# Source Control – Github Integration

Bharath RadhaKrishna.

Informatica™

# Agenda

- Use cases
- Source control Actions.
- Source control configuration.
- Monitoring source control logs.
- Demo
- Best Practices.
- IICS Asset Management CLI V2 Utility

# Use case:

- Backup Management
- Source control Management
- Rollback
- Promotion of assets across environments.

# Source control Actions.

- Pull an object. Pull an object to add it to the organization or update the object in the organization with the version in the source control repository.

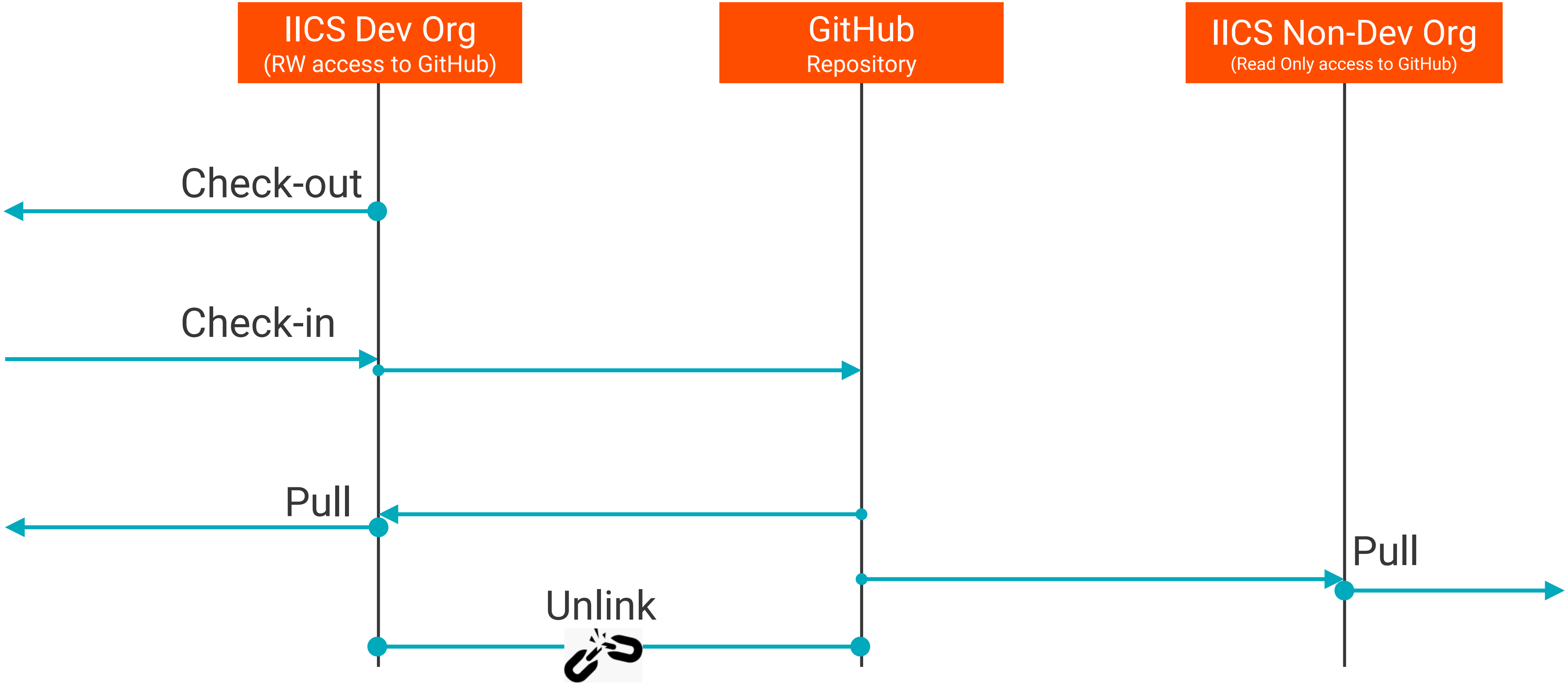
- Check out an object: Check out an object that you want to work on. When you check out an object, the object locks so that other users cannot make changes to it.

