

Linux Registry

*Turning Linux into a Trully Integrated
Operating Environment*

Avi Alkalay <avi@unix.sh>
Linux, Open Standards Consultant

<http://registry.sf.net>



Before We Start . . .

**FORGET ABOUT
THE NAME**

<http://registry.sf.net>



Future, Past, Present

Where the Desktop is going ?

Where the Underlying System is Going ?

Can we keep evolving the
Desktop without evolving the
Underlying OS structure ?

<http://registry.sf.net>



Why Linux/UNIX Needs a Registry

```
Section "InputDevice"  
    Identifier "Mouse0"  
    Driver "mouse"  
    Option "Protocol" "IMPS/2"  
    Option "Device" "/dev/input/mice"  
    Option "ZAxisMapping" "4 5"  
    Option "Emulate3Buttons" "yes"  
EndSection
```

In Fedora 2:

/etc/X11/xorg.conf

Other Distros:

God knows where...

- ◆ Difficult to package software that automatically integrates with other installed software, anywhere
- ◆ Selfish configuration files format
- ◆ Different location from distro to distro

<http://registry.sf.net>



What is the Linux Registry Project

- ✓ **Task force to build an integration ecosystem**
- ✓ An **universal key-value pair namespace** to store base system and user configuration atoms
- ✓ Clean **API** to access Keys database
- ✓ Manageable by simple system tools like **vi**, **cp**, **grep**, etc. Plain text storage
- ✓ **No dependencies**, Unicode ready, secure, lightweight, POSIX compliant (highly portable)
- ✓ Available anywhere, anytime, even to early boot stage programs like [/sbin/init](#)

<http://registry.sf.net>



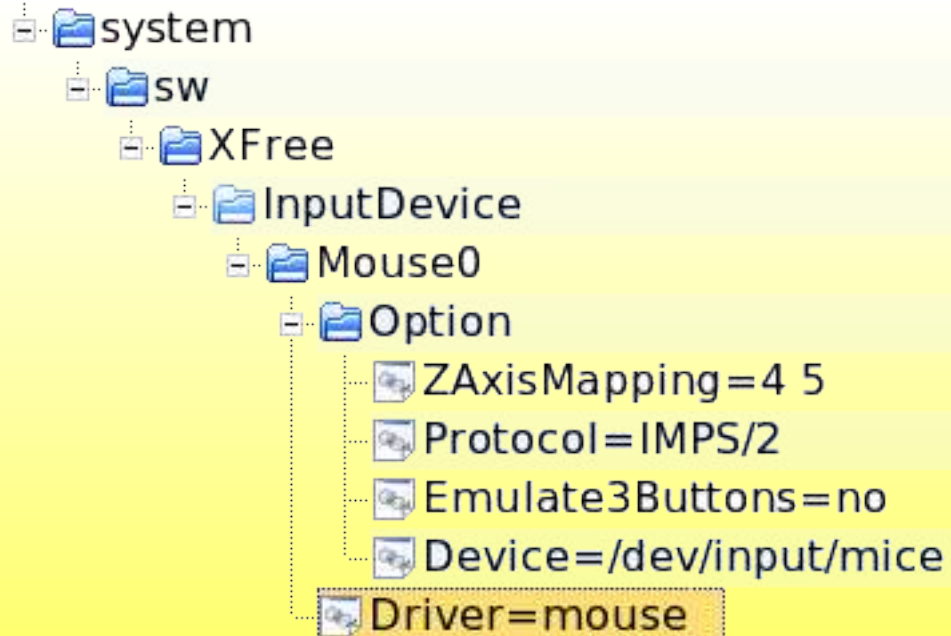
What Linux Registry is Not

- ✗ Is NOT Webmin, Linuxconf or YaST-like software
- ✗ Is NOT client-server software
- ✗ Is NOT an OS service that can become **unavailable**
- ✗ Does NOT access SQL databases
- ✗ Is NOT an alternative to LDAP or NIS

<http://registry.sf.net>



How About This ?



```
Section "InputDevice"
    Identifier "Mouse0"
    Driver "mouse"
    Option "Protocol" "IMPS/2"
    Option "Device" "/dev/input/mice"
    Option "ZAxisMapping" "4 5"
    Option "Emulate3Buttons" "yes"
EndSection
```

✓ Each information bit is easily accessible

✓ Understandable and editable by any program (without having to implement a config file compiler)

✓ Ready for admin GUIs

✗ Need human eyes and brains

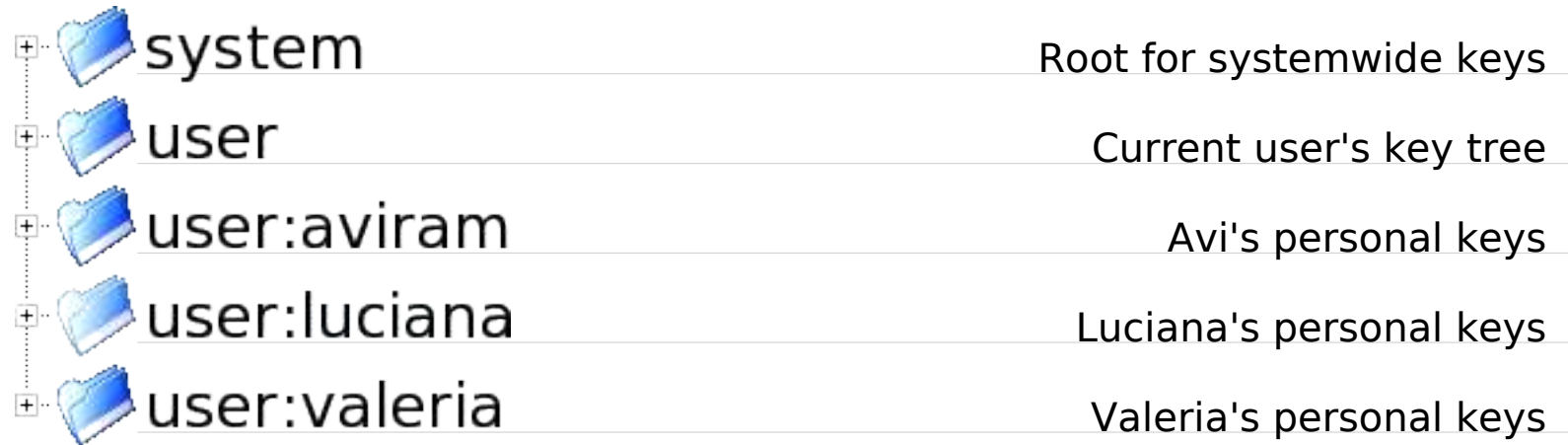
✗ Need a manual

✗ ... and vi

<http://registry.sf.net>













Key Hierarchy



- ◆ The [system/*](#) tree is stored under [/etc/registry/](#)
- ◆ The [user:\\$USER/*](#) tree is stored under [~\\$USER/.registry/](#)
- ◆ The [user/*](#) tree is a shortcut to current user's tree



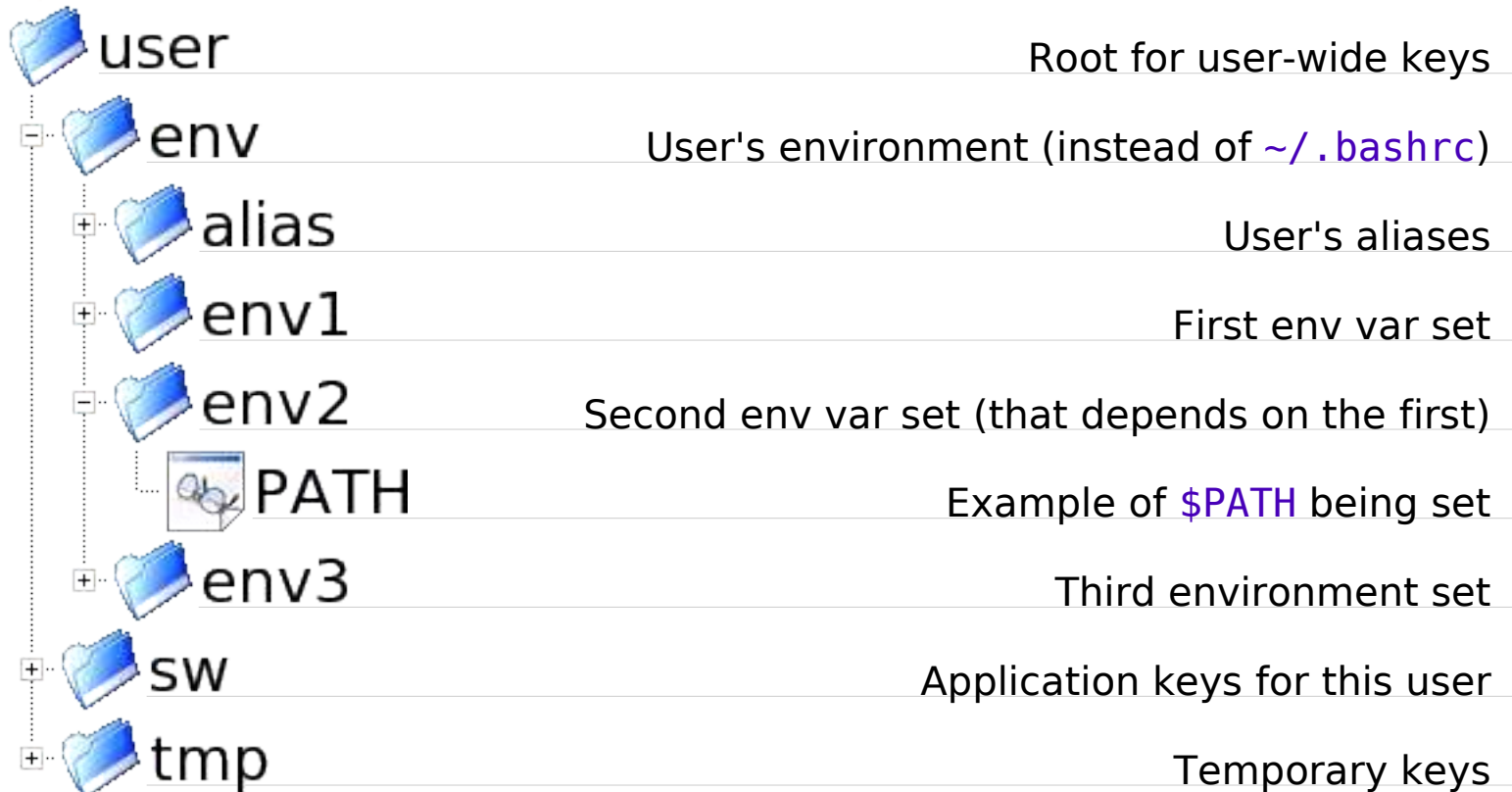
Key Hierarchy :: *system/...*

 system	Root for systemwide keys
 filesystems	Equivalent to /etc/fstab
 groups	Equivalent to /etc/group
 hw	Keys with detected HW
 init	Equivalent to /etc/inittab
 network	Network info, interface IPs, etc
 SW	Root for application global keys
 regedit	Application 1
 XFree	Application 2
 users	Equivalent to /etc/passwd

<http://registry.sf.net>



Key Hierarchy :: *user/...*



Key Names

<u>Key Name Example</u>	<u>Value Example</u>
system/filesystems/boot/device	/dev/hda1
system/net/resolver/server	192.168.10.1
system/net/ISP/AOL/phone	555-2132
system/net/ISP/.MSN/login	user@msn.com
system/sw/XFree/Device/Videocard0/Driver	radeon
system/sw/httpd/ServerName	/var/www/site1.com
user/env/alias	ls -Fh
user:valeria/sw/regedit/gui/width	747

