# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>FME Server Admin</td>
<td>1.1</td>
</tr>
<tr>
<td>About This Document</td>
<td>1.2</td>
</tr>
<tr>
<td>Course Overview</td>
<td>1.2.1</td>
</tr>
<tr>
<td>Course Resources</td>
<td>1.2.2</td>
</tr>
<tr>
<td>Chapter 1: Planning an FME Server Installation</td>
<td>1.3</td>
</tr>
<tr>
<td>Requirements for FME Workspaces</td>
<td>1.3.1</td>
</tr>
<tr>
<td>Installation Types</td>
<td>1.3.2</td>
</tr>
<tr>
<td>FME Server Architecture</td>
<td>1.3.3</td>
</tr>
<tr>
<td>Planning for Fault Tolerance</td>
<td>1.3.4</td>
</tr>
<tr>
<td>Disaster Recovery</td>
<td>1.3.5</td>
</tr>
<tr>
<td>Security Updates</td>
<td>1.3.6</td>
</tr>
<tr>
<td>Licensing FME Server</td>
<td>1.3.7</td>
</tr>
<tr>
<td>Chapter Review</td>
<td>1.3.8</td>
</tr>
<tr>
<td>Chapter 2: FME Server Connectivity</td>
<td>1.4</td>
</tr>
<tr>
<td>Firewalls and Ports</td>
<td>1.4.1</td>
</tr>
<tr>
<td>DNS</td>
<td>1.4.2</td>
</tr>
<tr>
<td>CORS</td>
<td>1.4.3</td>
</tr>
<tr>
<td>Chapter Review</td>
<td>1.4.4</td>
</tr>
<tr>
<td>Chapter 3: FME Server Security</td>
<td>1.5</td>
</tr>
<tr>
<td>Role- and User-Based Access</td>
<td>1.5.1</td>
</tr>
<tr>
<td>Security Policies</td>
<td>1.5.2</td>
</tr>
<tr>
<td>Exercise 1: Creating A New User</td>
<td>1.5.3</td>
</tr>
<tr>
<td>Running Services without Authentication</td>
<td>1.5.4</td>
</tr>
<tr>
<td>Running FME System Services under Different Accounts</td>
<td>1.5.5</td>
</tr>
<tr>
<td>Active Directory and FME Server</td>
<td>1.5.6</td>
</tr>
<tr>
<td>Exercise 2: Configuring Active Directory/LDAP</td>
<td>1.5.7</td>
</tr>
<tr>
<td>HTTPS/SSL and FME Server</td>
<td>1.5.8</td>
</tr>
<tr>
<td>Exercise 3: Configuring for HTTPS</td>
<td>1.5.9</td>
</tr>
<tr>
<td>Chapter Review</td>
<td>1.5.10</td>
</tr>
<tr>
<td>Q&amp;A Answers</td>
<td>1.5.11</td>
</tr>
<tr>
<td>Chapter 4: Scalability and Performance</td>
<td>1.6</td>
</tr>
<tr>
<td>Job Queues</td>
<td>1.6.1</td>
</tr>
<tr>
<td>Exercise 1: Job Queues</td>
<td>1.6.2</td>
</tr>
<tr>
<td>Adding FME Engines on a Separate Machine</td>
<td>1.6.3</td>
</tr>
<tr>
<td>Changing Database Provider for FME Server Database</td>
<td>1.6.4</td>
</tr>
<tr>
<td>Exercise 2: Changing the FME Server Database</td>
<td>1.6.5</td>
</tr>
<tr>
<td>System Cleanup</td>
<td>1.6.6</td>
</tr>
</tbody>
</table>
FME Server Administration Training Manual

This is the manual for the training course System Administration for FME Server.

The training will assist you to install and administer an FME Server installation and its users.

Course Structure

The full course is made up of seven sections. These sections are:

- Planning an FME Server Installation
- FME Server Connectivity
- FME Server Security
- Scalability and Performance
- FME Server Customization
- Migration and Upgrades
- Troubleshooting

Current Status

The current status of this manual is: **COMPLETE**. this manual can be used for training, subject to minor, last-minute fixes and creation of slides.

This manual applies to **FME2018.0**

The status of each chapter is:

- Chapter 0: Complete content. No exercises
- Chapter 1: Complete content. No exercises
- Chapter 2: Complete content. No exercises
- Chapter 3: Complete content and exercises
- Chapter 4: Complete content and exercises
- Chapter 5: Complete content and exercises
- Chapter 6: Complete content and exercises
- Chapter 7: Complete content. No exercises
- Chapter 8: Complete content. No exercises
- Slides: Incomplete
- FMEData: Complete
- Course Outline: Incomplete

**NB:** *Even for completed content, Safe Software Inc. assumes no responsibility for any errors in this document or their consequences, and reserves the right to make improvements and changes to this document without notice. See the full licensing agreement for further details.*
About This Document

This is the manual for the training course System Administration for FME Server.

Look out for residents of the City of Interopolis, who will appear from time-to-time to give you advice and dispense FME-related wisdom. In fact, here comes someone now:

**Mr. E.Dict (Attorney of FME Law) says...**

*On behalf of the City of Interopolis, welcome to this training course. Here is the standard legal information about this training document and the datasets used during the course. Be sure to read it, particularly if you’re thinking about re-using or modifying this content.*

**Licensing and Warranty**

Permission is hereby granted to use, modify and distribute the FME Tutorials and related data and documentation (collectively, the “Tutorials”), subject to the following restrictions:

1. The origin of the Tutorials and any associated FME® software must not be misrepresented.

2. Redistributions in original or modified form must include Safe Software's copyright notice and any applicable Data Source(s) notices.

3. You may not suggest that any modified version of the Tutorials is endorsed or approved by Safe Software Inc.

4. Redistributions in original or modified form must include a disclaimer similar to that below which: (a) states that the Tutorials are provided “as-is”; (b) disclaims any warranties; and (c) waives any liability claims.

Safe Software Inc. makes no warranty either expressed or implied, including, but not limited to, any implied warranties of merchantability, non-infringement, or fitness for a particular purpose regarding these Tutorials, and makes such Tutorials available solely on an “as-is” basis. In no event shall Safe Software Inc. be liable to anyone for direct, indirect, special, collateral, incidental, or consequential damages in connection with or arising out of the use, modification or distribution of these Tutorials.

This manual describes the functionality and use of the software at the time of publication. The software described herein, and the descriptions themselves, are subject to change without notice.

**Data Sources**
City of Vancouver

Unless otherwise stated, the data used here originates from open data made available by the City of Vancouver, British Columbia. It contains information licensed under the Open Government License - Vancouver.

Others

Forward Sortation Areas: Statistics Canada, 2011 Census Digital Boundary Files, 2013. Reproduced and distributed on an "as is" basis with the permission of Statistics Canada. © This data includes information copied with permission from Canada Post Corporation.

Digital Elevation Model: GeoBase®

Fire Hall Data: Some attribute data adapted from content © 2013 by Wikipedia, used under a Creative Commons Attribution-ShareAlike license

Stanley Park GPS Trail: Used with kind permission of VancouverTrails.com.

OpenStreetMap Datasets: © OpenStreetMap contributors. See www.openstreetmap.org/copyright.

Copyright

© 2005–2018 Safe Software Inc. All rights are reserved.

Revisions

Every effort has been made to ensure the accuracy of this document. Safe Software Inc. regrets any errors and omissions that may occur and would appreciate being informed of any errors found. Safe Software Inc. will correct any such errors and omissions in a subsequent version, as feasible. Please contact us at:

Safe Software Inc.
Phone: 604-501-9985
Fax: 604-501-9965
Email: train@safe.com
Web: www.safe.com

Safe Software Inc. assumes no responsibility for any errors in this document or their consequences and reserves the right to make improvements and changes to this document without notice.

Trademarks

FME® is a registered trademark of Safe Software Inc. All brand or product names are trademarks or registered trademarks of their respective companies or organizations.

Document Information

Document Name: FME Server Administration Training Manual 2018.0

All screenshots relate to FME Desktop and FME Server 2018.0; This manual has been tested with FME Desktop and FME Server 2018.0 Build 18310.
What's New?

A list of changes to this manual and its accompanying datasets can be found on GitHub at https://github.com/safesoftware/FMETraining/blob/Server-Admin-2018/WhatsNew.md. The file includes a list of general revisions compared to the previous year's materials. It is designed to help trainers become up-to-speed with new content, and for students to identify which FME functionality is new for the current release.
Course Overview

This training material covers how to install and administer FME Server.

The training will introduce the techniques and information required to efficiently manage FME Server and to troubleshoot any problems that might arise.

Prerequisites

This course assumes experience with FME Server. It is recommended to have completed the FME Server Authoring Course, and at least the Getting Started with FME Desktop and FME Server tutorials. Completion of the FME Desktop Basic and Advanced courses are suggested, though not necessary, to gain an understanding of the basic concepts and practices for FME. Knowledge of enterprise-level security and IT configurations is helpful, but not required.

Course Structure

The full course is made up of seven sections. These sections are:

- Planning an FME Server Installation
- FME Server Connectivity
- FME Server Security
- Scalability and Performance
- FME Server Customization
- Migration and Upgrades
- Troubleshooting

The instructor may choose to cover as many of these sections as they feel are required, or possible in the time permitted. They may also cover the course content in a different order and will skip or add new content to better customize the course to your needs.

Therefore the length and content of the course may vary, particularly when delivered online.

About the Manual

The FME Server administration training manual not only forms the basis for FME Server training – in-person or online – but is also useful reference material for future work you may undertake with FME.

This training material is designed specifically for use with FME2018.0. You may not have some of the functionality described if you use an older version of FME.

Note to Instructors

This training manual requires additional configuration and support of third-party software to complete the exercises as they are written.

- **Configuring for Active Directory:** To complete this exercise, training machines need to belong to a domain network and have access to Active Directory listings. Domain credentials must be provided in order to access Active Directory listings.
- **Changing the FME Server Database Provider:** To complete this exercise, training machines must have access to an external database where the schema for the FME Server database can be configured. A local installation of PostgreSQL is
used in this exercise, but Oracle or Microsoft SQL Server are alternatives. The database does not have to be local.

- **Web Connections:** To complete this exercise, a third-party web service with OAuth 2.0 authentication must be used. Dropbox is used in this exercise, but there are many other options. The web service does not need to have transformers available in FME – it is possible to leverage the HTTPCaller transformer for this purpose. It will be necessary to share Client Id and Client Secret credentials in order to configure the web service in FME Server.

- **Dashboards:** An exercise in this training manual configures FME Server with a self-signed certificate (Exercise: Configuring for HTTPS). The existing JobHistoryStatisticsGathering.fmw workspace is NOT compatible and must be replaced with JobHistoryStatisticsGathering_SelfSignedCert.fmw – this file is available in FMEData2018/Workspaces/ServerAdmin – either a new Schedule can be created or an update to the original.
Course Resources

A number of sample datasets and workspaces will be used in this course.

On Your Training Computer

The following applications may already be installed, licensed, and located on your training computer (real or virtual):

- Java Virtual Machine
- Apache Tomcat
- FME Desktop Version 2018.0
- FME Server Version 2018.0
- Microsoft SQL Server 2016

The data used in this training course is based on open data from the City of Vancouver, Canada.

Most exercises ask you to assume the role of a city planner at the fictional city of Interopolis and to solve a particular problem using this data.

Whether it’s a local computer or a virtual computer hosted in the cloud, you’ll find resources for the examples and exercises in the manual at the following locations:

<table>
<thead>
<tr>
<th>Location</th>
<th>Resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\FMEData2018\Data</td>
<td>Datasets used by the City of Interopolis</td>
</tr>
<tr>
<td>C:\FMEData2018\Resources</td>
<td>Other resources used in the training</td>
</tr>
<tr>
<td>C:\FMEData2018\Workspaces</td>
<td>Workspaces used in the student exercises</td>
</tr>
<tr>
<td>C:\FMEData2018\Output</td>
<td>The location in which to write exercise output</td>
</tr>
<tr>
<td>&lt; documents&gt;\FME\Workspaces</td>
<td>The default location to save FME workspaces</td>
</tr>
</tbody>
</table>

You should also find a digital copy of this manual.

Please alert your instructor if any item is missing from your setup.

You can find the latest version of FME Desktop and FME Server for Windows, Mac, and Linux - together with the latest Beta versions - on the Safe Software web site.

Course Etiquette

For online courses, please consider other students and test your virtual machine connection before the course starts. The instructor cannot help debug connection problems during the course!

For live courses, please respect other students’ needs by keeping noise to a minimum when using a mobile phone or checking e-mail.
Planning an FME Server Installation
Requirements for FME Workspaces

FME Workbench is used to author and publish the workspaces used by FME Server. FME Workbench is not part of FME Server; rather, it is a part of the FME Desktop product. If you do not have access to FME Desktop, you cannot publish workspaces to FME Server, although you can still perform and test the installation.

The version of FME Workbench you use must match the FME Server version you have installed. This requirement ensures that the workspaces you author will run when published to FME Server. If you are running a FME Server version that is older that your FME Workbench, it is possible to upgrade the FME Engines used by FME Server to match your FME Workbench version. These are best kept to minor release updates and changes.

For example, FME Desktop 2018.0.2 can publish to FME Server 2018.0.1 with 2018.0.2 Engines. However, FME Desktop 2018.1 should NOT publish to FME Server with 2018.0 Engines. Also FME Server 2018.0 does NOT support FME Server 2018.1 Engines.

Web Connections allow FME Workbench to connect to FME Server using the HTTP protocol. It requires the Web Application Server port to be open to communication between the FME Workbench computer and the FME Server computer.

Grid Shift Files

If you plan to run workspaces that transform data between datums NAD27 and NAD83, you can use datum shift files for Canada or the US. For information on Grid Shift Files, see the Datums in FME Documentation.

Third-Party Applications for FME

FME Workspaces often rely on third party applications for connecting to data sources. Examples include an Oracle client for connecting to an Oracle database and an Esri ArcGIS installation for connecting to Esri Geodatabases. For more information about Esri ArcGIS and FME Server, see the Using FME Server with Esri ArcGIS Software article. If you must install a third-party application for your FME Desktop installation, repeat that installation on your FME Server computer.

FME Workspaces using Python

FME Workspaces that contain python that use python interpreters different then the default interpreter that comes with FME must consider the FME Server environment and ensure the python interpreters are available and configured properly.

If any python modules have been added to the FME Desktop environment, the same modules will need to be uploaded to FME Server. Review the Uploading the Python Module section located on this page Using Python with FME Server in the FME Server Administrator's Guide for more information.

Web Service Connections

FME Workspaces can contain Web Services Connections. When you first publish a workspace to FME Server the web connection will be authenticated and ready to be used. Test the connection on FME Server to be certain it is working. Also, be aware of the any workspaces that contain web connections as they may require re-authorization/authentication from time to time.

After a workspace has been published, and the web connection uploaded as part of the workspace publish, the web connection will appear in the Web UI under Connections>Web Connections. The credentials for any web connection can be updated (authorized) through the Web UI on the Web Connections page.

Database Connections
FME Workspaces often connect to databases and if they are using a database connections in the workspace, FME Workbench will publish these to FME Server. If the database connection is using database credentials, then there should be no other action. However, if your database connection is using Windows Authentication, then it is key to understand that the user the workspaces will run under will be based on the FME Server Engine service Log on As settings in the Windows Services Dialog.

**TIP**

*When publishing workspaces with database connections to external systems ensure the FME Server system has network connectivity to the database and if 3rd party libraries (ex. Oracle Client) are required that they have been properly configured*

---

**Mr. Fibble says...**

*To perform data transformations, FME Server runs workspaces that have been created with FME Workbench. When a workspace runs on FME Server, it is referred to as a job.*
Installation Types

There are three options when you install FME Server: Express, Custom/Distributed, or Engine.

Express Installation

The Express option allows you to package all the components, or layers, of the FME Server architecture into a single machine. It is the quickest and easiest of the installation options since all components are provided for you, and you only need to provide a single server to host the installation.

Use the Express installation for any of these scenarios:

- You want to get started quickly with a single installation of FME Server.
- You are not planning to implement a distributed or fault tolerance environment with this system.

Custom/Distributed Installation

Another option is to install with the Distributed option. With Distributed, you can physically distribute the components into 3-tier or 2-tier configurations:
A Distributed Installation allows you to distribute the FME Server Application (FME Server Core, FME Server Queue, FME Server Web Application), FME Engines, FME Server Database, and the FME Server System Share across physically separate servers. As of FME Server 2018.0, it is recommended that the FME Server Web Application be installed on the same system with the FME Server Core.

