API Extensions

Jorge L. Williams

experience fanatical support
WWW.RACKSPACE.COM

rackspace HOSTING

Agenda

- The Problem
- Extensions
- Extensions in REST
- Promoting Extensions
- Challenges

The Problem

experience fanatical support®
WWW.RACKSPACE.COM
rackspace® HOSTING

Standardization vs Innovation and Differentiation

- We want to propose our APIs as Open Standards...
  - Defining Standard APIs good for Rackspace, OpenStack, and our Customers
  - We want to encourage others to implement our APIs
  - Standards need to be stable
    - Hard to develop against something that’s in constant flux
  - Standards need to be general
    - May be impossible for someone to adopt our standards if they are very specific to our business...
      - How do you standardize the idea of Managed Cloud in the Cloud Servers API? Does it make sense to do this?
  - The more general and stable the API, the more likely others will adopt it.
- We want to innovate and allow others to innovate
  - Quickly add features that differentiate Rackspace OpenStack from other implementations
    - Without breaking our clients
    - Without going through an approval process
  - We want to allow others to also make changes to the API
    - More likely to adopt OpenStack APIs if they can be modified
    - We may benefit from these changes
  - Developers should feel free to experiment with new features without worrying about the implications to the API as a standard.

OpenStack

- Open Stack presents another interesting challenge: Others can make changes to the code.
  - Rackspace Version of Cloud Servers API vs.
    - OpenStack Version vs.
    - Other Modified versions.
  - What does Cloud Servers API 1.1 mean if we have different implementations all with different capabilities?
  - How do we ensure compatibility among the different versions?

experience fanatical support
WWW.RACKSPACE.COM

rackspace HOSTING

Extensions

experience fanatical support®
WWW.RACKSPACE.COM
rackspace® HOSTING

Case Study: OpenGL

- The problem we're facing is not new.
  OpenGL faced a similar problem in the 90's
  - How do you define an open graphics library that:
    - Is considered a standard specification
    - Allows vendors to differentiate their products by adding special features
    - And yet is a governed spec
      - An architecture review board (ARB):
        - » Proposes and approves specification changes
        - » Marks new releases
        - » Ensures conformance testing
  - The solution was to allow extensions in the specification
    - Vendors can define special features as extensions
  - A very successful strategy
    - The core OpenGL API is general and uncluttered and an accepted standard.
    - Over 500 extensions have been defined over OpenGL's lifetime
      - Best become standard features; others abandoned
      - Different extensions for the same feature? Let the best one win.
      - Many innovations came via the extension process: vertex and fragment shaders, etc.
      - Extensions have been defined by many different vendors: NVidia, ATI, Apple, IBM, Intel, …

Extensions

- Extensions add capability to the API beyond those of the specification
- An API specification must be written to allow for extensibility
  - We need flexibility in the contract to allow for new data elements, actions, states, headers, and resource types.
  - The core API specification defines the extension mechanism, but extensions themselves are not part of the core.
- Implementors are only required to implement the core API.
- Extensions can be promoted
  - Extensions follow a promotion path, at the end of which an extension may become part of the next version of the core API.

Extensions vs. Versions

Versions	Extensions
Centralized: Versions are maintained by the entity that controls the API Spec: The OpenStack Architecture Board. Only the ARB can create a new version, only the ARB defines what OpenStack Compute 1.1 means.	**Decentralized:** Extensions are maintained by third parties: Rackspace, OpenStack developers, etc. Anyone can create an Extension.
Deal with **Core Functionality**	Deal with **Specialized/Niche Functionality**
Appear infrequently: Versions provide a stable platform on which to develop.	**Appear frequently:** Extensions bring new features to the market quickly, and in a compatible manner.
Are Queryable: You can programmatically tell what versions are available by doing a GET on the base URL (/) of the API endpoint.	**Are Queryable:** You can programmatically tell what extensions are available by doing a GET on the extensions resource (/v1.1/extensions).

