Introduzione ai servizi di OpenStack

Enrico Fattibene
INFN – CNAF

Tutorial Days di CCR
Napoli 17/12/2014

OpenStack

Cosa è OpenStack

- È un insieme di software con lo scopo di fornire infrastrutture cloud pubbliche o private, largamente scalabili
  - servizi di cloud storage, compute, and networking
- Ha un disegno architetturale aperto e modulare, principalmente sviluppato in Python
- Interopera con altri sistemi di virtualizzazione e di Cloud computing pubblici o privati
  - Ad es. VMware ESXi, Microsoft Hyper-V, Amazon EC2

Principi di OpenStack

- Modello di sviluppo **open source**
  - Con dipendenze di tipo open source
  - Può essere eseguito su piattaforme interamente open source (ad es. Linux)
- Processo di **sviluppo aperto**
  - Design summit ogni 6 mesi, in cui gli sviluppatori ricevono requisiti e scrivono le specifiche per la release successiva
- **Comunità aperta**
  - Decisioni prese con modello del tacito assenso
  - Tutti i processi sono documentati e trasparenti

Chi partecipa ad OpenStack

- Fondato da NASA e Rackspace nel 2010
- Collaborazione di sviluppatori e utenti di dimensioni mondiali
- Forte supporto da parte dell'industria
  - ad es. Rackspace, Intel, Cisco, Juniper, NetApp, HP, DELL, VMware, AT&T, IBM, Canonical, SUSE, RedHat, Yahoo!
- Governance interna ben definita che non è in mano a nessun singolo ente o impresa

Sponsor principali

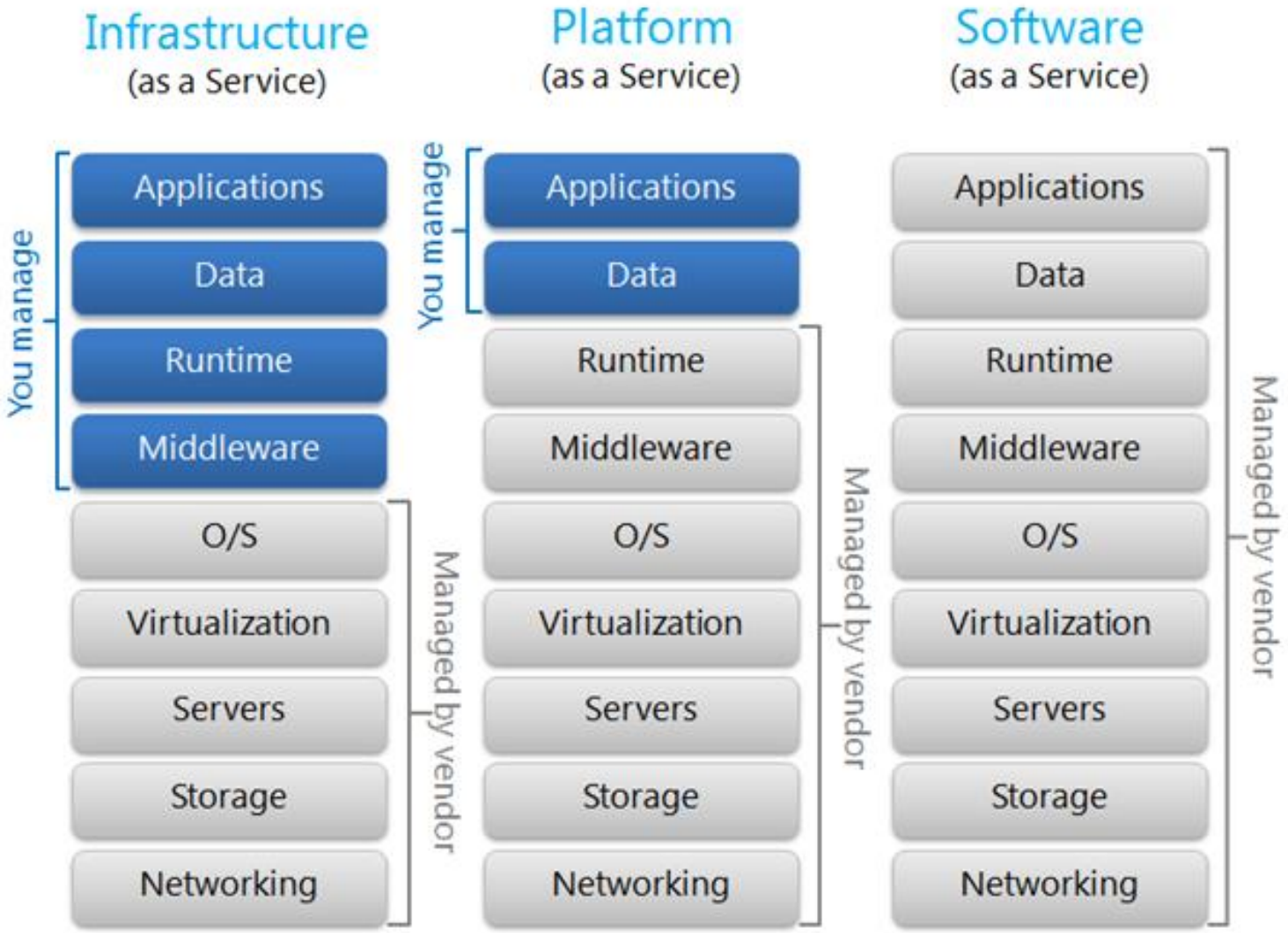
8 platinum
($500K/y)

19 gold
($50K÷200K/y)

