



## Cloud Cover

Delivering on the Value of the Cloud  
in Public Sector IT Organizations

By Ben Newton and Herb VanHook, BMC Software

TABLE OF CONTENTS

- EXECUTIVE SUMMARY . . . . . 1
  
- INTRODUCTION . . . . . 2
  
- CLOUD FUNDAMENTALS . . . . . 2
  - » Abstraction . . . . . 2
  - » Automation . . . . . 2
  - » Availability . . . . . 3
  
- THE CLOUD BANK — USING THE CLOUD AS A RESOURCE POOL . . . . . 3
  - » Dynamic Resource Pooling and Capacity Management . . . . . 3
  - » The Power of Community — Providing Location-Independent Resources . . . . . 4
  - » The Power of Public Cloud Services . . . . . 4
  
- CHARGING FOR THE CLOUD — ATTACHING BUSINESS VALUE TO A SERVICE . . . . . 4
  - » How Much Is a Service Worth? . . . . . 4
  - » Pay-Per-Use Models . . . . . 5
  
- AUTOMATION — THE KEY TO MANAGING THE CLOUD . . . . . 5
  - » Request Management — On-Demand Self-Service . . . . . 5
  - » Change Management . . . . . 6
  - » Configuration Automation . . . . . 6
  - » Compliance Management . . . . . 6
  - » Configuration Management Database (CMDB) or Configuration Management System (CMS) . . . . . 7
  - » Capacity Management . . . . . 7
  - » Availability Management and Service Level Agreements (SLAs) . . . . . 7
  
- SERVICE DESK . . . . . 7
  
- BSM FOR VIRTUALIZATION AND CLOUD COMPUTING:  
THE BEST WAY TO MANAGE THE CLOUD . . . . . 7

## EXECUTIVE SUMMARY

You believe that cloud computing can bring real value to your agency. You are certain there are real cost-savings to be achieved and real value to be passed on to internal stakeholders and external constituents alike — if only you can get beyond the hype and make cloud services a reality in your organization. But now what? There's no question that this is where the real fun starts. It is one thing to ponder how the private sector's next IT innovation will revolutionize the way the public sector does business. It is something completely different to actually deliver on that promise.

The good news is that with the right tools and the right approach, you have an amazing opportunity to leapfrog from last year's cutting-edge innovations right to cloud computing. This is because the public sector has great potential for huge economies of scale. The vast numbers of users, managed devices, and services within government IT organizations is staggering. It is time to tap the cloud to exploit these economies of scale to achieve the cost savings being requested in every agency and organization in the public sector.

The best way to do this is to make sure that you have the policies and processes in place to control and manage your cloud environment. Don't just get caught up in the excitement of a new technology. By defining, automating, managing, measuring, and governing your efforts, your chances of success in the cloud will skyrocket.

Business Service Management (BSM) for virtualization and cloud computing provides the best platform for doing just that. Through proven technology, this approach unifies the management of traditional (physical), virtual, hybrid, private, and public cloud-based resources, so you can first build and then manage your entire cloud environment from a single, unified platform. In so doing, BSM for virtualization and cloud computing gives public sector IT organizations like yours the solutions you need to adopt cloud computing — with less risk and lower costs.

## INTRODUCTION

A wealth of information about cloud computing models and types can be found on the National Institute of Standards and Technology (NIST) Web site<sup>1</sup> and from other sources. Instead of rehashing that material here, we will focus on the challenges inherent in the implementation and management of those models. What is often overlooked by many IT organizations is the effort involved in building the capabilities necessary for cloud computing and in maintaining them over time. Unlike the cloud computing heavyweights of private industry (Google, Amazon, Salesforce.com, etc.), public sector IT organizations do not have the benefit of years of trial and error — nor that of accumulated experience — to make implementations successful. Therefore, to move forward, it is essential to be well informed, with expectations properly set.

## CLOUD FUNDAMENTALS

A fundamental requirement for cloud computing is to eliminate the need for your internal stakeholders and external constituents to understand or even consider the complexities of the back-end technology infrastructure supporting the services being provided to them. In other words, the value cloud computing provides is that *it gives your customers the freedom to focus on their core business or mission.*<sup>2</sup> As the service provider, you have two primary techniques for providing them with this freedom: *abstraction* and *automation*.

