

Linux Telephony Overview

Featuring Asterisk Open Source PBX

Presented By

Tim Clark

Independent Consultant

Presentation Overview

- Overview of Telephony/VoIP
- Review Linux Telephony Applications
- Introduction to Asterisk Architecture
- Asterisk PBX Features
- The Asterisk Dialplan/Configuration
- Programmability/extensibility
- Supported Hardware
- Q & A, Hands on Demo

Telephony

Telephony is ***NOT*** VoIP

- TDM circuit switched dedicated point to point
- TDM-Time Division Multiplexing eg T1, E1 etc
- Circuits can be POTS, ISDN(BRI, PRI) etc
- Bearer protocol/CODEC only G711 at 64k
- CODEC -COmpressor / DECompressor
- Devices are FXO or FXS (RJ-11 jack or phone)
- FXO, FXS Foreign Exchange Office/Station
- PBX- Private Branch Exchange

VoIP

- VoIP is routed packet data
- VoIP Signal Protocols MGCP, H323, SIP, IAX etc
- Bearer Protocol-CODEC G723, G726, G729, G711, GSM, Speex .. sent with Real Time Transport Protocol (RTP) using UDP datagrams (the actual voice packet stream)
- Point to Point VoIP Signal/Bearer must be same
- A IP Switch can convert Signal/CODEC
- FXO device does not exist in VoIP world

VoIP Hardware

- IP Phones
- -SIP CISCO 7900's-, PingTel, H323 Dlink etc
- Analog Terminal Adapters-VoIP to POTS phone
- -ATA186, VoIP Blaster, Quicknet, Mulitech etc
- Gateway Devices (ata with more ports/protocols)
- -variety ports, codec's, & protocol support
- IP / Hybrid PBX's
- Cisco, Avaya, Shoreline, Mitel, 3Com, Asterisk

Linux Telephony Clients

- Pt to pt VoIP single protocol, uses mic/speakers
- gnophone.com: IAX based for use with asterisk
- openh323.org: Oh-phone H323 client
- linphone.org: SIP based GUI
- fobbit.org: uses VoIP Blaster
- Others SJPhone, Kphone, gPhone, SIPSet, Siphon, Free Phone, Speak Freely, Nautilus, Ethernet phone, Voicechat, efone,

Linux Telephony Servers

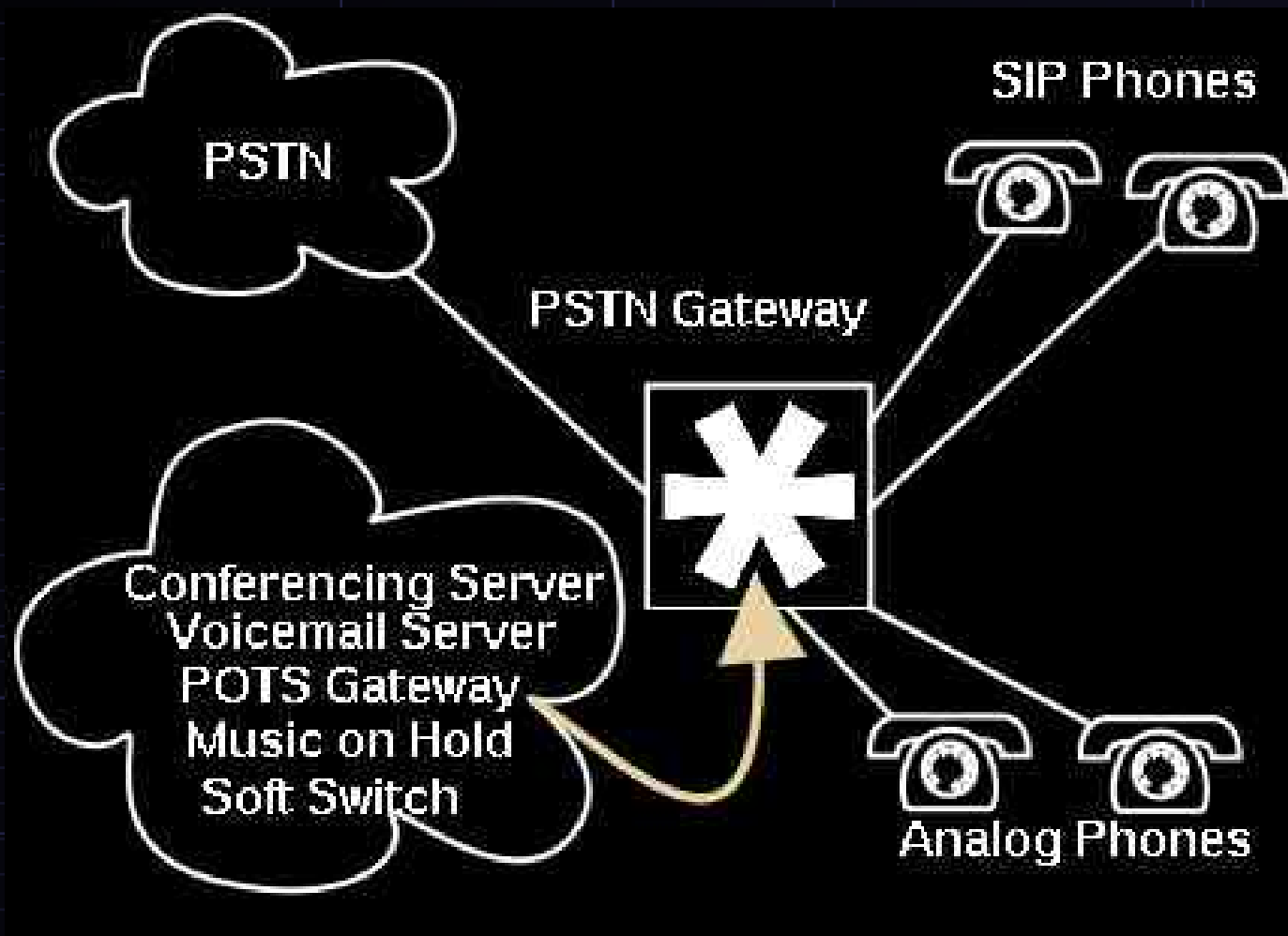
- asterisk.org : hybrid PBX, softswitch, IVR
- bayonne.sourceforge.net : scriptable IVR, softswitch application server not a PBX
- vovida.org : Open Communication Application Library (Vocal), VoIP library helpers (SIP, MGCP, RTP) & servers (STUN, Conference)
- [Openh323.org](http://openh323.org) :
Tool Kit/Lib for h323
-OPAL-Open Phone Abstraction Layer

Introduction to Asterisk

Asterisk is an Open Source hybrid TDM and packet voice PBX and IVR platform with ACD functionality...

