Index

Dashboard.................................................................4
Agents..................................................................................8
SIP.........................................................................................21
Queues...................................................................................26
Trunk.........................................................................................33
Dial Plan..................................................................................41
Audio......................................................................................48
Cally Square...........................................................................50
Analytics..................................................................................51
Realtime...................................................................................72
Voice Mail................................................................................75
Settings - General, Pauses, ChanSpy, BLF, Users.........................78
Settings - Integrations...............................................................92
Annex.......................................................................................95
Introduction

xCALLY Shuttle is the next xCALLY generation software suite, providing many key benefits if you are looking for a professional customer care solution for Asterisk.

Here are few of them:
- Responsive Supervisor web interface HTML5
- Support for 3 type of agent experiences:
  - Windows CTI phone bar
  - External CTI SIP phones
  - WebRTC (experimental)
- Cally Square Drag and Drop IVR full Web HTML5 Asterisk IDE
- Integration with 3rd party software (i.e. Zendesk) using the Shuttle Push Technology
- Advanced reportings
- Advanced call routing management
- Calendar
- Linux CentOS 6.X or 7.X super-easy installer.
**Dashboard**

In the Dashboard section there is an overview on the state of the system, in particular:

- real-time monitoring of agents, calls and queues;
- server and Disk Stats;
- analytics and real-time graphs.

The blocks on the top of the interface represent, in *realtime*:

- the total number of **Waiting Queue Calls** and **Active Queue Calls**, considering all the created Queues;

- the total number of **Completed Queue Calls** and **Abandoned Queue Calls**, considering all the created Queues. Note that the calls which are abandoned for timeout and joinempty are not counted in this value.

- the number of the **connected Browsers**;
Dashboard

- the number of the **Logged Agents** and **Paused Agents**;

- the total number of **Queue Calls** (completed QC + abandoned QC). Note that the calls which are ended in the IVR, without entering in the queue, are not counted in this value;

- the **Average Speed of Answer**, in seconds, considering all the Queues and the **Average Queue Talk Time**, which represents the average time of the call spent in conversation, considering all the Queues.

- the total **Outbound calls** and total **Tiger Dial calls** (Tiger Dial is the the predictive quality dialer module, available in the xCALLY Platinum Plan).

**Global Service Level**

This widget shows the Global Service Level of the Contact Center, considering all the Queues:
- the SL 90%, percentage of calls answered in 10 sec.
- the SL 80%, percentage of calls answered in 20 sec.
- the SL 90%, percentage of calls answered in 30 sec.
- the SL 70%+, percentage of calls answered in more than 30 sec.
**Dashboard**

**Real Time Graphs**
The Dashboard provides new real-time graphs to show dynamically:
- the Waiting and Active calls;
- the Answer Rate, showing the analytics about Completed, Abandoned, and Timeout Calls;
- the SL 90%[10s], SL 80%[20s], SL 70%[30s], SL 70%+[30+s].

![Real Time Graphs](image)

**Monitor**

It contains the list of the created **Queues**, showing for each of them:
- the number of Waiting and In Call calls;
- the number of Completed, Abandoned, and Timeout calls;
- The SLA. Read [here](#) how to set the Queue SLA Threshold.

The other list is dedicated to the logged **Agents** and provides data about their interface, their SIP status, and if they are on Pause.

**CPU and Disk Stats**

This section dedicated to the CPU and Disk Stats, to monitor the Current Use, the Total Memory and the Free Memory.
Dashboard

Missed Queue Calls Summary
This table indicates the reason why a call has been missed:

<table>
<thead>
<tr>
<th>Reason</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIMEOUT</td>
<td>The call was in the queue too long, and timed out. See the timeout parameter.</td>
</tr>
<tr>
<td>FULL</td>
<td>The queue was already full. See the maxlen setting.</td>
</tr>
<tr>
<td>JOINEMPTY</td>
<td>The caller could not join the queue, as there were no queue members to answer the call. See the joinempty setting.</td>
</tr>
<tr>
<td>LEAVEEMPTY</td>
<td>The caller joined the queue, but then all queue members left. See the leavewhenempty setting.</td>
</tr>
<tr>
<td>JOINUNAVAIL</td>
<td>The caller could not join the queue, as there were no queue members available to answer the call. See the joinempty setting.</td>
</tr>
<tr>
<td>LEAVEUNAVAIL</td>
<td>The caller joined the queue, but then all of the queue members became unavailable. See the leavewhenempty setting.</td>
</tr>
<tr>
<td>ABANDON (PORTLET)</td>
<td>The caller abandoned the call.</td>
</tr>
</tbody>
</table>

It’s really important to check frequently this data, in particular the Free Memory of the Disk, because if the disk is full there can be some problems using xCally Shuttle! The audio call recordings can occupy a lot of memory, especially if saved in .wav, so you should use an automatic backup strategy to avoid memory problems.

Reset Dashboard’s Stat
You can reset the Dashboard realtime data clicking on Actions -> Reset Stats.

Language Settings and Logout
On the right of the interface, on the top bar, there are two buttons useful to:
- change the language;
- set the Full Screen mode;
- logout from xCally Shuttle.
Agents

xCALLY Shuttle Agents are called operators. In this last generation system there is an important distinction between the users of the system.

**Agent**: The so-called call center operator, associated with a SIP only.

**User**: The system administrator for the configuration and management.

**SIP**: The actual SIP account (which can also be Web).

In the Agents section you can manage the call center agents.

In this screen there is the list of the agents already created, with their main information.

There are three actions that can be done quickly through the Agents table:

- View and edit the parameters specified in the creation of an Agent by clicking on **Edit**
- View and manage the Queue-Agent association by clicking on **Queues**
- Delete the Agent with a click on the button **Delete**

It’s also possible to copy the table to the clipboard or export it in CSV, Excel and PDF.
Agents

Create a new Agent
To create an Agent click on the button New Agent and fill the form with the following information:

Username: the Username of the Agent that you want to create, necessary for the Agent's login.
Name: Agent's Name. If you want to use the Zendesk Integration, this field must correspond exactly to the Agent's Name in the Zendesk profile.
Email: Agent's Email.
Password: a password, necessary for the Agent's login.
SIP Username: the SIP can be associated to the Agent choosing one SIP that already exists in the dropdown menu. If it's not specified, xCALLY Shuttle automatically creates and associates also a SIP Agent with the same credentials and default parameters, when a new agent is created.
Type: SIP (if the Agent uses external IP Phones or the xCALLY Phone bar) or WebRTC (if the Agent uses the xCally WebRTC interface) or External line (if the Agent uses his own personal phone).

If you select the External line please notice that:
- the username must be the Agent phone number, when you fill in the username field;
- the Ringall strategy will not work, you should choose another kind of strategy;
- the Queue Timeout will not be effective;
- the Agent, to receive calls on his own phone, must be logged into the Shuttle Web Interface, using the username and password credentials which you have to set in this creation form fields.

Voice Mail: Yes -> the Agent will have also his/her Voice Mail account. The account will be automatically created; you can find it in the Voice Mail section.
**Agents**

**Auto Logoff**: Yes -> when the agent exits the browser, he/she automatically logs out from the Agent’s interface and also from his/her Queues. No -> the Agent is logged in until he doesn’t perform the Log out action from his Agent’s interface.

**Module Permissions**: the Administrator can give the access to the Realtime and Monitor section to the Agent. Those permissions are useful in case the agents work with external SIP phones.

If *Realtime* is enabled -> The agent will be able to have a realtime dashboard when they logs-in inside the web interface. If *Monitor* is enabled -> The agent will be able to look at their recorded calls when they logs-in inside the web interface. If *Voice Mail* is enabled -> The agent will be able to look at the Voice Mail messages when they logs-in inside the web interface. If *Contact Management* is enabled [available in the Gold and Platinum plans] -> The agent will be able to look at the Contact List, according to the Contact List settings (see Contacts Management module) and to add customers to the List.
Agents

Edit an Agent

As said before, the information about an Agent can be viewed by clicking on the button Edit in the table row which contains his name.

It's possible to edit the parameters simply by clicking on them, like shown in the image below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Username</td>
<td>John</td>
<td>Agent’s Username, useful for the login in the xCALLY Shuttle system and xCALLY phone bar.</td>
</tr>
<tr>
<td>Name</td>
<td>John</td>
<td>The Name</td>
</tr>
<tr>
<td>Email</td>
<td><a href="mailto:john@callly.com">john@callly.com</a></td>
<td>Agent’s Email</td>
</tr>
<tr>
<td>Phone</td>
<td></td>
<td>Agent’s Phone Number</td>
</tr>
<tr>
<td>Mobile</td>
<td></td>
<td>Agent’s Mobile Number</td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td>Agent’s Address</td>
</tr>
<tr>
<td>ZipCode</td>
<td></td>
<td>Agent’s Zipcode</td>
</tr>
<tr>
<td>City</td>
<td></td>
<td>Agent’s city</td>
</tr>
<tr>
<td>Password</td>
<td></td>
<td>Agent’s Password</td>
</tr>
<tr>
<td>Auto Answer</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>After Call Working</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Auto Answer Timer</td>
<td></td>
<td>When the Auto Answer starts, in milliseconds. If not specified, the Auto Answer will be immediate.</td>
</tr>
<tr>
<td>Modules</td>
<td>Realtime, Monitor, Voice Mail, Contacts Management</td>
<td>Agent’s access to modules (realtime and monitor).</td>
</tr>
<tr>
<td>Sip</td>
<td><a href="mailto:john@callly.com">john@callly.com</a></td>
<td>The SIP associated with the Agent, useful to login into Softphones.</td>
</tr>
<tr>
<td>Show Phonebar Settings</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Auto Logoff</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Change Caller ID on login</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Agents

Agent-Queue Association
Another important feature offered by the Agent section is the Queue Association (a Queue can be created and configured in the section “Queues”).
To associate an Agent to a Queue click on , in his corresponding table row, and select one Queue from the list of the Available Queues.
Here you can also define the Agent Penalty to determine his priority in the queue (low value -> high priority).
If the association has been made correctly, you will see the positive feedback on the Association Results section, on the right of the screen, and the name of the Queue in the Associated Queues list.

To delete an association simply click on the name of the Queue in the Associated Queues list.

The Agent - Queue association can also be managed in realtime.
In the Realtime section the supervisor can quickly add or remove Agents on specific Queues, according to the Contact Center needs.
This is a temporary association: when the agent logs out, the association will be deleted.
3-way Agent Experience

Agents can login into the system and use xCally Shuttle in three different ways, thanks to:

1. the powerful **CTI Windows phone bar**.

2. the Shuttle Web interface and an associated **external SIP standard Phone**.
   http://IP-Shuttle or https://IP-Shuttle

3. the experimental **Beta version of WebRTC client**: the Agent can login inside the Shuttle Web interface with Google Chrome, if a supervisor has created the agent as a Web user (WebRTC is still an ongoing standard, thus you can use it at your own risk).
3-way Agent Experience

1. The CTI Windows phone bar or the Softphone Clients
After the installation of the CTI phone bar (http://www.xcally.com/bar/co/3/0/17/) the Agent can simply login using his credentials.

Dial the extension 600 to listen to the echo test

Here you can find the full guide http://www.xcally.com/shuttle/xCALLY-PhoneBAR-Guide.pdf and video https://www.youtube.com/watch?v=1SYi7pRqoBA about the CTI Windows Phone bar.

It’s possible also to use Softphone Clients, as X-Lite. In the xCALLY Agents section you can see that every Agent is associated to a SIP:
3-way Agent Experience

Under the SIP section you can find the credentials for X-Lite, in particular the SIP name and the Secret. You have to set those parameters in the X-Lite account and specify the Domain.

X-Lite for Windows

X-Lite for Mac
3-way Agent Experience

Remember that, if you want to use X-Lite or other Softphone Clients, it’s fundamental that the agent is also logged on the xCALLY Shuttle Web Agent Dashboard, not only on the Softphone!
3-way Agent Experience

2. External Phones
Agents can also use external IP phones. It’s possible to configure this mode through the following steps:

1. The supervisor has to associate the Agent to the proper SIP username related to the external SIP phone.
   Example: the supervisor has created the agent nick.brown associated to the external SIP phone corresponding to the username digium301.

2. Login using the Shuttle web interface. For example, if your xCALLy Shuttle server has been installed on 192.168.1.10, the agent nick.brown will need to browse and login here:

   https://192.168.1.10
   login: nick.brown
   pwd: his password
3-way Agent Experience

The agent **MUST keeps logged** on to the browser until he/she wants be able to use the external IP phone to manage the customer care calls.

After performing the login, the agent will get the powerful Web Workspace console, where he/she can see:

- the Dashboard with:
  - the realtime analytics about:
    - waiting queue and actual queue calls,
    - completed and abandoned queue calls;
  - his/her associated Queues, with the possibility to manage the pause status and see short Queues Analytics
  - the list of the other Agents which are logged.

If the Administrator gave the relative module permissions, the Agent can also see the Realtime and Monitor section.

Dashboard

Here the agent can have an overview on the state of his Calls and Queues.

The four blocks on the top of the interface represent:

- the total number of calls of Wating Queue Calls and Active Queue Calls;
3-way Agent Experience

- the Completed and the Abandoned Queue Calls;

The section *Monitor* contains the list of the Queues of the Agent, which can be paused, and the analytics about every Agent’s Queue (the number of waiting, in call, completed and abandoned calls).

At last, in the *Internals* there are shown the logged Agents, with their interface name, the call status, the SIP status and a button which starts a call with the selected agent.

**Realtime**

The Realtime module provides realtime data divided into four sections: Details, Queues, Interfaces, Integrations.

**Details**

![Queues Data](image)

**Monitor**

The Agent, according to the Administrator choice, could have also the access to the Monitor section, where he can find the list of the recorded calls, with the possibility to hear and download them.
3-way Agent Experience

3. WebRTC

WebRTC is a real time communication standard still under development. It is supported the best on the Google Chrome browser, however it is still under standardization and development: therefore, use such Beta feature at your own risk (no warranty or any guaranteed service)!

Here you can find the Google Chrome recommended version:

<table>
<thead>
<tr>
<th>Platform</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>chromium MAC</td>
<td><a href="https://provisioning.xcally.com/files/packages/chromium/mac/chrome-mac.zip">https://provisioning.xcally.com/files/packages/chromium/mac/chrome-mac.zip</a></td>
</tr>
<tr>
<td>chromium LINUX</td>
<td><a href="https://provisioning.xcally.com/files/packages/chromium/linux/chrome-linux.zip">https://provisioning.xcally.com/files/packages/chromium/linux/chrome-linux.zip</a></td>
</tr>
</tbody>
</table>

The agent who wants to use the WebRTC interface needs to:

1. Be enabled by the supervisor under the SIP section (Web User flag enabled).

   When an agent is associated to a SIP Web enabled user he/she can work ONLY with WebRTC (xCALLY Windows phone bar or external phones will not work).

