

IBM Commerce

WebSphere Portal – Migration to v8.5



Hermann Hübler, Digital Experience Lab
Services

14.-15. Juni 2016

WebSphere Portal Migration

Is ...

- not magic
- not rocket science

but ...

- can be a bit complex (depending on the environment)

therefore ...

- requires planning

to ...

- enable your environment for lots of new features and capabilities
- make your environment ready for the future requirements

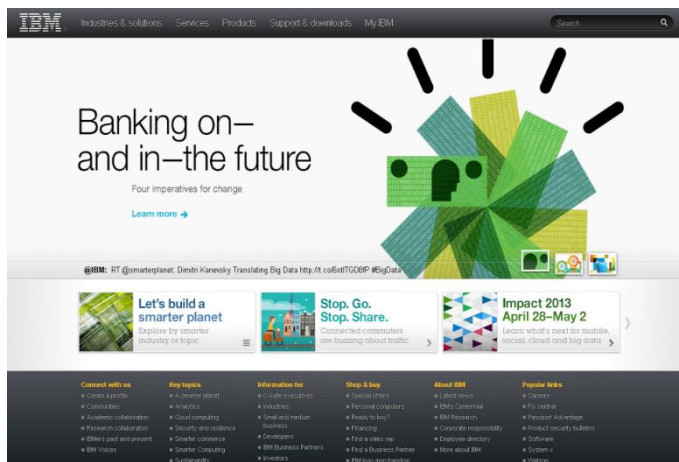
WebSphere Portal Migration - overview

- The first goal of Portal Migration is to move all content, applications, and pages from the source to target environment
 - 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.
 - Migration can follow the automated process (profile migration) or the manual process (manual migration)
- The second goal is to enable all the native new features and capabilities

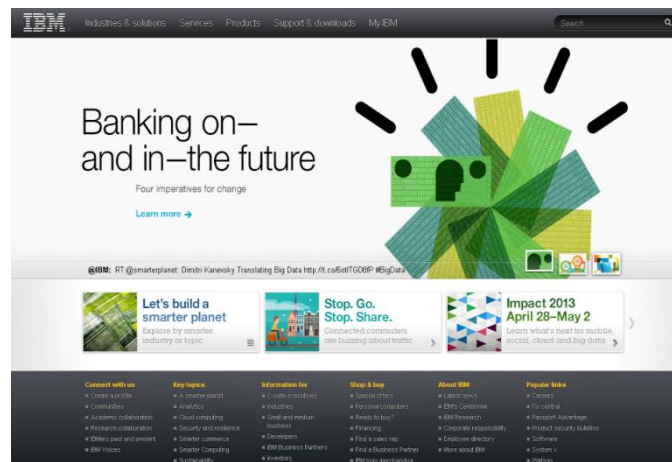
WebSphere Portal Migration - overview

- The first goal of Portal Migration is to move all content, applications, and pages from the source to target environment
 - Maintaining the same look and feel of the original website.

Before



After



WebSphere Portal Migration - overview

- Migration can be performed locally or remote. Remote migration is preferred.
- 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.
 - v7 → v8.5 is supported
 - v8 → v8.5 is supported
 - v6.1 → v8.5 is not supported using profile migration but **might be possible** following **manual** migration process

WebSphere Portal Migration - overview

- Migration is not cross platform.
 - v7 on Linux → v8.5 on Linux is supported
 - v7 on Windows → v8.5 on Linux (or any other OS except Windows) 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.

WebSphere Portal Migration - unsupported features (excerpt)

- IBM Portlet API
 - No longer supported
- Web Clipper
 - Use the Virtual Web Application Manager portlet instead.
- Composite applications
- IBM themes from a previous version
 - Supported if migrated to WebSphere Portal v8.5 themes
 - PageBuilder2 and 7.0.0.2 themes
 - No longer supported themes are
 - Portal, PortalWeb2 and Tab Menu - Page Builder

Note: These themes are migrated as is to WebSphere Portal Version 8.5. However, they may no longer work and are no longer supported.

