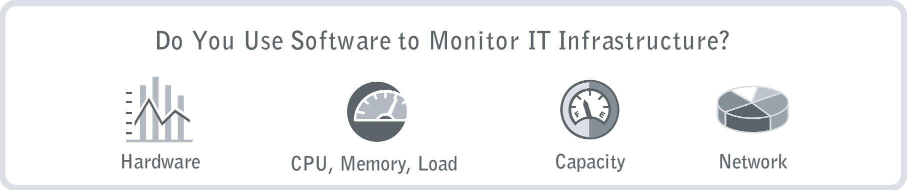
Most unavailability today is caused by change. Most IT organizations recognize the centrality of change to their operational effectiveness. Yet a gap persists between actual change activity and the documented Change Management Process. This change control gap results in manual activity by IT departments to control and minimize the costs of change. In this paper, we explain how adding Control to existing Change Management solutions can bridge this gap and enable your IT organizations to deliver highly available IT services.

Do you monitor for change?

Ensuring the continuous availability of critical systems is the Holy Grail for IT organizations. To move towards that goal, IT organizations have invested heavily over the past decades to build sophisticated systems that help them prevent, monitor, detect, and remediate failures in the least amount of time - maximizing uptime and minimizing downtime.

Historically, bottlenecks in infrastructure tended to be around hardware failures, capacity constraints and network bandwidth availability. To solve these problems, organizations invested in solutions to monitor these aspects of their daily operations. Today it is routine for organizations to monitor hardware, utilization (CPU, Memory, Network) and network traffic for anomalies or failures, and when we ask organizations whether they monitor these parts of their organization, we invariably have heads nodding in assent.

The organization's capabilities in these areas have become more sophisticated, and as a result, the bottlenecks have shifted. There is the occasional hardware failure or capacity overload, but these


Do You Use Software to Monitor IT Infrastructure?

are rare and easily resolved without a great deal of cost in most cases. Today, the bottleneck is different. The cause of most downtime has shifted to change. Industry research shows that ***up to 80% of system unavailability is caused by incorrectly applied changes.*** The natural question that follows is: *Do you monitor and control change?*

We tend to see far fewer people nodding their heads in agreement this time. It's not surprising because downtime caused by change is a more difficult beast to manage. To begin with, change happens almost continuously and with much greater frequency than a hardware upgrade, for example. Second, change tends to be complex and interdependent, with multiple parties involved. In most organizations, the impact, dependencies and ramifications of change are not known fully until it is actually deployed in production. Finally, the ways in which change occurs in the organization has its roots deep in the organization's culture and behavior.

Since most unavailability is caused by change, getting control of change in your environment would be the logical next step in the evolution of systems management for maintaining high availability. Given the difficulties outlined above, how might this be done? We will look at availability in two closely related, but separate categories. First, what can be done to increase uptime? And second, what can be done to recover quickly from downtime? In ITIL (Information Technology Infrastructure Library) terms, increasing uptime is about protecting the service while making changes as part of Change Management. Decreasing downtime is about quick resolution of incidents and is part of Incident Management. Figure 1 shows these two components of increased availability.

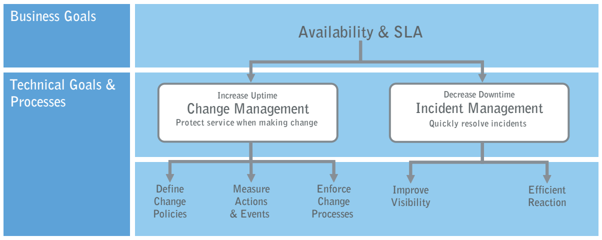


Figure 1: Two components of availability

Increasing Uptime: Change Management.

Organizations generally increase service uptime through improved change process. The most common means by which organizations improve their change process is by implementing a change management system. As the graphic below indicates, the key actions required for better change process are:

- **Defining Change Policies.** Define the rules and circumstances in which changes can be implemented. These rules can be encoded using a change management system (e.g. "high priority changes require two approvals"), but validating that the rules were followed is still an exercise in faith and hope in most cases.

