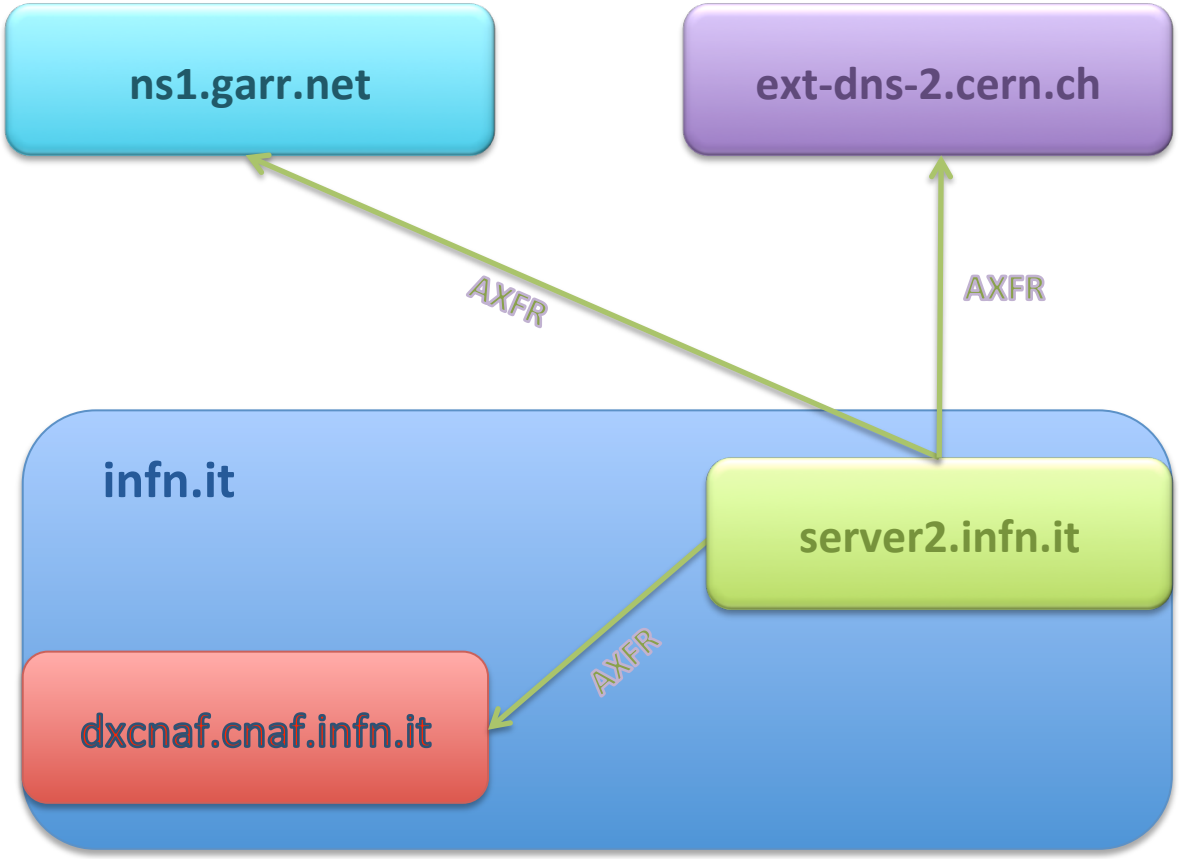
