knowledge16

Experience the Service Revolution



Update Sets Best Practices

JASON LEWKO
SOLUTION CONSULTANT
ServiceNow

ERICH DOBROTH
SENIOR SOLUTION CONSULTANT
ServiceNow

© 2016 ServiceNow All Rights Reserved

servicenow

Agenda

Update Sets

Merging Update Sets

Transferring Update Sets

Best Practices

Related Functionality

© 2016 ServiceNow All Rights Reserved

Update Sets

Update Sets are containers for capturing your configuration changes and moving them between instances.

© 2016 ServiceNow All Rights Reserved

Update Sets

Overview

- Update Set Definition
- The Release Process
- Update Set Lifecycle
- Update Sets Contents
- External Data
- Update Set Structure

© 2016 ServiceNow All Rights Reserved

Update Sets

Update Set Definition

- An Update Set is a container for capturing your configuration changes
- Update Sets are used to move your configuration changes from one instance to another
- Update Sets can be used to develop and deploy new applications as well as configure existing applications
- Update Sets are distinct/different/not-the-same-as
  - Applications published on the App Store (a different method for addressing same topic)
  - Import Sets (for data, not configuration)
  - Team Development (uses Update Sets)

© 2016 ServiceNow All Rights Reserved

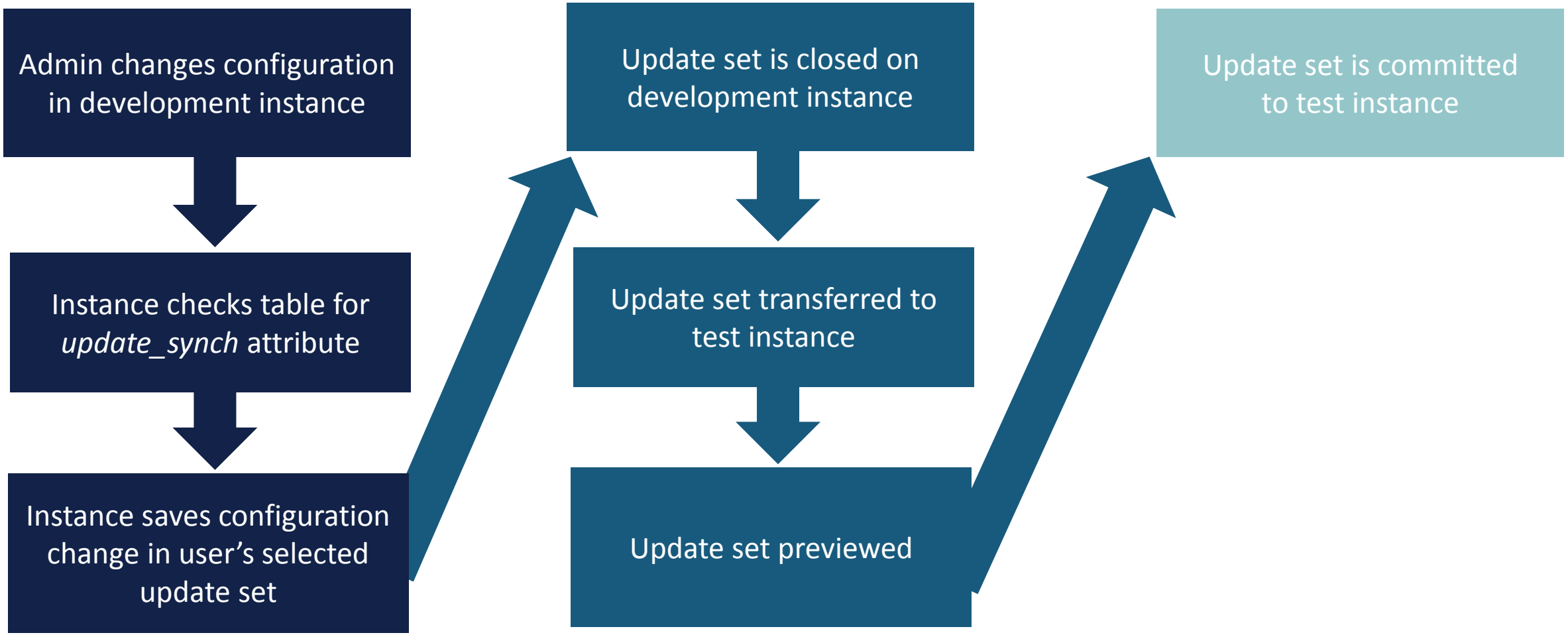
Update Sets

The Release Process

- A Release is defined to be one or more update sets comprising a new app or update to existing app, intended to be released/deployed to PROD as a single unit
- DEV -> TEST -> PROD is typical stack
  - All configuration changes should be done on DEV
  - When a configuration change is complete it is moved to TEST via update set(s)
  - Once testing is complete, a configuration change is moved to PROD via update set(s)
- Release process is the process that controls the movement of configuration changes from DEV to TEST to PROD
  - Best defined on a per-stack basis, to suit needs of those who maintain and depend on the stack

