

Leveraging HR Self Service as a Deployment Platform for Standard and Custom Functionality

Kimberly A. Sowers
Rochester Institute of Technology

Abstract

RIT deployed limited Employee and Manager Self Service functionality in 2002. The implementation included cultural, business process and technology changes. Since then the platform used by this functionality has been leveraged to deploy additional standard and custom features in support of an Institute wide strategic initiative to empower employees and students via self service technology. How we did this and our future plans will be discussed from a cultural, business process and technology perspective.

About Rochester Institute of Technology

Before beginning the discussion on how we've successfully leveraged HR self service as a deployment platform, it will help to know a little about our implementation. RIT initially implemented the Oracle eBusiness Suite Applications version 10.7 Financials (General Ledger, Accounts Payable, Purchasing, Fixed Assets, Cash Management) in 1998. The HR (including basic benefits) and Payroll modules were implemented in 1999. We upgraded to 11.5.5 in 2002, which allowed us to implement Employee Self Service, Manager Self Service and Standard benefits. We upgraded to 11.5.9 in 2004. In 2007 we went live with iRecruitment. Currently, we're running 11.5.9 with HR FP K RUP 2 and are in the process of upgrading our entire Oracle eBusiness Suite Applications implementation to R12.0.3.

RIT employees approximately 3,300 employees. We have approximately 15,600 students enrolled with approximately 7,500 of those students being paid as student workers. We run a bi-weekly payroll for our hourly employees and student workers and a semi-monthly payroll for our salaried employees.

Self Service is a journey

Implementing self service functionality in the Oracle eBusiness Suite Applications is a journey, not a destination, because as users become empowered to do more themselves via self service they'll want the ability to do more. Evaluating whether or not it's the right time to embark on this journey means evaluating your organization, the functionality offered by Oracle and the technology needed to support that functionality.

The organizational perspective is the most challenging to assess. Deploying self service functionality provides a new view on

- Who interacts with the HR / Payroll application
- How traditionally centralized employee management activities are performed
- Who does the "programming" supporting the new functionality
- How application changes are tested and migrated into production
- How to support self service users when they have issues / questions

Surprisingly, asking employees to interact with the HR / Payroll application didn't cause as many issues as initially expected. The one situation we did encounter is discussed later in the paper. While some parts of the Institute have pushed back on being asked to enter personnel management transactions on-line ("HR is asking us to do their jobs"), most areas have embraced the concept of self service as they see the value gained from the automation of transactions that used to be paper based.

Oracle's eBusiness Suite Applications use the concept of personalization to "customize" standard self service functionality to meet an organization's business needs. Personalization is built in functionality that allows a business analyst or power user to modify how the eBusiness Suite applications work without the need to involve the technical support or programming team. At RIT, our HR business analysts develop most of the "programming" changes needed to deploy standard eBusiness Suite self service functionality. These programming changes are not subject to the same change management rigor imposed on our systems development team (the technical support team for our Applications implementation that does all non-self service custom functionality development).

Supporting our self service deployment has required training for both our standard Help Desk as well as our eBusiness Suite specific support team (browser issues in particular cause users to seek assistance, especially since our implementation isn't supported with Internet Explorer 7 and we have a large number of users using browsers like Mozilla, Safari, and Firefox). Additionally, the HR and Payroll web sites have been expanded to include information on how to use the self service functionality we've deployed. Our iRecruitment login page for external candidates includes a link that provides instructions to applicants.

When assessing what functionality you want to deploy, you want to consider items such as

- How many users will the functionality be deployed to? Will the deployment be 'big bang' or phased?
- How many transactions that are currently processed centrally will be decentralized by implementing the functionality?
- What's the risk in getting 'bad' data by decentralizing the transactions selected?

At RIT, we've used both phased and big bang approaches to implementing self service functionality. This decision is often driven by the type of functionality being implemented and the target audience of the functionality. We've also addressed the "bad data" issue by automating as many data validation processes as possible during transaction data entry.

From a technology perspective, you need to determine whether you want self service users to be able to access some or all of the self service functionality when they're not connected to your organization's official network – this is the concept of public (accessing functionality via the standard Internet) versus private (accessing functionality via a Virtual Private Network when not physically located at one of your organization's sites and using your organization's official network when you are physically located at one of your organization's sites) access. You also need to assess whether your technical team possesses the appropriate skill sets to support the new technology used in deploying the self service functionality (e.g., fluency in Java and the Oracle Applications Framework standards instead of the Oracle Forms and Reports toolsets that are used in the traditional "professional" interface) if you want to customize or extend the self service applications.

