cadence®

Allegro® X Pulse Configuration Guide

Product Version 23.1
September 2023

Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA.

Contents

1
Configuration Flow 9

2
Preparing for First-Time Configuration 19
Pre-Checks for Pulse Configuration 19

3
Configuration of Pulse Primary Node 23
Recommended Operating System for Pulse Nodes 23
Physical Hardware and Virtualization 23
Disk Space Requirements 24
Checks in Pulse 24
Installing Pulse Primary Node as a Service 26
Linux 26
Windows 28
Accessing Pulse Service Manager Web Page 29
Starting Pulse Service Manager 29
Opening Pulse Service Manager Web Page 30
Configuring Use of FQDN for Pulse Access URL 30
Modifying Java Memory for Pulse Primary Node 31
Modifying Disk Quota for Pulse Nodes 33
Specifying Purge Settings for Disk Management on Pulse Primary Node 34
Specifying Email Server Settings for Pulse Primary Node 37
Specifying Security Settings for Pulse Primary Node 39
Defining Library Management Settings for Pulse Primary Node 41
Shutting Down and Restarting Pulse Primary Node 47

4
Configuration of Pulse for Use by Various Applications 49
User Management in Pulse 50
Accessing Pulse User Management Module 51
Creating Users for the Pulse Platform 53
Roles in the Pulse Platform 56
Assigning Roles to Users on Pulse Platform 62
Configuring LDAP Sync with the Pulse Primary Node 66
Creating Groups to Automatically Share Projects 68
Configuration of Features Specific to Pulse 74
Using Global Configuration File 75
Configuring Unified Search Globally 79
Configuring Unified Search at Global or Project-Specific Level 95
Configuring Date Format 103
Configuring Live BOM Headers 105
Resetting Pulse Web Dashboard Changes to Default 109
Creating Project Templates 110
Configuring Workflows 112
Working with Notifications 120
Additional Object Types in the Allegro EDM-Managed Library Database 124
Enabling Optional Allegro EDM-Managed Library Features 127
Working with New Part Requests 131
Configuring Forms 134
Enabling PLM-Connected Project Creation 154
Enabling Diagnostic Test Case Download by Non-Administrator Users 156
Configuration of Publish for Manufacturing 157

5
Advanced Configuration and Pulse Setup for Use by Various Applications 159
Identification of Regular Synchronization Tasks and Frequency 159
Synchronization of Managed Library Database Across Multiple Sites 159
Configuration of Publish for Manufacturing-Specific Features in Pulse 160

6
Managing Test Server 161

7
Monitoring of Production and Staging Server 165
Cluster Health and Monitoring 165
Pulse Notifications 166
Self-Monitoring of Pulse Nodes 168
External Monitoring of Pulse 170

8
Maintaining Production Server for OrCAD and Allegro X Software 173

9
Miscellaneous Maintenance Tasks for Pulse 175

10
Allegro EDM 177
Allegro Library Manager 177
Library Workflows 179

11
Allegro EDM Configuration 1
Defining an Allegro EDM Root Directory 2
Creating Client Startup Script 6
Configuring Allegro EDM Flow Manager in Organizations with Controlled Computing Environment 8
Launching Allegro EDM Configuration Manager 10
Site Management in Pulse and Allegro EDM Environment 11
Working with Sites in Allegro X Pulse and EDM 12
Enabling Cadence-Supplied Flows for Allegro EDM 17

Environment Files Management in Allegro X Pulse and EDM 19
Control of PCB Editor Environment in Allegro EDM 19
Use of PCB Editor Environment File in Pulse and Allegro EDM 19
How Allegro EDM Controls Environment Files 20
Management of PCB Editor Environment Files 21
How Allegro EDM Uses Environment Files 24
Handling Manufacturing Retargetability 25
About tools.ini 27
Use of startworkbench Variables in tools.ini 28

12
Configuring Allegro EDM-Managed Library 29
Symbol Check Rules for Footprint Verification 29
Design and Schematic Model Rules with Rules Checker 33
Customizing Rules for Library and Design Flows 33
Inclusion of Custom Rules in Symbol Rules Checking Utility 38
Customizing Error Messages in Allegro EDM 55
Contents and Syntax of Match Files 55
Customizing Match Files for Archives of Model Types 63

13
Multi-Release Support with Multi-Library Release Configuration 1
Configuring Multi-Library Release Environment 2

A
Appendix 7
Modifying Pulse Primary Node Settings After Initial Configuration and Restarting Server 7
Modifying SSL Settings for Pulse Primary Node 8
Manually Modifying Files in the Pulse Primary Node Home Directory 8
Configuring a Node as a Pulse Data Node 9
Switching from Managed to Unmanaged Libraries and Vice Versa in Pulse Primary Node 12
Creating and Using SSL Certificates in Pulse Environment 16

Mapping of Pulse Internal Service Names and UI Terms 17
Troubleshooting Pulse Server Management 18
Allegro EDM Utilities Called by Allegro EDM Flow Manager 19
Allegro EDM Configuration Manager User Interface 22

B
Configuration Manager User Interface 27
Menu 28
Welcome Screen 29
Set up or Manage Allegro EDM Servers 30
Set up or Manage Company & Site 30
Manage Company and Site 30
Context Menu 32
Command Buttons 32
Set up Client 33
Server Status 34

1

Configuration Flow

This table provides a high-level overview of the Pulse configuration flow.

Table 1-1 Pulse Configuration Flow

Task	Target Audience
Prepare for first-time Pulse configuration: **1.** Make sure the IT administrator is ready with all the data for all the described parameters in *Allegro X Pulse Configuration Basics*. **a.** Make sure system configuration is per the recommendations in *Allegro X Pulse Configuration Basics*. **b.** Evaluate whether the disk space is enough for 1-2 years of the production environment. **c.** Be ready with the email notifications recipient list and SMTP server details. **2.** Make sure the ECAD administrator is ready with all the data for all the described parameters in *Allegro X Pulse Configuration Basics*. **a.** Assess whether you need to modify the Java Virtual Memory heap size. **b.** Be ready with configurations of Unified Search, New Part Request, Publish for Manufacturing, Live BOM, Workflows. See Preparing for First-Time Configuration for details.	IT administrator

Table 1-1 Pulse Configuration Flow

Task	Target Audience
Configure Pulse Service Manager (Vista): **1.** Install Pulse as a service. **2.** Access the Pulse Service Manager page for Pulse server administrators. **3.** Choose whether you want to use a fully qualified domain name (FQDN) or hostname for the Pulse access URL. **4.** Modify Java memory, if required. **5.** Specify purge settings for disk management. **6.** Specify email server settings for the Pulse primary node. **7.** Specify security and encryption settings for the Pulse primary node. **8.** Define library management settings for the Pulse primary node. **9.** Assign a disk quota for all client machines connecting to Pulse Service Manager. **10.** Shut down and restart the Pulse primary node. See Configuration of Pulse Primary Node for details.	IT administrator

Table 1-1 Pulse Configuration Flow

Task	Target Audience
Configure and set up Pulse for use by various applications: **1.** Identify the memory requirements. **2.** Create users and assign roles, and complete other tasks related to user management on the Pulse platform. **3.** Configure features specific to Pulse: **a.** Configure Unified Search. ❍ Part search rules ❍ Part content providers ❍ Filter display order ❍ Exclude filters ❍ Contents of the *Summary* column in part search ❍ Wild card or regular expressions in part search ❍ Filter locking ❍ Display of search results - property visibility, column widths, column reordering **b.** Configure the date format according to your requirements. **c.** Configure Live BOM headers.	ECAD administrator

Table 1-1 Pulse Configuration Flow

Task	Target Audience
d. Create projects that can be used as templates by designers. This is useful if you want to include company standards for page borders, the TOC, block diagrams, or just to drive a consistent process.	
e. Configure workflows.	
f. Enable new part requests and other component database optional features using the `adwschema` utility.	
g. Enable the part request process with review, if required.	
h. Configure forms by defining the fields you require in the project creation and part request forms. Enable project attachments in a project creation form and learn how to define what can be uploaded and who has the permission to upload attachments.	
i. Enable the auto-pull of PLM part numbers to integrate them in new part requests.	
j. Configure multi-library indexing.	
k. Enable indexing relational tables for OrCAD CIS. This is for Allegro X System Capture with OrCAD library users.	
4. Configure Publish for Manufacturing.	
See Configuration of Pulse for Use by Various Applications for details.	

Table 1-1 Pulse Configuration Flow

Task	Target Audience
After configuring optional Pulse features, complete advanced configuration and setup for Pulse for use by various applications: **1.** Identify regular synchronization tasks and their desired frequency. **a.** Synchronize the managed library database with PLM-owned attributes such as Cost, Regulatory Compliances, Manufacturer Data, and so on. **b.** Configure Data Exchange update rules. **2.** Configure features specific to Pulse: **a.** Configure the Library Synchronization service with the PLM system. **b.** Configure Publish for Manufacturing templates. **c.** Filter assembly or board numbers based on queries. See Advanced Configuration and Pulse Setup for Use by Various Applications for details.	ECAD administrator
Set up the staging/test server. The staging server is used for: ■ Operating system patches ■ Non-Cadence software deployment ■ Checking backup data of the production server ■ Evaluation of new features in HotFixes ■ Testing before deploying to production See Managing Test Server for details.	IT and ECAD administrators

Table 1-1 Pulse Configuration Flow

Task	Target Audience
Maintain the Pulse production server for OrCAD and Allegro X software. Also maintain it for non-OrCAD and Allegro X software. **1.** Install the OrCAD and Allegro X software upgrades. **a.** Check the DOs and DON'Ts. **b.** Verify that the software update is done correctly **c.** Uprev the managed library and Pulse databases. **2.** If you want to start using any Pulse-specific features, enable them. **3.** Add or delete nodes to and from the cluster. **4.** Update server access or authentication parameters. For example, enable SSO or update the LDAP connection settings. **5.** Update the following: ❑ Publish for Manufacturing templates ❑ Workflows ❑ New Part Request form **6.** Ensure that the Pulse server is ready for use by design clients. **For non-OrCAD and Allegro X products:** ➡ Update SSL certificates. See Maintaining Production Server for OrCAD and Allegro X Software for details.	ECAD administrator for OrCAD and Allegro X software IT administrator for non-OrCAD and Allegro X software

Table 1-1 Pulse Configuration Flow

Task	Target Audience
Complete miscellaneous maintenance tasks. These are adhoc and not regular maintenance tasks. **1.** Manage active users and disable or remove users who no longer require access to Pulse. **2.** Track active users - total number of licenses versus active users versus overdraft. **3.** Create diagnostic packages, that is, the medic test case generator. See Miscellaneous Maintenance Tasks for Pulse for details.	ECAD administrator
Learn about Allegro EDM. Allegro EDM is a collection of applications using which you can manage library data. See Allegro EDM for details.	Librarian
Learn how to configure Allegro EDM. See Allegro EDM Configuration for details.	Librarian
Configure Allegro EDM-managed Library. If you configured Pulse with the Allegro EDM-managed library option, you can do the following: **1.** Develop footprint authoring rules for inclusion in the symbol rules checking utility, The utility is part of footprint verification in the library flow. These rules enforce library standards. **2.** Define design and schematic model rules with the Rules Checker utility. Using the utility, you can check for violations of design and schematic model rules. See Configuring Allegro EDM-Managed Library for details.	Librarian

Table 1-1 Pulse Configuration Flow

Task	Target Audience
If you are an existing customer, and the Pulse primary and data nodes are in on different versions, configure Multi-Library Release support. **1.** Set up the MLR-enabled Pulse primary node. **2.** Distribute parts from the Pulse primary node from a lower release to a higher release. **3.** Run `lib_dist_client`. See Multi-Release Support with Multi-Library Release Configuration for details.	Librarian

Table 1-1 Pulse Configuration Flow

Task	Target Audience
See the appendix for the following: ■ Information on configuring a node as a Pulse data node ■ Steps to configure a node as a Pulse data node ■ Details on open-source software packages used by the Pulse solution ■ Pulse deployment in the Cloud ■ Deployment on virtual machines and DOs and DON'Ts ■ Support for server-client version independence for Design Entry HDL part search ■ Steps to modify the Pulse primary node settings after initial configuration ■ Information on how to restart the Pulse central server ■ Information on how to create and use SSL certificates in the Pulse environment ■ Steps to modify SSL settings for the Pulse primary node ■ Procedure to manually modify files in the Pulse primary node home directory ■ Procedure to switch from managed to unmanaged libraries and vice versa in the Pulse primary node ■ A table that maps internal Pulse service names such as Vista, Pantheon, with user-understandable names. ■ Information on troubleshooting Pulse configuration ■ Information on creating a central index of part data for faster part search. This is relevant for users who do not work with Pulse Service Manager. See Appendix for details.	All users

2

Preparing for First-Time Configuration

To prepare for Pulse configuration, do the following:

1. Make sure the IT administrator is ready with all the data for all the described parameters in *Allegro X Pulse Configuration Basics*.
    a. Make sure system configuration is per the recommendations in *Allegro X Pulse Configuration Basics*.
    b. Evaluate whether the disk space is enough for 1-2 years of the production environment
    c. Be ready with the email notifications recipient list and SMTP server details.
1. Make sure the ECAD administrator is ready with all the data for all the described parameters in *Allegro X Pulse Configuration Basics*.
    a. Assess whether you need to modify the Java Virtual Memory heap size.
    b. Be ready with configurations of Unified Search, New Part Request, Publish for Manufacturing, Live BOM, Workflows.

Pre-Checks for Pulse Configuration

Before you start to configure Pulse, meet the following conditions:

- Core 23.1 Allegro X products are installed on the designer machines, Pulse primary node, and the data nodes

  For the list of products to be installed, see *Cadence Allegro X and OrCAD (Including EDM) 23.1 Release Installation Guide for Windows*.

- The `vn2012`, `vn2013`, `vn2015`, and `vn2019` packages are installed on the designer machines, Pulse primary node, and the data nodes

  Note: You need to manually install these packages only if you installed the Cadence products from a network location.

- Ensure that the following firewall ports are open on the Pulse primary node for Pulse Service Manager to function correctly:
  - port 5701

    Note: The port does not need to be open on the Pulse data nodes or the clients connected to Pulse.
  - 7100, or the port specified by the administrator in the Pulse Service Manager configuration, and the two preceding ports—7099 and 7098 if you are using the default port—on every Pulse cluster node.
- If your organization has the team design license, libraries should be accessible by all team members. Designers must not use local libraries. Use of libraries available only to themselves is not recommended for designers.

Important

For deployments of the Pulse primary node on Linux, read the following and make the required changes:

- *https://www.elastic.co/guide/en/elasticsearch/reference/7.0/file-descriptors.html*
- *https://www.elastic.co/guide/en/elasticsearch/reference/7.0/vm-max-map-count.html*
- *https://www.elastic.co/guide/en/elasticsearch/reference/7.0/max-number-of-threads.html*
- Depending on the number of cores in the Pulse primary node, you might need to increase the maximum number of threads, that is, the `nproc` value. Refer to the following for details: *https://www.elastic.co/guide/en/elasticsearch/reference/master/max-number-of-threads.html*

It is recommended that you install Pulse as a service to ensure that the Pulse server starts up automatically in case of a machine restart.

After you check for these prerequisites, do the following to quickly get started:

1. Start the Pulse primary node as a service. See the following:
   - Installing Pulse Primary Node as a Service
   - Configuration of Pulse Primary Node

Because you might want to first deploy a new release on a test server or staging environment instead of your production environment, you can use a test server. See Managing Test Server for instructions on how to set it up.

2. Add users for access to the Pulse platform and assign roles. See:
   - ❑ Creating Users for the Pulse Platform
   - ❑ Assigning Roles to Users on Pulse Platform
3. Connect the Allegro System Capture, PCB Editor, or APD Plus client applications to the Pulse server.
4. Connect to the Pulse primary or data node depending on your configuration.
5. Authenticate the user:
   - ❑ If you installed Pulse as a service and did not reboot after installing, access the Pulse Service Manager web page through a web browser using the default access URL: `http://<hostname>:7100/element`. Then check whether Pulse is RUNNING.
   - ❑ If you started Pulse Service Manager by navigating to `<Cadence installation directory>/server/bin`, check the Pulse icon in the taskbar notification area. A RUNNING status, which is displayed when you hover over the icon, indicates that Pulse has started and that Pulse services are now available.

     A yellow dot on the icon () means there is a notification. For example, Pulse displays a notification message that you can change the Pulse home after you log in for the first time, or an information message is displayed that Pulse is in maintenance because of a scheduled maintenance tasks, such as backing up data.

     If Pulse is unavailable because of scheduled maintenance tasks, it resolves the notification itself and the yellow dot disappears. If it is a task you need to address, do so, and the dot and notification is no longer displayed.
6. Decide whether you want to work with managed or unmanaged libraries.
   - ❑ If you want to work with unmanaged libraries, skip directly to Defining Library Management Settings for Pulse Primary Node.
   - ❑ If you want to work with managed libraries, do the following:

     a. Configure a client startup script to launch Allegro EDM Flow Manager.

     Among other things, this utility is a cockpit to launch library management utilities such as Database Editor, Database Administration, Library Distribution, and so on. See Creating Client Startup Script.

b. Start Flow Manager.

c. Create a library project.

d. Log in to Allegro EDM Database Editor.

e. Import library data into the Allegro EDM environment using Allegro EDM Library Import.

Related Documentation

Allegro EDM Flow Manager User Guide

Allegro EDM Database Editor User Guide

Allegro EDM Library Import User Guide

3

Configuration of Pulse Primary Node

To configure the Pulse primary node, ensure that resources are available to Pulse when needed. For this reason, it is recommended that dedicated hardware be available for Pulse deployment. If deploying on a virtual machine (VM), it is recommended that you do not share the host with other applications or other VMs.

Recommended Operating System for Pulse Nodes

The Pulse server relies on a collection of open-source software packages, which run on Windows and Linux. In general, the open-source packages used by the Pulse server are first developed on Linux and are subsequently ported to Windows. The result is that the system behavior on the native operating system—Linux—is more stable than on Windows.

Additionally, while the packages run on both operating systems, the bulk of the ecosystems using these software packages in production environments deploy on Linux. This results in a more rigorous testing cycle of various deployment conditions, the results of which are fed back to the development teams of the software packages.

These two factors result in a slightly higher probability of unexpected behavior occurring and impacting operations initiated by users in Pulse-connected applications on Windows. Users might perceive these occurrences as instability. For this reason, Cadence recommends that the Pulse server be deployed on a Linux-based platform instead of Windows.

Physical Hardware and Virtualization

Dedicated resources are required by the Pulse Service Manager to ensure availability for computing tasks, input/output, memory, network, and so on when needed. Users' perception of stability in a Pulse-connected application might be influenced by the server response time and its ability to process requests in a timely manner.

Because of the nature of virtualization and its shared resource pool, Pulse Service Manager might take additional time in responding to requests from client machines. Further, overprovisioning of the resource pool can have a sizable impact on the performance of the

server response time when computing load is applied to other virtual machines (VM) sharing the same resource pool.

During these periods of increased computing load, the increased response times can be perceived by users as instability because the application does not respond in the time frame that they are used to.

Because of these performance risks, Pulse Service Manager displays a warning when it is run on a VM to alert users that they may be subject to performance variance resulting from the shared resource pool.

Disk Space Requirements

The Pulse solution brings together a collaborative design environment integrated with native data management, including version control. These features are provided on a native data platform, Pulse, for Cadence ECAD applications.

Because the Pulse platform requires a minimum available disk space to support all the users on the platform, it displays notifications to system administrators when the available space is approaching its capacity limit. The actual rate of storage consumption is generally proportional to the usage activity, such as the number of users, design size, and the number of design versions created. It is important for a system administrator to monitor the disk space growth rate and plan capacity upgrades preventively to ensure continuous and smooth operation.

If sufficient disk space is not available, Pulse Service Manager displays warning messages by default to system administrators. When available disk space is less than 100 GB, this warning is also displayed to non-administrator users. And, system administrators are notified by email as well, if *Notify Critical System Events to Administrator* is selected during Pulse server configuration. See Specifying Email Server Settings for Pulse Primary Node. Any further use of the Pulse platform will consume the remaining disk space and must be addressed by the administrator.

When the available disk space crosses the 20 GB threshold, Pulse goes into MAINTENANCE mode. The system administrator must clean up space and restart the Pulse server. The RAM and CPU minimum and maximum requirements, and the disk space threshold cannot be configured.

Checks in Pulse

Pulse displays alerts for hardware problems and for other cases.

Hardware Check

Pulse reports an error and does not start when the machines where the Pulse cluster is deployed do not meet the minimum hardware requirements. The notification alerts contain details, such as the number of required cores, RAM, and minimum hard disk space.

Dynamic Resource Load Check

Pulse displays alerts that the cluster is in the maintenance state when:

- The usage of the CPU, RAM, or both exceeds 90% for a continuous 10 minutes

  Note: When the CPU and RAM meet the required load, the Pulse server automatically recovers and reverts to the running state.

- The free space on the hard disk drive is less than 20 GB

  Note: After the hard disk drive is cleaned up and more than 20 GB space is available, the administrator needs to restart the Pulse server manually.

In addition, Pulse displays warnings when:

- The usage of the CPU, RAM, or both exceeds 80% for a continuous 10 minutes
- When the free space on the hard disk drive is less than 100 GB

Operating System and System Type Check

If the Pulse cluster is deployed on the Windows operating system (OS) or on a virtual machine (VM), an alert is displayed with Linux as the recommended operating system for a Pulse cluster.

If a Pulse service is unresponsive, the Pulse cluster automatically restarts the service up to three times. If the cluster becomes unresponsive again, you need to manually restart the Pulse cluster. After a manual restart, the auto-recovery counter is reset. It is recommended that you monitor auto-restarts, analyze, and take corrective action to fix the root cause of such service problems.

To configure the Pulse nodes, complete the following tasks using the Pulse Service Manager web page, which can be accessed through a web browser:

- Installing Pulse Primary Node as a Service
- Configuring Use of FQDN for Pulse Access URL
- Modifying Java Memory for Pulse Primary Node
- Modifying Disk Quota for Pulse Nodes

- Specifying Purge Settings for Disk Management on Pulse Primary Node
- Specifying Email Server Settings for Pulse Primary Node
- Specifying Security Settings for Pulse Primary Node
- Defining Library Management Settings for Pulse Primary Node
- Shutting Down and Restarting Pulse Primary Node

Related Topic

Accessing Pulse Service Manager Web Page

Installing Pulse Primary Node as a Service

When you install and run Pulse as a service, Pulse Service Manager starts up automatically when you log in to the operating system (OS). Just as in the non-service mode, Pulse Service Manager installed as a service can be managed through the Pulse Service Manager web page.

For help on Pulse Service Manager as a service and a list of all the possible arguments, navigate to `<Cadence installation directory>/server/bin` and use the `vista -h` command in a Command Prompt window.

Permissions

- To install, remove, start, or stop the Pulse service on Linux, you must run the service-related commands as a root user. Use the `sudo` utility.
- To install, remove, start, or stop the Pulse service on Windows, you need administrative rights on your machine.

Linux

You can install, start, stop, or remove the Pulse primary node as a service on Linux using the `vista` command.

Important

Running the `vista` command on Linux causes the `/etc/adw/pulseservicemanager.sh` content to be re-initialized. If you want to avoid this, particularly if you have custom content in this file, use the native operating system

commands to start or stop the `pulseservicemanager` service. For example, on RHEL, the OS command you can use is `systemctl start/stop pulseservicemanager`.

To install, start, stop, or remove the Pulse server as a service on Linux, do the following:

1. Open an elevated command prompt.
2. Navigate to *<Cadence installation_directory>*`/server/bin`.
3. Use the following commands:

`Install`	**1.** Install the service: `sudo ./vista installservice -servicehome` *<value>* where ❑ `servicehome` is a mandatory argument and is for the location where you want the service configuration details to be stored. Because a running service might not have write permission for the user home, the 'service home' is used. *<value>* is the path to the location where you want the service configuration details to be stored. Spaces in the path are not supported. When you install Pulse as a service on a Linux machine, the installation script creates a Linux user and a group, both of which are named `pulse`. Because Pulse as a service is installed by a user named `pulse` on Linux, switch users to install the Pulse component database.
`Start`	`sudo ./vista startservice` The default port for Pulse Service Manager is 7100. If you launch more than one instance of Pulse Service Manager and 7100 is not available because it is in use, Pulse logs an error in the `adwserver.out` file in *<servicehome>*`/Pulse/vista_pulse/server/log`.
`Stop`	`sudo ./vista stopservice`

Remove	`sudo ./vista removeservice`

Windows

To install, start, stop, or remove the Pulse primary node as a service on Windows, do the following:

1. Open an elevated command prompt.
2. Navigate to `<Cadence installation_directory>/server/bin`.

   Note: You cannot install Pulse as a service from a UNC path or mapped drive.
3. Use the following commands:

Install	**1.** Install the service using the following command: `vista installservice -servicehome <value>` where ❑ `servicehome` is a mandatory argument and is for the location where you want the service configuration details to be stored. Because a running service might not have write permission for the user home, the 'service home' is used. ❑ `<value>` is the path to the location where you want the service configuration details to be stored. Spaces in the path are not supported. For example, the path can be: `C:\Pulse\service`. By default, the Pulse Service Manager service is created with the name `PulseServiceManager` and its details are stored in a folder called `vista_pulse` in the specified directory. The `vista_pulse` folder is created when you start the Pulse Service Manager service.
Start	`vista startservice`
Stop	`vista stopservice`

Remove	`vista removeservice`

Accessing Pulse Service Manager Web Page

The Pulse Service Manager configuration options are available in the Pulse Service Manager web page.

Supported Web Browsers

- Linux - Firefox; you must ensure that you define it as the default web browser.

  If you do not define it as the default web browser, open Pulse Service Manager using the following URL: `http://<Pulse access URL>:7100/element`.

- Windows - Google Chrome and Firefox

Starting Pulse Service Manager

To access the Pulse Service Manager web page, Pulse Service Manager must be running.

If you installed Pulse Service Manager as a service and did not reboot after installing, do the following:

- For Linux, start the Pulse service using `sudo ./vista startservice`.
- For Windows, start the Pulse service using `vista startservice`.

If you did not install Pulse as a service, do the following:

- On Linux:
  1. Open a terminal.
  2. Navigate to `<Cadence installation directory>/server/bin`.
  3. Type `vista`.
- On Windows:
  1. Navigate to `<Cadence installation directory>/server/bin`.
  2. Double-click `vista.bat`.

Opening Pulse Service Manager Web Page

1. Do one of the following:

   - If you installed Pulse Service Manager as a service and did not reboot after installing, access the web page by typing `http://<Pulse access URL>:7100/element` in a web browser.
   - If you started Pulse Service Manager by navigating to `<Cadence installation directory>/server/bin`, a web page opens automatically.

     The Pulse login dialog is displayed.

2. Specify the default administration credentials to log in to Pulse.

   The default credentials are user name = `admin` and password = `admin`. You can modify the password in the *Security* tab of Pulse Service Manager after you log in.

   Note: The user name cannot be modified.

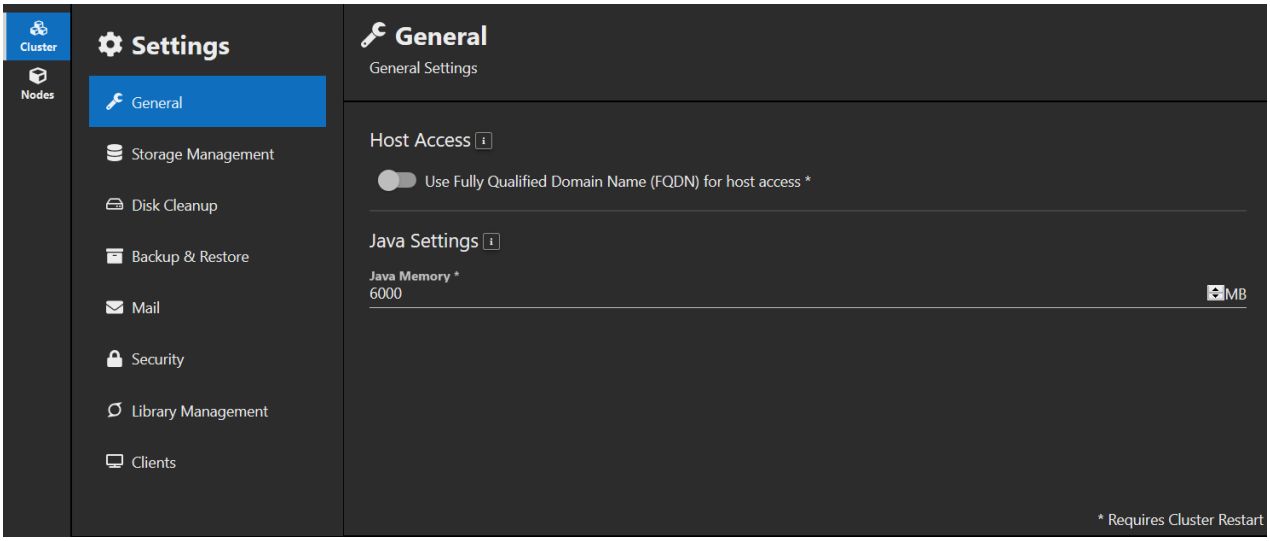
   Pulse Service Manager starts.

Configuring Use of FQDN for Pulse Access URL

To configure the access URL to use the fully qualified domain name (FQDN) for Pulse instead of the hostname, do the following:

1. Start Pulse Service Manager to access the web page with all the configuration options.
2. Click the gear *Settings* icon on the top right of the page to access the Settings dialog.

The Settings page is displayed.



3. In *General*, toggle on the fully qualified domain name (FQDN) button if required by the IT policy of your company, SSL authentication mechanisms, or network topology.

   For example, some network topologies can only resolve a DNS entry with the FQDN. A sample FQDN is as follows: `<hostname>.global.cadence.com`.

   Note: If you choose FQDN and designers specify the remote URL as `<hostname>:7100`, they cannot connect to the Pulse server.

Related Topics

Accessing Pulse Service Manager Web Page

Modifying Java Memory for Pulse Primary Node

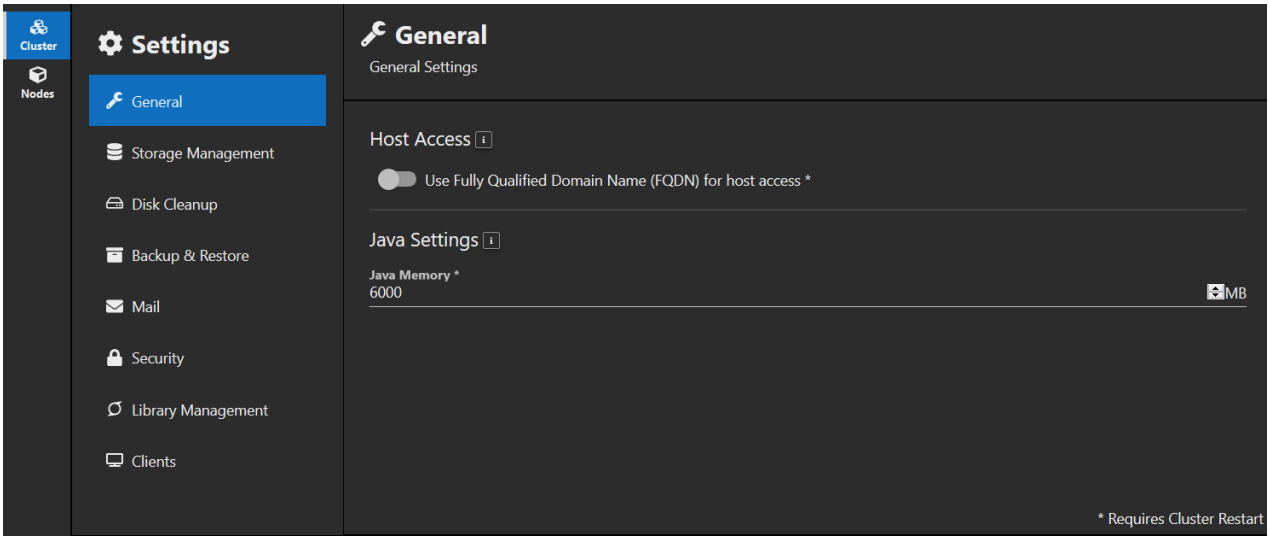
The Java memory setting provides the maximum amount of memory that the Pulse server is allowed to use from the system memory. It is recommended that you start with the default memory allocation.

If you encounter performance-related issues or receive an email alert from Pulse indicating that the server restarted because of out-of-memory problems, then consider an increase in memory. Typically, increases can be made to either 9000 or 12000 depending on the size of the library and the number of users accessing Pulse.

To modify the Java memory for the Pulse primary node, do the following:

1. Start Pulse Service Manager to access the web page with all the configuration options.
2. Click the gear *Settings* icon on the top right of the page to access the Settings dialog.

   The Settings page is displayed.



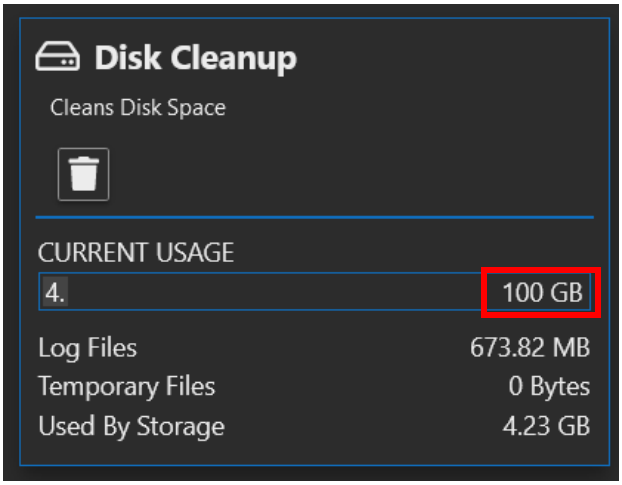
3. In *General*, modify the Java memory, if needed.

Related Topic

Accessing Pulse Service Manager Web Page

Modifying Disk Quota for Pulse Nodes

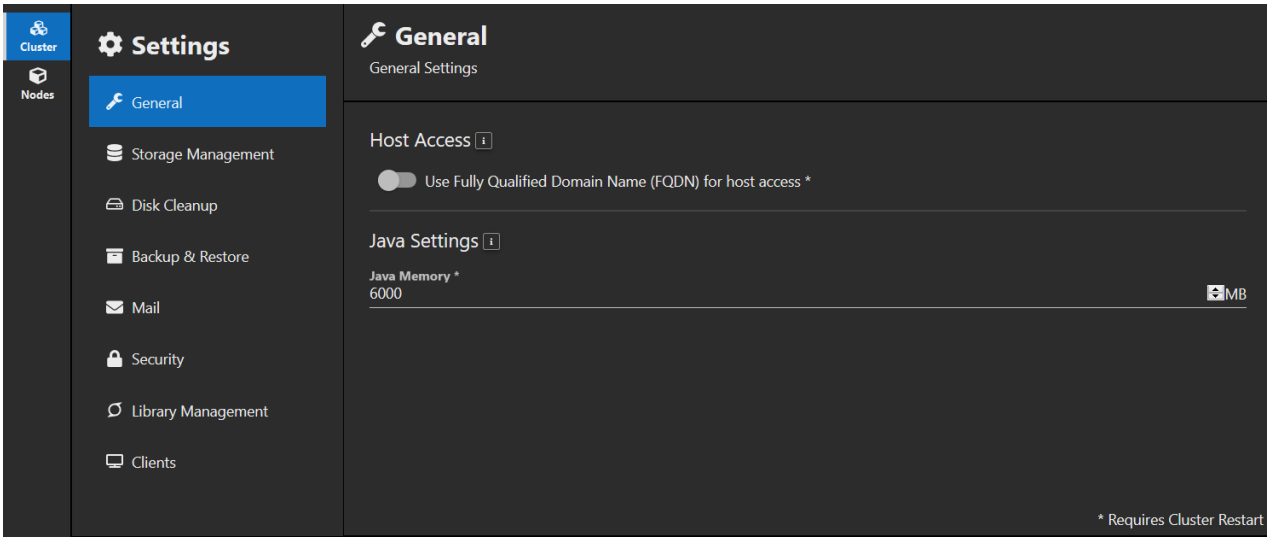
If you want to change the upper limit of the Pulse node consumption display on the Pulse Service Manager and Pulse Manager web pages, modify the disk quota. This helps you check whether the disk quota is full. If full, you can purge the Pulse data that is no longer needed.



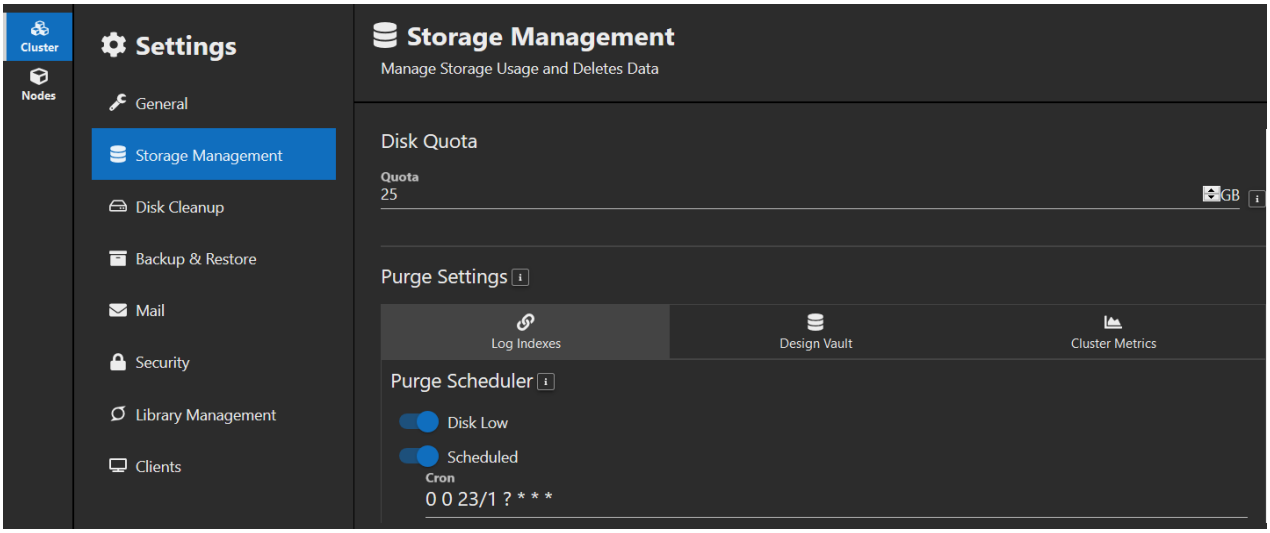
To modify the disk quota for the Pulse primary node, do the following:

1. Start Pulse Service Manager to access the web page with all the configuration options.
2. Click the gear *Settings* icon on the top right of the page to access the Settings dialog.

   The Settings page is displayed.



3. Select *Storage Management* in the left pane.



4. Modify the disk quota and use it as a reference for the usage monitor to keep an eye on disk usage growth.

Related Topic

Accessing Pulse Service Manager Web Page

Specifying Purge Settings for Disk Management on Pulse Primary Node

Specify purge settings to delete unwanted data from the database and remove deleted archives from the vault. This task is performed only when the Pulse server is running.

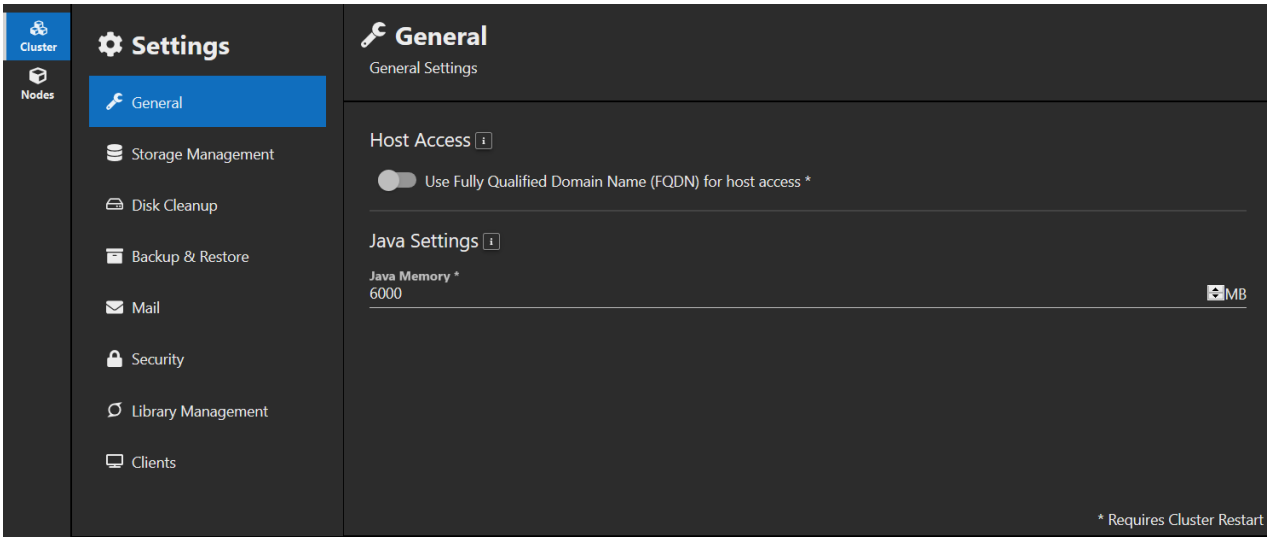
The extent of data deleted depends on the purge retention policy values. By default, Pulse deletes log index data older than 90 days from the purge date. However, if you define the purge policy as 45 days, Pulse deletes log indexes that are older than 45 days whenever purge runs.

The purge operation is scheduled as a Cron job or on detection of Disk Low.

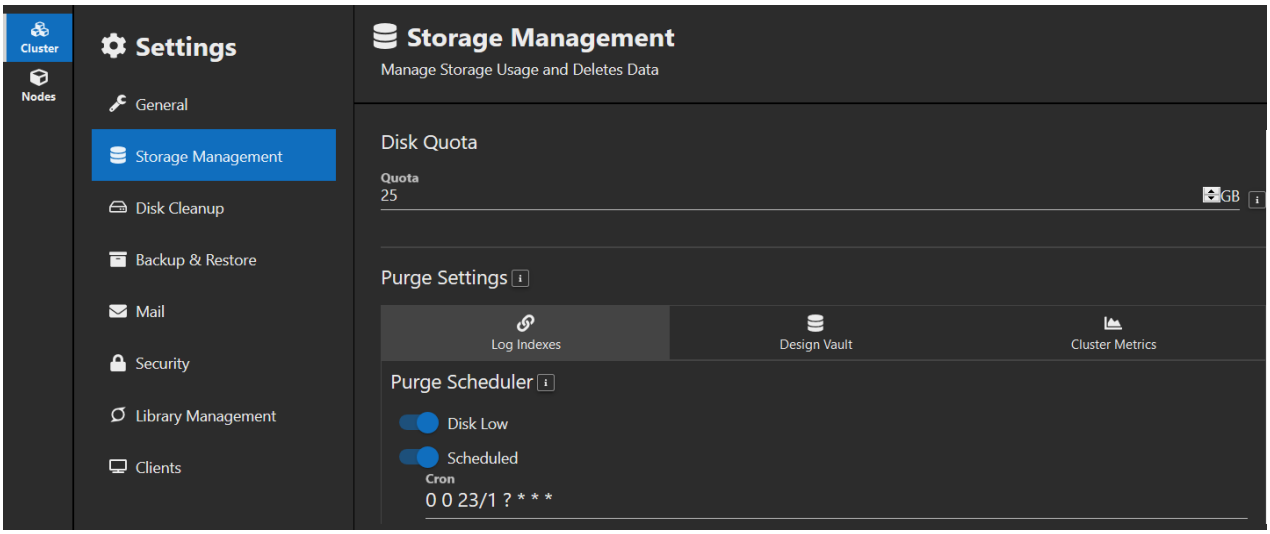
To specify the purge settings for the Pulse primary node, do the following:

1. Start Pulse Service Manager to access the web page with all the configuration options.
2. Click the gear *Settings* icon on the top right of the page to access the Settings dialog.

The Settings page is displayed.



3. Select *Storage Management* in the left pane.



4. To improve system performance, you can specify purge settings for three types of data:

- ❑ Log indexes

  By default, Pulse deletes 90 days of data from the date of purging.

- ❑ Design vault data

  This deletes design version branch files—PDF and `.sdax`—for one year prior to the data of purging.

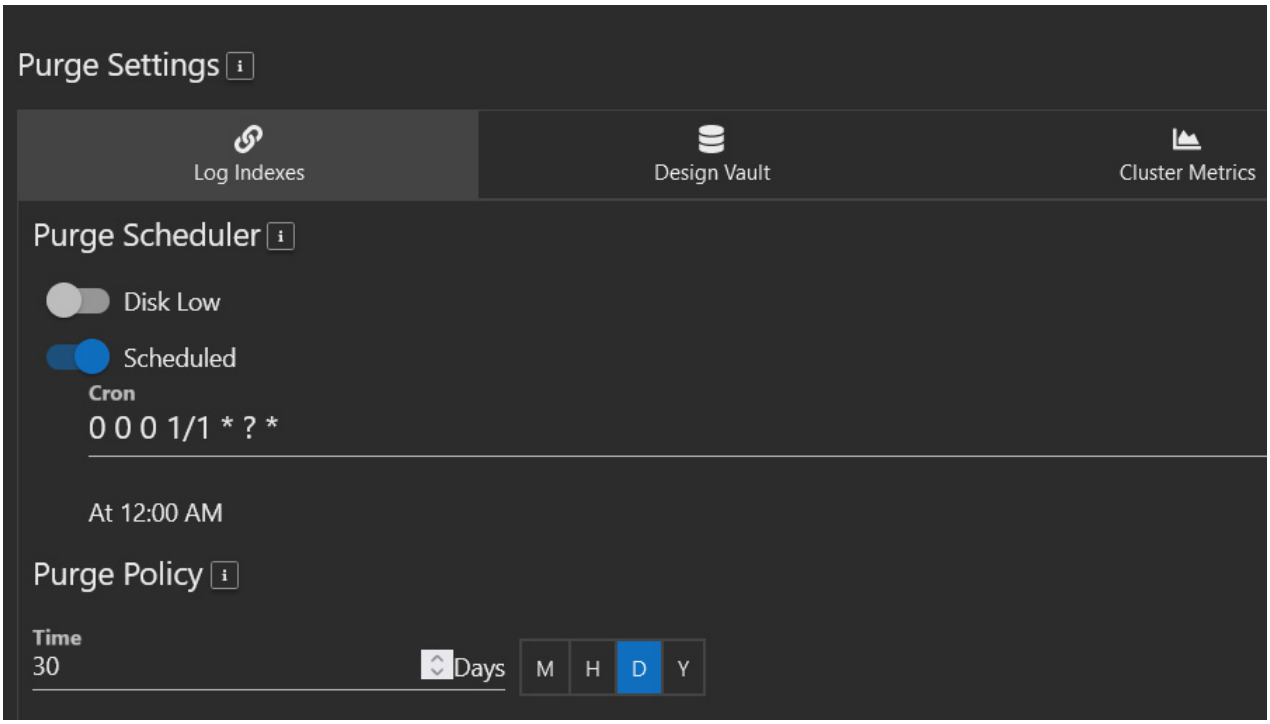
- ❑ Cluster metrics

  By default, Pulse deletes metric data older than 10 days.

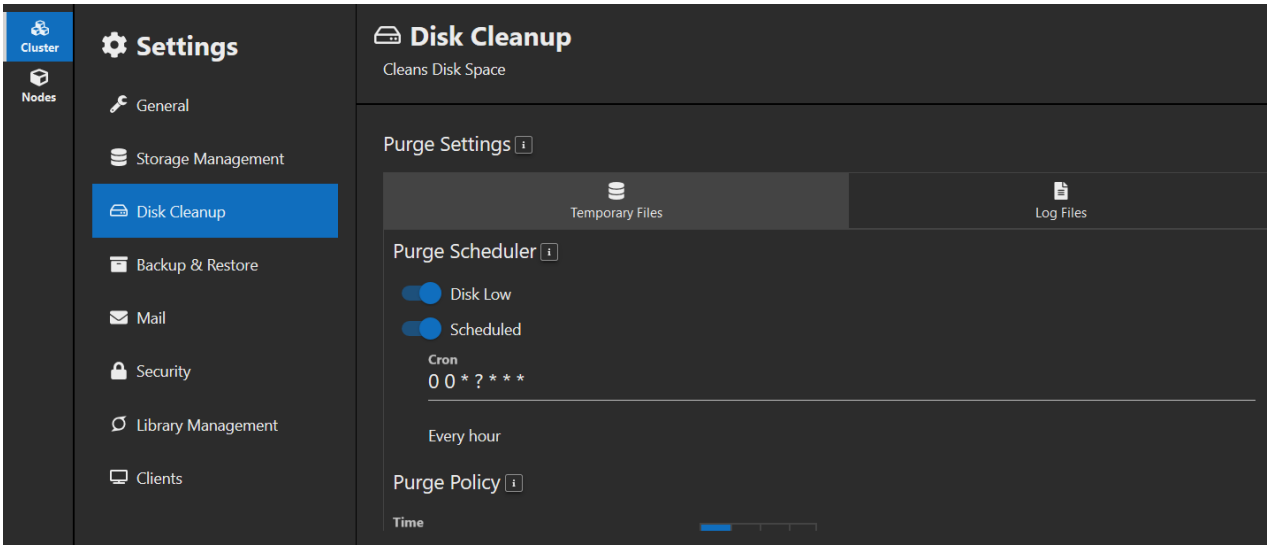
5. Select one or both of the following options:

   - ❑ *Disk Low* - Pulse purges data if the disk space falls below the default or specified disk quota.
   - ❑ *Scheduled* - Pulse purges data based on the defined Cron job. The Pulse Cron scheduler is based on the Quartz Job Scheduler. As a result, the Pulse Cron scheduler supports most of what has been described here: *Cron Trigger Tutorial*.

6. In *Purge Policy*, define the age of the data to be purged.



7. If you want to purge temporary or log system files, select *Disk Cleanup* in the left pane.



8. Specify the purge schedule and the age of the data to be purged for temporary or log files.

Related Topic

Accessing Pulse Service Manager Web Page

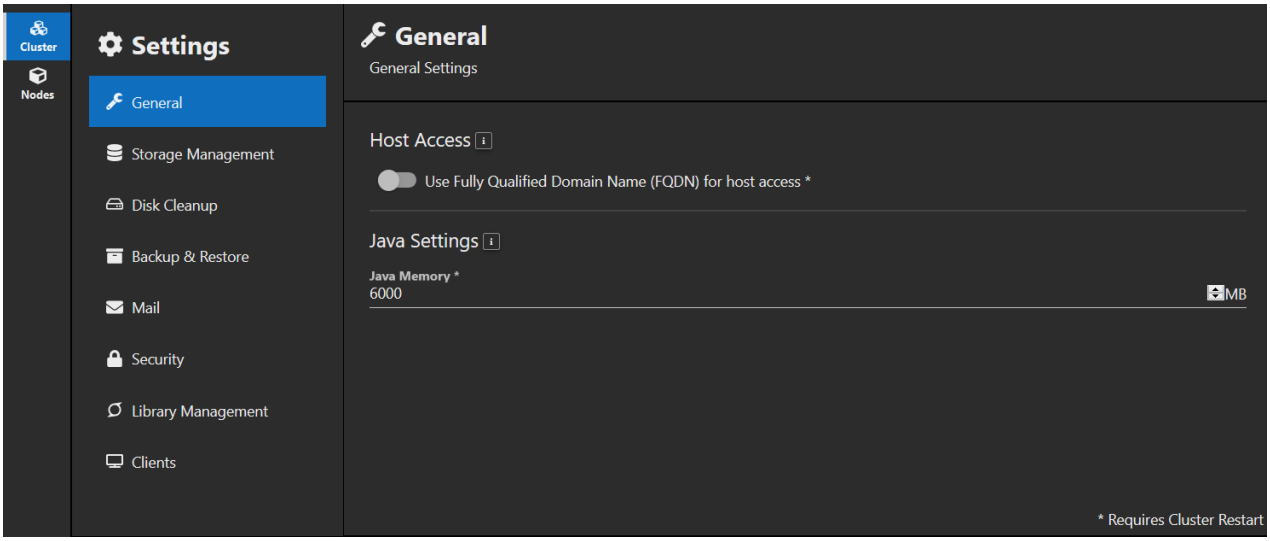
Specifying Email Server Settings for Pulse Primary Node

Configure email server settings to generate and send email notifications, which flag a variety of events to administrators or key stakeholders using an email alias.

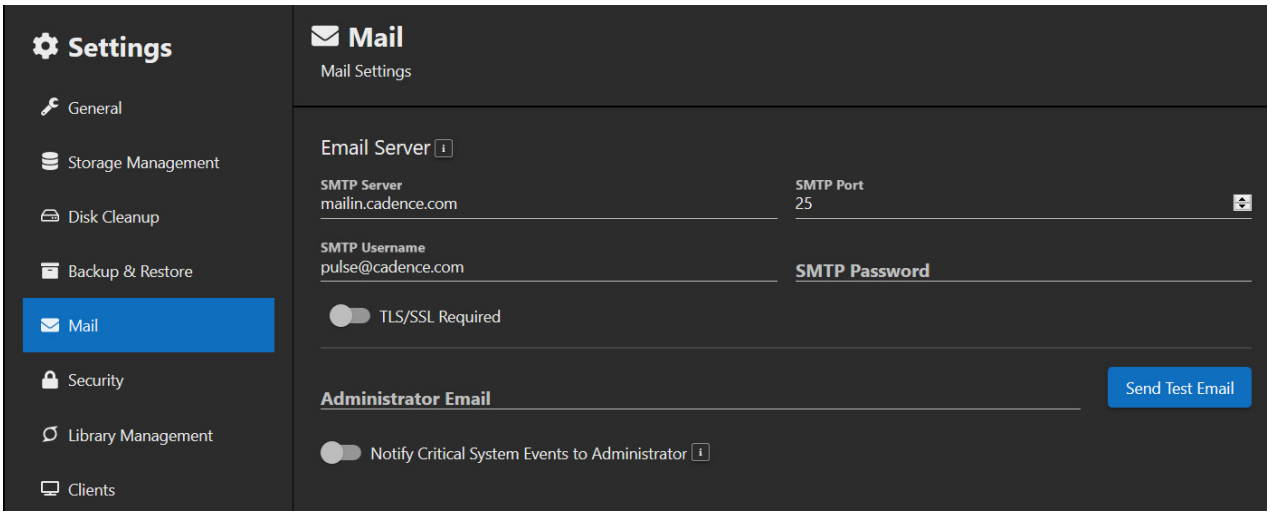
To specify the email settings for the Pulse primary node, do the following:

1. Start Pulse Service Manager to access the web page with all the configuration options.
2. Click the gear *Settings* icon on the top right of the page to access the Settings dialog.

The Settings page is displayed.



3. Select *Mail* in the left pane.



4. Specify the required settings.

Important

Because the *SMTP Username* is used for all email notifications, it is a mandatory field whether or not authentication is enabled on the SMTP server.

The *SMTP Password* is not mandatory.

If you toggle on the *Notify Critical System Events to Administrator* button, you are notified by email whenever the Pulse primary node goes into the maintenance mode due to a scheduled system maintenance task or because of an error.

Related Topic

Accessing Pulse Service Manager Web Page

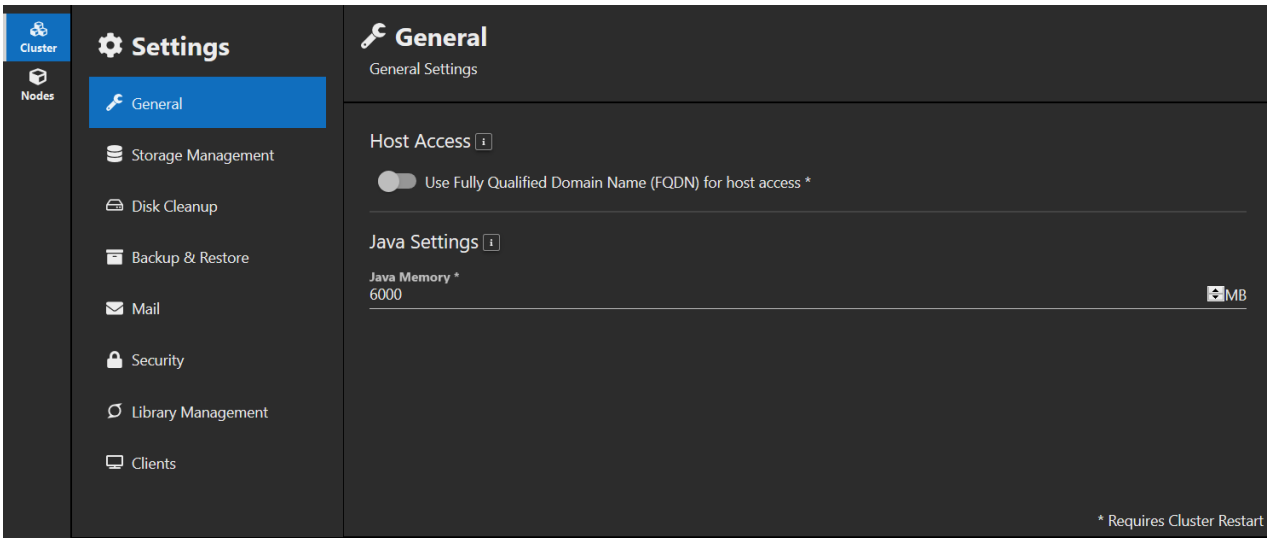
Specifying Security Settings for Pulse Primary Node

If you want to ensure encrypted communication between the Pulse primary and data nodes or client machines, you must first create the SSL certificate keystore—`edm.jks`–before toggling on the SSL button. See the related topics for details.

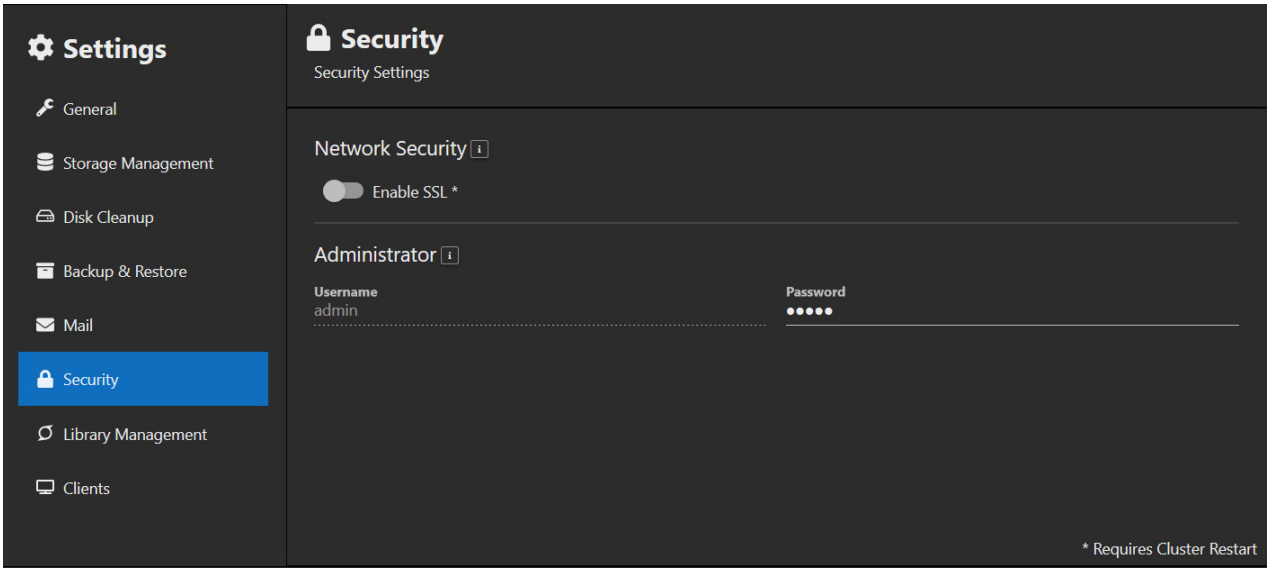
To protect the data stored on the server, specify security settings for the Pulse primary node by doing the following:

1. Start Pulse Service Manager to access the web page with all the configuration options.
2. Click the gear *Settings* icon on the top right of the page to access the Settings dialog.

   The Settings page is displayed.



3. Select *Security* in the left pane.



4. Toggle on the *Enable SSL* button to ensure encrypted communication between the Pulse primary and data nodes or client machines.
5. Specify the user name and password to access the *Pulse User Management* module, which is used to define user roles and permissions for the Pulse platform.

   This password also sets the credential for the Pulse Service Manager web page that has `/element` at the end of the Pulse access URL.

Related Topics

Accessing Pulse Service Manager Web Page

Creating and Using SSL Certificates in Pulse Environment for details on creating the Java KeyStore (JKS)

Accessing Pulse User Management Module

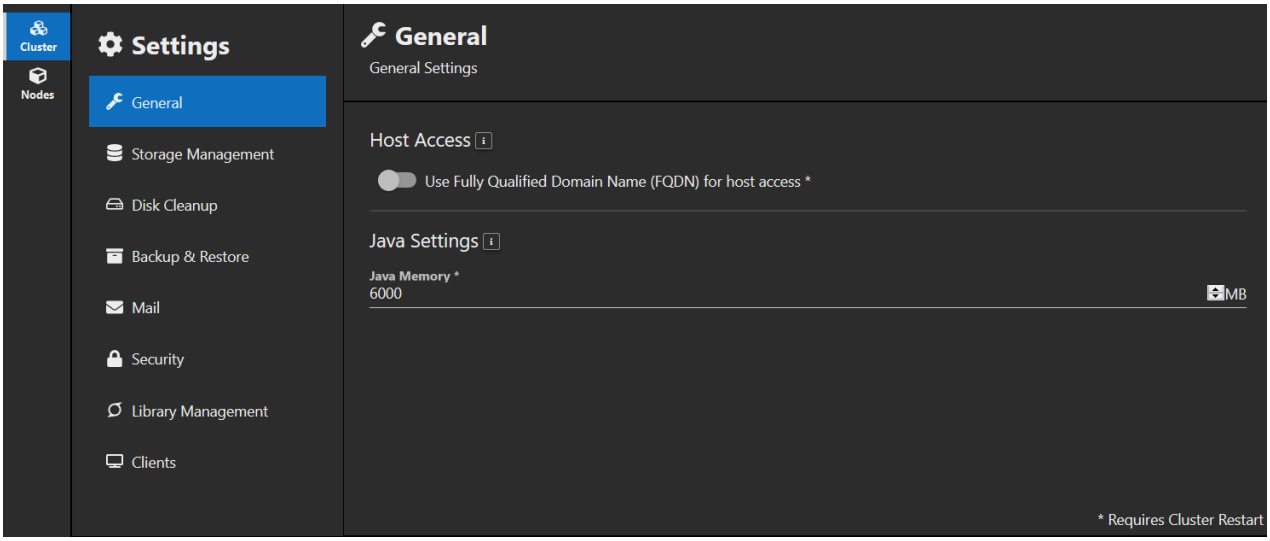
Creating Users for the Pulse Platform

Defining Library Management Settings for Pulse Primary Node

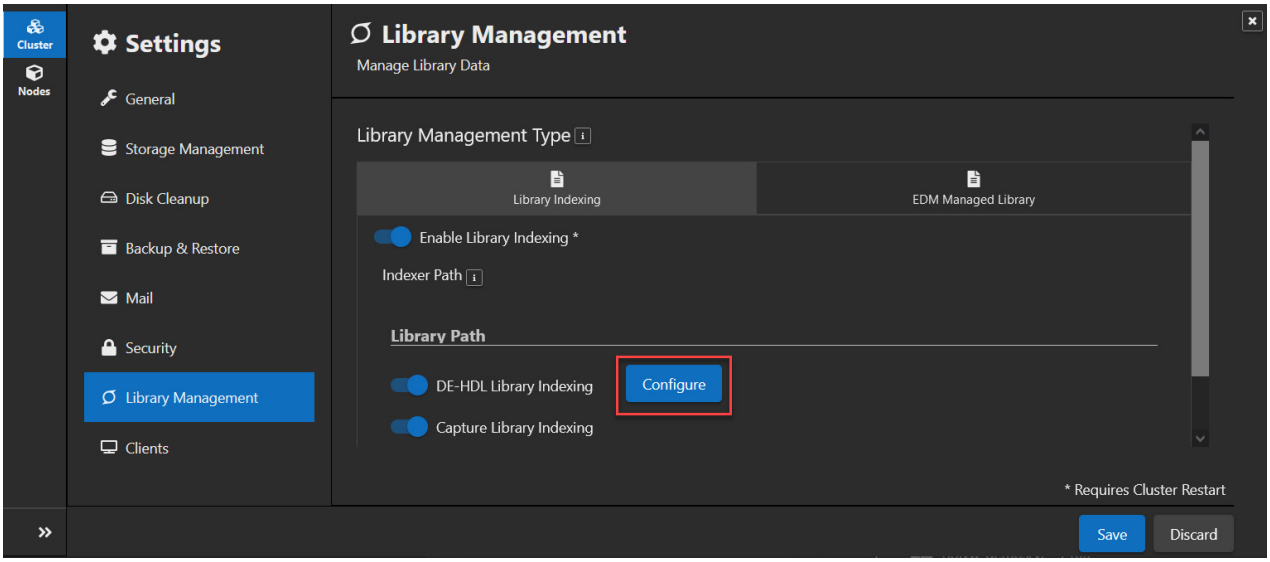
In the *Library Management* pane, define whether you want to work with managed or unmanaged libraries, and other library-related settings. To define the library management settings, do the following:

1. Start Pulse Service Manager to access the web page with all the configuration options.
2. Click the gear *Settings* icon on the top right of the page to access the Settings dialog.

   The Settings page is displayed.



3. Select *Library Management* in the left pane.



4. Select one of the following:

   - ❑ *Enable Library Indexing* to work with unmanaged libraries, which can be Design Entry HDL or OrCAD Capture libraries
   - ❑ *EDM Managed Library* to work with managed libraries

Configuring Unmanaged Libraries

If you select *Enable Library Indexing*, do the following:

1. In *Library Path*:

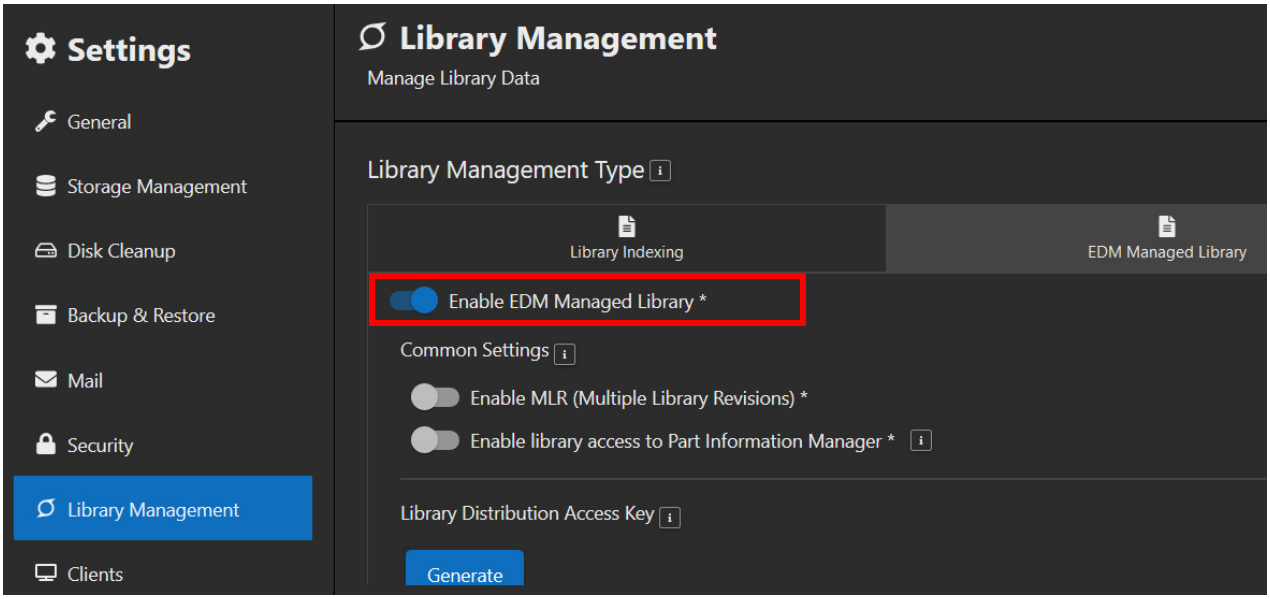
   - ❑ To index DE-HDL libraries, specify the `CDS_SITE` path, which has `cds.lib`
   - ❑ To index OrCAD Capture libraries, specify the `CDS_SITE` path, which has `Capture.ini`

     Ensure that `CDS_SITE` has:

     - ❍ `$CDS_SITE/cdssetup/OrCAD_Capture/23.1.0/Capture.ini`
     - ❍ `$CDS_SITE/cdssetup/projmgr/site.cpm`

Important

If you work with unmanaged libraries, remember to toggle off the *Enable EDM Managed Library* button.



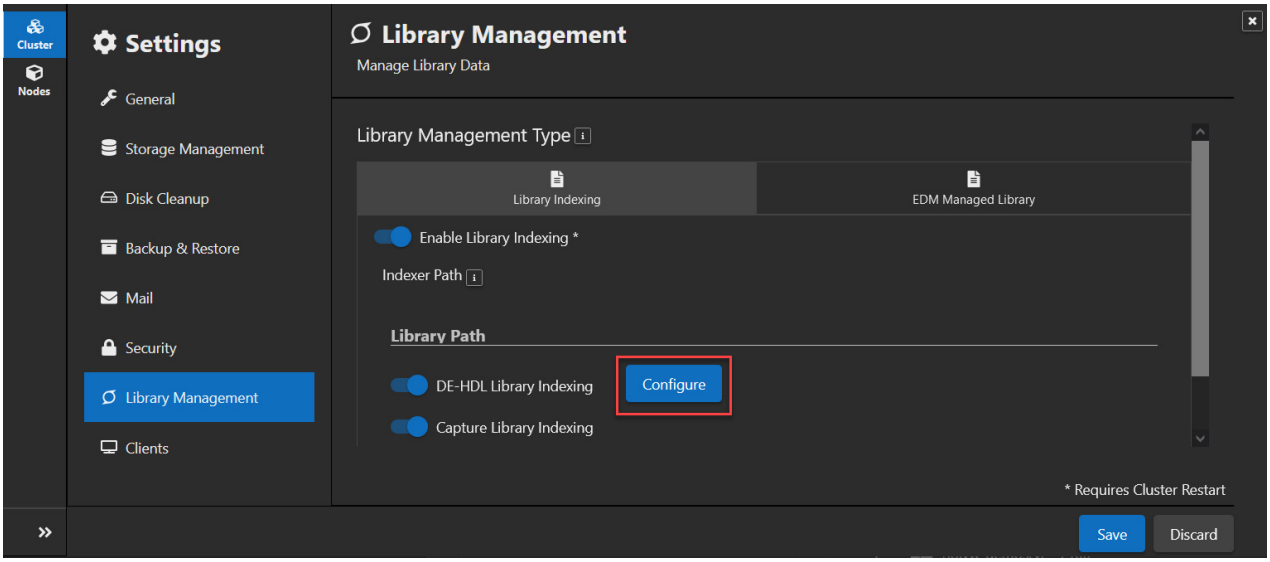
2. Toggle on the *DE-HDL Library Indexing* and or *Capture Library Indexing* buttons to index DE-HDL and or OrCAD Capture libraries.

   If `CDS_SITE` has both kinds of libraries, DE-HDL and OrCAD Capture, you can index both.

