



AVEVATM Edge Management

© 2015-2025 AVEVA Group Limited and its subsidiaries. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, photocopying, recording, or otherwise, without the prior written permission of AVEVA Group Limited. No liability is assumed with respect to the use of the information contained herein.

Although precaution has been taken in the preparation of this documentation, AVEVA assumes no responsibility for errors or omissions. The information in this documentation is subject to change without notice and does not represent a commitment on the part of AVEVA. The software described in this documentation is furnished under a license agreement. This software may be used or copied only in accordance with the terms of such license agreement. AVEVA, the AVEVA logo and logotype, OSIsoft, the OSIsoft logo and logotype, Archedra, Avantis, Citect, DYNsIM, eDNA, EYESIM, InBatch, InduSoft, InStep, IntelaTrac, InTouch, Managed PI, OASyS, OSIsoft Advanced Services, OSIsoft Cloud Services, OSIsoft Connected Services, OSIsoft EDS, PIPEPHASE, PI ACE, PI Advanced Computing Engine, PI AF SDK, PI API, PI Asset Framework, PI Audit Viewer, PI Builder, PI Cloud Connect, PI Connectors, PI Data Archive, PI DataLink, PI DataLink Server, PI Developers Club, PI Integrator for Business Analytics, PI Interfaces, PI JDBC Driver, PI Manual Logger, PI Notifications, PI ODBC Driver, PI OLEDB Enterprise, PI OLEDB Provider, PI OPC DA Server, PI OPC HDA Server, PI ProcessBook, PI SDK, PI Server, PI Square, PI System, PI System Access, PI Vision, PI Visualization Suite, PI Web API, PI WebParts, PI Web Services, PRISM, PRO/II, PROVISION, ROMEo, RLINK, RtReports, SIM4ME, SimCentral, SimSci, Skelta, SmartGlance, Spiral Software, WindowMaker, WindowViewer, and Wonderware are trademarks of AVEVA and/or its subsidiaries. All other brands may be trademarks of their respective owners.

U.S. GOVERNMENT RIGHTS

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the license agreement with AVEVA Group Limited or its subsidiaries and as provided in DFARS 227.7202, DFARS 252.227-7013, FAR 12-212, FAR 52.227-19, or their successors, as applicable.

AVEVA Legal Resources: <https://www.aveva.com/en/legal/>

AVEVA Third Party Software Notices and Licenses: <https://www.aveva.com/en/legal/third-party-software-license/>

Contents

What's New?	5
2024 Releases	5
November 2024	5
October 2024	6
September 2024	6
July 2024	6
June 2024	6
May 2024	7
April 2024	7
February 2024	7
2023 Releases	7
November 2023 - What's New?	8
October 2023 - What's New?	8
September 2023 - What's New?	8
August 2023 - What's New?	9
July 2023 - What's New?	9
April 2023 - What's New?	9
March 2023 - What's New?	10
January 2023 - What's New?	10
2022 Releases	10
December 2022 - What's New?	10
October 2022 - What's New?	10
August 2022 - What's New?	11
 Get Started	 12
Understand AVEVA Edge Management	12
Understand Operating System and Bandwidth Requirements	13
Understand Browser Support	15
Understand Supported Modules	16
Understand AVEVA Flex Licensing	16
Understand Firewall Exception Prerequisites	17
Allowlisting Domain Names for Outbound Communication	17
Understand Proxy Server Configuration Settings	18
Review Known Issues	19
 Setting Up AVEVA Edge Management	 20
Manage User Access	20
Create a User Account	20
Create a User Group	21
Understand User Roles	21
Assign a User Role	26
Configure a Federation Provider	27

Import a List of Users	27
Using AVEVA Edge Management	28
Working with the Device List Interface	28
Understand Your Device Status	28
Understand Your Device Connection Status	29
Understand Your AVEVA License Status	29
View Additional Keywords	29
Managing Devices	30
Add a Device Twin	30
Remove a Device Twin	31
Configure a Module	31
Linux x64 Modules	34
Special Considerations for Linux Modules on Windows Devices	34
Configure the AVEVA Edge IoT View Module	35
Configure the AVEVA Adapter for MQTT Module	35
Configure the AVEVA Adapter for OPC UA Module	37
Configure the Edge Data Store Module	38
Linux ARM64 Modules	39
Configure the AVEVA Adapter for MQTT Module	39
Configure the AVEVA Adapter for OPC UA Module	40
Configure the Edge Data Store Module	42
Pair a Device Twin with an Edge Device	43
Deploy a Device Twin	43
Change License Expiration Terms	44
View Device Logs	44
View Device Resource Usage	44
Uninstalling the IoT Edge Runtime	45
Example: Configure an EFLOW device	46
Working with the Template List Interface	49
Understand the Template List	49
View Template Details	49
Managing Templates	50
Create a Template	50
Update a Template	51
Delete a Template	52
Managing Settings	52
Configure License and Device Default Settings	53
Working Through Common Issues	54
Understand Common Issue Scenarios and Possible Solutions	54
Troubleshooting Errors with Device Deployment	56
Administration - Modules and Credits	56
Add a Module to the Product Catalog	57
Add a Module to an Account's Active Credit Agreement	57
Understand Helpful Commands	59

What's New?

This release documentation describes the new and enhanced functionality available in AVEVA Edge Management, providing an overview of the most significant changes. Any new features and enhancements are documented in this section.

Release Notes are available for the releases in the following years:

- [2024 Releases](#)
- [2023 Releases](#)
- [2022 Releases](#)

2024 Releases

Release notes are available for the releases in the following months:

- [November 2024](#)
- [October 2024](#)
- [September 2024](#)
- [July 2024](#)
- [June 2024](#)
- [May 2024](#)
- [April 2024](#)
- [February 2024](#)

November 2024

Fixes for critical and high priority security issues/vulnerabilities

1. Fixed high-priority security issues identified by Black Duck scan.
2. Fixed high-priority security issues identified by Polaris scan.
3. Fixed issues identified by Microsoft Defender Vulnerability Management scan.

Notable changes

1. Updated the Edge Management API Service to use Managed Identity for accessing blob storage.

October 2024

Fixes for critical and high priority security issues/vulnerabilities

1. Disabled anonymous blob storage access.

September 2024

Fixes for critical and high priority security issues/vulnerabilities

1. Fixed high-priority security issues identified by Black Duck scan.
2. Fixed high-priority security issues identified by Polaris scan.
3. Fixed issues identified by Microsoft Defender Vulnerability Management scan.

July 2024

Fixes for critical and high priority security issues/vulnerabilities

1. Fixed high-priority security issues identified by Black Duck scan.
2. Fixed high-priority security issues identified by Polaris scan.
3. Fixed issues identified by Microsoft Defender Vulnerability Management scan.
4. Resolved login failure issues.
5. Resolved intermittent HTTPS errors in the API image.

Notable changes

1. Support for IoT Edge runtime 1.5 LTS.
2. New devices use IoT Edge runtime 1.5 by default.
3. Migration to .NET 8 LTS from .NET 6 LTS.
4. Certificate renewal for all environments.

June 2024

Bug fixes

1. Fixed an issue preventing product modules from pulling from QA ACR.

May 2024

Fixes for critical and high priority security issues/vulnerabilities

1. Fixed high-priority security issues identified by Black Duck scan.
2. Fixed high-priority security issues identified by Polaris scan.
3. Fixed issues identified by Microsoft Defender Vulnerability Management scan.

Notable changes

1. Removed support for ARM32 devices.
2. AVEVA Insight - updated Auth0 grant type to Authorization Code.
3. Resolved backup issues with Blob Storage accounts and Key Vaults.
4. Resolved an issue preventing the garbage collector from removing orphaned iotHub devices.
5. Resolved an issue with starting services in EU region.
6. TLS 1.2 migration.

April 2024

Fixes for critical and high priority security issues/vulnerabilities

1. Fixed high-priority security issues identified by Black Duck scan.
2. Fixed high-priority security issues identified by Polaris scan.
3. Fixed issues identified by Microsoft Defender Vulnerability Management scan.

February 2024

Fixes for critical and high priority security issues/vulnerabilities

1. SQL Server authentication replaced with Azure EntraID.
2. Added support for Azure Keyvault with Azure EntraID.
3. Fixed issues identified by pen testing.
4. Fixed high-priority security issues identified by Black Duck and Polaris scans.
5. Replaced Azure Cosmos Table packages with the latest Azure DataTables packages.

2023 Releases

Release notes are available for the releases in the following months:

- [November 2023 - What's New?](#)

- [October 2023 - What's New?](#)
- [September 2023 - What's New?](#)
- [August 2023 - What's New?](#)
- [July 2023 - What's New?](#)
- [April 2023 - What's New?](#)
- [March 2023 - What's New?](#)
- [January 2023 - What's New?](#)

November 2023 - What's New?

Fixes for critical and high priority security issues/vulnerabilities

1. Fixed issues identified by Black Duck scan.
2. Fixed issues identified by Microsoft Defender Vulnerability Management scan.
3. Fixed an issue causing errors while uploading a file.

October 2023 - What's New?

Fixes for critical and high priority security issues/vulnerabilities

1. Fixed issues identified by Qualys vulnerability scan.
2. Fixed issues identified by Black Duck scan.
3. Fixed issues identified by Microsoft Defender Vulnerability Management scan.
4. Added non-root user to all core edge service images.

Notable features added

1. Product documentation is now available on the AVEVA Documentation Portal at docs.aveva.com.

September 2023 - What's New?

Fixes for critical and high priority security issues/vulnerabilities

1. Fixed issues identified by internal code scanning tools (Polaris, Black Duck).
2. Fixed a swapping issue with the log processor service.

August 2023 - What's New?

Notable features added

1. Secure secrets automatically propagate between version updates.
2. OIDC Data Hub credentials automatically propagate between version updates.
3. Improved error handling and messages when publishing devices.

Bug fixes

1. Fixed an issue that could prevent a device from being deleted.
2. Mutable metadata versions can now be removed.
3. Garbage collection is enabled through the provisioning pipeline.
4. Fixed a memory leak in the logger.

July 2023 - What's New?

Notable features added

1. Mutable metadata.
2. Garbage collection enabled.

Bug fixes

1. Device status no longer gets stuck in the published state after uploading a file with the same name as the previous upload.
2. Fixed an issue preventing a device from being updated due to an error deleting a secret from the keyvault.

April 2023 - What's New?

Notable features added

1. The correct module description displays when a module's description changes between versions
2. Added EFLOW file sharing support for devices running Windows 10 Professional 22H2
3. Added support for graceful shutdown of running modules when deploying updates to a device

Bug fixes

1. Fixed an access denied error for secure secrets with EFLOW

March 2023 - What's New?

Notable features added

1. Support for IoT Edge runtime 1.4 LTS.
2. New devices use IoT Edge runtime 1.4 by default.
3. Windows modules are no longer supported.

January 2023 - What's New?

Notable features added

1. Enhanced EFLOW bootstrap logging and error handling
2. Reduced number of reboots during EFLOW bootstrapping process
3. The following new modules are available for deployment:
 - a. AVEVA™ Adapter for MQTT module
 - b. AVEVA™ Adapter for OPC UA module
 - c. Edge Data Store module

2022 Releases

Release notes are available for the releases in the following months:

- [December 2022 - What's New?](#)
- [October 2022 - What's New?](#)
- [August 2022 - What's New?](#)

December 2022 - What's New?

Notable features added

1. EFLOW file sharing support
2. EFLOW runtime upgraded from 1.1 LTS to 1.4 LTS

October 2022 - What's New?

