Technosoft Solutions

Microsoft Azure

Introduction paper

Abstract

Microsoft Azure is Microsoft's application platform for the public cloud. The goal of this paper is to give a foundation for understanding the fundamentals of Azure.

Introduction

What is Azure?

Microsoft Azure (formerly Windows Azure) is a cloud computing facility created by Microsoft for building, testing, deploying, and managing applications and services through a global network of Microsoft-managed data centers. It provides software as a service (SAAS), platform as a service and infrastructure as a service and supports many different programming languages, tools and frameworks, including both Microsoft-specific and third-party software and systems.

What is cloud computing?

Cloud computing can be divided into three levels depending on the services offered by providers. From the innermost level to the outermost Cloud hosting services include: Infrastructure as a Service, Platform as a Service and Software as a Service. A brief description of each of these levels:

Infrastructure as a Service (IaaS – Infrastructure as a Service)

Infrastructure as a Service is a provision model in which an organization placed 'outside' the equipment used to support operations, this includes information storage, hardware, servers and networking components. Service provider. IaaS is sometimes also referred to as Hardware as a Service or HaaS "

The most obvious advantage of using an IaaS is to transfer to the supplier management problems related to computer equipment. Another attractive advantage is the reduction of costs, as generally in the technologies associated with cloud computing, to pay only for what is consumed. Infrastructure as a Service also allows scalability virtually automatic and transparent to the consumer, leaving the responsibility to service providers.

Other features are: providing Internet connectivity, policy-based services and the provision of a virtual desktop.

Platform as a Service (PaaS – Platform as a Service)

Cloud computing and its rapid growth has required "include platforms for building and running custom applications, this concept is known as PaaS. PaaS applications are also known as on-demand Web-based or SaaS solutions.

The provider, in addition to solving problems in the hardware infrastructure, the software also handles. The client uses this kind of solutions does not need to install , configure or maintain operating systems, databases and application servers as this is provided under this platform.

A platform as a service (PaaS) solves more problems when compared to a solution that only provides an infrastructure as a service (IaaS), as it has many limitations related to the runtime environment. Among these are the type of system, the programming language (in some cases they may use libraries), the database manager.

Software as a Service (SaaS – Software as a Service)

It is the best known of the levels of cloud computing. SaaS is a software distribution model that provides customers access to it through the network. Thus, they do not have to worry about the configuration, deployment and maintenance of applications, because all these tasks become the responsibility of the vendor. Distributed applications through a Software as a Service model can reach any company regardless of size or geographic location.

This model aims to end customers using the software to meet your organization's processes. The Software as a Service (SaaS) can be described as that application consumed over the Internet, usually through the browser, which payment is conditioned to use it and where the application logic and data reside on the platform of supplier. Examples of SaaS are Google App etc.

Services by Azure

Azure offers more than 600 services.

Virtual machines, infrastructure as a service (IaaS) allowing users to launch general-purpose Microsoft Windows and Linux virtual machines, as well as preconfigured machine images for popular software packages.

App services, platform as a service (PaaS) environment letting developers easily publish and manage Web sites.

Websites, high density hosting of websites allows developers to build sites using ASP.NET, PHP, Node.js, or Python and can be deployed using FTP, Git, Mercurial, Team Foundation Server or uploaded through the user portal. This feature was announced in preview form in June 2012 at the Meet Microsoft Azure event. Customers can create websites in PHP, ASP.NET, Node.js, or Python, or select from several open source applications from a gallery to deploy. This comprises one aspect of the platform as a service (PaaS) offerings for the Microsoft Azure Platform. It was renamed to Web Apps in April 2015.

WebJobs, applications that can be deployed to a Web App to implement background processing. That can be invoked on a schedule, on demand or can run continuously. The Blob, Table and Queue services can be used to communicate between Web Apps and Web Jobs and to provide state.

Mobile Engagement collects real-time analytics that highlight users' behavior. It also provides push notifications to mobile devices.

Storage Services provides REST and SDK APIs for storing and accessing data on the cloud.

Table Service lets programs store structured text in partitioned collections of entities that are accessed by partition key and primary key. It's a NoSQL non-relational database.

Blob Service allows programs to store unstructured text and binary data as blobs that can be accessed by a HTTP(S) path. Blob service also provides security mechanisms to control access to data.

Queue Service lets programs communicate asynchronously by message using queues.

File Service allows storing and access of data on the cloud using the REST APIs or the SMB protocol.

Azure Deployment Options

Azure App Service maintains the application framework for you (ASP.NET, PHP, Node.js, etc). Some frameworks are enabled by default while others, like Java and Python, may need a simple checkmark configuration to enable it. In addition, you can customize your application framework, such as the PHP version or the bitness of your runtime.

All the deployment methods use one of the following processes:

- FTP or FTPS: Use your favorite FTP or FTPS enabled tool to move your files to Azure, from FileZilla to full-featured IDEs like NetBeans. This is strictly a file upload process. No additional services are provided by App Service, such as version control, file structure management, etc.

- Kudu (Git/Mercurial or OneDrive/Dropbox): Kudu is the deployment engine in App Service. Push your code to Kudu directly from any repository. Kudu also provides added services whenever code is pushed to it, including version control, package restore, MSBuild, and web hooks for continuous deployment and other automation tasks. The Kudu deployment engine supports 3 different types of deployment sources:

    - Content sync from OneDrive and Dropbox
    - Repository-based continuous deployment with auto-sync from GitHub, Bitbucket, and Visual Studio Team Services

    - Repository-based deployment with manual sync from local Git

- Web Deploy: Deploy code to App Service directly from your favorite Microsoft tools such as Visual Studio using the same tooling that automates deployment to IIS servers. This tool supports diff-only deployment, database creation, transforms of connection strings, etc. Web.

