

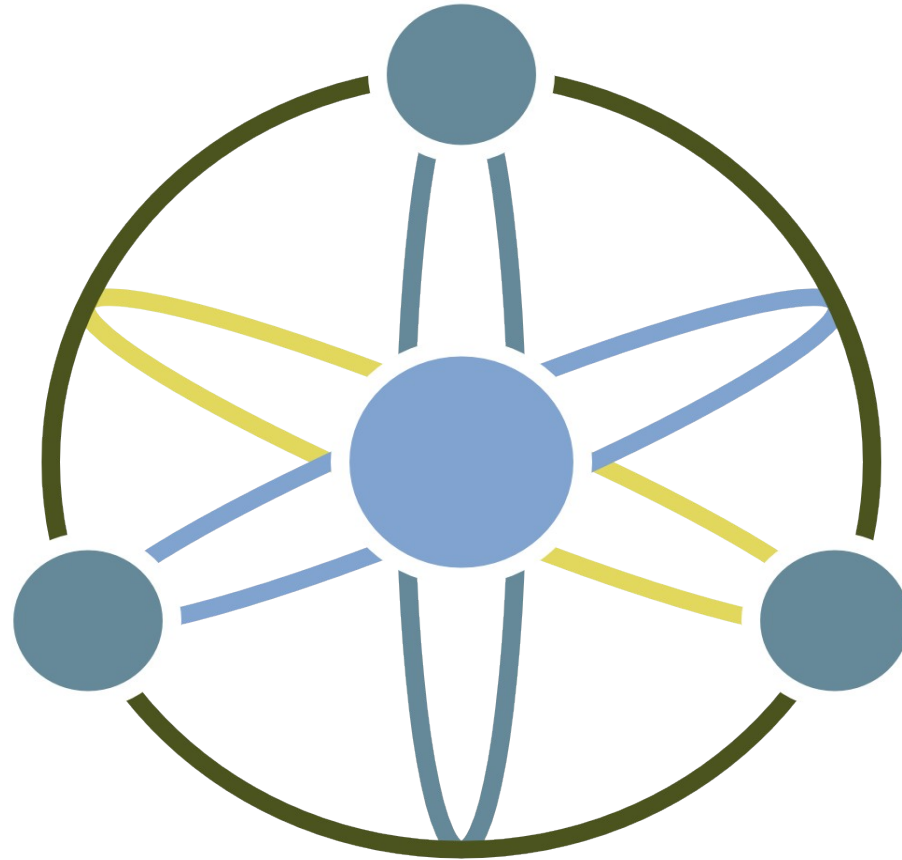
A decorative header featuring four overlapping spheres: a green one on the left, and blue, red, and yellow ones on the right, all partially cut off by the top edge of the slide.

Intro to the Atom Publishing Protocol

Joe Gregorio
Google



Atom



Atom Syndication Feed

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author>
    <name>John Doe</name>
  </author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>

  <entry>
    <title>Atom-Powered Robots Run Amok</title>
    <link href="http://example.org/2003/12/13/atom03" />
    <id>urn:uuid:1225c695-cfb8-4ebb-80da344efa6a</id>
    <updated>2003-12-13T18:30:02Z</updated>
    <summary>Some text.</summary>
    <link rel="edit" href="http://example.org/edit/1/" />
  </entry>
  ...
</feed>
```



Atom Syndication Feed

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author>
    <name>John Doe</name>
  </author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>

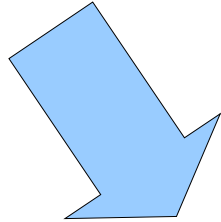
  <entry>
    <title>Atom-Powered Robots Run Amok</title>
    <link href="http://example.org/2003/12/13/atom03" />
    <id>urn:uuid:1225c695-cfb8-4ebb80da344efa6a</id>
    <updated>2003-12-13T18:30:02Z</updated>
    <summary>Some text.</summary>
    <link rel="edit" href="http://example.org/edit/1/" />
  </entry>

  ...
</feed>
```



Feed URI

<http://example.org/feed/>



GET

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author>
    <name>John Doe</name>
  </author>
  <id>urn:uuid:60a76c80-d399-11d9-0003939e0af6</id>

  <entry>
    <title>Atom-Powered Robots Run Amok</title>
    <link href="http://example.org/2003/12/13/atom03" />
    <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
    <updated>2003-12-13T18:30:02Z</updated>
    ...
```



Protocol Requirements

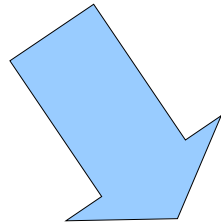
Create
Retrieve
Update
Delete

Protocol Requirements

Create
Retrieve
Update
Delete

Feed URI

