PORTAL MIGRATION

Bhudev Kumar, Senior Technology Architect, Infosys

Infosys®
Navigate your next
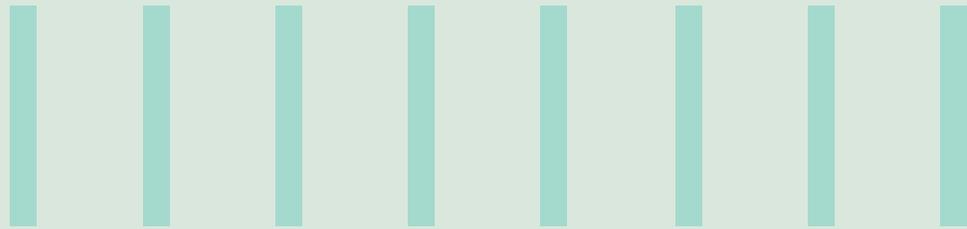
Executive summary

The evolution of enterprise portal products with changing business needs and trends, has threatened the survival of legacy portals and legacy applications. This is primarily because of their increasing maintenance costs, along with the growing demand for mobile, social, and cloud, over the last 5 years, which has transformed the modern portals market. Consequently, companies now have to reconsider not only their digital strategy, but also investing in legacy portal products altogether. The traditional portals were unable to deliver the expected results and despite this, in some scenarios, forced customers to increase their budgets. The new portals, on the other hand, employ digital technologies, such as mobile, social, and analytics, alongside modern Web 2.0 technologies, such as AJAX, widgets, representational state transfer (REST), and WOA / SOA approaches. Thus, companies are now considering investing in modern portal products that are more aligned with their business needs and better capable of handling changing business scenarios. In this paper, we have detailed out the migration of a legacy portal product to a modern, Web 2.0-compliant portal. We have discussed the migration steps (which can be applied to legacy portal migrations) with specific examples drawn from the migration of a legacy portal product (Plumtree) to a modern portal (Oracle WebCenter).

Migration considerations

Multiple business and technology factors drive the decision to migrate from one portal platform to another. Maintenance overheads of existing traditional systems and the evolution of next generation technologies, along with their impact on consumer services, presents a strong need for organizations to align their business policies with modern-day technologies.

Migration drivers

- Maintenance cost
- Scalability
- Agility
- Performance
- Minimum business impact and maximum reuse
- Improvements in features and capabilities
- Incorporating user engagement features, such as collaboration and self-service -
  - To facilitate responsive and interactive user experience
  - To enable an omnichannel experience
- Gain customer insights through web analytics

These drivers help in deciding the migration approach and planning. For example, if one of the major drivers were to avoid rewriting of existing code, the focus on the reuse of existing code would be greater in the new portal design. If scalability were a driving factor, all software and hardware infrastructure decisions would be impacted. Thus, migration drivers are important factors to be considered when designing the migration solution.

Best practices and guidelines

Understand the source and target systems

During migration projects, sometimes an SME with expertise on both portal products can help in the mapping of source and target portal capabilities. Understanding how the different components of the source system are interconnected in the architecture, can help in migration to the target portal architecture.

Understand the scope and scale of migration

Estimating the migration efforts has always been a challenge in migration projects. Most times, there are unforeseen surprises and additional scope that organizations may not have considered during the initial migration assessment. There are multiple reasons for this, including:
- Legacy systems that have gone through various changes through their lifecycle and have become complex to define and their scope harder to understand
- Unavailability of design and implementation documents
- Lack of expertise in the older products
- Diversified systems are a challenge; especially the portals that are based on a remote portlet architecture. Here, most of the business functionalities are built using portlet applications deployed on a remote application server, thus complicating the portal architecture

Minimum business impact

Typical objectives of migration include, reducing the maintenance cost of legacy systems, improving business processes, and investing in technologies that adapt to changing business needs. In most scenarios, the customer has already invested in maintaining the current systems for many years – they thus, would not want to rewrite the entire system to meet business objectives, if given a choice. Migration, in this scenario, should consider the technology landscape changes, while keeping the business functionalities unchanged. After migration, the same functionalities can be reused in the target environment, thereby meeting the main objective.

