

Cloud Platforms in Industry

Overview

9.1 Amazon web services

- 9.1.1 Compute services

- 9.1.2 Storage services

- 9.1.3 Communication services

- 9.1.4 Additional services

9.2 Google AppEngine

- 9.2.1 Architecture and core concepts

- 9.2.2 Application life cycle

- 9.2.3 Cost model

- 9.2.4 Observations

9.3 Microsoft Azure

- 9.3.1 Azure core concepts
- 9.3.2 SQL Azure
- 9.3.3 Windows Azure platform appliance
- 9.3.4 Observations

9.1 Amazon web services

- Development of flexible applications by providing solutions for elastic infrastructure scalability, messaging, and data storage.
- Accessible through SOAP (Simple Object Access Protocol) or RESTful (Representational State Transfer) Web service interfaces
- Expenses computed on a pay-as-you-go basis.

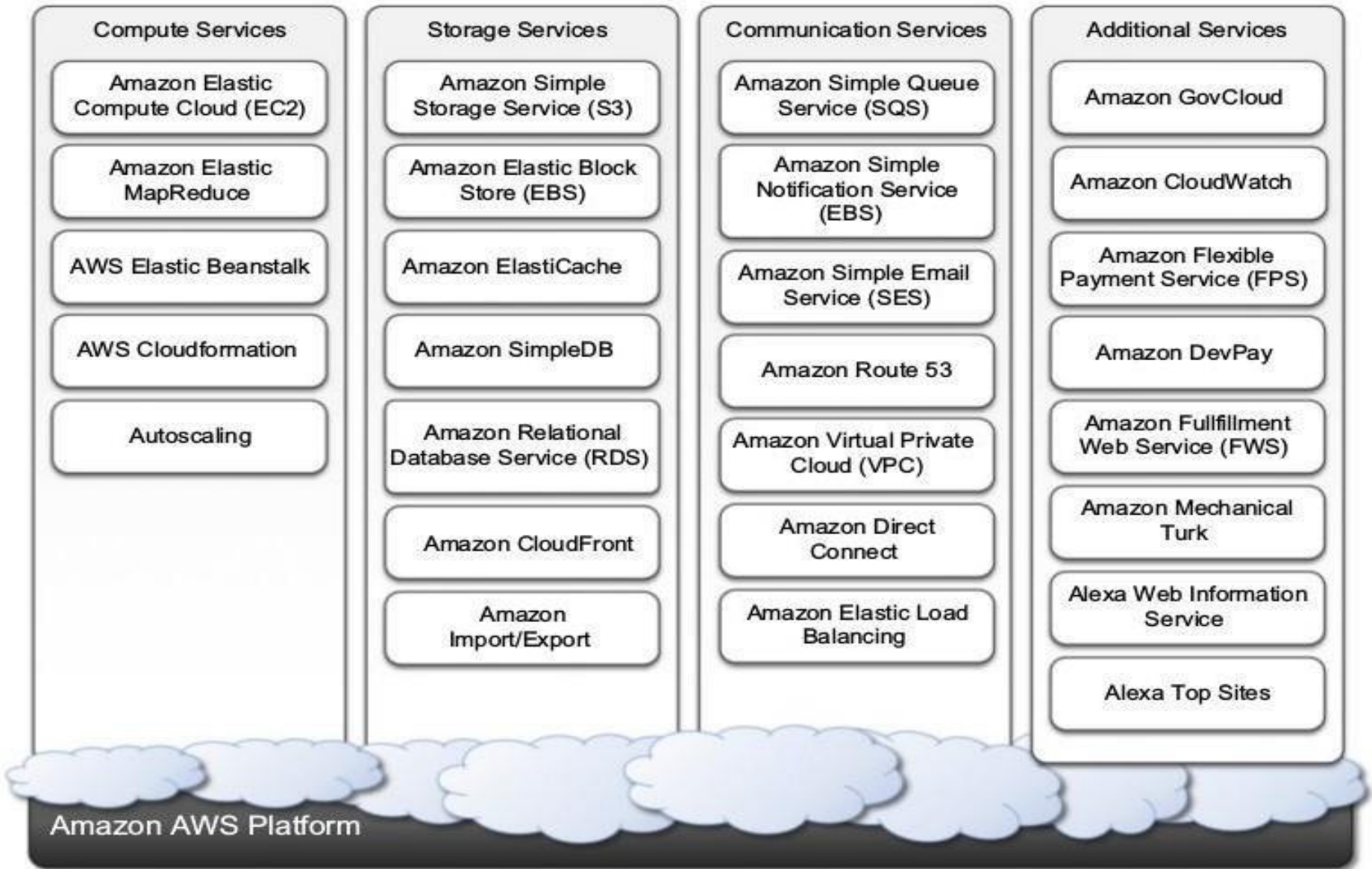


FIGURE 9.1

Amazon Web Services ecosystem.

9.1 Amazon web services

9.1.1 Compute services

- fundamental element of cloud computing systems.
- Amazon EC2, which delivers an IaaS solution.
- Amazon EC2 allows deploying servers in the form of virtual machine images.
- Images come with a preinstalled operating system and a software stack.

1 Amazon machine images

- Amazon Machine Images (AMIs) are templates from which it is possible to create a virtual machine.
- They are stored in Amazon S3.
- An AMI contains a physical file system layout with a predefined operating system installed.
- Predefined operating system installed as Amazon Ramdisk Image (ARI, id: ari-yyyyyy) and Amazon Kernel Image (AKI, id: aki-zzzzzz)

2 EC2 instances

EC2 instances represent virtual machines.

