

# Effective Change Management Strategies

A Maintenance Strategy for  
Technical and Functional Teams

Michael Parker

VP of Professional Services, Newmerix Corp

## Your Speaker – Michael Parker

- Newmerix VP of Professional Services
- 8 years with PeopleSoft Global Services and 12 years working with PeopleSoft products and technology
- Hundreds of PeopleSoft implementations and upgrades
- Deep knowledge of PeopleSoft application management best practices
- 19 years of experience with HR processes and best practices



# Newmerix

The Newmerix product suite reduces the cost and complexity of managing the ongoing, extensive changes inherent in packaged application environments. Our solution results in greater visibility and control over the application lifecycle and improved business agility.

# Enterprise Applications Require Change



## Business Process Changes

- **Overcome Competition**
- **Improve Service**
- **Cut Costs**
- **Improve Quality**
- **M&A**



## Vendor Driven Changes from Oracle and SAP

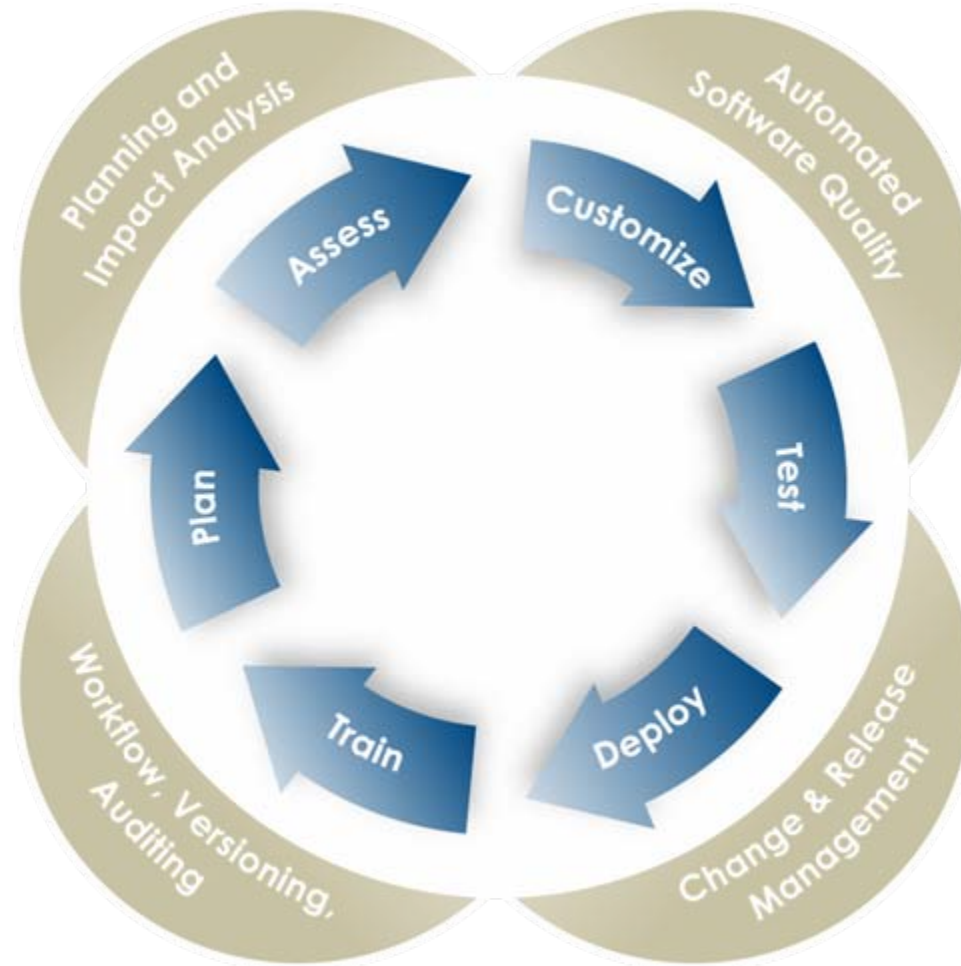
- **New Applications**
- **Patches**
- **Support Packs**
- **Upgrades**



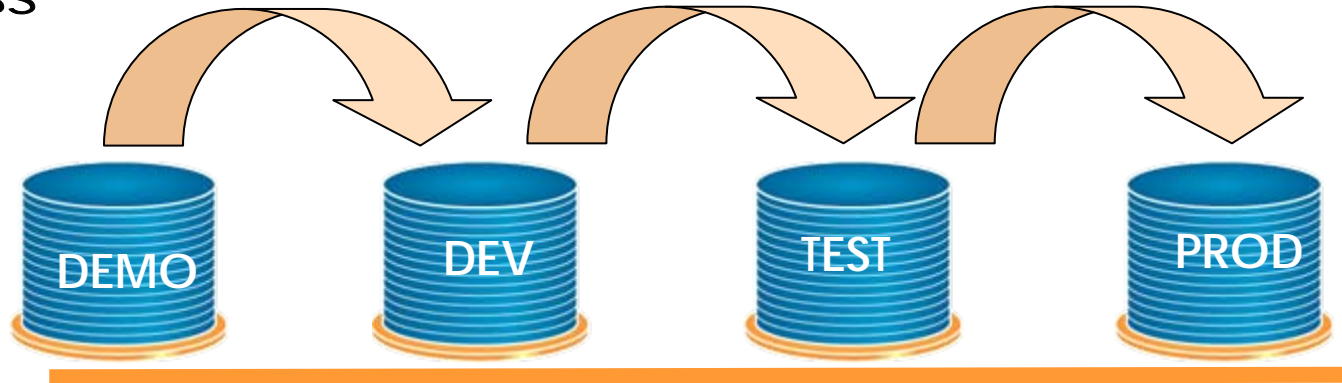
## Compliance and Best Practices

- **Federal, Provincial, and Municipal Regulations**
- **Best Practices (COBIT, ITIL, COSO)**

