ivanti | **Service and Asset Manager**

Ops Console On-Premise User Guide

2019.3.1

Contents

About This Guide

Use the information in this guide in conjunction with the *Installation and Deployment Guide for Service and Asset Manager*. That guide contains step-by-step instructions for installing and configuring your deployment, while this guide only describes procedures related to using the Operations Console.

> For a complete overview of all types of deployments and step-by-step instructions for installing and configuring each deployment, refer to the "*Installation and Deployment Guide for Service and Asset ManagerVersion 2019.3.1*". For information about using Ivanti Service and Asset Manager, including using packages and projects, see the Ivanti Service and Asset Manager online help. (See "Related Documentation" on the next page for more information about accessing the online help.)

This topic contains the following sections:

- "Intended Audience" below

- "List of Topics" below

- "How to Contact Us" on page 7

- "Related Documentation" on the next page

- "About This Guide" above

Intended Audience

This document is intended for system administrators, including both Ivanti personnel and members of other organizations, who will be installing and using the Operations Console to manage tenants and tenant instances and to migrate data.

This guide is aimed at the following:

- The system administrator for an on-premise organization that has purchased, installed, and configured Ivanti Service Manager and who needs to migrate data between tenant instances. An on-premise system administrator works only on the Ivanti Service Manager system for his organization.

> This guide uses the term "customer." For MSP system administrators and Ivanti Software employees, the term customer refers to the actual customer (users from a different organization), but for on-premise system administrators, the term customer refers to users within the same organization.

List of Topics

This document contains the following sections:

- "Operations Console Overview" on page 8: Describes the Operations Console. Includes a terminology list and instructions for installing and logging in to the Operations Console.

- "Before You Begin Migrating Data" on page 21: Lists the four different ways to migrate data and contains conceptual information.

- "Starting the Data Migration" on page 34: Contains information such as how to create a tenant and what you must do before you migrate data for every migration method.

- "Using the Ivanti Service Manager Development Project to Migrate Data" on page 41: Describes how to update an existing tenant instance by migrating data between tenant instances by using the Ivanti Service and Asset Manager development project.

- "Troubleshooting and Known Issues" on page 51: Lists troubleshooting tips and known issues with the Operations Console.

- "Appendix A: Using the Operations Console to Perform Administrative Tasks" on page 63: Explains how to use the Operations Console to configure landscapes and tenants, as well as how to work with users, system settings, and the audit history.

- "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94: Describes how to update an existing tenant instance by migrating data between tenant instances by using Simple Mode or Advanced Mode.

- "Appendix C: Using the ISM Development Project to Migrate Data" on page 128: Contains the information from the Ivanti Service Manager online help that describes how to use the Configuration Console to migrate data using the Ivanti Service and Asset Manager development project.

- "Appendix D: Working with Packages from the App Store" on page 142: Describes how to migrate data that came from the App Store.

Related Documentation

Ivanti Service and Asset Manager has online help available within the application. To access it, click the **Help** button on the top right of the application.

Additional documentation is available through

- The [Ivanti community](#) website. You may need to request user access if you cannot log in.

    Or through

- The [Ivanti Product Documentation](#) website. Click the Service Manager tile to see a list of the documents available.

The online help is different depending on the role you used to log in. There are three types of online help:

- **Online help for the Self-Service Portal**: For anyone who logs in to the Self-Service Portal.

- **Online help for administrators**: For anyone who logs in as an administrator to the Configuration Console.

- **Online help for users**: For anyone who logs in to the Service Desk Console.

How to Contact Us

To contact us about the documentation, or if you have any other questions or issues about Ivanti Service Manager, contact Ivanti Software Global Support services by logging an incident via Self Service at: https://www.ivanti.com/support/ivanti-support.

Operations Console Overview

This section contains the following:

- A brief description of the Operations Console features. See "About the Operations Console" below.

- Instructions for navigating the Operations Console interface. See "Operations Console Interface" on the next page.

- A summary of the actions that you can perform from each area of the Operations Console. See "About the Different Workspaces in the Operations Console" on the next page.

- Terminology used in this guide, including definitions of data types. See "Terminology" on page 10.

- Instructions for installing and logging in to the Operations Console. See "Installing the Operations Console" on page 18 and "Accessing the Operations Console" on page 19.

About the Operations Console

The Operations Console provides an interface in which you can make changes to your Ivanti Service Manager system (usually in the staging instance of the tenant), test the changes (usually in the UAT instance of the tenant), and then deploy them (to the production instance of the tenant). For example, in any landscape you can:

- Create tenant instances.

- Edit the properties of tenant instances.

- Lock a tenant instance so that its metadata cannot be modified.

- Deactivate and delete tenant instances.

- Update tenant instances by migrating data from one tenant instance to another.

- Control the timing of tenant development and deployment.

The Operations Console supports these capabilities by connecting to the Ivanti Service Manager configuration server in each landscape. The Operations Console has its own login credentials and database, which provide logging and other tenant management history capabilities.

The Operations Console also provides you with a direct connection to the Ivanti Service Manager configuration database in each landscape, allowing you to manage the Ivanti Service Manager configuration database for that landscape without going through the Ivanti Service Manager configuration server.

The Operations Console requires a database connection to the Ivanti Service Manager configuration database, a database connection to the Ivanti Service Manager application database, and a web services connection to the Ivanti Service Manager application database.

Operations Console Interface

Operations Console Main View (Tenants Workspace Displayed)
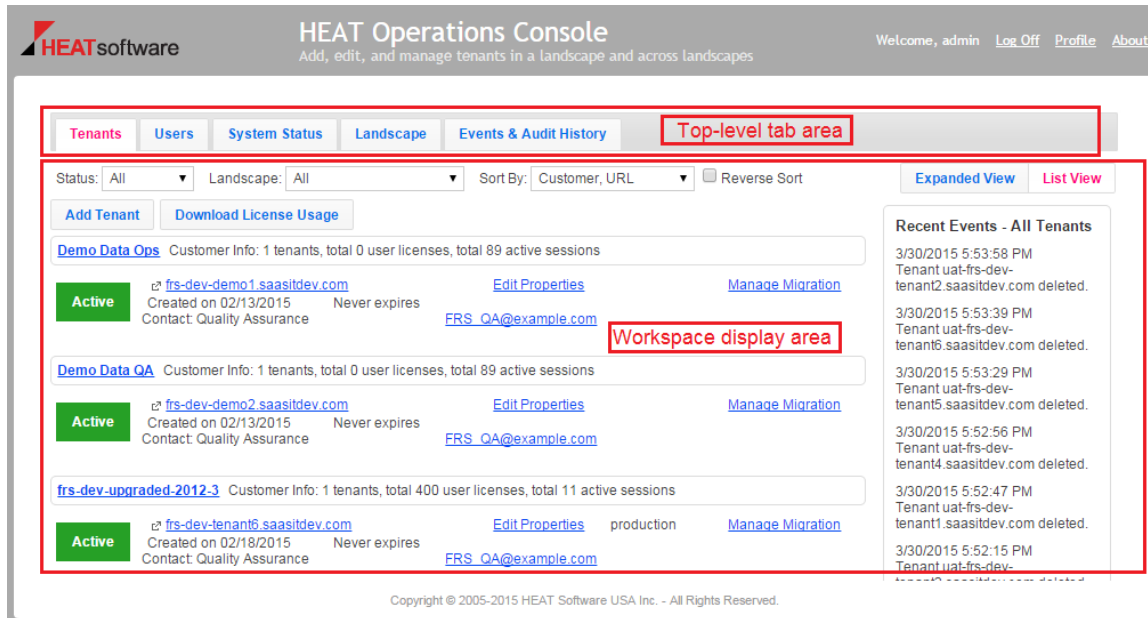
The Operations Console, shown in "Operations Console Main View (Tenants Workspace Displayed)" above, contains these areas:

- A top-level tab area that you use to select the workspace to display. Click any tab to open it in the workspace display area. When you select a tab, the system displays the tab name in a pink font.

- A workspace display area containing the controls and detailed information for the workspace that opens when you select a top-level tab.

About the Different Workspaces in the Operations Console

Click a top-level tab to open it in the workspace display area. The following sections describe the actions that you can perform in each workspace.

- "About the Tenants Workspace" on the next page

- "About the Users Workspace" on the next page

- "About the System Status Workspace" on the next page

- "About the Landscape Workspace" on the next page

- "About the Events & Audit History Workspace" below

About the Tenants Workspace

Use the **Tenants** workspace to perform a variety of tenant management activities, including creating, locking, migrating, and deleting tenants and tenant instances. See "Working with Tenants and Tenant Instances" on page 82.

About the Users Workspace

Use the **Users** workspace to view and edit information about the users who can access the Operations Console. See "Working With Operations Console Users" on page 63.

About the System Status Workspace

Use the **System Status** workspace to view status information about the entire Ivanti Service Manager system. See "Viewing System Information" on page 67.

About the Landscape Workspace

Use the **Landscape** workspace to view and edit information about landscapes. See "Configuring Landscapes" on page 71.

About the Events & Audit History Workspace

Use the **Events & Audit History** workspace to view information about each action that has occurred on a tenant or tenant instance. See "Viewing Events and Audit History" on page 70.

Terminology

This section defines key terms pertaining to the Operations Console and the tenant creation and data migration process.

- "General Terms" below

- "Types of Data" on the next page

- "Landscapes" on page 15

- "Tenant and Tenant Instance Types" on page 17

General Terms

- "Operations Console" on the next page

- "Terminology" above

- "Terminology" on the previous page

- "Backup Types" below

Operations Console

The tool used to create tenants and tenant instances, move data from one tenant instance to another (for example, from the staging instance of a tenant to the production instance), and perform various administrative tasks.

Backup Types

There are three steps to the process of backing up a database:

1. Back up the database on the source system.
2. Copy the database from the source system to the target system.
3. Restore the database on the target system.

Use Microsoft SQL to perform the backup. There are several backup types you can choose from when using the Operations Console:

- **From last MSSQL backup**: Copies the source database from the source database backup location to the target database backup location and restores the database from there.

- **From live MSSQL backup**: Performs a live Microsoft SQL backup of the source database and places the backup in the source database backup location. Copies the source database backup to the target database backup location and restores the database from there.

- **From copied MSSQL backup**: The source database has already been manually copied to the target database backup location. If you choose this option, the system adds a field called **Backup Filename** where you enter the name of the Microsoft SQL backup.

- **Database already restored**: The database was already restored manually outside of the Operations Console (for example, through Microsoft SQL Server Management Studio). Choosing this option means the database is not backed up or restored.

Types of Data

- "Configuration Data (Settings)" on the next page

- "Master Data" on the next page

- "Metadata" on page 13

- "Transactional Data" on page 14

- "Validation Data" on page 14

Configuration Data (Settings)

Configuration data, also called settings, is defined as core configuration attributes that define Ivanti Service Manager.

The only configuration data (settings) that the system migrates when you migrate configuration data (settings), as described in "Updating Configuration Data (Copy Settings)" on page 115, are the following:

- Allowed attachments

- Authentication providers such as LDAP, SAML, OpenId, and Windows Integrated Security

- Bulk upload profiles

- Dynamic styles

- Global constants

- Integration settings

- LDAP configurations

- Object matching lists

- Password policies

- Voice integrations

- Watch list items

- Email configurations (including approval status keywords, email status update mapping settings, ignore message with subject settings, object mappings for email, task assignment status keywords, truncate body settings, and trusted sender domains)

You cannot migrate other types of configuration data (settings) using the Operations Console. To migrate this data, do so manually outside of the Operations Console by using Microsoft SQL commands.

Master Data

Master data is defined as data sets that are used across an organization and are typically unique to that organization. User and role definitions, role assignments, and categories are examples of master data.

Master data is migrated together with metadata so that it is consistent with the business object definitions. There are four types of master data and you can migrate any or all of them:

- User data, except for user email addresses because they may have been scrubbed.

- User-related data, such as contact group, standard user team, organizational unit, and so on.

- Service request data, such as request offerings, request parameters, and so on.

- Escalation schedule.

You can now track these types of master data:

- Analytic metrics dashboards and configuration

- Schedule entries

- Escalation settings and schedules

- Escalation clock control

- Service level packages

- Service level targets

To see a list of master data for a tenant, if any has been defined, place your mouse over the destination tenant.

The only way to migrate master data is by enabling the Ivanti Service Manager development project and using the push link. See "The Four Ways to Migrate Data" on page 21 for more information about this method of migrating data and see "Example: Migrating Service Level Data" on page 126 for an example.

Metadata

Configured business objects, business object attributes, forms, grids, quick actions, searches, and so on. When you reconfigure your Ivanti Service Manager system, most of the changes are implemented by editing metadata.

The types of metadata are as follows:

- Business objects

- Counters

- Forms

- Grids

- Layouts

- Quick actions

- Rules

- Rule overrides

- Branding

- Roles

- Validation (pick list) definitions

- Business rules

- Searches (namely saved searches)

- Dashboards

- Chart-part metadata (one type of dashboard)

- Grid-part metadata (one type of dashboard)

- Special-part metadata (one type of dashboard)

- View-part metadata (one type of dashboard)

- Workflows

- Workflow types

- Reports (subscription, schedule, and distribution are not migrated from source to target)

- Validation data (of pure validation business objects and custom data)

- Images in the frs_def_images table

You can lock all of the metadata, except for quick actions, dashboards, searches, and reports. Those four types of metadata cannot be locked and are always open to be edited in the production instance of the tenant. To lock the other types of metadata, set the metadata to read-only. Each tenant instance has a **Metadata Locked** field that lets you know if the metadata is locked for an instance of a tenant.

Transactional Data

A customer's data records, including incidents, changes, service requests, and so on.

The only way to migrate transactional data is by enabling the Ivanti Service Manager development project and using the push link. See "The Four Ways to Migrate Data" on page 21 for more information about this method of migrating data and see "Example: Migrating Service Level Data" on page 126 for an example.

Validation Data

Validation data objects defined by customers. You must migrate all validation data together at the same time and you must migrate metadata and validation data together in the same migration. This guarantees that when metadata is migrated, it includes all pick list definitions that are associated with validation objects.

Validation data is any data that is configured as part of a *validation* business object. A *validation* business object and its associated data (for example, a pick list and the data in the list) are both considered validation data. But a *standard* business object and its associated data (for example, a pick list and the data in the list) are not both validation data. If the business object is a *standard* business object, then that is validation data, but the associated data is not considered validation data.

Business Object	Is it Validation Data?
Standard business object	Yes
Validation business object	Yes
Data associated with a standard business object	No
Data associated with a validation business object	Yes

Landscapes

- "Landscape Definition" below

- "Landscape Components" on the next page

- "Types of Landscapes" on page 17

- "Terminology" on page 10

Landscape Definition

A landscape is a set of IT infrastructure components that support tenant instances of the same type (where type refers to production, staging, and UAT). The production landscape only contains the production instances of tenants, the staging landscape only contains the staging instances of tenants, and the UAT landscape only contains the UAT instances of tenants.

"Typical Production Landscape" below shows a typical configuration for a production landscape. Components and their locations vary from customer to customer. For example, on-premise customers probably only have one production tenant in their production landscape.

Typical Production Landscape
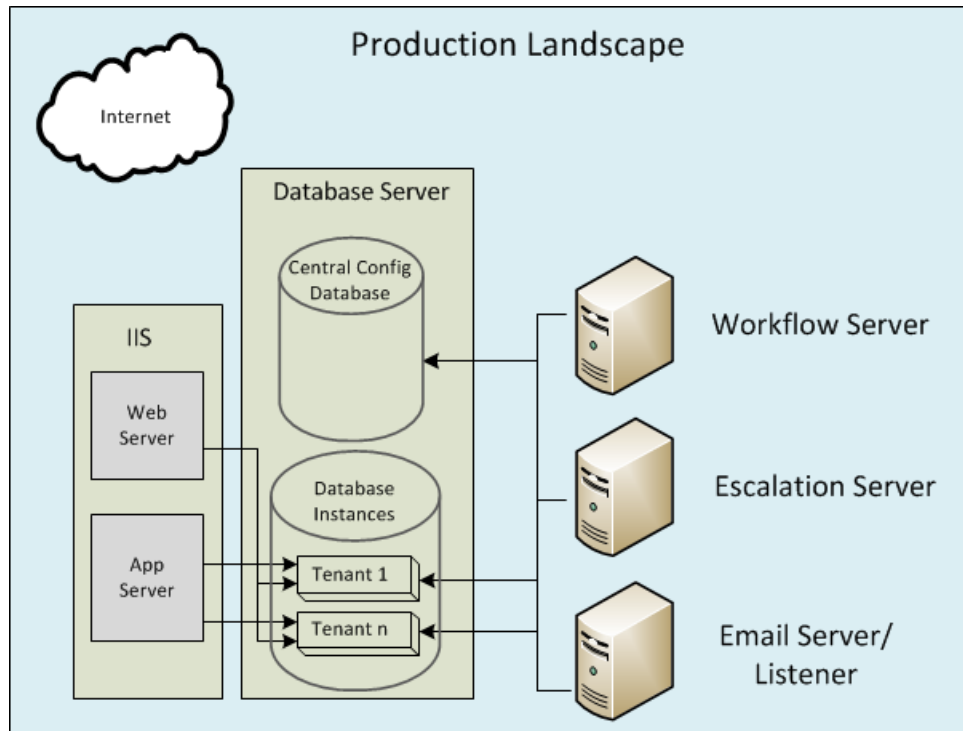
Landscape Components

At a minimum, a landscape contains these components:

- An individual, isolated Microsoft SQL Ivanti Service Manager application database (called IvantiSM) in each tenant (each tenant instance has its own database, but not its own database instance).

- A central Ivanti Service Manager configuration database (called ConfigDB) containing information about all tenant instances in the landscape.

- A shared pool of Ivanti Service Manager application servers for all Ivanti Service Manager instances in the landscape.

- A shared database cluster for the Ivanti Service Manager configuration and Ivanti Service Manager application databases.

- A shared pool of workflow servers used by all Ivanti Service Manager instances in the landscape.

- A shared pool of escalation engines used by all Ivanti Service Manager instances in the landscape. This is sometimes combined with the workflow server.

- A shared pool of email listeners used by all Ivanti Service Manager instances in the landscape.

- A shared pool of integration servers per function, such as survey, Discovery, Desktop Server and Management (DSM), Report Server, License Server, and so on.

Types of Landscapes

There are six types of landscapes:

- **Production landscape**: Hosts the production instances of tenants. Used during the data migration lifecycle. This is the environment where the product is deployed.

- **Staging landscape**: Hosts the staging instances of tenants. Used during the data migration lifecycle. This is the environment where development work is done.

- **UAT landscape**: Hosts the UAT instances of tenants. Used during the data migration lifecycle. This is the environment where testing is done.

- **Pilot landscape**: Optional. A standalone landscape that hosts the pilot tenant to provide customers with early access to a new Ivanti Service Manager point release.

- **Development landscape**: Optional. A standalone landscape that hosts tenant templates for demos, test drives, training, and so on.

- **Other landscape**: Optional. A standalone landscape type that can be used for any purpose.

Tenant and Tenant Instance Types

A tenant is a Ivanti Service Manager system for a particular customer or application.

An instance of the Ivanti Service Manager tenant within a landscape. By default, each customer receives a production instance of the tenant and a staging instance of the tenant. Additional optional tenant instances are also available. The name and URL for each tenant instance must be unique.

> ℹ️ The term "tenant instance" is NOT related to the term "database instance". The use of the word instance, when referring to a tenant, simply means one copy or version of it.

The available tenant instances are:

- **Production instance of a tenant**: A tenant instance containing your production database (including all configuration and transactional data). All customers have a production instance of their tenant. The production instance of the tenant resides in a hardened environment in the production landscape. We do not recommend making metadata or configuration changes directly to the production instance of the tenant during production usage. Instead, make those changes in the staging instance of the tenant, push them to the UAT instance of the tenant for testing, and then push them to the production instance of the tenant.

  You can make some changes to master data (such as adding employees, role assignments, and so on) directly to the production instance of the tenant.

- **Staging instance of a tenant**: A tenant instance for developing and configuring your Ivanti Service Manager system and business processes. This tenant instance is generally used for configuration and metadata changes (but not transactional data changes). It undergoes a baselining process, in which it is updated periodically or by customer request from the production instance of the tenant to ensure that it is in sync with the production instance of the tenant prior to configuration and metadata changes. It can also be used to test new releases, new capabilities, and for training. All customers have a staging instance of their tenant.

- **UAT instance of a tenant**: A temporary tenant instance used for configuration testing and validation before the new configuration is pushed to an existing production tenant instance. The UAT instance of the tenant is required when reconfiguring an existing production instance of the tenant. It is optional when provisioning the production instance of the tenant for the first time or when upgrading to a new Ivanti Service Manager release.

On-premise customers may or may not use a UAT instance of a tenant.

- **Pilot instance of a tenant**: During the upgrade process to a new release, an optional pilot instance of the tenant is made available to customers to allow early access to a new point release. The pilot instance of the tenant resides in the pilot landscape.

- **Development instance of a tenant**: An optional development instance of the tenant for testing or development. The development instance of the tenant resides in the development landscape.

- **Other instance of a tenant**: An optional instance of the tenant for testing or development. The other instance of the tenant resides in the other landscape.

- **Tenant environment**: Infrastructure that ties together the production, staging, and UAT instances of a tenant. This infrastructure allows a complete change management process to manage and control the development and deployment lifecycles between these instances of the tenant.

- **Tenant template**: A preconfigured ("out-of-the-box") tenant that is typically used to set up a new tenant. A template implements best practices according to ITIL standards, and optimizes processes to allow fast implementation.

Installing the Operations Console

- "Installing the Operations Console" on the next page

- "Accessing the Operations Console" on the next page

Installing the Operations Console

You install the Operations Console when you install Ivanti Service Manager. You can find complete instructions for installing Ivanti Service Manager in the *Installation and Deployment Guide for Ivanti Service Manager*.

We highly recommend that you only install the Operations Console in the staging landscape, and not in any other landscape. For migrating data, you only need to install the Operations Console in one landscape; you should not install it in more than one landscape.

The following is a summary of the installation steps:

1. Access the installation folder on the Ivanti Software product CD or download folder and run IvantiServiceManager.exe. The system displays the **Welcome** dialog box.

2. Click **Next**. The system displays the **License Agreement** dialog box.

3. Select **I accept the terms in the license agreement** and click **Next**. The system displays the **Destination Folder** dialog box.

4. Click **Next** to accept the default installation folder. The system displays the **Setup Type** dialog box.

5. Do one of the following:

   • Select **Complete** for the installation type. The installation program installs all of the Ivanti Service Manager components including the Operations Console.

   • Select **Custom** for the installation type. Choose the components to install but ensure that you select **Operations Console** under Setup/Tools.

6. Click **Install** in the **Ready to Install the Program** dialog box.

   The installation begins. The system displays a status dialog box, showing the installation progress of each component over the next few minutes.

7. If prompted, click **Run** to run the file.

Accessing the Operations Console

The Operations Console has the following default login credentials:
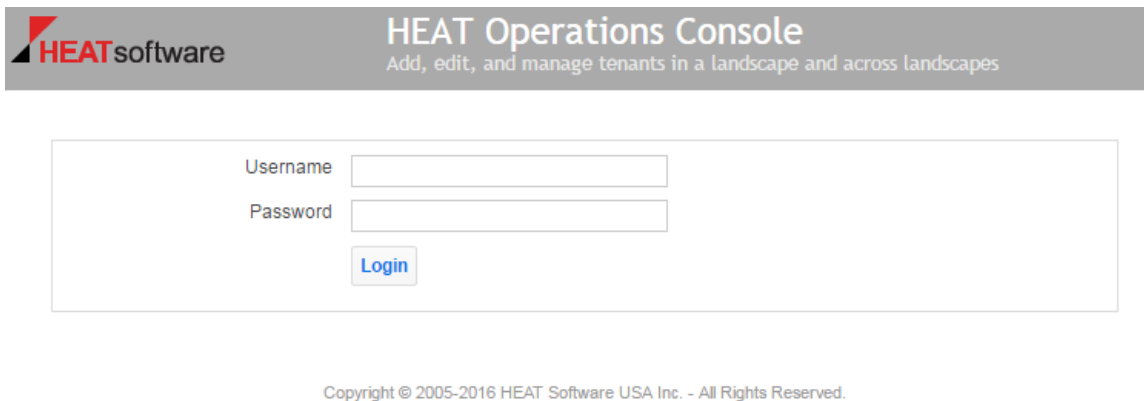
   • User name: admin

   • Password: manage

We highly recommend that you change the administrator password the first time that you access the system.

After installing Ivanti Service Manager, do the following to access the Operations Console:

1. Either click **Start > All Programs > HEAT Software > HEAT > Ops Console** or access the Operations Console from a URL in a browser. The system displays the Operations Console login page. See "Operations Console Login Page" below.

Operations Console Login Page

2. Enter the Operations Console user name and password.

3. Click **Login**.

Before You Begin Migrating Data

- "The Four Ways to Migrate Data" below

- "About Data Migration" on the next page

- "Conceptual Information: Migration Process Overview" on the next page

- "Items Excluded from Migration" on page 32

The Four Ways to Migrate Data

Before you begin migrating data, it is very important that you understand the different ways there are to migrate data. There are four ways that you can migrate data:

- Enabling the Ivanti Service Manager development project and applying a package. With a package, you can migrate a very specific set of configuration changes. For example, you can add service request data to a package and migrate it. See "Appendix C: Using the ISM Development Project to Migrate Data" on page 128.

