cih Solutions
SERVICE MANAGEMENT INTEGRATORS

Improving IT Change to drive Business Value

The third in a series of white papers from CIH Solutions that discuss topical issues in IT Service Management

Abstract

This white paper looks at how a number of improvements can be introduced into the Change Management process to drive greater business value by reducing both risk and cost to service delivery.

Mike Simpson
info@cihs.co.uk

This white paper looks at the relationship between Business Value and the IT Change Management process. The paper looks at a number of improvements that can be introduced into the Change Management process to increase efficiency with the aim of reducing both risk and cost to service delivery, and hence drive more Business Value. In addition the paper identifies four 'hot spots' based on the author's experience that outline common change management problems and suggests possible solutions.

Introduction

I think it is safe to assume that IT change management, as defined by ITIL v3, is well known to the majority of ITSM personnel. Therefore, in this paper it is not my intention to attempt a ground up design of a Change Management process, but rather to start with the premise that all IT organisations will by necessity be operating some form of IT change management.

In some IT organisations I've encountered the Change Management process is often viewed as just another administrative function. This often means it is not taken seriously, until a major incident occurs, of course. Furthermore, if the right checks and control points are not in place, or are too complex, then there will always be the temptation for personnel to work around the system to get things done quickly. This leads to a culture of 'fix a problem first, then do the paperwork later'. Of course, a lot of the time the paperwork fails to be completed so the system configuration never gets updated. But more on that later.

Adding value to the business

So, what exactly do I mean by adding business value? For me this is all about a continuous improvement in service delivery and the key to this is in the way we set up and operate the Change Management process within IT Service Management. This is fundamental to the success of service delivery. In my view the single biggest mistake is to view the Change Management process in isolation to the other key ITSM processes, namely Configuration Management, Incident Management and Problem Management.
Figure 1 below shows the interdependency between these three processes (a sort of IT circle of death) that can only be mitigated by the integration with Configuration Management. Or, more specifically, the Configuration Management Data Base (CMDB).

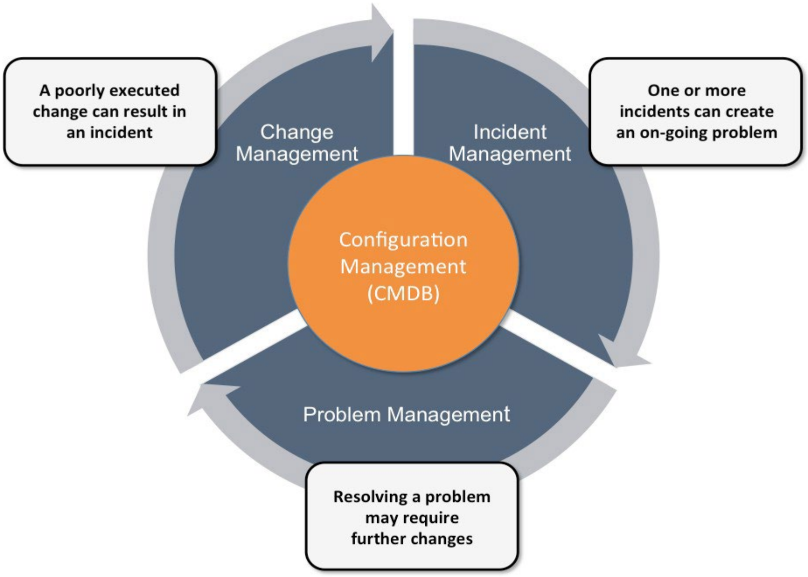

A poorly executed change can result in an incident
Change Management
Incident Management
One or more incidents can create an on-going problem
Configuration Management (CMDB)
Problem Management
Resolving a problem may require further changes

Figure 1 – Interdependency of ITIL Processes

Meaningful business value will only result once the Change Management process is truly integrated with the Configuration Management process and in particular the Configuration Management Data Base (CMDB). Without the integration of these two processes any Impact Analysis will be vulnerable to misinterpretation due to:

- an incomplete baseline for the current system configuration;
- insufficient understanding of dependencies and relationships between Configuration Items (CIs);
- unclear ownership of CIs;

In addition, the Incident Management and Problem Management processes must also be aligned to the Configuration Management process so that incidents and problems can be analysed in the context of any recent changes to the IT environment.

Typical Shortfalls in Change Management

As mentioned previously, in some IT organisations the Change Management process is often viewed as just another administration function, driven manually through a weekly Change Advisory Board (CAB) meeting. Even with the variety of ITSM tool sets that are available today, IT organisations appear reluctant to invest in workflow solutions that will bring together all the key ITSM processes, thus bringing benefits across the whole of service delivery.
Let’s start with a look at some typical outcomes of a poorly constructed Change Management process. Here are my top five:

1. Changes that are not authorised and never recorded;
2. Changes that are authorised and released and never recorded;
3. Changes that are authorised only after release;
4. Changes that are authorised without sufficient impact analysis;
5. Changes that are not completely tested before release.

Thoughts on Unauthorised Changes

First, one personal observation. In my experience the majority of IT personnel really do want to do the right thing when it comes to change management, but too often badly constructed processes create a barrier to getting things done. On the occasions when I have carried out an assessment of a client’s Change Management process, it was clear to me that IT staff who implemented unauthorised changes did so more out of frustration with the administration involved and not out of indifference to the principle of change management. After all, unauthorised changes will soon corrupt the CMDB and make it a lot harder to keep system configurations up-to-date.

I recall one interview with a server technician who gave the following reply to a standard question on unauthorised changes:

“I need to prepare for a server upgrade, but first I need to implement a minor change to do some pre-upgrade tests. What I find is that this small change gets bundled into a Regular change batch to be signed off by the weekly CAB, so for me the change management process is a barrier to be worked around.
So, knowing there is little or no risk I do the change anyway, so I can keep to my upgrade plan, but submit the change request at a later date to keep the records correct.”