- **Measure Actions and Events.** One of the key tenets of ITIL is measurement. How can change activity be measured and reconciled against the documented change process? Do you know how many changes were made in the organization? Which of those changes were made within prescribed change windows? Change management systems help somewhat by allowing the automated tracking of change requests. However, reconciling these requests against change activity is still a manual process.

- **Enforce Change Processes.** Once change policies are defined and socialized within the organization, there must be a way to enforce them. Today's best practices rely on edicts to adhere to change process passed on from those higher in the organization. To verify people follow the process, elaborate reporting and documentation requirements have been put in place to minimize the risk of out-of-process changes. In other words, enforcing change process is largely manual.

As Figure 2 shows, organizations trying to execute the actions required to achieve increased uptime are stymied by the manual effort involved, including socializing policies, relying on threats for enforcement, and manually reconciling change activity against process.

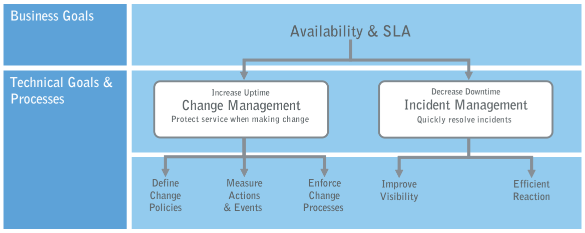


Figure 2: Methods to increase update are largely manual

Decreasing Downtime: Incident Management

Similar to the section on uptime, there are a set of required actions for decreasing the time necessary to respond and resolve incidents as they occur. The key actions required are:

- **Improve visibility**. In order to effectively diagnose a problem, the first requirement is obtaining comprehensive, up-to-date change data. In practice, most systems collect change data by periodic system scans which may not be current and may not reflect the more recent changes (the very changes which are much more likely to have caused the problem!). Lacking this information makes the diagnostics effort much more time-consuming and laborious as information is collected from disparate sources. In fact, industry research shows that about 75% of an unavailability window is consumed by just gathering appropriate log information. Once the appropriate information is available, fixing the problem tends to be much easier.

- **Efficient Reaction**. Once the problem has been diagnosed, the problem needs to be solved. In most cases, the approach followed is "fix, then validate." In other words, log into the production environment and execute a series of actions to get the service back up. This solves the immediate problem, but the system is now in an unknown state. Organizations try to mitigate this issue somewhat by requiring documentation after the fact and conducting post-mortem reviews. Once again, largely manual processes.

Figure 3 shows how the actions required for decreasing downtime are still executed in a largely manual way.

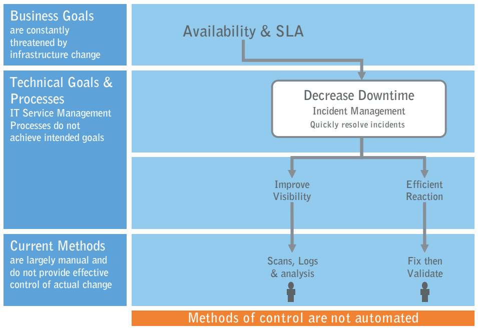

Business Goals
are constantly threatened by infrastructure change
Availability & SLA
Technical Goals & Processes
IT Service Management Processes do not achieve intended goals
Decrease Downtime
Incident Management
Quickly resolve incidents
Improve Visibility
Efficient Reaction
Current Methods
are largely manual and do not provide effective control of actual change
Scans, Logs & analysis
Fix then Validate
Methods of control are not automated

Figure 3: Methods to decrease downtime are largely manual

The Change Control Gap

The previous sections talked about the required actions to improve availability, in the context of increased uptime and reduced downtime. The conclusion is that a lot of these actions are manual. This manual effort is precisely what manifests itself as the daily struggle that IT organizations go through to meet service level agreements (SLAs). This manual effort is what we call the ***Change Control Gap***. It is the gap between desired and documented process, and actual change activity within a given organization. This gap results in areas of difficulty and increased cost, including:

- **The Visibility Gap.** Views of actual change are usually limited and out of date. When an incident occurs, comprehensive and up to date information is critical to minimizing downtime. When actual change information is incomplete or out of date, the unavailability window is considerably lengthened.