# The Lifecycle of Change



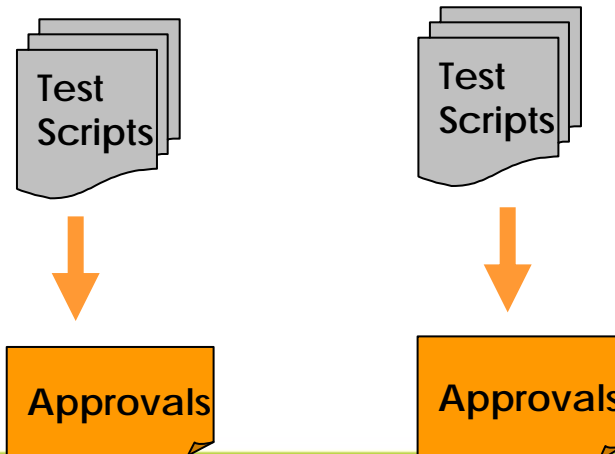
# Functional Users Involved Throughout the Change Process



Technical Team Requirements →



Functional Team Requirements →



# Challenges of Vendor-Driven Changes

#1: IT/Biz Relationship	Communicate changes to all involved
#2: Workflow Control	Define & enforce standard development workflows
#3: Approvals	Enforce approvals at key gateways
#4: Impact Analysis	Determine the impact of a change before migrating into any environment
#5: Separation of Duties	Define & enforce which users can perform specific activities in specific environments
#6: Systematic Testing	Conduct comprehensive, systematic testing of changes before deploying
#7: Documentation/ Audit Reporting	Document, track and report all changes related to the development process
#8: Restore Process	Version all assets and establish restore capability in the development process

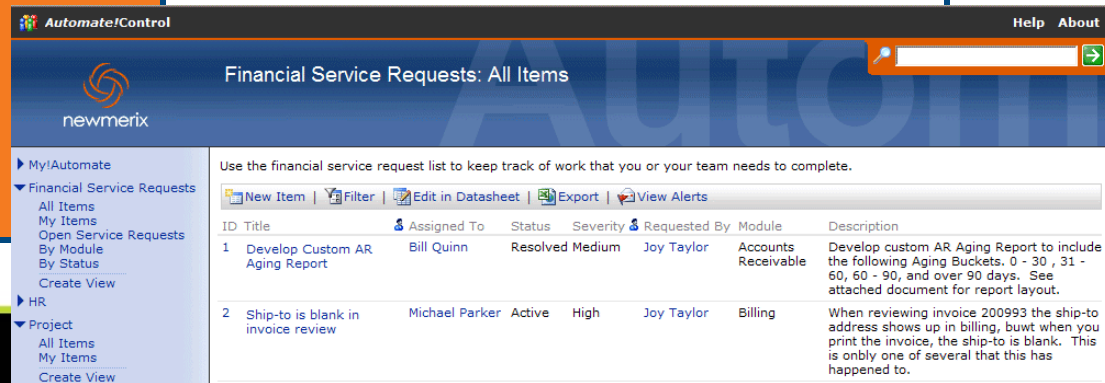
## Where are Functional Users Needed in the Process?

#1: IT/Biz Relationship	Communicate changes to all involved
#2: Workflow Control	Define & enforce standard development workflows
#3: Approvals	Enforce approvals at key gateways
#4: Impact Analysis	Determine the impact of a change before migrating into any environment
#5: Separation of Duties	Define & enforce which users can perform specific activities in specific environments
#6: Systematic Testing	Conduct comprehensive, systematic testing of changes before deploying
#7: Documentation/ Audit Reporting	Document, track and report all changes related to the development process
#8: Restore Process	Version all assets and establish restore capability in the development process



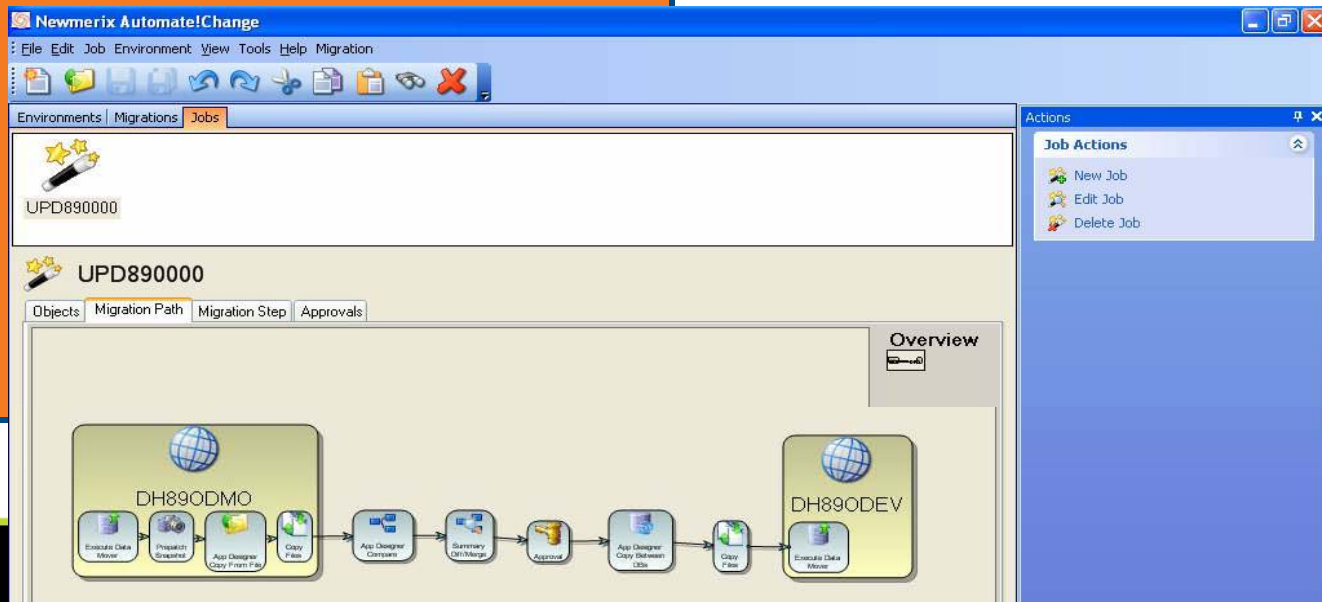
# #1: IT/Business Relationship

Manual	Automated
<ul style="list-style-type: none"> <li>■ Changes are communicated via emails, verbally at change control meetings, etc.— unauditible and unenforceable</li> <li>■ End users' service requests and enhancements enter a "black hole" with no way to track status or progress</li> <li>■ No method to centrally collect, categorize, prioritize, organize and track change requests</li> </ul>	<ul style="list-style-type: none"> <li>■ All change requests routed and prioritized based on predefined policies</li> <li>■ End users can submit and view the status of service requests they've submitted, creating a two-way communication between business users and IT</li> </ul>