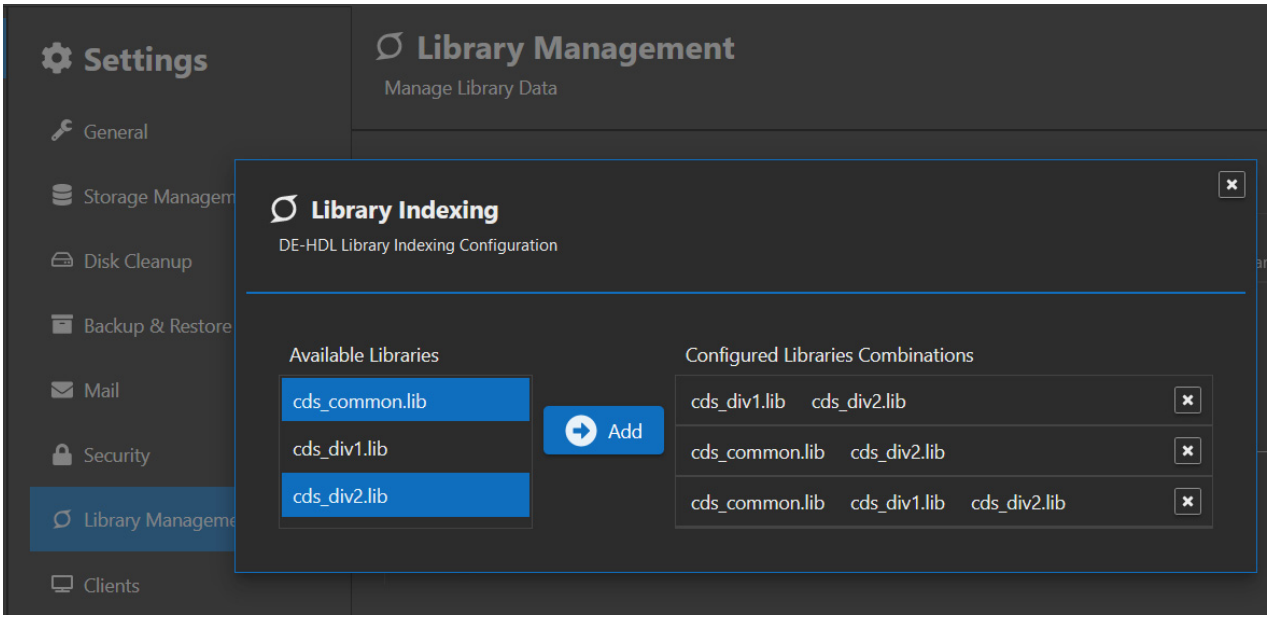
3. If you want to index multiple DE-HDL unmanaged libraries together, click *Configure* next to *DE-HDL Library Indexing*.

   Multi-library indexing allows you to segment your libraries to make only certain content available to certain projects. For example, an automotive design might only use automotive parts. If automotive parts are segmented in the library structure, multi-library indexing can help with this.

Note: Multi-library indexing is not supported for OrCAD Capture libraries.



The Library Indexing dialog box is displayed.



The *Available Libraries* box lists the `.lib` files available in the *Library Path* you specified.

4. Select the `.lib` files to be indexed and click *Add*.

   You can select multiple files together and non-contiguous files using Ctrl + click.

5. Toggle on the *Capture Library Indexing* button for OrCAD Capture libraries to be indexed.

   If you toggle this on, ensure that the sub-directory that contains `Capture.ini` is under the parent directory specified in *Library Path*.

   It is recommended that you run OrCAD Capture on the Pulse server, configure the libraries and ODBC connection to the CIS database, and specify the resulting `Capture.ini` for the Pulse server to use.

   Note: If you installed Pulse as a service, the ODBC connection must be configured under *SYSTEM* in the ODBC configuration user interface.

OrCAD library indexing is only supported on Windows.

Pulse automatically indexes all the configured libraries, and uses the library location path to access the library for parts. This secured and indexed library data can be used across divisions and sites.

If you are an existing user and you switch from managed to unmanaged libraries or vice versa, you must restart the Pulse cluster. Any open client applications, such as System Capture, must also be restarted by designers.

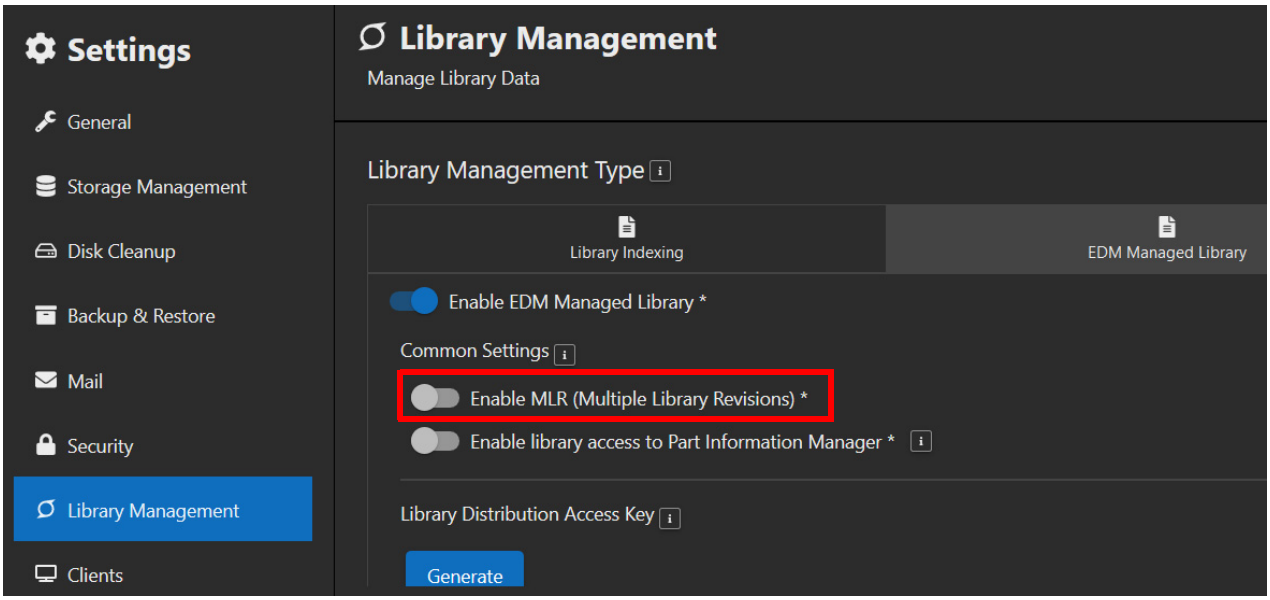
When you switch from:

- managed to unmanaged libraries, Unified Search continues to display parts from managed libraries until System Capture is restarted.
- unmanaged to managed libraries, Unified Search displays an error when designers try and view part details. Exiting and launching System Capture again addresses the error.

Configuring EDM-Managed Libraries

If you select EDM Managed Library, do the following:

1. Toggle on the *Enable MLR (Multiple Library Revision)* button if you are an existing customer who is still working with a lower release of an EDM-managed library but want to deploy a 22.1 Pulse primary node.



If you toggle on the *Enable MLR (Multiple Library Revision)* button, a variable called `MLR_PCBDW_LIB` is automatically added to the following file: `<PCBDW_LIB>\distribution\env\fetch_dump.ini`.

Important

Because read and write permissions might impact the addition of this variable to this file, verify that the variable is updated in `fetch_dump.ini`. Also check that the `fetch_dump` directive is `on` in `lib_dist.ini`.

After you complete the configuration of the Pulse primary node, and Pulse data nodes if you are using them, configure *<Allegro EDM Conf Root>*. During *<Allegro EDM Conf Root>* configuration, specify the value of the `MLR_PCBDW_LIB` variable in the *<startworkbench>* script as:

`MLR_PCBDW_LIB=<path to the PCBDW_LIB of the primary Pulse server>`

This is required so that the Pulse data nodes read and get component data from the 22.1 library server.

2. Toggle on the *Enable library access to Part Information Manager* button if you have designers in your organization still working with Design Entry HDL.

Design Entry HDL uses Part Information Manager, a component search utility.

3. Generate an access key, which is needed for library distribution to run successfully on a data node.

   It is recommended that you embed this key in the library distribution script on a server that runs library distribution as a scheduled task or Cron job.

   Set the library distribution access key by specifying `PULSE_LIBDIST_TOKEN` as a variable on Windows and Linux systems that run `lib_dist` or `lib_dist_client`.

If you revoke and generate a new access key, remember to provide it again to the users who manage data nodes running library distribution.

4. Click the *Export* button if you have configured LDAP as the identify provider in the *Pulse User Management* module.

   Clicking this button generates a file with LDAP authentication settings for the library management tools, such as Database Editor, Database Administrator, and so on.

   Pulse stores the LDAP settings in a `.conf` file in `<Pulse primary node home>/Pulse/vista_pulse/server/conf`.

   If you are working with an older library server, such as a 17.4-2019 server, and you had selected LDAP when configuring it, you must modify the Pulse-generated LDAP file and add the previous LDAP settings, such as:

   - Attribute that corresponds to the corporate user ID; for example, *uid*.
   - Distinguished Name (DN) of the directory that contains the list of users for an LDAP search. For example: *ou=people,o=cadence.com*

Related Topics

Accessing Pulse Service Manager Web Page

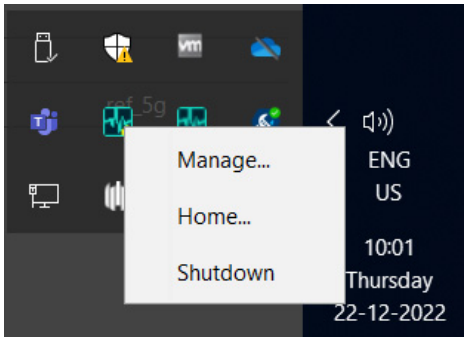
Configuring Multi-Library Release Environment

Configuring LDAP Sync with the Pulse Primary Node

Shutting Down and Restarting Pulse Primary Node

➡ To shut down Pulse, do one of the following:

- Click *SHUTDOWN* in Pulse Service Manager, the web page that provides access to the configuration and other system settings of the Pulse server.
- Click the Pulse icon ( ) in the taskbar notification area and click *Shutdown*.



- If you installed Pulse as a service, stop the Pulse server as a service using the Vista commands or the native operating system service control commands.

  To use the Vista commands, do the following:

  a. Open an elevated command prompt or a terminal window.

  b. Navigate to `<Cadence installation directory>/server/bin`.

  c. Use the following commands:

  - `vista stopservice` on Windows.
  - `sudo ./vista stopservice` on Linux.

Important

Closing the web browser being used to view the Pulse Manager web page does not shut down Pulse.

If you work with a cluster server setup and a Pulse data node is still starting up when you restart the Pulse primary node, the data node might exhibit unusual behavior. In such cases, it is recommended that you restart the relevant Pulse data node.

Related Topic

Accessing Pulse Service Manager Web Page

4

Configuration of Pulse for Use by Various Applications

To complete the service configurations for Pulse Service Manager, such as the user management service, do the following tasks:

- User Management in Pulse
  - Accessing Pulse User Management Module
  - Creating Users for the Pulse Platform
  - Roles in the Pulse Platform
  - Assigning Roles to Users on Pulse Platform
  - Configuring LDAP Sync with the Pulse Primary Node
- Configuration of Features Specific to Pulse
  - Configuring Unified Search Globally
  - Configuring Unified Search at Global or Project-Specific Level
  - Configuring Live BOM Headers
  - Resetting Pulse Web Dashboard Changes to Default
  - Creating Project Templates
  - Configuring Workflows
  - Working with Notifications
  - Enabling Optional Allegro EDM-Managed Library Features
  - Working with New Part Requests
  - Enabling Diagnostic Test Case Download by Non-Administrator Users
- Configuration of Publish for Manufacturing

User Management in Pulse

A user is someone who needs access to the Pulse primary node, data nodes, and other Pulse services. To add users to the Pulse platform, edit their properties, add them to groups, or remove them if necessary, you use the *Pulse User Management* module.

Only the `admin` account can manage users. Users defined with any other role, such as Administrator, cannot manage users. However, non-administrator users can change user passwords according to corporate company policies.

The *administrator* user in Pulse is different from the *admin* user.

admin user	Manages the Pulse cluster - has access to Pulse Service Manager and can create or remove users, give users different privileges
administrator user	A user with the `Administrator` role as defined in the *Pulse User Management* module. The `Administrator` is an out-of-the-box user who can create custom workflows, edit workflows, and configure a part request form.

Don’ts for Pulse User Management

The *Pulse User Management* module manages user roles and privileges using Keycloak, an open source identity and access management solution. Although Keycloak allows you to add user roles, modify client scopes, and so on, for the purposes of Pulse administration, **do not do the following**:

- add user roles or modify the following:
  - client scopes
  - realm settings
  - permissions
  - clients
- reset the `admin` account password in the *User Management* console. The password must only be set from the *Settings* page of Pulse Service Manager.

Related Topics

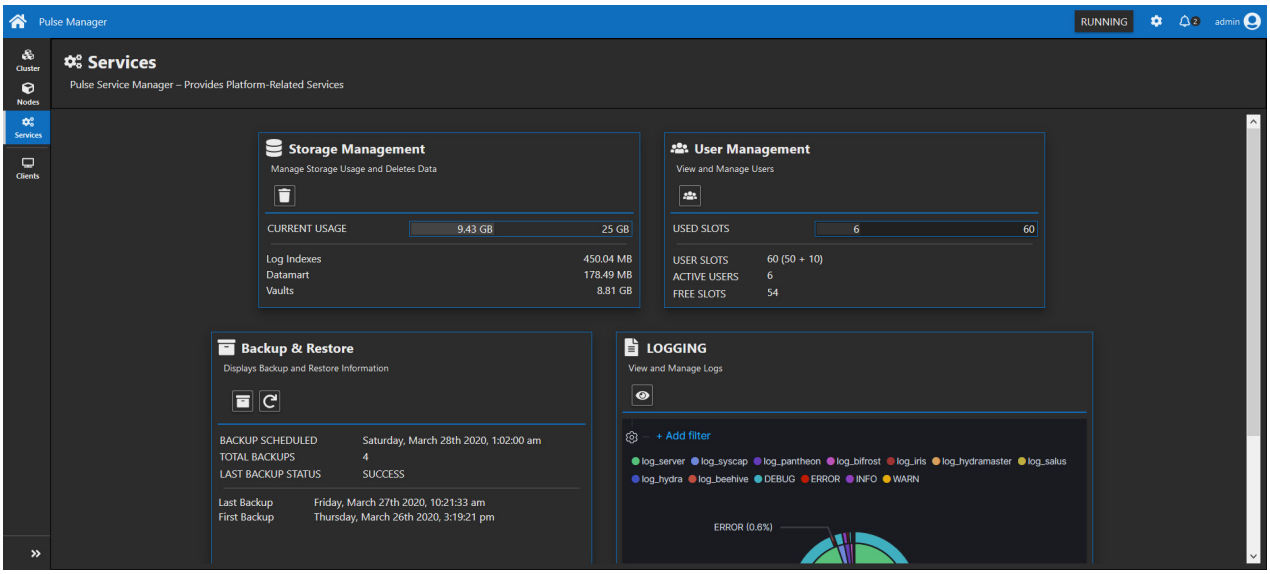
- Roles in the Pulse Platform
- Accessing Pulse User Management Module
- Creating Users for the Pulse Platform
- Assigning Roles to Users on Pulse Platform
- Accessing Pulse Service Manager Web Page

Accessing Pulse User Management Module

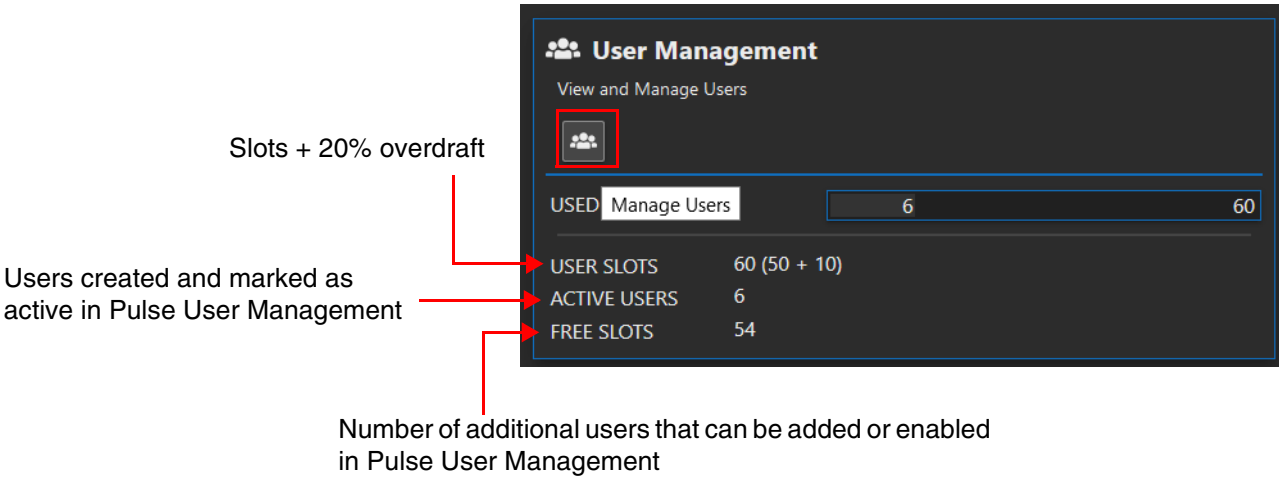
For all functions related to user management in Pulse, you work with the *Pulse User Management* module. Access this module by doing the following:

1. Open the Pulse Service Manager web page to access the system management and configuration options.

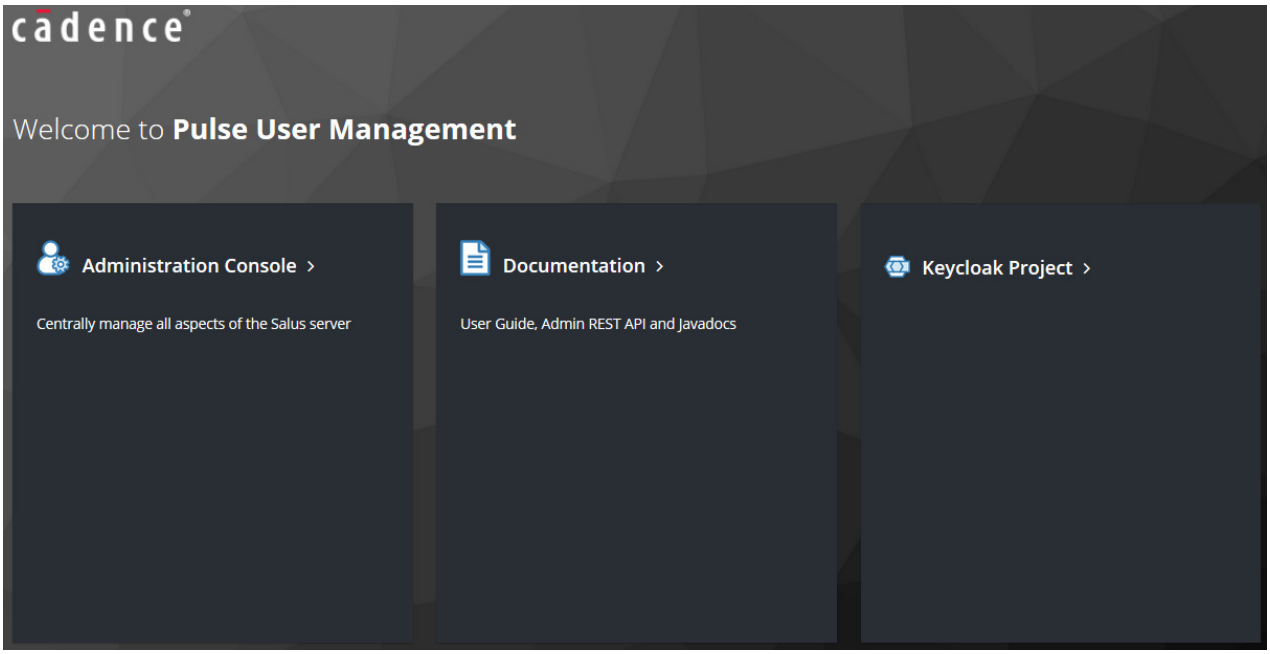
   Pulse Service Manager is displayed.



2. Click the *Manage Users* icon in the User Management tile.



The home page of the *Pulse User Management* module is displayed.

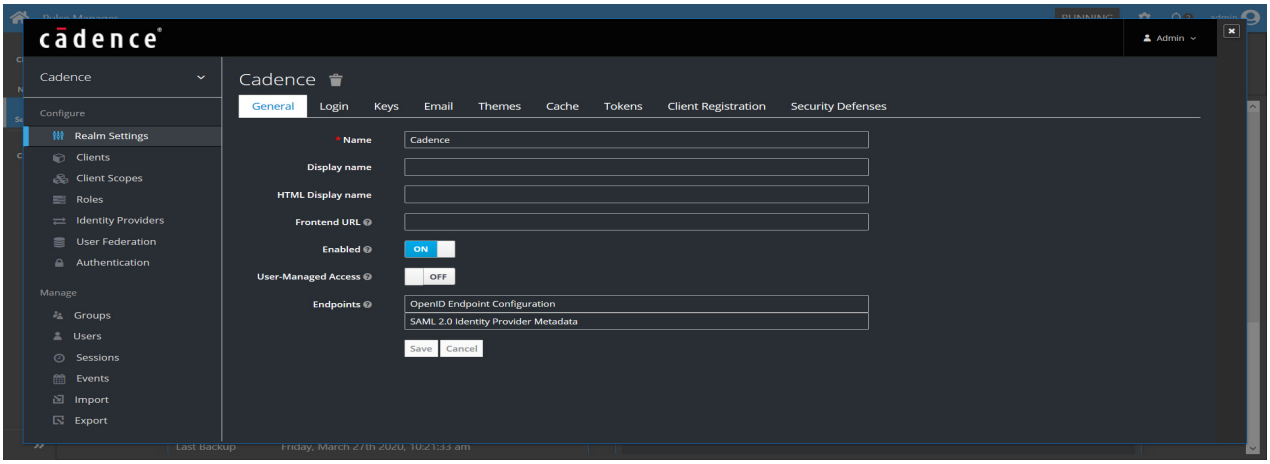


3. Click *Administration Console* to enter the login credentials for the *Pulse User Management* module.

The Login page is displayed. For a first time login, use the default user name and password: *admin* and *admin*.

You can change the administrator password in the *Security* tab of the Pulse Service Manager Settings page. You cannot modify the user name.

The *Pulse User Management* module is displayed.



Related Topics

- Roles in the Pulse Platform
- Assigning Roles to Users on Pulse Platform
- Accessing Pulse Service Manager Web Page

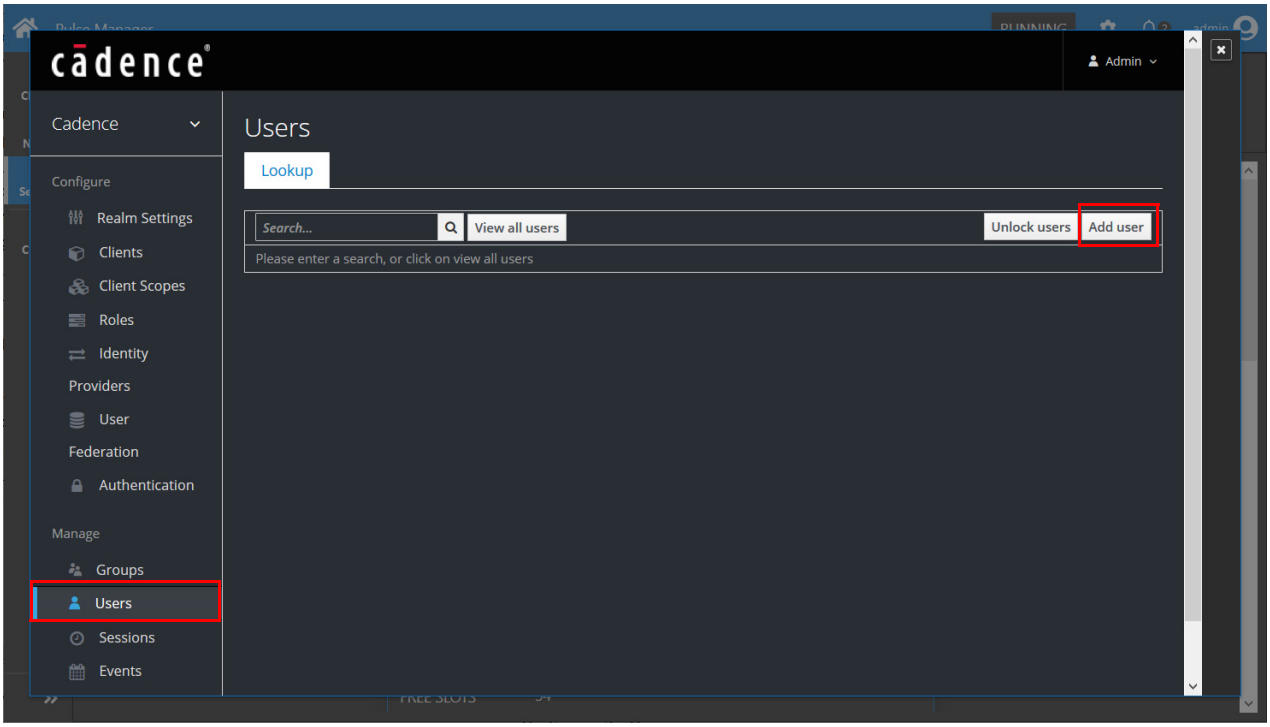
Creating Users for the Pulse Platform

You can create as many users as you have license slots to access and use the Pulse servers.

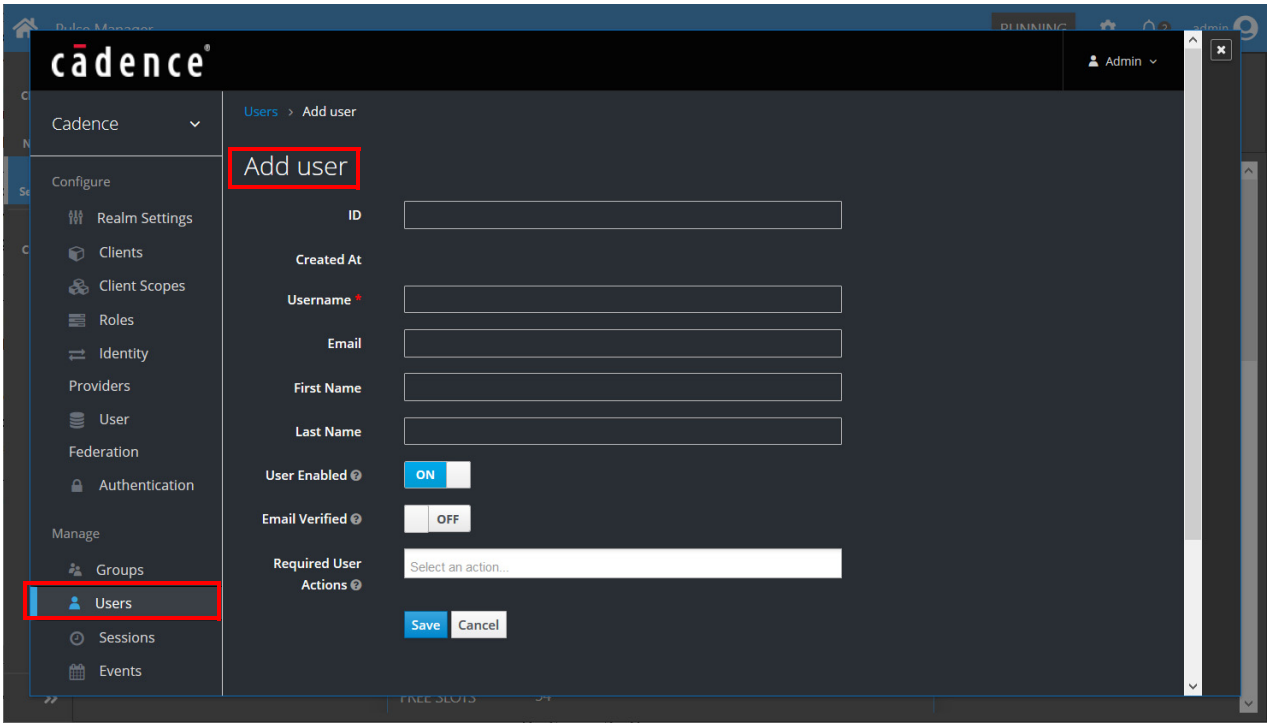
To create and register users for the Pulse platform, do the following:

1. Open the *Pulse User Management* module.

2. Click *Users* under Manage.



3. Click *Add user* on the top right of the page.



4. Specify the user name. It is the only mandatory field.

   When you add users, their login credentials are enabled by default.

   You are prevented from trying to add another user with the same user name as an existing user.

5. Click *Save*.

Related Topics

- User Management in Pulse
- Accessing Pulse User Management Module
- Assigning Roles to Users on Pulse Platform
- Accessing Pulse Service Manager Web Page

Roles in the Pulse Platform

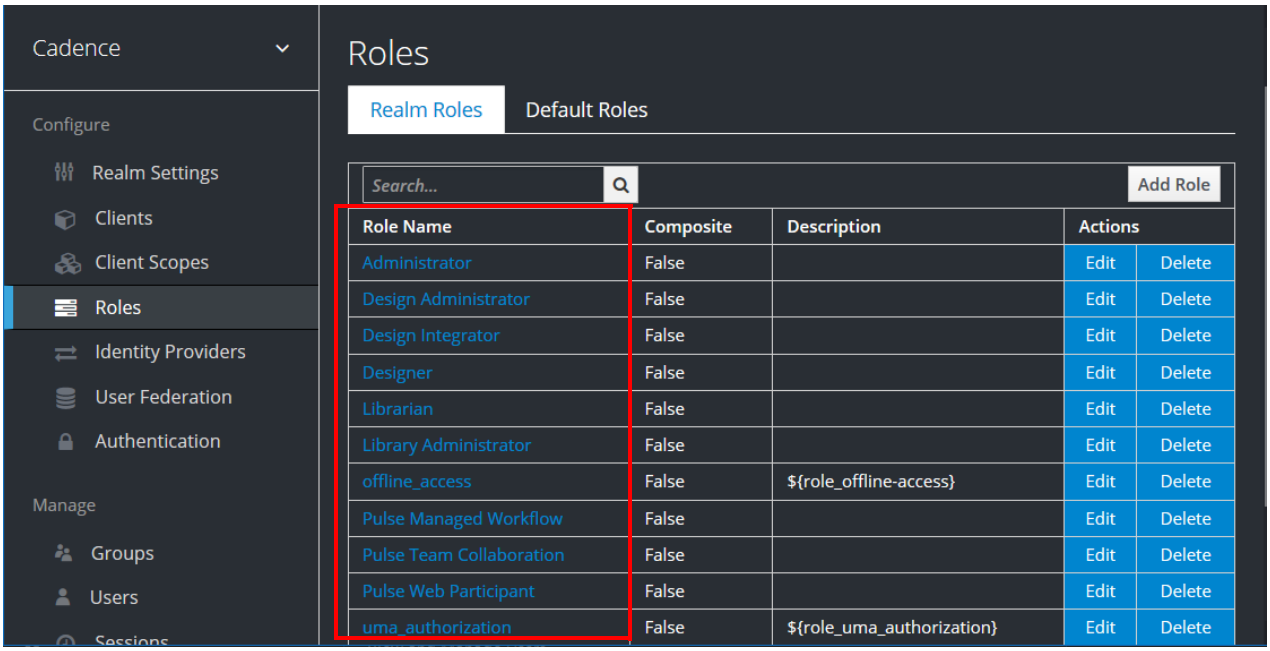
Roles allow you to organize your users based on various aspects, such as the division of the organization in which they are involved, or the design team they are in. Roles also make the sharing of Pulse server content, and the configuration of other Pulse server-served technologies, more streamlined.

Default Roles in Pulse Platform

Pulse provides the following default roles, which you cannot modify.

- Administrator
- Design Administrator
- Design Integrator - not used in this release
- Designer
- Librarian
- Library Administrator
- *offline_access* - in the user interface because it is available out of the box in Keycloak. Not required for Pulse.
- Pulse Managed Workflow
- Pulse Team Collaboration
- Pulse Web Participant

- *uma_authorization* - in the user interface because it is available out of the box in Keycloak. Not required for Pulse.



The following table lists the roles that can access or work with various Pulse features:

Allegro X Pulse Feature and Role Mapping

Feature	Role and Possible Tasks
Pulse Manager (web page)	**Designers**: Change Pulse Home, Set Remote Server
	Administrators of Master/Data Nodes: Set up Master/ Data Nodes, Pulse Cluster Management and Monitoring, Pulse User Management Configuration
Part Requests	**Designers**: Create/Edit/cancel Part Requests, Assign Part Requests, Subscribe/Unsubscribe to Email Notifications, Put on Hold, Free From Hold
	Librarians: Create Parts, Release Parts, Ask for Information, Put on Hold, Free From Hold
	Administrators: Edit and Publish Part Request Form, Put on Hold, Free From Hold

Allegro X Pulse Feature and Role Mapping

Publish for Manufacturing	**Designers**: Open Project, Generate Content, BOM Generation, BOM Sync, Publish Content
	Administrators and Design Administrators: Configure Publish for Manufacturing (Generate Content Utility Creation, Configuring BOM Attributes, Configure Publish Dialog, Set Mandatory and Non-Mandatory Rules)
Version Management and Design Sharing	**Designers**: Save Designs to Vault with Version on Save, View Version Graph, Share Designs, Check In/check Out Design Sub-Objects (including derived data objects such as PDF and Live BOM), Update Designs With Latest Library Changes
	Design owners (design authors): Share Designs With Other Designers, Release Lock
Unified Search/Live BOM	**Designers**: Search Parts From Cadence-supplied or Custom Libraries/Ultra Librarian/SamacSys to Add/Replace/ Modify Parts in Designs
	Administrators: Map Properties and define available properties
Unified Search	**Administrators and Library Administrators**: Enable/ Disable Data Sources, Define Attribute Aliases
Live BOM	**Administrators**: Enable/Disable Data Sources, Define Attribute Aliases
In-Design Workflow	**Administrators**: Edit/Create In-design Workflows, Publish
	Designers: View and Adhere to In-Design Workflow
Allegro Library Manager (Allegro EDM)	**Librarians**: Library Object Creation/Modification/Verification/ Release/Library Distribution, Managing Working Sets
	Library Administrators: Classifications Management, PPL Management/Lifecycle/User/Library Management, Flow Configuration

Allegro X Pulse and Allegro EDM Features and Roles Mapping

Role	Eligible Features	Equivalent Groups from Previous Allegro EDM Versions (from Database Administrator, Team Design Option, File System)	Comments
Administrator	Part Request Designer Operations		
	Part Request Administration Operations		
	Pulse Manager (web page) Designer Operations		
	Publish for Manufacturing Administration Operations		
	Publish for Manufacturing Designer Operations		
Librarian	Part Request Designer Operations	ECAD Librarian	
	Part Request Librarian Operations		
	Pulse Manager (web page) Designer Operation		

Allegro X Pulse and Allegro EDM Features and Roles Mapping

	Publish for Manufacturing Designer Operations	
Library Administrator	Part Request Designer Operations	ECAD Library Administrator
	Part Request Librarian	
	Pulse Manager (web page) Designer Operation	
	Publish for Manufacturing Designer Operations	
Designer	Part Request Designer Operations	
	Pulse Manager (web page) Designer Operation	
	Publish for Manufacturing Designer Operations	
Design Integrator	Pulse Manager (web page) Designer Operation	

Allegro X Pulse and Allegro EDM Features and Roles Mapping

Design Administrator	Part Request Designer Operations	
	Pulse Manager (web page) Designer Operation	
	Publish for Manufacturing Administration Operations	
	Publish for Manufacturing Designer Operations	
Roles: offline_access	Not to be modified	
Roles: uma_authorization	Not to be modified	
Special Users		**Default Password**
admin	Pulse Manager (web page) Master/Data Node Setup	admin
	User Management	
administrator		pwd
pulse	OS-level local user. Auto-created as part of Service installation on Linux	Service Account user without password

Related Topics

- User Management in Pulse
- Accessing Pulse User Management Module
- Creating Users for the Pulse Platform
- Accessing Pulse Service Manager Web Page
- For information about Pulse Managed Workflow, Pulse Team Collaboration, and Pulse Web Participant, refer to *Allegro X Pulse Configuration Basics.*

Assigning Roles to Users on Pulse Platform

To define the level of access a user has on the Pulse platform and for Pulse-enabled functions in client applications such as Allegro X System Capture, assign roles for users created in Pulse. You can assign multiple roles to a user.

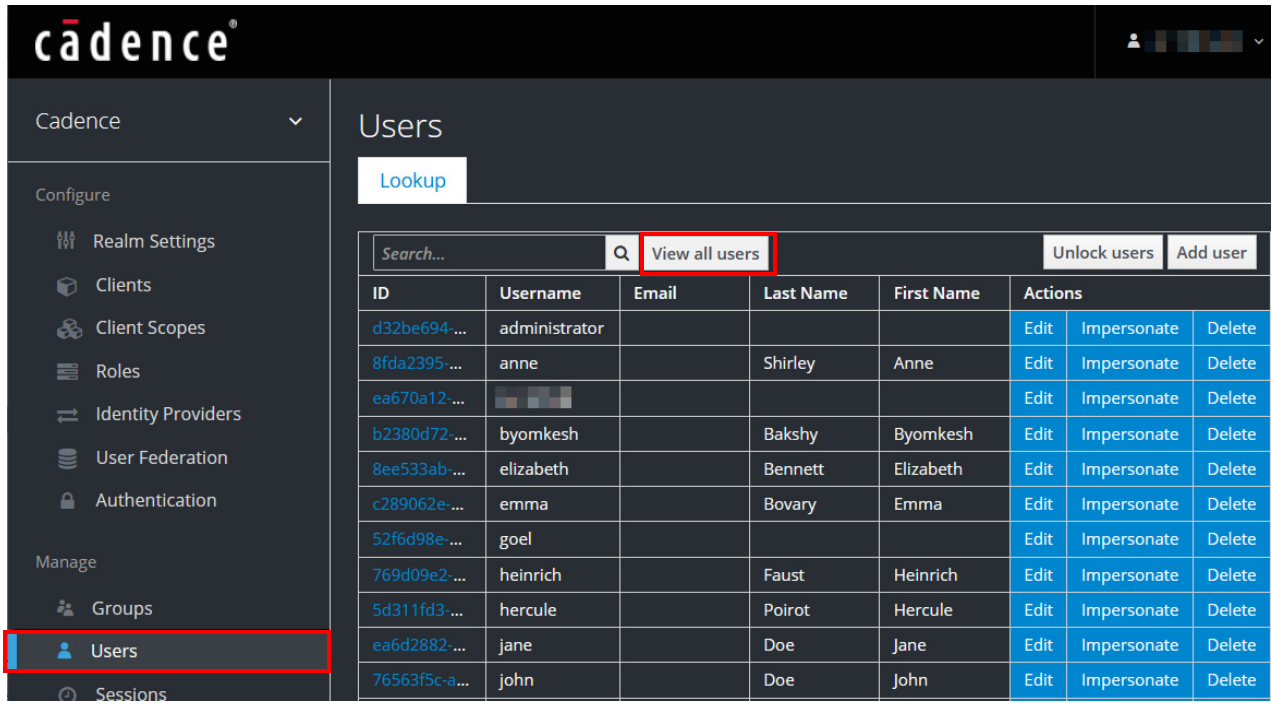
Role assignments allocate slot types to users. For example, if you assign a web participant role to a user, this uses one of the available web participant slots.

Slots are only for the Pulse Managed Workflow, Pulse Team Collaboration, and Pulse Web Participant tiers.

To define a role for a Pulse user, do the following:

1. Open the *Pulse User Management* module.
2. Click *Users* under Manage.

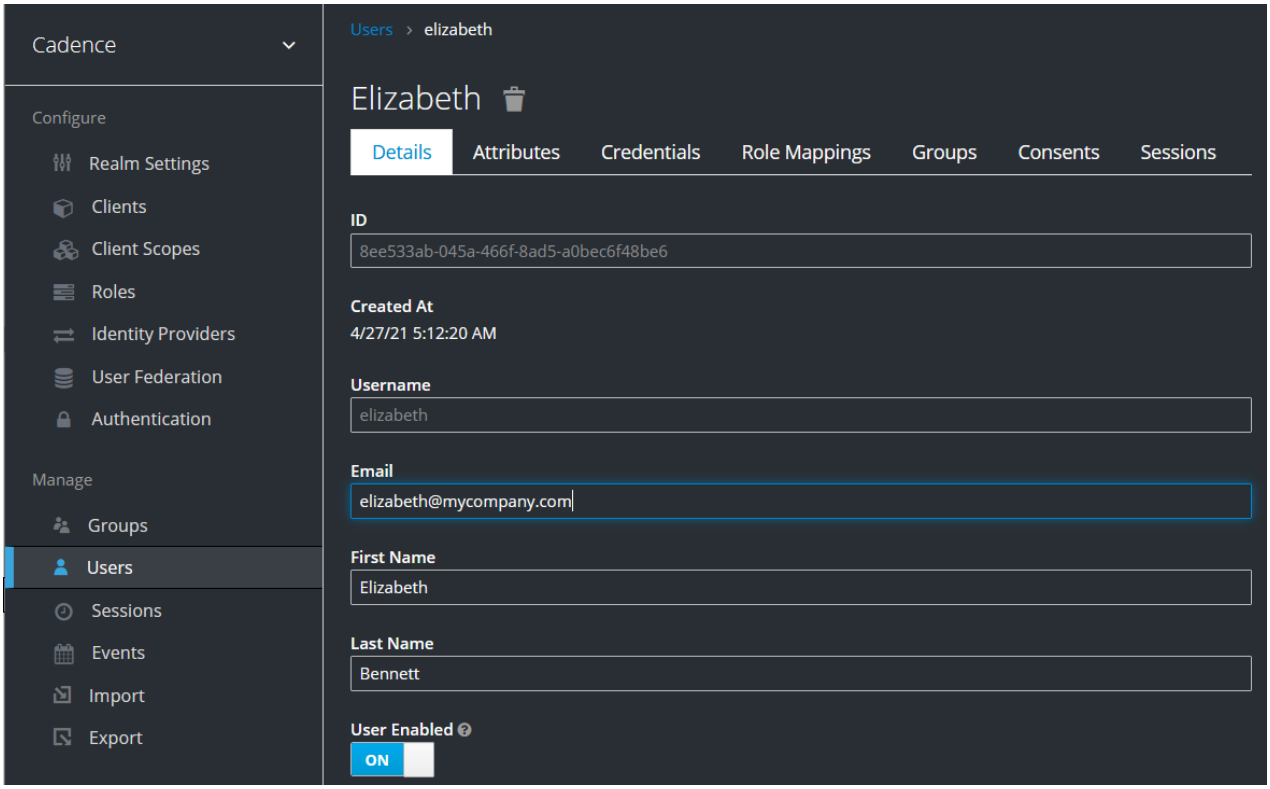
If the list of users already defined in Pulse is not displayed, click *View all users*. The list of users is displayed. Some details have been blurred in this image for confidentiality.



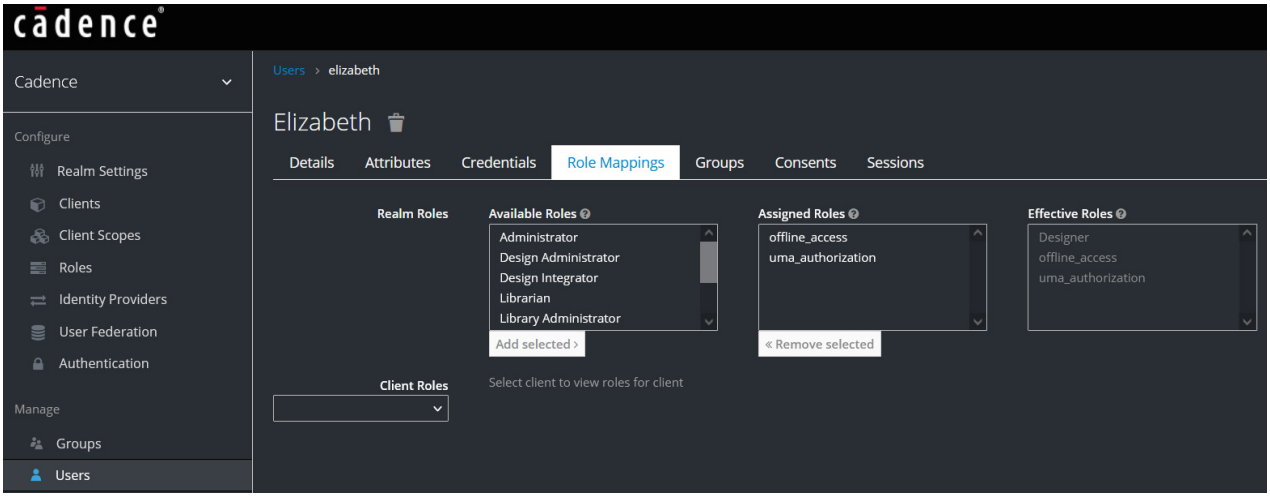
3. Click the ID of the user to whom you want to assign a role.

   Note: A non-administrator user cannot assign roles to users.

The user details are displayed.



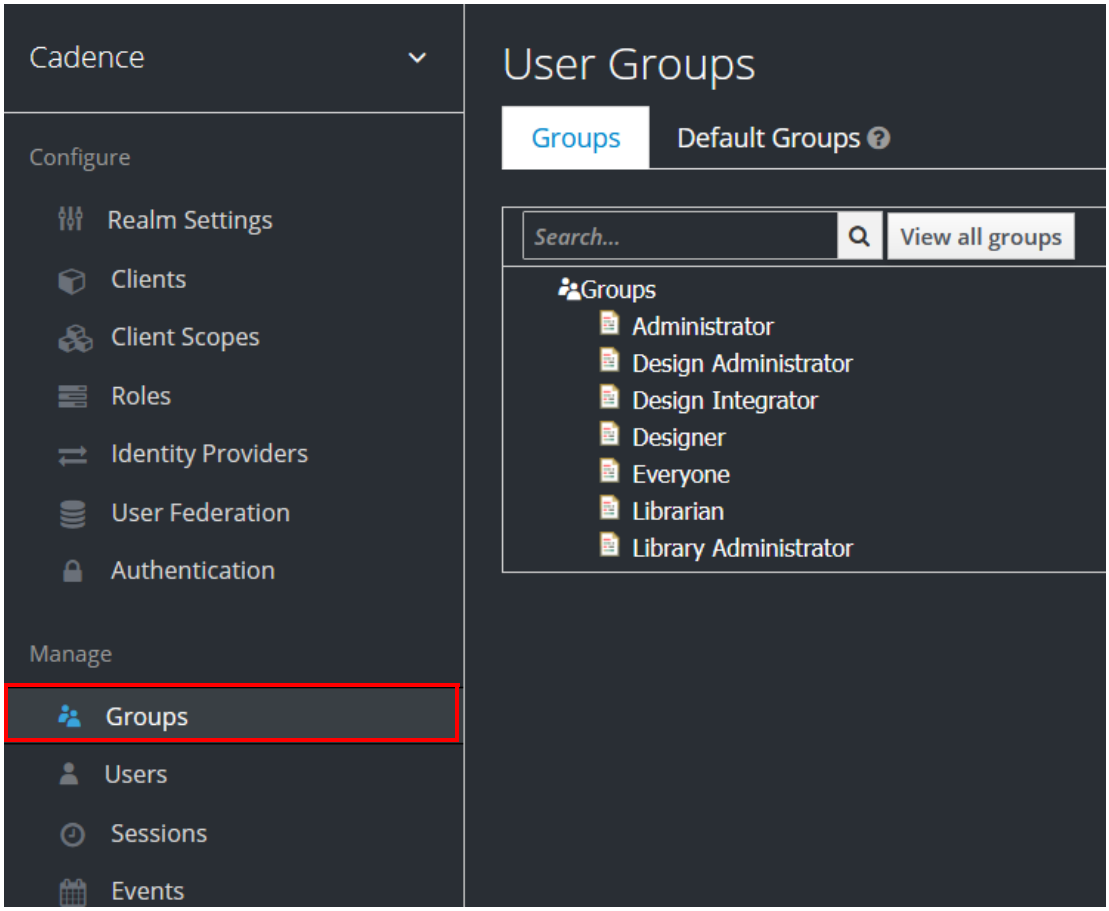
4. Click the *Role Mappings* tab.



5. Select the role you want to assign to the user from the *Available Roles* list and click *Add selected*.

- *Assigned Roles* lists the role that an administrator manually assigns to users.
- *Effective Roles* is a combination of the assigned role and roles implicitly assigned elsewhere, such as Groups.

Do not modify the Client Roles.



Related Topics

- User Management in Pulse
- Accessing Pulse User Management Module
- Creating Users for the Pulse Platform
- Accessing Pulse Service Manager Web Page

Configuring LDAP Sync with the Pulse Primary Node

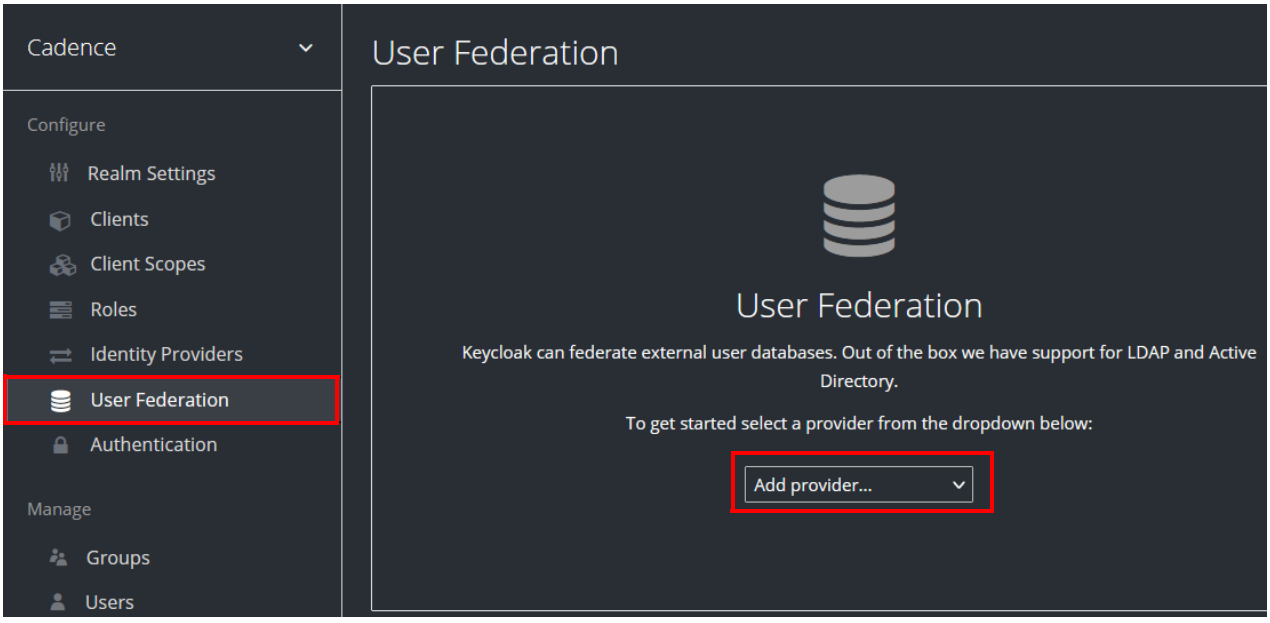
Instead of manually specifying user credentials one at a time in the *Pulse User Management* module, you can leverage the existing user name and password credentials of a network domain. When correctly set up, the *Users* page automatically has the user credentials, enabling any registered user to sign in to the Pulse server using their corporate network user name and password.

The Pulse server supports Standard LDAP and LDAPS (LDAP over SSL). If you want to create user credentials from LDAP automatically, it is recommended that you first remove any existing manually created users to avoid duplicate user names.

Taking advantage of Pulse support for Windows authentication, you can also use the Windows login credentials when signing in to the Pulse server. For details on using the Windows login credentials, review the Keycloak documentation on the Kerberos authentication mechanism with the SPNEGO protocol.

To configure LDAP sync, do the following:

1. Access the *Pulse User Management* module.
2. Click *User Federation* under Configure.



3. Click *Add provider* and select *ldap*.

The *Add user federation provider* page is displayed.

For Pulse services, select *Unsynced* as the Edit Mode.

For Pulse services, select *simple* as the Bind Type.

4. Specify the mandatory fields.

   Refer to the Keycloak documentation for details about each field.

LDAP Syncing Constraints on Pulse Platform

- If you have a limited number of connections or if more users than the number of available licenses need access, users connect to Pulse on a first come, first served basis.
- LDAP users are imported into Pulse only after successful authentication. However, later changes made to user definitions cannot be written back to the LDAP system that manages the users.

Administrators launching Pulse Service Manager might find that it takes some time to start because of LDAP server latency.

Creating Groups to Automatically Share Projects

Pulse provides the following default groups:

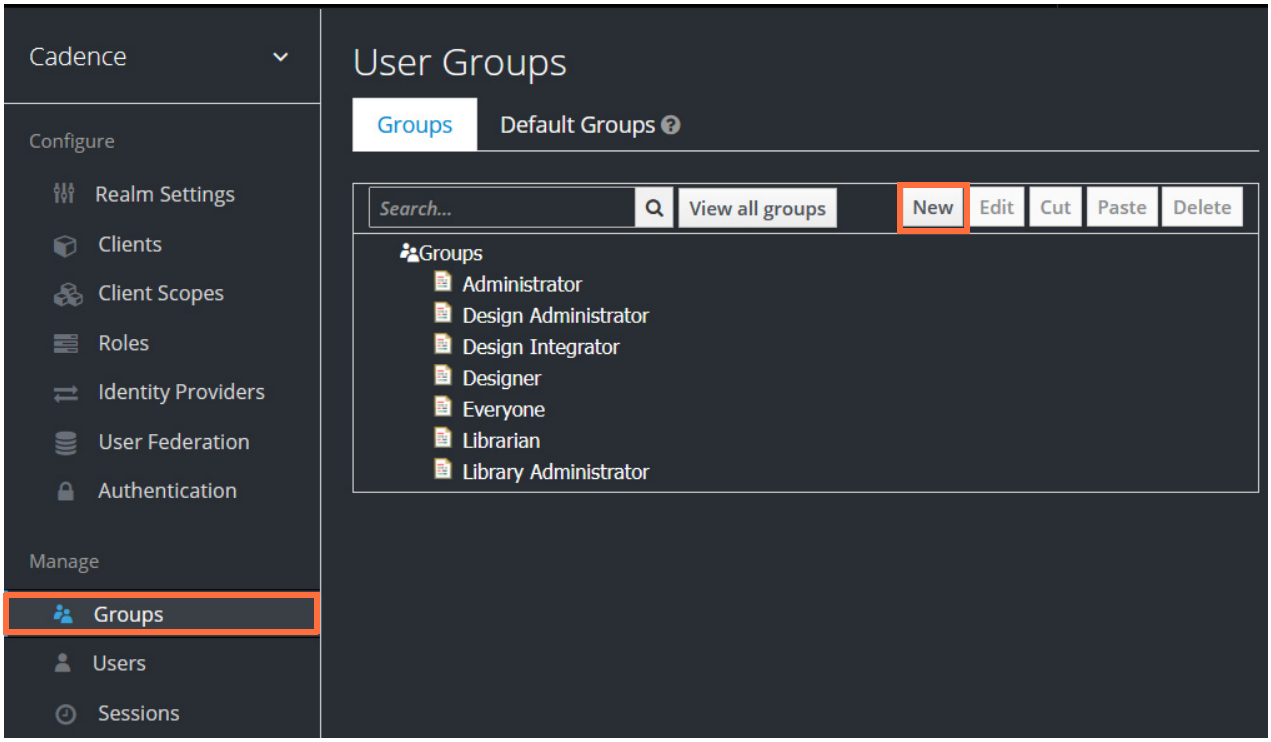
- Administrator
- Design Administrator
- Design Integrator
- Designer
- Librarian
- Library Administrator

You can create new groups. For example, if all designers in a business unit require read-only access to all the design or layout projects on a Pulse primary node, create a group and add the designers to it.

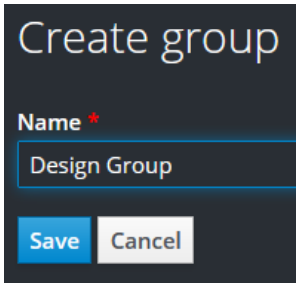
To create a group, do the following:

1. Open the *Pulse User Management* module.
2. Click *Groups* under Manage.

In the following image, `Everyone` is a custom group.

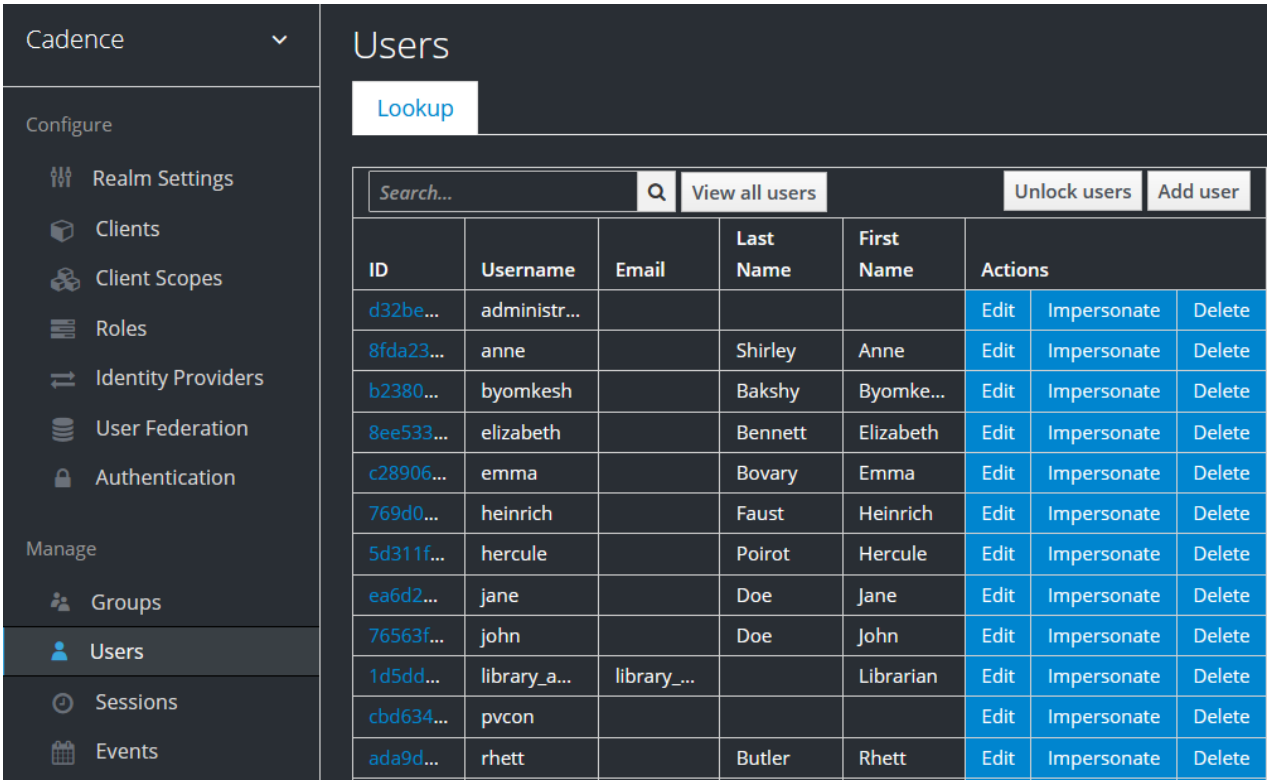


3. Click *New*.
4. Specify a name for the group and click *Save*.

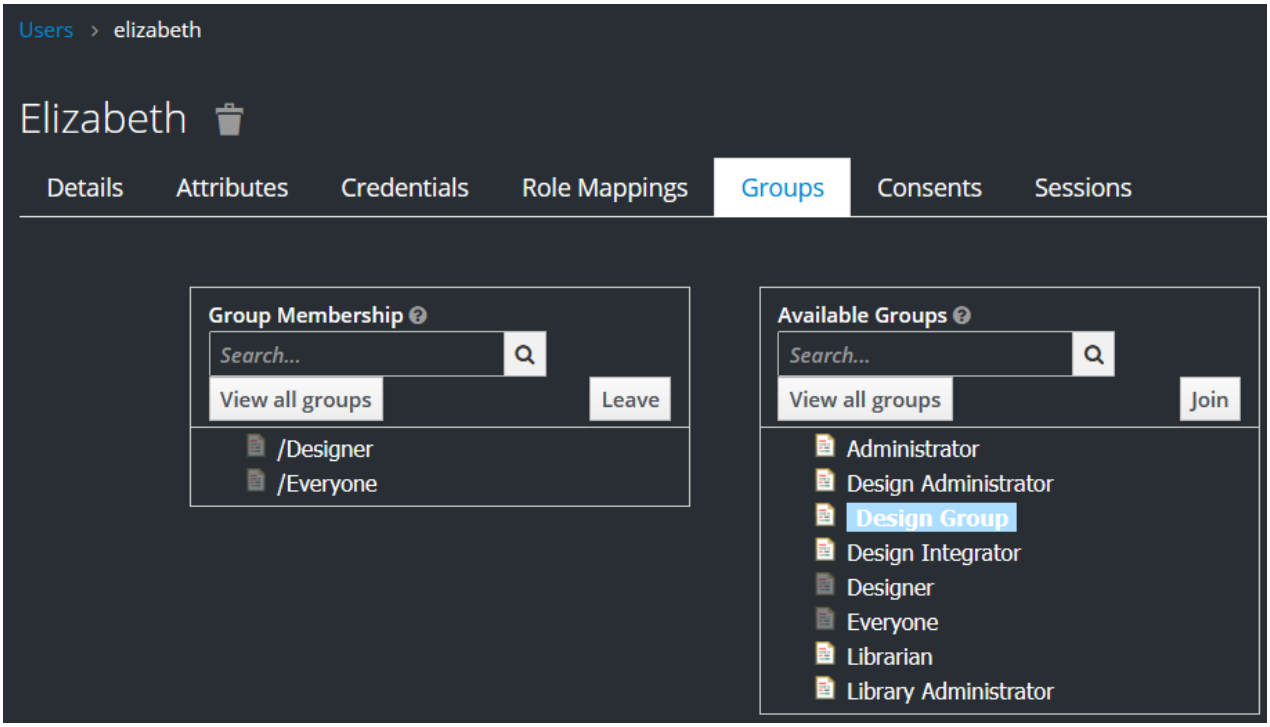


5. Add new members to the group by doing the following:

a. Click *Users*.



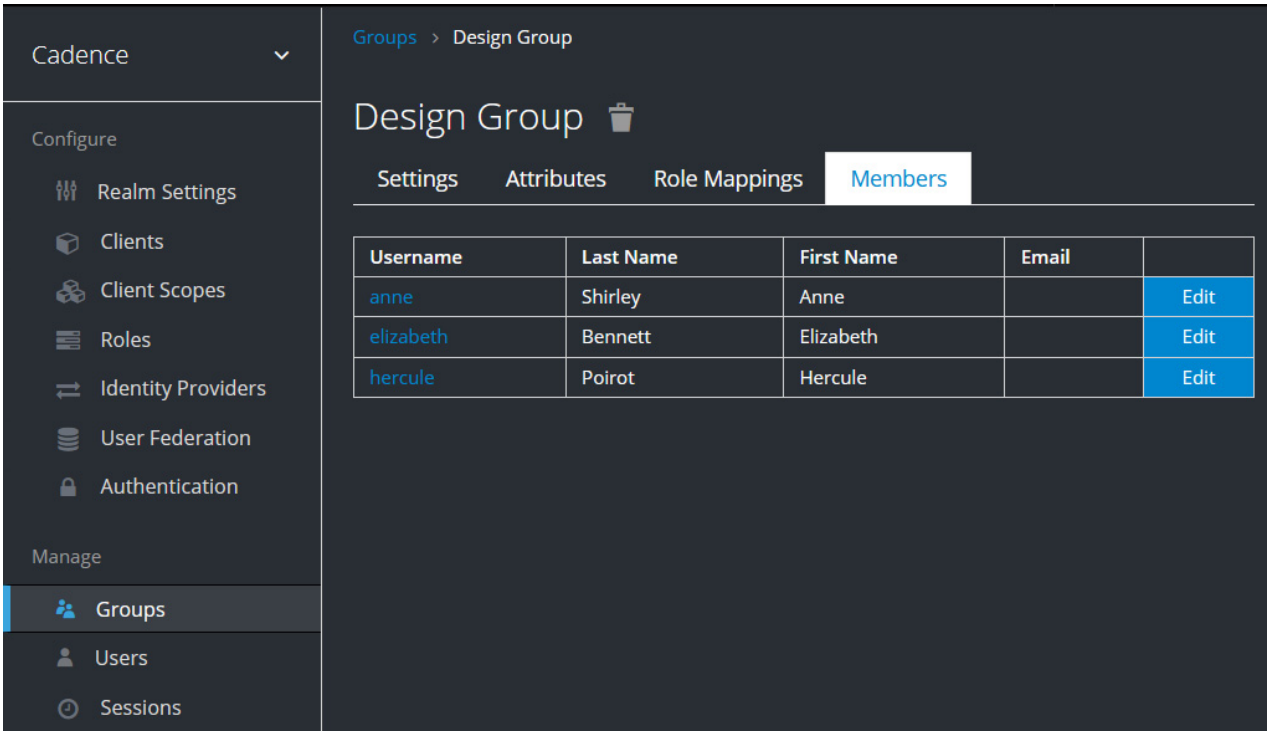
b. Click *Edit* for every user you want to add to the group.



c. Select the group to which you want to add the user.

d. Click *Join*.

In the following image, three users are added to a newly created group.



6. To provide default access to all designs and layouts in the server to all designers in a group, do the following:

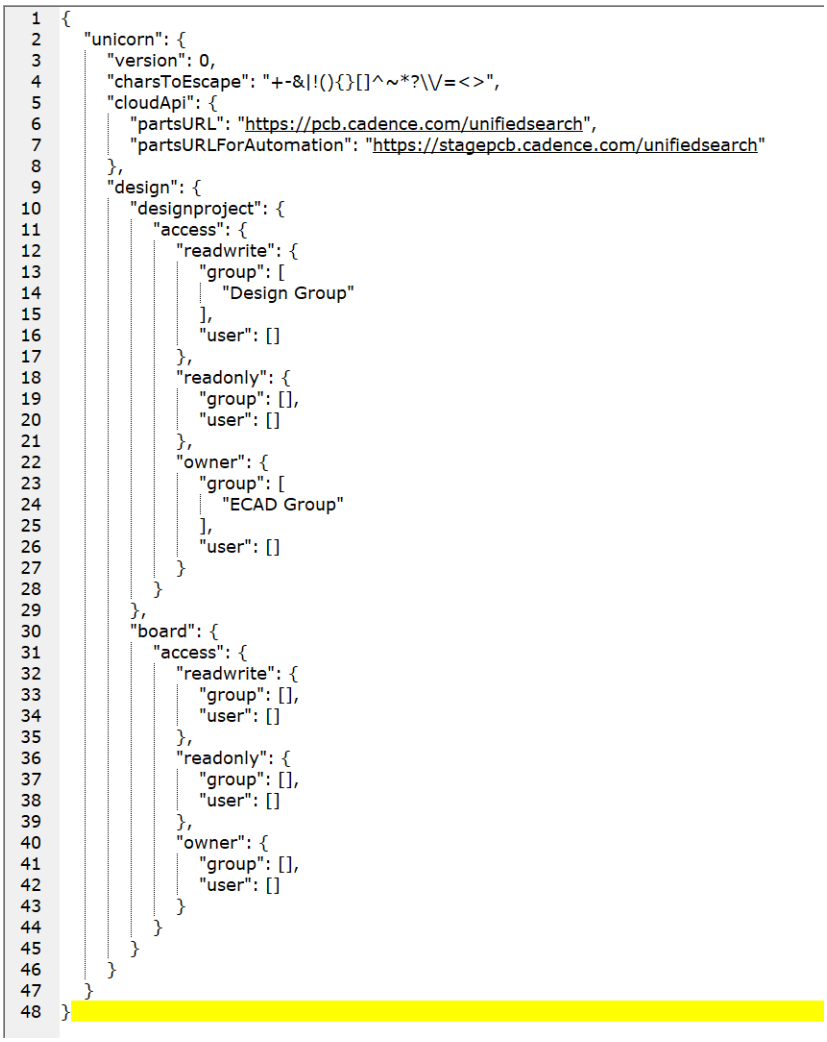
   a. Download the `search.config` file from the Pulse web dashboard.

   b. Add the `"design"` and `"board"` snippet from the out-of-the-box `search.config` file to the downloaded configuration file.

   c. Specify the group name in the relevant section.

      For example, provide read-write access to designers in *Design Group* and project owner access to users in *ECAD Group*.

A properly modified and configured `search.config` file would look as follows and this is the configuration file you upload to the Pulse web dashboard:



```
{
  "unicorn": {
    "version": 0,
    "charsToEscape": "+-&|!(){}[]^~*?\\/=<>",
    "cloudApi": {
      "partsURL": "https://pcb.cadence.com/unifiedsearch",
      "partsURLForAutomation": "https://stagepcb.cadence.com/unifiedsearch"
    },
    "design": {
      "designproject": {
        "access": {
          "readwrite": {
            "group": [
              "Design Group"
            ],
            "user": []
          },
          "readonly": {
            "group": [],
            "user": []
          },
          "owner": {
            "group": [
              "ECAD Group"
            ],
            "user": []
          }
        }
      },
      "board": {
        "access": {
          "readwrite": {
            "group": [],
            "user": []
          },
          "readonly": {
            "group": [],
            "user": []
          },
          "owner": {
            "group": [],
            "user": []
          }
        }
      }
    }
  }
}
```

d. Save and upload the modified configuration file.

The configuration is updated and the changes are available and visible to all designers connected to this Pulse server.

Related Topics

Accessing Pulse User Management Module

Using Global Configuration File

Configuration of Features Specific to Pulse

Features specific to Pulse can be configured globally for all designers connected to a Pulse server, and at a project-specific level. A user defined as `Administrator` in the Pulse User Management module can configure at both levels. `Administrator` users must be logged in with administrator credentials to configure.

The default user name and password are `administrator` and `pwd`. If you changed the administrator password in the *Security* tab of Pulse Service Manager, use that password to log in to the application or dashboard.

Project-specific configuration can also be done by design project owners.

Pulse feature configuration can be done in various ways. Where tasks can be done through Allegro X System Capture and the Pulse web dashboard, instructions and images are for the Pulse dashboard. It is assumed that administrators are unlikely to use System Capture when a task can be done through the dashboard.

