Forrester Consulting

MAKING LEADERS SUCCESSFUL EVERY DAY

FORRESTER®

December 21, 2009

The Application Performance Management Imperative

A commissioned study conducted by Forrester Consulting on behalf of CA

FORRESTER®
Headquarters
Forrester Research, Inc., 400 Technology Square, Cambridge, MA 02139 USA
Tel: +1 617.613.6000 • Fax: +1 617.613.5000 • www.forrester.com

TABLE OF CONTENTS

Executive Summary

In October 2009, CA commissioned Forrester Consulting to evaluate the issues surrounding the management of composite applications built on mainframe systems and open-system middleware solutions. New business services marry core enterprise legacy data sets and applications to newer technologies, middleware, and browser interfaces. These services can reach a broad audience and are often mission critical. Service degradation or interruptions therefore have dire financial consequences.

In conducting in-depth interviews with 15 IT professionals with direct responsibility for these business-critical applications, Forrester found that all enterprises interviewed had fundamental issues while managing the performance of these applications. These issues come from the lack of effective management tools, the difficulty of correlating information from different sources, and, finally, the quasi impossibility of coordinating the investigations of teams that do not share the same information and are entrenched in "competency silos."

In correlating data from all interviewees, concrete themes surface regarding the offerings of an ideal application performance management solution: 1) the ability to monitor end user experience; 2) the ability to identify the components used in delivering composite applications; and 3) the ability to bring together data from all these components such that it could be shared by the issue investigation teams.

These IT professionals feel that such a solution, if it proved its capabilities, could be sourced from a single vendor and could largely replace the disparate tools that they are currently using.

Key Findings

Forrester's study yielded the following key findings:

- **Large enterprises use mainframe-based applications and data sets in mission-critical applications.** Composite applications use the core data of the enterprise, often located on a mainframe, and front-end these tools with new application servers, linking them through a variety of techniques.

- **The cost of downtime for these applications can reach into millions of dollars per hour.** Depending on the enterprise activity sector, the cost of an hour of downtime varies from $150,000 to several million dollars.

- **The applications are difficult to manage.** Because of monitoring disparities, it is difficult to bring all data together. This leads to problems in finding root cause and in coordinating investigations among different competency teams, all looking exclusively at their own data.

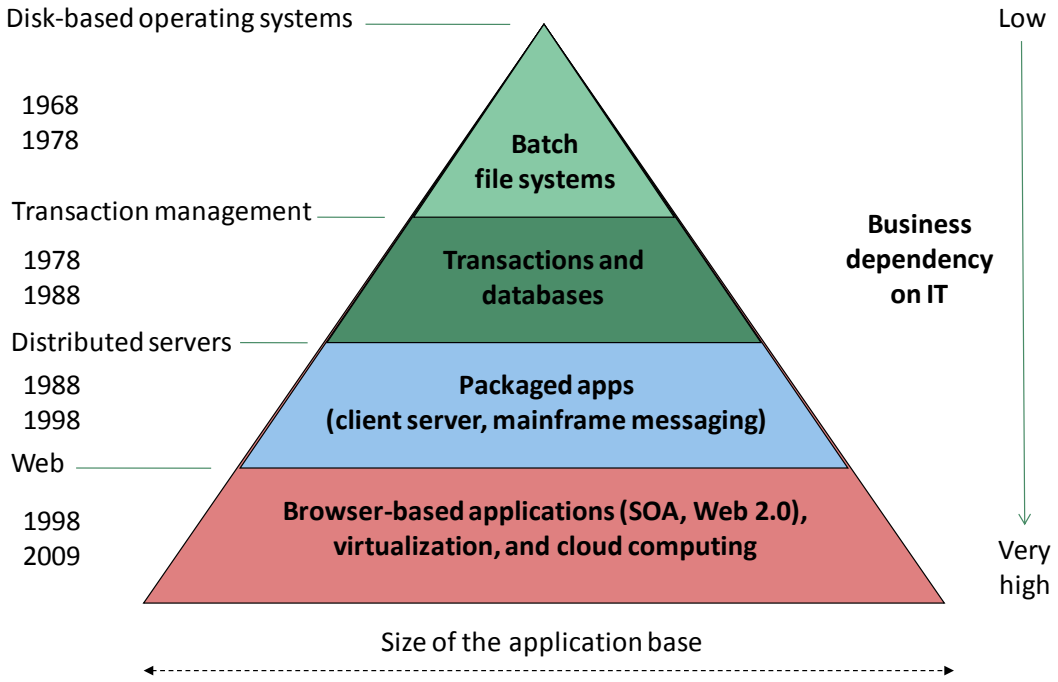
- **IT professionals want an end-to-end, integrated solution.** All interviewees saw a complete solution that encompasses end user experience, monitoring of all components involved in delivering the business services, and the ability to present all monitoring data on a single dashboard as the ideal solution.

