Documentation Library

Data Explorer

Administrator Guide

Contents summary

WELCOME TO DATA EXPLORER FOR ADMINISTRATORS 7

ROAD MAP 7

SECTION 1 - GETTING USERS STARTED 8

WHAT IS DATA EXPLORER? 8
INSTALLATION FOR USERS OF DE 10

SECTION 2 - ADMINISTRATION 12

ARCHITECTURE 12
ADMINISTRATOR RESPONSIBILITIES 13
BEFORE YOU START 13
INSTALL SERVER COMPONENTS 14
LICENCES 14
RELATED PRODUCTS 14
REGISTER USERS 15
DOCUMENTATION 17

SECTION 3 – HOUSEKEEPING AND TROUBLESHOOTING 19

LOGGING 19
TRACING 20
CERTIFICATE MANAGEMENT 21
TROUBLESHOOTING 22
BEFORE YOU CONTACT AXES SUPPORT 24

SECTION 4 - CONFIGURATION 25

THE CONFIGURATION PROCESS 25
ABOUT THE CONFIGURATION FILES 25
CONFIGURE PORTS 28
CONFIGURE JSM HTTP SERVER 29
CONFIGURE SERVICES 39
REFERENCE GUIDE: JSM HTTP SERVER CONFIGURATION 45

APPENDICES 53

GLOSSARY 53
REFERENCED DOCUMENTS AND SOURCES 54
ASSUMED AND PREREQUISITE KNOWLEDGE 55
BROWSER BEHAVIOUR DIFFERENCES 55
WORKSTATION AND DEVICE REQUIREMENTS 55
RUNNING DE AND WSFM SIDE BY SIDE 56

Content © LANSA Group
All Rights Reserved

Article I. Contents

WELCOME TO DATA EXPLORER FOR ADMINISTRATORS 7

ROAD MAP 7

SECTION 1 - GETTING USERS STARTED 8

WHAT IS DATA EXPLORER? 8
INSTALLATION FOR USERS OF DE 10
Starting DE 10
Log on 11

SECTION 2 - ADMINISTRATION 12

ARCHITECTURE 12
Components and structure 12
DE database 12
ADMINISTRATOR RESPONSIBILITIES 13
BEFORE YOU START 13
Prerequisites 13
Decisions you need to make 13
Port numbers 13
Register users 13
SQL policies 13
Installation and configuration process 14
INSTALL SERVER COMPONENTS 14
New installations 14
Upgrades 14
LICENCES 14
RELATED PRODUCTS 14
REGISTER USERS 15
Why register users? 15
What is registration? 15
How to register users manually 15
Use SQL to register users 15
Write a program to register users 16
Register multiple users as a batch job 16
How to register users automatically 16
How to remove registered users 16
User identifications beginning with the letter Q 16
DOCUMENTATION 17
Administrator and user guides 17
Portfolios 17

SECTION 3 – HOUSEKEEPING AND TROUBLESHOOTING 19

LOGGING 19
Log files: names and locations 19
Enable and disable logging 19
Archive log files 19
Delete log files 19
TRACING 20
Trace files: names and locations 20
Enable and disable tracing 20
Clear trace files 20
Archive trace files 21
CERTIFICATE MANAGEMENT 21

Content © LANSA Group
All Rights Reserved

TROUBLESHOOTING 22
Installation and configuration *22*
JSM HTTP server does not start 22
IBM Toolbox for Java not installed 22
File ownership and permissions for files in the IFS 22
DE database 22
Problems when running filters and queries *23*
SQL errors 23
Browser response is slow 23
BEFORE YOU CONTACT AXES SUPPORT 24

SECTION 4 - CONFIGURATION 25

THE CONFIGURATION PROCESS 25
ABOUT THE CONFIGURATION FILES 25
File locations *26*
Manager.properties *26*
httpd configuration file *26*
Axinfo configuration file *28*
CONFIGURE PORTS 28
CONFIGURE JSM HTTP SERVER 29
Server instance configuration *29*
Controlling access to the server instance *31*
MIME types for the server instance *32*
Virtual host configuration *33*
Virtual host access 34
Virtual host protect 35
Virtual host script 37
Virtual host MIME types 38
CONFIGURE SERVICES 39
JSM HTTP file service configuration *39*
Allow and deny user access to services *40*
User registration *42*
SQL policies *44*
Allowing or prohibiting SELECT INTO 44
REFERENCE GUIDE: JSM HTTP SERVER CONFIGURATION 45
Configuration item reference *45*
MIME types *48*
Access allow and deny directives *49*
Sample configurations *50*
Default configuration for DE 50

APPENDICES 53

GLOSSARY 53
REFERENCED DOCUMENTS AND SOURCES 54
ASSUMED AND PREREQUISITE KNOWLEDGE 55
BROWSER BEHAVIOUR DIFFERENCES 55
WORKSTATION AND DEVICE REQUIREMENTS 55
Hardware *55*
Software *56*
Connectivity *56*
RUNNING DE AND WSFM SIDE BY SIDE 56
Configuration options *56*
User registration when operating both DE and WSFM *57*
Configuration files *58*
axinfo.json 58

Content © LANSA Group
All Rights Reserved

Article II. List of Figures

Figure 1: Data Explorer Screen Components and Layout 8
Figure 2: Options for Viewing and Copying Query Data 9
Figure 3: View Query Data as Text (tab separated format) 9
Figure 4: Access Logon Page 11
Figure 5: Data Explorer Architecture and Components 12
Figure 6: DE and WSFM on a Single Instance and One Virtual Host 56
Figure 7: DE and WSFM on a Single Instance and Two Virtual Hosts 57
Figure 8: DE and WSFM on Two Instances and Two Virtual Hosts 57

Article III. List of Tables

Table 1: DE Database - SQL Tables 12
Table 2: Documentation: Administrator and User Guides 17
Table 3: Documentation: Administrator Portfolio 17
Table 4: Documentation: User Portfolio 18
Table 5: Information to Send to Support 24
Table 6: Configuration File Locations 26
Table 7: Explanation of manager.properties 26
Table 8: Example of an Updated manager.properties File 26
Table 9: Structure of the httpd Configuration File 27
Table 10: Mandatory and Optional Changes to Configuration Items and Parameters 27
Table 11: axinfo.json Configuration File for DE 28
Table 12: Configuration Items in axinfo.json 28
Table 13: Default Ports 28
Table 14: Configure Server Instance 29
Table 15: What to Configure for the Server Instance 30
Table 16: Control Access to the Server Instance 31
Table 17: What to Configure for Server Instance Access 31
Table 18: MIME Types for the Server Instance 32
Table 19: What to Configure for the Server Instance MIME Types 33
Table 20: Virtual Host Configuration 33
Table 21: What to Configure for the Virtual Host 33
Table 22: Virtual Host Access Directives 34
Table 23: What to Configure for Virtual Host Access 35
Table 24: Virtual Host Protect Configuration 36
Table 25: Configuring Virtual Host Protect 36
Table 26: Virtual Host Script Configuration 37
Table 27: Virtual Host MIME Type Configuration 38
Table 28: What to Configure for Virtual Host MIME Types 39
Table 29: Configure JSM HTTP File Service 39
Table 30: What to Configure for JSM HTTP File Service 40
Table 31: User Identification/Profile service.user Allow and Deny Parameter Syntax 40
Table 32: Example Configuration for service.user Allow and Deny Parameters 41
Table 33: Controlling Access to Services – service.user Allow and Deny Examples 42
Table 34: Autoregister Parameter for the Service axes/dbmservice.jsp 43
Table 35: Autoregister Parameter for Automated or Manual User Registration 43
Table 36: Configuring Use of the SELECT INTO Clause 44
Table 37: Allowing or Prohibiting SELECT INTO in SQL Queries 44

Content © LANSA Group
All Rights Reserved

TABLE 38: SERVER REFERENCE: CONFIGURATION ITEM REFERENCE 45
TABLE 39: JSM HTTP SERVER REFERENCE: MIME TYPE EXAMPLES 48
TABLE 40: SERVER REFERENCE: ACCESS ALLOW AND DENY ADDRESSES 49
TABLE 41: SERVER REFERENCE: ACCESS ALLOW AND DENY CONTENT LENGTH 49
TABLE 42: SERVER REFERENCE: ACCESS ALLOW AND DENY USER AGENTS 50
TABLE 43: SERVER REFERENCE: SAMPLE LISTS OF USER AGENTS 50
TABLE 44: SERVER REFERENCE: DEFAULT HTTPD CONFIGURATION FOR DE 51
TABLE 45: GLOSSARY OF ABBREVIATIONS AND TERMS 53
TABLE 46: REFERENCED DOCUMENTS AND INFORMATION SOURCES 54
TABLE 47: ASSUMED AND PREREQUISITE KNOWLEDGE 55
TABLE 48: AXINFO.JSON CONFIGURATION FILE FOR BOTH DE AND WS 58

Content © LANSA Group
All Rights Reserved

Welcome to Data Explorer for administrators

aXes Data Explorer (referred to as DE) provides query tools to retrieve data from databases on IBM i (System i, iSeries or AS/400) servers. Typical tasks are:

- Search for database tables using DE table filters
- Create queries with Structured Query Language (SQL)
- Run the queries to retrieve data from database tables
- Save the query definitions in folders ready for use again
- Download the data produced by queries to your computer or mobile device

Once you have downloaded or copied the query data, you can import the query data into business applications like Microsoft Excel and Word.

This guide describes how to manage aXes Data Explorer and the configuration options available to administrators.

Road Map

Use the road map to find information about aXes Data Explorer.

If you want to...	Refer to...
Get users started quickly	Section 1 - Getting users started Client or user installation, starting DE and log on instructions.
Find out about administrator activies and responsibilities	Section 2 - Administration This section explains administrative concepts and responsibilities.
Fix problems and manage day-to-day activities	Section 3 – Housekeeping and troubleshooting This section explains how to perform housekeeping tasks and track down problems.
Set values for configuration items and parameters	Section 4 - Configuration This section explains how to configure DE by setting values for the configuration items and parameters. The section includes a reference guide for the JSM HTTP server.
Read reference information	Appendices

For information about using aXes Data Explorer to create, edit and run queries refer to the aXes User Guide, Pocket Guide and the Questions and Answers document.

Content © LANSA Group
All Rights Reserved

Section 1 - Getting users started

Getting users started provides an overview of DE including an explanation of the page content, layout and what DE can do to help you query database tables.

This section describes how to install, start and log on to DE from your computer or mobile device.

What is Data Explorer?

DE is a Web application that you use to query database tables on an IBM i server. The DE page consists of two panes. The left-hand pane is the search filter and folder navigation pane. It contains filters used for table searches, queries and folders where you save the filters. The right hand pane consists of one or more tabs. The tabs contain table search results, query data, forms for editing queries and search criteria and setting options.

