

FME® Flow

Administrator's Guide

Version: 2024.0

Updated: March 2024

FME Flow

Administrator's Guide

FME Flow 2024.0



CONTENTS

Flow	1
FME Flow Administrator's Guide	
About This Guide	11
Additional FME Flow Resources	11
Planning an FME Flow Installation	
Things to Consider - An Overview	15
Choosing a Deployment Architecture	15
Express	16
Distributed	17
Fault-Tolerant	17
Distributed Engine	18
Disaster Recovery	18
Distributing FME Flow Components	19
Planning for Fault Tolerance	22
Planning for Disaster Recovery	26
Planning for Security Updates	28
Reporting Potential Vulnerabilities	29
Planning for Scalability and Performance	30
About Component and Job Recovery in FME Flow	32
System Requirements	33
Linux: Offline Installations	33
Linux: Online Installations	35
Requirements for FME Workspaces	37
FME Workbench	37
GridShift Files	38
Third-Party Applications for FME	38
Performing an Installation	
See Also	40
Install FME Flow: Express Installation for Windows	41

Obtain the Installer	41
Run the Installer	41
Starting and Stopping FME Flow	46
Log on to the Web User Interface	46
Request and Install a License	47
Test the Installation	49
Uninstalling	50
Install FME Flow: Express Installation for Linux	50
Obtain the Installer	51
Run the Installer	51
Starting and Stopping FME Flow	53
Log on to the Web User Interface	53
Request and Install a License	54
Test the Installation	56
Uninstalling	57
Installing a Scalable, Fault-Tolerant FME Flow	57
Before Installation	59
Obtain the Installer	62
Install FME Flow on All Servers	62
Configure the FME Flow Database on a Separate Database Server	68
Set Up the Load Balancer and Configure with FME Flow	88
Log on to the Web User Interface	90
Request and Install a License	91
Test the Installation	93
Distributing Components (2-Tier)	94
Before Installation	95
Obtain the Installer	102
Install the FME Flow Database	103
Install the FME Flow Core, FME Engines, and Web Application Server	106
Configure the FME Flow Database on a Separate Database Server	113
Log on to the Web User Interface	133
Request and Install a License	134

Test the Installation	137
Distributing Components (3-Tier)	137
Before Installation	139
Obtain the Installer	146
Install the FME Flow Database	146
Install the FME Flow Core and FME Engines	150
Install the FME Flow Web Services	155
Configure the FME Flow Database on a Separate Database Server	160
Log on to the Web User Interface	180
Update the WebSocket Host	181
Request and Install a License	181
Test the Installation	184
Adding FME Engines on a Separate Machine	185
Performing a Silent Installation (Windows)	191
Extracting Installation Files	192
Installer Flags	192
Specifying FME Flow Components to Install	193
Installation Properties	194
Post-Install Tasks	205
Examples	205
Performing a Silent Installation (Linux)	207
Specifying FME Flow Components to Install	207
Installation Properties	208
Post-Install: MSSQL Formats Only	222
Examples	223
Deploying FME Flow with Docker Compose	224
Limitations and Considerations	224
About the Docker Compose File for FME Flow	225
Deploying and Administering FME Flow on Docker Compose	227
Working with FME Flow Container Environment Variables	231
Upgrading to a Patch Build in Docker	244
Deploying FME Flow with Kubernetes	245

Prerequisites and Considerations	245
Performing the Kubernetes Deployment	246
Defining FME Engines, Queue Control, and Streams to a Kubernetes Deployment	250
Deploying with Kubernetes and a Trusted Certificate	251
Deploying with Kubernetes across Multiple Hosts	252
Deploying with Kubernetes to a Cloud Provider	257
Upgrading a Kubernetes Deployment of FME Flow	285
Upgrading to a Patch Build in Kubernetes	288
Deploying FME Flow in the Cloud	289
Azure Marketplace	289
Terraform	291
Vendor-Specific IaC Tools	291
Installing Remote Engine Services	291
FME Flow Remote Engine Installation Extractor	291
Destination Folder	292
Admin User	293
Web Application Server Port	294
Database User	294

Upgrading FME Flow

Full FME Flow Upgrade	297
Upgrading to a Second Machine, Using a Different Host Name	297
Upgrading to a Second Machine, Using the Same Host Name	297
Upgrading In-Place, on the Same Machine	298
Upgrading to a Second Machine, Using a Different Host Name	298
Upgrading to a Second Machine, Using the Same Host Name	299
Upgrading In Place, On the Same Machine	301
FME Engine Only Upgrade	306
Upgrading FME Engines on Remote Engine Services	307
Obtaining the Installer	307
Install on Windows	308
Install on Linux	309

Upgrading a Kubernetes Deployment of FME Flow	310
Performing the Upgrade	311

Optimizing FME Flow

Configuring the Number of FME Engines to Start	315
To add or remove FME Engines to Start	315
Adding FME Engines on a Separate Machine	315
Remote Engine Services	322
Capabilities and Limitations	322
Getting Started with Remote Engine Services	323
Administering Remote Engine Services	326
Changing the Database Provider for the FME Flow Database	327
Backup Your FME Flow Configuration	327
Configure the New Database Server	327
Configure the Database Connection	328
Restart FME Flow	332
Restore Your FME Flow Configuration	332
Remove Dependency, Disable, and Stop the Previous Database Service	332
Removing the FME Flow Core Dependency on the FME Flow Database	333
Moving the Web Applications Server to Another Machine	333
Install a web applications server on the new host machine (optional)	334
Install the FME Flow Web Services on the new host	334
Update Services URLs	334
Configure CORS filters (optional)	335
Using a Reverse Proxy with FME Flow	335

Securing FME Flow

Security Basics	339
Role-Based and User-Based Access Control	339
Default User Accounts and Passwords	342
Configuring Authentication for the FME Flow Web Services	344
Configuring for HTTPS	345
Configuring Integrated Windows Authentication	365

Updating the Windows Domain Configuration	366
Updating the Web Browser Configuration	368
Update the Tomcat Properties File	370
Troubleshooting Authentication Services	370
Failure to Connect to Directory Server (SSL)	371
Failure to Connect to Azure AD Through IIS Proxy	371
Unable to Use SASL Authentication	372
Unsupported SASL Mechanism	373
Single Sign-On Authentication Failure (Incorrect Mechanism)	373
Single Sign-On Authentication Failure (Negotiation Error)	374
Single Sign-On Authentication Failure (Negotiation Error Encryption)	375
Single Sign-On Authentication Failure (Service Account)	376
Single Sign-On Authentication Failure (Cross-Domain User)	377
Single Sign-On Authentication Failure (Same Machine Access)	378
SASL Authentication Failure 1	378
SASL Authentication Failure 2	379
SASL Authentication Failure 3	380
SASL Authentication Failure 4	380
SAML Authentication Failure	381
Importing a CA Certificate for SSL Connections	382
Running the FME Flow System Services Under Different Accounts (Windows)	383
Encrypting the FME Flow Database Password	384

Other Tasks

Licensing and Administration Tasks	387
Licensing	387
Configuring the FME Engine	389
Working with the FME Flow System Services	398
Purging Logs and History	407
Changing the Web Application Server Host Name and Port	408
Configuring International Character Encoding	409
Moving the FME Flow System Share to Another Machine	411

Providing Support for OGC Services	411
Backup, Restore, and Migration Tasks	411
Migrating an FME Flow Configuration	411
Performing Scheduled Backups of an FME Flow Configuration	412
Notifications Tasks	413
Configuring FME Flow to Receive Email	413
Configuring FME Flow to Send Email Notifications	414
Workflow and Resources Tasks	414
Sending Jobs to Specific Engines with Job Queues	415
Job Recovery	415
Sharing FME Engine Resources	415

Safe Software Inc. makes no warranty either expressed or implied, including, but not limited to, any implied warranties of merchantability or fitness for a particular purpose regarding these materials, and makes such materials available solely on an “as-is” basis. In no event shall Safe Software Inc. be liable to anyone for special, collateral, incidental, or consequential damages in connection with or arising out of purchase or use of these materials. The sole and exclusive liability of Safe Software Inc., regardless of the form or action, shall not exceed the purchase price of the materials described herein.

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.

Copyright

© 2005 – 2024 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. Contact us at:

Safe Software Inc.

Suite 1200 - 9639 137A St.

Surrey, BC Canada V3T 0M1

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 brands and their product names mentioned herein may be trademarks or registered trademarks of their respective holders and should be noted as such.

Document Information

Document Name: FME Flow Administrator's Guide

Version: FME Flow 2024.0

Updated: March 2024

Version Information

FME Flow 2024.0 is Version 2024.7.49 of the FME software.

FME Flow Administrator's Guide

About This Guide	11
Additional FME Flow Resources	11




About This Guide

The **FME Flow Administrator's Guide** is intended for system administrators and those who are responsible for installing and configuring FME Flow in a Windows or Linux environment.

This document assumes that system and FME Flow administrators have knowledge of or access to resources in the following areas:

- The location and structure of the spatial data resources you want to work with in FME Flow
- Your web application architecture
- FME Workbench, if you plan on installing FME Form and authoring workspaces

 **Note** In the printed document format (PDF) version of this guide, links to topics in the FME Flow Reference Manual and FME Flow Web User Interface documentation do not function. The corresponding pages of these topics are available in the FME Flow Documentation online at https://docs.safe.com/fme/html/FME_Server_Documentation/Home.htm.

Additional FME Flow Resources

Safe Software provides the following resources to assist with your learning about FME Flow:

- FME Community is the online source for all information on Safe Software products, including documentation, demos and support resources:
<https://community.safe.com/>

- Product documentation

The most up-to-date product documentation is available on the FME Community.

Installed product documentation is available in the FME Flow installation directory > Docs.

FME Flow product documentation includes:

- FME Flow Administrator's Guide: Information on installation, licensing, upgrading and common configurations.
 - FME Flow Tutorial: An introductory course on authoring and publishing in FME Flow.
 - FME Flow Reference Manual: Developer resources and detailed technical documentation.
 - FME Flow Web User Interface: A guide to using the FME Flow's web interface.
 - FME Flow Developer's Guide: Quick-start, tutorial and sample code for working with the FME Flow APIs.
 - FME Flow REST API Specifications: Developer reference for working with the REST API.
- Training
[Training courses](#) on FME Form and FME Flow (offered by Safe Software and our reseller network).

Planning an FME Flow Installation

Things to Consider - An Overview	15
Choosing a Deployment Architecture	15
Planning for Security Updates	28
Planning for Scalability and Performance	30
About Component and Job Recovery in FME Flow	32
System Requirements	33
Requirements for FME Workspaces	37



Things to Consider - An Overview

When you acquire FME Flow, you must decide how to deploy it on your system. There are a number of considerations:

- [Choosing a Deployment Architecture](#): Learn about your options for how to deploy FME Flow, either as a stand-alone system, or distributed on your network.
- [Planning for Security Updates](#): Understand how security updates on FME Flow-related components may affect how you decide to deploy FME Flow.
- [Planning for Scalability and Performance](#): Following installation, you can scale up your FME Flow by increasing job throughput and optimizing job performance. These are measures you can take anytime after you install. Although they may not affect your initial planning, it's important to understand your options both up front, and going forward as you work with FME Flow.

See Also

- [About Component and Job Recovery in FME Flow on page 32](#)
- [System Requirements on page 33](#)
- [Requirements for FME Workspaces on page 37](#)

Choosing a Deployment Architecture

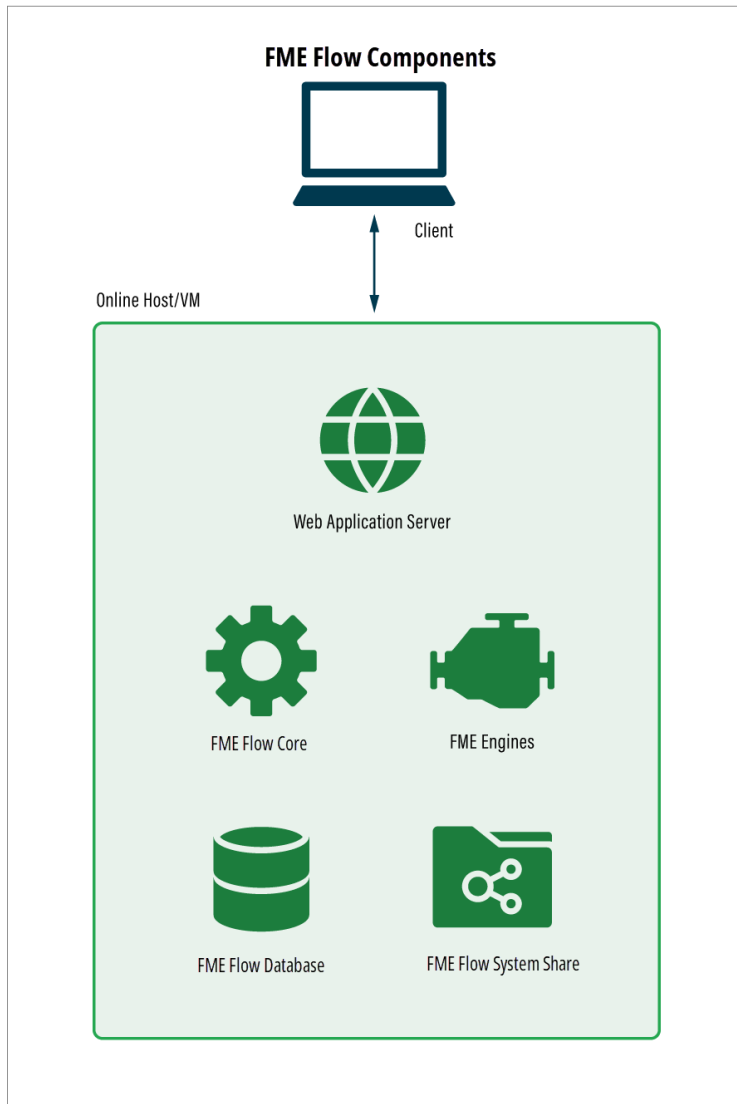
The FME Flow installer provides three options for installation:

- Express
- Distributed/Fault Tolerant
- Distributed Engine

Express

An Express architecture installs all the required components of FME Flow on a single host machine, and is the quickest and easiest way to get started with FME Flow.


Express Install



- [Install FME Flow: Express Installation for Windows on page 41](#)
- [Install FME Flow: Express Installation for Linux on page 50](#)

Distributed

A Distributed installation provides options for spreading components across a network to achieve a 2- or 3-tiered architecture. This approach allows greater control of some components that you provide and maintain yourself, such as a database server for the FME Flow Database, and a web application server. You can provide redundancy with some elements, but if you require high availability, consider a fault-tolerant architecture (see below) instead. If you are considering fault tolerance in the future, you should start with a distributed installation now for an easier transition.


 **Note** The FME Flow distributed and fault-tolerant options share the same installer.

To learn more, see [Distributing FME Flow Components on page 19](#).

Fault-Tolerant

A fault-tolerant installation provides options for spreading components across a network, ensuring high availability, and allows choices to maintain some components yourself, such as a database server for the FME Flow Database, and a web application server.

Comprised of redundant FME Flow Core and web application server components spread across additional host machines, this architecture ensures that if a hardware component fails, FME Flow remains online. This installation requires only a single FME Flow System Share and FME Flow Database server (which, optionally, can serve a custom database with replication.) A third-party load balancer is required, which directs incoming traffic to one of the redundant web application servers.

 **Note** The FME Flow distributed and fault-tolerant options share the same installer.

To learn more, see [Planning for Fault Tolerance on page 22](#).

Distributed Engine

The Distributed Engine installation allows you to build onto an existing FME Flow installation by [Adding FME Engines on a Separate Machine](#).

Reasons you might want to provide a distributed engine include:

- Utilizing an Esri ArcGIS license on a separate host machine.
- Hosting FME Engines on a different OS than your web application server and FME Flow Core.
- Utilizing system resources of other machines.
- Placing FME Engines in close proximity to your data.

For more information, see [Planning for Scalability and Performance on page 30](#).

Disaster Recovery

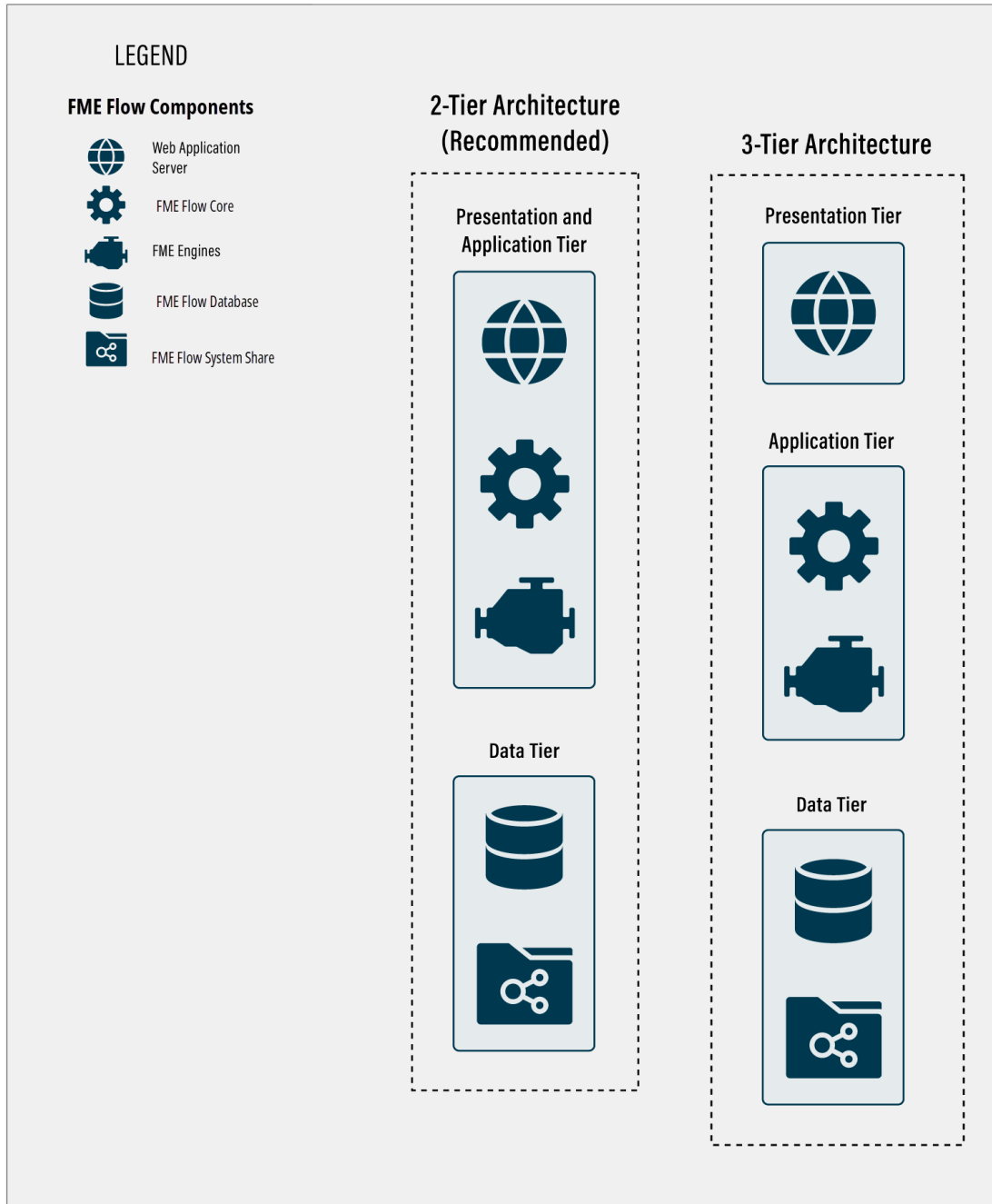
Disaster recovery is primarily concerned with recovering FME Flow operations and data in the event of a major failure of a data center. The general concept of disaster recovery is that if one data center fails, the second data center can be brought online after a manual redirect, and start accepting requests. Disaster recovery can be incorporated into any of the other architectures discussed here.

For more information, see [Planning for Disaster Recovery on page 26](#).

Distributing FME Flow Components

You can distribute the components of an FME Flow installation to achieve an n-tier architecture. Both 2-tier and 3-tier configurations are supported:

Distributed Component Supported Architecture Options



In a 2-tier architecture, an Apache Tomcat web application server runs the [FME Flow Web Services](#), and is installed alongside the the [FME Flow Core](#) and [FME Engines](#). Only the


[FME Flow Database](#) is installed separately, along with a remote file system for the FME Flow System Share.

Installing the web application server on the same host machine as the FME Flow Core is the recommended architecture for a distributed installation.


A 3-tier architecture separates the web application server to another host machine, splitting the presentation and application tiers.


In a 2- or 3-tier scenario, you can provide and manage the following components:

- The web application server provided by the installer is Apache Tomcat. Optionally, you can [provide your own web application server](#), instead of using the one shipped with the installer.
- The FME Flow Database can be hosted on a PostgreSQL database server provided by the installer, or you can [configure the database on your own server](#) using PostgreSQL (recommended), Microsoft SQL Server or Oracle. If using an Oracle database server, you must [obtain the Oracle Database JDBC Driver](#).

 **Note** PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.

- A remote file system to host the FME Flow System Share, which includes [Repositories](#) and [Resources](#).

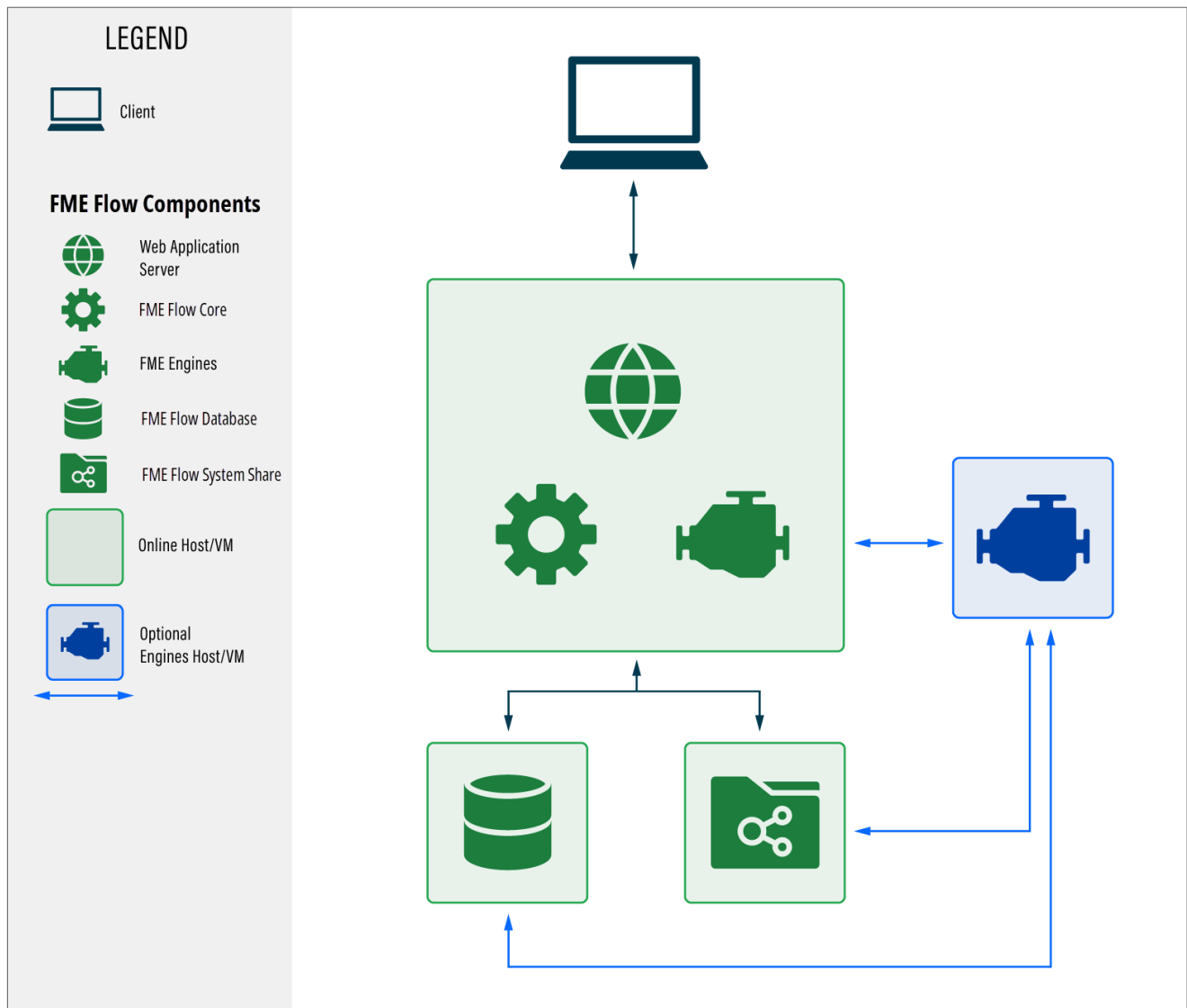
 **Note** Although the FME Flow Database and System Share will not have data redundancy by default, this can be set up by your organization.

 **Warning** We recommend installing the web application server on a system that is synchronized to the same time zone as the FME Flow Core. Differing time zones may result in improper timing of [Schedule Initiated](#) triggers.

Recommended 2-tier Architecture

This diagram demonstrates the recommended 2-tier architecture, with the web application server, FME Flow Core, and FME Engines running on the same host, and separate servers for the FME Flow Database and FME Flow System Share. Optionally, you can [add FME Engines on a Separate Machine](#) or access [Remote Engine Services](#).

Distributed Install



Implementing a Distributed Architecture

Perform a distributed installation of FME Flow according to the 2-tier or 3-tier procedure:

- [Distributing Components \(2-Tier\)](#)
- [Distributing Components \(3-Tier\)](#)

Or, after an [Express](#) installation, perform the following reconfigurations, where necessary:

- [Moving the Web Applications Server to Another Machine](#)
- [Changing the Database Provider for the FME Flow Database](#)
- [Moving the FME Flow System Share to Another Machine](#)

Planning for Fault Tolerance


The goal of a fault-tolerant environment is to ensure that if a hardware component fails, FME Flow remains online.

The fault tolerant architecture is comprised of multiple, redundant web application servers (to run the [FME Flow Web Services](#)) and [FME Flow Cores](#) spread across separate host machines. [FME Engines](#) can be hosted on the same machine, across [additional hosts](#) on the same network, or [across networks](#).


A third-party load balancer is required, which directs incoming traffic to one of the redundant web components.


The web application server provided by the installer is Apache Tomcat. Optionally, [you can provide your own web application server](#) instead of using the one shipped with the installer. We recommend installing the web application server on the same system as the Core, but this component can also be distributed.

The [FME Flow Database](#) can be hosted on a PostgreSQL database server provided by the installer, or you can [configure the database on your own server](#) using PostgreSQL (recommended), Microsoft SQL Server or Oracle. We recommend providing your own database server configured for redundancy. If using an Oracle database server, you must [obtain the Oracle Database JDBC Driver](#).

 **Note** PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.

Organizations are expected to maintain the FME Flow Database and FME Flow System Share (a file system for hosting [Repositories](#) and [Resources](#)) on their own fault-tolerant servers. This ensures the fault-tolerant FME Flow has reliable access to workspaces, repositories, resources, and other items.

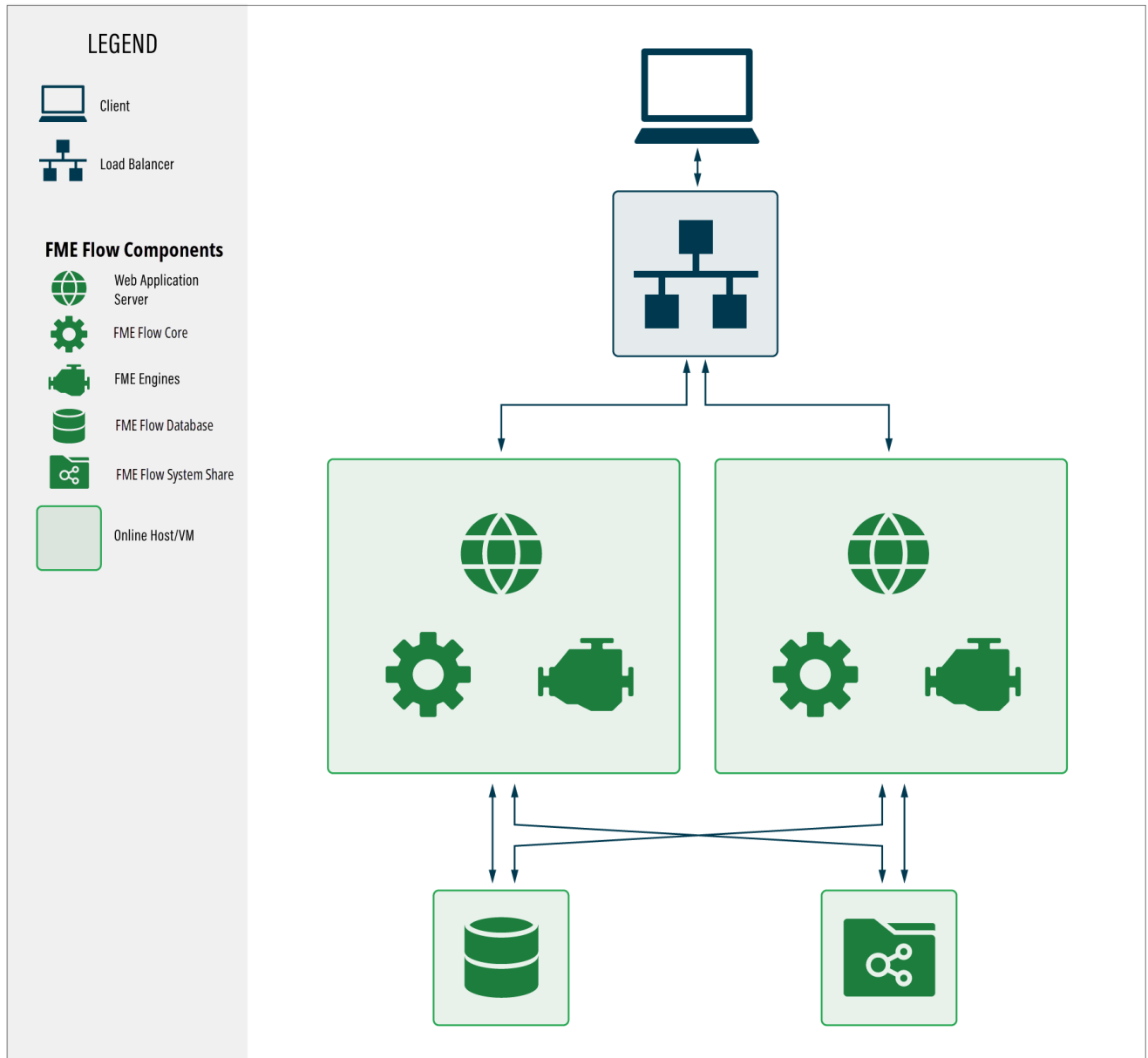
-  **Warning** We recommend installing all FME Flows on systems that are synchronized to the same time zone. If time zones differ across FME Flows, unexpected issues may arise, including:
- Improper timing of [Schedule Initiated](#) triggers.
 - Inconsistent or misleading timestamps in log files (accessed from [Resources](#)).

 **Note** In a fault tolerant installation of FME Flow, the Automations triggers [UDP Message Received](#) and [Email Received \(SMTP\)](#) (and corresponding Notification Service [UDP Publisher](#) and [SMTP Publisher](#)) are not supported. To receive email messages, consider the [Email Received \(IMAP\)](#) trigger instead.

Example Architectures

Example 1: FME Engines hosted on the same machine as the web application server and FME Flow Core

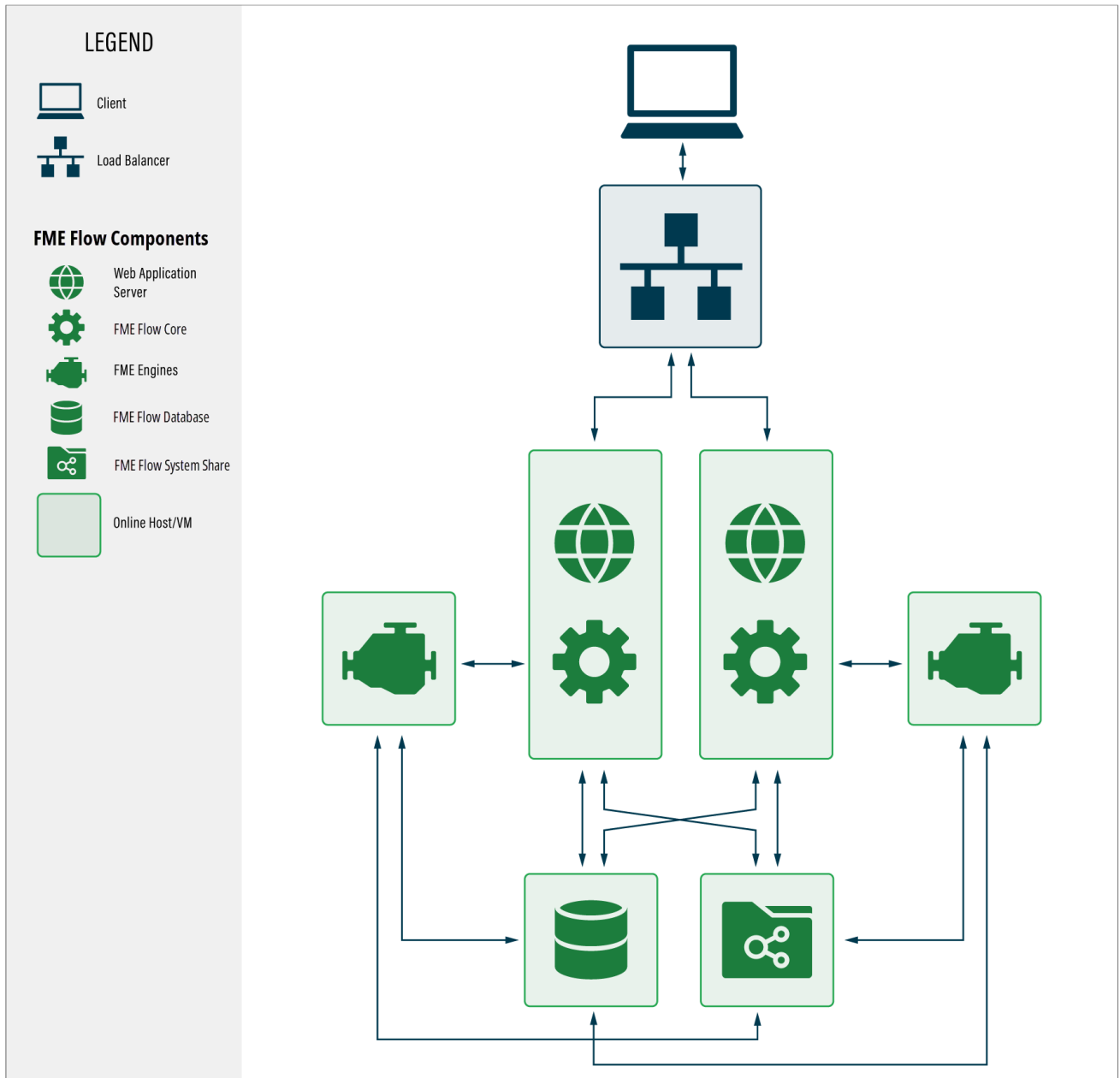
Fault Tolerant Install Example 1



Example 2: FME Engines distributed and hosted on a separate machine from the web application server and FME Flow Core

There can be any number of distributed FME Engine hosts associated with the FME Flow Core.

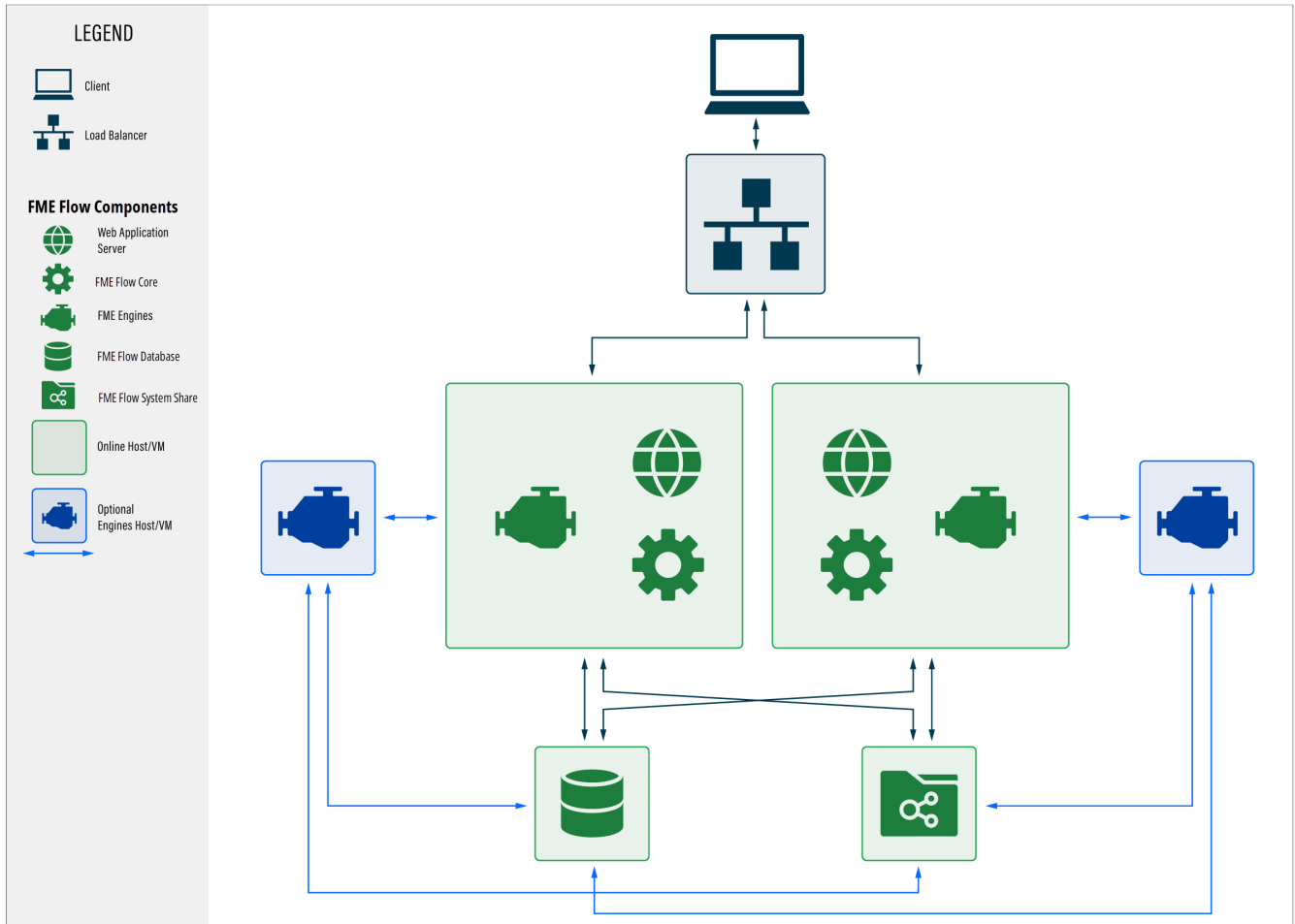
Fault Tolerant Install Example 2



Example 3: A combination of examples 1 and 2

Some FME Engines are hosted on the same machine as the web application server and FME Flow Core, and others are distributed.

Fault Tolerant Install Example 3



To Install a Fault Tolerant System

- Proceed to [Installing a Scalable, Fault-Tolerant FME Flow on page 57](#).

Planning for Disaster Recovery

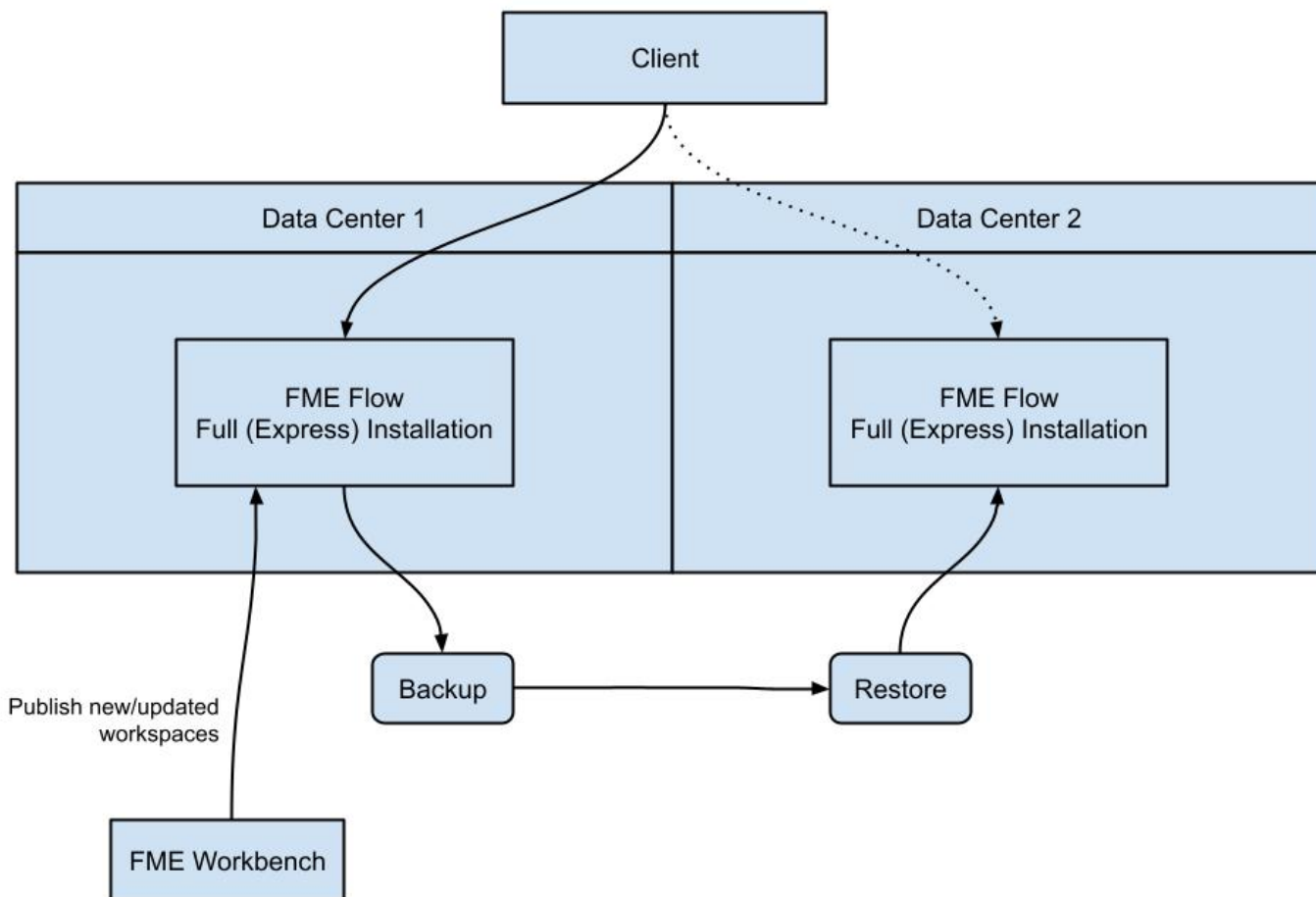
Disaster recovery is primarily concerned with recovering FME Flow operations and data in the event of a major failure of a data center. The general concept of disaster recovery is that if one data center fails, the second data center takes over, and the [FME Flow Core](#) located there becomes the 'active' Core. 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 in seconds to minutes.

Disaster recovery can be incorporated into a fault-tolerant architecture. Alternatively, if you are primarily concerned with disaster recovery, and less concerned about the fast recovery provided by fault tolerance, you may want to implement a different [architecture](#).

- ⚠ **Warning** We recommend installing all FME Flows on systems that are synchronized to the same time zone. If time zones differ across FME Flows, unexpected issues may arise, including:
- Improper timing of [Schedule Initiated](#) triggers.
 - Inconsistent or misleading timestamps in log files (accessed from [Resources](#)).

This example of disaster recovery is a combination of an Express installation and a fault-tolerant installation. However, instead of a third-party load balancer between systems, FME Flow clients must be manually redirected to the FME Flow Core host server of the second data center in the event of a disaster. Each data center houses full ("Express") installations of FME Flow, essentially configured to provide similar functionality. To ensure synchronicity of FME Flow system data between data centers, [Backup & Restore](#) operations are performed [regularly](#). (Otherwise, workspaces and other updates must be published twice - to the FME Flow Core hosts on each data center.)



Keep in mind that when planning for disaster recovery, all clients of FME Flow, including web browsers, the [FME Flow CLI](#), and the [FME Flow REST API](#), must connect to the "active" FME Flow Core host.

Planning for Security Updates

All installations of FME Flow, regardless of type, include the [FME Flow Core](#) and [FME Engines](#). These components are always provided directly from the FME Flow installation package. Two additional components - a [Web Application Server](#) and a server for the [FME Flow Database](#) - must also be installed. When you choose an [Express](#) installation of FME Flow, the install package provides its own versions of these components, including an Apache Tomcat web application servlet, and a PostgreSQL

database server. If you choose a [Distributed](#) installation of FME Flow, you may need to provide your own database server and web application server, depending on the scenario. One factor in deciding between a stand-alone or distributed installation of FME Flow 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 Flow (Express), keep in mind that any security updates to these components are dependent on updates to FME Flow releases in general. Each time FME Flow releases an update to its software (including both major and minor releases), any security updates for these components are included in that release. If you do not want to rely on updates to the FME Flow software in general for security updates to the web application and database servers, then we recommend a Distributed/Fault Tolerant installation. You can provide these components on your own, and maintain security for them separately. In the case of the web application server, Apache Tomcat version 9.0.x is supported. The FME Flow Database supports PostgreSQL (recommended), Oracle, and SQL Server. Alternatively, if your FME Flow 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.

- To proceed with a full installation of FME Flow, including web application and database servers provided with the install package, see [Install FME Flow: Express Installation for Windows on page 41](#) or [Install FME Flow: Express Installation for Linux on page 50](#).
- To read about installation scenarios that support providing your own web application and database servers, see [Distributed on page 17](#).

Reporting Potential Vulnerabilities

FME Flow is comprised of many Java and 3rd-party libraries. It is possible for virus scanners to identify security vulnerabilities at any time during the life of an FME Flow installation. When this occurs, you can search the [FME Community](#) for more information, or [contact Safe Software Support](#) to report the vulnerability. Safe Software takes all security vulnerability concerns seriously. Each concern is handled individually and we will report on the appropriate action once investigated.

Planning for Scalability and Performance


Scale your FME Flow to increase job throughput and optimize job performance.

Increasing Job Throughput

To increase the ability of FME Flow to run jobs simultaneously, consider any of these approaches:

Multiple Engines on the Same Machine

You can scale FME Flow to support a higher volume of jobs by [adding FME Engines on the same machine](#) as the FME Flow Core. A single active Core is all you need to scale processing capacity. The FME Flow Core contains a Software Load Balancer that distributes jobs to the FME Engines. Each FME Engine can process one job at any one time, so if you have ten engines, you can run ten jobs simultaneously. If you have many simultaneous job requests, with jobs consistently in the queue, consider adding engines to your Core machine.

 **Note** Adding engines to the same machine does not reduce the time a single translation takes to run. This time is dependent on the underlying hardware and the design of the workspace. Complex workspaces, big data manipulation, and large datasets take more time to run.

Having multiple engines on the same machine also helps with [Job Recovery](#).

Adding Engines on Separate Machines

If existing FME Engines are utilizing all system resources to process jobs, you can [add FME Engines on a separate machine](#) or access [Remote Engine Services](#). Either approach allows you to use the system resources of multiple machines, which allows additional concurrent jobs to be run.

Multiple FME Flow Installations

A [fault tolerant](#) architecture provides for multiple, stand-alone FME Flow installations. In addition to providing fault tolerance, this configuration distributes jobs between FME Flows via a third-party load balancer.

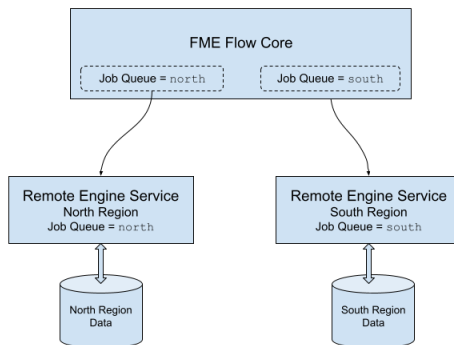
Increasing Job Performance

Using the following approaches, you can provide flexibility for running jobs in close physical proximity to the data they read and write:

- [Adding FME Engines on a Separate Machine](#): This approach requires the engine machines to be in the same network, and in the same data center or geographically close.
- [Remote Engine Services](#): This approach works well when you want to access FME Engines on servers outside of your network on accessible endpoints or in a Cloud service, while maintaining your primary FME Flow installation behind a firewall. It can also be deployed within a network.

To ensure each job is run by the intended engine, you must use either approach in combination with [Queue Control](#).

In this remote engine services example, consider two data sources - one located in a northern region, and another located in a southern region. To run jobs efficiently, it makes sense to access remote engines in both regions. Jobs that are run on queue `north` access data in the northern data store. These jobs are routed to remote engines located in the northern region. Likewise, jobs that are run on queue `south` access data in the southern data store. These jobs are routed to remote engines located in the southern region.



Controlling the Queue

To exercise a finer level of control over how jobs are processed, consider the following approaches:

Queue Control

[Queue Control](#) manages or spreads the work load of engines running workspaces. In a distributed environment, you may wish to run small jobs on certain engines, and larger jobs on other engines.

Or, you may have a mix of OS platforms on which certain FME formats can and cannot be run. For instance, consider an FME Flow on a Linux OS. Linux cannot run some formats that may be required by your business. So, it may be necessary to have a Windows OS configured with an additional FME Flow Engine.

Queues are also used when [Adding FME Engines on a Separate Machine](#) or with [Remote Engine Services](#), to route jobs to engines that are located in close physical proximity to the data they read and write.

You can set engines to process certain jobs based on the queue of the transformation request.

Queue Priority

FME Flow allows you to set job priority using the *Priority* directive of a [queue](#). Jobs in higher-priority queues may execute before jobs in lower-priority queues.

About Component and Job Recovery in FME Flow

FME Flow restarts components and jobs automatically when crashes occur—no additional planning is needed.


Component Recovery

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

Job Recovery

FME Flow also includes the ability to restart a translation (job) when a crash occurs.

FME Flow continues to resubmit a translation up to a specified number of attempts. As a result, jobs that experience temporary issues, such as a network hiccup, are resubmitted and run again. Job recovery is configurable and can be turned off entirely. For more information, see [Job Recovery](#).

 **Note** Resubmitted jobs may cause data duplication, such as when writing to database formats.

System Requirements

For system requirements for the Windows or Linux operating systems, see

<https://www.safe.com/fme/tech-specs/>.

For information about supported web browsers, see [Accessing the Web User Interface](#).

Linux: Offline Installations

FME on Linux requires many third party packages. These dependencies must be satisfied when FME is installed. The easiest way to satisfy these dependencies is to install FME on a computer that is connected to the internet, allowing packages to be downloaded as needed.

It is possible, however, to manually prepare an offline computer for FME installation.

1. Get dependencies for the FME Engines component of the installation.
 - a. Extract all files from the FME Flow installer with flags `--noexec` and `--keep`. For example:

```
./fme-flow-2023.0-b23277-linux-x64~ubuntu.22.04.run --noexec -  
-keep
```


These flags create a folder `fme-flow` in the current directory that contains the contents of the FME Flow installer, including the FME Engine `.deb` or `.rpm` package.

b. Get the package dependencies:

- Ubuntu/Debian: `dpkg -I <package_name>.deb`
- Red Hat/Rocky Linux: `yum deplist <package_name>.rpm`


2. Get dependencies for the FME Flow component of the installation

- Packages required: `zip`, `unzip`, `perl`, `tar`, `xhost`, `policycoreutils-python-utils`
- If your [deployment architecture](#) includes the PostgreSQL database server provided with the installation (Express and certain distributed architectures), the following additional packages are required:
 - Debian/Ubuntu: `postgresql`
 - Red Hat/Rocky Linux: `postgresql-server`

 **Note** RedHat and Rocky Linux require that NumPy and Matplotlib packages are installed via pip and pip3. This may be done before or after installing FME. For more information, see [Using Python with FME Flow](#).

For more information:

- Ubuntu/Debian: [Installing packages without an Internet connection](#)
- Red Hat/Rocky Linux: <https://access.redhat.com/solutions/29269>

 **Warning** If you are accessing local mirrors of public repositories, ensure they match the version of your Linux.

Linux: Online Installations

Internet-restricted Installations through a Proxy

If installing on a machine with limited internet access in which a proxy is configured to handle communication required by the installer, the proxy must be configured to allow connections to the system package repository for that operating system.

Check for Active Subscription (Red Hat Only)

FME cannot be installed on a Red Hat instance unless it has an active subscription. For more information:

- https://access.redhat.com/documentation/en-us/red_hat_subscription_management/1/html/quick_registration_for_rhel/

Make Sure the System is Up-to-Date

On Red Hat/Rocky Linux, run:

```
yum update
```

On Ubuntu/Debian, run:

```
apt update && apt upgrade
```

Install zip and unzip Utilities

On Red Hat/Rocky Linux, run:

```
yum install zip  
yum install unzip
```

On Ubuntu/Debian, run:

```
apt-get install zip  
apt-get install unzip
```

Red Hat/Rocky Linux: Configure Access to EPEL Repository

On Red Hat or Rocky Linux, you must first download and install the Extra Package for Enterprise Linux (EPEL) Community Project repository.

For more information:

- https://fedoraproject.org/wiki/EPEL#How_can_I_use_these_extra_packages.3F

See also:

- <https://docs.rackspace.com/support/how-to/install-epel-and-additional-repositories-on-centos-and-red-hat/>

Requirements for FME Workspaces

To perform data transformations, FME Flow runs workspaces. When a workspace runs on FME Flow, it is referred to as a job.

In this Section

- [FME Workbench](#)
- [GridShift Files](#)
- [Third-Party Applications for FME](#)

FME Workbench

FME Workbench is used to author and publish the workspaces used by FME Flow. FME Workbench is not part of FME Flow; rather, it is part of the FME Form product. Contact your Safe Software account manager or your local Safe Software reseller, if you would like more information about FME Form.

If you don't have access to FME Form, you cannot publish workspaces to FME Flow, although you can still perform and test the installation.

Connecting to FME Flow

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

Version of FME Workbench

The version of FME Workbench you use must match the FME Flow version you have installed. This requirement ensures that the workspaces you author will run when published to FME Flow.

If you are running an FME Flow version that is older than your FME Workbench, it is possible to upgrade the FME Engines used by FME Flow to match your FME Workbench version. Instructions for upgrading the FME Engine only are available in [FME Engine Only Upgrade](#)

GridShift Files

If you plan to run workspaces that transform data between the datums NAD27 and NAD83, you can use datum shift files for Canada or the US. For more information, see [Configuring Implicit NAD27 to NAD83 Datum Shifts](#).

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 ArcGIS installation for connecting to Esri Geodatabases.

If you must install a third-party application for your FME Form installation, repeat that installation on all machines hosting [FME Engines](#). Review our [FME formats matrix](#) and compare by platforms to check if the formats you plan to use are available.

Performing an Installation

See Also

- [Planning an FME Flow Installation](#)



Install FME Flow: Express Installation for Windows

This chapter provides information for an Express installation of FME Flow for Windows. Before getting started, take a few minutes to review [Planning an FME Flow Installation](#).

To Perform an Express Installation for Windows

1. [Obtain the Installer](#)
2. [Run the Installer](#)
3. [Starting and Stopping FME Flow](#)
4. [Log on to the Web User Interface](#)
5. [Request and Install a License](#)
6. [Test the Installation](#)

See Also

- [Uninstalling](#)

Obtain the Installer

To obtain the applicable FME Flow installation package, visit [the FME Downloads page](#). This page includes download links to release and beta installation packages for various platforms.

You can also contact sales@safe.com to get access.

What's Next?

- [Run the Installer](#)

Run the Installer


Run the installer under an administrator account. This account must have:

- Write privileges to the specified installation directory.
- "Log on as a service" rights on this machine. For more information, see [this Microsoft Docs article](#).

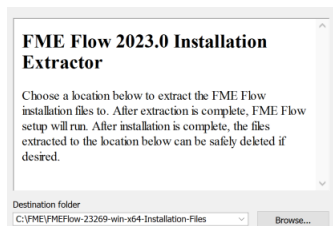
FME Flow Installation Extractor

Start the FME Flow installer.

On the Installation Extractor dialog, specify a *Destination folder* in which to extract installation files. Note that this folder *only* specifies where installation files are extracted—it does *not* specify where FME Flow program files are installed. That location is specified later in the installer.

 **Tip** Take note of the specified *Destination folder* location. Following installation, you may wish to:

- Review the installation .log file.
- Remove these files to save disk space.

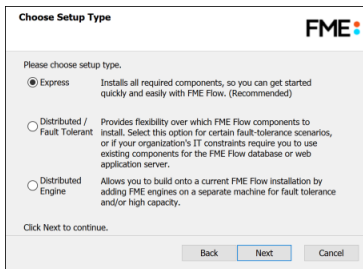


Click **Install**.

When extraction is complete, the installer opens. Proceed through the installer to the Choose Setup Type dialog.


Choose Setup Type

Select the **Express** option.




Destination Folder


- **Install FME Flow to:** By default, most files for your FME Flow installation are written to C:\Program Files\FMEFlow\. To change the installation directory, modify the path.
- **Install the FME Flow System Share to:** By default, FME Flow System Share files, which include [Repositories](#) and [Resources](#), are written to %ALLUSERSPROFILE%\Safe Software\FME Flow\.

 **Note** Unless modified, C:\ProgramData is the default value of the %ALLUSERSPROFILE% environment variable.

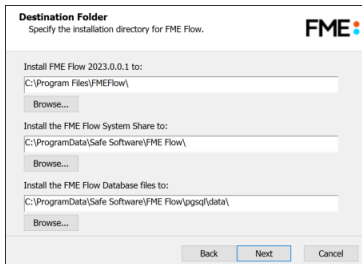
This configuration is recommended, because applications that use FME Flow to write data can access C:\ProgramData\Safe Software\FME Flow when all [FME Flow System Services](#) run under the local system account (the most common scenario), while the remaining files are protected in a read-only location. To change the installation directory of FME Flow System Share files, modify the path.

 **Note** If you plan to scale-out your FME Flow anytime following installation by [Adding FME Engines on a Separate Machine](#), the specified directory must be read/write accessible by the account that runs the additional FME Engines.

- **Install the FME Flow Database files to:** By default, [FME Flow Database](#) files are written to %ALLUSERSPROFILE%\Safe Software\FME Flow\pgsql\data.

 **Note** Unless modified, C:\ProgramData is the default value of the %ALLUSERSPROFILE% environment variable.

To change the installation directory of FME Flow Database files, modify the path.

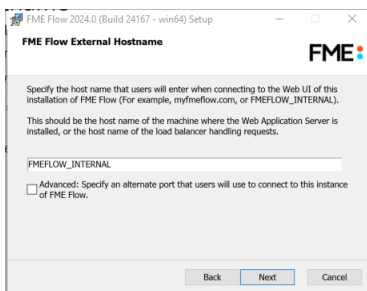


FME Flow External Hostname

If this FME Flow installation will be available to users outside an internal network (for example, on an external website), specify the external hostname that users will enter when connecting to FME Flow. For example, if you are hosting FME Flow on myfmefflow.com/myfmefflow, enter myfmefflow.com.

Otherwise, enter the internal server hostname, IP Address, or DNS.

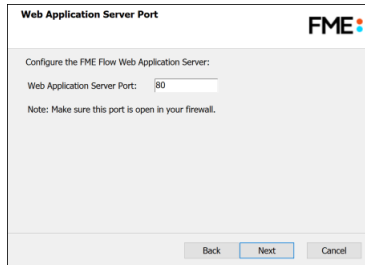
If a load balancer or reverse proxy is deployed through which to access FME Flow, check *Advanced* and provide the *External Web Traffic Port*. Specify this port only if it differs from the port on which the FME Flow Web Application Server listens, as specified in the next dialog.



Web Application Server Port

Port 80 is the recommended default. If this port is in use, port 8080 is recommended.

Note Common applications that may be using port 80 include Skype and Internet Information Services (IIS). Either turn off these services, or select a different port, such as 8080.

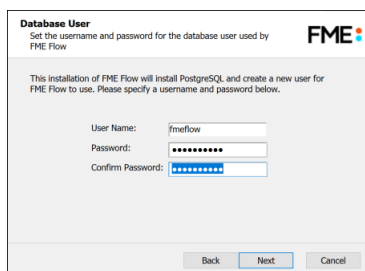


Database User

The installation creates the [FME Flow Database](#) on a PostgreSQL database server, and a user account under which to use the database. Specify a *User Name* (defaults to **fmeFlow**) and *Password* for the database user account.

Warning Do not specify *User Name* **postgres**. This name corresponds to the PostgreSQL master user. If the corresponding passwords do not match, the database will not install.

Note The password for the FME Flow Database user account is encrypted in FME Flow installation files.




On the last screen, click **Install**.

Proceed to [Starting and Stopping FME Flow below](#).

Starting and Stopping FME Flow

After performing the installation, the FME Flow service and associated applications start automatically. You might have to start and stop these applications at some point. See [Starting and Stopping FME Flow Manually](#) for instructions.

 **Note** If UAC (User Account Control) is enabled on your Windows system and the installation was not run from an elevated prompt (that is, Run as administrator), you might need to start the FME Flow Windows Services.

What's Next?

- [Log on to the Web User Interface](#)

Log on to the Web User Interface

To log in to the Web User Interface for the first time as an administrator following installation, specify *Username* `admin` and *Password* `admin`. You will be prompted to update the `admin` password upon initial use, based on the default [Password Policy](#) configuration.

For more information and support:

- [FME Flow Troubleshooting: Web Interface Login](#)
- [Accessing the Web User Interface](#)
- [Changing the Login Password](#)
- [Default User Accounts and Passwords](#)
- [Role-Based and User-Based Access Control](#)

What's Next?

- [Request and Install a License](#)

Request and Install a License

Note

License files are installed, by default, to:

» Windows: `C:\ProgramData\Safe Software\FME Flow\licenses`

» Linux: `/opt/fmeflow/licenses`

Dynamic Engine (CPU Usage) licensing is not available for engines on FME Flow Hosted.

Automatic Mode (Recommended)

If your FME Flow host machine has an internet connection, you can request and install a license online without any need for downloading and re-uploading a license file.

Note


- Licensing FME Flow over the internet requires no inbound connection. For more information, see [FME Flow Automatic Licensing URLs](#).
- If you purchased credits for Dynamic Engines (CPU Usage), automatic mode only is supported. Manual mode licensing (below) is not supported.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.
3. Complete the fields on the Request a New License page.

Note If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.


4. *Licensing Mode*: Select **Automatic**.

5. Click **OK**.
6. A message indicates that your license is installed, and your available engines appear momentarily on the [Engines](#) page.


 **Note** If the *Serial Number* you provided includes Dynamic Engines (CPU Usage), they are not configured to start by default. You can start and configure your Dynamic Engines on the [Engines](#) page.

Manual Mode

If you prefer not to use an internet connection for licensing, or if your FME Flow host machine does not have an internet connection, use this process to install a license.

 **Note** If you purchased credits for Dynamic Engines (CPU Usage), manual mode licensing is not supported. Use automatic mode for licensing (above). For more information, contact your Safe Software Inc. sales representative.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Flow engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.
3. Complete the fields on the Request a New License page.

 **Note** If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.

4. *Licensing Mode*: Select **Manual**.
5. Click **OK**. A license request .json file downloads.
6. Email the .json file to codes@safe.com.

7. You will receive an email from Safe Software Inc. with a license file, which has a `.fmeLIC` extension. Download the file.
8. Return to the Licensing page. Under Standard Engines, click **Upload License File** and select the `.fmeLIC` file to upload. Or, drag and drop the `.fmeLIC` file onto the page under Standard Engines. A message indicates that it is installed, and your available engines appear on the [Engines](#) page.

Problems?

- [FME Flow Troubleshooting: FME Flow Engines](#)
- Contact codes@safe.com.

See Also

- [FME Flow Automatic Licensing URLs](#)
- [FME Engines](#)

What's Next?

- [Test the Installation](#)

Test the Installation

To test your FME Flow installation, confirm that FME Flow can perform its primary function—running a job.

1. On the [Run Workspace](#) page, specify:
 - Repository: Samples
 - Workspace: austinApartments.fmw
 - Service: Job Submitter
2. Click Run Workspace.

If you see a success message, your install was a success.

See Also

- [FME Flow Troubleshooting: Submitting a Job in FME Flow](#)

Uninstalling

Things to Consider

Before uninstalling FME Flow, keep in mind the following:

- Before you upgrade to a newer version of FME Flow, see [Upgrading FME Flow on page 296](#) for instructions on backing up your existing configuration. You must uninstall any previous versions of FME Flow before installing an upgrade.
- Uninstalling FME Flow removes the contents of your Tomcat directory if you installed the default Web Application Server. This folder is located in the FME Flow installation directory under `Utilities\tomcat`. If you made custom modifications to this Tomcat, these modifications should be backed up before uninstall.

