Research Report

Change, Configuration, and Release Performance Study

Identifying IT best practices that predict the highest levels of performance

Based data from 341 IT organizations
and 11 executive interviews

IT Process Institute
www.itpi.org

Advancing the Science of IT Management

This study was funded in part by the generous support of our managing sponsor CA

Executive Summary

Change, configuration, and release practices are recognized as crucial operating practices that enable a stable and secure computing environment. The IT Process Institute conducted interviews with 11 IT executives, followed by a survey of 341 IT organizations, to determine which of 57 recognized change, configuration, and release practices best predict top levels of performance.

The survey revealed 30 individual practices—grouped into seven sets of practices that are commonly implemented together—that best predict performance variation among top-, medium-, and low-performing IT organizations in the study. Analysis reveals that many of the commonly recommended ITIL practices may be necessary but not sufficient to drive top levels of performance.

This research report highlights five key findings are designed to help IT organizations focus on implementing those practices that are shown to predict top levels of performance. Broad adoption of these key performance drivers can lead to consistently high and predictable levels of performance across locations and business units.

IT organizations can leverage the research findings to focus their ITIL implementation and process standardization and improvement efforts to attain top levels of performance. The study also provides benchmark data that can be used to compare practice and performance to the top-, medium-, and low-performers in the study.

Table of Contents

List of Tables

List of Figures

1 – Introduction

Business and IT executives don't like surprises caused by IT systems. Increasingly, they don't like location to location variation of IT service and system performance that comes from individual skills and preferences being applied to the management of business-critical IT systems.

To reduce surprises and normalize performance, IT organizations are turning to industry best practice operating procedures along with standardized consolidated and virtualized computer environments.

However, even with the ongoing evolution of best practice frameworks such as ITIL v3, the framework is too comprehensive for any organization to adopt in its entirety. Most organizations pick and choose specific best practices based on their most pressing challenges. They also inter-mingle best practices with internally developed operating procedures and IT controls to meet the specific needs of their organization.

If you can't implement an entire best practice framework, and certainly not all at once, where should you focus your process improvement efforts to get the biggest performance improvement? What specifically, of the hundreds of industry best practice activities actually improve performance? What is required for you to get the desired benefit from process improvement efforts?

To help answer these questions, we conducted a study on how many widely recognized operating practices actually impact performance. We focused on change, configuration and release practices, as these are widely recognized as central to the modification and upkeep of production systems. We interviewed 11 IT executives from organizations that are recognized for operational excellence. We then developed a survey to collect data from 341 other organizations to verify what we heard, and find out what actually works for IT organizations with a wide range of budgets and capabilities.

Key Performance Drivers

When we analyzed the survey data including 57 individual practices and 15 performance measures, we identified 12 sets of practices commonly implemented together. Seven of those sets of practices, including thirty individual practices, were shown to be statistically significant predictors of top levels of performance across the organizations in the study.

The key performance drivers listed from highest to lowest impact include:

- **Release scheduling and rollback**—In this set of practices, IT organizations develop and maintain a fine-tuned cycle of build and test, and then release only during maintenance windows with tested rollback plans. Data about the root causes of release exceptions is fed back to systematically improve the process.
- **Process culture**—IT executives demonstrate that following a specific process is "how we do things here." The company's hiring, reward, and promotion policies reinforce the basic job expectation that people will follow documented processes and procedures.
- **Pre-release testing**—Before release, changes are tested in a pre-production environment that is maintained to be sufficiently similar to the production environment.
- **Standardized configuration strategy**—Configurations are standardized with golden builds and production systems are updated from an approved standard. IT personnel receive information about target and actual configuration. Systems are regularly monitored for unapproved changes or configuration drift.
- **Change linkage**—Change requests are linked to both infrastructure components and business service or business need. Support personnel are given access to change history to aid incident and problem management.
- **Controlled production access**—In this set of practices, IT organizations remove developer access to production; maintain well-defined roles; separate development, test, and release duties; and match users to appropriate system access permissions.
- **Process exception management**—In this set of practices, IT organizations place significant focus on eliminating process variation, including identifying exceptions, diagnosing root cause, and implementing fixes. Executives participate in monitoring exception rates and improvements.

The following five sets of common practices, which include twenty seven individual best practices, did not predict performance variation:

- **Change process routing**—Standard and emergency change requests are routed, tracked, and approved on the basis of predefined criteria.
- **Multi-function phase gate**—Checkpoints are clearly identified throughout the process, and multiple functional groups within IT can include their requirements at the appropriate checkpoint.
- **Change oversight**—A review committee or change manager reviews all change requests categorized by level of potential risk, ensuring that requirements are met prior to release.
- **Development integration**—An end-to-end lifecycle integrates development activities into the change and release processes. The causes of release exceptions are fed back to development to help improve the process.
- **CMDB with dependencies**—A configuration management database (CMDB) describes the relationships and dependencies between configuration items (infrastructure components), and the change tracking system identifies upstream and downstream dependent systems that may be affected by a change request.

Summary of findings

Five key findings were identified based on the executive interview themes and statistical analysis of practices that do and don't predict performance variation.

1 - Release should be the destination. Rigorous build, test, and rollback practices have a broad impact on individual performance measures and overall performance. Change tracking and change oversight practices do not predict performance variation and may be necessary but not sufficient to achieve the highest levels of performance.

Change management is often identified as a logical starting point for ITIL implementations. However, release management should be the destination for those organizations wanting to achieve performance gain from standardizing on ITIL change and release practices.

Widespread use of release practices is the top predictor of the highest levels of performance across 8 of the 15 performance measures in this study.

2 - Process discipline matters. There are no change, configuration, and release silver bullets. IT organizations that actively encourage compliance with documented processes and procedures and that manage process variation achieve higher levels of performance. ITIL change, configuration, and release best practices are not likely to deliver expected results when implemented by an organization that does not have a culture or history of following documented processes and procedures.

Building a process-focused culture and managing process exceptions predicts top levels of performance across measures such as downtime, configuration drift, process variability, release impact rate, and change success rate—more than many of the industry-recognized best practices in these areas.

3 - Standardized configuration works. Practices related to using only approved production system configurations, providing IT personnel information about the approved and current configuration, and detecting configuration changes are primary predictors of a stable and secure computing environment. System-to-system variation grows exponentially if specific controls and practices are not implemented to reduce variation and prevent configuration drift.

Following a simple set of Standardized Configuration practices is an effective way to stabilize and secure a complex and dynamic computing environment.

4 - Controlled production access improves performance. The combination of defined roles and responsibilities, clear separation of duties, and reduced access to the production environment predicts higher levels of performance. Integrating development into an end-to-end development and release cycle and having cross-functional phase gates are more aspirational practices in the industry. Some of the leading-edge companies we interviewed highlighted integrated development efforts, but these are not as widely practiced in the greater survey population.

Limiting access to production systems and clearly separating development, test and build, and release activities are effective ways to reduce process variability and reduce the number of emergency changes, and increase the security posture of the organization.

5 - CMDB enables powerful practices. The use of a CMDB is not yet widespread. Only 19% of survey respondents have a CMDB broadly in use. However, CMDB-enabled change linkage practices predict higher levels of performance. Only 47% of top-performers in the study have CMDB-enabled change practices, such as linking change requests to infrastructure, business need, and incident tickets. Yet these practices are a statistically significant predictor of top levels of performance.

Linking change requests to a system-level and business-level context is a powerful way to reduce release rollback rate, reduce configuration drift, and improve incident response.

Top-Performer Profile

What emerged from the interviews and the survey analysis was a simple snapshot of how top-performing IT shops approach managing risk and performance in a business-critical production environment:

They maintain production systems in a known, tested, risk-reduced state. They standardize configurations to reduce complexity and improve scalability and supportability. They update production systems from golden-build configurations to minimize risk. And, they monitor for configuration drift and unauthorized changes to keep systems in a known good state.

They minimize access to production systems. They recognize that small changes have a big potential impact, and they remove development access to the production environment. They clearly define roles and responsibilities and separate the duties of development, test and build, and production release.

They allow modifications to the production environment only through a carefully controlled process. Every production modification is recognized as an availability and security risk. Each release meets build requirements, including documentation and support instructions. Releases are tested. Backout plans are tested. Releases are scheduled during maintenance windows and considered a failure if they do not exactly follow release instructions or are not completed on time. Release failures and process exceptions undergo root cause analysis to identify improvements that reduce process variation.

They use executive influence and human-resources practices to build a process oriented culture. Following documented processes and procedures is recognized as a basic job expectation. They focus on process exceptions as a way to identify the cause of variance and to identify process improvement efforts. IT executives actively participate in monitoring process variance and process improvement efforts.

Overall, this set of focus areas common to top-performers contribute to a systematic way to achieve the highest levels of performance. In top-performing IT organizations, consistently high and predictable performance is not dependent on individual preferences or skills, and can be achieved across locations and business units.

Performance Improvement Potential

Study participants were segmented based on their overall performance score as indicated by how many of the performance measures they scored in the top 50th percentile of all respondents.

Highlights of the striking performance differentials for top-performers in some of the most important measures include:

- **Downtime minutes per month**—The pre-release testing and process culture key performance drivers impact this measure. Top-performers average 30 minutes, which is 60% lower than medium performers and 67% lower than low performers.

- **Security breaches automatically detected**—The standardized configuration strategy and controlled production access key performance drivers impact this measure. Top-performers auto-detect an average of 91% of security breaches, which is 40% higher than medium performers and 51% higher than low performers.

- **Release impact rate**—The process culture key performance driver impacts this measure. Top-performers have an average 2.9% of releases that cause a service outage or incident, which is 48% lower than medium performers, and 74% lower than low performers.

- **Incidents fixed within SLA limits**—The process culture and change linkage key performance drivers impact this measure. Top-performers fix 93% of incidents within service level agreement (SLA) limits, which is 16% higher than medium performers and 43% higher than low performers.

Summary of chapters

The remainder of this report will detail the analysis and findings revealed by this study.

Chapter 2 – Includes an overview of the major steps of analysis that were used to reveal the findings. This helps you understand how to interpret study claims and apply the findings to their organization.

Chapter 3 –Summarizes the five key findings of this research, including lists of practices that do and do not predict top levels of performance for the IT organizations in the study.

Chapter 4 - Provides a detailed comparison of the 15 performance measures and 15 firmagrahic markers for the top, medium and low performers in the study. This data can be used to benchmark your organization to identify performance improvement opportunities.

Chapter 5 – Provides summary and conclusion from this study.

Appendix A – Offers a quantitative assessment of common practices identified from the executive interviews.

Appendix B – Provides a list of practice questions used in the survey, showing the average scores for top, medium and low performing organizations in the study.

Appendix C – Highlights survey respondent demographics.

Appendix D – Outlines the statistical analysis used to identify twelve groups of common practices.

Appendix E – Outlines the statistical analysis used to identify the key performance drivers that predict top-levels of performance.

Appendix F – Highlights the thirty practices in seven key areas with a summary of how broadly the top, medium and low performers in the study have these practices implemented.

Appendix G – Provides a cross reference of measures and key performance drivers. This can be used to benchmark your organization and identify practices you can implement to improve performance in specific areas.

2 – Study Overview

The IT Process Institute conducted a study of IT Change, Configuration and Release management practices and their impact on IT performance. This study builds on the findings of our previous research that identified these practice areas as differentiators of top performing IT organizations[1]. We used a two-phased approach of interviews to collect qualitative data, and web-based survey for quantitative data collection. We followed recognized statistical analysis techniques to identify sets of practices that are commonly implemented together, and reveal those practices predict different levels of performance in the organizations studied.

Study Objectives

We conducted this study to meet several objectives including:

1. Identify specific change, configuration and release practices that best predict performance variation across a wide sample of IT organizations. In other words – out of all the hundreds of industry best practices and recommended practices – which actually improves measurable performance?
2. Determine if other organization, management and process enablers are required to achieve performance improvement related to best practice.
3. Quantify performance improvement potential across top, medium, and low performance groups in the study.

Summary of Interviews

We started with interviews of 11 organizations recognized for their data center operational excellence. We used a structured open-ended interview process to ask a series of questions about change, configuration and release practices, and organization management and process enablers. We focused on current practices and past breakthroughs that contribute to top levels of performance. Interview data collection and analysis proceeded iteratively with early interviews being more open-ended, and later interviews being directed by five emerging themes of common practice. We also asked about key measures they used to gauge success in these areas.

A summary of the 11 companies interviewed include:

- Banking and financial services – 4 firms with employee count ranging from 35,000 to 110,000.
- IT hosted service providers – 2 firms one large with 2,500 employees, and one small with 15 employees.
- IT infrastructure management service providers – 3 firms that each managed internet or wireless IT infrastructure offered as their primary business ranging from 100 to 4,500 employees.
- Other – 2 firms including a mining company, and a software provider both with 4,500 employees.