2. Login using the Shuttle web interface using ONLY the Google Chrome recommended version via https (DO NOT use http). For example, if your xCALLY Shuttle server has been installed on 192.168.1.10, the agent emily.brown will need to browse and login here:

   https://192.168.1.10
   login: emily.brown
   pwd: her password

After the login the Agent can interact with the Shuttle Web interface, as described at page 13.
xCALLY Shuttle provides a dedicated SIP section to:
- view existing Asterisk SIP
- add new ones
- edit parameters available on Asterisk.

In this screenshot is shown the SIP section, with the list of the SIP already created and their main information, in particular the Type, SIP User or Web User.

There are two actions that can be done quickly through the SIP table:
- View and edit the parameters specified in the creation of the SIP by clicking on Edit
- Delete the SIP

It’s also possible to copy the table to the clipboard or export it in CSV, Excel and PDF.
**SIP**

*Create a new SIP*

To create a SIP click on the button **New SIP** and fill the form with the following information:

- **Username**: the name of the SIP that is being created, it can be either numeric or textual.
- **Password**: authentication password of the SIP
- **Confirm Password**: the confirmation of the authentication password
- **Caller-ID**: defines the identifier, when there are no other information available
- **Type**:
  - SIP
  - WebRTC: this will be used with a SIP WebRTC interface (for example the one offered in the xCALLY Shuttle Agent’s view).
**SIP**

*Edit a SIP*

The information about a SIP can be viewed by clicking on on the table row which contains the SIP of interest.

To edit one parameter, simply click on its Value.

---

**General Settings**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>the name of the SIP</td>
</tr>
<tr>
<td>Context</td>
<td>the context of the dialplan</td>
</tr>
<tr>
<td>Secret</td>
<td>authentication password of the SIP</td>
</tr>
<tr>
<td>Transport</td>
<td>set the default transports. The order determines the primary default transport.</td>
</tr>
<tr>
<td>Host</td>
<td>is the domain or host name for the SIP server.</td>
</tr>
<tr>
<td>Nat</td>
<td>this parameter specifies that the SIP device is behind a NAT (Network Address Translator)</td>
</tr>
<tr>
<td>Type</td>
<td>User: used to authenticate incoming Peer: for outgoing calls Friend: covers both characteristics of the above.</td>
</tr>
<tr>
<td>Caller-ID</td>
<td>defines the identifier, when there are no other information available</td>
</tr>
<tr>
<td>DTMF Mode</td>
<td>how DTMF (Dual-Tone Multi-Frequency) are sent: RFC2833: the default mode, the DTMF are sent with RTP but outside the audio stream. INBAND: The DTMF is sent in audio stream of the current conversation, becoming audible from the speakers. Requires a high CPU load. INFO: Although this method is very reliable, it is not supported by all PBX devices and many SIP Trunk.</td>
</tr>
<tr>
<td>Language</td>
<td>default language used by any Playback()/Background().</td>
</tr>
</tbody>
</table>
## SIP

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualify</td>
<td>to determine when the SIP is achievable</td>
</tr>
<tr>
<td>Allow Codec</td>
<td>with this parameter you can specify the codecs enabled for SIP; some of them are pre-selected by default</td>
</tr>
<tr>
<td>Category</td>
<td>WebRTC or SIP</td>
</tr>
<tr>
<td>Qualify freq.</td>
<td>It represents the timeout after a packet is sent before we consider the peer to be unreachable.</td>
</tr>
<tr>
<td>Callgroup</td>
<td>The SIP can be associated to one or several callgroups. The Callgroup settings are useful to letting someone else answer a call. You have to insert in this field the <strong>number of the callgroup</strong>, i.e. 1</td>
</tr>
<tr>
<td>Pickup Callgroup</td>
<td>The SIP can pickup the incoming calls of the Callgroups which are defined here, by calling *8 on the phone. You have to insert in this field the <strong>numbers relative to the Pickup Callgroups</strong> that you want to set for this SIP, i.e. 1-5,9</td>
</tr>
<tr>
<td>Description</td>
<td>It’s the description associated with the SIP, useful to describe in words his context.</td>
</tr>
</tbody>
</table>
### Advanced Settings

#### Limit on Peers
- **Description:** to define the call limits of a “peers” SIP type.
- **Details:** This can improve the status notification if you are using a “friend” SIP type for incoming calls. It also allows to evaluate incoming and outgoing calls; if it is set to yes value, Asterisk uses a counter for both incoming calls and outgoing.

#### Call Counter
- **Description:** If enabled, this parameter allows Asterisk to provide useful information about the status of SIP devices.

#### Can Reinvite
- **Description:** thanks to this parameter two devices can establish directly the SIP RTP connection (Real Time Protocol). The result is to minimize the use of resources needed to establish the full-duplex communication.

#### Direct Media
- **Description:** enabling this parameter, Asterisk tries to drive traffic between the caller and the callee. Not all devices support this feature.

#### Amflags
- **Description:** AMA (Automated Message Accounting) allows to classify the SIP calls in the CDR for billing of calls:
  - Default: sets the default system with the value ‘3’ in the CDR
  - Omit: does not record calls (setting ‘1’)
  - Billing: make calls setting ‘2’ in the CDR and classifying them as billable
  - Documentation: classifies the call as documentation, setting ‘3’ in the CDR

#### Subscribe context
- **Description:** this parameter allows you to specify a context for subscriptions. If it is not set, is the same of that specified in the parameter “context”.

#### Busy Level
- **Description:** this numeric parameter specifies the number of calls in progress in order to categorize the SIP device as busy.

#### RTP Timeout
- **Description:** it allows you to automatically terminate the call if you don’t detect RTP traffic within the specified number of seconds.

#### RTP Hold Timeout
- **Description:** this parameter is more restrictive than the previous one because, unlike RTP Timeout, if the call is put on hold (pause), at the end of this time, the call is terminated.
xCALLY Shuttle provides also a section dedicated to the Queues, useful to:
- create or remove Queues
- view the existing Queues and edit their parameters.

In this screenshot is shown the list of the Queues already created and their main information, in particular the type of strategy chosen to distribute calls.

There are three actions that can be done quickly through the Queues table:
- View and edit the parameters specified in the creation of a Queue by clicking on
- View and manage the Queue-Agent association by clicking on
- Delete the Queue with a click on the button

It’s also possible to copy the table to the clipboard or export it in CSV, Excel and PDF.
Create a new Queue

To create a Queue click on the button + New Queue and fill the form with the following information:

**Reference**: 4021

**Name**: Type a name (E.g.: Support, Sales) and check its availability.

**Strategy**: Ringall

**Name**: the name of the Queue.

**Strategy**: you can define a distribution policy for inbound calls, selecting one of the following strategies:

- Ringall: it contacts all Agents until one answers;
- RRMemory: one of the most used round robin memory. It resumes from the agent succeeding the one contacted during last call;
- Least recent: it contacts the Agent that has recently been less involved in the queue;
- Fewest calls: it contacts the Agent who has recently answered to the minimum number of calls in the queue;
- Random: it randomly routes calls to agents;
- Linear: it contacts the agents according to their order of login;
- Wrandom: it randomly routes calls to agents.
**Edit a Queue**

The information about a Queue can be viewed by clicking on Edit on the table row which contains the Queue of interest.

To edit one parameter, simply click on its Value.

**General Settings**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Support</td>
<td>Name</td>
</tr>
<tr>
<td>Music on Hold</td>
<td>default</td>
<td>Music on Hold</td>
</tr>
<tr>
<td>Strategy</td>
<td>repeat</td>
<td>Strategy</td>
</tr>
<tr>
<td>Timeout</td>
<td>0</td>
<td>Timeout</td>
</tr>
<tr>
<td>Max Len</td>
<td>0</td>
<td>Max Len</td>
</tr>
<tr>
<td>Retry</td>
<td>0</td>
<td>Retry</td>
</tr>
<tr>
<td>Wrapup Time</td>
<td>0</td>
<td>Wrapup Time</td>
</tr>
<tr>
<td>Weight</td>
<td>0</td>
<td>Weight</td>
</tr>
<tr>
<td>Joinempty</td>
<td>yes</td>
<td>Joinempty</td>
</tr>
<tr>
<td>Leavewhenempty</td>
<td>no</td>
<td>Leavewhenempty</td>
</tr>
<tr>
<td>Announce</td>
<td>Empty</td>
<td>Announce</td>
</tr>
<tr>
<td>Announce freq.</td>
<td>0</td>
<td>Announce freq.</td>
</tr>
<tr>
<td>Announce Round [s]</td>
<td>Empty</td>
<td>Announce Round [s]</td>
</tr>
<tr>
<td>Announce HoldTime</td>
<td>no</td>
<td>Announce HoldTime</td>
</tr>
<tr>
<td>Periodic Announce</td>
<td>Empty</td>
<td>Periodic Announce</td>
</tr>
<tr>
<td>Periodic Ann. freq.</td>
<td>0</td>
<td>Periodic Ann. freq.</td>
</tr>
<tr>
<td>Wrapup Time</td>
<td>0</td>
<td>Wrapup Time</td>
</tr>
</tbody>
</table>

In this section you can define the following parameters:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Music on Hold</td>
<td>it allows the choice of different tunes (previously loaded in the audio section) to entertain the caller during the waiting time. The default one is loaded into the Music on Hold Asterisk configurations.</td>
</tr>
<tr>
<td>Strategy</td>
<td>the distribution policy for inbound calls</td>
</tr>
<tr>
<td>Timeout</td>
<td>the time interval (in seconds) over which the agent is considered unavailable to take charge of a call.</td>
</tr>
<tr>
<td>Max Len</td>
<td>it specifies the maximum number of calls on hold. If equal to 0, this number is unlimited.</td>
</tr>
<tr>
<td>Retry</td>
<td>time interval (in seconds) expected from the system before re-contact all the agents in the queue in object.</td>
</tr>
<tr>
<td>Wrapup Time</td>
<td>minimum time interval (in seconds) to contact an agent between two different calls.</td>
</tr>
<tr>
<td>Weight</td>
<td>it defines a priority queue. If the agents are multi-skilled, inbound calls will be routed preferentially to the queue with the highest priority.</td>
</tr>
</tbody>
</table>
## Queues

<table>
<thead>
<tr>
<th>Setting</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Joinempty</strong></td>
<td>this setting check if callers can join an empty queue empty, without available agents.</td>
</tr>
<tr>
<td></td>
<td>In this case there are possible choices:</td>
</tr>
<tr>
<td></td>
<td><strong>yes</strong>: callers can enter on an empty queue or on a queue where agents are not available</td>
</tr>
<tr>
<td></td>
<td><strong>no</strong>: callers can’t get on an empty queue</td>
</tr>
<tr>
<td></td>
<td><strong>strict</strong>: callers can’t enter on an empty queue or on a queue where agents are not available</td>
</tr>
<tr>
<td></td>
<td><strong>loose</strong>: as strict, but the agents on pause on a queue are considered as available</td>
</tr>
<tr>
<td></td>
<td><strong>paused</strong>: an agent is not considered available when paused</td>
</tr>
<tr>
<td></td>
<td><strong>penalty</strong>: an agent is not considered available if his penalty is less than QUEUE_MAX_PENALTY</td>
</tr>
<tr>
<td></td>
<td><strong>in use</strong>: an agent is not considered available when he is calling</td>
</tr>
<tr>
<td></td>
<td><strong>ringing</strong>: an agent is not considered available if his phone is ringing</td>
</tr>
<tr>
<td></td>
<td><strong>unavailable</strong>: if the agent is part of the queue in question but it is not logged in then it is not</td>
</tr>
<tr>
<td></td>
<td>considered to be available</td>
</tr>
<tr>
<td></td>
<td><strong>invalid</strong>: an agent is not considered available if the device status is “invalid”</td>
</tr>
<tr>
<td></td>
<td><strong>unknown</strong>: it not considers a member as available if it is unable to determine the current state of</td>
</tr>
<tr>
<td></td>
<td>the device agents</td>
</tr>
<tr>
<td></td>
<td><strong>wrapup</strong>: an agent is not considered available if it is currently in its wrapuptime after a call</td>
</tr>
</tbody>
</table>

| **Leavewhenempty** | it works on calls already in the queue.                                                                 |
|                   | **yes**: if the queue is empty or agents are not available, the caller leaves the queue.                |
|                   | **no**: callers remain queued even if the queue is empty or agents are not available.                   |

| **Announce**      | you can choose different tunes (previously loaded in the Audio section) that will be heard by the      |
|                   | agent before talking with the customer. An agent that operates in multiple queues is able to           |
|                   | quickly understand what kind of call is and answer appropriately.                                     |

| **Announce freq.**| it sets how often (in seconds) a caller will hear an announcement that indicates its position in the   |
|                  | queue or the estimated waiting time. If it’s equal to 0, this function is disabled.                   |

| **Announce Round [s]** | it’s the level of rounding for the announcements of waiting times. If 0: minutes are considered and    |
|                        | announced, without seconds. The possible values are 0, 1, 5, 10, 15, 20, 30. For example, when the    |
|                        | value is set to 30, the waiting time 2:34 will be rounded to 2:30.                                    |

| **Announce Hold Time** | it indicates whether the estimated wait time will be announced after the information of the position   |
|                       | in the queue. The possible values are “yes”, “no” or “once”                                          |

| **Periodic Announce** | you can choose different tunes (previously loaded in the Audio section) to be played to the caller    |
|                      | periodically.                                                                                        |

| **Periodic Announce freq.** | it sets how often (in seconds) a caller will hear a periodic announcement.                            |

---

**Note:** The system won’t play the periodic announcement and the position of the queue if there are free agents.

If all the agents are busy on the phone, then it will play the periodic announcements.