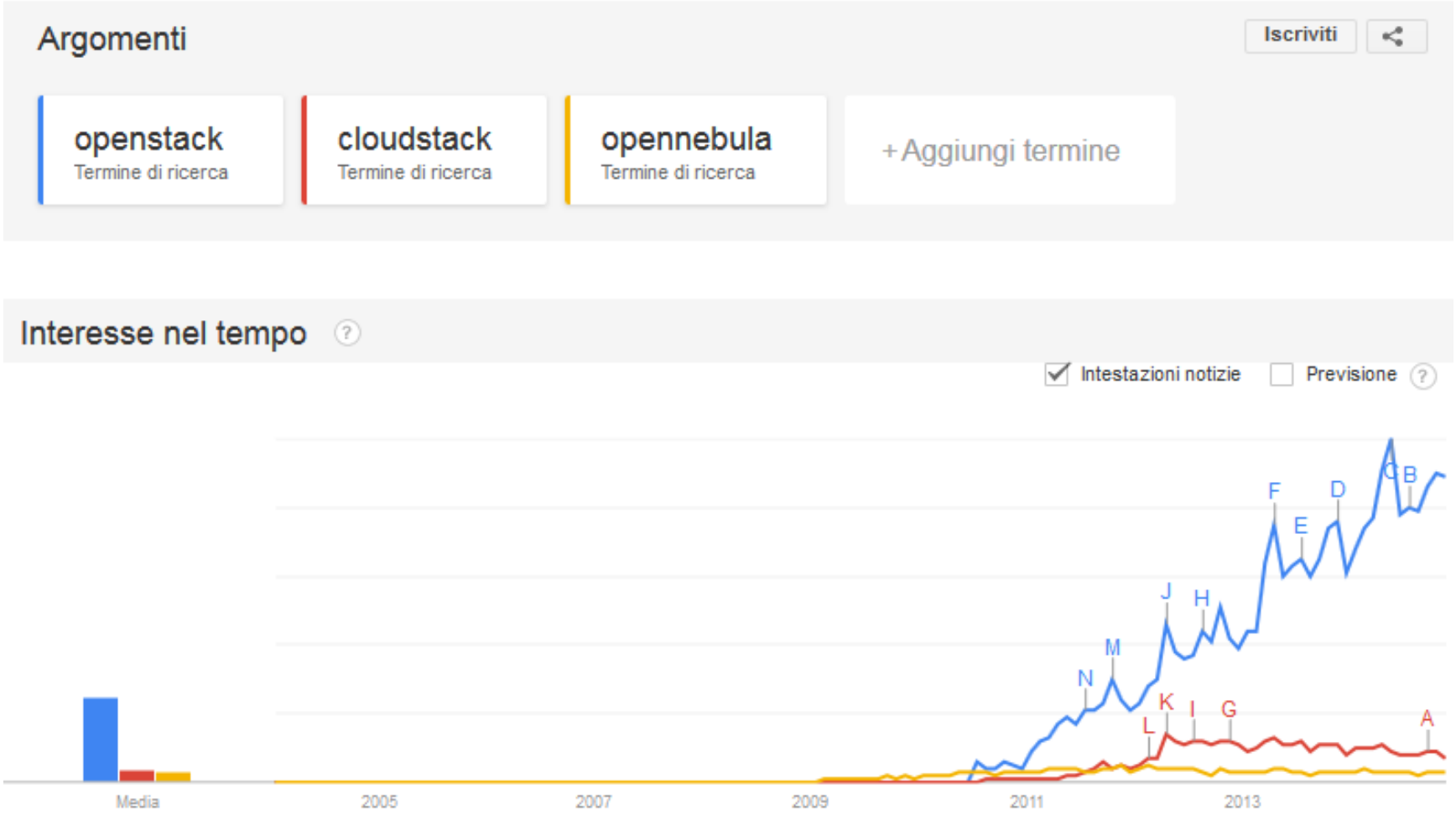
Qualche numero

- Summit di autunno 2014: Parigi
  - Affluenza record
    - >4600 tra sviluppatori e utenti (30% in più rispetto a un anno fa)
    - 49% dall'Europa (9% nell'edizione precendete ad Atlanta)
    - 60 nazioni
  - 1 azienda italiana tra gli espositori
  - Prossime edizioni
    - Vancouver (Primavera 2015)
    - Tokio (Autunno 2015)
- Sviluppo OpenStack
  - 2700 sviluppatori
  - 125 aziende
  - 130 paesi
  - 20 milioni di linee di codice

