

# Implementing Support and Monitoring For a Business-Critical Application Migrated to Windows Azure

Published: August 2011

**Microsoft IT had recently migrated BCWeb—a complex, business-critical application—to the Windows Azure™ platform. To ensure ongoing application availability, the team needed to implement a reliable and comprehensive monitoring and support solution for BCWeb. Microsoft IT accomplished this by combining the Windows Azure integration and monitoring capabilities with the Microsoft® System Center Operations Manager management capabilities.**

## Target Audience

IT Decision-maker, IT Implementer

## Situation

Microsoft IT needed to create and implement a support and monitoring solution for BCWeb—an enterprise application that was recently migrated to the Windows Azure platform.

## Solution

Microsoft IT leveraged the Windows Azure platform's flexibility and extensibility with the System Center Operations Manager 2007 R2 integration capabilities to provide a comprehensive, centralized, and manageable support and monitoring system for BCWeb.

## Benefits

- A consolidated management and support environment under System Center Operations Manager 2007 R2
- Accurate and timely monitoring and alerting for BCWeb critical components
- A large number of reusable monitoring components that can be leveraged in future Windows Azure applications
- Best practices to apply to future Windows Azure applications

## Introduction

Business Case Web (BCWeb) is an internal, web-based application that Microsoft uses to create the business case for product pricing exemptions. BCWeb is composed of three distinct application components: the core BCWeb component, the Workflow Routing and Approval system (WRAP), and Rapport. The core BCWeb component is responsible for providing a user interface, and for the underlying functionality that enables users to generate business cases for pricing exceptions. WRAP routes the pricing exception requests for approval within the Microsoft corporate infrastructure. Rapport provides a user interface for the WRAP approval process.

BCWeb has a user base of 2,500 internal Microsoft employees. In 2010, Microsoft used BCWeb to process approximately 27,000 pricing exception requests.

## BCWeb Platform Overview

BCWeb was migrated to Windows Azure as a pilot project to develop and capture best practices for migrating enterprise applications to Windows Azure. The core BCWeb components are hosted on the Windows Azure platform. However, BCWeb is also integrated with a number of components that are hosted on the Microsoft IT corporate network, and are external to the Windows Azure platform.

## Situation

The primary reason for migrating BCWeb to Windows Azure was as a migration pilot project. However, BCWeb was also experiencing performance and reliability issues in its previous environment. Although the Windows Azure migration brought increased reliability and performance to BCWeb, ongoing tuning of the application environment was required. Microsoft IT realized that it needed a comprehensive monitoring solution to enable ongoing reliability, and to measure internally established service level agreements (SLAs).

## BCWeb Architecture

BCWeb is divided into three distinct Windows Azure Services, which in turn house the main application components: BCWeb, WRAP, and Rapport. The three applications are separated by design to enable a modular approach to application updates and refactoring.

### **Windows Azure Components**

The first component application—the BCWeb core—is implemented as a Windows Azure Web role that hosts the UI for generating business case documents. BCWeb uses two Worker roles: the first Worker role hosts the core BCWeb Service and other Windows Communication Foundation (WCF)–based services, and the second Worker role hosts background and notification processes used by the BCWeb application. The WRAP application is implemented as a multi–instance Worker role that contains all of the necessary services required to perform the routing and approval operations for BCWeb–generated business case documents. The Rapport Windows Azure Service hosts the Rapport application. Rapport is composed of a Web role that hosts the UI, and a Worker role that hosts the Rapport Windows Communication Foundation (WCF) Service. SQL Azure databases host native data storage for the entire BCWeb application infrastructure.

### **On-Premises Distributed Components**

BCWeb includes several critical components that are not hosted on the Windows Azure platform. These components primarily provide access to external data that is required for BCWeb functionality. The two primary external components are SAP (for business data), and the Microsoft corporate Active Directory® Domain Services database (for infrastructure and organizational data). Both of these components are outside the management scope of BCWeb, but are critical to its functionality. Both components are also hosted on-premises within the Microsoft corporate network. An on-premises database—the Licensing Information Repository (LIR)—hosts information used for data warehousing. The BCWeb transactional SQL Azure databases export information on an ongoing basis to the on-premises LIR database (hosted on Microsoft SQL Server®)—for reporting purposes.