Reuse the existing and minimize the changes

Every portal product has its own proprietary components that need to be decommissioned while migrating to a new environment. However, most of the systems, including portals, also use generic specifications to develop these portal applications, such as JSR standards, web services, and other J2EE frameworks. While the principal challenge is to reduce dependency on existing portal technologies, there is also the challenge of reusing the open standards and frameworks.

Analyze the risk and define the mitigation plan

Migrating legacy portal applications to a new portal product involves risks that need to be identified and mitigated. Common risks include the unavailability of documentation, lack of expertise on older portal products, out-of-support software and hardware, as well as inconsistent design that has evolved over the years. If this were not enough, there is also the risk

of feature incompatibility between the old and new portal technologies. All these risks must be articulated and documented during the migration assessment phase.

Capability-mapping between source and target system

Features available in the current portal should be reviewed for deprecation or unavailability in the target. Having the same features in the target system can result in unpredictable behavior, post migration. There may also be cases where the new product has entirely new features that can be alternatives to existing features. All existing features have to be analyzed and correspondingly mapped to the new product in the migration assessment phase. This can help in defining a clear migration approach.

Defining the portal migration strategy

Enterprise portals use various components that can affect the migration approach in different ways. For example, the portal content can be migrated using a big bang approach or a batch approach, while the portlets can be migrated using a phase-wise approach, application-based approach, or iterative migration approach. The high-level migration strategy for each component needs to be defined for an effective portal migration.
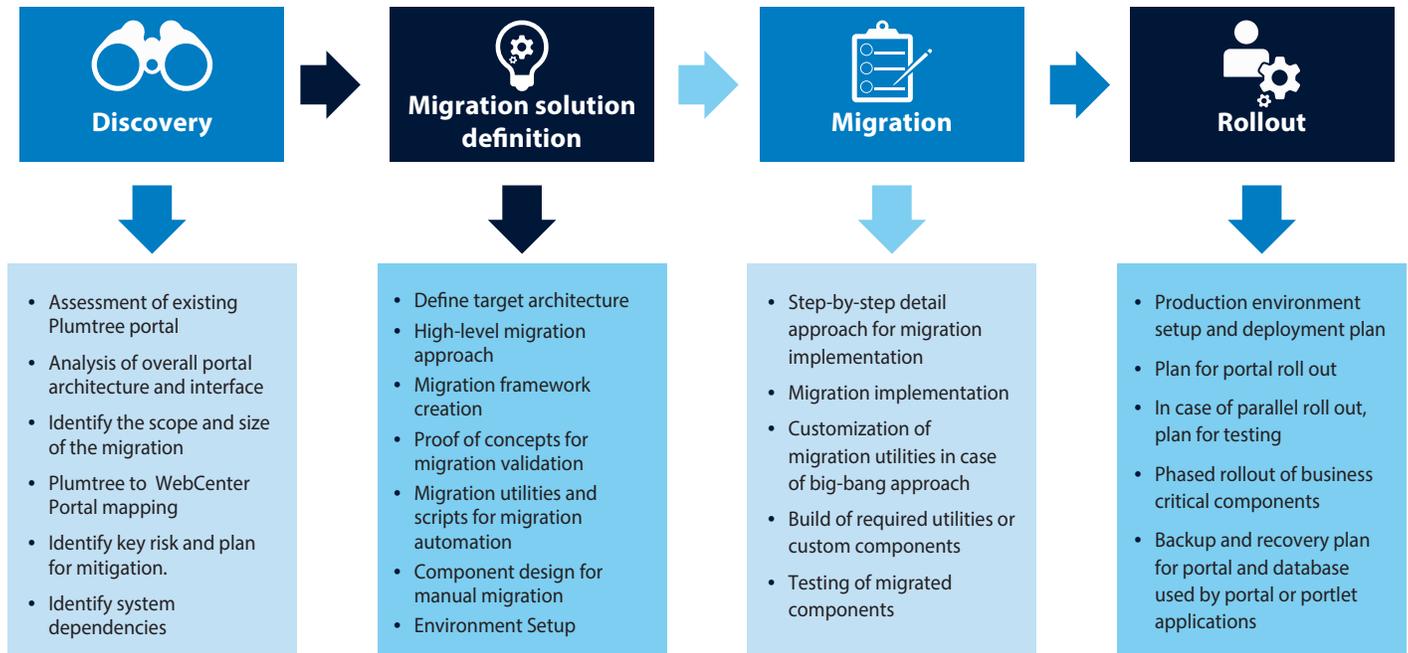
Migration methodology

In the process of portal migration, it is likely that some of the current portlets or portal features may not be available on the new platform. There could be portal components that can be migrated in an as-is state, while others may need to be built from scratch in the new environment.