The following table illustrates the features that can be configured:

Task	Application/Utility
■ Configuring Unified Search globally ■ Configuring Unified Search globally or at a project-specific level ■ Configuring Live BOM headers ■ Configuring project creation and part request forms	Allegro X System Capture Pulse web dashboard **Note:** Web participants can access Pulse only through a web browser.
Creating a project from scratch and marking it as a template for designers to use during project creation	Allegro X System Capture
Configuring workflows	Allegro X System Capture
Enabling optional Allegro EDM-managed library features	`adwschema` utility
Configuring outputs and the package structure to publish data to a PLM system	Publish for Manufacturing application

Related Topics

Configuring Unified Search Globally

Configuring Unified Search at Global or Project-Specific Level

Configuring Live BOM Headers

Resetting Pulse Web Dashboard Changes to Default

Configuring Forms

Creating Project Templates

Configuring Workflows

Enabling Optional Allegro EDM-Managed Library Features

Using Global Configuration File

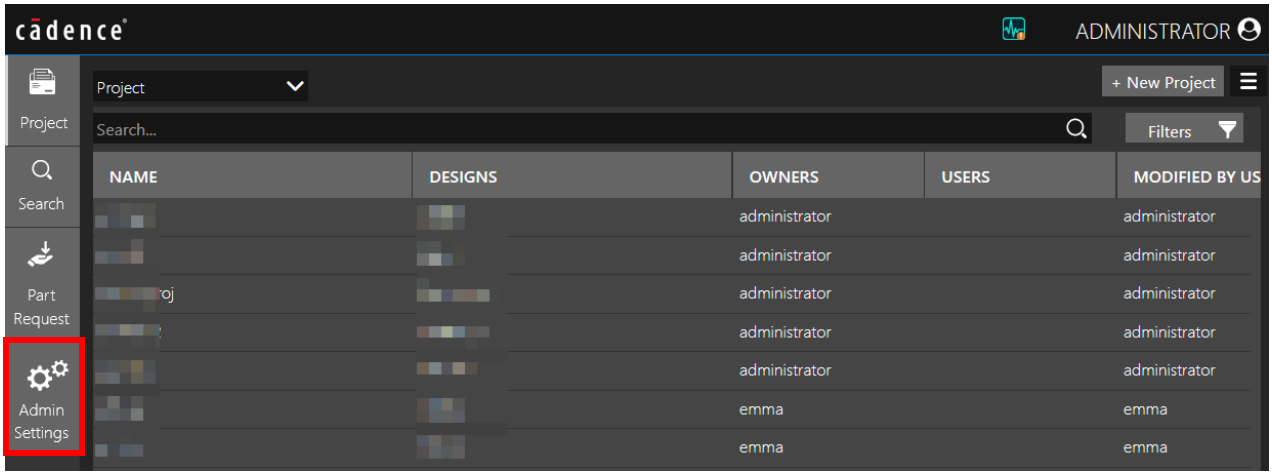
To configure features specific to Pulse at a global level, which means for all designers connected to a Pulse server, you use the `search.config` file, which uses JavaScript Object Notation. The `search.config` file can only be accessed from the Pulse web dashboard; it cannot be accessed from the Allegro X System Capture desktop application.

The steps to access the configuration file, download it, and upload it are the same regardless of the feature you want to configure at a global level.

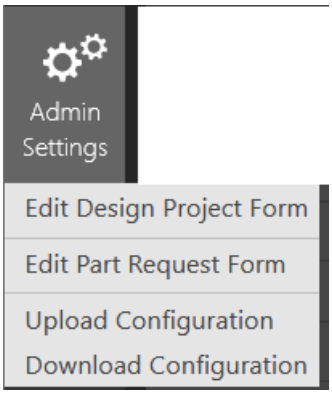
To configure, do the following:

1. Access the Pulse web dashboard by typing `http://<Pulse access URL>:7100/projects` in a web browser.

The Pulse dashboard is displayed.



2. Click *Admin Settings*.



3. Click *Download Configuration*.

   The `search.config` file is downloaded to the default download location.

4. Open the downloaded file in a text editor.
5. Copy the required code snippet from the out-of-the-box `search.config` file, which is available at: *<Cadence installation directory>*`\tools\pcbdw\configs\unicorn`.
6. Modify the contents in your downloaded `search.config` file as required.
7. Save the `search.config` file.
8. Click *Admin Settings*.

9. Click *Upload Configuration*.
10. Select the modified `search.config` file and click *Open*.

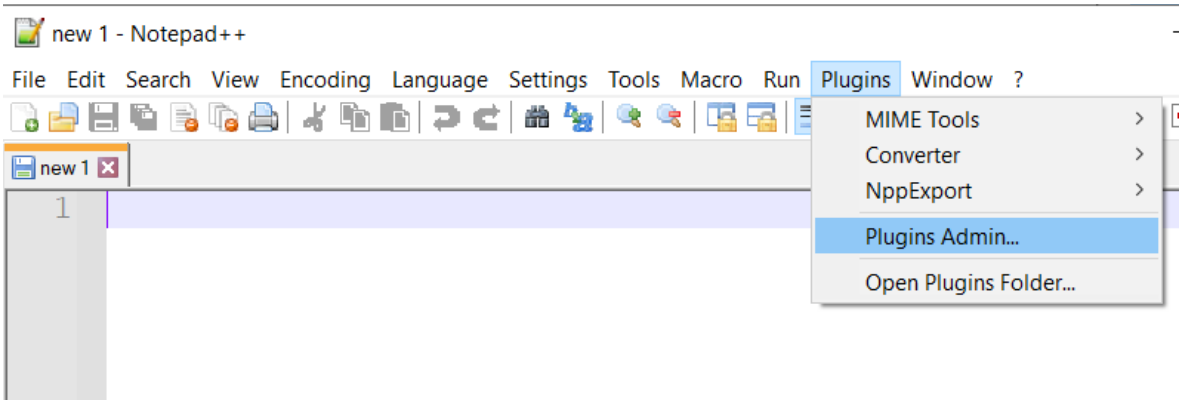
    The configuration is updated and the changes are available and visible to all designers connected to this Pulse server.

    If you are unfamiliar with JSON editing, consider going through the content at *https://www.json.org/json-en.html*.

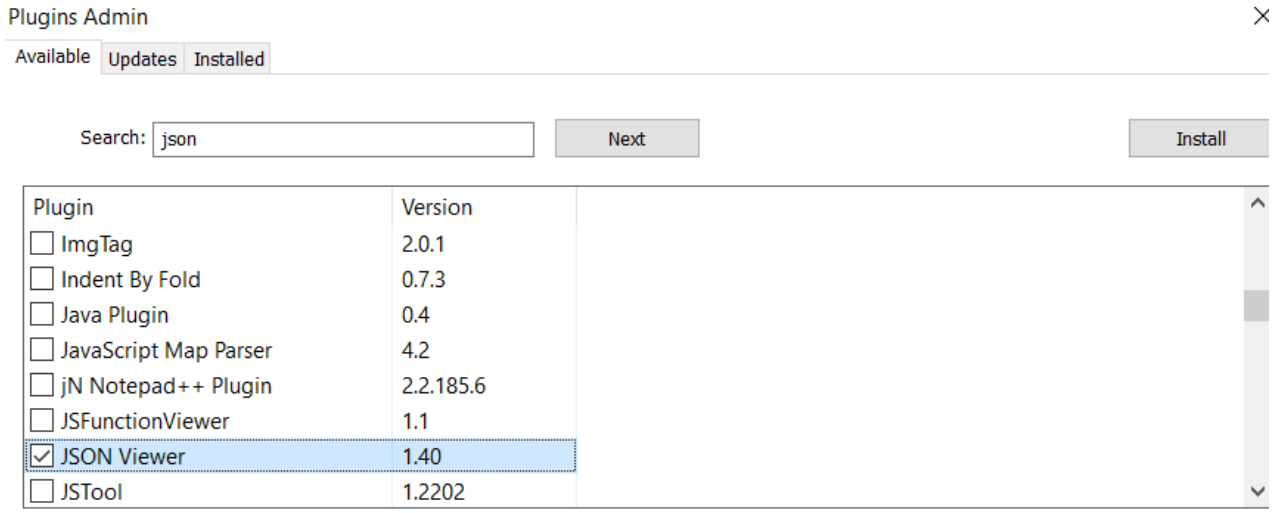
    Some text editors also provide JSON viewer plugins, which flag inconsistencies in the tags. For example, if you use Notepad++, you can enable the JSON viewer plugin by doing the following:

    a. Open Notepad++.

    b. Select *Plugins - Plugins Admin*.



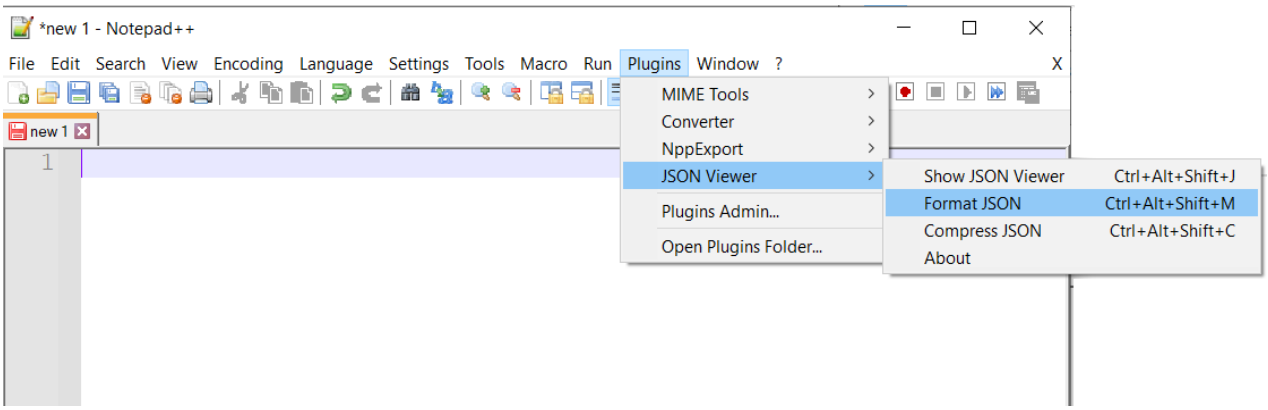
c. Search for JSON and select *JSON Viewer*.



d. Click *Install*.

Notepad++ closes and restarts automatically.

e. Modify the JSON as required and check for inconsistencies using the following option:



Related Topics

Modifying Filter Display Order

Configuring Contents of Summary Column in Part Search

Enabling Wild Card or Regular Expressions in Part Search

Configuring Date Format

Disabling Workflow Selection by Non-Administrator Users

Enabling Notifications

Enabling Part Request Process With Review

Hiding Pulse Part Request Feature

Configuring Unified Search Globally

You can configure part search at a global level for all projects on a Pulse server as follows:

By publishing changes for all designers:

- Configuring Part Search Rules
- Configuring Part Content Providers

By controlling globally through the `search.config` file:

- Modifying Filter Display Order
- Excluding Filters
- Configuring Contents of Summary Column in Part Search
- Enabling Wild Card or Regular Expressions in Part Search

  The `search.config` file can be downloaded and uploaded from:

  - System Capture user interface - The configuration applies the changes to all projects on the client machine.
  - Pulse web dashboard - The configuration applies the changes to all projects on the Pulse primary node.

Configuring Part Search Rules

You can direct or influence designers to use a particular set of parts defined at the organization level. These parts might be based on parameters such as reliability, availability, best deals based on vendor relations, design type, EOL date, and so on.

You can define these parameters in a way that ensures that their numbers are reduced in the unique BOM list across all projects or systems, which improves efficiency. This also has benefits such as:

- Fewer unique parts means less interaction with various vendors.
- Cost efficiency due to bulk purchases of parts
- Increases the probability of designers using recommended components, which improves efficiency due to correct part selection thus shortening the design cycle.

Part search facilitates visual indication for recommended components, restrictions for non-recommended parts, and the option to add parts with warnings that are completely configurable at the organization level.You can use various color combinations with actions to define standards at the organization level for a particular set of design types. For example, `Recommendation for RF Designs` can be marked with a green icon, or `Highspeed` with blue.

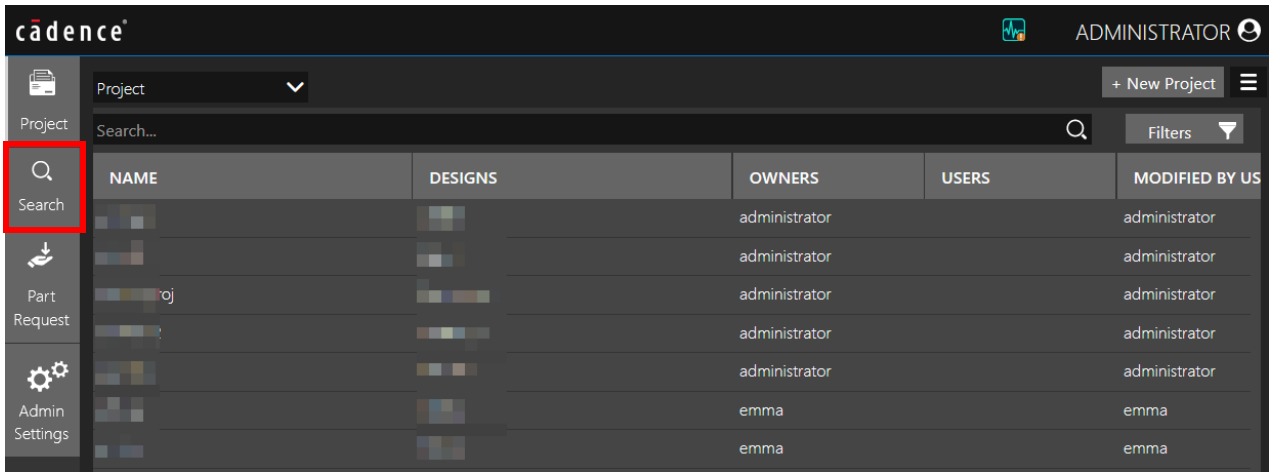
You can also restrict the use of certain parts based on specific criteria by assigning appropriate actions and notes as required.

To define the use of a particular set of parts by designers, do the following:

1. Access the Pulse web dashboard by typing `http://<Pulse access URL>:7100/projects` in a web browser.

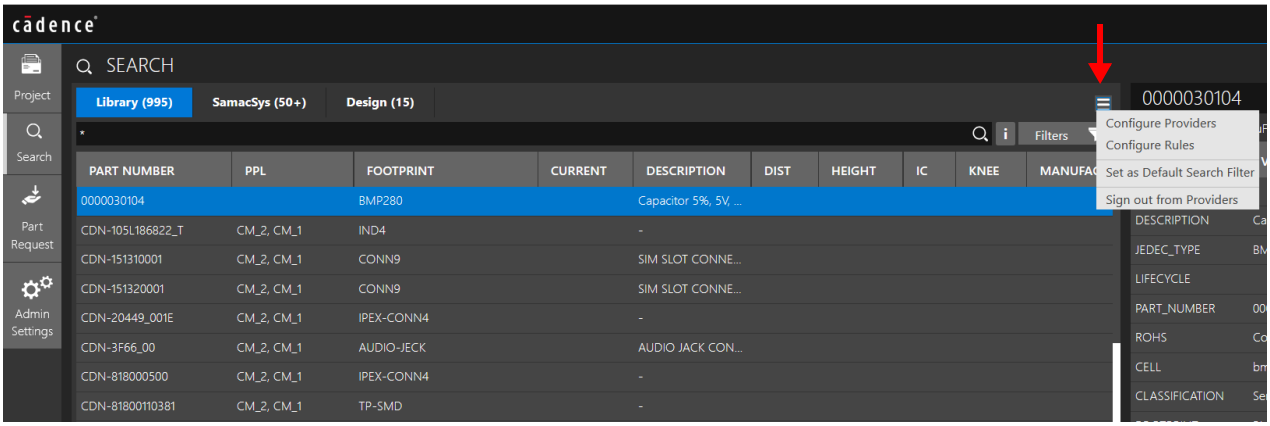
   The Pulse dashboard is displayed.

2. Click the *Search* tab on the left.



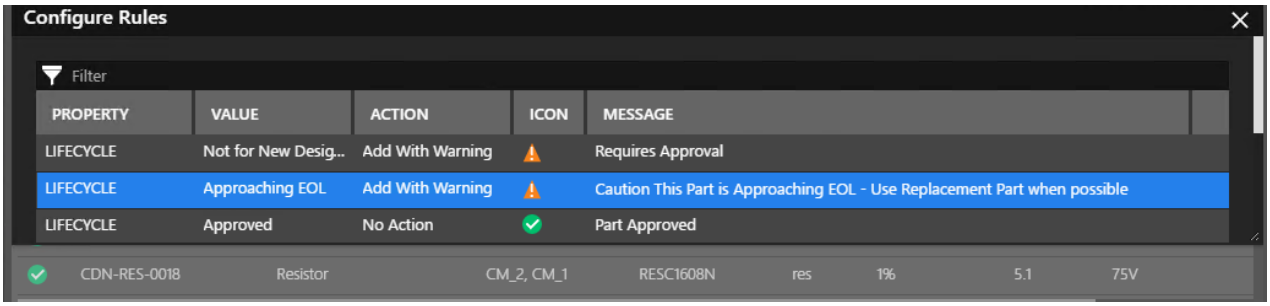
   The *Search* pane is displayed.

3. Click the hamburger menu on the top right of the *Search* pane.

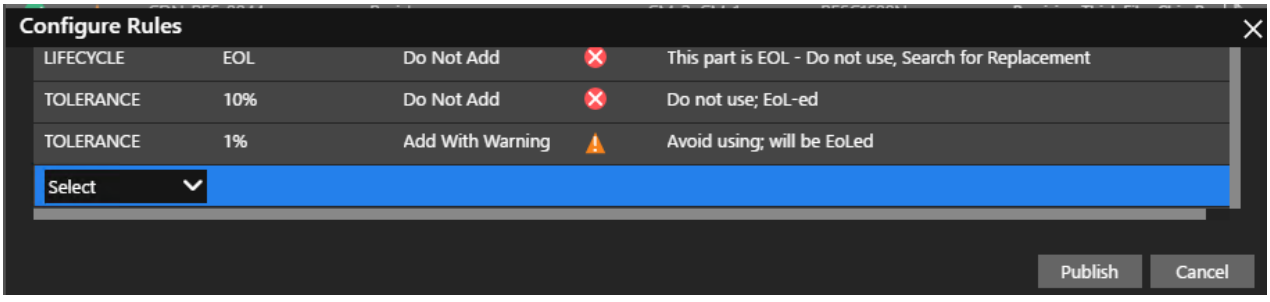


4. Click *Configure Rules*.

The *Configure Rules* dialog is displayed.

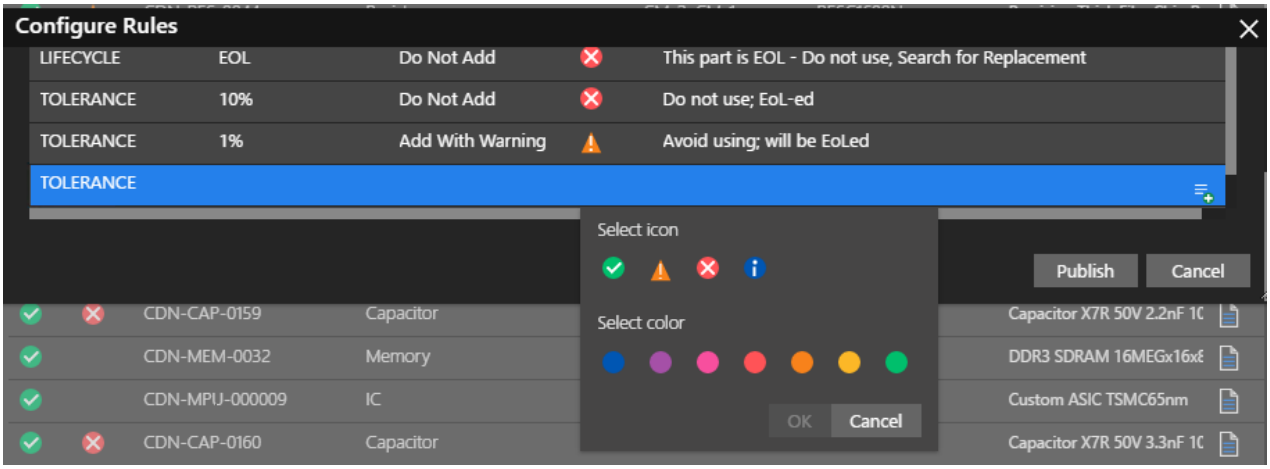


5. Click in an empty row.



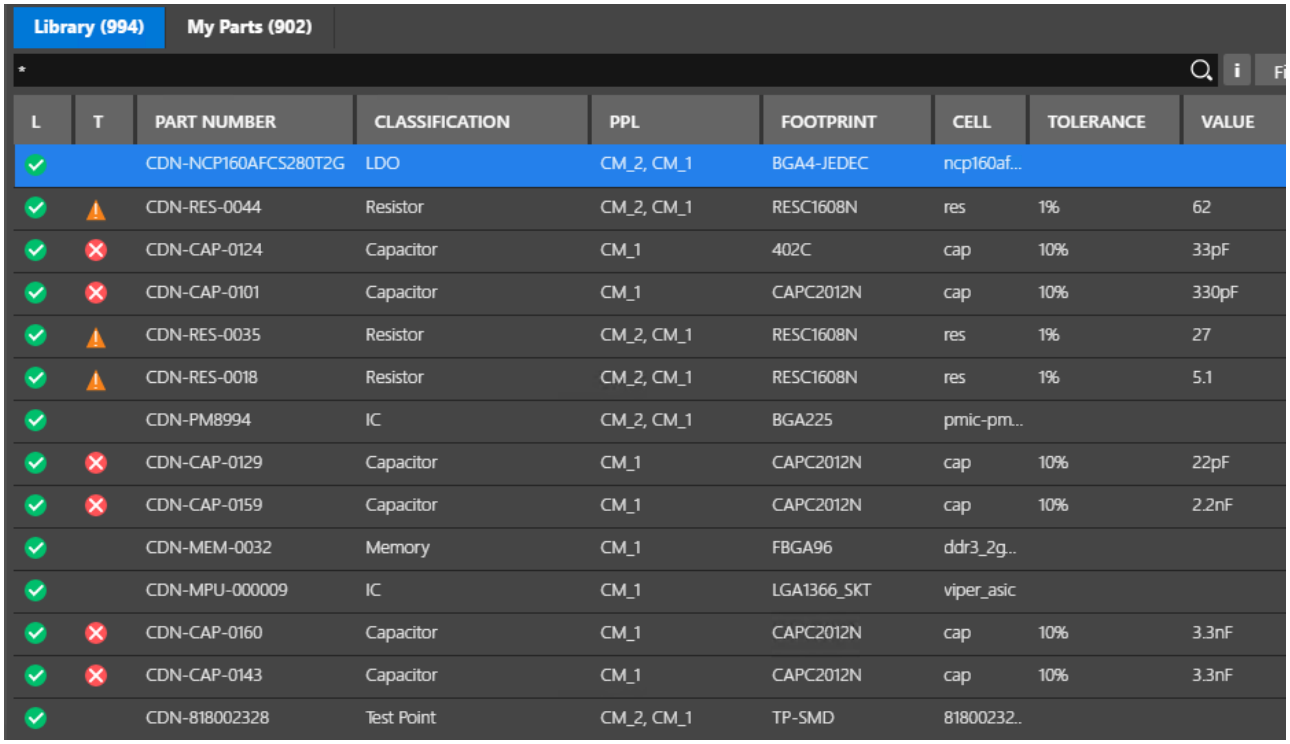
6. Select the required property, value, and action from the drop-down fields.

7. Select the required icon and color for the specified action.

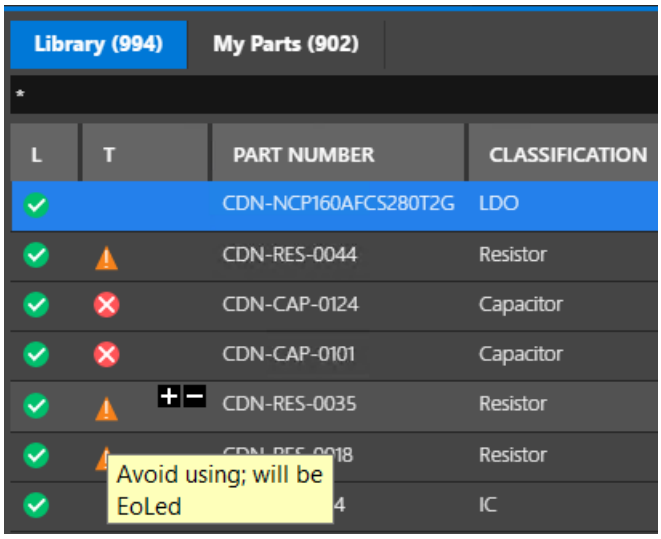


8. Add a note.
9. Click *Publish* to apply the new rule to parts that match its specifications.

   After the rules are specified and published, designers view parts as follows in the part search pane:



Hovering the cursor over a part displays a tool tip with more details.



Configuring Part Content Providers

Pulse provides access to external parts from SamacSys and Ultra Librarian. However, if you want to provide parts only from `CDS_SITE` or EDM-managed libraries, you can disable one or both content providers. If both are disabled, Allegro X System Capture designers are no longer prompted to log in to the part content providers.

You can also configure the following:

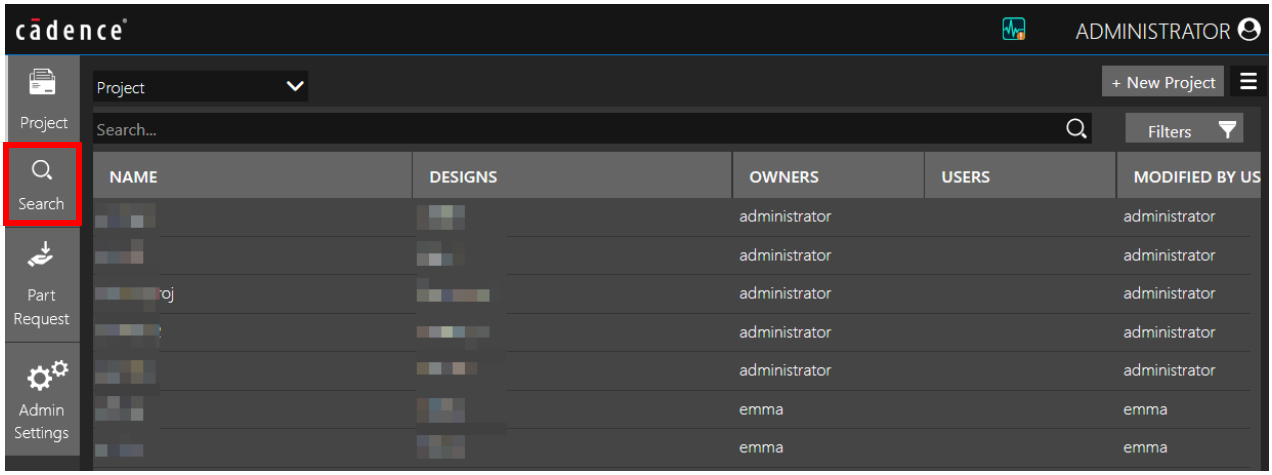
- whether designers can search for parts from a particular provider
- whether a new part request can be raised from a content provider
- whether a part from a provider can be placed on the Allegro X System Capture canvas
- which properties should be annotated when placing a part in the System Capture canvas

To disable content providers, do the following:

1. Access the Pulse web dashboard by typing `http://<Pulse access URL>:7100/projects` in a web browser.

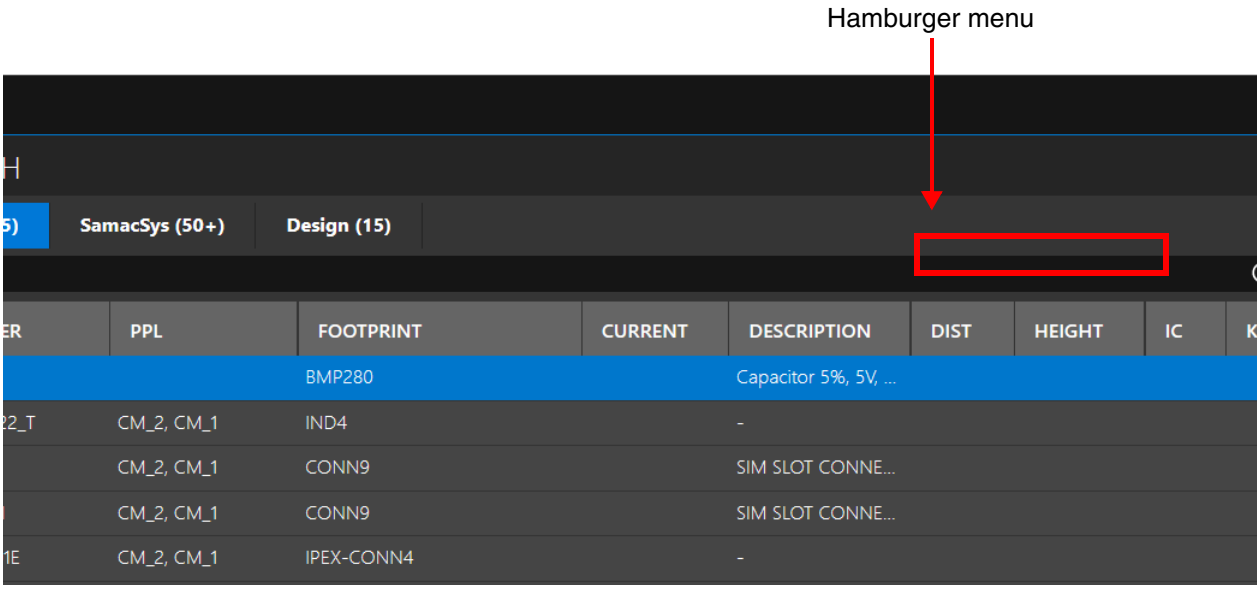
   The Pulse dashboard is displayed.

2. Click the *Search* tab on the left.

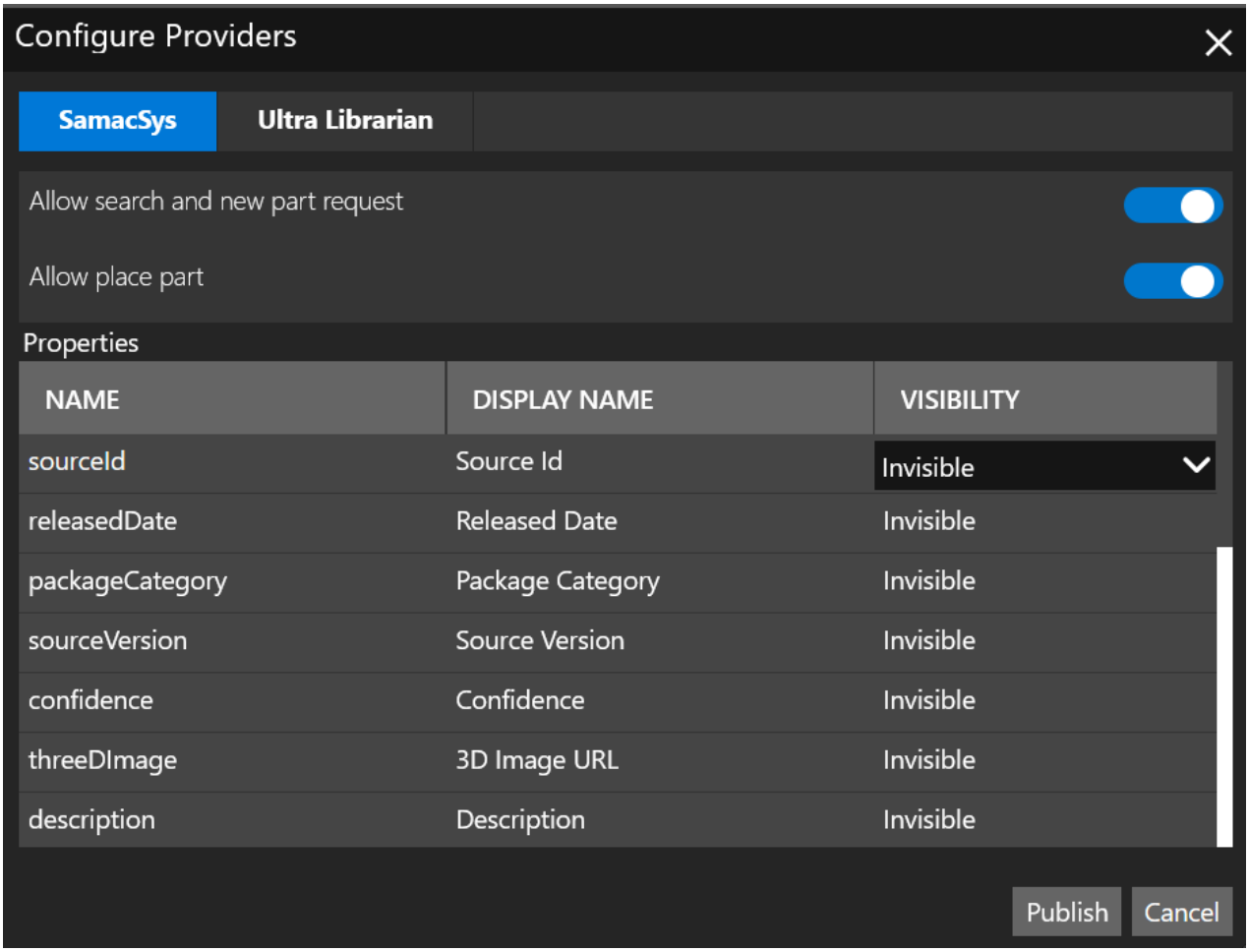


The *Search* pane is displayed.

3. Click the hamburger menu on the top right of the *Search* pane.
4. Select *Configure Providers*.

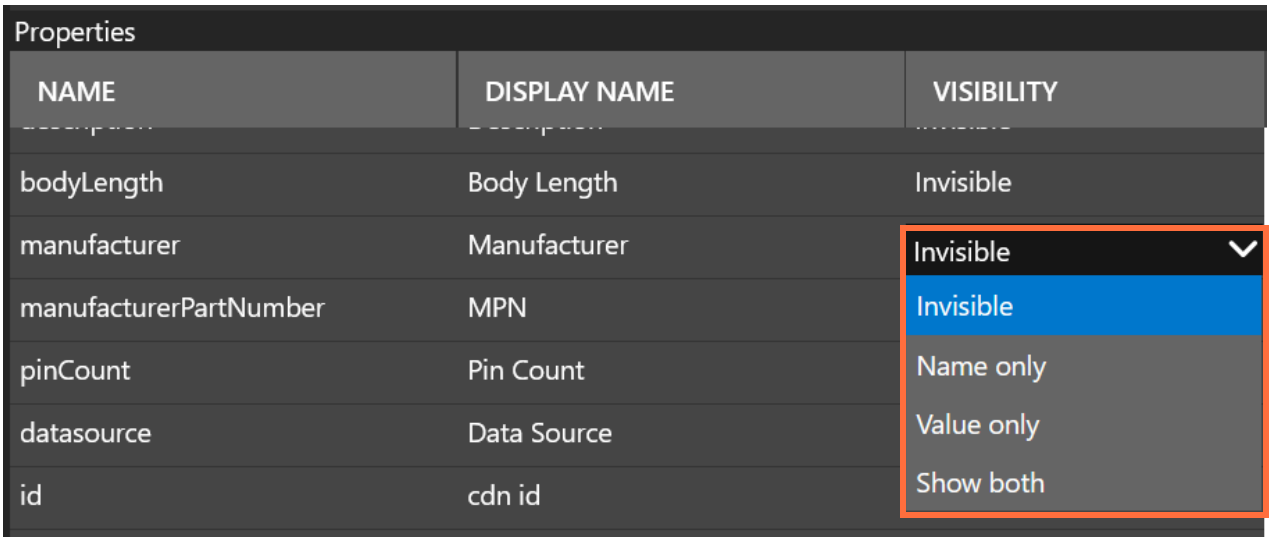


The Configure Providers dialog box opens.

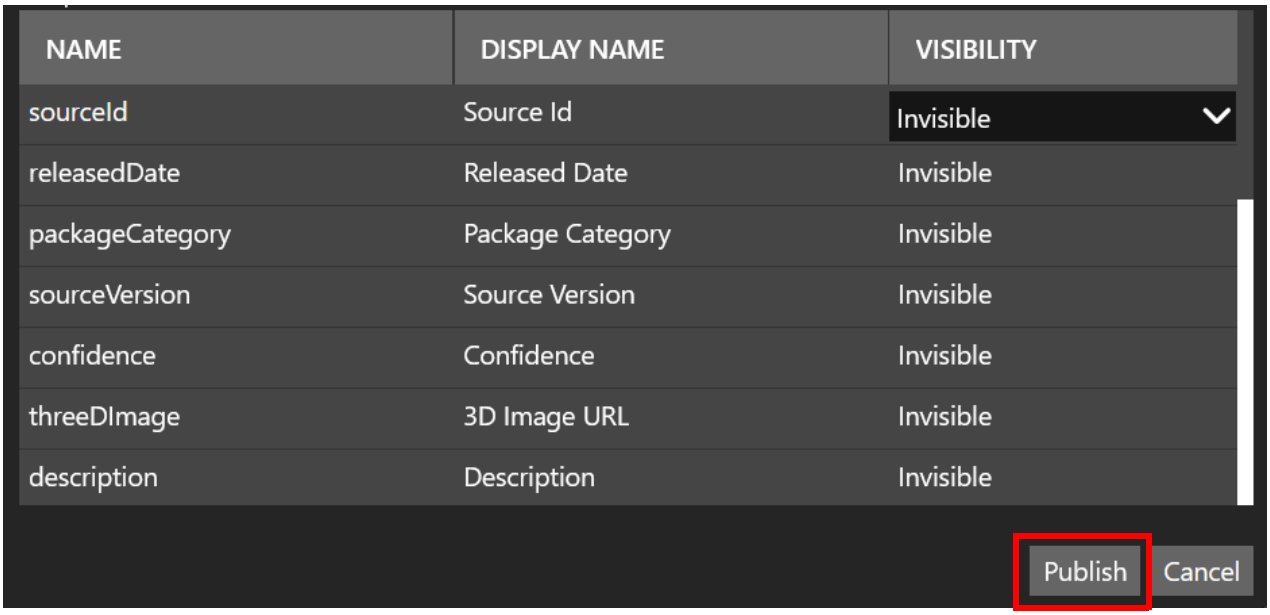


5. Toggle off the *Allow search part and new part request* button to:
   - ❑ make parts from a provider non-searchable.
   - ❑ disable new part requests for a provider.
6. Toggle off the *Allow place part* button to disable placement of parts from a provider on the Allegro X System Capture canvas.

7. In the *VISIBILITY* column, define which properties should be annotated on the canvas for a placed part in Allegro X System Capture.



8. Click *Publish*.

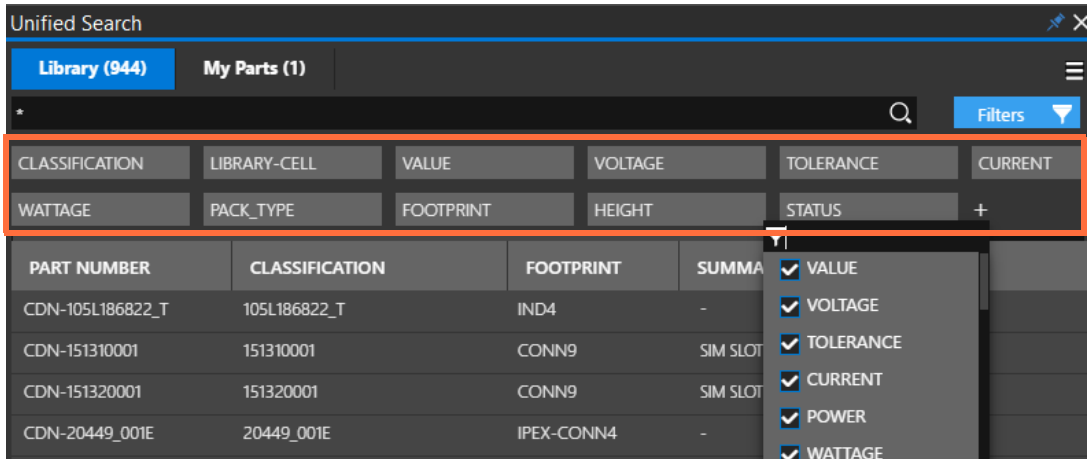


The new settings are published for the selected content provider.

Note: To reflect these modified settings, designers need to restart System Capture.

Modifying Filter Display Order

A filter is a list of default part properties, such as *CLASSIFICATION*, *VALUE*, *VOLTAGE*, *WATTAGE*, *TYPE.*



Filter ordering is controlled globally through the `search.config` file. Modified `search.config` settings apply to all places where part search is displayed, such as Allegro X System Capture and the Pulse web dashboard.

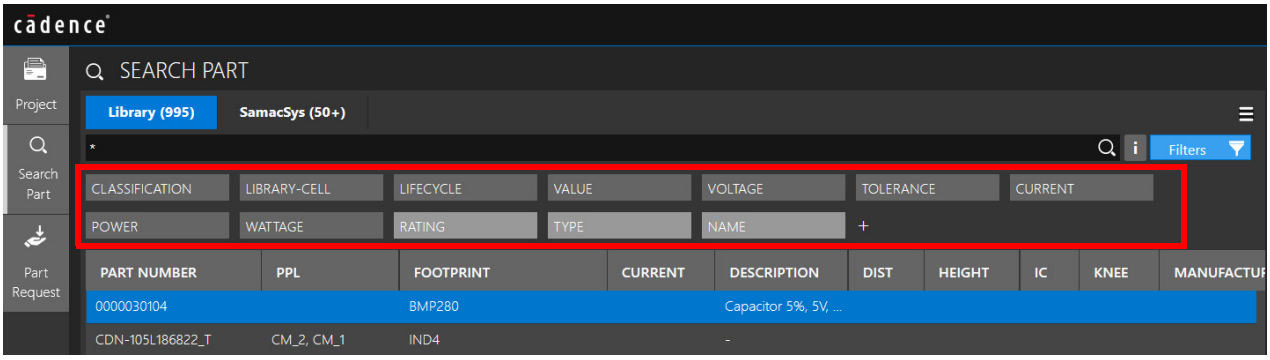
To configure the display order of the filters, do the following:

1. Add the `facets` code snippet from the out-of-the-box `search.config` file to the downloaded configuration file.

   The following is the out-of-the-box sample:

```
{
        "name": "facets",
        "display": "",
        "includes":
"classification,lifecycle,value,voltage,volt,tolerance,tol,c
urrent,curr,power,watt,wattage,rating,material,type,name,pac
k_type,footprint,height,status,rohs,cost,price,*",
        "excludes": ""
```

```
}
```

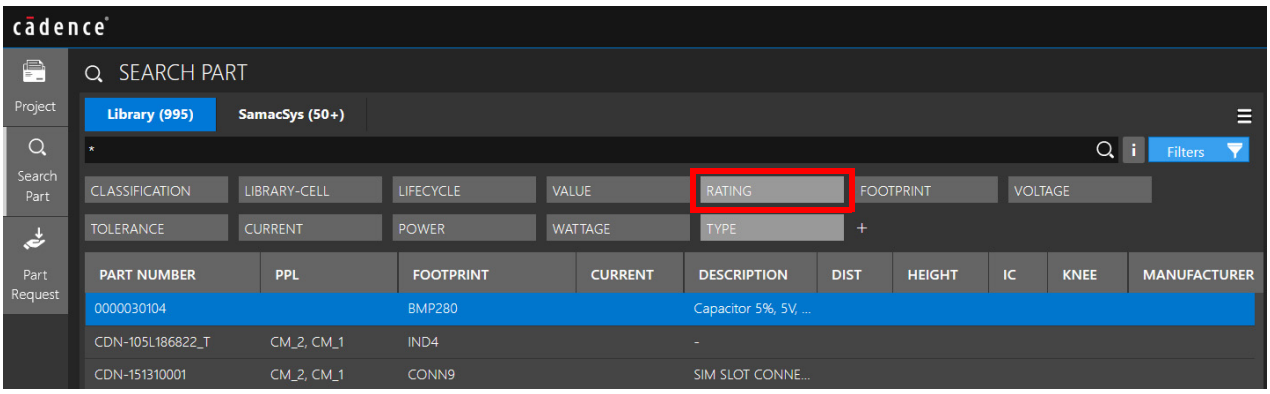


2. In the downloaded file, modify the order as required.

For example, change the order as follows:

```
"classification,lifecycle,value,rating,voltage,volt,tolerance,t
ol,current,curr,power,watt,wattage,material,type,name,pack_type,
footprint,height,status,rohs,cost,price,*",
```

Filters are now displayed as follows:



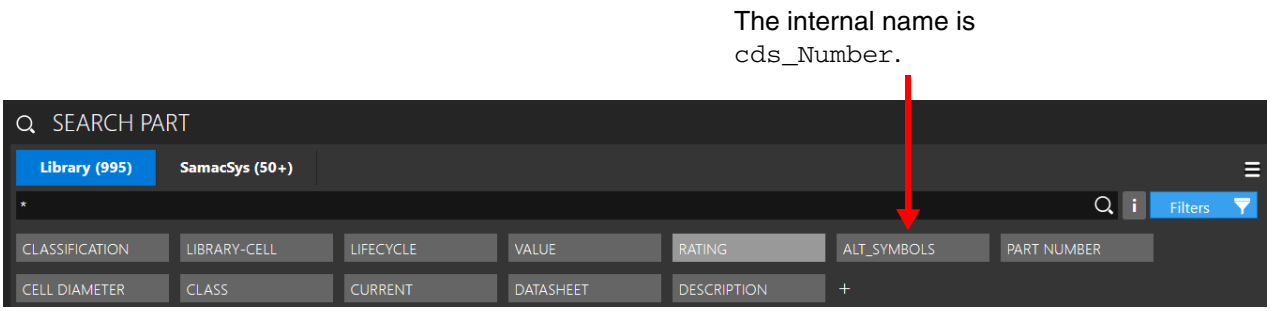
Note: The classification is always displayed first regardless of the order you set.

You can also use the asterisk symbol as a placeholder for everything after `<N>`. Define the filters you want to view in a particular order, then use the asterisk. All the other filters are displayed in alphabetical order. If you delete the asterisk, the only filters available are the ones explicitly specified in the list.

For example, specify the filter list as follows and upload the configuration file:
`"classification,lifecycle,value,rating,*",`

Note: Do not forget the comma before the asterisk.

The filters are displayed like this:



If you want `rating` and `footprint` after `value`, a properly modified and configured `search.config` file would look as follows and this is the configuration file you upload to the Pulse web dashboard:

```
{
    "unicorn": {
        "version": 0,
        "important_comment": "Do not remove version field from the file while changing the file.",
        "others": [
            {
                "name": "facets",
                "display": "",
                "includes": "classification,lifecycle,value,rating, footprint,
                voltage,volt,tolerance,tol,current,curr,power,watt,wattage,material,type,name,pack_type,height,status,rohs,cost,price,*",
                "excludes": ""
            }
        ]
    }
}
```

3. Save and upload the modified configuration file.

   The configuration is updated and the changes are available and visible to all designers connected to this Pulse server.

Excluding Filters

Using the `search.config` file, you can exclude filters that you do not want to view as a filter option in part search. Modified `search.config` settings apply to all places where part search is displayed, such as Allegro X System Capture and the Pulse web dashboard.

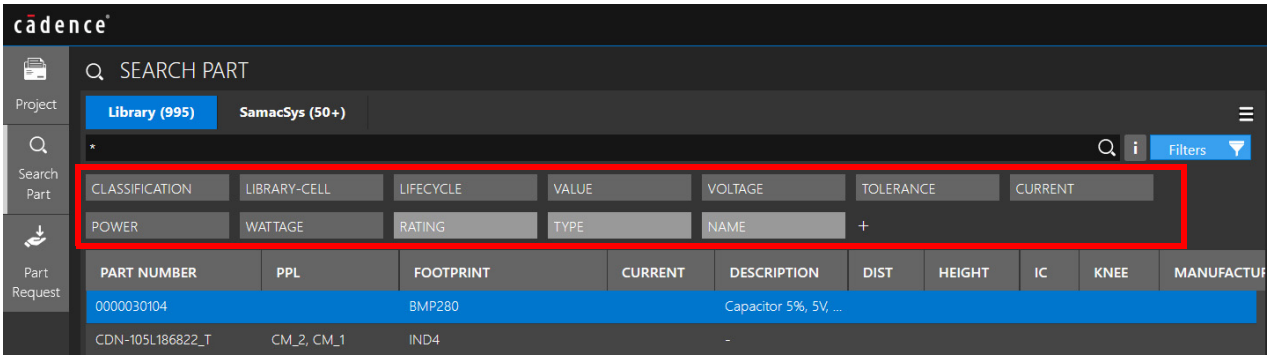
To exclude a filter, do the following:

1. Add the `facets` code snippet from the out-of-the-box `search.config` file to the downloaded configuration file.

   The following is the out-of-the-box sample:

   ```
   {
   ```

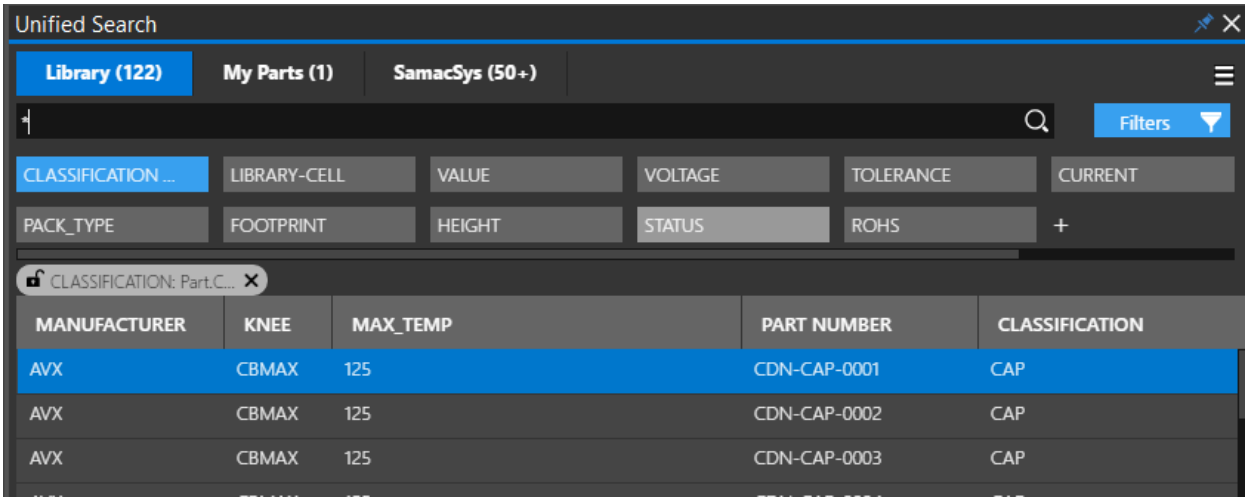
```
        "name": "facets",

        "display": "",

        "includes":
"classification,lifecycle,value,voltage,volt,tolerance,tol,c
urrent,curr,power,watt,wattage,rating,material,type,name,pac
k_type,footprint,height,status,rohs,cost,price,*",

        "excludes": ""

    }
```



2. In the downloaded file, type the names of the filters you want to exclude. For example, exclude `wattage` and `rating`.

```
{

        "name": "facets",

        "display": "",

        "includes":
"classification,lifecycle,value,voltage,volt,tolerance,tol,c
urrent,curr,power,watt,material,type,name,pack_type,footprin
t,height,status,rohs,cost,price,*",

        "excludes": "wattage,rating"

   }
```

Filters are now displayed as follows:



A properly configured `search.config` file with `wattage` and `rating` excluded as filters would look as follows and this is the configuration file you upload to the Pulse web dashboard:

```
{
    "unicorn": {
        "version": "1.0",
        "charsToEscape": "+-&|!(){}[]^~*?\\/=<>",
        "cloudApi": {
            "partsURL": "https://pcb.cadence.com/unifiedsearch",
            "partsURLForAutomation": "https://stagepcb.cadence.com/unifiedsearch"
        },
        "others": [
            {
                "name": "facets",
                "display": "",
                "includes":
                "classification,lifecycle,value,voltage,volt,tolerance,tol,current,curr,power,watt,material,type,name,pack_type,footprint,height,status,rohs,cost,price
                ,*",
                "excludes": "wattage, rating"
            }
        ]
    }
}
```

3. Save and upload the modified configuration file.

   The configuration is updated and the changes are available and visible to all designers connected to this Pulse server.

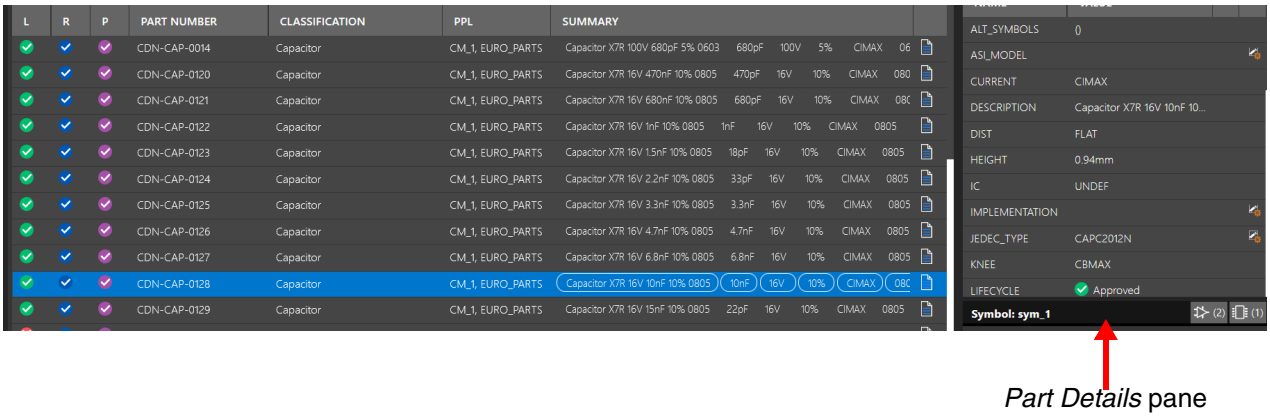
Configuring Contents of Summary Column in Part Search

Configuring the contents of the *Summary* column is controlled globally through the `search.config` file. Modified `search.config` settings apply to all places where part search is displayed, such as Allegro X System Capture and the Pulse web dashboard.

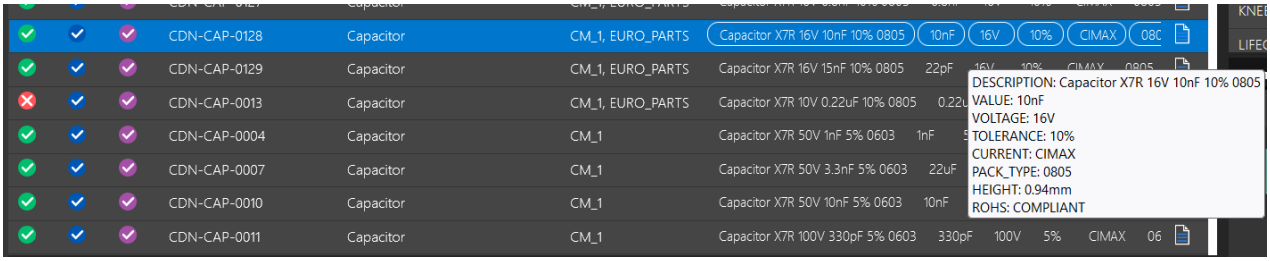
The *Summary* column is a concatenation of name-value pairs. For example:

- Name: `TOLERANCE`
- Value: `20%`

Instead of clicking a part to view its properties in the *Part Details* pane on the right, view part details using the *Summary* column. To view the name-value pairs of a part, you can also hover over the *Summary* column to read the tool tip.



Part Details pane



You can configure the *Summary* column for the following:

- if you have attribute names that differ from the out-of-the-box property names, such as `TOLERANCE` and `TOL`, or `VOL` and `VOLTAGE`, add these custom attribute names to the *Summary* column.
- define what you want to see in the *Summary* column.
- define the order of how the properties are displayed in the column.

To configure the *Summary* column, do the following:

1. Add the `summary` code snippet from the out-of-the-box `search.config` file to the downloaded configuration file.

   This is the out-of-the-box *Summary* section:

```
{
        "name": "summary",
        "includes":
"description,function_designation,value,voltage,volt,tolerance,t
ol,current,curr,power,watt,wattage,rating,material,type,name,pac
k_type,height,status,rohs,cost,price,",
        "excludes": ""
      }
```

2. Modify the `summary` section in the downloaded `search.config` file as required.

   For example, if you have an attribute in the library called `PLM_STATE`, add it to view it in the *Summary* column:

```
{
        "name": "summary",
        "includes":
"description,function_designation,value,voltage,volt,tolerance,t
ol,current,curr,power,watt,wattage,rating,material,type,name,pac
k_type,height,status,rohs,cost,price,plm_state",
        "excludes": ""
      }
```

   You can also use the asterisk symbol as a placeholder for everything after *<N>*. Define the properties you always want to view in a particular order, then use the asterisk. All the other properties are displayed in alphabetical order. If you delete the asterisk, the only properties displayed in the *Summary* column are the ones explicitly specified in the list.

   A properly configured `search.config` file with `plm_state` added would look as follows and this is the configuration file you upload to the Pulse web dashboard:

```
{
    "unicorn": {
        "version": 0,
        "important_comment": "Do not remove version field from the file while changing the file.",
        "others": [
            {
                "name": "summary",
                "includes":
                "description,function_designation,value,voltage,volt,tolerance,tol,current,curr,power,watt,wattage,rating,material,type,na
                me,pack_type,height,status,rohs,cost,price,plm_state",
                "excludes": ""
            }
        ]
    }
}
```

3. Save and upload the modified configuration file.

   The configuration is updated and the modified *Summary* column is available and visible to all designers connected to this Pulse server.

Enabling Wild Card or Regular Expressions in Part Search

Enabling wild card or regular expressions is controlled globally through the `search.config` file. Modified `search.config` settings apply to all places where part search is displayed, such as Allegro X System Capture and the Pulse web dashboard.

You can enable support for wild cards or regular expressions in search queries if Allegro X System Capture designers want to create such search queries in part search.

To enable support for wild cards or regular expressions, do the following:

1. Add the `"charsToEscape"` code snippet from the out-of-the-box `search.config` file to the downloaded configuration file.

   Here is the default code snippet from the out-of-the-box file:

   ```
   "unicorn": {
           "version": "1.0",
           "charsToEscape": "+-&|!(){}[]^~*?\\/=<>",
   ```

2. In `"charsToEscape"`, list the characters that are escaped by default.

   To use any of these characters as wild cards or in regular expressions, delete them from the downloaded configuration file.

   For example, if you want to use the asterisk and the question mark symbols as wildcards in part search, delete them from this list. A properly formatted file would look as follows:

```
{
    "unicorn": {
        "version": "1.0",
        "charsToEscape": "+-&|!(){}[]^~\\/=<>",
        "cloudApi": {
            "partsURL": "https://pcb.cadence.com/unifiedsearch",
            "partsURLForAutomation": "https://stagepcb.cadence.com/unifiedsearch"
        }
    }
}
```

Important

Colon (:) and double quote (“) cannot be escape characters because they have a special meaning. Colon is a separator for field names and values and the double quote is used for exact matches.

Note: You must use the capital "AND" "OR" for and/or conditions.

If you modify the list of characters under this section, ensure that the removed characters are not in the library data set.

3. Save and upload the modified configuration file.

   The configuration is updated and is available to all designers connected to this Pulse server.

Related Topic

Using Global Configuration File

Configuring Unified Search at Global or Project-Specific Level

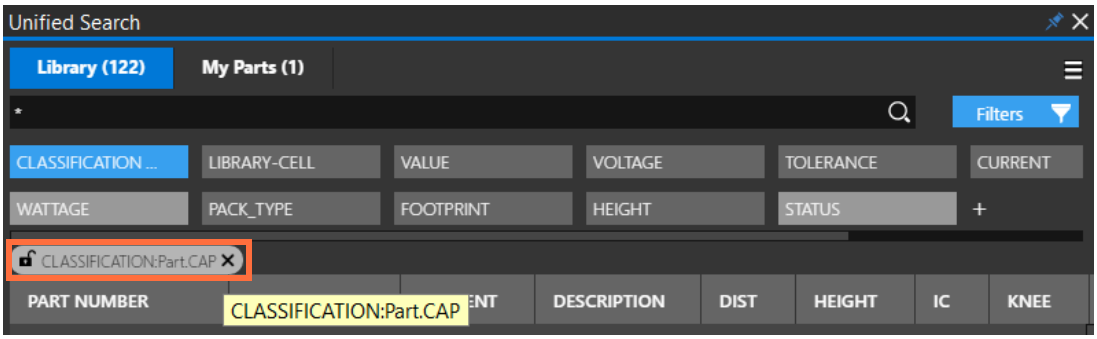
You can configure part search at a global or project-specific level by doing the following:

- Locking Search Filters
- Configuring Search Results Display

These tasks can be done through Allegro X System Capture and the Pulse web dashboard.

Locking Search Filters

A chiplet is an applied filter. It is a combination of a filter and its value. For example, a combination of *CLASSIFICATION* and *CAP* is a chiplet.



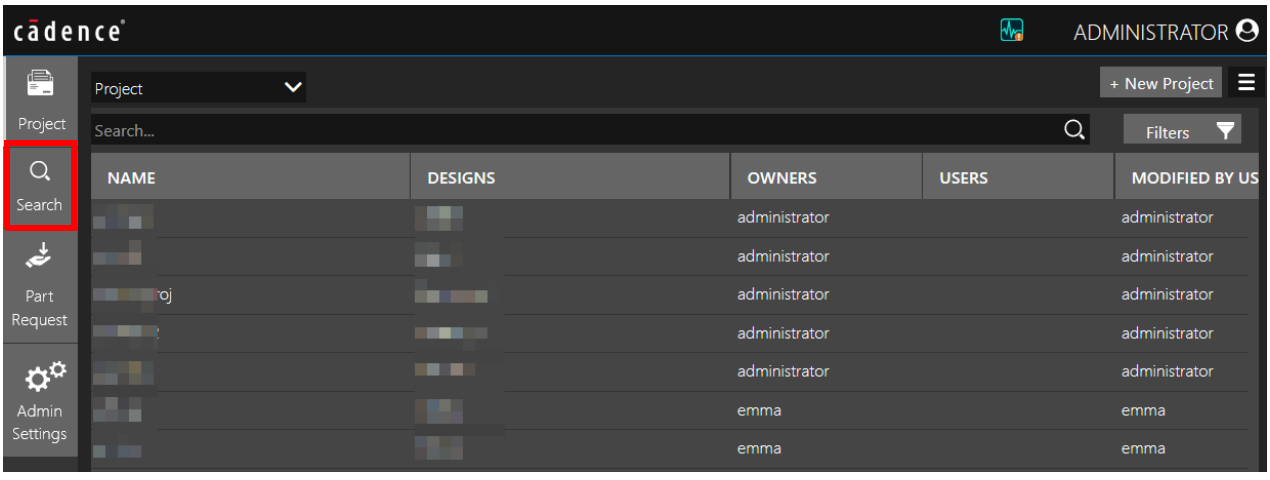
You can lock a chiplet to restrict designers from removing it and to make it a mandatory search criterion for all designers connected to a particular Pulse server. You can also lock chiplets for a specific project.

To lock a chiplet globally, do the following:

1. Access the Pulse web dashboard by typing `http://<Pulse access URL>:7100/projects` in a web browser.

   The Pulse dashboard is displayed.

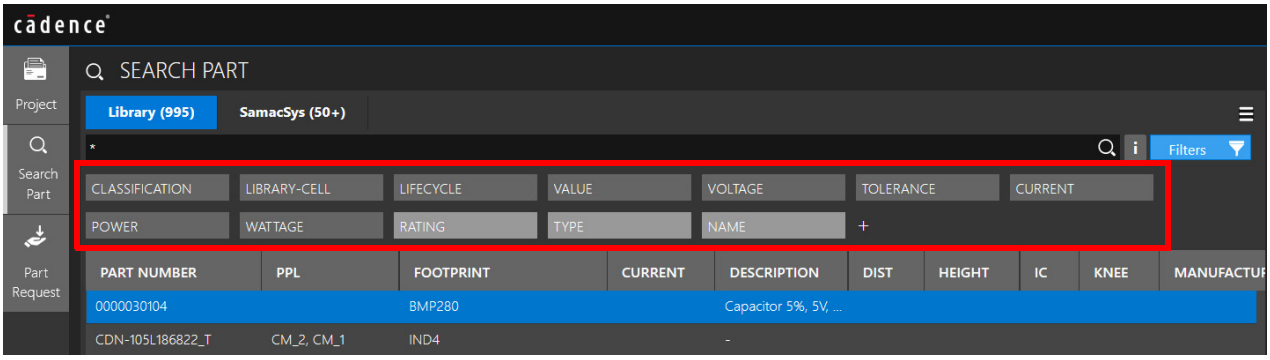
2. Click the *Search* tab on the left.



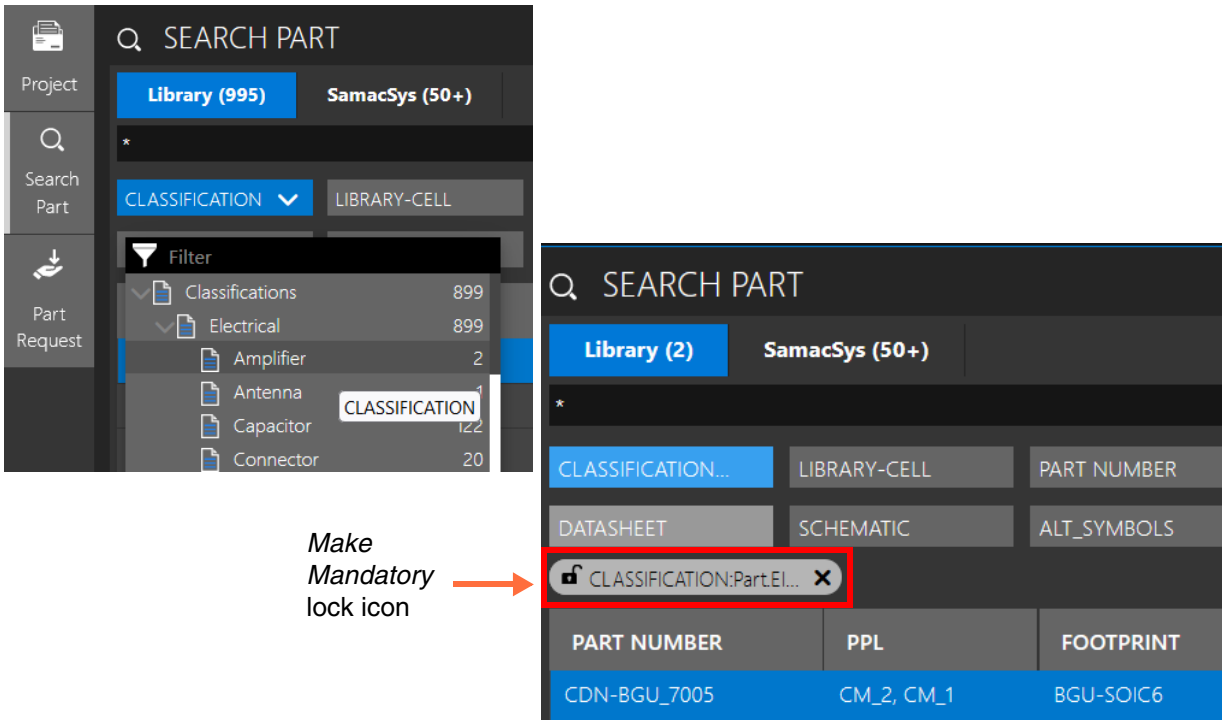
   The *Search* pane is displayed.

3. Click the *Filters* button on the top right of the part search pane.

The filters are displayed.



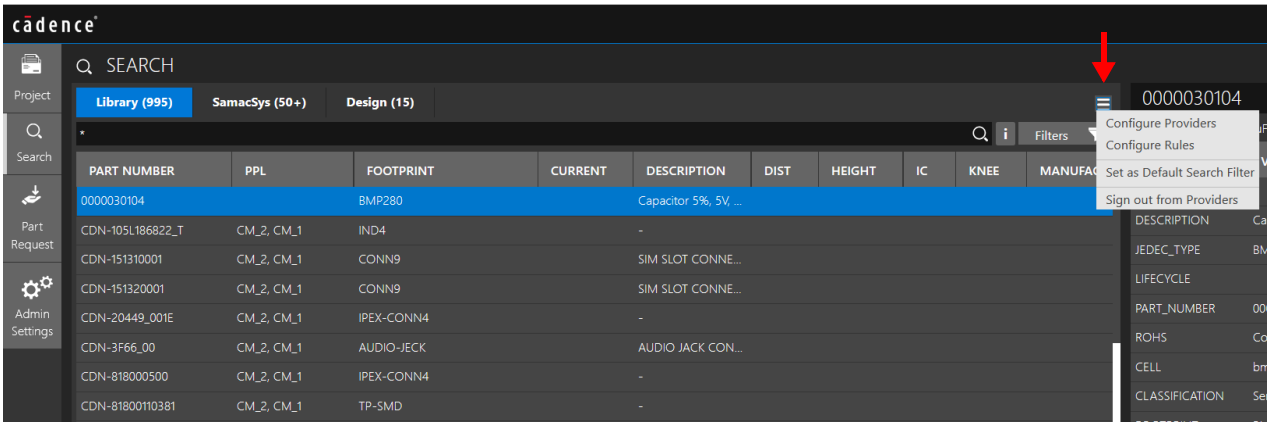
4. Click a filter. For example, click *CLASSIFICATION*.
5. Select a filter value. For example, select *Amplifiers*.



6. Click the *Make Mandatory* lock icon to restrict removal of a chiplet by designers.

   This makes the chiplet a mandatory search criterion.

7. Click the hamburger menu on the top right of the *Search* pane.



8. Select *Set as Default Search Filter*.

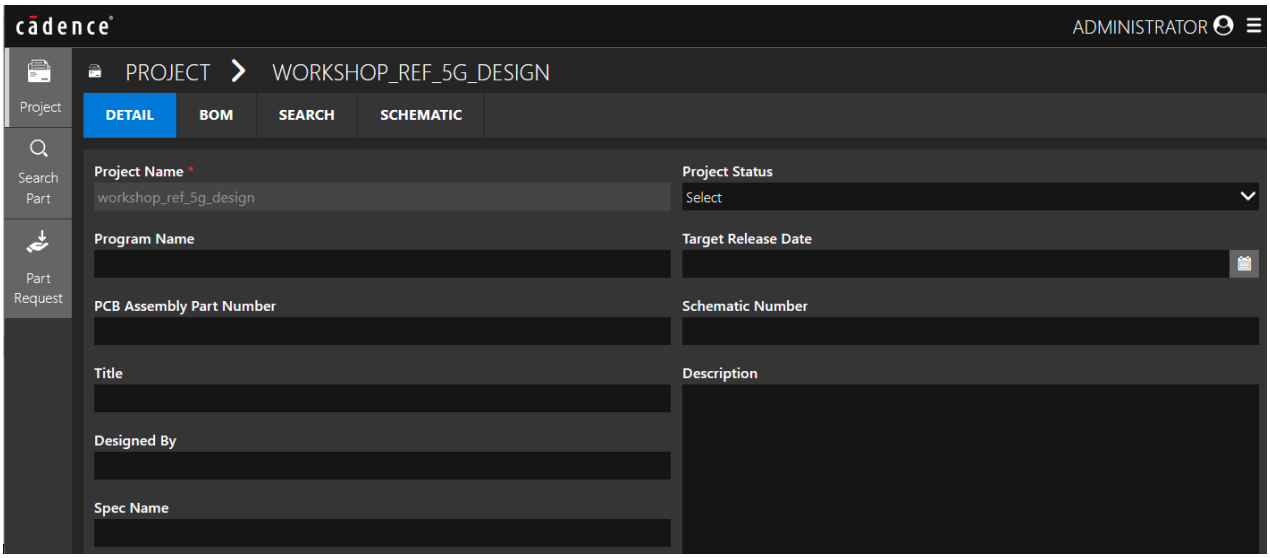
9. If you want to lock chiplets for a specific project, do the following:

a. Click the *Project* tab on the left in the dashboard.

b. Select a project from the project list.

c. Select a project.

