



telefaks* application server for FreeSWITCH

Peter Steinbach

Mein50Plus GmbH
Theo-Geisel-Str. 25
Usingen, Germany, 61250
Tel.: +49 6081 688 533
www.telefaks.de
Information@telefaks.de



Introduction

- This presentation is all about telephony services for
 - VoIP
 - POTS
 - Google Talk / Jabber
 - Messaging
- Building telephony services based on
 - OpenSource software
 - Standard server hardware



Who we are

- Coming from Asterisk
- On Freeswitch since beg. of June 2008
- Transferred all our applications to Freeswitch since then
- Strong focus on
 - Integrating Freeswitch
 - Ruby and Rails Development
 - Encryption



Freeswitch- the new swiss knife for VoIP (1)

- FreeSWITCH is a new alternative to Asterisk
- Developed by people who wanted to have a better code base compared to Asterisk and a better and more flexible structure
- Advantages
 - Call volume per server (3000+)
 - Configuration by XML instead of sometimes difficult Asterisk-Syntax
 - Higher stability at high call volumes
 - Better central administration by webservice
 - Several virtual PBXs on one server
 - Simpler call routing in bigger installations
 - Encryption via TLS and SRTP (currently the only OpenSource solution)
- Disadvantages
 - General available GUI missing, configuration via XML files
 - Not as established on the market compared to Asterisk (but more stable in produktion)
- Outlook
 - Will become one of the standards for larger installations



Freeswitch- the new swiss knife for VoIP (2)

telefaks*de

Can be used as:

- VoIP-Switch
- VoIP-Router
- IVR-System
- Phone conference server
- PBX
- B2BUA(Back to back user agent)
- Session border controller
- Basic Topology Hiding Session Border Controller,
- Application Server (VoiceMail, Konferenz, IVR)
- Integration platform
- Register proxy

The logo for Freeswitch, featuring a blue stylized antenna icon above the word "FreeSWITCH" in a bold, blue, sans-serif font.

FreeSWITCH



Freeswitch- the new swiss knife for VoIP (3)

telefaks*de

Availability:

- Mostly all Linux platforms
- Sun Solaris / OpenSolaris
- Windows
- Mac OS X
- BSD



Freeswitch- the new swiss knife for VoIP (4)

Key points

- Scalability
- Built-in redundancy mechanisms
- Supports a number of communication protocols (incl. Jabber und Skype)
- Encryption of Voice (SRTP) and call setup (TLS)
- Voice codecs up to 48KHz
- A number of interfaces for configuration and call control (synchronous and asynchronous), perfect for dynamic call routing
- Word recognition (Sphinx)
- Text-To-Speech via Cepstral TTS



Freeswitch-Highlights (1)

Skalability

- ~ 3000 simultaneous Calls including media
- Factor >> 10 with media outside Freeswitch
- Built-in redundancy mechanisms via XML-Curl for configuration and call control





telefaks* application server

Why an application server framework?

- Our Freeswitch projects usually have a larger scale than e.g. an Asterisk PBX
- A single Freeswitch is per default configured by XML files
- On top there exists a number of interfaces for configuration and synchronous/asynchronous call control
- Integrating large projects therefore requires a lot of groundwork to be done
- Some nice GUIs exist already, each one targeting a dedicated scenario (e.g. PBX, Callcenter)
- however, a system which will cover all scenarios by 100% will most probably never exist



Bottom line

We need a framework
to abstract functionalities for
integrating large Freeswitch projects



What is basically needed for that?

- Administration GUI
- Handling of more than one freeswitch server
- Customer hierarchies
- IVR functionalities
- Callcenter support
- Asynchronous call handling
- Realtime interface with web browser (e.g. push status)



What is it built of

- Freeswitch of course
- some Ruby processes for interfacing with Freeswitch
- Ruby on Rails for the web interface
- Javascript and AJAX for the web interface
- a bit of LUA
- a push server