- Deploy differs from Kudu in that application binaries are built before they are deployed to Azure. Similar to FTP, no additional services are provided by App Service.

Popular web development tools support one or more of these deployment processes. While the tool you choose determines the deployment processes you can leverage, the actual DevOps functionality at your disposal depends on the combination of the deployment process and the specific tools you choose. For example, if you perform Web Deploy from Visual Studio with Azure SDK, even though you don't get automation from Kudu, you do get package restore and MSBuild automation in Visual Studio.

Deploy manually by uploading files with FTP

Because many FTP tools don't provide diff-only copying and simply copy all the files. If you are used to manually copying your web content to a web server, you can use an FTP utility to copy files, such as Windows Explorer or FileZilla.1
The pros of copying files manually are:

- Familiarity and minimal complexity for FTP tooling.
- Knowing exactly where your files are going.
- Added security with FTPS.

The cons of copying files manually are:

- Having to know how to deploy files to the correct directories in App Service.
- No version control for rollback when failures occur.
- No built-in deployment history for troubleshooting deployment issues.
- Potential long deployment times

Deploy by syncing with a cloud folder

A good alternative to copying files manually is syncing files and folders to App Service from a cloud storage service like OneDrive and Dropbox. Syncing with a cloud folder utilizes the Kudu process for deployment (see Overview of deployment processes).

The pros of syncing with a cloud folder are:

- Simplicity of deployment. Services like OneDrive and Dropbox provide desktop sync clients, so your local working directory is also your deployment directory.
- One-click deployment.
- All functionality in the Kudu deployment engine is available (e.g. package restore, automation).

The cons of syncing with a cloud folder are:

- No version control for rollback when failures occur.
- No automated deployment, manual sync is required.

Deploy continuously from a cloud-based source control service

If your development team uses a cloud-based source code management (SCM) service like Visual Studio Team Services, GitHub, or BitBucket, you can configure App Service to integrate with your repository and deploy continuously.

The pros of deploying from a cloud-based source control service are:

- Version control to enable rollback.
- Ability to configure continuous deployment for Git (and Mercurial where applicable) repositories.
- Branch-specific deployment, can deploy different branches to different slots.
- All functionality in the Kudu deployment engine is available (e.g. deployment versioning, rollback, package restore, automation).

The con of deploying from a cloud-based source control service is:

- Some knowledge of the respective SCM service required.

How to deploy continuously from a cloud-based source control service

In the Azure Portal, you can configure continuous deployment from GitHub, Bitbucket, and Visual Studio Team Services.

- Continuous Deployment to Azure App Service.

Deploy from local Git

If your development team uses on-premises local source code management (SCM) service based on Git, you can configure this as a deployment source to App Service.

Pros of deploying from local Git are:

- Version control to enable rollback.
- Branch-specific deployment can deploy different branches to different slots.
- All functionality in the Kudu deployment engine is available (e.g. deployment versioning, rollback, package restore, and automation).

Cons of deploying from local Git is:

- Some knowledge of the respective SCM system required.
- No turn-key solutions for continuous deployment.

Deploy using an IDE

If you are already using Visual Studio with an Azure SDK, or other IDE suites like Xcode, Eclipse, and IntelliJ IDEA, you can deploy to Azure directly from within your IDE. This option is ideal for an individual developer.

Visual Studio supports all three deployment processes (FTP, Git, and Web Deploy), depending on your preference, while other IDEs can deploy to App Service if they have FTP or Git integration (see Overview of deployment processes).

The pros of deploying using an IDE are:

- Potentially minimize the tooling for your end-to-end application life-cycle. Develop, debug, track, and deploy your app to Azure all without moving outside of your IDE.

The cons of deploying using an IDE are:

- Added complexity in tooling.
- Still requires a source control system for a team

project. Additional pros of deploying using Visual Studio

with Azure SDK are:

- Azure SDK makes Azure resources first-class citizens in Visual Studio. Create, delete, edit, start, and stop apps, query the backend SQL database, live-debug the Azure app, and much more.
- Live editing of code files on Azure.
- Live debugging of apps on Azure.
- Integrated Azure explorer.
- Diff-only deployment.

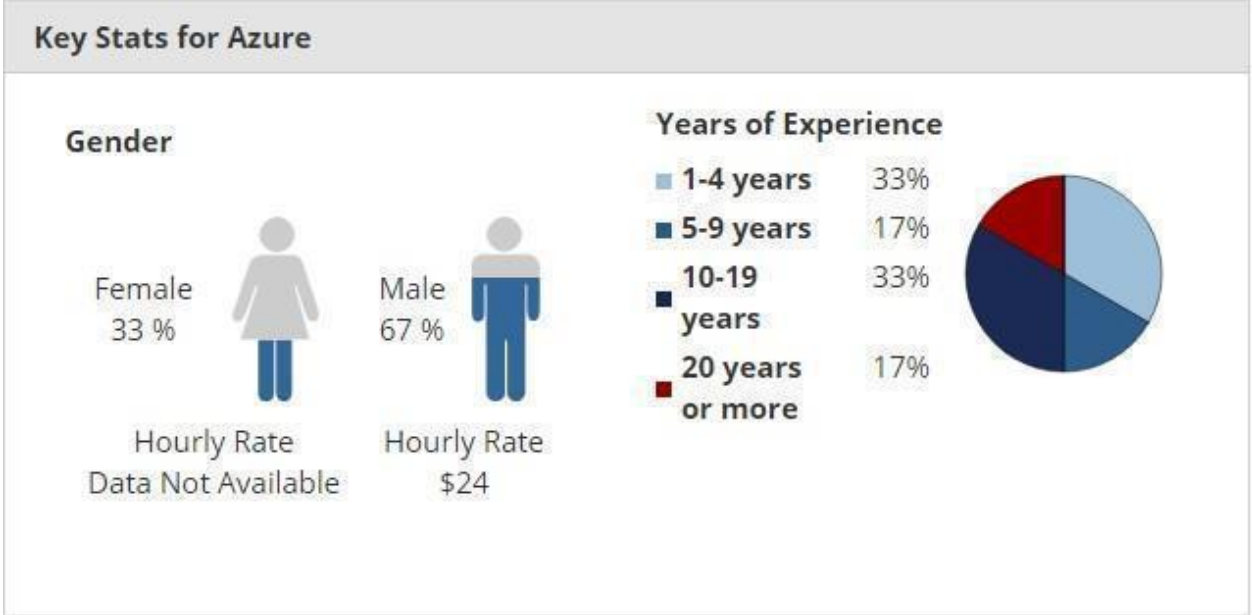
Automate deployment by using command-line tools