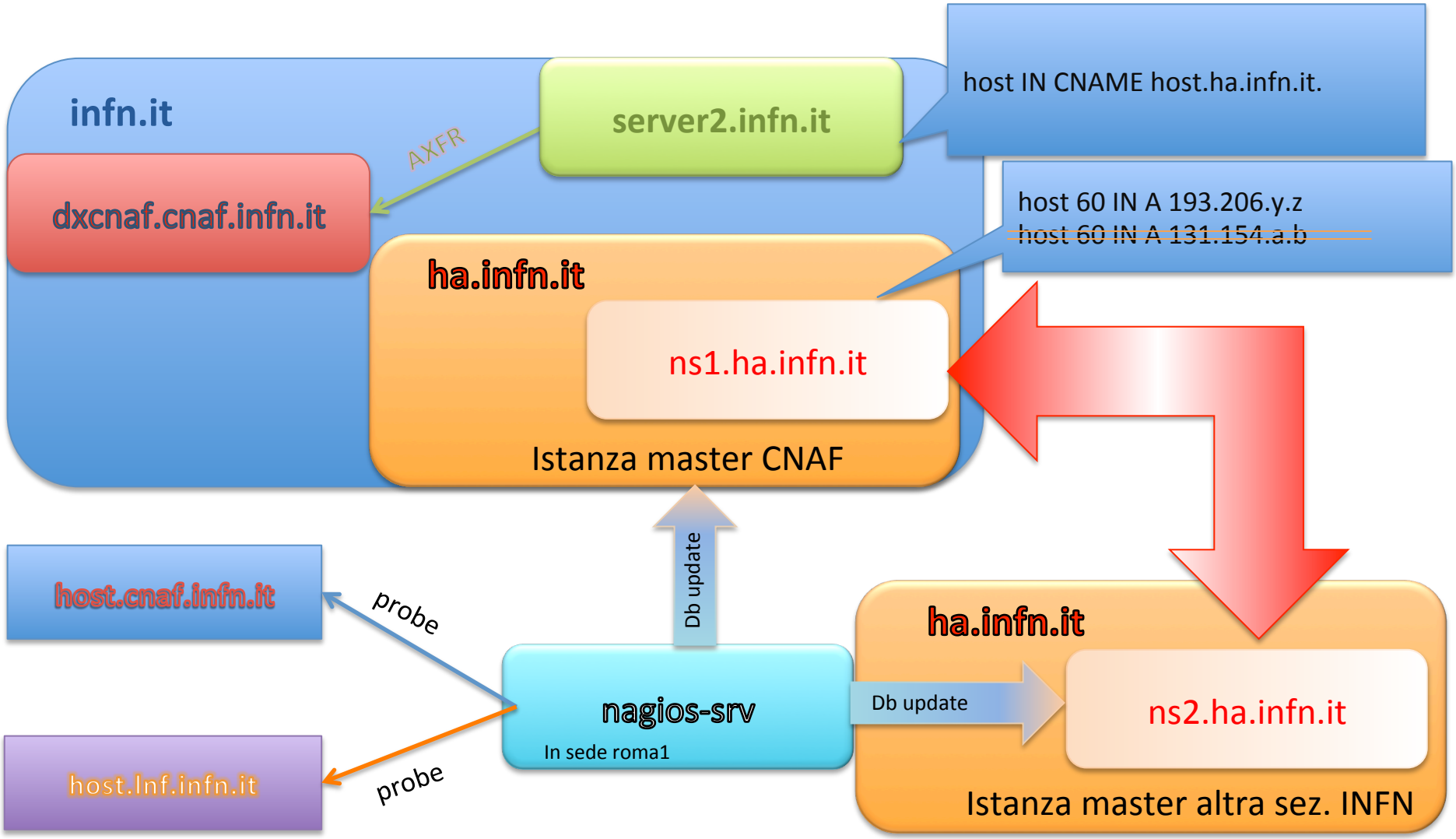
DNS HA