At RIT, we started our self service deployment only allowing access to Employee Self Service functionality via our private network. We now allow access to Employee Self Service via public and private networks. Our technology staff had Java knowledge when we started extending the standard Oracle eBusiness Suite Applications functionality, although there was some time needed to learn the Oracle Applications Framework standards.

In the beginning ...

RIT initially implemented self service functionality following our upgrade to 11.5.5 in order to improve the accuracy of employee maintained data (e.g., address, phone number); free HR and payroll staff from entering data in order to be able to better serve their constituencies; and to obtain financial savings from not sending paper documents via the US Postal Service. The transactions that were initially deployed (optional employee transactions like updating personal, banking and federal tax withholding information) were chosen in order to introduce the concept of self service computing to RIT's employee community. To assist in the transition to the self service model, paper forms for these transactions continued to be accepted for a limited period of time.

Our initial deployment of Employee Self Service functionality presented us with challenges in creating Oracle eBusiness Suite Applications accounts for all employees; distributing the passwords for those accounts and providing users with general instructions on how to use the new functionality. When the functionality was initially deployed, accounts were manually created for all employees who didn't already have an Oracle eBusiness Suite Applications account. The account name was based on the employees RIT computer account if they had one, otherwise their initials and the constant 'ORA' were used to create the account. Initial passwords were set to be a combination of initials and employee number. Information on how to access the Applications as well as how to figure out your password was sent to all employees with new Oracle eBusiness Suite Applications accounts via a letter.

After deploying the Employee Self Service functionality, a process needed to be developed to create user accounts for new employees. The development team wrote an alert that sends a daily email listing all newly hired employees to the data center staff, who creates the accounts for new employees (again, the Oracle eBusiness Suite Applications accounts are created using the same account name as the employees RIT computer account). After the account is created, a help desk person resets the password. The new password and detailed instructions on how to access the Applications are sent to the new employee via a letter. Since there is a requirement to have an employees Oracle eBusiness Suite Applications account be the same as their RIT computer account, there are no plans in place to automate this process at this time. If the dependency between RIT computer names and Oracle eBusiness Suite Applications accounts ceases to exist we will use the standard self request an account functionality to replace our existing manual process for creating eBusiness Suite Applications accounts.

Moving forward ...

About a year after we initially deployed Employee Self Service, senior management announced that over the course of several months all employees who were using direct deposit to receive their salary would no longer receive paper deposit advice slips – information related to the transfer of their salary to their bank(s) would be available on-line using Employee Self Service. Since this was our first deployment of mandatory Employee Self Service functionality to the entire employee base we decided to do a phased implementation of on-line pay slips and the ability to view previous years W-2's on-line (which is standard functionality). This would allow us to assess any performance impact that might occur as well as work through any support issues.

Since we use a custom program to produce our deposit advice slips, we created a custom form that our Payroll department uses to enter the GL account codes for the departments that should receive on-line deposit advice. The custom program references this table and doesn't print deposit advice slips for employees being paid from those accounts. As on-line pay slips were deployed to more departments, their GL account code was added to the table populated by this form and employees in that department were omitted from the printed deposit advice slips being produced. Today, aside from specific categories of employees, all employees who receive their salary via direct deposit must view their deposit advice information on-line via Employee Self Service.

Some challenges were faced in the deployment of this functionality. One of the largest challenges was the fact that the ASP hosting our Oracle eBusiness Suite Applications implementation required the use of a VPN to access the Applications from off-site. This meant that an employee who wanted to view their pay slip at home needed to download and install the custom RIT VPN software in order to be able to do so. This placed a substantial burden on numerous employees, so a large number chose to view their pay slips at work. We have since switched ASP's – our current ASP allows the use of a public middle tier so access to selected Employee Self Service functions can be accessed from off campus using basic Internet connectivity.

The VPN challenge was one we foresaw and anticipated needing to address. A challenge we didn't fully anticipate was the resistance to using Employee Self Service to access pay slips. While we expected some employees who had limited access to PC's to be reluctant to embrace the new functionality, the resistant employees were those who used a PC in their day to day activities and were more upset about the fact this change was implemented without their input than they were about the change. This

resistance reached a point where employees were using a public mail list to share ideas on how to continue receiving a paper "deposit advice slip" by having a very small portion of your salary being paid by check.