[1] IT Process Institute, IT Controls Performance Study 2006

The five common themes of practices that the companies we interviewed identified as causing their high levels of performance include:

1. These firms have rigorous change and release processes in place that are designed to improve and optimize activities that may introduce risk into the production environment.
2. These firms use a standardized configuration strategy to reduce the number of different configurations in the production environment, and to only use production systems that are configured to tested and pre-approved specifications. Specific procedures are used to detect and respond to unauthorized changes and configuration drift.
3. Specific steps have been taken to restrict access to production systems so that only production personnel with clearly defined roles and responsibilities can introduce modifications to the production environment. In order to meet developers' needs, these organizations also purposefully integrate development and production operations teams and processes, to create an end-to-end continuum of development, test/QA and production release procedures.
4. There is an intense focus on process management and exception monitoring. These organizations look for exceptions and identify root cause and systemic fixes to build increasing levels of process predictability, capability, and performance.
5. Many of the firms indicated using a CMDB and other automation tools as enablers of performance breakthrough. Some of the organizations had a production CMDB. One key practice was to use relationship data to tie change requests to infrastructure components and specific business needs. The relationship data helped automate communication of change requests to appropriate IT functions and affected business users. Other organizations captured and used relationship data for similar purposes without a formal CMDB. Other organizations leveraged development to production build versioning and backout system to enable release practices.

Note - Appendix A provides a more detailed view of the practices identified in the interviews.

Qualitative data collection – web-based surveys

From the common themes that emerged from the interviews, we designed a web-based survey to collect data from a broader sample of organizations with a wide range of practices and performance results. We collected data on a range of 57 practices related to the five interview themes. Use of each practice was graded on a scale of 0 to 10 – with 0) does not apply at all 5) applies to some parts of my organization, and 10) applies consistently across my organization.

We also designed the survey to collect data about 14 key performance measures that were identified in the interviews.

Overall Performance

1. Down Time (minutes per month)
2. Server / SysAdmin (ratio)
3. Security breaches auto-detected (%)
4. Process Variability (1-7 scale)
5. Configuration Drift (1-5 scale)

Change Measures

6. Change Success Rate (%)
7. Emergency Change Rate (%)
8. Unauthorized Change Rate (%)

Release Measures

9. Release impact rate (%)
10. Release rollback rate (%)
11. Release exceptions root cause (%)

Incident Response

12. Mean time to repair large outage (minutes)
13. SLA fix rate (%)
14. Support to Development escalation rate (%)

We also collected data about 13 demographic and sizing markers for things like budgets and spending ratios, staff turn over and CIO tenure, and change and release rates.

Note – Appendix B provides details on all survey questions used, including practices, measures, and demographic markers. The questions are presented with the average scores for top, medium, and low performers in the study.

Survey participant demographics

We collected data in March and April 2007, and compiled a dataset with answers from 341 IT organizations from companies and government agencies greater than $100M in revenue, and primarily based in North America.

A broad range of industries are represented in the study population, with the top represented industries including Banking, Finance and Insurance (18%), Manufacturing – not high-tech (18%), Health Care (14%), Government (7%), Business Service (6%) and High tech manufacturing (5%), and Retail (4%).

A broad range of company revenues are represented with 23% between $100M and $250M, 32% between $250M and $1B, 28% between $1B and $10B, and 17% from companies with >$10B.

Individual respondents were primarily from senior management positions with 59% of respondents at IT Director, Vice President, or Executive level. Individual respondents represent a range of expertise including development, QA and test, production administration, production support, security, and others including audit.

Note – Appendix C provides details on the study participant demographics.

Identification of Key Performance Drivers

We used factor analysis, a data reduction technique, to isolate 12 sets of practices that are commonly implemented together by the organizations in the study population. We then used regression analysis to identify the factors (common sets of practices) that predict performance variation in each the 14 individual performance measures, as well as an overall performance index measures. We assessed the predictive value of each of the 12 common practices to identify those that had the biggest and broadest impact on the 14 performance measures and 1 overall performance index.

We identified 7 groups of practices that predict performance variation, that we identify as key performance drivers, include (listed in order of highest to lowest impact):

- **Release scheduling and rollback**—In this set of practices, IT organizations develop and maintain a fine-tuned cycle of build and test, and then release only during maintenance windows with tested rollback plans. Data about the root causes of release exceptions is fed back to systematically improve the process.
- **Process culture**—IT executives demonstrate that following a specific process is "how we do things here." The company's hiring, reward, and promotion policies reinforce the basic job expectation that people will follow documented processes and procedures.
- **Pre-release testing**—Before release, changes are tested in a pre-production environment that is maintained to be sufficiently similar to the production environment.
- **Standardized configuration strategy**—Configurations are standardized with golden builds and production systems are updated from an approved standard. IT personnel receive information about target and actual configuration. Systems are regularly monitored for unapproved changes or configuration drift.
- **Change linkage**—Change requests are linked to both infrastructure components and business service or business need. Support personnel are given access to change history to aid incident and problem management.
- **Controlled production access**—In this set of practices, IT organizations remove developer access to production; maintain well-defined roles; separate development, test, and release duties; and match users to appropriate system access permissions.
- **Process exception management**—In this set of practices, IT organizations place significant focus on eliminating process variation, including identifying exceptions, diagnosing root cause, and implementing fixes. Executives participate in monitoring exception rates and improvements.

We also identified 5 of the common practices that do not predict performance variation, including (not listed in any specific order):

- **Change tracking –** There is a defined process for managing application, infrastructure, and security related changes, as well as emergency changes. The process of routing different categories of changes is pre-defined, and the inputs and outputs of each process step are well understood.
- **Change oversight** – A change advisory board or cross-functional review committee reviews all non-emergency change requests. A Change Manager function provides additional scrutiny and oversight of requested changes, and release plans. All change requests are categorized by level of risk.
- **Multi-function phase gate** - Process check points are clearly identified at different points in the process, the requirements that need to be met to pass each check point are widely understood, and multiple functional groups within IT can include their requirements at the appropriate checkpoint (i.e. security, production, QA, test etc.)
- **Development integration –** An end-to-end lifecycle integrates development into the change and release processes. Organizational changes or specific HR related programs integrate development and production resources and activities. Systems are used to track and version software releases as they go through development, test/QA, build, release, and rollback.
- **CMDB with dependencies -** A CMDB describes the relationships and dependencies between configuration items (infrastructure components), and the change tracking system that identifies configuration items and upstream and downstream dependent systems that may be affected by a change request.

Regression analysis identified four individual practices that impact performance on their own. These were added to related key performance drivers to simplify findings, including (not listed in specific order):

- A high level of awareness at all levels of the organization, that seemingly small production changes can have a significant impact on key performance measures such as availability.
- IT executives regularly communicate that people are expected to follow documented process and procedures, and participate in enforcement of consequences for not following documented process and procedures.
- Monitor systems for unauthorized change in order to identify configuration drift.
- Developers are never given root access to production environments, but may get limited access to help support production.

Note – Appendix D details the statistics used to identify the 12 sets of common practices. Appendix E details the statistics used to identify the 7 key performance drivers.

Linking practice to performance

The following Table 1 summarizes the linkage between the measures, and the key performance drivers that predict top levels of performance for each measure. All predict performance variation at a statistically significant level, but those that have the highest predictive power are identified as primary.

Overall Performance Measures	
Top-Half Count (measure of overall performance)	
Release Scheduling and Rollback	Primary
Pre-Release Testing	Primary
Standardized Configuration Strategy	Secondary
Down time (minutes per month)	
Pre-Release Testing	Secondary
Process Culture	Secondary
Server / SysAdmin (ratio)	
Pre-Release Testing	Secondary
Security breaches auto-detected (%)	
Standardized Configuration Strategy	Primary
Controlled Production Access	Secondary
Process Variability (1-7 scale)	
Process Culture	Primary
Release Scheduling and Rollback	Primary
Pre-Release Testing	Primary
Controlled Production Access	Secondary
Configuration drift (1-5 scale)	
Standardized Configuration Strategy	Primary
Release Scheduling and Rollback	Primary
Change linkage	Primary
Process Exception Management	Secondary
Change Measures	
Change Success Rate (%)	
Process Culture	Primary
***Emergency Change Rate (%)**	
Controlled Production Access	Secondary
***Unauthorized change rate (%)**	
Pre-Release Testing	Primary
Release Scheduling and Rollback	Primary

Release Measures	
***Release impact rate (%)**	
Process Culture	Primary
***Release rollback rate (%)**	
Pre-Release Testing	Primary
Change linkage	Secondary
Release exceptions root cause (%)	
Release Scheduling and Rollback	Primary
Process Exception Management	Secondary
Process Culture	Secondary
Incident Response Measures	
***Mean time to repair large outage (min)**	
Process Culture	Secondary
SLA Fix Rate (%)	
Process Culture	Primary
Change linkage	Secondary
***Mean time to repair large outage (min)**	
Process Culture	Secondary
***Support to Development escalation rate (%)**	
	None

Table 1 – Practices that impact individual performance measures

Note – Appendix G cross references performance measures with key performance drivers.

3 – Analysis of Five Key Findings

The study identified five key findings based on analysis of the common themes that emerged from the interviews, coupled with statistical analysis of survey data that revealed the practices that do and don't predict performance variation.

1 – Release Should be the Destination
2 – Process Discipline Matters
3 – A Standardized Configuration Strategy Works
4 – Controlled Production Access Improves Performance
5 – CMDB Enables Powerful Practices

Release Should be the Destination

The first key finding is that release build, testing and rollback practices have broad impact on individual performance measures and overall performance, while change tracking and change oversight practices don't.

Release scheduling and rollback and pre-release testing key performance drivers are the most significant predictors of performance variation out of the seven key performance drivers, with an impact potential on 8 of the 15 performance measures studied, including: overall performance top-half count, downtime, server to sysadmin ratio, process variability, configuration drift, unauthorized change rate, release rollback rate, and release exception root cause rate.

Our interviews identified a variety of rigorous change and release practices that enable highest levels of performance. Many executive mentioned having change management in place for years, but just recently applying a higher level of rigor to change and release practices. They shared the concept of an end-to-end life cycle that starts with development and goes through QA and test, to release into the production environment, and beyond to user acceptance testing.

Conclusion:

Change tracking and change oversight practices are necessary but not sufficient to achieve performance improvement on their own.

They focus on a defined process with projects and changes categorized by size, risk impact, and workflow phase gates that are determined by the type of change. Many if the interviewees mentioned having a mix of process measures and outcomes measures, and having significant focus on identifying process exceptions and identifying and implementing root-cause fixes to improve process capability.

A specific area of focus was process exception management and root cause analysis for failed releases. Some organizations took every release exception to root cause, and applied significant resources to identify and implement systemic fixes. Finally, these organizations shared a practice of developing a detailed set of build deliverables needed to accept a request for production release.

Analysis of the survey data revealed, however, that release practices are the most statistically significant predictors of performance of all practices studied. As a comparison, change tracking and change oversight functions do not predict performance variation on any performance measures.

Our conclusion based on comparing survey analysis with interviews and commonly accepted industry best practice – change tracking and change oversight practices are necessary but not sufficient to achieve performance improvement on their own.

Change tracking and change oversight functions enable cross-organization communications and help create and environment of consistent processing of change requests. Change tracking and oversight also helps ensure appropriate measures are taken to consider the risk and impact of pending changes.

However, the release functions more directly impact activities that affect the actual modification of production systems. Release practices affect the moment modifications are made to production systems, but also activities that can impact production system quality, that occur farther back in development life cycle. Change request and oversight typically improve communication and oversight before release when many development and QA activities have already taken place. You can't inspect quality in.

Practices that predict performance variation

The top-performers in this study have specific release practices that include creating a set of specific build deliverables and release instructions, pre-release testing, rollback plan testing, and focus on implementation during a maintenance window.

Key Performance Driver - Release Scheduling and Rollback

- A defined process for building software releases to meet various production build requirements.
- A defined process for testing releases before moving into the production environment.
- A defined process for developing and testing rollback plans before release to the production environment.
- A defined process for scheduling changes and releases only during scheduled maintenance windows.
- Controls ensure that people with production access are sufficiently careful when making changes.
- IT executive management has set significant performance goals and process maturity goals for key change, release, and configuration processes.

Key Performance Driver - Pre-release Testing

- Changes are thoroughly tested before release.
- Maintain a test environment that is sufficiently similar to the production environment.

These two key performance drivers predict performance variation across a broad range of key measures, including:

- Top-half count (overall performance) – primary impact
- Downtime – secondary impact
- Server sys admin ratio – secondary impact
- Process Variability – primary impact
- Configuration drift – primary impact
- Unauthorized change rate – primary impact
- Release rollback rate – primary impact
- Release exception root cause – primary impact

Common Practices that did not predict performance variation

The interviews revealed change routing and oversight functions that are considered industry best practice and that were common in the survey population as well, but did not predict performance variation for any measure in the study.

Change Process Routing

- A defined a process for managing application, infrastructure and security related change requests.
- A defined a process for handling emergency change requests.
- The process and routing of different categories of change requests is pre-defined, and all types of change requests have a standard routing.
- IT staff understand what criteria determines change request process routing.
- The inputs and outputs of each process step are well defined, and everyone involved in the process understands their role and procedures, and what other's roles and procedures are. (i.e. people know what to do and when to do it).

