

Google™



Launch your app inside of Google Apps with gadgets

Dan Holevoet
May 20, 2010

Got questions?

**We're using Wave and Google Moderator:
<http://bit.ly/diq3qc>**

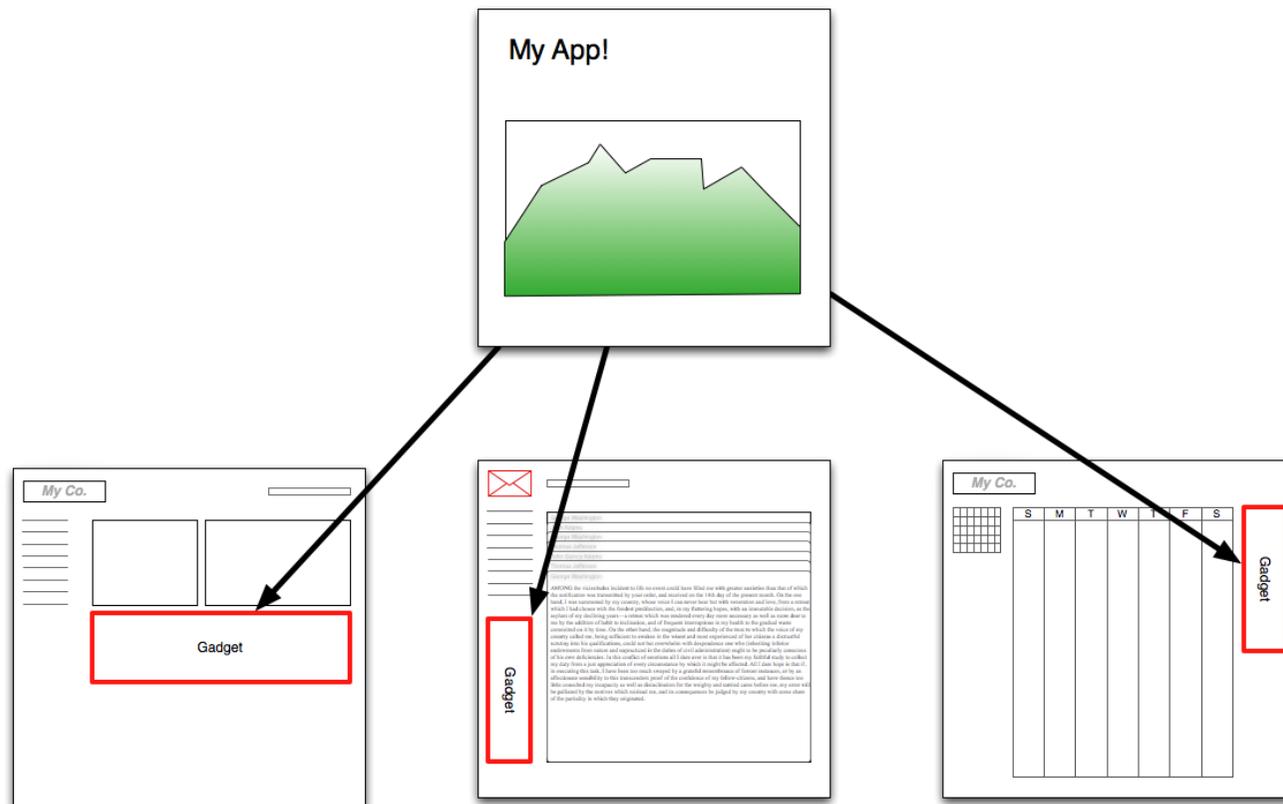
Agenda

- Why gadgets?
- What are gadgets?
- Where do gadgets work?
- Common tasks and best practices
- How do I distribute gadgets?
- An example application
- How do I get started?

Why gadgets?

Why gadgets?

- Gadgets bring your application into Google Apps
- People spend *a lot* of time in Google Apps
- Embeddable pieces of functionality



What are gadgets?

What are gadgets?

XML, HTML, and JavaScript

```
<?xml version="1.0" encoding="UTF-8" ?>
<Module>
  <ModulePrefs title="hello world example" />
  <Content type="html">
    <![CDATA[
      Hello, world!
    ]]>
  </Content>
</Module>
```

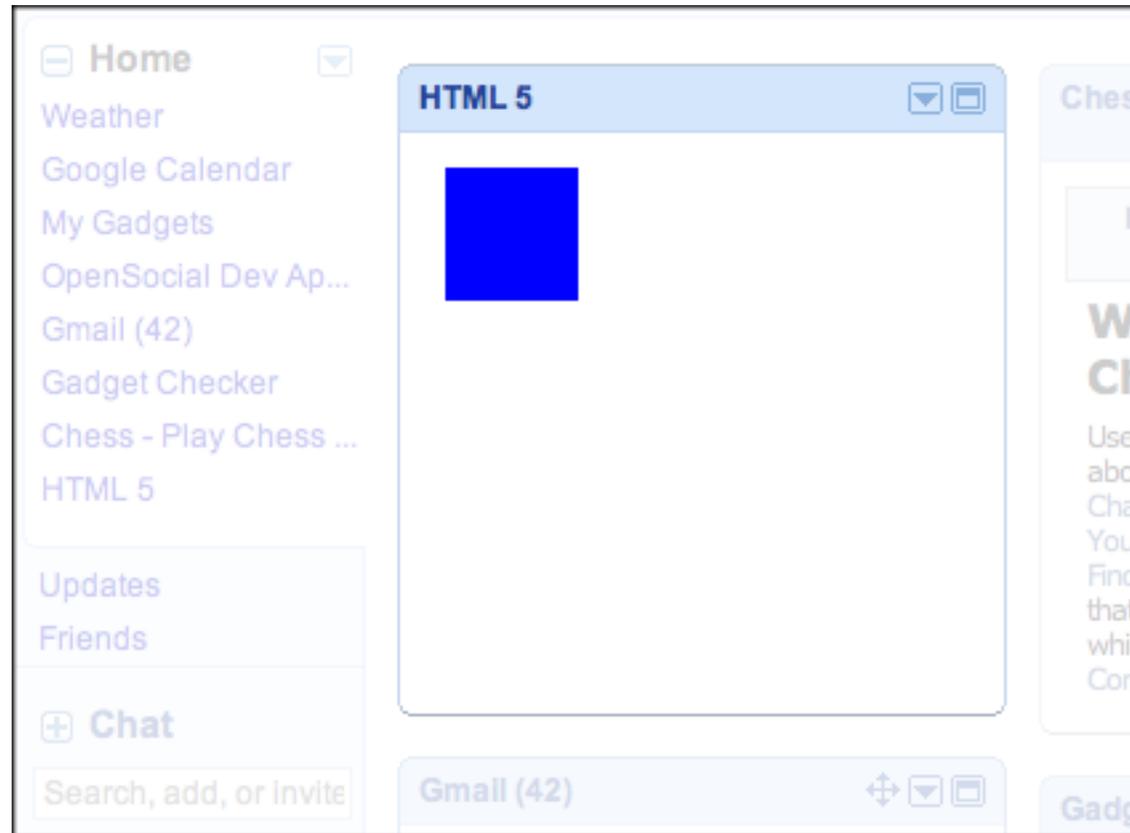
What are gadgets?

