MANO encompasses the following functions and services:

- **Orchestration**: Responsible for installing and configuring new network services (NS); NS lifecycle management; global resource management; and validation and authorization of resource requests.
- **VM manager**: Oversees lifecycle management of VM instances.
- **Infrastructure manager**: Controls and manages the interaction of a VM with computing, storage, and network resources under its authority, as well as their virtualization.

## OpenStack

OpenStack is an open-source software project of the OpenStack Foundation that aims to produce an open-source cloud operating system [ROSA14, SEFR12]. The principal objective is to enable creating and managing huge groups of virtual private servers in a cloud computing environment. OpenStack is embedded, to one degree or another, into datacenter infrastructure and cloud computing products offered by Cisco, IBM, Hewlett-Packard, and other vendors. It provides multitenant IaaS, and aims to meets the needs of public and private clouds regardless of size, by being simple to implement and massively scalable.

The OpenStack OS consists of a number of independent modules, each of which has a project name and a functional name. The modular structure is easy to scale out and provides a commonly used set of core services. Typically the components are configured together to provide a comprehensive IaaS capability. However, the modular design is such that the components are generally capable of being used independently.

To understand OpenStack it is useful to distinguish three types of storage that are part of the OpenStack environment:

- **Network block storage**: This type of storage makes data persistent by mounting one or more network block storage devices. It represents an allocation of persistent, readable, and writable block storage that could be utilized as the root disk for a VM instance, or as secondary storage that could be attached and/or detached from a VM instance.
- **Object storage**: Object storage is the persistent storage of objects on a network. From the object storage viewpoints, the objects are arbitrary, unstructured data. The storage objects are generally write-once, read-many. This is reliable storage with redundant copies. Access control lists determine visibility for the owner and authorized users.
- **Virtual machine image storage**: VM images are disk images that can be booted on a VM by a hypervisor. It can be a single image that contains the boot loader, kernel and operating system, or the boot loader and kernel can be separated. This type of storage allows for custom kernels and resizable images.

Figure 16.9, from [CALL15], illustrates the OpenStack conceptual architecture, with the interaction among the principal software components. Table 16.3 defines the functional interaction; the leftmost column indicates the source of an action, while the