Asterisk is fully Open Source
Written in C not C++. GPL, & Non-GPL Lic's

The Big Picture



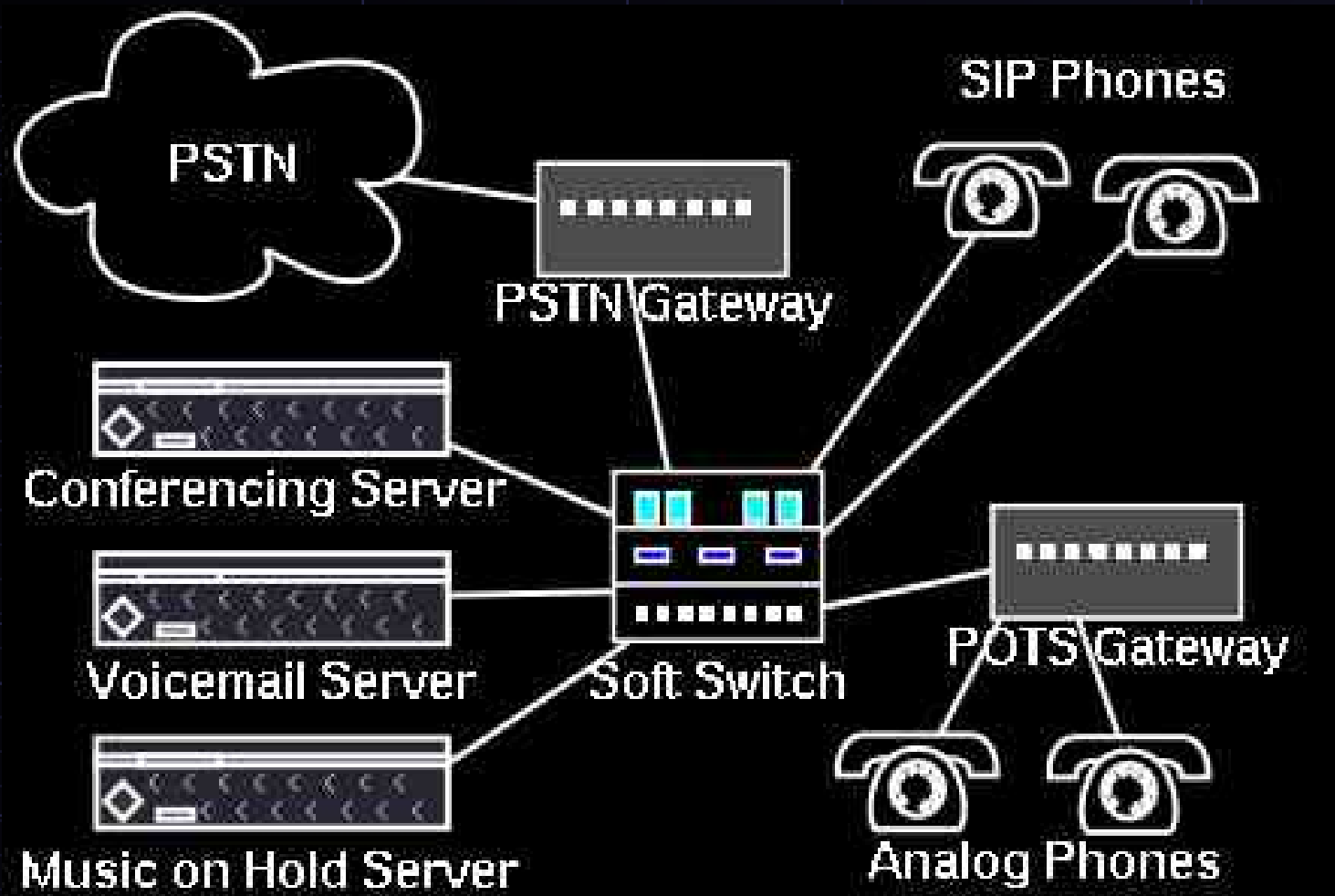
Asterisk as a Black Box



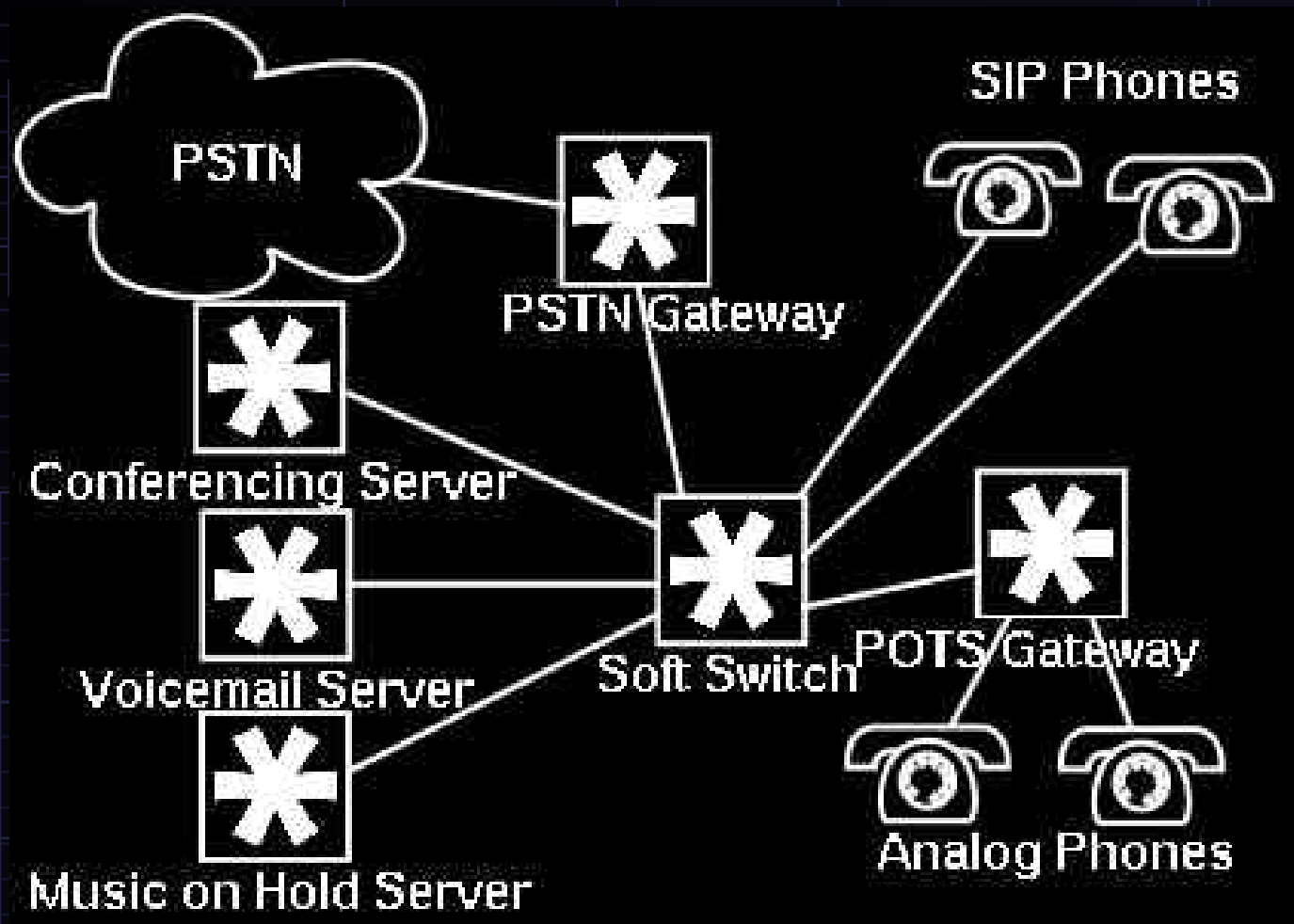
Asterisk connects any telephone, telephone line, specialized telephony circuit, or VoIP interface to any other interface or service through Asterisk Applications.

Asterisk is the middleware between the telephony interfaces and the applications themselves

