



ENOVIA SmarTeam

SmarTeam – Job Server Administration Guide

© Dassault Systèmes, 2009, 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: JBS-A1-200110

Contents

Chapter 1: Overview	1
Introduction	1
Multi-site Support for Job Server	2
Related Documentation	2
Internet Site	2
Chapter 2: Installation	3
Chapter 3: Setup	4
Adding Mechanisms to Database	4
Setting Sequences for Administration Classes	5
Define and Update Profile Cards	6
Chapter 4: Configuration	7
Administration Classes	7
Job Server Configuration Options in Admin Settings	8
Chapter 5: System Jobs	9
Job Objects	9
Mail Job	10
Mail Job Script	10
Admin Setting for Mail Job	10
Executable Definition	10
Attribute Mapping	10
System Configuration Keys of SMTP for Mail Job	10
Print Job	11
Print Job Script	11
Admin Setting for Print Job	11
Executable Definition	11
Attribute Mapping	12
Attribute Mapping Example	12
Silent Release	13
Silent Release Script	13
Admin Setting for Silent Release Job	14
Executable Definition	14
Attribute Mapping	14
Attribute Mapping Example	15

Prepare a Template File with Mapping	16
Custom Jobs	17
Admin Setting for Custom Job	17
Executable Definition	17
Attribute Mapping	18
Attribute Mapping Example	18
Creating a Custom Job	18
Chapter 6: Operating the Job Server	20
How to Use the Job Server?	20
Set up the Job Server Parameters	21
Executable Job Tab	21
Environment Tab	21
Job Status Information	23
Tracing the Job Server	23
Operational Level	23
Debugging Level	23
Appendix A: Example of Admin Settings for Job Server	25

Chapter 1: Overview

Introduction

This guide provides the necessary information for SmarTeam system administrators to implement SmarTeam – Job Server (JBS).

SmarTeam – Job Server software is a server-side application that handles central SmarTeam activities. The Job Server configuration frees SmarTeam client applications from running these activities and reduces the need to have an application installed on each client to process them.

Note: Although Job Server is installed on a server the Job Server application is a Microsoft® Windows application.

The SmarTeam – Job Server supports the following operations:

- **Mail Process:** Using a standard SMTP protocol, the Job Server automatically routes messages for required system notifications to administrators and end users. A Mail job can be hooked after SmarTeam commands.
- **Print Process:** Using the configured viewer of the document (by file type), the Job Server prints files as requested by the user. The printer needs to be accessed by the computer running Job Server software. In addition, metadata information from SmarTeam database records can be embedded automatically in the watermark of the printout in the header and/or footer in SmarTeam Viewer (Oracle AutoVue in SmarTeam – Editor).
- **Silent Release Process:** Provides additional functionality that occurs after a lifecycle operation is performed on a document. Silent Release updates a vaulted file by synchronizing mapped attributes from SmarTeam metadata into released files without the need to perform a lifecycle check out operation or creating a new version of the file.
- **Custom Job:** Job Server infrastructure enables the creation of custom jobs executing additional functionality required by the business practice. Admin Settings are used to write a class library and configure a new executable.

Multi-site Support for Job Server

Implementation that involves SmarTeam Multi-site requires installation of Job Server for each site.

Although there is a replication mechanism between different sites, on each site the Job Server processes only jobs that originate at that SmarTeam site.

Related Documentation

The following documents are referred to in this guide and are available on the ENOVIA SmarTeam Documentation CD.

Document	Remarks
SmarTeam – Job Server Installation Guide	Describes how to install SmarTeam – Job Server.
SmarTeam – Regulatory Compliance Administration Guide	Describes how to implement SmarTeam – Regulatory Compliance Framework.

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: Installation

Refer to SmarTeam – Job Server Installation Guide.

Chapter 3: Setup

After the SmarTeam – Job Server software installation, you must update the database structure in the SmarTeam Data Model Designer before you can use the SmarTeam – Job Server product.

To set up SmarTeam – Job Server, you must perform the following operations:

- Update Database Structure by adding mechanisms to the database
- Set Class Sequences

Adding Mechanisms to Database

To work with the Job Server you must add the relevant mechanisms to your database using the SmarTeam Data Model Designer utility. This operation only adds the relevant mechanisms. It does not affect the SmarTeam Data Model structure.

IMPORTANT! It is highly recommended that you perform a backup operation of the selected database before performing this update procedure.

To update your selected database:

- 1** From the taskbar Start button, select **Programs > SmarTeam > Administrative Tools > SmarTeam Data Model Designer** to launch the SmarTeam Data Model Designer.
- 2** In the Data Model Designer window, from the main menu select **File > Modify Database Structure** to display the Available Databases window.
- 3** In the Available Databases window, select the required database and click **OK**.
A SmarTeam message window appears advising you to perform a backup of the selected database before proceeding to update the selected database.
- 4** Verify you back up your database and click **Yes**.
- 5** The Wizard Login [Database Name] window appears. Type the applicable Username and Password for the selected database and then click **OK** to continue.
After a successful login, the SmarTeam Data Model Designer window displays the selected database.
- 6** The following new mechanisms are available in the SmarTeam Data Model Designer utility:
 - a** **Job Server**

- b Admin** (may be already selected in the database)
- 7** Select **Mechanisms**.
- 8** Click **Create** to update the database structure according to your selections in the previous steps.
- 9** When you receive a system message that the database has been successfully modified, click **OK** to save changes and exit the SmarTeam Data Model Designer.

