An Oracle White Paper
October 2013

Oracle RightNow Cloud Service Change Management Guide

Introduction

The most successful organizations understand the value of change management and take steps to enable it effectively. The most effective change management is achieved when an organization understands and adheres to change management practices for the products and services they depend on. Understanding how to stage and implement changes in the Oracle RightNow Cloud Service solution is valuable and necessary to ensure seamless integration with your existing change management strategy. The intent of this document is to take you through the areas of the Oracle RightNow Cloud Service solution that need to be considered when looking through the lens of change management. This document also provides recommendations for each area to address envisioning, developing, testing, and implementing change to your Oracle RightNow Cloud Service site(s).

The Value of a Change Management Strategy

The reasons to establish a change management strategy are vast. When it comes to the Oracle RightNow Cloud Service and change management, those reasons include:

- Ensuring consistent and predictable outcomes for changes introduced to the application environment.
- Bringing a vehicle of communication to impacted users and management approvals before changes are introduced.
- Establishing discipline within development, test and quality assurance teams.
- Forming a well-defined list of modifiable components to provide clarity and reduce confusion.
- Improving deployment and rollbacks (organized change is much easier to implement and, when needed, rollbacks are easier which reduces the impact to users and to your bottom line).
- Identifying cause and effect should troubleshooting be required.

Modifiable Components

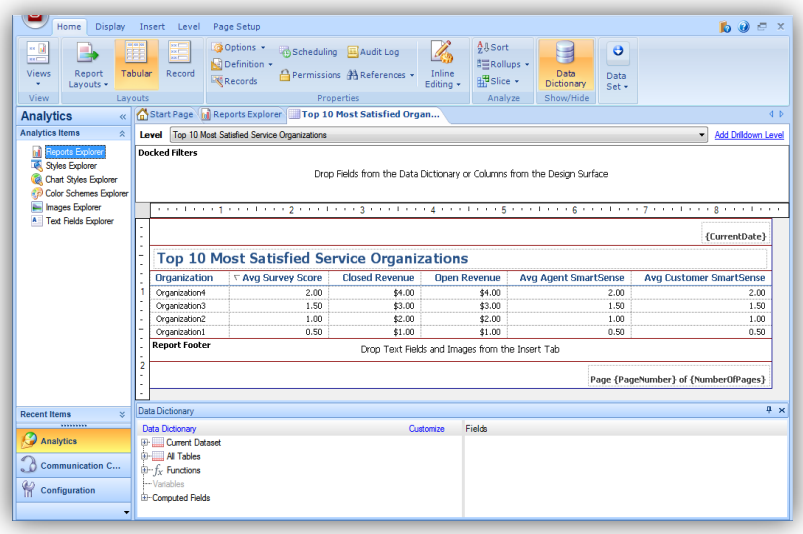
Oracle RightNow Cloud Service is feature-rich and provides you with the capability to customize your site in many ways. For a comprehensive staging and implementation strategy, you need to consider all customizable components for the most optimal change management experience. The customizable components addressed in this document include:

- Reports
- Customer Portal
- Workspaces and Workflows
- Business Rules
- Add-Ins
- Server-side Custom Integrations
- Integration with Public APIs
- Various Settings and Configurations

Every Oracle Service customer has the ability to request a *Test site*. The recommendations provided in this white paper are based on you utilizing both a *Test site* and a *Production site*.

Reports

Reports are a valuable tool for communicating information from your Oracle RightNow Cloud Service site. In addition to the standard reports, you're able to create your own custom reports.



Editing a report in the Report Editor

After creating or modifying reports in your *Test site*, it is possible to move the reports to your *Production site* with little effort. The first step is to identify all the reports you plan to move (newly created or modified). For each of those reports, you'll use the agent desktop to export the report definitions, which are stored in an XML file, to a location on your workstation or your local network. Then, from the agent desktop of your *Production site*, navigate to the location where you stored the exported report definitions (XML file) and import each report definition, making sure to save the report to a desired folder under the Public Reports directory.

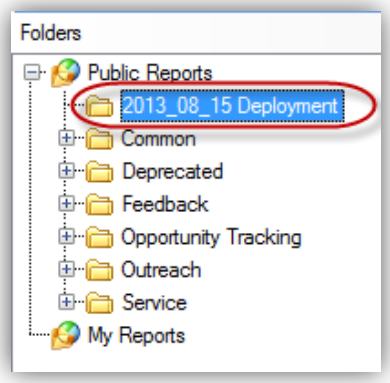
You need to check the permissions on each report after importing it and saving it in your *Production site*. If profile names change (profiles are discussed later), then permissions should be confirmed and reset as necessary.

Recommendations

Consider the following when developing your change management strategy regarding reports.

Keep Track of New or Modified Reports

Manually keep track of the reports that you created or modified in your Test site. This becomes your to-do list for what to export. In fact, a good way to organize this is to create a folder and store your new and modified reports in it.



An example of how to organize reports

Export, Import, and Verify Permissions

As a best practice, export each report definition from your *Test site* to local or network storage. Then, import each report into your *Production site*, and remember to save each report that you imported. Then, verify the permissions that are set for each report.

