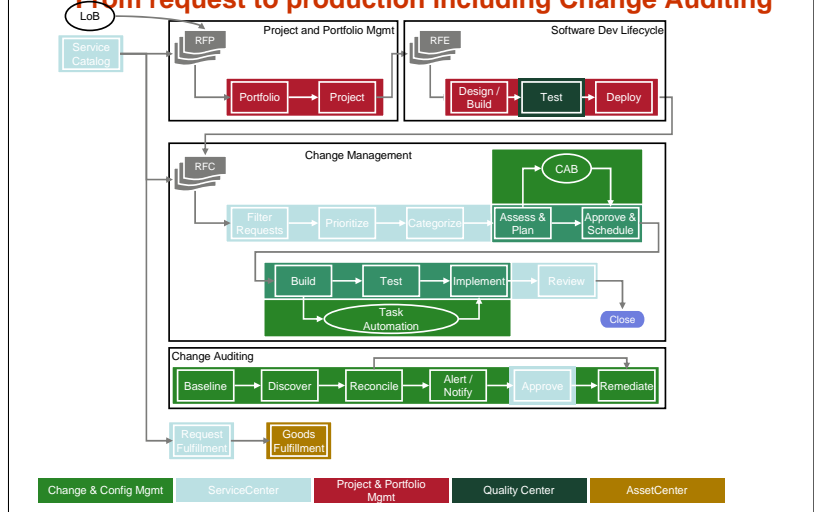


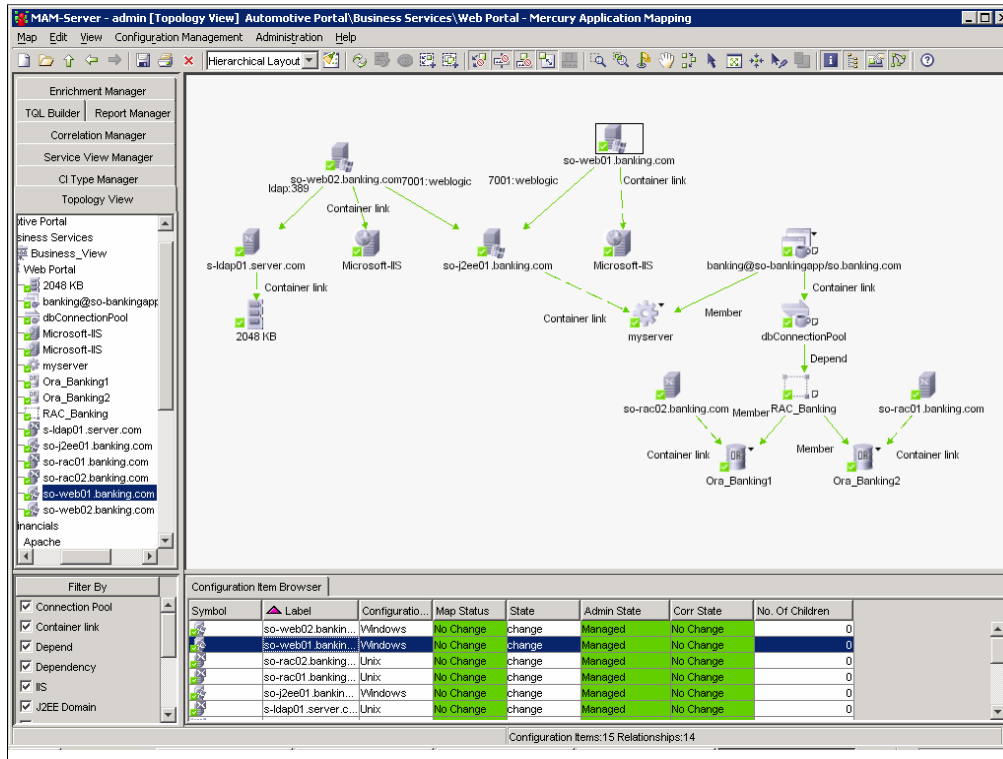
Taking Change Management From Firefighting to Fire Prevention