To Uninstall:

Ensure that FME Flow, FME Flow Application Server, or other Web Application servers are [shut down](#) before proceeding.

You can uninstall FME Flow as you would any other program by selecting Start menu > Control Panel > Programs and Features > FME Flow.

Install FME Flow: Express Installation for Linux

This chapter provides information for an Express installation FME Flow for Linux. Before getting started, take a few minutes to review [Planning an FME Flow Installation](#).

To Perform an Express Installation for Linux

1. Ensure [System Requirements](#) are met, including updates and dependencies.
2. [Obtain the Installer](#)

3. [Run the Installer](#)
4. [Starting and Stopping FME Flow](#)
5. [Log on to the Web User Interface](#)
6. [Request and Install a License](#)
7. [Test the Installation](#)

See Also

- [Uninstalling](#)

Obtain the Installer

To obtain the applicable FME Flow installation package, visit [the FME Downloads page](#). This page includes download links to release and beta installation packages for various platforms.

You can also contact sales@safe.com to get access.

What's Next?

- [Run the Installer](#)

Run the Installer

The installer should be executed as the root user. The recommended method is to call the install file directly. These examples assume you are in the same directory as the installer:

- As root user:

```
chmod +x fme-flow-b23278-linux-x64~ubuntu.22.04.run
./fme-flow-b23278-linux-x64~ubuntu.22.04.run
```

- Not root user, but with sudo privileges:

```
sudo chmod +x fme-flow-b23278-linux-x64~ubuntu.22.04.run
sudo ./fme-flow-b23278-linux-x64~ubuntu.22.04.run
```

Proceed through to the Choose Setup Type prompt. Select the Express option.

```
Do you accept the terms of the FME User License? Yes/No [No] : y


Express: Installs all required components, so you can get started quickly and easily with FME Server.

Custom / Distributed: Provides flexibility over which FME Server components to install. Select this option for certain fault-tolerance scenarios, or if your organization's IT constraints require you to use existing components for the FME Server database, web application, or license server.

Engine: Allows you to build onto a current FME Server installation by adding FME engines on a separate machine for fault tolerance and/or high capacity.

Please choose the setup type.
Valid Choices:
1) Express
2) Custom / Distributed
3) Engine
[Express] : 1
```

Proceed through all the prompts. For each prompt, the default answer appears in square brackets, such as `[default]`. If you press Enter without typing an answer, you are accepting the default answer.

 **Warning** When prompted for the database user name and password, do not specify user name **postgres**. This name corresponds to the PostgreSQL master user. If the corresponding passwords do not match, the database will not install.

Note

- The password for the FME Flow Database user account is encrypted in FME Flow installation files.
- Regardless of the installation directory you specify, [FME Engines](#) are installed directly under `/opt/fme-engine`.
- FME Flow installs an NGINX reverse proxy, which allows you to select port 80 or any port number less than 1024, and run the Apache Tomcat [Web Application Server](#) as the FME Flow user instead of root.

Proceed to [Starting and Stopping FME Flow on the next page](#).

Starting and Stopping FME Flow

After performing the installation, start the FME Flow System Services. For instructions on starting and stopping the services manually, see [Starting and Stopping FME Flow Manually on page 398](#). For instructions on configuring the services to start at boot time, see [Configuring FME Flow as a System Service \(Linux\) on page 404](#).

What's Next?

- [Log on to the Web User Interface](#)

Log on to the Web User Interface

To log in to the Web User Interface for the first time as an administrator following installation, specify *Username* `admin` and *Password* `admin`. You will be prompted to update the `admin` password upon initial use, based on the default [Password Policy](#) configuration.

For more information and support:

- [FME Flow Troubleshooting: Web Interface Login](#)
- [Accessing the Web User Interface](#)
- [Changing the Login Password](#)
- [Default User Accounts and Passwords](#)
- [Role-Based and User-Based Access Control](#)

What's Next?

- [Request and Install a License](#)

Request and Install a License

Note

License files are installed, by default, to:

» Windows: `C:\ProgramData\Safe Software\FME Flow\licenses`

» Linux: `/opt/fmeflow/licenses`

Dynamic Engine (CPU Usage) licensing is not available for engines on FME Flow Hosted.

Automatic Mode (Recommended)

If your FME Flow host machine has an internet connection, you can request and install a license online without any need for downloading and re-uploading a license file.

Note


- Licensing FME Flow over the internet requires no inbound connection. For more information, see [FME Flow Automatic Licensing URLs](#).
- If you purchased credits for Dynamic Engines (CPU Usage), automatic mode only is supported. Manual mode licensing (below) is not supported.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.
3. Complete the fields on the Request a New License page.

Note If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.


4. *Licensing Mode*: Select **Automatic**.

5. Click **OK**.
6. A message indicates that your license is installed, and your available engines appear momentarily on the [Engines](#) page.


 **Note** If the *Serial Number* you provided includes Dynamic Engines (CPU Usage), they are not configured to start by default. You can start and configure your Dynamic Engines on the [Engines](#) page.

Manual Mode

If you prefer not to use an internet connection for licensing, or if your FME Flow host machine does not have an internet connection, use this process to install a license.

 **Note** If you purchased credits for Dynamic Engines (CPU Usage), manual mode licensing is not supported. Use automatic mode for licensing (above). For more information, contact your Safe Software Inc. sales representative.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Flow engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.
3. Complete the fields on the Request a New License page.

 **Note** If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.

4. *Licensing Mode*: Select **Manual**.
5. Click **OK**. A license request .json file downloads.
6. Email the .json file to codes@safe.com.

7. You will receive an email from Safe Software Inc. with a license file, which has a `.fmeLIC` extension. Download the file.
8. Return to the Licensing page. Under Standard Engines, click **Upload License File** and select the `.fmeLIC` file to upload. Or, drag and drop the `.fmeLIC` file onto the page under Standard Engines. A message indicates that it is installed, and your available engines appear on the [Engines](#) page.

Problems?

- [FME Flow Troubleshooting: FME Flow Engines](#)
- Contact codes@safe.com.

See Also

- [FME Flow Automatic Licensing URLs](#)
- [FME Engines](#)

What's Next?

- [Test the Installation](#)

Test the Installation

To test your FME Flow installation, confirm that FME Flow can perform its primary function—running a job.

1. On the [Run Workspace](#) page, specify:
 - Repository: Samples
 - Workspace: austinApartments.fmw
 - Service: Job Submitter
2. Click Run Workspace.

If you see a success message, your install was a success.

See Also

- [FME Flow Troubleshooting: Submitting a Job in FME Flow](#)

Uninstalling

Things to Consider

Before uninstalling FME Flow, keep in mind the following:

- Before you upgrade to a newer version of FME Flow, see [Upgrading FME Flow on page 296](#) for instructions on backing up your existing configuration. You must uninstall any previous versions of FME Flow before installing an upgrade.
- Uninstalling FME Flow removes the contents of your Tomcat directory if you installed the default Web Application Server. This folder is located in the FME Flow installation directory under `Utilities\tomcat`. If you made custom modifications to this Tomcat, these modifications should be backed up before uninstall.
- Uninstalling removes the configuration of the following components, but does not uninstall them. If reinstalling or performing an upgrade, these components are reused and reconfigured:
 - The PostgreSQL [FME Flow Database](#), if applicable.
 - The NGINX reverse proxy.

To Uninstall:

Ensure that FME Flow, FME Flow Application Server, or other Web Application servers are [shut down](#) before proceeding.

Run the `uninstall.sh` script in the FME Flow installation directory. To ensure that everything gets uninstalled, run this script as root.


Installing a Scalable, Fault-Tolerant FME Flow

Before installing FME Flow for fault tolerance, make sure you have:


- Read and understood [Planning an FME Flow Installation](#), including [Planning for Fault Tolerance](#).
- Provided the prerequisites in [Before Installation](#).
- Linux: Ensured [System Requirements](#) are met on all machines you have allocated for your fault-tolerant architecture, including updates and dependencies.

To perform an installation for fault tolerance:

1. [Obtain the Installer](#)
2. [Install FME Flow on All Servers](#) you have allocated for your fault-tolerant architecture.
3. [Configure the FME Flow Database](#) on your database server.

 **Note** Optionally, you can [install the PostgreSQL database that is included with the FME Flow installer](#) , instead of configuring the FME Flow Database on your own database server. Be aware the default database is not configured for redundancy.

4. [Set Up the Load Balancer and Configure with FME Flow](#) to route traffic between each FME Flow instance you installed.
5. [Start the FME Flow system services](#). Following a fault-tolerant installation, the [FME Flow system services](#) do not start automatically. You must manually start them.
6. [Log on to the Web User Interface](#) and update the password for the admin user account (recommended).
7. [Request and Install a License](#) on all FME Flow installations.

 **Note** Specify the same serial number in all licensing requests.

8. [Test the Installation](#).

What's Next?

Optimize: You can scale-up your system by:

- Increasing the number of FME Engines that run on machines that are part of your FME Flow deployment.
- Adding or accessing FME Engines on additional machines.

For more information, see [Planning for Scalability and Performance](#).

Before Installation

Before you install FME Flow for fault tolerance, ensure the following components are in place on your network:

- Separate machines on which to install the number of FME Flows that you have allocated for your fault tolerant architecture.
- A load balancer to route traffic between each FME Flow instance you install.
- Fault-tolerant [database server](#) on a separate machine.
- Fault-tolerant [file system](#) on a separate machine.


Next Steps

Proceed to [Obtain the Installer](#).

Provide a Database Server

FME Flow uses its own database to manage jobs and workspace information. It is not a source or destination data source for FME workspaces.


When configuring a fault tolerant environment for FME Flow, you must provide your own database server on which to configure the FME Flow database. Supported database servers include PostgreSQL (recommended), SQL Server, and Oracle.

 **Note** PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.

Provide a database server that is physically separate from the machines on which the FME Flows are installed. Separating the FME Flow Database ensures that it also remains available in a failover scenario.

When providing a database server, keep in mind the following:

- In addition to being physically separate, the database server should, by itself, be configured for fault tolerance.
- When you install the FME Flows, the database server must be running and you must know the connection information.
- If using an Oracle database server, you must obtain the Oracle Database JDBC driver. For more information, see <https://www.oracle.com/technetwork/database/application-development/jdbc/downloads/index.html>.

 **Note** Oracle Database JDBC driver versions that use ojdbc10.jar are *not* supported.

- Following installation, you must be able to create a new database on the server using SQL scripts.

For information about acquiring a fault tolerant database server, consult your IT department.

Provide a Remote File System for the FME Flow System Share

- **Skill Level:** Intermediate
- **Estimated Time Required:** 5-15 minutes
- **Prerequisites:**
 - Domain [Service Account](#) has correct permissions on network shares.

FME Flow System Share files, which include [Repositories](#) and [Resources](#), are installed in specific directories that are shared to enable access from different computers.

Create a directory for the FME Flow System Share that is physically separate from the machine on which the FME Flow Core and web application server are installed. Separating the file system ensures it remains available in a fault tolerant scenario.

When providing a remote file system, keep in mind the following:

- The accounts that run the FME Flow components must have read and write permissions on the remote file system directory. For more information, see [Directory and Account Permissions](#).
- In addition to being physically separate, the file system should, by itself, be configured for fault tolerance.
- When the directory is specified during installation, it must be in the form of a UNC path. Mapped drives are not supported.
- The directory that holds the System Share must be empty, unless System Share folders have already been created there as part of an installation involving multiple FME Flow Cores.


For information about acquiring a fault tolerant file system, consult your IT department.

Linux Only

For distributed installations on Linux, it is necessary to create a user, prior to installation, on the machine that will host the installation of the FME Flow Core and Engines. This user must be named 'fmeflow' and belong to group 'fmeflow', and be granted permissions on the FME Flow System Share. For more information, see [Directory and Account Permissions](#).

Example Instructions

The following example creates user 'fmeflow' in group 'fmeflow', and allows mount /mnt/data to be writable by the newly-created user.

 **Note** The following instructions are provided as a general guideline for adding a user and mounting a directory. Exact instructions may vary depending on the distribution and share type in your Linux configuration.

1. Run the following commands in the Linux command line:

```
groupadd fmeflow
useradd -m -g fmeflow fmeflow
```

2. Mount the FME Flow System Share directory so that user 'fmeflow' has read and write permissions. For example, to mount a Windows-based file share:

```
mount -v -t cifs //network/share /mnt/data -o gid=1008,uid=1008
```

`gid` and `uid` are the group id and user id of the user and group that were created.

These id's can be found using these commands:

```
id -g fmeflow
id -u fmeflow
```

Obtain the Installer

To obtain the applicable FME Flow installation package, visit [the FME Downloads page](#). This page includes download links to release and beta installation packages for various platforms.

You can also contact sales@safe.com to get access.

What's Next?

- [Install FME Flow on All Servers](#)


Install FME Flow on All Servers

Perform the following installation on all machines you have allocated for your fault tolerant architecture.

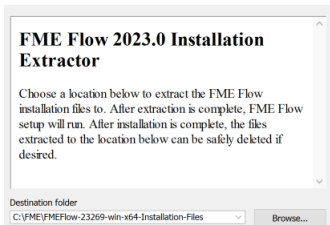
Start the FME Flow Installer.

(Windows only) Installation Extractor

On the Installation Extractor dialog, specify a *Destination folder* in which to extract installation files. Note that this folder *only* specifies where installation files are extracted—it does *not* specify where FME Flow program files are installed. That location is specified later in the installer.

 **Tip** Take note of the specified *Destination folder* location. Following installation, you may wish to:

- Review the installation .log file.
- Remove these files to save disk space.

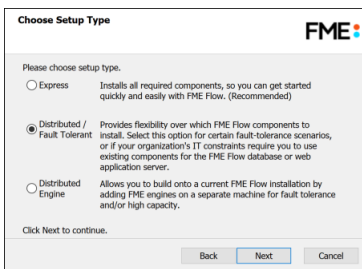


Click **Install**.

When extraction is complete, the installer opens. Proceed through the installer to the Choose Setup Type dialog.

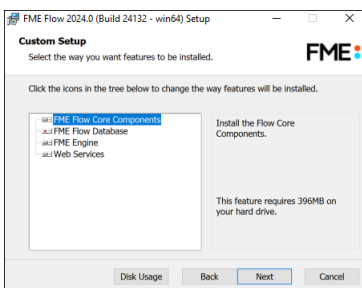
Choose Setup Type

Select the Distributed/Fault Tolerant install option.



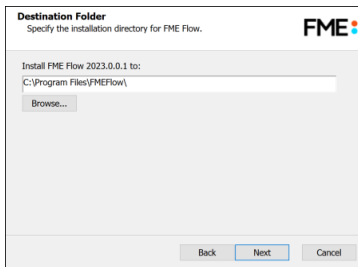
Custom Setup

Select FME Flow Core Components, FME Engine, and Web Services.



Destination Folder

Specify the installation directory for FME Flow.

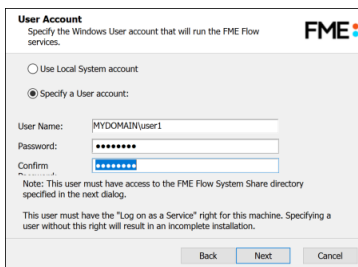


User Account

Specify the Windows user account that will run the FME Flow [System Services](#). This account must have:

- Read/Write access to the FME Flow System Share that you specify in the next dialog. The System Share is the location where FME Flow stores [Repositories](#) and [Resources](#) files. It must be accessible by all machines that comprise the FME Flow deployment. (For more information, see [Provide a Remote File System for the FME Flow System Share](#).)
- Read/Write access to the FME Flow installation directory that you specified in the previous dialog.
- "Log on as a service" rights on this machine. For more information, see [this Microsoft Docs article](#).

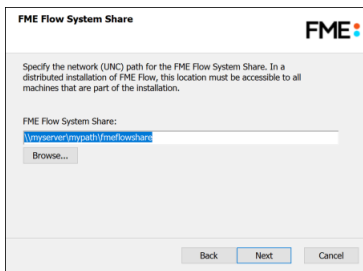
 **Note** The FME Flow Database service always runs under the local system account.



FME Flow System Share

Specify a shared directory visible to all machines that make up the FME Flow. This directory stores FME Flow System Share files, which include [Repositories](#) and [Resources](#). If this is a Linux installation, specify the mounted directory. For more information, see [Provide a Remote File System for the FME Flow System Share](#).

Note If you are [upgrading](#) to a newer version of FME Flow, ensure the specified directory is empty of files from the previous installation. Otherwise, conflicts may occur that can produce unexpected results in performance.

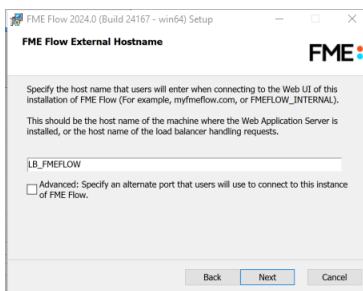


FME Flow External Hostname

Specify the name of the load balancing host through which FME Flow is accessed.

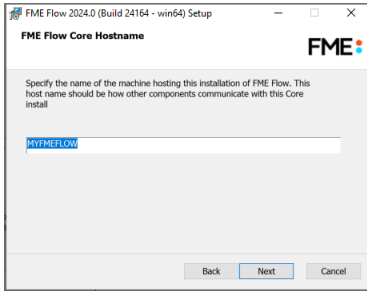
If the load balancer listens on a port different from the FME Flow [Web Application Server Port](#) (below), check *Advanced* and provide the *External Web Traffic Port*.

Note Do not specify "localhost".




FME Flow Core Hostname

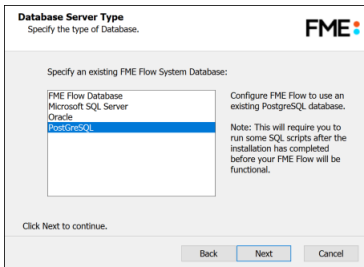
Specify the name of the machine hosting this installation of FME Flow.



Database Server Type

Specify the type of database for the FME Flow Database. For more information, see [Provide a Database Server](#).

 **Note** Do not specify the "FME Flow Database".

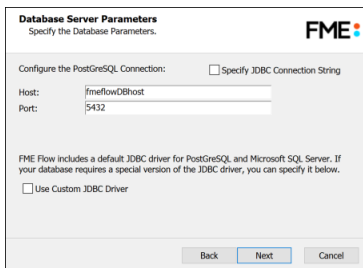


Database Server Parameters

Specify the connection parameters for the FME Flow Database. Optionally, to customize the connection parameters in the JDBC connection string, check **Specify JDBC Connection String**.

If **Microsoft SQL Server** was specified in the previous dialog, check *Use SQL Server instance name* to use that value instead of *Port*.

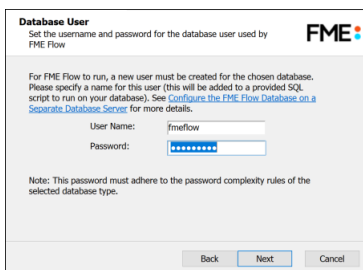
FME Flow includes a default JDBC driver for PostgreSQL and Microsoft SQL Server. If your database requires a special version of the JDBC driver, check *Use Custom JDBC Driver* and specify the *JDBC Driver*. If **Oracle** was specified in the previous dialog, you must specify the *JDBC Driver*.



Database User

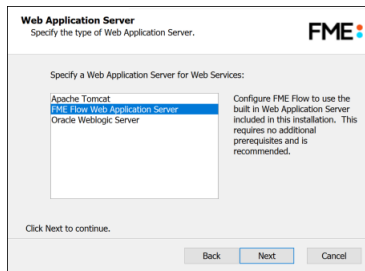
The installation creates a user account under which to use the FME Flow database. Specify a *User Name* (defaults to **fmeFlow**) and *Password* for the database user account.

Note The password for the FME Flow Database user account is encrypted in FME Flow installation files. Take note of the *User Name* and *Password* that you specified. You may need to reference it later.



Web Application Server

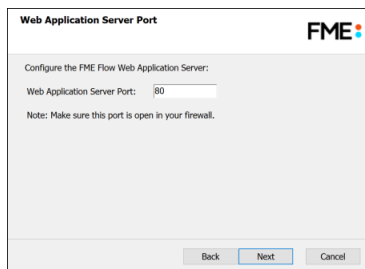
Specify the **FME Flow Web Application Server**.



Web Application Server Port

Specify the port to use for communication with the Web Application Server. On Windows, port 80 is the recommended default. If this port is in use, port 8080 is recommended. On Linux, port 8080 is recommended.

Note Common applications that may be using port 80 include Skype and Internet Information Services (IIS). Either turn off these services, or select a different port, such as 8080.




Follow the remaining dialogs to complete the installation.

What's Next?


- [Configure the FME Flow Database on a Separate Database Server](#)

Configure the FME Flow Database on a Separate Database Server

To configure FME Flow to use a separate database server, follow the steps below. You can configure FME Flow with a PostgreSQL (recommended), Microsoft SQL Server, or Oracle database, running on either Windows or Linux.

 **Note** PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.

1. Database Configuration: Set up FME Flow database tables and database users with permissions to access the FME Flow database.

 **Note** The specific way to set up database tables and user permissions on various production databases may differ depending on the available database tools and intended target platform.

Database Configuration - PostgreSQL

PostgreSQL is an open source database that can be downloaded from the Internet. For more information, see the PostgreSQL web site at:

<https://www.postgresql.com>

It's assumed that PostgreSQL is already installed .

One notation used is <FMEFlowDir>, which is the installation directory of FME Flow. In Windows, this is typically `C:\Program Files\FMEFlow`.

It's also assumed that you have JRE 5 or higher and PostgreSQL database installed before you proceed further. If you want to use the JDBC driver, there are several steps to perform.

If you are upgrading, you should back up any job history you want to keep.

Windows

This section describes how to configure a PostgreSQL database. You can configure this database from other PostgreSQL database tools, depending on your personal preference.

For a PostgreSQL database configuration, various SQL scripts are provided to assist you. These SQL scripts are located in the following directory:

`<FMEFlowDir>\Server\database\postgresql`

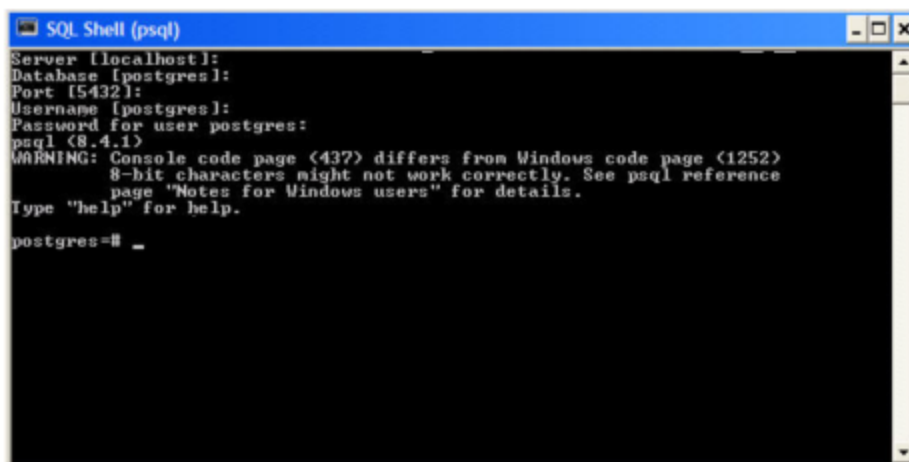
Note This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `postgresql_createUser.sql` – This script creates the FME Flow database user and password that you specified during installation, and grants all required permissions to this user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- `postgresql_createDB.sql` – This script creates the FME Flow database.
- `postgresql_createSchema.sql` – This script creates all FME Flow related tables, indexes, views, and triggers.
- `postgresql_dropUser.sql` – This script drops the FME Flow database user.
- `postgresql_dropDB.sql` – This script drops the FME Flow database.

Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install PostgreSQL SQL shell (`psql`) if you haven't done so already.
2. Open `psql` and log on as a user with the appropriate privileges:



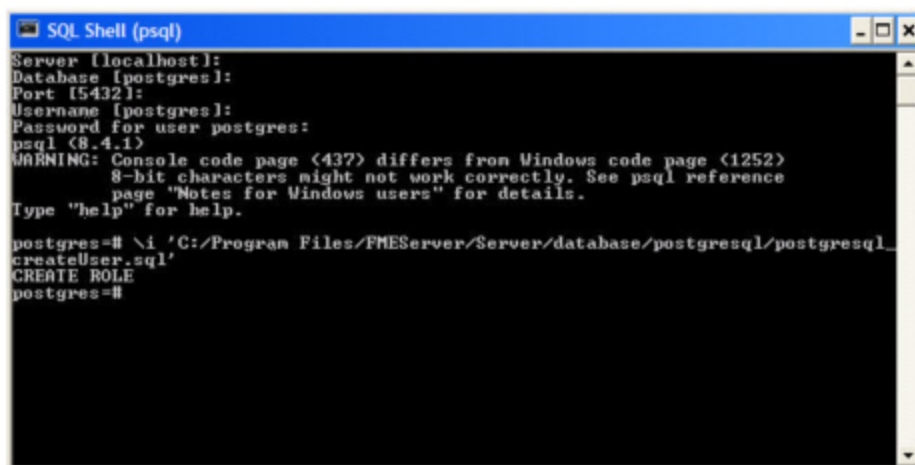
```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.
postgres=# _
```


3. Create an FME Flow database user as follows:

- a. Open `postgresql_createUser.sql` in a text editor, and replace `<<DATABASE_PASSWORD>>` with the password specified during the FME Flow installation.
- b. From the SQL prompt, run the `postgresql_createUser.sql` script by entering the following command:

```
\i '<FMEFlowDir>/Server/database/postgresql/postgresql_createUser.sql'
```

By default, the provided SQL script creates the FME Flow database user and password that you specified during installation:



4. Create an FME Flow database.

From the SQL prompt, run the `postgresql_createDB.sql` script by entering the following command:

```
\i '<FMEFlowDir>/Server/database/postgresql/postgresql_createDB.sql'
```

By default, the provided SQL script creates an FME Flow database named `fmeFlow` and grants all privileges on the database to the user.

```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

postgres=# \i 'C:/Program Files/FME Server/Server/database/postgresql/postgresql_
createUser.sql'
CREATE ROLE
postgres=# \i 'C:/Program Files/FME Server/Server/database/postgresql/postgresql_
createDB.sql'
CREATE DATABASE
GRANT
postgres=# _
```

5. Quit `psql` and log on as the FME Flow database user.

```
SQL Shell (psql)
Server [localhost]:
Database [postgres]: fmeserver
Port [5432]:
Username [postgres]: fmeserver
Password for user fmeserver:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

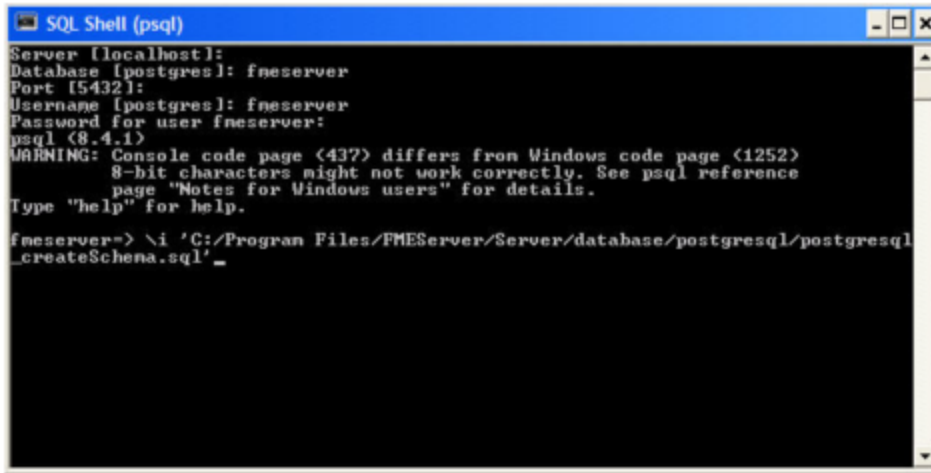
fmeserver=> _
```

6. Create the FME Flow database schema as follows:

From the SQL prompt, run the `postgresql_createSchema.sql` script by entering the following command:

```
\i '<FMEFlowDir>/Server/database/postgresql/postgresql_
createSchema.sql'
```

By default, the provided SQL script creates all FME Flow related tables, indexes, views, and triggers.



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]: fmeserver
Port [5432]:
Username [postgres]: fmeserver
Password for user fmeserver:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

fmeserver-> \i 'C:/Program Files/FME/Server/database/postgresql/postgresql
_createSchema.sql'
```


Linux

This section describes configuring PostgreSQL from both the command prompt and a user interface, such as the PostgreSQL Query Browser and the PostgreSQL Administrator.


For PostgreSQL configurations, various SQL scripts are provided to help with PostgreSQL database configuration. These SQL scripts are located in the `<FMEFlowDir>/Server/database/postgresql/` directory of the FME Flow Core machine.

Note This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `postgresql_createUser.sql` – This script creates the FME Flow database user and password that you specified during installation, and grants all required permissions to the FME Flow database user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.

 **Note** The password is encrypted and appears in the script as <<DATABASE_PASSWORD>>.

- `postgresql_createDB.sql` – This script drops the FME Flow database and creates a new one with the required FME Flow tables. By default this database is named `fmeflow`. Be careful when running this script because it drops the existing FME Flow database.
- `postgresql_createSchema.sql` – This script creates all FME Flow related tables, indexes, views, and triggers.

 **Note** This script must be run as user `fmeflow`.

- `postgresql_dropDB.sql` – This script drops the existing FME Flow database, which is named `fmeflow` by default.
- `postgresql_dropUser.sql` – This script drops the FME Flow database user.

Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install PostgreSQL Server if you haven't already.
2. Open a command prompt and change to the following directory, which contains the PostgreSQL SQL scripts:

```
<FMEFlowDir>/Server/database/postgresql/
```

3. Open `postgresql_createUser.sql` in a text editor, and replace <<DATABASE_PASSWORD>> with the password specified during the FME Flow installation.

4. Run the `postgresql` program with the `postgresql_createUser.sql` script:


```
psql -d postgres -h <hostname> -p <port> -f postgresql_createUser.sql
```

5. Run the postgresql program with the postgresql_createDB.sql script:

```
psql -d postgres -h <hostname> -p <port> -f postgresql_createDB.sql
```

6. Run the postgresql program with the postgresql_createSchema.sql script, as the user you specified during installation:

```
psql -U <username> -d fmeFlow -h <hostname> -p <port> -f postgresql_createSchema.sql
```

 **Note** If you drop the FME Flow database and login user, run the postgresql program with the appropriate postgresql_dropDB.sql or postgresql_dropUser.sql scripts.

After Database Creation

1. On the machine on which the FME Flow database server is installed, open pg_hba.conf. For default installation directories of this file:
 - PostgreSQL 10: `<PostgreSQLDir>\data\pg10\pg_hba.conf`
 - PostgreSQL 11: `C:\Program Files\PostgreSQL\11\data\pg_hba.conf`
2. Change the following lines:

```
host    all    all    127.0.0.1/32    md5
host    all    all    ::1/128         md5
```

to:

```
host    all    all    0.0.0.0/0      md5
host    all    all    ::/0           md5
```


3. [Restart](#) the FME Flow Database service.

Database Configuration - Microsoft SQL Server

It is assumed that SQL Server is installed.

One notation used is `<FMEFlowDir>`, which is the installation directory of FME Flow. In Windows, this is typically `C:\Program Files\FMEFlow`.

SQL scripts are provided to help with SQL Server database configuration. These SQL scripts are located in the `<FMEFlowDir>\Server\database\sqlserver\` directory of the FME Flow Core machine.

 **Note** This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

If you are upgrading, you should back up any job history you want to keep.

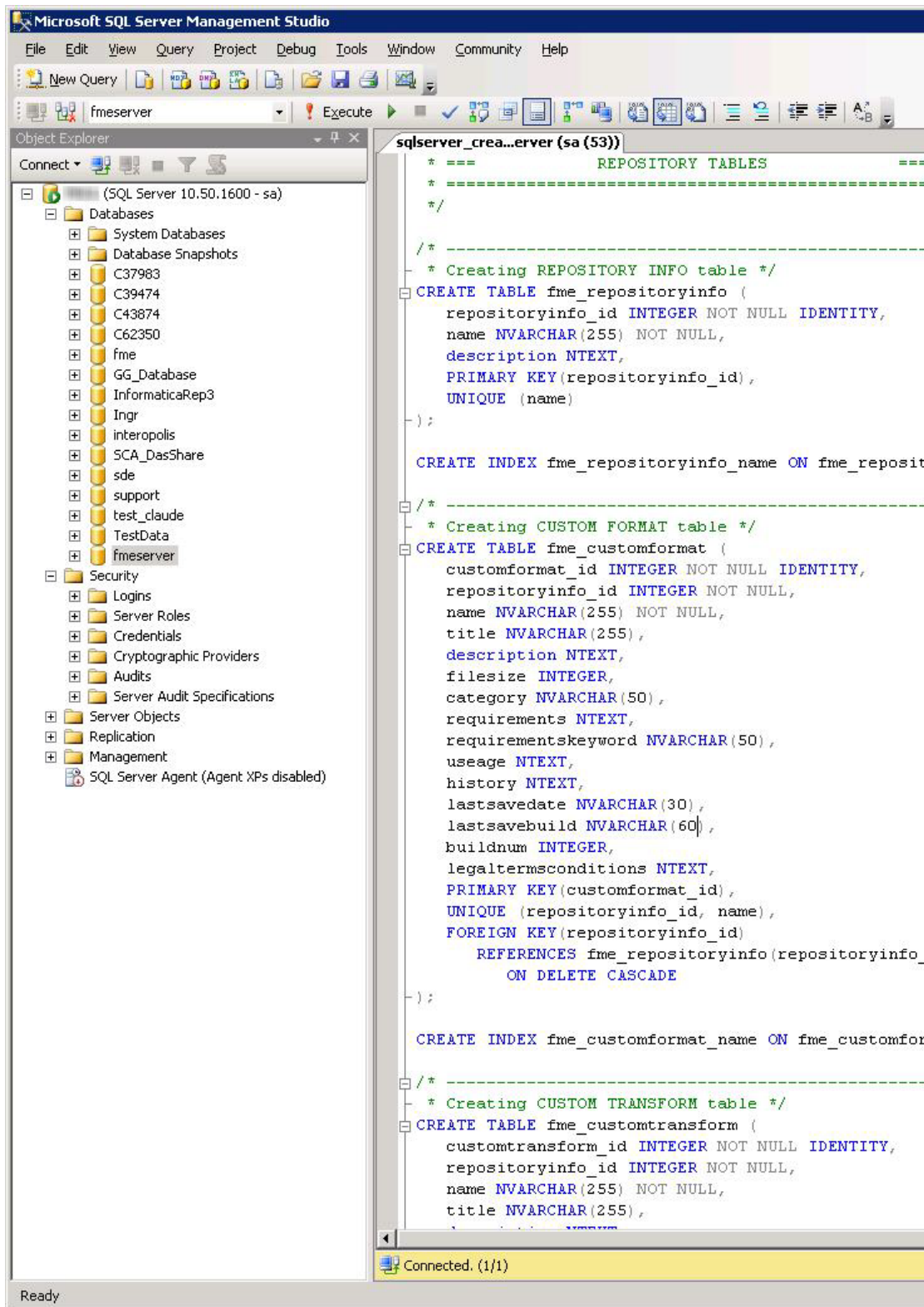
Prerequisite for Unsecured Database Servers

If your SQL Server database server is not secured with a valid SSL certificate, you must update the [fmeDatabaseConfig.txt](#) configuration file as follows:

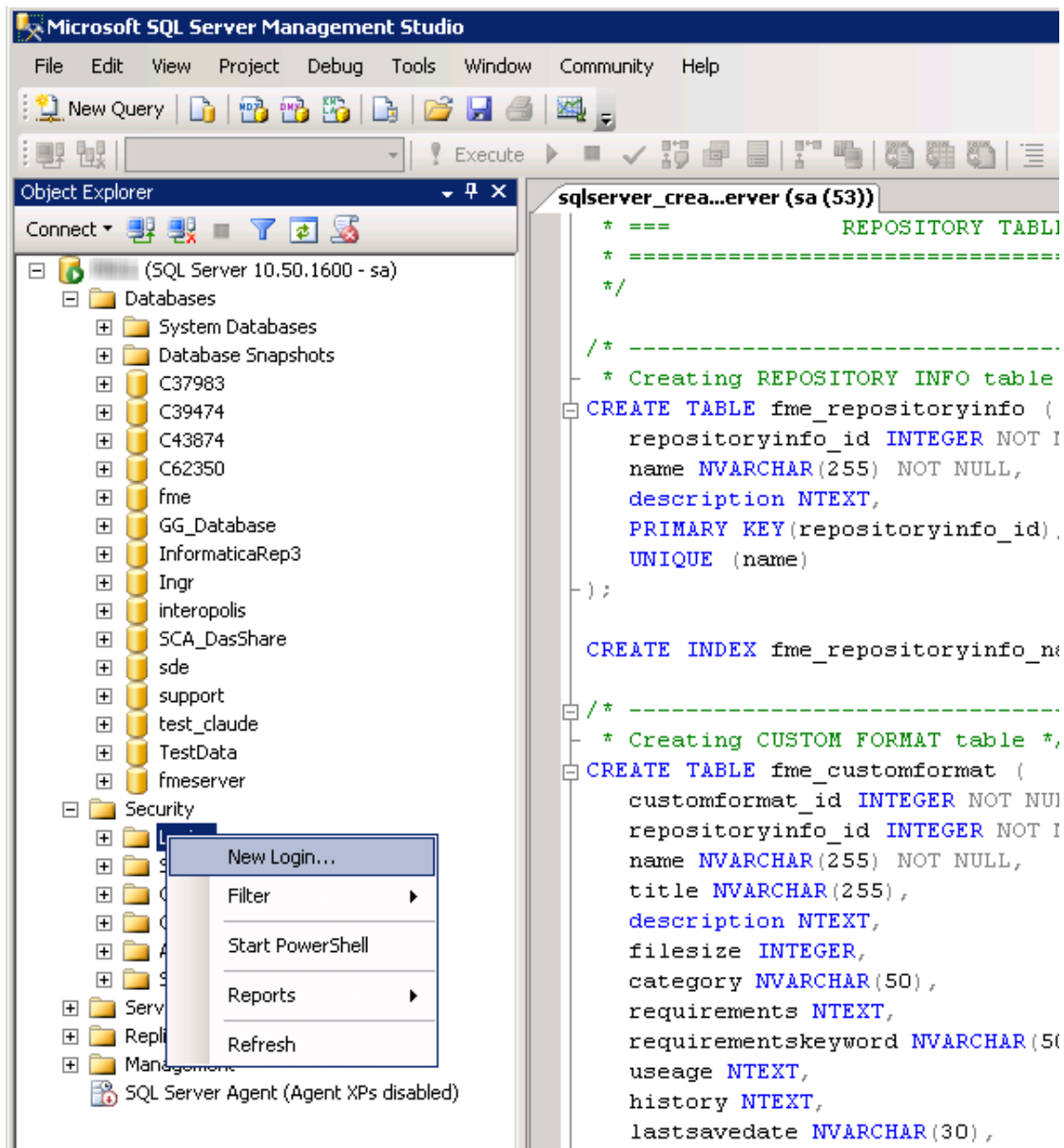
1. Near the bottom, under the heading `FME SERVER SETTINGS START`, locate the block of code beginning with `DB_TYPE=sqlserver`.
2. In the corresponding line for the `DB_JDBC_URL` parameter, update the `encrypt` value from `true` to `false`.
3. Save and close the file.

Instructions Using SQL Server Management Studio

1. Open the SQL Server Management Studio and connect to the SQL Server database engine.
2. Open `sqlserver_createDB.sql` and run the script against the `fmeFlow` database, as shown below:



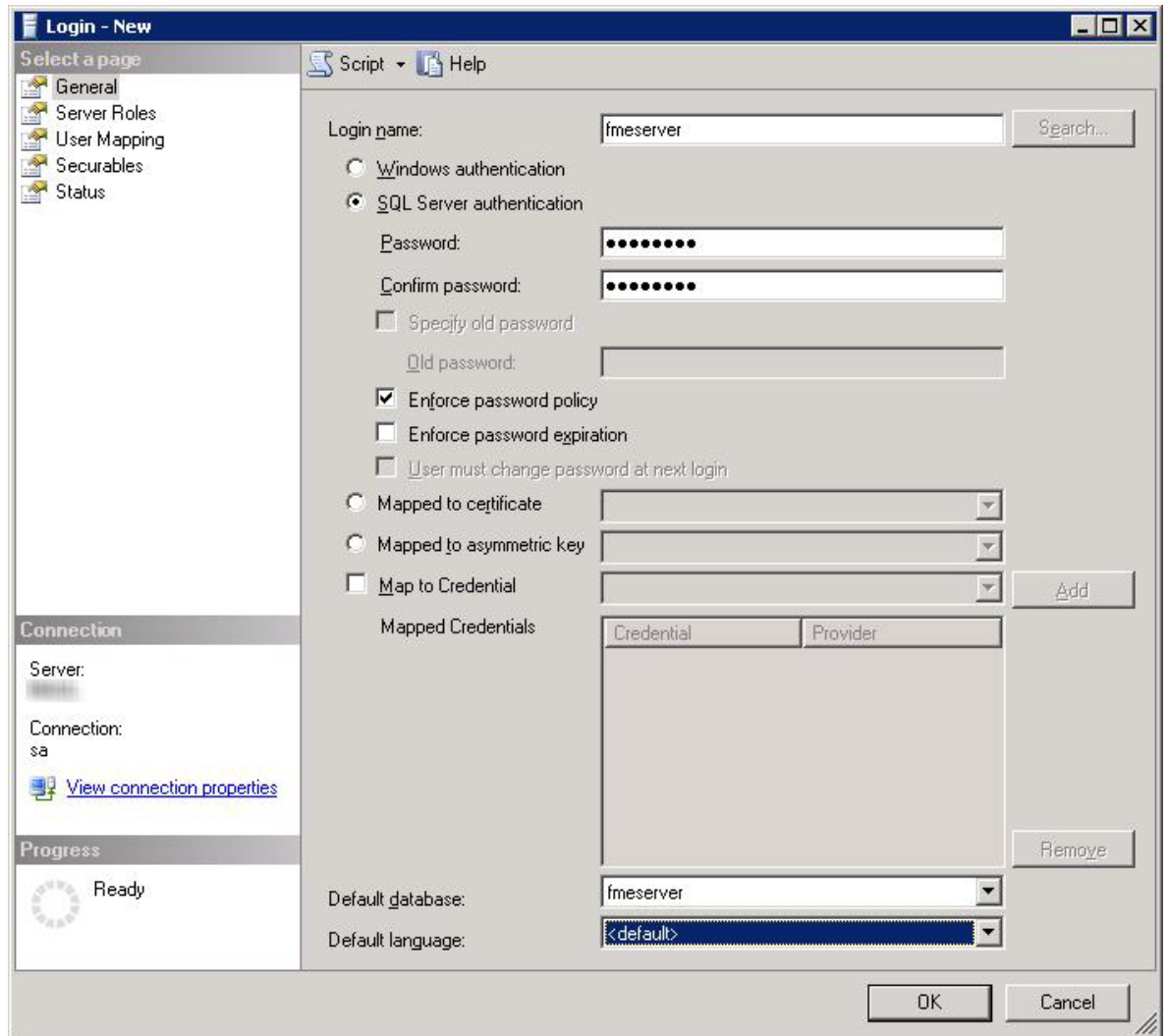
3. On the Object Explorer pane, expand the Security folder, right-click Login, and then select New Login:



4. On the Login – New page, click General to open the page shown in the example below.

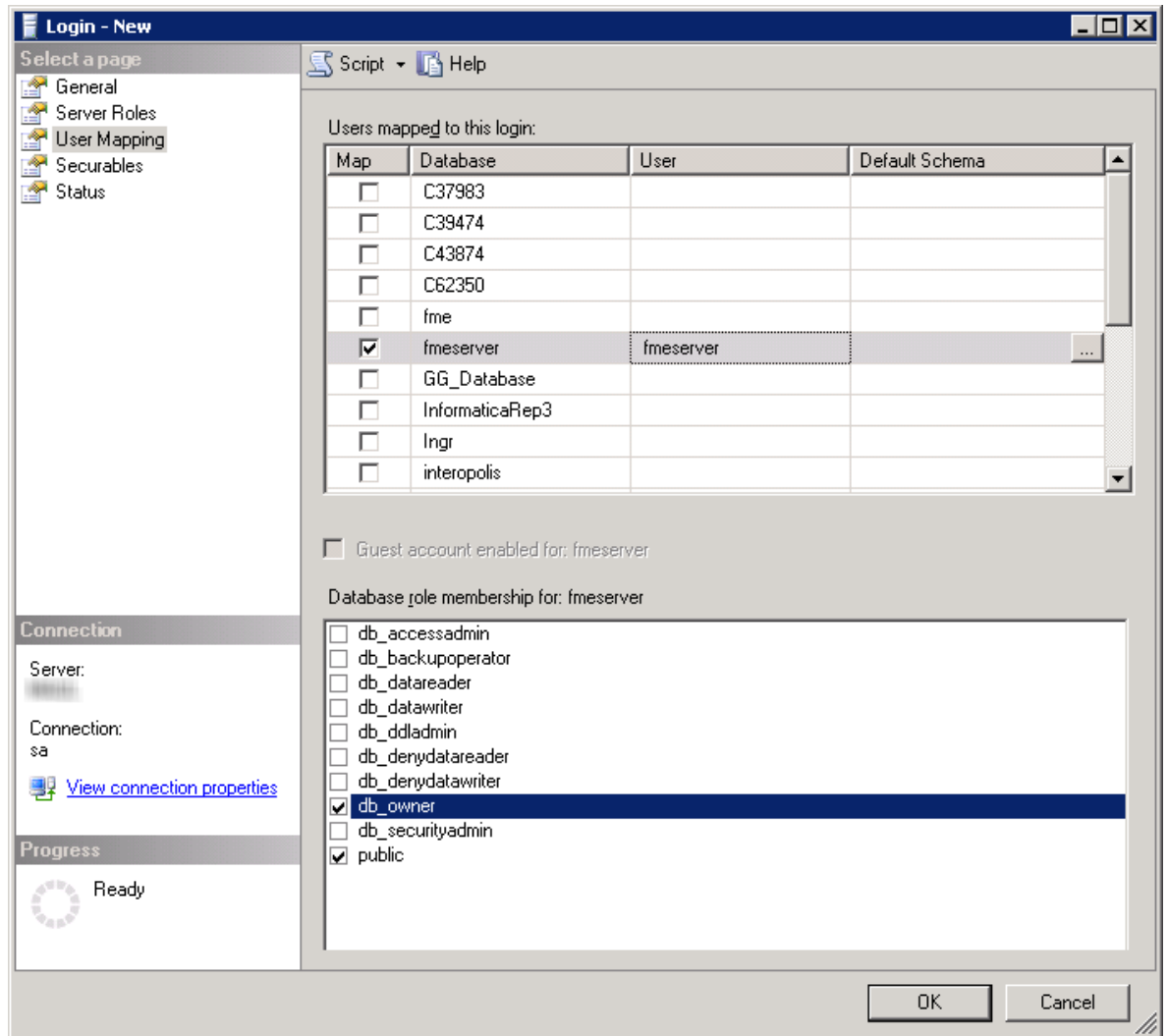
- At the Login name field, enter **fmeFlow**.
- Make sure the option button is active for SQL Server authentication.
- In the Password field enter the password that was specified for the database during the FME Flow installation, and in the Confirm password field enter the same password again.

- Make sure the Enforce password policy is active. If not, click the checkbox to activate it.
- If required, click the checkbox for Enforce password expiration to clear it.
- At the Default database field enter fmeserver.



5. On the same Login – New page, click User Mapping to open the page shown in the next figure.
 - At the Users mapped to this login location, in the Map column click the checkbox to activate it and map the login name to the fmeserver database.

- At the Database role membership for location, notice that the fme\flow database is specified and check `db_owner` to assign that role to this database.
- Click OK to close this page.




Instructions Without SQL Server Management Studio

1. Connect to the SQL Server database engine and run `sqlserver_createDB.sql`.
2. Open `sqlserver_createUser.sql`.
3. Look for the line that begins:

```
CREATE LOGIN fme\flow WITH PASSWORD =
```

Replace `<<DATABASE_PASSWORD>>` with the password specified during the installation.

4. Run the script.

 **Note** If your SQL Server uses an Always On availability group, you must also configure the `MultisubnetFailover` property of the `DB_JDBC_URL` parameter in the `fmeDatabaseConfig.txt` configuration file. For more information, see the SQL Server examples under [Configure the Database Connection](#).

Database Configuration - Oracle

Before proceeding, you must provide the following:

- Install an Oracle database server.
- Obtain the Oracle Database JDBC driver. For more information, see <https://www.oracle.com/technetwork/database/application-development/jdbc/downloads/index.html>.

`<FMEFlowDir>` is the installation directory of FME Flow. In Windows, this is typically `C:\Program Files\FMEFlow`.


If you are upgrading, you should back up any job history you want to keep.

Windows

This section describes how to configure an Oracle database. You can configure the Oracle database from other Oracle database tools depending on your personal preference.

For an Oracle database configuration, various SQL scripts are provided to help.

These SQL scripts are located in the `<FMEFlowDir>\Server\database\oracle` directory of the FME Flow Core machine.

 **Note** This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `oracle_createUser.sql` – This script creates the FME Flow database user, and grants all required permissions to the FME Flow database user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- `oracle_createDB.sql` – This script creates all FME Flow related database packages, tables, indexes, views, and triggers.
- `oracle_dropUser.sql` – This script drops the FME Flow database user and the database packages, tables, indexes, views, and triggers associated with the FME Flow database user.

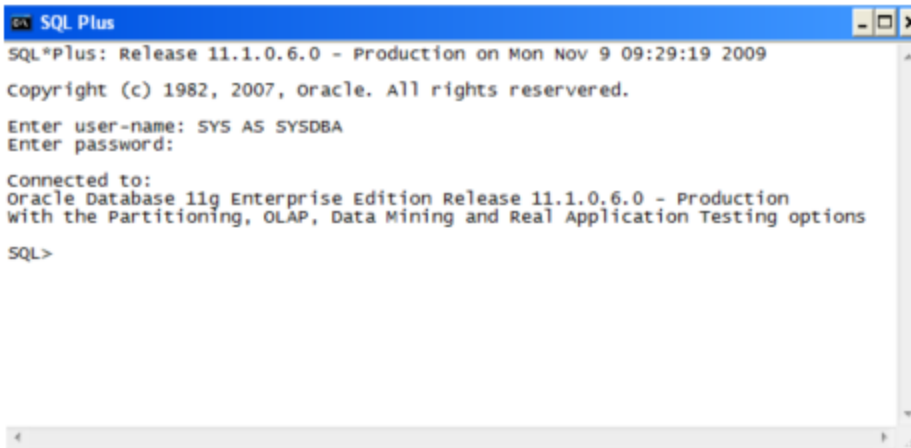
Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install Oracle SQL*Plus, if you haven't already.
2. Open Oracle SQL*Plus and log on as a user with adequate privileges, such as SYSDBA.

In this example, the following parameters were used and entered:

- User: SYS AS SYSDBA
- Password: fmefflow
- String: XX-ORACLE11G:1521/ORCL AS SYSDBA

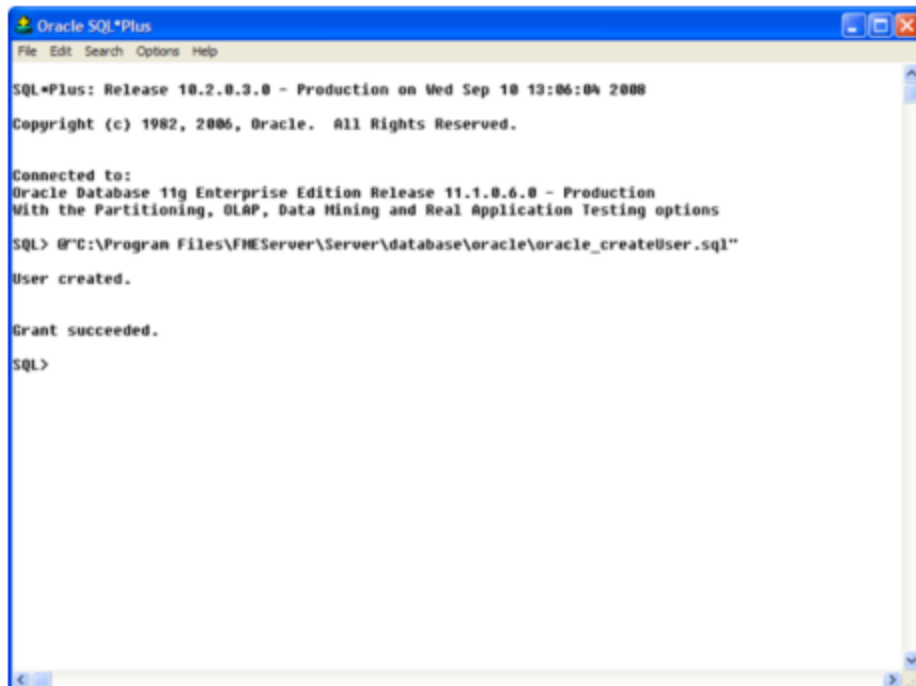


```
SQL Plus
SQL*Plus: Release 11.1.0.6.0 - Production on Mon Nov 9 09:29:19 2009
copyright (c) 1982, 2007, oracle. All rights reserved.
Enter user-name: SYS AS SYSDBA
Enter password:
Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
SQL>
```

3. Create an FME Flow database user as follows:

- a. In `oracle_createUser.sql`, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- b. From the SQL prompt, run the `oracle_createUser.sql` script by entering the following command:

`@<FMEFlowDir>\Server\database\oracle\oracle_createUser.sql`



```
Oracle SQL*Plus
File Edit Search Options Help
SQL*Plus: Release 10.2.0.3.0 - Production on Wed Sep 10 13:06:04 2008
Copyright (c) 1982, 2006, Oracle. All Rights Reserved.
Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
SQL> @C:\Program Files\FMEServer\Server\database\oracle\oracle_createUser.sql
User created.
Grant succeeded.
SQL>
```

The default TABLESPACE is the Oracle DB configured default. If the created user doesn't have read/write permissions on the default TABLESPACE, change the default TABLESPACE or create an appropriate TABLESPACE.

This command creates a TABLESPACE on the Oracle DB server called FMEFLOW shown in the following example:

```
CREATE SMALLFILE TABLESPACE FMEFLOW DATAFILE  
'E:\ORADATA\FMEFLOW01.DBF' SIZE 100000K REUSE AUTOEXTEND ON  
NEXT 2048K MAXSIZE 1024M LOGGING EXTENT MANAGEMENT LOCAL  
SEGMENT SPACE MANAGEMENT AUTO DEFAULT NOCOMPRESS;
```

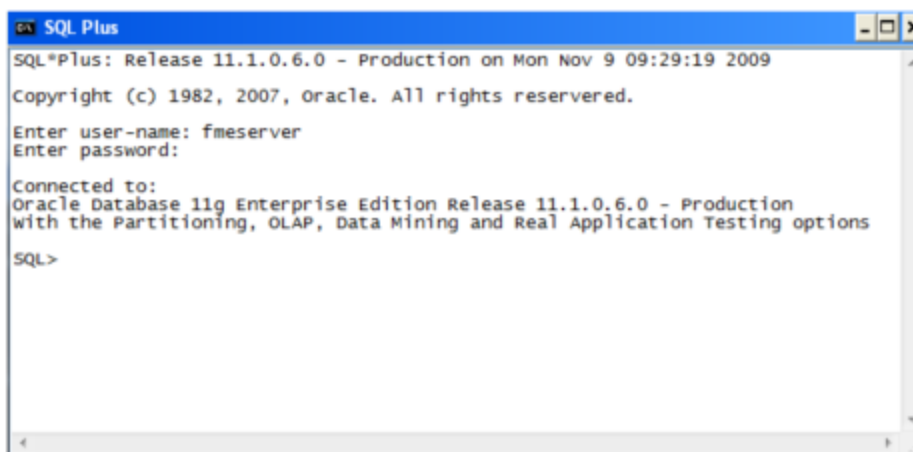
By default this command uses the TABLESPACE "FMEFLOW" as shown in the following example:

```
CREATE USER fmeflow IDENTIFIED BY fmeflow DEFAULT TABLESPACE  
"FMEFLOW" TEMPORARY TABLESPACE "TEMP";
```

4. When an FME Flow database user is created, you can connect to it by logging on again to SQL*Plus.

In this example, the following parameters were used and entered on the user interface:

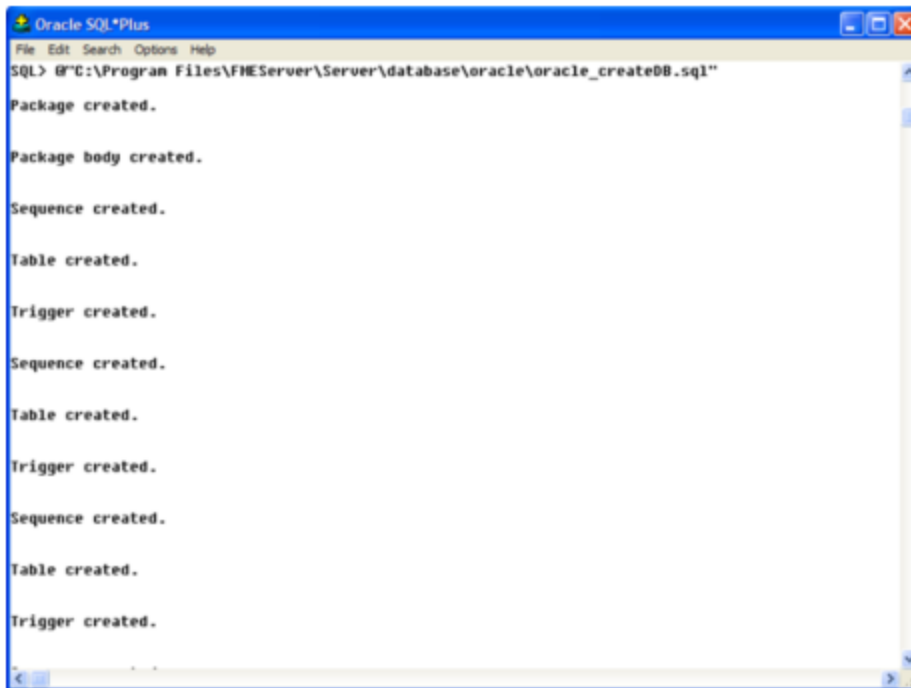
- User: fmeflow
- Password: fmeflow



Note The preceding example assumes your user name and password are fmeflow.

5. When connected to the FME Flow database, run the `oracle_createDB.sql` script by entering the following command:

```
@"<FMEFlowDir>\Server\database\oracle\oracle_createDB.sql"
```



Note If you drop the FME Flow database and login user, run the `oracle_dropUser.sql` script.

Linux

Oracle SQL*Plus is a freeware client for connecting to an Oracle database. You can download it from the Internet using the following address:


https://www.oracle.com/technology/software/tech/sql_plus/index.html

For more information, see the Oracle web site at:

https://www.oracle.com/technology/sample_code/tech/sql_plus/htdocs/demobld.html

The instructions that follow describe how to configure Oracle for use with FME Flow. These instructions describe using SQL*Plus for the database setup. Depending on your personal preference, you can also configure the Oracle database using other Oracle tools.

For Oracle configurations, various SQL scripts are provided to help with Oracle database configuration. These SQL scripts are located in the `<FMEFlowDir>/Server/database oracle/` directory of the FME Flow Core machine.

 **Note** This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `oracle_createUser.sql` – This script creates the FME Flow database user, and grants all required permissions to the FME Flow database user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- `oracle_createDB.sql` – This script creates all FME Flow related database packages, tables, indexes, views, and triggers.
- `oracle_dropUser.sql` – This script drops the FME Flow database user and the database packages, tables, indexes, views, and triggers associated with the FME Flow database user.

Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install the Oracle Client Tools, which includes SQL*Plus, if you haven't already.
2. Run `sqlplus` and log on as a user with adequate privileges, such as SYSDBA.

For example:

```
sqlplus SYS/<password>@<Host>:<Port>/<service> AS SYSDBA
```

3. Create the FME Flow database user:
 - a. In a text editor, open the `oracle_creatorUser.sql` script and update the password `<<DATABASE_PASSWORD>>` to the one specified during the FME Flow installation.
 - b. From the SQL prompt, run the `oracle_createUser.sql` script by entering the following command:

```
@ "<FMEFlowDir>/Server/database/oracle/oracle_createUser.sql"
```

The default TABLESPACE is the Oracle DB configured default. If the created user doesn't have read/write permissions on the default TABLESPACE, you need to change the default TABLESPACE or create an appropriate TABLESPACE.

This command creates a TABLESPACE on the Oracle DB server called FMEFLOW shown in the following example:

```
CREATE SMALLFILE TABLESPACE FMEFLOW DATAFILE  
'/ORADATA/FMEFLOW01.DBF' SIZE 100000K REUSE AUTOEXTEND ON  
NEXT 2048K MAXSIZE 1024M LOGGING EXTENT MANAGEMENT LOCAL  
SEGMENT SPACE MANAGEMENT AUTO DEFAULT NOCOMPRESS;
```

By default this command uses the TABLESPACE "FMEFLOW" shown in this example:

```
CREATE USER fmeFlow IDENTIFIED BY fmeFlow DEFAULT TABLESPACE  
"FMEFLOW" TEMPORARY TABLESPACE "TEMP";
```

4. When an FME Flow database user has been created, you can connect to it by logging on again to SQL*Plus or by entering the following command:

```
CONNECT <UserName>/<Password>@<Host>:<Port>/<service>;
```

5. When connected to the FME Flow database, run the oracle_createDB.sql script by entering the following command:

```
@"<FMEFlowDir>/Server/database/oracle/oracle_createDB.sql"
```

2. **Enable connections:** FME Flow connects to the database when it starts. Therefore, ensure that your database server is running and configured to accept incoming connections before FME Flow is started. The database must allow connections over TCP/IP with all machines on which the FME Flow Core and FME Engines are installed.
3. Ensure that all installations of FME Flow are running. For more information, see [Working with the FME Flow System Services](#).

What's Next?

- [Set Up the Load Balancer and Configure with FME Flow](#)

Set Up the Load Balancer and Configure with FME Flow

Use these instructions to:

1. Set up your load balancer to prepare for routing traffic between each FME Flow instance you installed under [Installing a Scalable, Fault-Tolerant FME Flow](#).
2. Configure FME Flow topic monitoring to recognize the load balancer URL.

1. Set Up the Load Balancer

Load Balancing the FME Flow Web Services

Required

- Route traffic on port 80 (Windows) or port 8080 (Linux) to each FME Flow instance in your deployment.

Recommended

- Set up a health check for FME Flow. A health check endpoint determines if either the [FME Flow Web Services](#) or [FME Flow Core](#) node is down, and if so, requests are not routed to that server. Without a health check configured, the load balancer may try to route client requests to an unhealthy (down) server. The consequences of routing requests to an unhealthy server vary depending on the load balancer, and may result in error responses or long delays in response due to request timeouts and retrying on a healthy server.

Optional

- Configure timeouts for client- and server-side inactivity. When configuring timeouts, consider that longer timeouts on the server are appropriate for time-consuming operations, particularly running workspaces that take a long time to complete.

Load Balancing the WebSocket Server

An FME Flow [WebSocket Server](#) runs on each FME Flow instance, but only one WebSocket Server should be used at a time. The WebSocket Server is used mainly for Notification Service [Topic Monitoring](#).

Required

- Route traffic on port 7078 to one WebSocket Server, and switch to others if the one in use fails. If the load balancer does not support automatic failover to backup servers, then manual intervention is required to route to a healthy WebSocket Server.

Recommended

- Set longer client- and server-side timeouts to allow for longer WebSocket connections.

2. Configure FME Flow Topic Monitoring with the Load Balancer URL

1. On the [Networking](#) page, expand Topic Monitoring.
2. Under Fault Tolerant Configuration, update the *Internal WebSocket URL* field with the public load balancer URL. [Automations](#) Destinations (and [Notification Service](#)

Subscribers) use this URL to communicate with the FME Flow WebSocket Server. The load balancer URL redirects requests to the active WebSocket Server.

3. If necessary, update Advanced Routing settings:

- *External WebSocket Host*: The load balancer name. For example:

`fme-flow-loadbalancer`

This setting does not likely need to be changed, and should only be set if you use separate hosts to route HTTP and WebSocket traffic to FME Flow.

- *External WebSocket Port*: If the WebSocket Server is accessed through a port other than 7078, specify the port number here. Otherwise, leave blank.

4. Click **Save**.

What's Next?

Proceed to [Log on to the Web User Interface](#).

Log on to the Web User Interface

To log in to the Web User Interface for the first time as an administrator following installation, specify *Username* `admin` and *Password* `admin`. You will be prompted to update the `admin` password upon initial use, based on the default [Password Policy](#) configuration.

For more information and support:

- [FME Flow Troubleshooting: Web Interface Login](#)
- [Accessing the Web User Interface](#)
- [Changing the Login Password](#)
- [Default User Accounts and Passwords](#)
- [Role-Based and User-Based Access Control](#)

What's Next?

- [Request and Install a License](#)

Request and Install a License

Note

License files are installed, by default, to:

» Windows: `C:\ProgramData\Safe Software\FME Flow\licenses`

» Linux: `/opt/fmeflow/licenses`

Dynamic Engine (CPU Usage) licensing is not available for engines on FME Flow Hosted.

Automatic Mode (Recommended)

If your FME Flow host machine has an internet connection, you can request and install a license online without any need for downloading and re-uploading a license file.

Note


- Licensing FME Flow over the internet requires no inbound connection. For more information, see [FME Flow Automatic Licensing URLs](#).
- If you purchased credits for Dynamic Engines (CPU Usage), automatic mode only is supported. Manual mode licensing (below) is not supported.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.
3. Complete the fields on the Request a New License page.

Note If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.


4. *Licensing Mode*: Select **Automatic**.

5. Click **OK**.
6. A message indicates that your license is installed, and your available engines appear momentarily on the [Engines](#) page.


 **Note** If the *Serial Number* you provided includes Dynamic Engines (CPU Usage), they are not configured to start by default. You can start and configure your Dynamic Engines on the [Engines](#) page.

Manual Mode

If you prefer not to use an internet connection for licensing, or if your FME Flow host machine does not have an internet connection, use this process to install a license.

 **Note** If you purchased credits for Dynamic Engines (CPU Usage), manual mode licensing is not supported. Use automatic mode for licensing (above). For more information, contact your Safe Software Inc. sales representative.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Flow engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.
3. Complete the fields on the Request a New License page.

 **Note** If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.

4. *Licensing Mode*: Select **Manual**.
5. Click **OK**. A license request .json file downloads.
6. Email the .json file to codes@safe.com.

7. You will receive an email from Safe Software Inc. with a license file, which has a `.fmelic` extension. Download the file.
8. Return to the Licensing page. Under Standard Engines, click **Upload License File** and select the `.fmelic` file to upload. Or, drag and drop the `.fmelic` file onto the page under Standard Engines. A message indicates that it is installed, and your available engines appear on the [Engines](#) page.

Problems?

- [FME Flow Troubleshooting: FME Flow Engines](#)
- Contact codes@safe.com.

See Also

- [FME Flow Automatic Licensing URLs](#)
- [FME Engines](#)

What's Next?

- [Test the Installation](#)

Test the Installation

To test your FME Flow installation, confirm that FME Flow can perform its primary function—running a job.

1. On the [Run Workspace](#) page, specify:
 - Repository: Samples
 - Workspace: austinApartments.fmw
 - Service: Job Submitter
2. Click Run Workspace.

If you see a success message, your install was a success.

See Also

- [FME Flow Troubleshooting: Submitting a Job in FME Flow](#)


Distributing Components (2-Tier)

Before installing a 2-tier distributed architecture, make sure you have:


- Reviewed [Planning an FME Flow Installation](#).
- Provided the prerequisites in [Before Installation](#).
- Linux: Ensured [System Requirements](#) are met, including updates and dependencies.

To perform a 2-tier installation of FME Flow:

1. [Obtain the Installer](#)
2. [Install the FME Flow Database](#)

 **Note** Complete this step *only* if you wish to install a PostgreSQL database that is included with the FME Flow installer, and do *not* wish to configure the FME Flow Database on your own database server. For more information, see [Provide a Database Server](#).

3. [Install the FME Flow Core, FME Engines, and Web Application Server](#).
4. [Configure the FME Flow Database](#)

 **Note** Complete this step *only* if you are configuring the FME Flow Database on your own database server, and did *not* install the PostgreSQL database that is included with the FME Flow installer. For more information, see [Provide a Database Server](#).

5. [Start the FME Flow system services](#). Following a distributed installation, the [FME Flow system services](#) do not start automatically. You must manually start them on each machine that is hosting FME Flow components.
6. [Log on to the Web User Interface](#)
7. [Request and Install a License](#)
8. (Recommended) [Update the Windows service accounts](#) that run the FME System Services.
9. (Upgrade Only) [Restore](#) your FME Flow Configuration from a Previous Installation.
10. [Test the Installation](#).

What's Next?

Optimize: You can scale-up your FME Flow by:

- Increasing the number of FME Engines that run when you start FME Flow.
- Adding or accessing FME Engines on separate machines that are closer to your data.

For more information, see [Planning for Scalability and Performance](#).

Before Installation

Before you install the FME Flow components for a 2-tier system, ensure the following non-FME Flow components are in place on your network:

- [Database server](#) on a separate machine.
- [File system](#) on a separate machine.
- [Web application server](#), *only* if you want to use your own Apache Tomcat web application server, instead of the one provided with the FME Flow installer.

Next Steps

Proceed to [Obtain the Installer](#).


Provide a Database Server

FME Flow uses its own [database](#) to manage jobs and workspace information. It is not a source or destination data source for FME workspaces.


To ensure the FME Flow Database remains available in a failover scenario, provide a machine on which to install the database that is physically separate from the machines on which the FME Flows are installed. In addition to being physically separate, the database server should, by itself, be configured for fault tolerance.

You can choose to install a PostgreSQL database that is included with the FME Flow installer, or you can configure the FME Flow Database on your own database server. If you use your own database server, keep in mind the following:

- PostgreSQL (recommended), Oracle, and SQL Server are supported. The following versions are recommended:

 **Note** PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.

- PostgreSQL: 16.1 or later.
- Oracle: 19c or later.
- SQL Server: 2016 or later.
- When you install the FME Flows, the database server must be running and you must know the connection information.
- If using an Oracle database server, you must obtain the Oracle Database JDBC driver. For more information, see <https://www.oracle.com/technetwork/database/application-development/jdbc/downloads/index.html>.

 **Note** Oracle Database JDBC driver versions that use ojdbc10.jar are *not* supported.

- Following installation, you must be able to create a new database on the server using SQL scripts.

For information about acquiring a fault-tolerant database server, consult your IT department.

Updating the FME Flow Database Password

If you choose to install a PostgreSQL database that is included with the FME Flow installer, and want to update the password for the database user account, follow these instructions:

1. Start pgAdmin4.exe from `<FMEFlowDir>\Utilities\pgsql\pgAdmin 4\bin\`.
2. If this is your first time starting pgAdmin4, it will prompt you to set a master password. Choose something you will remember.
3. If pgAdmin auto-connects to the FME Flow Database, an entry appears in the Object browser, under Servers, similar to the following:

```
fmeserver (<hostname>:7082)
```

If pgAdmin III does not auto-connect to the FME Flow Database, add a new Server for it. To add a New Server:

- a. Right-click on **Servers** and Select **Create > Server....**
- b. In the **General** tab of the Create - Server dialog, specify the *Name* for the server connection. This can be anything, such as `fmeFlow`.
- c. In the **Connection** tab of the Create - Server dialog, specify the server properties, using the `Database Connection` details in the [fmeDatabaseConfig](#) configuration

file, for `DB_TYPE=postgresql`, as follows:

Property	Value from fmeDatabaseConfig.txt
Host name/address	DB_JDBC_ URL=jdbc:postgresql://<hostname>:<port>/<database_name>
Port	DB_JDBC_ URL=jdbc:postgresql://<hostname>:<port>/<database_name>
Maintenance database	DB_JDBC_ URL=jdbc:postgresql://<hostname>:<port>/<database_name>
Username	DB_USERNAME=<username>
Password	DB_PASSWORD=<password>

d. Click **Save**.

- In the Object browser, select the FME Flow Database server, and locate and expand **Login/Group Roles**.
- Right-click the fmeserver database user account and select **Properties**. Under **Definition**, change the *Password* and click **Save**.
- In the [fmeDatabaseConfig](#) configuration file, update the value of the `DB_PASSWORD` parameter that corresponds to `DB_TYPE=postgresql` with the new password specified above.

When finished, save the file.

- (Recommended) [Encrypt](#) the FME Flow Database password.
- [Restart](#) FME Flow.

1. Open a command prompt and change to the following directory:

```
<FMEFlowDir>/Utilities/pgsql/bin
```

2. Run the following command to connect to the FME Flow Database:

```
./psql -d <databaseName> -p <port> -U <username>
```

To confirm the settings, check the **Database Connection** details in the [fmeDatabaseConfig](#) configuration file, for **DB_TYPE=postgresql**, as follows:

Property	Value from fmeDatabaseConfig.txt
-d	DB_JDBC_ URL=jdbc:postgresql://<hostname>:<port>/< database_name >
-h	DB_JDBC_ URL=jdbc:postgresql://< hostname >:<port>/<database_name>
-p	DB_JDBC_ URL=jdbc:postgresql://<hostname>:< port >/<database_name>
-U	DB_USERNAME=< username >
Password (when prompted)	DB_PASSWORD=< password >

3. Once connected, enter the following command to change the password for your user:

```
\password
```

4. Use `\q` to quit psql.

5. In the the [fmeDatabaseConfig](#) configuration file, update the value of the `DB_PASSWORD` parameter that corresponds to `DB_TYPE=postgresql` with the new password specified above, and save the file.
6. (Recommended) [Encrypt](#) the FME Flow Database password.
7. [Restart](#) FME Flow.

Provide a Remote File System for the FME Flow System Share

- **Skill Level:** Intermediate
- **Estimated Time Required:** 5-15 minutes
- **Prerequisites:**
 - Domain [Service Account](#) has correct permissions on network shares.

FME Flow System Share files, which include [Repositories](#) and [Resources](#), are installed in specific directories that are shared to enable access from different computers.

Create a directory for the FME Flow System Share that is physically separate from the machine on which the FME Flow Core and web application server are installed. Separating the file system ensures it remains available in a fault tolerant scenario.

When providing a remote file system, keep in mind the following:

- The accounts that run the FME Flow components must have read and write permissions on the remote file system directory. For more information, see [Directory and Account Permissions](#).
- In addition to being physically separate, the file system should, by itself, be configured for fault tolerance.
- When the directory is specified during installation, it must be in the form of a UNC path. Mapped drives are not supported.
- The directory that holds the System Share must be empty, unless System Share folders have already been created there as part of an installation involving multiple FME Flow Cores.


For information about acquiring a fault tolerant file system, consult your IT department.

Linux Only

For distributed installations on Linux, it is necessary to create a user, prior to installation, on the machine that will host the installation of the FME Flow Core and Engines. This user must be named 'fmeflow' and belong to group 'fmeflow', and be granted permissions on the FME Flow System Share. For more information, see [Directory and Account Permissions](#).

Example Instructions

The following example creates user 'fmeflow' in group 'fmeflow', and allows mount /mnt/data to be writable by the newly-created user.

 **Note** The following instructions are provided as a general guideline for adding a user and mounting a directory. Exact instructions may vary depending on the distribution and share type in your Linux configuration.

1. Run the following commands in the Linux command line:

```
groupadd fmeflow
useradd -m -g fmeflow fmeflow
```

2. Mount the FME Flow System Share directory so that user 'fmeflow' has read and write permissions. For example, to mount a Windows-based file share:


```
mount -v -t cifs //network/share /mnt/data -o gid=1008,uid=1008
```

`gid` and `uid` are the group id and user id of the user and group that were created.

These id's can be found using these commands:

```
id -g fmeflow
id -u fmeflow
```

Provide a Web Application Server


 **Note** Complete this step *only* if you want to use your own Apache Tomcat web application server, instead of the one provided with the FME Flow installer.

Provide a web application server on the local system of your FME Flow installation.

FME Flow supports Apache Tomcat version 9.0.x. On Windows, we recommend running the web application server on the same version of the Java Runtime Environment as FME Flow runs. To determine this version, use a text editor to open file `release` in `<FMEFlowDir>\Utilities\jre`.

The following are *not* supported:

- Apache Tomcat version 10.x.
- Java Runtime Environment 9, 16, 18 or later.

 **Note** On Linux, you must add the following environment variable to `etc/environment`:

```
JAVA_HOME="/usr/lib/jvm/java-8-openjdk-amd64"
```

You must know the port number on which your web application server receives incoming requests and the path to the directory where your web application server stores web application `.war` files. (This directory is normally called `webapps`.)

Additionally, you must allow the necessary system accounts permission to run the web application server. For more information, see [Directory and Account Permissions](#).

Obtain the Installer

To obtain the applicable FME Flow installation package, visit [the FME Downloads page](#). This page includes download links to release and beta installation packages for various platforms.


You can also contact sales@safe.com to get access.

What's Next?

If you are opting to install, on a separate machine, a PostgreSQL database that is included with the FME Flow installer, and do *not* wish to configure the FME Flow Database on your own database server, proceed to [Install the FME Flow Database](#). Otherwise, proceed to [Install the FME Flow Core, FME Engines, and Web Application Server](#).

Install the FME Flow Database


Install a PostgreSQL FME Flow Database on a separate machine on your network.

 **Note** Complete this step *only* if you wish to install, on a separate machine, a PostgreSQL database that is included with the FME Flow installer, and do *not* wish to configure the FME Flow Database on your own database server. For more information, see [Provide a Database Server](#).

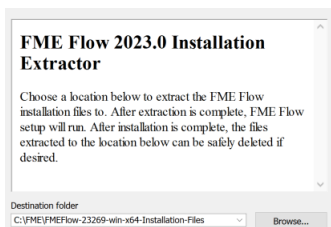
Start the FME Flow installer.

(Windows only) Installation Extractor

On the Installation Extractor dialog, specify a *Destination folder* in which to extract installation files. Note that this folder *only* specifies where installation files are extracted—it does *not* specify where FME Flow program files are installed. That location is specified later in the installer.

 **Tip** Take note of the specified *Destination folder* location. Following installation, you may wish to:

- Review the installation .log file.
- Remove these files to save disk space.

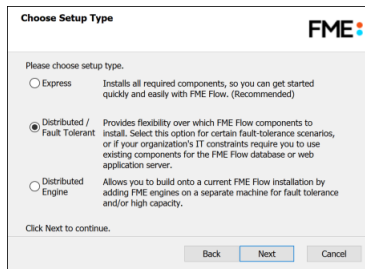


Click **Install**.

When extraction is complete, the installer opens. Proceed through the installer to the Choose Setup Type dialog.

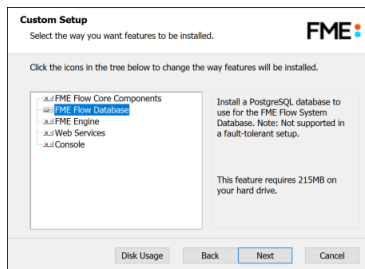
Choose Setup Type

Select the Distributed/Fault Tolerant install option.



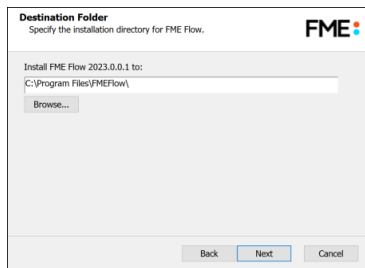
Custom Setup

Specify FME Flow Database. Unselect all other components.



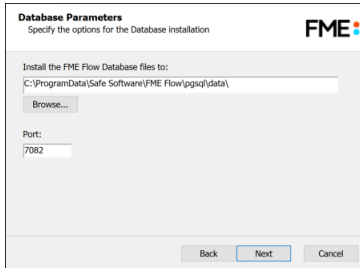
Destination Folder

Specify the installation directory for the FME Flow Database.



Database Parameters

- *Install the FME Flow Database files to:* Specify a directory to install the database files.
- *Port:* Specify the port that the database will listen on. Port **7082** is the default.

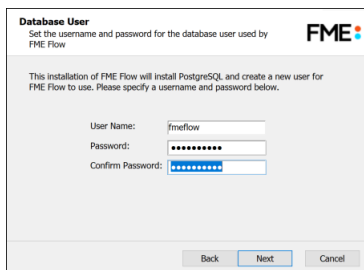


Database User

Specify a *User Name* (defaults to **fmeflow**) and *Password* for the database user account.

Warning Do not specify *User Name* **postgres**. This name corresponds to the PostgreSQL master user. If the corresponding passwords do not match, the database will not install.

Note The password for the FME Flow Database user account is encrypted in FME Flow installation files. Take note of the *User Name* and *Password* that you specified. You may need to reference it later.



Follow the remaining dialogs to complete the installation.

What's Next?


- [Install the FME Flow Core, FME Engines, and Web Application Server](#)

Install the FME Flow Core, FME Engines, and Web Application Server

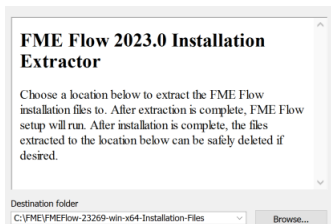
Start the FME Flow Installer.

(Windows only) Installation Extractor

On the Installation Extractor dialog, specify a *Destination folder* in which to extract installation files. Note that this folder *only* specifies where installation files are extracted—it does *not* specify where FME Flow program files are installed. That location is specified later in the installer.

 **Tip** Take note of the specified *Destination folder* location. Following installation, you may wish to:

- Review the installation .log file.
- Remove these files to save disk space.

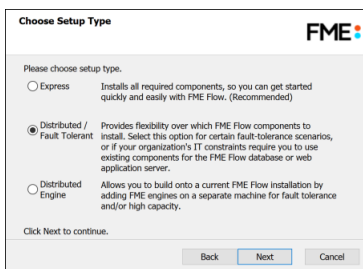


Click **Install**.

When extraction is complete, the installer opens. Proceed through the installer to the Choose Setup Type dialog.

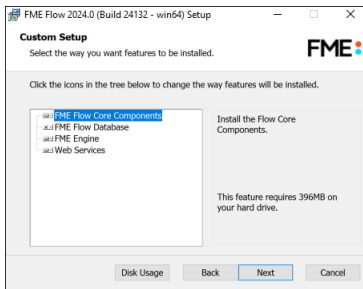
Choose Setup Type

Select the Distributed/Fault Tolerant install option.



Custom Setup

Select **FME Flow Core Components**, **FME Engine**, and **Web Services**.

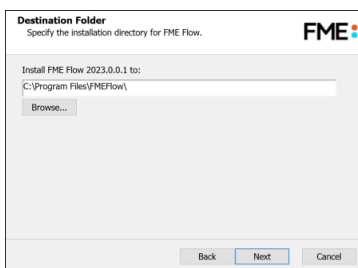


Optionally, select **FME Flow Database**. If you do so, note that:

- This option is supported, but not described in [Distributing FME Flow Components](#). Do not select **FME Flow Database** if you already completed the previous step, [Install the FME Flow Database](#), or if you plan to configure the FME Flow Database on your own database server. If you specify **FME Flow Web Application Server** under [Web Application Server](#) (below), the installation is equivalent to an [Express](#) installation.
- The step to define [Database Server Type](#) (below) will not appear.

Destination Folder

Specify the installation directory for FME Flow.




User Account


Specify the Windows user account that will run the FME Flow [System Services](#). This account must have:

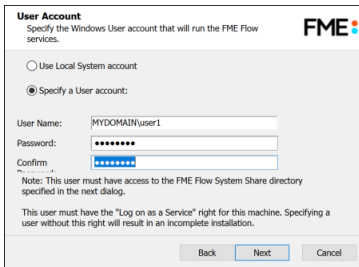
- Read/Write access to the FME Flow System Share that you specify in the next dialog. The System Share is the location where FME Flow stores [Repositories](#) and [Resources](#) files. (For more information, see [Provide a Remote File System for the FME Flow System](#))

[Share](#).)

- Read/Write access to the FME Flow installation directory that you specified in the previous dialog.
- "Log on as a service" rights on this machine. For more information, see [this Microsoft Docs article](#).


 **Note** If unsure, you can update account settings for the FME Flow system services after installation in the Services manager (Windows) or Linux equivalent.


 **Note** The FME Flow Database service always runs under the local system account.

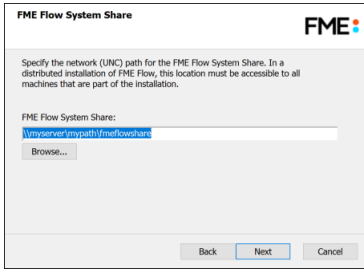


FME Flow System Share

Specify a directory on another machine to store FME Flow System Share files, which include [Repositories](#) and [Resources](#). If this is a Linux installation, specify the mounted directory. For more information, see [Provide a Remote File System for the FME Flow System Share](#).

 **Tip** As a best practice, specify the full UNC path to the FME Flow System Share, even if it is on the same (local) machine where you are running the installer.

 **Note** If you are [upgrading](#) to a newer version of FME Flow, make sure the specified directory is empty of files from the previous installation. Otherwise, conflicts may occur that can produce unexpected results in performance.

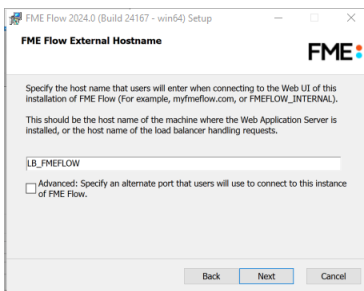


FME Flow External Hostname

If this FME Flow installation will be available to users outside an internal network (for example, on an external website), specify the external hostname that users will enter when connecting to FME Flow. For example, if you are hosting FME Flow on `myfmeflow.com/myfmeflow`, enter `myfmeflow.com`.

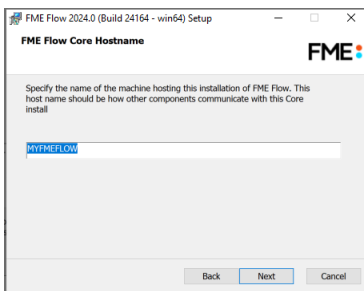
Otherwise, enter the internal server hostname, IP Address, or DNS.

If a load balancer or reverse proxy is deployed through which to access FME Flow, check *Advanced* and provide the *External Web Traffic Port*. Specify this port only if it differs from the [Web Application Server Port](#) (below).




FME Flow Core Hostname

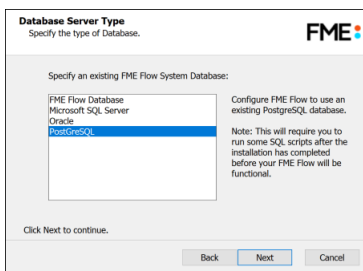
Specify the name of the machine hosting this installation of FME Flow.



Database Server Type

 **Note** This dialog does not appear if you elected to install the FME Flow Database in step [Custom Setup](#) (above).

Specify the type of database for the FME Flow Database. If you installed, on a separate machine, the PostgreSQL database included with the FME Flow installer, select **FME Flow Database**. If you are configuring the FME Flow Database on your own database server, select the database server type: **Microsoft SQL Server**, **Oracle**, or **PostgreSQL**. For more information, see [Provide a Database Server](#).

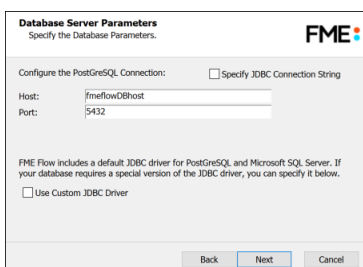


Database Server Parameters

Specify the connection parameters for the FME Flow Database. Optionally, to customize the connection parameters in the JDBC connection string, check **Specify JDBC Connection String**.


If **Microsoft SQL Server** was specified in the previous dialog, check *Use SQL Server instance name* to use that value instead of *Port*.

FME Flow includes a default JDBC driver for PostgreSQL and Microsoft SQL Server. If your database requires a special version of the JDBC driver, check *Use Custom JDBC Driver* and specify the *JDBC Driver*. If **Oracle** was specified in the previous dialog, you must specify the *JDBC Driver*.



If you elected to install the **FME Flow Database** in step [Custom Setup](#) (above), a Database Parameters dialog appears instead:


- **Install the FME Flow Database files to:** By default, [FME Flow Database](#) files are written to %ALLUSERSPROFILE%\Safe Software\FME Flow\pgsql\data.

 **Note** Unless modified, C:\ProgramData is the default value of the %ALLUSERSPROFILE% environment variable.

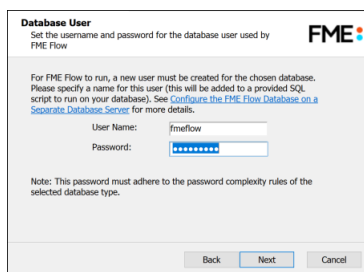
To change the installation directory of FME Flow Database files, modify the path.

Database User

The installation creates a user account under which to use the FME Flow database. Specify a *User Name* (defaults to **fmeflow**) and *Password* for the database user account.

 **Note** The password for the FME Flow Database user account is encrypted in FME Flow installation files. Take note of the *User Name* and *Password* that you specified. You may need to reference it later.

If you installed the PostgreSQL database included with the FME Flow installer, specify the same *User Name* and *Password* that you specified for Database User previously, under [Install the FME Flow Database](#).

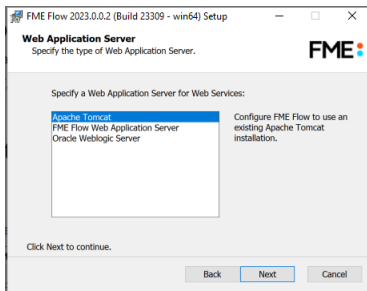


Web Application Server

Choose a Web Application Server:

Apache Tomcat *(Expand for instructions)*

This selection is valid if you have already completed the step to [Provide a Web Application Server](#).

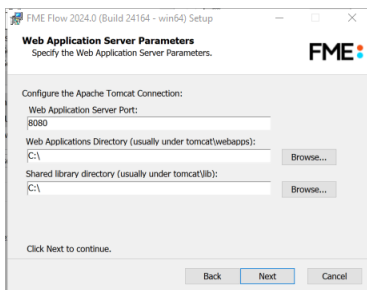


Web Application Server Parameters

Specify the connection parameters for the web application server.

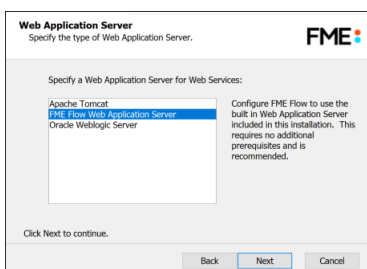
The *Web Application Server Port* should be the same as the one that was specified when Apache Tomcat was installed. The default Apache Tomcat port is 8080.

The *Web Applications Directory* and *Shared library directory* can be found in the webapps and lib folder, respectively, under `<ApacheTomcatInstallDir>/tomcat/`.



FME Flow Web Application Server *(Expand for instructions)*.

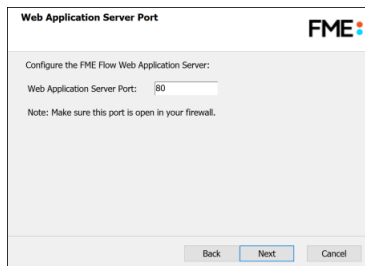
This selection is valid if you want to use the Apache Tomcat web application server provided with the installer, and have *not* provided your own under [Provide a Web Application Server](#).



Web Application Server Port

Specify the port to use for communication with the web application server. On Windows, port 80 is the recommended default. On Linux, port 8080 is recommended.

Note Common applications, including Skype and Internet Information Services (IIS), may already be using port 80. To free this port, turn off these services. Alternatively, select a different port, such as 8080.



Follow the remaining dialogs to complete the installation.


What's Next?

If you are configuring the FME Flow Database on your own database server, and did *not* install the PostgreSQL database that is included with the FME Flow installer, proceed to [Configure the FME Flow Database on a Separate Database Server](#). Otherwise, proceed to [Log on to the Web User Interface](#).


Configure the FME Flow Database on a Separate Database Server

Note Complete this step *only* if you are configuring the FME Flow Database on your own database server, and did *not* install the PostgreSQL database that is included with the FME Flow installer. For more information, see [Provide a Database Server](#).

To configure FME Flow to use a separate database server, follow the steps below. You can configure FME Flow with a PostgreSQL (recommended), Microsoft SQL Server, or Oracle database, running on either Windows or Linux.

 **Note** PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.

1. Database Configuration: Set up FME Flow database tables and database users with permissions to access the FME Flow database.

 **Note** The specific way to set up database tables and user permissions on various production databases may differ depending on the available database tools and intended target platform.

Database Configuration - PostgreSQL

PostgreSQL is an open source database that can be downloaded from the Internet. For more information, see the PostgreSQL web site at:

<https://www.postgresql.com>

It's assumed that PostgreSQL is already installed .

One notation used is <FMEFlowDir>, which is the installation directory of FME Flow. In Windows, this is typically `C:\Program Files\FMEFlow`.

It's also assumed that you have JRE 5 or higher and PostgreSQL database installed before you proceed further. If you want to use the JDBC driver, there are several steps to perform.

If you are upgrading, you should back up any job history you want to keep.

Windows

This section describes how to configure a PostgreSQL database. You can configure this database from other PostgreSQL database tools, depending on your personal preference.

For a PostgreSQL database configuration, various SQL scripts are provided to assist you. These SQL scripts are located in the following directory:

`<FMEFlowDir>\Server\database\postgresql`

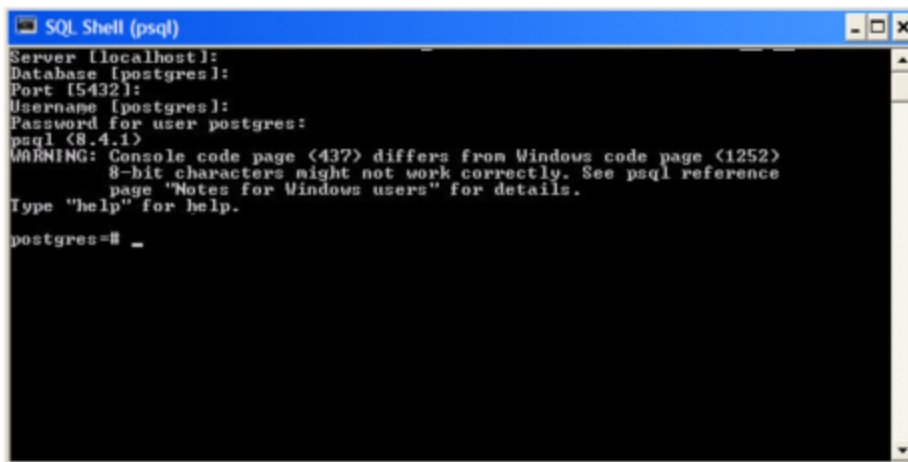
Note This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `postgresql_createUser.sql` – This script creates the FME Flow database user and password that you specified during installation, and grants all required permissions to this user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- `postgresql_createDB.sql` – This script creates the FME Flow database.
- `postgresql_createSchema.sql` – This script creates all FME Flow related tables, indexes, views, and triggers.
- `postgresql_dropUser.sql` – This script drops the FME Flow database user.
- `postgresql_dropDB.sql` – This script drops the FME Flow database.

Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install PostgreSQL SQL shell (`psql`) if you haven't done so already.
2. Open `psql` and log on as a user with the appropriate privileges:



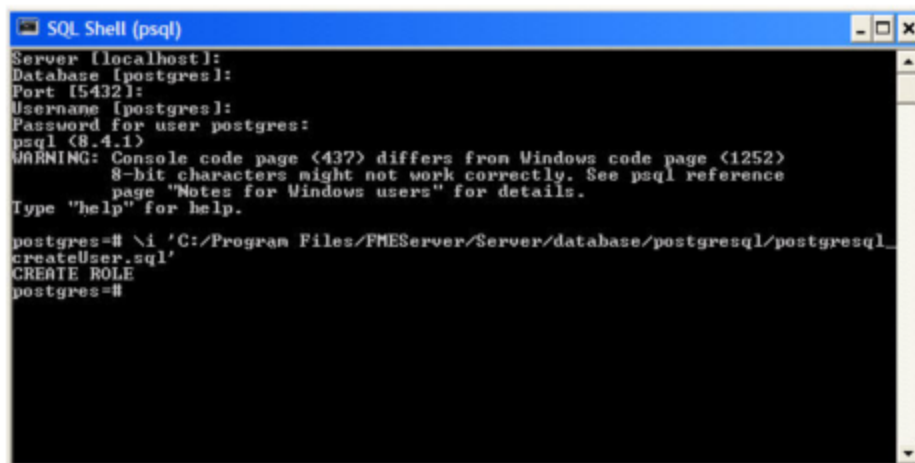
```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.
postgres=# _
```

3. Create an FME Flow database user as follows:

- a. Open `postgresql_createUser.sql` in a text editor, and replace `<<DATABASE_PASSWORD>>` with the password specified during the FME Flow installation.
- b. From the SQL prompt, run the `postgresql_createUser.sql` script by entering the following command:

```
\i '<FMEFlowDir>/Server/database/postgresql/postgresql_createUser.sql'
```

By default, the provided SQL script creates the FME Flow database user and password that you specified during installation:



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql <8.4.1>
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

postgres=# \i 'C:/Program Files/FMEServer/Server/database/postgresql/postgresql_createUser.sql'
CREATE ROLE
postgres=#
```

4. Create an FME Flow database.

From the SQL prompt, run the `postgresql_createDB.sql` script by entering the following command:

```
\i '<FMEFlowDir>/Server/database/postgresql/postgresql_createDB.sql'
```

By default, the provided SQL script creates an FME Flow database named `fmeFlow` and grants all privileges on the database to the user.

```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

postgres=# \i 'C:/Program Files/FME Server/Server/database/postgresql/postgresql_
createUser.sql'
CREATE ROLE
postgres=# \i 'C:/Program Files/FME Server/Server/database/postgresql/postgresql_
createDB.sql'
CREATE DATABASE
GRANT
postgres=# _
```

5. Quit `psql` and log on as the FME Flow database user.

```
SQL Shell (psql)
Server [localhost]:
Database [postgres]: fmeserver
Port [5432]:
Username [postgres]: fmeserver
Password for user fmeserver:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

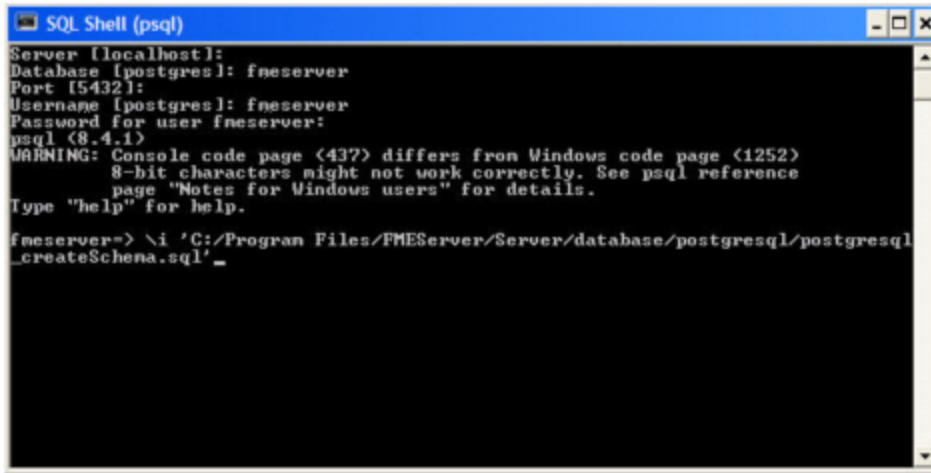
fmeserver=> _
```

6. Create the FME Flow database schema as follows:

From the SQL prompt, run the `postgresql_createSchema.sql` script by entering the following command:

```
\i '<FMEFlowDir>/Server/database/postgresql/postgresql_
createSchema.sql'
```

By default, the provided SQL script creates all FME Flow related tables, indexes, views, and triggers.



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]: fmeserver
Port [5432]:
Username [postgres]: fmeserver
Password for user fmeserver:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

fmeserver-> \i 'C:/Program Files/FMEServer/Server/database/postgresql/postgresql
_createSchema.sql'
```


Linux

This section describes configuring PostgreSQL from both the command prompt and a user interface, such as the PostgreSQL Query Browser and the PostgreSQL Administrator.


For PostgreSQL configurations, various SQL scripts are provided to help with PostgreSQL database configuration. These SQL scripts are located in the `<FMEFlowDir>/Server/database/postgresql/` directory of the FME Flow Core machine.

Note This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `postgresql_createUser.sql` – This script creates the FME Flow database user and password that you specified during installation, and grants all required permissions to the FME Flow database user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.

 **Note** The password is encrypted and appears in the script as <<DATABASE_PASSWORD>>.

- `postgresql_createDB.sql` – This script drops the FME Flow database and creates a new one with the required FME Flow tables. By default this database is named `fmeflow`. Be careful when running this script because it drops the existing FME Flow database.
- `postgresql_createSchema.sql` – This script creates all FME Flow related tables, indexes, views, and triggers.

 **Note** This script must be run as user `fmeflow`.

- `postgresql_dropDB.sql` – This script drops the existing FME Flow database, which is named `fmeflow` by default.
- `postgresql_dropUser.sql` – This script drops the FME Flow database user.

Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install PostgreSQL Server if you haven't already.
2. Open a command prompt and change to the following directory, which contains the PostgreSQL SQL scripts:

```
<FMEFlowDir>/Server/database/postgresql/
```

3. Open `postgresql_createUser.sql` in a text editor, and replace <<DATABASE_PASSWORD>> with the password specified during the FME Flow installation.


4. Run the `postgresql` program with the `postgresql_createUser.sql` script:

```
psql -d postgres -h <hostname> -p <port> -f postgresql_createUser.sql
```

5. Run the postgresql program with the postgresql_createDB.sql script:

```
psql -d postgres -h <hostname> -p <port> -f postgresql_createDB.sql
```
6. Run the postgresql program with the postgresql_createSchema.sql script, as the user you specified during installation:

```
psql -U <username> -d fmeFlow -h <hostname> -p <port> -f postgresql_createSchema.sql
```

 **Note** If you drop the FME Flow database and login user, run the postgresql program with the appropriate postgresql_dropDB.sql or postgresql_dropUser.sql scripts.

After Database Creation

1. On the machine on which the FME Flow database server is installed, open pg_hba.conf. For default installation directories of this file:
 - PostgreSQL 10: `<PostgreSQLDir>\data\pg10\pg_hba.conf`
 - PostgreSQL 11: `C:\Program Files\PostgreSQL\11\data\pg_hba.conf`
2. Change the following lines:

```
host    all    all    127.0.0.1/32    md5
host    all    all    ::1/128         md5
```

to:

```
host    all    all    0.0.0.0/0      md5
host    all    all    ::/0           md5
```


3. [Restart](#) the FME Flow Database service.

Database Configuration - Microsoft SQL Server

It is assumed that SQL Server is installed.

One notation used is `<FMEFlowDir>`, which is the installation directory of FME Flow. In Windows, this is typically `C:\Program Files\FMEFlow`.

SQL scripts are provided to help with SQL Server database configuration. These SQL scripts are located in the `<FMEFlowDir>\Server\database\sqlserver\` directory of the FME Flow Core machine.

 **Note** This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

If you are upgrading, you should back up any job history you want to keep.

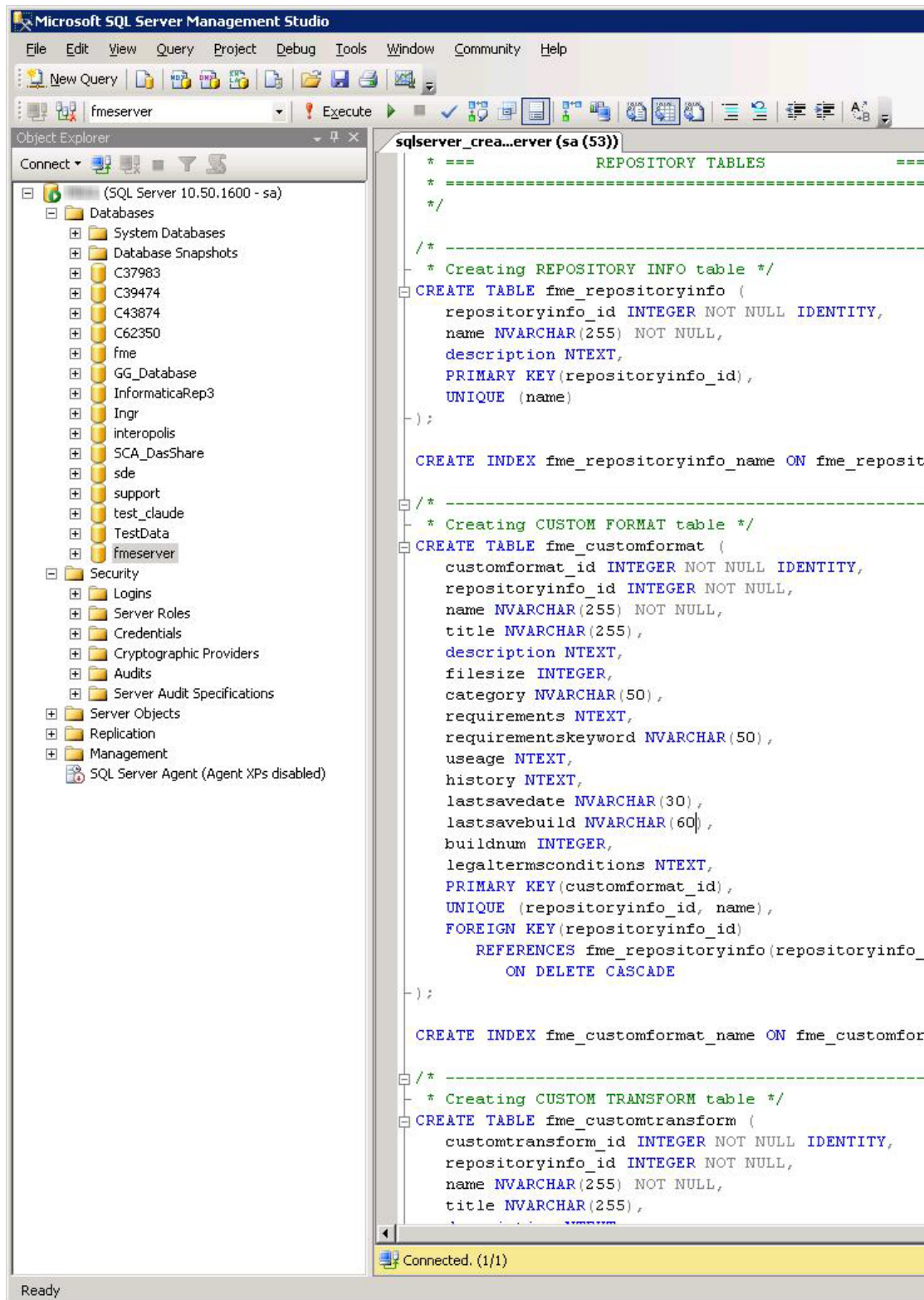
Prerequisite for Unsecured Database Servers

If your SQL Server database server is not secured with a valid SSL certificate, you must update the [fmeDatabaseConfig.txt](#) configuration file as follows:

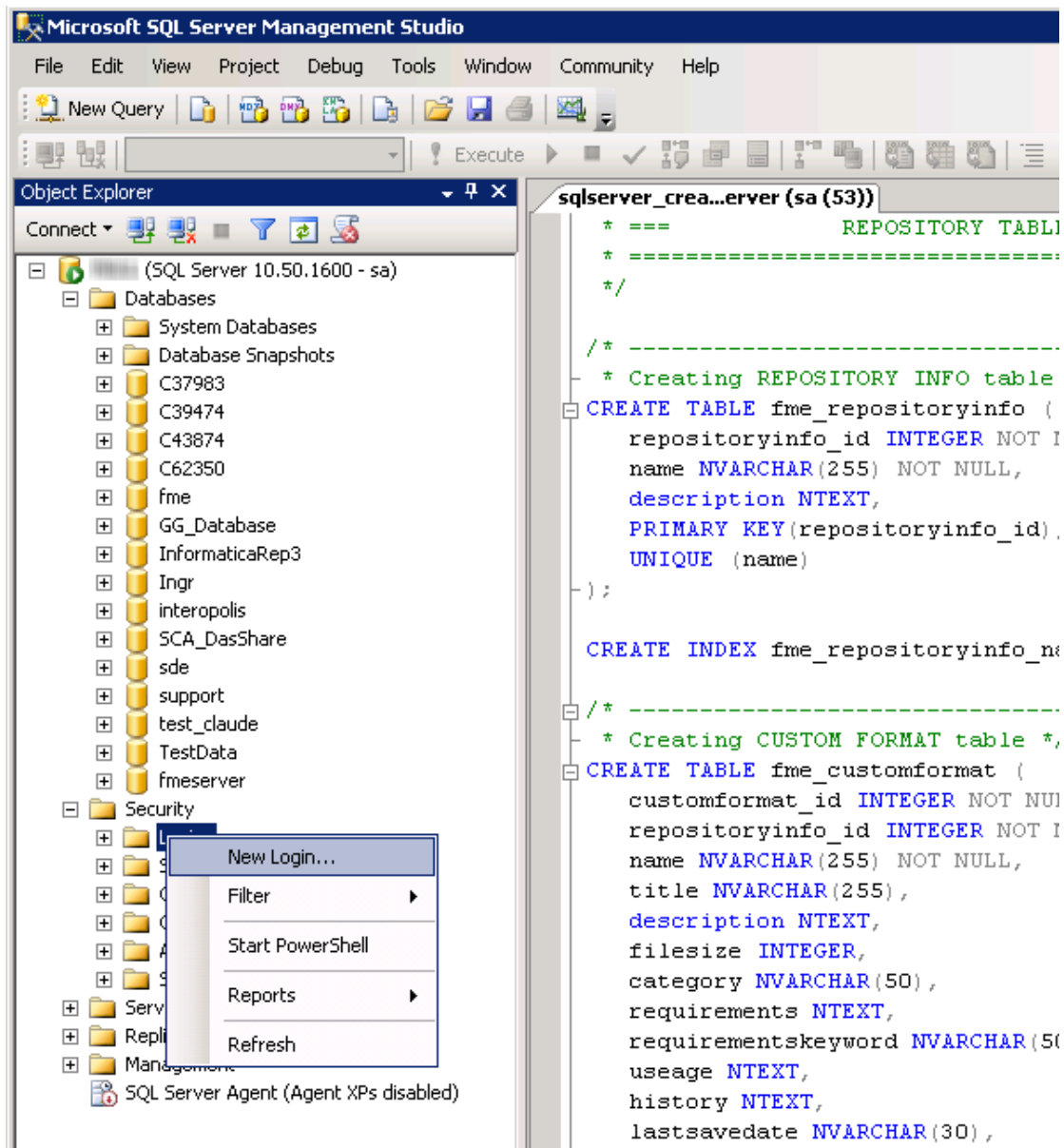
1. Near the bottom, under the heading `FME SERVER SETTINGS START`, locate the block of code beginning with `DB_TYPE=sqlserver`.
2. In the corresponding line for the `DB_JDBC_URL` parameter, update the `encrypt` value from `true` to `false`.
3. Save and close the file.

Instructions Using SQL Server Management Studio

1. Open the SQL Server Management Studio and connect to the SQL Server database engine.
2. Open `sqlserver_createDB.sql` and run the script against the `fmeFlow` database, as shown below:



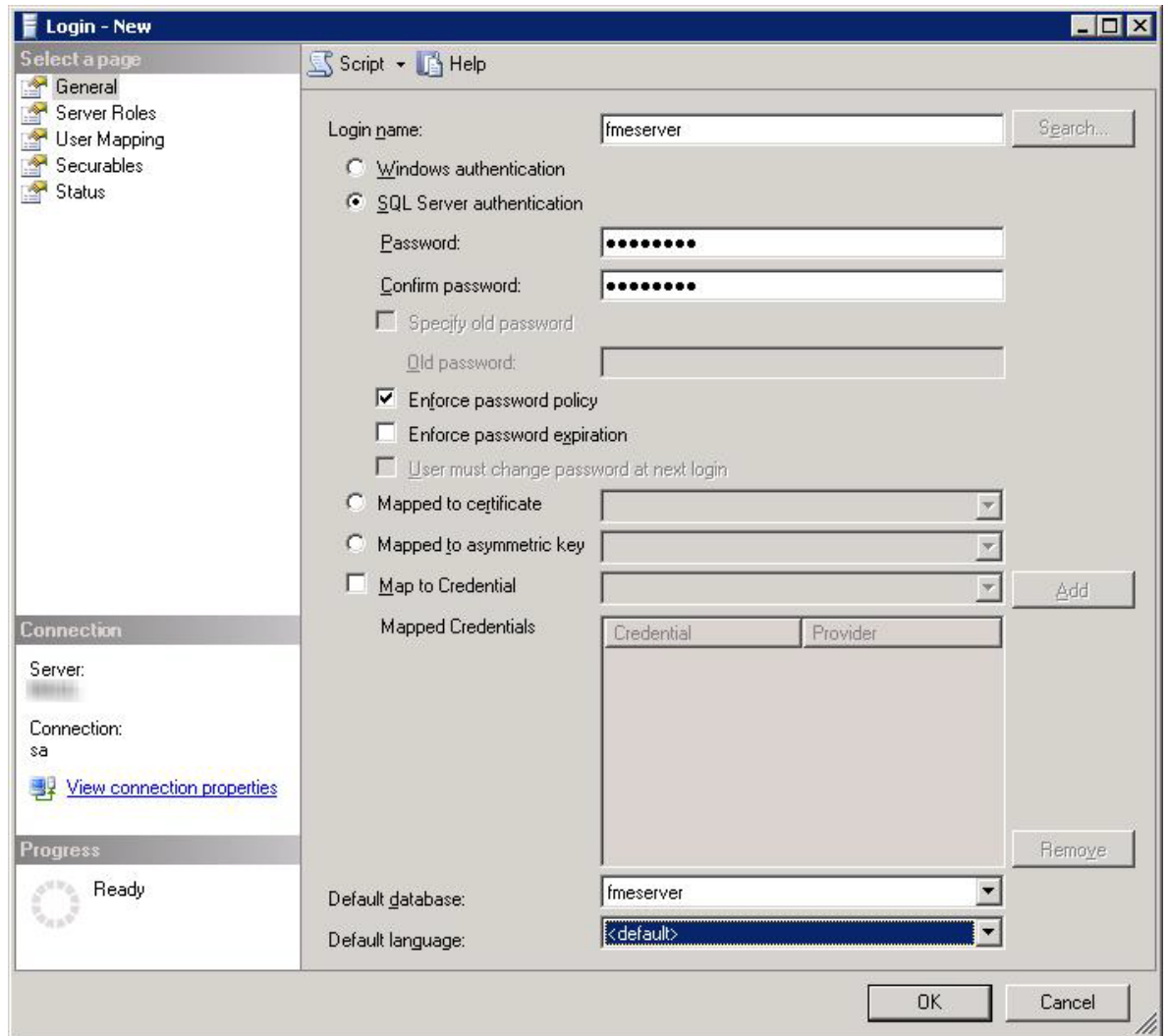
3. On the Object Explorer pane, expand the Security folder, right-click Login, and then select New Login:



4. On the Login – New page, click General to open the page shown in the example below.

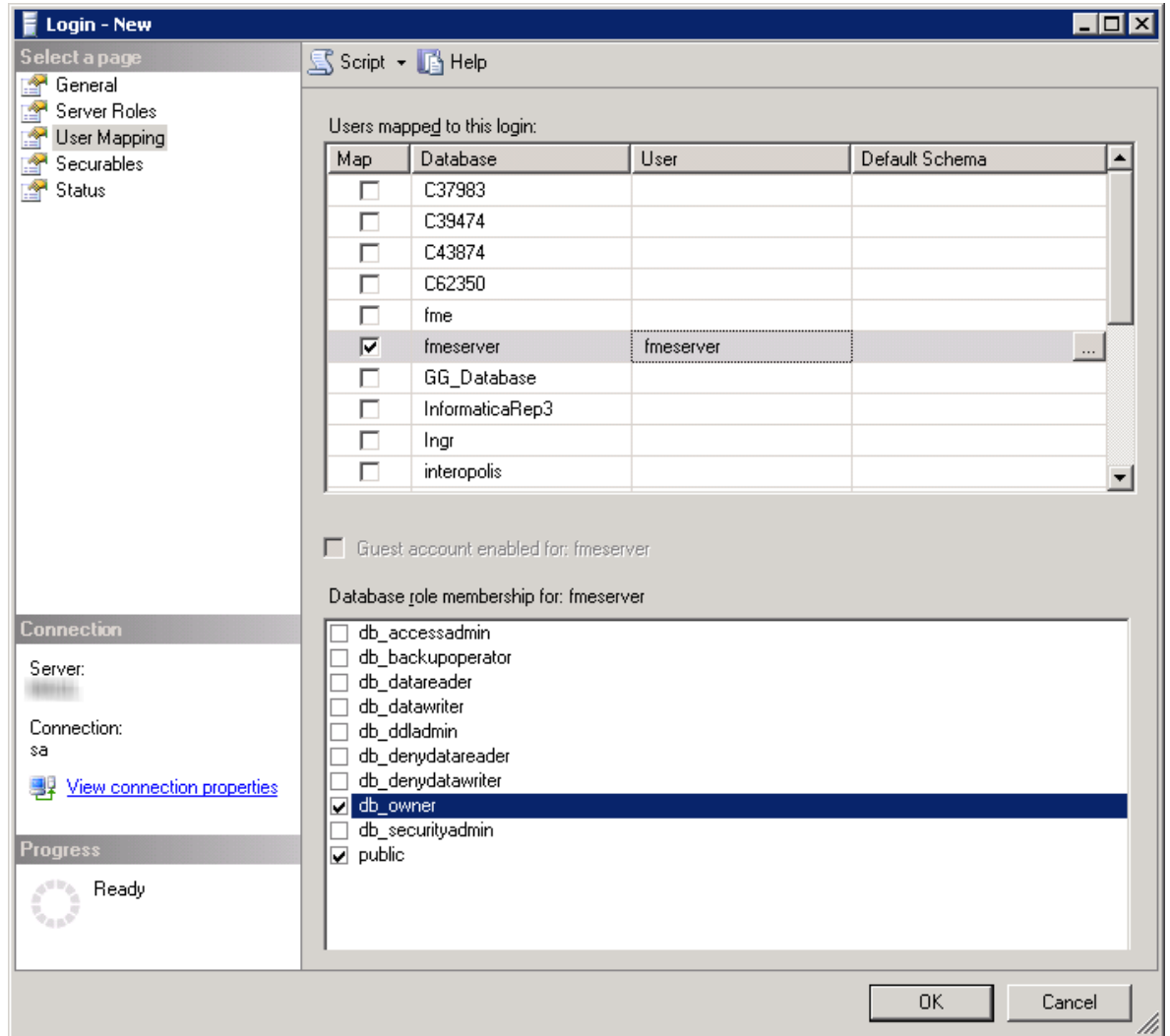
- At the Login name field, enter **fmeFlow**.
- Make sure the option button is active for SQL Server authentication.
- In the Password field enter the password that was specified for the database during the FME Flow installation, and in the Confirm password field enter the same password again.

- Make sure the Enforce password policy is active. If not, click the checkbox to activate it.
- If required, click the checkbox for Enforce password expiration to clear it.
- At the Default database field enter fmeserver.



5. On the same Login – New page, click User Mapping to open the page shown in the next figure.
 - At the Users mapped to this login location, in the Map column click the checkbox to activate it and map the login name to the **fmeserver** database.

- At the Database role membership for location, notice that the fme\flow database is specified and check `db_owner` to assign that role to this database.
- Click OK to close this page.




Instructions Without SQL Server Management Studio

1. Connect to the SQL Server database engine and run `sqlserver_createDB.sql`.
2. Open `sqlserver_createUser.sql`.
3. Look for the line that begins:

```
CREATE LOGIN fme\flow WITH PASSWORD =
```

Replace <<DATABASE_PASSWORD>> with the password specified during the installation.

4. Run the script.

 **Note** If your SQL Server uses an Always On availability group, you must also configure the `MultisubnetFailover` property of the `DB_JDBC_URL` parameter in the `fmeDatabaseConfig.txt` configuration file. For more information, see the SQL Server examples under [Configure the Database Connection](#).

Database Configuration - Oracle

Before proceeding, you must provide the following:

- Install an Oracle database server.
- Obtain the Oracle Database JDBC driver. For more information, see <https://www.oracle.com/technetwork/database/application-development/jdbc/downloads/index.html>.

<FMEFlowDir> is the installation directory of FME Flow. In Windows, this is typically `C:\Program Files\FMEFlow`.


If you are upgrading, you should back up any job history you want to keep.

Windows

This section describes how to configure an Oracle database. You can configure the Oracle database from other Oracle database tools depending on your personal preference.

For an Oracle database configuration, various SQL scripts are provided to help.

These SQL scripts are located in the <FMEFlowDir>\Server\database\oracle directory of the FME Flow Core machine.

 **Note** This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `oracle_createUser.sql` – This script creates the FME Flow database user, and grants all required permissions to the FME Flow database user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- `oracle_createDB.sql` – This script creates all FME Flow related database packages, tables, indexes, views, and triggers.
- `oracle_dropUser.sql` – This script drops the FME Flow database user and the database packages, tables, indexes, views, and triggers associated with the FME Flow database user.

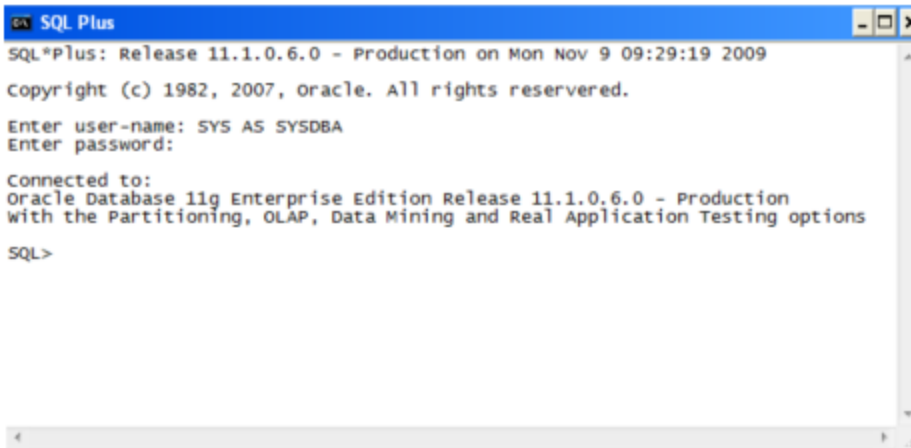
Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install Oracle SQL*Plus, if you haven't already.
2. Open Oracle SQL*Plus and log on as a user with adequate privileges, such as SYSDBA.

In this example, the following parameters were used and entered:

- User: SYS AS SYSDBA
- Password: fmefflow
- String: XX-ORACLE11G:1521/ORCL AS SYSDBA

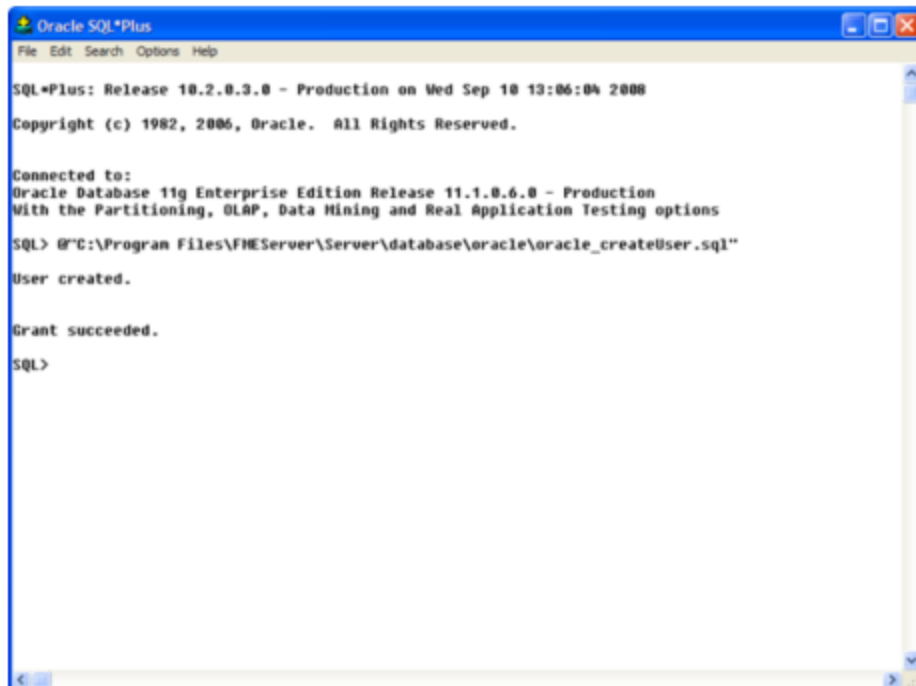


```
SQL Plus
SQL*Plus: Release 11.1.0.6.0 - Production on Mon Nov 9 09:29:19 2009
copyright (c) 1982, 2007, oracle. All rights reserved.
Enter user-name: SYS AS SYSDBA
Enter password:
Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
SQL>
```

3. Create an FME Flow database user as follows:

- a. In `oracle_createUser.sql`, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- b. From the SQL prompt, run the `oracle_createUser.sql` script by entering the following command:

`@<FMEFlowDir>\Server\database\oracle\oracle_createUser.sql`



```
Oracle SQL*Plus
File Edit Search Options Help
SQL*Plus: Release 10.2.0.3.0 - Production on Wed Sep 10 13:06:04 2008
Copyright (c) 1982, 2006, Oracle. All Rights Reserved.
Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
SQL> @C:\Program Files\FMEServer\Server\database\oracle\oracle_createUser.sql
User created.
Grant succeeded.
SQL>
```

The default TABLESPACE is the Oracle DB configured default. If the created user doesn't have read/write permissions on the default TABLESPACE, change the default TABLESPACE or create an appropriate TABLESPACE.

This command creates a TABLESPACE on the Oracle DB server called FMEFLOW shown in the following example:

```
CREATE SMALLFILE TABLESPACE FMEFLOW DATAFILE  
'E:\ORADATA\FMEFLOW01.DBF' SIZE 100000K REUSE AUTOEXTEND ON  
NEXT 2048K MAXSIZE 1024M LOGGING EXTENT MANAGEMENT LOCAL  
SEGMENT SPACE MANAGEMENT AUTO DEFAULT NOCOMPRESS;
```

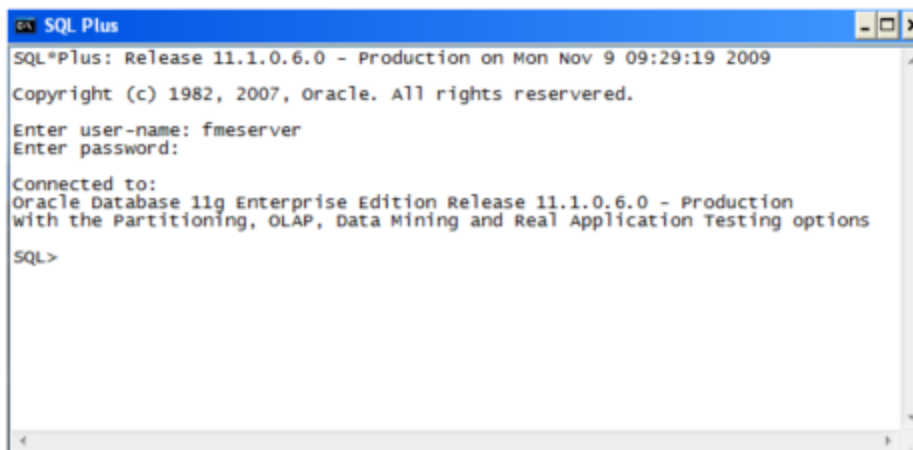
By default this command uses the TABLESPACE "FMEFLOW" as shown in the following example:

```
CREATE USER fme\flow IDENTIFIED BY fme\flow DEFAULT TABLESPACE  
"FMEFLOW" TEMPORARY TABLESPACE "TEMP";
```

4. When an FME Flow database user is created, you can connect to it by logging on again to SQL*Plus.

In this example, the following parameters were used and entered on the user interface:

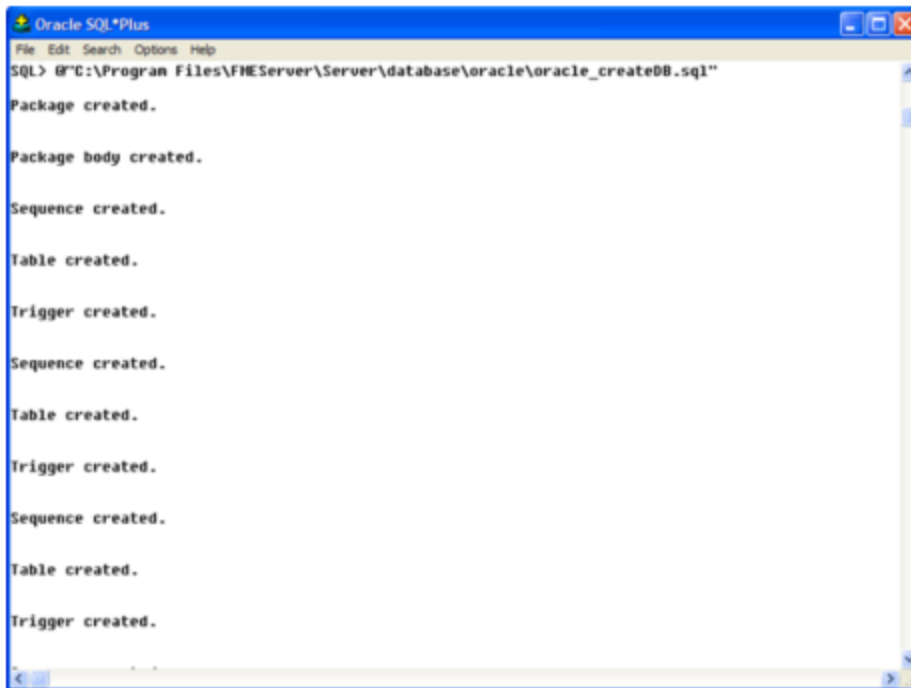
- User: fme\flow
- Password: fme\flow



Note The preceding example assumes your user name and password are fmeflow.

- When connected to the FME Flow database, run the `oracle_createDB.sql` script by entering the following command:

```
@ "<FMEFlowDir>\Server\database\oracle\oracle_createDB.sql"
```



Note If you drop the FME Flow database and login user, run the `oracle_dropUser.sql` script.

Linux

Oracle SQL*Plus is a freeware client for connecting to an Oracle database. You can download it from the Internet using the following address:


https://www.oracle.com/technology/software/tech/sql_plus/index.html

For more information, see the Oracle web site at:

https://www.oracle.com/technology/sample_code/tech/sql_plus/htdocs/demobld.html

The instructions that follow describe how to configure Oracle for use with FME Flow. These instructions describe using SQL*Plus for the database setup. Depending on your personal preference, you can also configure the Oracle database using other Oracle tools.

For Oracle configurations, various SQL scripts are provided to help with Oracle database configuration. These SQL scripts are located in the `<FMEFlowDir>/Server/database oracle/` directory of the FME Flow Core machine.

 **Note** This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `oracle_createUser.sql` – This script creates the FME Flow database user, and grants all required permissions to the FME Flow database user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- `oracle_createDB.sql` – This script creates all FME Flow related database packages, tables, indexes, views, and triggers.
- `oracle_dropUser.sql` – This script drops the FME Flow database user and the database packages, tables, indexes, views, and triggers associated with the FME Flow database user.

Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install the Oracle Client Tools, which includes SQL*Plus, if you haven't already.
2. Run `sqlplus` and log on as a user with adequate privileges, such as SYSDBA.

For example:

```
sqlplus SYS/<password>@<Host>:<Port>/<service> AS SYSDBA
```

3. Create the FME Flow database user:
 - a. In a text editor, open the `oracle_creatorUser.sql` script and update the password `<<DATABASE_PASSWORD>>` to the one specified during the FME Flow installation.
 - b. From the SQL prompt, run the `oracle_createUser.sql` script by entering the following command:

```
@<FMEFlowDir>/Server/database/oracle/oracle_createUser.sql"
```

The default TABLESPACE is the Oracle DB configured default. If the created user doesn't have read/write permissions on the default TABLESPACE, you need to change the default TABLESPACE or create an appropriate TABLESPACE.

This command creates a TABLESPACE on the Oracle DB server called FMEFLOW shown in the following example:

```
CREATE SMALLFILE TABLESPACE FMEFLOW DATAFILE  
'/ORADATA/FMEFLOW01.DBF' SIZE 100000K REUSE AUTOEXTEND ON  
NEXT 2048K MAXSIZE 1024M LOGGING EXTENT MANAGEMENT LOCAL  
SEGMENT SPACE MANAGEMENT AUTO DEFAULT NOCOMPRESS;
```

By default this command uses the TABLESPACE "FMEFLOW" shown in this example:

```
CREATE USER fmeFlow IDENTIFIED BY fmeFlow DEFAULT TABLESPACE  
"FMEFLOW" TEMPORARY TABLESPACE "TEMP";
```

4. When an FME Flow database user has been created, you can connect to it by logging on again to SQL*Plus or by entering the following command:

```
CONNECT <UserName>/<Password>@<Host>:<Port>/<service>;
```

5. When connected to the FME Flow database, run the oracle_createDB.sql script by entering the following command:

```
@"<FMEFlowDir>/Server/database/oracle/oracle_createDB.sql"
```

2. **Enable connections:** FME Flow connects to the database when it starts. Therefore, ensure that your database server is running and configured to accept incoming connections before FME Flow is started. The database must allow connections over TCP/IP with all machines on which the FME Flow Core and FME Engines are installed.
3. Ensure that all installations of FME Flow are running. For more information, see [Working with the FME Flow System Services](#).

What's Next?

- [Log on to the Web User Interface](#)

Log on to the Web User Interface

To log in to the Web User Interface for the first time as an administrator following installation, specify *Username* `admin` and *Password* `admin`. You will be prompted to update the `admin` password upon initial use, based on the default [Password Policy](#) configuration.

For more information and support:

- [FME Flow Troubleshooting: Web Interface Login](#)
- [Accessing the Web User Interface](#)
- [Changing the Login Password](#)
- [Default User Accounts and Passwords](#)
- [Role-Based and User-Based Access Control](#)

What's Next?

- [Request and Install a License](#)

Request and Install a License

Note

License files are installed, by default, to:

» Windows: `C:\ProgramData\Safe Software\FME Flow\licenses`

» Linux: `/opt/fmeflow/licenses`

Dynamic Engine (CPU Usage) licensing is not available for engines on FME Flow Hosted.

Automatic Mode (Recommended)


If your FME Flow host machine has an internet connection, you can request and install a license online without any need for downloading and re-uploading a license file.

Note


- Licensing FME Flow over the internet requires no inbound connection. For more information, see [FME Flow Automatic Licensing URLs](#).
- If you purchased credits for Dynamic Engines (CPU Usage), automatic mode only is supported. Manual mode licensing (below) is not supported.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.

3. Complete the fields on the Request a New License page.


 **Note** If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.

4. *Licensing Mode*: Select **Automatic**.
5. Click **OK**.
6. A message indicates that your license is installed, and your available engines appear momentarily on the [Engines](#) page.

 **Note** If the *Serial Number* you provided includes Dynamic Engines (CPU Usage), they are not configured to start by default. You can start and configure your Dynamic Engines on the [Engines](#) page.


Manual Mode

If you prefer not to use an internet connection for licensing, or if your FME Flow host machine does not have an internet connection, use this process to install a license.

 **Note** If you purchased credits for Dynamic Engines (CPU Usage), manual mode licensing is not supported. Use automatic mode for licensing (above). For more information, contact your Safe Software Inc. sales representative.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Flow engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.

3. Complete the fields on the Request a New License page.

 **Note** If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.

4. *Licensing Mode*: Select **Manual**.

5. Click **OK**. A license request .json file downloads.

6. Email the .json file to codes@safe.com.

7. You will receive an email from Safe Software Inc. with a license file, which has a `.fmeLIC` extension. Download the file.

8. Return to the Licensing page. Under Standard Engines, click **Upload License File** and select the `.fmeLIC` file to upload. Or, drag and drop the `.fmeLIC` file onto the page under Standard Engines. A message indicates that it is installed, and your available engines appear on the [Engines](#) page.

Problems?

- [FME Flow Troubleshooting: FME Flow Engines](#)
- Contact codes@safe.com.

See Also

- [FME Flow Automatic Licensing URLs](#)
- [FME Engines](#)

What's Next?

- (Recommended) [Update the Windows service accounts](#) that run the FME System Services.
- (Upgrade Only) [Restore](#) your FME Flow Configuration from a Previous Installation.
- [Test the Installation](#).

Test the Installation

To test your FME Flow installation, confirm that FME Flow can perform its primary function—running a job.

1. On the [Run Workspace](#) page, specify:
 - Repository: Samples
 - Workspace: austinApartments.fmw
 - Service: Job Submitter
2. Click Run Workspace.

If you see a success message, your install was a success.

See Also

- [FME Flow Troubleshooting: Submitting a Job in FME Flow](#)


Distributing Components (3-Tier)

Before installing a 3-tier distributed architecture, make sure you have:


- Read and understood [Planning an FME Flow Installation](#).
- Provided the prerequisites in [Before Installation](#).
- Linux: Ensured [System Requirements](#) are met, including updates and dependencies.

To perform a 3-tier installation of FME Flow:

1. [Obtain the Installer](#)
2. [Install the FME Flow Database](#)

 **Note** Complete this step *only* if you wish to install a PostgreSQL database that is included with the FME Flow installer, and do *not* wish to configure the FME Flow Database on your own database server. For more information, see [Provide a Database Server](#).

3. [Install the FME Flow Core and FME Engines](#)
4. [Install the FME Flow Web Services](#)
5. [Configure the FME Flow Database](#)

 **Note** Complete this step *only* if you are configuring the FME Flow Database on your own database server, and did *not* install the PostgreSQL database that is included with the FME Flow installer. For more information, see [Provide a Database Server](#).

6. [Start the FME Flow system services](#). Following a distributed installation, the [FME Flow system services](#) do not start automatically. You must manually start them on each machine that is hosting FME Flow components.
7. [Log on to the Web User Interface](#)
8. [Update the WebSocket Host](#)
9. [Request and Install a License](#)
10. (If necessary) [Update the Windows service accounts](#) that run the FME System Services.
11. (Upgrade only) [Restore](#) your FME Flow Configuration from a Previous Installation.
12. [Test the Installation](#).

What's Next?

Optimize: You can scale-up your FME Flow by:

- Increasing the number of FME Engines that run when you start FME Flow.
- Adding or accessing FME Engines on separate machines that are closer to your data.

For more information, see [Planning for Scalability and Performance](#).

Before Installation

Before you install the FME Flow components for a 3-tier architecture, ensure the following non-FME Flow components are in place on your network:

- [Database server](#) on a separate machine.
- [Web application server](#) on a separate machine; or, if using the Web Application Server provided with FME Flow, a machine on which to install it.
- [File system](#) on a separate machine.
- (Recommended) One or more Windows service accounts with read and write permissions on the network shares of the FME Flow Core/Engines and the Web Application Server. In most distributed installations, this account is necessary for these components to communicate with each other across a network. For more information, see [Running the FME Flow System Services Under Different Accounts \(Windows\)](#).

Next Steps

Proceed to [Obtain the Installer](#).


Provide a Database Server

FME Flow uses its own [database](#) to manage jobs and workspace information. It is not a source or destination data source for FME workspaces.


To ensure the FME Flow Database remains available in a failover scenario, provide a machine on which to install the database that is physically separate from the machines on which the FME Flows are installed. In addition to being physically separate, the database server should, by itself, be configured for fault tolerance.

You can choose to install a PostgreSQL database that is included with the FME Flow installer, or you can configure the FME Flow Database on your own database server. If you use your own database server, keep in mind the following:

- PostgreSQL (recommended), Oracle, and SQL Server are supported. The following versions are recommended:

 **Note** PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.

- PostgreSQL: 16.1 or later.
 - Oracle: 19c or later.
 - SQL Server: 2016 or later.
- When you install the FME Flows, the database server must be running and you must know the connection information.
 - If using an Oracle database server, you must obtain the Oracle Database JDBC driver. For more information, see <https://www.oracle.com/technetwork/database/application-development/jdbc/downloads/index.html>.

 **Note** Oracle Database JDBC driver versions that use ojdbc10.jar are *not* supported.

- Following installation, you must be able to create a new database on the server using SQL scripts.

For information about acquiring a fault-tolerant database server, consult your IT department.

Updating the FME Flow Database Password

If you choose to install a PostgreSQL database that is included with the FME Flow installer, and want to update the password for the database user account, follow these instructions:

1. Start pgAdmin4.exe from <FMEFlowDir>\Utilities\pgsql\pgAdmin 4\bin\.
2. If this is your first time starting pgAdmin4, it will prompt you to set a master password. Choose something you will remember.
3. If pgAdmin auto-connects to the FME Flow Database, an entry appears in the Object browser, under Servers, similar to the following:

`fmeserver (<hostname>:7082)`

If pgAdmin III does not auto-connect to the FME Flow Database, add a new Server for it. To add a New Server:

- a. Right-click on **Servers** and Select **Create > Server....**
- b. In the **General** tab of the Create - Server dialog, specify the *Name* for the server connection. This can be anything, such as `fmeFlow`.
- c. In the **Connection** tab of the Create - Server dialog, specify the server properties, using the `Database Connection` details in the [fmeDatabaseConfig](#) configuration

file, for `DB_TYPE=postgresql`, as follows:

Property	Value from fmeDatabaseConfig.txt
Host name/address	DB_JDBC_ URL=jdbc:postgresql://<hostname>:<port>/<database_name>
Port	DB_JDBC_ URL=jdbc:postgresql://<hostname>:<port>/<database_name>
Maintenance database	DB_JDBC_ URL=jdbc:postgresql://<hostname>:<port>/<database_name>
Username	DB_USERNAME=<username>
Password	DB_PASSWORD=<password>

d. Click **Save**.

- In the Object browser, select the FME Flow Database server, and locate and expand **Login/Group Roles**.
- Right-click the fmeserver database user account and select **Properties**. Under **Definition**, change the *Password* and click **Save**.
- In the [fmeDatabaseConfig](#) configuration file, update the value of the `DB_PASSWORD` parameter that corresponds to `DB_TYPE=postgresql` with the new password specified above.

When finished, save the file.

- (Recommended) [Encrypt](#) the FME Flow Database password.
- [Restart](#) FME Flow.

1. Open a command prompt and change to the following directory:

```
<FMEFlowDir>/Utilities/pgsql/bin
```

2. Run the following command to connect to the FME Flow Database:

```
./psql -d <databaseName> -p <port> -U <username>
```

To confirm the settings, check the **Database Connection** details in the [fmeDatabaseConfig](#) configuration file, for **DB_TYPE=postgresql**, as follows:

Property	Value from fmeDatabaseConfig.txt
-d	DB_JDBC_ URL=jdbc:postgresql://<hostname>:<port>/<database_name>
-h	DB_JDBC_ URL=jdbc:postgresql://<hostname>:<port>/<database_name>
-p	DB_JDBC_ URL=jdbc:postgresql://<hostname>:<port>/<database_name>
-U	DB_USERNAME=<username>
Password (when prompted)	DB_PASSWORD=<password>

3. Once connected, enter the following command to change the password for your user:

```
\password
```

4. Use `\q` to quit psql.

5. In the the [fmeDatabaseConfig](#) configuration file, update the value of the `DB_PASSWORD` parameter that corresponds to `DB_TYPE=postgresql` with the new password specified above, and save the file.
6. (Recommended) [Encrypt](#) the FME Flow Database password.
7. [Restart](#) FME Flow.

Provide a Remote File System for the FME Flow System Share

- **Skill Level:** Intermediate
- **Estimated Time Required:** 5-15 minutes
- **Prerequisites:**
 - Domain [Service Account](#) has correct permissions on network shares.

FME Flow System Share files, which include [Repositories](#) and [Resources](#), are installed in specific directories that are shared to enable access from different computers.

Create a directory for the FME Flow System Share that is physically separate from the machine on which the FME Flow Core and web application server are installed. Separating the file system ensures it remains available in a fault tolerant scenario.

When providing a remote file system, keep in mind the following:

- The accounts that run the FME Flow components must have read and write permissions on the remote file system directory. For more information, see [Directory and Account Permissions](#).
- In addition to being physically separate, the file system should, by itself, be configured for fault tolerance.
- When the directory is specified during installation, it must be in the form of a UNC path. Mapped drives are not supported.
- The directory that holds the System Share must be empty, unless System Share folders have already been created there as part of an installation involving multiple FME Flow Cores.


For information about acquiring a fault tolerant file system, consult your IT department.

Linux Only

For distributed installations on Linux, it is necessary to create a user, prior to installation, on the machine that will host the installation of the FME Flow Core and Engines. This user must be named 'fmeflow' and belong to group 'fmeflow', and be granted permissions on the FME Flow System Share. For more information, see [Directory and Account Permissions](#).

Example Instructions

The following example creates user 'fmeflow' in group 'fmeflow', and allows mount /mnt/data to be writable by the newly-created user.

 **Note** The following instructions are provided as a general guideline for adding a user and mounting a directory. Exact instructions may vary depending on the distribution and share type in your Linux configuration.

1. Run the following commands in the Linux command line:

```
groupadd fmeflow
useradd -m -g fmeflow fmeflow
```

2. Mount the FME Flow System Share directory so that user 'fmeflow' has read and write permissions. For example, to mount a Windows-based file share:

```
mount -v -t cifs //network/share /mnt/data -o gid=1008,uid=1008
```

`gid` and `uid` are the group id and user id of the user and group that were created.

These id's can be found using these commands:

```
id -g fmeflow
id -u fmeflow
```


Provide a Web Application Server

Provide a web application server on a separate machine from your FME Flow installation.

FME Flow supports Apache Tomcat version 9.0.x. On Windows, we recommend running the web application server on the same version of the Java Runtime Environment as FME Flow runs. To determine this version, use a text editor to open file `release` in `<FMEFlowDir>\Utilities\jre`.

The following are *not* supported:

- Apache Tomcat version 10.x.
- Java Runtime Environment 9, 16, 18 or later.

 **Note** On Linux, you must add the following environment variable to etc/environment:

```
JAVA_HOME="/usr/lib/jvm/java-8-openjdk-amd64"
```

You must know the port number on which your web application server receives incoming requests and the path to the directory where your web application server stores web application `.war` files. (This directory is normally called `webapps`.)

Additionally, you must allow the necessary system accounts permission to run the web application server. For more information, see [Directory and Account Permissions](#).

Obtain the Installer

To obtain the applicable FME Flow installation package, visit [the FME Downloads page](#). This page includes download links to release and beta installation packages for various platforms.


You can also contact sales@safe.com to get access.

What's Next?

If you are opting to install, on a separate machine, a PostgreSQL database that is included with the FME Flow installer, and do *not* wish to configure the FME Flow Database on your own database server, proceed to [Install the FME Flow Database](#). Otherwise, proceed to [Install the FME Flow Core and FME Engines](#).

Install the FME Flow Database


Install a PostgreSQL FME Flow Database on a separate machine on your network.

 **Note** Complete this step *only* if you wish to install, on a separate machine, a PostgreSQL database that is included with the FME Flow installer, and do *not* wish to configure the FME Flow Database on your own database server. For more information, see [Provide a Database Server](#).

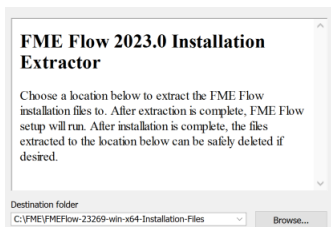
Start the FME Flow installer.

(Windows only) Installation Extractor

On the Installation Extractor dialog, specify a *Destination folder* in which to extract installation files. Note that this folder *only* specifies where installation files are extracted—it does *not* specify where FME Flow program files are installed. That location is specified later in the installer.

 **Tip** Take note of the specified *Destination folder* location. Following installation, you may wish to:

- Review the installation .log file.
- Remove these files to save disk space.

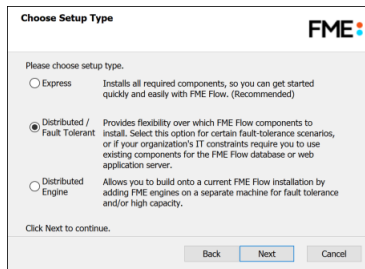


Click **Install**.

When extraction is complete, the installer opens. Proceed through the installer to the Choose Setup Type dialog.

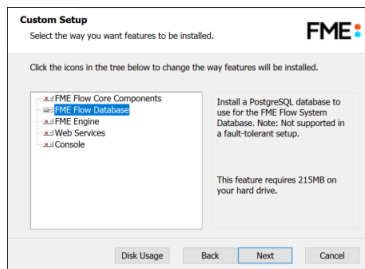
Choose Setup Type

Select the Distributed/Fault Tolerant install option.



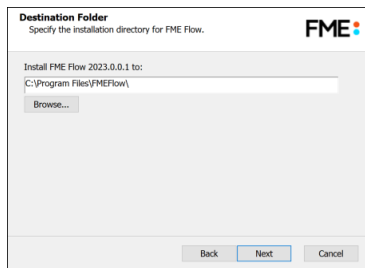
Custom Setup

Specify FME Flow Database. Unselect all other components.



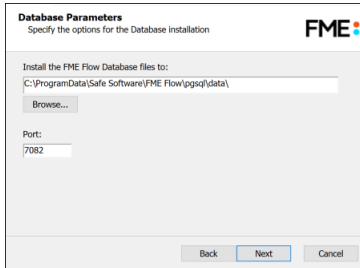
Destination Folder

Specify the installation directory for the FME Flow Database.



Database Parameters

- *Install the FME Flow Database files to:* Specify a directory to install the database files.
- *Port:* Specify the port that the database will listen on. Port **7082** is the default.

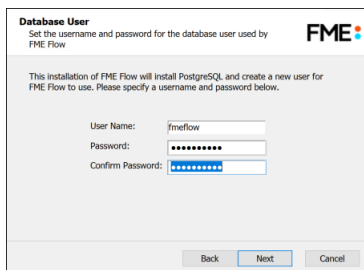


Database User

Specify a *User Name* (defaults to **fmeflow**) and *Password* for the database user account.

Warning Do not specify *User Name* **postgres**. This name corresponds to the PostgreSQL master user. If the corresponding passwords do not match, the database will not install.

Note The password for the FME Flow Database user account is encrypted in FME Flow installation files. Take note of the *User Name* and *Password* that you specified. You may need to reference it later.



Follow the remaining dialogs to complete the installation.

What's Next?


- [Install the FME Flow Core and FME Engines](#)

Install the FME Flow Core and FME Engines

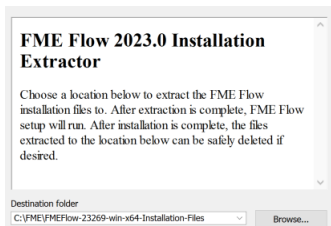
Start the FME Flow Installer.

(Windows only) Installation Extractor

On the Installation Extractor dialog, specify a *Destination folder* in which to extract installation files. Note that this folder *only* specifies where installation files are extracted—it does *not* specify where FME Flow program files are installed. That location is specified later in the installer.

 **Tip** Take note of the specified *Destination folder* location. Following installation, you may wish to:

- Review the installation .log file.
- Remove these files to save disk space.

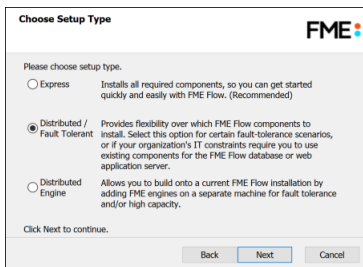


Click **Install**.

When extraction is complete, the installer opens. Proceed through the installer to the Choose Setup Type dialog.

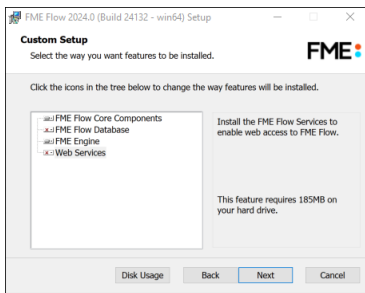
Choose Setup Type

Select the Distributed/Fault Tolerant install option.



Custom Setup

Select **FME Flow Core Components** and **FME Engine**. Do *not* select **FME Flow Web Services**.

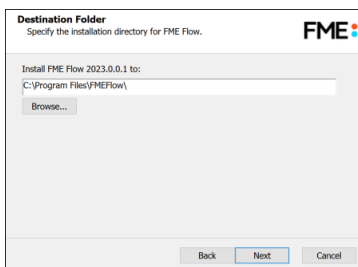


Optionally, select **FME Flow Database**. If you do so, note that:

- This option is supported, but not described in [Distributing FME Flow Components](#). Do not select **FME Flow Database** if you already completed the previous step, [Install the FME Flow Database](#), or plan to configure the FME Flow Database on your own database server.
- The step to define [Database Server Type](#) (below) will not appear.

Destination Folder

Specify the installation directory for FME Flow.




User Account


Specify the Windows user account that will run the FME Flow [System Services](#). This account must have:

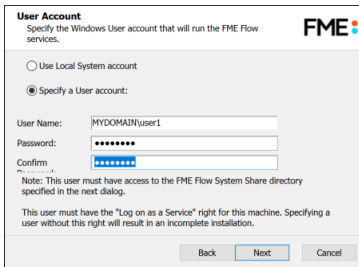
- Read/Write access to the FME Flow System Share that you specify in the next dialog. The System Share is the location where FME Flow stores [Repositories](#) and [Resources](#) files. (For more information, see [Provide a Remote File System for the FME Flow System](#))

[Share](#).)

- Read/Write access to the FME Flow installation directory that you specified in the previous dialog.
- "Log on as a service" rights on this machine. For more information, see [this Microsoft Docs article](#).


 **Note** If unsure, you can update account settings for the FME Flow system services after installation in the Services manager (Windows) or Linux equivalent.


 **Note** The FME Flow Database service always runs under the local system account.

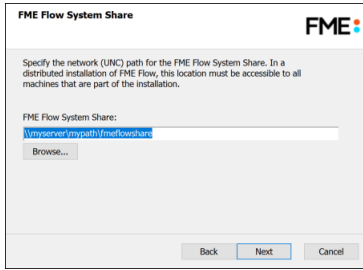


FME Flow System Share

Specify a directory on another machine to store FME Flow System Share files, which include [Repositories](#) and [Resources](#). If this is a Linux installation, specify the mounted directory. For more information, see [Provide a Remote File System for the FME Flow System Share](#).

 **Tip** As a best practice, specify the full UNC path to the FME Flow System Share, even if it is on the same (local) machine where you are running the installer.

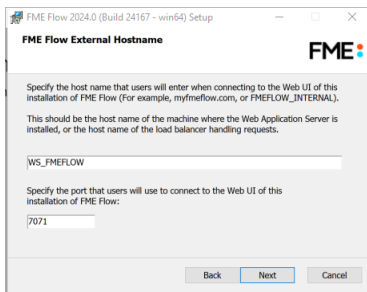
 **Note** If you are [upgrading](#) to a newer version of FME Flow, make sure the specified directory is empty of files from the previous installation. Otherwise, conflicts may occur that can produce unexpected results in performance.



FME Flow External Hostname

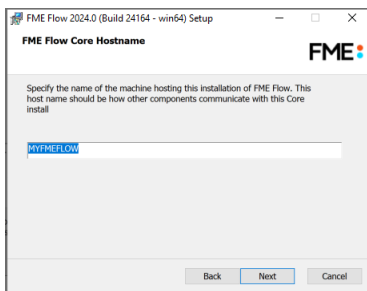
Specify the Web Application Server hostname through which users will connect to FME Flow.

Additionally, specify the port for this connection.




FME Flow Core Hostname

Specify the name of the machine hosting this installation of FME Flow.

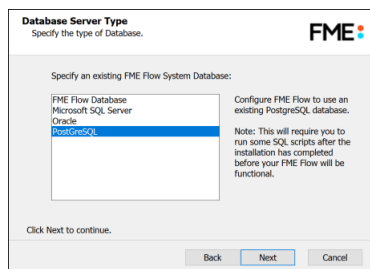


Database Server Type

 **Note** This dialog does not appear if you elected to install the **FME Flow Database** in step [Custom Setup](#) (above).

Specify the type of database for the FME Flow Database. If you installed, on a separate machine, the PostgreSQL database included with the FME Flow installer, select **FME Flow**

Database. If you are configuring the FME Flow Database on your own database server, select the database server type: **Microsoft SQL Server**, **Oracle**, or **PostgreSQL**. For more information, see [Provide a Database Server](#).

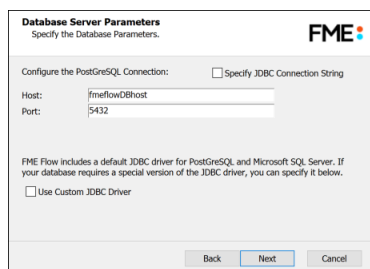


Database Server Parameters

Specify the connection parameters for the FME Flow Database. Optionally, to customize the connection parameters in the JDBC connection string, check **Specify JDBC Connection String**.


If **Microsoft SQL Server** was specified in the previous dialog, check *Use SQL Server instance name* to use that value instead of *Port*.

FME Flow includes a default JDBC driver for PostgreSQL and Microsoft SQL Server. If your database requires a special version of the JDBC driver, check *Use Custom JDBC Driver* and specify the *JDBC Driver*. If **Oracle** was specified in the previous dialog, you must specify the *JDBC Driver*.



If you elected to install the **FME Flow Database** in step [Custom Setup](#) (above), a Database Parameters dialog appears instead:


- **Install the FME Flow Database files to:** By default, [FME Flow Database](#) files are written to %ALLUSERSPROFILE%\Safe Software\FME Flow\pgsql\data.

 **Note** Unless modified, C:\ProgramData is the default value of the %ALLUSERSPROFILE% environment variable.

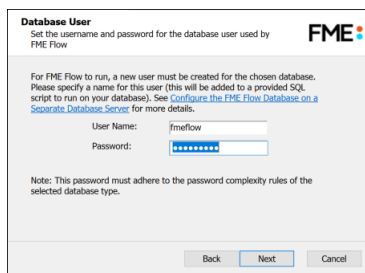
To change the installation directory of FME Flow Database files, modify the path.

Database User

The installation creates a user account under which to use the FME Flow database. Specify a *User Name* (defaults to **fmeflow**) and *Password* for the database user account.

 **Note** The password for the FME Flow Database user account is encrypted in FME Flow installation files. Take note of the *User Name* and *Password* that you specified. You may need to reference it later.

If you installed the PostgreSQL database included with the FME Flow installer, specify the same *User Name* and *Password* that you specified for Database User previously, under [Install the FME Flow Database](#).



Follow the remaining dialogs to complete the installation.

What's Next?

- Proceed to [Install the FME Flow Web Services](#)


Install the FME Flow Web Services

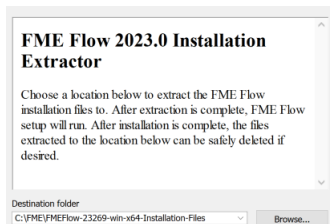
Perform all tasks, where applicable:

[Install the FME Flow Web Services](#)

Windows

1. Start the FME Flow Installer on the <webHost>.
2. On the Installation Extractor dialog, specify a *Destination folder* in which to extract installation files. Note that this folder *only* specifies where installation files are extracted—it does *not* specify where FME Flow program files are installed. That location is specified later in the installer.

-  **Tip** Take note of the specified *Destination folder* location. Following installation, you may wish to:
- Review the installation .log file.
 - Remove these files to save disk space.




Click **Install**.


When extraction is complete, the installer opens. Proceed through the installer to the Choose Setup Type dialog.

3. Select the Distributed/Fault Tolerant setup type.
4. On the Custom Setup dialog, select only the Web Services component for installation.
5. On the Destination Folder dialog, specify the installation directory.
6. On the User Account dialog, specify the Windows user account that will run the FME Flow [System Services](#). This account must have:
 - Read/Write access to the FME Flow System Share that you specify in the next dialog. The System Share is the location where FME Flow stores [Repositories](#) and [Resources](#) files. (For more information, see [Provide a Remote File System for the FME Flow System Share](#).)

- Read/Write access to the FME Flow installation directory that you specified in the previous dialog.
- "Log on as a service" rights on this machine. For more information, see [this Microsoft Docs article](#).


 **Note** If unsure, you can update account settings for the FME Flow system services after installation in the Services manager (Windows) or Linux equivalent.

7. On the Existing FME Flow System Share dialog, specify the UNC path to the [remote file system](#) for the FME Flow System Share.

 **Tip** As a best practice, specify the full UNC path to the FME Flow System Share, even if it is on the same (local) machine where you are running the installer.

8. On the FME Flow External Hostname dialog, enter the name of the machine hosting this installation of the FME Flow Web Services, as users will specify when connecting to the [FME Flow Web User Interface](#). If a load balancer or reverse proxy is deployed through which to access the Web Application Server, check *Advanced* and provide the *External Web Traffic Port*. Specify this port only if it differs from the Web Application Server Port (specified below).
9. On the FME Flow Core Hostname dialog, enter the host name of the computer on which the [FME Flow Core](#) is installed.
10. On the Database Server Type dialog, select the [Database Server Type](#) as specified previously in the [Install the FME Flow Core and FME Engines](#) wizard.
11. On the Database Server Parameters dialog, provide the [Database Server Parameters](#) for the FME Flow Database, as specified previously in the [Install the FME Flow Core and FME Engines](#) wizard.

12. On the Database User dialog, provide the [Database User](#) password, as specified previously in the [Install the FME Flow Core and FME Engines](#) wizard.
13. On the Web Application Server dialog, specify the [web application server](#). If already provided, it is recommended that you shut down the service during installation.
14. On the Web Application Server Port dialog, specify the port to use for communication with the web application server. Port 80 is the recommended default. If this port is in use, port 8080 is recommended.

 **Note** Common applications that may be using port 80 include Skype and Internet Information Services (IIS). Either turn off these services, or select a different port, such as 8080.

Linux

1. Start the FME Flow Installer.
2. Select the Distributed (Custom) install option.
3. At the feature selection dialog, select only the Web Services component.
4. At the External Hostname prompt, enter the name of the machine hosting this installation of the FME Flow Web Services, as users will specify when connecting to the [FME Flow Web User Interface](#). If a load balancer or reverse proxy is deployed through which to access the Web Application Server, specify the *External Web Traffic Port* as well. Specify this port only if it differs from the Web Application Server Port (specified below).
5. Enter the host name of the computer on which the [FME Flow Core](#) is installed.
6. The installer prompts you for the path to the remote file system for the FME Flow System Share. Enter the path to the mounted directory that was specified in [Provide a Remote File System for the FME Flow System Share](#).
7. Select the [Database Server Type](#) as specified previously in the [Install the FME Flow Core and FME Engines](#) wizard.

8. Enter the [Database Server Parameters](#) for the FME Flow Database, as specified previously in the [Install the FME Flow Core and FME Engines](#) wizard.
9. Enter the [Database User](#) password, as specified previously in the [Install the FME Flow Core and FME Engines](#) wizard.
10. The installer prompts you to specify the [web application server](#). If already provided, it is recommended that you shut down the service during installation. Port 8080 is recommended.
11. Manually start FME Flow and configure it to start automatically. See [Working with the FME Flow System Services](#).
12. In order for the Data Download and Data Streaming web services to reference the correct web host location when outputting results that are returned from an FME Engine, you must manually change this location in your FME Engine configuration files:
 - a. Locate your FME Engine configuration files, under `<FMEFlowDir>\Server\fmeEngineConfig*.txt`
 - b. For each file, under the SUB_SECTION for FILE_DOWNLOAD_SERVICE and STREAM_DOWNLOAD_SERVICE, modify the SUCCESS_RESPONSE directive as follows:

From:

```
SUCCESS_RESPONSE 0:Translation Successful|OutputLocation=!FME_AUTO_
DIR_NAME!
```

To:

```
SUCCESS_RESPONSE 0:Translation
Successful|OutputLocation=<webHost>/DefaultResults/!FME_AUTO_DIR_
NAME_SIMPLE!
```

Following the installation on `<webHost>`, the FME Flow Web Application Server automatically starts. If you used your own web application server, you can restart that service.


Configure Firewall Settings

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


What's Next?

If you are configuring the FME Flow Database on your own database server, and did *not* install the PostgreSQL database that is included with the FME Flow installer, proceed to [Configure the FME Flow Database on a Separate Database Server](#). Otherwise, proceed to [Log on to the Web User Interface](#).


Configure the FME Flow Database on a Separate Database Server

 **Note** Complete this step *only* if you are configuring the FME Flow Database on your own database server, and did *not* install the PostgreSQL database that is included with the FME Flow installer. For more information, see [Provide a Database Server](#).

To configure FME Flow to use a separate database server, follow the steps below. You can configure FME Flow with a PostgreSQL (recommended), Microsoft SQL Server, or Oracle database, running on either Windows or Linux.

 **Note** PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.

1. Database Configuration: Set up FME Flow database tables and database users with permissions to access the FME Flow database.

 **Note** The specific way to set up database tables and user permissions on various production databases may differ depending on the available database tools and intended target platform.

Database Configuration - PostgreSQL

PostgreSQL is an open source database that can be downloaded from the Internet. For more information, see the PostgreSQL web site at:

<https://www.postgresql.com>

It's assumed that PostgreSQL is already installed .

One notation used is <FMEFlowDir>, which is the installation directory of FME Flow. In Windows, this is typically `C:\Program Files\FMEFlow`.

It's also assumed that you have JRE 5 or higher and PostgreSQL database installed before you proceed further. If you want to use the JDBC driver, there are several steps to perform.

If you are upgrading, you should back up any job history you want to keep.

Windows

This section describes how to configure a PostgreSQL database. You can configure this database from other PostgreSQL database tools, depending on your personal preference.

For a PostgreSQL database configuration, various SQL scripts are provided to assist you. These SQL scripts are located in the following directory:

`<FMEFlowDir>\Server\database\postgresql`

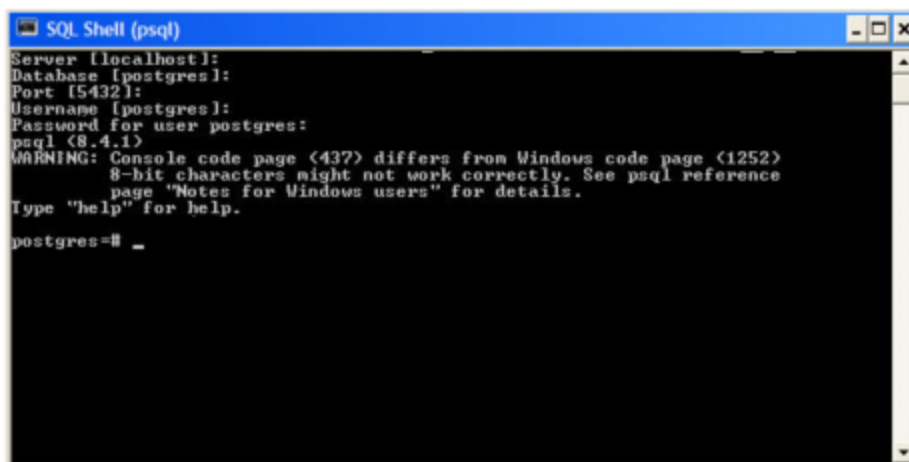
Note This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `postgresql_createUser.sql` – This script creates the FME Flow database user and password that you specified during installation, and grants all required permissions to this user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- `postgresql_createDB.sql` – This script creates the FME Flow database.
- `postgresql_createSchema.sql` – This script creates all FME Flow related tables, indexes, views, and triggers.
- `postgresql_dropUser.sql` – This script drops the FME Flow database user.
- `postgresql_dropDB.sql` – This script drops the FME Flow database.

Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install PostgreSQL SQL shell (`psql`) if you haven't done so already.
2. Open `psql` and log on as a user with the appropriate privileges:



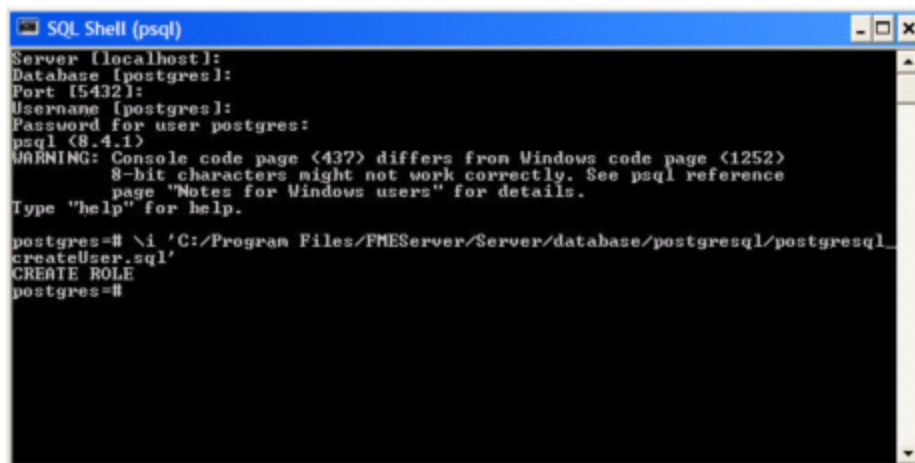
```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.
postgres=# _
```

3. Create an FME Flow database user as follows:

- a. Open `postgresql_createUser.sql` in a text editor, and replace `<<DATABASE_PASSWORD>>` with the password specified during the FME Flow installation.
- b. From the SQL prompt, run the `postgresql_createUser.sql` script by entering the following command:

```
\i '<FMEFlowDir>/Server/database/postgresql/postgresql_createUser.sql'
```

By default, the provided SQL script creates the FME Flow database user and password that you specified during installation:



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql <8.4.1>
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

postgres=# \i 'C:/Program Files/FMEServer/Server/database/postgresql/postgresql_createUser.sql'
CREATE ROLE
postgres=#
```

4. Create an FME Flow database.

From the SQL prompt, run the `postgresql_createDB.sql` script by entering the following command:

```
\i '<FMEFlowDir>/Server/database/postgresql/postgresql_createDB.sql'
```

By default, the provided SQL script creates an FME Flow database named `fmeFlow` and grants all privileges on the database to the user.

```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

postgres=# \i 'C:/Program Files/FME Server/Server/database/postgresql/postgresql_
createUser.sql'
CREATE ROLE
postgres=# \i 'C:/Program Files/FME Server/Server/database/postgresql/postgresql_
createDB.sql'
CREATE DATABASE
GRANT
postgres=# _
```

5. Quit `psql` and log on as the FME Flow database user.

```
SQL Shell (psql)
Server [localhost]:
Database [postgres]: fmeserver
Port [5432]:
Username [postgres]: fmeserver
Password for user fmeserver:
psql (8.4.1)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

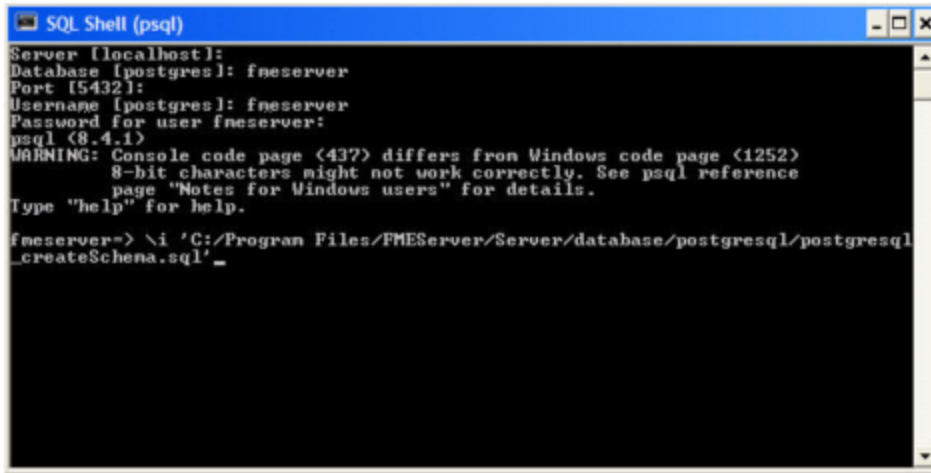
fmeserver=> _
```

6. Create the FME Flow database schema as follows:

From the SQL prompt, run the `postgresql_createSchema.sql` script by entering the following command:

```
\i '<FMEFlowDir>/Server/database/postgresql/postgresql_
createSchema.sql'
```

By default, the provided SQL script creates all FME Flow related tables, indexes, views, and triggers.




Linux

This section describes configuring PostgreSQL from both the command prompt and a user interface, such as the PostgreSQL Query Browser and the PostgreSQL Administrator.


For PostgreSQL configurations, various SQL scripts are provided to help with PostgreSQL database configuration. These SQL scripts are located in the `<FMEFlowDir>/Server/database/postgresql/` directory of the FME Flow Core machine.

Note This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `postgresql_createUser.sql` – This script creates the FME Flow database user and password that you specified during installation, and grants all required permissions to the FME Flow database user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.

 **Note** The password is encrypted and appears in the script as <<DATABASE_PASSWORD>>.

- `postgresql_createDB.sql` – This script drops the FME Flow database and creates a new one with the required FME Flow tables. By default this database is named `fmeflow`. Be careful when running this script because it drops the existing FME Flow database.
- `postgresql_createSchema.sql` – This script creates all FME Flow related tables, indexes, views, and triggers.

 **Note** This script must be run as user `fmeflow`.

- `postgresql_dropDB.sql` – This script drops the existing FME Flow database, which is named `fmeflow` by default.
- `postgresql_dropUser.sql` – This script drops the FME Flow database user.

Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install PostgreSQL Server if you haven't already.
2. Open a command prompt and change to the following directory, which contains the PostgreSQL SQL scripts:

```
<FMEFlowDir>/Server/database/postgresql/
```

3. Open `postgresql_createUser.sql` in a text editor, and replace <<DATABASE_PASSWORD>> with the password specified during the FME Flow installation.

4. Run the `postgresql` program with the `postgresql_createUser.sql` script:


```
psql -d postgres -h <hostname> -p <port> -f postgresql_createUser.sql
```


5. Run the postgresql program with the postgresql_createDB.sql script:

```
psql -d postgres -h <hostname> -p <port> -f postgresql_createDB.sql
```

6. Run the postgresql program with the postgresql_createSchema.sql script, as the user you specified during installation:

```
psql -U <username> -d fmeFlow -h <hostname> -p <port> -f postgresql_createSchema.sql
```

 **Note** If you drop the FME Flow database and login user, run the postgresql program with the appropriate postgresql_dropDB.sql or postgresql_dropUser.sql scripts.

After Database Creation

1. On the machine on which the FME Flow database server is installed, open pg_hba.conf. For default installation directories of this file:
 - PostgreSQL 10: `<PostgreSQLDir>\data\pg10\pg_hba.conf`
 - PostgreSQL 11: `C:\Program Files\PostgreSQL\11\data\pg_hba.conf`
2. Change the following lines:

```
host    all    all    127.0.0.1/32    md5
host    all    all    ::1/128         md5
```

to:

```
host    all    all    0.0.0.0/0      md5
host    all    all    ::/0           md5
```


3. [Restart](#) the FME Flow Database service.

Database Configuration - Microsoft SQL Server

It is assumed that SQL Server is installed.

One notation used is `<FMEFlowDir>`, which is the installation directory of FME Flow. In Windows, this is typically `C:\Program Files\FMEFlow`.

SQL scripts are provided to help with SQL Server database configuration. These SQL scripts are located in the `<FMEFlowDir>\Server\database\sqlserver\` directory of the FME Flow Core machine.

 **Note** This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

If you are upgrading, you should back up any job history you want to keep.

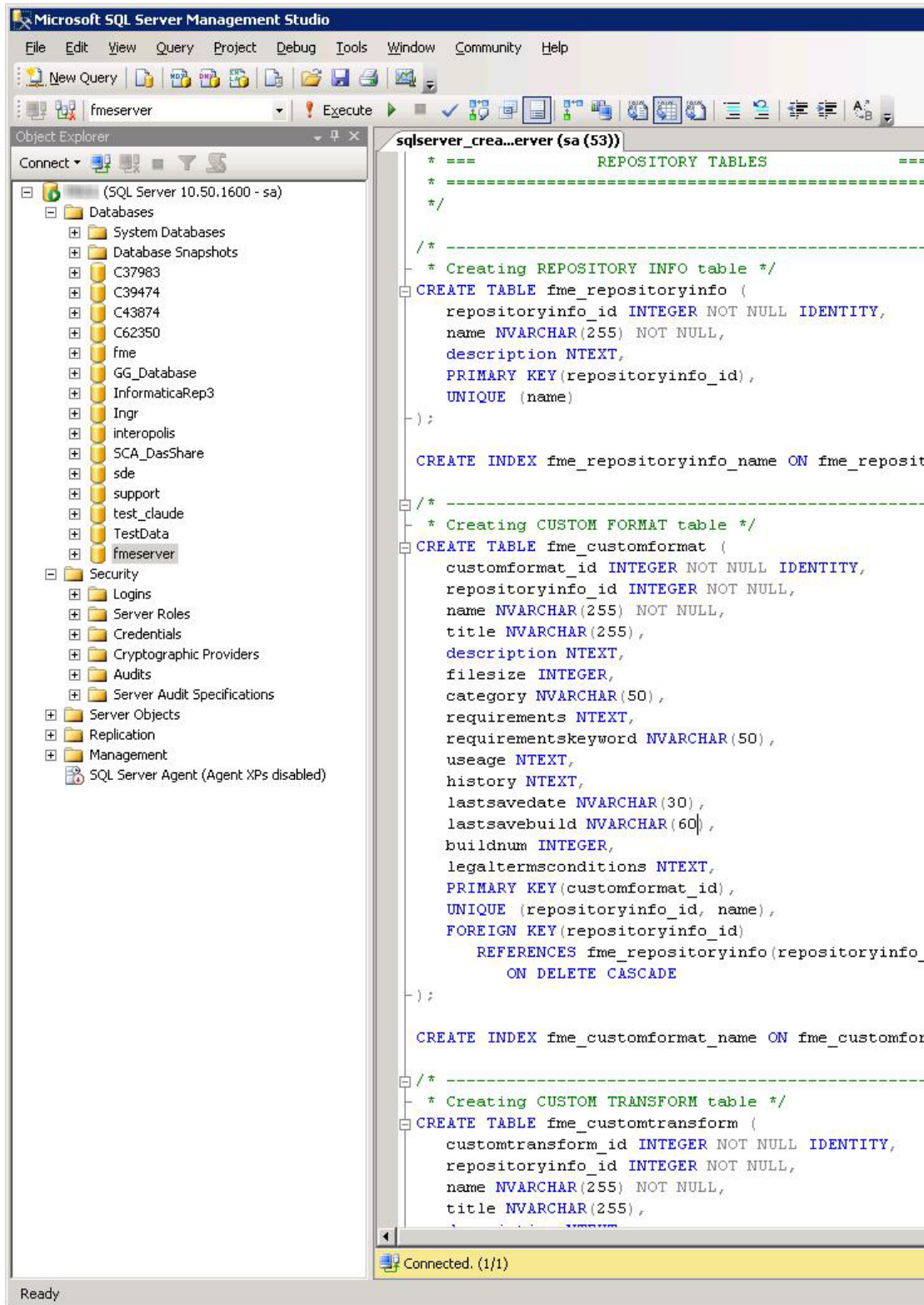
Prerequisite for Unsecured Database Servers

If your SQL Server database server is not secured with a valid SSL certificate, you must update the [fmeDatabaseConfig.txt](#) configuration file as follows:

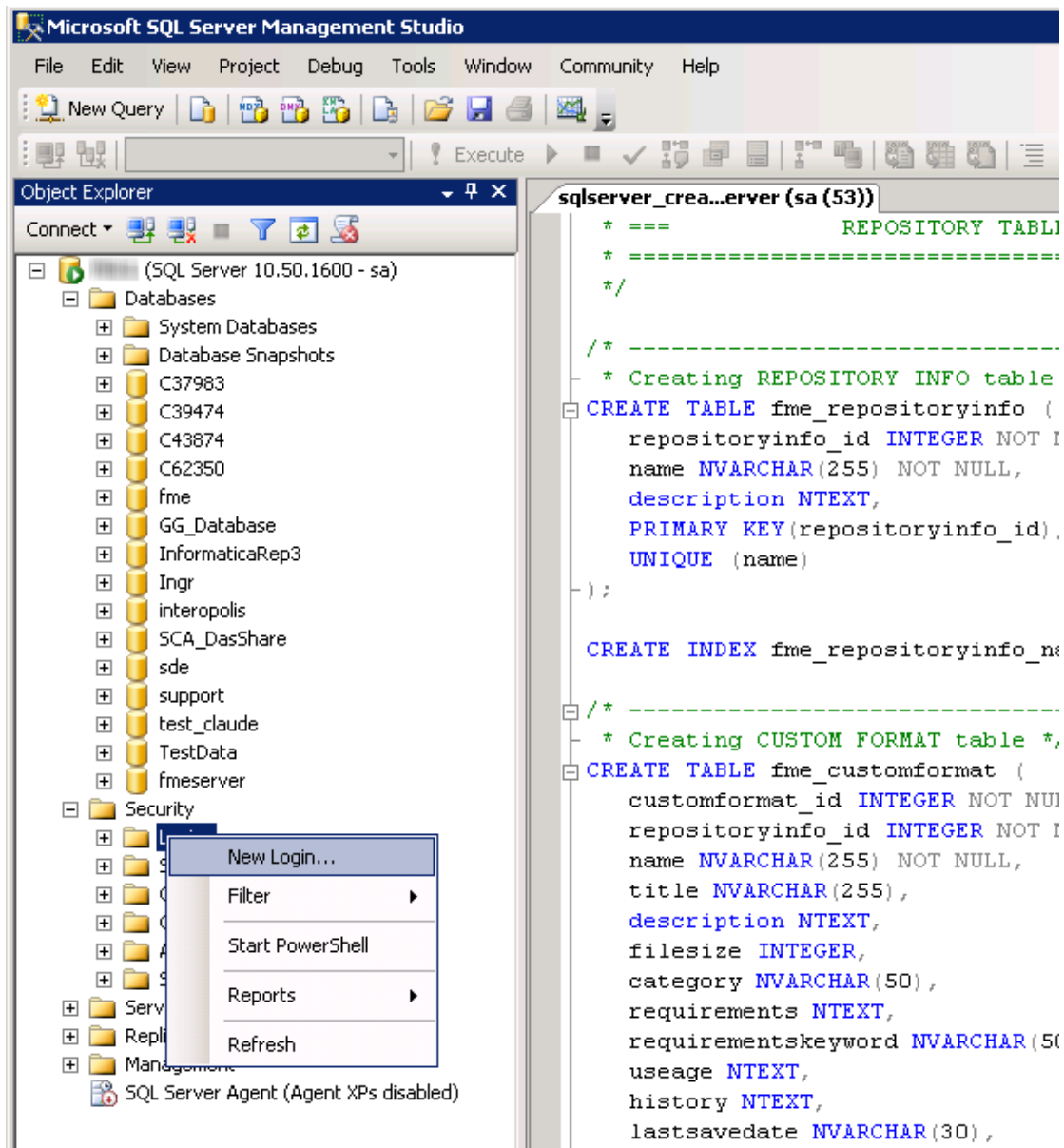
1. Near the bottom, under the heading `FME SERVER SETTINGS START`, locate the block of code beginning with `DB_TYPE=sqlserver`.
2. In the corresponding line for the `DB_JDBC_URL` parameter, update the `encrypt` value from `true` to `false`.
3. Save and close the file.

Instructions Using SQL Server Management Studio

1. Open the SQL Server Management Studio and connect to the SQL Server database engine.
2. Open `sqlserver_createDB.sql` and run the script against the `fmeFlow` database, as shown below:



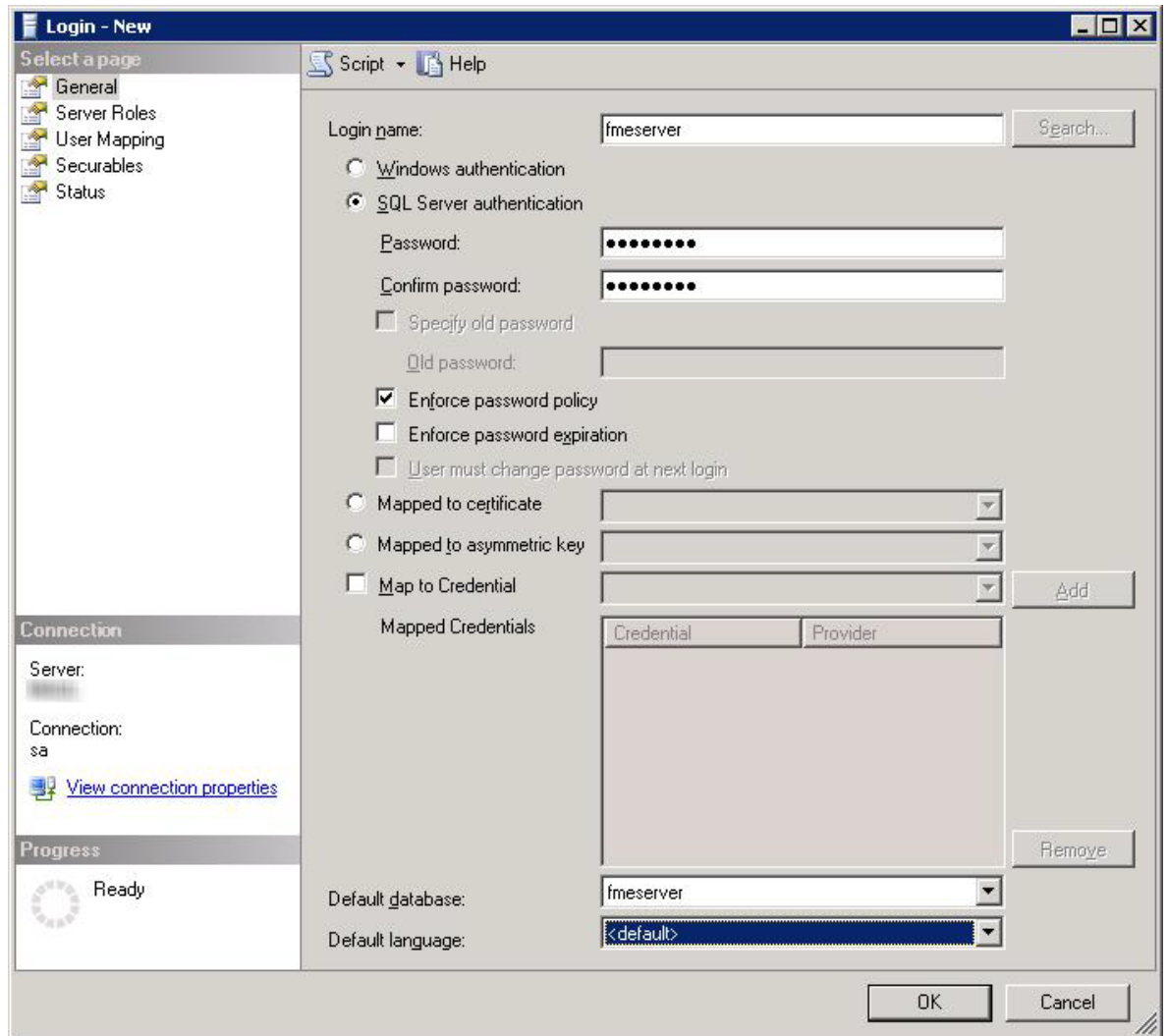
3. On the Object Explorer pane, expand the Security folder, right-click Login, and then select New Login:



4. On the Login – New page, click General to open the page shown in the example below.

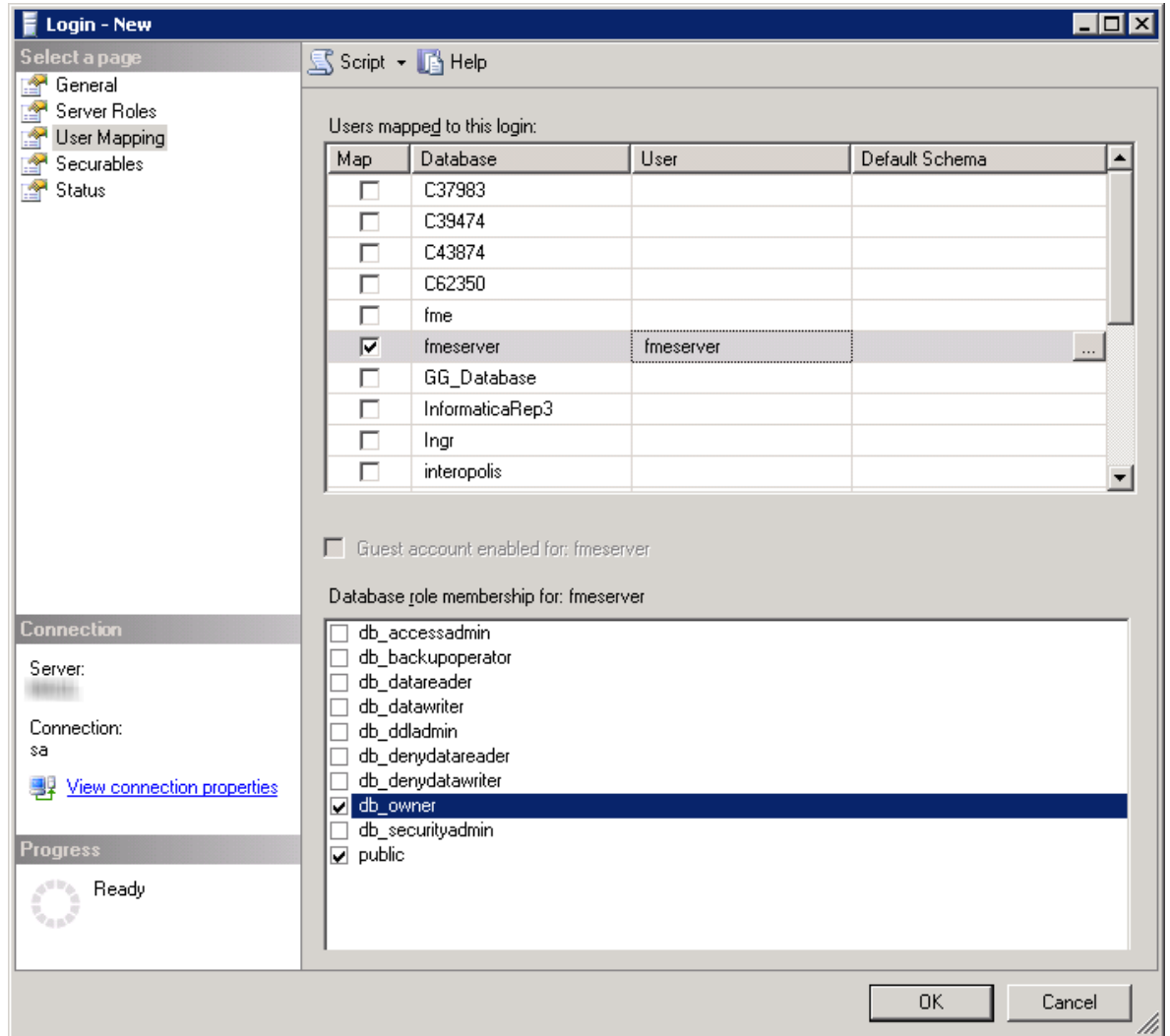
- At the Login name field, enter **fmeFlow**.
- Make sure the option button is active for SQL Server authentication.
- In the Password field enter the password that was specified for the database during the FME Flow installation, and in the Confirm password field enter the same password again.

- Make sure the Enforce password policy is active. If not, click the checkbox to activate it.
- If required, click the checkbox for Enforce password expiration to clear it.
- At the Default database field enter fmeserver.



5. On the same Login – New page, click User Mapping to open the page shown in the next figure.
 - At the Users mapped to this login location, in the Map column click the checkbox to activate it and map the login name to the fmeserver database.

- At the Database role membership for location, notice that the fme\flow database is specified and check `db_owner` to assign that role to this database.
- Click OK to close this page.




Instructions Without SQL Server Management Studio

1. Connect to the SQL Server database engine and run `sqlserver_createDB.sql`.
2. Open `sqlserver_createUser.sql`.
3. Look for the line that begins:

```
CREATE LOGIN fme\flow WITH PASSWORD =
```

Replace `<<DATABASE_PASSWORD>>` with the password specified during the installation.

4. Run the script.

 **Note** If your SQL Server uses an Always On availability group, you must also configure the `MultisubnetFailover` property of the `DB_JDBC_URL` parameter in the `fmeDatabaseConfig.txt` configuration file. For more information, see the SQL Server examples under [Configure the Database Connection](#).

Database Configuration - Oracle

Before proceeding, you must provide the following:

- Install an Oracle database server.
- Obtain the Oracle Database JDBC driver. For more information, see <https://www.oracle.com/technetwork/database/application-development/jdbc/downloads/index.html>.

`<FMEFlowDir>` is the installation directory of FME Flow. In Windows, this is typically `C:\Program Files\FMEFlow`.


If you are upgrading, you should back up any job history you want to keep.

Windows

This section describes how to configure an Oracle database. You can configure the Oracle database from other Oracle database tools depending on your personal preference.

For an Oracle database configuration, various SQL scripts are provided to help.

These SQL scripts are located in the `<FMEFlowDir>\Server\database\oracle` directory of the FME Flow Core machine.

 **Note** This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `oracle_createUser.sql` – This script creates the FME Flow database user, and grants all required permissions to the FME Flow database user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- `oracle_createDB.sql` – This script creates all FME Flow related database packages, tables, indexes, views, and triggers.
- `oracle_dropUser.sql` – This script drops the FME Flow database user and the database packages, tables, indexes, views, and triggers associated with the FME Flow database user.

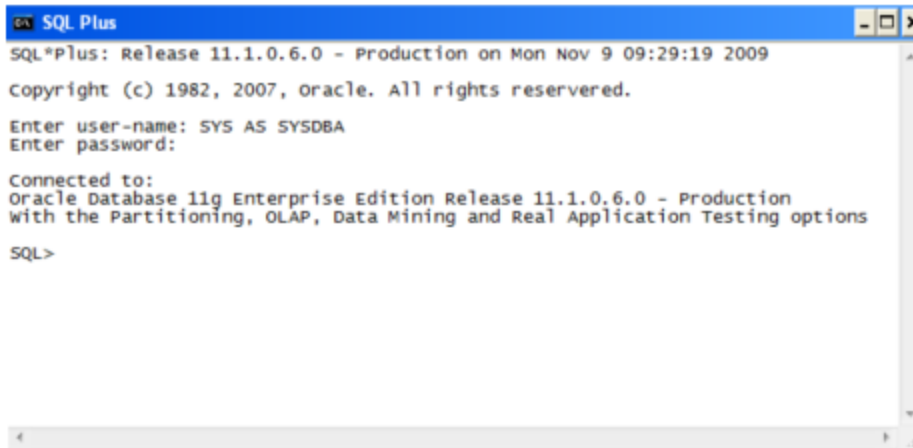
Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install Oracle SQL*Plus, if you haven't already.
2. Open Oracle SQL*Plus and log on as a user with adequate privileges, such as SYSDBA.