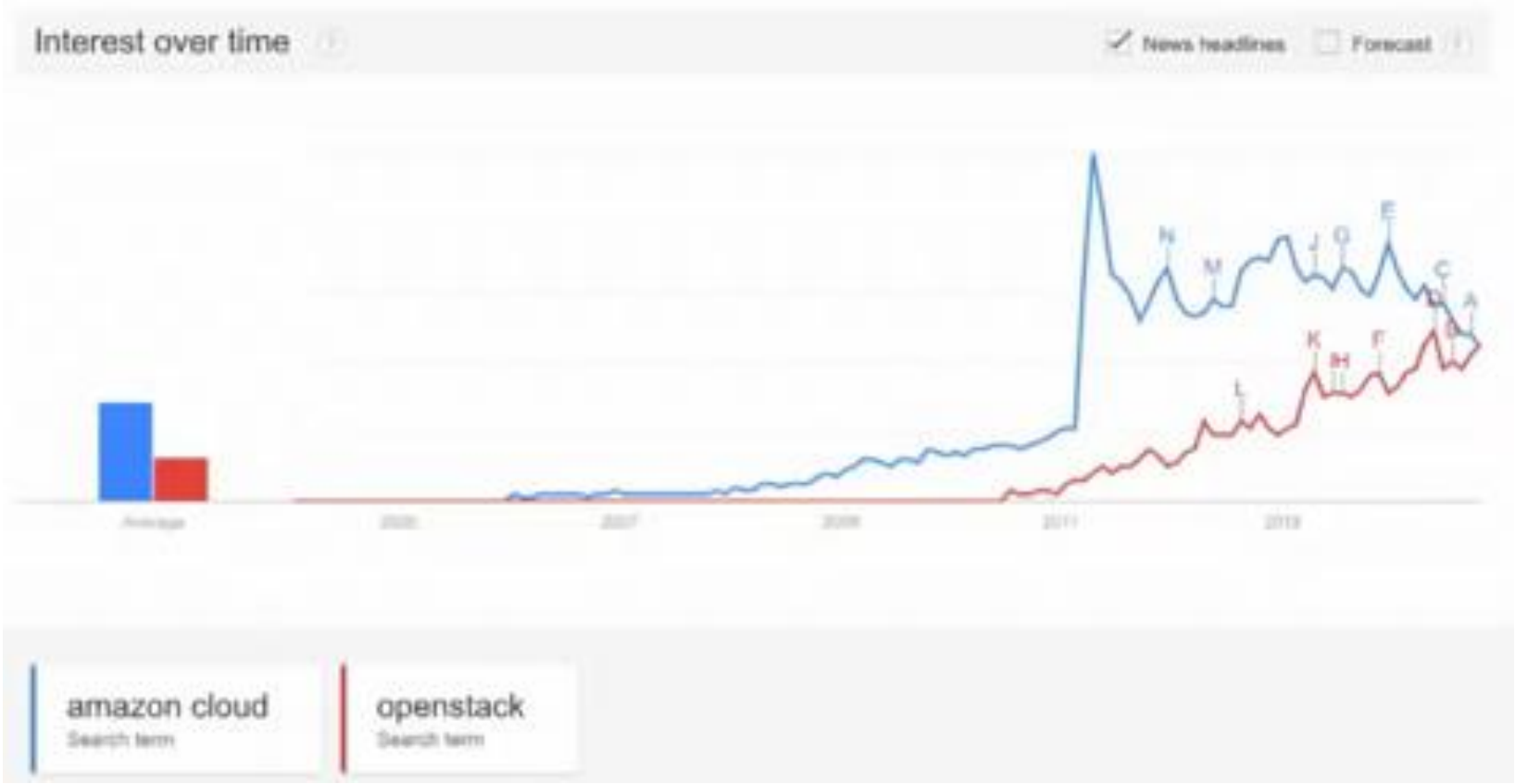
OpenStack: piattaforma IaaS



Trend di OpenStack



Amazon vs OpenStack



Distribuzione Cloud

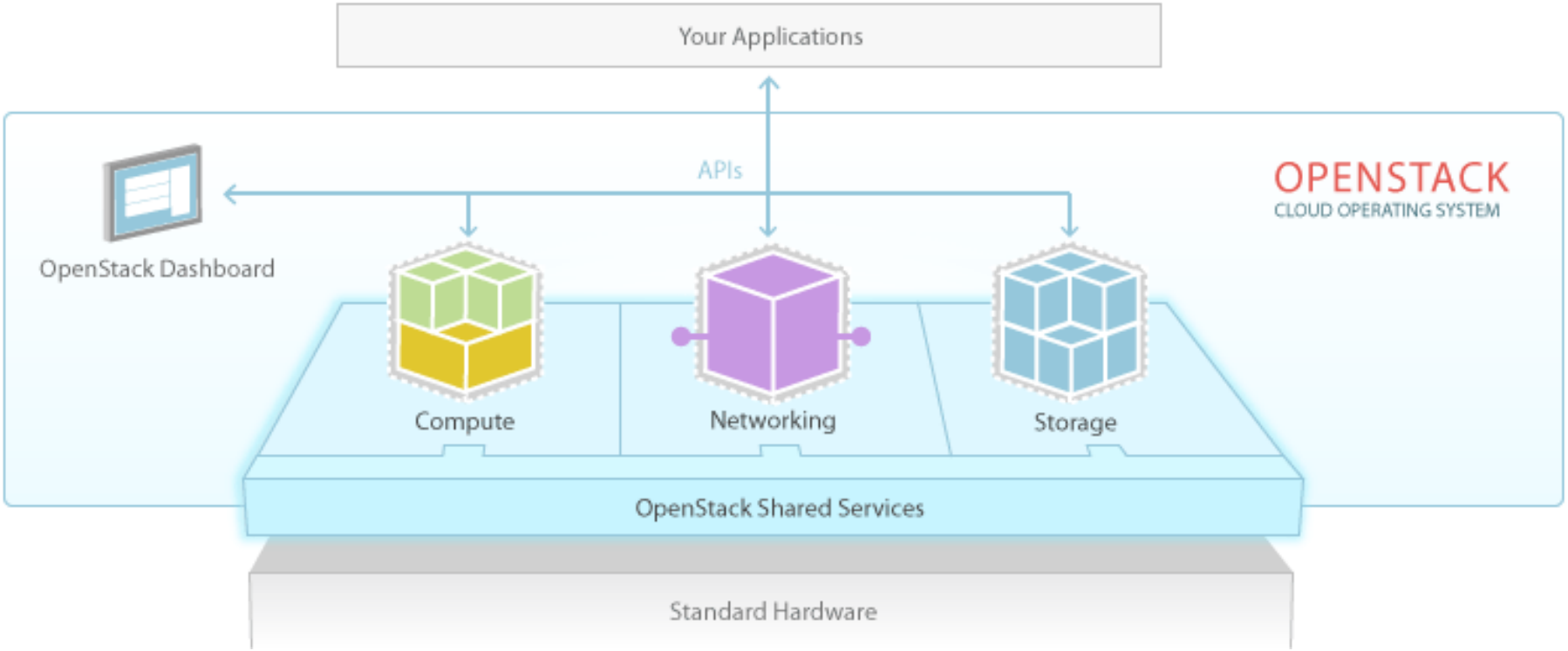
- Cloud pubbliche
  - Amazon
  - OpenStack

- Cloud private OpenStack
  - (da dati dell'ultima User Survey)

I nomi delle release

- In OpenStack ogni ***major release*** del software ha un nome in codice.
  - Il ciclo di rilascio delle major release è attualmente di 6 mesi (http://goo.gl/gMRhb).
  - La versione attuale, rilasciata a **ottobre 2014**, è chiamata **Juno**. È la decima release di OpenStack (http://goo.gl/MlPbu).
  - La versione successiva, chiamata *Kilo*, è prevista uscire il 30 aprile 2015.

