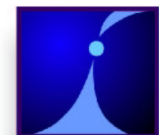


UniConf, The One True Configuration System

(Yes, Even Truer than GConf)

Double Speed Version!

Avery Pennarun <apenwarr@nit.ca>



Outline

- What is UniConf
- Your Config System Sucks
- UniConf Doesn't Suck as Much
- Wacky UniConf Tricks
- Questions



What is UniConf

- From our press release:
 - *UniConf, our advanced object-relational web-enabled remote management console with cross-platform transaction-oriented XML content delivery, will totally change the way people do business and revolutionize the world of Open Source.*
- Or more concisely:
 - UniConf is a thing for storing stuff.
 - Middleware



Your Config System Sucks

- (...no matter which config system you use)
- Ugly file formats
- Bad API
- Can't use Gopher for data transfer
- No notifications ("pull" not "push")
- No defaults/overrides
- Not ubiquitous



Bad File Format: XF86Config

```
Section "Screen"  
    Identifier          "Screen0"  
    Device              "Card2"  
    Monitor             "Monitor0"  
    DefaultColorDepth  16  
    SubSection          "Display"  
        Modes          "1024x768" "800x600"  
        Depth          16  
    EndSubSection  
EndSection
```



Bad API: The Windows Registry

```
// you probably only do this once  
CRegKey registry;  
registry.Create(HKEY_CURRENT_USER,  
    "Software\\Net Integration\\ExchangeIt");  
  
// but you do this every time  
TCHAR buf[1024] = {0};  
ULONG size = sizeof(buf) / sizeof(buf[0]);  
registry.QueryStringValue(  
    _T("Host"), buf, &size);  
if (buf[0] == 0)  
    strcpy(buf, defaultHost);
```



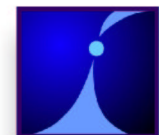
Good API: UniConf

```
// do this once  
UniConfRoot root("registry:localhost");  
UniConf cfg(  
    root["HKEY_CURRENT_USER/Software/Net  
    Integration/ExchangeIt"]);  
  
// but only this each time  
string buf = cfg["Host"].get();
```



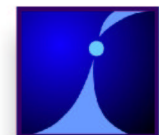
Strings, Strings and More Strings

- Some people think strings aren't enough:
 - Too inefficient, too inflexible, too inconvenient?
- Computers are fast! Get over it!
- You can represent *anything* as a string:
- Admit it, you do it all the time in your head!
 - strings: "I'm a string!"
 - numbers: 5, 5.15123e+4
 - boolean: yes
 - ...and yes, even lists...



Digression: tcl-style String Lists

- Mostly I don't have much use for TCL...
- ...but they got this part right
- A simple list:
 - apples carrots bacon
- A list with multi-word entries:
 - apples {orange carrots} {more bacon}
- A nested list:
 - {6 5.5 3} {1 0 1} {apples carrots bacon}



Digression: tcl-style .ini Files

- In fact, you can represent entire files as tcl-style lists...
- ...therefore I do!

```
[Section]
entry1 = value1
sub/entry2 = value2
sub/entry3 = {
    a multi-line
    value!
}
```



Digression: tcl-style Amazingly Super-Enhanced .ini Files

- And this is why

```
[ {Stupid Section Name  
Contains Newlines} ]  
entry1 = value1\nwith newlines  
{entry2  
can also be weird} = {value2 and  
have a funny value}  
entry3 + {  
    sub/1 = and how about  
    sub/2 = those subkeys?  
}
```

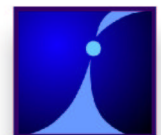


End of Digression...

- Anyway, the point was that you can encode anything as a string, so you might as well.
- Now, on with the other features...

Notifications: You Want Them

- If you use gconf, you already know this
- Config-centric programming
- "Push" vs. "Pull"
- What happens when someone edits your config file and doesn't tell you?



Frontends and Backends

- UniConf is the One True Configuration System
- So we need to transparently talk to everyone else
- And this is where it gets fun.

UniConf Frontend: API people use to talk to us

UniConf Backend: the API UniConf uses to talk to others

- There are several of each!



Some Standard Backends

- Generators:
 - temp / null
 - ini / initree (can handle KDE config files)
 - gconf
 - unix / tcp / ssl / wvstream
- Wrappers:
 - cache / persist
 - readonly / default / list



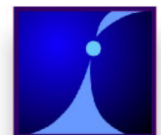
The UniConfDaemon is Optional

- Some people really want this
- "Not another stupid config daemon!"
- The UniConfClient is just another backend!
- But the daemon does make notifications cheaper - better than polling!
- With the "cache" and "persistent" wrappers, it's great for distributed configuration



Defaults and Overrides

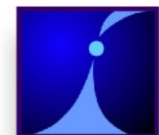
- Given a list of config files:
 - /etc/overrides.ini
 - ~/.mysettings.ini
 - /etc/defaults.ini
- overrides.ini always wins if it contains a key
- defaults.ini has keys just in case nobody else does
- Now imagine what would happen if those weren't just files...



Monikers are Fun

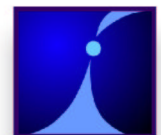
- Monikers: strings that produce objects
- Construct and combine 10 generators using only one line of code:

```
UniConfRoot cfg("cache:list:  
  readonly:ini:/etc/overrides.conf  
  ini:/home/apenwarr/.myapp.conf  
  default:ini:/etc/defaults.conf  
  unix:/tmp/uniconfdaemon.sock  
  tcp:server.local.com:4111  
  ssl:x.remoteclassroom.com:4112");
```



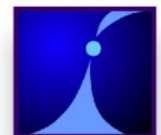
How About Those Frontends?

- C/Perl/PHP/etc bindings
- Gnome GConf
- KDE KConfig
- Command Line (un i)
- PAM/NSS
- Filesystem Frontend



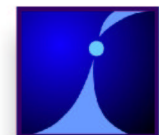
Filesystem Frontend - WvFuse

- WvFuse (based on FUSE) lets you write filesystems in userspace
- So we did
- But not the UniConf one yet :)
- Isn't it convenient that UniConf keys follow the same naming rules as Unix filenames?



Fun Things to Do With UniConf

- Use UniKonf (KDE) for configuring Gnome
- Use GConf for configuring KDE
- Access Control backends
- LDAP backend
- Cache it all to make it faster
- Email/vCard/vCal backend

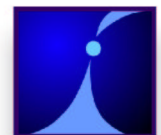


Demo



Summary

- Lots of backends
- Lots of frontends
- If we don't compete with anyone, we can't lose!



Credits

Original design - me

Total redesign so it works - Jeff Brown

More redesign so notifications work - Dr. Held

Permissions system - Joe Mason

Lots more cheesy front/backends - me

Avery Pennarun <apenwarr@nit.ca>

<http://open.nit.ca/wiki/?UniConf>



Any Questions?

Original design - me

Total redesign so it works - Jeff Brown

More redesign so notifications work - Dr. Held

Permissions system - Joe Mason

Lots more cheesy front/backends - me

Avery Pennarun <apenwarr@nit.ca>

<http://open.nit.ca/wiki/?UniConf>