© 2016 ServiceNow All Rights Reserved

Life of a Configuration Change



© 2016 ServiceNow All Rights Reserved

Update Sets

Update Set Lifecycle

- Two Types of Update Sets
  - Local Update Set (sys_update_set)
    - Used to capture and represent instance configuration changes
    - Created explicitly by user on DEV or as a result of committing Retrieved Update Set
  - Retrieved Update Set (sys_remote_update_set)
    - Artifact associated with moving update set into a new instance
    - Serves as container for previewing and committing
    - Committing creates associated Local Update Set (which represents the applied configuration changes)
    - Created when moving update on target instance

© 2016 ServiceNow All Rights Reserved

Update Sets

Update Set Lifecycle (cont.)

- Local Update Set States
  - In Progress: Update set can be used to capture configuration changes
  - Complete: Update set's configuration changes are ready to be moved to another instance
  - Ignore: Update set contains configuration changes that are not intended to be moved
- Remote Update Set States
  - Loading: Update set is in the process of being retrieved from source instance
  - Loaded: Update set has been retrieved from source instance
  - Previewed: Update set has been previewed and any problem must be addressed before committing
  - Partially Committed: Update set is committing (configuration changes are being applied to instance)
  - Committed: Update set has been committed (configuration changes have been applied to instance)
  - Backed Out: Update set has been backed out (associated local update set has been deleted)

© 2016 ServiceNow All Rights Reserved

Update Sets

Update Set Contents

- The Basics
  - Collection of configuration changes made when update set was specified as current
  - In Progress Update Sets capture current changes
    - User chooses their current update set via Update Set Picker (gear in upper-right banner)
  - Complete Update sets are closed and ready to be transferred to another instance
  - Update Sets capture metadata (data controlling the behavior of the app) such as:
    - Table, form, fields definitions
    - Business rules, client script, UI policy definitions
    - Views, label changes, section definitions
  - Non-metadata is not captured as updates in update set
    - For example, INT, CHG, or PRB records are not captured in an Update Set

© 2016 ServiceNow All Rights Reserved

Update Sets

Update Set Contents (cont.)

- The following are not automatically saved to Update Sets
  - Attachments
  - Schedules
  - Scheduled Jobs
  - Users
  - Groups
  - Role Assignments
  - Incident, problems, changes, (i.e. any row-level data)

© 2016 ServiceNow All Rights Reserved

Update Sets

External Data

- Releases often depend on some data that is not automatically captured in Update Sets
  - Groups with associated Roles
    - But Users in Groups should be specified post-install as roles are applied dynamically
  - Scheduled Jobs
- Two common methods for handling this
  - Create data manually on each instance as it moves up the stack
  - Add data to update set

© 2016 ServiceNow All Rights Reserved

Update Sets

External Data (cont.)

- Methods to manually add data to an update set
  - Fuji+: If working in an application, ‘Create Application File’ list view action
  - Pre-Fuji: See here (use judiciously)
  - Generally less brittle than alternative, manual process
- Alternatives (outside of an Update Set)
  - Export data using Export XML and import using Import XML
  - Requires more disciplined dev/release process