Change Oversight

- A change advisory board or cross-functional review committee reviews all non-emergency change requests. This group both verifies requirements are met prior to implementation, and verifies results after implementation.
- A Change Manager function provides additional scrutiny and oversight of requested changes, and release plans.
- All change requests are categorized by level of risk.

We also found that top-performers use a more rigorous definition of change success rate than medium and low performers. We asked study respondents to identify a definition of change success rate the most closely matches the one used in their organization.

Question - Which of the following best describes your definition of "change success rate"?
Function - Percent of changes that were implemented and met functional objectives.
Function and Time - Percent of changes that met functional objectives and were completed during planned time.
Function, Time, Procedures - Percent of changes that met functional objectives and were completed during planned time, and actions exactly followed build instructions.

As seen in Figure 1 what we found is that top-performers are more likely to use the more stringent change success rate definition that includes proper function, implementation during release window, and release activities that exactly followed release instructions. Whereas low performers more frequently use a definition of change success rate that measures just the proper function of the change.

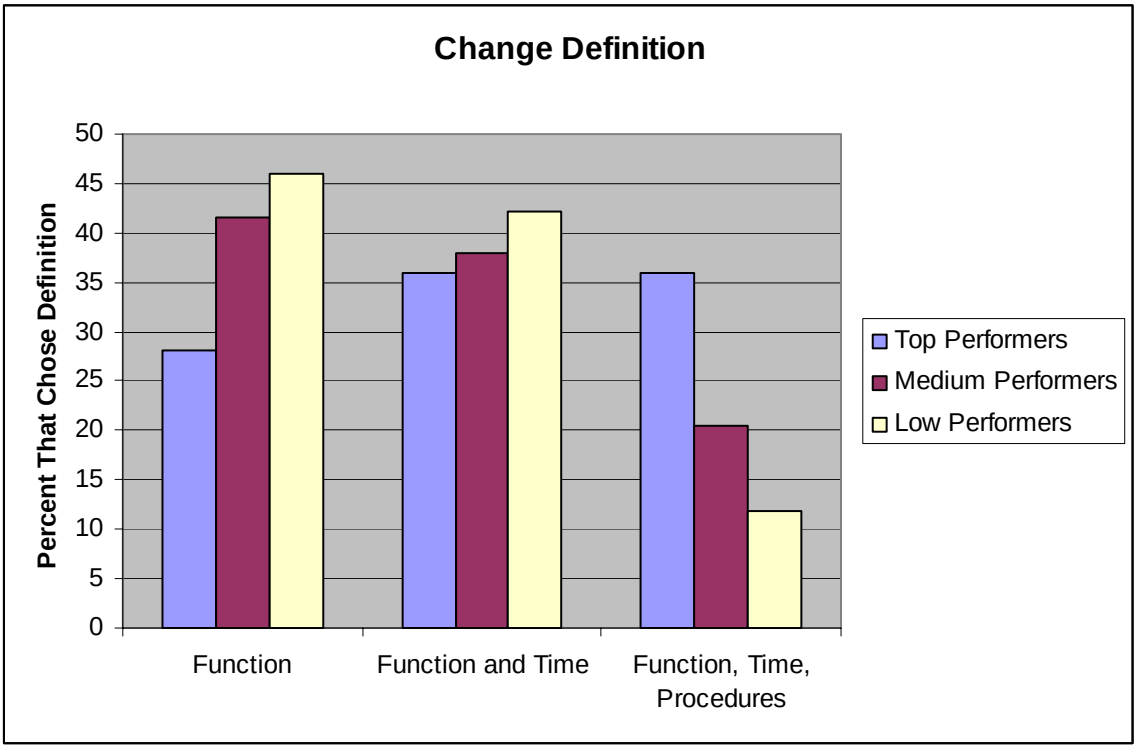

Figure 1 – Top, Medium and Low performer change success rate definitions

Interestingly, the top-performers, even though they are more likely to use the more stringent definition of change success rate, reported higher change success rate.

Question: What is your organization's average change success rate?		
Top Performers (average)	Medium Performers (average)	Low Performers (average)
96.4	92.5	81.3

The fact that top-performers have a more stringent definition of change success rate, and also score higher on the measure, helps illustrate the value of the release practices in this study.

Take Away – Focus on Release

The study findings paint a picture of top-performing IT organizations that clearly define and meet pre-defined build requirements, test releases in a pre-production test environment, have rollback plans that are tested, and schedule releases during maintenance windows.

Top-performers in the study measure success as changes that had the intended affect, that occurred within scheduled time, and that exactly followed release procedures. Any variation that occurs during release is considered an exception that has potential of introducing unknown levels of risk into production environment, and is cause for backing out the release, and pushing back to the build team for further testing.

Change management is often identified as a logical starting point for ITIL implementations.

However, release management should be the destination for those organizations wanting to achieve performance gain from standardizing on ITIL change and release practices.

An exemplary practice identified during one executive interview is that releases that deviate from build instructions that are backed out during a scheduled maintenance window are not counted as a failed change. This measure definition is designed to encourage people to not proceed with releases that may introduce unknown risk into the production environment.

Change management is often identified as a logical starting point for ITIL implementations. However, release management should be the destination for those organizations wanting to achieve performance gain from standardizing on ITIL change and release practices.

Process Discipline Matters

The second key finding is that process matters in IT. Following documented process and procedures, and monitoring and responding to process exceptions – are both strong predictors of top levels of performance.

Process culture and process exception management key performance drivers are the second most significant predictors of performance variation with an impact on eight of the fifteen performance measures including: downtime, process variability, configuration drift, change success rate, release impact rate, release exception root cause rate, mean time to repair large outages, and SLA fix rate.

Our interviews identified a variety of practices related to identifying and following "preferred" process and procedures that were identified as important drivers of performance. Many indicated that there was an executive level decision to follow a standardized process strategy. These organizations implemented a wide range of programs to boost process discipline and enhance employee accountability. These organizations indicated that processes were carefully designed, monitored and improved with checkpoints, process measures, and feedback mechanisms.

Analysis of the survey confirmed that these practices are the second most significant predictor of performance variation of all practices studied.

Conclusion:

Implementing change, configuration and release best practices without an organization wide commitment to procedural discipline is not likely to yield desired results.

Our conclusion based on interviews and study data is that there are no silver bullets. The intentional design and management of key change, configuration and release process and monitoring and responding to process exceptions predicts top levels of performance more than many of the industry recognized best practices in these areas. Implementing change configuration and release best practices without an organization-wide commitment to procedural discipline is not likely to yield desired results.

Top performing IT organizations in the study take specific actions to build a process culture, and have a specific focus on identifying and responding to process exceptions. The study data confirms the interview findings that many have made an intentional shift to staffing and promoting process oriented personalities that avoid a crisis - instead of rewarding heroic efforts that resolve a crisis. This is a departure from traditional technology focused cowboy culture of IT organizations in the past.

Identifying and responding to key-process exceptions seems to be the mark of exemplary organizations in this area. They share a primary focus on managing exceptions as a way to identify root-cause and fix systemic issues instead of responding to symptoms. The intense focus on identifying and correcting causes of process variation fits with both the audit mentality of "trust but verify" and the

continuous process improvement mentality of correcting systemic causes of process variation.

If an organization has identified a preferred way of completing high risk high impact procedures, and not everyone in the organization follows the procedures, then a wide range of results are likely.

If the IT organization isn't taking steps to implement a culture where compliance with specified operating procedures is the norm, then in a best case scenario – best practices become documentation on a shelf, and are not institutionalized. In a worst case scenario – IT organizations get varying and unpredictable results across location and business units. Without process discipline, performance is dependant on personnel preference, attention to detail, and skill level.

In order to fine tune a process to create a predictive path for modifications to the production environment, everyone needs to follow documented procedures!

Practices that predict performance variation

The top-performers in the survey population have specific process culture and process exception practices that predict their higher levels of performance.

Key Performance Driver - Process Culture

- Executive management clearly demonstrates through ongoing action, that following process is "how we do things here".
- Hiring criteria, performance reviews, promotion criteria clearly indicate that following defined process and procedures as a basic job expectation.
- It is widely understood that not following documented procedures is the exception.
- ** There is a high level of awareness that seemingly small production changes can have a significant impact on key performance measures such as availability.
- ** IT executives regularly communicate that people are expected to follow documented process and procedures, and participate in enforcement of consequences for not following documented process and procedures.

** These two practices predict performance variation as stand alone practices, and were added to this key performance driver for simplicity.

Key Performance Driver - Process Exception Management

- Monitor key processes to identify process exceptions.
- Have a defined process to analyze and diagnose the root cause of process exceptions.
- Have a defined process to analyze and diagnose the root cause of release exceptions.
- Practice an ongoing cycle of identifying specific causes of process variation, in order to identify and implement improvements.

- IT executives regularly review process exception rates and exception root-cause analysis as active participants and supporters of process improvement efforts.

Measures impacted by these key performance drivers:

- Downtime – secondary impact
- Process Variability – primary impact
- Configuration Drift – secondary impact
- Change Success Rate – primary impact
- Release Impact Rate – primary impact
- Release exception root cause – secondary impact
- MTTR large failures – secondary impact
- SLA fix rate – primary impact

Take Away – Process Matters

These findings help highlight for IT what Edward Deming knew about manufacturing in 1950 – management owns the process, and reducing process variability is a powerful way to improve predictability and capability. Our study analysis indicates that IT executive participation in ongoing efforts to assure compliance with key operating procedures predicts higher levels of performance.

A commitment to change, configuration and release process discipline is a remedy for IT executives who are tired of surprises, and want predictable performance and consistent results across geography, business unit, and individual skill level. Moving beyond a hero-worship culture to one that enables top-levels of performance from a wide range of skill levels requires a focus on process.

Change configuration and release best practices applied to an organization that does not have a culture or history of following documented process and procedures, is not likely to get desired results.

IT organizations that actively encourage compliance with documented process and procedure and managing process variation, achieve higher levels of performance. ITIL change configuration and release best practices applied to an organization that does not have a culture or history of following documented process and procedures, is not likely to get desired results.

Standardized Configuration Works

The third key finding is that having a small set of practices designed to standardize and control the configuration of production systems reduces configuration drift and increases security posture.

A range of practices related to specifying production system configuration, providing IT personnel about the approved and current configuration, and detecting configuration changes – are predictors of overall performance, configuration drift and security breach detection performance measures.

Executive interviews indicated that standardized and controlled configuration practices as enablers of a wide range of performance breakthroughs include reduce release effort, simplified admin and support, and reduced production and security risk. The standard configuration strategy was a common co-strategy with rigorous change and release practices. One organization indicted that when they started detecting changes, they didn't know what the target configuration was supposed to be, so they couldn't determine if the detected change was significant.

Most of these organizations we interviewed have some type of golden build strategy that reduces the complexity of the infrastructure, in order to improve supportability, improve service levels, improve security, and development/QA/test/release efficiency. One organization indicated that standardizing configurations for all similar server types reduced the effort required to update each server with a mission critical application from 9 man days to 2 hours because each target server was now identical. The same organization also saw a step function improvement in scalability, supportability, and security posture related to the shift to standard configurations.

The other benefit of standardizing and controlling configurations that emerged from the interviews was the idea of machines building machines. In order to completely standardize the infrastructure, one organization has all IT personnel sign an agreement that they will never load software from a CD. Only machines can provision or update production systems from a golden build reference. They take this strategy even farther requiring hardware vendors to agree to deliver identical systems for 2 years, down to the firmware level.

In addition, the interviews revealed widespread use of change detection solutions to monitor drift from target configuration, and to reconcile actual changes with approved changes in order to look for change process exceptions. Many of the organizations identified some form of configuration drift as the single most important security and operating measure. Their logic was that if they can't control the state of the production environment, they are operating with unknown levels of risk, both from a security and availability standpoint.

Analysis of the survey data confirmed that these practices are primary predictors of production configuration drift, and security breaches automatically detected measures. However, these practices did not predict performance variation across other measures,

even those it was clear from the interviews that a standardized configuration strategy was a key enabler of performance breakthroughs.

Some of the benefits of standardized configuration practices are difficult to measure in a survey. Benefits related to reduced release preparation effort, increased supportability, and improved scalability that were all clear from the interview, were not specifically tested in the survey. These types of improvements are more apparent from before and after comparison on a company by company basis.

Our conclusion is that the standardized configuration practices are proven to be an effective way to minimize configuration drift and increase security posture. We expect organizations to also achieve other efficiency related benefits from these practices, even though the study data did not confirm these interview findings.

Conclusion:

Controlled configuration practices are proven to be an effective way to minimize configuration drift and increase security posture.

If production systems are in an uncommon and unknown state as a result of different developers and administrators applying personnel preference to their set-up and configuration, then it logically follows that system-to-system variation causes a variety of issues related to managing a locked-down production environment. System-to-system variation causes unknown level of risk since each system is comprised of multiple layers of system software that each have different dependencies and capabilities. Unless each configuration variation has been through extensive testing, each minor variation may expose security or operating holes that result from layer to layer dependencies. Minor configuration variation may cause unpredictable performance variation. But most importantly, variations cause significant resource inefficiencies as release, administration, and support personnel can't count on similar systems being exactly the same, and therefore must spend time determining the "lay of the land" on each system before they can begin work.

