



**Alex Goulis**

Using OpenSIPS as your registrar to deliver load balanced class 5 features with Freeswitch

# My experience

- Designing multi-tenant business VoIP platforms since 2009
- Lead developer for Ratetel's Virtual PBX and trunking platform
- First certified OpenSIPS professional
- Ratetel is the US sales partner for OpenSIPS Solutions

# Advantages of using OpenSIPS

- Highly scalable
- Stable code base
- Can handle tens of thousands of registrations
- Central point for presence and billing
- Dynamic routing
- Packet mangling to alter packets for custom purposes
- Highly available

# Advantages of using Freeswitch

- Supports more concurrent calls than most other open source PBXs (asterisk)
- Rich media handling capabilities
- Many different config methods (flat xml, lua, dynamic xml, many others)
- Stable code base and long time affinity with Opensips
- So many class 5 features, even ones you didn't think you needed

# Making it work together

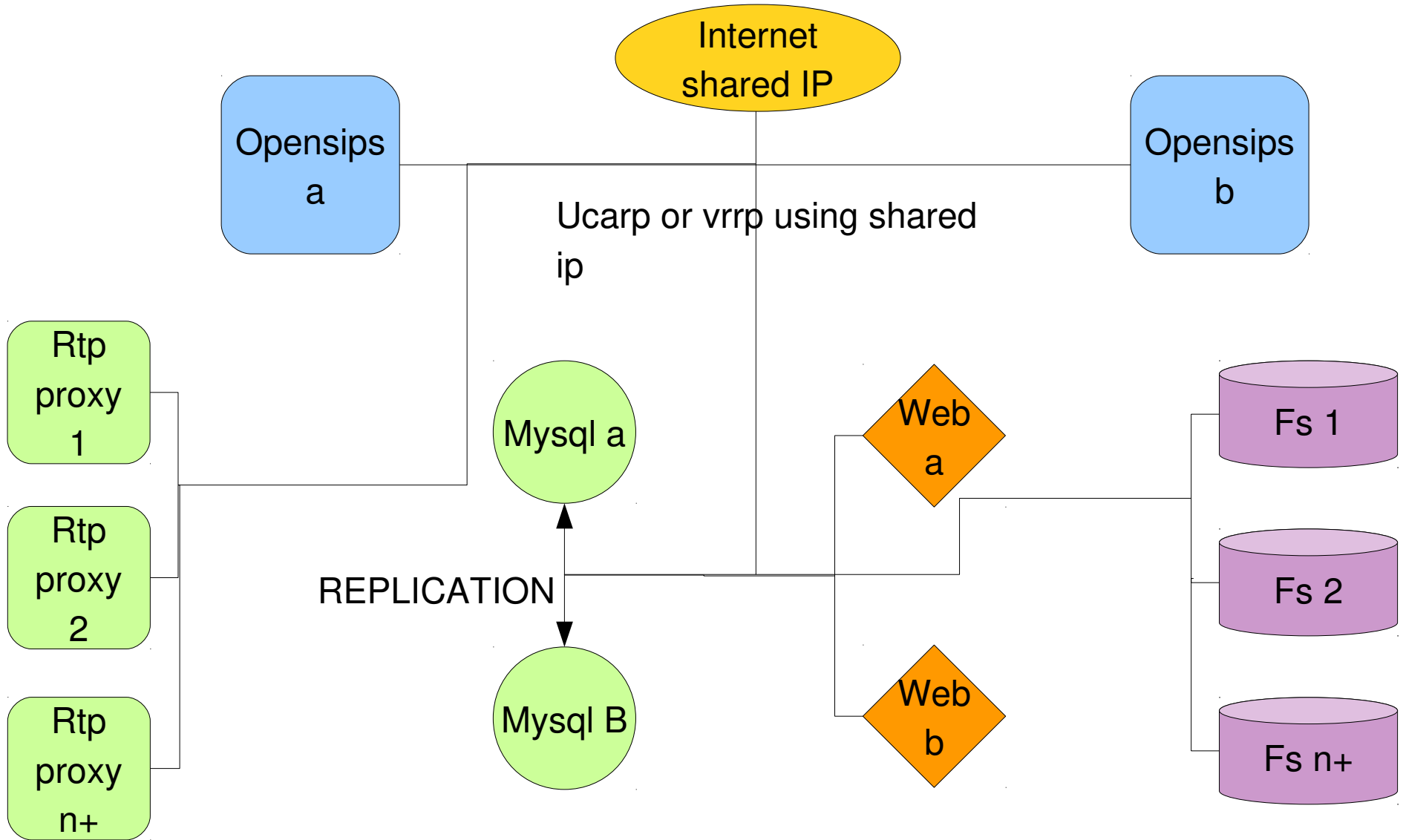
opensips

- Registrations
- Ip Authentication
- Carrier facing
- nat

Freeswitch

- Call routing
- Voicemail
- ivr
- Ring groups
- Queues
- conferences

# Network diagram



# Opensips configuration (multi-tenant)

- Use dispatcher module to catalog your Freeswitch servers
- Will automatically remove servers when down, and add them when available again.
- Dispatcher groups are configured with a primary and secondary server
- Physically shard your tenants across groups to spread resources available
- Problems will occur if traditionally load balanced