Notable features added

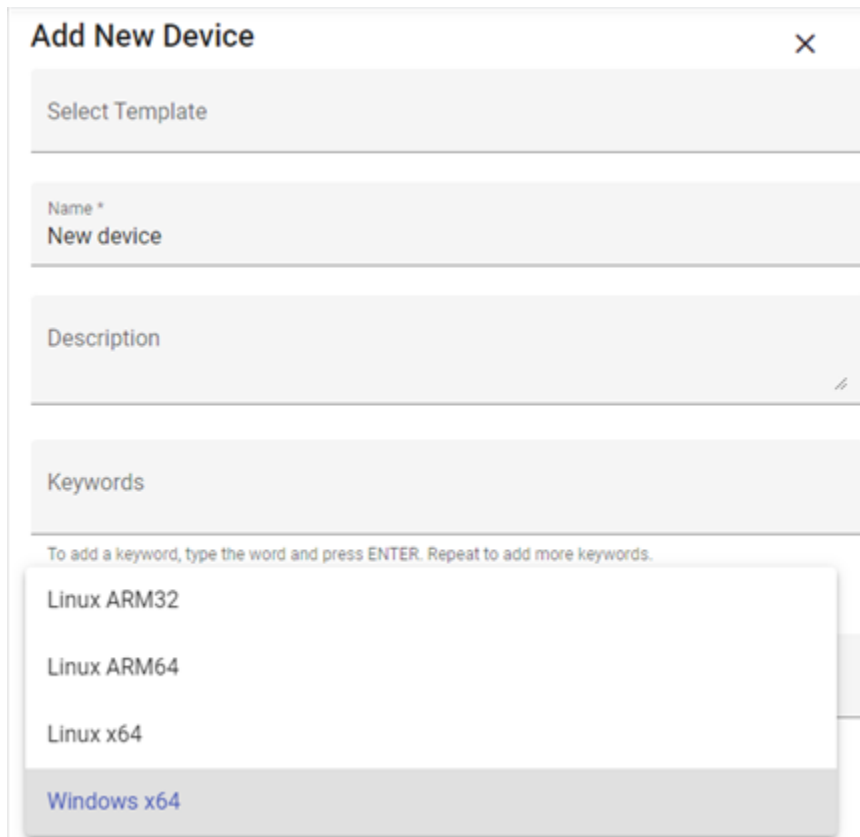
1. Template support for Linux modules on Windows devices
2. Disconnected devices can be deleted with "Force Delete" option

3. Edge services upgraded to .NET 6

August 2022 - What's New?

Notable features added

1. ARM 64-bit platform support
 - 32-bit and 64-bit ARM modules are separately available in the modules list for US and EU regions.
 - When choosing the platform type for a device, Linux ARM32 and Linux ARM64 are now available options:



The screenshot shows a web form titled "Add New Device" with a close button (X) in the top right corner. The form contains several input fields: "Select Template", "Name *" (with "New device" entered), "Description", and "Keywords". Below the "Keywords" field, there is a text prompt: "To add a keyword, type the word and press ENTER. Repeat to add more keywords." A dropdown menu is open below the "Keywords" field, displaying four options: "Linux ARM32", "Linux ARM64", "Linux x64", and "Windows x64". The "Windows x64" option is highlighted with a blue background and text.

2. Edge SDK System Monitoring module for 64-bit ARM is now available.
3. Edge SDK Empty module for 64-bit ARM is now available.

Get Started

This section helps you get started working with AVEVA Edge Management.

Understand AVEVA Edge Management

With this release of AVEVA Edge Management, you can quickly register edge devices in CONNECT, deploy and manage edge modules, and monitor the health and status of your edge devices in the form of a device twin. As part of CONNECT, you have access to numerous modules to help support your business.

Get Started: Understand the AVEVA Edge Management Flow

How Things Work

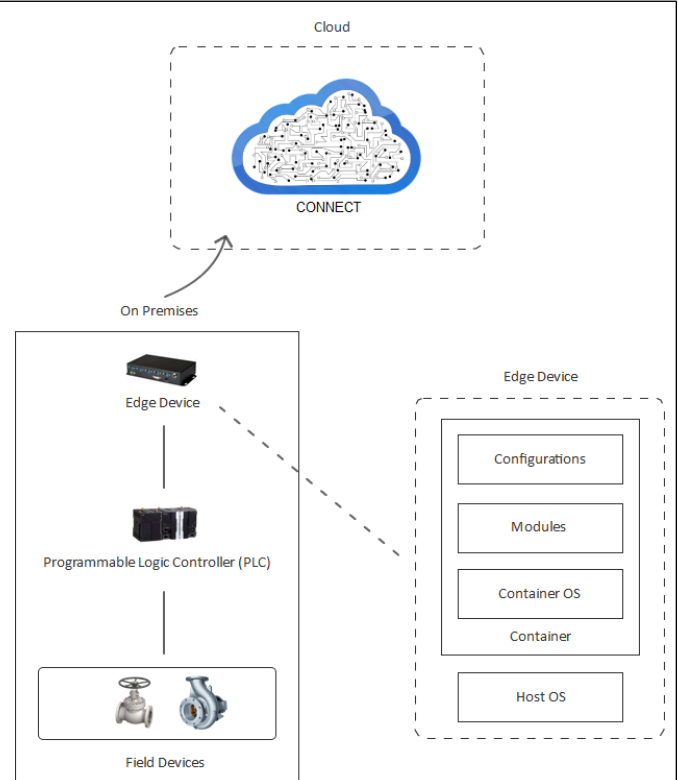
AVEVA Edge Management is a service within CONNECT that enables you to remotely deploy, configure, and manage AVEVA software on your edge devices.

What's an Edge Device?

Think of an edge device as a compact virtual machine that contains exactly what you need, without extra software bloat.

An edge device is composed of the following:

- **Configurations:** product-specific configurations the module will execute
- **Modules:** the AVEVA product executable or runtime that AVEVA provides
- **Container OS:** the components of the OS needed (Windows or Linux)
- **Host OS:** The host OS of the device (Windows or Linux)



What's a Device Twin?

A device twin is the virtual representation of your edge device in CONNECT.

Understand Operating System and Bandwidth Requirements

What Operating Systems are Supported?

Operating systems are classified as either Tier 1 or Tier 2, based on the level of support provided by Microsoft. Tier 1 operating systems are fully supported and tested by Microsoft. Tier 2 operating systems are compatible with Azure IOT Edge, but are not actively tested or maintained by Microsoft. See <https://learn.microsoft.com/en-us/azure/iot-edge/support?view=iotedge-1.5> for more information.

Edge modules currently include support for the following Tier 1 and Tier 2 operating systems and system architecture.

IMPORTANT: No other operating systems or operating system versions are supported at this time.

In order to successfully create, pair and deploy a device twin with an edge device, you must ensure that you install a supported operating system and version on the edge device. If you do not install a supported operating system and version compatible with your hardware, errors will occur. Additionally, you must apply all operating system updates prior to working with AVEVA Edge Management, as noted below.

Supported Tier 1 Operating Systems

Operating System	AMD64 Architecture	ARM64 Architecture	Notes
Ubuntu Server 20.04 LTS	✓	✓	<i>All operating system updates must be applied.</i>
Ubuntu Server 22.04 LTS	✓	✓	<i>All operating system updates must be applied.</i>
Windows 10 Professional 22H2	✓		<i>22H2, with all current cumulative updates installed.</i>
Windows 11 Professional 22H2	✓		<i>22H2, with all current cumulative updates installed.</i>
Windows Server 2019 Standard/Datacenter Version 1809	✓		<i>Minimum build 17763.5696, with all current cumulative updates installed.</i>
Windows Server 2022 Standard/Datacenter 21H2	✓		<i>Minimum build 20348.2402, with all current cumulative updates installed.</i>
Red Hat Enterprise Linux 8.8	✓		<i>All operating system updates must be applied. See additional notes below.</i>
Red Hat Enterprise Linux 9.0	✓		<i>All operating system updates must be applied. See additional notes below.</i>

Additional notes for Red Hat Enterprise Linux:

A Red Hat subscription is required before bootstrapping a device.

Run the following command on your device before attempting to bootstrap, substituting your Red Hat account details in place of <username> and <password>:

```
subscription-manager register --username <username> --password <password> --auto-attach
```

If this command does not complete successfully, bootstrapping your device will fail.

Supported Tier 2 Operating Systems

Operating System	AMD64 Architecture	ARM64 Architecture	Notes
Raspberry Pi OS 11 (Bullseye)		✓	<i>All operating system updates must be applied.</i>
Debian 11	✓	✓	<i>All operating system updates must be applied.</i>

What are the Minimum Bandwidth Requirements to Pair a Device?

750kbps.

What are the Recommended Bandwidth Requirements to Pair a Device?

1mbps or higher.

Understand Browser Support

What browsers are supported?

AVEVA Edge Management supports the following browsers:

Operating System	Google Chrome	Microsoft Edge	Safari	Mozilla Firefox
Windows 10/11	✓	✓		Limited support
Windows Server 2019/2022	✓	✓		Limited support
Ubuntu 18.04 LTS	✓			Limited support
Ubuntu 20.04 LTS	✓			Limited support
Ubuntu 22.04 LTS	✓			Limited support

Operating System	Google Chrome	Microsoft Edge	Safari	Mozilla Firefox
iPad IOS 10.2+	✓		✓ (with Retina display in landscape mode)	Limited support

Understand Supported Modules

AVEVA Edge Management currently supports the following modules:

Module Name	Supported Operating Systems
AVEVA™ Edge IoT View module	Linux (x64) on Linux or Windows devices
AVEVA™ Adapter for BACnet Module	Linux (x64) on Linux or Windows devices Linux (ARM64) on ARM devices
AVEVA™ Adapter for Modbus TCP Module	Linux (x64) on Linux or Windows devices Linux (ARM64) on ARM devices
AVEVA™ Adapter for MQTT module ^{1, 2}	Linux (x64) on Linux or Windows devices Linux (ARM64) on ARM devices
AVEVA™ Adapter for OPC UA module ^{1, 2}	Linux (x64) on Linux or Windows devices Linux (ARM64) on ARM devices
Edge Data Store module ^{1, 2}	Linux (x64) on Linux or Windows devices Linux (ARM64) on ARM devices

1 The container OS for these modules is Alpine Linux.

2 Additional security information: The indicated Linux modules are deployed as self-contained applications set to 'run-as' a non-privileged system account (emmuser). Login and shell are explicitly disabled for this account. See NIST SP 800-190 Application Container Security Guide for more information on container security measures.

Understand AVEVA Flex Licensing

When you select Edge Modules and deploy an Edge Device, AVEVA Flex credits are deducted from your license credit balance. The number of credits deducted is dependent on the duration for which you deploy your Edge Device. When you define a duration for deployment, the credits being subtracted for the deployment will be displayed alongside your credit balance remaining.

For detailed information on AVEVA Flex, refer to <https://sw.aveva.com/flex-subscription>.

Understand Firewall Exception Prerequisites

CONNECT and AVEVA Edge Management facilitate the transfer of data between your edge devices and device twins. The protocols we recommend help facilitate speed and continuous connectivity.

In order to ensure AVEVA Edge Management can communicate with your edge devices, you must open the following required outbound ports on your firewall depending on the protocol you wish to use. Ports marked as optional may improve connection speeds.

Protocol	Port	Required, or Optional?
TCP	80	Required
HTTPS	443	Required
MQTT	8883	Optional
MQTT over WebSockets	443	Optional
AMQP	5671	Optional
AMQP over WebSockets	443	Optional (Required if 5671 is not open)

Note: You do not need to open inbound ports.

Allowlisting Domain Names for Outbound Communication

In instances where your IT department is unable to fully open an outbound port, as described in [Understand Firewall Exception Prerequisites](#), they can instead allowlist the necessary ports for the following specific domains:

Device to Edge Management Service Communication

Domain Name	Port	Used For
edgeiothubprodus.azure-devices.net	443, 5671	Azure IoT Hub access (Port 5671 required for using AMQP)
gsr1edgmoduleacr.azurecr.io	443	Container Registry access
edgemanagement.connect.aveva.com	443	Downloading bootstrap scripts
edgestorageprodus.blob.core.windows.net	443	Downloading application modules

Domain Name	Port	Used For
edgestorageprodus.blob.core.windows.net	443	Downloading license certificates
edgemanagement.connect.aveva.com		
edgestoragestage.blob.core.windows.net		
stageedgemanagement.capdev-connect.aveva.com		
online.wonderware.com	443	AVEVA Insight REST APIs
configurator.online.wonderware.com		
mcr.microsoft.com	443	Downloading Edge Agent/Hub images
raw.githubusercontent.com	443	Deploying and initializing IoT Edge runtime and Edge Security Daemon
packages.microsoft.com	443	Downloading Linux packages

Browser Access to AVEVA Edge Management Portal

Domain Name	Port	Used For
connect.aveva.com	443	CONNECT Portal
identity.connect.aveva.com	443	Identity Server
signin.connect.aveva.com		
sso.aveva.com		
profile.connect.aveva.com		
edgemanagement.connect.aveva.com	443	AVEVA Edge Management Portal and REST API
api.connect.aveva.com	443	CONNECT REST API

Understand Proxy Server Configuration Settings

If your place of work requires that you connect to a proxy server, you must also specify that server for a device twin and edge device machine before you attempt to pair them. [Configure License and Device Default Settings](#) includes a step detailing where you can define a proxy server for a device twin. On an edge device, you must define the proxy server as an operating system-specific configuration setting.