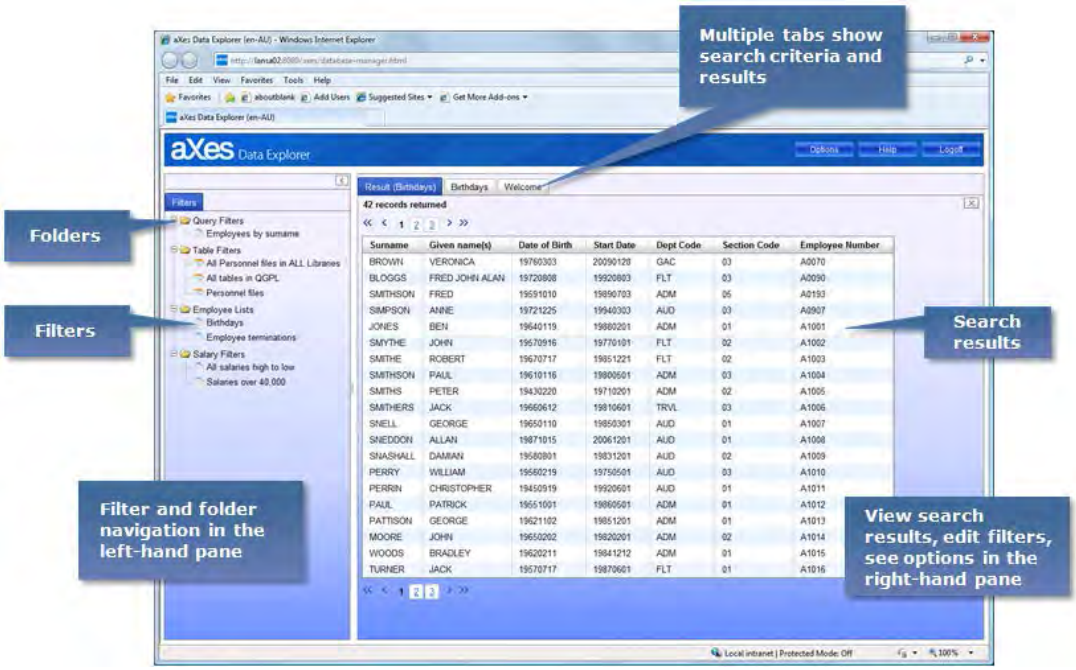


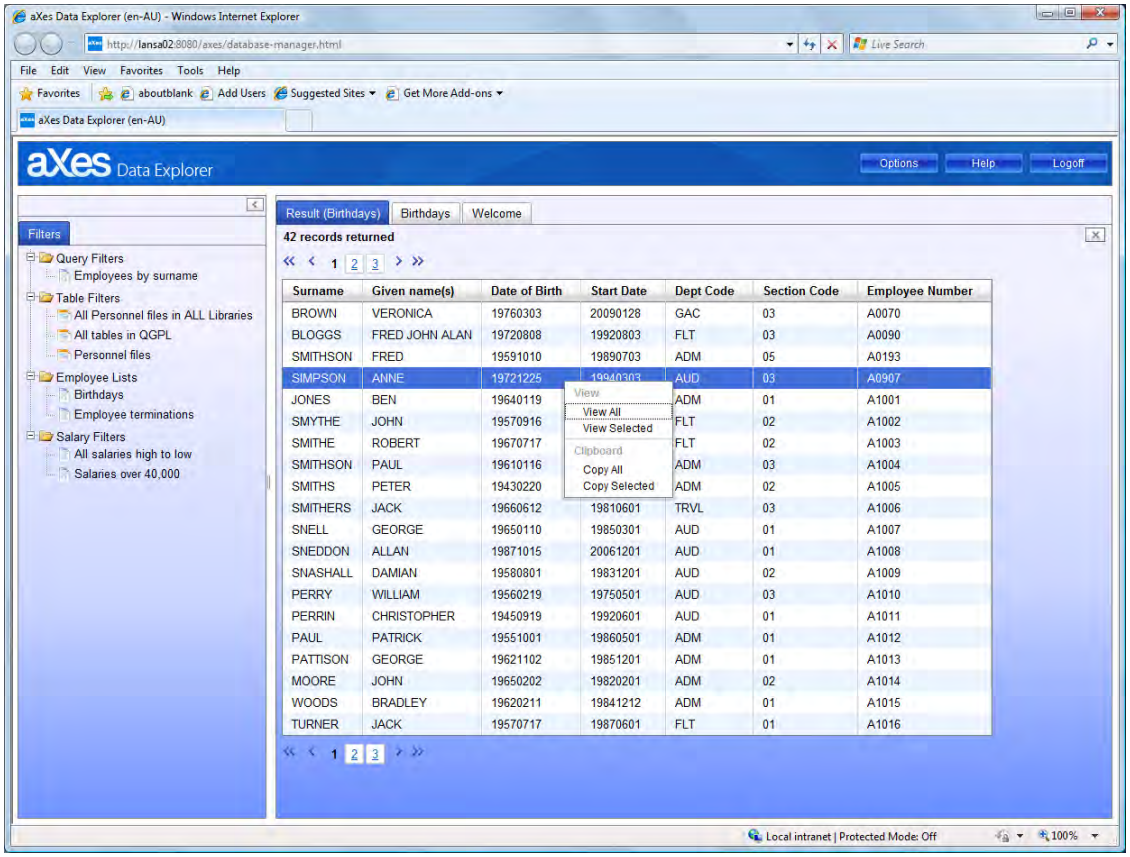
Figure 1: Data Explorer Screen Components and Layout

Figure 1 (page 8) shows three tabs in the right hand pane. The visible tab, labelled "Result (Birthdays)", presents the query data obtained by running a query filter. The second tab (out of view) titled Birthdays is the edit tab for the query filter and the third tab is the Welcome page (also out of view).

Figure 2 (page 9) shows a floating menu with options for working with query data. The options are viewing all or selected rows in the query data and copying all or selected rows.

Clicking the left mouse button on a row in the list will select the row. Holding the Ctrl key and clicking on multiple rows will select the rows.

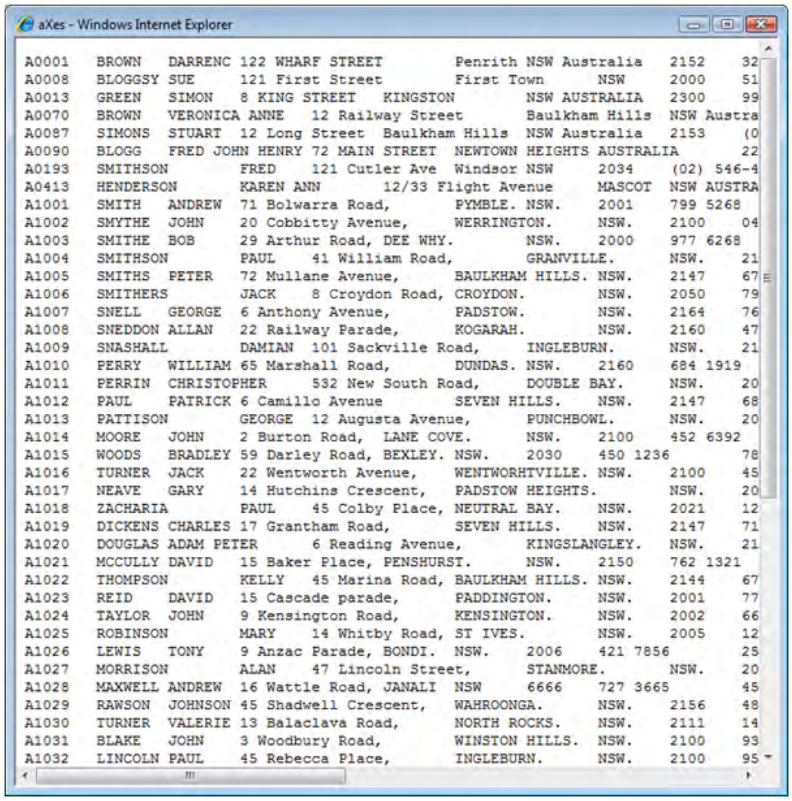
Clicking the right mouse button on a selected row opens the floating menu showing options for working with query data.



Surname	Given name(s)	Date of Birth	Start Date	Dept Code	Section Code	Employee Number
BROWN	VERONICA	19760303	20090128	GAC	03	A0070
BLOGGS	FRED JOHN ALAN	19720808	19920803	FLT	03	A0090
SMITHSON	FRED	19591010	19890703	ADM	05	A0193
SIMPSON	ANNE	19721225	19940303	AUD	03	A0907
JONES	BEN	19640119	[illegible]	ADM	01	A1001
SMYTHE	JOHN	19570916	[illegible]	FLT	02	A1002
SMITHE	ROBERT	19670717	[illegible]	FLT	02	A1003
SMITHSON	PAUL	19610116	[illegible]	ADM	03	A1004
SMITHS	PETER	19430220	[illegible]	ADM	02	A1005
SMITHERS	JACK	19660612	19810601	TRVL	03	A1006
SNELL	GEORGE	19650110	19850301	AUD	01	A1007
SNEDDON	ALLAN	19871015	20061201	AUD	01	A1008
SNASHALL	DAMIAN	19580801	19831201	AUD	02	A1009
PERRY	WILLIAM	19560219	19750501	AUD	03	A1010
PERRIN	CHRISTOPHER	19450919	19920601	AUD	01	A1011
PAUL	PATRICK	19551001	19860501	ADM	01	A1012
PATTISON	GEORGE	19621102	19851201	ADM	01	A1013
MOORE	JOHN	19650202	19820201	ADM	02	A1014
WOODS	BRADLEY	19620211	19841212	ADM	01	A1015
TURNER	JACK	19570717	19870601	FLT	01	A1016

Figure 2: Options for Viewing and Copying Query Data

Clicking the View All or View Selected items from the floating menu opens a new window to show the query data displayed as text.




```
A0001   BROWN   DARRENC 122 WHARF STREET        Penrith NSW Australia   2152    32
A0008   BLOGGSY SUE     121 First Street        First Town      NSW     2000    51
A0013   GREEN   SIMON   8 KING STREET   KINGSTON        NSW AUSTRALIA   2300    99
A0070   BROWN   VERONICA ANNE   12 Railway Street       Baulkham Hills  NSW Austra
A0087   SIMONS  STUART  12 Long Street  Baulkham Hills  NSW Australia   2153    (0
A0090   BLOGG   FRED JOHN HENRY 72 MAIN STREET  NEWTOWN HEIGHTS AUSTRALIA       22
A0193   SMITHSON        FRED    121 Cutler Ave  Windsor NSW     2034    (02) 546-4
A0413   HENDERSON       KAREN ANN       12/33 Flight Avenue     MASCOT  NSW AUSTRA
A1001   SMITH   ANDREW  71 Bolwarra Road,       PYMBLE. NSW.    2001    799 5268
A1002   SMYTHE  JOHN    20 Cobbitty Avenue,     WERRINGTON.     NSW.    2100    04
A1003   SMITHE  BOB     29 Arthur Road, DEE WHY.        NSW.    2000    977 6268
A1004   SMITHSON        PAUL    41 William Road,        GRANVILLE.      NSW.    21
A1005   SMITHS  PETER   72 Mullane Avenue,      BAULKHAM HILLS. NSW.    2147    67
A1006   SMITHERS        JACK    8 Croydon Road, CROYDON.        NSW.    2050    79
A1007   SNELL   GEORGE  6 Anthony Avenue,       PADSTOW.        NSW.    2164    76
A1008   SNEDDON ALLAN   22 Railway Parade,      KOGARAH.        NSW.    2160    47
A1009   SNASHALL        DAMIAN  101 Sackville Road,     INGLEBURN.      NSW.    21
A1010   PERRY   WILLIAM 65 Marshall Road,       DUNDAS. NSW.    2160    684 1919
A1011   PERRIN  CHRISTOPHER     532 New South Road,     DOUBLE BAY.     NSW.    20
A1012   PAUL    PATRICK 6 Camillo Avenue        SEVEN HILLS.    NSW.    2147    68
A1013   PATTISON        GEORGE  12 Augusta Avenue,      PUNCHBOWL.      NSW.    20
A1014   MOORE   JOHN    2 Burton Road,  LANE COVE.      NSW.    2100    452 6392
A1015   WOODS   BRADLEY 59 Darley Road, BEXLEY. NSW.   2030    450 1236        78
A1016   TURNER  JACK    22 Wentworth Avenue,    WENTWORHTVILLE. NSW.    2100    45
A1017   NEAVE   GARY    14 Hutchins Crescent,   PADSTOW HEIGHTS.        NSW.    20
A1018   ZACHARIA        PAUL    45 Colby Place, NEUTRAL BAY.    NSW.    2021    12
A1019   DICKENS CHARLES 17 Grantham Road,       SEVEN HILLS.    NSW.    2147    71
A1020   DOUGLAS ADAM PETER      6 Reading Avenue,       KINGSLANGLEY.   NSW.    21
A1021   MCCULLY DAVID   15 Baker Place, PENSHURST.      NSW.    2150    762 1321
A1022   THOMPSON        KELLY   45 Marina Road, BAULKHAM HILLS. NSW.    2144    67
A1023   REID    DAVID   15 Cascade parade,       PADDINGTON.     NSW.    2001    77
A1024   TAYLOR  JOHN    9 Kensington Road,      KENSINGTON.     NSW.    2002    66
A1025   ROBINSON        MARY    14 Whitby Road, ST IVES.        NSW.    2005    12
A1026   LEWIS   TONY    9 Anzac Parade, BONDI.  NSW.    2006    421 7856        25
A1027   MORRISON        ALAN    47 Lincoln Street,      STANMORE.       NSW.    20
A1028   MAXWELL ANDREW  16 Wattle Road, JANALI  NSW     6666    727 3665        45
A1029   RAWSON  JOHNSON 45 Shadwell Crescent,   WAHROONGA.      NSW.    2156    48
A1030   TURNER  VALERIE 13 Balaclava Road,      NORTH ROCKS.    NSW.    2111    14
A1031   BLAKE   JOHN    3 Woodbury Road,        WINSTON HILLS.  NSW.    2100    93
A1032   LINCOLN PAUL    45 Rebecca Place,       INGLEBURN.      NSW.    2100    95
```

Figure 3: View Query Data as Text (tab separated format)

Figure 3 (page 9) is an example of query data viewed as text that has been formatted as tab separated values (TSV). Clicking the Copy All or Copy Selected items from the floating menu copies the query data to the clipboard. You can paste the query data to other applications that support the clipboard (Microsoft Excel is an example).

Installation for users of DE

Users do not need to install any software on their computer or mobile device to use DE.

The prerequisites for DE are:

- A supported browser - Chrome, Firefox, Internet Explorer and Safari
- A network connection to the Internet and/or corporate network (fixed line or wireless)
- The address (URL) for DE at your site

Administrators will provide the address (URL) for DE or place an icon on the desktop.