Webinar Part 2

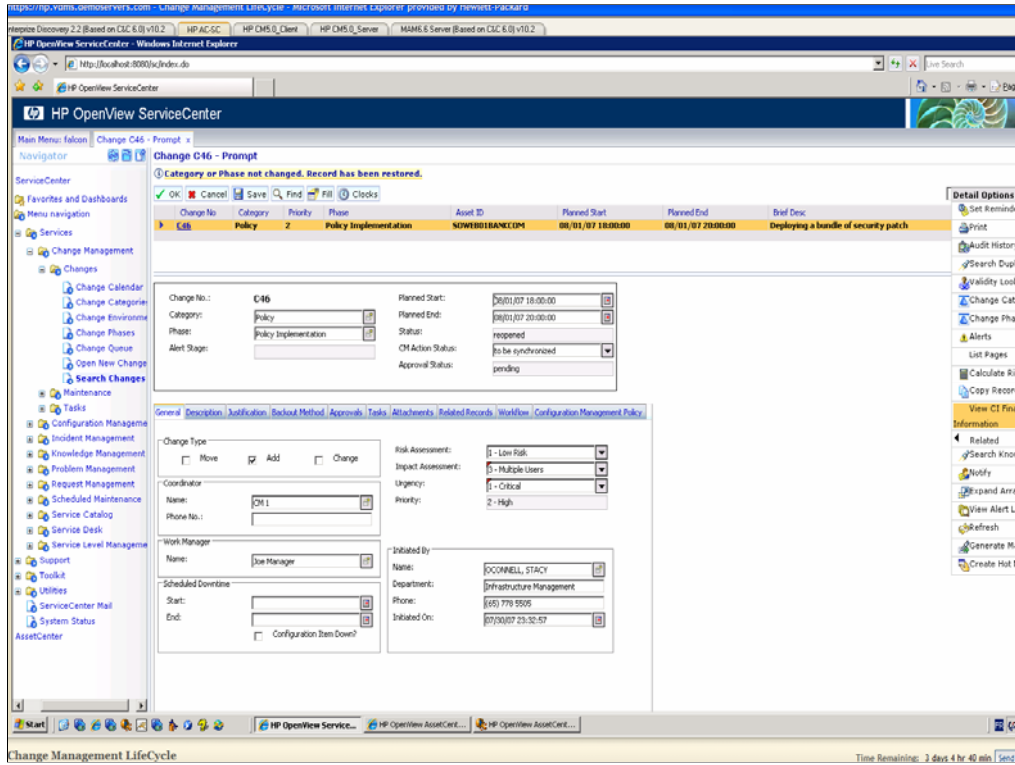
The change management lifecycle

From request to production including Change Auditing

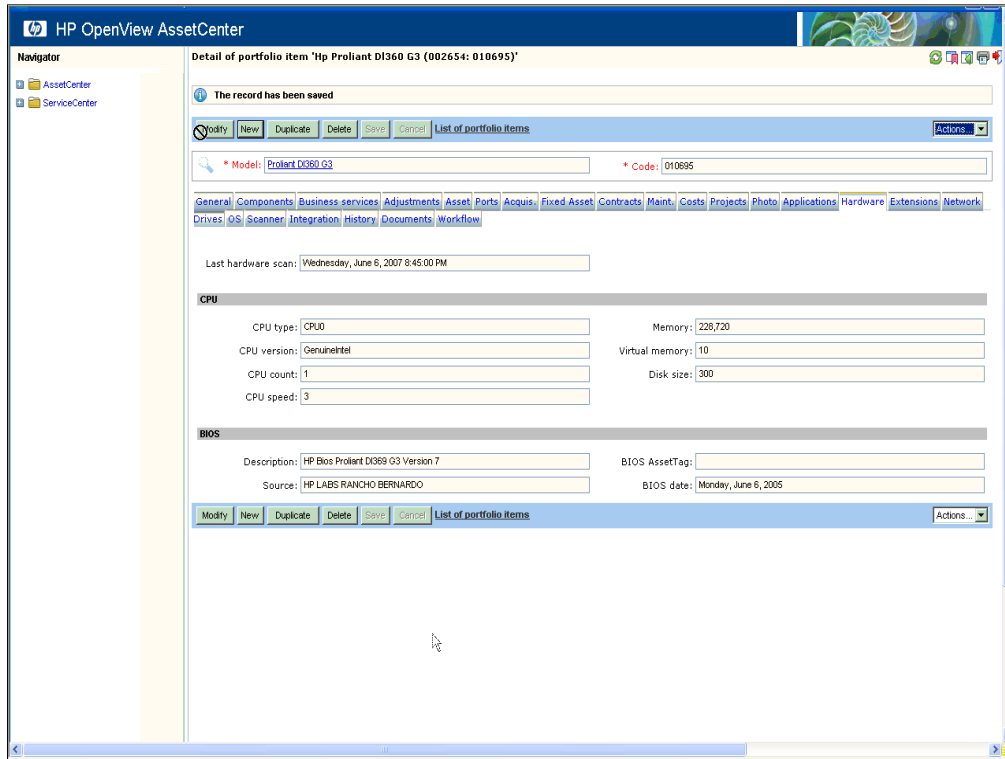




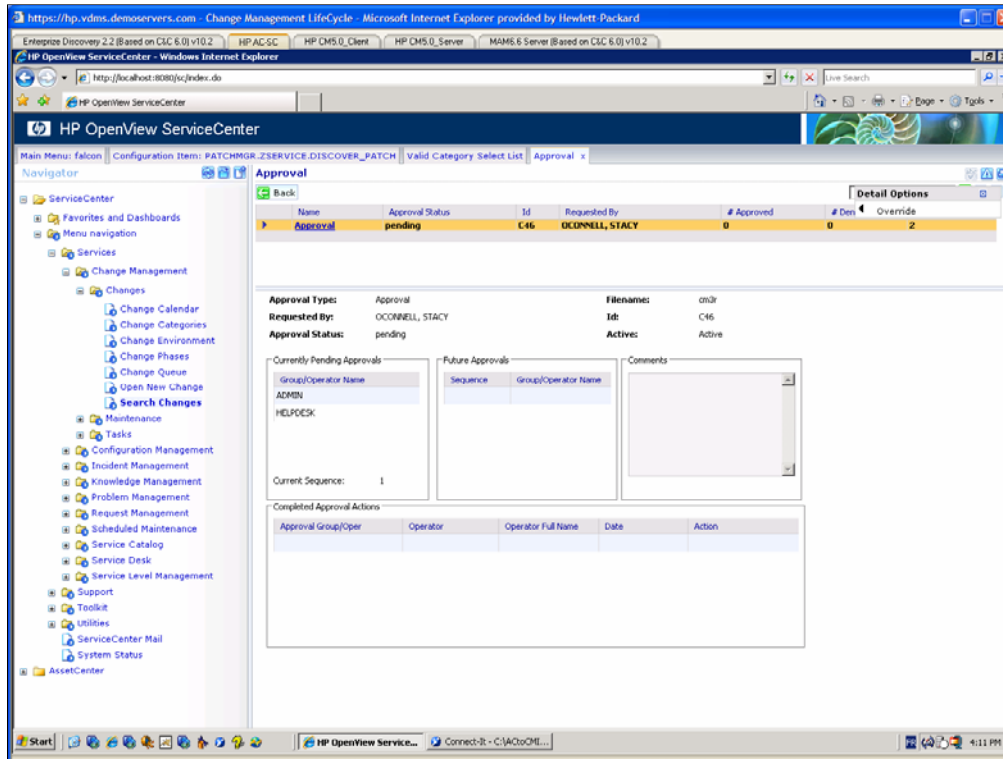
The first step is understanding what Configuration Items (CIs) are in the environment and what are the relationships and dependencies between CIs and services. Here we see that the online banking application service has the following CI dependencies. We need to apply the DST