AVEVA Edge Management expects that the proxy server URL is in the following format for Microsoft Windows and Linux machines: <URL>:<PORT>. For example, `http://gateway.mydomain.net:9999`. Please contact a system administrator at your place of work if you are unsure of your proxy server configuration or HTTP proxy URL.

Review Known Issues

There are a few things we want you to know about before you start to work with this release of AVEVA Edge Management.

The following table lists known issues at the time of release:

Issue	Workaround
Pairing more than one edge device with the same device twin may result in unpredictable deployment/connection states.	There are no specific workarounds. We recommend not pairing more than one edge device with the same device twin, except in the case of a device failure, when you want the new device to replace the failed device.
Uploading an invalid or corrupt application during module configuration may result in the device status getting stuck in the "Deploying" state.	Ensure that a valid package is uploaded during module configuration.
Specifying a value for the port number during module configuration that is already in use on the edge device may lead to the module stopping, or failing to start. This can happen when you inadvertently specify a port number that is already in use by other software running on the edge device.	Before configuring the module, ensure that the port number is not already in use on the edge device.

Setting Up AVEVA Edge Management

This section helps you configure AVEVA Edge Management so that you can start configuring and deploying Edge devices.



Manage User Access

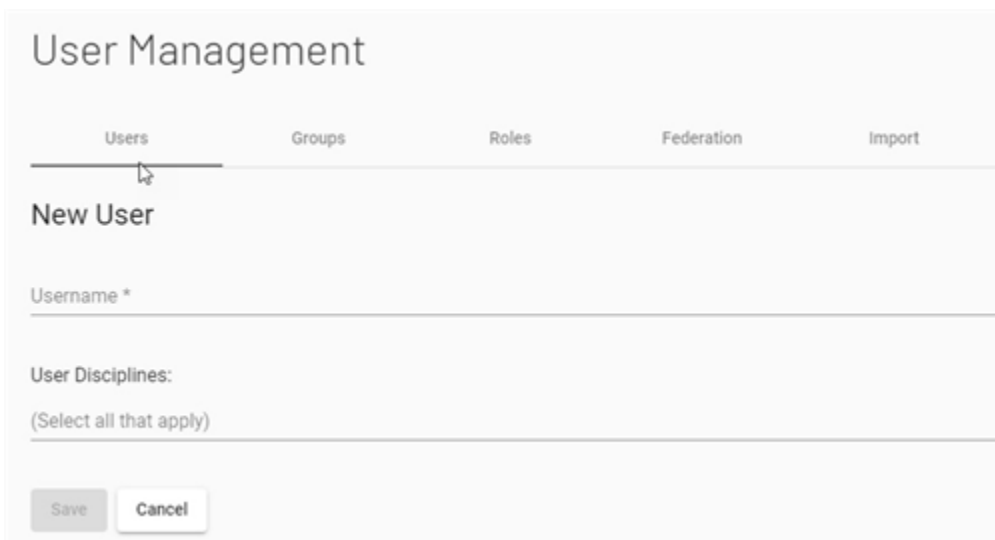
If you are an administrator with administrator security role, you can manage access to AVEVA Edge Management.

Create a User Account

If you are an administrator with administrator security role, you can invite users to use AVEVA Edge Management.

To create a user account:

1. From the CONNECT portal, select .
2. Select **User Management**.
3. Select the **Users** tab.
4. Select .
5. Specify the new **Username**.





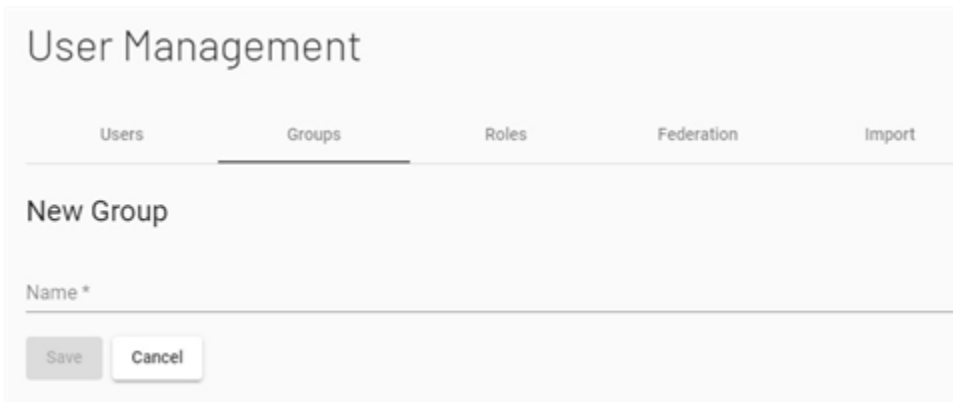
6. Select the new user's **User Disciplines**. The discipline(s) you select must provide insight on their role(s).
7. If you will be configuring a Federation Provider, select **Disable invitation email** to allow the provider to manage notifying the new user.
8. Select **Save**.

Create a User Group

If you are an administrator with administrator security role, you can create a new AVEVA Edge Management group.

To create a user group:

1. From the CONNECT portal, select .
2. Select **User Management**.
3. Select the **Groups** tab.
4. Select .
5. Specify the new group **Name**.



The screenshot shows the 'User Management' interface with tabs for Users, Groups, Roles, Federation, and Import. The 'Groups' tab is selected. Below the tabs is a 'New Group' section with a 'Name *' input field and 'Save' and 'Cancel' buttons.

6. Select **Save**.

Understand User Roles

There are five user roles in AVEVA Edge Management:

- Administrator
- Template Manager
- Deployer
- Operator
- Read Only

You can refer to the following table for detailed information on the access permissions for each user role:

Resource	Operation	Administrator Role	Template Manager Role	Deployer Role	Operator Role	Read Only Role
Templates	GetMany	✓	✓	✓	✓	✓
	GetSingle	✓	✓	✓	✓	✓
	Create	✓	✓			
	UpdateRuntime	✓	✓			
	Delete	✓	✓			
Template Modules	GetMany	✓	✓	✓	✓	✓
	GetSingleByTemplateModule	✓	✓	✓	✓	✓
	GetByTemplate	✓	✓	✓	✓	✓
	GetSingle	✓	✓	✓	✓	✓
	Create	✓	✓			

Resource	Operation	Administrator Role	Template Manager Role	Deployer Role	Operator Role	Read Only Role
Devices	Delete	✓	✓			
	Update	✓	✓			
	Get	✓	✓	✓	✓	✓
	bootstrap					
	command	✓	✓	✓	✓	
	New	✓	✓	✓		
	Edit/Update	✓	✓	✓		
	Deploy	✓	✓	✓		
	Delete	✓	✓	✓		
	UnDeploy	✓	✓	✓		

Resource	Operation	Administrator Role	Template Manager Role	Deployer Role	Operator Role	Read Only Role
Settings	EnableLog	✓	✓	✓	✓	✓
	Search	✓	✓	✓	✓	✓
	Get	✓	✓	✓	✓	✓
	Update	✓	✓	✓		
DeviceModules	Get	✓	✓	✓	✓	✓
	Add	✓	✓	✓		
	Update	✓	✓	✓		
	Delete	✓	✓	✓		
Logs	Get	✓	✓	✓	✓	✓
Modules	Get	✓	✓	✓	✓	✓



Resource	Operation	Administrator Role	Template Manager Role	Deployer Role	Operator Role	Read Only Role
ConfigTypes	Platforms	✓	✓	✓	✓	✓
	WindowsVersions	✓	✓	✓	✓	✓
	Get	✓	✓	✓	✓	✓
	Get	✓	✓	✓	✓	✓
	Create	✓	✓	✓		
	Delete	✓	✓	✓		
DeviceKeywords	Get	✓	✓	✓	✓	✓
	Create	✓	✓	✓		
Packages	Get	✓	✓	✓	✓	✓
	Create	✓	✓	✓		

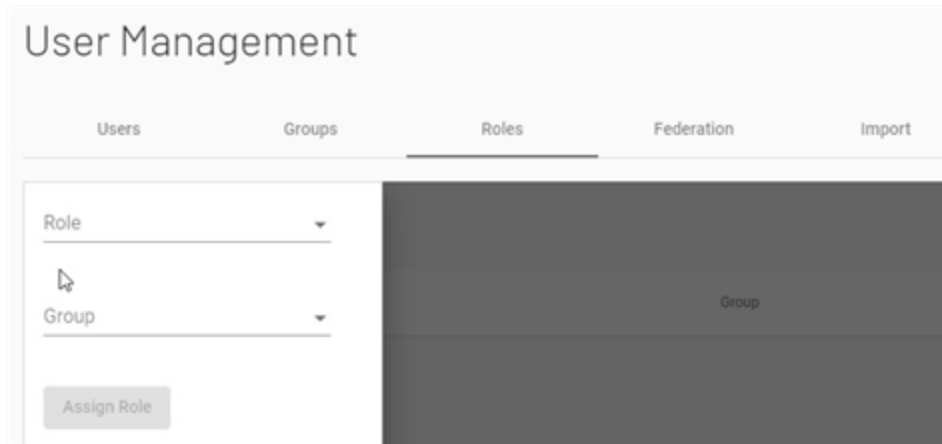
Resource	Operation	Administrator Role	Template Manager Role	Deployer Role	Operator Role	Read Only Role
Permissions	Upload	✓	✓	✓		
	Get	✓	✓	✓	✓	✓
Credits	Get Account Balance	✓	✓	✓	✓	✓
	Get Module Rates	✓	✓	✓	✓	✓
	Get Device Module Rates	✓	✓	✓	✓	✓

Assign a User Role

If you are an administrator with administrator security role, you can create a new AVEVA Edge Management user role or assign a user role.

To assign a user role:

1. From the CONNECT portal, select .
2. Select **User Management**.
3. Select the **Roles** tab.
4. Select the role you want to manage.
5. Select .
6. Select a **Role**.



7. Select a **Group** to assign the role to.
8. Select **Assign Role**.



Configure a Federation Provider

If you want to configure a federation provider, you must contact CONNECT Customer Support at connect.support@aveva.com for explicit instruction.

Import a List of Users

If you are an administrator with administrator security role, you can import a list of users of users directly into CONNECT through a Microsoft Excel document.


To import a list of users:

1. From the CONNECT portal, select .
2. Select **User Management**.
3. Select the **Import** tab.
4. Download the Microsoft Excel CONNECT import template from the URL provided on the page.
5. After populating the Microsoft Excel document, select .
6. Locate and select the Microsoft Excel document that contains the list of users, and select **Open**.

Using AVEVA Edge Management



This section helps explain how to work with AVEVA Edge Management.

Working with the Device List Interface

Before you create and start managing existing devices, you must understand the information that is displayed on the **Device List** page. Access the device list by selecting **Devices**  from the navigation panel.




Understand Your Device Status

Status symbols provide you with a visual indication of the deployment status of your virtual devices.

Deployment State	Status Symbol	Description
New	NEW	The device twin has been created, but does not contain any published modules.
Pending		A module has been added to or updated on the device twin, and needs to be deployed to the edge device.
Published		A module has been successfully published to the device twin, but the edge device has not yet downloaded the changes.
Deployed	DEPLOYED	All published modules have been downloaded to the edge device.
Unpublished	None	All modules have been removed from the device twin and the edge device.
Undeployed	None	All modules have been stopped and removed from the edge device.
Failed		A module has failed to deploy or undeploy successfully from the edge device, or there is a problem with the edge device itself.




Understand Your Device Connection Status

Status icons provide you with a visual indication of the connection status of your devices.


Device State	Status Icon	Description
Unpaired		The device twin is not yet paired with an edge device.
Paired	No icon displayed	The device twin and edge device have been paired.
Connected		The device twin is paired with an edge device, and a connection is established.
Disconnected		The device twin is paired with an edge device, but it has been disconnected for a duration longer than the timeout period. The default timeout period is 5 minutes.

Understand Your AVEVA License Status

AVEVA licensing symbols provide you with an at-a-glance status of the licensing of your devices.


Status	Description
No indicator	The device has not yet been licensed.
	The current license for the device will expire soon. Hovering over the icon displays a tooltip specifying how many days before the license expires.
	The current license for the device has expired.
	The device is licensed.

View Additional Keywords

If a device has more than one keyword associated with it, only the first keyword is displayed to reduce clutter. An additional icon  displays beside the keyword to indicate there are more keywords associated with the device than are currently showing.