Stefano.Longo@cnaf.infn.it
Riccardo.Veraldi@cnaf.infn.it

DNS INFN.IT


ns1.garr.net
ext-dns-2.cern.ch
AXFR
AXFR
infn.it
server2.infn.it
AXFR
dxcnaf.cnaf.infn.it

DNS HA.INFN.IT



Come funziona

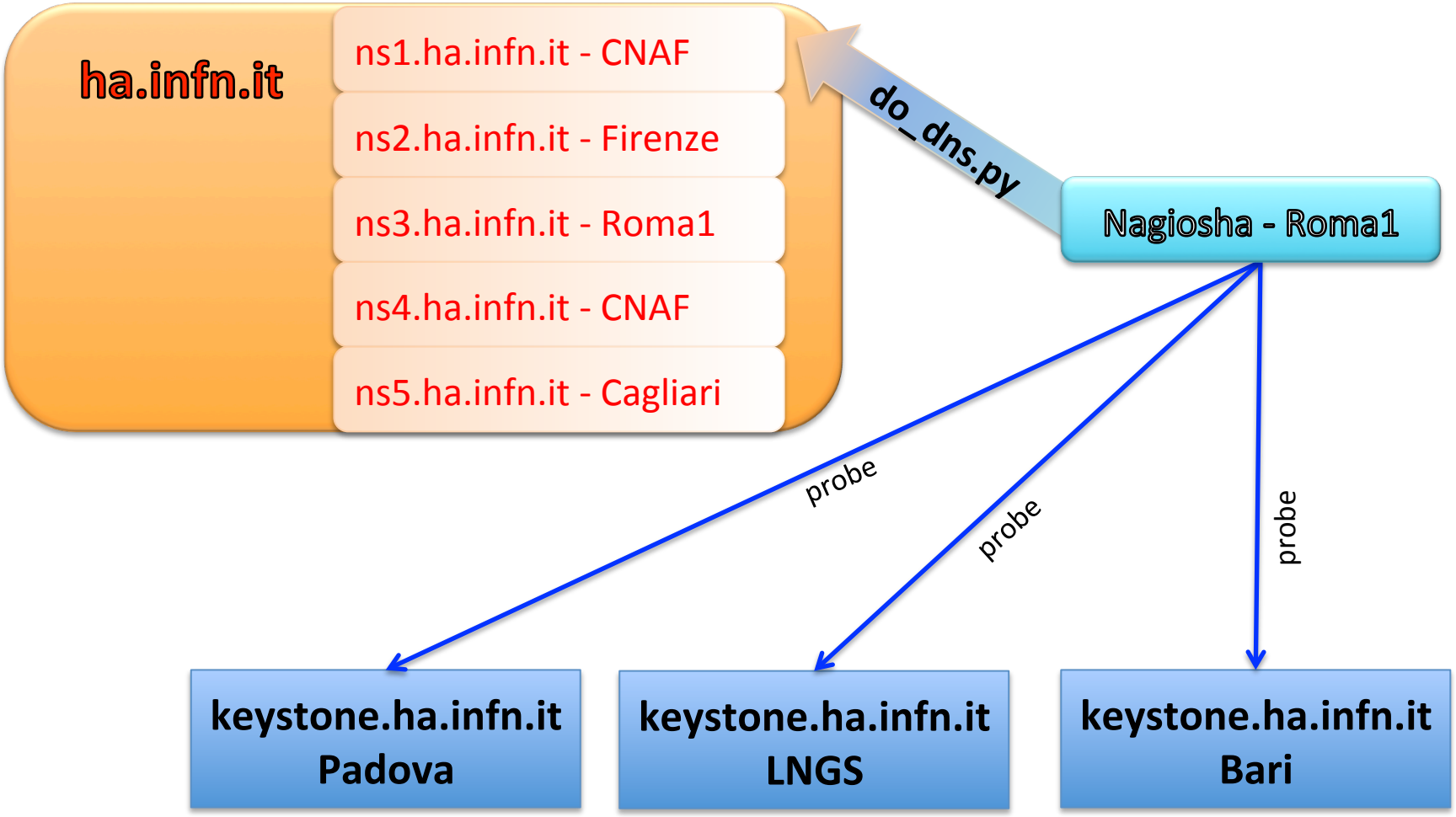
- I servizi (host.infn.it) vengono ridondati geograficamente in 2 sedi (ad es: CNAF e altra sede INFN)
- Il sottodominio ha.infn.it viene implementato con architettura Multi Master in due sedi INFN (ad es. CNAF e ROMA1)
  - **Viene impostata la delega per ha.infn.it su server2.infn.it verso ns1.ha.infn.it e ns2.ha.infn.it e gli host name dei servizi HA vengono definiti su server2.infn.it come CNAME che puntano a hostname sul dominio ha.infn.it**
- Gli hostname definiti su ns1.ha.infn.it e ns2.ha.infn.it puntano all'IP di un'istanza del servizio con TTL 60 in una delle due sedi in cui è installato
- Nella sede di ROMA1 è presente un nagios server che fa probe verso i server su cui è implementato un determinato servizio ridondato geograficamente
  - Se il server principale tra i due non risponde nagios fa partire un'opportuna procedura (db update) che modifica l'IP del servizio su ns1.ha.infn.it oppure su ns2.cnaf.infn.it se il primo non è raggiungibile
  - Lo script di aggiornamento dei record DNS automaticamente contatta il nameserver che risponde per primo