If you prefer the command-line terminal as the development environment of choice, you can script deployment tasks for your App Service app using command-line tools. Pros of deploying by using command-line tools are:

- Enables scripted deployment scenarios.

- Integrate provisioning of Azure resources and code deployment.
- Integrate Azure deployment into existing continuous integration scripts.
  Cons of deploying by using command-line tools are:
- Not for GUI-preferring developers.

Hourly rates on Azure



(Picture courtesy: http://www.payscale.com)

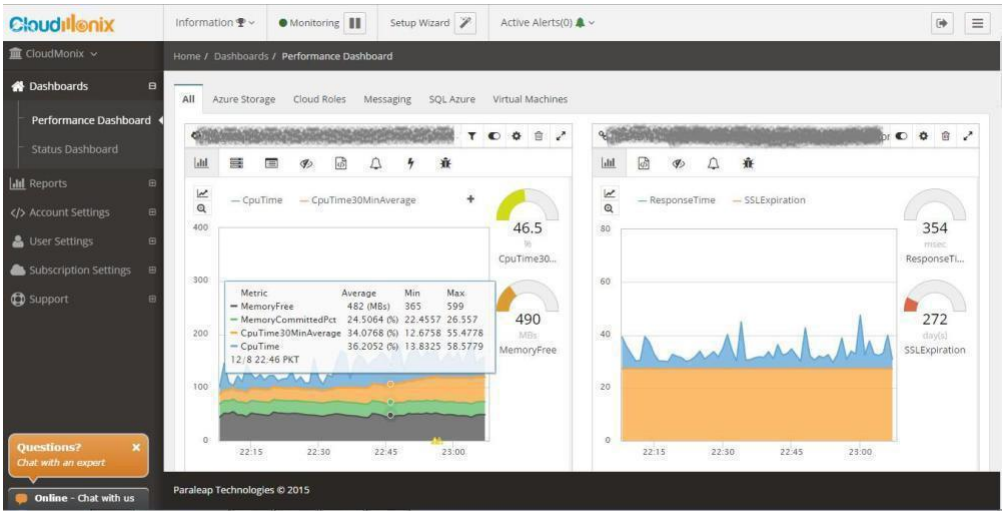
Azure Performance Monitoring Tools

Azure cloud monitoring made easy. This cloud monitoring tool monitors the performance metrics of Cloud Services running on Windows Azure Environment. It monitors the CPU, memory utilization of web and worker roles for any number of instances. Azure Monitoring tools help monitor, optimize and automate day to day Azure Cloud administration and management operations. They provide Azure Cloud administrators and engineers with knowledge and power to optimize their cloud systems for optimal performance in diverse, complex production environments.

Whether you manage a small or large Azure Cloud for your organization, Azure Monitoring Tools can help you improve your cloud operations.

To help Azure Administrators get the best out of their Cloud Infrastructures, we have searched and compiled a list of Azure monitoring tools. These tools will help Azure Administrators in cloud monitoring, performance management, automation, instant alerts, cost management and much more.

CloudMonix (Successor to AzureWatch)