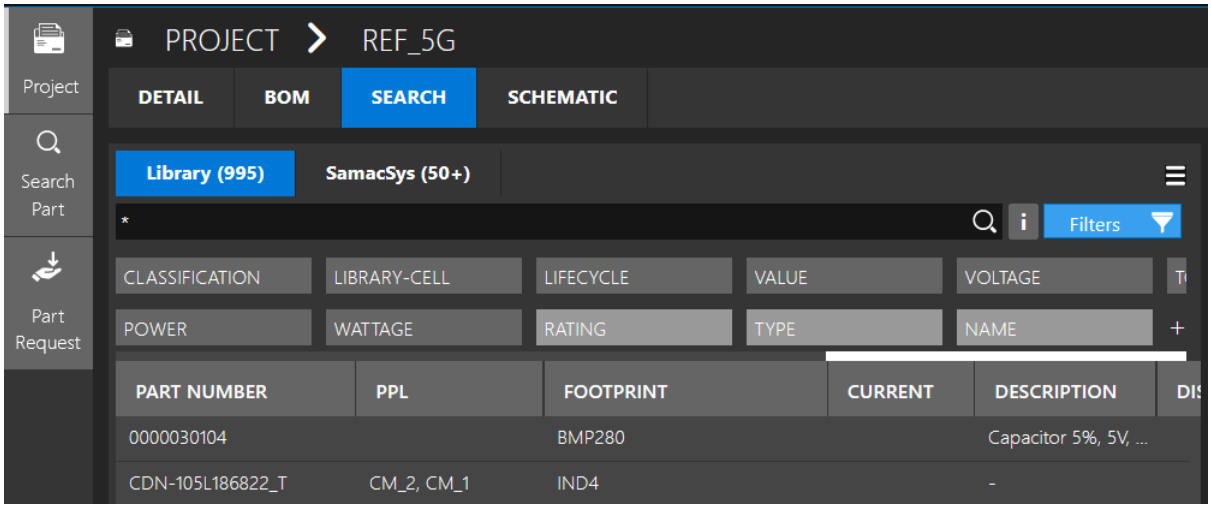
The project opens in a new tab.



10. Click the *SEARCH* tab.

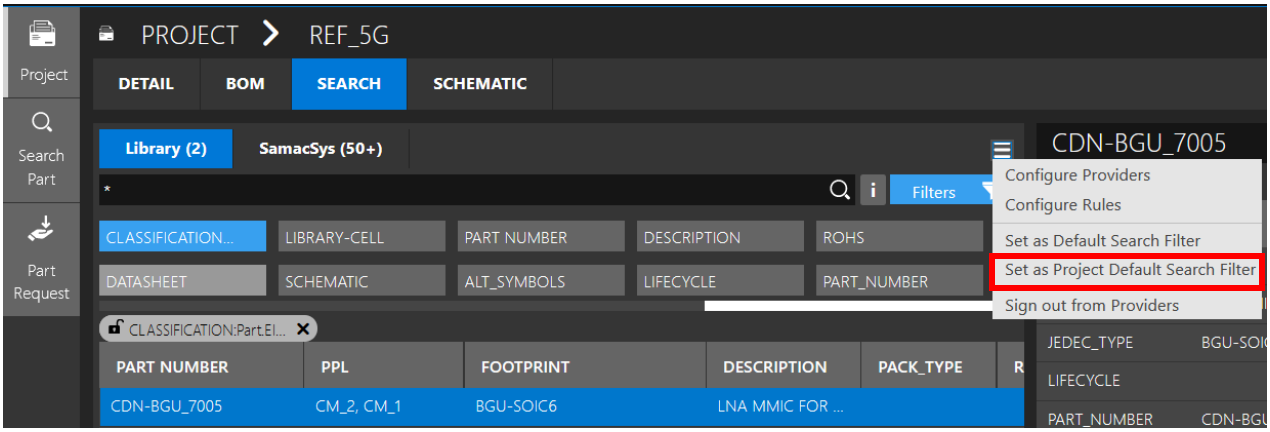
11. Click the *Filters* button on the top right of the part search pane.

The filters are displayed.



12. Follow steps 4 to 6.

13. Click the hamburger menu on the top right of the *Search* pane.



14. Select *Set as Project Default Search Filter*.

Related Topic

Resetting Pulse Web Dashboard Changes to Default

Configuring Search Results Display

You can configure how you want the part search results to be displayed by:

- setting column visibility
- modifying column widths
- defining the column order

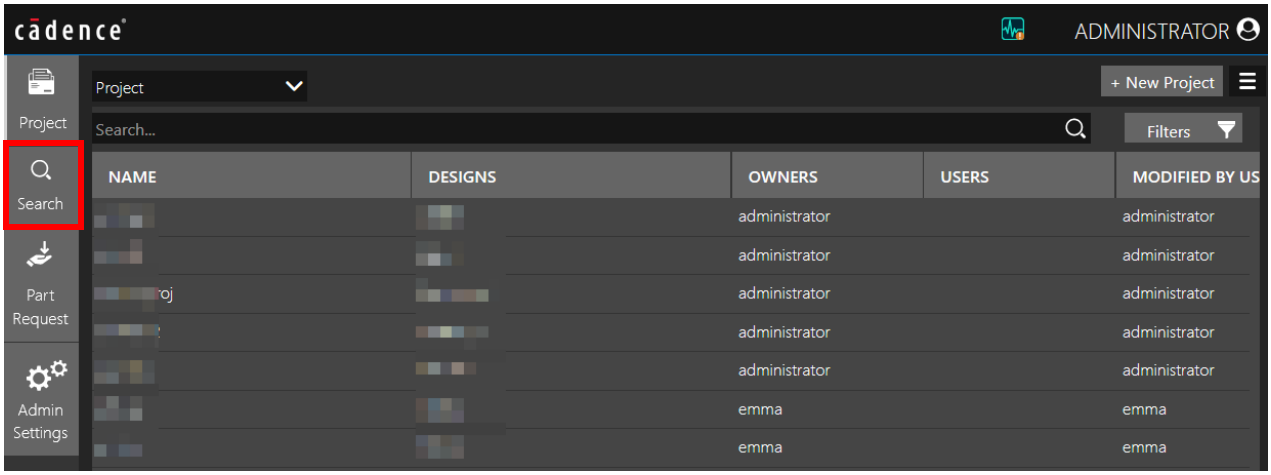
You can configure the display for a specific project or at a global level.

To configure the display of the search results globally, do the following:

1. Access the Pulse web dashboard by typing `http://<Pulse access URL>:7100/projects` in a web browser.

   The Pulse dashboard is displayed.

2. Click the *Search* tab on the left.

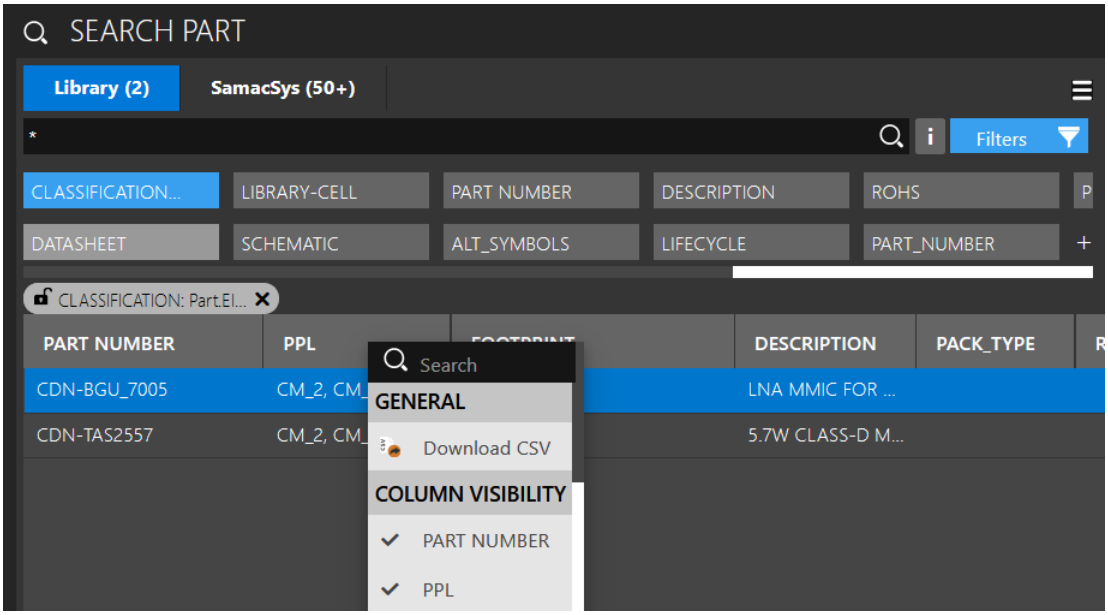


   The *Search* pane is displayed.

3. Apply a filter or specify keywords to search for parts.
4. Click *Enter* or use the magnifier icon to search.

   Part search results are displayed.

5. In the search results table, right-click the column header you want to hide and deselect it.

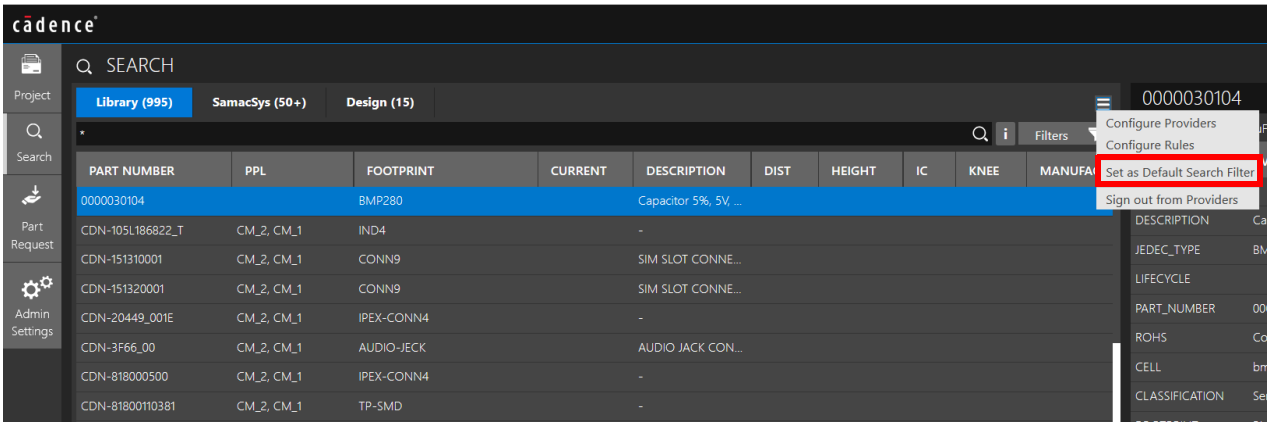


If you hide a column, it is hidden for all classifications, such as Capacitors, Connectors, Diodes, Amplifiers, and so on.

6. To modify a column width, click and drag a column header cell border until the column is the size you need.

7. To rearrange the order in which the columns are displayed, click and drag the selected column horizontally to the position that you want.

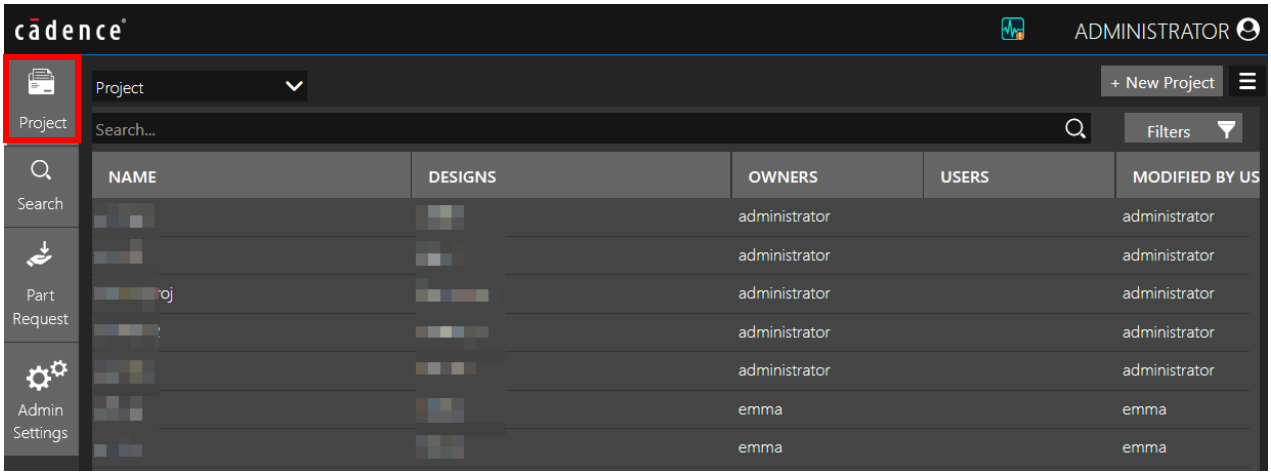
8. Click the hamburger menu on the top right of the *Search* pane.



9. Select *Set as Default Search Filter*.

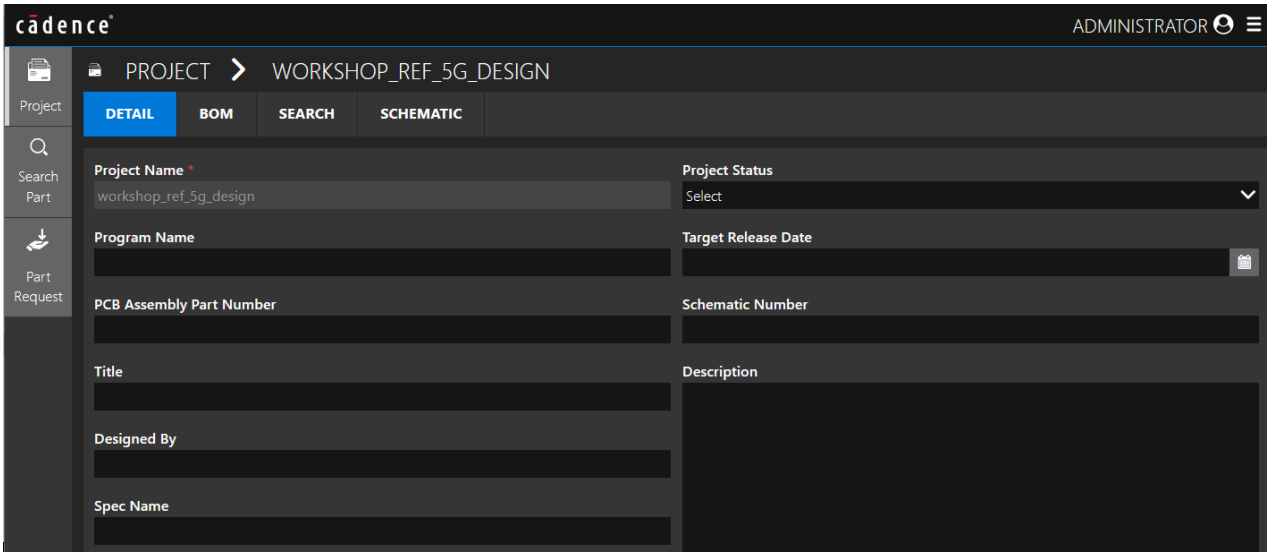
10. If you want to set column visibility, modify column widths, or define the column order for a specific project, do the following:

a. Click the *Project* tab on the left in the dashboard.



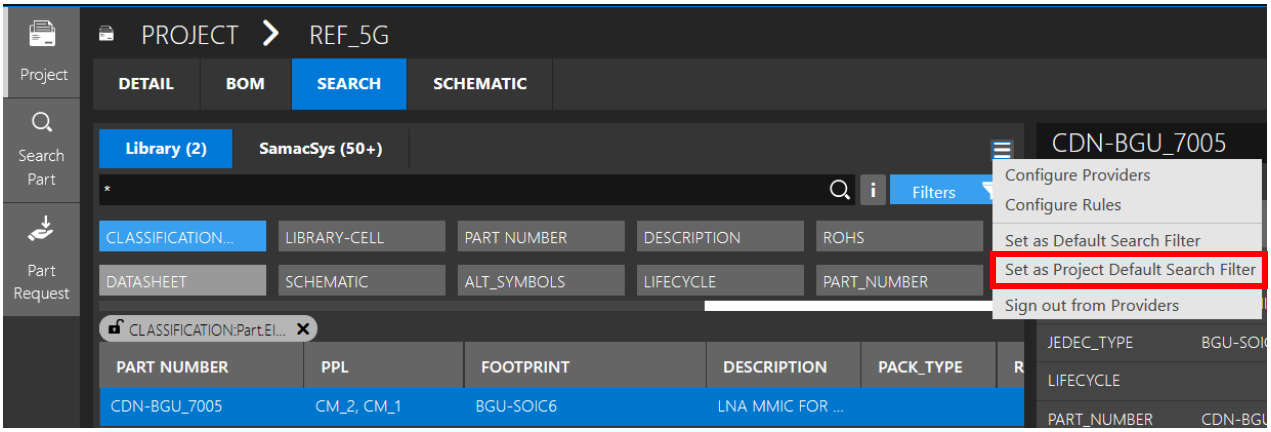
b. Select a project from the project list.

The project opens in a new tab.



c. Click the *SEARCH* tab.

d. Click the hamburger menu on the top right of the *Search* pane.



e. Select *Set as Project Default Search Filter*.

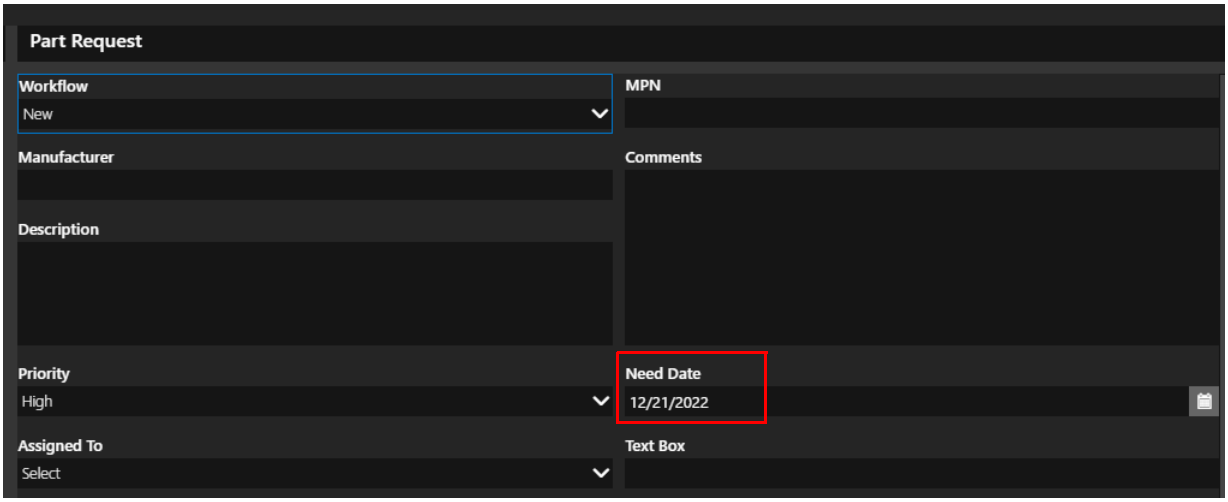
Related Topic

Resetting Pulse Web Dashboard Changes to Default

Configuring Date Format

Dates can be viewed in schematic page borders, the project and new part request lists, workflow notifications, project and new part request forms, and so on. The default date format in Pulse is `MM/DD/YYYY`.

The following image displays the date in the default format:



If you want to use a different format, do the following:

1. Copy the `others` code snippet from the out-of-the-box `search.config` file to the downloaded configuration file.

   Here is a sample `"others"` code snippet from the out-of-the box configuration file:

   ```
   "others": [
   {
   "name": "display_date_format",
   "display": "",
   "value": "mm/dd/yyyy"
   },
   ```

2. Modify the date format as required. For example, change it to `yyyy/mm/dd`.

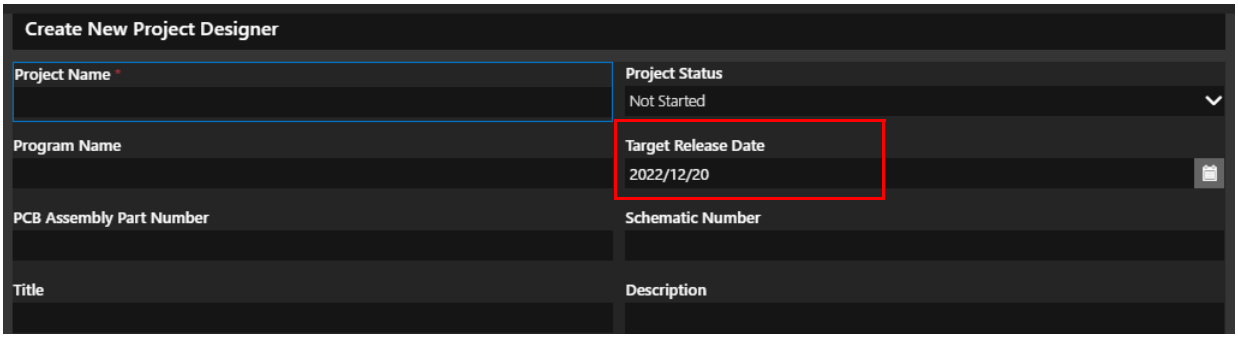
   The only supported formats are: `mm/dd/yyyy, yyyy/mm/dd, dd/mm/yyyy`

A sample of a modified and properly configured `search.config` would look as follows and this is the configuration file you upload:

```
{
    "unicorn": {
        "version": 0,
        "important_comment": "Do not remove version field from the
        "others": [
            {
                "name": "display_date_format",
                "display": "",
                "value": "yyyy/mm/dd"
            }
        ]
    }
}
```

3. Save and upload the modified configuration file.

   The date format changes.



Related Topic

Using Global Configuration File

Configuring Live BOM Headers

The bill of materials (BOM) provides a list of components, materials, and parts necessary to produce the board design, as well as other details. The Live BOM in Allegro X System Capture provides access to real-time data, such as library properties and the part life cycle state. Typically, parameters such as quantity, comment, description, reference designator, footprint, and so on are displayed.

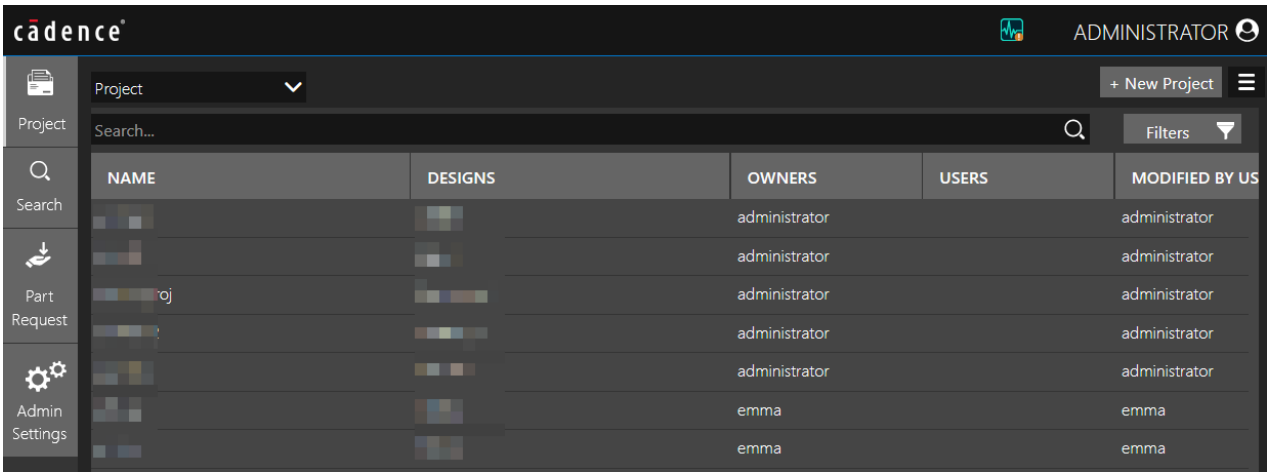
To analyze part usage more quickly, you can enable or display various parameters. For example, if you enable the display of tolerance, you can filter the Live BOM to view all resistors and their tolerance values. Live BOM header configuration is specific to a project. It cannot be set globally.

You can configure Live BOM headers using Allegro X System Capture or the Pulse web dashboard. Because administrators are more likely to use the dashboard, the instructions and images are for the Pulse dashboard.

To enable or disable the display of parameters in Live BOM, do the following:

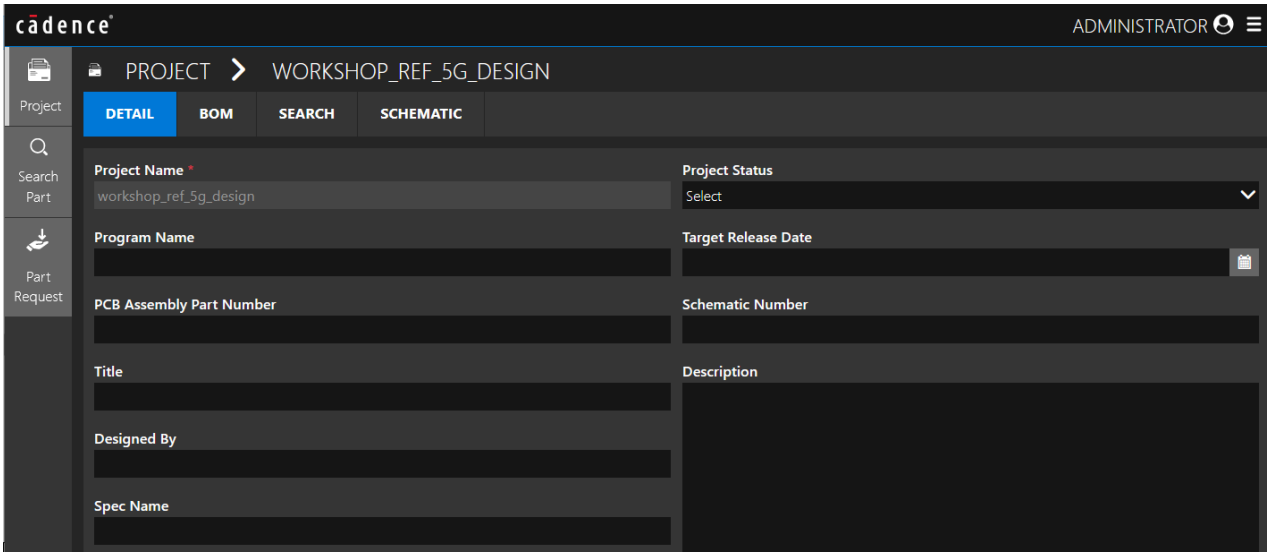
1. Access the Pulse web dashboard by typing `http://<Pulse access URL>:7100/projects` in a web browser.

   The Pulse dashboard is displayed.

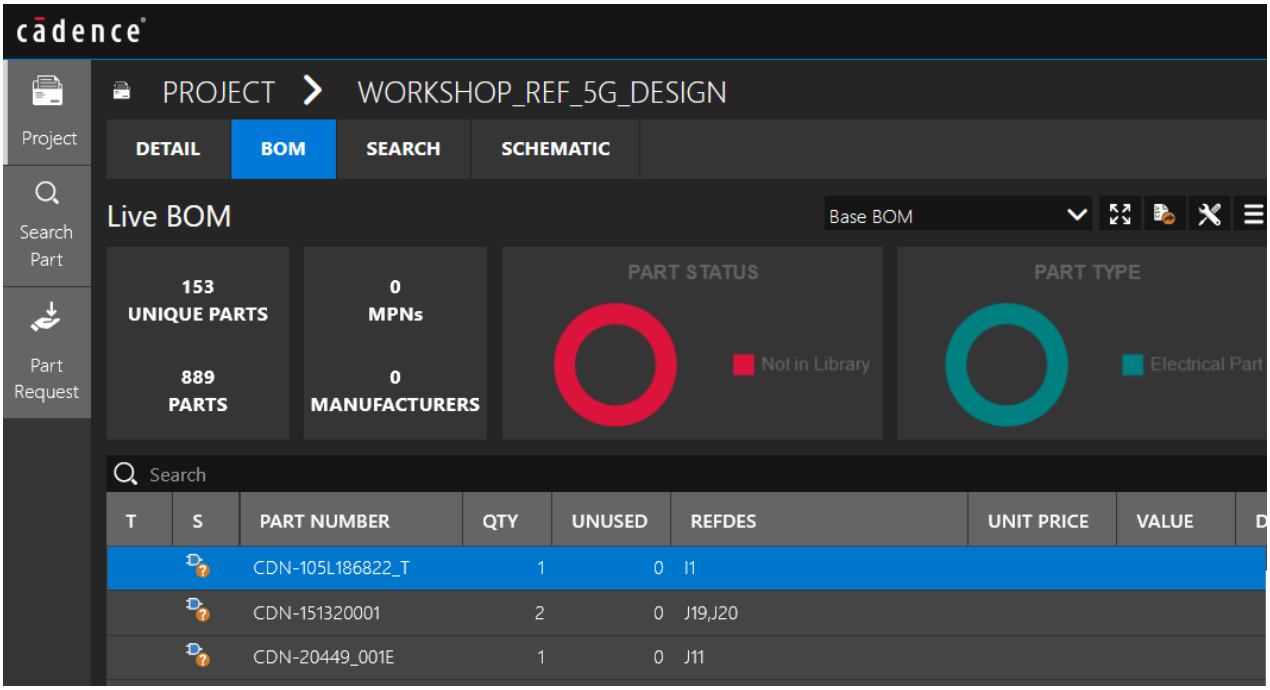


2. Select a project from the project list.

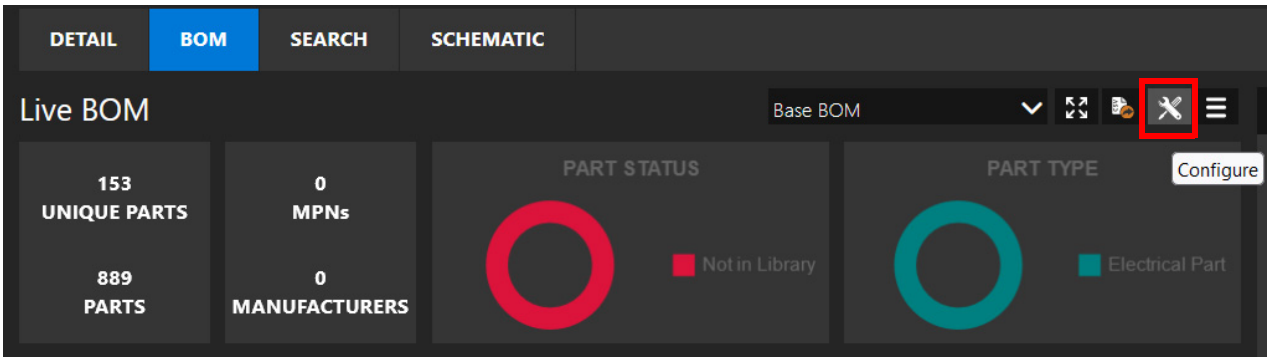
The project opens in a new tab.



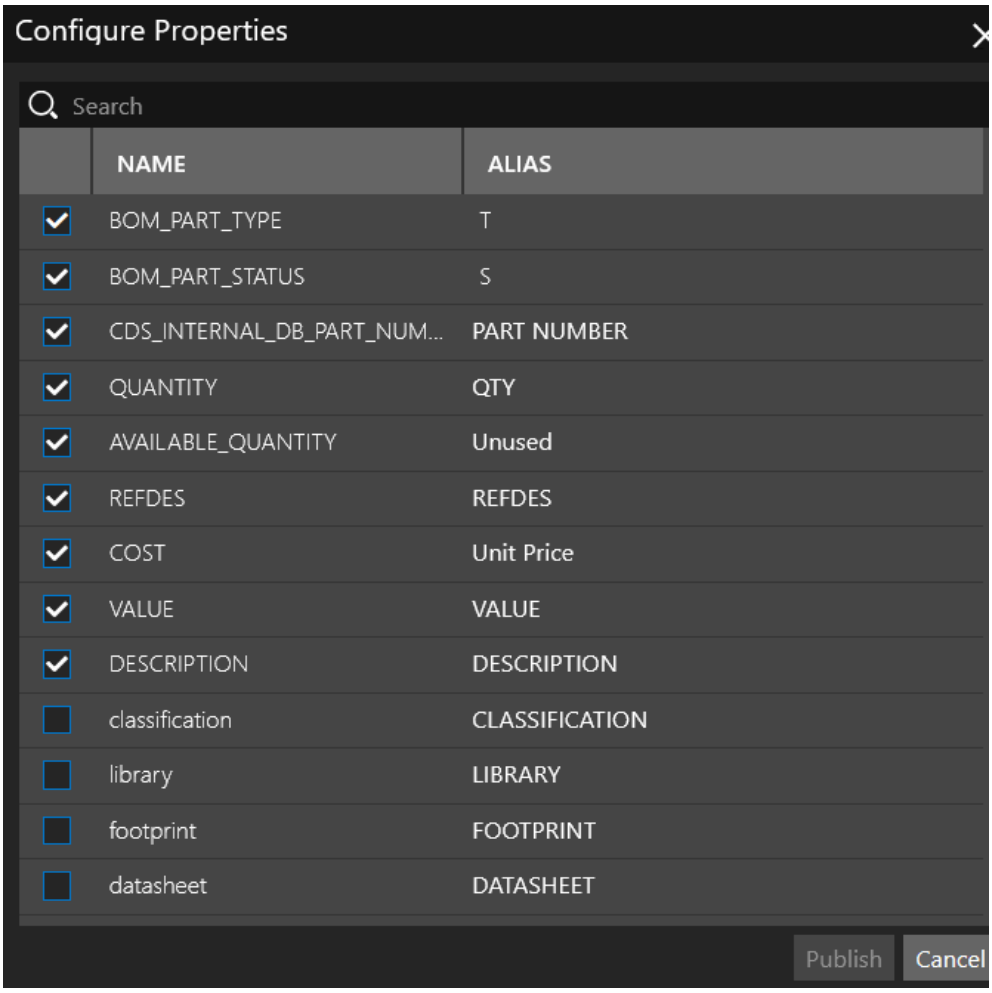
3. Click the *BOM* tab.



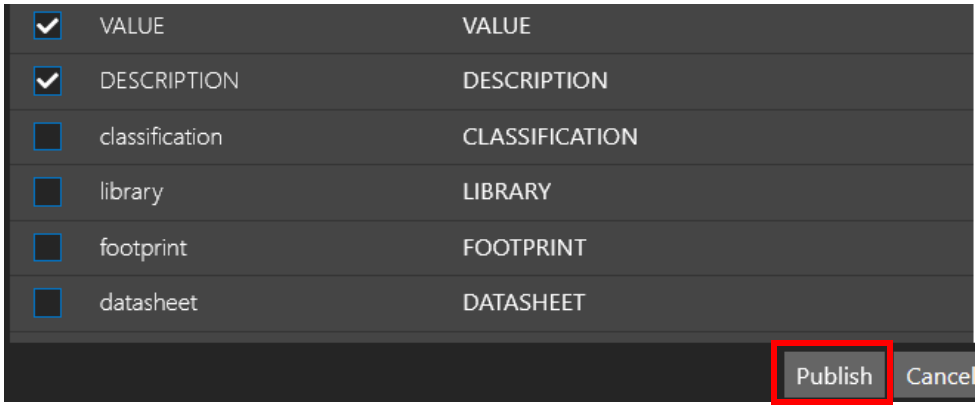
4. Click the *Configure* button in the top-right of the BOM pane to select which property headers should be displayed in the BOM display.



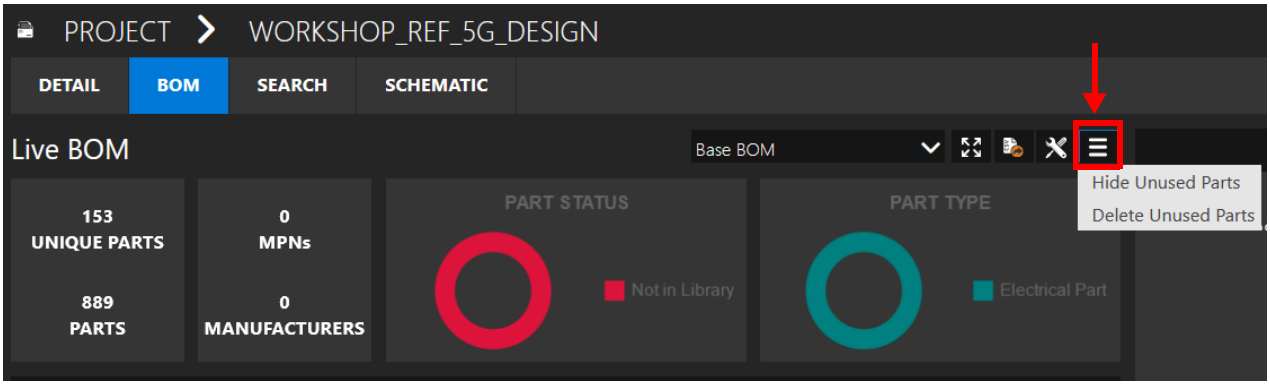
The Configure Properties dialog box is displayed.



5. Modify your selections as required.
6. Click *Publish*.



7. If you want to hide or delete unused parts from the BOM display, click the hamburger menu on the top right of the BOM pane.



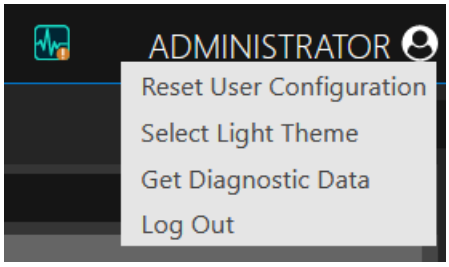
Resetting Pulse Web Dashboard Changes to Default

If you set any defaults related to filters, column widths, the light or dark theme, and so on, you can reset them back to the default. This does not affect settings that administrators or design project owners set as global defaults.

This option ensures that users can get back to the default or to the settings that the design owner or administrator set.

To reset to the default, do the following:

1. Click the profile icon next to the user name on the top right of the page.



2. Click *Reset User Configuration*.
3. Click *Yes* in the warning message that appears to reset all the settings back to their defaults.

   Pulse might take some time to reset back to the default and reload the application automatically. During this time, certain Pulse services might be down and part search might not work.

Note: Changes cannot be reset back to the default through the Allegro X System Capture application, They can only be reset using the Pulse dashboard.

Creating Project Templates

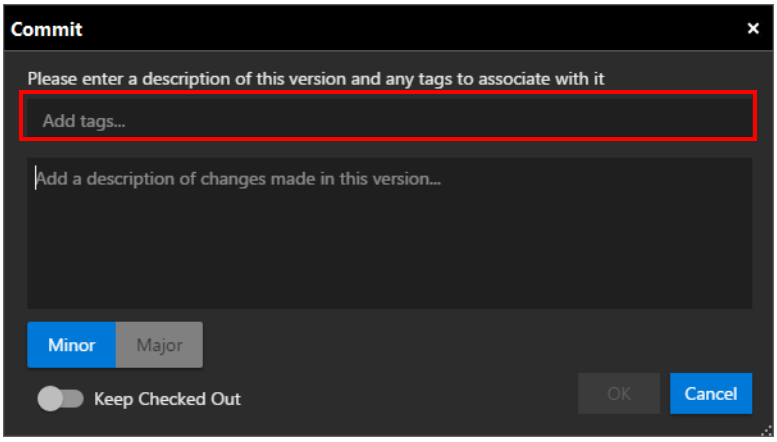
To include company standards for page borders, the TOC, block diagrams, or just to drive a consistent process, it is recommended that you provide some projects that can be used as a template by designers.

You can also include Live BOM content so that electrical parts added to the template Live BOM are preserved.

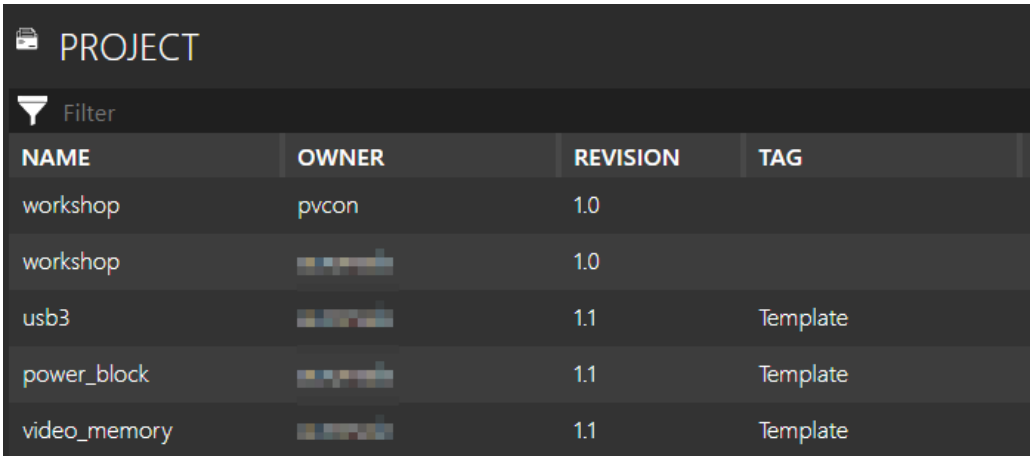
To create a project and define it as a template, do the following:

1. Launch Allegro X System Capture.
2. Click *New* in the Start Page to create a project.
3. Define the page border, TOC, and any other detail you want to include in the project as a template.
4. Add the keyword `template` when committing the design to the Pulse server.

The keyword is not case sensitive.

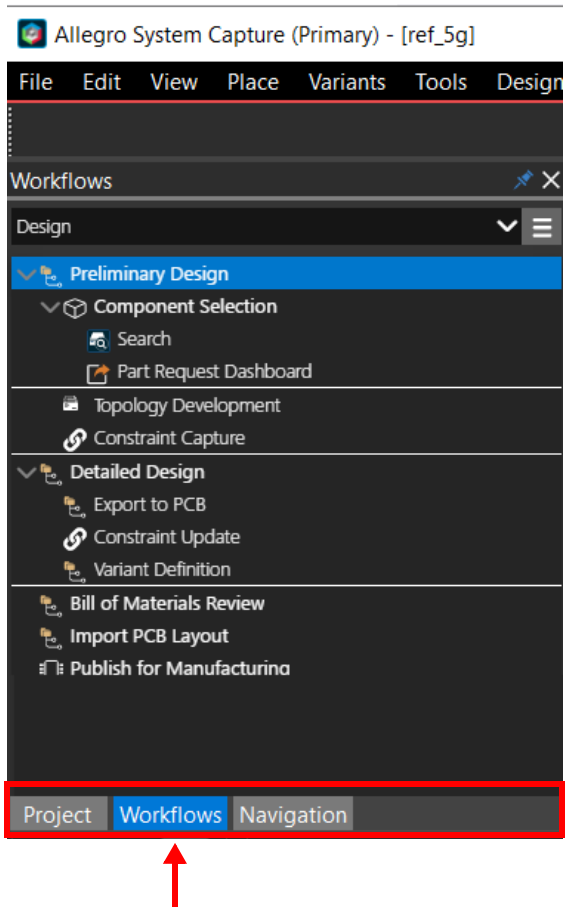


This project can now be used as designers to create template-based projects. The following image is from the Pulse dashboard with some projects marked as templates.



Configuring Workflows

To ensure that a design team follows a consistent methodology to create their designs, Pulse provides a default series of steps, referred as the *Design* workflow, to create an electronics design. This workflow is provided in Allegro X System Capture.



You can create a customized series of sequential tasks mapped in a way that the process and its associated activities and assets move seamlessly from step to step. Using these worklows, System Capture designers can follow the steps that guide them to bring a project from start to finish.

Out-of-the-box, default workflows cannot be customized. To customize, you must first make a copy of a default workflow after which editing options are available.

The default, and any customized workflows, are available to designers only in the following tiers of the multi-user environment: Managed Library and Workflow and Library and Adhoc Team Design tiers.

See the following:

- Creating Custom Workflows
- Configuring a Newly Created Workflow
- Disabling Workflow Selection by Non-Administrator Users

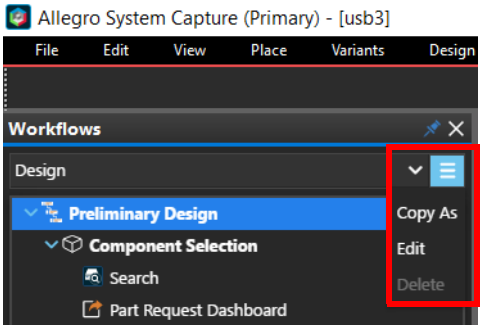
Creating Custom Workflows

To create a custom workflow, do the following:

1. Launch the required System Capture project.
2. Log in with administrator credentials.

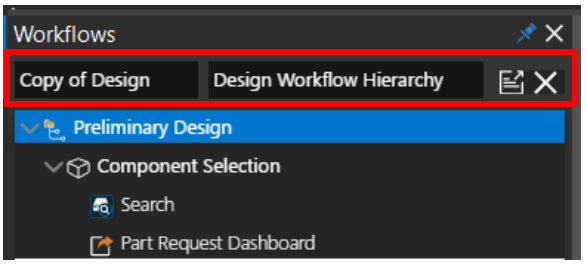
   The default user name and password are `administrator` and `pwd`. If you changed the administrator password in the *Security* tab of Pulse Service Manager, use that password.

   After you log in, editing options are available in the *Workflow* pane.
3. Click the hamburger button on the right of the pane to access the editing options.

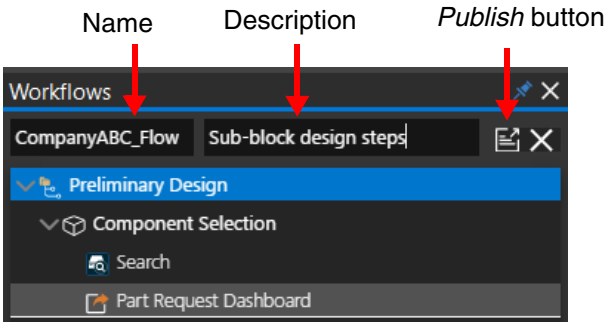


4. Select *Copy As*.

A copy of the selected workflow is created. Making a copy of an existing workflow is required because out-of-the-box workflows cannot be edited.



5. Type a name for the new flow. For example, `CompanyABC_Flow`.



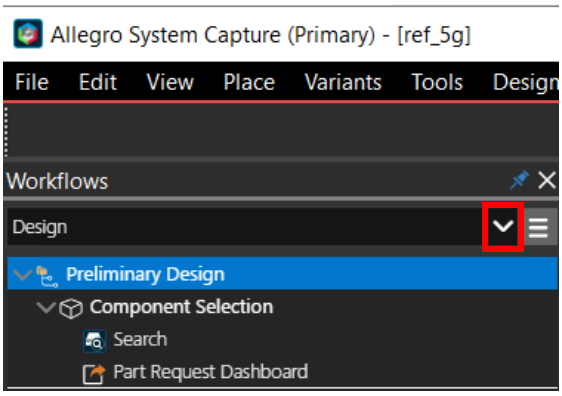
6. Type a description for the new flow, if needed.
7. Click the *Publish* button.

   The new flow is saved and is available to all designers connected to this Pulse server.

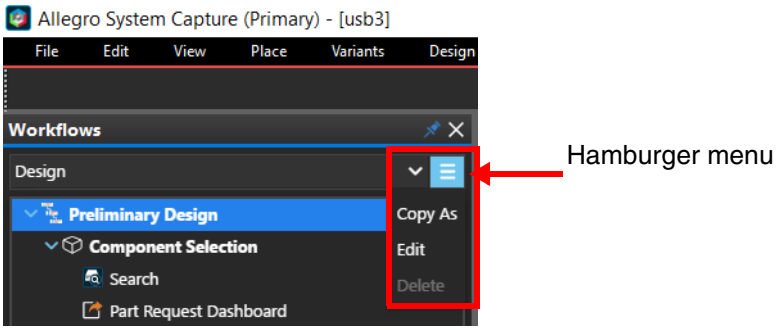
Configuring a Newly Created Workflow

To configure any custom workflow, do the following:

1. Select the flow that you want to configure from the drop-down list in the *Workflows* field.

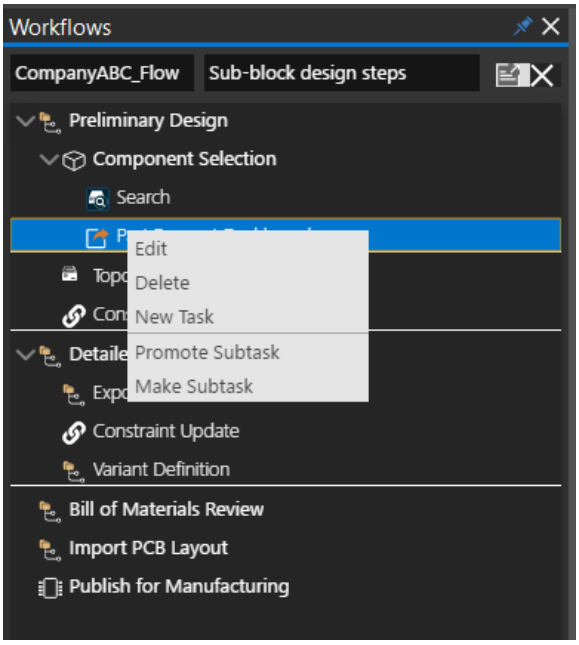


2. Click the hamburger button.



3. Select *Edit.*

4. Right-click any of the default tasks in the copied flow.



Creating a Task for a Custom Workflow

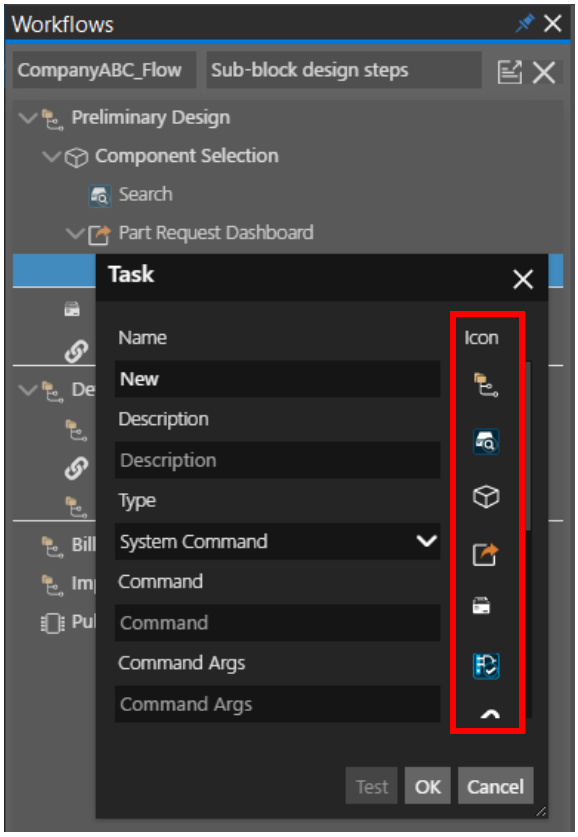
To understand the steps to create a new task, let us consider a scenario. Add a step that enables access to process documentation that your company has developed in a Wiki page or other internal sites.

Or for example, add steps for a migration flow that enables designers to bring legacy designs into Allegro X System Capture.

To add the required steps and buttons, do the following:

1. Complete steps 1 to 4.
2. Select *New Task*.

The *Task* dialog box is displayed.



3. Specify a name and description for the task.
4. Select between System Command or Tcl command for the new task.
5. Specify the command and the required arguments.

   All standard Windows system commands are supported.
6. On the right side of the dialog box, click the icon you want to assign to the new task.
7. Do one of the following:
   - ❑ Click *Test* to check whether the new task works as expected.
   - ❑ Click *OK*.

Customizing Flow Steps and Buttons

You can select a custom task button for a workflow, and define programmatic behavior for buttons and menu items. Programming, in this context, refers to the actions you can add to a button event, such as a click.

To customize a flow step and button, do the following:

1. Complete steps 1 to 4.
2. Do any of the following to:
   - ❑ edit the a task, select *Edit*.
   - ❑ delete the selected task, select *Delete*.
   - ❑ to create a new task for the custom flow, select *New Task*.
   - ❑ If you right-clicked a child task, you can promote it and make it a parent task by selecting *Promote Subtask*.
   - ❑ If you promoted a task and made it a parent task but want to make it a subtask again, select *Make Subtask*.

Disabling Workflow Selection by Non-Administrator Users

If you have customized workflows but want to restrict designers, including design owners, from selecting anything other than the default workflow, do the following:

1. Copy the `"workflow"` code snippet from the out-of-the-box `search.config` file to the downloaded configuration file.

Some sample lines from the `"workflow"` code snippet from the out-of-the-box `search.config` file are as follows:

```
            {
                "name": "generic_workflow",
                "object_type": "",
                "can_modify" : "",
                "tools" : [
                    {
                        "name" : "flowmanager",
                        "value": "generic_workflow"
                    }
                ]
            },
            {
                "name": "project_workflow",
                "object_type": "project",
                "can_modify" : "",
                "tools" : [
                    {
                        "name" : "flowmanager",
                        "value": "project_workflow"
                    }
                ]
            }
```

2. Change the `"can_modify"` value to `Administrator`.

   This is the only supported role for this field and it is a case-sensitive value.

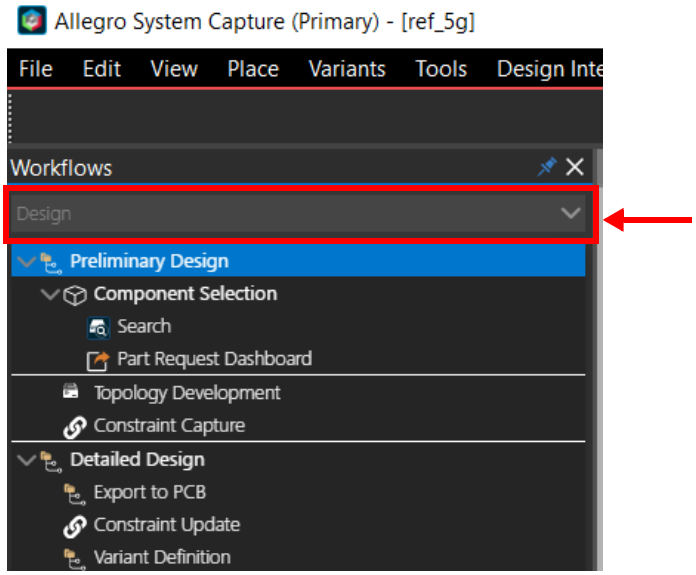
   A sample of a modified and properly configured `search.config` would look as follows and this is the configuration file you upload to the Pulse web dashboard:

```
{
    "unicorn": {
        "version": 0,
        "important_comment": "Do not remove version field from the file while changing the file.",
        "workflow": [
            {
                "name": "schematic_workflow",
                "object_type": "schematic",
                "can_modify": "Administrator",
                "tools": [
                    {
                        "name": "syscap",
                        "value": "design_workflow"
                    }
                ]
            }
        ]
    }
}
```

   In this sample, selection of the `schematic_workflow` has been disabled for non-administrator users.

3. Save and upload the modified configuration file.

   The configuration is updated and the *Workflows* drop-down option is disabled.



Related Topic

Using Global Configuration File

Working with Notifications

You can notify designers who are logged in to the Pulse server, which hosts the Cadence installation hierarchy, about important changes, such as the installation of the latest hotfix. For example, you can also request designers to let you know of issues they face with as a result of the upgrade.

Notifications are only visible in the web dashboard. They are not visible in System Capture or PCB Editor.

See the following:

- Enabling Notifications
- Sending Notifications to Designers
- Modifying Notifications

Enabling Notifications

To enable notifications, do the following:

1. Add the following lines from the out-of-the-box `search.config` file to the downloaded configuration file:

```
{
                    "name": "notifications",
                    "display": "",
                    "value": "false"
}
```

   The default value for notifications is `false`.

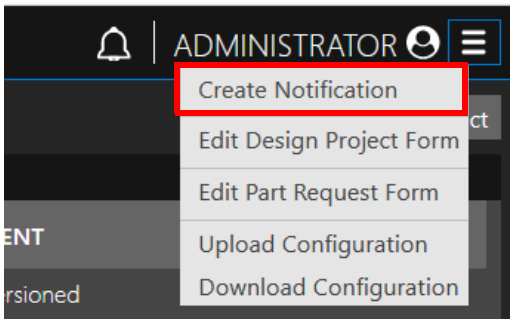
2. Ensure that the value is set to `true` in the downloaded `search.config` file.

   A properly configured `search.config` file would look as follows and this is the configuration file you upload to the Pulse web dashboard:

```
{
    "unicorn": {
        "version": 0,
        "important_comment": "Do not remove version field from the file while changing the file.",
        "others": [
            {
                "name": "notifications",
                "display": "",
                "value": "true"
            }
        ]
    }
}
```

3. Save and upload the modified configuration file.

The configuration is updated and you see a *Create Notification* option in the settings menu.



Related Topic

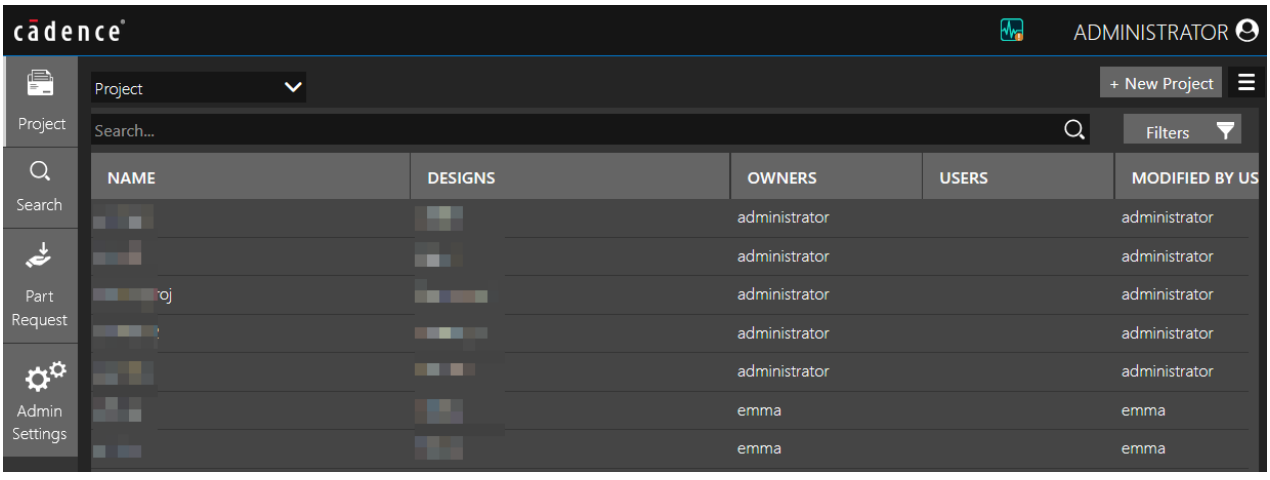
Using Global Configuration File

Sending Notifications to Designers

To notify designers about important changes, do the following:

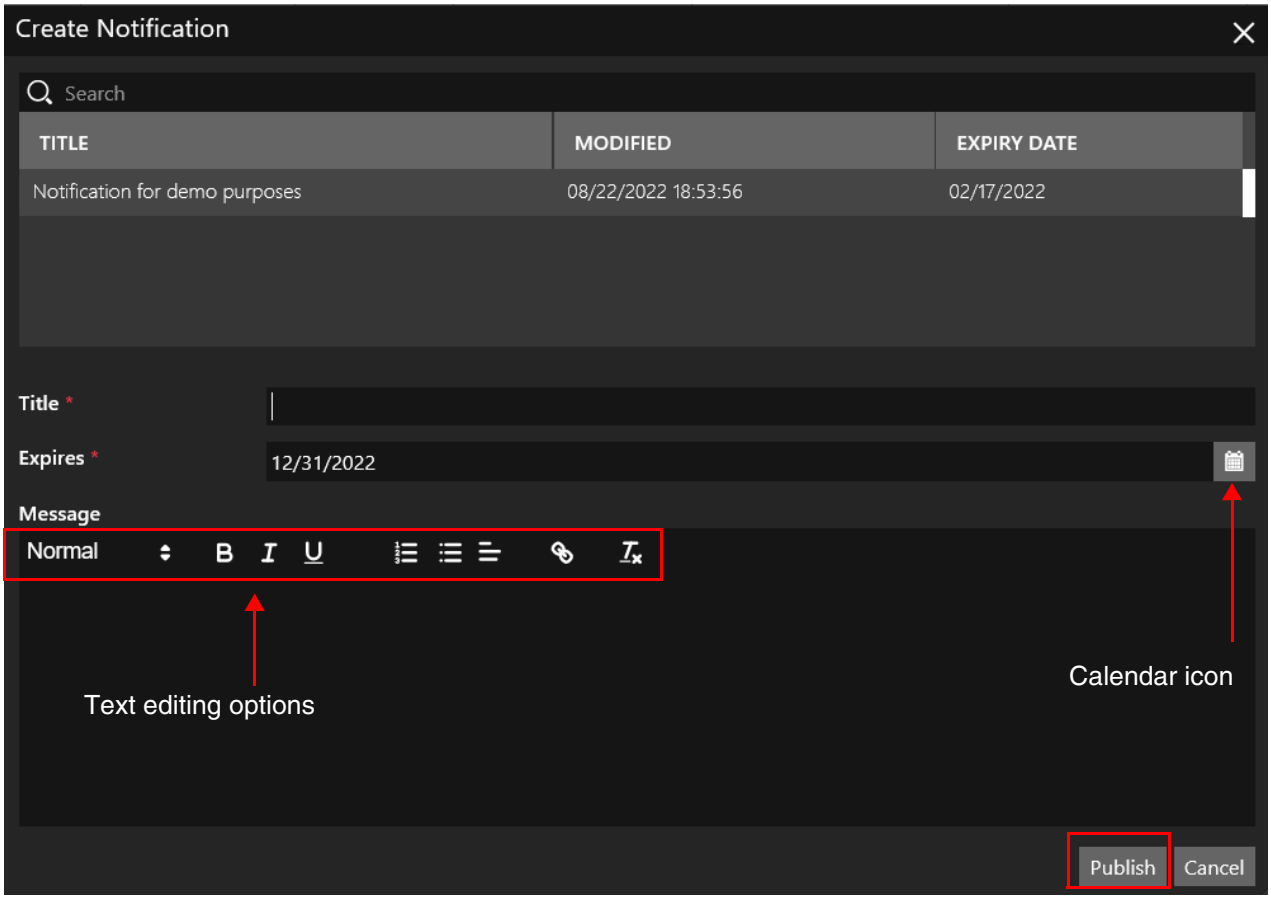
1. Access the Pulse web dashboard by typing `http://<Pulse access URL>:7100/projects` in a web browser.

   The Pulse dashboard is displayed.



2. Click *Admin Settings*.
3. Select *Create Notification*.

The Create Notification dialog is displayed.



4. Specify a title for the notification and use the calendar icon to choose its expiry date.

   You can use the editing options under *Message* to modify the font, font size, text color, and so on.

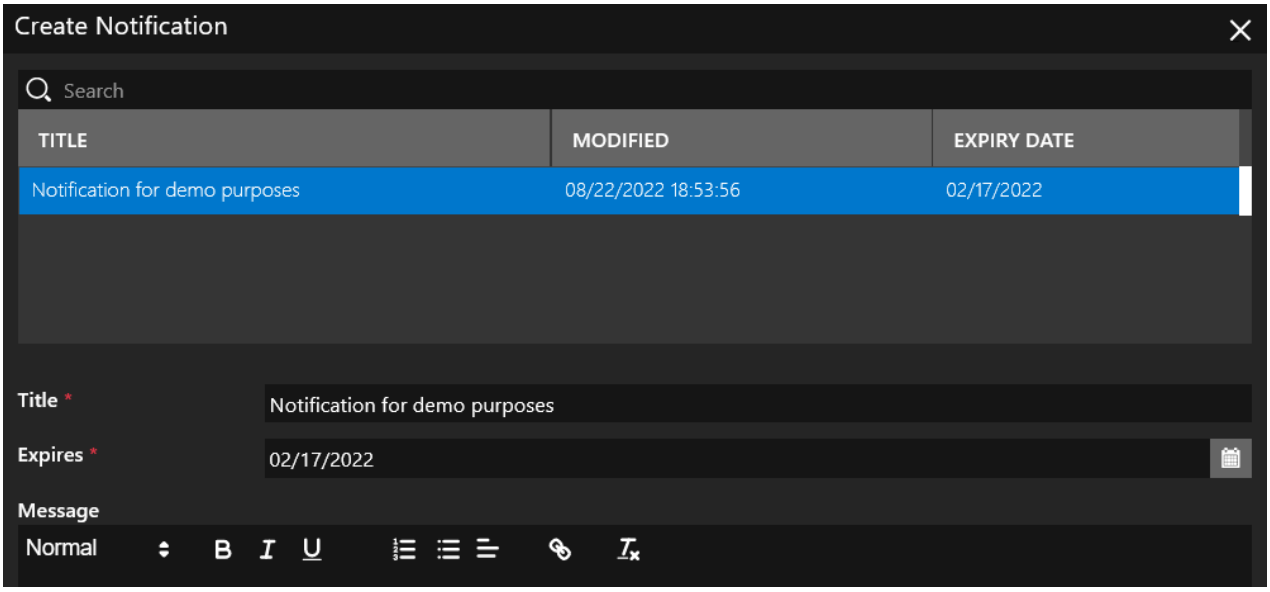
5. Click *Publish* for the notification to be visible to all designers connected to a particular Pulse server.

Modifying Notifications

As the author of a notification, you can modify the title, expiry date, and message of existing notifications by doing the following:

1. Follow steps 1 to 3.

2. Select the notification you want to modify from the top pane.



3. Modify the title, expiry date, and or the message.
4. Click *Publish* after making the required changes.

Additional Object Types in the Allegro EDM-Managed Library Database

Some library management features are available out-of-the-box but must first be enabled based on your requirements. For example, features such as support for the Unicode character set, the import of OrCAD Capture libraries into Allegro EDM and Pulse, new part requests.

The object types and options you can enable in the component database are as follows:

- Option to link a datasheet specification for a footprint
- Option for Allegro EDM Database Editor to write and store all electrical part numbers in uppercase letters in the component database
- Support for mixed case in PTF properties
- Support for the Unicode character set

  This is only available on the Windows platform. Enable support for the Unicode character set to specify non-ECAD properties in Allegro EDM in languages such as Chinese, Japanese, Korean, and so on.

Important

For Allegro EDM to support Unicode characters, you must ensure that the files you import into Allegro EDM using Data Exchange, Library Import, or Database Editor are in UTF-8 format.

All Unicode characters are supported in EDM except for the following:

- Double quotes (")
- Single quotes (')
- Comma (,)
- Asterisk (*)
- Question mark (?)

- Part requests

  This option enables Allegro X System Capture designers working in a Pulse server-connected environment to submit part requests to librarians. Librarians can then work with these requests and accept, reassign them, decline, and so on. When enabled, part request menu options are displayed in Allegro X System Capture.

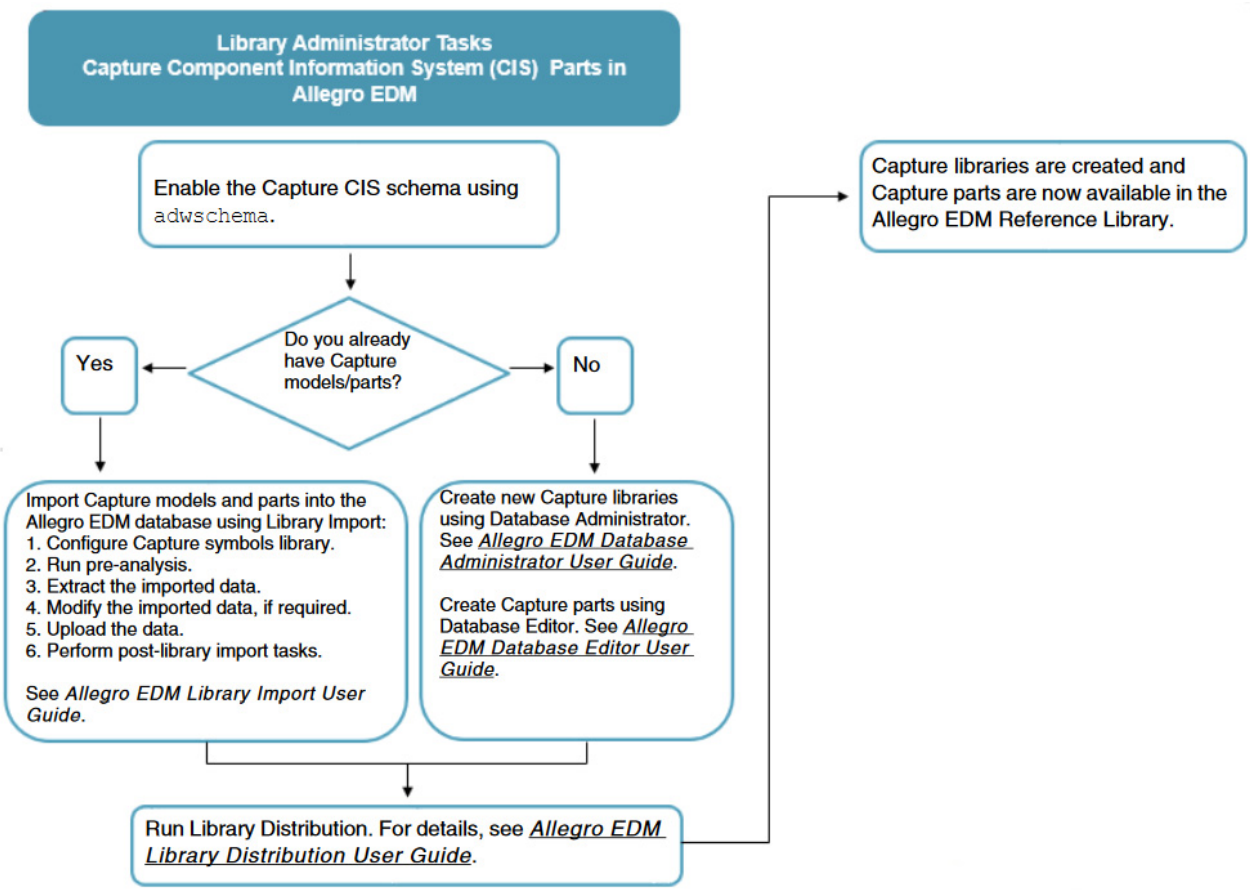
  Requests for parts similar to those in SamacSys or Ultra Librarian can also be submitted.

- Import of OrCAD Capture libraries into Allegro EDM

  Enable the Capture database schema in Allegro EDM to import or create OrCAD Capture libraries and parts in Allegro EDM.

  After you create, or import existing OrCAD Capture libraries and parts into Allegro EDM and modify them, Allegro EDM automatically creates revisions. You can distribute the updated design libraries to companies or specified design sites using Library Distribution, and use Data Exchange to synchronize data with external systems.

The flowchart illustrates the OrCAD Capture library flow in Allegro EDM:



You can also import existing OrCAD Capture libraries and parts into Allegro EDM using Library Import, or create new Capture libraries using Database Administrator.

- Manufacturer Part Numbers and Reliability

  You can enable manufacturer's part numbers (MPN) to associate MPNs to your organization’s part numbers. This is helpful for designers who know the manufacturer part number (MPN) and not the corporate part number and would like to search for a part using the MPN. It is also helpful for when librarians have applied qualification attributes to an MPN that can help designers decide whether to use a part in the design.