Legacy portal applications specially built on a diversified architecture may contain diverse application as part of the portal architecture. Therefore, identifying an appropriate migration strategy from the available standard migration strategies and applying it to the overall migration

is an important step. Shown below is a standard migration methodology that can be used to migrate a legacy portal to a new portal application.

Discovery	Migration solution definition	Migration	Rollout
• Assessment of existing Plumtree portal • Analysis of overall portal architecture and interface • Identify the scope and size of the migration • Plumtree to WebCenter Portal mapping • Identify key risk and plan for mitigation. • Identify system dependencies	• Define target architecture • High-level migration approach • Migration framework creation • Proof of concepts for migration validation • Migration utilities and scripts for migration automation • Component design for manual migration • Environment Setup	• Step-by-step detail approach for migration implementation • Migration implementation • Customization of migration utilities in case of big-bang approach • Build of required utilities or custom components • Testing of migrated components	• Production environment setup and deployment plan • Plan for portal roll out • In case of parallel roll out, plan for testing • Phased rollout of business critical components • Backup and recovery plan for portal and database used by portal or portlet applications

Discovery

Enterprise portals integrate different applications across the organization's boundaries to provide a consolidated view of the various business processes. Thus, it is critical to chart a migration path based on business criticality, functional and technical dependencies, and user-base considerations.

In the discovery phase, the existing portal application should be understood from a technical and functional perspective and each of the critical dependencies that could potentially be impacted during migration should be documented appropriately. Thus, due diligence on all these services is required before migration

to the new portal application. The discovery phase encourages analysis of all services, assessments for as-is migration, and possibilities for customizations. This phase analyzes the business-criticality of different applications and prioritizes them in the migration process.

Migration solution definition

This phase starts once there is an understanding of existing system and its integration with various downstream systems for business process. The solution is defined as the as-is and to-be architecture, with the identification of the migration approach. The migration approach contains migration of portlets

applications, portal feature, portal static and dynamic content, user profile, and user preferences. In this phase, the design is also created for the impacted functionalities, business rules that may need customization or has to be developed again in the target portal. Following are some of the generic activities performed during solution definition phase:

- High-level migration approach
- Existing and to-be architecture and mapping of features
- Migration framework creation
- Proofs of concept for migration validation

- Migration utilities and scripts for migration automation
- Component design for manual migration

Migration

In this phase, the schedule for the migration activities is created based on the priorities. The detailed migration approach, along with the execution steps, is documented and subsequently executed by the development team. The framework for automated and manual migration is customized. The migration may involve various portal components, such as portal theme, skin, user profiles, content, portlets, workflow, and business rules of existing portal application.

Access to the existing development environment and target development environment should be available during this phase of migration. Portal migration is an incremental process that goes with migration of critical functionalities and regression testing to validate the successful migration and dependencies. However, if the portal migration is a version upgrade of the same old portal products, then the migration may follow a big bang approach, rather than a phase-wise migration of business critical applications. Portal migrations use some out-of-the-box tools, as well as some custom utilities or enablers, to simplify the migration process and reduce the migration time.

Rollout plan

An effective and properly planned production deployment is the key for a successful portal migration. The type of migration, scope of migration, and size of migrations are the factors that decide the rollout plan. Some of the rollout options include:

Big bang approach

A big bang approach is used when the entire system has been migrated in a single operation. This could take anything from a few hours, to a few days; but in general, a big bang approach is short in nature. The entire activity has to be planned such that users of the old system can get easily accustomed to the new system, to the point that sometimes, they can't even differentiate between the old and the new system.

