Facilitating Organisational Change via an Automated Change Management System

Alan Litchfield
Service and Cloud Computing Research Lab
Auckland University of Technology
Alan.litchfield@aut.ac.nz

Amin Farokhpourlangroodi
School of Engineering, Computer and Mathematical Sciences
Auckland University of Technology
amin.faro@gmail.com

Abstract

An automated change management system is designed to provide a solution to the problem of organisational change management in complex environments in a telecommunication company. In the study, we propose that, just like the impact of the change on applications, the impact of a change on business processes and people needs to be addressed. The system provides an automated impact assessment report and other functions such as scheduling and notifications for key staff and decision makers.

Keywords

Automated change management system, ITIL, TOGAF, Archimate, eTOM

Introduction

Most organisations rely heavily on technology to perform their day-to-day operations and consequently, business processes are dependent on information systems (infrastructure and applications). Therefore every time a system change is proposed, corresponding impacts to the business’s operations occur and conversely, impacting business information systems.

In this paper, we describe a services-based approach to the design of a change management system. In this study, change management refers to any organisational change that has an impact on business operations, including systems, business processes, resources, and employees. In general, change management ensures that IT software, hardware and service changes are completed within budget, on time and with the least impact on other changes and current services (Hagen & Kemper, 2011). To manage operational changes, organisations implement various frameworks. Some common frameworks that focus on IT change include the Information Technology Infrastructure Library (ITIL), The Open Group Architecture Framework (TOGAF), and more specifically, the Enhanced Telecom Operations Map (eTOM) (TMForum, 2016).

However, managing technology change is as challenging as managing operational change, therefore organisations may choose to use a configuration management system to capture and manage technology change in a system (Enck, Moyer, McDaniel, Sen, Sebos, Spoerel, Greenberg, Sung, Rao & Aiello, 2009). But most organisations do not have a tool to manage both operational and technology changes, so this study investigates the relationship between business operational and technology change. The study identifies a need for impact assessment on enterprise architecture layers (business, information systems and technology) every time a change is required (Haren, 2011). An impact assessment can be difficult to complete in large enterprises, such as telecommunication companies, with multiple processes and systems. Therefore to capture the relationship between operations (business processes), information systems and resources, we propose an Automated Change Management System (ACMS). The ACMS is a tool for identifying the impact of a change across all enterprise architecture layers.

In order to investigate issues of change management and to evaluate the possibility of facilitating change using the ACMS, a telecommunications company has been selected as a case study. The design of the
ACMS is informed by ITIL and BABOK (International Institute of Business Analysis, 2009). The case has illustrated specific concerns that have affected a large and diverse organization and that has evolved to meet requirements over several decades. The case demonstrates how the company needed to put into place a range of management processes in order to address change management.

We conclude that scheduling and notification of change is important factors in managing change because planning for change becomes complex when multiple systems and processes are involved. Organisations normally have limited resources to allocate to complete change and any delay or mistake in change planning can have subsequent effects on other changes. This research satisfies the research question by proposing that the ACMS is used to capture information about enterprise systems. To store relationships between systems and business processes, ACMS has interfaces to a business process repository that already exists. Also, ACMS interfaces with ERP and Directory Service systems to capture the information related to enterprise resources.

Review of literature

Organisational change is related to factors such as business process change, system change or role changes. Any such change requires management where change management is the process of managing organisational change with the least disruption to services (Cartlidge, Rudd, Smith, Wigzel, Rance, Shaw & Wright, 2012). The importance of this is reflected in the Information Technology Information Library (ITIL) framework, used by many organisations worldwide, where change management is a key process (Cartlidge, Hanna, Rudd, Macfarlane, Windebank, & Rance, 2007). Luo, Kar, Sahu, Pradhan, & Shaikh (2008) report that each change management process has a requirement specification, an approval process, and subsequent implementation of the change. The change is tested and monitored and in the case of failure, roll back occurs.

Implementing a change management process is a challenging task. Large enterprises have IT environments that include complex infrastructure and information system capabilities (Luo et al., 2008). Having an up to date schedule for changing multiple systems in different locations and time zones is challenging (Salam, Maly, Mukkamala, & Zubair, 2008). Additionally, applications with a stated downtime window may be close to another application’s downtime window, making allowance for multiple changes in the same time window difficult (Muthuswamy, Kar, Sahu, Pradhan, & Sarkar, 2010).