Visione d'insieme

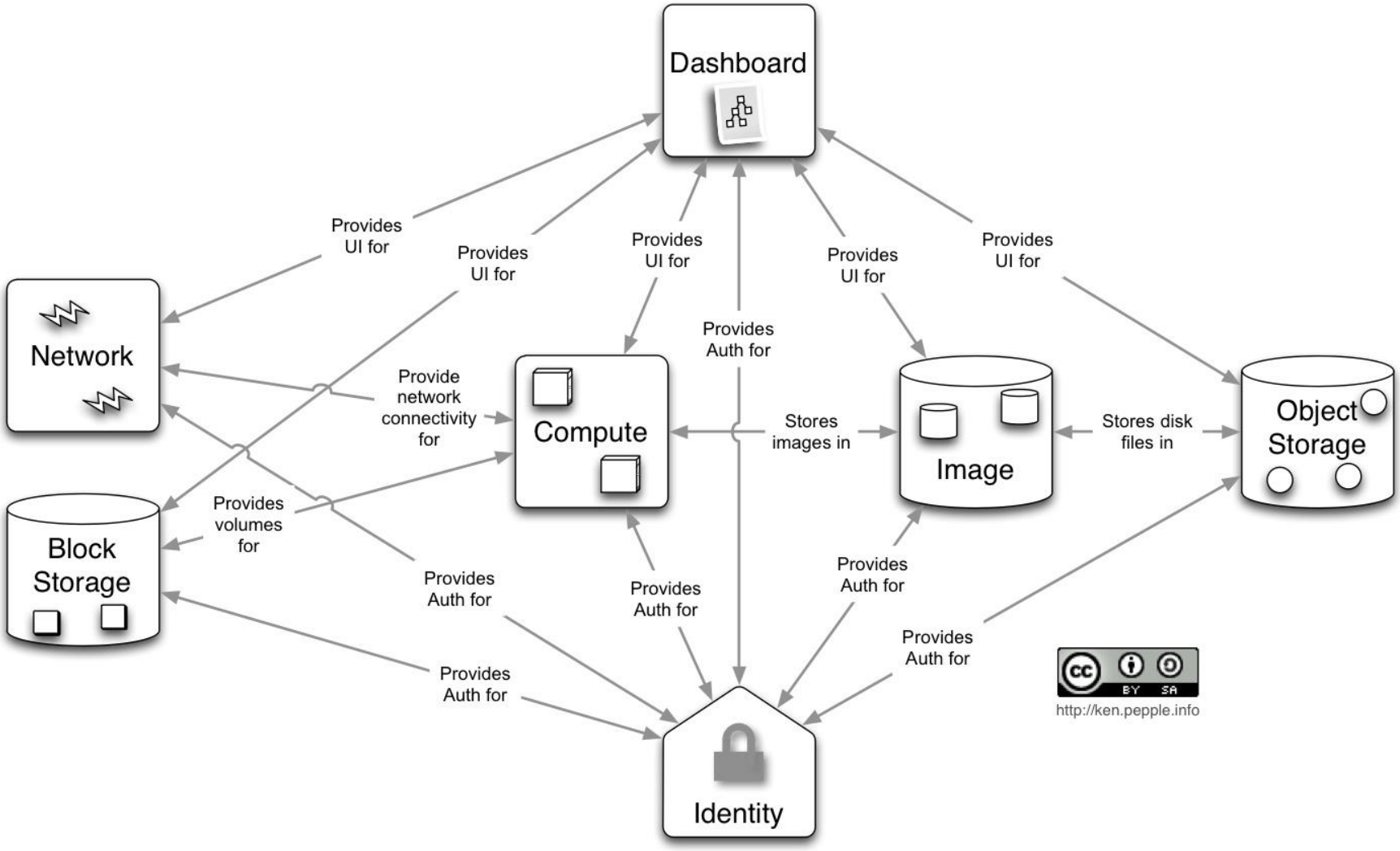


Fonte: OpenStack

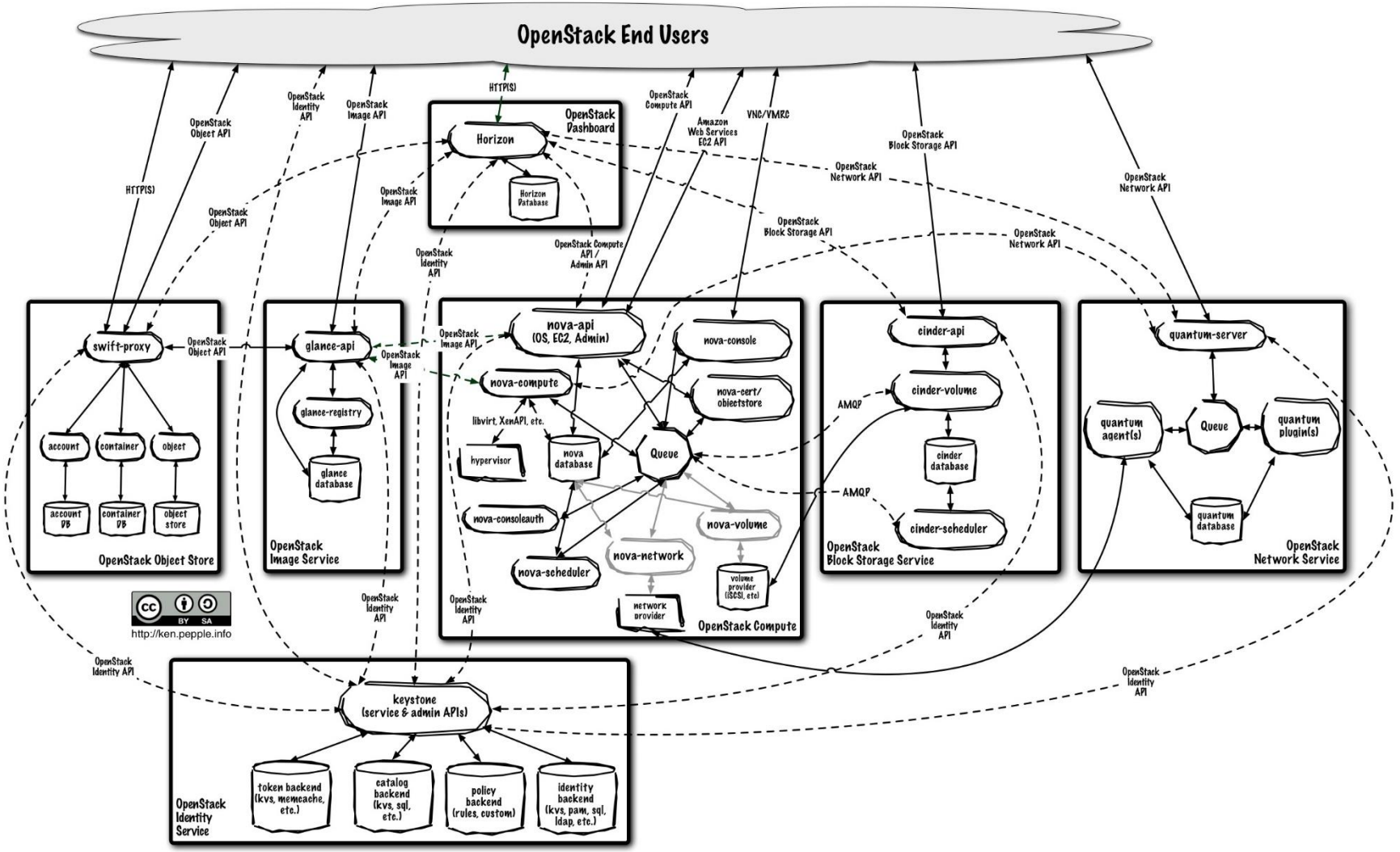
I nomi dei componenti

- Ogni componente funzionale (modulo) di OpenStack ha un nome in codice. Questi sono i moduli presenti nella release Juno:
  - Dashboard (web interface) → **Horizon**
  - Servizio di immagini (catalogo, repository) → **Glance**
  - Compute (server virtuali) → **Nova**
  - Network (gestione della rete) → **Neutron (Quantum)**
  - Identità (autenticazione, autorizzazione) → **Keystone**
  - Metering (controllo d'uso delle risorse) → **Ceilometer**
  - Orchestration (descrizione e automazione deployment dell'infrastruttura) → **Heat**
  - Block storage (gestione dei volumi) → **Cinder**
  - Object storage (gestione di files) → **Swift**
  - Database service (supporto di DBaaS per-tenant) → **Trove**
  - Data processing (fornitura di un cluster Hadoop) → **Sahara**

L’architettura a moduli



Il diavolo sta nei dettagli…



Ma in sintesi…

- Gli utenti finali interagiscono con i servizi attraverso una interfaccia web comune, oppure attraverso API specifiche di ogni servizio.
- Tutti i servizi hanno una autenticazione comune.
- I singoli servizi interagiscono tra di loro attraverso le rispettive API pubbliche.

Horizon (dashboard)

- Horizon è il modulo che fornisce l'interfaccia utente via Web per amministratori e per utenti finali.
  - È scritto in Django (http://goo.gl/WWtun), un framework per lo sviluppo in Python di applicazioni web.
    - La personalizzazione è resa possibile dalla separazione tra la parte di presentazione e quella di interfacciamento con il resto di OpenStack
- Nota: non è indispensabile usare Horizon come dashboard.
  - Si può senza problemi sviluppare un proprio pannello di controllo (magari solo per gli utenti) che comunichi con le API di OpenStack.
  - Scelta adottata da alcuni fornitori di servizi Cloud.

Utenti, tenant e ruoli

- **Utente**: una rappresentazione digitale di una persona, di un sistema o di un servizio che utilizza una parte di OpenStack.
- **Tenant**: un insieme che raggruppa risorse, oggetti, utenti.
  - Ad esempio: un'organizzazione, un progetto, una sede, un gruppo.
- **Ruolo**: i privilegi che vengono assegnati a un certo utente.
  - amministratore (globale, non del singolo tenant), o membro.

Keystone (identità)

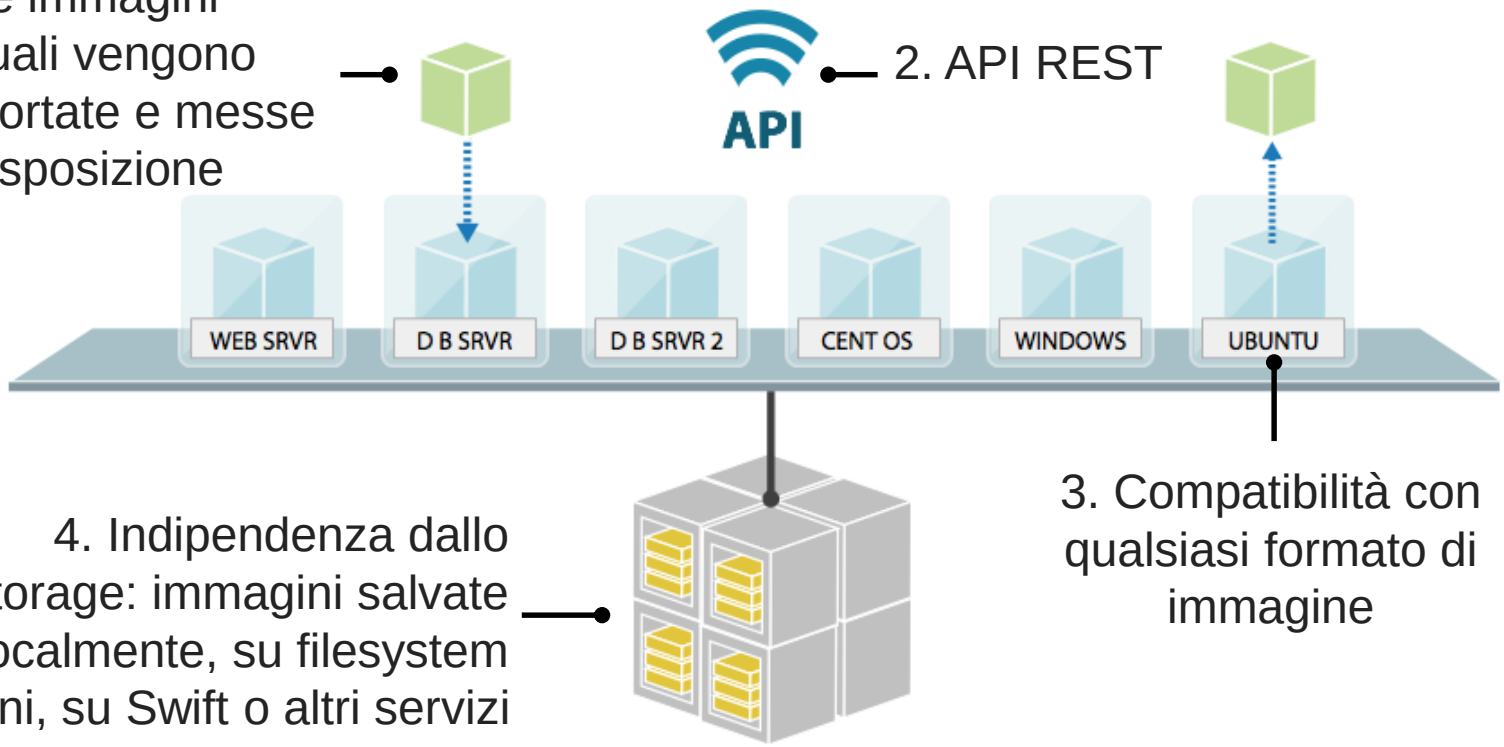
- Keystone implementa un unico componente di autenticazione per i diversi moduli di OpenStack. Fornisce autorizzazioni per credenziali di log-in multiple.
  - Identità: validazione delle credenziali per utenti, tenant e ruoli.
  - Token: validazione e gestione dei token per l'autenticazione, una volta che le credenziali siano stati verificate.
  - Catalogo dei servizi, con informazioni sugli endpoint (indirizzi accessibili via rete).
  - Gestione delle policy di autorizzazione.
- Supporta diversi back-end come identity provider.
  - il back-end di default è MySQL
  - è possibile memorizzare credenziali e dati di autorizzazione in un back-end separato (ad es. LDAP)
- E' possibile utilizzare un meccanismo di autenticazione esterna.

Glance (gestione immagini)

- È un servizio autonomo (può essere installato anche indipendentemente da OpenStack) che fornisce un catalogo per la memorizzazione e la gestione di immagini virtuali.
- È diviso in tre blocchi fondamentali:
  - Le API.
  - Un database (spesso MySQL o SQlite).
  - Il registro delle immagini. Questo può essere (come nei diagrammi precedenti) Swift, ma anche un altro tipo di servizio di memorizzazione delle immagini.
- Supporta immagini di diversi formati
  - Raw, AMI, VHD (Hyper-V), VDI (VirtualBox), qcow2 (QEMU/KVM), VMDK (VMware), OVF (VMware, altri).
- Può servire per salvare snapshots delle VM

Glance - schema grafico

1.Le immagini virtuali vengono importate e messe a disposizione



2. API REST

3. Compatibilità con qualsiasi formato di immagine

4. Indipendenza dallo storage: immagini salvate localmente, su filesystem esterni, su Swift o altri servizi di cloud storage