Additional Information

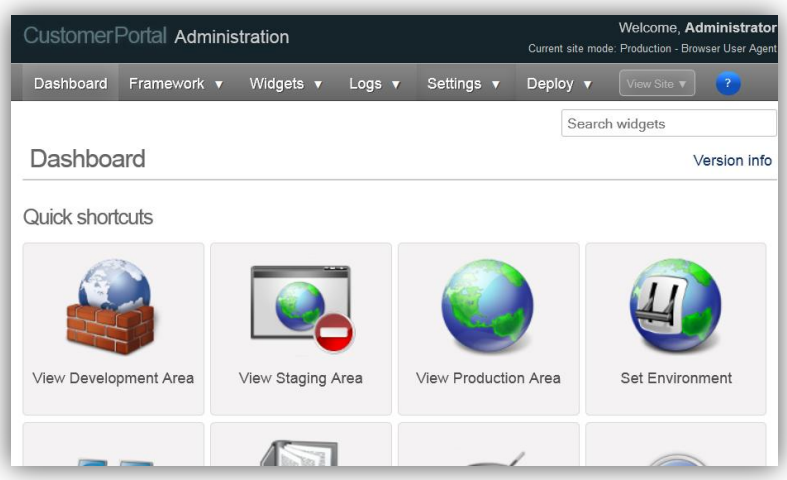
Here are useful links related to reports.

HELPFUL LINKS FOR REPORTS

TOPIC	HYPERLINK
Exporting report definitions	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_analytics_report_management.40.6.html#1130287
Importing reports	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_analytics_custom_reports.39.4.html

Customer Portal

As a customer, you can make changes to Customer Portal (CP). Therefore, it's important to understand how to manage these changes.

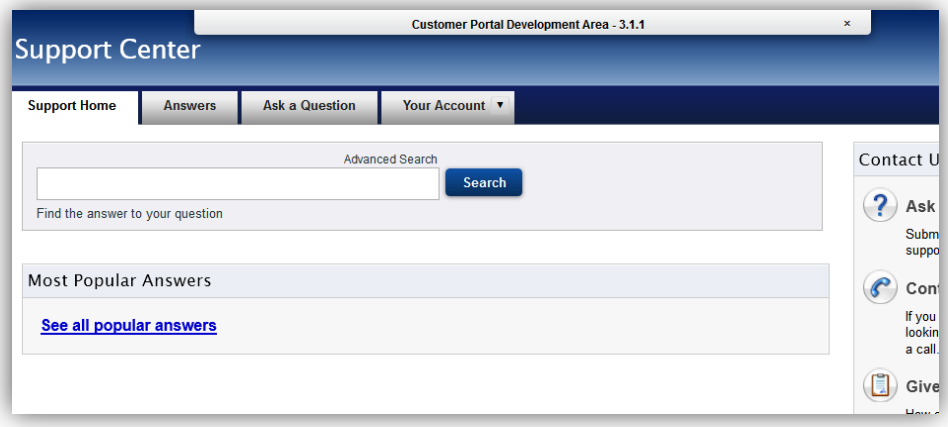


Customer Portal Administration View

Developers can actually perform development work without using a *Test site* because the Customer Portal on your *Production site* has three areas of its own for every interface: CP Development, CP Staging, and CP Production. On an interface-by-interface basis, your administrator grants CP Developer permissions to one or more developer accounts, and the CP developer can then set a cookie so that other visitors continue to view the Production pages while the developer sees the CP Development pages.

Initially, the CP Development and CP Production areas of a site look the same. Your developer can use WebDAV to manage the files in CP Development. Via normal code editors, the developer can edit HTML, CSS, JavaScript and PHP files to meet all of your customization needs. When the

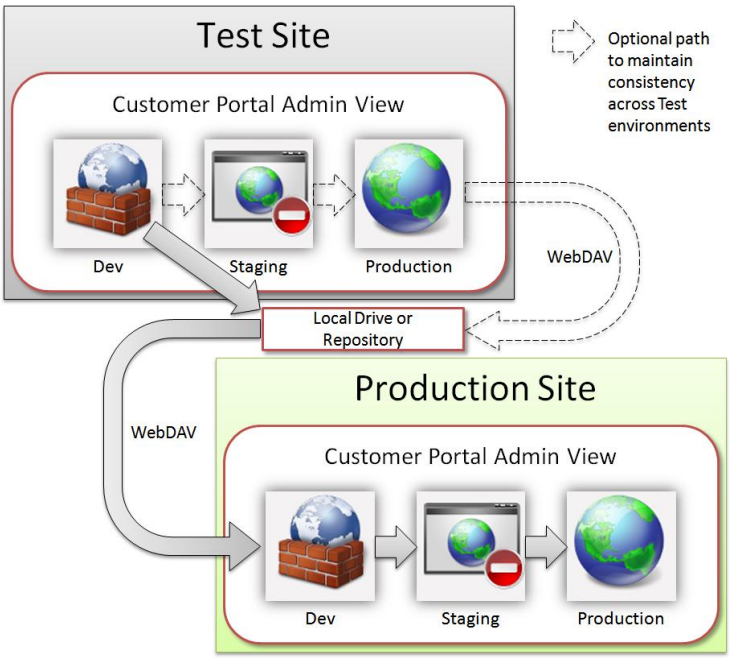
developer is ready, they use the admin tools to stage their changes in the Staging area. The Staging area files are fully optimized and the experience is identical to what would happen in Production. When ready, the developer or an admin user promotes the pages to Production. Moving files from Development to Staging (called "stage") and from Staging to Production (called "promote") is all done via either the CP Admin web site or the agent desktop, and these operations are carried out at the click of a button.