In this example, the following parameters were used and entered:

- User: SYS AS SYSDBA
- Password: fmefflow
- String: XX-ORACLE11G:1521/ORCL AS SYSDBA

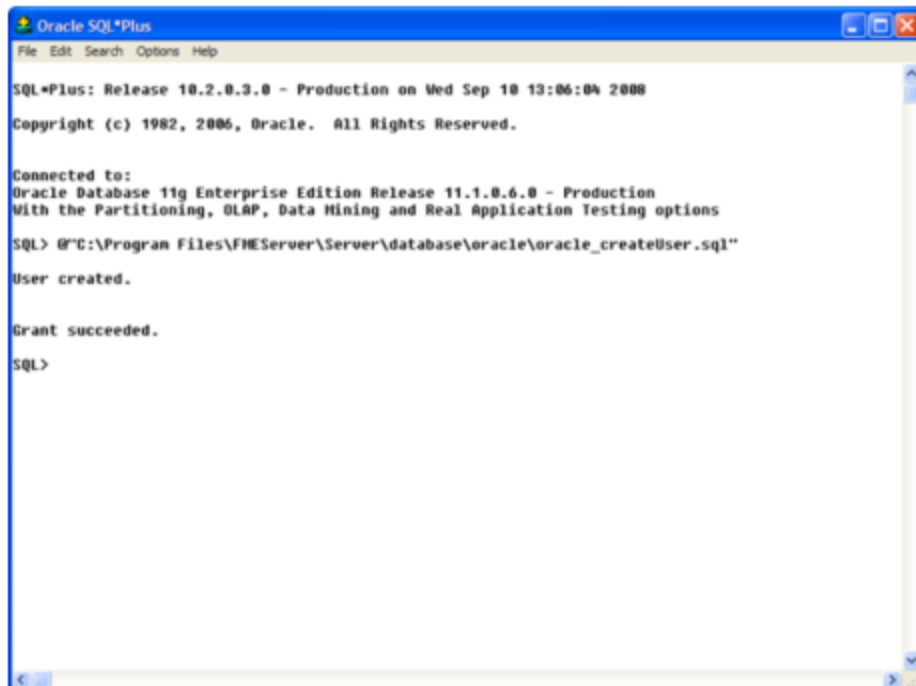


```
SQL Plus
SQL*Plus: Release 11.1.0.6.0 - Production on Mon Nov 9 09:29:19 2009
copyright (c) 1982, 2007, oracle. All rights reserved.
Enter user-name: SYS AS SYSDBA
Enter password:
Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
SQL>
```

3. Create an FME Flow database user as follows:

- a. In `oracle_createUser.sql`, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- b. From the SQL prompt, run the `oracle_createUser.sql` script by entering the following command:

`@>>"<FMEFlowDir>\Server\database\oracle\oracle_createUser.sql"`



```
Oracle SQL*Plus
File Edit Search Options Help
SQL*Plus: Release 10.2.0.3.0 - Production on Wed Sep 10 13:06:04 2008
Copyright (c) 1982, 2006, Oracle. All Rights Reserved.
Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 - Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
SQL> @>>"C:\Program Files\FMEServer\Server\database\oracle\oracle_createUser.sql"
User created.
Grant succeeded.
SQL>
```

The default TABLESPACE is the Oracle DB configured default. If the created user doesn't have read/write permissions on the default TABLESPACE, change the default TABLESPACE or create an appropriate TABLESPACE.

This command creates a TABLESPACE on the Oracle DB server called FMEFLOW shown in the following example:

```
CREATE SMALLFILE TABLESPACE FMEFLOW DATAFILE  
'E:\ORADATA\FMEFLOW01.DBF' SIZE 100000K REUSE AUTOEXTEND ON  
NEXT 2048K MAXSIZE 1024M LOGGING EXTENT MANAGEMENT LOCAL  
SEGMENT SPACE MANAGEMENT AUTO DEFAULT NOCOMPRESS;
```

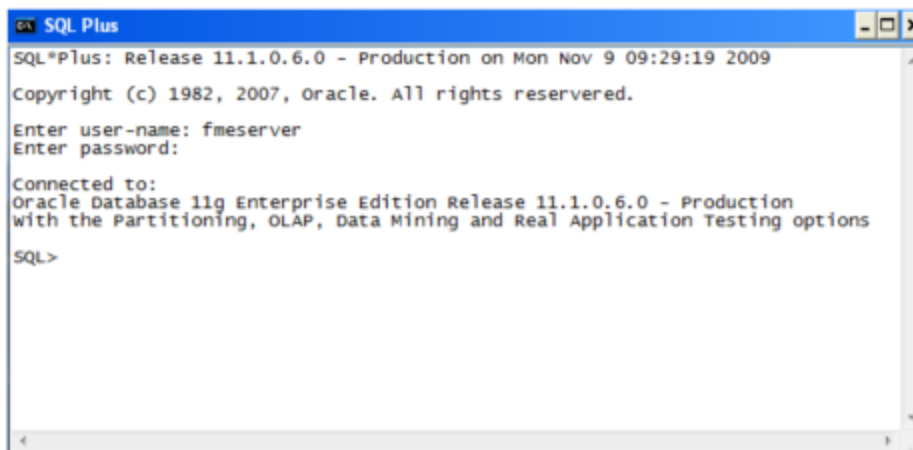
By default this command uses the TABLESPACE "FMEFLOW" as shown in the following example:

```
CREATE USER fmeflow IDENTIFIED BY fmeflow DEFAULT TABLESPACE  
"FMEFLOW" TEMPORARY TABLESPACE "TEMP";
```

4. When an FME Flow database user is created, you can connect to it by logging on again to SQL*Plus.

In this example, the following parameters were used and entered on the user interface:

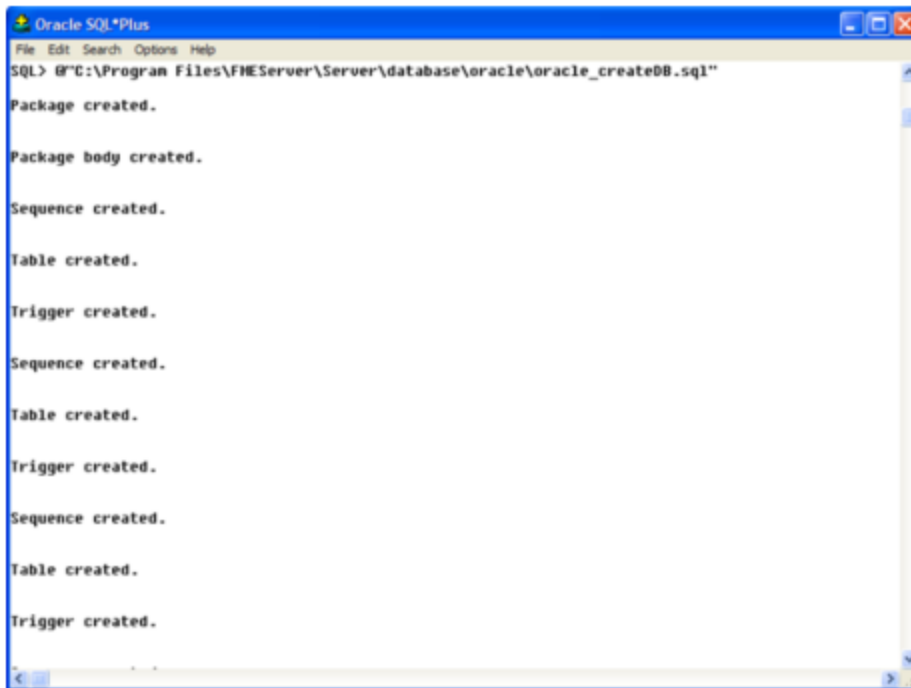
- User: fmeflow
- Password: fmeflow



Note The preceding example assumes your user name and password are fmeflow.

- When connected to the FME Flow database, run the `oracle_createDB.sql` script by entering the following command:

```
@ "<FMEFlowDir>\Server\database\oracle\oracle_createDB.sql"
```



Note If you drop the FME Flow database and login user, run the `oracle_dropUser.sql` script.

Linux

Oracle SQL*Plus is a freeware client for connecting to an Oracle database. You can download it from the Internet using the following address:


https://www.oracle.com/technology/software/tech/sql_plus/index.html

For more information, see the Oracle web site at:

https://www.oracle.com/technology/sample_code/tech/sql_plus/htdocs/demobld.html

The instructions that follow describe how to configure Oracle for use with FME Flow. These instructions describe using SQL*Plus for the database setup. Depending on your personal preference, you can also configure the Oracle database using other Oracle tools.

For Oracle configurations, various SQL scripts are provided to help with Oracle database configuration. These SQL scripts are located in the `<FMEFlowDir>/Server/database oracle/` directory of the FME Flow Core machine.

 **Note** This directory refers to the machine on which the FME Flow Core is installed (not the machine on which the database server is installed). To run these scripts, you must transfer this directory to the database machine, or reference it through a UNC path.

- `oracle_createUser.sql` – This script creates the FME Flow database user, and grants all required permissions to the FME Flow database user. The password appears as `<<DATABASE_PASSWORD>>`. Before running this script, replace `<<DATABASE_PASSWORD>>` with the password that was specified for the database during the FME Flow installation.
- `oracle_createDB.sql` – This script creates all FME Flow related database packages, tables, indexes, views, and triggers.
- `oracle_dropUser.sql` – This script drops the FME Flow database user and the database packages, tables, indexes, views, and triggers associated with the FME Flow database user.

Using the Command Prompt

To use the command prompt to configure the database, follow these instructions:

1. Install the Oracle Client Tools, which includes SQL*Plus, if you haven't already.
2. Run `sqlplus` and log on as a user with adequate privileges, such as SYSDBA.

For example:

```
sqlplus SYS/<password>@<Host>:<Port>/<service> AS SYSDBA
```

3. Create the FME Flow database user:
 - a. In a text editor, open the `oracle_creatorUser.sql` script and update the password `<<DATABASE_PASSWORD>>` to the one specified during the FME Flow installation.
 - b. From the SQL prompt, run the `oracle_createUser.sql` script by entering the following command:

```
@"<FMEFlowDir>/Server/database/oracle/oracle_createUser.sql"
```

The default TABLESPACE is the Oracle DB configured default. If the created user doesn't have read/write permissions on the default TABLESPACE, you need to change the default TABLESPACE or create an appropriate TABLESPACE.

This command creates a TABLESPACE on the Oracle DB server called FMEFLOW shown in the following example:

```
CREATE SMALLFILE TABLESPACE FMEFLOW DATAFILE  
'/ORADATA/FMEFLOW01.DBF' SIZE 100000K REUSE AUTOEXTEND ON  
NEXT 2048K MAXSIZE 1024M LOGGING EXTENT MANAGEMENT LOCAL  
SEGMENT SPACE MANAGEMENT AUTO DEFAULT NOCOMPRESS;
```

By default this command uses the TABLESPACE "FMEFLOW" shown in this example:

```
CREATE USER fmeFlow IDENTIFIED BY fmeFlow DEFAULT TABLESPACE  
"FMEFLOW" TEMPORARY TABLESPACE "TEMP";
```

4. When an FME Flow database user has been created, you can connect to it by logging on again to SQL*Plus or by entering the following command:

```
CONNECT <UserName>/<Password>@<Host>:<Port>/<service>;
```

5. When connected to the FME Flow database, run the oracle_createDB.sql script by entering the following command:

```
@"<FMEFlowDir>/Server/database/oracle/oracle_createDB.sql"
```

2. **Enable connections:** FME Flow connects to the database when it starts. Therefore, ensure that your database server is running and configured to accept incoming connections before FME Flow is started. The database must allow connections over TCP/IP with all machines on which the FME Flow Core and FME Engines are installed.
3. Ensure that all installations of FME Flow are running. For more information, see [Working with the FME Flow System Services](#).

What's Next?

- [Log on to the Web User Interface](#)

Log on to the Web User Interface

To log in to the Web User Interface for the first time as an administrator following installation, specify *Username* `admin` and *Password* `admin`. You will be prompted to update the `admin` password upon initial use, based on the default [Password Policy](#) configuration.

For more information and support:

- [FME Flow Troubleshooting: Web Interface Login](#)
- [Accessing the Web User Interface](#)
- [Changing the Login Password](#)
- [Default User Accounts and Passwords](#)
- [Role-Based and User-Based Access Control](#)

What's Next?

- [Update the WebSocket Host](#)

Update the WebSocket Host

Perform this step to allow communication with the FME Flow [WebSocket Server](#).

1. Log in to the [FME Flow Web User Interface](#) (see [Logging In to the Web Interface](#)).
2. Select **System Configuration > Networking > Topic Monitoring**.
3. Under Advanced Routing, set *External WebSocket Host* to the host name of the machine on which the FME Flow Core is installed.
4. Click **Save**.

What's Next?

- [Request and Install a License](#)

Request and Install a License

Note

License files are installed, by default, to:

» Windows: `C:\ProgramData\Safe Software\FME Flow\licenses`

» Linux: `/opt/fmeflow/licenses`

Dynamic Engine (CPU Usage) licensing is not available for engines on FME Flow Hosted.


Automatic Mode (Recommended)

If your FME Flow host machine has an internet connection, you can request and install a license online without any need for downloading and re-uploading a license file.


Note

- Licensing FME Flow over the internet requires no inbound connection. For more information, see [FME Flow Automatic Licensing URLs](#).
- If you purchased credits for Dynamic Engines (CPU Usage), automatic mode only is supported. Manual mode licensing (below) is not supported.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.
3. Complete the fields on the Request a New License page.


 **Note** If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.

4. *Licensing Mode*: Select **Automatic**.
5. Click **OK**.
6. A message indicates that your license is installed, and your available engines appear momentarily on the [Engines](#) page.


 **Note** If the *Serial Number* you provided includes Dynamic Engines (CPU Usage), they are not configured to start by default. You can start and configure your Dynamic Engines on the [Engines](#) page.

Manual Mode

If you prefer not to use an internet connection for licensing, or if your FME Flow host machine does not have an internet connection, use this process to install a license.

 **Note** If you purchased credits for Dynamic Engines (CPU Usage), manual mode licensing is not supported. Use automatic mode for licensing (above). For more information, contact your Safe Software Inc. sales representative.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the fmesuperuser [role](#). If none of your FME Flow engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.
3. Complete the fields on the Request a New License page.

 **Note** If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.

4. *Licensing Mode*: Select **Manual**.
5. Click **OK**. A license request .json file downloads.
6. Email the .json file to codes@safe.com.
7. You will receive an email from Safe Software Inc. with a license file, which has a `.fmelic` extension. Download the file.
8. Return to the Licensing page. Under Standard Engines, click **Upload License File** and select the `.fmelic` file to upload. Or, drag and drop the `.fmelic` file onto the page under Standard Engines. A message indicates that it is installed, and your available engines appear on the [Engines](#) page.

Problems?

- [FME Flow Troubleshooting: FME Flow Engines](#)
- Contact codes@safe.com.

See Also

- [FME Flow Automatic Licensing URLs](#)
- [FME Engines](#)

What's Next?

- (Recommended) [Update the Windows service accounts](#) that run the FME System Services.
- (Upgrade Only) [Restore](#) your FME Flow Configuration from a Previous Installation.
- [Test the Installation](#).

Test the Installation

To test your FME Flow installation, confirm that FME Flow can perform its primary function—running a job.

1. On the [Run Workspace](#) page, specify:
 - Repository: Samples
 - Workspace: austinApartments.fmw
 - Service: Job Submitter
2. Click Run Workspace.

If you see a success message, your install was a success.

See Also

- [FME Flow Troubleshooting: Submitting a Job in FME Flow](#)

Adding FME Engines on a Separate Machine

Warning This topic applies only to traditional FME Flow installations. If your FME Flow runs in a containerized environment, such as Docker Compose or a Kubernetes cluster, this architecture is not supported.

- **Skill Level:** Intermediate
- **Estimated Time Required:** 30-45 minutes
- **Prerequisites:**
 - Existing [Express, Distributed, or Fault-Tolerant](#) FME Flow installation.
 - Access to system, network, and FME Flow administrators.

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

When adding FME Engines, keep in mind the following:

- This approach requires the engine machines to be in the same network as your original FME Flow installation, and in the same data center or geographically close. To run jobs outside of your network, use [Remote Engine Services](#) instead.
- Adding FME Engines that do not match the primary release version of the [FME Flow Core](#) is not supported. The primary release version refers to the first decimal value following the release year. For example, if the release version of the FME Flow Core is 2020.0, FME Engines from release versions 2020.0.1 or 2020.0.2 are supported. However, FME Engines from release versions 2020.1 or 2021 are not supported.
- We recommend installing all FME Engines on systems that are synchronized to the same time zone as other FME Engines and the FME Flow Core. If time zones differ, unexpected issues may arise, including:
 - Difficulty accessing the [FME Flow Web User Interface](#).
 - Improper timing of [Schedule Initiated](#) triggers.
 - Inconsistent or misleading timestamps in log files (accessed from [Resources](#)).

For more information, see [About Times and Time Zones](#).

In the instructions below, the computer that hosts the FME Flow Core is the `<coreHost>`.

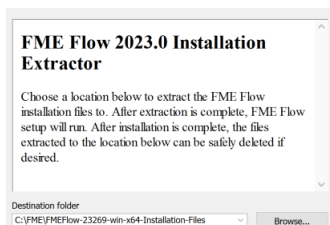
Note In a fault-tolerance environment, we recommend assigning unique names to FME Engines. However, if multiple FME Engine hosts have the same FME Engine name, the queue server configuration applies to all FME Engines with the same name in the same way regardless of which host it resides on. For more information, see [Job Queues](#).

Windows

1. [Obtain the Installer](#). Be sure to obtain the FME Flow Installer (not the Engine Installer).
2. Start the installer on another computer (`<engineHost>`).
3. On the Installation Extractor dialog, specify a *Destination folder* in which to extract installation files. Note that this folder *only* specifies where installation files are extracted—it does *not* specify where FME Flow program files are installed. That location is specified later in the installer.

Tip Take note of the specified *Destination folder* location. Following installation, you may wish to:

- Review the installation .log file.
- Remove these files to save disk space.




Click **Install**.

When extraction is complete, the installer opens. Proceed through the installer to the Choose Setup Type dialog.

4. On the Choose Setup Type dialog, select the Distributed Engine setup type.
5. On the Destination Folder dialog, specify the directory to install the FME Engines.
6. On the User Account page, specify the Windows user account that will run the <engineHost>. This account must have:
 - Read/Write access to the FME Flow System Share that you specify in the next dialog. The System Share is the location where FME Flow stores [Repositories](#) and [Resources](#) files.
 - Read/Write access to the FME Flow installation directory that you specified in the previous dialog.
 - "Log on as a service" rights on the <engineHost>. For more information, see [this Microsoft Docs article](#).
7. On the Existing FME Flow System Share dialog, provide the UNC path to the FME Flow System Share directory.

If the <coreHost> machine is an Express [installation](#) of FME Flow, the System Share is written, by default, to %ALLUSERSPROFILE% \Safe Software\FME Flow\, unless specified otherwise.


 **Note** Unless modified, C:\ProgramData is the default value of the %ALLUSERSPROFILE% environment variable.

In all cases, enter the UNC path to the System Share, including the final [Safe Software\FME Flow](#) directories, which are created by default. For example, [\\MyServer\Safe Software\FME Flow](#).

8. On the FME Flow Core Hostname dialog, enter the hostname of the <coreHost> computer. Additionally, if the machine hosting this installation of FME Engines is

identified by a name other than that specified under *Advanced*, check *Advanced* and specify the alternate name.

9. On the Database Server Type dialog, specify the database type that you specified when installing the FME Flow Core. If you specified a database other than the "FME Flow Database," enter the connection parameters in the next dialog.
10. On the Database User dialog, specify the user name and password for the FME Flow Database that you specified when installing FME Flow.
11. Complete the remaining dialogs of the FME Flow Installer.
12. If this *<engineHost>* is connecting to a *<coreHost>* that also hosts the built-in FME Flow database and the initial FME engines, as with an [Express Installation](#), allow the necessary connection to this machine:
 - a. On the *<coreHost>*, open file `pg_hba.conf`, located in `%ALLUSERSPROFILE%\Safe Software\FME Flow\pgsql\data`.

 **Note** Unless modified, `C:\ProgramData` is the default value of the `%ALLUSERSPROFILE%` environment variable.

- b. Change the following lines:

```
host    all        all        127.0.0.1/32    md5
host    all        all        ::1/128         md5
```

to:


```
host    all        all        0.0.0.0/0       md5
host    all        all        ::/0            md5
```

- c. [Restart](#) the FME Flow Database and FME Flow Engines services.
13. If the *<coreHost>* machine was an [Express Installation](#) of FME Flow, complete the following steps on the *<coreHost>* to configure the FME Flow Web Services:
 - a. Open the corresponding `propertiesFile.properties` file for the Data Download, Data Streaming, Job Submitter, and Notification services, located at

<FMEFlowDir>\Utilities\tomcat\webapps\<service_name>\WEB-INF\conf\propertiesFile.properties.

For example, the properties file for the Data Download Service is at <FMEFlowDir>\Utilities\tomcat\webapps\fmecatadownload\WEB-INF\conf\propertiesFile.properties

- b. For each properties file, locate the parameter REQUEST_DATA_DIR and change the value from a local file system path (for example, `C:/ProgramData/Safe Software/FME Flow//resources/system/temp/requestdata`) to a UNC path that references the <coreHost> (for example, `//MyHost/FME Flow/resources/system/temp/requestdata`).

 **Note** Use forward slashes only.

- c. After saving the changes in the properties file for each service, [restart](#) the FME Flow Application Server service.
14. On the <coreHost> machine, open configuration file [fmeFlowConfig.txt](#), and update the FME_SERVER_HOST_NAME parameter to a value that fully resolves to the IP address of the <coreHost> machine.
 15. [Restart](#) the FME Flow Core service.

Linux

1. If not already accomplished, mount the directory that holds the FME Flow System Share. (The System Share holds FME Flow [Repositories](#) and [Resources](#)):

```
sudo mount -t nfs <coreHost>:/<FMEFlowSystemShare> /<mntPath>
```
2. [Obtain the Installer](#). Be sure to obtain the FME Flow Installer (not the Engine Installer).
3. Start the FME Flow Installer:

```
sudo ./<installationFile>
```
4. From the setup type list, choose **Engine**.

5. The installer prompts you for a location to install the FME Flow Engine components. This can be on the local machine.
6. The installer prompts you for the name of the *<coreHost>*. In a fault-tolerant installation, the primary *<coreHost>* should always be specified. Do not specify the failover host.
7. The installer prompts you for the FME Flow System Share location. Enter the mounted directory created in step 1.
8. Enter the database type you specified when installing the FME Flow Core. If you specified a database other than the "FME Flow Database", enter the connection parameters.
9. Enter the user name and password for the FME Flow Database that you specified when installing FME Flow.
10. Complete the remaining dialogs of the FME Flow Installer.
11. Manually [start FME Flow](#), and configure it to start automatically.
12. If this *<engineHost>* is connecting to a *<coreHost>* that also hosts the built-in FME Flow database and the initial FME engines, as with an [Express Installation](#), allow the necessary connection to this machine:
 - a. On the *<coreHost>*, open file `pg_hba.conf`, located in `<FMEFlowSystemShare>\pgsql\data\`.
 - b. Change the following lines:


```
host    all    all    127.0.0.1/32    md5
host    all    all    ::1/128         md5
```

to:

```
host    all    all    0.0.0.0/0      md5
host    all    all    ::/0           md5
```
 - c. [Restart](#) the FME Flow Database and FME Flow Engines services.

13. Ensure that the user account that runs `<engineHost>` has the necessary permissions to access the FME Flow System Share and the FME Flow install directory. For more information, see [Directory and Account Permissions](#).
14. On the `<coreHost>` machine, open configuration file [fmeFlowConfig.txt](#), and update the FME_SERVER_HOST_NAME parameter to the fully-qualified domain name or IP address of the `<coreHost>` machine.
15. [Restart](#) the FME Flow Core service.

Verify the Installation

The installation on `<engineHost>` starts two FME Engines by default. The FME Engines start and register with the FME Flow Core running on `<coreHost>`. All licensed engines are listed on the [Engines](#) page of the Web User Interface.

 **Note** You may have to disable any firewalls that are running on this computer.

You can also use the Engines & Licensing page to perform the following tasks:

- To license FME Engines, see [Licensing](#).
- If you want this computer to start more or less FME Engines, see [Configuring the Number of FME Engines to Start](#).

For more information, see [Verify the Installation](#).

Performing a Silent Installation (Windows)

To perform a silent installation of FME Flow, extract and run an installer from a Windows command line, and override any default installation properties to customize the installation.

To obtain the installer, see [Obtain the Installer](#).

Extracting Installation Files

Installation files, which consist of a .msi file and .cab files, must be extracted before proceeding with installation. You can extract installation files in either of the following ways:

- Manual extraction. Following extraction, invoke the .msi file to perform a silent install.
- Invoke the installation .exe directly, and use the following parameters to control extraction and installation:

Flag	Description
- d< <i>path</i> >	Set the destination folder for unpacked files.
-s, -s1, -s2	Silent mode. Switch -s is a brief form of -s1. Both -s and -s1 hide the main dialog and extraction progress. -s2 hides the start dialog, but extraction progress is displayed.
- sp< <i>par</i> >	Installer Flags to pass to the extracted .msi installer when it is run.


Installer Flags

The following .msi-specific flags can be passed to the installer:

Flag	Description
/q[n b]	Signifies a quiet installation. /qn means quiet with no user interface, and /qb means a quiet installation with a basic progress bar.
/norestart	Do not restart the computer after installation.
/l*v <logfile>	Enable logging to the <logfile> specified.

Specifying FME Flow Components to Install

To select which components of FME Flow to install, as in a [Distributed](#) installation, use the `ADDLOCAL` property, and specify components in a comma-delimited list. If this property is not specified, all components are installed, as in an [Express](#) installation.

 **Note** Following a distributed or fault tolerant installation, the FME Flow system services do not start automatically. You must [manually start](#) them.

Feature	Value
FME Flow Core	FMEFlowCore
Web Services	Services
FME Engine	FMEEngine
FME Flow Database (on PostgreSQL)	FMEFlowDatabase

Example


To install the FME Flow Core and FME Engine services:

```
ADDLOCAL=FMEFlowCore,FMEEngine
```


Installation Properties

The following table lists the installation properties that you can set in advance, and their default values if left unset. These properties correspond to properties that can be adjusted in the user dialogs of a standard installation.


Property	Default Value	Description
EXTRACTONLY		Extracts the .msi installation file, but does not run the installer. A C++ runtime package and MSSQL driver only are installed.
COREHOSTNAME	<code>localhost</code> if an Express installation; otherwise, the machine name on which the installer is running.	Host name of the machine that runs the FME Flow Core .

Property	Default Value	Description
INSTALLDIR	C:\Program Files\FMEFlow or C:\apps\FMEFlow	<p>The FME Flow install directory. If you change this value the path must end in "\". For example, INSTALLDIR=C:\apps\FMEFlow\</p> <div data-bbox="1019 630 1461 1711" style="border: 1px solid #0070C0; padding: 10px; background-color: #E6F2FF;"> <p> Note You may require quotes to escape spaces in paths. From the Windows Command Prompt, use eight quotes. For example, """"""C:\Program Data\Safe Software\FME Flow"""""". From Windows PowerShell, use sixteen quotes. For example, """"""""""C:\Program Files\FMEFlow"""""".".</p> </div>



Property	Default Value	Description
FMEFLOWUSER	local system account	<p>The Windows user account that will run the FME Flow System Services. This account must have:</p> <ul style="list-style-type: none"> • Read/Write access to the FME Flow System Share, where FME Flow stores Repositories and Resources files. (For more information, see Provide a Remote File System for the FME Flow System Share.) • Read/Write access to the FME Flow installation directory. • "Log on as a service" rights on this machine. For more information, see this Microsoft Docs article.
FMEFLOWUSERPASSWORD		If FMEFLOWUSER is specified, the user account password.

Property	Default Value	Description
FMEFLOWSHAREDDAT A	<code>C:\ProgramData\Safe Software\FMEFlow</code>	<p>The path to the FME Flow System Share directory, where FME Flow stores Repositories and Resources files. Path names are case-sensitive.</p> <div data-bbox="1019 569 1463 1654" style="border: 1px solid #0070C0; border-radius: 10px; padding: 10px; background-color: #E6F2FF;"> <p> Note You may require quotes to escape spaces in paths. From the Windows Command Prompt, use eight quotes. For example, <code>""""""C:\ProgramData\SafeSoftware\FMEFlow""""""</code>. From Windows PowerShell, use sixteen quotes. For example, <code>""""""""""C:\ProgramFiles\FMEFlow""""""""</code>.</p> </div>

Property	Default Value	Description
NODENAME	The machine name on which the installer is running	If you are installing an FME Core or FME Engines, use this property to advertise a fully-qualified domain name (FQDN) for how other distributed components connect to this COREHOSTNAME. Specify a FQDN that resolves to the IP address of the machine via DNS.
OVERRIDE_NETWORK_CHECK		If Yes , the installer does not ensure the path specified for FMEFLOWSHAREDDATA exists and is writable by the user account specified for FMEFLOWUSER.

Property	Default Value	Description
DATABASETYPE	Default	<p>The database server type that will be used for the FME Flow Database. Possible values are <code>Default</code>, <code>MSSQL</code>, <code>Oracle</code>, <code>PostgreSQL</code>. <code>Default</code> assumes the PostgreSQL database server that is included with the FME Flow installer and located on the same machine.</p> <div data-bbox="1019 856 1463 1423" style="border-left: 2px solid #0056b3; border-right: 2px solid #0056b3; border-bottom: 2px solid #0056b3; padding: 10px; background-color: #e6f2ff;"> <p> Note PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.</p> </div>
DATABASEHOST	[COREHOSTNAME]	<p>The host that the database is on. Specify only if <code>DATABASETYPE</code> is not <code>Default</code>.</p>

Property	Default Value	Description
DATABASEPORT	MSSQL: 1433 PostgreSQL: 5432 Oracle: 1521	The port to connect to the database. Specify only if DATABASETYPE is not Default .
INSTANCENAME		If DATABASETYPE is MSSQL , the SQL Server instance name to use for connecting to the database, instead of DATABASEPORT.
DATABASECONNECTIONSTRING	A connection string is generated based on the values of DATABASETYPE, DATABASEHOST, DATABASEPORT, and (if specified) INSTANCENAME.	The JDBC connection string for connecting to the database.
ORACLESID		The SID for the Oracle database. Must be specified only if DATABASETYPE is Oracle .
JDBCDRIVER		If your database requires a JDBC driver that Safe does not provide, specify the full path to it here, and it will be copied to the appropriate place in the install directory.

Property	Default Value	Description
DATABASEUSER	fmeflow	<p>The FME Flow Database user account name. This property applies regardless of the value specified by <code>DATABASETYPE</code>.</p> <div data-bbox="1019 512 1463 1251" style="border: 1px solid #ccc; padding: 10px; background-color: #fff9e6;"> <p> Warning If <code>DATABASETYPE</code> is <code>Default</code>, do not specify <code>postgres</code>. This name corresponds to the PostgreSQL master user. If the corresponding passwords do not match, the database will not install.</p> </div>
DATABASEPASSWORD		<p>The password for the FME Flow Database user name.</p> <div data-bbox="1019 1509 1463 1902" style="border: 1px solid #ccc; padding: 10px; background-color: #e6f2ff;"> <p> Note The password for the FME Flow Database user account is encrypted in FME Flow installation files.</p> </div>

Property	Default Value	Description
PGDATA	C:\ProgramData\Safe Software\FMEFlow	If <code>DATABASETYPE=Default</code> , the directory to contain the data for the FME Flow PostgreSQL database.
SERVLETTYPE	Default	The type of Application Server (servlet engine) you will use with FME Flow. Possible values are <code>Default</code> or <code>Apache (Tomcat)</code> . <code>Default</code> installs the Apache Tomcat servlet provided with the FME Flow installer.
EXTERNALHOSTNAME	[COREHOSTNAME]	The hostname on which the Web Application Server is installed, or the hostname of the load balancer serving the Web Application Server, if applicable.
SERVLETPORT	80 (if open)	The port that the Web Application Server host uses or that the built-in Apache Tomcat server uses. If not specified, the installer checks ports 80, 8080-8085 and uses the first one that is not in use.

Property	Default Value	Description
EXTERNALPORT	[SERVLETPORT]	If applicable, a load balancer port or port that is otherwise different from the SERVLETPORT on which the Web Application Server is listening.
WEBAPPSDIR	[INSTALLDIR]\Utilities\tomcat\webapps	If an Apache Tomcat servlet is already installed on the machine, the path to the webapps directory for the servlet engine, for installing WAR files. This is a necessary property for distributed web applications.
SHAREDTCOMCATLIB	[INSTALLDIR]\Utilities\tomcat\lib	If an Apache Tomcat servlet is already installed on the machine, the path to the lib directory for the servlet engine, for installing JAR files. This is a necessary property for distributed web applications.
NOLOGPREFIX	No	If Yes , FME Flow log file names are not prefixed with the server host name.

Property	Default Value	Description
FIRSTLOGINCHANGEPASSWORD	true	If <code>true</code> , the user will be prompted to update the <code>admin</code> password upon initial login to the FME Flow Web User Interface following installation. If <code>false</code> , updating the <code>admin</code> password will not be required upon initial login.

Post-Install Tasks

Distributed/Fault Tolerant Installations

If you performed a distributed or fault tolerant installation (specifying the `ADDLOCAL` property), the FME Flow system services do not start automatically. You must [manually start](#) them.

MSSQL Formats

If you plan to run workspaces that reference Microsoft SQL Server and Azure SQL Database (MSSQL_*) readers and writers, you must manually install the SQL Server Native Client and Microsoft System CLR Types for SQL Server on all machines that host FME Engines. To install, navigate to the [INSTALLDIR] and run the following:

```
msiexec /i sqlncli.msi /qb
msiexec /i SQLSysClrTypes.msi /qb
```

Examples

The following examples assume you are running a command line from a directory that contains an installer named `fme-flow-win-x64.exe` to extract and install, or `fme-flow.msi` to install following manual extraction.

Invoking the .exe Installer to Extract and Install

Examples

Extract installer to C:\FME\installer and install with all default options (similar to Express install)

```
fme-flow-win-x64.exe -d"C:\FME\installer" -s -sp"/qb /norestart"
```

In Windows PowerShell:

```
start-process -FilePath <downloadpath>\fme-flow-win-x64.exe -  
Argumentlist '-dC:\FME\installer','-s','-sp"/qb /norestart''
```

Extract installer to C:\FME\installer and install core only with an FME Flow System Share directory on a share on another machine named "\\machine1\share"

```
fme-flow-win-x64.exe -d"C:\FME\installer" -s -sp"ADDLOCAL=FMEFlowCore  
FMEFLOWUSER=MYCOMPANY\user1 FMEFLOWUSERPASSWORD=password  
FMEFLOWSHAREDDATA=\\machine1\share /qb /norestart"
```

In Windows PowerShell:

```
start-process -FilePath <downloadpath>\fme-flow-win-x64.exe -  
Argumentlist '-dC:\FME\installer','-s','-sp"ADDLOCAL=FMEFlowCore  
FMEFLOWUSER=MYCOMPANY\user1 FMEFLOWUSERPASSWORD=password  
FMEFLOWSHAREDDATA=\\machine1\share /qb /norestart''
```

Invoking the .msi File to Install, Following Manual Extraction

Examples

Install with all default options (similar to Express install), with logging enabled

```
msiexec /i fme-flow.msi /qb /norestart /l*v install\FMEFlowLog.txt
```

Install core only with an FME Flow System Share directory on a share on another machine named "\\machine1\share"

```
msiexec /i fme-flow.msi ADDLOCAL=FMEFlowCore FMEFLOWUSER=MYCOMPANY\user1  
FMEFLOWUSERPASSWORD=password FMEFLOWSHAREDDATA=\\machine1\share" /qb  
/norestart /l*v install\FMEFlowLog.txt
```


Express install, but with an MSSQL database located on another server

```
msiexec /i fme-flow.msi DATABASETYPE=MSSQL DATABASEHOST=DatabaseMachine  
DATABASEPORT=1433 /qb /norestart /l*v installFMEFlowLog.txt
```

FME Engine-only install, with all other FME Flow components on another server named
"\\machine1\FMEFlowSystemShare"

```
msiexec /i fme-flow.msi /l*v "install.log" /qb /norestart  
INSTALLDIR="C:\Program Files\FMEFlow\" FMEFLOWUSER=MYCOMPANY\user1  
FMEFLOWUSERPASSWORD=password COREHOSTNAME="machine1" ADDLOCAL=FMEEngine  
FMEFLOWSHAREDDATA="\\machine1\FMEFlowSystemShare\  
DATABASETYPE="PostGreSQL" DATABASEHOST="FMEFlowCore" DATABASEPORT="7082"  
SERVLETPORT="80"
```

Performing a Silent Installation (Linux)

Silent installation of FME Flow on Linux is controlled by an `install.cfg` file that is passed to the installer. This file contains a list of properties that control different options in the installer. The command for silent installation is:

```
./<filename> -- --file install.cfg
```

Example `<filename>`: `fme-flow-2023.0.0.3-b23319-linux-x64~ubuntu.20.04.run`

Properties are listed in the `install.cfg` file in the form:

```
<property>=<value>
```

To obtain the installer, see [Obtain the Installer](#).

Specifying FME Flow Components to Install

To select which components of FME Flow to install, use the following set of properties.

These properties can be set to either "Yes" or "No" (case-sensitive). The default values are "Yes".

Feature	Property
FME Flow Core	FEATURE_FMEFlowCore_INSTALL
Web Services	FEATURE_Services_INSTALL
FME Engine	FEATURE_FMEEngine_INSTALL
FME Flow Database (on PostgreSQL)	FEATURE_FMEFlowDatabase_INSTALL

Installation Properties

The following table lists the installation properties that you can specify in the `install.cfg` file, and their default values if left unset. These properties correspond to properties that can be adjusted in the user dialogs of a standard installation.

Property	Default Value	Description
COREHOSTNAME	<code>localhost</code> if an Express installation; otherwise, the machine name on which the installer is running.	Host name of the machine that runs the FME Flow Core .
INSTALLDIR	<code>/opt/fmeflow</code>	The FME Flow install directory.
FMEFLOWSHAREDDATA	[INSTALLDIR]	The path to the FME Flow System Share directory, where FME Flow stores Repositories and Resources files. Path names are case-sensitive.

Property	Default Value	Description
NODENAME	The machine name on which the installer is running	If you are installing an FME Core or FME Engines, use this property to advertise a fully-qualified domain name (FQDN) for how other distributed components connect to this COREHOSTNAME. Specify a FQDN that resolves to the IP address of the machine via DNS.


Property	Default Value	Description
DATABASETYPE	Default	<p>The database server type that will be used for the FME Flow Database. Possible values are <code>Default</code>, <code>MSSQL</code>, <code>Oracle</code>, <code>PostgreSQL</code>. <code>Default</code> assumes the PostgreSQL database that is included with the FME Flow installer and located on the same machine.</p>

Property	Default Value	Description
		<div data-bbox="1166 243 1463 1329" style="border: 1px solid #005596; padding: 10px; background-color: #e6f2ff;"> <p> Note</p> <p>PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.</p> </div>
DATABASEHOST	[COREHOSTNAME]	The host that the database is on. This should only be specified if DATABASETYPE is not Default.

Property	Default Value	Description
DATABASEPORT	MSSQL: 1433 PostgreSQL: 5432 Oracle: 1521	The port to connect to the database. This should only be specified if DATABASETYPE is not Default.
INSTANCENAME		If DATABASETYPE is MSSQL , the SQL Server instance name to use for connecting to the database, instead of DATABASEPORT.
DATABASECONNECTIONSTRING	A connection string is generated based on the values of DATABASETYPE, DATABASEHOST, DATABASEPORT, and (if specified) INSTANCENAME.	The JDBC connection string for connecting to the FME Flow Database.
ORACLESID		The SID for the Oracle database. Must be specified only if DATABASETYPE is Oracle.

Property	Default Value	Description
JDBCDRIVER		If your database requires a JDBC driver that Safe does not provide, specify the full path to it here, and it will be copied to the appropriate place in the install directory.

Property	Default Value	Description
DATABASEUSER	fmeflow	<p>The FME Flow Database user account name. This property applies regardless of the value specified by <code>DATABASETYPE</code>.</p> <div data-bbox="1166 684 1463 1885" style="border: 1px solid orange; padding: 10px;"> <p>Warning If <code>DATABASETYPE</code> is <code>Default</code>, do not specify <code>postgres</code>. This name corresponds to the PostgreSQL master user. If the corresponding passwords do not match, the database will not install.</p> </div>


Property	Default Value	Description
DATABASEPASSWORD		<p>The password for the FME Flow Database user name.</p> <div data-bbox="1166 512 1463 1194" style="border: 1px solid #0056b3; padding: 10px; background-color: #e6f2ff;"> <p> Note The password for the FME Flow Database user account is encrypted in FME Flow installation files.</p> </div>
PGDATADIR	/opt/fmeflow//Utilities/pgsql/data	<p>If <code>DATABASETYPE=Default</code>, the directory to contain the data for the FME Flow PostgreSQL database.</p>

Property	Default Value	Description
SERVLETTYPE	Default	The type of Application Server (servlet engine) you will use with FME Flow. Possible values are Default or Apache (Tomcat). Default installs the Apache Tomcat servlet provided with the FME Flow installer.
EXTERNALHOSTNAME	[COREHOSTNAME]	The hostname on which the Web Application Server is installed, or the hostname of the load balancer serving the Web Application Server, if applicable.
NGINXPORT	80	If SERVLETTYPE is Default , the listening port of the NGINX reverse proxy.

Property	Default Value	Description
SERVLETPORT	8080	If SERVLETTYPE is Default , the listening port of the Apache Tomcat servlet engine host. Must be different from EXTERNALPORT.
EXTERNALPORT	[SERVLETPORT]	If applicable, a load balancer port or port that is otherwise different from the SERVLETPORT on which the Web Application Server is listening.

Property	Default Value	Description
WEBAPPSDIR	[INSTALLDIR]\Utilities\tomcat\webapps	<p>If an Apache Tomcat servlet is already installed on the machine, the path to the webapps directory for the servlet engine, for installing WAR files. This is a necessary property for distributed web applications.</p>
SHAREDTCOMCATLIB	[INSTALLDIR]\Utilities\tomcat\lib	<p>If an Apache Tomcat servlet is already installed on the machine, the path to the lib directory for the servlet engine, for installing JAR files. This is a necessary property for distributed web applications.</p>

Property	Default Value	Description
CONFIGSELINUX	Yes	If SERVLETYPE is "Default" and installation is on a Red Hat or Rocky Linux distribution, the SELinux module must be configured to work with the NGINX reverse proxy.

Property	Default Value	Description
		<div data-bbox="1166 243 1463 1444" style="border: 1px solid #005596; border-radius: 10px; padding: 10px; background-color: #e6f2ff;"> <p> Note The NGINX reverse proxy allows easy SSL configuration with the Apache Tomcat web application server, and the ability to choose ports under 1024 without root permission.</p> </div> <p>To configure SELinux, specify Yes. Else, No.</p>

Property	Default Value	Description
NOLOGPREFIX	No	If Yes, FME Flow log file names are not prefixed with the server host name.
FIRSTLOGINCHANGEPASSWORD	true	If true , the user will be prompted to update the admin password upon initial login to the FME Flow Web User Interface following installation. If false , updating the admin password will not be required upon initial login.

Post-Install: MSSQL Formats Only

If you plan to run workspaces that reference Microsoft SQL Server and Azure SQL Database (MSSQL_*) readers and writers, you must manually install the SQL Server Native Client and Microsoft System CLR Types for SQL Server on all machines that host FME Engines. To install, navigate to the [INSTALLDIR] and run the following:

```
msiexec /i sqlncli.msi /qb
msiexec /i SQLSysClrTypes.msi /qb
```


Examples

Install with all default options (similar to Express install), to
/home/user/fmeflow

```
INSTALLDIR="/home/user/fmeflow"
```

Install the FME Flow Core only with an FME Flow System Share directory
on a share on another machine named "\\machine1\share"

```
FMEFLOWSHAREDDATA="/mnt/machine1/share"
```

```
FEATURE_FMEFlowCore_INSTALL="Yes"
```

```
FEATURE_FMEEngine_INSTALL="No"
```

```
FEATURE_Services_INSTALL="No"
```

Express install, but with an MSSQL database located on another server

```
DATABASETYPE=MSSQL
```

```
DATABASEHOST=DatabaseMachine
```

```
DATABASEPORT=1433
```

Install the Web Services feature only with an Apache Tomcat Servlet
Engine:

```
FMEFLOWSHAREDDATA="//CoreMachine\share"
```

```
SERVLETTYPE=Apache
```

```
SERVLETPORT=8080
```

```
WEBAPPSDIR="C:\Program Files\tomcat\webapps"
```

```
FEATURE_FMEFlowCore_INSTALL="No"
```

```
FEATURE_FMEEngine_INSTALL="No"
```

```
FEATURE_Services_INSTALL="Yes"
```

FME Engine-only install, with all other FME Flow components on
another server named "machine1"

```
COREHOSTNAME="machine1"
```

```
INSTALLDIR="/home/user/FMEFlow"
```

```
FMEFLOWSHAREDDATA="\\machine1\FMEFlowSystemShare"  
DATABASETYPE="PostgreSQL"  
DATABASEHOST="FMEFlowCore"  
DATABASEPORT="7082"  SERVLETPORT="80"  
FEATURE_FMEFlowCore_INSTALL="No"  
FEATURE_FMEEngine_INSTALL="Yes"  
FEATURE_Services_INSTALL="No"
```

Deploying FME Flow with Docker Compose

As a multi-container application, FME Flow supports deployment with Docker Compose. We recommend deploying FME Flow this way only if you have worked with Docker before and understand its architecture.

Limitations and Considerations

When deciding if you want to deploy FME Flow with Docker Compose, keep in mind the following:

- Container deployments of FME Flow are built on Linux. Review our [FME formats matrix](#) and compare by platforms to see if the formats you plan to use are available on Linux. In particular:
 - Certain Esri formats cannot be used in a Docker deployment due to operating system compatibility. For integrating Esri formats with FME Flow, we recommend installing FME Flow on Windows machines and not using containers. Esri integrations with ArcPy/ArcGIS Pro require Windows libraries. ArcGIS Server for Linux is not a usable alternative.
 - Database formats (such as SQL Server) require the JDBC version of the format, which may require workspace modification if publishing from FME Form on Windows.
 - To use third-party libraries, JDBC drivers, and similar requires creating your own FME Flow engine container image.

- FME Flow deployed with Docker Compose is supported on Oracle Linux 7 only if the default Unbreakable Enterprise Kernel is used.
- [Adding FME Engines on a Separate Machine](#) that is not containerized is not tested, supported, or recommended by Safe Software. It is not possible to use this architecture to add FME Engines on a Windows VM to a Docker deployment of FME Flow for format compatibility.

About the Docker Compose File for FME Flow

Services

The Compose file defines containers for each of the services that comprise FME Flow. These service names can be specified to start, stop, scale, or run other commands on a specific container. They are also used as the host names for the containers to communicate with each other.

Service	Description
fmeflowcore	FME Flow Core
fmeflowwebsocket	WebSocket Server
fmeflowweb	FME Flow Web Services
fmeflowdb	FME Flow Database
fmeflowengine	FME Engines
fmeflowqueue	Job Queues <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p>Note The preferred method for assigning FME Engines to queues is through the ENGINEPROPERTIES FME Engine Container environment variable.</p> </div>
nginx	Reverse proxy for FME Flow; receives all web traffic and forwards it to the right container. Also responsible for handling the https connection.

See Also

- [Working with FME Flow Container Environment Variables](#)

Ports

The FME Flow deployment maps three ports to the host machine:

- nginx maps port 80 and 443 for the [FME Flow Web User Interface](#). (See [To Disable https](#) (below) for more details.)
- fmeflowcore maps port 25 for [SMTP relay](#).

Once FME Flow is deployed, any requests to these ports on the host machine are routed to the proper container. This means you can access the FME Flow Web User Interface on port 443 of your host machine even though the `fme-flow-web` service is actually running inside a container.

Volumes

The Compose file defines two main volumes:

- Database - Where the `fme-flow-database` service stores its database files.
- FME Flow System Share - Actually comprised of several volumes that restrict access to certain folders from containers that do not require it.

Deploying and Administering FME Flow on Docker Compose

Download the Docker Compose File for FME Flow

Docker Compose files are available at <https://www.safe.com/support/downloads/> for each available tag (version) of the FME Flow container deployment.

Working with the `docker-compose` Command

Use the `docker-compose` command to create, start, and stop FME Flow. This command looks in the current directory for a file named `docker-compose.yml`, and uses that for the definition of what to deploy. Alternatively, you can pass in a path to a compose file using the `-f` parameter. Example commands below assume that a `docker-compose.yml` exists in the current working directory.

To Launch a New FME Flow Stack

Use the `up` parameter to bring up a new set of containers as defined in `docker-compose.yml`. Pass in the `-d` parameter to run in the background.

```
docker-compose up -d
```

Docker Compose creates the networks, volumes and containers defined in the Compose file. When complete, you can connect to the [FME Flow Web User Interface](#) on `https://localhost/`. The administrator user name and password defaults to `admin` and `admin`, respectively. You can change this on the [Users](#) page.

To Launch a New FME Flow Stack with a Dynamic Engine (CPU Usage) License

```
docker-compose -f docker-compose.yaml -f docker-compose-dynamic-engine.yaml up -d
```

To Stop a Running FME Flow

```
docker-compose stop
```

To Start FME Flow

```
docker-compose start
```

To Restart FME Flow

```
docker-compose restart
```

To Specify Services in the Compose File

You can specify the [Services](#) defined in the Compose file to start and stop. For example, to stop just the FME Flow Web Services container:

```
docker-compose stop fmeflowweb
```

To Scale FME Engines

You can scale the number of running engines by launching additional FME Engines containers. (For more information, see [Configuring the Number of FME Engines to Start](#).)

Use the `scale` command to specify the number of FME Engines you want after the command executes. The following command scales the number of engines to 4:

```
docker-compose up --scale fmeflowengine=4
```

To Remove an FME Flow Deployment

The `down` command destroys all containers, volumes and networks that were created in the stack. The `-v` parameter removes the volumes as well.

```
docker-compose down -v
```

Without the `-v` parameter, the Database and FME Flow System Share volumes are not destroyed, and if the same stack were launched again using `up`, no data would be lost.

To Upgrade an FME Flow Deployment

 **Note** See also [Upgrading to a Patch Build in Docker](#).

1. Make a local backup of FME Flow on the [Backup & Restore](#) page of the Web User Interface. Make sure this file is saved somewhere outside of the FME Flow [Resources](#) folders, as those will be removed.
2. Remove the containers, volumes and networks.

```
docker-compose down -v
```
3. Download the latest version of the containers.

```
docker-compose pull
```
4. Deploy the new version of the containers.

```
docker-compose up -d
```
5. Restore your backup on the [Backup & Restore](#) page of the Web User Interface by uploading the backup file you saved in step 1.

To Run Multiple FME Flows on One Machine

While not recommended for production machines, running multiple FME Flow stacks on a single machine may be useful if you are prototyping or testing workflows on multiple versions of FME Flow.

When running multiple FME Flows on the same machine, keep in mind the following:

- When launching a stack, `docker-compose` prepends a project name, separated by an underscore, to all the containers, volumes and networks. The project name defaults to the name of the working directory when the stack is created. In the image below, the stack was launched in a directory named `server`, so `server_` is prepended to all the resources created:

```
Creating network "server_web" with driver "bridge"
Creating network "server_database" with driver "bridge"
Creating volume "server_engine" with local driver
Creating volume "server_key" with local driver
Creating volume "server_logs" with local driver
Creating volume "server_temp" with local driver
Creating volume "server_subscriptions" with local driver
Creating volume "server_database" with local driver
Creating volume "server_repositories" with local driver
Creating volume "server_dashboards" with local driver
Creating volume "server_localization" with local driver
Creating volume "server_upload" with local driver
Creating volume "server_requestdata" with local driver
Creating volume "server_engineresults" with local driver
Creating volume "server_emailattachments" with local driver
Creating volume "server_licenses" with local driver
Creating volume "server_tomcat" with local driver
Creating volume "server_backups" with local driver
Creating volume "server_data" with local driver
Creating server_fmserverdb_1
Creating server_fmservercore_1
Creating server_fmserverweb_1
Creating server_fmserverengine_1
```

If the docker-compose.yml files for each deployment are not in directories with different names, you can pass in a project name when launching the stack with the `-p` flag:

```
docker-compose -p projectname up -d
```

- Since a few [Ports](#) are mapped to the host machine, the same port cannot be used twice. You must either make sure only one is started at a time, or change the port mappings. For example, to use port 8080 instead of 443, delete the redirect port 80:8080 from the fmefflowweb container with this command:

```
EXTERNALPORT=8080 docker-compose up -d
```

This sets the external port on all relevant containers to 8080 and brings up FME Flow.

To Disable https

To run FME Flow without https:

1. Remove the exposed port 80 from the nginx container in the Docker Compose file (line 80:8080).

2. Start the FME Flow with `WEBPROTOCOL=http` and `EXTERNALPORT=80` variables. For example:

```
WEBPROTOCOL=http EXTERNALPORT=80 docker-compose up -d
```

To Use a Custom TLS/SSL Certificate

⚠ Warning Ensure a copy of the certificate is stored in a different location before mounting it in the nginx container. The code in the container may have untested "edge" cases that overwrite the certificate.

To use a valid SSL certificate, make the following files available in a directory on the FME Flow host:

- `nginx.crt`: The certificate, including full certificate chain, if necessary.
- `nginx.key`: Private key used to generate the certificate.

Once provided, pass in the location to the certificate and key by setting the `SSLCERTLOCATION=/path/to/certificates` variable.

See Also

- [Working with FME Flow Container Environment Variables](#)
- [Upgrading to a Patch Build in Docker](#)

Working with FME Flow Container Environment Variables

To perform additional configuration of an FME Flow deployment with Docker Compose, define an environment variable section under the relevant container service. Each service has a different set of environment variables that can be defined.

FME Flow Web Services Container

Environment Variable	Default Value	Description
EXTERNALHOSTNAME	localhost	The host name that users will use to connect to this FME Flow. This is commonly the hostname or IP address of the physical machine on which the Docker deployment is running, or the external endpoint of the load balancer. This variable is used for Topic Monitoring . If you do not need to use Topic Monitoring, leave as <code>localhost</code> .
EXTERNALPORT	80 (Windows) 8080 (Linux)	The port used to connect to the FME Flow Web User Interface . This should be the port exposed on the host or load balancer that users will use to connect.
FMEFLOWHOSTNAME	fmeflowcore	The service name or host name of the FME Flow Core container.
WEBSERVERHOSTNAME	fmeflowweb	The service name or host name of the FME Flow Web Services container.
WEBPROTOCOL	https	The protocol FME Flow uses. Can be either <code>http</code> or <code>https</code> .

Environment Variable	Default Value	Description
WEBSOCKETHOSTNAME	fmeflowwebsocket	The service name or host name of the WebSocket server container.
LOGPREFIX	The host name of the container	The prefix to use on the log files created by this container.
PGSQLHOSTNAME	fmeflowdb	The service name or host name of the FME Flow PostgreSQL database. If you want to use a different database type, see Changing the Database Connection , below.
PGSQLPORT	5432	The port for the FME Flow PostgreSQL database. If you want to use a different database type, see Changing the Database Connection , below.
POSTGRESQSSL	false	Set to <code>true</code> to force the PostgreSQL connection to use SSL. This adds the option <code>sslrequire=true</code> to the JDBC connection strings.
PGSQLUSERNAME	fmeflow	The user to create as part of the FME Flow schema that FME Flow will use to connect to the database.

FME Engine Container

Environment Variable	Default Value	Description
ENGINEPROPERTIES		A comma-delimited list of properties to set on this engine. Engine properties are useful when assigning FME Engines to queues .
EXTERNALHOSTNAME	localhost	The host name that users will use to connect to this FME Flow. This is commonly the hostname or IP address of the physical machine on which the Docker deployment is running. This is used for the data download results URL that is returned after running a job in the Data Download Service . If you don't use the Data Download service, leave as <code>localhost</code> .
EXTERNALPORT	443	Port used to connect to FME Flow.
FMEFLOWHOSTNAME	fmeflowcore	The service name or host name of the FME Flow Core container.
PGSQLHOSTNAME	fmeflowdb	The service name or host name of the FME Flow PostgreSQL database. If you want to use a different database type, see Changing the Database Connection , below.

Environment Variable	Default Value	Description
PGSQLPORT	5432	The port for the FME Flow PostgreSQL database. If you want to use a different database type, see Changing the Database Connection , below.
POSTGRESQSSL	false	Set to <code>true</code> to force the PostgreSQL connection to use SSL. This adds the option <code>sslrequire=true</code> to the JDBC connection strings.
PGSQLUSERNAME	fmeflow	The user to create as part of the FME Flow schema that FME Flow will use to connect to the database.
WEBPROTOCOL	https	The protocol FME Flow uses. Can be either <code>http</code> or <code>https</code> .
LOGPREFIX	The host name of the container	The prefix to use on the log files created by this container.
FME_INSTANCE_NAME	The host name of the container	The name of the engine that appears on the Engines page.
NODENAME	The host name of the container	The host name of the engine that the Core container uses to communicate with this engine.

FME Flow Core Container

Environment Variable	Default Value	Description
FMEFLOWHOSTNAME	fmeflowcore	The service name or host name of the FME Flow Core container.
PGSQLHOSTNAME	fmeflowdb	The service name or host name of the FME Flow PostgreSQL database. To use a different database type, see Changing the Database Connection , below.
PGSQLPORT	5432	The port for the FME Flow PostgreSQL database. To use a different database type, see Changing the Database Connection , below.
POSTGRESQSSL	false	Set to <code>true</code> to force the PostgreSQL connection to use SSL. This adds the option <code>sslrequire=true</code> to the JDBC connection strings.
PGSQLUSERNAME	fmeflow	The user to create as part of the FME Flow schema that FME Flow will use to connect to the database.

Environment Variable	Default Value	Description
WEBSERVERHOSTNAME	fmeflowweb	The service name or host name of the FME Flow Web Services container.
REDISHOSTS	fmeflowqueue	The service name or host name of the Memurai queue.
REDISPORT	6379	The port for the Memurai queue.

Environment Variable	Default Value	Description
PORTPOOL		<p>This setting is required only if FME Engines that are not running in Docker are connecting to this FME Flow Core. If engines are connecting from outside the docker network, perform the following:</p> <ol style="list-style-type: none"> 1. Open additional ports in the Compose file for the engines to connect. 2. Set this PORTPOOL variable to a port range, such as <code>PORTPOOL=9000-9200</code>. 3. Map the specified ports to the physical host in the Compose file under the <code>ports</code> section. For example: <code>9000-9200:9000-9200</code>.

Environment Variable	Default Value	Description
FIRSTLOGINCHANGEPASSWORD	true	If <code>true</code> , the user will be prompted to update the <code>admin</code> password upon initial login to the FME Flow Web User Interface following installation. If <code>false</code> , updating the <code>admin</code> password will not be required upon initial login.
MAX_TRANSACTION_RESULT_SUCCESSES		Specifies the number of successful translations until an engine process should be restarted.
MAX_TRANSACTION_RESULT_FAILURES		Specifies the number of failed translations until an engine process should be restarted.


Environment Variable	Default Value	Description
PRIMARY_PROCESS	core	Defines the purpose of this container. Valid values are <code>core</code> , <code>websocket</code> , or <code>initpgsql</code> . Setting to <code>websocket</code> starts only the websocket process. Setting to <code>initpgsql</code> runs the database initialization scripts against the configured database. For more information, see Initializing the PostgreSQL Database , below.

NGINX Container

Environment Variable	Default Value	Description
WEBPROTOCOL	https	The protocol FME Flow uses. Can be either <code>http</code> or <code>https</code> .
EXTERNALHOSTNAME	localhost	The host name that users will use to connect to this FME Flow. This is commonly the hostname or IP address of the physical machine on which the Docker deployment is running. This variable is used for Topic Monitoring . If you do not need to use Topic Monitoring, leave as <code>localhost</code> .
EXTERNALPORT	443	The port used to connect to FME Flow.

Environment Variable	Default Value	Description
NGINX_WORKER_PROCESSES	1	Number of NGINX workers to start.

Changing the Database Connection

 **Note** To configure FME Flow to connect to a Microsoft SQL Server or an Oracle database, [contact Safe Support](#).

Initializing the PostgreSQL Database

When a new stack is launched, a mode of the core container populates an empty PostgreSQL database with the FME Flow database schema before starting FME Flow. If you want to use an existing PostgreSQL database for your FME Flow, you must configure an initialization container using the following environment variables.

For example, the following command populates the schema in an existing PostgreSQL database running on host my-psgql-hostname, and then exits.

```
docker run -e PGSQHOSTNAME=my-psgql-hostname -e PGSQPORT=5432 -e PGSQADMINUSER=postgres -e PGSQADMINPASSWORD=postgres -e PRIMARY_PROCESS=initpgsql safesoftware/fmeflow-core:2024-latest
```

Environment Variable	Default Value	Description
PGSQLHOSTNAME	fmeflowdb	The host name of the PostgreSQL database that is to be initialized.
PGSQLPORT	5432	The port to connect to the PostgreSQL database.
PGSQLADMINUSER	postgres	A user that exists on the PostgreSQL database that has permissions to create a database.
PGSQLADMINPASSWORD		The password for the PGSQLADMINUSER.
PGSQLADMINDATABASE	Same value as PGSQLADMINUSER	Set this to the database to which to connect initially as the PGSQLADMINUSER to initialize the database schema on the fmeflowdbinit service.
PGSQLUSERNAME	fmeflow	The user to create as part of the FME Flow schema that FME Flow will use to connect to the database.
PRIMARY_PROCESS		Set to <code>initpgsql</code> .

Environment Variable	Default Value	Description
AZUREPOSTGRESQL	false	Set to <code>true</code> on the <code>fmeflowdbinit</code> service in the Docker Compose file when using an Azure database for PostgreSQL to ensure the connection string is formatted correctly.

Running the Containers as a Different User

By default, the FME Flow containers run as a user named `fmeflow` in group `fmeflow`, with UID and GID 1363. You may want the UID and GID of the user in the container to match a user defined on the host machine. For example, you may want to use a host bind mount for the FME Flow System Share volume instead of using Docker volumes. If mounting a host directory into the container, it is optimal if the permissions in that folder match a real user on the host.

To match the user and group running a container with a user defined on the host machine, pass in variables `FMEFLOW_UID` and `FMEFLOW_GID`, set to the UID and GID of the user and group ID you want to use, respectively. Additionally, you must run the containers as the root user so that the container init can set the right permissions on the files. This is done in a Docker `run` command with the `--user=0` flag, and in Compose with the attribute `user: 0` to run as root.

Running Containers in the Local Time Zone

By default, FME Flow containers run in Coordinated Universal Time (UTC). To adjust to the local time zone of the container hosts, mount time zone files to the `fmeflowweb`, `fmeflowcore`, `fmeflowqueue`, and `fmeflowengine` containers, as follows:

```
volumes:
  - "/etc/timezone:/etc/timezone:ro"
  - "/etc/localtime:/etc/localtime:ro"
```

Alternatively, set the TZ environment variable for each container, as follows:

```
TZ=<timezone>
```

where *<timezone>* is a valid country code. For a list of country codes, see https://en.wikipedia.org/wiki/List_of_tz_database_time_zones.

Upgrading to a Patch Build in Docker

When an FME Flow build is released, two tags are created for each component's image. For example, if the release version is 2021.2, these tags are named *2021.2* and *2021.2-202111029*, where *202111029* is the date the image was released.

The available tags for each FME Flow component can be found on [Dockerhub](#).

Patch Builds

Images of each build are patched periodically with security updates. When this occurs, each image is released again with the current date as a suffix to the build number (for example, *2021.2-202111030*). As well, the original tag (for example, *2021.2* is updated to refer to the most recently released version of the build. With this in mind, if it is preferable to lock down to a specific image and avoid updating to a patched version, use the timestamped tags. Otherwise, to always use the latest version of a build, use the non-timestamped tag (for example, *2021.2*).

To upgrade to the latest patch build in a Docker Compose deployment

1. Pull the latest images. Run:

```
>> docker-compose pull
```

2. Update the FME Flow deployment with the new images. Run:

```
>> docker-compose -f docker-compose.yml up -d
```

3. Remove the old images. (They should be listed with repository *<none>* and tag *<none>*.)

To upgrade to a specific patch build

1. Search for the build on [Dockerhub](#) or [Quay.io](#), which will display each of the patch builds for that build number.
2. In your docker compose file, update each image tag (after the ':' in the image name) to the desired build/timestamp.
3. Update the FME Flow deployment with the new images. Run:

```
>> docker-compose -f docker-compose.yml up -d
```

See Also

- [To Upgrade an FME Flow Deployment](#)


Deploying FME Flow with Kubernetes

Prerequisites and Considerations

To deploy FME Flow with Kubernetes, you must fulfill the following prerequisites:

Install Kubernetes

Install and configure a Kubernetes cluster. To install a local (single-node) cluster, we recommend installing [Docker Desktop](#), and following the Getting Started guide. Once installed, select [Enable Kubernetes](#) to start a single-node cluster.