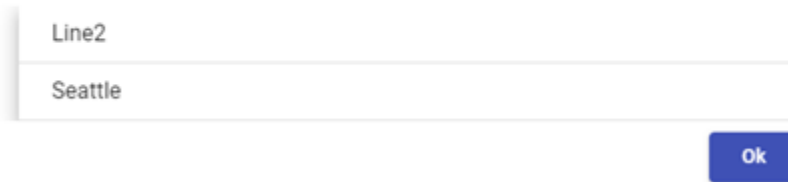
To view all keywords associated with a device:


1. Locate the device in the Device List.

2. Select  from the **Keywords** column.

A popup displays, showing a list of all keywords associated with the device.

Keywords for Seattle Production Line 2



3. Select  to dismiss the popup.

Managing Devices








This section helps explain how to work with individual devices.

Add a Device Twin




When you add a new device, it uses the latest supported version of the IoT Edge runtime, which is currently version 1.4 LTS.

If you have existing devices that were created with an older version of the IoT Edge runtime, you can uninstall the old runtime. See [Uninstalling the IoT Edge Runtime](#) for more details.

To add a new device twin:



1. From the device list, select **Add** . The **Add New Device** panel displays.
2. To add a device using a template, select a template from the **Select Template** list. This automatically sets some device values based on the template's settings.
 - Values that are derived from the template and cannot be changed are indicated by .
 - Values indicated by  are linked to the template, so their values are initially derived from the template. To change one of these values, select , changing it to  to indicate it is unlinked from the template. You can now edit the value.
 - To re-link a value to the template, select , changing it to . The value returns to the template's value.
3. Specify a **Name** to identify the device.
4. Specify a **Description** of the device.
5. Specify new **Keywords** or select existing keywords from the list, if desired.
A device twin can have as many keywords assigned as you like to help you classify your devices and facilitate searching.
6. Specify the **Device Type**.
7. Select the operating system **Platform** for the device. For example, if your device is a Raspberry Pi running a


32-bit Linux operating system, select **Linux ARM32**, while for a PC running Ubuntu, select **Linux x64**.

8. Select  to save the device information. To cancel adding a new device, select , , or **CANCEL**.
The new device is added to the list and is automatically selected.

Remove a Device Twin

To remove a device twin:

1. From the device list, locate and select the device you wish to remove. The device details display in the Device Information panel.
2. Select the **Summary** tab.
3. Scroll to the bottom of the **Summary** tab and select .
4. If the device is connected, you are prompted to confirm that you want to delete the device. Select  to delete the device.
5. If the device is disconnected, the system cannot communicate with the device to complete the deletion


process. You are prompted to confirm that you still want to delete the device. Select  to force the system to delete the device. This removes all references to the device from the system and prevents the device from reconnecting in the future.

Warning: When you delete a device, there is a 30-day grace period during which the device can be recovered. For assistance with recovering a deleted device, you can contact [AVEVA technical support](#). After the 30-day grace period has expired, a deleted device can no longer be recovered. Any remaining license credits on a deleted device are lost.







If an edge device is damaged or otherwise becomes inoperable, you can bootstrap a new edge device to link it with the existing device twin with no loss of license credits.

Configure a Module

To configure modules for a device:

1. From the Device List, locate and select the device you wish to configure.
The device details display in the Device Information panel.
2. Select the **Modules** tab.
A list of available modules displays. The modules available depend on the platform type configured for the device.
3. Drag a module tile from the list into the left column to associate the module with the device. You can also select the  icon on the module's tile.

Note: You can associate multiple modules with a device, but each individual module can only be associated with a device once. For example, you can configure a device with four different modules, but any specific module can only be configured once per device.

4. Specific configuration details differ by module. Refer to [Linux x64 Modules](#), [Linux \(ARM\) Modules](#), or [Linux ARM64 Modules](#) for more information.
5. If a device is created from a template, some module settings are controlled by the template.
 - Values that are derived from the template and cannot be changed are indicated by .
 - Values indicated by  are linked to the template, so their values are initially derived from the template. To change one of these values, select , changing it to  to indicate it is unlinked from the template. You can now edit the value.
 - To re-link a value to the template, select , changing it to . The value returns to the template's value.

About Secure Secrets

AVEVA™ Edge Management provides module developers with the ability to securely store sensitive information, such as user names and passwords. If the module you are configuring uses this feature, the module's settings page contains a **Secret Store** section.


To configure secure secrets for a module:

1. From the module's settings page, locate the **Secret Store** section and select **Setup**.

The **Setup module secret** dialog displays.



Note: This example is only a sample. The specific fields that are displayed vary by module.

2. Enter values for each of the fields in the **Setup module secret** dialog. By default, the text you enter is hidden, indicated by .

To toggle visibility and see the text you entered, select , changing it to . The text you enter in the field is now visible.

Setup module secret

User Name
user1

Password

Reset Create

- When you are finished, select **Create**.

A module command script is generated containing the secret information.

Setup module secret

Make sure the following steps are executed without fail for secure secrets changes to be propagated properly to the device.

- Deploy the device for changes to be reflected on the edge device.
- Run this command in the linux terminal on the edge device to pair the device to set up secrets on device.

Copy to clipboard

```
sudo wget -e use_proxy=yes -e
https_proxy= -O -
https://i/api/v1/moduleconfigs/880f880e-7d7c-46ae-
8446-56fda0fe6028/command | bash -s
d96aec70629179f2ea98159b2baaa3f6
s5YflrLdAytuSG0UqmvwdSN+mKZkd3G+cBlvzZtL7QEFwXlbKojxcaylUj
W/HgeINnAXHB0JjWjgVnoxKDDGL4CeXXG3oW5ag55JaBjeP700JY2b
t732V5voHrjXyVT
```

Close

- Copy this command, then run it on your edge device in a Linux terminal, or as an administrator in Windows Powershell, depending on your device's operating system.

If this is a new device, you can now pair your edge device with the device twin and deploy the device twin. See [Pair a Device Twin with an Edge Device](#) for more information.

If this is an existing device and you are updating the secret information, you must publish your device changes. See [Deploy a Device Twin](#) for more information.

Important: Every time you generate a new module command script to update secret information, you must run the script on your device, and also publish the device changes. Failure to perform both of these steps prevents the device from functioning correctly.

Linux x64 Modules

The following modules are currently supported on x64-based devices running Linux or Windows.

- [AVEVA™ Edge IoT View Module](#)
- [AVEVA™ Adapter for BACnet Module](#)
- [AVEVA™ Adapter for Modbus TCP Module](#)
- [AVEVA™ Adapter for MQTT Module](#)
- [AVEVA™ Adapter for OPC UA Module](#)
- [Edge Data Store Module](#)

Special Considerations for Linux Modules on Windows Devices

Exposing Services Externally on Windows Devices

For Linux modules hosted on Windows devices, the module's ports are not automatically exposed by the host device. Port forwarding enables external access to Linux modules hosted on a Windows device by redirecting communication from a specific port on the host device to a port on the module.

To expose a module port to the external network, run the following command in an administrative command prompt on the Windows device:

```
netsh interface portproxy add v4tov4 listenport=<port> connectaddress=%AVEVALocalEdgeVM%  
connectport=<port>
```

After running this command, incoming requests received by the device on the *listenport* are mapped to *connectport* on the module, enabling external access to the module.

For example, a Linux module is hosted on a Windows device, and is listening for requests on port 12345. There is a firewall rule in place that allows the Windows device to accept requests on port 12390. After running the following command, requests received on port 12390 by the Windows device are forwarded to port 12345, allowing the module to respond to the requests:

```
netsh interface portproxy add v4tov4 listenport=12390 connectaddress=%AVEVALocalEdgeVM%  
connectport=12345
```

Choosing Custom DNS Servers for Linux Modules on Windows Devices

When hosting a Linux module on a Windows device, the bootstrapping process creates a Linux virtual machine on the device to sandbox Linux modules. This virtual machine will use the DNS server 1.1.1.1 by default, but if this DNS server cannot be reached during the bootstrapping process, the DNS configuration from the host Windows device is used instead.

Configuring a custom DNS server

1. On the Windows device, open the PowerShell CLI in administrator mode.
2. Test your chosen DNS server by attempting to resolve google.com with the following command, replacing 1.2.3.4 with the address of your DNS server:

```
Resolve-DnsName -Name google.com -Server "1.2.3.4"
```

If no errors occur and the IP address for google.com is displayed, continue to the next step.

3. Configure the DNS server that will be used by Linux modules using the following command:
`Set-EflowVmDNSServers -dnsServers @(<DNS Server address >)`
 To configure a primary and secondary DNS server, you can comma-separate multiple addresses.
 For example, this command sets 1.1.1.1 as the DNS server:
`Set-EflowVmDNSServers -dnsServers @("1.1.1.1")`
 This command sets 1.1.1.1 and 8.8.8.8 as the primary and secondary DNS servers:
`Set-EflowVmDNSServers -dnsServers @("1.1.1.1", "8.8.8.8")`

Configure the AVEVA Edge IoT View Module

To configure the AVEVA Edge IoT View Module for Linux (x64):

1. Select the specific version of the module you wish to configure from the **Versions** drop-down.
2. Specify the **Studio Mobile Access Port** number. This is the TCP port that will be exposed on your edge device for accessing the Studio Mobile Web Access interface on your local network.
3. Specify the **OPC UA Server Port** number. This port will be exposed on your edge device for connections with the OPC UA server task.
4. Specify a value for Log Verbosity. This controls the detail level of messages logged to the IoTView log. Integer values from 0 to 5 are valid, where 0 results in no logging, and 5 results in the maximum amount of logging.
5. Select the **Project Configuration (Zip)** field to upload a configuration for the module. A file **Open** dialog displays. Locate and select the ZIP file containing your project, and click **Open**.

Important: Make sure your project is built using a version of AVEVA Edge Studio that is compatible with the module version you selected in step 1. Uploading an incompatible project configuration can cause your device to get stuck in the "deploying" state. If this occurs, you can fix the issue by selecting a compatible project ZIP file and updating your device once again.

6. Select  to save your changes.

Configure the AVEVA Adapter for MQTT Module

The AVEVA Adapter for MQTT Module is a data-collection component that transfers time-series data from source devices to OMF endpoints in AVEVA Data Hub or PI Servers. MQTT (Message Queuing Telemetry Transport) is a messaging protocol created for Machine-to-Machine (M2M)/Internet of Things (IOT) communication. The adapter can connect to any device that uses the MQTT protocol for communication with constrained devices and server applications for data exchange. The AVEVA Adapter for MQTT Module runs on Linux (x64) and Linux (ARM64).

Use the secret store to complete the secure transfer of secret values to the device. The configuration file should contain secret variables that act as placeholders for the actual secrets and passwords. Complete the secret store configuration in AVEVA Edge Management to define values for the secret variables. AVEVA Edge Management encrypts the values and provides the Setup module secret command. After you deploy the module configuration, run the Setup module secret command in a Linux terminal on the device to replace the secret variables with the encrypted values.

To configure the AVEVA Adapter for MQTT Module for Linux (x64):




1. Select the specific version of the module you want to configure from the **Versions** drop-down.
2. Specify the **Deployment Name**. This identifies the module deployment on the device.
3. Select the **Configuration File** field to upload a configuration file for the module. A file Open dialog displays. Locate and select the JSON file containing the configuration, then click **Open**. If you do not specify a configuration file, the module uses the default configuration.

For information on modifying the configuration on the device, refer to the [PI Adapter for MQTT](#) documentation.

4. Specify the **Connection Port**. This port is used for local configurations.
5. In the **Secret Store** section, select **Setup**.
6. In the **Setup module secret** dialog, enter the secrets or passwords to replace the secret variables in the configuration file.

Enter values only for the secrets and passwords required to configure the module.

- **Egress Endpoint Secret** - Enter the secret or password value to connect to the egress endpoint. When the egress endpoint is AVEVA Data Hub, this is the client secret. When the egress endpoint is PI Web API, this is the password.
- **Additional Egress Endpoint Secret** - Enter the secret or password for the additional endpoint when egressing data to more than one endpoint and the other endpoint requires a different password.
- **Data Source Secret** - Enter the data source password when the data source requires a password in order to connect.
- **Additional Data Source Secret** - Enter the secret or password for the additional data source when connecting to more than one data source and the other data source requires a different password in order to connect.

By default, the text you enter is hidden, indicated by . To toggle visibility and see the text you entered, select , changing it to . The text you enter in the field is now visible.

7. When you are finished, select **Create**.

A module command script is generated containing the encrypted secret information. Use this command to replace the variables in the configuration file on the device with the encrypted secrets.

