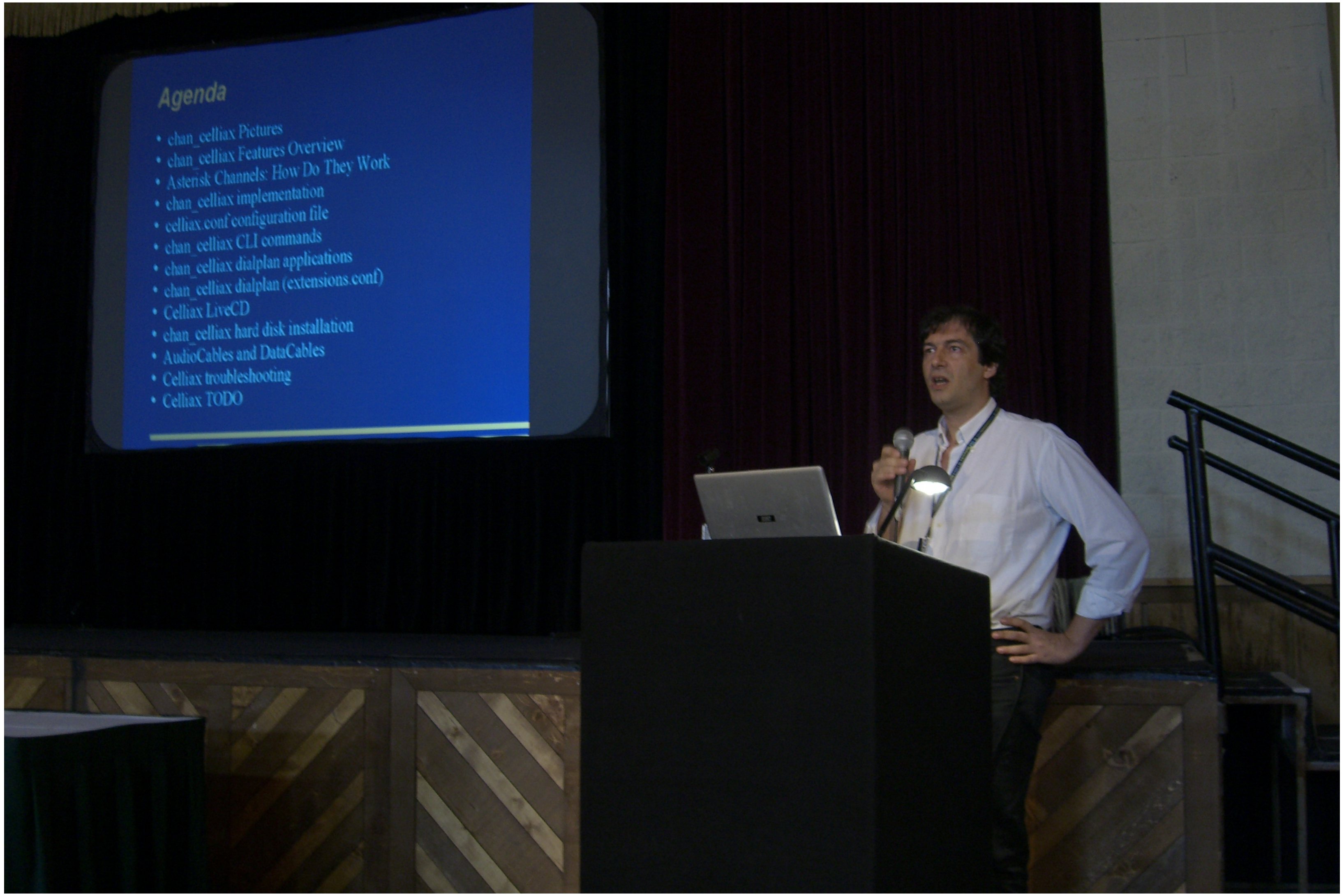
Industrial Grade FreeSWITCH

Scaling, Balancing and High Availability for SIP and WebRTC

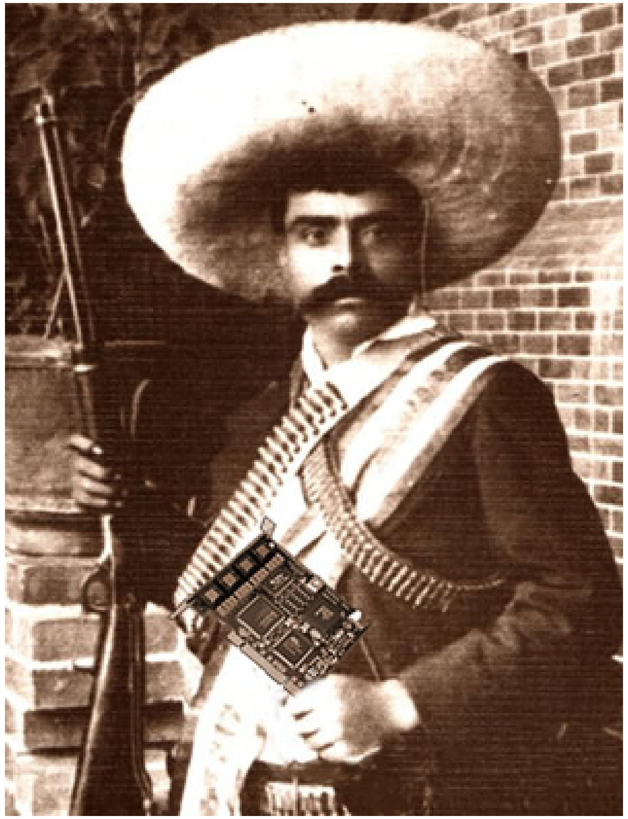
Giovanni Maruzzelli
gmaruzz@OpenTelecom.IT

First a memory...

Ten Years Ago
ASTRICON



Ten Years Ago
ASTRICON



JIM DIXON

AstriCon
ASTERISK USER
CONFERENCE & EXPO

September 27-29, 2016
Renaissance Phoenix Glendale Hotel & Spa

Jim "Dude" Dixon

Consultant

ZAPATA TELEPHONY PROJECT

The Zapata Telephony Project was conceived of by Jim Dixon, a telecommunications consulting engineer who was inspired by the incredible advances in CPU speeds that the computer industry has now come to take for granted. Dixon's belief was that far more economical telephony systems could be created if a card existed that had nothing more on it than the basic electronic components required to interface with a telephone circuit. Rather than having expensive components on the card, digital signal processing (DSP)[3] would be handled in the CPU by software. While this would impose a tremendous load on the CPU, Dixon was certain that the low cost of CPUs relative to their performance made them far more attractive than expensive DSPs, and, more importantly, that this price/performance ratio would continue to improve as CPUs continued to increase in power.

Like so many visionaries, Dixon believed that many others would see this opportunity, and that he merely had to wait for someone else to create what to him was an obvious improvement. After a few years, he noticed that not only had no one created these cards, but it seemed unlikely that anyone was ever going to. At that point it was clear that if he wanted a revolution, he was going to have to start it himself. And so the Zapata Telephony Project was born:

Since this concept was so revolutionary, and was certain to make a lot of waves in the industry, I decided on the Mexican revolutionary motif, and named the technology and organization after the famous Mexican revolutionary Emiliano Zapata. I decided to call the card the "tormenta" which, in Spanish, means "storm," but contextually is usually used to imply a big storm, like a hurricane or such.[4]

15 YEARS AGO

Zapata Telephony Organization, circa Nov. 2002 - Mozilla Firefox

www.zapatatelephony.org/oldindex.html

Welcome To Zapata Telephony!
(previously BSD Telephony Of Mexico)
Circa Nov. 2002

Gen. Emiliano Zapata, our inspiration

Zapata Telephony, dedicated to bringing the world a much-needed reasonable and affordable Computer Telephony platform, and hence a revolution in the arena of Computer Telephony.

Zapata -- PC-Based High Density Computer Telephony Project Information

Zapata Technology Design Philosophy

Zapata Project Status

Tormenta ISA Card, Rev. A -- Schematics and Board Artworks

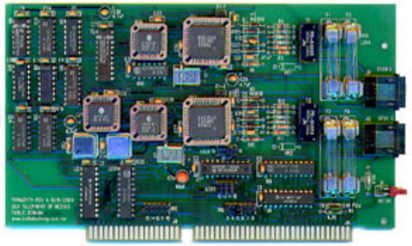
Tormenta 2 PCI Card, Rev. B -- Schematics and Board Artworks

Click Here for Radio Repeater/Remote Base Interface and Application

HARDWARE AVAILABILITY

Now Available!!

