

Distributed VoIP Platforms using OpenSIPS

Vlad Paiu
OpenSIPS Project Developer
OpenSIPS Solutions

Single SIP Server Instance

- **Optionally with Redundancy**
- **Limited capabilities**
 - **Number of Subscribers**
 - **Number of Calls per second**

Geographical (coverage, QoS)

- Cover the map
- Best quality everywhere

Load (balancing, scaling)

- Millions of concurrent calls
- Hundreds of millions of subscribers

Redundancy (HA)

- Having a backup is a good idea
- For the 5 of nines you need more than 100%

- **About OpenSIPS**
- **Distributed VoIP Architecture**
- **Sharing Information**
- **Hot Redundancy**
- **Conclusions**

Open

- GPL, Open Source project
- tens of contributors, community of tens of thousands
- used from SMB to enterprises and grade-carriers

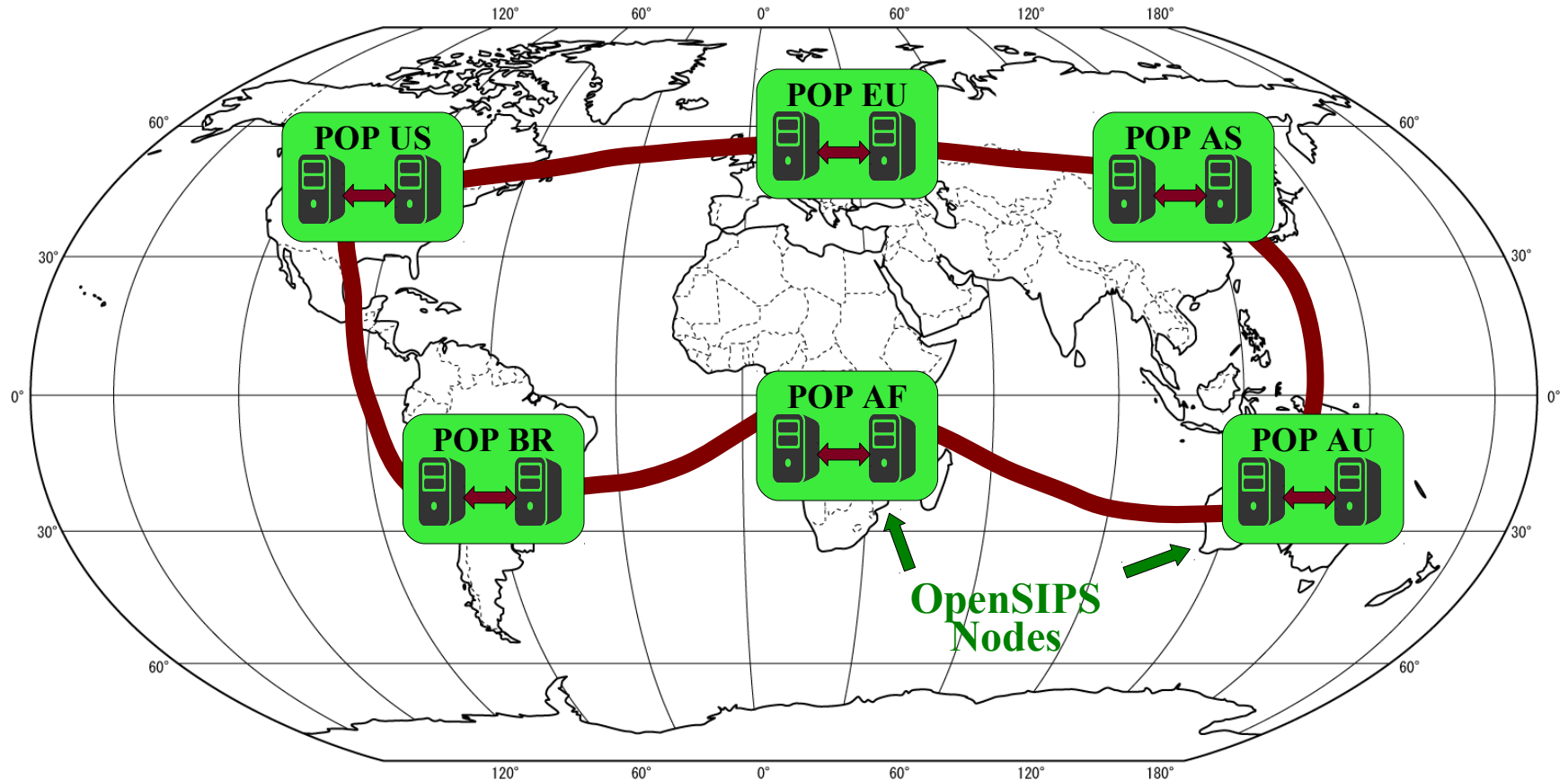
SIP

- SIP RFC 3261 + tens of SIP extensions
- SBC, trunking, billing, ITSP, router, call center

S

- Server (registrar, proxy, LB, B2BUA, SIMPLE, NAT, apps)
- 20000 cps, millions of parallel calls and subscribers
- Programmable and flexible (scripting with > 100 modules)

OpenSIPS builds and glues SIP infrastructures.