# High Availability

- Same dc failover for OpenSIPS using servers ucarp or vrrp
- Automatic removal of fs servers when down
- Automatically includes fs servers when they become available
- HA Proxy for web servers
- Mysql backend for fs configuration



# Registering at Opensips

- Takes the load off of Freeswitch
- Large numbers of registrations will start disrupting normal operations
- Opensips was built to handle registrations
- No presence info at Freeswitch, but handled by Opensips
- Freeswitch servers now act as just feature servers

# Mysql db

- Powerful and free
- Provides backend storage for configurations in both OpenSIPS and Freeswitch
- HA is implemented using Master/Master
- For geo located databases use Percona cluster
- Can add multiple db sources to OpenSIPS to cover loss of one db

# Apache web server

- Most widely used http server
- Built in support for PHP
- Resilient and easy to scale
- Easy to configure and tune

# mod\_xml\_curl

- Freeswitch module to feed realtime xml data to Freeswitch
- Simple to implement using many languages (php, python, java, etc)
- Using fs\_curl can be configured using included db tables or via custom xml hooks
- We will use PHP with xml writer for our custom extensions
- OpenSIPS Summit Austin 2015  
[https://freeswitch.org/stash/projects/FS/repos/freeswitch-contrib/browse/intranetman/PHP/fs\\_curl](https://freeswitch.org/stash/projects/FS/repos/freeswitch-contrib/browse/intranetman/PHP/fs_curl)

Table	Replication	Action	Rows	Type	Collation	Size	Overhead
acl_lists		★ Browse Structure Search Insert Empty Drop	3	MyISAM	latin1_swedish_ci	2.1 K B	-
acl_nodes		★ Browse Structure Search Insert Empty Drop	6	MyISAM	latin1_swedish_ci	2.2 K B	-
cdr		★ Browse Structure Search Insert Empty Drop	869,280	MyISAM	latin1_swedish_ci	199 M B	-
conference_advertise		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
conference_controls		★ Browse Structure Search Insert Empty Drop	35	MyISAM	latin1_swedish_ci	3.9 K B	-
conference_profiles		★ Browse Structure Search Insert Empty Drop	46	MyISAM	latin1_swedish_ci	4.8 K B	-
dialplan		★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	2 K B	-
dialplan_actions		★ Browse Structure Search Insert Empty Drop	196	MyISAM	latin1_swedish_ci	23 K B	-
dialplan_condition		★ Browse Structure Search Insert Empty Drop	4	MyISAM	latin1_swedish_ci	3.2 K B	-
dialplan_context		★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	2 K B	-
dialplan_extension		★ Browse Structure Search Insert Empty Drop	4	MyISAM	latin1_swedish_ci	2.1 K B	-
dialplan_special		★ Browse Structure Search Insert Empty Drop	2	MyISAM	latin1_swedish_ci	5.1 K B	-
dingaling_profiles		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
dingaling_profile_params		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
dingaling_settings		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
directory		★ Browse Structure Search Insert Empty Drop	2,359	MyISAM	latin1_swedish_ci	75.1 K B	-
directory_domains		★ Browse Structure Search Insert Empty Drop	3	MyISAM	latin1_swedish_ci	2.1 K B	-
directory_gateways		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
directory_gateway_params		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
directory_global_params		★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	2 K B	-
directory_global_vars		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
directory_groups		★ Browse Structure Search Insert Empty Drop	316	MyISAM	latin1_swedish_ci	27.7 K B	-
directory_group_user_map		★ Browse Structure Search Insert Empty Drop	84	MyISAM	latin1_swedish_ci	6.1 K B	78 B
directory_params		★ Browse Structure Search Insert Empty Drop	14,024	MyISAM	latin1_swedish_ci	1.2 M B	-
directory_vars		★ Browse Structure Search Insert Empty Drop	2,359	MyISAM	latin1_swedish_ci	114.9 K B	-
iax_conf		★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	2 K B	-
iax_settings		★ Browse Structure Search Insert Empty Drop	8	MyISAM	latin1_swedish_ci	2.2 K B	-
ivr_conf		★ Browse Structure Search Insert Empty Drop	3	MyISAM	latin1_swedish_ci	3.5 K B	-
ivr_entries		★ Browse Structure Search Insert Empty Drop	10	MyISAM	latin1_swedish_ci	3.4 K B	-
limit_conf		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
limit_data		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
local_stream_conf		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
modless_conf		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-
stream_conf		★ Browse Structure Search Insert Empty Drop	0	MyISAM	latin1_swedish_ci	1 K B	-

