user login name>

Example: bob

XML files that have been specifically modified for this user are shown in the User folder.

A sample of a User folder and its contents is shown.



Site Level

From R17, you can add / change configuration settings at the Site level. For example, when upgrading from R14 to R17, you can add configuration settings in a Test environment. All settings that are only relevant for the Test environment and should not be replicated in the Production environment should be saved in the Site level.

For example:

The key `SmarTeam.Database_Connection_Setup.1.Database_Connection_String` can be changed in the `legacyPreferences.xml` file in the Site Level in a Test environment to test the setup, but these changes are not replicated in the Production environment.

In Multi-site environments, if there are individual configuration settings on each site, this method works correctly. However, you cannot use a single configuration setting for more than one site in a SmarTeam - Multi-site environment.

Application Level

Not currently implemented.

Using the System Configuration Editor

The System Configuration Editor is an administrative tool that is only accessible to administrators. To access the System Configuration Editor, the user needs to provide authentication by supplying a username and password. When using the SmarTeam authentication protocol and when there is no specific database for authentication defined, the user can select the database to use when authenticating the username and password from the System Configuration Editor Login window.

Accessing the System Configuration Editor

To access the System Configuration Editor:

- From the Start Menu, select Programs > SmarTeam > Administrative Tools > System Configuration Editor.

OR

- Go to the following URL:
<http://<ComputerName>/SmarTeam/System/ConfigurationEditor/>
Where ComputerName is the name of the machine on which the utility is installed.

System Configuration Main Page

The System Configuration Service Main Page is divided into two areas:

Left panel: Contains a list or tree of the configuration sets

Right panel: Contains configuration keys and their values.

View Types

There are different ways of viewing information in the System Configuration Service. These are known as views.

To change the current view type:

- 1 Click **View Type** in the left panel.
- 2 The following is a list of View Types:
 - **Filtered View:** The configuration sets are represented as a hierarchical structure sorted into different groups. These groups can be customized according to your requirements. Note that configuration sets not added to a group cannot be seen in this view.
 - **Unfiltered View:** All configuration sets that exist in the System Configuration Service are shown even if they are not added to the Filtered View.
 - **Filter by Site:** The configuration sets for which a certain site has override values are shown.
 - **Filter by Machine:** The configuration sets for which a certain machine has override values are shown.
 - **Filter by Application:** The configuration sets for which a certain application has override values are shown.
 - **Filter by System:** The configuration sets for which a certain system ID has override values are shown.

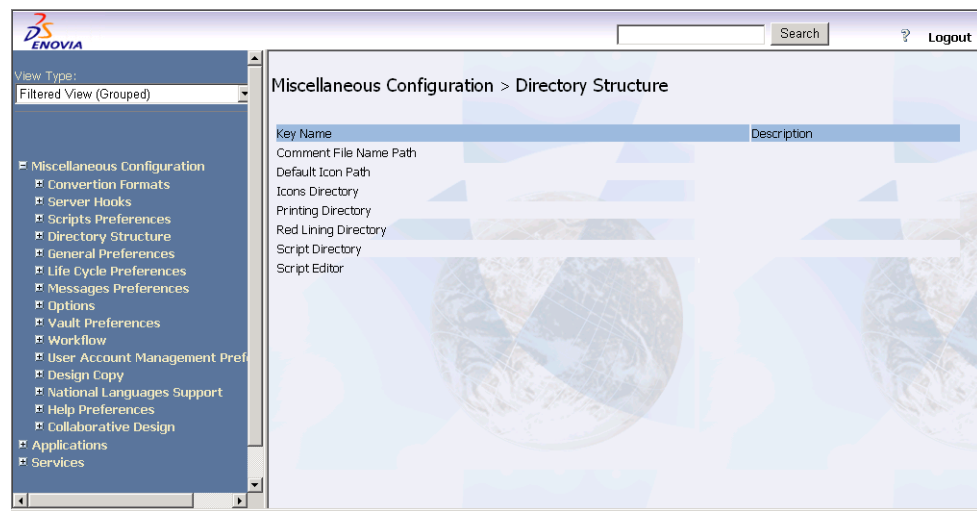
- **Filter by User:** The configuration sets for which a certain user has override values are shown.

Filtered View

In the Filtered View the configuration sets are represented in a hierarchical structure for ease of use. The configuration sets are sorted into different groups.

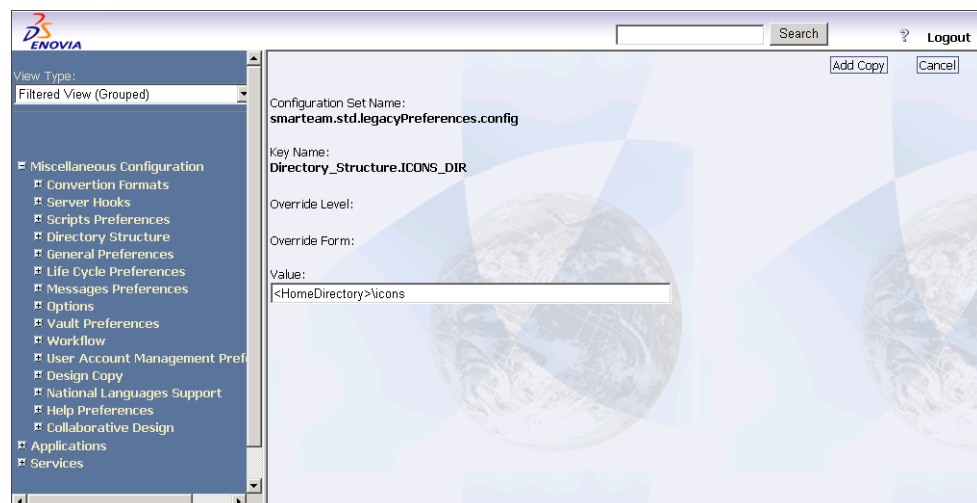
Note: These groups can be customized according to your requirements.

A sample filtered view is shown in the left panel:



To view the keys for a specific configuration set:

- 1 Click on **Configuration Set Name**.
A list of available override levels appears.
- 2 Click on an override level to view its override keys in the right panel.
- 3 Click on a key name to view its values, as shown in the following window image.



From this window, you can add a new key or delete an existing key. For more details, see [Adding a New Key Value](#), [Deleting a Key Value](#) and [Editing a Key Value](#).