If you choose this distributed architecture, you must provide and manage:

- A server for the FME Server Application (and optionally for any of the distributed FME Engines).
- You can run the FME Server Web Services on your own servlet (Apache Tomcat and Oracle WebLogic are supported), or use the Apache Tomcat servlet provided with the FME Server installation.
- A database server to host the FME Server Database (Oracle, PostgreSQL, and SQL Server are supported).
- A remote file system to host shared resources. This must be accessible using the UNC protocol and not through mapped network drives.

The diagram above shows the two distributed installations types recommended by Safe Software.

**Benefits of a distributed architecture**

Implementing a multi-tier architecture is good if you want to keep components separate so that each can be managed by the appropriate expert team. You also have finer control over applying security updates to the FME Server, FME Server Engines and Database servers when you supply your own. If you use the default FME provided components, you do not have the same amount of control for security.

**Implementing a multi-tier architecture**

You can implement a multi-tier architecture by:

1. Choosing one of the Distributed Installation options at installation time, or
2. After an Express installation by performing the following reconfiguration:
   - Changing the database provider for the FME Server Database.

**Engine Installation**

The third option for FME Server installation is an Engine-only installation. The Engine installation allows you to build onto a current FME Server distributed installation by adding FME Engines on a separate machine for fault tolerance and/or high capacity. By installing additional FME Engines on a separate computer from the FME Server Core, you can add processing...
NOTE 2018.0 to 2018.1 Engine Only Updates are not supported

**Silent Installation**

There is also the option for Silent installation with either Linux or Windows.

When performing a Silent installation, you can override any default installation properties to customize the installation. Installation properties can be set in advance or run with their default values.

The following command is an example of a silent installation on Windows for an install with all default options (similar to an Express install), with logging enabled:

```
msiexec /i fme-server-2018.0.1.1-b18312-win-x64.msi /qb /norestart /l*v
installFMEServerLog.txt
```
FME Server Architecture

FME Server has a number of components, some of which are considered part of the FME Server Core and others that are considered Clients of FME Server.

FME Server Clients include:

- Web Services (for example, the Job Submitter Service).
- Web Clients of FME Server such as the web interface.
- Non-Web Clients of FME Server, which include the FME Server Console, FME Workbench, and any custom application that uses the FME Server REST API.

Components that are part of the FME Server Core include:

- Process Monitor
- Repository Management
- FME Server Database
- FME Server Queue
- File System
- FME Engines
- Scheduling Manager
- Relay Manager
- FME Server Notifications
- WebSocket Server

FME Server Architecture Diagram
FME Server Components

The main components of FME Server to be aware of are:

- **FME Engines**: To carry out data transformation processing
- **Server Core**: To handle scheduling and notifications, and manage load balancing
- **Job Queue**: To queue jobs
- **System Database**: To store metadata related to workspaces, jobs, and configuration settings for FME Server operation
- **File System**: To store workspaces files, log files, and data shares
- **Web Services**: To handle networking capabilities

**FME Engines**

FME Engines process job requests by running FME Workspaces. This is the same core engine, carrying out the same processing, that is used by FME Desktop. An FME Server installation can possess multiple engines.

Each FME Engine processes a single request at a time, and FME Server processing can be scaled by adding FME Engines to the same computer or to separate computers within a distributed FME Server environment.

**Server Core**

The FME Server Core manages and distributes job requests (queueing, request routing, scheduling), the repository contents (workspaces, custom formats, custom transformers, data), and notification requests.

The FME Server Core contains a Software Load Balancer (SLB) that distributes jobs to available FME Engines.
FME Server Queue

New to 2018 - The Queue is a new component that is to provide a queuing mechanism for job submissions. When combined with a second FME Server system it will become fault tolerant automatically. The FME Server Queue is installed on the same system as the FME Server Core.

FME Server Database

The System Database is a critical component to the FME Server Core and must exist in order for the Core to properly function. The System Database is not required to be installed on the same system as the Core and can be hosted on an enterprise database on the local network. FME Server can be reconfigured after installation to use another database system if the default System Database was originally configured.

File System

The File System is where FME Server stores published workspaces and is the home of the Resources folders. Log files for the FME Server are stored here along with the completed job logs. This is often referred to as the FME Server System Share.

Web Services

Much of the FME Server networking capabilities are handled using what we call "Services." Services are software whose interface provides communication between server and clients.

FME Server has a number of services:

- Data Download
- Data Upload
- Data Streaming
- Job Submitter
- KML Network Link
- Token Security
- REST
- Notification

Some services (for example, Data Download) are “transformation” services that carry out data transformation, whereas others (for example, Token) are non-transforming “utility” services.

The Web Application Server is required in order to run the FME Server web interface, FME Server Web Services, and any other web clients. The web interface is included with FME Server and can be run in a browser. The Web Application Server is installed on the same system as the FME Server Core. Custom web clients can be developed on top of the FME Server REST API.
Planning for Fault Tolerance

Fault tolerance, or “high availability,” is critical to any successful business operation. To ensure that requests are processed in the event of failure, FME Server supports configuring fault tolerance throughout the multiple levels of an integrated system. FME Server provides fault tolerance in the following ways:

1. **Recovery**: Restarting components and jobs when crashes occur. FME Server provides component and job recovery automatically - no additional planning is needed.
2. **Redundancy**: Ensuring there is no single point of failure. New in FME Server 2018 is the fault tolerance architecture. When two FME Server systems are configured together, fault tolerance is achieved automatically.

A Note on Fault Tolerance in FME Server 2018.0

FME Server 2018 has a new and improved fault tolerance capability. In 2018.0 this feature is a tech preview. It is recommended that if you require fault tolerance capability that you work with FME Server 2018.1. We no longer support Active-Passive FME Server environments. The new fault tolerance requires a minimum of two systems with FME Server installed on both systems, using the same FME Server System Share and same FME Server System Database. There is also a requirement for a load balancer that monitors the health and distributes web requests between all of the FME Server environments. All the FME Server hosts are active at the same time.

**Recovery**

**Component Recovery**

FME Server comes out-of-the-box with component recovery. This means that, even on a single system, FME Server monitors and restarts components that fail, including the FME Engines and the FME Server Core. This is achieved through the FME Server Process Monitor. The ability for FME Server to monitor its own components ensures reliable uptime and dependability.

**Job Recovery**

FME Server also includes the ability to restart a job when a crash occurs. As a result, jobs that experience temporary issues, such as a network hiccup, are re-submitted and run again.
After FME Server submits a translation request to an FME Engine, it monitors the connection to that engine until a response is returned.

FME Server can resubmit a failed job if:

- The connection to the engine is lost.
- The engine crashes.

FME Server continues to re-submit a translation up to a specified number of attempts. To prevent FME Server from indefinitely retrying a job that fails, the default setting is to resubmit a failed job up to three times. This setting is configurable and can be turned off entirely.

**Ms Analyst says...**

**WARNING!** A failed translation request may cause an FME Engine to shut down improperly. If no maximum limit is imposed, the translation is resent indefinitely, which may cause repeated FME Engine failures. Re-submitted transactions may also cause data duplication, such as when writing to database formats or when writing mid-translation with the FeatureWriter. If a job is resubmitted because of a failure, and then succeeds the first job log file is overwritten and this will permanently delete why the job failed the first run. This is vary rare but in these cases you may need to set job resubmission to zero.

**Redundancy**
The goal of a fault tolerance environment is to remove single points of failure so that a component can fail, but not take the entire system offline. This is achieved by having multiple systems with FME Server installed on each and pointing to the same FME Server System Database and FME Server System Share.

The new fault tolerant architecture, at the simplest implementation duplicates most of the FME Server components on separate servers. Additional systems are configured similarly and provide the same functionality. A third-party load balancer directs incoming traffic to either of the available systems. There is no stickiness required for the client sessions. Requests are directed to any of the systems.

The following image shows 2 deployment examples. The recommended approach and a fully distributed deployment. By following the recommended approach you will gain the benefits of fault tolerance with the minimum number of systems.

---

**Basic Architecture Requirements**

- Load Balancer System
- FME Server Components (minimum two systems)
- Fault Tolerant Database
- Fault Tolerant File System

**Benefits**

- Simple to manage
- Fewer systems required.
- Can increase number of engines available on each system
- Easy to add additional systems to increase capacity

**Distributed Architecture Requirements**

- Load Balancer System
- FME Server Web (minimum two systems)
- FME Server Core (minimum two systems)
- FME Server Engine (minimum two systems)
- Fault Tolerant Database
- Fault Tolerant File System

**Benefits**

- Allows for use of own Web Servlet and thus security updates without disrupting other systems
- Allows engines to be deployed easily with 3rd party Software
- Finer control for scaling each system's capabilities (memory, CPU, disk space)

**TIP**

*In a fault tolerant installation of FME Server, the Notification Service UDP Publisher and SMTP Publisher are not supported. To receive e-mail notifications, consider the IMAP Publisher instead.*

---

**Load Balancer System**

The customer must provide their own load balancer (LB) and this can be configured to point to FME Server and perform regular health checks (if supported). The LB can also use timeouts on requests to redirect the requests to another FME Server system.

**FME Server Components**

It is recommended to install the FME Server Web Application, FME Server Core, & FME Server Engines (optional) on a single system and repeat this for a second system (see image above 'RECOMMENDED'). This provides you with the basic fault tolerant environment. The LB would then be directed to point to these 2 systems.

Further, similar additional systems can be added to the environment to expand the high availability. Systems with only the FME Server Engines can also be registered with the FME Server Cores to increase the engines available and distribute the processing across more systems.

The FME Server cores become aware of each other and will handle requests. There will be one Job Manager, and if this fails, the other Job Manager on the other system will take over and handle job requests. There should be minimal downtime when a core goes done. Allow a few moments (1-2 minutes) depending on the LB configuration.

Schedules will continue to operate normally.

**Fault Tolerant Database**

The customer is in charge of making the Database fault tolerant.

**Fault Tolerant File System**

The customer is in charge of making the File System fault tolerant.

**Tracking Core Failures**

The failed system can then be investigated while the second active system provides continued operation of FME Server. Once the new failed system is recovered and started, it will join the environment seamlessly.

The types of failures that typically cause faults are hardware and operating system crashes, in which the system goes down completely.
Log files must be reviewed on the affected system to understand why the FME Server core failed. When the core's availability is affected, the outcome is usually an unusable system.

Sister Intuitive says...

In the past, clients of Notification Service publishers did not failover but in 2018.1 this will also occur.
Planning for Disaster Recovery

Disaster recovery is primarily concerned with recovering FME Server operations and data in the event of a major failure of a data center. The time frame for disaster recovery is typically longer than fault-tolerant recovery. Disaster recovery may range from minutes, hours, or even days, while fault-tolerant recovery is typically expected in seconds or minutes, and seamless to the end user.

Disaster recovery can be incorporated into any of the fault-tolerant architectures. Alternatively, if you are primarily concerned with disaster recovery, and less concerned - or even not at all concerned - about the fast recovery provided by fault tolerance environment, you may want to implement a different architecture.

The general concept of disaster recovery is that if one data center fails, the second data center takes over, and the FME Server Core located there becomes the active core. Often this type of recovery involves redirecting network traffic to a new server address in the new data center.

This example of disaster recovery is an adaptation of an Active-Active architecture, but without the third-party load balancer between systems. Instead, FME Server clients must be manually redirected to the Core host server of the second data center in the event of a disaster. Each data center houses full (“Express”) installations of FME Server, essentially configured to provide similar functionality. To ensure synchronicity of the FME Server system data between data centers, Backup & Restore operations are performed regularly. (Otherwise, workspaces must be published twice - to the FME Server Core hosts on each data center).

Keep in mind that when planning for disaster recovery, all clients of FME Server, including web browsers, the FME Server Console, and the FME Server REST API, must connect to the active FME Server Core host.
Security Updates

All installations of FME Server, regardless of type, include the FME Server Core and FME Engines. These components are always provided directly from the FME Server installation package. Two additional components - a Web Application Server and a server for the FME Server Database - must also be installed.

When you choose to install a full, stand-alone version of FME Server, the install package provides its own versions of these components, including an Apache Tomcat web application servlet, and a PostgreSQL database server. This is the “Express” installation option. If you choose a Distributed installation of FME Server, you can either provide your own web application server or choose the default FME Server Web Application Server, and depending on the scenario (2-tier or 3-tier architecture), you may also provide a database server.

One factor in deciding between a stand-alone or distributed installation of FME Server is the degree of control you want in applying security updates to the web application and database servers. If you install a full, stand-alone FME Server, keep in mind that any security updates to these components are dependent on updates to FME Server releases in general. Each time FME Server releases an update to its software (including both major and minor releases), any security updates for these components are included in that release. You will need to re-install to receive these updates.

If you do not want to rely on updates to the FME Server software in general for security updates to the Web Application and Database servers, then we recommend a Distributed installation. In this case, you provide these additional server components on your own and maintain security updates for them separately.

- In the case of the Web Application server, both Apache Tomcat and Oracle WebLogic are supported.
- The FME Server Database supports PostgreSQL, Oracle, and SQL Server.

Alternatively, if your FME Server is entirely internal to your organization, and behind a firewall, then you may be more comfortable with the security updates provided with a full installation.

---

Mr. E. Dict, (Attorney of FME Law) says...

*Please be sure to review the FME Server Technical Specifications for the most up-to-date information on Web Application and Database Server compatibility.*
Licensing

FME Server 2016 introduced a new licensing mechanism that no longer relies on third-party software to serve licenses. FME Engine licenses are now served directly through the FME Server Core and are saved to the FME Server System Share - this introduces a variety of benefits including easier management, especially when configuring a Distributed/Custom FME Server installation.

Request and Install a License

Online Mode

The fastest and easiest method for licensing FME Server is to use the online method. FME Server requires only basic user information (Name, Email) and the serial number provided by your account manager. Requests are sent over HTTP/HTTPS to retrieve the license file from our backend databases.

You can request a license in the Engines & Licensing tab of FME Server:

By default, license files are installed to C:\ProgramData\Safe Software\FME Server\licenses.

Chef Bimm says...

If you change the machine FME Server is installed to, you do not have to contact Safe Support or your account manager - simply request the license again after FME Server is installed to the new machine!

Offline Mode

If the machine hosting the installation of FME Server is disconnected from the internet, or if firewall rules prevent your machine from communicating with our backend database, then there is an offline method for retrieving your FME Server license file.
In the Request License form, select 'No' to instead have a JSON file downloaded to your local file system. This file can then be forwarded to codes@safe.com where an automatic process (running FME in the background) will return a valid license file to you after a few minutes.

**Request A New License**

**Use automatic licensing via the internet?**

- [ ] Yes
- [ ] No (manual - no Internet or other connection issues)

After clicking "ok" below, a license request file will be downloaded. Please email this to codes@safe.com.

This file can then be drag-and-dropped onto the Engines & Licensing page to license FME Server.

**Extending an Evaluation License**

If you requested a 7-day evaluation license, you should have an e-mail from codes@safe.com that allows you to extend the license to 60 days. Any user can request an FME Server evaluation license without the need for a serial number.
Chapter Review

This chapter introduced you to how to plan for and license an FME Server installation.

What You Should Have Learned from this Module

Theory

- FME Workspaces are a part of FME Desktop, not FME Server. If you do not have access to FME Desktop, you cannot publish workspaces to FME Server, although you can still perform and test the installation.
- FME Server has multiple installation types: Express, Distributed/Custom, Engine, and Silent.
- A Distributed installation can be 2- or 3-tiered depending on how you want to distribute the FME Server Web Services, FME Server Application, and the FME Server Database.
- FME Server provides fault tolerance through Recovery and Redundancy.
- Component Recovery is achieved through the FME Server Process Monitor - automatically restarting components that fail.
- Job Recovery is the ability to restart a job when a crash occurs.
- Configuring for multiple cores and removing single points of failure so that a component can fail, but not take the entire system offline.
- There is a new fault-tolerant architecture coming in 2018.1 that replaces the previous modes of failover (Active-Passive and Active-Active).
- Disaster Recovery can be incorporated into any of the architectures.
- Security updates for a stand-alone FME Server are dependent on updates to FME Server releases.
- Distributed FME Server installations allow you to maintain the security updates for your provided server components (the Web Application Server and/or the Database Server).
- FME Server can be licensed using either Online or Offline methods.
FME Server Connectivity
Firewalls and Ports

Configuring Firewall Settings

The FME Server Web Services and other clients use the FME Server API to communicate with the FME Server Core over TCP/IP. Requests are sent to the FME Server Core over port 7071. Result messages are returned to clients over a randomly assigned port created by the FME Server Core. Therefore, be sure to configure any firewall settings on the <webHost> with a rule to allow for full communication with the <coreHost> machines.