The announcements and the position will be played in case of available free agents, just after the Queue Timeout (see please Timeout parameter inside the Queue configuration).
Advanced Settings

In this section you can define the following parameters:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autopause</td>
<td>No</td>
<td>It indicates if an Agent that wasn’t able to answer is automatically paused or not. Yes: it sets Autopause on the selected queue All: it sets Autopause on all Queues</td>
</tr>
<tr>
<td>Ringinuse</td>
<td>No</td>
<td>It indicates that queue’s calls have to be forwarded to the agents which have devices in use.</td>
</tr>
<tr>
<td>Member delay</td>
<td>0</td>
<td>It indicates the interval of silence (in seconds) that the caller hears before being connected to an agent.</td>
</tr>
<tr>
<td>Timeout restart</td>
<td>0</td>
<td>It indicates if the response timeout of an agent is reset after a busy signal or congestion. This can be useful for agents who are authorized to refuse calls.</td>
</tr>
<tr>
<td>Service level</td>
<td>0</td>
<td>It sets the threshold of service level; for example it sets the maximum wait time for callers. This is very useful for statistical analysis</td>
</tr>
<tr>
<td>Recording</td>
<td>inactive</td>
<td>The recording format: .wav or .mp3</td>
</tr>
<tr>
<td>Events when called</td>
<td>Yes</td>
<td>It sends events to the Asterisk Manager about the states of the agents before, during and after the call.</td>
</tr>
<tr>
<td>Events member status</td>
<td>Yes</td>
<td>It sends events to the Asterisk Manager about the states of the agents in the queue (default: yes).</td>
</tr>
<tr>
<td>Report hold time</td>
<td>No</td>
<td>It indicates the possibility to announce the caller’s waiting time to the agent. The possible values are yes and no.</td>
</tr>
<tr>
<td>Set interface var</td>
<td>Yes</td>
<td>If set to yes, just prior to the caller being bridged with a queue member, the MEMBERINTERFACE variable will be set with the interface name (eg. Agent/1234) of the queue member that was chosen and is now connected to be bridged with the caller.</td>
</tr>
<tr>
<td>Context</td>
<td>Empty</td>
<td>The context of the dialplan</td>
</tr>
<tr>
<td>CTI URL v3</td>
<td>Empty</td>
<td></td>
</tr>
<tr>
<td>CTI Event v3</td>
<td>Up</td>
<td>The color of the popup displayed when receiving a call on the xCALL PhoneBar</td>
</tr>
<tr>
<td>CTI Application v3</td>
<td>NA</td>
<td></td>
</tr>
<tr>
<td>Phonebar Popup Color</td>
<td>Blue</td>
<td></td>
</tr>
<tr>
<td>Smart ACW</td>
<td>No</td>
<td>If enabled, it puts automatically the Agent on pause (AFTERCALL) on all his queues, when he ends a call. The Agent won’t receive calls until he resumes his queues.</td>
</tr>
<tr>
<td>Smart ACW Timeout</td>
<td>30</td>
<td>The after call working pause duration</td>
</tr>
</tbody>
</table>

In this section you can define the following parameters:
**Smart After Call Working feature**

When an agent completes a call on a queue where the ACW is set (e.g., with a timeout of 20 sec), the agent goes to ACW status and cannot receive new calls. When the timeout expires, the agent can automatically return to the ready status. In any moment before the timeout expiration, the agent can manually return to ready by pressing the pause button of the phonebar.

In a multiskill environment, the agent status log will show ACW status only for the defined queue, on the other queues the agent status will be “UNAVAILABLE BY ACW”. Thanks to this different log management, it is possible to report about the ACW time for each queue.

Example, the agent John Doe belongs to the queues Sales and Support. These queues have the ACW set to ON with different timeouts (15 and 20 sec). The detailed report for agents status show the correct information:

In this example, the agent John Doe has taken two calls, one on the Sales queue and one on the Support queue. At the end of the first call, he went to ACW status for 15 seconds only for the related queue (Sales) while his status went to UNAVAILABLE BY ACW on the other queue (Support). Same thing for the second call on the Support queue.

NOTE: to use this feature the option After Call Work under the agents settings must be disabled. Adding the variable HANGUP-HANDLER is not needed anymore and the Asterisk settings (queues.conf) “shared_last_call” must be set to NO.
Queues

Queue-Agent Association

Another important feature offered by the Queue section is the Agent Association (an Agent can be created and configured in the section “Agents”).

To associate a Queue to an Agent click on **Agents**, in his corresponding table row, and select one Agent from the list of the Available Agents.

Here you can also define the Agent Penalty to determine his priority in the queue (low value -> high priority).

If the association has been made correctly, you will see the positive feedback on the Association Results section, on the right of the screen, and the name of the selected Agent in the Associated Agents list.

To delete an association simply click on the name of the Agent of interest in the Associated Agents list.

The Agent - Queue association can also be managed in realtime.

**In the Realtime section** the supervisor can quickly add or remove Agents on specific Queues, according to the Contact Center needs.

This is a temporary association: when the agent logs out, the association will be deleted.
Trunk

This xCally Shuttle Section is dedicated to the managing of Trunks, useful for the creation of the inbound external rules and outbound rules.

In this screenshot there is the list of the already created Trunks with their main information, like the Host IP.

There are two actions that can be done quickly through the Trunk table:
- View and edit the parameters specified in the creation of a Trunk by clicking on Edit
- Delete the Trunk with a click on the button Delete

Create a new Trunk

To create a Trunk click on the button + New Trunk and fill the form with the following data:

Create Trunk

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>1005</td>
</tr>
<tr>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>Host</td>
<td></td>
</tr>
<tr>
<td>Password</td>
<td></td>
</tr>
<tr>
<td>Confirm Password</td>
<td></td>
</tr>
<tr>
<td>Default User</td>
<td></td>
</tr>
<tr>
<td>Registry</td>
<td>username:password@host</td>
</tr>
</tbody>
</table>
**Trunk**

**Name:** Trunk name

**Host:** it must be an *IP Address*

**Password:** Trunk Secret

**Default User:** your SIP Username

**Registry:** it must be compiled in this way: `username:password@host`

**Edit a Trunk**

The information about a Trunk can be viewed by clicking on [Edit] on the table row which contains the Trunk of interest.

To edit one parameter, simply click on its Value.

**General Settings**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>1994</td>
<td>Reference</td>
</tr>
<tr>
<td>Name</td>
<td>sitaela</td>
<td>Name</td>
</tr>
<tr>
<td>Secret</td>
<td></td>
<td>Secret</td>
</tr>
<tr>
<td>Caller-ID</td>
<td><strong>&lt;&gt;</strong></td>
<td>Caller-ID</td>
</tr>
<tr>
<td>Call limit</td>
<td>20</td>
<td>Call limit</td>
</tr>
<tr>
<td>Default User</td>
<td></td>
<td>Default User</td>
</tr>
<tr>
<td>Host</td>
<td>voip.autela.it</td>
<td>Host</td>
</tr>
<tr>
<td>Type</td>
<td>friend</td>
<td>Type</td>
</tr>
<tr>
<td>DTMF mode</td>
<td>rtc3f3</td>
<td>DTMF mode</td>
</tr>
<tr>
<td>Nat</td>
<td>force_report.com</td>
<td>Nat</td>
</tr>
<tr>
<td>Qualify</td>
<td>no</td>
<td>Qualify</td>
</tr>
<tr>
<td>Allow codec</td>
<td>727, 727</td>
<td>Allow codec</td>
</tr>
<tr>
<td>Can reinvite</td>
<td>no</td>
<td>Can reinvite</td>
</tr>
<tr>
<td>Insecure</td>
<td>port443</td>
<td>Insecure</td>
</tr>
</tbody>
</table>

In this section you can define the following parameters:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secret</td>
<td>Trunk Secret</td>
</tr>
<tr>
<td>Caller-ID</td>
<td>defines the identifier, when there are no other information available</td>
</tr>
<tr>
<td>Call limit</td>
<td>number of simultaneous calls through this user/peer</td>
</tr>
<tr>
<td>Default User</td>
<td>your SIP Username</td>
</tr>
<tr>
<td>Host</td>
<td>IP address</td>
</tr>
<tr>
<td>Type</td>
<td>User: used to authenticate incoming Peer: for outgoing calls Friend: covers both characteristics of the above</td>
</tr>
</tbody>
</table>
Trunk

| DTMF Mode | how DTMF (Dual-Tone Multi-Frequency) are sent:  
RFC2833: the default mode, the DTMF are sent with RTP but outside the audio stream.  
INBAND: The DTMF is sent in audio stream of the current conversation, becoming audible from the speakers. Requires a high CPU load.  
INFO: Although this method is very reliable, it is not supported by all PBX devices and many SIP Trunk. |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Nat</td>
<td>this parameter specifies that the device is behind a NAT (Network Address Translator)</td>
</tr>
<tr>
<td>Qualify</td>
<td>defines the identifier, when there are no other information available</td>
</tr>
<tr>
<td>Allow Codec</td>
<td>with this parameter you can specify the codecs enabled for SIP; some of them are pre-selected by default</td>
</tr>
<tr>
<td>Can reinvite</td>
<td>thanks to this parameter two devices can establish directly the SIP RTP connection (Real Time Protocol). The result is to minimize the use of resources needed to establish the full-duplex communication.</td>
</tr>
<tr>
<td>Insecure</td>
<td>Specifies how to handle connections with peers.</td>
</tr>
</tbody>
</table>

Advanced Settings

In this section you can define the following parameters:

| Limit on Peers | yes | no, if set to yes use only the peer call counter for both incoming and outgoing calls |
| Call Counter | If enabled, this parameter allows Asterisk to provide useful information about the status of SIP devices. |
| From Domain | <domain>, it sets default From: domain in SIP messages when acting as a SIP ua (client) |
| From User | <from_ID>, it specify user to put in “from” instead of $CALLERID(number) (overrides the callerid) when placing calls _to_ peer (another SIP proxy). Valid only for type=peer |
| Outbound Proxy | IP address or DNS SRV name (excluding the _sip._udp prefix) : SRV name, hostname, or IP address of the outbound SIP Proxy. Valid only in [general] section and type=peer. |
| Userreqphone | yes | no, it indicates whether to add a “user=phone” to the URI. Default no. |
## Trunk

| **Trust rpid** | yes | no, if Remote-Party-ID SIP header should be trusted. Default no. |
| **Send rpid** | yes | no, if a Remote-Party-ID SIP header should be sent. Default no. |
| **Port** | <portno>, Default SIP port of peer. |
| **Language** | A language code defined in indications.conf- defines language for prompts |
| **Registry** | callerID:secret@host/callerID |

Click here for the xCALLY - Twilio SIP Trunk guide.
Trunk

Scenario 1

Example
**Trunk**

**Scenario 2**

![Diagram of xCALLY Shuttle and Media Gateway]

**Example - Gateway Inalp-Patton**

**General Settings**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td>1019</td>
<td>Reference</td>
</tr>
<tr>
<td>Name</td>
<td>trunk2</td>
<td>Name</td>
</tr>
<tr>
<td>Secret</td>
<td></td>
<td>Secret</td>
</tr>
<tr>
<td>Caller-ID</td>
<td></td>
<td>Caller-ID</td>
</tr>
<tr>
<td>Call limit</td>
<td>1000</td>
<td>Call limit</td>
</tr>
<tr>
<td>Default User</td>
<td>trunk2</td>
<td>Default User</td>
</tr>
<tr>
<td>Host</td>
<td>dynamic</td>
<td>Host</td>
</tr>
<tr>
<td>Type</td>
<td>Friend</td>
<td>Type</td>
</tr>
<tr>
<td>DTMF mode</td>
<td>rfc2833</td>
<td>DTMF mode</td>
</tr>
<tr>
<td>Nat</td>
<td>force, rport, comedia</td>
<td>Nat</td>
</tr>
<tr>
<td>Qualify</td>
<td>yes</td>
<td>Qualify</td>
</tr>
<tr>
<td>Allow codec</td>
<td>alaw, ulaw, gsm</td>
<td>Allow codec</td>
</tr>
<tr>
<td>Can replace</td>
<td>no</td>
<td>Can replace</td>
</tr>
<tr>
<td>Insecure</td>
<td>port, invite</td>
<td>Insecure</td>
</tr>
</tbody>
</table>
Trunk

Advanced Settings

Example - Gateway Digium E1
Trunk

**Advanced Settings**

For most part of deployments the default parameters in this advanced configuration are fine. Please be careful and edit the following parameters just with skilled personnel support.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limit on Peers</td>
<td>Yes</td>
<td>To define the call limits of a peers SIP type.</td>
</tr>
<tr>
<td>Call counter</td>
<td>Yes</td>
<td>If enabled, this parameter allows Asterisk to provide useful information about the status of SIP devices.</td>
</tr>
<tr>
<td>Directmedia</td>
<td>No</td>
<td>Enabling this parameter, Asterisk tries to drive traffic between the caller and the callee.</td>
</tr>
<tr>
<td>From Domain</td>
<td>Empty</td>
<td>It sets default Fromdomain in SIP messages when acting as a SIP ua (client)</td>
</tr>
<tr>
<td>From User</td>
<td>trunk1</td>
<td>Which user to put instead of callerid when placing calls_to_peer (another SIP proxy). Valid only for type-peer</td>
</tr>
<tr>
<td>Outbound Proxy</td>
<td>Empty</td>
<td>IP address or DNS SRV name (excluding the _sip._udp prefix): SRV name, hostname, or IP address of the outbound SIP Proxy. Valid only in [general] section and type-peer.</td>
</tr>
<tr>
<td>Userephone</td>
<td>No</td>
<td>It indicates whether to add a userephone to the URI. Default no.</td>
</tr>
<tr>
<td>Trust rpid</td>
<td>No</td>
<td>It indicates if Remote-Party-ID SIP header should be trusted. Default no.</td>
</tr>
<tr>
<td>Send rpid</td>
<td>No</td>
<td>It indicates if a Remote-Party-ID SIP header should be sent. Default no.</td>
</tr>
<tr>
<td>Port</td>
<td>5060</td>
<td>The default SIP port of peer</td>
</tr>
<tr>
<td>Language</td>
<td>English</td>
<td>The default language</td>
</tr>
<tr>
<td>Registry</td>
<td>Empty</td>
<td>callerID@secret@host/callerID</td>
</tr>
</tbody>
</table>

**Edit SIP Endpoint “xCALLY-Trunk”**

- **Main Endpoint Settings**
  - Enable Advanced Options
  - Name: xCALLY-Trunk
  - Username: trunk1
  - Password: your-trunk-password
  - Registration: This gateway registers with the endpoint
  - Destination Number: 100
  - Hostname or IP Address: 192.168.2.5
  - Use UDP: Yes
  - Use TCP: No
  - Use TLS: No
  - NAT Traversal: Yes

Your xCALLY SHUTTLE IP here
Dial Plan

The Dial Plan module is dedicated to the managing of the logical routing of communications. This section allows to define rules for the outbound and inbound calls routing. In addition, for each rule, it’s possible to define a specific behavior according to different moments of the day or of the week.