## BCWeb Windows Azure Architecture Diagram

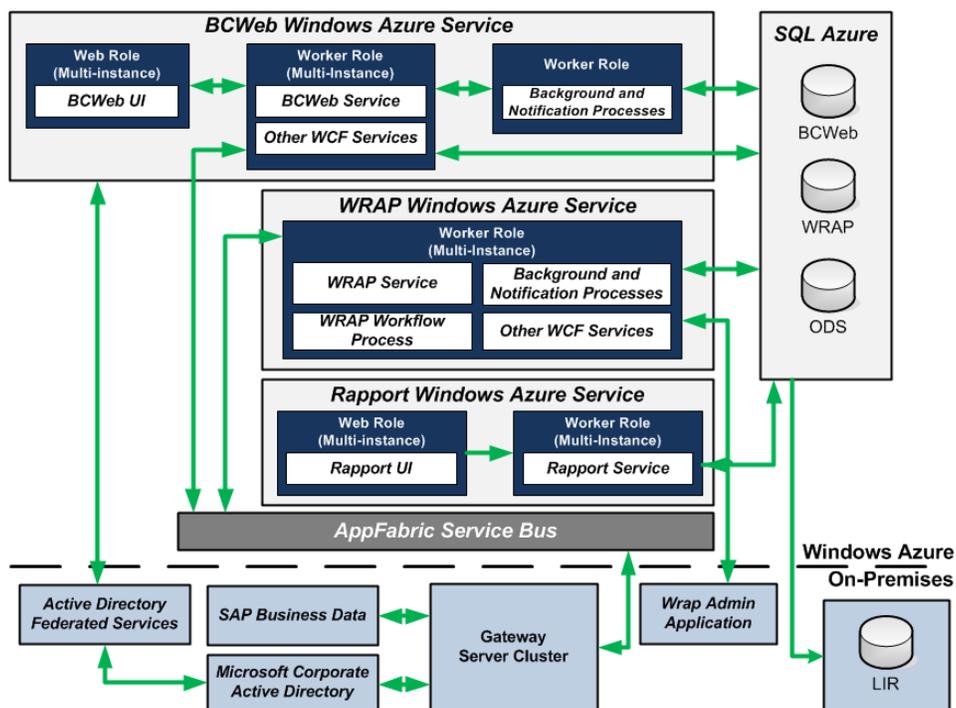


Figure 1. BCWeb Windows Azure Architecture

## Solution

Microsoft IT knew that implementing a support and monitoring solution for BCWeb would be a challenging task. The BCWeb migration to Windows Azure meant that the support and monitoring processes used with the previous BCWeb version would require reassessment and redesigning to accommodate the new application infrastructure.

## Design Goals

Microsoft IT began planning for the BCWeb support and monitoring solution with several general design goals in mind:

- The solution must provide support and monitoring for all critical aspects of BCWeb functionality, including components hosted on the Windows Azure platform, and components hosted on-premises that are external to Windows Azure.
- BCWeb monitoring should be centralized and consolidated into one management console.
- The solution should leverage existing Microsoft IT infrastructure as much as possible
- Windows Azure-based monitoring components should be used as much as possible.

## Providing Support for a Distributed Application

The new version of BCWeb contained both components from the Microsoft corporate network, and components from the Windows Azure platform. As a result, several changes to the previous support model were required.

The distributed nature of BCWeb on the Windows Azure platform forced Microsoft IT to reassess the methods used to support the application. In the previous BCWeb version, the scope of support was limited to the Microsoft corporate network. One of the important considerations when leveraging Windows Azure for internal enterprise applications is that corporate network users connect to resources outside of the of the network (Windows Azure) to run "internal" applications.

In the BCWeb Windows Azure version, the following components and their associated support teams became part of the application's support infrastructure:

- Windows Azure - core application
- SQL Azure - data storage
- Active Directory Federation Services (AD FS) - authentication
- The Microsoft corporate internet connection - access to Windows Azure components

These systems would need to be incorporated into the BCWeb support model, and the previously established SLAs would require reassessment to reflect the BCWeb support requirements' increased complexity.

The BCWeb team was still the contact point for end users, but BCWeb support now relied on the Windows Azure platform support team, the AD FS support team, and the Microsoft IT network support team, to provide support for their associated systems.