The *Tormenta ISA* PC card, supporting 2 full channelized-T1circuits (48 voice ports) is available in small quantity now. We are no longer building cards ourselves. There are, however, several entities that are making them currently. You can send us email (see below), and we will pass along your requests/orders to the entities currently manufacturing the boards. For those wishing to fabricate their own cards, the gerber photoplot files may be downloaded here. 
Please note Mandatory Engineering Change.



Tormenta ISA Card, Rev. A
(shown without mounting bracket)

PCI Version Development Completed

15 YEARS AGO

BIG APPLAUSE

FOR

JIM DIXON

You Tube

Search

¡Viva la Revolución! -- Jim "Dude" Dixon -- Astricon 2014, Las Vegas, NV

D

DudesKitchen

Subscribe 6

234 views

Add to Share ••• More 1 0

Published on Oct 30, 2014

Jim "Dude" Dixon, communications technology revolutionary, visionary and innovator, gives speech at Astricon 2014 regarding his involvement in the history of open-source telephony, Asterisk, and how it all came about.

Back to HA and Scalability

- SIP
- WebRTC
- (audio|video) Calls
- (audio|video) Conferencing & ACD
- Presence
- Instant Messaging

FreeSWITCH

- most powerful multimedia switch
- SIP/Verto/WebRTC/TDM support
- HD audio and video transcode and mixing
- enterprise PBX features
- static dialplan / dialplan from http / scripts execution / remote call management
- Auto Attendant / IVR / fully programmable access to DBs and legacy systems
- multi language sounds management with locale smart phrases

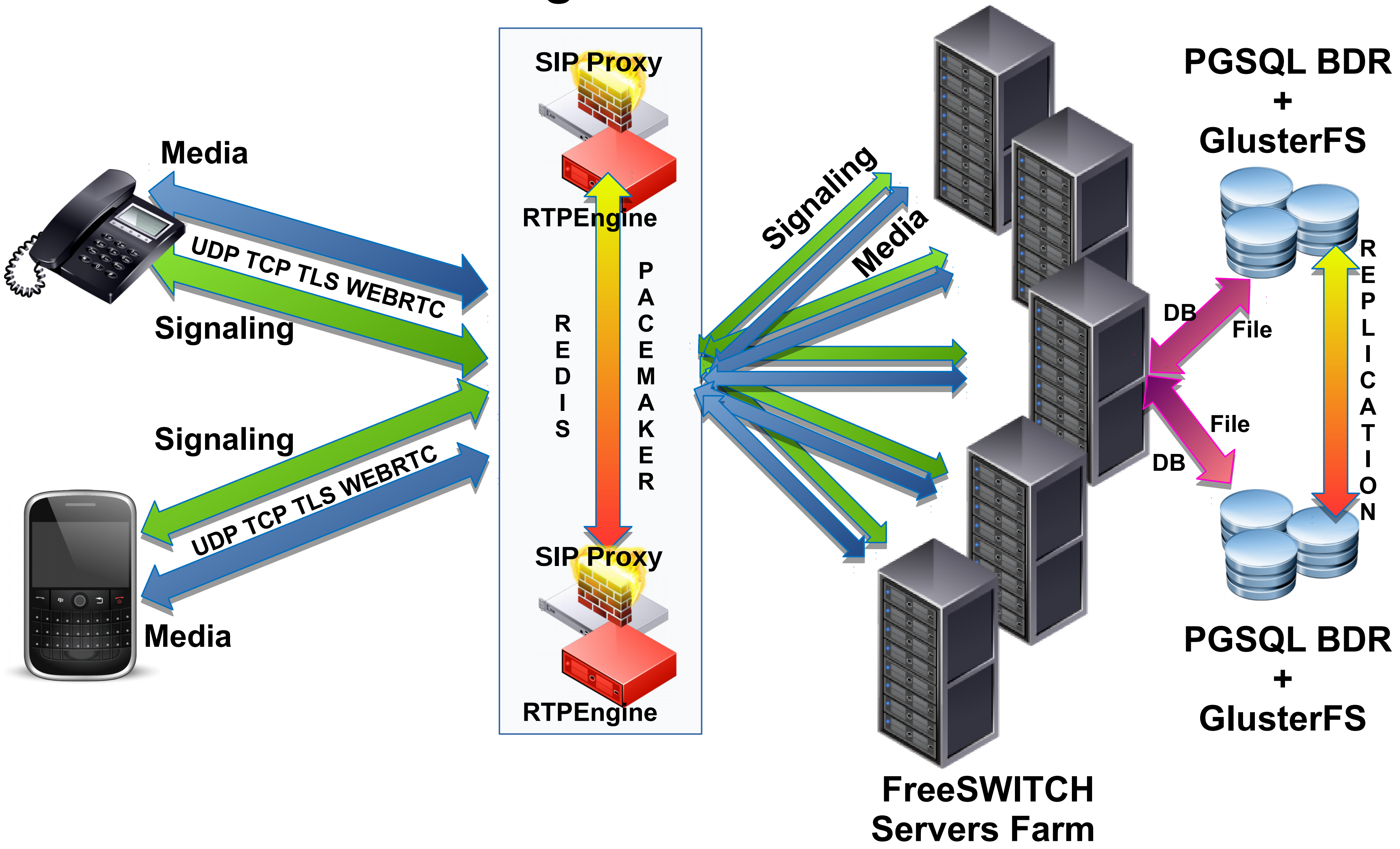
FreeSWITCH

- TTS and ASR
- audio/video conferencing
- enterprise video MCU and CG effects (and CPU-friendly video follow floor SFU mode)
- fully featured carrier grade voicemail
- callcenter / ACD / call queues management
- best fax/T38 support
- multiple SIP gateway support with failover
- complete multidimensional CDR generation

go with the pros

- Corosync, Pacemaker and crmsh/pcs available on Debian
- PostgreSQL BDR released stable
- GlusterFS got Perfomances on Small Files Workloads
- OpenSIPS got Clusterer, Mid-Registar, and FS realtime load
- Kamailio got KDMQ
- RTPEngine got restore at startup from Redis
- Redis got replication failover
- HAProxy balance WSS, HTTP(S), and PostgreSQL
- FusionPBX does FS config, user mngmt, and device provisioning
- HOMER do signaling capture, history, stats and monitoring
- CGRates do the mediation, rating, accounting and billing