See: https://www-01.ibm.com/support/knowledgecenter/SSHRKX_8.5.0/mp/reference/intr_depc.dita?lang=en
for full list of deprecated and unsupported features

WebSphere Portal Migration - deprecated features (excerpt)

- Shared pages
- Enabler, Builder, and Mashups components and API
- Stand-alone LDAP user registry
 - Configure the federated LDAP user registry instead.
- Active Credentials are deprecated from the Credential Vault portlet.
 - Passive credentials are still available.
- Parallel Portlet rendering
- LTPA version 1 token support
- Tagging and rating dialog and inline widgets
 - Replaced by a new pair of interactive inline widgets. The new widgets combine the functions of the separate widgets of earlier portal versions.
- URL mappings are deprecated but still working. Need to [install urlmapping.war manually](#) to manage

WebSphere Portal Migration – WebSphere Application Server (WAS) related

- Migrating WebSphere Portal implies an upgrade to WebSphere Application Server as well
 - WebSphere Portal v8.5 requires [WAS v8.5.5](#)
 - See [the following link](#) for further hardware and software requirements
- Upgraded environment will implicitly benefit from enhancements
 - Change in JDK level you are running (support for 1.7.x)
 - Change in JEE version being supported (JEE 7 on WAS v8.5 vs. JEE6 on WAS v8.0)
 - JSF implementations
 - Intelligent management
 - Etc. Etc.

**Need to check the
WebSphere Application
Server deprecation and
unsupported feature list as
well**

WebSphere Portal Migration options

■ Automated migration

- IBM supported migration path
- Driven by configuration wizard
- Is intended to provide a one-size fits all migration
- Brings over the entire source site to the target environment
 - Artefacts must be under WebSphere's control

WebSphere Portal Migration options

■ Manual migration

- You start with a newly installed, fresh installation
 - Then bring over the configuration and artifacts from your source environment
 - Uses administrative standard tools such as wsadmin, XMLAccess, and ConfigEngine to move/transfer your artefacts
 - Or simply redeploy your portal page using your automation tools
 - How will portal settings be migrated when doing manual migration (portlet preferences, PAC, etc)?
- You might end up with different unique IDs depending if you do an automated or a manual migration
- Be careful when migrating different environments with a different approach