Large enterprises such as telecommunication service providers use configuration management systems to track changes to their network and systems (Enck et al., 2009). Such systems provide an end-to-end view of the network and systems to allow change planning. The risk to the business operation associated with the impact of change on a system is high and needs to be analysed and mitigated (Wickboldt, Machado, Cordeiro, Lunardi, Santos, Andreis, Granville, Gaspary, Bartolini, Trastour, 2009). However, configuration management systems do not provide the capability to manage all the aspects of a change, such as impacts to business processes and roles. This highlights a need for a system that can capture the impact of a change on business processes and roles.

For consistency in changes made during the system and software lifecycle, organisations use configuration management systems (ISO, 2010). Such systems hold data about documents, software and hardware. Configuration management, a step in change management frameworks such as ITIL (Cartlidge et al., 2012), can provide automated capabilities to assist with change in the system. Capabilities include configuration identification (2010), configuration baselining, configuration control, configuration status accounting, configuration auditing (BSI, 1984), and release management and delivery (2010).

The Open Group Architecture Framework (TOGAF) provides the Architecture Development Method (ADM) through which managed changes can be introduced to an organization (Haren, 2011). Change impact assessment, addressed in the ADM requirements phase and not the TOGAF ADM change management phase, identifies the impact of any requested change.

According to the TOGAF framework, an impact assessment document includes the impact, the TOGAF phases to be revisited, priorities and recommendations (Haren, 2011). TOGAF provides guidelines for managing the impact on the architecture. It defines that if the change impacts two or more stakeholders then it is better to perform an architecture re-design. However, if only one stakeholder is impacted then
the change management process can deal with it. Therefore, a change management system takes into account the organisation’s overall architecture when the assessment output is produced.

To ensure the design of the ACMS is aligned with available change management standards, the configuration management system and TOGAF ADM capabilities are taken into account. However, these solve only some of the challenges that appear in change management. For example, gaining an approval is an important step in completing a change in a smooth and timely fashion but the TOGAF ADM does not consider the approval process (Luo et al., 2008).

To prepare a unified change plan with a single point of change request entry as recommended by ITIL (Hagen & Kemper, 2011) is challenging where planning for change may be done manually by a person or automatically by a system. This is further complicated where there are multiple change requesters (person or system) and for example, a person is scheduled to make a change on an infrastructure component while not knowing that there is an operating system running automatically in the background and that is not compatible with the infrastructure change. This identifies a need for the capability to track dependencies between changes planned by people and automatically by the system.

Thus, while a crucial part of change management is scheduling (Muthuswamy et al., 2010), scheduling may also disclose conflicts between changes (Luo et al., 2008). The challenge is greater if the system needs to be available 24x7. Also, resources allocated to make a change may not be available in the allowable time window. Therefore, to make a change to a system that considers multiple aspects and dependencies, a detailed schedule needs to be prepared.

A collaborative approach between change requester and system subject matter experts is one way to assess changes and to prepare schedules. However, Zhou, Li, Dang, Chen, Li and Liang (2014) state that taking a collaborative approach is not easy, effective, or sometimes is not possible. This approach relies on peoples’ knowledge and having successful collaborative sessions to discuss changes. The collaborative approach mentioned by Zhou et al. is useful in capturing the dependencies of current services and processes in the change system proposed by Wang, Yang and Zhao (2011) but it only resolves issues related to changes initiated by users. Collaborative planning does not cater for when system and user are simultaneously planning for a change. A solution to this problem is to provide an automated system where the change agent (the person sponsoring the change) requests the change in a system and is provided with time slots (Salam et al., 2008).

Further, Luo et al. (2008) prescribe steps to ensure success in the design of an ACMS that include: request change time (not too soon or too late), allocate the change to the right party, assess the impact of the change on existing infrastructure and services, follow change procedures, test the implementation of the change, and transition the change into the Business-As-Usual (BAU) service. We would add; calculate the impact of the system or service change on business processes (Wang et al., 2011).