### ABSTRACTION

In this context, abstraction is the process of hiding the details of the implementation to focus on service delivery. Abstraction is achieved primarily through virtualization. The main thrust of virtualization is to hide the complexity of supporting infrastructure and to provide cost-saving uniformity regardless of the supporting hardware. However, full abstraction also requires hiding process complexity behind the scenes. For example, according to IT Infrastructure Library® (ITIL®) guidelines, the complexity associated with the change and release management processes necessary to support the deployment of new services is abstracted through the service catalog. Through this catalog, customers select from a simple list of available services and choose those that meet their needs.

### AUTOMATION

While it is entirely possible to implement abstraction backed by manual efforts, this is a doomed approach. Abstraction depends on both uniformity and configuration controls to achieve any economies of scale and cost-savings. No amount of documentation and processes will ever guarantee the uniformity and consistency of work done by administrators manually. On the other hand, with the upfront investment of time in proper automation, complex service delivery can be reduced to a very defined and predictable process with significant cost savings over the long term.

Automation generally takes two forms: *process automation* and *configuration automation*.

- » **Process automation** is best understood in terms of automating the fulfillment of processes and activities in a model such as ITIL. Service requests, selected by the customer from the service catalog, must automatically be converted into the complex tasks necessary to achieve the end result, represented in the functions of configuration, change, and release management.
- » **Configuration automation** follows process automation and carries out the actual configurations represented by the change requests. It reshapes the technology environment “on the fly” to meet the needs of the requested services. Because of the sheer volume of changes necessary to maintain any sort of complex environment, cost-effective cloud computing isn’t remotely possible unless this automation is in place.

In considering these two types of automation, you may ask, *Why is process automation even necessary? Isn't it enough to merely automate the building of services? Doesn't the layer of process automation add unnecessary complexity to the management of cloud services?* The answers come down to the concept of configuration control. Without controls in place, you have no way to plan or design the infrastructure properly, since the end result is completely undefined. This lack of definition can lead to a variety of IT control issues, not the least of which is virtual sprawl — the uncontrolled and overwhelming growth of the virtual infrastructure.

## AVAILABILITY

*Availability* is a significant part of any cloud services discussion. It will be difficult to transition your infrastructure to a cloud services model if, by doing so, you lose real-time visibility into system availability. In fact, why would any IT organization forgo control over its infrastructure unless it has confidence that the quality of service will be more compelling in the cloud?

Service level agreements (SLAs), as defined by ITIL, are the method for both agreeing upon and communicating adherence to expectations. If you are to effectively meet your SLAs, it is essential that your cloud environment supports real-time availability and performance monitoring to ensure that expectations are being met. Conversely, it is also vital that your customers be provided a dashboard for visibility into the state of their services.

With a cloud architecture based on abstraction, automation, and availability, you have the core of cloud service management. As Figure 1 shows, these elements enable a constant feedback loop between you and your end users.

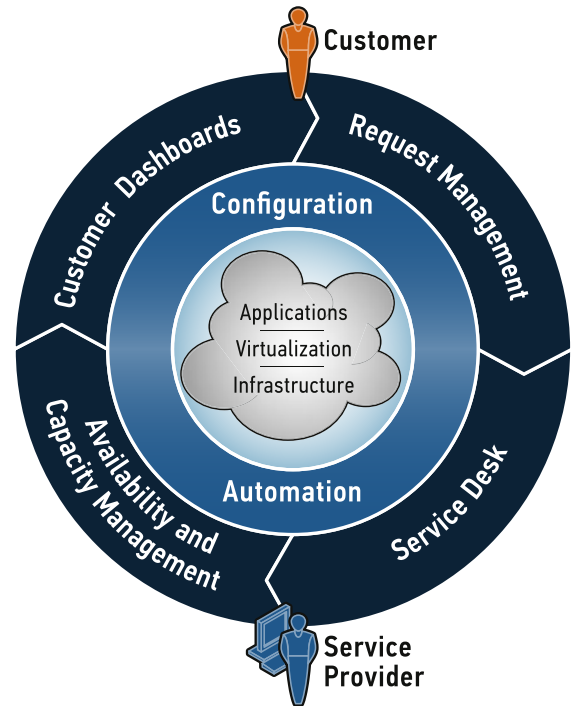


Figure 1: Service Management Processes in Cloud Management

## THE CLOUD BANK — USING THE CLOUD AS A RESOURCE POOL

If one of the fundamental tenets of cloud computing is to free your customers to focus on their core business requirements, as stated above, then the flip side of that coin is that the cloud also *gives your IT organization the freedom to focus on the best and most cost-effective way to deliver the requested services*. Many shared environments today struggle with the consequences of receiving fully formed resource requests (e.g., *I need this many servers with this much CPU and RAM*), rather than higher-level business requests (e.g., *I need enough capacity to perform this specific service*). The inevitable consequence of this arm's-length type of resource request management is twofold:

1. Customers overestimate requirements to “hedge their bets” and ensure acceptable service, and
2. IT is stuck in a reactive mode, with no understanding of actual business requirements, and subsequently, no way to properly plan ahead.