Extensions vs. Versions

- Our APIs should be both Extensible and Versionable

Extensions in REST

experience fanatical support
WWW.RACKSPACE.COM
rackspace HOSTING

Sample Extension Query

- Extensions are queryable via **/extensions**

```
<extensions xmlns="http://docs.openstack.org/api-specs/v1.0"
            xmlns:atom="http://www.w3.org/2005/Atom"
            >
    <extension name="Public Image Extension"
               namespace="http://docs.rackspacecloud.com/servers/api/ext/pie/v1.0"
               alias="RS-PIE"
               >
        <atom:link rel="describedby" type="application/pdf"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-pie-20111111.pdf"/>
        <atom:link rel="describedby" type="application/vnd.sun.wadl+xml"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-pie.wadl"/>
        <description>
            Adds the capability to share an image with other users.
        </description>
    </extension>
    <extension name="Cloud Block Storage"
               namespace="http://docs.rackspacecloud.com/servers/api/ext/cbs/v1.0"
               alias="RS-CBS"
               >
        <atom:link rel="describedby" type="application/pdf"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-cbs-20111201.pdf"/>
        <atom:link rel="describedby" type="application/vnd.sun.wadl+xml"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-cbs.wadl"/>
        <description>
            Allows mounting cloud block storage volumes.
        </description>
    </extension>
</extensions>
```

Sample Extension Query

- Human Readable Name and Description

```
<extensions xmlns="http://docs.openstack.org/api-specs/v1.0"
            xmlns:atom="http://www.w3.org/2005/Atom"
            >
    <extension  name="Public Image Extension"
                namespace="http://docs.rackspacecloud.com/servers/api/ext/pie/v1.0"
                alias="RS-PIE"
                >
        <atom:link rel="describedby" type="application/pdf"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-pie-20111111.pdf"/>
        <atom:link rel="describedby" type="application/vnd.sun.wadl+xml"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-pie.wadl"/>
        <description>
            Adds the capability to share an image with other users.
        </description>
    </extension>
    <extension name="Cloud Block Storage"
                namespace="http://docs.rackspacecloud.com/servers/api/ext/cbs/v1.0"
                alias="RS-CBS"
                >
        <atom:link rel="describedby" type="application/pdf"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-cbs-20111201.pdf"/>
        <atom:link rel="describedby" type="application/vnd.sun.wadl+xml"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-cbs.wadl"/>
        <description>
            Allows mounting cloud block storage volumes.
        </description>
    </extension>
</extensions>
```

Sample Extension Query

- Links to Documentation (in different formats)

```
<extensions xmlns="http://docs.openstack.org/api-specs/v1.0"
            xmlns:atom="http://www.w3.org/2005/Atom"
            >
    <extension name="Public Image Extension"
               namespace="http://docs.rackspacecloud.com/servers/api/ext/pie/v1.0"
               alias="RS-PIE"
               >
        <atom:link rel="describedby" type="application/pdf"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-pie-20111111.pdf"/>
        <atom:link rel="describedby" type="application/vnd.sun.wadl+xml"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-pie.wadl"/>
        <description>
            Adds the capability to share an image with other users.
        </description>
    </extension>
    <extension name="Cloud Block Storage"
               namespace="http://docs.rackspacecloud.com/servers/api/ext/cbs/v1.0"
               alias="RS-CBS"
               >
        <atom:link rel="describedby" type="application/pdf"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-cbs-20111201.pdf"/>
        <atom:link rel="describedby" type="application/vnd.sun.wadl+xml"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-cbs.wadl"/>
        <description>
            Allows mounting cloud block storage volumes.
        </description>
    </extension>
</extensions>
```

Sample Extension Query

- Unique Extension IDs

```
<extensions xmlns="http://docs.openstack.org/api-specs/v1.0"
            xmlns:atom="http://www.w3.org/2005/Atom"
            >
    <extension name="Public Image Extension"
               namespace="http://docs.rackspacecloud.com/servers/api/ext/pie/v1.0"
               alias="RS-PIE"
               >
        <atom:link rel="describedby" type="application/pdf"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-pie-20111111.pdf"/>
        <atom:link rel="describedby" type="application/vnd.sun.wadl+xml"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-pie.wadl"/>
        <description>
            Adds the capability to share an image with other users.
        </description>
    </extension>
    <extension name="Cloud Block Storage"
               namespace="http://docs.rackspacecloud.com/servers/api/ext/cbs/v1.0"
               alias="RS-CBS"
               >
        <atom:link rel="describedby" type="application/pdf"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-cbs-20111201.pdf"/>
        <atom:link rel="describedby" type="application/vnd.sun.wadl+xml"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-cbs.wadl"/>
        <description>
            Allows mounting cloud block storage volumes.
        </description>
    </extension>
</extensions>
```