  This requires the management of an alternate manufacturer list (AML) and attributes for a specific electrical part in the same system as the one used for corporate parts management. As a result, all the information related to AML is available in a single database. Designers can choose from this list of components.

You can integrate alternate manufacturer lists and reliability data into the Allegro EDM component database. After you create additional object types related to manufacturer parts or reliability using the `adwschema` utility, you can import their corresponding objects into EDM using Data Exchange.

- Reliability

  You can enable the ability to specify properties that define the reliability of a part, such as:

  - electrical or thermal reliability of electrical parts to compare and analyze the behavior of designs
  - temperature reliability attribute for integrated circuits with a minimum, maximum, or average-working temperature
  - Mean Time Between Failures (MTBF) as one of the attributes of durability
  - electrical interference that specifies the environment in which an electrical part can work

Important

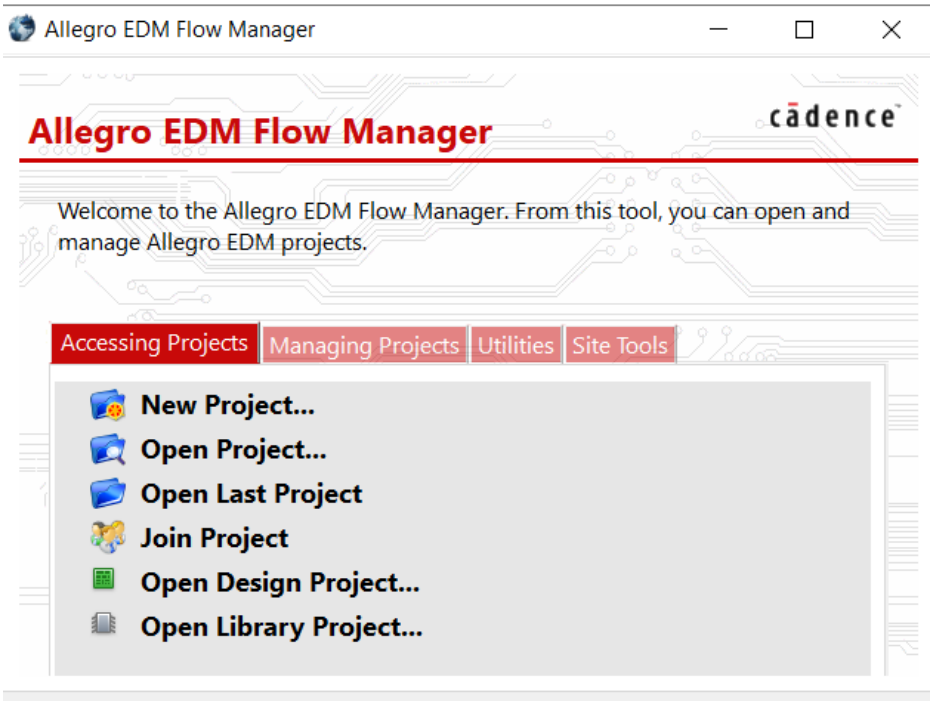
After you enable any object type, you must update the component database schema using `adw_uprev`.

Enabling Optional Allegro EDM-Managed Library Features

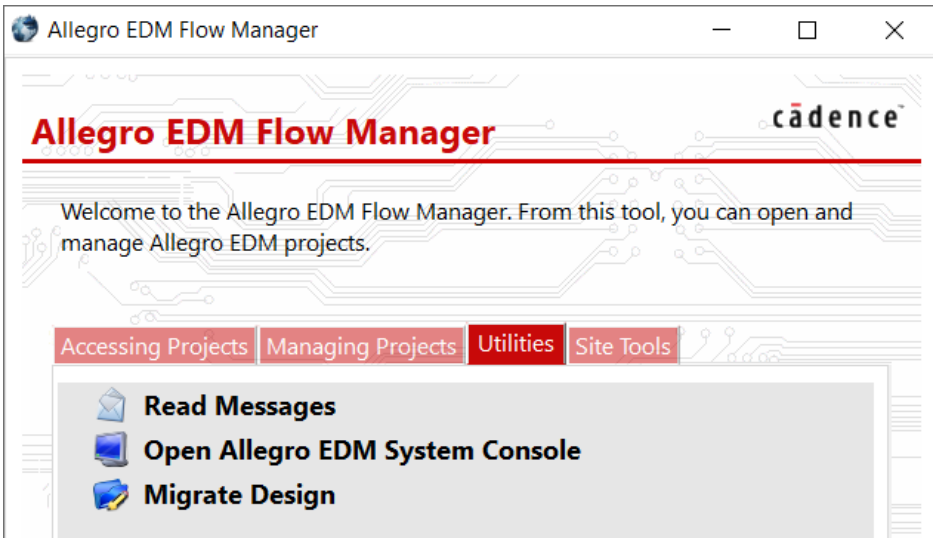
To enable optional Pulse features, which are object types in the component database, ECAD administrators use the `adwschema` utility and must do the following:

1. Navigate to the location of `<startworkbench>.bat` and run the batch file.

Allegro EDM Flow Manager is displayed.



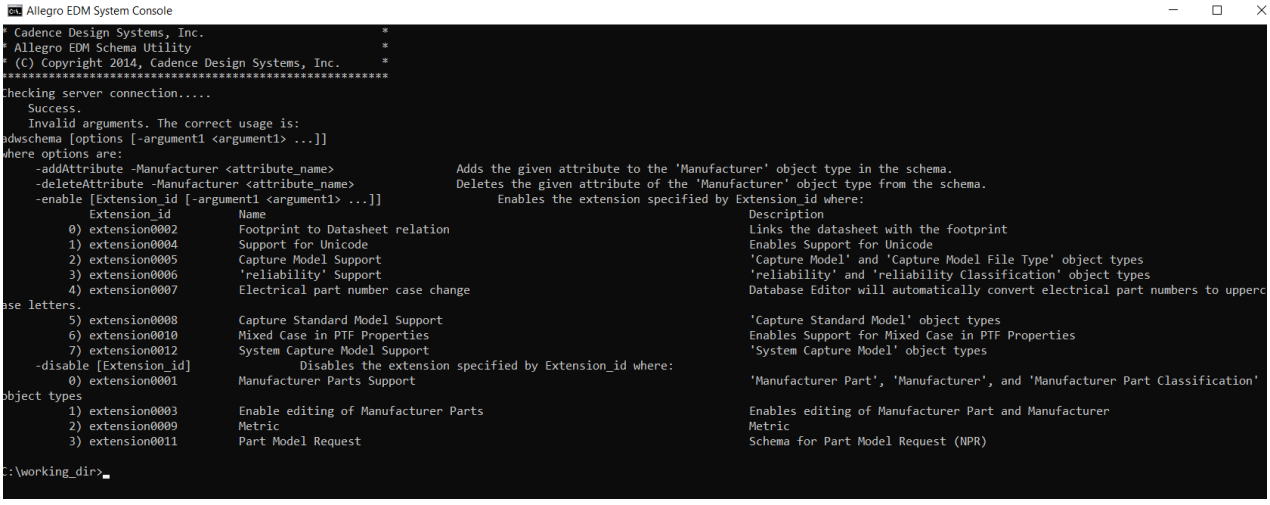
2. Click *Utilities*.



3. Click *Allegro EDM System Console*.
4. Type and run the following command: `adwschema <options>`

where `<options>` are explained in the following table:

Options	Description
`-enable [Extension_id]`	Enables the extension that is specified by `Extension_id`. You can either type `adwschema -enable [Extension_id]` on the command line or enter numbers from 0-9 to specify your choice.
`-disable [Extension_id]`	Disables a schema extension that has been enabled. The list of extensions displayed for this parameter depends on the extensions that have been enabled.
`-addAttribute -Manufacturer <attribute_name>`	Adds the *<attribute_name>* attribute to the `Manufacturer` object type. If the attribute name contains spaces, specify the name with double quotation marks
`-deleteAttribute -Manufacturer <attribute_name>`	Deletes the *<attribute_name>* attribute of the `Manufacturer` object type. You cannot delete these attributes if there are objects linked to the manufacturer object type



New Part Requests

To work with part requests, the new part request option must first be enabled in the component database. You can do the following with part requests:

- Assign the part request to someone.
- Ask the designer who submitted the request for more information.
- Put the request on hold or free it from hold.
- Subscribe to or unsubscribe from a request.

For all the tasks above, a comment is mandatory. You can also edit a submitted request or cancel it.

Important

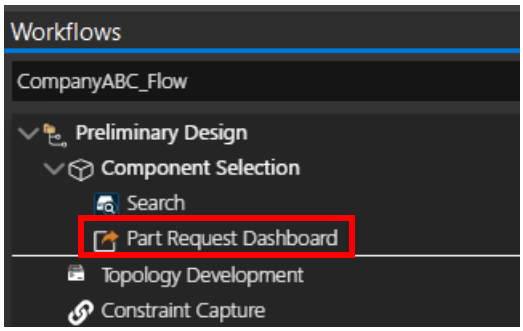
A part request can only be canceled from Allegro EDM Database Editor. It cannot be canceled from the Part Request Dashboard.

Librarians can manage part requests in two ways:

- Through a web part request dashboard by typing the following in a web browser:

  `<Pulse Service Manager URL>:7100/projects/npr`

- Using the Allegro X System Capture Part Request Dashboard



You will likely use this method if you have been assigned the role of Designer and/or Librarian in the *Pulse User Management* module. To use this method, you must log in to Allegro X System Capture with a librarian's user credentials.

You can also enable and configure the following:

- To make sure that a part linked to a part request or an ECO request is reviewed and that the librarian who checked in the part cannot release the part, an administrator or library administrator can also enable a part request with review process.
- If part requests are enabled, the default part request form can also be configured to add or remove the fields you require designers to fill in.

- To use parts from external providers such as SamacSys or Ultra Librarian in design projects, designers can submit new part requests for the same parts to be created in the PLM system and in the company's central repository of parts.

Working with New Part Requests

You can do the following tasks for part requests:

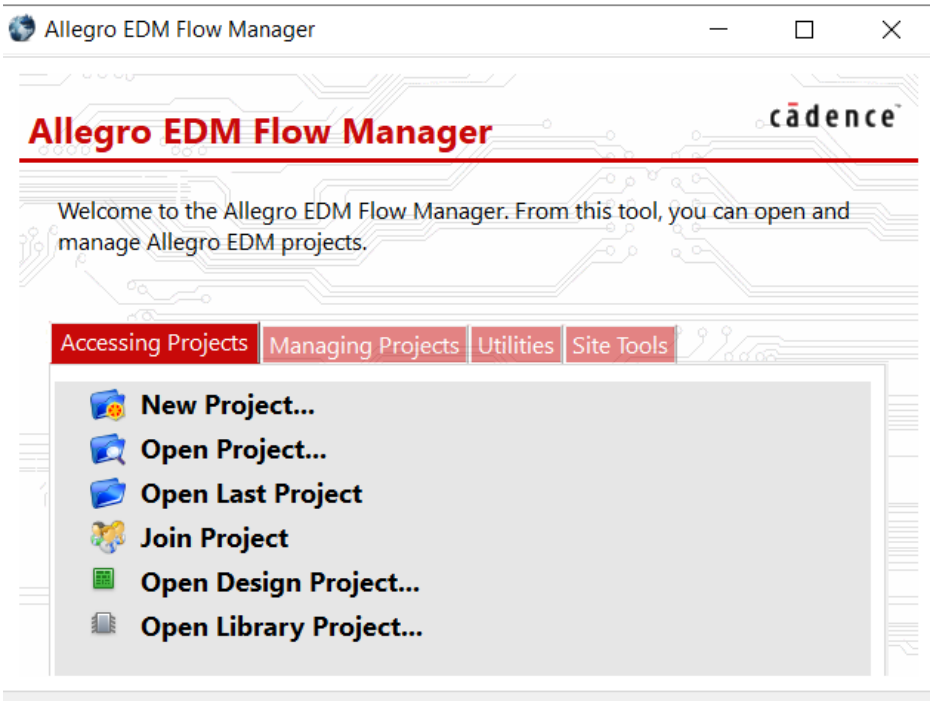
- Enabling New Part Requests
- Enabling Part Request Process With Review
- Configuring Forms
- Hiding Pulse Part Request Feature
- Enabling Auto-Pull of PLM Part Numbers in New Part Request

Enabling New Part Requests

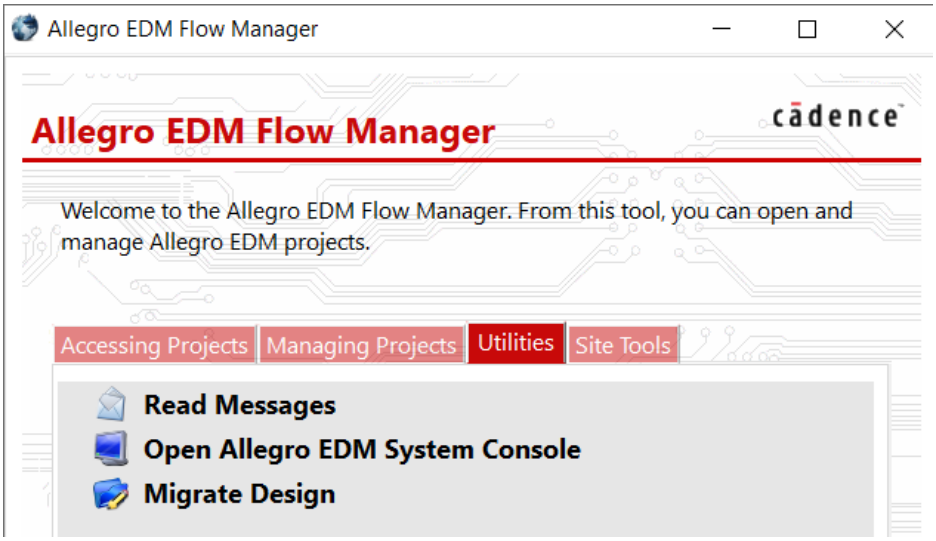
To enable part requests so that designers can submit requests for components not in the company parts database, do the following:

1. Navigate to the location of `<startworkbench>.bat` and run the batch file.

Allegro EDM Flow Manager is displayed.

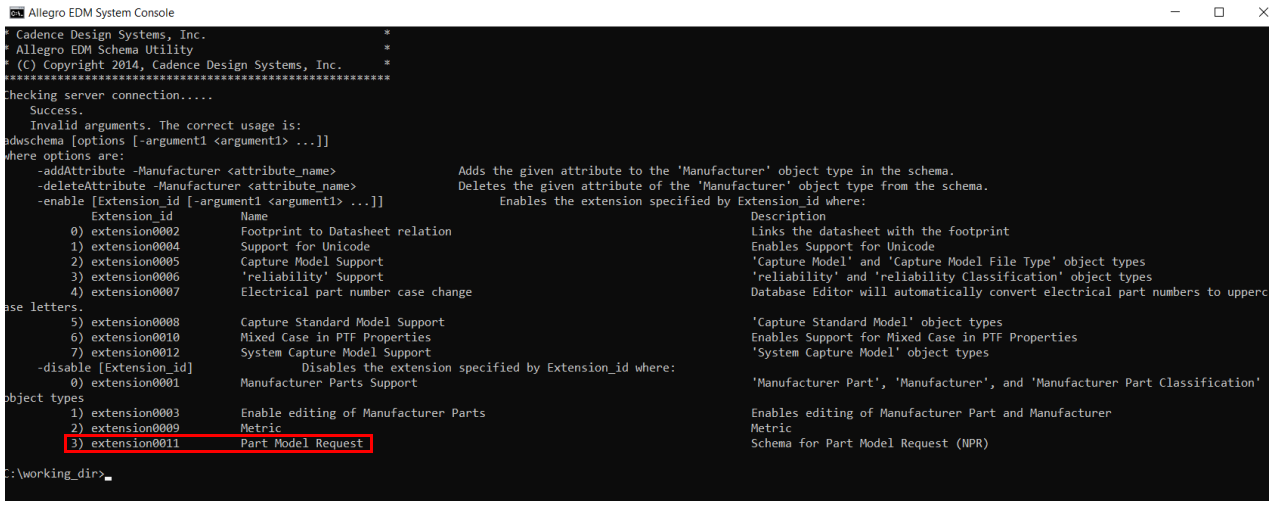


2. Click *Utilities*.



3. Click *Allegro EDM System Console*.

4. Type and run the following command: `adwschema -enable extension0011`



The part request menu options become available and are enabled in the System Capture and Pulse web dashboard user interfaces.

Enabling Part Request Process With Review

If you enable part requests, you can do the following to make sure that:

- a part linked to a part or an ECO request is reviewed.
- the librarian who checked in the part cannot release the part.

To enable the part request process with review, do the following:

1. Add the following lines from the out-of-the-box `search.config` file to the downloaded configuration file:

```
"workflow": [
            {
                "name": "part_request",
                "value": "npr"
            },
            {
                "name": "part_request_eco",
```

```
                "value": "npr"

            }
```

2. To enable the part request process with review, ensure that the value of `"part_request"` and/or `"part_request_eco"`, is set to `npr_review`.

   A properly configured `search.config` file would look as follows and this is the configuration file you upload to the Pulse web dashboard:

```
{
    "unicorn": {
        "version": 0,
        "important_comment": "Do not remove version field from the file while changing the file.",
        "workflow": [
            {
                "name": "part_request",
                "value": "npr_review"
            },
            {
                "name": "part_request_eco",
                "value": "npr_review"
            }
        ]
    }
}
```

3. Save and upload the modified configuration file.

   The configuration is updated and the part request process with review is enabled.

Related Topic

Using Global Configuration File

Configuring Forms

Pulse provides default forms for:

- project creation
- part request - This form is only available If you enable the part request process.

The forms have fields for key information that Cadence thinks you might want designers to specify, such as the project name, project status, schematic number, the manufacturer and part number, a description for the part being requested, any relevant datasheet.

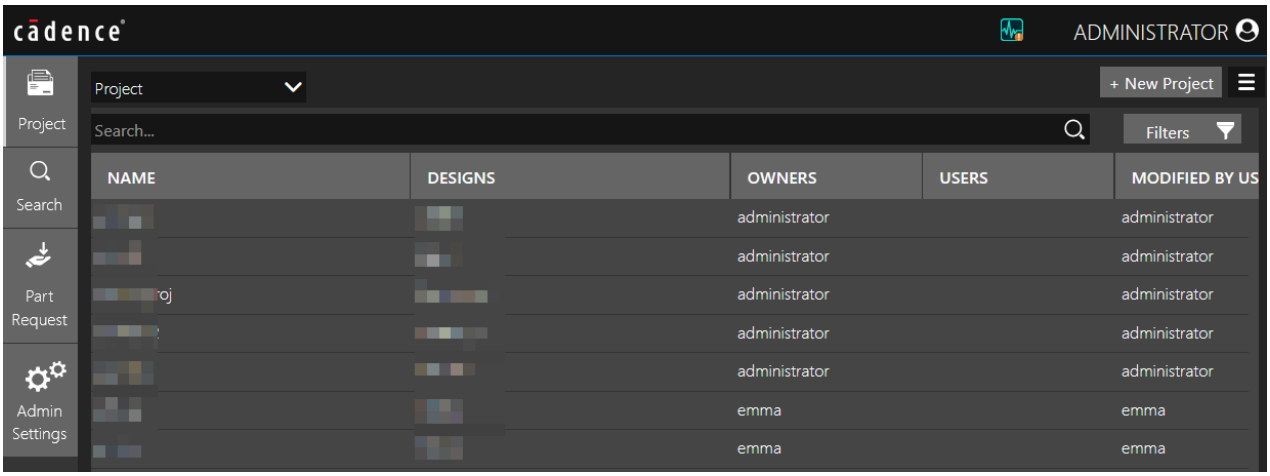
If your organization wants to configure these forms, users with the `Administrator` role can modify them to define the fields they require from designers. The number of fields you can add is only limited by the real estate of the user interface.

The project creation form has a dynamic database schema, so any changes you make are reflected instantly in the *New From Template* dialog box of the authoring application such as Allegro X System Capture. The fields you define in the project creation form, such as `$cds_JobNumber`, are resolved in the Allegro X System Capture user interface.

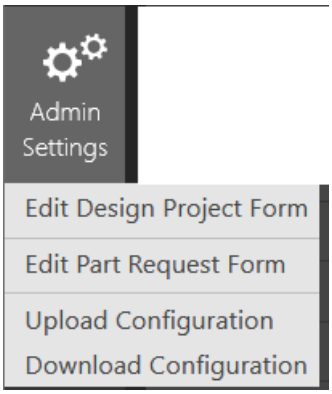
To configure the project creation or part request forms, do the following:

1. Access the Pulse web dashboard by typing `http://<Pulse access URL>:7100/projects` in a web browser.

   The Pulse dashboard is displayed.



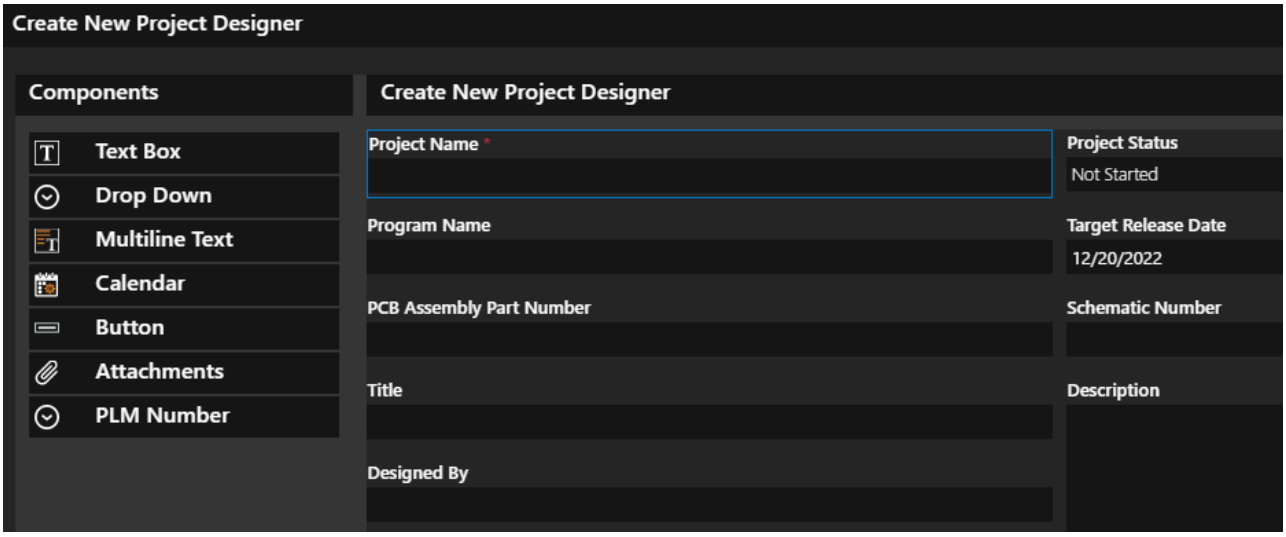
2. Click *Admin Settings*.



3. Select one of the following depending on what you want to configure:

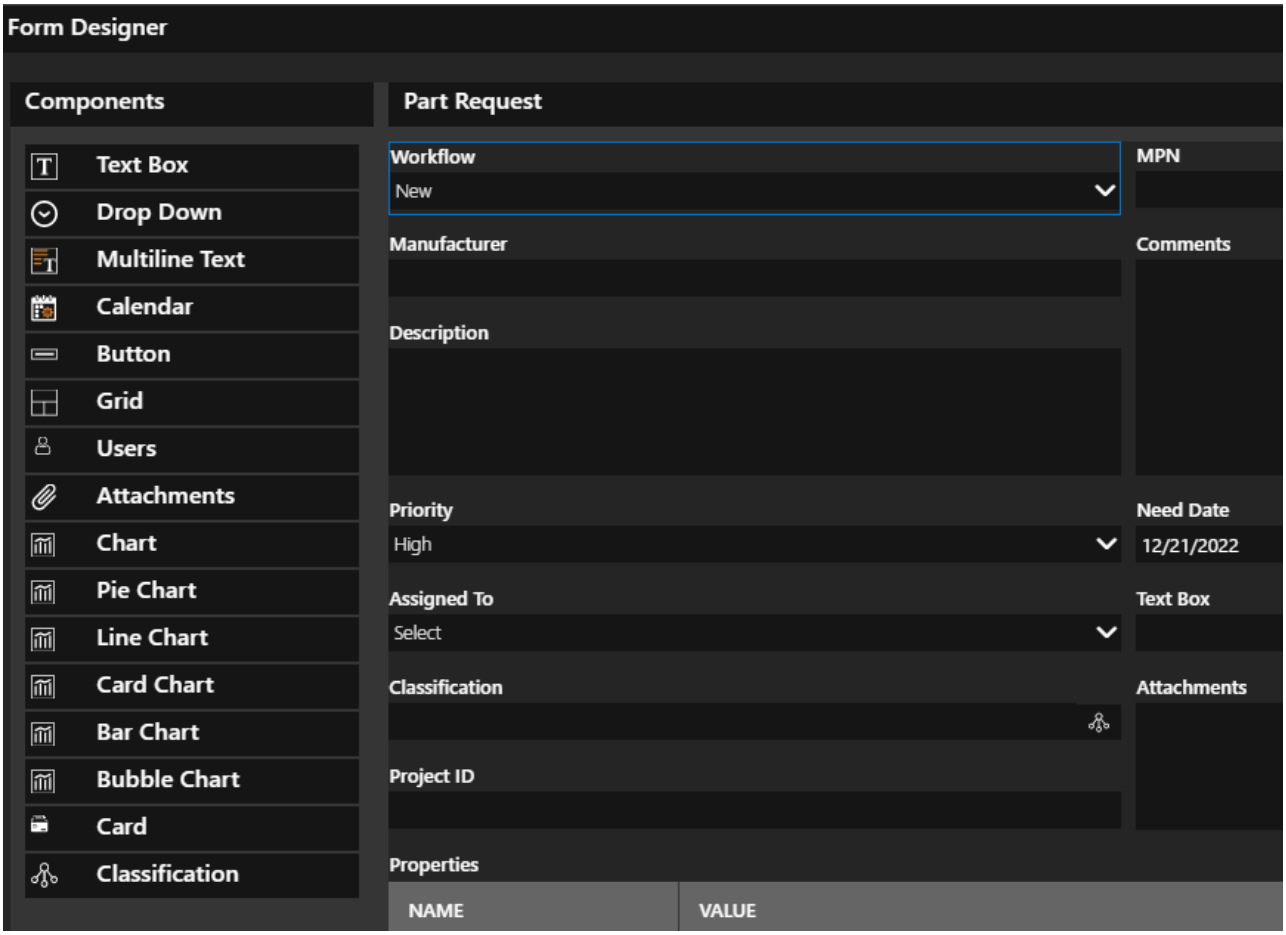
- *Edit Design Project Form*
- *Edit Part Request Form*

The default configurable form is displayed.



Important

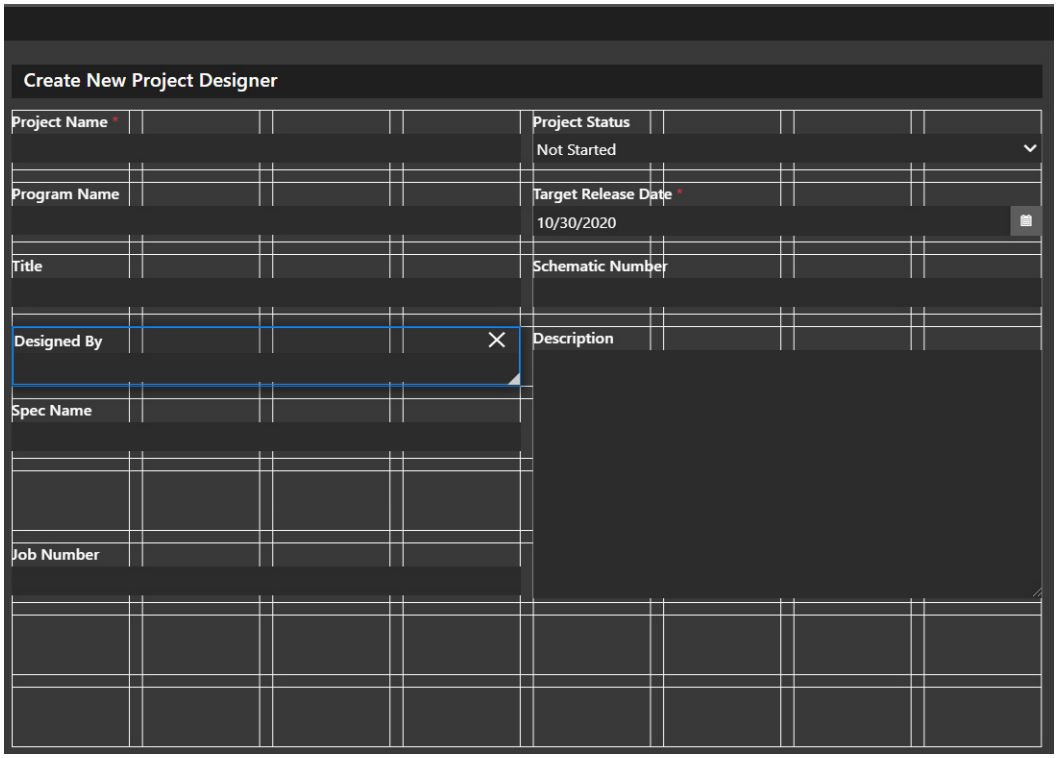
Do not modify or delete the Project Name in the Create New Project Designer form. Do not modify its label either.



4. To add a component to the form, drag and drop the required user interface controls from the *Components* list in the left pane to the *Create New Project Designer* or *Part Request* pane in the middle of the dialog box.

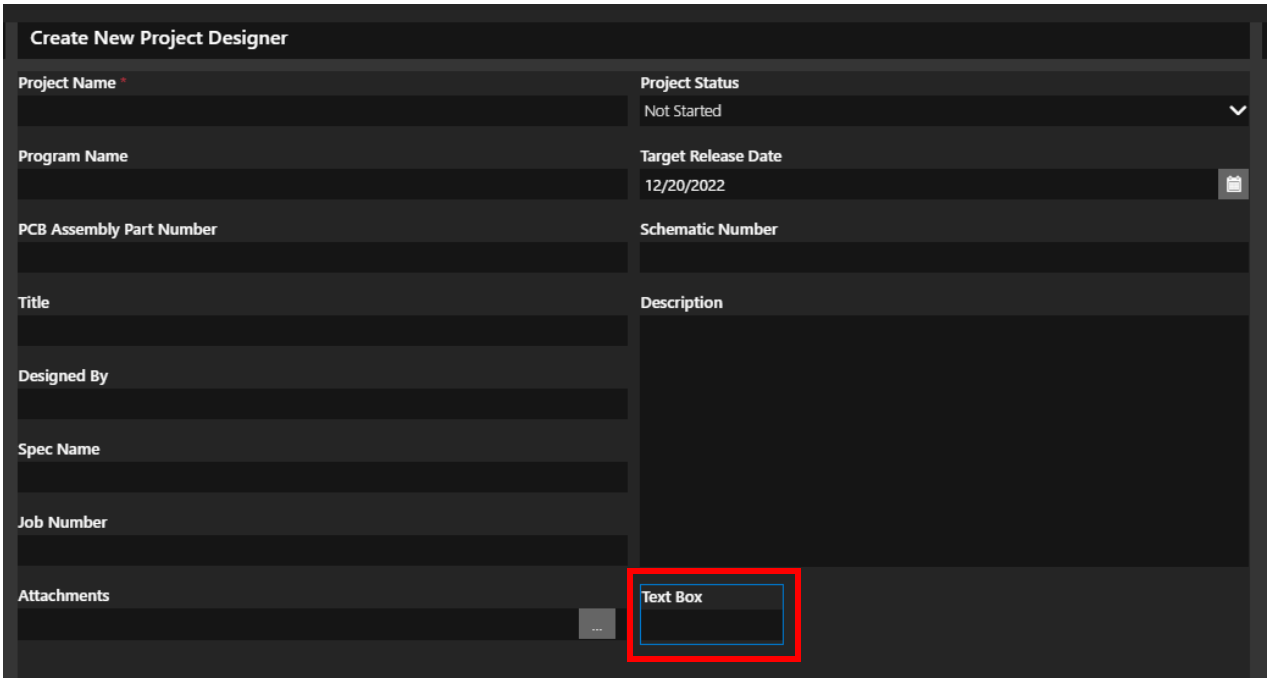
A useful widget for the project creation form is *Attachment*. If this field is in the form, designers can attach third-party documents and files to a project with some conditions.

To help you place a UI control more precisely, and to define the limits of a UI field, grid lines are displayed when you click and hold down a UI control in the middle pane. You can also resize a UI control in the middle pane to fit the data type text in the field.

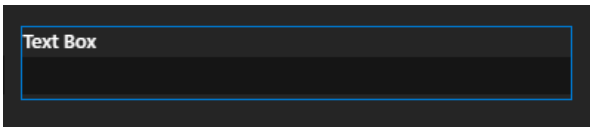


5. Hover the mouse cursor over the blue line to view the handles of the new widget.

In the following image, the new widget is *Text Box*.



6. Select one of the handles and drag until the box is the size you want.



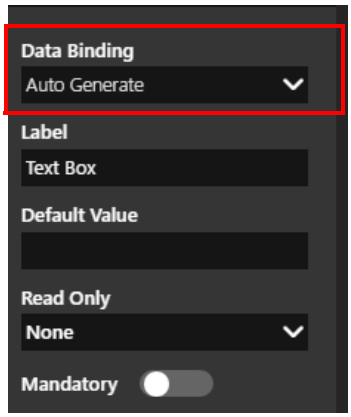
7. Generate a data binding by selecting the required user interface control in the middle pane. For example, select the *Text Box* control.

   The data binding establishes a connection between the application user interface and the data it displays.

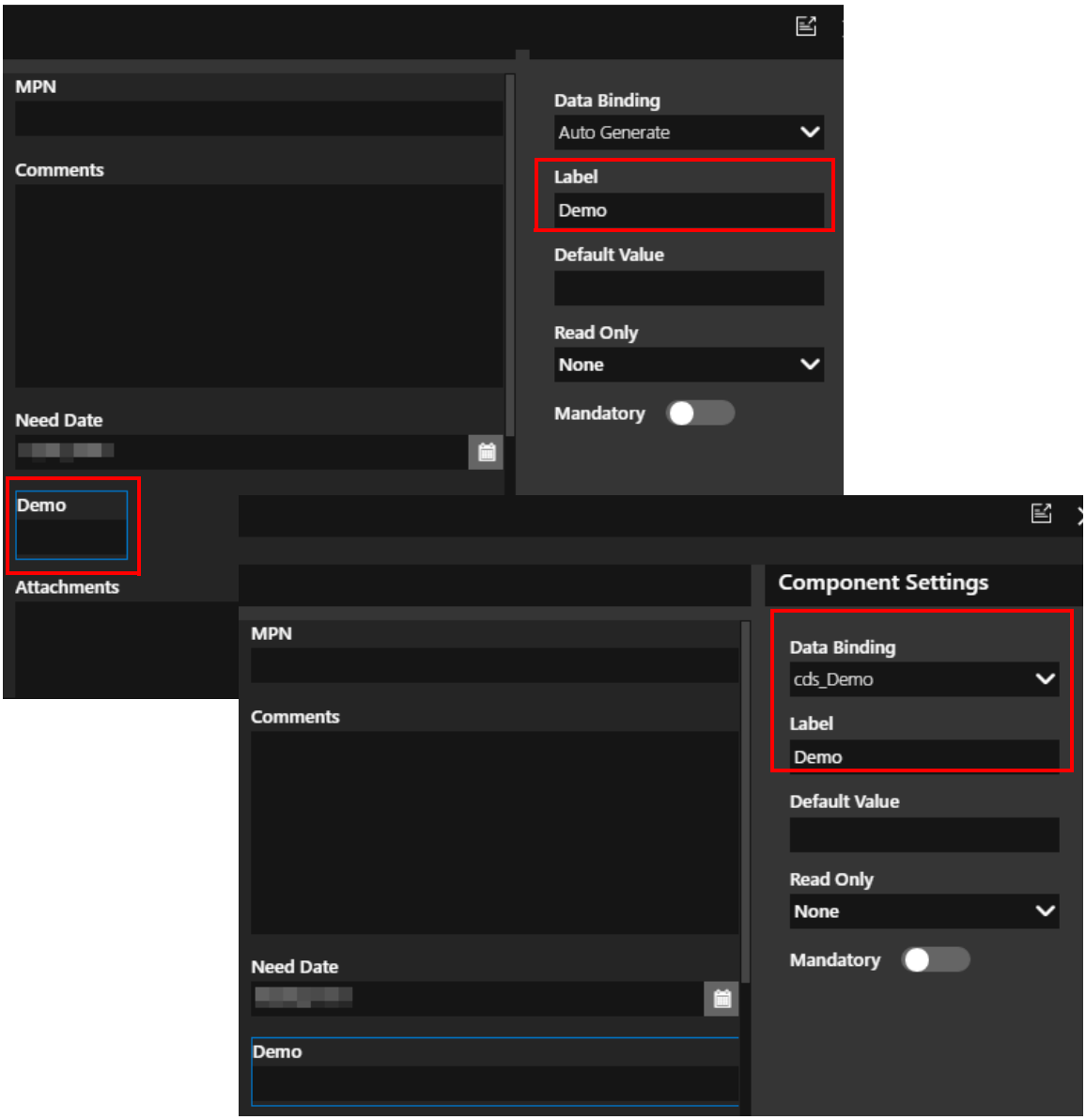
8. If you want a new data binding to be created, set `Auto Generate` in the *Data Binding* field.

   If there is a legacy data binding, such as for a a custom variable defined in the System Capture page border, manually type it into the data binding field.

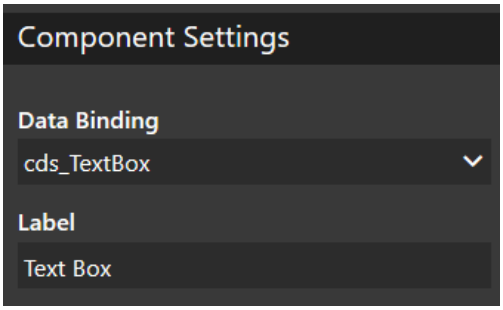
If you select `Auto Generate`, Pulse automatically assigns a data binding based on the label when you publish the form.



For example, if you label the text box as *Demo*, the data binding is *cds_Demo*.



If you did not specify any label for the UI control, the data binding is *cds_TextBox*.



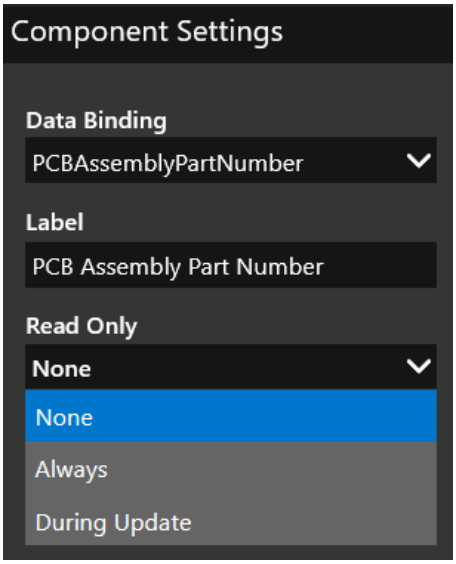
You can also select an existing data type from the *Data Binding* drop-down, if the selected data type is not already assigned to a field in the form.

Important

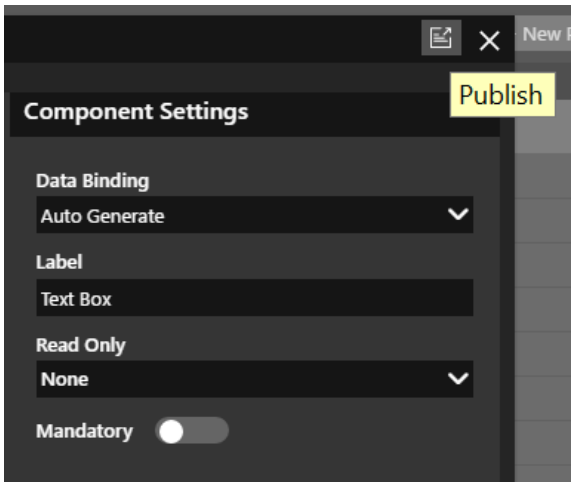
For new part requests, the data type to which you can bind a user interface control is predefined in the New Part Request database schema. This schema cannot be modified by users of any role, including administrators.

9. Toggle *Mandatory to d*efine which fields should be mandatory for designers during project creation.

10. In *Read Only*, choose between `None`, `Always`, or `During Update` if you want to restrict editing post project creation.



11. To make the configured form available to designers, click the *Publish* button on the top right of the form.



Hiding Pulse Part Request Feature

If your company has its own request workflows for parts, you might want to hide the Pulse Part Request feature in the *Project* tab of the Pulse web dashboard. Hiding the Pulse Part Request feature in your setup hides the following part request features:

- New Part Request tab in the Pulse web dashboard
- New Part Request option as a shortcut menu option in Unified Search
- Option by administrators to edit the New Part Request form

To hide the Pulse Part Request feature, do the following:

1. Add the following lines from the out-of-the-box `search.config` file to the downloaded configuration file:

```
"part_model_request": [
            {
                "name": "prefix",
                "value": "Request_"
            },
            {
"name": "enabled",
```

```
"value": true

}
```

The default value for part requests is `true`.

2. Ensure that "`Value`" is set to `false` in the downloaded `search.config` file.

   A properly configured `search.config` file would look as follows and this is the configuration file you upload to the Pulse web dashboard:

```
{
    "unicorn": {
        "version": 0,
        "charsToEscape": "+-&|!(){}[]^~*?\\/=<>",
        "cloudApi": {
            "partsURL": "https://pcb.cadence.com/unifiedsearch",
            "partsURLForAutomation": "https://stagepcb.cadence.com/unifiedsearch"
        },
        "others": [
            {
                "part_model_request": [
                    {
                        "name": "enabled",
                        "value": false
                    }
                ]
            }
        ]
    }
}
```

3. Save and upload the modified configuration file.

   The configuration is updated and the Pulse Part Request features are hidden.

Related Topic

Using Global Configuration File

Enabling Auto-Pull of PLM Part Numbers in New Part Request

To use parts from external providers such as SamacSys or Ultra Librarian in design projects, designers can submit new part requests for the same parts to be created in the PLM system and in the company's central repository of parts.

This ensures that multiple systems such as PLM systems and the ECAD library database are in sync with the new parts being requested by designers.

For System Capture designs to use this feature, you need to configure and set up a PLM connector using the Library Synchronization service.

To make it easier for you to identify newly created parts in the PLM system, it is recommended that you make the *Description* field in the New Part Request form mandatory for designers when they submit a part request for a PLM-generated part number.

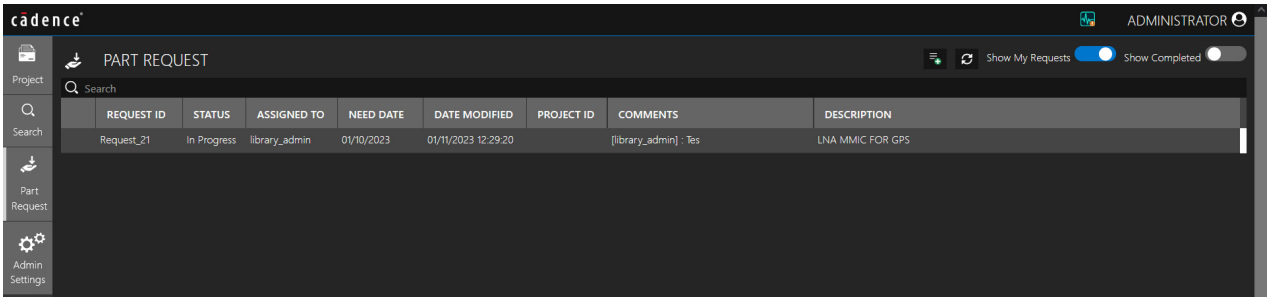
After a part request for a PLM-generated part number is submitted by a designer, do the following to add the part to the PLM system and the Pulse database:

1. Access the Part Request dashboard through Allegro X System Capture or a web browser.

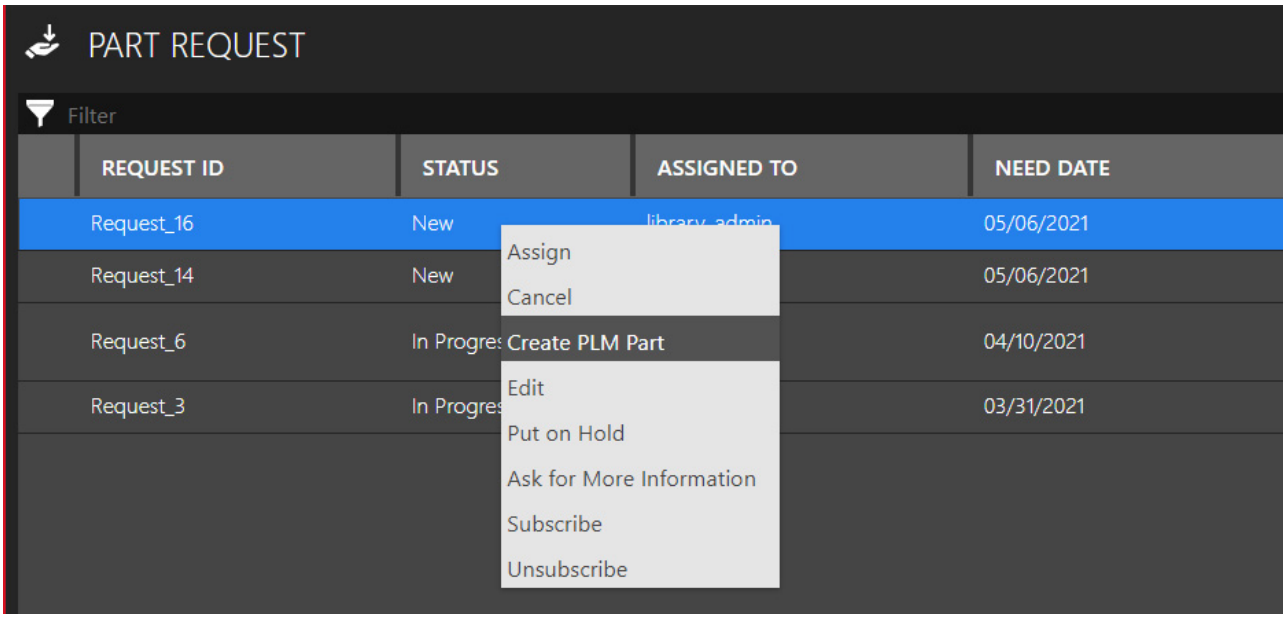
   As a librarian, you are unlikely to be working with System Capture, so it might be easier for you to access the dashboard through a web browser.

   To access the part request dashboard through a web browser, type the following in the web page URL field: `http://<Pulse access URL>/projects/npr`

   The Part Request dashboard is displayed.

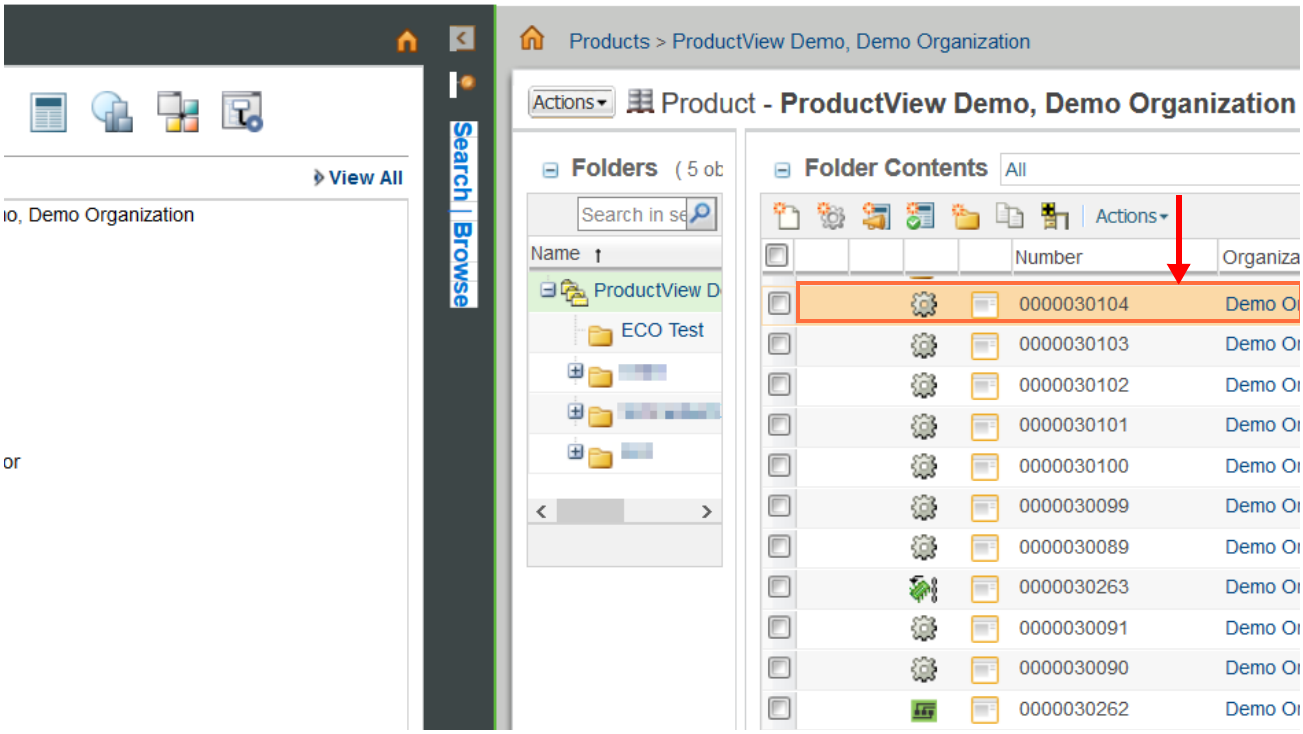


2. Right-click the part request whose part you want to add to the PLM system and select *Create PLM Part*.



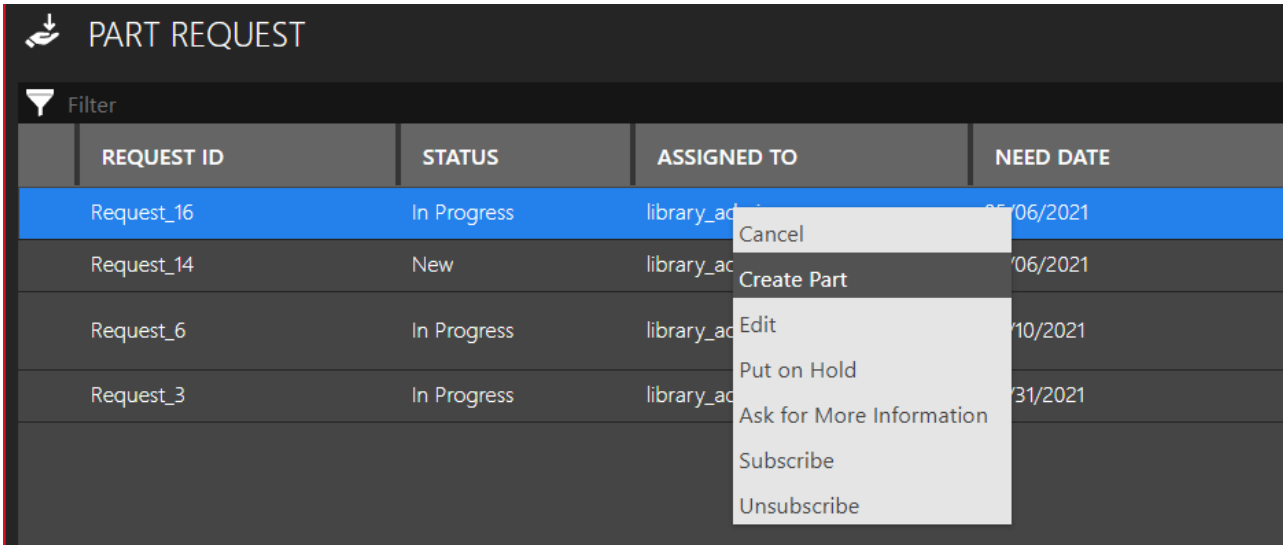
A message that the part has been added to the PLM system is displayed with the part number.

You can then check that the part has been created in the PLM system.

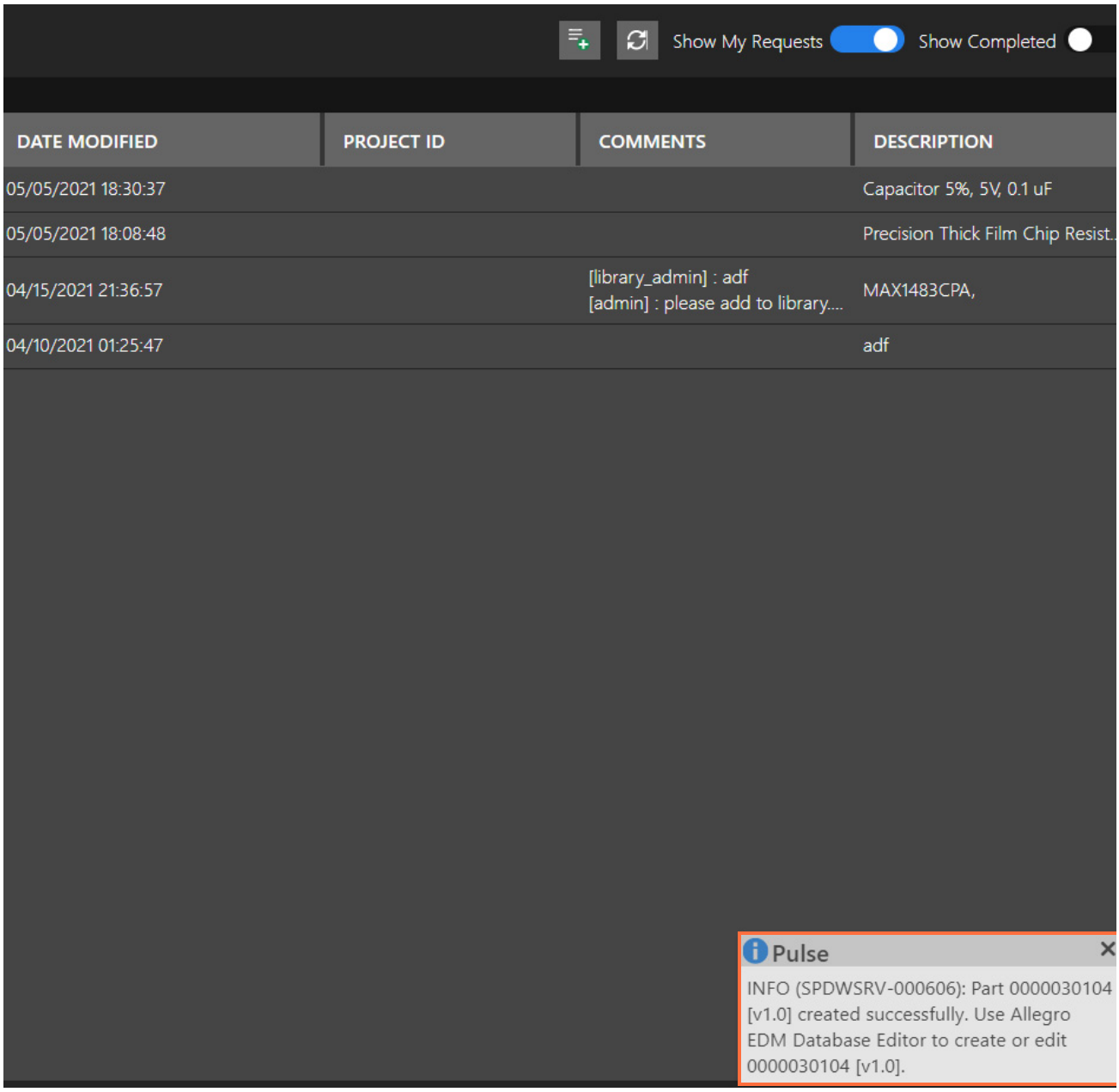


After the part is created in the PLM system, you can add the part to the Pulse component database.

3. In the Part Request dashboard, right-click the part request whose part you need to add to the Pulse component database and select *Create Part*.

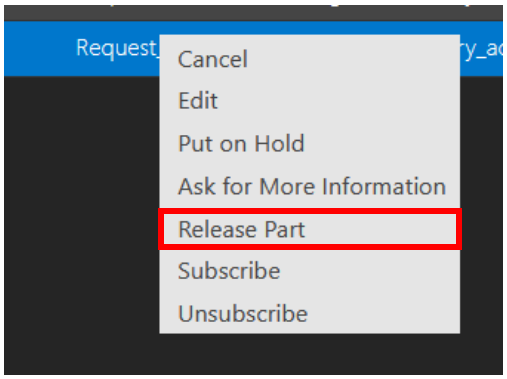


The part is added to the component database and a message is displayed with the part number.

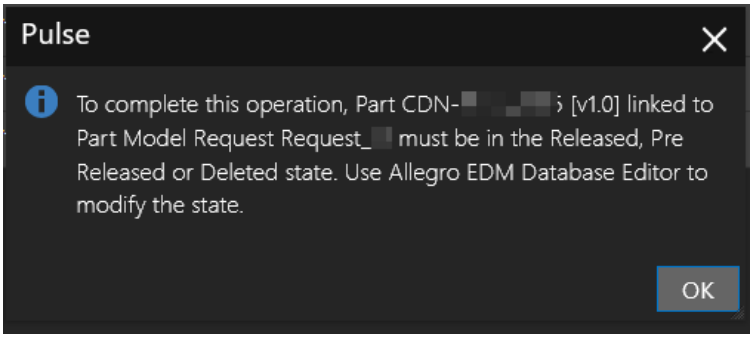


After the part is created in the Allegro EDM component database, use Allegro EDM Database Editor to edit the part, and release it.

When you right-click the part request in the Pulse web dashboard after the part has been created in the component dashboard, a new option is available - *Release Part*.

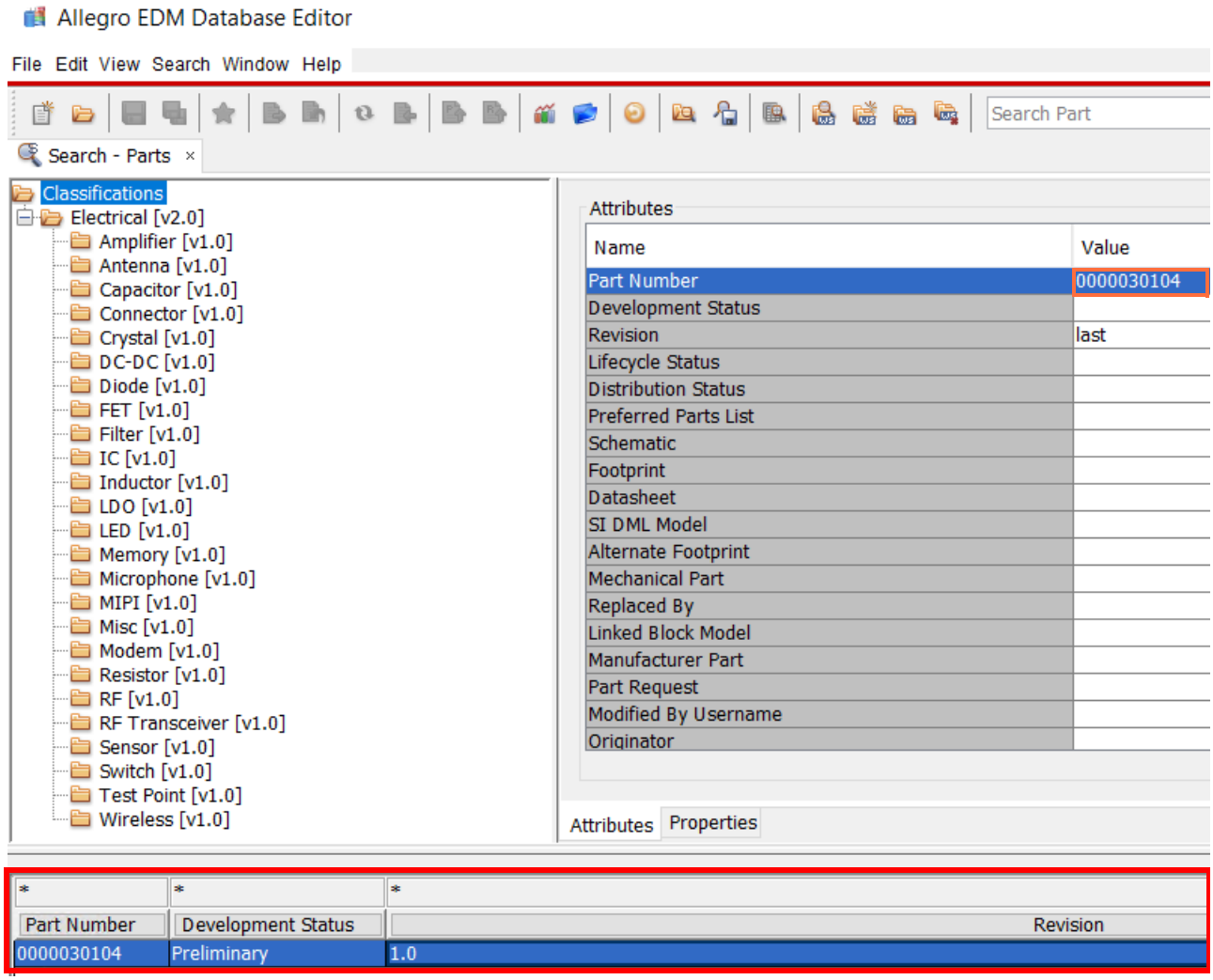


Clicking this option does not release the part. It prompts you about the steps to be followed to address and complete the part request as completed.

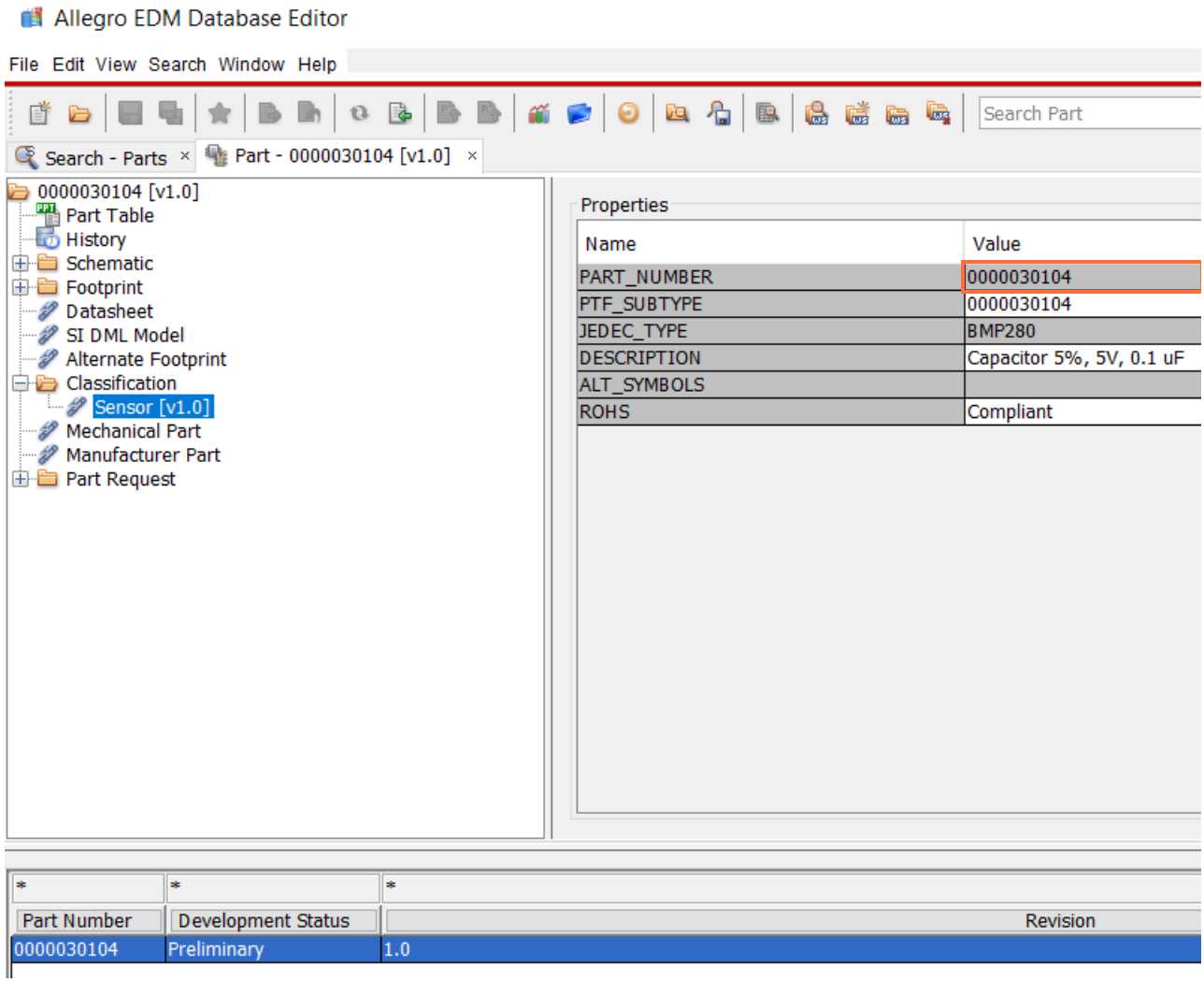


Note: The part and part request numbers have been blurred in this image.

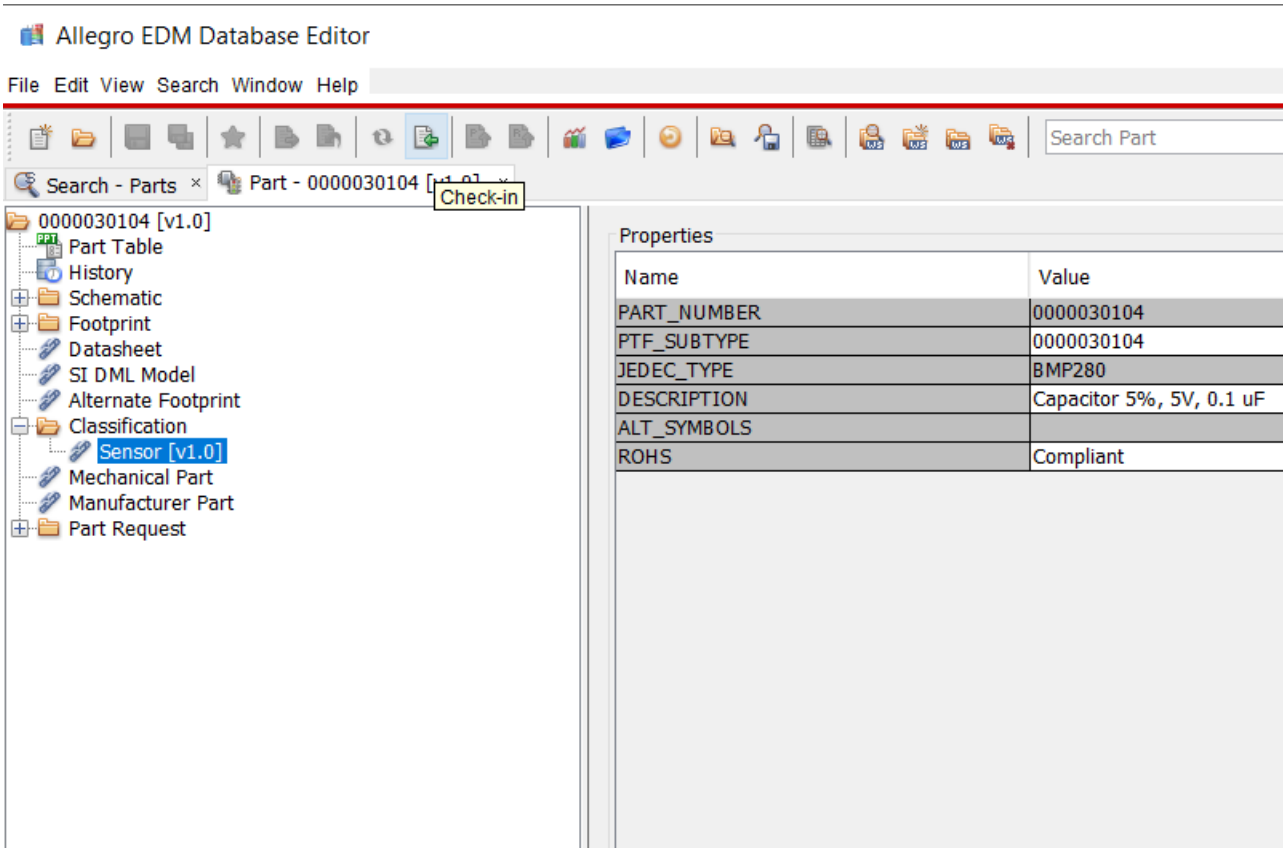
You can now search for this part in Allegro EDM Database Editor so as to edit it and release it.



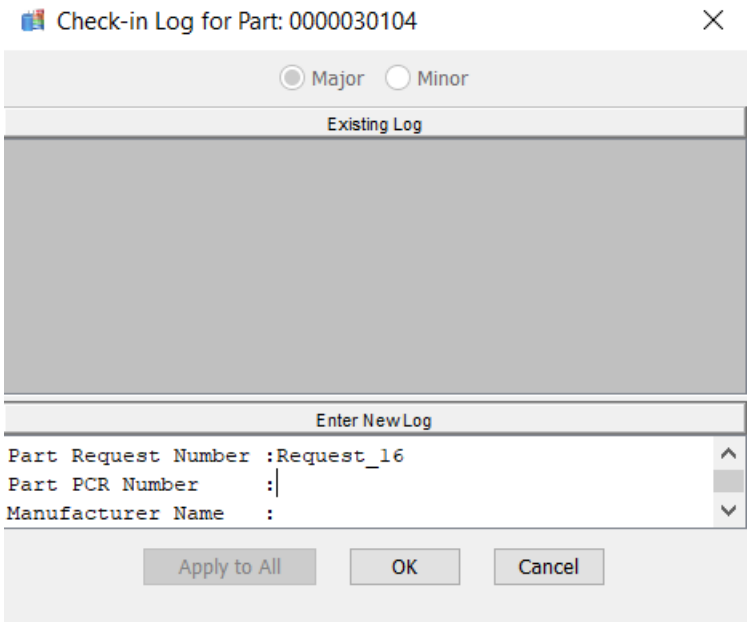
Modify the part as required.



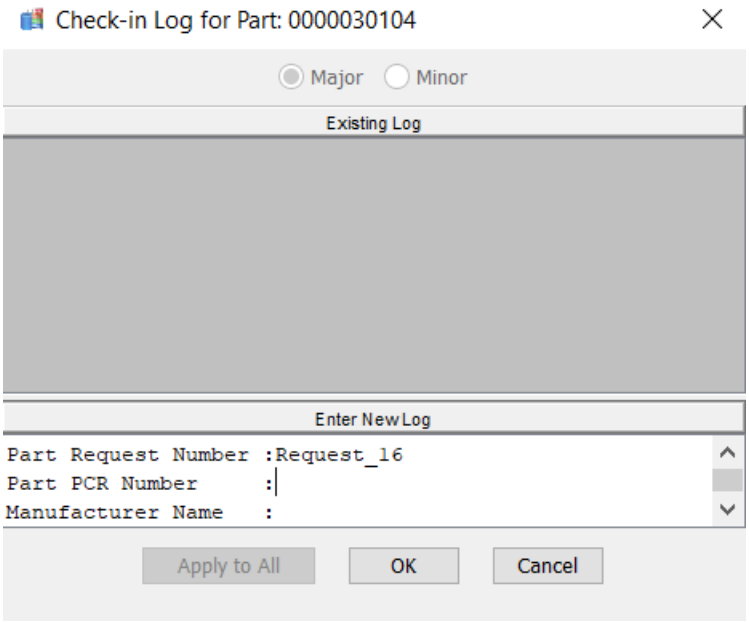
After modifying it, check it in, if needed, and release it.