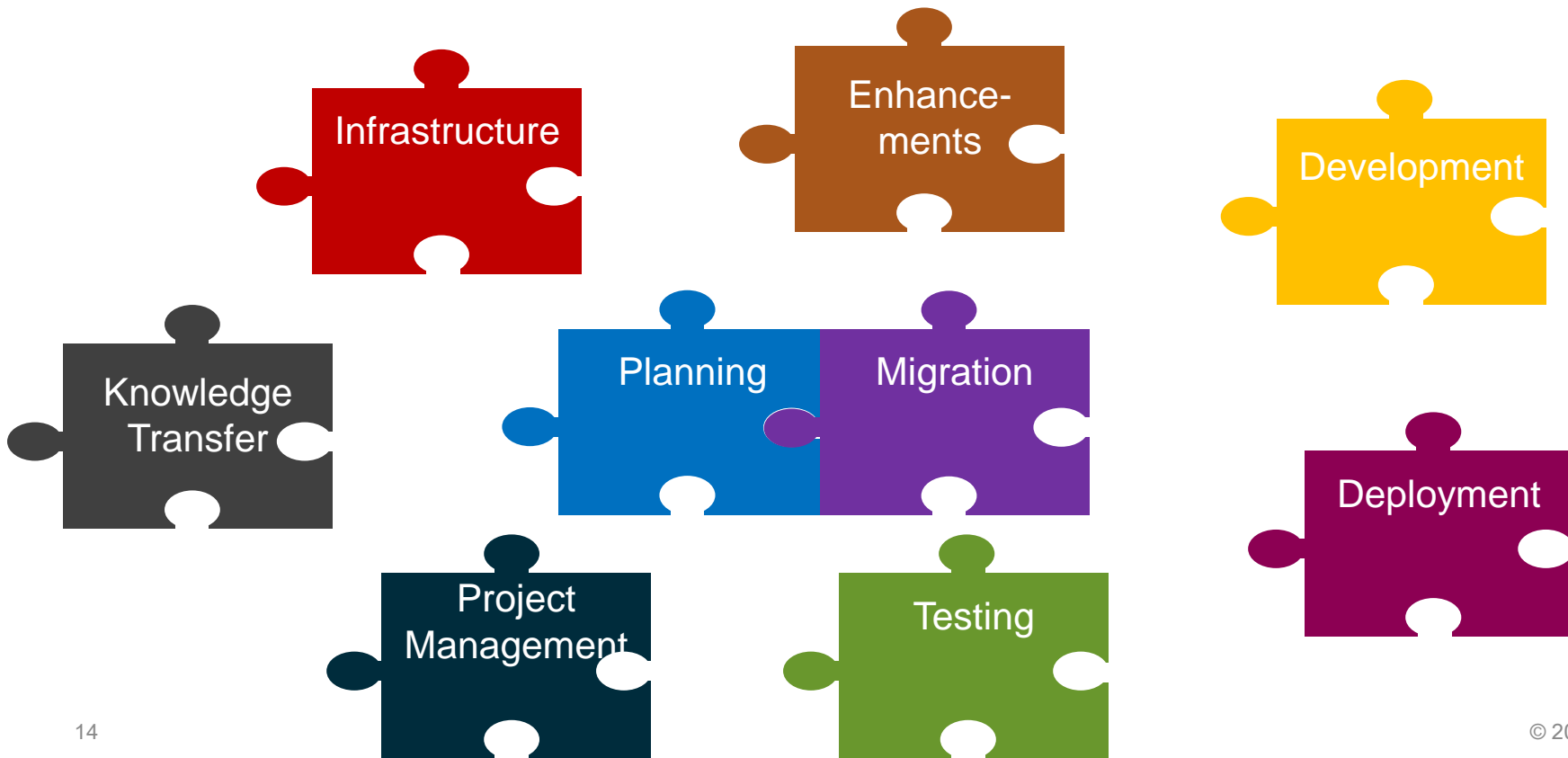
WebSphere Portal Migration options

- Consider using a manual migration if
 - There is a high degree of installation and deployment automation is implemented
 - For example using autodeploy
 - If you plan to re-create your site completely from scratch on the target version
 - If you already have a well-documented deployment process
 - use the core of that same process to create an equivalent version on the new version of Portal
 - Might be required to adapt the process for v8.5.

WebSphere Portal Migration options

- Consider using a manual migration if (cont'd)
 - If you are planning a two-step migration, from Version 6.1 to Version 8.0, and then Version 8 to Version 8.5
 - In most cases, the theme must be re-created and many references to deprecated portlets and pages must be removed → might be more efficient to start with a new 8.5 installation and manually bring over the required artifacts from your source environment, and then update them as they are brought over.

WebSphere Portal Migration activities



Before starting the migration



■ Plan

- Define complete migration plan for each environment
- Devise a rollback plan
- System Migration Path and Development Migration Path should be worked in parallel
- Migrate once, then stage other environments

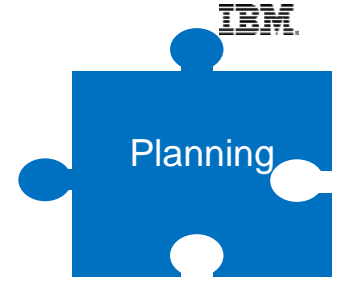
■ Prepare

- Scope required resources and time
- Obtain necessary software and hardware
- Determine which applications need to be updated and begin updating them in a sandbox environment

Before starting the migration - planning



- Define migration scope
 - Non functional requirements → Do they still fit current requirements?
 - When to enable Portal's new features? Which ones?
 - Which deprecated and unsupported features are currently used?
 - Which components require an update to fit Portal Hardware- and Software Requirements?
 - Operating System
 - Database
 - User Directory



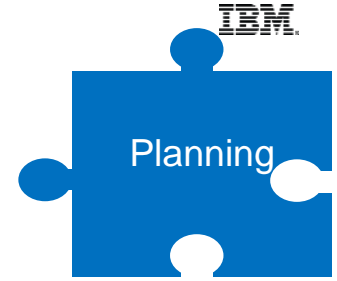
Before starting the migration - planning

- Define migration scope (cont'd)
 - Which other systems are connected with Portal / Portal services or are integrated?
 - External CMS
 - Business Process Management
 - Other business applications
 - External Search Engine(s)
 - External Security Components (proxies, reverse proxies, Tivoli Access Manager, CA Siteminder, etc.)
 - others ...

Before starting the migration - planning



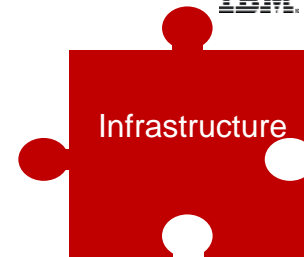
- How are changes brought to Portal stages?
 - Automated Deployment process / Tools
 - Release Builder
 - Xmlaccess, etc.
- Which Portal artifacts are currently in use?
 - Themes, Skins
 - Portlets (IBM Standard portlets, 3rd party portlets, custom developed portlets)
 - Other artifacts such as Servlet Filters, rendering Servlets, page resolution filters, etc.
 - Frameworks, that may require changes due to API changes
 - JSF



Before starting the migration - planning

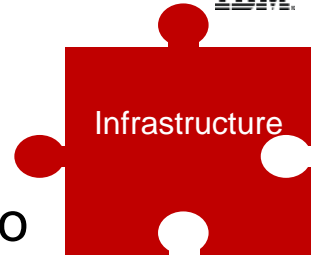
- What resources are required for the migration?
- In which order / sequence are the environments migrated
 - Bottom up or top down
- Tooling
 - Which currently used tools require an update?
 - New tools being implemented?