To be successful in competitive and growing markets such as telecommunications, businesses frequently change processes (Xiao, Guo, & Zou, 2007). Identifying the cost of business change is a deciding factor for a change to be feasible especially when, to accomplish a business goal such as telecommunication ordering processes, layers of processes exist. It is important to be able to capture the impact of a change on business processes and not just the IT systems. That is, many telecommunication companies use eTOM as a framework to organise their business activities and processes (TMForum, 2016). eTOM has different process layers, so to achieve a business goal creates many layers of business processes. A change to a business process is likely to have subsequent impacts on other processes. Also, a change in a system may impact multiple business processes because one process could be a sub-process for a larger activity.

To simplify and automate business processes and to fulfill business goals, organisations use business applications. However, such applications introduce a complex environment that involves users, business processes, business systems and underlying IT infrastructure. Frameworks exist that allow businesses to adopt a standardised structure for business and system architectures, such as ITIL, TOGAF and Zachman. While these frameworks outline basic processes for managing change, they do not assist with the assessment of end-to-end changes. Additionally, systems are designed and developed using multiple Application Program Interfaces (API’s) and web services, which may be tied together to provide a business system (Xiao et al., 2007). The web services themselves may be loosely coupled and changing one small part of a process may have underlying impacts on many web services, which means they may need to be recoded. Further, if the number of transactions processed through the API’s or web services
increase as a result of the change (for example for a marketing campaign), the more infrastructure resources, such as network bandwidth and server capacity, may need to be made available.

Therefore, impact assessment is difficult to achieve but is vital to ensure uninterrupted business activities.

Currently, no effective tool exists to assess the impact of a service or system change on business processes or vice versa (Wang et al., 2011). However, to assist with assessing the impact of a change, Wang et al propose a system called Service Change Analyser (SCA). The system assesses two change types, Process and Service, and each change type is classified into sub-types. SCA can be used to implement the dependencies between services and processes and provide assessments for each proposed change. Ideally, a change analyser should provide more than just dependencies, for example, a code change from a business process change may impact another business process (Xiao et al., 2007). Business analysts who work across the business to assess the impact of change may be only aware of business process impacts if a process is being used by another business process. They may not necessarily know about hidden impacts that the system change may introduce. Such hidden impacts affect the overall cost of change, leading to unexpected cost overruns and delays. If all underlying impacts are known upfront, the cost of the proposed change may not be considered justifiable. Xiao et al. (2007) propose an approach for business analysts to assess both the code level impacts and business process impact. The steps are used to develop a business process around the ACMS.

However, Xiao et al. (2007) do not look beyond the code in their solution, for example to infrastructure. A change in the execution time of a task in a process may not impact code but may, for the duration of the task, impact the availability of infrastructure. Alternatively, if the output of a task changes, the impact on other tasks may be significant and may require further code changes but could have minimal or no impact on infrastructure. Another scenario is when a new task is introduced to a process, where it is more than likely that the impact is across the layers of business processes, systems, and infrastructure.

In our view, the correct approach is to provide an end-to-end assessment, which we define as the capability to assess the impact of a proposed change on an information system that includes applications, business processes and people. Identification of end-to-end impacts (mapping the business layer and system layers) provides a challenge for the impact assessment process.

Aversano, Bodhuin & Tortorella (2005) propose an approach to detect the impact of a business change on the system change, that is focused on providing a strategy to detect the change. Misalignment detection can occur by mapping the task to systems and dependencies in the system. To show the magnitude of an impact, a process that can be measured may be fully or partially dependent on a system. For a system to assess the change and for dependencies to be defined, dependencies need to be modeled (Aversano et al., 2005), for example with Unified Modelling Language (UML) (OMG, 2015). This is particularly important if the change management system is introduced to an existing business where a current state and relationship of processes and systems are to be defined.

When the two approaches proposed by Xiao et al. (2007) and Aversano et al. (2005) are compared, we see that the latter provides an extra step, to plan identified changes. Thus the ACMS provides the capability to plan and provide suggestions for scheduling change, or by grouping some changes together. The change management system ought to provide a service to the business that saves time and effort when investigating the impact of any change and to reduce the need for collaborative sessions to assess changes. The system should provide notifications to the business of an upcoming change (an example could be a hardware or software life-cycle issue).

Therefore, the concept is that change management can be provided as a service that is consumed by other internal or external business units to introduce changes to systems and processes. The service provides internal or external business units with a detailed analysis of the impacts of change, as well as other features such as scheduling, tracking the change and notifications. Upon reviewing related work, we found little that related to an end-to-end change management Platform-as-a-Service.