8. Copy this command, then after you deploy the module configuration, run the command on the edge device in a Linux terminal.

If this is a new device, you can now pair your edge device with the device twin and deploy the device twin. See [Pair a Device Twin with an Edge Device](#) for more information.

If this is an existing device and you are updating the secret information, you must publish your device changes. See [Deploy a Device Twin](#) for more information.

Important: Every time you generate a new module command script to update secret information, you must run the script on your device, and also publish the device changes. Failure to perform both of these steps prevents the device from functioning correctly.

9. Select  to save your changes.

Configure the AVEVA Adapter for OPC UA Module

The AVEVA Adapter for OPC UA Module is a data-collection component that transfers time-series data from source devices to OMF endpoints in AVEVA Data Hub or PI Servers. OPC UA (OPC Unified Architecture) is an open standard, machine-to-machine communication protocol for industrial automation developed by the OPC Foundation. The adapter can connect to any device that uses the OPC UA communication protocol. The AVEVA Adapter for OPC UA Module runs on Linux (x64) and Linux (ARM64).

Use the secret store to complete the secure transfer of secret values to the device. The configuration file should contain secret variables that act as placeholders for the actual secrets and passwords. Complete the secret store configuration in AVEVA Edge Management to define values for the secret variables. AVEVA Edge Management encrypts the values and provides the Setup module secret command. After you deploy the module configuration, run the Setup module secret command in a Linux terminal on the device to replace the secret variables with the encrypted values.

To configure the AVEVA Adapter for OPC UA Module for Linux (x64):




1. Select the specific version of the module you want to configure from the **Versions** drop-down.
2. Specify the **Deployment Name**. This identifies the module deployment on the device.
3. Select the **Configuration File** field to upload a configuration for the module. A file Open dialog displays. Locate and select the JSON file containing the configuration, then click **Open**. If you do not specify a configuration file, the module uses the default configuration.

For information on modifying the configuration on the device, refer to the [PI Adapter for OPC UA](#) documentation.

4. Specify the **Connection Port**. This port is used for local configurations.
5. In the **Secret Store** section, select **Setup**.
6. In the **Setup module secret** dialog, enter the secrets or passwords to replace the secret variables in the configuration file.

Enter values only for the secrets and passwords required to configure the module.

- **Egress Endpoint Secret** - Enter the secret or password value to connect to the egress endpoint. When the egress endpoint is AVEVA Data Hub, this is the client secret. When the egress endpoint is PI Web API, this is the password.
- **Additional Egress Endpoint Secret** - Enter the secret or password for the additional endpoint when egressing data to more than one endpoint and the other endpoint requires a different password.
- **Data Source Secret** - Enter the data source password when the data source requires a password in order to connect.
- **Additional Data Source Secret** - Enter the secret or password for the additional data source when connecting to more than one data source and the other data source requires a different password in order to connect.

By default, the text you enter is hidden, indicated by . To toggle visibility and see the text you entered, select , changing it to . The text you enter in the field is now visible.

7. When you are finished, select **Create**.

A module command script is generated containing the encrypted secret information. Use this command to replace the variables in the configuration file on the device with the encrypted secrets.

8. Copy this command, then after you deploy the module configuration, run the command on the edge device in a Linux terminal.

If this is a new device, you can now pair your edge device with the device twin and deploy the device twin. See [Pair a Device Twin with an Edge Device](#) for more information.

If this is an existing device and you are updating the secret information, you must publish your device changes. See [Deploy a Device Twin](#) for more information.

Important: Every time you generate a new module command script to update secret information, you must run the script on your device, and also publish the device changes. Failure to perform both of these steps prevents the device from functioning correctly.

9. Select  to save your changes.

Configure the Edge Data Store Module

The Edge Data Store Module is a data-collection component that stores time-series data from source devices until it can be transferred to OMF endpoints in AVEVA Data Hub or PI Servers. The Edge Data Store Module runs on Linux (x64) and Linux (ARM64).




Use the secret store to complete the secure transfer of secret values to the device. The configuration file should contain secret variables that act as placeholders for the actual secrets and passwords. Complete the secret store configuration in AVEVA Edge Management to define values for the secret variables. AVEVA Edge Management encrypts the values and provides the Setup module secret command. After you deploy the module configuration, run the Setup module secret command in a Linux terminal on the device to replace the secret variables with the encrypted values.

To configure the Edge Data Store Module for Linux (x64):

1. Select the specific version of the module you want to configure from the Versions drop-down.
2. Specify the **Deployment Name**. This identifies the module deployment on the device.
3. Select the **Configuration File** field to upload a configuration for the module. A file Open dialog displays. Locate and select the JSON file containing the configuration, then click **Open**. If you do not specify a configuration file, the module is loaded with the default configuration.
For information on modifying the configuration on the device, refer to the [Edge Data Store](#) documentation.
4. Specify the **Connection Port**. This port is used for local configurations.
5. In the **Secret Store** section, select **Setup**.
6. In the **Setup module secret** dialog, enter the secrets or passwords to replace the secret variables in the configuration file.

Enter values only for the secrets and passwords required to configure the module.

- **Egress Endpoint Secret** - Enter the secret or password value to connect to the egress endpoint. When the egress endpoint is AVEVA Data Hub, this is the client secret. When the egress endpoint is PI Web API, this is the password.
- **Additional Egress Endpoint Secret** - Enter the secret or password for the additional endpoint when egressing data to more than one endpoint and the other endpoint requires a different password.
- **Data Source Secret** - Enter the data source password when the data source requires a password in order to connect.
- **Additional Data Source Secret** - Enter the secret or password for the additional data source when connecting to more than one data source and the other data source requires a different password in order to connect.

By default, the text you enter is hidden, indicated by . To toggle visibility and see the text you entered, select , changing it to . The text you enter in the field is now visible.

7. When you are finished, select **Create**.

A module command script is generated containing the encrypted secret information. Use this command to replace the variables in the configuration file on the device with the encrypted secrets.

8. Copy this command, then after you deploy the module configuration, run the command on the edge device in a Linux terminal.

If this is a new device, you can now pair your edge device with the device twin and deploy the device twin. See [Pair a Device Twin with an Edge Device](#) for more information.

If this is an existing device and you are updating the secret information, you must publish your device changes. See [Deploy a Device Twin](#) for more information.

Important: Every time you generate a new module command script to update secret information, you must run the script on your device, and also publish the device changes. Failure to perform both of these steps prevents the device from functioning correctly.

9. Select  to save your changes.

Linux ARM64 Modules

The following modules are currently supported on 64-bit ARM-based devices running Linux.

- [AVEVA™ Adapter for BACnet Module](#)
- [AVEVA™ Adapter for Modbus TCP Module](#)
- [AVEVA™ Adapter for MQTT module](#)
- [AVEVA™ Adapter for OPC UA module](#)
- [Edge Data Store module](#)

Configure the AVEVA Adapter for MQTT Module

The AVEVA Adapter for MQTT Module is a data-collection component that transfers time-series data from source devices to OMF endpoints in AVEVA Data Hub or PI Servers. MQTT (Message Queuing Telemetry Transport) is a messaging protocol created for Machine-to-Machine (M2M)/Internet of Things (IOT) communication. The adapter can connect to any device that uses the MQTT protocol for communication with constrained devices and server applications for data exchange. The AVEVA Adapter for MQTT Module runs on Linux (x64) and Linux (ARM64).

Use the secret store to complete the secure transfer of secret values to the device. The configuration file should contain secret variables that act as placeholders for the actual secrets and passwords. Complete the secret store configuration in AVEVA Edge Management to define values for the secret variables. AVEVA Edge Management encrypts the values and provides the Setup module secret command. After you deploy the module configuration, run the Setup module secret command in a Linux terminal on the device to replace the secret variables with the encrypted values.




To configure the AVEVA Adapter for MQTT Module for Linux (ARM64):

1. Select the specific version of the module you want to configure from the **Versions** drop-down.

2. Specify the **Deployment Name**. This identifies the module deployment on the device.
3. Select the **Configuration File** field to upload a configuration for the module. A file Open dialog displays. Locate and select the JSON file containing the configuration, then click **Open**. If you do not specify a configuration file, the module is loaded with the default configuration.
For information on modifying the configuration on the device, refer to the [PI Adapter for MQTT](#) documentation.
4. Specify the **Connection Port**. This port is used for local configurations.
5. In the **Secret Store** section, select **Setup**.
6. In the **Setup module secret** dialog, enter the secrets or passwords to replace the secret variables in the configuration file.

Enter values only for the secrets and passwords required to configure the module.

- **Egress Endpoint Secret** - Enter the secret or password value to connect to the egress endpoint. When the egress endpoint is AVEVA Data Hub, this is the client secret. When the egress endpoint is PI Web API, this is the password.
- **Additional Egress Endpoint Secret** - Enter the secret or password for the additional endpoint when egressing data to more than one endpoint and the other endpoint requires a different password.
- **Data Source Secret** - Enter the data source password when the data source requires a password in order to connect.
- **Additional Data Source Secret** - Enter the secret or password for the additional data source when connecting to more than one data source and the other data source requires a different password in order to connect.

By default, the text you enter is hidden, indicated by . To toggle visibility and see the text you entered, select , changing it to . The text you enter in the field is now visible.

7. When you are finished, select **Create**.
A module command script is generated containing the encrypted secret information. Use this command to replace the variables in the configuration file on the device with the encrypted secrets.
8. Copy this command, then after you deploy the module configuration, run the command on the edge device in a Linux terminal.

If this is a new device, you can now pair your edge device with the device twin and deploy the device twin. See [Pair a Device Twin with an Edge Device](#) for more information.

If this is an existing device and you are updating the secret information, you must publish your device changes. See [Deploy a Device Twin](#) for more information.

Important: Every time you generate a new module command script to update secret information, you must run the script on your device, and also publish the device changes. Failure to perform both of these steps prevents the device from functioning correctly.

9. Select  to save your changes.

Configure the AVEVA Adapter for OPC UA Module

The AVEVA Adapter for OPC UA Module is a data-collection component that transfers time-series data from source devices to OMF endpoints in AVEVA Data Hub or PI Servers. OPC UA (OPC Unified Architecture) is an open standard, machine-to-machine communication protocol for industrial automation developed by the OPC

Foundation. The adapter can connect to any device that uses the OPC UA communication protocol. The AVEVA Adapter for OPC UA Module runs on Linux (x64) and Linux (ARM64).

Use the secret store to complete the secure transfer of secret values to the device. The configuration file should contain secret variables that act as placeholders for the actual secrets and passwords. Complete the secret store configuration in AVEVA Edge Management to define values for the secret variables. AVEVA Edge Management encrypts the values and provides the Setup module secret command. After you deploy the module configuration, run the Setup module secret command in a Linux terminal on the device to replace the secret variables with the encrypted values.

To configure the AVEVA Adapter for OPC UA Module for Linux (ARM64):




1. Select the specific version of the module you want to configure from the **Versions** drop-down.
2. Specify the **Deployment Name**. This identifies the module deployment on the device.
3. Select the **Configuration File** field to upload a configuration for the module. A file Open dialog displays. Locate and select the JSON file containing the configuration, then click **Open**. If you do not specify a configuration file, the module uses the default configuration.

For information on modifying the configuration on the device, refer to the [PI Adapter for OPC UA](#) documentation.

4. Specify the **Connection Port**. This port is used for local configurations.
5. In the **Secret Store** section, select **Setup**.
6. In the **Setup module secret** dialog, enter the secrets or passwords to replace the secret variables in the configuration file.

Enter values only for the secrets and passwords required to configure the module.

- **Egress Endpoint Secret** - Enter the secret or password value to connect to the egress endpoint. When the egress endpoint is AVEVA Data Hub, this is the client secret. When the egress endpoint is PI Web API, this is the password.
- **Additional Egress Endpoint Secret** - Enter the secret or password for the additional endpoint when egressing data to more than one endpoint and the other endpoint requires a different password.
- **Data Source Secret** - Enter the data source password when the data source requires a password in order to connect.
- **Additional Data Source Secret** - Enter the secret or password for the additional data source when connecting to more than one data source and the other data source requires a different password in order to connect

By default, the text you enter is hidden, indicated by . To toggle visibility and see the text you entered, select , changing it to . The text you enter in the field is now visible.

7. When you are finished, select **Create**.

A module command script is generated containing the encrypted secret information. Use this command to replace the variables in the configuration file on the device with the encrypted secrets.

8. Copy this command, then after you deploy the module configuration, run the command on the edge device in a Linux terminal.