Continuing along the path ...

Having gotten the entire employee community familiar with the concept of Employee Self Service, our next step was to do a pilot implementation of on-line benefits enrollment. We had been displaying the benefit choices an employee currently had in effect since we initially deployed on-line pay slips. In Fall 2003 the decision was made to offer employees in selected units in our Finance and Administration division (e.g., HR, Payroll, the Controller's Office) as well as our Information Technology Services division the opportunity to complete their benefits enrollment process on-line if they wanted to (about 150 people in all). The pilot was successful, so RIT has been using the on-line benefits enrollment functionality for open enrollment for all employees since Fall 2004.

We do continue to face some challenges related to the use of on-line benefits enrollment. One of the biggest is the number of employees who do not access Employee Self Service on a regular basis, so that they don't know their password when they go to do their benefits enrollment. Many employees work with their significant other while performing the enrollment or running what-if scenarios – often this occurs outside the normal hours our Help Desk is open. We've decided not to enable the 'reset my password' functionality in the Oracle eBusiness Suite Applications due to concerns raised by our Information Security office, so every year we deal with frustrated users who can't access Employee Self Service when they want to because they've forgotten their password. We now encourage employees, in the enrollment materials we distribute ahead of time, to test their eBusiness Suite accounts prior to open enrollment starting in order to help alleviate this frustration.

Another challenge we face is that not all RIT employees have access to a PC during their normal workday. This has been resolved by the creation of 'kiosks' in areas with large numbers of employees who don't use PC's in their daily job duties (e.g., food service, facilities maintenance). Additionally, benefits "labs" are held a few times during the open enrollment period where employees can stop by a lab staffed by HR personnel who not only can reset their passwords but can also assist in the enrollment process itself.

The challenge we have yet to fully address is the fact that employees don't fully read the instructions that have been placed on the enrollment pages. While it's true that this also happened when benefits enrollment was done on paper, the employee could continue their enrollment process – an issue caused by not following instructions was uncovered by an HR person days or weeks later and addressed at that time. Now, when an employee doesn't follow instructions, they usually cannot continue the enrollment process until they fix their issue. This can lead to very frustrated employees.

The future for Employee Self Service

We continue to add functionality to Employee Self Service. We've added the ability for users to change their own passwords whenever they want to. Being an educational institute, it's important to know the educational background of our employees so we've deployed the "My Schools" functionality. Recently, we've modified the "My Personal Information" functionality to enable employees to enter data used by our Emergency Mass Notifications System (e.g., IM system and screen name). We've also added a function that allows employees to opt out of being notified by the Emergency Mass Notifications System. As our business processes change or new needs arise, we will consider whether Employee Self Service is an appropriate part of the solution.

At RIT, we prefer that all mandatory self service functionality an employee needs to use (on-line payslip, benefits enrollment, input data used for Emergency Mass Notifications) be deployed via Employee Self Service. We also prefer to leverage standard Employee Self Service functionality as much as possible when implementing these solutions. We also deploy all new functionality using the Oracle Applications

Framework. A challenge we face is that other organizations on campus that want to deploy mandatory functionality to employees often want to deploy a user interface that has a different look and feel than the standard Oracle eBusiness Suite Applications.

Including the student workers

In late 2005, it was decided to extend selected Employee Self Service functionality to student workers. They would be able to view their pay slips on-line, view their personal information stored in the Oracle eBusiness Suite Applications and use a link to access RIT's Student Information System (SIS) where they could update their personal information. (Note: at RIT the student records application is the system of record for any data pertaining to students – information for student workers needed by the eBusiness Suite is loaded into the Applications via a data synchronization process, which is why students need to maintain their personal information in SIS.)

There were 2 major differences in how we deployed Employee Self Service to students versus how we deployed it to employees. First, there was no phase out of paper deposit advice slips. Students were notified for 3 pay periods prior to the go-live date that as of March 20, 2006 paper deposit advice slips would no longer be produced. Students who received their salary via direct deposit would either have to use Employee Self Service or their banks on-line banking features to see how much was deposited into their accounts each pay day.