## DYNAMIC RESOURCE POOLING AND CAPACITY MANAGEMENT

It is here that cloud computing really shines and provides real cost savings. Using the cloud, you can offer up services that are satisfied from a *dynamic resource pool*. As resource needs fluctuate, you now have the flexibility to choose the best solution from the available resources, which is where the concept of Dynamic BSM comes into play. Dynamic BSM is the next evolution of management architecture where the goal is not only to address the core requirements of IT management but also to support the on-demand provisioning and management of services that leverage new computing platforms, such as virtualization and cloud computing infrastructures.

So what are the fundamentals of providing this kind of dynamism? The first element is having a fully virtualized infrastructure that allows for the simple and efficient movement of resources to achieve the best possible utilization of available capacity. There is significant literature addressing this part, and there are many excellent technologies to help achieve it. However, the question that is so often missed is, *How do we measure performance, and what qualifies as good versus poor performance?*

To answer this question, you need real-time, intelligent performance management to clearly define how the service is performing from the customer’s point of view, as well as capacity management to optimize cloud resources by identifying the customer’s current and future capacity requirements. Customer perception is based on agreed-upon SLAs, which, once defined, provide a very clear boundary between acceptable and unacceptable performance. Combined with defined levels of service (say, gold, silver, and bronze), SLAs also provide great flexibility for the cloud computing implementation in terms of pricing (discussed

below). Based on the performance against the SLA, the cloud provider can adjust capacity up and down to achieve the most efficient use of the available infrastructure.

Finally, it is one thing to talk about dynamic resource allocation; it is quite another to do it. While virtualization removes much of the complexity inherent in the underlying hardware, it does not provide the whole answer. Unless the cloud environment is mostly homogeneous, blindly “copying” virtualized resources in an effort to increase capacity will not achieve the desired result. It is essential to automate the entire application stack within the virtual machines, so that there is complete flexibility in choosing how to increase capacity.

### **THE POWER OF COMMUNITY — PROVIDING LOCATION-INDEPENDENT RESOURCES**

The community cloud model is an especially powerful idea within the public sector. Multiple agencies with similar concerns and requirements can create an environment that serves overlapping agency needs. This is especially relevant in the context of lower budgets and information-sharing requirements at both the federal and state levels. The community cloud provides many benefits of public cloud services, without the security concerns. The community cloud model stands or falls based on the power of dynamic resource allocation. As the size of the community grows, the economies of scale are more and more compelling. Conversely, the more diverse the interests served, the greater the need for the cloud provider to use abstraction and automation to their fullest effect. With the right networks in place, the community cloud provides a pool of *location-independent* resources that can be leveraged equally by each of the participating organizations.

### **THE POWER OF PUBLIC CLOUD SERVICES**

Today, even the most secure government agencies need to share some information with the public. Formerly unacknowledged intelligence agencies now have Web sites, and government data across the board is being shared with the public. Public cloud services can be used to provide overflow capacity or a fully hosted environment (e.g., USA.gov, the U.S. General Services Administration [GSA] Web site) for those government services with less severe security requirements. For example, a government agency might anticipate much higher usage of a service during a short period of time (e.g., tax season, natural disasters), and with the proper integrations in place, Internet traffic could easily be load-balanced between the public and private cloud to quickly increase capacity, all without incurring any long-term costs.

### **CHARGING FOR THE CLOUD — ATTACHING BUSINESS VALUE TO A SERVICE**

As you begin to determine the value of each service you provide and how much to charge for each service, here are some key points to keep in mind.

#### **HOW MUCH IS A SERVICE WORTH?**

One of the basic tenets of capitalism is that market-based pricing of goods provides the most efficient way to reconcile supply and demand. Prices help to deliver available goods to those who want or need them the most. Going back to our original example, a service catalog delivered through a cloud computing model would function as a simple marketplace, so many of the same rules would apply. That also means that giving services away for “free” reduces their worth, much as it does in the regular economy.

Pricing is a necessity for public cloud services, but it is often left out of private or community implementations. However, we would argue that pricing is just as important in a private or community cloud. Implementation costs are similar, if not higher, so it stands to reason that delivery costs should be communicated to customers. In other words, both internal and external customers should be made aware of the relative costs of one service option versus another.