- Check in an object: Check in an object to add it to the source control repository or update the source control repository with the latest version of the object.
  When you check in an object, the lock releases.

- Delete an object: Delete an object from the organization and the source control repository.
  Before you can delete an object, you must check it out.

- Restore an object version: Restore an object to a previous version.
- Undo a check out: Undo a check out if you don't want to save the changes you made to the object.
  When you undo a check out, a new version of the object isn't created in the source control repository and there is no record of the check out in the object's source control history.

# IICS Source Control Actions Flow



# Source control configuration.

- Users can configure source control for an org to enable version management for projects/folders/assets.

- Objects are not checked in automatically, users need to apply source control to 1 or more assets/projects or folders.

- Source control license is required for an org to configure source control in IICS. Contact us if you need the license.

- Configure read/write access to source control repository.
  When read/write access is configured, users in org can check-in or check out objects, pull version of objects and revert objects to previous version.

- Configure read only access.
  Users in the org can only pull version of objects from Repository. Here users cannot check-in or check-out objects.

- Change Repository access type.
  Change from read/write to read-only, users must first ensure that no objects are checked out. IICS does not allow to change the repository access type from read/write to read-only if any objects are checked out.

- Change repository URL.
  To do this, users must first unlink all source-controlled assets. IICS does not allow to change the repository URL if any assets are source-controlled.
  Administrator> settings> edit.
  Enter the new url for ex: https://github.com/Bkrishna/demo (The repository url must use HTTPS protocol.)

# Source control configuration.

- Disable source control.
  Users must unlink the objects from source control and then disable read/write access to source control repository.
  Administrator> settings> click on Edit in source control and disable **Allow push to git repository.**
  Click on save.

- Source control configuration for sub orgs.
  Source control for sub org can be configured on the settings tab in Sub org. It is recommended to have own source control repository for sub org.
  Note: Source control repository for sub org should be different from source control repository for parent org.

-

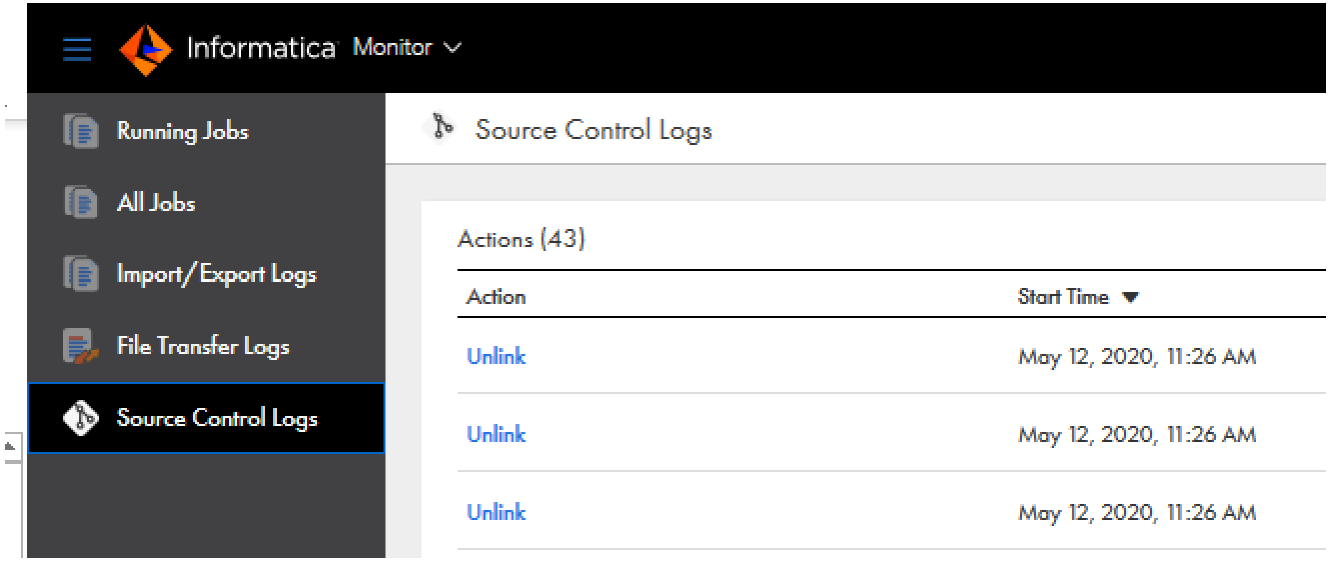
# Monitoring source control logs.

- Users can monitor the actions in an org performed on source controlled objects. IICS logs the actions each time an user performs an action on object. (Logs are retained for seven days.)

- To monitor source control actions, users need Asset - Source Control Logs privilege.
  Users having the below roles have it by default.

  Administrator
  Designer
  Monitor.

-

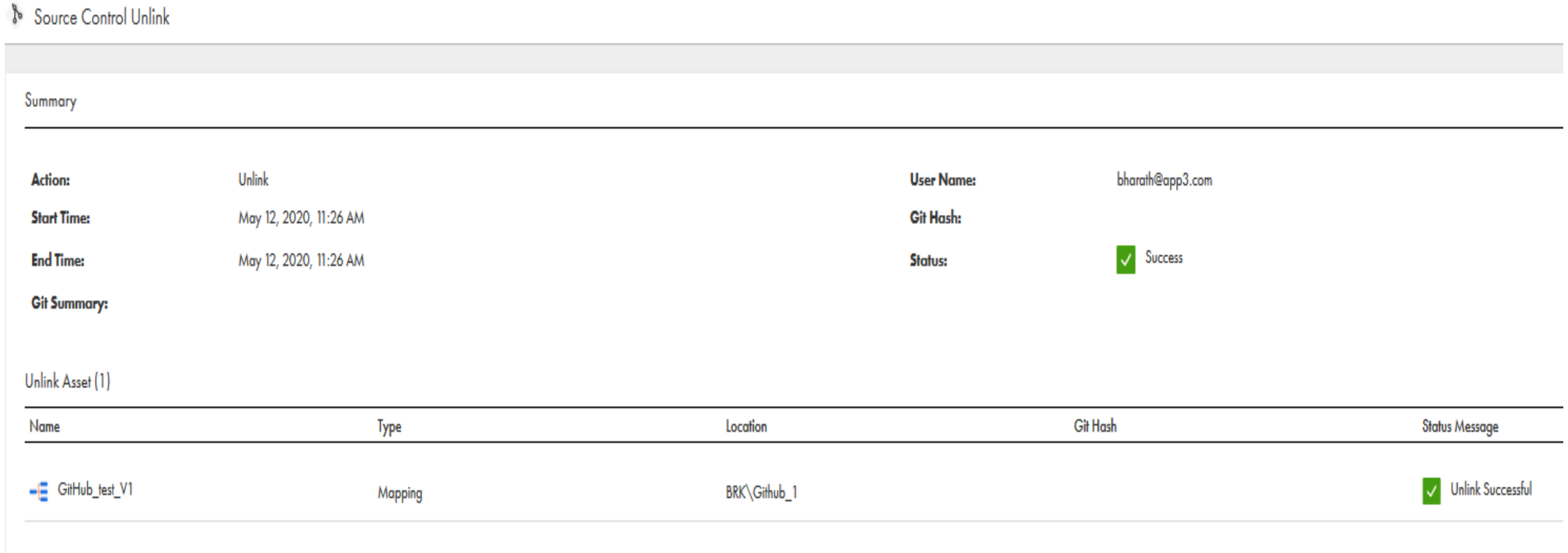


Actions (43)

| Action | Start Time ▼ |
|---|---|
| Unlink | May 12, 2020, 11:26 AM |
| Unlink | May 12, 2020, 11:26 AM |
| Unlink | May 12, 2020, 11:26 AM |

# Monitoring source control logs.

- To view detailed information about source control action> click on action column.