- **This solution could come from a single vendor.** All IT professional agreed that if this solution was able to demonstrate a positive return on investment, they would certainly replace their existing tools with a single-vendor solution.

The Building Of Critical Applications

Several decades ago, IT technology took a decisive turn toward online transactional applications. This was fueled by technical innovations such as new concepts in networking (X.25-based, dedicated data networks rather than telephone lines, for example), the generalization of transaction management supervisors, and the appearance of the first database concepts. Enterprises started to make huge investments in these technologies and built the core applications and data sets of their enterprise IT around them. But technical innovation is a nonrestrictive phenomenon: Technology can be seen as a continuous flow while application projects are, by necessity, discrete events that use a fixed set of technological hypotheses. The consequence is that, as new technologies appear, they are integrated as a new layer on top of the existing ones. Most enterprises today are integrating several types of technologies into their IT infrastructures (see Figure 1).

Figure 1: Application Layers As A Function Of Technology



Source: "Market Overview: The Application Performance Management Software Market," Forrester Research, Inc., October 17, 2008

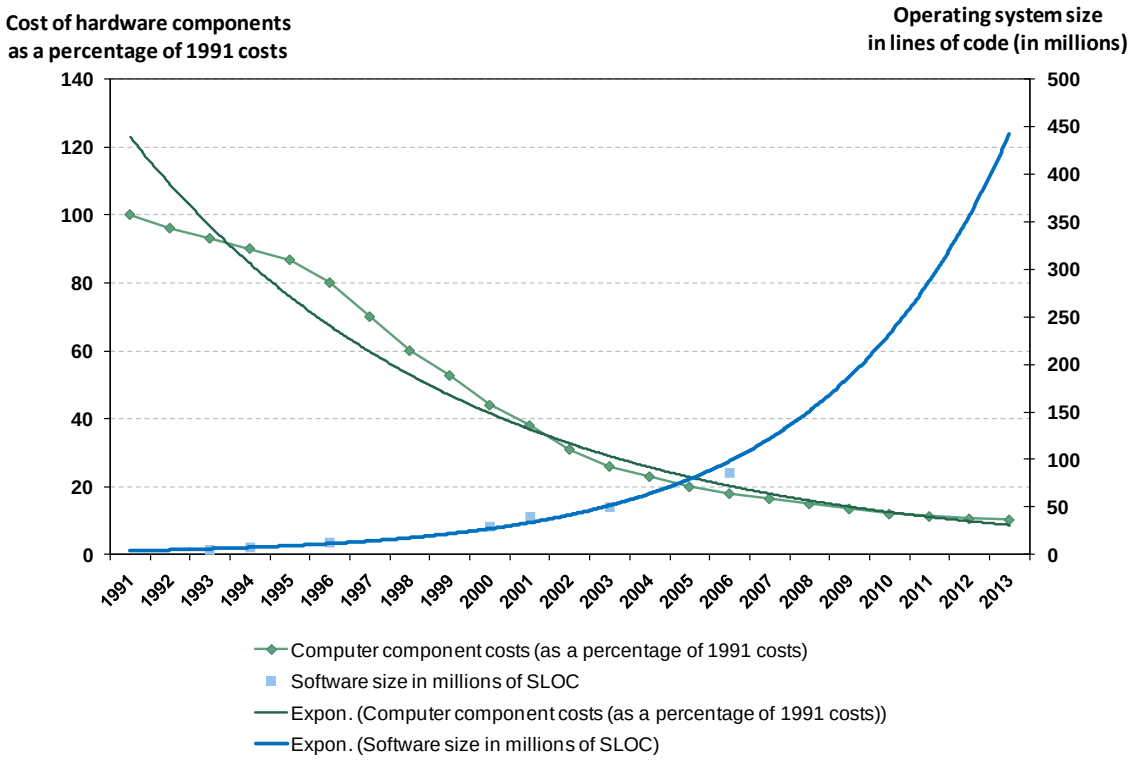
In the early 1990s, with a decisive turn toward simpler, commodity-based servers, the cost of hardware components started to drop exponentially. This had multiple consequences:

- **The number of applications multiplied.** As the availability of cheaper distributed systems grew, it became possible to automate business processes that presented a high cost-to-value ratio on a mainframe.

- **The core applications and data sets remained on the mainframe.** It did not make business sense to write off mission-critical and costly investments; new applications requiring enterprise-class data had to somehow reach into the mainframe for it.