Setting Sequences for Administration Classes

Because Admin Mechanism is needed, you must set unique values for the identifiers of the objects of these classes. This can be done by setting sequences.

Table 1 contains the list of classes and an example of Sequence definitions.

- 1** From the Admin Console, launch the SmarTeam Form Designer utility.
- 2** Open the Attribute Profile Card of the relevant class for each class mentioned in Table 1 and repeat the following steps:
 - a** Select control of the TDMX_ID attribute.
 - b** Double-click the control to open the Properties window.
 - c** Click on the Mask Name and click **Browse** in this field.
 - d** If the sequence does not already appear in the list, click **New** and create a new sequence.
 - e** Select the sequence and click Select to link it to the specific field.

Notes:

- These settings are used both by the SmarTeam – Regulatory Compliance Framework and SmarTeam – Job Server.
- For further information on the Administration Classes, see [Administration Classes](#).
- For further information on the Form Designer refer to the SmarTeam – Editor Administrator Guide.

Table 1: List of classes that require defined sequence values

Class Name	Sequence Name	Sequence Description	Sequence Pattern	Sequence Value
Admin	Admin	Admin	Admin-99999999	Admin-00000000
Admin Folder	Admin Folder	Admin Folder	Admin Folder-99999999	Admin Folder-00000000
Settings	Settings	Settings	Settings-99999999	Setting-00000000
Admin Set-tings	Admin Settings	Admin Settings	Admin Settings-99999999	Admin Settings-00000000
User Set-tings	User Settings	User Settings	User Settings-99999999	User Settings-00000000
Queues	Queues	Queues	Queues-99999999	Queues-00000000
Mail	Mail	Mail	Mail-99999999	Mail-00000000
Operation	Operation	Operation	Operation-99999999	Operation-00000000
Print	Print	Print	Print-99999999	Print-00000000

Define and Update Profile Cards

Using the Form Designer utility, update the Profile Cards of Admin Classes.

Chapter 4: Configuration

Administration Classes

Job Server settings are stored in SmarTeam Admin Classes, which provides better security for the settings, because they are managed and controlled by the database system and SmarTeam – Editor authorizations and security.

Configuration settings are stored in the Admin super class that consists of subclasses and leaf classes.

Level	Class Name	Description
1	Admin	Super Class
2	Settings	System and User Settings
3	Admin Setting	Global settings for an application/database
3	User Setting	User settings for a specific user
2	Messages	System messages (error, info, questions)
2	Queue	Queues for Job Server
3	Mail	Mail Queue for mail jobs
3	Print	Print Queue for print jobs
3	Operation	Operation Queue for Silent Release jobs

The Job Server uses the Queue class to handle Jobs with Queue mechanism. When a new Job is required, a new object of the relevant class is created and picked up for processing during Job Server runtime. The following queues are implemented:

- **Mail** – When a SmarTeam application needs to send an email message through the Job Server, it creates a job in the Mail queue. The job includes processing information and mail message information, such as the sender, receiver and message text.
- **Print** – When a SmarTeam application needs to print a file through the Job Server it creates a job in the Print queue. The job includes processing information and the SmarTeam object ID and class ID of the record for the file that needs to be printed.

- **Operation** – When a SmarTeam application needs to use the Job Server for a Silent Release operation, it creates a job in the Operation queue. The job includes processing information and the name of the authoring application that is supported in the Job Server. Job Server supports Microsoft® Word, Microsoft® Excel and Custom Jobs.

Note: Adding an Admin Setting does not influence system behavior. However, deleting Admin Settings can cause system failure. Therefore, it is highly recommended that only the SmarTeam System Administrator has update rights for Admin Settings.

Job Server Configuration Options in Admin Settings

■ Admin Setting Logic

- **ID Attribute** – Text attribute that fills based on a sequence, see Setting Sequences for Administration Classes.

If a sequence is not defined, the script file creates an ID created in the following format:

"JOB-" + Today Date in format: YYYYMMDDHHmmss (year, month, day, hour, minute, seconds).

Example: JOB-2009Aug1201271582

- The combination Subject/Section should be unique for each Admin Setting. SmarTeam generates an error message if two settings with the same identifiers are found.
- **Section Attribute** – Group several settings.
- **Subject Attribute** – Defer settings with the same Section attribute.
- Each Admin Setting can have additional information saved in attributes named – First Value, Second Value, Third Value, Fourth Value and Long Value.
- Description fields are a free text fields meant to assist the administrator to describe the usage of the Admin Setting. It has no impact on the system.

■ Admin Setting for Job Server

Type values for the following fields on each Profile Card:

- **Admin ID** – Sequence number. Refer definition in [Setting Sequences for Administration Classes](#).
- **Section** – This varies according to the following definitions:
 - **Executable** – Main definition. In this case, the value is always JOB SERVER.
 - **Attribute Mapping** – Detailed definitions for each Executable. May have different values, such as Microsoft® Word, Microsoft® Excel, and Print.
- **Subject** – This varies according to the following definitions:
 - **Executable** – Name of the Executable Job in the Job Server.
 - **Attribute Mappings** – Each executable for the content of this field.
- **Description** – Free text field.
- **First – Fourth Values** – Each executable for the description of these fields.
- **Long Value** – Not in use.

Chapter 5: System Jobs

All Job Server functions are located in the JobServer.ebs file. Functions exist: CreateMailJob, CreatePrintJob, CreateJobBasedOnFileType.

Job Objects

- 1** When a SmarTeam application needs to create a new job, the job is created in the appropriate Queue class as a new record. Based on system definitions, the job includes processing information and job-related details. The processing information includes:
 - a** Executing Server Name – Executable Name, such as Mail, Print, Microsoft® Word, CATIA, and SolidWorks.
 - b** Object ID and Class ID – Describes the Object in which the operation is performed (not in Mail).
Example: Document with file to Print or Release.
 - c** Not Processed – Flag indicating whether the job is already processed. When the job is created, it is always set to **True** (job is not yet processed by the Job Server).
 - d** Error Occurred – Flag indicating whether the job ended in an error. When the job is created, it is always set to the default value **False**, which means that an error has not occurred.
 - e** Job Details – Describes the job process. When the job is created, the job details field is empty. After processing the job, the Job Server completes this field with information regarding the success or failure of the job.
 - f** Priority – Describes the urgency of the job. The Job Server sorts the jobs to process by this field, the higher the value, the higher the job's priority.
- 2** The Administrator can define Query by Attributes to simplify the search for maintaining Job Server jobs, such as defining a query for all jobs where an error occurred.
- 3** The Job Server runs constantly on the designated server. Applications running on the Job Server are checked by the SmarTeam System, when a new job for one of these applications is added to the queue, the Job Server proceeds to execute it.
- 4** After processing the job, depending on the definition of the admin setting of Executable Definition – Second Value:
 - a** If a job finishes successfully, the Job Server does one of the following:
 - If the Second Value is **DELETE** – Deletes the job record from the queue.

- If the Second Value is **LEAVE** – Updates the job record with a success message. The Not Processed and the Error Occurred attributes are updated.
- b** If the job ends in an error, the Job Server updates the job record with a failure message and updates the Error Occurred flag accordingly.

Mail Job

Mail Job Script

The CreateMailJob script function creates a Mail job. The script creates a job depending on the operation to which the script is hooked. For example, to notify a user on a document's release, the following occurs:

- 1** A Script is hooked to a New Release operation in the After event of Document class.
- 2** New Release operation is performed on the SmarTeam document.
- 3** The New Release creates a Mail job in the Mail class (Table Name: TDMX_MAIL).
- 4** When the Job Server runs the next time, the Mail job is executed and mail is sent.

Admin Setting for Mail Job

Add the following Admin Setting for Job Server as follows.

Executable Definition

- Admin ID – Sequence number
- Section – JOB SERVER
- Subject – MAIL
- Description – Free text field
- First Value – Internal
- Second Value – DELETE or LEAVE

Attribute Mapping

Not Required

System Configuration Keys of SMTP for Mail Job

There must be an SMTP server running on the Job Server, such as Microsoft® Windows 2000 SMTP Server. To use the SMTP Server, you must configure two system configuration keys. Do the following:

- 1** Open the System Configuration Editor and define the keys:
 - a** Search for Messages.SMTPServer key
 - b** Click **Add** and complete the following values:
 - Override level = Domain
 - Value = <MailServer name>
 - c** Search for Messages.SMTPServerPort key

- d** Specify the port used for the SMTP server (default is 25)
- 2** Define Mail Script:
 - a** From SmarTeam Script folder, open JobServer.ebs.
 - b** Customize these email parameters:
 - TDMX_EMAIL_TO
 - TDMX_MESSAGE
 - TDMX_MESSAGE_SUBJECT
 - TDMX_EMAIL_FROM

Print Job

The Job Server utilizes SmarTeam – Editor's embedded viewer-printing capabilities to print documents via the viewer engine. When using the SmarTeam Viewer, metadata attributes can be automatically placed in the header, footer or watermark on the printout.

Notes:

- SmarTeam Viewer or any other viewer associated with the document's file format to be printed must be installed on the server.
- The print job occurs on the default printer of the job server machine.

Print Job Script

The CreatePrintJob script function creates a Print job. The script creates a job depending on the operation to which the script is hooked. For example, to print a document in a local printer after it is checked in:

- 1** Script is hooked to a Check in operation in the After event of Document class.
- 2** Check in operation is performed on the SmarTeam document.
- 3** Check in creates a Print job in the Print class (Table Name: TDMX_PRINT).
- 4** When the Job Server runs the next time, the Print job is executed and mail is sent.

Note: If a file is not attached to the object, the Print job is not executed successfully.

Admin Setting for Print Job

Add Admin Setting for Job Server as follows.

