Cloud@CNAF
The long road to Juno

Matteo Panella (INFN-CNAF/SDDS)
On behalf of the Cloud@CNAF team

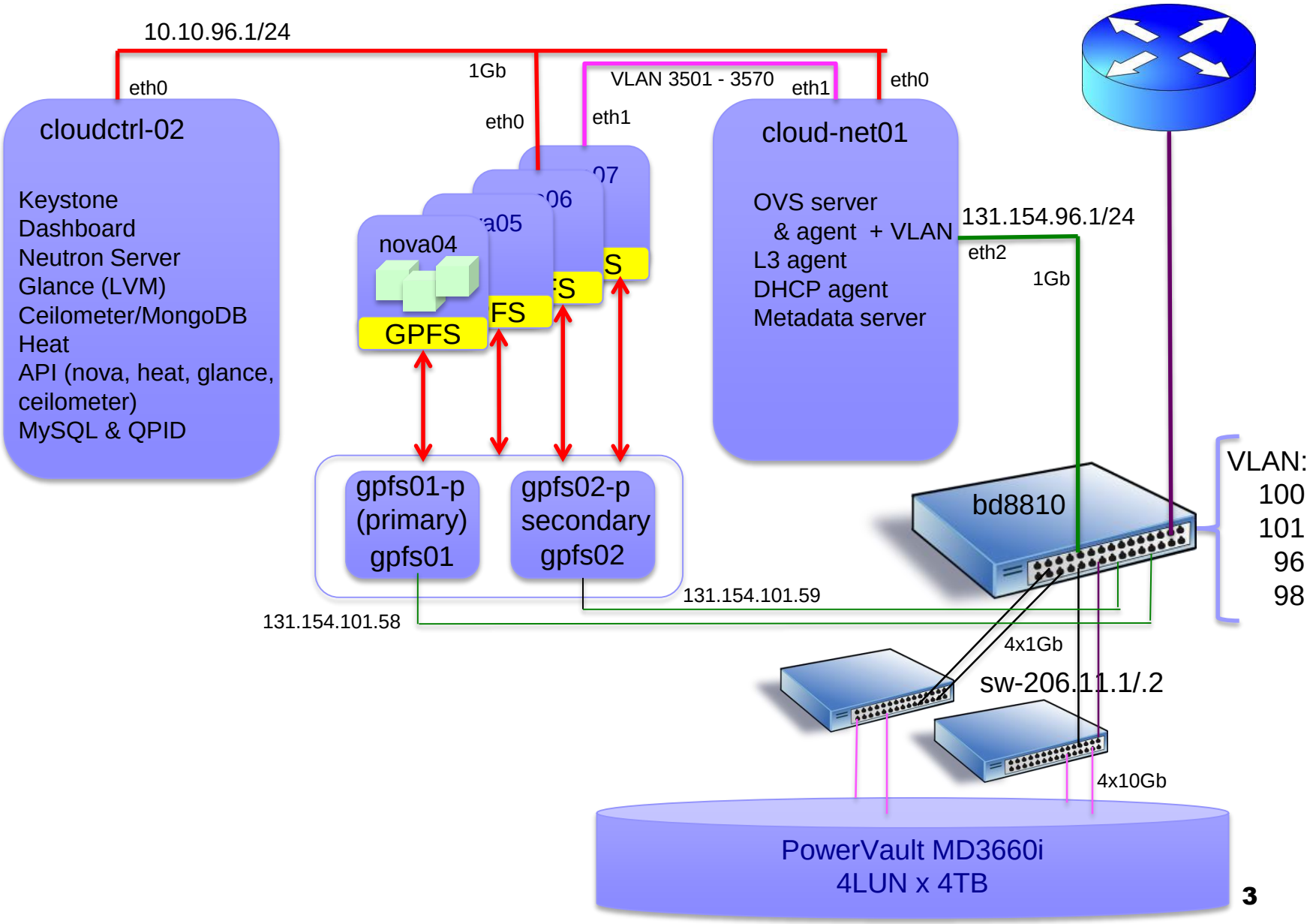
Agenda

- Dove eravamo: Havana
- Destinazione: Juno
- Un percorso “accidentato”:
  - GPFS
  - systemd
- Use case
- Convalida: Rally