- **The size of applications increased exponentially.** The availability of increasingly cheap hardware led to an increase in application size. Moore's law says that the density of transistors on a silicon chip doubles every two years. The corresponding "Humphrey's law" observes that the size of software effectively doubles every two years because it tends to take advantage of the available capacity and performance of the newest hardware (see Figure 2).

Figure 2: Software Size Increase and Hardware Price Decline



Source: "Future Trends In The Enterprise Software Market," Forrester Research, Inc., March 9, 2009

As new distributed application servers appeared, they became the ideal solution to expand the scope of online transactional applications while retaining their typical access to the mainframe back end, which remains the recipient of the critical enterprise data. As a banking IT executive stated:

> *"This type of architecture is used for everything from transaction processing to stock trading to workflow — about 600 apps: analysis of trading patterns, market surveillance systems, and market data systems — a wide variety of financial systems. Most of these use a mainframe back end. For some of them (for example, market surveillance) the back end is database storage, middle is Web, and front is Web/GUI."*

But this is also true in manufacturing, confirmed by the IT manager of a large manufacturing company:

> *"I support approximately 900 of 1,700 IT systems. Primary in this category are shop floor operations, which have very high criticality. Getting data to come off the*

mainframe, travel through a messaging solution, and get displayed on the floor for our users is critical."

It is true in retail as well:

"I work for a retailer, so these apps run all our stock ordering and management, and also run our customer-facing order systems and in-store purchase systems. They are absolutely vital: We could not survive without them."

Application Criticality

As applications and business services multiply in the enterprise, so does complexity. Applications that started as a simple revamping of the mainframe interface have now evolved into n-tier complex systems that use Web services and other forms of distributed inter-\application communication. This was made clear in our interview with an IT executive in the insurance and financial services industry:

"These applications have increased in complexity — the reason here is the technologies that are available. Years ago, all you had to worry about was the mainframe — it was all riding there. Once you started with the networks — distributed, etc. — you started adding complexity to these layers. Today you have control over only certain aspects of that spectrum."

Applications often grew organically, almost against the will of the IT organization, as stated by this government organization:

"We've gone down the road we didn't want to go down; we're (getting) more in the game with packages, but this n-tier environment is more bespoke to fit different customers' needs — and it's since grown larger than we ever intended."

And it seems that this trend will endure for quite some time. Surveys show that, even though a majority of custom developments use newer application server technologies, there is still a good part of mainframe development going on in large enterprises of more than 1,000 employees. Actually, the larger the enterprise, the more mainframe development takes place (see Figure 3).

Not only did these applications grow in size and complexity over time, they are often the most critical business services. This is confirmed across all industry verticals.

According to an IT executive in manufacturing:

"Downtime costs $150K per hour for every loss of use. It's done on shoestring level — only when big problems surface do we throw money at them. That's been a big issue for us. It was well into millions of dollars of resources to solve the shop floor issues and create a working system."

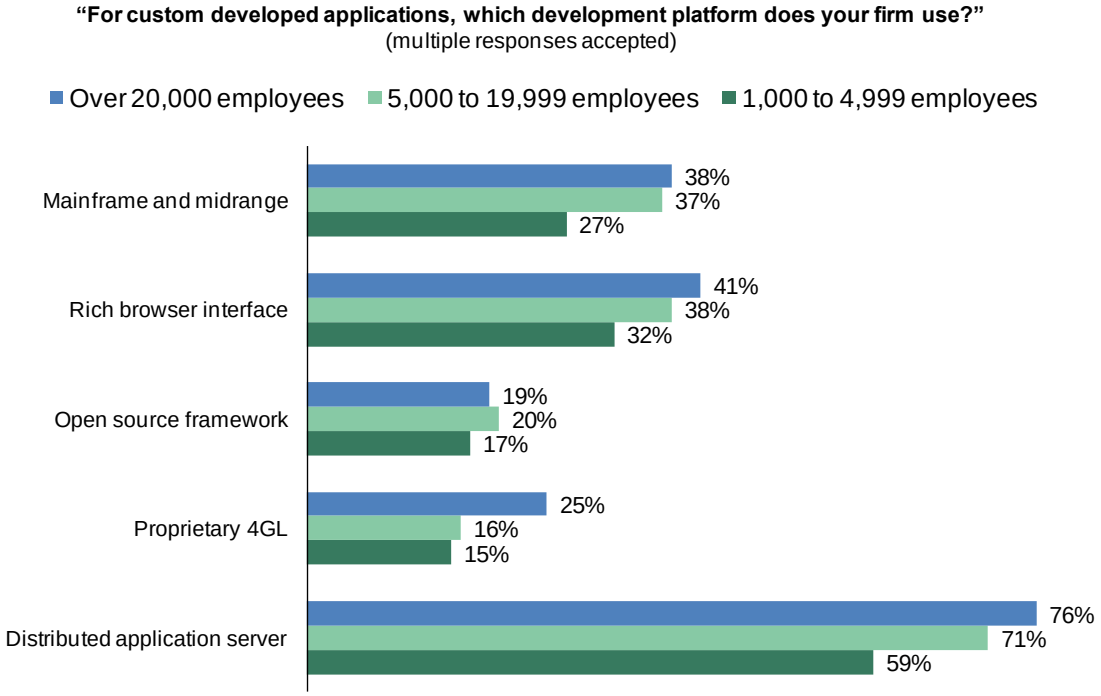
According to an IT executive in retail:

"If it were down in one store, it would be limited — around £20K to £30K. If it went down in all stores for an hour, it would be upward of £1 million. Resolving the issue can be from a few minutes to a few hours. If this happens on the weekend it's much longer, because the mainframe folks aren't there."

According to an IT executive in financial services (trading):

> *"It really depends on the value of what's going through. If [a brokerage firm] is making a $5 million move of stock, that's exactly what it's worth if there's downtime. The average estimate would be around $1 million per minute. We run a .999999 availability shop or higher."*

Figure 3: Application Development Platforms



Base: 512 North American and European enterprises (1,000-plus employees)
Source: Forrester Research Enterprise And SMB Software Survey, North America And Europe, Q4 2009

Applications have been built using the technology available at the time. As large enterprises adopted mainframes initially as their core IT technology and then shifted to distributed open systems and then Web-based applications, they were faced with the necessity of assembling business services from disparate applications on diverse platforms: The risk and the cost of ripping out large investments in time and money was too high to be really an option. These business services and applications are at the core of the business processes today, and their availability and performance are key to business productivity. Because of its considerable input/output capabilities, the mainframe remains the repository of the core data of many enterprises.

The Challenges Of n-Tier Application Performance Management

The business services built on these applications may be so business critical that enterprises will design redundant architecture to prevent downtime, as stated by an IT executive who supports the home banking applications in a very large North American bank:

> *"Our data centers supporting online banking have triplicate failover: West coast, east coast, and Midwest. As well as failover, these data centers load-balance. Response to code issues, disaster recovery, etc. are all automated in terms of traffic to maintain performance on the user end."*

But the cost of such architectures may be impossible to sustain in some cases, as implied by an IT executive in business services:

> *"We are doing analysis on the cost of 100% availability versus the benefit of 100% availability. The cost for this is tremendous — failover, redundancy, etc. . . . We (currently) spend four cents on the dollar (revenue) for all our internal IT."*

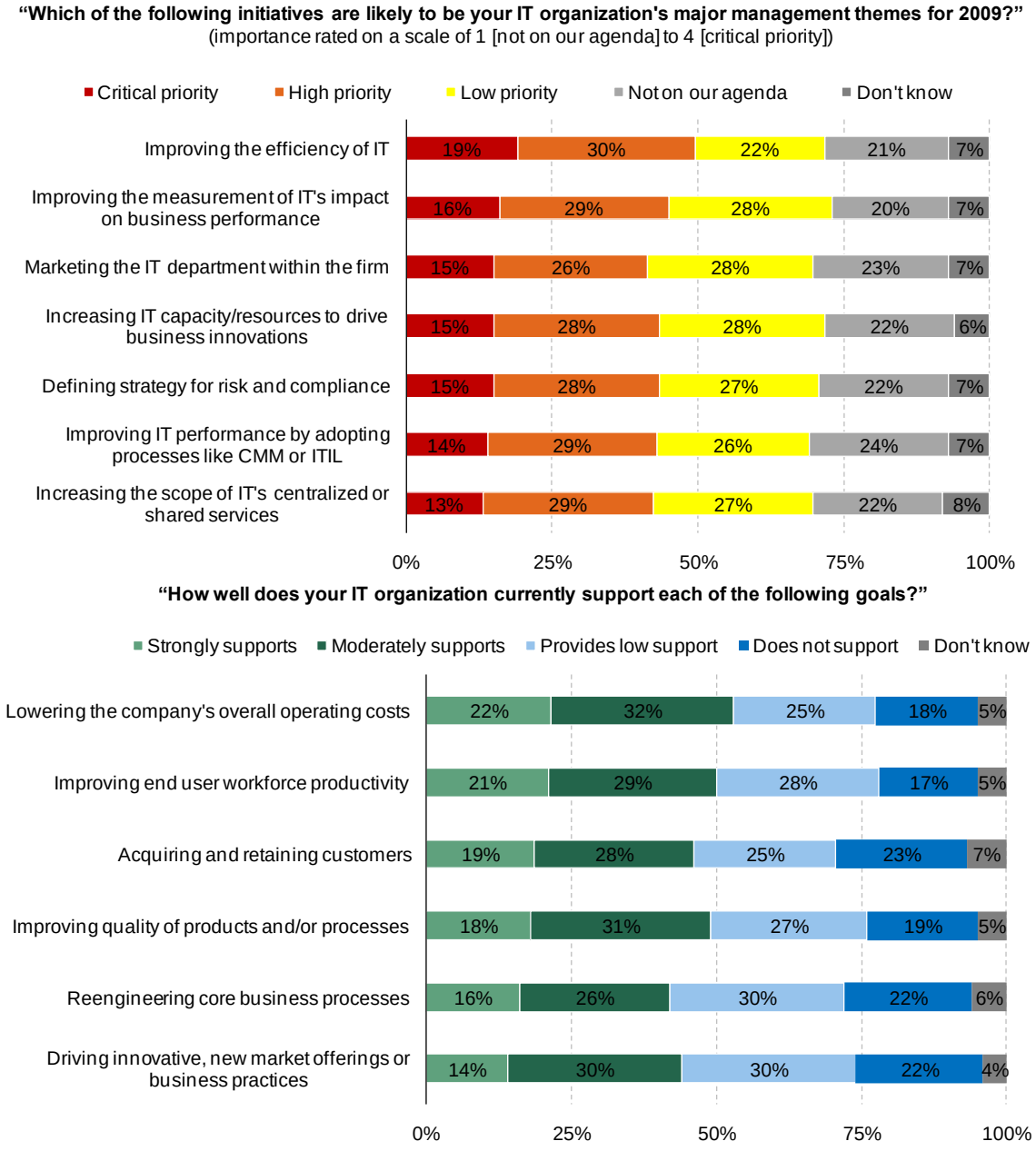
In many instances, IT has to find the right balance between end user productivity and IT efficiency. It means actually maintaining the required level of service quality while facing constantly shrinking budgets. These trends are verified across recent market studies (see Figure 4).

