

### **Introduction**

**Use Case 1: Monitoring Newly Provisioned Servers**

**Use Case 2: Ensuring Accurate Asset Inventory Across Multiple Management Systems**

**Use Case 3: Providing Standard Application Server Services to Developers**

**Use Case 4: Enabling Rapid Data Center Migration**

**Use Case 5: Accelerating Time to Repair via Integrated OSS Systems**

**Use Case 6: Automating Web Server Capacity Additions**

**Use Case 7: Expanding Service Capacity under Business Constraints**

## Introduction

DCML enables any IT organization to benefit from the utility computing vision of lower costs, increased agility, and improved service levels. Using DCML, companies now have a standard method to stitch together multiple management systems and tools to form a unified view of the environment. In this unified view, management systems can exchange domain knowledge about the environment with other management systems in the same environment. By providing the ability to exchange information about the data center environment, DCML can make the vision of utility computing a reality.

The following sections detail use cases that express how the DCML Specification, when implemented in these real-world scenarios, will deliver value to any IT organization by enabling previously expensive, error prone tasks to be performed automatically in a multi-vendor, heterogeneous data center environment.

## Use Case 1: Monitoring Newly Provisioned Servers

### Problem Statement

When the ABC Corp IT department server management group provisions a new server to run an application for one of its many business units, the group is responsible for alerting the ABC Corp IT monitoring group so they can identify it as a new server that needs to be monitored. The notification will also enable the ABC Corp network operations center to know when the server goes down and respond accordingly.

The ABC Corp provisioning system is fairly sophisticated. In addition to installing operating systems and software infrastructure on servers, it tracks the business application the server is destined to serve and the service level required by that application. This information is also designed to be input into the monitoring system.

ABC Corp's cross-system communication typically relies on ad hoc personal communication or manual data entry, processes that can break down due to forgetfulness or a lost email. Unfortunately, when servers go unmonitored and something is down, customers are often the first to notice. This customer service problem is embarrassing for ABC Corp, leading to lost revenue, poor customer satisfaction, and potential increased costs.

### Success Criteria

Success at ABC Corp hinges on consistently following one simple rule: A server is never provisioned and put into production without being monitored. This rule ensures that ABC Corp's NOC is the first to know when there's a problem, enabling them to immediately take corrective action, increasing uptime and customer satisfaction. For every alert the NOC receives, NOC personnel should be able to identify which business application is affected and how critical it is to get the server back up and working. By being well informed, NOC personnel can more effectively prioritize their work and quickly notify the appropriate business owners of problems.

### Solution Overview

ABC Corp proposes to solve this problem by linking its provisioning system to its monitoring system. DCML plays a key role in this linkage. Every time a machine is provisioned, a DCML document containing information about the provisioned machine will be transmitted automatically to the monitoring system, causing monitoring of the machine to be initiated. The DCML document contains information about the machine's IP address and any special ports and services running on the machine that are to be monitored. The DCML document also contains information about what application the server belongs to and who to contact, as well as how to contact them, when it goes down. This information is imported into the system that is available to NOC personnel for consultation when an alert occurs.

## Use Case 2: Ensuring Accurate Asset Inventory Across Multiple Management Systems

### Problem Statement

ABC Corp uses an IT asset management system to try and maintain an accurate device (e.g., server, desktop, network device, etc.) inventory across many systems. ABC Corp also has many other management applications (e.g., monitoring, change management, etc.) that require frequent synchronization with the asset management system.

Due to lack of integration resources and non-standard APIs across these applications, there are no automated synchronization mechanisms available. As a result, this synchronization typically occurs manually, infrequently, and results in an incomplete picture of assets under management across all of ABC Corp's IT management systems.

### Success Criteria

Success at ABC Corp relies on the IT asset management system, which serves as the central system of record for all devices under management. As such, other systems at ABC Corp need the ability to query the asset management system for the latest list of devices in inventory. This eliminates the problems of double entry and multiple, inconsistent device records across all management systems.