- Enabling the Ivanti Service Manager development project and using the push link. With this method, you can migrate master data, but not configuration data (because that is moved using packages). See "Appendix C: Using the ISM Development Project to Migrate Data" on page 128.

- Disabling the Ivanti Service Manager development project and using Simple Mode. With Simple Mode, you can only copy service request data and escalation schedules. When you use Simple Mode, you copy all of the configuration data from instance of a tenant to another. See "How to Use Simple Mode to Migrate Data" on page 95.

- Disabling the Ivanti Service Manager development project and using Advanced Mode. With Advanced Mode, you can choose which items to copy, such as all business objects but not workflows. However, you cannot specify which business objects (or whichever item you migrate). You can choose the type of items to migrate but not the specific items within that type. With Advanced Mode, you can also specify if you want to migrate master data, which you can use to copy the contents of a business object. See "Using Advanced Mode to Migrate Data" on page 100.

> We plan to remove Simple Mode and Advanced Mode in an upcoming release. The Ivanti Service Manager development project will then be the only data migration method available. This is because Simple Mode and Advanced Mode only migrate the difference in data between the source and the target, and do not take into account the sequence of the data changes between the two systems.

This document includes information about migrating data using all of the methods listed above. However, we highly recommend that you enable and use the Ivanti Service Manager development project, and therefore, this document focuses on those processes.

About Data Migration

The process of migrating data can be defined as moving information from the production instance of the tenant to the staging instance of the tenant, and then optionally moving information from the staging instance of the tenant to the UAT instance of the tenant, and then finally moving information back to the production instance of the tenant (from either the staging or UAT instance of the tenant).

There are two scenarios:

- Production instance --> staging instance --> UAT instance --> production instance.

- Production instance --> staging instance --> production instance.

The migration process has three parts:

1. **Migrating data from the production instance of the tenant to the staging instance of the tenant.** This process is described in "Creating the First Staging or UAT Instance of the Tenant from the Production Instance of the Tenant" on page 37.

2. **Migrating data from the staging instance of the tenant to the UAT instance of the tenant.** Although this step is optional, we recommend it. This step is made up of two parts:

   a. Migrating the metadata from the staging instance of the tenant to the UAT instance of the tenant. This process is described in "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94.

   b. Migrating the transactional data from the production instance of the tenant to the UAT instance of the tenant. This is the same process as "Creating the First Staging or UAT Instance of the Tenant from the Production Instance of the Tenant" on page 37.

3. **Migrating data from the UAT instance of the tenant to the production instance of the tenant** (if you migrated from the staging instance to the UAT) or **migrating data from the staging tenant to the production tenant** (if you did not migrate data to the UAT tenant). This process is described in "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94.

Conceptual Information: Migration Process Overview

There are two scenarios when you perform the migration process:

- When you initially purchase and provision your system. See "About Initially Provisioning the Production Instance of the Tenant" on the next page.

- When the database changes. See "About Migrating Data After Creating the Initial Production Instance of the Tenant" on page 27.

About Initially Provisioning the Production Instance of the Tenant

- "Overview of Initially Provisioning the Production Instance of the Tenant" below

- "Overview of the Initial Production Instance of the Tenant Creation Process" on page 25

Overview of Initially Provisioning the Production Instance of the Tenant

"Adding a Tenant" below shows the process for migrating data from the initial production instance of the tenant (version 0.0) to the staging instance of the tenant, and then optionally from the staging instance of the tenant to the UAT instance of the tenant, and then from the staging (or UAT) instance of the tenant back to the production instance of the tenant. This process establishes the first production instance of the tenant (version 1.0). See "Overview of the Initial Production Instance of the Tenant Creation Process" on page 25 for information about the numbered steps shown in "Adding a Tenant" below.
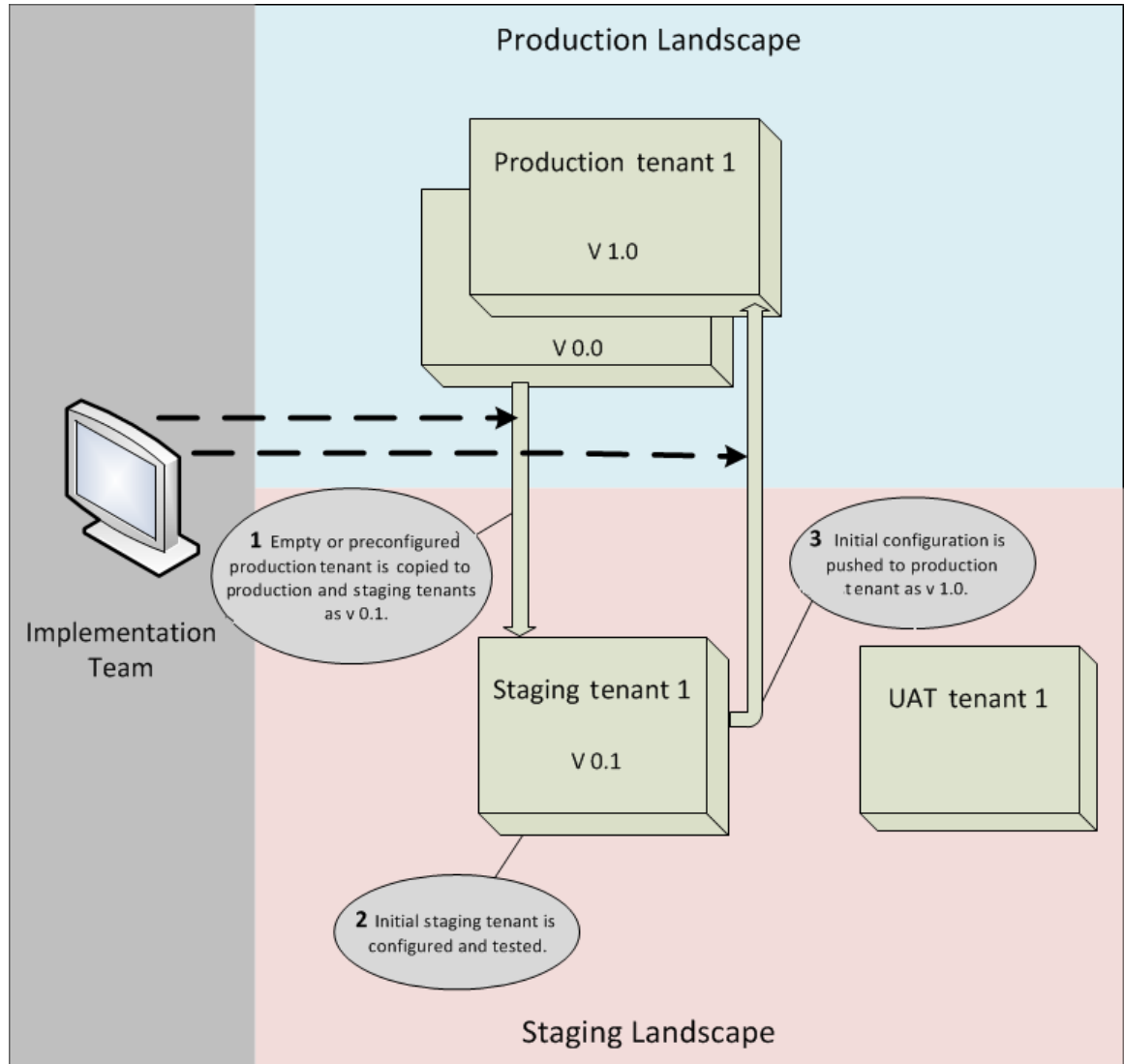
Adding a Tenant

Things to note in **"Adding a Tenant" on page 23**:

- The version numbers used in this example are for reference only. The software residing on a tenant does not actually have the version numbers shown here.

- Version 0.0 of the production tenant represents the starting state of the production tenant immediately after it was created. This version (or instance) typically does not contain any transactional (customer) data. You can preconfigure the metadata by either using a customer template, or if no template is available, in a way that provides a default user interface look and feel. Examples of templates are a retail business template and a high tech template.

- When creating the initial version (or instance) of the production tenant (version 1.0), the system usually bypasses the UAT instance of the tenant because there is no baseline configuration to use as a reference for regression testing.

You do not make changes to configuration data and metadata directly in a running database on the production instance of the tenant. Instead, you copy the database to the staging instance of the tenant and makes the changes there. The production instance of the tenant remains unlocked throughout the provisioning and configuration process so that you can update the production instance of the tenant with minimal overhead. The system locks the production instance of the tenant for the first time after you verify the production readiness of the production instance of the tenant.

Overview of the Initial Production Instance of the Tenant Creation Process

The following sections provide a high-level description of the initial production instance of the tenant creation process.

- "Before Starting the Migration" below

- "Step 1a: Creating the Initial Production Instance of the Tenant (Version 0.0)" on the next page

- "Step 1b: Creating the Initial Staging Instance of the Tenant (Version 0.1)" on the next page

- "Step 2: Updating the Data on the Staging Instance of the Tenant (Version 0.1)" on the next page

- "Step 3: Pushing the Data from the Staging Instance of the Tenant (Version 0.1) to the Production Instance of the Tenant (Version 1.0)" on the next page

Before Starting the Migration

- Determine your system requirements to help you design the initial production instance of the tenant (version 0.0).

Step 1a: Creating the Initial Production Instance of the Tenant (Version 0.0)

- Create the initial production instance of the tenant, called version 0.0. This initial production instance of the tenant contains basic help desk and other functionality based on industry standards and preliminary requirements. Version 0.0 is used as a starting point for additional configuration.

- The process for creating the initial production instance of the tenant is described in "How to Create a New Tenant" on page 85.

Step 1b: Creating the Initial Staging Instance of the Tenant (Version 0.1)

- Move the initial production instance of the tenant (version 0.0) to the staging instance of the tenant. The version on the staging instance of the tenant is renamed version 0.1.

- The process for moving the initial production instance of the tenant to the staging instance of the tenant is described in "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94.

- Timing: This activity typically takes 5 days.

Step 2: Updating the Data on the Staging Instance of the Tenant (Version 0.1)

- Edit the metadata (and optionally configuration data) in version 0.1 on the staging instance of the tenant to meet your requirements.

- Validate the updated configuration on the staging instance of the tenant.

- Timing: This activity typically takes from two to twelve weeks.

Step 3: Pushing the Data from the Staging Instance of the Tenant (Version 0.1) to the Production Instance of the Tenant (Version 1.0)

- Remove any sample and test data that had been temporarily added to the staging instance of the tenant.

- Push the version 0.1 metadata and configuration data from the staging instance of the tenant to the production instance of the tenant, where it becomes the first fully functional release, called version 1.0.

- The process for moving the staging instance of the tenant to the production instance of the tenant is described in "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94.

- Verify that the production instance of the tenant configuration (version 1.0) is ready to "go live."

- Lock the production instance of the tenant, either after you receive the "go live" readiness confirmation or 10 days after the "go live" date if you do not receive any confirmation.

- Timing: This activity typically takes from two to eight weeks.

About Migrating Data After Creating the Initial Production Instance of the Tenant

- "Requirements Before Migrating Data from the Staging or UAT Instance of the Tenant to the Production Instance of the Tenant" below

- "Overview of the Data Migration Process" below

- "Data Migration Process Overview" on the next page

Requirements Before Migrating Data from the Staging or UAT Instance of the Tenant to the Production Instance of the Tenant

Ensure that you have completed the following requirements before you migrate data from the staging or UAT instance of the tenant to the production instance of the tenant:

- Verify that the logs for the staging or UAT instance of the tenant are free of all configuration-related errors. There cannot be any logs for quick actions, workflow, email, and so on, and there cannot be any "object reference not set to an instance of an object" errors.

- Complete a spreadsheet that lists the data to be cleared from the staging instance of the tenant before migrating the data to the production instance of the tenant. This might include details such as "truncate the following tables", "delete … when…", and so on. Update the standard objects spreadsheet.

Overview of the Data Migration Process

"Data Migration Process" below shows the steps that occur during a major update to a functioning production tenant instance (in this example, from version 1.0 to version 2.0).

Data Migration Process
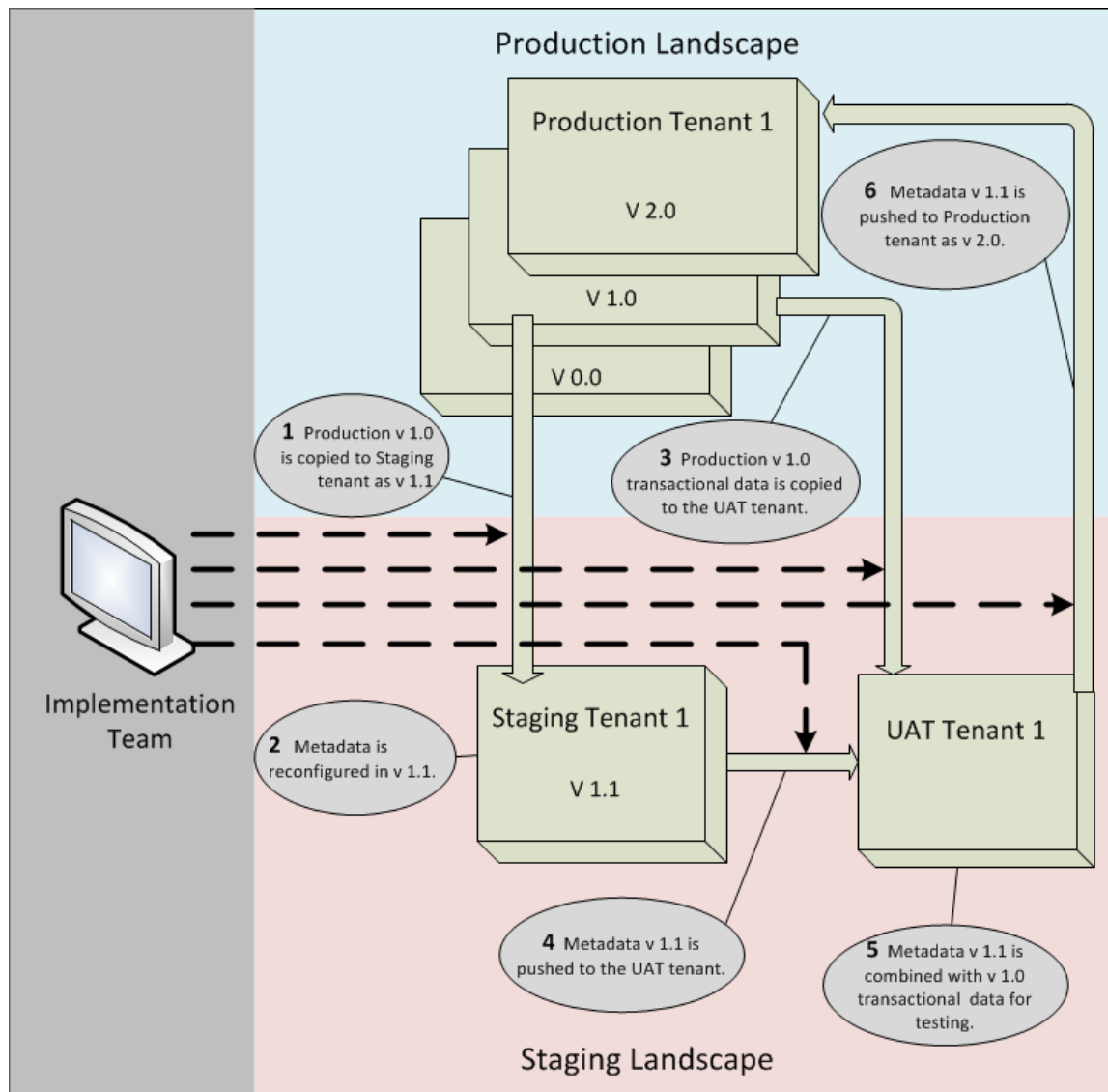
Note the following:

- The version numbers used in "Data Migration Process" on the previous page are for reference only. The software residing on a tenant does not actually have the version numbers shown here.

- Version 1.0 of the production tenant instance represents a functional, running production tenant instance whose configuration remains unchanged until step 6 below.

Data Migration Process Overview

This section describes the actions that occur when data is migrated to the production instance of a tenant.

- "Before Starting the Migration" on the next page

Before Starting the Migration

- Create a comprehensive project plan describing the requested changes, change verification testing, project team contact information, schedule constraints, and other relevant information.

Step 1

- Copy version 1.0 of the production instance of the tenant in its entirety (that is, master data, metadata, configuration data, and transactional data) to the staging instance of the tenant. The copied version on the staging instance of the tenant is renamed version 1.1.

- Disable all email configurations (@customer.mx.domain) for the production instance of the tenant in the staging instance of the tenant, and enable the staging instance of the tenant email configuration (@tenantname-stg.saasit.com).

- Lock the production instance of the tenant. The production instance of the tenant cannot be reconfigured until approved configuration changes are applied in step 6. While the production instance of the tenant is locked, no one can make changes that will affect the migration process or be overwritten by the new configuration.

- Shrink the data on the production instance of the tenant to 5GB or less.

- The staging instance of the tenant is considered current and active.

- Timing: This step usually takes five days.

Step 2

- Edit the metadata (and optionally the configuration data) in version 1.1 to meet your requirements.

- Record the configuration details in a change register.

- Timing: This step typically takes three weeks. It takes two weeks to prepare and migrate the configuration from the staging instance of the tenant to the UAT instance of the tenant, and one week within the UAT instance of the tenant for final testing.

Step 3

- Copy the transactional data (that is, customer data records) from the version 1.0 production instance of the tenant to the UAT instance of the tenant and if necessary shrink the data to less than 5 GB.

- Use the transactional data to test the metadata and configuration data updates in step 5. Version 1.0 metadata and other configuration data from the production instance of the tenant is not copied to the UAT instance of the tenant.

- Timing: This step typically takes three days.

Step 4

- Review and approve the metadata push from the staging instance of the tenant to the UAT instance of the tenant.

- Complete a tenant instance configuration migration template, which provides details about exactly what data should be migrated.

- Review the migration template.

- Push version 1.1 of the metadata and configuration data to the UAT instance of the tenant. Only migrate the items included in the template. This is a full metadata migration from the source to the target; that is, this is not a delta metadata migration.

- Lock the UAT instance of the tenant, making it read-only and under change control.

- Timing: This step typically takes at least one week.

Step 5

- Perform acceptance testing on the UAT instance of the tenant using version 1.1 metadata and configuration data combined with version 1.0 transactional data. Performs tests based on documented use cases to ensure that the configuration is "go-live" ready.

- Verify that the UAT instance of the tenant configuration and logs are "go-live" ready. Do not update the production instance of the tenant if the UAT instance of the tenant is not "go-live" ready based on predefined requirements.

  - All logs on the UAT instance of the tenant must be clear of any configuration-related failures. For example, there should be no logs reporting failures in workflows, quick actions, email, object references not set to instances of objects, and so on.

- We recommend that you create documentation such as a completed migration template, updated clean-up scripts (if necessary), descriptions of all configuration changes and additions, and test cases. You can use this information to verify tenant behavior.

- Approve the configuration push from the UAT instance of the tenant to the production instance of the tenant.

- Timing: Testing on the UAT instance of the tenant typically takes two days.

Step 6

- After you have approved the migration and submitted the clean-up scripts, remove any sample and test data that had been temporarily added to the staging instance of the tenant and then migrated to the UAT instance of the tenant.

- Push the UAT instance of the tenant configuration (that is, the version 1.1 metadata and configuration data) to the production instance of the tenant. This is a full metadata migration from the source to the target; that is, it is not a delta metadata migration.

- Timing: Because this step could result in a brief outage of the production instance of the tenant, we recommend that you schedule it to occur during a maintenance window (usually on a Friday or weekend).

Afterward

- Verify that the production instance of the tenant is go-live ready.

- Place the UAT instance of the tenant in maintenance mode. It is no longer available to you.

- Lock the production instance of the tenant.

- Copy the configuration for the new production instance of the tenant to the staging instance of the tenant while the staging instance of the tenant is still locked.

- Unlock the staging instance of the tenant. From now on, you cannot use the staging instance of the tenant for further configuration without refreshing the production instance of the tenant. Refresh the staging instance of the tenant from the production instance of the tenant periodically to ensure that both tenant instances remain synchronized. You can use the staging instance of the tenant for testing and debugging purposes.

- Timing: This activity typically takes three business days.

If a Step Fails

Step 6: If the metadata and configuration data migration from the UAT instance of the tenant to the production instance of the tenant in step 6 fails:

- Roll back the changes (that is, the backup copy is restored).

- Reinitiate the process of migrating the metadata and configuration data from the staging instance of the tenant to the UAT instance of the tenant.

Items Excluded from Migration

- "Configuration Data (Settings)" below

- "Master Data" on the next page

Configuration Data (Settings)

Configuration data, also called settings, is defined as core configuration attributes that define Ivanti Service Manager.

The only configuration data (settings) that the system migrates when you migrate configuration data (settings), as described in "Updating Configuration Data (Copy Settings)" on page 115, are the following:

- Allowed attachments

- Authentication providers such as LDAP, SAML, OpenId, and Windows Integrated Security

- Bulk upload profiles

- Dynamic styles

- Global constants

- Integration settings

- LDAP configurations

- Object matching lists

- Password policies

- Voice integrations

- Watch list items

- Email configurations (including approval status keywords, email status update mapping settings, ignore message with subject settings, object mappings for email, task assignment status keywords, truncate body settings, and trusted sender domains)

You cannot migrate other types of configuration data (settings) using the Operations Console. To migrate this data, do so manually outside of the Operations Console by using Microsoft SQL commands.

Master Data

You cannot selectively update master data. If you need to update only certain records of master data (for example, only one record in a table), use Microsoft SQL commands to do so outside of the Operations Console.

Starting the Data Migration

- "Starting the Tenant Instance Migration" below

- "Creating the First Staging or UAT Instance of the Tenant from the Production Instance of the Tenant" on page 37

Starting the Tenant Instance Migration

Perform the following steps to start migrating data from the one tenant instance to another. These steps apply for each of the four migration methods. (See "The Four Ways to Migrate Data" on page 21 for information about the different migration methods.)

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the tenant instance to perform the migration for and click **Manage Migration**.

   If the tenant instance is not assigned to a landscape, the system prompts you to assign the tenant instance to a landscape to prepare it for migration. See "Assigning a Tenant to a Landscape" below. (If you do not see this screen, your tenant instance is ready for migration. Go to step 5.)

Assigning a Tenant to a Landscape

For example, if you click **Manage Migration** in the screen shown in "Database Pattern Screen 1" below, the system displays the screen in "Database Pattern Screen 2" on the next page.

Database Pattern Screen 1

Database Pattern Screen 2

These screens are based on the landscape information, shown in "Database Pattern Screen 3" below.

Database Pattern Screen 3

There is only one landscape with the pattern **frs-dev-{tenantName}.saasitdev.com** and that is the production landscape. Therefore, on the bottom of "Database Pattern Screen 2" above, you can assign that tenant instance to that landscape. In the middle part of "Database Pattern Screen 2" above are the other available landscapes but these do not have the same pattern.

3. Select a landscape from the **Landscape** drop-down list.

If you do not want to assign it to any of the landscapes in the **Landscape** drop-down list, you can click **Cancel** and go to the **Edit Landscape** screen to edit the landscape database pattern. See "Editing a Landscape" on page 77.

4. Click **Assign**. The system displays the **Migration Configuration** screen, showing all of the tenants and tenant instances and available migration paths in the landscape. See "Migration Dashboard" below.

Migration Dashboard
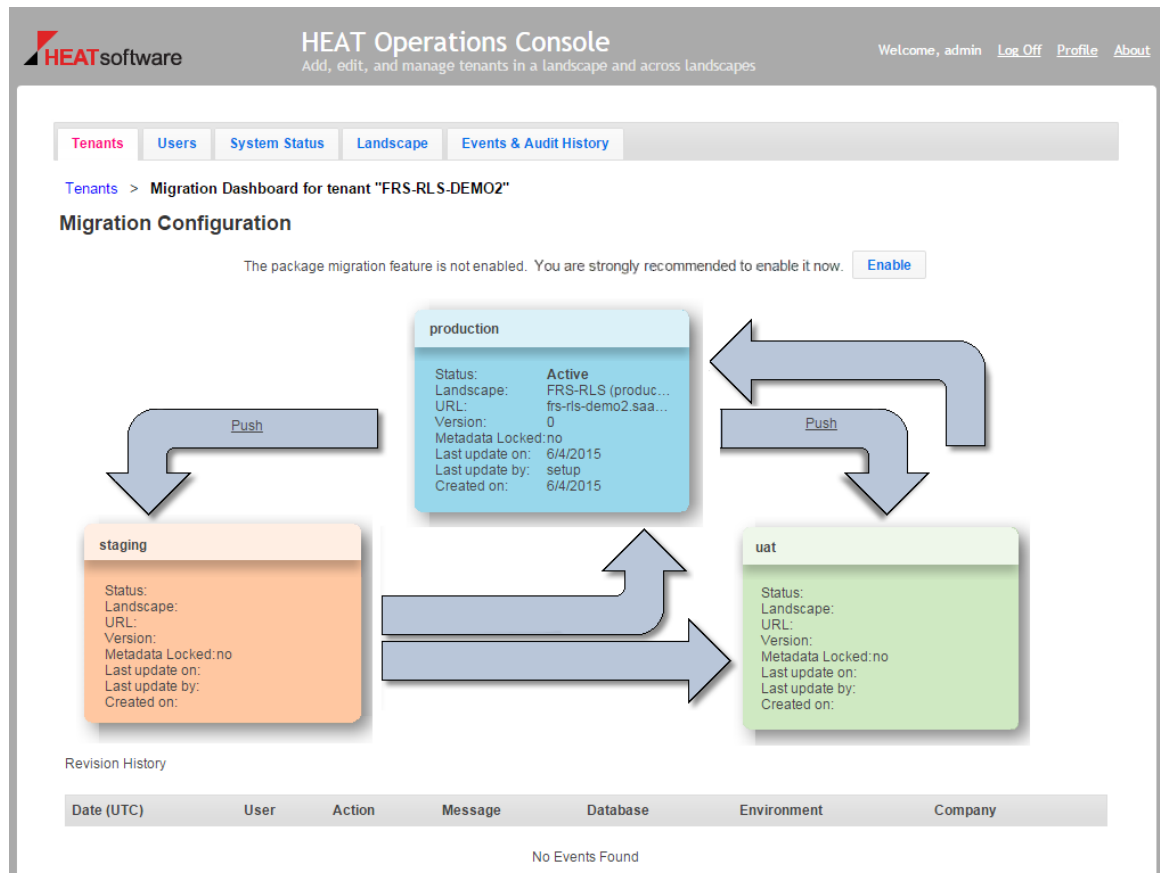
Things to note in "Migration Dashboard" above:

•   The display box for each tenant instance (production, staging, and UAT in this example) contains status information that is updated whenever a tenant instance is updated.