```
<entry>
  <author>
  <title>Robot False Alarm</title>
  <id>urn:uuid:1225c695-cfb8-4ebb-80da344efa6a</id>
  <updated>2003-12-14T10:33:00Z</updated>
  <content>Sorry about the false alarm.</content>
</entry>
```



POST

<http://example.org/feed/>

Feed

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author><name>John Doe</name></author>
  <id>urn:uuid:60a76c80-d399-11d9-0003939e0af6</id>

  <entry>
    <title>Robot False Alarm</title>
    <link href="http://example.org/2003/12/14/norobots" />
    <id>urn:uuid:1225c695-cfb8-4ebb-80da344efa6a</id>
    <updated>2003-12-14T10:33:00Z</updated>
    <content>Sorry about the false alarm.</content>
    <link rel="edit" href="http://example.org/edit/2/" />
  </entry>
  ...
</feed>
```



Protocol Requirements

Create

Retrieve

Update

Delete

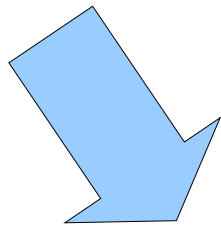
Feed

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author><name>John Doe</name></author>
  <id>urn:uuid:60a76c80-d399-11d9-0003939e0af6</id>

  <entry>
    <title>Robot False Alarm</title>
    <link href="http://example.org/2003/12/14/norobots" />
    <id>urn:uuid:1225c695-cfb8-4ebb-80da344efa6a</id>
    <updated>2003-12-14T10:33:00Z</updated>
    <content>Sorry about the false alarm.</content>
    <link rel="edit" href="http://example.org/edit/2/" />
  </entry>
  ...
</feed>
```



Remove Entry



DELETE

<http://example.org/feed/2/>

Protocol Requirements

Create

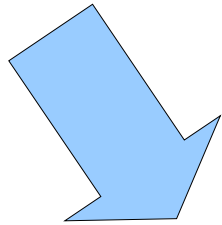
Retrieve

Update

Delete

Edit URI

<http://example.org/feed/2/>

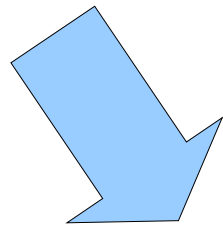


GET

```
<entry xmlns="http://www.w3.org/2005/Atom">
  <title>Robot False Alarm</title>
  <link
    href="http://example.org/2003/12/14/norobots"/>
  <id>urn:uuid:1225c695-cfb8-4ebbaaaa-a80d...</id>
  <updated>2003-12-14T10:33:00Z</updated>
  <content>Sorry about the false alarm.</content>
  <link rel="edit" href="http://example.org/edit/2/" />
</entry>
```

Update

```
<entry xmlns="http://www.w3.org/2005/Atom">
  <title>Atom Powered Robot False Alarm</title>
  <link
    href="http://example.org/2003/12/14/norobots"/>
  <id>urn:uuid:1225c695-cfb8-4ebbaaaa-80da344efa6a</id>
  <updated>2003-12-14T10:33:00Z</updated>
  <summary>Sorry about the false alarm.</summary>
  <link rel="edit" href="http://example.org/edit/2/" />
</entry>
```



PUT

<http://example.org/feed/2/>

Protocol Requirements

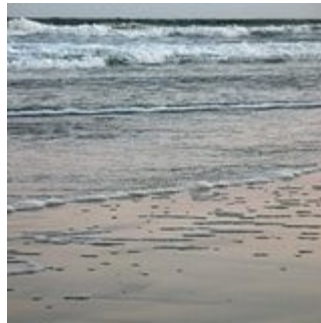
Create

Retrieve

Update

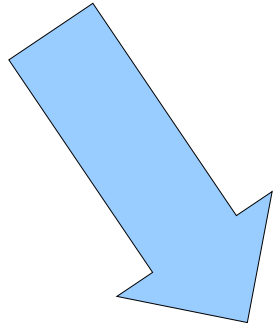
Delete

Media



Google™

Media



POST
Slug: The Beach

<http://example.org/feed/>

Feed

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author><name>John Doe</name></author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>
  <entry xmlns="http://www.w3.org/2005/Atom">
    <title>The Beach</title>
    <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
    <updated>2005-10-07T17:17:08Z</updated>
    <summary type="text" />
    <content type="image/png"
      src="http://media.example.org/the_beach.png" />
    <link rel="edit-media"
      href="http://media.example.org/edit/the_beach.png" />
    <link rel="edit"
      href="http://example.org/media/edit/the_beach.atom" />
  </entry>
  ...
</feed>
```