The Customer Portal Development area is initially identical to Production.

It's important to understand that your *Test site* also has the three CP areas of its own, which are separate from your *Production site*. And although you'll likely move files through the three CP areas of your *Test site*, you also need to move files from your *Test site* to your *Production site*.

On your *Test site*, you can use the CP Admin or the agent desktop to move files through your *Test site* CP areas. You'll use WebDAV to move files from your *Test site* to local storage or to a local code repository. From there, you'll want to verify that everything is working correctly in the CP Production Area of the *Test site*, and if it is, then move the files to your *Production site's* CP Dev area. Then, you'll use the CP Admin or the agent desktop to move those files through the CP areas of your *Production site*. See the illustration below.



Recommended flow of CP files from a Test site to a Production site

Note that it's not required that you stage and promote in your *Test site*, but if you do, you will maintain consistency across the CP areas of your *Test site*.

Recommendations

Consider the following when developing your change management strategy regarding Customer Portal.

Develop CP Changes in the Test Site, Then Transfer Them to the Production Site

To follow best practices, do development of your Customer Portal in your *Test site* in the CP Dev area, and push your changes through the Stage and Promote process. Then, export those files to a repository for safe-keeping before transferring them to your *Production site's* CP Dev area. Once satisfied with your changes, follow the Stage and Promote process. This protects your live site from any database changes, test contacts and incidents, etc., that might otherwise accumulate when testing functionality in Customer Portal.

Utilize the Development and Staging Areas of the Interfaces

In addition to using the *Test site*, make sure developers use the Development and Staging areas before promoting code to Production. Grant only developers the CP Developer permission to control the personnel who can edit code on your site.

Manage Your Code

Manage your custom code outside of the Oracle RightNow Cloud Service and use a version control system or code management system. Before promoting code from Staging to Production, ensure backups are available in your code management system and consider tagging your local code.

Additional Information

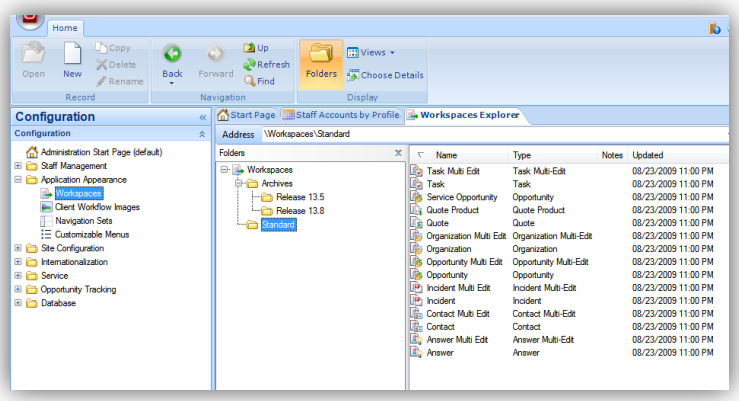
Here are useful links related to Customer Portal.

HELPFUL LINKS FOR CUSTOMER PORTAL

TOPIC	HYPERLINK
Customer Portal Deployment Manager	https://cx.rightnow.com/app/answers/detail/a_id/4236/
What is Customer Portal and how does it work?	https://cx.rightnow.com/app/answers/detail/a_id/4277/
Staging and Promoting the Customer Portal	https://documentation.custhelp.com/euf/assets/docs/february2013/olh/wwhelp/wwhimpl/common/html/wwhelp.htm?context=crm&file=staging_promoting.70.1.html
Tools for Working with the Customer Portal Framework	https://documentation.custhelp.com/euf/assets/docs/february2013/olh/wwhelp/wwhimpl/common/html/wwhelp.htm?context=crm&file=DevGuideIntro.71.4.html
Version Control of CP	http://communities.rightnow.com/posts/870a6caa0b

Workspaces and Workflows

Workspaces define the configuration of the content pane when staff members view, add, and edit records. Oracle RightNow Cloud Service contains default workspaces that address typical business needs, or you can create custom workspaces.



An example of how to organize Workspaces

Within the Workspaces Explorer, workspace administrators can create folders and organize archive copies of workspaces with flexibility. Standard workspaces are provided by default and are suitable for most common record editing. In addition to standard workspaces, Oracle RightNow Cloud Service gives you complete flexibility to create your own custom workspaces. How you organize workspaces is entirely up to your workspace administrators.

When it comes time to move workspaces or workflows from your *Test site* to your *Production site*, log on to the agent desktop for your *Test site*, export your workspace and workflow definitions to XML files on your workstation (one workspace or workflow per XML file), log on to the agent desktop for your *Production site*, and import the workspaces and workflows, being sure to save them after the import.