The processing power is expressed in terms of virtual cores and EC2 Compute Units (ECUs).

six major categories:

1. Standard instances.
2. Micro instances.
3. High-memory instances.
4. High-CPU instances.
5. Cluster Compute instances.
6. Cluster GPU instances.

3 EC2 environment

- virtual environment, which provides them with the services they require to host applications.
- In charge of allocating addresses, attaching storage volumes, and configuring security in terms of access control and network connectivity.
- Instances are created with an internal IP address, which makes them capable of communicating within the EC2 network and accessing the Internet as clients.
- Instance owners can partially control where to deploy instances. Instead, they have a finer control over the security of the instances as well as their network accessibility.
- EC2 instances are given domain name in the form `ec2-xxx-xxx-xxx.compute-x.amazonaws.com`

4. Advanced compute services

- **AWS CloudFormation** - model that characterizes EC2 instances.
- Concepts of templates, which are JSON formatted text files.
- Templates provide a simple and declarative way to build complex systems.
- **AWS Elastic Beanstalk** constitutes a simple and easy way to package applications and deploy them on the AWS Cloud.
- Service is available for Web applications developed with the Java/Tomcat technology stack.
- **Amazon Elastic MapReduce** provides AWS users with a cloud computing platform for MapReduce applications. It utilizes Hadoop as the MapReduce engine.

9.1 Amazon web services

9.1.2 Storage services

The core service is represented by Amazon Simple Storage Service (S3).
The core components of S3 are two: buckets and objects.

- 1 S3 key concepts
- 2 Amazon elastic block store
- 3 Amazon ElastiCache
- 4 Structured storage solutions
- 5 Amazon CloudFront

9.1 Amazon web services

9.1.2 Storage services

1 S3 key concepts

accessible through a Representational State Transfer (REST) interface.

- The storage is organized in a two-level hierarchy.
- Stored objects cannot be manipulated like standard files.
- Content is not immediately available to users.
- Requests will occasionally fail.

HTTP requests (GET, PUT, DELETE, HEAD, and POST).

Resource naming - uniform resource identifiers (URIs)

Amazon offers three different ways of addressing a bucket:

1. Canonical form: http://s3.amazonaws.com/bucket_name/
2. Subdomain form: <http://bucketname.s3.amazonaws.com/>
3. Virtual hosting form: <http://bucket-name.com/>

Buckets - container of objects. virtual drive hosted on the S3 distributed storage

Objects and metadata - object is identified by a name that needs to be unique within the bucket in which the content is stored.

Access control and security - access to buckets and objects by means of

9.1 Amazon web services

9.1.2 Storage services

Access Control Policies (ACPs). Different permissions can be used.

9.1 Amazon web services

9.1.2 Storage services

2 Amazon elastic block store

Amazon Elastic Block Store (EBS) provide EC2 instances with persistent storage in the form of volumes that can be mounted at instance startup. EBS volumes normally reside within the same availability zone of the EC2 instances that will use them to maximize the I/O performance.

3 Amazon ElastiCache

implementation of an elastic in-memory cache based on a cluster of EC2 instances.

based on a cluster of EC2 instances running the caching software, which is made available through Web services.

ElastiCache cluster can be dynamically resized according to the demand of the client applications.

9.1 Amazon web services

9.1.2 Storage services

4 Structured storage solutions

Amazon provides applications with structured storage services in three different forms:

- Preconfigured EC2 AMIs,
- Amazon Relational Data Storage (RDS), and
- Amazon SimpleDB.

Preconfigured EC2 AMIs are predefined templates featuring an installation of a given database management system. EC2 instances created from these AMIs can be completed with an EBS volume for storage persistence.

IBM DB2, Microsoft SQL Server, MySQL, Oracle, PostgreSQL, Sybase, and Vertica.

RDS is relational database service that relies on the EC2 infrastructure and is managed by Amazon.

Developers do not have to worry about configuring the storage for high availability, designing failover strategies, or keeping the servers up-to-date.

Amazon SimpleDB is a lightweight, highly scalable, and flexible data storage solution for applications that do not require a fully relational model for their data.

9.1 Amazon web services

9.1.2 Storage services

5 Amazon CloudFront

- Content delivery network.
- Collection of edge servers strategically located around the globe to better serve requests for static and streaming Web content.
- Content that can be delivered through CloudFront is static (HTTP and HTTPS) or streaming (Real Time Messaging Protocol, or RMTP).

9.1 Amazon web services

9.1.3 Communication services

Facilities to structure and facilitate the communication among existing applications and services residing within the AWS infrastructure.

These facilities can be organized into two major categories:

- 1. Virtual networking** and
- 2. Messaging.**

9.1 Amazon web services

9.1.3 Communication services

1. Virtual networking

- Virtual networking comprises a collection of services that allow AWS users to control the connectivity to and between compute and storage services.
- Amazon Virtual Private Cloud (VPC) and Amazon Direct Connect provide connectivity solutions in terms of infrastructure.
- Flexibility in creating virtual private networks within the Amazon infrastructure and beyond.

2. Messaging.

- Amazon Simple Queue Service (SQS),
- Amazon Simple Notification Service (SNS), and
- Amazon Simple Email Service (SES).