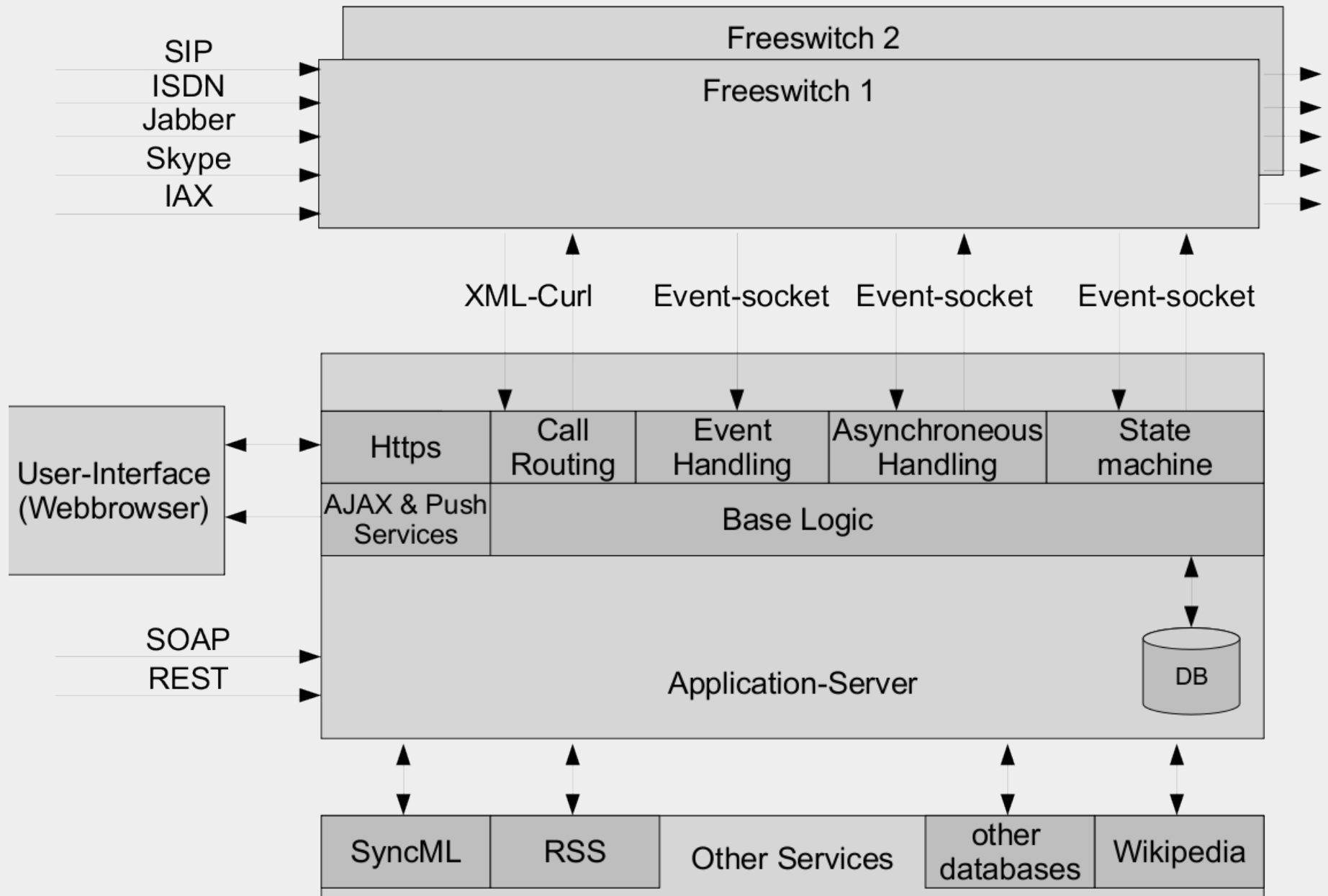


What ist covers

- Support of multiple Freeswitch servers
- Basic PBX functionalities (is needed almost everywhere)
- Conferencing (setup and „live“ management)
- Call Queues
- Callback/dialthru
- IVR State machine with setup via GUI
- Callcenter workflows with direct interaction between browser and freeswitch
- TTS and ASR Support
- Encryption of calls (TLS/SRTP)
- Complex routing algorithms for larger networks
- Prepared for billing functionalities
- Channel Spy
- Custom applications
- Interface to SyncML

