Content Deployment

The following content was written by Matt Danda and published on the Qrvey Partner Portal (https://partners.qrvey.com/docs/admin/Content%20Deployment/overview-of-content-deployment) in mid-2023.

Overview of Content Deployment

Content Deployment is the process of copying content from one application to another, in the same environment or to a different one. Content Deployment is a powerful tool for managing all of the content generated in a multi-tenant deployment. Some examples of where this feature may be used are:

- Copying an application that has been created in a staging environment to the production environment.
- Copying some content, or an entire application created by one user to other users' accounts.
- Releasing the new additions to a baseline application to multiple tenants of the system.
- Provisioning a copy of a baseline application to multiple tenants of the system.

To open the Content Deployment page in Qrvey Admin Center, click the Content Deployment icon in the left-hand navigation pane:

Content Deployment requires you to configure four steps:

1. **Servers** define the source and destination instances.
2. **Packages** create a snapshot copy of the source application.
3. **Deployment Definitions** specify the content to copy and what to change during the deployment.
4. **Deployment Jobs** specify the recipient users and the destination instances, and they determine the value of all marked tokens for each recipient or group of recipients.

Once these steps are complete, you can run the deployment job.

Servers

Each instance of the Qrvey platform is considered a "server". Servers are identified by the base URL for the Qrvey instance. For example, the base URL of a Qrvey instance looks something like `https://abcde.qrveyapp.com`. In this example, the Qrvey instance can be uniquely identified by its five character subdomain `abcde`. Each Qrvey instance is referred to as a "server" within Content Deployment, but this term is also synonymous with "environment". A server includes Qrvey Composer, Qrvey Admin Center, the ElasticSearch cluster, users of the instance, and all content and related metadata.

The first step in the Content Deployment process is to create connections to the servers that you wish to use. Each connection is also called a "Server" in the Qrvey Admin user interface, and these servers are displayed on the Server tab of the Content Deployment feature. Once the connection is established, you can use (and re-use) it in the Content Deployment process.

Create a Server

Before you begin

Obtain your unique API key. It was provided in the welcome email that your organization received when your Qrvey instance was created. For more information, see General FAQs in the Qrvey Partner Portal.

Create a server

1. In Qrvey Admin Center, click the **Content Deployment** icon in the left-side menu and display the Server tab.
2. Click **Add Server**. The Add Server dialog displays.
3. Enter the following information in the fields provided:
   - **Server name**. User-defined name of the server.
   - **Description**. User-defined description to help identify the server.
   - **URL**. The base URL of the Qrvey instance.
   - **API key**. The unique API key that was provided to the organization when their Qrvey environment was initially set up.
4. Click **Save**. The new server displays in the list.

Edit a Server

To edit the details of an existing server, click its corresponding three-dot menu in the server list and click **Edit**.

Delete a Server

To delete an existing server from the server list, click its corresponding three-dot menu in the server list and click **Delete**. This will delete the connection to the server, but it will not affect the actual environment.

Note: You cannot delete a server if a package is associated with the server.

Packages and Version

A package enables you to create a snapshot of a selected application with all of its content and dependencies. Each snapshot is called a "version". You can use the same package to create subsequent versions. This enables you to create different versions of this snapshot over time, as the source application changes.

Packages and versions give you the ability to "back up" source applications, which you can deploy at a later time - even if the source application has been changed or even removed.

Note: Packages include all of the objects contained within its source application, but they do not contain the actual data associated with any datasets from the source application. Rather, the data is loaded when the objects are deployed to the target application, after the deployment job completes.

Before You Begin

Verify that the desired source and target server connections have been created. For more information, see Servers.

Create a Package

1. In Qrvey Admin Center, click the **Content Deployment** icon in the left-side menu and display the **Packages** tab.
2. Click **Create Package**. The Package Description page displays.
3. Enter a package name and description in the fields provided.
4. Under Content Source, click the **Server** drop-down and select the server to use as the source.
5. Click the **User** drop-down and select the User that owns the application you wish to include in the package. This list contains only the users with Composer rights on the server.
6. Click the **Source Application** drop-down and select the application you wish to include in the package. The list of applications available is filtered based on the user you selected previously. The Create New Version button activates, and you may now create a new version.

Edit a Package

To edit the details of an existing package, click its corresponding three-dot menu in the package list and click **Edit**.

Delete a Package

To delete an existing package, click its corresponding three-dot menu in the package list and click **Delete**.

Note: You cannot delete a package if a deployment definition is associated with the package.

Create a New Version

1. In Qrvey Admin Center, click the **Content Deployment** icon in the left-side menu and display the **Packages** tab.
2. In the list of available packages, select the desired package. The Package Description page displays.
3. Click **Create New Version**. The Package Version dialog displays.
4. Enter a version name in the field provided, and click **OK**. A new version of the package is created and displays in the Package Versions list.

