



# **Aeolus Project**

Upstream and the future

# The end of **Aeolus**?

I believe Aeolus will die or be subsumed into another upstream project unless we are able to generate substantial upstream activity

# What can we do about it?

The first step is admitting you have a problem.

The second step is understanding *why*.

# What can we do about it?

Questions for discussion:

- Is the project mission compelling?
- Is the software easy to download and set up?
- Is the software easy to use?
- Is it easy to contribute to the project?
- Is there a natural constituency?
- Have we done enough promotion?

# Is the project mission compelling?

The website says Aeolus mission is:

*"To free users from cloud lock-in by making the choice of cloud provider - private, public, or hybrid - a simple launch-time option."*

I think this isn't bad, but we can do better -- especially for small-scale users.

# Is the software easy to download and set up for production?

- For RHEL 6: Yes, *if you buy the product*
- For Fedora 17: No
- For Debian: No
- For OSX: No
- For Fedora 16: Yes!

So our "open-source" project runs only on RHEL (if you pay for it) and a linux distro that is days from EOL. Oops...

# Is the software easy to download and set up for developers?

## YES!

*(Thanks Crag and Richard!)*

Except for Image Factory, of course... which is a problem... and we need way better docs about how, and we need to publicize the hell out of it on the website... (we'll get to that later).

# Is the software easy to use?

Well... it depends.

- The learning curve is like Mt. Everest
- We make users who just want to launch something do a whole bunch of extra stuff (make a deployable?) first
- It's not really clear who the UI is aimed at: end users? cloud admins? template and image builders?

BUT: I've seen far worse.



# Is it easy to contribute?

Well... much easier than it used to be.

- Everything's on github
- Plenty of helpful presence on mailing list, IRC

**BUT:**

- Conductor has > 50 model classes
- Conductor has at least 2 controllers that do exactly the same thing
- If you wanted to add one small feature to help with something... how would you do that?

# Is there a natural constituency?

**YES!**

*(but we're trying to serve all of them with one big app)*

Our personas:

- Infrastructure consumer
- Infrastructure admin
- System designer
- Security admin
- Purse string holder

All these folks will benefit from Aeolus, and the first three are natural code contributors.

# Have we done enough promotion?

Well, yes and no...

- Fudcon and other talks have been helpful
- Website looks good, is inviting, etc.

BUT

- Last news item on website is about Image Warehouse, from August
- Everything on the "About" page is wrong
- We rarely send release announcements, and not to the right lists

# So, what can we do about it?

The score so far:

- Is the project mission compelling?  
*✗ could be better*
- Is the software easy to download and set up?  
*✗ for real-world use, no*  
*✓ for developer use, yes*
- Is the software easy to use? *✗ no*
- Is it easy to contribute? *✗ no*
- Is there a natural constituency? *✓ yes*
- Have we done enough promotion? *✗ no*

# So, what can we do about it?

My take:

- Look at the personas and re-think the mission. Then think about what UI they each want and MORE IMPORTANTLY what API they each want.
- Make the software work well on Debian/Ubuntu and on OSX

*(continued)*

# So, what can we do about it?

My take, continued:

- Break the project into more bite-sized, individually valuable chunks to encourage outside contributions (and keep our sanity)
- Make the website the responsibility of the entire team

What do *you* think?

# Who Is Aeolus For?

To understand Aeolus' mission, we need to know who our users are and what they do

# Aeolus Personas

<http://redmine.aeolusproject.org/redmine/projects/aeolus/wiki/Personas>

- Marty (*infrastructure manager*)
  - 10% of usage
- Sarah (*security administrator*)
  - 10% of usage
- Sam (*system administrator*)
  - 40% of usage. GUI: medium; CLI: high
- Dennis (*application designer*)
  - 40% of usage. GUI: medium; CLI: medium
- Connor (*infrastructure consumer*)
  - 50% of target users do some of this, not currently well served by Conductor app. GUI: high; CLI: medium



# What does Sam do?

- Manage access to cloud resources
- Add and remove cloud resources from the system
- Aggregate cloud resources for users according to quality, cost, other factors
- Create and view reports on quality and usage to manage cloud resource aggregation
- Manage network and storage configuration for provider clouds, aggregated clouds
- Manage access to app catalogs for aggregated clouds

# What tools does Sam prefer?

- Solid CLI
- Scriptable API for actions and reporting
- Hooks for custom interactions
- Alerts and alert management
- GUI for at-a-glance reporting, CLI for day-to-day actions

# What does Dennis do?

- Design system images
- Design multi-system configurations, with parameter dependencies
- Check deployability of multi-system configurations ("are my images available on a particular cloud")
- Test multi-system configurations
- Publish tested applications
- Grant users permission to use applications (?)

# What tools does Dennis prefer?

- CLI for importing template definitions and app definitions
- API for automatically doing async stuff (building or v2v-ing lots of apps e.g.)
- CLI import/export of popular app definition formats (AppForm, TOSCA, Cloud Formations)
- CLI to launch/test for specific clouds
- GUI for viewing app configs, build status at a glance, user permissions on apps

# What does Connor do?

- Launch system configurations on a specific cloud
- Launch system configurations on an aggregated cloud
- Monitor system configurations over time
- Use system configurations that are highly available
- Use system configurations that automatically scale with load
- View reports on his resource usage and available future resource usage

# What tools does Connor prefer?

- GUI for system status reporting and lifecycle ops
- CLI for lifecycle ops
- Async alerts
- Quick monitoring via GUI
- Console access via GUI or CLI
- API for scripting launches

# What do Sam, Connor, and Dennis want, then?

- They all want various CLI tools they can use for apps, clouds, and aggregated clouds
- They all want APIs they can script for common operations -- adding providers and setting permissions for Sam, building apps for Dennis, launching clumps for Connor
- They want solid GUI reporting but they don't want it getting in their way
- They (probably) don't want one big tool they have to maintain to do all this

# **Break time!**

In small groups, come up with a better mission statement for Aeolus



# Proposed Aeolus mission stmts.

- Aeolus mission: easy management of your applications on all your clouds
- Aeolus mission: to provide superior tools and workflows for flexible construction, management, and monitoring of multi-instance systems across clouds (*we picked this one*)
- Aeolus is: an easy-to-use, multi-instance deployment management tool covering the whole cloud application lifecycle, from defining and building images to cross-cloud launch, scaling, and monitoring
- Aeolus mission: to manage cloud application lifecycle across any provider

# Governance

# Why bother?

- Avoid endless debates
- Keep on top of the website and upstream docs
- Keep on top of conferences
- Help subprojects align with overall goals

# Proposed committees

- APIs and technical issues
  - Resolve technical questions
  - Settle disputes or API issues between sub-projects
- Website
  - Keep main site content up to date
  - Hassle developers for new-feature pages
  - Make sure feature pages are obvious to website viewers
- Publicity (conferences, slides, papers)
  - Track conferences we should care about and make sure everyone knows about them

*(continued...)*

# Proposed committees

- **Publicity (continued)**
  - Keep a repository of slides and papers we use for talks
  - Identify topics for new papers and bug authors to write them
- **Releases**
  - Identify when a sprint roll-up is substantial enough for a new upstream release
  - Document any installation changes or issues with the new release
  - Announce the release on the website and appropriate mailing lists

# Committee membership

- 3-4 members?
- Switch off every release
- Elected committee chair
- Weekly meetings, or as required
- Meeting agendas to the list before meetings; minutes to the list afterwards.

# **Aeolus Priorities**

Prioritized features for the next two releases

# Aeolus next steps

- Upstream release next: priorities
  - OpenStack support -- Factory 2.0 -- TIM integration -- Build/Push UI rework
  - Rails 3.2
  - Integration testing
  - Website cleanup
  - Publicity cleanup
  - Some level of expanded API
  - Converge: Forms rework
  - Object names (AppForm Blueprints?)
  - What do we ship -- rpms? gems? For what distros?
    - Install DC-core
    - Install Factory/Oz
    - Fedora 18? Centos 6? Debian gem-only?



# Aeolus next steps

- Upstream release next +1: priorities
    - API
      - Lifecycle: Restful endpoints for current model objects
      - Lifecycle exposed for Winged Monkey
      - Hooks (blocking)
      - Hooks (non-blocking)
      - Tracker (to support hooks)
      - Lifecycle: Additional Heat-compatible layer?
    - Winged Monkey 0.1
    - Navigation rework
- (continued)

# Aeolus next steps

- Upstream release next +1: priorities (cont.d)
  - CLI (to match what currently exists)
  - Cost-o-matic (cost-based provider selection UI)
  - Stateful instance support
  - Rackspace/Google Compute/Azure
  - Debian/Ubuntu guest support
  - Dashboard?
  - Notifications?
  - Heat?

# Aeolus tech cabal

Mission: Make technical and architectural decisions across the Aeolus project. Settle disputes.

Proposed membership:

- Tomas Sedovic
- Martyn Taylor
- Jason Guiditta
- Scott Seago
- Eric Helms
- Michal Fojtik
- Greg Blomquist
- Jaromir Coufal (user experience advocate)

# Aeolus Website cabal

Mission: Keep [aeolusproject.org](http://aeolusproject.org) current and relevant. Identify and manage writing of new content (feature descriptions, documentation)

Proposed membership:

- Jirka Stransky
- Matt W. (awesome)
- Justin Clift
- Tzu-mainn Chen
- Andy Fitzsimon
- Francesco Vollero

# Aeolus Publicity cabal

Mission: Manage and raise Aeolus Project public profile. Identify speaking opportunities, maintain repos of slides and whitepapers, manage social media.

Proposed membership:

- Andy Fitzsimon
- Angus (awesome)
- Justin Clift
- Francesco Vollero
- Jarda Coufal
- Tzu-mainn Chen

# Aeolus Release cabal

Mission: Test, package, and ship upstream release components. Manage release process and downstream packaging. Produce release process documentation for team members. Produce release announcements.

Proposed membership:

- Martin Povolny
- Justin Clift
- Steve Linabery (awesome)
- Giulio Fidente