

Managing Python Packages with Pulp

Presented by:

Michael Hrivnak

Principal Software Engineer

mhrivnak@redhat.com

@michael_hrivnak



<http://www.pulpproject.org>



Distributing software is a messy business.



What is Pulp?

- Manage repositories of content
- Supports many content types
- Pull-through cache
- OSS under GPL2
- <https://github.com/pulp>
- Python web app



Create New Repositories

- Create a new repository in Pulp
- Sync content from a remote repo, manually or on a schedule
- Upload your own content
- Copies are cheap



Publish Your Repository

- Mix and match content, then publish
- Publish can mean many things



```
$ pulp-admin python repo create --repo-id pypi --feed  
https://pypi.python.org/ --package-names numpy,scipy
```

```
Repository [pypi] successfully created
```



```
$ pulp-admin python repo sync run --repo-id pypi
```

```
+-----+  
                Synchronizing Repository [pypi]  
+-----+
```

This command may be exited via ctrl+c without affecting the request.

Downloading and processing metadata.

```
[-]  
... completed
```

Downloading and processing Python packages.

```
[=====] 100%  
30 of 30 items  
... completed
```

Task Succeeded



```
$ pip install -i
http://pulp.example.com/pulp/python/web/my_own_pypi/simple/
pulp-python-plugins
Downloading/unpacking pulp-python-plugins
  Downloading pulp-python-plugins-0.0.0.tar.gz
  Running setup.py egg_info for package pulp-python-plugins

Installing collected packages: pulp-python-plugins
  Running setup.py install for pulp-python-plugins

Successfully installed pulp-python-plugins
Cleaning up...
```



Content Types

- RPM family
- Docker Images
- Puppet Modules
- Python Packages
- OSTree
- Regular Files
- Debian Packages (community)
- NPM?



Who uses Pulp?

- Red Hat Release Engineering
- Public Clouds
- Katello / Red Hat Satellite 6
- Community



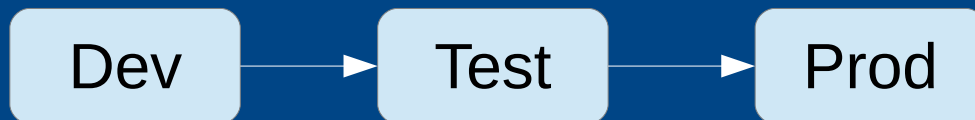
Use Case: Mirror Python Packages

- Sync packages from PyPI to a local repository
- Add or remove packages as desired
- Retain old versions
- Control which versions appear together



Use Case: Dev / Test / Production

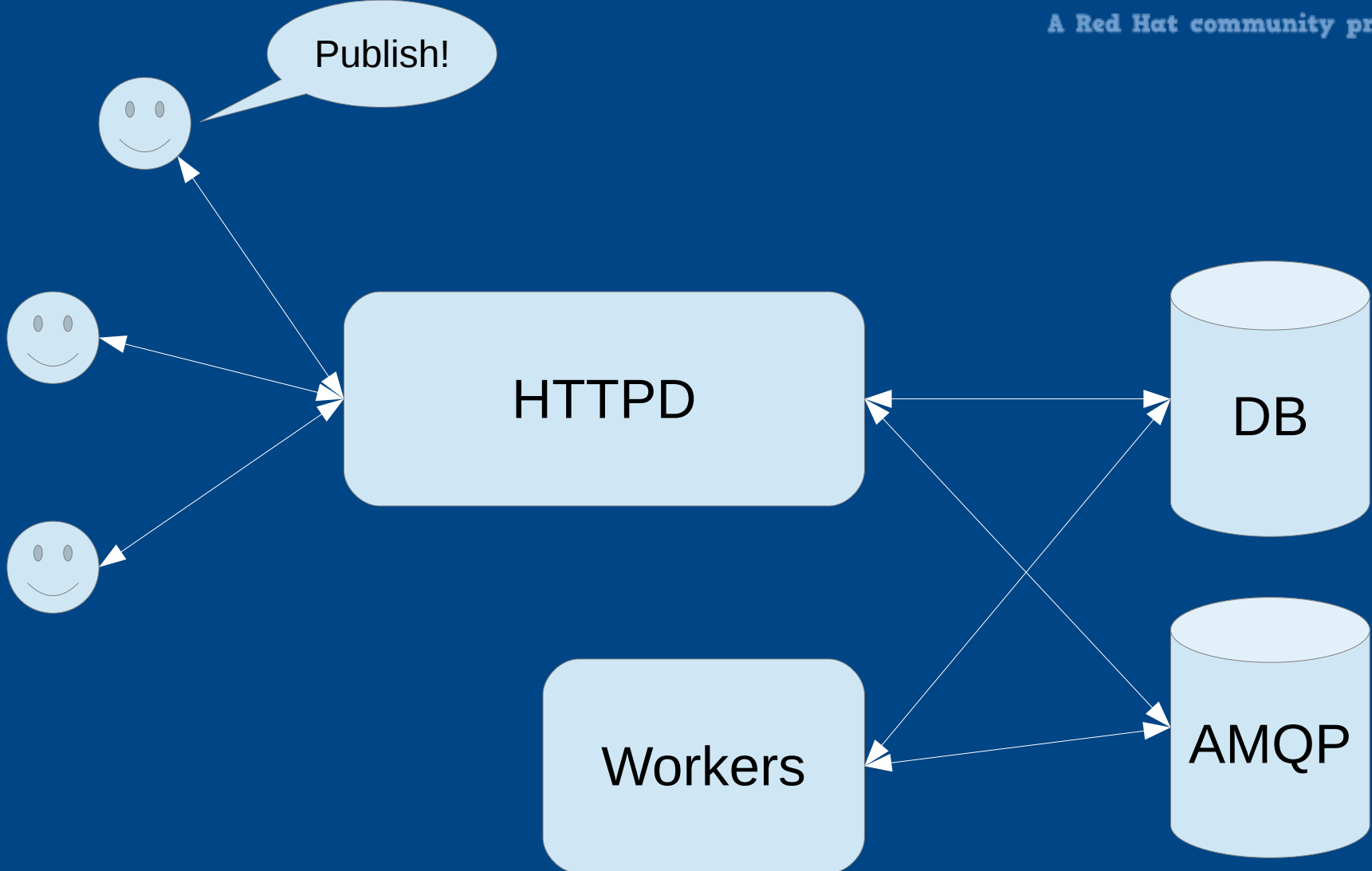
- Sync content into a development repository
- “Promote” by copying to a Test repository, and then to a Production repository
- Useful for testing upstream content, such as new RHEL point releases



