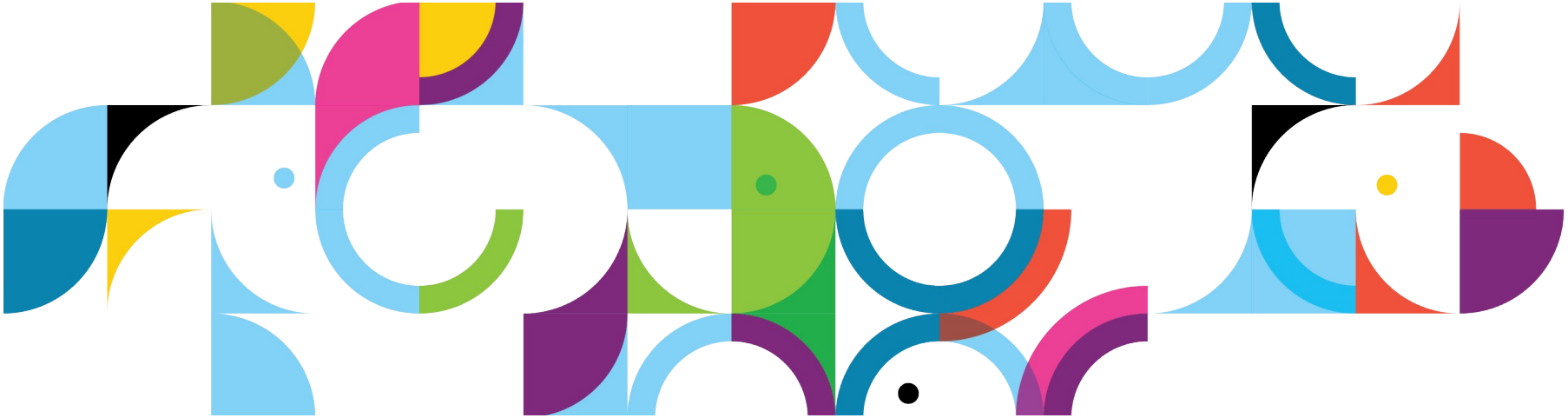
WebSphere Portal Migration: Best Practices Open Mic

13 March 2013

Joseph John | Portal Migration Development Lead

IBM Collaboration Solutions



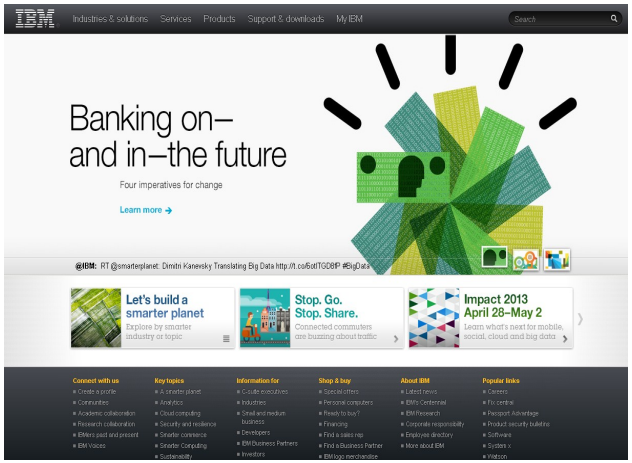
Agenda

- WebSphere Portal Migration
- Migration Walkthrough
- Migration Strategies
- Managed Pages
- Portal 8 Static Pages
- Resource Links
- Q&A

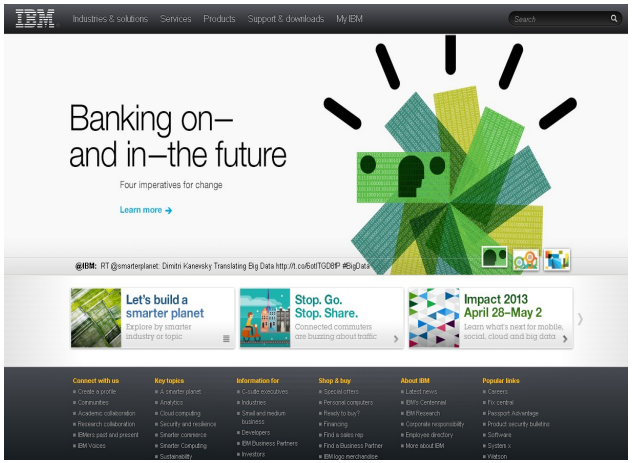
WebSphere Portal Migration

- The goal of Portal Migration is to move all content, applications, and pages from the source to target environment while maintaining the same look and feel of the original website.
  - To the end user, the website should look and be functionally the same.
  - New Portal features are installed, but not enabled.
  - Some applications and content may need to be manually updated to meet new specifications.

Before



After



WebSphere Portal Migration

- Migration can be performed locally or remote.
- Migration can be non-destructive and allow both the source and target to run simultaneously.
- WCM Support can be added to a Portal Server during migration.
- Migration is only supported from the two previous main versions.
  - v6.1 → v8 is supported
  - v7 → v8 is supported
  - v6 → v8 is not supported
- Migration is not cross platform.
  - v7 on Linux → v8 on Linux is supported
  - v7 on Windows → v8 on Linux is not supported
  - Platform changes should be done as a separate task before or after migration.
- Migration does not allow changing cell or node names.
  - Cell and node name changes should be done as a separate task before or after migration.

Before Starting

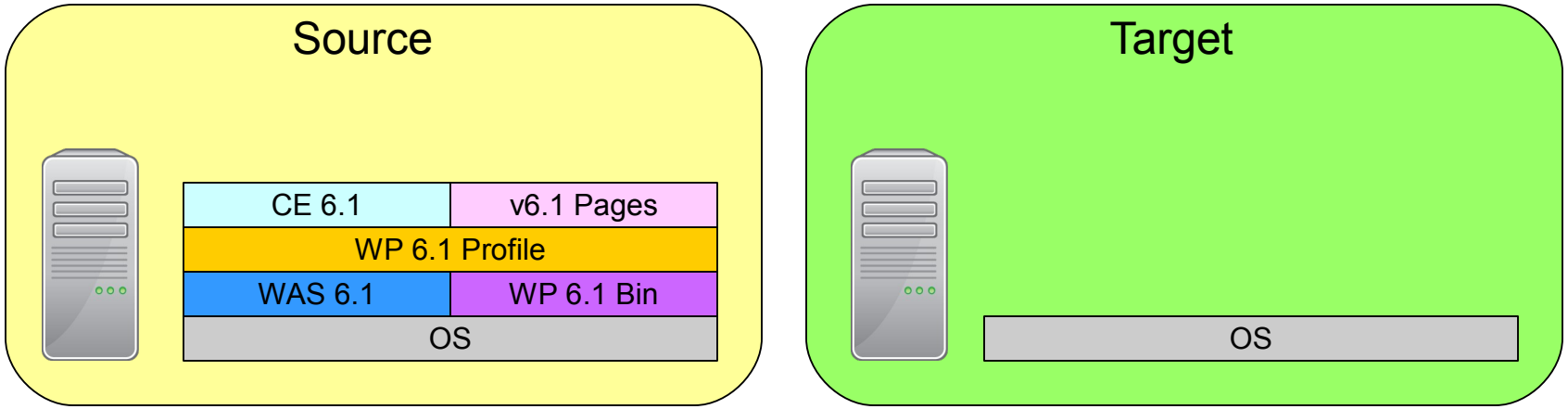
- Plan
  - Define complete for each environment
  - Create an iterative migration plan
  - Device a rollback plan
- Prepare
  - Scope required resources and time
  - Obtain necessary software and hardware
  - Determine which applications need to be updated and begin updating them
- Practice
  - Practice and become familiar with the process
  - Stage a migration practice run using an environment that mimics the production environment as closely as possible

Migration Walkthrough

- Part 1: Prepare source and target environment
- Part 2: Import source Portal Profile to target environment
- Part 3: Connect target environment to copies of the Databases
- Part 4: Upgrade the target Portal Profile
- Part 5: Post migration steps

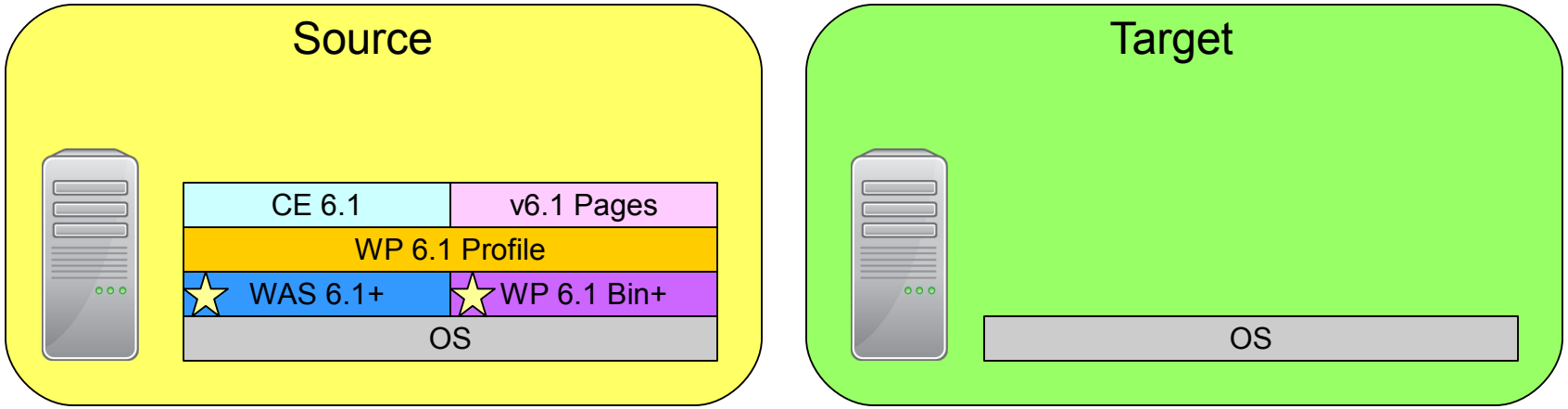
Prepare Source and Target Environment

- S: Apply latest fix pack
- T: Binary-only install
- T: Apply latest fix pack
- T: Create remote migration package
- S: Copy remote migration package
- S: Disable Automatic synchronization
- S: Prevent changes



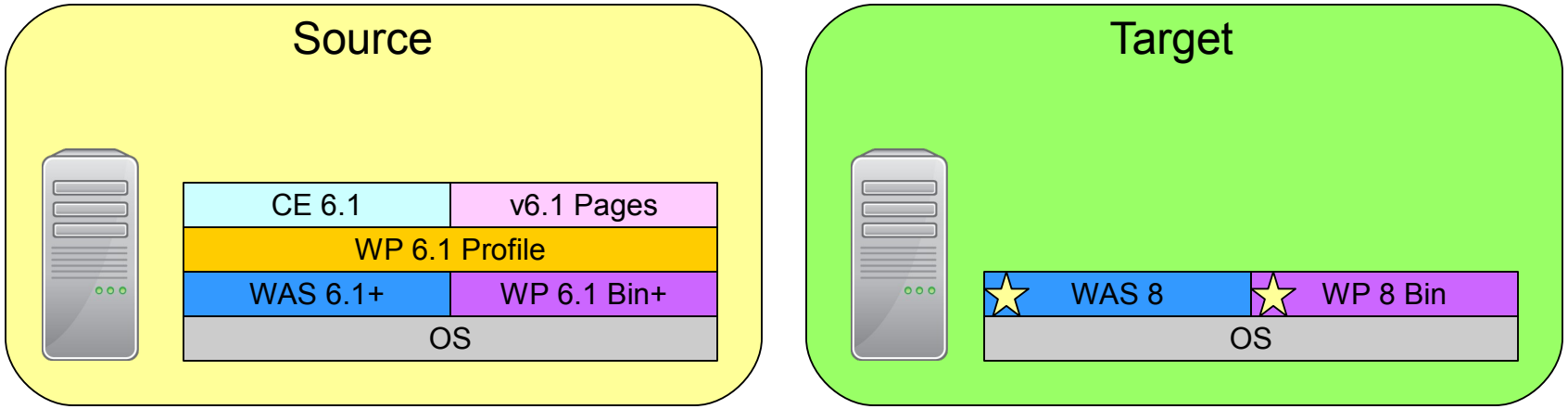
Prepare Source and Target Environment

- S: Apply latest fix pack
- T: Binary-only install
- T: Apply latest fix pack
- T: Create remote migration package
- S: Copy remote migration package
- S: Disable Automatic synchronization
- S: Prevent changes



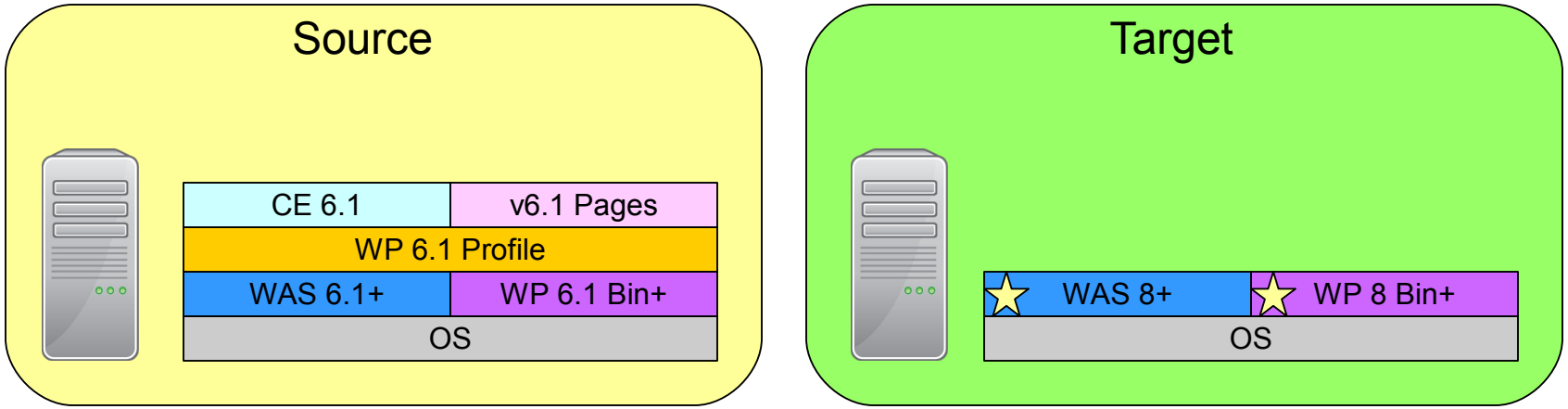
Prepare Source and Target Environment

- S: Apply latest fix pack
- T: Binary-only install
- T: Apply latest fix pack
- T: Create remote migration package
- S: Copy remote migration package
- S: Disable Automatic synchronization
- S: Prevent changes



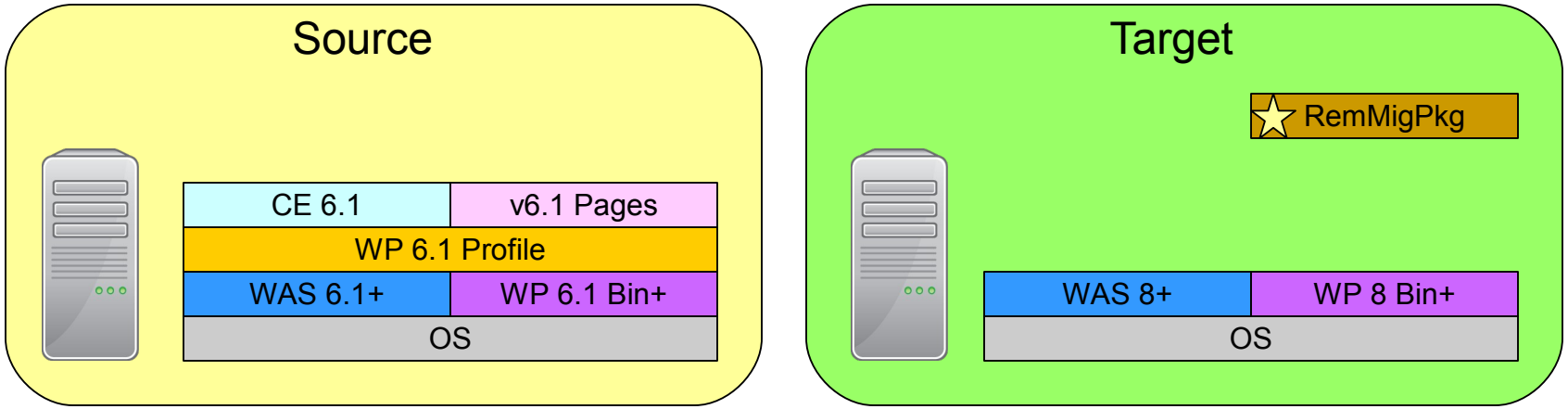
Prepare Source and Target Environment

- S: Apply latest fix pack
- T: Binary-only install
- T: Apply latest fix pack
- T: Create remote migration package
- S: Copy remote migration package
- S: Disable Automatic synchronization
- S: Prevent changes



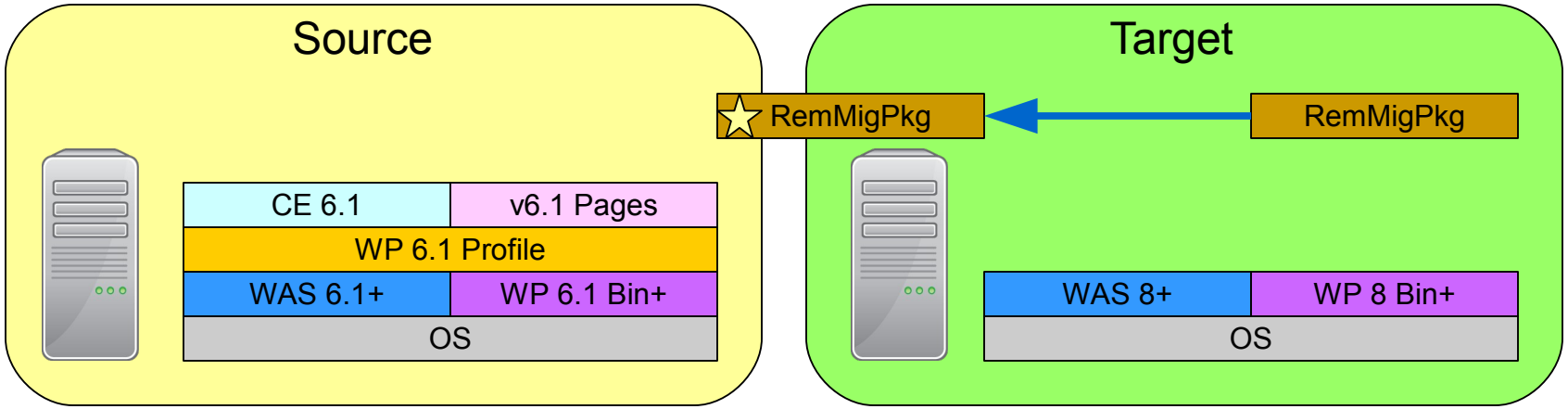
Prepare Source and Target Environment

- S: Apply latest fix pack
- T: Binary-only install
- T: Apply latest fix pack
- T: Create remote migration package
- S: Copy remote migration package
- S: Disable Automatic synchronization
- S: Prevent changes



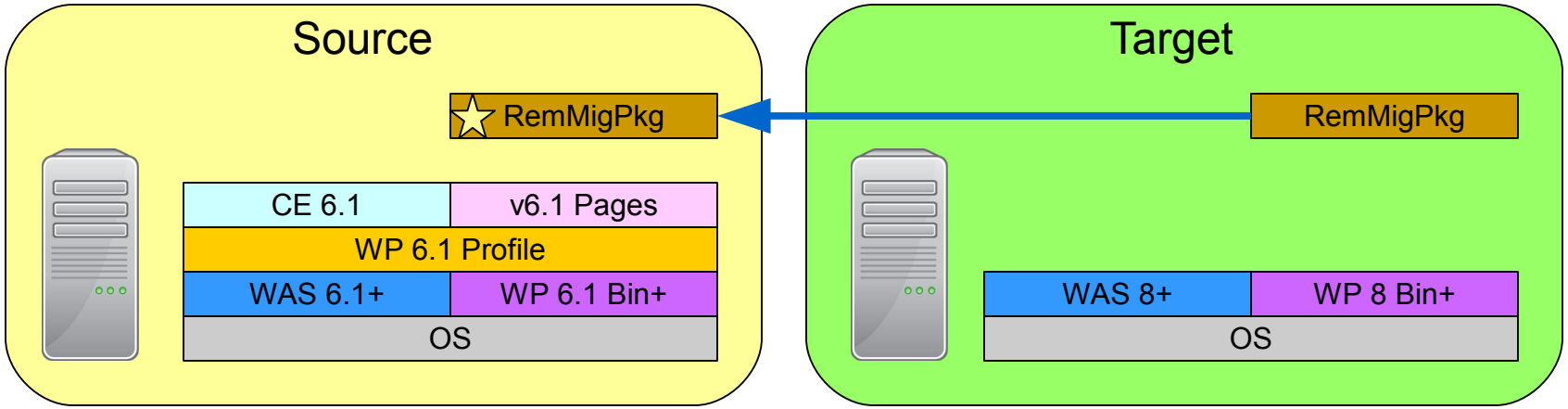
Prepare Source and Target Environment

- S: Apply latest fix pack
- T: Binary-only install
- T: Apply latest fix pack
- T: Create remote migration package
- S: Copy remote migration package
- S: Disable Automatic synchronization
- S: Prevent changes



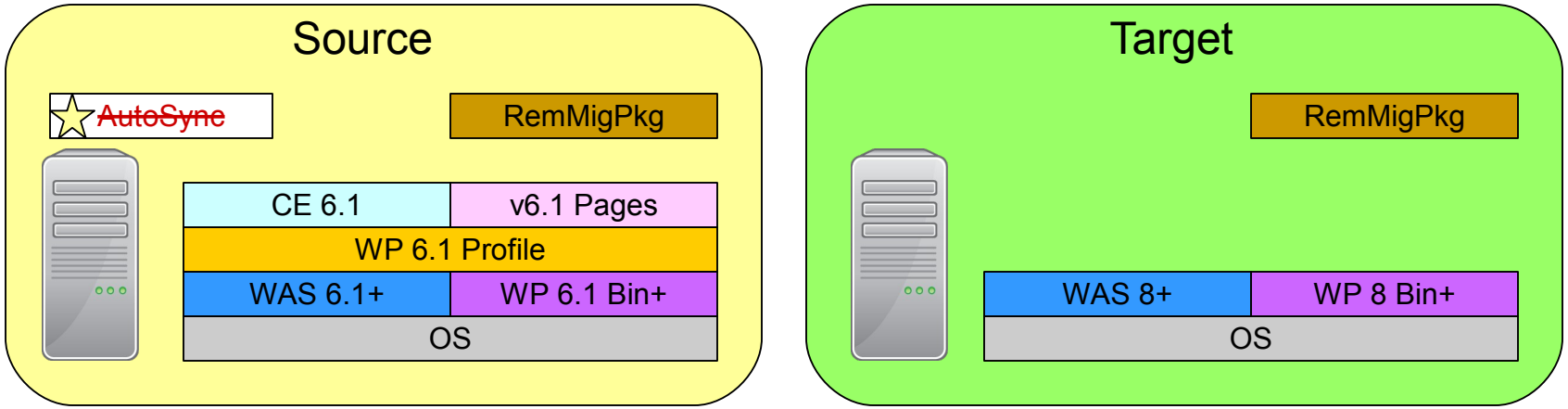
Prepare Source and Target Environment

- S: Apply latest fix pack
- T: Binary-only install
- T: Apply latest fix pack
- T: Create remote migration package
- S: Copy remote migration package
- S: Disable Automatic synchronization
- S: Prevent changes



Prepare Source and Target Environment

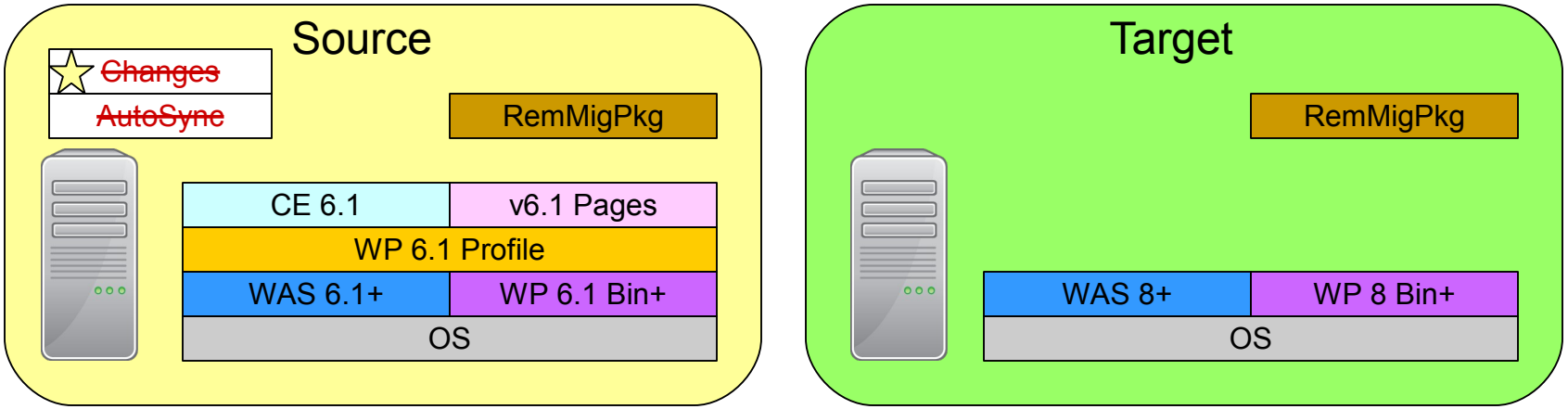
- S: Apply latest fix pack
- T: Binary-only install
- T: Apply latest fix pack
- T: Create remote migration package
- S: Copy remote migration package
- S: Disable Automatic synchronization
- S: Prevent changes



Prepare Source and Target Environment

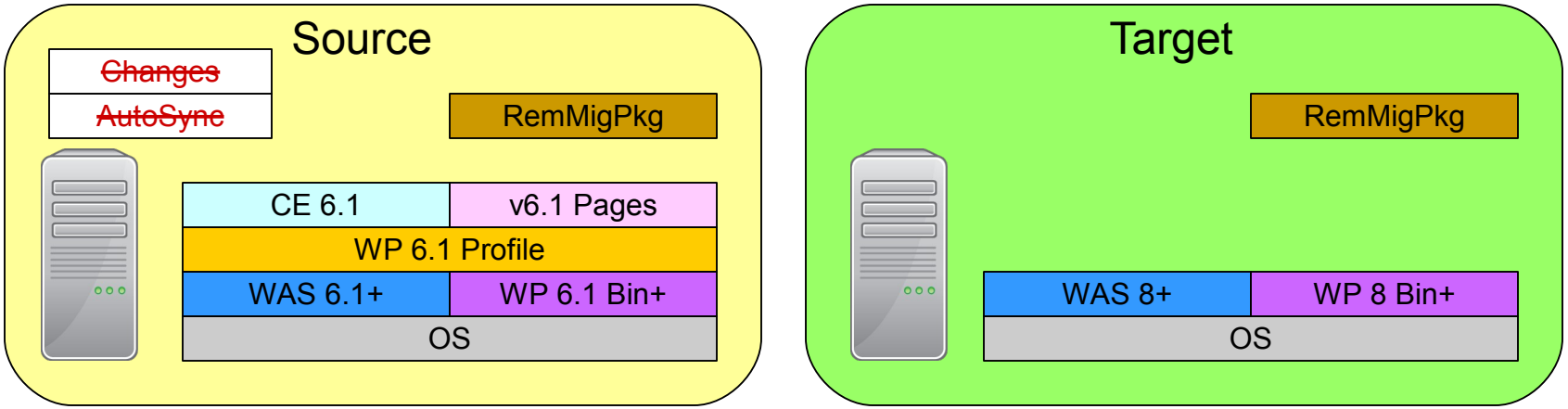
- S: Apply latest fix pack
- T: Binary-only install
- T: Apply latest fix pack
- T: Create remote migration package
- S: Copy remote migration package
- S: Disable Automatic synchronization
- S: Prevent changes

Refer to the wiki for additional preparation steps such as making backups, updating database statistics, and ensuring there is no locked content.