Edit a Version

To edit the name of an existing version, click it's three-dot menu in the version list and click **Edit**.

Delete a Version

To delete an existing version, click it's three-dot menu in the version list and click **Delete**.

Note: You cannot delete a version if a deployment definition is associated with the version.

Deployment Definitions

A Deployment Definition is a set of instructions for a deployment job. It defines the package version to use, which content from the package to include, and whether any of the content has to be modified during deployment. You can configure the deployment definition to create a new application or to update an existing application.

The deployment definition does not specify the target application. That information is in the deployment job. This separation enables you to use one deployment definition to update multiple target applications for any number of users. To take advantage of this capability, you must use parameter tokens, described next.

Using Parameter Tokens

In some cases, you may want to copy the same application to multiple instances but with different options. For example, you may want to have different connection strings in each destination application, depending on the recipient user or tenant. Qrvey supports user-defined variables called Parameter Tokens for this situation.

Parameter tokens are user-defined parameters that can replace any value and be configured for each recipient at the time of creating the deployment job. Parameter tokens use the following syntax:

```
{{tokenName}}
```

You can replace any value with a parameter token by typing your own token name inside two curly brackets. Note that as soon as you type in the first curly bracket the system will suggest a few names for the token, based on the context that you are in. These names are just suggestions and can be changed to any name that you prefer. For example, if the source application uses a connection to a database that is used for development and testing, and the copied destinations applications have to be each connected to a different production database, you can replace the host URL of the source application with a Parameter Token like `{{hostURL}}`. You may also accept one of the suggested token names, such as `{{connection.host_url}}`. The same process can be repeated to set the username and password of the connection to a token.

Parameter token example

In this example, we want to deploy an application that was created in a staging environment into several tenant accounts in a production environment. The source application is connected to a sample database that is used for development purposes, but each destination environment has its own database. To achieve this goal, replace the following attributes of the database connection with parameter tokens, such as:

- Host URL: `{{host-url}}`
- User Name: `{{db-username}}`
- Password: `{{prodPassword}}`

Later, when this deployment definition is used in a deployment job, the deployment job will resolve the appropriate values for each token and create the database connections.

Configure parameter tokens

1. In the deployment job, display the Deployment Definition Summary section.
2. Locate the parameter token you wish to configure, and click its **Configure** link. A configuration dialog displays.
3. Enter a value for each parameter token. To use the same value for multiple users, select those users, click the **Bulk Edit** link, and enter the common value in the box provided.
4. When finished, click **OK**.
5. To review the details for a user, click its **Details** link.

Using Content Tokens

Content Tokens are system-generated parameters that are automatically created for values that need to be resolved for each recipient. They are created in deployment definitions, and they are resolved when a deployment job runs. Just like Parameter Tokens, these tokens are formatted like `{{token}}`. However, these tokens are assigned by the system and apply to objects, rather than values. In other words, they identify things like "application" and "connection", rather than "application name" and "connection's username".

For example, if a deployment definition is created to update an application, the system assigns the application the content token, `{{ user.application }}`. Later, in the deployment job, if multiple users are picked as recipients of the deployed application, the system resolves the content token for each user. This enables the deployment job to update the application for each of the recipient users.

After determining which target application of each user is going to be updated, all of the other objects which depend on the selected target application, such as the connections, datasets, etc., have to be configured.

Configure content tokens

In the deployment job, display the Deployment Definition Summary section.

1. Locate the content token you wish to configure, and click its **Configure** link. A configuration dialog displays.

   > **Note**: If a Configuration link is disabled, one or more of its prerequisite content tokens has not been configured yet.
   >
   > **Note**: If a deployment job only deploys new content, there are no content tokens to configure.

2. Configure the destination content using the options provided.
3. When finished, click **OK**.

Create a Deployment Definition

Before you begin

Verify that the desired package version has been created. For more information, see Packages.

Create a deployment definition that creates a new application

1. In Qrvey Admin Center, click the **Content Deployment** icon in the left-side menu and display the Deployment Definitions tab.
2. Click **Create Definition**. The Deployment Definition Description page displays.

3. At the top of the page, enter a name and description in the fields provided.
4. Click the **Package** drop-down and select the package to use in this deployment.
5. Click the **Package Version** drop-down and select the package version to deploy.
6. Select the content to deploy:
   - Click **Select Content**. The Select Content dialog displays.
   - Select the individual items to include in the deployment. To include all the content in the application, select **Baseline (all content)**.
   - Click **OK**. The page updates with the dependencies of each of your selections.

> **NOTE**: If you select content that has dependencies, the system automatically includes them in the deployment. For example, if you select a dashboard, but you do not select the datasets that power the charts on that dashboard, the system will automatically include that dataset.