SIP
Scaling FreeSWITCHes

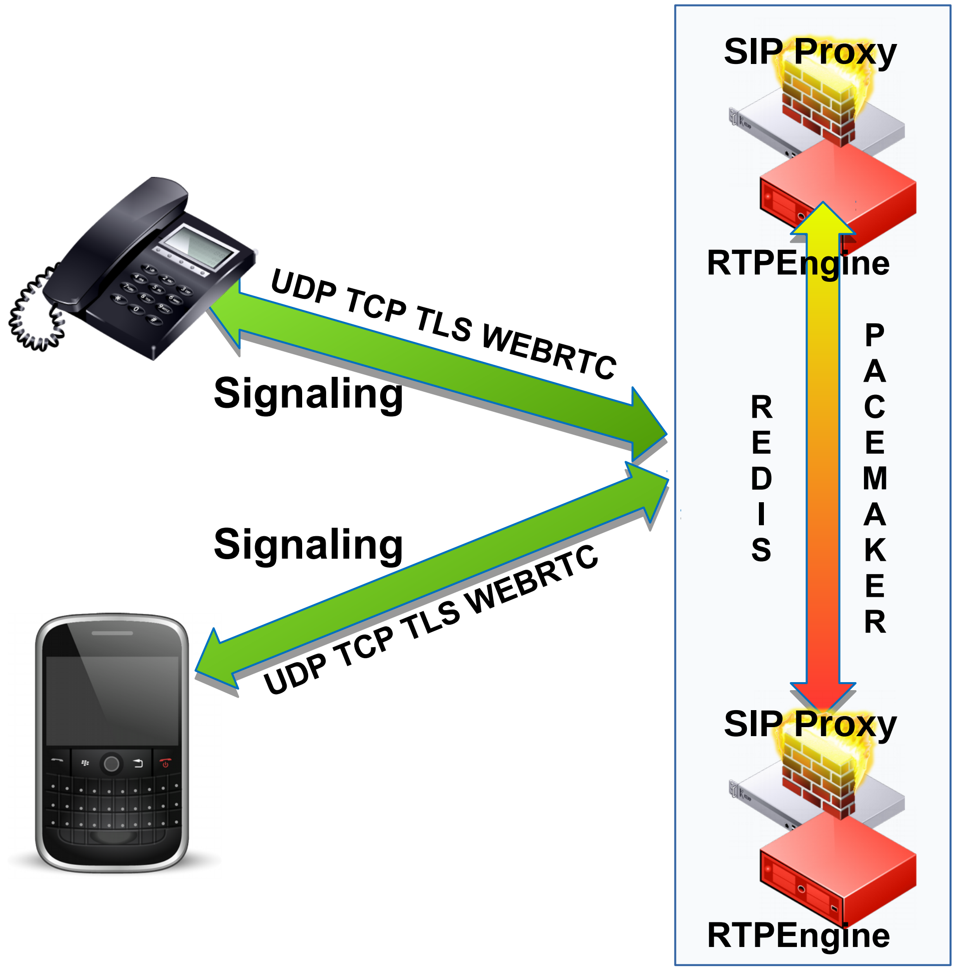


SIP and **NAT**

- Client is behind **NAT**, not directly reachable by server
- Client sends from its own IP:port a **REGISTER** request to Location Server IP:port, and in doing so it **opens a pinhole** in the NAT, waiting for server's answer
- NAT pinhole is only able to receive packets from **same IP:port** couple (Client/Server) it was open by, **and for a limited period** of time (30 seconds?)
- Location Server **sends periodically from same IP:port an** **OPTIONS** message to Client IP:port, Client answers, and in doing so it maintains the **pinhole open**
- When there is an incoming call for Client, Server sends the **INVITE** from **same IP:port** to Client IP:port

SIP

Load Balancing and Clients



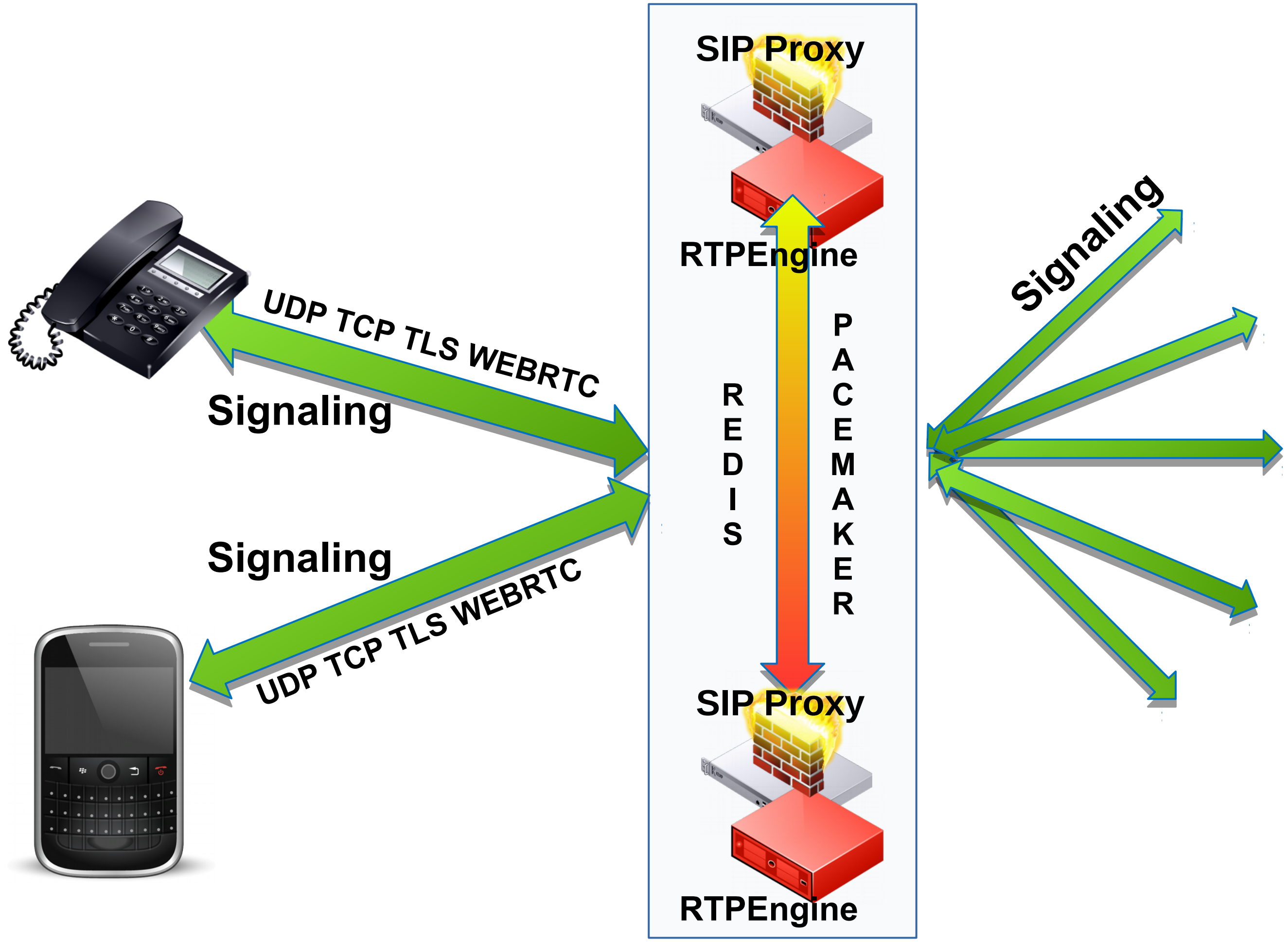
Where to put the **SIP** Registrar

- **ON LB (SIP Proxy) MACHINE**, directly interacting with Clients
  - REGISTER and NAT Keepalive (OPTIONS) are high volume, low load transactions
  - One robust box (in active-passive HA) will be able to serve tens of thousands clients
  - This is the most straightforward topology

- **REGISTRATION is then Forwarded to FreeSWITCH MACHINES**, load balanced by LB (SIP Proxy)
  - FreeSWITCHes are made aware of registration (eg, where the phone is) created and deleted
  - No periodic registration traffic, no NAT keepalive traffic

SIP

Load Balancing and Signaling



SIP Call Distribution: DISPATCHER & LOAD BALANCER

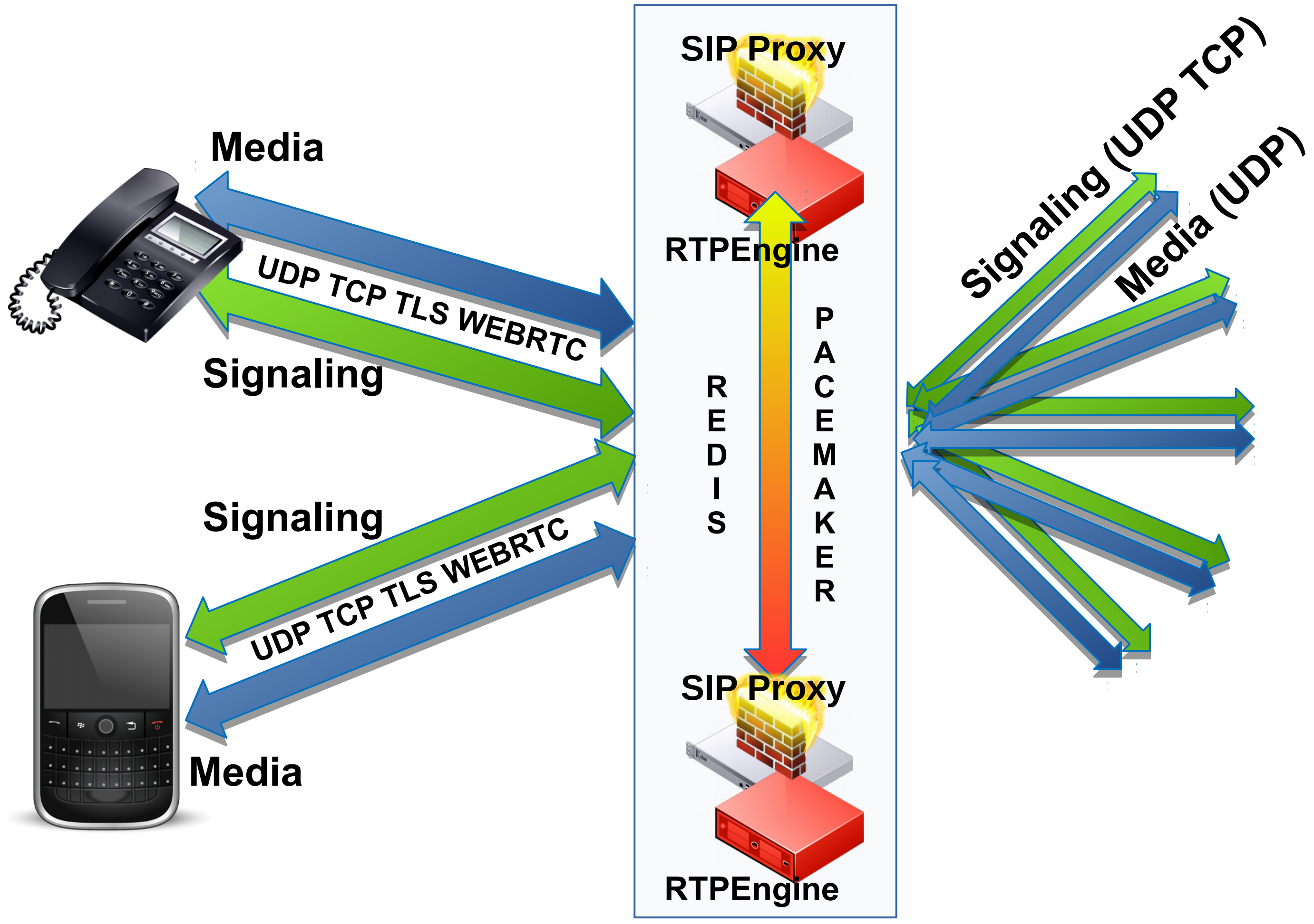
- SIP Proxy relays requests to multiple boxes using:
  - “static” algorithms
    - round robin
    - weighted
  - “dynamic” algorithms
    - actual number of active calls on each machine
    - actual load on each machine
- All proxy's algorithms are able to “ping” destinations, retry on failed destination, disable the failed box from list, and re-enable it when is back in order