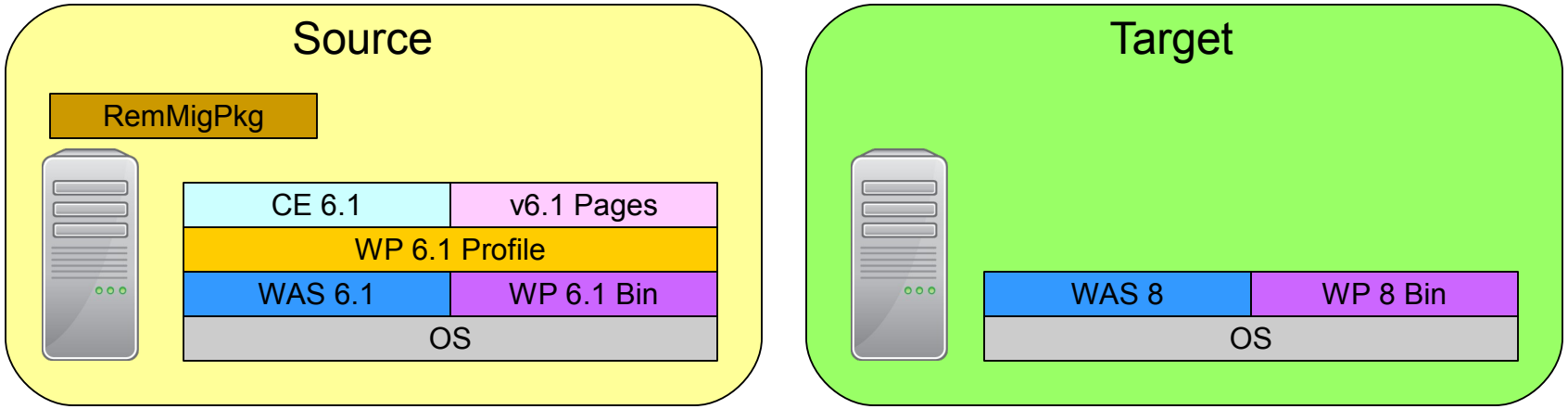
Best Practices

- Make sure to apply latest fix packs to source and target
- Disable AutoSync
- Limit changes in source environment



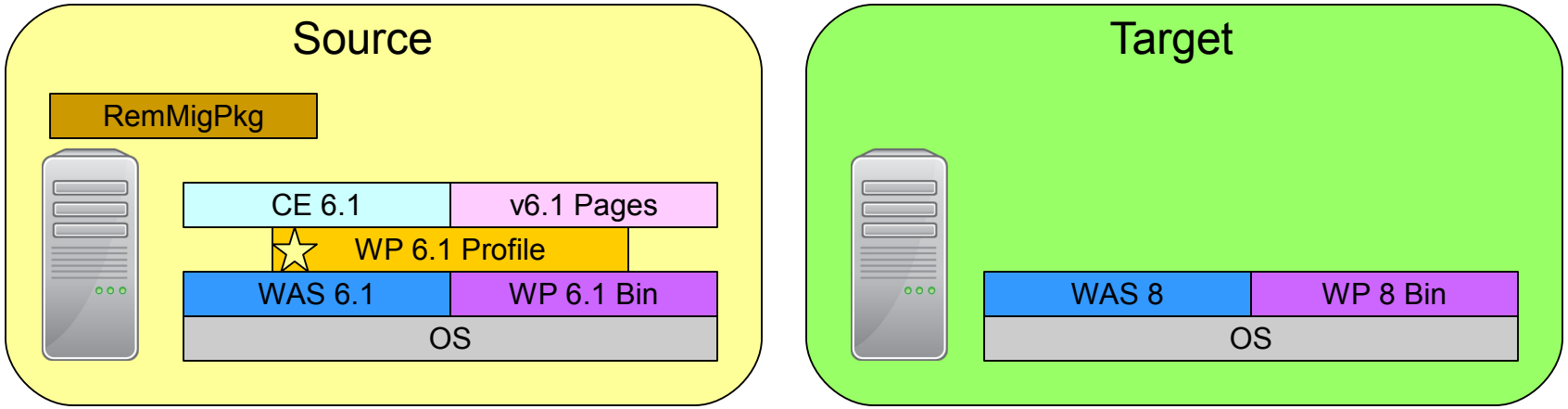
Import Source Portal Profile to Target Environment

- S: Export profile using `WASPreUpgrade`
- T: Copy exported backup profile
- T: Create a default profile using `manageprofiles`
- T: Import profile using `WASPostUpgrade`
- T: Upgrade ConfigEngine using `upgradeConfigEngine`



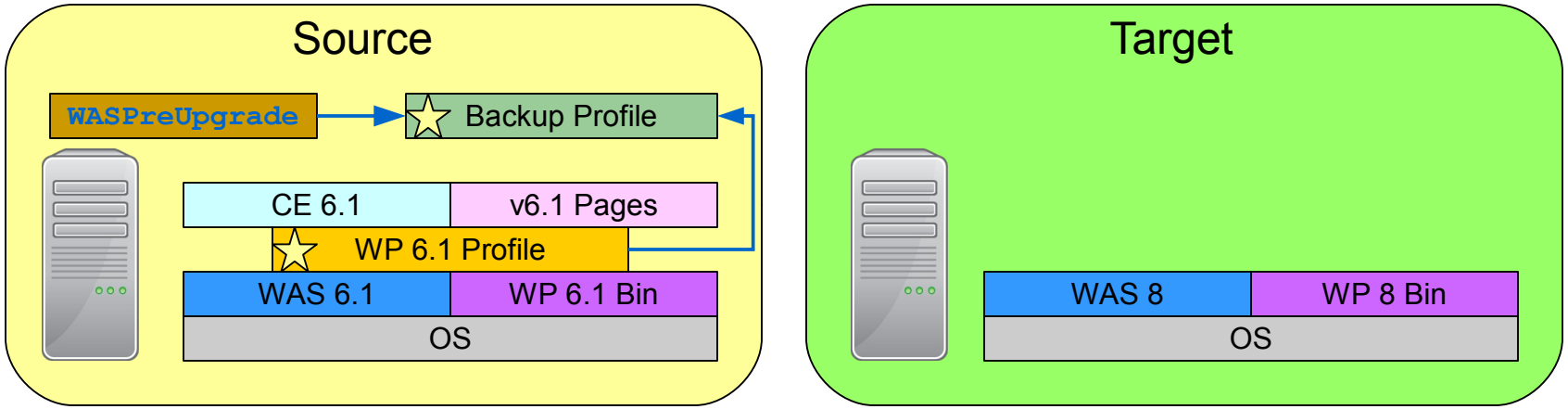
Import Source Portal Profile to Target Environment

- S: Export profile using **`WASPreUpgrade`**
  - **Clean profile prior to exporting**
  - Specify 2GB heap size with **`-javaoption -Xmx2048m`**
- T: Copy exported backup profile
- T: Create a default profile using **`manageprofiles`**
- T: Import profile using **`WASPostUpgrade`**
- T: Upgrade ConfigEngine using **`upgradeConfigEngine`**



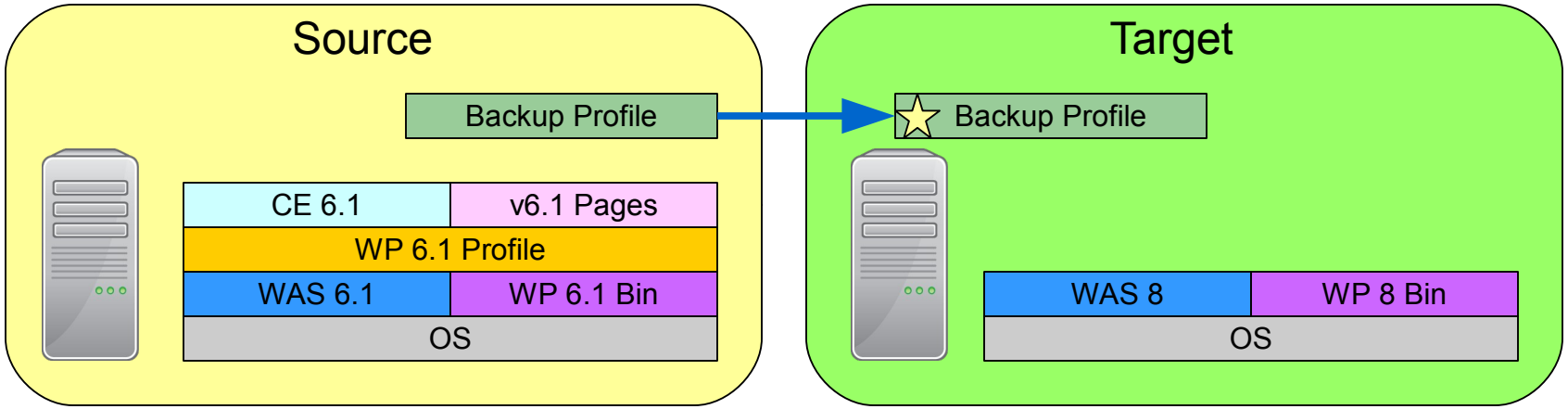
Import Source Portal Profile to Target Environment

- S: Export profile using **`WASPreUpgrade`**
  - **Clean profile prior to exporting**
  - Specify 2GB heap size with **`-javaoption -Xmx2048m`**
- T: Copy exported backup profile
- T: Create a default profile using **`manageprofiles`**
- T: Import profile using **`WASPostUpgrade`**
- T: Upgrade ConfigEngine using **`upgradeConfigEngine`**



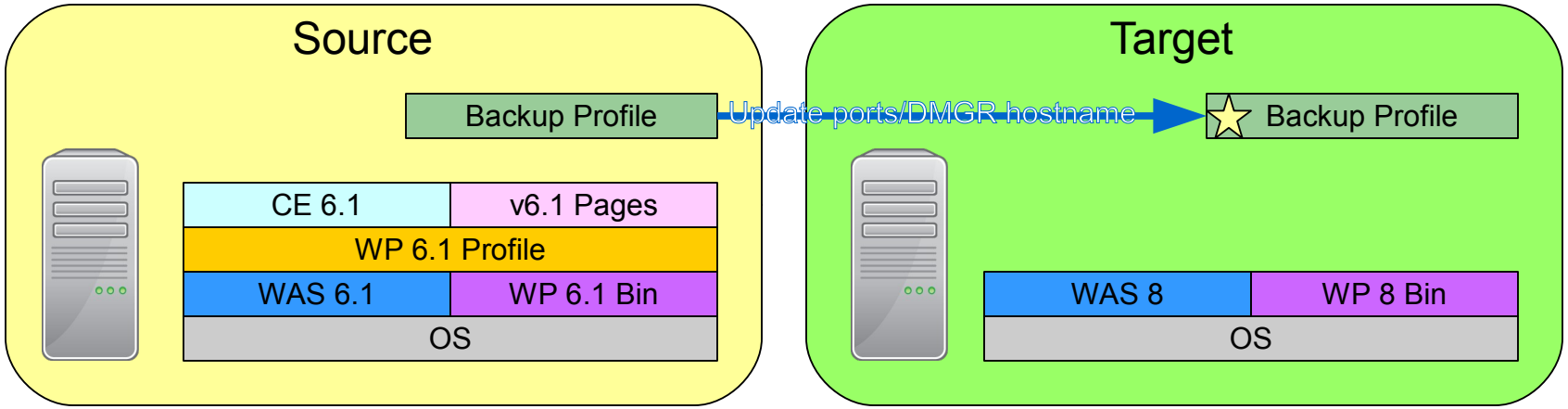
Import Source Portal Profile to Target Environment

- S: Export profile using `WASPreUpgrade`
- T: Copy exported backup profile
  - Update `wkplc*.properties` files
  - Clean backup profile
- T: Create a default profile using `manageprofiles`
- T: Import profile using `WASPostUpgrade`
- T: Upgrade ConfigEngine using `upgradeConfigEngine`



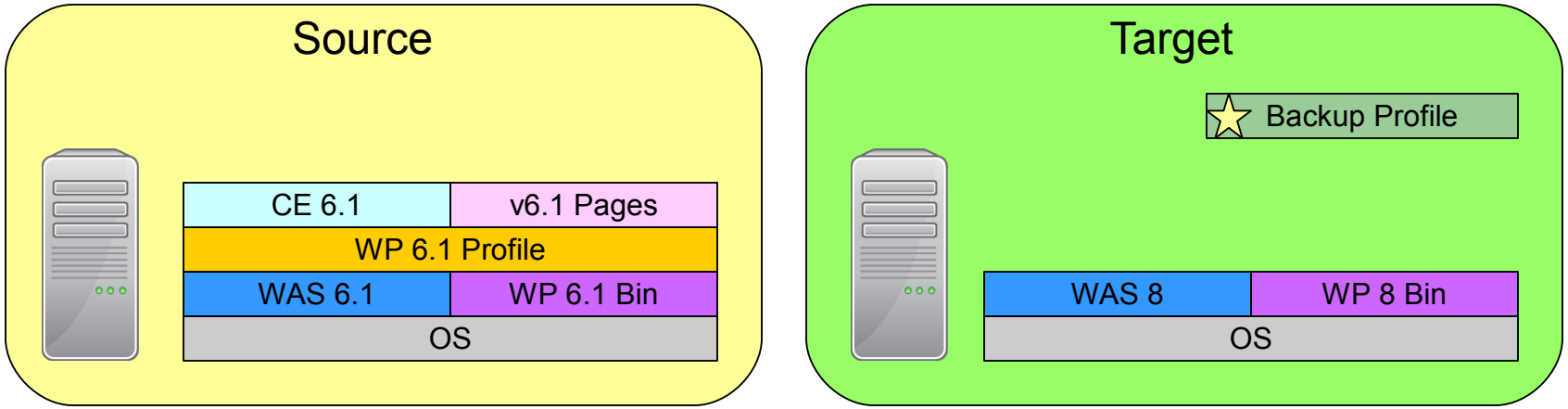
Import Source Portal Profile to Target Environment

- S: Export profile using `WASPreUpgrade`
- T: Copy exported backup profile
  - Update `wkplc*.properties` files
  - Clean backup profile
- T: Create a default profile using `manageprofiles`
- T: Import profile using `WASPostUpgrade`
- T: Upgrade ConfigEngine using `upgradeConfigEngine`



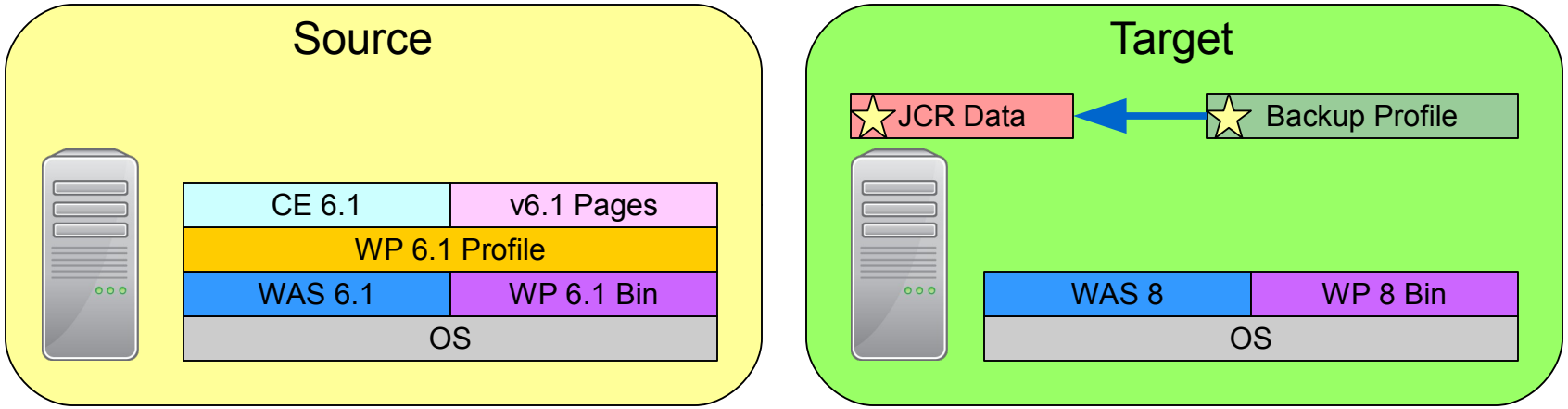
Import Source Portal Profile to Target Environment

- S: Export profile using `WASPreUpgrade`
- T: Copy exported backup profile
  - Update `wkplc*.properties` files
  - Clean backup profile
- T: Create a default profile using `manageprofiles`
- T: Import profile using `WASPostUpgrade`
- T: Upgrade ConfigEngine using `upgradeConfigEngine`



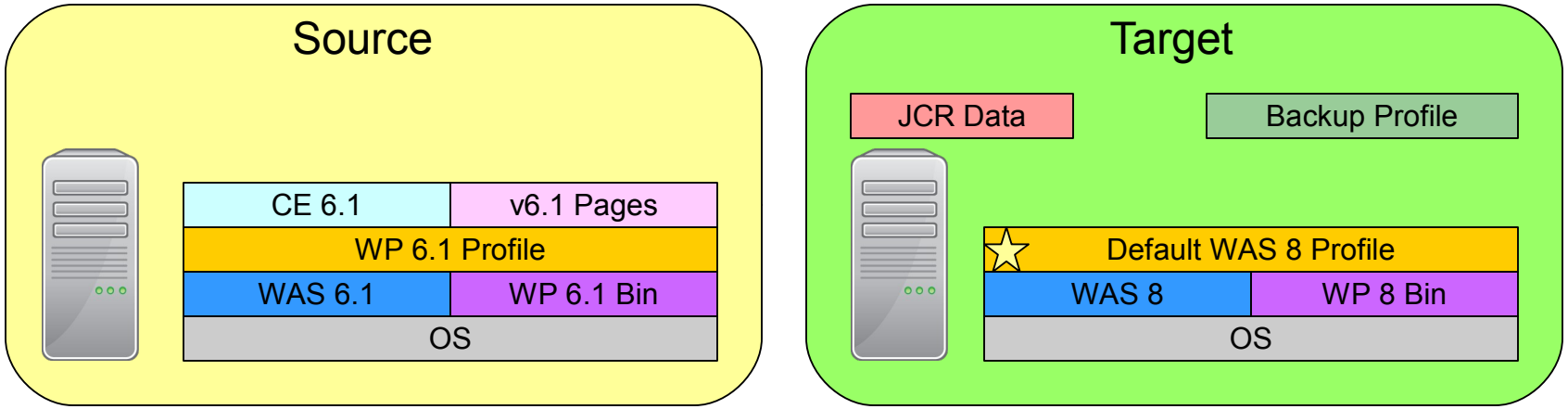
Import Source Portal Profile to Target Environment

- S: Export profile using `WASPreUpgrade`
- T: Copy exported backup profile
  - Update `wkplc*.properties` files
  - Clean backup profile
- T: Create a default profile using `manageprofiles`
- T: Import profile using `WASPostUpgrade`
- T: Upgrade ConfigEngine using `upgradeConfigEngine`



Import Source Portal Profile to Target Environment

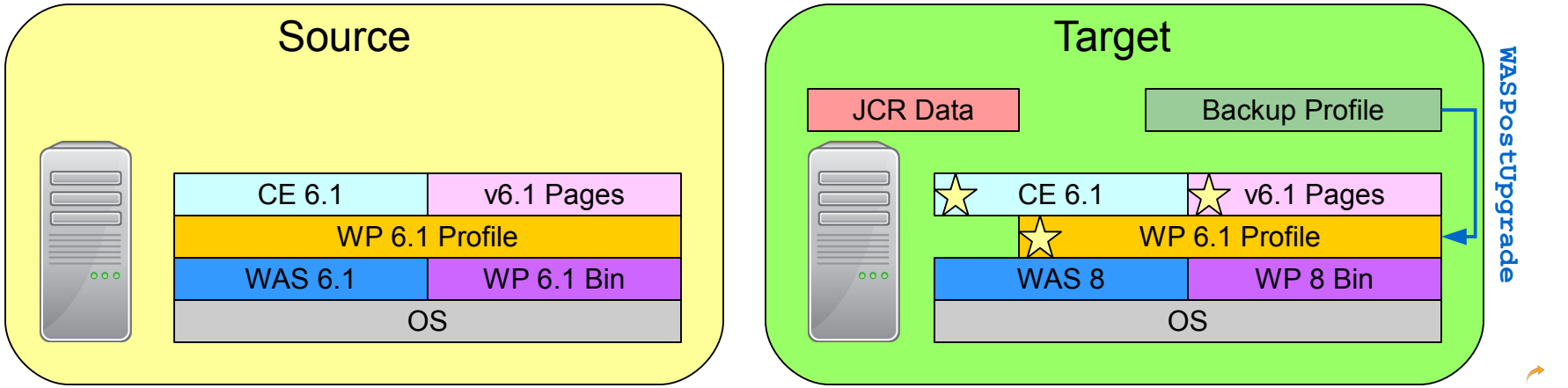
- S: Export profile using **`WASPreUpgrade`**
- T: Copy exported backup profile
- T: Create a default profile using **`manageprofiles`**
- T: Import profile using **`WASPostUpgrade`**
- T: Upgrade ConfigEngine using **`upgradeConfigEngine`**



Import Source Portal Profile to Target Environment

- S: Export profile using `WASPreUpgrade`
- T: Copy exported backup profile
- T: Create a default profile using `manageprofiles`
- T: Import profile using **`WASPostUpgrade`**
  - Specify 2GB heap size with **`-javaoption -Xmx2048m`**
  - Restore JCR data to backup profile
- T: Upgrade ConfigEngine using `upgradeConfigEngine`

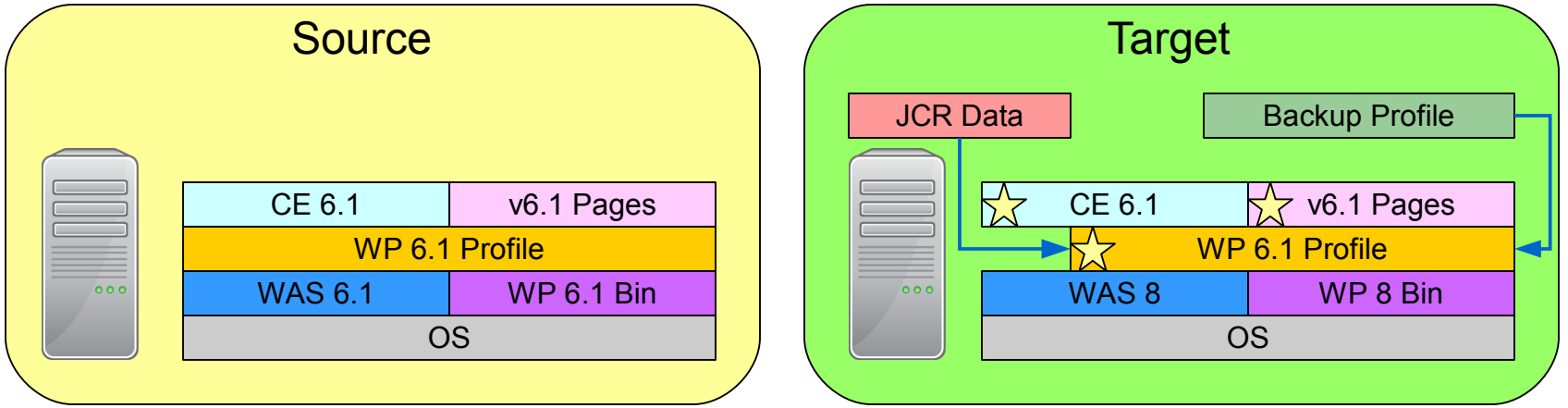
Make sure to bring over any dependencies for any custom code.



Import Source Portal Profile to Target Environment

- S: Export profile using `WASPreUpgrade`
- T: Copy exported backup profile
- T: Create a default profile using `manageprofiles`
- T: Import profile using **`WASPostUpgrade`**
  - Specify 2GB heap size with **`-javaoption -Xmx2048m`**
  - Restore JCR data to backup profile
- T: Upgrade ConfigEngine using `upgradeConfigEngine`

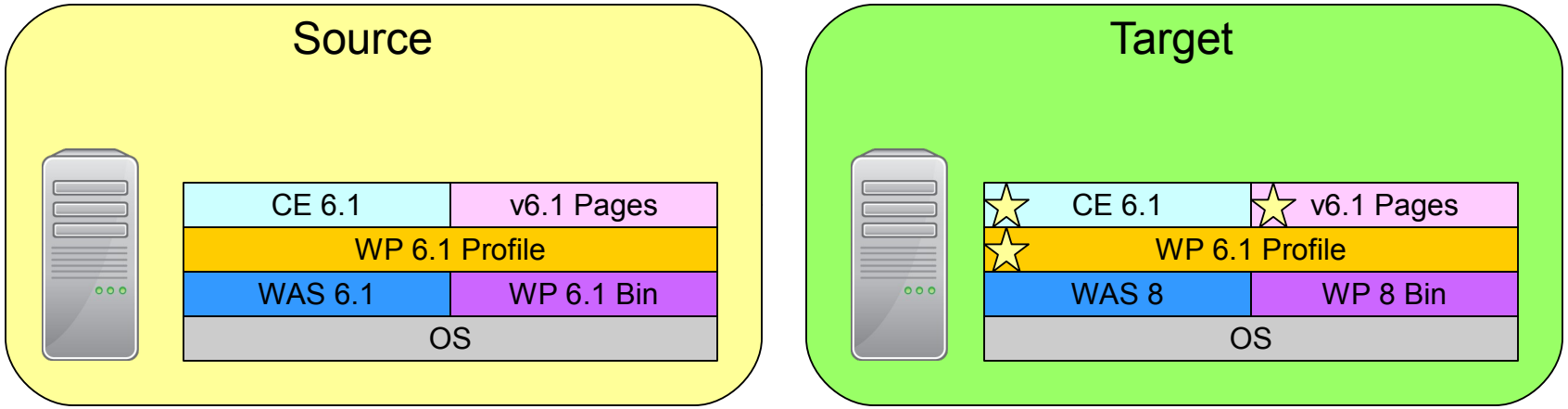
Make sure to bring over any dependencies for any custom code.



Import Source Portal Profile to Target Environment

- S: Export profile using **`WASPreUpgrade`**
- T: Copy exported backup profile
- T: Create a default profile using **`manageprofiles`**
- T: Import profile using **`WASPostUpgrade`**
  - Specify 2GB heap size with **`-javaoption -Xmx2048m`**
  - Restore JCR data to backup profile
- T: Upgrade ConfigEngine using **`upgradeConfigEngine`**

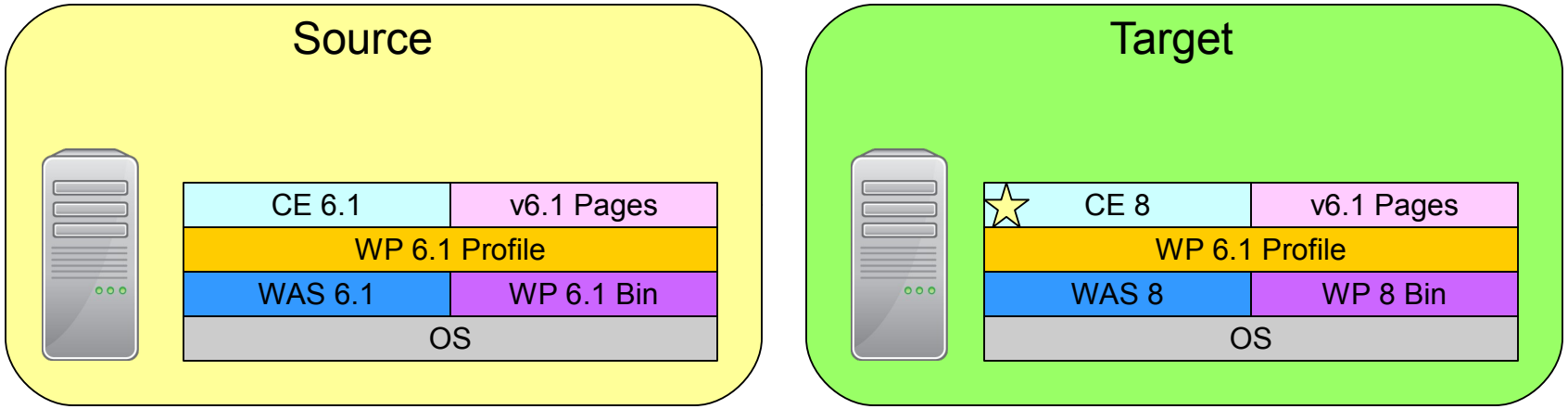
Make sure to bring over any dependencies for any custom code.