The Dial Plan Menu is divided into three sections, as shown in the image:
- Inbound Internal;
- Inbound External;
- Outbound.

In each of these sections you can manage the existing Rules and create new ones.

Through the Rules table it’s possible to:
- Change the parameters specified in the creation of the rule by clicking on Edit
- Delete the rule with a click on the button Delete

You can also copy the table to the clipboard or export it in CSV, Excel and PDF.
**Dial Plan**

*Create a new Inbound Rule*

To create an Inbound Rule click on the button **+ New Route** and fill the form with the following information:

- **Name:**
- **DID:**
  - Geographic number or internal numbering
  - Type a DID number (E.g: 6000, 6010) and check its availability.
- **Description:**

In addition to the creation of the Route it is necessary to specify the Rule settings, through the **Edit** button:

- In the **General Setting** section you can change the values of DID, Name and Description;
- In the **Advanced section** you can associate an Application to the Route through this button **+ New Application**.
For each application, in the creation form, it’s possible to define in which time interval the rule it’s valid and its priority over other applications, like shown in the image below.

There are different types of applications which can be associated to the route:

1. **Queue Application**
   - Queue Name
   - Queue Options
   - URL
   - Announce Override
   - Timeout
   - AGI
   - Macro
   - Gosub
   - Rule
   - Position

2. **Playback Application**
   - Audio File Path
   - Options (Skip, No Answer, J, Say)

   - Type/ID: phone call (eg SIP/1003)
   - Timeout: Timeout before the application terminates
   - Options: Refer to the Asterisk’s wiki
   - URL
**Dial Plan**

   - *Agi Script*: set the AGI’s name (Ex. agi :/ / 127.0.0.1/square, project = 3)
   - *Agi parameter 1 .. 6*: Possible parameters to AGI.

5. **Application IVR Cally-Square**
   It refers to a Cally Square project (previously loaded in the Cally Square section).

6. **Goto**
   - *Context*
   - *Extension*
   - *Priority*

7. **Hangup**
   - *Cause Code*

8. **Set**
   - *Variable*
   - *Value*

9. **Custom**
   - *Application Name*
   - *Params*

In the applications table it’s possible to:
   - Edit the application, clicking on "Edit"
   - Remove it, clicking on "Delete"
   - Set its priority over other applications, by drag and drop.
Dial Plan

Create a new Outbound Rule

To create an Outbound Rule click on the button and fill the form with the following information:

Name: rule’s name.

Pattern: it indicates the number to be filtered. It may be a fixed pattern or a pattern which represents a series of numbers. The operators most useful are “X”, which indicates a number from 0 to 9, and “.”, which indicates 0 or more numbers from 0 to 9. So, for example, “3X” means from 30 to 39 and “12X.” means all numbers starting with 12 plus at least one other number, like 124, 120, but not only 12.

Prefix: the prefix which is necessary to consider before dialing the call number.

Cut Digits: the digits that should be cut to the number before passing the call to the provider.

Campaign Log: it indicates the possibility to log all outgoing calls from the rule in question with a campaign name (previously configured in the Settings section).

Recording Call: it indicates the possibility to record outbound calls on the rule in question (in WAV or GSM format).

Description:

In addition to the creation form it is necessary to specify other rule settings, as described on the next page.
**Dial Plan**

*Edit an Outbound Rule*

The information about an Outbound Rule can be viewed and edited by clicking on the button.

In the **General Settings** it’s possible to edit the parameters set in the creation form, while in the **Route Section** you can add routes to the rule in question.

For each route, it’s possible to:
- Edit it, clicking on **Edit**
- Remove it, clicking on **Delete**
- Set its priority over other routes, by drag and drop.

To add a route click on the button **New Route** and then fill the form with the following parameters:
- **Suffix**: it is required by the provider before passing the call to the provider itself.
- **Trunk**: the output channel (which is previously configured in the Trunk section)
- **CallerID**: ID of the caller. In this case it’s the number to show to the called person.
- **Context**: the context name on which the rule runs.
- **Time, Weekday, MonthDay, Month**: these fields are used to determine when the route is valid.
**Dial Plan**

**Examples**

**Example 1**
Pattern: 0X.
Prefix: 014
-> It will match any dialed number like 0140X.
For example if you dial 0140444434, the system will place a call to the selected route trunk.
If you dial 0533353331, the call will not be routed through this route, because the first three digits do not match 014

**Example 2**
Suffix: 025
Prefix: 0X.
-> It will ADD 025 to the dialed number 0X..
For example if you dial 0743444434, the system will place a call to the number 0250743444434
Audio

The Audio menu is dedicated to the upload and the managing of different types of audio files, collected in three different sections:

- the Sounds Section, for generic audio files;
- the Music on Hold Section, for audio files to play during the caller’s waiting time;
- the Monitor Section, for the recorded calls.

In each of these sections there is the list of the audio files already created, as shown in this screenshot.

There are some actions that can be done quickly through the Audio table:

- Add a new Sound or a new Music on Hold, filling this form:

- Delete the audio file
- Edit the audio file parameters
- Download a copy of the audio file
- Listen to the audio file preview
The New Recordings section’s features
- Define a Rating for each Call Recording;
- Quickly distinguish the inbound and outbound calls thanks to the Type label;
- Quickly access to the call recordings and Quality monitor. The search is much more effective and fast.
The Cally Square menu contains the tool to create and manage IVR applications for your Asterisk based telephony system.

In this section is shown the list of the Cally Square IVR applications already created.

There are some actions that can be done quickly through the Cally Square table:
- Create a New IVR Project
- Delete the IVR application project
- Edit the IVR application project

Each time you get a new IVR channel license you need to login inside the Linux server via ssh and launch the following command to restart the service: `service agisquare restart`

You can find the full documentation of Cally Square at this link:
[http://www.callysquare.com/documentation/]
xcally shuttle includes also the analytics menu dedicated to reporting, divided into these sections:

**metrics**

from this section the administrator can create custom metrics by clicking on the button “new metric” and filling the following form, providing the metric name and description:

```
create metric

name *

description
```

this text area has a limit of 225 chars.

[Save changes]
After the Metric creation, you have to specify the relative Query, by clicking on the Edit button. Here you can find the Edit Metric Section with:

1. **General Settings**, where you can edit the following Metric parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The Metric Name, assigned in the creation form</td>
</tr>
<tr>
<td>Description</td>
<td>The Metric Description, assigned in the creation form</td>
</tr>
<tr>
<td>Tables for join</td>
<td>The tables of your database that you want to join for the query</td>
</tr>
</tbody>
</table>

2. **Query**, where you can define and type your SQL Query:

```
COUNT(CASE WHEN call_log.connect = 1 AND (call_log.outtime IS NOT NULL AND call_log.outtime IS NOT NULL) THEN 1 END)
```

3. **Help**, where you can find a list of the Asterisk’s tables columns in the database, really useful

The created Metrics can be used in the Report section, also combined with other queries, to obtain interesting and complex reportings.
Analytics

Reports
This smart new section is dedicated to the creation of realtime reportings, which can contain also the Metrics.

From the General Section you can create, rename, delete, cut, copy and paste Reports. To create a new Report click on the folder Analytics (or one sub-folder) with the right mouse button and then on “Create”; assign a name to the report and then start to organize the Query using the Query Designer block, on the right of the screen.

From the Query Designer block you can define:
- **result columns** of the report, selecting them also from the created Metrics
- **where condition**
- **group by condition**
- **sort condition**.
Analytics

While you are defining the parameters in the Query Designer block, you can see below the automatic creation of the Query and the Preview Result.

To view the Complete Report you have to define the time interval of interest, by clicking on this button on the left of the screen and then on Apply.
**Analytics**

You can still check if the data you defined are correct in this pop-up, which appears before the Complete Report Creation:

Please enter field date

`call_log.rtdtm BETWEEN '2014-07-08' AND '2014-08-06'`

After the confirmation of the pop-up, you can see the Complete Report and copy the table to the clipboard or export it in CSV, Excel and PDF.

Like in the Metric Menu, there is also the Help Section which helps the Administrator to manage the queries and to create complex and useful Reports.
**Analytics**

**Ready to use Reports**
xCally Shuttle provides also a list of Reports ready to use, dedicated to Queues Stats, Calls Stats and Agent Stats. In this way you can find the main analytics already available for your needs. Let’s analyze them:

- **Queues Stats**, filtered by Answered Calls, Unanswered Calls or by Time Distribution.

- **Calls Stats**, containing detailed data about every single call.

- **Agents Stats**, which provides data about the Agent’s login, pause status and other events.

- **Cally Square Stats**, dedicated to IVR analytics.

To view one report, simply click on its name.

For each report you can add:
- new result columns,
- *where* conditions,
- *group by* conditions,
- *sort* conditions,
according to your reporting interests.

The Report data are extracted from the main xCally DB tables: `agent_log`, `call_log`, `cdr` and `queue_log`. You can find the DB documentation in the xCally DB tables Annex.
Analytics - Queues Stats Reports

Answered / Summary

SELECT COUNT(call_log.calltime) AS "Answered Calls",
SEC_TO_TIME(SUM(call_log.calltime)) AS "Total Call Time",
SEC_TO_TIME(AVG(call_log.calltime)) AS "AVG Call",
SEC_TO_TIME(SUM(call_log.holdtime)) AS "Total Hold Time"
FROM call_log WHERE call_log.connect='1'

Answered / Calls by agent

SELECT call_log.agent AS "Agent name",
COUNT(call_log.calltime) AS "Calls",
COUNT(IF(call_log.connect = 1,1,NULL)) AS "% Call",
SEC_TO_TIME(SUM(call_log.calltime)) AS "Call Time",
TRUNCATE(100*SUM(IF(call_log.connect = 1,call_log.calltime,NULL)) /
(SELECT SUM(IF(call_log.connect = 1,call_log.calltime,NULL))
FROM call_log),2) AS "Total Call Time %",
SEC_TO_TIME(AVG(call_log.calltime)) AS "AVG Call Time",
call_log.holdtime AS "Hold time",
SEC_TO_TIME(AVG(call_log.holdtime)) AS "AVG Hold Time"
FROM call_log WHERE call_log.connect='1'
GROUP BY call_log.agent
### Analytics - Queues Stats Reports

#### Answered / Service Level

```sql
SELECT COUNT(IF(call_log.calltime > 0 AND call_log.calltime <= 15
    AND call_log.connect = 1,1,NULL)) AS "Call Time (within 15 sec)",
COUNT(IF(call_log.calltime > 0 AND call_log.calltime <= 30
    AND call_log.connect = 1,1,NULL)) AS "Call Time (within 30 sec)",
COUNT(IF(call_log.calltime > 0 AND call_log.calltime <= 45
    AND call_log.connect = 1,1,NULL)) AS "Call Time (within 45 sec)",
COUNT(IF(call_log.calltime > 0 AND call_log.connect = 1,1,NULL))
    AS "Call Time (within 60+ sec)"
FROM call_log
```

#### Answered / Calls by queue

```sql
SELECT call_log.queuename AS "Queue name",
COUNT(call_log.calltime) AS "Calls",
COUNT(IF(call_log.connect = 1,1,NULL)) AS "% Call",
SEC_TO_TIME(SUM(call_log.calltime)) AS "Call Time",
TRUNCATE(100*SUM(IF(call_log.connect = 1,call_log.calltime,NULL))/
    (SELECT SUM(IF(call_log.connect = 1,call_log.calltime,NULL))
    FROM call_log),2) AS "Total Call Time %",
SEC_TO_TIME(AVG(call_log.calltime)) AS "AVG Call Time",
SEC_TO_TIME(AVG(call_log.holdtime)) AS "AVG Hold time"
FROM call_log WHERE call_log.connect='1'
GROUP BY call_log.queuename
```
**Analytics - Queues Stats Reports**

**Answered / Transferred**

SELECT call_log.origcalldate AS "Orig call date",
call_log.origcalltime AS "Orig call time",
call_log.uniqueid AS "Uniqueid",
call_log.queue AS "Queue name",
call_log.agent AS "Agent name",
call_log.callerid AS "Callerid",
call_log.holdtime AS "Hold time",
call_log.ringtime AS "Ring time",
call_log.extension AS "Transfer Extension",
call_log.context AS "Transfer Context"
FROM call_log WHERE call_log.transfer='1' and call_log.connect='1'

**Answered / Disconnection Cause**

SELECT call_log.event AS "Last event",
COUNT(call_log.event) AS "Count"
FROM call_log WHERE call_log.connect='1'
GROUP BY call_log.event
Analytics - Queues Stats Reports

Distribution / Calls by date

```
SELECT call_log.origcalldate AS "Date",
    COUNT(IF(call_log.connect = 1,1,NULL)) AS "Answer",
    TRUNCATE(100*COUNT(IF(call_log.connect = 1,1,NULL))/
        (SELECT COUNT(IF(call_log.connect = 1,1,NULL)) FROM call_log),2)
    AS "Answer %",
    COUNT(IF(call_log.connect = 0,1,NULL)) AS "Unanswer",
    TRUNCATE(100*COUNT(IF(call_log.connect = 0,1,NULL))/
        (SELECT COUNT(IF(call_log.connect = 0,1,NULL)) FROM call_log),2)
    AS "Unanswer %",
    SEC_TO_TIME(AVG(call_log.calltime)) AS "AVG Call time",
    SEC_TO_TIME(AVG(call_log.holdtime)) AS "AVG Hold time"
FROM call_log WHERE call_log.connect='1' or call_log.connect='0'
GROUP BY call_log.origcalldate
```

**FOR EACH DATE**

- **The number of answered calls**
- **The percentage of answered calls compared to the total number of answered calls**
- **The number of unanswered calls**
- **The percentage of unanswered calls compared to the total number of unanswered calls**
- **The average call time**
- **The average hold time**

Distribution / Calls by month

```
SELECT call_log.origcallmonth AS "Month",
    COUNT(IF(call_log.connect = 1,1,NULL)) AS "Answer",
    TRUNCATE(100*COUNT(IF(call_log.connect = 1,1,NULL))/
        (SELECT COUNT(IF(call_log.connect = 1,1,NULL)) FROM call_log),2)
    AS "Answer %",
    COUNT(IF(call_log.connect = 0,1,NULL)) AS "Unanswer",
    TRUNCATE(100*COUNT(IF(call_log.connect = 0,1,NULL))/
        (SELECT COUNT(IF(call_log.connect = 0,1,NULL)) FROM call_log),2)
    AS "Unanswer %",
    SEC_TO_TIME(AVG(call_log.calltime)) AS "AVG Call time",
    SEC_TO_TIME(AVG(call_log.holdtime)) AS "AVG Hold time"
FROM call_log WHERE call_log.connect='1' or call_log.connect='0'
GROUP BY call_log.origcallmonth
```