Starting DE

Starting DE from your browser

To start DE from your browser:

1.	Open the browser.
2.	Type the address (URL) into the location bar and press the Enter key or the Go icon.
3.	Wait for DE to display the Access log on page and then follow the log on instructions.

Starting DE from a desktop shortcut

You or your administrator can set up a shortcut to open the browser and go to the address for DE immediately.

To start DE from a shortcut:

1.	Click on the shortcut (or open the shortcut) - the browser will start and connect to DE.
2.	Wait for DE to display the Access log on page and then follow the log on instructions.

To set up the desktop shortcut for Internet Explorer, create a new shortcut, open the properties and insert the equivalent of the following as the target property:

"C:\Program Files\Internet Explorer\iexplore.exe" www.myaXesDE-URL.com

To set up the desktop shortcut for Firefox, create a new shortcut, open the properties and insert the equivalent of the following as the target property:

"C:\Program Files\Mozilla Firefox\firefox.exe" www.myaXesDE-URL.com

The URL http://www.myaXesDE-URL.com:80/axes/database-manager.html is an example of the actual URL for DE at your site. This example uses port 80 and the default page for DE.

Content © LANSA Group
All Rights Reserved

Log on

When DE starts, the browser will show the Access log on page and display the Access dialogue window.

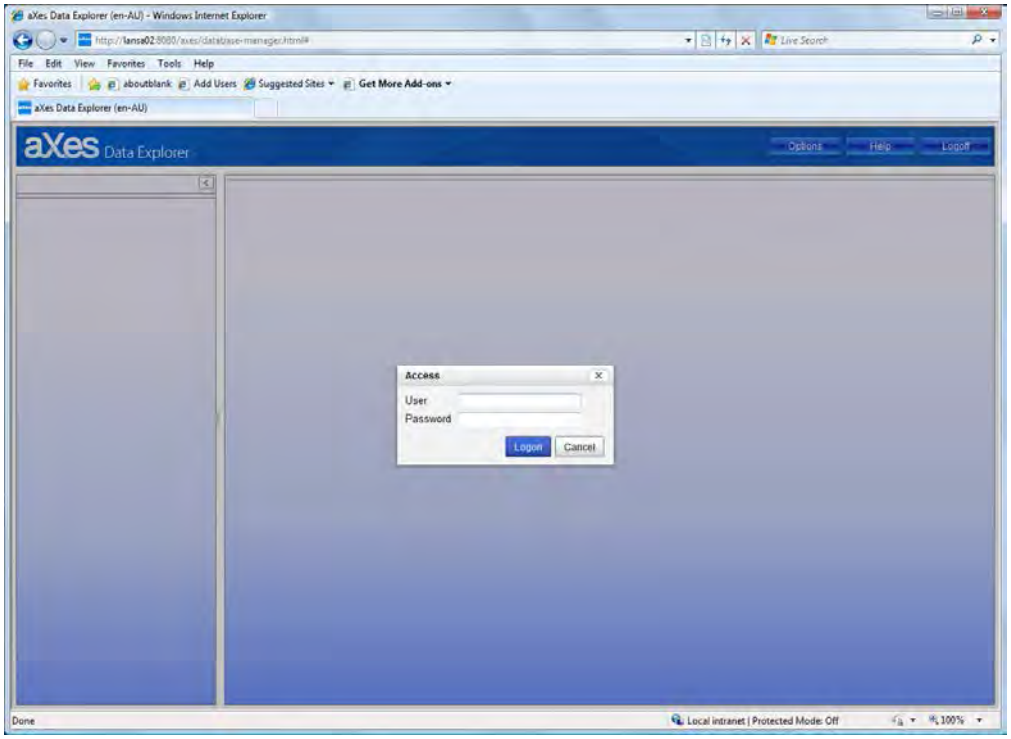


Figure 4: Access Logon Page

Your user identification and password

To log on to DE you must provide your user identification and password (credentials):

1.	Type your user identification.
2.	Type your password.
3.	Press the Logon button
4.	Wait while DE verifies your credentials. If your log on is successful, DE will display the next page. If your log on is unsuccessful, DE will display an error message.

What to do if log on fails

Log on will fail if you make a mistake typing the user identification and/or the password. Check you have the correct user identification and re-type the password.

If you forget your user identification and/or password contact your administrator.

If you do not know your user identification and/or password contact your administrator.

Cancel DE without logging on

To cancel DE without logging on click the Cancel button and wait while DE closes.

Content © LANSA Group
All Rights Reserved

Section 2 - Administration

This section provides conceptual information about DE that will help administrators manage DE effectively.

Architecture

Components and structure

Figure 5 (page 12) illustrates the architecture of DE and highlights the primary components.

DE is a Web application consisting of server components and a user interface presented in a browser. The user interface is a browser page consisting of HTML, CSS and JavaScript. No components of DE reside on the computer running the browser.

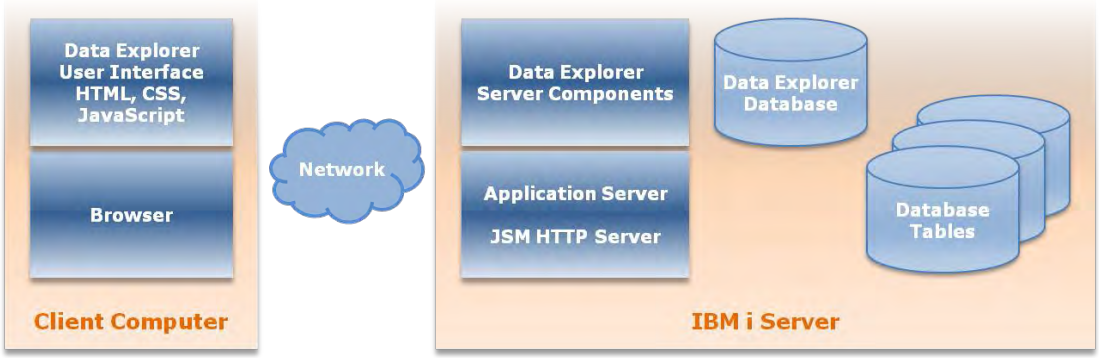


Figure 5: Data Explorer Architecture and Components

The server components are the application server (JSM HTTP server) and DE itself running on the application server. DE operates a database for its data management.

Users connect to and query database tables from their browser via DE. Users submit query filters containing SQL, DE verifies the SQL, submits the query, marshals the query data retrieved by the query, and sends the data to the browser.

DE database

DE uses its database to store registered users, the folders they create and the filters or queries they build. The DE database comprises the tables listed in Table 1 (page 12).

Table 1: DE Database - SQL Tables

Names	Properties	Purpose
DBMUSER	*FILE PF-DTA	User tables contain details about registered users
DBMFOLDER	*FILE PF-DTA	Folder tables describe folders created by users
DBMFILTER	*FILE PF-DTA	Filter tables describe the filters (queries) created by users

DE creates and manages these files.

The configuration file defines the location (or library) where the files reside.

Important reminders	Backup the DE database regularly. Backup the DE database before you uninstall and reinstall. Backup the DE database before you perform an upgrade.

Content © LANSA Group
All Rights Reserved

Administrator responsibilities

Administrators are responsible for:

- Installing DE on the server
- Configuring DE server components
- Managing users
- Ensuring regular backups occur

Installation and configuration are once only or occasional tasks. Housekeeping is an ongoing task. Troubleshooting is an occasional task.

Managing users is an ongoing task. Administrators are responsible for notifying users of the DE log on page URL. In cases where users operate a managed desktop, administrators will need to add a short cut to the desktop for users to start DE.

Before you start

This section describes issues you need to think through and decisions you must take before installing and configuring DE.

Prerequisites

The prerequisites for DE are:

- Installation media
- Licences

The prerequisite software from IBM is:

- iOS version 5, release 3 or later
- IBM Java JDK 1.4 (or greater)
- IBM host server daemons must be running to accept IBM Toolbox for Java connections
- IBM Toolbox for Java (the the JAR file jt400.jar is shipped with DE)

Install the prerequisite software from IBM before installing DE.

Important reminder	DE will not operate successfully without the prerequisite software.

Decisions you need to make

Port numbers

Select the port or ports the JSM HTTP server will use for DE.

The default ports are 5563 or 5564 when using TLS/SSL.

Register users

Choose a method for registering users. The choice you make will determine the pre-installation work you will need to do. If you choose to load users as a batch task, you must assemble the user details and write SQL and/or programs to load the user details into the user table. If you choose automatic registration, the only work necessary is setting a parameter value in the configuration file.

Read the section of this guide that describes user registration concepts ("Register users", page 15). The configuration section explains how to enable and disable automatic registration.

SQL policies

Decide the features of SQL that query writers may use.

Content © LANSA Group
All Rights Reserved

Installation and configuration process

The steps in the installation and configuration process are:

1.	Read the documentation. Read this guide and familiarise yourself with the DE User Guide.
2.	Install the prerequisite software on the server
3.	Install DE on the server
4.	Apply licences
5.	Configure the JSM HTTP server
6.	Configure DE
7.	Register users

You will reduce your workload if you are familiar with the options for configuration and registering users before you start.

Install server components

New installations

DE ships as part of aXes and installing aXes also installs DE. Refer to the aXes administration documentation to understand the installation procedures for DE.

During installation, DE copies the template configuration file named httpd-template.xml to an instance configuration named httpd.xml. This file contains default configuration settings. Once the installation is complete, DE requires no further installation or configuration tasks.

You can amend the instance configuration if necessary. Before amending the instance configuration, make a backup of the httpd.xml file.

Upgrades

During upgrades DE will add components, amend installed components and remove deprecated components. Upgrades will replace the existing copy of the configuration file named httpd-template.xml. Therefore, it is essential that you copy and backup configuration files before changing configuration items and before installing upgrades.

Upgrades will not replace the existing httpd.xml configuration file. However, upgrades may require changes to individual configuration items in the file.

Licences

Licence arrangements determine your use of DE.

To use DE your licence arrangements must include session and/or processor licences for DE.

Applying licences is part of the aXes installation process.

Related products

Web Spooled File Manager (WSFM) may be installed beside DE and share its configuration with DE.

The section of this guide titled, "Running DE and WSFM side by side" (page 56) explains the options for configuring WSFM and DE together.

Content © LANSA Group
All Rights Reserved

Register users

Why register users?

People using DE have access to corporate databases. Corporate best practice suggests that companies ought to know and control who accesses data in their databases.

Compliance and regulatory obligations require companies to maintain audit logs describing who accesses corporate information.

For these reasons you should ensure that only registered users can use DE to access the databases. Each person who uses DE needs identifying credentials (unique user name and password). Shared or generic user identifications will compromise the ability to determine who accesses the databases.

What is registration?

Users need an active IBM i user profile to register for DE.

DE includes a configuration parameter for user registration that determines how DE will behave when a user logs on. The parameter allows two behaviours:

- Register users manually
- Register users automatically

With DE configured for manual registration, everyone who logs on must already be known to DE. Manual registration means that an administrator must add each authorised user to the list of registered users before the person can use DE.

With DE configured to allow automatic registration, any person who logs on with an active IBM i user profile and is unknown to DE will be added as a registered user.

How to register users manually

To register a user to use DE you insert a record in the table DBMUSER.

There are several options for registering user manually:

- Use SQL to add a user
- Write a program to add a user
- Load multiple users as a batch job (using SQL and/or programs)

Using an appropriate CCSID is essential when registering users manually. DE's database is DBCS, uses a graphic field (CCSID 1200), and data in the tables includes square brackets [ ] and braces { }, also known as curly brackets. Use CCSID 37 as the job CCSID to ensure correct character conversions. This recommendation applies to all methods that register users manually, both interactive SQL and programmatic updates.

Using other CCSIDs may produce incorrect conversions, two examples are CCSID 37 and CCSID 500:

- CCSID 37 produces correct data: {"startupfilterid":0,"recent":[15],"showwelcome":true}
- CCSID 500 produces incorrect data:
  {"startupfilterid":0,"recent":¢15!,"showwelcome":true}

The data created using CCSID 500 interprets the square brackets incorrectly, [becomes ¢ and] becomes !. CCSID 37 produces the correct result.

The command to set CCSID 37 for a job is: CHGJOB CCSID(37)

Use SQL to register users

You can add a registered user by running the following SQL:

```
INSERT INTO [Library]/DBMUSER (PROFILE,OPTIONS)
VALUES('USER','{"startupfilterid":0,"showwelcome":true}')
```

Where **[Library]** is the name of the library in which the table resides (typically, the library name is AXES).

Where **USER** is the user profile name in uppercase.

To use this SQL the CCSID for the job must not be 65535. Use the CHGJOB command to change the CCSID.The user profile must be an active user profile for the IBM i server.

Before loading a new user, check DBMUSER to ensure that the user is not already loaded.

Write a program to register users

To automate user registration, write a program that accepts the user profile and inserts the user into the DBMUSER table. The program may use the SQL above to insert the user.