Secondly, student workers were allowed to opt in to using Employee Self Service. Detailed instructions on how to create an eBusiness Suite account are available on the Payroll department web site. Any student worker who wants access to the Oracle eBusiness Suite Applications logs in using a generic account called RITSTUDENT. After logging in, the student provides information about themselves (e.g., the name they're paid under, their employee number). After verifying that the student is indeed an active student worker and that they don't already have an Oracle eBusiness Suite Applications account, the student is allowed to create their own Oracle Applications account – with no account naming standards enforced. The final step of the process is to verify that the account name the student wishes to use doesn't already exist.

Adding Manager Self Service

December 2002 was the start of our deployment of Manager Self Service (MSS). This deployment continues today as new departments are formed or managers decide to push the functionality available in MSS lower in their organization. The functionality that is being deployed includes personal information about a manager's staff as well as the ability execute selected personnel actions (e.g., change supervisor). The data displayed to a manager is limited to the employees reporting to them and their subordinates' employees if a manager manages managers.

This security is implemented by using standard HRMS security, which required creating security profiles based on our organizational hierarchy using the *Security -> Profile* screen. We then linked the security profile to the responsibility it applies to via a responsibility level update to the *HR: Security Profile* system profile option. We've created approximately 100 responsibilities for managers who want access to this information so we've executed this process for each of those responsibilities. Further details on how to do this in 11i can be found in Metalink note 394083.1.

Today we offer two dozen custom reports to users of manager self service in addition to the standard personal information and change supervisor functionality. The reports are implemented as Oracle Discoverer workbooks that are created by our HR business analysts and saved to the database. The reports use standard Oracle and Noetix views. As needed, our technical support team creates custom views to make it easier for the HR staff to link data or to address query performance issues. These reports leverage our existing responsibility security via built in features from Oracle and Noetix. Our custom views replicate this security.

Our Manager Self Service report deployment strategy is to focus our efforts on reporting that's beneficial to the entire Institute rather using limited business analyst resources to assist selected colleges or divisions. This lead to one of the challenges we face with the Manager Self Service reports, which is creating and deploying reports that are useful to the entire Institute, rather than offering reports that are beneficial to an individual college, division or department. When reports are offered for a specific division or college, they're usually related to services that a division or college provides to others campus wide (e.g., interpreter services, student affairs).

We also faced some challenges when we upgraded to Discoverer 10g in locking down data we didn't want users to see. For example, while a manager can see an employees salary they shouldn't be allowed to see birth date or SSN. If those fields are available in a view being queried by the worksheet used for a report, Discoverer's default functionality will allow experienced Discoverer Viewer users the ability to add those data fields to the pre-written report. Using Discoverer Administrator we needed to manually prevent users from being able to do this for each report we've deployed.

Some Metalink notes that we found useful when integrating Discoverer 10g with the Oracle eBusiness Suite were

- 278095.1 – How to Create a Link to a Discoverer Workbook in Apps11i
 - Refer to note 471303.1 for R12
- 433397.1 – How to register Discoverer reports with the OracleOasis package and find its version
 - This discusses implementing eBusiness Suite 11i security in Discoverer
- 237607.1 – ALERT: Required and Recommended Patch Levels For All Discoverer Versions
 - Doesn't cover R12

Leveraging the platform

After successfully deploying our initial Employee Self Service and Manager Self Service functionality, we decided to start leveraging the self service platform to address specific business process efficiency, data quality, accountability and transaction timeliness challenges we were facing. In order to determine which functionality we would deploy new self service functionality for we used three criteria to assist in our decision making –

- The volume of transactions being considered for automation
- The business processes used in the organizations the transactions would be deployed to (e.g., are transactions approved by the same number and level of approvers in all colleges and divisions or do the approval processes vary?)
- The training and support efforts needed to support the deployment of the functionality (specifically, the number of users using the functionality and the complexity of the functionality deployed)

At RIT we have an Additional Payment (Add Pay) form. This is typically used by managers to request a one time payment for an employee for work performed beyond their regular salary. After reviewing the standard Oracle Individual Compensation Distribution (ICD) functionality, HR and Payroll decided that it could be implemented "as is" to replace the paper Add Pay forms we were using. This functionality was selected to be the first self service functionality deployed Institute wide because there was a large volume of Add Pay forms to be processed each pay period. While numerous departments submitted these forms, a limited number of departments created the majority of them every pay period. Deploying ICD to this subset of Add Pay form users meant we could train a relative small number of users in order to automate a substantial number of on-going paper transactions.