Practices that predict performance variation

The top-performers in survey population have specific standardized configuration practices that predict their higher levels of performance.

Key Performance Driver - Standardized Configuration Strategy

- Production management strategy includes identifying specific approved configurations, or golden standard configurations, for systems in the production environment.
- Test and update the approved configuration or golden build, and only update production systems from that approved build.
- Provide IT personnel with accurate information about the gold standard, or approved configuration of each system in the production environment

- Provide IT personnel with accurate information about the current configuration of systems in the production environment.
- ** Regularly monitor systems for unauthorized changes and unintended configuration drift.

** This practice predicts performance variation as a stand alone practice, and was added to this key performance driver for simplicity.

Measures impacted by this key performance driver:

- Top half count – secondary
- Configuration drift – primary impact
- Security breaches auto detected – primary impact

Take Away – Standardized Configuration works

System to system variations grow exponentially if specific controls and practices are not implemented to reduce variation and prevent configuration drift. Practices that are designed to control configuration do reduce measurable drift, and improve security.

There are a wide range of other efficiency related benefits that may result from this approach. Moving beyond developer and administrator preference, to only using standardized and tested configurations, may significantly improve operating efficiency and allow organizations to move beyond linear scaling of administrative and support resources.

We expect organizations to achieve efficiency related benefits related to standardized configuration practices, even though our study data did not confirm the interview findings.

We expect organizations to also achieve other efficiency related benefits from these practices, even though the study data did not confirm the interview findings.

Following a simple set of standardized configuration practices is an effective way to optimize a complex and dynamic computing environment.

Controlled Production Access Improves Performance

The fourth key finding is that having defined roles and responsibilities, clear separation of duties, and reduced development access to the production environment together predict higher levels of performance. Having an integrated development process and rigorous process phase gates doesn't predict performance variation.

The controlled access key performance drivers predict variation in three of the measures studied including security breaches auto detected, process variability, and emergency change rate.

Our interviews identified that many organizations have restricted access to the production environment to minimize unauthorized and untested changes. In order to meet development, QA and test requirements, these organizations have also integrating development and production organizations into the end-to-end change and release lifecycle. In order to effectively manage the integrated process, they have clearly identified roles and responsibilities of who does what at different stages of the lifecycle.

Conclusion:

Controlled Production Access is instrumental in creating an environment of a stable and secure production environment with controlled modifications to the production environment.

In most of the organizations we interviewed, there were three major roles in the lifecycle: 1) developers develop, 2) another group builds, tests, and creates release packets, 3) another group makes modifications to production systems following release or build instructions. Each role has process measures related to their activities, and together they share overall process measures including quality, timeliness, and variability.

Analysis of the survey data revealed, however, that the controlled production access practices are statistically significant predictors of performance variation. As a comparison, the integrated development and phase gate practices did not predict performance variation.

Our conclusion is that the controlled production access key performance driver is instrumental in creating an environment of a stable and secure production environment with controlled modifications to the production environment. However integrating development and having cross-functional phase gates are more aspirational in the industry. These practices are highlighted by some of the leading edge companies interviewed. But not as widely practiced in the greater survey population.

The top-performers in the study have average of 82.5% of controlled access practices implemented at level 7 or higher. However, top-performers have an average of 55.1% of integrated development, and 58.3% of phase gate practices implemented at a level 7 or higher.

When access to the production environment is reduced, organizations need to clearly identify who is responsible for what and when, in order to continue to make required changes and not tempt people to skirt the change and release process. Separating responsibilities requires that everyone know what to expect from each others (inputs, outputs) and that everyone share responsibility for measuring and optimizing the end-to-end process.

Practices that predict performance variation

The top-performers in survey population, have specific practices that control production access, that predict their higher levels of performance.

Key Performance Driver - Controlled production access

- We have well defined roles and responsibilities for IT personnel.
- We have a defined process to map or match user accounts to an authorized user.
- Separation of duties is enforced and recorded for audit purposes.
- **Developers are never given root access to production environments, but may get limited access to help support production.

** This practice predicts performance variation as a stand alone practice, and was added to this key performance driver for simplicity.

Measures impacted by this key performance driver:

- Security breaches auto-detected – secondary impact
- Process variability – secondary impact
- Emergency change rate – secondary impact

With access reduced, roles defined, and duties separated, it is easier to detect irregular activity in the form of security breaches. These practices also help reduce process variability. When the roles are defined and the process is working, then there is less need for emergency changes.

Common Practices that did not predict performance variation

The interviews revealed development integration and multi-function phase gate practices that were common in the survey population as well, but did not predict performance variation for any measure in the study.

Development integration

- Have an end-to-end lifecycle that integrates development lifecycle into the change and release processes.
- Intentionally gather, implement, and review both development and production requirements in our end-to-end process.
- Have made organizational changes or created specific HR related programs in order to better integrate development and production resources and activities.
- Have a system to track and version software releases as they go through development, test/QA, build, release, and rollback.

- The causes of release exceptions are fed back to development to help improve the end-to-end process.

Multi-function phase gates

- Process check points are clearly identified at different points in the process, and the requirements that need to be met to pass each check point are widely understood.
- Multiple functional groups within IT can include their requirements at the appropriate checkpoint (i.e. security, production, QA, test etc.)

Take Away – Controlled Production Access Improves Performance

Controlled production access including defined roles and responsibilities and separating of duties is instrumental in creating an environment of a stable and secure production environment.

Integrating production and development and having cross-functional phase gates are logical ways to enable ongoing operations when access to production systems is restricted. They are important practices needed to create an end-to-end lifecycle. However, limiting access to production and having defined roles and responsibilities is what drives performance improvement, not the integrated lifecycle.

Our conclusion is that you can't have restricted access without an integrated lifecycle with well defined phase gates. But the primary focus should be on managing access and clearly defining who does what when.

You can't have restricted access without an integrated lifecycle with well defined phase gates.

But the primary focus should be on managing access and clearly defining who does what when.

CMDB Enables Powerful Practices

The fifth key finding is that the emerging use of CMDB in the industry is not widespread, but the change related practices enabled by a CMDB such as linking change to infrastructure, business need, and incident tickets, do predict higher levels of performance.

The change linkage key performance driver is a statistically significant predictor of performance variation with an impact on configuration drift, release rollback rate, and SLA fix rate.

Our interviews identified a majority of organizations using a CMDB with a high percentage of Configuration Items (CIs) tracked in the CMDB. Specific practices enabled by the use of the CMDB include using relationship data to tie change requests to infrastructure components and specific business needs. The relationship data helped automate communication of change requests to appropriate IT functions and affected business users. Other organizations captured and used relationship data for similar purposes without a formal CMDB.

Conclusion:

Linking change requests and change history to the IT infrastructure and business context is an important driver of higher levels of performance.

Only 19% of survey respondents have a CMDB broadly in use. However, analysis of the survey data reveals that the change linkage practices that are enabled by the CMDB do have a statistically significant predictive value for various performance measures. Only 48% of top-performers studied have change linkage capabilities at a level 7 or greater, but even with that low level adoption, the practices predict performance variation.

Conclusion: linking change requests and change history to the IT infrastructure and business context is an important driver of higher levels of performance.

Practices that predict performance variation

The top-performers in survey population have specific change linkage practices that predict their higher levels of performance.

Key Performance Driver - Change linkages:

- Identify CIs related to a change request in order to automate communications about pending and implemented changes.
- Link change requests to business need through CI relationships.
- Provide change history information to personnel managing incidents and problems.

Measures impacted by this key performance driver:

- Configuration drift – primary
- Release rollback Rate – secondary
- SLA fix rate – secondary

Common Practices that did not predict performance variation

The interviews revealed CMDB dependencies that were common in the survey population as well, but did not predict performance variation for any measure in the study. Only 33.3% of top-performers have a CMDB that describes relationships and dependencies, and change tracking system that utilizes that information.

CMDB with dependencies

- CMDB describes the relationships and dependencies between configuration items (infrastructure components).
- Change tracking system identifies Configuration Items (CI) in a CMDB, including upstream and downstream dependent systems that may be affected by a change request.

Take Away – CMDB Supporting Role

Only 19% of survey respondents have a CMDB. And only 47.4% of top-performers have CMDB enabled change linkage practices. However, the change linkage practices are a statistically significant predictor of top levels of performance in release rollback rate, configuration drift, and SLA fix rate.

Linking change requests to underlying infractructure and business need, and linking change history to incident tickets – all improve performance of key measures.

Although this study does not identify specific implementation practices that streamline the implementation and adoption of a CMDB, the change linkage practices that are enabled by a CMDB provide evidence that supports applying resources to CMDB related efforts.

Although this study does not identify specific implementation practices that streamline the implementation and adoption of a CMDB, the Change Linkage practices that are enabled by a CMDB suggest building support for CMDB related efforts.

4 – Performance Improvement Potential

We have identified seven key performance drivers including thirty individual practices that best predict top levels of performance. The overall implication of these findings is that organizations can implement these practices with the expectation that it will increase their performance.

We segmented 341 study participants into top, medium and low performance based on the top-half count overall performance measure. Our performance segmentation includes:

Segment	Top-half Count	Number of organizations	Percent of total
Low Performers	1 – 7	137	40%
Medium Performers	8 – 10	138	40%
Top Performers	11 – 14	65	19%

These segments are used to illustrate the difference levels of performance for these three groups for each of the performance measures used in the regression models.

Segmenting all study participants into low, medium, and top performance groupings also provides a powerful tool for organizations to assess their own performance improvement potential. Identifying the top-performers as the top 20th percentile of performers in the study provides a compelling target level of performance. Individual measure comparisons offer a simple benchmark for organizations looking to focus resource on specific key performance drivers that are shown to impact each measure.

Table 2 provides a summary of all measures used and the average response of top, medium and low performers in the study. Overall, segmenting the study population based on overall performance provides and effective way to identify differences in performance for individual measures.

Performance Measure	Top Performers	Medium Performers	Low Performers
Overall Performance	average		
Top Half Count	12	9	5
Down Time (minutes per month)	31	74	93
Server / SysAdmin (ratio)	57	37	33
Security breaches auto-detected (%)	91%	68%	60%
Process Variability (1-7)	5.9	5.2	4.1
Configuration Drift (1-5)	3.5	2.9	2.4
Change Measures	average		
Change Success Rate (%)	96.4%	92.5%	81.3%
Emergency Change Rate (%)	7.1%	12.7%	22.9%
Unauthorized change rate (%)	0.7%	3.2%	11.4%
Release Measures	average		
Release impact rate (%)	2.9%	5.6%	11.1%
Release rollback rate (%)	3.3%	3.8%	8.5%
Release exceptions root cause (%)	69.4%	50.5%	31.7%
Incident Response	average		
Mean time to repair large outage (min)	84	123	177
SLA fix rate (%)	92.9%	79.5%	65.2%
Support to Development escalation rate (%)	13.6%	20.0%	25.9%

Table 2 – Performance Measures

Note – a full list of measure definitions is included in Appendix B.

Highlights of the striking performance differentials for top-performers in some of the most important measures include:

- **Overall Performance Top-Half Count**—The release scheduling and rollback, pre-release testing, and standardized configuration strategy key performance drivers impact the overall performance of IT organizations as measured by the top-half count. Top-performers have an average top-half count of 12, as compare to medium and low performers with 9 and 5 measures in the top-half of all respondents.

- **Downtime minutes per month**—The pre-release testing and process culture key performance drivers impact this measure. Top-performers average 30 minutes, which is 60% lower than medium performers and 67% lower than low performers.

- **Security breaches automatically detected**—The standardized configuration strategy and controlled production access key performance drivers impact this measure. Top-performers auto-detect an average of 91% of security breaches, which is 40% higher than medium performers and 51% higher than low performers.

- **Configuration Drift**—The release scheduling and rollback, standardized configuration strategy, change linkage and process exception management key performance drivers all impact the amount of configuration drift in the production environment as measured on a 1 to 5 scale. The scale includes; 1= We do not identify approved configurations for production systems, 3= Most systems match their golden build, but we still find that some systems drift, 5= We go to great lengths to ensure that all production systems exactly match the desired configuration. Top-performers have an average score of 3.5, while medium and low performers average 2.9 and 2.4.

- **Unauthorized Change Rate**— the release scheduling and rollback and pre-release testing key performance drivers impact this measure. Top-performers have and average of 0.7% of changes that are not tracked as part of the normal change management process as compared to 3.2% and 11.4% for medium and low performers.

- **Release impact rate**—The process culture key performance driver impacts this measure. Top-performers have an average 2.9% of releases that cause a service outage or incident, which is 48% lower than medium performers, and 74% lower than low performers.

- **Incidents fixed within SLA limits**—The process culture and change linkage key performance drivers impact this measure. Top-performers fix 93% of incidents within service level agreement (SLA) limits, which is16% higher than medium performers and 43% higher than low performers.