If this is a new device, you can now pair your edge device with the device twin and deploy the device twin. See [Pair a Device Twin with an Edge Device](#) for more information.

If this is an existing device and you are updating the secret information, you must publish your device changes. See [Deploy a Device Twin](#) for more information.

Important: Every time you generate a new module command script to update secret information, you must run the script on your device, and also publish the device changes. Failure to perform both of these steps prevents the device from functioning correctly.

9. Select  to save your changes.

Configure the Edge Data Store Module

The Edge Data Store Module is a data-collection component that stores time-series data from source devices until it can be transferred to OMF endpoints in AVEVA Data Hub or PI Servers. The Edge Data Store Module runs on Linux (x64) and Linux (ARM64).

Use the secret store to complete the secure transfer of secret values to the device. The configuration file should contain secret variables that act as placeholders for the actual secrets and passwords. Complete the secret store configuration in AVEVA Edge Management to define values for the secret variables. AVEVA Edge Management encrypts the values and provides the Setup module secret command. After you deploy the module configuration, run the Setup module secret command in a Linux terminal on the device to replace the secret variables with the encrypted values.

To configure the Edge Data Store Module for Linux (ARM64):




1. Select the specific version of the module you want to configure from the Versions drop-down.
2. Specify the **Deployment Name**. This identifies the module deployment on the device.
3. Select the **Configuration File** field to upload a configuration for the module. A file Open dialog displays. Locate and select the JSON file containing the configuration, then click **Open**. If you do not specify a configuration file, the module is loaded with the default configuration.

For information on modifying the configuration on the device, refer to the [Edge Data Store](#) documentation.

4. Specify the **Connection Port**. This port is used for local configurations.
5. In the **Secret Store** section, select **Setup**.
6. In the **Setup module secret** dialog, enter the secrets or passwords to replace the secret variables in the configuration file.

Enter values only for the secrets and passwords required to configure the module.

- **Egress Endpoint Secret** - Enter the secret or password value to connect to the egress endpoint. When the egress endpoint is AVEVA Data Hub, this is the client secret. When the egress endpoint is PI Web API, this is the password.
- **Additional Egress Endpoint Secret** - Enter the secret or password for the additional endpoint when egressing data to more than one endpoint and the other endpoint requires a different password.
- **Data Source Secret** - Enter the data source password when the data source requires a password in order to connect.
- **Additional Data Source Secret** - Enter the secret or password for the additional data source when connecting to more than one data source and the other data source requires a different password in order to connect.

By default, the text you enter is hidden, indicated by . To toggle visibility and see the text you entered, select , changing it to . The text you enter in the field is now visible.

7. When you are finished, select **Create**.

A module command script is generated containing the encrypted secret information. Use this command to replace the variables in the configuration file on the device with the encrypted secrets.

8. Copy this command, then after you deploy the module configuration, run the command on the edge device in a Linux terminal.

If this is a new device, you can now pair your edge device with the device twin and deploy the device twin. See [Pair a Device Twin with an Edge Device](#) for more information.

If this is an existing device and you are updating the secret information, you must publish your device changes. See [Deploy a Device Twin](#) for more information.


Important: Every time you generate a new module command script to update secret information, you must run the script on your device, and also publish the device changes. Failure to perform both of these steps prevents the device from functioning correctly.

9. Select  to save your changes.

Pair a Device Twin with an Edge Device

Once your device twin is deployed, it needs to be paired with your edge device.

To pair a device twin:

1. From the device list, locate and select the device you wish to pair. The device details display in the device information panel.
2. Select the **Summary** tab, and then select .
3. Copy the command and execute it on the edge device to associate the device twin with your edge device.


A bootstrap command line is generated.

Note: For Windows, run the command as an administrator using Windows PowerShell on the edge device. For Linux, run the command using the terminal on the edge device.

Deploy a Device Twin

To prepare your device twin for communicating with your edge device, it must be deployed.


To deploy a device twin:

1. From the device list, locate and select the device you wish to deploy. The device details display in the device information panel.
2. Select the **Deployment** tab.
3. For each module, specify the duration in months that the license will last before expiring.
4. For each module, select **Auto-renew** if you want the license automatically renewed when its term expires.
5. Select  to deploy the device.

Change License Expiration Terms

After your device has been deployed and paired, you can change the license expiration terms.

To change license expiration terms:

1. From the device list, locate and select the device you wish to update. The device details display in the Device Information panel.
2. Select the **Deployment** tab.
3. For each module, specify the duration in months that the license will last before expiring.
4. For each module, select **Auto-renew** if you want the license automatically renewed when its term expires.
5. Select  to update the device.

The remaining days the device is licensed is increased by the new duration entered in step 3. If you choose to have the license auto-renew, when the next expiration is due, the term is renewed by the new term length.

Example:

A device twin is deployed to and paired with an edge device, with its license set to expire after 1 month. After a few days have passed, the device has 23 days remaining before it needs to be re-licensed.

The device is reconfigured so the license is set to expire after 3 months, and redeployed.


Viewing the license status, the device now shows that 113 days are remaining before the license expires (90 days from the new term, added to the 23 days remaining on the previous term.)

Note: The number of days is determined by the number of days in the calendar month, rather than a fixed number of days per month.

View Device Logs

After your device has been deployed and paired, you can change the license expiration terms.

To view the error log for a device:

1. From the device list, locate and select the device you wish to view. The device details display in the device information panel.
2. Select the **Summary** tab, and then select .

A list of device status messages displays in a popup window.

View Device Resource Usage

After your device has been paired and deployed, you can view details about the system resources in use on your device.

To view the system resources used on a device:

1. From the device list, locate and select the device you wish to view. The device details display in the device information panel.

2. Select the **Summary** tab.

If the device is deployed and connected, the **Status** section displays details about the current system resources in use on the device, including:

- The current total amount of CPU usage on the device, expressed as a percent value
- The current total amount of system memory used and available on the device, expressed as a value in megabytes (MB) or gigabytes (GB)
- The current total amount of system storage used and available on the device, expressed as a value in gigabytes (GB)
- The amount of CPU and memory resources used by each module deployed to the device.

Notes: The resource usage details are collected when the device is selected, and represent a snapshot of resource usage at that moment in time. They are not continuously updated in real-time.

The sum of resources in use by all the modules deployed to the device is less than the totals shown in the **Status** section because the Edge runtime also uses some resources on the device. This usage is reflected in the totals, but not individually itemized.

Uninstalling the IoT Edge Runtime

To uninstall the IoT Edge runtime 1.1.x from a device:

1. Remove the IoT Edge runtime 1.1.x from your device with the following command:

```
sudo apt-get autoremove --purge iotedge
```
2. After the IoT Edge runtime is removed, any containers it created are stopped, but still exist on your device. Use the following command to view all containers on your device:

```
sudo docker ps -a
```
3. Delete each container from your device with the following command:

```
sudo docker rm -f <container name>
```
4. After deleting the containers, remove the container runtime with the following command:

```
sudo apt-get autoremove --purge moby-engine
```
5. Reboot your device.
The IoT Edge runtime is now removed from your device.

To uninstall the IoT Edge runtime 1.4.x from a device:

1. Remove the IoT Edge runtime 1.4.x from your device with the following command:

```
sudo apt-get autoremove --purge aziot-edge
```
2. After the IoT Edge runtime is removed, any containers it created are stopped, but still exist on your device. Use the following command to view all containers on your device:

```
sudo docker ps -a
```
3. Delete each container from your device with the following command:

```
sudo docker rm -f <container name>
```
4. After deleting the containers, remove the container runtime with the following command:

```
sudo apt-get autoremove --purge moby-engine
```

5. Reboot your device.

The IoT Edge runtime is now removed from your device.

Example: Configure an EFLOW device

The following example walks you through the process of setting up an Edge for Linux on Windows (EFLOW) device.

Set up your device

1. Install a supported version of Windows on a physical device or a virtual machine. See [Understand Operating System and Bandwidth Requirements](#) for details about supported Windows versions.
2. If your device is a virtual machine, ensure that nested virtualization is enabled.
 - a. If your device is an Azure virtual machine, choose a SKU that supports nested virtualization when you create the virtual machine. For example, Standard E8ds v5 (8 vcpus, 64 GiB memory).
 - b. If your device is a Hyper-V virtual machine:
 - i. Shut down the virtual machine.
 - ii. On the Hyper-V host, run the following command in an administrative PowerShell window, replacing <VMName> with the name of your VM as it is displayed in the Hyper-V management console:


```
Set-VMProcessor -VMName <VMName> -ExposeVirtualizationExtensions $true
```
 - iii. Start the virtual machine.
 - iv. Install Hyper-V within the virtual machine.

Configure your device

1. Create a device in the Edge Management portal, selecting Windows x64 as the Platform. See [Add a Device Twin](#) for more details.
2. Pair your device by running the pairing command in an administrative PowerShell window on your device. See [Pair a Device Twin with an Edge Device](#) for more details.

The pairing command installs required software on the device, and this may cause the device to reboot. If your device reboots, generate and run the pairing command again, then proceed to the next step. If your device does not reboot, continue with the next step.

3. The pairing process continues. When you are asked if you want to run software from an untrusted publisher identified as Microsoft Corporation, select A for Always run.

```
Stage completed.

Bootstrap supported client operating system with required build detected...

Deploying started, this will take few minutes depending upon machine configuration. Please Wait...

Do you want to run software from this untrusted publisher?
File C:\Program Files\WindowsPowerShell\Modules\AzureEFLOW\AzureEFLOW.psml is published by CN=Microsoft Corporation,
O=Microsoft Corporation, L=Redmond, S=Washington, C=US and is not trusted on your system. Only run scripts from trusted
publishers.
[V] Never run [D] Do not run [R] Run once [A] Always run [?] Help (default is "D"):
```

4. After the pairing process completes, verify the status of your device in the Edge Management portal.

Add the Edge Module SDK module to your device

1. Select your device in the Edge Management portal.
2. Select the **Modules** tab.
3. Add the **AVEVA Edge Module SDK Module** to your device.
4. Set the required parameters for the module:
 - a. Select which **Version** of the Edge Module SDK to install.
 - b. Enter the **Module Control Service Port** number.
 - c. Enter the **Module Control Service Local Host Port** number.
5. Select **Update** to save your changes.
6. Select the **Deployment** tab.
7. Enter a licence duration, then select **Publish** to update your device.
8. Allow a few minutes for your device to update, and then verify your device has changed to deployed status in the portal.

Verify the Edge Module SDK module

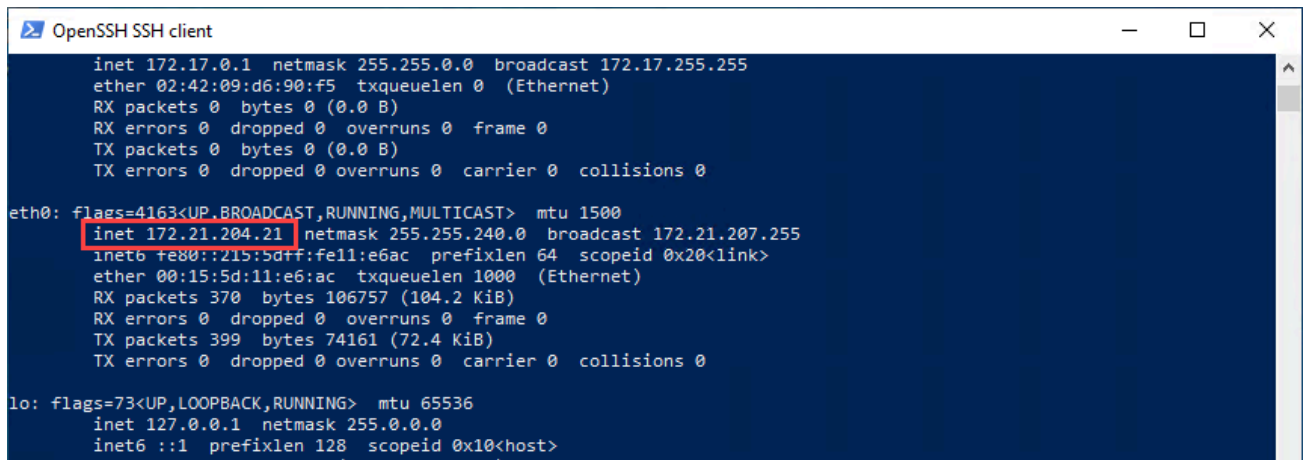
1. Run the following command in an administrative PowerShell window on your device to initiate a terminal connection to the Linux container:

```
Connect-EflowVm
```