patch, therefore we can see that the server so-web01.bankin.com is the affected CI for this change request. This information will also be used as the basis for impact analysis which we will see later on in the demo as part of Change Control Management (CCM).



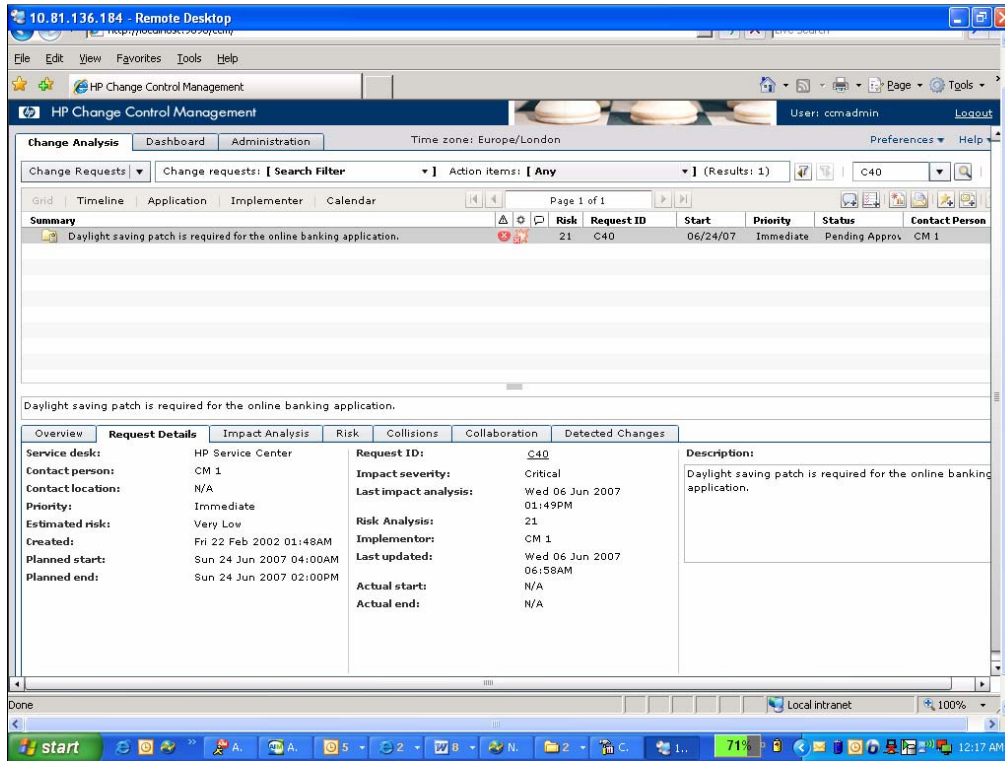
SC is also tightly integrated with AssetCenter in case the service desk user or change manager needs to review any financial information related to the configuration item (CI) before a change is approved for release.