 **Tip** Ensure sufficient CPU and memory resources are assigned to your Docker Engine, preferably higher than the default allocations. We recommend at least 4 CPUs and 8 GB memory.

Install Helm

The Helm tool is required for pulling the fme-flow Kubernetes chart.

1. [Install Helm](#).
2. (Helm v2 only) Initialize it into the cluster. Run:

```
helm init
```

3. Add the [fme flow chart](#) to the cluster:

```
helm repo add safesoftware https://safesoftware.github.io/helm-  
charts/
```

Install NGINX

FME Flow deployments do not ship with an NGINX ingress controller container. Instead, we leverage the official nginx-ingress controller. The instructions to deploy this are [here](#). We recommend using Helm as the simplest method:

- Helm v2: `helm install stable/nginx-ingress --name my-nginx`
- Helm v3: `helm install my-nginx stable/nginx-ingress`

Other Considerations

When deploying FME Flow to a Kubernetes cluster, keep in mind the following:

- It is not possible to use the FME Flow Web User Interface to deploy FME Engines or change the number of FME Engines. For more information, see [Defining FME Engines, Queue Control, and Streams to a Kubernetes Deployment](#).
- Any memory or CPU limits you define on containers are ignored by all FME Flow processes. Memory and CPU limits defined on FME Engine containers are honored.
- Pods for the FME Engines and FME Flow Core/Web Services can be scaled; no other Pods can be scaled.

Performing the Kubernetes Deployment

Use the Helm tool to deploy FME Flow with Kubernetes. This tool allows you to set various deployment parameters, depending on need.

Add the Safe Software Helm Repository

```
helm repo add safesoftware https://safesoftware.github.io/helm-charts/
```


At a minimum, you must provide the FME Flow release and build number.

Example: Install FME Flow 2024.0

```
helm install fmeflow safesoftware/fmeflow --set  
fmeflow.image.tag=2024.0
```

After running the `install` command, you can connect to the FME Flow Web User Interface on `https://<hostname>`. For more information, see:

- [Accessing the Web User Interface](#)
- [Logging In to the Web Interface](#)

Example: Install with User-Supplied Values

Create a `values.yaml` file that passes user-supplied parameter values to the Helm chart.

To get the default Helm chart values file, run:

```
helm show values <chart>
```

For example:

```
helm show values safesoftware/fmeflow
```

To write the values to a file:

```
helm show values safesoftware/fmeflow >> values.yaml
```

For more information on value files, see the [Helm Docs](#).

A current list of supported parameters for FME Flow can be found on [GitHub](#).

Parameters to Note

The following are parameters you may want to change in the `values.yaml` file:

Parameter	Example or Possible Value	Description
<code>fmeflow.image.tag</code>	2024.0	Set to the FME Flow major release version you want to deploy, such as <code>2024.0</code> or <code>2024.1</code> .
<code>fmeflow.engines[].name</code>	“standard-group”	The name of the engine group. This can be changed, particularly if creating multiple engine groups.
<code>fmeflow.engines[].engines</code>	2	Controls the number of engine pods to start.
<code>fmeflow.engines[].type</code>	STANDARD DYNAMIC	Controls the type of engine to start.
<code>deployment.hostname</code>		Set to the IP address of the ingress controller noted previously.
<code>deployment.numCores</code>	2	Starts two FME Flow Core pods for fault tolerance and load balancing.

Parameter	Example or Possible Value	Description
deployment.useHostnameIngress	true false	If DNS is set up for the ingress controller, specify the DNS name for deployment.hostname and leave this parameter to true. If not, specify false.

Install FME Flow

Ensure the values.yaml file is saved, then reference it with the `helm install` command:

```
helm install <name> <chart> -f values.yaml
```

For example:

```
helm install fmeflow safesoftware/fmeflow -f values.yaml
```

If you make changes to the values.yaml file after FME Flow is installed (such as to scale FME Flow engines), use the `helm upgrade` command:


```
helm upgrade <name> <chart> -f values.yaml
```

For example:

```
helm upgrade fmeflow safesoftware/fmeflow -f values.yaml
```

Scale FME Engines

To set the number of FME Engines, use the values.yaml file and set the `fmeflow.engines[]` parameters listed in [Parameters to Note](#) (above). Alternatively, see [Defining FME Engines, Queue Control, and Streams to a Kubernetes Deployment](#).

 **Note** Changing the number of FME Engines in the FME Flow Web User Interface is not supported in Kubernetes deployments.

Delete an FME Flow Deployment

```
helm uninstall fmeflow -n <namespace_name>
```

Defining FME Engines, Queue Control, and Streams to a Kubernetes Deployment

You can define multiple [FME Engines](#) for a Kubernetes deployment. Managing FME Engines in the FME Flow Web User Interface is not supported in Kubernetes deployments. Instead, engine definition is managed through Kubernetes and a `values.yaml` file.

Each engine deployment defines the `name` of the deployment, the license `type` (`STANDARD` or `DYNAMIC`), and the number of replicas (`engines`) for that deployment. As well, the deployment `name` serves as an engine property. When configuring [Queue Control](#) and [Streams](#), you can use this property when assigning engines to queues (in engine assignment rules) and streams. Additional properties can be assigned to engines with the `engineProperties` specifier.

Each engine deployment can also specify Kubernetes scheduling parameters `nodeSelector`, `tolerations` and `affinity`.

Example FME Engine Deployments

The following examples specify FME Engine deployments in a `values.yaml` file when deploying the FME Flow Helm chart.

This example shows a default Helm deployment of one FME Engine deployment with two replicas. In the example, property `standard-group` may be specified in an engine assignment for a queue or stream of *Type: **Property***.

```
fmeflow:
  engines:
    - name: "standard-group"
      type: "STANDARD"
      engines: 2
      affinity: {}
      nodeSelector: {}
      tolerations: []
```

The following example defines three engine deployments—one standard engine with two replicas, one standard engine with three replicas, and one dynamic engine with three replicas. When configuring engine assignment to queues or streams, use the `name` properties `standard-group`, `standard-group-2`, and `dynamic-group` to assign engines to queues or a stream. Two additional `engineproperties`, `property1` and `property2`, are defined on `standard-group` and `standard-group-2` respectively, and can also be used in engine assignment.

```
fmeflow:
  engines:
    - name: "standard-group"
      type: "STANDARD"
      engines: 2
      engineProperties: "property1"
    - name: "standard-group-2"
      type: "STANDARD"
      engines: 3
      engineProperties: "property2"
    - name: "dynamic-group"
      type: "DYNAMIC"
      engines: 3
```

Deploying with Kubernetes and a Trusted Certificate

To deploy FME Flow with a trusted certificate to a Kubernetes cluster, you can run [cert-manager](#) in your cluster, which issues the necessary certificates.

Alternatively, you can deploy with a certificate and private key that are already generated.

Deploying with cert-manager

Use the `deployment.certManager.issuerName` and `deployment.certManager.issuerType` parameters in the FME Flow Helm chart to add annotations to the ingress. The annotations instruct cert-manager to issue a certificate for that ingress.

Deploying with an existing certificate

1. Upload the certificate to the cluster:

```
kubectl create secret tls fmefflow-tls-cert --key  
localhost.self.fmefflow.key --cert localhost.self.fmefflow.crt
```

2. Install FME Flow, setting the TLS certificate:

Example (Helm v3):

```
helm install fmefflow safesoftware/fmefflow --set  
fmefflow.image.tag=2024.0,  
deployment.tlsSecretName=fmefflow-tls-cert,  
deployment.hostname=localhost
```


When finished, connect to the FME Flow Web User Interface on <http://localhost>. For more information, see:

- [Accessing the Web User Interface](#)
- [Logging In to the Web Interface](#).

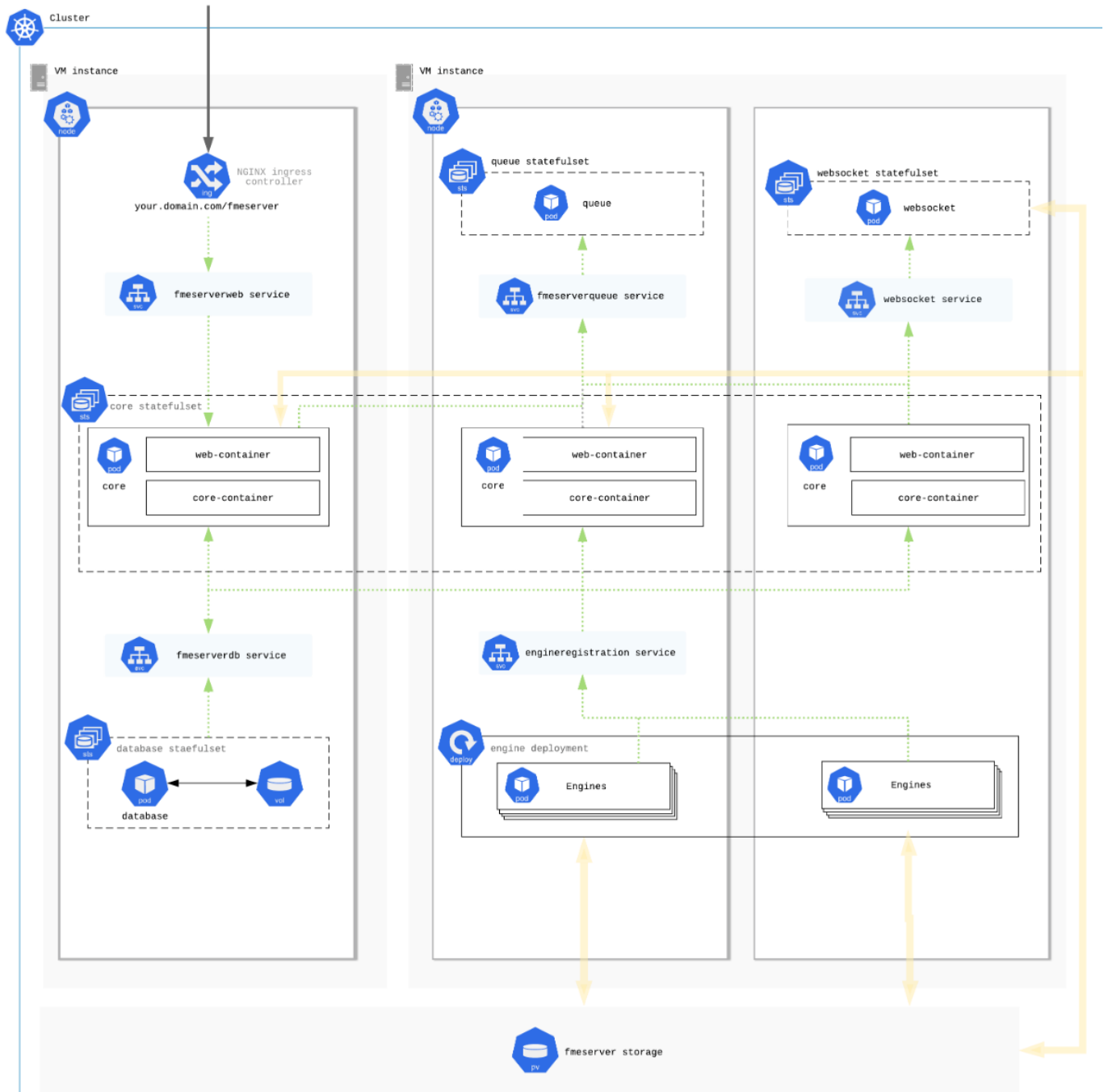
Deploying with Kubernetes across Multiple Hosts

You can duplicate the [FME Flow Core](#) across multiple hosts in a Kubernetes deployment. This duplication provides higher availability; if one host fails, the FME Flow Core on the other host can continue functioning.

To deploy FME Flow across multiple hosts, a network file system (NFS) or other distributed system is required. In a Kubernetes deployment, the storage volume must support a *ReadWriteMany* access mode. If no existing distributed file system is available, we recommend using an existing NFS server and configuring Kubernetes to use that server to provision volumes. Alternatively, use an offering from a Cloud provider, such as Azure Files.


 **Note** The Helm chart requires multiple hosts (nodes) to work with more than one FME Flow Core. Only one Core is launched per node.

Architecture of a High-Availability FME Flow Kubernetes Deployment



To Deploy Across Multiple Hosts


Deploying with an NFS Client Provisioner

 **Warning** Safe Software has not yet tested an NFS Client Provisioner deployment for production workflows.

You can deploy an NFS Client provisioner to use an existing NFS Server to provision volumes for Kubernetes to use. The provisioner must be installed before FME Flow and should not be removed before FME Flow is removed from the cluster.


1. Follow the instructions [here](#) to deploy the NFS client provisioner using the settings of your existing NFS Server.
2. Install FME Flow.

Example:

 **Note** The following script assumes the storage class name of the NFS Client Provisioner is unchanged from its default, "managed-nfs-storage."

- Helm v2: `helm install --namespace <fmeflow-namespace> -n <fmeflow-deployment-name> safesoftware/fmeflow --set fmeflow.image.tag=<fmeVersion>,deployment.numCores=2,storage.fmeflow.class=managed-nfs-storage,storage.fmeflow.accessMode=ReadWriteMany`
- Helm v3: `helm install <fmeflow-deployment-name> safesoftware/fmeflow --namespace <fmeflow-namespace> --set fmeflow.image.tag=<fmeVersion>,deployment.numCores=2,storage.fmeflow.class=managed-nfs-storage,storage.fmeflow.accessMode=ReadWriteMany`


Parameters

 **Note** To add other parameters, if required, see the relevant topic under [Deploying FME Flow with Kubernetes](#).

- *fmeFlow-namespace*: Kubernetes namespace in which to install FME Flow.
- *fmeFlow-deployment-name*: Helm deployment name, arbitrary.
- *fmeVersion*: FME Flow major release version (examples: [2024.0](#) or [2024.1](#)).


Deploying with an External Database

The [FME Flow Database](#) is a critical part of any FME Flow deployment. In a default Kubernetes deployment, it is a single point of failure. If it fails, FME Flow stops functioning. For a highly available deployment of FME Flow, we recommend that you use a highly available PostgreSQL database. For example, if you are deploying on AWS, you may want to use Amazon Relational Database Service (RDS), which is a highly available and scalable managed database.

 **Note** PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.

To use an external database outside of the Kubernetes cluster, perform the following:

1. Provision the database and take note of the host, master username, and master password.
2. Ensure the cluster has access on port 5432.
3. Create a secret in your cluster in the namespace where you will deploy that contains the admin password for the database. The password must be base64-encoded.
 - a. Create a `secrets.yaml` file similar to the following:

 **Note** You must manually encode the password in base64.

```
apiVersion: v1
kind: Secret
metadata:
  name: fmeFlowsecret
type: Opaque
data:
  password: Zm15473ja3M
```

b. Run: `kubectl create -f secrets.yaml`

c. Note the secret name and the name of the key that holds the value of the password.

4. Run the Helm chart, setting the following values:


- `deployment.deployPostgresql` (required): Set to `false` to use an external database.
- `fmeFlow.database.host` (required): The host name of the external database
- `fmeFlow.database.adminUser` (optional): The default is `postgres`. If the master user of the external database is not called this, you must set it.
- `fmeFlow.database.adminPasswordSecret` (required): The name of the secret you created.
- `fmeFlow.database.adminPasswordSecretKey` (required): The name of the key in the secret that holds the value of the base64-encoded password.
- `fmeFlow.database.port` (optional): The default port is 5432. If the external database runs on another port, you must set this.

When FME Flow is deployed, one of the containers will connect to the external database and run all of the SQL setup scripts.


Deploying with Kubernetes to a Cloud Provider

Deploying to Azure Kubernetes Service Using Azure Files

1. Launch the Cluster

 **Note** For Microsoft Azure documentation on deploying and connecting to an AKS cluster, click [here](#).

- a. Navigate to the Microsoft Azure portal in a web browser.
- b. Navigate to the Kubernetes services section.
- c. Click **Create > Create a Kubernetes Cluster**.
- d. On the **Basics** tab, specify the parameters:
 - *Subscription*: Specify your subscription and resource group that Kubernetes will run in.
 - *Cluster Details*: Specify a name for the cluster and the region to launch it. The Kubernetes version can remain as the default.
 - *Nodes*: For each FME Flow Core you wish to run, you need approximately 8GB of memory. For a single-Core deployment, we recommend 2 vCPUS, 8GB RAM, and 2 nodes. For a two-Core deployment, we recommend 4 vCPUs, 16GB RAM, and 3 nodes.

 **Note** More than one Core container requires a minimum of two nodes due to a constraint in the Helm chart that prevents Core containers residing on the same node. To ensure fault tolerance, we recommend three nodes because if a node is lost, two Cores can still run on the remaining two nodes.

- For a test deployment, leave all other parameters at their default settings.

- e. Click **Review + Create**.
- f. After validation checks complete, review the settings, and click **Create**.

2. Install Helm and an NGINX Ingress Controller

- a. Using [Azure Cloud Shell](#), connect to the Kubernetes cluster.
- b. Check that Helm is installed. Run:

```
helm version.
```

If not installed, [install Helm](#).

- c. Give Helm deploy permissions.

Run:

```
kubectl create clusterrolebinding add-on-cluster-admin --  
clusterrole=cluster-admin --serviceaccount=kube-system:default
```

- d. Install the NGINX ingress controller:

```
kubectl apply -f  
https://raw.githubusercontent.com/kubernetes/ingress-  
nginx/controller-  
<version>/deploy/static/provider/cloud/deploy.yaml
```

Replace `<version>` with the most recent version in the corresponding script in the [NGINX Ingress Controller Installation Guide](#). As well, replace other syntax elements with any changes in the updated script.

3. Setup Shared Storage using Azure Files

To deploy FME Flow across multiple nodes, create a storage class that defines a shared storage volume backed by Azure Files.

- a. Using Azure Cloud Shell, run:

```
az aks show --resource-group <resource_group> --name <cluster_  
name> --query nodeResourceGroup -o tsv
```

Replacing `<resource_group>` and `<cluster_name>` with what you defined when you deployed your cluster. This command provides the AKS node resource group for your cluster.

- b. Create a general-purpose storage account. Using the Azure CLI, run:

```
az storage account create --resource-group <node_resource_group> --name <storageaccountname> --sku Premium_LRS --kind FileStorage
```

Replacing `<node_resource_group>` with the result of the command in the previous step, and `<storageaccountname>` with a unique name for the storage account.

- c. Create a storage class in Kubernetes using the new Azure Files storage account. Use `vi` or `nano` to create the following file in Azure Cloud Shell:

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: azurefiles-fmeflow
provisioner: kubernetes.io/azure-file
parameters:
  skuName: Premium_LRS
  storageAccount: <storageaccountname>
mountOptions:
  - uid=1363
  - gid=1363
```

Replacing `<storageaccountname>` with the name you specified previously when creating the account.

- d. Run:

```
kubectl apply -f <path_to_file>
```

Where `<path_to_file>` is the YAML file created above.

To see `azurefiles-fmeflow` in the list, run:

```
kubectl get sc
```

- e. If RBAC is enabled, grant permissions to Azure Files:

```
kubectl create clusterrolebinding add-on-cluster-vol-binder --  
clusterrole=cluster-admin --serviceaccount=kube-  
system:persistent-volume-binder
```

4. Deploy FME Flow

Perform the following in the Azure Cloud Shell:

- a. Add the Safe Software Helm repository:

```
helm repo add safesoftware  
https://safesoftware.github.io/helm-charts/
```

- b. Get the IP address of the ingress controller:

```
kubectl get services --namespace ingress-nginx
```

Note the "External IP" of the nginx-ingress-controller. This is the ingress into the cluster. In a production environment, set up a DNS that points to this IP address and use that as the hostname when deploying FME Flow. In this procedure, we access FME Flow directly using the IP.

- c. Create a namespace in which to deploy. Run:

```
kubectl create namespace <namespace_name>
```

For example:

```
kubectl create namespace fmeflow
```

- d. Create a values.yaml file that will pass user-supplied parameter values into the Helm chart.

To get the default Helm chart values file, run:

```
helm show values <chart>
```

For example:

```
helm show values safesoftware/fmeflow
```

To write the values to a file:

```
helm show values safesoftware/fmeflow >> values.yaml
```

For more information on value files, see the [Helm Docs](#).

A current list of supported parameters for FME Flow can be found on [GitHub](#).

e. (Optional) Update parameters in values.yaml.

The following are parameters you may want to change in the values.yaml file:

Parameter	Example or Possible Value	Description
fmeflow.image.tag	2024.0	Set to the FME Flow major release version you want to deploy, such as <code>2024.0</code> or <code>2024.1</code> .
fmeflow.engines[].name	“standard-group”	The name of the engine group. This can be changed, particularly if creating multiple engine groups.
fmeflow.engines[].engines	2	Controls the number of engine pods to start.
fmeflow.engines[].type	STANDARD DYNAMIC	Controls the type of engine to start.

Parameter	Example or Possible Value	Description
deployment.hostname		If a DNS has been configured to point to the external IP address, set to that value. If using the external IP address only, leave as <code>localhost</code> and set <code>deployment.useHostnameIngress</code> to <code>false</code> .
deployment.numCores	2	Starts two FME Flow Core pods for fault tolerance and load balancing.
deployment.useHostnameIngress	true false	If DNS is set up for the ingress controller, specify the DNS name for <code>deployment.hostname</code> and leave this parameter to true. If not, specify false.
storage.fmeflow.class	azurefiles-fmeflow	Uses the Azure Files storage account set up previously.

f. Install FME Flow.

```
helm install <name> <chart> -f values.yaml -n <namespace>
```

For example:


```
helm install fmeflow safesoftware/fmeflow -f values.yaml -n
fmeflow
```


If you make changes to the values.yaml file after FME Flow is installed (such as to scale FME Flow engines), use the `helm upgrade` command:

```
helm upgrade <name> <chart> -f values.yaml -n <namespace>
```

For example:

```
helm upgrade fmeflow safesoftware/fmeflow -f values.yaml -n
fmeflow
```

- g. To access FME Flow after it deploys, invoke the external IP in your browser.

 **Note** If you are using the external IP address, and not a DNS, to access the deployment, you must update the URLs for each of the [FME Flow Web Services](#) to that address. To update, see [Services](#).


Deploying to Google Cloud Using Google Kubernetes Engine and Cloud Filestore

The following steps explain how to deploy an FME Flow Kubernetes deployment consisting of multiple FME Flow Core nodes to Google Cloud using Google Kubernetes Engine (GKE) and Cloud Filestore for storage.

1. Launch the Cluster

- a. Navigate to the [Google Kubernetes Engine](#) section of the Google Cloud Platform console.
- b. Select **Create Cluster** to deploy a new Kubernetes Cluster.
- c. If not already selected, select **Switch to Standard cluster** at the top right, and set the following parameters:
 - *Name*: Name of the cluster.
 - *Location type*: For test purposes, select **Zonal**.
 - *Zone*: Select the zone closest to you or your data.
 - *Master version*: Keep the default value.


- *Node pool*: For each FME Flow Core you want to run, you need approximately 8GB of memory. For a single-Core deployment, we recommend 2 vCPUS, 8GB RAM, and 2 nodes. For a two-Core deployment, we recommend 4 vCPUs, 16GB RAM, and 3 nodes.

 **Note** More than one Core container requires a minimum of two nodes due to a constraint in the Helm chart that prevents Core containers residing on the same node. To ensure fault tolerance, we recommend three nodes because if a node is lost, two Cores can still run on the remaining two nodes.

d. Click **Create**.

Expand instructions for an Autopilot cluster

- a. Navigate to the [Google Kubernetes Engine](#) section of the Google Cloud Platform console.
- b. Select **Create Cluster** to deploy a new Kubernetes Cluster.
- c. If not already selected, select **Switch to Autopilot cluster** at the top right, and set the following parameters:
 - *Name*: Name of the cluster.
 - *Region*: Select the region closest to you or your data.
 - *Networking*: Choose the correct network settings for your environment. You must ensure that the cluster is accessible to any other services you may use, such as Filestore or Cloud SQL.
- d. Click **Create**.

 **Note** When creating an Autopilot cluster, you may run into issues concerning resource requests. Ensure you update the resource requests before installing FME Flow to comply with the allowed minimum and maximum values. For more information, see [Resource requests in Autopilot](#).

2. Connect to the Cluster

Run the following commands in Cloud Shell (recommended). Alternatively, configure your local machine to run `kubectl` and `helm` commands to deploy resources to the cluster.

- a. Click **Connect** beside the Kubernetes cluster you just launched.
- b. Copy the "Command-line access" command and run in Google Cloud Shell or execute it on your terminal.
- c. To check if you are connected to the cluster, run:

```
kubectl get nodes
```

You should see all nodes in the cluster.

- d. Install the NGINX ingress controller:

```
kubectl apply -f
https://raw.githubusercontent.com/kubernetes/ingress-
nginx/controller-
<version>/deploy/static/provider/cloud/deploy.yaml
```

Replace `<version>` with the most recent version in the corresponding script in the [NGINX Ingress Controller Installation Guide](#). As well, replace other syntax elements with any changes in the updated script.

3. Set Up Shared Storage for the FME Flow System Share

Using [Cloud Filestore](#), storage for the FME Flow System Share is accessible across all nodes. This procedure creates a Cloud Filestore instance to host an NFS server for the Kubernetes cluster to use.

- a. Go to "Filestore" in the Google Cloud Platform console.
- b. Click **Create Instance**, and set the following:
 - *Instance ID*: Identifier for the instance.
 - *Instance tier*: **Standard**
 - *Location*: The same region and zone to which you deployed the Kubernetes cluster.
 - *Fileshare name*: Specify a name.
 - *Fileshare capacity*: **1TB** (default)
- c. Click **Create**.
- d. To deploy a storage class in the cluster to use the new Filestore instance of the NFS server, Kubernetes provides the [nfs-subdir-external-provisioner](#) Helm chart. First, go to the details of your Filestore instance and note the IP address and file share name.
- e. Run:


```
helm repo add nfs-subdir-external-provisioner
https://kubernetes-sigs.github.io/nfs-subdir-external-
provisioner/
helm install nfs-subdir-external-provisioner nfs-subdir-
external-provisioner/nfs-subdir-external-provisioner \
--set nfs.server=<ip_address> \
--set nfs.path=<fileshare_name>
```

Replace *<ip_address>* and *<fileshare_name>* with the values of your Fileshare.

For example:

```
helm install nfs-subdir-external-provisioner nfs-subdir-
external-provisioner/nfs-subdir-external-provisioner \
--set nfs.server=10.135.210.250 \
--set nfs.path=/fme-flow-storage
```

This command creates a storage class in the Kubernetes cluster called "nfs-client" that uses this Filestore NFS server for volumes in Kubernetes.

 **Note** For more information, see the [NFS Subdirectory External Provisioner Helm Chart](#) documentation.

4. Deploy FME Flow

- a. Add the Safe Software Helm repository:

```
helm repo add safesoftware  
https://safesoftware.github.io/helm-charts/
```

- b. Get the IP address of the Ingress Controller:

```
kubectl -n ingress-nginx get svc
```

Note the "EXTERNAL-IP" of the "ingress-nginx-controller". This is the ingress to the cluster. In a production environment, set up a DNS that points to this IP address and use that as the hostname when deploying FME Flow. In this procedure, we access FME Flow directly using the IP.

- c. Create a namespace in which to deploy. Run:

```
kubectl create namespace <namespace_name>
```

For example:

```
kubectl create namespace fmefflow
```

- d. Create a values.yaml file that will pass user-supplied parameter values into the Helm chart.

To get the default Helm chart values file, run:

```
helm show values <chart>
```

For example:

```
helm show values safesoftware/fmefflow
```

To write the values to a file:

```
helm show values safesoftware/fmeflow >> values.yaml
```

For more information on value files, see the [Helm Docs](#).

A current list of supported parameters for FME Flow can be found on [GitHub](#).

e. (Optional) Update parameters in values.yaml.

The following are parameters you may want to change in the values.yaml file:

Parameter	Example or Possible Value	Description
fmeflow.image.tag	2024.0	Set to the FME Flow major release version you want to deploy, such as <code>2024.0</code> or <code>2024.1</code> .
fmeflow.engines[].name	“standard-group”	The name of the engine group. This can be changed, particularly if creating multiple engine groups.
fmeflow.engines[].engines	2	Controls the number of engine pods to start.
fmeflow.engines[].type	STANDARD DYNAMIC	Controls the type of engine to start.

Parameter	Example or Possible Value	Description
fmeflow.healthcheck.liveness.initialDelaySeconds	60	For Autopilot clusters, set this value to 900, as FME Flow pods take longer to come online when using Autopilot. This will ensure the pods have enough time to start without premature health checks.
fmeflow.healthcheck.readiness.initialDelaySeconds	60	For Autopilot clusters, set this value to 900, as FME Flow pods take longer to come online when using Autopilot. This will ensure the pods have enough time to start without premature health checks.

Parameter	Example or Possible Value	Description
deployment.hostname		If a DNS has been configured to point to the external IP address, set to that value. If using the external IP address only, leave leave as <code>localhost</code> and set <code>deployment.useHostnameIngress</code> to <code>false</code> .
deployment.numCores	2	Starts two FME Flow Core pods for fault tolerance and load balancing.
deployment.useHostnameIngress	true false	If DNS is set up for the ingress controller, specify the DNS name for <code>deployment.hostname</code> and leave this parameter to true. If not, specify false.

Parameter	Example or Possible Value	Description
storage.fmeflow.class	nfs-client	Uses the Cloud Filestore NFS Server set up previously.

The following resource requests have been configured to comply with minimum resource requests allowed by Autopilot.

Note Autopilot may automatically increase resource requests for some pods. Autopilot rejects pods that have anti-affinity rules and do not meet the minimum or allowed resource request values.

For more information, see [Resource requests in Autopilot](#).

Expand for resource requests (Autopilot clusters only)

Note The following resource request values are appropriate for a single FME Flow Core, single FME Engine deployment on Autopilot.

```

1 engines:
2   groups:
3     - name: "standard-group"
4     resources:
5       requests:
6         memory: 512Mi
7         cpu: 250m

```

```

1 resources:
2   core:

```

```

3     requests:
4       memory: 1.5Gi
5       cpu: 250m
6   web:
7     requests:
8       memory: 1Gi
9       cpu: 250m
10    limits:
11      memory:
12      cpu:
13  queue:
14    requests:
15      memory: 512Mi
16      cpu: 500m
17  websocket:
18    requests:
19      memory: 512Mi
20      cpu: 250m

```

```

1  postgresql:
2    primary:
3      persistence:
4        existingClaim: fmeserver-postgresql
5      resources:
6        requests:
7          memory: 512Mi
8          cpu: 250m

```

f. Install FME Flow.

```
helm install <name> <chart> -f values.yaml -n <namespace>
```

For example:

```
helm install fmeFlow safesoftware/fmeFlow -f values.yaml -n
fmeFlow
```


If you make changes to the values.yaml file after FME Flow is installed (such as to scale FME Flow engines), use the `helm upgrade` command:

```
helm upgrade <name> <chart> -f values.yaml -n <namespace>
```

For example:

```
helm upgrade fmefflow safesoftware/fmefflow -f values.yaml -n
fmefflow
```

- g. To access FME Flow after it deploys, use your browser to invoke the external IP in your browser.

 **Note** If you are using the external IP address, and not a DNS, to access the deployment, you must update the URLs for each of the [FME Flow Web Services](#) to that address. To update, see [Services](#).

Deploying to Amazon Elastic Kubernetes Service Using Elastic File System

1. Launch the Cluster and Connect

To launch and connect to a cluster in Amazon Elastic Kubernetes Service (EKS), follow the instructions in the [Getting Started with Amazon EKS](#) documentation. You must configure some prerequisite resources such as an Amazon Virtual Private Cloud (VPC) and Identity and Access Management (IAM) role for the cluster.

To deploy an FME Flow to the cluster for test purposes, we recommend the following settings:

- *NodeAutoScalingGroupMinSize*: 3
- *NodeAutoScalingGroupDesiredSize*: 3
- *NodeAutoScalingGroupMaxSize*: 4
- *NodeInstanceType*: t3.large
- *Disk size*: 40 GB

You can skip the step to "Launch a Guest Book Application."

2. Install Helm and an NGINX Ingress Controller

- a. Check if Helm is already installed. Run:

```
helm version
```

If not installed, [install Helm](#).

- b. Give Helm deploy permissions.

Run:

```
kubectl create clusterrolebinding add-on-cluster-admin --  
clusterrole=cluster-admin --serviceaccount=kube-system:default
```


- c. Install the NGINX ingress controller for AWS:

```
kubectl apply -f  
https://raw.githubusercontent.com/kubernetes/ingress-  
nginx/controller-  
<version>/deploy/static/provider/aws/deploy.yaml
```

Replace `<version>` with the most recent version in the corresponding script in the [NGINX Ingress Controller Installation Guide](#). As well, replace other syntax elements with any changes in the updated script.


3. Configure Shared Storage Using Amazon Elastic File System

To deploy FME Flow across multiple nodes, configure storage for the FME Flow System Share using Amazon Elastic File System (EFS).

 **Note** The default Amazon EFS performance options may not be adequate for how your FME Flow is used. If you experience performance issues, see the [Amazon EFS Performance](#) documentation.

- a. From the AWS Management Console, navigate to EFS.
 - i. Click **Create file system**.
 - ii. Select the new VPC created for the Kubernetes cluster.
 - iii. Click **Customize**.
 - iv. Add any tags as desired. Everything can be left as default. Click **Next**.
- b. The subnets should change to the EKS subnets. Add the EKS Node security group to the list of security groups so that our nodes have permissions to mount them.

- i. Click **Next** and then **Create**.
- ii. Create Access Point. We recommend the following settings:
 - Root directory path: /data/fmeflowdata
 - POSIX user
 - User ID: 1363
 - Group ID: 1363
 - Root directory creation permissions
 - Owner user ID: 1363
 - Owner group ID: 1363
 - Permissions: 0755
- c. Install the [Amazon EFS CSI Driver](#) for your region.

 **Note** You can use either dynamic or static provisioning for your EFS drive. The example in the next step shows how to use dynamic provisioning for the FME Flow System Share on EFS.

- d. To create the EFS storage class, save the following .yaml snippet and apply with kubectl:

```
kind: StorageClass
apiVersion: storage.k8s.io/v1
metadata:
  name: efs-sc
provisioner: efs.csi.aws.com
parameters:
  provisioningMode: efs-ap
  filesystemId: fs-<file-system-id>
  directoryPerms: "700"
```

Replace `<file-system-id>` with the File System ID for your EFS.

- e. Create the persistent volume using the following yaml file as an example. You will need to replace the `volumeHandle` with your File system ID and Access point ID from the EFS.

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: fme-flow-data
spec:
  capacity:
    storage: 10Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteMany
  persistentVolumeReclaimPolicy: Retain
  storageClassName: efs-sc
  csi:
    driver: efs.csi.aws.com
    volumeHandle: fs-<file-system-id>::fsap-<access-point-id>
```

- f. Add the [Amazon EBS CSI driver](#) add-on to your cluster.

4. Deploy FME Flow

- a. Add the Safe Software Helm repository:

```
helm repo add safesoftware
https://safesoftware.github.io/helm-charts/
```

- b. Get the IP address of the Ingress Controller:

```
kubectl describe service ingress-nginx-controller --namespace
ingress-nginx
```

Note the "LoadBalancer Ingress:" of the `ingress-nginx-controller`. This is the ingress into the cluster.

c. Create a namespace in which to deploy. Run:

```
kubectl create namespace <namespace_name>
```

For example:

```
kubectl create namespace fmeFlow
```

d. Create a values.yaml file that will pass user-supplied parameter values into the Helm chart.

To get the default Helm chart values file, run:

```
helm show values <chart>
```

For example:

```
helm show values safesoftware/fmeFlow
```

To write the values to a file:

```
helm show values safesoftware/fmeFlow >> values.yaml
```

For more information on value files, see the [Helm Docs](#).

A current list of supported parameters for FME Flow can be found on [GitHub](#).

e. (Optional) Update parameters in values.yaml.

The following are parameters you may want to change in the values.yaml file:

Parameter	Example or Possible Value	Description
fmeFlow.image.tag	2024.0	Set to the FME Flow major release version you want to deploy, such as 2024.0 or 2024.1.

Parameter	Example or Possible Value	Description
<code>fmeflow.engines[].name</code>	“standard-group”	The name of the engine group. This can be changed, particularly if creating multiple engine groups.
<code>fmeflow.engines[].engines</code>	2	Controls the number of engine pods to start.
<code>fmeflow.engines[].type</code>	STANDARD DYNAMIC	Controls the type of engine to start.
<code>deployment.hostname</code>		If a DNS has been configured to point to the external IP address, set to that value. If using the external IP address only, leave as <code>localhost</code> and set <code>deployment.useHostnameIngress</code> to <code>false</code> .
<code>deployment.numCores</code>	2	Starts two FME Flow Core pods for fault tolerance and load balancing.

Parameter	Example or Possible Value	Description
deployment.useHostnameIngress	true false	If DNS is set up for the ingress controller, specify the DNS name for deployment.hostname and leave this parameter to true. If not, specify false.
storage.fmeflow.class	efs-sc	Uses the EFS storage class set up previously. To verify the name, run <code>kubectl get sc</code>

f. Install FME Flow.

```
helm install <name> <chart> -f values.yaml -n <namespace>
```

For example:

```
helm install fmeflow safesoftware/fmeflow -f values.yaml -n fmeflow
```


If you make changes to the values.yaml file after FME Flow is installed (such as to scale FME Flow engines), use the `helm upgrade` command:

```
helm upgrade <name> <chart> -f values.yaml -n <namespace>
```


For example:

```
helm upgrade fmeflow safesoftware/fmeflow -f values.yaml -n fmeflow
```

- g. To access FME Flow after it deploys, invoke the external IP in your browser.

 **Note** If you are using the external IP address, and not a DNS, to access the deployment, you must update the URLs for each of the [FME Flow Web Services](#) to that address. To update, see [Services](#).

Deploying to Alibaba Cloud Using Network Attached Storage

 **Note** Official Alibaba documentation on deploying and connecting to a Kubernetes cluster is available [here](#).

1. Launch the Cluster

- a. Navigate to the Alibaba Cloud console.
- b. Click **Container Service**.
- c. Click **Create Kubernetes Cluster**.
- d. Click the **Managed Kubernetes** tab at top.
- e. Specify the following parameters:
 - *Cluster name, Region, Zone, and VPC.*
 - Choose an *Instance type* and *Node count*. For example, **ecs.mn4.large Instance type**, quantity of **3**. Uncheck **Attach Data Disk**.
 - Specify an *ssh key pair* for logging in.
 - For all other parameters, keep the default settings.
- f. Click **Create**. The cluster may take 5-10 minutes to create.

2. Connect to the Cluster

- a. Go to **Container Service** in the Alibaba Cloud console.
- b. Click on the cluster you just created.

- c. At the bottom of the page is a kubectl config file. Copy and paste this into a .yaml file on your machine.
- d. Set the environment variable KUBECONFIG to point to this file in your terminal.
- e. To check you are connected, run:

```
kubectl get nodes
```

3. Install Helm

- a. Check that Helm is installed. Run:

```
helm version
```

If not installed, [install Helm](#).

- b. Give Helm deploy permissions.

Run:

```
kubectl create clusterrolebinding add-on-cluster-admin --  
clusterrole=cluster-admin --serviceaccount=kube-system:default
```

4. Configure Network Attached Storage (NAS)

- a. In the Alibaba Cloud console, navigate to the NAS console
- b. Click **Create File System**.
- c. Specify the parameters:
 - *Region* and *Zone*: Specify the same as your Kubernetes cluster.
 - Under *Storage Type*, select **SSD performance-type**
 - *Storage Type*: **NFS**
- d. Click **OK**.
- e. Click **Add Mount Point**.
- f. Specify the parameters:
 - *Mount Point Type*: **VPC**
 - Select the same VPC as the one in which the Kubernetes cluster is running.
 - Select the default VPC permission group.

- g. Click **OK**.
- h. To add an NAS storage class to the Kubernetes cluster, go to the [Alibaba documentation on using NAS storage in a Kubernetes cluster](#).
 - i. Under "Dynamic storage volumes", copy and paste the "Install the plug-in" config file into a file on your local machine.
 - ii. In that file, modify the value of `NFS_SERVER` in the controller container, and the "server" in the volume to be the mount point URL created for the NAS.
 - iii. Remove the `nodeSelector` section completely.
 - iv. Apply to the cluster with `kubectl apply -f <path_to_yaml>`, where `<path_to_yaml>` is the config file created in the previous step.
- i. You should now have a storage class in your cluster with the name "alicloud-nas" that can use the FME Flow System Share.

5. Deploy FME Flow

- a. Add the Safe Software Helm repository:

```
helm repo add safesoftware  
https://safesoftware.github.io/helm-charts/
```

- b. Construct a hostname for the FME Flow deployment. Alibaba's Kubernetes Ingress Controller resolves a DNS name in the form:

```
*.<cluster-id>.<region-id>.alicontainer.com
```

For example, you can use DNS hostname:

```
fmeflow.<cluster-id>.<region-id>.alicontainer.com
```

`<cluster-id>` and region can be found on the Container Service page of the Alibaba Cloud console. See the [Alibaba documentation](#) for a mapping of regions to `<region-id>`.

- c. Create a values.yaml file that will pass user-supplied parameter values into the Helm chart.

To get the default Helm chart values file, run:

```
helm show values <chart>
```

For example:

```
helm show values safesoftware/fmeflow
```

To write the values to a file:

```
helm show values safesoftware/fmeflow >> values.yaml
```

For more information on value files, see the [Helm Docs](#).

A current list of supported parameters for FME Flow can be found on [GitHub](#).

d. (Optional) Update parameters in values.yaml.

The following are parameters you may want to change in the values.yaml file:

Parameter	Example or Possible Value	Description
fmeflow.image.tag	2024.0	Set to the FME Flow major release version you want to deploy, such as <code>2024.0</code> or <code>2024.1</code> .
fmeflow.engines[].name	“standard-group”	The name of the engine group. This can be changed, particularly if creating multiple engine groups.
fmeflow.engines[].engines	2	Controls the number of engine pods to start.
fmeflow.engines[].type	STANDARD DYNAMIC	Controls the type of engine to start.

Parameter	Example or Possible Value	Description
deployment.hostname		If a DNS has been configured to point to the external IP address, set to that value. If using the external IP address only, leave as <code>localhost</code> and set <code>deployment.useHostnameIngress</code> to <code>false</code> .
deployment.numCores	2	Starts two FME Flow Core pods for fault tolerance and load balancing.
deployment.useHostnameIngress	true false	If DNS is set up for the ingress controller, specify the DNS name for <code>deployment.hostname</code> and leave this parameter to true. If not, specify false.
storage.fmeflow.class	alicloud-nas	Uses the Alibaba NAS set up previously.

e. Install FME Flow.

```
helm install <name> <chart> -f values.yaml -n <namespace>
```

For example:

```
helm install fmeflow safesoftware/fmeflow -f values.yaml -n
fmeflow
```


If you make changes to the values.yaml file after FME Flow is installed (such as to scale FME Flow engines), use the `helm upgrade` command:

```
helm upgrade <name> <chart> -f values.yaml -n <namespace>
```

For example:

```
helm upgrade fmeflow safesoftware/fmeflow -f values.yaml -n
fmeflow
```


- f. To access FME Flow after it deploys, invoke the external IP in your browser.

 **Note** If you are using the external IP address, and not a DNS, to access the deployment, you must update the URLs for each of the [FME Flow Web Services](#) to that address. To update, see [Services](#).

Upgrading a Kubernetes Deployment of FME Flow

With little downtime, you can upgrade a Kubernetes deployment of FME Flow to a later version using the following procedure:

- Launch a new FME Flow of the desired build version in a new namespace.
- Backup the current FME Flow and restore it to the new FME Flow.
- Update the ingress hostname from the old FME Flow to the new FME Flow.
- Remove the old FME Flow.

 **Warning** You may have [Schedules](#), [Automations](#) workflows, or [Notifications](#) that would conflict with themselves during the time that both the new and old FME Flows are running. Disable these tools before proceeding with a backup.

 **Note** See also [Upgrading to a Patch Build in Kubernetes](#).

Performing the Upgrade

1. Create a new namespace for the new version of FME Flow:

```
kubectl create namespace <namespace-new-version>
```

For example:

```
kubectl create namespace fme-flow-2024-1-1
```

2. If your FME Flow uses an external PostgreSQL database for the FME Flow Database, copy the admin password secret, using the namespace of the current version of FME Flow, to the new version. (If your FME Flow uses the PostgreSQL container that is deployed with the Helm chart, skip this step.)

```
kubectl get secret postgresql-admin-password --namespace  
<namespace-old-version> --export -o yaml | kubectl apply --  
namespace=<namespace-new-version> -f -
```

For example:

```
kubectl get secret postgresql-admin-password --namespace fme-flow -  
-export -o yaml | kubectl apply --namespace=fme-flow-2024-1-1 -f -
```

3. Make a copy of the values .yaml file for the current FME Flow and update the following values:
 - `deployment.hostname`: Set to a temporary hostname. (You will reset it later in this procedure.)
 - `fme-flow.image.tag`: Set to the container image tag you want to deploy for the upgraded FME Flow.
 - If your FME Flow uses an external PostgreSQL database for the FME Flow Database, set the following values to be different from the ones in use for the current FME Flow:
 - `fme-flow.database.name`
 - `fme-flow.database.username`

4. Update your Helm repository:

```
helm repo update
```


5. Deploy FME Flow in the new namespace using Helm and the updated values .yaml file.

```
helm install --namespace <namespace-new-version> -n <helm-  
deployment-name> safesoftware/fmeflow -f <new-values-filename>
```

For example:

```
helm install --namespace fmeflow-2024-1-1 -n fmeflow  
safesoftware/fmeflow -f values-2024.1.1.yaml
```

6. From the [FME Flow Web User Interface](#), perform the following:

- a. [License](#) FME Flow.
- b. [Backup](#) the old FME Flow.
- c. [Restore](#) the backup to the new FME Flow.

7. Change the DNS of the old FME Flow to a different, temporary URL. Doing so will allow you to update the DNS of the new FME Flow to your production URL.

- a. In the old values .yaml file, change the value of `deployment.hostname`.
- b. Upgrade the Helm chart of the previous deployment.

```
helm upgrade --namespace <namespace-old-version> <helm-  
deployment-name> safesoftware/fmeflow -f <old-values-filename>
```

For example:

```
helm upgrade --namespace fmeflow fmeflow safesoftware/fmeflow  
-f values-2024.0.0.yaml
```

8. Update the DNS of the new FME Flow to your production URL.

- a. In the new values .yaml file, change the value of `deployment.hostname`.
- b. Upgrade the Helm chart of the new deployment.

```
helm upgrade --namespace <namespace-new-version> <helm-  
deployment-name> safesoftware/fmeflow -f <new-values-filename>
```

For example:

```
helm upgrade --namespace fmeflow-2024-1-1 fmeflow  
safesoftware/fmeflow -f values-2024.1.1.yaml
```

9. Verify that the upgraded FME Flow works as expected. If you find issues, you can switch back the `deployment.hostname` values in the old and new values `.yaml` files so that your production URL points back to the old FME Flow.
 10. Remove the old version of FME Flow.
 - Helm v2: `helm del --purge <helm-deployment-name>`
 - Helm v3: `helm uninstall <helm-deployment-name> -n <namespace_name>`
- For example:
- Helm v2: `helm del --purge fmeflow`
 - Helm v3: `helm uninstall fmeflow -n fmeflow`
11. If your FME Flow uses an external PostgreSQL database for the FME Flow Database, drop the database for the old FME Flow.

Upgrading to a Patch Build in Kubernetes

When an FME Flow build is released, two tags are created for each component's image. For example, if the release version is 2021.2, these tags are named `2021.2` and `2021.2-202111029`, where `202111029` is the date the image was released.

The available tags for each FME Flow component can be found on [Dockerhub](#).

Patch Builds

Images of each build are patched periodically with security updates. When this occurs, each image is released again with the current date as a suffix to the build number (for example, `2021.2-202111030`). As well, the original tag (for example, `2021.2` is updated to refer to the most recently released version of the build. With this in mind, if it is preferable to lock down to a specific image and avoid updating to a patched version, use the timestamped tags. Otherwise, to always use the latest version of a build, use the non-timestamped tag (for example, `2021.2`).

To upgrade to a patched image in a Kubernetes deployment

1. Add the FME Flow Helm repository and the values file that were initially used to deploy FME Flow. Run:

```
>> helm repo add safesoftware https://safesoftware.github.io/helm-  
charts
```

2. Update `fmeflow.image.tag` in the values file to the desired tag. Run:

```
>> helm upgrade --namespace <namespace> safesoftware/fmeflow -f  
values.yml
```

See Also

- [Upgrading a Kubernetes Deployment of FME Flow](#)

Deploying FME Flow in the Cloud

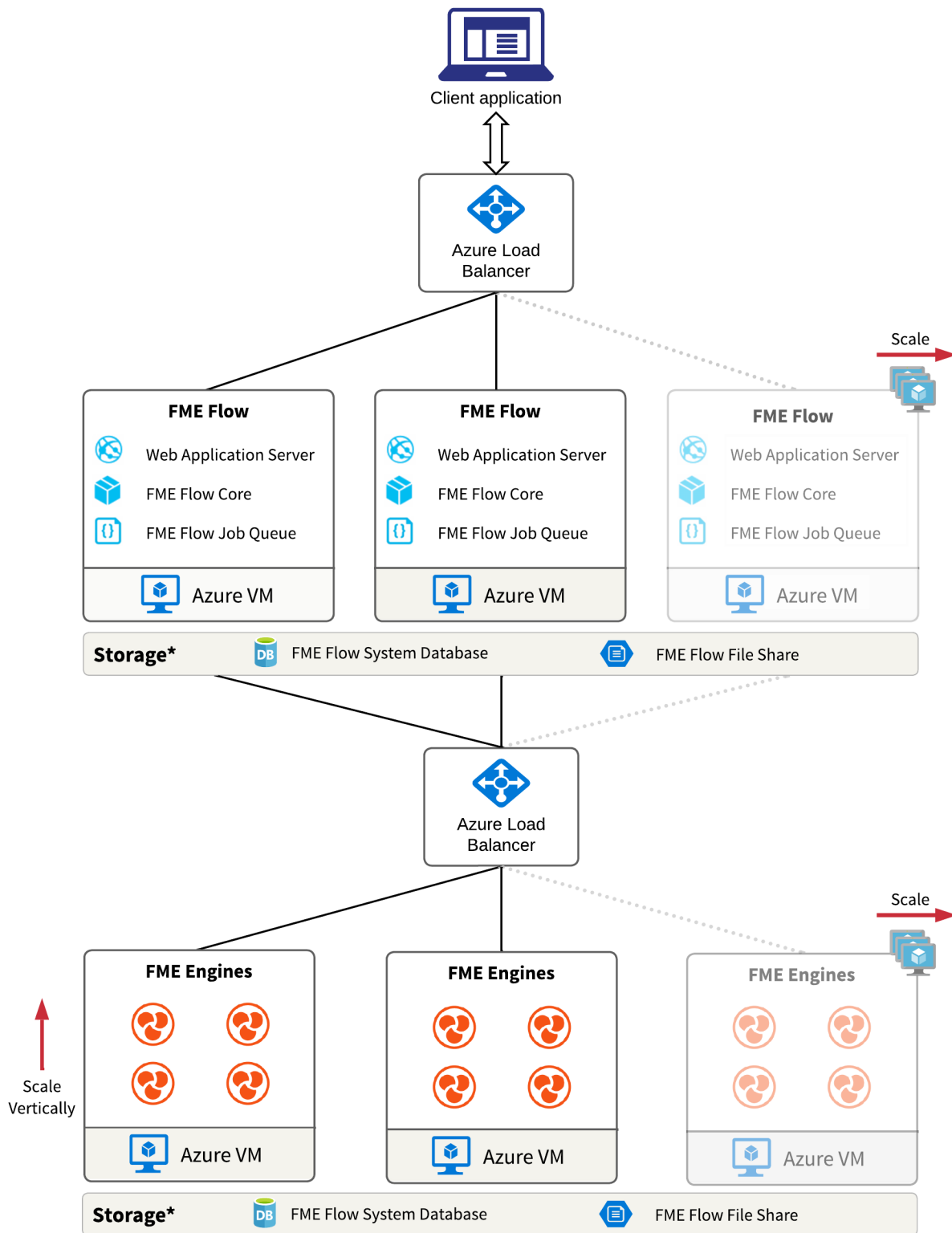
Safe Software Inc. provides a variety of options for deploying FME Flow to a Cloud environment.

Azure Marketplace

Microsoft Azure Marketplace allows you to deploy FME Flow on Windows, optimized for Azure Cloud. This option, based on a [distributed](#) FME Flow architecture, defines the virtual machine configuration, file storage, load balancers, and databases. It also leverages [Azure Virtual Machine Scale Sets](#), making it easy to scale the [FME Engines](#).

This deployment is not as flexible as deploying FME Flow manually using installers. For example, you cannot easily change the underlying images (for example, installing Esri software) or switch to a different database.

Mouse over to view architecture:



* The same FME Flow database and file share need to be accessed by both the FME Engine VM and the FME Flow Core VM.

To get started with an Azure Marketplace deployment, click [here](#).

Terraform

[Terraform](#) is a popular Infrastructure as Code (IaC) tool to create templates for multiple cloud providers. The following Terraform templates allow for [distributed](#) FME Flow deployments on Windows:

- [Amazon Web Services](#)
- [Microsoft Azure](#)

Vendor-Specific IaC Tools

As alternatives to Terraform for a Windows [distributed](#) FME Flow deployment on Amazon Web Services or Microsoft Azure, the following vendor-specific options are available:

- [AWS CloudFormation](#)
- [Azure Resource Manager \(ARM\) Bicep templates](#)


Installing Remote Engine Services

Use these instructions to install [Remote Engine Services](#) on a remote server. To obtain a Remote Engine Service installer, visit the [FME downloads page](#). Remote Engine Service installers are named beginning with `fme-flow-remote-engine-*`.

FME Flow Remote Engine Installation Extractor

Start the FME Flow remote engine installer.

On the Installation Extractor dialog, specify a *Destination folder* in which to extract installation files. Note that this folder *only* specifies where installation files are extracted—it does *not* specify where FME Flow remote engine program files are installed. That location is specified later in the installer.

 **Tip** Take note of the specified *Destination folder* location. Following installation, you may wish to:

- Review the installation .log file.
- Remove these files to save disk space.




Click **Install**.

When extraction is complete, the installer opens. Proceed through the installer to the Destination Folder dialog.

Destination Folder

- **Install FME Flow Remote Engine to:** By default, most files for your FME Flow remote engine installation are written to C:\Program Files\FMEFlowRemoteEngine\. To change the installation directory, modify the path.
- **Install the FME Flow System Share to:** By default, FME Flow System Share files, which include [Repositories](#) and [Resources](#), are written to %ALLUSERSPROFILE%\Safe Software\FME Flow\.

 **Note** Unless modified, C:\ProgramData is the default value of the %ALLUSERSPROFILE% environment variable.

This configuration is recommended, because applications that use FME Flow to write data can access C:\ProgramData\Safe Software\FME Flow when all [FME Flow System Services](#) run under the local system account (the most common scenario), while the

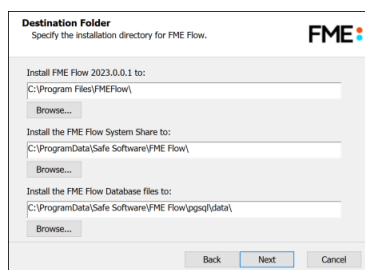
remaining files are protected in a read-only location. To change the installation directory of FME Flow System Share files, modify the path.

Note If you plan to scale-out your FME Flow anytime following installation by [Adding FME Engines on a Separate Machine](#), the specified directory must be read/write accessible by the account that runs the additional FME Engines.

- **Install the FME Flow Database files to:** By default, [FME Flow Database](#) files are written to %ALLUSERSPROFILE%\Safe Software\FME Flow\pgsql\data.

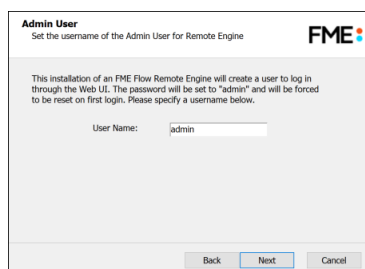
Note Unless modified, C:\ProgramData is the default value of the %ALLUSERSPROFILE% environment variable.

To change the installation directory of FME Flow Database files, modify the path.




Admin User

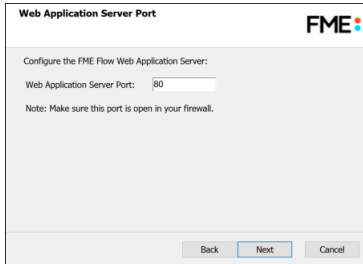
This installation of an FME Flow Remote Engine Service creates a user to log in to the [FME Flow Web User Interface](#). To log in to the Web User Interface for the first time as an administrator following installation, enter the *User Name* specified below and *Password* **admin**. You will be prompted to update the password upon initial use, based on the default [Password Policy](#) configuration.



Web Application Server Port


Port 80 is the recommended default. If this port is in use, port 8080 is recommended.


 **Note** Common applications that may be using port 80 include Skype and Internet Information Services (IIS). Either turn off these services, or select a different port, such as 8080.

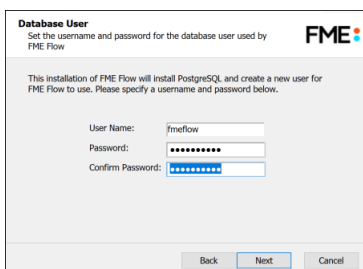


Database User

The installation creates the [FME Flow Database](#) on a PostgreSQL database server, and a user account under which to use the database. Specify a *User Name* (defaults to **fmeFlow**) and *Password* for the database user account.

 **Warning** Do not specify *User Name* **postgres**. This name corresponds to the PostgreSQL master user. If the corresponding passwords do not match, the database will not install.

 **Note** The password for the FME Flow Database user account is encrypted in FME Flow installation files.



On the last screen, click **Install**.

Upgrading FME Flow

Full FME Flow Upgrade	297
FME Engine Only Upgrade	306
Upgrading a Kubernetes Deployment of FME Flow	310



Full FME Flow Upgrade

There are multiple approaches to upgrading your FME Flow installation, each with advantages and disadvantages. Read below to decide which is right for you.

Upgrading to a Second Machine, Using a Different Host Name

Advantages

- Less risk because the FME Flow configuration remains available from the old installation.
- The existing FME Flow installation remains accessible.
- No DNS entry modification is required by your IT department.

Disadvantages

- Requires provision of additional machine.
- Users and applications must be updated to access FME Flow through a different host name.

For procedure, click [here](#).

Upgrading to a Second Machine, Using the Same Host Name

Advantages

- Less risk because the FME Flow configuration remains available from the old installation.
- The existing FME Flow installation remains accessible from the same host name without interruption.

Disadvantages

- Requires provision of additional machine.
- Requires DNS entry modification by your IT department.

For procedure, click [here](#).

Upgrading In-Place, on the Same Machine

Advantages

- Does not require provision of additional machine.
- Does not require DNS entry modification by your IT department.

Disadvantages


- More risk because the FME Flow configuration cannot be retrieved if the backup file is corrupted.
- The FME Flow installation is inaccessible for a period of time.

For procedure, click [here](#).


See Also

- [FME Engine Only Upgrade](#)


Upgrading to a Second Machine, Using a Different Host Name

 **Note** If the FME Flow you are upgrading from was [configured for HTTPS](#), you must reconfigure HTTPS after the upgrade is complete.


1. Provide a new machine for the upgrade.
2. [Backup](#) the FME Flow configuration from the old machine.

 **Warning** If [System Encryption](#) is **Standard** (default) on the FME Flow that is backed up, it must also be **Standard** on the FME Flow that is restored, and use the same custom encryption key. Before proceeding, ensure your custom encryption key is downloaded and safely saved.

3. [Install](#) FME Flow on the new machine. On the FME Flow Hostname dialog, specify the new hostname.


 **Note** If your FME Flow installation includes an FME Flow Database that is configured on your own database server, you must perform the step to *Configure the FME Flow Database on a Separate Database Server* as part of installation. You cannot reuse the FME Flow Database from a previous installation because the table schemas may differ from the upgraded version.

4. [Restore](#) the backed-up FME Flow configuration to the new machine.
5. Test that your configuration has been migrated successfully.

 **Note** Hardcoded paths to the previous host name in workspaces, transformers and HTTP push subscriptions must be manually updated.

6. [Uninstall](#) FME Flow from the old machine (or recycle the machine).

Upgrading to a Second Machine, Using the Same Host Name

 **Note** If the FME Flow you are upgrading from was [configured for HTTPS](#), you must reconfigure HTTPS after the upgrade is complete.

1. Provide a new machine for the upgrade.
2. Note the host name you are using to access your currently-installed FME Flow.

3. On the new machine, configure the hosts file so that this host name resolves to the new machine.

Windows:

1. In a text editor, open C:\Windows\System32\Drivers\etc\hosts
2. Add a line like this: `<localhost_ip> <hostname>`


For example: `127.0.0.1 fmefflow.com`

Linux:


1. In a text editor, open /etc/hosts
2. Add a line like this: `<localhost_ip> <hostname> <hostname>`

For example: `127.0.0.1 fmefflow.com fmefflow.com`

4. [Backup](#) the FME Flow configuration from the old machine.

 **Warning** If [System Encryption](#) is **Standard** (default) on the FME Flow that is backed up, it must also be **Standard** on the FME Flow that is restored, and use the same custom encryption key. Before proceeding, ensure your custom encryption key is downloaded and safely saved.


5. [Install](#) FME Flow on the new machine. On the FME Flow Hostname dialog, specify the same host name as the previous installation.

 **Note** If your FME Flow installation includes an FME Flow Database that is configured on your own database server, you must perform the step to *Configure the FME Flow Database on a Separate Database Server* as part of installation. You cannot reuse the FME Flow Database from a previous installation because the table schemas may differ from the upgraded version.


6. [Restore](#) the backed-up FME Flow configuration to the new machine.

7. Test that your configuration has been migrated successfully by running a workspace from the Web User Interface in a browser on the new machine. Alternatively, if you want to test from a remote machine, you must temporarily configure the machine to route requests for the FME Flow host to the new FME Flow machine. Use the same procedure as step 3 above.
8. Instruct your IT department to switch the DNS setting for the host name so that it routes to the new machine instead of the old machine.
9. [Uninstall](#) FME Flow from the old machine (or recycle the machine).


Upgrading In Place, On the Same Machine

 **Note** If the FME Flow you are upgrading from was [configured for HTTPS](#), you must reconfigure HTTPS after the upgrade is complete.

1. [Back Up Configuration](#).


 **Warning** If [System Encryption](#) is **Standard** (default) on the FME Flow that is backed up, it must also be **Standard** on the FME Flow that is restored, and use the same custom encryption key. Before proceeding, ensure your custom encryption key is downloaded and safely saved.

2. Verify the integrity of the backup by [restoring the configuration](#) to the existing FME Flow. A successful restore indicates the backup is valid.
3. [Uninstall](#) FME Flow.
4. [Install](#) the newer version.

 **Note** If your FME Flow installation includes an FME Flow Database that is configured on your own database server, you must perform the step to *Configure the FME Flow Database on a Separate Database Server* as part of installation. You cannot reuse the FME Flow Database from a previous installation because the table schemas may differ from the upgraded version.


5. [Restore Configuration](#).

Back Up Configuration

 **Warning** If [System Encryption](#) is **Standard** (default) on the FME Flow that is backed up, it must also be **Standard** on the FME Flow that is restored, and use the same custom encryption key. Before proceeding, ensure your custom encryption key is downloaded and safely saved.

To back up an FME Flow, perform the following:


1. Back up the FME Flow configuration files.
2. Back up repository file versions, if applicable.
3. Back up the FME Flow configuration .

 **Note** Before proceeding, be sure to note your product's serial number. It may be required when you restore the configuration to the upgraded FME Flow. You can find the serial number on the [Licensing](#) page.

Back Up the FME Flow Configuration Files

Before proceeding with the primary task of backing up your FME Flow configuration, we recommend manually backing up the [FME Flow Configuration Files](#) to a location outside of your FME Flow installation directory, in case you need to refer to them later. These files contain configurations for the FME Flow Database 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** Configuration files from a backed-up FME Flow are for reference only. Do not copy them to the corresponding directory of a restored FME Flow.


Back up Repository Versions

If [Version Control](#) is enabled on the current FME Flow, and it is *not* configured with a remote Git repository, previous versions of [repository files](#) are *not* available following an upgrade.


You can view all versions from the Repositories page ([History](#)), and download them as desired.

Back up the FME Flow Configuration

The FME Flow Web User Interface makes it easy to back up your FME Flow. For information about what is included in a backup, see [Which FME Flow Configuration Components are Migrated?](#).

 **Note** Backing up job history is not supported.

To perform the backup, use the Web User Interface [tool](#).

 **Note** To restore your configuration after you upgrade FME Flow, see [Restore Configuration](#).


Uninstall

To uninstall FME Flow, complete the following steps:

1. Stop the FME Flow System Services
2. Drop the FME Flow Database (if required)

3. Uninstall FME Flow

Stop the FME Flow System Services

 **Note** Before proceeding, be sure to note your product's serial number. It may be required when you restore the configuration to the upgraded FME Flow. You can find the serial number on the [Licensing](#) page.