... more

How it's designed





PBX functionalities

Sample PBX functionalities

- Serve multiple clients
- Clients can be spread over multiple instances of Freeswitch
- User administration with client hierarchies
- Management of SIP endpoints
- Voicemail
- Call forwarding (parallel + sequential hunting)
- Short numbers for each endpoint
- One-time numbers (or n times usage), obfuscated numbers
- Dialthru/Callback
- Special numbers
- Conferences
- Call queues
- Encryption TLS/SRTP
- ... more



Sample PBX functionalities

Telefaks Freeswitch Management

Login: peter Role: Super Admin(Telefaks)

Phone Numbers/Conferences

- Operator Panel
- Phone numbers
- Voicemails
- User Parameter
- Masked numbers
- Short numbers
- Special numbers
- Dialthrus
- Conferences
- Active Conferences
- Queues
- SIP registrations

Customers

- Routing
- Callcenter
- Basic Setup
- IVR State Machine
- Billing
- Freeswitch Mgt.
- Test XML requests
- Memcache
- System Status

Editing directory

Customer id
Telefaks

Exten
835331

Password

Gateway
sip5.telefaks.biz

Fullname
Peter Steinbach

External CID
06081688533

Enable direct callforward

Direct callforward to
06081688533

Voicemail
835331

Vm-Password

Vm-Email
steinbach@telefaks.biz

SyncML User
peter_s

Forwards

835331 835333
01.71.336

All numbers entered in one line will be called altogether.
Numbers in the following lines will be called in sequence.
Call timeout is 15 seconds.
Call timeout for the last line is 30 seconds.
If you enter any numbers here, do not forget to add the current extension also.

Available Numbers

for exten number

Customer	Exten_from	Exten_to
Telefaks	83533	8353399999
Internal	99998	99998
	9999999900	9999999999
Mein50Plus	26824	2682499999
	83534	8353499999

Sample Conferencing functionalities

- Conference definition

Telefaks Freeswitch Management

Phone

Numbers/Conferences

- Operator Panel
- Phone numbers
- Voicemails
- User Parameter
- Masked numbers
- Short numbers
- Special numbers
- Dialthrus
- Conferences
- Active Conferences
- Queues
- SIP registrations

Customers

Routing

Callcenter

Basic Setup

IVR State Machine

Billing

Freeswitch Mgt.

Test XML requests

Memcache

System Status

Editing conference

Host

sip5.telefaks.biz

Customer

Mein50Plus

Conference description

Sales

Number

26824200

Conference-type

Conference 8KHz en ComfortNoise EnergyLevel 3000

Valid from

2008 September 26 - 13 : 33

Valid to

2009 November 27 - 13 : 33

Active

Yes

Pin

Kick all members out of the conference after initiator hangs up

No

Record whole conference

No

Conference Numbers to invite

No	Extension	Active	Originator
1	835331	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	01712 2022	<input checked="" type="checkbox"/>	<input type="checkbox"/>
3	06081688533	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4		<input type="checkbox"/>	<input type="checkbox"/>
5		<input type="checkbox"/>	<input type="checkbox"/>
6		<input type="checkbox"/>	<input type="checkbox"/>
7		<input type="checkbox"/>	<input type="checkbox"/>

Available number ranges

for Conferences

Customer	Range_from	Range_to
Telefaks	83533	8353399999
Internal	99998	99998
	9999999900	9999999999
Mein50Plus	26824	2682499999
	83534	8353499999

Sample Conferencing functionalities

- Conference live management

Telefaks Freeswitch Management

Login: peter Role: Super Admin(Telefaks)

Phone Numbers/Conferences

- Operator Panel
- Phone numbers
- Voicemails
- User Parameter
- Masked numbers
- Short numbers
- Special numbers
- Dialthrus
- Conferences
- Active Conferences
- Queues
- SIP registrations

Customers

- Routing
- Callcenter
- Basic Setup
- IVR State Machine
- Billing
- Test XML requests
- Test with https
- Freeswitch
- Memcache
- System Status

Listing active_conferences

Conference

Conference Name	Record conference	Conference lock	Conference PIN	Send data to all members	Invite into conference
83533200	<input checked="" type="checkbox"/>	<input type="checkbox"/>	PIN	<input type="text" value="conference/8000/conf-welcome.wav"/> <input type="button" value="play"/> <input type="text"/> <input type="button" value="Speak"/> <input type="text"/> <input type="button" value="DTMF"/>	<input type="text"/> <input type="button" value="INVITE"/>