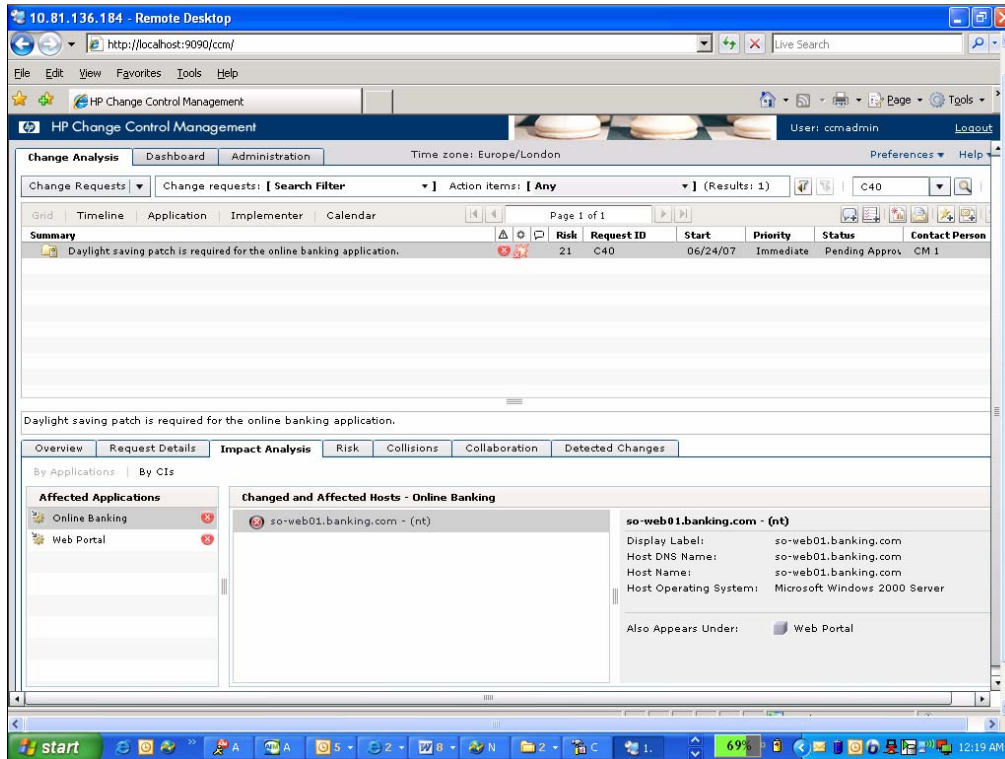
The following screen shot shows the attributes and details of the server, along with other asset and financial data that might be needed as part of the change process. Enterprise Discovery is integrated with AC to provide the listed CI attributes. Unrelated to this specific story, this could come in handy when evaluating the number of available licenses to deploy software (e.g. marketing dept needs MS Project...the request comes into SC and you need to verify how many licenses are used—Enterprise Discovery would provide this information—with the number of purchased licenses in AC).



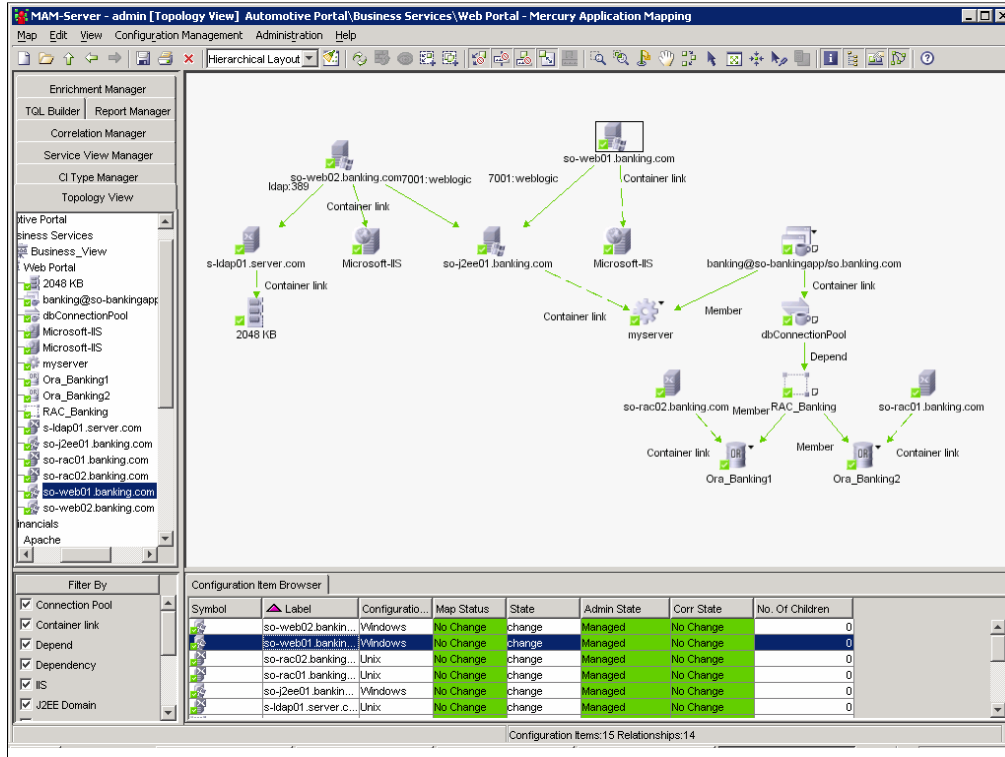
This screen in SC shows the multiple levels of approval that are needed before moving a change into release. This is important to show the difference between SC and CCM, as there are multiple approvals both on the IT and the LOB side that need to happen which SC captures and tracks. The CAB is just one of the approvals that needs to occur, and CCM is the automated tool that provides decision-support for the CAB.