•   Arrows show all possible migration paths. If an arrow contains a **Push** or **Apply Package** link, the path is available to use based on the current status of all tenant instances in the landscape. Depending on the status of the tenant instance, some migrations paths might not be available.

•   A migration history log is displayed at the bottom of the migration dashboard. If no migrations have taken place yet, "No Events Found" displays there as shown in "Migration Dashboard" above.

- To enable the Ivanti Service Manager development project to migrate data, click **Enable**. See "Using the Ivanti Service Manager Development Project to Migrate Data" on page 41.

5. To start the migration, click either **Push** or **Apply Package** in the arrow representing the migration path to use. The system displays the **Copy *source_tenant_instance* to *target_tenant_instance*** (for example, **Copy Production to Staging**) dialog box.

6. The remaining steps in the migration procedure differ depending on which migration method you use and whether this is the first migration for this migration path. If this is the first migration, the system *creates* the first configuration on the target tenant instance. If a migration has already been performed and this is a subsequent migration, the result is an *update* of the configuration on the target tenant instance.

    - If this is the first migration, go to "Creating the First Staging or UAT Instance of the Tenant from the Production Instance of the Tenant" below. Follow this process for each of the four migration methods. (See "The Four Ways to Migrate Data" on page 21 for information about the different migration methods.)

    - If this is *not* the first migration and you are enabling the Ivanti Service Manager development project and either using the push link or applying a package, go to "Using the Ivanti Service Manager Development Project to Migrate Data" on page 41.

    - If this is *not* the first migration and you are disabling the Ivanti Service Manager development project and using either Simple Mode or Advanced Mode, go to "Starting the Data Migration" on page 34. *We do not recommend disabling the Ivanti Service Manager development project and using either Simple Mode or Advanced Mode.*

Creating the First Staging or UAT Instance of the Tenant from the Production Instance of the Tenant

This process applies for each of the four migration methods. (See "The Four Ways to Migrate Data" on page 21 for information about the different migration methods.)

At the beginning of the migration process, you migrate the data from the production instance of the tenant to the staging instance of the tenant. When you migrate the data from the staging instance of the tenant to the UAT instance of the tenant, you must also migrate data from the production instance of the tenant to the UAT instance of the tenant. That way, the UAT instance of the tenant has the latest data from both the staging and production instances of the tenant.

Follow these steps to create the first staging or UAT instance of the tenant from the production instance of the tenant. You also follow these same steps if you want to refresh a staging or UAT instance of the tenant from the production instance of the tenant.

Perform steps 1 to 4 in "Starting the Tenant Instance Migration" on page 34 before you perform the steps in this section.

When you migrate data from the production instance of the tenant to either the staging or UAT instance of the tenant for the first time, the system creates the first configuration on the staging or UAT instance of the tenant and you see a **Copy Production to Staging** (or **Copy Production to UAT**) dialog box similar to "Copy Dialog Box (Creating a Tenant Instance Configuration from the Production Tenant Instance)" below.

Copy Dialog Box (Creating a Tenant Instance Configuration from the Production Tenant Instance)

1. Enter the following information in the **Copy** dialog box:

> You cannot enter data for the following:
>
> In the **Operation** field, **Create** is the only option available. (This is because the system *creates* the first populated iteration of the target tenant instance.)
>
> For the **Target Landscape**, the system automatically populates this field based on where the **Push** arrow is pointing. For example, in "Migration Dashboard" on page 36, if you click **Push** in the arrow between the production instance of the tenant and the staging instance of the tenant, the system automatically enters the staging tenant instance into this field.

- Select a backup option in the **Target DB Option** field. Because this is the first time that you are creating the staging or UAT instance of the tenant, there is no backup, so you must select to use a live backup. See "Backup Types" on page 11 for information about the options.

- In the **Target DB Server** field, if the target landscape contains more than one database server, select the server that hosts the Configuration Database.

- Check **Keep existing users enabled in the target tenant** to keep all the users, and not just the administrators, active.

- Check **Delete audit, log, change log and history data** to delete the data in these tables:

    - Logs

    - Frs_ops_logon_history

    - Frs_ops_gbu_pipeline_history

    - Frs_ops_metadata_history

    - Frs_ops_upload_cache

    - Journal

    - LDAPImportLog

    - Frs_ops_package_item

    - Frs_ops_change_log

    - Frs_ops_change_log_txid

    - Frs_ops_project except for the "default" project

    - Frs_ops_package

    - All audit tables, with the name Audit_*table*

- Check **Scrub email address in users** to ensure that you do not accidentally email users repeatedly during configuration testing on the target tenant instance. If you check this option, the system changes all email fields containing customer email to use the @it.com domain instead of the company domain. This change applies to the user's primary email, contact email, manager email, and team email.

- Check **Scrub email address in other places** to scrub email accounts saved in transactional data (such as in incidents, service requests, workflow tasks, and so on).

- Check **Trim data to keep last 90 days only** to only copy transactional data that was created or modified within the last 90 days. If you are updating the database from a copy of your existing customer database, most records are likely to be current within 90 days.

Do not check this option if you are creating the database from the sample ("out of the box") database, because the records are likely to be older than 90 days in the sample database.

- Check **Shrink database after scrubbing and trimming** to truncate unnecessary data when the tenant instance is created from a backup. This is performed as a background task. If the database is very large, this task can take a long time or could abort. The Operations Console does not report any status on this task. If your database is very large, we recommend that you consider shrinking it outside of the Operations Console instead of in this procedure.

2. Click **Execute**. In the confirmation dialog box, click **OK**.

3. If there are any error or warning messages, the system displays them. Click **OK**.

   The system migrates the data from the production instance of the tenant to the staging or UAT instance of the tenant, and configures the staging or UAT instance of the tenant for the first time.

Using the Ivanti Service Manager Development Project to Migrate Data

ⓘ This is the recommended method for migrating data. See "The Four Ways to Migrate Data" on page 21 for a list of the ways to migrate data.

For information about using the Ivanti Service Manager development project with packages that have been downloaded from the App Store, see "Appendix D: Working with Packages from the App Store" on page 142.

- "About Using the Ivanti Service Manager Development Project to Migrate Data" below

- "Cautions About the Ivanti Service Manager Development Project" on the next page

- "About the Ivanti Service Manager Development Project" on the next page

- "Using the Ivanti Service Manager Development Project to Create a Data Package" on page 44

- "Using the Ivanti Service Manager Development Project to Migrate a Data Package" on page 44

- "About Metadata Read-Only Mode" on page 46

- "Example: Migrating Request Offering Changes By Enabling the Ivanti Service Manager Development Project" on page 46

About Using the Ivanti Service Manager Development Project to Migrate Data

Use the Ivanti Service Manager development project in the Ivanti Service Manager Configuration Console (not within the Operations Console) to create packages to migrate data from one instance of a tenant (such as staging) to another instance of a tenant (such as UAT). In this way, you can pick and choose which data to migrate instead of migrating the entire system. The Ivanti Service Manager development project allows Ivanti Service Manager to maintain a history of changes in metadata, validation data, configuration settings, and a few specific master data business objects. It also allows you to package a unit of integral changes into a package for migration to another tenant instance.

Cautions About the Ivanti Service Manager Development Project

- The Ivanti Service Manager development project gives you the ability to update the production instance of your tenant with configuration changes made in the staging instance of the tenant. If these changes conflict in some way with your Ivanti Service Manager system, they cannot be undone.

- We recommend that you enable and use the Ivanti Service Manager development project to migrate data, and that you do not switch between enabling and disabling the Ivanti Service Manager development project. See "The Four Ways to Migrate Data" on page 21 for a list and description of the four different ways that you can migrate data.

- We recommend that you plan and prepare for moving a package as described in this section. We strongly encourage you to make a complete backup of the target tenant instance and to create a procedure for restoring the backup if need be:

  - If there is a problem with the UAT instance of your tenant, you can restore it from the production instance of your tenant using the Operations Console, by refreshing the UAT instance of your tenant.

  - If there is a problem with the production instance of your tenant, you need to restore the production instance of the tenant from the backup as quickly as possible using Microsoft SQL tools, as any transactions that are created in the production instance of the tenant after a backup and before a restore will be lost.

- Before you enable and use the Ivanti Service Manager development project, check with Support by visiting https://support.ivanti.com .

About the Ivanti Service Manager Development Project

- "Limitations" below

- "Terminology" on the next page

- "About Using Your Named Account to Log In" on the next page

Limitations

- The Ivanti Service Manager development project has not yet provided a way to validate package dependency. You need to ensure that if you make a change that depends on another change, you migrate both changes.

- Before handing out a package for migration, determine whether the package is complete and self-contained. The package should contain all dependent changes. Not having all the required changes in the package may cause the system to fail when you migrate the package and you may have to restore the target system from the back-up system.

- You cannot use the Ivanti Service Manager development project and Simple Mode or Advanced Mode at the same time. (See "The Four Ways to Migrate Data" on page 21 for a list and description of the four different ways that you can migrate data.) To use Simple Mode or Advanced Mode, you must disable the Ivanti Service Manager development project.

- You cannot use the Ivanti Service Manager development project to migrate certain types of data. For a list of the data that cannot be migrated, see "Items Excluded from Migration" on page 32.

Terminology

The Ivanti Service Manager development project uses the following terms:

- **Transaction**: An individual change to the database.

- **Transaction set**: A group of transactions.

- **Project**: A group of transaction sets.

- **Package**: Contains the transaction sets from multiple projects.

- **Aspect**: Information about a feature, a component, a patch, or even a configuration. An aspect presence indicates the presence of the feature.

About Using Your Named Account to Log In

The first time that you log into Ivanti Service Manager as an administrator, the system displays a message stating that you are logging in as an administrator. This is not suitable for making configuration changes. If you make configuration changes while logged in as an administrator, you may not be able to package the changes properly.

We recommend that you no longer use a generic administrator account when you are in the Configuration Console. With the Ivanti Service Manager development project, the Ivanti Service Manager system tracks all the changes made and attributes them to the user who made the changes. If all users log in using the generic administrator account, there is no way to know who made the changes.

The system displays a message stating that the Configuration Console is in production metadata read-only mode. If you see this message, click **OK** and then log out as an administrator and log back in using your named account. See "About Metadata Read-Only Mode" on page 46.

Using the Ivanti Service Manager Development Project to Create a Data Package

See "Appendix C: Using the ISM Development Project to Migrate Data" on page 128 for complete information about how to use the Ivanti Service Manager Configuration Console to create packages, which can be used when migrating data. You perform those tasks in the Ivanti Service Manager Configuration Console and not in the Operations Console. This is the same information that is found in the Ivanti Service Manager online help and is only copied here for your convenience.

Using the Ivanti Service Manager Development Project to Migrate a Data Package

- "Enabling the Ivanti Service Manager Development Project" below

- "Migrating the Package" on the next page

- "Checking the Data Migration" on the next page

- "Disabling the Ivanti Service Manager Development Project" on the next page

Enabling the Ivanti Service Manager Development Project

To enable the Ivanti Service Manager development project, follow these steps:

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the instance of the tenant for which you will perform the migration and click **Manage Migration**.

3. If the tenant is not part of a landscape, the system prompts you to add the tenant to a landscape to prepare it for migration. Click **Proceed**. The migration dashboard opens showing all of the tenants and tenant instances and available migration paths in the landscape.

4. To enable the Ivanti Service Manager development project, click **Enable**.

5. Click **OK** at the confirmation prompt.

    The migration dashboard looks like this:

For customers who have been using HEAT Service Management since Release 2014.2 or earlier, you may not see the **Apply Package** link, even though you have enabled the Ivanti Service Manager development project.

If you do not see the **Apply Package** link, refresh the staging instance of the tenant by following the process in "Creating the First Staging or UAT Instance of the Tenant from the Production Instance of the Tenant" on page 37.

Migrating the Package

> ⓘ To migrate a package from the staging instance of a tenant to the production instance of a tenant, the package must be closed. You cannot move open packages to the production instance of a tenant. The package does not have to be closed, however, if you are migrating from the staging instance of the tenant to the UAT instance of the tenant.
>
> The system displays an informational message if you migrate the same package to the same tenant instance more than once.

To migrate a data package, do the following:

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the instance of the tenant for which you will perform the migration and click **Manage Migration**.

3. To migrate the data package, click **Apply Package** in the corresponding arrow. The system displays the **Applying Package To** page.

4. Enter selections as follows:

   a. Select the name of the source package from the drop-down menu.

   b. We highly recommend that you do not check **Skip Target Tenant Backup**.

   c. Select **Apply without validation** to apply the package without first validating it.

5. Click **Execute** to migrate the package or click **Preview** to create an XML file with changes that you can review before you migrate the data.

6. Click **OK** at the confirmation prompt.

   When the migration is complete, the system displays a message that the package has been applied successfully. When you refresh the staging or UAT instances of the tenant, all transaction sets are cleared.

Checking the Data Migration

To check that the data migrated successfully, log in to the UAT instance of the tenant and search for the changes.

1. Log in to the Ivanti Service Manager system for the UAT instance of the tenant. The system displays a message stating that the UAT instance of the tenant is locked.

2. Search for the changes.

Disabling the Ivanti Service Manager Development Project

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the instance of the tenant for which you will perform the migration and click **Manage Migration**.

3. If the tenant is not part of a landscape, the system prompts you to add the tenant to a landscape to prepare it for migration. Click **Proceed**. The migration dashboard opens showing all of the tenants and tenant instances and available migration paths in the landscape.

4. To disable the Ivanti Service Manager development project, click **Disable**.

About Metadata Read-Only Mode

When you use the Operations Console to migrate data, the production and UAT instances of the tenant are always in metadata read-only mode, which is equivalent to being locked. When the system is locked, you cannot make changes to most types of metadata in the Configuration Console. (You can only make changes to the following four items: saved searches, reports, quick actions, and dashboards.) While migrating data, the system temporarily unlocks the metadata on the target system.

You cannot import high-impact packages if your system is in metadata read-only mode.

To put a system into metadata read-only mode, do the following:

1. Log in to the Operations Console.

2. Click **Tenant**. The system displays the **Tenants** workspace.

3. Find the tenant to edit and click **Edit Properties**.

4. Check **Metadata Locked**.

5. Click **Save**.

Example: Migrating Request Offering Changes By Enabling the Ivanti Service Manager Development Project

This example is an end-to-end example of migrating request offering changes by enabling using the Ivanti Service Manager development project. It describes how to make a change to a request offering in the staging instance of a tenant and then migrate the change to the production instance of the tenant. In this example, the Ivanti Service Manager development project is enabled and data is migrated by both applying a package and by using the push link.

This example is copied directly from the Ivanti Service Manager online help.

- "About this Example" on the next page

- "Before You Begin" on the next page

- "Packaging the Changes Using Ivanti Service Manager" on the next page

- "Migrating the Data from the Staging Instance to the UAT Instance" on page 48

- "Making Changes to the Request Offering in the Configuration Console" on page 49

- "Testing the Change" on page 49

- "Migrating the Data from the Staging Instance to the Production Instance" on page 49

- "Migrating the Service Request Data from the UAT Instance to the Production Instance" on page 50

About this Example

Ivanti Service Manager does not have a system for tracking request offerings, so to migrate a request offering from one environment to another, you must create and apply a package *and* also do a master data push using the Operations Console. Creating and applying a package alone does not migrate the request offering change.

Before You Begin

Before you begin, use the Operations Console to refresh the staging instance of the tenant from the production instance of the tenant. This ensures that you have the same baseline in both environments before you make the change.

Packaging the Changes Using Ivanti Service Manager

When you make changes to request offerings, Ivanti Service Manager tracks the changes made to the request offering and associated workflows and quick actions, but does not track the changes made to some related content. Ivanti Service Manager does not record changes for the organizational unit, configuration item service, location, or request offering links to organizational units, configuration item services, or locations. This can cause confusion when you move changes from one system to another.

The changes that the system records may differ depending on if you are logged in as an administrator or as another role when you make the changes.

- If you make the changes as a service owner, you cannot select the project to apply the changes to. Therefore, all changes are added to the default project.

- If you make the changes as an administrator, you can select the project to apply the changes to. Before you make the changes to the request offering, we recommend that you create a new project and apply the changes to that project. See "Creating a Project" on page 129.

1. (Optional. For administrators only.) Go to the Configuration Console, create a project, and then select it.

2. Make the changes to the request offering. The system tracks the changes to the workflows and quick actions associated with the request offering.

3. Create a package with the transaction sets associated with the changes that you made to the request offering. Ensure that the package contains all of the changes made to the system since you made the baseline, as described in "Before You Begin" on the previous page, regardless of if they are related to your project.

Migrating the Data from the Staging Instance to the UAT Instance

1. Log into the Operations Console.

2. Refresh the UAT instance of the tenant from the production instance of the tenant.

3. From the **Tenants** tab, find the instance to migrate and click **Manage Migration**.

4. At the top of the page, click **Enable** to enable the Ivanti Service Manager development project. At the confirmation message, click **OK**.

5. In the arrow going from the staging instance of the tenant to the UAT instance of the tenant, click **Apply Package**.

6. Do the following:

   • In the **Source Package** field, select the package to migrate.

   • Check **Skip Target Tenant Backup**.

   • Select **Apply without validation**. This allows you to apply the package even if there are errors. You may need to fix the errors after the change is applied, if you do not want to restore the database and fix the root cause.

   • Click **Execute**.

     The system migrates the package.

7. In the arrow going from the staging instance of the tenant to the UAT instance of the tenant, click **Push**.

8. Do the following:

   • In the **Operation** field, select **Copy Master Data Only**.

   • Check **Delete data items in uat that do not exist in staging**.

   • Check **Service request**.

   • Click **Execute**.

   The system migrates all of the service requests.

Making Changes to the Request Offering in the Configuration Console

If the target tenant instance does not have an equivalent request offering, (for example, if you are importing a package from the App Store or a package from another system), you must manually link the configuration item service, organizational unit, and location.

1. Log into the Service Desk Console on the UAT instance of the tenant.

2. Open the **Request Offering** workspace.

3. Find and open the existing request offering.

4. On the **1. Define request offering** page, for the **Service** field, select the configuration item service associated with this request offering. Note that in many cases, the configuration item service is represented by a GUID and not by its name in the drop-down list. Be sure to select the correct configuration item service.

5. Click **Next** twice until you reach the **4. Publish Action Access** page.

6. Click **Add** to add a new organizational unit and location. The system displays a new line.

7. Click **Any** under the **Team** header and select the organizational unit associated with this request offering. Click **OK**. Note that in many cases, the organizational unit is represented by a GUID and not by its name in the drop-down list. Be sure to select the correct organizational unit.

8. Click **Any** under the **Location** header and select the location associated with this request offering. Click **OK**. Note that in many cases, the location is represented by a GUID and not by its name in the drop-down list. Be sure to select the correct location.

9. Click **Save & Exit**.

Testing the Change

1. Log into the UAT instance of the tenant and test the changes.

2. If anything is wrong, perform the procedures in both "Packaging the Changes Using Ivanti Service Manager" on page 47 and "Migrating the Data from the Staging Instance to the UAT Instance" on the previous page.

3. When you are finished testing the change and are satisfied with the results, go back to the staging instance of the tenant and close the package. This ensures that no further changes in the system are added to the package.

Migrating the Data from the Staging Instance to the Production Instance

1. Log into the Operations Console.

2. From the **Tenants** tab, find the instance to migrate and click **Manage Migration**.

3. At the top of the page, click **Enable** to enable the Ivanti Service Manager development project. At the confirmation message, click **OK**.

4. In the arrow going from the staging instance of the tenant to the production instance of the tenant, click **Apply Package**.

5. Do the following:

   • In the **Source Package** field, select the package to migrate.

   • Ensure that **Skip Target Tenant Backup** is not checked.

   • Select **Apply without validation**. This allows you to apply the package even if there are errors. You may need to fix the errors after the change is applied, if you do not want to restore the database and fix the root cause.

   • Click **Execute**.

   The system migrates the package.

Migrating the Service Request Data from the UAT Instance to the Production Instance

You must publish the service request master data from the UAT instance of the tenant to the production instance of the tenant because the UAT instance has the tested service request data. (If you are sure that no changes were made to the staging instance of the tenant, you can push the service request data from there to the production instance of the tenant instead.)

1. Log into the Operations Console.

2. From the **Tenants** tab, find the instance to migrate and click **Manage Migration**.

3. In the arrow going from the UAT instance of the tenant to the production instance of the tenant, click **Push**.

> If there is no **Push** link in the arrow, you must push the data from the staging instance to the UAT instance. In the **Operation** field, select **No Op (Increase version only)** and click **Execute**.

4. Do the following:

   • In the **Operation** field, select **Copy Master Data Only**.

   • In the **Additional Data** section, select any or all of **Service (CI.Service)**, **Organizational Unit (OrganizationalUnit)**, and **Location (Location)**.

   • Check **Skip target tenant backup before operation**.

   • Check **Delete data items in production that do not exist in uat**.

   • Click **Execute**.

   The system migrates all of the service requests.

Troubleshooting and Known Issues

- "Debugging the Migration" below

- "Known Issues in the Operations Console" on page 54
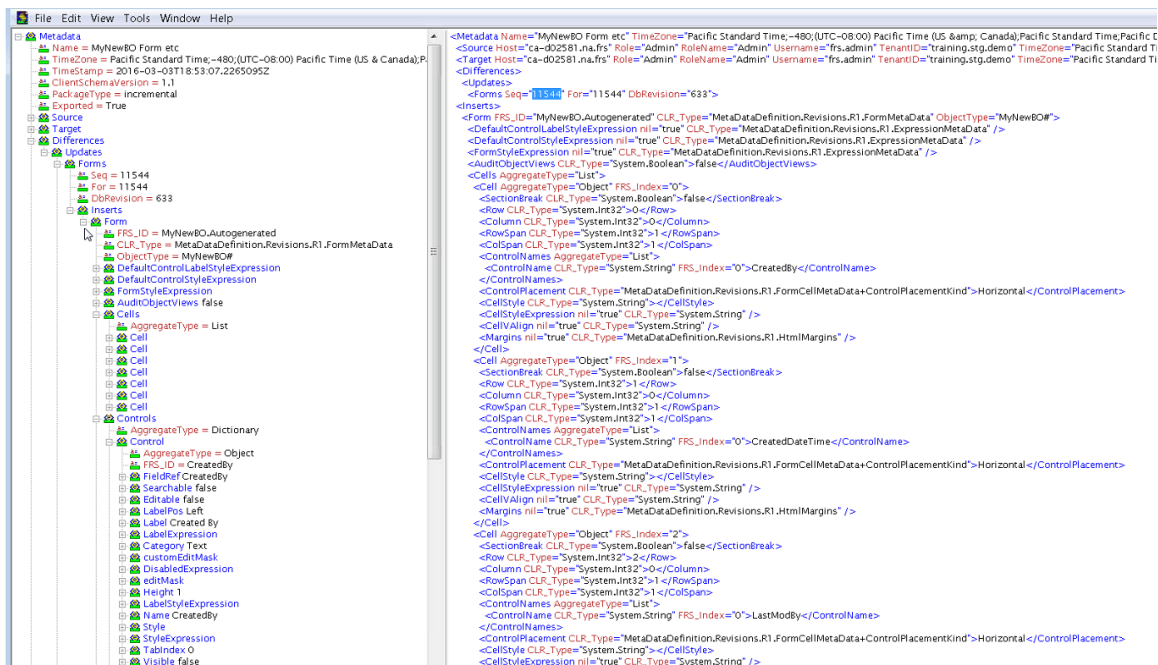
Debugging the Migration

There are several ways to debug the data migration:

- "Previewing the Package Before Migrating It" below

- "Viewing the Warning and Error Messages" on the next page

- "Reviewing the Package Content" on page 53

- "Viewing the Operations Console Log Files" on page 53

- "Running Packages in the Correct Order" on page 53

- "Checking the API Keys" on page 53

Previewing the Package Before Migrating It

If you used the Ivanti Service Manager development project, as described in "Appendix C: Using the ISM Development Project to Migrate Data" on page 128, you can preview the changes to ensure that they are correct. On the **Applying Package To** screen, click **Preview**. The system downloads an XML file with the proposed changes. Open the file in an XML editor and ensure that the changes are correct. The system associates all warnings to a change record by using a unique sequence number.