The ACMS can be built to have web-services exposed to other businesses so they can interact and be aware of changes in another organisation. To minimise the impact of a change, it is important to assess the end-to-end impact of change, as is the importance of going beyond impact assessment, to scheduling and planning for change using systems.
Method

This study addresses the question: Can an automated information system facilitate change management in an organisation by providing an automatically generated impact assessment report and change schedule of changes to the change agent?

The study offers a system design that can provide an end-to-end view of the organisation in order to test our hypothesis; that an ACMS can facilitate change management by providing an automatically generated report of the impacts of proposed changes.

The study focuses on the identification of issues in the case study, capture the requirements related to the issue and provide a solution. Therefore, methods employed to guide this research to propose and design the solution are DSR (Hevner & Chatterjee, 2010) and Requirements Engineering (RE) (Attarha & Modiri, 2011; Brooks, 1987; Liu, Li & Peng, 2010; Marschall & Schoenmakers, 2003). RE introduces the International Institute of Business Analysis (IIBA) Business Analysis Body of Knowledge (BABOK) which is used to elicit requirements (Brennan, 2015).

The output and findings of the study are models that detail the system design, from which the ACMS is designed. Therefore other than RE, ArchiMate Modeling Language for high level architecture and design (Jonkers, Proper, Lankhorst, Quartel & Iacob, 2012) and Business Process Modeling Notation (BPMN) for modeling business processes (OMG, 2013) have been applied. Unified modeling Language (UML) is used for functional modeling and for use case diagrams, plus UML data models for data modeling and analysis (OMG, 2015). The ACMS designed in this study is specific to the case study below, however to extend the service to other organizations should not require significant change to the design as it is provided as a web based service.

Case Study

The telecommunication service provider in New Zealand provides wholesale services to other service providers. The company has many suppliers that build and maintain the telecommunication network across the country. However, to manage disparate operations, communication and IT management within the company has become fragmented. Examples of departments include the marketing department finds new customers and markets, the customer services department deals with customer satisfaction and order fulfillment, and service assurance and billing is a separate department that deals with network build and maintenance.

All the company departments use a shared IT department that reports to a Chief Information Officer (CIO). The CIO team is a bridge between the telecommunication company's business and technology. To facilitate that, the CIO office has a change management team that employs change agents, people responsible for organisational changes. The change agents own systems and processes. They are expected to provide change control for systems and business processes from one of the following change types:

1. Business request: new initiatives, operational issues such as software or hardware faults, and business development.
2. Project: involves significant changes that need careful management and implementation.
3. Application or infrastructure lifecycle: infrastructure and applications have a discrete lifetime. They normally go out of support after a few years and need to be replaced.
4. New Technologies: to provide business agility, to create new products or business opportunities, new trends are adopted and emerging IT capabilities are required.

Business Analysts use a business process repository to create business processes using BPMN which stakeholders can review. To ensure usability is in place, solution architects use the application to draw system related diagrams. The system provides the ability to map business processes to use cases, and requirements and business rules to use cases and business processes.

The change team in the telecommunication company is responsible for identifying the change type and to manage it. As there may be unknown impacts, a challenging task in a complex environment such as a
Automated Change Management System to Facilitate Organisational Change

telecommunication company involves providing cost estimates for proposed changes (this step is not always taken).

The lifecycle change type provides an example of the extent of issues within the company. Without a change management system in place, a change agent must rely on their own knowledge to identify lifecycle issues. For example, Microsoft’s official support for a certain Windows operating system ends on a particular day and it is up to the change agent to identify the impacts of such lifecycle issue. Given that making a change in a large enterprise is time restricted, the change needs to be planned months in advance. While programs can be used to remind the change agent, the challenge is identifying what servers are running the soon to be outdated operating system. The next challenge is for the change agent to identify the supplier that manages the server, which could be in-house, hosted in another company’s data centre, or hosted on the cloud. Then further analysis is required to confirm if the server is part of any other changes occurring or that need to be completed in the near future.

All of the above are not sufficient for the change agent to decide if a change can go ahead. The main challenge for the change agent is to identify the impacts of the proposed change on other systems because a server can run multiple applications.

Added to this complexity, many applications in a telecommunication environment are legacy and written in old programing languages. Such programs may not run on a later version of the Windows operating system, so that when a server’s operating system needs to change and to understand the product roadmap and confirm if the application can be migrated to a later operating system, the application supplier needs to be engaged.