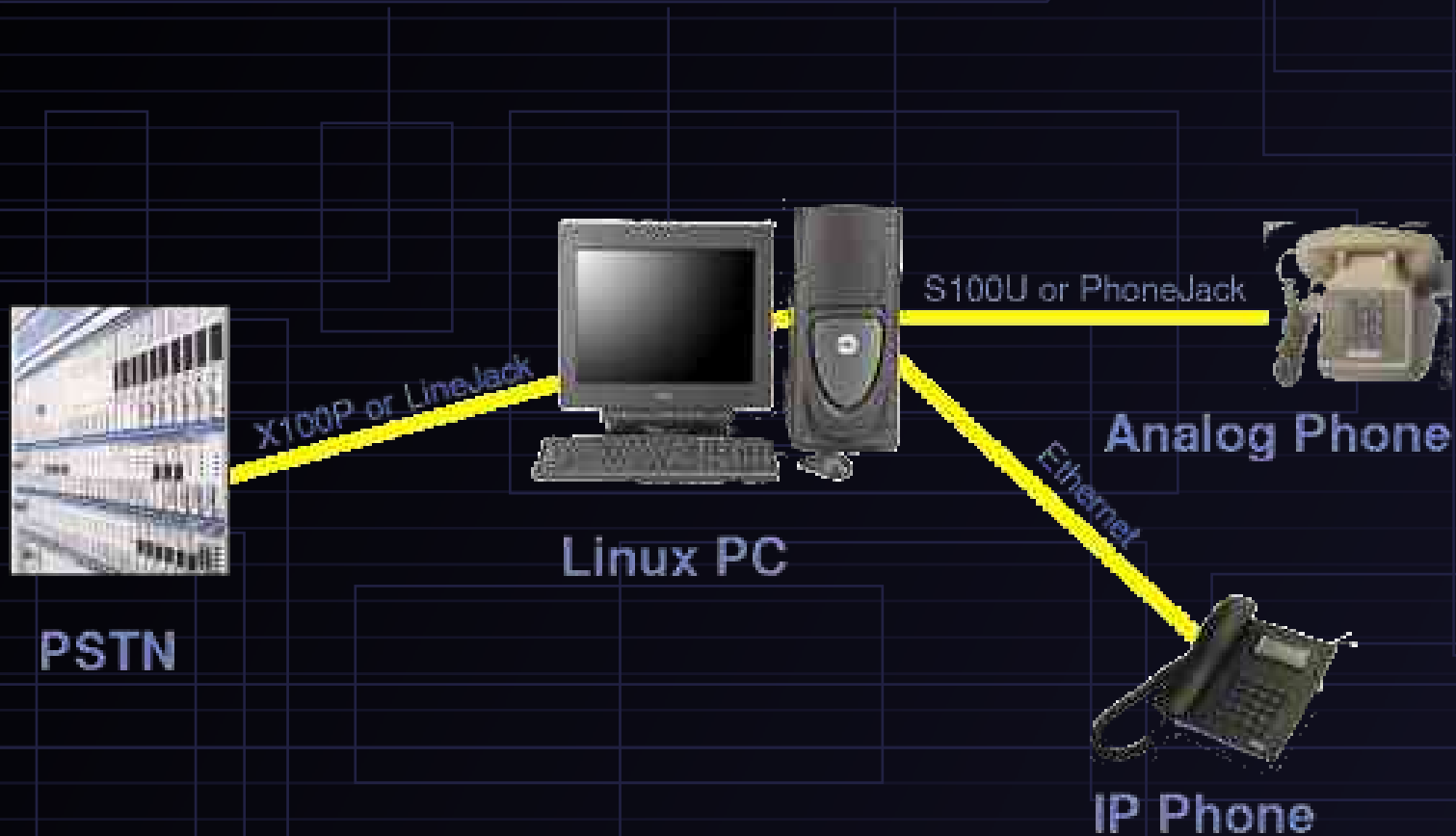
Conventional Softswitch Network



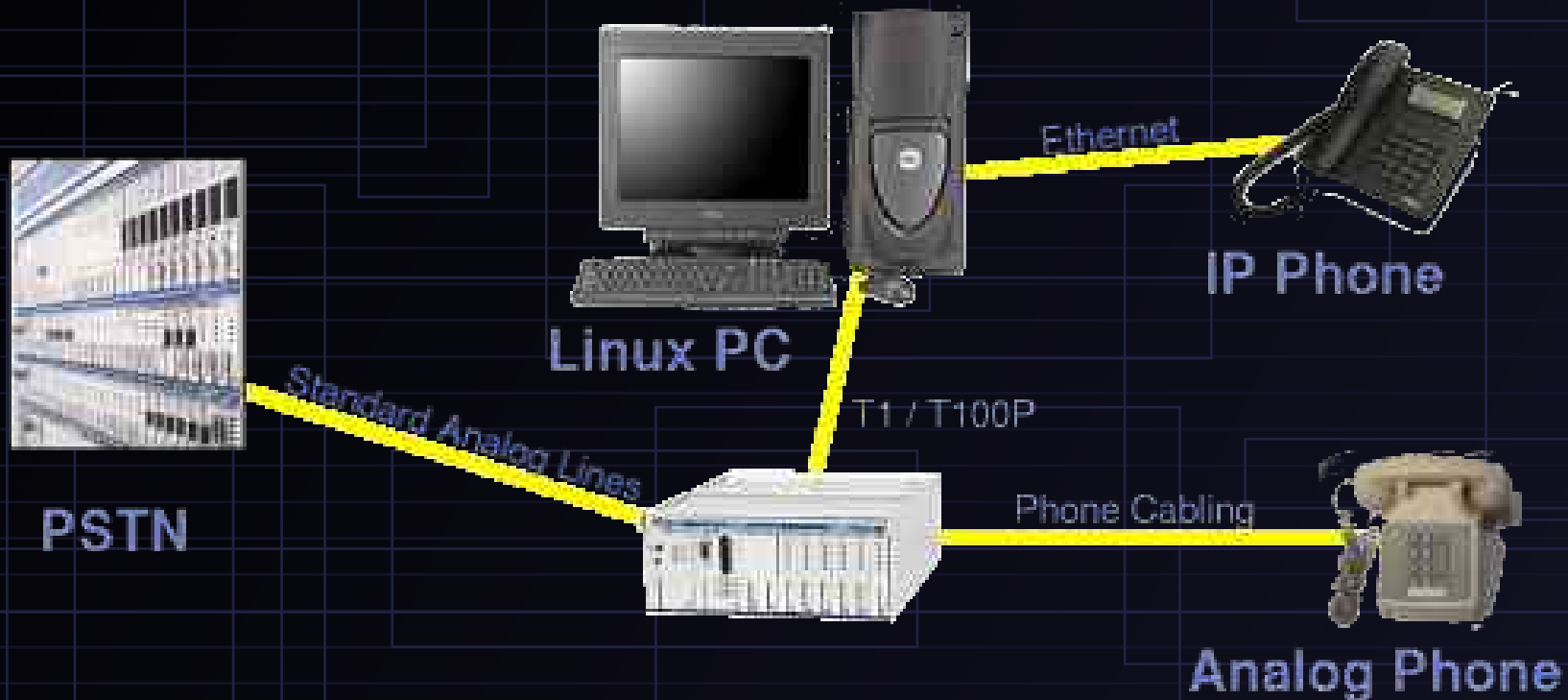
Where Asterisk Fits



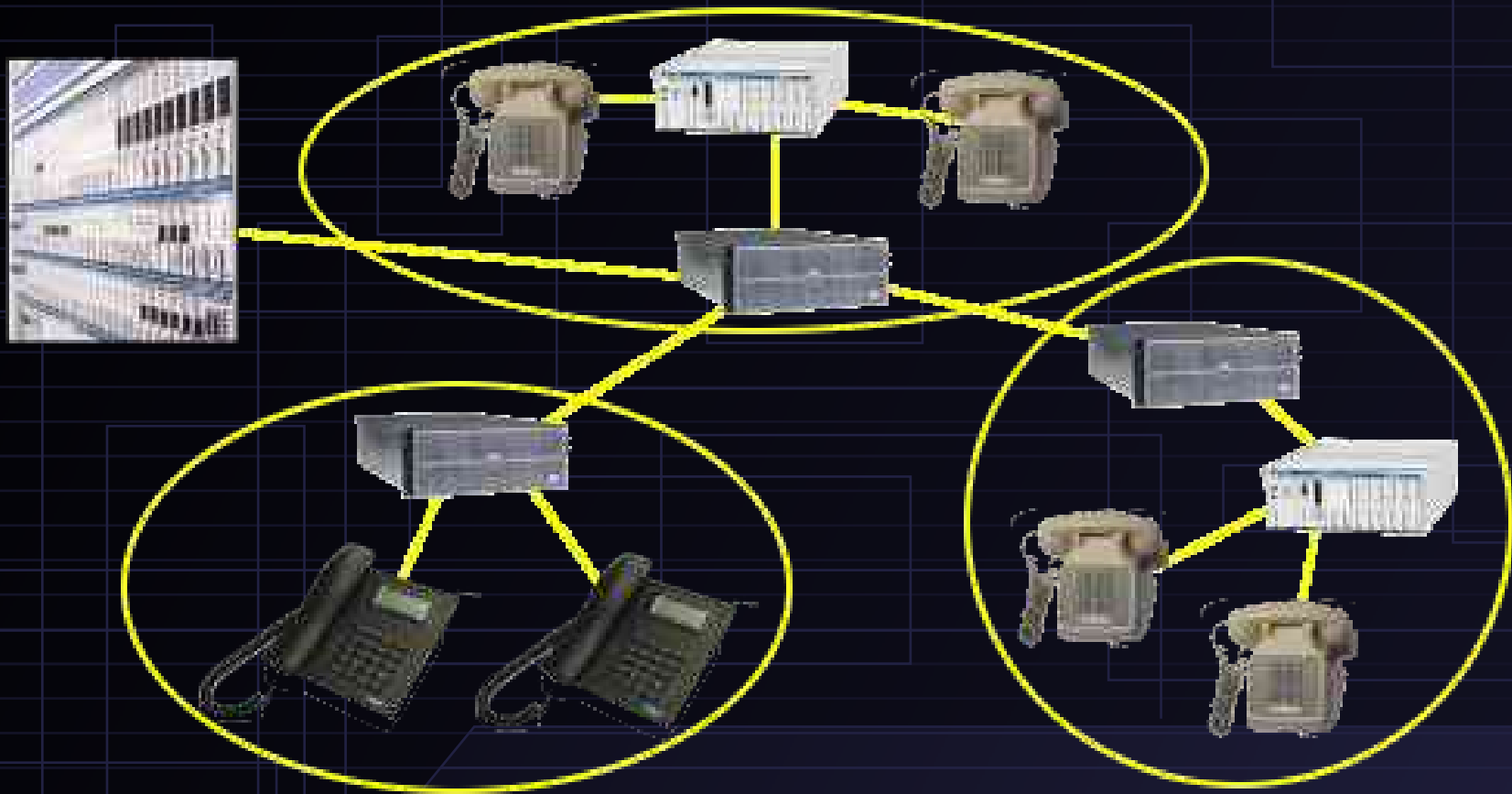
Example: 1x1 PBX



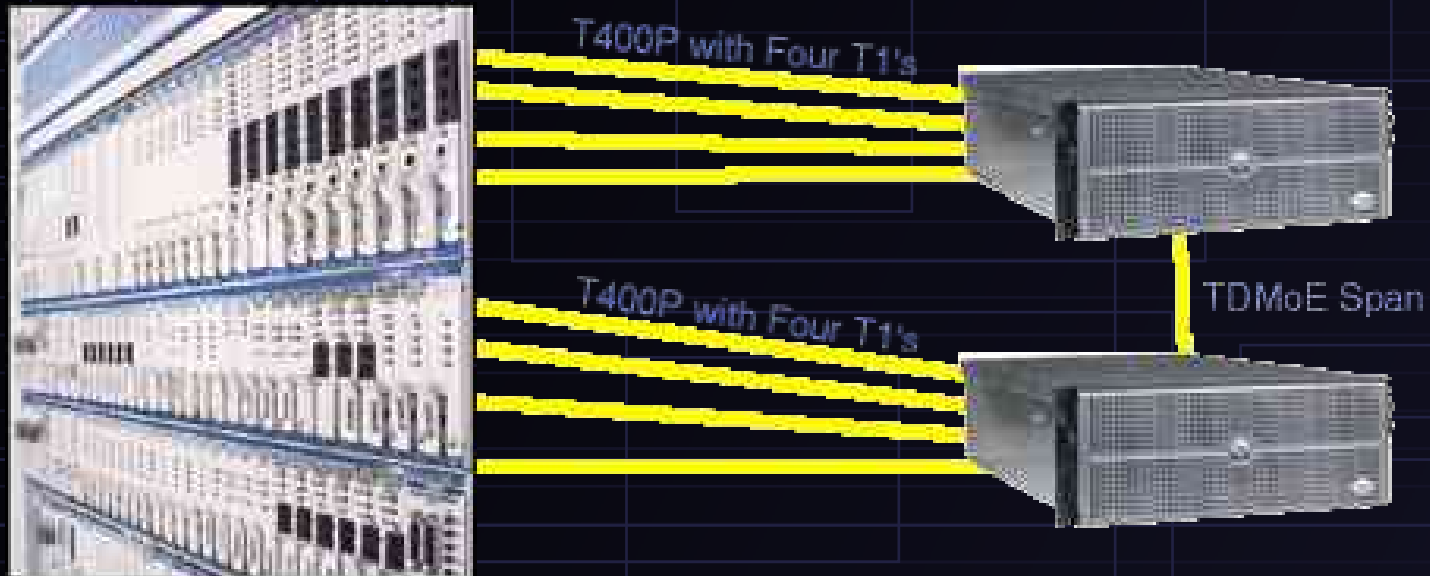
Example: 8x16 PBX



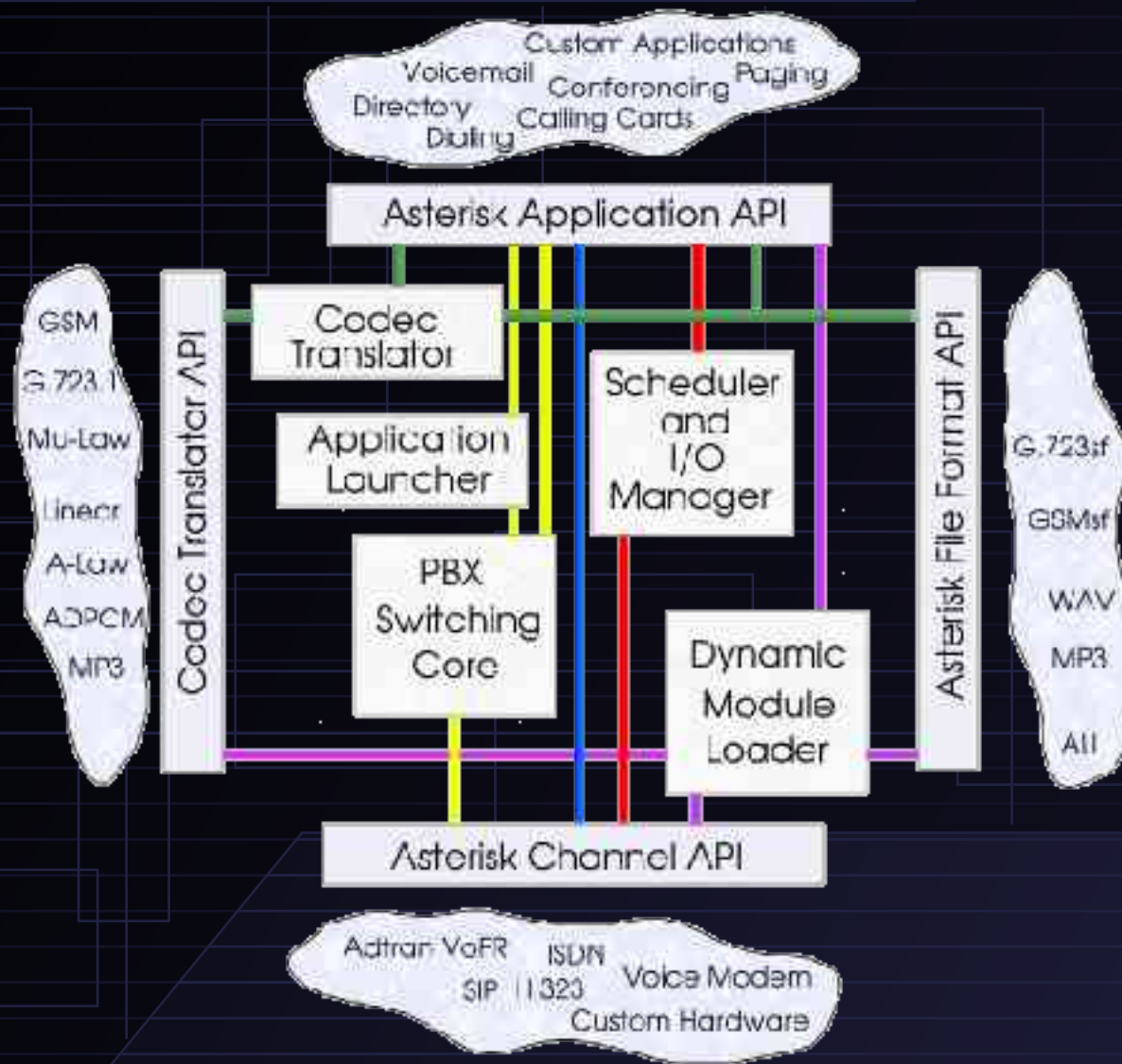
Small/Medium Business



High Density IVR/Conferencing



Asterisk Architecture



Asterisk Features

- Asterisk is a Private Branch Exchange (PBX)
 - Seamlessly supports VoIP, Digital, & Analog channels
 - Standard call features like:
 - 3-way calling
 - Caller*ID/Caller*ID + Call Waiting
 - Call Waiting
 - Dynamic Call Forwarding, Follow Me Call Forwarding
 - Advanced features like:
 - Voicemail
 - Conferencing
 - Interactive Voice Response (IVR) system
 - Automatic Call Distribution (ACD)

Asterisk Features (in English)

- Asterisk supports traditional circuits
 - TDM (Time Division Multiplexing)
 - T1/E1 PRI/PRA & RBS (Robbed Bit Signal) modes
 - Analog phone lines/phones (POTS)
 - ISDN (Integrated Services Digital Network)
 - Both BRI (Basic Rate) and PRI (Primary Rate)

Asterisk Features (continued)

- Asterisk supports packet voice (VoIP)
 - Voice over Frame Relay
 - Session Initiation Protocol (SIP)
 - H.323 (ITU standard, contributed support)
 - Inter-Asterisk eXchange (IAX)
 - Media Gateway Control Protocol (MGCP)
- Asterisk seamlessly integrates VoIP & TDM

Asterisk Features(continued)

- Asterisk is an Interactive Voice Response (IVR) platform
 - Hardware independent API's, internally and externally
 - Powerful C-level API
 - Asterisk Gateway Interface (AGI) similar to CGI
 - Programming can be done with extension logic alone
- Asterisk has some ACD functionality
 - Call queues and remote agents
- Asterisk IS the Apache of the PBX world :)

The Asterisk Dialplan

- Directs routing of all calls through Asterisk
- Composed of extension contexts
- Contexts are groups of extensions
- Contexts can include one another
- Each step in the Dial Plan is an Application
- Each step is assigned a priority sequence