Conference members

Member	Member's Speaker	Member's Mikrophone	Energy Level	Kick out	Send data to this member	Transfer member
835331 TLS SRTP	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	0 Set	<input type="checkbox"/>	<input type="text" value="conference/8000/conf-welcome.wav"/> <input type="button" value="play"/> <input type="text"/> <input type="button" value="Speak"/> <input type="text"/> <input type="button" value="DTMF"/>	<input type="button" value="select conf & transfer"/> <input type="text"/> <input type="button" value="enter conf & transfer"/> <input type="text"/> <input type="button" value="enter no & transfer"/>
835333 TLS SRTP	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	<input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/> <input type="checkbox"/>	0 Set	<input type="checkbox"/>	<input type="text" value="conference/8000/conf-welcome.wav"/> <input type="button" value="play"/> <input type="text"/> <input type="button" value="Speak"/> <input type="text"/> <input type="button" value="DTMF"/>	<input type="button" value="select conf & transfer"/> <input type="text"/> <input type="button" value="enter conf & transfer"/> <input type="text"/> <input type="button" value="enter no & transfer"/>



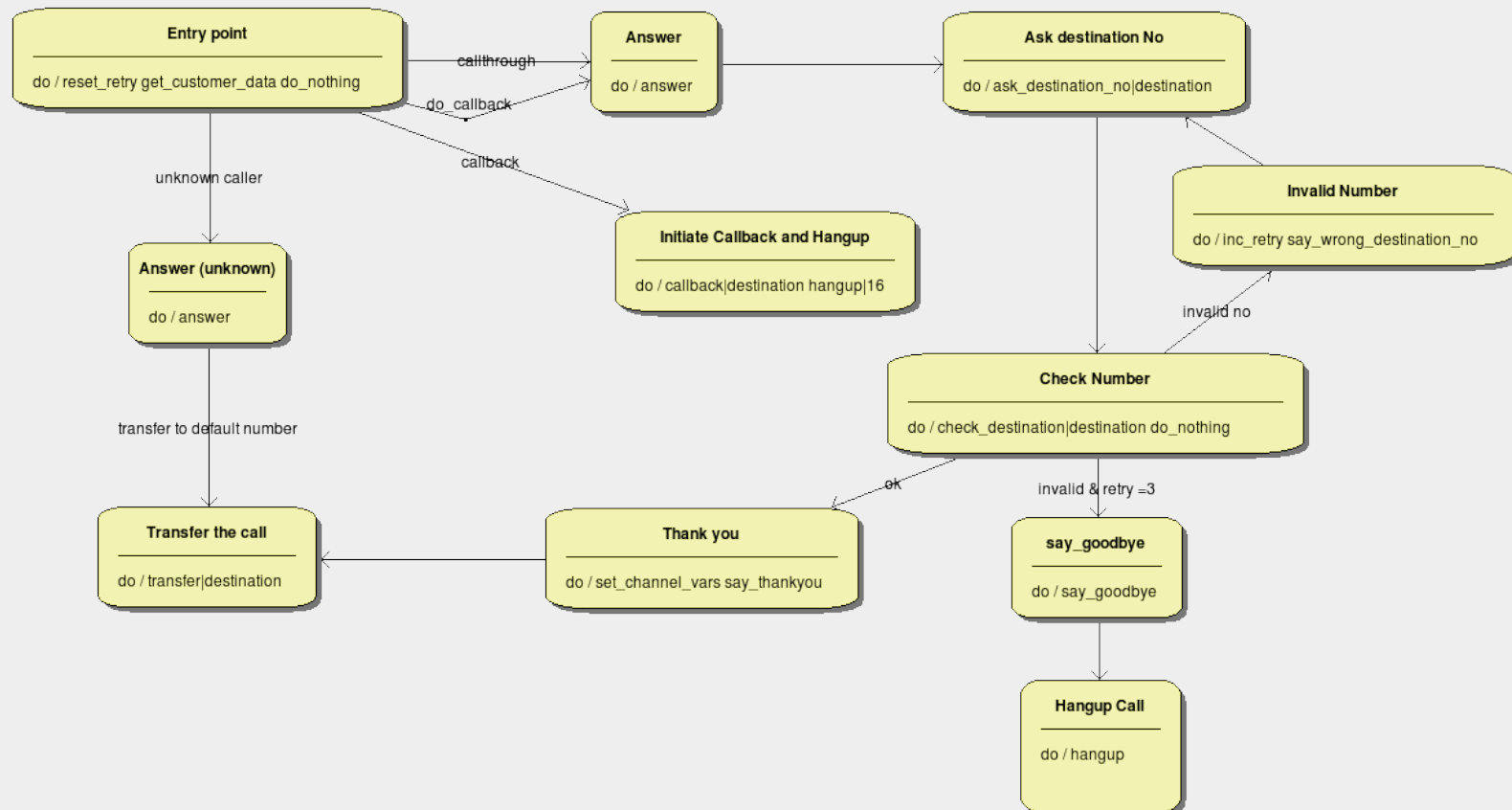
IVR functionalities

IVR Callback and Callthru application

Step1: Draw the workflow

Goal:

- Identify client/caller
- Hangup, then store callback number if client is callback customer
- Next step: callback to the client
- Offer to enter target number via DTMF and connect the call



IVR Functionalities