### Solution Overview

The asset management system would provide the ability to be queried for the latest inventory, and return the data in a DCML compliant format. The administrator configures the monitoring system to query the asset management system every four hours, receiving the latest inventory update and synchronizing it with the monitoring system. The monitoring system retrieves the inventory as a DCML document, and compares that document against its current inventory (which also can be represented in a DCML document), making the changes indicated by the differences.

## Use Case 3: Providing Standard Application Server Services to Developers

### Problem Statement

ABC Corp's IT architecture standards organization has made a commitment to the J2EE framework for all new Web applications. ABC Corp has many disparate application development organizations, many of which are new to n-tier application development and deployment best practices. Requiring each development organization to not only develop their application, but also configure and deploy a complete J2EE application server environment to host their application, is wasteful on every level (e.g., time, resources, maintainability, etc.).

### Success Criteria

Success for the ABC Corp IT architecture team and their distributed development teams would entail the delivery of a standardized, self-describing set of application server services for deployment of any newly developed J2EE application. This set of services would make it easier for developers to quickly understand the application server environment and its corresponding services. Developers could then ensure that their applications took advantage of these standard services, enabling them to concentrate on their primary responsibilities, designing and developing the right application for the problem at hand. Additionally, the IT architecture and production deployment teams will have additional time to focus their energies on ensuring the deployment environment is reliable, available, and scalable.

## Solution Overview

To facilitate a developer's ability to quickly understand the application server environment and its corresponding services, the IT architecture team creates a profile of the standard environment via a DCML document that serves as the canonical reference for Web application development.

Consider the case of a J2EE application server, such as BEA WebLogic or IBM WebSphere. These application servers may use different conventions for specifying application server services and configuration information. Rather than leave a developer to figure out this information on his/her own, DCML is used to provide a normalized, unified description of the application server services.

The following examples depict the type of information needed to describe a collection of application server services:

- Application server administration information – owner, contact information, etc.
- Maintenance window
- Standard deployment directory
- JDK version required or available
- Java run time flags turned on – details on garbage collection, heap size, etc.
- Application server specific configuration / property files
- Database information – server name, IP, database type, database connection pooling, etc.
- J2EE services and service related information – HTTP, EJBs, JNDI, JMS, etc.
- Log file information – location of application log files, log file rotation, etc.
- Network connectivity information – needed firewall rules between application layer and database layer

Using a DCML document that already contains all of this information, developers can more easily design and implement J2EE applications that leverage common application server services.

## Use Case 4: Enabling Rapid Data Center Migration

### Problem Statement

ABC Corp is expanding its customer base and opening a second data center in another location from its primary. The original data center had servers for several different functions, but the new data center will initially only support one specific function, namely the on-line purchasing system. ABC Corp's IT service delivery group would like to use their service provisioning software to quickly migrate the on-line purchasing system to the new data center – automatically installing the correct operating systems, patches, and applications onto the new hardware. Unfortunately, this task involves a significant amount of manual labor including gathering detailed hardware information (e.g., make, model, memory, network card, IP address, customer, etc.), mapping OS, patch, and configuration settings from the existing environment to the new environment, and gathering that data across multiple sources via multiple interfaces.

### Success Criteria

With minimum effort, ABC Corp's environment in the new data center is provisioned quickly and accurately, is secure, and the on-line purchasing system has been properly migrated to the new environment with the correct software elements. ABC Corp's data center is populated with DCML-compliant servers, networking equipment, and storage devices. In addition, ABC Corp's provisioning software is DCML-compliant.

### Solution Overview

The administrator will retrieve either from a software system or from a document library a DCML specification of the existing data center, specifically focused on the complete on-line purchasing service – hardware, software, network, storage, etc. The administrator will import this DCML document into ABC Corp's IT service provisioning system. The administrator will have the ability to map this existing DCML service description to the new environment and simulate the proposed deployment, troubleshooting any potential problems before changes are committed. Once satisfied, the automated service provisioning process can be initiated, ensuring that the correct server, network, and storage elements will be provisioned and configured, and the application will be deployed and available in the new data center.