Note – Figure 6 in Appendix E provides a good summary of how much of the variance of each measure is predicted by the key performance drivers in the study.

5 – Summary and Conclusions

The study identified a set of seven key performance drivers including thirty individual practices that are shown to predict top levels of performance. The study also linked key performance drivers to specific performance measures. The performance differential for top-performers is significant.

IT organizations can compare their use of key performance drivers and levels of performance to top, medium and low performing IT organizations in the study to identify focus areas for process improvement efforts.

The study findings confirm some widely held beliefs about change configuration and release practices such as that configuration management works, and the using a CMDB enhances the overall change and release process.

The study challenges other industry consensus about the primary focus on change management. The findings indicate that change management process tracking and oversight functions may be necessary but not sufficient to predict top levels of performance. Release management practices from build, test and production release including backout plans and release instructions have a broad and significant improvement potential on a wide range of IT operating performance measures.

The study also indicates that executive management and organizational culture relating to process discipline have a significant effect on performance, and can help IT executives normalize performance and predictability across business units and location.

Finally, the study reveals that controlling access to production and having well defined roles and responsibilities in development, test and quality assurance, and production administration functions, predict top levels of performance.

The two phase study approach is an effective way to find out what is working for a broad sample of IT organizations. Some of the interview findings were not supported by analysis of data from the broader study sample. Following a small sample of executive interviews with data collection from a larger sample is an effective way to add more rigorous analysis and empirical evidence to themes that emerge from the interviews.

IT executives can leverage these findings and empirical data about both use of change configuration and release practices, and performance levels – to justify process improvement projects, and estimate potential improvement of specific measures.

Appendix A – Interview Themes

We conducted 11 interviews in December 2006, and January 2007 with IT executives selected based on public reputation of having operational excellence or performance breakthrough. We used a structured open-ended interview process to ask a series of questions about change, configuration and release practices, and organization management and process enablers.

A summary of the 11 companies interviewed include:

- Banking and financial services – 4 firms with employee count ranging from 35,000 to 110,000
- IT hosted service providers – 2 firms one large with 2,500 employees, and one small with 15 employees.
- IT infrastructure management service providers – 3 firms that each managed internet or wireless IT infrastructure offered as their primary business ranging from 100 to 4,500 employees
- Other – 2 firms including a mining company, and a software provider both with 4,500 employees.

We followed a structured interview process and asked about both current practices that are responsible for operational success, and about specific enhancements that have led to performance breakthroughs. We also asked about key measures that are used to determine success and progress in these areas. Data collection, coding, and analysis proceeded iteratively with early interviews being more open ended, and later interviews being directed by emerging concepts.

Five key themes emerged from the interviews that created a picture of common practices that these leading organizations felt contributed to their high levels of performance.

1 – Rigorous change and release
These organizations have a rigorous change and release process. Many mentioned having change management in place for years, but just recently applying a higher level of rigor to change and release practices. They shared the concept of an end-to-end life cycle that starts with development and goes through QA and test, to release into the production environment, and beyond to user acceptance testing. They focus on a defined process with project and changes categorized by size, risk impact, and the flow has phase gates that are determined by the type of change. Many mentioned having a mix of process measures and outcomes measures, and having significant focus on identifying process exceptions and identifying and implementing root-cause fixes to improve process capability. A specific area of focus for exception management and root cause analysis was failed releases. Some organizations took every release exception to root cause, and applied significant resources to identify and

implement systemic fixes. Finally, these organizations shared a practice of developing a detailed set of build deliverables needed to accept a request for production release.

2 – Standard configuration strategy
The second theme that emerged from the interviews was that these organizations use a strategy to limit the number of unique approved configurations in the production environment. The standard configuration strategy was a common co-strategy with rigorous change and release practices. Most of these organizations have some type of golden build strategy that reduces the complexity of the infrastructure, in order to improve supportability, improve service levels, improve security, and development/QA/test/release efficiency. There was widespread use of change or configuration detection tools to monitor for drift. Many of the organizations identified configuration drift as a key measure.

3 – Rigorous process focus
The third theme that emerged from the interviews was related to identifying and following "preferred" process and procedures. Many indicated that there was an executive level decision to follow a standardized process strategy. These organizations implemented a wide range of programs to boost process discipline and enhance employee accountability. These organizations indicated that processes were carefully designed, monitored and improved with checkpoints, process measures, and feedback mechanisms.

4 – Integrated development
The fourth theme that emerged was the purposeful integration of the development and production organizations into the end-to-end change and release lifecycle. One organization moved development and production under a common reporting structure. One had development and production work together to design the end-to-end process and share setting and attaining goals. One took specific steps to feed release exceptions back to development to identify systematic process enhancement solutions. One created a QA center of excellence to foster a shared quality improvement culture. One used HR to conduct team-building exercises to build bridges between development and production.

5 – Use of CMDB and tools
In addition, many of the firms indicated using a CMDB and other automation tools as enablers of performance breakthrough. Some of the organizations had a production CMDB. One key practice was to use relationship data to tie change requests to both infrastructure components and specific business needs. The relationship data helped automate communication of change requests to appropriate IT functions and affected business users. Other organizations captured and used relationship data for similar purposes without a formal CMDB. Other organizations leveraged development to production build versioning and backout system to enable release practices.

Figure 2 summarizes the count of how many of the 11 companies indicated that the practice or enabler was a key contributor to improved performance.

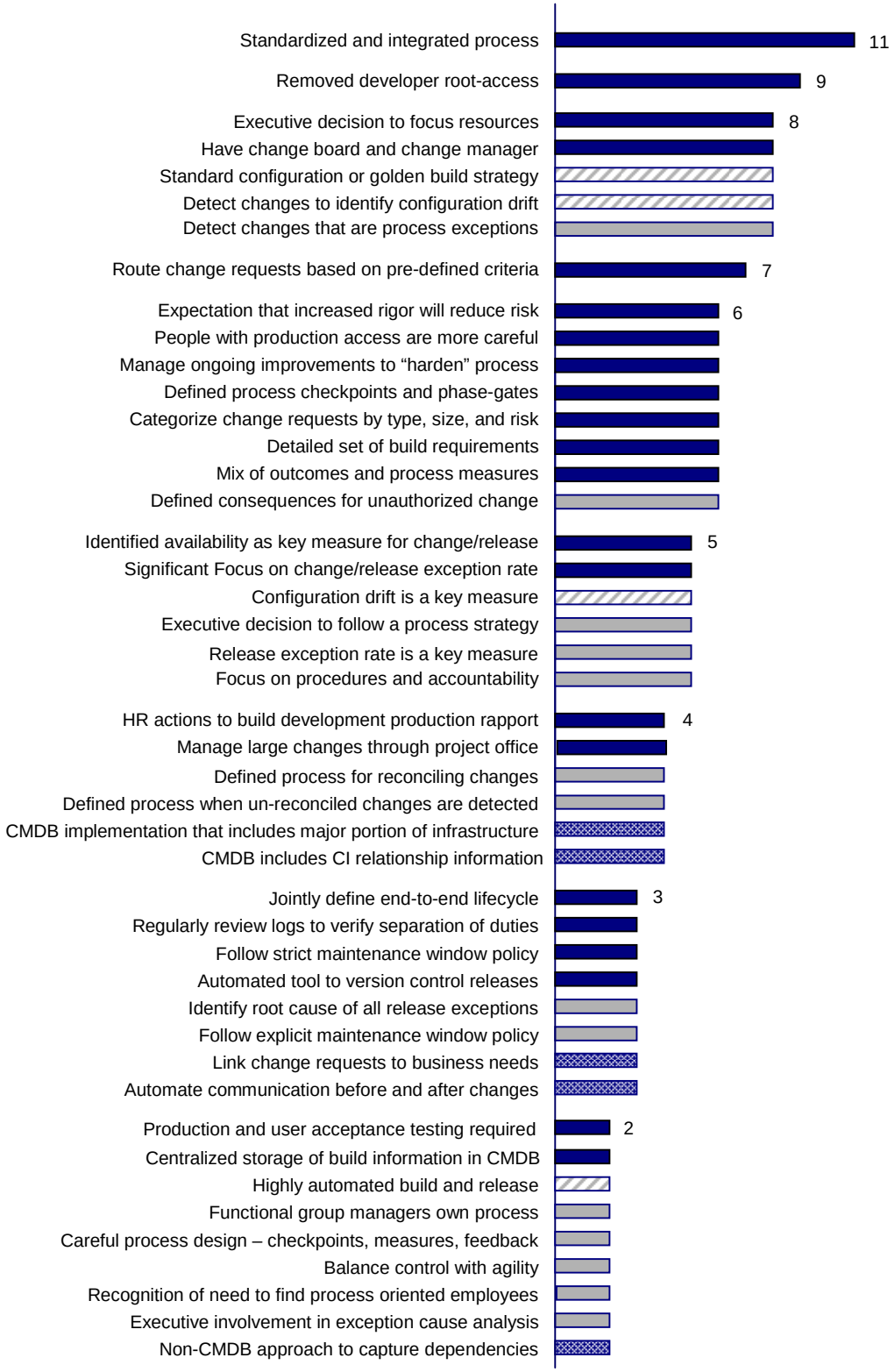

Figure 2 – Common practices identified during interviews

Appendix B – Survey Questions

The five themes of common practices revealed during the interviews, were used to develop survey questions in order to identify practices that predict top-levels of performance across a broad range of IT organizations. The survey we used to collect data about 57 individual practices, 14 performance measures, and 13 demographic and size markers.

Practice Questions

We asked 57 survey questions that were designed to gauge how widespread the practice is used in the organization.

All practices questions were answered on a zero to ten scale where

0 = Does not apply
5 = Applies to some parts of my organization
10 = Applies consistently across my organization

All 57 practice questions are shown below in Tables 3 through 11 sorted by general type of question. The questions are listed with the percentage of top, medium, and low performers in the study that answered the question at a level seven or higher. This presentation indicates how pervasive the individual practices are for the organizations identified as top, medium, and low performers.

Note - the level of use does not necessarily predict performance variation. There are some practices that are widely used by top-performers, and not widely used by bottom performers that do not predict performance variation. Conversely, there are some practices that are not widely used, or are used at a similar average level across performance groups that do predict variation in performance.

Questions with grey text are practices that do not have statistically significant impact on performance. Normal black text indicates practices that were identified as predicting top levels of performance, and are included in the seven key performance drivers that have statistically significant impact on performance.

See Appendix D and E for analysis that determined which practices predict performance.

Activities Linked to Risk	Top Performers	Med Performers	Low Performers
	% that scored 7-10		
** There is a high level of awareness at all levels of my IT organization, that seemingly small production changes can have a significant impact on key performance measures such as availability.	86.0%	68.6%	53.3%
We have implemented a wide range of change, configuration, and release procedures and controls in order to make people with production access more careful when making changes.	73.7%	65.0%	54.0%

IT executive management has prioritized and allocated resources to improve the rigor of change, release, and configuration processes, more than other processes.	66.7%	56.2%	52.6%
IT executive management has set significant process maturity goals and performance measure goals for key change, release, and configuration processes.	66.7%	48.9%	40.9%
Process Flows and Exceptions	**Top Performers**	**Med Performers**	**Low Performers**
	% that scored 7-10		
There has been an executive level decision to use a processes focused approach to managing key IT processes as a way to improve consistency of practice and predictability of results, or to achieve higher service levels at lower cost.	75.4%	62.0%	54.0%
We have centralized or standardized processes in order to improve consistency of practice and predictability of results across dispersed IT organizations or business units.	73.7%	62.0%	56.2%
The process and routing of different categories of change requests is pre-defined, and all types of change requests have a standard routing.	84.2%	60.6%	50.4%
IT staff understand what criteria determines process routing.	71.9%	55.5%	45.3%
The inputs and outputs of each process step are well defined, and everyone involved in the process knows what their role and procedures are, and what others roles and procedures are.	64.9%	51.8%	40.9%
Process check points are clearly identified at different points in the process, and the requirements that need to be met to pass each check point are widely understood.	61.4%	40.1%	36.5%
Multiple functional groups within IT can include their requirements at the appropriate checkpoint (i.e. security, production, QA, test etc.)	63.2%	46.0%	43.1%
We continuously monitor key processes to identify process exceptions.	59.6%	46.7%	38.0%
We have a defined process to analyze and diagnose the root cause of process exceptions.	63.2%	37.2%	33.6%
We practice an ongoing cycle of identifying specific causes of process variation, in order to identify and implement improvements.	56.1%	38.0%	39.4%
We regularly allocate resources to improvement projects that are identified as top priority.	66.7%	54.0%	48.9%
IT executives regularly review process exception rates and exception cause analysis as active participants and supporters of process improvement efforts.	45.6%	38.0%	35.0%
Process Culture	**Top Performers**	**Med Performers**	**Low Performers**
	% that scored 7-10		
It is widely understood that people in our organization are expected to follow documented process and procedures as a normal way of doing things, and that not following documented procedures is the exception.	87.7%	66.4%	56.9%
Hiring criteria, performance reviews, promotion criteria clearly indicate that following defined process and procedures as a basic job expectation.	61.4%	52.6%	46.0%
Executive management clearly communicates expectations and demonstrates through ongoing action, that following process is “how we do things here”.	66.7%	59.9%	48.9%