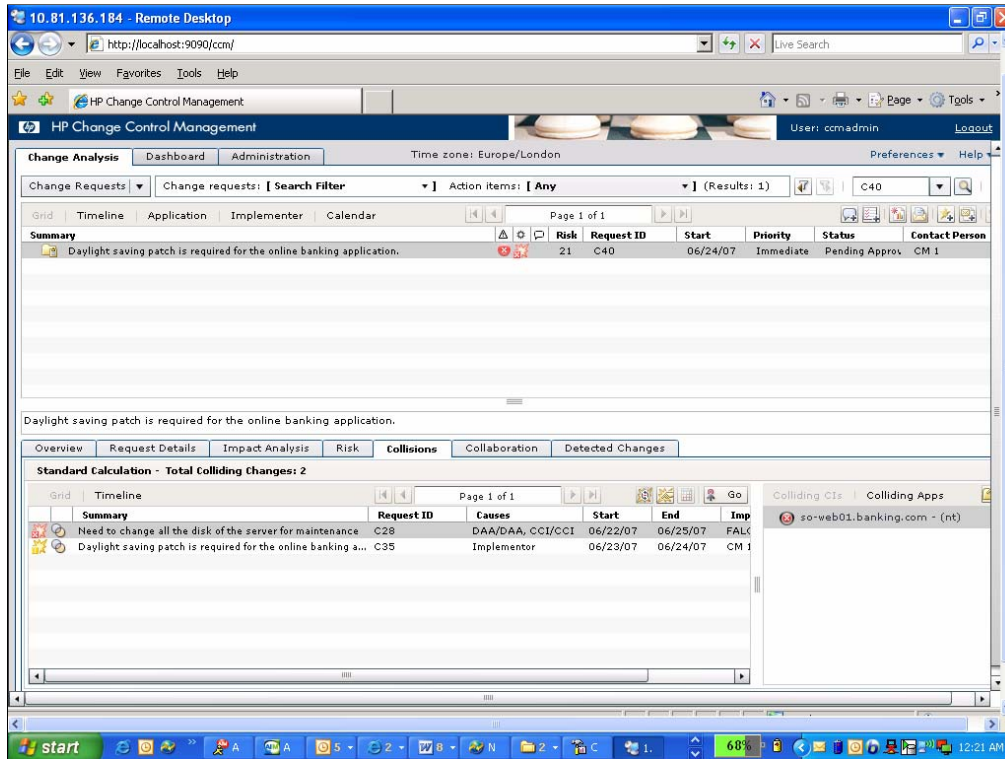
CCM provides visibility by presenting a global view of change demand, bringing in change requests from multiple, traditionally siloed IT teams (strategic, application, and operational changes from multiple service desks) into a consistent view so the CAB can make better decisions. This screen shot of CCM shows the DST patch change request information being imported from ServiceCenter, including all of the relevant details and description related to the change.



Part of the value of CCM is automating impact analysis—a traditionally manual process for the change manager and the CAB. CCM is integrated with the Universal CMDB and Application Mapping to the CAB can get an objective view of impact based upon the actual production environment. It's not just about the impacted CI, but all of the other components and services that are directly or indirectly affected/impacted by the change. Here we can see that the server/CI not only touches the online banking application but also the web portal, both customer-facing/revenue generating services. So if something were to go wrong with the change, it could directly affect the company's bottom line, indicating the impact and risk level are actually high (not low as the change ticket suggests), and require further analysis by the CAB.