Recommendations

Consider the following when developing your change management strategy regarding workspaces and workflows.

Maintain a Backups Folder or an Archives Folder for Archive Copies of Workspaces in Production

As a safety net, create a Backups or Archives folder and store copies of your current Production workspaces there. This makes it easy to quickly revert back to them, if necessary. Also, remove very old workspaces to avoid confusion and to have a well-groomed environment. A good rule of thumb is to keep only the most recent one or two versions of workspaces; anything older is probably not useful and could be confusing to some workspace administrators.

Alternatively, since workspaces can be exported to XML files and stored, you can easily store them in an external code management system.

Export Workspaces From your Test Site, Import Them into your Production Site

Using the Agent Desktop, export each workspace from your Test site to a local XML file. Then, import each XML file into your Production site and then remember to save each workspace.

Additional Information

Here are useful links related to workspaces and workflows.

HELPFUL LINKS FOR WORKSPACES AND WORKFLOWS

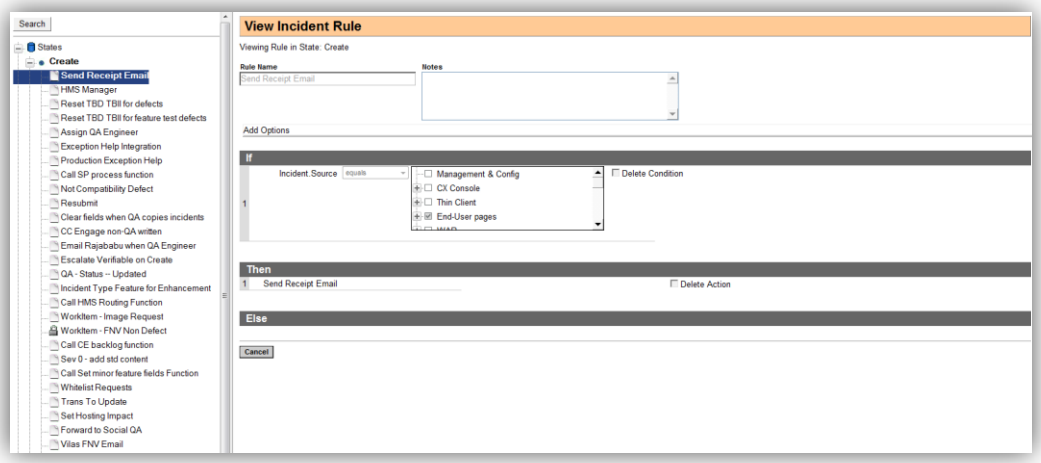
TOPIC	HYPERLINK
Best Practices for Workspace Design in Oracle RightNow Dynamic Agent Desktop Cloud Service	http://www.oracle.com/us/products/applications/best-practices-for-workspace-design-1639582.pdf
Oracle RightNow Cloud Service User Guide	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/wwhelp/wwhimpl/js/html/wwhelp.htm
Exporting and Importing Workspaces	http://documentation.custhelp.com/euf/assets/docs/august2012/olh/dad_workspaces.28.18.html#1394366

HELPFUL LINKS FOR WORKSPACES AND WORKFLOWS

TOPIC	HYPERLINK
Exporting and Importing Workflows	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/dad_workflows.34.32.html

Business Rules

Business Rules are available for customizing a site to fit an organization's requirements. Business rules are powerful tools for simplifying and automating common business tasks. In their simplest form, business rules are if-then statements that you create to fit your specific business needs.



Example of the Business Rules area

Business Rules can be created for incidents, answers, contacts, organizations, tasks, or opportunities, and they are one of the most advanced components to work with when it comes to change management. Business rules have two modes: Active and Edit. When a rule is being edited, it's not active, and rules need to be made active when you want to implement them. If you delete a business rule, it must be re-created.

Rules that are created in your *Test site* must be recreated in your *Production site*. To help facilitate the proper construction of rules, the Rule Log can be used to evaluate how a rule behaved after firing. This lets you see what rules fired and whether the Then or Else clause was used for processing.

Recommendations

Consider the following when developing your change management strategy regarding business rules.

Order Matters

Business Rules are executed from top to bottom, so to make sure you get the same results in Production as you get on your *Test site*, make sure the rules you create on your *Production site* are in the same order.

Delegate rule management to a small body

Because business rules are critical to incident routing and management, you should identify a very small group of people who are allowed to edit and activate your business rules. This way, changes to rules can be tightly controlled.

Manually Recreate Business Rules From Your Test Site in Your Production Site

There are no tools to allow the exporting or importing of business rules from one site to another. Therefore, a good way to do this is to log on to both your *Test site* and your *Production site* and arrange them side by side on your screen in order to duplicate in Production the rules from Test. Compile and double-check syntax of your rules before activating them.

Verify Rules Using the Rule Log

Make use of the Rule Log as a way to verify rule behavior.