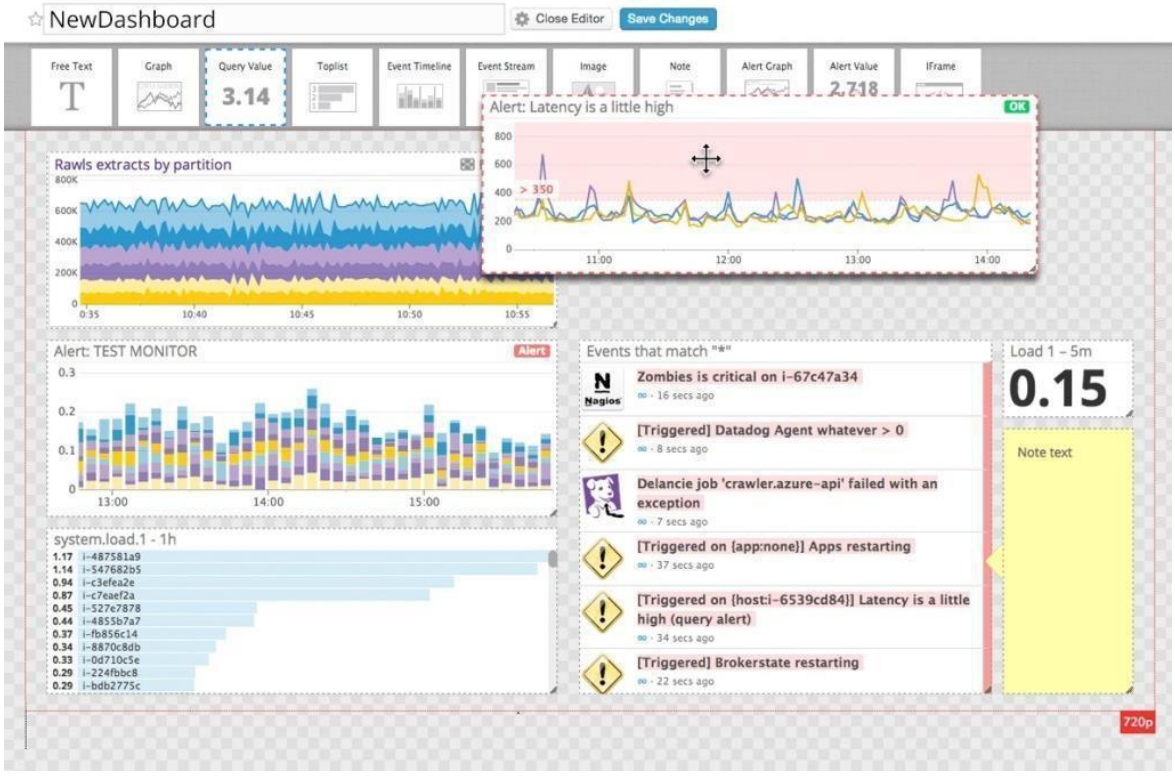


CloudMonix allows for comprehensive Azure monitoring and automation across one or more Azure subscriptions. CloudMonix features in-depth support for large footprint of the Azure platform (VMs, Cloud Roles, SQL Azure, Service Bus, Storage, Virtual Networks, more) , delivers live dashboards, immediate alerts, self-healing automation, auto-scaling and more. CloudMonix is a successor to AzureWatch, which was one of the first comprehensive Azure performance diagnostic, monitoring and auto-scaling solutions.

New Relic

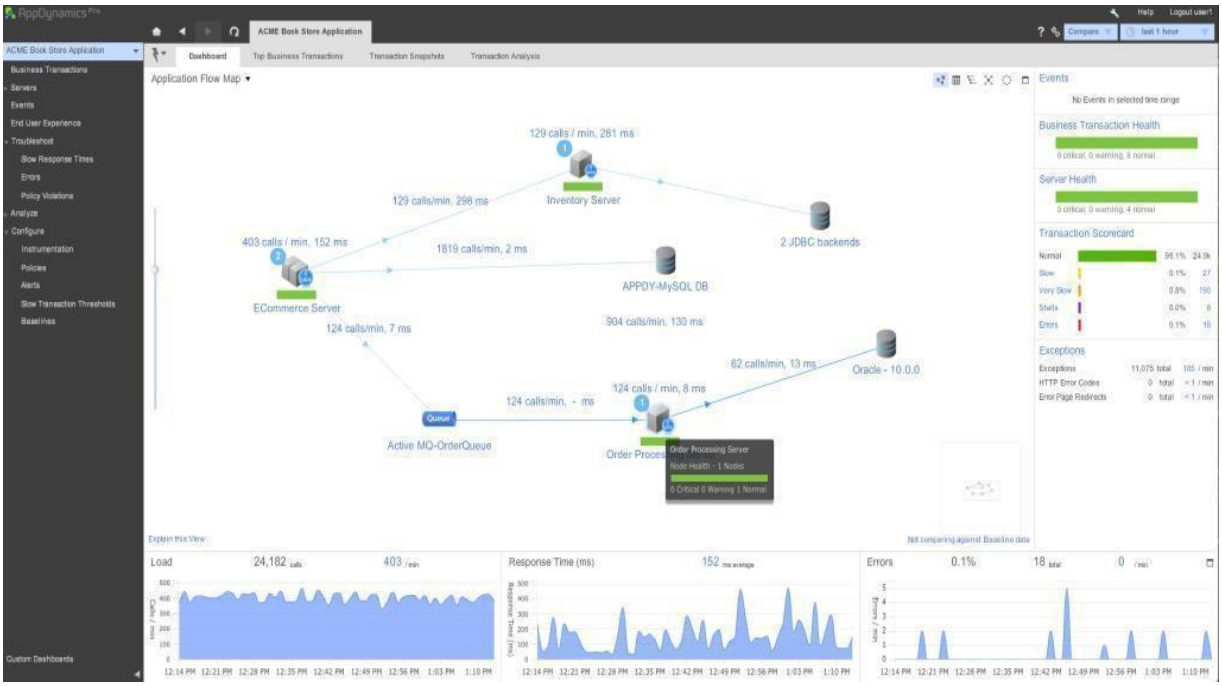
New Relic provides instant visibility into web and mobile apps hosted on Azure. It enables in monitoring and diagnosing problems in Azure-hosted apps from a single dashboard. Administrators can see and optimize resources that are slowing down app performance.