9.2 Google AppEngine

- Google AppEngine is a PaaS implementation.
- Distributed and scalable runtime environment that leverages Google's distributed infrastructure to scale out applications.

9.2.1 Architecture and core concepts

9.2.2 Application life cycle

9.2.3 Cost model

9.2.4 Observations

9.2 Google AppEngine

9.2 Google AppEngine

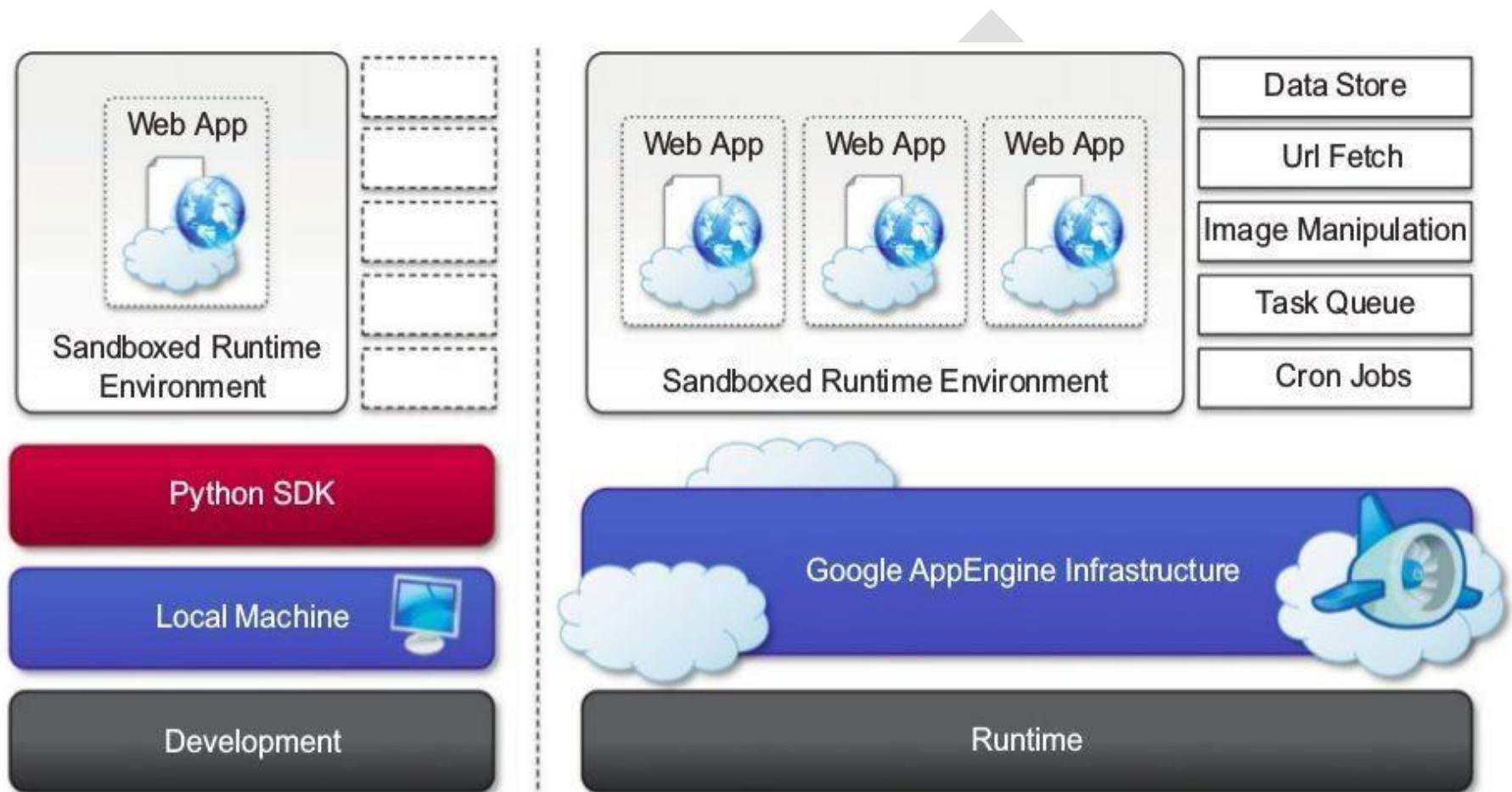


FIGURE 9.2

Google AppEngine platform architecture.

9.2 Google AppEngine

9.2.1 Architecture and core concepts

The platform is logically divided into four major components:

- 1 Infrastructure
- 2 Runtime environment
- 3 Storage
- 4 Application services
- 5 Compute services

9.2 Google AppEngine

9.2.1 Architecture and core concepts

1 Infrastructure

- AppEngine's infrastructure takes advantage of many servers available within Google datacenters. For each HTTP request, AppEngine locates the servers hosting the application that processes the request, evaluates their load, and, if necessary, allocates additional resources (i.e., servers) or redirects the request to an existing server.
-
- Also responsible for monitoring application performance and collecting statistics on which the billing is calculated.

9.2 Google AppEngine

9.2.1 Architecture and core concepts

2 Runtime environment

represents the execution context of applications hosted on AppEngine.

Sandboxing

Major responsibilities of the runtime environment is to provide the application environment with an isolated and protected context.