Security / Fraud Detection / DOS

- you want to block things OUTSIDE your perimeter
  - at most on the LB (SIP Proxy)
- DOS / DDOS
  - Pike
- Fraud Detection
  - Identify Suspect Patterns, or Anomalies
- Block Traffic
  - malformed

SIP

Load Balancing and Media



SIP Media Relaying

SIP proxy has nothing to do with media flow, it does not touch RTP

- Proxy modify SIP headers and SDP bodies so clients behind restrictive NAT use a relay, and it directly command that relay
- Relay then knows which RTP stream must be relayed to which client

- Original relay software is “Rtpproxy”
- More recent relays (feat: kernel space, encryption, transcoding):
  - MediaProxy
  - RtpEngine
- All of them can scale indefinitely
- RtpEngine can restore sockets from Redis at startup

SIP calls don't drop

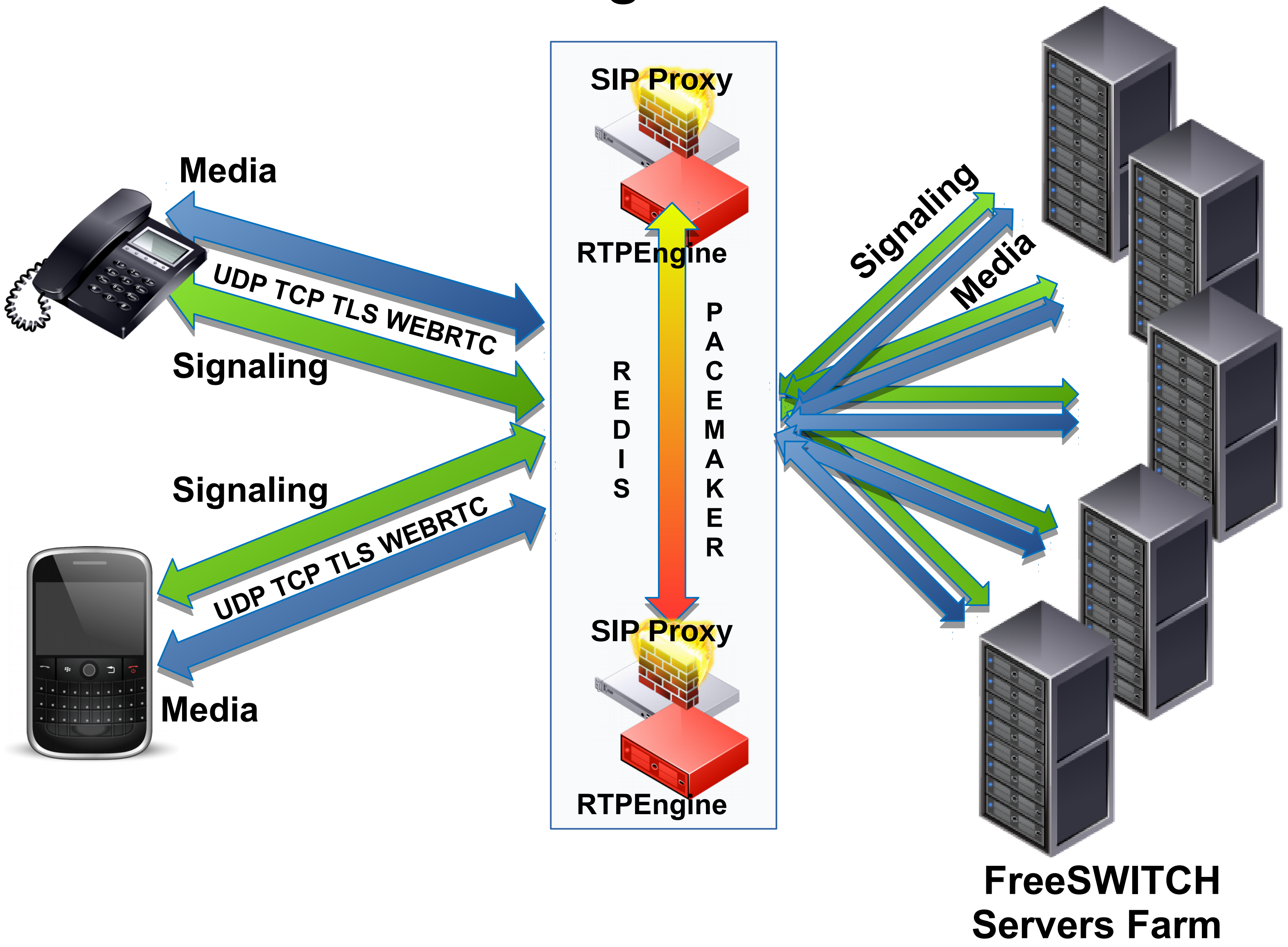
- client ip/port to relay/proxy ip/port
- relay/proxy goes down
- relay/proxy ip moves
- new relay/proxy online
- registrations, NAT, etc taken care
- RTPEngine restart from REDIS
- client ip/port to relay ip/port restored

Standard Calls

(no need for special landing)

- Registered Phone to Registered Phone
(eg “Internal Calls”)
- Registered Phone to ITSP gw
(eg “Outbound Calls”)
- ITSP to Registered Phone
(eg “Inbound DID Calls”)
- Registered Phone to VoiceMail
(eg Check Messages)
- ITSP to VoiceMail
(eg Leave Message)
- IVR / Automated Attendant

SIP
Scaling FreeSWITCHes



SIP Signaling, Again
(Presence, BLF, Messaging)

ALL PURE SIGNALING ARE BELONG TO SIP PROXY

- Presence
  - **SUBSCRIBE PUBLISH NOTIFY**
    - Event: State (Available, Busy, Do Not Disturb, Away)

- Blinking Field Lamp (BLF)
  - **SUBSCRIBE PUBLISH NOTIFY**
    - Event: Dialog (Idle, Ringing, Calling, in a call)

- Messaging, Chat
  - **MESSAGE (SIMPLE)**

SIP Signaling, Again
(MWI, SLA/SCA, QUEUEs)