# #2: Workflow Control

Manual	Automated
<ul style="list-style-type: none"> <li>Change procedures are maintained via Word/Excel – prone to missed steps</li> <li>Change procedures are not enforceable, creating SOX compliance/fraud issues</li> </ul>	<ul style="list-style-type: none"> <li>Predefined change control workflows manage each step in the change process through each environment</li> <li>Different workflows can be established for different types of changes</li> </ul>



# #3: Approvals

Manual	Automated
<ul style="list-style-type: none"> <li>■ Approvals communicated via emails, verbally at change control meetings, or in paper forms— unauditible and unenforceable</li> <li>■ Changes can get to production without the required approvals</li> </ul>	<ul style="list-style-type: none"> <li>■ All approval requests routed and prioritized based on predefined policies</li> <li>■ Changes are “locked down”— unable to complete migration until approvals are formalized with electronic signoffs and authentication</li> </ul>

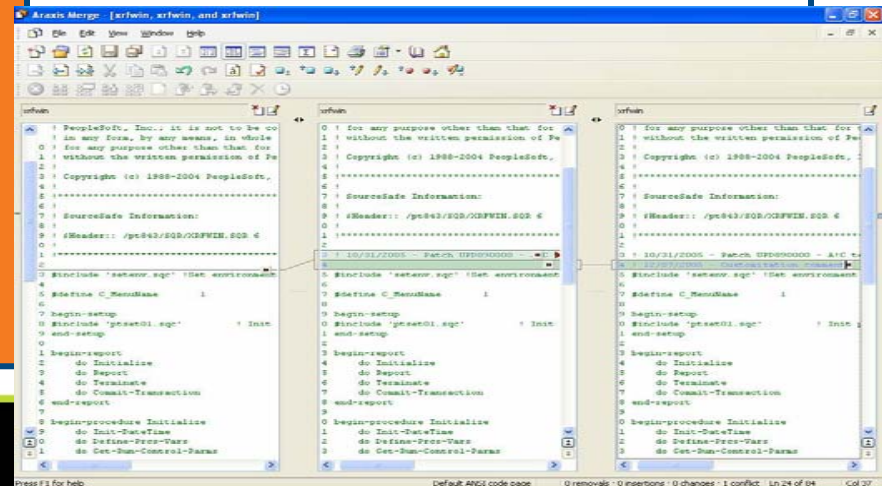
# #4: Impact Analysis

## Manual

- Tedious process that requires comparison between multiple file types and environments
- App Designer provides compare reports for only metadata objects
- External file objects (SQRs, COBOLs, etc.) not included in App Designer compare reports
- Mapping technical changes to functional business processes based on vendor documentation

## Automated

- Impact analysis for all relevant objects (metadata and external files) in a consolidated view
- “Three-way diff-merge” provides visibility across multiple environments
- Automatically mapping technical changes to functional business processes



# #5: Separation of Duties

Manual	Automated
<ul style="list-style-type: none"> <li>■ Separation may be documented, but not enforced</li> <li>■ Each environment maintains separate access control</li> <li>■ Access control changes (new hires, terminations, etc.) must be updated for each environment</li> <li>■ SOX compliance is difficult to maintain</li> </ul>	<ul style="list-style-type: none"> <li>■ Secure application enforces who can make changes in each application environment, who approves changes, and who migrates changes</li> <li>■ No overlap between duties</li> </ul>


# #6: Systematic Testing

Manual	Automated
<ul style="list-style-type: none"> <li>■ Takes business users away from their "day jobs" and requires them to test the same business processes time and again</li> <li>■ Failed business processes can be missed due to human error</li> <li>■ Test frequency is limited by resource availability</li> </ul>	<ul style="list-style-type: none"> <li>■ Tests can be recorded once, and run multiple times</li> <li>■ Automated testing accurately captures error messages, pop-up messages, etc. that may be missed in manual testing</li> <li>■ Over time, a robust test bed can be created that tests multiple data sets and variables</li> <li>■ Consistent testing quality across different testing cycles and resources</li> </ul>

# #7: Documentation & Auditing

Manual	Automated
<ul style="list-style-type: none"> <li>■ Documenting changes is done via email, Word documents— is inconsistent and non-compliant with SOX mandates</li> <li>■ Document version control is usually using a naming convention for documents saved to a server.</li> <li>■ Establishing a change log during an audit means cobbling together database logs. This does not capture <u>why</u> the change was made</li> </ul>	<ul style="list-style-type: none"> <li>■ All documentation gathered and versioned in a central repository</li> <li>■ Audit log tracks what changes were made to the application, when the change was made, who made the change, who approved it, who migrated it to production, etc.</li> </ul>

*PeopleSoft Object Change History*  
*PRD Environment*



Object		Object Type			
Version	Date	User	SR	Job	Reason
PTDTWRK.CBL		Cobol File			
1.1	6/24/2004	jbrown	SR125	CS Enhancement	enhancement for Customer Service
1.2	7/15/2004	jbrown	SR384	Cust Entry Bug	fixed bug in Customer Entry screen
1.3	8/31/2004	tcollins	SR732	Patch 12345	Apply Patch 12345
PTPSQLGS.CBL		Cobol File			
1.1	8/31/2004	tcollins	SR732	Patch 12345	Apply Patch 12345
SYSOPTZN.SQC		SQR File			
1.1	7/15/2004	jbrown	SR384	Cust Entry Bug	fixed bug in Customer Entry screen
1.2	8/31/2004	tcollins	SR732	Patch 12345	Apply Patch 12345
DBENCODE.SQC		SQR File			
1.1	6/24/2004	jbrown	SR125	CS Enhancement	enhancement for Customer Service
1.2	7/15/2004	jbrown	SR384	Cust Entry Bug	fixed bug in Customer Entry screen