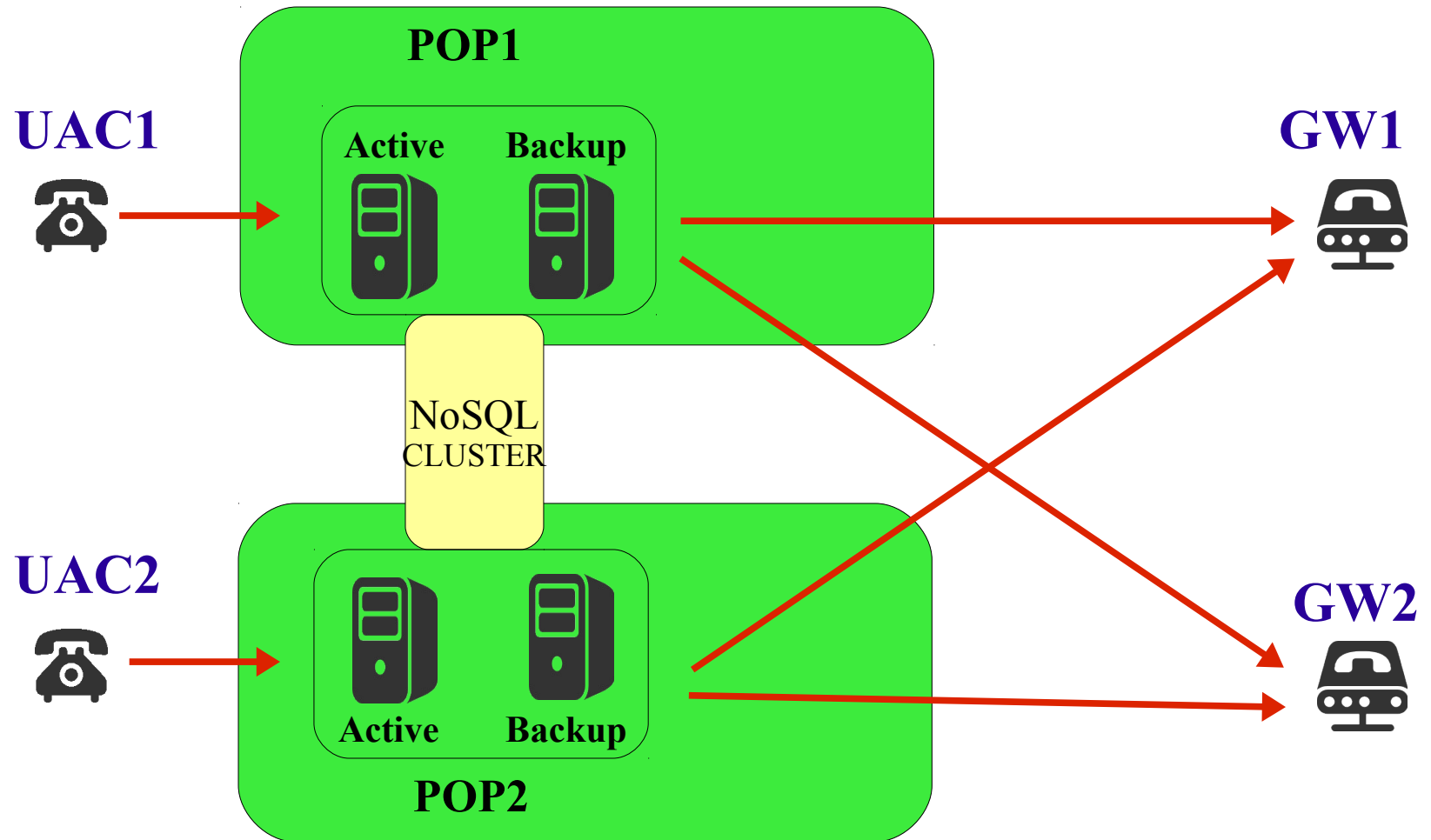


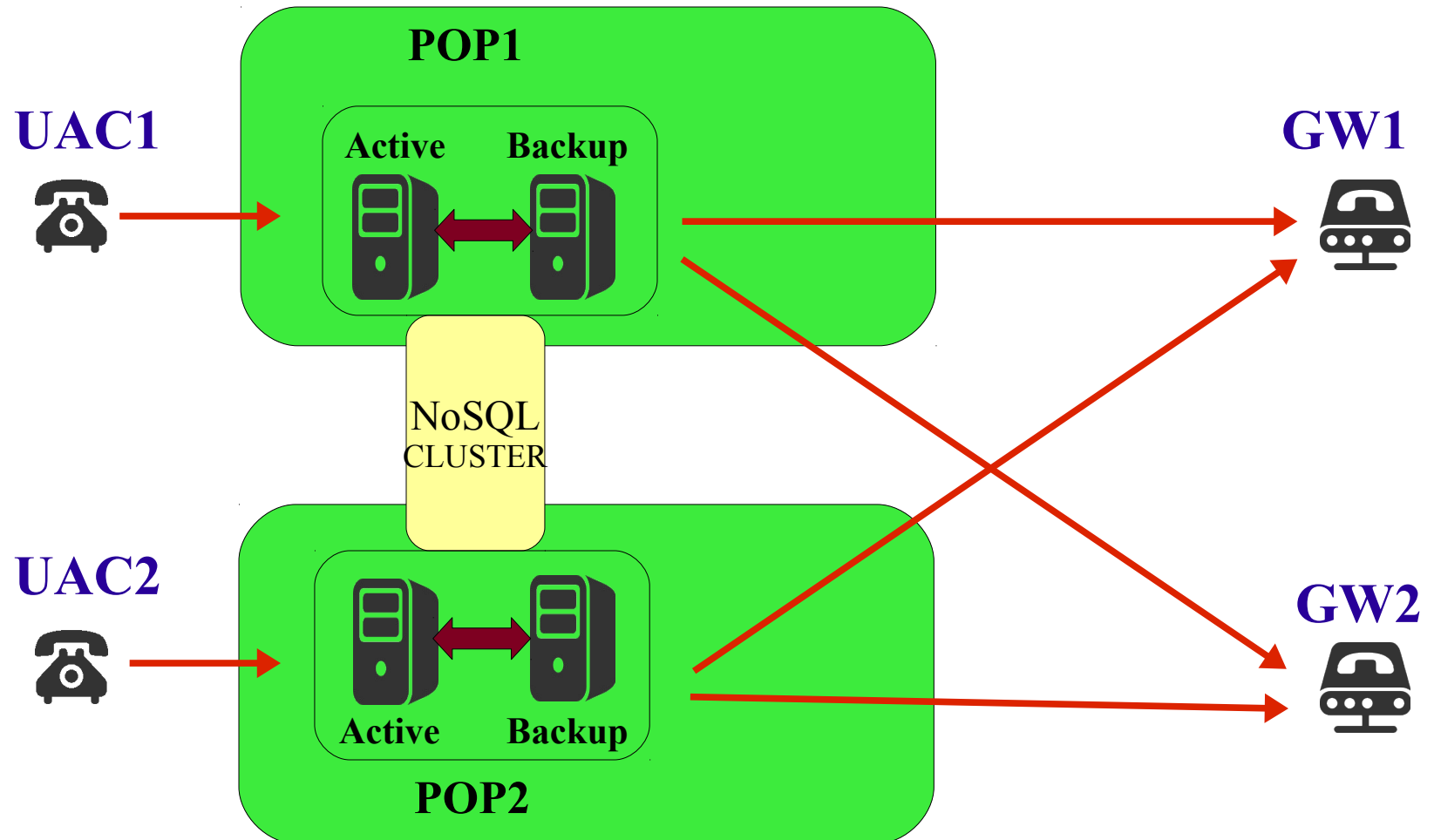
Multiple OpenSIPS instances located in geo-distributed POPs that act as a whole by **sharing internal data** (calls, registration, counters, statistics) or runtime external data (limits, credits, caches)