Carefully and Deliberately Activate Rule Changes

Use care when activating rules. Activate a rule only when you are confident that it is written correctly. Although there is a feature that allows you to "undo" the latest change to a rule, if the previously activated rule also had a problem, your rollback will not be effective and you'll need to correct the problem as quickly as possible.

Additional Information

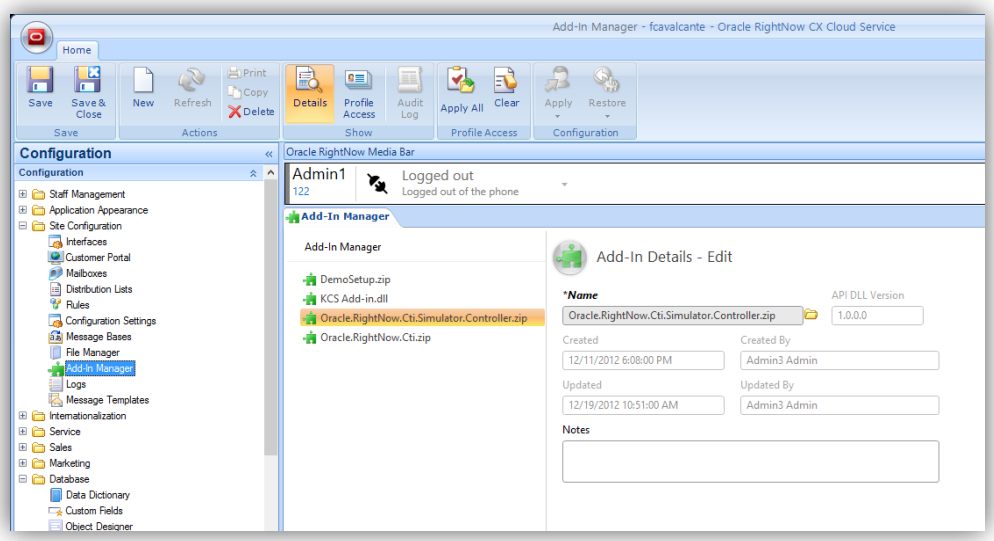
Here are useful links related to business rules.

HELPFUL LINKS FOR BUSINESS RULES

TOPIC	HYPERLINK
Business Rules Management	http://documentation.custhelp.com/euf/assets/docs/august2013/olh/crm_admin_business_rules_management.11.1.html#1134421

The Connect Desktop Add-In Framework

The Connect Desktop Add-In Framework provides a robust add-in framework for the agent desktop allowing integration developers to build custom .NET components, controls, and applications that live in the agent desktop and automate, extend, and integrate the agent desktop with other systems to create a unified business process.



Add-In Manager

When developing add-ins, developers can make use of a profile configuration known as "Developer Mode" which when enabled, allows a developer to place an add-in assembly and related files in a pre-determined directory, where the agent desktop will load the files. This allows developers to test add-ins without having to upload the add-ins to the server.

When staging a new add in, you'll need to deploy the add-in(s) to your *Production site* and limit the add-in(s) to the profile(s) that represent your testing team (no Staging area is required). Once your tester logs on, they can test the add-in(s). Once tested, deploying the add-in(s) is just a matter of enabling it in additional profiles.

Because add-ins are authored in C# and compiled on a developer's machine before they can be deployed to your site, consider using a source code management strategy and naming convention that:

1. Ensures that you have backups in case of a local system failure.
2. Relates specific source code versions to particular builds of your add-ins.

Recommendations

Consider the following when developing your change management strategy regarding add-ins.

Enable Developer Mode In Your Test Site

Developer Mode is a simple profile configuration option that will greatly reduce the time required for your developers to test their work.

Maintain Backups of All Add-Ins

The process of uploading an add-in will replace an existing add-in named the same, so reverting back to the prior version is a simple matter of uploading the prior version's file. Therefore, it is important to keep copies of all add-ins in a safe location.

Manage your add-ins outside of the agent desktop and use a version control system. Before uploading an add-in package that replaces an existing one, ensure backups are available in your code management system.

Deploy to Production the Add-Ins Deployed in Test

Once you are ready to deploy add-ins to Production, take inventory of the add-ins in your *Test site* and re-deploy them to your *Production site*.

Version Every Add-In Update and Identify that Version in SCC

To make sure add-in updates are easy to identify, we recommend you update the version on your add-in assemblies any time you make changes and upload your add-in. This allows you to match the version running in your site against your codebase.

Use the Add-In Update Option Instead of Creating New Add-Ins

When publishing a NEW version of the SAME add-in, update the existing add-in entry instead of creating a new one. This ensures that your updated add-in retains the same record ID so that references to it are not impacted (e.g. Customized Workspaces).

Additional Information

Here are useful links related to add-ins.

HELPFUL LINKS FOR ADD-INS

TOPIC	HYPERLINK
Oracle RightNow Desktop Add-Ins Cloud Service	http://documentation.custhelp.com/euf/assets/devdocs/february2013/Connect_AddIn_Framework/Default.htm

Server-Side Custom Integrations

Server-side custom integrations include custom scripts, external events, custom process models, and social community. Often times, organizations will enlist the help of Oracle Consulting Services (OCS) to develop, and implement custom integrations, and some custom integrations can also be managed by organizations themselves.