I must admit to having some sympathy with this view. Clearly, this is a case of the Change Management process currently in place acting as a barrier. However, this approach raises a red flag when retrospective change requests get put to one side and then forgotten. The key worry here is not so much the lost change request, but the long term impact on the CMDB. The small change mentioned above could have an unforeseen impact on the associated CI(s).

Suggested Improvements

So, I'm going to outline my five top improvements that I believe will help streamline a Change Management process:

Improvement 1 - Switch the focus away from being CAB-centric and move to on-line approval;
Improvement 2 - Employ greater use of a CMDB as a way of driving accurate impact analysis;
Improvement 3 - Implement a change calendar to reduce change conflicts;
Improvement 4 - Implement a dynamic link between CIs and incidents;
Improvement 5 - Introduce a small set of metrics to monitor change management performance.

I'm then going to look at four 'hot spots' where poor change management can result in issues that impact the Configuration, Incident and Problem Management processes. My final 'hot spot' will look at potential issues around the Release and Deployment management process and how they can be mitigated.

Improvement No 1 - Move to On-line Approval

Figure 2 below shows a typical (but simplified) arrangement for a typical Change Management process. Although many ITSM tool set vendors these days have bundled offerings that include a Service Desk and a selection of workflows, for the purpose of this example I'm assuming that the Service Desk contains a simple change management workflow to drive what is essentially a manual process.

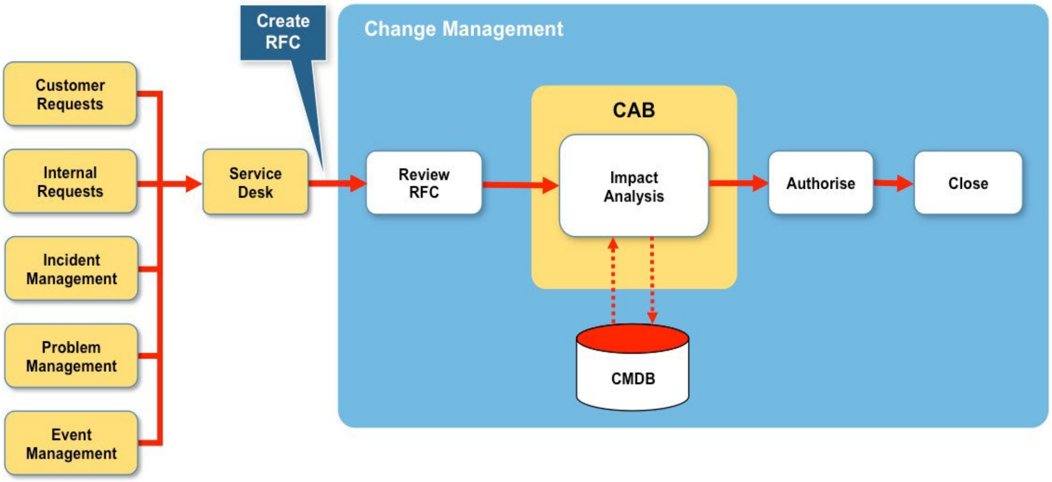


Figure 2 – Simple Change Management Process

The usual sequence is for a Request for Change (RFC) to be logged by the Service Desk and then reviewed by the change management team who will 'review' the RFC and determine who takes responsibility for both the assessment and impact analysis of the change. The change could be rejected at this point.

Although most Service Desk applications go some way to provide a workflow, it is still basically a semi-automated process that relies on a simple ticket routing to departments and supporting email notifications. The change management team will determine the priority of the change, based on a number of criteria, for example:

- the type of request
- the impact and urgency
- the requestor status - client or internal
- the service or system impacted

The change will then be scheduled for either a regular or emergency CAB. This is what I mean by CAB-centric. Note: It is assumed that as all the Standard changes (pre-approved) will processed directly by the Service Desk.

In order to streamline this process the focus must switch from a 'default' choice of a CAB forum and move to on-line collaboration using a change management workflow. There are a number of ITSM tool set applications available to create this workflow and Figure 3 shows one possible construct.