NoSQL DB support to **link everything together** -
modules using the NoSQL interface to communicate
and share

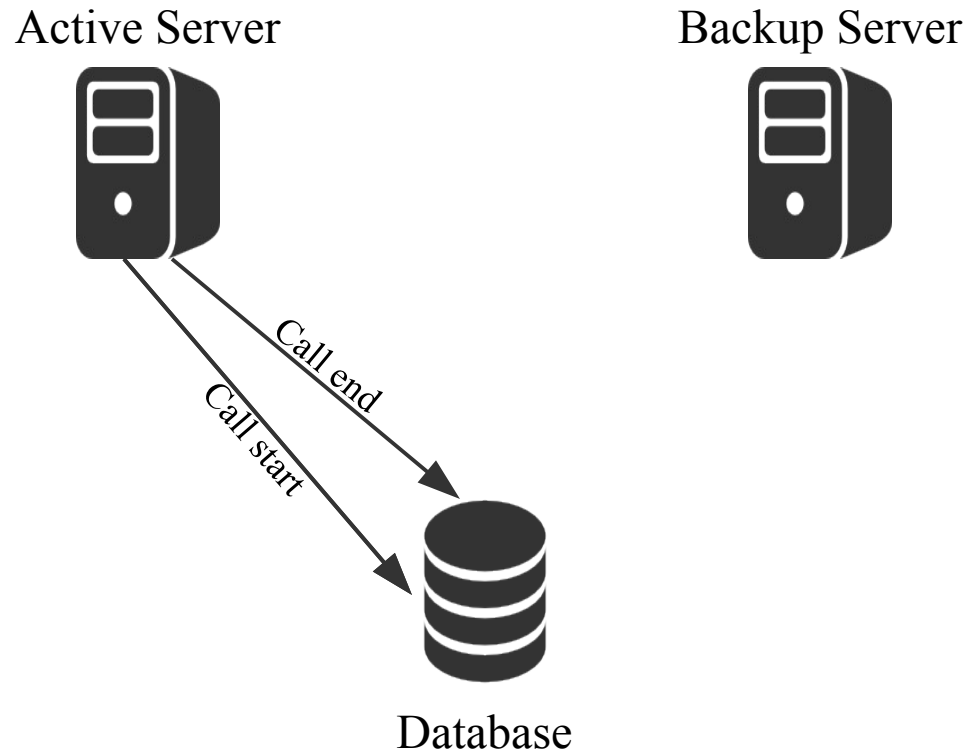
- Supported NoSQL Back-ends
 - Redis
 - Memcached
 - Cassandra
 - MongoDB
 - CouchBase
- Uniform access to NoSQL backends
 - Simple API like `fetch_key()`, `remove_key()`, etc
- Raw back-end specific queries
- SQL to NoSQL convertor
 - Only for MongoDB