DataDog



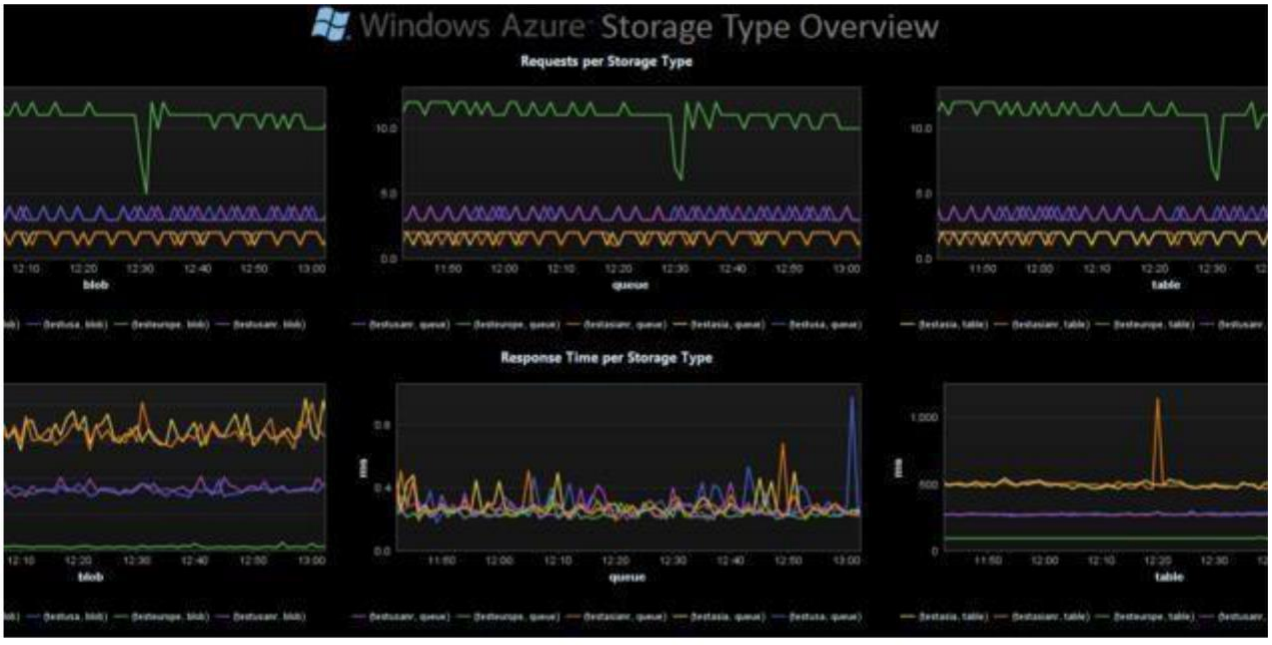
Datadog is a monitoring and analytics platform for Azure based IT infrastructure, operations and development team's .DataDog collects data from servers, databases, applications, tools and services to present a unified view of the applications and services, executed on Azure.

AppDynamics



AppDynamics seamlessly integrates its application performance management solutions with Windows Azure to provide the best possible solution for Azure users. Windows Azure users can isolate problems with code-level diagnostics and get full visibility into every business transaction.

Dynatrace

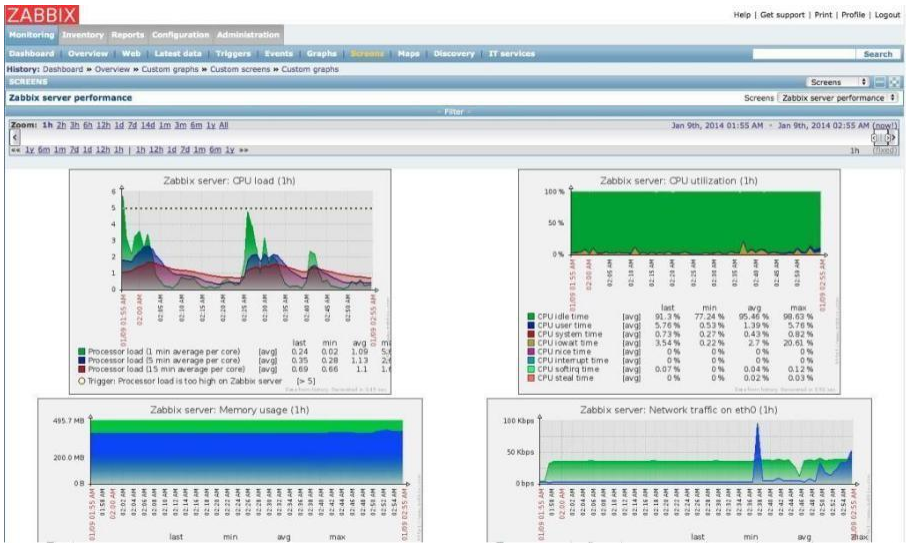


Dynatrace enables you to analyze all transactions down to the method level. Dynatrace User Experience Management (UEM) deploys in Azure to provide visibility and analysis of end-user experience from the browser all the way through your Azure Web and Worker Roles and to the storage layer.

CA Unified Infrastructure Management for Azure

CA Unified Infrastructure Management (CA UIM) for Windows Azure is designed to boost Windows Azure implementations and reliability through improved visibility. Its Azure monitoring capabilities delivers detailed quality-of-service (QoS) metrics, comprehensive usage metering and sophisticated alarms and automation.

Zabbix



Zabbix is an open source enterprise-level software designed for real-time monitoring of metrics collected from servers, virtual machines and network devices, hosted / sourced from Azure.