SMART, APPLICATION LEVEL SIGNALING COMES FROM FREESWITCH

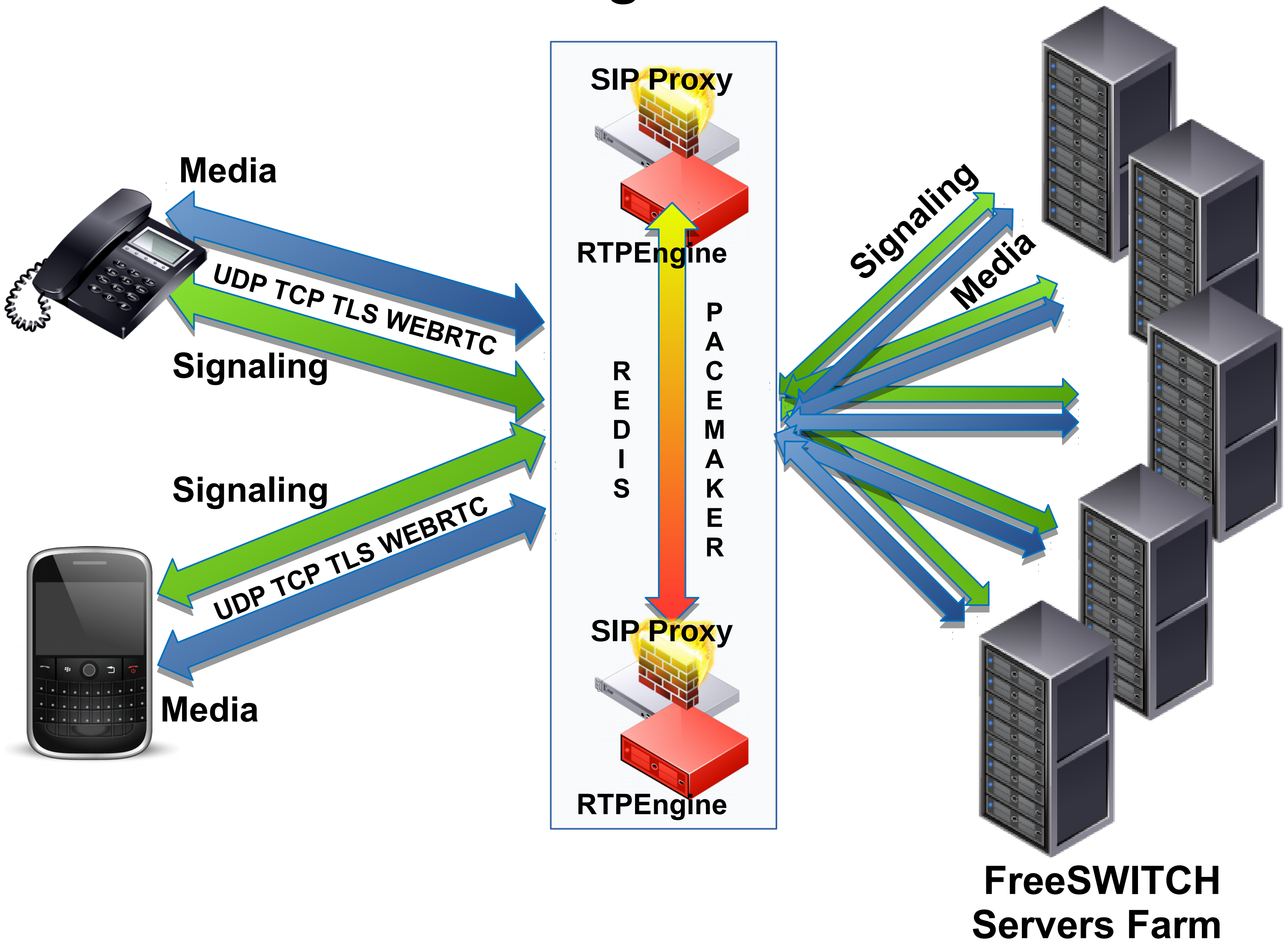
- Message Waiting Indicator
  - **NOTIFY**
    - Event: Voicemail (at first REGISTER, then when msgs changes)

- Shared Line Appearance / Shared Call Appearance
  - **SUBSCRIBE NOTIFY**
    - Event: Call Ongoing / Handset Off Hook (Manager/Assistant)

- Queues
  - **SUBSCRIBE NOTIFY**
    - Event: Calls Num (Calls into Queue)

SIP

Scaling FreeSWITCHes



Special Cases
(must be managed)

- Load Balancing is predicated on a server farm of equivalent and equipollent (eg: interchangeable) servers

- There are cases for which this is not true:

  - Conferences
  - Call Queues
  - Call Park – Unpark
  - Call/Group Pickup (Intercept)
  - And so on, and so on (quot. Zizek)

Conferences, Call Queues, Call Parks
(must be local to one FS machine)

- **Conferences** are multiple calls' media streams mixed together (think multitrack video/audio editing software), result stream is then broadcasted to participants

- **Call Queues** are stacks of incoming calls, all of them listening to Music on Hold, waiting to be dispatched to answering agents. It is possible to inject streams to single callers (eg "You are 3rd in line, your average waiting time is 9 minutes")

- **Call Parks** are named stalls where you put a call, and after a while you or someone else pick it up

Special Cases

(hash on destination)

opensips.cfg (~) - VIM

```
        $var(destination) = "" + $rU;
        $var(port) = "5060";

        $var(destinationmd5) = $(var(destination){s.md5}{s.substr,0,1}) ;
        $var(destinationmd5hex)="0x" + $var(destinationmd5);
        $var(destinationmd5int) = (int)$var(destinationmd5hex);
        $var(destinationmd5intmodulo) = $var(destinationmd5int) mod 3;

        switch ($var(destinationmd5intmodulo)) {
                case 0:
                        $du = "sip:192.168.1.117:" + $var(port) ;
                        break;
                case 1:
                        $du = "sip:192.168.1.116:" + $var(port) ;
                        break;
                case 2:
                        $du = "sip:192.168.1.113:" + $var(port) ;
                        break;
                default:
                        $du = "N/A";
        }
                                                            3,0-1          Top
```

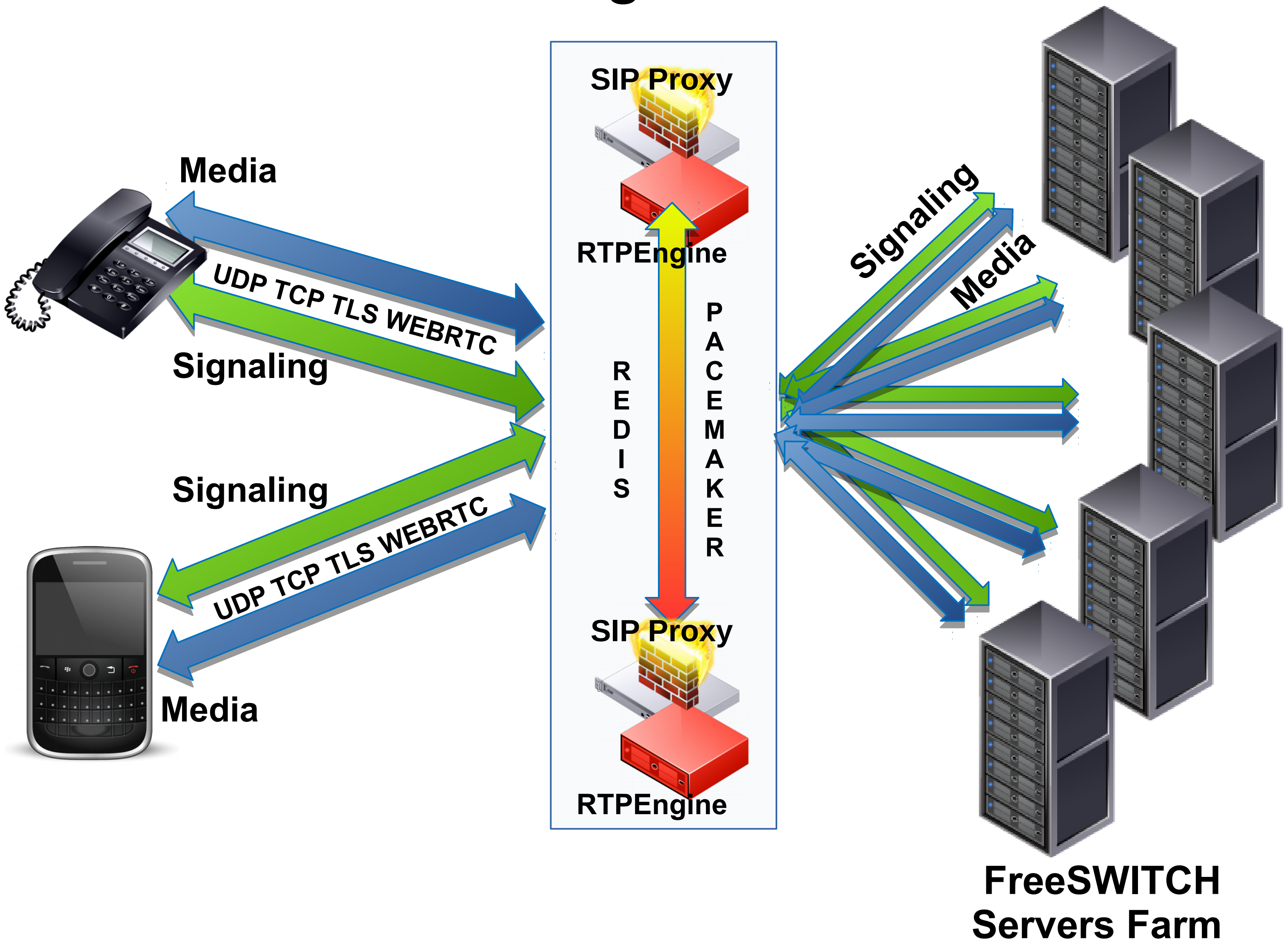
Call/Group Pickups

- **Call (extension) Pickup:**
  - A call is ringing on a phone a desk away on your same group, you press *4EXT and answer the call
- **Group (last call) Pickup:**
  - Someone answered a call, you are in her same group, she stares at you and nod, you press *8 and pickup the call