When specific ports are desired, the port should be made available and not blocked by firewall settings.

**Note:** An Inbound Rule "Allow All" is created for <FMEServerInstall>Server\fme\fme.exe on Windows machines when FME Server is installed.

Ports

- Ports 25, 7125, and 465 are for the SMTP Publisher.
- Ports 6379 are for FME Server Queue.
- Ports 7069 and 7082 are for FME Server Database communications.
- Ports 7070 and 7501 manage FME Engine processes.
- Port 7071 uses the REST API to send requests to the FME Server Core.
Ports 7072 - 7076 manage Notification Services.

Ports 7077 and 7081 are for Configuration, Backup & Restore requests and System Cleanup tasks.

Port 7078 handles WebSocket Server requests.

Port 7079 handles FME Server Resource requests.

Port 7500 manages FME Server Core processes.

**Miss Vector says...**

For an up-to-date and complete list of ports used by FME Server, and more detailed descriptions, please see FME Server Ports documentation.
DNS

It is important to know the name of your host computer when using FME Server. During the installation, you will be prompted:

- To enter the hostname for connecting to FME Server (with Express installation),
- To specify the host that will run the FME Server Core (with Distributed installation), or
- To specify the name of the machine hosting the primary FME Server Core (with Engine installation).

It is essential to make sure you have the correct hostname for proper control and management of FME Engines as changing this post-installation is currently only supported for the FME Server Web Services.

Police Chief Webb-Mapp says...

Do not use “localhost” as your hostname if you want FME Server to be accessed remotely!
CORS

Cross-Origin Resource Sharing (CORS) allows you to specify websites hosted on other domains that can access resources from the FME Server through Ajax requests.

CORS is enabled by default to allow any host to access FME Server resources.

To disable CORS:

1. Click Load Template, and select Disable CORS.
2. Click Save Changes. (Or to cancel, click Revert Changes).

To re-enable CORS:

1. Click Load Template, and select Allow All Hosts or Allow Specific Hosts.
2. Configure the remaining settings as desired.
3. Click Save Changes. (Or to cancel, click Revert Changes).

CORS Settings Options

*Allowed Origins*: A comma-separated list of hosts that are allowed access to the FME Server. An asterisk (*) allows access from any host. An asterisk cannot be specified if any origins are passing credentials. For an example of how to specify the list of hosts, click Load Template and select Allow Specific Hosts.

*Allowed Methods*: A comma-separated list of HTTP methods that may be used in requests from the allowed origins.

*Allow Headers*: A comma-separated list of permitted request headers from the allowed origins. A request header is any custom header set by the browser JavaScript application through method XMLHttpRequest.setRequestHeader().
Exposed Headers: A comma-separated list of non-standard response headers that are safe to expose to the requestor (initiated through the XMLHttpRequest.getResponseHeader() method). This information is returned in the Access-Control-Expose-Headers response header.

Pre-flight Max Age: Specifies how long, in seconds, the results of a pre-flight request can be cached by the requestor. This information is returned in the Access-Control-Max-Age response header.

Support Credentials: If TRUE, allows the requestor to include credentials to authorize with the FME Server, including cookies, HTTP authentication (tokens), or client-side certificates. This value is returned in the Access-Control-Allow-Credentials response header.

---

Miss Analyst says...

FME Server 2017+ now has "Allow All Hosts" as the default setting for CORS.
Chapter Review

This chapter introduced you to FME Server connectivity - Ports, DNS, and CORS settings.

What You Should Have Learned from this Module

Theory

- Firewall settings should not block FME Server ports.
- FME Server utilizes multiple ports with their own important functions.
- The correct hostname allows for proper control and management of FME Engines.
- Cross-Origin Resource Sharing (CORS) allows you to specify websites hosted on other domains that can access resources from FME Server.
- CORS is enabled to Allow All Hosts by default.

FME Skills

- How to disable and re-enable CORS.
FME Server Security
Role- and User-Based Access

FME Server security is based on two primary concepts:

- **Users**: Users are the individual accounts that access FME Server. When FME Server is installed for the first time, default user accounts are created.
- **Roles**: Roles are comprised of one or more users.

FME Server security controls access to resources either through role-based or user-based access.

Role-Based Access

Roles make it easy to assign the same set of permissions to multiple users based on job function. Permissions to perform certain operations are assigned to specific roles. In turn, these permissions apply to the users who belong to that role.

For example, a request by user *user1* could be to run a workspace in the Samples repository for the Data Download Service. FME Server security grants access if any of the roles to which *user1* is assigned has permission to run workspaces in the Samples repository, and also has access to the Data Download Service.

A default set of roles is defined when FME Server is installed. These are:

- **fmesuperuser**: For users with unlimited access to the system, including Backup & Restore tasks.
- **fmeadmin**: For users who need to carry out specific administration tasks.
- **fmeauthor**: For users who are authoring workspaces to run on FME Server.
- **fmeuser**: For users who need to run (but not author) workspaces.
- **fmeguest**: For temporary users who need a minimal set of permissions.

A number of default accounts are created too. These are:

- **admin**: Assigned to the fmesuperuser and fmeadmin roles.
- **author**: Assigned to the fmeauthor role.
- **user**: Assigned to the fmeuser role.
- **guest**: Assigned to the fmeguest role.

Chef Bimm says...

*Don’t forget, these are just default accounts that FME creates. You can create any role necessary for your system, assign any specific security settings to it, and create any number of users assigned to that role.*
On the Roles page of the Web User Interface, an administrator can:

- Create and remove roles.
- Configure users in roles.
- Configure permissions of roles.

**User-Based Access**

Another way for FME Server to determine if a user can access a resource is whether the user owns it, or has been given permissions on it.

**User Ownership**

Anything a user creates in FME Server, such as a repository, is owned by that user. When you own something, you have full permissions on it. This permission supersedes the permissions you have on other items in FME Server based on the role to which you belong.

Additionally, as an owner, you can:

- Share permissions on the items you own with other users or roles.
- Assign ownership of something to another user.

**User Permission**

Users can be granted permissions on resources, and these permissions may supersede the permissions available to them through their role. (In fact, it is not even necessary for a user to belong to a role.)

On the Users page of the Web User Interface, an administrator can:

- Create and remove users.
- Configure users into roles.
- Configure permissions of users.

---

**Sister Intuitive says...**

On the Active Directory page of the Web User Interface, an administrator can integrate the organization’s Active Directory users and groups into its FME Server security configuration.

---

**Miss Vector says...**

If I want one user to have a higher level of access to other users in the same role (say I wish to let an FME author be able to manage engines) what must I do?

1. Simply select that user from the user list and enable the manage Engines & Licensing policy
2. Promote that role to superuser status so that the user has a higher level of security
3. Create a new role with the manage Engines & Licensing policy enabled and move that user to it
4. Create a new role with the manage Engines & Licensing policy enabled and add that user to it as well as the original role
Security Policies

There are various sections of security policies that can be set for each role or user. While there are permissions that can be set on each individual item in FME Server, the visibility of pages in the FME Server web interface are managed by the Access or Manage privileges.

Run Workspace

Run Workspace controls the ability to - you guessed it - run a workspace on FME Server. The Advanced option allows a user to access advanced job directives and the direct URL to run a workspace.

Jobs

Access to the Jobs page allows the user to view the jobs they have run, or cancel any of their jobs that are currently running or in the queue. The Manage option allows them to view the full job history on FME Server and the ability to cancel any job, or remove any job from the history.

Schedules

Schedules are one of the core functions FME Server provides. Access to each schedule can be controlled.

Repositories

Repositories are a place to store and categorize workspaces. Each role or user can be given different permissions for every repository - download, read, publish, run, remove. Access permission is not necessary to run a workspace - only run permission on the applicable repository is required.

Version Control

Version Control is a new feature in FME Server 2018.0, and it allows users to version the workspaces they have published to FME Server, either during publishing or by using the FME Server web interface. A user can be granted Access or Manage permissions.

Workspace Viewer

Workspace Viewer is a new feature in FME Server 2018.0, and it allows a user to view published workspaces through the FME Server web interface. A user can be granted access to use this feature.

Publications

Publications are a subset of the FME Server Notification System. Different permissions - read, write (edit), remove - can be assigned for each publication.

Subscriptions

Subscriptions are a subset of the FME Server Notification System. Different permissions - read, write (edit), remove - can be assigned for each subscription.

Topics
Topics are related to Notifications (Publications and Subscription). Different capabilities – read, write, publish, remove – can be assigned for each topic created.

**Resources**

Resources are files and datasets stored on FME Server. Different permissions – access, list, write, upload, remove – can be assigned to each resource.

**Connections**

Connections are predefined logins to either web services or databases. Each role or user can be given permission to manage the different connections stored in FME Server. With this permission, the user can create new definitions, and manage connections in the FME Server web interface.

**Projects**

Projects are created to manage a subset of FME Server resources, for example - workspace, schedules, publications, and subscriptions. Different permissions - Can View, Can Edit, Full Access - can be assigned to a user or role for each project.

**Dashboards**

Dashboards offer an easy way to view FME Server health using the FME Server web interface.

**Engines & Licensing**

The Engines & Licensing page concerns licensing FME Server, managing the number of active FME Server Engines, and the number of connected hosts.

**Security**

Granting the permission for Security allows the role or user to create, enable, disable, and remove users and roles, as well as configuring for Active Directory.

**System Cleanup**

System Cleanup tasks are defined in the FME Server web interface and help to manage temporary files and log files among other items.

**Services**

Services are key items of functionality on FME Server. They are the different methods by which a workspace can be run and output data delivered. Each role or user can be allowed – or not – to use a particular service.
Your company has recently hired a new analyst who will be accessing FME Server. The new employee doesn't quite fit into the current FME Server Roles you have in place so you need to create a new role for them.

1) Connect to FME Server
Open the FME Server web interface, either through the Web Interface option on the Windows Start Menu or directly in your web browser ([http://localhost/fmeserver](http://localhost/fmeserver)), and log in using the username and password **admin**.

Click Security, under the Admin heading on the left sidebar to expand the menu, and then select Users to see a list of your current users.

2) Create a New User
Let's create a new FME Server user account for the new analyst. Click New to add a new user:

When prompted, create a new user with the following parameters:

- **User Name:** NewUser
- **Full Name:** NewUser
- **Password:** NewUser1

3) Configure Permissions
Now that we have the credentials for our new user account specified, let's set the permissions for what features and items in FME Server they have access to.

By selecting the **Load Template** button, you have the option to copy permissions from an existing role.

Select **fmeguest** from the Load Template options. This can help speed up security configurations. Notice that Run Workspace and Jobs have been now granted Access.

Configure permissions to match the following:

- **Run Workspace:** Access
- **Jobs:** Access
- **Schedules:** Create
- **Repositories**: Create
- **Projects**: Create

Notice how by selecting Create, the Access privilege is automatically granted.

Select **OK** at the bottom to create the user.

4) **Test the New User Account**

It's important to verify the options we set have been honored.

Either logout of the admin account or open a new private browsing window, and login using the credentials for the new user account we just created.

Notice how this user only sees a limited set of menu options: Run Workspace, Jobs, Schedules, Repositories, and Projects.

---

**CONGRATULATIONS**

By completing this exercise you have learned how to:
- Create a new user on an FME Server installation
- Set permissions from an existing FME Server role
- Test a newly created account to ensure it works correctly
Running Services without Authentication

A special account referred to as the **trusted user account**, can be used to provide unauthenticated access to any component of FME Server. By default, this trusted account is named **guest** and is assigned to the **fmeguest** role. By default, the **fmeguest** role is configured to allow unauthenticated access to the FME Server Web Services. This means it is possible to invoke a service URL without providing any credentials.

**Police Chief Webb-Mapp says...**

>If you want all of the FME Server Web Services to prompt for authentication, remove the guest account after you configure your own set of users and access control for your server.

The trusted user account is configured in the `propertiesFile.properties` file for each web service. If your FME Server installation uses the built-in Apache Tomcat servlet, these files are located under:

```
C:\Program Files\FMEServer\Utilities\tomcat\webapps\<service&qt>\WEB-INF\conf\propertiesFile.properties
```

You will have to change `<service&qt>` to one of the various services located in the webapps folder

To change the username and password of the trusted account for a service, configure the **DEFAULT_USER_ID** and **DEFAULT_PASSWORD** parameters:

```
#DEFAULT_USER_ID - The default user ID to log in FME Server
DEFAULT_USER_ID=guest

#DEFAULT_PASSWORD - The default password for default user ID
DEFAULT_PASSWORD=guest
```
Running FME System Services under Different Accounts (Windows)

By default, the FME Server Core, FME Server Engines, and FME Server Application Server Windows Services run under the "Local System" account, which may not have network permissions. You may need to run these services under different accounts that can read and write data to the FME Server System Share, particularly in a distributed installation where these services are installed on separate machines.

For instructions on how to run the FME Server System Services under a different account, please see current 2018.0 documentation.

In an Express Installation or a Distributed Installation that is using the FME Server System Database, it is not necessary to change the "Log on as a Service" setting for this service.

Police Chief Webb-Mapp says...

When installing FME Server it is optional to provide a "domain service account" to configure the services to start with. On a new Windows OS system, ensure the domain service account has been added to the system as an Administrative user and ensure the Local Policy - "Log on as a Service" has been granted to this user before installing FME Server. This will ensure the services are installed correctly and can start up properly the first time.
Active Directory and FME Server

FME Server allows you to connect to an existing Active Directory / LDAP server and incorporate available users and groups into your FME Server security configuration.

Once a connection is created, you can specify which user(s) and role(s) will be imported into FME Server - noting that their passwords and membership will continue to be managed by the Active Directory server itself. The existing users and roles on FME Server can coexist with those imported. FME Server roles can contain both System (FME Server) and Active Directory users.

FME Server can manage any number of Active Directory connections - this means that you can connect to multiple domains.

Police Chief Webb-Mapp says...

One cautionary note when working with multiple domains is if a second domain contains a username that is the same as in the first domain (and has already been imported into FME Server), the second user will not be imported and an alternative name will be required an prompted for during the import of users.

First-Officer Transformer says...

While you can import Active Directory Roles, you cannot modify membership in FME Server.
FME Server only has read permission on any connected Active Directory listing.

Integrated Windows Authentication

With Integrated Windows Authentication, also known as "single sign-on," you can enable the users you import from your Active Directory connections to integrate their Windows login credentials with FME Server. When single sign-on is enabled:

- There is no need to log in to the FME Server web interface. Instead, select Use Windows Credentials on the Sign In page.
- Similarly, there is no need to log in to FME Server when using FME Workbench to publish a workspace. Instead, simply check Use Windows session credentials in the Publish to FME Server wizard.

Note: When publishing a workspace to a Notification Service, you must still provide your FME Server credentials in the HTTP Authentication fields of the Edit Service Properties dialog of the wizard.

To enable single sign-on

1. Update the Windows domain configuration to allow FME Server to authenticate using single sign-on.
2. Update the web browser configuration to use single sign-on.

Chef Bimm says...

Once Integrated Windows Authentication is configured, users will need to log into FME Server using the **Use Windows Credentials** button in the browser. At this time, a user cannot be automatically logged in when connecting to FME Server the first time. Once a user has been logged, however, and closes their browser, they may be automatically logged back in when returning to FME Server Web Interface.
subsequent visits or until the session expires.

TIP

Internet browser session: A web browser session to FME Server do not expire as long as the browser is active and the user remains logged in. Logging out of FME Server will end the web browser session.
Exercise 2: Configuring FME Server for Active Directory (LDAP)

<table>
<thead>
<tr>
<th>Data</th>
<th>N/A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Goal</td>
<td>Connect FME Server to an existing Active Directory service</td>
</tr>
<tr>
<td>Demonstrates</td>
<td>Configuring Active Directory in FME Server, Importing Users and Groups</td>
</tr>
</tbody>
</table>

This exercise is for demonstration purposes only

This lab requires a Windows domain controller to be present and available to connect to from the FME Server system. The training environment being used today does not have access to a domain controller. The following steps and video are presented as a guide for configuring the typical active directory to work with FME Server. It does not cover all possible configurations that may be required for your particular active directory.

Sister Intuitive says...