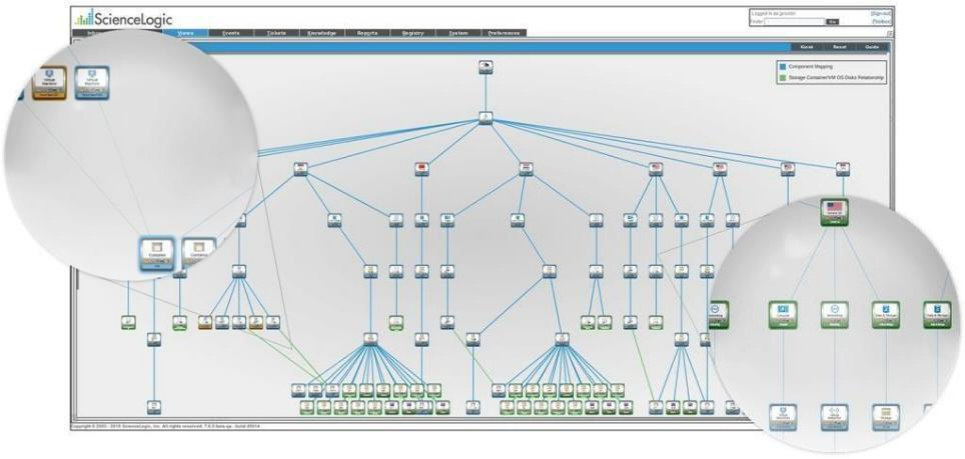
Zenoss

ZenPack by Zenoss provides support for monitoring Microsoft Azure Services. With Zenoss, Azure user can discover Azure Subscription entities, monitor storage service & site components, event management and monitoring for certain Azure states.

Windows Azure App by iKosmik

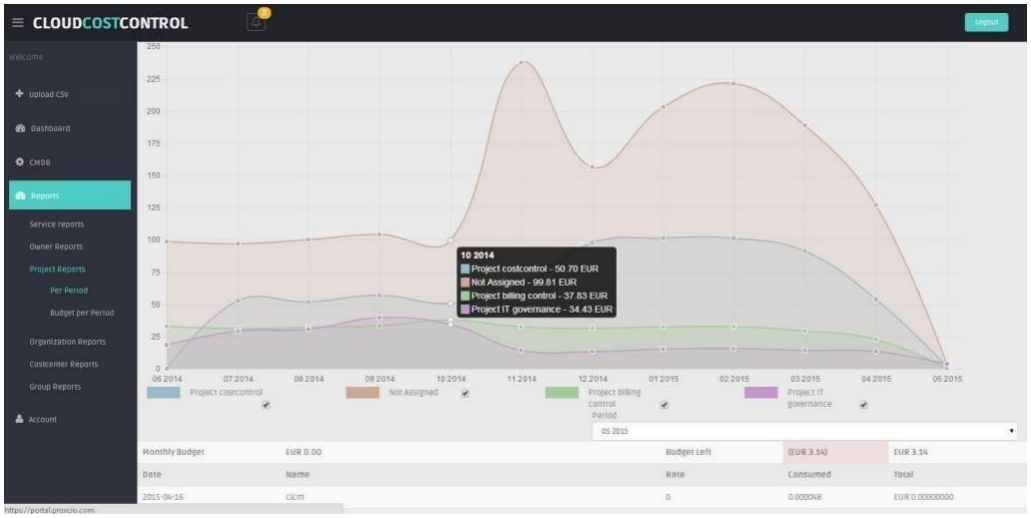
Azure Monitoring app by iKosmik helps you monitor your Microsoft Windows Azure Storage Accounts (Tables, Queues, Blobs) and Websites right from your mobile. The App can monitor any number of Windows Azure Storage Accounts (i.e. tables, blobs and queues) and Websites at a time.

Science Logic



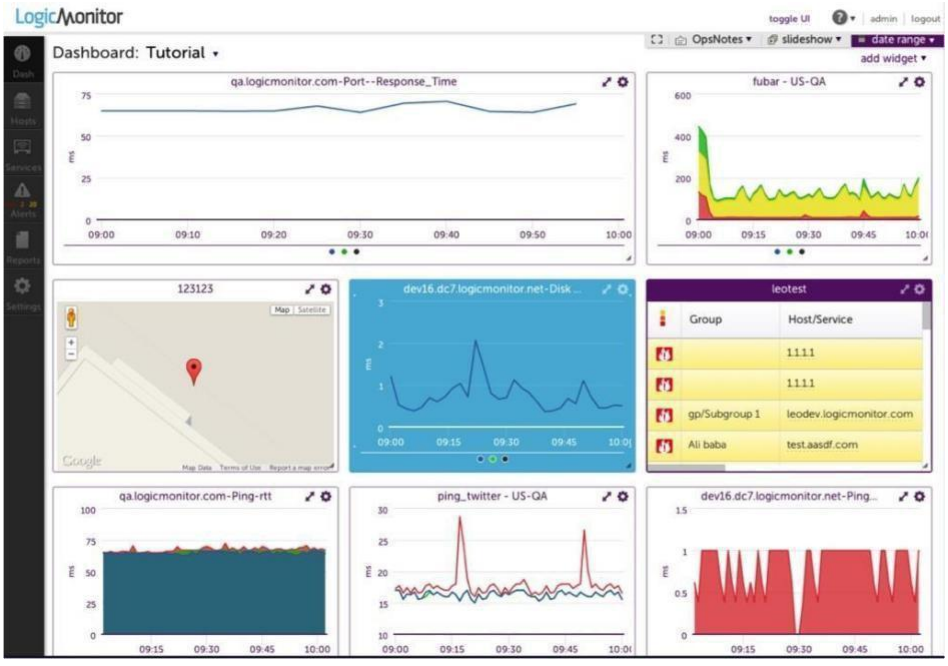
Science Logic's Azure Monitoring tools provide a clear view of your Microsoft Azure-based workload. The solution helps you troubleshoot issues, monitor availability, map dependencies between onsite vs Azure Cloud environments and keep tabs on workload migrations, and manage everything in between.