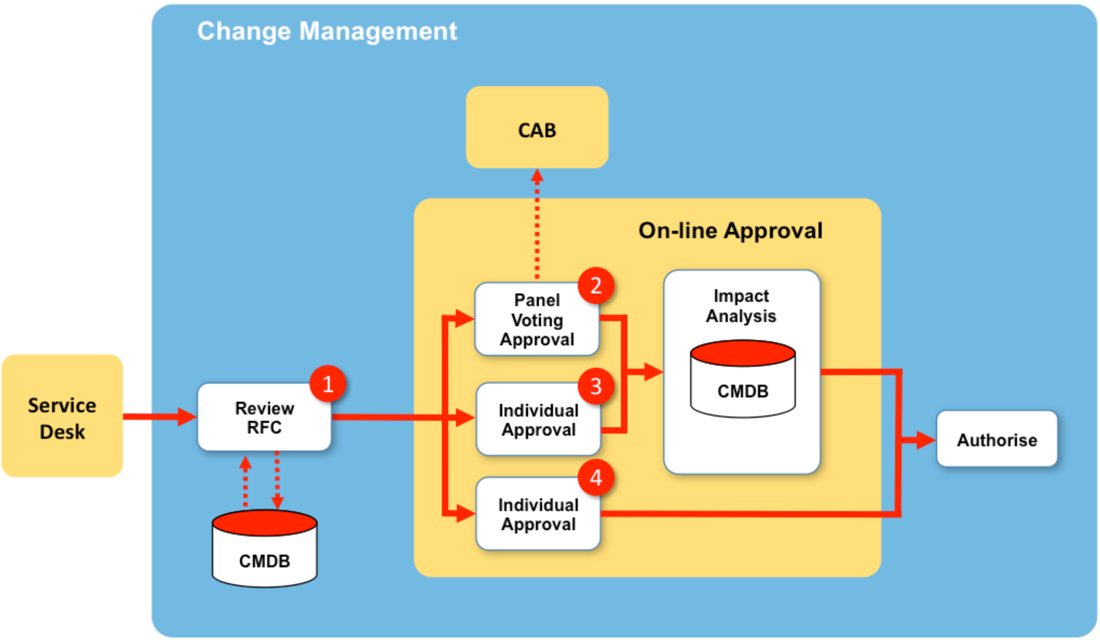


Figure 3 – Suggested Structure for On-line Approval

Basically, our On-Line Approval process will introduce a more robust Review RFC process that acts a gate keeper to filter and route tickets to the key SMEs involved in approving the change.
This is different from the Change Management process of Figure 2 in that we now have several paths to approval. For my streamlined process I have chosen four key decision points and these are shown in red in Figure 3 and described below.

Point 1 - Enhanced Review of RFC

The first decision point will give greater emphasis on the review of all RFCs based on a set of agreed criteria. For example, the complexity of the change (across systems or services), the requestor (client) or the number of configuration items likely to be impacted (inter-dependent sets of CIs). The change management team will use the CMDB as a key resource to identify where best to route the RFC.
I have selected three possible 'RFC' decision points. Panel Voting and two types of Individual Approval.

Point 2 - Panel Voting

The review stage will send the CR - both Normal and Emergency - to a selection of SMEs who can then proceed to a system of panel voting. The SMEs are selected automatically from a look-up table based on the type of request and service and systems impacted.

Panel voting can be set-up in a number of different ways, depending on what workflow system is in place. However, by way of example I suggest the following sequence:

1. Voting is conducted by a pre-selected group of SMEs;
2. Additional SMEs can be brought into the risk analysis stage by online request;
3. Weighting can be assigned to different SMEs to reduce the number of additional approvers;
4. Expression of interest emails to other associated SMEs to request input;
5. Escalation process to convene a CAB;
6. A single step sign-off or different levels of approval or rejection.

The pre-selected group (1) will be identified at the Review RFC stage by using the CMDB attributes of system, service, and responsibilities etc., associated with the scope of change. The review team will identify the actual SME names to form the group from a look-up table linked to the 'Responsibilities' attribute of the CMDB. However, the lead SME for the CR can also bring in additional SMEs into the risk analysis stage by online request (2).

In the case of (3) the weighting can be set up so that the SME with the most technical control over the change has the highest weighting. Here are two examples of how this might work in practice.

> Example 1 - If the change is mainly a network change then the Network Manager's decision would have the greatest impact on the approval and so he might be assigned a weighting of say, 80%. On the other hand a Capacity Planning SME might have just 10% and a Server Cluster SME the other 10%.
>
> Should either (or both) of these SMEs with the 10% weighting not respond during a given time frame then the approval could still proceed with just the Network Manager's approval. Clearly, there is some risk involved here by a zero response.

However, in this example the risk is both limited and manageable and places a responsibility on the additional SMEs to either respond with 'no impact' or express a concern (4) that might require further analysis. As the lead SME the Network Manager can then decide on the course of action.

> Example 2 - For some complex changes there could be four or five SMEs involved. In these cases there might be two SMEs both with 40% weighting and two further SMEs each with 10% weighting. In addition to assigning weightings to the SMEs, there could be other peripheral SMEs that were identified by the look-up table.

These additional SMEs would not form part of the panel weighting but they would be sent a description of the change and invited under the Expression of Interest category to give their input to the approval. If none is forthcoming then these will be classified as a 'no interest'.

All of the above takes place on-line via the Change Management workflow and email. However, if a conflict occurs then there is a process to convene a Change Approval Board (5). This may be considered as essential in the case of some Emergency changes. The difference here is that we have moved away from regular weekly CABs to a fall back CAB to resolve a particular dispute on a complex change request.

Once the change has been analysed then the final approval (6) can take place on-line either as a single step sign-off, or as a workflow to collect a group sign-off from senior SMEs or departmental leads.

Point 3 - Individual Approval (Normal)

Where a change is confined to a single CI, or a small set of CIs and the risk is confined then the change request can be assigned to an individual SME who will conduct the impact analysis and either approve the change on his own or send his approval to the next level. As with the Panel Voting, the SME could invite other SME under the Expression of Interest category to give their input to the approval. If none is forthcoming then again these will be classified as a 'no interest'.

Point 4 - Individual Approval (Fast Track)

I've introduced this route as a way of reducing the temptation of unauthorised changes. In my example at the beginning of this paper where I quote a server technician who said that his small change got bundled into a batch change to be signed off by the weekly CAB. The way this works is for the technician to input his change request online so it is recorded along with his assessment of minimal impact analysis and effectively self-approve, or approved just by his line manager. Again, there might be risks here but these are probably negligible.

Improvement No 2 - Employ greater use of a CMDB

For my second improvement I'm going to look at what I consider to be at the core of successfully managing change without impacting service delivery - robust management of all CIs and the availability of up-to-date system configuration data.

In my previous White Paper[1] - *The Enduring Myth of CMDB* - I talked about the importance of the establishing an interface between the Change Management and Configuration Management processes so that all changes to a CI or group of CIs will be reflected in the CMDB. I now want to look at this relationship with the CMDB from the perspective of the Change Management process.

In that paper I discuss how a CMDB can be used to limit the risk of a change causing a system outage. The risk we are trying to mitigate here is one of a poorly managed change request that leads to an uncontrolled service outage and associated breach in SLAs.

Typically, this is due to inadequate Impact Analysis resulting from:

- an incomplete baseline for the current system configuration;
- insufficient understanding of dependencies and relationships between the CIs;
- unclear ownership of CIs;
- poor communication between technical teams due to no common configuration baseline.