Due to security requirements and restrictions it is not possible to complete this exercise. Instead, please watch this video demonstrating the exercise.

1) Connect to FME Server
Open the FME Server web interface, either through the web interface option on the Windows Start Menu or directly in your web browser `http://**<your fmeserver host>**/fmeserver`, and log in with an admin account.

Click Security, under the Admin heading on the left sidebar, and then select Active Directory.

2) Create Connection to Active Directory
By creating a new connection, you can incorporate your organization’s Active Directory users and groups into your FME Server security configuration.

To get started, select New to open the Create New Server Connection page.

Enter the following information:

- **Name:** FME Active Directory
- **Host:** dc.fme.com
- **Port:** 389
- **Search Account Name:** DC\Administrator
- **Search Account Password:** dcAdmin2017

Click OK to save the new Active Directory connection. You will be returned to the Active Directory page. Wait for the Status to change from Yellow to Green, indicating that the connection is successful.

3) Import Users
Now that the connection is established, select the Import Users icon to add users from the Active Directory connection.

On the Browse Users page, type in mvector and press Enter. Select Miss Vector's user and click Import.
A notification will appear in the top right of the web browser window to indicate that the user was successfully imported.

Note: If Miss Vector belonged to any Active Directory groups, we could have instead imported that as an FME Server Role – and all users that are a member of would be imported automatically.

**TIP: Import Error**

*When importing users from Active Directory you may encounter this message. This is because a username of the same value already exists in the SYSTEM users.*

*It is recommended that you remove the SYSTEM user account, and reimport the Active Directory user. This error can also occur if you are importing users from a second domain that contains a same named user as the first domain. In this case it will be necessary to provide a different username on this dialog to represent the user from the second domain.***

NOTE: FME Server creates an alias for the imported usernames and this is linked to the user account in the Active Directory.

---

4) Configure User Permissions

After the Active Directory user is imported to FME Server, you must configure the permissions.

Select Security > **Users** under the Admin heading on the left sidebar of the FME Server web interface. Click on the **Miss Vector** user that was just created to open the Edit User page.

Click in the text box area for **Assigned Security Roles** and select **fmeauthor**. Notice all the inherited permissions from the fmeauthor Role that are now selected.

Select OK at the bottom to apply the changes.

5) Test the New User Account

Test that the import and assigning permissions was successful by logging into FME Server as Miss Vector.

Either logout of the admin account or open a new private browsing window, and login using the credentials below:

- **Username**: mvector
- **Password**: dcFME2017
CONGRATULATIONS!

By completing this exercise you have learned how to:

- Connect FME Server to an existing Active Directory configuration
- Import Users and Groups from Active Directory
HTTPS/SSL and FME Server

HTTPS ensures that communication between the client and server is encrypted, so that if it is intercepted, the third party cannot easily view or use the information. For FME Server, you can use HTTPS to ensure that sensitive login information is not exposed.

When configuring FME Server for HTTPS, both Certification Authority (CA)-issued and self-signed certificates are supported. Wildcard Certificates are also supported.
Your company is rapidly expanding and hiring many new employees. Now, instead of having everyone able to access to FME Server, you have set up logins so only trusted personnel have access. You also want to set up extra precautions to keep the transferred information secure.

HTTPS ensures that communication between the client and the server is encrypted, so that if it is intercepted, the third party cannot easily view or use the information. For FME Server, you can use HTTPS to ensure that sensitive login information is not exposed.

For any HTTPS (SSL) page, a certificate is required. For development and testing purposes, self-signed certificates are supported. For production use, we recommend that you use SSL certificates from a verified SSL certificate authority (CA).

1) Create a Keystore File
First, you must generate a keystore that contains a certificate chain using the Java Keytool from the Java Developer Kit (JDK).

Open a **Command Prompt** as an administrator.

Navigate to the FME Server Java bin directory:

```
cd C:\Program Files\FME\Server\Utilities\jre\bin\n```

Run the following command to create a new keystore file:

```
keytool -genkey -alias tomat -keyalg RSA -keystore tomat.keystore
```

Set the following values when prompted:
- **Keystore Password**: tomcat
- **First and Last Name**: localhost
- **<Remaining Parameters>**: <leave_blank>

Enter yes when prompted if the input is correct. When prompted for the key password for <tomcat>, press RETURN.
A new keystore is created in `C:\Program Files\FMEServer\Utilities\jre\bin\`

Copy the new keystore file to the tomcat directory in the FME Server installation:

```
copy tomcat.keystore "C:\Program Files\FMEServer\Utilities\tomcat\tomcat.keystore"
```

**TIP**

Ensure the keystore file is **COPIED** NOT moved.  
This is most important when working with a distributed FME Server Core and FME Server Web Application.

2) Working with the Certificate

The new keystore must be imported into the FME Server keystore for trusted certificates. In the command prompt, enter the following command:

```
keytool -importkeystore -srckeystore tomcat.keystore -destkeystore "C:\Program Files\FMEServer\Utilities\jre\lib\security\cacerts"
```

You will be prompted to enter two passwords. One for the destination keystore and one for the source keystore. The password for the destination keystore is **changeit**. The password for the source keystore is **tomcat**.
Configuring Tomcat

In the next steps, we need to modify three configuration files of Apache Tomcat. All three files are located in the FME Server installation directory: C:\Program Files\FMEServer\Utilities\tomcat\conf

It is a good idea to make copies of any files you will be changing and place them in a separate directory until you have verified that the edits are working successfully.

3) Configure server.xml

Open C:\Program Files\FMEServer\Utilities\tomcat\conf\server.xml file in a text editor in administrator mode.

Locate the SSLEngine setting in the <Listener> element, including className="org.apache.catalina.core.AprLifecycleListener" and change the “on” value to “off”.

```xml
<Listener className="org.apache.catalina.core.AprLifecycleListener" SSLEngine="off" />
```

Locate the <Connector> element that contains protocol="org.apache.coyote.http11.Http11NioProtocol" and replace it with the following:

```xml
<Connector protocol="org.apache.coyote.http11.Http11NioProtocol" port="8443" minSpareThreads="5" enableLookups="true" disableUploadTimeout="true" acceptCount="180" maxThreads="200" scheme="https" secure="true" SSLEnabled="true" keystoreFile="C:\Program Files\FMEServer\Utilities\tomcat\tomcat.keystore" keystorePass="tomcat" clientAuth="false" sslEnabledProtocols="TLSv1, TLSv1.1, TLSv1.2" sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation" ciphers="TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA256,TLS_RSA_WITH_AES_128_CBC_SHA256,TLS_RSA_WITH_AES_256_CBC_SHA256,TLS_RSA_WITH_AES_128_CBC_SHA,TLS_RSA_WITH_AES_256_CBC_SHA,SSL_RSA_WITH_3DES_EDE_CBC_SHA" URIEncoding="UTF8" />
```

Save and close the server.xml file.

Exercise 3: Configuring for HTTPS
4) Configure web.xml
Open the web.xml file in a text editor in administrator mode.

Add the following code block to the end of the file, just before the closing </web-app> element:

```
<security-constraint>
    <web-resource-collection>
        <web-resource-name>HTTPSOnly</web-resource-name>
        <url-pattern>/*</url-pattern>
    </web-resource-collection>
    <user-data-constraint>
        <transport-guarantee>CONFIDENTIAL</transport-guarantee>
    </user-data-constraint>
</security-constraint>
```

Save and close the web.xml file.

5) Configure context.xml
Open the context.xml file in a text editor in administrator mode.

Add the following to the end of the file, just before the closing </context> element:

```
<Valve className="org.apache.catalina.authenticator.SSLAuthenticator"
disableProxyCaching="false" />
```

Save and close the context.xml file.

6) Verify the Configuration
Now that we have made our changes, we want to verify that HTTPS was configured correctly for FME Server.

Restart the FME Server Application service from the Start menu > FME Server 2018.0 > Restart FME Server.

Open a browser and navigate to https://localhost:8443/fmeserver.

You should see the FME Server login page in a secured format.

![FME Server login page](https://example.com/fmeserver.png)

Note: If a self-signed certificate is used for testing, your browser may report the page as not secure:
For self-signed certificates, some browsers will allow you to add an exception for https://localhost:8443/.

7) Modify Service URLs to Use HTTPS
To enable SSL for FME Server Services, login to the FME Server web interface (username and password admin), and select Services on the left sidebar.
On the Services page, you can update specific services or all services at once. Let's update all services. Click **Change All Hosts**.

The *Change All Hosts* dialog opens. Make sure **Host** is set to `https://localhost:8443` and click **OK**.
Exercise 3: Configuring for HTTPS

TIP

If you are making use of WebSockets with FME Server please review the FME Server Admin Guide and the section titled Enable SSL on the WebSocket Server (Optional)

CONGRATULATIONS!

By completing this exercise you have learned how to:

- Create a self-signed certificate
- Import a certificate in the FME Server Java keystore
- Change FME Server Web Services to use HTTPS URLs
Chapter Review

This chapter introduced you to FME Server security.

**What You Should Have Learned from this Module**

**Theory**

- **Users** are the individual accounts that access FME Server.
- **Roles** are comprised of one or more users.
- Default roles and accounts are created when FME Server is installed, but you can also add your own.
- You can integrate your organization’s Active Directory users and groups into FME Server.
- The **Trusted User Account** provides unauthenticated access to any component of FME Server.
- FME Server Windows Services can be changed to any **Service Account** to allow network access.
- FME Server can be configured for HTTPS, supporting both CA-issued and self-signed certificates.

**FME Skills**

- The ability to create a new user and assign permissions.
- The ability to create a connection to an Active Directory server and import users and roles.
- The ability to encrypt the communication between the client and server machines with HTTPS.
Questions and Answers

Here are the answers to the questions in this chapter.

**Miss Vector says...**

If I want one user to have a higher level of access to other users in the same role (say I wish to let an FME author be able to manage engines) what should I do?

1. **Simply select that user from the user list and enable the manage Engines & Licensing policy**
2. Promote that role to superuser status so that the user has a higher level of security
3. Create a new role with the manage Engines & Licensing policy enabled and move that user to it
4. **Create a new role with the manage Engines & Licensing policy enabled and add that user to it as well as the original role**

Security policies can be set at both the user and role levels. While you can create a new role and assign the user to it (Option 4) – enabling the user to be a member of two roles – it is much easier to simply edit the permissions of the individual user (Option 1).
Scalability and Performance
**Job Queues**

Job queues are a mechanism for sending specific jobs to specific FME Engines. The reasons for using job queues include:

- Sending jobs to an FME Engine in close proximity to a data source
- Sending jobs to an FME Engine that supports a particular format
- Reserving an FME Engine for a scheduled task
- Reserving an FME Engine for quick jobs

When you create a job queue, you assign one or more FME Engines to the queue. Then, when you run a job, you can specify which queue to handle the job – this ensures that only the specified FME Engines will process that job.

By extension, you can also assign a repository to a queue. By default, all jobs are assigned to the queue of their respective workspace repository, unless another queue is specified for the job.

All engines and repositories must be assigned to a queue. If an engine or repository is not assigned to a queue explicitly, it is assigned to the Default queue.

Queue priority can be set on each queue. Priority must be an integer between 1 and 10 and defaults to 5 if not set explicitly.

The highest priority is 1 and the lowest priority is 10.

---

**Sister Intuitive says...**

*When importing from older versions of FME Server using the Backup and Restore command, any historic priority will be restored. You want to review the adjusted priority. In previous versions of FME Server these were on a scale of 1-200 and will be adjusted to be between 1-10. If a Queue with the same priority does not exist when a new job runs it will be automatically created and added to the Engines. Review Job Directives and the backward compatibility note on tm_priority.*

---

**NEW**

"Job Queues" used to be referred to as "Job Routing" in older versions of FME Server.

In older versions, job tags were created in either configuration files or via the FME Server REST API.
Exercise 1: Job Queues

<table>
<thead>
<tr>
<th>Exercise 1</th>
<th>Job Queues</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data</strong></td>
<td>N/A</td>
</tr>
<tr>
<td><strong>Overall Goal</strong></td>
<td>Send a job through a specific FME Engine</td>
</tr>
<tr>
<td><strong>Demonstrates</strong></td>
<td>Creating a job queue and assigning jobs to queues</td>
</tr>
<tr>
<td><strong>Start Workspace</strong></td>
<td>None</td>
</tr>
<tr>
<td><strong>End Workspace</strong></td>
<td>C:\FMEData2018\Workspaces\ServerAdmin\Scalability-Ex1-JobQueues-Complete.fmw</td>
</tr>
</tbody>
</table>

Your GIS department is all onboard with FME Server and translating jobs with the web interface, but jobs are always being queued, even the quick translations. You are wondering if there is a way to set aside one of the FME Server Engines for quick translations only so that you and your fellow technical analysts do not have to wait too long for your smaller jobs to complete.

With job queues, you can allocate specific engines to specific tasks.

1) Create a Job Queue

Job queues are created in the FME Server web interface.

Login to the FME Server web interface and select **Admin > Engines & Licensing > Configure** in the left sidebar.

Scroll down to the bottom of the Engines & Licensing page and select **Create Queue**.

![Create Queue](image)

Give it the name *Quick Translations* and click OK.

2) Assign FME Engines

Now that the job queue has been created, specific FME Engines – and repositories – can be assigned to the queue.

Click on the *edit button*. Give the Job Queue the description of "FME Server Engine for Quick Translations," then select `<localhost>_Engine1` from the drop-down selection for Engines.
Next assign a job priority of 1.

To save your edits click the edit button again.

3) Create FME Workspace
To confirm that the job queue is operating correctly, we can run a workspace in FME Server that specifies the Quick Translations queue. For this exercise, we do not need a complicated workspace, just a job that will run.

Open FME Workbench and create a new Blank Workspace.

Add a Creator transformer and connect it to a Logger transformer.

4) Publish to FME Server
Publish the workspace to FME Server by selecting Publish to FME Server from the File menu in FME Workbench:

When prompted in the Publish to FME Server Wizard, connect to your FME Server then publish the workspace to:

- **Repository Name**: Training
- **Workspace Name**: JobQueue_TestJob.fmw
- **Service**: Job Submitter

**Miss Vector says...**

If you have completed the Configure for HTTPS exercise, remember that the URL to connect to FME Server is now [https://localhost:8443/fmeserver](https://localhost:8443/fmeserver) and NOT [http://localhost/fmeserver](http://localhost/fmeserver)!
5) Assign and Run Workspace in Job Queue
Back in the FME Server Web Interface, once you have a published to FME Server, you can run the `JobQueue_TestJob` workspace and set the Job Queue parameter.

Select **Run Workspace** in the left sidebar of the FME Server web interface.

On Run Workspace page, fill out the parameters as follows:

- **Repository**: Training
- **Workspace**: JobQueue_TestJob

Next, expand the **Advanced** options on the Run Workspace page. Set the **Job Queues** parameter to **QuickTranslations** (the name of the queue created in Step 1):

Click **Run** at the bottom of the Run Workspace page.

6) Verify Job Queue Configuration
You want to make sure that the job was routed to the correct engine and not just the first available engine.

In the left sidebar of the FME Server web interface select **Jobs > Completed**.

Select the workspace that just ran to open the **Job Details** page.

Click to expand the **Request Data** section. Next to the **queue** parameter, you will see the name of the specified job queue:
Exercise 1: Job Queues

Go back to Jobs > Completed to verify that the job was sent to the correct engine.
When testing, you may consider submitting the job multiple times for an added verification step, and peace of mind, but this isn't necessary of course!

CONGRATULATIONS!

By completing this exercise you have learned how to:

- Create a Job Queue
- Successfully route a job through a specific engine
Adding FME Engines on a Separate Machine

You can add processing capacity to your FME Server by installing additional FME Engines on a separate computer from the FME Server Core.

The number of licensed FME Engines you can add is limited only by the host’s CPU and memory resources, which constrain the maximum concurrent request throughput.

The additional FME Engines can be of any architecture (32- or 64-bit) and installed to any supported operating system (Windows or Linux) - they do not have to match the specifications of the FME Server Core. It is important to note that the major versions must match - you should not install FME Server 2018 and try to add engines from FME Server 2017.

Keep your FME Engines Close to the data

One of the main reasons a user may want to do this, is to get an FME engine closer to the data, for example, the data may be located in a different office and it makes more sense to process the data in the geographical office next to the physical location versus involving large distances and network latency.

Another reason is to gain access to 3rd party formats that may not be installed on the FME Server Core system.
Changing Database Provider for FME Server Database

FME Server comes equipped with a PostgreSQL Database completely configured and ready for use. However, you may want to leverage a system that is already established or is restricted by your company policies.

If you want to change the database provider for your FME Server database, and you have already installed FME Server - for example, an "Express" installation - you can do this by performing an "in-place" backup and restore procedure:

1. Backup your FME Server configuration.
2. Configure the new database server.
3. Update the database connection settings in an FME Server configuration file.
4. Restart FME Server.
5. Restore your FME Server configuration.
6. Remove dependency, disable, and stop the previous database service.

FME Server supports PostgreSQL, Microsoft SQL Server, or Oracle databases.

Mr. Flibble says...

Even if you plan on changing the FME Server database provider, it is recommended to install with the default PostgreSQL database - this allows you to verify if the FME Server installation is successful.

NEW

Any time the database provider was changed in FME Server 2017 and older it was necessary to run the post-installation scripts. New in FME Server 2018.1 this step will not be required. In FME Server 2018.0 only one post-installation step is required as will be seen in Exercise 2.
Exercise 2: Changing the FME Server Database Provider

Data

| C:\Program Files\FMEServer\Server\database\sqlserver\sqlserver_createDB.sql |
| C:\Program Files\FMEServer\Server\database\sqlserver\sqlserver_createUser.sql |

Overall Goal

Change the database provider for FME Server

Demonstrates

Configuring a new SQL Server database

Your company has an Express Installation of FME Server already installed, but your Database Administrator has just informed you that you that the company will be switching its database providers from the default FME Database to a SQL Server database to allow for more control over database security.

Miss Vector says...

If you have completed the Configuring for HTTPS exercise, remember that the URL to connect to FME Server is now https://localhost:8443/fmeserver and NOT http://localhost/fmeserver!

1) Backup FME Server

Backing up your current FME Server instance is an important step before performing any post-installation configurations. Once you have changed the database provider, you can restore FME Server configurations all at once instead of having to go through the process of manually changing any settings, republishing workspaces, etc.

Note: If you have already created an FME Server backup you can use this existing .fsconfig file and continue to Step 2.

Open the FME Server web interface, either through the Web Interface option on the Windows Start Menu or directly in your web browser, and log in using the username and password admin.

Find Backup & Restore in the left sidebar, under the Admin heading in the FME Server web interface, and click to expand, then click Backup.

Select Download to save a backup file of FME Server - this can be thought of as a "snapshot." It will take a short time to run processes in the background to compile the FME Server backup, and once this is complete, it will automatically save to your local downloads folder.

TIP

If you have a PostgreSQL installation instead of SQL Server, you can follow the instructions in the Server Administrator 2017 Course. Just note that the path names might have changed slightly from 2017 to 2018.

2) Initial Database Configuration

For the purposes of this exercise a separate SQL Server database has been installed to the Training Machines.

The two SQL scripts we'll be using for this step and the next (3) can be found at: C:\Program Files\FMEServer\Server\database\sqlserver\sqlserver_createDB.sql

- sqlserver_createDB.sql
- sqlserver_createUser.sql
We need to create the FMESERVER database schema by configuring the local SQL Server database for FME Server. From the Windows Start Menu open Command Prompt.

First, create a new database using the sqlserver_createDB.sql script. You can review the SQL scripts in a text editor if you wish, but it isn’t necessary. In the Command Prompt run the following:

```
sqlcmd -S FMETRAINING -i "C:\Program Files\FMEServer\Server\database\sqlserver\sqlserver_createDB.sql"
```

The output should look like this now in the Command Prompt window:

```
(1 rows affected)
(1 rows affected)
(1 rows affected)
(1 rows affected)
```

This confirms the new Database “fmeserver” was created. This SQL script creates all FME Server related tables, indexes, views, and triggers.

3) Create the FME Server Database User

Next, we will create the new user fmeserver and grant all necessary permissions to the new user. This SQL script creates a new user fmeserver with password fmeserver. It uses the sqlserver_createUser.sql script.

```
sqlcmd -S FMETRAINING -i "C:\Program Files\FMEServer\Server\database\sqlserver\sqlserver_createUser.sql"
```

The new user will be created with the appropriate login and permissions to access the fmeserver database.

```
(1 rows affected)
```

Exit the Command Prompt.
5) Configure the Database Connection

Open the fmeCommonConfig.txt file, located in the C:\Program Files\FMEServer\Server\ directory, using a text editor in administrator mode.

Under the heading FME SERVER SETTINGS START, locate the section titled Database Connection. We want to disable the default connection to the Postgres database and instruct FME Server to connect to the SQL Server database.

Comment out the DB_TYPE=postgresql section, by adding a number sign (#) in front of each line and uncomment the DB_TYPE=sqlserver section. The final edits should look as follows:

```plaintext
#DB_TYPE=postgresql
#DB_DRIVER=org.postgresql.Driver
#DB_JDBC_URL=jdbc:postgresql://localhost:7082/fmeserver
#DB_USERNAME=fmeserver
#DB_PASSWORD=fmeserver
#DB_CONNECT_EXPIRY=60
DB_SQLSTMTS_PATH=C:/Program Files/FMEServer/Server/database