Import Source Portal Profile to Target Environment

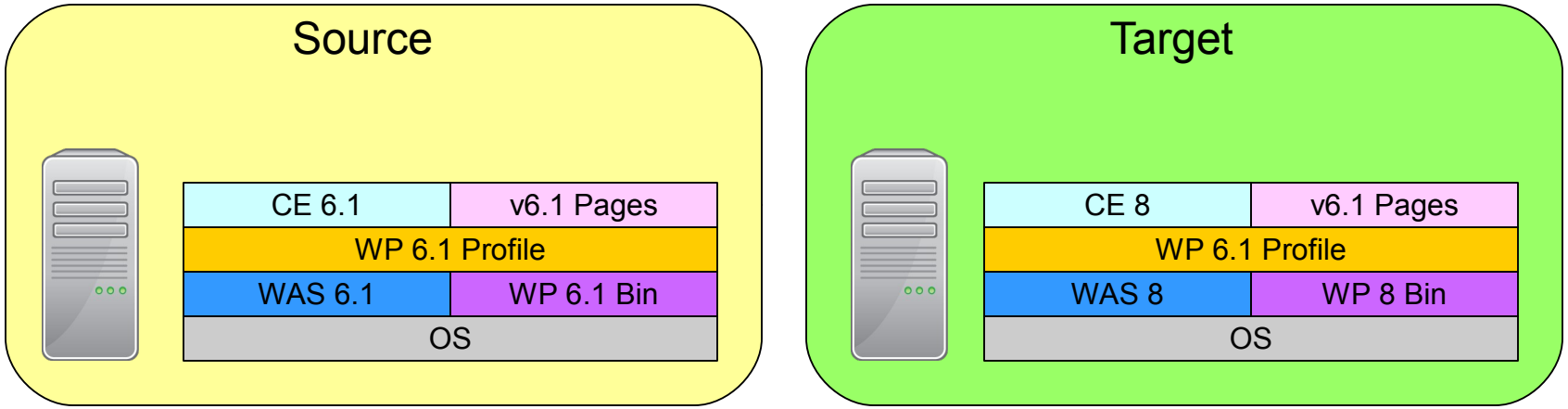
- S: Export profile using `WASPreUpgrade`
- T: Copy exported backup profile
- T: Create a default profile using `manageprofiles`
- T: Import profile using `WASPostUpgrade`
- T: Upgrade ConfigEngine using `upgradeConfigEngine`

At this point you can start the Portal Server for a sanity check. It should start and serve web pages, but expect to see many errors in the log files.



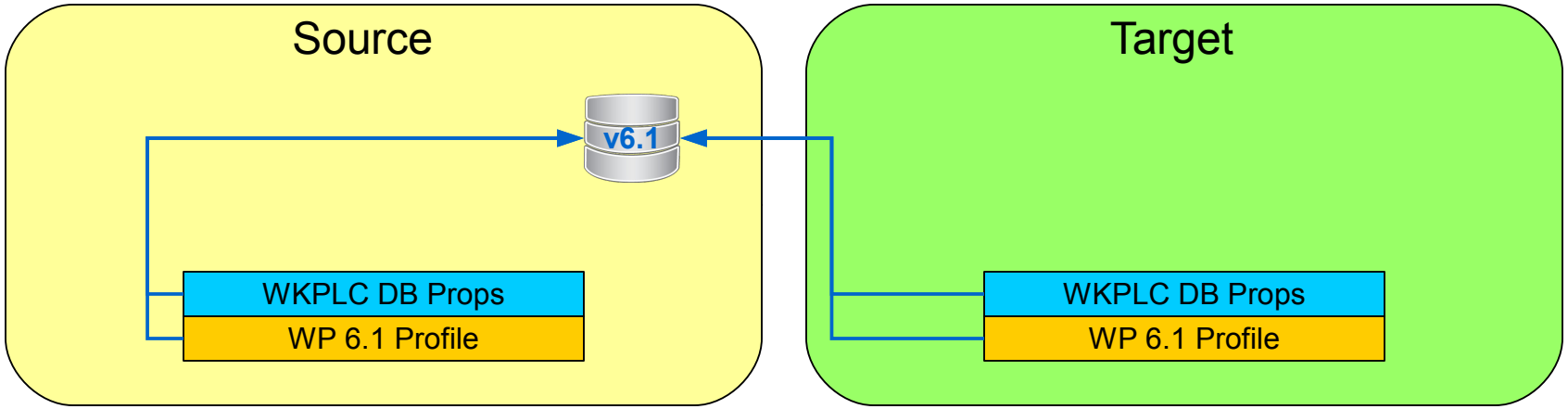
Best Practices

- Clean up **source** profile and **backup** profile to prevent OOM
  - `wp_profile` should be less than 2GB
  - Temporarily remove JCR Data if needed
- Use appropriate heap size for `WASPreUpgrade` and `WASPostUgprade`
  - Usually 2GB is sufficient, but larger can be used if enough physical memory
- If a local standalone migration, update `serverindex.xml` ports in backup profile to prevent port collisions
- Copy any custom jars or files that will be needed by your target environment



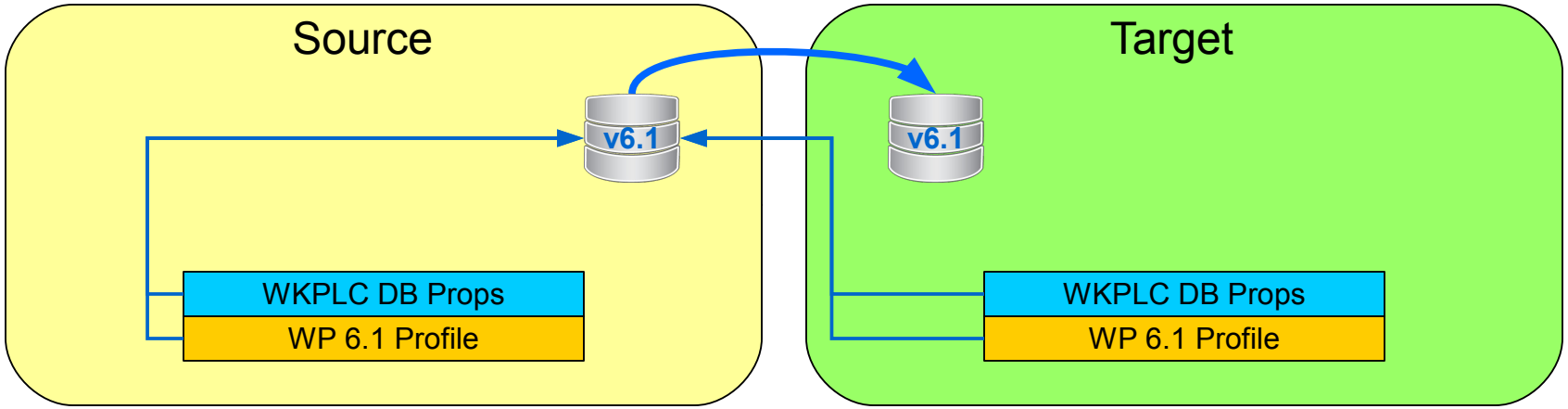
Connect Target Environment to Database Copies

- S: Make copies of the databases
- T: Update the `db_type` and `db_domain` properties files
- T: Run `ConfigEngine validate-database` task
- T: Run `ConfigEngine connect-database` task
- T: **For z/OS only,** update the database with `ConfigEngine upgrade-database`



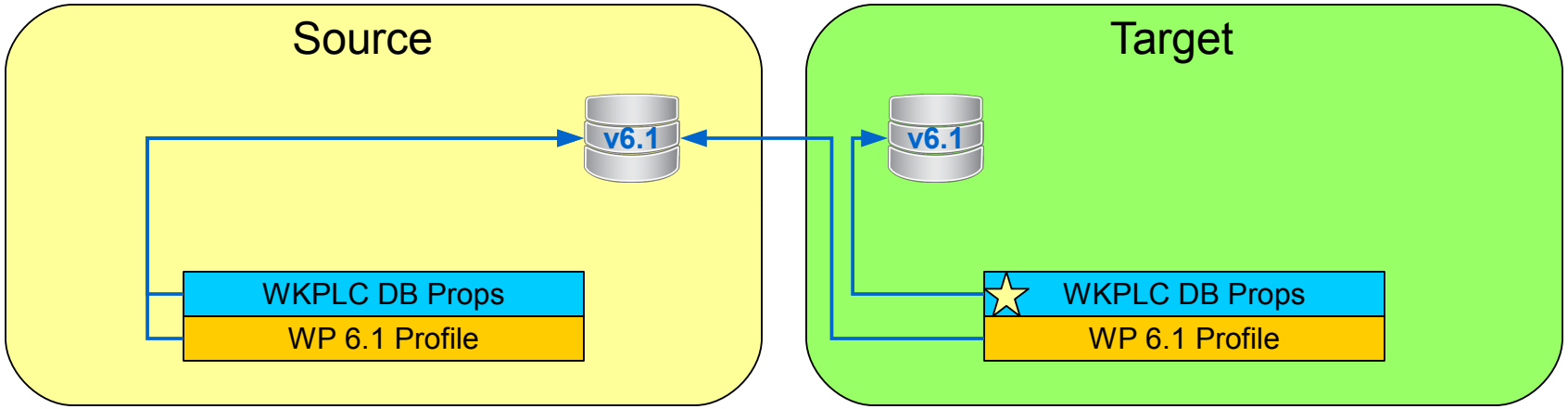
Connect Target Environment to Database Copies

- S: Make copies of the databases
- T: Update the `db_type` and `db_domain` properties files
- T: Run `ConfigEngine validate-database` task
- T: Run `ConfigEngine connect-database` task
- T: **For z/OS only,** update the database with `ConfigEngine upgrade-database`



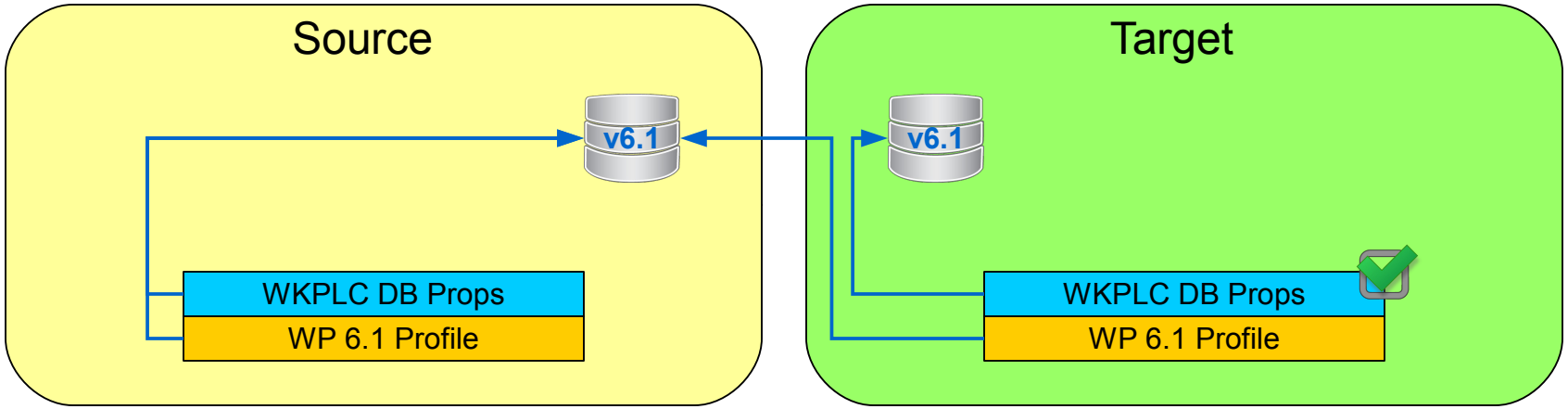
Connect Target Environment to Database Copies

- S: Make copies of the databases
- T: Update the **`db_type`** and **`db_domain`** properties files
  - JDBC 4 drivers are required
- T: Run **`ConfigEngine validate-database`** task
- T: Run **`ConfigEngine connect-database`** task
- T: **For z/OS only**, update the database with **`ConfigEngine upgrade-database`**



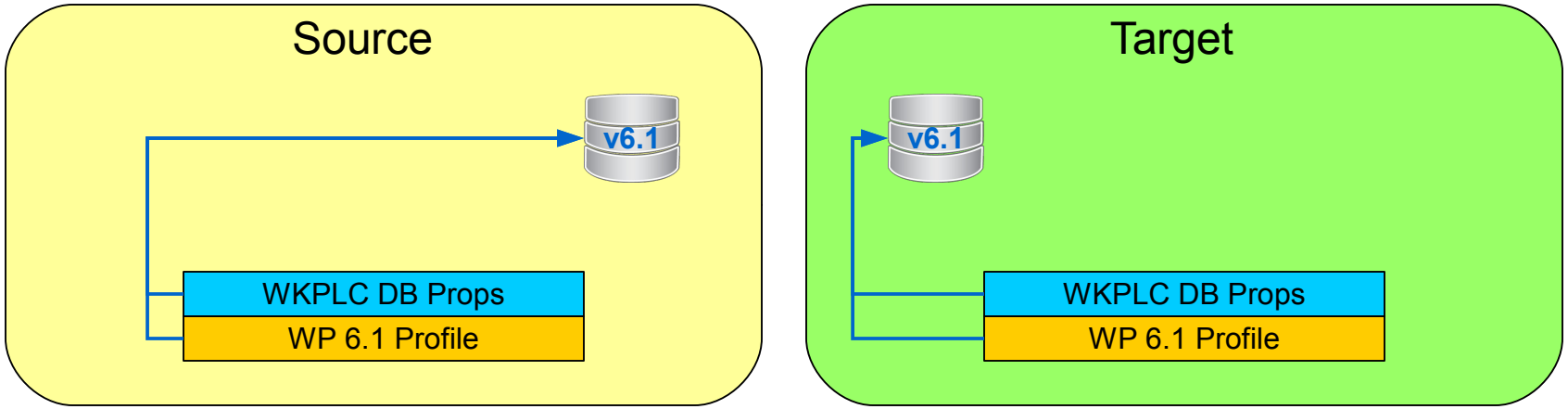
Connect Target Environment to Database Copies

- S: Make copies of the databases
- T: Update the `db_type` and `db_domain` properties files
- T: Run **`ConfigEngine validate-database`** task
- T: Run `ConfigEngine connect-database` task
- T: **For z/OS only,** update the database with `ConfigEngine upgrade-database`



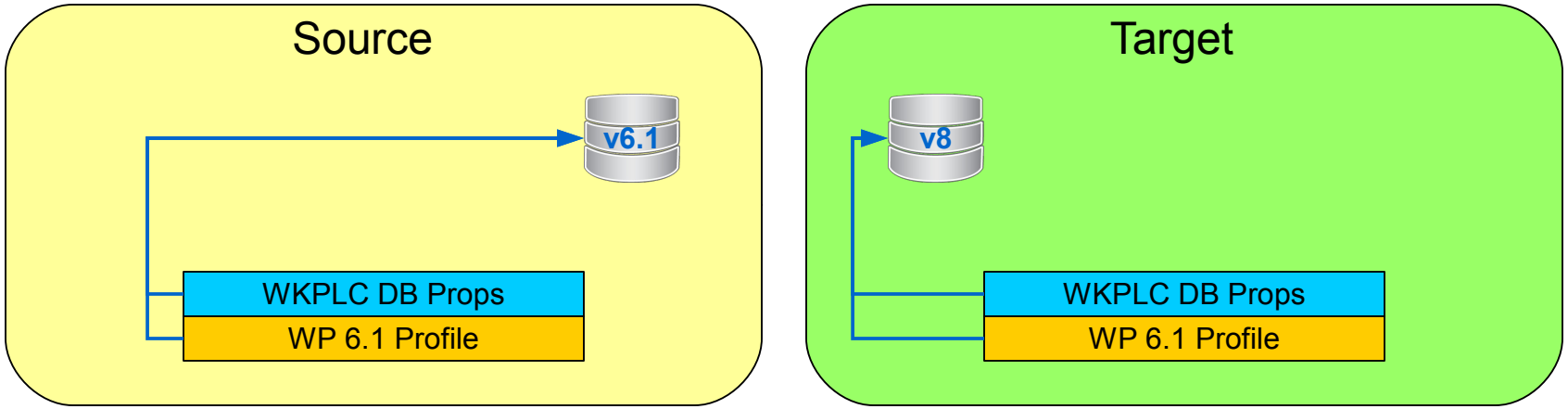
Connect Target Environment to Database Copies

- S: Make copies of the databases
- T: Update the `db_type` and `db_domain` properties files
- T: Run `ConfigEngine validate-database` task
- T: Run **`ConfigEngine connect-database`** task
- T: **For z/OS only,** update the database with `ConfigEngine upgrade-database`



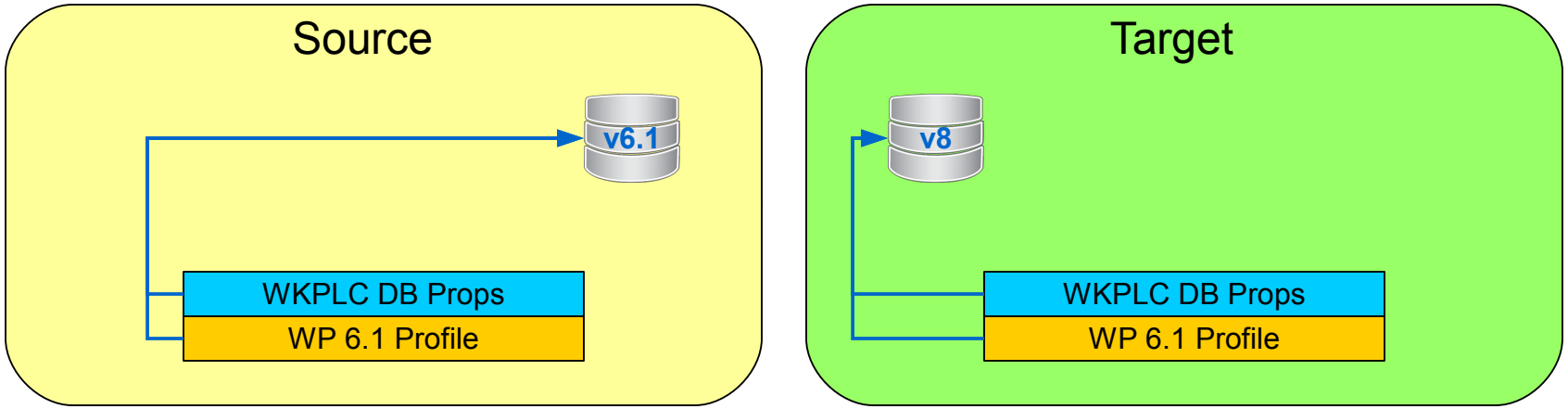
Connect Target Environment to Database Copies

- S: Make copies of the databases
- T: Update the `db_type` and `db_domain` properties files
- T: Run `ConfigEngine validate-database` task
- T: Run `ConfigEngine connect-database` task
- T: **For z/OS only**, update the database with `ConfigEngine upgrade-database`



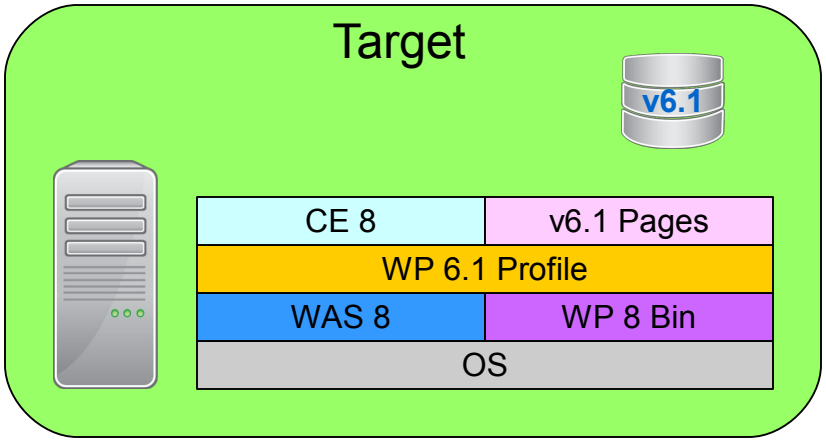
Best Practices

- Use copies of the Databases
- Make sure to update database properties in target environment and specify JDBC 4 drivers
- There are many database updates and the database conversion will generally take the majority of the time
- Invalid or corrupt data may exist in the DB from previous versions or test code so make sure to use latest fix pack



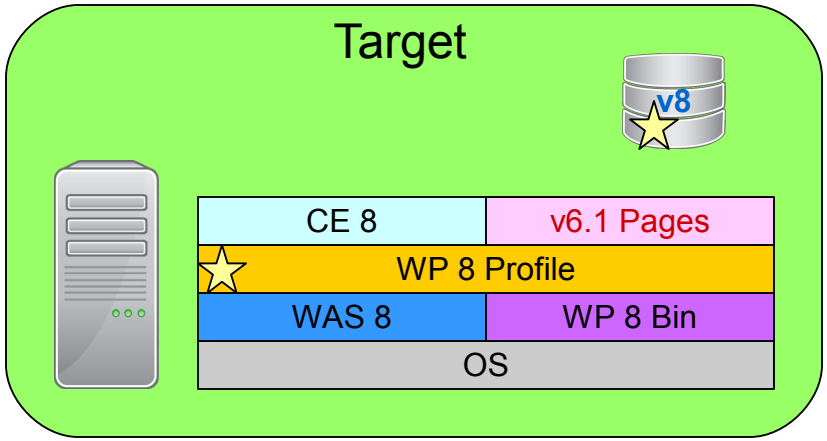
Upgrade the target Portal Profile

- Run `ConfigEngine upgrade-profile`



Upgrade the target Portal Profile

- Run **`ConfigEngine upgrade-profile`**
  - Upgrades profile to v8
  - Upgrades Databases
  - Pages are not modified

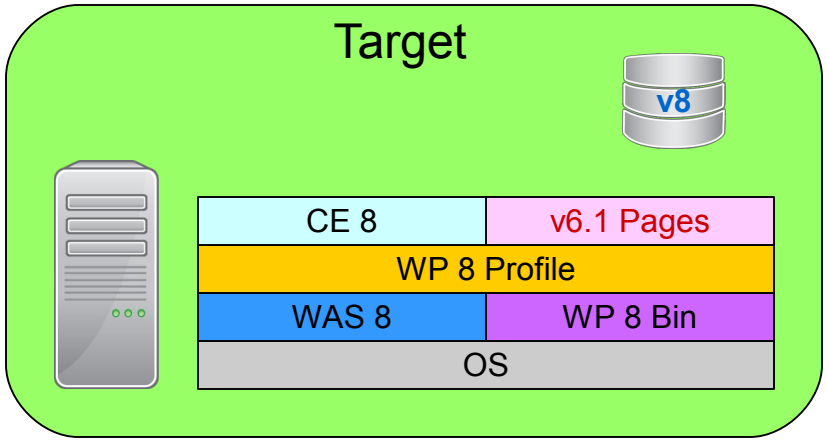


Best Practices

- Restart migration from where it left off with `-Dwp.migration.framework.resume=<resume point>`

  Example: `-Dwp.migration.framework.resume=deploy-apps`

- For troubleshooting, use `-DskipCleanup="true"` to preserve work folder
- If using both parameters, make sure to clean the contents of `wp_profile/ConfigEngine/config/work`
- **Content-Only Migration is not supported in Portal 8**

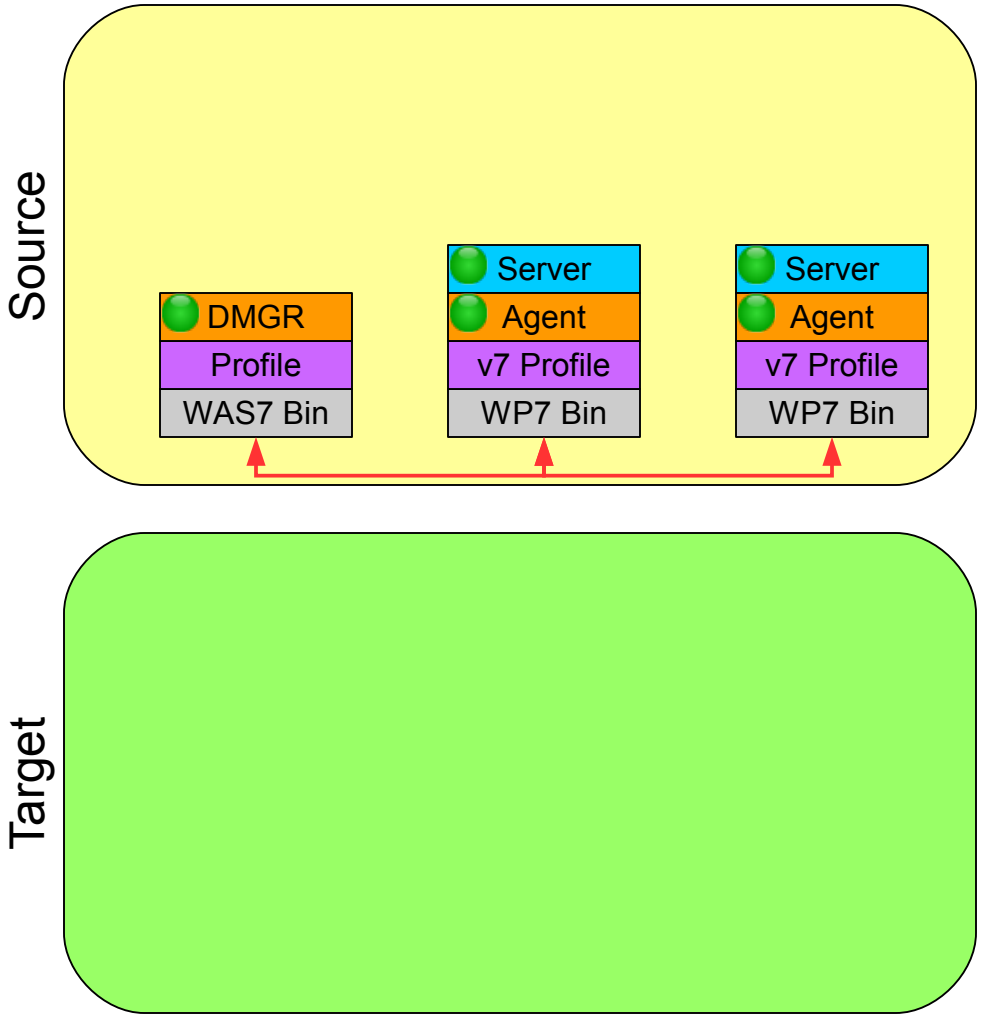


Post Migration Tasks

- Always complete post migration tasks before enabling new features
- Update hard-coded paths of Custom Themes
  - Custom themes in v6.1 wps.ear are placed in Migrated_Themes.ear retain original context root (default: **`wps`**)
  - New wps.ear is bound to <org>_migrated (default: **`wps_migrated`**)
  - Context-root for Migrated_Themes.ear must be updated before wps.ear
- Update Default Portal 8 Themes
  - **`ConfigEngine action-upgrade-theme-admin-oob`**
  - **`ConfigEngine action-upgrade-wcm-theme-admin-oob`**
- Regression and Performance Testing
- Develop and deploy updated themes
- Re-Enable Automatic Synchronization
- Update data using **`wcm-post-migration-data-update`**
  - **Must be done before enabling managed pages and migrating virtual portals**
- Refer to the wiki for many other Post Migration Tasks