# #8: Restore Process

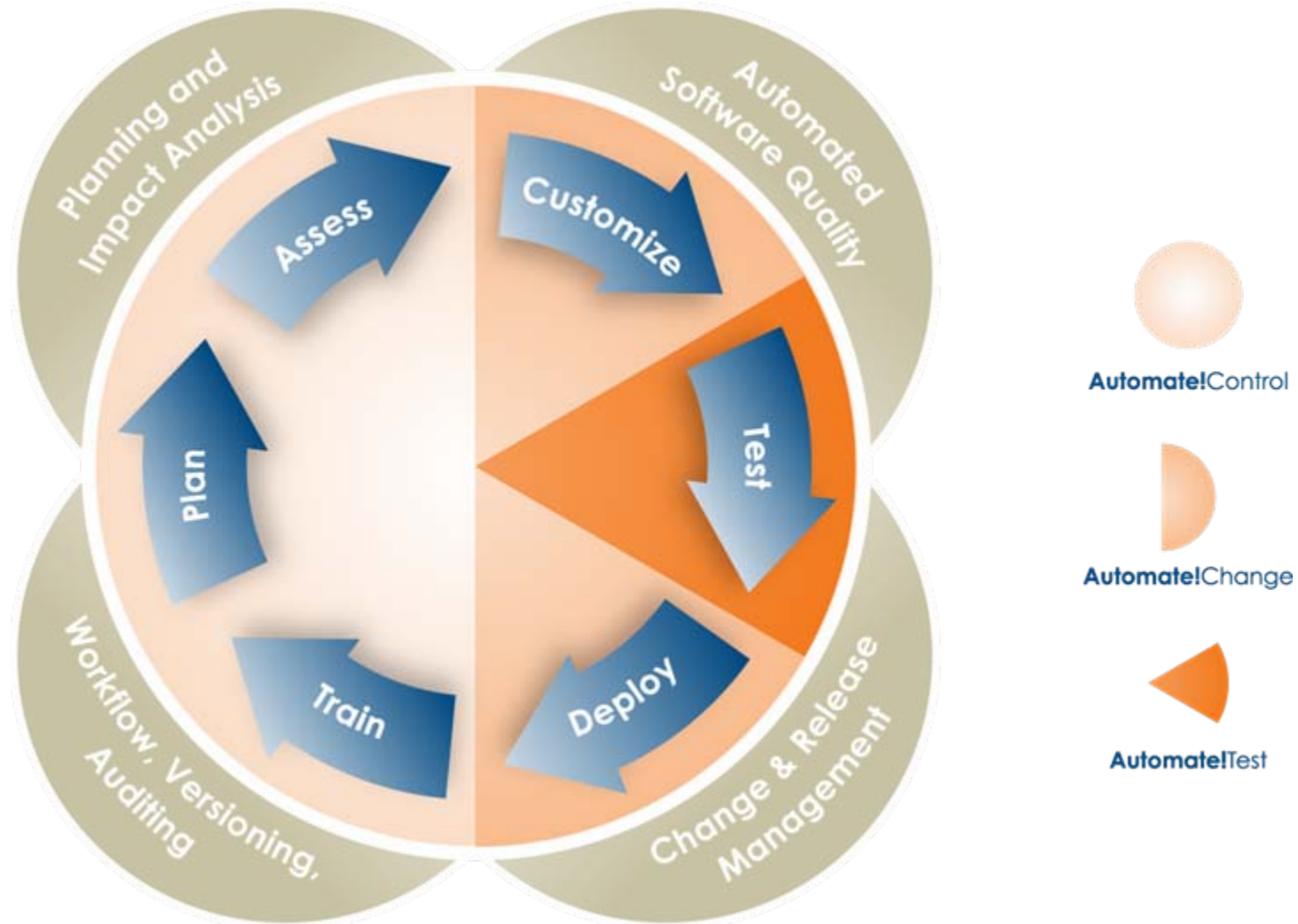
Manual	Automated
<ul style="list-style-type: none"> <li>■ DBA “cuts” a copy of the current database prior to applying a change</li> <li>■ Upon script or process failure the DBA “restores” the old database by replacing the updated copy with the old copy</li> <li>■ All changes made between the start of the process and the failure point are lost and must be re-performed</li> </ul>	<ul style="list-style-type: none"> <li>■ Allows users to “snapshot” a pre-change version of every file and object</li> <li>■ Restoring to previous state is achieved with the click of a button</li> </ul>



# 4 Easy Ways to Gain Control Over Vendor-Driven Changes

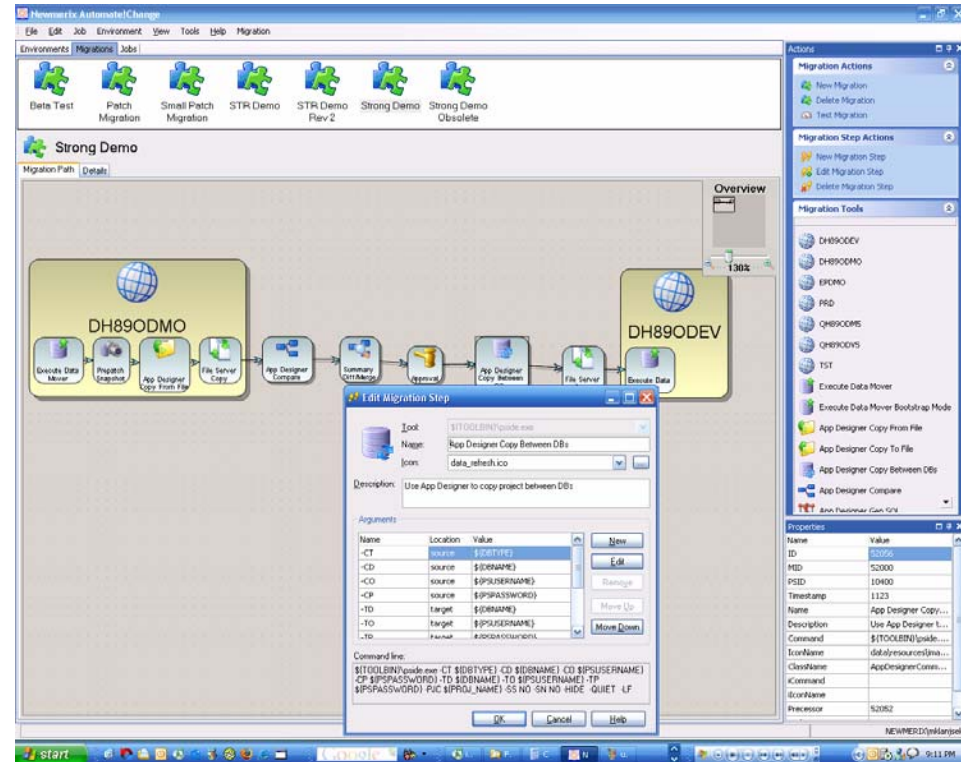
1. Standardization
2. Control Points
3. Automation
4. Visibility