** IT executives regularly communicate that people are expected to follow documented process and procedures, and participate in enforcement of consequences for not following documented process and procedures.	54.4%	50.4%	37.2%
We have defined consequences for intentional unauthorized change.	50.9%	43.8%	39.4%
Roles and Responsibilities	**Top Performers**	**Med Performers**	**Low Performers**
	% that scored 7-10		
We have well defined roles and responsibilities for IT personnel.	93.0%	71.5%	65.7%
We have a defined process to map or match user accounts to an authorized user.	89.5%	72.3%	67.2%
Separation of duties is enforced and recorded for audit purposes.	73.7%	54.0%	45.3%
**Developers are never given root access to production environments, but may get limited access to help support production.	73.7%	47.4%	47.4%
Integrated Development	**Top Performers**	**Med Performers**	**Low Performers**
	% that scored 7-10		
We have an end-to-end lifecycle that integrates development lifecycle into the change and release processes.	57.9%	38.7%	35.8%
We intentionally gather, implement, and review both development's requirements and production's requirements in our end-to-end process.	56.1%	43.1%	39.4%
We have made organizational changes or created specific HR related programs in order to better integrate development and production resources and activities.	43.9%	30.7%	28.5%
The causes of release exceptions are fed back to development to help improve development process.	54.4%	40.1%	32.8%
Change Practices	**Top Performers**	**Med Performers**	**Low Performers**
	% that scored 7-10		
We have defined a process for managing application, infrastructure and security related change requests.	87.7%	73.0%	65.7%
We have defined a process for handling emergency change requests.	87.7%	69.3%	63.5%
All change requests are categorized by type (application, security, infrastructure etc.)	78.9%	65.0%	63.5%
All change requests are categorized by size (small, medium, large etc.)	64.9%	50.4%	54.7%
All change requests are categorized by level of risk.	73.7%	48.2%	46.0%
We have a change advisory board or cross-functional review committee that reviews all non-emergency change requests. This group both verifies requirements are met prior to implementation, and verifies results after implementation.	56.1%	42.3%	42.3%
We have a Change Manager function that provides additional scrutiny and oversight of requested changes, and release plans.	52.6%	40.1%	45.3%
Release Practices	**Top Performers**	**Med Performers**	**Low Performers**
	% that scored 7-10		
We have a system to track and version software releases as they go through development, test/QA, build, release, and rollback.	63.2%	48.2%	41.6%
We have a defined process for building software releases.	77.2%	56.2%	43.8%
We have an automated process for building software releases.	45.6%	32.8%	28.5%
We maintain an identical testing and production environment.	77.2%	59.1%	57.7%
We have a defined process for testing releases before moving into the production environment.	78.9%	71.5%	59.9%

Changes are thoroughly tested before release.	73.7%	62.0%	55.5%
We have a defined process for scheduling changes and releases, and changes are made only during scheduled maintenance windows.	78.9%	70.8%	60.6%
We have a defined process for developing and testing rollback plans before release to the production environment.	71.9%	57.7%	45.3%
We categorize and track the frequency and cause of release exceptions because we recognize that these exceptions introduce unknown levels of risk into the production environment.	49.1%	38.7%	36.5%
We follow a defined process to analyze and diagnose the root cause of release exceptions.	56.1%	42.3%	35.8%
Configuration Practices	**Top Performers**	**Med Performers**	**Low Performers**
	% that scored 7-10		
Our overall production management strategy includes identifying specific approved configurations, or golden standard configurations, for systems in the production environment.	59.6%	54.0%	48.2%
We provide IT personnel with accurate information about the gold standard, or approved configuration of each system in the production environment.	66.7%	51.1%	43.8%
We provide IT personnel with accurate information about the current configuration of systems in the production environment.	78.9%	59.1%	48.9%
** We monitor systems for unauthorized changes in order to identify configuration drift.	66.7%	46.0%	44.5%
We actively respond to unauthorized changes or configuration drift to restore systems back to an authorized state.	63.2%	52.6%	43.1%
We update our approved configuration or golden build, and only update production systems from that approved build.	56.1%	50.4%	40.9%
Use of CMDB	**Top Performers**	**Med Performers**	**Low Performers**
	% that scored 7-10		
We have a CMDB that describes the relationships and dependencies between configuration items (infrastructure components).	33.3%	19.7%	13.9%
We have a change tracking system that identifies Configuration Items (CI) in a CMDB, including upstream and downstream dependent systems, that may be affected by a change request.	33.3%	23.4%	17.5%
We provide change history information to personnel managing incidents and problems.	54.4%	37.2%	32.1%
We can link change requests to business need through CI relationships.	42.1%	27.7%	29.2%
We identify CIs related to a change request in order to automate communications about pending and implemented changes.	45.6%	27.0%	27.7%

Table 3 – Survey best practice questions with average responses

Measure Questions

We asked 14 measure questions which are listed below starting with a combined performance score called the top-half count. The definition of each measure is listed below:

We also calculated an overall performance index called the top-half count, which is a count of the measures where the respondent scored in the top 50th percentile of all respondents. There are fourteen measures, so the top-half count range of potential scores is from zero to fourteen.

Measure	Survey Question	Top Performers	Med Performers	Low Performers
		Average		
Top Half Count	The number of measures that you scored in the top 50Th percentile of all survey respondents.	12	9	5
Down time (minutes per month	The overall availability measure for your organization, in terms of minutes of unplanned down time per month. (i.e. downtime that is outside of planned maintenance windows) This measure should be representative of a mission critical system.	31	74	93
Server / SysAdmin (ratio)	Ratio of answers to two questions: 1) The number of component servers (i.e., servers, networks devices, firewalls, etc.) in your IT organization, including virtualized environments. 2) The number of IT administrator staff (FTE) that are responsible for ongoing monitoring and maintenance of these component servers.	57	37	33
Security breaches auto-detected (%)	The percentage of security breaches (both internal and external) that are automatically detected.	91%	68%	60%
Process Variability (1-7)	Answer on 1-7 scale, the question “How predictable and consistent is the output of key change, config and release processes?” where 1 = inconsistent results at a single location and across different locations, 7= consistent results at a single location and across different locations.	5.9	5.2	4.1
Configuration drift (1-5).	Answer to 1-5 scale question “Which of following best describes how well your production systems match their approved or golden build configuration? 1= We do not identify approved configurations for production systems 3= Most systems match their golden build, but we still find that some systems drift 5= We go to great lengths to ensure that all production systems exactly match the desired configuration.	3.5	2.9	2.4

Change Success Rate (%) -	Your organization's average change success rate. Note top-performers tend to use a more restrictive definition.	96.4%	92.5%	81.3%
Emergency Change Rate (%)	The percentage of changes considered "emergency changes?" In other words, these changes are tracked, but do not get standard review before they are implemented (for example, changes implemented before next weekly change meeting).	7.1%	12.7%	22.9%
*Unauthorized change rate (%) -	The percentage of changes that are unauthorized. In other words, these changes are made without being tracked by the standard change/release process. As an example, you detect changes and reconcile them with approved changes and identify changes that occurred but were not tracked or approved according to the standard process.	0.7%	3.2%	11.4%
*Release impact rate (%) -	The percentage of production releases cause a service outage or incident.	2.9%	5.6%	11.1%
*Release rollback rate (%)	The percentage of production changes in the last 12 months that were rolled back.	3.3%	3.8%	8.5%
Release exceptions root cause (%)	The percentage of these exceptions (failed changes) get root cause analysis to identify potential process improvements.	69.4%	50.5%	31.7%
*Mean time to repair large outage (min)	The average time to repair a severe and large incident or service outage? These are typically called "Severity 1 Incidents/Outages", and are defined as service issues that impact the most amount of end-users and requiring 6 or more IT staff to be mobilized to repair the issue.	84	123	177
SLA fix rate (%)	The average percentage of incidents or service outages fixed within SLA limits.	92.9%	79.5%	65.2%
*Support to Development escalation rate (%)	The percentage of incidents or help desk tickets that are escalated and require assistance from development resources.	13.6%	20.0%	25.9%

Table 4 – Survey measure questions with average responses

Demographic Comparison Questions

We also collected information about a range of other measures from the companies we studied which are not considered performance measures, but may provide helpful data for comparison purposes.

Performance Measure	Top Performers	Medium Performers	Low Performers
Demographics	25th to 75th percentile		
What is your company's or government agency's annual revenue? If you work for a government agency, what is your agency's annual operating budget? ($M)	350 – 5,000	300 – 4,000	230 – 5,000
How many employees does your company or government agency employ?	1,000 – 26,000	1,000 – 12,000	800 – 13,000
What is your company's or government agency's annual IT budget including both operating budget and capital budget? ($M)	4 - 100	3 – 50	2 - 40
How many employees work in your company's or government agency's IT department (Consider both employees and 3rd party service providers – e.g. contractors)	25 – 500	17 – 369	15 - 300
How many component servers (e.g., servers, network devices, firewalls, etc.) are in your IT organization?	80 – 1,200	46 – 638	25 - 500
Key Ratios	25th to 75th percentile		
IT Budget / Revenue (%)	0.067 – 2.67	0.60 – 3.24	0.62 – 3.00
Employees / IT Staff	12 - 83	22 - 87	20 - 77
* What percentage of production systems are recorded or tracked as CIs in the CMDB? (%)	78 – 100	37 – 97	15 - 80
Assume your total annual IT budget (operating and capital budgets) is divided into two parts 1) portion spent on running what is already in place 2) portion devoted to managing new projects. What percentage of your IT budget is devoted to new projects? (%)	21 - 50	22 - 42	25 - 50
Staffing	25th to 75th percentile		
How long has the current CIO (or highest ranking IT executive) been in their current role? (months)	10 - 72	7 - 72	7 - 60
What percentage of your IT staff turn over each year? (%)	2.5 – 10	3 – 10	5 - 10
Change and Release Rates	25th to 75th percentile		
What percentage of changes are tested before release to production? (%)	95 – 100%	85 – 100%	75 – 95%
On average, what percentage of releases have a tested rollback plan? (%)	40 – 97%	7 – 90%	10 – 50%
How many hours per week on average are scheduled for maintenance windows? (hrs)	2 – 8	2 – 8	2 - 8

* Question limited to those organizations that have a CDMB.

Table 5 – Survey sizing questions with average responses

Appendix C – Survey Demographics

We developed a web-based survey to collect data to verify the impact of various change, configuration and release practices on the key performance measures identified in the interviews. Surveys were conducted in March and April 2007 by GCR Insights, a custom research firm. The survey respondents are primarily from North American Based companies.

A broad range of industries are represented in the study population, with the top represented industries including Banking, Finance and Insurance (18%), Manufacturing – not high-tech (18%), Health Care (14%), Government (7%), Business Service (6%) and High tech manufacturing (5%), and Retail (4%).

A broad range of company revenues are represented as shown in Figure 3, with 23% less than $250M, 32% between $250M and $1B, 28% between $1B and $10B, and 17% from companies with >$10B.

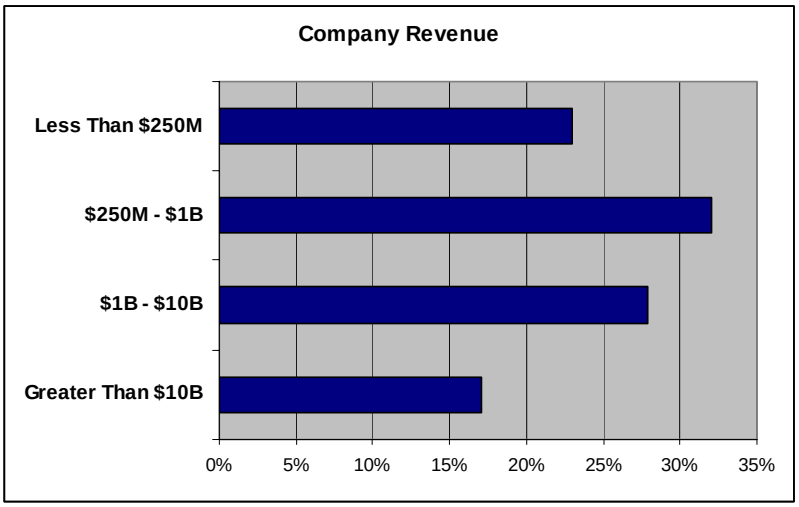

Figure 3 – Survey respondents by Company Revenue

Individual respondents were primarily from senior management positions as shown in Figure 4 with 59% of respondents at IT Director, Vice President, or Executive level.