2. Run the following command in the Linux container to display the current network configuration:

```
ifconfig
```

3. Identify the current IP address for eth0.



```
OpenSSH SSH client
inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
ether 02:42:09:d6:90:f5 txqueuelen 0 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 172.21.204.21 netmask 255.255.240.0 broadcast 172.21.207.255
    inet6 fe80::215:5dff:fe11:e6ac prefixlen 64 scopeid 0x20<link>
    ether 00:15:5d:11:e6:ac txqueuelen 1000 (Ethernet)
    RX packets 370 bytes 106757 (104.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 399 bytes 74161 (72.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    txqueuelen 1000 (Local Loopback)
```

4. Identify the Module Control Service Port configured for your Edge Module SDK module.

EFLOW TEST

Summary

Details

Modules

Deployment

Back To Add Modules >>

AVEVA Edge Module SDK Module

5.5/mo

⊖

Deployment Name *

edge_module_sdk_module

Module Control Service Port *

10000

Environment Name *

default_runtime_environemnt

Project Configuration (Zip) *

-

⬆

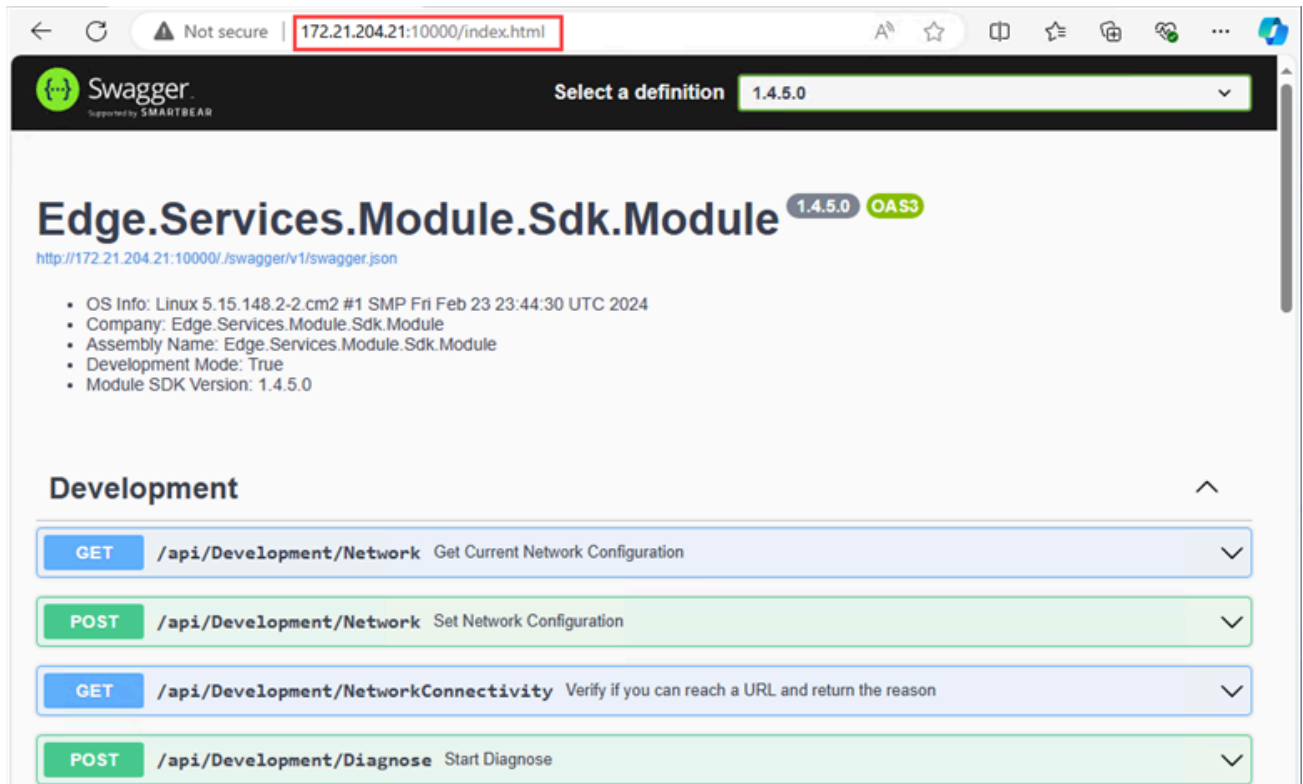
Example for Twin *

1

Module Control Service Local Host Po...


10001

- Open a browser on your device, and navigate to `http://<IP address>:<Module Control Service Port>` to display the SDK test page, verifying that your device and module are configured and running.



Working with the Template List Interface

Templates make managing groups of devices easier. A template contains device settings and module associations, just like a device twin. By creating device twins from a template, you can later apply changes to all the template's associated devices at the same time.

Before you create and start managing existing templates, you must understand the information that is displayed on the **Template List** page. Access the template list by selecting **Templates**  from the navigation panel.

Understand the Template List

The template list displays a list of configured templates. Each template is listed with a summary of useful information about the template, including:

- **Name** - The name of the template.
- **Description** - A description of the template (optional).
- **Devices** - The number of devices associated with the template.
- **Keywords** - A list of keywords associated with the template.

View Template Details

Select a template from the list to display the template details panel.


The **Summary** tab contains an overview of the template's status, including the number of devices associated with the template, the name and description of any modules associated with the template. It also displays the description, device type, platform type, and keywords associated with the template.

The **Details** tab enables you to edit the template's properties, including the name, description, associated keywords, device type, and platform type.

The **Modules** tab enables you to configure modules that will be associated with any devices created from the template.



The **Devices** tab lists the name, description and status of all the devices associated with the template.





Managing Templates

This section helps explain how to work with templates. Templates are used to configure and manage multiple devices as a group. Access the template list by selecting **Templates**  from the navigation panel.

Create a Template





To create a template:

1. From the Template List, select .
The **Add New Template** panel displays.
2. Specify a **Template Name** to identify the template.
3. Specify a **Template Description** to help identify the purpose of the template.
4. Specify new **Keywords** or select existing keywords from the list, if desired.
You can assign as many keywords assigned as you like to help you classify your devices and facilitate searching.
5. Specify the **Device Type**. By default, this field is locked, indicated by . When locked, all devices created using this template will use this device type setting.


To unlock this field, select  to change it to . When unlocked, a device created with this template uses the template's device type initially, but the value can be changed.
6. Select the operating system **Platform** for the template. This field is locked, and cannot be unlocked, indicated by . This means all devices created with this template use this operating system platform.
7. Select  to save the template.
The new template is added to the list and is automatically selected.

Update a Template

To update a template:

1. From the template list, locate and select the template you wish to update.
The template details display in the Template Information panel.
2. Select the **Details** tab.
3. Modify the template information and add keywords as required.
4. The icon beside the **Device Type** field indicates whether the field can be modified for a device created from the template. Select the icon to toggle between the following two states:
 -  indicates the field is locked, meaning devices created from the template inherit this value and it cannot be changed.
 -  indicates the field is unlocked, meaning devices created from the template inherit this value, but it can be modified.
5. Select the **Modules** tab.
6. Follow the steps in [Configure a Module](#) to associate a module with the template.
All devices created from this template will be configured to use the modules associated with the template.
7. The icon beside each of the module's fields indicates whether the field can be modified for a device created from the template. Select the icon to toggle between the following two states:
 -  indicates the field is locked, meaning devices created from the template inherit this value and it cannot be changed.
 -  indicates the field is unlocked, meaning devices created from the template inherit this value, but it can be modified.

Note: If the lock icon is faded  then it cannot be unlocked.

8. Select  to save your changes.
Changes made to the template are applied to any devices associated with the template, unless you have made module configuration changes that require publishing changes to the associated devices. In this case, a **Confirm update** dialog displays, listing all the affected devices.
9. Select the checkbox next to each device to which you want to publish your changes. Select the checkbox in the column header to select/deselect the entire list.


Confirm update and publish


Affected devices

Unaffected devices

<input checked="" type="checkbox"/> Publish	Name	Description	Keywords
<input type="checkbox"/>	L1		---
<input checked="" type="checkbox"/>	L2		---
<input type="checkbox"/>	L3		---
<input checked="" type="checkbox"/>	L5		---

4 device(s) to update and 2 device(s) to publish
Cancel
Update and publish

Notes: If a device is not currently licensed, an indicator  displays instead of a checkbox, and you cannot publish changes to that device. A device must be licensed before it can be published to.

If your change involves removing a module, and that module is the only module associated with a particular device, an indicator  appears instead of a checkbox and you cannot publish changes to that device. At least one module must be associated with a device before it can be published to.

10. Select UPDATE AND PUBLISH to save your changes and publish to the selected devices.
11. If any errors occur while publishing changes to the selected devices, an error message displays and you are returned to the **Confirm update** dialog where you can try again.

Delete a Template

To delete a template:

1. From the template list, locate and select the template you wish to remove.
The template details display in the Template Information panel.
2. Select the **Modules** tab.
3. Remove all modules associated with the template. You cannot delete a template if it has any modules associated with it.

After removing all associated modules, select UPDATE to complete module removal.

4. Select the **Summary** tab.
5. Select Delete template.
6. Select Delete to confirm deletion of the template.


Managing Settings

This section helps explain how to manage settings.

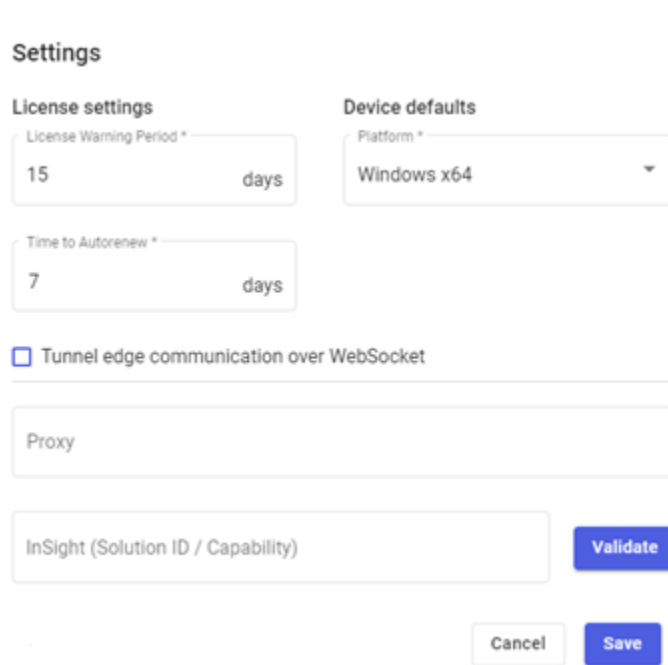
Configure License and Device Default Settings


The settings dialog allows you to control the default licensing behavior, and the default options for your devices.

To configure licensing behavior and device default settings:

1. From the navigation panel, select **Settings** .

The **Settings** dialog displays:



2. Specify the **License Warning Period**. This determines the number of days in advance of license expiration that a warning status is displayed.
For example, if the License Warning Period is set to 5 days, when a device has less than 5 days remaining before its license expires, the license status indicator changes from ✓ to ⚠.
3. Specify the **Time to Autorenew**. This determines the number of days in advance of license expiration that the license will be automatically renewed.
4. Select the default operating system **Platform** your edge devices use.
5. If the outgoing AMQP port 5671 is blocked by your firewall and you are not using a proxy server, select **Tunnel edge communications over WebSocket**.
6. Specify the URL of a **Proxy** server for your edge devices to use, if applicable.
7. If you also use AVEVA Insight, specify your **Insight Solution ID**. If not, leave this field blank.
8. Select .

Working Through Common Issues

This section provides you with information on how to work through some common issues and commands that will make your experience with the product better.

Understand Common Issue Scenarios and Possible Solutions

If an error occurs, refer to this section for ideas to help you quickly solve your problem and get you back to work.

I'm having trouble creating a new device twin.

Scenario	Possible Solution
When I try to add a device twin, an error occurs.	You may have already added a device twin with the same name. Try giving the device twin another name.

I'm having trouble configuring a module for my device twin.

Scenario	Possible Solution(s)
When I try to upload a project configuration file, an error occurs.	<p>You may be trying to upload an invalid file format or corrupted project configuration file. Ensure that the file type is correct and the integrity of the file has not been corrupted.</p> <p>You may be trying to upload an extraordinarily large project configuration file (over 30mb in size).</p> <p>You may have selected an old or unsupported module version number.</p>

I'm having trouble deploying my device twin. What are some common issues that may occur?