DB_TYPE=sqlserver
DB_DRIVER=com.microsoft.sqlserver.jdbc.SQLServerDriver
DB_JDBC_URL=jdbc:sqlserver://localhost:1433;databaseName=fmeserver
DB_USERNAME=fmeserver
DB_PASSWORD=$FME$1ser$ver$
DB_CONNECT_EXPIRY=60
DB_SQLSTMTS_PATH=C:/Program Files/FMEServer/Server/database
```

Save and close the fmeCommonConfig.txt file.

6) SQL Server Modifications

From the Start menu, Open Microsoft SQL Server 2016 > SQL Server Management Studio Accept the default server name FMETRAINING and Windows Authentication and click Connect.

The first step is to set the Database Server to allow for SQL Server Authentication. This will permit the new fmeserver user to connect to the database.

In the Object Explorer tree right-click on the FMETRAINING and select Properties.
In the resulting dialog click on Security and select the SQL Server and Windows Authentication option.
Click OK, acknowledging that SQL Server requires a restart.

In the Object Explorer tree right-click on the **FMETRAINING** and select Restart.
And when prompted click Yes.

Finally, Restart FME Server. From the start menu select FME Server 2018.0.0.3 > Restart FME Server.

6) Post-Configuration Script
Starting in 2018.0 the Post-Configuration scripts are now run by the Core when connecting to a new FME Server System Database. There is one script that still needs to be run (step follows Add Queue). Allow a few minutes for the scripts to run and complete loading the new metadata into the database. Attempting to log in during this time may result in an incomplete Web UI appearing. Wait a few moments and refresh the browser.

Add Job Queue metadata: There is one script that still requires to be run manually (this is resolved in 2018.1). Open C:\Program Files\FME Server\Clients\utilities\, and invoke addQueueNode.bat by right-clicking the file and selecting Run as administrator. (only for 2018.0)

This script will take a moment to run and create the default job queue. Next, Restart FME Server. From the start menu select FME Server 2018.0.0.3 > Restart FME Server.

7) Restore Your FME Server Configuration
Since a backup of FME Server was created in Step 1, we can now restore that same FME Server instance which contains all of the previous FME Server configuration settings.

Login to the FME Server web interface and select Backup & Restore > Restore from the left sidebar.

Upload your saved backup configuration file from the beginning of this exercise. Navigate to C:\Users\Administrator\Downloads\ and locate the FME Server backup file (Hint: It has .fsconfig extension!). Drag-and-drop this file onto the FME Server Restore page:
The FME Server web interface will report if the restore is successful. If it is not, the log file is easily accessible from this page if further investigation is needed.

8) **Update Service URLs to HTTPS** Finally, since restoring the configuration does not update the service URLs we must redo the step from Chapter 3, Exercise3. The reason the service URLs are not updated when restoring a configuration is that we may be restoring to an entirely different system with different URLs.

To enable SSL for FME Server Services, login to the FME Server web interface (username and password **admin**), and select Services on the left sidebar.
On the Services page, you can update specific services or all services at once. Let's update all services. Click **Change All Hosts**

The **Change All Hosts** dialog opens. Make sure Host is set to `https://localhost:8443` and click OK.
CONGRATULATIONS!

By completing this exercise you have learned how to:
- Change the database provider for FME Server
System Cleanup

When FME Server is used heavily for a long period of time, a number of files can build up and use system resources. These files are either resource files (including multiple types of log files) or are job history records.

Both of these are cleaned up automatically by FME Server using tasks defined on the System Cleanup page of the FME Server web interface.

Resources

The System Cleanup page looks like this:

Notice that there are multiple types of logs and results files that can be cleaned. Each of these entries represents a task that is run automatically by FME Server from time-to-time. When the task runs and finds files of the specified type, that are older than the specified age, those files are deleted.

Should you wish to keep the files for longer than the pre-defined period you may either edit the file age setting (click on the specific System Cleanup Task to open a dialog for this), select and remove the task, or simply disable that task.

It’s also possible to set up new tasks that search for files and clean them up. For example, you might create a cleanup task to remove files that are periodically uploaded to a resources folder.

Delete_Job_Logs Configuration

The Delete_Job_Logs configuration looks like this:
It's a lot more simple compared to resource file cleanup tasks because there is only one type of information to specify. In this case, you only need to set the maximum length of time that job history is kept for before being removed - and do not have to worry about setting the filter type or pattern.

Miss Vector says...

What exactly are the entries in the resources cleanup dialog?

1. They are simply shortcuts to workspaces in the utilities category that I can run on demand.
2. They are simply shortcuts to scheduling tasks that run at the described interval.
3. They are specific tasks that FME Server runs once a day to help in system maintenance.
4. They are specific tasks triggered when the system is low on resources.
Chapter Review

This chapter introduced you to concepts and configurations regarding FME Server scalability and performance.

What You Should Have Learned from this Module

Theory

- Job Queues are used to send specific jobs to specific engines.
- Job Queues can be used to assign specific repositories to specific engines.
- Queue Priority can be set on the Job Queue.
- You can have as many FME Engines as you want so long as the host's CPU and memory resources can handle the number.
- You can change your database provider at any time. PostgreSQL, Microsoft SQL Server, and Oracle databases are supported with FME Server.
- FME Server periodically clears out resources and job history records that are older than the specified age.

FME Skills

- The ability to configure Job Queues.
- The ability to route a job through a specific engine.
- The ability to change the database provider for FME Server.
- The ability to configure a database for use with FME Server.
## Miss Vector says...

**What exactly are the entries in the resources cleanup dialog?**

1. They are simply shortcuts to workspaces in the utilities category that I can run on demand.
2. They are simply shortcuts to scheduling tasks that run at the described interval.
3. **They are specific tasks that FME Server runs once a day to help in system maintenance.**
4. They are specific tasks triggered when the system is low on resources.

These are their own set of tasks, separate to anything defined as a schedule or utility workspace. They are run once a day and remove any specified file older than the set time. However, #4 is partially correct - when the system’s free disk space reaches specified minimum levels, FME Server will trigger a critical level cleanup event in which all enabled cleanup tasks are carried out. The difference being it will disregard the Remove Files Older Than setting.
FME Server Customization
Server Dashboards

The FME Server Dashboards panel is used to display reports that show the general health of the FME Server installation.

There are five default reports that exist in FME Server:

<table>
<thead>
<tr>
<th>Name</th>
<th>Last Updated</th>
</tr>
</thead>
<tbody>
<tr>
<td>FailuresByWorkspace</td>
<td>Today at 12:21:37</td>
</tr>
<tr>
<td>DailyTotalRunningTime</td>
<td>Today at 12:21:37</td>
</tr>
<tr>
<td>DailyAverageQueuedTime</td>
<td>Today at 12:21:36</td>
</tr>
<tr>
<td>DailyTotalJobs</td>
<td>Today at 12:21:35</td>
</tr>
<tr>
<td>AverageRunningTime</td>
<td>Today at 12:21:35</td>
</tr>
</tbody>
</table>

Each of these reports are simple static HTML files. Clicking on any of the links will open the HTML file in your web browser and display the report. Here is an example of the Job Report HTML file:
In the above image, for example, the report shows how many jobs have been run on FME Server per day.

Police Chief Webb-Mapp says...

These reports are generated using - what else - FME workspaces! These workspaces use the HTMLReportGenerator transformer and HTML format writer to create the graphs shown in reports.

These are fairly simple workspaces and it is entirely possible to create your own reports - preferably by using an existing report as a template workspace. The Dashboards documentation explains how, but - in brief - any HTML document that is written to the folder resources\dashboards\dashboards will appear as a report in the Dashboards panel of the server web interface.

Generating and Using Reports

The reports in the dashboard panel are not generated and/or updated by default. To have them appear - and be updated - you must carry out the following steps:

1. In the FME Server web interface, click Run Workspaces and choose the Dashboards repository.
2. Run the workspace JobHistoryStatisticsGathering.fmw to generate updated statistics.
3. Return to Run Workspaces and choose the Dashboards repository again.
4. Run the workspace for the report you wish to add/update; for example DailyTotalJobs.fmw.
5. Click Dashboards.
6. Click on the chosen report to see the latest statistics.
The workspace JobHistoryStatisticsGathering.fmw is also set up to run on a schedule. The schedule is called DashboardStatisticsGathering, runs every 24 hours, and is set to **Disabled** by default. Therefore to get daily statistics - automatically - requires this schedule to be first enabled.
As your role as FME Server Administrator, you are responsible for managing and maintaining the health of your company's FME Server. To easily monitor the health FME Server has default dashboards that can be enabled.

**WARNING**

*Only complete step 1 if you have finished the Configuring for HTTPS exercise. If you did not complete this exercise, move on to step 2.*

1) **Update workspaces**

   It is a known issue that when you configure FME Server with a self-signed certificate that Dashboards will fail. To fix this we will need to republish the dashboard workspaces that have a parameter changed in the HTTPCallers. For more detailed information about this issue, see the FME Server Dashboards Troubleshooting article.

   In FME Desktop, open the Customization-Ex1-JobHistoryStatisticsGathering.fmw workspace and publish it to FME Server. Ensure that the Repository is set to Dashboards and rename the workspace to JobHistoryStatisticsGathering.fmw. When publishing, a warning message will appear, click Yes to overwrite the existing workspace. This is the controller workspace, now that it is updated, we can run our other dashboards.

2) **Login to FME Server**

   Open the FME Server web interface, either through the Web Interface option on the Windows Start Menu or directly in your web browser, and log in using the username and password **admin**.

3) **Run dashboard workspaces**

   We first need to enable dashboards before they will appear on the Dashboards page. FME Server comes with five default dashboards which are triggered by running a workspace.

   On the side menu bar click on Run Workspace. On the Run Workspace page, select Dashboards as the repository. Then select JobHistoryStatisticsGathering.fmw as the workspace. In the parameters set the username and password to **admin** and click Run:
When the workspace has finished running, go back to the Run Workspace page, and run another dashboard. On the Run Workspace page, select the DailyTotalRunningTime.fmw and then click Run.

Run one more workspace so that we will have some data to look at. On the Run Workspace page, change the Repository to Samples and select austinApartments.fmw and click Run.

4) View a dashboard
On the side menu bar click on Dashboards. This is the Dashboard page where any dashboards that have been run can be viewed.

Select the DailyTotalRunningTime to view the dashboard. Since we've only ran the austinApartments.fmw workspace once, our dashboard doesn't tell us much, but overtime dashboards become a valuable resource for the server administrator.
As an advanced exercise, set up a schedule to run all of the dashboards once daily, so that when you start your day, you will have an up to date snapshot of how your server is performing.

**CONGRATULATIONS**

By completing this exercise you have learned how to:
- Set up an existing dashboard
- View a dashboard in FME Server
- Update the dashboards for self-signed certificates
32- and 64-bit Engines on Same Machine

In some cases, you may want to install 32- and 64-bit Engines on the same machine. For example, you may currently be running only 64-bit engines, but you want to run a workspace from FME Server containing a format that runs only on a 32-bit engine.

This information is included in the FME Server Admin training guide so to make you, the FME Server Admin, aware that this configuration can be done.

Please visit the existing FME Server Admin Documentation for more information. Installing 32- and 64-bit Engines on the Same Machine

An alternative to installing a different bit engine on the same system is to have a distributed FME Server system where you install the 32-bit engines and join this to the existing core system. This requires performing a distributed installation, where the FME Engine is distributed onto a separate system. Refer to the FME Server Admin Documentation for more information on Distributed Installations. Distributed Installation

Police Chief Webb-Mapp says...

*WARNING! This procedure removes the ability to configure FME Engines on the Engines & Licensing page of the web interface. Run the same bit version of FME Engines unless absolutely necessary.*

Use Job Queues to configure specific jobs to run on the new engine. It is important to understand when installing different bit engines that it is necessary to direct the jobs to the correct engine. A format that requires 32-bit engine must be directed to the 32-bit engine otherwise the job will fail when attempting to run on the 64-bit engine. Making use of the Job Queue functionality will be key if you do decide to use mixed bit engines in your FME Server environment.

Miss Vector says...

*Which bit combinations are possible on the same machine?*

1. 32-bit engine on a 64-bit machine.
2. 64-bit engine on a 64-bit machine.
3. 64-bit engine on a 32-bit machine.
4. 32-bit engine on a 32-bit machine.
Configuring Custom Coordinate Systems/Grid Transformations

Workspaces that are run from FME Server can use custom formats, custom transformers, and custom coordinate systems in the same way that FME Workbench does. There are three ways to upload these shared custom resources to FME Server:

- **Use the Resources page of the FME Server web interface**: this method makes custom formats, custom transformers, and custom coordinate systems available to all workspaces on FME Server.
- **Publish to FME Server**: makes a custom format or custom transformer available to all workspaces in the same repository to which it is published.
- **Publish to FME Server with a Workspace**: this method allows you to add resources when publishing a workspace. You can upload the resources to a repository or a shared resource folder. If you upload to a shared resource folder, the resources are available to all other workspaces on FME Server. If you choose a repository, the resources are available to all other workspaces in the same repository to which the current workspace is published.

If custom coordinate system files are uploaded to any of the applicable Engine subfolders, the FME Server Engines service must be restarted before they can be used in a workspace. These include the CoordinateSystemsExceptions, CoordinateSystemGridOverrides, CoordinateSystems, and CsmatTransformationExceptions subfolders.
Using Python with FME Server

If you run workspaces from FME Server that reference Python, the FME Engines must know which Python interpreter to use. For workspaces built with FME Desktop 2016 or later, the interpreter is determined by the setting of the Python Compatibility workspace parameter. For workspaces built with previous versions of FME Desktop, the interpreter is determined by the Preferred Python Interpreter setting.

Changing the Python Interpreter

For workspaces built with the FME Desktop 2016 or later, the Python interpreter you want the FME Engines to use must be specified in the workspace, according to the Python Compatibility workspace parameter.

To use a custom Python interpreter, regardless of workspace version, additional configurations are required. Review Using Python with FME Server for more information.

**TIP: Python 2.7 Deprecation**

*In FME Desktop and FME Server 2018.1 and greater Python 2.7 will be deprecated. Please review the article, Python 2.7 Deprecation, for more information.*
Using R with FME Server

To initiate R scripts from FME Server, such as through the RCaller transformer in an FME Desktop workspace, you must perform the following on all machines that run FME Engines:

1. Install R.
2. Install the sqldf package for R.

Alternatively, you can direct FME Server to an R interpreter that is installed in a non-standard location. For more information on using R in FME, see the RCaller documentation.
Adding Shared Resources

The Resources page is a convenient way to store and access the following:

- FME Server backup files.
- HTML reports generated for Dashboards.
- Data files that can be shared between different workspaces that are run from FME Server, regardless of the repository.
- FME Engine Resources, including custom formats, custom transformers, custom coordinate systems, and custom Java and Python modules that can be shared between different workspaces that are run from FME Server.
- FME Server and database log files.
- Connections to your own network resources and Amazon Web Services (AWS) S3 buckets.
- Temporary files created by FME Server.

Uploading Files and Folders

1. To overwrite existing files with the same names as those being uploaded, select **Overwrite Existing Files**.
2. Select **Upload** and specify whether to upload an entire folder or specific files. Alternatively, drag and drop a file or folder on to the page.

Creating New Network Resources

New in 2018.0 you can create new network shares in the FME Server Web Interface. Ensure the FME Server Web Application, FME Server Core, and FME Server Engine **services** are running as a domain service account with the correct network access permissions as required. *(Read|Write)*
Upon successfully adding the connection it will appear in the Resources page.

**Sharing Resources**

To share a resource, click the *Share with Others* icon. Through sharing, you can grant levels of permissions on an item to other users. You can share an item if you own it, or if you are a user with Manage Security permissions, such as an administrator.

**Miss Vector says...**

*How would you allow other users to access shared resources?*

1. Grant permissions to specific users.
2. Grant permissions to a role.
3. Tag a user you want to share the resources with when you upload the file(s).
Database Connections

Many workspaces that are run from FME Server require you to connect to and authenticate a database service. For example, a workspace that contains an Oracle Spatial Relational reader requires you to provide a service name, username, and password to connect to an Oracle Spatial Relational dataset.

FME Server allows you to store these authentication parameters as self-contained objects, and the Database Connections page provides a convenient and secure way to manage them. When you run a workspace from FME Server that requires a database connection, you can reference the connection name, rather than re-entering the connection parameters. If the connection parameters change, you can edit them in one place, rather than in each workspace that uses the connection.

Using Database Connections

When you run a workspace, any connections that are compatible with the database type in the workspace appear in the Connection drop-down list, under Published Parameters.

Adding Database Connections

There are two ways to add database connections:

1. **On the Database Connections page**: Click New. Or, check the box beside an existing connection and click Duplicate. On the Create New (or Duplicate) Database Connection page, provide a name for the connection, and specify the type of database. Provide the remaining connection and authentication parameters, depending on type. Click OK.

2. **From FME Workbench**: When you upload a workspace that contains a database connection to FME Server, you can specify whether to upload the connection along with the workspace, or use a connection that already exists on FME Server. If you choose to upload the connection, it is added to this page.

Editing Database Connections
Click on a connection name. On the Editing Database Connection page, specify the new parameters, and click OK.

Sharing Database Connections

To share a connection, click the Share with Others icon. Through sharing, you can grant levels of permissions on an item to other users. You can share an item if you own it, or if you are a user with Manage Security permissions, such as an administrator. For more information about sharing and ownership, see Role-Based and User-Based Access Control.

Removing Database Connections

Check the box beside one or more connections you want to remove, and click Remove. In the confirmation dialog, click OK.
Web Connections

The Web Connections page lists the web connections you have uploaded to FME Server along with workspaces.

Many workspaces that are run from FME Server require you to connect and authenticate to a web service. For example, a workspace that contains an HTTPCaller transformer may require you to authenticate to a web service, such as Google Fusion Tables, by providing your username and password.

FME Server allows you to securely store and reference the authentication parameters to these web services as self-contained objects, in the same manner as they can be stored in FME Workbench. When you run a workspace from FME Server that requires a web connection, the credentials associated with the stored connection are also referenced, so you do not have to re-enter them.

Adding Web Connections

There are two ways to add web connections:

1. From FME Workbench: When you upload a workspace that contains a web connection to FME Server, you can specify whether to upload the connection along with the workspace. If you choose to upload the connection, it is added to this page.

2. On the Web Connections page: Click New. Or, check the box beside an existing connection and click Duplicate. On the Create (or Duplicate) Web Connection page, provide a name for the connection and specify the type of service. Provide the remaining connection and authentication parameters, depending on type. Click OK.

Note: You can only add web connections for services to which FME Server is already registered. See 'Managing Web Services' below.

Using Web Connections

To use a web connection, it must be authorized with the web service it references. Workspaces that are uploaded from FME Desktop can run with the authorization credentials provided with the upload, as long as the access token from the web service is still valid. If the access token becomes invalid or cannot be refreshed with the web service, you must authorize the web connection directly with FME Server. Alternatively, republish the workspace from FME Desktop.

To authorize a web connection with FME Server

1. If not already completed, configure the web service definition with FME Server. For more information, see Managing Web Services, below.
2. Authorize the connection:
   - Click on the connection to open it.
   - On the Edit page, click Authorize.

Managing Web Connections

When you upload a workspace from FME Desktop that contains a web connection, a reference to the associated web service is also uploaded. To see the services that are uploaded, click Manage Web Services. The Web Services page opens. (To remove a web service from this page, select and click Remove).

In most cases, FME Server populates your web service credentials from the web connection information uploaded from FME Desktop. These credentials are enough to run workspaces, as long as the web service access token is valid. If the access token is no longer valid or cannot be refreshed, you must complete the web service configuration in order to authorize the connection directly with FME Server.

To configure a web service definition in FME Server
1. Click Manage Web Services.
2. On the Web Services page, click on the web service you want to edit.
3. On the Editing Web Services page, specify the service parameters. For most web services, you must provide authorization and connection credentials:

   **Tip:** For information about creating an app, retrieving the app credentials, and setting the redirect URI, click the link beside External Help Links.

   - **Client ID and Client Secret (OAuth2) or Consumer Key and Consumer Secret (OAuth1):** Credentials associated with an app. You must create an app in the web service and then retrieve the credentials for that app.
   - **Redirect URI (OAuth2) or Callback URL (OAuth1):** A web address that resolves to the location of the fmeoauth web application. If your FME Server is completely installed in the public domain, such as on FME Cloud, the Redirect URI will have been set correctly when the web service was published. If FME Server is on-premises, you will likely need to change the URI. You must set the correct Redirect URI in the app you have created.
     - If your FME Server is configured to access web services through an entry in a public DNS service, use the Redirect URI generated from the service.
     - If your FME Server is configured to access web services through a DMZ, use the Redirect URI `https://<host>:<port>/fmeoauth` for the custom installation of the fmeoauth web service on the public machine.
   - **Click OK.**

**Removing Web Connections**

Check the box beside one or more connections you want to remove, and click Remove. In the confirmation dialog, click OK.
# Exercise 2: Adding a Web Connection

<table>
<thead>
<tr>
<th><strong>Data</strong></th>
<th>C:\FMEData2018\Resources\ServerAdmin\DropboxWebConnection.xml</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Overall Goal</strong></td>
<td>Authenticate a web connection for FME Server</td>
</tr>
<tr>
<td><strong>Demonstrates</strong></td>
<td>How to add and authenticate a web connection for Dropbox</td>
</tr>
<tr>
<td><strong>Start Workspace</strong></td>
<td>None</td>
</tr>
<tr>
<td><strong>End Workspace</strong></td>
<td>C:\FMEData2018\Workspaces\ServerAdmin\Customization-Ex2-WebConnections-Complete.fmw</td>
</tr>
</tbody>
</table>

Your GIS department is working with several other organizations on one big project. It is a lot to organize, so whenever there are additional files to be shared, each organization drops the file into the shared Dropbox for easy access for all organization members. Your task is to configure FME Server to access the Dropbox.

## 1) Create a Workspace

You must first create a Dropbox web connection. The first step in creating this web connection is to have a workspace to run! Open FME Workbench and create a new Blank Workspace.

The **DropboxConnector** transformer can access a Dropbox account and perform Delete, Download, List, and Upload actions.

Add a **Creator** transformer and a **DropboxConnector** transformer to the workspace. Join the Creator to the DropboxConnector.

Add a **Logger** transformer connected to the Output port of the DropboxConnector.

## 2) Configure DropboxConnector and Create Web Connection

Select DropboxConnector and open the parameters dialog or view them via the Parameter Editor pane.

Change the **Dropbox Action** to **List**.

Then select the drop-down for **Dropbox Connection** and select **Add Web Connection**…. The Dropbox Connection dialog box opens.

Set the **Connection Name** to **DropboxWebConnection** and click **Authenticate**….
This opens a new window with a direct, secure connection to Dropbox. Fill in the Web Service Authentication credentials as follows:

- **Email:** fmedropbox@gmail.com
- **Password:** <distributed_during_course>

...and then click **Sign in.**

Note: The above email and password should be used solely for this exercise. You can use your own Dropbox account, but for this course, we have provided an account to use.

Click **Allow** to allow FME to access the Dropbox account.

Your **DropboxConnector** parameters should now look like the following:

![DropboxConnector Parameters](image)

Click **OK** to apply the changes.

3) **Run the Workspace**

It is a best practice to first run the workspace in FME Desktop before uploading it to FME Server. If the workspace does not run in FME Desktop, then it will not run in FME Server!

Click **Run** to make sure the translation is successful. Now we are ready to publish the workspace to FME Server.

4) **Publish to FME Server**

Select **Publish to FME Server** under the File Menu. Use the Publish to FME Server Wizard to place the workspace in the **Training** repository.
On the *Upload Connections* step, make the DropboxWebConnection is selected and click **Next**. We will authorize our web connection using the FME Server web interface.

![Upload Connections](image)

*DropboxWebConnection* will be uploaded, and will need to be authorized on FME Server.

Make sure that the workspace is registered with the **Job Submitter** FME Server Service. Click **Publish**.

5) **Login to FME Server**

Open the FME Server web interface, either through the Web Interface option on the Windows Start Menu or directly in your web browser, and log in using the username and password *admin*.

---

**Miss Vector says...**

*If you have completed the Configure for HTTPS exercise, remember that the URL to connect to FME Server is now https://localhost:8443/fmeserver and NOT http://localhost/fmeserver!*

---

6) **Configure the Dropbox Web Service**

From the left sidebar go to **Connections > Web Connections**.

Click **Manage Web Services** on the Web Connections page.

Select **Dropbox**. The *Editing Web Service "Dropbox" page* opens.

Fill in the *Client Information* parameters as follows:

- **Client Id**: lxx2amcu6xfs11r
- **Client Secret**: <distributed_during_course>
- **Redirect Uri**: https://localhost:8443/fmeoauth
The Client Id and Client Secret are how you connect your client to the web service. They are generated when you create a new API app for a web service. REST API Documentation pages such as this one for Dropbox explains in more detail about web service app creation.

Click OK to save these updates.

7) Authorize Web Connection

Go back to Connections > Web Connections

Select DropboxWebConnection in your list of Web Connections.

On the Edit page, click the Authorize button:
A window opens with the login screen for Dropbox. Sign in with:

- **Email**: fmedropbox@gmail.com
- **Password**: <distributed_during_course>

The window closes, and a message pops up:

![Authorized Successfully]

You have now successfully authorized a Dropbox web connection in FME Server for you to use in your workspaces!

**CONGRATULATIONS!**

*By completing this exercise you have learned how to:*

- Access a web service
- Publish a web connection to FME Server
- Configure a web service in FME Server
- Authorize a web connection on FME Server
Workspace Versioning in FME Server

Workspace Versioning has been added to FME Server 2018.0.

Version control allows you to access previous versions of your repositories files. Optionally, when you configure version control with a remote Git repository, you can access previous versions of files from all members of your team who commit to the same repository.