Example of XML File with Proposed Changes

Viewing the Warning and Error Messages

After you perform the first migration from the production instance of the tenant to the staging instance of the tenant, review the error messages. You can review them in any of the following places:

- The **Event and Audit History** tab in the Operations Console. See "Viewing Events and Audit History" on page 70.

- The patch log in the Ivanti Service Manager configuration database (ConfigDB).

- The patch log in the Ivanti Service Manager target tenant instance. See the Ivanti Service Manager online help for information.

- If you used the Ivanti Service Manager development project, as described in "Appendix C: Using the ISM Development Project to Migrate Data" on page 128, the system displays errors and warnings on the bottom of the **Applying Package To** screen.

Sample Errors for a Package

Reviewing the Package Content

Review the package content:

- This is located in a file called *source-tenant-timestamp*.MetadataPatch.zip which resides in the c:\Temp\OpsConsoleDiff directory on the server on which the Operations Console is installed. This package file lists the contents of the migration.

- Check the **Transaction Details** workspace in Ivanti Service Manager. This workspace lists the details of the package migrations.

Viewing the Operations Console Log Files

Find and open the log files from the source tenant instance and the target tenant instance and compare the metadata.

Running Packages in the Correct Order

If you used the Ivanti Service Manager development project, as described in "Appendix C: Using the ISM Development Project to Migrate Data" on page 128, and your migration contains multiple packages, ensure that you migrate the packages in the correct order. If items must be migrated in a specific order and you do not follow that order, the system may display an error. To avoid this, we recommend creating a master package that contains the incremental packages in the correct order.

Checking the API Keys

Check the API keys. Each API key must be unique and cannot be shared between tenant instances. See "About System Information" on page 67.

Known Issues in the Operations Console

This section contains information about issues and limitations in the Operations Console.

Copying Child Business Objects

Symptom

When copying data from the staging or UAT instance of a tenant, selecting a business object does not copy the data of its child business objects.

Affected Release

All releases.

Resolution

To copy the data for a child business object, you cannot simply select its parent business object. You must select the child business object.

For example, to copy all of the knowledge articles in the **QandA** category, you cannot select only the **FRS_Knowledge** business object. You must select the child business object, which in this case is **FRS_Knowledge.QandA**.

The configuration items are listed in the drop-down list by their display names. See below for an example.

Configuration Items

Migrating Logos

Symptom

When migrating data that includes a logo, the updated logo does not migrate.

Affected Release

All releases.

Resolution

The Operations Console does not touch the logo during data migration. It does not remove the logo, but it also does not update the logo. Therefore, if you change the logo and then migrate data, after the data migration you must manually upload the new logo.

Also, changing the logo is not something that gets tracked in Ivanti Service Manager. Therefore, you cannot use the package process for this item.

Five Minute Timeout Causes Problems When Migrating Data in a Load-Balancing Deployment

Symptom

When migrating data on a system that uses a load-balancing deployment, the system is usually set to time out after 5 minutes for security reasons, but often it takes more than 5 minutes to migrate data. This causes the data migration to fail and the system displays an error message stating "Unable to retrieve metadata from *tenant_name* due to the underlying connection was closed."

Affected Release

All releases.

Resolution

You can configure the Operations Console to go directly to the Ivanti Service Manager application server without going through the load balancer. To do this, go to the file called C:\Windows\System32\drivers\etc\hosts and add the host name for the multi-instance service URL. You can also use another port number for the multi-instance service URL in the Ivanti Service Manager application server. See "Editing a Landscape" on page 77 for more information about the multi-instance service URL.

You can also change the timeout setting in your network, but we do not recommend doing this.

Operations Console Does Not Work with Microsoft Internet Explorer Releases 8 and 9

Symptom

The Operations Console does not work with Microsoft Internet Explorer Release 8 or Release 9 because they do not support the latest third-party Javascript code.

Affected Release

All releases.

Resolution

Upgrade to a newer release of Microsoft Internet Explorer or to another browser.

Data Not Showing in Microsoft Internet Explorer Release 11

Symptom

The Operations Console does not work in Microsoft Internet Explorer Release 11 due to a new security feature in Microsoft Internet Explorer Release 11.

Affected Release

All releases.

Resolution

Add the URL of the Operations Console to the trusted sites list in Microsoft Internet Explorer Release 11. Follow these steps:

1. Go to **Tools > Internet Options > Security**.

2. Highlight **Trusted sites** and click **Sites**.

3. Click **Add**.

There is also a known issue with Microsoft Internet Explorer Release 11 and the French operating system. If you run into this problem when using the French operating system, clear the Microsoft Internet Explorer browser cache.

There is No Push Link

Symptom

There are three scenarios where this happens:

1. There is no **Push** link between the staging instance of the tenant and the production instance of the tenant if the production instance of the tenant is not at version 0.

2. There is no **Push** link between the UAT and production instances of the tenant after migrating data from UAT to production.

3. There is no **Push** link between the UAT and production instances of the tenant after you apply a closed package from the staging instance of the tenant to the production instance of the tenant.

You can only migrate data from one instance to another if there is a different version number. The system automatically increases the version number after a successful migration.

In a three-tier landscape group, you can only migrate data from staging to production when the production instance of the tenant is at version 0.

> With a two-tier landscape group, the migration path is only between two tenant instances, usually the production instance of the tenant and the staging instance of the tenant. With a three-tier landscape group, there are migration paths between three tenant instances, usually the production instance of the tenant, the staging instance of the tenant, and the UAT instance of the tenant.
>
> We recommend always using a three-tier landscape group unless you have a demo environment.

Affected Release

All releases.

Resolution

For the first scenario, we recommend temporarily switching from a three-tier landscape group to a two-tier landscape group to immediately migrate data between the staging instance of the tenant and the production instance of the tenant. See "Editing a Landscape Group" on page 74 for information about doing this. After the migration, switch back to a three-tier landscape group.

For the second and third scenarios, you must increase the version of the UAT instance of the tenant. To do this, click **Push** and in the **Operation** field, select **No Op (Increase version only)**. This increases the version of the UAT instance of the tenant so that the **Push** link appears. See "Increasing the Version Number (No Op [Increase Version Only])" on page 118.

Cannot Migrate Data if a Business Object Contains More than 100,000 Records

Symptom

The Operations Console cannot migrate database tables with more than 100,000 records. In some deployments, the **ServiceReqParam** table can have more than 100,000 records.

Affected Release

All releases.

Resolution

We have removed the **ServiceReqParam** tables so they are no longer part of the service request copy in the Operations Console.

We recommend as a best practice that your database tables be smaller than 100,000 records. We also recommend that if you do have a table with more than 100,000 records, you do not migrate it using the Operations Console.

If you do have a table with more than 100,000 records that needs to be migrated, use Microsoft SQL Server commands to manually copy the data and its relationships, or break the table into smaller tables, migrate the smaller tables, and then reform the data into a larger table after the migration.

Wrong Session Key Format Error

Symptom

When migrating data with the Operations Console, the system gives a "Wrong Session Key Format" error.

Affected Release

All releases.

Resolution

The system gives this error when you copy the database from another server and restore it for a tenant without using the push action in the Operations Console to set up the tenant. To solve this, migrate the data using the Operations Console using the copy to replace with database already restored option.

You can also use Microsoft SQL Server to truncate the table using this command: **truncate table Frs_ ops_session_secondary_key;**.

Cannot Validate or Duplicate the API Key

Symptom

When migrating data or adding a new tenant with the Operations Console, the system gives an error message about being unable to validate the API key.

Affected Release

All releases.

Resolution

Check that all active tenants in the Ivanti Service Manager configuration database have a valid database and proper login credentials.

Make sure that none of the tenants has duplicate API keys by logging into the Operations Console, clicking the **System Status** tab, and looking for duplicate API key entries. If you see a duplicate entry, click **Fix** to fix the issue. See "Fixing Problems" on page 69 for more information.

Report Template and Schedule Issues

Symptom

The Operations Console does not migrate out-of-the-box report templates (.rdl files) if the source and target tenants are on different servers.

The Operations Console does not migrate custom-created report templates (.rdl files) even if the source and target tenants are on the same server. (These report templates are stored in the Report Server, which is specific to the tenant.)

The Operations Console does not migrate the report schedules, report subscriptions, or report distributions for migrated reports.

The system does not provision a report when its report template is not out-of-the-box (that is, if it is a custom report template), or if the report template has not been migrated.

Affected Release

All releases.

Resolution

The Operations Console does migrate new report definitions, regardless of if there is a template or not.

The Operations Console does migrate updated report definitions. You can overwrite the target if the source definition is older than the target definition.

You can delete report definitions in the target tenant if it does not exist in the source tenant.

The system provisions the new report only if the report template is out-of-the-box, or if the report template exists for the tenant on the Report Server.

To perform all other report migration and provisioning tasks, use the Ivanti Service Manager reporting service.

Some Dashboard and Saved Search Properties are Not Migrated

Symptom

Some properties (such as "Set as default for myself", "Default for ...", and "My default") for dashboards and saved searches are not migrated.

Affected Release

All releases.

Resolution

These "default" properties are not metadata and are therefore not copied.

Duplicate Data or Database Constraint Errors

Symptom

When you migrate data, the system gives this error:

> Error during metadata commit operation: DataLayer.ConstraintViolationException: DB constraint violation exception: Violation of PRIMARY KEY constraint .....

> Error during metadata commit operation: DataLayer.ConstraintViolationException: DB constraint violation exception: Violation of UNIQUE KEY constraint .....

Affected Release

All releases.

Resolution

This happens when you add or edit data in both the source and target instance of the tenant. When you use the Operations Console to migrate data, you cannot add or edit data on both the source instance of the tenant and the target instance of the tenant.

If you try to migrate data after adding or editing data on both the source instance of the tenant and the target instance of the tenant, the system duplicates the records. You must ignore the error message and fix the errors manually.

Problems Creating Packages

Symptom

When creating packages, it can be difficult to create the correct package for the migration. It can be challenging to find the right pieces of metadata to package and to find all of the dependencies.

Affected Release

All releases.

Resolution

- Before making any changes, always add a Ivanti Service Manager development project.

- Before making any changes, select a project. It is easier for you to include all changes in a project to a package.

- Review the transaction details to find and review transaction sets. You can search based on the object type, business object, entity name, and the name of the person who made the change. You can then add the transaction sets to the package.

- You may need to do a few iterations before you get the right package.

No Tenants in the Operations Console After Upgrading

Symptom

After upgrading Ivanti Service Manager, the system updated the connection strings for the landscape with the incorrect name for the data source.

Affected Release

All releases.

Resolution

In the Operations Console, edit each landscape to update the data source to the correct database server name. See "Editing a Landscape" on page 77 for more information on how to do this.

Appendix A: Using the Operations Console to Perform Administrative Tasks

- "Working With Operations Console Users" below

- "Viewing System Information" on page 67

- "Viewing Events and Audit History" on page 70

- "Reviewing License Usage" on page 71

- "Configuring Landscapes" on page 71

- "Working with Tenants and Tenant Instances" on page 82

Working With Operations Console Users

- "About Operations Console Users" below

- "About the Users Workspace" on the next page

- "Adding a New Operations Console User" on page 65

- "Editing a Operations Console User" on page 66

- "Deleting a Operations Console User" on page 66

- "Changing Your User Display Name" on page 66

- "Changing Your Password" on page 66

About Operations Console Users

Use the **Users** workspace to add and edit Operations Console users. These are users who can log in to and use the Operations Console; these are not tenant users. Your environment may have just one admin login ID, which is used by multiple users, or it may have multiple login IDs with one for each user.

The administrator users here in the Operations Console are not the same as the administrator users in Ivanti Service Manager.

Users who are administrators can delete non-administrator users, and have access to all areas of the Operations Console. Users who are not administrators have a limited set of abilities.

Function	Administrators	Non-Administrators
View the **Users**	Y	Y

Function	Administrators	Non-Administrators
tab		
Create a user	Y	N
Edit their own user account	Y (Can delete their own administrator status)	Y (Cannot make themselves an administrator)
Edit other users	Y (But cannot delete administrator status from the user call Admin)	N
Delete their own user account	N	N
Delete other users who are not administrators	Y (But cannot delete the user called Admin)	N
Delete other users who are administrators	N	N
View the **Tenants** tab	Y	Y
View all tenants and tenant instances in the system	Y	N (Can only view tenants and tenant instances associated with their company, unless they leave the company name blank in which case they can see all tenants and tenant instances)
View the **Landscape** tab	Y	N
View the **System Status** tab	Y	N
View the **Events & Audit History** tab	Y	Y

About the Users Workspace

Use the **Users** workspace to view and edit information about the users who can access the Operations Console. See "Users Tab When Logged In as an Administrator" below.

Users Tab When Logged In as an Administrator

From this workspace you can:

- View the login ID, first name, last name, and company name of each user.

- Create a new user (if you have permission).

- Edit the information for a user (if you have permission).

- Delete a user (if you have permission).

Adding a New Operations Console User

Only administrators can add new users.

To add new users who can access the Operations Console, do the following:

1. Log in to the Operations Console.
2. Click the **Users** tab.
3. Click **Create New**.
4. Enter data in the fields. See "Adding a New User Page" below.

Adding a New User Page

5. Click **Save**.

Editing a Operations Console User

If you are not an administrator, you can only edit yourself, but you cannot make yourself an administrator.

If you are an administrator, you can edit yourself or other users. However, you cannot edit other administrator users to make them non-administrators. You can, however, edit yourself to make yourself a non-administrator, but once you do so, you cannot change yourself back to an administrator.

1. Log in to the Operations Console.
2. Click the **Users** tab.
3. Click **Edit** for the user to edit.
4. Update the fields as needed.
5. Click **Save**.

Deleting a Operations Console User

Only administrators can delete users, but they cannot delete themselves.

1. Log in to the Operations Console.
2. Click the **Users** tab.
3. Click **Delete** for the user to delete.
4. Click **Yes** at the verification prompt.

Changing Your User Display Name

You can change the name that is displayed for first and last name.

1. Log in to the Operations Console.
2. Click **Profile** in the upper right corner.
3. Update your first or last name, or both.
4. Click **Save**.

If you need to change the user display names of other users, follow the process described in "Editing a Operations Console User" above.

Changing Your Password

To change your password, do the following:

1. Log in to the Operations Console.
2. Click **Profile** in the upper right corner.

3. Click **Change password**.

4. Enter the old password, the new password, and then reenter the new password.

5. Click **Save**.

If you need to change the passwords of other users, for example if a user forgets his password, follow the process described in "Editing a Operations Console User" on the previous page.

Viewing System Information

- "About System Information" below

- "Viewing Additional Information" on the next page

- "Fixing Problems" on page 69

About System Information

Use the **System Status** workspace to view status information about the Ivanti Service Manager environments. Information that you can view includes general information such as the Operations Console environment ID and the timeout value used throughout the environment. See "System Status Tab" below.

System Status Tab

For each environment, you can also see the following landscape information:

- The URL for the Ivanti Service Manager configuration server

- The URL for the multi-instance service

- The connection string settings, which describes where the Operations Console resides. Usually, you use the same configuration database as the Operations Console database; however, it can be an independent database anywhere outside of Ivanti Service Manager. You configure this connection string information when you initially install Ivanti Service Manager using the System Configuration Wizard.

For each Ivanti Service Manager database server, you can view:

- IP address

- Backup location

- Data location

- Password

- Administrator user name

For each Ivanti Service Manager application server, you can view:

- The login ID and password for the database that logs events.

- The Ivanti Service Manager database server host name.

- The API key. There can be multiple API keys. The one without any modifiers, such as **API Key (other)**, is the Operations Console API key and is most important. The API key must be unique and cannot be shared between tenant instances. The system automatically generates the API key if you add a tenant instance using the Operations Console, but the field is blank if you add the tenant instance manually.

Viewing Additional Information

When you click **System Status** and view the system information, there may be errors. See "System Status Problems" below. To see information about the error, click **Info**.

System Status Problems

-- Tenant: frs-dev-tenant2.saasitdev.com

LoggingDB_Login	Frs2Wr1teL0g
LoggingDB_Password	***
DBServer.HostName	[blurred]
⚠ API Key	N/A
⚠ API Key (Other)	16A6E7695C0747CF90D2EA[blurred]

-- Tenant: frs-dev-tenant3.saasitdev.com

LoggingDB_Login	Frs2Wr1teL0g
LoggingDB_Password	***
DBServer.HostName	[blurred]
⚠ API Key	N/A

-- Tenant: frs-dev-upgraded-2012-3.saasitdev.com

LoggingDB_Login	Frs2Wr1teL0g
LoggingDB_Password	***
DBServer.HostName	[blurred]
⚠ API Key	N/A
⚠ API Key (Other)	364298F4E7084BBBA7B72[blurred]

-- Tenant: frs-dev-upgraded-2013-1.saasitdev.com

LoggingDB_Login	Frs2Wr1teL0g
LoggingDB_Password	***
DBServer.HostName	[blurred]
API Key	E92116ACEDF54763829E0A[blurred]
⚠ API Key (Other)	364298F4E7084BBBA7B722[blurred]
⚠ API Key (Other)	827D85EE325149238C5924[blurred]

Fixing Problems

When you view the system information, there may be errors. See "System Status Problems" on the previous page. If the system can fix the error, it displays the word **Fix** next to the item. To fix the error, click **Fix**.

One problem that must be fixed is having duplicate API keys. If a tenant instance has a duplicate API key, the system cannot communicate with the tenant instance. To fix this problem, click **Fix**. If the API key is blank, you can generate an API key by clicking **Fix**.
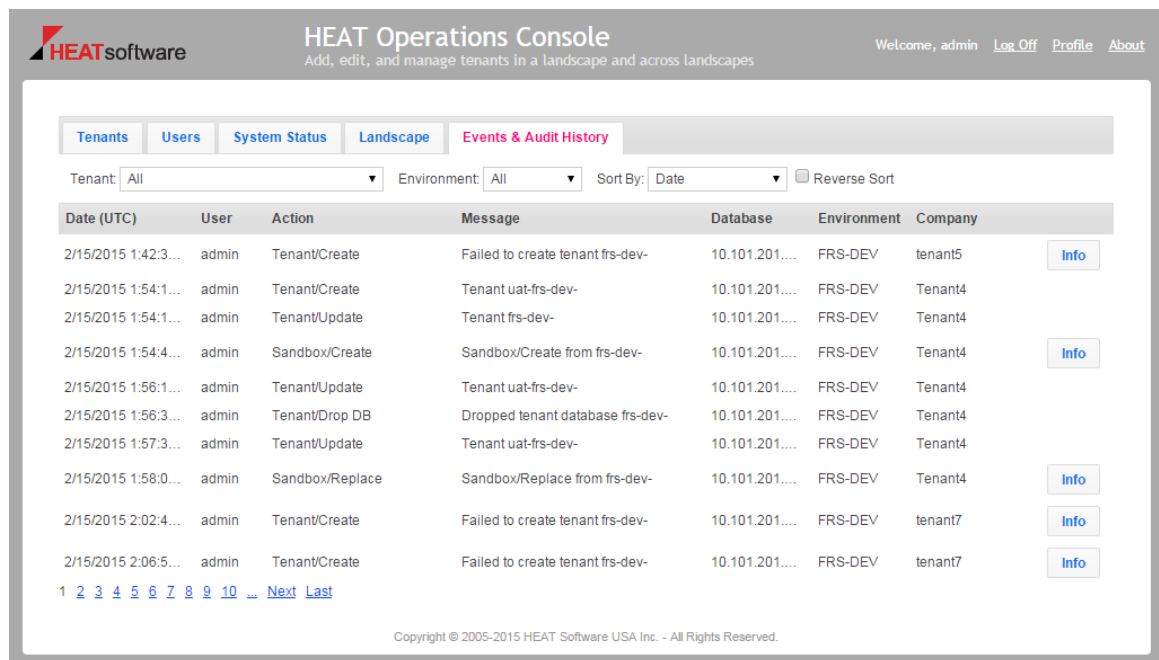
Viewing Events and Audit History

- "About Events and Audit History" below

- "Manipulating Events and Audit History Data" on the next page

- "Viewing Additional Information about Events and Audits" on the next page

About Events and Audit History

Use the **Events & Audit History** workspace to view information about each action that has occurred on a tenant or a tenant instance. You can use this information to help with troubleshooting problems. It lists each action that has occurred in chronological order and displays additional information for errors. See "Events and Audit History" below.

Events and Audit History

The page lists the following information:

- The action date, in UTC time.

- Who performed the action.

- What the action was.

- The message that displayed when the action took place.

- The database on which the action was performed.

- The environment in which the action was performed.

- The company to which the tenant or tenant instance belongs.

Manipulating Events and Audit History Data

Use the controls to do the following:

- Filter the results by tenant, tenant instance, or environment.

- Sort the results by date, tenant, tenant instance, or company name.

- Sort the results in reverse order.

- Move from page to page of data.

Viewing Additional Information about Events and Audits

Some items may have additional information. To see this information, if available, click **Info** next to the message.

Reviewing License Usage

You can use the Operations Console to review the license usage for all tenants and tenant instances by doing the following:

1. Log in to the Operations Console.
2. Click **Tenants**.
3. Click **Download License Usage**.

   The system downloads a Microsoft Excel file with the license information for all tenants and tenant instances. The file lists the date the license was used, the number of licenses used, and the types of licenses used.

Configuring Landscapes

For a definition of landscape and common landscape types, see "Landscapes" on page 15.

- "About the Landscape Workspace" on the next page

- "Working with Landscape Groups" on the next page

- "Working with Landscapes" on page 74

- "Working with Database Servers" on page 78

About the Landscape Workspace

Use the **Landscape** workspace to perform a variety of landscape management activities, including creating and configuring landscape groups, landscapes, and database servers. You must create a landscape group before you can create a landscape, and you must create a landscape before you can create any tenants.

Working with Landscape Groups

- "About Landscape Groups" below

- "Adding a Landscape Group" on the next page

- "Editing a Landscape Group" on page 74

- "Deleting a Landscape Group" on page 74

About Landscape Groups

A landscape group is a group comprised of different landscapes. The following are examples of landscape groups:

- **Independent landscapes group**: The production, staging, and UAT landscapes all have different Ivanti Service Manager configuration databases. Each landscape is on a separate server which makes it easier to independently shut down or restart the servers, if needed. However, this option requires more hardware, which means you need additional licenses.

- **Shared landscapes group**: The production landscape has its own Ivanti Service Manager configuration database and the staging and UAT landscapes share the same Ivanti Service Manager configuration database. This makes it easier to perform upgrades because you have to upgrade the staging and UAT landscapes at the same time. (All landscapes must use the same release of Ivanti Service Manager in order to migrate data.) This is typically how Ivanti Service Manager Cloud systems are deployed.

- **All-in-one landscape group**: The production, staging, and UAT landscapes all share the same Ivanti Service Manager configuration database and servers. This deployment is used mainly by testing and development teams or systems with limited resources.

We recommend using either an independent or shared landscape group. However, there are advantages and disadvantages to each landscape group. See the *Installation and Deployment Guide for Ivanti Service Manager* for more information about the different deployment types.

> We highly recommend that you only install the Operations Console in the staging landscape, and not in any other landscape. For migrating data, you only need to install the Operations Console in one landscape; you should not install it in more than one landscape.

Adding a Landscape Group

You must add a landscape group before you can add any landscapes. To add a landscape group, do the following:

1. Log in to the Operations Console.

2. Click the **Landscape** tab. The system displays the **Landscape** workspace.

3. Click **Landscape Group** on the left. The system displays all of the landscape groups in the system.

4. Click **Add Landscape Group**. The system displays the **Add New Landscape Group** page. See "Add New Landscape Group Page" below.

Add New Landscape Group Page

5. Enter a name for the landscape group.

6. Choose the default backup type. You can either select Microsoft SQL or LiteSpeed. Note that LiteSpeed is not supported for Ivanti Service Manager on-premise customers.

7. If you selected LiteSpeed as the default backup type, you must check **LiteSpeed Enabled**. Otherwise, leave it empty.

> Ivanti Service Manager supports using LiteSpeed for Microsoft SQL Server for backing up your Ivanti Service Manager databases. Before you can use LiteSpeed for Microsoft SQL Server, you must purchase it, install it on your system, and configure the Operations Console to support it.

8. Ensure that **Two Tier Landscape** is not checked. With a two-tier landscape group, the migration path is only between two tenant instances, usually the production instance of the tenant and the staging instance of the tenant. If you do not check th is option, you have a three-tier landscape group. With a three-tier landscape group, there are migration paths between three tenant instances, usually the production instance of the tenant, the staging instance of the tenant, and the UAT instance of the tenant.

> We recommend always using a three-tier landscape group unless you have a demo environment.

9. Click **Save**.

Editing a Landscape Group

1. Log in to the Operations Console.
2. Click the **Landscape** tab. The system displays the **Landscape** workspace.
3. Click **Landscape Group** on the left. The system displays all of the landscape groups in the system.
4. Do one of the following:

    • Click the underlined name of the landscape group. In the resulting dialog box, click **Edit**.

    • Click the pencil icon to the right of the landscape group name.