© 2016 ServiceNow All Rights Reserved

Update Sets

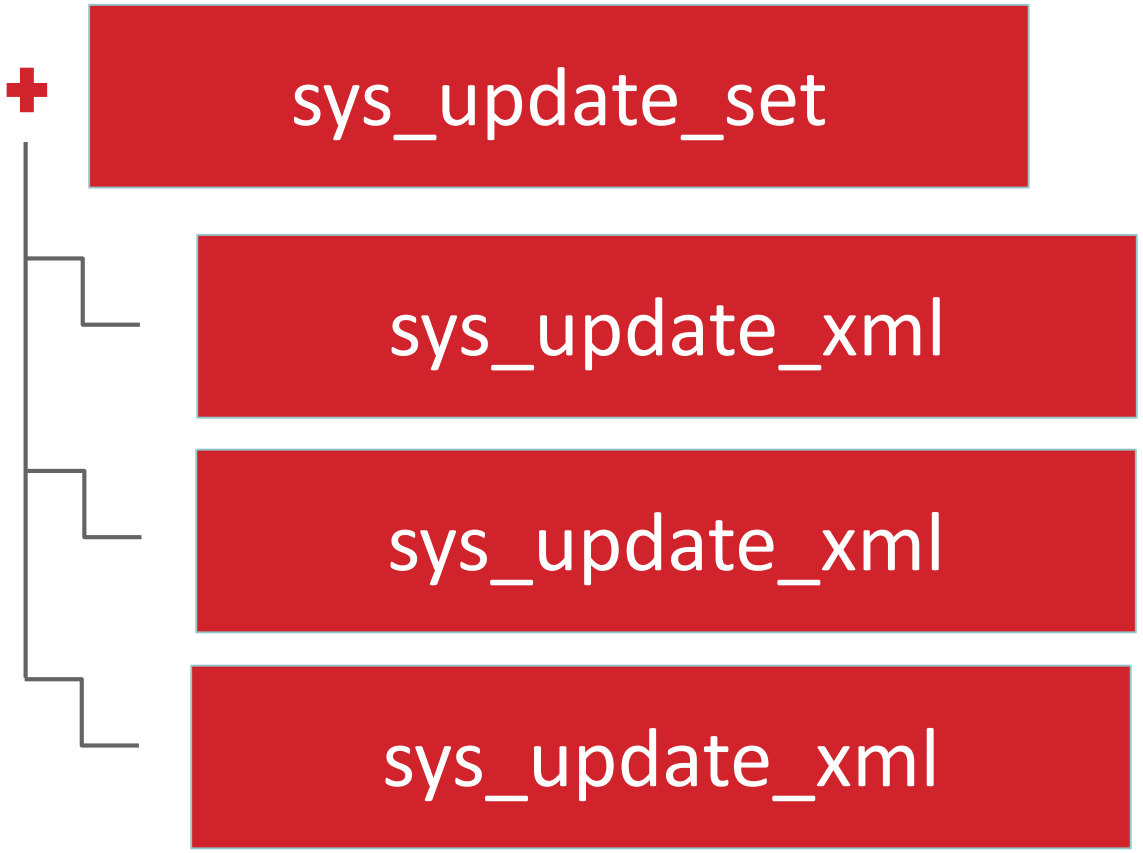
Update Set Structure

- Update Set Contains Customer Update (sys_update_xml) records
- Scope of a given sys_update_xml record usually maps 1:1 to item being customized, but there are some types that have a broader scope (to ensure consistency)
  - Choice Lists
  - Form Sections
  - Dictionary Entries and Field Labels
  - Lists and Related Lists
  - Workflows

© 2016 ServiceNow All Rights Reserved

Update Sets

Update Set Structure (cont.)



sys_update_xml payload structure

<?xml version="1.0" encoding="UTF-8”>
<record_update table="sys_db_object">
<sys_db_object action="INSERT_OR_UPDATE">
<create_access_controls>false<create_access_controls>
<is_extendable>false</is_extendable>
<label>AIX Server</label> … etc

© 2016 ServiceNow All Rights Reserved

Merging Update Sets

Merge Update Sets to consolidate configuration changes contained in multiple sets. Merging just prior to transferring to another instance simplifies the transfer process.