Although cost effective, this approach involves significant risks and unpredictability. This is the main reason why a big bang approach always recommends running the old and new systems in parallel. If the target system presents too many issues, the user can always switch back to the old system. In case of portal migration, a version upgrade and migration to products of a similar architecture could be a smooth transition.

Phased approach

In a phased migration approach, the entire application is logically grouped into different phases and then sequentially migrated, phase-by-phase, into the target system. These logical groups can be based on functionality, language, geographical rollout location, and customer-specific content and sections of the site.

Unlike a big bang approach, the entire system is not rolled out at once, and there are chances to identify issues early in each phase. These identified issues can then guide the implementation of the rest of the phases. Phase-wise rollout helps users understand the system gradually and thus get adjusted to the changes. This approach also provides options to accommodate new changes and customer feedback into the system through testing at each phase. The main problem with this approach is that the client needs to maintain multiple environments until the roll out is completed and successfully executed. Simultaneously working on different

systems can sometimes be confusing for users. In this case, they can to work on one business functionality on the old system and another one on the new system. In most scenarios, especially in portal migration, the data source for the new and old system remain the same, except the fact that migration of user preferences and phasewise migrations cause data redundancy and data quality issues.

Parallel approach

In a phase-wise approach, users work with both systems during the migration phase and then gradually learn the new system. During this time, the old and new systems deal with production data, and production user-base and volume. Sometimes, these factors can impact the overall performance of the system. In high-availability systems, it is a good option to go with a parallel approach to avoid business impact and to gradually adopt the new system. While running a parallel approach, there is duplication of efforts as users have to enter data into both systems. This can sometimes also lead to data redundancy and data quality issues. However, if the two systems are not fully compatible with each other, parallel adoption is not a viable migration approach. Parallel adoption is most suitable for complex and high-availability systems. But sometimes, it can also be the most expensive and time-consuming approach.
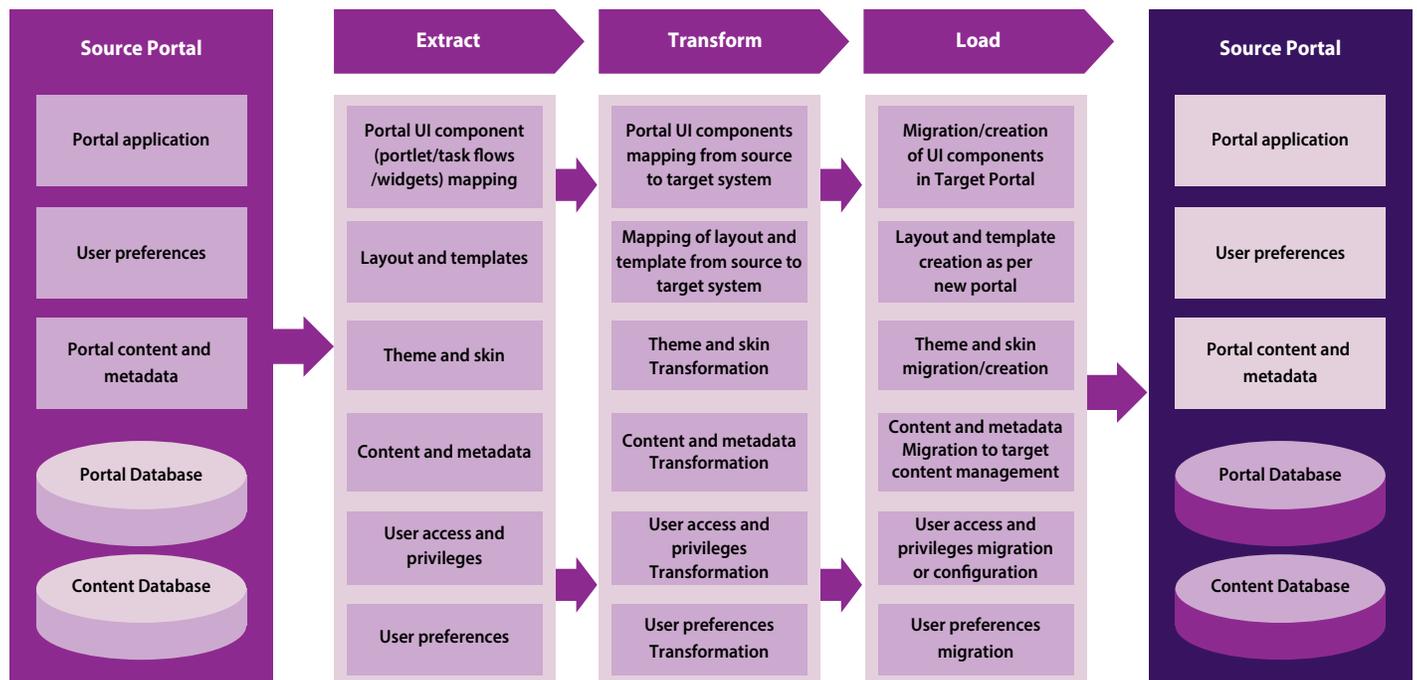
Pilot approach