Supported runtimes

- Java, Python, and Go.
- Currently supports Java 6, Java Server Pages (JSP), and applications interact with environment by using the Java Servlet standard.
- Developers can create applications with the AppEngine Java SDK.
- Python is provided by an optimized Python 2.5.2 interpreter.
- Python Web application framework- webapp.
- Go programming language - SDK includes compiler and standard libraries for developing applications in Go and interfacing it AppEngine services.

9.2 Google AppEngine

9.2.1 Architecture and core concepts

3 Storage

operate differently depending on the volatility of the data.

Static file servers

- Dynamic and static data.
- Dynamic data - logic of the application and the interaction with the user.
- Static data constitute of components that define the graphical layout of the application (CSS files, plain HTML files, JavaScript files, images, icons, and sound files) or data files.

DataStore

- DataStore is a service that allows developers to store semistructured data.
- DataStore is based on Bigtable a redundant, distributed, and semistructured
- data store that organizes data in the form of tables.

9.2 Google AppEngine

9.2.1 Architecture and core concepts

4 Application services

Simplify common operations that are performed in Web applications:

- access to data,
- account management,
- integration of external resources,
- messaging and communication,
- image manipulation, and
- asynchronous computation.

UrlFetch

- The sandbox environment does not allow applications to open arbitrary connections.
- HTTP/HTTPS by means of the UrlFetch service.
- Applications can make synchronous and asynchronous Web requests and integrate the resources obtained.
- Ability to set deadlines for requests so that they can be completed (or aborted) within a given time.

9.2 Google AppEngine

9.2.1 Architecture and core concepts

4 Application services

MemCache

- caching services by means of MemCache.
- distributed in-memory cache that is optimized for fast access and provides volatile store for objects that are frequently accessed.
- significantly reduce the access time to data.

Mail and instant messaging

- Email can also be used to trigger activities in Web applications.
- send and receive mails through Mail service - allows sending email on behalf of the application to specific user accounts.
- Extensible Messaging and Presence Protocol (XMPP), Any chat service that supports XMPP, such as Google Talk, can send and receive chat messages to and from the Web application.

Account management

- Web applications can conveniently store profile settings in the form of key-value pairs, attach them to a given Google account, and quickly retrieve them once the user authenticates.

9.2 Google AppEngine

9.2.1 Architecture and core concepts

5 Compute services

- Users navigate the Web pages and get instantaneous feedback.
- Feedback is result of some computation happening on the Web application.
- Immediate feedback and notification once the required operation is completed.
- **Task queues**
- Task Queues allow applications to submit a task for a later execution.
- **Cron jobs**
- Schedule the required operation at the desired time.

9.2 Google AppEngine

9.2.2 Application life cycle

Testing and development, deployment, and monitoring.

Java SDK and Python SDK

1 Application development and testing

2 Application deployment and management

1 Application development and testing

- **Java SDK** - facility for building applications with the Java 5 and Java 6 runtime environments.
- Eclipse development environment by using the Google AppEngine plug-in - install Java SDK, Google Web Toolkit, and Google AppEngine plug-ins into Eclipse.
- **Python SDK** - Python 2.5.
- Standalone tool, called GoogleAppEngineLauncher, for managing Web applications locally and deploying them to AppEngine.
- **webapp** - includes a set of models, components, and tools that simplify development of Web applications.

9.2 Google AppEngine

9.2.2 Application life cycle

2 Application deployment and management

- Deployed on AppEngine with a simple click or command-line tool.
- Also from GoogleAppEngineLauncher and Google AppEngine plug-in.
- <http://<application-id>.appspot.com>.
- Also map the application with registered DNS domain name.

9.2 Google AppEngine

9.2.3 Cost model

- Set up a billing account and be charged on a pay-per-use basis.
- Developers to identify the appropriate daily budget.
- An application is measured against billable quotas, fixed quotas, and per-minute quotas.
- Once an application reaches the quota, resource is depleted and will not be available to application until the quota is replenished.
- Once a resource is depleted, subsequent requests to that resource will generate an error or an exception.

9.3 Microsoft Azure

Built on top of Microsoft datacenters infrastructure and provides with collection of services.

- Services range from compute, storage, and networking to application connectivity, access control, and business intelligence.
- Figure 9.3 provides an overview of services provided by Azure.