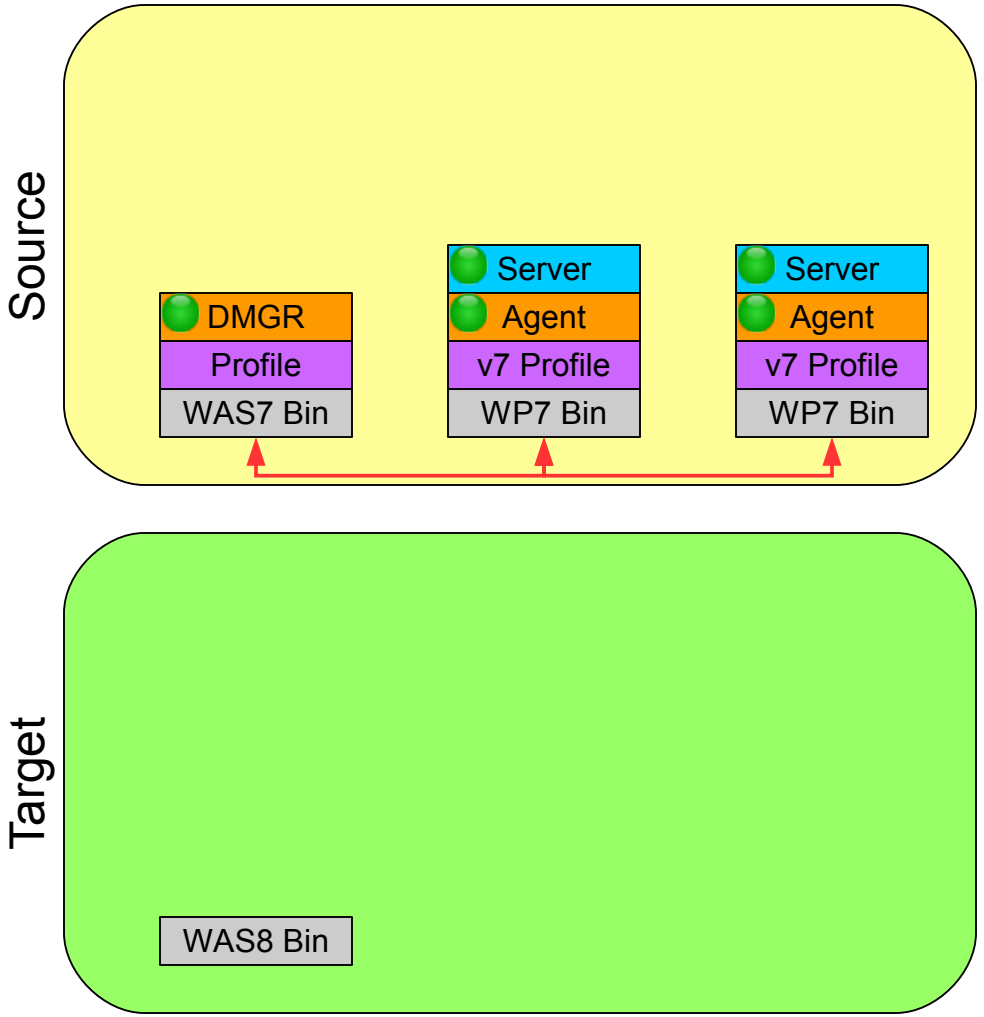
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations



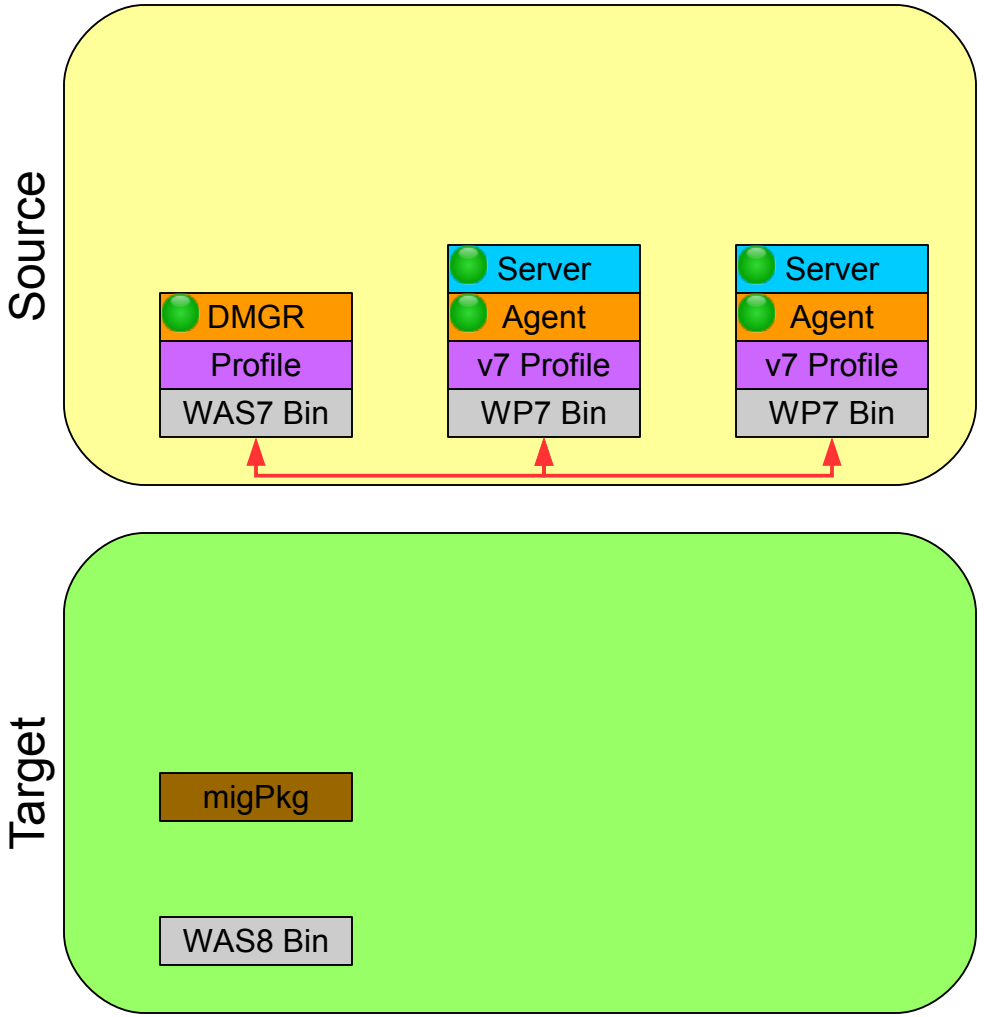
Cluster Migration

- Migrate DMGR profile
  - S: Disable AutoSync
  - **T: Install binaries**
  - T: Create fileForDmgr.zip
  - S: Copy fileForDmgr.zip
  - S: WASPreUpgrade
  - T: Copy backup profile
  - T: WASPostUpgrade
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations



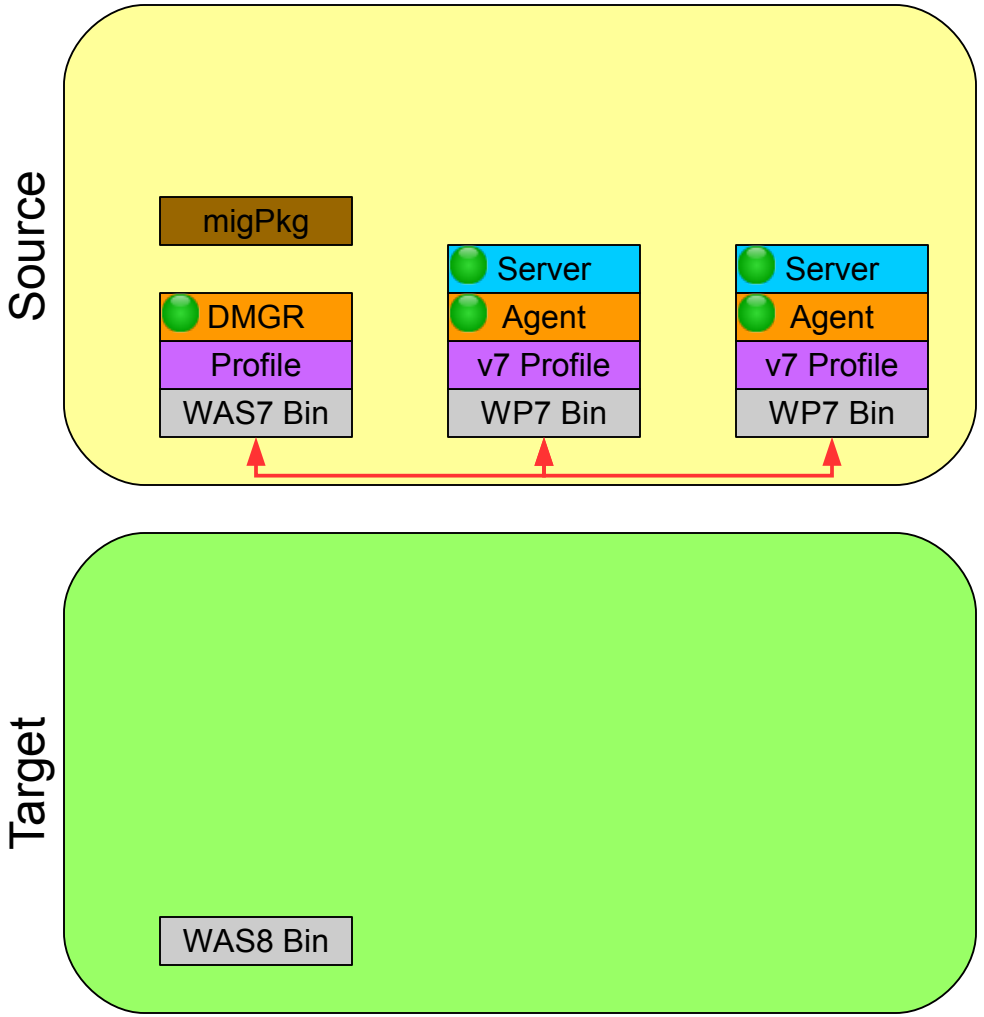
Cluster Migration

- Migrate DMGR profile
  - S: Disable AutoSync
  - T: Install binaries
  - **T: Create fileForDmgr.zip**
  - S: Copy fileForDmgr.zip
  - S: WASPreUpgrade
  - T: Copy backup profile
  - T: WASPostUpgrade
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations



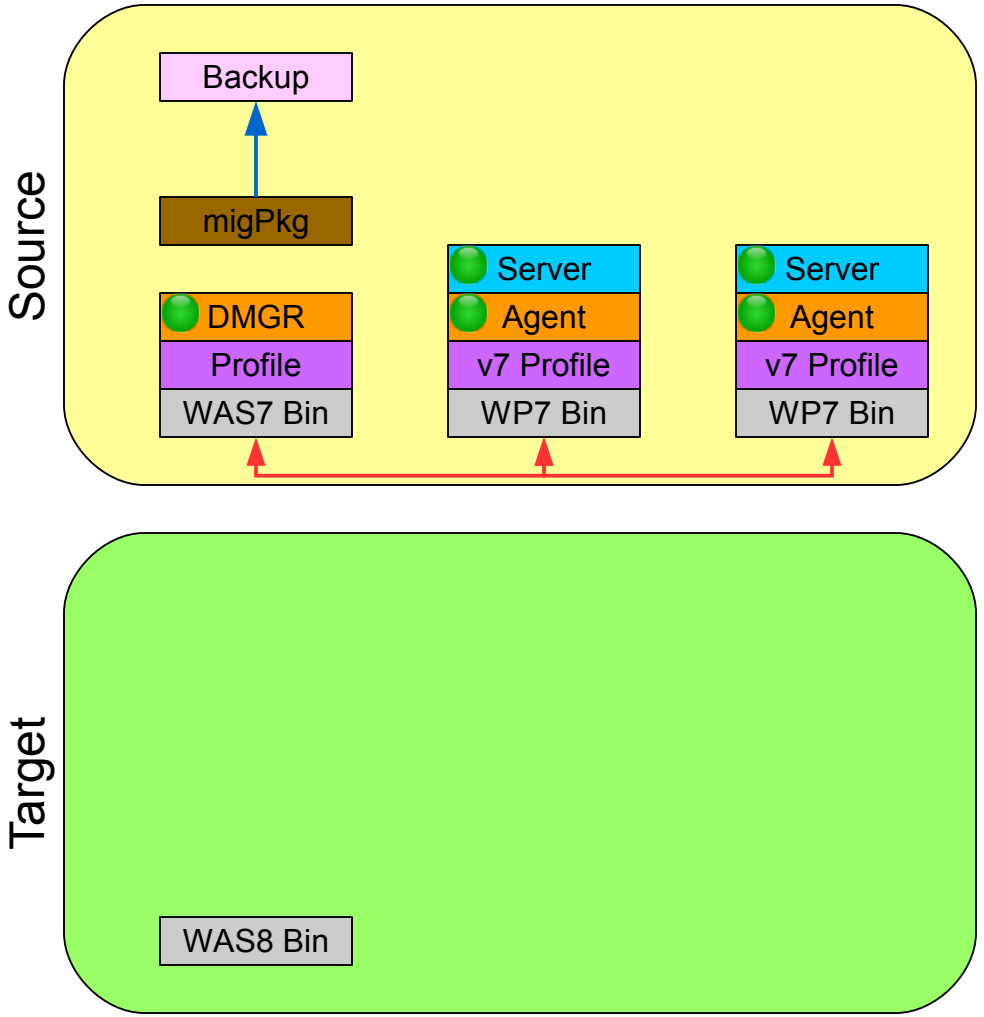
Cluster Migration

- Migrate DMGR profile
  - S: Disable AutoSync
  - T: Install binaries
  - T: Create fileForDmgr.zip
  - **S: Copy fileForDmgr.zip**
  - S: WASPreUpgrade
  - T: Copy backup profile
  - T: WASPostUpgrade
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations



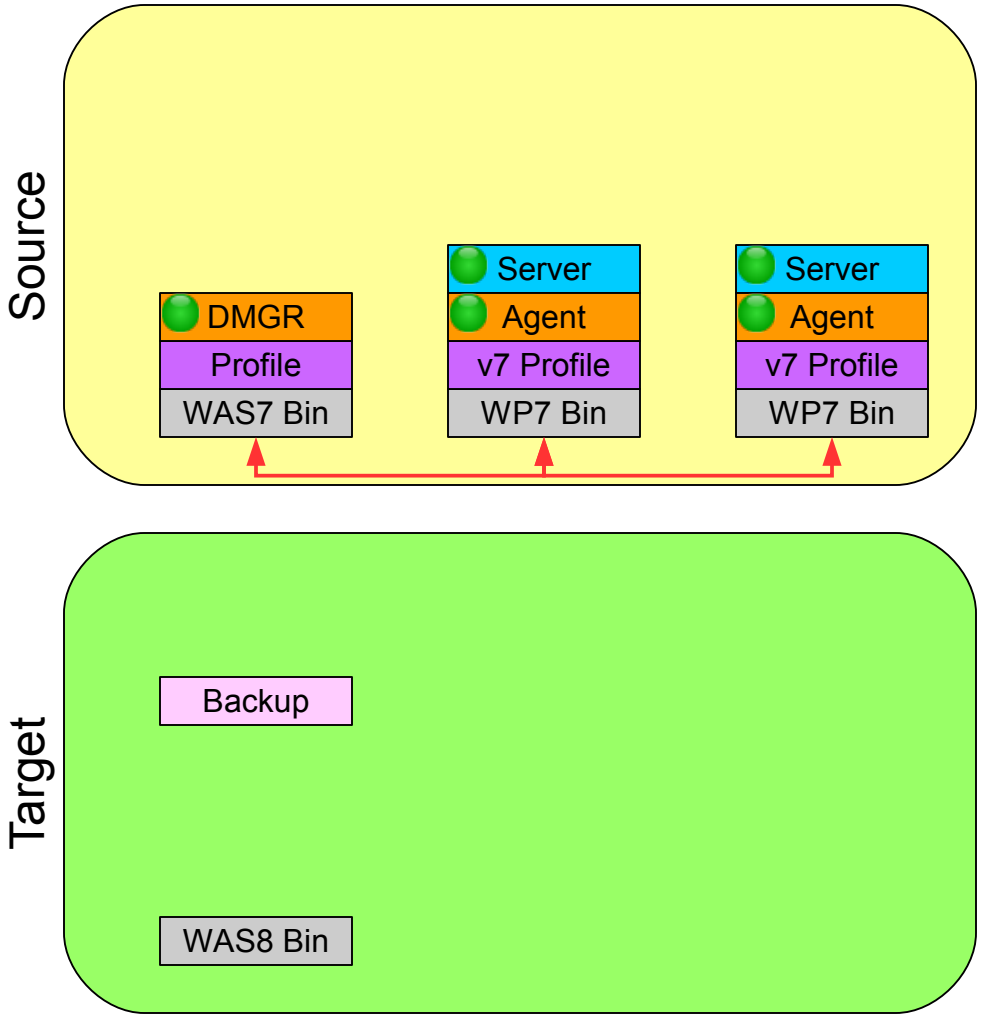
Cluster Migration

- Migrate DMGR profile
  - S: Disable AutoSync
  - T: Install binaries
  - T: Create fileForDmgr.zip
  - S: Copy fileForDmgr.zip
  - **S: WASPreUpgrade**
  - T: Copy backup profile
  - T: WASPostUpgrade
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations



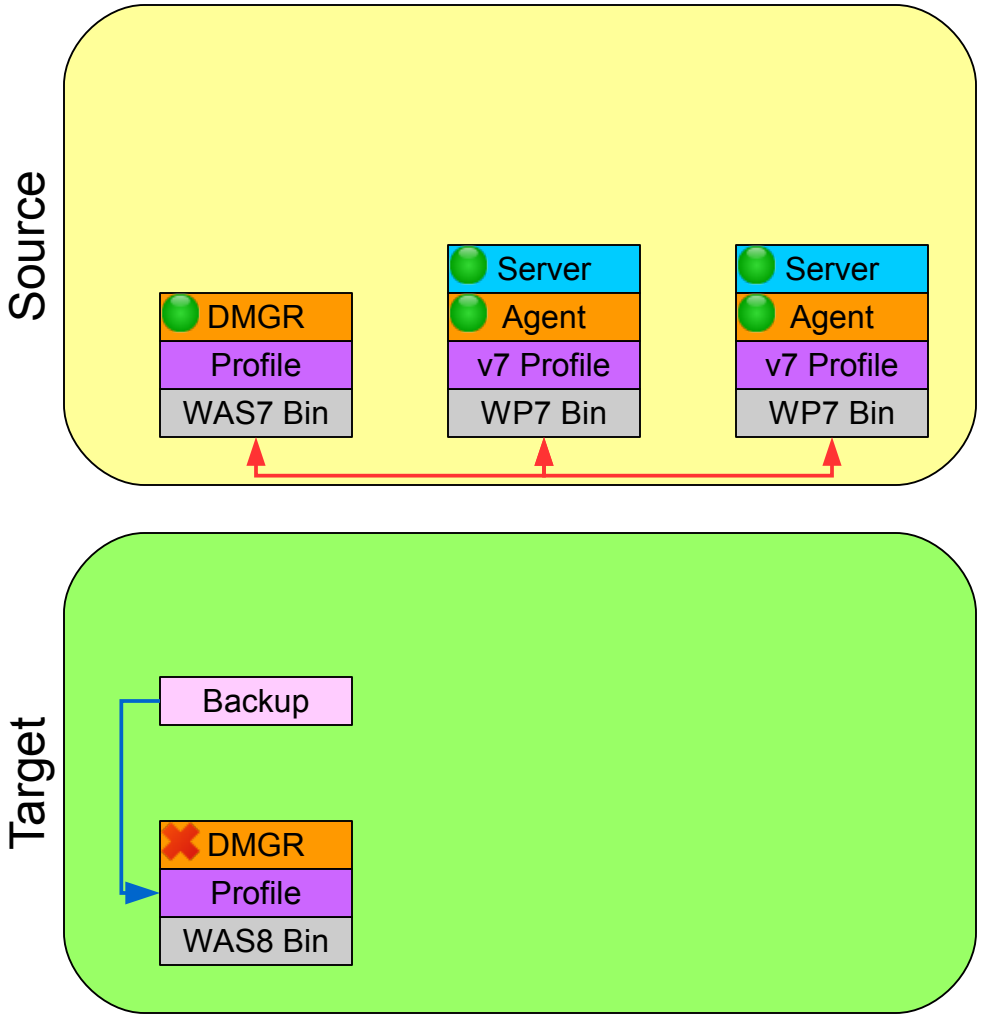
Cluster Migration

- Migrate DMGR profile
  - S: Disable AutoSync
  - T: Install binaries
  - T: Create fileForDmgr.zip
  - S: Copy fileForDmgr.zip
  - S: WASPreUpgrade
  - **T: Copy backup profile**
  - T: WASPostUpgrade
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations



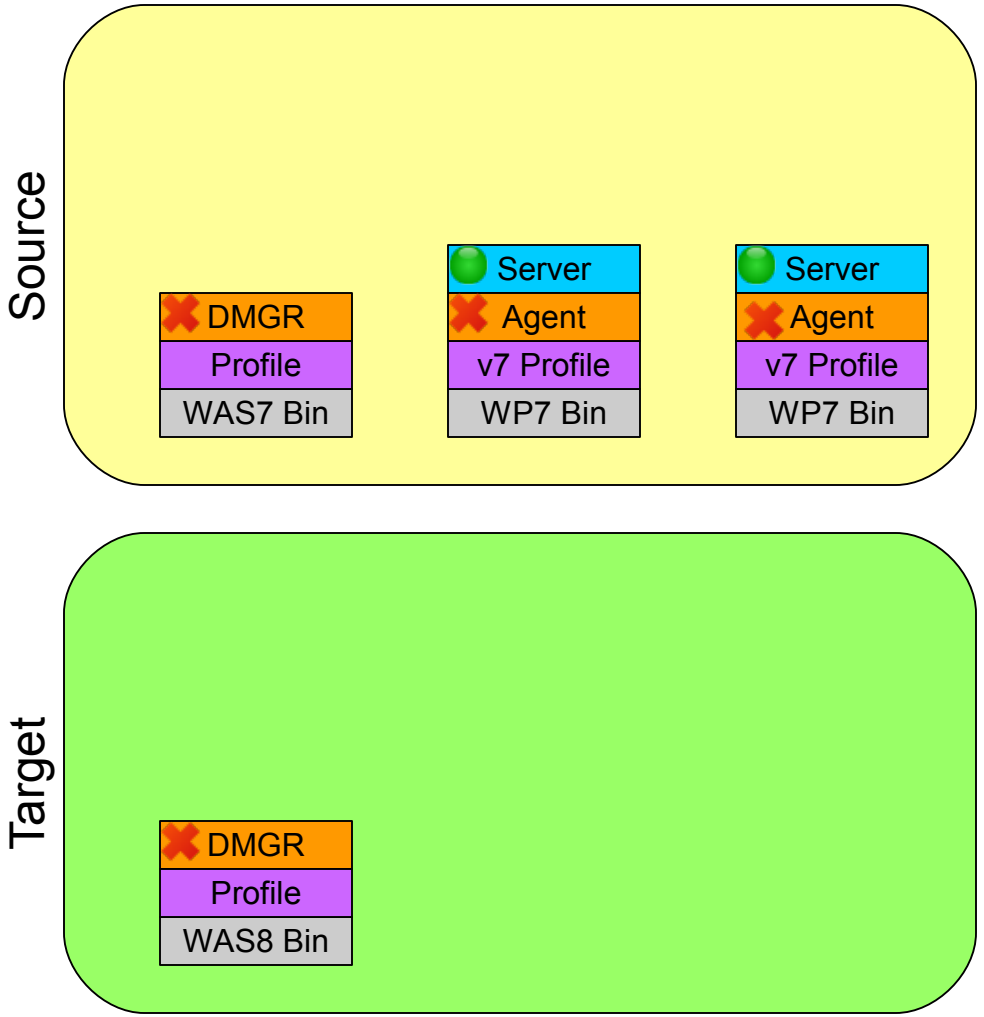
Cluster Migration

- Migrate DMGR profile
  - S: Disable AutoSync
  - T: Install binaries
  - T: Create fileForDmgr.zip
  - S: Copy fileForDmgr.zip
  - S: WASPreUpgrade
  - T: Copy backup profile
  - **T: WASPostUpgrade**
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations



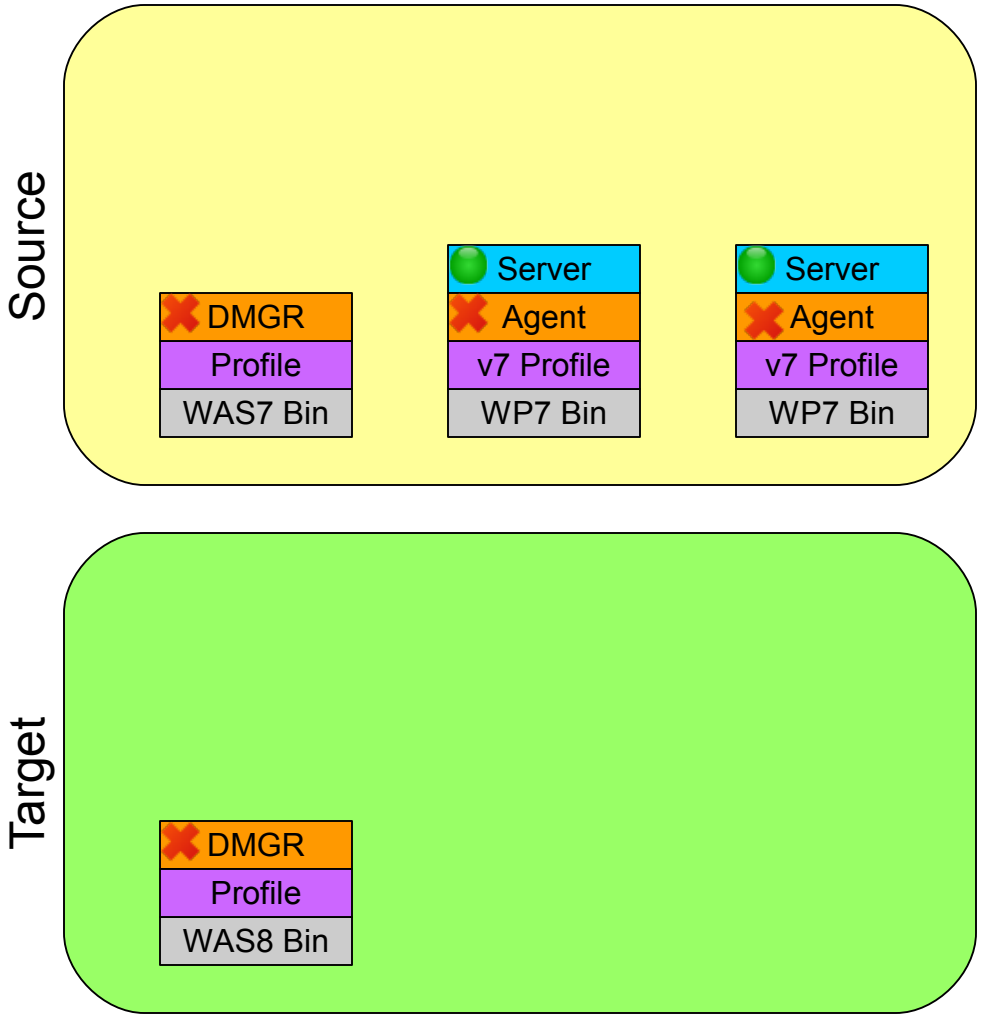
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations



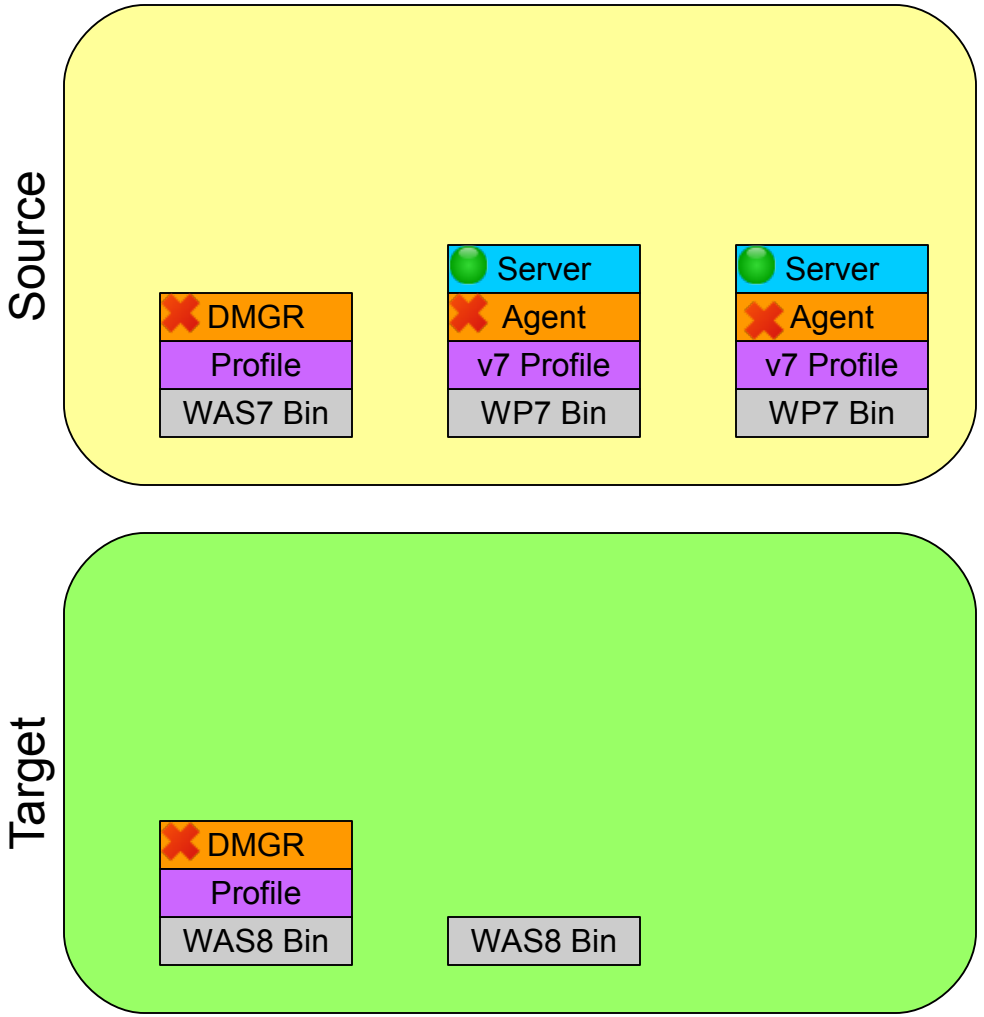
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
  - Install binaries
  - Create MigRemPkg
  - WASPreUpgrade
  - WASPostUpgrade
- Set up unique ports
- Complete node migrations