# Only Newmerix Covers the Lifecycle of Change



# Standardization

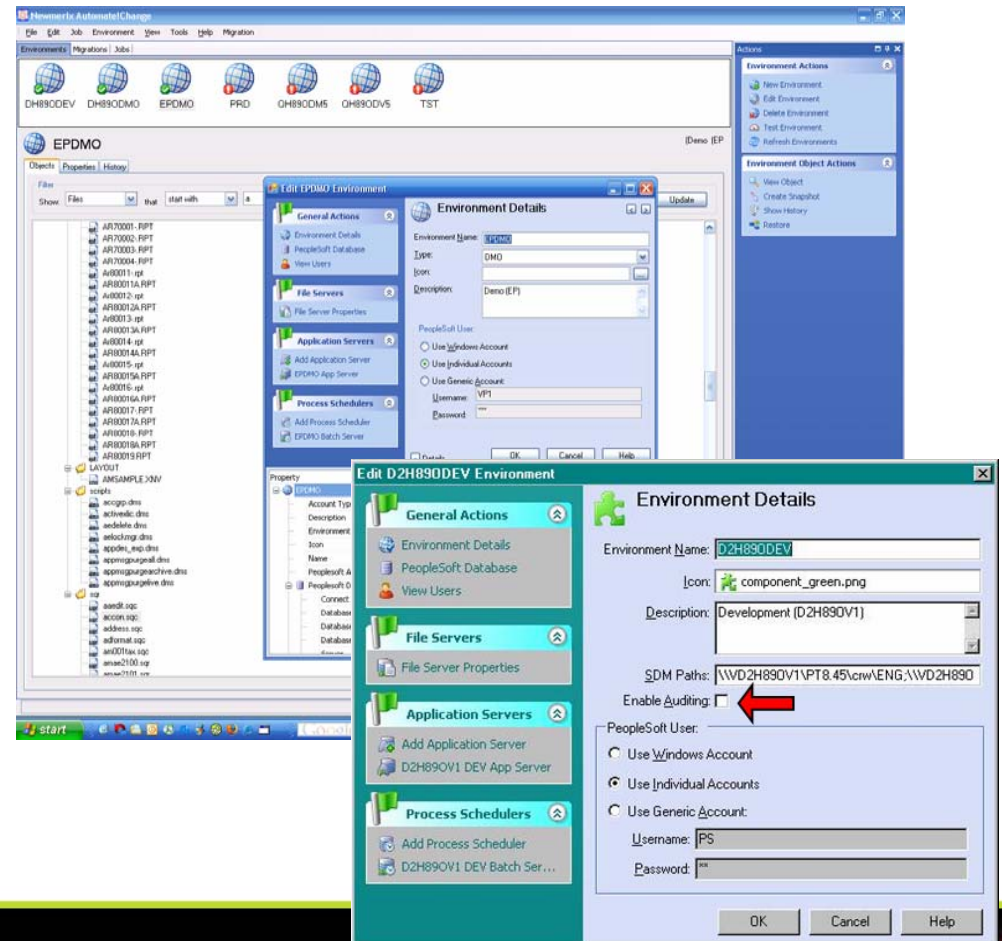
- Visually design your change management processes.
- Design different processes for patches, customizations, or emergency fixes.
- Automatically recognize which objects are in a patch/project and which steps need to be run.
- Integrated with AppDesigner so work done in AppDesigner is recognized.
- Handles both file and metadata objects.



Automate!Change lets you design standard change processes

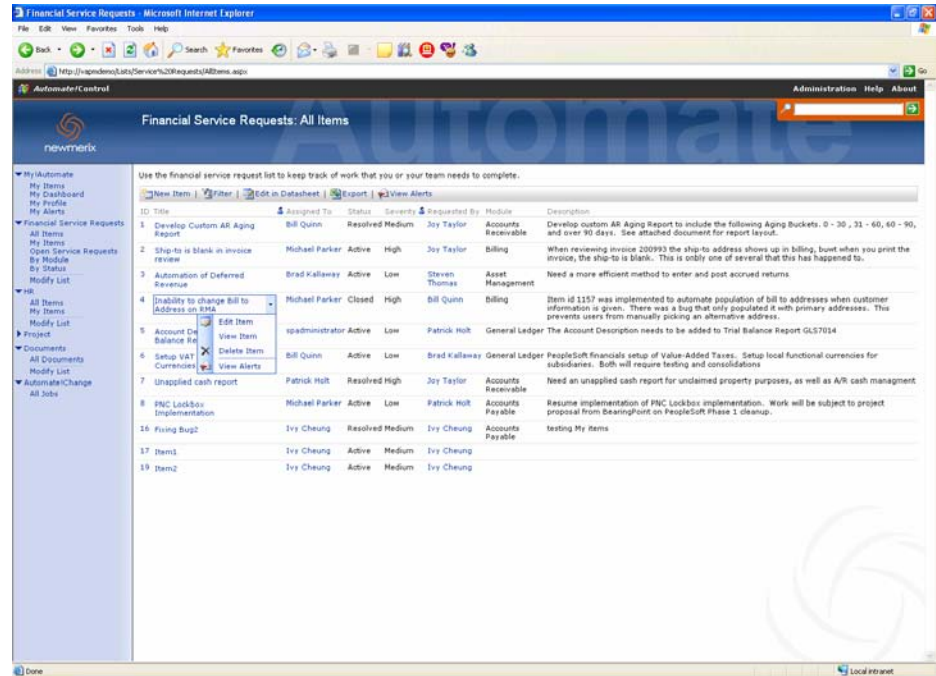
# Control Points

- Each PeopleSoft environment can be configured to automatically enforce separation of duties.
- Each PeopleSoft environment can be configured to automatically audit any file or metadata changes made to it.