As an institute of higher education, RIT hires a significant number of adjuncts to teach courses each quarter. This is a regularly occurring high volume transaction executed by all eight of our colleges. The decision was made to automate the hiring of an RIT employee or an individual RIT considers an active adjunct via self service functionality. This "hire adjunct" functionality is an extension of the standard Oracle Individual Compensation Distribution functionality used to automate our Add Pay forms. RIT's technical support team has inserted custom software in this standard functionality to validate the GL code

the adjuncts salary should be charged to. It also requires that the course number the adjunct faculty is teaching is entered before processing the transaction.

The following screen shots show selected aspects of the hire adjunct functionality. Figure 1 is standard ICD functionality (which is used in the Add Pay functionality previously mentioned). Figure 2 is also standard ICD functionality – it's implemented slightly differently for hire adjuncts than Add Pay because the elements used to pay adjuncts have specific effective start / end dates whereas the elements used in the Add Pay functionality do not. Figure 3 is the customization we inserted into the standard ICD functionality. Note how the look and feel of this functionality is exactly the same as the preceding screens.

This screenshot shows a web-based application for managing compensation awards. At the top, there's a navigation bar with links for 'File', 'Edit', 'View', 'Favorites', 'Tools', and 'Help'. Below that is a toolbar with icons for 'Back', 'Forward', 'Search', 'Favorites', and other common functions. The address bar shows the URL: https://mybzdev.rit.edu/oa_HTML/OA.jsp?_jcr_IHR_PERSON_TREE_TOP_SS8_n=8008_t=53358235&language_code=en&apCalledFrom=RIT_BEN_SS_HGR_ICD&OFunc=RIT_BEN_SS_HGR_ICD&apc=4. The main content area has a header 'Human Resources' with the RIT logo. Below it, the title 'ICD Award Entry with Cost Account - RIT: Compensation Type' is displayed. On the left, there's a sidebar with 'Effective Date' set to '24-Feb-2008'. In the center, there are fields for 'Employee Name' (set to 'Manager Kimberly Sowers'), 'Organization Email Address' (left empty), 'Employee Number' (left empty), 'Department' (set to 'ITS'), and 'Job' (set to 'Professional/Administrative'). To the right of these fields is a table titled 'Select Compensation Type' with columns for 'Frequency' and 'Status'. The table contains eight rows, each representing a different compensation type. The first row, 'Adjunct Winter Course 1', is selected (indicated by a green circle next to the radio button). The other seven rows are empty circles. A note at the bottom of the table says 'TIP: Distributions that are Pending Approval may not be changed.' At the bottom of the page, there are links for 'Diagnostics', 'Home', 'Logout', and 'Preferences', along with a 'Privacy Statement' link. The status bar at the bottom of the browser window shows the time as '9:34 AM'.

Figure 1 - Select pay element screen from standard ICD

Figure 2 - Enter payment amount

Figure 3 - Enter salary account and course information

Building our own functionality

After building our self service deployment expertise (functional, technical and support) by deploying standard Oracle functionality and customizing standard Oracle functionality, we decided we were ready to write our own totally custom self service functionality. The business process we chose to automate first

was our Assignment Costing Change Form. This is a high volume transaction used by a limited number of departments on campus – usually to change the GL accounts salaries are being charged to for employees who are performing sponsored research or working on grants. These accounts and / or percentages being charged to a specific account can change as frequently as monthly.

This functionality is totally custom, leveraging standard Oracle API's and packages. It was developed by RIT's eBusiness Suite's technical support team and leverages the Oracle Applications Framework (OAF) so that it has the same look and feel as standard Oracle functionality. In fact, if you didn't know this was custom functionality you couldn't tell it wasn't standard Oracle functionality based on how the functionality looks and works. Logic was added to the on-line functionality to address some of the specific issues the HR and Payroll departments were experiencing with the paper forms – primarily ensuring that the GL accounts salaries being charged to were appropriate and that the percentages being charged added up to 100%. The functionality we implemented also routes all assignment costing changes to one individual per our oversight rules using workflow.

The following screen shots show this custom development. In Figure 4 you see where the user has the opportunity to change either the current or future costing of an employee. Note that users must provide an end date for the current costing in order to enter future dated costing. Failure to do so will result in an error. This is illustrated in Figures 5 and 6. The current costing is end dated in Figure 7, and the upper part of the confirmation page is shown Figure 8. Notice how the look and feel of this totally custom functionality is exactly like the standard self service look and feel. Wherever possible standard functions are used (e.g., enter account information, person search) to ensure this occurs.