For the uninstall process to complete properly, you must first stop the FME Flow System Services, as well as any third-party web application and database services. For more information, see [Starting and Stopping FME Flow Manually](#).

If you have FME Flow components installed on multiple machines, you must stop the services on all applicable machines. For example, if you have installed multiple FME Flows for [fault tolerance](#), you must stop all services on their respective machines. Likewise, if you have [FME Engines installed on multiple machines](#), you must stop the FME Flow Engines service on all applicable machines.

To stop the web application server and database services depends on which ones you installed. If you installed FME Flow using the Express option, you can use the instructions at [Starting and Stopping FME Flow Manually](#) to stop these services as well. If you installed FME Flow using a [Distributed Installation](#), then you are using your own database server, and you may have opted to use your own web application server. To stop these services, refer to your product-specific documentation.

Drop the FME Flow Database

If your installation of FME Flow is a [Fault-Tolerant installation, or a Distributed Installation](#) in which the FME Flow Database is configured on your own database server, you must drop the [FME Flow Database](#). Use one of the following sets of SQL scripts, depending on the server you are using to run the database:

 **Note** It is only necessary to drop the FME Flow database. It is not necessary to uninstall the database server.

Microsoft SQL Server

Script location: `<FMEFlowDir>\Server\database\sqlserver`

Scripts:

`sqlserver_dropUser.sql`

`sqlserver_dropDB.sql`

Oracle

Script location: `<FMEFlowDir>\Server\database\oracle`

Scripts:

`oracle_dropUser.sql`

PostgreSQL

Script location: `<FMEFlowDir>\Server\database\postgresql`

Scripts:

`postgresql_dropDB.sql`

`postgresql_dropUser.sql`


Uninstall FME Flow

- On Windows, uninstall FME Flow through the Control Panel.
- On Linux, run the `uninstall.sh` script in the FME Flow installation directory. To ensure that everything gets uninstalled, run this script as root.

If you have FME Flow components installed on multiple machines, you must uninstall the components on all applicable machines. For example, if you have installed multiple FME Flows for [fault tolerance](#), you must uninstall FME Flow from all machines. Likewise, if you have [FME Engines installed on multiple machines](#), you must uninstall the FME Flow Engines from all applicable machines.

If you installed FME Flow by [distributing components](#) and opted to use your own web application server, it is not necessary to uninstall it. Instead, delete all folders and `.war` files from the web application installation directory that begin with "fme" (that is, are named like `fme*`).

Finally, if you installed FME Flow using any [Distributed/Fault Tolerant Installation](#), you must remove the [FME Flow System Share](#), assuming you have already performed a backup of FME Flow, or upgraded FME Flow to a second machine.

 **Note** On Linux, uninstalling removes the configuration of the following components, but does not uninstall them. They are reconfigured when FME Flow is reinstalled. If you want to use the more current version of these components, remove them completely from the system before re-installing:

- The PostgreSQL [FME Flow Database](#), if applicable.
- The NGINX reverse proxy.

Install

Depending on the type of installation you want to perform and your operating system, perform the installation and licensing procedures provided in one of the following sections:

- [Express Installation for Windows](#)
- [Express Installation for Linux](#)
- [Fault-Tolerant Installation](#)
- [Distributed Installation \(2-tier\)](#)
- [Distributed Installation \(3-tier\)](#)

Restore Configuration

To restore an FME Flow that you backed up previously, use the [Restore tool](#) in the FME Flow Web User Interface.

FME Engine Only Upgrade

[FME Engines](#) are the component of FME Flow that perform the actual data translations and transformations. At times, you might want to upgrade only your FME Engines and not any other FME Flow 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.

The procedures described in this section use the standard FME installer provided on the Safe Software web site, and can be performed with an official or beta release of the FME Engine installer.

FME Flow supports installing newer FME Engines if those engines share the same version year as the current installation of FME Flow. For example, on FME Flow 2023.0, the FME Engines may be upgraded to a future 2023.1.3, but may *not* be upgraded to a future 2024.0.

Upgrading FME Engines on Remote Engine Services

FME Flow supports installing newer FME Engines on [Remote Engine Services](#) if those engines share the same version year and major point (.#) version of the primary FME Flow installation. For example:

FME Flow Version	FME Engine Version on Remote Engine Services	Supported?
2023.0	2023.0	Yes
2023.0	2023.0.1	Yes
2023.0	2023.1.0	No
2023.1	2023.0	No

Obtaining the Installer

Install the upgraded FME Engine in its own path rather than overwriting the existing FME Engine. This way, you can reverse the upgrade if required.

To upgrade only the FME Engine component when installing an FME Flow environment, use a stand-alone FME Engine installer, not the FME Flow installer, and follow these steps.

To obtain the appropriate FME Engine Installer, visit the [FME Downloads page](#).

Install on Windows

To install FME Engine in a Windows environment

1. Run the FME installer on any computers that are running FME Engines that you want to upgrade.

Be sure to leave the existing FME Engine in place as a backup:

```
<FMEFlowDir>\Server\fme
```

and create a new directory for the upgraded FME:

```
<FMEFlowDir>\Server\FMEEngineUpgrade
```

2. Open Windows Explorer.
3. Make a copy of the file `<FMEFlowDir>\Server\FMEEngineUpgrade\fme.exe`, place that copy in the same `<FMEFlowDir>\Server\FMEEngineUpgrade` directory, and rename it `FMEEngine.exe`.
4. Edit the `<FMEFlowDir>\Server\processMonitorConfigEngines.txt` file.

Look for lines that start FME Engines, such as the following:

```
TEMPLATE_START_ENGINE="C:/Program  
Files/FMEFlow/Server/fme/FMEEngine.exe"
```

and change the path to the new FME, for example:

```
TEMPLATE_START_ENGINE="C:/Program  
Files/FMEFlow/Server/FMEEngineUpgrade/FMEEngine.exe"
```

5. Restart FME Flow.
6. To confirm that you are using the upgraded FME Engine from the new path and that you are using the FME build that you require, check the [process monitor engine log](#) (`fmeprocessmonitorengine.log`) on any computer running an FME Engine.

In the log, you should see your new FME Engine path and build number reflected in the FME Start-up sections.

Install on Linux

1. [Stop](#) FME Flow.
2. Move the currently-installed engines to a different directory. For example:

```
mv /opt/fme-engine-2020 /opt/fme-engine-2020-old
```

3. Perform the installation according to your Linux environment:

Debian (Ubuntu)

Using a GUI:

Double-click on the .deb install package, and follow the package manager to install FME Engines.

Using the command line:

Run the following commands under the root account:

```
apt install <installer_filename>.deb
```

For example:

```
apt install fme-engine-2023_2023.1.23619~ubuntu.20.04_amd64.deb
```

RedHat, Rocky Linux

Using a GUI:

Double-click on the .rpm install package, and follow the package manager to install FME Engines.

Using the Command Line:

Run the following command under the root account:

```
yum install <installer_filename>.rpm
```

For example:

```
yum install fme-engine-2023_2023.1-23619.el9.x86_64.rpm
```

4. [Start](#) FME Flow.

5. Use the [Engines](#) page to confirm that you are using the upgraded FME Engines. Under "Engines" verify that the "Build" number matches the build number of the filename for the Engines you have installed.

To Undo an Engine Upgrade

1. Run the following command under the root account to uninstall the newer FME Engine package:

Debian (Ubuntu)

```
apt remove <fme_engine_version>
```

For example:

```
apt remove fme-engine-2023
```

RedHat, Rocky Linux

```
yum remove <fme_engine_version>
```

For example:

```
yum remove fme-desktop-2023
```

2. Move the older FME Engines back to the current directory. For example:

```
mv /opt/fme-engine-2020-old /opt/fme-engine-2020
```

Upgrading a Kubernetes Deployment of FME Flow

With little downtime, you can upgrade a Kubernetes deployment of FME Flow to a later version using the following procedure:

- Launch a new FME Flow of the desired build version in a new namespace.
- Backup the current FME Flow and restore it to the new FME Flow.
- Update the ingress hostname from the old FME Flow to the new FME Flow.
- Remove the old FME Flow.

⚠ **Warning** You may have [Schedules](#), [Automations](#) workflows, or [Notifications](#) that would conflict with themselves during the time that both the new and old FME Flows are running. Disable these tools before proceeding with a backup.

📄 **Note** See also [Upgrading to a Patch Build in Kubernetes](#).

Performing the Upgrade

1. Create a new namespace for the new version of FME Flow:

```
kubectl create namespace <namespace-new-version>
```

For example:

```
kubectl create namespace fme-flow-2024-1-1
```

2. If your FME Flow uses an external PostgreSQL database for the FME Flow Database, copy the admin password secret, using the namespace of the current version of FME Flow, to the new version. (If your FME Flow uses the PostgreSQL container that is deployed with the Helm chart, skip this step.)

```
kubectl get secret postgresql-admin-password --namespace  
<namespace-old-version> --export -o yaml | kubectl apply --  
namespace=<namespace-new-version> -f -
```

For example:

```
kubectl get secret postgresql-admin-password --namespace fme-flow -  
--export -o yaml | kubectl apply --namespace=fme-flow-2024-1-1 -f -
```

3. Make a copy of the values .yaml file for the current FME Flow and update the following values:
 - `deployment.hostname`: Set to a temporary hostname. (You will reset it later in this procedure.)
 - `fme-flow.image.tag`: Set to the container image tag you want to deploy for the upgraded FME Flow.

- If your FME Flow uses an external PostgreSQL database for the FME Flow Database, set the following values to be different from the ones in use for the current FME Flow:

- `fmeflow.database.name`
- `fmeflow.database.username`

4. Update your Helm repository:

```
helm repo update
```

5. Deploy FME Flow in the new namespace using Helm and the updated values .yaml file.

```
helm install --namespace <namespace-new-version> -n <helm-  
deployment-name> safesoftware/fmeflow -f <new-values-filename>
```

For example:

```
helm install --namespace fmeflow-2024-1-1 -n fmeflow  
safesoftware/fmeflow -f values-2024.1.1.yaml
```

6. From the [FME Flow Web User Interface](#), perform the following:

- a. [License](#) FME Flow.
- b. [Backup](#) the old FME Flow.
- c. [Restore](#) the backup to the new FME Flow.

7. Change the DNS of the old FME Flow to a different, temporary URL. Doing so will allow you to update the DNS of the new FME Flow to your production URL.

- a. In the old values .yaml file, change the value of `deployment.hostname`.
- b. Upgrade the Helm chart of the previous deployment.

```
helm upgrade --namespace <namespace-old-version> <helm-  
deployment-name> safesoftware/fmeflow -f <old-values-filename>
```

For example:

```
helm upgrade --namespace fmeflow fmeflow safesoftware/fmeflow  
-f values-2024.0.0.yaml
```

8. Update the DNS of the new FME Flow to your production URL.
 - a. In the new values .yaml file, change the value of `deployment.hostname`.
 - b. Upgrade the Helm chart of the new deployment.

```
helm upgrade --namespace <namespace-new-version> <helm-deployment-name> safesoftware/fmeflow -f <new-values-filename>
```

For example:

```
helm upgrade --namespace fmeflow-2024-1-1 fmeflow safesoftware/fmeflow -f values-2024.1.1.yaml
```

9. Verify that the upgraded FME Flow works as expected. If you find issues, you can switch back the `deployment.hostname` values in the old and new values .yaml files so that your production URL points back to the old FME Flow.
10. Remove the old version of FME Flow.
 - Helm v2: `helm del --purge <helm-deployment-name>`
 - Helm v3: `helm uninstall <helm-deployment-name> -n <namespace_name>`

For example:

- Helm v2: `helm del --purge fmeflow`
- Helm v3: `helm uninstall fmeflow -n fmeflow`

11. If your FME Flow uses an external PostgreSQL database for the FME Flow Database, drop the database for the old FME Flow.

Optimizing FME Flow

Configuring the Number of FME Engines to Start	315
Adding FME Engines on a Separate Machine	315
Remote Engine Services	322
Changing the Database Provider for the FME Flow Database	327
Moving the Web Applications Server to Another Machine	333
Using a Reverse Proxy with FME Flow	335



Configuring the Number of FME Engines to Start

[FME Engines](#) connect to an [FME Flow Core](#) and process requests from the Core. Adding FME Engines to start with the Core provides these benefits:

- Increases the Core's ability to perform concurrent processing of requests.
- Helps with [Job Recovery](#).


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.

By default, FME Flow is configured to start two FME Engines, if two licenses are available. If only one license is available, only one FME Engine starts.

To add or remove FME Engines to Start

Open the [Engines](#) page of the Web User Interface, and follow the steps under [Configuring the Number of FME Engines to Start](#).

Adding FME Engines on a Separate Machine

 **Warning** This topic applies only to traditional FME Flow installations. If your FME Flow runs in a containerized environment, such as Docker Compose or a Kubernetes cluster, this architecture is not supported.

- **Skill Level:** Intermediate
- **Estimated Time Required:** 30-45 minutes
- **Prerequisites:**
 - Existing [Express, Distributed, or Fault-Tolerant](#) FME Flow installation.
 - Access to system, network, and FME Flow administrators.


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

When adding FME Engines, keep in mind the following:

- This approach requires the engine machines to be in the same network as your original FME Flow installation, and in the same data center or geographically close. To run jobs outside of your network, use [Remote Engine Services](#) instead.
- Adding FME Engines that do not match the primary release version of the [FME Flow Core](#) is not supported. The primary release version refers to the first decimal value following the release year. For example, if the release version of the FME Flow Core is 2020.0, FME Engines from release versions 2020.0.1 or 2020.0.2 are supported. However, FME Engines from release versions 2020.1 or 2021 are not supported.
- We recommend installing all FME Engines on systems that are synchronized to the same time zone as other FME Engines and the FME Flow Core. If time zones differ, unexpected issues may arise, including:
 - Difficulty accessing the [FME Flow Web User Interface](#).
 - Improper timing of [Schedule Initiated](#) triggers.
 - Inconsistent or misleading timestamps in log files (accessed from [Resources](#)).


For more information, see [About Times and Time Zones](#).

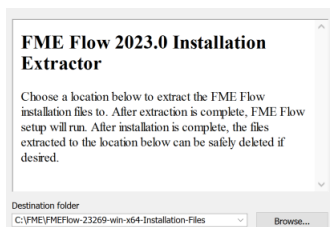
In the instructions below, the computer that hosts the FME Flow Core is the `<coreHost>`.

 **Note** In a fault-tolerance environment, we recommend assigning unique names to FME Engines. However, if multiple FME Engine hosts have the same FME Engine name, the queue server configuration applies to all FME Engines with the same name in the same way regardless of which host it resides on. For more information, see [Job Queues](#).

Windows

1. [Obtain the Installer](#). Be sure to obtain the FME Flow Installer (not the Engine Installer).
2. Start the installer on another computer (`<engineHost>`).
3. On the Installation Extractor dialog, specify a *Destination folder* in which to extract installation files. Note that this folder *only* specifies where installation files are extracted—it does *not* specify where FME Flow program files are installed. That location is specified later in the installer.

-  **Tip** Take note of the specified *Destination folder* location. Following installation, you may wish to:
- Review the installation .log file.
 - Remove these files to save disk space.



Click **Install**.


When extraction is complete, the installer opens. Proceed through the installer to the Choose Setup Type dialog.

4. On the Choose Setup Type dialog, select the Distributed Engine setup type.
5. On the Destination Folder dialog, specify the directory to install the FME Engines.
6. On the User Account page, specify the Windows user account that will run the `<engineHost>`. This account must have:
 - Read/Write access to the FME Flow System Share that you specify in the next dialog. The System Share is the location where FME Flow stores [Repositories](#) and [Resources](#) files.

- Read/Write access to the FME Flow installation directory that you specified in the previous dialog.
- "Log on as a service" rights on the `<engineHost>`. For more information, see [this Microsoft Docs article](#).

7. On the Existing FME Flow System Share dialog, provide the UNC path to the FME Flow System Share directory.

If the `<coreHost>` machine is an Express [installation](#) of FME Flow, the System Share is written, by default, to `%ALLUSERSPROFILE%\Safe Software\FME Flow\`, unless specified otherwise.


 **Note** Unless modified, `C:\ProgramData` is the default value of the `%ALLUSERSPROFILE%` environment variable.

In all cases, enter the UNC path to the System Share, including the final `Safe Software\FME Flow` directories, which are created by default. For example, `\\MyServer\Safe Software\FME Flow`.

8. On the FME Flow Core Hostname dialog, enter the hostname of the `<coreHost>` computer. Additionally, if the machine hosting this installation of FME Engines is identified by a name other than that specified under *Advanced*, check *Advanced* and specify the alternate name.
9. On the Database Server Type dialog, specify the database type that you specified when installing the FME Flow Core. If you specified a database other than the "FME Flow Database," enter the connection parameters in the next dialog.
10. On the Database User dialog, specify the user name and password for the FME Flow Database that you specified when installing FME Flow.
11. Complete the remaining dialogs of the FME Flow Installer.

12. If this *<engineHost>* is connecting to a *<coreHost>* that also hosts the built-in FME Flow database and the initial FME engines, as with an [Express Installation](#), allow the necessary connection to this machine:

- a. On the *<coreHost>*, open file `pg_hba.conf`, located in `%ALLUSERSPROFILE%\Safe Software\FME Flow\pgsql\data`.

 **Note** Unless modified, `C:\ProgramData` is the default value of the `%ALLUSERSPROFILE%` environment variable.

- b. Change the following lines:

```
host    all        all        127.0.0.1/32      md5
```

```
host    all        all        ::1/128           md5
```

to:

```
host    all        all        0.0.0.0/0         md5
```

```
host    all        all        ::/0              md5
```

- c. [Restart](#) the FME Flow Database and FME Flow Engines services.


13. If the *<coreHost>* machine was an [Express Installation](#) of FME Flow, complete the following steps on the *<coreHost>* to configure the FME Flow Web Services:

- a. Open the corresponding `propertiesFile.properties` file for the Data Download, Data Streaming, Job Submitter, and Notification services, located at `<FMEFlowDir>\Utilities\tomcat\webapps\<service_name>\WEB-INF\conf\propertiesFile.properties`.

For example, the properties file for the Data Download Service is at `<FMEFlowDir>\Utilities\tomcat\webapps\fmadatadownload\WEB-INF\conf\propertiesFile.properties`

- b. For each properties file, locate the parameter `REQUEST_DATA_DIR` and change the value from a local file system path (for example, `C:/ProgramData/Safe Software/FME Flow//resources/system/temp/requestdata`) to a UNC path

that references the `<coreHost>` (for example, `//MyHost/FMEFlow/resources/system/temp/requestdata`).

 **Note** Use forward slashes only.

- c. After saving the changes in the properties file for each service, [restart](#) the FME Flow Application Server service.
14. On the `<coreHost>` machine, open configuration file [fmeFlowConfig.txt](#), and update the `FME_SERVER_HOST_NAME` parameter to a value that fully resolves to the IP address of the `<coreHost>` machine.
15. [Restart](#) the FME Flow Core service.

Linux

1. If not already accomplished, mount the directory that holds the FME Flow System Share. (The System Share holds FME Flow [Repositories](#) and [Resources](#)):

```
sudo mount -t nfs <coreHost>:/<FMEFlowSystemShare> /<mntPath>
```
2. [Obtain the Installer](#). Be sure to obtain the FME Flow Installer (not the Engine Installer).
3. Start the FME Flow Installer:

```
sudo ./<installationFile>
```
4. From the setup type list, choose **Engine**.
5. The installer prompts you for a location to install the FME Flow Engine components. This can be on the local machine.
6. The installer prompts you for the name of the `<coreHost>`. In a fault-tolerant installation, the primary `<coreHost>` should always be specified. Do not specify the failover host.
7. The installer prompts you for the FME Flow System Share location. Enter the mounted directory created in step 1.

8. Enter the database type you specified when installing the FME Flow Core. If you specified a database other than the "FME Flow Database", enter the connection parameters.
9. Enter the user name and password for the FME Flow Database that you specified when installing FME Flow.
10. Complete the remaining dialogs of the FME Flow Installer.
11. Manually [start FME Flow](#), and configure it to start automatically.
12. If this *<engineHost>* is connecting to a *<coreHost>* that also hosts the built-in FME Flow database and the initial FME engines, as with an [Express Installation](#), allow the necessary connection to this machine:
 - a. On the *<coreHost>*, open file `pg_hba.conf`, located in `<FMEFlowSystemShare>\pgsql\data\`.
 - b. Change the following lines:



```
host    all    all    127.0.0.1/32    md5
host    all    all    ::1/128         md5
```

to:

```
host    all    all    0.0.0.0/0       md5
host    all    all    ::/0            md5
```
 - c. [Restart](#) the FME Flow Database and FME Flow Engines services.
13. Ensure that the user account that runs *<engineHost>* has the necessary permissions to access the FME Flow System Share and the FME Flow install directory. For more information, see [Directory and Account Permissions](#).
14. On the *<coreHost>* machine, open configuration file [fmeFlowConfig.txt](#), and update the `FME_SERVER_HOST_NAME` parameter to the fully-qualified domain name or IP address of the *<coreHost>* machine.
15. [Restart](#) the FME Flow Core service.

Verify the Installation

The installation on `<engineHost>` starts two FME Engines by default. The FME Engines start and register with the FME Flow Core running on `<coreHost>`. All licensed engines are listed on the [Engines](#) page of the Web User Interface.

 **Note** You may have to disable any firewalls that are running on this computer.

You can also use the Engines & Licensing page to perform the following tasks:

- To license FME Engines, see [Licensing](#).
- If you want this computer to start more or less FME Engines, see [Configuring the Number of FME Engines to Start](#).

For more information, see [Verify the Installation](#).

Remote Engine Services

Select **Engine Management > Remote Engine Services**.

Remote Engine Services allow you to use queues to run jobs on separate, specialized installations of FME Flow that may be closer to your data, while bypassing [Queue Control](#) rules. You can install Remote Engine Services and connect to them on servers that are part of your network, available outside your network on accessible endpoints, or in the Cloud, such as Azure Functions or Google Cloud Functions. Customers of FME Flow Hosted can also connect to Remote Engine Services. Unlike [Adding FME Engines on a Separate Machine](#), Remote Engine Services may be especially useful if you want to run jobs on servers outside your network, while maintaining your primary FME Flow installation behind a firewall.

Capabilities and Limitations

When running jobs on Remote Engine Services, keep in mind the following:

- [FME Engines](#) of Remote Engine Services can run on any type of [license](#), Standard or Dynamic. Multiple engines of both types are supported, depending on licensing terms.
- Connecting to Remote Engine Services through a proxy is supported.
- FME Flow queues jobs when Remote Engine Services are offline, and submits them for processing once they are online.
- Remote Engine Services can accept workspaces that are registered to run only the [Job Submitter Service](#). Other [FME Flow Web Services](#) are not supported.
- Remote Engine Services can run jobs submitted from [Run a Workspace](#) or [Run a Dynamic Workspace](#) automations actions, [Schedules](#), and [Workspace Apps](#).
- Remote Engine Services cannot run [custom formats](#) or [linked transformers](#).
- A Remote Engine Service can connect with only one FME Flow instance at a time. Simultaneous connections with multiple FME Flow instances are not supported.

Getting Started with Remote Engine Services

To target remote engine services to run jobs:

1. Install remote engine services on your remote server.
2. FME Flow Hosted users only: Request a license for a Remote Engine Service.
3. Create queues (if necessary).
4. Create a connection to a Remote Engine Service and associate it with one or more queues.
5. Run the job on one of the specified queues.

Install Remote Engine Services

Follow the instructions [here](#) to install remote engine services on your remote server. To obtain a remote engine service installer, visit the [FME downloads page](#). Remote engine service installers are named beginning with `fme-flow-remote-engine-*`.

FME Flow Hosted Only: Request a License for a Remote Engine Service

On your FME Flow Hosted instance connection, request a license from the [Licensing](#) page, under the Remote Engine Service heading.

Create Queues (if necessary)

You may already have queues that you use to target jobs to FME Engines based on existing [Queue Control](#) rules. When specified explicitly in a [Run Workspace](#) directive, those same queues can run jobs on remote engine services, effectively bypassing queue control rules. Alternatively, you can create new queues and use those instead. To create new queues, select the **Queues** tab, and click **New**.


Create a Remote Engine Services Connection and Associate it with Queues

1. On the Remote Engine Services tab of the Engine Management page, click **Create** and complete the following fields:
 - *Name*: Specify a name of your choice for the Remote Engine Service connection.
 - *URL*: The URL of the remote server on which the FME Flow Remote Engine Service was installed.
 - *Username*: The username of the FME Flow Remote Engine Service account.
 - *Password*: The password to authenticate the account *Username*.
 - *Queues*: The queues to associate with the Remote Engine Services connection. When one of these queues is specified explicitly in a run workspace directive, the job is routed to this connection.
 - *Standard Engines*: Specify the number of Standard [FME Engines](#) you want to start on the Remote Engine Service for running jobs.



Note This number is limited to your [Licensing](#) terms.

- *CPU Usage Engines*: Specify the number of Dynamic (CPU Usage) [FME Engines](#) you want to start on the Remote Engine Service for running jobs.


 **Note** This number is limited to your [Licensing](#) terms.

- *Proxy Enabled:* To connect to the Remote Engine Service through a proxy, check this box and complete the following:
 - *Host:* The proxy server hostname.
 - *Port:* The port through which communication with the proxy server takes place.
 - *No Proxy For:* (optional) A list of hostnames and IP addresses that will be accessed through a direct connection, bypassing the proxy. To avoid entering multiple hosts in the same domain, hosts can be prefixed and appended with the wildcard character `*`.

To delimit the list, use commas, semicolons, or press the tab or enter keys. Do not copy-paste a list of entries. To delineate properly in the display, each entry must be added separately.

Example:

No Proxy For (optional) | localhost x 127.0.0.1 x *.example.* x MyHostname.MyDomain.* x

 **Note** Specifying `localhost` does not include, by default, `127.0.0.1` or `:::1`. To bypass the proxy when the local host is referenced, you must explicitly specify any other expected forms for referencing local traffic.

- *Authentication Required:* Click if authentication is required to access the proxy server, and complete the following:
 - *Username:* The account username for running the proxy server.
 - *Password:* The password for the proxy server account.
 - *Authentication Method:* The authentication method used by the proxy server, either **Basic**, **Digest**, or **NTLM**.

2. (Optional) If the Remote Engine Service is currently online, click **Test** to ensure the specified credentials are valid.
3. Click **OK**.

The Remote Engine Services connection appears on the Engine Management page. You may need to refresh the page until the STATUS icon shows a green check mark.

Run a Job with the Remote Engine Services Connection

From any of the following run workspace scenarios (under Advanced settings), specify the desired *Job Queue* associated with the Remote Engine Services connection that you want to run the job:

- [Run Workspace](#)
- [Run a Workspace](#) or [Run a Dynamic Workspace](#) automations action
- [Schedules](#)

The specified queue overrides the queue that would otherwise be assigned based on queue control rules, and instead routes the job to the Remote Engine Services connection.

Administering Remote Engine Services

Administrators of Remote Engine Services installations can access the [Jobs](#), [Analytics](#), and [Engines](#) pages of the [FME Flow Remote Engine Service Web User Interface](#).

Resetting the Cache

When jobs are submitted to remote engines, FME Flow automatically uploads contents to the remote engine service that are necessary to run the job, including the workspace and its dependencies, such as resources, packages, web connections, web services, and database connections. To clear this cache, select the Remote Engine Service for the cache you want to clear, click **Actions**, and select **Reset Cache**.

Resetting the Remote Engine Service Password

You can update the password of a Remote Engine Service user account from the Remote Engine Services page on the primary FME Flow installation.

Click on the Remote Engine Service to open it. On the Editing page, click Change Password. The new password must comply with the [Password Policy](#) configuration.

Removing Remote Engine Services


Select the Remote Engine Service you want to remove, click **Actions**, and select **Remove**.

Changing the Database Provider for the FME Flow Database

To change the database provider for the [FME Flow Database](#), perform an “in-place” backup and restore procedure:

1. [Backup Your FME Flow Configuration](#)
2. [Configure the New Database Server](#)
3. [Configure the Database Connection](#)
4. [Restart FME Flow](#)
5. [Restore Your FME Flow Configuration](#)
6. [Remove Dependency, Disable, and Stop the Previous Database Service](#)

FME Flow supports PostgreSQL (recommended), SQL Server, or Oracle databases.

 **Note** PostgreSQL is the recommended database server with FME Flow, offering enhanced performance and stability through targeted optimizations.


Backup Your FME Flow Configuration

Perform a [backup](#) of your FME Flow configuration.

Configure the New Database Server

Run the necessary database configuration scripts and post-configuration scripts. Follow the steps in [Configure the FME Flow Database on a Separate Database Server](#).

Configure the Database Connection

 **Note** If using an Oracle database server, you must obtain the Oracle Database JDBC driver and place it in the following directory:

Windows: `<FME FlowDir>\FMEFlow\Utilities\jdbc`

Linux: `/opt/fmeflow/Utilities/jdbc`

Driver versions recommended: 12.1 (ojdbc7.jar), 12.2 (ojdbc8.jar), 18.3 (ojdbc8.jar), 19.3 (ojdbc8.jar), 19.3, 19.6, 19.7 (ojdbc8.jar).

Driver versions *not* recommended: All ojdbc10.jar versions. For more information, see <https://www.oracle.com/technetwork/database/application-development/jdbc/downloads/index.html>.

1. Encrypt the database password you generated in the previous step. In an initial installation of FME Flow, this password is encrypted in installation files by default. However, it is no longer encrypted when the database server is updated. Follow the procedure in [Encrypting the FME Flow Database Password](#).
2. Locate the [fmeDatabaseConfig.txt](#) configuration file.
3. Under the heading 'FME SERVER SETTINGS START', locate the section titled 'Database Connection' and update the parameters for the database you want to use for your repository.

Parameters

DB_TYPE - Identifies the database server: postgresql, sqlserver, oracle.

DB_DRIVER - The JDBC driver name used for connecting to the database.

DB_JDBC_URL - The JDBC URL used for connecting to the database.

DB_USERNAME - The database user name.

DB_PASSWORD - The database user password.

DB_CONNECT_EXPIRY - The database connection expiry time, in seconds.

DB_SQLSTMTS_PATH - The path to the SQL statement resource bundle.

Examples

Example - PostgreSQL

```
DB_TYPE=postgresql
DB_DRIVER=org.postgresql.Driver
DB_JDBC_URL=jdbc:postgresql://localhost:5432/fmeflow
DB_USERNAME=fmeflow
DB_PASSWORD=fmeflow
DB_CONNECT_EXPIRY=60
DB_SQLSTMTS_PATH=C:/Apps/FMEFlow/Server/database
```

Examples - SQL Server

To use a local installation of SQL Server Express

Prerequisites:

- In SQL Server Configuration Manager, ensure the *Protocols for SQLExpress* setting for *TCP/IP* traffic is set to **Enable**. (In SQL Server Express, this setting is disabled by default.)
- Ensure the Windows service *SQL Server Browser* is enabled and running for SQL Server Express. This service is required for the JDBC driver to locate a dynamic port. Alternatively, use SQL Server Configuration Manager to configure a fixed port, and specify `port=<xxxx>` in the `DB_JDBC_URL` parameter setting, where `<xxxx>` is the value of the specified port. For example, `port=1433`.
- In SQL Server Management Studio, set *Server authentication* to **SQL Server and Windows Authentication Mode**. This configuration is required because FME Flow uses the `DB_USERNAME` property to authenticate.

```
DB_TYPE=sqlserver
DB_DRIVER=com.microsoft.sqlserver.jdbc.SQLServerDriver
DB_JDBC_
URL=jdbc:sqlserver://localhost;instanceName=SQLEXPRESS;databaseName=
fmeserver;encrypt=true;trustServerCertificate=true
```

```
DB_USERNAME=fmeserver
DB_PASSWORD=$FMEserver
DB_CONNECT_EXPIRY=60
DB_SQLSTMTS_PATH=C:/Apps/FMEServer/Server/database
```

Note: To resolve encryption issues due to invalid certificates, specify `encrypt=false` in place of `encrypt=true;trustServerCertificate=true`.

To use a named instance of a SQL Server

Specify the `instanceName` property with the `DB_JDBC_URL` parameter:

```
DB_TYPE=sqlserver
DB_DRIVER=com.microsoft.sqlserver.jdbc.SQLServerDriver
DB_JDBC_
URL=jdbc:sqlserver://<
  ServerName>;databaseName=<database>;instanceName=<InstanceName>
DB_USERNAME=fmeflow
DB_PASSWORD=$FMEflow
DB_CONNECT_EXPIRY=60
DB_SQLSTMTS_PATH=C:/Apps/FMEFlow/Server/database
```

To use a domain service account instead of a named database user or ***fmeflow*** (the default database username)

Complete the following steps:

1. Specify the `integratedSecurity` property with the `DB_JDBC_URL` parameter. FME Flow uses the *Log on as* credentials of the FME Flow Core and FME Flow Application Server [services](#) to connect to the SQL Server database. Do not set the `DB_USERNAME` or `DB_PASSWORD` parameter values.

```
DB_TYPE=sqlserver
DB_DRIVER=com.microsoft.sqlserver.jdbc.SQLServerDriver
```

```
DB_JDBC_  
URL=jdbc:sqlserver://<  
  ServerName>;databaseName=<database>;integratedSecurity=true  
DB_USERNAME=  
DB_PASSWORD=  
DB_CONNECT_EXPIRY=60  
DB_SQLSTMTS_PATH=C:/Apps/FMEFlow/Server/database
```

2. Go to directory <FMEFlowDir>\Server\fme\ and complete the following:
 - a. Locate the file with a name that begins mssql-jdbc-auth*.dll
 - b. Copy the file to the <FMEFlowDir>\Server\lib\fmeutil\ directory.
 - c. In the <FMEFlowDir>\Server\lib\fmeutil\ directory, rename the file to sqljdbc_auth.dll

To specify the database connection port

Specify the port after <ServerName> in the `DB_JDBC_URL` parameter. For example:

```
DB_JDBC_URL=jdbc:sqlserver://<ServerName>;port=1755;
```

To specify an Always On availability group

If using an Always On availability group, include the `MultisubnetFailover` property with the `DB_JDBC_URL` parameter:

```
DB_JDBC_  
URL=jdbc:sqlserver://<  
  ServerName>;databaseName=<database>;MultisubnetFailover=true
```

Examples - Oracle

To specify an on-premise database with a System ID (SID)

Use the following syntax for the `DB_JDBC_URL` parameter:

```
DB_JDBC_URL=jdbc:oracle:thin:@<host>:<port>:<sid>
```

For example:

```
DB_TYPE=oracle
DB_DRIVER=oracle.jdbc.driver.OracleDriver
DB_JDBC_URL=jdbc:oracle:thin:@localhost:1521:orcl
DB_USERNAME=fmeflow
DB_PASSWORD=fmeflow
DB_CONNECT_EXPIRY=60
DB_SQLSTMTS_PATH=C:/Apps/FMEFlow/Server/database
```

To specify an on-premise database with a service name

Use the following syntax for the `DB_JDBC_URL` parameter:

```
DB_JDBC_URL=jdbc:oracle:thin:@<host>:<port>/<servicename>
```

For example:

```
DB_JDBC_URL=jdbc:oracle:thin:@hostname.mydomain.com:1521/mydatabase1
```

To specify an Oracle Cloud database

Use the following syntax for the `DB_JDBC_URL` parameter:

```
DB_JDBC_URL=jdbc:oracle:thin:@<ip>:<port>/<servicename>
```

For example:

```
DB_JDBC_URL=jdbc:oracle:thin:@129.44.44.2:1521/orcl_
iad2fd.mycorpz1.mycorp.myvcn.com
```

Restart FME Flow

[Restart](#) all of the FME Flow System Services.

Restore Your FME Flow Configuration

Perform a [restore](#) of your FME Flow configuration:

Remove Dependency, Disable, and Stop the Previous Database Service

If the database service from which you migrated is the PostgreSQL database that was included in a default installation of FME Flow (such as an express install), you must remove

the dependency of the FME Flow Core on that database, and then disable and stop it. For more information, see [Removing the FME Flow Core Dependency on the FME Flow Database](#).

Removing the FME Flow Core Dependency on the FME Flow Database

You may need to disable and stop the service of the PostgreSQL database that is included in a default installation of FME Flow (known as the "FME Flow Database"). A common reason for performing this action is after [Changing the Database Provider for the FME Flow Database](#). However, shutting down the old FME Flow Database also attempts to shut down the FME Flow Core, unless this service dependency is removed.

1. Remove the Dependency of the FME Flow Core on the FME Flow Database
 - a. Open a Command Prompt as administrator (Start Menu > Command Prompt > right click and select "Run as administrator").
 - b. Type the following command into the Command Window and press Enter.

```
sc config "FME Server Core" depend= /
```
2. Disable and Stop the FME Flow Database
 - a. Open the Windows Services (Control Panel > Administrative Tools > Services)
 - b. Open the FME Flow Database service.
 - c. In the Properties dialog, change *Startup Type* to **Disabled**.
 - d. Under *Service status*, click **Stop**.
 - e. Click **Apply**.

Moving the Web Applications Server to Another Machine

To improve performance and free up resources, you may wish to move the web applications server, which runs the [FME Flow Web Services](#), to another machine.

⚠ Warning We recommend installing the web application server on a system that is synchronized to the same time zone as the FME Flow Core. Differing time zones may result in improper timing of [Schedule Initiated](#) triggers.

This move requires the following steps:

1. [Install a web applications server on the new host machine \(optional\)](#)
2. [Install the FME Flow Web Services on the new host](#)
3. [Update Services URLs](#)
4. [Configure CORS filters \(optional\)](#)

Install a web applications server on the new host machine (optional)

To run the FME Flow Web Services, you can use the Apache Tomcat web application server that is automatically provided when you install the services. In this case, skip this step and proceed to the next step.

Alternatively, you can install and use your own Apache Tomcat web application server for running the FME Flow Web Services. FME Flow supports Apache Tomcat version 9.0.x.

You must know the port number on which your web application server receives incoming requests and the path to the directory where your web application server stores web application `.war` files. (This directory is sometimes called `webapps`.)

Install the FME Flow Web Services on the new host

On the new web applications server host, follow the instructions under [Install the FME Flow Web Services](#), for either Windows or Linux.

Update Services URLs

On the machine hosting the [FME Flow Core](#):

1. Open the [fmeFlowConfig.txt](#) configuration file.

Locate the FME_SERVER_WEB_URL directive and update the URL value to the new web applications server host. For example:

```
FME_SERVER_WEB_URL=http://MY_WEBAPP_HOST:80
```

2. Open the [FME Flow Web User Interface](#).
 - a. [Login](#) and navigate to [Services](#).
 - b. Click on each service to open the Editing Service page.
 - c. In the URL Pattern field, update the URL value to the new web applications server host, and click OK when done.

For example, to update the Data Download Service:

```
http://MY_WEBAPP_HOST/fmedatadownload
```

Configure CORS filters (optional)

If you are a web application developer, you may want to configure [Cross-Origin Resource Sharing](#) (CORS) filters, so that your application can bypass manual authentication when accessing the FME Flow Web Services on another domain.

See Also

- [Changing the Web Application Server Host Name and Port](#)

Using a Reverse Proxy with FME Flow

A reverse proxy forwards requests from the internet or intranet to a server that fulfills the requests. Often a reverse proxy configured with SSL provides a secure entry point into a network through a DMZ.

The instructions for configuring your reverse proxy depend on the brand of web server or other product you select to fill this role.

Use these instructions to:

1. Set up your reverse proxy to route traffic to FME Flow.
2. Configure FME Flow to recognize the reverse proxy URL.

Set Up the Reverse Proxy

Routing Traffic to the FME Flow Web Services

Recommended

- Route two-way traffic between the [FME Flow Core](#), [FME Flow Web Services](#), and the reverse proxy.

Optional

- Configure timeouts for client- and server-side inactivity. When configuring timeouts, consider that longer timeouts on the server are appropriate for time-consuming operations, particularly running workspaces that take a long time to complete.

Routing Traffic to the WebSocket Server

An FME Flow [WebSocket Server](#) runs on each FME Flow instance, but only one WebSocket Server should be used at a time. The WebSocket Server is used mainly for Notification Service [Topic Monitoring](#).

Required

- Route traffic on port 7078 to one WebSocket Server, and switch to others if the one in use fails. If the reverse proxy does not support automatic failover to backup servers, then manual intervention is required to route to a healthy WebSocket Server.

Recommended

- Set longer client- and server-side timeouts to allow for longer WebSocket connections.

Configure FME Flow with the Reverse Proxy URL

To configure FME Flow to recognize the reverse proxy URL, perform the following tasks in the [FME Flow Web User Interface](#):

Update Service URLs

- On the [Services](#) page, click **Change All Hosts**. In the Change All Hosts dialog, enter the public reverse proxy URL, and click **OK**.

Update Topic Monitoring URL

1. On the [Network & Email](#) page, expand Topic Monitoring.
2. Under Fault Tolerant Configuration, update the *Internal WebSocket URL* field with the public reverse proxy URL. [Automations](#) external actions (and [Notification Service](#) Subscribers) use this URL to communicate with the FME Flow WebSocket Server. The reverse proxy URL redirects requests to the active WebSocket Server.

3. If necessary, update Advanced Routing settings:

- *External WebSocket Host*: The reverse proxy name. For example:

`fme-flow-reverseproxy`

This setting does not likely need to be changed, and should only be set if you use separate hosts to route HTTP and WebSocket traffic to FME Flow.

- *External WebSocket Port*: If the WebSocket Server is accessed through a port other than 7078, specify the port number here. Otherwise, leave blank.

4. Click **Save**.

Update Reset Password URL (Optional)

If you plan to enable the [Reset Password](#) feature, update the *Public URL* field on the Reset Password page with the public reverse proxy URL, and click **Save**.

Securing FME Flow

Security Basics	339
Configuring Integrated Windows Authentication	365
Troubleshooting Authentication Services	370
Importing a CA Certificate for SSL Connections	382
Running the FME Flow System Services Under Different Accounts (Windows)	383
Encrypting the FME Flow Database Password	384



Security Basics

To get started with FME Flow security, there are three main tasks to perform:

- Configuring [Role-Based and User-Based Access Control](#)
- [Configuring Authentication for the FME Flow Web Services](#)
- [Configuring for HTTPS](#)

Role-Based and User-Based Access Control

FME Flow security is based on two primary concepts:

- **Users:** The individual accounts that access FME Flow. When FME Flow is installed for the first time, [default user accounts](#) are created. Note that the default *Status* of these accounts, except the **admin** account, is **Disabled**.
- **Roles:** Comprised of one or more users.

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

Optionally, you can incorporate your organization's directory server (for example, Windows Active Directory) groups and users into your FME Flow security configuration.

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 Flow 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.

FME Flow provides a set of default roles:

[Click to view default roles](#)

Role	Description	User
fmeadmin	Provides full access to FME Flow, including the Web User Interface.	admin
fmeauthor	Provides workspace authors access to FME Flow to publish, author, and test new workspaces.	author
fmeguest	Provides the minimal access to FME Flow required to run workspaces.	guest
fmesuperuser	Authorized to access all resources of FME Flow, including existing and newly-created resources.	admin
fmeuser	Provides users access to the Web User Interface and Web Services.	user


On the [Roles](#) page of the Web User Interface, an administrator can:

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

On the [Active Directory](#) page of the Web User Interface, an administrator can integrate the organization's Windows Active Directory, LDAP, or other directory server groups and users into its FME Flow security configuration.


User-Based Access

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

 **Note** In versions of FME Flow prior to 2023.0, the guest user account could provide unauthenticated access to the FME Flow Web Services. For security reasons, this feature is removed.

User Ownership

Anything a user creates in FME Flow, 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 Flow 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.

 **Note** To change ownership of an item, edit it from the [Items](#) page. Access to the Items page requires Manage permission in User Management.

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 in roles.

- Configure permissions of users.

On the [Active Directory](#) page of the Web User Interface, an administrator can integrate the organization's Windows Active Directory, LDAP, or other directory server users and groups into its FME Flow security configuration.

Shared Access

Through sharing, users can grant different levels of access to items in FME Flow to other users or roles. A user can share an item if their account is enabled for sharing, and either of the following is true:


- The user owns the item.
- The user has Manage permission in User Management (usually granted to an administrator).


See Also

- [Configuring Authentication for the FME Flow Web Services](#)
- [Authentication and Authorization](#)

Default User Accounts and Passwords

The FME Flow installer creates the following user accounts by default:

 **Note** The default *Status* of these accounts, except the **admin** account, is **Disabled**.


Username/Password	Description	Default Status	Role
admin/admin	<p>Access to all FME Flow resources.</p> <div data-bbox="553 453 927 1020" style="border: 1px solid #007080; background-color: #e6f2ff; padding: 10px;"> <p> Note You are prompted to update the admin password upon initial use, based on the default Password Policy configuration.</p> </div>	Enabled	fmeadmin, fmesuperuser
author/author	An account intended for FME Workbench authors.	Disabled	fmeauthor
guest/guest	An account that provides the minimal access to FME Flow required to run workspaces.	Disabled	fmeguest
user/user	An account intended for the FME Flow Web User Interface .	Disabled	fmeuser

See Also

- [Changing the Login Password](#)
- [Role-Based and User-Based Access Control](#)

Configuring Authentication for the FME Flow Web Services

All workspaces that are run from FME Flow use at least one of the [FME Flow Web Services](#), and these services must authenticate with the [user account](#) or security token that requests them. The user account or token must have Run permission on the repository in which the workspace resides, and Access permission to the service.

 **Note** In versions of FME Flow prior to 2023.0, the guest user account could provide unauthenticated access to the FME Flow Web Services. For security reasons, this feature is removed.

Authenticating with Tokens

A [token](#) can provide the credentials required for authentication. We recommend using tokens in conjunction with user accounts that have only the required permissions for their service requests to succeed, and no additional permissions.


Using Directory Server Accounts

If you have imported [Active Directory](#) accounts into FME Flow as users, you can provide those accounts with access to repositories and services. Configure directory server user permissions on the [Users](#) page. Alternatively, if your directory server accounts belong to roles, you can configure role access on the [Roles](#) page.


See Also

- [Authentication and Authorization](#)

Configuring for HTTPS

 **Note** The task described here should be undertaken by advanced users only. Before proceeding, consider your options for alternative solutions until you are certain you wish to proceed. For additional resources, consult the [FME Community](#) or [FME Support](#).

- **Skill Level:** Advanced
- **Estimated Time Required:** 45-60 minutes
- **Prerequisites:**
 - Test jobs and services to ensure FME Flow is fully functional. For more information, see [Verify the Installation](#)
 - Familiarity with the Certificate Authority (CA) instructions from your certificate provider, particularly for generating the Certificate Signing Request (CSR).
 - (Recommended) Familiarity with your web application server's SSL configuration and certificates. (Apache Tomcat is the servlet for an [Express](#) or [Fault-Tolerant](#) installation of FME Flow, and as an option with certain [Distributed](#) installations.)
 - (Recommended) Access to the person who generates your certificates.

 **Note** If using Microsoft IIS Application Request Routing (ARR), refer to [this FME Community article](#).

Configuring for HTTPS

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

The following are two supported methods for securing FME Flow with HTTPS. For alternative methods, such as using a self-signed certificate, see [Configuring FME Flow for HTTPS](#) in the FME Community.

Note The following instructions provide steps for setting up SSL for Apache Tomcat, which is the application server included with an [Express](#) or [Fault-Tolerant](#) installation of FME Flow, and as an option with certain [Distributed](#) installations. Instructions to set up SSL on different web application servers vary.

For more information about configuring Apache Tomcat for HTTPS, or if you are using a different version of Apache Tomcat, see the documentation for your version on <https://tomcat.apache.org/>.

Windows

Using a CA-issued Certificate

This method requires you to generate a certificate signing request from FME Flow, which your IT team can use to create a CA certificate with the .cer or .crt extensions. If the certificate uses the .pfx extension, follow [Using a PFX or P12 Certificate](#) (below) instead.

1. Create a Keystore Generation Script

Open a text editor and copy the example script below, replacing the argument values with your own.

Note The `storepass` and `keypass` arguments must be the same and at least 6 characters.

```
keytool -genkey -noprompt -keyalg RSA -keystore tomcat.keystore -  
alias <alias> -dname "<dname>" -storepass <storepass> -keypass  
<keypass> -ext san="<san>" -deststoretype pkcs12
```

Keytool Arguments

Argument	Description
genkey	The keytool program command to generate a new keystore.

Argument	Description
noprompt	Using this argument in the command removes any interaction with the user.
keyalg	The algorithm to generate a private/public key pair.
keystore	The keystore file name.
deststoretype	Keystore type, <code>pkcs12</code> or <code>jks</code> .
dname	The CN name, Organization Unit, Organization, Location (city), State, and two-letter country code. The distinguished name is a set of values used to create the certificate and should be entered as you would like them to be presented to FME Flow users and visitors.
storepass, keypass	The password of the key and keystore. The value must be a minimum of six characters and must be the same for both arguments.
ext san	The subject alternative name is a structured way to indicate all of the domain names and IP addresses that are secured by the certificate.
alias	The name of the key inside the keystore being created.

Example:

```
keytool -genkey -noprompt -keyalg RSA -keystore tomcat.keystore -
alias tomcat -dname "CN=fmeflow.example.org, OU=support,
O=SafeSoftware, L=Surrey, S=BC, C=CA" -storepass password1 -
```

```
keypass password1 -ext san="dns:fmefflow.example.org,dns:fmefflow" -
deststoretype pkcs12
```

2. Run the Keystore Generation Script

- a. Open a command prompt as administrator and navigate to the FME Flow installation Java bin directory:

```
cd <FMEFlowDir>\Utilities\jre\bin\
```

Where *<FMEFlowDir>* is the location of the FME Flow installation folder.

- b. Execute the command created in step 1.

3. Generate a Certificate Signing Request (CSR)

In the command prompt, remain in *<FMEFlowDir>\Utilities\jre\bin* and run:

```
keytool -certreq -keyalg RSA -alias <alias> -file <filename> -
keystore tomcat.keystore -ext san="<san>"
```

Specify the certificate signing request path to the *<filename>*, and update *<alias>* and *<san>* to match that set in step 1.

Example:

```
keytool -certreq -keyalg RSA -alias tomcat -file certreq.csr -
keystore tomcat.keystore -ext
san="dns:fmefflow.example.org,dns:fmefflow"
```

4. Obtain a Certificate

Submit the CSR (for example, *certreq.csr*) generated in step 3 to your CA to obtain a certificate, according to your CA's instructions.

5. Import the Certificate into the Keystore

If you have multiple certificates, install them in the following order, and be sure to update the alias and certificate path for each.

- a. Import the root certificate (if you have one):

```
keytool -import -alias root -keystore tomcat.keystore -
trustcacerts -file <path/certificate_filename>
```

b. Import the intermediate certificate (If you have one):

```
keytool -import -alias intermediate -keystore tomcat.keystore  
-trustcacerts -file <path/certificate_filename>
```

c. Import the certificate:


```
keytool -import -alias <alias> -keystore tomcat.keystore -  
trustcacerts -file <path/certificate_filename>
```

 **Note** Use the same *<alias>* specified in step 1.

6. Import the Keystore into FME Flow's trusted certs

In a command prompt, from *<FMEFlowDir>\Utilities\jre\bin*, use the following command to import the keystore into FME Flow's trusted certs, specifying the *srcstorepass* argument with the password from step 1.

```
keytool -importkeystore -noprompt -srckeystore tomcat.keystore -  
destkeystore "<FMEFlowDir>\Utilities\jre\lib\security\cacerts" -  
deststorepass changeit -srcstorepass <password>
```

 **Note** Ignore the warning that the destination type must default to jks.

7. Back up the Tomcat XML Configuration Files

Navigate to *<FMEFlowDir>\Utilities\tomcat\conf* and make backups of *server.xml*, *web.xml*, and *context.xml*. We recommend this step so that you can easily revert the configuration at any point if necessary.

8. Configure *server.xml*

a. Run a text editor as an administrator and open *server.xml*, located in *<FMEFlowDir>\Utilities\tomcat\conf*.

- b. Locate the `SSLEngine` setting in the `<Listener>` element, including `className="org.apache.catalina.core.AprLifecycleListener"` and change the "on" value to "off".
- c. Locate the `<Connector>` element that contains `protocol="org.apache.coyote.http11.Http11NioProtocol"` and replace the entire element with:

```
<Connector protocol="org.apache.coyote.http11.Http11NioProtocol"
port="443"
minSpareThreads="5"
enableLookups="true"
disableUploadTimeout="true"
acceptCount="100"
maxThreads="200"
maxHttpHeaderSize="16384"
scheme="https"
secure="true"
SSLEnabled="true"
maxParameterCount="1000"
keystoreFile="<file>"
keystorePass="<password>"
clientAuth="false" sslEnabledProtocols="TLSv1.1,TLSv1.2"
sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation"
ciphers="TLS_AES_256_GCM_SHA384,TLS_CHACHA20_POLY1305_SHA256,TLS_AES_128_GCM_
SHA256,DHE-RSA-AES256-GCM-SHA384,DHE-RSA-AES128-GCM-SHA256,ECDHE-RSA-AES256-GCM-
SHA384,ECDHE-RSA-AES128-GCM-SHA256,DHE-RSA-AES256-SHA256,DHE-RSA-AES128-
SHA256,ECDHE-RSA-AES256-SHA384,ECDHE-RSA-AES128-SHA256,ECDHE-RSA-AES256-
SHA,ECDHE-RSA-AES128-SHA,DHE-RSA-AES256-SHA,DHE-RSA-AES128-SHA"
URIEncoding="UTF8" />

<Connector port="80" protocol="HTTP/1.1" redirectPort="443"/>
```

Make sure to update the `keystoreFile` and `keystorePass` parameters to the keystore location and password set in step 1. For an example, see [this server.xml reference](#). If the password contains invalid XML characters `<` `>` `"` `'` `&`, they must be escaped.

Note

- To disable TLS 1.1, remove `TLSv1.1`, from the `sslEnabledProtocols` setting.
- The list of `ciphers` is not exhaustive. If your certificate was generated with a different algorithm, the applicable cipher must be added. Any algorithms not in use can be safely removed from this list. For a full list of ciphers supported by Tomcat, see the [Apache Tomcat 9.0.69 API Documentation](#).

- (Optional) To change the port for HTTPS communication, change `443` to the desired port, for both the `port` and `redirectPort` directives.
- Save and close the `server.xml` file.

9. Configure `web.xml`

- Open `web.xml`, located in `<FMEFlowDir>\Utilities\tomcat\conf`.
- 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.

10. Configure `context.xml`

- a. Open context.xml, located in `<FMEFlowDir>\Utilities\tomcat\conf`.
- b. Add the following to the end of the file, just before the closing `</context>` element:

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

- c. Save and close the context.xml file.

11. Update the FME Flow Web URL to Use HTTPS

- a. Run a text editor as an administrator and open [fmeFlowConfig.txt](#).
- b. Update the `FME_SERVER_WEB_URL` directive by changing `http` to `https` and change the port to the same one specified in step 8.
- c. Save and close the file.

12. Verify the HTTPS Configuration

- a. [Restart](#) FME Flow.
- b. Open a web browser and navigate to `https://localhost/`. If you configured Tomcat to use a port other than the standard port 443, also specify the port (`https://localhost:<port>`).
- c. You should see the FME Flow login page in a secured format.

13. Modify Service URLs to Use HTTPS

To submit jobs on FME Flow via HTTPS, you must enable SSL for the [FME Flow Web Services](#).

- a. In the FME Flow Web User Interface, open the [Services](#) page.
- b. Click **Change All Hosts** and, in the *URL Pattern* field, change **HTTP** to **HTTPS**. (FME Flow may have already set this change.) If required, modify the port number—typically SSL is configured on either port 8443 or 443. When finished, click **OK**.

- c. [Run](#) a sample workspace with the data download and job submitter services to confirm your FME Flow is working with HTTPS.


Your FME Flow is now configured to work via HTTPS. However, if you are using the [WebSocket Server](#) or [Integrated Windows Authentication](#), some additional steps are required.

14. (Optional) Enable SSL on the WebSocket Server

The FME Flow [WebSocket Server](#) supports insecure (ws://) or secure connections (wss://). This configuration is only required if you want to use the WebSocket Server or [Topic Monitoring](#) (legacy).


- a. Run a text editor as an administrator and open the `fmeWebSocketConfig.txt` file in your FME Flow installation directory (`<FMEFlowDir>\Server`).
- b. Set `WEBSOCKET_ENABLE_SSL=true`.
- c. Uncomment the `WEBSOCKET_KEYSTORE_FILE_PATH` directive and set it to reference the keystore file set in `server.xml` in step 8. For example:

```
WEBSOCKET_KEYSTORE_FILE_  
PATH=<FMEFlowDir>/Utilities/tomcat/tomcat.keystore
```


 **Note** Use forward slashes, which may be different from the path in `server.xml`.

- d. Uncomment the `WEBSOCKET_KEYSTORE_FILE_PASSWORD` directive and set it to reference the keystore file password set in `server.xml` in step 8. For example:


```
WEBSOCKET_KEYSTORE_FILE_PASSWORD=password1
```

 **Note** Do not enclose the password in quotes.

- e. Specify the same settings for the WEBSOCKET_ENABLE_SSL, WEBSOCKET_KEYSTORE_FILE_PATH, and WEBSOCKET_KEYSTORE_FILE_PASSWORD directives in the following files:
 - `<FMEFlowDir>\Server\config\subscribers\websocket.properties`
 - `<FMEFlowDir>\Server\config\publishers\websocket.properties`
- f. In the following files, update the protocol in the `value` property of the `PROPERTY` directive from `"ws:"` to `"wss:"`
 - `<`
`FMESharedResourceDir`
`>\localization\publishers\websocket\publisherProperties.xml`
 - `<`
`FMESharedResourceDir`
`>\localization\subscribers\websocket\subscriberProperties.xml`

 **Note** `<FMESharedResourceDir>` refers to the location of the FME Flow System Share, specified during [installation](#).

- g. Run the following .bat files, located in `<FMEFlowDir>\Clients\utilities`:
 - `addPublishers.bat`
 - `addSubscribers.bat`
 - h. [Restart](#) FME Flow.
 - i. To test that the configuration is complete, [run jobs](#) and view [Topic Monitoring](#).
15. (Optional) Update the SSO Authentication URL to use HTTPS

 **Note** This step is applicable only if you want to use [Integrated Windows Authentication](#) (single sign-on) to access the FME Flow Web Interface.

- a. Run a text editor as an administrator and open the `fmeserver` `propertiesFile.properties`, located in `<FMEFlowDir>\Utilities\tomcat\webapps\fmeserver\WEB-INF\conf\`.
- b. Locate the `SINGLE_SIGN_ON_AUTH_URL` parameter, and update the host name and port portion of the URL to match the host name through which the FME Flow Web User Interface is accessed.

For example:

```
SINGLE_SIGN_ON_AUTH_
URL=https://<MyFMEFlowHost>:443/fmetoken/sso/generate
```

Using a PFX or P12 Certificate

Use these instructions to configure your FME Flow for HTTPS using a .pfx or .p12 certificate obtained from a Certificate Authority (CA). If you do not have a certificate, follow the instructions for [Using a CA-issued Certificate](#) (above) instead.

1. Make a backup of the Tomcat XML configuration files

Go to `<FMEFlowDir>\Utilities\tomcat\conf` and make backups of `server.xml`, `web.xml`, and `context.xml`.

2. Configure `server.xml`

- a. Run a text editor as an administrator and open `server.xml`, located in `<FMEFlowDir>\Utilities\tomcat\conf`.
- b. Locate the `SSLEngine` setting in the `<Listener>` element, including `className="org.apache.catalina.core.AprLifecycleListener"` and change the `"on"` value to `"off"`.
- c. Locate the `<Connector>` element that contains `protocol="org.apache.coyote.http11.Http11NioProtocol"` and replace the entire element with:

```
<Connector protocol="org.apache.coyote.http11.Http11NioProtocol"
```

```

port="443"
minSpareThreads="5"
enableLookups="true"
disableUploadTimeout="true"
acceptCount="100"
maxThreads="200"
maxHttpHeaderSize="16384"
scheme="https"
secure="true"
maxParameterCount="1000"
SSLEnabled="true"
keystoreFile="<file>"
keystorePass="<password>"
keystoreType="PKCS12"
clientAuth="false" sslEnabledProtocols="TLSv1.1,TLSv1.2"
sslImplementationName="org.apache.tomcat.util.net.jsse.JSSEImplementation"
ciphers="TLS_AES_256_GCM_SHA384,TLS_CHACHA20_POLY1305_SHA256,TLS_AES_128_GCM_
SHA256,DHE-RSA-AES256-GCM-SHA384,DHE-RSA-AES128-GCM-SHA256,ECDHE-RSA-AES256-GCM-
SHA384,ECDHE-RSA-AES128-GCM-SHA256,DHE-RSA-AES256-SHA256,DHE-RSA-AES128-
SHA256,ECDHE-RSA-AES256-SHA384,ECDHE-RSA-AES128-SHA256,ECDHE-RSA-AES256-
SHA,ECDHE-RSA-AES128-SHA,DHE-RSA-AES256-SHA,DHE-RSA-AES128-SHA"
URIEncoding="UTF8" />

<Connector port="80" protocol="HTTP/1.1" redirectPort="443"/>

```

The `keystoreFile` should point to your .pfx certificate. We recommend storing this certificate in FME Flow's Tomcat directory: `<FMEFlowDir>\Utilities\tomcat\`. Make sure the `keystorePass` is set to the password for your PFX certificate. If the password contains invalid XML characters `< > " ' &`, they must be escaped.

Note

- To disable TLS 1.1, remove `TLSv1.1`, from the `sslEnabledProtocols` setting.
- The list of `ciphers` is not exhaustive. If your certificate was generated with a different algorithm, the applicable cipher must be added. Any algorithms not in use can be safely removed from this list. For a full list of ciphers supported by Tomcat, see the [Apache Tomcat 9.0.69 API Documentation](#).

- d. (Optional) To change the port for HTTPS communication, change 443 to the desired port, for both the `port` and `redirectPort` directives.
- e. Save and close the `server.xml` file.