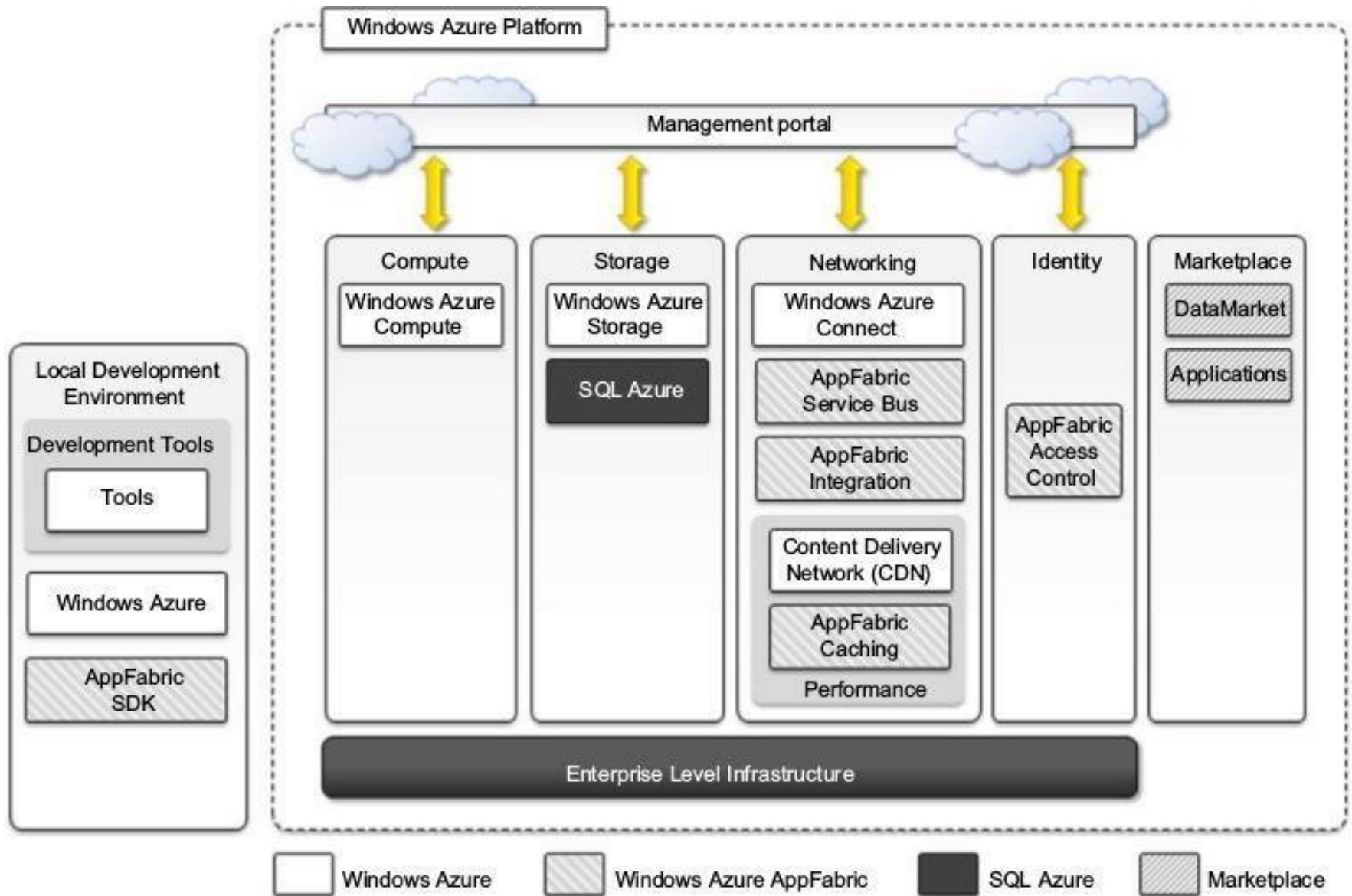


FIGURE 9.3

9.3 Microsoft Azure

9.3.1 Azure core concepts

9.3.2 SQL Azure

9.3.3 Windows Azure platform appliance

9.3.4 Observations

9.3.1 Azure core concepts

- Made up of **foundation layer** and a set of developer services that can be used to build scalable applications.
- Services cover **compute**, **storage**, **networking**, and **identity management**, which are tied together by middleware called **AppFabric**.

1 Compute services

Web role

Worker role

Virtual machine role

2 Storage services

Blobs

Azure drive

Tables

Queues

9.3 Microsoft Azure

9.3.1 Azure core concepts

3 Core infrastructure: AppFabric

- Access control

- Service bus

- Azure cache

4 Other services

- Windows Azure virtual network

- Windows Azure content delivery network

9.3 Microsoft Azure

9.3.1 Azure core concepts

1 Compute services

- Compute services are core components, they are delivered by means of abstraction of roles.
- A role is a runtime environment that is customized for specific compute task.
- Roles are managed by Azure OS and instantiated on demand.

Web role

- The Web role is designed to implement scalable Web applications.
- hosted on the IIS 7 Web Server, which is a component of the infrastructure that supports Azure.
- .NET technology supports Web roles; developers can directly develop their applications in Visual Studio, test them locally, and upload to Azure.
- Used to run and scale PHP Web applications on Azure.

Worker role

- Worker roles are designed to host general compute services on Azure.
- Quickly provide compute power or to host services.
- Runs continuously from the creation of its instance.
- Azure SDK provides APIs and libraries that allow connecting the role with the service provided by runtime and easily controlling its startup.

9.3 Microsoft Azure

9.3.1 Azure core concepts

1 Compute services

Virtual machine role

- allows to fully control the computing stack of compute service by defining a custom image of Windows Server 2008 R2 OS & their applications.
- Hyper-V virtualization technology.
- Developers can image Windows server installation into a **Virtual Hard Disk (VHD)**.

Table 9.7 Windows Azure Compute Instances Characteristics, 2011–2012

Compute Instance Type	CPU	Memory	Instance Storage	I/O Performance	Hourly Cost (USD)
Extra Small	1.0 GHz	768 MB	20 GB	Low	\$0.04
Small	1.6 GHz	1.75 GB	225 GB	Moderate	\$0.12
Medium	2 × 1.6 GHz	3.5 GB	490 GB	High	\$0.24
Large	4 × 1.6 GHz	7 GB	1,000 GB	High	\$0.48
Extra Large	8 × 1.6 GHz	14 GB	2,040 GB	High	\$0.96

9.3 Microsoft Azure

9.3.1 Azure core concepts

2 Storage services

- Windows Azure provides different types of storage solutions that complement compute services with a more durable and redundant option compared to local storage.