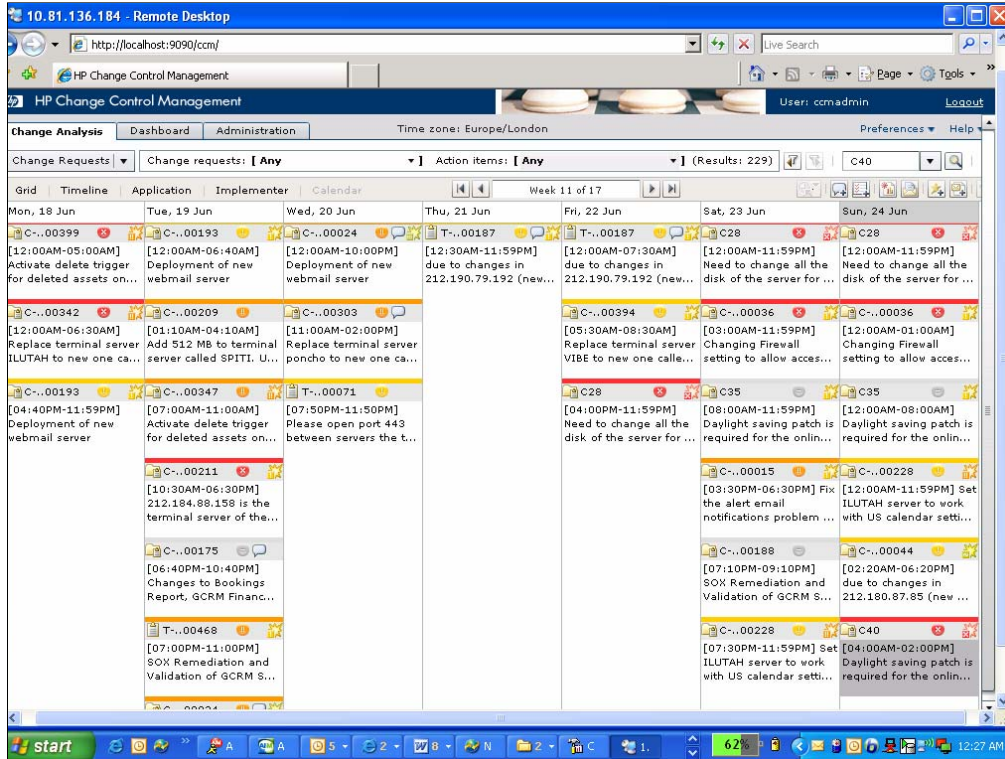


The first step is understanding what Configuration Items (CIs) are in the environment and what are the relationships and dependencies between CIs and services. Here we see that the online banking application service has the following CI dependencies. We need to apply the DST patch, therefore we can see that the server so-web01.bankin.com is the affected CI for this change request. This information will also be used as the basis for impact analysis which we will see later on in the demo as part of Change Control Management (CCM).

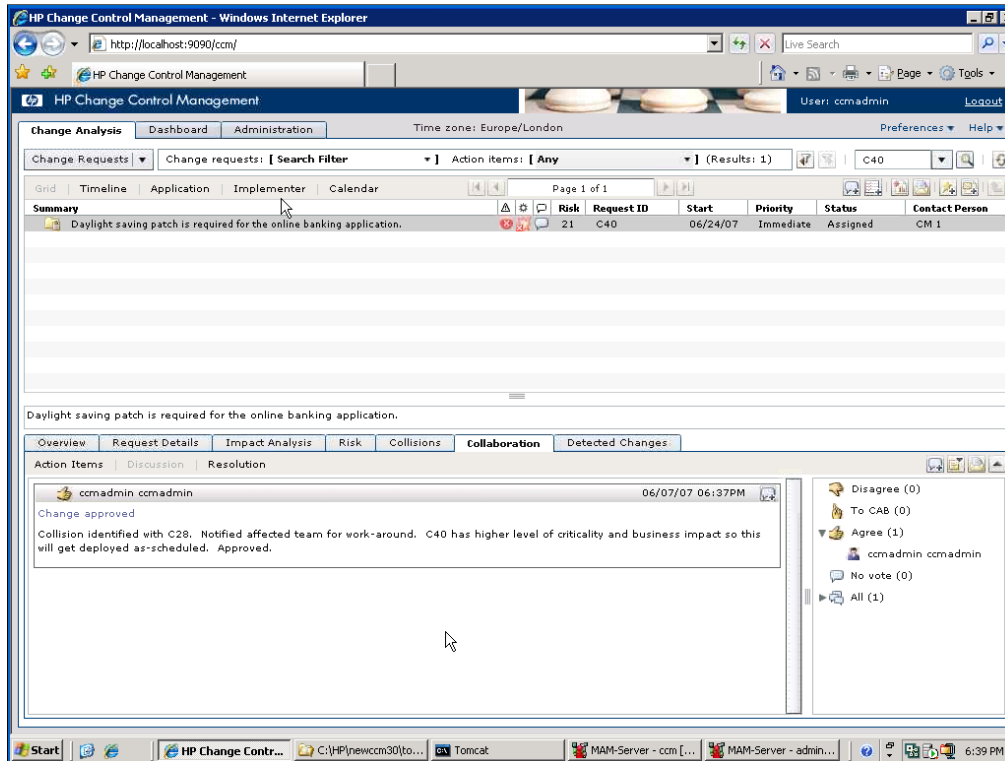


Here we see the change collisions which may potentially cause a service disruption if not detected prior to production. Without a tool like CCM to help automate this process, it's extremely difficult for the CAB to identify collisions prior to deployment. Some collisions are easier to detect (e.g. two change requests are scheduled to work on the same CI at the same time), but others are more difficult (e.g. a network router goes down for maintenance but it's sitting between the server and the team doing an upgrade while causes a failure), or they are resource based (e.g. Joe is scheduled to perform 2 changes at the same time).

