

API Servers

Defining an API Server is optional, but provides several significant advantages if your repositories are hosted on Azure DevOps, Bitbucket, GitHub, or GitLab and you have a MDWorkflow Pipeline license:

1. Commits of changes made to the code directly on the IBM i can be automatically performed by MDCMS on the developer's behalf, eliminating the need for developers to perform manual Git operations in their local workspace.
2. Pull Requests can be automatically generated once MDCMS completes deployment to a specified target level.
3. Webhook definitions can be generated directly from within MDOpen.

If none of these features applies to your situation, skip to [Git Repositories](#).

If you already have an API Server defined for Pipeline or Task features that points at the correct server for the correct project or organization, you can also use that API Server for Git features in MDOpen.

In MDOpen: **Settings** → **DevOps Settings** → **API Servers** → **Add**

General Settings

Server ID

A unique identifier of up to 10 characters for this server configuration.

Description

A human-readable description of the server.

Server Type

The type of Git server being connected to. One of: Azure DevOps, Bitbucket, GitHub, or GitLab.

API Server URL

The base URL of the Git server, for example `https://dev.azure.com`.

Proxy Address

The URL of the proxy server to use when connecting to the Git server, if applicable.

Proxy Port

The port of the proxy server, if applicable.

Proxy Type

The type of the proxy server. One of: HTTP or SOCKS5

Proxy User

The user to authenticate with the proxy server, if applicable.

Set New Proxy Password

The password to authenticate with the proxy server, if applicable.

Automatically Follow Redirects

- **True** — Automatically follow HTTP redirects when connecting to the Git server. This is typically recommended unless your server is known to have misconfigured redirects.
- **False** — Do not follow HTTP redirects when connecting to the Git server. This may be necessary if your server is known to have misconfigured redirects, but in general should be set to True.

Follow Authorization Header on Redirects

- **True** — Follow the Authorization header when following HTTP redirects.
- **False** — Do not follow the Authorization header when following HTTP redirects.

Maximum Number of Redirects

The maximum number of HTTP redirects to follow when connecting to the Git server.

Allow Certificate Authentication Errors

- **True** — Allow connections to the Git server even if there are SSL certificate errors, such as an expired certificate or a hostname mismatch. This is not recommended for production environments, but may be necessary in some cases such as when connecting to a server with a self-signed certificate.
- **False** — Do not allow connections to the Git server if there are SSL certificate errors. This is typically recommended for production environments.

Azure DevOps Settings

User

The user ID on the server, typically an email address.

Set New Token

A valid Personal Access Token with rights to repositories, webhooks, and pull requests. In Azure DevOps, generate this via **User Settings** → **Personal Access Tokens**.

Organization

The name of the Azure DevOps organization that the project containing the repositories belongs to.

Project

The name of the Azure DevOps project containing the repositories.

Queue Number

The MDAZUR job number that should be used for importing Work Items from Azure DevOps. This parameter is not relevant for Git features.

Bitbucket Cloud Settings

Token Type

Specifies which kind of token will be used for authentication.

- **Access Token** — A token defined at the repository, project, or workspace level.
- **API Token** — A token defined for a specific user.

User

The user the API Token belongs to. Leave blank if using an Access Token.

Set New Token

A valid Access Token or API Token with rights to repositories, webhooks, and pull requests.

Workspace

The ID of the Bitbucket workspace containing the repositories. This value is visible in the URL when viewing the Repositories list.

Bitbucket Data Center (or Server) Settings

User

The user ID on the Bitbucket Data Center or Server instance.

Set New Token

A valid Personal Access Token with rights to repositories, webhooks, and pull requests. In Bitbucket, generate this via **Manage User** → **Personal Access Tokens**.

Project

The key of the Project containing the repositories.

GitHub Settings

User

The user ID on the GitHub server.

Set New Token

A valid Personal Access Token with rights to repositories, webhooks, and pull requests. In GitHub, generate this via **User Settings** → **Settings** → **Developer Settings** → **Personal Access Tokens**.

Organization

The name of the GitHub organization that the repositories belong to, if applicable. Leave blank if the repositories are not under an organization.

GitLab Settings

User

The user ID on the GitLab server.

Set New Token

A valid Personal Access Token with rights to repositories, webhooks, and pull requests. In GitLab, generate this via **User Settings** → **Preferences** → **Access Tokens**. Set the token scope to `api`.