5. Edit the values.
6. Click **Save**.

Deleting a Landscape Group

1. Log in to the Operations Console.
2. Click the **Landscape** tab. The system displays the **Landscape** workspace.
3. Click **Landscape Group** on the left. The system displays all of the landscape groups in the system.
4. Click the red X to the right of the landscape group name to delete it.
5. In the resulting dialog box, click **OK**.

Working with Landscapes

About Landscapes

A landscape consists of a Ivanti Service Manager configuration database and two Ivanti Service Manager web servers for connecting to the Ivanti Service Manager application server.
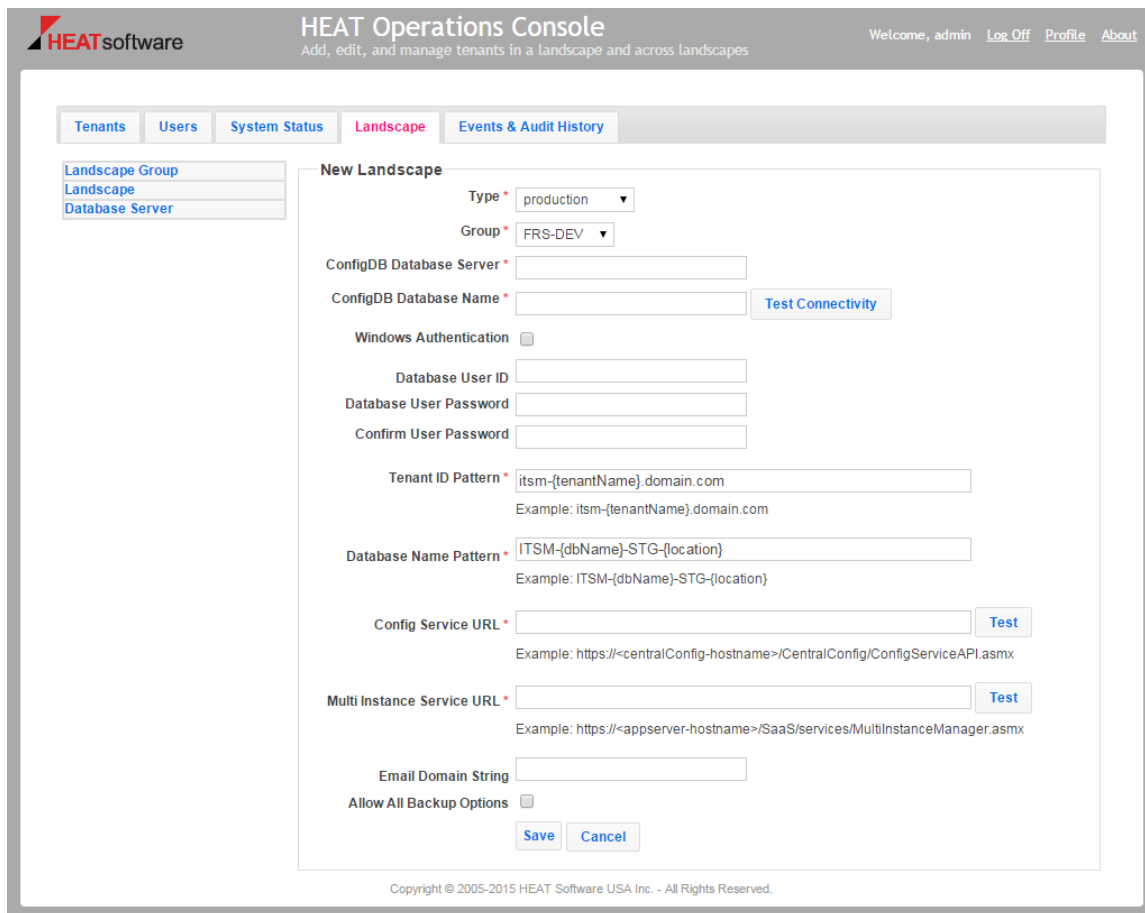
After you create a landscape group, you can start creating landscapes to add to it. You can have a maximum of one of the following landscape types in a landscape group:

- Production

- Staging

- UAT

- Development

- Pilot

Adding a Landscape

1. Log in to the Operations Console.

2. Click the **Landscape** tab. The system displays the **Landscape** workspace.

3. Click **Landscape** on the left. The system displays all of the landscapes in the landscape group.

4. (Optional) To view the landscapes for a different landscape group, select a different landscape group from the drop-down list and click **Go**. The system displays all of the landscapes in the landscape group.

5. Click **Add Landscape**. The system displays the **New Landscape** page. See "New Landscape Page" below.

New Landscape Page

6. Select a landscape type from the drop-down list.

7. Select the landscape group to which this landscape will be attached.

8. Enter the name of the Ivanti Service Manager configuration database server.

9. Enter the name of the Ivanti Service Manager configuration database.

10. Click **Test Connectivity** to test the connection from the Operations Console to the Ivanti Service Manager configuration database.

11. If you are using Windows Integrated Security, check **Windows Authentication**.

> If you check **Window Authentication**, the system hides the **Database User ID**, **Database User Password**, and **Confirm User Password** fields. These fields are only displayed if you do not check **Window Authentication**, which means that you are using Microsoft SQL authentication.

12. If you did not check **Window Authentication**, enter values into the **Database User ID**, **Database User Password**, and **Confirm User Password** fields.

13. Enter a tenant ID pattern. This is the format in which all tenants in this landscape will be named. You can optionally use the **{tenantName}** variable as a placeholder for the tenant name. The following are examples:

    • **{tenantName}** for the production instance of the tenant.

    • **{tenantName}.stg** for the staging instance of the tenant.

    • **{tenantName}.uat** for the UAT instance of the tenant.

    For example, if the value of the **{tenantName}** parameter is **demo1**, then for the production landscape, the tenant ID is demo1 and the URL is http://demo1/HEAT; for the staging landscape, the tenant ID is demo1-STG and the URL is http://demo1-STG/HEAT; and for the UAT landscape, the tenant ID is demo1-UAT and the URL is http://demo1-UAT/HEAT.

14. Enter a database name pattern. This is the format in which all databases in this landscape will be named. You can optionally use the **{dbName}** variable as a placeholder for the database name. The following are examples:

    • **{dbName}** for the production instance of the tenant.

    • **{dbName}-STG** for the staging instance of the tenant.

    • **{dbName}-UAT** for the UAT instance of the tenant.

    For example, if the value of the **{dbName}** parameter is **TRAINING**, then the database name for the production landscape is TRAINING; the database name for the staging landscape is TRAINING-STG; and the database name for the UAT landscape is TRAINING-UAT.

15. Enter the URL for the Ivanti Service Manager configuration service, which is the location of the Ivanti Service Manager configuration database. The value must be in the format **http[s]://***Central_ConfigDB_hostname***/CentralConfig/ConfigServiceAPI.asmx**. Click **Test** to ensure that Ivanti Service Manager can contact the URL.

16. Enter the URL for the multi-instance service. This is the web service URL for connecting to a tenant. The value must be in the format **http[s]://***App_Server_hostname***/HEAT/services/MultiInstanceManager.asmx**. Click **Test** to ensure that Ivanti Service Manager can contact the URL.

17. Enter the email domain string for your organization.

18. Check **Allow All Backup Options** to allow all backup options for tenant and tenant instances that are part of this landscape. See "Backup Types" on page 11 for more information about database backup types.

19. Click **Save**.

Editing a Landscape

1. Log in to the Operations Console.

2. Click the **Landscape** tab. The system displays the **Landscape** workspace.

3. Click **Landscape** on the left. The system displays all of the landscapes in the landscape group.

4. (Optional) To view the landscapes for a different landscape group, select a different landscape group from the drop-down list and click **Go**. The system displays all of the landscapes in the landscape group.

5. Do one of the following:

   - Click the underlined name of the landscape. In the resulting dialog box, click **Edit**.

   - Click the pencil icon to the right of the landscape name.

6. Edit the values.

7. Click **OK**.

Deleting a Landscape

1. Log in to the Operations Console.

2. Click the **Landscape** tab. The system displays the **Landscape** workspace.

3. Click **Landscape** on the left. The system displays all of the landscapes in the landscape group.

4. (Optional) To view the landscapes for a different landscape group, select a different landscape group from the drop-down list and click **Go**. The system displays all of the landscapes in the landscape group.

5. Click the red X to the right of the landscape name.

6. In the resulting dialog box, click **OK**.

Working with Database Servers

- "About Database Servers" below

- "Adding a Database Server" on the next page

- "Editing a Database Server" on page 81

- "Deleting a Database Server" on page 81

About Database Servers

Each landscape must have one or more Ivanti Service Manager database servers associated with it to host the tenant instance databases. The Ivanti Service Manager database server is controlled by the Ivanti Service Manager configuration database. You must specify the data and backup locations for the database server before you can add a tenant using the Operations Console.

Ivanti Service Manager database servers either use Windows Integrated Security or by Microsoft SQL Server authentication, in which case you must enter a user name and password. You define which authentication model to use when you initially install Ivanti Service Manager using the System Configuration Wizard.

Adding a Database Server

To add a database server, do the following:

1. Log in to the Operations Console.

2. Click the **Landscape** tab. The system displays the **Landscape** workspace.

3. Click **Database Server** on the left. The system displays all of the database servers in the landscape.

4. (Optional) To view the database servers for a different landscape, select a different landscape from the drop-down list and click **Go**. The system displays all of the database servers in the landscape.

5. Click **Add Database Server**. The system displays the **Database Server** screen as shown in "Add a Database Server Screen" below.

Add a Database Server Screen

6. Enter a name for the database server. This is usually the host name, IP address, or machine name.

7. Enter a database login ID and password. The database login ID must have permission to back up any tenant or tenant instance database; must have permission to restore any tenant or tenant instance database; and must become the database owner of any new databases that are created on the tenant or tenant instance. The database login ID must have read and write access to the data location and backup locations.

8. Click **Test Connectivity** to test the connectivity between the Operations Console and the Ivanti Service Manager databases.

> If you are using Windows Integrated Security (as opposed to Microsoft SQL authentication), the **Database Login Id**, **Login Password**, and **Confirm Password** fields are not shown. These fields are only displayed if you are using Microsoft SQL authentication.

9. Do not check **Active**, **Clustered**, or **Mirrored**. These are internal settings that should not be used.

10. Enter a location for the database server data. Do not enter a UNC path. Change the application pool identity so that it has read access to the data location.

11. Enter the location of the transaction log. Do not enter a UNC path. Change the application pool identity so that it has read access to the transaction log location.

12. Enter a location for the backup data for the database server. This is used for the automatic transfer of the database backup files from the source Microsoft SQL database server to the target Microsoft SQL database server.

    •   If you have more than one database server, each database server must have its own backup data location, but it must be a shared location that is dedicated for Ivanti Service Manager backups. Use a UNC path if you are using a shared location. Do not use the Microsoft SQL Server default backup location if you have more than one database server.

        •   Change the application pool identity so that it has read and write permissions on both the **Sharing** and **Security** tabs in the Windows Explorer **Properties** dialog box for the shared backup location.

        •   The identity (user) that the database server runs must have read and write permissions on both the **Sharing** and **Security** tabs in the Windows Explorer **Properties** dialog box for the shared folder used for the database backup location of the given database server.

        •   Change the application pool identity to an account that has read and write permission to the backup data location.

    •   If all of the landscapes share the same database instance, you can use one backup data location.

13. To test the backup and restore function, select a database from the drop-down list. Click **Test Backup & Restore**. This may take some time.

> The drop-down list is only populated if you successfully tested the connection as shown in step 8 above.

> Select a small database, such as the Ivanti Service Manager configuration database, to quickly test the user's permission to back up and restore, to ensure that the backup location is set with the correct permissions, and to ensure that the data location for creating the database is correct.

14. Click **Save**.

Editing a Database Server

1. Log in to the Operations Console.

2. Click the **Landscape** tab. The system displays the **Landscape** workspace.

3. Click **Database Server** on the left. The system displays all of the database servers in the landscape.

4. (Optional) To view the database servers for a different landscape, select a different landscape from the drop-down list and click **Go**. The system displays all of the database servers in the landscape.

5. Do one of the following:

   • Click the underlined name of the database server. In the resulting dialog box, click **Edit**.

   • Click the pencil icon to the right of the database server name.

6. Edit the values. See "Configuring Landscapes" on page 71.

7. Click **Save**.

Deleting a Database Server

> You cannot delete a database server that is in use.

1. Log in to the Operations Console.

2. Click the **Landscape** tab. The system displays the **Landscape** workspace.

3. Click **Database Server** on the left. The system displays all of the database servers in the landscape.

4. (Optional) To view the database servers for a different landscape, select a different landscape from the drop-down list and click **Go**. The system displays all of the database servers in the landscape.

5. Click the red X to the right of the database server name.

6. In the resulting dialog box, click **OK**.

Working with Tenants and Tenant Instances

- "About the Tenants Workspace" below

- "How to Create a New Tenant" on page 85

- "Editing a Tenant" on page 91

About the Tenants Workspace

Ivanti Service Manager uses the concept of tenants so that it can update a system database while not touching the official version of the database. The official version of a database is always located on the production instance of the tenant.

If you need to make changes to the official system database, you do not want to make the changes realtime to a live version because you might make a mistake and accidentally delete everything. Therefore, the Operations Console allows you to make a copy of the production tenant, called an instance, and make changes there. When the database changes have been reviewed and approved, only then are they moved back to the production instance of the tenant.

Use the **Tenants** workspace to perform a variety of tenant and tenant instance management activities, including creating, locking, migrating, and deleting tenants and tenant instances.

- "Workspace Filter Controls" below

- "Tenant Display Order Controls" on the next page

- "Expanded View and List View" on the next page

Workspace Filter Controls

Use the **Status** and **Landscape** drop-down boxes to filter which tenants and tenant instances are displayed in the workspace:

- **Status**: Displays tenants and tenant instances based on their status. You can select **All**, **Pending**, **Active**, **OnHold**, **Closed**, or **Demo**.

- **Landscape**: Displays tenants and tenant instances from a landscape type, such as production, or you can choose **All** to display tenants and tenant instances from all landscapes.

> If you do not see as many tenants or tenant instances as you expect when you first open the **Tenants** workspace, select **All** from both the **Status** and **Landscape** drop-down boxes to display all of the tenants and tenant instances on your system.

See "Tenants Expanded View" below.

Tenants Expanded View
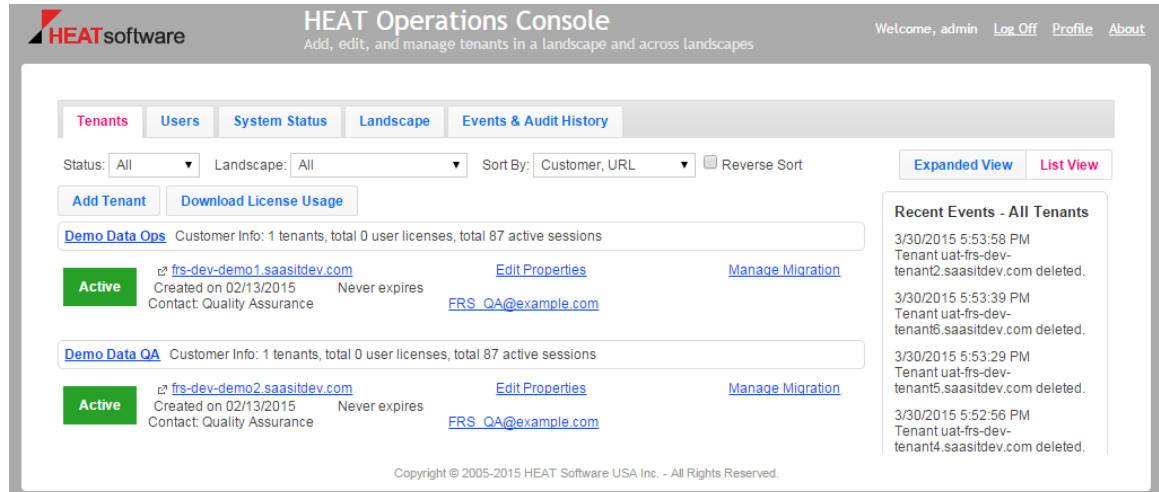
Tenant Display Order Controls

Use the **Sort By** field and the **Reverse Sort** button to configure the order in which tenants and tenant instances are displayed:

- **Sort By**: Select whether to sort by customer and then URL (Customer, URL option); customer and then status (Customer, Status option); or by customer and then expiration date (Customer, Exp Date option) as the tenant and tenant instance display order.

- **Reverse Sort**: Check here to display tenants and tenant instances in reverse alphabetical or numerical order.

See "Tenants Expanded View" on the previous page.

Expanded View and List View

Use the **Expanded View** and **List View** buttons to switch between expanded and list views. The name of the currently selected view is shown in a pink font.

- "Expanded View" below

- "List View " on the next page

Expanded View

Use the **Expanded View** to display the maximum amount of information for each tenant and tenant instance. This information is displayed:

- Tenant or tenant instance name (for example, ConfigDB, Server_01, etc.).

- Ivanti Service Manager database name. This is sometimes the same as the tenant or tenant instance name (for example, ConfigDB), depending on how the tenant or tenant instance was originally configured.

- Tenant or tenant instance creation and expiration dates.

- The name and contact information of the primary contact for the tenant or tenant instance.

- The number of licenses, sessions, and timeout minutes for each tenant or tenant instance.

- A log of recent tenant or tenant instance changes.

You may also see some or all of these links, depending on how the tenant or tenant instance is configured:

- **Edit Properties**: Opens the **Edit** dialog box for the tenant or tenant instance where you can set these parameters:

  - Company name.

  - Status of the tenant or tenant instance (for example, active, pending, on hold, closed, and demo).

  - Put the tenant or tenant instance into maintenance mode.

  - Lock metadata.

  - Tenant or tenant instance expiration date.

  - The number of named and concurrent licenses.

  - The session timeout value.

  - The person who is the point of contact for matters involving the tenant or tenant instance.

- **Manage Migration**: Opens the **Tenants** workspace, from which you can migrate data between tenants or tenant instances.

- **Extend Expiration Date**: Opens the **Expiration Date** dialog box, from which you can select a new expiration date for the tenant or tenant instance.

- **Deactivate**: Opens the **Deactivate** dialog box, from which you can deactivate the tenant or tenant instance.

- **Clear Cache**: Clears the cache of the Ivanti Service Manager application server without having to shut down the application.

List View

Use the **List View** to display a minimal amount of information for each tenant and tenant instance. This information is displayed:

- Tenant or tenant instance name (for example, ConfigDB, Server_01, and so on).

- Tenant or tenant instance creation and expiration dates.

- The name and contact information of the primary contact for the tenant or tenant instance.

- One or both of these links, depending on how the tenant or tenant instance is configured:

  - **Edit Properties**: Opens the **Edit** dialog box for the tenant or tenant instance.

  - **Manage Migration**: Opens the **Tenants** workspace, from which you can migrate data between tenants or tenant instances.

How to Create a New Tenant

When you install Ivanti Service Manager, you automatically create a tenant with a production instance. However, in some situations, you may need to create a new tenant. For example, a customer may have two applications running on one system and each application requires its own tenant.

The following sections describe two ways to create a new tenant:

- "Creating a New Tenant by Using the Operations Console" below

- "Creating a Tenant by Copying an Existing Tenant" on page 89

Creating a New Tenant by Using the Operations Console

The easiest way to create a new tenant is to use the Operations Console.

> ℹ This process creates a new tenant. It does not create instances, such as staging or UAT, of existing tenants.

1. Log in to the Operations Console.
2. Click **Tenant**. The system displays the **Tenants** workspace.
3. Click **Add Tenant**. See "Adding a Tenant" below.

Adding a Tenant

4. Enter information in the fields about the source that will act as a starting point for your new tenant:

   • Choose a source landscape. This is the landscape where the source tenant resides. You can only select **production** or **other**.

- If you select **production**, you can create a tenant in either the **production** or **other** landscapes.

- If you select **other**, you can only create a tenant in the **other** landscape.

- (Optional) Choose a source tenant filter. This allows you to narrow down the source tenants that the system displays in the **Source Tenant** field below.

- Choose a source tenant. These are all the tenants in the source landscape that you selected above.

- Choose a source database option. This is where the database for the source tenant comes from.

    - See "Backup Types" on page 11 for more information about database backup types.

    - The options you see here depend on if you checked **Allow All Backup Options** when you created the landscape. If you checked **Allow All Backup Options**, then you can choose to use live backups when creating a tenant. If you did not check it, you can only restore your database from a previously backed up version. See "Adding a Landscape" on page 75 for more information about the **Allow All Backup Options** parameter.

5. Enter information in the fields about the new tenant:

- Choose a tenant landscape. This is the landscape where the new tenant will reside. You can either choose the production landscape or the other landscape, depending on what you entered in the first bullet of step 4 above.

- Enter a tenant URL pattern. This is the format in which all tenant URLs will be named. You can optionally use the **{tenantName}** variable as a placeholder for the tenant name.

- Choose the tenant database server.

- Enter a tenant database name pattern. This is the format in which all databases will be named. You can optionally use the **{dbName}** variable as a placeholder for the database name.

- Enter the file storage path.

- Enter the company name.

- Enter the tenant status. Can be pending, active, on hold, closed, or demo.

- Enter the expiration date for the tenant. You can choose a specific date or select **never expires**. Choose the expiration date carefully. When a tenant has expired, you cannot log in to it anymore. You must contact Ivanti Software to extend the expiration date.

- Enter the number of named licenses for the tenant.

- Enter the number of concurrent licenses for the tenant.

- Enter the session timeout in minutes. If the session is dormant for longer than this time, you must log in to the system again.

- Choose the primary point of contact for this tenant.

- Enter the inbound and outbound email address for this tenant. Any domain names that you already entered on the **Adding a Landscape** page are populated in the drop-down menu. You can either choose a domain name from the drop-down menu or you can enter a domain name.

- Check **Reuse mailbox if already exists** to reuse the email address if it was already used.

- Check **Enable email listener** to enable the email listener.

- Check **Keep existing users enabled** to keep the existing users enabled.

- Check **Delete audit, log, change log and history data** to delete the data in these tables:

  - Logs

  - Frs_ops_logon_history

  - Frs_ops_gbu_pipeline_history

  - Frs_ops_metadata_history

  - Frs_ops_upload_cache

  - Journal

  - LDAPImportLog

  - Frs_ops_package_item

  - Frs_ops_change_log

  - Frs_ops_change_log_txid

  - Frs_ops_project except for the "default" project

  - Frs_ops_package

  - All audit tables, with the name Audit_*table*

- Check **Scrub email address in users** to ensure that you do not accidentally email users repeatedly during configuration testing on the target tenant instance. If you check this option, the system changes all email fields containing customer email to use the @it.com domain instead of the company domain. This change applies to the user's primary email, contact email, manager email, and team email.

- Check **Scrub email address in other places** to scrub email accounts saved in transactional data (such as in incidents, service requests, workflow tasks, and so on).

- Check **Trim data to keep last 90 days only** to only copy transactional data that was created or modified within the last 90 days. If you are updating the database from a copy of your existing customer database, most records are likely to be current within 90 days.

- Check **Scrub email address in users** to ensure that you do not accidentally email users repeatedly during configuration testing on the target tenant. If you check this option, the system changes all email fields containing customer email to use the @it.com domain instead of the company domain. This change applies to the user's primary email, contact email, manager email, and team email.

- Check **Scrub email address in other places** to scrub email accounts saved in transactional data (such as in incidents, service requests, workflow tasks, and so on).

- Enter the host name of the report server.

- Enter the user name and password for the person who will sign in and use the report server.

6. Click **Create**. The system creates the tenant. This may take two or more minutes. After you create the tenant, you can log in to it.

Creating a Tenant by Copying an Existing Tenant

- "About Copying an Existing Tenant" below

- "Option 1: Tenants are in Separate Environments and do not Share Resources" on the next page

- "Option 2: Tenants are in Separate Environments and Share Resources" on page 91

About Copying an Existing Tenant

Under normal circumstances, you can use the Operations Console to create a new tenant and manage migrations in an automated fashion in a Ivanti Service Manager deployment. In that scenario, all environments are managed as separate tenants on the same installation. That process is described in "Creating a New Tenant by Using the Operations Console" on page 85.

However, there are times when you might need to move or copy a tenant from one landscape (for example, a separate development environment) to another installation (for example, a final production environment). The end result is the creation of a new tenant in the target environment.

We recommend only using this process for creating templates or baselines of existing tenants. We do not recommend using this process to create a new tenant. Use the process described in "Creating a New Tenant by Using the Operations Console" on page 85 to create new tenants.

The following sections describe two similar ways to create a new tenant by copying an existing tenant. In the first scenario (described in "Option 1: Tenants are in Separate Environments and do not Share Resources" below), the source environment and target environment do not share resources. In the second scenario (described in "Option 2: Tenants are in Separate Environments and Share Resources" on the next page), databases and other resources are shared between the source and target environments.

For both options, both the source and destination tenants must be running the same Ivanti Service Manager software release.

We recommend that you only use this method to *create* a tenant. We do not recommend using this method to update an existing tenant.
If the tenant already exists in the destination system and you use this method to update that tenant, **this method completely overwrites the existing tenant, including transactional data**. Therefore, this method is not suitable for moving data or tenants between staging and production servers on an ongoing basis. The person performing the procedure is responsible for ensuring that data is moved correctly and does not accidentally overwrite critical data.