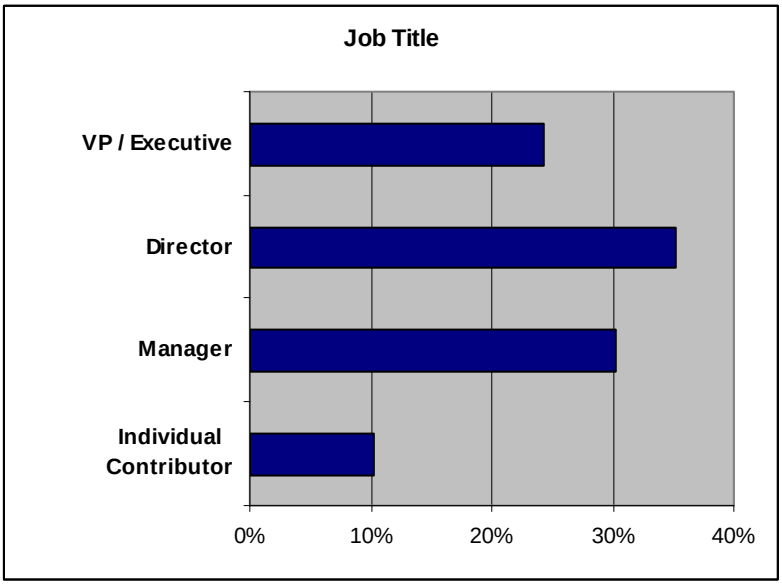

Figure 4 – Survey respondents by Job Title

Individual respondents represent a range of expertise as shown in Figure 5 including development, QA and test, production administration, production support, security, and other including audit.

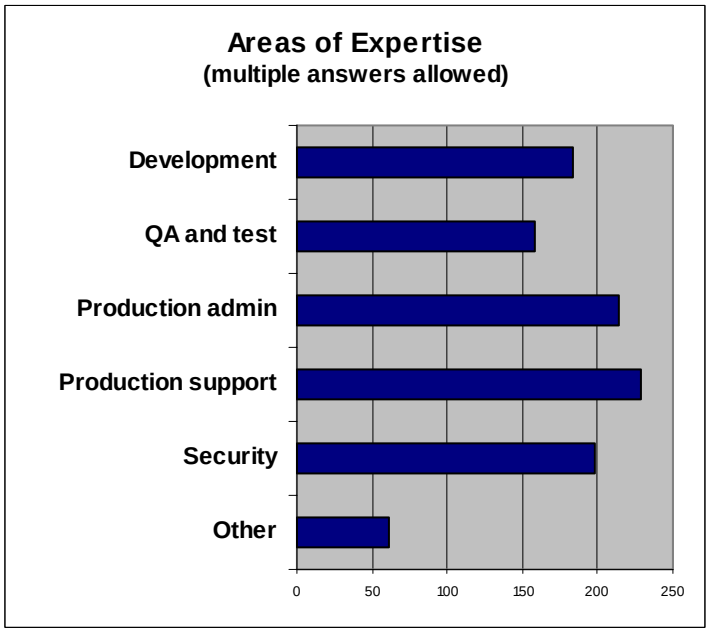

Figure 5 – Survey respondents by Area of Expertise

Appendix D – Statistical Analysis - Common practices

We conducted exploratory factor analysis to identify groupings of practices based on their common use. Factor analysis is a data reduction technique used to represent multiple observed random variables in terms of fewer derived or synthetic random variables called factors. Factor analysis originated in psychometrics, and is used in behavioral sciences, social sciences, marketing, product management, operations research, and other applied sciences that deal with large quantities of data.

All 57 practice variables were entered into principle axis factoring using varimax rotation with Kaiser normalization. Variables with low loadings (<0.40) were removed. The first factor solution included 8 primary factors all with loadings higher than 0.4. The three factors with the most variables were then factored again to sub-divided and create more granular sub-factors.

The results were the identification of twelve factors which represent a combination of 44 of the 57 individual practices. These twelve factors are synthetic variables that represent a combination of the practice questions for the survey respondents. Factor analysis identifies these groups of practices that are implemented in a similar way across participants. A helpful interpretation is that these sets of practices are commonly implemented together in the study population.

Although this analysis step doesn't identify which factors or individual practices predict performance variation, they do help identify sets of common practices.

The twelve sets of common practices (factors) and thirteen practices not included in the factors are listed below in no particular order. The list of practices in each factor is ordered from the most similar to the overall factor, to least similar.

Note – those individual practices with ** were later identified as predicting performance variation on their own, and manual added to the closest matching factors in order to simplify the analysis and application of findings. They are listed both as part of the factor, and in the list of individual practices that were not initially included in the factors.

Common Practice 1: Process exception management

- We have a defined process to analyze and diagnose the root cause of process exceptions.
- We practice an ongoing cycle of identifying specific causes of process variation, in order to identify and implement improvements.
- We continuously monitor key processes to identify process exceptions.
- IT executives regularly review process exception rates and exception cause analysis as active participants and supporters of process improvement efforts.
- We follow a defined process to analyze and diagnose the root cause of release exceptions.

Common Practice 2: Change to CI Linkage

- We identify CIs related to a change request in order to automate communications about pending and implemented changes.
- We can link change requests to business need through CI relationships.
- We provide change history information to personnel managing incidents and problems.

Common Practice 3: Change Process Routing.

- IT staff understand what criteria determines process routing.
- The inputs and outputs of each process step are well defined, and everyone involved in the process knows what their role and procedures are, and what other's roles and procedures are. (i.e. people know what to do and when to do it).
- The process and routing of different categories of change requests is pre-defined, and all types of change requests have a standard routing.
- We have defined a process for handling emergency change requests.
- We have defined a process for managing application, infrastructure and security related change requests.
- IT executive management has prioritized and allocated resources to improve the rigor of change, release, and configuration processes, more than other processes.

Common Practice 4: Change Oversight.

- We have a change advisory board or cross-functional review committee that reviews all non-emergency change requests. This group both verifies requirements are met prior to implementation, and verifies results after implementation.
- We have a Change Manager function that provides additional scrutiny and oversight of requested changes, and release plans.
- All change requests are categorized by level of risk.

Common Practice 5: Multi-function Phase Gate.

- Process check points are clearly identified at different points in the process, and the requirements that need to be met to pass each check point are widely understood.
- Multiple functional groups within IT can include their requirements at the appropriate checkpoint (i.e. security, production, QA, test etc.)

Common Practice 6: Release Scheduling & Roll-Back.

- We have a defined process for developing and testing rollback plans before release to the production environment.
- We have a defined process for building software releases.
- IT executive management has set significant performance goals and process maturity goals for key change, release, and configuration processes.
- We have a defined process for testing releases before moving into the production environment.
- We have a defined process for scheduling changes and releases, and changes are made only during scheduled maintenance windows.
- We have implemented a wide range of change, configuration, and release procedures and controls in order to ensure that people with production access are sufficiently careful when making changes.

Common Practice 7: Pre-Release Testing.

- Changes are thoroughly tested before release.
- We manage our testing environment so that it is sufficiently similar to the production environment.

Common Practice 8: Development Integration.

- We intentionally gather, implement, and review both development and production requirements in our end-to-end process.
- We have an end-to-end lifecycle that integrates development lifecycle into the change and release processes.
- The causes of release exceptions are fed back to development to help improve the end-to-end process.
- We have made organizational changes or created specific HR related programs in order to better integrate development and production resources and activities.
- We have a system to track and version software releases as they go through development, test/QA, build, release, and rollback.

Common Practice 9: Standardized Configuration Strategy.

- We provide IT personnel with accurate information about the gold standard, or approved configuration of each system in the production environment
- Our overall production management strategy includes identifying specific approved configurations, or golden standard configurations, for systems in the production environment.
- We provide IT personnel with accurate information about the current configuration of systems in the production environment
- We update our approved configuration or golden build, and only update production systems from that approved build.
- ** We monitor systems for unauthorized changes and unintended configuration drift.

Common Practice 10: CMBD with Dependencies.

- We have a change tracking system that identifies Configuration Items (CI) in a CMDB, including upstream and downstream dependent systems that may be affected by a change request.
- We have a CMDB that describes the relationships and dependencies between configuration items (infrastructure components).

Common Practice 11: Controlled production access.

- We have well defined roles and responsibilities for IT personnel.
- We have a defined process to map or match user accounts to an authorized user.
- Separation of duties is enforced and recorded for audit purposes.
- **Developers are never given root access to production environments, but may get limited access to help support production.

Common Practice 12: Process Culture.

- Executive management clearly communicates expectations and demonstrates through ongoing action, that following process is “how we do things here”.
- Hiring criteria, performance reviews, promotion criteria clearly indicate that following defined process and procedures as a basic job expectation.
- It is widely understood that people in our organization are expected to follow documented process and procedures as a normal way of doing things, and that not following documented procedures is the exception.
- ** There is a high level of awareness at all levels of my IT organization, that seemingly small production changes can have a significant impact on key performance measures such as availability.
- ** IT executives communicate that people are expected to follow documented process and procedures, and participate in enforcement of consequences for not following documented process and procedures.

Thirteen individual practices were not sufficiently similar to other practices included in the twelve factors:

- ** There is a high level of awareness at all levels of my IT organization, that seemingly small production changes can have a significant impact on key performance measures such as availability.
- ** IT executives communicate that people are expected to follow documented process and procedures, and participate in enforcement of consequences for not following documented process and procedures.
- ** We monitor systems for unauthorized changes and unintended configuration drift.
- **Developers are never given root access to production environments, but may get limited access to help support production.
- There has been an executive level decision to have a process oriented approach to managing IT as a way to improve consistency of practice and predictability of results, or to achieve higher service levels at lower cost.
- We have centralized or standardized processes in order to improve consistency of practice and predictability of results across dispersed IT organizations or business units.
- We have defined consequences for intentional unauthorized change.
- All change requests are categorized by type (application, security, infrastructure etc.)
- All change requests are categorized by size (small, medium, large etc.)
- We have a system to track and version software releases as they go through development, test/QA, build, release, and rollback.
- We categorize and track the frequency and cause of release exceptions because we recognize that these exceptions introduce unknown levels of risk into the production environment, and may cause SLA violations.
- We actively respond to unauthorized changes or configuration drift to restore systems back to an authorized state.
- We regularly allocate resources to improvement projects that are identified as top priority.

Appendix E – Statistical Analysis - Key performance Drivers

We used stepwise linear regression to measure how well the presence of the twelve factors and thirteen individual practices predict top levels of performance. Stepwise regression is a statistical method used to generate an estimate of how independent variables predict variations in a dependent variable. With stepwise regression, multiple independent predictive variables (common practices) are analyzed with a single dependent variable (performance measure), and the relationship of the set is determined by an algorithm.

In each step of the regression, variables are automatically added or removed from the set to eventually develop the set of predictive variables that best accounts for the greatest variance in the dependent variable. Overall, there are a wide range of things that impact performance that are not analyzed by this study. Regression analysis identifies factors that are statistically significant, and provide a gauge of how much of the variation in the measure is attributed to the variables in the model.

We performed stepwise regression analysis of twelve factors and thirteen individual practices against each of the 14 individual performance measures and 1 overall performance index. The result was 15 different regression models as shown in Table 14 without the individual variables shown.

The predictive strength of each of the 15 models is indicated by the adjusted r-squared value of the model. For example, the SLA fix rate has an adjusted r-squared value of 0.06. This can be interpreted to mean that 6% of the variance in this measure is predicted by the presence of the independent variables in the model. The process culture and CI linkage factors have r-squared values of 0.046 and 0.014 – which is interpreted to mean that the process culture practices predict 4.6% of the variation in the SLA fix rate, and the CI linkage practices predict 1.4% of the variation in the SLA fix rate.

R-squared values above 0.005 are statistically significant predictors of performance variance.

Fifteen regression models for each of 14 measures and overall performance top-half count. Primary predictors of performance variation are shaded in blue. Factors that have had an individual practice added in are indicated by *.

	Sum of r-squared across columns	Top Half Count (14)	Downtime minutes per month	Server to Sysadmin ratio	Security Breaches auto-detected	Process Variability	Configuration Drift	Change success rate	Emergency Change Rate	Unauthorized change rate	Release Impact Rate	Release rollback rate	Release Exception Root Cause	MTTR large outages	SLA Fix Rate	Support to Development escalation rate
		0.239	0.023	0.028	0.105	0.451	0.276	0.031	0.018	0.105	0.063	0.033	0.183	0.013	0.060	0.000
Release Sched Roll	0.424	0.14				0.077	0.058			0.024			0.125			
*Process Culture	0.433		0.011			0.27		0.031			0.052		0.01	0.013	0.046	
Pre-release	0.237	0.063	0.012	0.016		0.075				0.05		0.021				
*Standardized Config	0.238	0.018			0.041		0.179									
Change CI linkage	0.057						0.031					0.012			0.014	
*Controlled access	0.052				0.015	0.019			0.018							
Process Exception	0.029						0.014						0.015			
Phase Gate	0.019				0.019											
CMDB Dependencies	0.015									0.015						
Change Process	0															
Change Oversight	0															
Development integration	0															

Table 6 – Summary of regression models

Figure 6 shows the relative predictive value of each of the fifteen regression models. The strongest model was related to the process variability measure. 45% of the variability in this measure is predicted by the model. The weakest model was related to the support to development escalation rate. The model did not predict statistically significant variation in this measure.