Service request management and service level management are central to the problem of pricing. Like a menu in a restaurant, prices should be clearly attached to the available services in the service catalog. In addition, the service levels provided for each option should be clearly communicated to the customer. The result is that the customer gets what they really want for a price they can afford, and the provider clearly understands what is important to the customer.

Finally, one often overlooked aspect of dynamic resource allocation is the harvesting of deactivated services. If the lifetime of a service is not clearly defined up front in the service request, the customer may not take the time to release any resources that are no longer needed, particularly if no cost is associated with them. Even if the customer can override the deactivation when

the time comes, the inclusion of a time limit clearly establishes a process where unused resources can be reclaimed. Another option would be to provide incentives for clear time limits by offering cost rebates on services that are “reserved” for a specific time period (e.g., “I need ten servers on March 1 for 60 days”), versus “spot requests” (e.g., “I need ten servers right now”). The bottom line is that real cost reductions over the long term will be difficult without effective resource harvesting.

### PAY-PER-USE MODELS

Once pricing has been established as part of the service request process, it is a small step to consider more finely tuned pricing models. The flexibility inherent in virtualization for using computing resources more efficiently also lends itself to the idea of customers paying only for what they use, as opposed to paying a fixed price. This is beneficial to both parties. Customers can be confident that their costs are directly tied to the value they receive, and you can more easily squeeze every drop of capacity from the available system, while also passing on the true costs of delivery.

So, how can you achieve this kind of flexible pricing? It can be as simple as charging the customer per virtual machine operating under their service or based on comprehensive business modeling. The essential element of business modeling of cloud service pricing is to have the visibility into all of the performance, usage, and configuration data across the infrastructure. This data can then be subjected to demand modeling, cost modeling, consumption reporting, planned versus actual, and so on. These analyses can then be fed into the services pricing to provide constant feedback based on usage and capacity.

### AUTOMATION — THE KEY TO MANAGING THE CLOUD

By using automation to tie together processes across silos, you can ensure that you balance the agility and control needed to succeed in your cloud initiatives. The automated processes necessary to build and manage your cloud services can be divided into two parts (see Figure 2). First, there are the processes that deliver the service. For example, request management accepts the service request from the end user, the request is translated into a request for change (RFC) in change management, and then that change is executed in the cloud by configuration automation. Second, the feedback processes assure that the delivered service meets the standards required by the provider and the requester. For example, compliance management helps to ensure that the resources provided meet all applicable security standards. Availability management helps to ensure that SLAs are met and that the service desk resolves any incidents. Capacity management helps IT meet future SLAs by adding or removing resources as needed. These processes all leverage the configuration management system (CMS) or configuration management database (CMDB) as the record of the current state of the architecture.

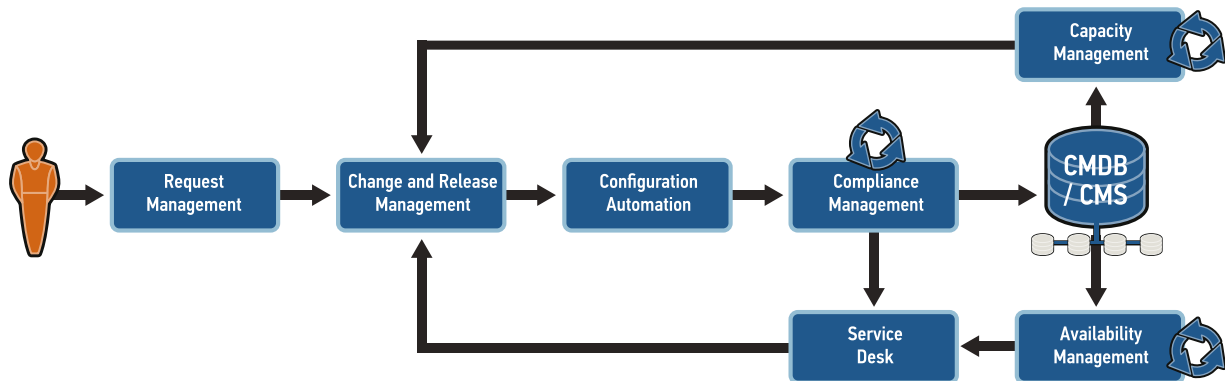


Figure 2: Cloud Computing Process Automation

### REQUEST MANAGEMENT — ON-DEMAND SELF-SERVICE

The first step in the delivery of a service will always be taking a request from the customer (request management). There are two essential elements to delivering true automation at this stage: the customer interface and the integration with change management. The “cloud experience” is doomed to fail if your request management interface is not simple and user friendly. The crux of the whole experience for your customers lies in their ability to focus entirely on what they need — not on how the IT organization will fulfill that need. Despite the complexity behind the scenes, the information you require from your customers

should be simple, easy to understand, and directly related to their own understanding of what they think they need. To accomplish this objective, narrow down your organization’s capabilities to a select set of available services, and then outline those offerings in a *service catalog*.