## Use Case 5: Accelerating Time to Repair via Integrated OSS Systems

### Problem Statement

ABC Corp network operations center personnel get an alert signaling that a storage device is about to fail on a server. The support engineer needs to quickly determine the machine configuration, customer, SLA, application, location, and break-fix plan in order to repair the system. Unfortunately, all of this information exists across multiple systems, hard-copy documents, and in people's heads.

### Success Criteria

The support engineer will leverage a fully integrated operational support system (OSS) to quickly determine the "who, what, when, where, why" of the server and IT service delivered by that server, determine the appropriate course of action, both short term and long-term, and remediate the situation before it becomes a significant service outage.

### Solution Overview

ABC Corp's IT management and automation systems are DCML-compliant. As a result, when the support engineer receives a notification of a drive failure on the server foo.abccorp.com in his monitoring system, he can quickly query the help desk system for the status of foo.abccorp.com. The help desk application queries the asset management system for the machine specifications. The asset management system returns a DCML document containing the machine configuration identifying the required disk drive replacement.

Because of the integrated OSS, the support engineer can choose to either quickly replace the drive or re-map the service to leverage available network-attached storage, eliminating any downtime. Regardless of the option, DCML can be used to exchange essential data between these OSS systems - management, monitoring, provisioning, ticketing, etc.

## Use Case 6: Automating Web Server Capacity Additions

### Problem Statement

ABC Corp runs a distribution partner e-commerce site to enable online ordering of products. In the last week of each quarter, demand spikes tenfold. However, due to a special sale, ABC Corp is seeing a fifteen-fold increase in order volume earlier than anticipated. The system administrator needs to quickly add ten more web servers to handle increased site requirements. The new servers have already been purchased and installed and now must be configured.

### Success Criteria

The system administrator can leverage her configuration and change management system to rapidly provision the new servers, ensuring they all have the required software and are configured to the specifications provided in the DCML blueprint. In addition, these servers will be automatically added to the load balancer without any interruption to the existing site capacity.

### Solution Overview

The administrator opens a change management console. She chooses from the site blueprint a specification for the particular IT service and dedicated web server. The blueprint is described in DCML and leverages data from existing systems management, monitoring, and provisioning tools, also provided in DCML format. In this case, an inventory specification for this service is returned. The administrator edits the specification as needed, and submits it as a blueprint for the web server application. The updated blueprint is applied to the target machine and the change management software automatically installs and configures the target.

## Use Case 7: Expanding Service Capacity under Business Constraints

### Problem Statement

ABC Corp is seeing early adoption of its new web-based self care solution. The company wants to expand their subscriber base without procuring additional hardware to handle the increase in users.

### Success Criteria

ABC Corp can use DCML to increase its web-serving capability without adding capex cost to the business (e.g. through the purchase of new hardware).

### Solution Overview

The ABC Corp solution architect can create DCML Rules to quickly ascertain what is required to increase the subscriber base, leveraging existing dynamic load-increasing statements. From a product description system, the architect can identify which DCML Blueprints are required as part of the web-based self care solution, including the range of options for building the web server (e.g., whether it uses IIS or Apache, Windows or Solaris or Linux, etc.).

The architect can then determine whether ABC Corp has sufficient hardware and software licenses for the Blueprint elements, using a DCML inventory report from an IT asset management system to identify hardware and software availability. In this case, it is determined that enough hardware and software is available. DCML's Blueprint Relationship further ensures that ABC Corp will use the best, logical combinations of assets (e.g., IIS running Solaris is not a sensible combination). These findings can be verified against other Rules specifying preferred configurations (both technically and commercially) to deliver the required incremental capacity to support the expansion of ABC Corp's subscriber base.