Blobs

- Store large amount of data in the form of binary large objects (BLOBs).
- Two types of blobs are available:
 - **Block blobs.** composed of blocks optimized for sequential access;
 - Blocks are of 4 MB, and a single block blob can reach 200 GB.
 - **Page blobs.** pages that are identified by an offset from the beginning of blob.
 - Split into multiple pages or constituted of single page.
 - Optimized for random access.
 - Maximum dimension of a page blob can be 1 TB.

Azure drive

- Entire file system in the form of single Virtual Hard Drive (VHD) file.
- NTFS file system, providing persistent and durable storage.

Tables

- semistructured storage solution. Tables are more similar to spreadsheets.
- Handle large amounts of data and queries returning huge result sets.

9.3 Microsoft Azure

- Currently, table can contain up to 100 TB of data, and rows can have up to 255 properties, with a maximum of 1 MB for each row.

9.3 Microsoft Azure

9.3.1 Azure core concepts

2 Storage services

Queues

- Queue storage allows applications to communicate by exchanging messages through durable queues, thus avoiding lost or unprocessed messages.
 - Applications enter messages into a queue, and other applications can read them in a first-in, first-out (FIFO) style.
 - When an application reads a message it is marked as invisible; hence it will not be available to other clients.
 - Once Application has completed processing message, it explicitly deletes message from queue.
 - Alternative to reading a message is **peeking**, which allows retrieving message but letting it stay visible in the queue.
-
- All the services described are **geo-replicated three times** to ensure their availability in case of major disasters.
 - **Geo-replication** involves the copying of data into a different datacenter that is **hundreds or thousands of miles away** from the original

9.3 Microsoft Azure

datacenter.

9.3 Microsoft Azure

9.3.1 Azure core concepts

3 Core infrastructure: AppFabric

- Middleware for developing, deploying, and managing applications on cloud or for integrating existing applications with cloud services.
- Scaling out and high availability; sandboxing and multitenancy; state management; and dynamic address resolution and routing.
- Simplify many common tasks in distributed application, such as communication, authentication and authorization, and data access.

Access control

- AppFabric provides capability of encoding access control to resources in Web applications.
- Services into set of rules that are expressed outside application code base.
- Applications can leverage Active Directory, Windows Live, Google, Facebook, and other services to authenticate users.

Service bus

- Messaging and connectivity infrastructure.
- Designed to allow transparent network traversal and to simplify development of loosely coupled applications letting developers focus on logic of interaction.
- Applications need to be connected to bus, which provides these services.

9.3 Microsoft Azure

9.3.1 Azure core concepts

3 Core infrastructure: AppFabric

Azure cache

- provides a set of durable storage solutions that allow applications to persist their data.
- Azure Cache is a service that allows developers to quickly access data persisted on Windows Azure storage or in SQL Azure.
- implements a distributed in-memory cache of which, size can be dynamically adjusted by applications.

9.3 Microsoft Azure

9.3.1 Azure core concepts

4 Other services

- Simplify the development and integration of applications.

Windows Azure virtual network

- includes **Windows Azure Connect** and **Windows Azure Traffic Manager**.
- **Windows Azure Connect** allows easy setup of IP-based network connectivity among machines hosted on private premises and roles deployed on Azure Cloud.
- **Windows Azure Traffic Manager** provides load-balancing features for services listening to HTTP or HTTPS ports and hosted on multiple roles.

Windows Azure content delivery network

- Content delivery network solution that improves content delivery capabilities of Windows Azure Storage and Microsoft Windows Update, Bing maps.

9.3 Microsoft Azure

9.3.2 SQL Azure

- SQL Azure is a relational database service hosted on Windows Azure and built on the SQL Server technologies.
- via SQL Server - provides developers with scalable, highly available, and fault-tolerant relational database.
- Service is manageable using REST APIs, allowing developers to control databases.