© 2016 ServiceNow All Rights Reserved

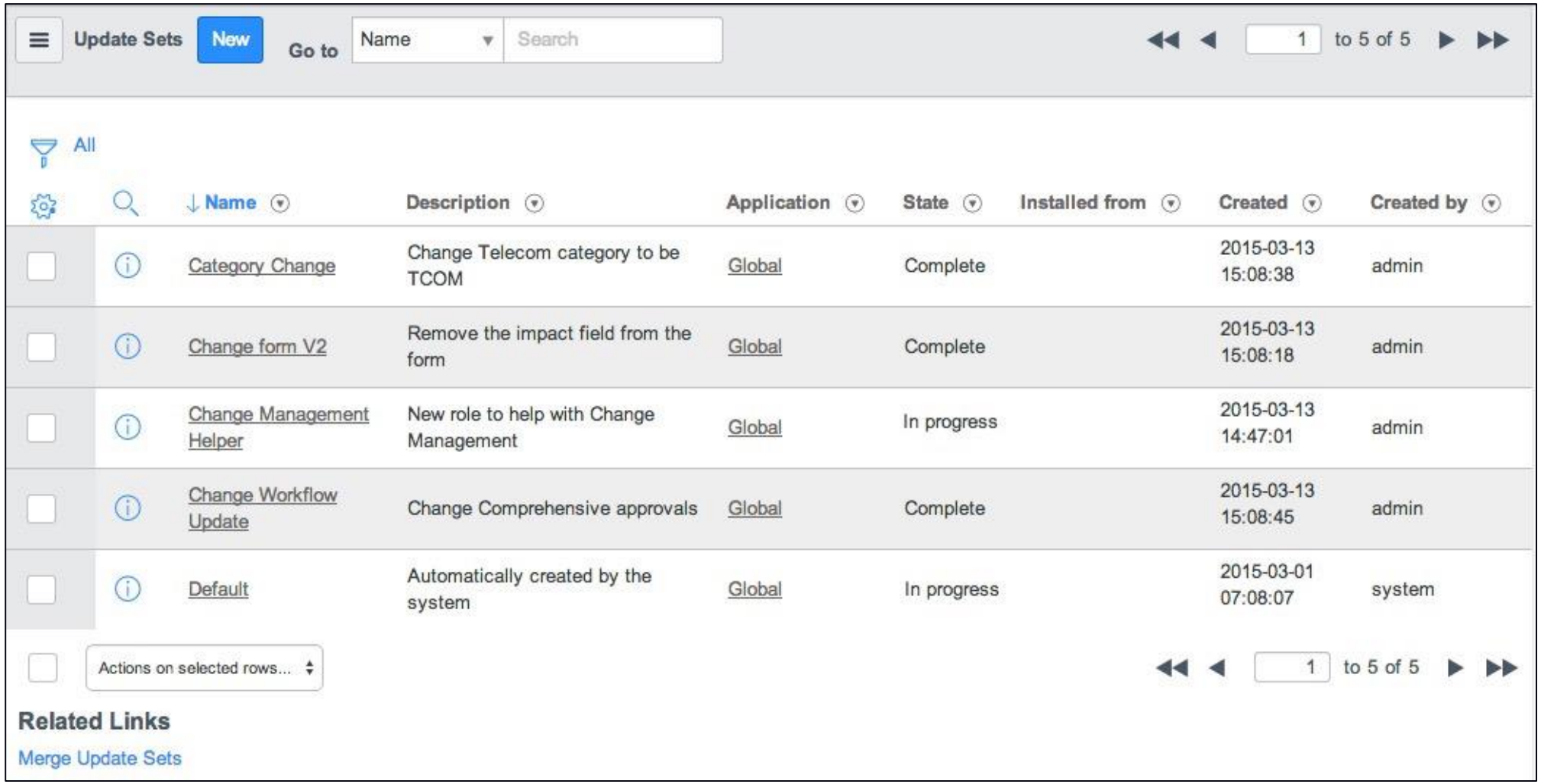
Merging Update Sets

When, Why, How

- When?
  - Prior to moving multiple Update Sets from one instance to another
- Why?
  - Simplifies Migration
  - Reduces the amount of update sets to preview and commit
  - Reduces chance of update set conflict
  - Removes the need to know in which order to apply update sets
- How?
  - Using the Merge Modules, or the Merge Link in the Local Update Set List

© 2016 ServiceNow All Rights Reserved

Merging Update Sets – When, Why, and How



© 2016 ServiceNow All Rights Reserved

Transferring Update Sets

How to move an update set from one instance to another and then commit the changes.

© 2016 ServiceNow All Rights Reserved

Transferring Update Sets

Overview

- Transfer update set from source to target instance
  - Creates Retrieved Update Set
- Preview Retrieved Update Set
  - Identifies problems – all must be addressed before proceeding
  - Deleting Retrieved Update Set
- Commit Retrieved Update Set
  - Creates Local Update Set and applies configuration changes to the target instance
- Backing Out Update Set
  - Removes configuration changes associated with committed update set

© 2016 ServiceNow All Rights Reserved

Transferring Update Sets

Transfer Methods

- XML File
  - ‘Export to XML’ creates file containing update set (from source instance)
    - Update set must be in Complete state
  - ‘Import Update Set from XML’ loads update set into target instance
    - This is a Related Link on the ‘Retrieved Update Sets’ module
- Update Source
  - Define Update Source on target instance (once)
    - Specifies source instance from which to pull complete update sets
  - ‘Retrieve Completed Update Sets’ loads all Completed update sets from source instance
    - Transfer is performed without user having to create file

© 2016 ServiceNow All Rights Reserved

Transferring Update Sets

Previewing Retrieved Update Sets

- Previewing an Update Set detects problems that may occur if you commit the updates on the local instance – you must address all problems before you can commit the update set
- Preview an Update Set
- Open the update set record and look at the Update Set Preview Problems related list
- Review a problem description to determine the cause
- Resolve the problem or chose one of the available options
  - Accept remote update: Try and commit the update without fixing the problem
  - Skip remote update: Skip the update, and proceed without it

© 2016 ServiceNow All Rights Reserved

Transferring Update Sets

Previewing Retrieved Update Sets (cont.): Problem Types

- Missing object: The object or a referenced object does not exist on the target instance.
- Type mismatch: The type of a column on the target instance does not match the type in the update set.
- Collision: A change in the update set is older than a change to the same object on the local instance.
- Uncommitted update: The object exists in another remote update set that has not been committed.

© 2016 ServiceNow All Rights Reserved

Transferring Update Sets

Previewing Retrieved Update Sets (cont.): Deleting Retrieved Update Set

- If preview problems indicate that the update set was not ready to be committed, you can delete the update set instead of attempting to address the problems
  - Delete the retrieved update set (before committing)
  - Go back to the source instance and re-open update set to capture needed changes
  - Close and move to test when ready

© 2016 ServiceNow All Rights Reserved

Transferring Update Sets

Backing Out Update Sets

- Removes Update Set configuration changes
- ‘Back out’ button on Local Update Set
  - Prior to Fuji, available only on most recently committed Update Set
  - Update Set and associated Update Records are delete
    - The Retrieved Update Set is not deleted
- Use sparingly
  - If the Update Set deleted a table, back out will restore the table, but not its data
  - Backing out an update set can require backing out associated changes contained in subsequent Update Sets
  - Frequent use of back out indicates a poor development and/or release process

© 2016 ServiceNow All Rights Reserved

Best Practices

Practices that lead to a quality development and release process.