Sample Extension Query

- Vendor Identifiers

```
<extensions xmlns="http://docs.openstack.org/api-specs/v1.0"
            xmlns:atom="http://www.w3.org/2005/Atom"
            >
    <extension name="Public Image Extension"
               namespace="http://docs.rackspacecloud.com/servers/api/ext/pie/v1.0"
               alias="RS-PIE"
               >
        <atom:link rel="describedby" type="application/pdf"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-pie-20111111.pdf"/>
        <atom:link rel="describedby" type="application/vnd.sun.wadl+xml"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-pie.wadl"/>
        <description>
            Adds the capability to share an image with other users.
        </description>
    </extension>
    <extension name="Cloud Block Storage"
               namespace="http://docs.rackspacecloud.com/servers/api/ext/cbs/v1.0"
               alias="RS-CBS"
               >
        <atom:link rel="describedby" type="application/pdf"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-cbs-20111201.pdf"/>
        <atom:link rel="describedby" type="application/vnd.sun.wadl+xml"
                   href="http://docs.rackspacecloud.com/servers/api/ext/cs-cbs.wadl"/>
        <description>
            Allows mounting cloud block storage volumes.
        </description>
    </extension>
</extensions>
```

Vendor Identifiers

- An extension alias always contains a prefix that identifies the vendor. Prefixes are **not** case sensitive:

```
<extension name="Public Image Extension"
               namespace="http://docs.rackspacecloud.com/servers/api/ext/pie/v1.0"
               alias="RS-PIE">
```

Prefix	Vendor
OS	OpenStack
MLTI	Multi-Vendor
ARB	ARB Approved
RS	Rackspace
NASA	Nasa
CTX	Citrix
...	...

Vendor Identifiers

- Namespaces also help ID the vendor

Namespace	Vendor
http://docs.openstack.com/ext/OS/...	OpenStack
http://docs.openstack.com/ext/ARB/...	ARB Approved
http://docs.rackspacecloud.com/...	Rackspace
http://docs.nasa.org/...	Nasa
http://docs.citrix.com/....	Citrix
...	...

Vendor ID Registry

- OpenStack should maintain a registry of Vendor IDs (prefix and namespaces).
- Anyone should be able to register a Vendor ID.

What can be extended

- Extensions may define:
  - New data types, elements, attributes
  - New actions
  - New headers
  - New states
  - New resources

experience fanatical support
WWW.RACKSPACE.COM

rackspace HOSTING

Data Extensions

- Add additional Data.
  - In XML, attribute may be added to elements so long as they are in the extension namespace
  - In XML, Elements added after last element assuming “Unique Particle Attribution” is not violated
  - In JSON, use alias followed by a colon “:”

```
<image xmlns="http://docs.rackspacecloud.com/servers/api/v1.0"
       xmlns:RS-PIE="http://docs.rackspacecloud.com/servers/api/ext/pie/v1.0"
       id="1" name="CentOS 5.2"
       serverId="12"
       updated="2010-10-10T12:00:00Z"
       created="2010-08-10T12:00:00Z"
       status="ACTIVE"
       RS-PIE:shared="true"
       />

{
    "image" : {
        "id" : 1,
        "name" : "CentOS 5.2",
        "serverId" : 12,
        "updated" : "2010-10-10T12:00:00Z",
        "created" : "2010-08-10T12:00:00Z",
        "status" : "ACTIVE",
        "RS-PIE:shared" : true
    }
}
```

New Actions

- In XML, actions are defined in the extension namespace
- In JSON, use alias followed by a colon “:” for the action name

```
<mount_volume xmlns="http://docs.rackspacecloud.com/servers/api/ext/cbs/v1.0"
    CBSID="123"/>
```

```
{
    "RS-CBS:mount_volume" : {
        "CBSID" : "123"
    }
}
```