Register multiple users as a batch job

To load multiple users as a batch job, create a text file containing all users. Then load the users into the DBMUSER table using SQL or write a program to load the users into the table from the text file.

How to register users automatically

DE will register users when they log on, if configured to allow automatic registration.

Registering a large number of users will be a significant effort for administrators. One way to reduce this effort is to allow automatic registration for a short period after installation. As users log on DE will register their user identification. At the end of a short automatic registration period, change the registration parameter to manual. This approach will allow people to register automatically and reduce the administrative effort needed to register users.

How to remove registered users

Remove a registered user by deleting the user from the DE database. That is, delete the user from the filter table (DBMFILTER), the folder table (DBMFOLDER) and finally the user table (DBMUSER). Deleting the user from the user table but not the other tables will leave folders and filters without a related user.

You can delete the rows manually, write SQL or write a program.

The steps in the removal process are:

1.	Determine the identification associated with the user. Run a SELECT query with a WHERE clause user profile is "user profile".
2.	Delete rows in the filter table using the identification.
3.	Delete rows in the folder table using the identification.
4.	Delete rows in the user table using the identification.

An automated removal process (using SQL or a program) will follow the same process.

User identifications beginning with the letter Q

DE prohibits users logging on with a user identification that begins with the letter Q.

If you wish to override this policy and allow the use of these user identifications, you must configure a parameter for each user identification you wish to allow.

Content © LANSA Group
All Rights Reserved

Documentation

The documentation consists of a collection of documents formatted as PDF (Adobe Portable Document Format).

Administrator and user guides

Table 2 (page 17) lists the documentation available for DE and explains the purpose of each document.

Table 2: Documentation: Administrator and User Guides

Document	Purpose
Administrator Guide	The Administrator Guide explains how to configure DE and to manage access to services by people using DE. This guide is intended for use by administrators. The Administrator Guide is this document.
Pocket Guide	The Pocket Guide is a condensed version of the User Guide. This guide is a quick reference for DE users who run queries more often than prepare queries.
Questions and Answers	The Questions and Answers document contains a brief overview of DE and the questions and answers sections from the User Guide. This guide is intended for use by any person who wants a quick answer to a question about DE.
User Guide	The User Guide provides conceptual information about DE and explains how to use DE to query database tables. This guide is mandatory reading if you want to understand how to make the best use of DE. The guide is intended for use by people preparing and running queries. Administrators should familiarise themselves with this guide when they provide service desk assistance to DE users. This is the most extensive of the documents.

The User Guide is essential reading for users who want to create queries.

Administrators who provide advice to users need to be familiar with the content of all the documentation.

Portfolios

A portfolio is a collection of documents. The documentation is packaged in two portfolios. The portfolios are the administrator portfolio and the user portfolio.

Table 3 (page 17) shows the content of the administrator portfolio.

Table 3: Documentation: Administrator Portfolio

Documents
Administrator Guide
Pocket Guide
Questions and Answers

Content © LANSA Group
All Rights Reserved

Documents
User Guide

Table 4 (page 18) shows the content of the user portfolio.

Table 4: Documentation: User Portfolio

Documents
Pocket Guide
Questions and Answers
User Guide

The user portfolio is accessible from the Help button at the top of the DE page.

Content © LANSA Group
All Rights Reserved

Section 3 – Housekeeping and Troubleshooting

This section describes how to perform housekeeping tasks, track down problems and contact support.

Logging

This section explains administrator tasks for managing access log files and error log files. These files accumulate with use and administrators should archive and/or clear the files periodically.

Log files: names and locations

In a typical installation log files reside in a logging folder.

Location	axes/jsm/instance/www/instance/logs/
Log File Names	error.log access.log

Examining the content of these files will reveal information about activity and errors.

Enable and disable logging

The errorlog and accesslog directives in the configuration file control logging activity.

To enable logging for both error and access logs, set the value of the enabled parameter as true:

```
<errorlog enabled="true" file="www/instance/logs/error.log"/>
<accesslog enabled="true" file="www/instance/logs/access.log"/>
```

To disable logging for both error and access logs, set the value of the enabled parameter as false:

```
<errorlog enabled="false" file="www/instance/logs/error.log"/>
<accesslog enabled="false" file="www/instance/logs/access.log"/>
```

Error and access logs operate independently. For example you can enable error logging and disable access logging.

Archive log files

Archive log files by copying them to another location.

DE provides no archiving tools.

Delete log files

Do not delete the active error.log and access.log files.

DE will archive the active log files periodically. Before deleting the archived log files, ensure you either have a backup or archive copy of the files.

Content © LANSA Group
All Rights Reserved

Tracing

This section explains the administrator tasks for collecting data to assist with tracking errors. Administrators need to enable and disable tracing services and clear trace files when they are no longer needed.

Tracing allows you to collect information about the activity of DE. When enabled, tracing directs STDOUT and STDERR to the trace files STDOUT.TXT and STDERR.TXT.

The Java Virtual Machine and instance information is logged to a MANAGER.TXT file.

Trace files: names and locations

In a typical installation trace files reside in two folders.

The first location provides high level trace information.

Location	axes/jsm/instance/trace/[job number]/
Log File Names	CLASSPATH.TXT MANAGER.TXT STDERR.TXT STDOUT.TXT
Sample	axes/jsm/instance/trace/189002/STDOUT.TXT

Where [job number] is the job number of the active job when tracing occurred.

The second location provides detailed trace information.

Location	axes/jsm/instance/trace/[job number]/[date]/
Log File Directory Names	HTTP00000000 HTTP00000001 HTTP00000002 HTTP00000003
Sample	axes/jsm/instance/trace/309001/2009-11-24/HTTP00000004

Enable and disable tracing

The trace and clienttrace directives in the configuration file control tracing activity. These directives are in the virtual host section of and are associated with services.

To enable tracing for a service, set the value of the trace and/or clienttrace parameter as true:

```
trace="true" clienttrace="true"
```

To disable tracing for a service, set the value of the trace and/or clienttrace parameter as false:

```
trace="false" clienttrace="false"
```

Tracing directives operate independently. For example you can enable tracing for one service or all services.

Clear trace files

Use the command CLRJSM command to clear trace files and subdirectories. The command has the following parameters:

INSTANCE	The instance defaults to a value of *DEFAULT. This is the recommended value.
TRACEDIR	The option defaults to a value of *YES. Valid values are *YES, *NO. The value *YES removes files and subdirectories in the trace directory.
TEMPDIR	The option defaults to a value of *YES. Valid values are *YES, *NO. The value *YES removes all files and sub-directories in the TEMP directory. Do not clear the TEMP directory while JSM services are running, as this action will delete temporary files used by the services. Always use *NO when clearing trace files from a running JSM instance.
KEEP	Valid values are in the range 0 to 99 days. Zero means do not keep the files. Running the command will delete ALL files and sub-directories. Use 1 to keep only today's files. Use 2 to keep today's and yesterday's files.

Archive trace files

Archive trace files by copying them to another location, before clearing the files.

DE provides no archiving tools.

Certificate management

Use the Java keytool command to create a private key and certificate suitable for the TLS/SSL server credentials.

```
%JDKPATH%\bin\keytool  -genkey —alias sslkey —keyalg RSA —keysize 1024
                       -sigalg SHA1withRSA
                       -keypass password —keystore wwwssl.jks
                       -storepass password —validity 365
                       -dname "CN=Web Applications,OU=Web Services,O=My
                       Company,L=My Location,S=My State,C=US"
```

Substitute your equivalent values for My Company, My Location, My State and C (country).

Troubleshooting

The troubleshooting section suggests the parts of DE to check when it does not operate as expected.

Installation and configuration

JSM HTTP server does not start

1.	Check the manager.properties httpd directive: httpd=system/httpd.xml Ensure that the name specified for the httpd configuration file is the name of the actual httpd configuration file.
2.	Check the JSM trace file MANAGER.TXT for JSM HTTP server start up messages.
3.	Review the configuration directives in the instance tag.
4.	Confirm that the JSM HTTP server is started by using a browser to access the home index.html file.

Ensure that the configuration is complete before attempting the start the JSM HTTP server.

IBM Toolbox for Java not installed

1.	Ensure that the IBM Toolbox for Java is installed. Verify that the file, "jt400.jar", resides in the JSM jar directory.

Installing the IBM Toolbox for Java is an essential prerequisite for operating DE.

File ownership and permissions for files in the IFS

1.	The JSM HTTP server by default runs as the JSM job description user profile. Any files and directories created should be owned by that user profile.
2.	Verify the user profile, file and directory ownership.
3.	Change ownerships if necessary.

DE database

1.	DE uses tables to store its information. The tables must exist in the library specified in the database.library directive: <parameter name="database.library" value="AXES"/> In this example the library is AXES

Content © LANSA Group
All Rights Reserved

2.	Verify that the tables are in the library defined in the database.library directive. The tables are: DBMUSER (PF-DTA) DBMUSER1 (LF) DBMUSER2 (LF) DBMFOLDER (PF-DTA) DBMFOLDER1 (LF) DBMFILTER (PF-DTA) DBMFILTER1 (LF) DBMFILTER2 (LF)
3.	Change the database.library directive to the library where the files reside. Or, move the files to the library defined in the database.library directive.

Problems when running filters and queries

SQL errors

Users encounter SQL errors when a filter runs a query.

1.	Check the syntax of the SQL in the filter.
2.	Ensure the SQL includes no unsupported clauses.
3.	Ensure the table names are correct.

DE provides basic support to help users create filters and run queries. This support does not include SQL syntax checking.

DE assumes that users know SQL.

Browser response is slow

The page appears slow and unresponsive.

1.	Browsers expect fast responses from HTML pages and want to exit from long running or waiting requests. The reason for this behaviour is to minimise the wait time that users experience. When pages appear to wait for some time the browser will ask the user if they wish to cancel the request.
2.	IE displays a blank page. IE displays a blank page (or white window) when it waits for long running requests. Wait until IE re-displays the page or close the browser.
3.	When JavaScript is doing a lot of processing, the browser may assume the page is in a loop and prompt the user to end the JavaScript execution. Users may continue JavaScript execution, end JavaScript execution or close the browser.
4.	Cancelling the request or closing the browser will end the DE session. Users will have to log on again. Cancelling a request or closing the browser has no effect on spooled files.

Content © LANSA Group
All Rights Reserved

Before you contact aXes support

Table 5 (page 24) contains the information you need to send to Support if you experience a problem with aXes.

Table 5: Information to Send to Support

Context	Information
Contact and customer details	Your company name Your name Customer name
aXes version(s)	What version of aXes being is being reported? What aXes PTFs are installed and applied to this version? Are other versions of aXes also installed on the same IBM i? Is Terminal Server (TS) installed? Is the Web Spooler (WS) installed? Is the Intersession Option installed as well? Has another version of aXes been successfully deployed on the same IBM i? (If so, please provide details). How is aXes licensed - processor or session based? If processor based, how many processors are licensed? If session based, how many sessions are licensed?
IBM hardware	Model number (QMODEL) Serial number (QSRLNBR) Processor Feature Number (QPRCFEAT) Processor Group (Tier) Is aXes running in an LPAR? If yes, how many Processors are allocated to that LPAR?
IBM operating system software	What operating system version is running aXes? (Version, Release and Modification) What operating system version of the IBM i Licensed Internal Code (LIC) is installed? What IBM operating system PTFs have been installed? Are the latest PTFs applied?
Problem report	Provide a detailed description of the problem. Include a full history of events leading up to the occurrence of the problem. For example, did the problem occur after you installed new software, applied PTFs or upgraded the operating system? The more detailed the information, the easier it is for Support to understand your problem.

Content © LANSA Group
All Rights Reserved

Section 4 - Configuration

This section explains the tools and configuration options available to administrators who will manage DE.

The configuration process

The steps in the configuration process are:

1.	Copy and rename the configuration file: httpd-template.xml. The recommended name for the copied file is: system/httpd.xml You can use any valid name but you should choose a name that indicates the file content.
2.	Change the httpd property in the manager.properties file to the new name allocated when you copied the original httpd configuration file. For example, if you used the recommended name for the copied httpd configuration file, change the httpd property in manager.properties to: httpd=system/httpd.xml If not, use the name allocated to the copied file.
3.	Configure axinfo.json
4.	Configure the JSM HTTP server instance.
5.	Configure the services.
6.	Configure the database connections.
7.	Configure the user registration properties.
8.	Set the policies for using SQL.

The configuration process assumes you have completed the installation successfully.

Important reminders	Always make a copy of the httpd configuration file before you edit its contents. Keep a separate copy of the httpd configuration file that is active in your production environment.

Use a text editor or XML editor when changing configuration items and parameters in the httpd configuration file. Exercise care when editing and make sure you do not change tag names.

About the configuration files

The configuration details reside in three files: manager.properties, httpd.xml and axinfo.json. This section describes the content of the configuration files. Later sections describe how to change individual configuration items.

Content © LANSA Group
All Rights Reserved

File locations

Table 6 (page 26) provides the folder locations of the configuration files.