Figure 4 - Update current or future payroll costing information

The screenshot shows the Oracle Self-Service Human Resources Extra Information Type page. At the top, there's a navigation bar with links like File, Edit, View, Favorites, Tools, Help, Back, Search, Bookmarks, Print, High Speed Print, Preview, Options, Duplex, and View Print List. Below the navigation is the RIT Human Resources logo and a banner.

The main content area is titled "Assignment Cost Change: Extra Information". It has sections for "Current Assignment Cost Information" and "Future Assignment Cost Information".

Current Assignment Cost Information:

Select Object:	Update	Add	Employee Name	Employee Number
Select Status:	Cost Allocation			
<input checked="" type="radio"/> New	50	01	75100 SYSTEMS DEVELOPMENT ADMN PROF - SALARIES	70050 INFORMATION & TECHNOLOGY SERVICES ADMN PROF - SALARIES
				Effective End Date
100% 01 75100 70050 55.00000 00000 01-APR-2001				

Future Assignment Cost Information:

Select Object:	Update	Add	Proportion	Entity	Department	Object	Expense	Project	Program	Effective Start Date	Effective End Date
<input type="radio"/> New	50	01	75100 SYSTEMS DEVELOPMENT ADMN PROF - SALARIES	70050 INFORMATION & TECHNOLOGY SERVICES ADMN PROF - SALARIES	55	00000 NONE	00000 NONE	00000 NONE	00000 NONE	01-Mar-2008	
<input checked="" type="radio"/> New	50	01	75200 DATA CENTER OPERATIONS ADMN PROF - SALARIES	70050 INFORMATION & TECHNOLOGY SERVICES ADMN PROF - SALARIES	55	00000 NONE	00000 NONE	00000 NONE	00000 NONE	01-Mar-2008	

Buttons at the bottom include Cancel, Save For Later, Back, and Next.

Figure 5 - Data entry for future dated costing change that didn't end date the current costing

The screenshot shows the Oracle Self-Service Human Resources Review page. At the top, there's a navigation bar with links like File, Edit, View, Favorites, Tools, Help, Back, Search, Bookmarks, Print, High Speed Print, Preview, Options, Duplex, and View Print List. Below the navigation is the RIT Human Resources logo and a banner.

The main content area is titled "ERROR". It displays the following message:

Proportion = 200 on 01-MAR-2008. Proportion = 200 after 01-MAR-2008.
Click BACK to return to the previous page to review and correct any errors, click CANCEL to cancel this action.

Buttons at the bottom include Cancel, Printable Page, Save for Later, and Back.

Figure 6 - Error message showing over allocation of costing due to not end dating current costing

Employee Name
Organization Email Address

Employee Number

Enter any changes below. Click Ok to continue this action, click Cancel to cancel this action and return to the previous page.

Cost Allocation	100% 01 75100 70050 55.00000 00000 01-APR
Effective End Date	29-feb-2008

[Diagnostics](#) [Home](#) [Logout](#) [Preferences](#)

Copyright (c) 2006, Oracle. All rights reserved.
[About this Page](#)

[Diagnostics](#) [Home](#) [Logout](#) [Preferences](#)

[Privacy Statement](#)

Figure 7 - Entering the end date of the current costing

Employee Name
Organization Email Address

Employee Number

Review your changes and, if needed, attach supporting documents
 Indicates Changed Items

Extra Information Type

Future Assignment Cost Information

Proposed	Proportion 50 Entity 01 Department 75200 Object 70050 Expense 55 Project 00000 Program 00000
Proposed	Proportion 50 Entity 01 Department 75100 Object 70050 Expense 55 Project 00000 Program 00000

Effective Start Date 01-Mar-2008

Current Assignment Cost Information

Current	Cost Allocation 100% 01 75100 70050 55.00000 00000 01-APR-2001	Proposed	100% 01 75100 70050 55.00000 00000 01-APR-2001
Effective End Date	29-Feb-2008		

Figure 8 - Confirmation of costing change

The next transaction we built custom functionality for was our payroll change form. This paper form is used by managers of student workers to adjust the hours worked for an employee (e.g., due to a missed punch). The on-line functionality links to our Kronos timekeeping system and automates many of the manually checks the payroll department used to perform with the paper forms (e.g., ensuring that an employee isn't being paid twice for the same hours).