# Control Points

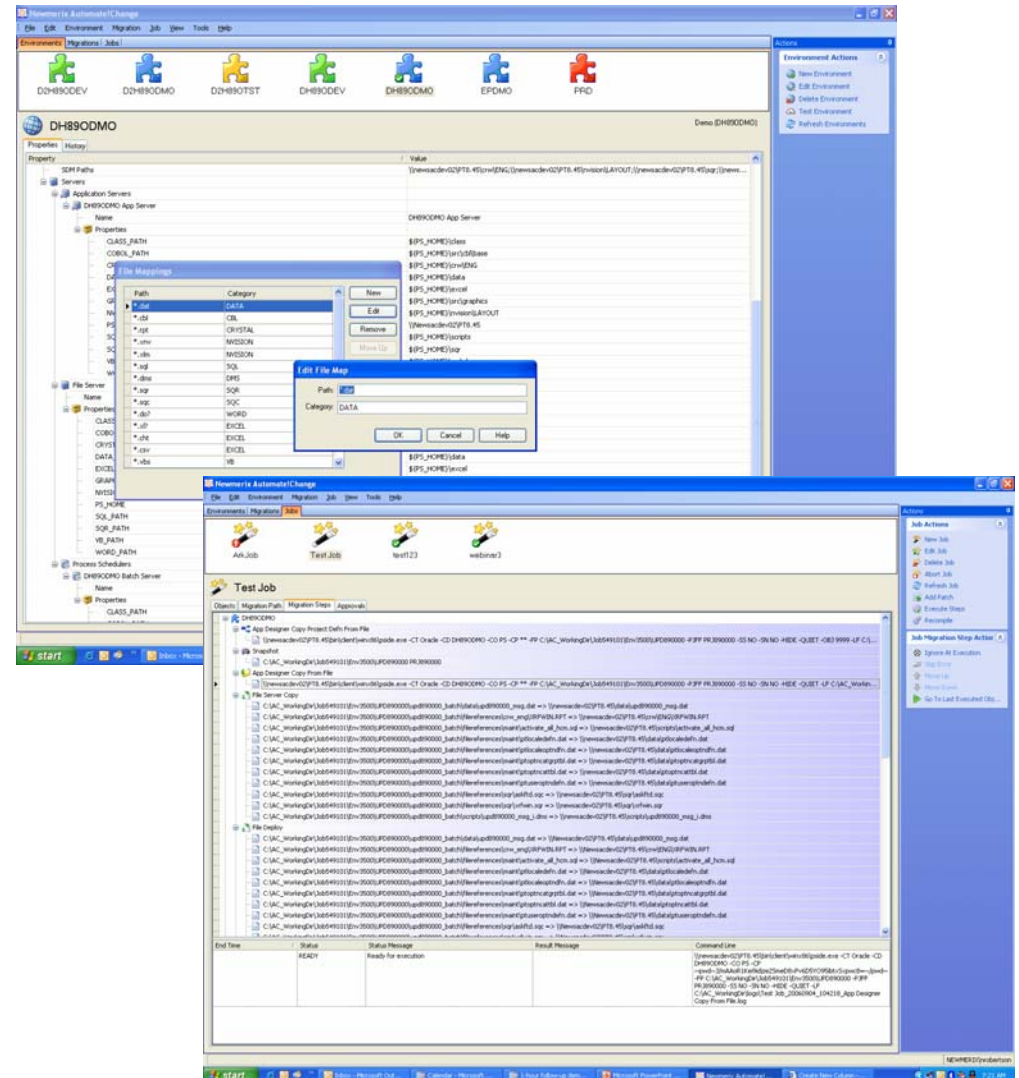
- Gather all service requests in Automate!Control.
- Design approvals directly into standard migrations.
- Approve via the web (Automate!Control) or desktop (Automate!Change).
- Manage all test plans, test cases and test results in one place.
- Complete document versioning system built in with integration into Excel and Word.



Automate!Control lets you collect authenticated approvals from anyone over the web or through Automate!Change.

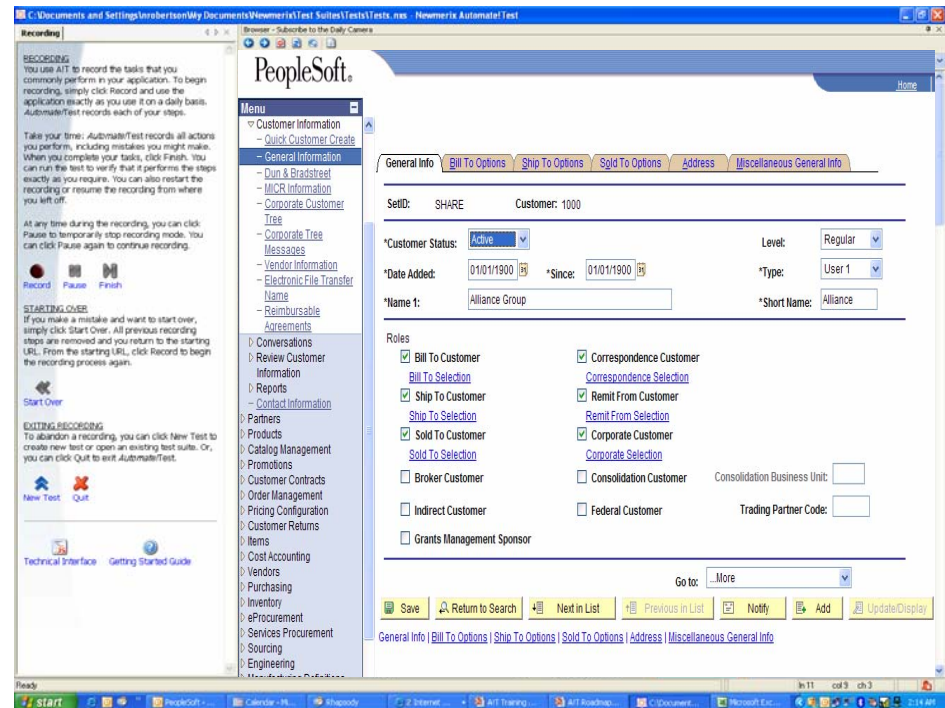
# Automation

- Automatically deploy PeopleSoft patches or project files.
- Automatically snapshot files and metadata before moving a patch/project to a new environment.
- Automate the process of deploying files to file servers, process schedulers and application servers.



