

---

# Project Structures and Team Rights Guide

<b>Overview</b>	<b>2</b>
<b>Integration Storage Hierarchy</b>	<b>3</b>
Recommended Hierarchy Naming Conventions	3
Shared Folders	4
Root Shared	4
Project Space Shared	4
<b>User Groups</b>	<b>5</b>
<b>Default Users</b>	<b>6</b>
<b>Hierarchy Permissions</b>	<b>7</b>
Recommended access to the root “shared” project:	7
Recommended access to the Project Space “shared” project:	7
Recommended access to the Projects:	7

---

## Overview

This document outlines the recommended organizational structure for Project Spaces, Projects and team rights within your SnapLogic org.

Authors: [SnapLogic Enterprise Architecture team](#)

## Integration Storage Hierarchy

In the SnapLogic platform, integrations (pipelines) are managed within the following hierarchy:

- Organization - Multiple customer organizations are configured based on their development environment e.g. DEV, QA, STAGING, UAT, and PROD etc.
- Project Space - Team in a Business Unit or Business Group with an access to a workspace to collaborate and implement integration at one central location
- Project - Group pipelines with an integration, aggregation, and reporting required for the type of function
- Pipeline - A specific implementation of an integration

For example, the path “/MyDevOrg/SLEntArch/SamplePipelines/WorkdayToSnowflake” is broken down into the following components:

- Organization - MyDevOrg
- Project Space - SLEntArch
- Project - SamplePipelines
- Pipeline - WorkdayToSnowflake

## Recommended Hierarchy Naming Conventions

Clarity is one of the most important factors when naming projects spaces, projects, and pipelines. Naming each level of the hierarchy must make sense to all developers and administrators so that integrations can be found quickly and easily. If using Triggered Tasks, it is important to ensure that no characters are used that will violate HTTP URI naming conventions. Further, you may wish to use characters that don't require an URL encoding, such as spaces, commas, etc.

Below is an example naming convention for your project spaces, projects, and pipelines:

- Project spaces are named based on business unit or project team:
  - Sales\_and\_Marketing
  - EnterpriseBI
  - ProfessionalServices
- Projects are named based on business target endpoint
  - Sales\_and\_Marketing/MarketingForecast
  - EnterpriseBI/SalesDashboardReporting
  - ProfessionalServices/CommunityExamples
- Integrations (pipelines) are named based on business function
  - Sales\_and\_Marketing/MarketingForecast/Fall\_Marketing\_New\_Customers\_To\_SalesForce
  - EnterpriseBI/SalesDashboardReporting/Supplier\_Invoices\_To\_Workday
  - ProfessionalServices/CommunityExamples/Community\_27986\_Example\_JSON\_MultiArray\_Join

Keep in mind the shallow hierarchy (project space/project) when considering your naming scheme for project spaces and projects. In most orgs, it is acceptable to assign a project space to each business unit and allow that business unit to create projects within their project space based on integration function or target. However, if you expect a very large number of pipelines to be created by a single business unit, you might want to consider an allowance for multiple project spaces for a given business unit.

## Shared Folders

### Root Shared

A special project named “shared” is added to each SnapLogic Organization (org). Using the org name in the above example, this would be /MyDevOrg/shared. This is commonly referred to as the “root shared” folder. This folder will always exist and is automatically assigned with Full Access (read, write, execute) to all members of the “admins” group of the org and Read-Execute access to all other users. As a best practice, the root shared folder should only contain objects (accounts, files, pipelines, and tasks) that all SnapLogic users in your org should have access to use. Some examples may include:

- SMTP account used for the Email Sender snap
- Readonly database account used to access common, public tables
- Shared expression libraries that contain global static variables or user defined functions for common string/date manipulation
- Shared pipelines such as error handlers or globally re-usable code

### Project Space Shared

Another special project named “shared” is added to each project space in the org. Using the example path above, this would be /MyDevOrg/SLEntArch/shared. This folder will always exist under each project space and inherits the permissions assigned to the project space. As a best practice, the project space shared folder should only contain objects (accounts, files, pipelines, and tasks) that all SnapLogic users with access to the Project Space should have access to use. Some examples may include:

- Database accounts used to access common tables within your business unit
- Shared expression libraries that contain static variables and user defined functions common to your business unit
- Shared/reusable pipelines common to your business unit

---

## User Groups

We recommend that you create the following Groups in all of your SnapLogic orgs:

- Operators - this group contains the users that may need to manually execute a pipeline but do not require Full Access to the projects
- Migrators - this group contains the users that will perform object migrations in your orgs but do not need to Execute pipelines

You should also create Developer groups specific to each Project Space and/or Project within your org. Using the example project spaces and projects listed in the [Naming Conventions](#) section of this document, you may want to add the following Groups:

- Groups specific to Project Space
  - Sales\_and\_Marketing\_Developers
  - EnterpriseBI\_Developers
  - ProfessionalServices\_Developers
- Groups specific to Project
  - MarketingForecast\_Developers
  - SalesDashboardReporting\_Developers
  - CommunityExamples\_Developers

You may choose to enable access for developers to only see objects within the project they are working in, or you could allow read-only access to all projects within their project space to allow for some cross-project design examples.

Typically, the Developer groups will have Full Access in your development org for the Projects that they are working in, with Read-Execute access to the Project Space “shared” folder and Read-Execute access to the root “shared” folder. Developer groups will also have Read-Only access in all non-development orgs for the same Project Space “shared” and Projects that they can access in your development org.

If you have a larger SnapLogic development community in your organization, you may wish to distribute the administration of Projects and create Admin groups for each Project Space who will be assigned ownership of the Project Space, which allows them to create new Projects and maintain all permissions within the Project Space.

---

## Default Users

We recommend that the following service accounts be added to your org(s):

- `snaplogic_admin@<yourdomain>` - this user should own the root SnapLogic shared folder, and all/most SnapLogic Project Spaces in your org(s); add this user to the “admins” group
- `snaplogic_service@<yourdomain>` - this user should own all of your SnapLogic tasks and have permissions related to executing tasks for all Projects. Note that Read-Execute Access is required as a minimum; Full Access is required if any files are written back to the SLDB of the Project during processing. Add this user to the “Operators” group

Note that during migration of tasks to your non-development org(s), you should either use the `snaplogic_service@<yourdomain>` user to perform the migration, or use the [Update Asset Owner API](#) to change the owner of the task after migration. Tasks are owned by the user that creates it; so if a user in the Migrators group performs the migration, they will be assigned as the owner and may not have permissions to successfully execute the task in the target org(s).

## Hierarchy Permissions

### Recommended access to the root “shared” project:

- [admin@snaplogic.com](mailto:admin@snaplogic.com) - Owner
- “admins” group - Full Access
- “members” group - Read/Execute Access
- “Operators” group - Read/Execute Access
- “Migrators” group - Full Access
- “Support” group - Read/Execute Access

You may wish to limit Execute Access to only certain teams. If so, change the “members” group to Read Only Access and grant Read/Execute Access to your desired team groups.

If you perform migrations only within specific day/time windows, you can add/remove users from the Migrators group using a scheduled task that calls the [Groups API](#) to replace all members of the Migrators group and either remove all users from the group (close the migration window) or restore users to the group (open the migration window).

### Recommended access to the Project Space “shared” project:

- [admin@snaplogic.com](mailto:admin@snaplogic.com) - Owner
- “admins” group - Full Access
- “members” group - Read-Only Access (optional)
- “Operators” group - Read/Execute Access
- “Migrators” group - Full Access
- “<Project>\_Admins” group(s) - Full access in development
- “<Project>\_Developers” group(s) - Read/Execute Access in development

You may choose to grant Read-Only access to your <Project>\_Admins and <Project>\_Developers groups in non-development environments depending on your support team structure

### Recommended access to the Projects:

- [admin@snaplogic.com](mailto:admin@snaplogic.com) - Owner
- “admins” group - Full Access
- “members” group - Read-Only Access (optional)
- “Operators” group - Read/Execute Access
- “Migrators” group - Full Access
- “<Project>\_Admins” group(s) - Full Access (only in development)
- “<Project>\_Developers” group(s) - Full Access (only in development)

You may choose to grant Read-Only access to your <Project>\_Admins and <Project>\_Developers groups in non-development environments depending on your support team structure