
Alphanumeric extensions

Leif Madsen & Olle E. Johansson

Version 1.1 – May 1 2005

The problem

- We have no proper pattern matching for characters or strings
- SIP uri's are UTF 8
- Caller ID names are ISO8859-1
- Asterisk dial patterns are not standardized, but can be defined to be ASCII (we guess)
- IAX2 dial strings are then also ASCII
- *So how do we handle alphanumeric strings?*

A proposal

extensions.conf

```
exten = 1234, 1, app
```

```
exten = _12NXXX, 3, app ; Ast Exp
```

```
alphaexten = jörgen,1,app
```

```
;Reg exp
```

```
alphaexten = _j[öää-z]rgen,1,app
```

```
;In alphaexten "@" is a valid character
```

```
alphaexten = oej@edvina.net,2,app
```

How do you enter from SIP?

```
;Is this a numeric extension?
```

```
alphaexten = _[1-9-.\(\)] ,1, goto(1234,1)
```

```
;It is a numeric extension
```

```
exten = 1234,1,app
```

Dialplan function

`${isalpha <exten>}`

`$EXTEN` should handle both alpha and numeric (tel-url like) extensions

Caller ID needs to implement

- `NUM` exten (Tel-uri/ASCII)
- `NAME` Text name (ISO 8859-1)
- `ALPHANAME` Text name (UTF 8)
- `ALPHANUM` SIP uri (UTF 8)

Dialplan Priorities

1. alphaextensions
2. extensions (numeric only)
3. Pattern matching alphaextensions
4. Pattern matching extensions
5. switches
6. included contexts

Definition

EXTEN Tel-URL like extensions
 383838
 1-800-3983

ALPHAEXTEN UTF8 extension, full SIP uri's

CID_NAME ISO 8859
CID_UTFNAME UTF8

The IAX2 dilemma

- IAX2 accepts text strings
 - but which character set?
- The at sign is reserved for separation of dial string and context – we need to get it back to be able to use a SIP uri
- `chan_iax2` (as well as `chan_sip`) uses a clean-up function to remove `"-."` – the dot and the hyphen are valid characters in a SIP uri
- Let's define the current IAX2 dialstring as numeric (tel-uri like) and add an IE flag for sending an alphaexten

CDR records?

- What do we store in CDR records?
- Do we need to add an alphaexten field so that we do not crash current CDR billing systems?
- The destination in CDR is currently "Asterisk extension", which means ASCII
- The Caller ID names is ISO 8859-1
- We need to have full SIP uri's in both from and to versions, with alphaexten (uri) and full caller ID name

Alphahints

- Do we need to implement an numeric hint from an alpha to rescue some applications?
- alphaexten = numhint, 123-2444-44

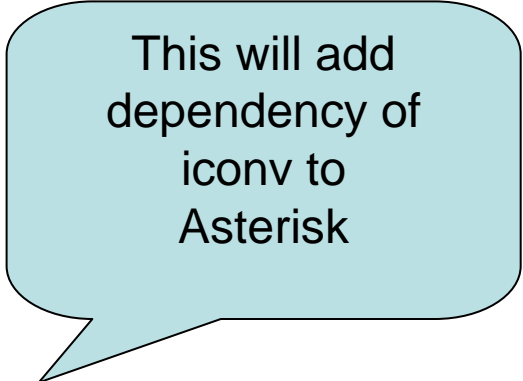
Do not think so, can't find a reason. We will enhance applications to support the addition of alphaextensions, but they will continue to work as today.

Application revision

- Will any applications, modules or functions be able to handle this
- a char[] string will still work, so most apps should be ok
...
- **Consider**
 - We are not changing the exten, just removing extra characters from the destination (everything not included in tel uri's)
 - We are not changing caller ID name, just enabling a new "full" Caller ID name in utf8
 - Dialplans will continue to work, enhanced dial plans will work better for those using utf8 and name/uri based dialing

Conversions needed

- SIP caller ID (utf 8) to cid_name (ISO8859-1)
- cid_name to SIP caller ID (utf 8)
- Conversion when logging to console/file/syslog
 - Should console log be ISO8859-1 ?
 - Syslog character set?



This will add
dependency of
iconv to
Asterisk

Alphaextensions over IAX2

One way to do this:

- URL-encode the dialstring from Windows UTF 8 into ASCII 7 bit
- ..or BASE64 encode
- If IAX2 dialstring begins with ---- it should be parsed as encoded alphaexten
- Encoded @ is part of alphaexten, unencoded @ is separator of dialstring and context in remote host

Another way is to add an IE and send alphaextensions with a flag indicating alphaexten

Functions to look at

- LookupCIDname
 - We will need LookupCIDnameutf8
- SetCalleridname
 - Need SetCallerIDnameutf8
- `${CALLERIDNAME}` variable
 - `${CALLERIDNAMEU}` needed
- `$(<string> <operator> <string>)` expressions
 - Do we need to create new operators that operate on UTF8?

Dialplan addition

```
exten = 1,100,noop
```

```
exten = (prev),n,dial(SIP/jörgen)
```

(prev) indicates copy previous extension.

This is really handy with complex regular expressions, like

```
alphaexten => _^[1-9\.a-z]+@[a-z0-9-]+\[a-z]+,n,noop()
```

```
alphaexten => (prev),n,dial(SIP/someone)
```

Thank you for listening

- Are we ready to start coding ?