

Configure CI/CD Deployment Set

Last Modified on 08/27/2026 5:22 am EDT

V11.4

Overview

The Deployment Set feature provides a centralized, structured way to package, export, and deploy platform components across environments. It replaces the fragmented existing Deploy Package experience with a unified Deployment Hub and guided action flow, and a dedicated Deployment Settings management surface.

A Deployment Set is a named, reusable bundle of platform components (like workflows, solutions, configuration, global settings, scripts, etc.) that can be exported to source control and deployed into a target environment in a controlled, repeatable sequence.

In the first phase, you can perform the following actions:

Action	Path	Purpose
Create a Deployment Set	Administration>Deployment Settings	Define a new Set and select the components it contain.
Edit a Deployment Set	Document Set Editor	Adjust components, deployment order, scripts, and secrets.
Manage Deployment Sets	Deployment Sets table	Edit or Delete existing Sets.
Export a Deployment Set	Deployment Hub>Export Set	Package the Set and commit it to Azure DevOps.
Deploy a Deployment Set	Deployment Hub>Deploy Set	Apply a Set's components to the current environment.

Scope and limitations

Phase 1:

- Export and import destination: Azure DevOps
- Custom scripts: SQL scripts
- Secrets: secrets.json is created empty at export; Key Vault references are populated manually before import. Automated tokenization/variable substitution is deferred.
- Components: Phase 1 covers the core component set.
 - Pre-Deployment Scripts
 - Employees
 - Organizations
 - App Variables
 - Global Variables
 - ConfigSets
 - AI Plugins

- Workflows
- Solutions
- Data Streams (Analytics)
- Shared Views
- Global Settings (excluding Application Variables + Global Variables)
- Post-Deployment Scripts

Phase 2 (upcoming releases):

- Export and import destination: External file-storage
- Custom scripts: PowerShell script
- Secrets: Automated tokenization/variable substitution.
- Components: Covers the additional modules.
 - AI Services
 - Archiving
 - Lookup Tables
 - Portal Settings
 - Workflow Security
 - Extended Organization Settings

Key Concepts

Following are the key concepts for the Deployment Set.

- **Deployment Set:** A named package of selected platform components, stored with kind DeploymentSet. It is stored in both the database (for reuse/re-export) and the repository (for version control).
- **Deployment Hub:** In the Administration>Manage Applications, the central screen for executing deployment actions (Export Set, Deploy Set, Deploy a Single Package).
- **Deployment Settings:** The management module (in Administration tile) where sets are listed, created, edited, and deleted.
- **Document Set Editor:** The editor to fine-tune the Set's content, deployment order, scripts, and secrets.
- `manifest.json` : The metadata file at the root of a Deployment Set that lists all selected components and defines the order in which they deploy. Each item carries a mandatory order attribute (integer, default 0, editable).
- `global-settings/` : Environment-specific system modules (connection strings, credentials, listeners, variables, jobs) that act as the override point for environment values without changing workflow definitions.
- `custom-scripts/` : Optional pre-deploy/ and post-deploy/ SQL scripts executed before/after the set is deployed.
- `secrets.json` : Holds Key Vault secret references (populated manually before import).

Deployment settings

In the Administration site, go to **Administration>Deployment Settings>Deployment Sets** to open the Deployment Sets table. The Deployment Sets table lists all existing sets with their metadata (aligned to the AI Plugins table).

Click **Add Deployment Set** below the table to create a new deployment set.

You can manage (update and delete) each row in the table with the Edit  and Delete  options.

genpact

Orchestration AI Suite 11.3.0.417

Global ▼

Administration

AI Services

Analytics

Archiving

Deployment Settings

Global Settings

Lookup Tables

Deployment Sets

+

 Add Deployment Set

Edit	Name	ID	Type	Description	Updated At	Delete
	Business-Fields-Err	5c74ab99-a005-415d-996a-	Default		8/24/2026 8:32:17 AM	
	DashboardSuites_A	faf16e39-5f43-46c6-a3f3-d	Default		8/24/2026 2:26:26 PM	
	inProc	55ead4f2-7f0f-41a1-a02e-7	Default		8/24/2026 6:57:29 AM	
	OrganizationsAndUsers_A	d4b884c3-a6ef-4af5-8aff-09	Default		PM	

Create a Deployment Set

From the Deployment Sets table, click Create Deployment Set. The Create modal dialog opens with three steps.

Step1: Name and Description

Enter a Name (required) and an optional Description. The Create button stays disabled until you enter a name. Click Cancel to close the modal and return to the table with no Set created.