Keep in mind that if you want to add a new service to the catalog, you should already have in place the underlying automation necessary to deliver that service *end-to-end*. Leaving it out will inevitably lead to a bad customer experience, since, more likely than not, the implied manual processes will not scale to support the increased demand.

### CHANGE MANAGEMENT

The basic information gathered from the customer is translated into an RFC based on existing delivery processes. The RFC can be abstracted out to the service level (e.g., “add Web capacity”), or the server request can generate multiple, specific RFCs (e.g., one for creating a new virtual machine, another for installing the required application, and a third for configuring the application to the customer’s needs). If necessary, each of these requests could then be subjected to the standard approval processes, so that the manual IT resources required to manage the request have been reduced to the approval step only. However, in most cases, the manual intervention can be entirely eliminated through a pre-approval process. No longer do you need a support analyst to manage the entire process. This efficiency gain also leads to huge cost savings right up front.

### CONFIGURATION AUTOMATION

Configuration automation delivers on the change management process, deploying the server, client, network, or application required by your customer. There may be a temptation to reduce configuration automation to the mere juggling of virtual images or templates (a fully configured virtual server image that can be copied over and over to build capacity). While this image-based approach may be appropriate for the most simple, homogeneous implementations (which require little-to-no custom configuration), it does not scale to an environment with any sort of complexity or size. The most flexible approach requires that the delivery process be separated into “atomic components,” each of which is automated (see Figure 3). In the context of a virtual server, this means separating the applications, configurations, and custom data from the virtualized foundation. When a new service is delivered or service capacity is expanded, the pieces required can be assembled from the building blocks to deliver the entire stack required in real time.

### COMPLIANCE MANAGEMENT

Compliance management in the public sector — even within internal support networks — has become one of the most essential, yet least automated, processes in the data center. The complexity and quantity of compliance requests keep increasing, yet many administrators still wade through the morass of impenetrable rules and overwhelming data without sufficient automation to increase efficiency.

While this situation might have been accepted, or at least tolerated, in a legacy environment, it is unacceptable within the cloud. Resorting to manual methods of compliance can quickly eliminate any cost savings and customer satisfaction improvements. The value of the cloud is entirely dependent on its ability to reduce costs and simplify the customer experience; the rapid pace and ease of change within a virtualized infrastructure only magnify those problems.

Compliance management, in this context, takes on two very important aspects. First, the application of compliance rules must be both separate from, and integrated into, the delivery process. The compliance automation fixes and patches and will then be just another element laid atop of the underlying operating system and applications, thus reducing the need to constantly update images. Second, the compliance process must be tightly integrated into the incident management process (outlined below). This allows unexpected compliance findings to be dealt with in the same efficient process that is used to deal with service failures.

Additionally, the same compliance automation that is used to initially harden the virtual system is the same mechanism that is used to keep current systems secure. Now, only one set of policies needs to be updated, greatly reducing complexity and minimizing the risk of error.

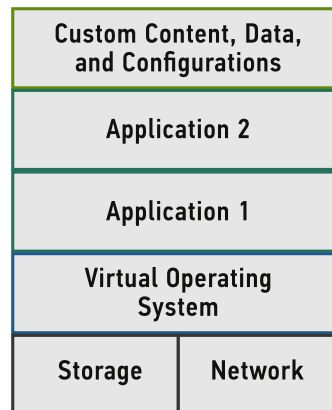


Figure 3: Virtual Server Stack of “Atomic Components”

## CONFIGURATION MANAGEMENT DATABASE (CMDB) OR CONFIGURATION MANAGEMENT SYSTEM (CMS)

After the entire virtual stack has been deployed or updated, the next step is to follow the ITIL configuration management process and update the CMDB (ITIL v2) or CMS (ITIL v3) through an automated discovery process. The pace of change within the cloud means that any configuration management, by necessity, must be tightly integrated with the virtual infrastructure and aware of changes as they happen. What's more, these updates should not be confined only to the initial build. The very nature of a cloud infrastructure implies the ability to rapidly reconfigure capacity and recover from failure. Therefore, it is essential that your CMDB/CMS be proactively updated any time the virtual infrastructure is changed, whether or not that change is planned.