Feed

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author><name>John Doe</name></author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>
  <entry xmlns="http://www.w3.org/2005/Atom">
    <title>The Beach</title>
    <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
    <updated>2005-10-07T17:17:08Z</updated>
    <summary type="text" />
    <content type="image/png"
      src="http://media.example.org/the_beach.png" />
    <link rel="edit-media"
      href="http://media.example.org/edit/the_beach.png" />
    <link rel="edit"
      href="http://example.org/media/edit/the_beach.atom" />
  </entry>
  ...
</feed>
```

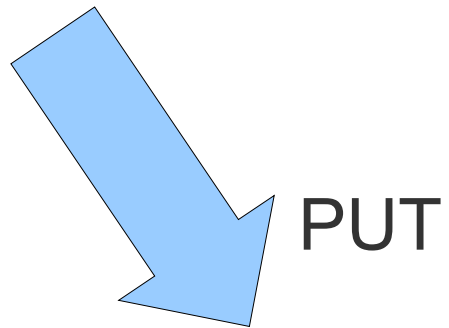


Feed

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author><name>John Doe</name></author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>
  <entry xmlns="http://www.w3.org/2005/Atom">
    <title>The Beach</title>
    <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
    <updated>2005-10-07T17:17:08Z</updated>
    <summary type="text" />
    <content type="image/png"
      src="http://media.example.org/the_beach.png" />
    <link rel="edit-media"
      href="http://media.example.org/edit/the_beach.png" />
    <link rel="edit"
      href="http://example.org/media/edit/the_beach.atom" />
  </entry>
  ...
</feed>
```



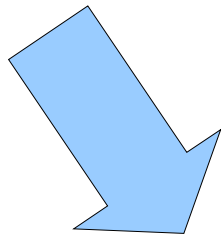
Update Media



`http://media.example.org/edit/the_beach.png`

Update Media Link Entry

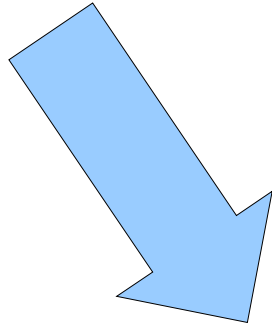
```
<entry xmlns="http://www.w3.org/2005/Atom">
  <title>A Shark At The Beach</title>
  <id>urn:uuid:1225c695-cfb8-4ebb-80da344efa6a</id>
  <updated>2005-10-07T17:17:08Z</updated>
  <summary type="text" />
  <content type="image/png"
    src="http://media.example.org/the_beach.png"/>
  <link rel="edit-media"
    href="http://media.example.org/edit/the_beach.png" />
  <link rel="edit"
    href="http://example.org/media/edit/the_beach.atom" />
</entry>
```



PUT

http://example.org/media/edit/the_beach.atom

Remove Media



DELETE

`http://example.org/media/edit/the_beach.atom`



Corrections

More than one collection



Service Document

```
<?xml version="1.0" encoding='utf-8'?>
<service xmlns="http://www.w3.org/2007/app"
        xmlns:atom="http://www.w3.org/2005/Atom">
  <workspace>
    <atom:title>Main Site</atom:title>
    <collection
      href="http://example.org/blog/main" >
      <atom:title>My Blog Entries</atom:title>
    </collection>
    <collection
      href="http://example.org/blog/pic" >
      <atom:title>Pictures</atom:title>
      <accept>image/png</accept>
      <accept>image/jpeg</accept>
      <accept>image/gif</accept>
    </collection>
  </workspace>
</service>
```



Service Document

```
<?xml version="1.0" encoding='utf-8'?>
<service xmlns="http://www.w3.org/2007/app"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <workspace>
    <atom:title>Main Site</atom:title>
    <collection
      href="http://example.org/blog/main" >
      <atom:title>My Blog Entries</atom:title>
    </collection>
    <collection
      href="http://example.org/blog/pic" >
      <atom:title>Pictures</atom:title>
      <accept>image/png</accept>
      <accept>image/jpeg</accept>
      <accept>image/gif</accept>
    </collection>
  </workspace>
</service>
```



Service Document

```
<?xml version="1.0" encoding='utf-8'?>
<service xmlns="http://www.w3.org/2007/app"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <workspace>
    <atom:title>Main Site</atom:title>
    <collection
      href="http://example.org/blog/main" >
      <atom:title>My Blog Entries</atom:title>
    </collection>
    <collection
      href="http://example.org/blog/pic" >
      <atom:title>Pictures</atom:title>
      <accept>image/png</accept>
      <accept>image/jpeg</accept>
      <accept>image/gif</accept>
    </collection>
  </workspace>
</service>
```