Step 2: Component Selection

Choose the components to include. Categories behave in one of two ways:

- Expandable categories (item-level checkboxes), pick individual items:
 - Workflows
 - Solutions
 - Analytics Dashboards (technically Data Streams/Dashboard Suites)
 - ConfigSets
 - AI Plugins
 - Shared Views
- All-or-nothing categories (single checkbox, no drill-down):
 - Application Variables
 - Employees
 - Organization
 - Jobs

NOTE

Jobs that are automatically created by listener activities are excluded from the Jobs selection.

Step 3: Review and Confirm

The dialog displays a full summary of all selected components, with a Back option to make changes. On confirmation, the dialog closes, the Deployment Sets table refreshes, and the Document Set Editor opens in a new browser tab with the newly created Set.

Edit a Deployment Set

Before exporting a Deployment Set, you can edit and refine it in the Document Set Editor.

- Component content: Review and adjust the included components.
- Deployment order: Each item has a mandatory order attribute (integer, default 0). Edit these values to control sequencing; module-level order follows the export order in the manifest (see Export a Deployment Set section below).
- Custom scripts: Add SQL scripts to custom-scripts/pre-deploy/ and custom-scripts/post-deploy/. The manifest records explicit pre_deployment_scripts / post_deployment_scripts filename lists in execution order.
- Global settings: Environment-specific overrides (connection strings, listeners, variables, jobs, etc.).
- Secrets: secrets.json starts empty; add Key Vault references manually before the Set is imported into a target environment.

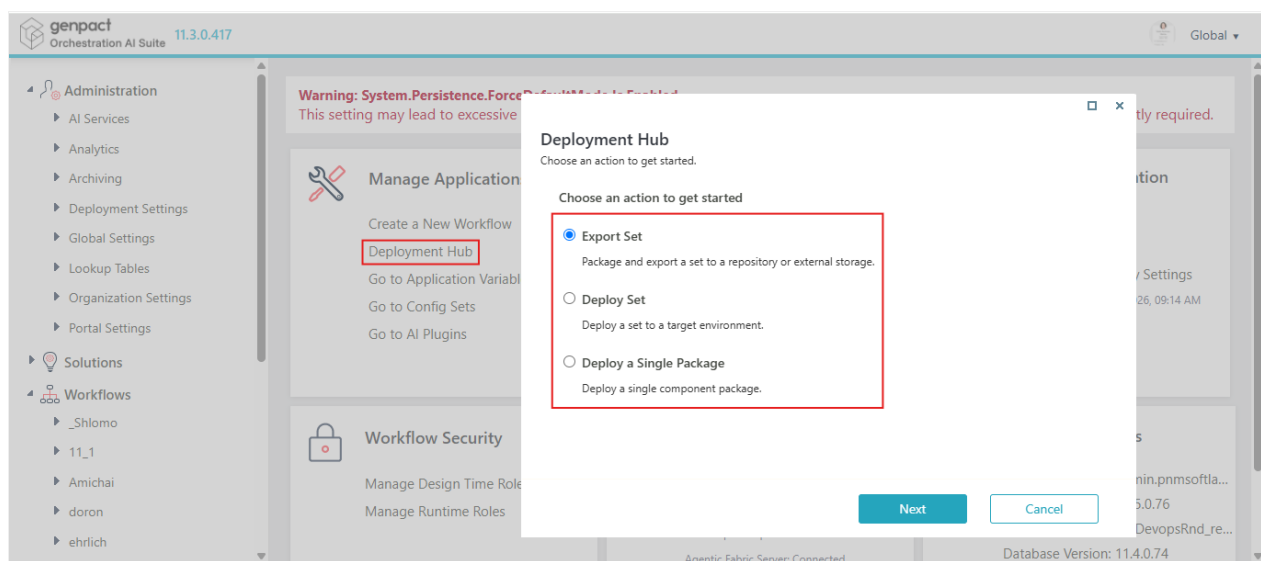
Manage a Deployment Set

From the Deployment Sets table, select a Set to manage. You can perform the following actions:

- Edit: Reopen and update a Set in the Document Set Editor.
- Delete: Remove a Set.

Deployment Hub

1. Open the Admin site homepage.
2. In the Manage Applications tile, select Deployment Hub.
3. The Deployment Hub action dialog displays three actions, **Export Set**, **Deploy Set**, and **Deploy a Single Package**.



NOTE

Access to these tools is restricted to Admin Users and Workflow Developers.

Export a Deployment Set

In the Admin site, open **Deployment Hub**>**Export Set**. The Export dialog guides you through selection, review, destination, and execution.

1. Availability: The export option is disabled when no Deployment Sets exist.

2. Select a Set: From the list of all available Deployment Sets, select a Set for export.
3. Review content: The dialog displays a full summary of the included components. You can deselect components to exclude from this export. Missing or incompatible components are flagged with the message "*Unsupported or missing components detected*". Confirm the selection to proceed.
4. Configure destination: The destination is Azure DevOps. Choose a repository and a branch before continuing.
5. Run the export: Start the export and monitor progress, an overall progress indicator (processed items out of total), and per-component progress (processed out of total per component).
6. Commit: When all components are complete, the system displays a "Committing Deployment Set..." indication while the package is committed to the repository. A Git tag is created after all modules commit successfully.
7. Result: Displays a clear success or failure state. The exported files are committed to *OrchArtifacts/DeploymentSets/*.