**FOR EACH MONTH**

- **The number of answered calls**
- **The percentage of answered calls compared to the total number of answered calls**
- **The number of unanswered calls**
- **The percentage of unanswered calls compared to the total number of unanswered calls**
- **The average call time**
- **The average hold time**
Analytics - Queues Stats Reports

Distribution / Calls by hour

SELECT call_log.origcallhour AS "Hour",
COUNT(IF(call_log.connect = 1,1,NULL)) AS "Answer",
TRUNCATE(100*COUNT(IF(call_log.connect = 1,1,NULL))/
(SELECT COUNT(IF(call_log.connect = 1,1,NULL)) FROM call_log),2)
AS "Answer %",
COUNT(IF(call_log.connect = 0,1,NULL)) AS "Unanswer",
TRUNCATE(100*COUNT(IF(call_log.connect = 0,1,NULL))/
(SELECT COUNT(IF(call_log.connect = 0,1,NULL)) FROM call_log),2)
AS "Unanswer %",
SEC_TO_TIME(AVG(call_log.calltime)) AS "AVG Call time",
SEC_TO_TIME(AVG(call_log.holdtime)) AS "AVG Hold time"
FROM call_log WHERE call_log.connect='1' or call_log.connect='0'
GROUP BY call_log.origcallhour

Distribution / Calls by day

SELECT call_log.origcallday AS "Day",
COUNT(IF(call_log.connect = 1,1,NULL)) AS "Answer",
TRUNCATE(100*COUNT(IF(call_log.connect = 1,1,NULL))/
(SELECT COUNT(IF(call_log.connect = 1,1,NULL)) FROM call_log),2)
AS "Answer %",
COUNT(IF(call_log.connect = 0,1,NULL)) AS "Unanswer",
TRUNCATE(100*COUNT(IF(call_log.connect = 0,1,NULL))/
(SELECT COUNT(IF(call_log.connect = 0,1,NULL)) FROM call_log),2)
AS "Unanswer %",
SEC_TO_TIME(AVG(call_log.calltime)) AS "AVG Call time",
SEC_TO_TIME(AVG(call_log.holdtime)) AS "AVG Hold time"
FROM call_log WHERE call_log.connect='1' or call_log.connect='0'
GROUP BY call_log.origcallday

FOR EACH HOUR

THE NUMBER OF ANSWERED CALLS
THE PERCENTAGE OF ANSWERED CALLS COMPARED TO THE TOTAL NUMBER OF ANSWERED CALLS
THE NUMBER OF UNANSWERED CALLS
THE PERCENTAGE OF UNANSWERED CALLS COMPARED TO THE TOTAL NUMBER OF UNANSWERED CALLS
THE AVERAGE CALL TIME
THE AVERAGE HOLD TIME

FOR EACH DAY OF THE MONTH

THE NUMBER OF ANSWERED CALLS
THE PERCENTAGE OF ANSWERED CALLS
THE NUMBER OF UNANSWERED CALLS
THE PERCENTAGE OF UNANSWERED CALLS
THE AVERAGE CALL TIME
THE AVERAGE HOLD TIME
Analytics - Queues Stats Reports

Unanswered / Summary

SELECT COUNT(*) AS "Number of Unanswered Calls",
SEC_TO_TIME(AVG(call_log.waittime)) AS "Avg wait time",
TRUNCATE(AVG(call_log.position),0) AS "Avg queue position",
TRUNCATE(AVG(call_log.origposition),0) AS "Orig position"
FROM call_log WHERE call_log.connect='0'

Unanswered / Disconnection cause

SELECT call_log.event AS "Last event",
COUNT(call_log.event) AS "Count"
FROM call_log WHERE call_log.connect='0'
GROUP BY call_log.event

For which event the call has been terminated:
- Abandon (The caller abandoned their position in the queue)
- Exitempty (The caller was exited from the queue forcefully because the queue had no reachable members and it’s configured to do that to callers when there are no reachable members)
- Exitwithkey (The caller elected to use a menu key to exit the queue)
- Exitwithtimeout (The caller was on hold too long and the timeout expired)
Analytics - Queues Stats Reports

Unanswered / Calls by Queue

```
SELECT call_log.queuename AS "Queue name",
COUNT(IF(call_log.connect = 0,1,NULL)) AS "Count",
TRUNCATE(100*COUNT(IF(call_log.connect = 0,1,NULL)) /
(SELECT COUNT(IF(call_log.connect = 0,1,NULL)) FROM call_log),2)
AS "Count %"
FROM call_log WHERE call_log.connect='0'
GROUP BY call_log.queuename
```

For each queue:

- **The number of unanswered calls**
- **The percentage of the unanswered calls compared to all the unanswered queues calls**
Analytics - Calls Stats Reports

CDR

SELECT cdr.calldate AS "Call Date",
cdr.clid AS "Clid",
cdr.src AS "Source",
cdr.dst AS "Destination",
cdr.dcontext AS "Context",
cdr.channel AS "Channel",
cdr.dstchannel AS "Dst. Channel",
cdr.lastapp AS "Last App.",
cdr.lastdata AS "Last Data",
cdr.duration AS "Duration",
cdr.billsec AS "Billsec",
cdr.disposition AS "Disposition",
cdr.uniqueid AS "Uniqueid",
cdr.linkedid AS "Linkedid"
FROM cdr
Outbound

```
SELECT call_log.uniqueid AS "Uniqueid",
call_log.agent AS "Agent name",
call_log.callerid AS "Callerid",
SEC_TO_TIME(call_log.duration) AS "Total duration",
SEC_TO_TIME(call_log.billsec) AS "Total billsec",
call_log.tag AS "Tag",
call_log.outboundcalldtm AS "Outbound Call Date",
call_log.outboundcalltime AS "Outbound Call Time"
FROM call_log WHERE call_log.outboundcall='1'
```

Summary

```
SELECT cdr.disposition AS "Disposition",
COUNT(cdr.disposition) AS "Count"
FROM cdr
GROUP BY cdr.disposition
```

**DISPOSITION**

- **ANSWERED**: A successful dial. The caller reached the callee.
- **BUSY**: The dial command reached its number but the number is busy.
- **CONGESTION**: The dialled number is not recognised.
- **FAILED**
- **NO ANSWER**: The dial command reached its number, the number rang for too long, then the dial timed out.
Analytics - Calls Stats Reports

Answered

```sql
SELECT cdr.src AS "Source",
cdr.dst AS "Destination",
cdr.duration AS "Duration",
cdr.billsec AS "Billsec",
cdr.uniqueid AS "Uniqueid",
cdr.start AS "Start",
cdr.answer AS "Answer",
cdr.end AS "End"
FROM cdr WHERE   cdr.disposition='ANSWERED'
```

**FOR EACH CALL**

- **The Caller Source Number**
- **The Destination Number**
- **The Duration**
- **Billsec**: The Total time call is up, in seconds, from answer to hangup
- **The Unique ID**
- **The Start Time of the Call**
- **The Answer Time of the Call**
- **The End Time of the Call**

Busy

```sql
SELECT cdr.src AS "Source",
cdr.dst AS "Destination",
cdr.duration AS "Duration",
cdr.billsec AS "Billsec",
cdr.uniqueid AS "Uniqueid",
cdr.start AS "Start",
cdr.end AS "End"
FROM cdr WHERE   cdr.disposition='BUSY'
```

**FOR EACH CALL**

- **The Caller Source Number**
- **The Destination Number**
- **The Duration**
- **Billsec**: The Total time call is up, in seconds, from answer to hangup
- **The Unique ID**
- **The Start Time of the Call**
- **The End Time of the Call**
Analytics - Calls Stats Reports

Congestion

SELECT cdr.src AS "Source",
cdr.dst AS "Destination",
cdr.duration AS "Duration",
cdr.billsec AS "Billsec",
cdr.uniqueid AS "Uniqueid",
cdr.start AS "Start",
cdr.end AS "End"
FROM cdr WHERE cdr.disposition='CONGESTION'

For each call

The Caller Source Number  The Destination Number  The Duration  Billsec  The Unique ID  The Start Time of the Call  The End Time of the Call

No Answer

SELECT cdr.src AS "Source",
cdr.dst AS "Destination",
cdr.duration AS "Duration",
cdr.billsec AS "Billsec",
cdr.uniqueid AS "Uniqueid",
cdr.start AS "Start",
cdr.end AS "End"
FROM cdr WHERE cdr.disposition='NOANSWER'

For each call

The Caller Source Number  The Destination Number  The Duration  Billsec  The Unique ID  The Start Time of the Call  The End Time of the Call

Failed

SELECT cdr.src AS "Source",
cdr.dst AS "Destination",
cdr.duration AS "Duration",
cdr.billsec AS "Billsec",
cdr.uniqueid AS "Uniqueid",
cdr.start AS "Start",
cdr.end AS "End"
FROM cdr WHERE cdr.disposition='FAILED'

For each call

The Caller Source Number  The Destination Number  The Duration  Billsec  The Unique ID  The Start Time of the Call  The End Time of the Call
Analytics - Agents Stats Reports

Detail

```
SELECT agent_log.agent AS "Agent name",
agent_log.queuename AS "Queue name",
agent_log.event AS "Agent Event",
agent_log.type AS "Type Agent Event",
agent_log.enterdtm AS "Enter Agent Date",
agent_log.entermtime AS "Enter Agent Time",
agent_log.exittim AS "Exit Agent Date",
agent_log.exittime AS "Exit Agent Time",
agent_log.duration AS "Duration Event Agent"
FROM agent_log

FOR EACH AGENT

HIS/HER NAME  HIS/HER QUEUE
HIS/HER AGENT EVENTS, LIKE MEMBER, QUEUE, PAUSE, RINGNOANSWER
HIS/HER TYPE EVENT: ADDMEMBER (MEMBER), AFTERCALL, BACKOFFICE, PAUSE (PAUSE), LOGINTIME, NOTINUSE (QUEUE)

THE EVENT START DATE  THE EVENT START TIME
THE EVENT END DATE  THE EVENT END TIME
THE EVENT DURATION

Pause - Pause by Queue

SELECT agent_log.queuename AS "Queue name",
SEC_TO_TIME(SUM(agent_log.duration)) AS "Total Pause"
FROM agent_log WHERE agent_log.event='PAUSE'
GROUP BY agent_log.queuename

FOR EACH QUEUE

THE QUEUE NAME  THE QUEUE TOTAL PAUSE DURATION
```
Analytics - Agents Stats Reports

Pause - Pause Total

SELECT agent_log.agent AS “Agent name”,
agent_log.queuename AS “Queue name”,
agent_log.type AS “Type Pause”,
agent_log.enterdtm AS “Enter Agent Date”,
agent_log.enteretime AS “Enter Agent Time”,
agent_log.exitdtm AS “Exit Agent Date”,
agent_log.exittime AS “Exit Agent Time”,
SEC_TO_TIME(agent_log.duration) AS “Duration”
FROM agent_log WHERE agent_log.event=’PAUSE’

FOR EACH AGENT

HIS/HER NAME
HIS/HER QUEUE
HIS/HER PAUSE TYPE:
  AFTERCALL
  BACKOFFICE
  PAUSE
HIS/HER PAUSE START DATE
HIS/HER PAUSE START TIME
HIS/HER PAUSE END DATE
HIS/HER PAUSE END TIME
HIS/HER PAUSE DURATION

Pause - Pause by Agent

SELECT agent_log.agent AS “Agent name”,
SEC_TO_TIME(SUM(agent_log.duration)) AS “Total Pause”
FROM agent_log WHERE agent_log.event=’PAUSE’
GROUP BY agent_log.agent

FOR EACH AGENT

HIS/HER NAME
HIS/HER TOTAL PAUSE DURATION
Analytics - Agents Stats Reports

Login

SELECT agent_log.agent AS "Agent name",
    agent_log.queue AS "Queue name",
    agent_log.enterdtm AS "Enter Agent Date",
    agent_log.enter AS "Enter Agent Time",
    agent_log.exitdtm AS "Exit Agent Date",
    agent_log.exit AS "Exit Agent Time",
    SEC_TO_TIME(agent_log.duration) AS "Duration"
FROM agent_log
WHERE agent_log.event='QUEUE'
    AND agent_log.type='LOGINTIME'

Notinuse

SELECT agent_log.agent AS "Agent name",
    agent_log.queue AS "Queue name",
    agent_log.enterdtm AS "Enter Agent Date",
    agent_log.enter AS "Enter Agent Time",
    agent_log.exitdtm AS "Exit Agent Date",
    agent_log.exit AS "Exit Agent Time",
    SEC_TO_TIME(agent_log.duration) AS "Duration"
FROM agent_log
WHERE agent_log.event='QUEUE'
    AND agent_log.type='NOTINUSE'
**Analytics - Cally Square Stats Reports**

**Detail**

```sql
SELECT call_log.uniqueid AS "Uniqueid",
call_log.callerid AS "Callerid",
call_log.ivrchoices AS "IVR choices",
call_log.ivrblocks AS "IVR blocks",
call_log.ivrtime AS "IVR duration",
call_log.ivrname AS "IVR Name",
call_log.ivrgoals AS "IVR GOAL"
FROM call_log WHERE call_log.ivrstart='1'
```

**FOR EACH IVR CALL**

- **Unique Id**
- **Caller Id**
- **IVR Choices**
- **IVR Blocks**
- **IVR Duration**
- **IVR Name**
- **IVR Goal**

When the IVR is in the GOAL block.

**IVR and Queue**

```sql
SELECT call_log.uniqueid AS "Uniqueid",
call_log.ivrstartdtm AS "IVR start date",
call_log.ivrstarttime AS "IVR start time",
call_log.enterqueuetime AS "Enter queue time",
call_log.queuename AS "Queue name",
SEC_TO_TIME(call_log.calltime) AS "Answer (Call)",
SEC_TO_TIME(call_log.waittime) AS "Unanswer (Wait)",
SEC_TO_TIME(call_log.duration) AS "Total duration"
FROM call_log WHERE call_log.ivrstart='1' and call_log.enterqueue='1'
```

- **Unique ID**
- **Start Date**
- **Start Time**
- **Enter Queue Time**
- **Queue Name**
- **Answer - Call Time**
- **Unanswer - Wait Time**
- **Total Duration**
This really useful section permits to the Contact Center Administrator to monitor Queues, Agents, Trunks, Integrations and Dial Plans in realtime.