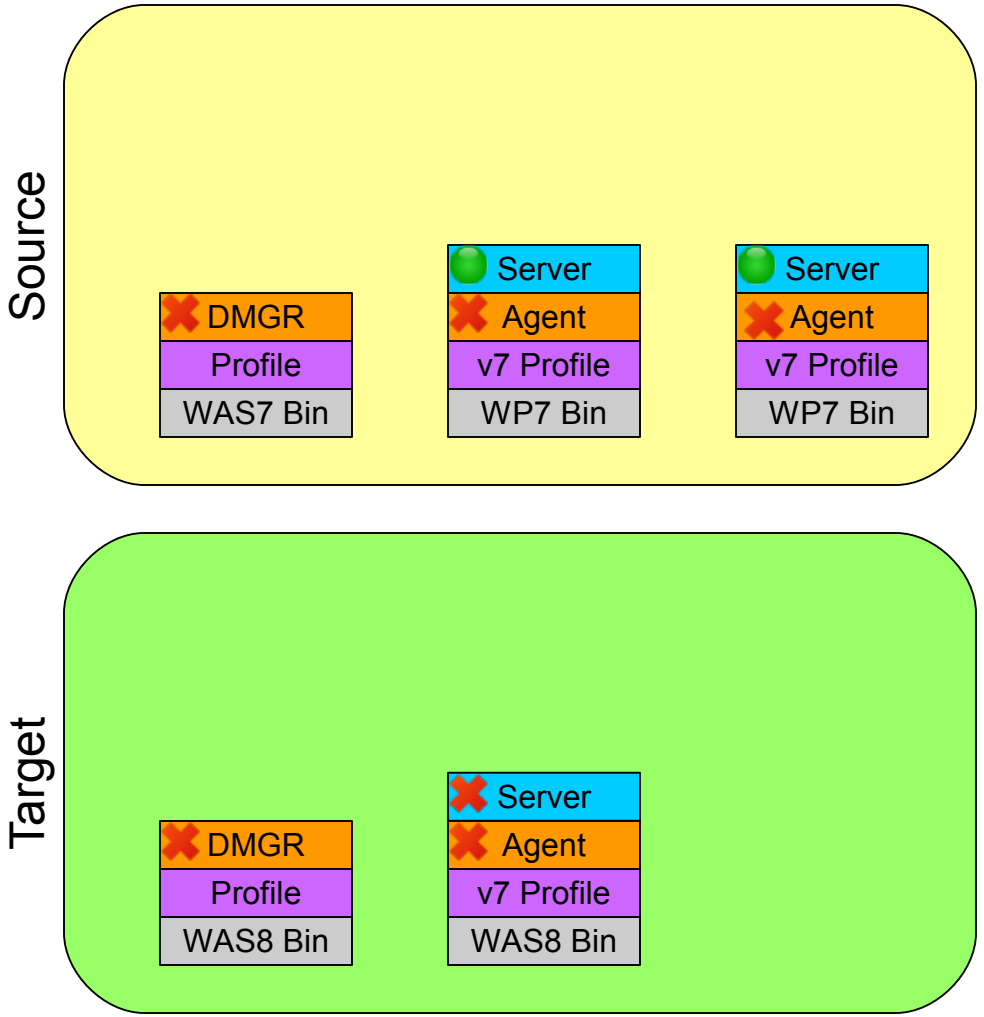
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
  - Install binaries
  - Create MigRemPkg
  - WASPreUpgrade
  - WASPostUpgrade
- Set up unique ports
- Complete node migrations



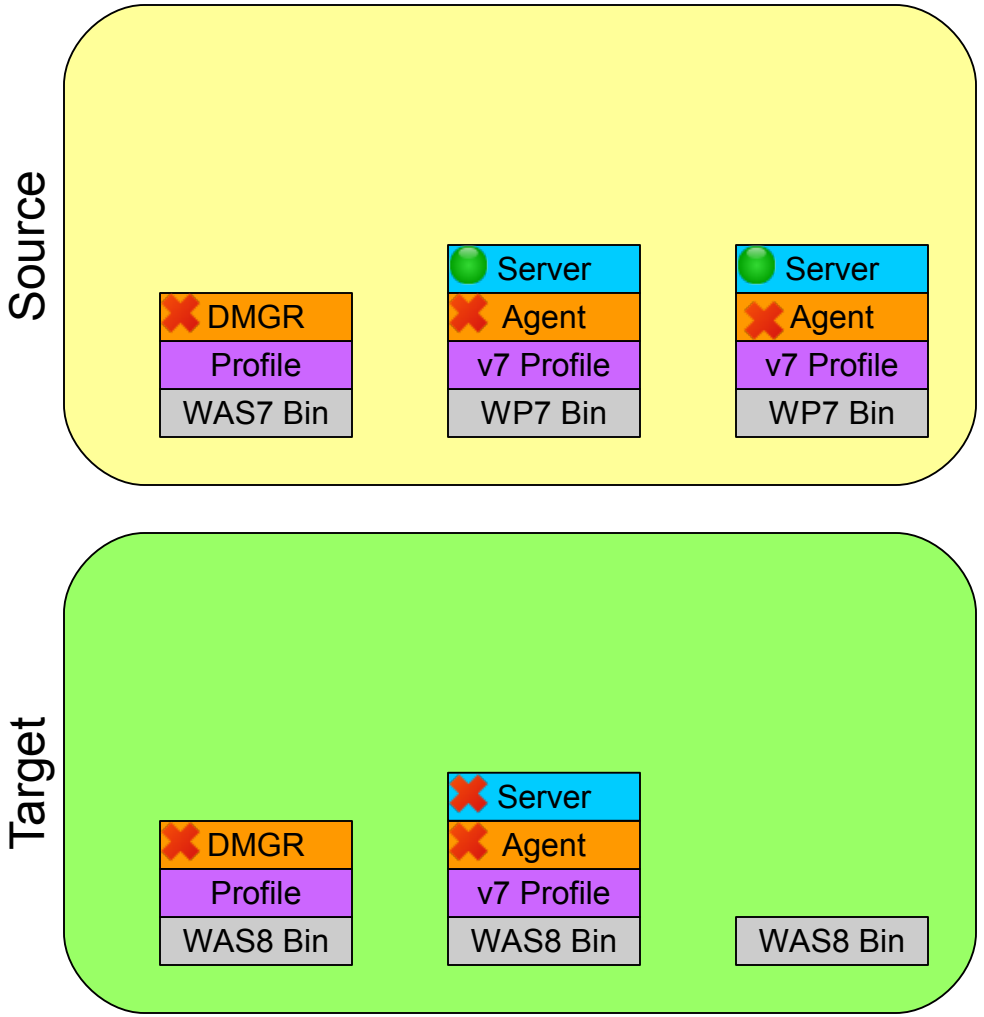
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
  - Install binaries
  - Create MigRemPkg
  - WASPreUpgrade
  - WASPostUpgrade
- Set up unique ports
- Complete node migrations



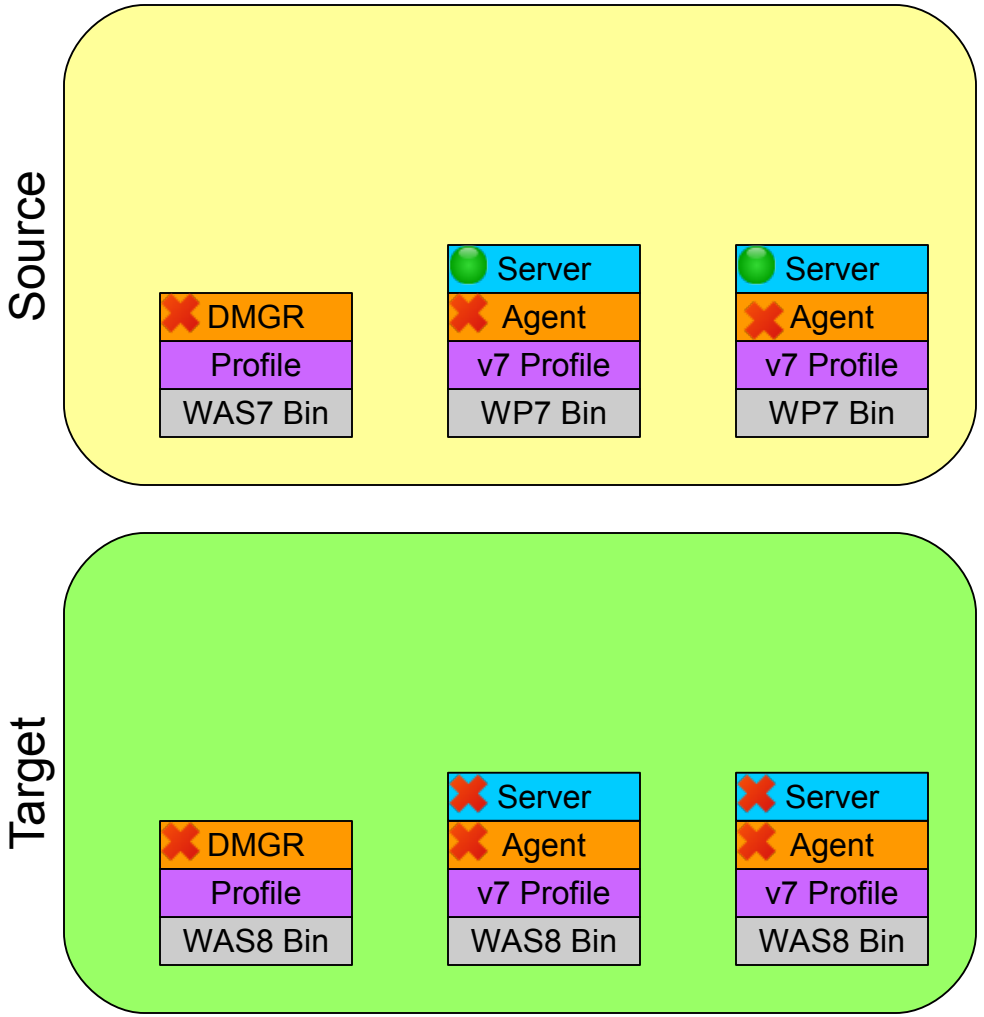
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
  - Install binaries
  - Create MigRemPkg
  - WASPreUpgrade
  - WASPostUpgrade
- Set up unique ports
- Complete node migrations



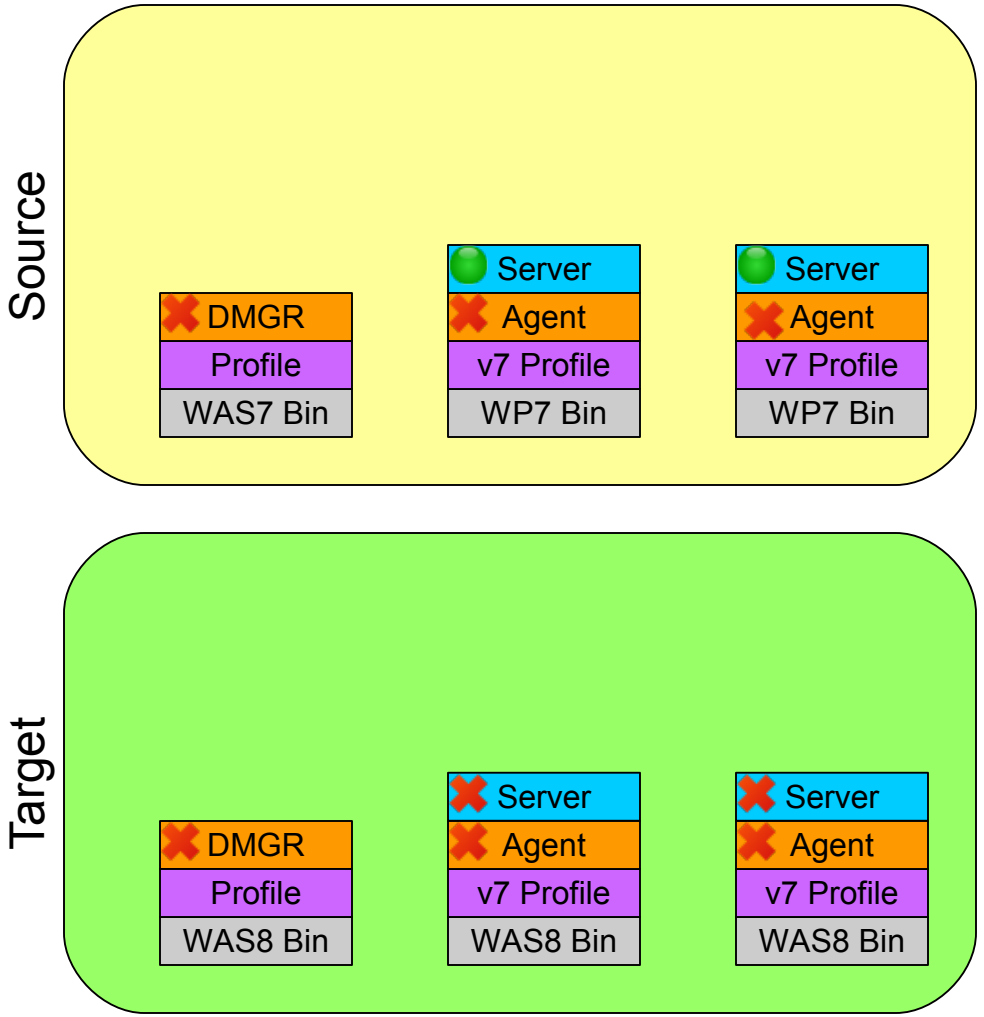
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
  - Install binaries
  - Create MigRemPkg
  - WASPreUpgrade
  - WASPostUpgrade
- Set up unique ports
- Complete node migrations



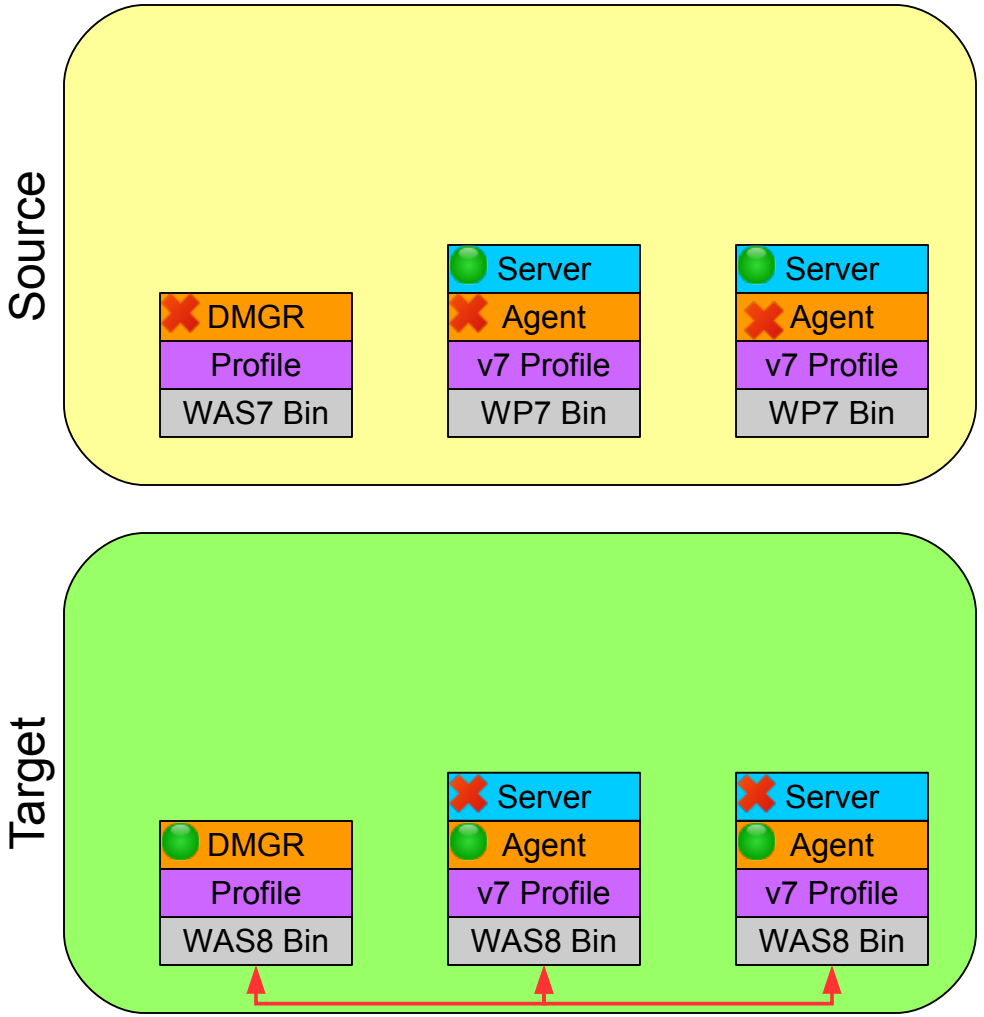
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
  - Start DMGR/Node Agents
  - Update ports in ISC
  - Fully re-synchronize
  - Restart DMGR/Node Agents
- Complete node migrations



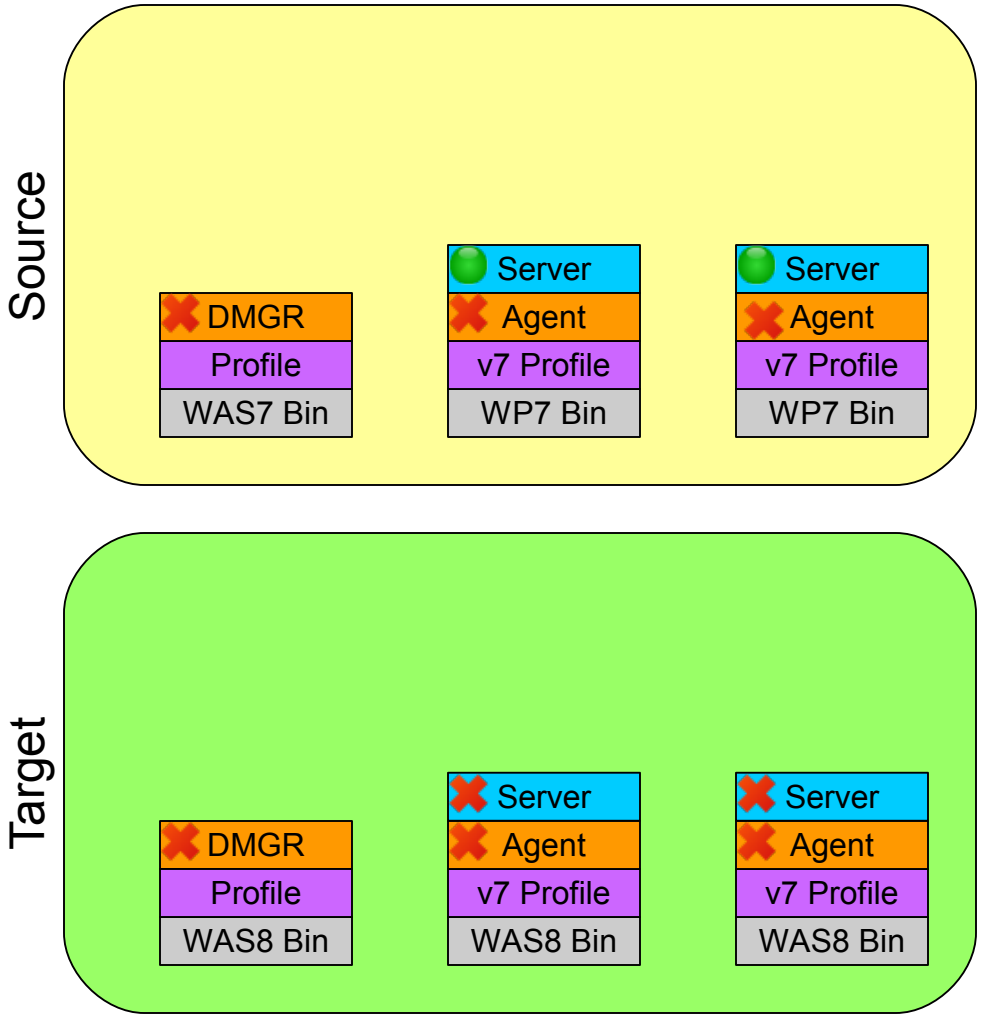
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
  - Start DMGR/Node Agents
  - Update ports in ISC
  - Fully re-synchronize
  - Restart DMGR/Node Agents
- Complete node migrations



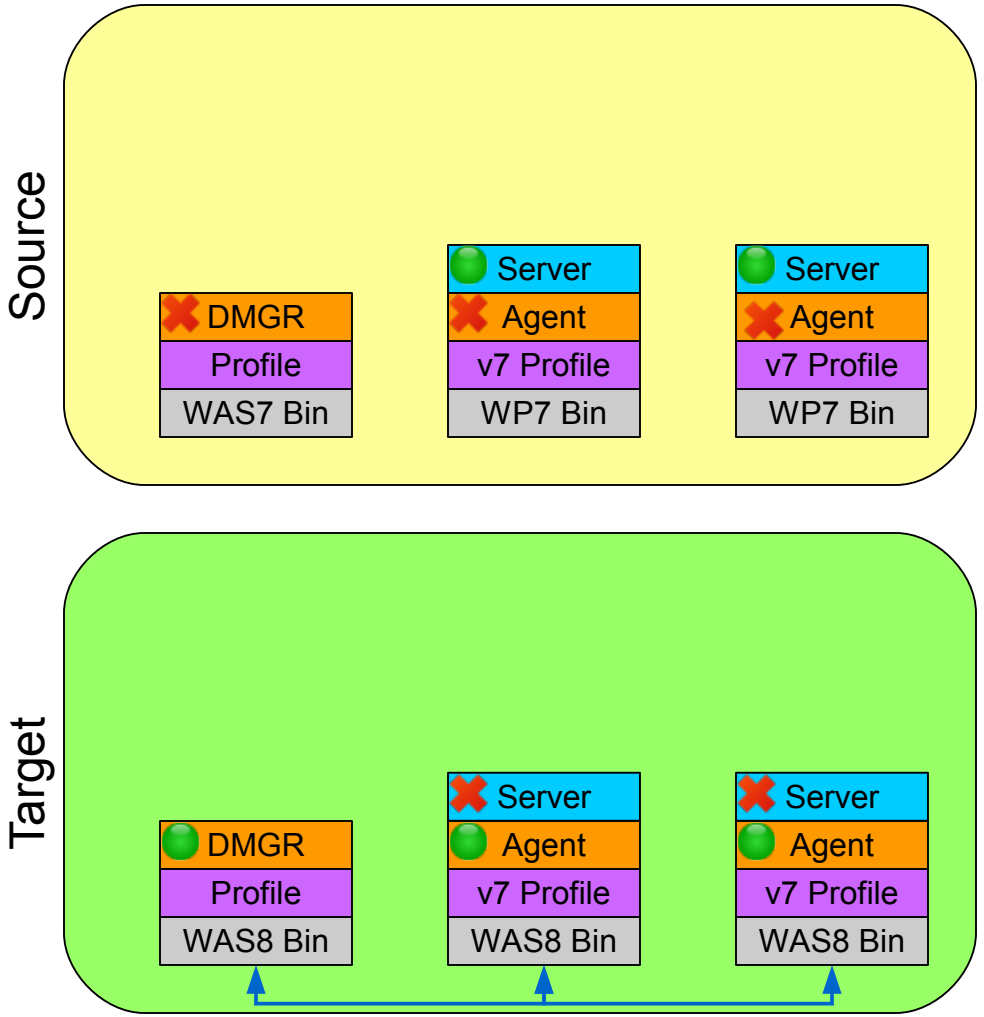
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
  - Start DMGR/Node Agents
  - Update ports in ISC
  - Fully re-synchronize
  - Restart DMGR/Node Agents
- Complete node migrations



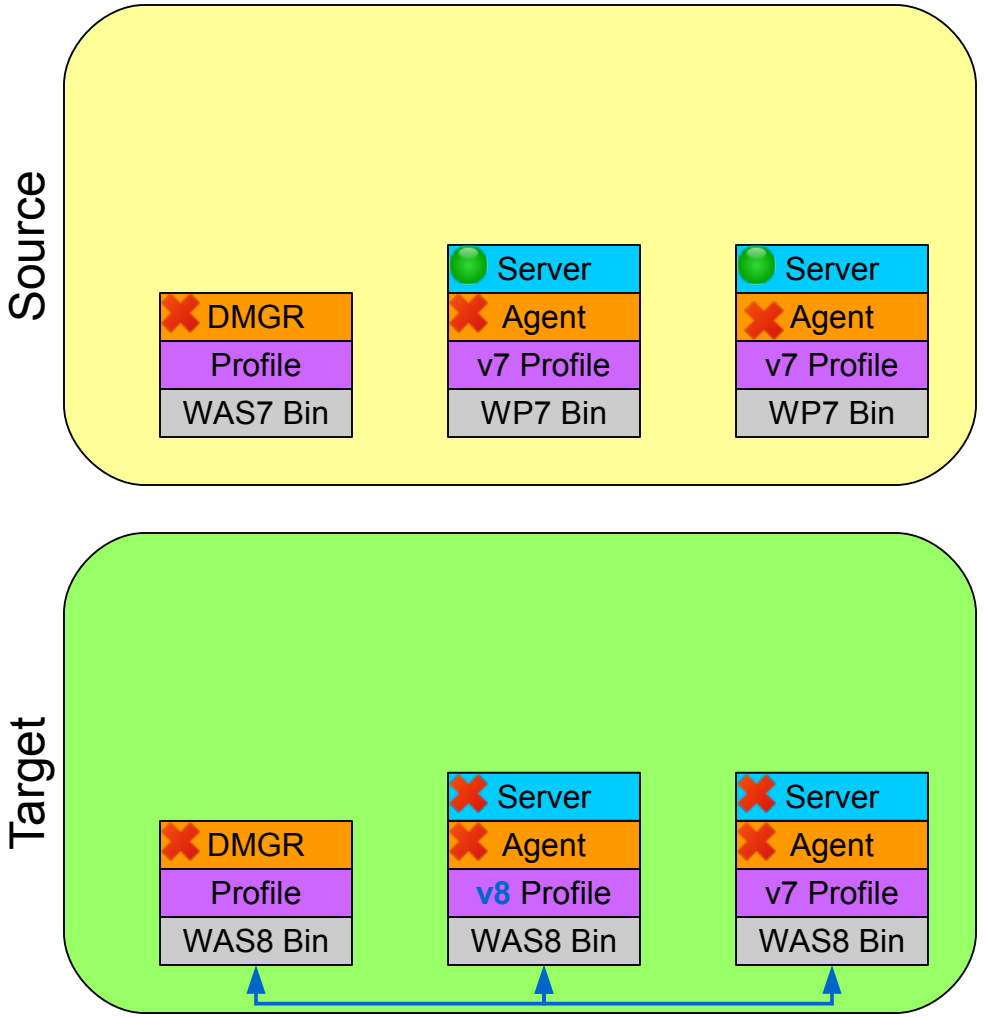
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
  - Start DMGR/Node Agents
  - Update ports in ISC
  - Fully re-synchronize
  - Restart DMGR/Node Agents
- Complete node migrations