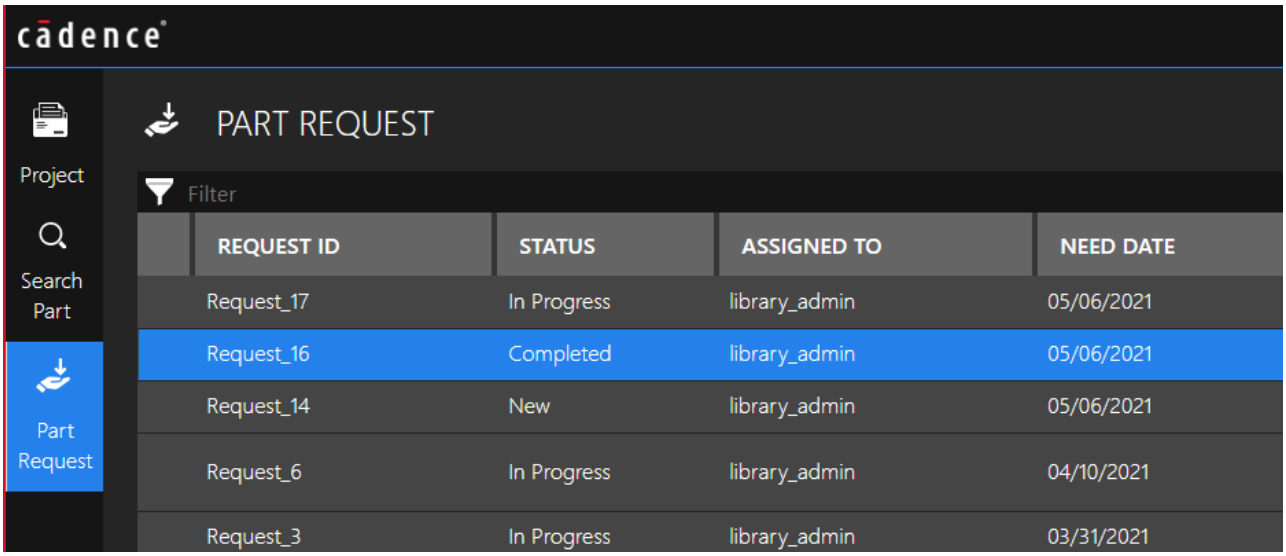
When checking in a part that was created for a part request, specify the request number.



After checking in the part, you can release it.



After you release the part, the status of the part request changes to *Completed* in the Part Request dashboard.



Related Topics

Allegro X System Capture User Guide for details on Allegro X System Capture Part Request Dashboard

Allegro EDM Flow Manager User Guide

Library Synchronization Between PLM System and Allegro X Pulse

Enabling PLM-Connected Project Creation

If Allegro System Capture designers need to type in board or assembly numbers, or they have a PLM connector deployed and are using Publish for Manufacturing, they must work with PLM-connected projects.

To enable PLM-connected project creation by designers, do the following:

1. Add the following lines from the out-of-the-box `search.config` file to the downloaded configuration file:

```
{
                "name": "plm_create_project",
```

```
"display": "PLM Project Form",
"value": "create_project",
"enabled": false,
"forms": [
  {
    "name": "plm_designproperties"
  }
```

2. Set the value of `"enabled"` to `true`.

   A properly configured `search.config` would look as follows and this is the configuration file you upload:

```
{
    "unicorn": {
        "version": "1.0",
        "charsToEscape": "+-&|!(){}[]^~*?\\/=<>",
        "cloudApi": {
            "partsURL": "https://pcb.cadence.com/unifiedsearc
            "partsURLForAutomation": "https://stagepcb.cadenc
        },
        "workflow": [
            {
                "name": "plm_create_project",
                "display": "PLM Project Form",
                "value": "create_project",
                "enabled": true,
                "forms": [
                    {
                        "name": "plm_designproperties"
                    }
```

3. Save and upload the modified configuration file.

   The configuration is updated for all designers connected to this Pulse server.

Related Topic

Using Global Configuration File

Enabling Diagnostic Test Case Download by Non-Administrator Users

Users can generate a diagnostic test case for Cadence to review for debugging purposes. The test case contains the client and server logs along with some additional diagnostic files from Pulse. After generation, it is automatically created as a zip file in the Pulse primary node at `<Pulse Home>/server/data/medic/testcases`.

If you want to enable the download of the test case on client machines, do the following:

1. Add the following lines from the out-of-the-box `search.config` file to the downloaded configuration file:

```
"others": [

{

"allowDiagnosticsDownload": true

}

]
```

2. To enable diagnostic test case generation by non-administrator users, modify the value of `"allowDiagnosticsDownload"` to `true`.

   A properly configured `search.config` file would look as follows and this is the configuration file you upload to the Pulse web dashboard:

```
{
    "unicorn": {
        "version": 0,
        "charsToEscape": "+-&|!(){}[]^~*?\\/=<>",
        "cloudApi": {
            "partsURL": "https://pcb.cadence.com/unifiedsearch",
            "partsURLForAutomation": "https://stagepcb.cadence.com/unifiedsearch"
        },
        "others": [
            {
                "allowDiagnosticsDownload": true
            }
        ]
    }
}
```

3. Save and upload the modified configuration file.

   The configuration is updated. When non-administrator users on client machines generate a test case using the Tcl command, the zip file is downloaded to the client hard disk.

The zip file is downloaded using the web browser download preference, which is typically user prompt for the download location or an auto-download to the Downloads folder.

Related Topic

Using Global Configuration File

Configuration of Publish for Manufacturing

Publish for Manufacturing enables you to publish data to your Product Lifecycle Management system so that the product can be manufactured.

You can also use Publish for Manufacturing to create a manufacturing data set that includes the ECAD Bill of Materials (BOM) as well as source and derived files to through a structured template. Configuration allows you to define the required outputs and the package structure per your release process.

Related Documentation

Publish for Manufacturing User Guide

5

Advanced Configuration and Pulse Setup for Use by Various Applications

If you would like to configure data synchronization of library metadata with external data sources, such as a PLM system, as a scheduled task, read this section.

Identification of Regular Synchronization Tasks and Frequency

Component engineers in companies generally maintain data such as pricing and component availability information in product lifecycle management (PLM) systems. You can enrich the component data in Allegro EDM through a synchronization process, which is done using the Allegro EDM Data Exchange application. To simplify synchronization of PLM-owned data with the Allegro X Pulse-managed library, it is a good idea to identify a schedule for such tasks.

Note: If the names of attributes in the PLM system and the Pulse-managed library are different, consider using the attribute mapping feature of Allegro EDM Data Exchange before you synchronize data.

If you work with the Windchill PTC PLM system, in addition to Data Exchange, you can also use the Library Synchronization service to configure and set up a data synchronization schedule.

Synchronization of Managed Library Database Across Multiple Sites

Library administrators manage a repository of enterprise component libraries at a primary site. Designers from various client sites connect to this primary site to access component data.

When you create new library components or modify existing ones and publish them, the components are fetched and utilized by the client sites. This process of making component libraries available to the designers is called library distribution. This allows component

changes, initiated by the primary site, to be available and synchronized with all other client sites in the enterprise.

Configuration of Publish for Manufacturing-Specific Features in Pulse

Before designers start working with Publish for Manufacturing (PFM), complete the following PFM-related configuration tasks.

- Configuration of Library Synchronization Service with PLM System

  To synchronize Windchill-sourced data (PLM) with the Allegro X Pulse-managed library, you use the Library Synchronization service that leverages the Allegro EDM Data Exchange function to update the Pulse-managed library.

- Configuration of Publish for Manufacturing templates

  Publish for Manufacturing is shipped with two default templates - for Allegro X System Capture and Design Entry HDL. These templates contain the template name, description, the connector for which the template is being used, such as Windchill, the PLM version for which it is valid, and so on.

  You can create custom templates and name them as required to simplify your work. For example, if you have two Windchill servers for different purposes, you can name them `PRODUCTION` and `TEST`.

Related Topic

Allegro EDM Library Distribution User Guide

Library Synchronization Between PLM System and Allegro X Pulse

Publish for Manufacturing User Guide

6

Managing Test Server

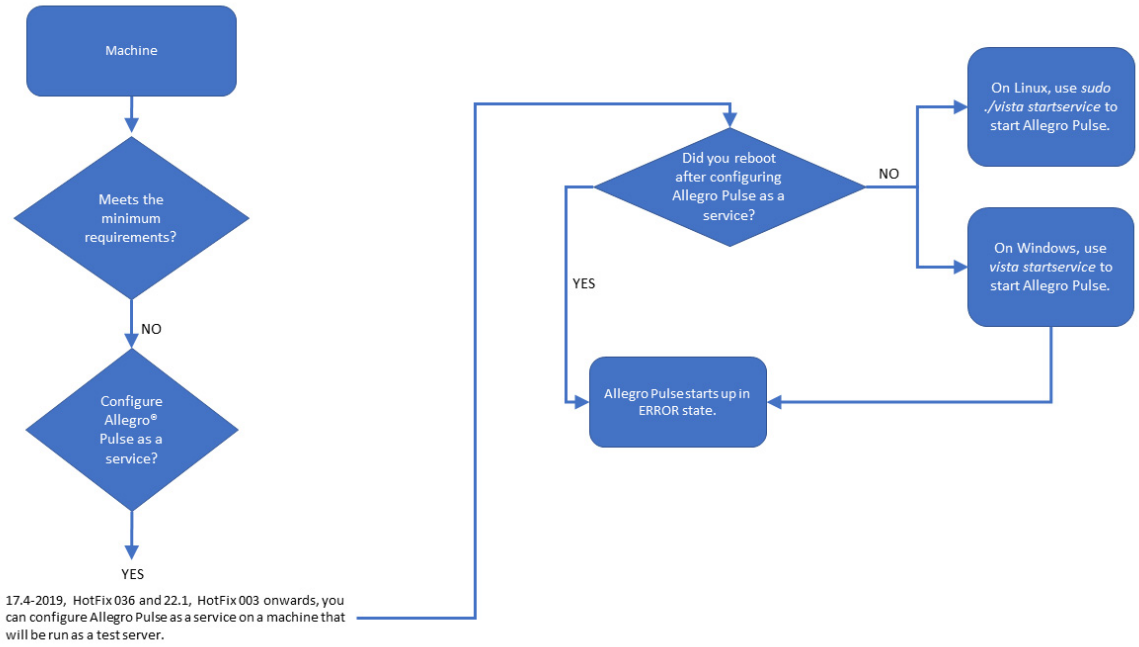
A test, or a staging server, is used for:

- Validation of operating system patches
- Non-Cadence software deployment
- Checkup of backup data on the production server
- Evaluation of new features in HotFixes
- Testing before deploying to production

It is recommended that you deploy a new release on a test server or staging environment instead of the production environment. The test environment can also be used to evaluate the software updates that are periodically released.

Allegro X Pulse in the test server mode requires fewer resources than the Allegro X Pulse server (Vista). Refer to *Allegro X Platform System Requirements* for the resource requirements.

However, if the machine where you start the Pulse server does not meet the minimum resource and OS requirements for the Allegro X Pulse (Vista) server, it starts in the ERROR state.

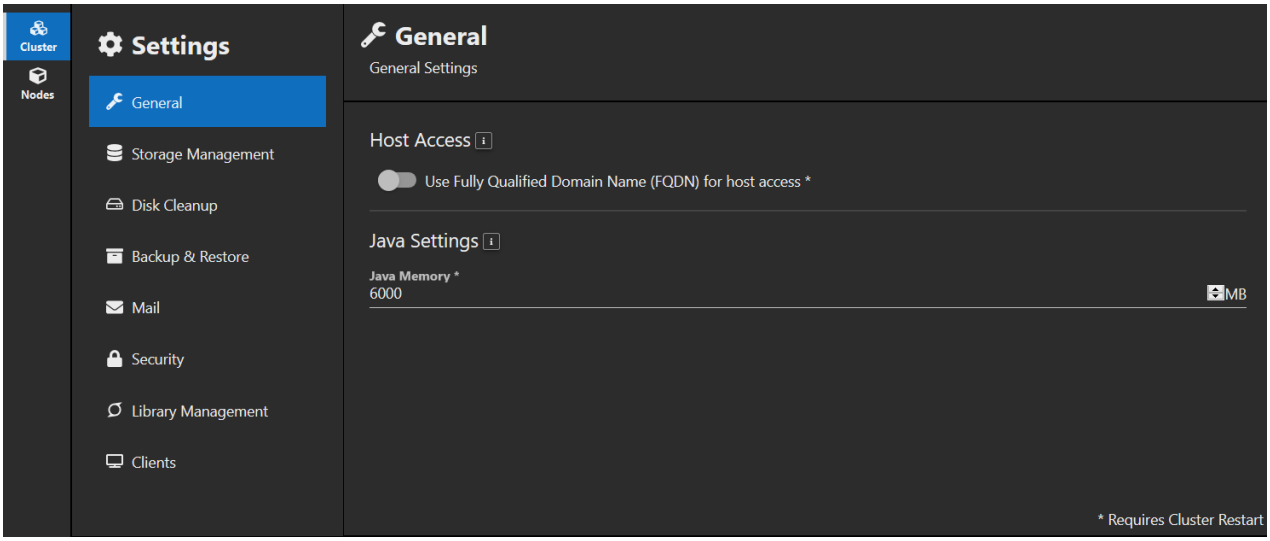


Note: It is recommended that you configure Pulse as a service. If you did not configure it as a service, see Starting Pulse Service Manager for instructions on how to start Pulse.

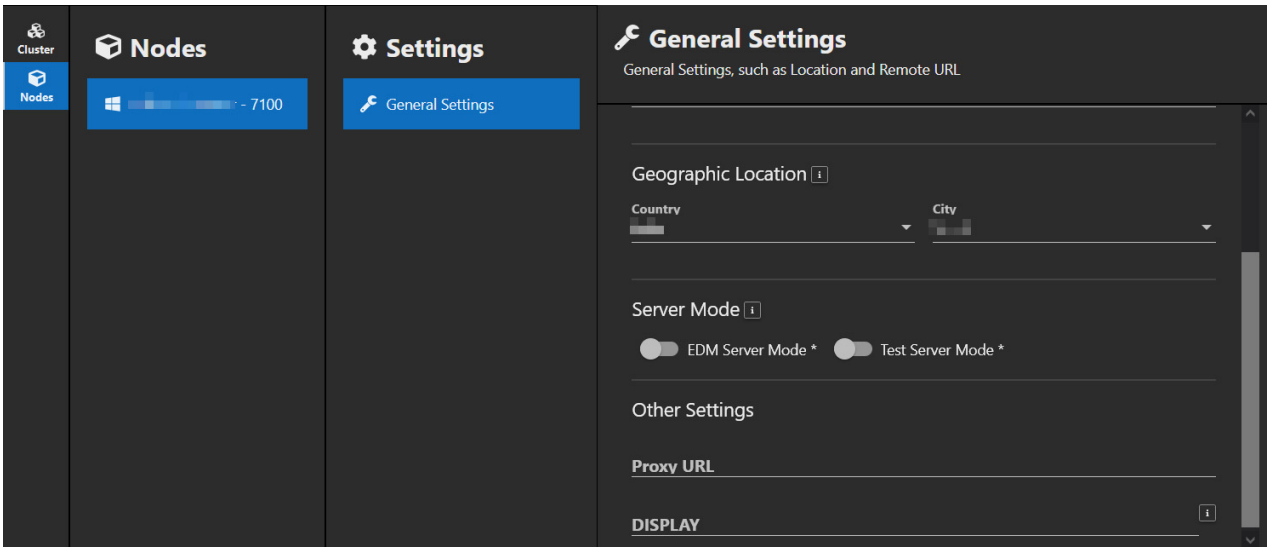
When the Pulse server starts in the ERROR state, do the following to set it up as a test server:

1. Open the Pulse web page.
   - ❑ If you installed Pulse as a service and did not reboot, access the Pulse page by typing `http://<Pulse access URL>:7100/element` in a web browser.
   - ❑ If you started Pulse by navigating to `<Cadence installation directory>/server/bin`, the Pulse web page opens automatically.
2. Specify the administration credentials to log in to Pulse.
3. Click the gear *Settings* icon on the top right of the page.

The Settings page is displayed.



4. Click *Nodes*.



The default server running is Allegro X Pulse in production mode with EDM-managed library support.

- ❑ To run the Pulse server with Allegro EDM capabilities only, toggle on the *EDM Server Mode* button. The required minimum configuration is 4 cores and 8 GB RAM.

The EDM Server Mode option is required for Design Entry HDL users who do not use Allegro X Pulse for design data management.

- To run it as a test or staging server, toggle on the *Test Server Mode* button. The minimum required configuration is 8 cores and 16 GB RAM.
- To run the Pulse server as a test or staging server with Allegro EDM capabilities, toggle on the *EDM Server Mode* and *Test Server Mode* buttons. The required minimum configuration is 4 cores and 8 GB RAM.

Note: Releases 17.4-2019, HotFix 036 and 22.1, HotFix 003 onwards, the `vista -test true` command-line argument is no longer supported.

5. Click *Save*.
6. Restart the Pulse server.

   The server starts and transitions to the RUNNING state.

Related Topics

Accessing Pulse Service Manager Web Page

Installing Pulse Primary Node as a Service

7

Monitoring of Production and Staging Server

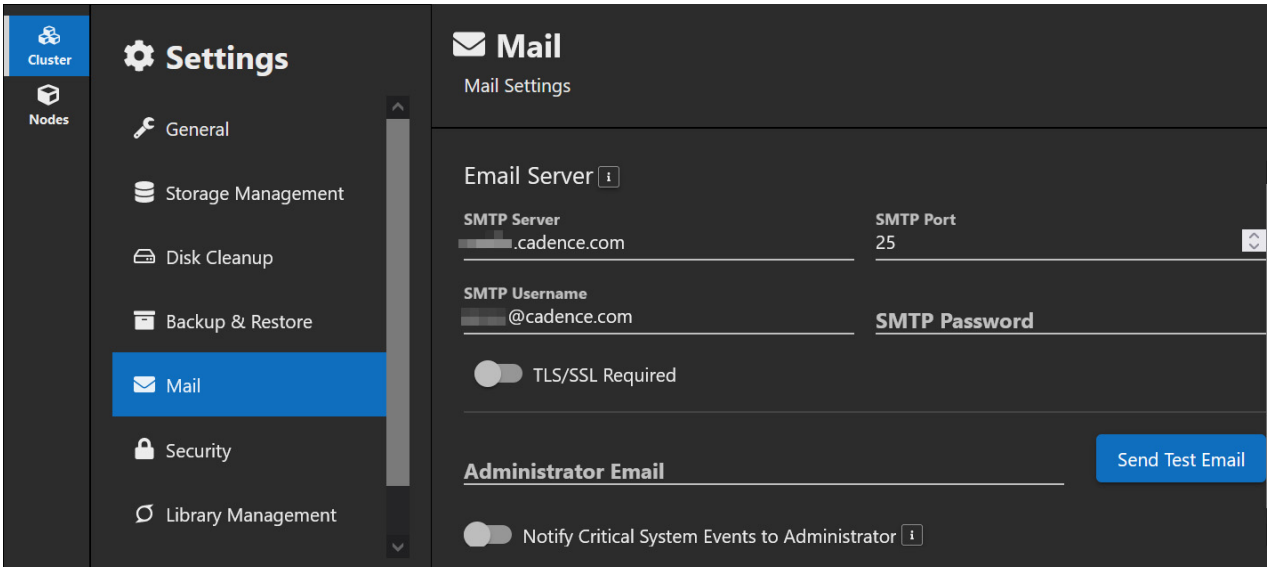
Pulse provides native monitoring systems and the ability to integrate external systems to track your operational data and identify and fix issues.

Cluster Health and Monitoring

- Pulse native monitoring covers the following:
  - Critical and non-critical services

    If a critical service, such as the Pulse metadata storage service, is down, the Pulse server goes into the maintenance mode. If non-critical services are down, a warning is displayed. Warnings do not trigger the maintenance mode.
  - Pulse node states

Email notifications are sent to administrators on state transitions.



- Host metrics

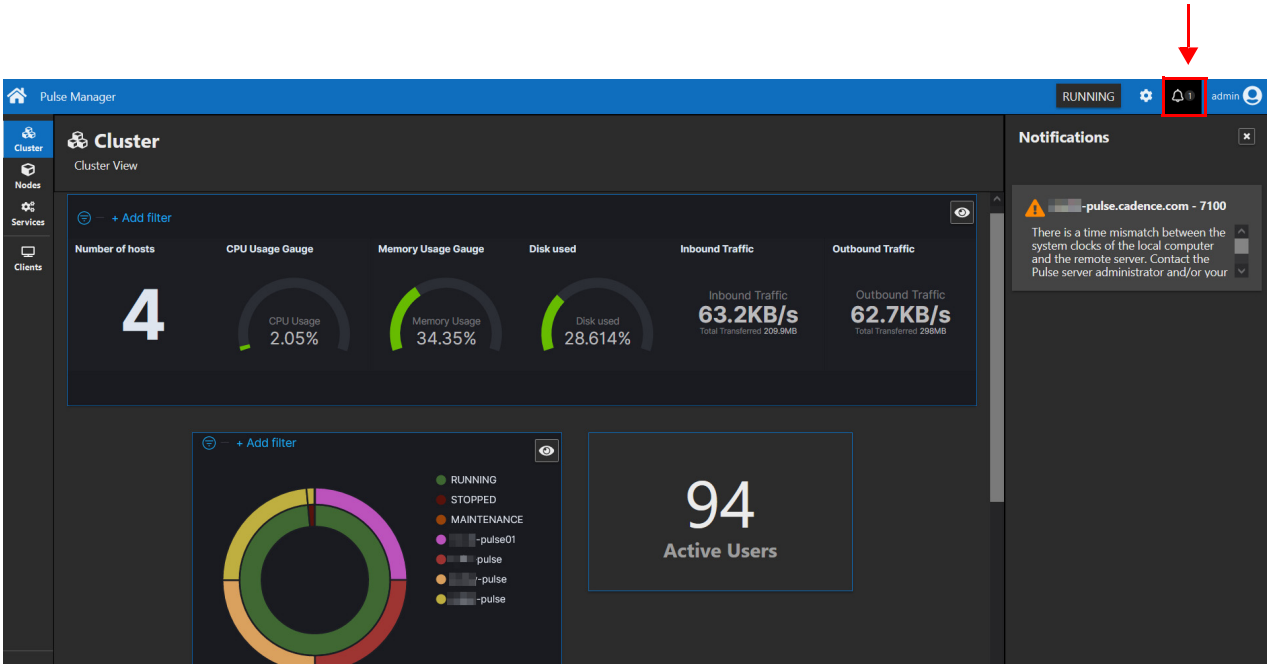
- Third-party monitoring, which includes application performance monitoring (APM) and remote monitoring, is possible in Pulse through:
  - Non-authentication health endpoint
  - Support for APM monitoring agents

Pulse Notifications

Pulse provides automated notifications using a hosted email notification service, which is configured with the corporate SMTP.

Administrator notification details are available from the Pulse Service Manager page. Review notifications for details on what triggered the maintenance mode by clicking the bell notification icon. Details are also available in logs.

Bell icon to view notifications



Pulse displays notifications for:

- Core Pulse node states
  - Running – All critical services running
  - Maintenance – Critical service error

    When the primary node goes into maintenance, the entire cluster goes into maintenance mode. Otherwise, the maintenance mode only applies to the node with the error.
  - Recovery

    While data backup restoration is in progress prior to Go Live of the Pulse server
- Critical and non-critical services
- Specific notifications
  - Primary Java Virtual Memory (VM) memory monitor

  - Auto-recovery in case of out-of-memory conditions
  - Email notifications to administrators after recovery.

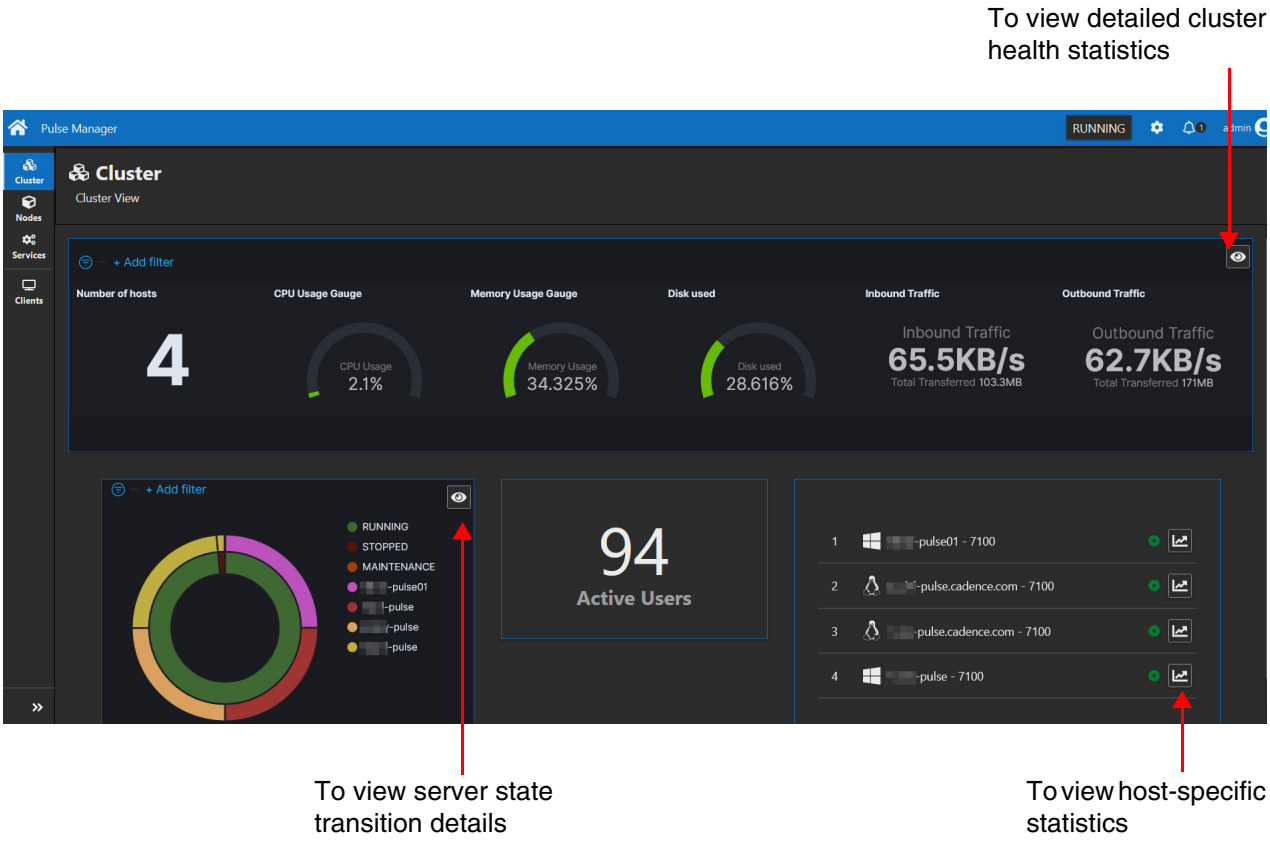
- Notifications for non-user-initiated cluster state transitions, such as when a critical service stops and fails to automatically restart, are sent as auto-emails.

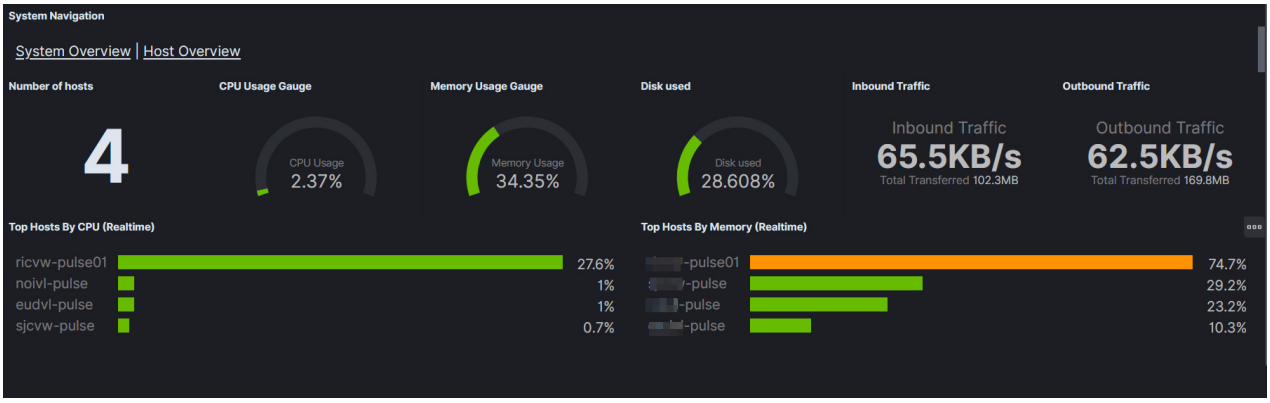
Self-Monitoring of Pulse Nodes

You can monitor the Pulse cluster status and hosts using native mechanisms in the Pulse Manager web page. The health details include:

- Cluster health overview
- Time-based historical cluster and host metrics

- Server state transitions and cluster state summary





External Monitoring of Pulse

Pulse can be integrated with enterprise application monitoring services.

- To enable remote monitoring, connect an enterprise monitoring service to the HTTP health endpoint URL, which is as follows:

- `<Pulse server URL>/api/v1/element/health`
  - For example: `https://my-pulse-server.my-company.com:7100/api/v1/element/health`
  - HTTP 200 is the standard response when the server is running smoothly. HTTP 5XX is the response for abnormal conditions.

- Local APM agent monitoring on each Pulse node
  - Independent, that is, third-party services might monitor host metrics, such as:
    - Server service state
    - CPU, memory allocation, disk usage
    - Native service metrics as supported by infrastructure inbuilt in Pulse

Related Topic

Accessing Pulse Service Manager Web Page

8

Maintaining Production Server for OrCAD and Allegro X Software

To ensure smooth running of applications, regular maintenance is a must.

For regular maintenance, do as follows:

1. Upgrade OrCAD and Allegro X software after doing the following:
   a. Check for the following dos and don'ts:
      - Make sure that all Pulse and OrCAD and Allegro X-related processes are stopped.
      - Verify that licenses are available and of the correct version.
      - Verify the integrity of the software download prior to running the installation.
      - Verify that disk space is available to complete the upgrade.
      - Review the migration guide for Allegro X platform products for new releases, and verify that all new or changed requirements are met.
   b. Verify that the software update is done correctly by starting Pulse Service Manager and checking that there are no error notifications.
   c. Uprev the EDM-managed library and Pulse databases.

      Run the `adw_uprev` command to upgrade the database to the latest version.

      Click the *Migrate* button in the Pulse Service Manager web page.
2. If you want to start using any Pulse-specific features, enable them.
3. Add or delete nodes to and from the Pulse server cluster.
4. Update server access or authentication parameters.

   For example, enable SSO or update the LDAP connection settings.
5. Update the following:

- ❑ Publish for Manufacturing templates
- ❑ Workflows
- ❑ New Part Request Form

6. Ensure that the Pulse server is ready for use by design clients.

Related Topics

- Configuration of Publish for Manufacturing-Specific Features in Pulse
- Refer to the *Allegro X Pulse and Allegro EDM* section of *Migration Guide for Allegro X Platform Products* for details on upreving the component database and updating the following:
  - ❑ Publish for Manufacturing templates
  - ❑ Workflows
  - ❑ New Part Request Form
- Configuration of Features Specific to Pulse

9

Miscellaneous Maintenance Tasks for Pulse

The following are infrequent but perhaps necessary maintenance steps:

1. Manage active users and disable or remove users who no longer require access to Pulse.
2. Track the active users.

   This includes the tracking of the total number of available and overdraft licenses, and the number of licenses used.

   You can monitor user management through the Pulse Service Manager web page. It is recommended that you ensure that excess user capacity is available.
3. If needed, create diagnostic packages, that is, the medic test case generator.

   This is only required when an issue occurs and diagnostics need to be provided to Cadence support.
4. Upgrade computing and storage resources or replace hardware resources.
   a. Ensure that Pulse services are completely stopped before any hardware or operating system maintenance.
   b. Change the host, if required.

Related Topics

Allegro X Pulse Maintenance Guide

10

Allegro EDM

Allegro EDM is a suite of products that helps you implement a collaborative environment involving design teams, methodologies, corporate design databases, and tools.

In an enterprise design chain, there are usually multiple design sites that are independent as far as design development is concerned, but are dependent on the corporate library databases for the use of approved and preferred parts, the reuse of design blocks, and so on.

For Allegro X System Capture users, library data is managed by Allegro EDM. Library development and management is done using the following Allegro EDM utilities, such as Flow Manager, Database Editor, Database Administration, Library Distribution.

Allegro Library Manager

Allegro Library Manager is used by Allegro X System Capture component engineers, CAD librarians, and library administrators who are involved in the development, management, and distribution of part libraries.

Librarians across design sites might be involved in developing, modifying, and distributing parts and their associated data to design teams for use with Allegro X System Capture, Allegro Design Entry HDL, OrCAD Capture Component Information System (CIS) for schematic symbols, and Allegro X PCB Editor for PCB footprints. As a result, a standardized library development flow and automatic update of changes to design sites is critical for enterprise library developers.

Automatic synchronization helps design centers remain up-to-date with changes to components and libraries by librarians. Using Library Manager, librarians can define graphic and parametric information, and metadata required for design tools and Allegro EDM.

Librarians can test library elements in the same environment that is used in production and can perform all the tasks that a designer performs when using the libraries. The Allegro Library Manager server, used as a central repository for librarians, can optionally connect to a product life cycle management (PLM) server for the synchronization of business metric data that provides design engineers with real-time decision data.

Some of the key benefits of Allegro Library Manager include:

- The ability to set up standard part creation methodologies using Allegro EDM Flow Manager to streamline the library development process. Librarians can create different flows with access to different tools for different types of parts.
- The ability to create and manage preferred parts, EDA parametric databases, and tool-specific EDA libraries
- A central primary library of preferred parts and approved library data that is automatically distributed to various design sites as changes are made in the library by librarians. All design sites therefore have access to up-to-date library and component information.

Library Workflows

Before you configure and start working with Allegro EDM, it is important that you familiarize yourself with the library flows that Allegro EDM supports. Allegro EDM Flow Manager is the one-stop tool that allows you to create, use, and manage library flows.

These flows are aimed at generating, assembling, and releasing all the library data elements, such as symbol, package or footprint, padstacks, and behavioral models, for the tools used in the design process. Library Manager provides the following flows:

- Library Development

  Allows you to create, validate, and release component libraries

- Library Import

  Allows you to import legacy and existing component libraries into the Allegro EDM solution. This is an important library flow for bringing legacy libraries into Allegro EDM before your designers start using Allegro EDM.

- Library Distribution

  Allows you to distribute modified and up-to-date component libraries to multiple design sites

Related Topics

Allegro X EDM Database Editor User Guide

Allegro X EDM Database Administrator User Guide

Allegro X EDM Library Distribution User Guide

Allegro X EDM Library Import User Guide

Allegro X EDM Report Generator User Guide

Allegro X EDM Data Exchange Reference Guide

Allegro X EDM Flow Manager User Guide

11

Allegro EDM Configuration

If you have customized configurations for multiple sites, you must define a root directory and create a client startup script.

Root Directory

To help manage a distributed and complex configuration environment, you must define a root directory, which stores all the information for customized configurations for multiple sites.

This root directory is referred to as Allegro EDM Conf Root or also *<Allegro EDM Conf Root>*.

The root directory has the names of the:

- Company
- Master site
- Default site

  It also has the site-level (CDS_SITE) settings for Allegro design and layout tools, such as Design Entry HDL, PCB Editor.

Client Startup Script

A client startup script in Allegro EDM is used to launch Allegro EDM Flow Manager, a utility that is, among other things, a cockpit to launch library management utilities such as Database Editor, Database Administration, Library Distribution.

The startup script, called *<startworkbench>*, also has information such as the Cadence application version you are running, the path to Allegro EDM Conf Root, the default web browser that will be used by some EDM applications, and so on.

Depending on the configuration information in *<startworkbench>*, the Allegro EDM client application, in this case a library-related utility, connects to a company and site, which in turn connect to a server.

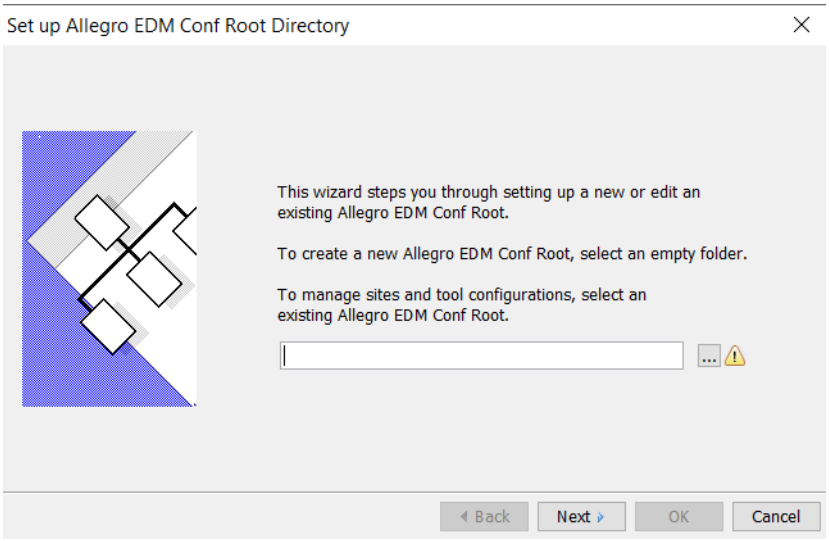
- If the server is the Pulse primary node, the client application can access all the library flow and database administration utilities and designers can modify library elements. Library elements are objects in the Cadence libraries, which can be any one of the following:
  - Schematic Models - logical parts or symbols used by a logic designer during the design capture phase.
  - PCB Models - refers to back-end models, such as shapes, padstacks, and footprints, which are required while designing a board.
  - Datasheets - support to add part datasheets to the library database.
  - Parts - an electronic representation of an off-the-shelf part with complete information about:
    - the logical data required to capture the design logic
    - physical data required to complete the physical layout
    - information required for procuring and the manufacturing the part
- If the server is a Pulse data node, the Allegro EDM client application:
  - cannot run any library-related utility on the Pulse data node.
  - can run only design-related client applications on the Pulse data node.

Defining an Allegro EDM Root Directory

To define a root directory, do the following:

1. Launch Allegro EDM Configuration Manager.

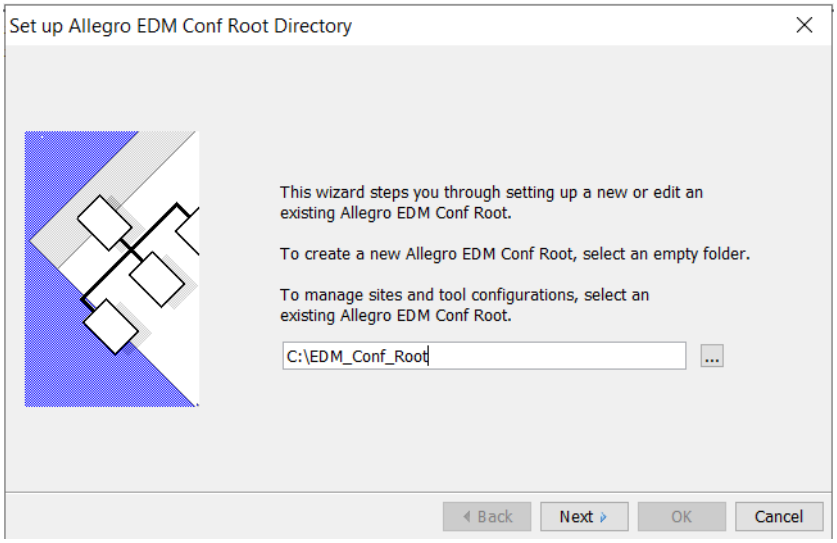
2. Click *Set up or Manage Company & Site*.



3. Do one of the following:

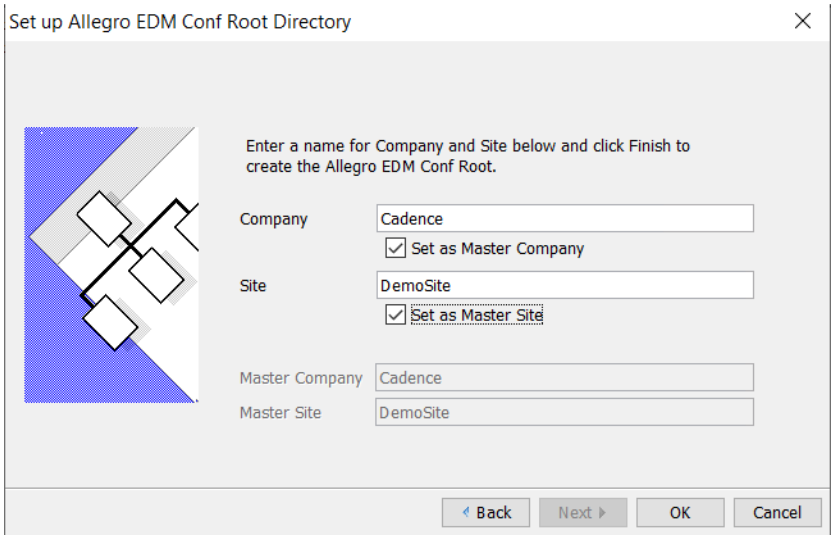
- ❑ To define a new location for Allegro EDM Conf Root, type a path or navigate to a directory.
- ❑ To modify the details of existing sites, select an existing Allegro Conf Root.

  The company and site cannot be changed when using an existing Allegro EDM Conf Root. If there is a previously defined company, sites set as the default sites are displayed in the Allegro EDM Conf Root page.



4. Click *Next* and do one of the following:

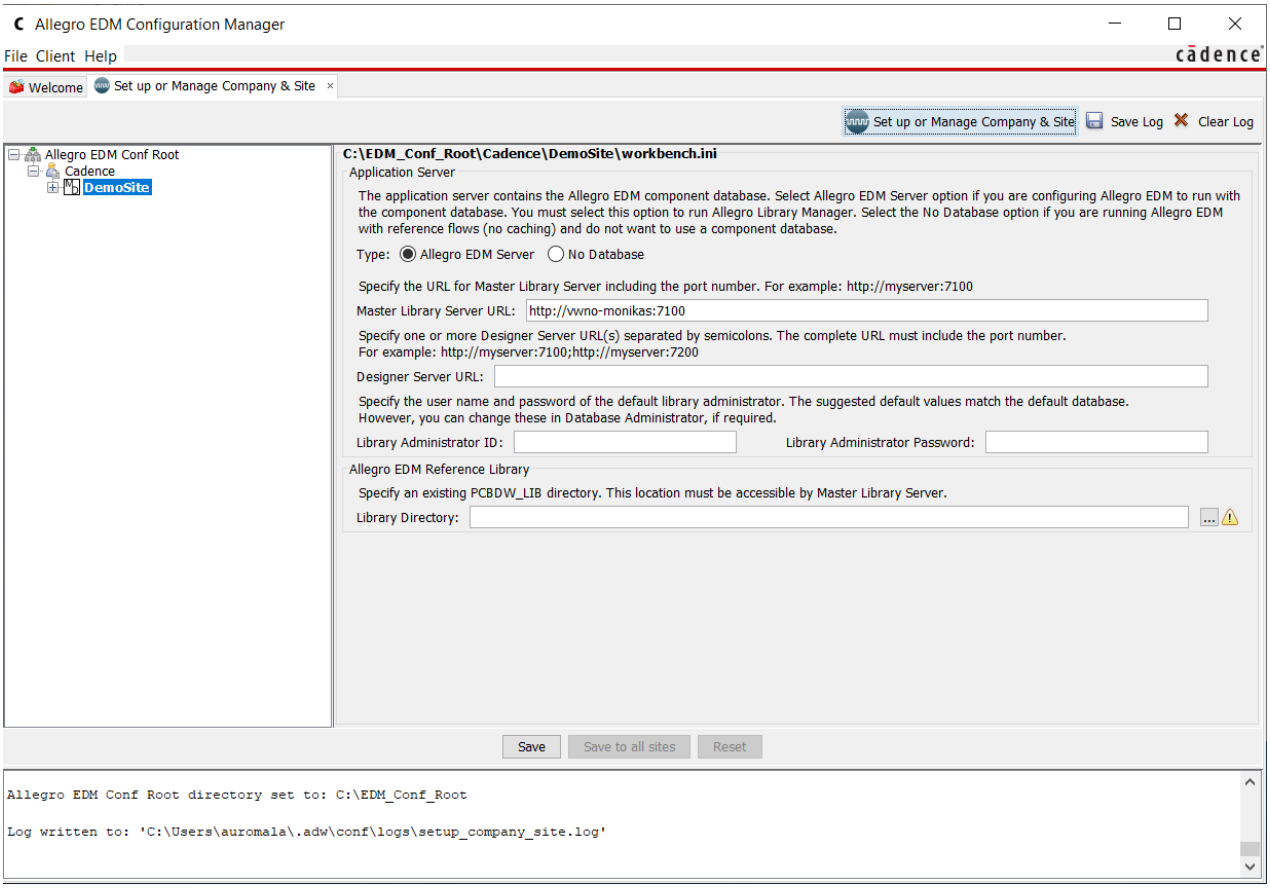
   - ❑ If you are setting up Allegro EDM Conf Root for the first time, specify the company and site details.
   - ❑ If you are modifying an existing Allegro EDM Conf Root, modify the company and site details that are automatically displayed.



A primary company and site are responsible for library development and distribution of component information to all the client sites. There can be only one primary company and site in the Allegro EDM environment. A primary site is defined by the `ATDM_MASTER_SITE` environment variable in `<Allegro EDM Conf Root>\here.tcl`.

5. Click *OK* to complete the setting up of a new Allegro EDM Conf Root.

The `workbench.ini` file interface is displayed.



This file is specific to a company and site and contains the configuration parameters for accessing the component database and reference library.

In a collaborative design environment, because each company or site can use a different Pulse primary node and reference library, you can use this file to customize the site-level settings for the database.

Note: Custom configuration settings at the company level are currently not supported.

Related Topics

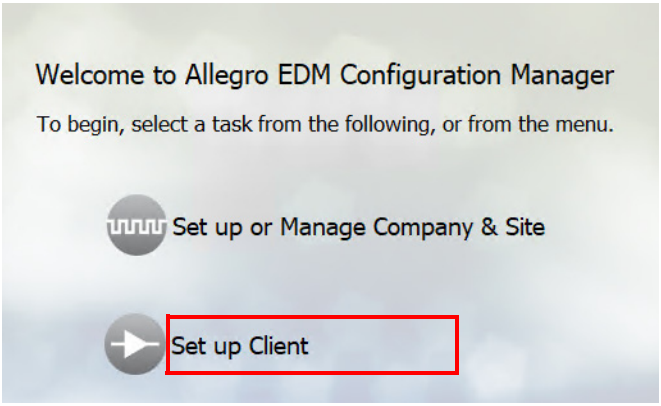
- Launching Allegro EDM Configuration Manager

Creating Client Startup Script

To create a client startup script, do the following:

1. Launch Allegro EDM Configuration Manager.
2. Click *Set up Client* on the *Welcome* screen.

   You can create a new client application or modify an existing one.

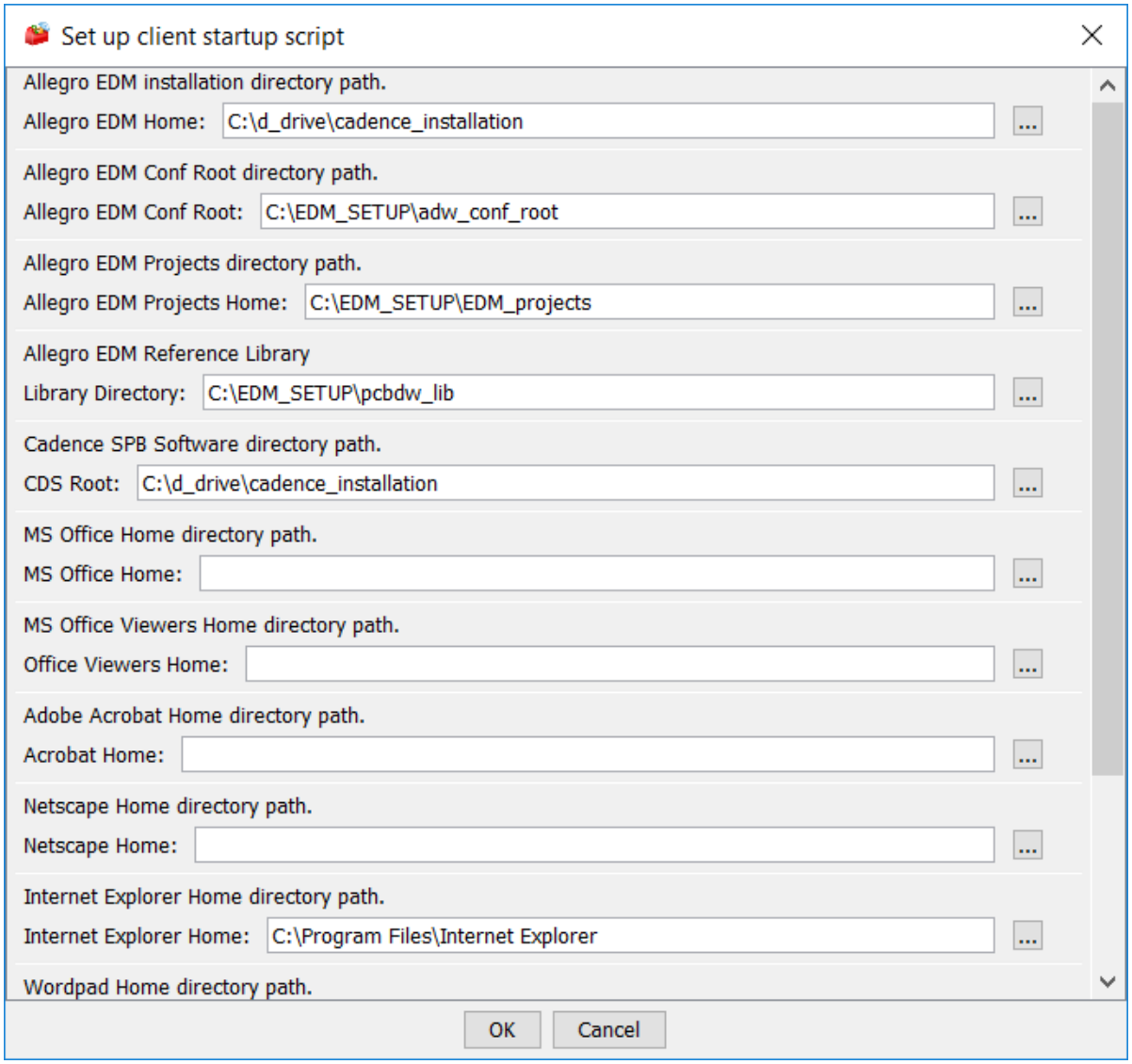


3. Specify a location for the client startup script file.
4. Do one of the following:
   - ❑ Enter a name for the new client startup script, such as `startworkbench`.
   - ❑ Select an existing client startup script file.
5. Click *Create/Edit*.

   A page is displayed to set up or modify the client startup script.
6. Specify the values for the fields in this script:
   - ❑ Allegro EDM Home
   - ❑ Allegro EDM Conf Root
   - ❑ Allegro EDM Projects Home
   - ❑ Library Directory
   - ❑ CDS Root

     Note: By default, the value of Allegro EDM Home and CDS Root is the same and is automatically populated by the script.

- ❑ MS Office Home
- ❑ Office Viewers Home
- ❑ Acrobat Home
- ❑ Netscape Home
- ❑ Internet Explorer Home
- ❑ Wordpad Home



7. Click *OK*.

The client startup script is created.

8. To access any Allegro EDM tool, double-click the client startup script to launch Allegro EDM Flow Manager.

   Flow Manager is, among other things, a cockpit through which you can access various EDM library development and management tools. Before you start working with Flow Manager, you must configure it if you work in companies with strict firewall policies.

Related Topics

- Allegro EDM Configuration
- Launching Allegro EDM Configuration Manager

Configuring Allegro EDM Flow Manager in Organizations with Controlled Computing Environment

To comply with the security policies of the IT division of your company, ECAD administrators might need to configure Java applet-based EDM applications, such as Allegro EDM Flow Manager, as secure applications.

To address the issue of security and compatibility in browser applets, you can use a Deployment Rule Set (DRS) to whitelist (mark as trustworthy) Allegro EDM Flow Manager. A DRS is a signed JAR file called `DeploymentRuleSet.jar`.

As the person responsible for maintaining the Cadence installation hierarchy (often an ECAD/CAD administrator), you need to work with the IT division of your company to create and push the Deployment Rule Set to all the machines that work with Allegro EDM Flow Manager. As an administrator, you can deploy a DRS for every system that has the EDM client. In this case, each designer must contact IT to get the DRS deployed on their machine and to whitelist their Cadence installation directory location.

Note: Currently, the Java runtime in the Cadence installation is at version 1.8. As a result, Flow Manager does not support the `-tsa` (for time stamping) option in JDK 1.8 to sign the `DeploymentRuleSet.jar`.

Important

Allegro EDM does not support the whitelisting of individual/specific folders or files in the Cadence installation directory.

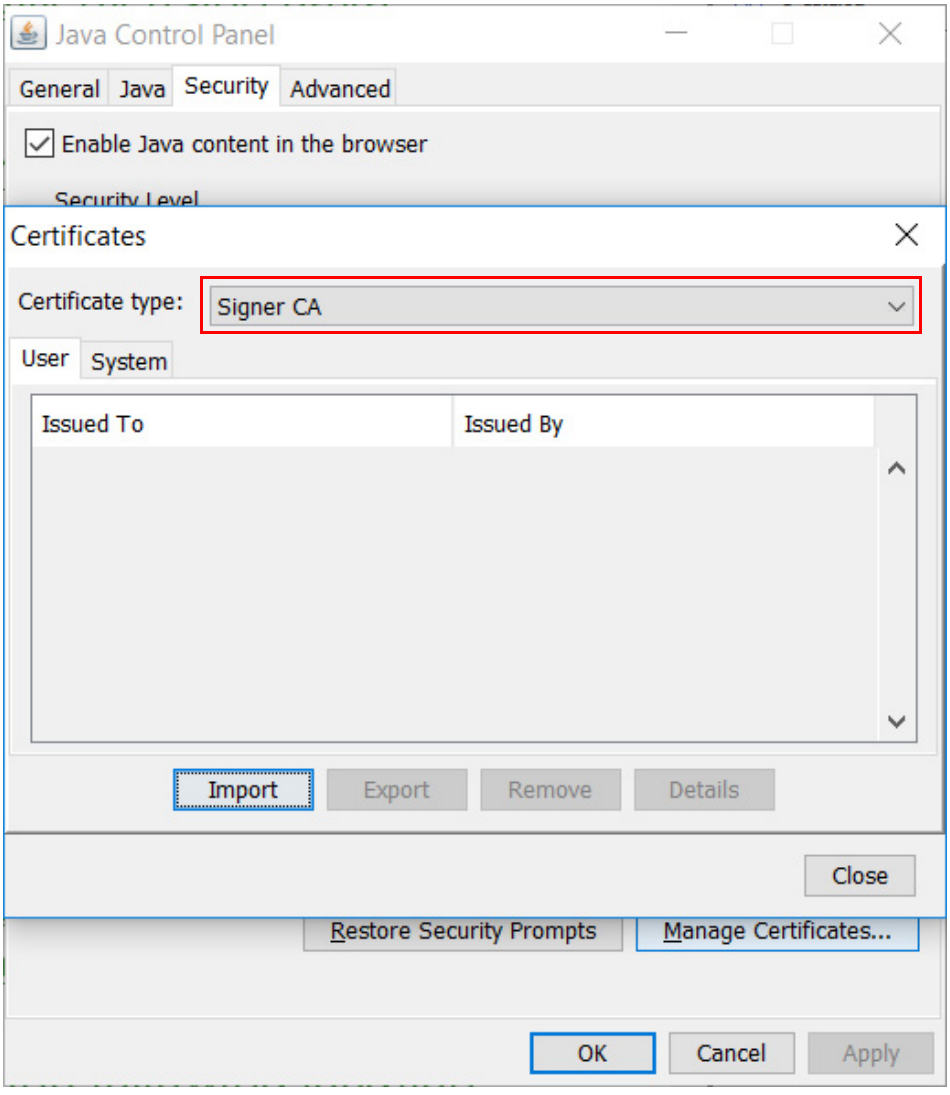
See the Oracle documentation for details about deployment rule sets and jarsigner time stamp options.

Self-Signed Certificates in the Java Deployment Rule Set

If you decide to use a self-signed certificate to create and sign the Java Deployment Rule Set, designers cannot work with Flow Manager unless you define the certificate as a trusted certificate.

To define the self-signed certificate as trusted, as the ECAD administrator, do the following:

1. Launch Java Control Panel using `<Cadence installation directory>\tools\pcbdw\jre\bin\javacpl.exe`.
2. In Java Control Panel, select the *Security* tab and click *Manage Certificates*.
3. In the Certificates dialog, select Signer CA from the *Certificate* type drop-down list.



4. Import the self-signed certificate file. You might need to set the file type filter to *All Files* to view the certificate.
5. Close the dialog.

   After you import the self-signed certificate, the `trusted.cacerts` file at the following location is updated:

   `%USERPROFILE%\AppData\LocalLow\Sun\Java\Deployment\security`

   Copy this `trusted.cacerts` file to *<Allegro EDM Conf Root>*`/`*<Company>*`/`*<Site>*`/cdssetup/projmgr/JavaDeployment`.

   This is a one-time task.

   Note: If the Allegro EDM client is set up on each individual designer's machine, each designer needs to copy the trusted certificate to their *<Allegro EDM Conf Root>*.

   To simplify this task, as the ECAD administrator, you can share the trusted certificate with the designers and they can copy it to their *<Allegro EDM Conf Root>*`/`*<Company>*`/`*<Site>*`/cdssetup/projmgr/JavaDeployment`.
6. Start Allegro EDM Flow Manager.

Launching Allegro EDM Configuration Manager

To launch Allegro EDM Configuration Manager, do the following:

1. Shut down any running instances of Pulse Service Manager (`vista.bat`).

   To know whether there are any running instances of Pulse Service Manager, hover the mouse over the Pulse icon () in your system tray and check the status. If the status is RUNNING, click the Pulse icon and select *Shutdown* from the pop-up menu.

   If you installed Pulse as a service, see Installing Pulse Primary Node as a Service for instructions on stopping the Pulse service.
2. Navigate to *<Cadence installation directory>*`\conf`.

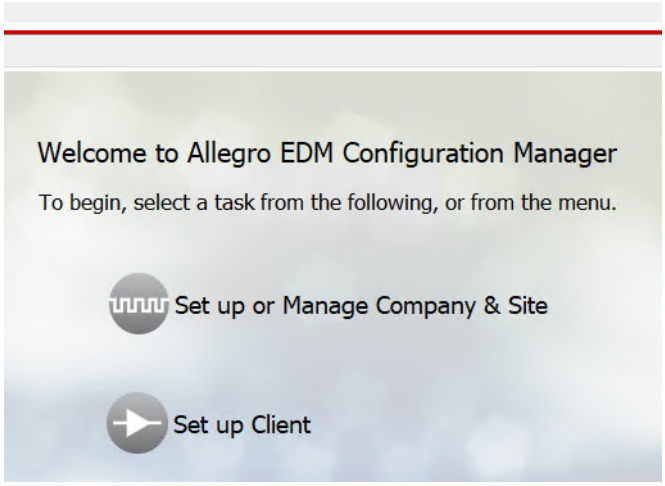
   For example: `C:\Cadence\SPB_23.1\conf`
3. Launch Allegro EDM Configuration Manager by doing the following:
   - For UNIX, run the `conf` script.
   - For Windows, run the `conf.bat` file.

On Windows 10 or later versions that have User Account Control enabled, you must launch Allegro EDM Configuration Manager using one of the following ways even if you are a local administrator:

a. Open an Elevated Windows Command Prompt.

b. Right-click the `<Cadence installation directory>\conf\conf.bat` file in the Explorer window and choose *Run as Administrator*.

The Allegro EDM Configuration Manager window appears.



Site Management in Pulse and Allegro EDM Environment

Depending on your requirements, you can define multiple sites within an organization, which can be:

- physical locations, such as London, Munich, Rome
- a mix of physical locations and functional teams, such as London, Motherboard, Memory, or just functional teams, and so on.

Allegro EDM Default or Master Sites

If multiple sites are configured in Allegro EDM Conf Root, default site is an attribute that indicates an active client site.

Local site administrators can designate their sites as the default so that all the configuration changes they make apply to their sites only.

For example, a company could have:

- Three sites: `tokyo`, `boston`, and `denver`.
- The primary site is `boston`.
- The default site is `tokyo`.

You can set the default site using Allegro EDM Configuration Manager. This information is saved in the *<Allegro EDM Conf Root>*`\here.tcl` file and is defined by the `ATDM_SITE` variable in this file.

Related Topics

Editing Site-Specific Configuration Parameters

Working with Sites in Allegro X Pulse and EDM

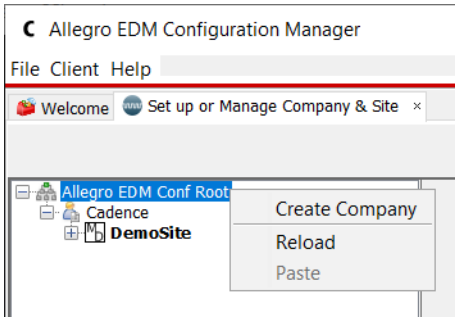
If you have multiple sites in your company, you can do the following tasks:

- Adding a Site
- Editing Site-Specific Configuration Parameters
- Comparing Two Sites Within a Company

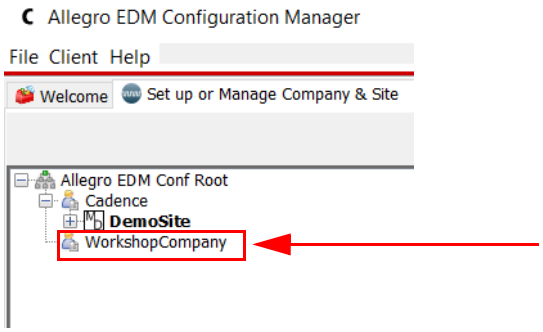
Adding a Site

To add a site in your setup and specify details, do the following:

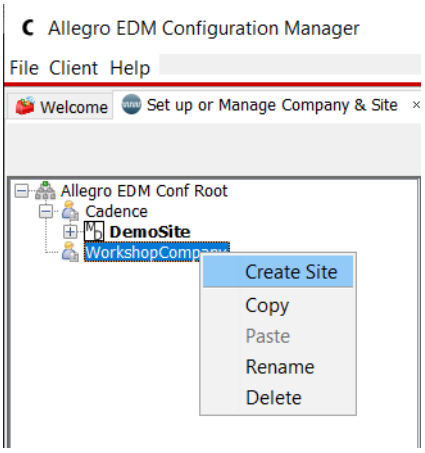
1. Launch Allegro EDM Configuration Manager.
2. Right-click *Allegro EDM Conf Root*.



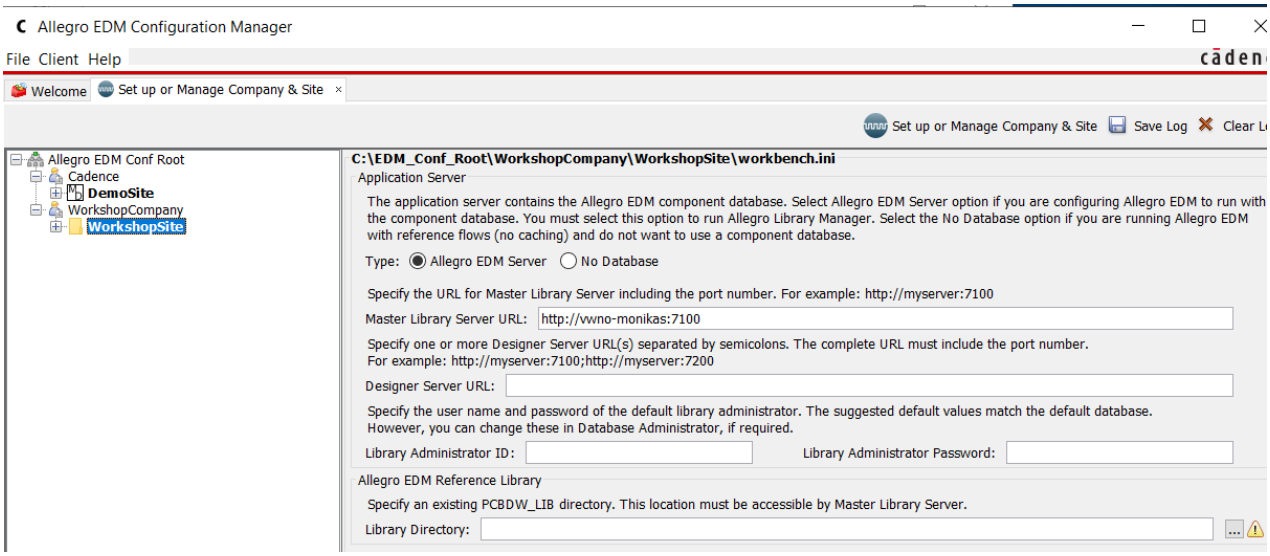
3. Choose *Create Company*.
4. Specify a name for the new company. For example, `WorkshopCompany`.
5. Click *OK*.



6. If you need to specify sites for the new company, do the following:
   a. Right-click the company name.
   b. Select *Create Site*.



A new site is added.



7. Rename an existing company or site by right-clicking its name and selecting *Rename*.
8. Delete an existing company or site by right-clicking its name and selecting *Delete*.

   If you select *Delete*, the company and all its sites and related configuration files are deleted.

Important

You cannot delete a company that contains the primary or default site. Should you need to do that, assign a site in another company as the primary or default. This makes that company the primary site. Then delete the earlier company.

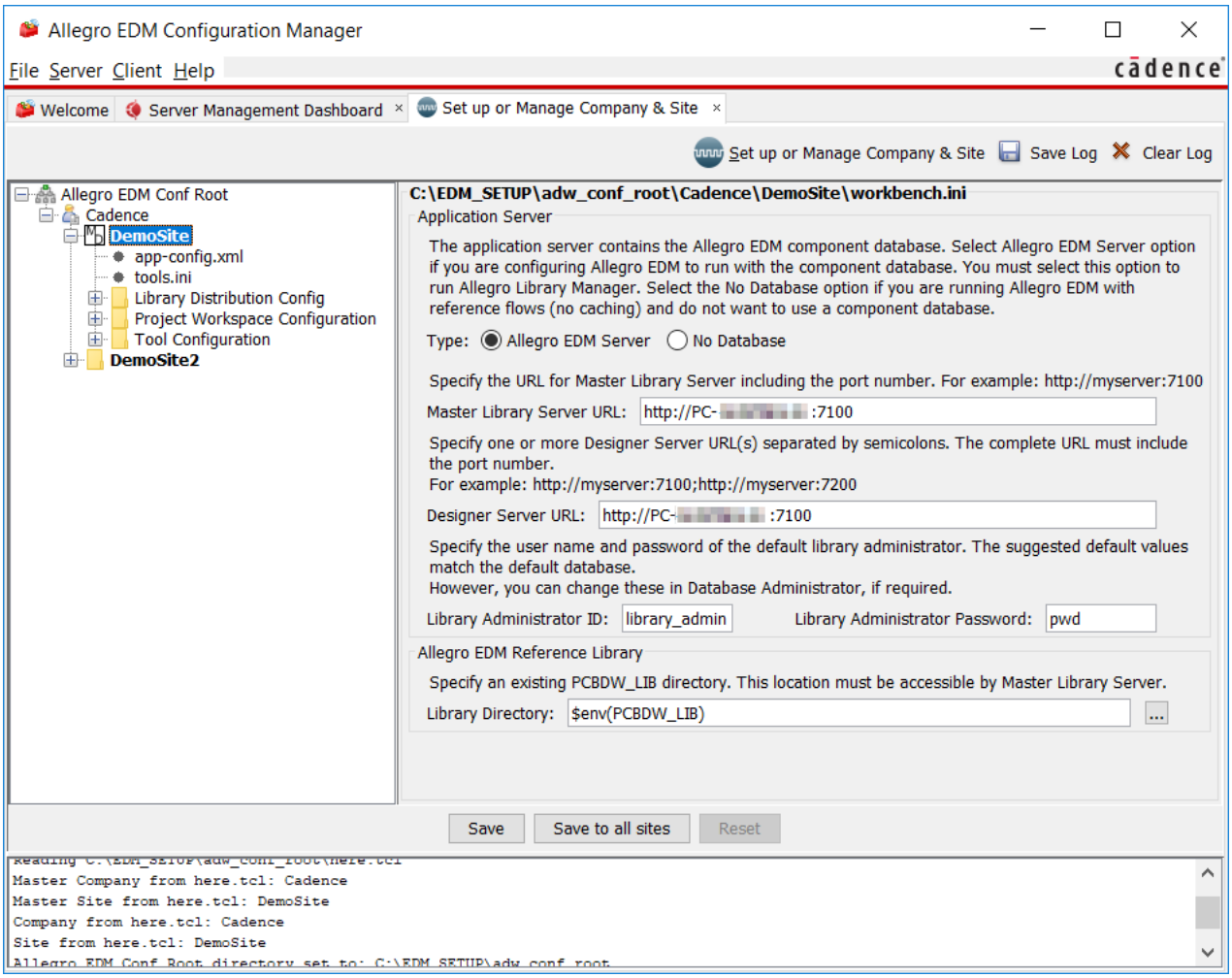
Editing Site-Specific Configuration Parameters

The Workbench Configuration File (`workbench.ini`) file contains the configuration parameters for accessing the component database and reference library. The file is specific to a company and site.

To edit the workbench configuration file for a site, do the following:

1. Launch Allegro EDM Configuration Manager.
2. In the *Set up or Manage Company & Site* tab, click the site name in the tree on the left.

The *workbench.ini* is displayed with the Allegro EDM component database and information specific to the reference library of the selected company and site.



3. Make the required changes in this configuration file.

For each configured site in the company, specify the following information:

- Type of database
  - Allegro EDM Server

    This is the default value for all Allegro EDM installations.
  - No Database

This is used for Allegro EDM Flow Manager-only installations, in which you use the `board_ref` type of project workspace that functions in the non-cache enabled mode.

- URL for the Pulse primary node.

Caution

The site-level workbench.ini file also contains the Pulse primary node URL under the adwadmin_properties and adwmetrics_properties sections, which are internally used by Allegro EDM. Do not modify these unless advised by Cadence Customer Support or Services.

- One or more Pulse data node URLs separated by semicolons. The complete URL must include the port number. For example: `http://myserver:7100;http://myserver:7200`

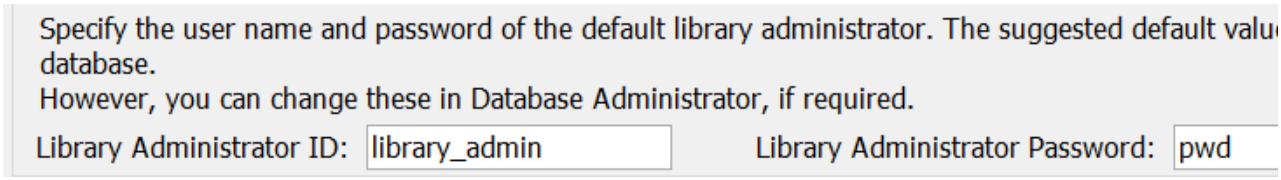
  In the *Designer Server URL* field, you can also specify a variable `$env(<variable_name>)`, where:

  `<variable_name>` is a predefined environment variable.