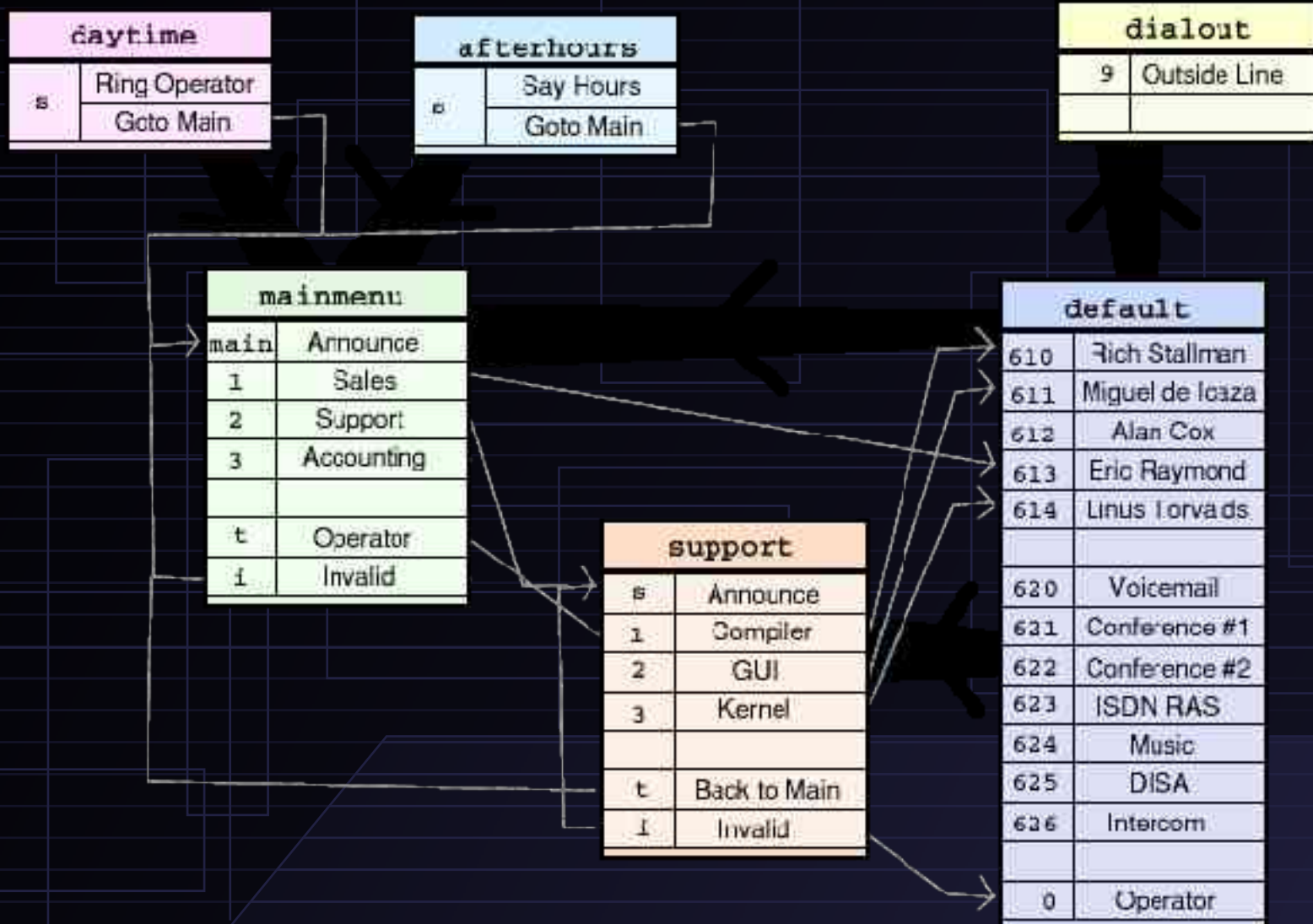
Extension Contexts

- Group of extensions
- Extensions may exist in more than one context
- Contexts can include other contexts
- Macros can simplify similar extensions
- External switches may augment dial plan
 - IAX switch can pull dialplan from another IAX host
 - MySQL can pull dial plan from a Database

Uses of Extension Contexts

- Creative use of contexts can be used for:
 - Security
 - Voice Menus
 - Authenticated Services
 - PBX Multi-Hosting
 - Callback Services
 - Daytime/Nighttime Modes
 - Speed Dials
 - Call Forwards
 - Direct Inward System Access (DISA)

Example Asterisk Dialplan



Extensions

- Conventional dialplans assign extension to physical interface
- Asterisk assigns extensions to list of applications
- Execution goes through priorities, one by one
- Applications can modify call flow
- Seamlessly integrates IVR apps into PBX
- May route by dialed and calling number

Extension Names

- Exten => Name, Priority, Application(, arguments)
- May be any number of digits (max 128)
- May be string literals
- May pattern match (when preceeded with '_')
 - 'N' – matches digits from 2 to 9
 - 'X' – matches digits from 0 to 9
 - '.' – signifies end of pattern matching

Special Extensions

- 's' the start extension, where calls begin when no digits are available
- 't' the timeout extension, where calls begin when not enough digits are entered
- 'i' the invalid extension, when an invalid extension is entered
- 'o' the operator extension
- 'h' the hangup extension, when channel is disconnected
- 'fax' the fax extension id cng tone detected

Applications-Major

Over 70 applications currently

VoiceMail: Enter voicemail system

MeetMe: Simple MeetMe conference bridge

MusicOnHold: Play Music On Hold

Directory: Directory of voicemail extensions

Monitor: Record a channel

DISA: DISA (Direct Inward System Access)

Festival: Say text to the user

AGI: Executes an AGI compliant application

Authenticate: Authenticate a user

PrivacyManager: Require Phone# if no CallerID sent

Zapateller: Block telemarketers with SIT

Dial Application-Helpers

Dial: Make a call and connect the current channel

Answer: Answer a channel if ringing

BackGround: Play a audio file while awaiting dtmf

Goto: Goto a priority, extension, or context

GotoIf: Conditional goto

Local: virtual channel driver for extensions

DB???: **PUT,GET,DELE**let from the database