The implementation of a CMDB as outlined in the white paper will eliminate most of the above deficiencies, but of course no technology solution on its own will compensate for poor teamwork and team communications. Our streamlined Change Management process will use a CMDB as the source of the current baseline, plus supporting analysis from technical teams based on ITSM tool sets and local experience from SMEs.

At the Review RFC stage the Change Manager will run a report from the CMDB that shows a baseline for the current configuration for that part of the IT environment under proposed change. This is expressed as a set of CIs with known attributes that can be used as the starting point for the Impact Analysis process.

During On-line Approval the SMEs are tasked with evaluating the impact of the proposed change. The SMEs will use a variety of data sources together with the CMDB baseline report to determine how the CIs will be impacted in terms of the relationships and dependencies with other CIs. Access to this level of configuration detail is essential to reduce the risks involved in Impact Analysis.

Improvement No 3 - Implement a Change Calendar

For my third improvement I would take a close look at the type of Change Calendar currently in place.
If a spreadsheet is still being used as a calendar then I would recommend implementing an on-line calendar.

Some, but not all, ITSM workflow or Service Desk software suites will include a Change Calendar and it is likely that this is not being utilised correctly, or even enabled. There are a number of stand-alone calendars available from software vendors that offer a wide range of features. Here are a number of features that I consider essential.

A Change Calendar must be able to:

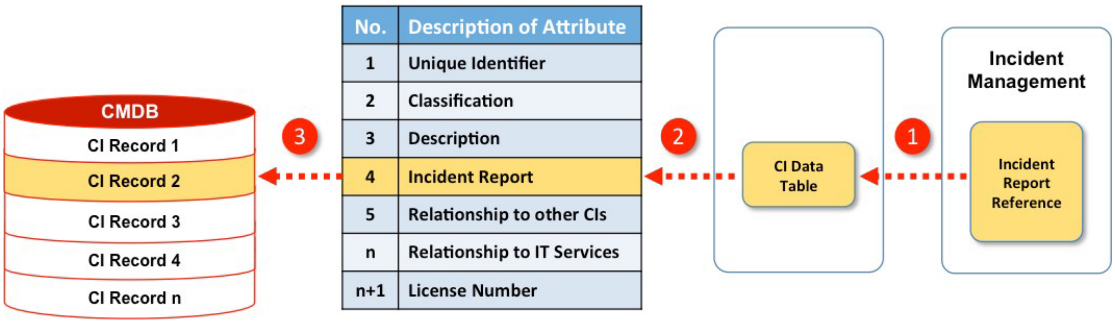
- integrate with the On-line Approval outlined in Improvement No 1;
- be shared across the various technical teams involved in managing changes;
- provide a clear overview of all planned changes in a single view;
- highlight all the changes that were recently implemented;
- highlight potential conflicts between changes;
- highlight resource conflicts to ensure that the same person is not inadvertently scheduled to implement more than one change at the same time;
- schedule recurring changes;
- automatically notify by email change managers and SMEs that a scheduled event has occurred or has been changed;
- send email reminders as the event nears and give requestors the ability to reschedule through the same email.

Clearly, the key to successful implementation of a Change Calendar is both integration with the on-line approval system plus visibility across all management technical teams. However, there are other features that could also be introduced, like connection to a resource planning system to ensure that changes are not assigned to personnel who will be on holiday over the scheduled change dates.

Improvement No 4 - Implement a link between CIs and Incidents

For my fourth improvement I want to look at how the creation of a link between CIs and Incidents can help identify if a recent change has caused an incident, and if so identify the CIs involved. Note that it may not be enough to look just at the CIs directly impacted with the recent change. Also, a poorly implemented change does not always cause an immediate outage or issue. It can often be the case that an incident will occur at some future time after the change has been made due to a number of events coinciding - for example, time based events like end of month or capacity thresholds being breached. Usually these types of events are predicted during Impact Analysis, but not always. Often these events are not apparent until they actually happen and then we need to refer to historical incident reports to aid the investigation.

To do this we need to establish a link between the CIs and Incident Reports. Our starting point is the CMDB and in particular the CI Record. In *The Enduring Myth of CMDB* I suggest some attributes that could make up a CI Record. One of those attributes is the Relationship of CIs to Incidents. This will take the form of a link between a CI record and any associated incident reports that are created as part of the Incident Process. Figure 4 shows the proposed sequence.



No.	Description of Attribute
1	Unique Identifier
2	Classification
3	Description
4	Incident Report
5	Relationship to other CIs
n	Relationship to IT Services
n+1	License Number

Figure 4 – Sequence for linking Incident Reports to CMDB

1. As part of the Incident Management process an incident report will be produced each time an incident is resolved. The report will contain details of the type of incident, along with the associated CI(s) and any other information;
2. The reports will be collated by the Configuration Management team and attached to the CI Upload file;
3. The reports are then linked to the CI Record of the associated CI(s) in the CMDB.

Once this link is established and a new incident occurs then the incident management team are in a position to compare the details of the new incident with previous incidents. This comparison can map CIs or groups of CIs involved with the new incident to historical incidents. If there is a close correlation with these CIs and the CIs involved in a recent change, then the Change Management team should be alerted and the SMEs asked to investigate as this may be the root cause of the incident.

If there is no correlation then the incident will be handled by the Incident Management team according to the standard process. This is a relatively small change to the Incident Process and I will cover this in greater detail in Hot Spot 2.

Improvement No 5 - Introduce Metrics for Change Management

I must admit to a fascination for IT metrics. I really believe that the correct use of metrics, especially in the ITSM environment, is an excellent way to monitor both service delivery improvements and, conversely, a deterioration in service delivery. Although metrics by their very nature lag events, they are useful in spotting the direction of a trend. Once a threshold has been crossed that can be a trigger to intervene.