User productivity has always been directly linked to system performance. Usability studies show that after 10 seconds the user's attention starts to wander and that this should be considered an upper limit in most transactional applications.[1] Application performance is therefore the key factor of end user productivity and, consequently, business productivity.

In this context, many IT organizations are facing major challenges when it comes to managing performances within their allocated budget, as stated by this IT executive in an insurance company:

> *"Impact is severe at times in terms of productivity and politics — claims processing, for example. The system is much improved but suffers performance and availability problems, and it's [like] "poking around in the dark" to find and fix issues — manually looking at logs, etc. It's very painful and labor intensive, and the troubleshooting results are insufficient. When claims processors can't process claims, there's a huge political and reputational impact. That's one of our biggest challenges. In a few places we've introduced tools that have taken us very far, [but] we've been struggling from time and budget standpoints to retrofit these tools for visibility."*

Figure 4: Improving User Productivity And IT Efficiency Are Major Trends



Base: 1,019 North American and European enterprise IT budget decision-makers
(percentages may not total 100 because of rounding)

Source: Forrester Research Enterprise And SMB Global IT Budgets And Spending Survey, Q2 2009

One of the key issues faced by IT operations when dealing with performance management in the context of cooperating mainframe and distributed system operation is the ability to detect problems before they impact the end user. An IT executive in the banking sector says it all:

> *"We can look at server performance, but we don't know all the time if there is performance degradation on user side. We measure value on failed customer interactions (FCIs) as well; now we have 'Big FCIs' and 'little FCIs.' Ability to measure service-level agreements (SLAs) is our second biggest problem. [Our] final [challenge] is the inability to measure app performance in pre-production. We've had cases where we deployed and it brought the system down. We're best in class with our apps, etc., but we don't have a real end-to-end pre-performance environment — it's a huge issue for us."*

Another issue is to obtain application visibility across all the tiers and develop the capability to bring it all together, says an IT executive in the public sector:

> *"We've been happy with our distributed application server monitor, but it doesn't give us a full-field view. It gives us Web performance stats and certain components of infrastructure, but what we lack is something that is very precise in measuring end user performance. We get tickets with vague descriptions or with a number of possible causes, and we also have no way of being able to proactively determine that baseline performance has drifted; even in the course of troubleshooting, we don't have that. The second piece is, 'Where in the chain is it broken?' — user side, communication, middleware? It's a morass of different components, and you can't pinpoint where the slowdown is."*

This is confirmed by an IT executive in the insurance industry:

> *"Part of the monitoring challenge is that there is no single solution to cover the entire environment. There are great tools for application servers but not as good for C++ or other objects. The network layer analysis also has to be done after the fact and not in real time."*

