

SIP high availability

Motivation

„VoIP telephony needs to be reliable“

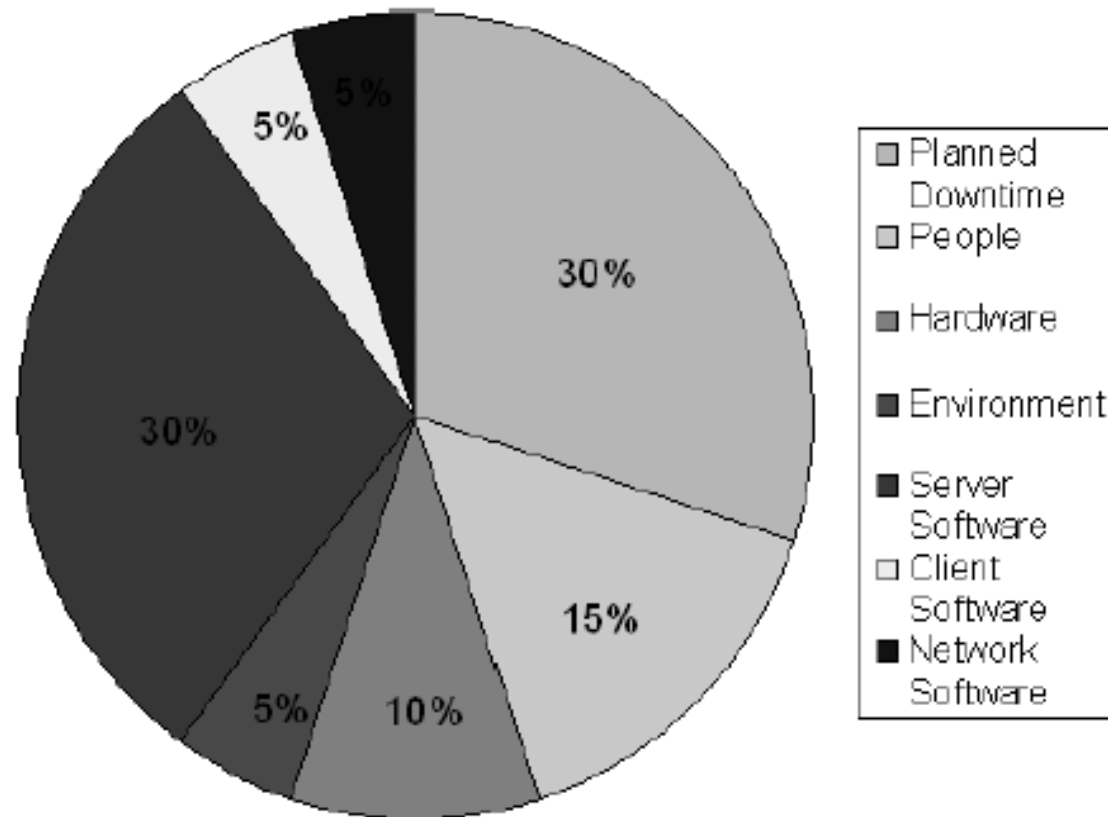
It is not acceptable for a telephony service to be unavailable for a longer period of time.

98% availability means $0.02 \times 24\text{h} = 28.8 \text{ min. outage per day !!}$
This is not „reliable“

An availability of 99.99% would only mean $0.0001 \times 24\text{h} = 8.64\text{s}$ outage per day, which is acceptable.

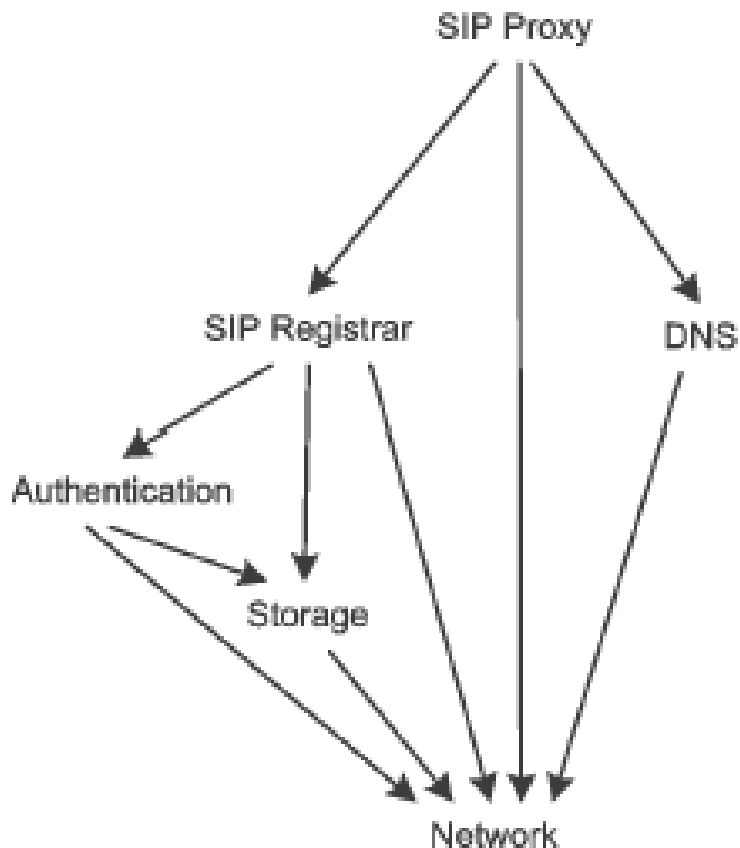
SIP high availability

Reasons for service unavailability



SIP high availability

Dependencies of SIP components



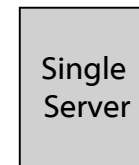
Example: REGISTER

- Do DNS lookups
- Forward to Registrar
- Do authentication
- Store contact in DB

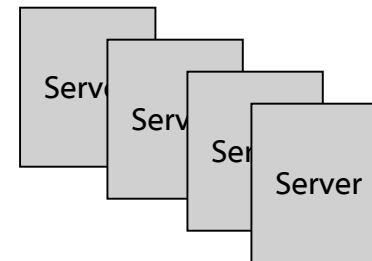
All components must be available.