Service Document

```
<?xml version="1.0" encoding='utf-8'?>
<service xmlns="http://www.w3.org/2007/app"
        xmlns:atom="http://www.w3.org/2005/Atom">
  <workspace>
    <atom:title>Main Site</atom:title>
    <collection
      href="http://example.org/blog/main" >
      <atom:title>My Blog Entries</atom:title>
    </collection>
    <collection
      href="http://example.org/blog/pic" >
      <atom:title>Pictures</atom:title>
      <accept>image/png</accept>
      <accept>image/jpeg</accept>
      <accept>image/gif</accept>
    </collection>
  </workspace>
</service>
```



Service Document

```
<?xml version="1.0" encoding='utf-8'?>
<service xmlns="http://www.w3.org/2007/app"
        xmlns:atom="http://www.w3.org/2005/Atom">
  <workspace>
    <atom:title>Main Site</atom:title>
    <collection
      href="http://example.org/blog/main" >
      <atom:title>My Blog Entries</atom:title>
    </collection>
    <collection
      href="http://example.org/blog/pic" >
      <atom:title>Pictures</atom:title>
      <accept>image/png</accept>
      <accept>image/jpeg</accept>
      <accept>image/gif</accept>
    </collection>
  </workspace>
</service>
```





The End

RFC 4287

RFC 5023

Jeff Fisher – Intro to Google Data on YouTube

<http://code.google.com/apis/gdata/>





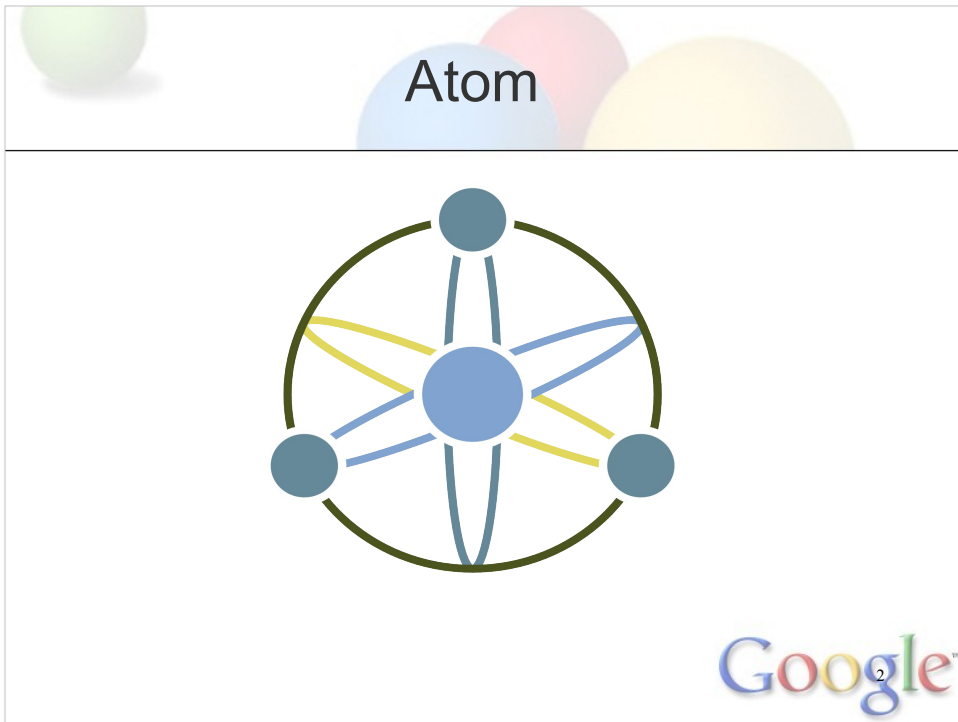
Intro to the Atom Publishing Protocol

Joe Gregorio
Google



The Atom Publishing Protocol is a simple protocol for publishing content on the web. It uses the Atom Syndication Format and works over HTTP.

This is an introduction to the Atom Publishing Protocol, meant to give you a high-level overview of the protocols fundamentals. Watching this will make it easier read RFC 5023, the specification that defines the Atom Publishing Protocol, something you should do since I will skip some of the details of the protocol.



Just as syndication feeds started out initially for weblogs and then got applied to other areas, the same is true for the Atom Publishing Protocol.

We will use blogging as our motivating example, but realize that the Atom Publishing Protocol can be applied to a much wider range of activities.