© 2016 ServiceNow All Rights Reserved

Best Practices

Practices that lead to a quality development and release process

- Always move configuration changes from the bottom of the stack up
  - Ensures down-stack instances match up-stack instances
  - Configuration changes introduced mid-stack can be overwritten by future pushes from down-stack
  - Common scenarios include
    - Fixes needed in test or prod – always push them from dev up
    - Common Prod admin configuration change such as choice lists – always push them from dev up
- Always review updates contained in an Update Set before transferring
  - Look for updates associated with other dev efforts and updates associated with testing
  - Watch for Sys Properties and Integration End-Points changes
    - Ex: Pushing a sys_property change that directs all email to a test email account
  - Move updates to scrap update set rather than deleting the update

© 2016 ServiceNow All Rights Reserved

Best Practices

Practices that lead to a quality development and release process (cont.)

- Always test after pushing
  - Ensure that all desired configuration changes were captured and applied as you expected
- For a release involving multiple developers, nominate a single developer to manage all aspects of the Update Set lifecycle
  - Responsible for creating, pushing, and merging update sets associated with dev effort
  - Responsible for communicating the current update set to rest of team
  - Responsible for reviewing update sets and communicating pushes to entire team
  - Responsible for documenting deployment process
    - Order of applying update sets, execution of scripts, and data that must be loaded

© 2016 ServiceNow All Rights Reserved

Best Practices

Practices that lead to a quality development and release process (cont.)

- Decide and use a consistent naming convention for Update Sets
  - Name should include: release id, short description, and set number, for example
    - ‘CHG1234567: On-Boarding App: Release 1: set 1’
- Limit who has admin on the development stack – use impersonation
  - Ensures un-intended configuration changes are not made by non-developers (e.g. doc, testers)
- When possible, favor shorter release cycles
  - Reduces quantity of in-flight work, ensuring better match between prod and dev
  - Simplifies work associated with re-cloning dev stack
- Consider backing up your important, in-development update sets
  - Easy to do, and provides piece of mind, esp. if portions of team are inexperienced

© 2016 ServiceNow All Rights Reserved

Best Practices

Practices that lead to a quality development and release process (cont.)

- In stacks with multiple, parallel releases, ensure communication and coordination between the development teams
  - Best to partition work between teams so that code overlap/dependencies are minimized
  - Changes to shared entities (e.g. task) need to be recognized/discussed at the design phase
    - Place shared configuration change into separate release that precedes dependent releases
  - Goal is to identify any implicit release sequence dependencies and surface to release process
    - Release B has changes on top or Release A, so Release A must be pushed ahead of Release B
- Consider timing of release deployment and upgrades
  - Prefer upgrading when fewer releases are in-flight
  - Prefer pushing from dev-to-test post-upgrade versus test-to-prod post-upgrade
  - Prefer upgrading when all in-flight update sets are committed (versus loaded or previewed)

© 2016 ServiceNow All Rights Reserved

Best Practices

Practices that lead to a quality development and release process (cont.)

- Clone sub-prods at appropriate frequency
  - Ensures better match between prod and dev/test for data dependent behavior
  - Identify issues before they occur in prod
- Define a release process
  - Goal is to have a light-weight process that ensures quality releases to Prod
  - Fewer individuals involved in chain leads to higher throughput
  - Consider using CHG to manage post-dev-complete portion of release lifecycle
    - Use CHG task to document track required steps
    - Scales to organizations requiring review by Change Approval Board prior to Prod deployment
    - Including CHG in update set name links update with all the info associated with CHG

© 2016 ServiceNow All Rights Reserved

Best Practices

Practices that lead to a quality development and release process (cont.)

- Customize base-table functionally with care
  - Understand the scope of the change on extended tables and their apps
- Avoid experimenting on your dev instance
  - Your configuration change can be accidentally captured and migrated by other team members
- Don’t do your development in the Default Update Set
  - Undisciplined use of Update Sets results in missed configuration changes and unnecessary time spent locating and moving required sys_update_xml records into the correct update set
- Minimize manual steps to move configuration changes up-stack
  - Consider bundling non-auto-tracked updates (e.g. Groups) in update sets when appropriate
  - Prefer defining run-once deployment scripts in script include rather that passing via emails

© 2016 ServiceNow All Rights Reserved

Best Practices

Practices that lead to a quality development and release process (cont.)

