Breaking down widget silos with a friendly Wookie

Ross Gardler (@rgardler)
wookie-dev@incubator.apache.org
http://incubator.apache.org/wookie

With thanks to Sander van der Waal

Apache Wookie (Incubating)

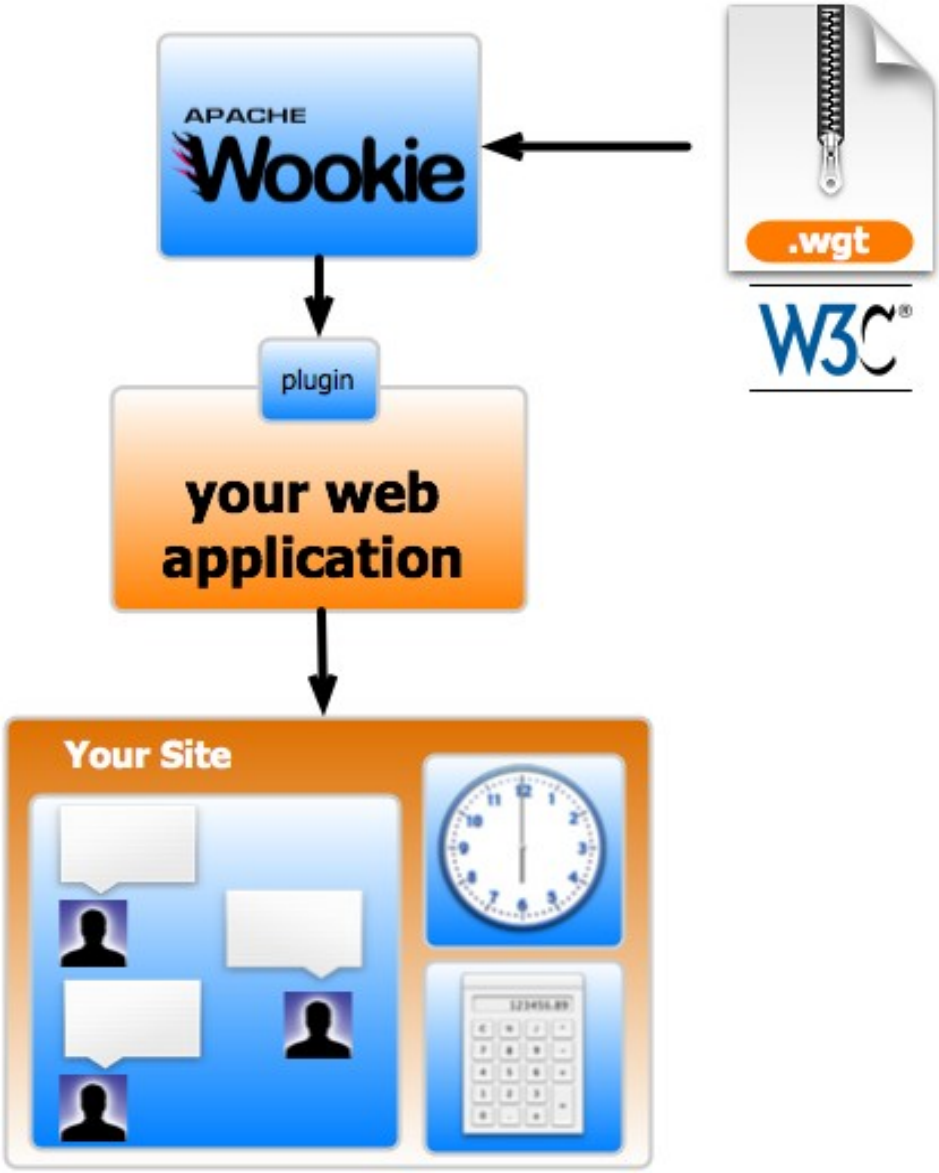
- Upload and deploy widgets for your applications
- Entered incubator July ‘09
  - That's a long time ago
- Originally developed in EU project TenCompetence (Framework 6 IP)
- http://incubator.apache.org/wookie/

Widgets?