9

9.1

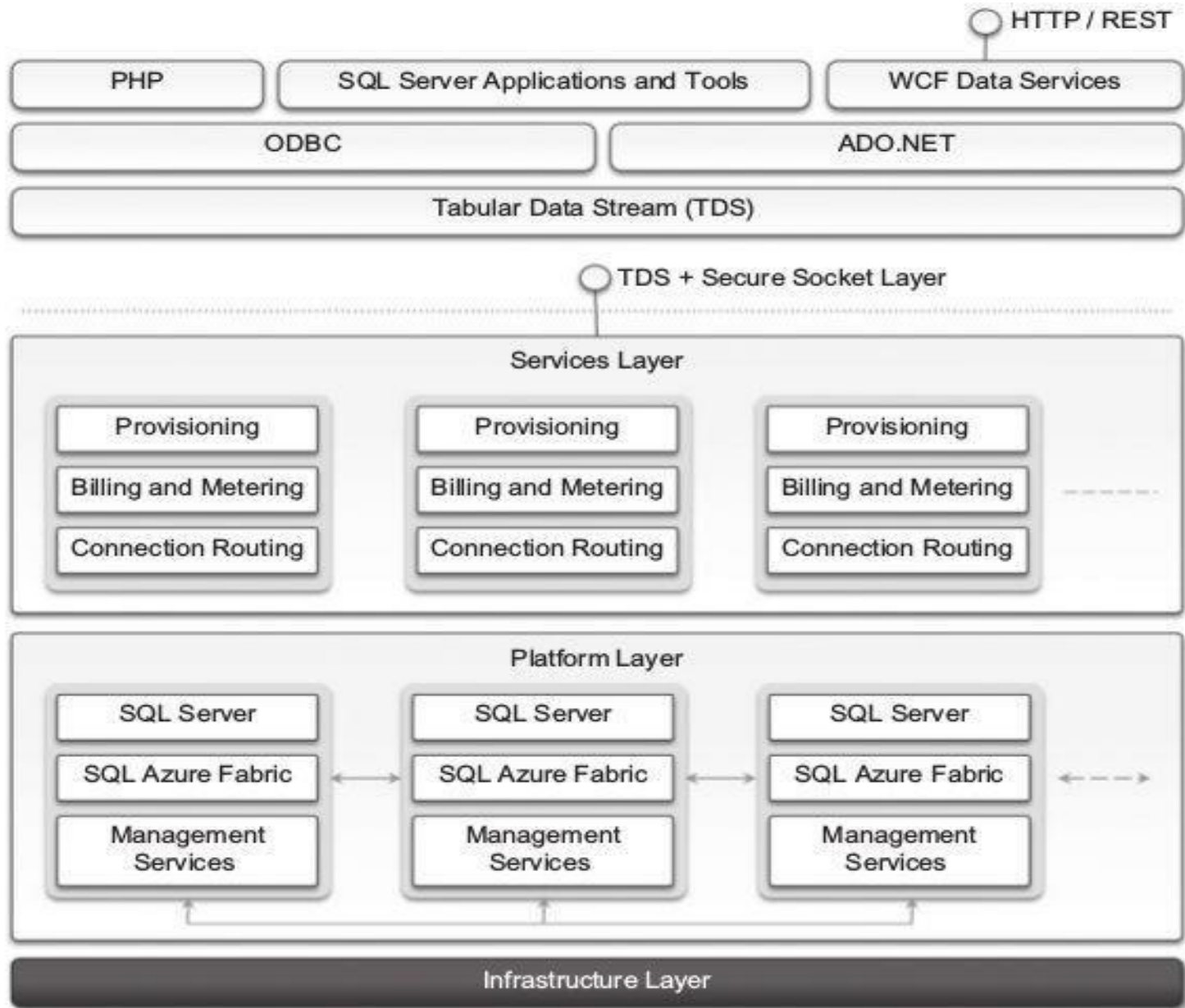


FIGURE 9.4

9.3 Microsoft Azure

9.3.2 SQL Azure

- Developers have to sign up for Windows Azure account in order to use SQL Azure.
- Closely resemble physical SQL Servers: They have a fully qualified domain name under the database.windows.net (i.e., **server-name.database.windows.net**) domain name.
- This simplifies management tasks and the interaction with SQL Azure from client applications.

9.3 Microsoft Azure

9.3.3 Windows Azure platform appliance

- Can also be deployed as appliance on third-party data centers and constitutes the cloud infrastructure governing the physical servers of datacenter.
- Includes Windows Azure, SQL Azure, and Microsoft- specified configuration of network, storage, and server hardware.
- The appliance targets governments and service providers who want to have their own cloud computing infrastructure.
- Full-featured implementation of Windows Azure.
- Goal is to replicate Azure on a third-party infrastructure and make available its services beyond boundaries of Microsoft Cloud.

The appliance addresses two major scenarios:

- institutions that have very large computing needs (such as government agencies) and
- institutions that cannot afford to transfer their data outside their premises.

9.3 Microsoft Azure

9.3.4 Observations

- Windows Azure is Microsoft's solution for developing cloud computing applications.
- PaaS layer.
- Provides developer with collection of services and scalable middleware hosted on Microsoft datacenters that address compute, storage, networking, and identity management needs of applications.
- Individually or all together for building both applications.
- Compute services are based on the abstraction of roles.
- AppFabric, constitutes distributed and scalable middleware of Azure.
- SQL Azure is another important element of Windows Azure and provides support for relational data in the cloud.
- Platform is based on .NET technology and Windows systems.

Chapter 10 – Cloud Applications

- Huge popularity in industry due to its ability to host applications for which the services can be delivered to consumers rapidly at minimal cost.
- We discuss Application case studies, detailing their architecture and how they leveraged various cloud technologies.
- Range of domains, from scientific to engineering, gaming, and social networking.

Overview

10.1 Scientific applications

- 10.1.1 Healthcare: ECG analysis in the cloud
- 10.1.2 Biology: protein structure prediction
- 10.1.3 Biology: gene expression data analysis for cancer diagnosis
- 10.1.4 Geoscience: satellite image processing

10.2 Business and consumer applications

10.2.1 CRM and ERP

- 1 Salesforce.com
- 2 Microsoft dynamics CRM
- 3 NetSuite

10.2.2 Productivity

- 1 Dropbox and iCloud
- 2 Google docs
- 3 Cloud desktops: EyeOS and XIOS/3

10.2.3 Social networking

- 1 Facebook

10.2.4 Media applications

- 1 Animoto
- 2 Maya rendering with Aneka
- 3 Video encoding on the cloud: Encoding.com

10.2.5 Multiplayer online gaming

10.1 Scientific applications

- increasingly using cloud computing systems and technologies.
- high-performance computing (HPC) applications, high-throughput computing (HTC) applications, and data-intensive applications.
- minimal changes need to be made to existing applications.
- IaaS solutions, offer optimal environment for running bag-of-tasks applications and workflows.
- Scientists explore new programming models for tackling computationally challenging problems.
- Applications - redesigned and implemented on top of cloud programming application models and platforms to leverage their unique capabilities.