Server-side custom integrations refer to:

- Custom scripts (located in interface.cfg/scripts/custom/)
- External Events (EEs)

- Custom Process Models (CPMs)
- Desktop Integration Add-Ins
- Social Community

If you enlist OCS for custom integrations, OCS will manage the staging and implementation of those components of your sites that they are responsible for, and they will coordinate with you when changes are implemented.

External Events and Social Community are not customer-accessible, so if you require customizations to these components, an engagement with OCS is required. Therefore, OCS will manage the promotion of these items from Development to Staging and from Staging to Production, and OCS will coordinate with you so that these changes are implemented according to your schedule.

If you lack the expertise in-house or if you deem your customizations critical enough, you might engage OCS to manage your custom integrations.

If you manage custom integration components without the help of OCS, a best practice is to implement a process like the following:

- Manage your custom integrations using version control software.
- Manually move your custom objects from your Test site to your Production site.
- Organize and track all of the components that constitute your custom integration, which can include:
    - Logging and notification
    - Configuration Settings
    - Workspaces
    - Reports
    - Agent Desktop Add-Ins
    - Customer Portal
    - Custom Scripts

Managing these components together can be a challenge, so try to channel changes through a change coordinator who serves as a gateway through which all changes must pass before being implemented. Also, a back-out plan for each component is highly recommended so that changes can be rolled back quickly and accurately when needed.

Custom integrations often require database schema changes which are done through Custom Business Objects. Deleting a field from the database will irrevocably delete all of the data in the field and potentially cascade to related data elements. It is highly advisable to have backups of "hard to reproduce data" if your project includes schema changes. Practicing the deployment of data level changes on a *Test site* is also advisable for a couple of reasons: 1) estimating and managing your deployment window is important because database operations on large implementations can take from seconds to several hours to complete depending on the objects affected and, 2) rehearsing the database changes on a fresh copy of you production site reduces the chances of unintended collisions between concurrent projects and it also validates your deployment and rollback plans.

Recommendations

Consider the following when developing your change management strategy regarding server-side custom integrations.

Keep Backups of "Hard to Reproduce Data"

If your custom integration implementation includes database schema changes, keep a backup of your database so that a rollback is possible if needed.

Engage with OCS or Manually Coordinate Your Changes

Custom integrations can be complex and sometimes require an engagement with Oracle Consulting Services. If your components do not require OCS, or you prefer to manage your custom integrations in-house, identify a change coordinator who is a central contact point for all changes, and develop roll-back plans as a safety measure.

Additional Information

Here are useful links related to server-side custom integrations.

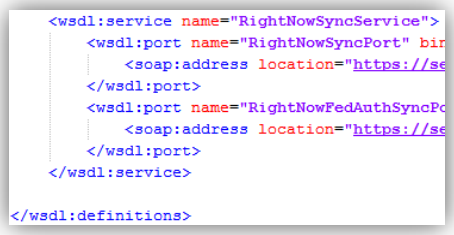
HELPFUL LINKS FOR SERVER-SIDE CUSTOM INTEGRATIONS

TOPIC	HYPERLINK
Professional services and add-on purchases available	https://cx.rightnow.com/app/answers/detail/a_id/1731/kw/test%20site

Integration with Public APIs / Connect Web Services

Oracle RightNow Cloud Service makes public several APIs to allow customers to integrate their applications with their Oracle RightNow Cloud Service-based data. This family of APIs is often referred to as the RightNow Connect APIs. APIs that are part of this family include:

- Connect for PHP
- Connect Web Services for SOAP
- Connect Knowledge Foundation API
- Chat Consumer Interface API
- Chat Third Party Queue Integration APIs



It is important to note that when Oracle releases new API versions, your consumption of these is not automatic. Your use of the existing APIs will continue, as long as you are consuming a supported version. Currently, APIs are fully supported from the date they are released to a minimum of two years after deprecation. For you to consume new API versions requires you to deliberately change the reference to the new APIs in your WSDL configuration files (for SOAP-based APIs) or the new

Connect for PHP version. Since you have a *Test site*, consume new API versions there first. When you are ready, make changes in your *Production site* to use the new APIs.

Due to the broad array of available APIs and functionality exposed by those APIs, it is important to know which APIs and, specifically, which functionality you depend on. To document this, create use cases and define requirements that clearly illustrate why you use the APIs and what you accomplish with them. Documenting your needs in this way helps you in a couple of ways. First, you will understand better how to implement the functionality should a particular API operation become deprecated and you need to maintain the same behavior. In other words, your use cases describe what you accomplished, so adapting new API functionality becomes easier. Second, it means you'll easily recognize the possibility of enhancements in your application when new features are released.

Recommendations

Consider the following when developing your change management strategy regarding public APIs.

Document Your Use of Public APIs

Document how and why your applications make use of Oracle RightNow Cloud Service APIs and document which functionality of the APIs is being consumed. Create use cases and requirements that clearly identify why you use this functionality and how it was implemented. When new API versions are released, it will be much easier to know which parts of your applications need to be updated and why.