As a result, the following areas needed reassessment:

- **SLAs for response and resolution time.** The BCWeb support team had to include the response times for the other support teams in its overall response and resolution time SLAs.
- **SLAs for performance and availability.** BCWeb application SLAs needed to integrate performance and availability benchmarks from all integrated components. Performance and availability for BCWeb was now subject to the performance and availability of several components outside the control of the BCWeb team.

The support team quickly discovered that with a hybrid application, support complexity and dependencies increase as more third-party components are involved. All of these components had an impact on the BCWeb end-to-end SLAs.

### **Determining Key Points of Failure**

The first task in establishing a reliable and comprehensive monitoring solution for BCWeb was to determine the key points of failure for the application. The BCWeb support team identified the key points of failure within BCWeb, and then put the appropriate monitoring processes in place to either prevent failure, or quickly identify when a failure occurred.

When Microsoft IT designed the monitoring solution, these Points of failure were the first aspects of BCWeb that they addressed.

### **Designing Operational Monitoring for BCWeb**

Microsoft IT outlined the following general monitoring requirements for BCWeb:

- **Error logging.** Record warning and error-related messages from all applicable components.

- **Platform monitoring.** Monitor important aspects of Windows Azure platform health, including:
  - Operating system/SQL/Internet Information Services health
  - Services health
  - Disk capacity
  - Basic performance counters
- **Application monitoring.** Monitor performance and reliability for all critical aspects of BCWeb application functionality.
- **Key external services monitoring.** Monitor performance and availability of connections with external services including:
  - SAP
  - AD DS

When considering monitoring methods for BCWeb, Microsoft IT identified that the Windows Azure platform could not natively support the level of monitoring that BCWeb would require. Additionally, the on-premises components outside of Windows Azure would need monitoring. Thus, Microsoft IT required a monitoring solution that would allow the BCWeb support team to accurately assess the application's condition based on all of its various components.

### **Leveraging System Center Operations Manager 2007 R2 to Consolidate Monitoring and Support**

Microsoft IT decided to use System Center Operations Manager 2007 R2 to monitor the new version of BCWeb. Microsoft IT chose System Center Operations Manager for the following reasons:

- Monitoring could be centralized into one console, and consolidated to include Windows Azure and on-premises components.
- BCWeb used System Center Operations Manager–compliant instrumentation (Windows Events and Performance Counters).
- System Center Operations Manager was already in use in the environment, thus no significant time or capital investment was required.
- Using System Center Operations Manager limited the amount of custom coding required.
- System Center Operations Manager already had available a Windows Azure Management Pack that provided monitoring solutions for some of the BCWeb key components.

### **Using, Extending, and Creating System Center Operations Manager Functionality**

Microsoft IT identified four key BCWeb-monitoring categories:

- End-user perspective and SLA requirements
- Web and Worker role performance
- Application health
- SQL Azure performance and state

Microsoft IT approached each of these categories differently using System Center Operations Manager.

## End-User Perspective and SLA Requirements

Microsoft IT used the System Center Operations Manager Web Application template to enable scripted website navigation that mimicked typical end-user interactions with the different BCWeb UI components. This enabled the team to monitor true availability of the web applications and implement alerts. It also enabled Microsoft IT to collect historical availability data to compare with established SLAs.

## Web and Worker Role Performance

The development team discovered that the built-in Windows Azure Diagnostics feature could provide a large amount of diagnostic information regarding the state of the Windows Azure Compute roles—the Web and Worker roles in the case of BCWeb. When the development team combined System Center Operations Manager with the Windows Azure Management Pack, they were able to access a large number of performance counters and events that contained the information they needed about the Web and Worker roles. By building trending and alerting functionality, the team was able to monitor the health of the Compute roles. The team used the Windows Azure Management Pack to:

- Discover each Windows Azure application.
- Provide status of each Windows Azure role instance.
- Collect and monitor Windows Azure performance information.
- Collect and monitor Windows events.
- Collect and monitor the Microsoft .NET Framework trace messages from each Windows Azure role instance.
- Selectively delete performance, event, and .NET Framework trace data from the Windows Azure storage account to manage storage space.

## Application Health

The overall health of BCWeb depends on several components, including Windows Azure. To monitor the Windows Azure part of BCWeb, and address some of the aspects of the BCWeb application that were not natively monitored by the Windows Azure Management Pack—especially monitoring on-premises components—the development team extended the capabilities of the Windows Azure Management Pack to monitor key aspects of application health. Specifically, they created performance counters that monitored application-specific items such as requests to ASP.NET Application objects and .NET Framework CLR exceptions. The development team also extended the Windows Azure management pack to monitor business logic exception events when accessing on-premises components.