Executable Definition

- Admin ID – Sequence number
- Section – JOB SERVER
- Subject – PRINT
- Description – Free text field
- First Value – Internal
- Second Value – DELETE or LEAVE

Attribute Mapping

Mapping header, footer and watermark attributes are managed in Admin Settings. Mapping is used only with the SmarTeam Viewer. Other viewers used to view other file formats cannot use this feature.

- If the Subject is PrintHeadersFooter_Server, the Section value can be one of the following values:
CenterFooter, LeftFooter, RightFooter, CenterHeader, LeftHeader, RightHeader.
- If the Subject is PrintWaterMark_Server, the Section value must be **Text**.
- First Value – The text that is part of the final string.
- Second Value – The name of the SmarTeam attribute.
- Third Value – The print date and time (SmarTeam Viewer formatting parameters).
- Fourth Value – The date format, when required, for printing the date.

Attribute Mapping Example

Example for Footer describes Attribute Mapping for PrintHeadersFooters_Server (Section), for the LeftFooter area (Subject). This mapping inserts 'Status =' (First Value) and the value that is mapped by STATE (Second Value). STATE is the SmarTeam internal attribute name.

Attribute Name	Example of Footer	Example of Watermark
Admin ID	ADM-00000151	ADM-00000152
Section	PrintHeadersFooters_Server	PrintWaterMark_Server
Subject	LeftFooter	Text
Description		
First Value	Description =	Approval Date =
Second Value	TDM_DESPCRPTION	APPROVAL_DATE
Third Value		printed %D-%M-%Y at %H:%U
Fourth Value		dd-MMM-YYYY
Long Value		

Silent Release

The Silent Release process provides additional functionality that occurs after a lifecycle operation is performed on a document. The Silent Release updates a vaulted file by synchronizing mapped attributes from SmarTeam metadata into released files without the need to perform a Check out lifecycle operation or create a new version of the file.

A Silent Release operation takes place after the actual release operation on a document. The server copies the file from the vault server into a temporary directory on the server. It then opens the file with its authoring system; retrieves approval information from the metadata information stored in the SmarTeam database and writes this information to predefined, mapped properties on the file. The file is then saved and copied back to the vault.

This is usually required after the file has been released to the vault and additional information, such as Electronic Signature, is added to the metadata, which needs to be embedded in the attached file. The function of embedding information in the file requires installing the authoring system on the computer. The Job Server configuration reduces the need to install the authoring system on all clients.

During a company's Approve and Release production process, there may be some time between approving (Electronic Signature) a document and releasing it to the vault. During this time, performing changes to the document are not usually allowed. To prevent a user from performing changes to the document, which include approval information (approver username and time of approving), a Silent Release operation is required.

To prevent a manual release of the document, apply one of the following:

- In Admin Options > Workflow Options, select Allow lifecycle operations only through Workflow.

OR

- For classes that use Electronic Signatures, remove permissions for Release for all users except the one user who activated the Job Server.

SmarTeam provides several Authoring System integrations with Job Server software, such as Microsoft® Word, Microsoft® Excel and several CAD applications. For additional integration, custom jobs can be developed and implemented, see Custom Jobs.

Note: If you use the Job Server, Microsoft® Word and Microsoft® Excel executables, download and register Microsoft® Office XP primary Interop assemblies (PIAs). For additional information, see: <http://www.microsoft.com>.

Silent Release Script

The **CreateJobBasedOnFileType** script function creates a Release job. The script creates a job depending on the File Type. For example:

- 1 Script is hooked to a **Release** operation in the **After** event of a Document class.
- 2 **Release** operation is performed on the SmarTeam document that contains a file.
- 3 Supported integrations: Microsoft® Word, Microsoft® Excel, CATIA, SolidWorks and AutoCAD.
- 4 The **Release** creates a Release job in the Operation class (Table Name: TDMX_OPERATION).

- 5 When the Job Server runs the next time, the Print job is executed and mail is sent.

Note: If a file is not attached to the object, the Release job is not executed successfully.

Admin Setting for Silent Release Job

Add Admin Setting for Job Server as follows:

Executable Definition

- Admin ID – Sequence number
- Section – **JOB SERVER**
- Subject – Name of the Executable Job in the Job Server, e.g. **Microsoft® Word, Microsoft® Excel, CATIA, SolidWorks or AutoCAD**
- Description – Free text field
- First Value – **Internal**
- Second Value – **DELETE** or **LEAVE**

Attribute Mapping

Note: Attribute Mapping is needed for Microsoft® Word and Microsoft® Excel. For CAD Integration, only Title Block can be synchronized via mapping.

- Section – Name of the Executable Job in the Job Server, e.g. Microsoft® Word.
- Subject – <class name>_mapping_x, where class name is the name of the class of the object in SmarTeam and X is a number e.g. Document_mapping_1.
- When mapping Regular SmarTeam Attribute:
 - First Value – Microsoft® Word / Excel field name as defined in the Microsoft® Word / Excel file.
The field name is case sensitive!
 - Second Value – SmarTeam attribute name.
 - Third Value – If the mapped field is a date then the value is date format. If not, the value should be empty.
 - Fourth Value – Indicates the Microsoft® Word field type. Values can be:
 - **B** for Bookmark (Microsoft® Word Only)
 - **R** for Range (Microsoft® Excel Only)
 - **P** for Property (Microsoft® Word and Microsoft® Excel)
 - **C** for Custom Property (Microsoft® Word and Microsoft® Excel)
- When mapping SmarTeam Electronic Signature Attribute (if E is used in the Fourth Value):
 - Signatures are mapped in a table
 - First Value – <Word Table Name>.<column name> to map the Signature
The field name is case sensitive!
 - Second Value – TDMX Signature attribute name, see list below
 - Third Value – If the mapped field is a date then the value is date format. If not, the value should be empty

- Fourth Value – E for Electronic Signature
- TDMX Signature attribute names:
 - TDMX_NODE_OBJECT_ID – ID of the node on which this Signature is performed
 - TDMX_NODE_NAME – Name of the node on which this Signature is performed
 - TDMX_ES_OBJECT_ID – Object ID of the signed object
 - TDMX_ES_CLASS_ID – Class ID of the signed object
 - TDMX_SIGNEE_NAME – First and Last name of the user who signed at the moment the object was signed (if the username changes afterwards, this field is not affected)
 - TDMX_SIGNEE_OBJECT_ID – Object ID of the user who signed
 - TDMX_DATE_TIME – Date Time of the Signature
 - TDMX_MEANING_OF_SIGNATURE – Meaning of Signature

Attribute Mapping Example

Example for Property describes Attribute Mapping for Microsoft® Word (section), for the **Document** class (Subject). Field name is Title, SmarTeam Attribute is TDM_DESCRIPTION and field type is **P** for Property (Fourth Value).

Example for Custom Property Attribute Mapping for Microsoft® Word (section), for the **Document** class (Subject). Field name is Status, SmarTeam Attribute is STATE and field type is **C** for Custom (Fourth Value).

Example for Bookmark describes Attribute Mapping for Microsoft® Word (section), for the **Document** class (Subject). Field name is **Date completed**, SmarTeam Attribute is **CN_FINISH_DATE** and field type is **B** for Bookmark (Fourth Value).

Attribute Name	Example of Property	Example of Custom Property	Example of Bookmark
Admin ID	ADM-00000161	ADM-00000162	ADM-00000163
Section	MICROSOFT WORD	MICROSOFT WORD	MICROSOFT WORD
Subject	Document_Mapping_1	Document_Mapping_2	Document_Mapping_3
Description			
First Value	Title	Status	Date completed
Second Value	TDM_DESCRIPTION	STATE	CN_FINISH_DATE
Third Value			mm/dd/yyyy hh:nn
Fourth Value	P	C	B

Example for Electronic Signature describes three Attribute Mappings of field type is **E** for Electronic Signature (Fourth Value) for Microsoft® Excel (section), for the **Document** class (Subject).

For field name, **Signee Role**, SmarTeam Attribute is **TDMX_NODE_NAME**.

For field name, **Signee Name**, SmarTeam Attribute is **TDMX_SIGNEE_NAME**.

For field name, **Sign Date**, SmarTeam Attribute is **TDMX_DATE_TIME**.

	Example of Electronic Signature		
Attribute Name	First Mapping	Second Mapping	Third Mapping
Admin ID	ADM-00000171	ADM-00000172	ADM-00000173
Section	MICROSOFT EXCEL	MICROSOFT EXCEL	MICROSOFT EXCEL
Subject	Document_Mapping_11	Document_Mapping_12	Document_Mapping_13
Description			
First Value	Signee Role	Signee Name	Sign Date
Second Value	TDMX_NODE_NAME	TDMX_SIGNEE_NAME	TDMX_DATE_TIME
Third Value			mm/dd/yyyy hh:nn
Fourth Value	E	E	E

The example can be changed from Microsoft® Excel to Microsoft® Word by implementing the following changes:

- Section = MICROSOFT WORD
- First Value = <Bookmark_Name>.<Header_Name>

For example: ES.Signee Role

Prepare a Template File with Mapping

Prepare a template file for Microsoft® Word and Microsoft® Excel for mapping with the following definitions:

- 1 From all Admin Settings that are created for Microsoft® Word / Microsoft® Excel, copy the First and Fourth values.
- 2 If the Fourth value is **B** for Bookmark:
 - a Add Bookmark with the First Value.
- 3 If the Fourth value is **R** for Range:
 - a Define Name (for Microsoft® Excel 2007) or Name (for Microsoft® Excel 2003) with the First Value.
- 4 If the Fourth value is **P** for Property:
 - a File Property already exists.
 - b In Microsoft® Word you can insert the File Property.
- 5 If the Fourth value is **C** for Custom Property:
 - a Create Custom Property with the First Value.
 - b In Microsoft® Word you can insert the Custom Property.
- 6 If the Fourth value is **E** for Electronic Signature:
 - a In Microsoft® Word:
 - i. Add a table with columns for Electronic Signature.
 - ii. Add header names.

- iii. Select the table and add a new Bookmark.
- iv. First Value should be: <Bookmark_Name>.<Header_Name>)
- b** In Microsoft® Excel:
 - i. Add header names in a row (Recommended).
 - ii. The row below the headers should include names corresponding to the headers:
 - For Microsoft® Excel 2003 – Create a **Name**.
 - For Microsoft® Excel 2007 – Create a **Define Name**.
 - iii. First Value should be <Defined Name>.
- 7** Click **Add** and save the document.
- 8** Execute the Job Server to see the Microsoft® Word document with a generated electronic signature.