Those two cases can be managed by inserting call groups' belonging info into a DB table

SIP

Scaling FreeSWITCHes



Multi Tenancy
Small - Medium Domains

- Multi Tenant = Multiple SIP/WebRTC domains, managed independently
- Farm is partitioned on Domains by the Proxy, each domain goes to a particular machine
- This solves the conferencing-queues-transfer-pickup issues (eg locality of calls/users)

- High Availability by one or more SPARE machines, ready to take the role of the failed machine

Multi Tenancy

(hash on domain)

opensips.cfg + (~) - VIM

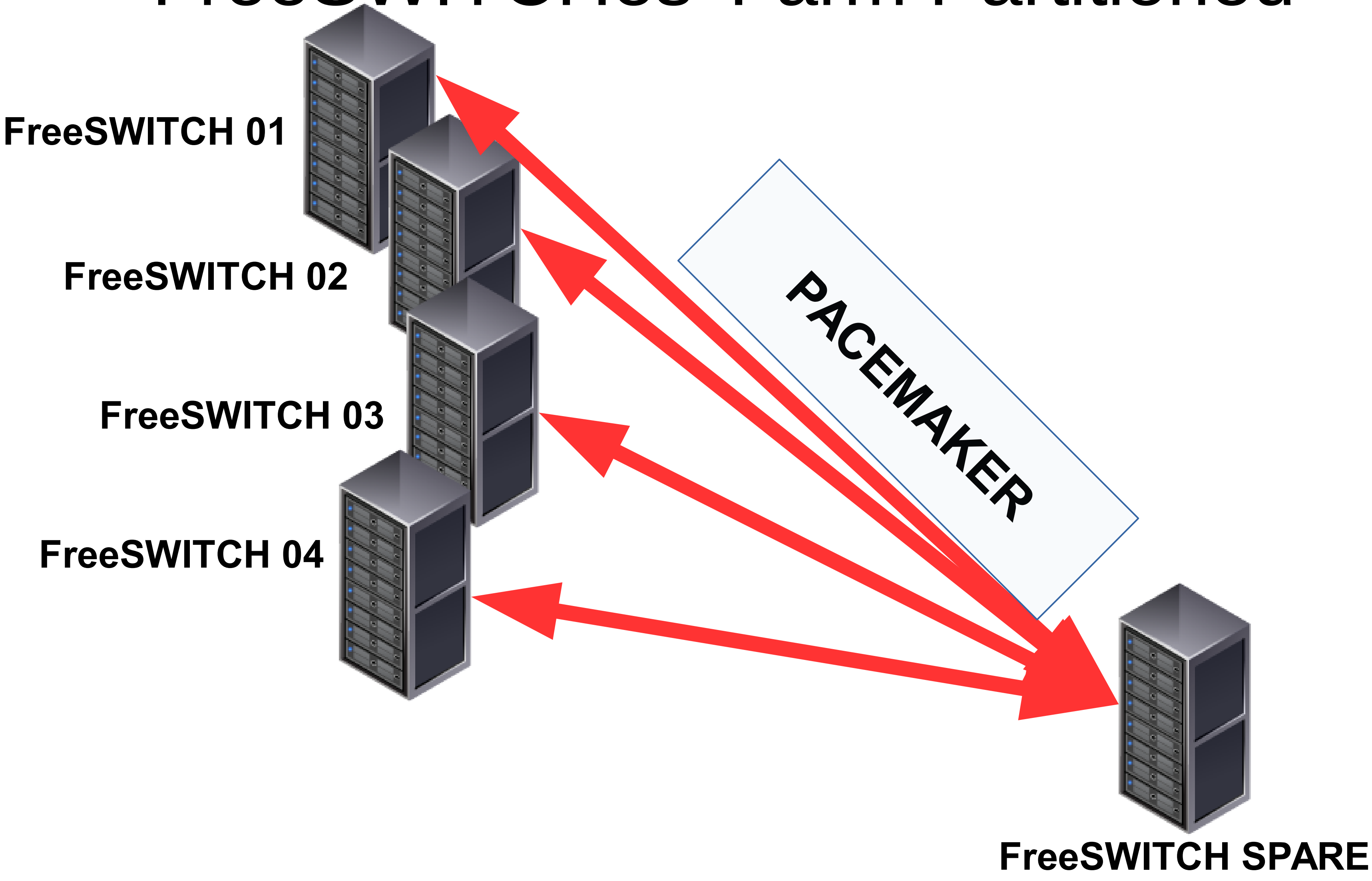
```
        $var(domain) = "" + $td;
        $var(port) = "5060";

        $var(domainmd5) = $(var(domain){s.md5}{s.substr,0,1}) ;
        $var(domainmd5hex)="0x" + $var(domainmd5);
        $var(domainmd5int) = (int)$var(domainmd5hex);
        $var(domainmd5intmodulo) = $var(domainmd5int) mod 3;

        switch ($var(domainmd5intmodulo)) {
                case 0:
                        $du = "sip:192.168.1.117:" + $var(port) ;
                        break;
                case 1:
                        $du = "sip:192.168.1.116:" + $var(port) ;
                        break;
                case 2:
                        $du = "sip:192.168.1.113:" + $var(port) ;
                        break;
                default:
                        $du = "N/A";
        }
                                                        1,1           Top
```

Multi Tenancy

FreeSWITCHes' Farm Partitioned



VERTO User Partitioning

- **VERTO**, at this moment, has **NO TRUNKING**
  - Each FreeSWITCH Server is a VERTO Island!
  - As of today, you use **SIP to Trunk** from one FS VERTO server to another VERTO server

- **VERTO**, at this moment, has **no external** “VERTO **proxies**” and “VERTO **registrars**”
  - VERTO users (extensions) atm must be partitioned at client side
  - Client is under our control! (is a web page!)
  - Each users partition (by domain and/or by extension) is sent to a specific FS server via port forwarding