SIP high availability

- Reliability by redundancy
 - redundant components
 - host redundancy
 - cluster
- load balancing
- protection against
 - software failure
 - attacks
 - misconfiguration
 - power outage
 - natural catastrophes

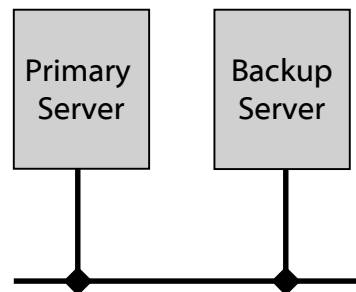


VS.



SIP high availability

- If a single system has a failure probability of 1%, using two redundant systems gives a probability of only 0.01% that they will fail both. Three systems would have 0.0001% failure probability.
- By increasing the number of redundant systems, every desired availability probability $< 100\%$ is achievable.
- master/primary server is the active instance
- backup servers are running but stay inactive (hot stand-by)



SIP high availability

How to realize service take over ?

- Static configuration
 - Pre-configure each UA with the list of redundant servers.
 - Problems:
 - 1) Support in the UAs is required.
 - 2) UAs outside the providers domain would not be able to use this.
- DNS Update
 - on server failure, change the address associated with the symbolic name in the DNS servers database.
 - Problem: UAs must not store their DNS lookup results.
- Multiple A records
 - The DNS server responds to lookup requests in a round robin fashion over the A records.
 - Problem: Support in the UAs is required: If a server does not respond, do the lookup again to obtain another server.

SIP high availability

How to realize service take over ?

- SRV records
 - on SRV lookup, distribute a prioritized list of servers to contact.
 - covered by RFC 3261 & 3263.
 - Problem: Correct implementation of SRV lookup behaviour in the UAs is nearly always missing.

```
$ORIGIN example.com.  
@ SOA server.example.com. root.example.com. ( 1995032001 3600 3600 604800 86400 )  
_sip._udp. SRV 1 0 5060 primary.example.com.  
           SRV 2 0 5060 bk1.example.com.  
           SRV 2 0 5060 bk2.example.com.  
primary   A 172.30.79.11  
bk1       A 172.30.79.13  
bk2       A 172.30.79.15
```

SIP high availability

All previous methods are client sided and have severe problems.

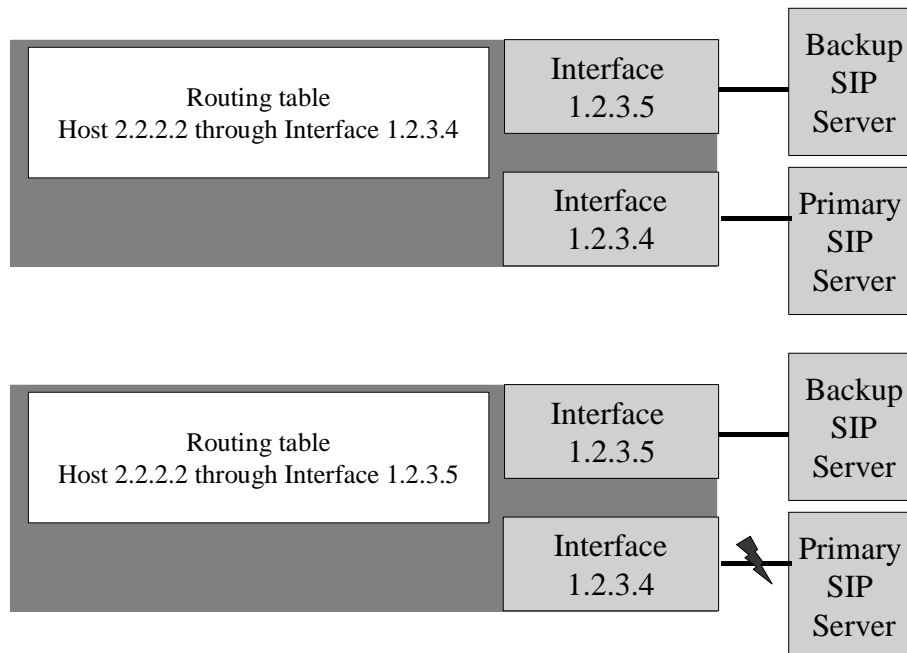
Better approach:

- Hide the HA architecture from the clients
- Let the servers transparently take over
- Transfer of IP address (IP take over)

SIP high availability

How to realize service take over ?

- IP take over by routing table modification



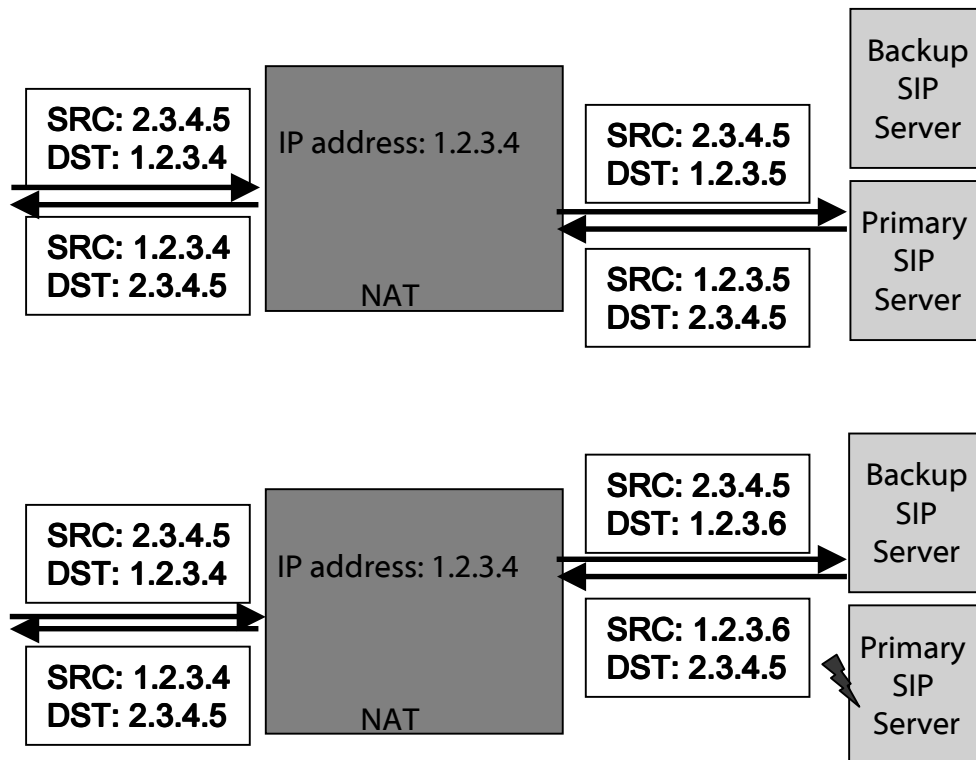
- Both SIP servers have the same IP address.
- Only the active SIP server gets traffic from the router.
- The router must not forward traffic from the backup system -> IP collision
- In case of take over, the backup SIP server changes the routing table to get the traffic.