Atom Syndication Feed

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author>
    <name>John Doe</name>
  </author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>

  <entry>
    <title>Atom-Powered Robots Run Amok</title>
    <link href="http://example.org/2003/12/13/atom03" />
    <id>urn:uuid:1225c695-cfb8-4ebb-80da344efa6a</id>
    <updated>2003-12-13T18:30:02Z</updated>
    <summary>Some text.</summary>
    <link rel="edit" href="http://example.org/edit/1/" />
  </entry>
  ...
</feed>
```



To begin, let's look at a feed in the Atom Syndication Format.

Atom Syndication Feed

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author>
    <name>John Doe</name>
  </author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>

  <entry>
    <title>Atom-Powered Robots Run Amok</title>
    <link href="http://example.org/2003/12/13/atom03" />
    <id>urn:uuid:1225c695-cfb8-4ebb80da344efa6a</id>
    <updated>2003-12-13T18:30:02Z</updated>
    <summary>Some text.</summary>
    <link rel="edit" href="http://example.org/edit/1/" />
  </entry>
  ...
</feed>
```



A feed contains one entry for each weblog entry that has been recently created or updated. It contains the data for that weblog entry in a machine readable format.

Feed URI

<http://example.org/feed/>



GET

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author>
    <name>John Doe</name>
  </author>
  <id>urn:uuid:60a76c80-d399-11d9-0003939e0af6</id>

  <entry>
    <title>Atom-Powered Robots Run Amok</title>
    <link href="http://example.org/2003/12/13/atom03" />
    <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
    <updated>2003-12-13T18:30:02Z</updated>
    ...
  </entry>
</feed>
```



The way syndication works is that the feed lives at a URI and you issue an HTTP GET to that URI to retrieve the feed.

In syndication you do those GETs periodically and look for new entries.

If there are new or updated entries they will appear at the beginning of the feed.

Protocol Requirements

Create
Retrieve
Update
Delete



Our publishing protocol and be broken down into four basic operations you want to perform on individual entries. Those are to create new entries, retrieve the entries that make up a weblog, update an entry and remove an entry.

Protocol Requirements

Create
Retrieve
Update
Delete



So a feed actually satisfies one of our operations: Retrieve.
So we've got 1 out of 4 already!

Here's the fundamental idea of the Atom Publishing Protocol:
That if we make changes to the feed, those changes get reflected in the blog.

For various reasons you probably want the 'feed' that goes into an aggregator to be different from the one you use for editing. For example, some entries may be 'drafts' and we don't want those published until we explicitly remove them from 'draft' status, but we still want to be able to edit them with our publishing protocol. We'll differentiate the aggregator feed from the editing feed by calling the editing feed a 'collection feed'.

Feed URI

```
<entry>
  <author>
    <title>Robot False Alarm</title>
    <id>urn:uuid:1225c695-cfb8-4ebb-80da344efa6a</id>
    <updated>2003-12-14T10:33:00Z</updated>
    <content>Sorry about the false alarm.</content>
  </entry>
```



POST

<http://example.org/feed/>



We have Retrieve, how do you add a new entry to a collection?

We do the simplest thing you could think of, you merely send the entry in an HTTP POST request to the URI of the collection feed.

This is just an entry on its own, it's not sitting inside a feed.

Feed

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author><name>John Doe</name></author>
  <id>urn:uuid:60a76c80-d399-11d9-0003939e0af6</id>

  <entry>
    <title>Robot False Alarm</title>
    <link href="http://example.org/2003/12/14/norobots" />
    <id>urn:uuid:1225c695-cfb8-4ebb-80da344efa6a</id>
    <updated>2003-12-14T10:33:00Z</updated>
    <content>Sorry about the false alarm.</content>
    <link rel="edit" href="http://example.org/edit/2/" />
  </entry>
  ...
</feed>
```



That entry gets added to the collection feed.

Like we said before, this change to the feed will also get reflected in the weblog.

Protocol Requirements

Create
Retrieve
Update
Delete



Now we have the Create and Retrieve cases covered.

Let's go back and look at that newly added entry.

Feed

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author><name>John Doe</name></author>
  <id>urn:uuid:60a76c80-d399-11d9-0003939e0af6</id>