Details

This table shows realtime analytics about all the Queues, like the number of waiting and managed calls, the hold time, the number of completed and abandoned calls, the routing strategy, the weight and so on. It's possible to reset this realtime data by clicking on the Reset button.

Agents - General

<table>
<thead>
<tr>
<th>Name</th>
<th>Interface</th>
<th>Ip Status</th>
<th>Search</th>
</tr>
</thead>
<tbody>
<tr>
<td>alexandra.becone</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>barbara.boglio</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>brina.pzo</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>diana.com</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>giuseppi.fiori</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>marcel</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>sarah</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>oscar</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>sara3</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>veiratel1</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>veiratel2</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>veiratel3</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
<tr>
<td>veiratel4</td>
<td>192.168.1.1.1</td>
<td>AVAILABLE</td>
<td></td>
</tr>
</tbody>
</table>

7th
**Realtime**

Here the supervisor can view which agents are logged, their Interface and their actual SIP status. Thanks to the two button present on the web interface he can:

- **add Agents** on a specific Queue in **realtime**. This is a temporary Agent-Queue association: when the agent logs out, the association will be deleted.

- **pause and unpause Agents**, selecting the pause type. This action involves all the Agent’s Queues.

---

### Agents - Detail

<table>
<thead>
<tr>
<th>Queue</th>
<th>Name</th>
<th>Interface</th>
<th>Penalty</th>
<th>Calls Taken</th>
<th>Last call</th>
<th>Status</th>
<th>Paused</th>
</tr>
</thead>
<tbody>
<tr>
<td>Billing</td>
<td>alessandra.boccione</td>
<td>SR_F</td>
<td>0</td>
<td>0</td>
<td></td>
<td>UNAVAILABLE</td>
<td></td>
</tr>
<tr>
<td>Billing</td>
<td>marcello</td>
<td>SR_F</td>
<td>0</td>
<td>0</td>
<td></td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>CRM</td>
<td>alessandra.boccione</td>
<td>SR_F</td>
<td>0</td>
<td>0</td>
<td></td>
<td>UNAVAILABLE</td>
<td></td>
</tr>
<tr>
<td>Sugar</td>
<td>alessandra.boccione</td>
<td>FR_J</td>
<td>0</td>
<td>0</td>
<td></td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Sugar</td>
<td>verenel1</td>
<td>FR_F</td>
<td>0</td>
<td>0</td>
<td></td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td>giovanni1</td>
<td>FR_J</td>
<td>0</td>
<td>0</td>
<td></td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td>giovanni2</td>
<td>FR_J</td>
<td>0</td>
<td>0</td>
<td></td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td>giovanni3</td>
<td>FR_J</td>
<td>0</td>
<td>0</td>
<td></td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td>marcello</td>
<td>FR_F</td>
<td>0</td>
<td>2</td>
<td>Dec 16, 2014 11:58:47 AM</td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td>cassandro1</td>
<td>FR_Fiammetta</td>
<td>0</td>
<td>0</td>
<td></td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td>salome1</td>
<td>FR_fhanel1</td>
<td>0</td>
<td>2</td>
<td>Dec 16, 2014 22:19:40 PM</td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td>casole</td>
<td>FR_fhanel2</td>
<td>0</td>
<td>0</td>
<td></td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td>verenel2</td>
<td>FR_fhanel1</td>
<td>0</td>
<td>1</td>
<td>Dec 16, 2014 19:30:27 AM</td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td>marcello</td>
<td>FR_fhanel2</td>
<td>0</td>
<td>0</td>
<td></td>
<td>NOT IN USE</td>
<td></td>
</tr>
<tr>
<td>Support</td>
<td>verenel3</td>
<td>FR_fhanel3</td>
<td>0</td>
<td>4</td>
<td>Dec 17, 2014 04:43:25 PM</td>
<td>NOT IN USE</td>
<td></td>
</tr>
</tbody>
</table>

---

In this table there are more detailed information about the Agents-Queue association, as the number of calls managed by an Agent in a specific Queue and when he had the last call in a specific Queue.

From this view the supervisor can:

- set the pause status for each Agent on a single specific Queue, thanks to the Pause or Resume button.
- remove one single Agent from a Queue in realtime, thanks to the Remove button.
Realtime

Queues - Waiting Calls + Current Calls
These two sections contain, in realtime, the list of all the Waiting and Current queue calls. When a call is answered, the corresponding row will disappear from the Waiting Calls section and appear in the Current Calls section. For each call there are data about the Queue, the Caller, the Current position in the Queue.

Queues - Agent Status
Here the supervisor can visualize two tables containing:
- the Called Agents, with info about the Caller, the Queue and the Extension
- the Connected Agents, with info about the Agent which is managing the call, the Queue, the Caller, the Hold time and the Talk time of the call.
Voice Mail

This section is relative to the Voice Mail configuration. The Voice Mail lets the Customer to leave a message in case his call can’t be managed by the Agents.

**Voicemail creation**
To create a Voicemail click on the New Voice Mail button and fill in the form with the following information:
Voice Mail

- Mailbox: a Voice Mail reference
- Password: the relative password
- Full Name: the name associated to the Mailbox
- Email: the Mailbox address that will receive the notification of the message

We suggest you to put in the Mailbox field the Agent’s Username and in the Password field the Agent’s Password.
To enable the Voice Mail section to Agents, please see at page 9 the Modules Permissions field.

Edit a Voice Mail

If you would like to edit one of your created Voice Mail, click on the Edit button.
You will find 2 tabs relative to the Voice Mail Settings (General and Advanced) and one tab where you can find the Recorded Messages.

General Settings

The General Settings are the following:
- the Context: from-voicemail
- the Mailbox name: the Agent’s Username
- the Password: the Agent’s Password
- the Full name
- the Email: the Agent’s Email.
Voice Mail

Advanced Settings

The Advanced Settings are the following:

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timezone</td>
<td>central</td>
<td>Timezone</td>
</tr>
<tr>
<td>Attach</td>
<td>yes</td>
<td>Attach</td>
</tr>
<tr>
<td>Envelope</td>
<td>no</td>
<td>Envelope</td>
</tr>
<tr>
<td>Delete</td>
<td>no</td>
<td>Delete</td>
</tr>
<tr>
<td>Email body</td>
<td>Empty</td>
<td>Email body</td>
</tr>
<tr>
<td>Email subject</td>
<td>Empty</td>
<td>Email subject</td>
</tr>
<tr>
<td>Max seconds</td>
<td>180</td>
<td>Max seconds</td>
</tr>
<tr>
<td>Max messages</td>
<td>100</td>
<td>Max messages</td>
</tr>
</tbody>
</table>

Timezone To define custom timezones
Attach The audio file of the voicemail message is sent to the user as an attachment in an the e-mail notification message. Yes|No
Envelope It controls whether or not Asterisk will play the message envelope (date/time) before playing the voicemail message. Yes|No
Delete Yes: the message will be deleted from the voicemailbox (after having been emailed). Yes|No
Email Body It overrides the normal message text seen in the body of a voicemail notification message.
Email Subject It sets the custom Subject: line of the voicemail notification message. The value passed is a string containing the text to put in the Subject line.
Max Seconds It defines the maximum amount of time in seconds of an incoming message. If is set to 0 there will be no maximum time limit enforced.
Max Messages It sets the maximum number of messages allowed in a voicemail folder. When a mailbox has more than this number of messages in it, new messages can’t be recorded and vm-mailboxfull is played to the caller. Default: 100, Max: 9999

Messages

In this section you can find the list of the Messages, with info relative to the Caller-ID, the Date and the Duration.
**Settings - General**

xCally Shuttle provides a Settings Menu containing these sections:

- **General**, about the xCally settings, license and the Asterisk parameters;
- **Integrations**, dedicated to the integration of xCally Shuttle with third party applications, like Zendes and SugarCRM;
- **Pauses**, to manage different kinds of Agent’s pause;
- **Tags**;
- **ChanSpy**, useful for the supervisor to define how he would listen to the calls between agents and customers;
- **BLF**, useful to manage codes associated to the external SIP Phones, for the login, logout and pause of the Agent;
- **Users**, useful to assign different roles and different accesses to the users of xCally Shuttle.

### General Settings

In the **General** section there are some parameters about the xCally Shuttle platform, like the Name of the **Organization** which uses it, the **Email** reference, the xCally Shuttle **Domain**, the preferred **Language**, **Date and Time Format** and also the preferred **Recording Format** (.gsm or .wav). **Max Preview** and **Records per page** are related to the Analytics - Reports section, about the data disposition in the preview report and in the complete report.

You can also set the **Auto Answer**, **Auto Answer Time**, **Secure Connection**, **Recordings Custom Path** and **Session Timeout**.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>XCALLY SHUTTLE</td>
<td>Name of the Organization</td>
</tr>
<tr>
<td>Email</td>
<td><a href="mailto:support@repeply.com">support@repeply.com</a></td>
<td>Email reference</td>
</tr>
<tr>
<td>Domain</td>
<td><a href="http://192.168.1.7:80">http://192.168.1.7:80</a></td>
<td>Domain</td>
</tr>
<tr>
<td>Language</td>
<td>English</td>
<td>Language</td>
</tr>
<tr>
<td>Date Format</td>
<td>03/07/2015</td>
<td>Date Format</td>
</tr>
<tr>
<td>Time Format</td>
<td>14:55:04</td>
<td>Time Format</td>
</tr>
<tr>
<td>Recording Format</td>
<td>record</td>
<td>Recording Format</td>
</tr>
<tr>
<td>Max Preview</td>
<td>20</td>
<td>Max Preview</td>
</tr>
<tr>
<td>Records per page</td>
<td>1900</td>
<td>Records per page</td>
</tr>
<tr>
<td>Auto Answer</td>
<td>No</td>
<td>Auto Answer</td>
</tr>
<tr>
<td>After Call Working</td>
<td>No</td>
<td>After Call Working</td>
</tr>
<tr>
<td>Specific the typology of SIP</td>
<td>SIP</td>
<td>Specific the typology of SIP</td>
</tr>
<tr>
<td>Auto Answer Timer</td>
<td>0</td>
<td>Auto Answer Timer</td>
</tr>
<tr>
<td>Secure Connection</td>
<td>No</td>
<td>Secure Connection</td>
</tr>
<tr>
<td>Recordings Custom Path</td>
<td>/var/www/html</td>
<td>Recordings Custom Path</td>
</tr>
<tr>
<td>Session Timeout</td>
<td>7600</td>
<td>Session Timeout (seconds)</td>
</tr>
</tbody>
</table>
Settings - General

All this data can be simply edited by clicking on the written value.

The data displayed in the License section can’t be edited, except for the License string.

If you edit IP or License Settings, then you have to restart services!

In the License tab you can find the License Order button, that will show you a form useful to request new licenses or update the existing ones adding new modules, for example Tiger Dial or the Contact Management. Our team will receive your request and provide you by email the new License.
**Settings - General**

**Socket Settings**
Here you can define the information about the Socket Server (the server to which the browsers are connected) and the Socket port. If you edit these parameters you have to execute the `service agisquare restart` and `service xcall-ly-realtime restart`.

![Socket Settings Table]

**Asterisk Manager**
Here you can set the Asterisk data (server, port, username and password of the user enabled on that Asterisk machine) and verify if the Asterisk connection is ok by clicking on the **Check Connection** button. If you edit these parameters you have to execute the `service agisquare restart` and `service xcall-ly-realtime restart`.

![Asterisk Manager Settings]

**Asterisk Network**

![Localnet]

![Externip]

![STUN Address Table]
**Settings - General**

Here you can manage some advanced Network configuration: it is possible to create a new Localnet (providing the **IP Network** and **Netmask**), new Externip (providing the **IP Address**) and define the STUN Address.

**Advanced**

This section can be useful to set, for example, the automatic numeration of the SIP that you will create in the SIP section. If you need extensions like ‘1000’ onwards, you can set 1000 as SIP reference value and that number will be incremented every time a new SIP is created.

**Cronjob**

Important To activate the cronjob insert a new line in cronjob list:

in the linux console run the command `crontab -e`

(use the internal xcally server ip address).
**Settings - General**

Here you can define the time interval of the **Automatic Database Backup** and the **Backup Type** (Config only or Config + Logs). In order to enable this feature, you have to set the **Cronjob** and the **Automatic Database Backup** properties as Active.

In the **History** table you can download and delete the backups. If there is not enough free memory on the disk you will see a row containing the Backup error, that means that the backup has not been created. The backups are stored in the `var/www/html/files/backup` directory.

**Outbound**

In the Outbound tab you can set the **IP Address** (or hostname) and the **port** of the outbound service. If you edit these parameters you have to execute the `service agisquare restart` and `service xcally-realtime restart`.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outbound Server</td>
<td>192.168.2.57</td>
<td>Outbound Server</td>
</tr>
<tr>
<td>Outbound Port</td>
<td>10080</td>
<td>Outbound Port</td>
</tr>
</tbody>
</table>
Settings - General

How to enable Secure Connection

To set xCALLY in the https secure connection go to the Secure Connection and set Yes. After this change you have to reset the xcally-realtime service, using the command service xcally-realtime restart. When you access to the xCALLY URL for the first time, you have also to accept the website certificate.

![Secure Connection Settings](image)

Open the linux console on your xcally server and edit the file:

/etc/asterisk/res_xcalld.conf

```plaintext
[general]
enabled = yes
bindaddr = X.X.X.X  // XCALLY IP ADDRESS
port = 8999
debug = no
webservice_secure = yes
response_format = json

[script]
authenticate = /root/bin/auth/add-iptables-rules.sh %s
unregistratr = /root/bin/peer_remove/peer_remove.sh %s

[mysql]
hostname = 127.0.0.1
schema = xcall
user = xcall
secret = xcall1234
port = 3306
query = SELECT view_peers.* FROM view_peers WHERE BINARY Peer = '4s' AND BINARY Secret = '4s'
```

Reload the Asterisk configuration, type: `asterisk -rx "reload"`
**Settings - Pauses**

This section gives the supervisor the possibility to create and manage different kinds of Agent Pause, like a General one, a Backoffice Pause and an Aftercall Pause, as shown in the screenshot below.

The Pause creation is really simple: you just have to click on the “New Pause” button and then fill the form with the Pause Type and Description.