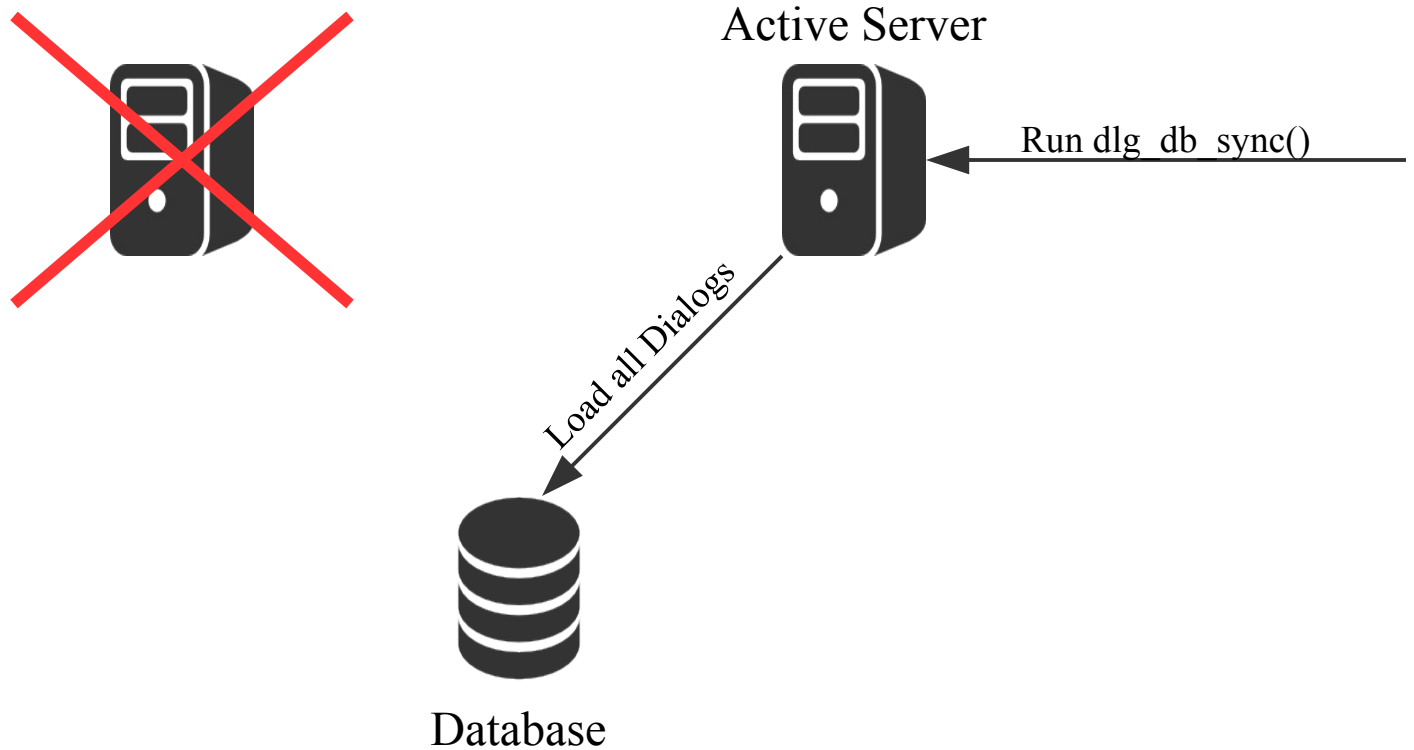


- **We need(ed) a DB to store dialogs**
 - **Persistency**
 - **Failover**

OpenSIPS 1.8 failover solution



OpenSIPS 1.8 failover solution

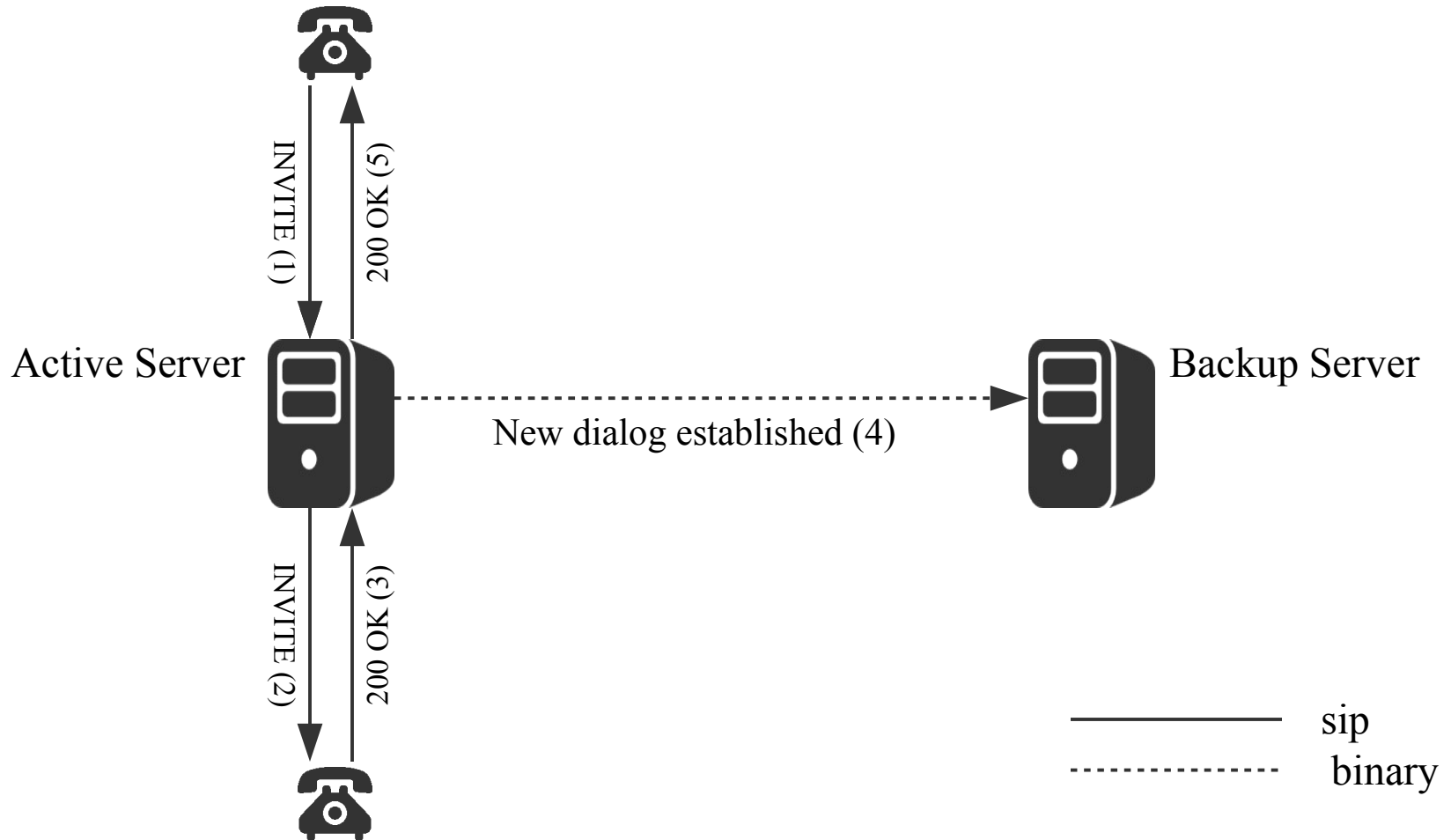


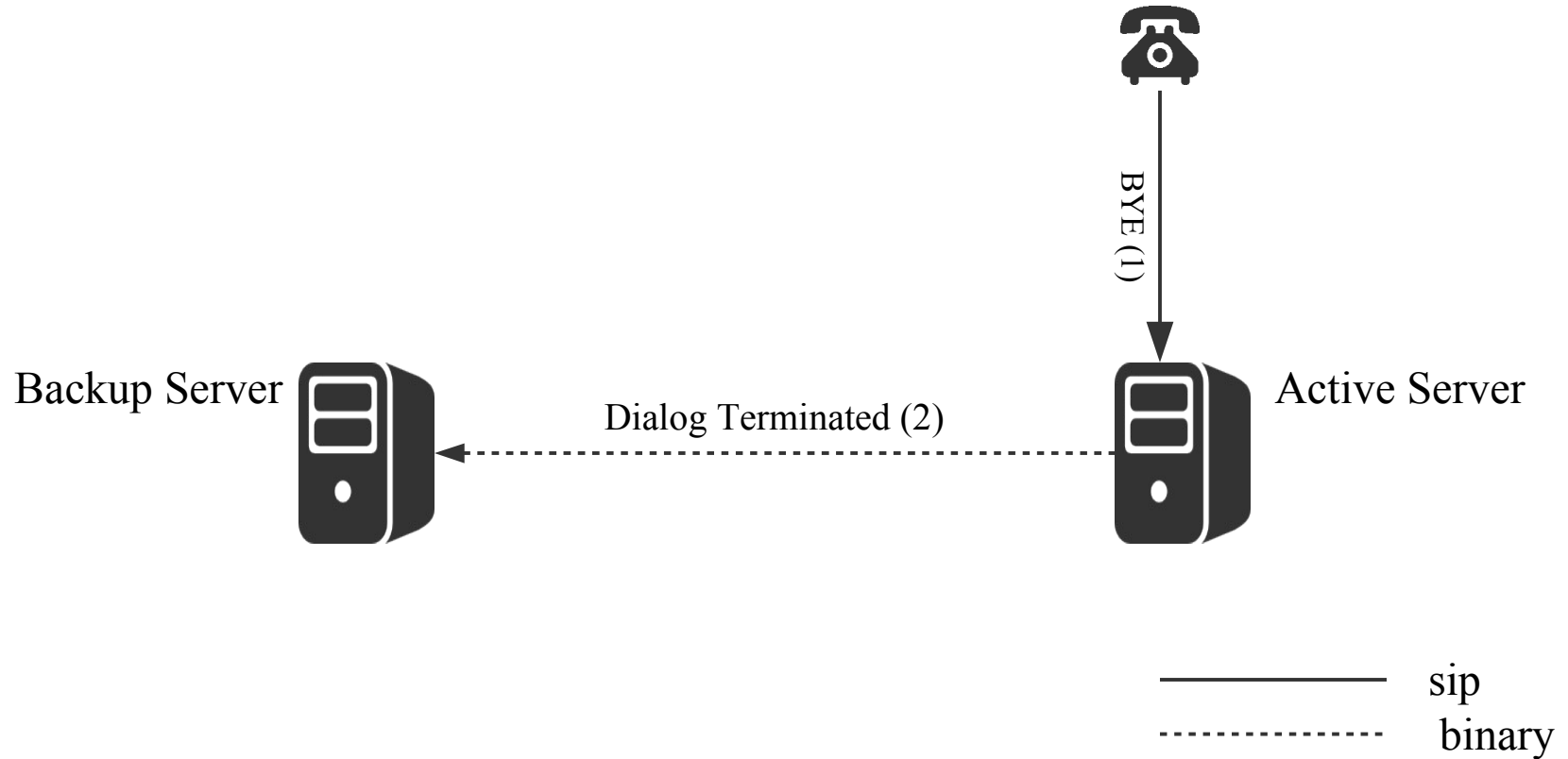
- **What If we have 1 Million concurrent calls and our server fails?**

- The backup will kick in , issue `dlg_db_sync` and try to load 1 Million rows from the DB...
- It will take a LONG time
- By the time we've loaded everything, most likely those dialogs would have already ended

- **For large deployments, we NEED real-time replication**
 - For fast fail-over

- **Fast and Efficient communication channel between OpenSIPS instances**
- **To be used for real-time data replication**
 - Dialog state
 - Registrations
 - Transactions
- **Introduced in OpenSIPS 1.10**





- **bin_listen = 10.0.0.150:5062**
- **bin_children = 5**

- **modparam("dialog", "accept_replicated_dialogs", 1)**

- **modparam("dialog", "replicate_dialogs_to", "10.0.0.150:5062")**
- **modparam("dialog", "replicate_dialogs_to", "192.168.2.129:5060")**

- **Backup servers are kept in perfect sync**
- **The dialog module can now be massively scalable, no longer being limited by the DB back-end**
- **Can be used to replicate any real-time data between different OpenSIPS instances**

- **OpenSIPS is a great choice if you want**
 - High performance on a single box
 - Strong geo-distribution capabilities

- **Ideal for high traffic deployment types**
 - SBCs
 - Load Balancers
 - Trunking

Thank you for your attention
You can find out more at www.opensips.org
vladpau@opensips.org

Questions are welcome