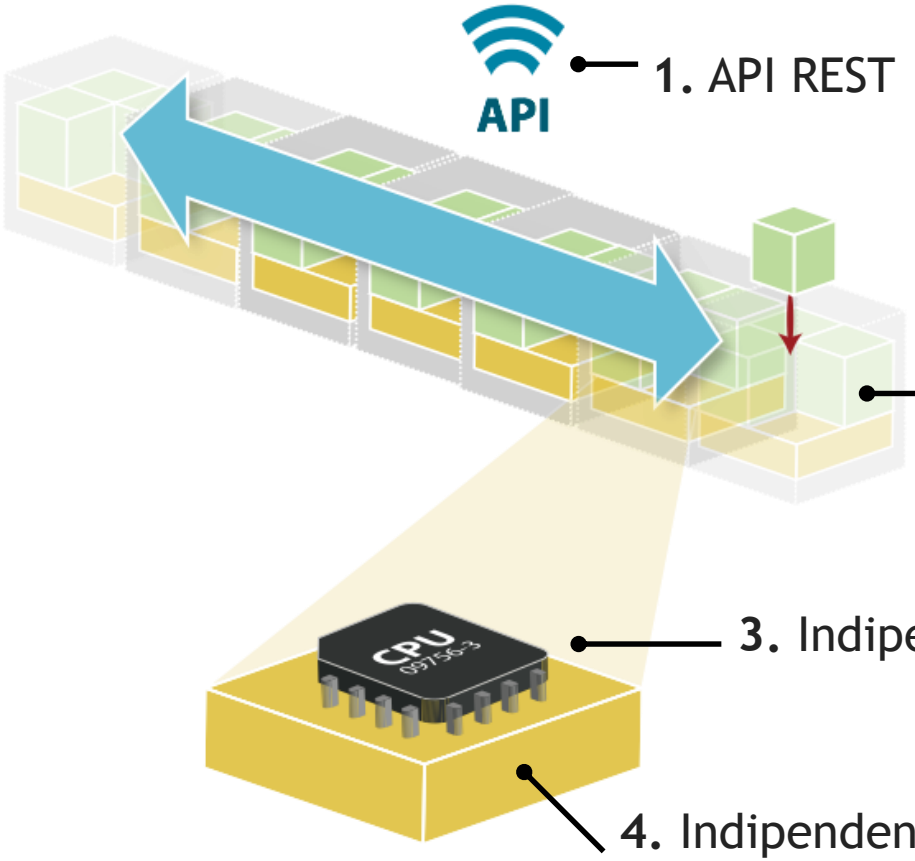
Nova (compute)

- È un framework per la fornitura e la gestione su larga scala di istanze virtuali.
- Supporta diversi tipi di hypervisor (cf. http://goo.gl/mVdjG per dettagli).
  - XenServer, KVM, QEMU, LXC, ESXi, Hyper-V
- Nova utilizza
  - un sistema di messaggistica per la comunicazione con gli altri componenti basato su code che sfrutta l'Advanced Message Queuing Protocol (AMQP, http://goo.gl/WmxSO).
    - L'implementazione di AMQP può essere data ad esempio da prodotti come Apache Qpid, RabbitMQ o ZeroMQ.
  - un database per la memorizzazione delle configurazioni e degli stati a run-time dell'infrastruttura, ad esempio quali tipi di istanza sono disponibili, quali istanze sono in uso, etc.
    - Normalmente si usa MySQL o PostgreSQL.

Nova - servizi

- nova-api
  - accetta e risponde alle richieste di computazione da parte degli utenti
  - supporta le API Compute di OpenStack, EC2 di Amazon e delle speciali API Admin per permettere ad utenti privilegiati di effettuare operazioni di amministrazione
- nova-scheduler
  - prende da una coda una richiesta di creazione di una macchina virtuale e determina su quale host fisico deve essere eseguita
  - sistema modulare personalizzabile con algoritmi complessi
- nova-console, nova-vncproxy, nova-consoleauth
  - si occupano della gestione dell'accesso alle console delle macchine virtuali
- nova-compute
  - è un demone che crea e termina le istanze di macchine virtuali attraverso le API dell'hypervisor
  - preleva i task da una coda, esegue i comandi di sistema relativi al task prelevato ed aggiorna nel database lo stato delle operazioni effettuate

Nova - schema grafico



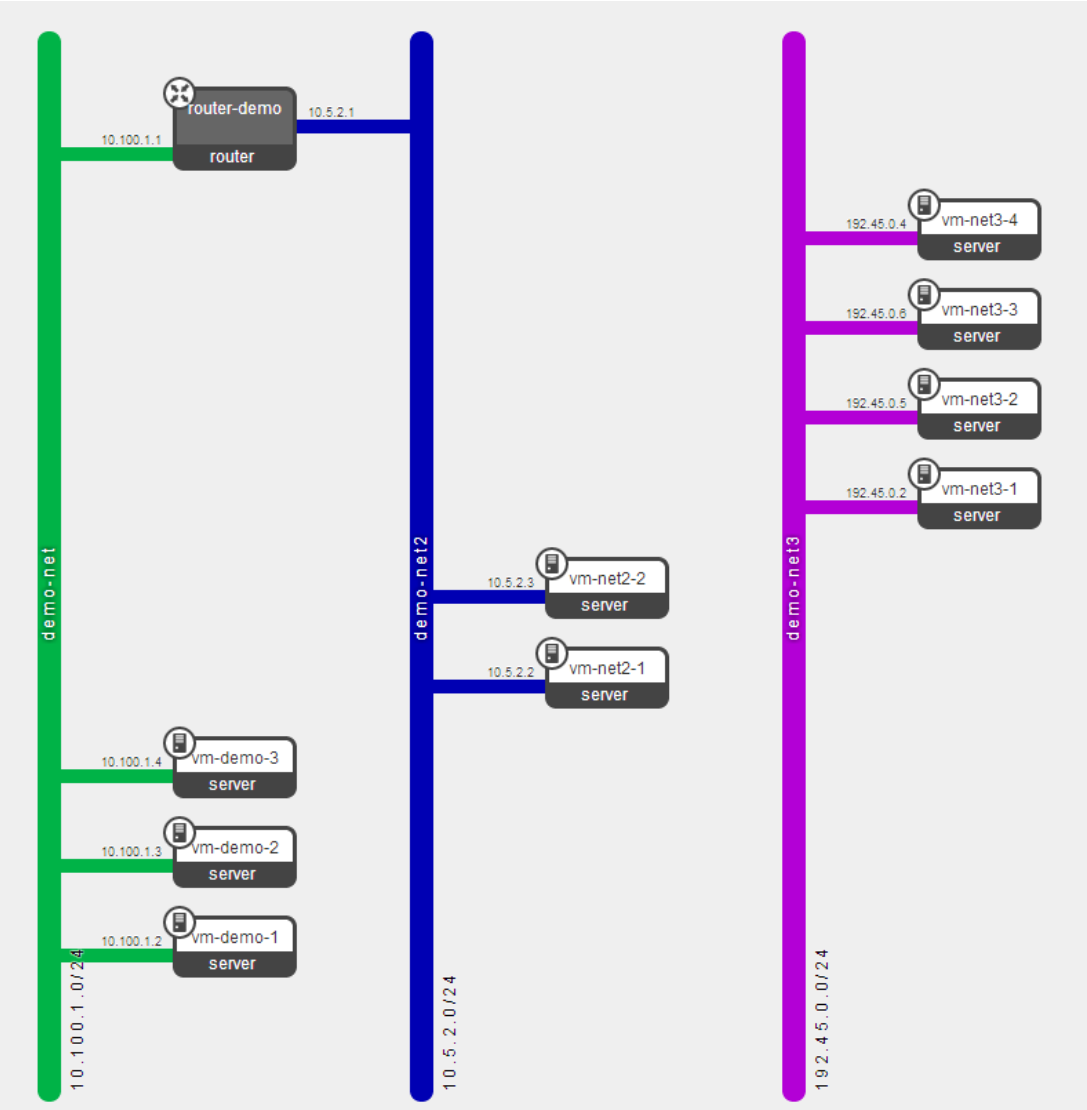
Neutron (rete)