Through the Pauses table you can:
- Edit a pause, clicking on
- Remove a pause, clicking on
Settings - Tags

In this section you can manage Tags, that you can use in the Dialplan and Integration sections.

First of all, you can create a Tag clicking on the New Tag button.
Simply type the Tag Name, its Description and click on Save Changes.

After that, you will see your new Tag into the Tag tab, where you can edit or delete it.

Tags - Dialplan
In the Dialplan -> Outbound section, when you edit a Route, you can specify one or more tags for it in the General Settings.

Tags - Zendesk Integration
In the Integration -> Trigger section, when you set a Trigger for the Zendesk Integration, you can specify one or more tags that will be added to the Zendesk ticket created by the integration.
**Settings - ChanSpy**

Here the supervisor can define and create ChanSpy, really useful for the Agent monitoring. Through a ChanSpy he can listen to calls or whisper into a conversation.

For example, you can find in this screenshot three kinds of ChanSpy:

- The supervisor can listen and talk to the agent. The customer will not listen to the supervisor voice.
- The supervisor can only listen the agent channel (neither the agent nor the customer will listen to the supervisor voice).
- The supervisor can listen and talk both to the agent and to the customer, like a pure 3 way conference room.

To create a ChanSpy click on the “New ChanSpy” button and then specify its Name, a Code, its ChanSpy Options and its Description.
The ChanSpy options are the following:

```
ChanSpy Options
- x: Only spy on channels involved in a bridged call.
- B: Instead of whispering on a single channel, barge in on both channels involved in the call.
- E: Exit when the spied-on channel hangs up.
- Q: Only listen to audio coming from this channel.
- D: Don't play a beep when beginning to spy on a channel, or speak the selected channel name.
- S: Skip the playback of the channel type when speaking the selected channel name.
- W: Stop when no more channels are left to spy on.
- X: Enable whisper mode, so the spying channel can talk to the spied-on channel.
- V: Enable private whisper mode, so the spying channel can talk to the spied-on channel but cannot listen to that channel.
```

Through the ChanSpy table you can:
- Edit a ChanSpy, clicking on Edit
- Remove a ChanSpy, clicking on Delete

In order to use the ChanSpy you have to write into the xCALLY phone bar the following structure: [chanSpy code][the SIP of the Agent] as shown in the image below and click on the green button.
**Settings - BLF**

The new xCALLY release contains a section dedicated to the BLF configuration, that permits the Agents to login, logout and manage their pause status directly from their SIP Phone. In this way Agents can manage the Customer Care calls from their phone without the need to be logged on the Shuttle Web interface.

For example, if the Agent types *90, on his SIP Phone, he will be logged on xCALLY and he can receive his Queue Calls, even if it is not connected to the xCALLY Shuttle Web Interface.

To create a BLF click on the “New BLF” button and then specify its Code, the Value (as Login, Logout Pause, Unpause...) and the State, which is associated to the multi colour LEDs of the phone. You can also enable the audio status message, that is an audio feedback for the Agent.
**Settings - Users**

This section, which is contained in the Settings Menu like shown in the image, shows the Users list. It’s possible to assign different roles and different accesses to the xCally Shuttle Modules, for each User.

The Users table shows information about their Role, Status (if users are active or not), their Last Login date and if they are Administrators or not.

Through the web interface the Administrator can:

- Delete a User  
  ![Delete](Delete)

- Edit the User parameters  
  ![Edit](Edit)

- Define User’s permissions, as described at page 73  
  ![Permissions](Permissions)

- Create a new User, by clicking on  
  ![New User](New User)  and compiling a form, where it’s possible to indicate the User Role, his Status, his Module Permissions and if he is an Admin or not.
**Settings - Users**

As said in the previous page, to create a new User you have to fill the form showned in the image below.

Let’s focus on the main fields:
- Username and Password will be used by the User to login into the system
- If you want to create an Admin User you have to select “yes” in the Admin field
- In the Module Permissions you can select which of the xCally Shuttle sections can be viewed and managed by the User.
**Settings - Users**

**Permissions**
It's possible to define different Users Permissions: each supervisor user will be able to see just the reports and real time infos related to the assigned Queues and Agents.

The Permissions Section contains the list of all the Agents and all the Queues. From these lists you can select which Agents and Queues can be managed by the supervisor, simply by clicking on their name. If the association User-Agent or User-Queue has been made correctly, you will see the positive feedback on the Results section, on the right of the screen.

To delete a User-Queue or User-Agent association simply click on the name of the Queue or of the Agent in the Access List.
**Settings - Integrations**

xCALLY Shuttle provides a seamless CTI integration with third party applications, like Zendesk and SugarCRM.

To manage this kind of integrations there is a dedicated sub-section in the Settings Menu, divided into three sections:

- Integrations
- Triggers
- Custom Fields

**Integrations**

Through the Integrations table the Administrator can:

- Delete an Integration
- Edit the Integration parameters
- Create a new Integration, by clicking on [New Integration] and compiling the following form with data about a valid Admin Account and a valid URI of Zendesk or SugarCRM, according to your integration type.
**Settings - Integrations**

**Triggers**
In the Triggers interface you can set how and when the integration works, thanks to the creation of different Triggers, which can be associated to different Queues.

Through the Triggers table the Administrator can:
- Delete a trigger
- Edit the Trigger parameters
- Create a new Trigger, by clicking on and compiling the following form with data about which integration you want to use, on which Queue and which creation strategy (Ringing: when the phone is ringing, before the agent’s answer - Up: when the agent picks up the call - Hang up: when the agent hangs up the call - Unmanaged call: when the caller has hang up before talking with an agent).

![Create Trigger Form](Create Trigger Form)
**Settings - Integrations**

You can find more detailed instructions about the Integrations Module on the two following guides:

- Zendesk Integration Guide
- SugarCRM Integration Guide
- Salesforce Integration Guide
Annex - Architecture
Annex - xCally DB Tables

AGENT_LOG

This table contains all the events regarding Agents, like when they are added to queues, when they login or pause their activity.

<table>
<thead>
<tr>
<th>field</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Event ID.</td>
</tr>
<tr>
<td>rtdtm</td>
<td>Last update date. Date of the last event.</td>
</tr>
<tr>
<td>rtime</td>
<td>Last update time. Time of the Last Event.</td>
</tr>
<tr>
<td>agent</td>
<td>Agent Name.</td>
</tr>
<tr>
<td>queuename</td>
<td>Queue Name.</td>
</tr>
<tr>
<td>event</td>
<td>Agent Event.</td>
</tr>
<tr>
<td>type</td>
<td>Agent Event Type.</td>
</tr>
<tr>
<td>enterdtm</td>
<td>Enter Agent Date.</td>
</tr>
<tr>
<td>entertime</td>
<td>Enter Agent Time.</td>
</tr>
<tr>
<td>exitdtm</td>
<td>Exit Agent Date.</td>
</tr>
<tr>
<td>exittime</td>
<td>Exit Agent Time.</td>
</tr>
<tr>
<td>duration</td>
<td>Event Duration.</td>
</tr>
</tbody>
</table>

The following table contains the list of the possible event and type values:

<table>
<thead>
<tr>
<th>event</th>
<th>type</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MEMBER</td>
<td>ADDMEMBER</td>
<td>When a member is added to a queue.</td>
</tr>
<tr>
<td>QUEUE</td>
<td>LOGINTIME</td>
<td>When the agent has been logged.</td>
</tr>
<tr>
<td></td>
<td>NOTINUSE</td>
<td>When the agent has been logged and he didn't managed calls.</td>
</tr>
<tr>
<td>PAUSE</td>
<td>PAUSE</td>
<td>When the agent starts his pause.</td>
</tr>
<tr>
<td></td>
<td>BACKOFFICE</td>
<td>When the agent starts his Backoffice pause.</td>
</tr>
<tr>
<td></td>
<td>AFTERCALL</td>
<td>When the agent starts his Aftercall pause.</td>
</tr>
<tr>
<td>RINGNOANSWER</td>
<td>call Unique_id</td>
<td>After trying for ringtime ms to connect to the available queue member, the attempt ended without the member picking up the call.</td>
</tr>
</tbody>
</table>
Annex - xCally DB Tables

CALL_LOG

This table contains all the data about Calls: their duration, their last event type, their queue, the agent who answered and so on...

<table>
<thead>
<tr>
<th>field</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Event ID.</td>
</tr>
<tr>
<td>unique_id</td>
<td>The unique ID for the src channel. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>queuename</td>
<td>The Queue Name.</td>
</tr>
<tr>
<td>agent</td>
<td>The Agent Name.</td>
</tr>
<tr>
<td>event</td>
<td>The name (type) of the last event. The parameters which follow depend on the event type.</td>
</tr>
<tr>
<td>ivrevent</td>
<td>The name (type) of the last event generated by IVR, like IVRSTART, IVRHANGUP, IVRAPPEND, IVRGOAL.</td>
</tr>
<tr>
<td>rtdtm</td>
<td>Last update date. Date of the last event.</td>
</tr>
<tr>
<td>rtttime</td>
<td>Last update time. Time of the last event.</td>
</tr>
<tr>
<td>url</td>
<td>URL. (if specified in the queue configuration).</td>
</tr>
<tr>
<td>callerid</td>
<td>Information about the caller.</td>
</tr>
<tr>
<td>enterqueue</td>
<td>Caller was placed in the queue. Flag(0/1). Default value = 1.</td>
</tr>
<tr>
<td>enterqueuedtm</td>
<td>Date of the enterqueue event.</td>
</tr>
<tr>
<td>enterqueuetime</td>
<td>Time of the enterqueue event.</td>
</tr>
<tr>
<td>holdtime</td>
<td>Represents the amount of time the caller was on hold.</td>
</tr>
<tr>
<td>bridgedchanneluniqueid</td>
<td>Contains the unique ID of the queue member channel that is taking the call.</td>
</tr>
<tr>
<td>ringtime</td>
<td>The time the queue members phone was ringing prior to being answered.</td>
</tr>
<tr>
<td>connect</td>
<td>Caller was connected to an agent. Flag(0/1). Default value = 1.</td>
</tr>
<tr>
<td>connectdtm</td>
<td>Date of the connect event.</td>
</tr>
<tr>
<td>connecttime</td>
<td>Time of the connect event.</td>
</tr>
<tr>
<td>calltime</td>
<td>The length of the call.</td>
</tr>
<tr>
<td>origposition</td>
<td>The caller’s original position in the queue.</td>
</tr>
<tr>
<td>completeagent</td>
<td>The caller was connected to an agent, and the call was terminated normally by the agent. Flag(0/1). Default value = 1.</td>
</tr>
<tr>
<td>completeagentdtm</td>
<td>Date of the completeagent event.</td>
</tr>
<tr>
<td>completeagenttime</td>
<td>Time of the completeagent event.</td>
</tr>
<tr>
<td>completecaller</td>
<td>The caller was connected to an agent, and the call was terminated normally by the caller. Flag(0/1). Default value 1.</td>
</tr>
<tr>
<td>completecalledtm</td>
<td>Date of the completecaller event.</td>
</tr>
<tr>
<td>completecalleertime</td>
<td>Time of the completecaller event.</td>
</tr>
</tbody>
</table>
### Annex - xCally DB Tables

<table>
<thead>
<tr>
<th>field</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>position</td>
<td>The caller abandoned their position in the queue.</td>
</tr>
<tr>
<td>waittime</td>
<td>How long the call had been waiting in the queue at the time of disconnect.</td>
</tr>
<tr>
<td>abandon</td>
<td>When a caller in a queue hangs up before his call is answered by an agent.</td>
</tr>
<tr>
<td>abandondtm</td>
<td>Date of the abandon event.</td>
</tr>
<tr>
<td>abandontime</td>
<td>Time of the abandon event.</td>
</tr>
<tr>
<td>exitempty</td>
<td>When the caller is removed from the queue due to a lack of agents available to answer the call (as specified by the leavewhenemptyparameter).</td>
</tr>
<tr>
<td>exitemptydtm</td>
<td>Date of the exitempty event.</td>
</tr>
<tr>
<td>exitemptytime</td>
<td>Time of the exitempty event.</td>
</tr>
<tr>
<td>exitwithkey</td>
<td>Written when the caller exits the queue by pressing a single DTMF key on his phone to exit the queue and continue in the dialplan.</td>
</tr>
<tr>
<td>exitwithkeydtm</td>
<td>Date of the exitwithkey event.</td>
</tr>
<tr>
<td>exitwithkeytime</td>
<td>Time of the exitwithkey event.</td>
</tr>
<tr>
<td>keypress</td>
<td>The key used to exit the queue.</td>
</tr>
<tr>
<td>exitwithtimeout</td>
<td>The caller is removed from the queue due to timeout.</td>
</tr>
<tr>
<td>exitwithtimeoutdtm</td>
<td>Date of the exitwithtimeout event.</td>
</tr>
<tr>
<td>exitwithtimeouttime</td>
<td>Time of the exitwithtimeout event.</td>
</tr>
<tr>
<td>transfer</td>
<td>When a caller is transferred to another extension.</td>
</tr>
<tr>
<td>transferdtm</td>
<td>Date of the transfer event.</td>
</tr>
<tr>
<td>transfertime</td>
<td>Time of the transfer event.</td>
</tr>
<tr>
<td>extension</td>
<td>The extension name to which the call is routed. This field it’s valid for the Transfer event.</td>
</tr>
<tr>
<td>context</td>
<td>The context name to which the call is routed. This field it’s valid for the Transfer event.</td>
</tr>
<tr>
<td>ivrstart</td>
<td>When the IVR starts. 1 = IVR started.</td>
</tr>
<tr>
<td>ivrstartdtm</td>
<td>Date of the ivrstart event.</td>
</tr>
<tr>
<td>ivrstarttime</td>
<td>Time of the ivrstart event.</td>
</tr>
<tr>
<td>ivrchoices</td>
<td>Which choices where made by the IVR Menu blocks.</td>
</tr>
<tr>
<td>ivrblocks</td>
<td>Which blocks of the IVR are involved.</td>
</tr>
<tr>
<td>ivrgoals</td>
<td>When the IVR is in the GOAL block.</td>
</tr>
<tr>
<td>ivrhangup</td>
<td>When the IVR is terminated.</td>
</tr>
<tr>
<td>ivrhangupdtm</td>
<td>Date of the ivrhangup event.</td>
</tr>
<tr>
<td>ivrhanguptime</td>
<td>Time of the ivrhangup event.</td>
</tr>
<tr>
<td>ivrttime</td>
<td>IVR duration (from the start to the end).</td>
</tr>
</tbody>
</table>
Annex - xCally DB Tables