SubString: Save substring digits in a given variable

SetCallerID: Set CallerID

SetVar: Set local/global variable to value

?Timeout: Absolute, Response,& Digit max time of call

Hangup: Unconditional hangup

Congestion: Indicate congestion and stop

System: Execute a system command

Simple Extension w/ Voicemail

```
exten => 600,1,Dial(Zap/9,15)
```

```
exten => 600,2,VoiceMail(u600)
```

```
exten => 600,102,VoiceMail(b600)
```

Try dialing on Zap/9 for up to 15 seconds. If there is no answer send them to voicemail, preceded by “unavailable” message. If the interface is busy, send them to voicemail with a “busy” message.

Extension with Tax Man

```
Exten => 600/8009595525,1,Congestion  
exten => 600,1,Dial(Zap/9,15)  
exten => 600,2,Voicemail(u600)  
exten => 600,102,Voicemail(b600)
```

If the Caller*ID matches the Tax man, provide immediate congestion tone. Otherwise try dialing on Zap/9 for up to 15 seconds. If there is no answer send them to voicemail, preceded by 'unavailable' message. If the interface is busy, send them to voicemail with a 'busy' message.

Example: Operator

```
exten => 0,1,Dial(Zap/9,15)
```

```
exten => 0,2,Dial(Zap/10&Zap/11&Zap/12,15)
```

```
exten => 0,3,Playback(companymailbox)
```

```
exten => 0,4,Voicemail(600)
```

```
exten => 0,5,Hangup
```

Dial on Zap/9 for up to 15 seconds. If 9 isn't available, try 10, 11, and 12 for another 15 seconds. If there's still no answer, announce that they can leave a message in the company mailbox, and then dump them in extension 600's mailbox with no announcement.

Example: Least Cost Routing (LCR)

```
exten => _9NXXXXXX,1,Dial(IAX/oh/${EXTEN})  
exten => _9NXXXXXX,2,Dial(Zap/g2/${EXTEN:1})  
exten => _9NXXXXXX,3,Congestion
```

Dial VoIP through other IAX host oh (including the nine). If that's busy, try connecting the call viaon any Group 2 interface, this time dropping the 9 . Finally generate congestion tone.

Example: AGI Integration

```
exten => s,1,AGI(agi-lookup.agi)
```

```
exten => s,2,Background(intro)
```

```
exten => 100,1,AGI(agi-save.agi)
```

```
exten => 100,2,Dial(Zap/9,15)
```

```
exten => 100,3,Voicemail(u100)
```

```
exten => 120,1,AGI(agi-save.agi)
```

```
exten => 120,2,Dial(Zap/24,15)
```

```
exten => 120,3,Voicemail(u101)
```

When an incoming call comes in, send it first to “agi-lookup.agi” which checks the caller's Caller*ID against a database and route them to the last person they spoke with.