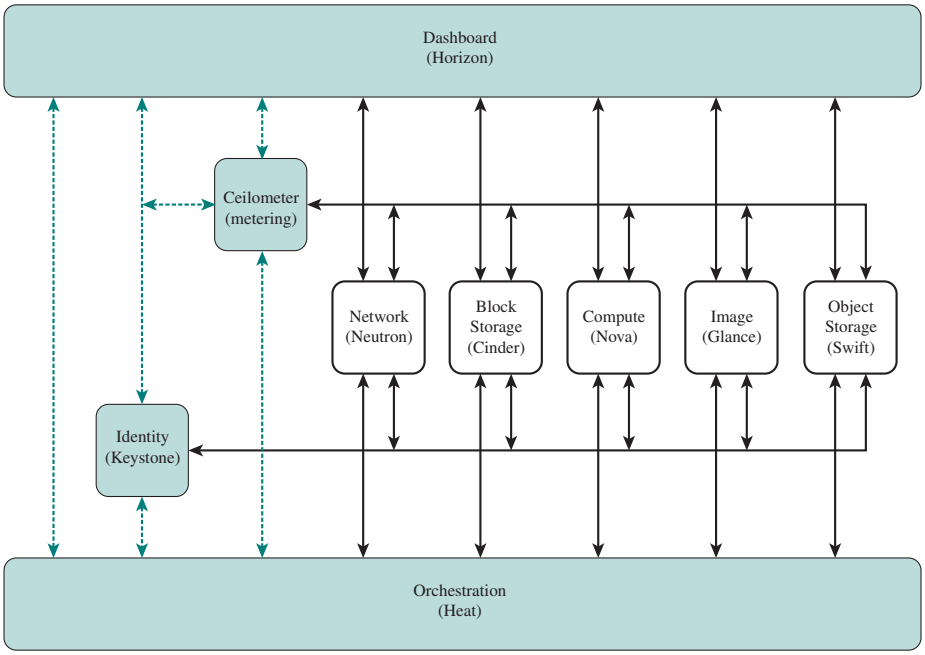


**Figure 16.9** **OpenStack High Level Architecture**

**Table 16.3** OpenStack Functional Interactions

| | **Glance (image)** | **Horizon (dashboard)** | **Nova (compute)** | **Swift (object storage)** | **Cinder (block storage)** | **Neutron (network)** |
|---|---|---|---|---|---|---|
| Glance (image) | | | sends images to | stores disk files | stores blocks on | |
| Horizon (dashboard) | provides UI | | provides UI | provides UI | provides UI | provides UI |
| Nova (compute) | receives images from | | | stores volumes on | | |
| Swift (object storage) | supplies disk files | | provides volumes for | | | |
| Cinder (block storage) | provides volumes for | | | | | |
| Neutron (network) | | | | | | |
| Keystone (identity) | authenticates with | authenticates with | authenticates with | authenticates with | authenticates with | authenticates with |
| Heat (orchestration) | orchestrates | orchestrates | orchestrates | orchestrates | orchestrates | orchestrates |
| Trove (database) | | | provides instances to | | | |
| Ceilometer | monitors | monitors | monitors | monitors | monitors | monitors |
| VM | retrieves image files | | | | | |

(*Continued*)

**Table 16.3** OpenStack Functional Interactions (*continued*)

| | **Keystone (identity)** | **Heat (orchestration)** | **Trove (database)** | **ceilometer** | **VM** |
|---|---|---|---|---|---|
| Glance (image) | authenticates with | | | | supplies image files |
| Horizon (dashboard) | authenticates with | provides UI | provides UI | provides UI | provides UI |
| Nova (compute) | authenticates with | | receives instances from | | launches volume |
| Swift (object storage) | authenticates with | | | | |
| Cinder (block storage) | authenticates with | | | | |
| Neutron (network) | authenticates with | | | | provides network connectivity |
| Keystone (identity) | | authenticates with | authenticates with | authenticates with | authenticates with |
| Heat (orchestration) | authenticates with | | orchestrates | orchestrates | |
| Trove (database) | authenticates with | | | | |
| Ceilometer | authenticates with | monitors | monitors | | |
| VM | authenticates with | | | | |

topmost row indicates the destination of an action. These components can be roughly divided into five functional groups:

- **Computing**: Compute (Nova), Image (Glance)
- **Networking**: Network (Neutron)
- **Storing**: Object Storage (Swift), Block Storage (Cinder)
- **Shared Services**: Security (Keystone), Dashboard (Keystone), Metering (Ceilometer), Orchestration (Heat)
- **Other Optional Services**: Discussed subsequently.

We now examine each of the components listed in the first four bullet items, then briefly discuss other components.

***COMPUTE (NOVA)*** Nova is the management software that controls virtual machines within the IaaS cloud computing platform. It manages the lifecycle of compute instances in an OpenStack environment. Responsibilities include spawning, scheduling, and decommissioning of machines on demand. Thus,

