

Introduction to FreeNAS development

John Hixson

john@ixsystems.com
iXsystems, Inc.



A bit about me

- BSD geek that does geeky BSD things
- Maintains jail, plugin, and directory service systems on FreeNAS
- Occasional committer to PC-BSD
- Hacks on FreeBSD once in a blue moon
- All my time goes into FreeNAS!



A bit about iXsystems

- Corporate sponsor of PC-BSD
- Corporate sponsor of FreeNAS
- Employ several FreeBSD committers
- Sponsor several open source conferences every year
- Contribute heavily to FreeBSD



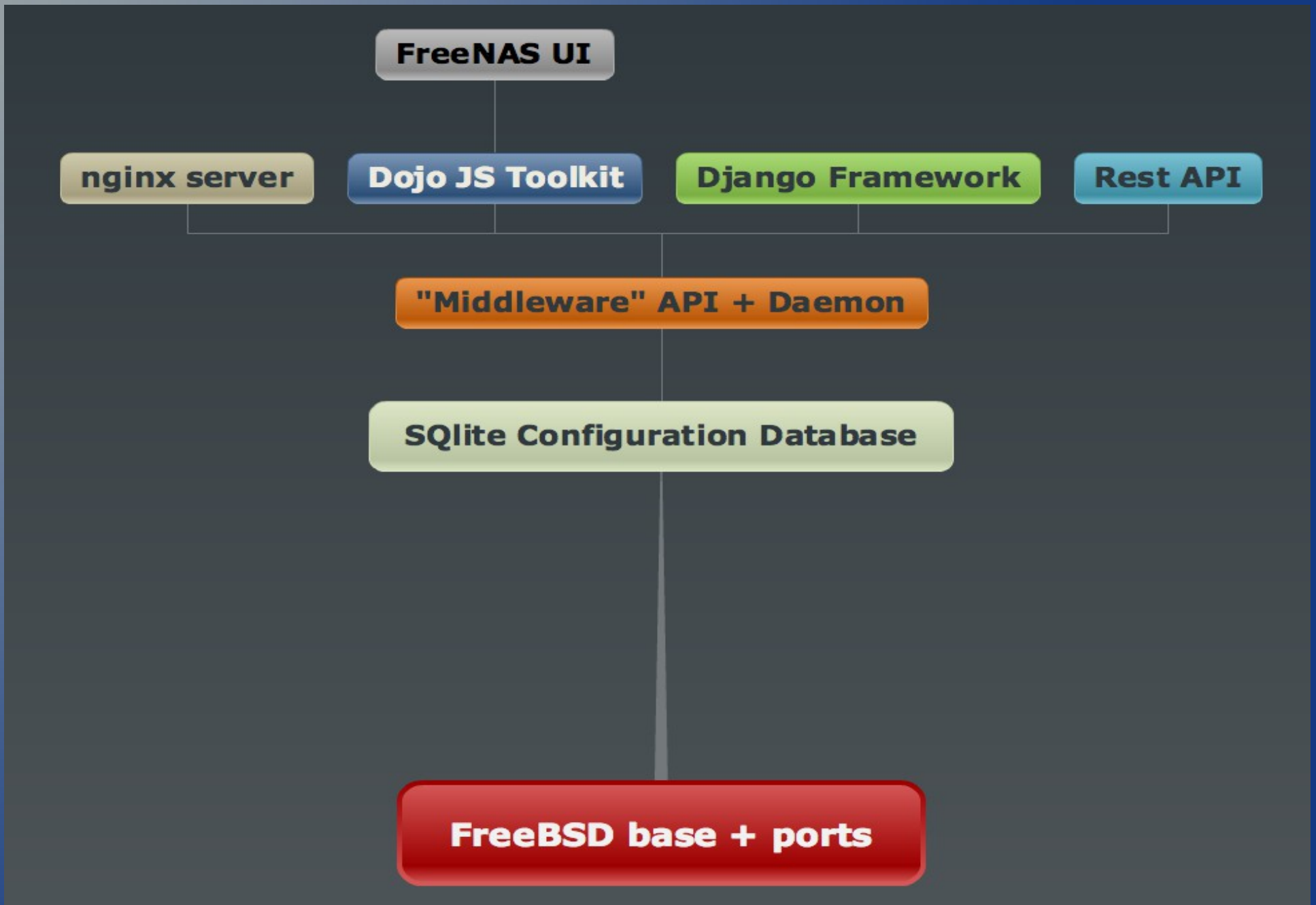
What this talk will cover

- What FreeNAS is
- Adding a feature to the UI
- Adding software to the operating system
- How to contribute

What FreeNAS is:

- 100% free and open source network attached storage operating system
- Built on FreeBSD and ZFS
- CIFS, NFS, AFP, iSCSI, etc
- Over 6 million downloads to date





A quick look at FreeNAS



Adding a feature to the UI

- Create feature model, form and view
- Add service entry for the feature
- Make the middleware aware of feature
- Add a startup / shutdown script for the feature
- Javascript to toggle feature on and off

