

# Git Repositories

A Git Repository is the definition of a Git repo on another server that MDCMS can connect to and clone into the IFS under `/MDCMS/EXTREF/<repo-id>/tree`.

In MDOpen: **Settings** → **DevOps Settings** → **Git Repositories**

---

## Repository Settings

### Repository ID

A unique identifier of up to 10 characters for this repository.

### URL

The network address (including `http(s)://` or `git@`) for the repository. The correct URL can usually be copied from the Git server administration screen. If the URL is in HTTP format and contains `//user@host`, remove the `user@` portion before saving.

### Git Type

The type of Git server hosting the repository. MDCMS has a specific webhook defined for Azure DevOps, Bitbucket, GitHub, and GitLab, enabling direct CI triggering. Selecting **Other** requires a pipeline tool such as Jenkins to act as an intermediary, which then triggers CI via the MDCMS API endpoint `/git/checkout`. You may also choose **Other** for a supported server type if you prefer MDCMS not to be involved until later in the DevOps process and want a master pipeline tool to orchestrate CI instead.

### Description

A human-readable description of the repository.

### Credential ID

The ID of the credentials used to access the repository. Required unless the repository allows anonymous access. Use content-assist to select from defined credentials.

### API Server ID

The ID of an API Server tied to this repository. Required to enable simplified webhook registration and automated commit and Pull Request activities. Use content-assist to select from defined API servers.

### MDGIT Clone 1 Job

The primary MDGIT service job number that should be used to perform Git activities for the repository. The number may not be higher than the maximum defined for the MDGIT service. It is useful to have multiple MDGIT jobs if MDCMS connects to multiple repositories.

### MDGIT Clone 2 Job

The secondary MDGIT service job number that should be used to perform Git activities for the repository. This is used in conjunction with the Clone 1 Job to provide high availability for Git activities when multiple branches for the same repository are involved. The number may not be higher than the maximum defined for the MDGIT service.

### MDGIT Clone 3 Job

The tertiary MDGIT service job number that should be used to perform Git activities for the repository. The number may not be higher than the maximum defined for the MDGIT service.

### URL Value in Webhook

When the MDCMS webhook is triggered after a commit push, MDCMS maps the URL in the webhook request body to a repository ID. If the URL in the request body differs from the URL used by MDCMS to clone or sync with the origin, enter the webhook URL value here. The correct value can be obtained from the Git Webhook Deliveries entry after the webhook has been triggered at least once.

### Main Branch

The name of the primary branch for the repository. Typically `main` or `master`, though it may differ based on team conventions. This branch is used for cross-referencing and is the default branch for other operations.

---

## After Saving

After saving the repository definition, the MDGit service on the IBM i will attempt to connect to the Git server using the specified credentials.

If the connection fails, click the **Connection Logs** icon for the Repository ID and open today's log for the correct MDGIT job number to review the exception details. If no log is available, the MDGIT service job did not auto-start — review the MDGIT service job settings such as job queue and auto-start configuration.

If the connection succeeds, MDCMS will automatically clone the main branch. A re-clone can be triggered at any time by selecting **Re-Clone Main Branch** for the Repository ID.