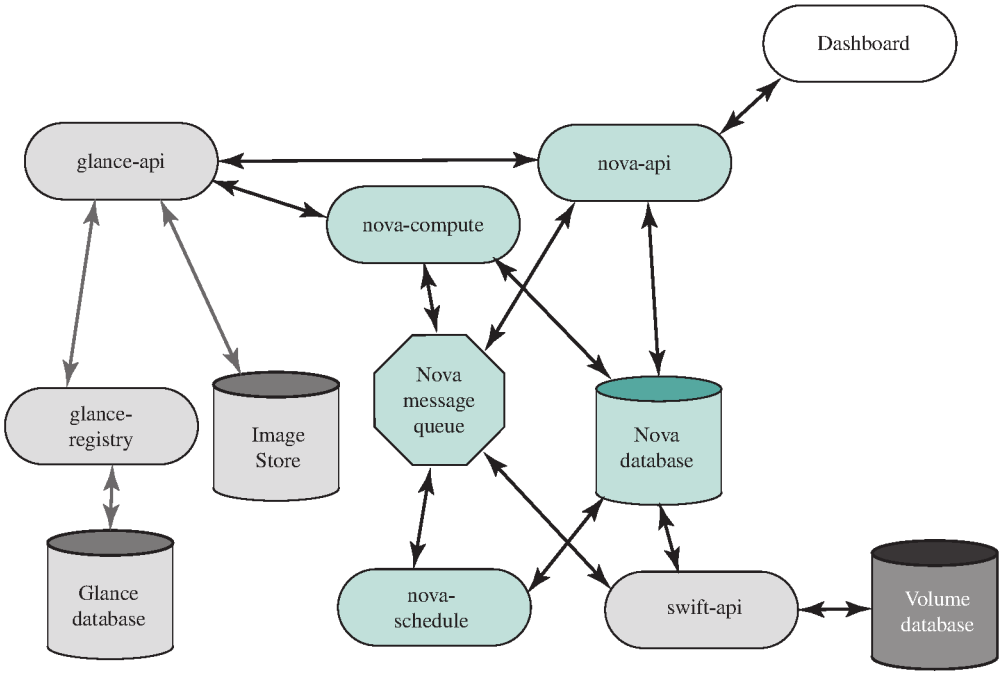


**Figure 16.10** **Nova Logical Architecture**

Nova enables enterprises and service providers to offer on-demand computing resources, by provisioning and managing large networks of virtual machines. Nova is similar in scope to Amazon Elastic Compute Cloud (EC2). Nova is capable of interacting with various open-source and commercial hypervisors. Nova does not include any virtualization software; rather, it defines drivers that interact with underlying virtualization mechanisms that run on the host operating system, and it provides functionality over a Web API. Thus, Nova enables the management of large networks of virtual machines and supports redundant and scalable architectures. It includes instances management for servers, networks, and access control. Nova requires no prerequisite hardware and is completely independent of the hypervisor.

Nova consists of five main components (see Figure 16.10):

- **API server**: This is the external interface to the Dashboard for users and applications.
- **Message queue**: Nova components exchange info through the queue (actions) and database (information) to carry out API requests. The message queue implements the mechanism for dispatching the exchanged instructions to facilitate communication.
- **Compute controller**: Handles the lifecycle of virtual machine instances, it is responsible for creating and manipulating virtual servers. It interacts with Glance.

- **Database**: Stores most of the build-time and run-time state for a cloud infrastructure. This includes the instance types that are available for use, instances in use, networks available and projects.
- **Scheduler**: Takes virtual machine instance requests and determines where (on which compute server host) they should be executed.

Note several components interact with Swift. Swift manages the creation, attaching and detaching of volumes to compute instances.

***IMAGE (GLANCE)*** Glance is a lookup and retrieval system for virtual machine (VM) disk images. It provides services for discovering, registering, and retrieving virtual images through an API. It also provides an SQL-style interface for queries for information on the images hosted on various storage systems. OpenStack Compute makes use of this during instance provisioning.

***NETWORK (NEUTRON)*** Neutron is an OpenStack project designed to provide network connectivity as a service between interface devices managed by other OpenStack services (e.g., NOVA). A Neutron server provides a Web server that exposes the Neutron API and passes all Web service calls to the Neutron plugin for processing. In essence, Neutron provides a consistent set of network services for use by other elements, such as virtual machines, systems management modules, and other networks. Users interact with networking functions via the Dashboard GUI; other management systems and networks interact with networking services using Neutron's API.

Currently Neutron implements Layer 2 virtual LANs (VLANs) and IP-based (Layer 3) routers. There are also extensions to support firewalls, load balancers, and IPSec virtual private networks (VPNs).

Three key benefits of using Neutron are the following [PARK13]:

- By using a consistent approach to networking for multiple types of virtual machines, Neutron helps providers operate efficiently in heterogeneous environments, which is frequently the requirement in service provider systems.
- By supplying a consistent set of APIs for plugging in a variety of physical network underlays, providers gain flexibility in altering the design of their underlying physical network while keeping the cloud service logically intact.
- Orchestration and system management suppliers, as well as providers' own technical teams, can use the Neutron API to integrate management of the network for the cloud with multiple higher-level service management tasks. This offers a range of opportunities, including service-level agreement monitoring, as well as integration into automation platforms like catalogs and portals for dynamic management of customer clouds.