For on-premises components, the development team also leveraged built-in .NET Framework components to monitor application health through performance and historical trends. For example, the team planned to use the **StopWatch** class to time calls to the SAP web service, and then represent the results as a performance counter that System Center Operations Manager could then monitor.

## SQL Azure Performance and State

One large deficiency in the available solutions through System Center Operations Manager was the lack of any monitoring capability for SQL Azure.

In the previous version of BCWeb, a large portion of system monitoring used tools native to SQL Server. Unfortunately, three key legacy BCWeb tools were not available on SQL Azure:

**Table 1. SQL Azure Component Comparison**

| SQL Server Component | Feature Purpose   | Feature Status on SQL Azure |
|----------------------|---|-----------------------------|
| SQL Agent            | Manage and execute automated tasks (SQL Server jobs)              | Not Available               |
| SQL Profiler         | Capture and analyze SQL Server performance data                   | Not Available               |
| DMVs                 | Provide diagnostic and configuration information about SQL Server | Partially Available         |

As a result of these discrepancies, the development team elected to build a custom management pack using both historical trending and threshold alerting to monitor the health and performance of SQL Azure

For example, the team created a performance counter that measured the size of a SQL Azure database using a Transact-SQL (TSQL) query. System Center Operations Manager collected this data daily, using the following script.

```
SELECT SUM(reserved_page_count)*8.0/1024
FROM sys.dm_db_partition_stats;
GO
```

The development team also used the following T-SQL script that provided the number of connections to a SQL Azure database.

```
SELECT Count(*)
FROM sys.dm_exec_sessions
```

The result of this script was a performance counter that System Center Operations Manager monitored every five minutes.

Additionally, the development team examined the application code for references to DMV information that was not available in SQL Azure, and then refactored the code to remove the references and retrieve the information from alternate DMV locations in SQL Azure.

## Benefits

Microsoft IT used System Center Operations Manager 2007 R2, the Windows Management Pack for System Center Operations Manager, and custom-designed performance counters within Windows Azure to realize the following benefits:

- A consolidated management and support environment within System Center Operations Manager 2007 R2
- Accurate and timely monitoring and alerting for BCWeb critical components
- A large number of reusable monitoring components that can be leveraged in future Windows Azure applications

## Best Practices

Microsoft IT established the following best practices when implementing Windows Azure monitoring:

- Use System Center Operations Manager 2007 R2 and the Windows Azure Management Pack for consolidated and centralized application monitoring.
- Extend or create management packs for non-Azure application components.
- Create custom monitoring components for SQL Azure.
- Use Worker roles to host custom code for application monitoring.
- Develop applications with the most recent version of the Windows Azure Software Development Kit (SDK) to implement the newest monitoring features.

## Conclusion

By using System Center Configuration Manager 2007 R2, the Windows Management Pack for System Center Operations Manager, and custom-designed management pack components, Microsoft IT was able to provide a robust and centralized monitoring environment for BCWeb.

The solution included monitoring of the BCWeb Windows Azure-based components, and the critical aspects of on-premises components that were not native to Windows Azure. Microsoft IT also captured numerous best practices that will be used in future distributed application migrations.

## Products & Technologies

- Windows Azure Web role
- Windows Azure Worker role
- Windows Azure AppFabric
- SQL Azure
- Microsoft SQL Server 2008 R2
- Microsoft Visual Studio® 2010
- Windows Azure SDK 1.4
- System Center Operations Manager 2007 R2
- Windows Azure Management Pack for Operations Manager

## For More Information

For more information about Microsoft products or services, call the Microsoft Sales Information Center at (800) 426-9400. In Canada, call the Microsoft Canada Order Centre at (800) 933-4750. Outside the 50 United States and Canada, please contact your local Microsoft subsidiary. To access information via the World Wide Web, go to:

<http://www.microsoft.com>

<http://www.microsoft.com/technet/itshowcase>

© 2011 Microsoft Corporation. All rights reserved.

Microsoft, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. The names of actual companies and products mentioned herein may be the trademarks of their respective owners.

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY.