Chapter 9

Cloud Management Mechanisms

9.1 Remote Administration System

9.2 Resource Management System

9.3 SLA Management System

9.4 Billing Management System

Cloud-based IT resources need to be set up, configured, maintained, and monitored. The systems covered in this chapter are mechanisms that encompass and enable these types of management tasks. They form key parts of cloud technology architectures by facilitating the control and evolution of the IT resources that form cloud platforms and solutions.

The following management-related mechanisms are described in this chapter:

- Remote Administration System
- Resource Management System
- SLA Management System
- Billing Management System

These systems typically provide integrated APIs and can be offered as individual products, custom applications, or combined into various product suites or multi-function applications.

9.1 Remote Administration System

The *remote administration system* mechanism (Figure 9.1) provides tools and user-interfaces for external cloud resource administrators to configure and administer cloud-based IT resources.

A remote administration system can establish a portal for access to administration and management features of various underlying systems, including the resource management, SLA management, and billing management systems described in this chapter (Figure 9.2).

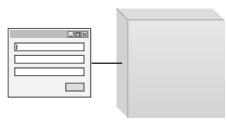


Figure 9.1
The symbol used in this book for the remote administration system. The displayed user-interface will typically be labeled to indicate a specific type of portal.

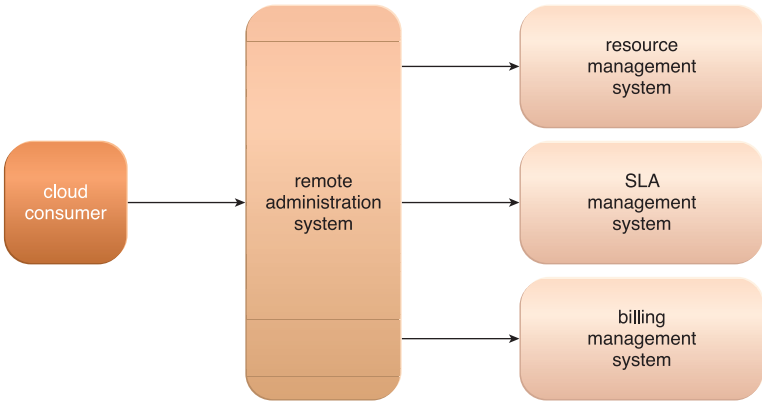


Figure 9.2
The remote administration system abstracts underlying management systems to expose and centralize administration controls to external cloud resource administrators. The system provides a customizable user console, while programmatically interfacing with underlying management systems via their APIs.

The tools and APIs provided by a remote administration system are generally used by the cloud provider to develop and customize online portals that provide cloud consumers with a variety of administrative controls.

The following are the two primary types of portals that are created with the remote administration system:


usage and administration portal

- *Usage and Administration Portal* – A general purpose portal that centralizes management controls to different cloud-based IT resources and can further provide IT resource usage reports. This portal is part of numerous cloud technology architectures covered in Chapters 11 to 13.


self-service portal

- *Self-Service Portal* – This is essentially a shopping portal that allows cloud consumers to search an up-to-date list of cloud services and IT resources that are available from a cloud provider (usually for lease). The cloud consumer submits its chosen items to the cloud provider for provisioning. This portal is primarily associated with the rapid provisioning architecture described in Chapter 12.

Figure 9.3 illustrates a scenario involving a remote administration system and both usage and administration and self-service portals.