  For example, the value of *Designer Server URL* is `$env(my_designers)` where:

  `my_designers` is a predefined environment variable and its value is set to `http://myserver:7100;http://myserver:7200`

- Library Administrator ID and password to connect to the component database.



For Allegro EDM utilities that require login information, login credentials can be maintained in `workbench.ini` or in an encrypted form on the disk by using the `-login` option.

For example, if you do not specify credentials in `workbench.ini` and use the `-login` option to log into `adwschema`, Allegro EDM saves the information in encrypted form on the disk and uses it the next time you log into `adwschema`. When logging into Allegro EDM utilities that require login information, Allegro EDM first

checks for the encrypted information. If it is not available, it fetches user credentials from `workbench.ini`.

Note: Specifying login credentials in `workbench.ini` is not mandatory. If not specified, librarians are prompted for login credentials when they launch any application that accesses the component database.

- Location of the Allegro EDM Reference Library as specified during server setup. *<PCBDW_LIB>* is created at this location for the site.

4. Click *Save*.

Comparing Two Sites Within a Company

If you have two sites configured for a company, Allegro EDM Configuration Manager allows you to compare them to view differences between sites. This can help you decide whether you need to modify certain settings for some sites.

To compare two sites, do the following:

1. Launch Allegro EDM Configuration Manager.
2. Click *Set up or Manage Company & Site.*

   The *Set up or Manage Company & Site* tab displays the `workbench.ini` file.
3. Press the CTRL key to select both the sites together.
4. Right-click any site and choose *Compare Sites*.

   The *Comparison Results* tab appears on the right panel showing the files that differ.

Related Topic

Launching Allegro EDM Configuration Manager

Enabling Cadence-Supplied Flows for Allegro EDM

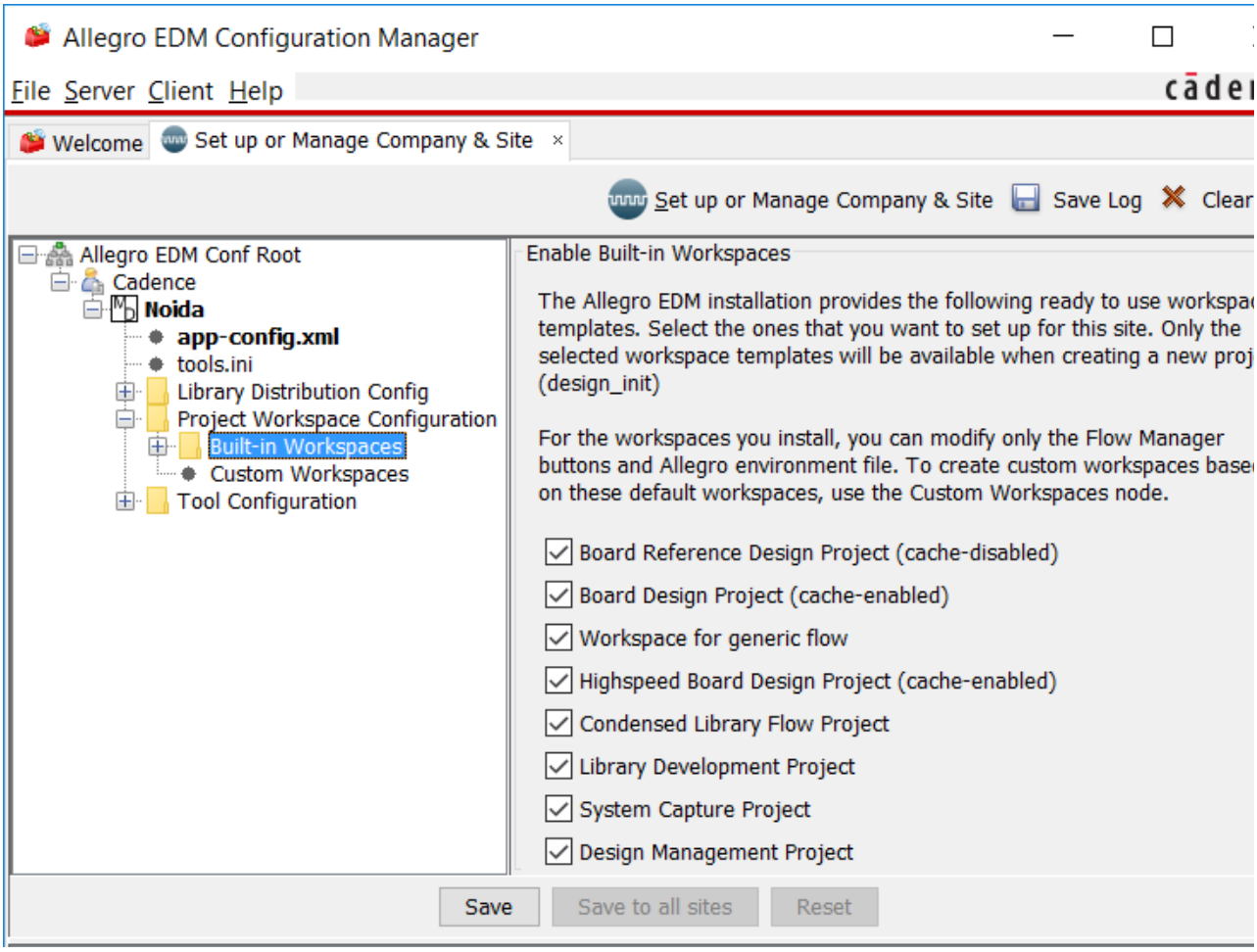
Because designers and librarians have different objectives and tasks, Allegro EDM provides various out-of-the-box PCB flows for board, reference, and high-speed schematic designs, for library development, import, and distribution.

As an administrator, you can enable the flows for your librarians or for designers in the Design Entry HDL environment.

To enable the required Cadence-supplied flow files for your environment, do the following:

1. Launch Allegro EDM Configuration Manager.
2. Click *Set up or Manage Company & Site.*

   The *Set up or Manage Company & Site* tab displays the `workbench.ini` file.
3. On the left tree panel, choose *Allegro EDM Conf Root – <company> – <site> – Project Workspace Configuration – Built-in Workspaces*.
4. Choose the required workspace templates by selecting the corresponding check box.



5. Click *Save*.

Related Topics

- Library Workflows
- Launching Allegro EDM Configuration Manager

Environment Files Management in Allegro X Pulse and EDM

The Cadence Allegro X family of tools, including Allegro X PCB Editor, relies on external environment files, which have an `.env` extension, for many of the settings that control the behavior of these tools.

Environment files specify the paths to the libraries that are searched when any of the Cadence Allegro X tools need to find design elements such as padstacks and footprint symbols.

Control of PCB Editor Environment in Allegro EDM

Environment files establish the search path for many Allegro X PCB Editor design objects, such as padstacks, footprints, script files, SKILL program files, and library verification rules. Allegro X PCB Editor needs these settings so that it knows where to look for these objects to load when placing components on a board.

The environment files contain such settings as:

- Search paths so that PCB Editor can find external design objects, such as padstacks and footprints
- Environment variable settings used by PCB Editor to control the appearance and behavior of PCB Editor. Examples include automatic save functions, plotting, and file versioning.
- Keyboard shortcuts
- Aliases
- References to other environment files that should be read by PCB Editor

Use of PCB Editor Environment File in Pulse and Allegro EDM

There is a primary environment file located in the PCB Editor installation directory: `$allegro_install_root\share\pcb\text`.

Whenever PCB Editor is launched, it looks for the environment file in this location. If the file is not found, an error message is displayed.

Note: The environment file provides basic settings that must not be changed because future Cadence patches or upgrades overwrite this file.

Customizations to the Allegro X PCB Editor paths and settings, using the environment files, should be done so that:

- the library search paths to be shared among an engineering team are set in the `$CDS_SITE/site.env` file.
- settings such as shortcuts, aliases, which are personal preferences, are set in the `$HOME/pcbenv/env` file.

Note: The environment file is modified when you use the settings editor within the Allegro X PCB Editor application.

How Allegro EDM Controls Environment Files

When Allegro EDM launches Allegro X PCB Editor layout tools such as PCB Editor, Allegro EDM controls the environment that the layout engineer uses during the design session through the use of environment files.

By controlling the environment file at startup, especially by controlling the search paths, Allegro EDM ensures that all engineers get their footprints from the same authorized library.

By ensuring consistency in the environment files, Allegro EDM ensures that each layout designer uses library elements from the same library, using the same scripts, and the same SKILL programs.

Control of the Allegro X PCB Editor environment files by Allegro EDM provides two benefits:

- Environment variables critical to the design process must be able to be locked by a CAD manager so that designers benefit from consistent and reliable library use.
- Settings that are personal to a designer and are not critical to the process must be allowed and preserved. Most designers use shortcuts and aliases that enhance their productivity. These are preserved.

To provide a consistent, error-resistant environment for layout designers, Allegro EDM can enforce control over the critical environmental variable settings. System administrator or CAD managers determine which of the environment variables are critical, but these most likely include all library path environment variables.

These library paths control access to library elements such as footprints and padstacks. These settings should not be changed by board engineers. They must remain under the control of the system administrator or CAD manager.

In Allegro EDM, the system administrator has the authority to establish and lock settings. These settings are established in the `site.env` file, and other related files, and are available in the read-only mode.

Management of PCB Editor Environment Files

PCB Editor environment files are managed as follows:

- Step 1 - Determining Types of Environment Settings Required
- Step 2 - Creating Allegro X PCB Editor Environment File
- Step 3 - Locking Critical Environment Settings
- Step 4 - Correctly Naming Allegro X PCB Editor Environment Files
- Step 5 - Putting Allegro X Environment Files in the Right Location
- Step 6 - Identifying Project Types to Use Environment Settings

Step 1 - Determining Types of Environment Settings Required

This involves deciding how many different sets of library paths are required to control Allegro X layout tools. Allegro EDM can support an unlimited number of environment files, so you can create as many as you wish. Usually, you need at least three environment files, with one each to control the library paths for the following:

- Board flows
- Library flows
- Padstack editing within the library flow

Step 2 - Creating Allegro X PCB Editor Environment File

This step involves creating an environment file that defines library path settings for each of your PCB flows. You can set up the paths for all of your Allegro X PCB Editor physical libraries in this file. You can also determine other settings to include so that they can be shared among your layout team. You can share aliases, shortcuts, and so on. A typical layout environment file has the following settings:

```
set PSMPATH = . $PAST/pad_shape \

$JEDECS/mechanical $JEDECS/format \

$JEDECS/footprints $PAST/pad_flash
```

Step 3 - Locking Critical Environment Settings

As a system administrator, you can decide settings and paths to lock. Locking an environment variable prevents a user from changing the variable during a layout session. As an example, you should consider locking the library search paths for footprints and padstacks.

This ensures that board engineers can use footprints and padstacks only from authorized libraries. You can lock the critical settings by using the `readonly` command, as shown in the example:

```
set PSMPATH = . $PAST/pad_shape \

$JEDECS/mechanical $JEDECS/format \

$JEDECS/footprints $PAST/pad_flash

readonly PSMPATH
```

Note: By setting the variable (`psmpath` in the above example) to read only, you can prevent engineers from changing that variable during the layout session.

Step 4 - Correctly Naming Allegro X PCB Editor Environment Files

For Allegro EDM to use various environment files, you must adhere to the following naming convention:

```
ADW_<unique_flow_name>.env
```

Where `unique_flow_name` is any name you give to a particular set of layout environment variables. By default, you have three flow names: `board`, `library` and `library_padstack`.

Note: This `unique_flow_name` matches a setting in each project. That is how Allegro EDM Flow Manager determines which environment file to use.

Step 5 - Putting Allegro X Environment Files in the Right Location

After you have created an environment file for each flow type, copy these files into *<Allegro EDM Conf Root>*/*<company>*/*<site>*/pcb. There are four pre-existing default environment files in the Cadence installation directory if you have a fresh installation of *<Allegro EDM Conf Root>*. If you do not have these files, do the following:

a. Save your *<Allegro EDM Conf Root>* directory with a new name.

b. Install a fresh *<Allegro EDM Conf Root>*.

c. Copy the pcb folder from the newly installed *<Allegro EDM Conf Root>* into your saved *<Allegro EDM Conf Root>*.

d. Restore the renamed *<Allegro EDM Conf Root>*.

Step 6 - Identifying Project Types to Use Environment Settings

Project types are automatically identified for newly created custom workspaces.

The tasks described in this section are needed only for custom workspaces created in older releases. To identify the environment file to use for each project in a custom workspace, do the following:

1. Set the pcb_flow_type property in the Allegro EDM workspaces.
2. Browse a workspace and locate the atdm.ini file.

   For example: *<Allegro EDM Conf Root>**<company>**<site>*\cdssetup\pcbdw\workspaces\customflow_ws\15.5\archindep\customflow\method\local\atdmdir
3. Edit the atdm.ini file as follows:

```
[design_global]

design_type          = board

gui_type             = board

pcb_flow_type        = board

design_name          = @project@

design_author        = @di_author@

design_manager       = @di_manager@
```

```
project_ppl          = @di_ppl@
project_name         = @contract@
company_name         = $env(ATDM_COMPANY)
site_name            = $env(ATDM_SITE)
reuse_module         =
```

4. Follow steps 1 to 3 for each custom workspace.

   After you modify each workspace in this manner, every project created from the custom workspaces has these settings set in `atdm.ini`.

How Allegro EDM Uses Environment Files

When Allegro EDM is installed, it creates a directory called `<Allegro EDM Conf Root>`, which contains company and site-specific subfolders. The site-specific subfolder already contains a `cdssetup` folder.

Allegro EDM also installs a folder called `pcb`: *`<Cadence installation directory>\<Allegro EDM Conf Root>\<company_name>\<site_name>`*.

This `pcb` folder initially contains the following files:

- `site.env`
- `ADW_board.env`
- `ADW_library.env`
- `ADW_library_padstack.env`

Note: The `pcb` folder is generated only when the *`<Allegro EDM Conf Root>`* directory is created during installation.

The `site.env` file sources one of the many environment files, depending on the type of the current project. For example, the `site.env` can contain:

```
ifvar ADW_PCB_FLOW_TYPE "source $CDS_SITE/pcb/
ADW_${ADW_PCB_FLOW_TYPE}.env"

ifnvar ADW_PCB_FLOW_TYPE "source $CDS_SITE/pcb/ADW_board.env"
```

The system environment variable, `ADW_PCB_FLOW_TYPE`, is set by Allegro EDM Flow Manager based on project settings. Whenever you open a project, the Flow Manager:

- reads the `atdm.ini` file.
- locates the `pcb_flow_type` setting.
- sets it as the `ADW_PCB_FLOW_TYPE` environment variable.

Handling Manufacturing Retargetability

To support retargetability, sites might use parallel libraries of footprints and padstacks designed for different technologies. Switching between these libraries can be controlled by redirecting the `psm` and `padpath` to new settings.

There are significant library management issues associated with maintaining parallel libraries. However, if you have parallel libraries in your setup, Allegro EDM can be modified to support the use of these libraries.

For example, you have two parallel footprint libraries – one for standard rules and another for highspeed rules - and these libraries are located in the following directories:

- `/sharedLocation/Cadence/Libraries/footprint_models/standard`
- `/sharedLocation/Cadence/Libraries/footprint_models/highspeed`

To manage them, do the following:

1. Create an environment file for the two design types:
   - *<Allegro EDM Conf Root>*`\<company>\<site>\pcb\ADW_standard.env`: This contains a line similar to the following:

     ```
     set PSMPATH = . /sharedLocation/Cadence/Libraries/
     footprint_models/standard

     readonly PSMPATH
     ```

   - *<Allegro EDM Conf Root>*`\<company>\<site>\pcb\ADW_highspeed.env`: This contains a line similar to the following:

     ```
     set PSMPATH = . /sharedLocation/Cadence/Libraries/
     footprint_models/highspeed

     readonly PSMPATH
     ```

2. Modify your workspaces to add the `pcb_flow_type` entry in the `atdm.ini` file.

   Note: If you do not have customized workspaces, run the `createflow` command to

create a workspace. When you create projects using the default templates of Allegro EDM, Allegro EDM automatically adds the path to the library in the project file.

If you use the `createflow` option to create a project template or workspace, ensure that you have the `ref_cds_lib` directive in the project CPM file. The value of this directive is the path to the library.

a. For standard designs, find the `atdm.ini` file at:

<Cadence installation directory>`\share\cdssetup\pcbdw\workspaces\board_ws\15.5\archindep\board\method\local\atdmdir`

b. Add `pcb_flow_type=standard` to the `[design_global]` section.

c. For highspeed designs, find the `atdm.ini` file at:

<Cadence installation directory>`\share\cdssetup\pcbdw\workspaces\highspeed_ws\15.5\archindep\highspeed\method\local\atdmdir`

d. Add `pcb_flow_type=highspeed` to the `[design_global]` section.

3. Launch the Allegro EDM Flow Manager and create a new project using each workspace.

- When you launch PCB Editor in a highspeed project, you access the `ADW_highspeed.env` file rules.
- When you launch PCB Editor in a standard project, you use the `ADW_standard.env` file rules.

Because the library flow launches tools for editing footprints as well as padstacks, two separate environment files are needed.

For this reason, if you create `ADW_standard_library.env`, you must also create a file called `ADW_standard_library_padstacks.env`.

- `ADW_standard_library.env` - used for footprint editing
- `ADW_standard_library_padstacks.env` - used for padstack editing

Important

Allegro EDM Configuration Manager does not control environment settings but allows you to make changes to these files for sites.

About tools.ini

The `tools.ini` file contains the configuration parameters for accessing common tools, PCB tools, and tools specific for a given project type. This file is available only at the site level.

The applications include:

- TCL/TK
- Java
- Cadence tools
- Database tools (Oracle)
- Document editing and management tools:
  - MS Office
  - Wordpad
  - MS Office Viewers
  - Open Office
  - Adobe Acrobat
- Browsers
  - Netscape
  - Internet Explorer

For each application, the following information is stored:

- windows/windowsnt.5

  Path to the application executable file for the Windows platforms

- windows/windowsnt.6

  Path to the application executable file for Windows

- unix/linux

  Path to the application executable file for Linux

- path_extension

  For Windows, additional folder names for this tool that are to be appended to the system path when starting Allegro EDM. The names are separated by a semicolon.

- dlib_path_extension

  For UNIX platforms, additional folder names for this tool that are to be appended to the system path when starting Allegro EDM. The names are separated by a colon.

- Environment variables
  - home_var

    The primary environment variable to be set for this tool.

  - Variables to be set for this specific tool. For example:

    ```
    CHDL_LIB_INST_DIR=$env(CDSROOT)|CONCEPT_INST_DIR=$env(CDSROOT)|PCBDW=TRUE
    ```

  Multiple variables are separated by the pipe (|) character.

Use of startworkbench Variables in tools.ini

Instead of defining the site variables again for `tools.ini`, you can use the variables that are specified in *<startworkbench>*. This saves the effort of specifying the same variables again and eliminates any mismatches between settings across files.

To use the variables set in *<startworkbench>*`.bat`, use the following syntax:

`$env(`*<variable name in startworkbench.bat>*`)`

For example:

```
$env(PCBDW_TCL_INST_DIR)
$env(PCBDW_JAVA_HOME)
$env(PCBDW_CDSROOT)
$env(CDS_LIC_FILE)
$env(PCBDW_ORA_INST_DIR)
```

Related Topic

Creating Client Startup Script

12

Configuring Allegro EDM-Managed Library

Pulse supports:

- Allegro EDM-managed libraries
- Indexed libraries for unmanaged libraries

EDM-managed libraries are based on the legacy Allegro EDM Library Manager applications and flows. In this, components go through a defined process of library development and management by CAD librarians, and distribution to design teams.

If you work with Allegro EDM-managed libraries, you can:

- develop custom footprint authoring rules for the symbol rules checking utility
- modify parameters and the severity level of library and design flow rules
- customize match files for archives of model types

  When you create a new model type in Allegro EDM Database Administrator, Allegro EDM creates a match file. The match file dictates what content is included when a library model is checked into Allegro Library Manager. The file also specifies the structure and behavior of archives of a model type during their archival and extraction in various library flow processes.

Symbol Check Rules for Footprint Verification

As a librarian, you can develop footprint authoring rules for inclusion in the symbol rules checking utility, which is part of footprint verification in the library flow. The footprint authoring rules configure the model verification rules that enforce library standards.

Source Code

The source code for symbol rules is developed using the AXL-SKILL functions available in Allegro X PCB Editor. You can create multiple functions for developing the rules. There must be one function that the rules check tool calls.

The following file calls the various functions: *<Allegro EDM Conf Root>**<company>**<site>*\pcb_symbol_vfy\15.5\rule_check_tables.il

Function Inputs

The following two variables are used as inputs in the rule set function call:

- `file_ptr`
- `log_file`

The `file_ptr` variable is used to send information to a marker file, `symchk.mkr`, which contains message and location information for any element found in a drawing. The markers file is read by the markers file viewer. The `log_file` variable is the file pointer to the log file that is generated during run time of the checks.

Function Returns

The rule function call must return a list of two elements. The first element is an integer of the number of errors found during the rule check. The second list item is the number of warnings found during the rule check.

```
return(list(errors warnings))
```

Writing to symchk.mkr

The marker file, `symchk.mkr` contains message and location information for any element found in a drawing, which might be an error, warning, or message. The function that is to be called to write to the marker file is as follows:

```
_PAC_report_errors(file_ptr error_type short_message long_message
object_kind object_name parent_name dwg_name)
```

Where:

- `file_ptr`: The pointer to the output port for the markers file

- `error_type`: Indicates the type of message to be displayed. This is an integer type element. See Global Variables for details on `rule_check_globals.il`, the global variables file.
- `short message`: An ASCII single-string entry that provides a brief description of the problems being reported.
- `long_message`: An ASCII single-string entry that gives a detailed description of the problem being reported.
- `object_kind`: An ASCII string for the name of the elements type, that is, `"PIN" "TEXT"`.
- `object_name`: An ASCII string that denotes the XY location and class or subclass description of the item. The string must be in the following format:

  `"x.x:y.y=class/subclass"`

  Example:

  `"1.000:1.250=PACKAGE GEOMETRY/ASSEMBLY_TOP"`
- `parent_name`: An ASCII string that describes the element type and its XY location. This must be in the following format:

  `"item@(x.x y.y)"`

  Example:

  `"PIN@(3.450  4.623)"`
- `drawing_name`: An ASCII string that describes a drawing or drawing element.

Example

Global variables defined:

```
TEST_CASE_ERROR = ERROR

TEST_CASE_SHORT = "Test element missing"

TEST_CASE_LONG = "Missing add test element at origin.\n Please
add."
```

The function call would be as follows:

```
_PAC_report_errors(file_ptr TEST_CASE_ERROR TEST_CASE_SHORT
TEST_CASE_LONG  "TEXT" "0.00:0.00=PACKAGE_GEOMETRY/DISPLAY_TOP"
```

```
"CIRCLE@(0.00 0.00)"  "test_symbol.dra")
```

Writing to symchk.log

The log file can be written to using the standard `fprintf` functions available in SKILL. The pointer variable to the output port for the log file is `log_file`. This pointer must be passed into the function.

Example

Function Input/Output

```
defun(your_function(file_ptr log_file)
prog(()
code
return(list(errors warnings))
))
```

Design and Schematic Model Rules with Rules Checker

Rules Checker, also referred to as Checkplus, is a utility that lets librarians check for violations of design and Schematic Model rules. The utility includes a set of default rules and design checks. Design rules for example can check for the proper placement of elements on the drawing, consistency between the logic and body drawing, properties and property values, unconnected elements, and so on.

The default rules provided by Cadence are available at:

`<Cadence installation directory>/tools/checkplus_exp/concept/rules`

Each `.rle` file at this location is a compiled version of a file written in the Rules Checker Rule Language. The source files of these compiled files have an `.erl` file extension and are available at:

`<Cadence installation directory>/tools/checkplus_exp/concept/rules_source`

Rules Checker also stores information and a short and long message about each rule violation in help message files, which have a `.h` extension. If you want to customize these messages, you must copy the help message file to Allegro EDM Conf Root.

Customizing Rules for Library and Design Flows

By default, custom rules are read from *<Cadence installation directory>*. If you want to customize a default rule for an Allegro EDM project, you need to define the location in Allegro EDM Conf Root so that Rules Checker can locate the customized rule files in your site setup.

To customize design and library flow rules, see the following:

- Configuring Allegro EDM Conf Root for Custom Rules
- Modifying Parameters in Library and Design Flow Rules
- Modifying Severity Level in Library and Design Flow Rules
- Removing a Rule from a Rule Set
- Removing a Rule from a Rule Set
- Creating a Rules Checker Rule

Configuring Allegro EDM Conf Root for Custom Rules

To configure Allegro EDM Conf Root and specify the location of the customized rules, do the following:

1. Open the `site.cpm` file from:

   *<Allegro EDM CONF ROOT>**<company>**<site>*\cdssetup\projmgr.

2. In the `START_CHECKPLUS` section of the `.cpm` file, replace `$CDS_SITE` with:

   <Allegro EDM CONF ROOT>/<company>/<site> in `RULE_SEARCH_PATH` and `INCLUDEPATH`.

   By default, the `custom_rules` path is read from the *<Cadence installation directory>* directory, so this step is needed to configure custom rules.

   ```
   rundir 'checkplus'

   VIEW_LOGICAL 'cfg_package'

   RULE_SEARCH_PATH '$ADW_CONF_ROOT/<company>/<site>/checkplus/
   rules/' '$ADW_CONF_ROOT/<company>/<site>/checkplus/
   custom_rules/rules' '$CDS_INST_DIR/tools/checkplus_exp/concept/
   rules/' '$CDS_INST_DIR/tools/checkplus_exp/concept/custom_rules/
   '

   INCLUDEPATH '$ADW_CONF_ROOT/<company>/<site>/checkplus/
   rules_include/' '$ADW_CONF_ROOT/<company>/<site>/checkplus/
   custom_rules_include/' '$CDS_INST_DIR/tools/checkplus_exp/
   concept' '$CDS_INST_DIR/tools/checkplus_exp/concept/templates'

   END_CHECKPLUS
   ```

3. To be able to customize the source rule files (`.erl`), copy the `.erl` files from *<Cadence installation directory>*\tools\checkplus_exp\concept\rules_source*<rulefile_to_be_changed>*.erl to *<Allegro EDM Conf Root>**<company>**<site>*\checkplus\custom_rules_source

4. If you want to customize the `.h`, and `.rle` files, do the following:

   a. Copy the `.h` file from:

      <Cadence installation directory>\tools\checkplus_exp\concept\rules_include/*<rule_group_file_to_be_changed>*.h

      to:

*<Allegro EDM Conf Root>**<company>**<site>*\checkplus\custom_rules_include

b. Copy the `.rle` file from:

<Cadence installation directory>\tools\checkplus_exp\concept\rules\<compiled_rule_file>.rle

to: *<Allegro EDM Conf Root>**<company>**<site>*\checkplus\custom_rules\rules

Modifying Parameters in Library and Design Flow Rules

To customize a default rule for an Allegro EDM project by modifying the rule parameters, do the following:

1. Complete the steps in Customizing Rules for Library and Design Flows.
2. Edit the `<rule_group>.h` rule parameter files, which are available at:

   *<Allegro EDM Conf Root>**<company>**<site>*\checkplus\custom_rules_include.

   For example, if you want to modify the `body_exceeds_max_size` rule, you find it in the `body_drawing_checks` rule group. So, copy `body_drawing_checks.h` to:

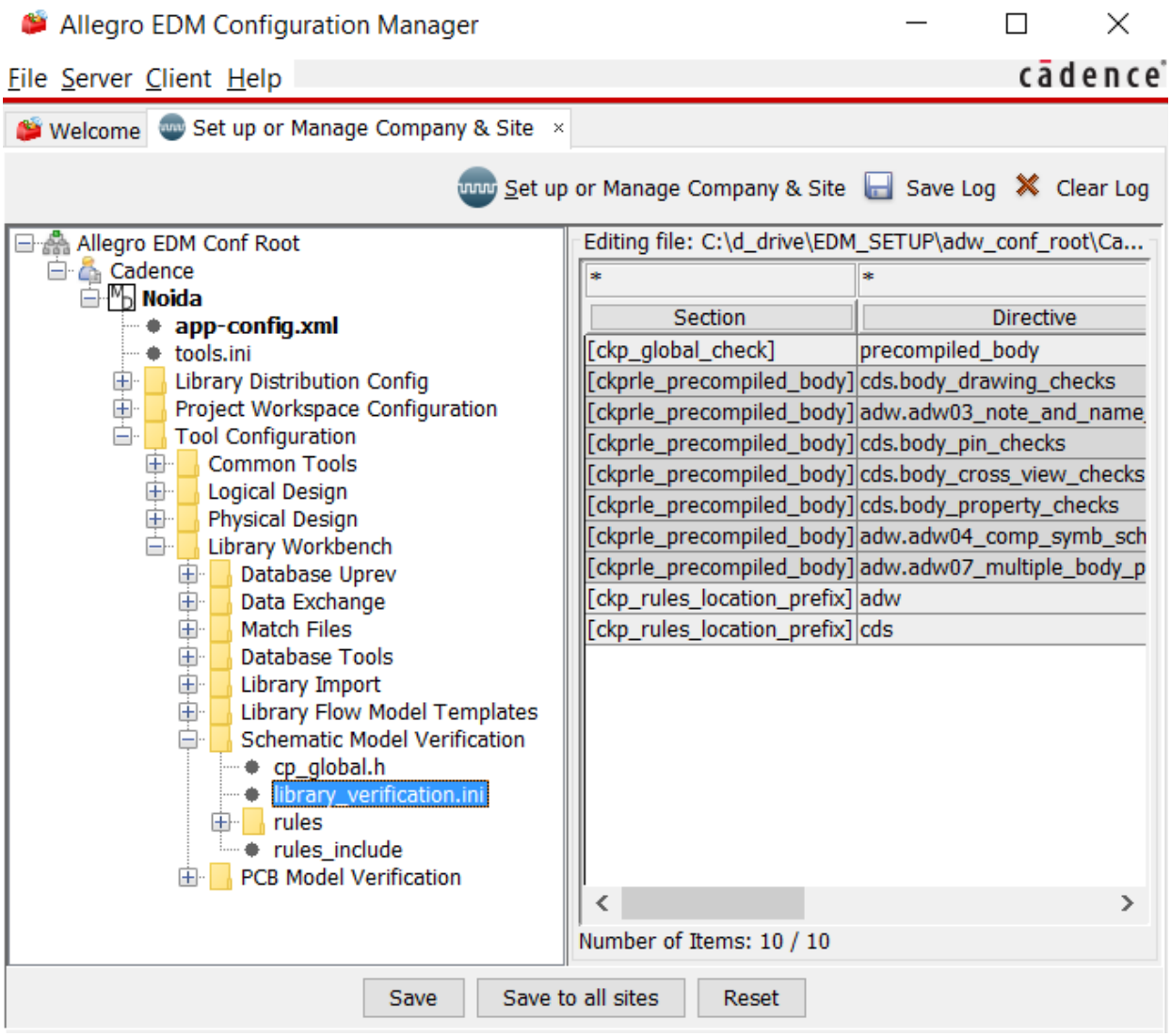
   *<Allegro EDM Conf Root>**<company>*/*<site>*\checkplus\custom_rules_include.

   Search for `body_exceeds_max_size` and modify the required parameter values:

   ```
   PARAM MAX_SPECIFIED_HEIGHT 2000

   PARAM MAX_SPECIFIED_WIDTH  2000
   ```

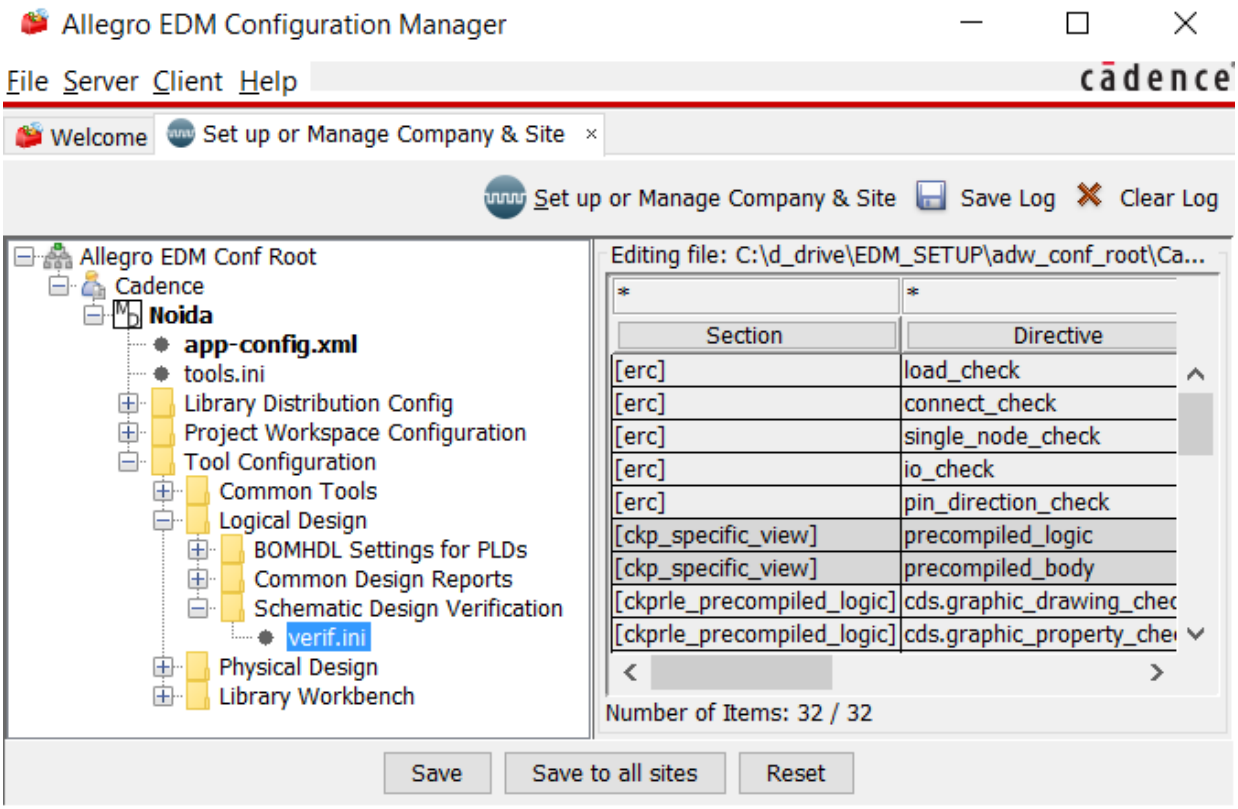
3. Save this parameter file.
4. Launch Allegro EDM Configuration Manager.
5. Click *Set up or Manage Company & Site*.
6. Do one of the following:
   - If you modified the `body_exceeds_max_size` rule for a library flow, navigate to:

Allegro EDM Conf Root – <company> – <site> – Tool Configuration – Library Workbench – Schematic Model Verification – library_verification.ini.



- If you modified the `body_exceeds_max_size` rule for a design flow, navigate to:

Allegro EDM Conf Root – <company> – <site> – Tool Configuration – Logical Design – Schematic Design Verification – verif.ini.



7. Click the *Save* button on the right panel.
8. Open one of the following and make changes:
   - If you modified the `body_exceeds_max_size` rule for a library flow, open:

     <Allegro EDM Conf Root>`\`*<company>*`\`*<site>*`\library_verification\15.5\library_verification.ini`.
   - If you modified the `body_exceeds_max_size` rule for a design flow, open:

     <Allegro EDM Conf Root>`\`*<company>*`\`*<site>*`\verification\15.5\verif.ini`.
9. Double-click the *Site Value* column.
   a. Add the path to the custom rule:

      `[ckp_rules_location_prefix]`

```
myrules = $ADW_CONF_ROOT/<company>/<site>/checkplus/
custom_rules
```

b. Add the custom rule:

```
[ckprle_precompiled_body]

myrules.body_drawing_checks = on
```

10. Launch Allegro EDM Flow Manager.

11. Run Rules Checker.

The new rule is loaded and executed.

Inclusion of Custom Rules in Symbol Rules Checking Utility

To include rules in the symbol rules checking utility, open the `rules_check_tables.il` file. The file is in the local SKILL directory where the symbol check command files are stored.

The file is a function that returns a list of lists. The format of the list cannot be changed. The first list level is the inclusion of all the lists, with the next level being the category or rules class definition. The last level of lists is lists of paired values, the rule name, and the specific SKILL function call for that rule.

```
List(
    list( category1
    list( rule_name1 skill_function1)
    list( rule_name2 skill_function2)
    )
    list( category2
    list( rule_name1 skill_function1)
    list( rule_name2 skill_function2)
    list( rule_name3 skill_function3)
    )
    list( category3
    list( rule_name2 skill_function2)
```

```
   list( rule_name1 skill_function1)
   )
)
```

Category Example 1

```
list(
   ;; Added the origin and geometry rules check
   list( "Orientation"
      list( '("Origin Check" "check_origin(file_ptr)")
      ;;'("Orientation Check" "check_orientation(file_ptr)")
         '("Orientation Check NEW" "check_orientation(file_ptr)")
      )
   )
```

Where:

- `Orientation` is category 1.
- `Origin Check` is the rule name.
- `check_origin` is the SKILL function to which the `file_ptr` parameter is passed.

To delete a rule, either remove the entry or add `;;` before it.

Similarly, you can add a new rule as shown: `("Orientation Check NEW" "check_orientation(file_ptr)")`

Category Example 2

```
      list( "misc"
      list( "Text Rotation" "check_text_rotation(file_ptr)")
      list( "Print All Text" "print_out_all_text(file_ptr
      log_file)")
      )
```

Global Variables

Global variables for setting messages and error or checking type values should be defined in the `rule_check_globals.il` file. This file is a central global variable value repository for all checking functions. Formats can be observed in the file and can be copied or modified as required.

Note:

- Do not change the values for ERROR, WARNING and INFO definitions.
- Do not remove or change the name of existing GLOBAL variable defined in this file. The values may be changed, but do not change the variable name.

The SKILL program file created with the custom rules set should be saved in a directory to which users, who use the rules, have access. A suggested location is the directory where the current rule set and global file reside.

Custom Rules Initialization

To initialize custom rules, the `allegro.ilinit` file, which is in the `pcbenv` directory must be edited. The Allegro X tool environment uses this file to locate SKILL program files.

Edit this file by adding a line at the bottom of the file using the load function call. A literal path might be required.

```
load("my_test_program.il")
```

Modifying Severity Level in Library and Design Flow Rules

Every rule included with Rules Checker has a default severity level of `Fatal`, `Error`, `Warning`, `Oversight`, or `Info`. The severity is defined in the *<rulefilename>*`.h` file.

You can only modify the severity level for custom rules; you cannot define a new severity level. Rules Checker stops checking when it encounters a fatal error.

To modify the severity level of a rule for a library or design flow, do the following:

1. Complete the steps defined in Customizing Rules for Library and Design Flows.
2. Modify `cp_config.h` located at:

   <Allegro EDM Conf Root>`\`*<company>*`\`*<site>*`\checkplus\custom_rules_include`.

For example, the `invalid_part_name` rule is in the `body_cross_view_checks.h` rule group. If you need to modify the severity of this rule, you must remove the existing text in `cp_config.h` and add the following entries:

```
STARTENV  Body

STARTRULE  invalid_part_name

SEVERITY  BODY_INVALID_PART_NAME_SEVERITY
Warning

ENDRULE

ENDENV
```

For example, if you need to modify the severity of the `cap_check` rule, which is in the rule group called `electrical_checks.h`, you must remove the existing text in `cp_config.h` and add the following entries:

```
STARTENV  Logical

STARTRULE cap_check

SEVERITY CAP_CHECK_SEVERITY                Error

ENDRULE

ENDENV
```

3. Save the `cp_config.h` parameter file.
4. Launch Allegro EDM Configuration Manager.
5. Click *Set up or Manage Company & Site*.
6. Do one of the following:
   - If you modified `invalid_part_name`, navigate to:

     Allegro EDM Conf Root – <company> – <site> – Tool Configuration – Library Workbench – Schematic Model Verification – library_verification.ini.
   - If you modified `cap_check`, navigate to:

     Allegro EDM Conf Root – <company> – <site> – Tool Configuration – Logical Design – Schematic Design Verification – verif.ini.
7. Click the *Save* button on the right panel.
8. Do one of the following:

- Open *<Allegro EDM Conf Root>**<company>**<site>*\library_verification\15.5\library_verification.ini.
- Open *<Allegro EDM Conf Root>**<company>**<site>*\verification\15.5\verif.ini.

9. Add the path to the custom rule as follows:

```
[ckp_rules_location_prefix]

myrules = $ADW_CONF_ROOT/<company>/<site>/checkplus/
custom_rules
```

10. Add the custom rule:

```
[ckprle_precompiled_body]

myrules.body_cross_view_checks = on
```

11. Launch Allegro EDM Flow Manager.
12. Run Rules Checker.

    The new rule is loaded and executed.

Removing a Rule from a Rule Set

You can modify Library and Design Flow Rules by removing a rule from a rule set. To remove a rule from a rule set, do the following:

1. Complete the steps in Customizing Rules for Library and Design Flows.
2. Delete the required rule from a rule set by either:
   - modifying the source rule (`.erl`) in a text editor. See Modifying Source Rules in a Text Editor.
   - launching Rules Checker from Allegro EDM Flow Manager and modifying the rules. See Modifying Source Rules by Configuring Rules Checker.

Modifying Source Rules in a Text Editor

1. Delete the text for the rule you want to remove from a particular rule set. For example, remove the `physical_to_body_check` rule, which is part of the `body_cross_view_checks` rule set.

   To remove the rule, open the following in a text editor:

*<Allegro EDM Conf Root>**<company>*/*<site>*\checkplus\custom_rules_include\body_cross_view_checks.erl

Remove the physical_to_body_check rule. To remove it, delete the entire rule starting from the rule name to highlighted object, as illustrated:

```
/*****************************************************************************
RULE NAME : physical_to_body_check
SYNOPSIS  :
        Check that the each pin name in packaging exists in body drawing
DESCRIPTION :
Check for the presence of PACK_TYPE property on the design
   If no PACK_TYPE property is specified, perform checks on all packages in chips.prt file
   else perform checks on the package(s) corresponding to the PACK_TYPE in the design
foreach packaging
            Get pinname(term) in package file
            Get the pin number of these terms
            Get the Sections in which these pin numbers fall
                foreach Section
                   Get the Section pins
                   Get the Terms of the pins above
                  if number of terms got above is greater than number of pins on symbol
                   Pins in the chips.prt(extra) not matching with those on body, store the difference in val_pin_mismatch
else if number of terms is lesser than those on body drawing
There is a mismatch in the pins in a section, i.e., no single section carries all the body pins, store the difference in val_pin_mismatch
           else if number of terms is equal to those on body drawing
             (check by counting the elements of val_pin_mismatch, if NULL, then equal)
                   then exit out of loop (passed,no error)
                endfor
       Check for the count of sections of pins failing and the sections actually checked
         If equal , proceed (i.e. None of the sections match with bodypins)
endfor
foreach package in a list of packages got from above,
  Repeat all the steps above
          Give out an error appended with val_pin_mismatch
```

```
      endfor
DEFAULT MESSAGE SEVERITY :  Error
DEFAULT_SHORT_ERROR_MESSAGE :
 Pin name(s) not found in body/chips.prt
DEFAULT_LONG_ERROR_MESSAGE :
Body pin(s) do not a have a matching section in the packtype(s)
         Package(s): <packaging>
 Pin name not found in body/chips.prt
         Pin Name  : <pin name in chips_prt>
Body      : <design-name>
HIGHLIGHTED OBJECT : physPackType
*******************************************************************************/
```

2. Save the `body_cross_view_checks.erl` source file.
3. Launch Allegro EDM Configuration Manager.
4. Click *Set up or Manage Company & Site*.
5. Do one of the following:
   - If you modified `body_cross_view_checks.erl` for a library flow, navigate to:

     Allegro EDM Conf Root – <company> – <site> – Tool Configuration – Library Workbench – Schematic Model Verification – library_verification.ini.
   - If you modified `body_cross_view_checks.erl` for a design flow, navigate to:

     Allegro EDM Conf Root – <company> – <site> – Tool Configuration – Logical Design – Schematic Design Verification – verif.ini.
6. Click the *Save* button on the right panel.
7. Open one of the following and make changes:
   - If you modified `body_cross_view_checks.erl` for a library flow, open:

     <Allegro EDM Conf Root>`\`*<company>*`\`*<site>*`\library_verification\15.5\library_verification.ini`.
   - If you modified `body_cross_view_checks.erl` for a design flow, open:

     <Allegro EDM Conf Root>`\`*<company>*`\`*<site>*`\verification\15.5\verif.ini`.
8. Double-click the *Site Value* column.

9. Add the path to the custom rule:

```
[ckp_rules_location_prefix]

myrules = $ADW_CONF_ROOT/<company>/<site>/checkplus/
custom_rules
```

10. Add the custom rule:

```
[ckprle_precompiled_body]

myrules.body_cross_view_checks = on
```

Modifying Source Rules by Configuring Rules Checker

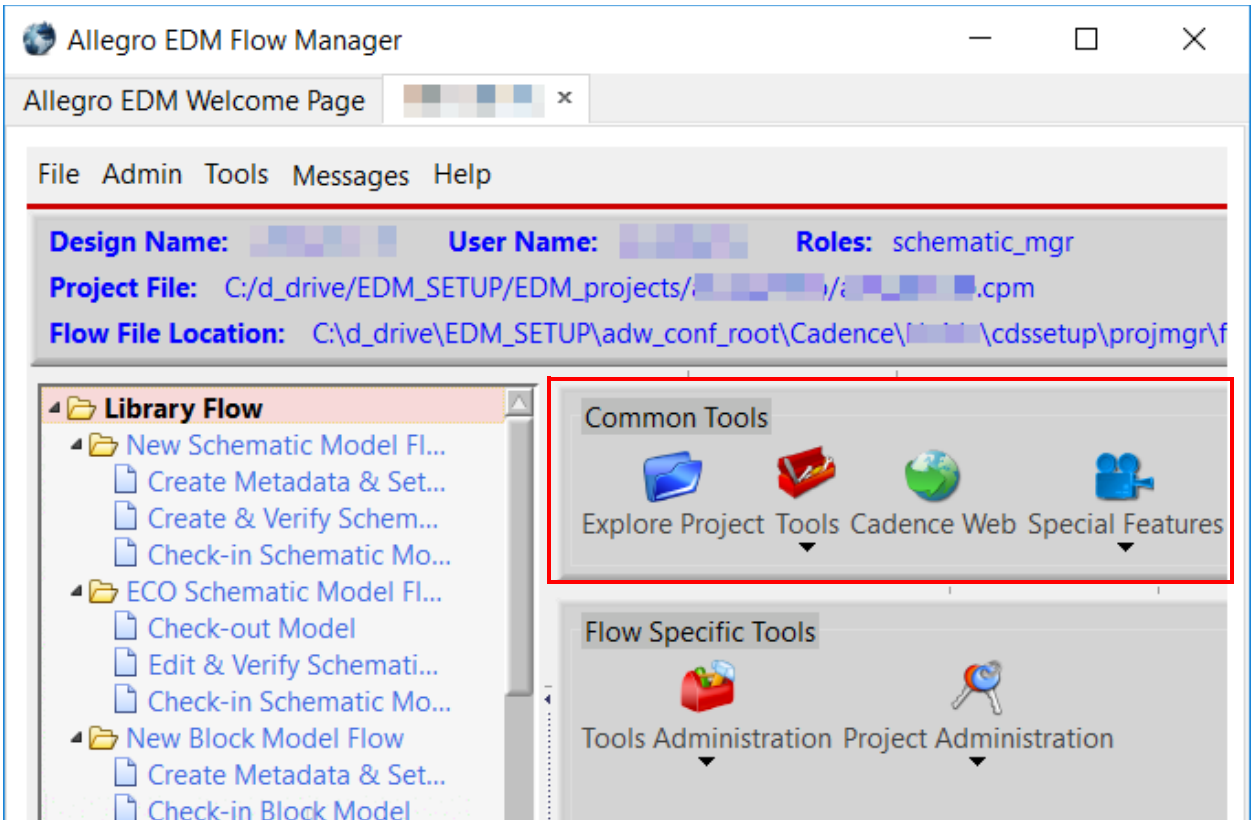
You can modify source rules by configuring Rules Checker in Allegro EDM Flow Manager. To modify a rule using Rules Checker, you need `checkplusui.exe`. This executable file is not part of the out-of-the-box library flow in Allegro EDM.

You can launch Rules Checker as a standalone utility or add a button for this executable file in your library flow using Allegro EDM Flow Manager.

To customize any flow in Flow Manager, you must be defined as a flow administrator in the `flowmanager.properties` file. The `flowmanager.properties` file is provided by default in Allegro EDM Flow Manager. It is available at *<Allegro EDM Conf Root>/<Company>/<Site>*`/cdssetup/projmgr/flows`.

Typically, companies define an ECAD administrator as a flow administrator. To define a user as a flow administrator, you must have write access to the `flowmanager.properties` file. This permission is typically provided when Allegro EDM is installed and configured. To define a user as a flow administrator, add the login ID of the user you want to define as a flow administrator in the `admin` variable.

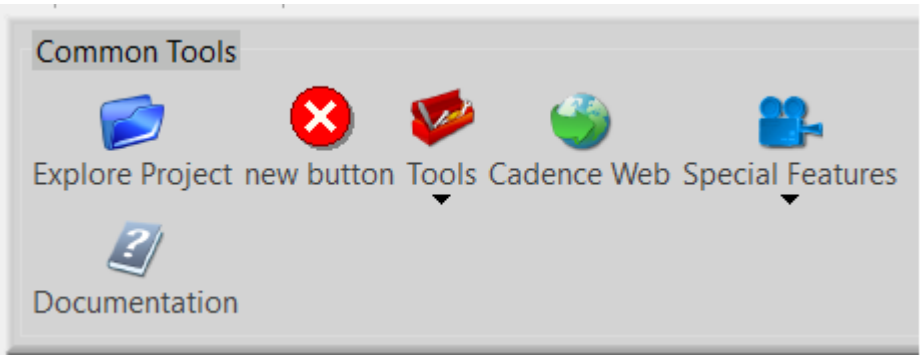
Typically, custom buttons for a particular flow are added to the Common Tools pane.



To add a button to the Common Tools pane, do the following:

1. Right-click any existing icon in the Common Tools pane.
2. Choose *Insert New*.

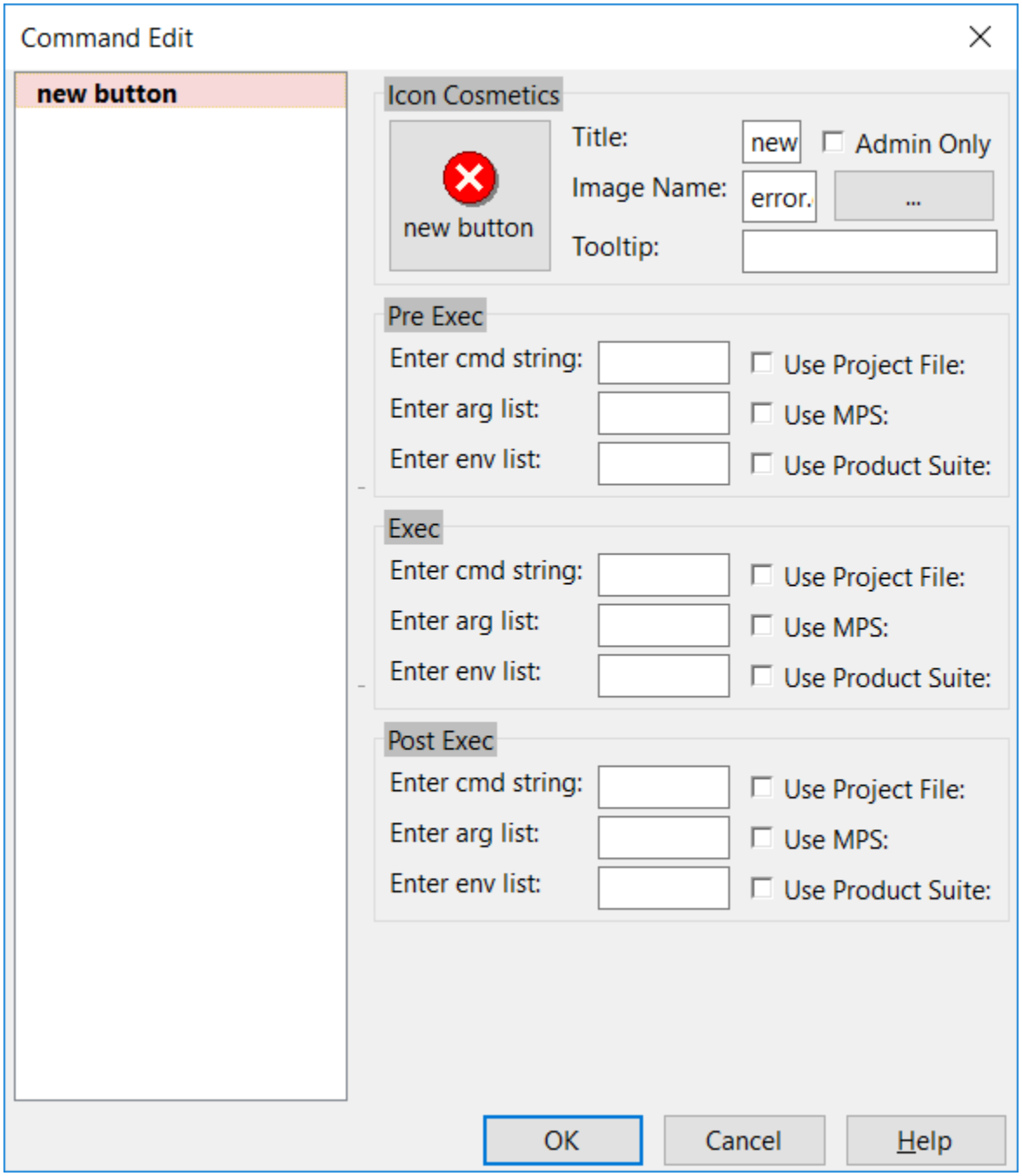
   A new button is added in the pane.



3. Right-click the new button.

4. Choose *Properties*.

   The Command edit dialog opens.

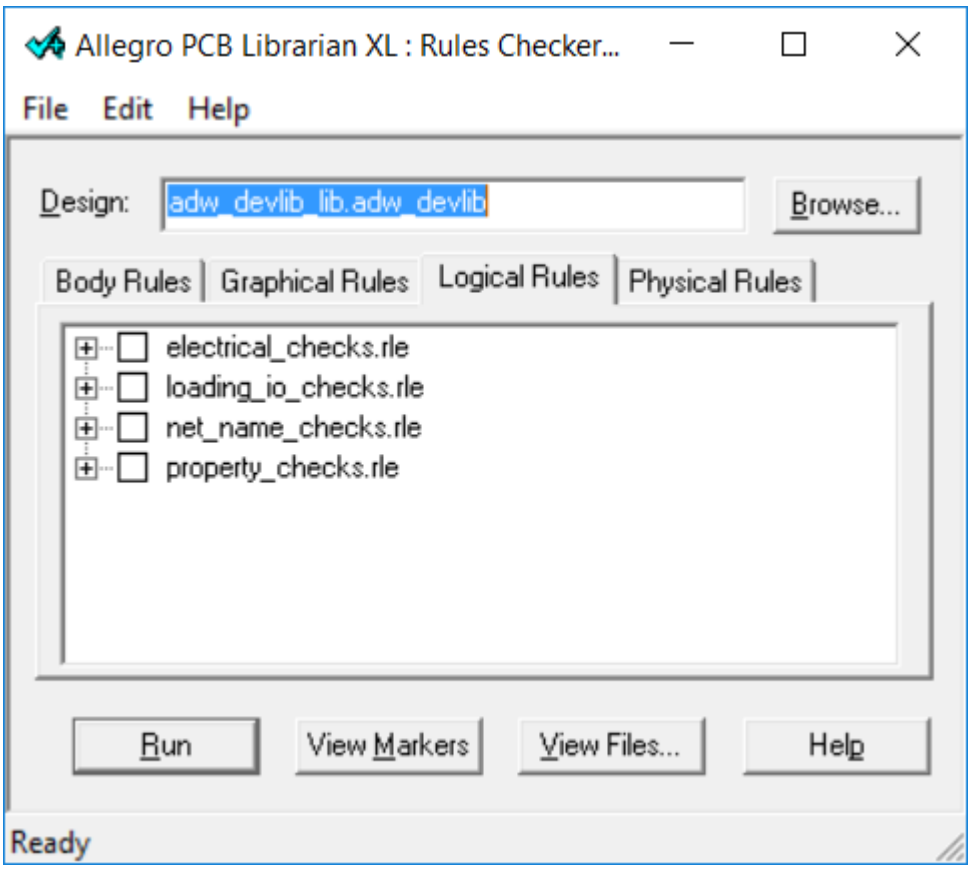


5. Type a name for the button in *Title*. For example, `HDL Rule Checker/Compiler`.
6. In *Image Name*, click the browse button and select an image file for the icon.

7. In the Exec section, specify the following command string: `checkplusui`.
8. Select the *Use Project File* check box to pass the `-proj <your_project>.cpm` string to the command.
9. Specify the following command line argument: `-product <license name>`.

   For example: `-product PCB_Librarian_Expert`
10. Click the newly created *HDL Rule Checker/Compiler* button.

    The Allegro PCB Librarian XL: Rules Checker dialog opens.

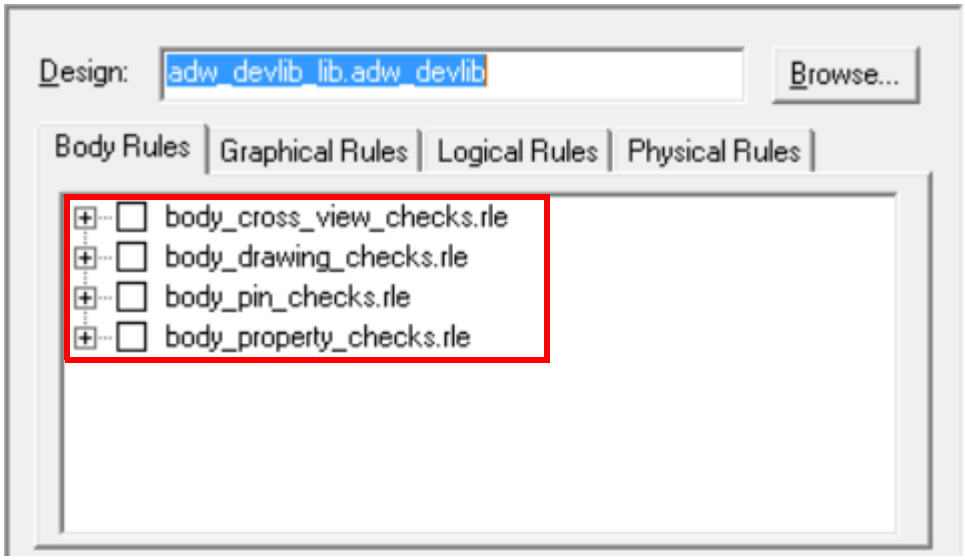


    The example in this document modifies the `body_cross_view_checks` rule set by deleting the `physical_to_body_check` rule from the rule set.
11. Select the *Body Rules* tab.

    Four rule sets are available:

    - `body_cross_view_checks.rle`

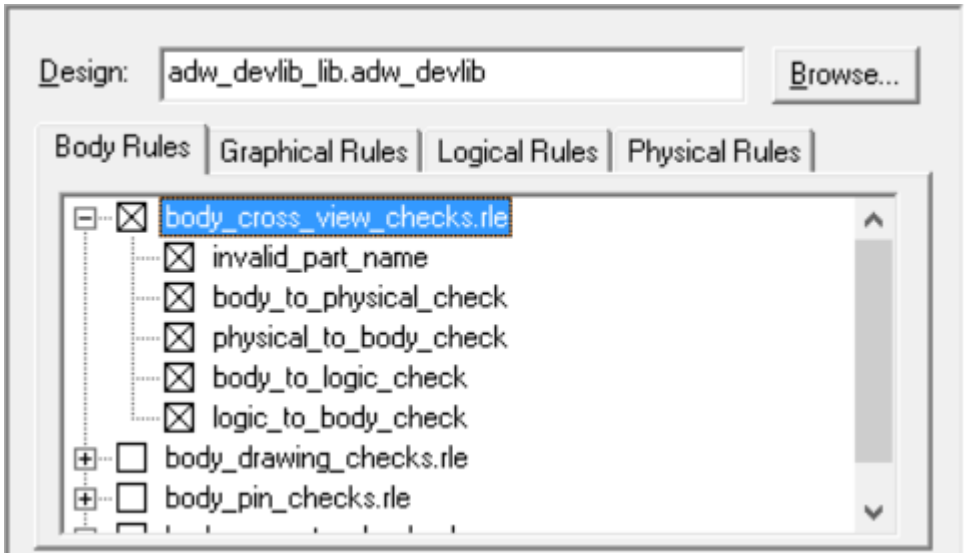
- `body_drawing_checks.rle`
- `body_pin_checks.rle`
- `body_property_checks.rle`



The `body_cross_view_checks.rle` rule set has are five rules:

- `Invalid_part_name`
- `Body_to_physical_check`
- `Physical_to_body_check`
- `Body_to_logic_check`

- `Logic_to_body_check`

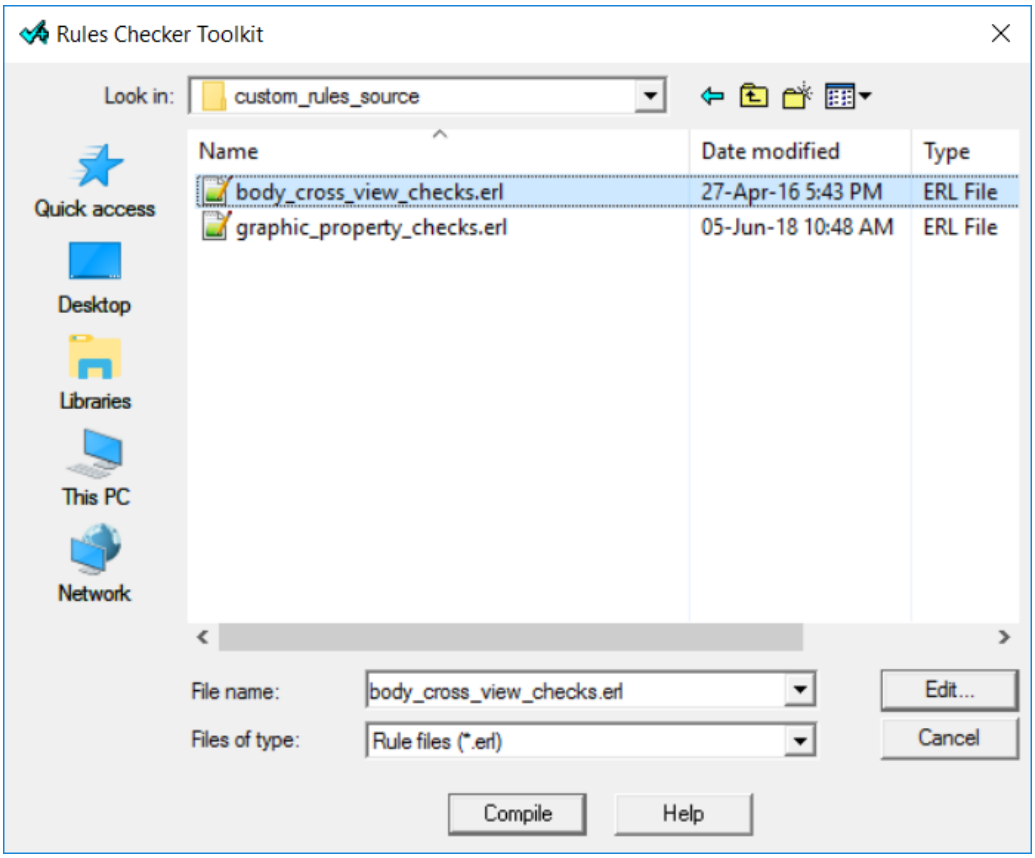


12. Deselect the `Physical_to_body_check` check box.

13. Select *Edit — Rules*.

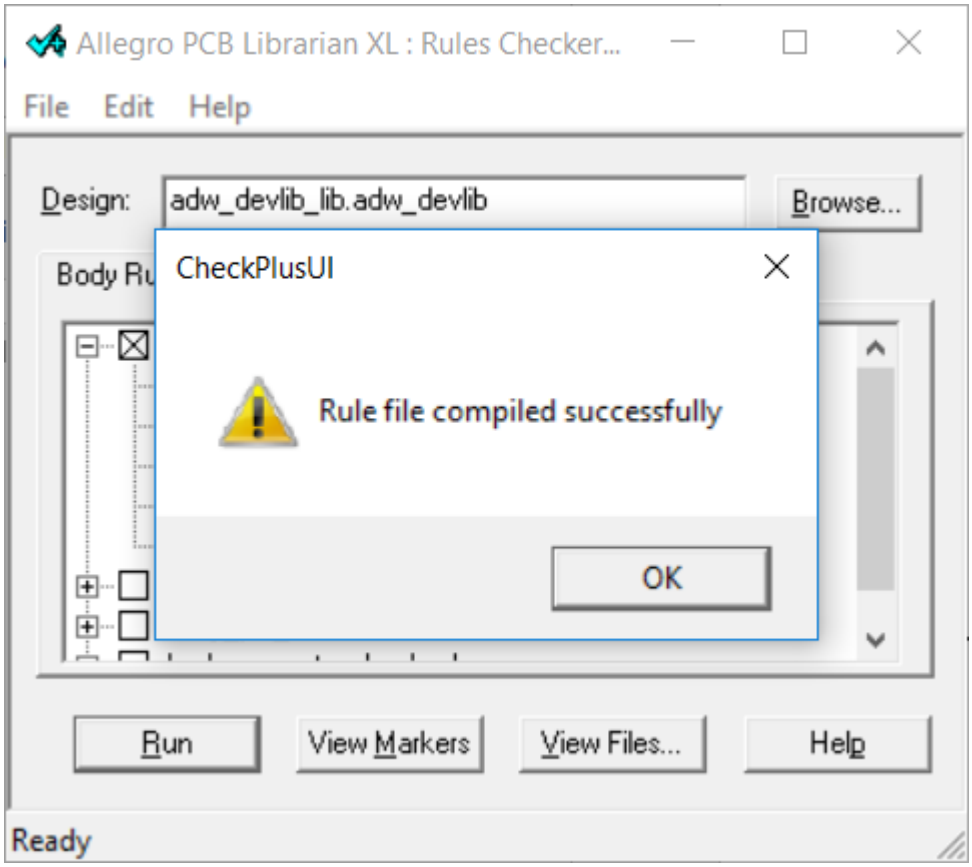
14. Browse to the `body_cross_view_checks.erl` source file in:

*<Allegro EDM Conf Root>**<company>**<site>*\checkplus\custom_rules_source.



15. Click *Compile*.

A message that the rule is successfully compiled is displayed.



16. Move the new compiled rule, `body_cross_view_checks.rle`, from:

    *<Allegro EDM Conf Root>**<company>**<site>*\checkplus\custom_rules_source

    to

    *<Allegro EDM Conf Root>**<company>**<site>*\checkplus\custom_rules

17. Exit Rules and launch it again.
18. Launch Allegro EDM Flow Manager.
19. Run Rules Checker.

    The new rule is loaded and executed.

Creating a Rules Checker Rule

To create a Rules Checker rule from scratch rather than modify an existing default rule and save it, do the following:

1. Open the `site.cpm` file from:

   *<Allegro EDM Conf Root>**<company>**<site>*\cdssetup\projmgr.

2. In the `START_CHECKPLUS` section of `site.cpm`, replace `$CDS_SITE` with *$ADW_CONF_ROOT/<company>/<site>* in:

   - `RULE_SEARCH_PATH`
   - `INCLUDEPATH`

     This step is required to get the custom rules into Rules Checker running in the user interface mode. Here is the relevant section from the `site.cpm` file:

     ```
     START_CHECKPLUS

     rundir 'checkplus'

     VIEW_LOGICAL 'cfg_package'

     RULE_SEARCH_PATH '$ADW_CONF_ROOT/<company>/<site>/
     checkplus/rules/' '$ADW_CONF_ROOT/<company>/<site>/
     checkplus/custom_rules/rules' '$CDS_INST_DIR/tools/
     checkplus_exp/concept/rules/' '$CDS_INST_DIR/tools/
     checkplus_exp/concept/custom_rules/'

     INCLUDEPATH '$ADW_CONF_ROOT/<company>/<site>/checkplus/
     rules_include/' '$ADW_CONF_ROOT/<company>/<site>/checkplus/
     custom_rules_include/' '$CDS_INST_DIR/tools/checkplus_exp/
     concept' '$CDS_INST_DIR/tools/checkplus_exp/concept/
     templates'

     END_CHECKPLUS
     ```

3. Create an `.erl` file at:

   *<Allegro EDM Conf Root>**<company>**<site>*\checkplus\custom_rules_source.

4. Create the required `.h` file at:

   *<Allegro EDM Conf Root>**<company>**<site>*\checkplus\custom_rules_include.

5. Compile the `.erl` file to `.rle` file and copy it to:

*<Allegro EDM Conf Root>**<company>**<site>*\checkplus\custom_rules\rules.

6. Launch Allegro EDM Configuration Manager.
7. Click *Set up or Manage Company & Site*.
8. Do one of the following:
   - If you created a rule for a library flow, navigate to:

     Allegro EDM Conf Root – <company> – <site> – Tool Configuration – Library Workbench – Schematic Model Verification – library_verification.ini.
   - If you created a rule for a design flow, navigate to:

     Allegro EDM Conf Root –<company> – <site> – Tool Configuration – Logical Design – Schematic Design Verification – verif.ini.
9. Click the *Save* button on the right panel.
10. Open *<Allegro EDM Conf Root>**<company>**<site>*\library_verification\15.5\library_verification.ini and make the following changes:
    a. Add the path to the rule you just created:

    ```
    [ckp_rules_location_prefix]

    myrules = $ADW_CONF_ROOT/<company>/<site>/checkplus/
    custom_rules
    ```

    b. Add the name of the rule you just created:

    ```
    [ckprle_precompiled_body]

    myrules.my_custom_rule = on
    ```
11. Launch Allegro EDM Flow Manager.

    A new rule is loaded and executed.

Related Topics

- For details on how to write Rules Checker rule files, see *Allegro Design Entry HDL Rules Checker User Guide*.
- Control of PCB Editor Environment in Allegro EDM
- Customizing Rules for Library and Design Flows

- Modifying Parameters in Library and Design Flow Rules
- Launching Allegro EDM Configuration Manager

Customizing Error Messages in Allegro EDM

Allegro EDM displays messages with error codes. You can override these messages or add your own details to messages, if needed, using an entry in the site-level `messages.properties` file. This entry contains an override flag, which can be different for each error message depending on your requirement.