I also believe that the single biggest mistake IT management make with metrics is to choose too many. With metrics, it really is a case of less is more. A careful selection, well managed for data integrity, will give more meaningful results than a vast array of metrics that are inaccurate and poorly understood.

In the case of our key ITSM processes of Change, Configuration, Incident and Problem Management, I suggest just five metrics for each process. This gives us a total of just twenty metric data points to collect and publish.

With that in mind, I've chosen five metrics that can be used to monitor the performance of the Change Management process. All the data needed for these metrics should be available in the Change, Incident and Configuration workflows, but this depends on the type of ITSM workflow deployed. Most ITSM workflows have some form of configurable dashboard where the metrics can be displayed. Alternatively, the data can be collected manually.

Metric 1 - The average change closure duration, in terms of days between the registration of a change and the closure.

This metric is one indicator of just how well the Change Management process is working. The life cycle of a change - from when the RFC was raised until closure - is a good indicator of the efficiency of the process.

Metric 2 - The number of unauthorised changes relative to authorised changes over a particular time period.

This metric is an indicator of how well the Change Management process is able to reduce the practice (i.e. temptation) of implementing an unauthorised change.

Metric 3 - The % of changes that cause incidents.

Essentially, this metric is an indicator of how thoroughly an Impact Analysis of a proposed change is conducted. It is yet another indicator of how well the CMDB is integrated with the Change Management process.

Metric 4 – *The* % of emergency changes.

The number of emergency changes relative to the total number of changes opened in a selected time period, like monthly or quarterly. This metric is an indicator of the potential risk to service delivery that is building up due to system problems in the IT environment.

Metric 5 - *The % of unplanned outages or service unavailability due to changes.*

Out of our five metrics this is the most critical. Whilst Metrics 1 to 4 contribute to the overall efficiency of a Change Management process, Metric 5 highlights the actual relationship between change management and service delivery.

Clearly, there is much more to discuss here, but there is a future White Paper planned that will cover in greater detail the implementation of ITSM metrics to drive improvements in service delivery.

Example ‘Hot Spots’

I’ve identified four typical ‘hot spots’ involving the impact of poor change management that have been the cause of major service outage.

Figure 4 is based on a simplified ITSM organisation that could be either a MSP dedicated to external clients, or an ITSM organisation providing IT services to an internal client. The IT Operations can be either internal or external hosting with or without applications support. It assumes that all key components are under the control of the IT organisation.

For the purpose of this paper it is assumed that the IT Operations is in-house and provides hosting, communications and applications support - within an overall governance framework.

There are four example ‘hot ‘spots’ shown in Figure 4.

- Configuration Management – Risk to service delivery due to poor integration between the Change Management process and the CMDB.
- Incident Management – Risk of service failure due to poor incident resolution times caused by recent changes.
- Problem Management – Risk to service recovery due to lack of data on known errors.
- Release and Deployment Management – Risk to a new deployment due to poor communication between the Release team and the Change Management team when releasing a new service or system.

All of the above examples involve insufficient knowledge of current IT system configuration and the following narrative will describe how the integration of the Change Management process with a CMDB can help mitigate these risks.

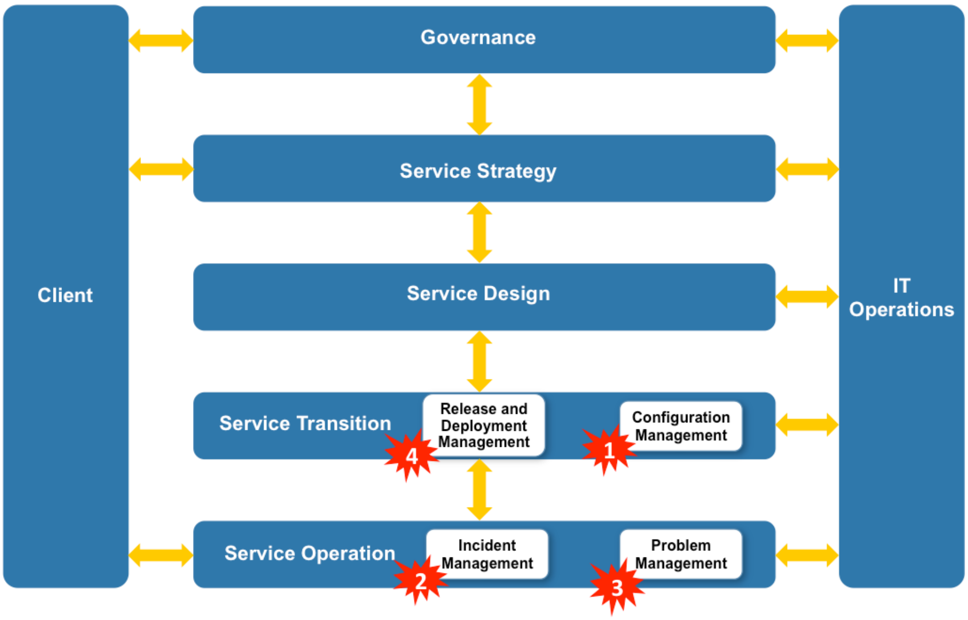


Figure 5 – Example Hot Spots

Hot Spot 1 - Configuration Management

As outlined in my Improvement 2 I'm going to expand on how the On-Line Approval will operate with the SMEs using the various sources of data available to them when analysing the impact of a Change Request. There are five key steps in this process and these are shown in Figure 6.