Unfiltered View

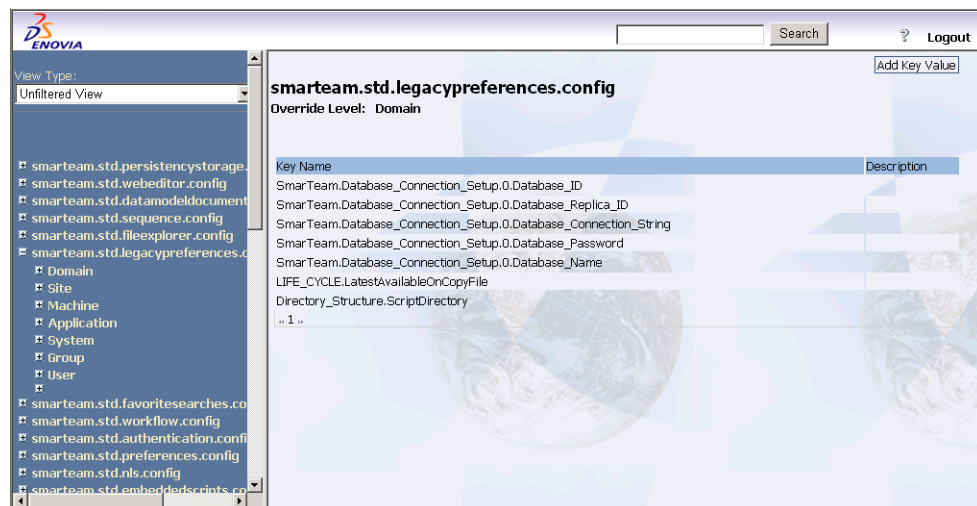
In the Unfiltered View the configuration sets are presented in a running list in the left panel.

To view the keys for a specific configuration set:

- 1 Click on **Configuration Set Name**.
A list of domains appears.
- 2 Click on a domain to view its override keys in the right panel.
- 3 Click on a key name to view its values, as shown in the sample window image.

From this page, you can add a new key or delete an existing key value. For more details, see [Adding a New Key Value](#), [Deleting a Key Value](#) and [Editing a Key Value](#).

A sample unfiltered view is presented in the following window image:



In addition to the Filtered and Unfiltered Views, you can also view the configuration sets according to Site, Machine, System, Application and User.

Adding a New Key Value

From each view type, a new key value can be added.

To add a new key value:

- 1 Click **Add Key Value**.
The Add Key Value window appears.
- 2 Complete the fields as follows:
 - Value: Type the value for the key.
 - Override Level: The desired override level for which this value is inserted
 - Override Form: The override form (depending on the override level) for which this value is relevant. For example, when typing a value that is relevant to a certain machine, the override form will contain its name.

For example, in the window image for a Time Stamp, a value of TestApp at the Application level overrides any values that appear at the System level or lower.

Deleting a Key Value

From each view type, an existing key value can be deleted.

To delete a key value:

- 1 Select a **Key Name**.
- 2 Click **Delete Key Value**.

The key value is removed from the System Configuration Service.

Editing a Key Value

You can update the value of a key at any time.

To update a key value:

- 1 Click on the Value field to display the values available for this key.
- 2 Click on the desired value.
- 3 Click **Save Changes**.

The key value is updated.

Implementation

Key Types

There are three types of keys in the System Configuration Service:

- **Static Key:** System keys that are optimized for read-only operations. Using these keys for read/write operations results in a tax of system resources.
- **Dynamic Key:** A specific type of user-defined key that is dynamically and automatically changed during a client's work. This type of key is normally defined during the customization process.
- **User-defined Key:** A key that is specifically defined by the user for customization purposes.

Note: When adding a user-defined key, it must be saved in a local .INI file on the client machine.

Connecting SmarTeam Applications to System Configuration Service through a Firewall

This section is related to the connection of SmarTeam applications to the System Configuration Service through Firewalls.

The System Configuration Service is a communication module implemented using Microsoft's Remoting.NET / SOAP Web Service technology. By implementing the System Configuration Service with Remoting.NET / SOAP Web Service, the System Configuration Service can work on top of any Remoting.NET-supported protocol. By default, the supported protocols are TCP and HTTP.

After installation, the default settings for the System Configuration Service are the protocol TCP on port 5607.

To change the default settings of the service, open the SystemConfigurationRemoting.config file located in SmarTeam's Bin directory on the machine on which the System Configuration Service was installed.

Mapping Pre-V5R13 Repositories

Configuration information that was previously stored in .INI files, registry and database preferences is now part of the System Configuration Service.

All types of keys that were inherited from previous SmarTeam versions are currently located in the smarteam.std.legacypreferences.config configuration set.

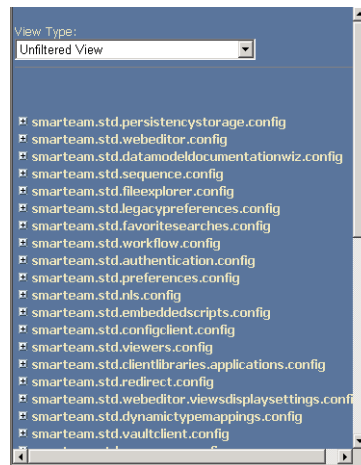
The System Configuration Service includes a search facility that allows users to find keys by their names. The search finds all elements whose name matches exactly or contains the specified text.

The search facility presents a list of all corresponding configuration elements and the configuration set to which they belong.

Clicking on the key name redirects you to the key view which shows all available values for this key.

In addition, you can use the information described in the following table.

Note: To view all keys of a certain configuration set, you must work in Unfiltered view. A sample Unfiltered view is shown:



The following table describes the method of mapping configuration data for all sources (.INI, Registry, database) to the System Configuration Service. The left column contains samples of entries in the old .INI files, Registry and Database Preferences. The right column contains the location and entries in the System Configuration Service that replaces these.

Configuration Location (PREVIOUS)	Mapping in System Configuration (CURRENT)
.INI Files	
Format: [SectionName] KeyName=Value Example: [Directory_Structure] ScriptDirectory=<HOMEDIRECTORY>\script	Format: SectionName.KeyName This parameter is found in the Configuration Set: smarteam.std.legacypreferences.config under the Key Name: Directory_Structure.ScriptDirectory
Registry	
Example: \$Admin\Database Connection Info\0\Database ID	This entry is found in the Configuration Set: smarteam.std.legacypreferences.config under the Key Name: Database_Connection_Info.0.Database_ID
Preferences (Inside the database)	
Example: CONVERSION_FORMATS.Date	Example: This entry is found in the Configuration Set: smarteam.std.legacypreferences.config under the Key Name: CONVERSION_FORMATS.Date

Transporting System Configuration

General

An administrator can prepare system configuration in one environment and then apply it safely to another environment. For example, you can create system configuration in the Test environment and then apply it in the Production environment.