New Headers and States

- With headers, append name with an X- followed by the alias
  - X-**RS-CBS**-Header1: Value
  - X-**RS-CBS**-Header2: Value
- With states, use alias followed by a ":"

```
<image xmlns="http://docs.rackspacecloud.com/servers/api/v1.0"
       xmlns:RS-PIE="http://docs.rackspacecloud.com/servers/api/ext/pie/v1.0"
       id="1" name="CentOS 5.2"
       serverId="12"
       updated="2010-10-10T12:00:00Z"
       created="2010-08-10T12:00:00Z"
       status="RS-PIE:PrepareShare" progress="80"
       RS-PIE:shared="true"
       />

{
    "image" : {
        "id" : 1,
        "name" : "CentOS 5.2",
        "serverId" : 12,
        "updated" : "2010-10-10T12:00:00Z",
        "created" : "2010-08-10T12:00:00Z",
        "status" : "RS-PIE:PrepareShare",
        "progress" : 80,
        "RS-PIE:shared" : true
    }
}
```

experience fanatical support®
WWW.RACKSPACE.COM

rackspace® HOSTING

New Resources

- Extensions are always defined at /path/to/resource/**ext/ext-alias/newResource** All major resources can reference a /ext
- A CBS Volume: /v1.0/12345/servers/ext/RS-CBS/volume

Promoting Extensions

experience fanatical support®
WWW.RACKSPACE.COM

rackspace® HOSTING

New Features Should Start as Extensions

- This gives us the ability to try things out before a feature enters the standard.
- Allows competing extensions to co-exist

experience fanatical support®
WWW.RACKSPACE.COM

rackspace® HOSTING

Promotion Path

- Extensions may follow a promotion path
  - Vendor Specific → ARB Approved → Core Feature
- Some extensions may be developed by multiple vendors; these are known as Multi-Vendor extension, the prefix is MLTI.
  - Multi-Vendor (MLTI) → ARB Approved → Core Feature
- An extension may start as a vendor specific extension and become a multi-vendor extension.
  - Vendor Specific → Multi-Vendor (MLTI) → ARB Approved → Core Feature

experience fanatical support®
WWW.RACKSPACE.COM
rackspace® HOSTING

Promotion Path

- Not all extensions should be promoted to core features
  - Extensions may implement niche functionality that doesn't make sense in the core API.

experience fanatical support
WWW.RACKSPACE.COM

rackspace HOSTING

ARB Approved Extensions

- The ARB “blesses” an extension by making it an ARB-approved extension.
- ARB-approved extensions use ARB as the vendor prefix.
- An ARB-approved extension denotes
  - That the extension is being considered for the next revision of the specification
  - That extension is a niche extension that is very useful; it may not make it as a standard feature, but implementors are encouraged to implement it nonetheless.

Challenges

experience fanatical support®
WWW.RACKSPACE.COM
rackspace® HOSTING

Implementation Challenges

- Services must be implemented in such a manner that the extensible part of the code is separate from the core implementation. This is doable with modern service toolkits, but must be done with care.
  - Filter approach: The extension is implemented via middle-ware filters
  - Sub-Type Services: Services can be extended in OOP languages; extensions can be written in separate service implementations.

WADLs

- In order to promote accuracy the default service WADL should contain a description of all of the extensions active in a particular implementation.
  – Doing a get on version details (**/v1.1/**) should give you a pointer to a WADL that describes the service with all available extensions.
  – There may be a need to maintain multiple WADLs, these WADLs describe only the changes made by the extension. These are the WADLs pointed to by the extension resource (**/v1.1/extensions**).
  – We need to educate developers on how to write extensible schemas and WADLs.

Language Bindings

- Extensions should be supported at the language binding layer
  - Language bindings may be written to detect and give access to extensions given a WADL
  - A simpler approach may be to allow the language bindings themselves to be extensible, so that extensions may be simply added to an existing binding.
    - The language binding framework, should support this even if we didn't define extensions, as this helps with version changes.

Thanks

experience fanatical support®
WWW.RACKSPACE.COM
rackspace® HOSTING