Proxico



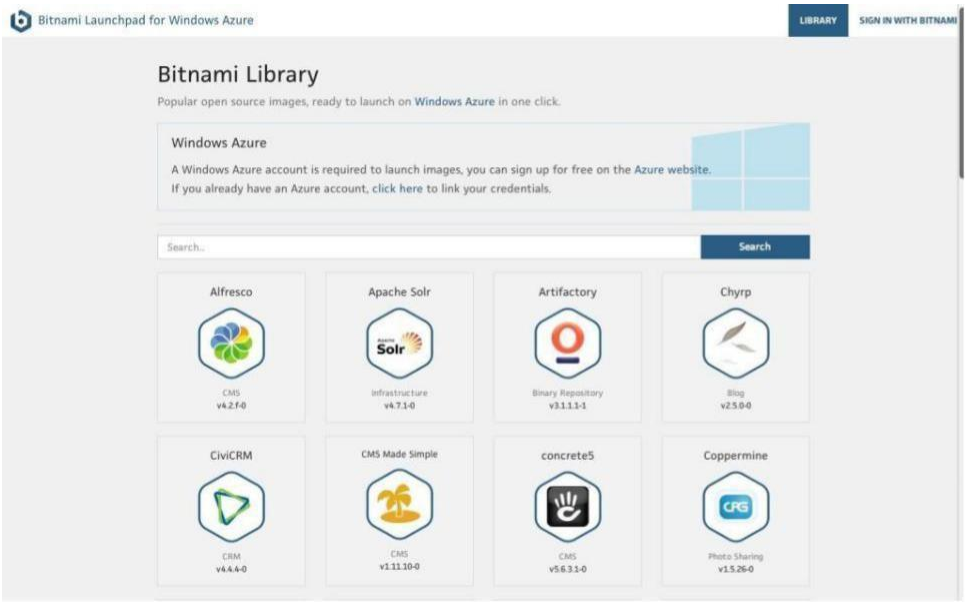
ProXcio's C3 analyze your Azure spending, optimize your costs and manage your corporate IT finance. You can manage your Azure cloud spending across services, departments, projects, cost centers to get deep insight into your usage and costs.

Logic Monitor



LogicMonitor includes support for monitoring Azure metrics. It collects data from virtual machine or servers instances and you'll be able to monitor critical Azure hosted infrastructure performance metrics and build out dashboards that show the data critical to your IT Operations.

Bitnami



The Bitnami Cloud Tools is designed to enable using and monitoring Microsoft Azure services from the command line. The Bitnami Cloud Tools installer is a self-contained, easy to use distribution that packages the Microsoft Azure

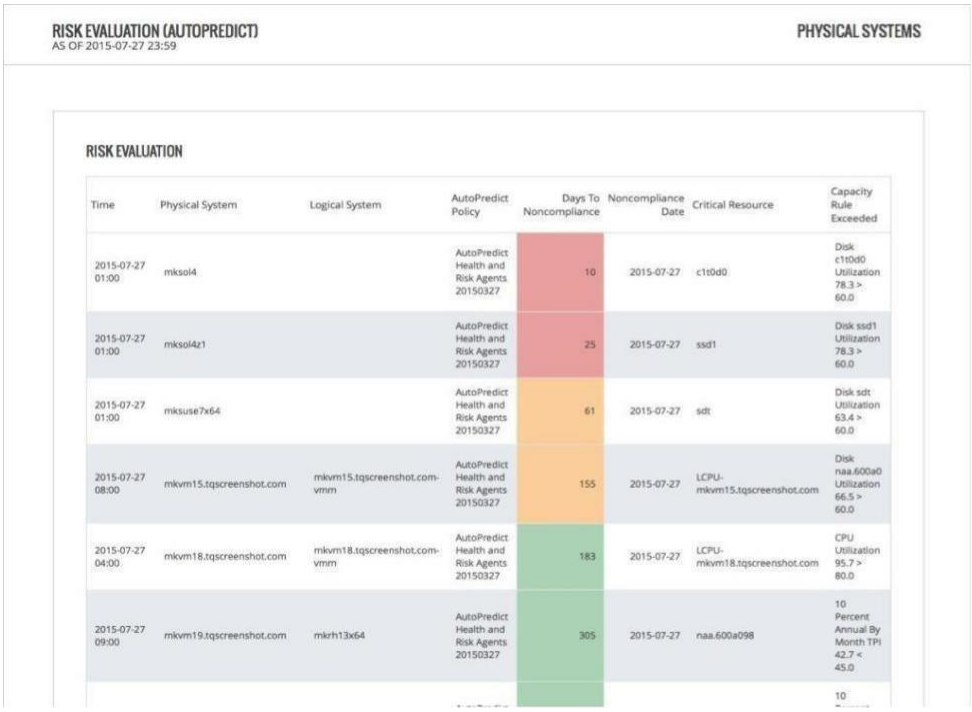
command line tools together with preconfigured Node.js and Python language runtimes.

Cloud Harmony

Cloud Harmony's, CloudSquare service provides a general overview and availability of the Microsoft Azure Cloud services across the different global regions. It provides availability of Azure Virtual Machines, Azure CDN, Azure Compute and Storage resources.