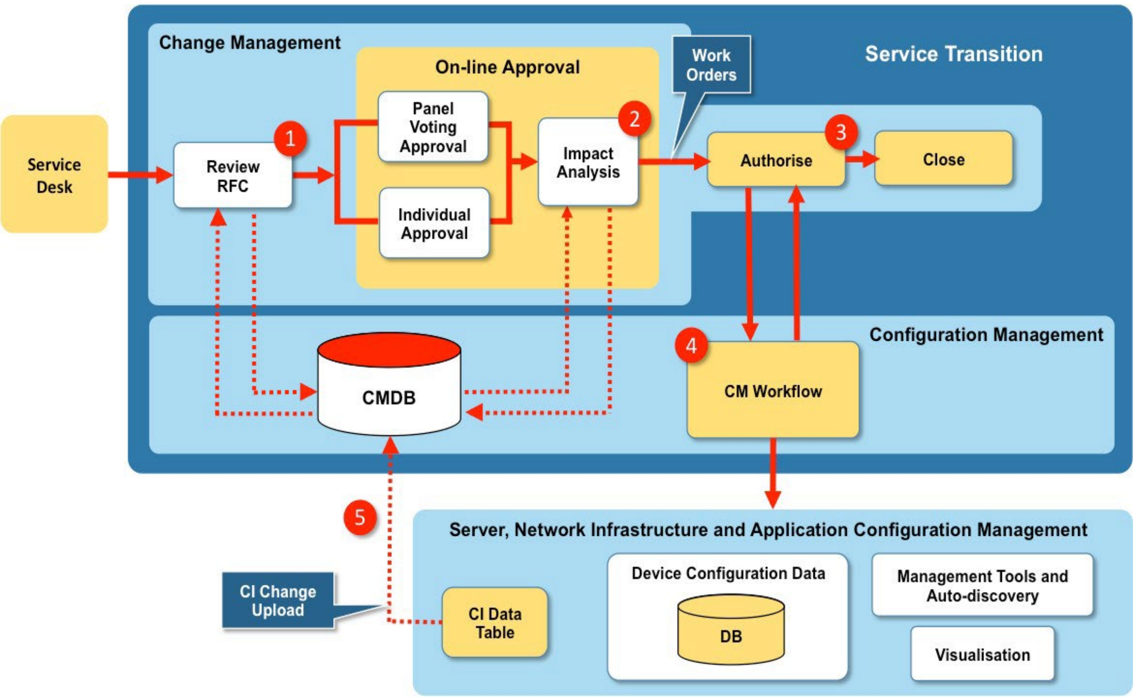


Figure 6 – Integrating On-Line Approval with Configuration management

Step 1 – Review RFC

The Change Manager will run a report from the CMDB that shows a baseline for the current configuration for that part of the IT environment under proposed change. This is expressed as a set of CIs with known attributes that can be used as the starting point for the Impact Analysis process. The CAB comprises a group of SMEs tasked with evaluating the proposed change to the IT system - i.e. Impact Analysis. These will typically be drawn from the server, network and application infrastructure and design teams, service owners, IT Security and IT Operations.

Step 2 – Impact Analysis

The SMEs will use a variety of data sources together with the CMDB baseline report to determine how the CIs will be impacted in terms of the relationships and dependencies with other CIs. The data sources, already discussed, will be outputs from:

- Visualisation tools, for example Server node modelling;
- Network device management tools;
- Software Asset Management (SAM) tools that can identify dependencies between software modules;
- Auto-discovery tools.

Plus of course a considerable amount of SME experience to interpret and model all possible change scenarios likely to impact the IT environment.

Step 3 – Authorise - Once the analysis is complete the Change Manager will authorise the change to be made via Work Orders to the appropriate technical teams. There will also be an agreed roll-back plan should the change fail. This is managed by the Configuration Management team.

Step 4 – CM Workflow - The CM Workflow will be used to manage the Work Orders through to completion.

Step 5 – CI Data Table - The final step is for all the changes to the CIs to be collected and uploaded to the CMDB to create the current configuration baseline resulting from the change.

Hot Spot 2 – Incident Management

As outlined in my Improvement 4 I'm going to expand on how the creation of a link between CIs and Incidents can help identify if a recent change has caused an incident and if so identify the CIs involved.
Incidents can be any type of failure or interruption to an IT service. Incidents are created from a number of sources, like customers' phone calls and technical staff alerts. Incidents can often occur due to a recent change (hours), or even a change that was made some time ago (weeks) but the impact has only just occurred due to a combination of events.

Figure 7 details a simplified Incident Management process and how this links with Configuration Management.