Option 1: Tenants are in Separate Environments and do not Share Resources

Perform the following procedure to create a new tenant.

1. On the source system, use Microsoft SQL Management Server or a similar tool to make a full Microsoft SQL backup of the Ivanti Service Manager tenant database.

2. Paste the Microsoft SQL backup onto the destination system.

3. On the destination system, log in to the Operations Console.

4. If you are going to overwrite a tenant, delete the existing tenant by doing the following:

   a. Click **Tenants**. The system displays the **Tenants** workspace.

   b. Click **Expanded View**.

   c. Select the tenant to delete.

   d. Select **Deactivate** to deactivate the tenant.

   e. Select **Delete** to delete the tenant. Deleting it also deletes the database.

5. Click **Tenants**. The system displays the **Tenants** workspace.

6. Click **Add Tenant**.

7. In the **Source DB Option** field, select **From copied MSSQL backup**.

8. In the **Backup Filename** field, enter the fully-qualified domain name of the source Microsoft SQL backup.

9. Enter the remaining tenant details (for example, tenant URL, DB Server, and so on), and provide a valid expiration date. See "Creating a New Tenant by Using the Operations Console" on page 85 for information on the remaining fields.

10. Click **Create**.

The system creates the new tenant with the data from the Microsoft SQL backup. This creates or overwrites an existing tenant with new data.

Option 2: Tenants are in Separate Environments and Share Resources

You can manually reconfigure the Operations Console on the development and staging instances of a tenant to support the actual production instance and the staging/UAT instances of the tenant on separate database servers. These environments may be accessed by locally-installed Ivanti Service Manager systems as well.

For the purposes of this document, we assume that both the staging and UAT instances of the tenant are managed on a single Ivanti Service Manager system and the actual production environment is deployed on a separate system. The actual production environment must be accessible through database queries and UNC paths from the server on which the staging and UAT instances reside.

Perform the following procedure to create a new tenant:

1. Deploy Ivanti Service Manager in both the staging/UAT and production environments.

2. In the staging/UAT environment, use the Operations Console to push the initial production instance of the tenant to the staging/UAT instances of the tenant. (See "Creating the First Staging or UAT Instance of the Tenant from the Production Instance of the Tenant" on page 37 for details about this procedure.) These will become the permanent staging/UAT instances of the tenant.

3. Use the Operations Console on the staging/UAT instance of the tenant to move the metadata from each tenant instance to the next as required to manage the development lifecycle.

> Be sure to only use the Operations Console in the staging/UAT environment.
> We recommend that you do not install the Operations Console in the production environment in this scenario.

Editing a Tenant

- "Editing the Properties of a Tenant" on the next page

- "Placing a Tenant into Maintenance Mode (Deactivating a Tenant)" on the next page

- "Deleting a Tenant" on page 93

- "Clearing the Cache" on page 93

Editing the Properties of a Tenant

Follow these steps to edit an existing tenant:

> **i** If you only want to extend the expiration date of the tenant, you can click **Extend Expiration Date** from the **Tenants** workspace

1. Log in to the Operations Console.
2. Click **Tenants**. The system displays the **Tenants** workspace.
3. Find the tenant that you want to edit and click **Edit Properties**.
4. Edit any or all of the following fields:

   • Company name

   • Status

   • In maintenance

   • Metadata locked

   • Expiration date

   • Named license

   • Concurrent license

   • Session timeout (in minutes)

   • Contact name

   • Enable email listener

5. Click **Save**.

Placing a Tenant into Maintenance Mode (Deactivating a Tenant)

To place a tenant into maintenance mode, which is also known as deactivating it, do the following:

1. Log in to the Operations Console.
2. Click **Tenants**. The system displays the **Tenants** workspace.
3. Find the tenant that you want to place into maintenance mode (deactivate) and do one of the following:

   • Click **Deactivate**.

   • Click **Edit Properties**, check **In Maintenance**, and click **Save**.

Deleting a Tenant

Before you can delete a tenant, you need to move it into maintenance mode. See "Placing a Tenant into Maintenance Mode (Deactivating a Tenant)" on the previous page.

1. Log in to the Operations Console.

2. Click **Tenants**. The system displays the **Tenants** workspace.

3. Find the deactivated tenant that you want to delete and click **Delete**.

4. Click **OK** in the confirmation dialog box.

Clearing the Cache

If the Ivanti Service Manager application server starts returning incorrect data, you may need to clear the cache. You can click **Clear Cache** to clear the metadata cache on the Ivanti Service Manager application server without having to shut down the Ivanti Service Manager application server. This returns the Ivanti Service Manager application server to the right state and synchronizes it with Ivanti Service Manager.

1. Log in to the Operations Console.

2. Click **Tenants**. The system displays the **Tenants** workspace.

3. Click **Clear Cache**.

Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode

We do not recommend using these methods for migrating data. See "The Four Ways to Migrate Data" on page 21 for a list of the ways to migrate data.

We recommend enabling and using the Ivanti Service Manager development project for migrating data, as described in "Using the Ivanti Service Manager Development Project to Migrate Data" on page 41.

- "About Using Simple Mode and Advanced Mode" below

- "How to Use Simple Mode to Migrate Data" on the next page

- "Using Advanced Mode to Migrate Data" on page 100

- "Migration Scenarios Using Simple Mode and Advanced Mode" on page 120

About Using Simple Mode and Advanced Mode

The relationship data between two business objects is copied from the source to the target under the following operations and conditions:

Operations:

- Simple mode copy

- Copy configuration and master data

- Copy master data only

Condition:

- Only both business objects on both sides of the relationship are included for copying at the same time.

For example, if you copy user-related data, but do not copy the users, then data such as user to team and user to group assignment is not copied. As another example, if you want the service to category assignment to be copied, you must copy both the service (CI#Service) and the category (Category#) in the same push operation.

Choose between the following:

- Simple Mode: See "How to Use Simple Mode to Migrate Data" on the next page.

- Advanced Mode: See "Using Advanced Mode to Migrate Data" on page 100.
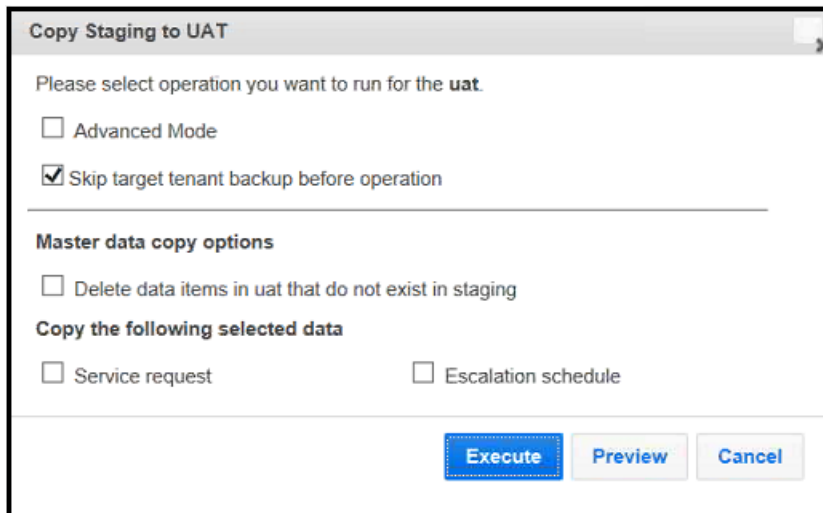
How to Use Simple Mode to Migrate Data

- "Using Simple Mode to Migrate Data" below

- "Simple Mode Migrated Data" on the next page

Using Simple Mode to Migrate Data

When you start the migration, you see the **Copy** dialog box in "Operations Console with Simple Mode and Advanced Mode" below.

Operations Console with Simple Mode and Advanced Mode

1. Do not check **Advanced Mode**, which means you are choosing to use Simple Mode, which migrates a predefined set of data.

2. Check **Skip target tenant backup before operation** to not perform a database backup. Leave the box empty if you want to back up the database before proceeding with the migration.

3. Check **Delete data items in *target_tenant_instance* that do not exist in *source_tenant_ instance*** to delete the data items in the target tenant instance that do not exist in the source tenant instance.

4. Under **Copy the following selected data**, check **Service request** if you want to migrate service request information, and check **Escalation schedule** if you want to migrate escalation schedule information.

5. Do one of the following:

   - Click **Preview** to preview the changes that will be migrated. The system generates an XML file for you to review. You must save the XML file and open it with an XML editor to see the changes. If you need to make additional changes, go back into the system and make them. If you are satisfied with the changes, go back to the Operations Console and click **Execute**.

- Click **Execute** to start the data migration.

Simple Mode Migrated Data

- "Metadata" below

- "Custom Validation Data" on the next page

- "Master Data" on the next page

- "Settings" on page 99

Metadata

> See "Types of Data" on page 11 for information about metadata.

- Deletes the metadata items in the target tenant instance that do not exist in the source tenant instance for the following items (equivalent to checking the box next to the metadata item name in the **Delete metadata items in *target_tenant_instance* that do not exist in *source_tenant_instance*** section):

  - Role

  - Layout

  - Form

  - Grid

  - Trigger

  - Rule

  - Rule override

  - Validation data

  - Validation list definition

- Does not delete the metadata items in the target tenant instance that do not exist in the source tenant instance for the following items (equivalent to not checking the box next to the metadata item name in the **Delete metadata items in *target_tenant_instance* that do not exist in *source_tenant_instance*** section):

  - Business object

  - Branding

  - Workflow

  - Counter

- Grid-part metadata

- Chart-part metadata

- Special-part metadata

- Search

- Report

- Dashboard

- Quick action

- View part

Does not replace the metadata items in the target tenant instance that are newer than those in the source tenant instance for the following items (equivalent to not checking the box next to the metadata item name in the **Replace metadata items in *target_tenant_instance* that are newer than those in *source_tenant_instance*** section):

- Search

- Report

- Dashboard

- Quick action

See "Simple Mode Options -- Top Half" on the next page.

Custom Validation Data

See "Types of Data" on page 11 for information about validation data.

Deletes all custom validation data objects in the target tenant instance that do not exist in the source tenant instance. This is equivalent to checking **Delete data items in *target_tenant_instance* that do not exist in *source_tenant_instance*** in the **Custom defined validation list data objects** section.

See "Simple Mode Options -- Top Half" on the next page.

Master Data

See "Types of Data" on page 11 in the "Terminology" on page 10 section for information about master data.

- Migrates user-related groups of master data. This is equivalent to checking **Delete data items in *target_tenant_instance* that do not exist in *source_tenant_instance*** in the **Master data copy options** section.

- Migrates user-related data (including employee, external contact, profile base, and profile employee information), which is the equivalent of checking **User related** in the **Copy the following selected data** section.

See "Simple Mode Options -- Top Half" below and "Simple Mode Options -- Bottom Half" below.

Simple Mode Options -- Top Half

Simple Mode Options -- Bottom Half

Master data copy options

☑ Delete data items in production that do not exist in staging

Copy the following selected data

☐ User Only

☐ Service request

☐ Escalation schedule

☑ User related

Additional Data

```
Access Point (CI.AccessPoint)
Activity Call Result (FRS_IPCM_VoiceActivityCallF
Address (Address)
ADFS/SAML (Frs_AuthenticationProvider.Saml)
Agent Skill (FRS_IPCM_AgentSkill)
Agent Task (SystemTask.Agent)
Agreement Type (FRS_LM_AgreementType)
Approval (FRS_Approval)
Approval Vote Tracking (FRS_ApprovalVoteTracki
Asset Data Mapping (FRS_IM_AssetDataMapping
```

Execute   Preview   Save Setting   Cancel

Settings

ℹ️ See "Types of Data" on page 11 for information about settings, also referred to as configuration data.

- Deletes settings in the target tenant instance that do not exist in the source tenant instance. This is equivalent to checking **Delete setting records in *target_tenant_instance* that do not exist in *source_tenant_instance*** in the **Settings copy options** section.

- Updates settings that were changed. This is equivalent to checking **Update setting records that were changed** in the **Settings copy options** section.

- Applies the two options above to the following settings:

  - Allowed attachments

  - Dynamic styles

  - Password policy

  - Approval status keywords

  - Email status update mapping settings

  - Ignore message with subject settings

  - Object mapping for email

  - Task assignment status keywords

- Truncate body settings

- Trusted sender domains

See "Simple Mode -- Settings Options" below.

Simple Mode -- Settings Options

Using Advanced Mode to Migrate Data

-

-

-

-

-

-

- "Updating Configuration Data (Copy Translation)" on page 117

- "Increasing the Version Number (No Op [Increase Version Only])" on page 118

Migrating Metadata and Validation Data (Copy Configuration)

Perform the following steps to update the configuration of a tenant instance by migrating metadata and validation data. (See "Types of Data" on page 11 for information about metadata and validation data.) When you perform this migration, metadata and validation data that exist on the target tenant instance are overwritten by the data on the source tenant instance.

> ℹ️ Perform steps 1 to 4 in "Starting the Tenant Instance Migration" on page 34 before you perform the steps in this section.

1. Check **Advanced Mode**.

2. In the **Copy** dialog box, for the **Operation** field, choose **Copy Configuration** to configure the Operations Console to copy metadata and validation data. The dialog box expands and appears similar to the dialog box in "Copy Dialog Box (Migrate Metadata and Validation Data)" below.

> ℹ️ You cannot change the value for the **Target Landscape** field, as it is read-only and depends on the target you chose before you started the migration

Copy Dialog Box (Migrate Metadata and Validation Data)

3. Check **Skip target tenant backup before operation** to select whether to back up the target tenant instance prior to the data migration.

   - To back up the target tenant instance, leave this checkbox empty. If you have not yet backed up the target tenant outside of the Operations Console, we highly recommend that you back it up here.

   - To not back up the target tenant instance, select this checkbox. If you have already backed up the target tenant outside of the Operations Console, it is generally not necessary to back it up here.

4. Use the **Delete metadata items in** *target_tenant_instance* **that do not exist in** *source_tenant_instance* section to select how to process items that were deleted in the source tenant instance prior to the migration.

For example, assume that staging is the source tenant instance and UAT is the target tenant instance. If you deleted some business objects on the staging tenant instance as part of a reconfiguration procedure, you can select whether to delete or retain those business objects on the UAT tenant instance when you perform the data migration.

- To delete business objects on the UAT tenant instance, if those business objects were also deleted on the staging tenant instance, check **Business Objects**.

- To retain business objects on the UAT tenant instance even though they were deleted on the staging tenant instance (that is, to *merge* the staging and UAT business object lists), do not check **Business Objects**.

- Use the checkboxes for other items (layouts, workflows, rules, and so on) in the same way.

If you do not select any options in this section, all data is merged, and no data on the UAT instance of the tenant is deleted.

> ℹ️ Even if the target tenant instance is locked, you can still create, edit, and delete dashboards, quick actions, reports, and searches. If you select any of these items, be aware that they could be deleted or updated when you execute a migration even if the target tenant instance is locked

5. Use the **Replace metadata items in *target_tenant_instance* that are newer than those in *source_tenant_instance*** section to select how to process items that exist on both the source tenant instance and the target tenant instance when the version on the target tenant instance is newer than the version on the source tenant instance. This situation is not typically encountered or recommended, but the ability to respond to it is provided because it does occur occasionally.

For example, assume that staging is the source tenant instance and UAT is the target tenant instance. During testing on the UAT tenant instance, it was determined that a saved search needed to be modified, and the modification was performed on the UAT instance of the tenant. At the same time, another item (for example, change management processes and approvals) was updated on the staging instance of the tenant as part of a scheduled reconfiguration.

- To *merge* the data on the staging instance of the tenant with the data on the UAT instance of the tenant, check **Search**. This results in the most recent change updates (from staging) and the most recent saved search updates (from UAT) coexisting on the UAT instance of the tenant after the migration. This is the recommended procedure because it ensures that the most recent version of all data is present on the UAT instance of the tenant after the migration.

- To *overwrite* all of the data on the UAT instance of the tenant with data from the staging instance of the tenant (including overwriting the saved searches that were updated on the UAT instance of the tenant), do not check **Search**. This results in the most recent change updates (from staging) migrating to the UAT instance of the tenant, but the saved search changes that were made on the UAT instance of the tenant prior to the migration are overwritten and reverted to the version on the staging instance of the tenant.

- Use the checkboxes for other items (dashboards, quick actions, and reports) in the same way.

If you do not select any checkboxes in this section, all data on the UAT instance of the tenant is *overwritten* by data from the staging instance of the tenant. To ensure that all data on the staging instance of the tenant is *merged* with the data on the UAT instance of the tenant (instead of being overwritten), select every checkbox in this section.

> Even if the target tenant instance is locked, you can still create, edit, and delete dashboards, quick actions, reports, and searches. If you select any of these items, be aware that they could be deleted or updated when you execute a migration even if the target tenant instance is locked.

6. Use the **Custom defined validation list data objects** section to select how to process validation lists that have been customized. *Customized* validation lists are not included in the **Validation Data** and **Validation List Definition** checkboxes shown in "Copy Dialog Box (Migrate Metadata and Validation Data)" on page 101 and described in step 3. To migrate customized validation lists, you must specify them in this step instead. (See "Types of Data" on page 11 in the "Terminology" on page 10 section for information about validation data.)

   For example, assume that staging is the source tenant instance and UAT is the target tenant instance. If a customized validation list already exists on the UAT tenant instance, and the validation list is then further customized on the staging tenant instance, you can select whether to *overwrite* the UAT version with the staging version or *merge* the two validation lists.

   - To see the customized validation lists, place the mouse pointer over the checkbox.

   - To delete customized validation list items on the UAT instance of the tenant that were also deleted on the staging instance of the tenant (that is, to *overwrite* the UAT validation list with the staging validation list), check **Delete data items in UAT that do not exist in Staging**.

   - To retain customized validation list items on the UAT instance of the tenant even though they were deleted on the staging instance of the tenant (that is, to *merge* the staging and UAT validation lists), leave the **Delete data items in UAT that do not exist in Staging** checkbox empty.

7. Select one of these options to start or postpone the data migration:

- **Execute**: Starts the migration immediately.

- **Preview**: Creates an XML-based patch file called **MetadataDiff.xmlMetadataDiff.MetadataPatch**. You can review this file to see details of the data that will be migrated. A preview takes the same amount of time to execute as an actual migration.

- **Save Setting**: Saves your current selections without executing a migration. The next time you open this dialog box it is populated with your saved settings.

- **Save UAT Settings**: Saves your current selections for use when you migrate from the UAT instance of the tenant to the production instance of the tenant.

- **Cancel**: Exits the dialog box without saving any settings.

When the data migration finishes, the system creates a log entry for it and updates the status of the affected tenants and tenant instances in the Operations Console migration dashboard. The system also displays an informational message with the results of the migration. Check the **Events & Audit History** workspace to review the warnings and errors before continuing. (See "About the Events & Audit History Workspace" on page 10.)

Updating Metadata, Validation Data, and Master Data (Copy Configuration & Master Data)

Do not use this feature if you have more than 100,000 records as the migration will fail. If you have more data, we recommend that you either exclude some of the records from the data migration or use Microsoft SQL Server scripts or some other tool to migrate the data.

From the Operations Console you can migrate master data together with metadata and validation data in the same migration. (See "Types of Data" on page 11 in the "Terminology" on page 10 section for information about master data, metadata, and validation data.)

Master data consists of data sets that are used across an organization and are typically unique to that organization. User and role definitions, role assignments, and categories are examples of master data.

Perform the following steps to update the configuration for a tenant instance by migrating master data together with metadata and validation data. When you perform this migration, master data, metadata, and validation data that exist on the target tenant instance (UAT in this example) are overwritten by the data on the source tenant instance (staging in this example).

Perform steps 1 to 4 in "Starting the Tenant Instance Migration" on page 34 before you perform the steps in this section.

1. Check **Advanced Mode**.

2. In the **Operation** field in the **Copy** dialog box, choose **Copy Configuration & Master Data** to configure the Operations Console to copy master data together with metadata and validation data. The dialog box expands and appears similar to the dialog box shown in "Copy Dialog Box (Migrate Metadata, Validation Data, and Master Data) -- Top Half" below and "Copy Dialog Box (Migrate Metadata, Validation Data, and Master Data) -- Bottom Half" below .

   Note that the top portion of the dialog box is identical to the dialog box shown in "Copy Dialog Box (Migrate Metadata and Validation Data)" on page 101. The bottom portion of the dialog box shown in "Copy Dialog Box (Migrate Metadata, Validation Data, and Master Data) -- Bottom Half" below (beginning with **Master data copy options**) is new.

Copy Dialog Box (Migrate Metadata, Validation Data, and Master Data) -- Top Half

Copy Dialog Box (Migrate Metadata, Validation Data, and Master Data) -- Bottom Half

Master data copy options

☐ Delete data items in production that do not exist in staging

Copy the following selected data

☐ User
Only
          ☐ Service   ☐ Escalation
          request   schedule
☐ User
related

Additional Data
    Access Point (CI.AccessPoint)
    Activity Call Result (FRS_IPCM_VoiceActivityCallR
    Address (Address)
    ADFS/SAML (Frs_AuthenticationProvider.Saml)
    Agent Skill (FRS_IPCM_AgentSkill)
    Agent Task (SystemTask.Agent)
    Agreement Type (FRS_LM_AgreementType)
    Approval (FRS_Approval)
    Approval Vote Tracking (FRS_ApprovalVoteTracki
    Asset Data Mapping (FRS_IM_AssetDataMapping

[Execute] [Preview] [Save Setting] [Cancel]

3. Fill in the top portion of the dialog box (the settings for metadata and validation data) as described in "Migrating Metadata and Validation Data (Copy Configuration)" on page 101.

4. Use the **Master data copy options** section to select how to process master data that was deleted in the source tenant instance prior to the migration.

   For example, assume that staging is the source tenant instance and UAT is the target tenant instance. If you deleted master data on the staging tenant instance as part of a reconfiguration procedure, you can select whether to delete or retain that master data on the UAT tenant instance when you perform the migration.

   • To delete master data on the UAT tenant instance that was also deleted on the staging tenant instance (that is, to *overwrite* the UAT master data with the staging master data), check **Delete data items in UAT that do not exist in Staging**. Keep in mind that if deleted master data record definitions are used elsewhere in Ivanti Service Manager, the deletion could break existing relationships or otherwise adversely affect the content of data records.

   • To retain master data on the UAT tenant instance even though it was deleted on the staging tenant instance (that is, to *merge* the staging and UAT master data), do not check **Delete data items in UAT that do not exist in Staging**.

   Because of the implications of possibly breaking relationships and data records by deleting master data records in this step, it is sometimes advisable to fill in the **Copy the following selected data** section as described in steps 4 and 5 before making your choice in this step.

5. Use the **Copy the following selected data** section to select which master data to migrate. If you do not select any items in this section, no master data is migrated.

   Master data selection choices are as follows:

- **User Only**: Migrates user-related data (including employee, external contact, profile base, and profile employee information).

- **User related**: Place the mouse pointer over this item to display an expanded list of all business objects contained in this item. Includes contact group, department, email address, organizational unit, and standard user team data). When you select this item, the Operations Console performs a full copy of all master data records in the tenant instance from each of the listed business objects.

If a tenant instance has more than 20,000 users in the system, we do not recommend selecting **User Related** due to performance impact.

- The profile base object is included in this selection, meaning that Profile.Employee and Profile.ExternalContact are migrated. If the configuration includes a newly created child of the group-object Profile# (this is the profile base object), you must include the master data of the associated object as part of the additional data to be copied (described in step 5).

- The organizational unit definition is also included in this item. If any changes were made to organizational units on other tenants through the Ivanti Service Manager Configuration Console or by other means, they are overwritten when this item is migrated.

The **Address.Email** object is also included in this item. If the email addresses for the staging instance of the tenant are "scrubbed" to use the @saasitdemo.com domain instead of the company domain, the scrubbed email addresses for all Profile.object records are migrated to the target domain. If this configuration is eventually migrated to the production instance of the tenant, the @saasitdemo.com domain setting is still in place, and various login, LDAP import, notification, and email delivery issues will occur. A workaround for this scenario is to not scrub email addresses on the staging instance of the tenant, and instead block all outbound email or redirect outbound email to a single email address. The options to block email and to re-direct all mail to a single account can be set in the email configuration business object

- Because user and team email addresses on the staging and UAT instances of the tenant might be scrubbed, the migration does not update existing users or teams. Instead, it migrates only new users and user teams.

- User role assignments and user team assignments are updated as part of this selection.

- **Service request**: Place the mouse pointer over this item to display an expanded list of all business objects contained in this item. When you select this item, the Operations Console performs a full copy (not a partial or selective copy) of all master data records in the tenant instance from each of the listed business objects.

To migrate a subset of the master data records shown in the list (for example, to migrate only the service request category (which has the object name "category")), do not check **Service Request**. Instead, include category in the **Additional Data** section as described in step 5. When you migrate data from the **Additional Data** section, it is a full migration (not a partial or selective migration).

Request offerings and their accompanying workflows are migrated when you select this item. Only those workflows that are part of a request offering design are migrated. Workflows that are associated with the service request business object are not migrated.

- **Escalation schedule**: Place the mouse pointer over this item to display an expanded list of all business objects contained in this item, such as escalation schedule, escalation setting, service level package, and service level target. When you select this item, the Operations Console performs a full copy of all master data records in the tenant instance from each of the listed business objects.

As a general rule, all of the escalation parameters displayed in the Ivanti Service Manager Configuration Console **Escalation** tab are migrated.