# Php xml writer

- Shipped with PHP
- Easy to understand and simple to use
- Builds structured XML for Freeswitch to consume
- Allows you to build real time hooks into your data effortlessly

# Freeswitch call groups

```
$obj -> xmlw -> startElement('section');
```

```
$obj -> xmlw -> writeAttribute('name', 'dialplan');
```

```
$obj -> xmlw -> writeAttribute('description', 'FreeSWITCH Dialplan');
```

```
$obj -> xmlw -> startElement('context');
```

```
$obj -> xmlw -> writeAttribute('name', 'ring_group');
```

```
$obj -> xmlw -> startElement('extension');
```

```
$obj -> xmlw -> startElement('condition');
```

```
$obj -> xmlw -> startElement('action');
```

```
$obj -> xmlw -> writeAttribute('application', 'bridge');
```

```
$obj -> xmlw -> writeAttribute('data', 'sofia/1021%${domain_name},sofia/multi.domain-a.com/1022%${domain_name}');
```

```
$obj -> xmlw -> endElement();
```

```
$obj -> xmlw -> endElement(); //</extension>
```

```
$obj -> xmlw -> endElement(); //</context>
```

```
$obj -> xmlw -> endElement(); //</section>
```

# Freeswitch call groups

bob@opensips  
Calls group A





# Freeswitch voicemail

```
$obj -> xmlw -> startElement('section');
```

```
$obj -> xmlw -> writeAttribute('name', 'dialplan');
```

```
$obj -> xmlw -> writeAttribute('description', 'FreeSWITCH Dialplan');
```

```
$obj -> xmlw -> startElement('context');
```

```
$obj -> xmlw -> writeAttribute('name', 'ring_group');
```

```
$obj -> xmlw -> startElement('extension');
```

```
$obj -> xmlw -> startElement('condition');
```

```
$obj -> xmlw -> startElement('action');
```

```
$obj -> xmlw -> writeAttribute('application', 'voicemail');
```

```
$obj -> xmlw -> writeAttribute('data', 'default ${domain} ${dialed_extension}');
```

```
$obj -> xmlw -> endElement();
```

```
$obj -> xmlw -> endElement(); //</extension>
```

```
$obj -> xmlw -> endElement(); //</context>
```

```
$obj -> xmlw -> endElement(); //</section>
```

# Freeswitch Voicemail challenges

- Loss of MWI as Freeswitch is not aware of the user
- Use `api_hangup_hook` to execute external script to send at the end of the call
- Shared voicemail storage limited (nfs, samba)
- Use a distributed filesystem like gluster to geo distribute
- Save messages to mysql and replicate

# Freeswitch conference

```
$obj -> xmlw -> startElement('section');
```

```
$obj -> xmlw -> writeAttribute('name', 'dialplan');
```

```
$obj -> xmlw -> writeAttribute('description', 'FreeSWITCH Dialplan');
```

```
$obj -> xmlw -> startElement('context');
```

```
$obj -> xmlw -> writeAttribute('name', 'conference');
```

```
$obj -> xmlw -> startElement('extension');
```

```
$obj -> xmlw -> startElement('condition');
```

```
$obj -> xmlw -> startElement('action');
```

```
$obj -> xmlw -> writeAttribute('application', 'conference');
```

```
$obj -> xmlw -> writeAttribute('data', conference_name');
```

```
$obj -> xmlw -> endElement();
```

```
$obj -> xmlw -> endElement(); //</extension>
```

```
$obj -> xmlw -> endElement(); //</context>
```

```
$obj -> xmlw -> endElement(); //</section>
```

# Headers for multi-tenant and call tracking

- Currently, there is no language or protocol to make OpenSIPS and Freeswitch work together
- Both projects working on a common solution
- In the meantime, headers are your best friend.
- Use OpenSIPS to add headers that define customer ID, target servers, and permissions
- Use Freeswitch to add headers on dialplan exit to let OpenSIPS see where the call has been



**Thank You!**

**Alex Goulis**  
**a.g@ratetel.com**

**Questions?**