Dove eravamo: Havana

- Openstack Havana – no HA, SSL solo per Horizon
- 1 Controller Node
  - Keystone, Glance (LVM) , Heat, Horizon, Ceilometer, MySQL, QPID
  - 2x8 HT (32) Intel(R) Xeon(R) CPU E5-2450 0 @ 2.10GHz, 64 GB di RAM
- 1 Network Node
  - Neutron con OVS + VLAN
  - 2x6HT (24) Intel(R) Xeon(R) CPU E5-2450 0 @ 2.10GHz, 64 GB di RAM
- 4 x Compute Node
  - Nova per KVM/QEMU
  - 2 x 8 core AMD con 64 GB RAM per nodo (tot: 64 core e 256 GB di RAM)
- Storage condiviso (PowerVault + 2 server GPFS)
  - 16 TB su GPFS per backend di Nova
- 1 Web Proxy per la dashboard

Cloud@CNAF - Havana



Dove eravamo: Havana

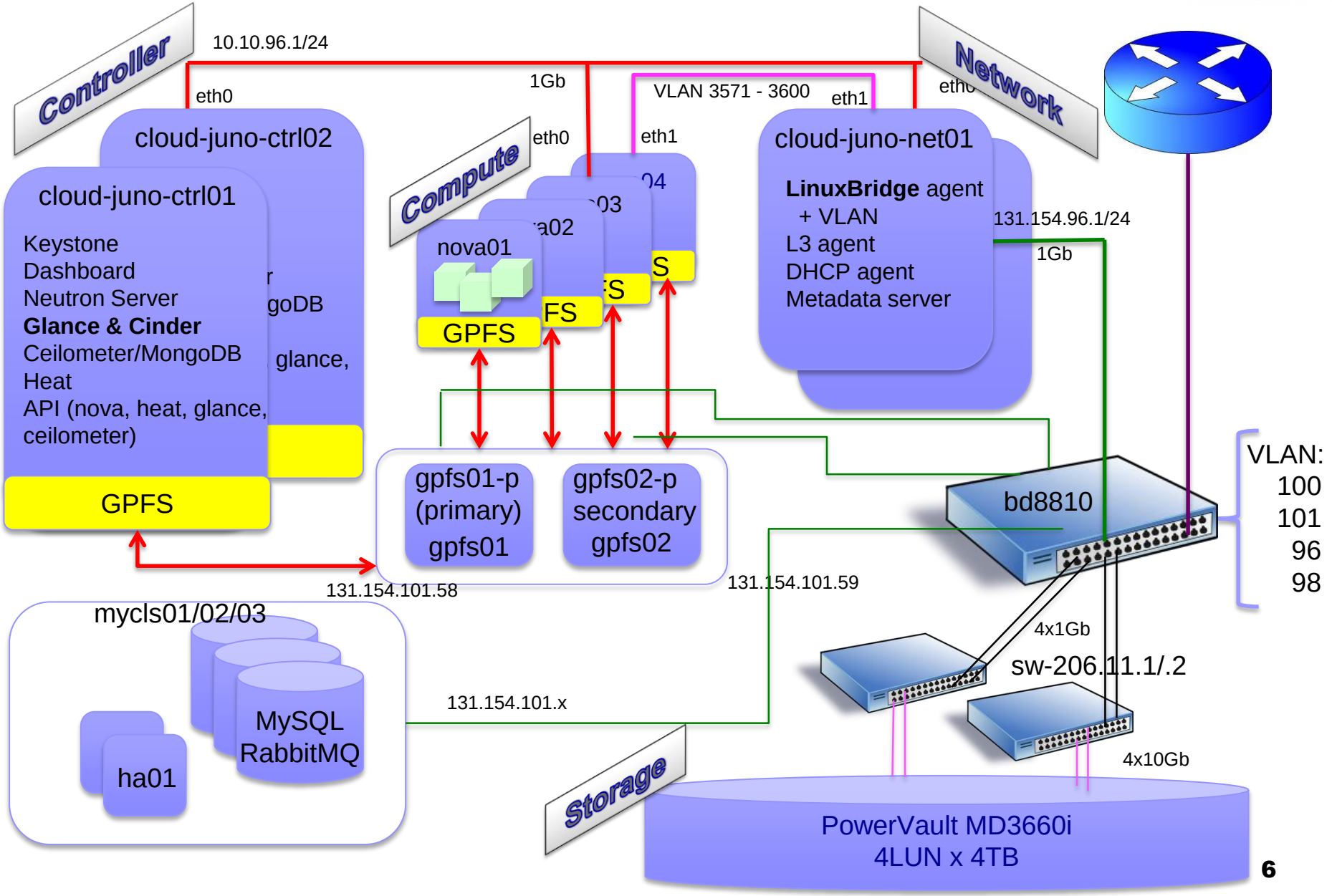
Pain point:

- Assenza di HA
- Problemi con OpenVSwitch:
  - Iniziale malfunzionamento dei security group
  - Perdita periodica di connettività delle istanze
- Assenza di SSL/TLS sugli API endpoint
- Forti limitazioni strutturali su Heat
- Il grande assente: Cinder

Destinazione: Juno

- **2 x Controller Node – HA active/active per tutti i servizi**
  - Keystone, Heat, Horizon, Ceilometer, Neutron server, Trove
  - HAProxy & Keepalived x API
  - Glance & Cinder
- **2 x Network Node – HA active/active parziale (DHCP agent in active/active, L3 agent in hot standby)**
  - Neutron con LinuxBridge + VLAN
  - 2x6 HT (24) Intel(R) Xeon(R) CPU E5-2450 0 @ 2.10GHz, 64 GB di RAM
- 4 x Compute Node => + 4 Compute-Node Havana = **128 CPU + ~500GB RAM**
  - Nova per KVM/QEMU, LinuxBridge agent
  - 2 x 8 core AMD Opteron 6320 @ 2.8GHz con 64 GB RAM
- **16TB su GPFS** per backend di Nova (istanze), Glance (immagini), Cinder (volumi)
- 1 Web Proxy per la dashboard
- 3x Percona XtraDB, RabbitMQ, MongoDB, ZooKeeper
- 2x HAProxy per Percona (failover, no roundrobin!)

Cloud@CNAF - Juno



Destinazione: Juno

Uno sguardo al futuro: Brocade VDX

- Tutte (o quasi) le funzioni di Neutron L3 agent implementate su ASIC (SNAT, floating IP…)
- Configurazione delle porte orchestrata direttamente da Neutron…
- … a patto di avere tutta la catena dal centro stella ai top-of-the-rack compatibile col plugin Brocade per Neutron

Un percorso “accidentato”

Il deployment di Juno è stato piagato da una serie di “incidenti di percorso”:

- Interazioni problematiche tra qemu-img e GPFS
- Problemi di performance di KVM su GPFS
- Errata pacchettizzazione di open-iscsi in CentOS 7 e conseguenti interazioni “problematiche” con systemd
- Difficoltà d’integrazione tra iSCSI, GPFS, systemd ed OpenStack
- Last but not least: VENOM

Incidenti di percorso: GPFS

GPFS e Glance sembrano “odiarsi” a vicenda:

- In teoria avere le immagini su GPFS permette di trasferirle mediante copia diretta sui compute node
- In realtà, spesso e volentieri le immagini risultanti sono corrotte
  - Da qemu-img convert
  - Per una strana interazione con la cache di GPFS
- **Workaround: disattivare la conversione a raw in nova-compute e vietare l’uso di immagini che non siano qcow2 e raw**

Incidenti di percorso: GPFS

Problemi di raw performance:

- Bottleneck sugli NSD
- Esposte le LUN direttamente ai client ed apportate modifiche alla configurazione del cluster su indicazioni di Vladimir Sapunenko (CNAF)
- Ottime prestazioni per I/O sequenziale
- I/O random “problematico”
- Molte ipotesi sul piatto, ma ben pochi riscontri

Incidenti di percorso: systemd

Juno è supportata su EL7 e successive, quindi addio upstart/sysvinit e “benvenuto” systemd:

- Re-training per tutti gli operatori abituati a sysvinit
- Default assolutamente inadatti alla produzione
- Notevoli difficoltà ad assicurare un corretto ordine di boot per GPFS (tuttora non del tutto risolte!)
- **Almeno in un caso, una unit con dipendenze errate fornita da CentOS ha causato un crash catastrofico**

Incidenti di percorso: systemd

Per default, systemd cattura stdout e stderr dei demoni e li redirige sul journal (e in ultima analisi su syslog):

- Ottimo in fase di configurazione per determinare errori
- Pessimo a regime – soprattutto per demoni come glance-registry che amano usare stderr per riportare backtrace inutili
- Restart di journald → SIGPIPE ai demoni che mantengono aperti stdout e stderr alla prima write
- **Workaround: reboot (sì, reboot)**

Incidenti di percorso: systemd

Zbigniew Jedrzejewski-Szmek 2014-10-12 23:44:05 UTC Comment 3

I don't think that there's a workaround, except to not restart systemd-journald. But current situation sucks, and putting workaround in many different units does not seem like a good option.