<table>
<thead>
<tr>
<th>field</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ivrname</td>
<td>The IVR Project name.</td>
</tr>
<tr>
<td>duration</td>
<td>The number of seconds between the start and end times for the call.</td>
</tr>
<tr>
<td></td>
<td>This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>billsec</td>
<td>The number of seconds between the answer and end times for the call.</td>
</tr>
<tr>
<td></td>
<td>This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>origcalldate</td>
<td>The call date.</td>
</tr>
<tr>
<td>origcalltime</td>
<td>The call time.</td>
</tr>
<tr>
<td>origcallyear</td>
<td>The call year.</td>
</tr>
<tr>
<td>origcallmonth</td>
<td>The call month.</td>
</tr>
<tr>
<td>origcallday</td>
<td>The call day.</td>
</tr>
<tr>
<td>origcallhour</td>
<td>The call hour.</td>
</tr>
<tr>
<td>origcallmin</td>
<td>The call minutes.</td>
</tr>
<tr>
<td>origcallsec</td>
<td>The call seconds.</td>
</tr>
<tr>
<td>outboundcall</td>
<td>The type of the call is outbound. Flag(0/1).</td>
</tr>
<tr>
<td>outboundcalldtm</td>
<td>The date of the outboundcall event.</td>
</tr>
<tr>
<td>outboundcalltime</td>
<td>The time of the outboundcall event.</td>
</tr>
<tr>
<td>tag</td>
<td>If the outbound call has been tagged from the admin interface.</td>
</tr>
</tbody>
</table>

The call_log table is really useful to get information about call timing:

**HOLDTIME**  Answered calls  The amount of time the caller was on hold

**RINGTIME**  Answered calls  How long the phone (of the agent who answered) has rang.

**CALLTIME**  Answered calls  The length of the call.

**WAITTIME**  Unanswered calls  How long the call had been waiting in the queue at the time of disconnect.

**DURATION**  The number of seconds between the start and end times for the call.

**BILLSEC**  The number of seconds between the answer and end times for the call.
Annex - xCally DB Tables

CDR

This table is called *Call Detail Record* and contains all these fields:

<table>
<thead>
<tr>
<th>field</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>recid</td>
<td>Call Record ID.</td>
</tr>
<tr>
<td>calldate</td>
<td>Call Date &amp; Time.</td>
</tr>
<tr>
<td>cid</td>
<td>The full Caller ID, including the name, of the calling party. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>src</td>
<td>Source: the calling party’s caller ID number. It is set automatically and is read-only.</td>
</tr>
<tr>
<td>dst</td>
<td>The destination extension for the call. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>dcontext</td>
<td>The destination context for the call. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>channel</td>
<td>The calling party’s channel. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>dst channel</td>
<td>The called party’s channel. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>lastapp</td>
<td>The last dialplan application that was executed. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>lastdata</td>
<td>The arguments passed to the lastapp. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>duration</td>
<td>The number of seconds between the start and end times for the call. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>billsec</td>
<td>The number of seconds between the answer and end times for the call. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>disposition</td>
<td>An indication of what happened to the call. This may be NO ANSWER, FAILED, BUSY, ANSWERED, or UNKNOWN.</td>
</tr>
<tr>
<td>amaflags</td>
<td>The Automatic Message Accounting (AMA) flag associated with this call. This may be one of the following: OMIT, BILLING, DOCUMENTATION, or Unknown.</td>
</tr>
<tr>
<td>accountcode</td>
<td>An account ID. This field is user-defined and is empty by default.</td>
</tr>
<tr>
<td>sequence</td>
<td>A field that can be combined with uniqueid and linkedid to uniquely identify a CDR.</td>
</tr>
<tr>
<td>tag</td>
<td>It has a value if the call has been tagged from the admin interface.</td>
</tr>
<tr>
<td>uniqueid</td>
<td>The unique ID for the src channel. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>linkedid</td>
<td>A unique identifier based on uniqueid. Unlike uniqueid, but spreads to other channels as transfers, dials, etc are performed</td>
</tr>
</tbody>
</table>
## Annex - xCally DB Tables

<table>
<thead>
<tr>
<th>field</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>userfield</td>
<td>The channel's user specified field.</td>
</tr>
<tr>
<td>start</td>
<td>The start time of the call. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>answer</td>
<td>The answered time of the call. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>end</td>
<td>The end time of the call. This field is set automatically and is read-only.</td>
</tr>
<tr>
<td>rtdtm</td>
<td>Last update date. Date of the last event.</td>
</tr>
<tr>
<td>rtttime</td>
<td>Last update time. Time of the Last Event.</td>
</tr>
</tbody>
</table>
**Annex - xCally DB Tables**

**QUEUE_LOG**

This table contains all the events regarding Queues.

<table>
<thead>
<tr>
<th>field</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Event ID.</td>
</tr>
<tr>
<td>time</td>
<td>Timestamp of the event.</td>
</tr>
<tr>
<td>callid</td>
<td>Unique Call ID.</td>
</tr>
<tr>
<td>queuename</td>
<td>Queue Name.</td>
</tr>
<tr>
<td>agent</td>
<td>Name of bridged channel.</td>
</tr>
<tr>
<td>event</td>
<td>Event Type.</td>
</tr>
<tr>
<td>data</td>
<td>Data of the Event; they depend on the event type.</td>
</tr>
<tr>
<td>dtm</td>
<td>Date and time of the last event.</td>
</tr>
</tbody>
</table>

The following table contains the list of the possible queue_log events and their relative data:

<table>
<thead>
<tr>
<th>event</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABANDON</td>
<td>Written when a caller in a queue hangs up before his call is answered by an agent.</td>
</tr>
<tr>
<td></td>
<td>data 1  the position of the caller at hangup</td>
</tr>
<tr>
<td></td>
<td>data 2  the original position of the caller when entering the queue</td>
</tr>
<tr>
<td></td>
<td>data 3  the amount of time the caller waited prior to hanging up</td>
</tr>
<tr>
<td>ADDMEMBER</td>
<td>Written when a member is added to the queue.</td>
</tr>
<tr>
<td></td>
<td>The bridged channel name will be populated with the name of the channel added to the queue.</td>
</tr>
<tr>
<td>AGENTDUMP</td>
<td>Indicates that the agent hung up on the caller while the queue announcement was being played, prior to them being bridged together.</td>
</tr>
<tr>
<td>AGENTLOGIN</td>
<td>Recorded when an agent logs in. The bridged channel field will contain something like Agent/9994 if logging in withchan_agent.</td>
</tr>
<tr>
<td></td>
<td>data 1  the channel logging in (e.g., SIP/0000FFFF0001)</td>
</tr>
<tr>
<td>AGENTLOGOFF</td>
<td>When an agent logs off.</td>
</tr>
<tr>
<td></td>
<td>data 1  how long the agent was logged in for</td>
</tr>
<tr>
<td>BACKOFFICETIME</td>
<td>Recorded when the agent has a backoffice Pause</td>
</tr>
<tr>
<td></td>
<td>data 1  the Backoffice Time Pause duration</td>
</tr>
<tr>
<td></td>
<td>data 2  the Backoffice Timestamp Pause start time</td>
</tr>
<tr>
<td></td>
<td>data 3  the Backoffice Timestamp Pause end time</td>
</tr>
</tbody>
</table>
### Annex - xCally DB Tables

<table>
<thead>
<tr>
<th>event</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMPLETEAGENT</td>
<td>Recorded when a call is bridged to an agent and the agent hangs up. The position at which the caller entered the queue. &lt;br&gt;data 1 - the amount of time the caller was held in the queue &lt;br&gt;data 2 - the length of the call with the agent &lt;br&gt;data 3 - the original position at which the caller entered the queue.</td>
</tr>
<tr>
<td>COMPLETECALLER</td>
<td>Same as COMPLETEAGENT, except the caller hung up and not the agent.</td>
</tr>
<tr>
<td>CONFIGRELOAD</td>
<td>Indicates that the queue configuration was reloaded (e.g., via module reload app_queue.so).</td>
</tr>
<tr>
<td>CONNECT</td>
<td>Written when the caller and the agent are bridged together. &lt;br&gt;data 1 - the amount of time the caller waited in the queue &lt;br&gt;data 2 - the unique ID of the queue member’s channel to which the caller was bridged &lt;br&gt;data 3 - the amount of time the queue member’s phone rang prior to being answered.</td>
</tr>
<tr>
<td>ENTERQUEUE</td>
<td>Written when a caller enters the queue. &lt;br&gt;data 1 - the URL (if specified) &lt;br&gt;data 2 - the caller ID of the caller &lt;br&gt;data 3 - the time the call has been in the queue</td>
</tr>
<tr>
<td>EXITWITHKEY</td>
<td>Written when the caller exits the queue by pressing a single DTMF key on his phone to exit the queue and continue in the dialplan. &lt;br&gt;data 1 - the key used to exit the queue &lt;br&gt;data 2 - the position of the caller in the queue upon exit &lt;br&gt;data 3 - the original position the caller entered the queue at &lt;br&gt;data 4 - the amount of time the caller was waiting in the queue.</td>
</tr>
<tr>
<td>EXITEMPTY</td>
<td>Written when the caller is removed from the queue due to a lack of agents available to answer the call (as specified by the leavewhenempty parameter). &lt;br&gt;data 1 - the position of the caller in the queue &lt;br&gt;data 2 - the original position at which the caller entered the queue &lt;br&gt;data 3 - the amount of time the caller was held in the queue.</td>
</tr>
</tbody>
</table>
### Annex - xCally DB Tables

<table>
<thead>
<tr>
<th>event</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>EXITWITHTIMEOUT</strong></td>
<td>Written when the caller is removed from the queue due to timeout</td>
</tr>
<tr>
<td></td>
<td>data 1  the position the caller was in when exiting the queue</td>
</tr>
<tr>
<td></td>
<td>data 2  the original position of the caller when entering the queue</td>
</tr>
<tr>
<td></td>
<td>data 3  the amount of time the caller waited in the queue.</td>
</tr>
<tr>
<td><strong>INFO</strong></td>
<td>It contains data about IVR</td>
</tr>
<tr>
<td></td>
<td>data 1  the IVR event: IVRSTART</td>
</tr>
<tr>
<td></td>
<td>data 2  CallerID</td>
</tr>
<tr>
<td></td>
<td>data 3  DID</td>
</tr>
<tr>
<td></td>
<td>data 4  Context (from-sip)</td>
</tr>
<tr>
<td></td>
<td>data 5  IVR Project name</td>
</tr>
<tr>
<td></td>
<td>data 1  IVRAPPEND</td>
</tr>
<tr>
<td></td>
<td>data 2  pressed key</td>
</tr>
<tr>
<td></td>
<td>data 3  IVR block Menu name</td>
</tr>
<tr>
<td><strong>LOGINTIME</strong></td>
<td>The Agent login on a queue</td>
</tr>
<tr>
<td></td>
<td>data 1  The login time duration</td>
</tr>
<tr>
<td></td>
<td>data 2  The login start timestamp</td>
</tr>
<tr>
<td></td>
<td>data 3  The login end timestamp</td>
</tr>
<tr>
<td><strong>MOH</strong></td>
<td>Music on hold Event</td>
</tr>
<tr>
<td></td>
<td>data 1  The Music on Hold duration in seconds</td>
</tr>
<tr>
<td><strong>NOTINUSE</strong></td>
<td>When the agent is logged on a queue but he doesn’t manage calls</td>
</tr>
<tr>
<td></td>
<td>data 1  The Not in Use duration</td>
</tr>
<tr>
<td></td>
<td>data 2  The Not in Use start time</td>
</tr>
<tr>
<td></td>
<td>data 3  The Not in Use end time</td>
</tr>
<tr>
<td><strong>PAUSE</strong></td>
<td>Written when a queue member is paused.</td>
</tr>
<tr>
<td></td>
<td>data 1  The Pause Type</td>
</tr>
<tr>
<td><strong>PAUSEALL</strong></td>
<td>Written when all members of a queue are paused.</td>
</tr>
<tr>
<td><strong>PAUSETIME</strong></td>
<td>data 1  The Pause duration</td>
</tr>
<tr>
<td></td>
<td>data 2  The Pause start timestamp</td>
</tr>
<tr>
<td></td>
<td>data 3  The Pause end timestamp</td>
</tr>
<tr>
<td><strong>PENALTY</strong></td>
<td>Written when a member’s penalty is modified. Manager Interface, or the Asterisk CLI commands.</td>
</tr>
<tr>
<td></td>
<td>The penalty can be changed through several means, such as the QUEUE_MEMBER_PENALTY() function, through using Asterisk</td>
</tr>
</tbody>
</table>
### Annex - xCally DB Tables

<table>
<thead>
<tr>
<th>event</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QUEUESTART</td>
<td>The queue was started.</td>
</tr>
<tr>
<td>REMOVEMEMBER</td>
<td>Written when a queue member is removed from the queue. The bridge channel</td>
</tr>
<tr>
<td></td>
<td>field will contain the name of the member removed from the queue.</td>
</tr>
<tr>
<td>RINGNOANSWER</td>
<td>Logged when a queue member is rung for a period of time, and the timeout</td>
</tr>
<tr>
<td></td>
<td>value for ringing the queue member is exceeded. data 1 the amount of time</td>
</tr>
<tr>
<td></td>
<td>the member’s extension rang</td>
</tr>
<tr>
<td>SYSCOMPAT</td>
<td>Recorded if an agent attempts to answer a call, but the call cannot be</td>
</tr>
<tr>
<td></td>
<td>set up due to incompatibilities in the media setup.</td>
</tr>
<tr>
<td>TRANSFER</td>
<td>Written when a caller is transferred to another extension. data 1 the</td>
</tr>
<tr>
<td></td>
<td>extension and context the caller was transferred to data 2 the hold time</td>
</tr>
<tr>
<td></td>
<td>of the caller in the queue data 3 the amount of time the caller was</td>
</tr>
<tr>
<td></td>
<td>speaking to a member of the queue data 4 the original position of the</td>
</tr>
<tr>
<td></td>
<td>caller when he entered the queue</td>
</tr>
<tr>
<td>UNPAUSE</td>
<td>Written when a queue member is unpaused. data 1 The Pause Type</td>
</tr>
<tr>
<td>UNPAUSEALL</td>
<td>Written when all members of a queue are unpaused.</td>
</tr>
</tbody>
</table>