VERTO and NAT

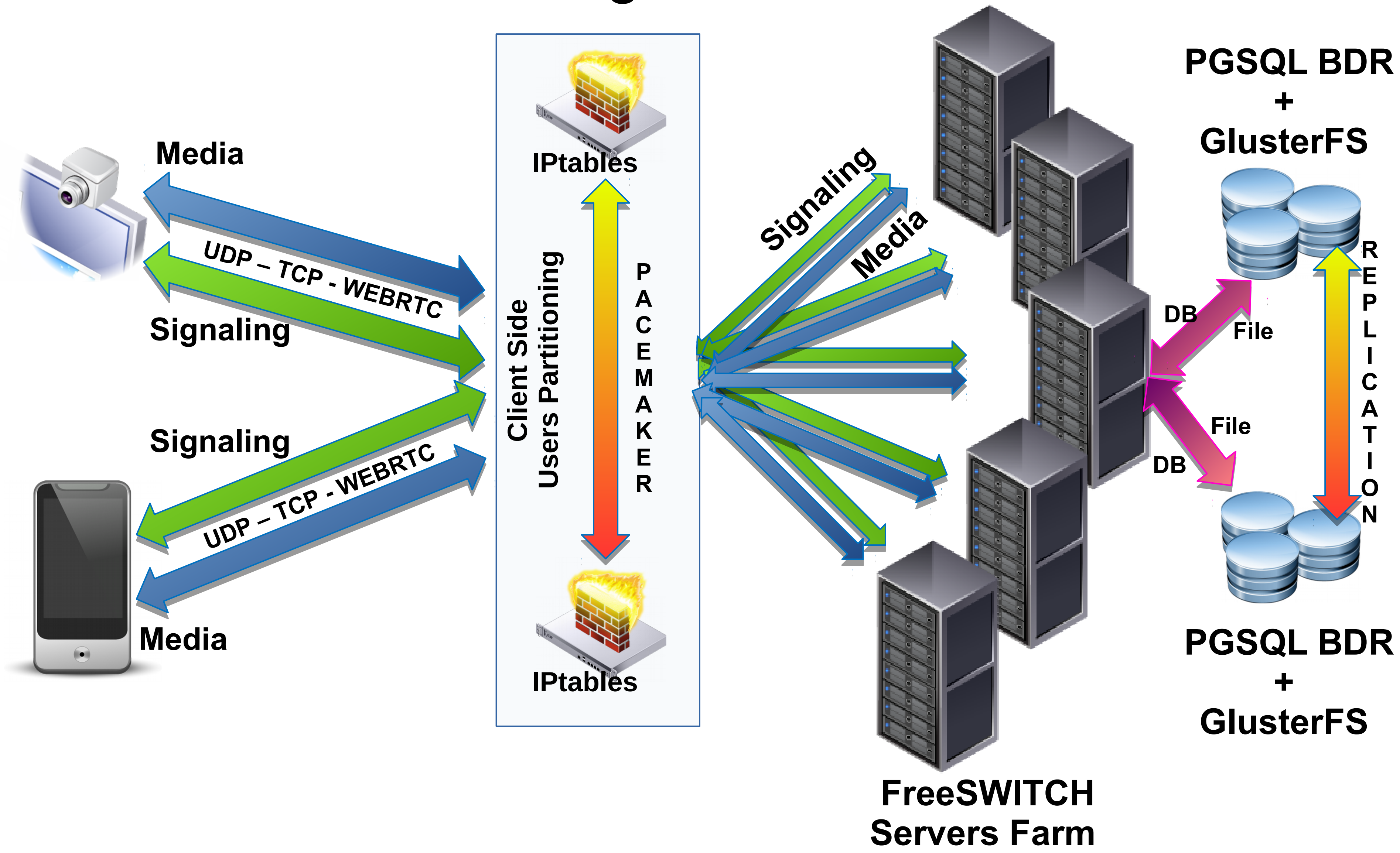
ICE

VERTO Call Balancing:

RTP IP, IPTables & IP Ranges

- All FreeSWITCH servers have ext-rtp-ip set to LB address in verto.conf.xml
- Each FreeSWITCH server has its own range of RTP ports set in switch.conf.xml
- IPTables will forward RTP back and forth from LB to the correct FreeSWITCH
- If a FreeSWITCH server dies, clients will automatically reconnect to the new instance of that server (that's the beauty of TCP wss)

VERTO
Big Picture



Platform Components

...go with the Pros

Pacemaker - Corosync

- Proven, Professional Cluster Framework
- Active-Passive, Ative-Active, N+1, N to M models
- Resource Management
- Quota Management
- Split Brain Avoidance
- Fencing
- STONITH

GlusterFS

- Distributed Cluster Filesystem
- Servers exports Bricks (can be striped and multimachine)
- Bricks can be replicated real time
- Clients mount Bricks
- Data vs Metadata access (favor big files)
- Small Files WorkLoad Improvements

Redis

- Network Key/Value Store
- Built in Replica and High Availability
- Many Primitives
- Hard Disk Persistency
- Very Fast
- Used by FreeSWITCH, OpenSIPS, RTPEngine, Kamailio

PostgreSQL BDR

- PGSQL most reliable DB
- Bi Directional Replication
- Master - Master
- BDR Just reached Release
- Real Time Replication
- Need a PK on each Table (UUID anyone?)
- Need care when modify table schema

HAproxy

- fastest, featureful and most stable:
  - HTTP(S) Load Balancer
  - SSL Gateway
  - Application High Availability
  - TCP and WSS Tunnel
  - PostgreSQL Connection Failover

OpenSIPS / Kamailio

- Most powerful SIP Proxies
- They command Media Proxies
- They do security, filtering, decoupling
- Kamailio got DMQ for clustering
- OpenSIPS got Clusterer
- OpenSIPS got FS realtime Load and Mid-Registrar
- They can manage ten of thousands users on a modest box
- REGISTER - OPTION - MESSAGE - NOTIFY - SUBSCRIBE

RTPEngine

- advanced Media Proxy
- commanded by SIP Proxy
- connects RTP streams between non routable endpoints
- in Kernel packet moving
- encryption and recording management
- able to restore all sockets and states at startup, from Redis

FusionPBX

- FreeSWITCH Configuration and Management GUI
- Multi Domain
- User Management
  - permission
  - groups (superadmin/admin/user)
- Device Provisioning
- Dialplan, IVR, Queue, Fax, VoiceMail, etc
- Distributed, High Available

HOMER

- gets signaling traffic from all your SIP and RTC network
- store it in a DB
- give you stats, graphs, and history
- real time queries
- real time and long term troubleshoting
- monitoring
- trends

CGRates

- Flexible and Performant Call Rating
- GOlang and ESL
- Many Billing Models
  - prepaid
  - postpaid
- Authorization
- Legacy Interfacing

Going the Further Steps

- Multiple Data Centers
  - Geo Distribution
  - High Availability

- Disaster Recovery
  - your Data Center goes offline

Multi Data Center

- SRV DNS Records
- Geographic Distribution
  - network lag
- Use Route53 !
- AS / BGP
  - ok, that's another level
- PostgreSQL BDR, GlusterFS, Redis, OpenSIPS, Kamailio can be cross DC replicated

Disaster Recovery

High Availability is NOT **Disaster Recovery**

- When your Data Center goes offline
  - Power Failure
  - Network Partition
  - Natural Disaster
- You replicate your entire platform on Cloud
  - BIG, POWERFUL, and COSTLY machines
  - Fresh Configuration Backups, versioned and real time
  - Withstand a Registration Storm, then scale down

Scalability, High Availability,

HOW MUCH DO THEY COST?

If you have to ask the price, you can't afford it

If You Need to Start Cheap

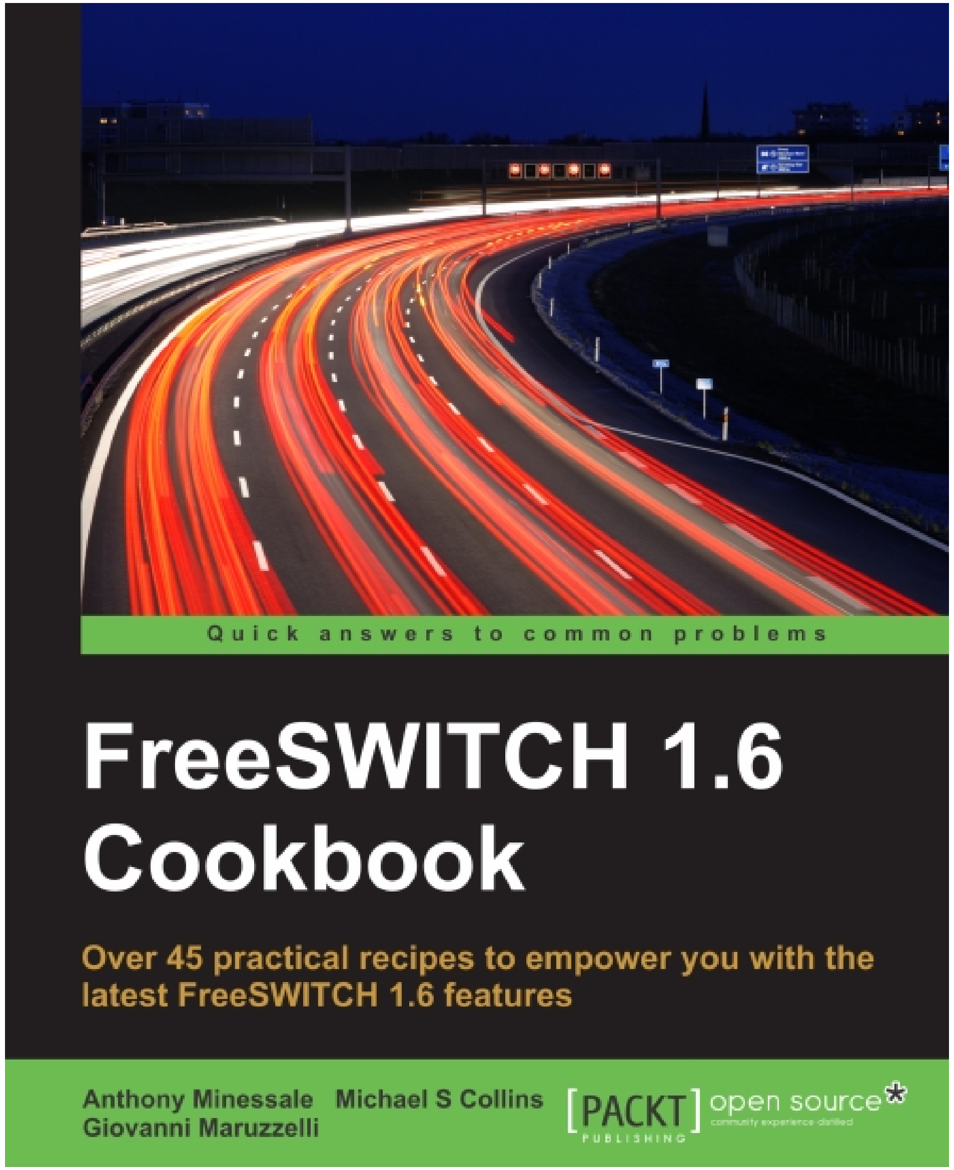
- minimum 3 hardware machines
- they will host LXC containers (virtual machines)