- Customize out-of-box functionality wisely
  - Prefer configuration changes that extend OOB functionality where possible (so all updates associated with an upgrade will be applied)
  - Scoped apps helps to minimize this going forward

© 2016 ServiceNow All Rights Reserved

Related Functionality

App Creator

Team Development

© 2016 ServiceNow All Rights Reserved

App Creator

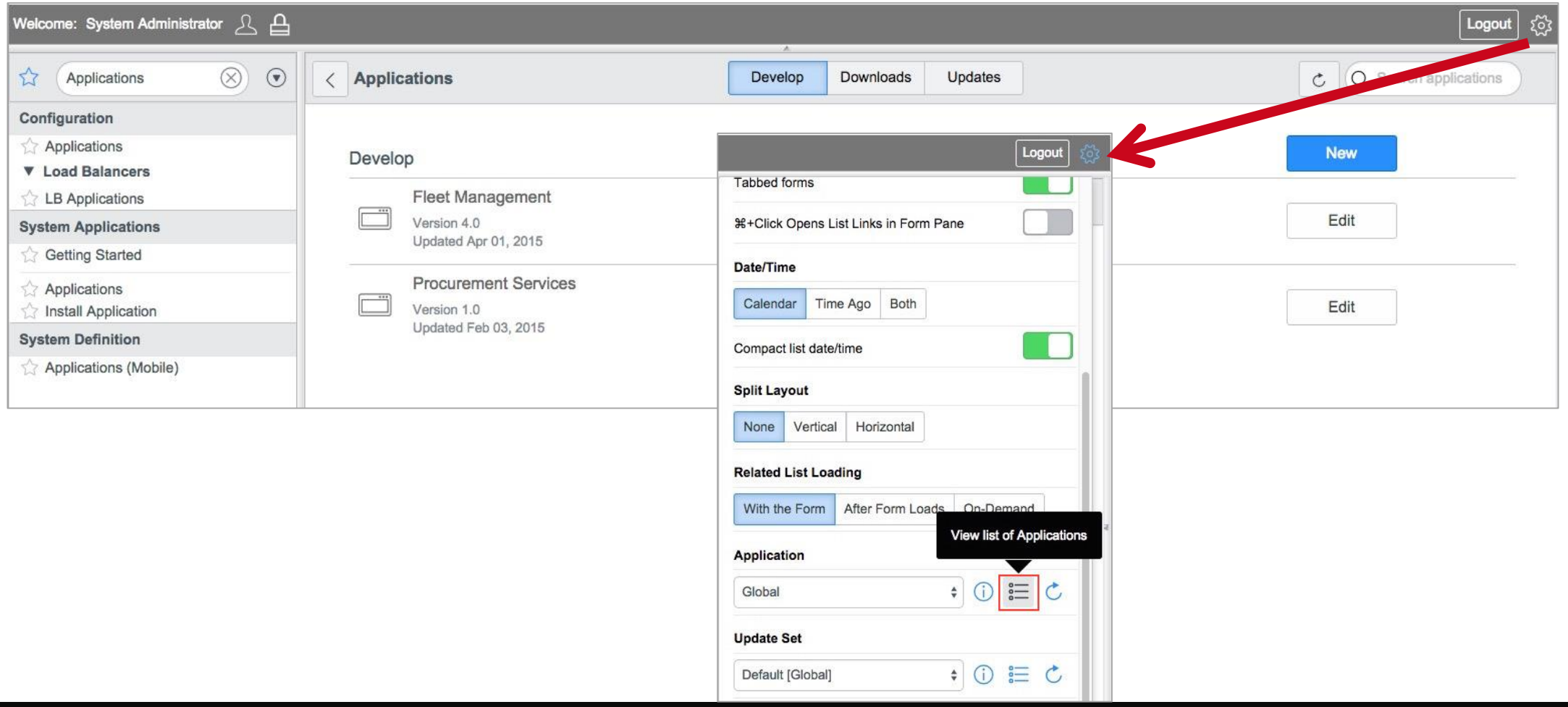
Overview

- Improved in Geneva
- You provide a definition of an application
- All configuration, data structures, and logic are bundled
- Can create an update set to transfer applications between instances
- You can publish applications to the ServiceNow store
- You can publish applications internally using the ServiceNow repository