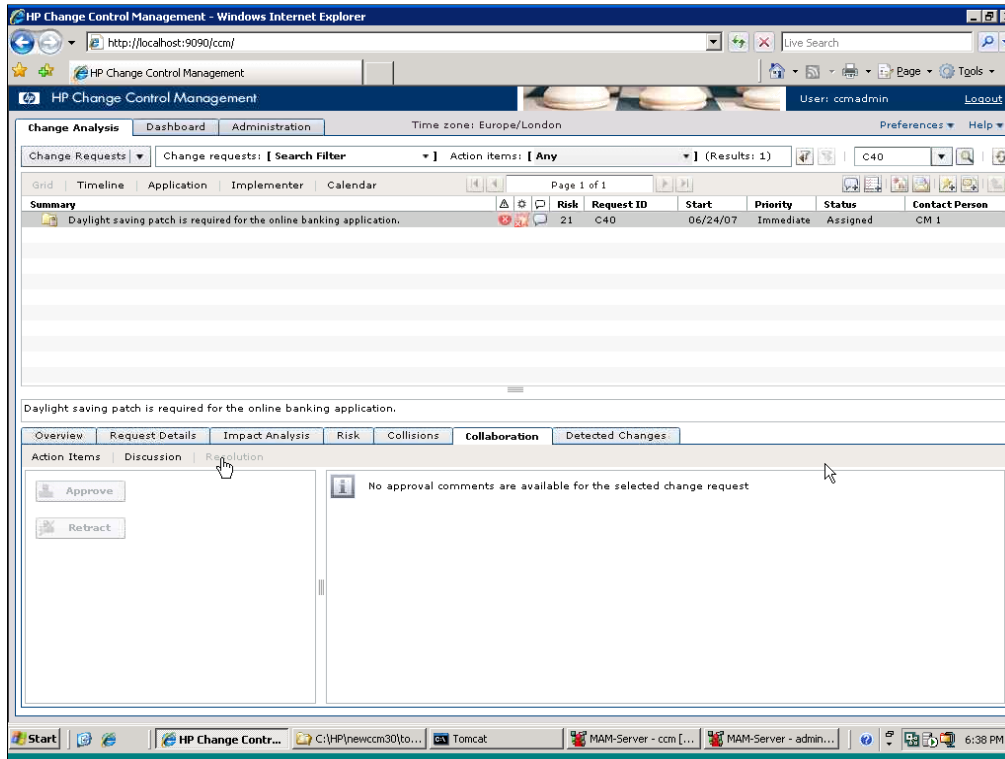
In this example, we see that there is actually a collision with the DST patch change request as there is another change request which is scheduled to work on the same CI at the same time. Let's look into this further....



Here is a screen shot of the Forward Schedule of Change. Unlike a simple FSC in a service desk, which only shows a simple calendar view, the FSC in CCM is shown based upon multiple views (here we see it in an Outlook style form, but you can also see it in a timeline view, by application/service, and by implementer). Here, for each change we can easily see the impact (based upon the color coding), and if there are collisions and/or comments related to the change, making it very easy for the change manager/CAB to filter out the high risk or negative changes to discuss during the CAB meeting.



One of the additional benefits of CCM is the collaboration. Within CCM you can assign and track action items, capture all of the discussion threads, and vote on a change all within the tool. Some customers are using this to get to a completely Virtual CAB so they don't even have to meet in person anymore. Here we can see that since a collision was identified, the impacted teams were notified and a work around was scheduled. Since the DST patch is critical to deploy before at the original time and the business impact is greater, the CAB decided the other change should be re-scheduled, so this change is ready to be approved and moved into release.



One of the new integrations with CCM 3.0 (GA in June 2007) is the ability to approve changes in CCM and this information is automatically updated within both SC and SD to keep the change process moving. Now the change manager doesn't have to toggle between tools but can do all of the impact analysis and approval notification within CCM.

The screenshot displays the HP OpenView ServiceCenter interface. The main window title is "10.81.136.182 - Remote Desktop" and the application title is "HP OpenView ServiceCenter". The interface includes a "Navigator" on the left with a tree view of service categories such as "Change Management", "Maintenance", "Tasks", "Configuration Management", "Incident Management", "Knowledge Management", "Problem Management", "Request Management", "Scheduled Maintenance", "Service Catalog", "Service Desk", "Service Level Management", "Support", "Toolkit", "Utilities", and "ServiceCenter Mail".

The main content area shows an "Approval" page. At the top, there is a "Back" button and a "List of Valid Parent Changes" link. Below this is a table with the following data:

Name	Approval Status	Id	Requested By	# Approved	# Denied
Approval	approved	C40	O'CONNELL, STACY	2	0

Below the table, there are several fields and sections:

- Approval Type:** Approval
- Requested By:** O'CONNELL, STACY
- Approval Status:** approved
- Filename:** cm3r
- Id:** C40
- Active:** Active

There are three main sections for approvals:

- Currently Pending Approvals:** A table with columns "Group/Operator Name" and "Sequence".
- Future Approvals:** A table with columns "Sequence" and "Group/Operator Name".
- Comments:** A text area with a "No" icon.

At the bottom, there is a "Completed Approval Actions" table:

Approval Group/Oper	Operator	Operator Full Name	Date	Action
ADMIN	falcon	Jennifer Falcon	06/06/07 23:23:32	Approved
HELPDESK	falcon	Jennifer Falcon	06/06/07 23:23:32	Approved

Now that the CAB has approved the change, we can see here in ServiceCenter that all of the necessary approvals have been obtained and this change is ready to move into release. At this point the management action triggers Configuration Management to deploy the patch.