Scenario	Possible Solution
When I configure my device twin for deployment, validation errors occur.	You must ensure that the data you enter is valid.

I'm having trouble pairing my device twin with an edge device. What are some common issues that may occur?

Scenario	Possible Solution
I am unable to pair a device twin with an edge device when they each run on two different operating systems.	You must only attempt to pair a Linux device twin with a Linux edge device, etc.
I cannot execute commands in Powershell.	You must ensure that you run Powershell and Powershell ISE with administrator privileges.
When I execute commands in Powershell, errors occur.	You must ensure that you are running the 64-bit version of Powershell.
My module is not running.	In rare instances, your module may not start running due to a docker issue. To restart a module, execute the following: <code>iotedge restart <module name></code>
I am unable to access the AVEVA Edge Management web portal.	Clear your browser cache and try again.
After rebooting my Microsoft Windows edge device, my device is no longer paired.	Re-pair your edge device with your device twin.

Troubleshooting Errors with Device Deployment

If an error occurs while you are deploying a device, refer to this section for ideas to help you quickly solve your problem and get you back to work.

Error Message	Possible Solution
No previous deployments were detected. Can't perform Configuration Update Only. Suggest full redeploy for module {{moduleID}}.	<p>You are attempting to deploy a module for the first time without specifying a license duration. Select the Deployment tab, then select the module referred to by {{moduleID}} and specify the duration in months that the license will last before expiring.</p> <p>See Change License Expiration Terms for more information.</p>
The license for the module {{moduleID}} is expired. Please renew the license to update your configuration.	<p>You are attempting to deploy a module with an expired license. Select the Deployment tab, then select the module referred to by {{moduleID}} and specify the duration in months that the license will last before expiring. You can prevent this from happening again by selecting Auto-Renew.</p> <p>See Change License Expiration Terms for more information.</p>
Deployment Aborted. Failed to acquire license for one or more modules [{{moduleID}}].	<p>Edge Management was unable to fetch a license for one or more modules.</p> <p>Contact the Commercial Manager of your account to verify that credits are configured correctly for the module(s) you want to deploy.</p>

Administration - Modules and Credits

If you are a CONNECT administrator, and an account user is experiencing difficulties deploying a module due to licensing issues, please ensure the following items are correctly configured for the account:

1. Ensure the module has been added to the Product Catalog in CONNECT, and is correctly configured to use credits. See [Add a Module to the Product Catalog](#).
2. Ensure the associated CONNECT account has an active credit agreement configured and enabled. See [Add a Module to an Account's Active Credit Agreement](#).
3. Ensure the module is added to the active credit agreement of the associated CONNECT account. See [Add a Module to an Account's Active Credit Agreement](#).
4. Ensure the active credit agreement has sufficient credits available.

Add a Module to the Product Catalog

To add a module to the product catalog:

1. Log in to CONNECT with administrative privileges.
2. Select **Product Catalog** from the main menu. The Product Catalog displays.

Product Catalog

Filter by Product type, Product ID, Name or Code
iotview

Items per page: 10 Viewing 1 - 8

Hosting type	Product type	Product ID	Product name	Product code	Service	Rate plan	Chargeable type	Feature set or resource	Charging algorithm	Billing period	Unit threshold	Unit price
On-premises	AVEVA Edge HMIedge_hmi_iotvie (IoTView)	w_module	AVEVA Edge HMIedge_hmi_iotvie (IoTView)	w_module		Standard						
							Product	Product	FixedPerCreditsADay	-	10	
						Standard						
							User	Default	UniqueUsers	Day	-	20
						Standard						
							Resource	Default	Resource	Month	-	10
On-premises	AVEVA Edge HMIedge_hmi_iotvie (IoTView)	w_module_armv4 (IoTView)	AVEVA Edge HMIedge_hmi_iotvie (IoTView)	w_module_armv4		Standard						
							Resource	Default	Resource	Month	-	10
On-premises	AVEVA Edge HMIedge_hmi_iotvie (IoTView)	w_module1	AVEVA Edge HMIedge_hmi_iotvie (IoTView)	w_module1		Standard						
							Resource	Default	Resource	Month	-	5
On-premises	Edge Manageme nt System	iotviewmodule	InTouch Edge (IoTView) - Official	iotviewmodule		Standard						
							Resource	Default	Resource	Month	-	10
On-premises	Edge Manageme nt System	iotviewmodulelevi shwa3	iotviewmodulelevi shwa3	iotviewmodulelevi shwa3		Standard						

3. Select Add (+) to add a new module to the catalog.
4. Enter the **Product Type**, **Product ID**, and **Product Name**. Make sure you enter the values exactly as they are defined in the module's metadata.
5. Define the credit usage rate. For example:

Product type *
AVEVA Edge SD edge_module_sdk_module AVEVA Edge SDK Linux

Standard

Usage Resource Month \$

6. Click Save (💾) to finish updating the catalog.

Add a Module to an Account's Active Credit Agreement

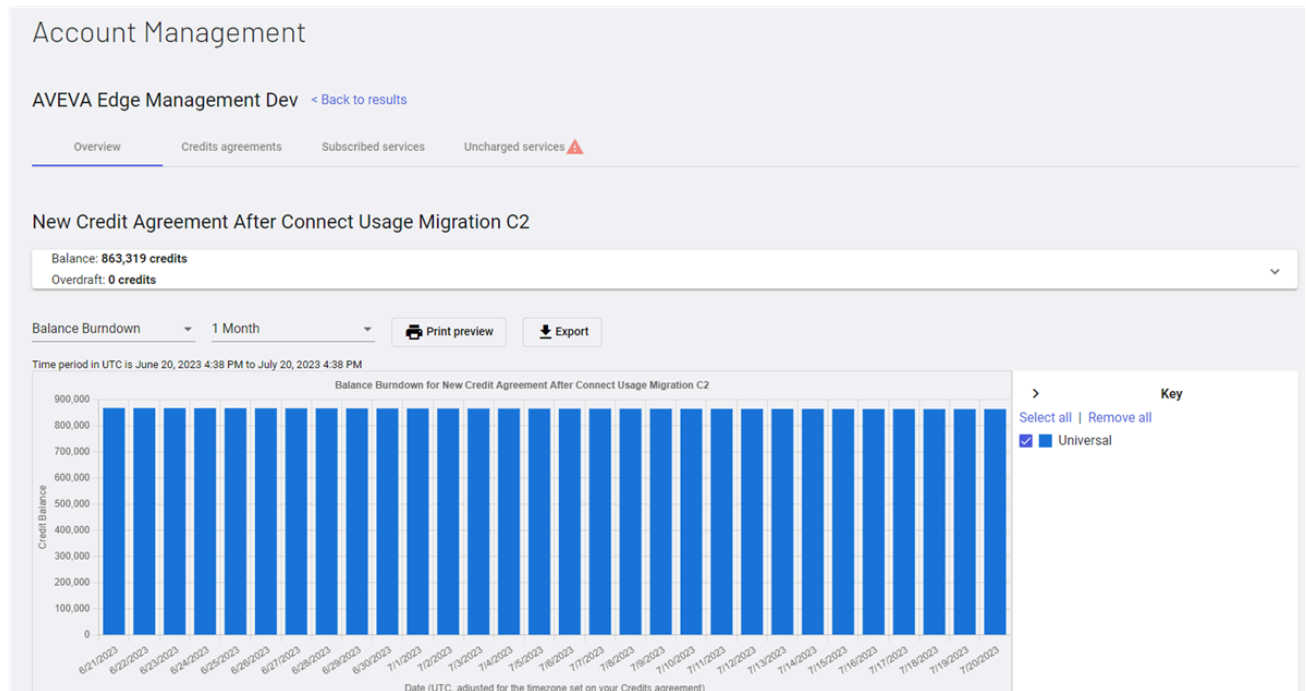
To add a module to an account's active credit agreement:

1. Log in to CONNECT with administrative privileges.
2. Select **Account Management** from the main menu.

Note: If you do not have access to the account management page, contact your account manager to perform

these steps.

- Use the filter options to locate and select the CONNECT account to which you want to add the module. The account details page displays for the selected account.



- Select the **Credits Agreement** tab. A list of all the account's credits agreements displays.

Account Management

AVEVA Edge Management Dev [Back to results](#)

Overview Credits agreements Subscribed services Uncharged services ▲

Credits agreements

ID		Effective Date	End Date	Status	Active Default ⓘ	Pending Default ⓘ	Reference	
AVEVA Edge Mangement Dev	View transactions	2/18/2020	2/18/2021	Expired			b4cb8c08-5260-45e0-9701-fadd7b82fa2a	
AVEVA Edge Mangement Dev Redundant	View transactions	9/17/2020	9/7/2023	Active			f975539f-930d-412d-9760-78c1d9962553	⋮
AVEVA Edge Mangement Dev 1	View transactions	3/14/2022	3/14/2023	Expired			a89fc147-fb09-482b-928d-70980cc8c59a	
AVEVA Edge Mangement Dev 2	View transactions	3/15/2022	3/13/2023	Expired			6b3c2631-96e0-4649-81c0-8c1fe2d1a899	
AVEVA Edge Management Dev4	View transactions	3/17/2022	4/11/2024	Active			9f81658c-9902-4e43-95c7-082efd57f460	⋮
AVEVA Edge Management Dev3	View transactions	3/17/2022	3/15/2023	Expired			d84344f2-e60d-4bc8-8e24-523fffb20dab	
AVEVA Edge Management Dev5	View transactions	4/20/2022	4/21/2023	Expired			52c1c569-5618-4e95-a7ce-3a21378c07e8	
New Credit Agreement After Connect Usage Migration C1	View transactions	6/30/2022	6/29/2023	Expired			2a5b0c91-480e-4cfd-97b0-ef0adbc06ff1	
New Credit Agreement After Connect Usage Migration C2	View transactions	6/30/2022	12/31/2024	Active			831ff43d-e7e9-49d1-a410-9c1ace3c3ec3	

- Select the active credit agreement to which you want to add the module.

Note: If there is not already an existing credits agreement available with sufficient credits, create a new contract, or contact your administrator who is authorized to create a credit agreement for you.

AVEVA Edge Management Dev < Back to results

Overview Credits agreement Subscribed services Uncharged services ⚠

Credits agreement: New Credit Agreement After Connect Usage Migration C2

Edit View History Discard

Status:	Active
Initial Credits:	
Universal:	5000
Total:	5000
Overdraft:	0
Effective Date:	6/30/2022
End Date:	12/31/2024
Account users' time zone:	(UTC - 06:00) Central Time (US & Canada), Easter Island, Guadalajara, Mexico City, Monterrey (Used for Entitlement access and Credit Charge periods. To ensure accurate charging, this is set to the primary location for account users.)
Charge credits for AVEVA domain users:	Yes

Rate Plans: AVEVA Edge HMI (IoTView)

Hosting Type	Product Code	Product Name	Service	Rate Plan Name	Chargeable Type	Feature Set or Resource	Charging Algorithm	Unit Threshold	Unit Price	User Limit
On-premises	edge_hmi_iotview_module	AVEVA Edge HMI (IoTView)		Standard	Resource	Default	Resource / Month	-	10	-

6. Select **Edit**, locate the module in the list, and configure the **Rate Plan** accordingly. Enable the **Override** option if you want to use different values than those defined in the product catalog.
7. When you are finished configuring the rate plan, scroll to the bottom of the page and select **Save**.

Note: Please allow up to 15 minutes for the rate plan changes to take effect.

Understand Helpful Commands

There are a few commands that will make your experience using AVEVA Edge Management better. This section contains a simple list of commands that you can execute on Windows and Linux operating systems.

This section will grow with each release.

How do I list modules that are running?

Operating System	Command
Windows	<code>iotedge list</code>
Linux	<code>sudo iotedge list</code>

How do I view the pairing time logs?

Operating System	Command
Windows	Get-IoTEdgeLog
Linux	sudo journalctl -u iotedge --no-pager --no-full

How do I view the proxy and environment status on a Linux machine?

Operating System	Command
Linux	sudo -E printenv http_proxy



AVEVA Group plc

High Cross
Maddingley Road
Cambridge
CB3 0HB
UK

Tel +44 (0)1223 556655

www.aveva.com

To find your local AVEVA office, visit **www.aveva.com/offices**

AVEVA believes the information in this publication is correct as of its publication date. As part of continued product development, such information is subject to change without prior notice and is related to the current software release. AVEVA is not responsible for any inadvertent errors. All product names mentioned are the trademarks of their respective holders.