A pilot approach ensures that the system is working as per expectations and ready for adoption by the business users. During the pilot migration, the test group users provide valuable feedback on the system to make for a smoother rollout to all users. Once the test group has approved the system, it can be rolled out across the organization. The test users can subsequently help train other employees for the new system.

Portal migration solution

There are several portal products in the market and each products has its own architecture and technology stack. Automation of migration becomes more challenging in this situation; especially in product migration, which is always a mix of automation and manual migration.

Portals supporting the portlet API based on JSR standards are easy to migrate as most portlet applications can be moved to the target system with no or minimal changes. However, at the same time, other components such as portal layout, theme, skin, and other portal configurations need

to be developed in the target system. Portal migration solutions provide a generic approach to migration that can be applied to any portal migration project.

Source Portal	Extract	Transform	Load	Source Portal
Portal application	Portal UI component (portlet/task flows /widgets) mapping	Portal UI components mapping from source to target system	Migration/creation of UI components in Target Portal	Portal application
User preferences	Layout and templates	Mapping of layout and template from source to target system	Layout and template creation as per new portal	User preferences
Portal content and metadata	Theme and skin	Theme and skin Transformation	Theme and skin migration/creation	Portal content and metadata
Portal Database	Content and metadata	Content and metadata Transformation	Content and metadata Migration to target content management	Portal Database
Content Database	User access and privileges	User access and privileges Transformation	User access and privileges migration or configuration	Content Database
	User preferences	User preferences Transformation	User preferences migration	

As per the above migration diagram, the entire migration process can be divided into three sub-processes – extract, transform, and load. Extract is a process to identify the migration components and extract them as-is from the source system. These components can be portal UI components, such as portlet, task flows,