SIP high availability

How to realize service take over ?

- IP take over by NAT

- Inbound: change destination (DNAT)
- Outbound: change source (SNAT)



- Both SIP servers have different IP addresses.
- The NAT box changes dest. address and forwards to the active server.
- The NAT box must not forward traffic from the backup system -> misrouting on return path.
- In case of take over, the backup SIP server changes the DNAT relation to get the traffic.

SIP high availability

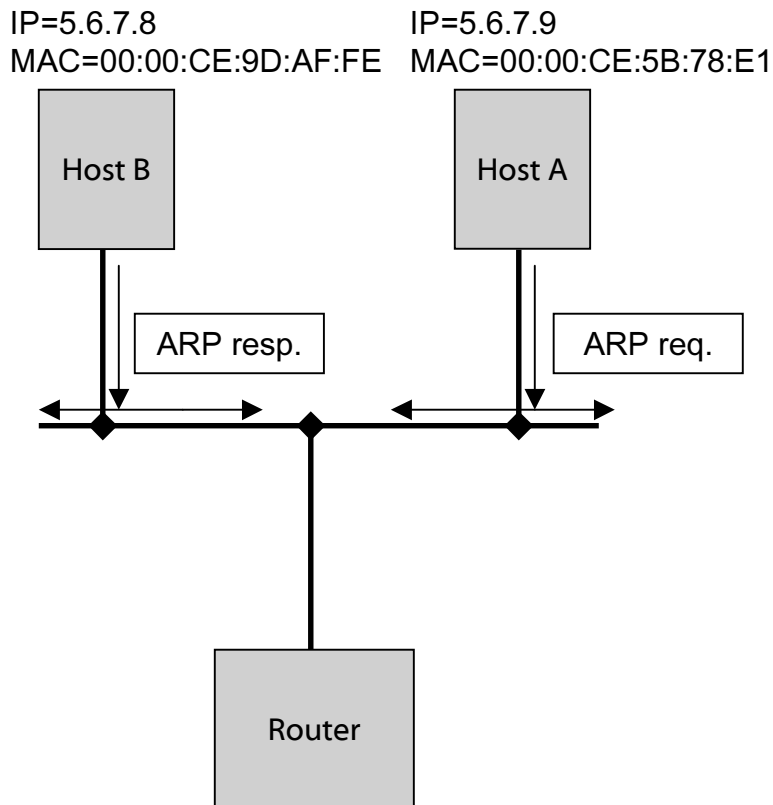
How to realize service take over ?

- VRRP (virtual router redundancy protocol) RFC 3768
 - election protocol to dynamically assign responsibility to a router in a LAN.
 - dynamic failover in case of master unavailability.
 - uses IP multicast for communication
 - enables/disables virtual IP address

SIP high availability

How to realize service take over ?

- VRRP : understanding ARP (address resolution protocol)



ARP is used to resolve IP addresses to MAC addresses in local network.

Each host has a table (ARP cache) with entries of the form <MAC-address> : <IP-address>

If a host wants to send to eg. 5.6.7.8, it broadcasts for its MAC address („Who has 5.6.7.8“)

Responses are then stored in the ARP cache.

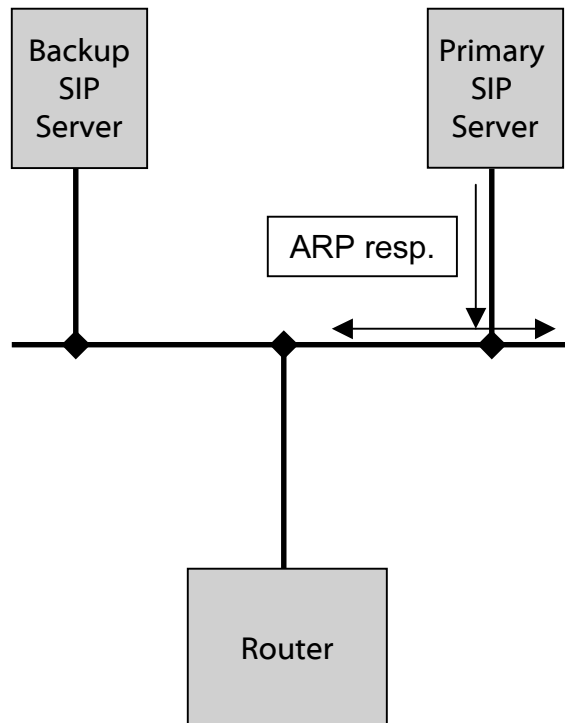
Responses are even stored if the receiver of a response did not ask for the mapping.

SIP high availability

How to realize service take over ?

- VRRP normal operation

Virtual IP=1.2.3.4	Virtual IP=1.2.3.4
Real IP=5.6.7.8	Real IP=5.6.7.9
Virt. MAC=00:00:CE:5B:78:E1	Virt. MAC=00:00:CE:5B:78:E1
MAC=00:00:CE:9D:AF:FE	MAC=00:00:CE:A4:13:5C



- Both servers have the same virtual IP, but different real IP addresses. Thus they can have two IP addresses on the same Interface.