- **Attraverso il CNAME definito su server2.infn.it il servizio sarà sempre raggiungibile nella sede in cui è UP and running**

Architettura Multi-Master

- Tutti i nameserver sono autoritativi e paritetici per la zona ha.infn.it
  - ns1.ha.infn.it
  - ns2.ha.infn.it
- La modifica su un nameserver viene propagata automaticamente sull'altro
- Se uno dei due nameserver non è raggiungibile si possono continuare a operare modifiche su quello disponibile e quando il secondo tornerà UP automaticamente si sincronizzera' con il suo peer

Situazione in produzione



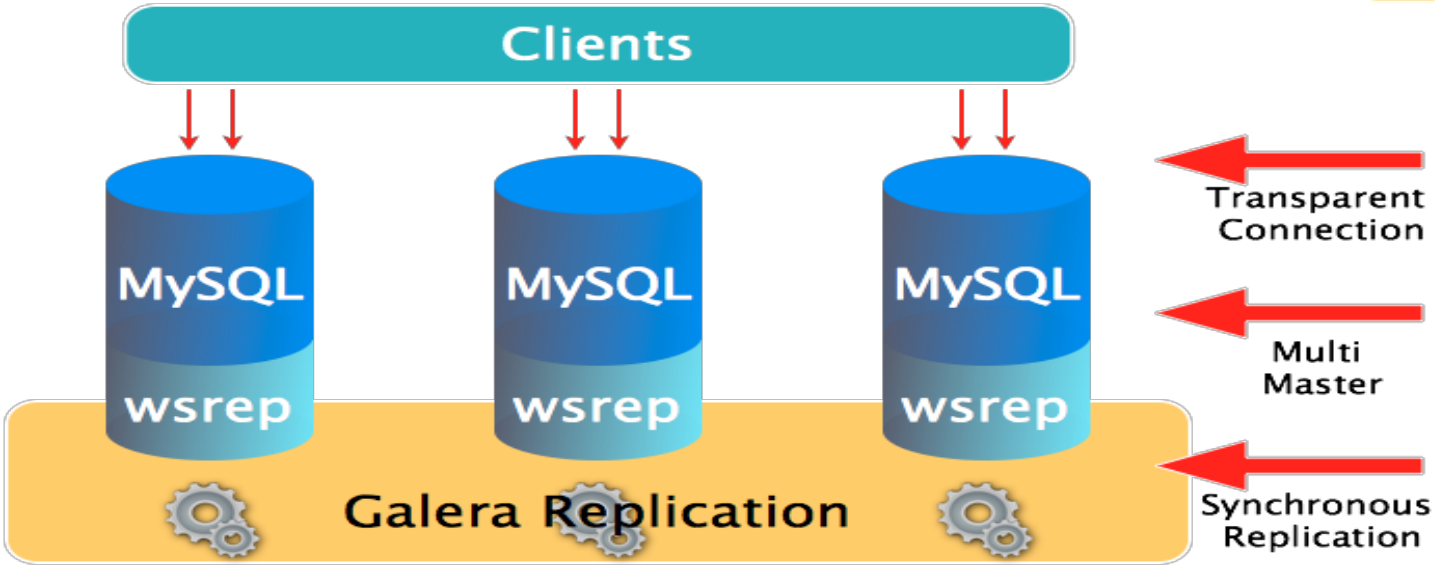
Evoluzione e Problemi nella roadmap

- (2011) bind con backend mysql scritta da Riccardo
  - Funzionante ma poco mantenibile
    - Modifica alla patch per ogni nuova versione di mysql
      - Testare il tutto ogni volta
      - Produre le rpm specifiche
  - Non prevista HA del database
- (2011 – 2012)bind-dlz con MySQL backend
  - Caratteristiche multi-master date dalla configurazione specifica di Mysql
    - Sistema funzionante anche con un solo nodo mysql UP
    - Configurazione complessa
    - Difficoltà ad aggiungere o modificare l'IP di un nodo del cluster
    - Scalabilità a più di 2-3 nodi difficilmente realizzabile a causa della complessità (facilità di commettereerrori ecc.)
- (2013) Crash improvviso di bind nell'uso delle librerie mysql

Realizzazione definitiva (2014)

- Utilizzo di bind-dlz con MySQL backend e cluster GALERA
  - Soluzione Cluster MySQL ben consolidata
  - Sistema da pochissimo in produzione ma sembra sufficentemente robusto
  - Semplificazione della complessità di avere più nodi MySQL multi-master
  - Fail della rete significa problemi al cluster
    - Abbiamo aumentato il numero di nodi geografici

GALERA Cluster



Write**S**et **R**eplication (wsrep) definisce un insieme di API per la replicazione dei dati di un'applicazione. In un cluster con wsrep

- I client modificano lo stato di un'applicazione (ad esempio un DBMS)
- Variazioni di stato sono rappresentate da una serie di operazioni atomiche.
- I nodi del cluster mantengono la sincronizzazione replicando le variazioni atomiche nello stesso ordine di esecuzione