## CAPACITY MANAGEMENT

Performance and usage data flows in one end of the process, and recommended environment changes come out of the other. These changes may cover a broad range of actions — from simply adding resources to running virtual devices to spinning up entirely new capacity out of the available pool (see the discussion of dynamic resource allocation above).

## AVAILABILITY MANAGEMENT AND SERVICE LEVEL AGREEMENTS (SLAs)

Real-time and intelligent availability management provides the finger on the “pulse” of the cloud. Systems need to be monitored against SLAs, and any service failures need to be dealt with in a quick and decisive manner. For example, if the system detected that the performance of an application was deteriorating and threatening the SLA, new resources could be dynamically allocated to compensate. At the same time, a problem resolution process could be initiated to perform a baseline audit to detect if any changes were made to the application, and an RFC could be submitted for the remediation of any unapproved changes found in the audit.

## SERVICE DESK

The final function in a closed-loop IT management process — and one that is essential to delivering the cloud — is the service desk, or in ITIL terms, *incident and problem management*. More than any other process, this function affects customer satisfaction with the cloud, since the service desk is the customer view into the timely resolution of their issues. In many of the simpler virtualized architectures in place today, it is likely that troublesome virtual appliances will be merely “recycled” as opposed to “fixed.” The complexity inherent in any real-world application of cloud computing necessitates a more robust approach to problem solving. A truly enterprise-level service desk is the only answer. By fully integrating and automating the service desk into your virtual environment, you can lessen the effects of unstable complexity and quickly solve problems when they are found.

## BSM SOLUTIONS FOR CLOUD COMPUTING: THE BEST WAY TO MANAGE THE CLOUD

As you continue to explore the benefits that cloud computing can bring to your public sector IT organization, make sure you don't just get caught up in the excitement of a new technology. By defining, automating, managing, measuring, and governing your efforts, your chances of success in the cloud skyrocket.

BSM provides a comprehensive and unified platform for running IT — across mainframe, distributed, virtual, and cloud environments. Using proven technology, you can gain the automation and control you need to pursue cloud computing in your public sector IT organization — with less risk and greater cost savings.

For more information about cloud computing solutions from BMC Software, please visit [www.bmc.com/cloud](http://www.bmc.com/cloud).

## END NOTES

- 1 Peter Mell and Tim Grance, “The NIST Definition of Cloud Computing,” Version 15, <http://csrc.nist.gov/groups/SNS/cloud-computing/cloud-def-v15.doc>.
- 2 The public sector does not define *customers* in the commercial sense commonly used in the private sector. However, every IT organization does have its own customers. These customers may be internal stakeholders, individual end users, other agencies, or constituents.

## **Business runs on IT. IT runs on BMC Software.**

Business thrives when IT runs smarter, faster, and stronger. That's why the most demanding IT organizations in the world rely on BMC Software across both distributed and mainframe environments. Recognized as the leader in Business Service Management, BMC provides a comprehensive and unified platform that helps IT organizations cut costs, reduce risk, and drive business profit. For the four fiscal quarters ended December 31, 2009, BMC revenue was approximately \$1.90 billion. Visit [www.bmc.com](http://www.bmc.com) for more information.

### **About the Authors**

Ben Newton is the technical director for Department of Defense and Intelligence Community at BMC Software, where he manages technical requirements for BMC's defense and intelligence community customers. Newton came to BMC with the acquisition of BladeLogic, Inc., where he was a senior application engineer. For the last nine years, he has specialized in the various aspects of data center automation, particularly application release and compliance automation. Before joining BladeLogic, he worked as a systems architect for EDS and Northrop Grumman, where he was the lead architect for the design of the Army Knowledge Online (AKO) Disaster Recovery project. He graduated in 2000 from Cornell University with a master's degree in computer science.

Herb VanHook is vice president of strategy in the Office of the CTO at BMC Software, with particular focus on the technology impacts and opportunities presented by cloud computing models. He has worked in strategic, corporate development, and business planning functions since joining BMC in 2005. Previously, he held several executive positions at industry analyst firm META Group (now Gartner, Inc.), including executive vice president and research director, and last serving as META's interim president and chief operating officer. VanHook has more than 30 years of experience in information technology — across operations, development, support, and management — including positions at IBM, Computer Associates, and Legent Corporation.