***OBJECT STORAGE (SWIFT)*** Swift is a distributed object store that creates a redundant and scalable storage space of up to multiple petabytes of data. Object storage does not present a traditional file system, but rather a distributed storage system for static data such as virtual machine images, photo storage, email storage, backups, and archives. It can be used by Cinder components to back up VM volumes.

***BLOCK STORAGE (CINDER)*** Cinder provides persistent block storage (or volumes) to guest virtual machines. Cinder can use Swift to back up the VM's volumes. Cinder also interacts with Nova, providing volumes for its instances, allowing through its API the manipulation of volumes, volume types, and volume snapshots.

***IDENTITY (KEYSTONE)*** Keystone provides the shared security services essential for a functioning cloud computing infrastructure. It provides main services:

- **Identity**: This is user information authentication. This information defines a user's role and permissions within a project, and is the basis for a role-based access control (RBAC) mechanism.
- **Token**: After a username/password log on, a token is assigned and used for access control. OpenStack services retain tokens and use them to query Keystone during operations.
- **Service catalog**: OpenStack service endpoints are registered with Keystone to create a service catalog. A client for a service connects to Keystone, and determines an endpoint to call based on the returned catalog.
- **Policies**: This service enforces different user access levels.

Figure 16.11 illustrates the way in which Keystone interacts with other OpenStack components to launch a new virtual machine.

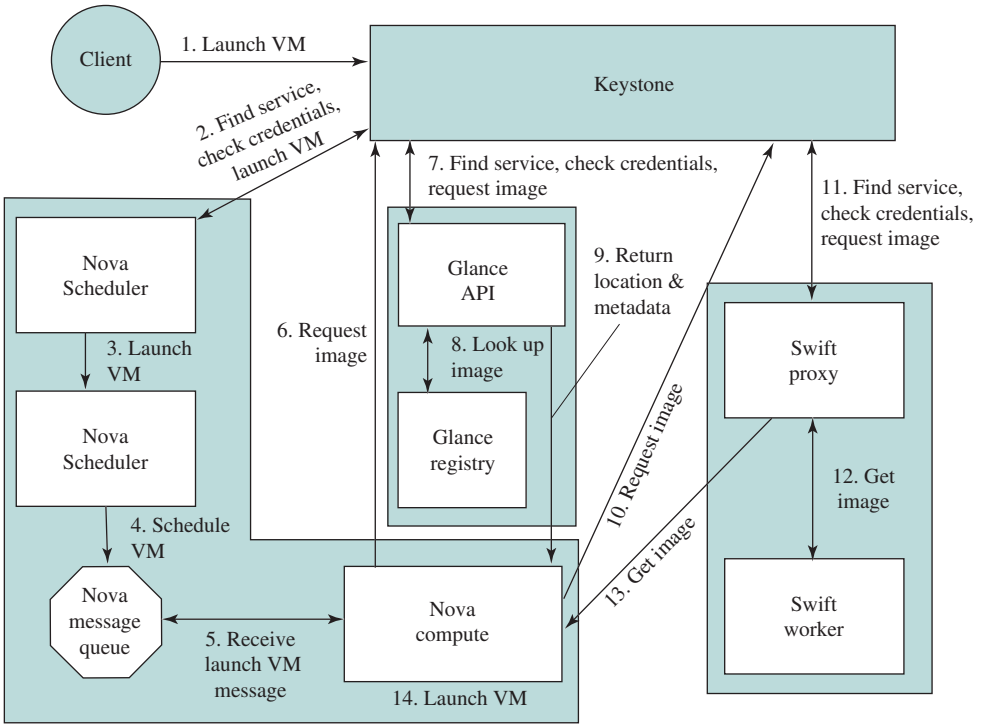


**Figure 16.11 Launching a Virtual Machine**

***Dashboard (Horizon)*** The dashboard is the Web user interface for cloud infrastructure management. It provides administrators and users a graphical interface to access, provision, and automate cloud-based resources. The extensible design makes it easy to plug in and expose third-party products and services, such as billing, monitoring, and additional management tools. It interacts with the APIs of all the other software components. For example, Horizon enables a user or application to launch an instance, assign IP addresses, and configure access controls.