***This little dot makes
inactive the entire
"MSN/*" subtree***



Key Properties



system/users/root/shadowPassword = \$1\$yM93nU

user:valeria/env/env2/PATH = \$PATH:/usr/sbin

- *Name*
- *Value*
- *Value type*
- *User domain or owner*
- *Description/Comment*
- *System UID & GID*
- *Access Permissions*
- *Active or inactive key*
- *Last change time*
- *Last access time*

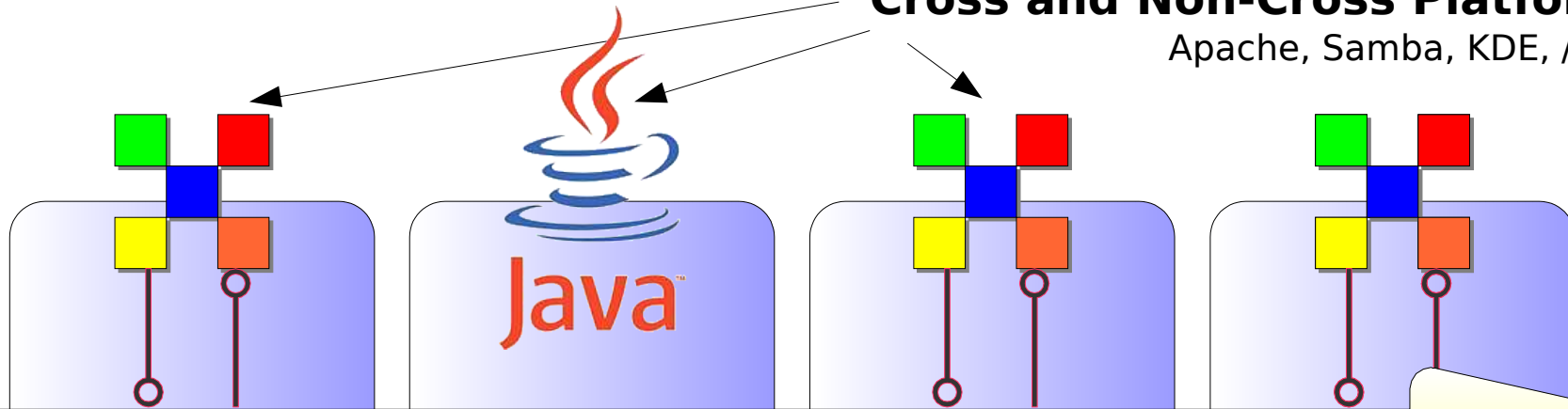
<http://registry.sf.net>



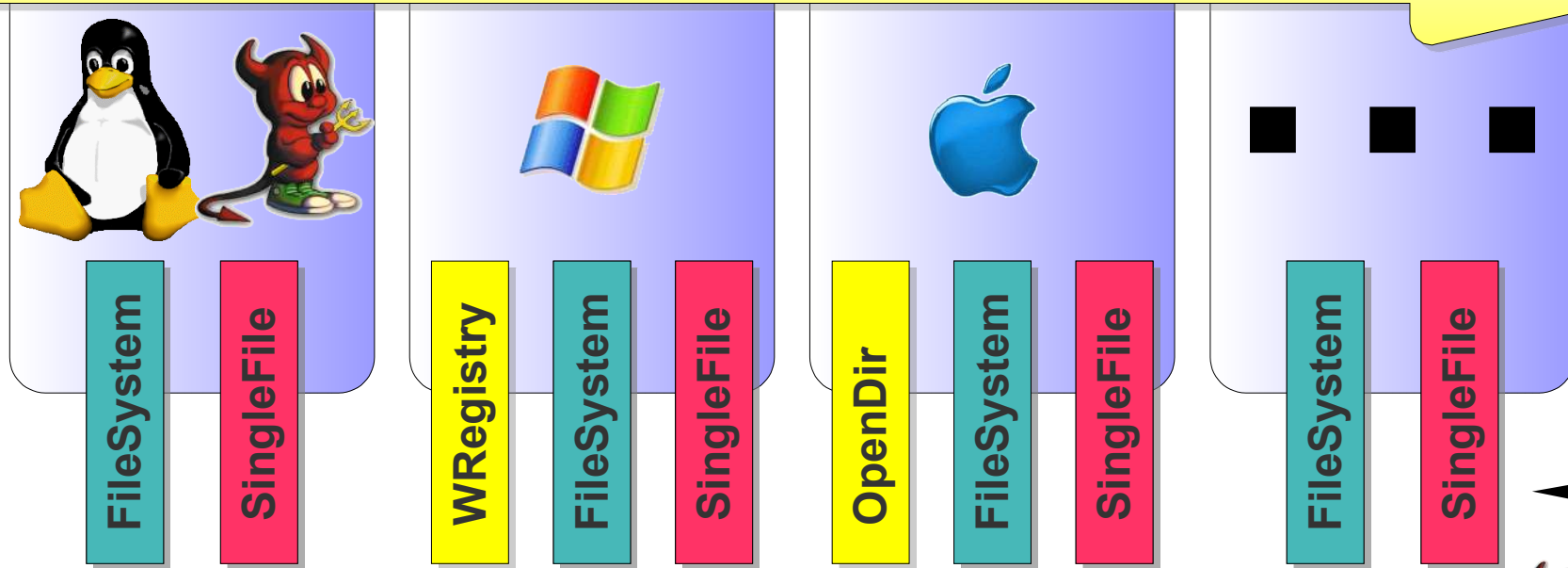
Configuration System Utopia

Cross and Non-Cross Platform Software

Apache, Samba, KDE, /sbin/init, inittools



Registry API & Namespace :: Abstraction Layer



<http://registry.sf.net>



While GConf is Fat and Dependent...

```
bash$ ldd /usr/bin/gconfd-1
libgconf-1.so.1 => /usr/lib/libgconf-1.so.1
liboaf.so.0 => /usr/lib/liboaf.so.0
libORBitCosNaming.so.0 => /usr/lib/libORBitCosNaming.so.0
libORBit.so.0 => /usr/lib/libORBit.so.0
libIIOP.so.0 => /usr/lib/libIIOP.so.0
libORBitutil.so.0 => /usr/lib/libORBitutil.so.0
libm.so.6 => /lib/tls/libm.so.6
libgmodule-1.2.so.0 => /usr/lib/libgmodule-1.2.so.0
libglib-1.2.so.0 => /usr/lib/libglib-1.2.so.0
libdl.so.2 => /lib/libdl.so.2
libc.so.6 => /lib/tls/libc.so.6
libpopt.so.0 => /usr/lib/libpopt.so.0
libwrap.so.0 => /usr/lib/libwrap.so.0
/lib/ld-linux.so.2 => /lib/ld-linux.so.2
libnsl.so.1 => /lib/libnsl.so.1
```

<http://registry.sf.net>



Registry is Light . . .



```
bash$ ldd /lib/libregistry.so
libc.so.6 => /lib/tls/libc.so.6
/lib/ld-linux.so.2 => /lib/ld-linux.so.2
```