GALERA è un'implementazione multi-master di wsrep per MySQL

GALERA Cluster

GALERA associa ad ogni nodo un **GTID** (Global Transaction ID) che identifica lo stato del nodo (UUID), la sequenza di modifiche atomiche in corso e la posizione corrente nella sequenza**.**

Un cluster basato su GALERA

- Garantisce l'esecuzione delle transazioni su ogni nodo
- E' multi-master (modifiche anche sullo stesso oggetto ammesse da nodi diversi)
- Gestisce i conflitti tra writesets, permettendone l'applicazione parallela
- Elimina single point of failures

GALERA Cluster

Alcune caratteristiche:

- Ogni nodo monitora lo stato degli altri nodi mediante messaggi keepalive
- L'algoritmo di quorum determina il *Primary Component*, l'insieme di nodi che può operare modifiche allo stato del cluster.
- Per il DNS-HA sono in esecuzione 5 istanze, garantendo l'operatività con una primary component minima di 3 server
- In assenza di una primary component si verifica una condizione di split-brain: il cluster diviene non operativo
- I nodi della primary component sono paritetici: è possibile operare indifferentemente su ognuno di essi.

ToDo

- A breve inserimento nel DNS HA della risoluzione per i ridirettori di CMS
- Implementazione della risoluzione per i servizi centralizzati di AAI
- Inserimento dei record in HA per alcuni servizi del sistema informativo

Ringraziamenti

- Cristina Bulfon
  - Per il lavoro svolto a Roma1 a supporto del progetto DNS HA, installazione macchine e risoluzione problemi con il BIOS
- Daniele Gregori del CNAF per il kick-off su nagios
- Leandro Lanzi
  - Per avere messo a disposizione gli hypervisor della sez. di firenze
- Antonio Silvestri
  - Per avere dato piena disponibilità a inserire nel cluster VMware l'istanza DNS HA con considerevole lavoro di conversione di immagini delle VM
- Stefano Stalio
  - Per avere collaborato attivamente al primo use case di DNS HA ovvero il servizio Cloud di autenticazione Nazionale KEYSTONE e per avere portato pazienza quando le cose inizialmente non erano ancora a pieno regime di funzionamento