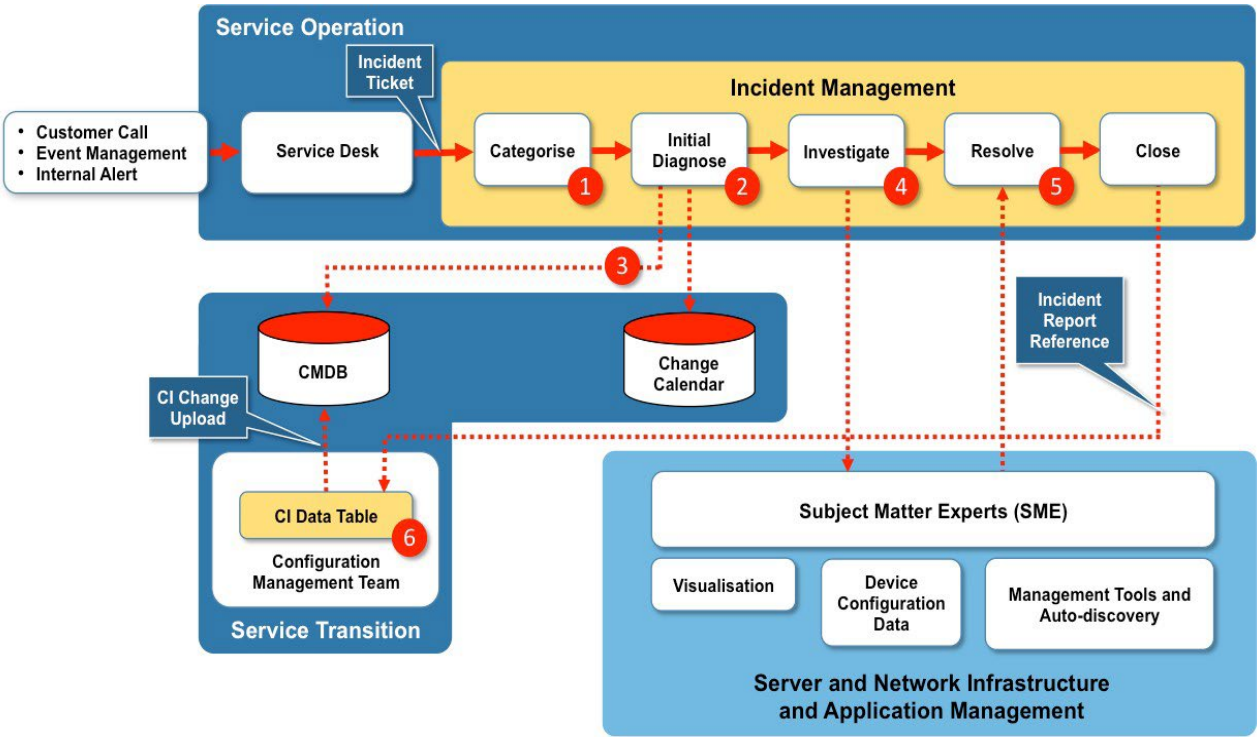


Figure 7 – Implement a dynamic link between CIs and Incidents

In this scenario all the Incidents are routed through the Service Desk where the incident is logged and an incident ticket raised. The incident ticket will contain a brief description of the incident and its impact on services or systems.

Step 1 - The incident ticket is received from the Service Desk and categorised – by priority and urgency – by the Incident Management team.

Step 2 – The ticket is then passed to the Initial Diagnose stage where the incident is checked against the Change Calendar to see if a recent change might be the cause of the incident. The Incident Management team can compare the CIs involved in a recent change with the CIs associated with the incident.

If there is a close correlation then the Change Management team should be alerted and the SMEs asked to investigate and if necessary roll-back the change.
If there is no correlation then the incident will be handled by the Incident Management team according to the standard process.

Step 3 - The Incident Manager will compile a report based on the service impacted, the location, known hardware or software failure. However, in our scenario the Incident Management process is integrated with a Configuration Management process that has a CMDB as the source of CI Records.

This will provide the Incident Management team an extra source of data when SMEs are required to diagnose the root cause of an incident. The report will list all the CIs and dependent CIs that have these attributes in their CI Records. The report is then attached to the incident ticket.

Step 4 - The SMEs will conduct diagnostics using the various tools sets already outlined, namely Visualisation tools, Device Configuration tools and System Management and Auto-discovery tools.

Step 5 - Once the diagnosis is complete the SMEs will notify the IM team that the incident is now resolved. The resolution may involve an update or replacement of hardware, or infrastructure, which in turn will result in a Change Request. The SMEs then update the CMDB with and CI changes.

Step 6 – As described previously (See Improvement 4) it is possible to associate CIs to an incident. This can be done by adding an attribute – *Relationships to Incident Records* to each CI Record.
Once an incident is resolved the Incident Management team will send the Incident Record Reference to the Configuration Management team who will add the Record Reference to the appropriate CIs in the CI Data Table.

This will help build up a picture of how types of incidents effect certain CIs and other CIs with dependent relationships. Over time, collecting this feedback on the relationship between incident types and particular CIs will help speed future incident resolution.

Finally, the CI Data Table is uploaded to the CMDB and updates the *Relationships to Incident Records* Attribute.

Hot Spot 3 – Problem Management

For the third hot spot I want to look at the relationship between Change Management and Problem Management. The role of problem management is correcting known problems and there are several aspects to this role. Establishing root cause, conducting workarounds and creating a Known Error report for faster diagnosis. It's this latter function that I want to look at in relation to the Change Management process.

I've discussed previously the benefit of linking Incident Reports to the CMDB, but there is also a case to be made for collecting Known Error reports and making these available to change management. However, I do not propose the creation of a Known Errors Database (KEDB) as defined by ITILv3.0, but rather make use of a Knowledge Management (KM) System. The reasoning behind this approach can be found in my White Paper *Knowledge Management within ITSM*[2].

My thinking here is that all the workarounds, root cause analyses reports, future change proposals that are written up by SMEs must be shared with other members of the IT support teams. The best way to do this is via a KM system that makes use of taxonomy, controlled vocabulary and metadata to compile, store and easily retrieve at the point of need – i.e. Impact Analysis.

Figure 8 below shows a simple arrangement whereby a wide range of Problem Reports can be assembled within a KM System for use by the change management team during Impact Analysis.