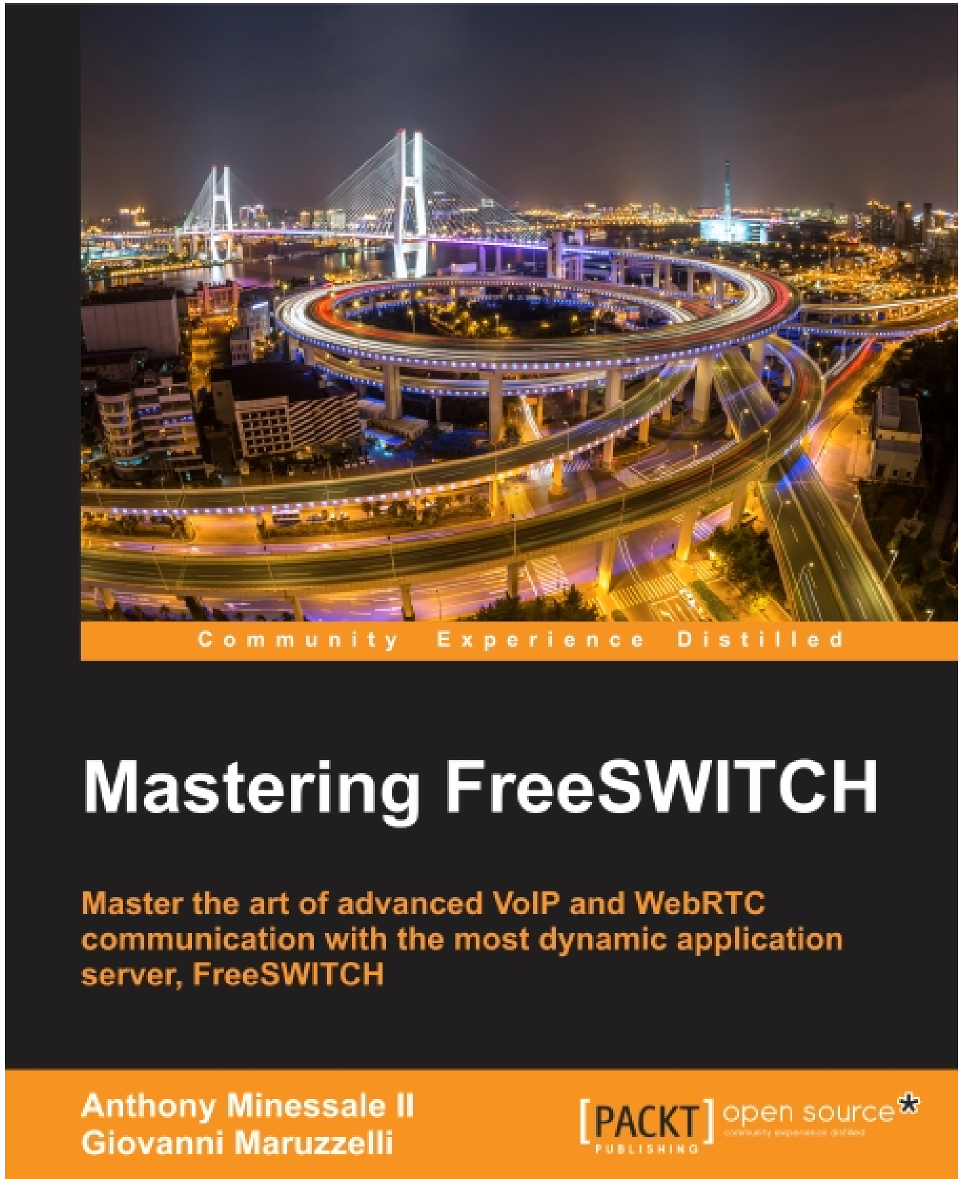
a cluster of 2

is a

NO NO

www.packtpub.com

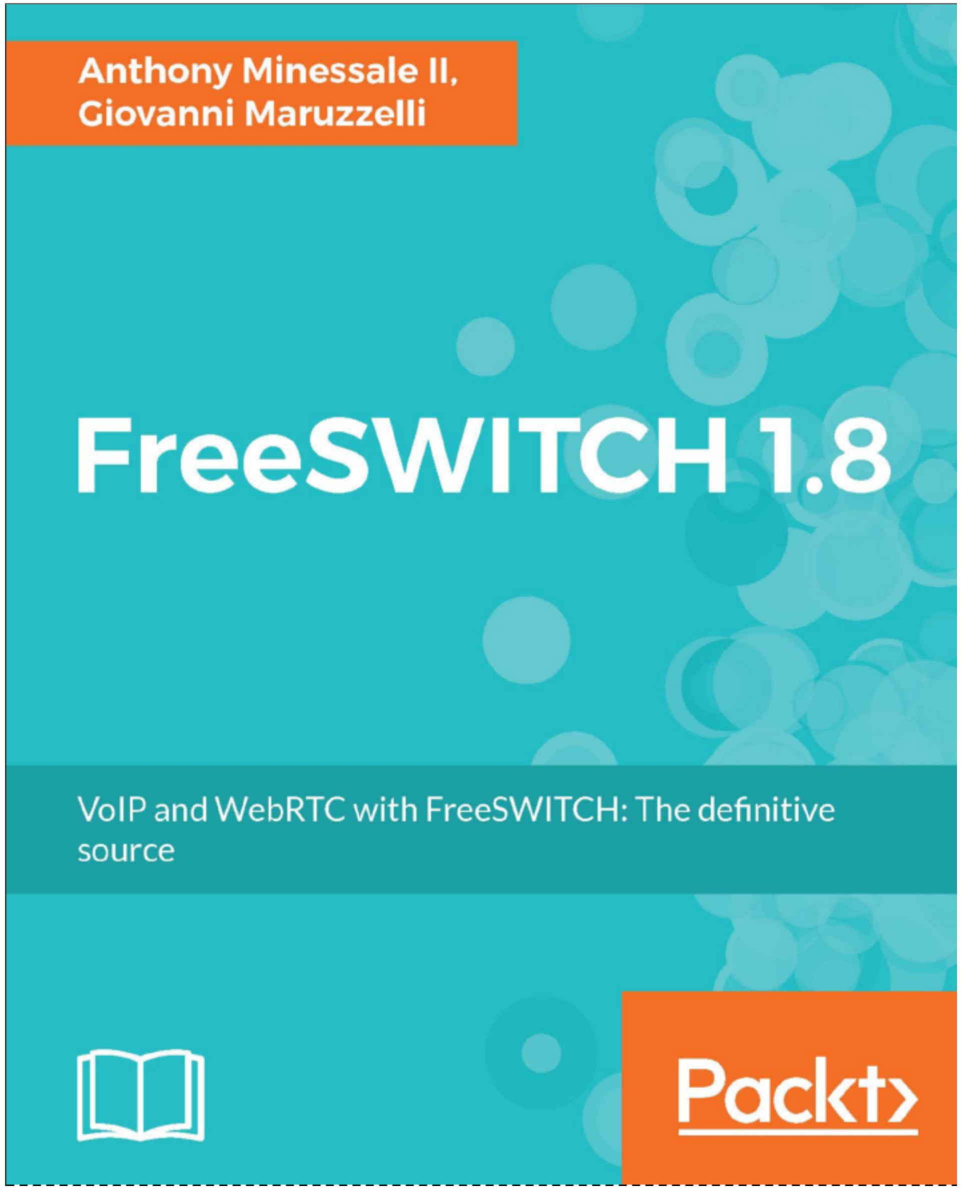




www.packtpub.com

NEW !

(cover FS 1.6/1.8)



NEW !

(cover FS 1.6/1.8)

Thank You

QUESTIONS ?

Giovanni Maruzzelli
gmaruzz@OpenTelecom.IT