and widgets. They can also be other portal components like the theme, skin, content, metadata, and user access and preferences. Transform is the intermediate process that analyzes the source and target system components, and the technology stack, and creates a migration map. This process also categorizes which components can
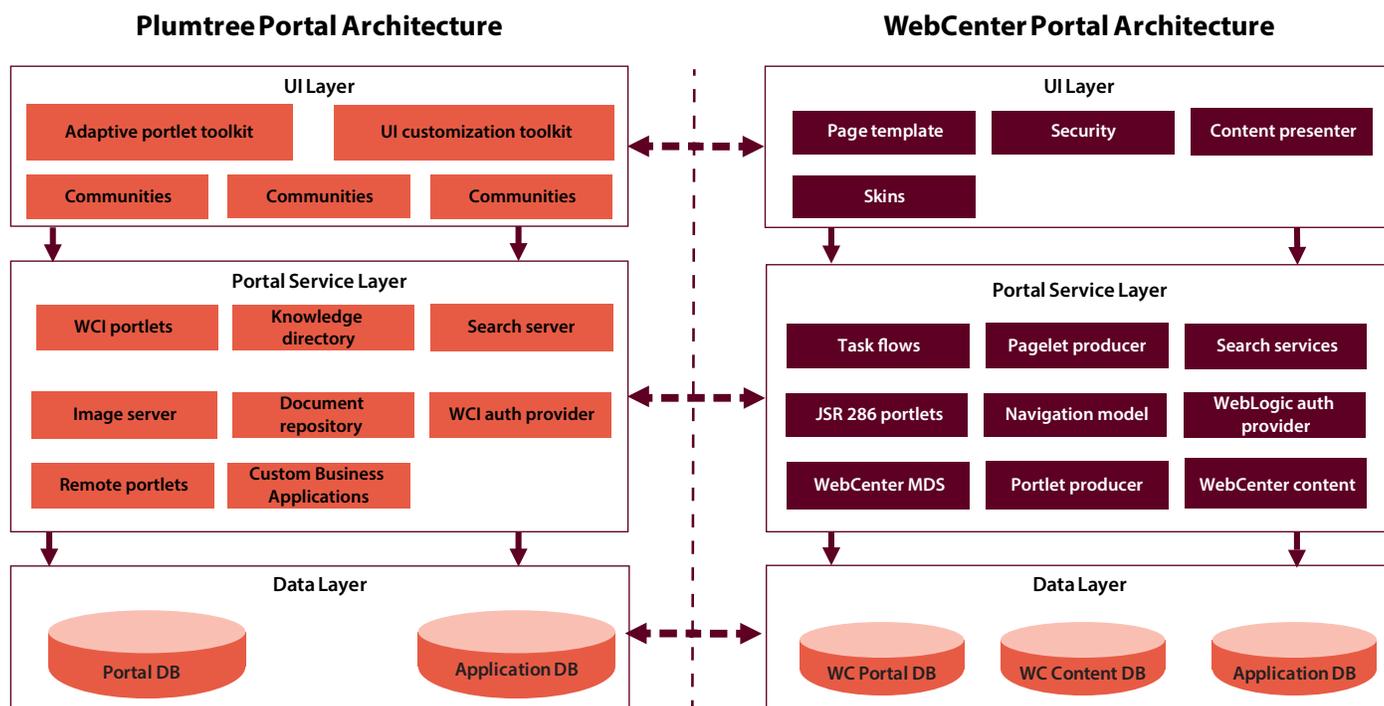
be automated and what needs to be developed again in target system. Load is the process that takes inputs from the transform process and implements the migration accordingly. It can be automated, manual, or a mix of both.

Plumtree Portal is based on "The Enterprise Web" architectural framework that follows the strategy of creating composite applications using Web services as the integration. The portal server is complemented by an additional set of application servers that provide

foundation services like search, automation, and content. Portlets are deployed on remote application servers as a way to integrate content and functionality from external systems, within the portal page.

WebCenter Portal is an Oracle portal product built on a JSF-based Application Development Framework (ADF). It provides design-time and runtime capabilities to build enterprise portals.

Plumtree Portal Architecture

UI Layer

Adaptive portlet toolkit		UI customization toolkit
Communities	Communities	Communities

Portal Service Layer

WCI portlets	Knowledge directory	Search server
Image server	Document repository	WCI auth provider
Remote portlets	Custom Business Applications	

Data Layer

Portal DB

Application DB

WebCenter Portal Architecture

UI Layer

Page template	Security	Content presenter
Skins		

Portal Service Layer

Task flows	Pagelet producer	Search services
JSR 286 portlets	Navigation model	WebLogic auth provider
WebCenter MDS	Portlet producer	WebCenter content

Data Layer

WC Portal DB

WC Content DB

Application DB

Plumtree portal architecture components

- Portal server handles incoming HTTP requests, builds HTML responses, and returns the appropriate portal pages to each client. The portal server creates communities for the customers and internal users, pages for those communities, and maintain a consistent UI across the entire portal.

- Remote servers (WebLogic application servers) host portlet applications and expose them as Web services written using the enterprise web development

kit. There are three managed servers that host the portlet applications that are integrated with portal server pages using WSRP1. These remote server-exposed web services WSDL as part of WSRP, integrate with the portal to display the portlet UI on portal pages.

- The Plumtree database (PT DB) is used to store metadata about portal objects and content in the portal Knowledge Directory. This database also stores user information that synchronizes with LDAP users and groups using jobs maintained and run by the automation server.

- Plumtree automation server manages portal job scheduling and implementation. The automation server also accesses remote crawlers, and profile and authentication services that store and retrieve information from within the Plumtree database. This server also maintains the content crawlers' jobs that bring the content metadata into the Plumtree database.