Figure 9 shows the search screen used to select the specific student worker whose timecard needs adjusting. Note how it looks like the person search screen in all standard Oracle functionality. Figure 10 shows the timecard details for the specific student for the pay period in question. Note how it shows all of the jobs the student has charged time to, ensuring that a student cannot be paid twice, for two different jobs, for the same hours. Figure 11 shows the interface the user uses to correct the students timecard. Note that every field relating to the costing, pay code and reason for adjust have drop down lists in order to ensure accurate data is entered. Lastly, Figure 12 shows the updated timecard for the student.

Employee Name	Employee Number	University ID	Type	Action
Burke, Adam			Student Worker	
Burke, Mr. David C.			Student Worker	
Burke, Ryan			Student Worker	
Burke, Sean			Student Worker	
Burke, Stephen			Student Worker	
Burkett, Michael			Student Worker	

Figure 9 - Search screen in Payroll Change Form functionality

Kronos Time Card and Pending Request Data

Status	Account	Entry Date	Punch IN	Punch OUT	Pay Code	Job Code	Hours	Details	Update	Delete
	01.96000.71350.30.00000.00000	12-Oct-2007	10:37 AM	12:25 PM	04		1.8			
	01.82600.71350.10.00000.00000	13-Oct-2007	05:00 PM	10:01 PM	03		5.02			
	01.47510.71350.10.02008.00000	15-Oct-2007	09:55 AM	01:09 PM	01		3.23			
	01.82600.71350.10.00000.00000	15-Oct-2007	01:52 PM	05:56 PM	03		4.07			
	01.82600.71350.10.00000.00000	17-Oct-2007	07:42 PM	11:07 PM	03		3.42			
	01.96000.71350.30.00000.00000	19-Oct-2007	10:38 AM	12:20 PM	04		1.7			
	01.47510.71350.10.02008.00000	22-Oct-2007	09:56 AM	11:57 AM	01		2.02			
	01.82600.71350.10.00000.00000	22-Oct-2007	01:47 PM	05:45 PM	03		3.97			
	01.82600.71350.10.00000.00000	24-Oct-2007	08:12 PM	11:18 PM	03		3.1			

Figure 10 - Pay period details for selected student worker

Payroll Change Request - Add New Time

This page is used for adding new change requests.

Employee Name	Pay Period Begin	12-Oct-2007
Employee Number	Pay Period End	25-Oct-2007
University ID	Pay Date	02-Nov-2007
Account Number	Entry Date	25-Oct-2007
Pay Code	IN (Hr:Min Am/Pm)	10:00
Reason for Adjustment	OUT (Hr:Min Am/Pm)	10:30