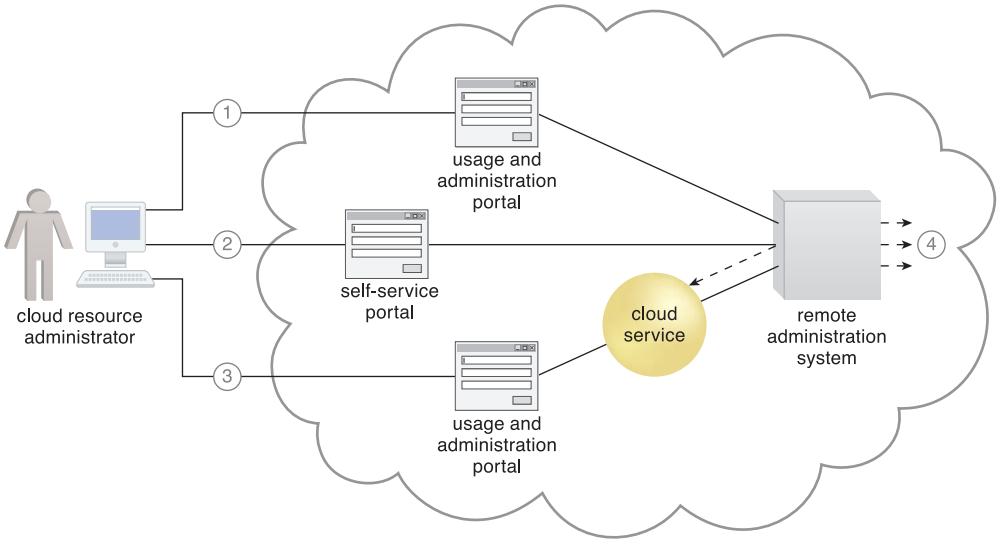


Figure 9.3

A cloud resource administrator uses the usage and administration portal to configure an already leased virtual server (not shown) to prepare it for hosting (1). The cloud resource administrator then uses the self-service portal to select and request the provisioning of a new cloud service (2). The cloud resource administrator then accesses the usage and administration portal again to configure the newly provisioned cloud service that is hosted on the virtual server (3). Throughout these steps, the remote administration system interacts with the necessary management systems to perform the requested actions (4).

Depending on:

- the type of cloud product or cloud delivery model the cloud consumer is leasing or using from the cloud provider,
- the level of access control granted by the cloud provider to the cloud consumer, and
- further depending on which underlying management systems the remote administration system interfaces with,

…tasks that can commonly be performed by cloud consumers via a remote administration console include:

- configuring and setting up cloud services
- provisioning and releasing IT resource for on-demand cloud services
- monitoring cloud service status, usage, and performance
- monitoring QoS and SLA fulfillment
- managing leasing costs and usage fees
- managing user accounts, security credentials, authorization, and access control
- tracking internal and external access to leased services
- planning and assessing IT resource provisioning
- capacity planning

While the user-interface provided by the remote administration system will tend to be proprietary to the cloud provider, there is a preference among cloud consumers to work with remote administration systems that offer standardized APIs. This allows a cloud consumer to invest in the creation of its own front-end with the fore-knowledge that it can reuse this console if it decides to move to another cloud provider that supports the same standardized API. Additionally, the cloud consumer would be able to further leverage standardized APIs if it is interested in leasing and centrally administering IT resources from multiple cloud providers and/or IT resources residing in cloud and on-premise environments.