- **The Accountability Gap.** Control of infrastructure relies on people following process. As long as people follow process, things are fine. Problems arise when the process is circumvented or when proof is required that process has been followed (e.g. for compliance).

- **The Enforcement Gap.** When process is put in place but not followed, the law of diminishing returns sets in. That is, the more reality diverges from the process, the less likely it is that the process will be followed in the future. It is a vicious cycle that is costing IT departments money and lost opportunity to add value to the business, rather than incident response.

Figure 4 shows the gap between process and infrastructure – the Change Control Gap.

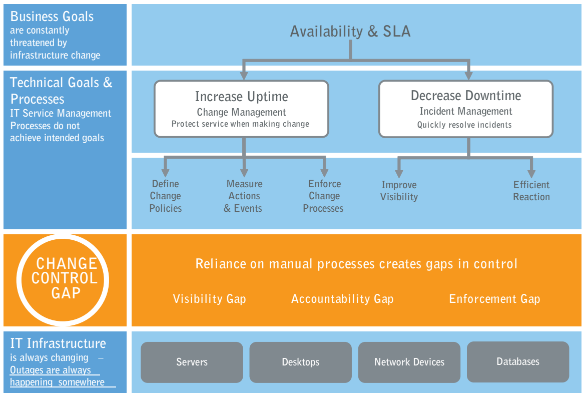


Figure 4: The Change Control Gap

Adding Control to Change Management

Giiven the costs of the change control gap, how can this gap be bridged? In a word, automation. If all the manual steps required to manage change could be automated, you could bridge the gap between change process and change reality. We will show, in this section, how adding control to Change Management is the means by which companies are successfully addressing this issue. So what is required to add control to Change Management? The good news is that the set of requirements is small and well-defined. In addition, this solution leverages existing investments of time and resources in change management systems. Adding control to change management could be boiled down to a set of three functional requirements:

- **Real time change visibility.** Everything starts by gaining insight into what is actually changing, as it happens. Real-time change visibility is the answer to the question: "Do you monitor for change?" With real-time visibility come tools for effective forensics and insight into actual change behavior.

- **Accountability.** The key to effective process and high service levels is accountability. The fundamental requirement for accountability is the ability to measure. Questions about the number of changes being made in the environment, by whom, and when should be precisely answered to measure effectiveness. Linking actual changes to change request approvals in an automated fashion provides the crucial information about changes that follow process, and changes that do not. A side benefit of automating this link is that documentation for compliance can be done automatically.

- **Change Policy Enforcement.** The ultimate aim of any documented policy is that the policy be followed. Whether the policy requires that changes must be approved before deploying into production, or that the weekly maintenance schedule must be adhered to, there is a high correlation between following the policy and maintaining highly available IT services. Most enforcement mechanisms rely on edict and post facto documentation. Change policies should be automated through technical means so that documented policies can be enforced with certainty, rather than hope.

This set of three requirements adds control to change management and provides the means to bridge the gap between actual change activity and documented change process. We hope to have shown you that change control is at the heart of providing highly available IT services in your organization. To learn more about adding control to change management, please visit http://www.solidcore.com/products.

About Solidcore Systems

Solidcore adds control to change management. Solidcore's S3 Control software is the industry's first and only solution to automate the enforcement of change management policies. Solidcore automatically reconciles infrastructure changes against change tickets, and provides real-time change auditing so enterprises can measure the effectiveness of change management processes and policies. Customers trust Solidcore to improve service availability, implement ITIL initiatives, and lower costs related to Sarbanes-Oxley compliance. Solidcore also provides change control for embedded systems and is used by major device manufacturers to securely leverage open systems to meet their business requirements. Solidcore is headquartered in Palo Alto, California. For more information, visit www.solidcore.com.

solidcore®

Solidcore Systems, Inc.
3408 Hillview Avenue, Suite#180
Palo Alto, CA 94304

Email: sales@solidcore.com
Web: http://www.solidcore.com
Tel: 888.210.6530