Figure 11 - Enter details on missing time clock punch

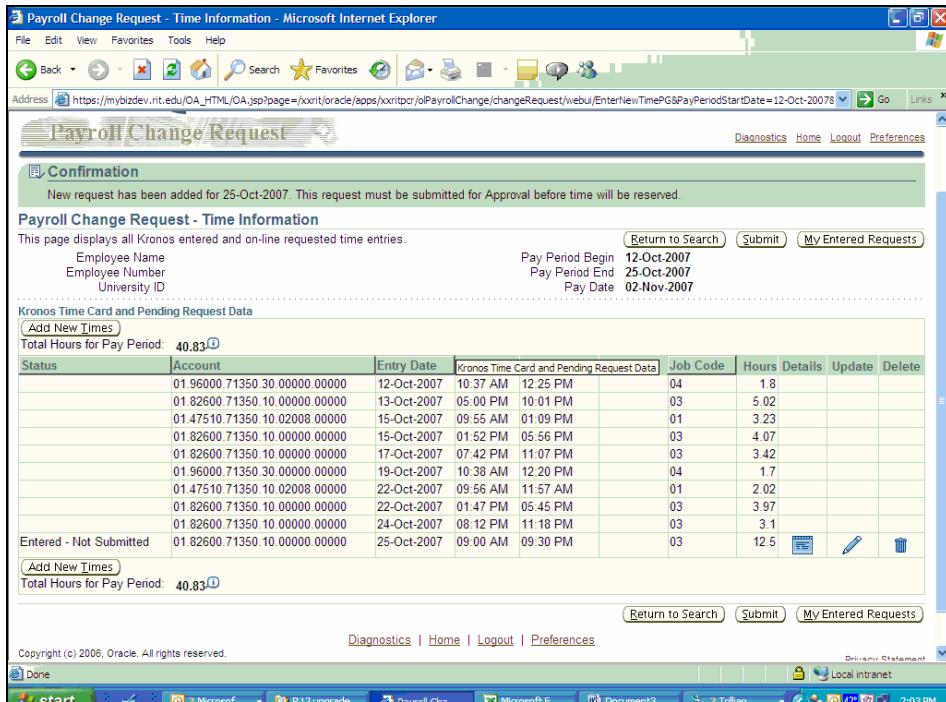


Figure 12 - Updated timecard

When deploying custom self service functionality it's important to use the correct version of JDeveloper for the specific Applications / ATG.H combination you're running. Metalink note 416708.1 contains useful information on this topic. Additionally, Metalink note 283158.1 contains other useful information about using the OA Framework.

Going off campus with self service

In June 2007 we replaced our home grown recruiting site with iRecruitment. This allows external candidates to create their own account in the Oracle eBusiness Suite Applications to manage their interaction with RIT while they're exploring job opportunities on campus (employees access the functionality via a new responsibility). This functionality was implemented primarily by the HR business analysts with limited support from RIT's technical support team.

iRecruitment offers a plethora of functionality. RIT decided to initially implement the ability for departments to create job postings to be displayed on the mycareer.rit.edu web site, allow candidates to review job postings and apply for those positions, and allow managers to review applicants resume and applications on-line. Managers can process an application on-line, letting candidates check on the status of their application, which facilitates the creation of EEO reports. In addition to this standard functionality we created some custom software to notify managers when an applicant applies for one of their positions and to send a weekly notice to all RIT employees letting them know what the newly posted positions are.

We encountered some challenges during this implementation related to managing Oracle Applications accounts (these same issues also exist when we hire student workers as employees). In order for our Employee Self Service functionality to work we need to link the Oracle eBusiness Suite Applications account used by the employee to their person record. Applicants who applied via iRecruitment have a person record as an applicant and another as an employee – the newly created employee Oracle eBusiness Suite Applications account needs to be linked to their employee person record. This issue has resulted in new processes being developed for use by our data center in the creation of new employee eBusiness Suite accounts.

We also faced similar challenges with employees who didn't want to use the new responsibility available to them as employees when applying for jobs using iRecruitment because they didn't want anyone at RIT to know they were looking for a new position.

Future plans

We expect to deploy iExpenses within the next 12 months. This deployment will start as a limited deployment to departments that tend to travel a lot (e.g., Admissions) then progressively be deployed to the entire Institute. We anticipate that this implementation will include some programmatic enforcement of Institute travel policies. We also expect it to include an electronic feed from our corporate credit card provider.

We also anticipate expanding our Employee and Manager self service offerings. Plans are underway to offer employees the Total Compensation Statements that are currently printed every other year on-line via Employee Self Service. Discussions are also underway on whether to allow employees to perform their annual conflict of interest disclosure on-line. In Manager Self Service there is an expectation that Performance Appraisals will be deployed. And in iRecruitment, consideration is being given to generating offer letters to candidates from the application and doing resume parsing.

Summary

This paper has shared RIT's self service deployment journey. The cultural, business process and technology changes we faced on this journey were shared.

Specific functionality deployed via self service as well as the rationale for deploying that functionality was discussed. It has shared reasoning behind phased versus "big bang" deployment of various functionality we've deployed as well as the challenges we encountered while deploying specific types of functionality.

Acknowledgements

The functionality discussed in this paper has been envisioned, developed, deployed and supported by individuals in RIT's HR, Payroll and Systems Development organizations (Christa Abugasea, Steve Boese, Julie Hawk, Laurie Jacobson, Scott Mitchell, Kitty Stappenbeck, Wei Wang, and Gina Williams). Without their daily efforts to support RIT this paper wouldn't have been possible.

About the author

Ms. Sowers works in the Information and Technology Services division at Rochester Institute of Technology where she is the Assistant Director for Financial and Administrative Systems. She has over 18 years experience with Oracle technology, including working with the Oracle eBusiness Suite Applications for over 12 years. She can be reached via email at kimberly.sowers@rit.edu.