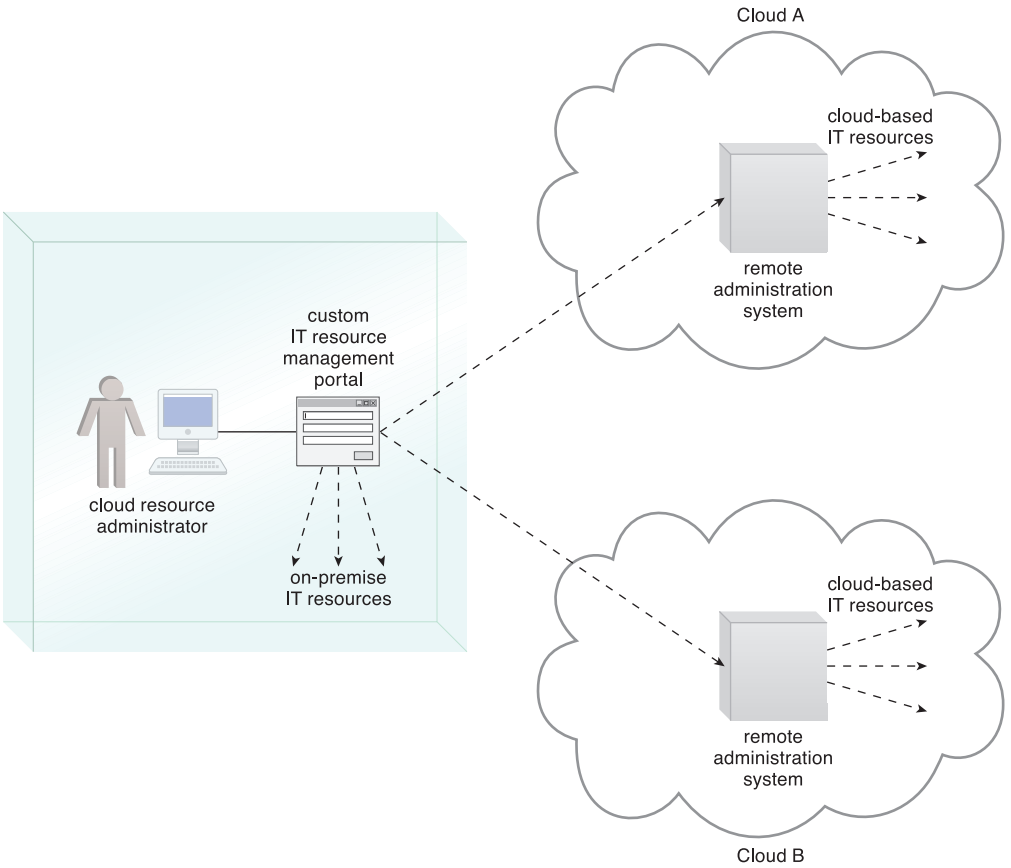


Figure 9.4

Standardized APIs published by remote administration systems from different clouds enable a cloud consumer to develop a custom portal that centralizes a single IT resource management portal for both cloud-based and on-premise IT resources.

CASE STUDY EXAMPLE

DTGOV has been offering its cloud consumers a user-friendly remote administration system for some time, and recently determined that upgrades are required in order to accommodate the growing number of cloud consumers and increasing diversity of requests. DTGOV is planning a development project to extend the remote administration system to fulfill the following requirements:

- Cloud consumers need to be able to self-provision virtual servers and virtual storage devices. The system specifically needs to interoperate with the cloud-enabled VIM platform's proprietary API to enable self-provisioning capabilities.
- A single sign-on mechanism (described in Chapter 10) needs to be incorporated to centrally authorize and control cloud consumer access.
- An API that supports the provisioning, starting, stopping, releasing, up-down scaling, and replicating of commands for virtual servers and cloud storage devices needs to be exposed.

In support of these features, a self-service portal is developed and the feature-set of DTGOV's existing usage and administration portal is extended.

9.2 Resource Management System

The *resource management system* mechanism helps coordinate IT resources in response to management actions performed by both cloud consumers and cloud providers (Figure 9.5). Core to this system is the virtual infrastructure manager (VIM) that coordinates the server hardware so that virtual server instances can be created from the most expedient underlying physical server. A VIM is a commercial product that can be used to manage a range of virtual IT resources across multiple physical servers. For example, a VIM can create and manage multiple instances of a hypervisor across different physical servers or allocate a virtual server on one physical server to another (or to a resource pool).

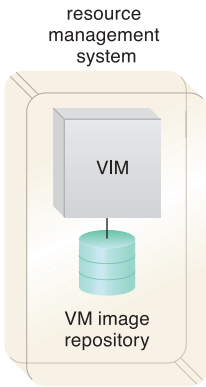


Figure 9.5
A resource management system encompassing a VIM platform and a virtual machine image repository. The VIM may have additional repositories, including one dedicated to storing operational data.

Tasks that are typically automated and implemented through the resource management system include:

- managing virtual IT resource templates that are used to create pre-built instances, such as virtual server images
- allocating and releasing virtual IT resources into the available physical infrastructure in response to the starting, pausing, resuming, and termination of virtual IT resource instances
- coordinating IT resources in relation to the involvement of other mechanisms, such as resource replication, load balancer, and failover system
- enforcing usage and security policies throughout the lifecycle of cloud service instances
- monitoring operational conditions of IT resources

Resource management system functions can be accessed by cloud resource administrators employed by the cloud provider or cloud consumer. Those working on behalf of a cloud provider will often be able to directly access the resource management system's native console.

Resource management systems typically expose APIs that allow cloud providers to build remote administration system portals that can be customized to selectively offer resource management controls to external cloud resource administrators acting on behalf of cloud consumer organizations via usage and administration portals.

Both forms of access are depicted in Figure 9.6.