Note that when you migrate escalation schedules, existing schedules on the target tenant instance are overwritten.

6. Use the **Additional Data** section to list additional master data or transactional data to migrate. Select one or more items to migrate. The **Additional Data** section contains business objects that:

   - Are not a part of metadata.

   - Are not a part of custom defined validation data.

   - Are not a part of the **User Only**, **User related**, **Service request**, or **Escalation schedule** categories described in step 5.

   - Are not a part of typical transaction data such as incident, task, and change.

   - Might be transactional data. The Operations Console cannot autonomously differentiate between master data and transactional data business objects.

   Guidelines for selecting items are as follows:

   - You can select zero or more items from the **Additional Data** list.

   - Do not select a business object that contains more than 10,000 records due to performance concerns.

- If an object that is usually configured as metadata was for some reason configured as master data (or results in the creation of a related master data object), select the item here so that it is migrated.

For example, validation list data is migrated only if the validation list (that is, the pick list) is based on a business object that is part of the business object list for the validation business object. If the validation list (that is, the pick list) is based on a master business object instead of a validation business object, it will not be migrated by checking **Validation Data**. Instead, you must select the ValidationListData item in the **Additional Data** list (as described here in step 5) in order for the validation list to be migrated.

Click or <ctrl>-click items in the **Additional Data** list to migrate custom data configured as business objects. Keep in mind that migrating business objects with thousands of records can impact performance; these business objects should usually be migrated outside of the Operations Console. Smaller objects (such as Category) are better candidates to select here for migrating within the Operations Console.

Note that the Operations Console cannot perform selective copies during a migration. That is, it can only copy a table to a table. For example, if you select the **Additional Data** item **Table ADFS/SAML (frs_AuthenticationProvider.SAML)** ("Example Item Selected in the Additional Data List" below), *all* of the records from the source database are copied to the target database where frs_AuthenticationProvider.AuthType = SAML. There is no option to copy only "active" or some selective option from the table frs_AuthenticationProvider.SAML.
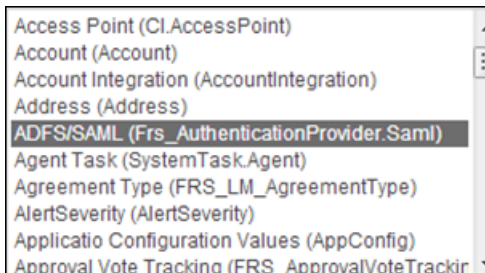
Example Item Selected in the Additional Data List

7. Select one of these options to start or postpone the data migration:

- **Execute**: Starts the migration immediately.

- **Preview**: Creates an XML-based patch file for review.

- **Save Setting**: Saves your current selections without executing a migration. You can continue the migration process later using your saved settings.

- **Save *Target* Settings**: Saves your current selections for use with the next migration in the migration sequence. For example, if the current migration is from staging to UAT, choose this option to save the settings for the UAT to production migration that you will perform next.

- **Cancel**: Exits the dialog box without saving any settings.

When the data migration finishes, the system creates a log entry for it and updates the status of the affected tenants and tenant instances in the Operations Console migration dashboard as shown in "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34. The system also displays an informational message with the results of the migration. Check the **Events & Audit History** workspace to review the warnings and errors before continuing.

Updating Master Data (Copy Master Data Only)

From the Operations Console you can migrate master data only. Only do this after you have migrated the configuration data at least once.

> ⓘ Do not use this feature if you have more than 100,000 records as the migration will fail. If you have more data, we recommend that you either exclude some of the records from the data migration or use Microsoft SQL Server scripts or some other tool to migrate the data.

Master data consists of data sets that are used across an organization and are typically unique to that organization. User and role definitions, role assignments, and categories are examples of master data. (See "Types of Data" on page 11 for information about master data.)

Perform the following steps to update the configuration of a tenant instance by migrating master data only. When you perform this migration, master data that exists on the target tenant instance (UAT in this example) is overwritten by the data on the source tenant instance (staging in this example).

> ⓘ Perform steps 1 to 4 in "Starting the Tenant Instance Migration" on page 34 before you perform the steps in this section.

1. Check **Advanced Mode**.
2. In the **Operation** field in the **Copy** dialog box, choose **Copy Master Data Only** to configure the Operations Console to copy master data only. The dialog box expands and appears similar to the dialog box shown "Copy Dialog Box (Migrate Master Data Only)" below.

Copy Dialog Box (Migrate Master Data Only)

3. Fill in the dialog box (master data copy options) as described in steps 4, 5, 6, and 7 of "Updating Metadata, Validation Data, and Master Data (Copy Configuration & Master Data)" on page 105.

When the data migration finishes, the system creates a log entry for it and updates the status of the affected tenants and tenant instances in the Operations Console migration dashboard as shown in "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34. The system also displays an informational message with the results of the migration. Check the **Events & Audit History** workspace to review the warnings and errors before continuing.

Overwriting all Metadata and Validation Data (Copy to Replace)

From the Operations Console you can perform a migration that overwrites all metadata and validation data on the target tenant instance with data from the source tenant instance. (See "Types of Data" on page 11 for information about metadata and validation data.) This migration is equivalent to creating the first configuration on the target tenant instance as described in "Creating the First Staging or UAT Instance of the Tenant from the Production Instance of the Tenant" on page 37.
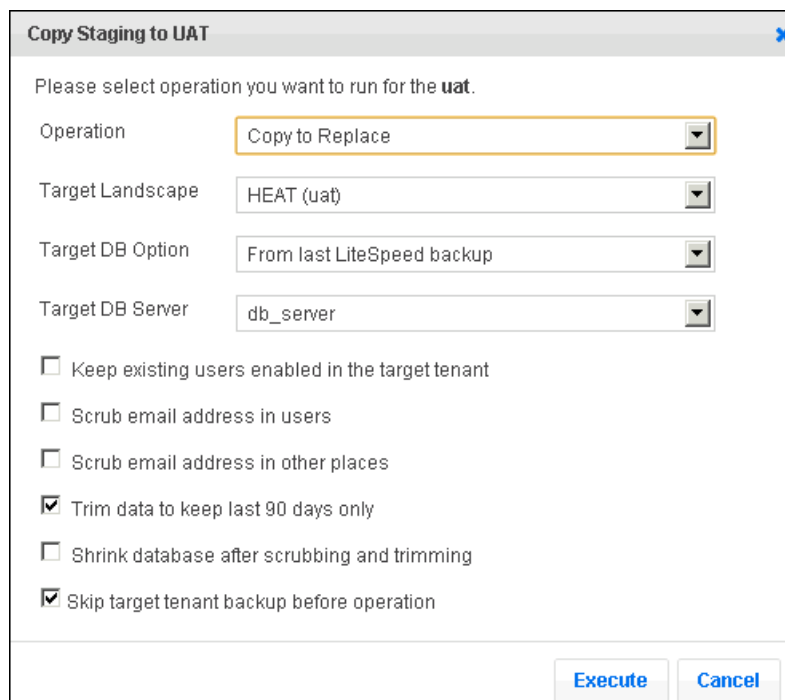
Perform the following steps to update a the configuration for a tenant instance by overwriting all metadata and validation data.

> ℹ️ Perform steps 1 to 4 in "Starting the Tenant Instance Migration" on page 34 before you perform the steps in this section.

1. Check **Advanced Mode**.

2. In the **Operation** field in the **Copy** dialog box, select **Copy to Replace**. This configures the Operations Console to overwrite all metadata and validation data on the target tenant instance. The dialog box appears similar to the dialog box shown in "Copy Dialog Box (To Overwrite Metadata and Validation Data)" below.

3. Perform the procedure described in "Updating Metadata, Validation Data, and Master Data (Copy Configuration & Master Data)" on page 105.

Copy Dialog Box (To Overwrite Metadata and Validation Data)

4. Click **Execute** to initiate the migration.

When the data migration finishes, the system creates a log entry for it and updates the status of the affected tenants and tenant instances in the Operations Console migration dashboard as shown in "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34. The system also displays an informational message with the results of the migration. Check the **Events & Audit History** workspace to review the warnings and errors before continuing.

Updating Only Transactional Data (Copy Data Only)

ⓘ The relationship data between two business objects is not copied from the source to the target when you select the **Updating Only Transactional Data (Copy Data Only)** option.

From the Operations Console you can migrate transactional data (data records, including incidents, changes, service requests, and so on) without migrating any other type of data.

Perform the following steps to update the configuration for a tenant instance by migrating transactional data. When you perform this migration, transactional data that exists on the target tenant instance is overwritten by the data on the source tenant instance.

ⓘ Perform steps 1 to 4 in "Starting the Tenant Instance Migration" on page 34 before you perform the steps in this section.

1. Check **Advanced Mode**.

2. In the **Operation** field in the **Copy** dialog box, select **Copy Data Only** to configure the Operations Console to copy transactional data. The dialog box expands and appears similar to the dialog box shown in "Copy Dialog Box (Migrate Only Transactional Data)" below.

Copy Dialog Box (Migrate Only Transactional Data)

3. Check **Skip target tenant backup before operation** to select whether to back up the target tenant instance prior to the data migration.

   • To back up the target tenant instance, leave this checkbox empty. If you have not yet backed up the target tenant instance outside of the Operations Console, we highly recommend that you back it up here.

   • To not back up the target tenant instance, select this checkbox. If you have already backed up the target tenant instance outside of the Operations Console, it is generally not necessary to back it up here.

4. Use the **Data only copy options** section to select how to process transactional data that was deleted in the source tenant instance prior to the migration.

   For example, assume that staging is the source tenant instance and production is the target tenant instance. If you deleted transactional data on the staging tenant instance as part of a reconfiguration procedure, you can select whether to delete or retain that data on the production tenant instance when you perform the data migration.

   • To delete transactional data on the production tenant instance if that data was also deleted on the staging tenant instance, check **Delete data items in production that do not exist in staging**.

   • To retain transactional data on the production tenant instance even though it was deleted on the staging tenant instance (that is, to *merge* the staging and production transactional data), leave this checkbox empty.

5. In the **Transactional Data** list, select the items to migrate.

6. Click **Execute** to initiate the migration or click **Preview** to create an XML-based patch file for review.

When the data migration finishes, the system creates a log entry for it and updates the status of the affected tenants and tenant instances in the Operations Console migration dashboard as shown in "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34. The system also displays an informational message with the results of the migration. Check the **Events & Audit History** workspace to review the warnings and errors before continuing.

Updating Configuration Data (Copy Settings)

From the Operations Console you can migrate configuration data, also known as settings.

> ℹ️ Perform steps 1 to 4 in "Starting the Tenant Instance Migration" on page 34 before you perform the steps in this section.

1. Check **Advanced Mode**.

2. In the **Operation** field in the **Copy** dialog box, choose **Copy Settings** to configure the Operations Console to copy configuration data. See "Copy Dialog Box (Migrate Settings)" below.

Copy Dialog Box (Migrate Settings)

3. Check **Skip target tenant backup before operation** to select whether to back up the target tenant instance prior to the data migration.

   • To back up the target tenant instance, leave this checkbox empty. If you have not yet backed up the target tenant instance outside of the Operations Console, we highly recommend that you back it up here.

   • To not back up the target tenant instance, select this checkbox. If you have already backed up the target tenant instance outside of the Operations Console, it is generally not necessary to back it up here.

4. Do one of the following:

   • To delete configuration data (settings) on the target tenant instance if that data was also deleted on the source tenant instance, check **Delete setting records in *target_tenant_ instance* that do not exist in *source_tenant_instance***.

- To retain configuration data (settings) on the target tenant instance even though it was deleted on the source tenant instance (that is, to *merge* the source and target configuration data), leave this check box empty.

5. To update configuration data (settings) that has changed in both the source and the target tenant instances, check **Update setting records that changed**.

6. In the **Setting Type** list, select the configuration data types (settings) to migrate.

7. Click **Execute** to initiate the migration or click **Preview** to create an XML-based patch file for review.

When the data migration finishes, the system creates a log entry for it and updates the status of the affected tenants and tenant instances in the Operations Console migration dashboard as shown in "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34. The system also displays an informational message with the results of the migration. Check the **Events & Audit History** workspace to review the warnings and errors before continuing.

Updating Configuration Data (Copy Translation)

From the Operations Console you can migrate translation. The system connects to the source database and copies over any translations that are different or more recent in the target database. If there is no record in the source database but there is a record in the target, the system does not delete the record in the target database.

> ⓘ Perform steps 1 to 4 in "Starting the Tenant Instance Migration" on page 34 before you perform the steps in this section.

1. Check **Advanced Mode**.

2. In the **Operation** field in the **Copy** dialog box, choose **Copy Translation** to configure the Operations Console to copy translated data. See "Copy Dialog Box (Migrate Settings)" on the previous page.

Copy Dialog Box (Copy Translation)

3. Check **Skip target tenant backup before operation** to select whether to back up the target tenant instance prior to the data migration.

   - To back up the target tenant instance, leave this checkbox empty. If you have not yet backed up the target tenant instance outside of the Operations Console, we highly recommend that you back it up here.

   - To not back up the target tenant instance, select this checkbox. If you have already backed up the target tenant instance outside of the Operations Console, it is generally not necessary to back it up here.

4. Click **Execute** to initiate the migration.

When the data migration finishes, the system creates a log entry for it and updates the status of the affected tenants and tenant instances in the Operations Console migration dashboard as shown in "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34. The system also displays an informational message with the results of the migration. Check the **Events & Audit History** workspace to review the warnings and errors before continuing.

Increasing the Version Number (No Op [Increase Version Only])

You can only migrate data from one instance to another if there is a **Push** link and the system only displays a **Push** link if the tenant instances have different version numbers. (The system automatically increases the version number of a tenant instance after a successful migration.) Also, you can only migrate data from staging to production when the production instance of the tenant is at version 0. See "Version Numbers of Tenant Instances" below for an example with **Push** links and different version numbers for each tenant instance.

Version Numbers of Tenant Instances

From the Operations Console you can increase the version number without changing any data by selecting **No Op (Increase version only)**.

> Perform steps 1 to 4 in "Starting the Tenant Instance Migration" on page 34 before you perform the steps in this section.

1. Check **Advanced Mode**.

2. In the **Operation** field in the **Copy** dialog box, choose **No Op (Increase version only)** to increase the version number. See "Copy Dialog Box (No Op [Increase Version Only])" below.

Copy Dialog Box (No Op [Increase Version Only])

3. Click **Execute** to initiate the migration.

When the data migration finishes, the system creates a log entry for it and updates the status of the affected tenants and tenant instances in the Operations Console migration dashboard as shown in "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34. The system also displays an informational message with the results of the migration. Check the **Events & Audit History** workspace to review the warnings and errors before continuing.

Migration Scenarios Using Simple Mode and Advanced Mode

This section shows the steps to take when migrating data in different scenarios and when disabling the Ivanti Service Manager development project and using Simple Mode and Advanced Mode. It skips the conceptual information and only shows the procedure steps. See "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94 for conceptual information and details about these steps.

These examples migrate data by disabling the Ivanti Service Manager development project and using Simple Mode and Advanced Mode.

- "Migrating Data from the Production Instance of the Tenant to the Staging or UAT Instance of the Tenant" on the next page

- "Refreshing the Staging or UAT Instance of the Tenant" on the next page

- "Performing a Quick Migration from the Staging or UAT Instance to the Production Instance" on page 122

- "Migrating Certain Types of Metadata and Validation Data from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance" on page 123

- "Migrating Certain Types of Metadata, Validation Data, and Master Data from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance" on page 123

- "Migrating Only Transactional Data from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance" on page 124

- "Overwriting All Metadata and Validation Data from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance" on page 125

- "Migrating Configuration Data (Settings) from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance" on page 126

- "Example: Migrating Service Level Data" on page 126

Migrating Data from the Production Instance of the Tenant to the Staging or UAT Instance of the Tenant

In this scenario, you are just starting out and have only created an initial production instance of your tenant.

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the landscape you will perform the migration in and click **Manage Migration**.

3. Click **Push** in the arrow representing the migration path to use. (See "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34.) The system displays the **Copy** *source_tenant_instance* **to** *target_tenant_instance* (for example, **Copy Production to Staging**) dialog box.

4. Enter the following information for the first data migration to a tenant instance:

   • Select an option in the **Target DB Option** field.

   • In the **Target DB Server** field, if the target landscape contains more than one database server, select the server that hosts the Ivanti Service Manager configuration database.

   • (Optional) Check **Keep existing users enabled in the target tenant** to keep all the users, and not just the administrators, active.

   • (Optional) Check **Scrub email address in users** to scrub user emails.

   • (Optional) Check **Scrub email address in other places** to scrub email accounts saved in transactional data.

   • (Optional) Check **Trim data to keep last 90 days only** to only copy transactional data that was created or modified within the last 90 days.

   ❌ Do not check this option if you are creating the database from the sample ("out of the box") database, because the records are likely to be older than 90 days in the sample database.

   • (Optional) Check **Shrink database after scrubbing and trimming** to truncate unnecessary data when the tenant instance is created from a backup.

5. Click **Execute**.

Refreshing the Staging or UAT Instance of the Tenant

In this scenario, you have a database on the staging or UAT instance of the tenant and you want to update it with the latest version from the production instance of the tenant.

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the landscape you will perform the migration in and click **Manage Migration**.

3. Click **Push** in the arrow representing the migration path to use. (See "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34.) The **Copy *source_tenant_instance* to *target_tenant_instance*** (for example, **Copy Production to Staging**) dialog box appears.

4. Enter the following information for the first data migration to a tenant instance:

   • Select an option in the **Target DB Option** field.

   • In the **Target DB Server** field, if the target landscape contains more than one database server, select the server that hosts the Ivanti Service Manager configuration database.

   • (Optional) Check **Keep existing users enabled in the target tenant** to keep all the users, and not just the administrators, active.

   • (Optional) Check **Scrub email address in users** to scrub user emails.

   • (Optional) Check **Scrub email address in other places** to scrub email accounts saved in transactional data.

   • (Optional) Check **Trim data to keep last 90 days only** to only copy transactional data that was created or modified within the last 90 days.

   • (Optional) Check **Shrink database after scrubbing and trimming** to truncate unnecessary data when the tenant instance is created from a backup.

5. Click **Execute**.

Performing a Quick Migration from the Staging or UAT Instance to the Production Instance

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the landscape you will perform the migration in and click **Manage Migration**.

3. Click **Push** in the arrow representing the migration path to use. (See "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34.) The **Copy *source_tenant_instance* to *target_tenant_instance*** (for example, **Copy Staging to Production**) dialog box appears.

4. In the **Copy** dialog box, make these selections:

   • Do not check **Advanced Mode**.

   • (Optional) Check **Skip target tenant backup before operation** to not perform a database backup.

   • (Optional) Under **Copy the following selected data**, check **Service request** if you want to migrate service request information, and check **Escalation schedule** if you want to migrate escalation schedule information.

5. Click **Execute**.

Migrating Certain Types of Metadata and Validation Data from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the landscape you will perform the migration in and click **Manage Migration**.

3. Click **Push** in the arrow representing the migration path to use. (See "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34.) The **Copy** *source_tenant_instance* **to** *target_tenant_instance* (for example, **Copy Staging to Production**) dialog box appears.

4. In the **Copy** dialog box, check **Advanced Mode**.

5. In the **Copy** dialog box, for the **Operation** field, choose **Copy Configuration**. (See "Copy Dialog Box (Migrate Metadata and Validation Data)" on page 101 in "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94.)

6. (Optional) Check **Skip target tenant backup before operation** to not back up the target tenant instance prior to the data migration.

7. (Optional) Check the items to delete if those objects were deleted on the target tenant instance in the **Delete metadata items in** *target_tenant_instance* **that do not exist in** *source_tenant_instance* section. If you do not select any checkboxes in this section, all data is merged, and no data on the source tenant instance is deleted.

8. (Optional) Check the items under the **Replace metadata items in** *target_tenant_instance* **that are newer than those in** *source_tenant_instance* section to delete items that exist on both the source tenant instance and the target tenant instance when the version on the target tenant instance is newer than the version on the source tenant instance. If you do not select any checkboxes in this section, all data on the target tenant instance is *overwritten* by data from the source tenant instance.

9. (Optional) Check the items under the **Custom defined validation list data objects** section to delete customized validation list items on the target tenant instance that were also deleted on the source tenant instance.

10. Click **Execute**.

Migrating Certain Types of Metadata, Validation Data, and Master Data from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the landscape you will perform the migration in and click **Manage Migration**.

3. Click **Push** in the arrow representing the migration path to use. (See "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34.) The system displays the **Copy** *source_tenant_instance* **to** *target_tenant_instance* (for example, **Copy Staging to Production**) dialog box.

4. In the **Copy** dialog box, check **Advanced Mode**. The system displays more options in the dialog box.

5. In the **Copy** dialog box, for the **Operation** field, choose **Copy Configuration & Master Data**. (See "Copy Dialog Box (Migrate Metadata, Validation Data, and Master Data) -- Top Half" on page 106 in "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94.)

6. Fill in the top portion of the **Copy** dialog box (the settings for metadata and validation data) as described in "Migrating Certain Types of Metadata and Validation Data from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance" on the previous page.

7. (Optional) Check **Delete data items in** *target_tenant_instance* **that do not exist in** *source_tenant_instance* to delete master data on the target tenant instance that was also deleted on the source tenant instance.

8. (Optional) Check any items under the **Copy the following selected data** section to include them when the data is migrated.

9. (Optional) Select one or more items to migrate from the **Additional Data** list.

10. Click **Execute**.

Migrating Only Transactional Data from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the landscape you will perform the migration in and click **Manage Migration**.

3. Click **Push** in the arrow representing the migration path to use. (See "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34.) The system displays the **Copy** *source_tenant_instance* **to** *target_tenant_instance* (for example, **Copy Staging to Production**) dialog box.

4. In the **Copy** dialog box, check **Advanced Mode**.

5. In the **Copy** dialog box, for the **Operation** field, choose **Copy Data Only**. (See "Copy Dialog Box (Migrate Only Transactional Data)" on page 114 in "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94.)

6. (Optional) Check **Skip target tenant backup before operation** to not back up the target tenant instance prior to the data migration.

7. (Optional) Check **Delete data items in** *target_tenant_instance* **that do not exist in** *source_tenant_instance* under the **Data only copy options** section to delete transactional data on the target tenant instance if that data was also deleted on the source tenant instance.

8. (Optional) In the **Transactional Data** list, select the items to migrate.

9. Click **Execute**.

Overwriting All Metadata and Validation Data from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the landscape you will perform the migration in and click **Manage Migration**.

3. Click **Push** in the arrow representing the migration path to use. (See "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34.) The system displays the **Copy** *source_tenant_instance* **to** *target_tenant_instance* (for example, **Copy Staging to Production**) dialog box.

4. In the **Copy** dialog box, check **Advanced Mode**.

5. In the **Copy** dialog box, for the **Operation** field, choose **Copy to Replace**. (See "Copy Dialog Box (To Overwrite Metadata and Validation Data)" on page 113 in "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94.)

6. Enter the following information:

   • Select an option in the **Target DB Option** field.

   • In the **Target DB Server** field, if the target landscape contains more than one database server, select the server that hosts the Ivanti Service Manager configuration database.

   • (Optional) Check **Keep existing users enabled in the target tenant** to keep all the users, and not just the administrators, active.

   • (Optional) Check **Scrub email address in users** to scrub user emails.

   • (Optional) Check **Scrub email address in other places** to scrub email accounts saved in transactional data.

   • (Optional) Check **Trim data to keep last 90 days only** to only copy transactional data that was created or modified within the last 90 days.

   • (Optional) Check **Shrink database after scrubbing and trimming** to truncate unnecessary data when the tenant instance is created from a backup.

   • (Optional) Check **Skip target tenant backup before operation** to not back up the target tenant instance prior to the data migration.

7. Click **Execute**.

Migrating Configuration Data (Settings) from the Staging or UAT Instance to the Production Instance, or from the Staging Instance to the UAT Instance

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the landscape you will perform the migration in and click **Manage Migration**.

3. Click **Push** in the arrow representing the migration path to use. (See "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34.) The system displays the **Copy *source_ tenant_instance* to *target_tenant_instance*** (for example, **Copy Staging to Production**) dialog box.

4. In the **Copy** dialog box, check **Advanced Mode**.

5. In the **Copy** dialog box, for the **Operation** field, choose **Copy Settings**. (See "Copy Dialog Box (Migrate Settings)" on page 116 in "Appendix B: Migrating Data from One Tenant Instance to Another Using Simple Mode or Advanced Mode" on page 94.)

6. (Optional) Check **Skip target tenant backup before operation** to not back up the target tenant instance prior to the data migration.

7. (Optional) Check **Delete setting records in *target_tenant* that do not exist in *source_tenant*** to delete configuration data (settings) on the target tenant instance if that data was also deleted on the source tenant instance.

8. (Optional) To update configuration data (settings) that has changed in both the source and the target tenant instances, check **Update setting records that changed**.

9. In the **Setting Type** list, select the configuration data types (settings) to migrate.

10. Click **Execute**.

Example: Migrating Service Level Data

This example shows how to migrate service level data such as service level agreements, operational level agreements, underpinning contracts, service level targets, and service level packages. All of these are made up of master data and transactional data.

1. Log in to the Operations Console and click the **Tenants** tab.

2. Navigate to the landscape you will perform the migration in and click **Manage Migration**.

3. Click **Push** in the arrow representing the migration path to use. (See "Migration Dashboard" on page 36 in "Starting the Data Migration" on page 34.) The system displays the **Copy *source_ tenant_instance* to *target_tenant_instance*** (for example, **Copy Staging to UAT**) dialog box.