7. (Optional) If you intend to use APIs for Content Deployment, select **Show Content ID**. You will need this information to call the API.
8. Leave **Create New Application** selected.
9. Enter a name in the **New Application Name** field.
10. (Optional) Enter a description in the **Application Description** field.
11. Expand the selections to reveal all their properties. Modify the selections as needed.
12. For datasets, select whether to load the data or to create the dataset without the data. The default setting is to load the data.
13. Insert parameter tokens as needed. For more information, see "Using Parameter Tokens" above.

Create a deployment definition that updates an existing application

1. In Qrvey Admin, click the **Content Deployment** icon in the left-side menu and display the Deployment Definitions tab.
2. Click **Create Definition**. The Deployment Definition Description page displays.
3. At the top of the page, enter a name and description in the fields provided.
4. Click the **Package** drop-down and select the package to use in this deployment.
5. Click the **Package Version** drop-down and select the package version to deploy.
6. Select the content to deploy:
   - Click **Select Content**. The Select Content dialog displays.
   - Select the individual items to including in the deployment. To include all the content in the application, select **Baseline (all content)**.
   - Click **OK**. The page updates with the dependencies of each of your selections.
7. (Optional) If you intend to use APIs for Content Deployment, select Show Content ID. You will need this information to call the API.
8. Select Update an Existing Application. The page updates to display the Select Fields to Update field.
9. Click the Select Fields to Update drop-down and select one of the following options:
   - **Select All**
   - **Application Name**. If selected, a field displays for you to enter a name.
   - **Application Description**. If selected, a field displays for you to enter a description.
10. Expand the selections to reveal all their properties. Modify the selections as needed.

11. For datasets, select whether to load the data or to create the dataset without the data. The default setting is to load the data.
12. Insert parameter tokens as needed. For more information, see "Using Parameter Tokens" above.

> **Note**: An `unknown` label may indicate a new connection type that has not yet been updated in the Content Deployment interface. The new connection type will still be deployed, set up, and used properly.

Delete a Deployment Definition

To delete an existing deployment definition from the list, click its corresponding three-dot menu in the list and click **Delete**. Note that you cannot delete a deployment definition if it is associated with a deployment job.

Deployment Jobs

A Deployment Job executes the type of deployment specified by one or more deployment definitions against one or more target servers. Deployment jobs consist of "blocks" of instructions, with each block configuring the deployment of one deployment definition to any number of users on one destination server. That means that you can have one Deployment Job deploy multiple applications to as many users as necessary, on multiple environments. Deployment jobs require at least one block.

Before You Begin

- Verify that the desired deployment definition has been created. For more information, see Deployment Definitions.
- If you intend to give access to an application to all users from one organization, in the same environment, consider making the application public instead of deploying it to all users. This prevents the application from being replicated into copies of the original.

Create a Deployment Job

1. In Qrvey Admin, click the Content Deployment icon in the left-side menu and display the **Deployment Jobs** tab.
2. Click **Create Job**. The Deployment Job Description page displays.
3. Enter a name and description in the fields provided.
4. Click in the empty block. The block updates to display new fields related to the block.
5. For each block that you wish to add, perform the following:
   - Click the **Select Definition** drop-down and select a deployment definition
   - Click the **Server** drop-down and select the server to which to deploy.
   - Click the **Users** drop-down and select the users to receive the application. The page updates to display details of what is included in the deployment definition.
6. To view additional details, click the **Details** link.

7. Configure content tokens. See "Using content tokens" in Deployment Definition.
8. Configure parameter tokens. See "Using parameter tokens" in Deployment Definition.
9. To add a new block, click **Add Block** and repeat the process described above.

> **Note**: To deploy one application to several destination servers, create a block for each server.

10. To review the details of a section of the deployment definition, click its **Details** link.

Run a Deployment Job

1. In Qrvey Admin, click the **Content Deployment** icon in the left-side menu and display the **Deployment Jobs** tab.
2. Locate the job that you want to deploy in the list, click its three-dot menu, and click **Edit**.
3. Configure content tokens, if necessary. See "Using content tokens".
4. Configure parameter tokens, if necessary. See "Using parameter tokens".
5. To review the details of a section of the deployment definition, click its **Details** link.
6. Click Deploy. A dialog displays providing options for handling failures.
7. Select one of the options provided, and then click **OK**:
   - Skip the installation for the user that failed and continue installing the remaining users
   - Deploy anyway for that user despite the broken or faulty assets The deployment runs.

> **Note**: You cannot modify a deployment job once it has been executed.

Delete a Deployment Job

To delete an existing deployment job, click its corresponding three-dot menu in the deployment job list and click **Delete**.