Note that version control does not, by itself, enable you to update your local working copy of repositories files. Instead, version control allows you to download previous versions. Once downloaded, you can update your working copy by republishing to FME Server.

You can enable Version Control in FME Server via the System Configurations page.

Once turned on users that publish workspaces to FME Server from FME Desktop will have the capability to Commit an FME Workspace and a new version of the workspace file will be saved on your local FME Server system.

When you turn Version Control on it is not configured by default with Github. Instead, all commits are stored in a repository on the local FME Server system. You do not need to configure a remote Github repository to use Version Control.
Backup and Restore does not include version history. If workspace version history is important to you please configure a remote Github repository.

**Using FME Workbench to Version**

When publishing a workspace to FME Server with Version Control enabled the user will see a **Commit** button on the publishing wizard dialog.

![Publish to FME Server dialog](image)

**Using FME Server to Version**

It is also possible to create a version of an existing workspace that has been published to FME Server. This is good for those times when a workspace exists and has been tested on the FME Server Environment and a user wants to create a version of that workspace.
Using FME Server with a Remote Git Repository

For more information on using FME Server with a remote repository, please review the FME Server Admin [Version Control documentation](#).
**Exercise 2**  
**Versioning a Workspace**

<table>
<thead>
<tr>
<th>Overall Goal</th>
<th>Version a Workspace</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demonstrates</td>
<td>How to version a workspace that has been published to FME Server.</td>
</tr>
<tr>
<td>Start Workspace</td>
<td>C:\FMEData2018\Workspaces\ServerAdmin\Customization-Ex3-Versioning-Begin.fmw</td>
</tr>
<tr>
<td>End Workspace</td>
<td>C:\FMEData2018\Workspaces\ServerAdmin\Customization-Ex3-Versioning-Complete.fmw</td>
</tr>
</tbody>
</table>

Your GIS department has two staff members that regularly author and make changes to their own workspaces and others when necessary. On a few occasions, changes were made to workspaces that caused a workspace to fail after the edit was made. The original working workspace was not backed up and was lost. This resulted in extra time to uncover the cause and restore the previous working workspace.

Your Management has learned about the new Version Control feature in FME Server and wonder if it might reduce these issues from occurring. Your task is to enable Version Control and ensure it is functioning correctly.

---

**Miss Vector says...**

*If you have completed the Configure for HTTPS exercise, remember that the URL to connect to FME Server is now https://localhost:8443/fmeserver and NOT http://localhost/fmeserver!*

---

1) **Login to FME Server**

Open the FME Server web interface, either through the Web Interface option on the Windows Start Menu or directly in your web browser, and log in using the username and password **admin**.

2) **Enable Version Control** You can enable Version Control in FME Server via the System Configurations page. Select Features, and then click the toggle next to the Version Control.

---

The toggle will turn green indicating enabled. We do not need to update the **Remote Settings** for basic local Version Control functionality.
3) Create a Workspace
Next, you must create a simple workspace that you can use to test the Version Control feature with.

Open FME Workbench and create a new Blank Workspace.

Add a **Creator** transformer to the workspace.

Save the workspace.

4) Run the Workspace
It is a best practice to first run the workspace in FME Desktop before uploading it to FME Server. If the workspace does not run in FME Desktop, then it will not run in FME Server!

Click **Run** to make sure the translation is successful. Now we are ready to publish the workspace to FME Server.

5) Publish to FME Server
Select **Publish to FME Server** under the File Menu. Use the Publish to FME Server Wizard to place the workspace in a new repository called **Version Test**. Create the new Repository by clicking on **New**... button if it does not exist.
Click OK, to create the new repository and return the Publish to FME Server dialog.

Next, you should see a new button on the Publish to FME Server dialog called Commit....

Click the Commit button, and the following dialog will appear:
Click the checkbox and populate theTextbox with Version 0.1 Initial Workspace and click OK.

Click OK to dismiss the Version Options dialog and return to the Publish to FME Server dialog.
Click Next Make sure that the workspace is registered with the Job Submitter FME Server Service. Click Publish.

6) Review Version History in Web UI

Open the FME Server web interface, either through the Web Interface option on the Windows Start Menu or directly in your web browser, and log in using the username and password admin.

Click on the Repositories in the left menu. This will list all Repositories. Next click the checkbox next to the repository Version Test that you created in step 5 and then click the History button.

The following dialog should be displayed showing the recent Version Control entries for this Repository.
Click **OK** to dismiss the dialog.

6) Downloading a versioned workspace

When it is necessary to download a versioned workspace, and you want to retrieve a particular version you must do this from the Web UI. FME Workbench cannot select a versioned workspace through the Download Wizard. Download the Version 0.1 of the Workspace we just published.

Log back into FME Server as *admin* and in the FME Server Web UI, click on the **Repositories** in the left menu. This will list all Repositories. Next click on the **Version Test** repository. This will display the workspaces inside this repository. You should only see the one workspace called *VersionControlTest.fmw*.

Let's review the Version History again. Click the checkbox next to the workspace and then click the **History** button.

This will return the following dialog window list the available versions.
Let's download this workspace and open it in FME Desktop. In the dialog below click on the Download icon next to the workspace, you wish to download. In this example that is VersionControlTest.fmw & Version 0.1 - Initial Workspace. Remember where you saved the workspace if you selected another location then the default one.

7) Edit workspace and republish same workspace

Next, we want to make a small edit to the workspace and republish it to FME Server. We will then visit the Web UI and Version the workspace thereafter having successfully tested that it works.

Open the workspace that we downloaded in step 6. The default download location is C:\Users\Administrator\Downloads and the workspace should called VersionControlTest.fmw

With the workspace open in FME Workbench, add a Logger to the Creator transformer.
Save the workspace and Run the workspace. Next, let's publish it to FME Server.

This time, **DO NOT** use the commit button on the publish wizard. You will get a warning that the workspace already exists in the repository, this is ok since we have version control enabled. Click Yes to update the workspace. We will commit the workspace using the Web UI in the next step.

8) **Version the Workspace using FME Server Web UI**

We've published the workspace and our colleague has confirmed the workspace runs. Now we will create a new version for the workspace using the Web UI.

Log into FME Server Web UI with the `admin` user.

Review the Version History again for the workspace. Click on `Repositories` in the left menu and navigate to the `Version Test` repository, and finally, click on the repository name to view the contents.

There should be one workspace called `VersionControlTest.fmw`. 
Next, select the workspace and click the Commit button.

This will open the Commit Item dialog. Enter a commit comment and click theCommit button.

You should receive the following success confirmation that the Version was created.
You have successfully enabled and tested Version Control for FME Server.

TIP

If there have been no changes to the workspace since the last commit you will be notified.
This confirms a version of the workspace already exists.

CONGRATULATIONS!

By completing this exercise you have learned how to:
- Enabled Version Control
- Tested Version Control from FME Workbench
- Downloaded a Versioned workspace
- Versioned a workspace in FME Server's Web UI
- Reviewed the history of a Versioned workspace
Reset Password Feature

You can give your Users the ability to reset their passwords for logging on to FME Server. When Reset Password is enabled, a "Forgot your password?" link appears on the login page of the FME Server Web User Interface. This link enables users to receive an e-mail to reset their password.

Configuring Rest Password

You can enable the Rest Password feature in the System Configuration page of the FME Server Web UI. Only members of the fmesuperuser role can enable this feature.

Ensure all users have a valid e-mail address configured as the command requires this information to function correctly.

It is also essential to configure the Email Settings to a valid e-mail server.

TIP

If your FME Server is configured for Active Directory, then your users cannot use this feature to reset their domain password. In this case, users should seek out their IT Admins for password management or recovery.
FME Server Encryption

FME Server encrypts all data in the FME Server Database, as well as passwords and tokens of FME Server configuration backups. By default, this encryption is managed using an encryption key that is common to any FME Server installation. You may wish to enhance encryption security by generating your own custom encryption keys, which you can apply on a rotating basis.

Using your encryption key

When using custom encryption keys, keep in mind the following:

- Do not lose track of any custom keys you generate. Data that is encrypted under a lost key cannot be accessed.
- When performing a Backup & Restore of an FME Server configuration, you must restore to an FME Server that uses the same custom encryption key as the backup.

Enabling Encryption

Only members of the fmesuperuser role can enable this feature.

Using the FME Server Web UI you can enable custom encryption. When enabling custom encryption do allow time for the process to complete.

You can stop using custom encryption in the future by setting the encryption mode back to Secure (Default).
Chapter Review

This chapter introduced you to some of the options for customizing an FME Server installation.

What You Should Have Learned from this Module

Theory

- Dashboards display reports about FME Server general health.
- 32- and 64-bit engines can be installed on the same machine, but a 64-bit engine will not run on a 32-bit version of Windows.
- Custom formats, transformers, and coordinate systems can be uploaded to FME Server.
- Custom coordinate system files uploaded to any of the applicable Engine subfolders will require the FME Server Engines service to be restarted before they can be used in a workspace.
- FME Server is compatible with Python.
- FME Server is compatible with R.
- You can share resources with other users.
- Database connections can be created and saved for future use on the Database Connections page.
- Web connections can be created and saved for future use on the Web Connections page.
- You can version workspaces during publishing or in the FME Server Web UI.
- Passwords can be retrieved using the Reset Password feature
- FME Server can encrypt all the data in the FME Server Database

FME Skills

- The ability to upload a file or folder, share the resource with others, and grant levels of permissions on the item if desired.
- The ability to add and manage web connections.
- The ability to version a workspace during publishing or in the FME Server Web UI.
Questions and Answers

Here are the answers to the questions in this chapter.

Miss Vector says...

Which bit combinations are possible on the same machine?

1. 32-bit engine on a 64-bit machine.
2. 64-bit engine on a 64-bit machine.
3. 64-bit engine on a 32-bit machine.
4. 32-bit engine on a 32-bit machine.

Numbers 1, 2 and 3 are all possible combinations for FME Server.

Miss Vector says...

How would you allow other users to access shared resources?

1. Grant permissions to specific users.
2. Grant permissions to a role.
3. Tag a user you want to share the resources with when you upload the file(s).

Permissions on an item can be granted directly to a user, or indirectly by granting permissions to a role.
Migration and Upgrades
Backup and Migration

To migrate the configuration of an FME Server instance to another instance, perform a backup and restore operation using the tools on the Backup & Restore page of the FME Server web interface.

Make sure to manually back up the FME Server configuration files to a location outside of your FME Server installation directory in case you need to refer to them later. These configurations are not included in the primary backup procedure.

Which FME Server Configuration Components are Migrated?

The following components of your FME Server are migrated:

- Repositories and the items stored in them, including workspaces, custom transformers, custom formats, templates, and resources. Any designations by users of a repository item as a ‘favorite’ are also migrated.
- Version Control information is **NOT** migrated. Only the active workspace is backed up. If a remote Git repository is configured, perform a **Push** before dropping the FME Server system to capture version control information.
- FME Server Services.
- Notifications components, including Topics, Subscriptions, and Publications.
- Security configuration, including Users, Roles, tokens, and Active Directory.
- Schedules.
- Job Queues.
- Resources, which include the following:
  - Resources whose definitions and files are both stored within FME Server, and are configured for migration. By default, the DATA, ENGINE, and DASHBOARD Resources are configured for migration. To determine if default resource folders are migrated, open the FME Server configuration file located at `C:\Program Files\FMEServer\Server\fmeServerConfig.txt`. Resources with a `SHAREDRESOURCE_ISMIGRATABLE_*` value of TRUE are migrated.
  - Online resources whose definitions are stored within FME Server, but whose files are stored outside FME Server. In the current version of FME Server, these resources are limited to any Amazon Web Service (AWS) S3 bucket connections and Network-based resources – for example, directories accessed via UNC. Only connection definitions are migrated (such as bucket names and authentication parameters). The files contained within these resources are not stored on FME Server.
- Database Connections and Web Connections.

The following components of your FME Server are **NOT** migrated:

- Job Logs: Notably the job logs cannot be migrated and must be manually backed up. The default location is `C:\ProgramData\Safe Software\FME Server\resources\logs`.
- Version Control: If the FME Server you are backing up has Version Control enabled the versioning information will not be migrated. If Versioning is important ensure to keep this information in a Remote Repository like GitHub.

**WARNING**

*If Encryption is enabled on the FME Server that is backed up, it must also be enabled on the FME Server that is restored, and use the same custom encryption key.*

*Ensure you have your custom encryption key downloaded and safely saved.*
Sister Intuitive says...

Resource folders containing large amounts of data require considerable time to back up, and result in large configuration files. It is recommended that you define large folders as non-migratable. Following backup and restore, manually migrate these folders to the new FME Server. Alternatively, host big data on a UNC path, then point the new FME Server to that path.

Performing a Scheduled Backup of an FME Server Configuration

FME Server provides a workspace that you can configure to perform regular (scheduled) backups of your Server configuration to an FME Server configuration (.fsconfig) file.

Miss Vector says...

How often can you schedule a backup of your FME Server configuration to occur?

1. Once a year.
2. Once a month.
3. Once a week.
4. Once a day.
5. Once an hour.
Backup Configuration Files

Before proceeding with the primary task of backing up your FME Server configuration, we recommend manually backing up the FME Server Configuration Files to a location outside of your FME Server installation directory, in case you need to refer to them later. These files contain configurations for the FME Server Database, HTTPS, and miscellaneous settings for the Transformation and Repository Managers. These configurations are not included in the primary backup procedure. To restore any changes that were made in these files, you must be able to refer to them when the upgrade is complete so that you can make the corresponding changes in the replacement file.

**WARNING**

*IMPORTANT: When restoring FME Server configuration files, do not replace or overwrite the files in the new installation. The structure of the configuration files *may change between releases*, so it is best practice to go through the old configuration file and the new configuration file together and update configurations in the new file as you come across them. The configuration files support line commenting for tracking specific changes.*

Configuration Files

Depending on the level of customization of your FME Server system you may not need to backup any of these configuration files. For example, an Express Installation of FME Server would not requiring any of these files to be backup. However, if you customized your Express Install of FME Server to connect to external FME Server System Database it might be helpful to back up the fmeCommonConfig.txt file where the external database settings have been updated.

Here is a list of possible configuration files you may backup:

- `<FMEServerDir>\Server\processMonitorConfigCore.txt` - For starting and configuring FME Server Core processes.
- `<FMEServerDir>\Server\processMonitorConfigEngines.txt` - Sets the initial default FME Engines to start, manages engine upgrades, and port numbers for Engine Management requests.
- `<FMEServerDir>\Server\fmeServerConfig.txt` - Configures the FME Server transformation, manager, repositories, shared resources, scheduler, notifications, logs, failover clustering, and others.
- `<FMEServerDir>\Server\fmeCommonConfig.txt` - Configures the FME Server security and the database used by FME Server.
- `<FMEServerDir>\Server\fmeWebSocketConfig.txt` - Configures the FME Server WebSocket server.
- `<FMEServerDir>\Server\fmeEngineConfig.txt` - Configures the FME Engines, including subsections, pre and post commands, and success and failure responses.
- `<FMEServerDir>\Server\config\<subscribers_or_publishers>\websocket.properties` - Configures the FME Server WebSocket server.
- `<WebAppDir>\<fmeServiceName>\WEB-INF\conf\propertiesFile.properties` - Configures properties for FME Server Web Services. Use this file to set the host for the service and various settings.
- `<FMEServerDir>\Utilities\pgsql\data\pg_hba.conf` - Opens the database connections for express install.
- `<FMEServerDir>\Utilities\tomcat\conf\server.xml` - Configures web application server port, and connector protocols.
- `<FMEServerDir>\Utilities\tomcat\conf\web.xml` - Configures international encodings.
- `<FMEServerDir>\Utilities\tomcat\conf\context.xml` - Configures FME Server web application.
Exercise 1: Backup and Migration

<table>
<thead>
<tr>
<th>Data</th>
<th>N/A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall Goal</td>
<td>Perform Backup and Restore operations for an FME Server installation</td>
</tr>
<tr>
<td>Demonstrates</td>
<td>Backing up and restoring automatic and manual FME Server configurations</td>
</tr>
</tbody>
</table>

Your organization has decided to upgrade FME Server to the latest version just released. You want to make sure all the configurations, settings, and resources you have made to your current FME Server will be transferred over to the new FME Server installation. By backing up your current FME Server and then migrating the configurations and settings to the new version you will save lots of time!

Miss Vector says...

If you have completed the Configure for HTTPS exercise, remember that the URL to connect to FME Server is now https://localhost:8443/fmeserver and NOT http://localhost/fmeserver!

1) Connect to FME Server
Open the FME Server web interface, either through the Web Interface option on the Windows Start Menu or directly in your web browser, and log in using the username and password admin.

2) Backup FME Server
When migrating FME Server, you need to back up your current FME Server configurations.

On the left sidebar, click Backup & Restore.

On the Backup page, make sure the default parameter is set:

- **Backup To**: Download

Click the Download button to start the FME Server backup process and save the file to your computer.

3) Backup Configuration Files
It is also a best practice to manually back up any FME Server configuration files you have altered to a safe location outside of the
FME Server installation directory. Certain configurations are not included in the primary backup procedure and you may want to reference the changes made to these files.

In this training course, if you have completed all exercises up until this point, we have modified server.xml, web.xml, context.xml, cacerts, and created a tomcat.keystore file in the Configuring FME Server for HTTPS exercise.

So if we were trying to migrate our current FME Server configurations, these are the files that we would want to have copies of for reference later on when restoring FME Server configurations.

Save these files with your backup configuration file to easily find during a restore.

4) Backup Log Files
Like configuration files, FME Server Log Files are not automatically backed up.

FME Server log files can be found on the Resources page of the web interface in the Logs folder.

For this exercise, we will backup only a couple of the log files. Go to Resources > Logs > core > current and place a checkmark in the boxes beside fmeconfiguration.log, fmeconnection.log, and fmeprocessmonitorcore.log.

Click Download and save these log files in the same location where you saved your BackupFMEServer configuration and your configuration files.

Log files from FME Server can be backed up but can not be restored to a new instance of FME Server. However, it is still a good idea to backup log files in case you need to reference them later on. If you do not backup your log files, they will disappear when you uninstall FME Server.

Dr. Workbench says...

If you want to backup the entire logs folder, you can find it on your file system inside the resources folder of the FME Server System Share.
5) Verify Backup, Install FME Server, Restore Configurations

At this point, during a regular migration workflow, this is when you verify the backup and install your new FME Server. For the purposes of this exercise, we will end the exercise here.

It is vitally important that you verify the FME Server backup first, by installing a new instance of FME Server, before you uninstall FME Server. Once you uninstall there isn't an easy way to backtrack unless you regularly perform backups of your entire file system.

When restoring the configuration files that are manually backed up, it is strongly suggested to go through each configuration file instead of simply copying the old configuration file into the new FME Server directory - file structures and parameters may have changed between releases!

---

**CONGRATULATIONS!**

By completing this exercise you have learned how to:

- Backup your FME Server instance
- Backup additional Configuration Files
- Backup Log Files
Upgrading FME Server

Full FME Server Upgrade

There are multiple approaches to upgrading your FME Server installation, each with its own advantages and disadvantages. The following table outlines several factors for upgrading to a second machine with a different hostname, upgrading to a second machine and using the same hostname, and upgrading FME Server in-place, on the same machine.

<table>
<thead>
<tr>
<th>FME Server configuration remains available from the old installation</th>
<th>Upgrading to a Second Machine, Using a Different Host Name</th>
<th>Upgrading to a Second Machine, Using the Same Host Name</th>
<th>Upgrading In-Place, on the Same Machine</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Existing FME Server installation remains accessible without interruption</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>No DNS entry modification is required by your IT department</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Does not require provision of additional machine</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Users and applications must be updated to access FME Server through a different host name</td>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FME Engine Only Upgrade

FME Engines are the component of FME Server that perform the actual data translations and transformations. At times, you might want to upgrade only your FME Engines and not any other FME Server components. For example, you might want to provide support for recently added formats or other newer functionality included in a new release, which requires only an upgrade of the FME Engines.

Safe supports installing newer FME Engines, as long as the engines share the same major version/year as the current installation of FME Server. For example, on FME Server 2016.0, the FME Engines may be upgraded to 2016.1.3 but may not be upgraded to 2017.0

For 2018.0 there is no support for an Engine only upgrade to 2018.1 due to technical changes that have been made to the product.
Projects

Use the Projects page to bring together FME Server items into groups that share a common solution, or ‘project,’ in your organization. By grouping items in projects, you can reference them in one easy place. You can also export projects and import them to another instance of FME Server.

What Can a Project Hold?

A project can include just about any FME Server object you create or upload, including:

- Workspaces, custom formats, custom transformers, and templates
- Repositories
- Schedules
- Topics, subscriptions, and publications
- Resources
- Resource Connections (currently includes any connections to Amazon Web Services (AWS) S3 buckets that appear on the Resources page)
- Cleanup Tasks
- Database connections and web connections
- Users
- Other projects

Exporting and Importing a Project

You can export the contents of a project, and import the project to the same instance, or another instance, of FME Server. For example, you can migrate a project when you are ready to use it on a production server after you have developed and tested its contents on a separate system.

During export of any project you can choose the Exclude Sensitive Information option. If checked, and the project includes database or web Connections, their associated usernames and passwords are not included in the export.

During import of any project the following options are available:

- Overwrite Existing Items (Notifications & Resources)
- Pause Notification System (Publishers & Subscribers)
- Disable Projects on Import (Schedules, System Cleanup Tasks, & User accounts)

For further information on Export and Importing Projects visit Exporting and Importing Projects

Migrating Projects for Other Users (fmesuperuser only)

When you include users in a project, you can backup and restore a project for those users to work with on another system. Any permissions a user has on the current system are granted on the system to which the project is restored if both of the following conditions are true:

- The permission is on an item in the project.
- The permission was granted to the user directly, rather than through membership in a role.

Any other permissions a user has on the current system do not carry over.

Viewing and Editing Projects
To view or edit a project, click on it to open the *Edit* page. Alternatively, select the project and click **Edit**. You can:

- Edit the project name and description.
- Add and remove items.
- Get information about items: Select an item under *Contents* and click the information icon to the far right for item details.

---

**Miss Vector says...**

*To add users to a project, you must belong to which role?*

1. **fmesuperuser.**
2. **fmeguest.**
3. **fmeauthor.**
4. **fmeuser.**
5. **Any user can add other users to a project.**
Chapter Review

This chapter introduced you to migration and upgrades for FME Server, including FME Server Projects.

What You Should Have Learned from this Module

Theory

- Configuration files need to be manually backed up for reference when restoring any changes made to those files for the new FME Server.
- You are able to backup your repositories, notifications, schedules, security, and services with FME Server.
- You can backup an FME Server configuration to either a resource location or to your local system.
- FME Server can be scheduled for backups.
- FME Server provides a workspace that can be scheduled to perform a backup of the Server configuration to an FME Server Configuration File (*.fsconfig). The configuration backup includes repositories, services, notifications, security settings, and schedules.
- You can upgrade FME Server to a second machine, using a different hostname or by using the same hostname, or by upgrading in-place, on the same machine.
- It is possible to upgrade only the FME Engines for additional functionality of FME Server.
- FME Server Projects are great for bringing together several FME Server items into groups that are part of a common solution, for easy reference.

FME Skills

- The ability to backup your FME Server instance.
- The ability to backup your FME Server configuration files.
- The ability to backup FME Server log files.
- The ability to restore FME Server configurations.
## Questions and Answers

Here are the answers to the questions in this chapter.

### Miss Vector says...

**How often can you schedule a backup of your FME Server configuration to occur?**

1. *Once a year.*
2. *Once a month.*
3. *Once a week.*
4. *Once a day.*
5. *Once an hour.*

You can schedule a backup of your FME Server configuration by the second, by the year, and everything in between. Schedule a backup for the time range that works best for you and your organization.

### Miss Vector says...

**To add users to a project, you must belong to which role?**

1. *fmesuperuser.*
2. *fmeguest.*
3. *fmeauthor.*
4. *fmeuser.*
5. *Any user can add other users to a project.*

To add users to a project you must belong to the fmesuperuser role.
Troubleshooting
Initial Troubleshooting

Verification

After performing each of the following solutions, you can verify if the issue has been corrected by running through a quick verification process for either Windows or Linux.

Restart FME Server

Restart FME Server and make sure all the Windows services are running. If any services fail to start, then reboot the operating system and try to restart the services again.

Look for Orphan Processes

An orphan process is a computer process whose parent process has finished or otherwise ended, though it remains running itself. In this case, we are looking for unintentionally orphaned processes, such as when the parent process ends or crashes. To find these, stop all services listed in the documentation (FME Server System Services), and proceed as follows:

Windows

- Open the Task Manager and look for any orphaned processes starting with FME (often found under the Details tab). If any are found, end those processes and start the services again. Also look for any postgres.exe processes that should not be running when the FME Server System Database is shut down. (Ensure there isn't another PostgreSQL database configured on your system before closing these processes.)

Linux

- Run the following command to list processes:

```
# ps -A
```

Look for the processes as described in the linked documentation article above. For reference, these may include:

- `fme` – one for each running FME Engine
- `FMEServer` - FME Server Core
- `FMEScheduler` - FME Server scheduler - requires the core to be running
- `FMENotifier` - FME Server notifier - requires the core to be running
- `FMESubscriber` - FME Server subscriber - requires the core to be running
- `FMERelayer` - FME Server Relay Manager
- `FMESMTPRelay` - FME Server Email receiver
- `FMEWebSocket` - FME Server WebSocket Server
- `FMECleanup` - FME Server Cleanup service
- `tomcat7` – FME Server Web Application Server
- `postgres` – FME Server Database Server

For example, you might see listed: 53521 pts/3 00:00:03 fme

To stop this process, run the following:

```
# kill 53521
```
**Restart the System**

Restart the operating system. Perform the verification linked above and see if the problem still exists.

**Revert any Recent Changes**

Check if there have been any recent changes to the machine. Has a software update been applied or new hardware added? Configuration files edited? If so, try reverting the system to the state prior to the update and see if that resolves the issue.
FME Server Log Files

Read the FME Server Log Files for error messages. This can be quite helpful to find the cause of an issue.

Access from the Web Interface
Open the web interface and navigate to the Resources page to access the log files.

Access from the File System
If the web interface is not accessible, you can access the log files from the FME Server System Share: \\resources\logs

The FME Server System Share location can be specified at install times. By default, it is set to C:\ProgramData\Safe Software\FME Server on Windows, which is a hidden folder.

What to look for in FME Server Log Files

In the logs there are three types of messages: Inform, Warn and Error. You should be looking for Warning and Errors in the logs. Inform messages may not signify a problem, even if the word “Error” is located within its text.

Example Inform Error:

2018-05-09 13:54:20| 0.9| 0.0|INFORM|FAILURE_RESPONSE !FME_ERROR_NUMBER!:!FME_ERROR_MSG!

Example Warn Error:


Example Error:


If a Warn/Error message is encountered, it may be beneficial to search the documentation and FME Knowledge Center for the error text in order to find assistance.
Additional Troubleshooting

Some additional troubleshooting techniques.

**FME Workbench-FME Server Connection**

If you are unable to connect from FME Workbench to FME Server then the following suggestions may be of help:

- Check if there is a firewall running on either your computer or the FME Server. If so, you must open port 80 (or 8080) to use the Web Connection. A popular tool for checking ports is netstat, a command line utility on Windows, or telnet on Linux systems.
- Restart the FME Server Services. On Windows, go to Start > Programs > FME Server > right-click Restart FME Server and select Run as administrator.

**Web Interface**

If you are unable to access the web interface then the following suggestions may be of help:

- Confirm that FME Server is installed and running! The easiest way to be sure is to restart FME Server using Start > Programs > FME Server > Restart FME Server
- Check whether FME Server was installed using an application server port other than 80. For example, if port 80 was already being used the installer might have used a different port; 8080 is most common. To check, try entering the URL with this syntax http://<host>:<port>/fmeserver - for example, http://localhost:8080/fmeserver

**Workspaces are Queued but not Run**

If a workspace appears in the FME Server queue but is never initiated, then it may be because no engines are running.

- Check the web interface (Engines & Licensing > Engine Management) to confirm engine status. If no engines are available, check licensing and update as necessary. Restart FME Server once a proper license is in place.
- If engine status is good review the job queues to make sure the engines all have queues assigned to them that match the queued job requests.

**Workspaces Fail when Run**

If a workspace fails when it is being run then the following suggestions may help:

- Run the workspace first on FME Desktop. If it does not work there, it may not work on FME Server.
- Data paths can cause problems when moving from a local Desktop machine to a Server environment. Check the dataset parameters (Reader and Writer) to ensure they are not referring to a local path that does not exist on the Server. You may need to change the parameter to use the Resources dialog (Browse) and not a file path dialog (Specify Location). File permissions and the service account used to run the FME Server Windows Services is commonly the fault.
- Check the FME log file using Jobs > Completed in the web interface. This may help to explain why there is a problem.

**Scheduling**

If a scheduled workspace appears to have not been run at the expected time then the following suggestions may be of help:
- Ensure an engine is available, and that the scheduled job is not in a queue.
- Check the date and time very carefully to ensure the correct values were entered.
- Check the timezone is correct. The web interface operates on local timezone, which is not necessarily the same timezone as where the server is physically located.

**Cannot Connect to WebSockets Server**

If you cannot connect to a WebSockets server then the following may be of help:

- Ensure the FME Server's WebSockets port (default 7078) is open.
- Ensure you are using the correct stream_id for sending and receiving between your applications.
FME Community

The FME Community has an extensive collection of articles and Q&A postings. Post a question for others in the community Forums or check if someone else had the same question and already received an answer!

One particular set of knowledgebase articles, the FME Server Troubleshooting Guides, identify several commonly encountered FME Server issues and their resolutions. It is an excellent resource for troubleshooting!
**Troubleshooting Steps:**

If you’ve experienced a problem with FME Server, some initial troubleshooting questions are:

- Have you followed the documentation? Previous documentation versions are available [here](#).
- Has this ever worked for me? If yes, what changed between then and now?
- Are there any error messages?
- Does the log file have any information?
- What type of FME Server installation do I have? (Distributed/Express)
- What version of FME Server am I running?
- Have I done any additional configurations to FME Server?
- If I restart FME Server, does it work?
- Are the correct ports open? You can see ports [here](#) or [here](#).
- What troubleshooting steps have I taken already?

If you ever need to report a problem with FME Server to Safe Software, please provide as many answers to these questions as possible, which will help the expert address your issue.

**Troubleshooting Topics**

- Are you having trouble, or need to find, the log files in FME Server?
- Are you having trouble connecting to FME Server from FME Workbench?
- Are you having trouble upgrading FME Server?
- Are you having trouble uninstalling FME Server?
- Are you having trouble with security and FME Server?
  - Users/Roles
  - Tokens
  - Active Directory
  - HTTPS/SSL
  - IWA/SSO
- Are you having trouble logging in or accessing parts of FME Server through the web interface?
- Are you having trouble accessing your licensing FME Server?
- Are you having trouble with your FME Server engines?
- Are you having trouble accessing your repositories in FME Server?
- Are you having trouble with schedules in FME Server?
- Are you having trouble with web or database connections in FME Server?
- Are you having trouble with projects in FME Server?
- Are you having trouble with the FME Server REST API?
- Are you having trouble with proxies and FME Server?
- Are you having trouble with coordinate systems on FME Server?
- Are you having trouble with dashboards on FME Server?
- Are you having trouble with FME Server Resources?
- Are you having trouble submitting a job to FME Server?
- Are you having trouble with cleanup tasks in FME Server?
- Are you having trouble with the FME Server Job Submitter?

**Are you still experiencing issues?**

Please search for your issue or post your question to the FME Community Q&A.
If you need to report a bug or defect with FME Server, please report a problem.

**Have ideas on how to improve FME Server?**

You can add ideas or product suggestions to our Ideas Exchange.
Course Wrap-Up

Although your FME training is now at an end, there is a good supply of expert information available for future assistance.
Product Information and Resources

Safe Software Web Site

The Safe Software web site is the official information source for all things FME. It includes information on FME products, Safe Software services, FME solutions, FME support and Safe Software itself.

Safe Support Team

Behind FME are passionate, fun, and knowledgeable experts, ready to help you succeed, with a support team philosophy built on the principle of knowledge transfer.

Your Local Partner

Safe Software has partners and resellers around the world to provide expertise and services in your region and your language.
You can find a list of official partners on the Safe Software web site.

Safe Software Blog

The Safe Software blog provides technical information about FME, articles about customers' use cases, and general thoughts on spatial data interoperability.

FME Manuals and Documentation

For FME Server documentation, look on our web site under Learning & Support > Documentation
For FME Desktop use the Help function in FME Workbench to access help and other documentation. Alternatively, look on our website under the Learning & Support > Documentation.
Community Information and Resources

Safe Software actively promotes users of FME to become part of the FME Community.

The FME Community

The FME Community is a one-stop shop for all community resources, plus tools for browsing documentation and downloads.

Knowledge Base

The FME Knowledge Base contains a wealth of information; including tips, tricks, examples, and FAQs. There are sections on both FME Desktop and FME Server, with articles on topics from installation and licensing to the most advanced translation and transformation tasks.

Forums

FME community members post FME-related messages, ask questions, and share in answering other users’ questions. Members earn “reputation” and “badges,” and there is a leaderboard of the top-participating users. Join the conversation to see how the community helps each other with their FME projects!

Ideas Exchange

FME development is very much user-driven. The Ideas Exchange gives users the chance to post their ideas for new FME functionality, or improvements to existing functionality, and allows everyone to vote on the proposed ideas. The more votes an idea gets, the more likely it is to be implemented!
The FME Channel

This FME YouTube channel is for those demos that can only be properly appreciated through a screencast or movie. Besides this, there are a host of explanatory and helpful movies, including recordings of most training and tutorials.
Feedback and Certificates

The format of this training course undergoes regular changes prompted by comments and feedback from previous courses.

Course Feedback

Miss Vector says...

There’s one final set of questions – and this time you’ll be telling me if the answers are correct or not!

Safe Software greatly values feedback from training course attendees, and our feedback form is your chance to tell us what you really think about how well we’re meeting your training goals.

You can fill in the feedback form now, but you’ll also be reminded by email shortly after your course. Safe Software's partners who carry out training may ask that you fill in a separate form, but you can also use the official Safe Software form if you wish.

Certificates

Mr. E. Dict, (Attorney of FME Law)says...

In order to prove you have taken this training course, a certificate will be emailed automatically to anyone who was logged on for the duration of Safe Software hosted courses.
Thank You

Thank you for attending this FME training course.

All of us at Safe Software wish you fun and success as you use FME to speed your data on its way.