Source Control Unlink

Summary

| | | | |
|---|---|---|---|
| **Action:** | Unlink | **User Name:** | bharath@app3.com |
| **Start Time:** | May 12, 2020, 11:26 AM | **Git Hash:** | |
| **End Time:** | May 12, 2020, 11:26 AM | **Status:** | ✓ Success |
| **Git Summary:** | | | |

Unlink Asset (1)

| Name | Type | Location | Git Hash | Status Message |
|---|---|---|---|---|
| GitHub_test_V1 | Mapping | BRK\Github_1 | | ✓ Unlink Successful |

- Action – Performed in source control repository.
  Start time/End time.
  Git summary – Description (Applicable for check-in and pull)
  Username: User who performed this action.
  Git hash – Git identifier.
  Status:
  Success – Action completed successfully.
  In progress – Action is still in progress.
  Warning – Completed with warnings.
  Error – Action failed because of an error.

# Source Control Management

## Native GitHub Integration with IICS

- Using GitHub cloud hosted repository to version IICS assets
- Plug and play.
- Clean design and simple to use from IICS UI directly

## IICS Asset Management CLI

- Need to use my existing repository and not to use GitHub cloud repo
- Setup your local workspace to extract assets to/from IICS

# Source control- Github DEMO

# Best Practices.

- Github Repositories on Cloud are supported currently.
- Before pulling assets from GitHub, ensure that connections and runtime agents are created in the org.
- Recommendation is to maintain different Test, staging and production orgs.
  When you maintain different organizations, you maintain isolation across environments so that changes in one environment do not affect other environments. For example, changes to assets in the testing environment are not accidentally deployed in the production environment.
- Dev Org can be configured for Read/write.
- Prod/Test org can be configured for Read only.
  This ensures that only users in a development organization can make changes to assets. It also prevents users in a non-development environment from accidentally pushing changes to the source control repository.
- Ensure that only one development organization uses a particular source control repository.
  Maintaining separate repositories ensures that users in one organization do not accidentally change or overwrite assets in a different organization.
- When you enable source control for the organization, select an empty repository.
  Ensure that the repository does not contain a folder named "Explore" because Informatica Intelligent Cloud Services stores assets under the Explore folder in the Git repository.

# Best Practices.

- Do not share source control credentials among users in IICS.
  Separate credentials maintain security and makes it easier to track which user made a change.

- Create connections and ensure runtime environment exists before pulling assets from Github.

- Avoid moving or renaming source controlled assets that are used by other assets.
  When users move or rename a source controlled asset, references to the asset can break. Users cannot perform an undo checkout on a asset that has been moved or renamed.

- Enter the comments when users are checking in an asset.
  Users can enter a release name in the summary field and enter more descriptive comments in the description field.
  The git summary field in IICS will show the release name that is associated with the asset.

- Checking – in multiple assets at one time.
  When checking in multiple assets, try limiting the number of assets to 1000 or fewer. Checking in more than 1000 assets at one time can decrease the performance between IICS and Github.

# IICS Asset Management CLI V2 Utility

- https://network.informatica.com/docs/DOC-18245

# IICS Helpful links.

- Whitepaper: https://network.informatica.com/docs/DOC-18438
- Source control configuration: https://docs.informatica.com/integration-cloud/cloud-platform/current-version/administrator/source-control-and-service-upgrade-settings/source-control-configuration.html
- Source control Asset Management. https://docs.informatica.com/integration-cloud/cloud-data-integration/current-version/asset-management/source-control.html

# Thank You