This inability to bring information from different sources together brings up another problem. With support engineers organized by "competency silos," each using his own tools, cooperation between these different groups — and egos — may be a daunting task to achieve, as shown in this quote from an IT executive in government:

> *"Competency silos are very much an issue; we have specialists on [databases], network specialists, etc. Everyone thinks they are the linchpin that holds it all together — getting them to let go of that idea and see themselves as part of the team is difficult at times."*

In very large organizations, this is an even more complex problem, as described by this IT executive from a worldwide bank:

> *"It's complicated by our size: We not only have silos [of competency/technology], we have silos within the silos. Even in the same division there is noncommunication."*

The main difficulties facing IT operations in monitoring mission-critical services built on a mix of mainframe and distributed applications is clear:

- **The first major difficulty is to understand end user performance.** In many instances performance lapses are identified by the end user calling the service desk. In the case of public-facing applications, the clients would simply move on to a competing Web site, resulting in a loss of revenue. A user's call to the service desk does not often provide enough information to clearly identify the issues; visibility into end user performance is needed, and it should be a source of proactive alerts even before the issues are perceived by the business users.

- **The second difficulty is to understand the origin of the problem.** Because tools are providing fragmented information, it is difficult to find the real origin of a slowdown. Point solutions concentrate on certain parts of the infrastructure or diverse applications and leave the analysis and conclusions to the operations staff — quite obviously a very difficult task in a complex environment.

- **The third major difficulty is to promote cooperation between teams.** Because tools are providing fragmented results, nobody has complete visibility into a problem. Each tool may have its own timeline, for example, and aligning these between multiple tools often provides inconclusive data. This does not promote cooperation because each team looks at its own data to exonerate themselves from blame. Further, the data is often of an arcane nature that precludes the business side from participating in defining the remedial action: If the performance issue needs to be corrected by a design change, the participation of the business owner is critical but can only be achieved with clear, holistic, and actionable information.

Defining A Solution

The solution requirements are clearly derived from the challenges faced by IT operations in managing the performance of complex multitier applications. The first requirement is clearly stated by this IT executive in the public sector:

> *"Certainly, end user performance — synthesized transactions, end user monitoring, and monitoring of end user experience [are what we'd look for in a solution]. I'd also like to view performance of components from end to end for each app, whatever it may be. . . . The product should store the baselines and alert in real time if there's danger of any component exceeding these thresholds — but it would have to do this end to end. It would also be important to have those things stored for historical reference, trending, and reporting. If it can do predictive analysis as well based on this historical data, trends, etc., that would be great."*

This should be complemented by an end-to-end root cause analysis of the issue, as stated by this IT executive in the banking sector:

> *"First and foremost, [we need] a responsive system that would respond automatically or, at a minimum give me root-cause analysis. If it were truly end to end, it should give me an analysis; it's one thing to give me an alert — and quite another to give me an analysis and help me fix the problem with recommendations."*

This is confirmed by an executive in manufacturing, who also introduces the notion of a dashboard grouping all information together:

> *"Right now, if I get an alert notice today that something is down, I don't know what's running on it automatically, etc. We need appropriate tagging on all systems, not just the shop floor. If I had that in a dashboard, it would be ideal."*

Most IT professionals interviewed for this study see this notion of a dashboard grouping information from disparate systems in the application chain is seen as a solution to the problem of team cooperation. From an executive in financial services:

> *"Bringing together fragmented teams is an issue here. IT is so wide a group. There are also generational gaps with technologies and expertise, etc. We feel there should be something to click in and see how all these interfaces are interacting. This type of thing would bring distinct teams together, so we are all speaking the same language."*

An IT executive in the retail sector makes almost the same comment:

> *"There is an issue with lots of disparate IT folks trying to talk to each other about these ongoing [problems] — they are from totally different planets and communicate accordingly. If I could do a quick search or click on a pane and look at the same view as someone in another tier, we could have the right conversation."*

An IT executive in government sums up the core need, when asked about what an ideal solution should provide:

> *"Increase of in-depth and wide breadth of monitoring — increase how many people can see what's going on."*

From these comments, we see that the ideal solution for monitoring and managing the performance of composite applications that include several tiers of communicating applications — especially mainframe applications communicating with open system applications — should include:

- **End user monitoring for alerting and SLA management.** This would help reduce the number of calls from end users, but would also provide a better characterization of problems. It could also provide historical trends that would help prevent problems.

- **Visibility into the application path.** As multiple system tiers are brought together, IT has the need to understand where the problem is likely coming from. This includes understanding the different systems involved in a transaction and requires the collection of information from all moving parts.