Distributed Application

- REST API
- Content served via HTTP
- Worker Processes for Async Jobs





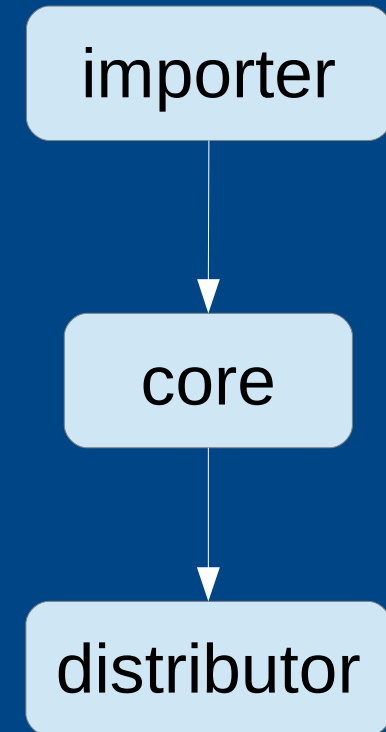
Extensible

- Standard ways to support new content types
 - How does content flow in?
 - How does content flow out?



Plugins

- Type Definition
 - Model Class (using mongoengine)
- Importer
 - Used to pull in upstream content
 - One per repository
- Distributor
 - Used to publish content
 - Many per repository

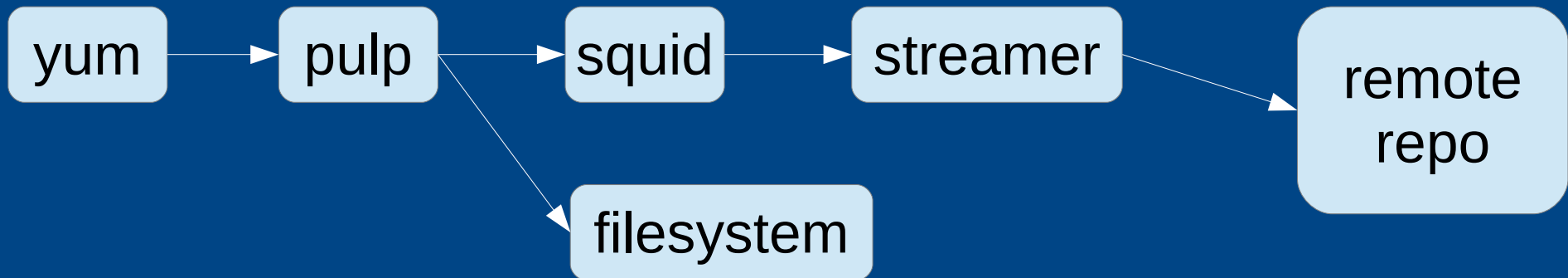


Integration

- REST API
- Events published to AMQP topic exchange
- HTTP callbacks



Pull-through Cache



- Available in 2.8.0 Beta
- Adds metadata to database without downloading files
- Retrieves files on-demand or in the background



Documentation

<http://www.pulpproject.org/docs/>



Stickers!



Questions?



Michael Hrivnak
mhrivnak@redhat.com
@michael_hrivnak

<http://www.pulpproject.org>