- Built-in state machine for defining IVRs and other workflows
- IVRs are defined the following way:
 - Step 1: Draw the callflow as UML state diagramm
 - define actions
 - define transitions
 - Step 2: Upload UML state diagram to the application server
 - Step 3: Specify actions for each state on the web GUI
 - Step 4: Test the state machine on the web GUI (html)
 - Step 5: Take the state machine into production (now with voice)



IVR Functionalities

- Interaction with the caller
 - Play sound files or external sound streams (play multiple files and variables)
 - Text to speech
 - Read DTMF
 - Voice menus (DTMF)
 - Record users voice and playback later
 - Word recognition (ASR)
 - Answer a call
 - Hangup a call
 - Dial a number
 - Transfer a call
 - Numerous customized actions
- ++ Numerous asynchroneous actions during a call
- early media mode for some actions



IVR Callback and Callthru applikation

Step2: Specify actions in detail

Action name

ask_destination_no

Tts text

Please enter the destination Number (10 or 11 digits)

Voice files (separate multiple sound files by linefeeds)

welcome.wav
you-are-using.wav
\$service\$.wav
please-enter-num-to-call.PCMU

Do TTS? (Otherwise play sound files)

No of Digits when asked for Input

11

Interruptable by keypress?

Hear Params



IVR Callback and Callthru applikation

Step 2: Test workflow on the web browser



State: 128312 "Ask destination No"

Compare: No conditions



Executed: ask_destination_no|destination



Input:

Play zigit/welcome.wav

Play zigit/you-are-using.wav

Play zigit/callback.wav

Play zigit/please-enter-num-to-call.PCMU

Input:



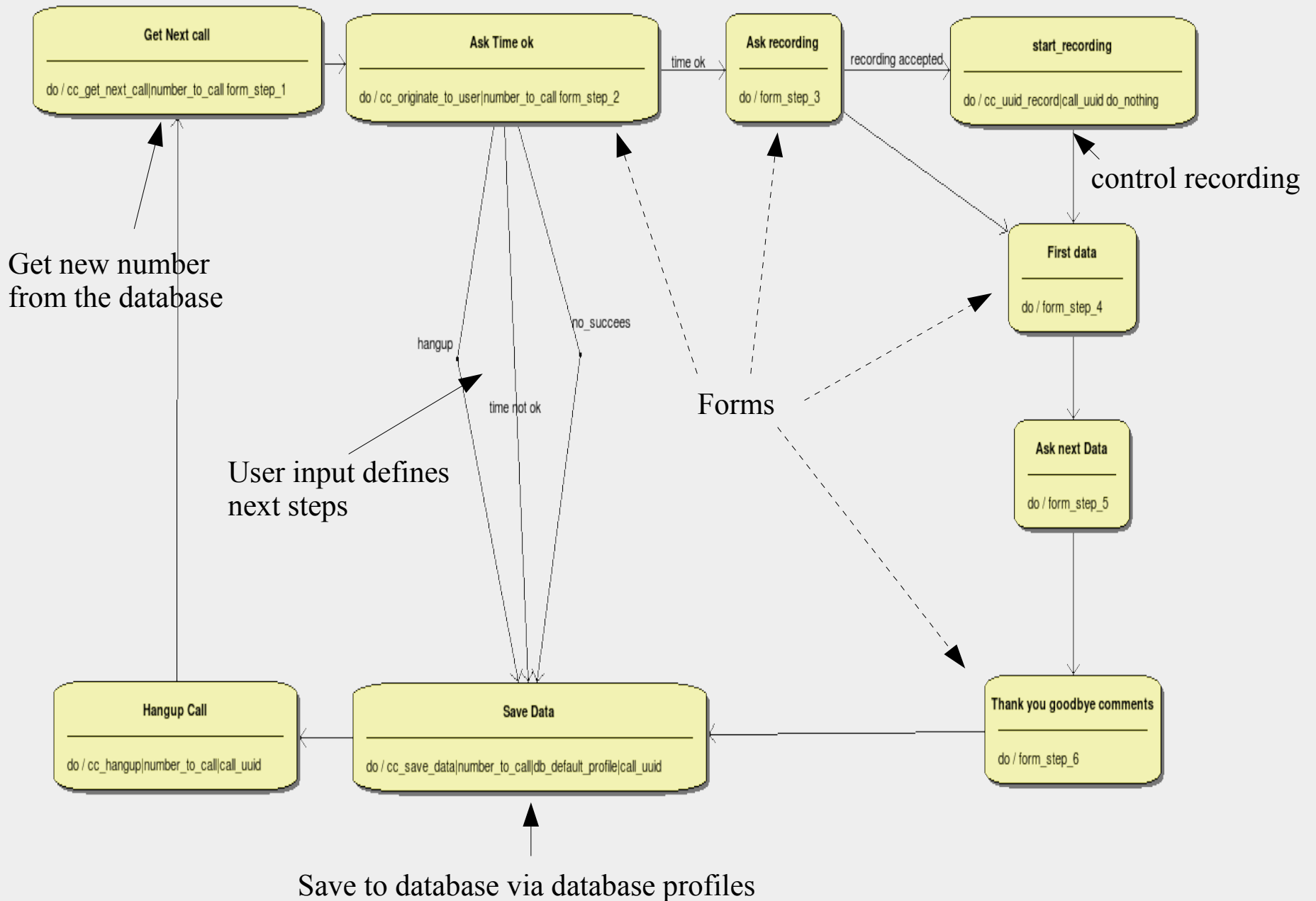


Callcenter functionalities



- Extension to IVR Application
- Webbrowser initiates actions on Freeswitch
- Freeswitch pushes data to the web browser (AJAX push services)
- Interactions to Freeswitch
 - Dial a number from a database
 - Answer a call
 - Play messages
 - Start recording
 - Stop recording
 - Forward call
 - Hangup Call
- Push services to the web browser
 - Show status of a call
 - Alert incoming calls
 - Open CRM window

Sample callcenter application: Step 1: Define Workflow



Sample callcenter application: Step 2: Define Forms

Callcenter form assistant

Enter variable name to be used in processing:

favourite

Please select control type:

select

Enter description text for this control:

What is your favourite brand to buy a vehicle

Enter parameters for this control:

Volvo,Mercedes,DAF,Iveco,MAN,Toyota,VW,Audi,BMW

Enter default options for this control:

Volvo

Show controls on Form

Add control to form

Delete last control

Delete all controls

Submit form & close

Define new form elements

Form preview

Ask for new procurements

When are you planning to buy your next vehicles?

- not_planned_yet
- 2009
- 2010
- 2012
- 2013
- later

What is your favourite brand to buy a vehicle

Volvo

Customer history:

Date	User	Campaign	FollowUp	Comment
------	------	----------	----------	---------

Preview new form

Sample callcenter application: Step 3: Run workflow



Telefaks Freeswitch Management

- Phone
- Numbers/Conferences
- Customers
- Routing
- Callcenter
- Basic Setup
- IVR State Machine
- Test XML requests
- Test with https
- Freeswitch
- Memcache
- System Status

Callcenter form state: 128626 "Ask next Data"

Compare: No conditions

Executed: form_step_5

Ask for new procurements

When are you planning to buy your next vehicles?