Active server: issues ARP responses for both of its IP addresses. Sends VRRP heartbeats to a MC group.

Backup servers: issues ARP responses for its real IP addresses.

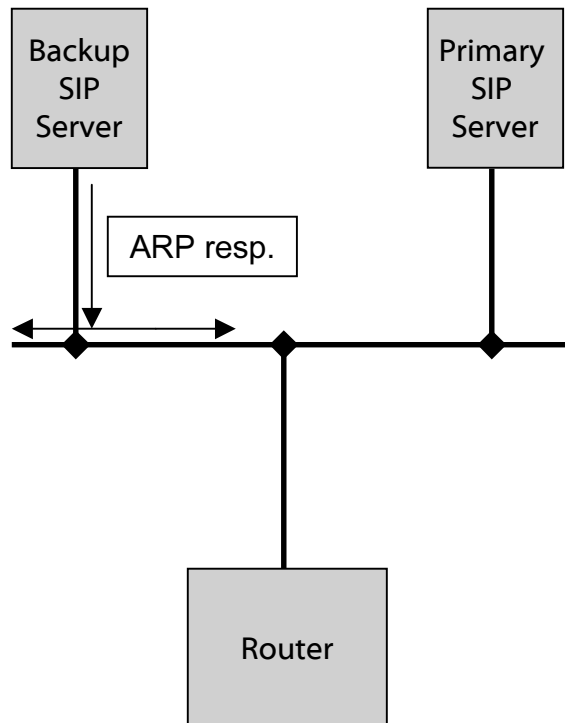
- All redundant servers are subscribed to MC group.

SIP high availability

How to realize service take over ?

- VRRP take over

Virtual IP=1.2.3.4	Virtual IP=1.2.3.4
Real IP=5.6.7.8	Real IP=5.6.7.9
Virt. MAC=00:00:CE:5B:78:E1	Virt. MAC=00:00:CE:5B:78:E1
MAC=00:00:CE:9D:AF:FE	MAC=00:00:CE:A4:13:5C



- The backup server notices the failure of the master, if it does not receive VRRP heartbeats for a certain time.
- The backup server starts to issue gratuitous ARP requests to update the switches in the LAN, so that the virt. MAC address has “moved” in the LAN.

SIP high availability

Problems when server take over appears

The backup server must decide whether the master is still working or not.

- VRRP heartbeats
- SIP OPTION requests („SIP ping“)

This decision can be wrong !

Every failover mechanism must make sure that a „dead“ master does not receive any further messages.

- VRRP: Switch tables changed
- DNS: address association changed. Unsafe if UA is caching.
- Routing table/NAT: guaranteed by the router/NAT box
- Static config: not guaranteed, UAs will try to contact the dead server.

SIP servers are stateful. Thus it is a problem if the backup server does not have the same state at the time point of failure as the master.

Thus server knowledge must be replicated in real time.

SIP high availability

Summary of takeover methods

	Router	NAT	VRRP
Same IP Address	Yes	No	No
Virtual Addresses	No	No	Yes
Packet Address Change	No	Yes	No
ARP involved	No	No	Yes
3rd Party State	Yes	Yes	No
MAC Layer involved	No	No	Yes
Backup Server utilization	No	No	No