Example: Ringback

```
[ringback]
exten => s,1,Ringing
exten => s,2,Wait(1)
exten => s,3,Congestion
exten => h,1,System(callback)

[default]
exten => 1234,1,DISA(4321,trusted)
exten => 1235,1,Goto(ringback,s,1)
```

When extension 1235 is dialed, the call is sent into the 'ringbak' context which eventually calls a script to call the user and dump them into the DISA (with a password) when they answer.

Example: Macro

```
[macro-stdexten]
```

```
exten => s,1,Dial(${ARG1},20)
```

```
exten => s,2,Voicemail(u${ARG2})
```

```
exten => s,102,Voicemail(b${ARG2})
```

```
[default]
```

```
exten => 1234,1,Macro(stdexten,Zap/1,1234)
```

```
exten => 1235,1,Macro(stdexten,Zap/2&Zap/3,1234)
```

Extension 1234 and 1235 are essentially identical, dialing one (and two) interfaces for up to 20 seconds, and doing voicemail.

VoiceMail application takes a mailbox # argument, we assign exten # & voicemailbox the same number

Configuration Files

<code>/etc/asterisk</code>	-- Location of config files
<code>alsa.conf</code> and <code>oss.conf</code>	-- Sound card configuration
<code>asterisk.adsi</code>	-- ADSI feature script
<code>extensions.conf</code>	-- Static dialplan
<code>iax.conf</code>	-- IAX users, peers, friends, and parameters
<code>logger.conf</code>	-- Log files, log levels, etc.
<code>meetme.conf</code>	-- Meetme conference configuration
<code>modules.conf</code>	-- Modules, preloads, globals, and noloads
<code>musiconhold.conf</code>	-- Music on hold configuration
<code>voicemail.conf</code>	-- Voicemail mailboxes, general parameters
<code>zapata.conf</code>	-- Zaptel TDM compatible device configuration

Config File Syntax

- Ironically, closest to win.ini config format
- Comments delimited by ';' instead of '#'
- Sections begins with name in square brackets
- Individual configuration lines take two formats
 - variable = value
 - object => parameter
- Grammar categories
 - Interface, Simple Groups, Individual Entities

Interface Config: zapata.conf

```
[channels]
callerid = "Joe Foo" <(250) 123-4567>
signalling = fxo_ks
threewaycalling = yes
channel => 9
```

```
callerid = "V LUG" <(250) 428-6236>
callwaiting = no
channel => 10
callerid = asreceived
signalling = fxs_ks
channel => 1-8
```


Simple Groups: extensions.conf

```
[globals]
; global variables
FRONTDESK=Zap/1
SALES1=Zap/2

[context]
;exten => extension,priority,app[options]
exten => s,1,Wait(3)
exten => s,2,Answer
exten => s,3,Dial(${FRONTDESK},20)
exten => s,4,Voicemail(u6275)
```

Individual Entities: iax.conf

```
[general]
```

```
portno = 5036
```

```
[RemoteOffice]
```

```
type=friend
```

```
secret=mypass
```

```
callerid="Remote Sales Office" <(250) 123-0000>
```

```
[Joe]
```

```
type=friend
```

```
host = dynamic
```

Programmability/Extensibility

- Generally beyond scope of presentation
- Easiest programmability is through dialplan
- AGI, similar to CGI
 - Commands written to stdout/results from stdin
 - Very rapid development
- C-level API
 - Full access to multiple channels
 - New device drivers, etc.
 - Read the sources LUKE !

Why Asterisk for Telecom?

- *Extreme* cost reduction
- Take control of your phone system
- Rapid and easy development environment
- Rich, broad feature base
- Highly customizable
- Deploy dynamic content via a telephone
- Powerful and flexible dialplan
- **!!!Runs on Linux!!!**

Support Hardware / Protocols

- Zaptel TDM compatible devices from Digium
 - X100P (Single FXO Analog interface, PCI) \$150
 - T100P (Single T1/PRI interface, PCI) \$750
 - E100P (Single E1/PRA interface, PCI) \$750
 - T400P (Quad T1/PRI interface, PCI) \$2500
 - E400P (Quad E1/PRA interface, PCI) \$2500
 - S100U (Single FXS interface, USB) \$150
 - S400P (Quad modular FXS interface, PCI) \$500

Supported Hardware (continued)

- OSS or ALSA compatible full duplex sound cards
- Linux telephony (/dev/phone) interface
 - Quicknet Phonejack, Linejack, PhoneCard,
 - VoIP Blaster (issue's of g723 codec)
- ISDN4Linux supported cards
- CAPI (Common ISDN Application Programming Interface)
- ADSI (Analog Display Services Interface)
Compatible Phones using Zaptel hardware

Hands On Asterisk

- Examples

- Direct IAX Dial, IAXTel VoIP Dial
- Auto Attendant
- Transfer/Parking/Call Forwarding
- VoiceMail
- ADSI Phone Menu System
- Conference Bridge
- Direct Inward System Access (DISA)

More Resources for Asterisk

- Following commands at CLI> prompt:
 - Help
 - show applications
 - show application <foo>
 - show dialplan
- Helpful sites
 - www.asterisk.org
 - www.digium.com
 - [#asterisk](http://irc.freenode.net)
 - www.google.com/custom?sitesearch=www.marko.net

Top Ten Reasons to Run Asterisk

Number 10

Convenient, unambiguous single non-alphanumeric
abbreviation: *

Number 9

Dial-an-MP3

Number 8

Can call you 20 minutes into a blind date as
'emergency exit'

Number 7

It's a lot more 'leet than an answering machine

Number 6

Time to move to something more legal, useful, and entertaining than Cap'n Crunch whistle

Number 5

Only way to build a call center on your laptop

Number 4

Teleconferencing with your friends allows you to be more lazy/unsocial than you already are

Number 3

Build your own phone company from recycled dot-com's

Number 2

Finally tell telemarketers 'All representatives of the household are currently busy assisting other telemarketers. Your call will be answered in the order it was received.'

Number 1

“Why settle for being just another webmaster, hostmaster, or postmaster when you too can be an *astmaster* like me!”

Thank You
Hands On

<http://www.asterisk.org>