© 2016 ServiceNow All Rights Reserved

App Creator

Creating a New Application

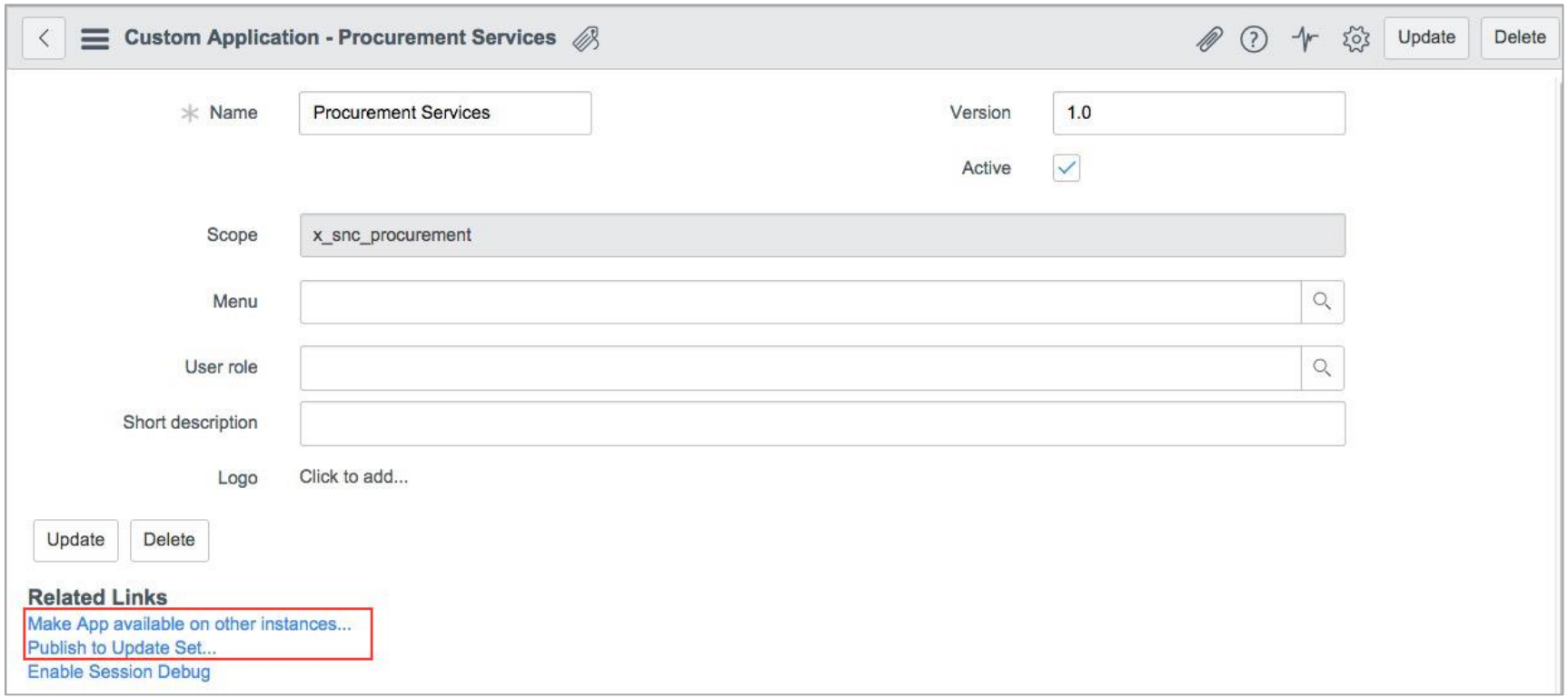


© 2016 ServiceNow All Rights Reserved

App Creator

Publishing an Application

- From the Application record, you can Publish to the AppRepository or to an update set

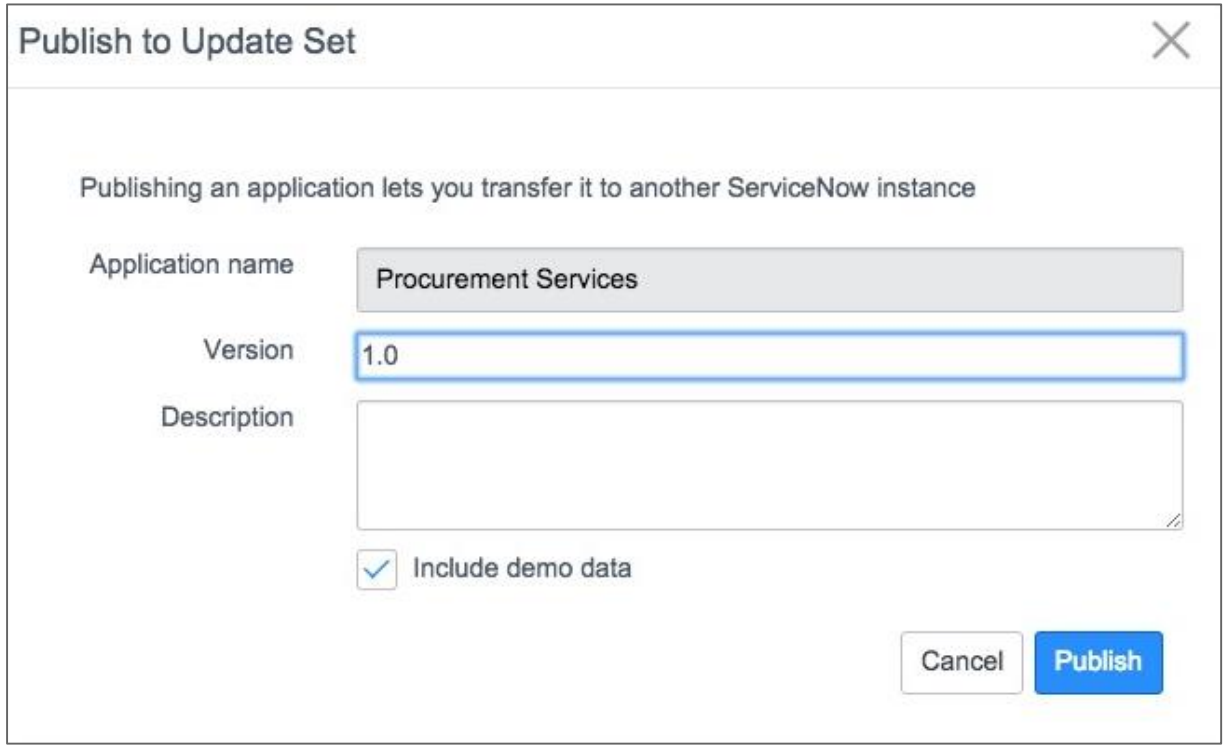


© 2016 ServiceNow All Rights Reserved

App Creator

Publish to an Update Set

- Produces an update set in Local Updates that you can then use to distribute your application



© 2016 ServiceNow All Rights Reserved

Past Functionality?