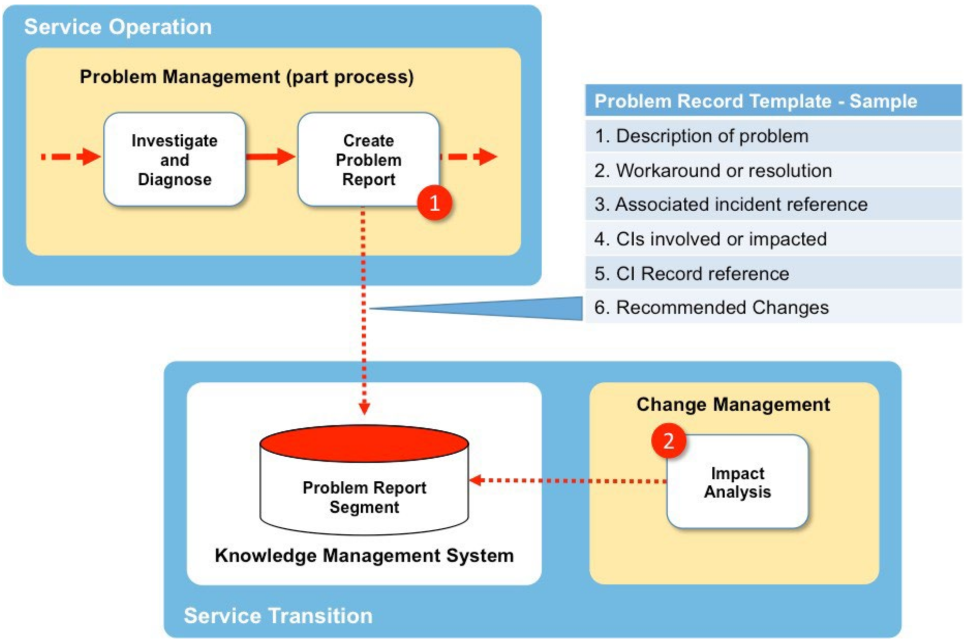

Service Operation
Problem Management (part process)
Investigate and Diagnose
Create Problem Report
1
Problem Record Template - Sample
1. Description of problem
2. Workaround or resolution
3. Associated incident reference
4. CIs involved or impacted
5. CI Record reference
6. Recommended Changes
Problem Report Segment
Knowledge Management System
Change Management
2
Impact Analysis
Service Transition

Figure 8 – Problem Records within a KM System

Step 1 – Involves the creation of a Problem Report within the KM system using a standard template that provides a consistent format of headings. A sample is shown in Fig 7 with some suggested headings. The template also includes a metadata table that speeds retrieval.

Step 2 – During the Impact Analysis stage of a Change Request the change management team can now search all the Problem Reports using 'key words' (based on the type of change in scope) and to check if that change is likely to create a new (or repeat) incident.

Hot Spot 4 – Release and Deployment Management

This hot spot briefly covers the introduction into service of a new, revamped business system or service and the relationship to the Change Management process. This is a big subject so I'm only going to touch on a few key points to consider. First, my approach is not to start with just IT change management - as defined by ITIL - but to consider change management in the context of IT Governance as well.

I find it interesting that there is little mention of change control within ITIL v3, but there is in IT Governance. (See ISO 38500: the standard for the corporate governance of IT[3]). This is critical for managing significant changes to a business critical system. Also, a large scale introduction into service is likely to be introduced in stages to reduce risk, meaning that close coordination is needed with all the other scheduled changes that might be taking place. To manage a large scale introduction into service I believe we need both IT change control and IT change management.

> Change Control - is all about critically evaluating each major change to ensure that it is the "right" thing to do in the context of the business and the integrity of the IT systems. Change control is the main path between the IT Steering Committee and the configuration management team.
>
> Change Management - on the other hand is mainly about the process we follow to assess the impact of a change and get approval to implement, whilst ensuring that changes are recorded in the CMDB.

So, when it comes to the introduction of a new service or system the hierarchy I believe we need is one that starts from IT Governance and the relationship between the:

- IT Steering Committee
- Change Control Board (CCB)
- Change Advisory Board (CAB)

Role of IT Steering Committee

The IT Steering Committee should comprise both business personnel from the client and senior personnel from the IT Service Provider. This is the main forum for the impact analysis on the business and IT environments due to the introduction of a new service or system.

Role of the Change Control Board (CCB)

The focus of the CCB must be on overall IT governance with the CCB driving the various changes through the Change Management process according to an agreed schedule. It should comprise one of the senior IT personnel from the IT Steering Committee and members of the Change Management team. The CCB will provide guidance for the Change Management team.

Role of Change Advisory Board (CAB)

Although I have previously proposed in this paper a move to online approval I would suggest that a regular face-to-face CAB meeting be set up for the introduction of any large scale new service. Certainly, on-line approval will form part of the change process but in this instance it should be subordinate to the CAB.

If the above arrangement appears over complex I do suggest taking a look at a case study of a major outage at RBS in 2012. Here is an extract from the root cause report[4] of the outage that brought down their payment system. The RBS report makes for interesting reading as it identifies the outage not just as a technical failure but also a failure of governance. To address this in the future one of the RBS report recommendations states:

> *3.12 (g) A new **Production Change Board** is to be instituted and a panel of experienced and senior technology leaders has been formed to scrutinise project implementation plans for potentially high impact changes to production systems.*

They are effectively creating a high level CCB for controlling the major changes to key Business Critical systems.

Conclusion

This paper looks at how improvements in the IT Change Management process can drive greater business value by reducing risk and costs involved in making IT system changes. The focus is on the relationship between the Change Management process and the Configuration, Incident and Problem processes.

By moving to on-line approval as opposed to regular CAB meetings the end-to-end change process can be streamlined to reduce the time to implement a change. Also, by integrating the data held in the CMDB the impact analysis stage is de-risked as the CMDB should reflect the latest system configuration data.

The addition of an integrated Change Calendar will also reduce the risk of change conflicts. The paper also introduces a number of key metrics that can be used to monitor improvements to the Change Management process. The paper also looks at how a Knowledge Management system can be used to store all problem reports in one location and make this available to the change management team to aid Impact Analysis.

Finally, the paper concludes with some example ‘hot spots’ that describe scenarios where poor Change Management can impact service delivery.

Biographical Note: The author is a partner consultant with CIH Solutions and can be contacted at info@cihs.co.uk.

References:

[1] CIH Solutions White Paper – *The Myth of CMDB* – August 2015
[2] CIH Solutions White Paper – *Knowledge Management within ITSM* – May 2015
[3] ISO 38500 – http://www.itgovernance.co.uk/shop/p-636-isoiec-38500-the-it-governance-standard.aspx
[4] RBS Report - http://www.rbs.com/content/dam/rbs/Documents/News/2014/11/RBS_IT_Systems_Failure_Final.pdf