Use the Latest APIs in Test When They Become Available

When new Oracle RightNow Cloud Service APIs are made available to you, make an effort to begin consuming them in your *Test site* so that you can identify and address any issues before consuming them in Production. Staying current with the latest APIs means you'll always be taking advantage of the latest capabilities exposed by the APIs, and you'll be aware of new functionality available to you.

Additional Information

Here are useful links related to integration with Public APIs.

HELPFUL LINKS FOR INTEGRATION WITH PUBLIC APIS

TOPIC	HYPERLINK
Connect Web Services for SOAP	http://documentation.custhelp.com/euf/assets/devdocs/may2013/Connect_Web_Services_for_SOAP/Default.htm
Oracle RightNow Connect PHP API Cloud Service	http://documentation.custhelp.com/euf/assets/devdocs/may2013/Connect_PHP/Default.htm

Additional Considerations

In addition to the above areas where change must be addressed, the following are additional areas within the Oracle RightNow Cloud Service that may need your attention.

Custom Business Objects

Custom Business Objects, or CBOs, give you the ability to extend the database schema. The Object Designer tool is used to create a package which contains the objects, fields, and relationships that you create. Once you're done creating your package, you'll export it from your *Test site,* which creates a zip file. This zip file can be imported into your *Production site*. Once the import has been completed, be sure to deploy the newly created custom business objects in Production.

Use your *Test site* and test database as an area to rehearse your custom object implementation. Write a detailed, well-scripted implementation plan and practice it against a fresh copy of your *Production site* and your production database.

A best practice when it comes to custom business objects is to maintain backups of all components that you develop. Version control software is highly recommended for custom business objects. This allows you to manage multiple versions of your components and also serves to simplify the deployment process.

Guided Assistance Explorer

Similar to workspaces and workflows, Guided Assistance items can be exported from your *Test site* to XML files and then imported into your *Production site*.

Permissions

Anything that can be exported from Test and imported into Production can have its permissions altered inadvertently, especially if your Profile names are not the same between Test and Production. Therefore, after any import, such as reports, workspaces, workflows, or guided assistance objects, best practice suggests you verify the permissions on those objects and adjust them, if necessary.

Products, Categories, Dispositions, and Custom Fields

Any linking that needs to be configured between Product/Category and/or Product/Disposition should be replicated on your *Production site* in the same way as it was done on your *Test site*. All of these items get unique identifiers, so if a customization references any unique identifiers, the order in which you create your categories, products, or dispositions becomes important.

Also, be sure to check the proper interface visibility checkboxes when you are creating/updating your Products and Categories.

Items to Compare Side-by-Side

For the following items, you will take inventory of the changes in your *Test site* and duplicate those changes in your *Production site*. An easy way to do this is to launch the agent desktops for both Test and

Production, display them side-by-side, and compare them, bringing your *Production site* in to line with Test or confirming that the *Production site* has production values. Some of the items for which you will do this comparison include:

- Site Configuration:
    - Interfaces
    - Mailbox Configuration Options
    - Distribution Lists
    - Configuration Settings
    - Message Bases
    - Engagement Engine Rules
- Profiles/Staff Groups
- Service Level Agreements
- Customizable Menus
- General Service Configuration:
    - Products, Categories, and Dispositions
    - Custom Fields
    - Variables
    - Channels and Channel Accounts
- Sales Configuration
- Marketing Configuration
- Outreach (Questions and Surveys)

Additional Information

Here are useful links related to the topics above.

HELPFUL LINKS FOR VARIOUS COMPONENTS

TOPIC	HYPERLINK
Deploying custom objects	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_custom_objects.13.13.html
Configuring guided assistance	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/dad_guided_assistance.31.2.html

HELPFUL LINKS FOR VARIOUS COMPONENTS

TOPIC	HYPERLINK
Interface options and settings	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_multiple_interfaces.21.4.html
Configuring mailboxes	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_multiple_interfaces.21.15.html
Managing distribution lists	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_email_management.16.15.html
Exporting configuration settings	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_site_configuration.15.8.html
Editing message bases	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_site_configuration.15.22.html
Configuring engagement engine rules	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/css_admin_chat_configuration.75.18.html
Customizing profiles	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_staff_management.9.2.html
Configuring service level agreements	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/css_admin_SLAs.49.2.html
Customizable Menus	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_customizable_menus.10.1.html
Organizing information with products, categories, and dispositions	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_customizable_menus.10.11.html
Adding and editing custom fields	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_custom_fields.12.2.html
Adding variables	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/crm_admin_business_rules_management.11.13.html
Exporting and importing guides	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/dad_guided_assistance.31.13.html
Channels	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/socex_channels.90.1.html
Sales Process	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/sa_sales_process.94.1.html
Outreach and Feedback	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/Marketing_and_Feedback_intro.78.1.html

Enterprise Deployments

For organizations with multiple sites, or who need to have multiple in-house developers working on sites collaboratively, change management becomes crucial and challenges may arise. By understanding the challenges, you can develop ways to mitigate the risk.

The main challenge that arises in Enterprise deployments is preventing multiple developers from overwriting each other's work. This can be managed by using a reliable code management system and preventing all other developer access so that changes do not circumvent your system. In other words, require your developers to bring in their changes through the front door (code management system) instead of through a side door.