- **A single pane of glass.** As reported by the IT professionals interviewed for this paper, firms could best achieve cooperation between different competencies and constituencies if all information was reported on a single pane of glass that helped participants understand the "big picture" of the application performance. This should regroup all data from monitored tiers — especially the mainframe and distributed system tiers — and provide some level of automated analysis.

Study Conclusions

Innovation is a nonrestrictive phenomenon, which means it is often seen as a continuous stream of new technologies, sometimes built on top of one another, rather than as a series of discrete events. IT applications and projects, on the other hand, are purely discrete: This implies that technology choices are made and cast in stone at the beginning of a project and, with periodic revisions, for the life of the application.

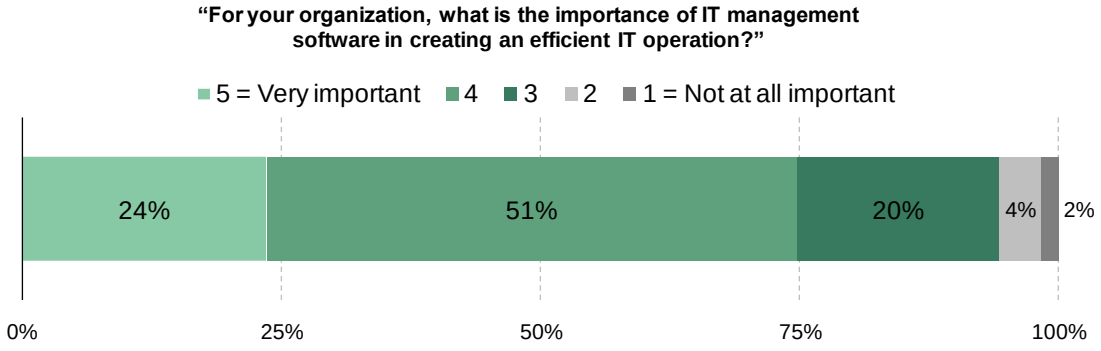
We see now that investments made over time are so critical that it becomes increasingly difficult to renew them today because the risk and cost are too great. It is also often the case that critical data sets were grown early on, and that their size precludes a move to another technology.

In the meantime, the ever-decreasing cost of hardware has made a large number of applications economically viable and possible, and this phenomenon has been amplified greatly by the availability of quasi-universal communication. As the number and size of applications has increased, the number of potential users has also increased dramatically.

Because these applications require access to corporate data, multi-tier composite applications have been built on distributed system application servers linked to mainframe applications and data sets. Most of these composite applications are mission critical, and downtime or brownouts usually cost a lot of money for the enterprise using them: The IT professionals interviewed in this survey offered a potential range from several hundreds of thousands of dollars to several millions of dollars per hour of downtime.

To prevent these issues, IT organizations have taken some measures, from building fully redundant infrastructures to collecting performance data through point solutions at the component — whether hardware or software — level. A previous study commissioned by CA and conducted by Forrester Consulting shows that in fact, IT management solutions are seen as an important part of addressing IT efficiency issues (see Figure 5).

Figure 5: Importance of IT Management Software In IT Efficiency



Base: 174 IT infrastructure decision-makers

Source: "IT's Assessment Of Support Services," a commissioned study conducted by Forrester Consulting on behalf of CA, September 2009

The ideal IT solution for the management of composite application performance should offer a number of key characteristics:

- **End user experience monitoring and analysis and proactive resolution.**
- **Identification of the components used to deliver the application.**
- **The monitoring of hardware and software components, legacy or open systems.**
- **The presentation of all information on a single pane of glass, with the potential for root cause analysis and historical trending.**

Many of the enterprises interviewed are using IT management solutions today, and would readily buy such a solution from a single vendor, as stated by an IT executive in manufacturing:

> *"We have a tendency to go to a single vendor due to economies of scale and ease of relationship, so we'd probably lean that way. Worldwide support is critical as well."*

Many enterprises are using several point solutions in their management of applications. However, presented with a solution to these requirements, they would certainly accept the idea of a single vendor replacing their existing disparate solutions. As stated by an IT executive in retail:

> *"I think this would be all from one vendor. This is part of our problem as it is — having homegrown and multiple different vendors/processes — so bringing this together would be obvious. [But] we'd also want some integration with those homegrown apps — sharing data, etc."*

Proof of return on investment would be critical to acceptance of a single solution, however, as discussed by this executive in financial services:

> *"It would depend on what we are getting for the full environment. We'd need to see results very quickly to prove it was worth it to make this level of switch. If we were convinced, there would be some risk, but given time to compare results between the two, we'd go with the best option."*