3. Configure `web.xml`

- a. Open `web.xml`, located in `<FMEFlowDir>\Utilities\tomcat\conf`.
- b. 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>
```

- c. Save and close the `web.xml` file.

4. Configure `context.xml`

- a. Open `context.xml`, located in `<FMEFlowDir>\Utilities\tomcat\conf`.
- b. Add the following to the end of the file, just before the closing `</context>` element:

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

- c. Save and close the `context.xml` file.

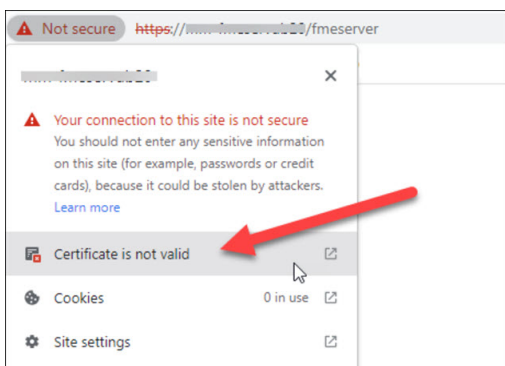
5. Update the FME Flow Web URL to Use HTTPS

- a. Run a text editor as an administrator and open [fmeFlowConfig.txt](#).
- b. Update the `FME_SERVER_WEB_URL` directive by changing `http` to `https` and change the port to the same one specified in step 2.

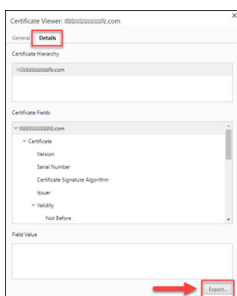
- c. Save and close the file.
- 6. (Optional) Export the certificate from the browser in base 64 format and import it into the cacerts trust store

Note This step is required only if your certificate was not obtained from a Trusted Certificate Authority. Instructions may differ slightly in a browser other than Chrome.

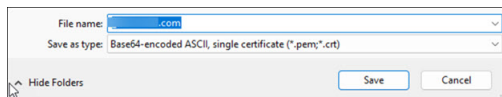
- a. Restart the FME Flow Application Server service
- b. Open a browser and navigate to `https://localhost/`. If you configured Tomcat to use a port other than the standard port 443, also specify the port (`https://localhost:<port>`).
- c. View the certificate from the browser.



- d. Select the **Details** tab and click **Export**.



e. Save as **Base-64-encoded ASCII, single certificate (CRT)** to local disk (for example, `<certpath>\mycert.crt`)



f. Import the keystore into FME Flow's trusted certs

In a command prompt, as an administrator, change directory to `<FMEFlowDir>\Utilities\jre\bin\`. Use the following command to import the keystore into FME Flow's trusted certs:

```
keytool -import -trustcacerts -keystore  
"<FMEFlowDir>\Utilities\jre\lib\security\cacerts" -storepass  
changeit -noprompt -file <certpath>\mycert.crt
```

7. Verify the HTTPS Configuration

- a. [Restart](#) FME Flow.
- b. Open a web browser and navigate to `https://localhost/`. If you configured Tomcat to use a port other than the standard port 443, also specify the port (`https://localhost:<port>`).
- c. You should see the FME Flow login page in a secured format.

8. Modify Service URLs to Use HTTPS

To submit jobs on FME Flow via HTTPS, you must enable SSL for the [FME Flow Web Services](#).

- a. In the FME Flow Web User Interface, open the [Services](#) page.
- b. Click **Change All Hosts** and, in the *URL Pattern* field, change **HTTP** to **HTTPS**. (FME Flow may have already set this change.) If required, modify the port number—typically SSL is configured on either port 8443 or 443. When finished, click **OK**.
- c. [Run](#) a sample workspace with the data download and job submitter services to confirm your FME Flow is working with HTTPS.


Your FME Flow is now configured to work via HTTPS. However, if you are using the [WebSocket Server](#) or [Integrated Windows Authentication](#), some additional steps are required.

9. (Optional) Enable SSL on the WebSocket Server

The FME Flow [WebSocket Server](#) supports insecure (ws://) or secure connections (wss://). This configuration is only required if you want to use the WebSocket Server or [Topic Monitoring](#) (legacy).


- a. Run a text editor as an administrator and open the `fmeWebSocketConfig.txt` file in your FME Flow installation directory (`<FMEFlowDir>\Server`).
- b. Set `WEBSOCKET_ENABLE_SSL=true`.
- c. Uncomment the `WEBSOCKET_KEYSTORE_FILE_PATH` directive and set it to reference the keystore file set in `server.xml` in step 2. For example:

```
WEBSOCKET_KEYSTORE_FILE_  
PATH=<FMEFlowDir>/Utilities/tomcat/mycert.pfx
```

 **Note** Use forward slashes, which may be different from the path in `server.xml`.

- d. Uncomment the `WEBSOCKET_KEYSTORE_FILE_PASSWORD` directive and set it to reference the keystore file password set in `server.xml` in step 2. For example:


```
WEBSOCKET_KEYSTORE_FILE_PASSWORD=password1
```

 **Note** Do not enclose the password in quotes.


- e. Specify the same settings for the `WEBSOCKET_ENABLE_SSL`, `WEBSOCKET_KEYSTORE_FILE_PATH`, and `WEBSOCKET_KEYSTORE_FILE_PASSWORD` directives in the following files:

- `<FMEFlowDir>\Server\config\subscribers\websocket.properties`
 - `<FMEFlowDir>\Server\config\publishers\websocket.properties`
- f. In the following files, update the protocol in the `value` property of the `PROPERTY` directive from `"ws:"` to `"wss:"`

- `<`
`FMESharedResourceDir`
`>\localization\publishers\websocket\publisherProperties.xml`
- `<`
`FMESharedResourceDir`
`>\localization\subscribers\websocket\subscriberProperties.xml`

 **Note** `<FMESharedResourceDir>` refers to the location of the FME Flow System Share, specified during [installation](#).

- g. Run the following `.bat` files, located in `<FMEFlowDir>\Clients\utilities`:
- `addPublishers.bat`
 - `addSubscribers.bat`
- h. [Restart](#) FME Flow.
- i. To test that the configuration is complete, [run jobs](#) and view [Topic Monitoring](#).
10. (Optional) Update the SSO Authentication URL to use HTTPS

 **Note** This step is applicable only if you want to use [Integrated Windows Authentication](#) (single sign-on) to access the FME Flow Web Interface.

- a. Run a text editor as an administrator and open the `fmeserver` `propertiesFile.properties`, located in `<FMEFlowDir>\Utilities\tomcat\webapps\fmeserver\WEB-INF\conf\`.

- b. Locate the `SINGLE_SIGN_ON_AUTH_URL` parameter, and update the host name and port portion of the URL to match the host name through which the FME Flow Web User Interface is accessed.

For example:

```
SINGLE_SIGN_ON_AUTH_  
URL=https://<MyFMEFlowHost>:443/fmetoken/sso/generate
```

Linux

FME Flow on Linux includes an NGINX reverse proxy that allows easy SSL configuration and the ability to choose ports under 1024 without root permission. HTTPS is configured on the NGINX reverse proxy rather than the Apache Tomcat web application server.

1. Create a directory for the certificate:

```
sudo mkdir /etc/nginx/ssl
```

2. Place a certificate and key in the new directory.

- If you are using a certificate issued by a certificate authority (CA), you will need the certificate or certificate bundle (.crt) and certificate key (.key).
- If you are using a .pfx or .p12 certificate, you must convert it into .crt and .key format. For more information, see [Configuring FME Flow for HTTPS](#) in the FME Community.
- Alternatively, you can generate a self-signed SSL certificate and keystore with the following command:

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -  
keyout /etc/nginx/ssl/<KeyName>.key -out  
/etc/nginx/ssl/<CertName>.crt
```

Replace `<KeyName>` and `<CertName>` with the key and certificate filenames, respectively. For example, `nginx.key` and `nginx.crt`.

3. Enable Diffie-Hellman key exchange:

```
sudo openssl dhparam -out /etc/nginx/ssl/dhparam.pem 2048
```

```
sudo chmod 400 /etc/nginx/ssl/dhparam.pem
```

4. Open file `/etc/nginx/conf.d/fmeflow.conf`.
5. Uncomment the `server` block that contains instructions to listen on port 80 and redirect to port 443. This block should appear similar to the following:

```
# Redirect from 80 to 443 when SSL is enabled
server {
    listen 80;
    server_name <<hostname>>;

    location / {
        return 301 https://$host$request_uri;
    }
}
```

6. In the main `server` block, comment the line with instructions to listen on port 80, and uncomment the lines that instruct to listen on port 443 with SSL and include the SSL configuration. When complete, these lines should appear as follows:

```
#listen 80;
listen 443 ssl;
include /etc/nginx/fmeflow/ssl.conf;
```

7. In the `websocket server` block, comment and uncomment the applicable lines to ensure that listening occurs on port 7078 with SSL. When complete, these lines should appear as follows:

```
#listen 7078;
listen 7078 ssl;
include /etc/nginx/fmeflow/ssl.conf;
```

8. Save and close the file.
9. Open file `/etc/nginx/fmeflow/ssl.conf` and make sure the SSL certificate and key are pointed to the correct name and directory of your certificate, as defined in steps 1 and 2.
10. Reload the NGINX configuration:

```
service nginx reload
```

11. Open file server.xml as administrator. This file is located in `<FMEFlowDir>\Utilities\tomcat\conf`.

12. Update the `proxyPort` directive to 443:

```
proxyPort="443"
```

13. Update the `scheme` directive to `https`:

```
scheme="https"
```

14. Save and close the file.

15. Open the following configuration file:

```
/opt/fmeflow/Utilities/tomcat/webapps/fmeserver/WEB-INF/conf/propertiesFile.properties
```

16. Update the `WEB_SOCKET_SERVER_PORT` directive to 443:

```
WEB_SOCKET_SERVER_PORT=443
```

17. Save and close the file.

18. Update the FME Flow Web URL to use HTTPS:

- a. Run a text editor as an administrator and open [fmeFlowConfig.txt](#).
- b. At the end of the file, under `FME SERVER SETTINGS START > Port and Host Assignments`, update the `FME_SERVER_WEB_URL` directive from `http` to `https`, and change the port to the same one specified in step 2.
- c. Save and close the file.


19. [Restart](#) FME Flow.

See Also

- [FME Flow Troubleshooting: HTTPS and SSL](#)
- [FME Support](#)


Configuring Integrated Windows Authentication

- **Skill Level:** Advanced
- **Estimated Time Required:** 30-45 minutes
- **Prerequisites:**
 - All required authentication credentials are available.
 - Access to system, network, and FME Flow administrators.

 **Note** The task described here should be undertaken by advanced users only. Before proceeding, consider your options for alternative solutions until you are certain you wish to proceed. For additional resources, consult the [FME Community](#) or [FME Support](#).

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

- There is no need to log in to the FME Flow Web User Interface. Instead, select **Use Windows Credentials** on the [Sign In](#) page.
- Similarly, there is no need to log in to FME Flow when using FME Workbench to publish a workspace or download an item. Instead, check *Use Alternate Login Method* and specify **Windows Credentials** in the Publish or Download wizard.

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

 **Note** Single sign-on is currently supported on Microsoft Edge, Firefox and Chrome.

To enable single sign-on:

1. [Update the Windows domain configuration](#) to allow FME Flow to authenticate using single sign-on.
2. [Update the web browser configuration](#) to use single sign-on.
3. If you have not already done so, enable single sign-on as part of SASL authentication of a [Windows Active Directory connection](#).
4. (External-facing URL for Apache Tomcat only) [Update the Tomcat Properties File](#) with the external-facing URL.

Updating the Windows Domain Configuration

To configure FME Flow to use single sign-on authentication, the Windows domain must recognize FME Flow as a domain service. The following steps are required:


- A. Represent FME Flow as a domain service by assigning it a service principal name (SPN).
- B. Register the SPN (or SPNs) to the [service account](#).
- C. Ensure that the service account requires Kerberos pre-authentication.
- D. Ensure your domain controllers do not allow RC4 encryption.

A) Assign a Service Principal Name

An SPN has the form: `<service>/<host>`, where:

`<service>` is the service type. In the context of FME Flow, this is `http`.

`<host>` is the name of the machine hosting FME Flow's web application server. To provide flexibility, we suggest assigning both the unqualified and fully-qualified versions of the host name.

 **Note** If FME Flow is configured for access through a DNS alias (CNAME), SPNs must also be registered using this alias.

To obtain the unqualified and fully-qualified versions of the host name:

1. From the FME Flow host machine, click the Start menu, right-click 'Computer' or 'My Computer' and select 'Properties'.
2. For the unqualified host name, refer to 'Computer name'.
3. For the fully-qualified host name, refer to 'Full computer name'.

For example, if the unqualified host name is 'MyETLServer' and the fully-qualified host name is 'MyETLServer.domain.net', the SPNs are:

- http/MyETLServer
- http/MyETLServer.domain.net

B) Register an SPN to a Service Account

1. From the [Domain Controller](#), open a command prompt (cmd.exe) via the Start menu.
2. Type `setspn -S <spn> <account>` to register the SPN to the service account.
3. Ensure that the command succeeded with the message 'Updated object'. If the message 'Unable to locate account ...' appears, the account name is incorrectly specified.
4. Repeat until all SPNs are added.

For example, using the SPNs in the previous example, and supposing the service account is 'fmeflowadmin', the following commands would be entered:

```
setspn -S http/MyETLServer fmeflowadmin
setspn -S http/MyETLServer.domain.net fmeflowadmin
```

C) Ensure the Service Account Requires Kerberos Pre-authentication:

1. From the [Domain Controller](#), open 'Active Directory Users and Computers' via the Start menu.
2. In the console tree, navigate to the service account.
3. Right-click the service account, and select Properties.
4. Select the Account tab.

5. Under Account Options, scroll to the bottom and ensure that 'Do not require Kerberos preauthentication' is unchecked.
6. Click Ok.

D) Ensure Your Domain Controllers do not Allow RC4 Encryption

If the domain controller local policy or group policy allows RC4 encryption, and this policy cannot be changed, you can allow FME Flow to use less secure encryption:

1. On the FME Flow machine navigate to `<FMEFlowDir>\Utilities\jre\conf\security`
2. Create a new file called `krb5.conf`
3. Edit the new file with the following text and save:

```
[libdefaults]
allow_weak_crypto = true
```

4. [Restart](#) FME Flow.

Updating the Web Browser Configuration

Depending on the browser used to access the FME Flow Web User Interface, additional configuration steps may be required for Integrated Windows Authentication (single sign-on).

Microsoft Edge

In most common Windows domain configurations, no additional configuration is necessary. However, if a user is unable to use single sign-on with FME Flow, check the following:

1. Open 'Internet Options' from the Start menu.
2. On the 'Security' tab, select 'Local intranet'.
3. If FME Flow is not a part of the local intranet, add it to the list of local intranet sites. To do so, click Sites -> Advanced, then add the site address (for example, `http://fmefflow.domain.net`).
4. On the 'Security' tab, select 'Local intranet' and click 'Custom level...'

5. Ensure that under User Authentication -> Login, 'Automatic logon only in Intranet zone' is selected.
6. On the 'Advanced' tab, ensure that Security -> Enable Integrated Windows Authentication is checked.
7. Restart the browser.

Chrome

Single sign-on is supported in Chrome versions 8 and later.

In most common Windows domain configurations, no additional configuration is necessary. However, if a user is unable to use single sign-on with FME Flow, perform the same steps outlined under Microsoft Edge.

Firefox

Single sign-on is supported on all versions of Firefox.

To enable single sign-on in Firefox:

1. In the address bar, type `about:config`. Accept the warning if one appears.
2. In the search bar, type `network.negotiate`.
3. Select the preference 'network.negotiate-auth.trusted-uris', and double-click it to modify its value.
4. Enter FME Flow's site address (for example, `http://fmeflow.domain.net`), and click **OK**. Do not specify a port or path. Multiple site addresses can be listed, separated by a comma (,).
5. If using a private browsing tab, search for `network.auth.private-browsing-ssso`.
6. Double-click on `network.auth.private-browsing-ssso` to change the *Value* to **true**.
7. Restart the browser.

Apple Safari and Opera

Single sign-on is currently not supported on these browsers.

Update the Tomcat Properties File

Complete this step if both of the following are true:

- Your FME Flow uses an Apache Tomcat servlet to run the FME Flow Web Services. Apache Tomcat is provided with an Express installation of FME Flow, and some [distributed installations](#). Or, your distributed installation may be using your own Apache Tomcat servlet.
- Your [FME Flow Web User Interface](#) is [accessible](#) through an external-facing URL, or you have otherwise changed the URL through which the Web User Interface is accessed, such as by [Configuring for HTTPS](#).

To update the Tomcat properties file

1. Open the Apache Tomcat properties file:

```
<FMEFlowDir>\Utilities\tomcat\webapps\fmserver\WEB-INF\conf\propertiesFile.properties
```

2. Locate the `SINGLE_SIGN_ON_AUTH_URL` parameter, and update the host name and port portion of the URL to match the host name through which the FME Flow Web User Interface is accessed. For example:

```
SINGLE_SIGN_ON_AUTH_URL=http://MyFMEFlowHost:80/fmetoken/sso/generate
```

3. Save and close the file.
4. [Restart](#) the FME Flow services.

Troubleshooting Authentication Services

To begin troubleshooting [Authentication Services](#), look in the `fmeServer_*` [log files](#) for messages related to an authentication service, such as (`Directory Server`), or (`Single Sign-On`).

Failure to Connect to Directory Server (SSL)

Symptom

When connecting to [Active Directory](#) with *Encryption Method* = [SSL](#), the following message appears in the Directory Servers page:

```
An error occurred while communicating with directory server (81) or (91)
```

Alternatively, the following message appears in the log file:

```
(Directory Server) Exception: "An error occurred while attempting to send the LDAP message to "...": javax.net.ssl.SSLHandshakeException: sun.security.validator.ValidatorException: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target caused by sun.security.validator.ValidatorException: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target caused by sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target"
```

```
(Directory Server) Successfully connected using directory configuration "<xxx>" and encountered 1 errors.
```

Cause

SSL certificate verification was enabled, but failed because the Certificate Authority (CA) was not trusted.

Resolution

If available, [import the Certificate Authority \(CA\) certificate](#).

Failure to Connect to Azure AD Through IIS Proxy

Symptom

When logging in to FME Flow with credentials from an Azure AD server that you have configured to route through a Microsoft Internet Information Services (IIS) [reverse proxy](#),


authentication fails.

Cause

The login HTTP request to the Azure AD server sends a redirect URI of the FME Flow Core hostname rather than the URI of the reverse proxy.

Resolution

1. Append the URL of your reverse proxy to the `fmeserver.sso.custom.baseurl=` line in the `SSO application.properties` file. If your FME Flow uses an Apache Tomcat web application server provided with the installation, this file is located in `<FMEFlowDir>\Utilities\tomcat\webapps\fmesso\WEB-INF\classes\`.
2. [Restart](#) FME Flow.
3. Ensure the *Redirect URI* setting on your Azure Active Directory portal is set to the URL of your reverse proxy (rather than your FME Flow hostname), appended with `/fmesso/azuread/redirect`.

 **Tip** In your IIS reverse proxy settings, clear *Reverse rewrite host in response headers*. In IIS Manager, this setting is located in the Application Request Routing Cache tool, under **Actions** > **Server Proxy Settings**.

Unable to Use SASL Authentication

Log file message:

```
(Directory Server) Not using SASL for authentication, because configuration is incomplete.
```

Cause

Authentication Method = **SASL** is specified for a [Active Directory](#) connection, but the [Realm](#) was not explicitly specified in the form of a fully-qualified domain name (FQDN).

Resolution

Use the *Realm* setting on the [Edit Server Connection](#) page to specify an FQDN.

Unsupported SASL Mechanism

Log file message:

(Directory Server) SASL mechanism "... " is not supported by Directory Server.

Cause

Authentication Method = **SASL** is specified for a [Active Directory](#) connection, but the specified SASL mechanism was not supported by the directory server.

Resolution

The *SASL* field of a [directory server connection](#) must be set to an SASL mechanism that is supported by the directory server.

To see the SASL mechanisms supported by a directory server:

From [AD Explorer](#):

1. Connect to the Active Directory.
2. Right-click the Active Directory root, and select 'Properties...'.
3. Select the 'RootDSE Attributes' tab.
4. The supported SASL mechanisms for this Active Directory server are listed under the 'supportedSASLMechanisms' attribute.

MSDN:

1. Determine the version of Microsoft Windows® Server.
2. The supported SASL mechanisms are listed under in this MSDN article:
https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-adts/a98c1f56-8246-4212-8c4e-d92da1a9563b.

Single Sign-On Authentication Failure (Incorrect Mechanism)

Log file message:

(Single Sign-On) Client attempted NTLM authentication; single sign-on authentication supports only Kerberos V5 authentication. Refer to single sign-on documentation for resolution.

(Single Sign-On) Failed authentication because of an invalid client token. Refer to single sign-on documentation for resolution.

Cause 1

The service principal name (SPN) is missing.

Resolution 1

See [Updating the Windows Domain Configuration](#) for information on how to create and register a service principal name (SPN).

Cause 2

The client is not using a supported web browser, or is using an incorrectly configured web browser.

Resolution 2

See [Updating the Web Browser Configuration](#) for information on supported browsers and on how to update web browser configurations.

Single Sign-On Authentication Failure (Negotiation Error)

Log file messages (Symptom 1):

(Single Sign-On) Negotiation reported an error: "Failure unspecified at GSS-API level (Mechanism level: Checksum failed)".

(Single Sign-On) Failed authentication because of an negotiation error. Refer to single sign-on documentation for resolution.

Log files messages (Symptom 2):

(Single Sign-On) Negotiation reported a defective token from client: "...".

(Single Sign-On) Failed authentication because of an negotiation error. Refer to single sign-on documentation for resolution.

Log files messages (Symptom 3):

```
(Single Sign-On) Negotiation reported an error: "...".
```

```
(Single Sign-On) Failed authentication because of an negotiation error.
```

```
Refer to single sign-on documentation for resolution.
```

Cause 1

The service principal name (SPN) wasn't registered to the [service account](#) used by FME Server.

Resolution 1

Ensure that the service account used for SPN registration matches the one specified in the *Service Account Name* field of the [Active Directory connection](#). For more information, see [Updating the Windows Domain Configuration](#).

Cause 2

[Single sign-on authentication](#) was attempted and failed, and the user does not exist in the configured Windows domain.

Resolution 2

Ensure that the user account used to log into the client machine is a part of the Windows domain that FME Flow is configured to use.

For example, if FME Flow is configured to use Active Directory for 'Domain1', clients logged in using a 'Domain2' user account will not be able to authenticate with FME Flow.

Single Sign-On Authentication Failure (Negotiation Error Encryption)

Log file message:

```
INFORM requesthandler 408053:(Single Sign-On) Negotiation reported an error: "Failure unspecified at GSS-API level (Mechanism level: Encryption type RC4 with HMAC is not supported/enabled)".
```

```
WARN requesthandler 408058 : (Single Sign-On) Failed authentication because
```

```
of an negotiation error. Refer to single sign-on documentation for resolution.
```

Cause

The domain controller allows RC4 encryption while FME Flow does not.

Resolution

Allow FME Flow to use less secure encryption:

1. On the FME Flow machine navigate to `<FMEFlowDir>\Utilities\jre\conf\security`
2. Create a new file called `krb5.conf`
3. Edit the new file with the following text and save:

```
[libdefaults]
allow_weak_crypto = true
```

4. [Restart](#) FME Flow.

Single Sign-On Authentication Failure (Service Account)

Log file message:

```
(Single Sign-On) Failed authentication using pre-authenticated
credentials (for a service account).
```

```
(Single Sign-On) Failed to create server credentials. Ensure that pre-
authenticated credentials (for a service account) are correctly
specified in single sign-on configuration.
```

Cause 1

The [service account](#) used by FME Flow was not correctly specified. The account name or password may be incorrect, or both.

Resolution 1

Ensure that the service account details are correctly specified in the *Service Account Name* and *Service Account Password* fields of an [Active Directory connection](#).

Cause 2

Authentication Method **SASL** is not specified, or the *SASL Mechanism* is not **GSSAPI**.

Resolution 2

SASL must be specified as the *Authentication Method* and **GSSAPI** must be specified as the *SASL Mechanism* in the [Active Directory connection](#).

See Also

- [SASL Authentication](#)

Single Sign-On Authentication Failure (Cross-Domain User)

Log file messages:

```
(Single Sign-On) Negotiation complete; authentication granted for user "...".
```

```
(Single Sign-On) Failed authentication because user "." could not be found in Active Directory.
```

Cause 1

[Single sign-on authentication](#) was attempted and succeeded, but the user does not exist in the configured Windows domain.

Resolution 1

Ensure that the user account used to log into the client machine is a part of the Windows domain that FME Flow is configured to use.

For example, if FME Flow is configured to use Active Directory for 'Domain1', clients logged in using a 'Domain2' user account will not be able to authenticate with FME Flow.

Cause 2

[Single sign-on authentication](#) was attempted and succeeded, but FME Flow did not have the right privileges to find the user. This may be caused by the [service account](#) setting 'Do not require Kerberos preauthentication'.

Resolution 2

Kerberos pre-authentication must be enabled for the service account. See [Updating the Windows Domain Configuration](#) for information on how to configure the service account.

Single Sign-On Authentication Failure (Same Machine Access)

Log file messages:

```
(Single Sign-On) Negotiation reported a defective token from client:  
"Defective token detected (Mechanism level: GSSHeader did not find the  
right tag)".
```

```
(Single Sign-On) Failed authentication because of an negotiation error.  
Refer to single sign-on documentation for resolution.
```

Cause

[Single sign-on authentication](#) was attempted on the same machine as FME Flow.

Resolution

Use a different machine to access FME Flow.

SASL Authentication Failure 1

Symptom

A [Active Directory](#) connection species **SASL** as *Authentication Method* and **GSSAPI** as *SASL Mechanism*, and the following error appears in the log file:

```
(Directory Server) Exception: "LDAPException(resultCode=82 (local  
error), errorMessage='An error occurred while attempting to initialize  
the JAAS login context for GSSAPI authentication:  
javax.security.auth.login.LoginException: Pre-authentication information  
was invalid (24) caused by KrbException: Pre-authentication information  
was invalid (24) caused by KrbException: Identifier doesn't match  
expected value (906)')"
```

Or

```
(Directory Server) Exception: "LDAPException(resultCode=82 (local error), errorMessage='An error occurred while attempting to initialize the JAAS login context for GSSAPI authentication: javax.security.auth.login.LoginException: Client not found in Kerberos database (6) caused by KrbException: Client not found in Kerberos database (6) caused by KrbException: Identifier doesn't match expected value (906)')"
```

Cause

The user's login password was incorrectly specified.

Resolution

Specify the correct password.

SASL Authentication Failure 2

Symptom

When using [SASL authentication](#) (Kerberos V5), the following error appears in the log file:

```
(Directory Server) Authenticating user "... " using SASL mechanism "GSSAPI" with KDC address "... " and realm "...".  
(Directory Server) Successfully established a new connection to "...".
```

Above message may appear multiple times.

```
(Directory Server) Failed to authenticate user.
```

Cause

FME Flow did not have the right privileges to find the user. This may be caused by the [service account](#) setting 'Do not require Kerberos preauthentication'.

Resolution

Kerberos pre-authentication must be enabled for the service account. See [Updating the Windows Domain Configuration](#) for information on how to configure the service account.

SASL Authentication Failure 3

Symptom

When using [SASL authentication](#) (Kerberos V5), the following error appears in the log file:

```
(Directory Server) Exception: "LDAPException(resultCode=82 (local error), errorMessage='An error occurred while attempting to initialize the JAAS login context for GSSAPI authentication: javax.security.auth.login.LoginException: Clock skew too great (37) caused by KrbException: Clock skew too great (37) caused by KrbException: Identifier doesn't match expected value (906)')"
```

Cause

The authentication failure is caused by a significant discrepancy in the system time between the FME Flow host machine and the directory server.

Resolution

Update the system time on one or both systems.

SASL Authentication Failure 4

Symptom

A [Active Directory](#) connection specifies [SASL](#) as *Authentication Method* and **GSSAPI** as *SASL Mechanism*, and the following error appears in the log file:

```
ERROR main 408010 : (Directory Server) Exception: "LDAPException (resultCode=82 (local error), errorMessage='An error occurred while attempting to initialize the JAAS login context for GSSAPI authentication: javax.security.auth.login.LoginException: Message stream modified (41) caused by KrbException: Message stream modified (41)')"  
ERROR main 408003 : (Directory Server) Failed to connect to an available server, or no servers were available.  
FATAL main 405405 : Security Manager init FAILED
```

Cause

The *Realm* setting does not specify the [Authentication Realm](#) as a fully-qualified domain name (FQDN).

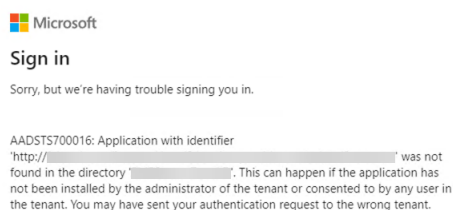
Resolution

Use the *Realm* setting on the [Edit Server Connection](#) page to specify an FQDN.

SAML Authentication Failure

Symptom

When logging in to FME Flow with credentials from a SAML identity provider, authentication fails with a message similar to the following:



This issue may occur on [Distributed](#) installations of FME Flow that use an Apache Tomcat web application server, and connect to a SAML identity provider through a [reverse proxy](#).

Cause

The SAML login HTTP request did not send the URL of the reverse proxy to the SAML identity provider.

Resolution

1. Append the fully-qualified hostname of your reverse proxy to the `fmeserver.saml.custom.baseurl=` line in the SAML application.properties file. If your FME Flow uses an Apache Tomcat web application server provided with the installation, this file is located in `<FMEFlowDir>\Utilities\tomcat\webapps\fmesaml\WEB-INF\classes\`.
2. [Restart](#) FME Flow.

Importing a CA Certificate for SSL Connections

To authenticate SSL connections using a CA certificate, import the certificate and instruct FME Flow to add it to the list of trusted certificates.

1. Make a backup of the current list of trusted Certificate Authorities:

```
<FMEFlowDir>/Utilities/jre/lib/security/cacerts
```

2. Obtain your RootCA certificate as a .cer file:

```
http://<domain-controller>/certsrv/certcarc.asp > 'install this CA certificate'
```

Or, contact your domain administrator

3. Add the certificate to the list of trusted ones:

- Linux:

Open the command terminal and issue the following commands:

```
cd <FMEFlowDir>/Utilities/jre
./bin/keytool -import -trustcacerts -alias <rootca-alias> -file
<cer-file> -keystore ./lib/security/cacerts
```

- Windows:

Launch cmd.exe and issue the following commands:

```
cd <FMEFlowDir>\Utilities\jre
bin\keytool -import -trustcacerts -alias <rootca-alias> -file
<cer-file> -keystore
lib\security\cacerts
```

When prompted for keystore password, default is `changeit`.


When prompted to trust certificate, answer `yes`.

Where:


- `<rootca-alias>` is the custom alias for the certificate.
- `<cer-file>` is the path to the .cer file downloaded in step 2.

4. Restart FME Flow.

Running the FME Flow System Services Under Different Accounts (Windows)

 **Warning** These instructions are recommended for Windows installations only. To avoid unexpected results on Linux installations, we recommend that all FME Flow system services run under the same user account.

By default, the FME Flow Core, FME Flow Engines, and FME Flow Application Server Windows services run under the local system account, which does not have network permissions. You may need to run these services under different accounts that can read and write data to the FME Flow System Share and FME Flow installation directory, particularly in a distributed installation where these services are installed on separate machines. For more information about required permissions for the FME Flow System Services, see [Directory and Account Permissions](#).

 **Note** When making this change, we recommend updating all three services to use the same [Service Account](#) or user account.

To Run the FME Flow System Services Under Different Accounts

1. **(FME Flow Engines service only)** If your FME Engines are installed separately from the FME Flow Core ([Adding FME Engines on a Separate Machine](#)), you may need to update the user account that runs the FME Engines to provide read/write access to the FME Flow System Share and FME Flow installation directory. Ensure that account has the necessary permissions on the FME Flow Core and FME Flow Database machines. For more information, see [Running the FME Flow Engines Under a Different Account](#).


2. Open the Windows Services:

Select Start > Control Panel > Administrative Tools > Services.

Or

Run `services.msc`.

3. In the Services list, right-click FME Flow Application Server, FME Flow Core, or FME Flow Engines, and select Properties.

 **Note** The FME Flow Core service is dependent on the FME Flow Database service, which must remain under the Local System account.

4. In the Properties dialog, click the Log On tab.
5. Select This account, and specify a user account and password that provides permissions to the FME Flow System Share and FME Flow installation directory. If you are updating the FME Flow Engines account, make sure the account corresponds to the user you specified in Step 1.
6. If [Cross-Origin Resource Sharing](#) settings have been updated, and you are updating the FME Flow Engines account, grant Write permissions on file cors.properties to the user you specified in Step 1. This file is located in the \localization\clients folder of the FME Flow System Share. For more information, see [Directory and Account Permissions](#). (You must have direct access to this file; it is not available from the Resources page of the Web User Interface.)
7. Click OK.
8. [Restart the FME Flow System Services](#) you updated.

Encrypting the FME Flow Database Password

The password for the [FME Flow Database](#) is stored in the [fmeDatabaseConfig.txt](#) configuration file.

By default, the password is encrypted. The password may be unencrypted if it was manually updated after installation. To encrypt the FME Flow Database password, follow these steps:

1. From all machines on which the FME Flow Core is installed, open a command prompt and start a PowerShell session. For example, on Windows, the command is:

```
powershell.exe
```

2. Change the directory to `<FMEFlowDir>\Clients\utilities\`

3. Run `encryptConfigSetting.ps1` (Windows) or `encryptConfigSetting.sh` (Linux), and supply the following parameters:

- `DB_PASSWORD`
- The `DB_PASSWORD` value, enclosed in single quotes ('). Find this in `fmeDatabaseConfig.txt`, and ensure it corresponds to your database server type. If your FME Flow is an Express installation, your database server (`DB_TYPE`) is `postgresql`. If your FME Flow is a Custom installation, your database server (`DB_TYPE`) is either `postgresql`, `sqlserver`, or `oracle`.

For example:

```
.\encryptConfigSetting.ps1 DB_PASSWORD 'fmeFlow'
```

4. Copy the encrypted password that is generated from the command prompt output and paste it over the unencrypted `DB_PASSWORD` value in `fmeDatabaseConfig.txt`.

5. Save `fmeDatabaseConfig.txt`, and [restart](#) FME Flow.

⚠ Warning Any change to [System Encryption](#), including generating a new custom encryption key, or reverting to default encryption, also affects the FME Flow Database password. To ensure this password is based on the current key, you must perform these steps again following any changes performed in System Encryption.

Other Tasks

Licensing and Administration Tasks	387
Backup, Restore, and Migration Tasks	411
Notifications Tasks	413
Workflow and Resources Tasks	414



Licensing and Administration Tasks

Licensing

Select **System Configuration > Licensing**.

Note

License files are installed, by default, to:

» Windows: `C:\ProgramData\Safe Software\FME Flow\licenses`

» Linux: `/opt/fmeflow/licenses`

Dynamic Engine (CPU Usage) licensing is not available for engines on FME Flow Hosted.

Automatic Mode (Recommended)


If your FME Flow host machine has an internet connection, you can request and install a license online without any need for downloading and re-uploading a license file.

Note


- Licensing FME Flow over the internet requires no inbound connection. For more information, see [FME Flow Automatic Licensing URLs](#).
- If you purchased credits for Dynamic Engines (CPU Usage), automatic mode only is supported. Manual mode licensing (below) is not supported.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.

3. Complete the fields on the Request a New License page.


 **Note** If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.

4. *Licensing Mode*: Select **Automatic**.
5. Click **OK**.
6. A message indicates that your license is installed, and your available engines appear momentarily on the [Engines](#) page.

 **Note** If the *Serial Number* you provided includes Dynamic Engines (CPU Usage), they are not configured to start by default. You can start and configure your Dynamic Engines on the [Engines](#) page.


Manual Mode

If you prefer not to use an internet connection for licensing, or if your FME Flow host machine does not have an internet connection, use this process to install a license.

 **Note** If you purchased credits for Dynamic Engines (CPU Usage), manual mode licensing is not supported. Use automatic mode for licensing (above). For more information, contact your Safe Software Inc. sales representative.

1. Open the [FME Flow Web User Interface](#) as a user assigned to the `fmesuperuser` [role](#). If none of your FME Flow engines are licensed, you will see a prompt to activate FME Flow. Click on the prompt. Otherwise, select **System Configuration > Licensing**.
2. Click **Request New License**.

3. Complete the fields on the Request a New License page.

 **Note** If you are licensing multiple FME Flow installations as part of a [fault-tolerant architecture](#), specify the same *Serial Number* for each installation.

4. *Licensing Mode*: Select **Manual**.

5. Click **OK**. A license request .json file downloads.

6. Email the .json file to codes@safe.com.

7. You will receive an email from Safe Software Inc. with a license file, which has a `.fmeLIC` extension. Download the file.

8. Return to the Licensing page. Under Standard Engines, click **Upload License File** and select the `.fmeLIC` file to upload. Or, drag and drop the `.fmeLIC` file onto the page under Standard Engines. A message indicates that it is installed, and your available engines appear on the [Engines](#) page.

Problems?

- [FME Flow Troubleshooting: FME Flow Engines](#)
- Contact codes@safe.com.

See Also

- [FME Flow Automatic Licensing URLs](#)
- [FME Engines](#)

Configuring the FME Engine

Configuring Implicit NAD27 to NAD83 Datum Shifts

If you plan to run workspaces that transform data between the datums NAD27 and NAD83 without explicitly choosing a transformation in a CsmarReprojector transformer, you must decide whether to use datum shift files for Canada or the US.

To apply the setting to all FME processes run on the machine

Run one of the following scripts:


Windows

- Canada: `<FMEFlowDir>Server/fme/utilities/setNAD2783datumCanada.bat`
- US: `<FMEFlowDir>Server/fme/utilities/setNAD2783datumUSA.bat`


Linux

- Canada: `<FMEFlowDir>Server/fme/utilities/setNAD2783datumCanada.sh`
- US: `<FMEFlowDir>Server/fme/utilities/setNAD2783datumUSA.sh`

To apply the setting only to FME processes run by the current user

 **Note** A current user can also be a [Service Account](#).

Run one of the following from a command prompt:

 **Note** You must run the following commands as an Administrator.

- Canada: `<FMEFlowDir>Server/fme/fme APPLY_SETTINGS "CoordSys/NAD2783 Datum Shift" "canada only"`
- US: `<FMEFlowDir>/Server/fme/fme APPLY_SETTINGS "CoordSys/NAD2783 Datum Shift" "usa only"`

To apply different settings for specific FME Engines

In your FME Flow architecture, you may want a set of FME Engines on one machine to run jobs that use the Canada datum shift files, and another set of FME Engines on another machine to run jobs that use the US datum shift files.

First, identify and configure separate operating system user accounts to run the distributed FME Engines. For more information, see [Running the FME Flow Engines Under a Different Account](#).

Next, for each user account configured to run a set of FME Engines, run the applicable command (above).

Finally, use [Job Queues](#) to ensure your workspaces are run by the intended engine.

Configuring PROJ Network Settings

To run workspaces from FME Flow that use transformers to perform PROJ library reprojections (such as PROJReprojector and PROJAttributeReprojector), you may need to configure applicable PROJ networking settings.

To configure PROJ networking settings, run the following commands as an Administrator from a command prompt:

To enable PROJ networking

PROJ networking is enabled by default. To disable:

```
<FMEFlowDir>Server/fme fme APPLY_SETTINGS  
Reproject/PROJ/Networking/Enabled "false"
```

To re-enable:

```
<FMEFlowDir>Server/fme fme APPLY_SETTINGS  
Reproject/PROJ/Networking/Enabled "true"
```

To set a CDN endpoint

By default, the PROJ content delivery network (CDN) accesses grid files from <https://cdn.proj.org>. To change this endpoint:

```
<FMEFlowDir>Server/fme fme APPLY_SETTINGS "Reproject/PROJ/Networking/CDN  
Endpoint" "<Endpoint>"
```

To set the maximum grid cache size

By default, the maximum size of the cache to allocate on the local system for grid files is unlimited, which corresponds to a value of -1. To limit the disk cache size, specify a value, in megabytes, with this command:

```
<FMEFlowDir>Server/fme fme APPLY_SETTINGS "Reproject/PROJ/Networking/Grid  
Cache Maximum Size" "<GridCacheMaxSize>"
```

Configuring Grid Transformation Definitions

To reproject between coordinate systems that have different datums in workspaces run from FME Flow, use a grid transformation override file that references custom grid files.

You must make the desired changes in FME Workbench, then manually copy the applicable grid override file to FME Flow.

1. In FME Workbench, click **Tools > FME Options > Coordinate Systems**, and edit the applicable grid transformation.
2. On the [Resources](#) page of the [FME Flow Web User Interface](#), open the **Engine** folder. If the sub-folder **GridData** does not exist, create it.
3. Upload the grid file(s) to **GridData**.
4. Locate the applicable grid override file from the FME Form machine:

Windows:

```
<user>\My Documents\FME\CoordinateSystemGridOverrides\<grid_  
override_file>.txt
```

Linux:

```
> ~/.fme/CoordinateSystemGridOverrides
```

5. Make a copy of the grid override file.
6. Open the copy in a text editor. You will see a relative path to the `.\GridData\<GRID_SHIFT_FILE>` file. Change the relative path to a fully-qualified path:


Windows:

C:/ProgramData/Safe

Software/FMEFlow/resources/engine/GridData/<GRID_SHIFT_FILE>.gsb

Linux:

/data/fmeflow/resources/engine/GridData/<GRID_SHIFT_FILE>.gsb

 **Note** If any part of the path contains spaces, enclose the entire path in quotation marks (""). If paths containing spaces are not enclosed in quotation marks, the FME Flow Engines service may fail to restart.

7. Save and close the file.
8. On the Resources page, upload the grid override text file to the **CoordinateSystemGridOverrides** folder, under **Engine**.
9. [Restart](#) FME Flow.

Using Python with FME Flow

If you run workspaces from FME Flow that reference Python, the FME Engines must know which Python interpreter to use. Workspaces built with FME Form 2016 or later tell FME which version of Python is required to run Python code through the [Python Compatibility](#) workspace parameter.

The Python interpreter that the FME Engines actually use is the one that most closely matches the Python Compatibility workspace parameter, which, by default, is FME's latest supported version of Python 3. To change the default Python interpreter used by the FME Engines, see [Changing the Python Interpreter \(Windows only\)](#) (below).

Using Additional Python Modules

If you need to reference a Python module from FME Flow that is not part of the Python standard library, follow these steps to: 1) Obtain the module (if you have not already); and 2) Upload the module.

Obtaining the Python Module

If you do not already have the Python module you want to reference, install the package with the following command:

- **Windows:** `fme.exe python -m pip install <package_name> --target <package_destination_folder>`
- **Linux:** `./fme python -m pip install <package_name> --system --target <package_destination_folder>`

<package_name> is the name of the Python package to install.

<package_destination_folder> is the folder in which to install the Python package . If not specified, the default folder is:

- **Windows:** `C:\Users\<user>\Documents\FME\Plugins\Python>`
- **Linux:** `~/fme/Plugins/Python`


For example, to install a package named PyExecJS to the default folder:


- **Windows:** `fme.exe python -m pip install PyExecJS`
- **Linux:** `./fme python -m pip install PyExecJS --system`

Uploading the Python Module


1. Locate the files or folders of the Python module you installed, such as .pyc or .py files, as well as any sub-directories that may contain init.pyc or main.pyc files of your package.
2. Use the Upload tool on the [Resources](#) page of the Web User Interface, and upload these file(s) and/or folder(s) to the folder under Engine\Plugins\python that corresponds to the Python release compatible with the module (for example, python36).

Changing the Python Interpreter (Windows only)

 **Important** Consider changing the Python interpreter *only* if the Python interpreter shipped with FME Flow is incompatible with third-party Python libraries required to run the Python scripts referenced in your workspaces. If you require an Esri ArcGIS Python interpreter to use ArcPy, set the [Python Compatibility](#) workspace parameter to the applicable **Esri ArcGIS Python 3.x** value in FME Workbench prior to uploading to FME Flow. If set properly in this case, changing the Python interpreter explicitly in FME Flow is not necessary.

 **Note** Beginning with FME Flow version 2019.0 on Linux, changing the Python interpreter is not supported.

Sometimes you will want to use a Python interpreter other than the ones shipped with FME Flow. In that case, you can change the interpreter through the following procedure:


 **Note** The Python interpreter specified by this procedure does not guarantee the FME Engines will use the interpreter for all workspaces. Instead, the FME Engines use the specified interpreter for all workspaces that have a *Python Compatibility* value that is compatible with the specified interpreter. If the *Python Compatibility* of a workspace is not compatible with the specified interpreter, the FME engines use one of their own Python interpreters that is compatible.

Open a command prompt, specifying **Run as administrator**, and run the following commands:

FME Flow 2018.0 and earlier

```
<FMEFlowDir>Server/fme/fme.exe APPLY_SETTINGS SYSTEM "Python/Use Custom Python" true
```

```
<FMEFlowDir>Server/fme/fme.exe APPLY_SETTINGS SYSTEM "Python/Python Interpreter" c:/<path_to_dll>.dll
```

 **Note** For information on how to locate Python dll's (<path_to_dll>), see [Choosing a different Python Interpreter](#) in the FME Community.

FME Flow 2018.1 and later

```
<FMEFlowDir>Server/fme/fme.exe APPLY_SETTINGS SYSTEM "Python/Use Custom Python 64" true
```

```
<FMEFlowDir>Server/fme/fme.exe APPLY_SETTINGS SYSTEM "Python/Python Interpreter 64" c:/<path_to_dll>.dll
```

```
<FMEFlowDir>Server/fme/fme.exe APPLY_SETTINGS SYSTEM "Python/Custom Python Home 64" c:/<path_to_python_home>
```

 **Note** <path_to_python_home> is the installation location of Python.

Using R with FME Flow

To execute R scripts from FME Flow, such as through the RCaller transformer in an FME Form 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 the FME Engines to an R interpreter that is installed in a non-standard location.

Install R

Download R installers from <https://www.r-project.org/>.

Windows

Run the installer as administrator. Right-click on the installer in the Start menu and select **Run as administrator**.

Linux

To install R on Linux, we recommend using a package manager.

Install the sqldf package for R

Choose one of the following options:

- Standard install (perform on all machines that run FME Engines)
- Upload to Resources

Standard Install

1. Open the R command prompt as administrator, and run the following command:

```
install.packages('sqldf')
```

2. A window launches prompting you to select a download mirror. After selecting a mirror, the sqldf package is installed to the system-wide R library.

3. To verify that sqldf is installed correctly, run the following command:

```
.libPaths()
```

The returned path should include a folder `sqldf`.


Upload to Resources

Use the Upload tool on the [Resources](#) page of the [FME Flow Web User Interface](#), and upload the package libraries under Engine\Plugins\R.

Using an R Interpreter Installed in a Non-Standard Location

You can direct the FME Engines to use an R interpreter that is installed in a non-default location. To set the path to the interpreter you want to use, run the following command from a command prompt, on all machines running FME Engines:

Windows

 **Note** To avoid errors, the Windows Command Prompt must likely be run as administrator, even if you are currently running your machine with administrator privileges. From the Start Menu, right-click on Command Prompt and select Run as administrator.

- `<FMEFlowDir>Server/fme/fme.exe APPLY_SETTINGS SYSTEM "Settings/Rscript Interpreter" <path_to_executable>`

Linux

- `<FMEFlowDir>Server/fme/fme APPLY_SETTINGS SYSTEM "Settings/Rscript Interpreter" /<path_to_executable>`

Working with the FME Flow System Services

About the FME Flow System Services

FME Flow is comprised of a series of components, and each component is installed as a standard Windows service.

FMEFlowAppServer

On install, if you use the default FME Flow Web Application Server, the service automatically starts at system boot time. If you are using your own web application server, it is important to note that the FME Flow Web Services are deployed and run by your web application server, not by FME Flow itself. Ensure that your web application server is started for users to access any FME Flow web services including the Web User Interfaces.

FME Flow Core

The [FME Flow Core](#) is the main FME Flow service and should always be running. By default, this service is started at system boot time.

FME Flow Database

On install, if you use the default [FME Flow Database](#), it creates a service called FME Flow Database, which automatically starts at system boot time. If you used your own database, such as Oracle, ensure that the service starts prior to the FME Flow Core service starting.

FME Flow Engines

This service starts and runs the [FME Engines](#).

Starting and Stopping FME Flow Manually

Windows

Although the FME Flow System Services start automatically, you can start and stop the services manually in the following ways:


- From the Services window (the procedure for accessing the services may vary by Windows version):

Control Panel > Administrative Tools > Services

Or

Start > Run > services.msc

In the Services window, select one or more of the [services](#), and start or stop as required.

 **Note** On Windows 8, you must be running as Administrator to start or stop the FME Flow System Services.

Linux

Linux scripts for starting and stopping FME Flow are located in:

<FMEFlowDir>/Server/

Although you can configure FME Flow to start automatically, you can start it manually by running *startServer.sh* as the root user.

This script starts all the [FME Flow System Services](#) in the necessary order. Alternatively, to start each service individually, run the following scripts, located in the same directory:

- *startCore.sh*: FME Flow Core service
- *startEngines.sh*: FME Engines
- *startDatabaseServer.sh*: FME Flow Database
- *startApplicationServer.sh*: FME Flow Application Server
- *startWebSocketServer.sh*: FME Flow WebSocket Server
- *startCleanup.sh*: FME Flow Cleanup service
- *startRootServices.sh* - Starts all the services that run as the root user. This script calls *startWebSocketServer.sh*, *startCleanup.sh*, and *startApplicationServer.sh*.

To stop FME Flow, run *stopServer.sh* as the root user.

Alternatively, stop the services individually with these scripts:

- `stopCore.sh`
- `stopEngines.sh`
- `stopDatabaseServer.sh`
- `stopApplicationServer.sh`
- `stopWebSocketServer.sh`
- `stopCleanup.sh`
- `stopRootServices.sh`

What Happens to Jobs After a Manual Restart?

When FME Flow is restarted, the behavior of jobs that are in a [Running](#) or [Queued](#) state depends on how the jobs were submitted:

	Run Workspace	Flow Apps	Schedules	Automations	Streams
Full Restart	Running jobs are resubmitted to the queue. Queued jobs run before resubmitted jobs.			Running jobs fail and are not resubmitted.* Queued jobs run once the FME Flow services are running.	Running jobs are resubmitted to the queue. The job ID may change depending on availability of FME Engines.
FME Engines Restart					
Core Restart					
Database Restart					
Web Application Server Restart	Jobs continue to run as the FME Engines process the job queue.				

* If *Retry on Failure* was checked on the workspace action of an automation, the job is resubmitted based on the Retry settings once the FME Flow services are running.

A resubmitted job has the same ID as its initial run, and assumes the job_<ID>.log filename. Logs for previous submissions of the same job can be accessed in the same folder. For more information, see [FME Job Logs](#) and [About Log Files in FME Flow](#).

Determining FME Flow Processes

To determine the FME Flow processes that are running:

- Windows: Launch the Windows Task Manager. Ensure that the Task Manager is set to show processes from all users.
- Linux: Examine the output of:

```
ps -A | grep -i fme 'fme\|postgres\|redis'
```

The following FME Flow processes might be listed:

Windows	Linux	Description
FMECleanup.exe	FMECleanup	FME Flow System Cleanup service
FMEConfiguration.exe	FMEConfiguration	FME Flow Backup & Restore
FMEConnection.exe	FMEConnection	FME Flow Database Connections and Web Connections access
FMEEngine.exe	fme	One for each running FME Engine
FMEMountPoint.exe	FMEMountPoint	FME Flow Resources
FMENotifier.exe	FMENotifier	FME Flow Subscriber Manager - requires the FME Flow Core to be running
FMEProcessMonitorCore.exe	FMEProcessMonitorCore	FME Flow Process Monitor for Core processes
FMEProcessMonitorEngines.exe	FMEProcessMonitorEngines	FME Flow Process Monitor for Engine processes

Windows	Linux	Description
FMEPublisher.exe	FMEPublisher	FME Flow Publisher (includes Automations Sources)
FMERelayer.exe	FMERelayer	FME Flow Publisher Manager
FMEScheduler.exe	FMEScheduler	FME Flow Schedules service - requires the FME Flow Core to be running
FMEFlow.exe	FMEFlow	FME Flow Core
FMEFlow_ApplicationServer.exe	java	The Apache Tomcat Web Application Server (servlet) that is installed with an Express installation of FME Flow, or as an option with a Distributed installation of FME Flow.
FMESMTPRelay.exe	FMESMTPRelay	FME Flow SMTP email receiver. For more information, see Configuring FME Flow to Receive Email .

Windows	Linux	Description
FMESubscriber.exe	FMESubscriber	FME Flow Subscriber (includes Automations Destinations) - requires the FME Flow Core to be running
FMEWebSocket.exe	FMEWebSocket	FME Flow WebSocket Server. For more information, see WebSocket Publisher and WebSocket Subscriber .
postgres.exe	postgres	The PostgreSQL server for the FME Flow Database that is installed with an Express installation of FME Flow.
memurai.exe	redis-server	FME Flow Queue, for managing Job Queues .

Configuring FME Flow as a System Service (Linux)

You can configure FME Flow as a system service that starts when you start your system. You should do this only if you have set up system services before and understand the different run levels on your system.

Warning Configuring scripts to run at start-up on Linux is an advanced task. The steps for doing so vary, depending on your configuration. Use the steps listed in this section as a guide only.

FME Flow is composed of a series of components, and each component has its own scripts. Both systemd and SysV scripts are provided.

Configuring with systemd Scripts

Getting Started

The following FME Flow systemd startup scripts are provided upon installation:

- `<FMEFlowDir>/Server/startup/systemd/fmeflow-appserver.service`
- `<FMEFlowDir>/Server/startup/systemd/fmeflow-cleanup.service`
- `<FMEFlowDir>/Server/startup/systemd/fmeflow-core.service`
- `<FMEFlowDir>/Server/startup/systemd/fmeflow-database.service`
- `<FMEFlowDir>/Server/startup/systemd/fmeflow-engines.service`
- `<FMEFlowDir>/Server/startup/systemd/fmeflow-websocket.service`

1. Copy these scripts and place them in the following directory, depending on your privilege level:

- root privilege: `/etc/systemd/system/`
- user-level privilege (no system administration rights): `/usr/lib/systemd/system/`

2. If the Web Application Server component of FME Flow is configured for communication on a port under 1024, this service runs as the root user. Open `fmeflow-appserver.service` and remove the following lines:

```
User=fmeflow
```

```
Group=fmeflow
```

3. Create a new folder `tomcat` in `/var/run` with `fmeflow:fmeflow` ownership.

4. Change ownership of tomcat logs in `/opt/fmeflow/Utilities/tomcat/logs/` to `fmeflow:fmeflow`.

To enable startup scripts to run at system boot time

Run the following command for each script:

```
systemctl enable *.service
```

For example:

```
systemctl enable fmefflow-core.service
```

To disable startup scripts from running at system boot time

Run the following command for each script:

```
systemctl disable *.service
```

For example:

```
systemctl disable fmefflow-core.service
```

To start services manually

Run the following command for each script:

```
systemctl start *.service
```

For example:

```
systemctl start fmefflow-core.service
```

To stop services manually

Run the following command for each script:

```
systemctl stop *.service
```

For example:

```
systemctl stop fmefflow-core.service
```

Configuring with SysV Scripts

Make a copy of the FME Flow startup scripts:

- `<FMEFlowDir>/Server/startup/SysV/FMEFlowAppServer`
- `<FMEFlowDir>/Server/startup/SysV/FMEFlowCleanup`

- `<FMEFlowDir>/Server/startup/SysV/FMEFlowCore`
- `<FMEFlowDir>/Server/startup/SysV/FMEFlowDatabase`
- `<FMEFlowDir>/Server/startup/SysV/FMEFlowEngines`
- `<FMEFlowDir>/Server/startup/SysV/FMEFlowWebSocket`

Place the scripts here:

- `/etc/init.d/FMEFlowAppServer`
- `/etc/init.d/FMEFlowCleanup`
- `/etc/init.d/FMEFlowCore`
- `/etc/init.d/FMEFlowDatabase`
- `/etc/init.d/FMEFlowEngines`
- `/etc/init.d/FMEFlowWebSocket`

From the `/etc/init.d/` directory, update permissions on these files with the following `chmod` command:

```
sudo chmod +x FME*
```

Register the FME Flow services with the following command:

```
sudo update-rc.d FMEFlow* defaults
```

Purging Logs and History

As FME Flow processes jobs, the history of these jobs is stored as records in the repository database and as log files in the file system. Eventually, the volume of these records and logs can build up and cause issues, such as slow web interface response time and low disk space.

To purge log files and job history, use the tools on the [System Cleanup](#) page of the Web User Interface.

Changing the Web Application Server Host Name and Port

Often you install FME Flow and then need to update the web application server host name or port.

The following instructions assume the Apache Tomcat server that is bundled by default with an FME Flow installation is the one currently in use.

Update Service URLs (For Updating Host Name or Port)


1. [Log in](#) to the FME Flow Web User Interface.
2. Navigate to [Services](#).
3. Click on each service to open the Editing Service page.
4. In the *URL Pattern* field, make the appropriate changes and click **OK** when done.

Update FME Flow Web URL (For Updating Host Name or Port)

On the machine hosting the FME Flow core:

1. Open the [fmeFlowConfig.txt](#) configuration file.
2. Locate the FME_SERVER_WEB_URL directive and update the URL value to the new web applications server host and/or port. For example:

```
FME_SERVER_WEB_URL=http://MY_WEBAPP_HOST:80
```

 **Note** If [Topic Monitoring](#) no longer functions after updating the host name, perform the following additional configuration:

1. Open the [propertiesFile.properties file](#) for the FME Flow service.
2. Add the line `WEB_SOCKET_SERVER_NAME=<hostname>`, where `<hostname>` is the new host name. Optionally, add the line `WEB_SOCKET_SERVER_PORT=<port>`, where `<port>` is the new port.
3. [Restart](#) FME Flow.

Update the Web Application Server XML File (For Updating Port)

To update the web application server xml file (server.xml), first determine if your FME Flow is [configured for HTTPS traffic](#). Then, follow the applicable instructions:

FME Flow not configured for HTTPS

1. Open the server.xml file, located in `<FMEFlowDir>\Utilities\tomcat\conf\`.
2. Locate the first instance of a line that begins: `<Connector port=`
3. Change the specified port as desired. For example, change `Connector port="80"` to `Connector port="8080"`.
4. Save the server.xml file.
5. [Restart](#) the Tomcat application server.

FME Flow configured for HTTPS

1. Open the server.xml file, located in `<FMEFlowDir>\Utilities\tomcat\conf\`.
2. Locate the first instance of a line that begins: `<Connector protocol=`
3. Change the specified port as desired. For example, change `port="443"` to `port="4443"`.
4. Locate the line that begins: `<Connector port=`
5. Change the `redirectPort` parameter to the same port specified in the previous step. For example, change `redirectPort="443"` to `redirectPort="4443"`.
6. Save the server.xml file.
7. [Restart](#) the Tomcat application server.

See Also

- [Moving the Web Applications Server to Another Machine](#)

Configuring International Character Encoding

By default, FME Flow tries to detect how your web service handles international character encodings. Beginning with FME Flow 2013, you can manually configure the FME Flow Web

Services and Web User Interface for international character encoding. To do this, you must configure each service and web application separately.

To configure international character encoding for a particular FME Flow Web Service or for the Web User Interface:

1. Open the web application's deployment descriptor web.xml file. This file is installed at `<webAppDir>/WEB-INF/web.xml`.
2. Find the `<filter>` element with `<filter-name>i18nFilter` and `<filter-class>COM.safe.web.servlet.CharsetEncodingFilter`.
3. Add/modify the `<init-param>` directives to configure international character encodings for this web application according to the table below:

Parameter Name	Type	Description
charset	string	Sets the character set to use. UTF-8 is recommended.
detect-encodings	boolean	“true” to automatically detect encodings. “false” to disable.
auto-encoded	list	A list of comma-separated keywords that specify what should be encoded by the specified character set (see charset). Available keywords: <ul style="list-style-type: none">• querystring – Encode query strings in the URI.• path – Encode the path component of the URI.• body – Encode the request body for each request.

For further instructions, see the J2EE Servlet Specification v2.5.

Moving the FME Flow System Share to Another Machine

The FME Flow System Share is the location for storing [Repositories](#) and [Resources](#). You may want to specify a new location for the System Share, and move all of its existing contents, to another networked machine.

To move the System Share, perform an [upgrade](#) operation (even if you are 'upgrading' to the same version of FME Flow). At the Install procedure (after Backup and Uninstall), select a Distributed/Fault Tolerant installation. During the install, on the FME Flow System Share dialog, specify the path to the new location. After reinstalling, perform a Restore.

Providing Support for OGC Services

Beginning with FME Flow 2016, Safe Software recommends providing Open Geospatial Consortium (OGC) services by configuring FME Form workspace templates that implement the OGC standard, and registering them to the FME Flow Data Streaming Service to provide output. This method provides fuller control over OGC features than OGC services that were native to FME Flow in previous releases.

To get started, workspace templates are available for both the OGC Web Feature Service (WFS) and OGC Web Mapping Service (WMS). You can then configure them to provide basic services with your data. For instructions and downloads, see the following:

- OGC WFS: [How to Create a WFS Service Using FME Flow](#)
- OGC WMS: [How to Create a WMS Service Using FME Flow](#)


Backup, Restore, and Migration Tasks

Migrating an FME Flow Configuration

To migrate the configuration of an FME Flow instance to another instance, perform a backup and restore operation using the tools on the [Backup & Restore](#) page of the FME Flow [Web User Interface](#).

Performing Scheduled Backups of an FME Flow Configuration

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

 **Note** For information about the contents of a backup, see [Backup & Restore](#).

To Configure a Scheduled Backup

1. [Log in](#) to the [FME Flow Web User Interface](#).
2. Select **Schedules**.
3. Click the Backup_Configuration entry.
4. On the Editing Schedule page, check the *Enabled* box.
5. Edit other parameters as necessary. See [Adding a Scheduled Task](#) for more information.

The following published parameters are specific to the Backup_Configuration schedule:

- *Append Date*: If **Yes**, the date and time of the backup is appended to the filename.
 - *Server Username*: The user performing the backup. The user must belong to the the fmesuperuser [role](#).
 - *Server Password*: Password of the user.
 - *Backup Destination*: The file to store the backup. You can specify a [Resources](#) location, or another location that is accessible by the engine running the workspace. It can refer to a UNC path.
6. Click **OK**.

Notifications Tasks

Configuring FME Flow to Receive Email

FME Flow can receive email via Simple Mail Transfer Protocol (SMTP) or Internet Message Access Protocol (IMAP). Using [Automations](#), you can publish emails to Notification Service topics using either the [Email - SMTP](#) or the [Email - IMAP](#) trigger.


SMTP or IMAP?

When deciding between SMTP or IMAP, keep in mind the following:


- IMAP is a pull-based mechanism. The Email (IMAP) Trigger polls an existing IMAP server (such as `imap.gmail.com`) for messages. As long as IMAP access is enabled on the email account, no additional configuration is needed. However, because polling is conducted at intervals, FME Flow may not receive email messages as immediately as those sent over SMTP.
- SMTP is a push-based mechanism. FME Flow acts as its own lightweight SMTP server, and receives emails directly, rather than polling for them. As a result, emails may be sent to FME Flow topics (and their subscribed workspaces) more immediately than emails delivered over IMAP.
- SMTP functions only on an instance of FME Flow with a proper DNS record (that is, the server is accessible over the internet). If your instance of FME Flow runs locally or on an intranet, you must use IMAP.
- SMTP is not supported in a [fault tolerant](#) installation of FME Flow.

Changing the Default SMTP Port (Optional)

By default, SMTP operates over port 25 on Windows and port 7125 on Linux. If you want to change the default port (non-FME Flow Hosted installations only), follow these steps:

 **Note** Be sure to configure your network firewall settings to allow transmission of emails to FME Flow on the applicable port.


1. Determine the port on the FME Flow Core host that will be used to receive emails.

 **Note** Be sure to configure your network firewall settings to allow transmission of emails to FME Flow on this port.

2. Open the SMTP publisher configuration file:

```
<FMEFlowDir>\Server\config\publishers\email.properties
```

3. Locate parameter **SMTP_PORT**, and update the setting to the desired port. By default, this parameter is set to port 25 on Windows installations, and 7125 on Linux.

 **Note** To specify a port under 1024 on Linux, configure either of the following:

- Elevated privilege on the FME Flow System Services (run as root). For more information, see [Starting and Stopping FME Flow Manually](#).
- Port forwarding: Configure the FME Flow Core to route and forward from a source port to a destination port. For example, route traffic on port 25 to port 7125.

4. [Restart](#) the FME Flow Core service.

Configuring FME Flow to Send Email Notifications

FME Flow can send emails when a variety of system events occurs, such as the success or failure of jobs that are run from FME Flow, or when a repository is created.

For more information, see [Monitoring FME Flow with Notifications](#).

Workflow and Resources Tasks

Sending Jobs to Specific Engines with Job Queues

Job queues are a mechanism for sending specific jobs to specific FME Engines. Job queues are configured in the FME Flow Web User Interface. For more information, see [Job Queues](#).

Job Recovery

After FME Flow submits a translation request to an [FME Engine](#), it monitors the connection to that engine until a response is returned.

If the connection to the engine is lost, or the engine crashes, FME Flow resubmits a job to the next available FME Engine in the [queue](#).

To prevent FME Flow from indefinitely resubmitting a job, the default setting is to resubmit a job up to three times. This setting is configurable.

Configuring Job Recovery

The `MAX_FAILED_TRANSACTION_REQUEST_RETRIES` parameter in the [fmeFlowConfig.txt](#) configuration file determines the number of times any transaction is resubmitted before FME Flow stops resending the request. A value of < 0 means that no maximum limit is imposed, and requests are resubmitted indefinitely. A value of 0 means that automatic resubmission is disabled. If this parameter is absent, a default value of 3 is used.

⚠ Warning A failed job request may cause an FME Engine to shut down improperly. When no maximum limit is imposed, the translation is resent indefinitely, which may cause repeated FME Engine failures.

Sharing FME Engine Resources

Workspaces that are run from FME Flow 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 Flow:

- Use the [Resources](#) page of the FME Flow Web User Interface: This method makes custom formats, custom transformers, and custom coordinate systems available to all

workspaces on FME Flow.

- [Publish to FME Flow](#): Makes a custom format or custom transformer available to all workspaces in the same repository to which it is published.
- [Publish to FME Flow with a Workspace](#): This method allows you to add resources when publishing a workspace. You can upload the resources to a repository or to a resources folder. If you upload to a resources folder, the resources are available to all other workspaces on FME Flow. If you choose a repository, the resources are available to all other workspaces in the same repository to which the current workspace is published.

To publish custom resources to FME Flow

1. Open the custom format or custom transformer in FME Workbench:

File > Open > Select custom format/transformer

2. Publish the resource to FME Flow as you would publish a workspace.

The resource then becomes available to workspaces within the repository to which you published it.

To publish custom resources to FME Flow when publishing a workspace

1. Open the workspace you want to publish in FME Workbench.
2. Start the Publish to FME Flow wizard:


File > Publish to FME Server

3. On the Publish Workspace dialog, click Select Files.
4. On the Upload Files dialog, click Add Files.
5. Select the resource file you want to include.
6. On the Upload Files dialog, click Change Location.
7. On the Change Upload Location dialog, specify an Upload Directory:

- **Upload to Repository:** All resources published with the workspace are accessible to any user with permission on the same repository. Other workspaces in the repository can access these resources, provided their respective source dataset

parameters are manually set to point to the resources.

- Upload to a shared resource folder: All resources published with the workspace will be available to all other workspaces run from FME Flow, from any repository. Resources must be uploaded to the applicable subdirectory under **Engine** in order for workspaces to use them. For example, a custom format must be uploaded to the **Engine > Formats** folder.

 **Note** If custom coordinate system files are uploaded to any of the applicable **Engine** subfolders, the FME Flow Engines service must be [restarted](#) before they can be used in a workspace. These include the ***CoordinateSystemExceptions***, ***CoordinateSystemGridOverrides***, ***CoordinateSystems***, and ***CsmapTransformationExceptions*** subfolders.

8. Complete the Publish to FME Flow wizard as usual.