SIP high availability

Replication

- Client based replication
- Server based replication

SIP high availability

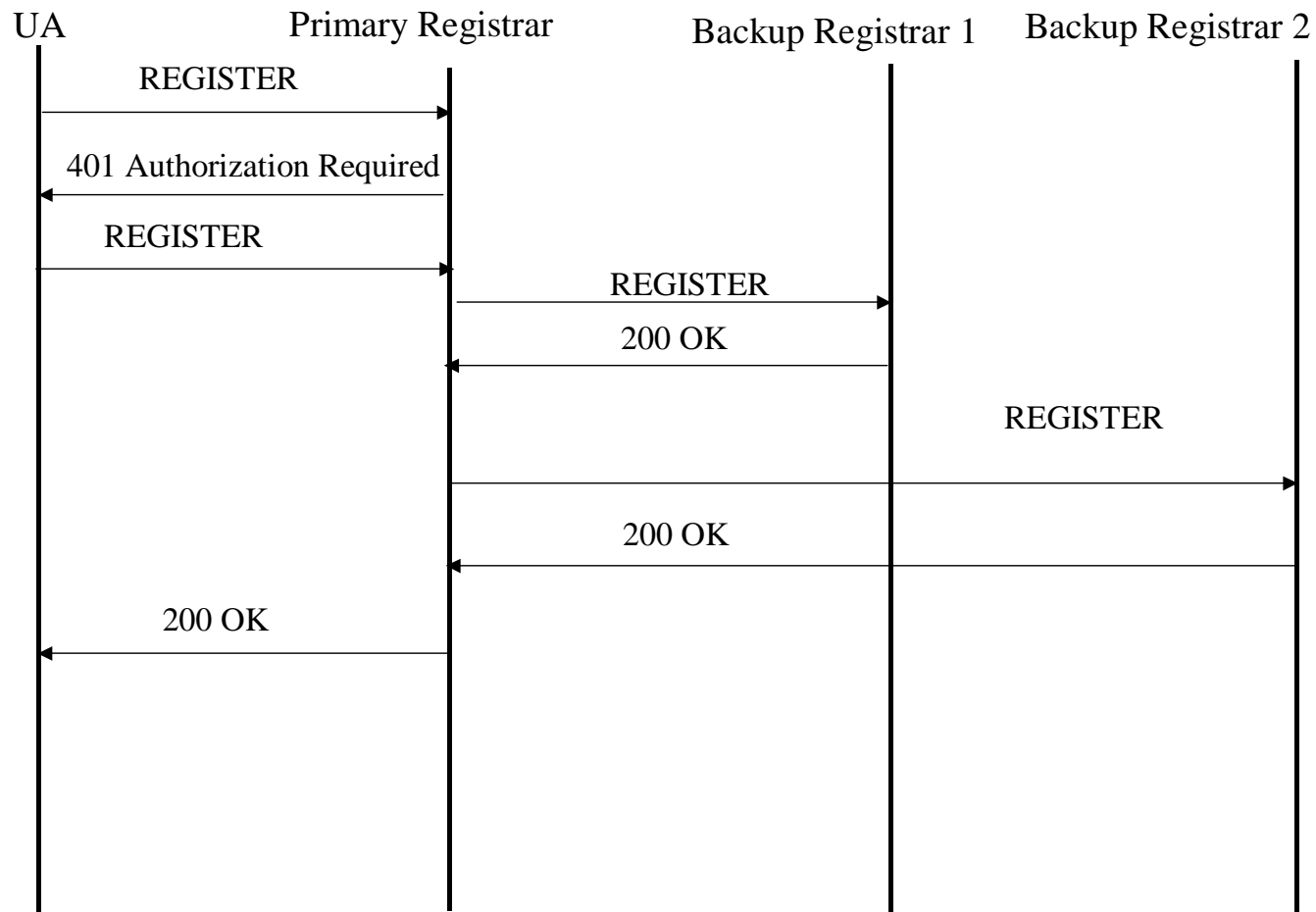
- Client based replication

Problems:

- All backup servers must be reachable by the Client. This is a contradiction to all the introduced failover mechanisms except for static configuration.
- If a server does not respond, the client will have to re-transmit messages, finally giving up. In case of just a temporary outage, this results in an inconsistency in server knowledge.
- This concept violates the concept of high-availability being transparent.

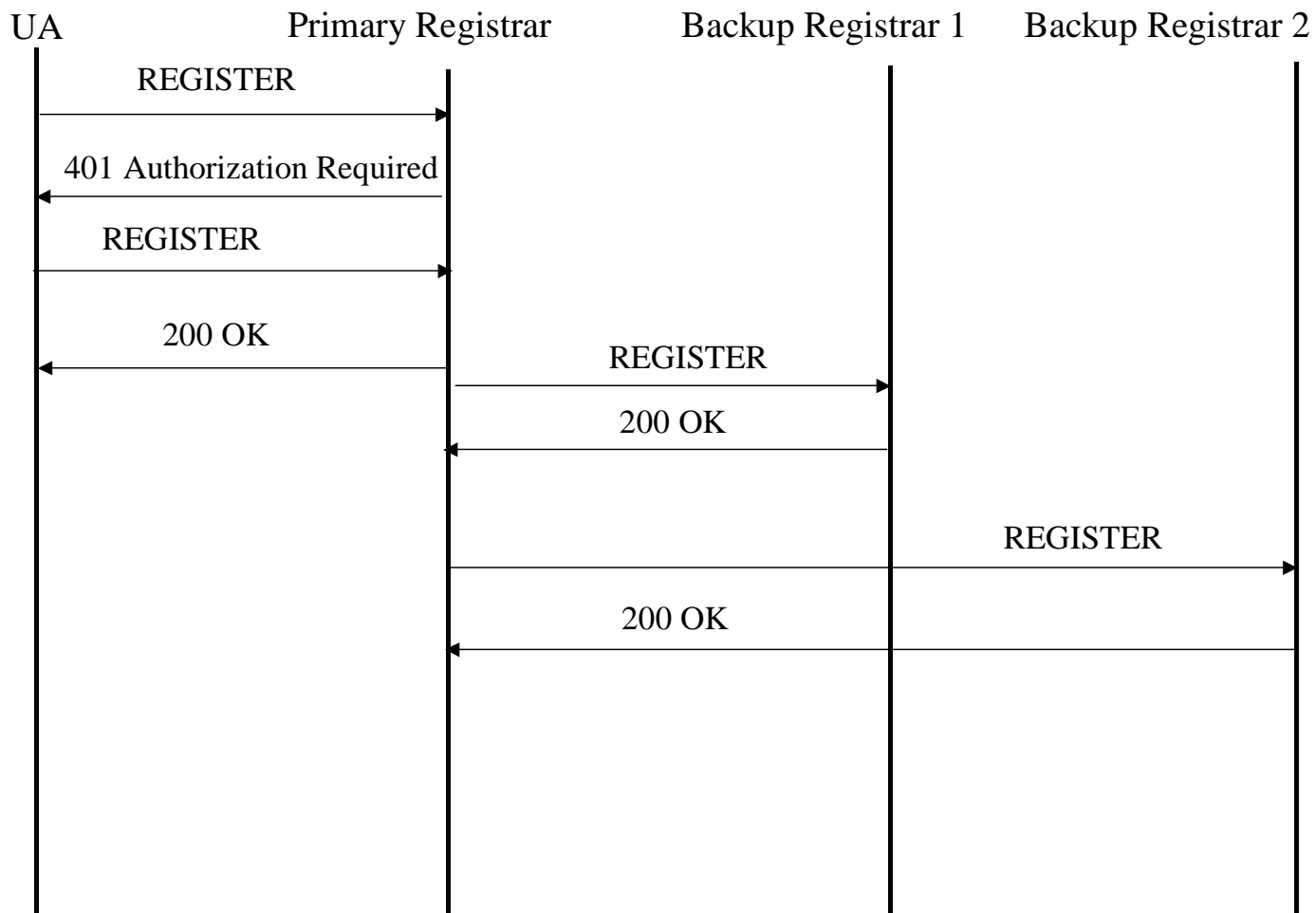
SIP high availability

- Server based replication (synchronous)



SIP high availability

- Server based replication (asynchronous)



SIP high availability

- Server based replication: consistency problem

If a backup server fails there must be a mechanism to re-distribute the data if it comes back online.