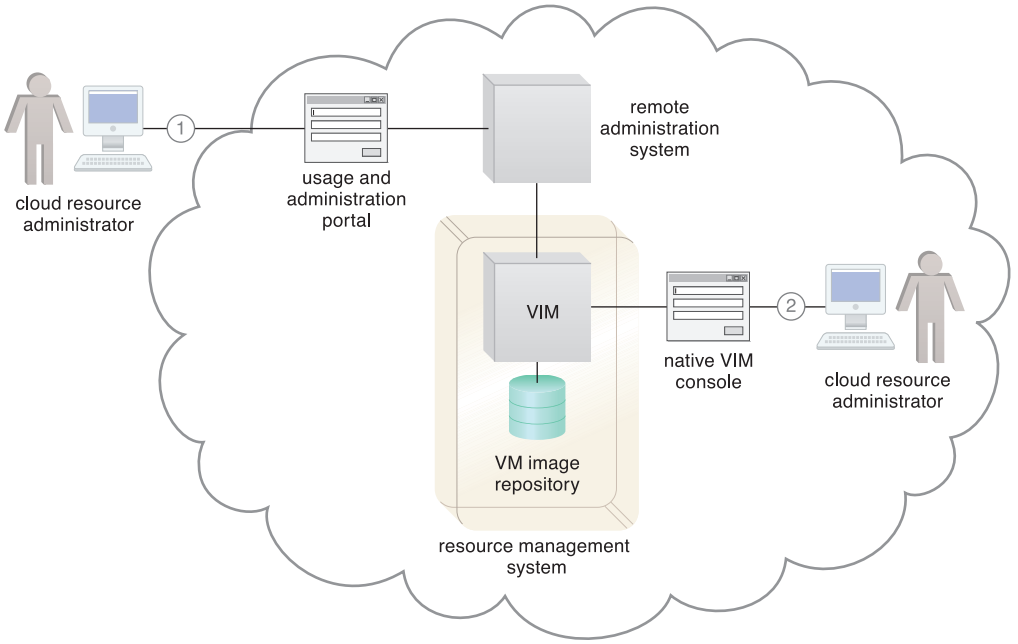


Figure 9.6
The cloud consumer's cloud resource administrator accesses a usage and administration portal externally to administer a leased IT resource (1). The cloud provider's cloud resource administrator uses the native user-interface provided by the VIM to perform internal resource management tasks (2).

CASE STUDY EXAMPLE

The DTGOV resource management system is an extension of a new VIM product it purchased, and provides the following primary features:

- management of virtual IT resources with a flexible allocation of pooled IT resources across different data centers
- management of cloud consumer databases
- isolation of virtual IT resources at logical perimeter networks
- management of a template virtual server image inventory available for immediate instantiation
- automated replication ("snapshotting") of virtual server images for virtual server creation

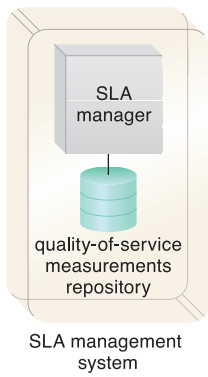
- automated up-down scaling of virtual servers according to usage thresholds to enable live VM migration among physical servers
- an API for the creation and management of virtual servers and virtual storage devices
- an API for the creation of network access control rules
- an API for the up-down scaling of virtual IT resources
- an API for the migration and replication of virtual IT resources across multiple data centers
- interoperation with a single sign-on mechanism through an LDAP interface

Custom-designed SNMP command scripts are further implemented to interoperate with the network management tools to establish isolated virtual networks across multiple data centers.

9.3 SLA Management System

The *SLA management system* mechanism represents a range of commercially available cloud management products that provide features pertaining to the administration, collection, storage, reporting, and runtime notification of SLA data (Figure 9.7).

Figure 9.7
An SLA management system encompassing an SLA manager and QoS measurements repository.



An SLA management system deployment will generally include a repository used to store and retrieve collected SLA data based on pre-defined metrics and reporting parameters. It will further rely on one or more SLA monitor mechanisms to collect the SLA data that can then be made available in near-real time to usage and administration portals to provide on-going feedback regarding active cloud services (Figure 9.8). The metrics monitored for individual cloud services are aligned with the SLA guarantees in corresponding cloud provisioning contracts.