Custom Jobs

The Job Server enables custom operations that can be done on the SmarTeam database, such as Silent Release of additional file formats. Additional file types can be added if a Silent Release server application is available as part of the Job Server. To handle this file an executable implementation of interface ICustomJob must be created.

Queue can be handled in a new leaf class that you add as a child of the Queue abstract class in the DMD.

Queue can also be handled in the existing leaf classes of the Queue.

The user can add any required custom job, e.g., to support PowerPoint files. To do this, the user must define new Admin Settings.

Admin Setting for Custom Job

Add Admin Setting for Job Server as follows.

Executable Definition

- Admin ID – Sequence number.
- Section – **JOB SERVER**.
- Subject – Name of the Executable Job in the Job Server, e.g. **Microsoft® PowerPoint**.
- Description – Free text field.
- First Value – Customer Job Name. The Custom Job has a DLL name and a Class name. For example: **PowerPoint.Main**, where AutoCAD is the DLL name and Main is the name of the DLL class, which contains the two required functions (IsAvailable and ExecuteJob)
- Second Value – This is the Archive Option, which is the action after the Queue item is processed with no error. Possible values for the Archive Option are **DELETE** or **LEAVE**.

Attribute Mapping

Not Required.

Attribute Mapping Example

Example for AutoCAD defines a new Job Server Executable (Section), which will enable supporting AutoCAD files. The name under which the Executable is seen in the Job Server Executable list is AutoCAD (Subject). The name of the user's DLL for this job is AutoCAD (the text written in the First Value before the dot), and the name of the class, which holds the two required functions (IsAvailable and ExecuteJob) is Main (the text after the dot in the First Value). These definitions also instruct the Job Server not to delete the jobs from SmarTeam after successful execution.

Attribute Name	Example for AutoCAD
Admin ID	ADM-00000181
Section	JOB SERVER
Subject	AutoCAD
Description	
First Value	AutoCAD.Main
Second Value	LEAVE

Creating a Custom Job

To create a custom job creates your own DLL with a main class containing these two mandatory functions:

- 1 bool IsAvailable (ref string ErrorInfo)
 - a IsAvailable – A function returns a Boolean variable (True/False) indicating if the current computer can run this type of job.

Example: The IsAvailable function in the Print class checks if the user has defined at least one printer on this computer. If the result of this function is true, the user checks the Job Item in the Job Server Executable list. If the result is false, the user is not able to mark the Job and a message box appears with the text from the ErrorInfo parameter.

 - b ErrorInfo – An output string parameter that describes the error if it occurs. A message box appears with the text from this parameter.
- 2 bool ExecuteJob (ref SmApplic.ISmObject ToDoJob, ref string ErrorInfo, ref SmUtil.SmLogFile LogFile, ref SmRecList.SmRecordList Parameters)
 - a ExecuteJob – A function returns a Boolean variable (True/False) indicating if the Job has been executed successfully. This function decides what happens to Jobs of this type. For example, add date of creation and name of creator to an AutoCAD file.
 - b ToDoJob – An input/output ISmObject on which the changes in the code are performed.
 - c ErrorInfo – An output string parameter that describes the error if it occurs. This error message appears in the Error Occurred field of the Job.

- d** LogFile – An input/output SmLogFile that is the Application's Log file.
- e** Parameters – An input/output SmRecordList of additional parameters to be used in the execution of the function. The content of the record list is currently the work directory of the Job Server Application.

Note: Additional functions can be added according to your requirements. Once a customer specific Assembly (DLL) has been created, the DLL should be stored in the same directory as the Job Server Executable.

Chapter 6: Operating the Job Server

How to Use the Job Server?

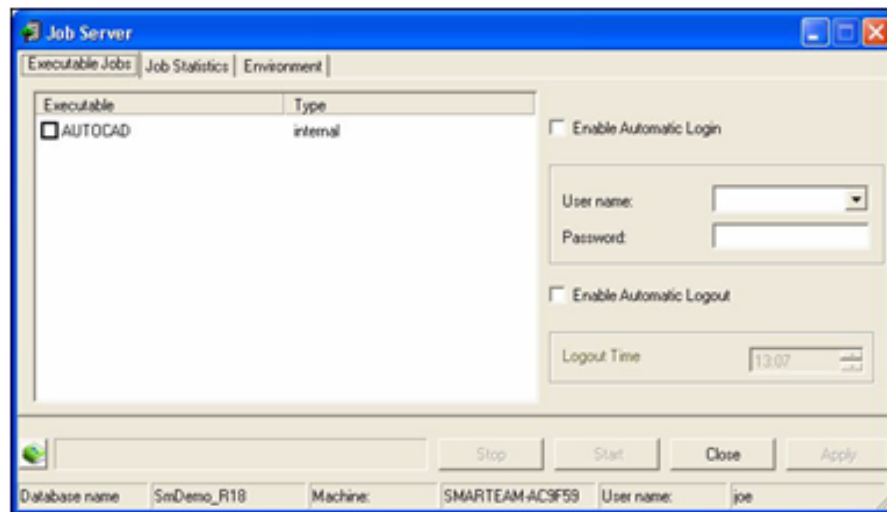
Job Server software runs as an application session. Several Job Server applications can be executed at the same time on several computers, handling different Job executables or handling the same Job executables in parallel. To define and set up jobs for a specific Job Server, the SmarTeam Administrator should use a dedicated tool.