System Configuration files created in one SmarTeam system contain a database identifier that must be changed while moving to another SmarTeam system.

The database identifier of the source system should be replaced with the database identifier of the target system.

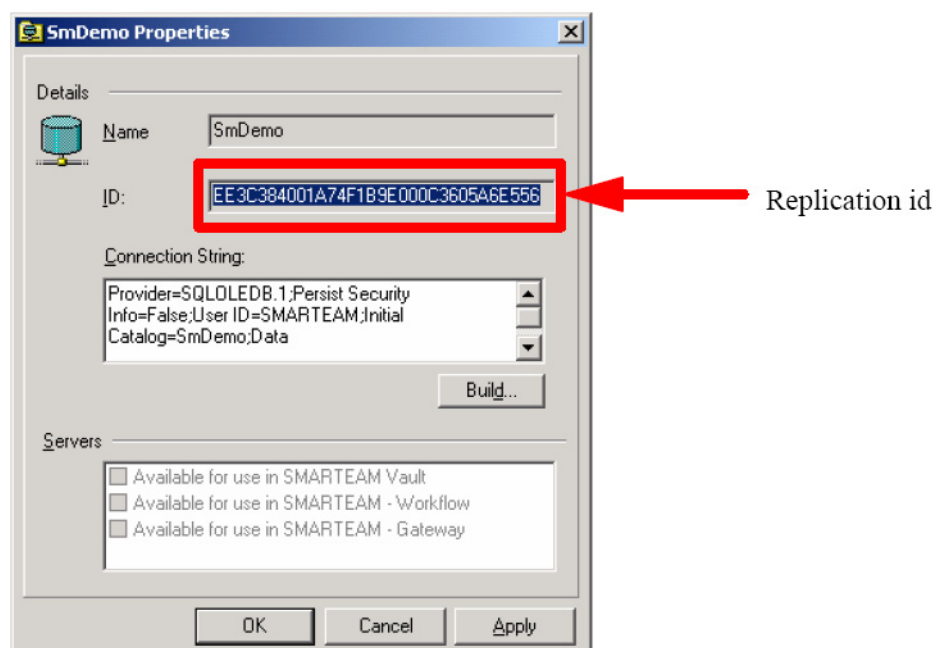
All entries in the System Configuration that are directly linked to the source environment elements, for example, computer names, IP addresses etc., should be replaced with appropriate data for the production environment.

To transport system configuration schema:

- 1 Copy the configuration files from the source server to the appropriate location on the target server.
- 2 Change all entries in the configuration files of the target configuration from source database DATABASE_ID (select DATABASE_ID from TDM_DB_VERSION) to the target database DATABASE_ID.

Note: Note: If you do not know the ID of the target database, you can copy it from the Database Properties window as follows:

- a From the Database Connection Manager, click on the required database. The Database Properties window appears.



- b Copy the Database ID string from the ID field.

Domain Level

The main file to be changed at the domain level is:

<SmarTeam>\ConfigurationSettings\Data\Domain\smarteam.std.legacyPreferences.config.xml

IMPORTANT! When migrating, before attempting to connect to the system, change the connection string. For full details on migration, see [Upgrading SmarTeam to V5R17.pdf](#). In SmarTeam - Multi-site systems, this issue is related to a site to which the System Configuration is copied. After copying the files, the connection string must be changed for each site.

User Level

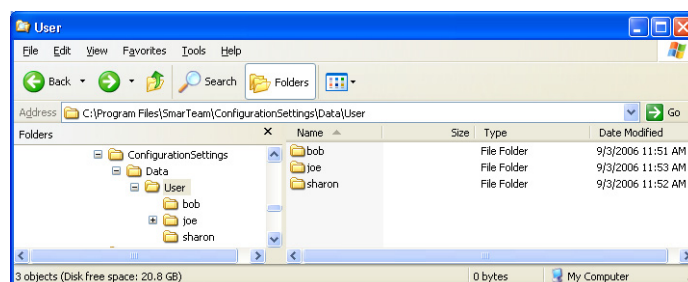
User-specific preferences can be created for one user and then distributed to other users. To do this, specific folders must be created inside the User System Configuration folder.

For example, if there are two users, such as Joe and Bob, a folder named **joe** is created in the USER System Configuration folder when Joe logs in for the first time.

To distribute this folder to Bob, do the following:

- 1 After creating the necessary System Configuration settings for Joe, such as appropriate Visual settings, log out of SmarTeam.
- 2 Copy the **joe** folder from the User System Configuration folder to the **bob** folder as shown.
- 3 When Bob logs in for the first time, he sees the same Visual settings as Joe.

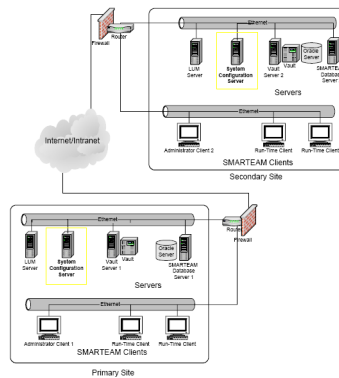
Note: If Bob has already logged in, even once, his own System configuration settings already exist. Therefore, to apply settings of Joe to Bob, you must first delete Bob's settings from the Bob folder and then replace the folder with the Joe folder.



Managing System Configuration in the MUS Environment

The System Configuration in the MUS environment behaves as in a regular environment, except for a few small changes. For full details, see the [SmarTeam - Multi-site Administration Guide](#).

The following diagram shows a typical Multi-site system:



SmarTeam – Multi-site Specific Changes

When transporting System Configuration in the SmarTeam - Multi-site environment, you can prepare System Configuration at one site and distribute the configuration to other sites.

To enable users to connect to their relevant sites, when transporting, the connection string in the System Configuration must be changed. This can be done in the System Configuration Editor.

The entire process of transporting System Configuration holds true also for the SmarTeam - Multi-site system. The only exception is DATABASE_ID, which is the same for all sites within the SmarTeam - Multi-site system.

Synchronization of System Configuration Files

When changing keys at one site, if you want to synchronize between sites, relevant files may be distributed to other sites or the same changes can be made once again at each site. This should be done when there are no users connected to avoid a situation in which sites have different settings for the same modules (e.g., Lifecycle), which may result in data corruption.