- not_planned_yet
- 2009
- 2010
- 2012
- 2013
- later

What is your favourite brand to buy a vehicle

Volvo

Customer history:

Date	User	Campaign	FollowUp	Comment
2009-06-09 15:01:35	peter	poll_cars		Completed
2009-06-08 13:07:07	peter	poll_cars	2009-06-09 15:00	
2009-06-07 13:06:29	peter	poll_cars	2009-06-08 13:00	

submit

← History



Push services

Push services

- every GUI user has an assigned phone number
- web browser registers on this phone number
- web browser gets status pushed from Freeswitch
 - Example: successful hangup

Status	Incoming Call from: 723321(Peter Steinbach FS) and IP 217.11.186 Event: CHANNEL_EXECUTE_COMPLETE, state: CS_HANGUP	Login: peter Role: Super Admin Subscribed to phone#: 723321
--------	---	--

Telefaks Freeswitch Management Login: peter Role: Super Admin

Phone Numbers/Conferences
Phone numbers

Listing directories
New directory entry (exten)

- Incoming call:

Status	Call from: 835333(835333) and IP 217.11.186 Event: CHANNEL_PROGRESS, state: CS_CONSUME_MEDIA	Click here to manage this call in the Callcenter Application Click here to manage in CRM	Login: peter Role: Super Admin Subscribed to phone#: 835331
--------	---	---	--

- Active call:

Status	Call from: 723321(Peter Steinbach FS) and IP 217.11.186 Event: CHANNEL_ANSWER, state: CS_EXECUTE	Login: peter Role: Super Admin Subscribed to phone#: 723321
--------	---	--





Customizing your application

Call Routing with regular expressions



- Phone Numbers/Conferences
- Customers
- Routing**
 - Gateways
 - Gateway-types
 - Gateway aliases
 - Host_gateways
 - Routings
 - Routings1
 - Routings2
 - Routings3
 - Routings4
 - Routings5
 - Dialplans
 - XML Parameter
- Callcenter
- Basic Setup
- IVR State Machine
- Billing
- Freeswitch Mgt.
- Test XML requests
- Memcache
- System Status

Editing routing

Profile
1

Description
German Provider QSC International

Active
No

Dialplan
Next Gateway insecure

Context from
default

Gateway from
sip5.telefaks.biz

Time Interval
whole day

Sort Id
0

Number From (enter regular expression)
`^(00[1-9]\d{4,13})$`

Context to
default

Gateway to
QSC_07141

Number To (enter fixed number, or \$1 for the dialled number or \$target_number\$ for the replaced number)
\$target_number\$

Number To Match (Regex which shall apply to the finally dialled number, leave empty if no change shall apply)
`^00([1-9]\d{4,13})$`

Comment
Strip off 00 from 0049xxx

Call handling via templates

```
<!-- start a generic conference with the settings of the "default" conference profile -->
<!-- Target No $target_number$ -->
<extension name="conference $conference_name$"
  <condition field="destination_number" expression="^\d+$">
    <action application="set" data="dialplan_comment=$dialplan_comment$"/>
    <!-- this is filled up with external participants and a hangup hook if needed -->
    $conference_invitations$
    <action application="answer"/>
    <action application="send_display" data="Conference $1"/>
    <action application="conference" data="$conference_number@$context$"/>
  </condition>
</extension>
```

- Application server defines additional variables
- Variables are expanded at runtime

Customizing your own applications

Example: Wikipedia

- Special numbers can be used to trigger own dialplan actions
- dialplan actions can be XML templates or customized Ruby code

Telefaks Freeswitch Management

Phone Numbers/Conferences

- Operator Panel
- Phone numbers
- Voicemails
- User Parameter
- Masked numbers
- Short numbers
- Special numbers
- Dialthrus
- Conferences
- Active Conferences
- Queues
- SIP registrations

Customers

- Routing
- Callcenter
- Basic Setup
- IVR State Machine
- Billing
- Test XML requests
- Test with https
- Freeswitch
- Memcache
- System Status