10.1 Scientific applications

10.1.1 Healthcare: ECG analysis in the cloud

- cloud technologies to support doctors in providing more effective diagnostic processes.
- Internet connectivity and its accessibility from any device at any time has made cloud technologies an attractive option for developing health-monitoring systems.

10.1 Scientific applications

10.1.1 Healthcare: ECG analysis in the cloud

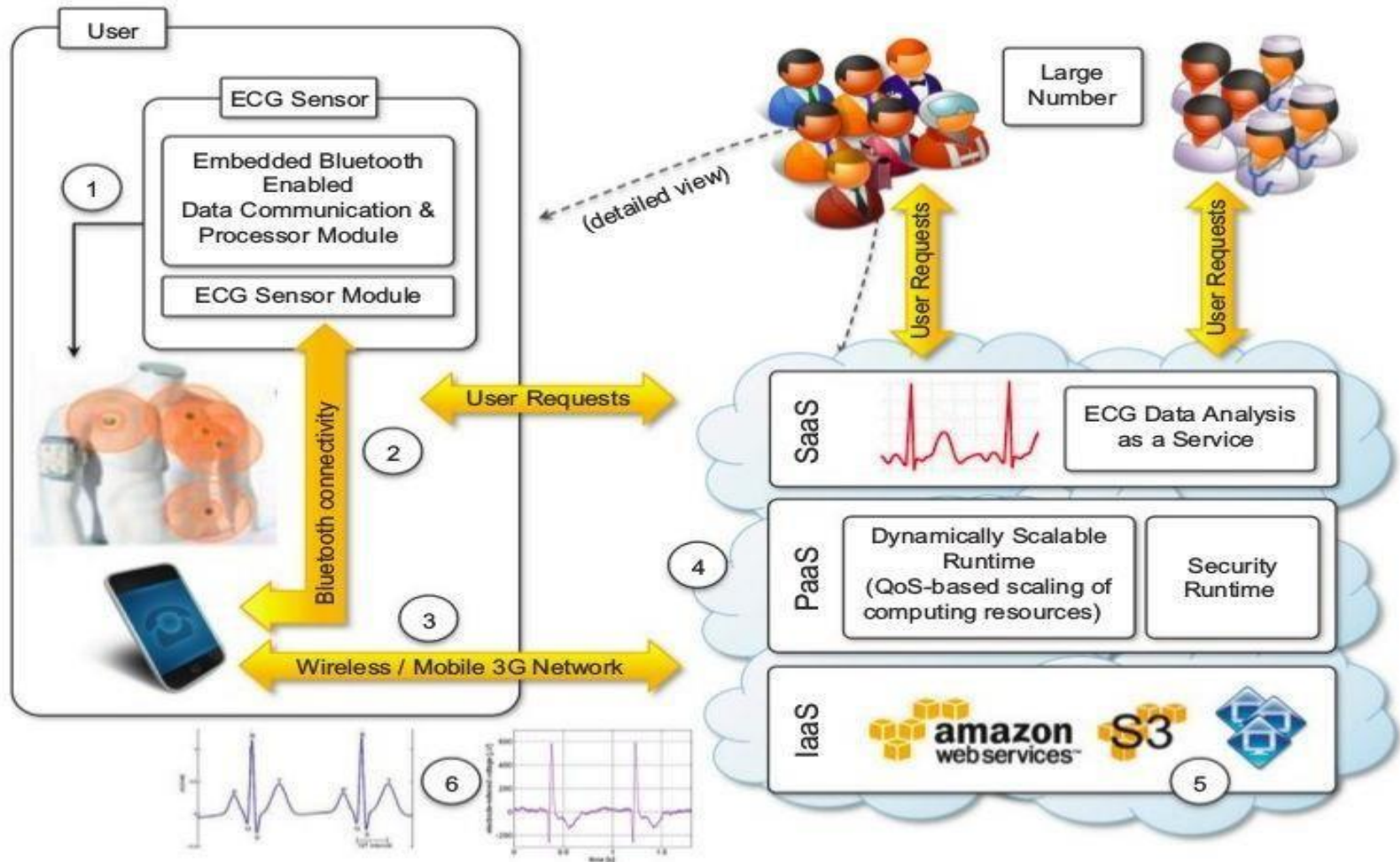


FIGURE 10.1

An online health monitoring system hosted in the cloud.

10.1 Scientific applications

10.1.2 Biology: protein structure prediction

- Require high computing capabilities and often operate on large data-sets that cause extensive I/O operations.
- extensive use of supercomputing and cluster computing infrastructures.
- Protein structure prediction is a computationally intensive task.
- The geometric structure of a protein cannot be directly inferred from the sequence of genes that compose its structure, but it is the result of complex computations aimed at identifying the structure.
- This task requires investigation of space with massive number of states, creating large number of computations for each of these states.

- One project that investigates the use of cloud technologies for protein structure prediction is Jeeva — an integrated Web portal that enables scientists to offload the prediction task to a computing cloud based on Aneka (**Figure 10.2**).
- Uses machine learning techniques.
- E,H,C classification.