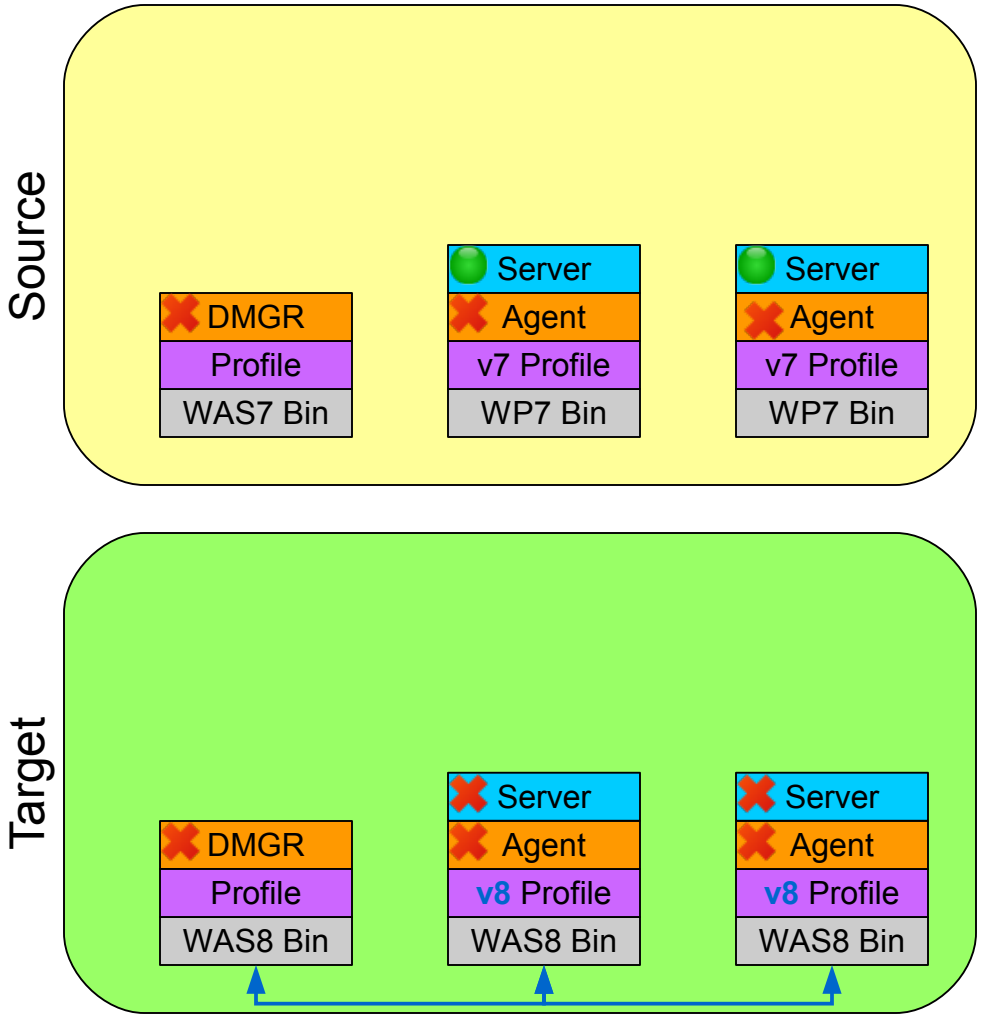
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations
  - Upgrade ConfigEngine
  - Connect to DB Copies
  - Run upgrade-profile
  - Restart Source Env
  - Post migration tasks



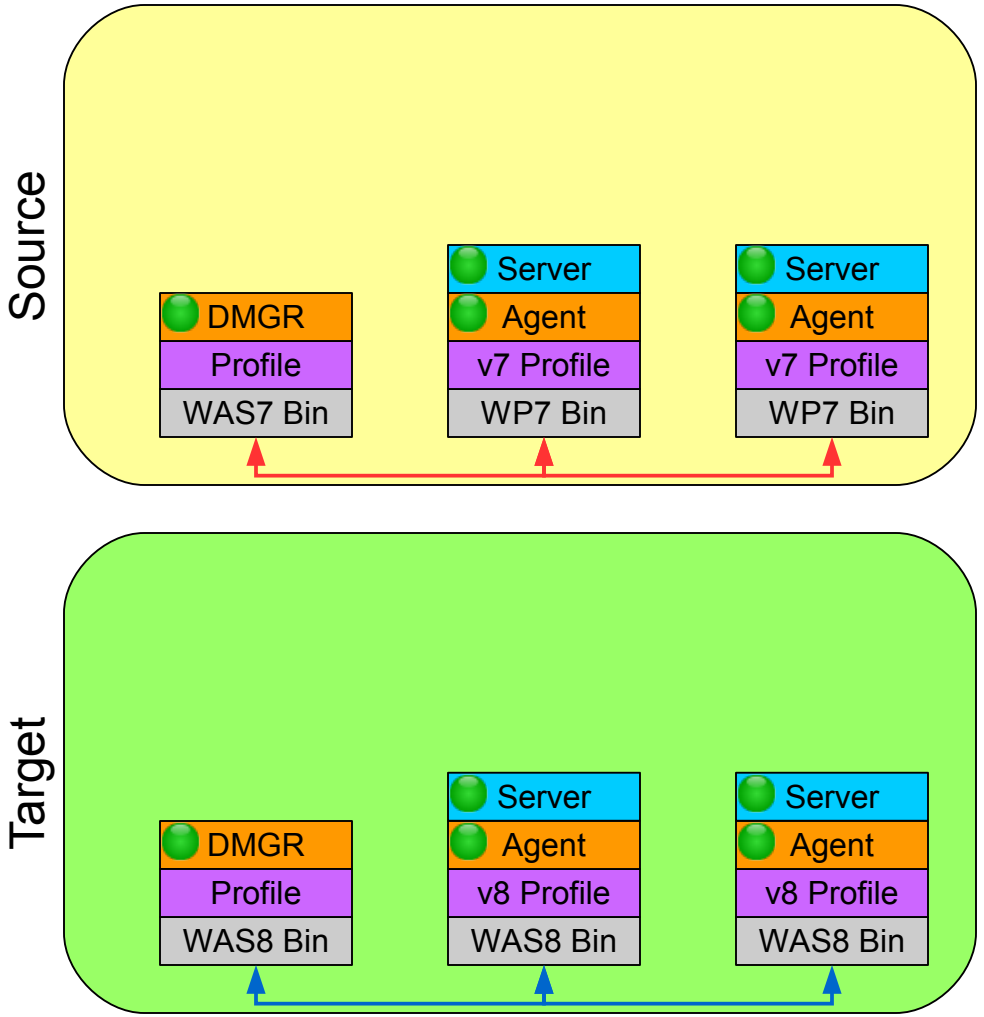
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations
  - Upgrade ConfigEngine
  - Connect to DB Copies
  - Run upgrade-profile
  - Restart Source Env
  - Post migration tasks



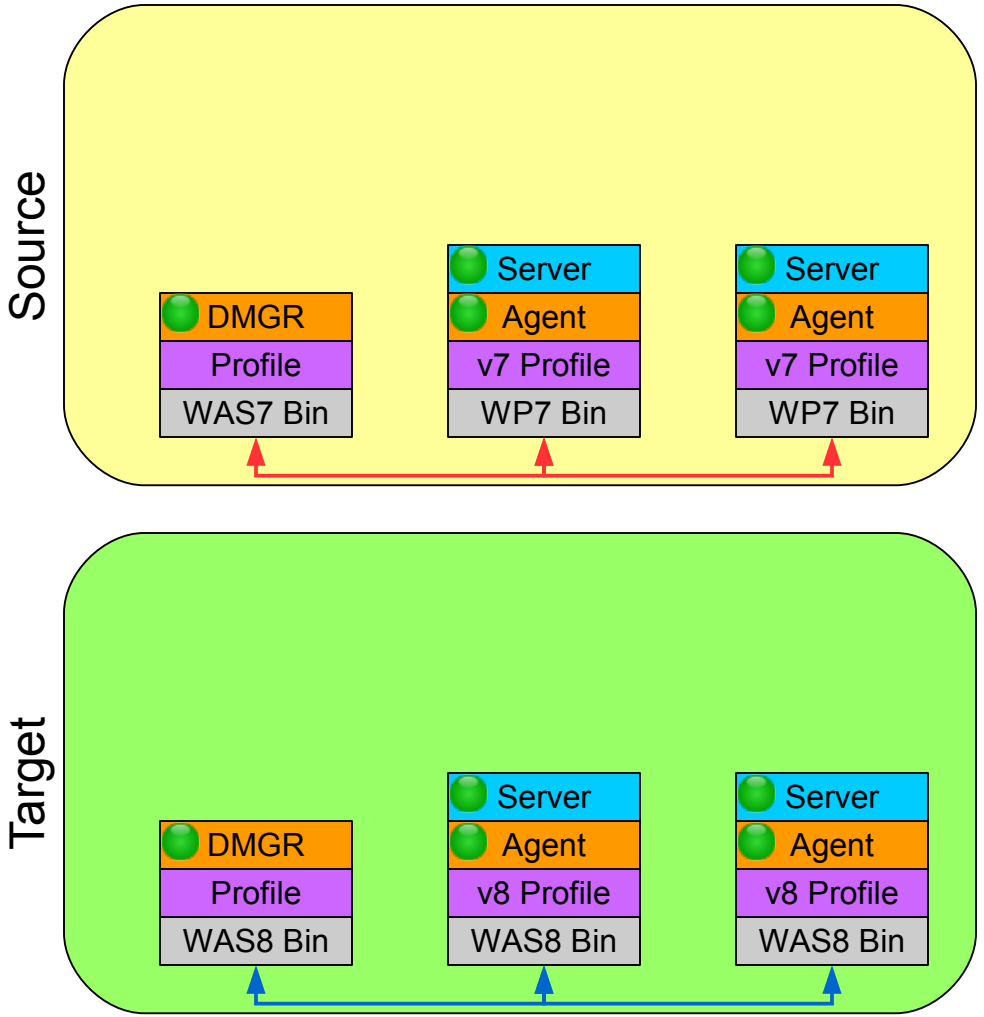
Cluster Migration

- Migrate DMGR profile
- Stop source DMGR and node agents
- Migrate node profiles
- Set up unique ports
- Complete node migrations
  - Upgrade ConfigEngine
  - Connect to DB Copies
  - Run upgrade-profile
  - Restart Source Env
  - Post migration tasks



Best Practices

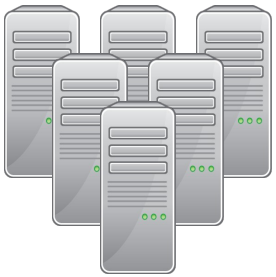
- Make sure AutoSync is disabled
- Make sure the Source DMGR/Node Agents are stopped
- Don't forget to set up unique ports