Incidenti di percorso: systemd

Default inadatti alla produzione:

- Il “runtime journal” di journald è configurato per occupare al massimo il 10% dello spazio disponibile su /run
  - Quindi il 5% della memoria fisica (tmpfs al 50% della memoria fisica)
  - Nel caso dei nostri controller node: 3GB di memoria consumati inutilmente
- **Soluzione**: RuntimeMaxUse=128M in /etc/systemd/journald.conf
  - … e relativo reboot!

Incidenti di percorso: systemd

L'integrazione con GPFS si presenta estremamente problematica:

- GPFS si avvia con uno script sysvinit
  - Ma dipende da iSCSI e multipathd, che hanno unit native
  - Che vengono considerate attive da systemd prima che le LUN siano effettivamente visibili!
- I servizi OpenStack che usano GPFS devono partire solo dopo il mount di GPFS, complicando ulteriormente le cose
- **Troppe race condition**

Incidenti di percorso: systemd

Mercoledì 13 un reboot di routine su uno dei due controller ha portato alla **completa corruzione del root filesystem**.

- Kernel in completa confusione, filesystem locali non smontati per colpa di iscsid in uninterruptible sleep
- Systemd aveva disattivato la rete prima di fare logout dalle LUN
- **Zero downtime dell'infrastruttura**
  - e convalida "accidentale" del setup HA ☺
- Bug risolto in EL7.1

Incidenti di percorso: VENOM

CVE-2015-3456, per gli amici VENOM:

- VM breakout tramite stack/heap overflow in qemu-kvm
- Abuso di un bug nel floppy disk controller virtuale
- **Non può essere disabilitato**
- Richiede root nel guest
  - ovvero la norma in un setup OpenStack
- **Soluzione: aggiornare tutti gli hypervisor a CentOS 7.1**
  - … e contestualmente aggiornare GPFS a 3.5.0.24+ o 4.1.0.7+ (anche se ufficialmente EL7.1 non è supportata da IBM)

Use case: !CHAOS

!CHAOS è “l’utente zero” di Juno:

- Applicazione fortemente cloud-oriented
- Requisiti complessi per setup automatizzato e autoscaling
- Obiettivo finale: “one click deployment” di un’infrastruttura completa (applicativo+servizi di backend) su cloud con possibilità di autoscaling in base al carico applicativo

Use case: !CHAOS

Perché Juno:

- Largo uso di Heat per deployment dei servizi di backend (Ceph per filesystem on demand, OpenVPN)
  - Impossibile utilizzare Havana (e in parte Icehouse) per problemi di Heat
  - Demo effettuata con successo al General Meeting !CHAOS del 5 Maggio
- Clustering di MongoDB supportato in Trove solo a partire da Juno
  - Work-in-progress, primi test positivi

Use case: Extreme Energy Events

Ricerca sull'origine dei raggi cosmici

42 scuole con telescopio

3 telescopi presso sezioni INFN (Bologna, Pisa, Catania)