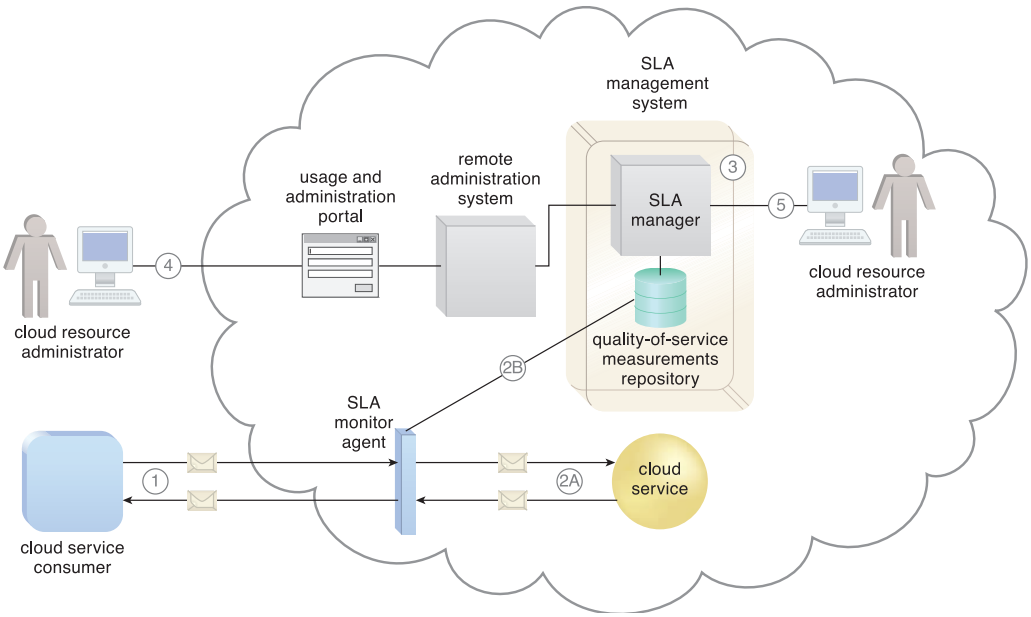


Figure 9.8

A cloud service consumer interacts with a cloud service (1). An SLA monitor intercepts the exchanged messages, evaluates the interaction, and collects relevant runtime data in relation to quality-of-service guarantees defined in the cloud service's SLA (2A). The data collected is stored in a repository (2B) that is part of the SLA management system (3). Queries can be issued and reports can be generated for an external cloud resource administrator via a usage and administration portal (4) or for an internal cloud resource administrator via the SLA management system's native user-interface (5).

CASE STUDY EXAMPLE

DTGOV implements an SLA management system that interoperates with its existing VIM. This integration allows DTGOV cloud resource administrators to monitor the availability of a range of hosted IT resources via SLA monitors.

DTGOV works with the SLA management system's report design features to create the following pre-defined reports that are made available via custom dashboards:

- *Per-Data Center Availability Dashboard* – Publicly accessible through DTGOV's corporate cloud portal, this dashboard shows the overall operational conditions of each group of IT resources at each data center, in realtime.
- *Per-Cloud Consumer Availability Dashboard* – This dashboard displays realtime operational conditions of individual IT resources. Information about each IT resource can only be accessed by the cloud provider and the cloud consumer leasing or owning the IT resource.
- *Per-Cloud Consumer SLA Report* – This report consolidates and summarizes SLA statistics for cloud consumer IT resources, including downtimes and other time-stamped SLA events.

The SLA events generated by the SLA monitors represent the status and performance of physical and virtual IT resources that are controlled by the virtualization platform. The SLA management system interoperates with the network management tools through a custom-designed SNMP software agent that receives the SLA event notifications.

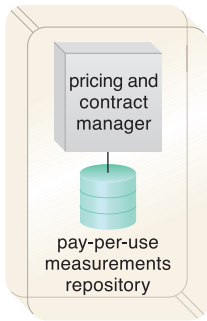
The SLA management system also interacts with the VIM through its proprietary API to associate each network SLA event to the affected virtual IT resource. The system includes a proprietary database used to store SLA events (such as virtual server and network downtimes).

The SLA management system exposes a REST API that DTGOV uses to interface with its central remote administration system. The proprietary API has a component service implementation that can be used for batch-processing with the billing management system. DTGOV utilizes this to periodically provide downtime data that translates into credit applied to cloud consumer usage fees.

9.4 Billing Management System

The *billing management system* mechanism is dedicated to the collection and processing of usage data as it pertains to cloud provider accounting and cloud consumer billing. Specifically, the billing management system relies on pay-per-use monitors to gather runtime usage data that is stored in a repository that the system components then draw from for billing, reporting, and invoicing purposes (Figures 9.9 and 9.10).

Figure 9.9
A billing management system comprised of a pricing and contract manager and a pay-per-use measurements repository.



The billing management system allows for the definition of different pricing policies, as well as custom pricing models on a per cloud consumer and/or per IT resource basis. Pricing models can vary from the traditional pay-per-use models, to flat-rate or pay-per-allocation modes, or combinations thereof.

Billing arrangements be based on pre-usage and post-usage payments. The latter type can include pre-defined limits or it can be set up (with the mutual agreement of the cloud consumer) to allow for unlimited usage (and, consequently, no limit on subsequent billing). When limits are established, they are usually in the form of usage quotas. When quotas are exceeded, the billing management system can block further usage requests by cloud consumers.