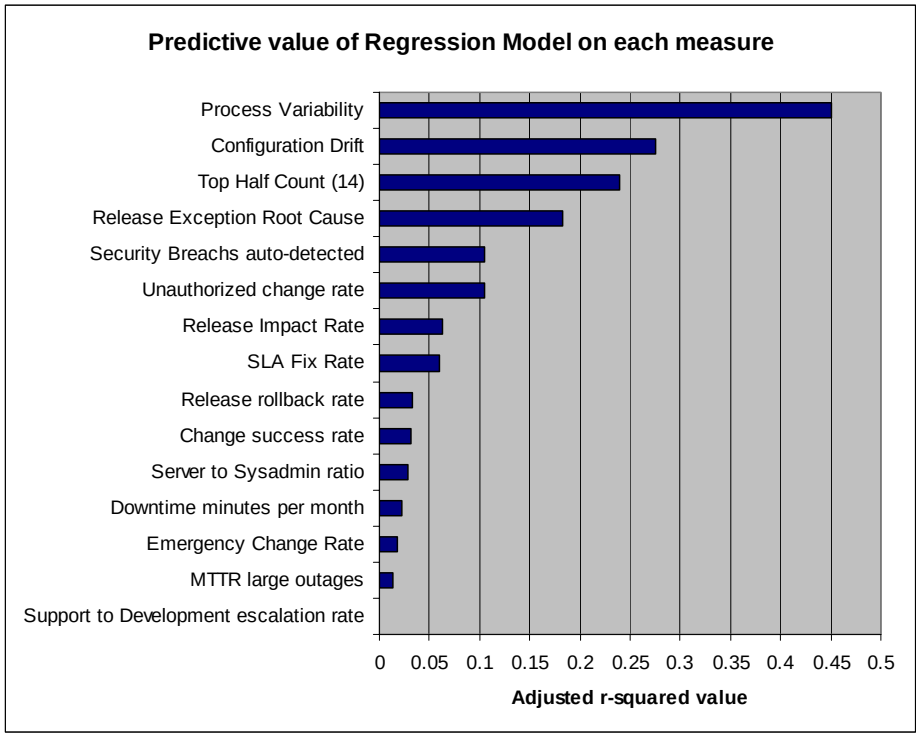

Figure 6 – Predictive value of Regression Models

Figure 7 shows the relative predictive value of each of the twelve factors. The factor with the strongest predictive value was process culture. The weakest factor that we considered a key performance driver was the process exception factor.

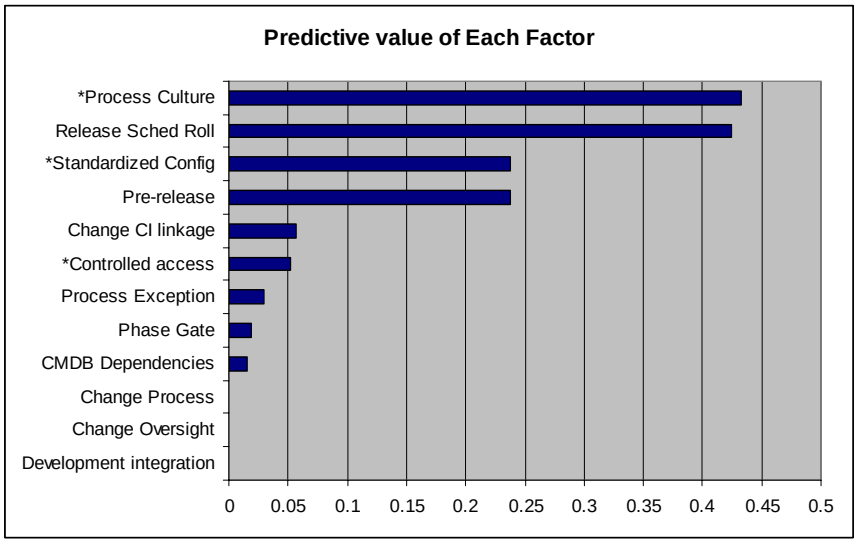

Figure 7 – Predictive value of Each Factor

These 15 models were analyzed to determine which practices best predict top levels of performance. The analysis of the 15 regression models included:

- Identifying primary and secondary predictors of variation. Those with r-squared at 0.02 and higher were considered primary, and are highlighted in blue in Table 14
- Sum the r-squared values across the 15 regression models, to create overall r-squared values for each factor, as seen in 1st column in Table 14 called sum of r-squared.
- Sort the factors (rows) based on overall r-squared values.
- Count the measures that have a primary impact by each factor.
- Prioritize factors with higher count of primary impact, over other factors with similar overall r-squared values. For example, Release scheduling and rollback has .424 r-squared sum, which is lower that process culture .433. However release scheduling and rollback has primary impact on a greater number of measures, including overall performance.

The resulting prioritized list of key performance drivers (factors) that best predict performance variation include:

1. Release scheduling and rollback
2. Process culture
3. Pre-release testing
4. Standardized configuration strategy
5. Change linkage
6. Controlled production access
7. Process exception management

Appendix F – Average Use of Key Performance Drivers

The following table summarizes the presence of each individual practice and total usage score for each of the seven key performance drivers.

Performance Driver	Top Performers	Med Performers	Low Performers
	% that scored 7-10		
1 - Release Scheduling & Roll-Back (total)	**74.6%**	**61.7%**	**50.7%**
We have a defined process for developing and testing rollback plans before release to the production environment.	71.9%	57.7%	45.3%
We have a defined process for building software releases.	77.2%	56.2%	43.8%
IT executive management has set significant performance goals and process maturity goals for key change, release, and configuration processes.	66.7%	48.9%	40.9%
We have a defined process for testing releases before moving into the production environment.	78.9%	71.5%	59.9%
We have a defined process for scheduling changes and releases, and changes are made only during scheduled maintenance windows.	78.9%	70.8%	60.6%
We have implemented a wide range of change, configuration, and release procedures and controls in order to ensure that people with production access are sufficiently careful when making changes.	73.7%	65.0%	54.0%
2 - Process Culture (total)	**71.2%**	**59.6%**	**48.5%**
** IT executives communicate that people are expected to follow documented process and procedures, and participate in enforcement of consequences for not following documented process and procedures.	54.4%	50.4%	37.2%
** There is a high level of awareness at all levels of my IT organization, that seemingly small production changes can have a significant impact on key performance measures such as availability.	86.0%	68.6%	53.3%
Executive management clearly communicates expectations and demonstrates through ongoing action, that following process is “how we do things here”.	66.7%	59.9%	48.9%
Hiring criteria, performance reviews, promotion criteria clearly indicate that following defined process and procedures as a basic job expectation.	61.4%	52.6%	46.0%
It is widely understood that people in our organization are expected to follow documented process and procedures as a normal way of doing things, and that not following documented procedures is the exception.	87.7%	66.4%	56.9%

3 – Pre-Release Testing (total)	**75.5%**	**60.6%**	**56.6%**
Changes are thoroughly tested before release.	73.7%	62.0%	55.5%
We manage our testing environment so that it is sufficiently similar to the production environment.	77.2%	59.1%	57.7%
4 – Standardized Configuration Strategy (total)	**65.6%**	**52.1%**	**45.3%**
** We monitor systems for unauthorized changes and unintended configuration drift.	66.7%	46.0%	44.5%
We provide IT personnel with accurate information about the gold standard, or approved configuration of each system in the production environment	66.7%	51.1%	43.8%
Our overall production management strategy includes identifying specific approved configurations, or golden standard configurations, for systems in the production environment.	59.6%	54.0%	48.2%
We provide IT personnel with accurate information about the current configuration of systems in the production environment	78.9%	59.1%	48.9%
We update our approved configuration or golden build, and only update production systems from that approved build.	56.1%	50.4%	40.9%
5 – Change Linkage (total)	**47.4%**	**30.7%**	**29.7%**
We identify configuration items (infrastructure components) related to a change request in order to automate communications about pending and implemented changes.	45.6%	27.0%	27.7%
We can link change requests to business need through CI relationships.	42.1%	27.7%	29.2%
We provide change history information to personnel managing incidents and problems.	54.4%	37.2%	32.1%
6 – Controlled Production Access (total)	**82.5%**	**61.3%**	**56.4%**
**Developers are never given root access to production environments, but may get limited access to help support production.	73.7%	47.4%	47.4%
We have well defined roles and responsibilities for IT personnel.	93.0%	71.5%	65.7%
We have a defined process to map or match user accounts to an authorized user.	89.5%	72.3%	67.2%
Separation of duties is enforced and recorded for audit purposes.	73.7%	54.0%	45.3%
7 – Process Exception Management (total)	**56.1%**	**40.4%**	**36.4%**
We have a defined process to analyze and diagnose the root cause of process exceptions.	63.2%	37.2%	33.6%
We practice an ongoing cycle of identifying specific causes of process variation, in order to identify and implement improvements.	56.1%	38.0%	39.4%
We continuously monitor key processes to identify process exceptions.	59.6%	46.7%	38.0%
IT executives regularly review process exception rates and exception cause analysis as active participants and supporters of process improvement efforts.	45.6%	38.0%	35.0%
We follow a defined process to analyze and diagnose the root cause of release exceptions.	56.1%	42.3%	35.8%

Table 7 – List of key performance drivers and average survey responses

Appendix G – Cross Reference Key Performance Drivers to Measures

The table below provides a convenient summary table that lists the key performance drivers that our regression model shows predict the variation of the specific measures. For each measure listed in bold, the average score for shown for top, medium and low performance groups is shown. For key performance drivers, the average number of participants that have the set of practices implemented at a level 7 or higher is also shown.

** average instead of range

Overall Performance		Top Performers	Med Performers	Low Performers
		% that scored 7-10		
Top Half Count (measure of overall performance)		**12**	**9**	**5**
Primary	Release Scheduling and Rollback	74.6%	61.7%	50.7%
Primary	Pre-Release Testing	75.5%	60.6%	56.6%
Secondary	Standardized Configuration Strategy	65.6%	52.1%	45.3%
Down time (minutes per month)		**31**	**74**	**93**
Secondary	Pre-Release Testing	75.5%	60.6%	56.6%
Secondary	Process Culture	71.2%	59.6%	48.5%
Server / SysAdmin (ratio)		**57**	**37**	**33**
Secondary	Pre-Release Testing	75.5%	60.6%	56.6%
Security breaches auto-detected (%)		**91%**	**68%**	**60%**
Primary	Standardized Configuration Strategy	65.6%	52.1%	45.3%
Secondary	Controlled Production Access	82.5%	61.3%	56.4%
Process Variability (1-7) **		**5.9**	**5.2**	**4.1**
Primary	Process Culture	71.2%	59.6%	48.5%
Primary	Release Scheduling and Rollback	74.6%	61.7%	50.7%
Primary	Pre-Release Testing	75.5%	60.6%	56.6%
Secondary	Controlled Production Access	82.5%	61.3%	56.4%
Configuration drift (1-5) **		**3.5**	**2.9**	**2.4**
Primary	Standardized Configuration Strategy	65.6%	52.1%	45.3%
Primary	Release Scheduling and Rollback	74.6%	61.7%	50.7%
Primary	Change linkage	47.4%	30.7%	29.7%
Secondary	Process Exception Management	56.1%	40.4%	36.4%

Change Measures		Top Performers	Med Performers	Low Performers
		% that scored 7-10		
Change Success Rate (%)		**96.4%**	**92.5%**	**81.3%**
Primary	Process Culture	71.2%	59.6%	48.5%
Emergency Change Rate (%)		**7.1%%**	**12.7%**	**22.9%**
Secondary	Controlled Production Access	82.5%	61.3%	56.4%
Unauthorized change rate (%)		**0.7%**	**3.2%**	**11.4%**
Primary	Pre-Release Testing	75.5%	60.6%	56.6%
Primary	Release Scheduling and Rollback	74.6%	61.7%	50.7%

Release Measures		Top Performers	Med Performers	Low Performers
		% that scored 7-10		
Release impact rate (%)		**2.9%**	**5.6%**	**11.1%**
Primary	Process Culture	71.2%	59.6%	48.5%
Release rollback rate (%)		**3.3%**	**3.8%**	**8.5%**
Primary	Pre-Release Testing	75.5%	60.6%	56.6%
Secondary	Change linkage	47.4%	30.7%	29.7%
Release exceptions root cause (%)		**69.4%**	**50.5%**	**31.7%**
Primary	Release Scheduling and Rollback	74.6%	61.7%	50.7%
Secondary	Process Exception Management	56.1%	40.4%	36.4%
Secondary	Process Culture	71.2%	59.6%	48.5%

Incident Response Measures		Top Performers	Med Performers	Low Performers
		% that scored 7-10		
Mean time to repair large outage (min)		**84**	**123**	**177**
Secondary	Process Culture	71.2%	59.6%	48.5%
SLA Fix Rate (%)		**92.9%**	**79.5%**	**65.2%**
Primary	Process Culture	71.2%	59.6%	48.5%
Secondary	Change linkage	47.4%	30.7%	29.7%
Support to Development escalation rate (%)		**13.6%**	**20.0%**	**25.9%**
None				

Table 8 – Measures cross-referenced to key performance drivers