Preparing the migration



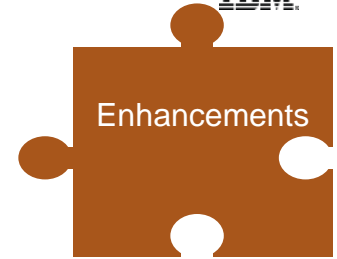
- The new Portal infrastructure must be available before migration can start
- Size the hardware based on the source infrastructure, its usage metrics and future workload – use techline if needed
- Review the system context diagram to ensure it is current
- Review the system in terms of operational aspects (Monitoring, backup/recovery etc.)
- Treat each tier in a multi-tiered environment separately
- Developers environments and tools may require an update either (e.g. RAD)
- Prepare other systems such as firewalls, proxy servers, SSO providers etc. for the target environment in each tier if applicable
 - Prepare test environments for functional and non-functional tests

Preparing the migration

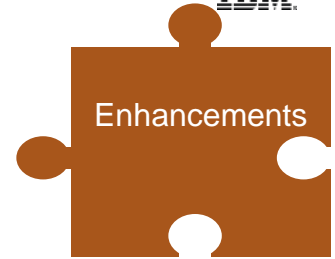


- Investigate the [Portal Hard- and Software requirements](#) to fit to a supported environment
 - Upgrade directly connected systems such as LDAP, database, etc. if required!
 - Ensure that browsers and the underlying operating system fit to Hard- and Software requirements of the target system
- Plan the transition phase
 - when the target environment is not productive yet but may already use other productive systems
- Develop a detailed “go-live” plan
 - Best case - just a DNS switch

Preparing the migration



- New Portal features enhance the overall user experience of Portal
 - Investigate the new features and define if and how they fit to the target Portal configuration
 - DDC – Digital data connector
 - Script portlet
 - Managed pages – Portal pages are treated as content items
 - WCM and editing enhancements – inline and inplace editing
 - Enhancements may require major changes to the current Portal implementation or tooling



Preparing the migration

- Keep migration plan simple and do not introduce unnecessary change and complexity
 - **BUT** make the necessary changes to the target environment
 - Consider the migration restriction section of unsupported or deprecated configurations
- Update custom Themes to modular Themes
- Use static pages if not already in place
 - Consider using the [Page Migration Tool](#)
- Implications of new features to existing deployment processes

Development considerations



- As Portal APIs may have changed each custom developed artifact may need to be reviewed
- Required changes of development frameworks used by the Portal applications may increase development efforts significantly during migration projects
 - JS Frameworks such as Dojo
 - Java Server Faces (JSF)
 - Customer's own development frameworks
 - others
- Investigate each custom developed Portal artifact in order to determine what has to be changed
- Limit changes during the migration phase to the required ones

Deployment considerations



- Most customers running Portal have a managed/ automated build and deployment process in place or consider the introduction of one in the target environment
 - Autodeploy
 - Urban Code Depoloy
- Build management tool such as Apache Maven help developers to identify dependencies that are required to be resolved in the target environment (if all components are “mavenized”)
- Portal managed pages and syndication may require revision of the current deployment process
- Cross-version syndication should be used to keep WCM content current during migration and go-live phase

Testing considerations



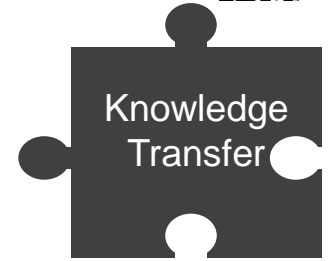
- Functional testing
 - Functional testing is essential for Portal migrations
 - Try to automate as much of testing as possible
 - Try to involve correspondent business units for functional tests as early as possible
 - Have valid test cases well documented and a feedback cycle to the developers in place
 - For larger environments when performing major changes tool based functional testing such as Selenium tests may be beneficial

Testing considerations



- Performance testing
 - Use Portal Performance Tuning Guide as a good start
 - The [portal server tuning tool](#) simplifies the initial tuning of the environment
 - Performance settings from the source environment may not be suitable for the target environment
 - Try to run performance and stress tests in order to validate the assumptions taken in the Portal Sizing
 - Have all infrastructure components involved but make sure you don't use systems unintentionally
 - Use realistic use cases for performance testing

Knowledge Transfer



- Make sure operations team is educated in the new environment
- New features in the target Portal environment may cause the requirement to train the involved people such as end users, developers, operations, etc.
- Consider an appropriate amount of time for trainings
- **Involve IBM services** if necessary

Vielen Dank



Hermann Huebler

<mailto:huebler@de.ibm.com>



Join the
conversation

[Blog](#), [YouTube](#),
[Twitter](#) and [Facebook](#)

IBM Commerce