What if you've made configurations outside of an update set?

- Use Cases:
  - Built something as a proof-of-concept or during a demo and didn't capture everything in an update set
  - Continuing development on delayed or postponed work
  - Experiencing issues with missing components after promoting an update set or merged set

© 2016 ServiceNow All Rights Reserved

Team Development

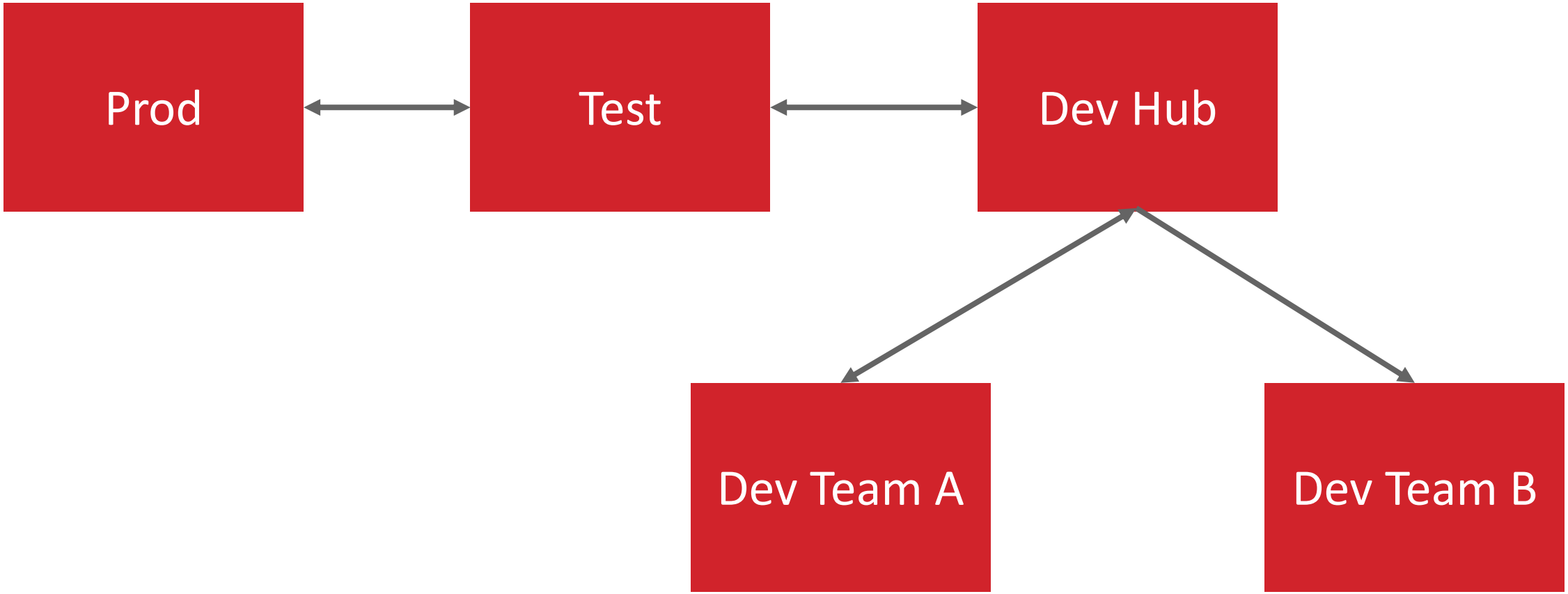
Overview

- Introduced in Dublin – active by default
- Branching operations, including pushing and pulling record versions between instances
- Ability to compare a development instance to other development instances
- Provides a central dashboard for all team development activities
- Use when you have multiple SN admins building separate functionality across the platform
- Uses Update Sets to move functionality between instances

© 2016 ServiceNow All Rights Reserved

Team Development

Instance Architecture



© 2016 ServiceNow All Rights Reserved

Top Takeaways

1. Have a strict DEV -> TEST -> PROD regime.

2. Promote update sets through all instances.

3. Don’t let production be the first time you’ve implemented your configuration changes!

© 2016 ServiceNow All Rights Reserved

Take the Survey

Please take a moment to complete a session survey in the Knowledge16 app.

Thank You

Jason Lewko

Solution Consultant

ServiceNow

Erich Dobroth

Senior Solution Consultant

ServiceNow

#Know16

© 2016 ServiceNow All Rights Reserved