HTML 5, too!

```
<Content type="html" >
<![CDATA[
  <!DOCTYPE HTML>
  <body onload="drawSquare();">
  <script type="application/javascript">
    function drawSquare() {
      var canvas = $('#canvas');
      var ctx = canvas.getContext("2d");
      ctx.fillStyle = "rgb(0,0,255)";
      ctx.fillRect (0, 0, 50, 50);
    }
  </script>
  <canvas id="canvas" width="80" height="80"/>
]]>
</Content>
```

What are gadgets?

HTML 5, too!



What are gadgets?

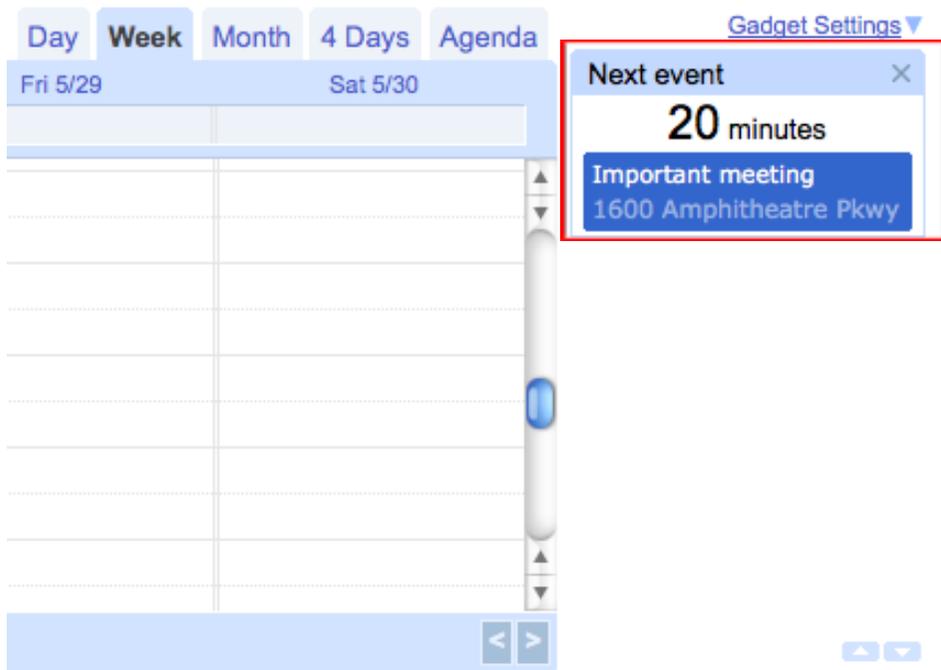
Gadget-specific features

- 'dynamic-height' allows the gadget to grow and shrink depending on content
- 'views' allow multiple facets of an application
 - 'home' and 'canvas' views in most implementations
- osapi.http allows remote data access
 - Provides OAuth signing
 - Supports multiple HTTP methods
 - Supports JSON parsing of responses

Where do gadgets work?

Where do gadgets work?

Calendar

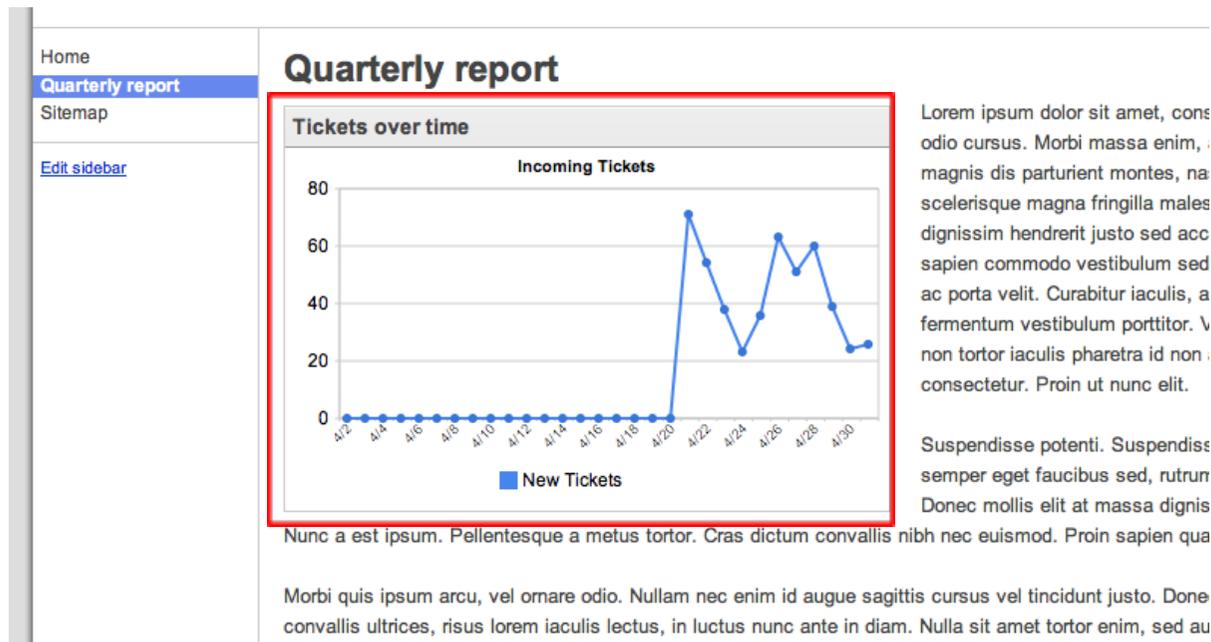


- Calendar right-nav
- Provides 'nav' and 'canvas' views
- google.calendar APIs

Where do gadgets work?

Sites

- Embeddable
- Provides 'profile' and 'configuration' views
- Best way to embed HTML/JS in Sites
- Provide dynamic content instead of just images



Where do gadgets work?

Gmail

[Compose Mail](#)

[Inbox \(15\)](#)

[Buzz \(1\)](#)

[Sent Mail](#)

[Drafts](#)

[Spam](#)

[family \(4\)](#)

[9 more ▾](#)

[Contacts](#)

[Tasks](#)

+ Daniel Holevoet

Search, add, or invite

+ Web Search

- Google Calendar

May 2010						
S	M	T	W	T	F	S
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

- Gmail left-nav
- Provides 'nav' and 'canvas' views

This is a gadget, too!

Where do gadgets work?

Gmail (contextual)

- Match on email fields and contents
- Provides matched data to a gadget displayed at the bottom of the email

Important document Inbox | X

★  John Doe

https://docs.google.com/Doc?id=d99p6wv_5gm5fh499

Google Docs - Items from this e-mail



Important document

[Show preview](#)

[Open in new window](#)

[↩ Reply](#)

[→ Forward](#)

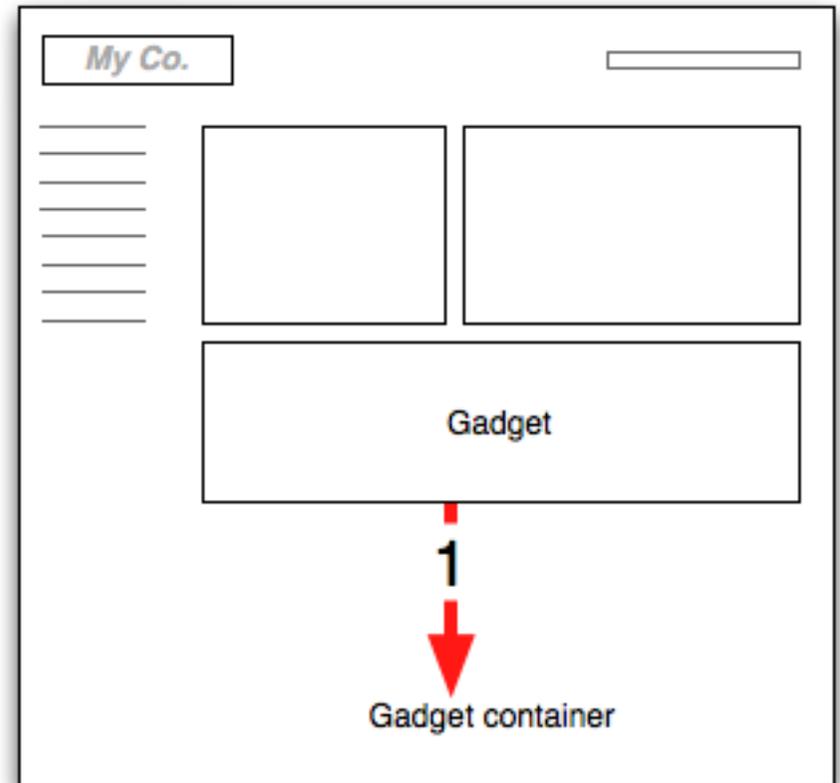
Common tasks and best practices

Common tasks and best practices

Fetching data

1. Gadget sends content request to parent page (container) using `osapi.http.get` (or `.post`)

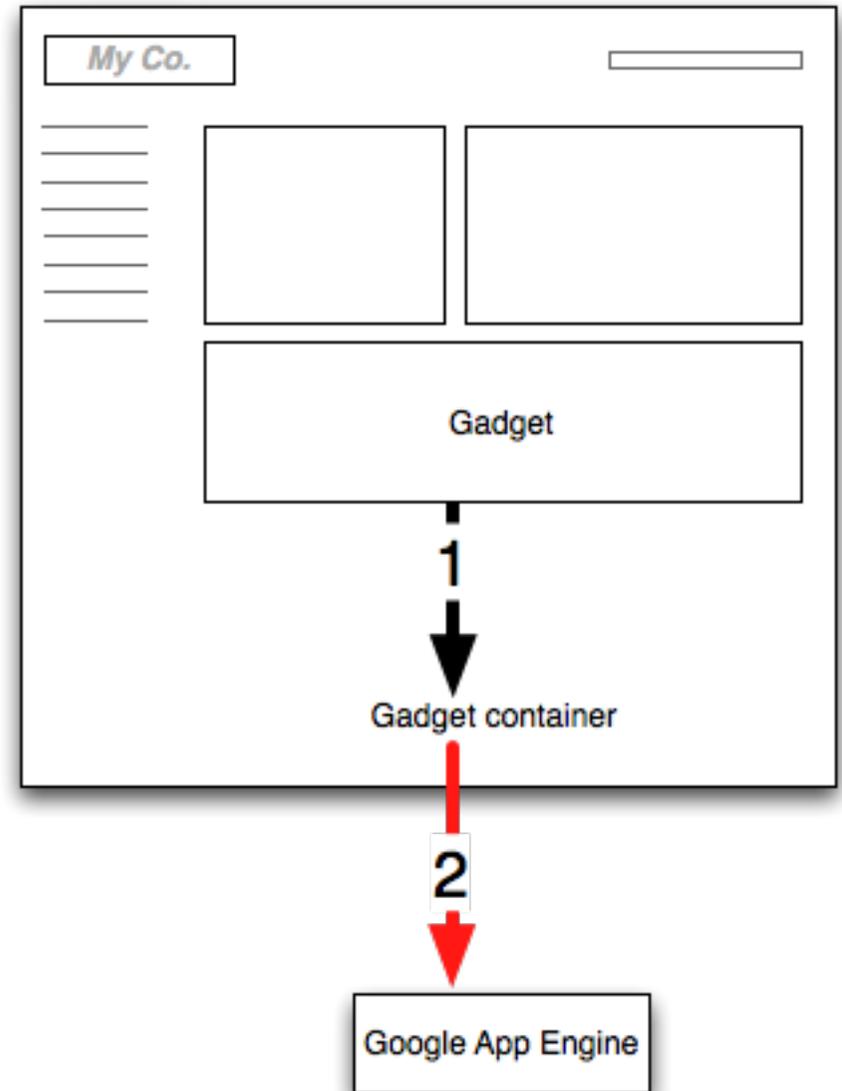
```
osapi.http.get({  
  'href': 'http://example.com',  
  'format': 'json',  
  'authz': 'signed'  
}).execute(handleResponse);
```



Common tasks and best practices

Fetching data

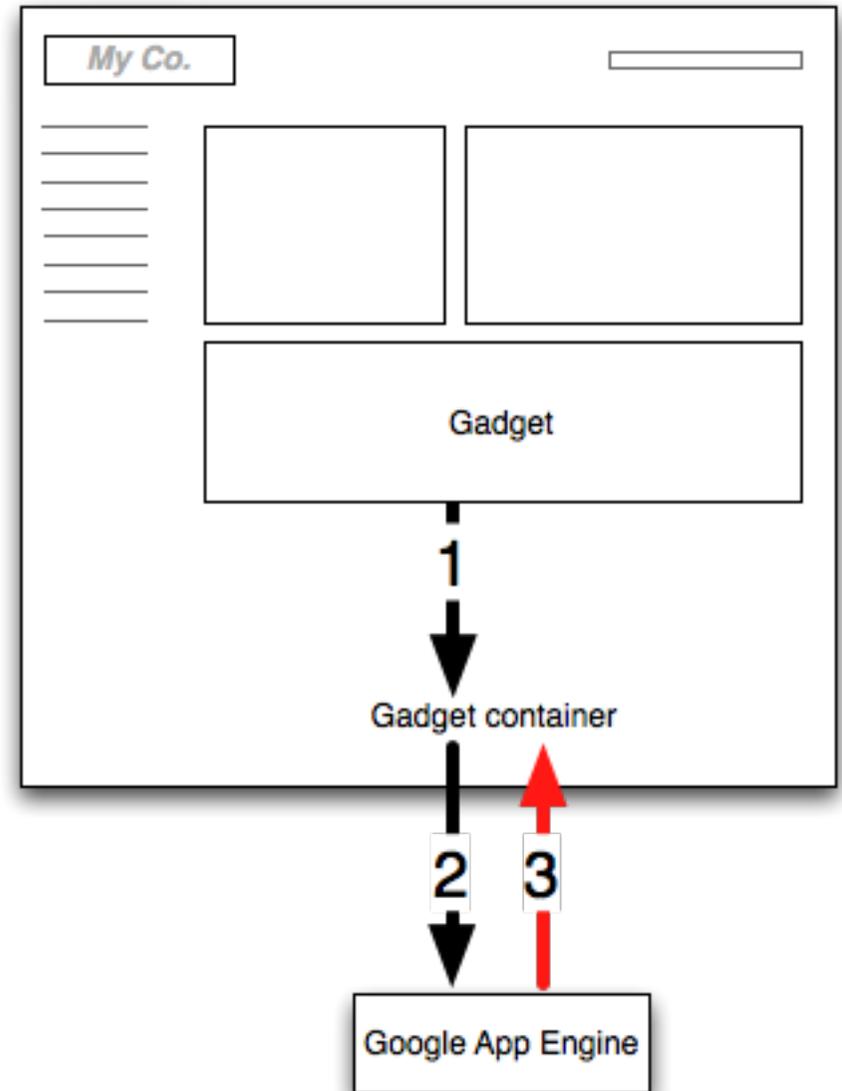
1. Gadget sends content request to parent page (container) using `osapi.http.get` (or `.post`)
2. Container proxies request to remote server



Common tasks and best practices

Fetching data

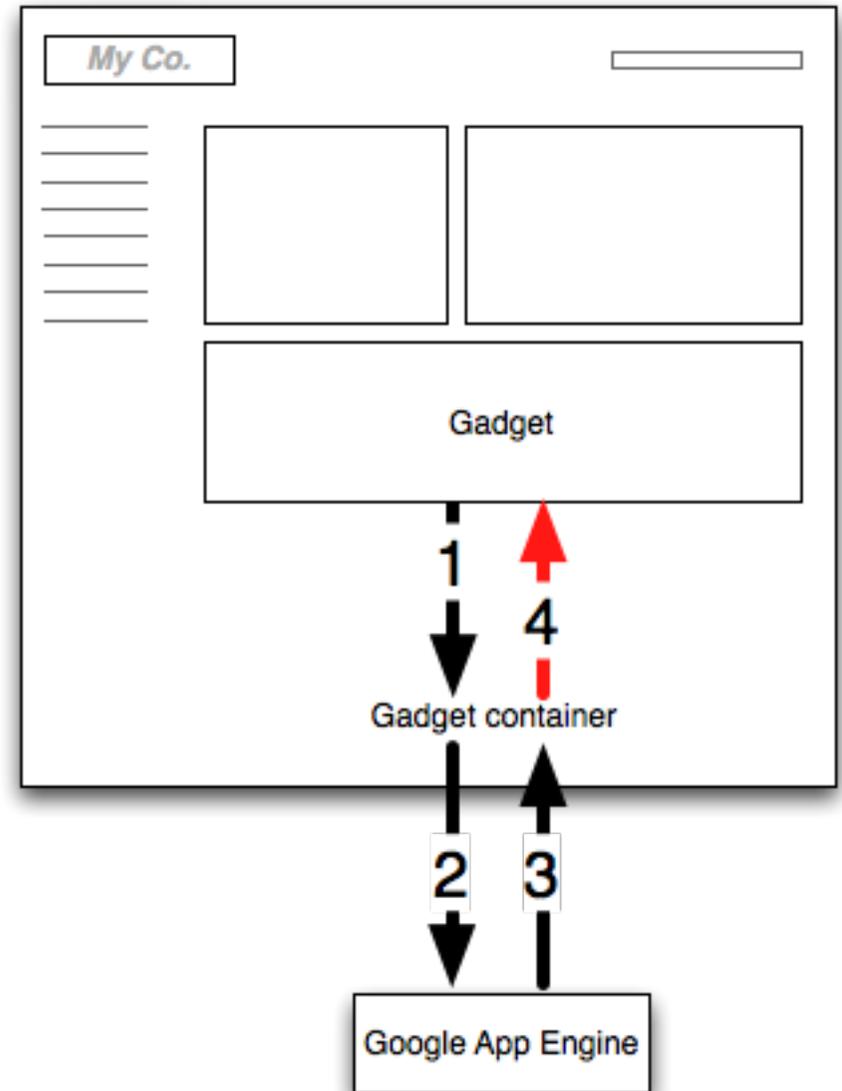
1. Gadget sends content request to parent page (container) using `osapi.http.get` (or `.post`)
2. Container proxies request to remote server
3. Server receives request and returns data



Common tasks and best practices

Fetching data

1. Gadget sends content request to parent page (container) using `osapi.http.get` (or `.post`)
2. Container proxies request to remote server
3. Server receives request and returns data
4. Data is passed to the gadget

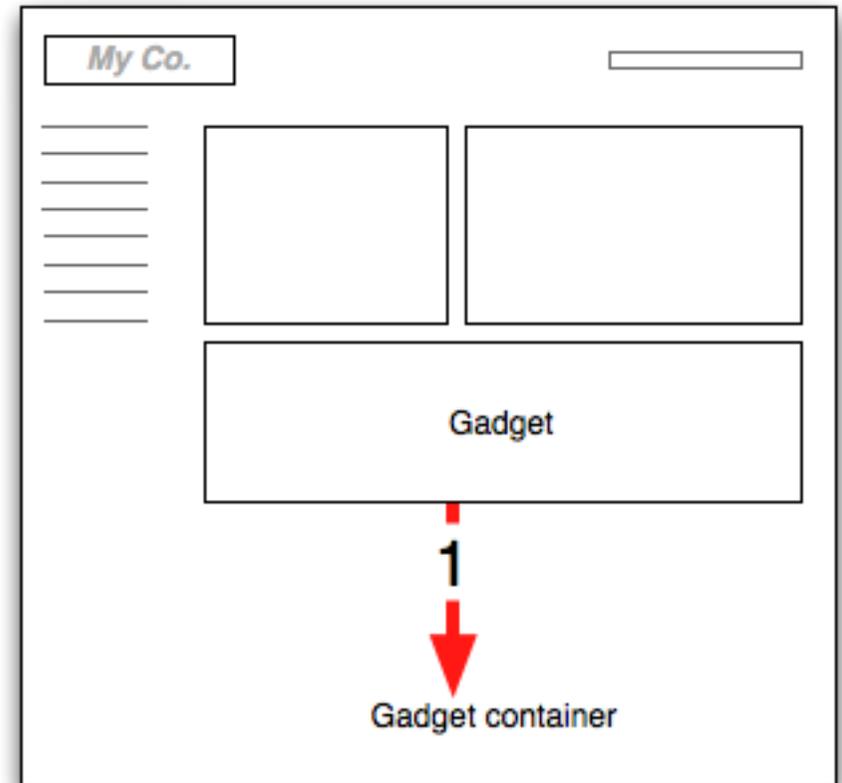


Common tasks and best practices

Authentication

1. Gadget sends content request to parent page (container) using `osapi.http.get` (or `.post`)

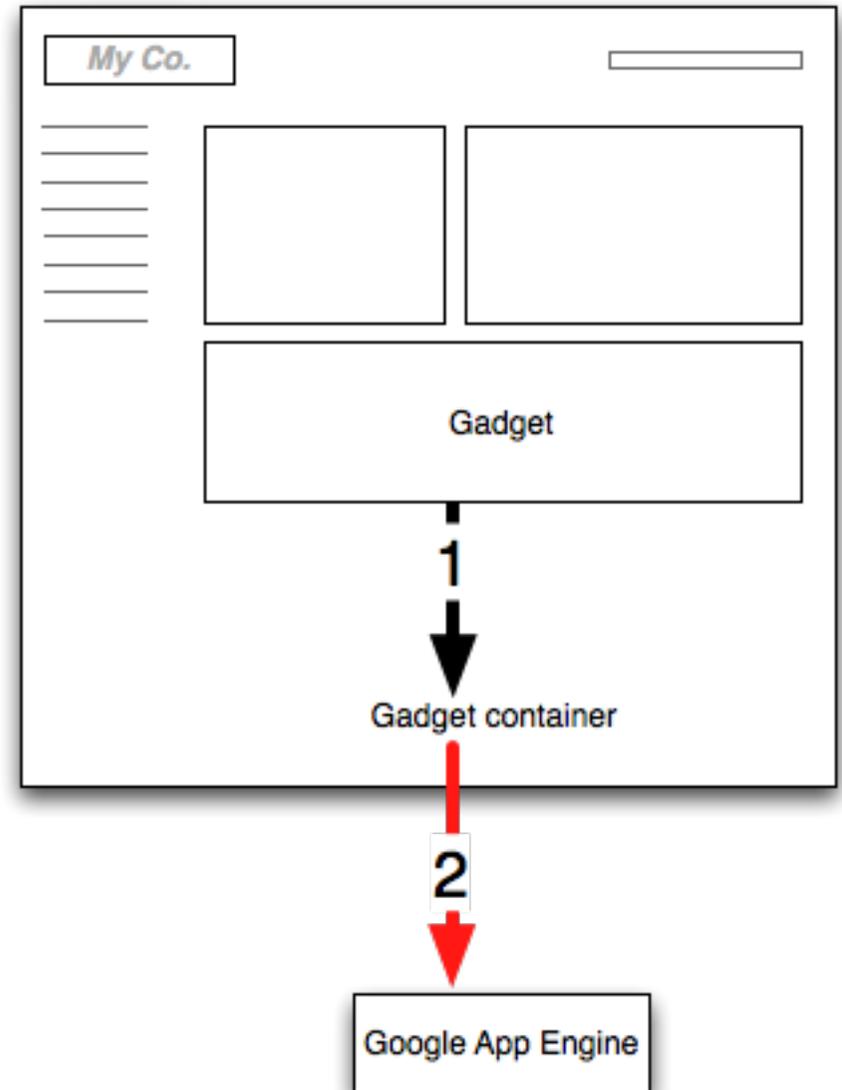
```
osapi.http.get({  
  'href': 'http://example.com',  
  'format': 'json',  
  'authz': 'signed'  
}).execute(handleResponse);
```



Common tasks and best practices

Authentication

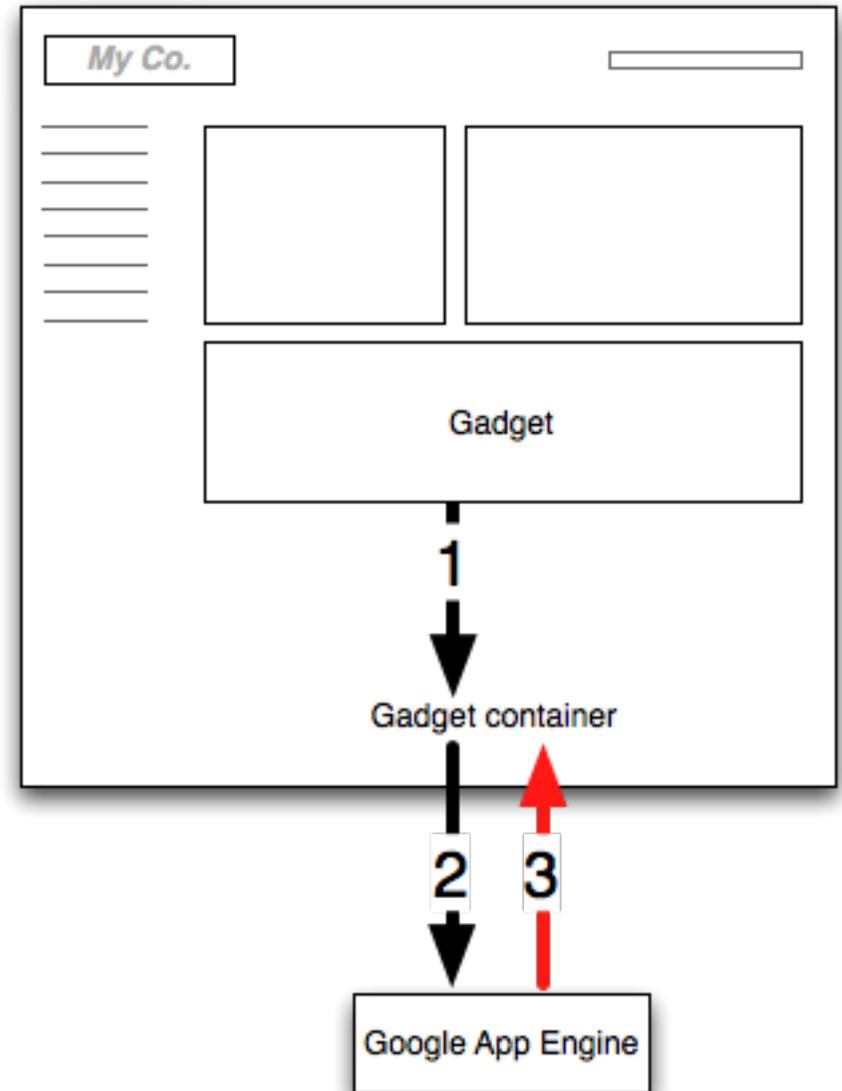
1. Gadget sends content request to parent page (container) using `osapi.http.get` (or `.post`)
2. Container appends unique user ID (OpenSocial ID) and signs request, before transmitting to remote server



Common tasks and best practices

Authentication

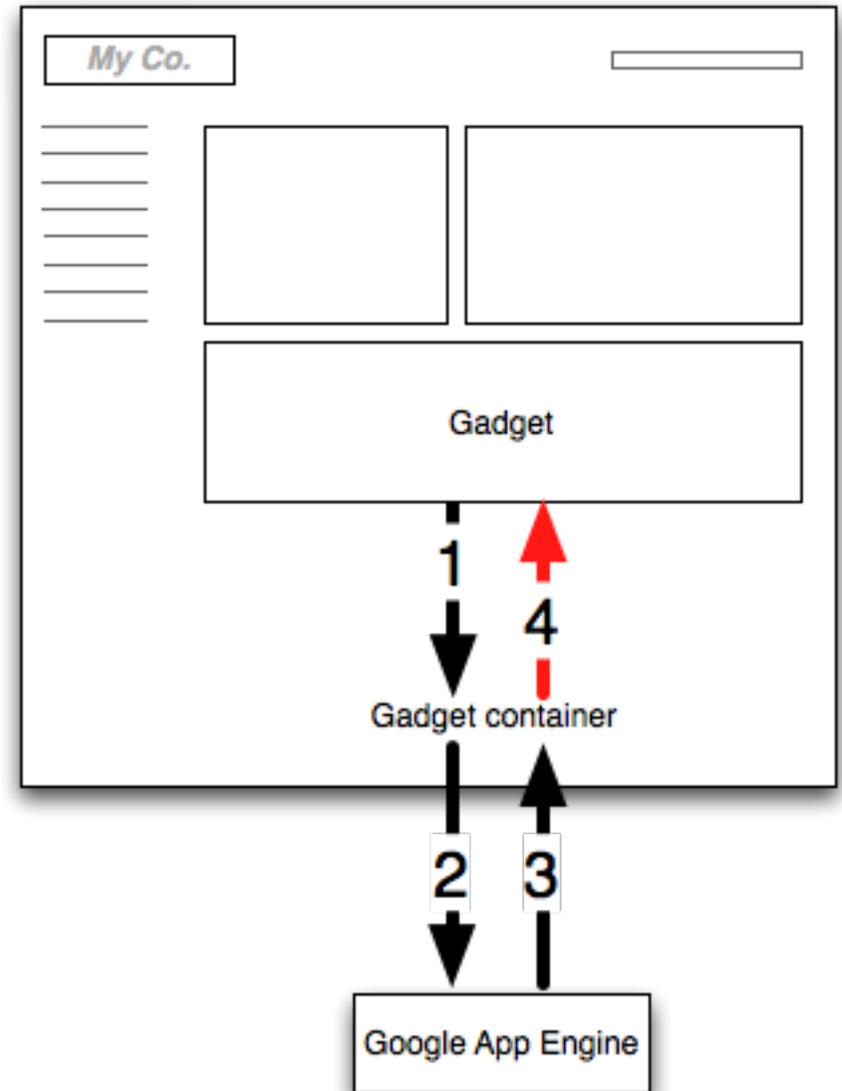
1. Gadget sends content request to parent page (container) using `osapi.http.get` (or `.post`)
2. Container appends unique user ID (OpenSocial ID) and signs request, before transmitting to remote server
3. Server receives request and returns data



Common tasks and best practices

Authentication

1. Gadget sends content request to parent page (container) using `osapi.http.get` (or `.post`)
2. Container appends unique user ID (OpenSocial ID) and signs request, before transmitting to remote server
3. Server receives request and returns data
4. Data is passed to the gadget



Common tasks and best practices

What if I need more than an ID?

- Existing applications can add an OpenSocial ID to their 'users' table
- When signed requests come in, see if that OpenSocial ID exists in the database — if it does, you're all set
- When the ID doesn't exist, create a one-time use token, and return that to the gadget
- Have the gadget open a window to your server (append the token to the request!) and log in, using OpenID — save the OpenSocial ID stored with that token in the database

Common tasks and best practices

Authentication (in code)

```
osapi.http.get({  
  'href': 'http://example.com/json',  
  'format': 'json',  
  'authz': 'signed'  
}).execute(handleResponse);
```

Common tasks and best practices

Authentication (in code)

```
user = Users.gql('WHERE opensocial_id = :1',  
    self.request.get('viewer_id')).get()
```

if not user:

```
    temp = TempUser(opensocial_id =  
        self.request.get('viewer_id'),  
        token = session_token)
```

```
    return {'status': 'not found',  
        'token': session_token}
```

else:

```
    return {'status': 'found'}
```

Common tasks and best practices

Authentication (in code)

```
var popup = window.open(openid_url, 'OpenID',  
    'width=300,height=300');
```

```
finishedInterval = setInterval(function() {  
    if (popup.closed) {  
        osapi.http.get({  
            'href': 'http://example.com/json',  
            'format': 'json',  
            'authz': 'signed'  
        }).execute(handleResponse);
```

```
clearInterval(finishedInterval);  
    }  
}, 100);
```

Common tasks and best practices

Writing safe gadgets

- Stick to HTML and CSS specs
- Use only safe DOM operations
- Use supported event handlers and timers
- Use supported frameworks:
 - Google Web Toolkit
 - OpenSocial Templates
 - JQuery

Common tasks and best practices

The good, the bad, and the ugly

Good:

```
jQuery('#content').append('hello world!');  
document.getElementById('content').  
  appendChild(document.createTextNode(  
    'hello world!'));
```

Bad:

```
document.write('hello world!');
```

Ugly:

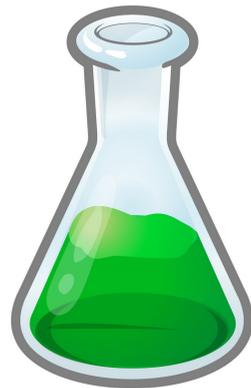
```
window.parent.location = myApi.get(url);
```

How do I distribute gadgets?

How do I distribute gadgets?

Gmail and Calendar

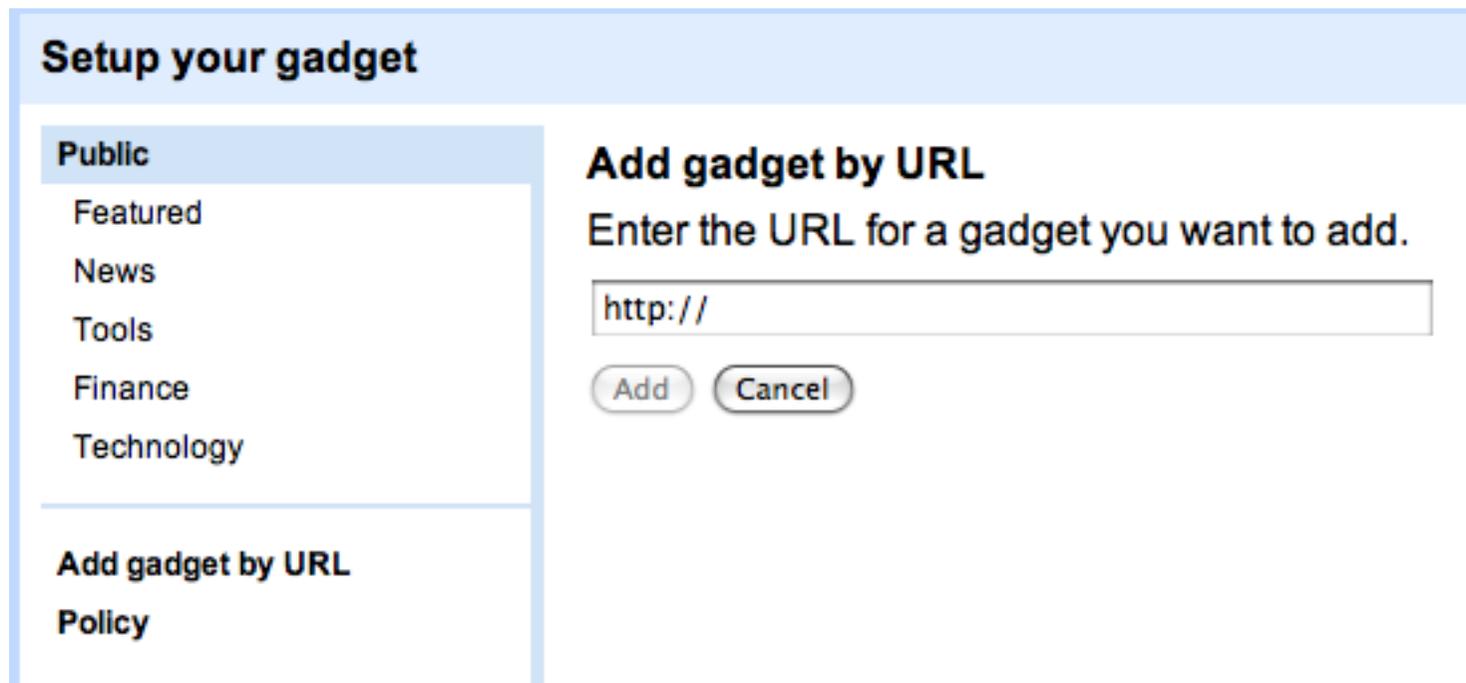
- Gmail contextual gadgets are installed via the Google Apps Marketplace
- Sidebar gadgets are a 'Labs' feature in Gmail and Calendar — the Add Gadget by URL lab
- Calendar gadgets can also be enabled by sharing a custom made URL: **http://www.google.com/calendar/render?gadgeturl=MY_GADGET_URL**



How do I distribute gadgets?

Sites

- Sites features a gadget directory (shared with iGoogle):
<http://google.com/ig/submit>
- Users can also add by URL



The screenshot shows the 'Setup your gadget' interface. On the left, there is a sidebar menu with the following items: 'Public' (highlighted), 'Featured', 'News', 'Tools', 'Finance', and 'Technology'. Below this menu, there are links for 'Add gadget by URL' and 'Policy'. The main content area is titled 'Add gadget by URL' and contains the instruction 'Enter the URL for a gadget you want to add.' Below this instruction is a text input field containing 'http://'. At the bottom of the main area, there are two buttons: 'Add' and 'Cancel'.

How do I distribute gadgets?

Google Apps Marketplace

- To add a link to Google's universal navigation, support OpenID whitelisting, or to write a Gmail contextual gadget, list in the marketplace

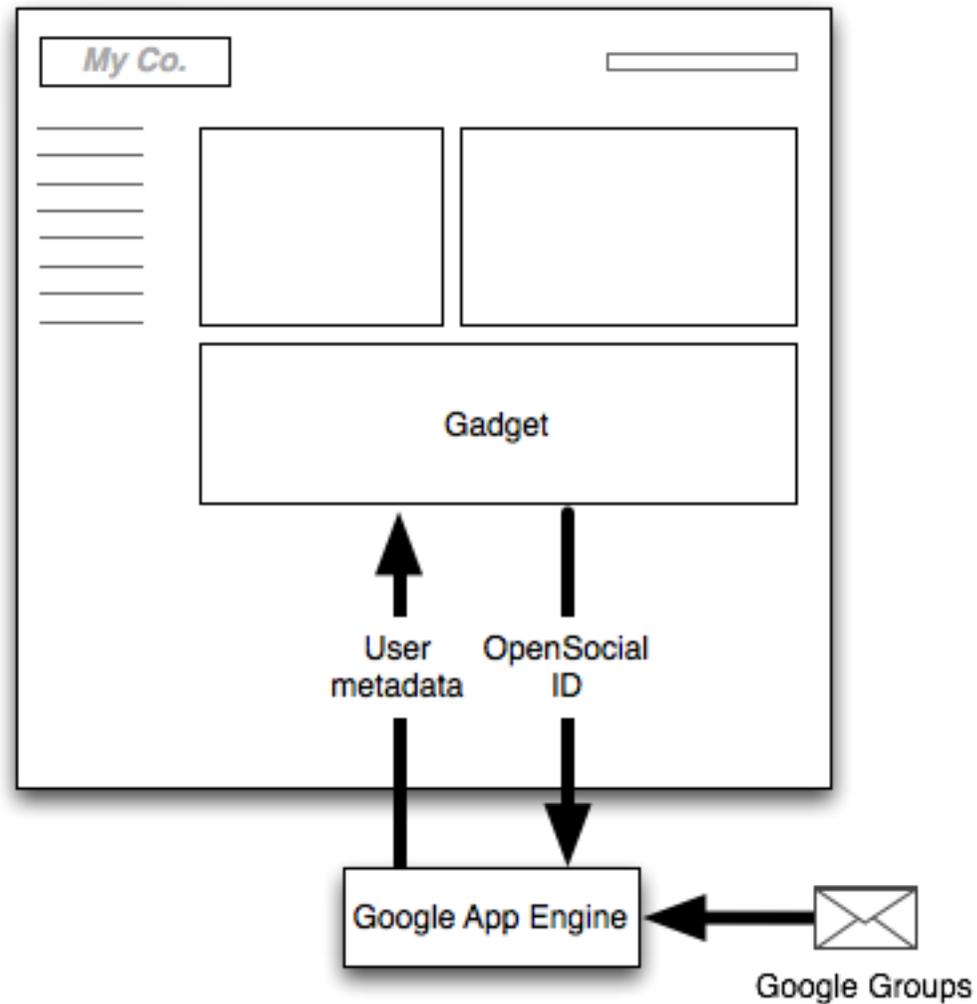
An example application

An example application

Feature summary

- Ticket tracking system
- Provides options to assign owner and status to email threads
- Gadgets provide dashboard views embeddable in Google Apps

An example application Architecture diagram



An example application

Gadgets in Calendar

- Shows a list of tickets resolved during the selected date range

An example application

Gadgets in Calendar

```
google.calendar.subscribeToDates(function(dates) {  
  var start =  
    google.calendar.utils.toDate(  
      dates.startTime).getTime();  
  var end =  
    google.calendar.utils.toDate(  
      dates.endTime).getTime();  
  ds.getThreadsFromPeriod(start, end,  
    function(data) {  
      displayAgentThreads(agent);  
      gadgets.window.adjustHeight();  
    });  
  });
```

An example application

Gadgets in Sites

- A single gadget with 4 display settings:
 - Total/resolved tickets
 - Average resolution time
 - Tickets per day
 - Tickets per agent

An example application

Gadgets in Sites

```
<UserPref name="display_type"  
  display_name="Display Type" default_value="0"  
  datatype="enum" required="true">  
  <EnumValue value="0"  
    display_value="Numbers"/>  
  <EnumValue value="1"  
    display_value="Resolve time"/>  
  <EnumValue value="2"  
    display_value="Tickets over time"/>  
  <EnumValue value="3"  
    display_value="Ownership"/>  
</UserPref>
```

An example application

Gadgets in Sites

```
var type = prefs.getInt('display_type');
```

```
switch(type) {  
  case 0:  
    displayTotals();  
    break;  
  case 1:  
    displayAvgResolution();  
    break  
  ...  
}
```

An example application

Gadgets in Gmail

- Provides a list of tickets the current user owns
- Options to unassign, accept, and resolve tickets
- Clicking on a thread searches for the thread in Gmail

An example application

Application manifest

```
<Extension id="navLink" type="link">  
  <Name>TicTrac</Name>  
  <Url>http://tictrac-test.appspot.com/?  
domain=${DOMAIN_NAME}</Url>  
</Extension>
```

```
<Extension id="realm" type="openIdRealm">  
  <Url>http://tictrac-test.appspot.com</Url>  
</Extension>
```

```
<!-- Gadget extensions coming soon! -->
```

How do I get started?

Getting started today

- Developer documentation:
 - Gmail: <http://code.google.com/apis/gmail/gadgets>
 - Calendar: <http://code.google.com/apis/calendar/gadgets>
 - Sites: http://code.google.com/apis/sites/gadgets/sites_gadgets.html
 - Marketplace: <http://code.google.com/googleapps/marketplace>
- Developer forum:
 - <http://bit.ly/bqPXht>

Other talks

Other talks

- Gmail contextual gadgets, today@4:45, room 4
- Check out the other Enterprise talks when they are posted online: <http://code.google.com/events/io/2010/>

Got questions?

Got questions?

**We're using Wave and Google Moderator:
<http://bit.ly/diq3qc>**

Google™