Note: It is strongly recommended that you locate the System Configuration service on a different server than the one on which SmarTeam is installed.

Sometimes there is a need to connect remote clients to a SmarTeam database and the MUS is not yet applicable. In that case, the SmarTeam recommendation is to place the System Configuration Server along with the NLS storage closer to the remote users. This may result in several System Configuration Servers related to the same database.

After using scheduled synchronization, note that to connect users to the correct vault, the following parameters need to be synchronized manually:

- Database Connection - Domain level
- Default Vault - System level

Adding Complex Keys to the System Configuration Service

IMPORTANT! Before adding complex keys, verify you back up all configuration setting files in the Configuration Settings Data directory.

To add a complex key:

- 1 Define the complex key in an XML editor, e.g., Visual Studio. While writing the definition of a key, verify the XML syntax is valid and the key name is a valid SmarTeam key name.
- 2 Use the editor's validation tool to ensure that the XML syntax is valid.
- 3 Open the SmarTeam Configuration Editor and find the relevant key.
- 4 If the key definition for this key does not exist, select the Add option to add it.
- 5 If the key definition for this key already exists, append the new definition to the end of the existing definition using the Edit option.
- 6 Copy the key definition from the XML editor, and paste it into the multi-line text box in the Configuration Editor.
- 7 For example, to add a new submenu containing two commands to the Application bar, add the following entries in the <applicationBar> key:

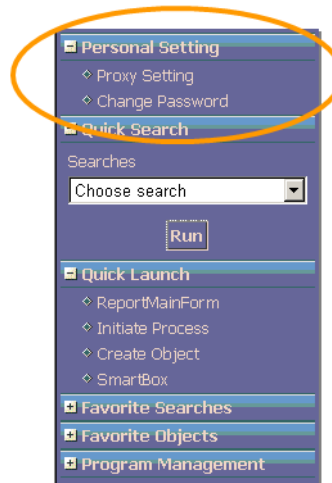
```
<menuItem>
  <commandInternalName>ud.ProxySettings</commandInternalName>
  <URL>/Views/UserDefinedTools/StatusReport.aspx</URL>
</menuItem>
<menuItem>
  <commandInternalName>ud.ChangePassword</commandInternalName>
  <URL>/Views/UserDefinedTools/StatusReport.aspx</URL>
</menuItem>
```

- 8 Save the changes.

IMPORTANT! After adding each key, it is highly recommended to run the SmarTeam application and test the changes you made.

Example:

The following example shows how to add a new submenu in the SmarTeam – Web Editor application bar.

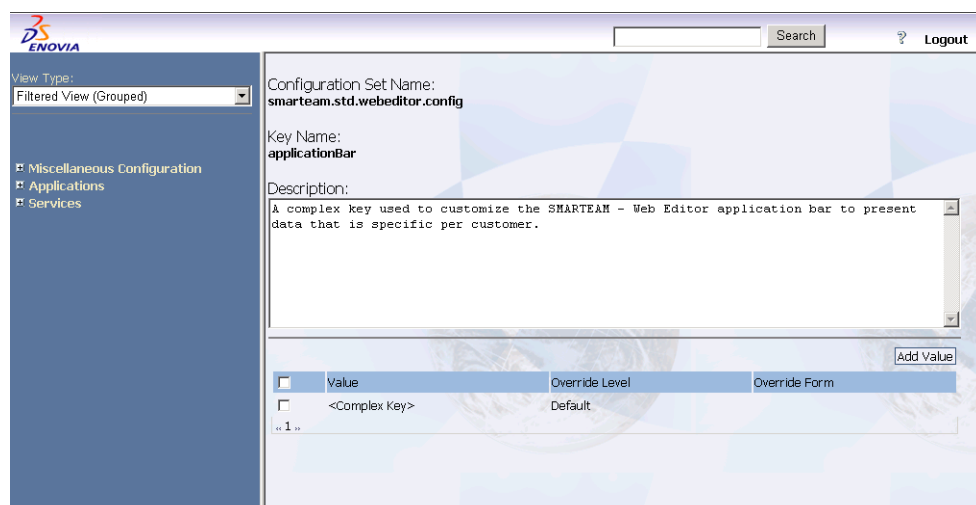


To define this new submenu in the SmarTeam Configuration Editor:

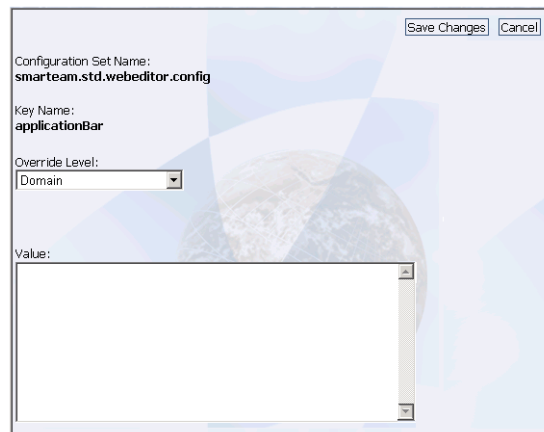
- 1 Open an XML editor and type the code as shown in the following image:



- 2 Open SmarTeam Configuration Editor and find the key <applicationBar>.



- 3 Click on **Save Changes**.



- 4 Restart SmarTeam Services.
- 5 Restart the SmarTeam – Web Editor and test your changes.

Adding a New Key to the System Configuration

To add a new key to the system configuration:

- 1 Create a schema file in the folder in which the current version of SmarTeam is installed: [SmarTeam HOME]\ConfigurationSettings\Schemas
- 2 In this schema file, define the new or updated key. This file must contain only the changes for this version and not the complete data. For example, if the original version contains two keys and you want to add one more key in this version, the file for this version must contain the new key only and not all three keys.
- 3 Add a new XML file in the corresponding location in the data folder ([SmarTeam HOME]\ConfigurationSettings\Data). This XML file should only contain delta information, according to the schema file.

IMPORTANT! In any event, it is prohibited to change SmarTeam's schema files!

Writing to the System Configuration Service using SmarTeam API (session smConfig)

In V5R10, users can add their own configuration parameters for customizing applications. These parameters are written in the SmarTeam INI files, e.g., SMTEAM32.INI with separate headers. Users can also create their own INI files, which can be accessed using SmConfig API functionality.

In the current version, the following rules should be applied when creating user-defined configuration parameters.