Editing special_number

Customer

Telefaks

Number

83533405

Description

Speak Wikipedia

Active



Show on panel



Dialplan

Execute

Execute (select Execute dialplan before when using

```
Custom.speak_wikipedia("Frankfurt_am_Main")
```

Update

[Show](#) | [Back](#)

Available number ranges

for Special numbers

Customer	Range_from	Range_to
Telefaks	83533	8353399999
Internal	99998	99998
Telefaks_public	9999999900	9999999999
Mein50Plus	26824	2682499999
Telefaks_private	83534	8353499999

Customizing your own applications

Example: Wikipedia

```
def self.speak_wikipedia(search_exp)
  text=self.get_wikipedia_text(search_exp)
  master="<action application=\"speak\" data=\"cepstral|katrin|$text$\"/>\n"
  erg= "<!-- Wikipedia entry to speak: '#{search_exp}' -->\n"
  if text
    text.each do |line|
      if !line.strip.empty?
        erg+=master.gsub("$text$", line)
      end
    end
  end
  erg
end
```

```
<!-- Wikipedia entry to speak: 'Dresden' -->
<action application="speak" data="cepstral|katrin|Dresden. "/>
<action application="speak" data="cepstral|katrin|aus Wikipedia, der freien Enzyklopaedie. "/>
<action application="speak" data="cepstral|katrin|512.234 (31. Dez. 2008). "/>
<action application="speak" data="cepstral|katrin|Dresden (sorbisch: Drježdźany; abgeleitet aus dem altsorbischen
Dreždany fuer Sumpf- oder Auwaldbewohner) ist die Landeshauptstadt des Freistaates Sachsen. Sie liegt in der Dresdner
Elbtalweitung an den Uebergaengen vom Ober- zum Mittellauf der Elbe und von der Mittelgebirgsschwelle zum
Norddeutschen Tiefland.. "/>
<action application="speak" data="cepstral|katrin|Archaeologische Spuren auf dem spaeteren Stadtgebiet deuten auf
eine Besiedlung schon in der Steinzeit hin. In erhaltenen Urkunden wurde Dresden 1206 erstmals erwaeht und entwickelte
sich zur kurfuerstlichen und koeniglichen Residenz.. "/>
<action application="speak" data="cepstral|katrin|Dresden ist das politische Zentrum des Freistaates, der in Dresden
seine staatlichen Kultureinrichtungen konzentriert. Es hat den Status einer kreisfreien Stadt, ist Sitz des Direktionsbezirks
Dresden und zahlreicher Hochschulen. Dresden bildet den Kern des gleichnamigen Ballungsgebietes in Mitteleuropa und ist
dadurch Verkehrsknotenpunkt und ein wirtschaftliches Zentrum. Die Region gilt als eine der wirtschaftlich dynamischsten in
Deutschland. Zusammen mit den Ballungsraeumen Chemnitz-Zwickau sowie Leipzig-Halle bildet dieser Ballungsraum die
„Metropolregion Sachsendreieck“.. "/>
<action application="speak" data="cepstral|katrin|Die Stadt wird auch „Elbflorenz“ genannt, urspruenglich wegen ihrer
```

Some examples for customizing

- Wikipedia as shown before
- Speak selected content of news sites
- Speak RSS feeds
- Speak file contents
- Speak meter values from external interfaces
- Access calendar from SyncML (Funambol)
- Intercom, global announcements
- Reverse internet CID lookup



Performance

- using caching techniques wherever applicable
 - „Memcache“ allows distributed caching over multiple servers
- Tested under High Load
 - up to 250 call setups per second out of the box on a Dual Core AMD 2,5GHz (caching enabled)
 - up to 160 call setups per second out of the box on a Dual Core AMD 2,5GHz (caching disabled)
- Outlook:
 - scales well with the number of processors (processes are CPU intensive)
 - scales well with the number of machines (http cluster techniques used)
 - Further performance improvement with Ruby 1.9 and optimized, self-compiled Ruby binaries



Thank you!

telefaks*de

Peter Steinbach
steinbach@telefaks.biz

Hans-Jürgen Bornhorst
bornhorst@telefaks.biz