- A partire da Folsom, è disponibile il componente Neutron che implementa il concetto di “Network as a Service”.
  - Prima la rete era gestita dal servizio nova-network
  - “Servizio” tra interfacce (per esempio interfacce virtuali) gestite da Nova
  - Come altri componenti in OpenStack, è composto da plug-in per la massima configurabilità.
  - E’ possibile di scrittura di plug-in esterni per realizzare ad esempio servizi come:
    - Load Balancer as a Service (LBaaS), cf. http://goo.gl/mtmxsY
    - VPN as a Service (VPNaaS), cf. http://goo.gl/da4NnM
    - Firewall as a Service (FWaaS), cf. http://goo.gl/dwO9KI

Tipologie di rete (1/3)

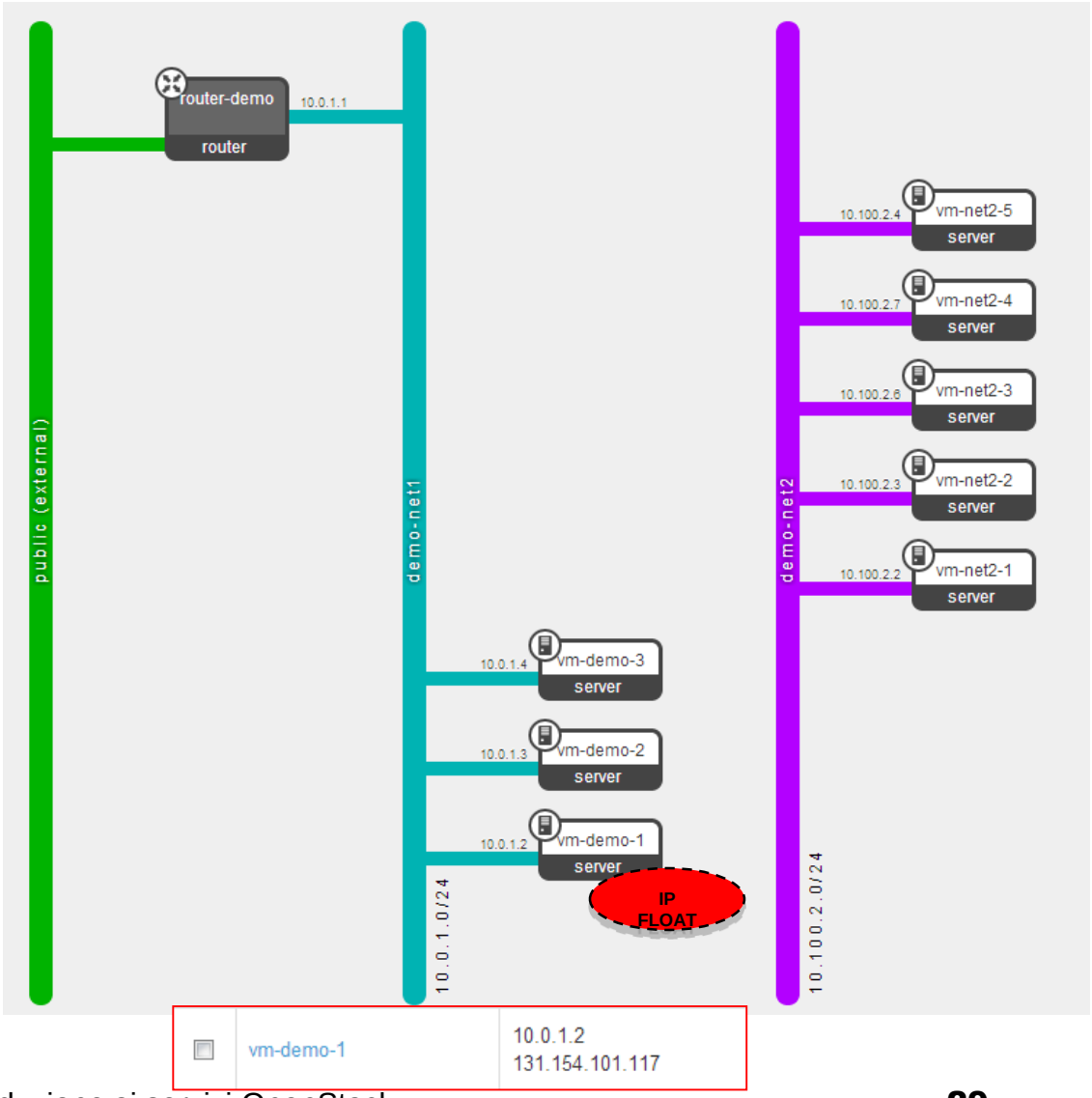
- Private Tenant Network
  - All'interno di ogni progetto (Tenant), gli utenti possono creare una o piu' reti private e uno o piu' apparati di rete (router) con cui connettere le reti in vario modo.
  - La definizione delle reti in ogni tenant non richiede l'intervento di un amministratore dell'infrastruttura ed e' garantito l'isolamento delle VM sia tra progetti diversi che tra reti diverse all'interno dello stesso progetto.
  - Le VM possono (o meno) avere outbound connectivity (Masquerade NAT), ma non sono raggiungibili dall'esterno se non hanno un floating IP assegnato (vedi slide successive).