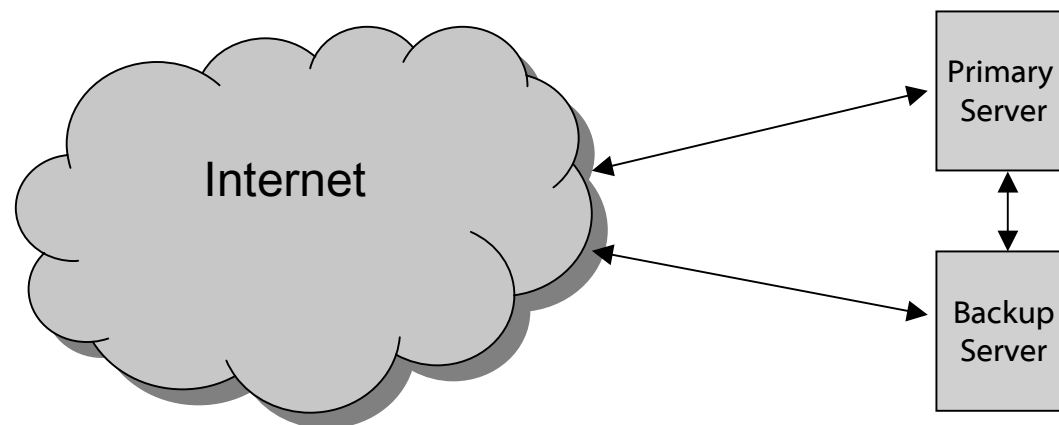
- master could remember which backup server is missing which records. (Where it did not receive „200 OK“ from for the REGISTER)
- If the master fails too, another backup server would have to do the re-distribution. Thus the knowledge about failed responses needs to be distributed too. SIP does not provide a method for this.

SIP high availability

So far: High Availability by Stand-By

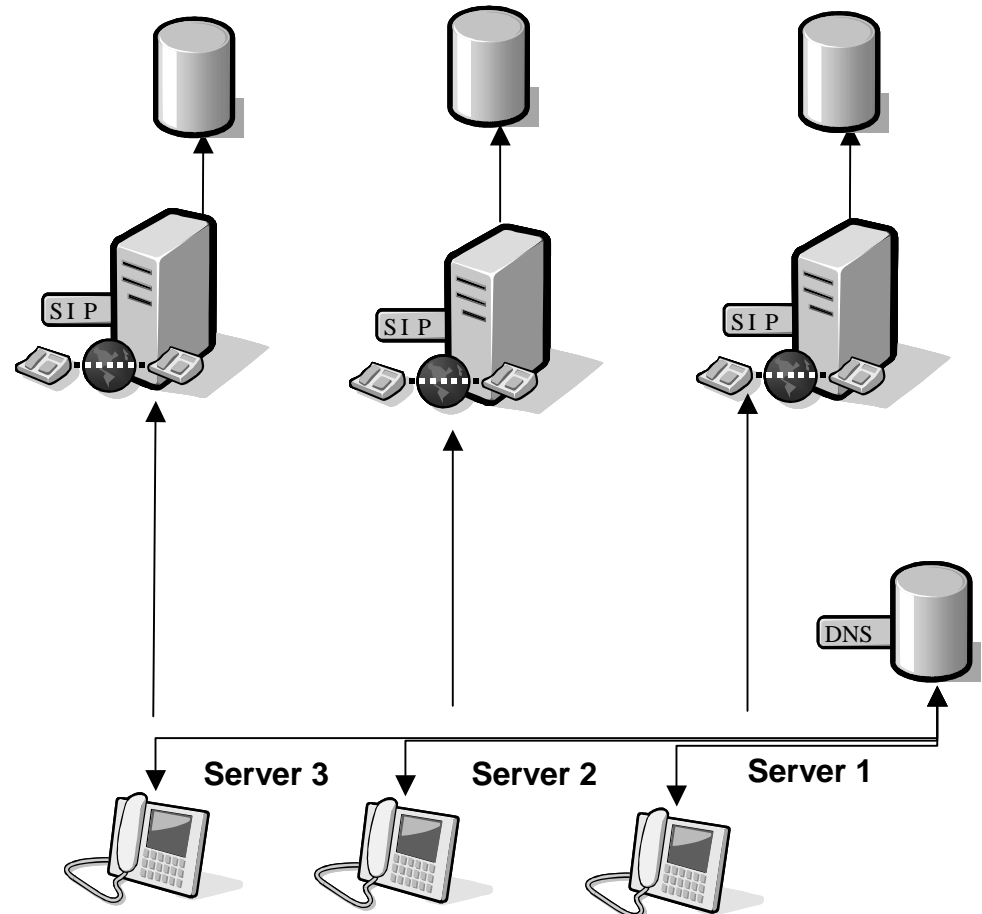
General drawbacks:

- Waste of Resources (Only One Worker)
- No Load Control



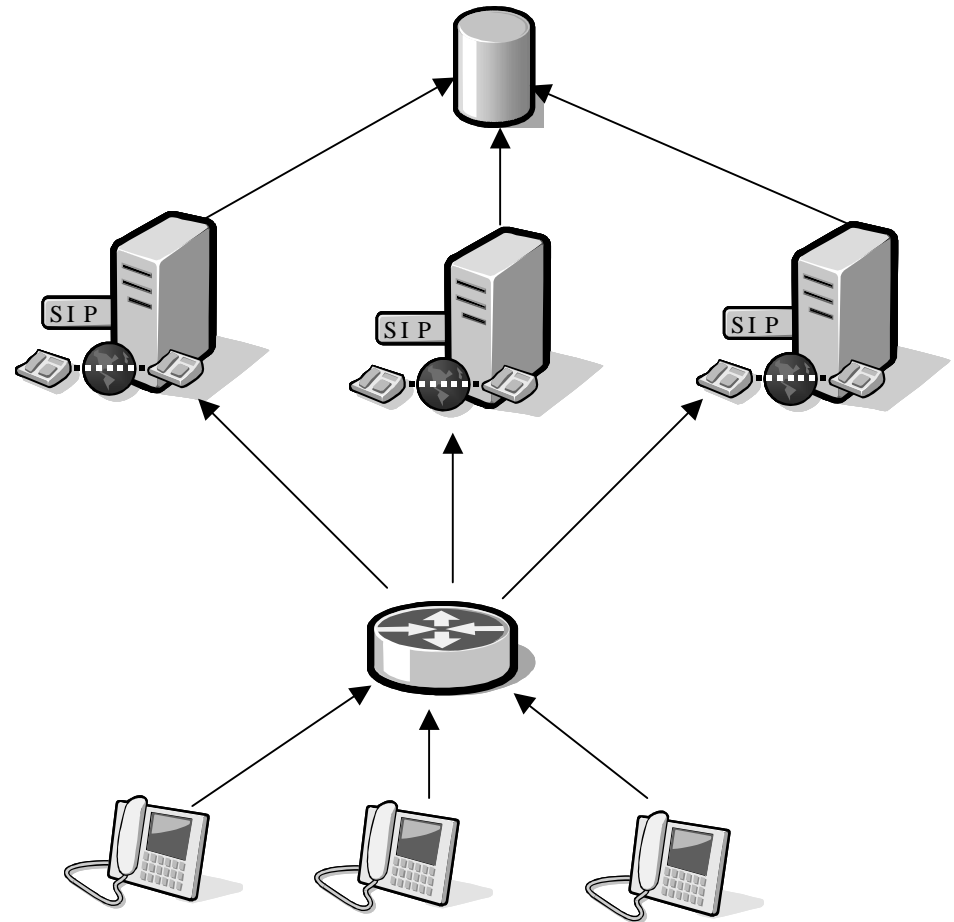
Scaling the Internet Way

- DNS returns different addresses on different queries
- Each proxy is responsible for a subset of users
- A user agent might receive different replies to different queries
 - Different requests from the same user agent might reach different proxies
 - NAT traversal not supported for all scenarios



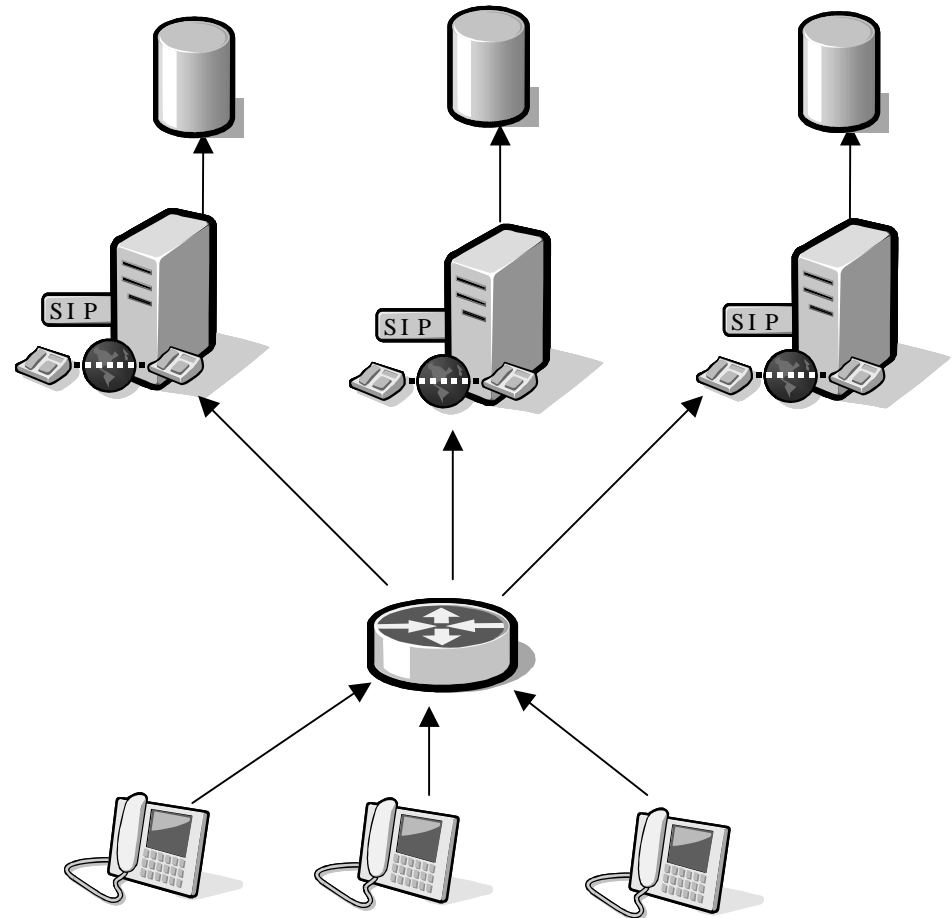
Scaling the Internet Way

- Use commercial traffic load balancers to distribute traffic
- All similar proxies
- Database shared between all proxies
- Load and size of database is huge
- Load balancer not SIP aware
 - NAT traversal not supported properly
- Single point of failure