- ✓ *Ready to be used in a restrictive environment (no net, no OS services), even by /sbin/init*
- ✓ *No **daemon**, so*
 - ◆ *No single point of failure*
 - ◆ *One process can't harm another's affairs with the key database*
 - ◆ *No comm protocol, so no extra fat here*
 - ◆ *No security holes, since all security is handled directly by the OS*
 - ◆ *Clean and simple*

<http://registry.sf.net>



Key Database Administration

<http://registry.sf.net>



The rg Command: Perfect for Scripts

Full manual pages available !

```
bash# rg set system/sw/XFree/Screen/Display/Modes 1280x1024
bash$ rg get system/filesystems/boot/mpoint

bash$ rg export system/sw/myapp > myapp.conf.xml
bash# rg import < myapp.conf.xml

bash$ rg edit -R user/sw/myapp

bash$ rg ls -Rv system/sw/myapp

bash$ rg monitor system/init/id/runlevel

bash$ rg rm user/env/alias/vnc

bash$ rg ln system/sw/myapp/key1 user/sw/myapp/key1
```

<http://registry.sf.net>



XML Import and Export

```
bash$ rg export user/env/alias > file.xml
bash$ rg import < file.xml
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE keyset PUBLIC "-//Avi Alkalay//DTD Registry 0.1.0//EN"
"http://registry.sf.net/dtd/registry.dtd">

<keyset>

  <key name="user:aviram/env/alias/ls"
    type="string" uid="aviram" gid="aviram" mode="0664">

    <value>ls -Fh --color=tty</value>
  </key>

  <key name="user:aviram/env/alias/vnc"
    type="string" uid="aviram" gid="aviram" mode="0664">

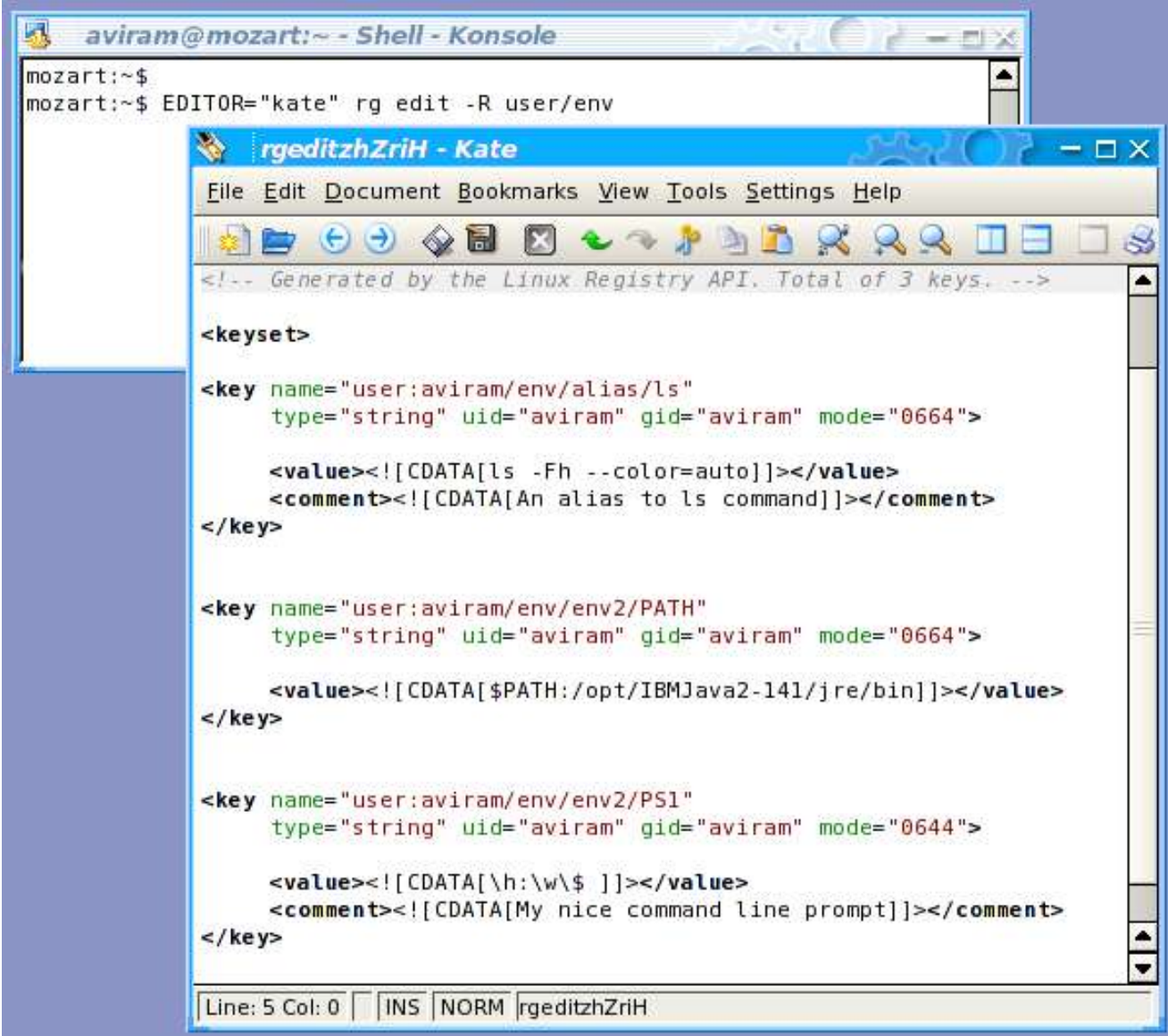
    <value>vncserver -geometry 900x650</value>
  </key>

</keyset>
```