# Automation

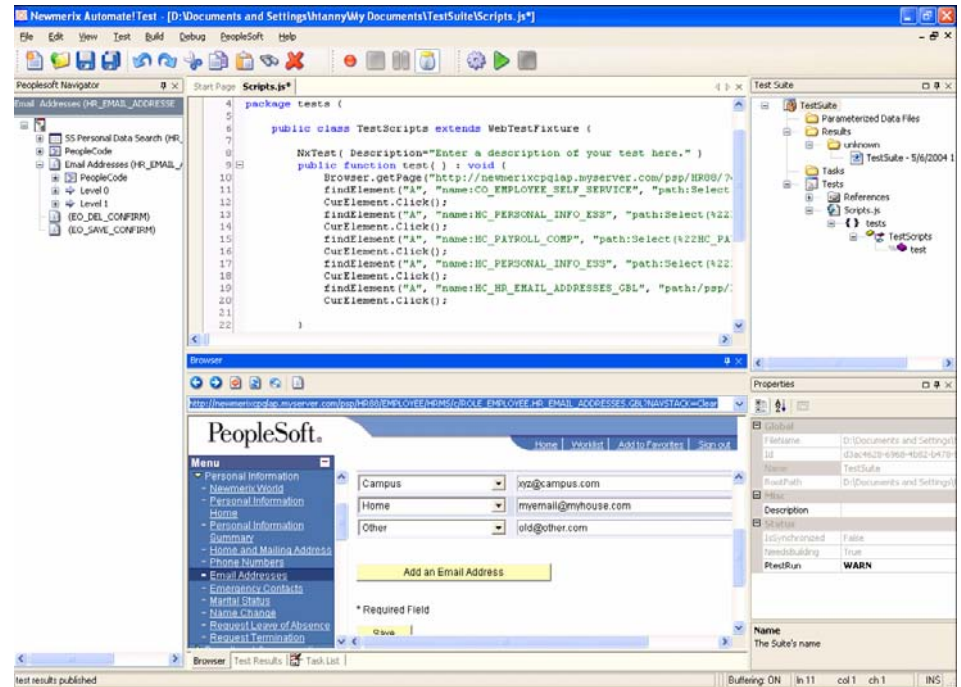
- Automated testing helps decrease the time required by your functional users, increases the breadth and frequency of testing and can be used by auditors!
- Use the functional interface to capture your end user knowledge about business processes.
- Automate!Test uses metadata to understand the changes in your PeopleSoft environment and how they affect your test scripts.



The Functional User Interface records what Your users do in PeopleSoft.

# Automation

- Automate!Test uses metadata to understand the changes in your PeopleSoft environment and how they affect your test cases.
- Metadata features can pinpoint expected failures even in test suites not related to the change.

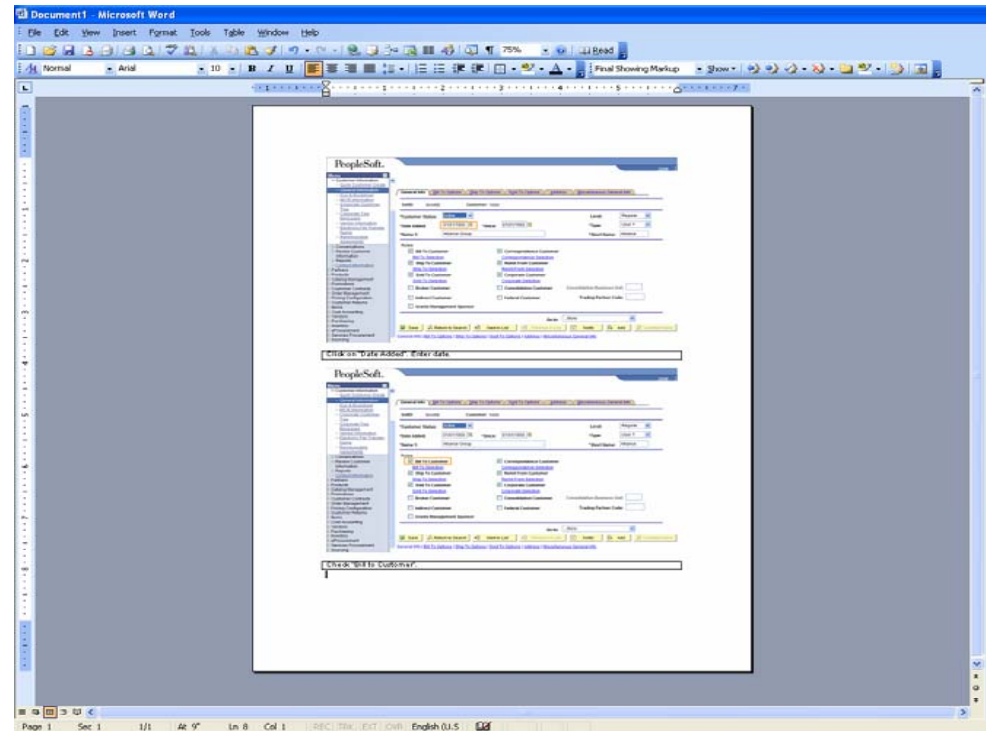


Automate!Test lets you automate the testing of standard business processes. Automate!Test watches changes in metadata to alert you of any changes needed to your test suites.