- Plumtree image server hosts images and other static content used by the portal and remote Web services. This is hosted on the Apache server and

maintains a static header and footer, specific to each customer. This also maintain css, js, html, and image files used by the portal and various portlet applications.

- Plumtree search server indexes and searches all the documents, information, applications, communities, discussions, websites, and other content accessible through the portal. This server runs search services used by different portlet applications for the content search. The content crawler retrieves metadata of the content from a remote content managed server, into the portal, and helps the portlet applications to search within the content.

WebCenter portal architecture components

- WebCenter Portal: WebCenter Portal is an Oracle portal product built on top of a JSF-based Application Development Framework (ADF). WebCenter Portal provides design-time and runtime capabilities to build enterprise portals.

- Discussion server (managed server in WebCenter Portal): Discussion server is a part of WebCenter Portal that runs collaboration services. This helps in integrating discussion forums and notifications with the main portal application.

- Utilities server: Utilities server provides various services such as analytics, activity graphs, and personalization to integrate with the main portal application.

- Portlet producer server (wc_portlet): Oracle WebCenter Portal provides several preconfigured portlet producers and hold various built in portlets. A custom portlet producer managed server can be used to deploy custom portlet applications.

- Web asset (static images/content): WebAsset project or a file structure can be used for any static content such as the customer's logo or portal images, used by the portal application. This project will be deployed on an Apache server.

- Portal database: WebCenter portal database maintains schemas used by WebCenter Portal. Each of the portal components, such as the managed server, have their own schema in the portal database.

Portal feature migration

Features	Plumtree Portal	Web Center Portal
Authentication	Plumtree Portal users are getting authenticated against LDAP.	There is a SSO implementation using siteminder, and users will be authenticated using siteminder integration with WebCenter Portal.
Authorization	A scheduled job is running on Plumtree Automation Server to bring LDAP users and groups in Plumtree database. Plumtree database is maintaining the resource access.	Retrieving user profile from the SSO and use this for fine-grain access control using portals user roles
Templates	Portal Templates, Community Templates	Portal Templates, space templates and skins UI Components
UI Components like forms, tree, tables, etc.	Plumtree components	Spaces Rich UI components(AJAX enabled) like grids, tree, data visualization components
Experience Definition	User Experience Definition	Page Hierarchy, Skins, Navigation Model, Personalization
Contents	Knowledge Directory : File server	Oracle WebCenter Contents for storing and managing documents and web content – company logo and creative assets.
Search Documents	Custom crawler to search file system and Plumtree search	WebCenter portal search or Oracle Enterprise Search to search from external content management system.
Communities	Communities: Join Community Unregister from Community Add Pages Remove Pages Calendar	Portal templates [Communities are not used as a forum in portal and mostly used to provide page level access to users. Communities will be used in the portal navigation with and providing access to respective community pages.
Portlets	JSR 168 portlets	JSR 268 Portlets
Non Portlet (External Applications)	Integrated using links/services	Integration using pagelet Producer
Jobs	Automation Server	Job Schedulers/Events

Portal template migration

Portal builder will be used to create a new portal template for the portal. Portal builder comprises the portal creating, editing, and administration areas of WebCenter Portal. There are various templates available in the portal builder, such as portal, discussion, site, or team site. Appropriate templates according to business needs can be used to create a new portal.

Portal assets

Portal assets are the objects that define the structure, look and feel, and the content of portals. There are various portal assets, such as page template, pages, resource catalogues, skin, and navigation model. A resource catalogue can be created to store all the portal-related assets and used throughout the portal for creating page templates, pages, navigations, and skins.

Portal content migration

Plumtree image server maintains all content, such as logos, images, JavaScript, and CSS. The same content can be hosted on an Apache server in the WebAsset Project of WebCenter Portal. WebCenter Portal comes with WebCenter Content that can also be used to maintain all content required by the portal.

Page template migration

Page templates are used to maintain a consistent look and feel across the portal. Source and target portals have page template features, but the underlying framework is different. There is a need to create new page templates in WebCenter Portal using content and text of Plumtree portal's page templates.