A less common approach an organization may take to manage code changes is to create developer-specific sites. Although this option may be expensive, it gives developers flexibility and better reduces the problem of developers stepping on each other's code. The challenge introduced by this approach is how to keep the configuration of the sites in sync.

Recommendations

Consider the following when developing your change management strategy regarding enterprise deployments.

Utilize Code Management Software

Code management software creates a single door through which all code must go through and can simplify your deployments.

Designate a Change Coordinator

Designate someone in your organization to coordinate and schedule the various changes within your environments. Also, if an Oracle Client Success Manager (CSM) is assigned to your account, your change coordinator should engage the CSM to ensure coordination with schedules, versioning, or business impact that could affect your change management initiatives.

Develop Roll-back Plans

Encourage the development of roll-back plans, so that when an emergency occurs related to your latest deployment, you are able to quickly return to a prior working state with minimal impact.

Upgrades

From time to time, you will want to consume new upgrades to the Oracle RightNow Service product. Therefore, you should become familiar with the upgrade process and how to request an upgrade. Ultimately, you can contact Customer Care who will familiarize you with the process.

The Upgrade Guide is intended for Oracle RightNow Cloud Service administrators or any staff member responsible for configuring, administering, or upgrading an Oracle RightNow Cloud Service site.

The online upgrade guides will help you upgrade your installation by providing an overview of the upgrade process, describing specific steps and showing you how to test and evaluate your upgrade. A series of guided questions will lead you through only the sections that apply to your installation.

When new releases or service packs are made public, you will want to first test them against your *Test site*. This gives you a chance to identify and address potential problems and issues. If you discover problems that need attention, you can make changes in your *Test site* first to confirm that the problems are resolved before upgrading your *Production site*.

Recommendations

Your change management strategy regarding upgrades should include the following considerations.

Always Run on a Supported Release

As an Oracle RightNow Cloud Service customer, you will want to take advantage of the features, enhancements, and fixes that come with the latest version of the Oracle RightNow Cloud Service. By being on a supported version, Customer Care engagements are likely to be more efficient and effective. Therefore, stay current and make every effort to begin the upgrade process when new versions of Oracle RightNow Cloud Service are released.

Upgrade Your Test Site First

Upgrade your *Test site* and perform testing against it. Once you are satisfied with the results of your testing, upgrade your *Production site*.

Review the Latest Upgrade Guide

Follow the link below to see our Upgrade Guides. It will familiarize you with the process and how to request an upgrade.

Consult with your Client Success Manager

If you're a large organization and have an Oracle Client Success Manager (CSM) assigned to your account, consult with your CSM since they are very familiar with and can facilitate the upgrade process.

Additional Information

Here are useful links related to upgrades.

HELPFUL LINKS FOR UPGRADES

TOPIC	HYPERLINK
Upgrade Guides	https://cx.rightnow.com/app/answers/detail/a_id/5167

HELPFUL LINKS FOR UPGRADES

TOPIC	HYPERLINK
HMS Guide	http://documentation.custhelp.com/euf/assets/docs/unversioned/pdfs/hms_guide.pdf
Test Site Product Description	https://cx.rightnow.com/app/answers/detail/a_id/5483/kw/test%20site
Customizations and Test Site Refreshes	https://cx.rightnow.com/app/answers/detail/a_id/5129
Partner Clone Site Request (North America)	https://cx.rightnow.com/app/answers/detail/a_id/5663/kw/test%20site
What's Included in a RightNow Upgrade?	https://cx.rightnow.com/app/answers/detail/a_id/4606/kw/test%20site

Additional General Links

The following are some general links that may be helpful.

ADDITIONAL HELPFUL LINKS

TOPIC	HYPERLINK
Technical Documentation and Sample Code	https://cx.rightnow.com/app/answers/detail/a_id/5169
Documentation for Oracle RightNow CX Products	https://cx.rightnow.com/app/answers/detail/a_id/5168/related/1
Oracle RightNow Cloud Service User Guide	https://documentation.custhelp.com/euf/assets/docs/may2013/olh/wwhelp/wwhimpl/js/html/wwhelp.htm

Conclusion

Change management is crucial to your business. The Oracle RightNow Cloud Service includes many flexible and robust capabilities to enable integration across the enterprise. As the depth of integration and customization increases it is important to ensure the many moving parts are managed and coordinated across the many resources who may be working on any given project. Changes can be easily managed if you know about them, know the best practices to managing them, and know how to engage with your Oracle team to best implement changes. The guidelines provided here are very specific to your Oracle RightNow Cloud Service product but can be aligned with your existing change management processes.

Oracle RightNow Cloud Service Change Management Guide
October 2013
Author: Sean Montgomery and Jeremy Holck
Contributing Authors: Fabio Cavalcante, Tom Cloos, Luke Davison, Christopher Martin, James McDowell, Jens Lundell, Roderick Peace, Patrick Walsh, Chris Warner

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com

Oracle is committed to developing practices and products that help protect the environment

Hardware and Software, Engineered to Work Together