2 telescopi al CERN

47 telescopi in totale

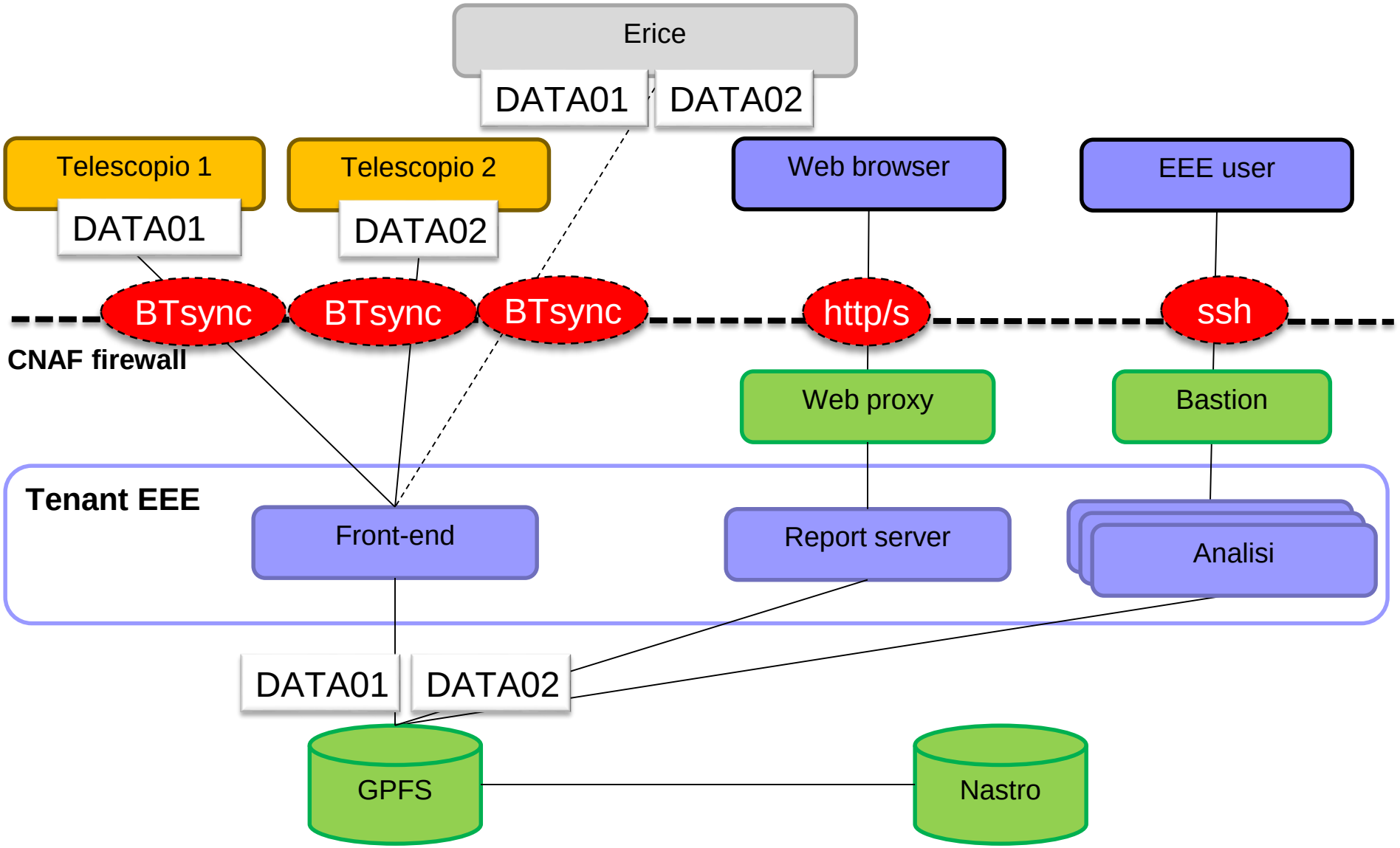
300 MB di dati raw al giorno per telescopio

55 TB in 5 anni (tra dati raw e ricostruiti)

Cloud@CNAF per EEE:

- Front-end raccolta dati provenienti dalle scuole
  - Con software di sincronizzazione BTSync usato tra la scuola e il CNAF
  - Dati su filesystem GPFS esportato via NFS alle VM
- Risorse per l'analisi dei dati con software specifico
  - On-demand
  - Per ora instanziate da un «VO manager»
- Server di reportistica - Database e web server

Infrastruttura EEE



Use case: EEE - Sviluppi futuri

- Passaggio a infrastruttura Juno
  - Utenti e VM
- Cluster Mysql con DBaaS (Trove)
  - Ora il database del monitoring è un SPOF
- Risorse di analisi on-demand per utenti EEE
  - Accesso non privilegiato alle VM
- Accesso cloud ai dati da parte degli utenti
  - Seguire gli sviluppi in Indigo WP4.2 (Cross Protocol support for storage solutions)
- Portale open data

Convalida Juno: OpenStack Rally

OpenStack Rally (ex Mirantis Rally) è un tool di testing, benchmarking e convalida per deployment OpenStack.