Private Tenant Network



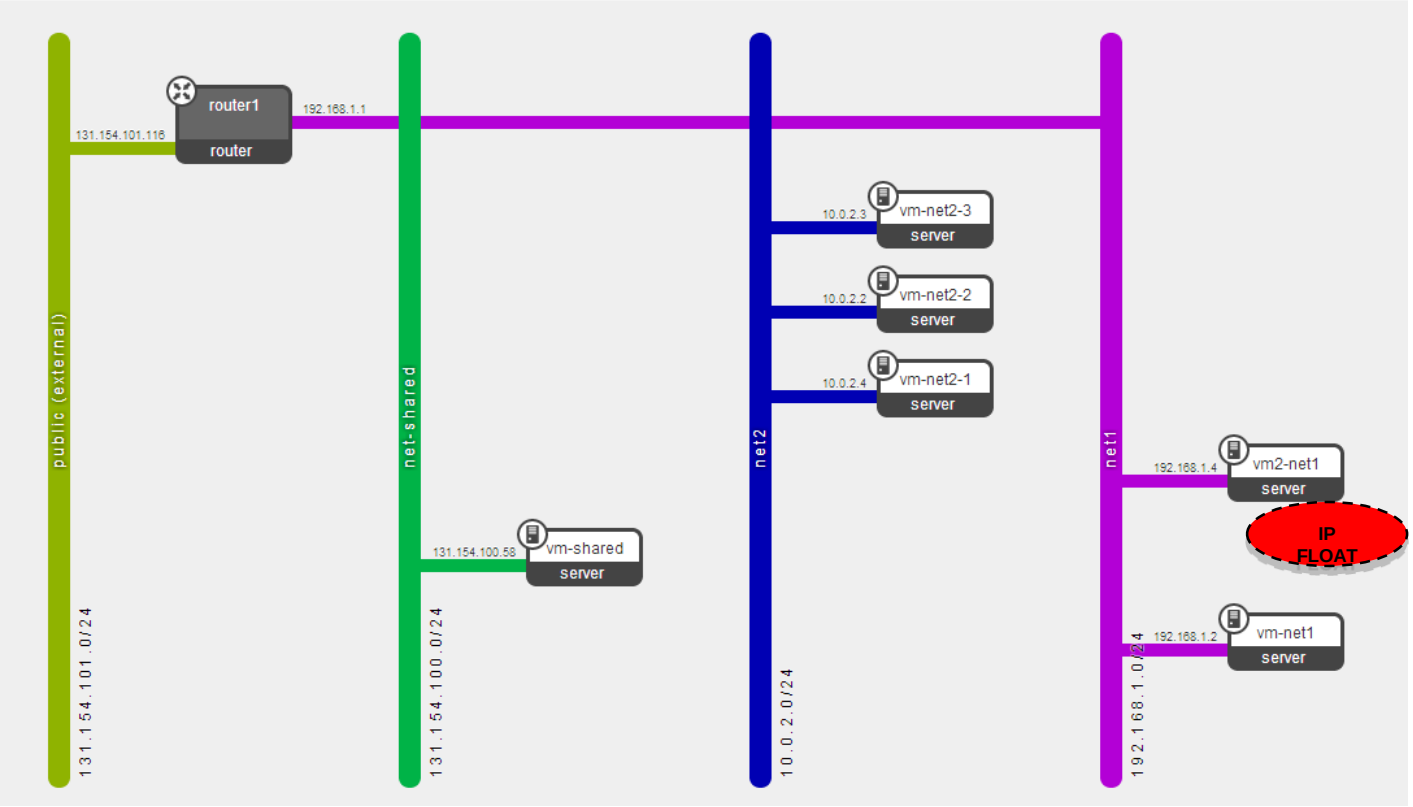
Tipologie di rete (2/3)

- External Network
  - Tipologia di rete necessaria per assegnare floating IP a VM istanziate sulle Private Tenant Network e renderle quindi accessibili via NAT dall'esterno.
  - Le reti External sono condivise tra tutti i progetti e definite dall'amministratore dell'infrastruttura.
  - Le VM non possono partire con un ip assegnato su una rete External, deve esistere una rete Private.
- Workflow
  - L'utente crea una rete privata
  - L'utente crea un router che connette la rete External con la rete privata
  - L'utente fa partire una VM sulla rete privata a cui assegna anche un floating IP
  - La VM ha due IP: uno privato e uno pubblico



Tipologie di rete (3/3)

- Shared Network
- La shared network e' una tipologia di rete attraverso la quale si mette a disposizione dell'infrastruttura OpenStack una rete esistente nel centro di calcolo per poter creare delle VM sulla rete stessa.
- Le VM possono essere instanziate sulla rete shared, l'IP viene loro fornito da un DHCP esterno.
- Le policy della rete shared non sono gestite da OpenStack. No NAT a nessun livello.



Cinder (block storage)

- Nel *block storage* si creano dischi virtuali da legare alle singole macchine virtuali.
- Permette di estendere la quantità di spazio di archiviazione che le macchine virtuali hanno a disposizione, legando un numero arbitrario di dischi, secondo le specifiche necessità.
- Pensiamo ad una chiavetta USB che possiamo di volta in volta montare su pc diversi; in maniera analoga un volume di Cinder puo' essere montato di volta in volta su VM diverse.

Swift (object storage)

- Swift è un object store distribuito, scalabile, multi-tenant e ad alta disponibilità.
  - Il goal è la memorizzazione a basso costo di grandi moli di dati non strutturati attraverso API di tipo REST.
  - Supporta direttamente le API S3 di Amazon, gestisce quote, controllo accessi e si interfaccia a diversi sistemi di storage.
  - Non è:
    - Un file system (non si può montare sulle VM).
    - Pensato per memorizzare dei DB live o delle gerarchie di file.
- È pensato per scalabilità, alta affidabilità, gestione di elevata concorrenza nelle transazioni, storage replicato geograficamente.
- Una possibile applicazione di Swift non legata direttamente a Cloud storage ad esempio può essere quella di utilizzarlo per la distribuzione geografica di immagini Glance.

Ceilometer (telemetry)

- Misurazione del consumo delle risorse
  - per scopi di monitoring e accounting
- Raccolta delle metriche di utilizzo dai componenti Openstack
- Eventuale aggregazione delle metriche raccolte
- Pubblicazione delle metriche verso diverse destinazioni
  - incluso il collector di Ceilometer stesso
- Conservazione delle metriche
  - di default in un database MongoDB
- Lettura delle metriche attraverso API Rest

Heat (orchestration)

- Servizio di orchestrazione di componenti, supportato a partire dalla release Grizzly
- Stack
  - Un insieme di risorse Cloud connesse tra loro (ad es. VM, volumi, reti, etc.)
- Gli stack vengono creati attraverso templates
  - Utilizzano le stesse strutture e le stesse astrazioni presenti in Amazon CloudFormation (http://goo.gl/4tvGl)
  - Possibilità di gestire alta affidabilità e autoscaling in maniera automatica

Conclusioni

- **OpenStack** è un framework Cloud open
  - sta crescendo molto rapidamente, sia come maturità che come feature supportate
  - d'altra parte (e probabilmente è una cosa in qualche modo *voluta*) è **complicato** e richiede un certo numero di competenze, test ed ottimizzazioni
  - è il prodotto open di più grande prospettiva in ambito Cloud
  - presenta diverse opportunità come base per la creazione e lo sviluppo di applicazioni avanzate grazie alle sue possibilità di adattamento