Table 6: Configuration File Locations

Configuration Files	Locations
axinfo.json	myaxes/jsm/
httpd.xml	myaxes/jsm/instance/system/
httpd-template.xml	myaxes/jsm/instance/system/
manager.properties	myaxes/jsm/instance/system/

The root folder (myaxes) in the location is the folder name chosen on installation.

Manager.properties

The properties file, manager.properties, contains information about the installed instance of DE. Table 7 (page 26) explains the properties manager.properties.

Table 7: Explanation of manager.properties

Properties	Definitions
httpd=system/httpd.xml	Name of the httpd configuration file
httpd.axes.release=V2R1M0	Version of aXes installed (used for licence checking)
httpd.axes.instance=AXES	Name of the instance (used for licence checking)

After updating the httpd configuration file property in manager.properties, the configuration file will look like the following example (Table 8, page 26).

Table 8: Example of an Updated manager.properties File

Properties

# httpd=system/httpd-template.xml 	
httpd=system/httpd.xml
httpd.axes.release=V2R1M0
httpd.axes.instance=AXES
#
tcp.port=5560
console.tcp.port=5561
studio.client.address=*none
console.client.address=*none

The line beginning with a hash (#) is a comment line.

The name of the instance (httpd.axes.instance=AXES) in manager.properties does not need to be the same as the instance name in the httpd configuration file.

httpd configuration file

The configuration file named httpd.xml contains the configuration items and parameters for DE. The file exists in two forms. The files httpd-template.xml contains default setting and is the template for the operational file named httpd.xml.

Content © LANSA Group
All Rights Reserved

Table 9: Structure of the httpd Configuration File

Structure of httpd Configuration File

<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
  index="index.html">
  <errorlog enabled="true" file="www/instance/logs/error.log"/>
  <accesslog enabled="true" file="www/instance/logs/access.log"/>
  <listen secure="false" store="pki/wwwssl.jks" password="password"
  port="5563"
  sslport="5564"
  interface="*ALL" backlog="256"
  buffersend="-1" bufferreceive="-1"
  nodelay="false" timeout="5" />
  +<access>
  +<mimetype>
  +<virtual host="*" active="true">
  </instance>
</configuration>

Table 9 (page 27) shows the structure of the httpd configuration file.

The configuration contains an instance of the JSM HTTP server. The instance has a set of configuration parameters (instance, errorlog, accesslog and listen tags). The instance includes configuration items for access, MIME types (or Internet Media Types) and virtual hosts.

Table 10 (page 27) provides a list of recommendations for changing configuration items and parameters in the httpd configuration file.

Table 10: Mandatory and Optional Changes to Configuration Items and Parameters

Configuration Items	Change Mandatory/Optional
JSMHTTPServiceFile (match URI, class)	Do not change these items
MIME types (server instance)	Optional, but not recommended
MIME types (virtual host)	Optional, but not recommended
Ports	You must choose a port number
Realm (protect tag, virtual host)	Do not change any items in the protect tag
Virtual host name and active	Do not change these items unless you choose multiple instances and/or multiple virtual hosts

The default values of many configuration items and parameters in the httpd configuration will be appropriate for operating DE. Some values you must change and some are optional.

Important reminders	Copy the original httpd configuration file. Do not use the original httpd configuration file for your configuration settings. Version upgrades and fixes may alter the parameters in the httpd configuration file and will over-write your configuration settings when installed.

Axinfo configuration file

The axinfo configuration file defines whether DE is enabled, the ports and URI for the DE service.

Table 11: axinfo.json Configuration File for DE

axinfo.json Configuration Example

`{`
`  "dbm":`
`  {`
`  "enabled":  true,`
`  "port":  5563,`
`  "sslport":  5564,`
`  "uri":  "/axes/database-manager.html"`
`  }`
`}`

Table 12 (page 28) explains the items in axinfo.json.

Table 12: Configuration Items in axinfo.json

Configuration Items	Definitions
Enabled	"enabled":true indicates that DE is active "enable":false indicates that DE is inactive (that is, the service is unavailable)
Ports	Port = the non SSL port (not encrypted) SSLPort = the SSL port (encrypted)

Configure ports

The axinfo and httpd configuration files contain items to define the ports DE will use. Table 13 (page 28) shows the default ports.

Table 13: Default Ports

Port Numbers	Definitions
5560	JSM Server TCP port is used internally and does not accept connections.
5561	JSM Console TCP port is used internally and does not accept connections.
5563	JSM HTTP Server (default port for DE without TLS/SSL)
5564	JSM HTTPS Server (default port for DE with TLS/SSL)

Content © LANSA Group
All Rights Reserved

To change ports follow these steps:

1.	Edit axinfo.json.
2.	Change the port parameter. Change the sslport parameter.
3.	Save axinfo.json
4.	Edit httpd.xml (or the equivalent file if you use a different name).
5.	Under the listen tag of the instance: Change the port parameter. Change the sslport parameter.
6.	Save httpd.xml

The configuration files shipped with DE are configured to use the default ports. You do not need to change the port configurations if the default ports are suitable for your installation.

Configure JSM HTTP server

This part of the administrator guide explains how to configure the sections of the httpd configuration file that apply to the JSM HTTP server. There are separate sections that explain how to configure individual services that the JSM HTTP server supports.

Server instance configuration

Table 14 (page 29) shows the set of configuration items and parameters that apply to the JSM HTTP server instance.

Table 14: Configure Server Instance

Configure Server Instance Example

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
            index="index.html">
        <errorlog enabled="true" file="www/instance/logs/error.log"/>
        <accesslog enabled="true" file="www/instance/logs/access.log"/>
        <listen secure="false" store="pki/wwwssl.jks" password="password"
            port="5563"
            sslport="5564"
            interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
            nodelay="false" timeout="5"/>
        +<access>
        +<mimetype>
        +<virtual host="*" active="true">
    </instance>
</configuration>
```

In this example, the instance of DE has error logging and access logging enabled (the value of the parameter is "true"). The server will place errors into error.log and access events into access.log. The instance will listen on port 5563.

Table 15: What to Configure for the Server Instance

To configure	Apply these settings	Change default
Access logging	Set accesslog enabled to true. Define the access log file name.	Optional
Active	Active must be true	Do not change
Backlog	Choose the depth of the TCP/IP queue.	Optional
Buffer receive	Set the size (bytes) of the receive buffer	Optional
Buffer send	Set the size (bytes) of the send buffer	Optional
Error logging	Set errorlog enabled to true. Define the error log file name.	Optional
Index	Define the index document	Optional
Instance name	Choose a name	Optional
Interface	Use *ALL for all addresses or choose the interface address. Default is all interfaces.	Optional
No delay	Set to True to enable TCP/IP no delay. False uses the operating system settings.	Optional
No TLS/SSL	Set listen secure to false. When listen secure is false, the JSM HTTP server ignores store and password values.	Optional
Port	Insert the port number you want to use.	Mandatory The server expects either port and/or sslport.
Root	Define the root directory (folder)	Optional
Secure connection with TLS/SSL	Set listen secure to true. Define the name of the store. Insert the password.	Optional
SSLport	Insert the port number you want to use.	Mandatory The server expects either port and/or sslport.
Timeout	Set the connection timeout in seconds.	Optional

Whether you need to change the optional items depends on the environment of your DE installation and your performance requirements.

Content © LANSA Group
All Rights Reserved

Controlling access to the server instance

The access directive (at the instance level) specifies rules for accessing the instance of the JSM HTTP server. Using the access directive you can:

- Allow addresses
- Deny addresses

You use combinations of allow and deny directives to control access to the server instance. Addresses can be specific (10.2.45.1), masks (10.2) or generic (indicated by the asterisk (*)).

Do not use directives for user agents or content lengths in this section.

Table 16 (page 31) shows an example of allow and deny directives (at the instance level).

Table 16: Control Access to the Server Instance

Server Instance Access (Allow and Deny) Configuration Example

<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
  index="index.html">
  <errorlog enabled="true" file="www/instance/logs/error.log"/>
  <accesslog enabled="true" file="www/instance/logs/access.log"/>
  <listen secure="false" store="pki/wwwssl.jks" password="password"
  port="5563"
  sslport="5564"
  interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
  nodelay="false" timeout="5"/>

<access> <allow address="*"/> </access>
+<mimetype> +<virtual host="*" active="true">   </instance> </configuration>

In this example <allow address= "*"/> allows access from any address. The other allow and deny directives are inside comments and ignored by the JSM HTTP server.

Table 17: What to Configure for Server Instance Access

To configure	Apply these settings	Change defaults
Allow any address	Use allow address=* <allow address="*"/>	Optional
Access for specific addresses	Add new a directive for each allowed address. <allow address="10.2.1.45"/>	Optional
Deny any address	Use deny address=* <deny address="*"/>	Optional

To configure	Apply these settings	Change defaults
Deny access for specific addresses	Add new a directive for each denied address. <deny address="10.2.1.45"/>	Optional

MIME types for the server instance

Table 18 (page 32) shows the configuration for MIME types. This part of the httpd configuration file defines MIME types applicable to the whole instance.

The MIME type directives allow the JSM HTTP server to correctly understand the nature of files. Administrators do not need to change these directives.

Table 18: MIME Types for the Server Instance

Server Instance MIME Types Configuration Example

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
            index="index.html">
        <errorlog enabled="true" file="www/instance/logs/error.log"/>
        <accesslog enabled="true" file="www/instance/logs/access.log"/>
        <listen secure="false" store="pki/wwwssl.jks" password="password"
            port="5563"
            sslport="5564"
            interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
            nodelay="false" timeout="5"/>
        +<access>
        <mimetype>
            <map extension="png" type="image/png"/>
            <map extension="gif" type="image/gif"/>
            <map extension="jpg" type="image/jpeg"/>
            <map extension="jpeg" type="image/jpeg"/>
            <map extension="tiff" type="image/tiff"/>
            <map extension="ico" type="image/x-icon"/>
            <map extension="pdf" type="application/pdf"/>
            <map extension="css" type="text/css; charset=utf-8"/>
            <map extension="xsl" type="text/xls; charset=utf-8"/>
            <map extension="xml" type="text/xml; charset=utf-8"/>
            <map extension="htm" type="text/html; charset=utf-8"/>
            <map extension="html" type="text/html; charset=utf-8"/>
            <map extension="js" type="application/x-javascript; charset=utf-8"/>
        </mimetype>
        +<virtual host="*" active="true">
    </instance>
</configuration>
```

MIME type directives in the virtual host section of the configuration file override MIME types specified in this section.

To determine allowed MIME types, the JSM HTTP server looks at MIME types in the virtual host section of the httpd configuration file and then at MIME types in the server instance. Place MIME types in the server instance that will apply all virtual hosts. Place MIME types that are unique to a virtual host in the MIME types section of the virtual host.

Table 19: What to Configure for the Server Instance MIME Types

To configure	Apply these settings	Change defaults
MIME types	Use the default list	Not recommended
Add MIME types	Add new a directive for each MIME type.	Optional

Virtual host configuration

The JSM HTTP server is capable of managing multiple virtual hosts. Table 20 (page 33) shows the virtual host configuration for DE.

Table 20: Virtual Host Configuration

Virtual Host Configuration Example

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
            index="index.html">
        <errorlog enabled="true" file="www/instance/logs/error.log"/>
        <accesslog enabled="true" file="www/instance/logs/access.log"/>
        <listen secure="false" store="pki/wwwssl.jks" password="password"
            port="5563"
            sslport="5564"
            interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
            nodelay="false" timeout="5"/>
        + <access>
        + <mimetype>
        <virtual host="*" active="true">
            + <access>
            + <protect>
            + <script>
            + <mimetype>
        </virtual>
    </instance>
</configuration>
```

The host is the name of the virtual host to match with the HTTP host property. In this case the asterisk ("*") indicates acceptance of requests from any host.

Table 21: What to Configure for the Virtual Host

To configure	Apply these settings	Change defaults
Virtual host	Use the default value asterisk (*)	Do not change
Active	Use the value true	Do not change

The active parameter has two values "true" and "false". The value of the active parameter must be "true" for DE to operate.

If you wish to use multiple instances and/or multiple virtual hosts in your configuration refer to the section titled, "Running DE and WSFM side by side" (page 56).

Access, protect, script and MIME type are sub sections of the virtual section in the httpd configuration file.

Virtual host access

The access directives in the virtual host section in the httpd configuration file control access to services provided by the JSM HTTP server. These directives override the access directives defined for the server instance. Using the virtual host access configuration you can:

- Allow addresses
- Deny addresses
- Allow user agents
- Deny user agents
- Allow content lengths
- Deny content lengths

Table 22 (page 34) shows an example of configuring virtual host access directives.

Table 22: Virtual Host Access Directives

Virtual Host Access Directives Configuration Example

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
            index="index.html">
        <errorlog enabled="true" file="www/instance/logs/error.log"/>
        <accesslog enabled="true" file="www/instance/logs/access.log"/>
        <listen secure="false" store="pki/wwwssl.jks" password="password"
            port="5563"
            sslport="5564"
            interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
            nodelay="false" timeout="5"/>
        +<access>
        +<mimetype>
        <virtual host="*" active="true">
            <access>
                <allow address="*"/>
                <deny useragent="webos"/>
                <deny useragent="android"/>
                <deny useragent="ipad"/>
                <deny useragent="iphone"/>
                <allow useragent="*"/>
                <allow useragent="?"/>
                <deny contentlength="4096"/> <!-- deny content GT value -->
            </access>
            +<protect>
            +<script>
```

Virtual Host Access Directives Configuration Example
+<mimetype> </virtual> </instance> </configuration>

In this example, requests from any address are acceptable, deny access is explicit for several user agents, other user agents are acceptable and content length greater than 4096 is unacceptable.