HP CM Reporting Server

Current Reporting View: Compliance by Devices

Search Options

Directory/Group Filters

HPCCMSRV

Data Filters

- Inventory Management Related
- Patch Management Related
- Usage Management Related

Display Options

Reporting Views

- Baseline Auditor Reports
- Inventory Management Reports
- Patch Management Reports
- Usage Management Reports
- Server Management Reports

Search Criteria: None

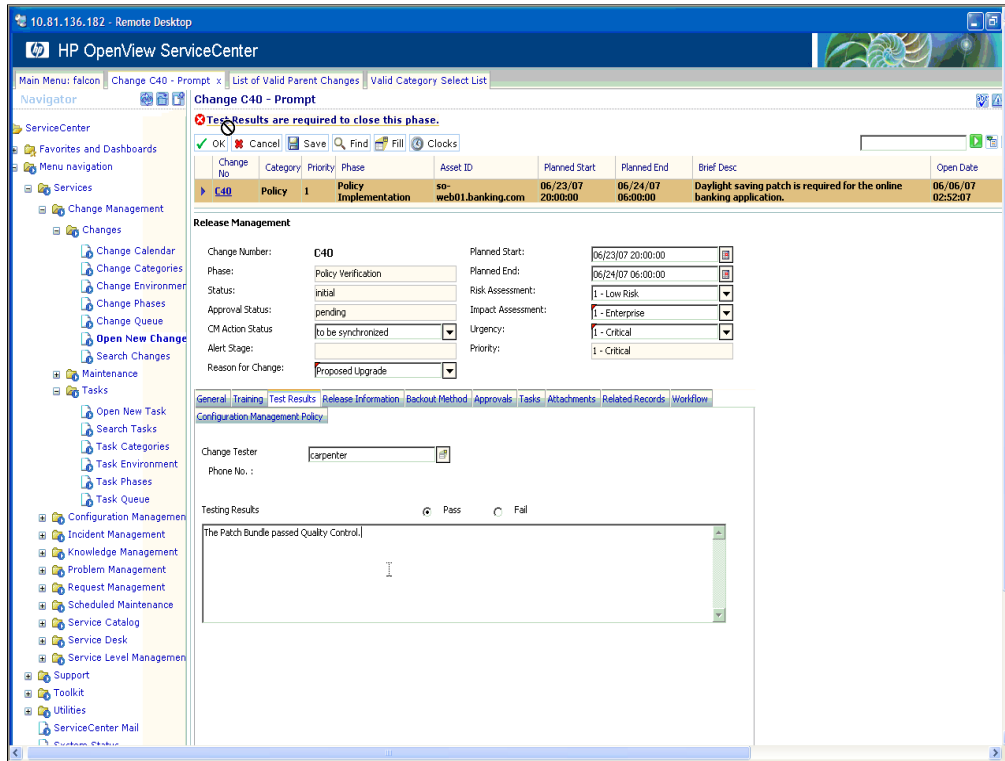
Compliance by Devices

15 items | 1 - 12 of 12 items

Details	Status	Device	Last Scanned	Applicable Products	Applicable Bulletins	Patched	Warning	Not Patched	Other	Reboot Pending	Total
	✘	SOWEB01BANKCOM	2007-06-06 11:57:40	3	34	3	0	31	0	0	34
	✔	SOWEB10BANKCOM	2007-06-06 09:06:42	3	34	34	0	0	0	0	34
	✔	SOWEB09BANKCOM	2007-06-06 00:55:15	3	34	34	0	0	0	0	34
	✔	SOWEB08BANKCOM	2007-06-05 23:33:47	3	34	34	0	0	0	0	34
	✔	SOWEB07BANKCOM	2007-06-05 23:04:02	3	34	34	0	0	0	0	34
	✔	SOWEB06BANKCOM	2007-06-05 22:35:44	3	34	34	0	0	0	0	34
	✔	SOWEB05BANKCOM	2007-06-05 22:07:32	3	34	34	0	0	0	0	34
	✔	SOWEB04BANKCOM	2007-06-05 21:41:45	3	34	34	0	0	0	0	34
	✔	SOWEB02BANKCOM	2007-06-05 20:04:26	3	34	34	0	0	0	0	34
	✔	SOWEB03BANKCOM	2007-06-05 18:42:52	3	34	34	0	0	0	0	34
	?	DEMOCLT0308	2007-05-19 16:03:37	0	0	0	0	0	0	0	0
	?	DEMOCLT0108	2007-05-16 10:32:52	0	0	0	0	0	0	0	0

[Return to Compliance by Devices](#) | [Return to Top of Page](#)

This is a compliance report summary from CM, so you can see on a per-device basis which machines do not have the necessary patch levels. Here we see that the server does not meet the compliance requirements around the DST patch. This can be run to quickly identify machines without the appropriate compliance levels vs a manual process.



Part of the CM/SC integration is validation from CM to SC that the change was successful. We see here that the change has been validated as complete, passed quality control, and now we are ready to close the ticket.