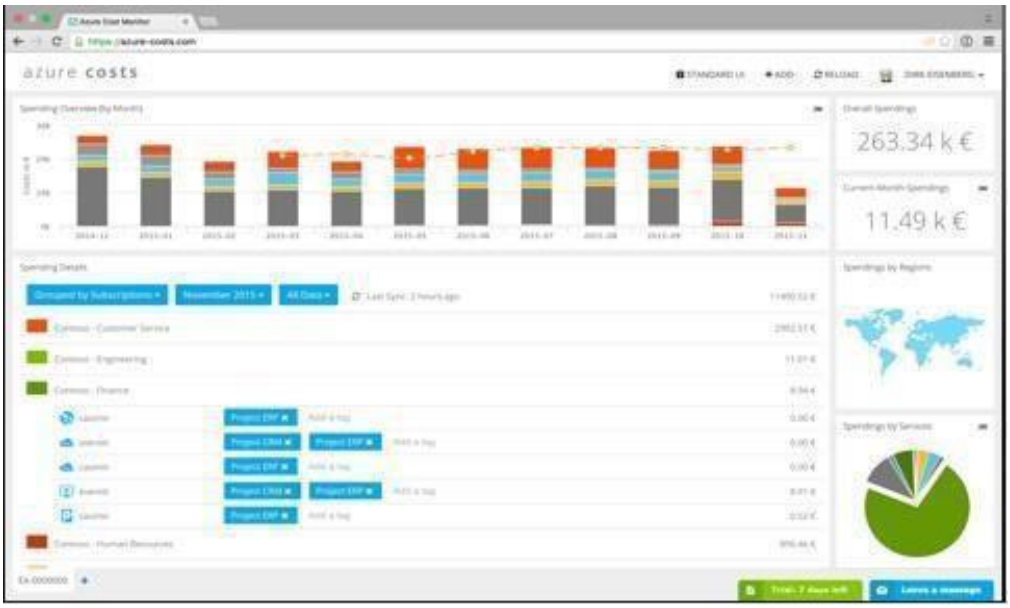
TeamQuest

M

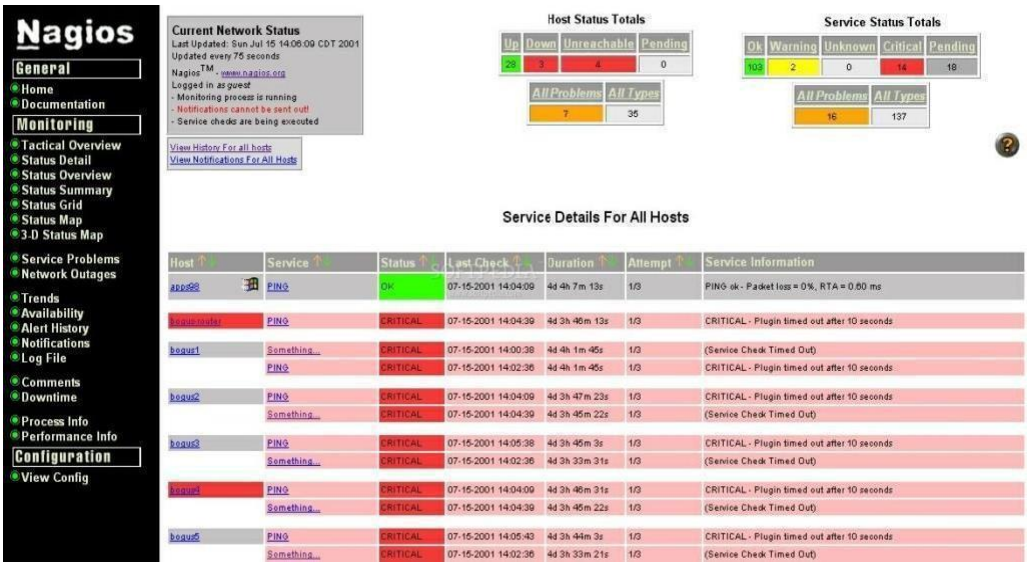


TeamQuest provides capacity management and application health capabilities for the Microsoft Azure cloud computing platform. IT professionals can better monitor and analyze the performance of virtual services running on Azure together with services running on in-house IT infrastructure, increasing environment visibility when using hybrid cloud environments.

Azure Costs



Azure Cost is a simple tool to manage Azure based cost. It enables in identifying costs, tracking spending, optimizing costs, resource tagging and cost reporting across the Azure infrastructure.



Windows Azure Monitor by Nagios provides a set of different utilities to monitor several aspects of Windows Azure. With the plugins / utility, Azure Administrators can monitor Azure AD, Compute Roles, PaaS Roles, SQL Azure, and Azure Storage.

Windows Azure Storage Redundancy Options

Windows Azure Storage provides following options for redundancy for Blobs, Tables and Queues:

1. **Locally Redundant Storage (LRS):** All data in the storage account are made durable by replicating transactions synchronously to three different storage nodes within the same region. The below section will cover more details on LRS including on how to select LRS.

2. **Geo Redundant Storage (GRS):** This is the default option for redundancy when a storage account is created. Like LRS, transactions are replicated synchronously to three storage nodes within the primary region chosen for creating the storage account. However, the transaction is also queued for asynchronous replication to another secondary region (hundreds of miles away from the primary) where data is again made durable by replicating it to three more storage nodes there. The below section will cover in depth the asynchronous replication process, information on region pairings and the failover process.

3. **Read Access - Geo Redundant Storage (RA-GRS):** For a GRS storage account, Azure has introduced in limited preview the ability to turn on read only access to a storage account's data in the secondary region. Since replication to the secondary region is done asynchronously, this provides an eventual consistent version of the data to read from.

About Technosoft Solutions Inc.

Technosoft Solutions Inc. is leading outsourcing software and application development company dedicated in Healthcare Software Development and Integration Services. We are offering a full range of IT solution and support services to companies developing healthcare solution for the U.S market. For almost 14 years, our commitment to providing reliable healthcare system software solutions has remained constant and stable.

Many of our clients are well versed in software development and have healthcare solutions deployed at main hospitals. We augment their resources with our HL7 integration, HIPAA privacy & security services.

We are endowed with quality certifications and standards, including CMMI ML2, ISO 9001:2008, Sun, MySQL and as IBM Certified Partner.