

Google™



# Building Applications in the Cloud

Eric Bidelman & Anil Sabharwal  
Google

Sandosh Vasudevan  
gDocsBar

Ondrej Hrebicek  
Syncplicity

5/27/09

Post your questions for this talk on Google Moderator:  
<http://bit.ly/io-cloud-apps>





Welcome



## Brief API History - Then (~1yr ago)

- Read user's (private) list of documents
- Limited query model
  - **?title=AwesomeDoc&title-exact=true**
  - **/-/trashed, /-/starred**
  - **/-/document|spreadsheet|presentation**
- Upload documents, presentations, spreadsheets
- Trash documents

# Brief API History - Now

- Folder management
- ACLs @ user or domain level, **<gd:writersCanInvite>**
- Metadata: **<gd:lastViewedBy>**, **<gd:lastModifiedBy>**
- Download docs (export in multiple formats)
- Advanced query model:
  - who: **/-/shared-with-domain, /-/mine, /-viewed, owner, writer, reader**
  - what: **showfolders, showdeleted**
  - where: **/-/folder, /-/AnyFolderName, /-/hidden, /-/private**
  - when: **orderby, opened-min, opened-max**
- Replace document content. Append content (plaintext).
- Revision History
- ...

# Cloud Applications and API Use Cases

# Cloud Applications and API Use Cases

**Migrate** - *Upload documents to the cloud, download and export documents in bulk, access revisions.*

# Cloud Applications and API Use Cases

**Migrate** - *Upload documents to the cloud, download and export documents in bulk, access revisions.*

**Sync** - *Generate a feed / download of items that have recently changed.*



# Cloud Applications and API Use Cases

**Migrate** - *Upload documents to the cloud, download and export documents in bulk, access revisions.*

**Sync** - *Generate a feed / download of items that have recently changed.*

**Query** - *Access documents and metadata in an intuitive interface.*

# Cloud Applications and API Use Cases

**Migrate** - *Upload documents to the cloud, download and export documents in bulk, access revisions.*

**Sync** - *Generate a feed / download of items that have recently changed.*

**Query** - *Access documents and metadata in an intuitive interface.*

**Share** - *Manage sharing permissions (ACLs) and interoperate across different applications.*

# Cloud Applications and API Use Cases

**Migrate** - *Upload documents to the cloud, download and export documents in bulk, access revisions.*

**Sync** - *Generate a feed / download of items that have recently changed.*

**Query** - *Access documents and metadata in an intuitive interface.*

**Share** - *Manage sharing permissions (ACLs) and interoperate across different applications.*

**Discover** - *Search over metadata or content, find documents that meet a specific criteria, export as a zip file (2-Legged-OAuth).*



# Partner Demonstrations

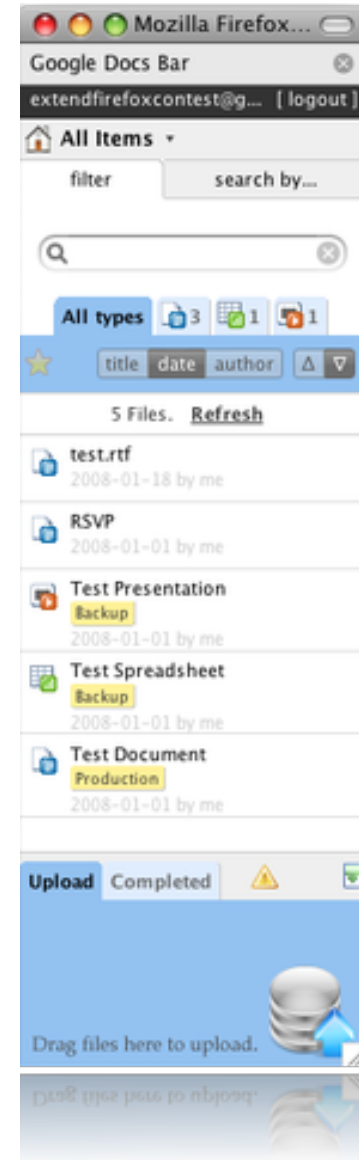


# DEMO



<http://www.gdocsbar.com/>

- How: JS, FF extension
- Demonstrates:
  - Full text queries
  - Title queries
  - Doc upload/export
  - Folders
- Goodies: Drag-n'-drop upload, great UI



# DEMO



<http://www.syncplicity.com/>

- How: Local client, .NET
- Demonstrates:
  - Rapid business collaboration through instantaneous sync
  - Lets MS Office and Google Docs users work on documents together
  - Bridges online and offline apps by syncing data amongst them



# The Protocol



# Google Data Protocol - crash course

- HTTP, REST (CRUD), Atom publishing protocol (XML), JSON, etc.
- Extends APP: authz for desktop/web, data model (`<gd:lastModifiedBy>`), batch operations), responses in alt formats (`rss`, `json`, `json-in-script`)





# Under the Hood

**GET** /feeds/documents/private/full/ HTTP/1.1

**Host:** docs.google.com

**Authorization:** AuthSub token="*SESSION\_TOKEN*"

# Under the Hood

**GET** /feeds/documents/private/full/ HTTP/1.1  
**Host:** docs.google.com  
**Authorization:** AuthSub token="*SESSION\_TOKEN*"

**HTTP/1.1 200 OK**

**Content-Type:** application/atom+xml; charset=UTF-8

```
<feed xmlns='http://www.w3.org/2005/Atom'  
  xmlns:gd='http://schemas.google.com/g/2005'>  
  <title>Available Documents - john.doe@gmail.com</title>  
  <updated>...</updated>  
  <entry>  
    <id>http://docs.google.com/feeds/.../document%3Aabc123</id>  
    ...  
    <title type='text'>Document Title</title>  
    <category scheme="..." term="..." label="document"/>  
    <content type="text/html" src="..." />  
    <link rel="alternate" type="text/html" href="..." />  
    <author>...</author>  
    <gd:resourceId>document:abc123</gd:resourceId>  
    <gd:lastViewed>2009-04-09T17:13:17.453Z</gd:lastViewed>  
    ...  
  </entry>  
  <entry>...</entry>  
</feed>
```

# Client Libraries



- Provide high level APIs to the protocol (better than XML!)
- Sample applications - <http://tr.im/jQvc>



# 1. Migrate



# Migrate

- Document/folder creation:
  - from Atom XML (metadata)
  - from media content (mime multi-part)
  - both

**POST** /feeds/documents/private/full/ HTTP/1.1

**Host:** http://docs.google.com

**Content-Type:** application/atom+xml

**Content-Length:** 288

**Authorization:** AuthSub token="*SESSION\_TOKEN*"

```
<entry xmlns="http://www.w3.org/2005/Atom">  
  <category scheme="http://schemas.google.com/g/2005#kind"  
    term="http://schemas.google.com/docs/2007#folder"  
    label="folder"/>  
  <title>Example Folder</title>  
</entry>
```

# Migrate - Uploading Content

**POST** /feeds/documents/private/full/ HTTP/1.1

**Host:** http://docs.google.com

**Authorization:** <your authorization header here>

**Content-Length:** 73612

**Content-Type:** multipart/related; boundary=**END\_OF\_PART**

**Slug:** test.doc

**--END\_OF\_PART**

**Content-Type:** application/atom+xml

```
<entry xmlns="http://www.w3.org/2005/Atom">
  <category scheme="http://schemas.google.com/g/2005#kind
    term="http://schemas.google.com/docs/2007#document"
    label="document"/>
  <title>example document</title>
</entry>
```

**--END\_OF\_PART**

**Content-Type:** application/msword

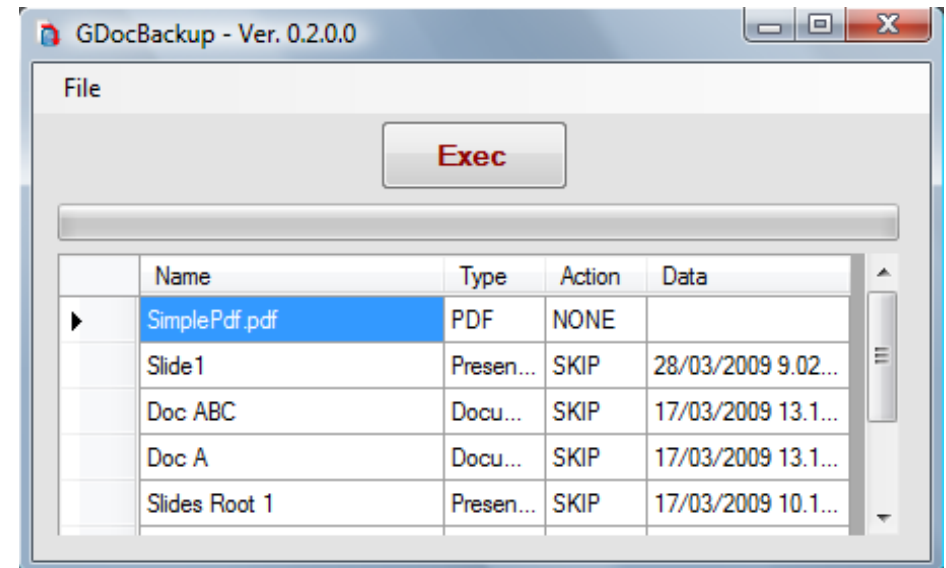
... doc contents here ...

**--END\_OF\_PART--**

# DEMO - GDocsBackup

<http://gs.fhtino.it/gdocbackup>

- How: C#, .NET library, open-source: <http://bit.ly/gdocbackup>
- Demonstrates:
  - Bulk export
  - Retains folder structure
  - 'Sync' based on last updated timestamp



# Migrate - How Backup is done...

- Client sends authenticated request to export servlet

## Documents & Presentations

**GET** /feeds/download/documents/Export ?  
docID=doc\_id&exportFormat=format

## Spreadsheets

**GET** /feeds/download/spreadsheets/Export ?  
key=doc\_id&fmcmd=format

Document formats: doc, html, odt, pdf, png, rtf, txt, zip

Presentation formats: pdf, png, ppt, swf, txt

Spreadsheet formats: xls (4), csv (5), pdf (12), ods (13), tsv (23), html (102)





## 2. Sync



# Sync - Python client library: simple example

- Imagine a cron job for backup:

```
0 21 * * * python path/to/simple_backup.py 2>&1
```

- Use **updated-min** / **updated-max** query parameters for basic syncing.

# Sync - Python client library: simple example

- Imagine a cron job for backup:

```
0 21 * * * python path/to/simple_backup.py 2>&1
```


- Use **updated-min** / **updated-max** query parameters for basic syncing.

```
import gdata.docs.service as gdocs


client = gdocs.DocsService(source='yourCo-AppName-v1')
client.ClientLogin('user@example.com', 'pa$$word')

ms = gdata.MediaSource(file_path='path/to/log.txt',
                       content_type='text/plain')
entry = client.UploadDocument(ms, 'TitleForDocument')

print '%s is online at: %s' % (entry.title.text,
                              entry.GetAlternateLink().href)
```



## 3. Query



# Query

- Supports standard query model (**q, title, author, update-min|max, start-index**)
- Additional parameters for metadata and content discovery
- Advanced category syntax

## Advanced queries:

**<http://docs.google.com/feeds/document/private/full>**

**GET** /feeds/.../full?updated-min=2009-05-20&orderby=title

**GET** /feeds/.../full/-/spreadsheet|document/-viewed

**GET** /feeds/.../full/-/mine/pdf/shared-with-domain

**GET** /feeds/.../full/-/document?showfolders=true

**GET** /feeds/.../full?writer=user1@gmail.com|user2@example.com&alt=json

**GET** /feeds/.../full?opened-max=2009-05-20&q=Needle+in+the+haystack

# Query

```
<entry>
  <id>http://docs.google.com/feeds/.../document%3Aabc123</id>
  <updated>2009-04-09T19:35:57.330Z</updated>
  <published>2008-08-15T17:32:33.142Z</published>
  <title type='text'>Document Title</title>
  <category scheme="http://schemas.google.com/g/2005/labels" term="..."
    label="viewed"/>
  <category scheme="http://schemas.google.com/g/2005#kind" term="..."
    label="document"/>
  <category scheme="..." term="ChildFolder" label="ChildFolder"/>
  <content type="text/html" src="...">
  <link rel="http://schemas.google.com/docs/2007#parent"
    type="application/atom+xml" href="..." title="ParentFolder"/>
  <link rel="alternate" type="text/html" href="...">
  <link rel="self" type="application/atom+xml" href="...">
  <link rel="edit" type="application/atom+xml" href="...">
  <link rel="edit-media" type="text/html" href="...">
  <author>
    <name>John Doe</name>
    <email>john.doe@gmail.com</email>
  </author>
  <gd:resourceId>document:abc123</gd:resourceId>
  <gd:lastModifiedBy>
    <name>John Doe Sr.</name>
    <email>pops@gmail.com</email>
  </gd:lastModifiedBy>
  <gd:lastViewed>2009-04-09T17:13:17.453Z</gd:lastViewed>
  <docs:writersCanInvite value="false"/>
  <gd:feedLink rel="..." href="...">
</entry>
```

# Query

```
<entry>
```

```
  <id>http://docs.google.com/feeds/.../document%3Aabc123</id>
```

```
<category
```

```
  scheme="http://schemas.google.com/g/2005/labels"
```

```
  term="..." label="viewed"/>
```

```
<category
```

```
  scheme="http://schemas.google.com/g/2005#kind"
```

```
  term="..." label="document"/>
```

```
<category scheme="..." term="ChildFolder"
```

```
  label="ChildFolder"/>
```

```
</author>
```

```
<gd:resourceId>document:abc123</gd:resourceId>
```

```
<gd:lastModifiedBy>
```

```
  <name>John Doe Sr.</name>
```

```
  <email>pops@gmail.com</email>
```

```
</gd:lastModifiedBy>
```

```
<gd:lastViewed>2009-04-09T17:13:17.453Z</gd:lastViewed>
```

```
<docs:writersCanInvite value="false"/>
```

```
<gd:feedLink rel="..." href="..."/>
```

```
</entry>
```

# Query

```
<entry>  
  <id>http://docs.google.com/feeds/.../document%3Aabc123</id>  
  <updated>2009-04-09T19:35:57.330Z</updated>  
  <published>2008-08-15T17:32:33.142Z</published>  
  <title type='text'>Document Title</title>  
  <category scheme="http://schemas.google.com/g/2005/labels" term="..."  
    label="viewed"/>
```

```
<link rel="alternate" type="text/html"  
  href="..." />
```

```
<link rel="self" type="application/atom+xml"  
  href="..." />
```

```
<link rel="edit" type="application/atom+xml"  
  href="..." />
```

```
<link rel="edit-media" type="text/html"  
  href="..." />
```

```
<name>John Doe Sr.</name>  
<email>pops@gmail.com</email>  
</gd:lastModifiedBy>  
<gd:lastViewed>2009-04-09T17:13:17.453Z</gd:lastViewed>  
<docs:writersCanInvite value="false"/>  
<gd:feedLink rel="..." href="..." />  
</entry>
```



# Query

```
<entry>
  <id>http://docs.google.com/feeds/.../document%3Aabc123</id>
  <updated>2009-04-09T19:35:57.330Z</updated>
  <published>2008-08-15T17:32:33.142Z</published>
  <title type='text'>Document Title</title>
  <category scheme="http://schemas.google.com/g/2005/labels" term="..."
    label="viewed"/>
  <category scheme="http://schemas.google.com/g/2005#kind" term="..."
    label="document"/>
  <category scheme="..." term="ChildFolder" label="ChildFolder"/>
  <content type="text/html" src="...">
  <link rel="http://schemas.google.com/docs/2007#parent"
    type="application/atom+xml" href="..." title="ParentFolder"/>
  <link rel="alternate" type="text/html" href="...">
  <link rel="self" type="application/atom+xml" href="...">
  <link rel="edit" type="application/atom+xml" href="...">
  <gd:resourceId>document:abc123</gd:resourceId>
  <gd:lastModifiedBy>
    <name>John Doe Sr.</name>
    <email>pops@gmail.com</email>
  </gd:lastModifiedBy>
  <gd:lastViewed>2009-04-09T17:13:17.453Z</gd:lastViewed>
  <docs:writersCanInvite value="false"/>
  <gd:feedLink rel="..." href="...">
    <docs:writersCanInvite value="false"/>
    <gd:feedLink rel="..." href="...">
</entry>
```

# DEMO - DocList Manager

<http://doclistmanager.googlecodesamples.com/>

- How: RoR, open-source

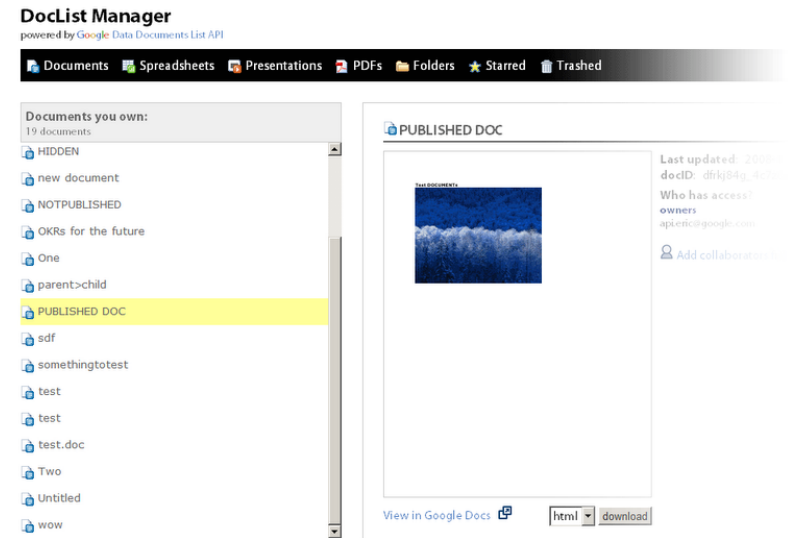
[http://bit.ly/ror\\_sample](http://bit.ly/ror_sample)

- Uses Ruby util lib, open-source

<http://bit.ly/gdata-ruby-util>

- Demonstrates:

- Category queries
- Export .png as thumbnail preview...
- Mashup with Contacts API to add ACLs...



# Google Spreadsheet APIs



- **Table and Records feeds** *(New Data API!)*
  - Programmatically define tables within a sheet
  - Record feed defined from the Table feed
  - Consistent data store (row IDs and column meta data)
- **Spreadsheet gadgets API**
  - Uses the gadgets and google visualization APIs

Spreadsheets API Office Hours: 2pm - 3pm



## 4. Share (ACLs)



# Share - ACLs

- Each document **<entry>** contains a link to it's ACL feed

```
<gd:feedLink rel="..."
```

```
href=".../feeds/acl/private/full/spreadsheet%3Aabc123"/>
```

- Allows
  - roles: owner, writer (collaborator), reader
  - share with 'everyone' in domain (Google Apps)
  - batch support
  - **writersCanInvite** flag on doc upload / update

# Share - ACLs

- Each document `<entry>` contains a link to it's ACL feed

```
<gd:feedLink rel="..."  
href=".../feeds/acl/private/full/spreadsheet%3Aabc123"/>
```

- Allows
  - roles: owner, writer (collaborator), reader
  - share with 'everyone' in domain (Google Apps)
  - batch support
  - **writersCanInvite** flag on doc upload / update
- **TIP:** ACLs can be inlined into doc list feed

**GET** /feeds/documents/private/expandAcl

# Share - Example ACL Entry

**GET** /feeds/acl/private/full/spreadsheet%3Aabc123

<entry>

...

<category scheme='http://schemas.google.com/g/2005#kind'  
term='http://schemas.google.com/acl/2007#accessRule' />

<title>Document Permission - a.reader@example.com</title>

<link rel='self' type='application/atom+xml' href='...' />

<link rel='edit' type='application/atom+xml' href='...' />

<gAcl:role value='reader' />

<gAcl:scope type='user'  
value='a.reader@example.com' />

</entry>

# Share - Example ACL Entry

**GET** /feeds/acl/private/full/spreadsheet%3Aabc123

<entry>

...

```
<category scheme='http://schemas.google.com/g/2005#kind'  
  term='http://schemas.google.com/acl/2007#accessRule' />  
<title>Document Permission - a.reader@example.com</title>  
<link rel='self' type='application/atom+xml href='...' />  
<link rel='edit' type='application/atom+xml href='...' />  
<gAcl:role value='reader' />  
<gAcl:scope type='user'  
  value='a.reader@example.com' />
```

</entry>

Discovery if document has been shared outside of your domain:

```
<gAcl:scope type='everyone' value='otherguys.com' />
```





## 5. Discover



# Discover - 2 Legged OAuth

- Possible with the 2 legged OAuth
  - Google Apps Premier & Education editions
  - No user intervention!
  - Supported by all of the Google Data APIs (in gApps)

Dashboard Users and groups Domain settings **Advanced tools** Support Service settings

[« Back to Advanced tools](#)

## Manage OAuth Access

OAuth consumer key: **appstraining0.com**

---

OAuth consumer secret: **[REDACTED]**

---

**X.509 certificate:** We do not have a certificate for your domain.

Upload a certificate: (Optional)

File must be in PEM format. [Learn More](#)

---

**Two-legged OAuth**  Enable Two-legged OAuth

If two-legged OAuth is enabled, administrators at this domain can use the consumer secret :

---

# Discover - 2 Legged OAuth

- Possible with the 2 legged OAuth
  - Google Apps Premier & Education editions
  - No user intervention!
  - Supported by all of the Google Data APIs (in gApps)
- Other examples possible w/ 2 legged OAuth:
  1. Modify ACL permissions of a user's document
  2. See if user has read 'Company Policies' document
  3. Add/update standard set of company's important contacts
  4. Set weekly calendar reminder for users

# DEMO - Legal Discovery Tool

Main Audit Log Control Panel

## Document Export

Has the words:

Named:

Type: All types

Search: All items

Sharing: Any

Shared with:

Collaborators:

Owner is:  e.g. johndoe@google.com

Date: Modified

from:  x to:  x

Custodian:

Case Id

## Request Status

Case	Key	Requestor	Custodian	Query Detail	Document Count	Time	Status	Zip
22	ftv51wy5_2wczts5_qu8c98r2bp9h7bher5r6ek8llc	anilsa@google.com	spambox@meeviews.com	<input type="checkbox"/> Show Detail	6	Apr 23, 2009 05:34 PM GMT+10:00	FINISHED	<a href="#">Zip File</a>

- How: AppEngine Java [demo](#)
- Demonstrates:
  - 2 Legged OAuth
  - Advanced Search
  - Export as ZIP

# Discover - How It's Done

```
import com.google.gdata.client.docs.*;
import com.google.gdata.client.authn.oauth.*;

GoogleOAuthParameters oauthParameters = new GoogleOAuthParameters();
oauthParameters.setOAuthConsumerKey("example.com");
oauthParameters.setOAuthConsumerSecret("YOUR_CONSUMER_SECRET");

DocsService client = new DocsService("yourCompany-YourAppName-v1");
client.setOAuthCredentials(oauthParameters,
    new OAuthHmacSha1Signer());

String user = "any.user@example.com";
URL feedUrl =
    new URL("http://docs.google.com/feeds/documents/private/full" +
        "?xoauth_requestor_id=" + user);

DocumentListFeed resultFeed = client.getFeed(feedUrl,
    DocumentListFeed.class);

for (DocumentListEntry entry : resultFeed.getEntries()) {
    System.out.println(entry.getTitle().getPlainText());
}
```

# DEMO



<http://www.offisync.com/>

- How: Microsoft Office Extension, .NET
- Demonstrates:
  - Open and Save
  - Sharing
  - Search
  - MUCH more!





Wrap Up



# Conclusion

## Documentation Resources:

Documents List API

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

2 Legged OAuth

<http://tr.im/kfAX>

Code samples:

<http://code.google.com/apis/documents/code.html>

Developer forum

<http://groups.google.com/group/Google-Docs-Data-APIs>



## Q & A

Post your [questions](#) for this talk on Google Moderator:  
<http://bit.ly/io-cloud-apps>

Google™