- "Frontend" per Tempest (test suite OpenStack)
- Scenari complessi di benchmarking e convalida
- Possibilità di stabilire un SLA da rispettare
- Ottimo per generare load sintetico sull'infrastruttura ed avere un'overview dei colli di bottiglia

Get it: https://github.com/openstack/rally/

Conclusioni

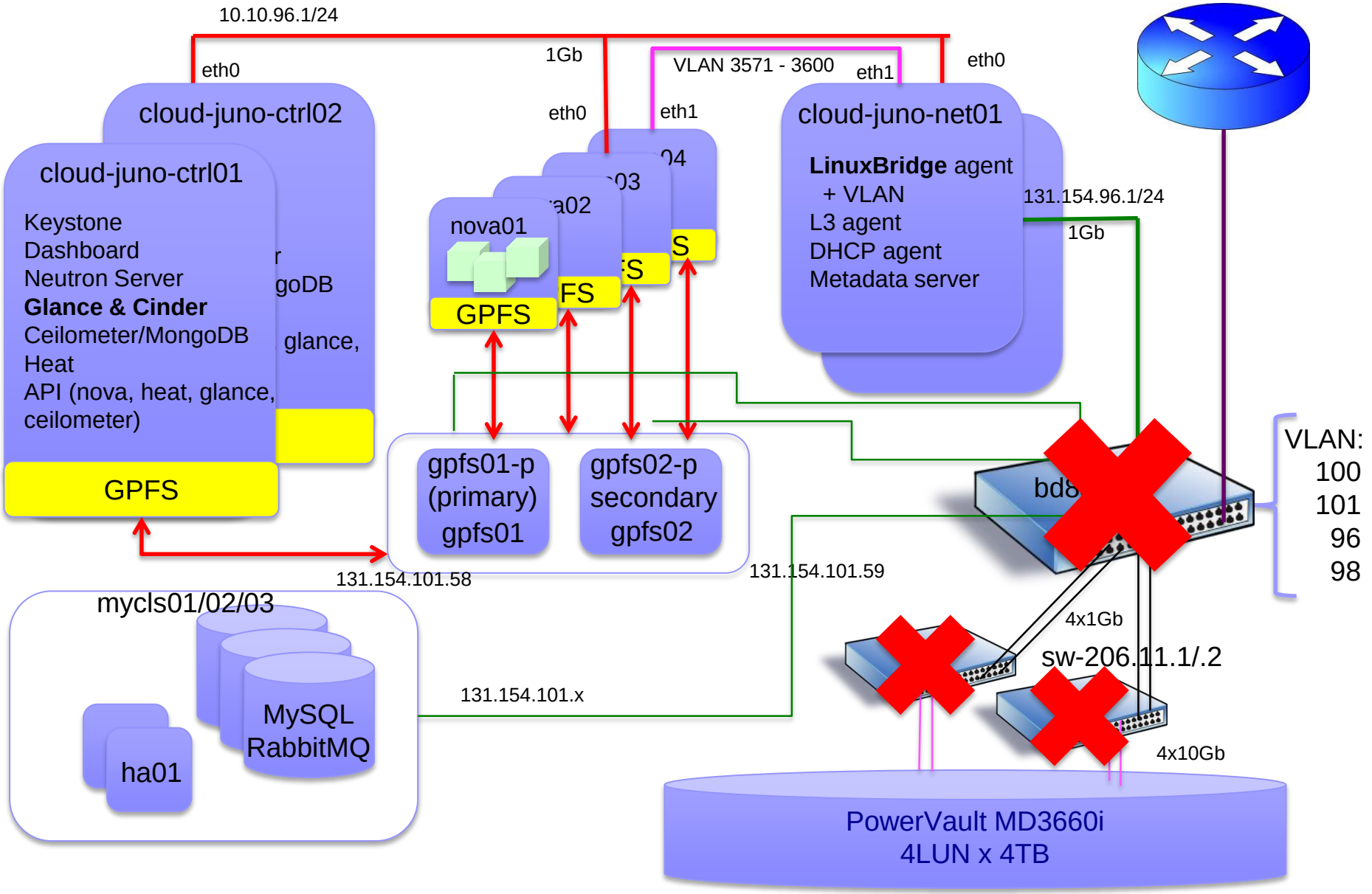
- Cloud@CNAF marcia a passo spedito verso la produzione
- Il deployment di Juno in HA non è stato eccessivamente complesso
  - Ma neanche semplice!
  - E nel lungo termine paga
- GPFS ha bisogno di ulteriori interventi
- Systemd e le sue idiosincrasie hanno causato più problemi di quanto preventivato in partenza
- Qualcuno se la sente di provare Kilo? ☺

Collaborazione

Domande

BACKUP SLIDES

Struttura Cloud@CNAF: Juno (fase 2)



Struttura Cloud@CNAF: Juno (fase 2)