Navigation model

Both portals provide navigation capabilities. WebCenter Portal even supports various out-of-box navigation and custom navigation models. A new or similar navigation model can be created in WebCenter Portal using content and navigation design from the existing portal's navigation model.

Community migration

Communities are not used as collaboration features in Plumtree portal and are mostly used to provide page-level access to users. WebCenter Portal provides role-based authorization to users. Roles can be created by the portal admin to map with LDAP users or user group in WebCenter Portal.

Portal theme

Plumtree portal branding customizes the look of the portal. It manages the user experience and defines the look and feel of communities. WebCenter Portal uses portal themes and skins to perform portal branding. Both portals have different approaches to portal branding and the WebCenter Portal needs to create a new portal theme after understanding the existing portal's look and feel.

Portlet producer

In Plumtree's Web Services architecture, most portlets are hosted remotely. Each portlet is self-contained and executes its particular functionality in a separate process. These are deployed on a remote server and consumed by the portal server. Plumtree supports portlet 168.
In the WebCenter Portal, WSRP pools portlet Producer Bridge will be used to host the JSR286-based portlet applications. These portlet producers will be used to integrate the portlet within the WebCenter Portal.
As-is portlets can be migrated with changes to remove Plumtree-specific APIs. The portlets will be migrated as portlet producer applications in WebCenter Portal and then integrated with it.

Pagelet producer

All non-portlet applications deployed on remote servers will be integrated with the portal as services. In the Plumtree portal, the word service refers to the code that resides on a remote server, including portlets, crawlers, search services, authentication services, and profile services. The term "Web Service" refers to the object in the portal that represents the service code on the remote server machine.
In WebCenter Portal, non-portlet applications can be integrated using Pagelet Producer applications. Pagelets are sub-components of a web page accessed through the Pagelet Producer that can be injected into any proxy's application. Any application on a Pagelet Producer resource that returns mark-up can be registered as a Pagelet, which can then be displayed in Oracle WebCenter.

Portal search implementation

Plumtree portal provides the search services to search the portal content and content hosted on external content management systems. Oracle WebCenter provides out-of-the-box search to search portal objects. WebCenter portal search will search only objects and content from internal sources and WebCenter Content. If Plumtree is using crawling and search services to search the external content, these need to be migrated to WebCenter Portal. To do this, Oracle SES along with WebCenter Portal would be a good option.

Portal jobs migration

Plumtree portal schedules jobs to bring users from LDAP to Plumtree portal database and then creates groups to provide them the required access in the Plumtree portal. WebCenter Portal can be directly integrated with LDAP to bring users into WebCenter Portal in real-time. Most profile jobs do not require the migration and rather reduce the portal users' onboarding time onto the portal.

Summary

Portal migrations are dependent on the source and target portal products we have. Migration of the same product to its upgraded version could be an automated process, whereas the migration between two different products could be a mix of manual and automated processes. The compatible features, such as JSR standard-based portlets, content, user data, and preferences are mostly automated, whereas the portal configurations can be manually achieved in the migration solution. As Plumtree was acquired by BEA Systems and, subsequently, BEA by Oracle, most of the Oracle customers are moving to WebCenter Portal. Pre-migration plan assessment and understanding the migration drivers, especially the risk, maintenance cost saving, and cost of migration, are critical to achieve the actual value and business benefits from migration projects. Once an organization understands the drivers behind the migra¬tion, as well as the advantages and disadvantages in each potential migration scenario, it becomes easy to plan the migration and create a migration approach. If lacking in this understanding, organizations may be unprepared for decisions and trade-offs that must be made throughout the planning process. This document helps create a generalized process for portal migration projects and also focuses on the technical approach of Plumtree to WebCenter Portal migration.

About the author

Bhudev Kumar

Senior Technology Architect, Infosys

Bhudev Kumar is a Senior Technology Architect with Infosys Digital Marketing Group. He has 15 years of industry experience and specializes in software design and development. He has experience in portal, content management, social networking, Web 2.0, and J2EE architecture-based applications. His strength lies in his ability to provide technical solutions, integration analysis, and designing and implementation of software.

For more information, contact askus@infosys.com

Infosys
Navigate your next

Infosys.com | NYSE: INFY

Stay Connected   SlideShare