4. In the **Copy** dialog box, for the **Operation** field, select **Copy Master Data Only**.

5. (Optional) Check **Delete data items in *target_tenant_instance* that do not exist in *source_ tenant_instance*** to delete master data on the target tenant instance that was also deleted on the source tenant instance.

6. Under the **Copy the following selected data** section, check **Service request** and **Escalation schedule**.

7. From the **Additional Data** list, hold down the control key and select all of the following:

- Service Agreement (ServiceAgreement)

- Service Level Agreement (ServiceLevelAgreement.SLA)

- Service Level Objective (ServiceLevelObjective)

- Service Level Package (ServiceLevelPackage)

- Service Level Target (ServiceLevelTarget)

- Service Level Target – General (ServiceLevelTargetGeneral)

- Vendor (Vendor)

8. Click **Execute**.

Appendix C: Using the ISM Development Project to Migrate Data

The information in this appendix is copied directly from the Ivanti Service Manager online help. It describes how to use the Ivanti Service Manager Configuration Console to create packages, which can be used when migrating data. The online help contains additional information about the Ivanti Service Manager development project, such as cautions and limitations about the using the Ivanti Service Manager development project, terminology, information about logs and validation, and examples. (See "Related Documentation" on page 6 for more information about accessing the online help.)

You perform these tasks in the Ivanti Service Manager Configuration Console *and not in the Operations Console.*

This appendix contains the following sections:

- Create a project. See "Creating a Project" on the next page.

- Optional. View the project. See "Viewing a Project" on the next page.

- Make changes to Ivanti Service Manager. See "Creating a Transaction Set" on page 130.

- Optional. View the transactions sets. See "Viewing Transaction Sets" on page 130.

- Create a package. See "Creating a Package" on page 132.

- Optional. View the package. See "Viewing a Package" on page 132.

- Assign transaction sets to a package. See "Assigning Transaction Sets to a Package" on page 133.

- Add an aspect to a package. See "Adding an Aspect to a Package" on page 134.

- Map a package. See "Adding Packages to a Master Package (Mapping)" on page 135.

- Close the package. See "Closing a Package" on page 136.

- Optional. Export a package to another system. See "About Exporting a Package" on page 136.

- Optional. Import a package from another system. See "Appendix C: Using the ISM Development Project to Migrate Data" above. If you get errors, see "Appendix C: Using the ISM Development Project to Migrate Data" above.

- Optional. Save the package as a file. See "Saving a Package as a File" on page 141.

- Optional. Delete a package. See "Deleting a Package" on page 141.

After performing all of these steps, you can migrate the data to the target system.

For information about the packages that have been migrated, see the "Viewing the Patch Log" topic in the Ivanti Service Manager online help. (See "Related Documentation" on page 6 for more information about accessing the online help.)

Creating a Project

You must create a project before you make any configuration changes. If you do not make any projects, all changes are logged to the default project. Use projects to organize the changes made to Ivanti Service Manager. After you create a project and *before* you make any configuration changes, always select your project from the **Project** drop-down list on the top right.

Follow these steps, on the system (such as staging) on which to make changes, to create a project:

1. Log in to Ivanti Service Manager using your named account and not as an administrator. See the Ivanti Service Manager topic called "About Using Your Named Account to Log In" for information about why you must use your named account to log in. (See "Related Documentation" on page 6 for more information about accessing the online help.)

2. Click **Configure Application** to go to the Configuration Console.

3. Click **Build > HEAT Development Project > Projects**. The system displays the **Projects** workspace.

4. Click **New Record**. The system displays a blank project page.

5. Enter a name and description for the project and click **Save**.

6. Click **Refresh**. You can now see the new project name under the **Project** drop-down menu in the top right section of the toolbar (next to the **Help** link).

Viewing a Project

1. From the Configuration Console, click **Build > HEAT Development Project > Projects** to display the **Projects** workspace. The system displays a list of all the projects.

2. Double click a project name to see details of the project.

3. Click the **Transaction Set** tab to see a list of the transaction sets in this project. See "Viewing Transaction Sets" on the next page.

4. Click the **Transaction Detail** tab to see the details of each transaction set. See "Viewing Transaction Sets" on the next page.

5. View the activity history by clicking the **Activity History** tab, any attachments by clicking the **Attachment** tab, or the audit history by clicking the **Audit History** tab.

Creating a Transaction Set

Every time you add, delete, or modify the Ivanti Service Manager configuration, the system records the changes. These recorded changes are called a transaction set. For example, every time that you save changes for a business object, the system records the changes in a transaction set. The system tracks every change and assigns each change a sequential number.

1. From the Configuration Console, click **Build > HEAT Development Project > Projects** to display the **Projects** workspace. The system displays a list of all the projects.

2. From the **Project** drop-down menu in the toolbar, select the project to which the changes you are going to make will belong.

3. Make changes to Ivanti Service Manager as normal. For example, you can create a new business object, form, or list.

In certain cases, when you make changes to Ivanti Service Manager, you must enter your user name and password. When you create and later export the transaction set as part of a package, the package file exposes the user name and password.

To avoid exposing this potentially sensitive information, we recommend that when setting the user name and password for an integration connection, instead of entering a value, you use a global constant. See the "Using Global Constants to Hide Login ID and Password Information" topic in the Ivanti Service Manager online help for information. See the "Viewing, Cloning, and Deleting Web Service Connections" topic in the Ivanti Service Manager online help for an example of using a global constant in this situation. (See "Related Documentation" on page 6 for more information about accessing the online help.)

If you do not use a global constant for the user name and password, the system removes that information from the transaction set.

Viewing Transaction Sets

From the **Transaction Sets** page, you can view the relevant changes that are linked to a transaction set. You can search by transaction set ID. Use the transaction set ID when creating a package.

1. From the Configuration Console, click **Build > HEAT Development Project > Transaction Sets** to open the **Transaction Sets** workspace. The system displays a list of all the transaction sets.

    The system displays the following information:

Field	Description
ID	The ID for this transaction set.
Created On	The date when the transaction set was created.

Field	Description
User ID	The ID of the user who created the transaction set.
Session ID	The ID of the session when the transaction set was created.
DB Revision	The database revision.
Source	Specifies where the transaction was made. *AdminUI* means that the change was made in the Configuration Console, and a tenant URL means that the change was made from a package migration.
Project Name	The name of the Ivanti Service Manager development project to which this transaction set is attached.

2.   Double click a transaction set and click the **Transaction Detail** tab to see the following information:

Field	Description
Project Name	The name of the Ivanti Service Manager development project to which this transaction set is attached.
Transaction Set ID	A sequential number linked to the change. A transaction set may contain zero or more changes. (A zero change indicates the change was aborted, and was not recorded.)
Detail ID	A sequential number linked to the change. A detail may contain zero or more changes. (A zero change indicates the change was aborted, and was not recorded.)
User ID	The ID of the user who created the transaction set.
Operation Date Time	The date and time when the change was recorded.
Operation Type	The type of operation. Can be **C** for new record, **U** for modification, or **D** for deleted.
Object Type	The type for which the change applies. The values can be **BusinessObject** for a business object change, **Form** for a form change, or a business object name such as **AccountStatus#** for change in its data, etc.
Business Object	The name of the business object to which the change is linked, if any.
Entity Name	The key for the business object record itself. In some cases, it is the GUID and in other case, it is a name, such as a business object name, a form name, or a role name.
Source ID	The key for the business object record itself. In some cases, it is the GUID and in other case, it is a name, such as a business object name, a form name, or a role name.
To Publish	Specifies if this transaction set can be published:

Field	Description
	Y: The change is packaged.
	N: This change is a side effect of another change that is published.

Creating a Package

Follow these steps, on the system (such as staging) on which you have made changes, to create a package:

1. From the Configuration Console, click **Build > HEAT Development Package > Packages**. The system displays the **Packages** workspace.

2. Click **New Record**. The system displays a blank package page.

3. Enter a descriptive name for the package. The name must be unique across the system.

4. From the **Type** drop-down list, select if this is an incremental or master package. You cannot edit the type after you save create the package.

   - An incremental package is a single change. It can contain a set of metadata changes and can be imported and exported. There are two types of incremental packages: the first type is exported from the tracked changes in the data and metadata. Every data and metadata change is tracked with a sequence number, which specified the import order. The second type is a result of comparing the full metadata from two tenant instances (such as UAT and staging). There is no sequence associated with these changes. This content comes from the traditional Operations Console migration.

   - A master package contains one or more incremental, master, or file packages. The packages in the master package are called sub-packages. For example, you may want to create a master package with several incremental packages (sub-packages) in it.

5. In Ivanti Service Manager Release 2019.3.1, the **Predicate** field is for internal use only. Do not enter anything here.

6. Enter a description.

7. Click **Save**.

8. Click **Refresh** to see the **Close Package**, **Save As File**, and **Export Package** icons on the top and the tabs at the bottom.

Viewing a Package

1. From the Configuration Console, click **Build > HEAT Development Package > Packages** to open the **Packages** workspace. The system lists the Ivanti Service Manager development packages.

2. Double click a package to view its details.

    The system displays the following tabs:

- **Transaction Set** tab (for incremental packages only). See "Viewing Transaction Sets" on page 130.

- **Transaction Detail** tab (for incremental packages only). See "Viewing Transaction Sets" on page 130.

- **Package Mapping** tab (for master packages only). See "Adding Packages to a Master Package (Mapping)" on page 135.

- **Package Aspect** tab. See "Adding an Aspect to a Package" on the next page.

- **Activity History** tab. See the "About Activity History" topic in the Ivanti Service Manager online help.

- **Attachment** tab. See the "Adding an Attachment" topic in the Ivanti Service Manager online help.

- **Audit History** tab. See the "About Accessing the Audit History" topic in the Ivanti Service Manager online help.

(See "Related Documentation" on page 6 for more information about accessing the Ivanti Service Manager online help.)

Assigning Transaction Sets to a Package

You must create an incremental type package before you can assign transaction sets to it. See "Creating a Package" on the previous page.

Follow these steps, on the system (such as staging) on which you have made changes, to assign transaction sets to a package:

1. From the Configuration Console, click **Build > HEAT Development Project > Transaction Sets** to open the **Transaction Set** workspace.

2. Select the transaction sets to include in the package.

3. Click **Assign To Package**.

4. Use the drop-down menu to select the name of the package to which to assign the transaction sets and click **OK**.

> You can also assign a transaction set to a package by clicking **Build > HEAT Development Package > Packages** to open the **Packages** workspace, clicking the **Transaction Set** tab, and clicking **Add Change Set**.

5. To check that the transactions set are in the package, do the following:

   a. Click **Build > HEAT Development Package > Packages**. The system displays the **Packages** workspace.

b. Open the package. (If you do not see it, click **Refresh**.) The system opens the package.

c. Click the **Transaction Set** tab to view all of the transaction sets.

6. To remove a transaction set from a package, do the following:

a. Click **Build > HEAT Development Package > Packages**. The system displays the **Packages** workspace.

b. Open the package. (If you do not see it, click **Refresh**.) The system opens the package.

c. Click the **Transaction Set** tab to view all of the transaction sets.

d. Highlight the transaction set.

e. Click **Remove From Package**.

7. To add a transaction set to another package, do the following:

a. Click **Build > HEAT Development Package > Packages**. The system displays the **Packages** workspace.

b. Open the package. (If you do not see it, click **Refresh**.) The system opens the package.

c. Click the **Transaction Set** tab to view all of the transaction sets.

d. Highlight the transaction set to add to the package.

e. Click **Assign To Package**.

f. Select a package name from the drop-down list and click **OK**.

Adding an Aspect to a Package

You can only add aspects to a package if it is open. When you created the package, the system copied the aspects from the original package. You can remove the original aspects and add new ones.

1. From the Configuration Console, click **Build > HEAT Development Project > Transaction Sets** to open the **Transaction Sets** workspace.

2. Assign one or more transaction sets to a package.

3. Click the **Package Aspect** tab.

   A package can have zero or more aspects. When the package is imported into target tenant instance, these aspects are entered into or removed from the target tenant instance to flag the aspect presence.

4. View the following information on the **Package Aspect** tab:

Field	Description
To Uninstall	Specifies if the aspect should be removed.
Version	Numeric whole number for this aspect version. The version number for an aspect grows in its development life.
ID	A unique name for the aspect.
Vendor Name	The name of the vendor who provides the aspect. An example is (such as feature/component), Ivanti Software.
Vendor ID	A unique ID that identifies the aspect provider. Currently, the system uses "Ivantisoftware" for aspects provided by Ivanti Software. In future releases, other providers will have a unique vendor ID.
Created By	The ID of the administrator who added the aspect.
Created On	The date and time when the aspect was created.
Modified By	The ID of the administrator who modified the aspect.
Modified On	The date and time when the aspect was modified.

5. Click **New Package Aspect**. The system displays the **New Package Aspect** dialog box.

6. Check **Is Uninstalled** if this package aspect is uninstalled.

7. Enter the following information:

   • **Version number**: Any number.

   • **ID**: Any text.

   • **Vendor ID**: Any text.

8. Click **Save**.

Adding Packages to a Master Package (Mapping)

If you have a master package, you can add incremental or other master packages to it. In this way, the master package contains many changes. This eliminates the need to export incremental packages one at a time.

1. From the Configuration Console, click **Build > HEAT Development Package > Packages** to open the **Packages** workspace.

2. Open a master package.

3. Click the **Package Mapping** tab.

4. Click **Assign Package**. The system displays the **New Master Package Mapping** form.

5. In the **Sub Package** field, click the search icon to display a list of packages that can be mapped.

6. Select a sub-package and click **Use Selected HEAT Release Package**.

7. Enter a value into the **Relative Order** field. This determines the relative order of this sub-package compared to other sub-packages in the master package. The system applies the sub-packages to the tenant in this order (when you export and import the package).

8. Click **Save**.

9. Repeat steps 4 through 8 to add additional sub-packages to the master package.

Closing a Package

You must close a package before you can use the Operations Console to migrate it from staging to production. Closing a package means that you cannot do any more work on it, such as adding or removing packages, adding or removing predicates, changing aspects, and so on. You do not have to close a package to export it.

We recommend that you only close a package after you have completed testing on it and you are ready to migrate it from either staging or UAT to production. After you close a package, you cannot reopen it.

When a package is closed, the system does not display the **Add Change Set** and **Remove From Package** buttons on the **Transaction Set** tab, nor does it display the **New Package Aspect** and **Delete** buttons on the **Package Aspect** tab.

Follow these steps to close a package.

1. From the Configuration Console, click **Build > HEAT Development Package > Packages**. The system displays the **Packages** workspace.

2. Highlight the name of the package to close.

3. Click **Close Package**.

4. At the confirmation prompt, click **OK**.

About Exporting a Package

You can export a package of any type as an XML file and send it to another system, even if the package is empty. You can export any open or closed package. You can use any text editor to view the exported package.

- "About Compacting Export Data" below

- "Exporting a Package" on the next page

About Compacting Export Data

> ℹ️ Compacting export data is a beta feature. Use it with caution.

Starting in HEAT Service Management Release 2016.1, you can compact the transaction sets for a package when you export it. If you make multiple changes to one business object consecutively, compacting combines the individual changes into one compact change. When you import the package, the system only implements the one compacted change instead of the individual changes, which speeds up the process of importing a package.

For example, if you changed a quick action 12 times consecutively, then without compacting, the package would have 12 transaction sets. With compacting, there is only one transaction set.

There are a few caveats to this:

- The system only compacts changes to metadata.

- The system does not combine creating a business object and changing a business object into one compact package. The system only compacts changes to a business object and in this instance, changing a business object does not include creating it.

- The system only compacts changes to the same business object of the same metadata type. If you change two business objects, the system does not compact these two changes together.

Exporting a Package

1. From the Configuration Console, click **Build > HEAT Development Package > Packages** to open the **Packages** workspace.

2. Highlight the name of the package to export.

3. Click **Export Package**. The system gives you the option of compacting the transaction sets. This is a beta feature.

4. Click **Yes** to compact the export data or **No** to not compact the export data. The system exports the package.

Importing a Package

You should never import a package into the production environment from the Configuration Console when you have the Operations Console in place, as doing so could cause system instability.

We highly recommend that you always back up your database before importing a package.

Importing a package is an irreversible operation and cannot be undone.

You can import a package that has been previously exported. For master packages, the system imports the subpackages in the given relative order. The system imports the package if its predicate evaluates to true. If the predicate evaluates to false or an error, the system does not import the package or its subpackages. After importing the package, the system uses the package aspects to update the tenant.

Starting in HEAT Service Management Release 2016.1, the system scans the import package and assigns one of three impact levels (high, medium, and low) to each package.

Impact Level	Definition
High	Contains a database schema change.
	Deletes a business object.
	Updates a stored property for a field of a business object.
	Updates the **IsLocalized** property for a field of a business object.
	Updates the data type, length, or precision for a field of a business object.
	Updates a unique property for a field of a business object.
	Updates a nullable property for a field of a business object.
	Changes the index.
	Changes the full-text search index.
	Adds or deletes an audit field.
Medium	Contains changes to the metadata for a business object.
	Adds a business object.
	Changes the metadata of a business object, other than the changes described above.
	Changes or overrides a business rule.
	Changes or deletes a form, list, layout, quick action, pick list definition, dashboard, chart part, list part, or special part.
	Changes or deletes a role.
	Changes or deletes branding.
	Changes a workflow, counter, view part, global constant, request offering, or report.
	Changes the email configuration.
	Changes integration settings.
	Changes the LDAP configuration or authentication provider.
	Changes the password policy.
	Changes an escalation schedule.
Low	Does not contain a database schema change or changes to the metadata for a business object, or anything listed above.
	Adds a form, list, layout, quick action, pick list definition, dashboard, chart part, list part, special part.
	Add a role.
	Adds branding.

Unlike in previous releases, starting in HEAT Service Management Release 2016.1, you can sometimes import a package even when the system is in metadata read-only mode.

When the system is in metadata read-only mode, note the following:

- For high impact packages, you can only validate the import package. You cannot import the package.

- For medium impact packages, you can validate and import the package, but there may be risks with importing the package. We highly recommend that you back up the database first.

- For low impact packages, you can validate and import the package. We highly recommend that you back up the database first.

Follow these steps to import a package:

1. From the Configuration Console, click **Build > HEAT Development Package > Packages** to open the **Packages** workspace.

2. Click **Import Package**.

3. Click **Browse...** to navigate to the import package.

4. Highlight the file and click **Open**. The system displays a dialog box that lists the file name of the import package.

5. Check **Show Detail** to see the details of the changes in the import package and then click **Preview Impact** to scan the complexity of the import package. Depending on the impact of the import package, the system displays a screen that lists the details of the import package impact.

   If you do not check **Show Detail**, the system only displays the first high or medium level change in the package.

6. Click **Close**. The system displays the **Import Package** dialog box stating that you should back up the database before importing the package, and that this action is irreversible.

7. Do one of the following:

   - Select **Validate** to check the integrity of the package without applying it. If there are any errors, you can go back and fix them before importing the package.

   - Select **Validate and apply if no errors** to import the package only if there are no errors. Note that the system may import the package even if there are errors on the application side. The system displays a message stating if there are any errors and where they are, either on the application side or on the validation side.

   - Select **Apply without validation**. We do not recommend this option. With this option, you assume all responsibility to correct any errors after the import. This imports the package even if there are errors. You should always validate the package and understand the errors first, and then select this option only if you agree to ignore the errors.

8. Click **Execute**

9. At the confirmation prompt, click **Yes**. The system starts the import process.

> If the package contains many changes, the import process can take a long time. Your system may time out or you may accidentally get disconnected from the system due to loss of wifi or VPN. If this happens, log in again. The system continues to import the package and display the importing message. However, the system may not dismiss the importing message, even after the import process has finished. In this case, click **Cancel** or **Refresh** to dismiss it.

> The system displays a message about the success or failure of the import, including specifying if there are any warnings or errors.

> If there are any errors, you cannot import the package. If you do not understand the impact of the errors, we recommend that you restore your database back to the production environment, as the error may disrupt your production operation. See "Appendix C: Using the ISM Development Project to Migrate Data" on page 128 or contact Ivanti Software for further information about the errors.

10. Click **Close**.

11. To view the imported package, do the following:

   a. Click **Build > HEAT Development Project > Projects** to open the **Projects** workspace.

   b. Under the **Name** column, double-click the project that you just imported, to open it.

Troubleshooting Package Import Errors

If you get any errors while importing a package, the package import fails. You must correct the errors before importing the package. If you get any warnings, the import continues; however, we strongly recommend that you correct the warnings before importing the package.

The system displays warnings in orange text and errors in red text.

Try the following for troubleshooting import package warnings and errors:

• If the error message references a sequence number, open the package in a text editor such as Notepad and look for that sequence number. Examine the package for information.

• From the Configuration Console, click **Build > HEAT Development Project > Transaction Sets** to open the **Transaction Set** workspace. Open the transaction set associated with the warning or error. Examine the transaction set for information.

Saving a Package as a File

You can save an incremental or master package as a file (but you cannot save a package as a file if it is already a file) so that you can have a record of it at a certain time. This is similar to versioning the package. You can export or import packages saved as files. All aspects in the package are copied to the file, but you can edit them after they are copied to the file.

Follow these steps to save a package as a file:

1. From the Configuration Console, click **Build > HEAT Development Package > Packages** to open the **Packages** workspace.

2. Highlight the name of the package to save.

3. Click **Save As File**.

4. Enter a new name or description, or leave the default values.

5. Click **Save**. The system saves the package as a file.

6. Click **Refresh** to see the new file in the file list.

Deleting a Package

You can only delete a package if it is open and if it is not being used by another package.

1. From the Configuration Console, click **Build > HEAT Development Package > Packages** to open the **Packages** workspace.

2. Highlight the name of the package to delete.

3. Click the **delete** icon ✖.

Appendix D: Working with Packages from the App Store

- "About the App Store" below

- "Accessing the App Store" below

- "Downloading and Importing an App Store Package" on the next page

About the App Store

Starting in HEAT Service Management Release 2015.1, we released a new applications store (App Store) where you can download optional software created by both Ivanti Software and other companies. The optional software in the App Store is bundled into packages.

The App Store is a collection of apps that can extend your Ivanti Software applications. Some apps include special packages that you can import into Ivanti Service Manager to modify the behavior of the system; other apps are integrations to other applications.

To create your own app, see the link "How to Develop an App" on the home page of the App Store.

The App Store provides different certification levels for the apps, which help you understand the level of support provided for a given app. Some apps are available on an "as-is" basis as a sample, and are available to use at your own risk. Some apps are supported directly by Ivanti Software and some apps come from certified software partners.

Not all apps have an associated package; some require licensing additional software from partners to enable the functionality. These requirements are described in the app description.

Accessing the App Store

Follow these steps to access the App Store:

1. Navigate to https://support.heatsoftware.com.

2. Do the following:

    a. Enter your user name and password.

    b. Press **Login**.

    The system may require you to log in using external authentication. If so, enter your user name and click **Sign in with *your_company_name***.

3. If the system prompts you, select the Self Service role and click **Submit**. The system opens to the Self Service home page.

4. From the top tool bar, click **More...** and select **App Store**. You are now in the App Store.

Downloading and Importing an App Store Package

1. Access the App Store as described in "Accessing the App Store" on the previous page.

2. Open the app to use. Download the documentation associated with the app, and ensure that you meet all of the prerequisites.

3. Download the .metadatapatch files associated with the app. These are the actual package files to be imported.

4. Import the package into the staging instance of your tenant by following the directions in "Importing a Package" on page 137. This automatically creates a project for this package in your staging system.

> You should always validate any new package before applying it to a production environment, even partner certified or Ivanti Software-supplied packages. Package behavior can vary based on your specific configuration. By doing the validation in staging, any necessary changes can also be applied.

5. Test the package on the staging instance of your tenant.

6. If you need to make any changes to your system based on your testing, do the following:

   a. Open the project that the system created when you imported the package. Ensure that this is the active project so that the system adds any changes that you make to this project.

   b. Make the required changes.

   c. Create a new package for the changes. See "Creating a Package" on page 132.

   d. Export the package. See "Exporting a Package" on page 137.

> If you did not make any changes, you do not need to create a new package as you can just use the one that you imported in step 3.

7. Refresh the UAT instance of your tenant. It is very important that the UAT instance accurately reflects the current state of the production instance of your tenant, to get an accurate UAT assessment.

8. Import the package into the UAT instance of your tenant by following the directions in "Importing a Package" on page 137. This automatically creates a project for this package in your UAT system.

9. Test the package on the UAT instance of your tenant.

10. If you need to make any changes to your system based on your testing, do the following:

      a.   Open the project that the system created when you imported the package. Ensure that this is the active project so that the system adds any changes that you make to this project.

      b.   Make the required changes.

      c.   Create a new package for the changes. See "Creating a Package" on page 132.

      d.   Export the package. See "Exporting a Package" on page 137.

> **ⓘ** If you did not make any changes, you do not need to create a new package as you can just use the one that you imported in step 7.

11. Close the package when you are satisfied with it and know that you will make no more changes.

12. Back up the production instance of your tenant.

13. Import the package. If you have locked the production instance of your tenant, enable the Ivanti Service Manager development package and then push the package. (See "The Four Ways to Migrate Data" on page 21 for a list of the ways to migrate data.)