  <entry>
    <title>Robot False Alarm</title>
    <link href="http://example.org/2003/12/14/norobots" />
    <id>urn:uuid:1225c695-cfb8-4ebb-80da344efa6a</id>
    <updated>2003-12-14T10:33:00Z</updated>
    <content>Sorry about the false alarm.</content>
    <link rel="edit" href="http://example.org/edit/2/" />
  </entry>
  ...
</feed>
```



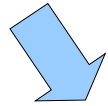
When I create a new entry in the collection, the server will add some elements to the entry.

The edit URI is the URI is one of those things. That URI is where the entry lives.

It's the URI by which we can retrieve, update or delete an entry.

Let's cover the simplest case first: delete.

Remove Entry



DELETE

<http://example.org/feed/2/>



The way to remove an entry from the collection feed is to send a DELETE to its edit URI.

You'll notice there's no body to our request, just the DELETE method.

This removes the entry from the collection feed.

Protocol Requirements

Create
Retrieve
Update
Delete



That's one more feature down, we have now covered creating, deleting and retrieving entries.

Actually when I marked Retrieve off the list as complete it wasn't really true.

The entry in the collection feed might not be complete. In order to retrieve the complete entry we need to GET the entry from the edit URI.

Edit URI

<http://example.org/feed/2/>



```
<entry xmlns="http://www.w3.org/2005/Atom">
  <title>Robot False Alarm</title>
  <link
    href="http://example.org/2003/12/14/norobots"/>
  <id>urn:uuid:1225c695-cfb8-4ebbbaaaa-a80d...</id>
  <updated>2003-12-14T10:33:00Z</updated>
  <content>Sorry about the false alarm.</content>
  <link rel="edit" href="http://example.org/edit/2/">
</entry>
```

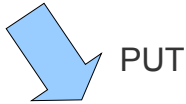


That entry is full. And if we want to make changes to an entry, that's the one to make changes to.

Once we've made those changes, how do we tell the server about those changes? We PUT the entry back to the edit URI. That is, we send an HTTP PUT request to the edit URI with the updated entry in the body of the request.

Update

```
<entry xmlns="http://www.w3.org/2005/Atom">
  <title>Atom Powered Robot False Alarm</title>
  <link
    href="http://example.org/2003/12/14/norobots"/>
  <id>urn:uuid:1225c695-cfb8-4ebbbaaa-80da344efa6a</id>
  <updated>2003-12-14T10:33:00Z</updated>
  <summary>Sorry about the false alarm.</summary>
  <link rel="edit" href="http://example.org/edit/2/">
</entry>
```



PUT

<http://example.org/feed/2/>



Here we've update the title of our entry and are PUTting it back to the edit URI.

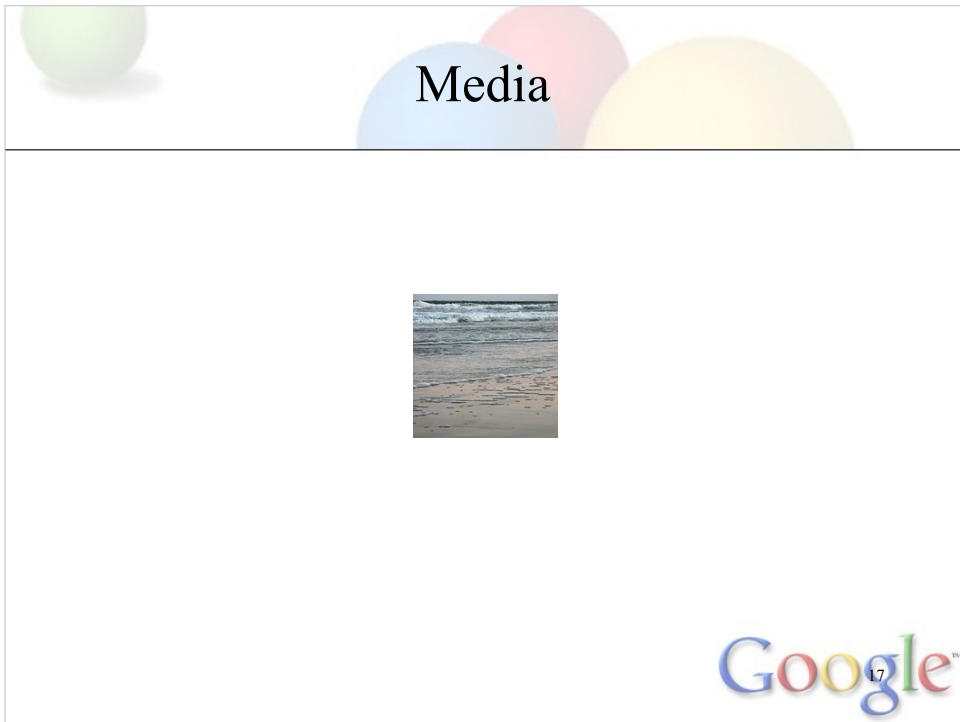


Protocol Requirements

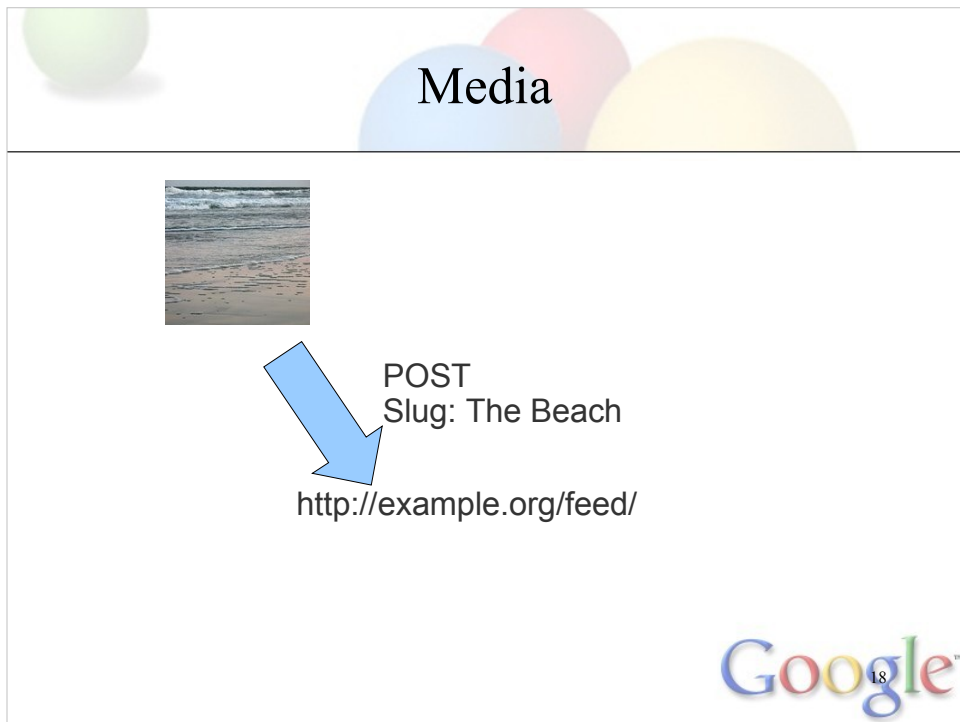
Create
Retrieve
Update
Delete



Now we're done, we've covered all four parts of the publishing protocol, well at least for text.



But what about media, such as images, or sound, or video? How does the Atom Publishing Protocol handle those?



Just like in the entry case, we do the simplest possible thing, we just POST the media to the URI of the collection feed.

Along with the POST method we also add in a Slug: header, which is used to give a hint to the server about how to build the URI of the created resources.

The POSTing of media to a collection actually creates TWO new resources.

Feed

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author><name>John Doe</name></author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>
  <entry xmlns="http://www.w3.org/2005/Atom">
    <title>The Beach</title>
    <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
    <updated>2005-10-07T17:17:08Z</updated>
    <summary type="text" />
    <content type="image/png"
      src="http://media.example.org/the_beach.png" />
    <link rel="edit-media"
      href="http://media.example.org/edit/the_beach.png" />
    <link rel="edit"
      href="http://example.org/media/edit/the_beach.atom" />
  </entry>
  ...
</feed>
```