Assuming supplier engagement goes well, the change agent has a limited window of time to make the change. This is where the change agent engages the business to learn out about processes used by applications on the server. Also, it is common for a telecommunication company to run promotions for which a highly available system, processes, and resources are required to meet market demand. The change agent will have to work with the business to find a date and time where the change can happen and with the least risk to the marketing campaign.

Once the change date and time has been agreed, the change agent needs IT resources, team or supplier, to perform the change.

While this process for a change represents a best-case scenario, if the application running on the server is not supported on the new operating system, then more work is required to upgrade the application and that may have other business impacts, such as training or changes to processes. Also, there is a risk that the list of application servers is not up-to-date (this process is manual).

Therefore, managing a change is resource intensive and error prone. A change agent may be able to estimate the cost of delivering the change but to prepare the impacts of the change, there is no way to account for the number of hours consumed by multiple resources across the company and its suppliers.

Additionally, in the telecommunication industry most commercial customers use business-to-business (B2B) services to get access to the company’s systems and services. There are industry standards and regulations in place to advise the B2B users of any change that is going to impact their processes within a certain timeframe. Breaching the timeframe costs the company in penalties, therefore any change resulting in a customer process change will need to be communicated to the customer as early as possible so they can get ready to make changes in their own systems and processes.

ACMS Solution

In this section, we outline the design solution for the AMCS (see Figure 1). According to Brennan (2015), enterprise analysis should focus on the business situation, existing capabilities, solution approach, solution scope, and requirements.

Business Situation: A principal goal of the company is cost saving, however it has a large fleet of legacy hardware and software so the company is focused on exiting most legacy systems. When addressing lifecycle issues, legacy systems will be replaced. However, the change notification that the business normally receives from either the software or hardware vendor or the change management team is not always sufficient to assess end-to-end impact. Insufficient time to react to the change has made creating a
workable architecture very difficult to achieve as the architecture roadmap can only address known issues and tends not to focus on the impact of proposed changes.

Existing Capabilities: This is a mature company with many capabilities across multiple areas, however change management capability is very limited. For example, Business Analysts use an existing application as a ‘business processes repository’ to create business processes with BPMN (OMG, 2013). A read-only right for the application is provided to stakeholders so they can review the content of the process models. To ensure consistency, solution architects use the application to draw system-related diagrams. The system provides the ability to map business processes to use cases and can also map requirements and business rules to use cases and business processes. The business process repository capability is used to design the ACMS.

Also, a software configuration management system exists but is an outsourced capability and direct access is not granted to the company. The vendor that provides the application support and development uses the configuration management application to track changes. There is no link between the software configuration management system and network configuration management systems because most of the changes to, and management of, the network is carried by various vendors. However, the software configuration management system may be used if a vendor is carrying both network and application changes but again, there is no visibility for the company. We assume that the configuration management systems will be consolidated and access to ACMS will be managed via web-services.

Currently, most application and system lifecycle statuses are captured in spreadsheets and from information provided by the application vendor or supplier. A word processor is used to document the impacts of a change. Email is used to distribute documents to stakeholders and gain approvals. Approvals are embedded into the document and the document is loaded on a Microsoft SharePoint page.

There are no specific capabilities to track changes across the organization, so to track the change by some change agents, self-made project management style Gantt charts are used and then stored in a spreadsheet. Also, there is no capability for technology owners to capture system or application details such as license data or end of life year. This is a principal reason behind some rushed and failed projects.
Solution Approach: The solution is to provide a system that provides the capability to manage end-to-end change, so that the impact of a change is assessed from the business process to application and infrastructure, with consideration for the impact on resources. For the solution to work, integration with both vendor and supplier (as stakeholders) change management systems or processes is required. Therefore, it is important to engage with suppliers early on and capture their requirements.

Given that current-state analysis and the migration of data will be protracted, an iterative waterfall approach to developing a solution for the application is adopted, whereas for the development of the system’s GUI, an agile approach ought to ensure the features are aligned with user requirements, expressed as user stories.