IMPORTANT! The following names are reserved names and CANNOT be used for manipulating user-defined configuration keys. Using these names will cause serious performance degradation.

smteam32.ini
SMGRID32.INI
SMGRDC32.INI
SCRGEN32.INI
SmVlt32.ini
SmWorkFlow.ini
SmERPSyncServer.ini
ServerSafeScripts.ini
WebServerSafeScripts.ini
smwiza32.ini
UpgradeSmartDatabase.ini
SmartDesk.ini

Notes:

If one of these .INI files is typed into the IniFileName property of SmConfig, it turns to the System Configuration instead.

It is not recommended to manually update SmarTeam configuration parameters using a SmarTeam API.

System parameters that need to be constantly updated at runtime should be stored in individual INI files located on the end-user's workstation.

Defining the Default Connection for the SmarTeam Database for Multiple Clients

The default client connection to a SmarTeam database is stored in the local computer registry and not in the System Configuration Service. This means that defining the default database at the System Configuration Server does not affect client computers using this System Configuration.

To avoid defining a default connection manually for each client computer, the relevant section of the registry on the System Configuration Server must be exported and imported at each client computer. This can be done as a part of the batch procedure.

The registry setting is located in:

HKEY_CURRENT_USER\Software\SmarTeam\SmarTeam Application\Recent Databases

Using Individual Configurations for Different Users

If the administrator needs to use configuration that is completely different from what is used for regular users, set up an additional Core Services computer and point admin client workstation on this server.

To switch client from one Configuration to another, do following:

- 1 Go to C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\CONFIG.

- 2 Open the file machine.config.
- 3 Locate the line (there should be 3 entries like this):
`"<smarteam><sessionManagement><client><url>tcp: 127.0.0.1:5607"`
- 4 Change the TCP/IP, for example 127.0.0.1 to the TCP/IP address of the new server.

Manually Editing System Configuration XML Files

It is not recommended to edit System Configuration XML files manually (i.e., in Notepad), as this can cause file corruption.

If you edit System Configuration XML files manually it is highly recommended to save them in UTF8 code page (UNICODE) for availability of the system configuration in far eastern languages, such as Japanese and Chinese.

System Configuration Service

A Microsoft® Windows application, which supports the system configuration is installed with the SmarTeam - Foundation as part of the Core Services. It connects SmarTeam applications with the information from the Configuration schema.

Configuration Schema

The configuration schema is a collection of definitions of all possible SmarTeam System Configuration keys available from the system.

The configuration schema format is a standard W3C XSD Schema. For more details, seeS <http://www.w3.org/XML/Schema>.

Configuring the System Configuration Service to Work with the Windows Protocol

The System Configuration Service can be configured to work with the Windows protocol.

In the web.config file located at <SmarTeam Home Dir>\Web\System\ConfigurationEditor the following two keys appear:

```
<authentication mode="Forms">
  <forms loginUrl="Authentication/LoginPage.aspx" name=".SMARTAUTH"/>
</authentication>

<authorization>
  <deny users="?"/>
</authorization>
```

- 1 Change the <authentication mode="Forms"> key to <authentication mode="Windows">.
- 2 Comment out or delete the line starting with "<forms...".
- 3 Comment out or delete the Authorization tag in its entirety including "<deny.." and its closing tag as shown in the following example.

Example:

```
<authentication mode="Windows">  
<!-- forms loginUrl="Authentication/LoginPage.aspx" name=".SMARTAUTH"/-->  
</authentication>  
<!--authorization>  
<deny users="?"/>  
</authorization-->
```

- 1** In the IIS Manager, go to the Default Web Site \SmarTeam\System\ConfigurationEditor.
- 2** Right click and select **Properties**.
- 3** Select the Security tab and click **Integrated Windows Authentication**.
- 4** Restart IIS.

IMPORTANT! You DO NOT have to change to Windows authentication in the Authentication Manager.

Configuring Core Services on a Multi-Network Card Machine

The connectivity used by SmarTeam for communication between Foundation clients and the server is .NET Remoting.

To ensure that Foundation services (Session Management and System Configuration) installed on a machine with multiple network cards work correctly, binding configurations must be applied manually.

The TCP channels in each Foundation service should be bound to the same network card.

To configure binding for the network card, you need to edit the `SessionManagementRemoting.config.xml` and `SystemConfigurationRemoting.config.xml` located in the `SmarTeam\bin` folder.

Each xml file contains tcp channel sections (beginning with `<channel ref=`) that contain the following code:

`SessionManagementRemoting.config`

```
<channel ref="tcp" port="5606">
```

```
...
```

```
...
```

```
</channel>
```

`SystemConfigurationRemoting.config`

```
<channel ref="tcp" port="5607">
```

```
...
```

```
...
```

```
</channel>
```

To configure binding:

- Add the `bindTo` attribute with the IP address of the network card to the "`<channel ref`" line as follows:

`SessionManagementRemoting.config`

```
<channel ref="tcp" port="5606" bindTo="x.x.x.x">
```

```
...
```

```
...
```

```
</channel>
```

`SystemConfigurationRemoting.config`

```
<channel ref="tcp" port="5607" bindTo="x.x.x.x">
```

```
...
```

```
...
```

```
</channel>
```

Where `x.x.x.x` is the IP address of the network card.

Configuring IIS 64bit to Work with an Application Running at 32bit

If you need to configure your 64 bit IIS machine to run an application at 32 bit, perform this procedure.

To configure an IIS 64bit for 32bit application:

- 1 Verify that ASP.NET is not installed on your server.
If ASP.NET is installed, remove it using the following example:
`%SYSTEMROOT%\Microsoft.NET\Framework64\v2.0.50727\aspnet_regiis.exe -ua`
- 2 Type the following command to enable 32bit mode:
`cscript %SYSTEMDRIVE%\inetpub\adminscripts\adsutil.vbs SETW3SVC/AppPools/Enable32bitAppOnWin64 1`
- 3 Install the ASP.NET.
If you uninstalled ASP.NET, you can use the following examples.
`%SYSTEMROOT%\Microsoft.NET\Framework64\v2.0.50727\aspnet_regiis.exe -i`
`%SYSTEMROOT%\Microsoft.NET\Framework\v1.1.4322\aspnet_regiis.exe -i`
- 4 Go to Computer Management, Services and Applications, Internet Information Services, Web service Extensions and verify that ASP.NET (32-bit) is set to: **Allowed**

XML Troubleshooting

This section contains tips to help you solve any errors that may occur due to problems with your XML syntax.