we can see those two resources when we look at the entry that gets added to the collection.

Feed

```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author><name>John Doe</name></author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>
  <entry xmlns="http://www.w3.org/2005/Atom">
    <title>The Beach</title>
    <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
    <updated>2005-10-07T17:17:08Z</updated>
    <summary type="text" />
    <content type="image/png"
      src="http://media.example.org/the_beach.png" />
    <link rel="edit-media"
      href="http://media.example.org/edit/the_beach.png" />
    <link rel="edit"
      href="http://example.org/media/edit/the_beach.atom" />
  </entry>
  ...
</feed>
```



The 'edit' link points to one of the created resources
the Media Link Entry
contains the meta-data for the image

Feed

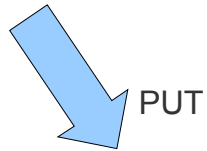
```
<feed xmlns="http://www.w3.org/2005/Atom">
  <title>Example Feed</title>
  <link href="http://example.org/" />
  <updated>2003-12-13T18:30:02Z</updated>
  <author><name>John Doe</name></author>
  <id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</id>
  <entry xmlns="http://www.w3.org/2005/Atom">
    <title>The Beach</title>
    <id>urn:uuid:1225c695-cfb8-4ebb-aaaa-80da344efa6a</id>
    <updated>2005-10-07T17:17:08Z</updated>
    <summary type="text" />
    <content type="image/png"
      src="http://media.example.org/the_beach.png" />
    <link rel="edit-media"
      href="http://media.example.org/edit/the_beach.png" />
    <link rel="edit"
      href="http://example.org/media/edit/the_beach.atom" />
  </entry>
  ...
</feed>
```



the “edit-media” link is the image

supports GET/PUT

Update Media



PUT

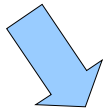
http://media.example.org/edit/the_beach.png