***Monitor (Ceilometer)*** Ceilometer provides a configurable collection of functions for metering data, such as processor and storage usage and network traffic. This is a unique point of contact for billing, benchmarking, scalability, and statistical purposes.

***Orchestration (Heat)*** Heat orchestrates multiple cloud applications. The objective is to create a human- and machine-accessible service for managing the entire lifecycle of infrastructure and applications within OpenStack clouds. It implements an orchestration engine to launch multiple composite cloud applications based on templates in the form of text files that can be treated like code. Heat is compatible with Amazon Cloudformation, which is becoming a de facto standard.

***Other Optional Services*** As the OpenStack project evolves, new components are being developed by various OpenStack members. As of this writing, the following components are available or in development:

- **Database (Trove)**: Trove is a database-as-a-service that provisions relational and nonrelational database engines. By default, Trove uses MySQL as its relational database management system, enabling the other services to store configurations and management information.
- **Messaging service (Zaqar)**: Zaqar is a multitenant cloud messaging service for Web and mobile developers. The service features an API that developers can use to send messages between various components of their SaaS and mobile applications, by using a variety of communication patterns. Underlying this API is an efficient messaging engine designed with scalability and security in mind.
- **Key management (Barbican)**: Barbican provides an API for the secure storage, provisioning, and management of secret values such as passwords, encryption keys, and X.509 Certificates.
- **Governance (Congress)**: Congress provides policy as a service across any collection of cloud services in order to offer governance and compliance for dynamic infrastructures.
- **Elastic map reduce (Sahara)**: Sahara aims to provide users with simple means to provision Hadoop clusters by specifying several parameters such as Hadoop version, cluster topology, and nodes hardware details. After a user fills all the parameters, Sahara deploys the cluster. Sahara also provides means to scale an already provisioned cluster by adding and removing worker nodes on demand.

- **Shared Filesystems (Manila)**: Manila provides coordinated access to shared or distributed file systems. While the primary consumption of file shares is across OpenStack Compute instances, the service is also intended to be accessible as an independent capability.
- **Containers (Magnum)**: Magnum provides an API service for making container orchestration engines such as Docker and Kubernetes available as resources in OpenStack.
- **Bare-metal provisioning (Ironic)**: Ironic provisions bare-metal machines instead of virtual machines, forked from the Nova baremetal driver. It is best thought of as a bare-metal hypervisor API and a set of plugins that interact with the bare-metal hypervisors.
- **DNS service (Designate)**: Designate provides DNS services for OpenStack users, including an API for domain/record management.
- **Application catalog (Murano)**: Murano introduces an application catalog to OpenStack, enabling application developers, and cloud administrators to publish various cloud-ready applications in a browsable categorized catalog.

These modular components are easily configured to enable an IaaS cloud service provider to tailor a cloud OS to its particular mission.

## 16.3 THE INTERNET OF THINGS

The Internet of Things is the latest development in the long and continuing revolution of computing and communications. Its size, ubiquity, and influence on everyday lives, business, and government dwarf any technical advance that has gone before. This section provides a brief overview of the Internet of Things, which is dealt with in greater detail later in the book.

### Things on the Internet of Things

The Internet of Things (IoT) is a term that refers to the expanding interconnection of smart devices, ranging from appliances to tiny sensors. A dominant theme is the embedding of short-range mobile transceivers into a wide array of gadgets and everyday items, enabling new forms of communication between people and things, and between things themselves. The Internet now supports the interconnection of billions of industrial and personal objects, usually through cloud systems. The objects deliver sensor information, act on their environment, and in some cases modify themselves, to create overall management of a larger system, like a factory or city.

The IoT is primarily driven by deeply embedded devices. These devices are low-bandwidth, low-repetition data capture, and low-bandwidth data-usage appliances that communicate with each other and provide data via user interfaces. Embedded appliances, such as high-resolution video security cameras, video VoIP phones, and a handful of others, require high-bandwidth streaming capabilities. Yet countless products simply require packets of data to be intermittently delivered.