IMPORTANT! Anti-Virus software's might identify the **SmarTeam.Std.Solutions.JobServer.Exe** as a mole and prevent mail jobs from being executed. You should define this Exe as safe in your Anti-Virus software settings.

To run the Job Server:

- 1 Browse to <SmarTeam>\Bin directory and run **SmarTeam.Std.Solutions.JobServer.Exe**.
- 2 Log in as SmarTeam Administrator.
When running the Job Server for the first time on the dedicated server, default settings are created on the designated server.
- 3 From the Executable Jobs tab, in the Job Server window, select one or more executables to run on the server.
- 4 Click **Start** to run the selected Jobs executables. At least one of the jobs must be selected.
 - a After the Job Server starts to run, it constantly checks queues for new jobs.
Only Jobs in the queue from the selected Jobs executables that are not yet processed will run.
 - b Jobs that need to be executed by applications run on the server. E.g., Mail jobs and Silent Release.
 - c While the Job Server is running, status information and statistics appear, see Job Status Information.
- 5 Click **Stop** to stop the Job Server. All running Jobs stop.
- 6 If at any stage you need to change the Setup options:
 - a Click **Stop** to stop the Job Server (if running).
 - b Make the necessary changes.
 - c Click **Apply** to save the changes.

- d Click **Start** to run the selected Jobs executables.
- 7 Click **Close** to exit the application.



Set up the Job Server Parameters

Job Server parameters are saved in the SmarTeam database as an Admin Setting where Section is Job Server Setting and Subject is Server Name.

Executable Job Tab

- 1 In the Job Server window a list of available Job executables appears. The available Job executables are defined in the database via Admin Settings (Section = Job Server).
- 2 You can schedule the Job Server to run automatically:
 - a Enable Automatic Login – Enable an automatic login to the Job Server application with selected username and password.
 - b Enable Automatic Logout – Ensure the Job Server stops executing jobs in defined time for a period of one hour.
This option is recommended to allocate a time for the System Administrator to back up the databases.
 - c Logout Time – Specify the time for Automatic Logout.

Note: When using Automatic Logout you can configure Automatic login as follows:

1. Define the user name and password in **Enable Automatic Login**.
2. In Windows scheduler, specify the time at which the **SmarTeam.Std.Solutions.JobServer.Exe** should be launched.
3. As Automatic Login is defined, the system will be launched automatically and will not ask for a user name or password.

Environment Tab

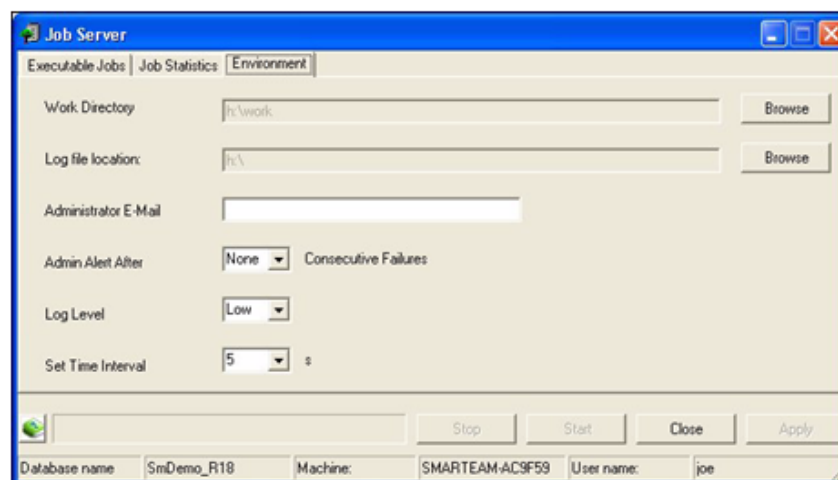
- 1 Work directory – Directory on the server for Silent Release and Print operation copy files. The default setting is <Job Server Application Location>\Work. The directory must be empty,

otherwise a prompt to remove all contents before running the Job Server appears. To select a directory other than default directory, click **Browse**.

- 2 Log file location – Directory on the server where the log file is created. The default setting is <Job Server Application Location>\SmartJobServer.log. To select a different directory for the log file, click **Browse**.
- 3 Administrator Email – Defines the email address for Job Server alerts emails. You can select or type a value.
- 4 Admin Alert After "X" Consecutive Failures – Defines the number of consecutive Job Server failures that trigger an automatic email (to: Administrator Email) notifying the number of failures.
- 5 Log Level – The level of log messages during the server's operation. Possible values:
 - a Low – Log messages are not recorded (Default)
 - b Medium – Important messages are logged
 - c High – All messages are logged

Note: For additional information about the log level, see Tracing the Job Server.