PUT an new image to update

Update Media Link Entry

```
<entry xmlns="http://www.w3.org/2005/Atom">
  <title>A Shark At The Beach</title>
  <id>urn:uuid:1225c695-cfb8-4ebb-80da344efa6a</id>
  <updated>2005-10-07T17:17:08Z</updated>
  <summary type="text" />
  <content type="image/png"
    src="http://media.example.org/the_beach.png"/>
  <link rel="edit-media"
    href="http://media.example.org/edit/the_beach.png" />
  <link rel="edit"
    href="http://example.org/media/edit/the_beach.atom" />
</entry>
```



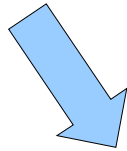
PUT

http://example.org/media/edit/the_beach.atom



PUT an updated MLE to update the metadata

Remove Media



DELETE

`http://example.org/media/edit/the_beach.atom`



you send a DELETE to the “edit” URI to remove both the meta-data and the media.



Corrections

More than one collection



don't want all the entries intermingled with the media, or maybe a link log, etc.
How to learn about all the feeds a service supports
And the media they support?

Service Document

```
<?xml version="1.0" encoding='utf-8'?>
<service xmlns="http://www.w3.org/2007/app"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <workspace>
    <atom:title>Main Site</atom:title>
    <collection
      href="http://example.org/blog/main" >
      <atom:title>My Blog Entries</atom:title>
    </collection>
    <collection
      href="http://example.org/blog/pic" >
      <atom:title>Pictures</atom:title>
      <accept>image/png</accept>
      <accept>image/jpeg</accept>
      <accept>image/gif</accept>
    </collection>
  </workspace>
</service>
```



Contains information about the collections a service supports

Service Document

```
<?xml version="1.0" encoding='utf-8'?>
<service xmlns="http://www.w3.org/2007/app"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <workspace>
    <atom:title>Main Site</atom:title>
    <collection
      href="http://example.org/blog/main" >
      <atom:title>My Blog Entries</atom:title>
    </collection>
    <collection
      href="http://example.org/blog/pic" >
      <atom:title>Pictures</atom:title>
      <accept>image/png</accept>
      <accept>image/jpeg</accept>
      <accept>image/gif</accept>
    </collection>
  </workspace>
</service>
```



workspace = blog
(or other grouping)

Service Document

```
<?xml version="1.0" encoding='utf-8'?>
<service xmlns="http://www.w3.org/2007/app"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <workspace>
    <atom:title>Main Site</atom:title>
    <collection
      href="http://example.org/blog/main" >
      <atom:title>My Blog Entries</atom:title>
    </collection>
    <collection
      href="http://example.org/blog/pic" >
      <atom:title>Pictures</atom:title>
      <accept>image/png</accept>
      <accept>image/jpeg</accept>
      <accept>image/gif</accept>
    </collection>
  </workspace>
</service>
```



collection == Collection

Service Document

```
<?xml version="1.0" encoding='utf-8'?>
<service xmlns="http://www.w3.org/2007/app"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <workspace>
    <atom:title>Main Site</atom:title>
    <collection
      href="http://example.org/blog/main" >
      <atom:title>My Blog Entries</atom:title>
    </collection>
    <collection
      href="http://example.org/blog/pic" >
      <atom:title>Pictures</atom:title>
      <accept>image/png</accept>
      <accept>image/jpeg</accept>
      <accept>image/gif</accept>
    </collection>
  </workspace>
</service>
```



the URI of the Collection Feed

Service Document

```
<?xml version="1.0" encoding='utf-8'?>
<service xmlns="http://www.w3.org/2007/app"
  xmlns:atom="http://www.w3.org/2005/Atom">
  <workspace>
    <atom:title>Main Site</atom:title>
    <collection
      href="http://example.org/blog/main" >
      <atom:title>My Blog Entries</atom:title>
    </collection>
    <collection
      href="http://example.org/blog/pic" >
      <atom:title>Pictures</atom:title>
      <accept>image/png</accept>
      <accept>image/jpeg</accept>
      <accept>image/gif</accept>
    </collection>
  </workspace>
</service>
```



This media collection accepts only these image types



The End

RFC 4287

RFC 5023

Jeff Fisher – Intro to Google Data on YouTube

<http://code.google.com/apis/gdata/>



The standards the define Atom Pub

The Google Data APIs are based on AtomPub

Watch Jeff's video

Visit site