Table 23: What to Configure for Virtual Host Access

To configure	Apply these settings	Change defaults
Allow any address	Use allow address=*	Optional
Access for specific addresses	Add a directive for each address.	Optional
Allow any user agent	Use allow useragent=*	Optional
Allow specific user agents	Add a directive for each user agent	Optional
Allow content lengths	Use allow contentlength= "value"	Optional
Deny any address	Use deny address=*	Optional
Deny access for specific addresses	Add a directive for each address.	Optional
Deny any user agent	Use deny useragent=*	Optional
Deny specific user agents	Add a directive for each user agent	Optional
Deny content lengths	Use deny contentlength= "value"	Optional

Access directives in the virtual host section of the configuration file override access directives specified in the access section of the server instance.

Virtual host protect

Warning	You do not need to configure this section of the httpd configuration file to use DE.

The protection section of the httpd configuration file maps authentication methods to parts of the Web site or application. The realm describes the authentication method and the match URI associates the realm with the protected part of the Web site or application.

Table 24 (page 36) shows an example of the protect configuration.

Content © LANSA Group
All Rights Reserved

Table 24: Virtual Host Protect Configuration

Virtual Host Protect Configuration Example
<?xml version="1.0" encoding="UTF-8"?> <configuration> <instance name="HTTP Instance" active="true" root="www/instance/htdocs" index="index.html"> <errorlog enabled="true" file="www/instance/logs/error.log"/> <accesslog enabled="true" file="www/instance/logs/access.log"/> <listen secure="false" store="pki/wwwssl.jks" password="password" port="5563" sslport="5564" interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1" nodelay="false" timeout="5"/> +<access> +<mimetype> <virtual host="*" active="true"> +<access>
<protect> <realm name="Area 51"> <user name="user" access="bb644a9819425bfd8586b408896a1031"/> </realm> <match uri="/restricted" realm="Area 51" authentication="basic,digest"/> </protect>
+<script> +<mimetype> </virtual> </instance> </configuration>

In this example the realm is "Area 51", the user name is "user", and access is a hash of the user, password and realm information. The configuration uses both basic and digest authentication methods. The match uri = "/restricted" associates the realm with URIs including the match URI.

Table 25: Configuring Virtual Host Protect

To configure	Apply these settings	Change defaults
Realm	Define the realm name.	Do not change the default values.
Realm / user	Add one or more user names with their access hash into the realm. You need to generate an access hash for each user in each realm and include them in the configuration. This applies even when the same user is in different realms. The tool to generate the access hash is not supplied with this version of DE.	Do not change the default values.

To configure	Apply these settings	Change defaults
Match URIs	Add one or more full or partial URIs with their associated realms.	Do not change the default values.

You do not need to configure this section of the httpd configuration file to use DE.

Virtual host script

The script section of the httpd configuration file contains configuration items associated with the available services. DE uses the following services:

- Query service (HTTPServiceQuery)
- File service (JSMHTTPServiceFile)

Each service has one or more parameters that control the way the service operates. Examples of parameters are:

- Autoregister
- Allow.query.clause.into
- Database.library

Table 26 (page 37) shows an example of a script configuration.

Table 26: Virtual Host Script Configuration

Virtual Host Script Configuration Example

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
            index="index.html">
        <errorlog enabled="true" file="www/instance/logs/error.log"/>
        <accesslog enabled="true" file="www/instance/logs/access.log"/>
        <listen secure="false" store="pki/wwwssl.jks" password="password"
            port="5563"
            sslport="5564"
            interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
            nodelay="false" timeout="5"/>
        +<access>
        +<mimetype>
        <virtual host="*" active="true">
            +<access>
            +<protect>
            <script>
                <match uri="/axes/dbmservice.jsp"
                    class="com.lansa.jsm.service.HTTPServiceQuery"
                    trace="false" clienttrace="false">
                    <parameter name="autoregister" value="false"/>
                    <parameter name="allow.query.clause.into" value="false"/>
                    <parameter name="database.host" value="LOCALHOST/>"
                    <parameter name="database.library" value="AXES"/>
                </match>
                <match uri="/" class="com.lansa.jsm.JSMHTTPServiceFile"
                    trace="false" clienttrace="false">
```

Virtual Host Script Configuration Example

```
                <parameter name="cache.maxage" value="28800"/>
                <parameter name="cache.maxage.pdf" value="28800"/>
                <parameter name="cache.maxage.image" value="28800"/>
            </match>
        </script>
        +<mimetype>
    </virtual>
  </instance>
</configuration>
```

DE requires configuration for the query service, JSM HTTP file service to operate successfully. This guide includes individual sections that describe how to configure each service; see the sections under "Configure services", (from page 39).

Virtual host MIME types

Table 27 (page 38) shows the configuration for MIME types applicable to the virtual host.

Table 27: Virtual Host MIME Type Configuration

Virtual Host MIME Type Configuration Example

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
        index="index.html">
    <errorlog enabled="true" file="www/instance/logs/error.log"/>
    <accesslog enabled="true" file="www/instance/logs/access.log"/>
    <listen secure="false" store="pki/wwwssl.jks" password="password"
      port="5563"
      sslport="5564"
      interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
      nodelay="false" timeout="5"/>
    +<access>
    +<mimetype>
    <virtual host="*" active="true">
      +<access>
      +<protect>
      +<script>
      <mimetype>
        <map extension="pdf" type="application/pdf"/>
      </mimetype>
    </virtual>
  </instance>
</configuration>
```

The MIME types defined in this section of the httpd configuration file override MIME types defined for the server instance.

Table 28: What to Configure for Virtual Host MIME Types

To configure	Apply these settings	Change defaults
MIME types	Use the default list	Do not change
Add MIME types	Add new a directive for each MIME type.	Optional

Configure services

The configure services section provides explanations for configuring each of the services that support DE. The configuration items and parameters reside in the script section of the httpd configuration file.

JSM HTTP file service configuration

The JSM HTTP server uses the service titled, com.lansa.jsm.JSMHTTPServiceFile, to retrieve files associated with the requested page.

Table 29 (page 39) shows the configuration for this service.

Table 29: Configure JSM HTTP File Service

Configure JSM HTTP File Service

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
            index="index.html">
        <errorlog enabled="true" file="www/instance/logs/error.log"/>
        <accesslog enabled="true" file="www/instance/logs/access.log"/>
        <listen secure="false" store="pki/wwwssl.jks" password="password"
            port="5563"
            sslport="5564"
            interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
            nodelay="false" timeout="5"/>
        +<access>
        +<mimetype>
        <virtual host="*" active="true">
            +<access>
            +<protect>
            <script>
                +<match uri="/axes/dbmservice.jsp"
                <match uri="/" class="com.lansa.jsm.JSMHTTPServiceFile"
                    trace="false" clienttrace="false">
                    <parameter name="cache.maxage" value="28800"/>
                    <parameter name="cache.maxage.pdf" value="28800"/>
                    <parameter name="cache.maxage.image" value="28800"/>
                </match>
            </script>
            +<mimetype>
        </virtual>
```

Configure JSM HTTP File Service
</instance> </configuration>

In this example the match URI is set to "/" to indicate the match applies to all URIs. The cache parameters set the maximum period for caching files and images. In this case the value used for both parameters is 28,800 seconds (or 8 hours).

Table 30: What to Configure for JSM HTTP File Service

To configure	Apply these settings	Change defaults
Match URI	Use the default "/"	Do not change
Class	Use com.lansa.jsm.JSMHTTPServiceFile	Do not change
Cache Maxage	The default maximum age for cached files is 28,800 seconds (8 hours). Do not remove this parameter.	Optional
Cache Maxage Image	The default maximum age for cached image files is 28,800 seconds (8 hours). Do not remove this parameter.	Optional

Do not remove this service from the httpd configuration file.

Allow and deny user access to services

The parameters, service.user.allow and service.user.deny, define identifications (or profiles) that are permitted (or allowed) to use aXes or excluded from (or denied access to) DE services. The service.user.allow parameter defines acceptable users and the service.user.deny parameter defines users not permitted to use DE services. Table 31 (page 40) explains the syntax of these parameters. You may include multiple instances of the service.user.allow and service.user.deny parameters.

Table 31: User Identification/Profile service.user Allow and Deny Parameter Syntax

Parameter	Action by aXes
<parameter name= "service.user.allow" value= "UserId"/>	The parameter value defines allowed (or acceptable) user identifications.
<parameter name= "service.user.allow" value= "UserId,UserId,UserId,UserId"/>	The parameter value is a list of user identifications separated by commas. The service.user.allow parameter allows (or accepts) user identifications in the list.
<parameter name= "service.user.deny" value= "UserId"/>	The parameter value denies (excludes) access to one user.
<parameter name= "service.user.deny" value= "UserId,UserId,UserId,UserId"/>	The parameter value is a list of user identifications separated by commas. The service.user.deny parameter denies (excludes) access to user identifications in the list.
<parameter name= "service.user.allow" value= "*USER"/>	The value *USER is a special case. It is a collective value that allows all user identifications.

Parameter	Action by aXes
<parameter name= "service.user.deny" value= "*USER"/>	The value *USER is a special case. It is a collective value that denies all user identifications.

The user authentication process looks for instances of the service.user.deny parameter and then instances of the service.user.allow parameter.

By default, DE denies access to user identifications beginning with the letter Q. To enable access for these user identifications, each user identification must be configured explicitly using a service.user.allow parameter.

Table 32 (page 41) illustrates a configuration that includes the service.user.allow and service.user.deny parameters.

Table 32: Example Configuration for service.user Allow and Deny Parameters

Configuration for service.user.allow and service.user.deny Parameters

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
            index="index.html">
        <errorlog enabled="true" file="www/instance/logs/error.log"/>
        <accesslog enabled="true" file="www/instance/logs/access.log"/>
        <listen secure="false" store="pki/wwwssl.jks" password="password"
            port="5563"
            sslport="5564"
            interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
            nodelay="false" timeout="5"/>
        +<access>
        +<mimetype>
        <virtual host="*" active="true">
            +<access>
            +<protect>
            <script>
                <match uri="/axes/dbmservice.jsp"
                    class="com.lansa.jsm.service.HTTPServiceQuery"
                    trace="false" clienttrace="false">
                    <parameter name="service.user.deny" value="JohnS,MaryB"/>
                    <parameter name="service.user.deny" value="GeorgeD,WendyF"/>
                    <parameter name="service.user.allow" value="*USER"/>
                    <parameter name="service.user.allow" value="QSYSOPR "/>
                    <parameter name="autoregister" value="false"/>
                    <parameter name="allow.query.clause.into" value="false"/>
                    <parameter name="database.host" value="LOCALHOST"/>
                    <parameter name="database.library" value="AXES"/>
                </match>
                +<match uri="/" class="com.lansa.jsm.JSMHTTPServiceFile"
            </script>
            +<mimetype>
```

Configuration for service.user.allow and service.user.deny Parameters
 </virtual> </instance> </configuration>

Table 33 (page 42) provides examples and explanations for configuring the service.user.allow and service.user.deny parameters.

Table 33: Controlling Access to Services – service.user Allow and Deny Examples

Parameter	Explanation
<parameter name= "service.user.allow" value= "*USER"/>	This value allows every user identification, except for those beginning with the letter Q.
<parameter name= "service.user.deny" value= "*USER"/>	This value denies access to every user identification, including those beginning with the letter Q. Using the *USER value on a service.user.deny parameter locks out every user. It overrides all values for the service.user.allow parameter.
<parameter name= "service.user.allow" value= "JohnS,MaryB,GeorgeD,WendyF"/>	This configuration allows all user identifications in the list.
<parameter name="service.user.deny" value= "JohnS,MaryB"/> <parameter name="service.user.allow" value= "*USER"/>	This configuration denies access to the users JohnS and MaryB, but allows all other users, except those beginning with the letter Q. This example illustrates the optimum method for allowing most users and denying a small number of users.
<parameter name="service.user.allow" value= "JohnS,MaryB"/>	This configuration allows only users JohnS and MaryB.
<parameter name="service.user.allow" value= "*USER,QSECOFR,QSYSOPR"/>	This service.user.allow value allows all users and both QSECOFR and QSYSOPR. It is unnecessary to include service.user.deny parameters for user identifications beginning with the letter Q; aXes denies access to these user identifications by default.