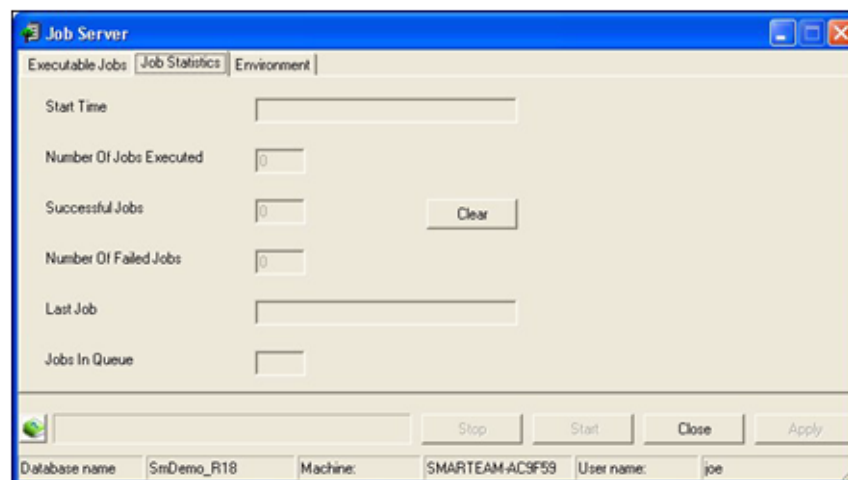
- 6 Set Time Interval – Monitors queues for new jobs. You can select or type a new value in seconds.



Job Status Information

While the Job Server is running, the following status information and statistics are displayed:

- Start Time – Starting time of the Job Server
- Number Of Jobs Executed – Total number of executed jobs, both successful and unsuccessful
- Successful Jobs – Number of successfully executed jobs
- Number Of Failed Jobs – Number of jobs that resulted in an error
- Last Job – Identifier of the last job
- Jobs In Queue – Number of jobs in the queue waiting to be launched



Tracing the Job Server

The next time the Job Server runs after defining the tracing, a log file is created. The logging starts immediately when the application is launched.

You can trace the behavior of the Job Server on an Operational level or on a Debugging level.

Operational Level

If the system is running on an operational level, the company might want to track the activities of the Job Server with minimal affect on the Job Server performance.

The definition of the log level is in the Job Server tool. There is no need for an environment variable.

Debugging Level

If the system is running on a debugging level, the company might want to track the activities of the Job Server with a more detailed log file.

The definition of the log level is defined in an environment variable. The administrator needs to add an environment variable where the Job Server runs as follows:

- Right-click My Computer and select Properties > Advanced > Environment Variables > System Variables > New.
- In the New System Variable window, type JOB_SERVER_LOG_LEVEL.
- Type variable value 1, 2, 3 or 4:
 - 1 – Log messages are not recorded (Default)
 - 2 – Important messages are logged
 - 3 – All messages are logged
 - 4 – System details are also displayed
- Click **OK**.

Appendix A: Example of Admin Settings for Job Server

Subject	Section	First Value	Second Value	Third Value	Fourth Value
Microsoft Word	Job Server	Internal	Leave		
Microsoft Excel	Job Server	Internal	Leave		
Print	Job Server	Internal	Delete		
Mail	Job Server	Internal	Leave		
CATIA	Job Server	Internal	Leave		
SolidWorks	Job Server	Internal	Leave		
Document_Mapping_1	MICROSOFT WORD	Title	CN_DESCRIPTION		P
Document_Mapping_2	MICROSOFT WORD	Status	STATE		C
Document_Mapping_3	MICROSOFT WORD	ES.APP_DATE	TDMX_DATE_TIME	dd-MMM-yyyy	E
Document_Mapping_4	MICROSOFT WORD	ES.APP_BY	TDMX_SIGNEE_NAME		E
Document_Mapping_5	MICROSOFT WORD	ES.Meaning of Signature	TDMX_MEANING_OF_SIGNATURE		E
Document_Mapping_6	MICROSOFT WORD	ES.Node Name	TDMX_NODE_NAME		E
Document_Mapping_1	MICROSOFT EXCEL	STATE	STATE		R
Document_Mapping_2	MICROSOFT EXCEL	ES.APP_DATE	TDMX_DATE_TIME	dd-MMM-yyyy	E
Document_Mapping_3	MICROSOFT EXCEL	ES.APP_BY	TDMX_SIGNEE_NAME		E
Document_Mapping_4	MICROSOFT EXCEL	ES.Meaning of Signature	TDMX_MEANING_OF_SIGNATURE		E
Document_Mapping_5	MICROSOFT EXCEL	ES.Node Name	TDMX_NODE_NAME		E

LeftFooter	PrintHeadersFooters_Server	Description =	TDM_DESCRIPTION		
LeftHeader	PrintHeadersFooters_Server	Approval Date =	APPROVAL_DATE	printed %D-%M-%Y at %H:%U	DD-MM-YYYY
RightFooter	PrintHeadersFooters_Server				
RightHeader	PrintHeadersFooters_Server				
Text	PrintWaterMark_Server	CON-TROLLED BY DMS			