<http://registry.sf.net>



Using rg to Edit in Batch



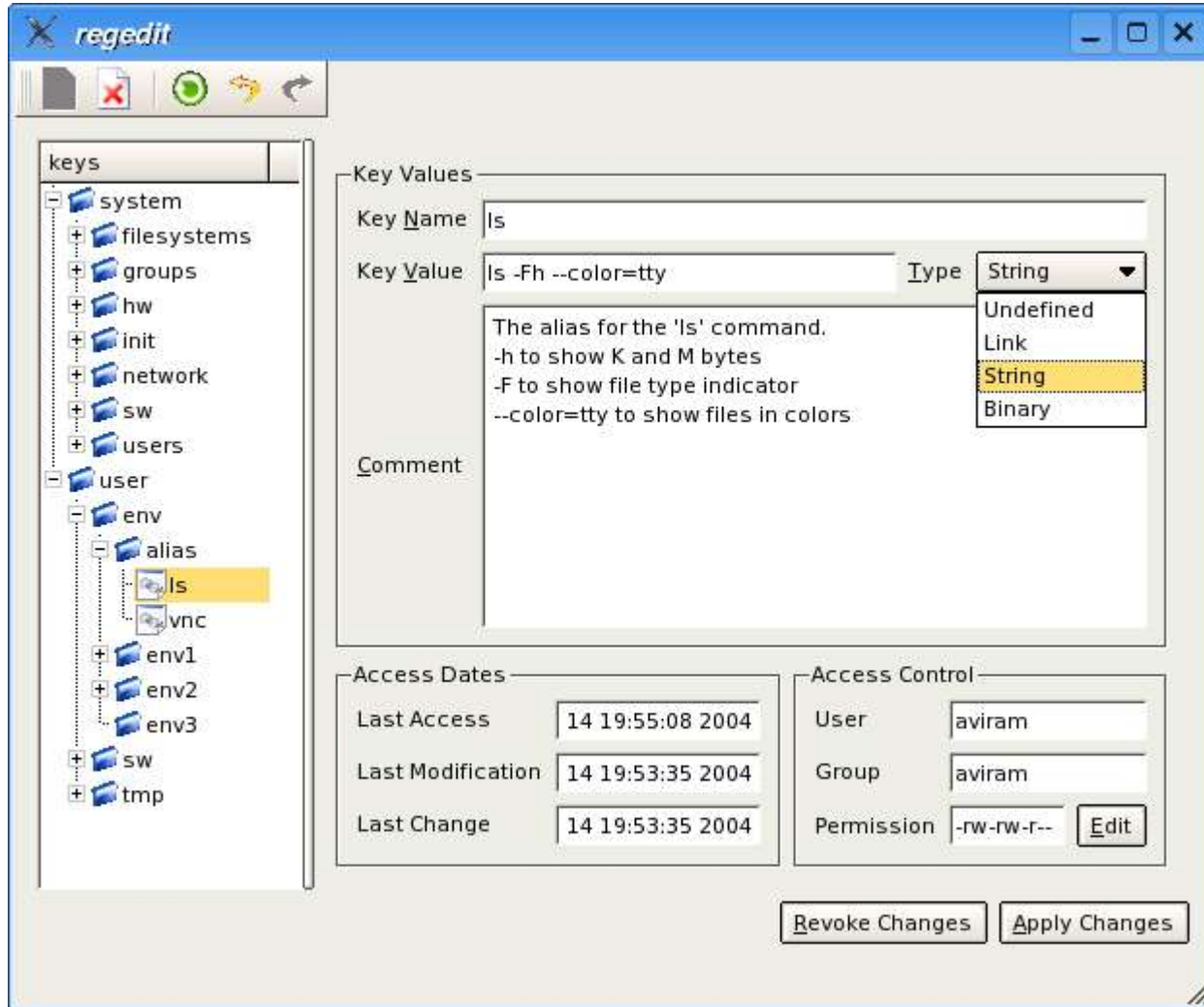
```
aviram@mozart:~ - Shell - Konsole
mozart:~$
mozart:~$ EDITOR="kate" rg edit -R user/env

rgeditzhZriH - Kate
File Edit Document Bookmarks View Tools Settings Help
<!-- Generated by the Linux Registry API. Total of 3 keys. -->
<keyset>
<key name="user:aviram/env/alias/ls"
  type="string" uid="aviram" gid="aviram" mode="0664">
  <value><![CDATA[ls -Fh --color=auto]]></value>
  <comment><![CDATA[An alias to ls command]]></comment>
</key>
<key name="user:aviram/env/env2/PATH"
  type="string" uid="aviram" gid="aviram" mode="0664">
  <value><![CDATA[$PATH:/opt/IBMJava2-141/jre/bin]]></value>
</key>
<key name="user:aviram/env/env2/PS1"
  type="string" uid="aviram" gid="aviram" mode="0644">
  <value><![CDATA[\h:\w\$ ]]></value>
  <comment><![CDATA[My nice command line prompt]]></comment>
</key>
Line: 5 Col: 0 | INS | NORM | rgeditzhZriH
```

- Use any editor to edit in XML
- Keys you **add** will be added
- Keys you **change** will be changed
- Keys you **delete** will be deleted
- (thanks to the ***ksCompare()*** API method)



Regedit: The GUI Registry Edit Tool



- More friendly than the **rg** command
- Hierarchical view
- Supports UNDOs and REDOs

*If everything else fails, you can still directly edit the key database with **vi**, because it is all plain text !*

<http://registry.sf.net>



The API

<http://registry.sf.net>



Ready to Be Used Today !!!

➤ **Develop today**

- ✓ C/C++
- ✓ Shell
- ✓ XML
- ✓ Python
- ✓ Ruby
- ✓ Java

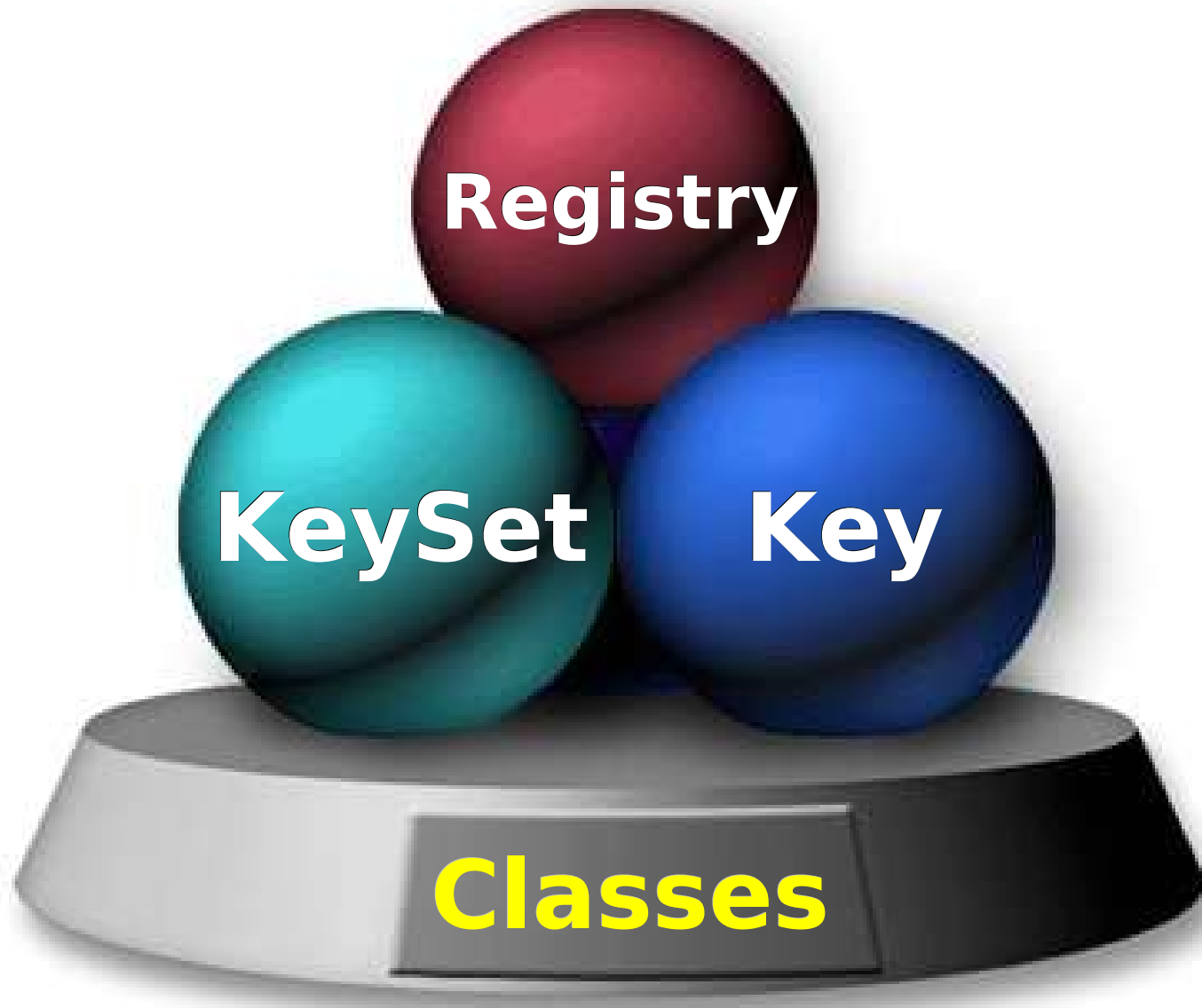
➤ **To come . . .**

- ✓ Perl
- ✓ PHP
- ✓ Others . . .
- ✓ (any contribution for a language binding ?)

<http://registry.sf.net>



Only 3 Classes



<http://registry.sf.net>



Things You Can Do With the Classes

➤ *Registry class*

- ✓ Retrieve and commit **Keys** and **KeySets**, recursively or not
- ✓ Retrieve and commit individual **Keys** value, by absolute name or relative to parent
- ✓ Monitor and notify changes in **Keys** and **KeySets**
- ✓ Create and delete **regular**, **folder** or **symbolic link** **Keys**

<http://registry.sf.net>



Classes continued...

➤ *Key class*

- ✓ Get and Set key properties like **name**, root and base name, **value**, **type**, **permissions**, changed time, **description**, etc
- ✓ Compare all properties with other keys
- ✓ Test if changed, if is a user/ or system/ key, etc
- ✓ Flag it and test if key has a flag
- ✓ Export to **XML**

➤ *KeySet class*

- ✓ Linked list of **Key** objects
- ✓ Insert and append entire **KeySets** or **Keys**
- ✓ Work with its internal cursor
- ✓ Compare entire **KeySets**
- ✓ Export to **XML**

<http://registry.sf.net>



C API Methods

Class Registry

```
registryOpen()  
registryClose()  
registryGetValue()  
registryGetKeyByParent()  
registryGetKeyByParentKey()  
registryGetValueByParent()  
registrySetValue()  
registrySetValueByParent()  
registryRemove()  
registryLink()  
registryGetKeyByParent()  
registryGetKeyByParentKey()  
registryGetValueByParent()  
registryGetComment()  
registrySetComment()  
registryStatKey()  
registryGetKey()  
registrySetKey()  
registryGetChildKeys()  
registryGetRootKeys()  
registrySetKeys()  
registryMonitorKey()  
registryMonitorKeys()
```

Class Key

```
keyInit()  
keyClose()  
keyIsInitialized()  
keyNeedsSync()  
keyIsSystem()  
keyIsUser()  
keyGetNameSpace()  
keyIsDir()  
keyIsLink()  
keyToStream()  
keyDup()  
keyGetType()  
keySetType()  
keyGetNameSize()  
keyGetFullNameSize()  
keyGetName()  
keySetName()  
keyGetFullName()  
keyGetFullRootName()  
keyGetFullRootNameSize()  
keyGetBaseName()  
keyGetBaseNameSize()  
keyGetRootName()  
keyGetRootNameSize()  
keyCompare()  
keyGetUID()  
keySetUID()  
keyGetGID()  
keySetGID()  
keyGetCommentSize()  
keyGetComment()  
keySetComment()  
keyGetAccess()  
keySetAccess()  
keyGetOwner()  
keySetOwner()  
keyGetDataSize()  
keyGetString()  
keySetString()  
keyGetBinary()  
keySetBinary()  
keyGetLink()  
keySetLink()  
keyGetMTime()  
keyGetATime()  
keyGetCTime()  
keyGetParentName()  
keySetFlag()  
keyGetFlag()
```

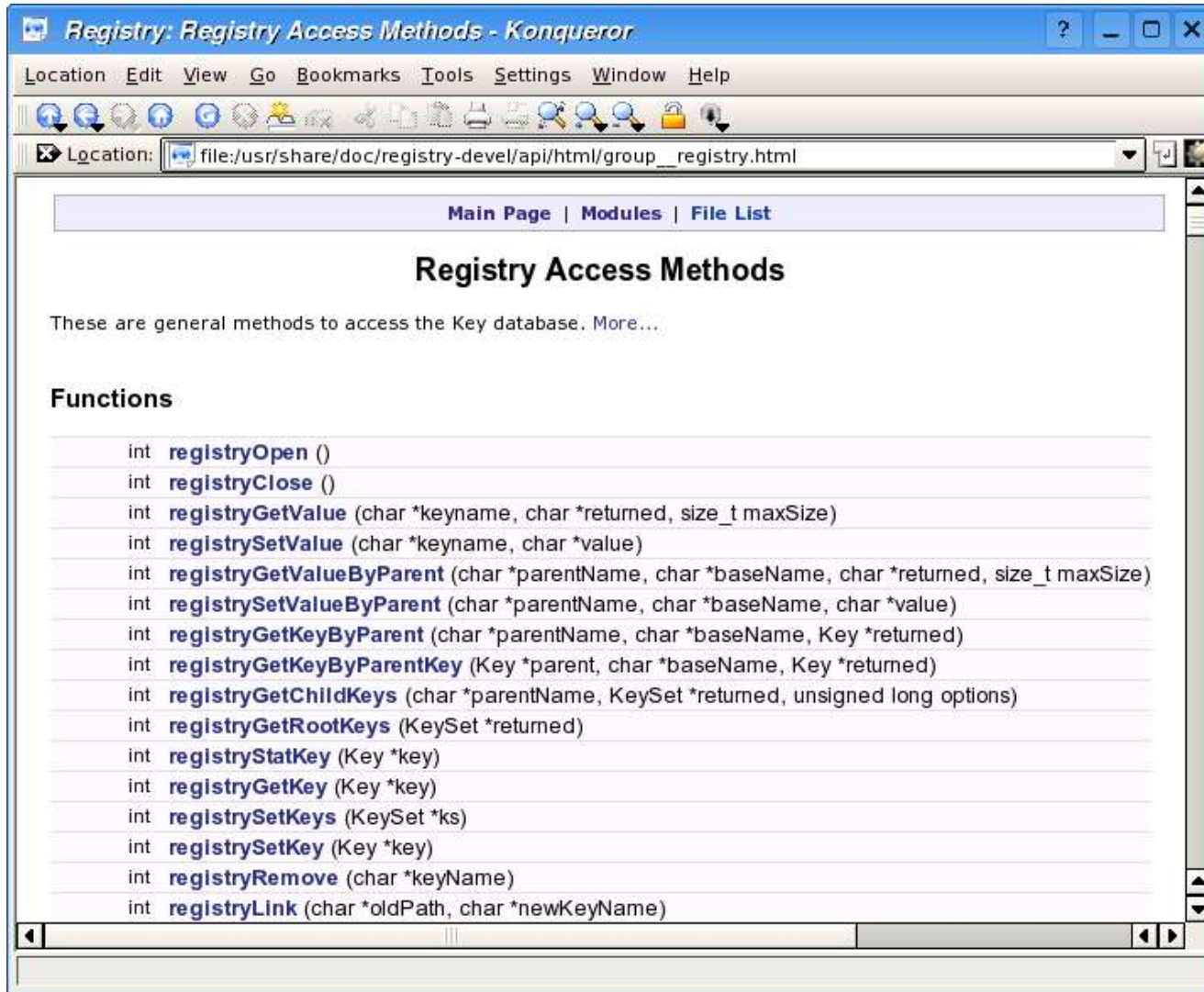
Class KeySet

```
ksInit()  
ksClose()  
ksInsert()  
ksAppend()  
ksInsertKeys()  
ksAppendKeys()  
ksToStream()  
ksCompare()  
ksNext()  
ksRewind()
```



The API is Fully Documented !!

<http://registry.sf.net/#man>



- ✓ API doc in browsable Doxygen format
- ✓ API man pages
- ✓ Many code examples

<http://registry.sf.net>



Current Ecosystem

- **Regedit GUI, to Key database editing**
- **NSSwitch module**
 - Users database and authentication from the Registry
- **All those language bindings**
- **Glibc patches**
 - fstab
 - resolver
 - etc

<http://registry.sf.net>



Demo

<http://registry.sf.net>



The Linux Registry Project needs YOU !

Talk about it

Port It

***Contribute with patches that
registry-enable old software***

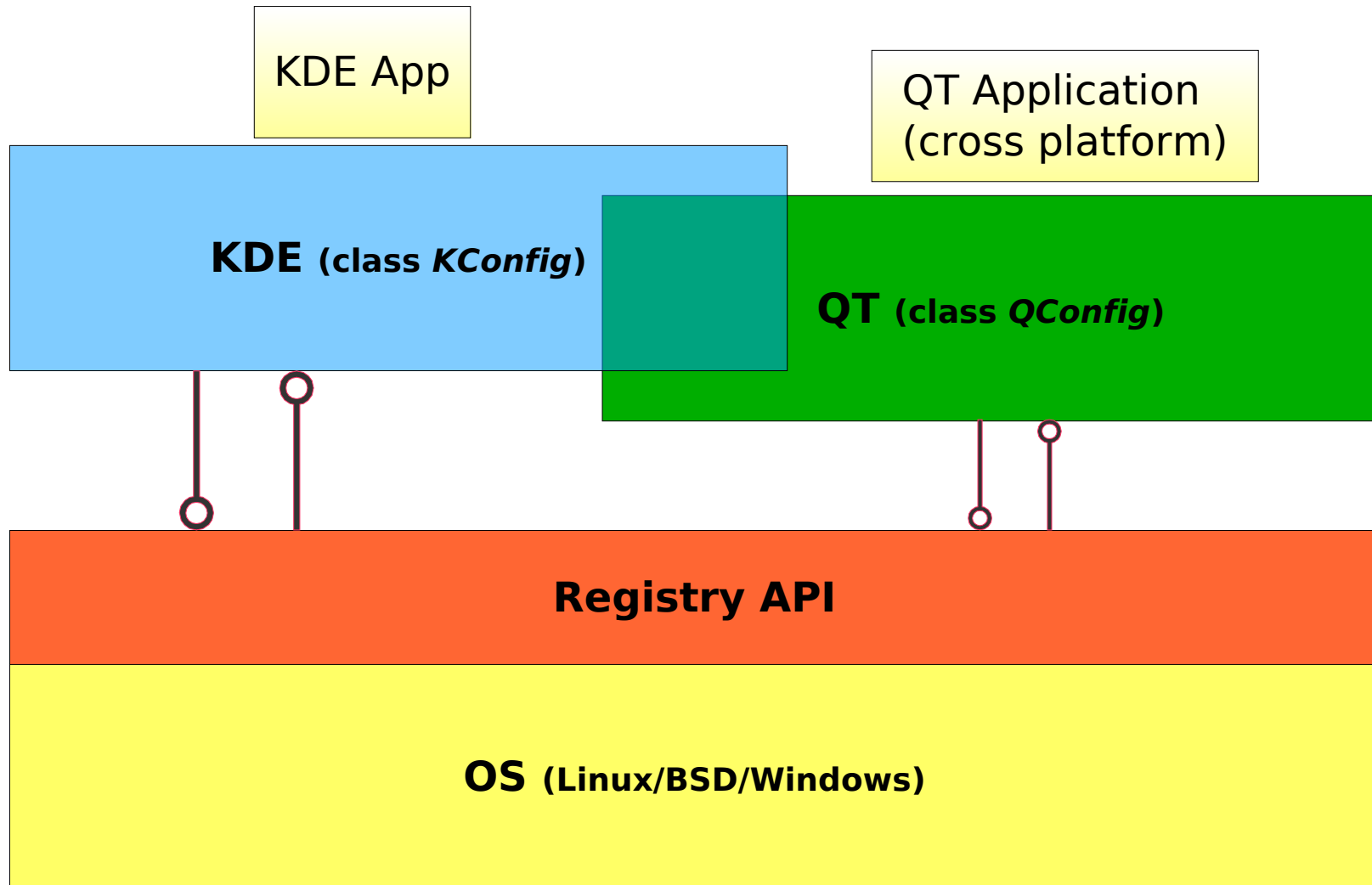
Use it in your own software

It's free. It's for you.

<http://registry.sf.net>



KDE, QT and Registry



<http://registry.sf.net>



Thank You

➤ IBM

- Support for a personal Open Source project to make Linux better
- Travel expenses, so I can show it to you, here, now
- Hours in airports and flights, so I have time to code :-)

➤ **Linux Registry Project community**

- Maintaining Subversion repository, Wiki space, etc
- Kickoff for the Doxygen documentation
- Patches, usefull software, language bindings, etc

<http://registry.sf.net>



Linux Registry

*Turning Linux into a Trully Integrated
Operating Environment*

Avi Alkalay <avi@unix.sh>
Linux, Open Standards Consultant

<http://registry.sf.net>





IBM

Backup

<http://registry.sf.net>