All XML Elements Must Have a Closing Tag

In XML, all elements must have a closing tag, like this:

```
<p>This is a paragraph</p>
<p>This is another paragraph</p>
```

XML Tags Are Case Sensitive

With XML, the tag <Letter> is different from the tag <letter>.

Opening and closing tags must therefore be written using the same case:

```
<Message>This is incorrect</message>
<message>This is correct</message>
```

All XML Elements Must Be Properly Nested

In XML all elements must be properly nested within each other like this:

```
<b><i>This text is bold and italicized</i></b>
```

All XML Documents Must Have a Root Element

All XML documents must contain a single tag pair to define a root element.

All other elements must be within this root element.

All elements can have sub-elements (child elements). Sub-elements must be correctly nested within their parent element:

```
<root>
  <child>
    <subchild>.....</subchild>
  </child>
</root>
```

Attribute Values Must Always Be Quoted

With XML, it is not permitted to omit quotation marks around attribute values.

XML elements can have attributes in name/value pairs just like in HTML. In XML the attribute value must always be quoted as shown in the following example:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<note date="12/11/2002">
  <to>Tove</to>
  <from>Jani</from>
</note>
```

White space Is Preserved Using XML

With XML, the white space in a document is not truncated.

Comments in XML

The syntax for writing comments in XML is as follows:

```
<!-- This is a comment -->
```

Configuring Session Management

Session Management is an integral part of SmarTeam's Core Services. In the System Configuration Service, you can define the parameters of the Session Management. The End Time of a Session parameter defines the length of idle time of a session. After this time has passed, the Session Management automatically closes the session. This feature reduces workload on the system and subsequently releases licenses for other users. This is useful in cases where some users leave their SmarTeam applications open when they are not at their desks.

To configure the end-time of a System Configuration session:

- 1 Open the file `smarteam.std.sessionManagement.service.host.exe.config` located in the `<SmarTeam Home directory>\bin` directory.
- 2 Update the value of the `<expiration>` tag. The value is in minutes.
- 3 Save the file.
- 4 Restart the Session Management Service.

Chapter 10: Vault Redundancy Core and Flow Setup on MSCS 2003

Clusters

What is a Computer Cluster?

A computer cluster is a group of loosely coupled computers that work together closely so that in many respects it can be viewed as though it were a single computer. Clusters are usually deployed to improve speed and/or reliability over that provided by a single computer, while they are typically much more cost-effective than single computers of comparable speed or reliability.

High-availability (HA) clusters (supported by SmarTeam): Clusters that are implemented primarily for the purpose of improving the availability of services which the cluster provides. They operate by having redundant nodes, which are then used to provide service when system components fail. The most common size for an HA cluster is two nodes, which is the minimum required to provide redundancy. HA cluster implementations attempt to manage the redundancy inherent in a cluster to eliminate single points of failure.

What is Microsoft Cluster Server?

Microsoft Cluster Server (MSCS) is software designed to enable servers to work together as one machine to provide failover and increased availability of applications. For the remainder of this document, Microsoft Cluster Server will be referred to as MSCS.

Installing SmarTeam Vault Redundancy, Core and Flow on MSCS enables the system to detect a failure and automatically switch to the redundant Vault.

Note: All the documentation mentioned in this document, unless specified otherwise, is available on the SmarTeam Documentation CD.

Setting up SmarTeam Vault Server on MSCS

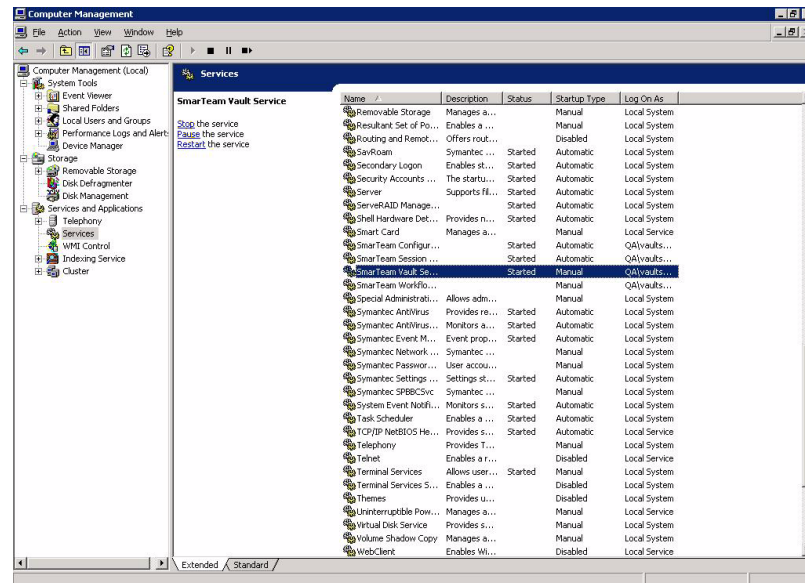
This section provides a step-by-step description of the process for setting up SmarTeam Vault Server on MSCS.

To set up SmarTeam Vault Server on MSCS:

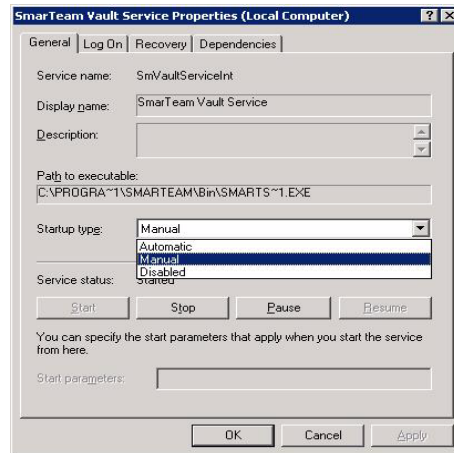
- 1 Install SmarTeam Vault server on each of the servers in the cluster.

Note: Define the same Domain Controller (DC) for all machines and create the relevant vault group, accordingly.

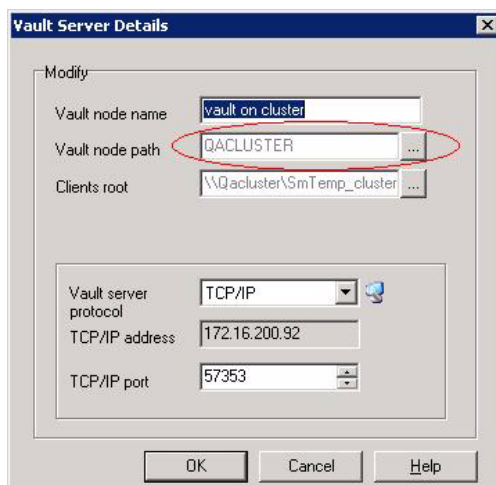