Deploying a Deployment Set

In the Admin site, open **Deployment Hub>Deploy Set**. The Deploy dialog mirrors the export flow.

1. Select a set: Choose an existing Deployment Set. Details of the selected Set are loaded.
2. Review content: Displays a full content summary before deployment begins.
3. Confirm: Confirm to trigger the deployment process.
4. Monitor progress: Displays an overall progress indicator (processed out of total) and per-component progress.
5. Result: Displays a clear success or failure state.

Deployment order

Components (mentioned in the Scope and limitations section) are applied in a fixed, dependency-safe sequence (item-level order values fine-tune ordering within this flow).

File Structure Reference

An exported Deployment Set produces the following structure under the existing *OrchArtifacts* repository root. Component-type folders (AIPlugins, DataStreams, ConfigSets, PII, SharedViews, Solutions, Workflows) are pre-existing and referenced via the Set's `manifest.json`. This feature introduces the

`DeploymentSets/<DeploymentSetName>/subtree`.

```

OrchArtifacts/
├─ AIPlugins/
├─ DataStreams/
├─ ConfigSets/
├─ PII/
├─ SharedViews/
├─ Solutions/
├─ Workflows/
└─ DeploymentSets/
  └─ /
    ├── custom-scripts/
    │   ├── pre-deploy/      (SQL scripts only in Phase 1)
    │   └─ post-deploy/     (SQL scripts only in Phase 1)
    ├── global-settings/
    │   ├── ConnectionStringsModule.json
    │   ├── CredentialsModule.json
    │   ├── FileConnectionsModule.json
    │   ├── HttpConsumersModule.json
    │   ├── HttpListenersModule.json
    │   ├── InProcessServiceConsumersModule.json
    │   ├── KafkaConnectionsModule.json
    │   ├── KafkaProducersModule.json
    │   ├── MessageBusPipelinesModule.json
    │   ├── ExternalServiceConsumersModule.json
    │   ├── AppVariablesModule.json
    │   ├── GlobalVariablesModule.json
    │   └─ Jobs.json
    ├── organization-settings/
    │   ├── employees.json
    │   └─ organizations.json
    ├── manifest.json
    └─ secrets.json

```

Sample:

[OrchArtifacts_Example Phase1.zip](#) 

Limitations and workarounds

The following are known technical limitations of the Deployment Set feature in the Phase 1 (V11.4) release, for users to understand current behavior and recommended workarounds.

Limitation	Description	Workaround
Manifest module ordering ("order") is not currently enforced.	The manifest includes an "order" property intended to control the sequence in which modules are imported. In this release, the property is present but has no effect on import order.	If your deployment includes modules with sequencing dependencies, you are responsible for ordering them correctly yourself, for example, by controlling what you include in each export/import pass. Do not rely on the "order" value to enforce sequencing in Phase 1.

Limitation	Description	Workaround
Import mode is fixed to Automatic	The manifest includes an internal "automaticMode" property that is always set to true and is not exposed in the UI. There is currently no supported way to change how import behaves — every import runs in automatic mode.	No configuration is needed or available for import mode in this release; this is expected behavior, not an error.
"Unsupported / missing component detected" error	This message appears specifically when the manifest for the selected deployment set is missing or corrupted.	If you see this error, verify that the deployment set's manifest file exists and is valid in the source repository, then retry. This is not related to the components inside the set.
Exporting/importing Users and Organizations into an environment that already has them.	Users and Organizations can be exported and imported together. However, they are exported with their internal record IDs intact.	- Importing into a new, empty environment is safe. There is no ID collision. - Importing into an environment that already contains users or organizations with the same IDs will overwrite those existing records, which can corrupt data. Only do this if that is the intended outcome. - When deploying Users and Organizations together, Organizations must be processed before Users. Because manifest ordering is not currently enforced (see limitation 1), this sequencing is not automatic; you must manage it manually, for example, by exporting/importing Organizations in a separate pass ahead of Users.

Quick Reference

Task	Path
Create/Manage Sets	Administration>Deployment Settings>Deployment Sets
Export a Set	Administration>Manage Application>Deployment Hub>Export Set
Deploy a Set	Administration>Manage Applications>Deployment Hub>Deploy Set

Task	Path
Deploy a single package	Administration>Manage Applications>Deployment Hub>Deploy a Single Package
Edit Set content	Document Set Editor (opens after Create, or via Edit).

To become familiar with the product terminology, see [this article](#).