Migration Strategies

- Scenario 1: Tiered Environment

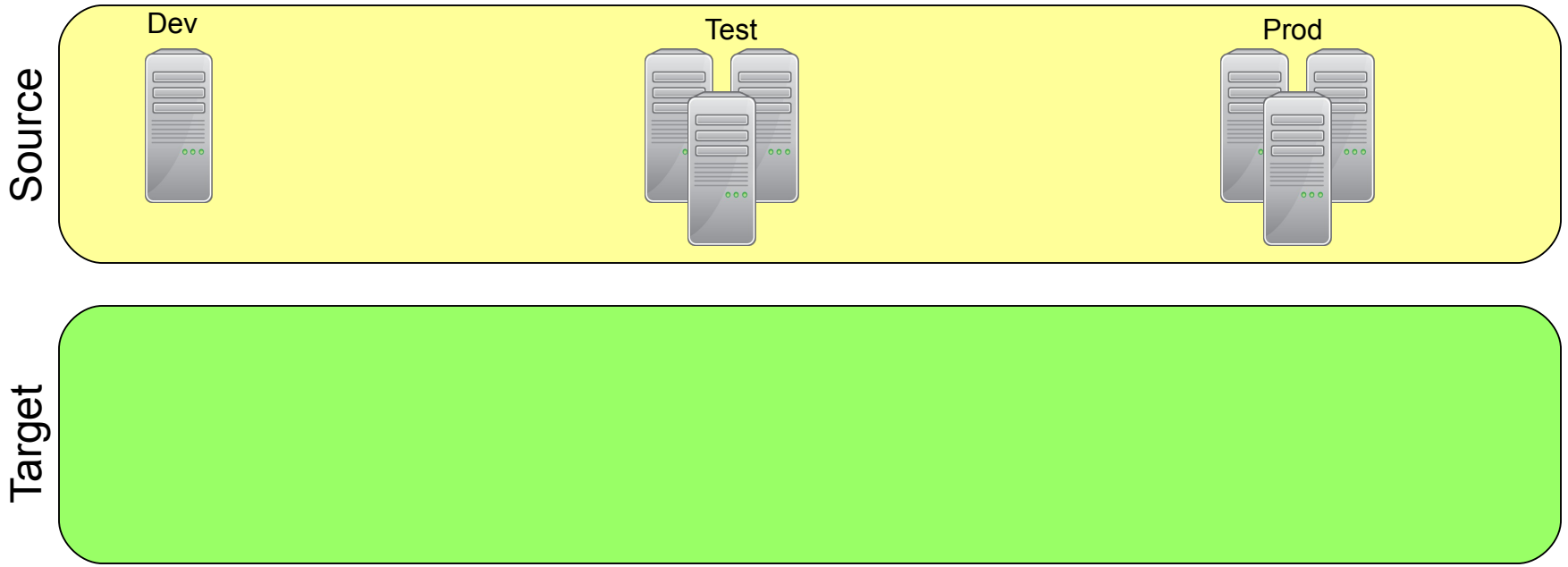
- Scenario 2: Large Cluster



- Scenario 3: Locked-down Environment

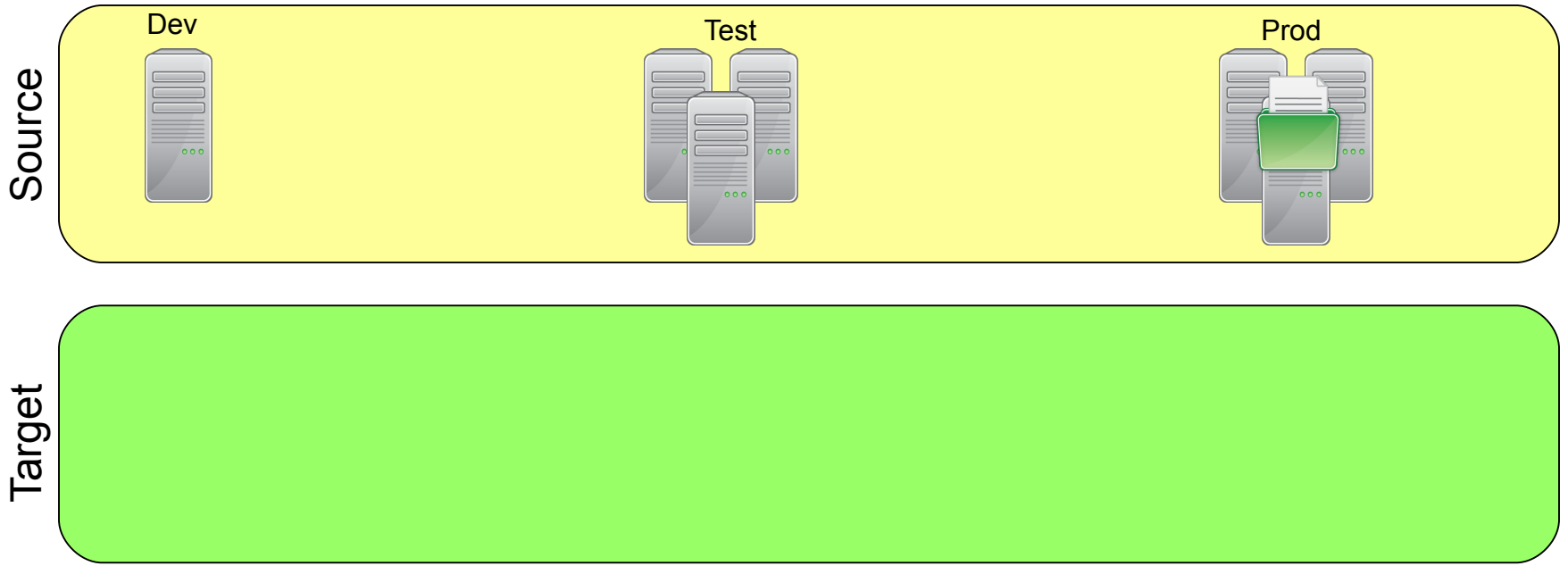
Tiered Environment

- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



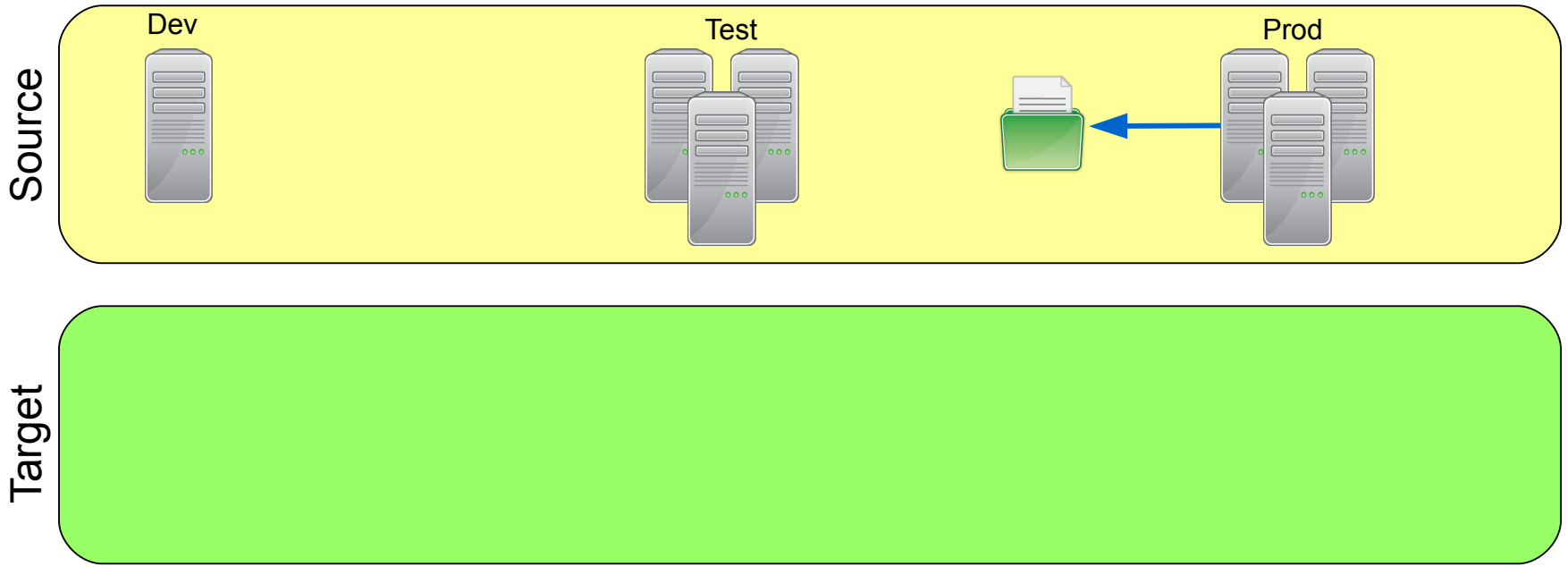
Tiered Environment

- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



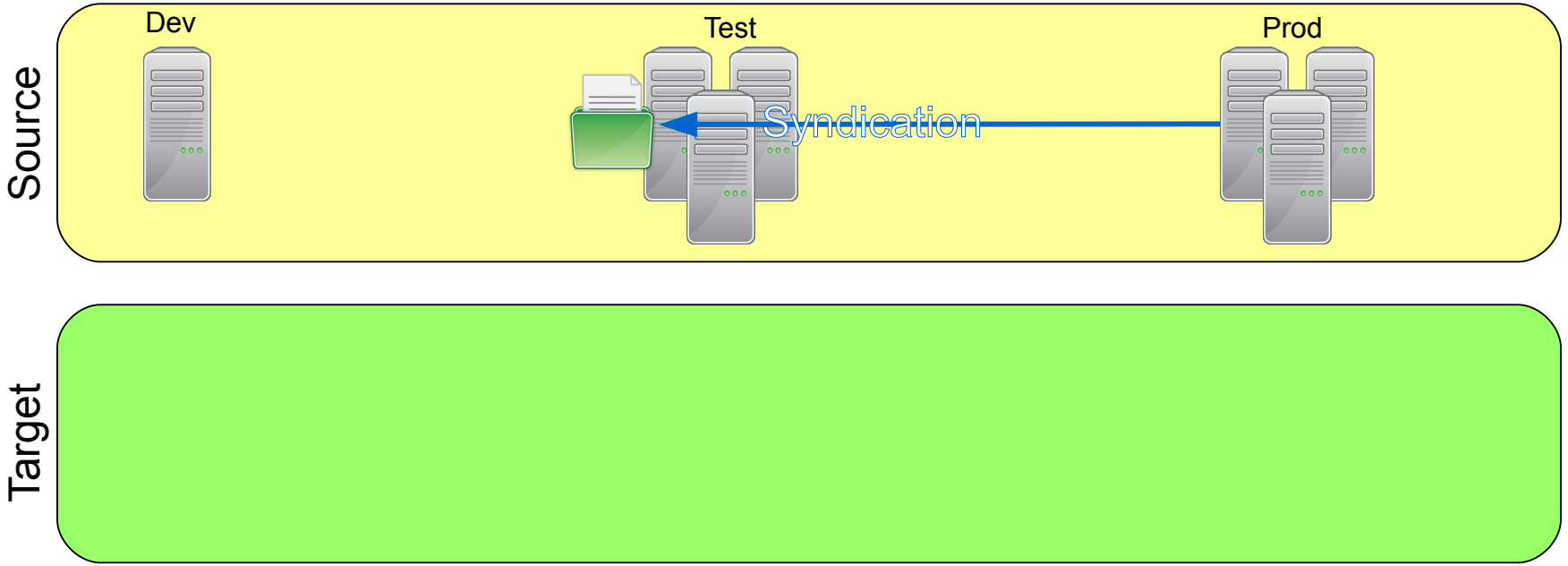
Tiered Environment

- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



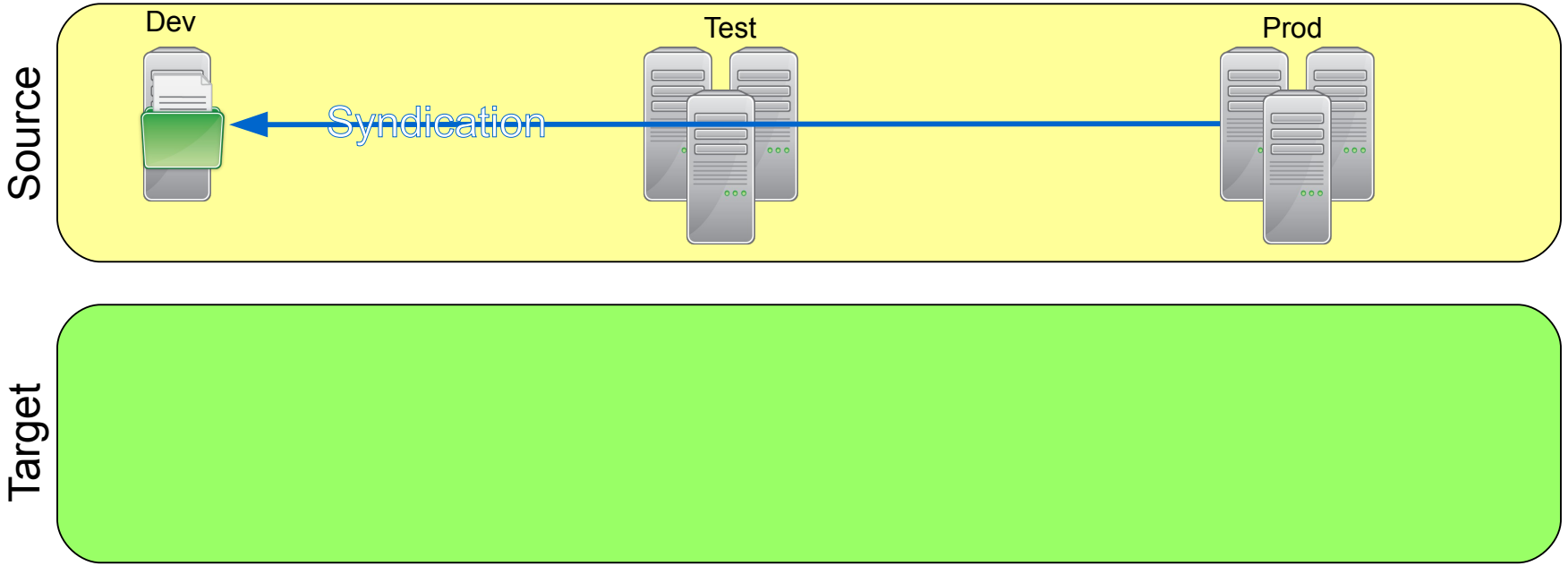
Tiered Environment

- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



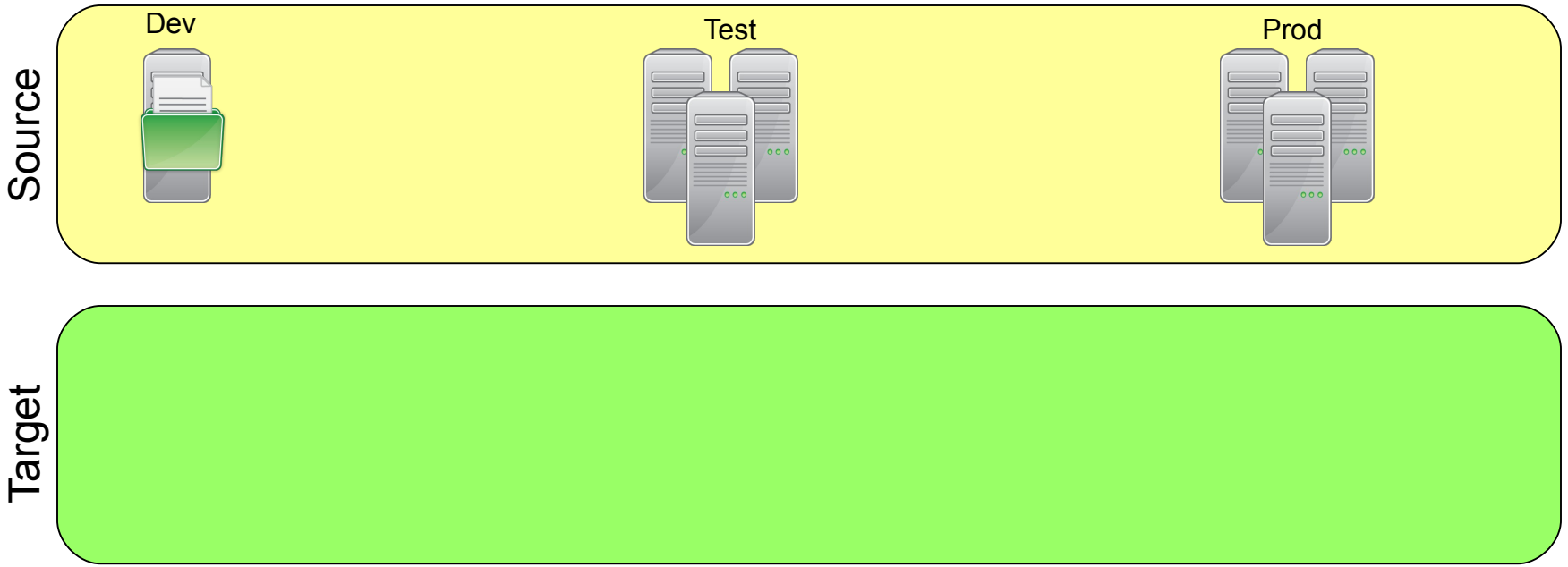
Tiered Environment

- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



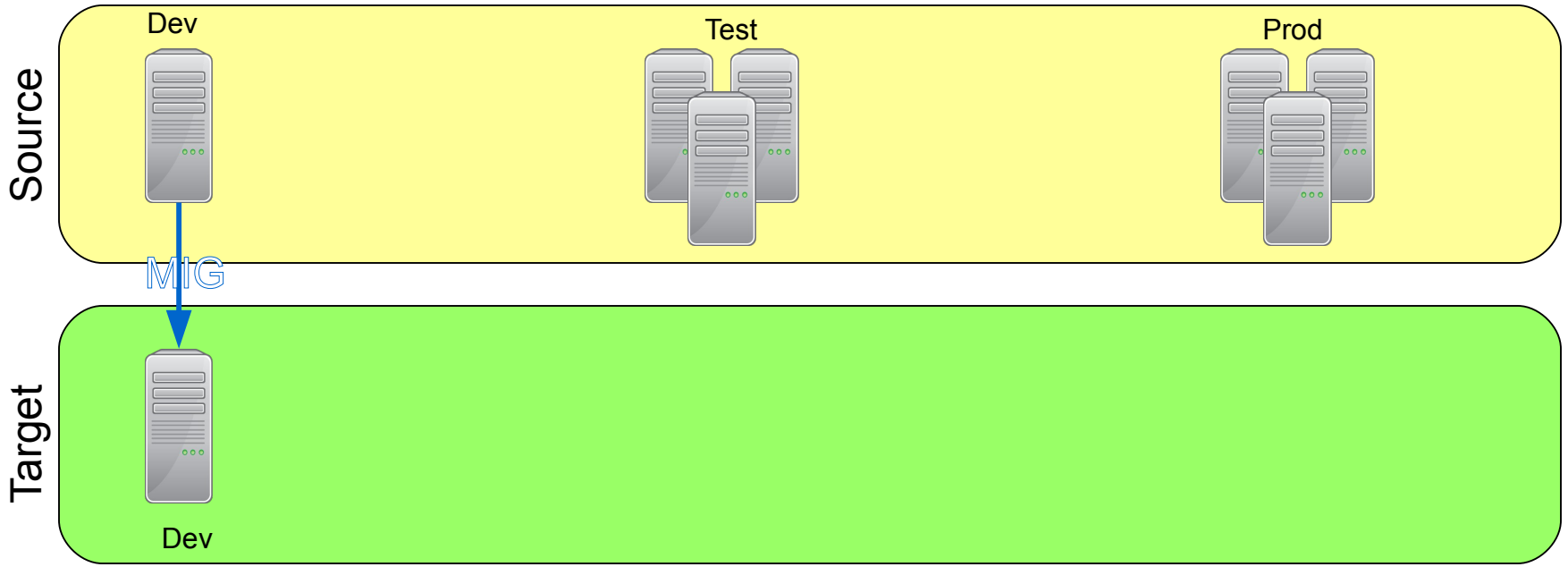
Tiered Environment

- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



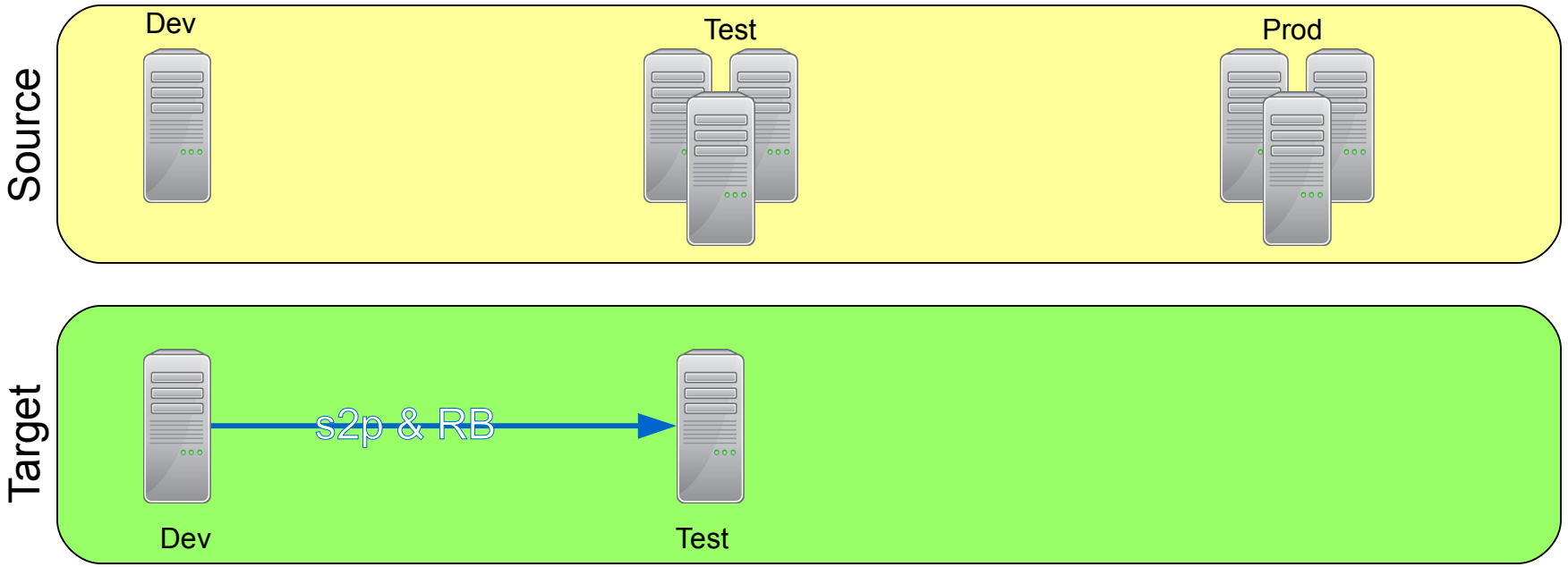
Tiered Environment

- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



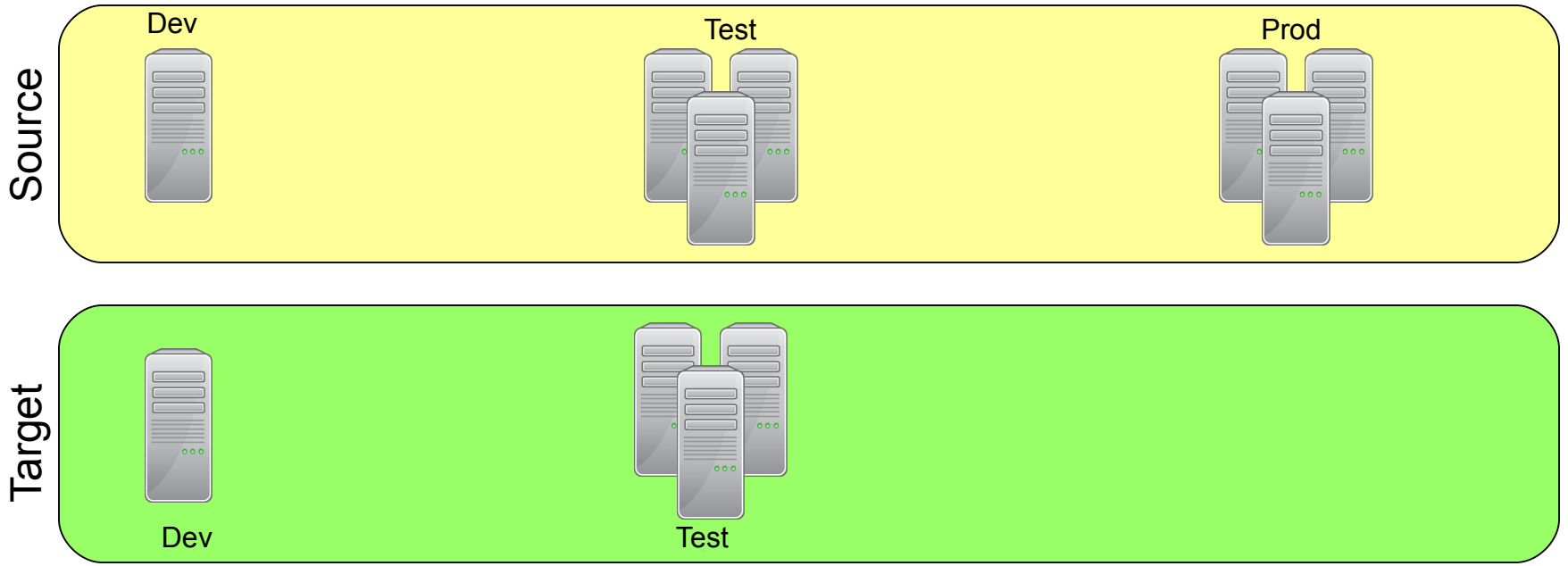
Tiered Environment

- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



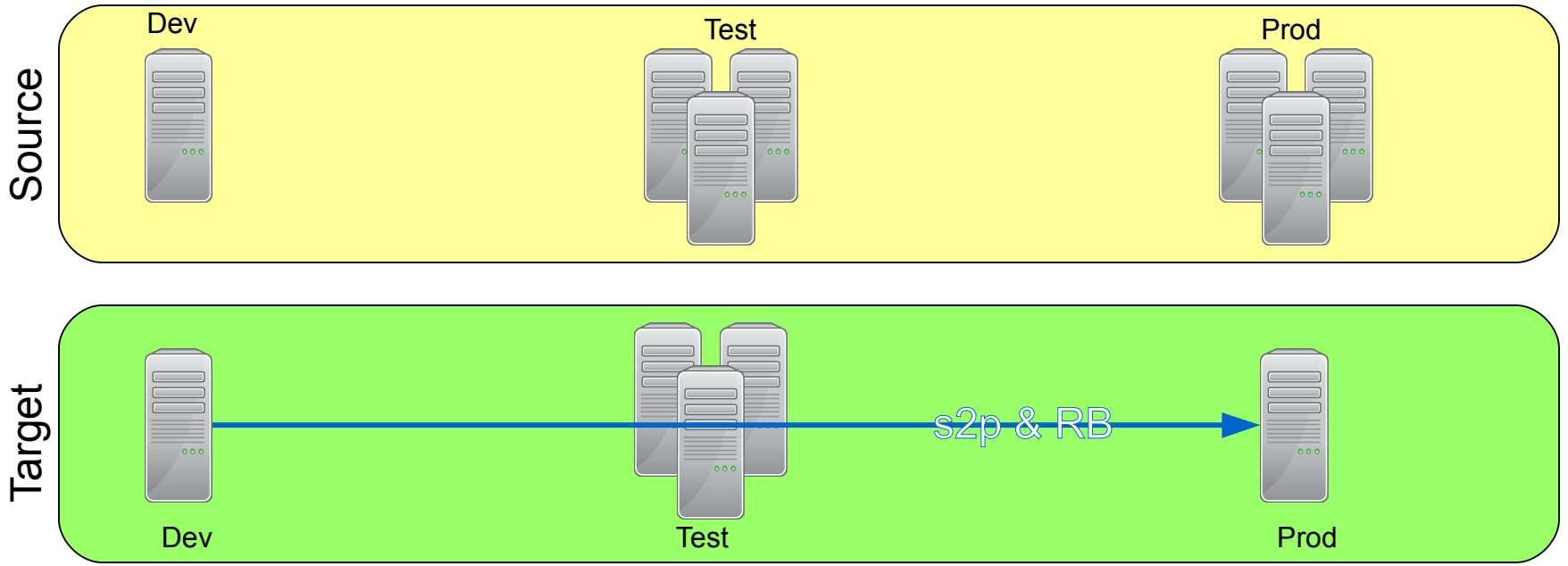
Mixed Tiered Environment

- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



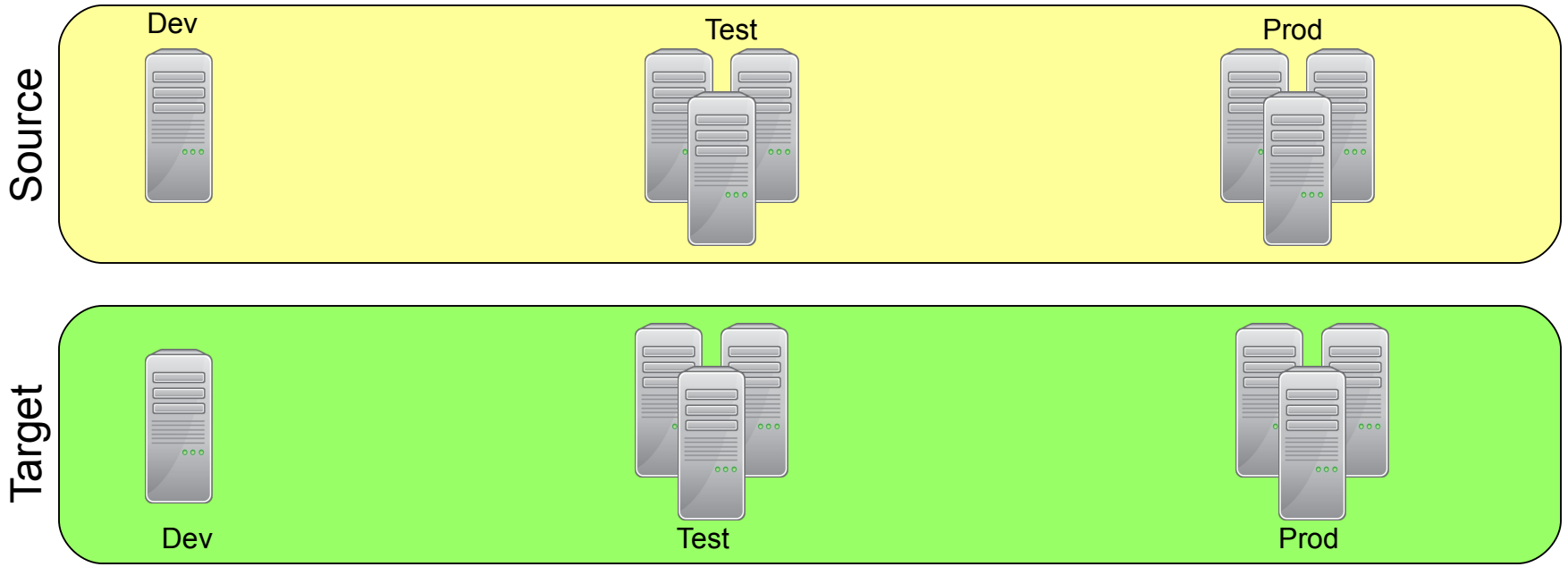
Mixed Tiered Environment

- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



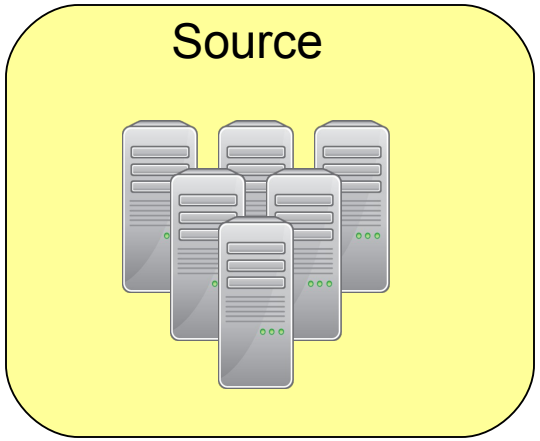
Tiered Environment

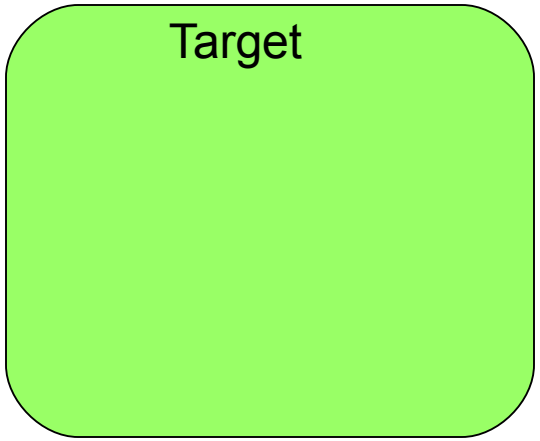
- Make sure lowest tier standalone server has most current content
- Migrate lowest tier
- Build out the rest of the environments using s2p and RB



Large Cluster

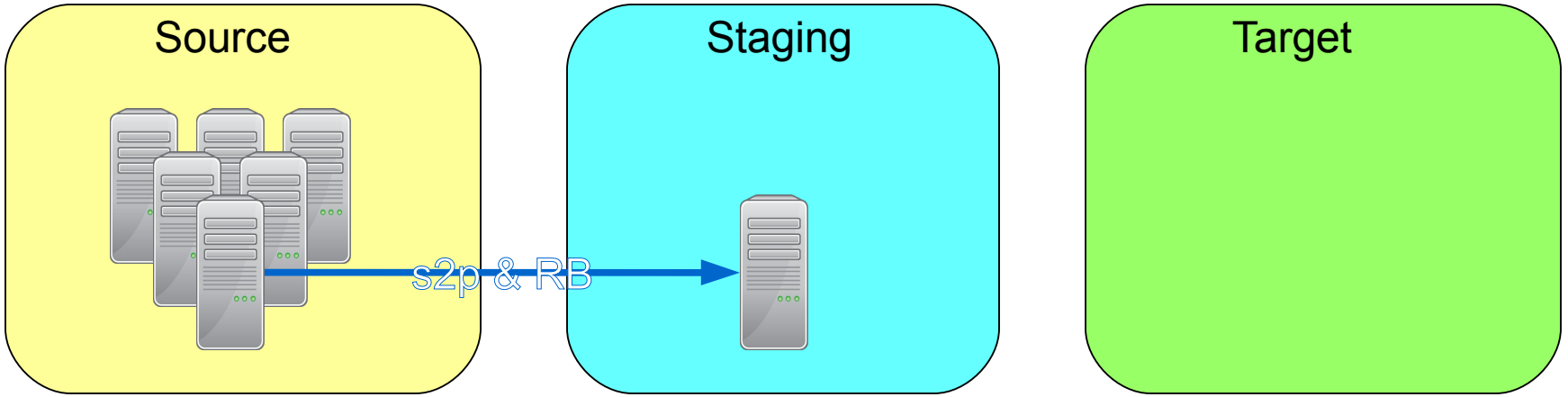
- Create staging server using s2p and RB
- Migrate staging server
- Build out the rest of the cluster





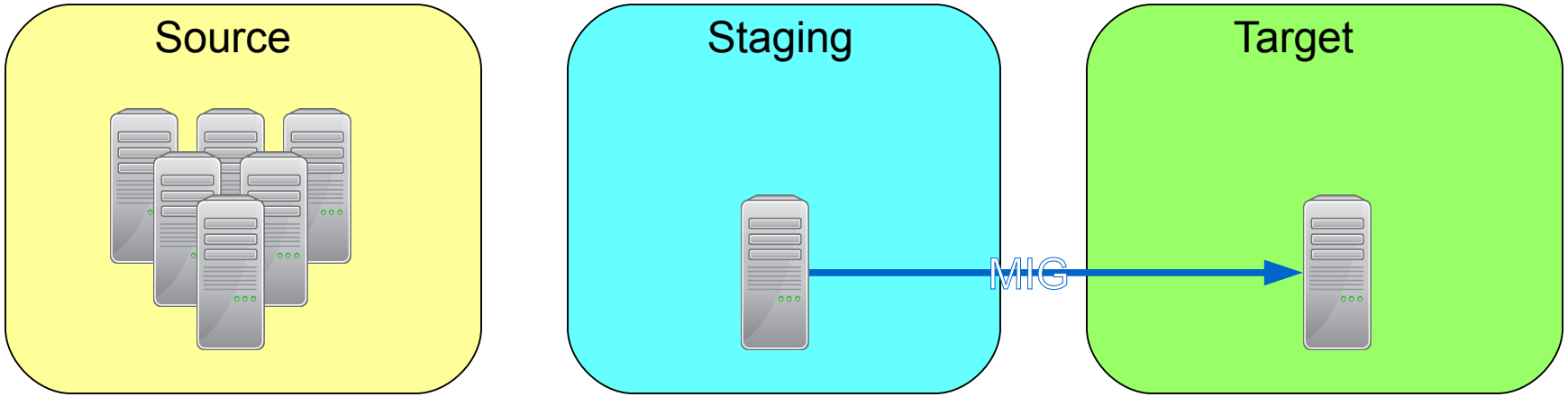
Large Cluster

- Create staging server using s2p and RB
- Migrate staging server
- Build out the rest of the cluster



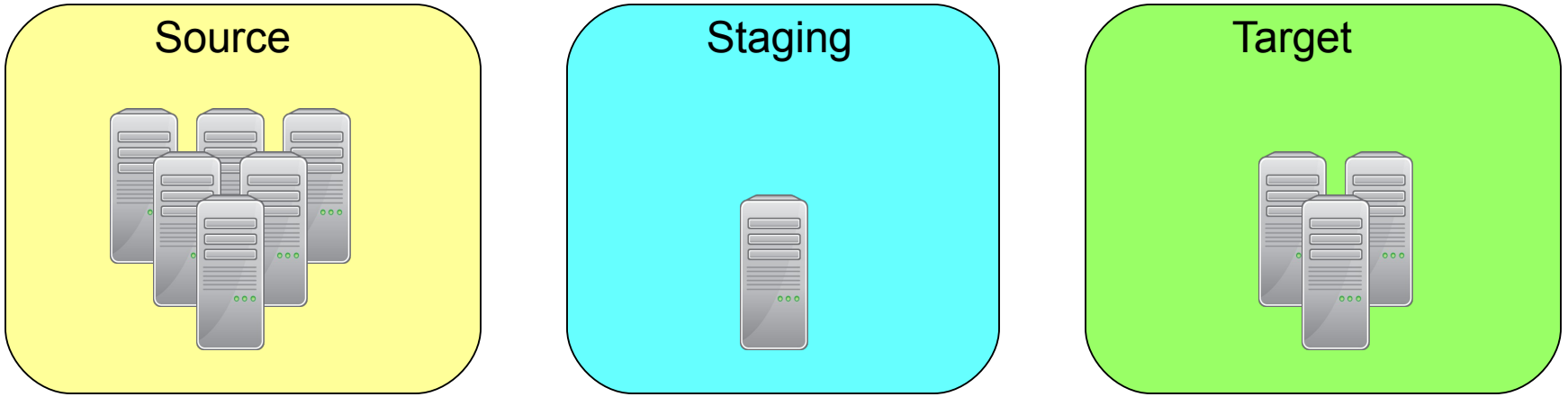
Large Cluster

- Create staging server using s2p and RB
- Migrate staging server
- Build out the rest of the cluster



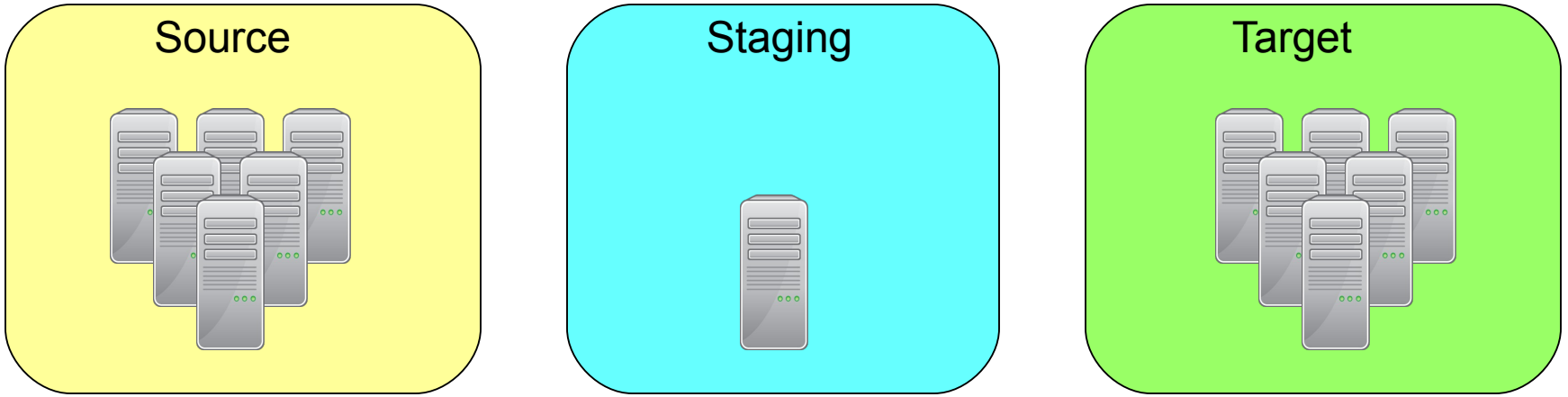
Large Cluster

- Create staging server using s2p and RB
- Migrate staging server
- Build out the rest of the cluster



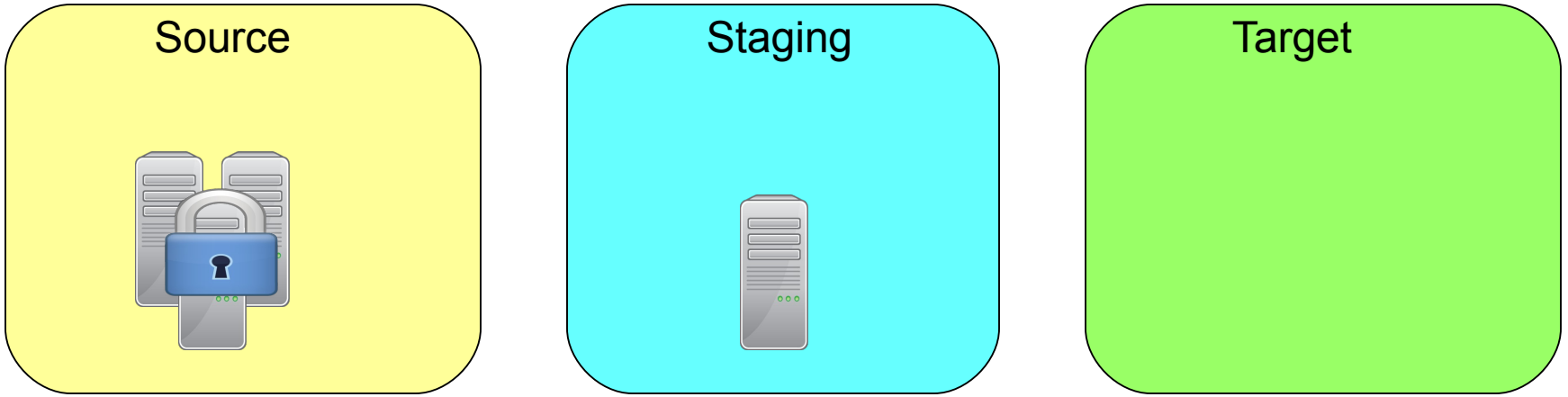
Large Cluster

- Create staging server using s2p and RB
- Migrate staging server
- Build out the rest of the cluster



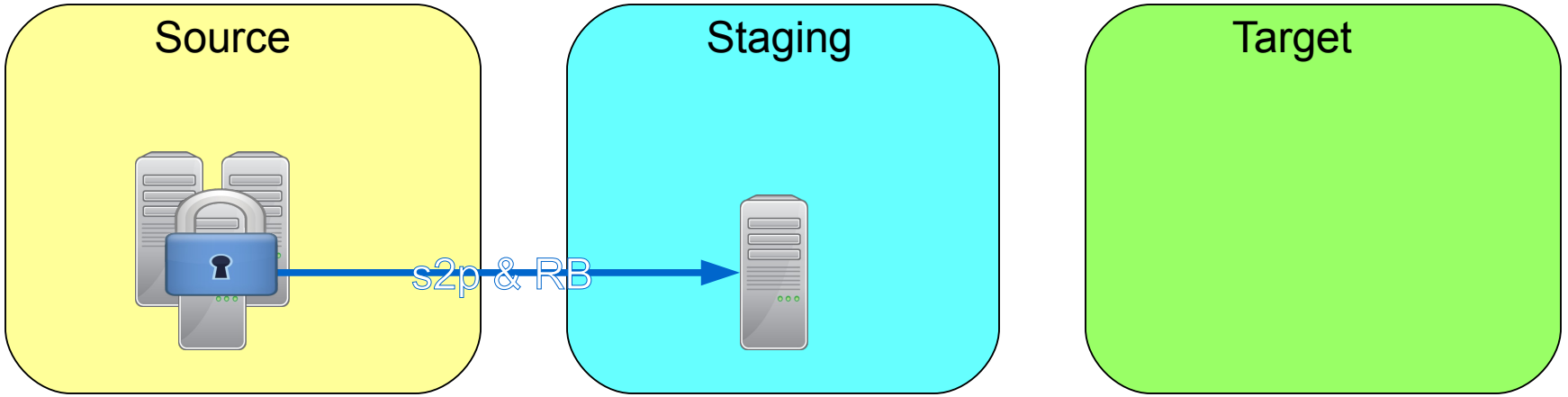
Locked-down Environment

- Create staging server using s2p and RB
- Migrate staging server
- Build out the rest of the cluster