To use DE services, users must be authorised by an implicit or explicit service.user.allow parameter and also registered as a DE user.

User registration

User registration is one of the tasks managed by the service dbmservice.jsp. Table 34 (page 43) describes the values for the autoregister parameter associated with the service dbmservice.jsp.

Table 34: Autoregister Parameter for the Service axes/dbmservice.jsp

Parameter	Action by DE
<parameter name= "autoregister" value= "true"/>	DE will register any person who logs on with an active profile on the server.
<parameter name= "autoregister" value= "false"/>	Setting the value to false prevents DE registering users automatically. Administrators will register users manually.

For manual user registration change the value of the autoregister parameter to "false".

For automated user registration change the value of the autoregister parameter to "true".

Table 35 (page 43) shows an example of a configuration file using manual user registration.

Table 35: Autoregister Parameter for Automated or Manual User Registration

Configuration for the Auto Register Parameter

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
            index="index.html">
        <errorlog enabled="true" file="www/instance/logs/error.log"/>
        <accesslog enabled="true" file="www/instance/logs/access.log"/>
        <listen secure="false" store="pki/wwwssl.jks" password="password"
            port="5563"
            sslport="5564"
            interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
            nodelay="false" timeout="5"/>
        +<access>
        +<mimetype>
        <virtual host="*" active="true">
            +<access>
            +<protect>
            <script>
                <match uri="/axes/dbmservice.jsp"
                    class="com.lansa.jsm.service.HTTPServiceQuery"
                    trace="false" clienttrace="false">
                    <parameter name="autoregister" value="false"/>
                    <parameter name="allow.query.clause.into" value="false"/>
                    <parameter name="database.host" value="LOCALHOST"/>
                    <parameter name="database.library" value="AXES"/>
                </match>
                +<match uri="/" class="com.lansa.jsm.JSMHTTPServiceFile"
            </script>
            +<mimetype>
        </virtual>
    </instance>
</configuration>
```

Table 35 (page 43) shows an example httpd configuration file <script> section and the configuration of the dbmservice. In this example the value of the autoregister parameter is "false", indicating manual user registration.

Users cannot set the value of the autoregister parameter from the DE user interface.

SQL policies

The configuration items in this section determine the SQL clauses that users may include in their queries.

Allowing or prohibiting SELECT INTO

DE provides query or read-only access to tables provided the user has read access to the tables. All queries use the SELECT clause but using the SELECT INTO clause is optional. Allowing or prohibiting its use is set by the parameter, "allow.query.clause.into" in the httpd configuration file. Table 36 (page 44) explains the options for the "allow.query.clause.into" parameter.

Table 36: Configuring Use of the SELECT INTO Clause

Parameter	Action by DE
<parameter name= "allow.query.clause.into" value= "true"/>	DE allows the SELECT INTO clause in queries
<parameter name= "allow.query.clause.into" value= "false"/>	DE prohibits the SELECT INTO clause in queries "False" is the default value.

To allow SELECT INTO change the value of allow.query.clause.into to "true".

To prohibit SELECT INTO change the value of allow.query.clause.into to "false".

Table 37: Allowing or Prohibiting SELECT INTO in SQL Queries

Configuration for the Query Clause INTO Parameter: Prohibit Example

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
            index="index.html">
        <errorlog enabled="true" file="www/instance/logs/error.log"/>
        <accesslog enabled="true" file="www/instance/logs/access.log"/>
        <listen secure="false" store="pki/wwwssl.jks" password="password"
            port="5563"
            sslport="5564"
            interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
            nodelay="false" timeout="5"/>
        +<access>
        +<mimetype>
        <virtual host="*" active="true">
            +<access>
            +<protect>
            <script>
                <match uri="/axes/dbmservice.jsp"
                    class="com.lansa.jsm.service.HTTPServiceQuery"
                    trace="false" clienttrace="false">
```

Configuration for the Query Clause INTO Parameter: Prohibit Example

```
                <parameter name="autoregister" value="false"/>
                <parameter name="allow.query.clause.into" value="false"/>
                <parameter name="database.host" value="LOCALHOST"/>
                <parameter name="database.library" value="AXES"/>
            </match>
            +<match uri="/" class="com.lansa.jsm.JSMHTTPServiceFile"
        </script>
        +<mimetype>
    </virtual>
  </instance>
</configuration>
```

Table 37 (page 44) shows an example httpd configuration file <script> section. In this example the value of the allow.query.clause.into parameter is "false".

Users cannot set the value of the allow.query.clause.into parameter from the DE user interface.

Reference guide: JSM HTTP server configuration

The reference guide provides explanations of the individual items in the httpd configuration files.

Configuration item reference

Table 38 (page 45) provides explanations of configuration items in the JSM HTTP server httpd configuration file. The items in the table are in name sequence. The configuration item name is the name and position of the parameters in the httpd configuration file. For example, the item named "listen port" is the port on the listen directive.

Table 38: Server Reference: Configuration Item Reference

Configuration Items	Definitions
Access allow	List of addresses, content lengths and/or user agents allowed access to the JSM HTTP server. Example: <allow address="10.2.1.45">
Access deny	List of addresses, content lengths and/or user agents denied access to the JSM HTTP server. Example: <deny address="10.2.1.45">
Access log enabled	DE logs all access events when access log enabled is "true". DE logs no access events when access log enabled is "false".
Access log file	Name and location of the access log file. Example: "www/instance/logs/access.log"
Cache maxage	Maximum age for cached files in seconds.
Cache maxage image	Maximum age for cached image files in seconds.
Cache maxage pdf	Maximum age for cached PDF files in seconds.

Configuration Items	Definitions
Error log enabled	DE logs all errors when error log enabled is "true". DE logs no errors when error log enabled is "false".
Error log file	Name and location of the error log file. Example: "www/instance/logs/error.log"
Instance active	To operate DE this parameter will always be "true". The JSM HTTP server supports multiple instances and needs this parameter to indicate which instances to activate at run time.
Instance index	Name of the index page. Example: "index.html"
Instance name	Name allocated the JSM HTTP server. Example: "HTTP Instance" This name does not need to be the same as the instance name in the manager.properties file (httpd.axes.instance=AXES).
Instance root	Name of the directory from which the documents will be served by the JSM HTTP server. Also known as document root. Example: "www/instance/htdocs"
Listen backlog	The backlog defines the depth of the TCP/IP queue.
Listen buffer receive	TCP/IP receive buffer size in bytes. Special case value "-1" means use operating system default.
Listen buffer send	TCP/IP send buffer size in bytes. Special case value "-1" means use operating system default.
Listen interface	TCP/IP interface address that the JSM HTTP server will bind to and accept connections on. Default value is *ALL *ALL will bind to all interfaces on the server.
Listen no delay	True enables TCP/IP no delay option. False means use the operating system setting for this parameter.
Listen password	Password that will open the store file used for TLS/SSL configuration. Example: "password"
Listen port	TCP/IP port number the server will use to accept connections on.
Listen secure	When true, the JSM HTTP server uses TLS/SSL When false, the JSM HTTP server uses plain sockets.

Content © LANSA Group
All Rights Reserved

Configuration Items	Definitions
Listen sslport	TCP/IP port number the server will use to accept connections when using TLS/SSL.
Listen store	Path for the store file that contains the private key and public certificates (used when TLS/SSL enabled). Fully qualified name and location of the store file. Example: "pki/wwwssl.jks"
Listen timeout	Time out count in seconds (integer).
Mimetype map extension	File extension used to identify the MIME type. Example: "png"
Mimetype map type	The type describes the nature of the MIME type. Examples: "image/png"
Protect match authentication	Type of authentication, values are: "basic", "digest" or "basic,digest". Basic: Basic authentication is a concatenation of user name, a colon and the password encoded with the Base64 algorithm. Digest: Digest authentication is an application of MD5 cryptographic hashing of user credentials. It provides stronger encoding than basic authentication.
Protect match realm	Name of the realm used for authentication to parts of the Web site or application. The realm is associated with the matched URIs.
Protect match uri	URIs used to match against requests. When a match occurs the associated realm provides the authentication details.
Protect realm name	Name assigned to a realm.
Protect realm user access	Access is a digest of user information (including the password).
Protect realm user name	User name for authentication.
Script match	The match definitions describe services that the JSM HTTP server will use depending on the match criteria in the URI.
Script match class	Class is the name of a service.
Script match client trace	Use this parameter to trace activity associated with specific clients. Tracing will slow performance. When "true" tracing will occur from the client URI using the "?trace=true" query string parameter. The value "false" disables tracing.

Content © LANSA Group
All Rights Reserved

Configuration Items	Definitions
Script match trace	Use this parameter to trace activity associated with all clients. Broader scope than client trace. The value "true" enables tracing for all services and user agents that match the match URI. The value "false" disables tracing.
Script match URI	The match URI is the match criteria the JSM HTTP server uses to determine the services to use.
Virtual access allow	List of addresses, content lengths and/or user agents allowed access to the JSM HTTP server. Example: <allow useragent="safari">
Virtual access deny	List of addresses, content lengths and/or user agents denied access to the JSM HTTP server. Example: <deny useragent="safari">>
Virtual active	When the value is "true" this virtual host is active. When the value is "false" the virtual host is inactive. To operate DE the value must be "true". Services configured in the virtual host are unavailable when the virtual host is inactive.
Virtual host	Name of the virtual host to match with the HTTP host property. This allows multi homing. HTTP requests can be directed to different virtual host sections of the configuration in the server instance. If a virtual host is not found then the connection request is rejected. The special case value "*" accepts requests from any HTTP host. Specific names take precedence.
Virtual mimetype map extension	File extension used to identify the MIME type. Example: "png"
Virtual mimetype map type	The type describes the nature of the MIME type. Examples: "image/png"

MIME types

MIME type describes the nature of content of file for the JSM HTTP server. Table 39 (page 48) provides examples of MIME types.

Table 39: JSM HTTP Server Reference: MIME type Examples

Extension	Type
css	text/css; charset=utf-8
gif	image/gif
htm	text/html; charset=utf-8
html	text/html; charset=utf-8

Content © LANSA Group
All Rights Reserved

Extension	Type
ico	image/x-icon
jpeg	image/jpeg
jpg	image/jpeg
js	application/x-javascript; charset=utf-8
pdf	application/pdf
png	image/png
tiff	image/tiff
xls	text/xls; charset=utf-8
xml	text/xml; charset=utf-8

Access allow and deny directives

Table 40 (page 49) shows examples of the access allow/deny directive for addresses.

Table 40: Server Reference: Access Allow and Deny Addresses

Allow / Deny	Syntax and Examples
Allow any address	<allow address="*">
Allow addresses in a range	<allow address="10.2.1">
Allow a specific address	<allow address="10.2.1.45">
Deny any address	<deny address="*">
Deny addresses in a range	<deny address="10.2.1">
Deny a specific address	<deny address="10.2.1.45">

Table 41 (page 49) shows examples of the access allow/deny directive for content length.

Table 41: Server Reference: Access Allow and Deny Content Length

Allow / Deny	Syntax and Examples
Allow access for content less than or equal to the specified length	<allow contentlength="4096">
Zero content length is a special case to allow access for no content connections from the browser	<allow contentlength="0">
Deny access for content greater than the specified length	<deny contentlength="4096">

User agents are applications or services that act on behalf of the user. When a user requests a web page (or URL), the browser acts as a user agent by sending the page request to the JSM HTTP server. Examples of user agents are browsers, web crawlers, link checkers, bots and email clients. Access allow and deny directives control which user agents the JSM HTTP server will allow or deny access.

Table 42 (page 50) shows examples of user agents and the syntax of the allow access and deny access directives.

Content © LANSA Group
All Rights Reserved

Table 42: Server Reference: Access Allow and Deny User Agents

Allow / Deny	Syntax and Examples
Allow access for any user agent	<allow useragent="*"
Allow access if no user agent provided	<allow useragent="?"
Allow access to Chrome	<allow useragent="chrome"
Allow access to the Internet Explorer	<allow useragent="explorer"
Allow access to Firefox	<allow useragent="firefox"
Allow access to Safari	<allow useragent="safari"
Deny access for any user agent	<deny useragent="*"
Deny access if no user agent provided	<deny useragent="?"
Deny access to Chrome	<deny useragent="chrome"
Deny access to Internet Explorer	<deny useragent="explorer"
Deny access to Firefox	<deny useragent="firefox"
Deny access to Safari	<deny useragent="safari"

The evaluation of the directives starts with the first item in the list and continues until it finds a true condition. Any combinations of address, user agent and content length are acceptable. However, it is possible to negate the effect of a directive by its position in the list. For example, placing an allow any user agent (<allow useragent="*"/>) ahead of a deny for a specific user agent (<deny useragent="webos"/>) will cause the JSM HTTP server to ignore the deny directive.