Solution Scope: The scope of the solution is the system that manages end-to-end change. This system needs to be provided as a service to customers and also be able to consume services from vendors and suppliers. However, the definition of exposed or consumed web-services is out of scope for the study. While the system is designed to manage customer, supplier, and vendor changes, it can be developed further to provide web services to external users or be used as a change management Platform-as-a-Service, serving multiple clients. Also out of scope, the study does not address cost or charging models for the system, but these will be necessary later. To assess the impact on business processes and users, integration with the organisation’s directory services and business process repository are required. Additionally, to gain information about employee status, such as leave, resignation, and working hours integration with the company’s ERP system is required.

Requirements: These are divided into business, stakeholder, solution, and transition requirements. Ten high level requirements have been identified that include those aspects already discussed and the need for correct authentication.

The ACMS has interfaces with other enterprise systems and the business process repository, an application that captures all the business processes of the company. It is used by business analysts and solution designers, system architects, engineers, and operations teams. The repository captures relationships between processes and between process activities and other systems. This interfaces with the ERP to capture information about resources, assets, costs, contracts, invoices, and many other enterprise resources. The ACMS needs access to information in the ERP to assess the availability of resources and their lifecycle records. A Directory Service interface captures employee information such as account information, passwords, location, and rights to applications. ACMS uses the directory service interface to identify and authorise the user. The Configuration Management interface provides the main input to ACMS to retrieve information about the status of systems, scheduled changes, and the relationship between system components. And, the Change Management Service interface provides automated change services for users inside or outside the organisation. The service will depend on functions developed and exposed as web-services.

To provide an understanding of the ACMS, UML is used for use case modeling and to detail functions for the system (OMG, 2015). For the project, we have documented those that provide unique functions for the ACMS (due to space limitations, we cannot provide a figure to illustrate the component design). Enterprise Systems (ES) in the use case includes applications, an application component and infrastructure for the applications. An enterprise system is defined as any kind of computer system such as a network component (e.g. a firewall), an infrastructure component (e.g. a server), an application (e.g. Microsoft Exchange) and an application component (e.g. the customer database of a customer relationship management system).

The ACMS needs to have the capability to monitor enterprise systems because the components of an enterprise system can change from time to time, be retired or replaced. Also, new components may be introduced into the enterprise architecture as the business grows or responds to business opportunities. It is preferred that the Enterprise System Management (ESM) function automatically detects any new or changed component. The ESM consists of five use cases and associated sequence diagrams, however due to space limitations, these are not included in this paper.

ACMS will be used to request a change. A change can be related to a change, to a business process, or to an enterprise system. There are business rules associated with requesting a change, for example, a change can only be requested if an impact assessment has already been completed for the change, or that a change has to be approved by approvers and then communicated to stakeholders.
The ACMS will be used to run impact assessment prior to a request for a change being made. The system provides a detailed analysis of all the relationships the ES or businesses have with other ES and processes in the company. These functions represent important differences between the proposed system and a typical configuration management system used to manage change. The ability to complete an automated impact assessment can save an organisation time and resources, and therefore cost.

**Conclusion**

We believe the ACMS meets the requirements of DSR, that is, a high quality IT artefact that provides a solution to real (related) problems (Hevner & Chatterjee, 2010). IIBA’s recommended approach completes the enterprise analysis and defines the problem. Requirements are captured in multiple areas with traceability to goals of the study and we consider the ACMS to be the right solution to the business problem outlined. Presently, the company suffers from manual and time-consuming change processes, therefore we propose to automate many aspects of change management using the ACMS. Additionally, using the criteria proposed by Aier & Fischer (2011) to evaluate DSR related solutions, we find the study satisfies most of the criteria. The research alignment with DSR needs to be reassessed following development and testing of the proposed solution, to investigate if limitations have had any impacts on the DSR alignment.

Considering the use of frameworks, we did not design ACMS to reflect business layers but that may be a useful approach for a future research. Embedding ArchiMate into a change management system will provide capabilities for impact assessment on changes to architecture. For example, TOGAF considers any change impacting more than two stakeholders to be a candidate for an architecture re-design (Haren, 2011). We believe that by embedding the TOGAF framework, we can map changes to stakeholders, and provide a view of the impact on the architecture areas. This allows change agents allocate the right architecture resource to assess the architecture impact further. The architecture impact assessment report shows the areas in which the architect needs to focus on. However, we do not believe that the proposed ACMS will correctly detect entire-architecture impacts.

**References**