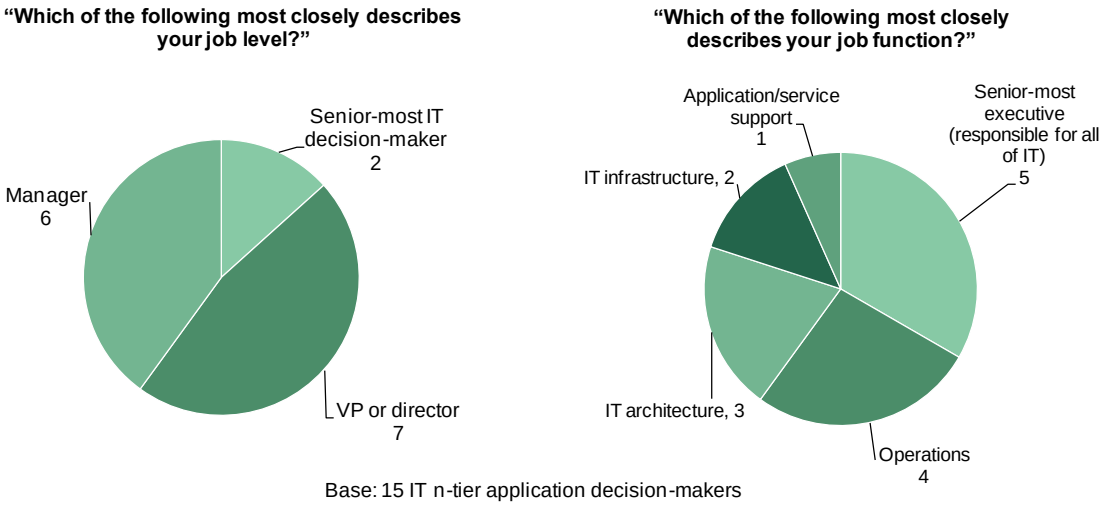
Overall, a single integrated solution that fulfills the key requirements of composite application performance management seems to be, for all IT professionals interviewed for this study, a key element in creating a more efficient IT organization that effectively supports business productivity.

Appendix A: Methodology

In October 2009, CA commissioned Forrester Consulting to evaluate the issues surrounding the management of composite applications built on mainframe systems and open system middleware solutions. In this study, Forrester conducted 15 in-depth interviews (via telephone) of IT decision-makers responsible for management of n-tier applications in their organization. Details on interviewee demographics are as follows:

- **Geographical focus.** 10 US and five UK respondents.
- **Organizational focus.** Enterprises with application server middleware spanning distributed and mainframe platforms.
- **Size focus.** Revenues of $1 billion or more. Six respondents reported revenues of $1 billion to $5 billion, and nine respondents reported revenues of more than $5 billion.
- **Role focus.** Manager level and above; personnel with direct responsibility for n-tier applications architecture and/or support (see Figure A.1).
- **Industries.** Distribution across six key verticals (see Figure A.2).

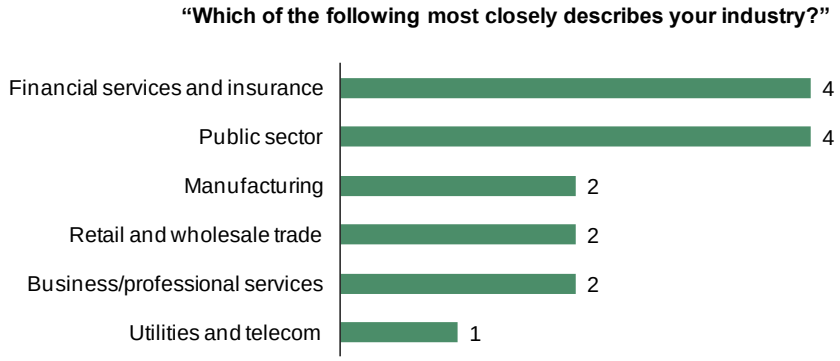
Figure A.1: Roles and Job Functions Represented



Base: 15 IT n-tier application decision-makers
(shown by count only; sample is not statistically valid)

Source: "Current State Of N-Tier End-To-End Performance Management," a commissioned study conducted by Forrester Consulting on behalf of CA, November 2009

Figure A.2: Industries Represented



Base: 15 IT n-tier application decision-makers
(shown by count only; sample is not statistically valid)

Source: "Current State Of N-Tier End-To-End Performance Management," a commissioned study conducted by Forrester Consulting on behalf of CA, November 2009

Appendix B: Supplemental Material

Related Forrester Research

“Market Overview: The Application Performance Management Market,” Forrester Research, Inc., October 17, 2008

“The State Of Enterprise IT Budgets: 2009,” Forrester Research, Inc., August 7, 2009

Appendix C: Endnotes

[1] Source: Jakob Nielsen, *Usability Engineering*, Morgan Kaufmann, 1993.