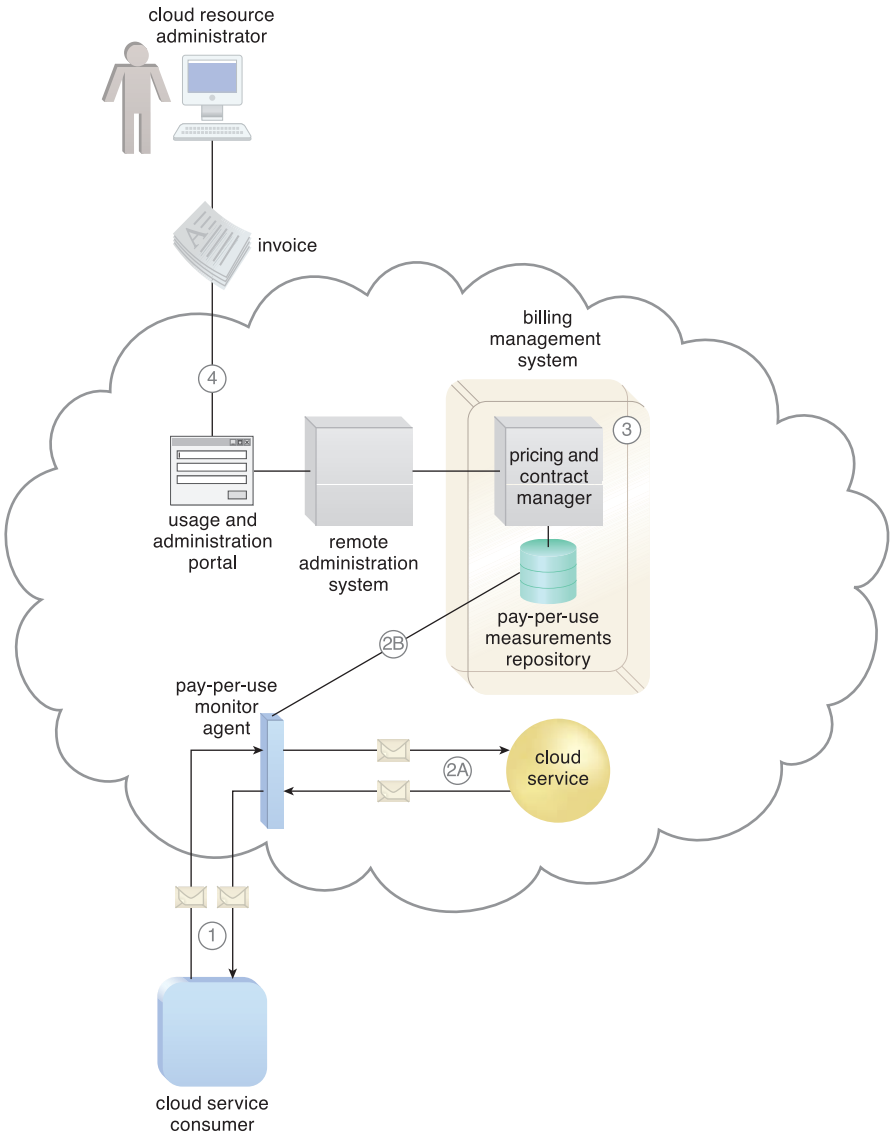


Figure 9.10

A cloud service consumer exchanges messages with a cloud service (1). A pay-per-use monitor keeps track of the usage and collects data relevant to billing (2A), which is forwarded to a repository that is part of the billing management system (2B). The system periodically calculates the consolidated cloud service usage fees and generates an invoice for the cloud consumer (3). The invoice may be provided to the cloud consumer through the usage and administration portal (4).

CASE STUDY EXAMPLE

DTGOV decides to establish a billing management system that enables them to create invoices for custom-defined billable events, such as subscriptions and IT resource volume usage. The billing management system is customized with the necessary events and pricing scheme metadata.

It includes the following two corresponding proprietary databases:

- billable event repository
- pricing scheme repository

Usage events are collected from pay-per-use monitors that are implemented as extensions to the VIM platform. Thin-granularity usage events, such as virtual server starting, stopping, up-down scaling, and decommissioning, are stored in a repository managed by the VIM platform.

The pay-per-use monitors further regularly supply the billing management system with the appropriate billable events. A standard pricing model is applied to most cloud consumer contracts, although it can be customized when special terms are negotiated.