➡ To customize error messages, copy the `messages.properties` file from:

<Cadence installation directory>\adw_conf_root*<company_name>**<site_name>*\messages

to

<Allegro EDM Conf Root>\<company>\<site>\messages

This is a standard Java property file and follows the property file syntax. Customize this file to add messages and configure their display. For example:

```
#message for client-server version mismatch

GENERIC-00261.override.install=true

GENERIC-00261.message=\n\nMessage from Site Administrator: The
Allegro EDM Hot fix installer can be downloaded from
\\\\server_name\\shared_folder.
```

Contents and Syntax of Match Files

When you create a new model type in Allegro EDM Database Administrator, Allegro EDM creates a match file. The match file dictates what content is included when a library model is checked into Allegro Library Manager.

The match file also specifies the structure and behavior of archives of a model type during their archival and extraction in various library flow processes.

This section describes the contents, syntax, and usage of the match file.

Contents of Match Files

Section name	Description
UPREV	This section is used to identify files in the archive that must be upreved when archives are being upreved from one version to another. This section is only used for backend models. The `install_model` utility reads this section when installing models in `reflib` while migrating to a new release such as from 22.1 to 23.1.
CHECKSUM	This section is used to identify files that are used to generate checksum that is added in the `.status` file. This section is used by Allegro EDM Database Editor and Library Import while creating archives for new models.
ARCHIVE	This section is used to identify files to be archived when creating archives. During the archival process, if any of the specified files is not available in the source area, no error is generated. This section is used by Allegro EDM Database Editor and Library Import while creating archives for new models.
INSTALL	This section is used to identify files to be extracted from an archive when you create `reflib` or when you extract archives in `flatlib` for editing models. This section is read by `install_model` during library distribution for creating `reflib`. It is also read by Allegro EDM Database Editor while revising models.

Section name	Description
ARCHIVE_TIMESTAMP_SOURCE	This section is used to identify files that need to be checked to identify whether a change is Major or Minor. The specified files are checked against the checkout time of the model. This section is used by Allegro EDM Database Editor while checking in a model to identify whether the change is Major or Minor.
ARCHIVE_TIMESTAMP_DERIVED	This section is used to specify any derived files such as verification log files. These specified files are then verified while creating archives. The verification process checks that the timestamp of the derived files is greater than that of the modified model files. This check has two advantages. It ensures that the: ■ Verification checks have been run after the modification of the models ■ Derived files which form a part of the archive have been modified after the modification of the model files This section is used by Database Editor while checking in a model.
ARCHIVE_MUST_EXIST_FILES	This section is used to specify files that must exist in `flatlib` (where the model is edited) when creating the archive. This section only checks that before creating any archive, the specified files exist. This does not mean that all specified files are necessarily part of the archive. To add these files to the archive, they must be specified in the `ARCHIVE` section. This section is used by Database Editor while creating an archive for a model being checked in.

Section name	Description
ARCHIVE_CLEANUP_FILES	This section is used to specify files to be deleted from the work area after the archive is created. This is helpful to clean up any temporary files from the work area to ensure that they do not cause any problem for other models being used in the same work area. This section is used by Database Editor after checking in a model to clear the work area.

Syntax and Usage of Match Files

The following figure displays the sections of `match_files.allegro.flash`.

Editable File

D:\170_test\adw170\adw_conf_root\CAD170\NOIDA170\db_install\15.7\\match_files.allegro.flash

```
# ATDM_VERSION : 1.0
# 1.0 WP 2001/07 Initialisation uprev
# 1.1 WP 2002/11 remove use of pad_conv, this parameter file is for uprev_allegro
# 1.2 WP 2003/06 comment in english and check

#$comp_name environment variable set in the java calling program
#$clasx    environment variable set in the java calling program

BEGIN UPREV
 F ( $comp_name.dra )
 F ( $comp_name.fsm )
END UPREV

BEGIN CHECKSUM
 F ( $comp_name.dra )
 F ( $comp_name.fsm )
END CHECKSUM

BEGIN ARCHIVE
 F ( $comp_name.dra )
 F ( $comp_name.fsm )
 F ( $comp_name.log.flash )
 F ( $comp_name.status )
END ARCHIVE

BEGIN INSTALL
 F ( $comp_name.dra )
 F ( $comp_name.fsm )
 F ( $comp_name.log.flash )
 F ( $comp_name.status )
END INSTALL

BEGIN ARCHIVE_TIMESTAMP_SOURCE
 F ( $comp_name.fsm )
END

BEGIN ARCHIVE_TIMESTAMP_DERIVED
END

# Wild-card characters are not supported for this section
BEGIN ARCHIVE_MUST_EXIST_FILES
 F ( $comp_name.fsm )
END

BEGIN ARCHIVE_CLEANUP_FILES
 F ( {$comp_name}_symchk.log )
 F ( {$comp_name}_symchk.log,* )
 F ( {$comp_name}_symchk.mkr )
END
```

Each section contains:

- Section name
- Blank space followed by the component names followed by the extension name of the model type. You can use the actual extension names or wild cards to specify model types:

- `F ( $comp_name.pad )`
- `F ( $comp_name.* )`
- `F ( $comp_name/*.* )`

Important

Ensure that there is a blank space after and before the starting and closing brackets.

For example, `match_files.allegro.flash` contains the following sections:

Section	Description and Use

`BEGIN UPREV`
`F ( $comp_name.dra )`
`F ( $comp_name.fsm )`
`END UPREV`
	The following files of the model type `flash` are upreved:
	■ `<model_name>.dra`
	■ `<model_name>.fsm`

`BEGIN CHECKSUM`
`F ( $comp_name.dra )`
`F ( $comp_name.fsm )`
`END CHECKSUM`
	The following files of the model type `flash` are used to generate checksum:
	■ `<model_name>.dra`
	■ `<model_name>.fsm`

`BEGIN ARCHIVE`
`F ( $comp_name.dra )`
`F ( $comp_name.fsm )`
`F ( $comp_name.log.flash )`
`F ( $comp_name.status )`
`END ARCHIVE`
	The following files of the model type `flash` are archived when creating archives:
	■ `<model_name>.dra`
	■ `<model_name>.fsm`
	■ `<model_name>.log.flash`
	■ `<model_name>.status`

Section	Description and Use
BEGIN INSTALL F ($comp_name.dra) F ($comp_name.fsm) F ($comp_name.log.flash) F ($comp_name.status) END INSTALL	The following files of the model type `flash` are extracted from an archive: ■ <model_name>.dra ■ <model_name>.fsm ■ <model_name>.log.flash ■ <model_name>.status
BEGIN ARCHIVE_TIMESTAMP_SOURCE F ($comp_name.fsm) END	The following files of the model type `flash` are checked for modification time while checking in the model to identify if the change is Major or Minor: ■ <model_name>.fsm
BEGIN ARCHIVE_MUST_EXIST_FILES F ($comp_name.fsm) END	The following files of the model type `flash` must exist in `flatlib` when creating the archive: ■ <model_name>.fsm
BEGIN ARCHIVE_CLEANUP_FILES F ({$comp_name}_symchk.log) F ({$comp_name}_symchk.log,*) F ({$comp_name}_symchk.mkr) END	The following files of the model type `flash` must be deleted from work area after the archive is created: ■ <model_name>_symchk.log ■ <model_name>_symchk.log,* ■ <model_name>_symchk.mkr

Use of Operators in Match Files

The following example shows how operators are used in match files:

```
sch_dir = F ( $comp_name/sch_* )
all_file = ( $sch_dir || /* )
tmp_file = ( ( $sch_dir || /*,[0-9] ) | ( $sch_dir || /*.lck )  | ( $sch_dir || /
     *.old ) )
sch_file = ( $all_file - $tmp_file )
( ( $sch_file ) IF $sch_file )
```

The description of the various operators that are used in the match files are:

Operator Symbol	Operator	Description and Example
`(`	OPEN_PAR	Example: `( $sch_dir || /* )`
`)`	CLOSE_PAR	Example: `( $sch_dir || /* )`
`-E`	EXIST	Example: `-E ( $comp_name/metadata/revision.dat )` Checks for the existence of a file and returns name of file if it exists.
`F`	FILTER	Example: `sch_dir = F ( $comp_name/sch_* )` Filters the files with the format `sch_*` for the object *<component_name>*.
`&`	AND	Example: `sch_file = ( $all_file & $tmp_file )` Lists the files that are present in both `$all_file` and `$tmp_file`.
`|`	OR	Example: `tmp_file = ( ( $sch_dir || /*,[0-9] ) | ( $sch_dir || /*.lck ) | ( $sch_dir || /*.old ) )` `tmp_file` lists the files in *<sch_dir>* which end with numerals 0 to 9, with extension `.lck`, or with extension `.old`.
`^`	XOR	Example: `sch_file = ( $all_file ^ $tmp_file )` Returns the list of files, which are present in `$all_files` but not present in `$tmp_file` and vice versa.

Operator Symbol	Operator	Description and Example
-	MINUS	Example: `sch_file = ( $all_file - $tmp_file )` Returns the list of files, which are present in `$all_file` but not present in `$tmp_file`.
=	SET	Example: `all_file = ( $sch_dir || /* )` Sets the value returned from the expression to the `all_file` variable.
IF	IF	Example: `( ( $sch_file ) IF $sch_file )` Checks for empty value.
IN	IN	Example: `sch_file = ( "module ddr3" IN $all_file )` Returns the list of files from `$all_files` that contain the string `module ddr3`.
\|\| | CONCAT | Example:
`all_file = ( $sch_dir \|\| /* )`
`all_file` lists the files in *<sch_directory>*. |

Customizing Match Files for Archives of Model Types

You can customize the default Allegro EDM match file based on your needs for Design Entry HDL and Allegro X System Capture.

To edit an out-of-the-box match file, do the following:

1. Launch Allegro EDM Configuration Manager.
2. Click *Set up or Manage Company & Site*.

   The Set up or Manage Company & Site tab displays the `workbench.ini` file.

3. On the left tree panel, choose *Allegro EDM Conf Root – <company> – <site> – Tool Configuration – Library Workbench – Match Files*.
4. Choose the required match file.

   The right panel displays the *Reference File* and the *Editable File*.
5. Review or edit the match file configuration.
6. Click *Save*.

Related Topics

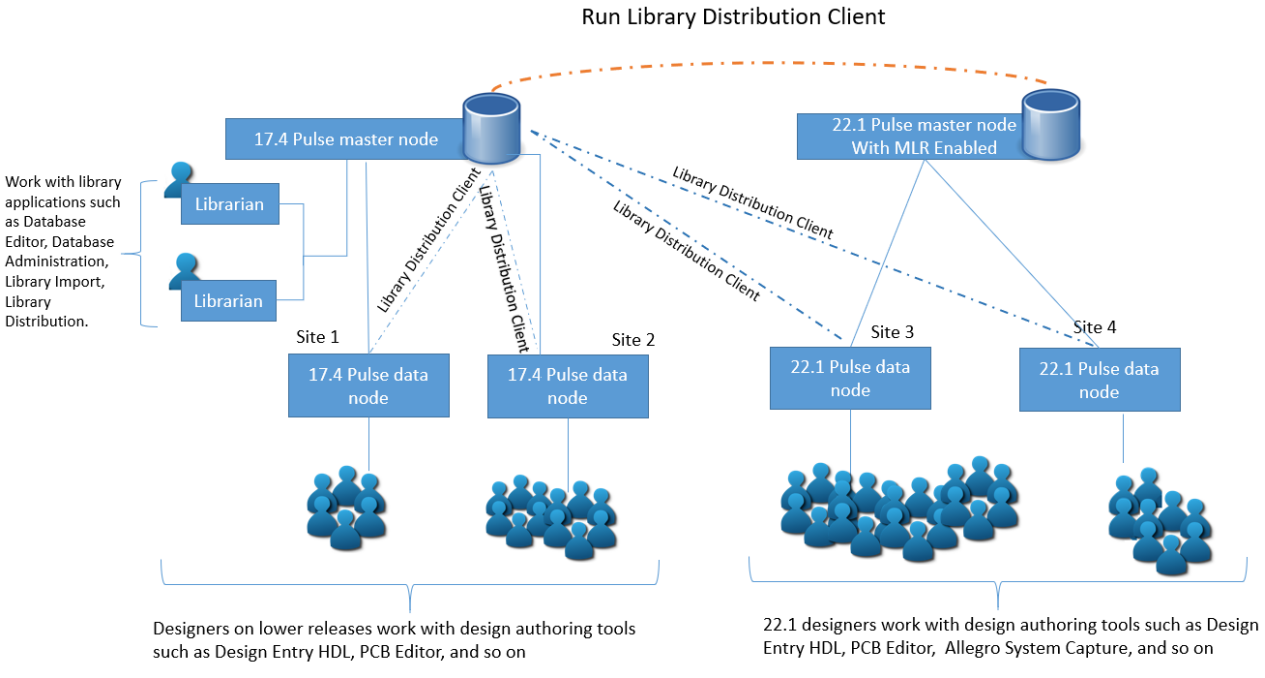
- Allegro Library Manager

13

Multi-Release Support with Multi-Library Release Configuration

If you are an existing customer on release 22.1, it might not always be possible or practical to move all the designers across all your organization's divisions and sites to the Pulse environment in release 23.1. As a workaround, you can use the Multiple Library Revision (MLR) functionality, which helps you move site by site to the new release.

The primary challenge when moving to a new release are the part libraries that are common for all the divisions or sites. After you migrate to a new release, libraries might not work with older releases. As a result, it is recommended that you continue to author, develop, and manage libraries in the lowest release—17.4—until all the design sites move to the latest release, which is 23.1.



***After all the Pulse data nodes have been moved to 22.1 data nodes, library authoring can be moved to 22.1 after running *adw_uprev*

A Pulse primary node uses a library server license. If you do not toggle on the MLR option during the Pulse primary node setup, the 23.1 Pulse primary node starts in maintenance mode because the library server license is already being used by the 22.1 Pulse primary node. Enabling MLR and restarting the 23.1 primary node with MLR enabled ensures that the 23.1 Pulse primary node runs and does not go into maintenance mode.

Note: Unless a remote URL is set, a node functions as the Pulse primary node.

Related Topic

Defining Library Management Settings for Pulse Primary Node

Configuring Multi-Library Release Environment

Before you start configuring the MLR environment, be aware that the Pulse-Allegro X System Capture integration feature of part requests is not supported in Pulse running in an MLR environment or in a site-specific configuration.

It is recommended that you do not use these features in MLR or site-specific configurations to avoid possible data issues.

It is also recommended that Allegro X System Capture users managing design data and using Pulse version control should point to the 23.1 Pulse primary node in a 22.1-23.1 MLR environment.

Whenever you migrate an existing MLR environment between a Pulse primary node and a Pulse data node from an older release to a newer one, do the following:

1. Set up the MLR-enabled Pulse primary node.

   This ensures that the MLR-enabled Pulse primary node does not use a library license when you run the 23.1 Pulse primary node. The library development and management related utilities of Database Editor, Database Administration, and so on are not allowed to run in the 22.1 Pulse primary node; they only run in the 23.1 Pulse primary node.

   In the MLR environment, data from the Pulse primary node of a lower release is distributed to clients in a higher release using the `lib_dist` utility.

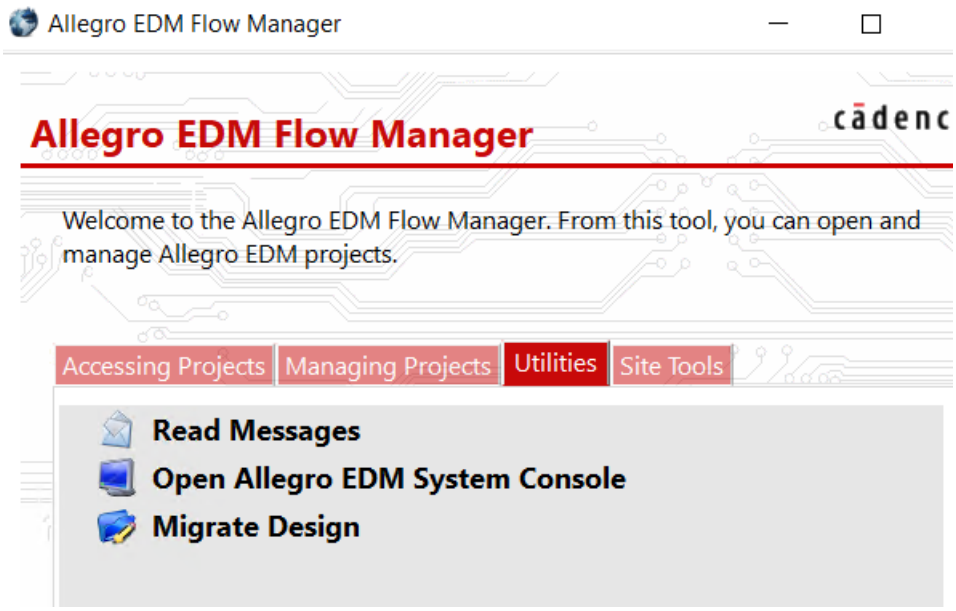
2. To automatically distribute parts from the Pulse primary node from a lower release to a higher release, do the following:

   a. Open an elevated Command Prompt window.

   b. Navigate to the Allegro EDM Conf Root directory and run the Allegro EDM `<startworkbench>.bat` file.

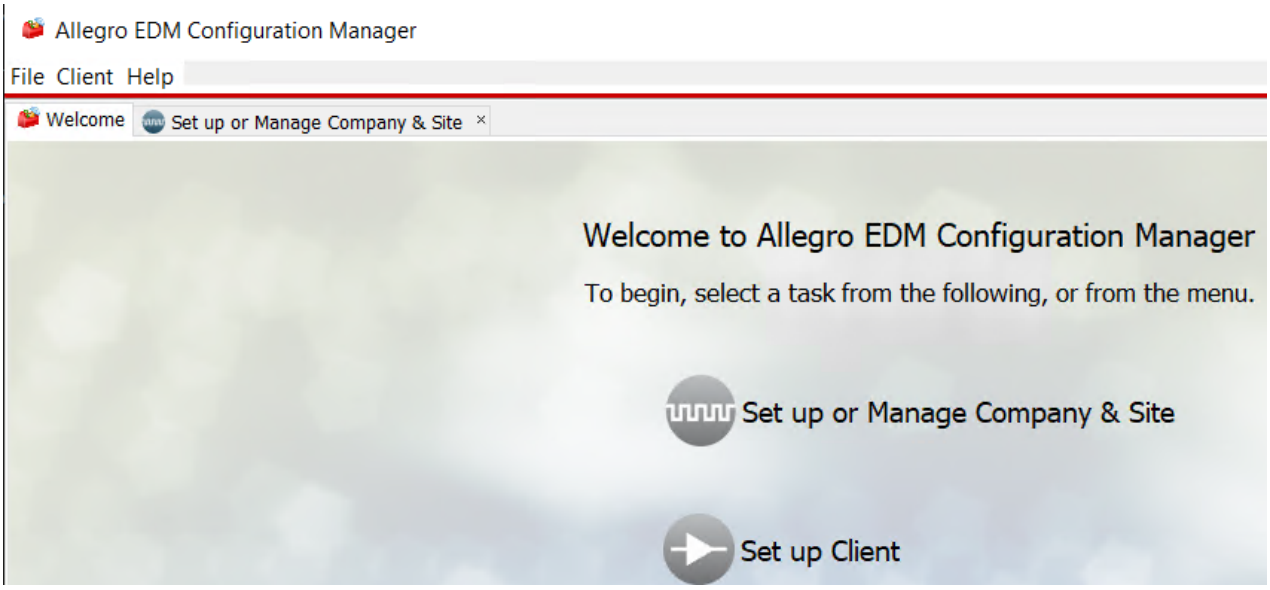
   Allegro EDM Flow Manager is displayed.

   c. Click *Utilities — Open Allegro EDM System Console*.



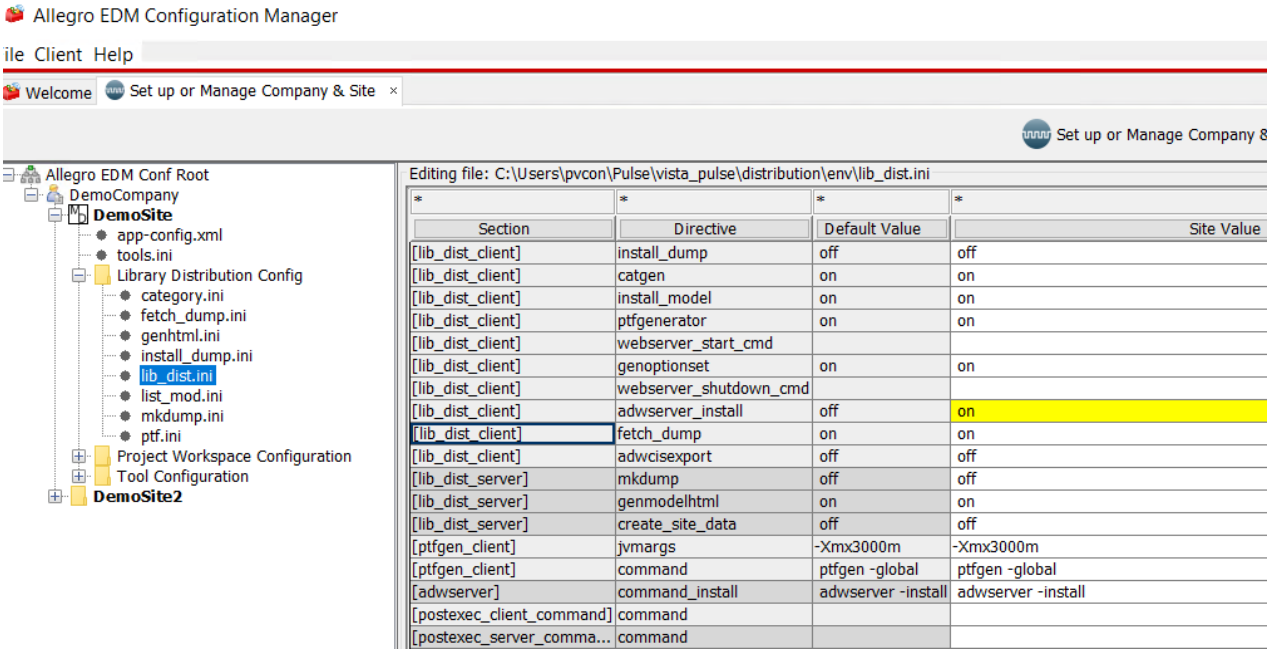
   d. In the console window, type `conf`.

Allegro EDM Configuration Manager is displayed.



e. Click *Set up or Manager Company & Site*.

f. Set `adwserver_install=on` in the `lib_dist.ini` file.



Section	Directive	Default Value	Site Value
[lib_dist_client]	install_dump	off	off
[lib_dist_client]	catgen	on	on
[lib_dist_client]	install_model	on	on
[lib_dist_client]	ptfgenerator	on	on
[lib_dist_client]	webserver_start_cmd		
[lib_dist_client]	genoptionset	on	on
[lib_dist_client]	webserver_shutdown_cmd		
[lib_dist_client]	adwserver_install	off	on
[lib_dist_client]	fetch_dump	on	on
[lib_dist_client]	adwcisexport	off	off
[lib_dist_server]	mkdump	off	off
[lib_dist_server]	genmodelhtml	on	on
[lib_dist_server]	create_site_data	off	off
[ptfgen_client]	jvmargs	-Xmx3000m	-Xmx3000m
[ptfgen_client]	command	ptfgen -global	ptfgen -global
[adwserver]	command_install	adwserver -install	adwserver -install
[postexec_client_command]	command		
[postexec_server_comma...	command		

Running `lib_dist_client` with this setting ensures that:

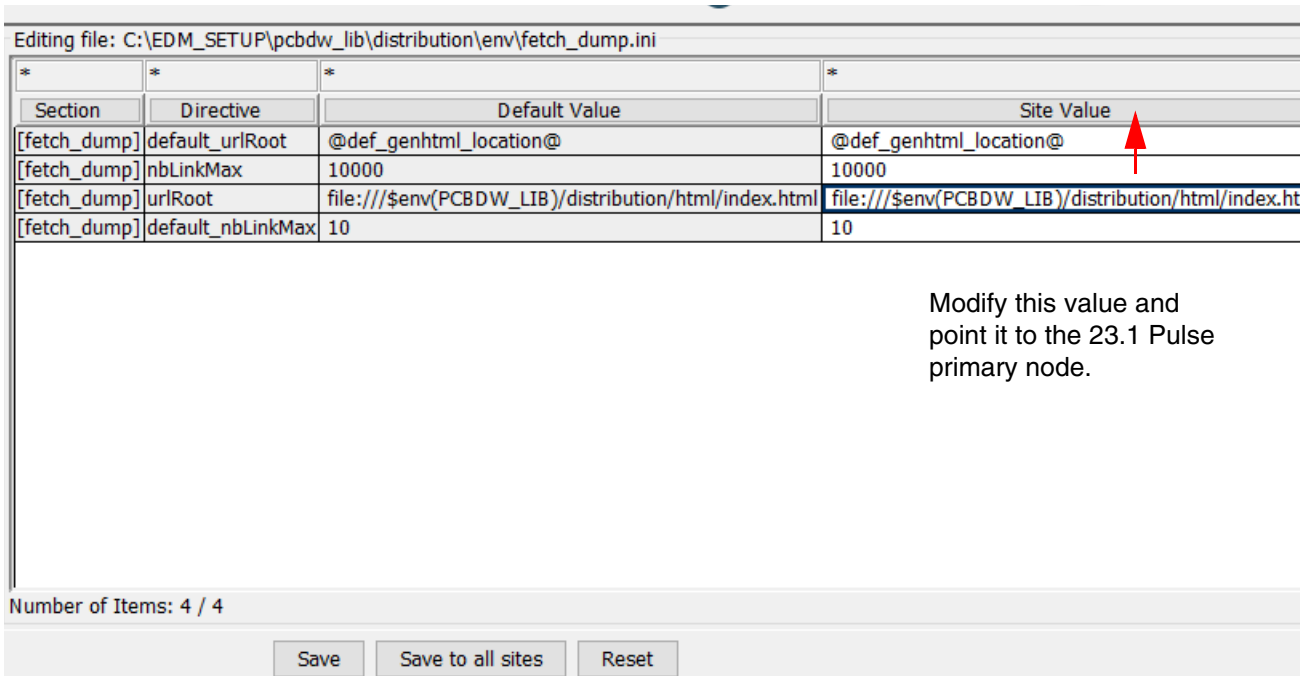
- The Pulse primary node from a lower release is upreved before being installed in the higher release.
- Models from the lower release are upreved while populating the `reflib` folder of the higher release.

g. Ensure one of the following:

- That the `MLR_PCBDW_LIB` variable in the *<PCBDW_LIB>*`/distribution/env/fetch_dump.ini` file is set to the *<PCBDW_LIB>* of the Pulse primary node. This must be a mapped drive.

  This is required because the Pulse data server reads the `MLR_PCBDW_LIB` variable, which points to the Pulse primary node, after which the library distribution process creates the library in the Pulse data node.

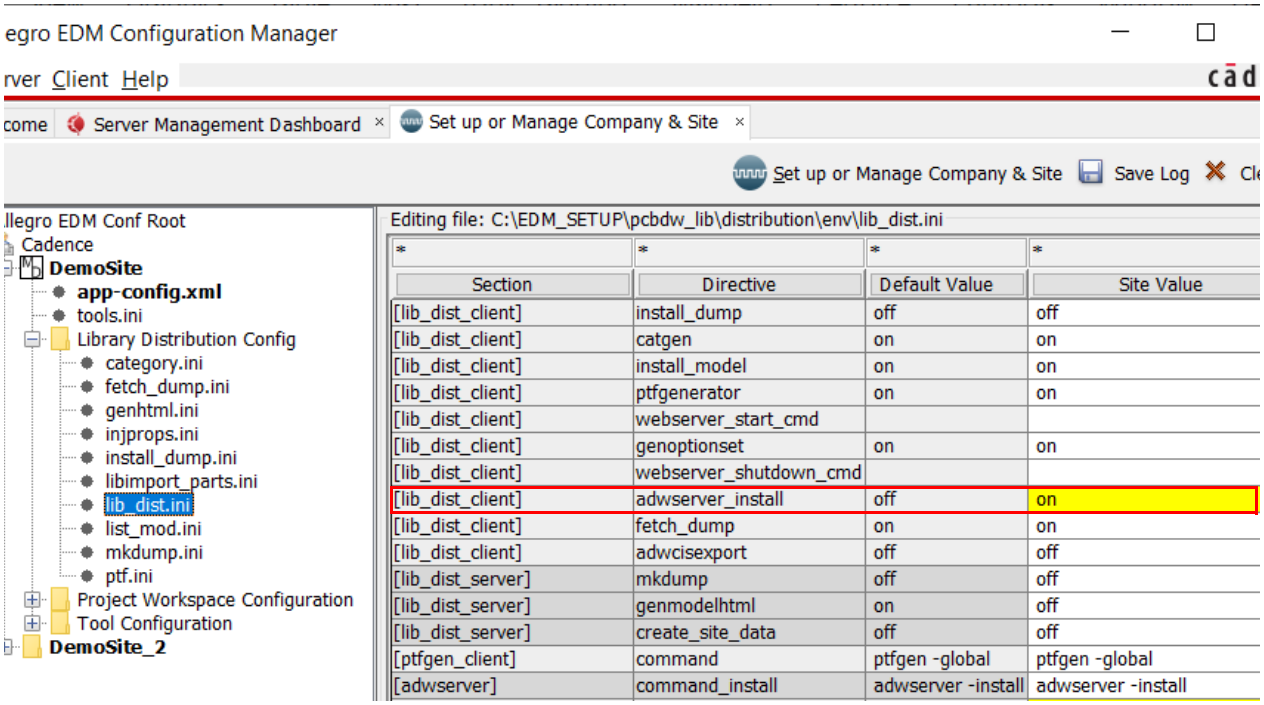
- If the Pulse data node points to the 23.1 Pulse primary node, ensure that `fetch_dump.ini` points to the same 23.1 Pulse primary node. This is needed so models are fetched from the correct Pulse primary node.



a. To update `fetch_dump.ini`, run Allegro EDM Configuration Manager.

b. Update the `[adwserver]` section of the `lib_dist.ini` file.

c. Modify the *Site Value* to `on`.



Section	Directive	Default Value	Site Value
[lib_dist_client]	install_dump	off	off
[lib_dist_client]	catgen	on	on
[lib_dist_client]	install_model	on	on
[lib_dist_client]	ptfgenerator	on	on
[lib_dist_client]	webserver_start_cmd		
[lib_dist_client]	genoptionset	on	on
[lib_dist_client]	webserver_shutdown_cmd		
[lib_dist_client]	adwserver_install	off	on
[lib_dist_client]	fetch_dump	on	on
[lib_dist_client]	adwcisexport	off	off
[lib_dist_server]	mkdump	off	off
[lib_dist_server]	genmodelhtml	on	off
[lib_dist_server]	create_site_data	off	off
[ptfgen_client]	command	ptfgen -global	ptfgen -global
[adwserver]	command_install	adwserver -install	adwserver -install

3. Run `lib_dist_client` at the 23.1 Pulse data node machines to update the parts library.

Running `lib_dist_client` for the first time after a Pulse data node upgrade can take some time as the complete library data is installed.

Related Topic

Migration Guide for Allegro X Platform Products

A

Appendix

This section provides information on miscellaneous topics.

- Modifying Pulse Primary Node Settings After Initial Configuration and Restarting Server
  - Modifying SSL Settings for Pulse Primary Node
  - Manually Modifying Files in the Pulse Primary Node Home Directory
- Configuring a Node as a Pulse Data Node
- Switching from Managed to Unmanaged Libraries and Vice Versa in Pulse Primary Node
- Creating and Using SSL Certificates in Pulse Environment
- Mapping of Pulse Internal Service Names and UI Terms
- Troubleshooting Pulse Server Management
- Allegro EDM Configuration Manager User Interface

Modifying Pulse Primary Node Settings After Initial Configuration and Restarting Server

There are some maintenance tasks that you might need to do on a regular basis after you complete the initial configuration, such as registering users, backing up server data, and so. The tasks are as follows:

- Modifying SSL Settings for Pulse Primary Node
- Manually Modifying Files in the Pulse Primary Node Home Directory

After some of these operations, or when you need to install an update, you might need to shut down and restart the server.

Modifying SSL Settings for Pulse Primary Node

If you did not select the encrypted communication option between the Pulse primary and data nodes or client machines during the initial configuration of the servers, you can modify the SSL setting later.

To modify the SSL setting, do the following:

1. Shut down all the Pulse data nodes.
2. In the Pulse Service Manager settings page, modify the SSL setting.
3. Restart the Pulse primary node.

   Because the Pulse primary node changes from HTTP to HTTPS, all the data nodes will be in an error state because they cannot establish a connection with the Pulse primary node.
4. Modify the *Remote URL* field in the data nodes.
5. Restart the data nodes one by one to connect them to the Pulse primary node.

Manually Modifying Files in the Pulse Primary Node Home Directory

It is neither recommended nor will you typically need to modify files in the Pulse primary node home directory, except in certain cases, which are as follows:

- When you configure library distribution for a Pulse data node, you might need to edit the `<Pulse primary node home>/distribution/lib_dist.ini` and `fetch_dump.ini` files.
- If you enable SSL/TLS communication, you create the `<Pulse primary node home>/server/conf/aurora/security` folder and place the Java KeyStore (`edm.jks`) in the `security` folder.
- If you need to migrate the Publish for Manufacturing configuration, you manually move the configuration to `<Pulse primary node home>/server/data/Polaris`.
- On **Linux,** if you install and run Pulse as a service on Linux, the installation script creates a Linux user and a group, both of which are called `pulse`.

  When you modify files in the Pulse primary node home directory, you must obtain the required authorization to write to the folders owned by the `pulse` user. Or, you can become the `pulse` user and make changes to files and folders in the Pulse primary node home directory by using the following command on a terminal:

  ```
  sudo su -l pulse -s /bin/bash
  ```

Related Topics

- Specifying Security Settings for Pulse Primary Node
- Creating and Using SSL Certificates in Pulse Environment
- Accessing Pulse Service Manager Web Page
- Creating and Using SSL Certificates in Pulse Environment
- Enabling Part Request Process With Review
- Installing Pulse Primary Node as a Service
- Allegro EDM Library Distribution User Guide

Configuring a Node as a Pulse Data Node

If you decide on a cluster of a Pulse primary node and multiple data nodes, you must specify the URL of the Pulse primary node, called the remote URL, in the machines that work as data nodes.

After you specify the remote URL in the data nodes, library and design data is fetched from the Pulse primary node to the data nodes and is then accessed by the client machines.

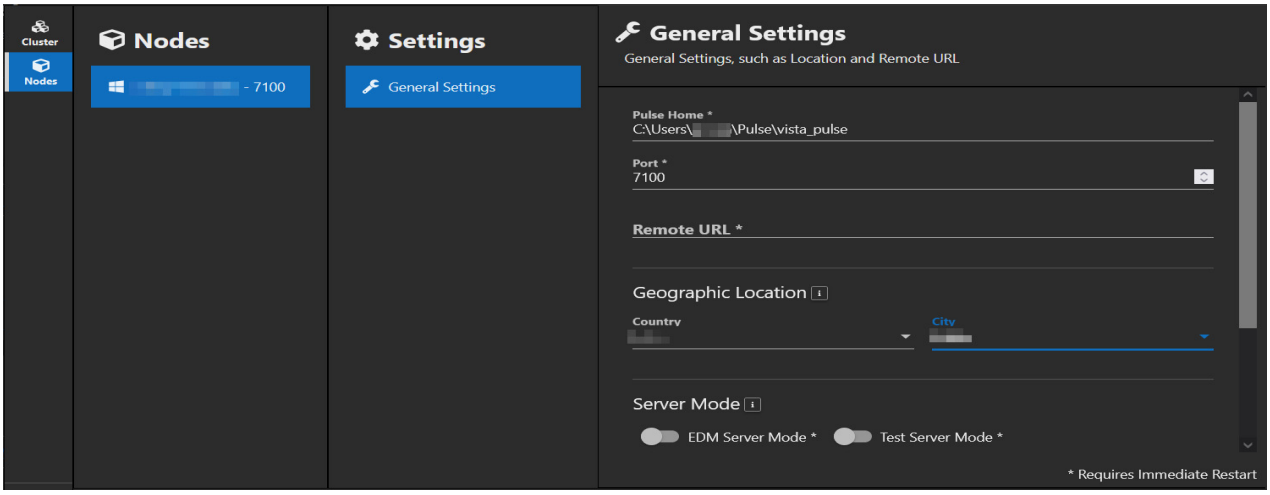
Important

If the library server license, `PCB_Library_Server_XL`, is not available, a node starts up in maintenance mode. In this mode, Pulse services are not available. You can still set the remote URL to configure the node as a data node. On restarting Pulse, the data node does not check out a license. Only the Pulse primary node checks out a license for the Pulse cluster.

To configure a node as a Pulse data node, do the following:

1. Start Pulse Service Manager to access its web page with the configuration options.

2. Select *Nodes* in the left pane.



3. You can do the following:

- ❑ Modify the Pulse Home, which is where Pulse stores data.

  Pulse resets data if the new location is empty or attempts to read the data if an existing Pulse home structure is selected. A possible result is that the administrator password is changed or reset to the default, which is `admin`. You must modify the default again and set it to your required password.

  When a node is configured as a data node, it inherits the primary node's administrator credentials.

- ❑ Modify the server port, if required.

  This is usually needed when certain ports cannot be used because of firewall restrictions in your organization. If you modify the port here, ensure that the two preceding ports are open on every Pulse data node machine.

  The two preceding, additional open ports are needed for Pulse functions.

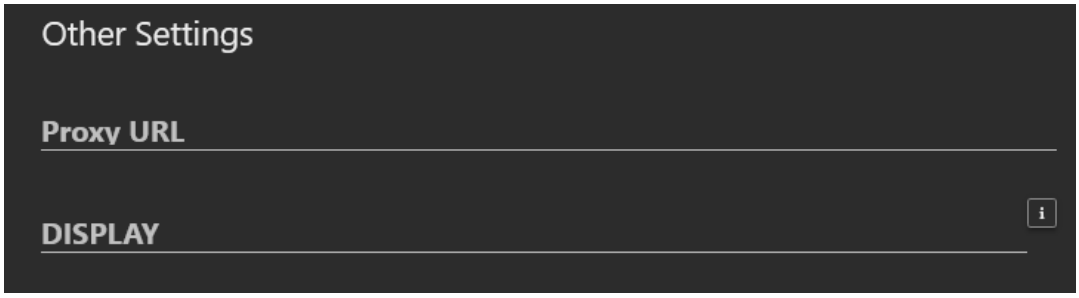
- ❑ Specify the URL of the Pulse primary node.

  - ❍ Multiple Pulse data nodes can be connected to the same Pulse primary node.
  - ❍ A Pulse data node cannot be connected to two different Pulse primary nodes.

  Note: You do not define a remote URL for the Pulse primary node.

- ❑ For easier identification of a machine, you can specify the geographical location of the machine. Pulse Manager displays the location of various machines on a map to help you track and locate them.

- In *Server Mode*, define how you want to run the server. See Managing Test Server for details.
- *Proxy URL* is a legacy option that might be required in older HotFixes of Allegro X System Capture.
- On Linux systems, specify the value of the `DISPLAY` environmental variable.



This is used by Pulse to determine what to use as a DISPLAY because a DISPLAY is required for the server.

The `DISPLAY` variable is mandatory for each node of the Pulse server cluster and is specified as *`<hostname>:<displaynumber>.<screennumber>`* where:

- *`<hostname>`* - Specifies the name of the machine to which the display is physically connected.
- *`<displaynumber>`* and *`<screennumber>`* – For displays with multiple monitors that share a common keyboard and pointer, specify the display number and screen number. If the screen number is not given, screen 0 is used.

Notes About Converting Pulse Primary Node to Pulse Data Node

If you convert a Pulse primary node to a Pulse data node, cluster-level settings are not migrated.

For example, if you defined some Java settings for the primary node, these settings are not carried over to the new machine now defined as the primary node.

You must specify these settings again in the new Pulse primary node.

Related Topic

Accessing Pulse Service Manager Web Page

Switching from Managed to Unmanaged Libraries and Vice Versa in Pulse Primary Node

You can work with managed or unmanaged libraries depending on the requirements of your design flow. If you switch from managed to unmanaged libraries or vice versa, you **must** restart the Pulse cluster. Any open client applications, such as System Capture, must also be restarted by designers.

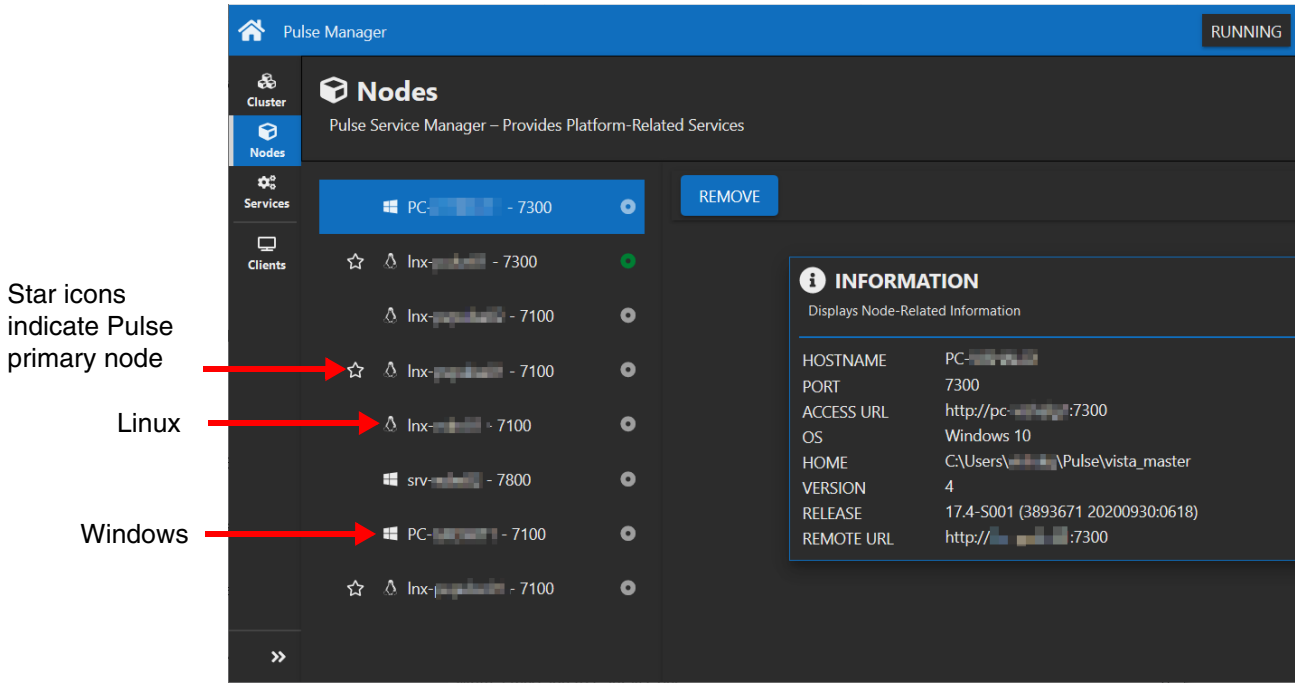
When you switch from:

- managed to unmanaged libraries, Unified Search continues to display parts from managed libraries until System Capture is restarted.
- unmanaged to managed libraries, Unified Search displays an error when designers try and view part details. Exiting and launching System Capture addresses this problem.

For both tasks, first do the following:

1. Start Pulse Service Manager to access its web page with the configuration options.
2. Select *Nodes* in the left pane.

   Servers with star icons next to them indicate the Pulse primary nodes. Icons also indicate whether a machine is a Windows or Linux machine.

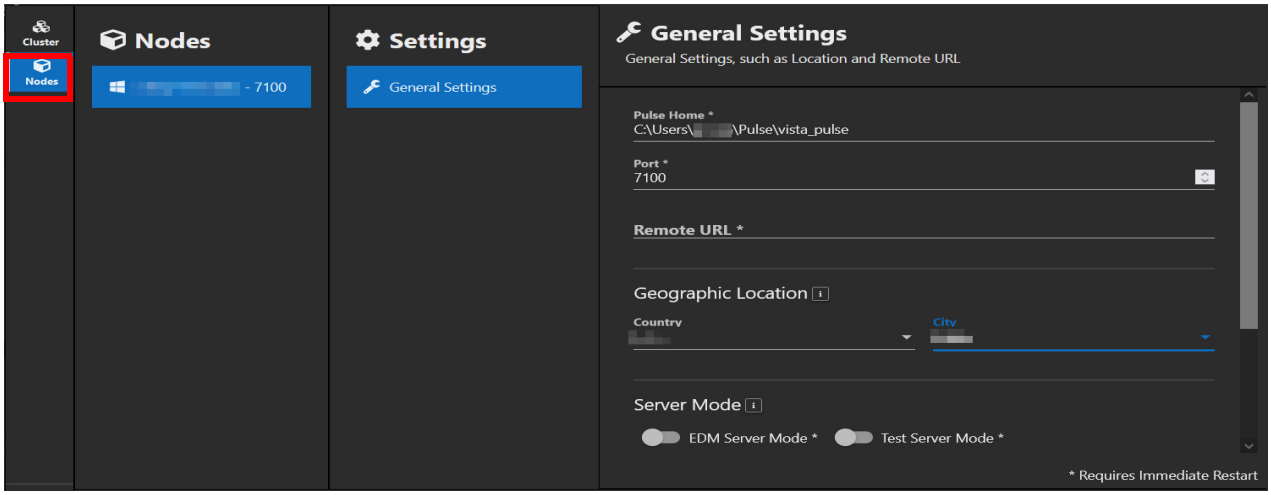


3. Select the Pulse primary node whose library option you want to modify.
4. Click the gear *Settings* icon on the top right of the page.

Switching From Managed to Unmanaged Libraries

To switch from managed to unmanaged libraries, do the following:

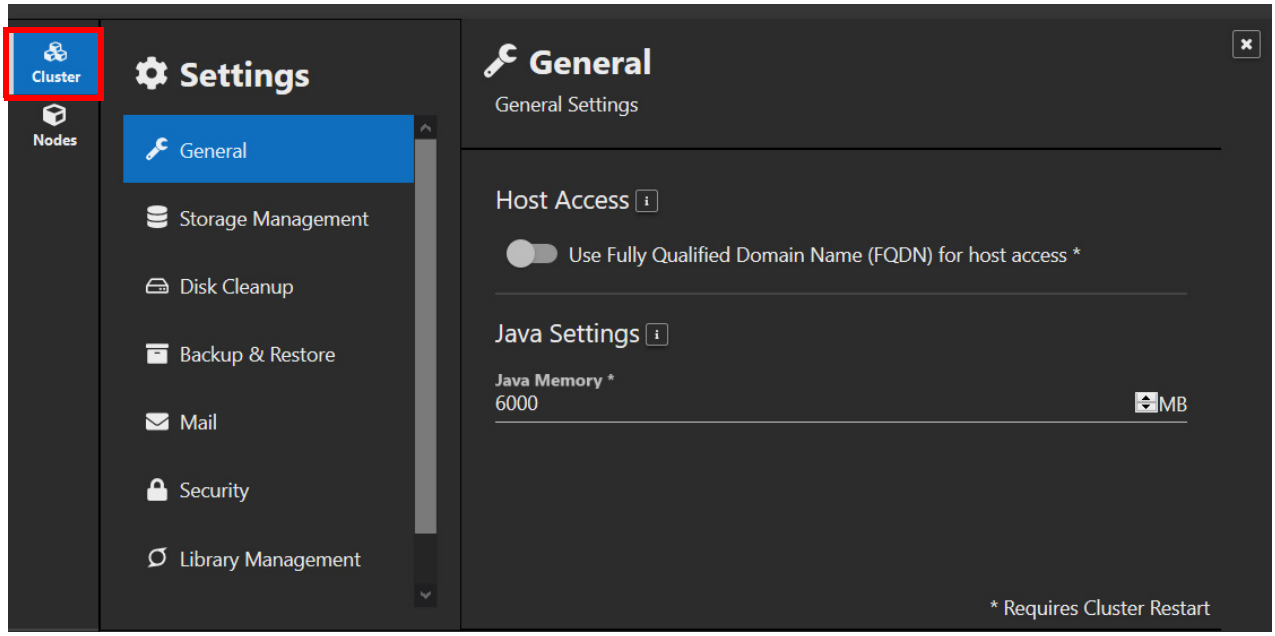
1. Complete steps 1 to 4.
2. Select *Nodes*.



3. Delete the value in the Pulse Home field.

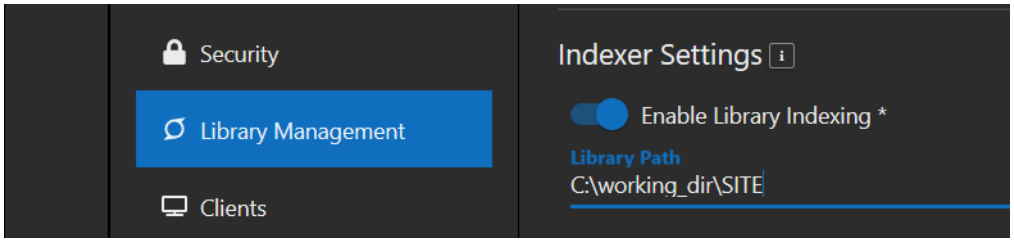
   When you delete the value, a Save button is displayed.
4. Click the *Save* button.

5. Click *Cluster*.



6. Click *Library Management*.
7. Under *Indexer Settings*, enable library indexing to work with unmanaged libraries.

   If you choose this option, you must set the `<CDS_SITE>` environment variable because the Pulse primary node fetches library data from `<CDS_SITE>`.



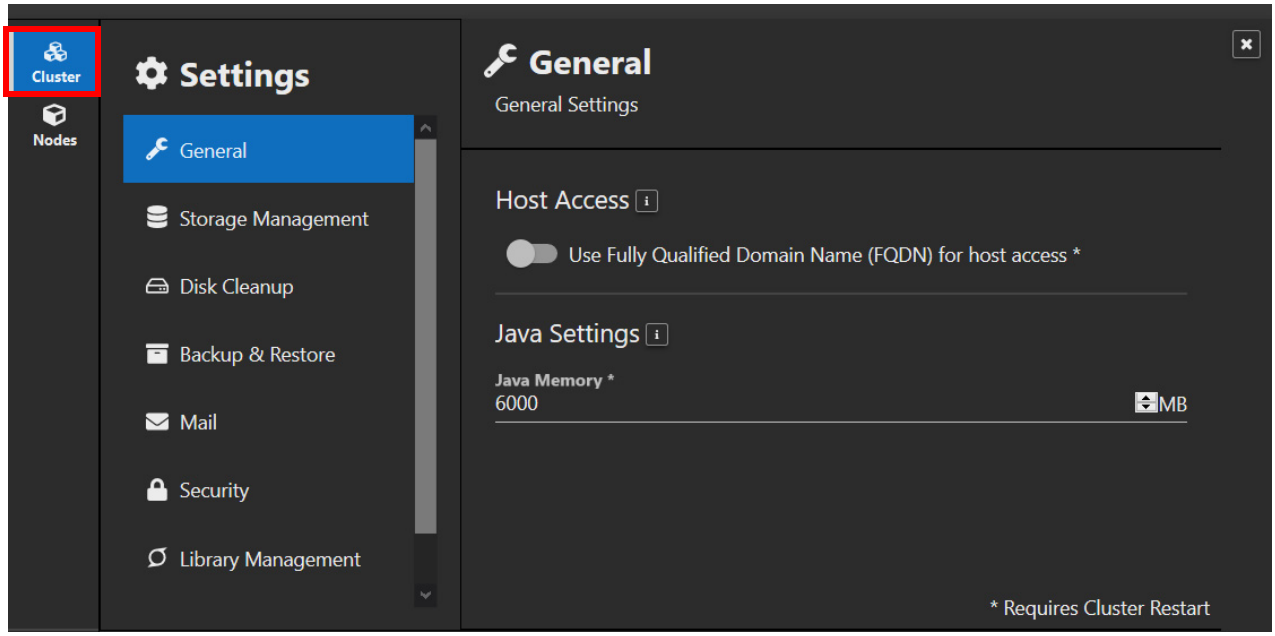
8. Provide the path to the unmanaged libraries if you toggle on the *Enable Library Indexing* button.

Switching From Unmanaged to Managed Libraries

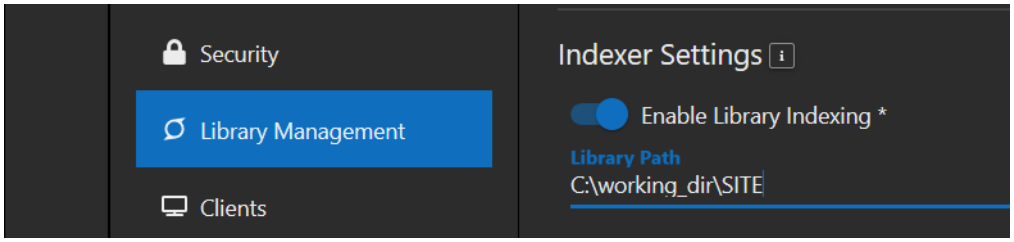
To switch from unmanaged to managed libraries, do the following:

1. Complete steps 1 to 4.

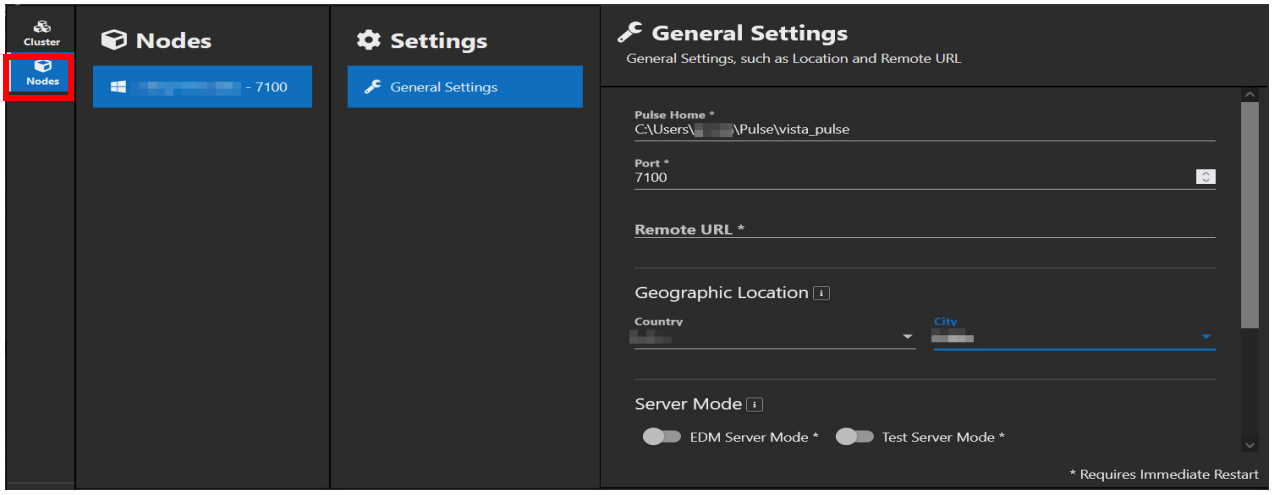
2. Click *Cluster*.



3. Click *Library Management*.
4. Under *Indexer Settings*, disable library indexing and delete the path to the library.



5. Select *Nodes*.



6. If you have a different directory, specify the path to the Allegro EDM-managed library in the Pulse Home field.

   If you do not have a different directory, you can start creating or installing a managed library in the existing Pulse Home.

7. Click the *Save* button.

Related Topic

Accessing Pulse Service Manager Web Page

Creating and Using SSL Certificates in Pulse Environment

To ensure encrypted communication between the Pulse primary and data nodes or client machines, you must do the following:

- Enable SSL encryption.
- Create a Java KeyStore (JKS), which is a repository of security certificates. These can be authorization or public key certificates plus corresponding private keys, used for SSL encryption.
- Use the JKS in the Pulse environment.

To ensure encrypted communication, do as follows:

1. Enable the SSL option in Pulse Service Manager.
2. Shut down the Pulse primary node.
3. Obtain an SSL certificate (public and private key pair) from a Certification Authority.

   The IT team can provide guidance on how this is done at a company.

   a. Use Java keytool or openSSL to import this keypair into a Java KeyStore (`.jks`).

   b. Ensure that alias = `edm` and the KeyStore password = `changeit`.

   c. To generate the self-signed certificate, you can use any key and certificate management utility. For example, openSSL or keytool.
4. For Pulse to use `edm.jks` as a Key Store, do the following:

   a. Copy `edm.jks` to `<Pulse primary node home>/server/conf/aurora/security` folder.

   The `security` folder needs to be manually created at this location. If you run Pulse as a service on a Linux machine, also see Manually Modifying Files in the Pulse Primary Node Home Directory.

   b. Restart the Pulse primary node.

   It is now HTTPS enabled.

Related Topic

Specifying Security Settings for Pulse Primary Node

Mapping of Pulse Internal Service Names and UI Terms

As a microservice framework that provides services such as library management, part search, embedded data management, enterprise PLM integration, Pulse has internal names for various services.

Although these service names are visible in the graphical user interface with **user-understandable** terms, the internal service names are visible in command prompt windows and in Pulse Manager and Pulse Service Manager.

For a better understanding of which internal name maps with which GUI term, see the following table:

Internal Service Name	User-Facing Term
Apollo	Pulse API Documentation
Athena	Pulse Dashboard
Atom	Pulse
Beehive	Pulse Datamart
Bifrost	Pulse Data Management
Element	Pulse Manager
Executrix	Pulse Task Manager
Felix	Authentication Service
Workflow	Pulse Workflow
Hydra	Pulse Block Storage
Iris	Messaging
Kronos	Version Graph
Magneto	Pulse Message Bus
Minerva	Pulse Library Manager
Oculus	Pulse Event Viewer
Pantheon	Pulse Metadata Store
Raven	Pulse Email Notification
Salus	Pulse User Management
Shipper	Pulse Logging
Terminus	Pulse API
Unicorn	Unified Search
Vault	Allegro EDM Vault
Vista	Pulse Service Manager

Troubleshooting Pulse Server Management

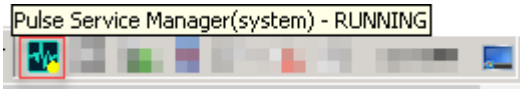
This section lists some commonly encountered problems when you run Pulse.

Pulse Service Manager web page not automatically displayed when the Pulse server is started

If Pulse was **not** configured as a service, the Pulse Service Manager page is automatically displayed in a browser when you start the Pulse server so that you can specify the login credentials.

If the page is not automatically displayed, do the following:

1. Hover the mouse cursor over the Pulse icon ( ) in the taskbar notification area and check that the status is RUNNING.



2. Click the Pulse icon.

   A yellow dot on the icon ( ) means there is a notification. For example, a message is displayed that Pulse is in the maintenance mode because of a scheduled task, such as backing up data.

   If Pulse is unavailable because of scheduled maintenance tasks, it resolves the notification itself and the yellow dot disappears. If it is a task you need to address, do so, and the dot and notification are no longer displayed.

3. Select *Manage* to open the Pulse Service Manager web page.

   Pulse Service Manager is displayed in a web browser.

Allegro EDM Utilities Called by Allegro EDM Flow Manager

This section lists the Allegro EDM utilities that are called from Allegro EDM Flow Manager. Some of these can also be invoked from Allegro EDM System Console.

cksyn

Used internally for synonym checks from design and highspeed flows. It can be used directly for these flows only.

create_sch

Used only for library projects and is used in library flow. This utility generates a schematic for the checked out cell model by first generating a cpm file and then it creates a view tst_sch_1.

createflow

Creates new customized flows. This creates a new project template and a new flow file. These flow files and template files are then edited to meet specific needs.

dbadmin

Launches the Database Administrator tool.

design_init

Launches the Project Creation Wizard. This allows you to create a new project from the given project templates.

design_init_update

Updates the project files of the active project.

diag_check

This utility launches the standard checkplus tool for design verification. From the license dialog box, when you select the appropriate license, the standard checkplus GUI appears.

diffnet

Used with design projects only. Compares and generates a report on differences between two netlists generated by mknet.

diffnet_ui

Compares reference and new netlist. Launches the Diffnet wizard where the reference and new netlists are specified.

dispenv

Displays the complete set of environment variables used in Allegro EDM environment.

editconfig

Launches the `editconfig` wizard. Has different modes to edit the configuration files.

find_project

Displays the current/active project with complete path.

genmodelhtml

Creates an HTML file corresponding to each model type. Takes each model-type and retrieves model data for it.

getexetoolspath

Displays the complete path to exe_tools that are being used in that session.

getlang

This utility displays the language that the tool version is using. The output of this command is `en_US`.

install_dump

Updates the database with the data that has been fetched from the primary site, using the `fetch_dump` utility, onto the client-site database server.

install_model

The models to install are available in archived form in the client site's integration area. This utility installs the models received on the client site specifically in the `reflib` folder of the reference library structure.

launch_concepthdl

Should be run only after test schematic has been run. This should check if the test schematic has been generated and project has been created.

- This utility launches Design Entry HDL.
- This utility works only with a library project (with a model set to active).

libimport

Moves the legacy library data into Allegro EDM-compliant libraries.

library_diag_check

This utility is used to run some checkplus rules on current active model in the library flow.

library_verif

Launches the Library Design Verification utility, which is used to run a set of predefined verification rules on the active schematic model.

mkdump

Exports the primary site database.

mknet

Generates the netlist for the design.

adwdbcheck

The adwdbcheck utility checks the Allegro EDM Component Database for invalid and erroneous library data. Such data can cause the database to become invalid or unusable. You need to check the report generated by this utility to identify and analyze all such errors to prevent any data loss or corruption.

Allegro EDM Configuration Manager User Interface

This section details the interface components of Allegro EDM Configuration Manager.

- Menu
- Welcome Screen
- Set up or Manage Company & Site
  - Context Menu
  - Command Buttons
- Set up Client

Important

In the Allegro EDM Configuration Manager dialog boxes, exclamation signs indicate mandatory fields. After you specify a value for a field, the exclamation sign next to it is removed.

Menu

The menu has the following options:

Table A-1 Configuration Manager Menu Bar

Menu	Option	Description
File	Exit	Prompts you to save all changes and then closes the Configuration Manager.

Table A-1 Configuration Manager Menu Bar

Menu	Option	Description
Client	Set up or Manage Company & Site	Launches the Allegro EDM Conf Root Directory wizard if you have not created the root directory, else it displays the site-level `workbench.ini` file.
	Set up Client	Opens a dialog box where you are prompted for the information required for a client to connect to an Allegro EDM server. This information gets written to the *<startworkbench>* file.
Help	Documentation	Opens the Allegro EDM documentation landing page.
	Web Resources	Lists Cadence links where additional information can be found.
	About	Shows the version of the Allegro EDM software you are using.

Welcome Screen

Option	Description
Set up and Manage Company & Site	Launches the Allegro EDM Conf Root Directory wizard if you have not created Allegro EDM Conf Root, else it displays the site-level `workbench.ini` file for an existing Allegro EDM Conf Root.
Set up Client	Opens a dialog box where you are prompted for the information required for a client to connect to an Allegro EDM server. This information gets written to the *<startworkbench>* file.

Set up or Manage Company & Site

Allegro EDM Configuration Manager checks if the location you provided has an existing Allegro EDM Conf Root or not.

- If there is a previously defined company and its site(s), the wizard closes and the Allegro EDM Conf Root is loaded in the *Set up or Manage Company & Site* tab.
- If no Allegro EDM Conf Root is found, the wizard prompts you for company and site details and then Allegro EDM Conf Root is loaded.

Specify the company and site as the primary, if applicable.

- If this is the first time you are performing this task, you should select the primary company and site.
- You can set the primary site and company later on also, or if needed, modify the values you set now.

Manage Company and Site

The Set up or Manage Company & Site displays the:

- The Allegro EDM Conf Root structure, with all the companies and sites under it
- Master site
- Default site
- `workbench.ini` file, specifying:
  - Type of database
    - Allegro EDM Server: This is the default value for all Allegro EDM installations.
    - No Database: This is used for Allegro EDM Flow Manager only installations, in which you use the `board_ref` type of project workspace that functions in non-cache enabled mode.
  - URL for Master Library Server.
  - One or more designer server URLs separated by semicolons. The complete URL must include the port number. For example: `http://myserver:7100;http://myserver:7200`

    Note: In the *Designer Server URL* field, you can also specify a variable `$env(<variable_name>)`, where *`<variable_name>`* is a predefined environment variable.

For example:

Value of *Designer Server URL* is `$env(my_designers)`,

where `my_designers` is a predefined environment variable and its value is set to: `http://myserver:7100;http://myserver:7200`

- Library Administrator ID and password to connect to the component database.
- Location of the Allegro EDM Reference Library

  The *<PCBDW_LIB>* is created at this location for the site.

- The tree in the left panel can be expanded to view and change configuration information:
  - The applications configured for each site
  - The tools configuration (tools.ini)
  - Library distribution configuration
  - Project workspace configuration
  - Tool configuration

    The following figure shows you an expanded list. The files are grouped logically and you can expand and modify if required for your site. To know about each of the entries, read the tooltip displayed.

Context Menu

The context menu is available at the Allegro EDM Conf Root, company, and site levels. The various menu options are self-explanatory:

- Create company
- Reload
- Create Site
- Cut
- Copy
- Paste
- Rename
- Delete

- Set as Master Site
- Set as Default Site

Command Buttons

Option	Descriptions
Save	Updates the configuration files with the changes you made
Save to all sites	Replicates the changes to all the sites in the company
Reset	Resets the values in the configuration files to the older saved versions

Set up Client

Specify values for the fields in the `<startworkbench>` file:

- Allegro EDM Home
- Allegro EDM Conf Root
- Allegro EDM Projects Home
- Library Directory
- CDS Root
- MS Office Home
- Office Viewers Home
- Acrobat Home
- Internet Explorer Home
- Wordpad Home

B

Configuration Manager User Interface

This section details the interface components of Allegro EDM Configuration Manager.

- Menu
- Welcome Screen
- Set up or Manage Allegro EDM Servers
- Set up or Manage Company & Site
  - Context Menu
  - Command Buttons
- Set up Client
- Server Status

Important

In Allegro EDM Configuration Manager dialog boxes, the exclamation signs indicate mandatory fields. Once you specify a value for a field, the exclamation sign next to it is removed.

Menu

The menu has the following options:

Table B-1 Configuration Manager Menu Bar

Menu	Option	Description
File	Exit	Prompts you to save all changes and then closes the Configuration Manager.
Server	Server Management Dashboard	Launches server setup wizard. After you have installed Allegro EDM, or when you need to change any of the server settings, you use this option.
	Server Status	Shows the current status of the server for the currently active site
	Metrics Dashboard	Shows a quick overview of the Allegro EDM server and client systems at the current site.
Client	Set up or Manage Company & Site	Launches the Allegro EDM Conf Root Directory wizard if you have not created the root directory, else it displays the site-level `workbench.ini` file.
	Set up Client	Opens a dialog box where you are prompted for the information required for a client to connect to an Allegro EDM server. This information gets written to the *<startworkbench>* file.

Table B-1 Configuration Manager Menu Bar

Menu	Option	Description
Help	Documentation	Opens the Allegro EDM documentation landing page.
	Web Resources	Lists Cadence links where additional information can be found.
	About	Shows the version of the Allegro EDM software you are using.

Welcome Screen

Option	Description
Set up or Manage Allegro EDM Servers	Launches server setup wizard. After you have installed Allegro EDM, or when you need to change any of the server settings, you use this option.
Set up and Manage Company & Site	Launches the Allegro EDM Conf Root Directory wizard if you have not created Allegro EDM Conf Root, else it displays the site-level `workbench.ini` file for an existing Allegro EDM Conf Root.
Set up Client	Opens a dialog box where you are prompted for the information required for a client to connect to an Allegro EDM server. This information gets written to the *<startworkbench>* file.
Server Status	Shows the status of the current server.
Metrics Dashboard	Shows a quick overview of the Allegro EDM server and client systems at the current site.

Set up or Manage Allegro EDM Servers

Master Library Server refers to the librarian server.

- All Allegro EDM library-flow tools have write access to this server.
- To setup and run the Master Library Server, you require its corresponding license.

Designer Server:

- Hosts a read-only copy of the librarian server
- Allows only the design tools to access this server

No changes can be made to this library server.

Note: A Designer Server is not mandatory for Allegro EDM design team members. It is used when a group of designers are located apart in different geographies away from the library development site and accessing data directly from the Master Library Server will be a slow process because of network latency. In such a case, having a local Designer Server helps improve the performance of the design tools.

Set up or Manage Company & Site

Allegro EDM Configuration Manager checks if the location you provided has an existing Allegro EDM Conf Root or not.

- If there is a previously defined company and its site(s), the wizard closes and the Allegro EDM Conf Root is loaded in the *Set up or Manage Company & Site* tab.
- If no Allegro EDM Conf Root is found, the wizard prompts you for company and site details and then Allegro EDM Conf Root is loaded.

Specify the company and site as the master, if applicable.

- If this is the first time you are performing this task, you should select the master company and site.
- You can set the master site and company later on also, or if needed, modify the values you set now.

Manage Company and Site

The Set up or Manage Company & Site displays the:

- The Allegro EDM Conf Root structure, with all the companies and sites under it
- Master site
- Default site
- `workbench.ini` file, specifying:
  - Type of database
    - Allegro EDM Server: This is the default value for all Allegro EDM installations.
    - No Database: This is used for Allegro EDM Flow Manager only installations, in which you use the `board_ref` type of project workspace that functions in non-cache enabled mode.
  - URL for Master Library Server.
  - One or more designer server URL(s) separated by semicolons. The complete URL must include the port number. For example: `http://myserver:7100;http://myserver:7200`

    Note: In the *Designer Server URL* field, you can also specify a variable `$env(<variable_name>)`, where *`<variable_name>`* is a predefined environment variable.

    For example:

    Value of *Designer Server URL* is `$env(my_designers)`,

    where `my_designers` is a predefined environment variable and its value is set to: `http://myserver:7100;http://myserver:7200`
  - Library Administrator ID and password to connect to the component database.
  - Location of the Allegro EDM Reference Library

    The *`<PCBDW_LIB>`* is created at this location for the site.
- The tree in the left panel can be expanded to view and change configuration information:
  - The applications configured for each site
  - The tools configuration (tools.ini)
  - Library distribution configuration
  - Project workspace configuration
  - Tool configuration

The following figure shows you an expanded list. The files are grouped logically and you can expand and modify if required for your site. To know about each of the entries, read the tooltip displayed.

Context Menu

The context menu is available at the Allegro EDM Conf Root, company, and site levels. The various menu options are self-explanatory:

- Create company
- Reload
- Create Site
- Cut
- Copy
- Paste
- Rename
- Delete
- Set as Master Site
- Set as Default Site

Command Buttons

Option	Descriptions
Save	Updates the configuration files with the changes you made
Save to all sites	Replicates the changes to all the sites in the company
Reset	Resets the values in the configuration files to the older saved versions

Set up Client

Specify values for the fields in the *<startworkbench>* file:

- Allegro EDM Home
- Allegro EDM Conf Root
- Allegro EDM Projects Home
- Library Directory
- CDS Root
- MS Office Home
- Office Viewers Home
- Acrobat Home
- Internet Explorer Home
- Wordpad Home

Server Status

This tab displays the following information about the Allegro EDM server:

- Installed Server Components
- Server Schema Version
- Allegro EDM Server Version
- Application Server details:

  URL, Server Type, Server Version, Schematic Flow Directive
- Allegro EDM Library Location
- PCBDW_LIB location
- Allegro EDM Installation Root
- Allegro EDM Configuration Root
- Current Default Company
- Current Default Site
- Cadence SPB Software Root
- Java Virtual Machine Path
- Java Virtual Machine Version
- Operating System Details
- System Hardware Information