- 2 In the Computer Management Services window, double-click **SmarTeam Service Vault** in the Name field.



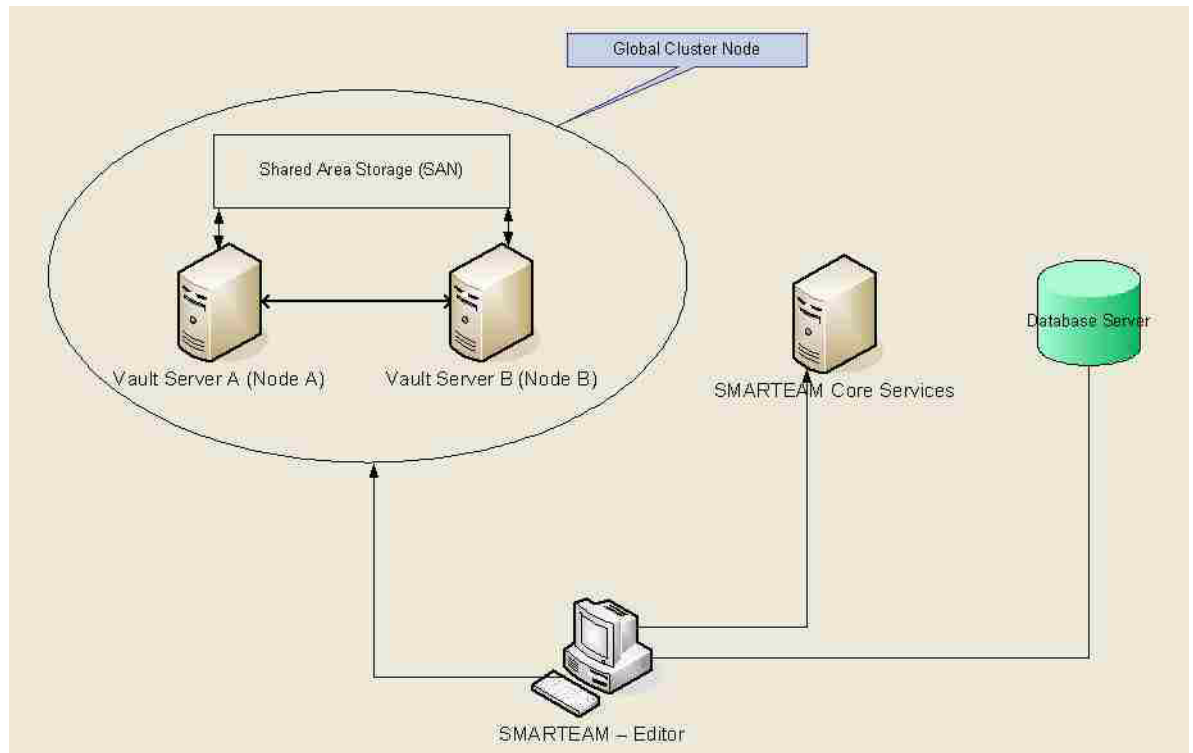
- 3 In the SmarTeam Vault Service Properties dialog box Startup type field:
 - Select **Manual** from the drop-down combo list.
 - Click **OK**.



- 4 Create a shared directory on each of the servers (nodes) and give it the same name on both machines (these directories are used as Temp directories for the vault work).
- 5 On each server (node) define the Global Cluster Name (using the Vault Server Setup) as the Vault Node Path , such as **QACLUSTER** in the network.



- 6 Create a shared directory for SmarTeam Vault on the Shared Area Network (SAN). Use this directory as the actual Vault containing the **CheckIn**, **Release** and **Obsolete** directories. See MSCS System Layout (below) for a visual illustration of this setup.

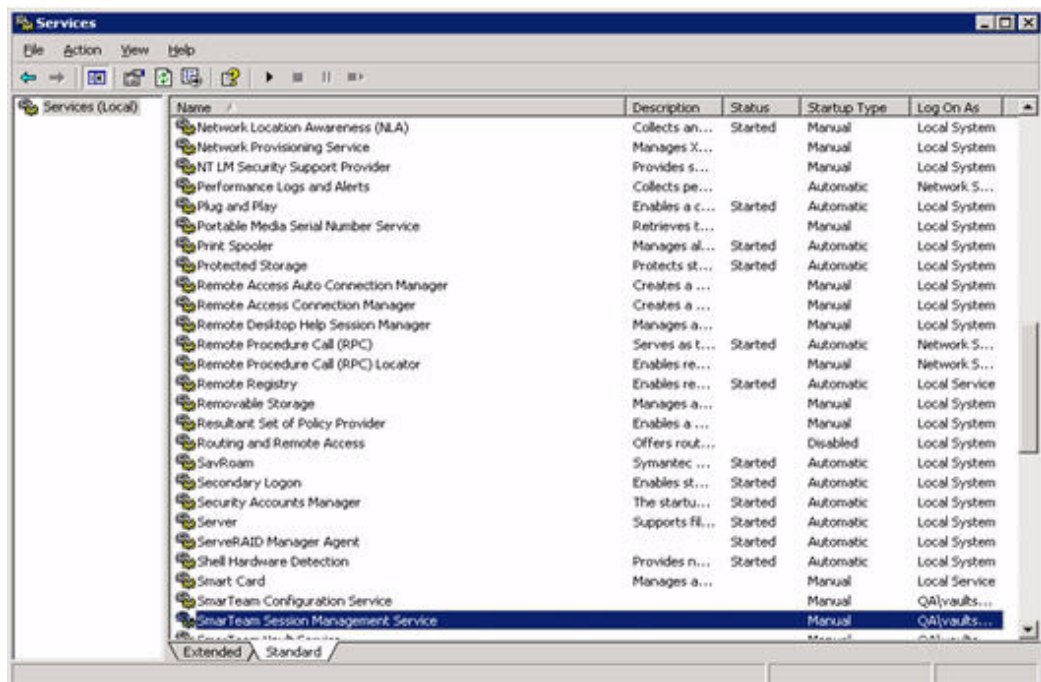


Setting up SmarTeam Core Services on MSCS

This section provides a step-by-step description of the process for setting up SmarTeam Core Services on MSCS.

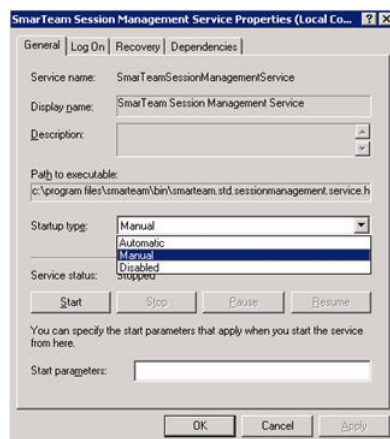
To set up SmarTeam Core Services on MSCS:

- 1 Install SmarTeam Core Services on each of the servers in the cluster.
- 2 Set SmarTeam Session Management Service and SmarTeam Configuration Service services to manual:
In the Computer Management Services window, double-click **SmarTeam Session Management Service** in the Name field.



3 In the SmarTeam Session Management Service Properties dialog box, Startup type field:

- Select **Manual** from the drop-down combo list.
- Click **OK**.



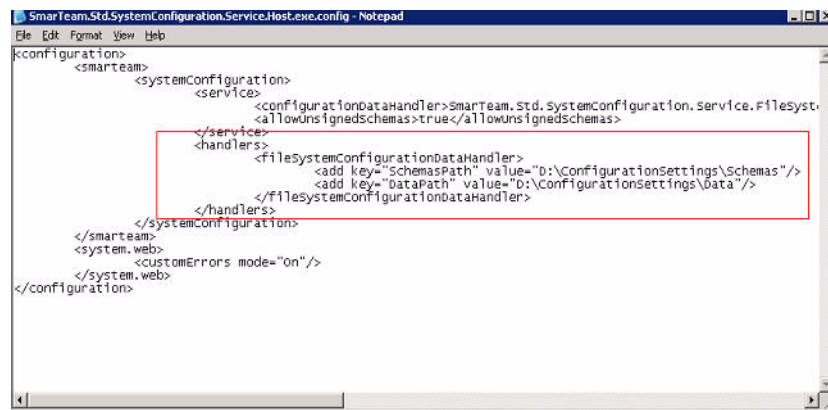
4 Copy **ConfigurationSettings** folder located in <SmarTeam Home Directory> to SAN.

5 Set -R -W permission for all SmarTeam users on ConfigurationSettings folder.

6 From the <SmarTeam Home Directory>\Bin:

- Open the **SmarTeam.Std.SystemConfiguration.Service.Host.exe.config** file.
(SmarTeam recommends that you back up this file before performing any updates.)
- In the tag <handler> update the **SchemasPath** key and **DataPath** key to the SAN location.

Note: The SAN must be defined as a partition driver.

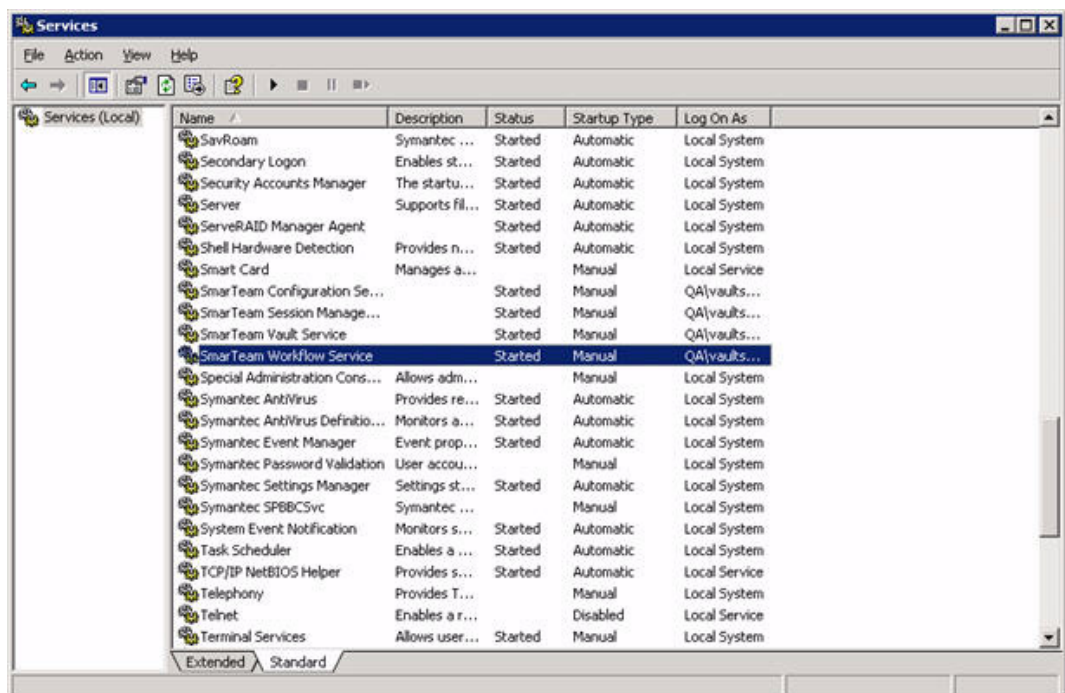


Setting up SmarTeam Workflow Services on MSCS

This section provides a step-by-step description of the process for setting up SmarTeam Workflow Services on MSCS.

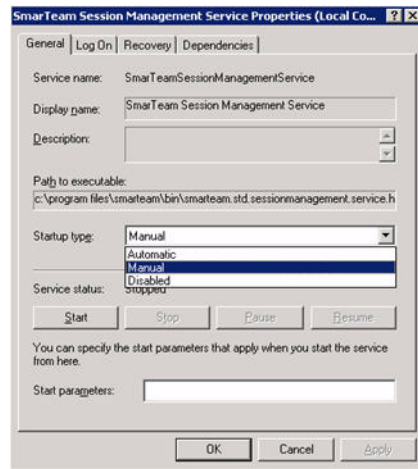
To set up SmarTeam Workflow Services on MSCS:

- 1 Install SmarTeam Workflow Services on each of the servers in the cluster.
- 2 Set SmarTeam Workflow Service to manual:
In the Computer Management Services window, double-click **SmarTeam Workflow Service** in the Name field.



- 3 In the SmarTeam Workflow Service Properties dialog box, Startup type field:

- Select **Manual** from the drop-down combo list.
- Click **OK**.



Chapter 11: System Configuration Keys

For a list of keys available in the system, see the attached V5R20 System Configuration Keys Excel file.