Table 43 (page 50) presents lists of user agents.

Table 43: Server Reference: Sample Lists of User Agents

Browser User Agents
android
chrome
explorer
firefox
opera
safari
webos

Bots and Device User Agents
googlebot
googletoolbar
ipad
iphone
lansaua
msnbot
yahoobot

Sample configurations

Default configuration for DE

Table 44 (page 51) shows the default httpd configuration file including MIME types, allow and deny directives.

Table 44: Server Reference: Default httpd Configuration for DE

Default Configuration for DE

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <instance name="HTTP Instance" active="true" root="www/instance/htdocs"
            index="index.html">
        <errorlog enabled="true" file="www/instance/logs/error.log"/>
        <accesslog enabled="true" file="www/instance/logs/access.log"/>
        <listen secure="false" store="pki/wwwssl.jks" password="password"
            port="5563"
            sslport="5564"
            interface="*ALL" backlog="256" buffersend="-1" bufferreceive="-1"
            nodelay="false" timeout="5"/>
        <access>
            <allow address="*"/>
            -->
        </access>
        <mimetype>
            <map extension="png"  type="image/png"/>
            <map extension="gif"  type="image/gif"/>
            <map extension="jpg"  type="image/jpeg"/>
            <map extension="jpeg" type="image/jpeg"/>
            <map extension="tiff" type="image/tiff"/>
            <map extension="ico"  type="image/x-icon"/>
            <map extension="pdf"  type="application/pdf"/>
            <map extension="css"  type="text/css; charset=utf-8"/>
            <map extension="xsl"  type="text/xls; charset=utf-8"/>
            <map extension="xml"  type="text/xml; charset=utf-8"/>
            <map extension="htm"  type="text/html; charset=utf-8"/>
            <map extension="html" type="text/html; charset=utf-8"/>
            <map extension="js"   type="application/x-javascript; charset=utf-8"/>
        </mimetype>
        <virtual host="*" active="true">
            <access>
                <deny useragent="webos"/>
                <deny useragent="opera"/>
                <deny useragent="android"/>
                <deny useragent="ipod"/>
                <allow useragent="*"/>
                <allow useragent="?"/>
            </access>
            <protect>
                <realm name="Area 51">
                    <user name="user" access="bb644a9819425bfd8586b408896a1031"/>
                </realm>
                <match uri="/restricted" realm="Area 51" authentication="basic,digest"/>
```

Default Configuration for DE

```
            </protect>
            <script>
                <match uri="/axes/dbmservice.jsp"
                    class="com.lansa.jsm.service.HTTPServiceQuery"
                    trace="false" clienttrace="false">
                    <parameter name="autoregister" value="false"/>
                    <parameter name="allow.query.clause.into" value="false"/>
                    <parameter name="database.host" value="LOCALHOST"/>
                    <parameter name="database.library" value="AXES"/>
                </match>
                <match uri="/" class="com.lansa.jsm.JSMHTTPServiceFile"
                    trace="false" clienttrace="false">
                    <parameter name="cache.maxage" value="28800"/>
                    <parameter name="cache.maxage.pdf" value="28800"/>
                    <parameter name="cache.maxage.image" value="28800"/>
                </match>
            </script>
            <mimetype>
                <map extension="pdf" type="application/pdf"/>
            </mimetype>
        </virtual>
    </instance>
</configuration>
```

Content © LANSA Group
All Rights Reserved

Appendices

Glossary

Table 45 (page 53) presents definitions for abbreviations and terms used in this guide.

Table 45: Glossary of Abbreviations and Terms

Abbreviations and Terms	Definitions and Explanations
CCSID	Coded Character Set Identifier
Data selection criteria	The term "data selection criteria" refers to the SQL defined in a query filter.
DBCS	Double Byte Character Set
DE	Data Explorer
Directives	Configuration directives are the parameters and settings that control the behaviour of the JSM HTTP server.
Files	IBM i servers to refer to data residing on disks as physical files. SQL uses the word table.
Filter	DE uses the word "filter" to describe a collection of search criteria when searching for tables and selection criteria for queries.
IFS	The Integrated File System is a part of the IBM i operating system. It supports stream input/output and storage management capabilities like personal computer and UNIX operating systems.
Internet Media Type	Internet Media Type (IMT) is an identifier for file formats on the Internet. An example is the IMT for the Portable Document Format: application/pdf See also MIME type.
JSM	Java Services Manager
MIME	Multipurpose Internet Mail Extensions MIME is an Internet standard that describes content. An example is PNG for Portable Network Graphic files. See also Internet Media Types.
Queries	Queries describe the selection criteria inherent in SQL statements.
Query data	Query data refers to data sets produced by running query filters. A data set contains data that matches the selection criteria specified in the query filter. The query data consists of data retrieved from database tables.

Content © LANSA Group
All Rights Reserved

Abbreviations and Terms	Definitions and Explanations
Realm	A realm is a database containing a list of valid users of a web application. The user information in the database is user name, password and a set of roles associated with the user.
SBCS	Single Byte Character Set
Search criteria	The search criteria determine what DE looks for when searching for tables. Suppose you want a list of tables in the QGPL library, the search criteria is a specific library and any table. You express the search criteria as library "QGPL" and table names "*ALL".
Selection criteria	The selection criteria determine the rules used by queries when retrieving data from tables. DE uses SQL to express the selection criteria.
SQL	Structured Query Language (SQL) is the language used to interrogate relational databases. An example of an SQL query is: SELECT surname, first_name, start_date, annual_salary FROM employees WHERE annual_salary < 50000 ORDER BY start_date This SQL selects employees whose annual salary is less than $50,000 and sequences the query data by the date the employee started with the company. The query data consists of surname, first name, start date and annual salary.
System directory	System directory is a synonym for system folder.
System folder	The location of the system folder is: axes/jsm/instance/system
Table	IBM i servers to refer to data residing on disks as physical files. SQL uses the word table.
Toolbox for Java JDBC driver	The Toolbox for Java JDBC driver provides access to the database.
WSFM	Web Spooled File Manager WSFM is a product related to DE and may share the same httpd configuration file.

Referenced documents and sources

Table 46 (page 54) defines documents and information sources referenced in this guide.

Table 46: Referenced Documents and Information Sources

References
IBM i and System i Information Center at http://publib.boulder.ibm.com/iseries/
IBM Toolbox for Java (IBM i software) at http://www-03.ibm.com/systems/i/software/toolbox/faqjdbc.html#faqA1

Assumed and prerequisite knowledge

This guide provides no tutorial for creating databases or SQL. Table 47 (page 55) defines the prerequisite knowledge needed to use the guide.

Table 47: Assumed and Prerequisite Knowledge

Subject Matter	Explanations
Libraries	The guide assumes that you understand the concepts behind libraries for IBM i servers.
Query concepts	The guide assumes that you understand the concepts behind querying tables in databases on IBM i servers.
SQL	The guide assumes that you understand how to create SQL, especially for cases when you need complex SQL (examples are joining tables and complicated WHERE clauses).
Tables and files	The guide assumes that you understand the concepts behind tables (or files) on IBM i servers, including physical files, logical files and views.
Using the clipboard	You need to know how to use the clipboard for copying and pasting data between applications.
XML files	The directives for DE reside in the configuration file, formatted as XML. You need to be able to read and edit the XML.

Browser behaviour differences

Every browser is different. Browser manufacturers implement components such as JavaScript, DOM, HTML, CSS, event handling and network connectivity in different ways. Therefore, DE will vary in its appearance and behaviour depending on the browser you use. Examples of these differences are as follows.

- Slight variations in the way the pages look.
- Differences in behaviour.
  For example, some browsers stop running animated GIF files after a short period.
- Network activity will vary.
  Some browsers will wait longer for responses.

Browser behaviour can change from one version to the next. Each browser is on a different enhancement timeline. Some vendors implement new features and bug fixes quickly.

Performance differences will be more noticeable when you run spooled file filters or output queue filters that take a long time to finish.

Workstation and device requirements

This section describes prerequisites and system requirements for running DE on your computer or mobile device.

Hardware

Devices with faster processors and more memory will provide better performance.

DE requires no disk space other than the cache used by the browser.

Content © LANSA Group
All Rights Reserved

Software

To use DE you need one of the supported browsers and apart from the operating system no other software is necessary.

Connectivity

To use WSFM you need a TCP/IP connection to an IBM i server from an internal network or the Internet.

Running DE and WSFM side by side

Web Spooled File Manager (WSFM) may be installed beside DE and share its configuration with DE.

Several configuration options are available when installing both DE and WSFM. The choice of which configuration to use depends on the user communities who will use each product. You need the least access separation when the same user communities will use both products. You need the most access separation when different user communities will use each product. For example, you may allow internal user communities to use DE but external user communities may use only WSFM. When all user communities are internal the need for separate access reduces.

Configuration options

The simplest configuration is one instance and one virtual host for both DE and WSFM. When both products are configured in the same instance they share the same port.

Figure 6: DE and WSFM on a Single Instance and One Virtual Host

Characteristics of the configuration

Single instance (both DE and WSFM)

One virtual host (both DE and WSFM)

DE and WSFM share the same port

Simplest configuration

Fewest options for separating access to DE and WSFM

Suitable for deployment when the same user communities will operate both WSFM and DE.

You cannot have WSFM active and close down DE or keep DE active and close down WSFM with a configuration consisting of a single instance and one virtual host.

With both products installed, the virtual host script section of the httpd configuration file may include configuration items for both DE and WSFM. Alternatively, you could use two virtual hosts, one virtual host for DE and a second virtual host for WSFM.

Content © LANSA Group
All Rights Reserved

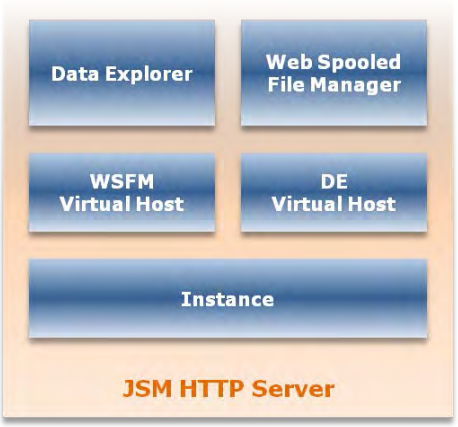


Figure 7: DE and WSFM on a Single Instance and Two Virtual Hosts

Characteristics of the configuration

Single instance (both DE and WSFM)

Virtual host for DE

Virtual host for WSFM

DE and WSFM share the same port

More effort to configure than one virtual host

More options for separating access to DE and WSFM than one virtual host

Suitable for deployment when one port is adequate but you want flexibility in separating access to DE and WSFM

Separate virtual hosts for DE and WSFM provide more flexibility for deployment than a single virtual host as the products share fewer configuration items and the configuration allows you to close down DE and leave WSFM active or close down WSFM and leave DE active.

The best separation is to use one port for DE and a second port for WSFM. To use different ports the httpd configuration file must include two instances, one instance for DE and a second instance for WSFM.

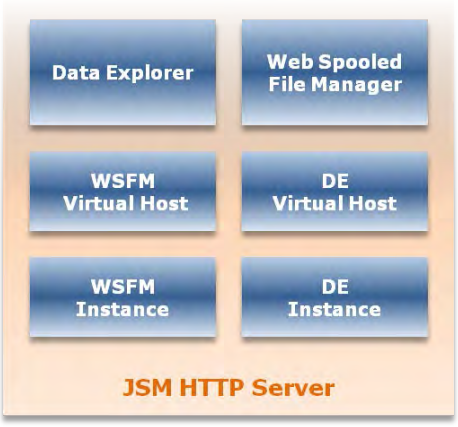


Figure 8: DE and WSFM on Two Instances and Two Virtual Hosts

Characteristics of the configuration

An instance for DE

A virtual host for DE

An instance for WSFM

A virtual host for WSFM

DE has a port

WSFM has a port

Most complex configuration

Best separation of access to DE and WSFM

Suitable for deployment when different user communities will operate DE and WSFM

Deploying two instances provides separate virtual hosts as a virtual host will reside in each instance. This configuration allows you to close down DE and leave WSFM active or close down WSFM and leave DE active.

User registration when operating both DE and WSFM

The user registration for DE and WSFM are discrete processes. Any user who wishes to operate both DE and WSFM must register for each product.

The separated registration ensures that administrators do not provide access to corporate databases indiscriminately.

Content © LANSA Group
All Rights Reserved

Configuration files

axinfo.json

Table 48: axinfo.json Configuration File for both DE and WS

axinfo.json Configuration Example both DE and WS

```
{
    "sfm":
        {
            "enabled":    true,
            "port":       5563,
            "sslport":    5564,
            "uri":        "/axes/spooled-file-manager.html"
        },
    "dbm":
        {
            "enabled":    true,
            "port":       5563,
            "sslport":    5564,
            "uri":        "/axes/database-manager.html"
        }
}
```