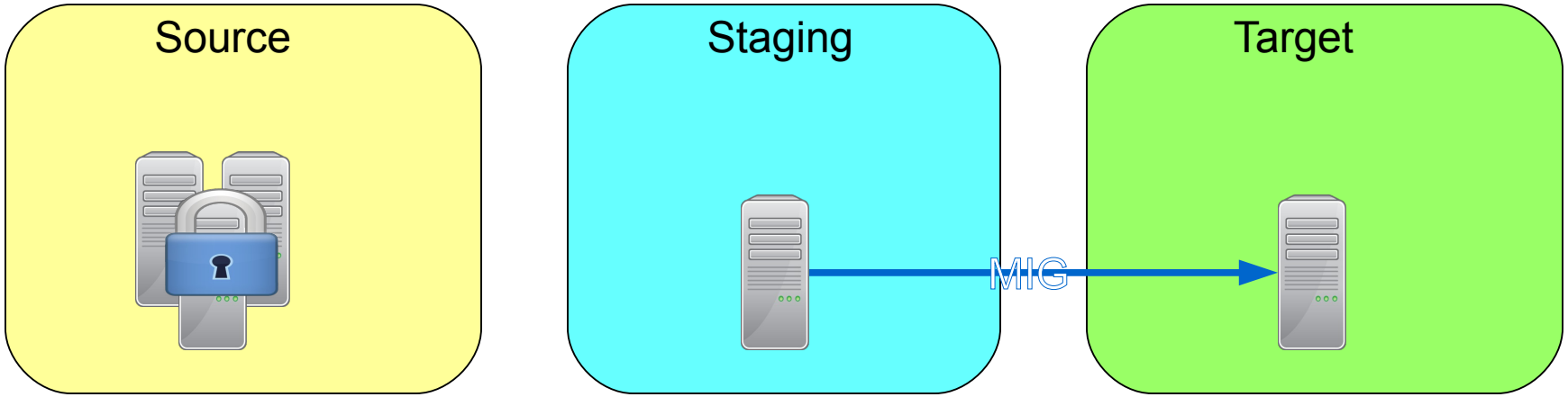
Locked-down Environment

- Create staging server using s2p and RB
- Migrate staging server
- Build out the rest of the cluster



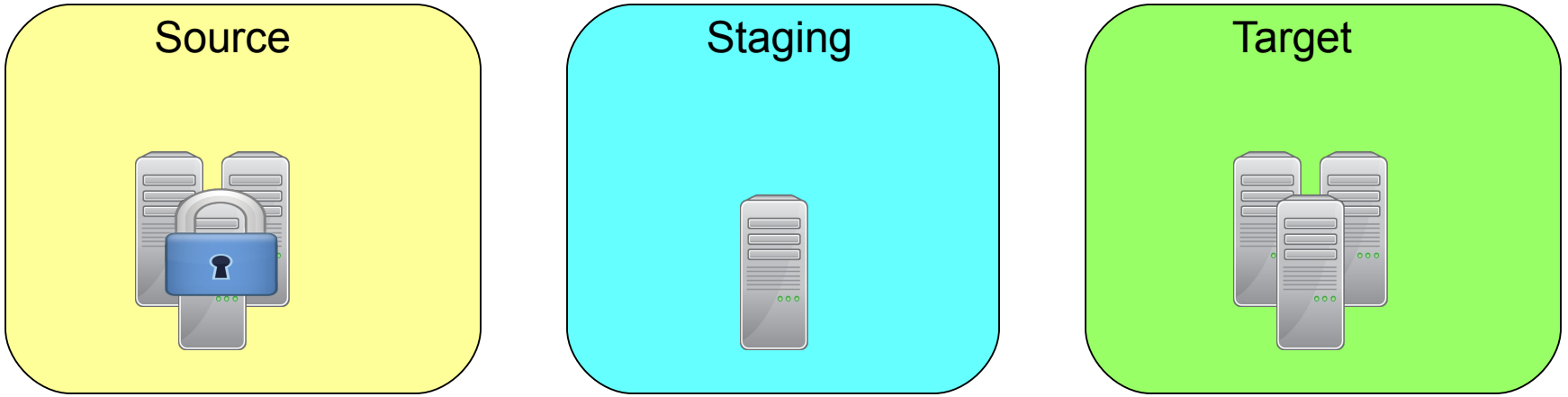
Locked-down Environment

- Create staging server using s2p and RB
- Migrate staging server
- Build out the rest of the cluster



Locked-down Environment

- Create staging server using s2p and RB
- Migrate staging server
- Build out the rest of the cluster

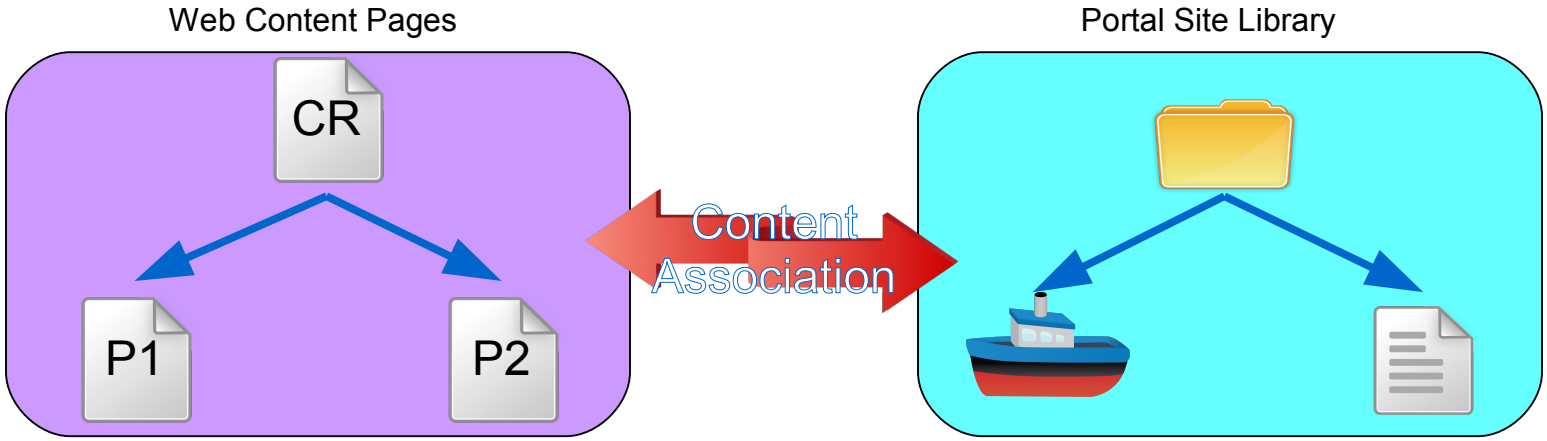


Best Practices

- Keep migration plan simple and do not introduce unnecessary changes
- Use Staging to Production and ReleaseBuilder along with migration to build out the target environments
  - Make sure to consider differences in the environments – such as content, customization, test applications or site libraries – when using this strategy
- Migrate standalone servers and then build out the clusters
- Staging to Production and ReleaseBuilder can be used to create a staging environment for migration
  - This strategy is also used for migrating to IBM PureSystems or a Portal Farm from a cluster
- **Content-only migration is not supported in Portal 8**

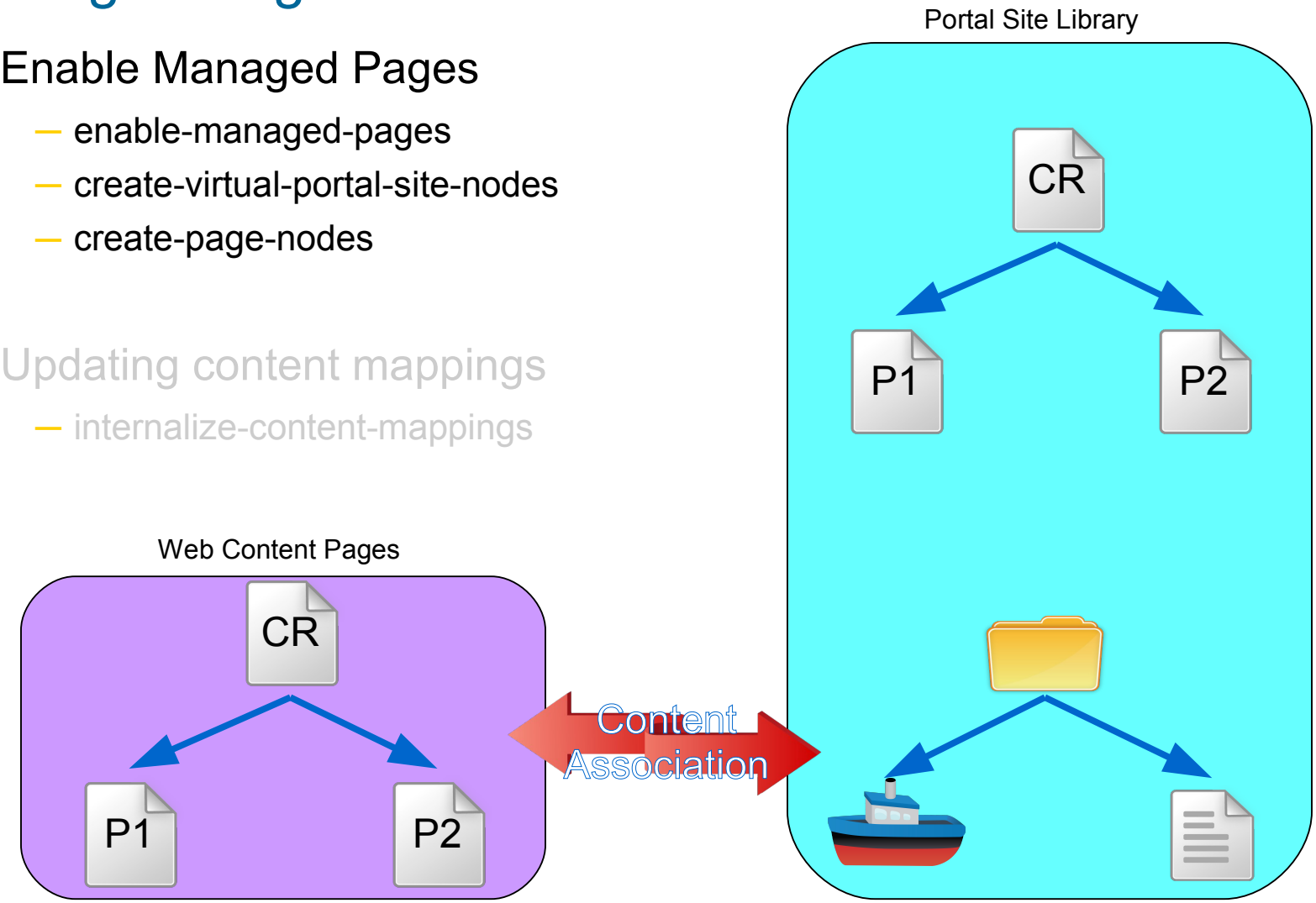
Managed Pages

- Pages and content were maintained separately
  - This could lead to dangling pointers
- Managed Pages treats pages a content items



Managed Pages

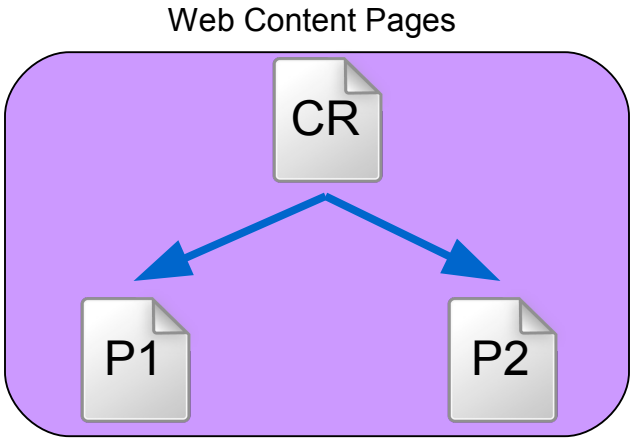
- Enable Managed Pages
  - enable-managed-pages
  - create-virtual-portal-site-nodes
  - create-page-nodes
- Updating content mappings
  - internalize-content-mappings

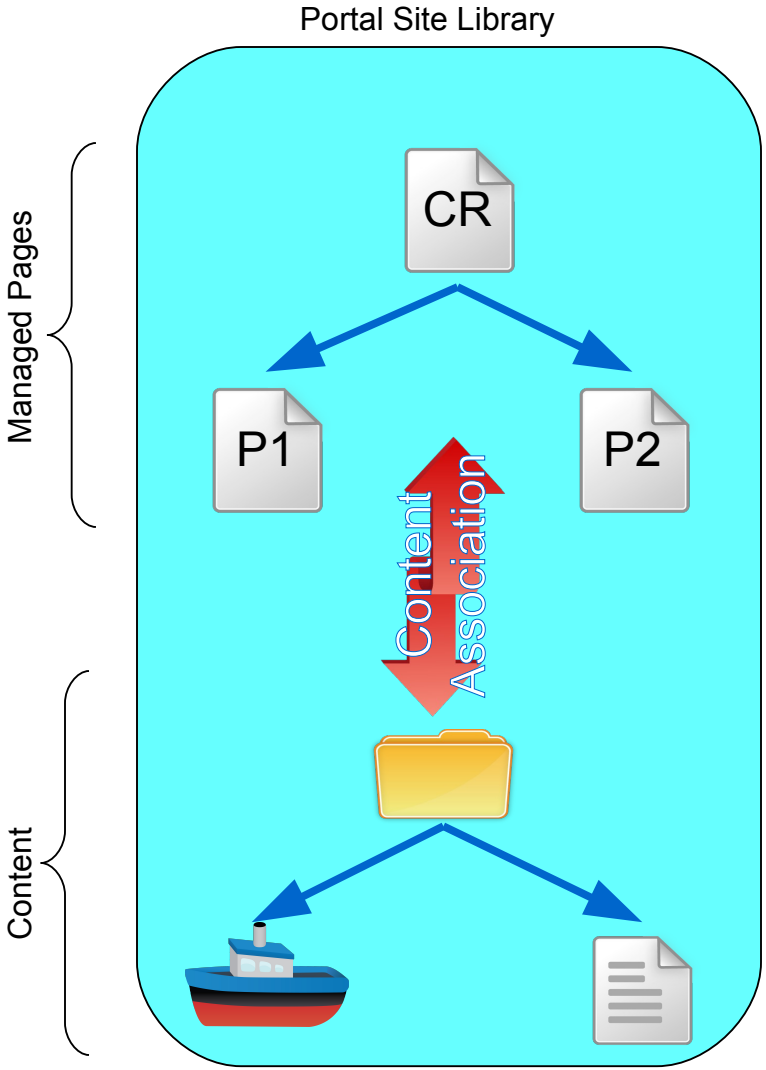


Managed Pages

- Enable Managed Pages
  - enable-managed-pages
  - create-virtual-portal-site-nodes
  - create-page-nodes

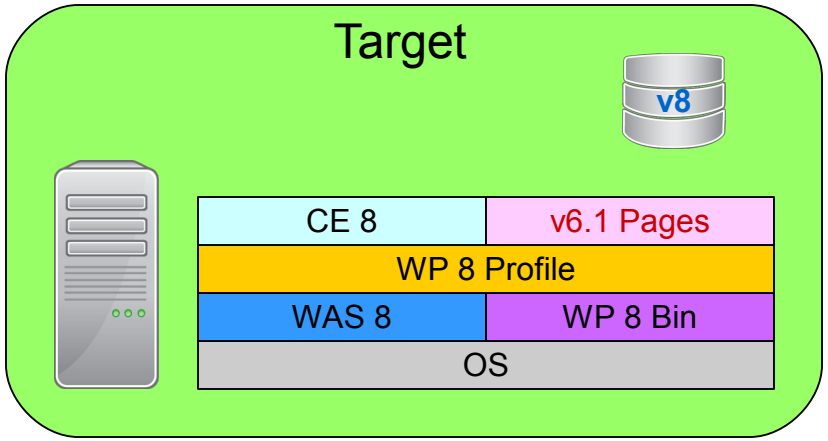
- Updating content mappings
  - internalize-content-mappings





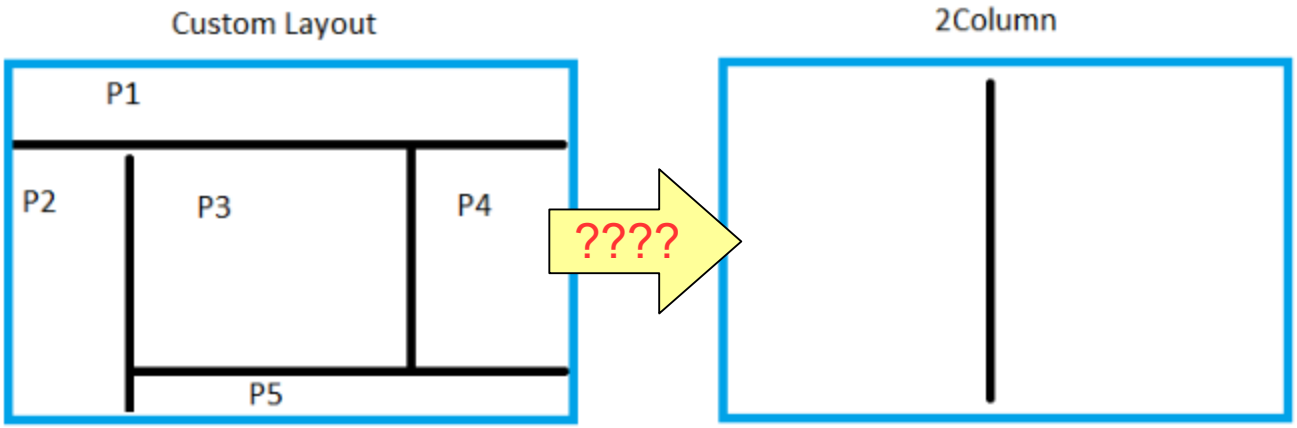
Portal 8 Static Pages

- New page format introduced in Portal 8
- Includes style information in the portlet containers
- Fixed layout which must be defined on the server
- Required to take full advantage of new Portal 8 WCM UI
- Portal 8 themes can only be applied to Portal Static Pages



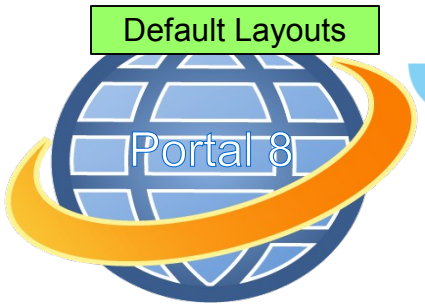
Portal Page Migration

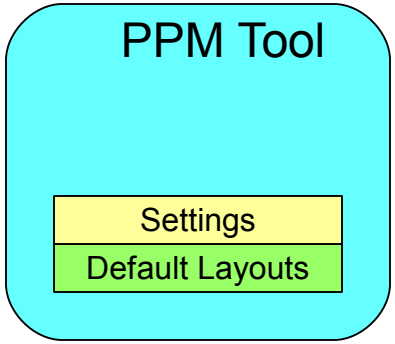
- Updates Page Tree XML exports
- Original pages are not modified
- Default layouts 
- Cannot be fully automated because of Custom Layouts



Portal 8 Static Pages

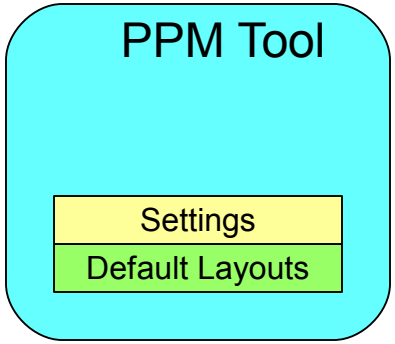
- Export pages
- Define custom layout definitions for the tool
- Create custom layouts for Portal 8
- Setup the tool
- Run analysis
- Run migration
- Import new pages

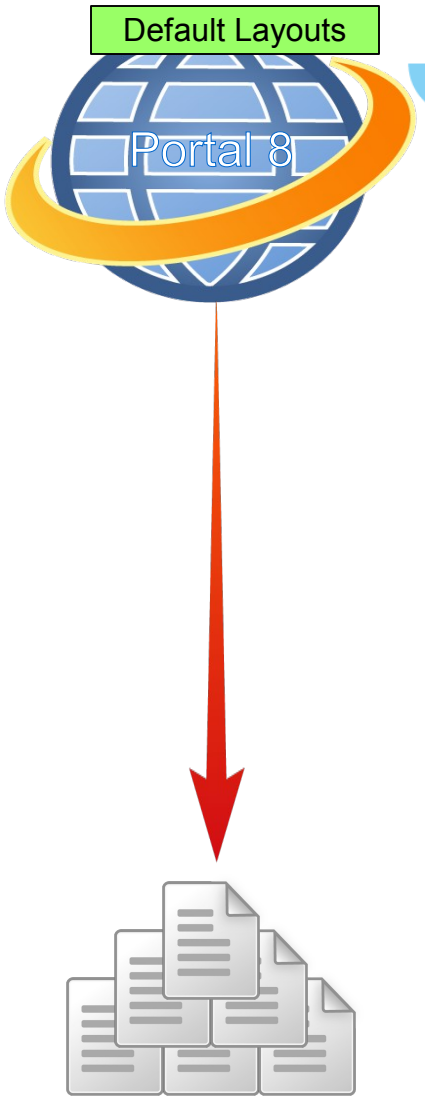




Portal 8 Static Pages

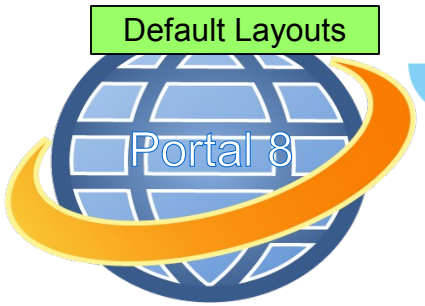
- Export pages
- Define custom layout definitions for the tool
- Create custom layouts for Portal 8
- Setup the tool
- Run analysis
- Run migration
- Import new pages

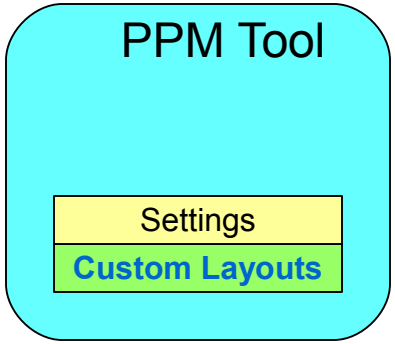


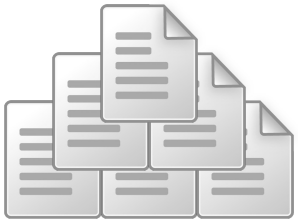


Portal 8 Static Pages

- Export pages
- Define custom layout definitions for the tool
- Create custom layouts for Portal 8
- Setup the tool
- Run analysis
- Run migration
- Import new pages

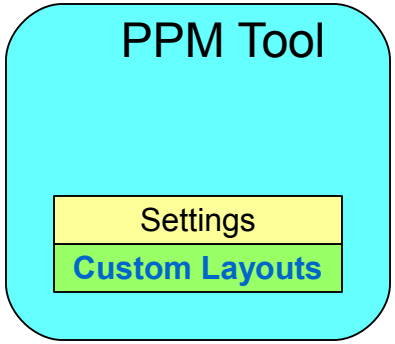


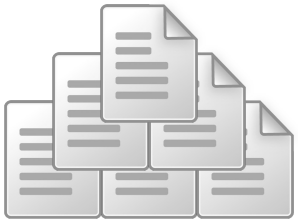




Portal 8 Static Pages

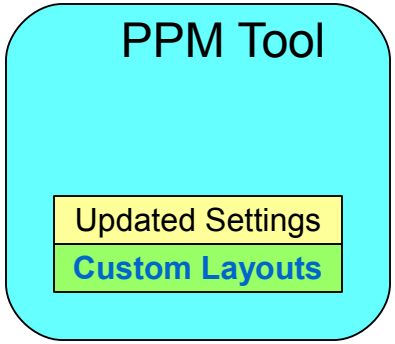
- Export pages
- Define custom layout definitions for the tool
- Create custom layouts for Portal 8
- Setup the tool
- Run analysis
- Run migration
- Import new pages

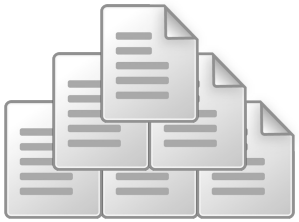




Portal 8 Static Pages

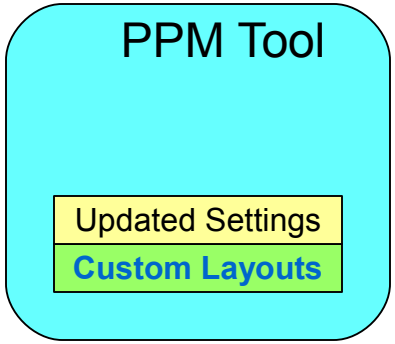
- Export pages
- Define custom layout definitions for the tool
- Create custom layouts for Portal 8
- Setup the tool
- Run analysis
- Run migration
- Import new pages





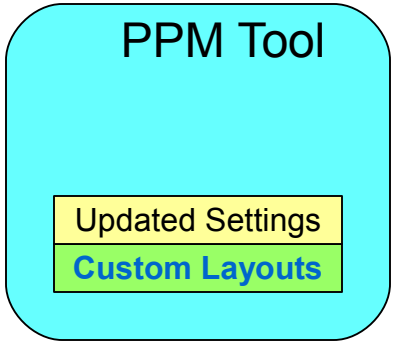
Portal 8 Static Pages

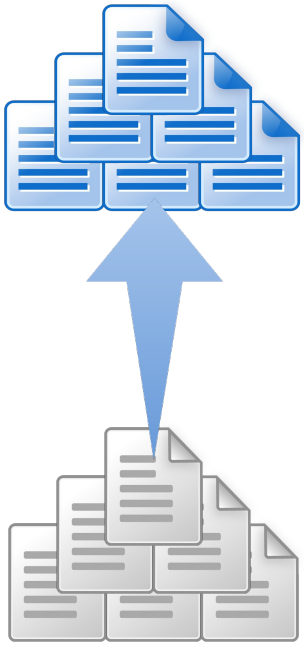
- Export pages
- Define custom layout definitions for the tool
- Create custom layouts for Portal 8
- Setup the tool
- Run analysis
  - Repeat steps if needed
- Run migration
- Import new pages



Portal 8 Static Pages

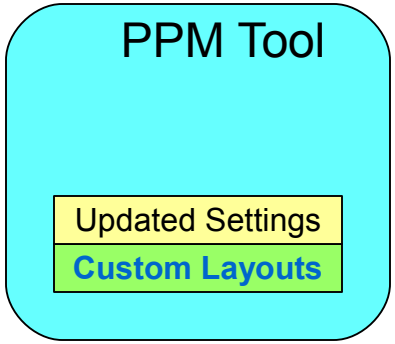
- Export pages
- Define custom layout definitions for the tool
- Create custom layouts for Portal 8
- Setup the tool
- Run analysis
- Run migration
- Import new pages

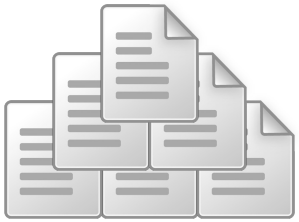




Portal 8 Static Pages

- Export pages
- Define custom layout definitions for the tool
- Create custom layouts for Portal 8
- Setup the tool
- Run analysis
- Run migration
- Import new pages
  - Exported pages not modified





Resources

- Portal 8 Wiki
- http://www-10.lotus.com/ldd/portalwiki.nsf/xpDocViewer.xsp?lookupName=IBM+WebSphere+Portal+8+Product+Documentation#action=openDocument&content=catcontent&ct=prodDoc
- Recommended fixes
- http://www-01.ibm.com/support/docview.wss?uid=swg27007603
- Staging to Production
- http://www-10.lotus.com/ldd/portalwiki.nsf/xpDocViewer.xsp?lookupName=IBM+WebSphere+Portal+8+Product+Documentation#action=openDocument&res_title=Staging_to_production_wp8&content=pdcontent
- Release Builder
- http://www-10.lotus.com/ldd/portalwiki.nsf/xpDocViewer.xsp?lookupName=IBM+WebSphere+Portal+8+Product+Documentation#action=openDocument&res_title=ReleaseBuilder_wp8&content=pdcontent
- Portal Page Migration Tool
- https://greenhouse.lotus.com/plugins/plugincatalog.nsf/assetDetails.xsp?action=editDocument&documentId=5190EB71148A2EC485257B2400510466

Press *1 on your telephone to ask a question.

Visit our Support Technical Exchange page or our Facebook page for details on future events.

To help shape the future of IBM software, take this quality survey and share your opinion of IBM software used within your organization: https://ibm.biz/BdxqB2

IBM Collaboration Solutions Support page
http://www.facebook.com/WebSpherePortalSupport

WebSphere Portal
http://twitter.com/PortalSupport