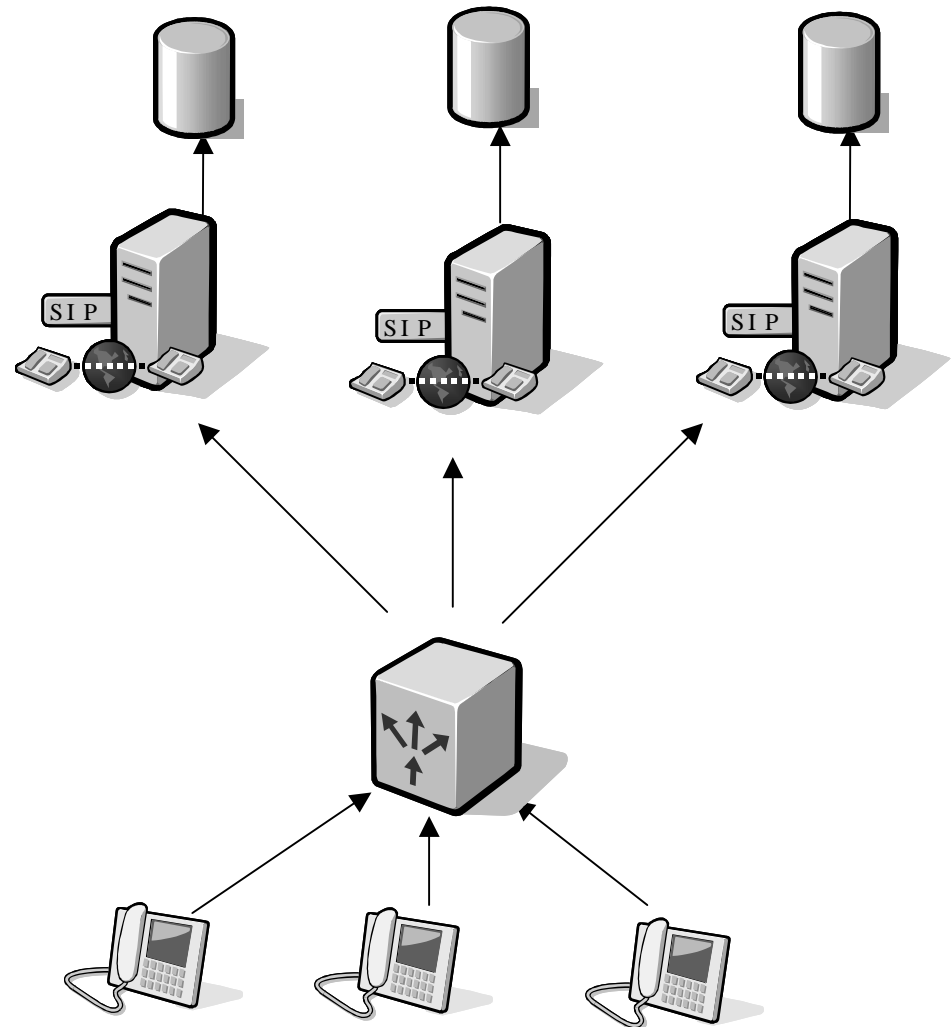
Scaling the Internet Way

- Use commercial traffic load balancers to distribute traffic
- Each proxy is responsible for a subset of users
- URI based distribution not sufficient
 - Users might have different aliases that do not map to the same proxy
- Load balancer not SIP aware
 - NAT traversal not supported properly
- Single point of failure



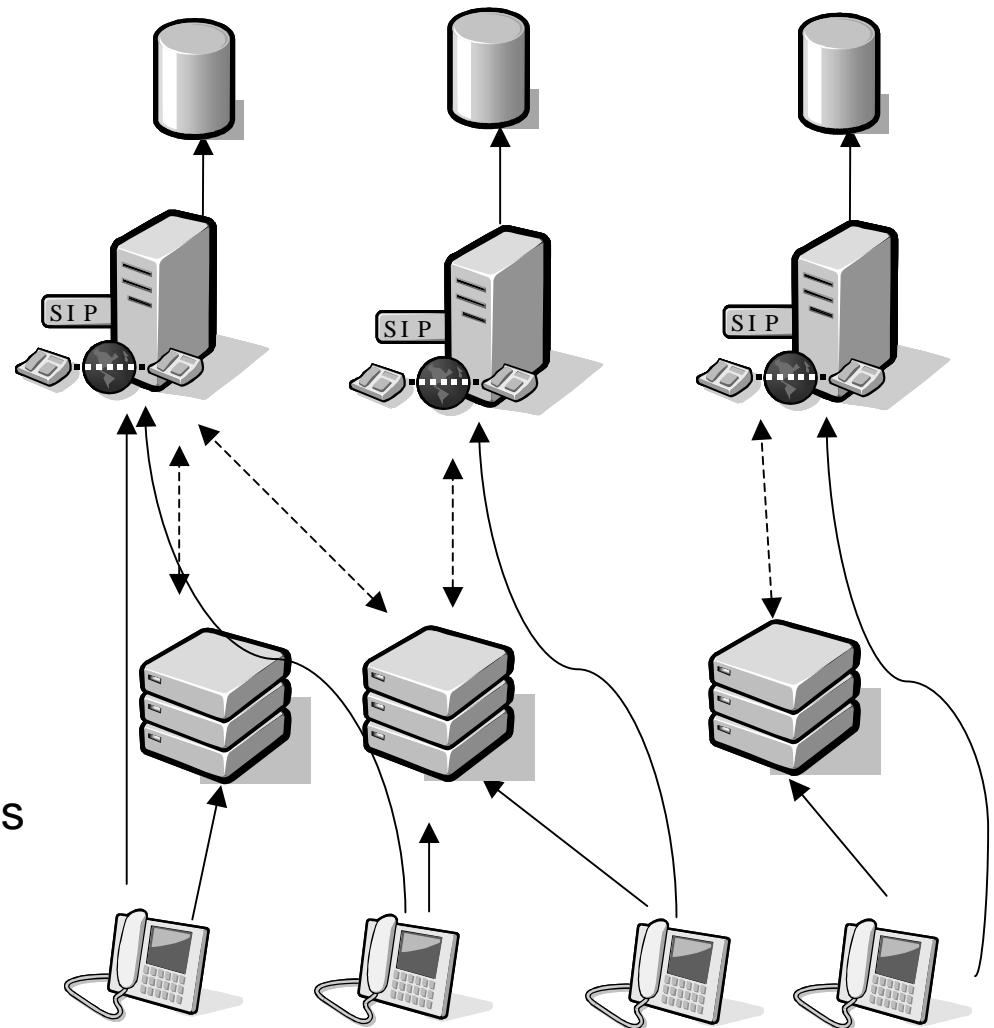
Scaling the SIP Way

- Use SIP-based load balancers to distribute traffic
- Each proxy is responsible for a subset of users
- Cooperation between load balancer and home proxies ensures
 - All requests from the same user will reach the same home proxy
 - Support the needed mechanisms for NAT traversal
- Single point of failure



Scaling the SIP Way

- Use SIP-based redirect servers to redirect users to their home proxies
- Each proxy is responsible for a subset of users
- User agents find a redirect server using DNS
- Redirect servers determines the user's home proxy and redirect the user to it
- Users contact their home proxy directly
- Cooperation between redirect servers and home proxies ensures
 - All requests from the same user will reach the same home proxy
 - Support the needed mechanisms for NAT traversal
- No single point of failure



Scaling the SIP Way

- Several Approaches exist (SIP aware vs. unaware, lb vs. redirect)
- Issues in
 - NAT Traversal (Who is the first proxy ?)
 - Session Integrity (How to ensure Client-2-Server Relation ?)
 - Flexibility (How to re-distribute load ?)

SIP high availability

Thanks !

Questions ?

jens.fiedler@fokus.fraunhofer.de