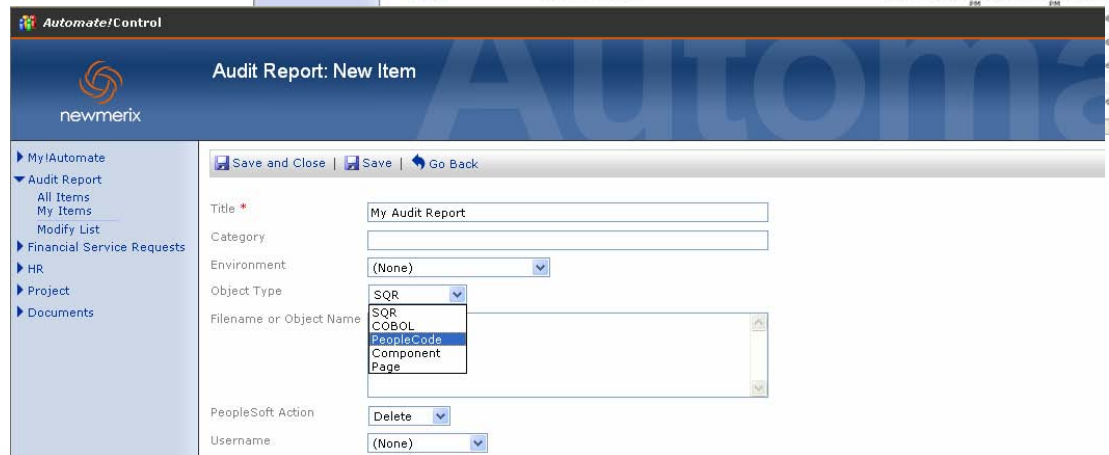
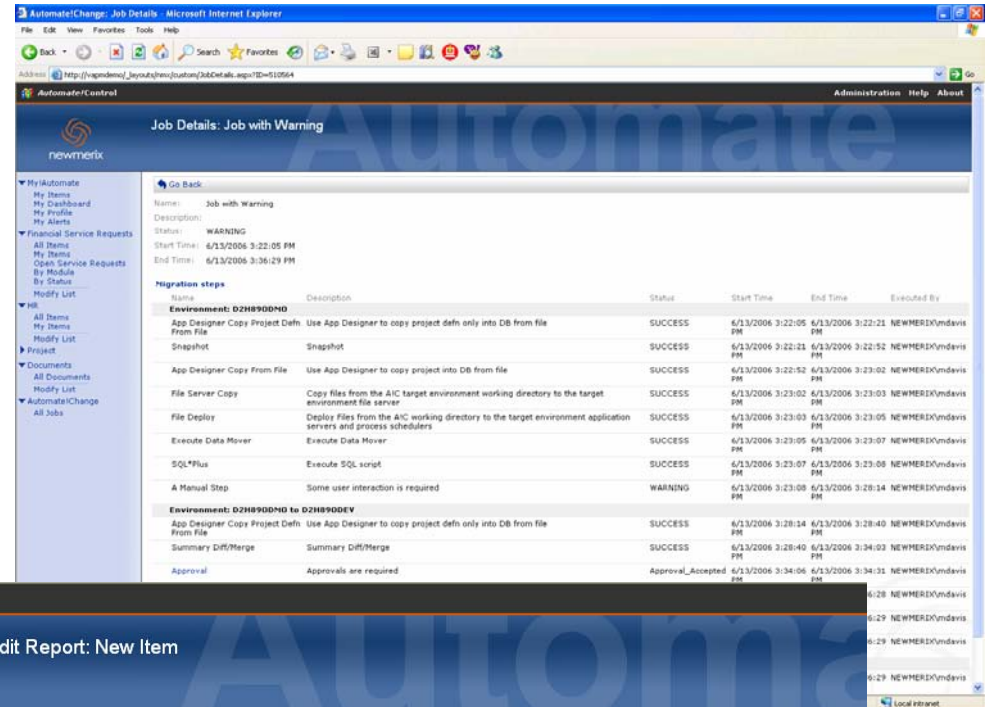
# Automation

- Generate end user documentation directly from Automate!Test using the business process documentation features.
- Publish this documentation (Word format) centrally using Automate!Control.
- Works with functional interface or standard interface.



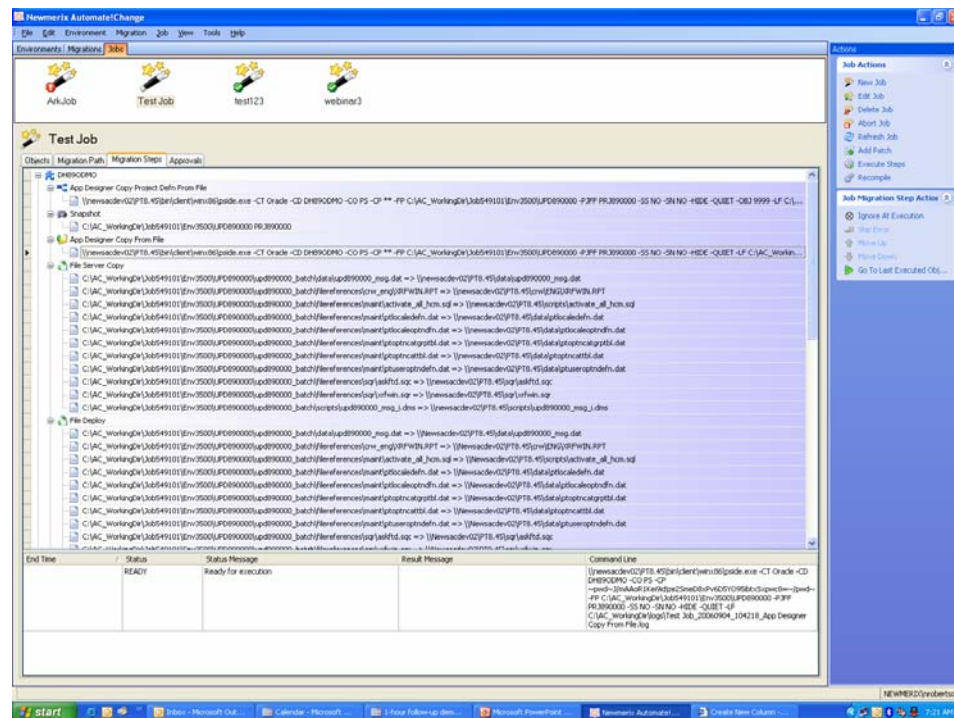
# Visibility

- Total visibility into Automate!Change.
- Approve from the web.
- Generate change management audit reports from the web.
- Central repository for Automate!Test results.



# Visibility

- Detailed recording of every change management action taken.
- All user information recorded for each action (who did what and when).
- Historical data (jobs, actions, migration steps) can be saved for as long as you like.



# Newmerix Customers



Government of Alberta

**NEC**

**TOMORROW NOW**

**HERCULES**



randstad



**KAISER PERMANENTE**



# Q&A Session

For questions or a copy of the presentation:

[info@newmerix.com](mailto:info@newmerix.com)

**(Refer to “COLLABORATE - PeopleSoft Patches,  
Service Packs, and Bundles”)**

To schedule a demo:

**Newmerix Sales ([sales@newmerix.com](mailto:sales@newmerix.com))**

**(303) 350-3900**

Additional Resources:

**[www.newmerix.com](http://www.newmerix.com)**

# PeopleSoft Patches, Service Packs, and Bundles

A Maintenance Strategy for Technical and Functional Teams