PONIES service

Portable Object New Interactive Engine Service



```
${src}/gui/services/models.py
```

```
class Ponies(Model):  
    class Meta:  
        verbose_name = _(u"Ponies")  
        verbose_name_plural = _(u"Ponies")  
  
    class FreeAdmin:  
        deletable = False
```

```
${src}/services/forms.py
```

```
class PoniesForm(ModelForm):  
    class Meta:  
        fields = '__all__'  
        model = models.Ponies
```

`${src}/services/views.py`

```
def core(request):  
    ....  
    return render(request, 'services/core.html', {  
        ...  
        'ponies': ponies,  
    })
```

```
def servicesToggleView(request, formname):  
    form2namemap = {  
        ....  
        'ponies_toggle': 'ponies',  
    }  
    ....
```

Adding a feature to the UI

- Create feature model, form and view
- Add service entry for the feature
- Make the middleware aware of feature
- Add a startup / shutdown script for the feature
- Javascript to toggle feature on and off

```
src/gui/services/migrations/0104_auto__add_ponies.py
```

```
class Migration(SchemaMigration):  
    def forwards(self, orm):  
        ....  
        p = orm.Ponies()  
        p.save()  
        s = orm.Services()  
        s.srv_service = 'ponies'  
        s.srv_enable = False  
        s.save()
```

Adding a feature to the UI

- Create feature model, form and view
- Add service entry for the feature
- Make the middleware aware of feature
- Add a startup / shutdown script for the feature
- Javascript to toggle feature on and off

`${src}/middleware/notifier.py`

```
class notifier:
```

```
....
```

```
def _started_ponies(self):  
    self._system("/usr/sbin/service ponies status")
```

```
def _start_ponies(self):  
    self._system("/usr/sbin/service ponies start")
```

```
def _stop_ponies(self):  
    self._system("/usr/sbin/service ponies stop")
```

```
def _restart_ponies(self):  
    self._system("/usr/sbin/service ponies restart")
```



Adding a feature to the UI

- Create feature model, form and view
- Add service entry for the feature
- Make the middleware aware of feature
- Add a startup / shutdown script for the feature
- Javascript to toggle feature on and off

```
${src}/nanobsd/Files/etc/ix.rc.d/ponies
```

```
exit 0
```

Adding a feature to the UI

- Create feature model, form and view
- Add service entry for the feature
- Make the middleware aware of feature
- Add a startup / shutdown script for the feature
- Javascript to toggle feature on and off

``${src}/static/lib/js/freeadmin.js`

```
ponies_on = false;
toggle_ponies = function(obj, onSuccess) {
  if (ponies_on == false) {
    BrowserPonies.start();
    ponies_on = true;
  } else {
    BrowserPonies.stop();
    ponies_on = false;
  }

  return toggle_service(obj, onSuccess);
}
```

Let's see the newly added feature



Adding audio to FreeNAS

- Add sound driver to kernel config
- Add mp3 program to NanoBSD config
- Build images
- Upgrade system

```
${src}/build/nano_env
```

```
...
```

```
add_nano_modules \  
sound sound/driver/ich sound/driver/hda
```

Adding audio to FreeNAS

- Add sound driver to kernel config
- Add mp3 program to NanoBSD config
- Build images
- Upgrade system


```
${src}/nanobsd/os-base
```

```
...  
add_port audio/mpg123
```

Adding audio to FreeNAS

- Add sound driver to kernel config
- Add mp3 program to NanoBSD config
- Build images
- Upgrade system

```
[root@freenas /mnt/vol0/git/freenas]# make  
No git repo choice is set. Please use "make git-  
external" to build as an external developer or "make git-  
internal" to build as an iXsystems internal developer.  
You only need to do this once.  
*** [git-verify] Error code 1
```

Stop in /mnt/vol0/git/freenas.

```
[root@freenas /mnt/vol0/git/freenas]# make git-external  
You are set up for external (github) development. You  
can use the standard make targets (e.g. build or  
release) now.
```



Adding audio to FreeNAS

- Add sound driver to kernel config
- Add mp3 program to NanoBSD config
- Build images
- Upgrade system

FreeNAS with audio



Debugging

- Logging
- /var/log/messages
- /var/log/nginx-*
- /var/log/debug.log
- settings.py: DEBUG = True
- freenas-debug
- Javascript debuggers in web browsers

How to get involved

- FreeNAS always needs help!
- Patches can be submitted
- Pull requests on github can be made
- Documentation can be contributed
- Bugs can be filed
- Use cases can be documented
- Help with translations

Where to get the code

- FreeNAS source code is hosted on GitHub
- git clone <http://github.com/freenas/freenas.git>
freenas
- Must be built on FreeBSD or PC-BSD 9.2 system with a full development environment
- README file at root of checkout contains build requirements and instructions

Resources

- Website: <http://www.freenas.org>
- Docs: <http://doc.freenas.org>
- Forum: <http://forums.freenas.org>
- Bugs: <http://bugs.freenas.org>
- Email: <http://lists.freenas.org>
- IRC channel on freenode

Conclusion

- FreeNAS is 100% open source
- FreeNAS is very powerful
- FreeNAS can be extended and customized to fit your needs
- Demonstrating a simple application will hopefully give those interested that extra kick to do so!
- Everyone can contribute to FreeNAS!



Questions?