- Mini portable applications
  - (typically) Small view area
- Examples include games, clocks, feed displays
  - But not limited to frivolous uses
- Lots of competing models:
  - iPhone SDK, Android SDK, Apple Dashboard, Opera Widgets, Nokia Widgets, Google OpenSocial etc

Wookie is…

- A Widget runtime for websites
- Lets you upload and deploy widgets
- Supports
  - W3C Widgets P&C, W3C Widgets Interface
  - Google / OpenSocial Gadgets
- Provides 3rd party APIs as features
  - Google Wave Gadget API
  - BONDI client APIs


APACHE
Wookie
.wgt
W3C
plugin
your web application
Your Site

W3C Widget Specs

- Packaging and Configuration
  - Recommendation (27 Sep 2011)
  - 100 % conformance
- The Widget Interface
  - Last Call Working Draft 7 June 2011
  - 97 % conformance
- Widget Access Request Policy (WARP)
  - Recommendation (20 April 2010)
  - Conformance tests TBA

W3C DAP

- Device APIs & Policy Working Group
- Collection of client-side APIs
- Specifications for:
  - Camera
  - Contacts
  - Device status and settings
  - Messaging
  - Media gallery
- Input from eg. Nokia's APIs and BONDI

W3C Widget How-to

- Create :
  - HTML + CSS + JavaScript + image files
  - Add a “config.xml” file with some basic info
- Zip the lot
- Change the extension to .wgt
- Deploy in widget server
- That's it.

Or use Wookie skeletons

- “ant seed-widget”
- Edit a few files
  - HTML
  - CSS
  - Javascript
- “ant build-widget” or “ant deploy-widget”

Or use Wookie templates

- “mkdir WIDGETNAME; cd WIDGETNAME”
- “cp ../templates/default.widget.properties”
- Edit even fewer files
  - “*.properties”
  - Perhaps
    - HTML/CSS
    - Javascript
- “ant generate-widgets”

Wookie is also…

- A server for integrating widgets in 3rd party apps
  - Moodle
  - Wordpress
  - Drupal
  - And more:
    - Elgg, LAMS, BlackBoard
- Provides connector frameworks for integration
  - Perl, PHP, Python, Ruby, C#

Wookie under the hood

- Java web application

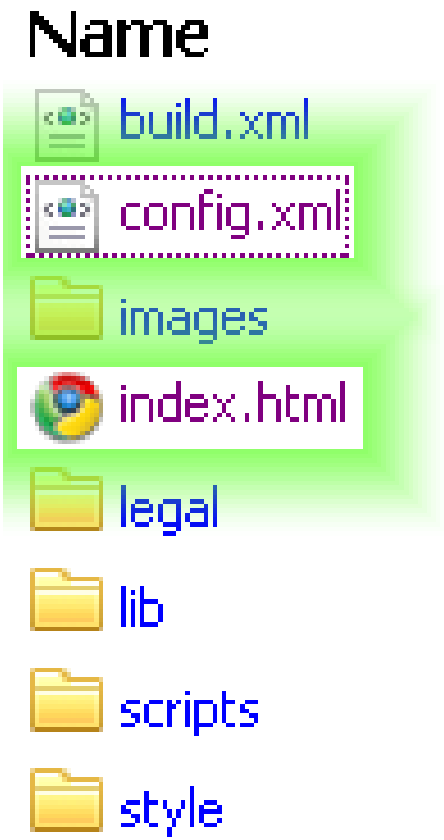
- REST API for widget management
  - (DEPRECATED) Admin UI for managing W3C Widgets
- JavaScript API objects for widgets at runtime
- JPA for server-side storage
- Comet / DWR used for processing JS events

Wookie REST API

- GET /widgets
  - Gets the list of widgets installed
- POST /widgetinstances
  - Instantiate a widget
- POST /participants
  - Add a participant to a widget instance
- Lots more at

  http://incubator.apache.org/wookie/docs/api.html

Widget File Structure



confix.xml File Structure

```
<widget xmlns="http://www.w3.org/ns/widgets"
    id="http://www.getwookie.org/widgets/weather"
    height="125" width="125">
  <name>Weather</name>
  <description>A silly Weather widget</description>
  <icon src="images/icon.png"/>
  <content src="index.html"/>
  <author>Scott Wilson</author>
  <licence>Licensed under the Apache 2.0 License (see
http://www.apache.org/licenses/LICENSE-2.0)
  </licence>
</widget>
```

Widget API example

- Widget.preferences
  - Gives access to preferences stored for the widget
- Widget.preferences.setItem(name, value)
  - Widget.preferences.setItem('displayStatus', this.checked)
- Widget.preferences.getItem(name)
  - Widget.preferences.getItem("displayStatus")
- Widget.proxify(external_url)
  - Access external url

Security and privacy

- Can store user credentials in widget.preferences
  - not very secure
- In-development oAuth functionality
- Same-origin policy;
  - Wookie provides a proxy
  - CORS is on roadmap

Widget runtime APIs

```
<feature name="http://bondi.omtp.org/api/camera.capture"
         required="true" />
```

- W3C Widget Object : preferences, metadata
- BONDI camera API, W3C DAP
- Google Wave Gadget API : state, participants
- SCORM CMI API (eLearning)
- .. anything else!

Let's see that in action..

- Simple widgets
  - Bubbles
- Widgets using 3rd party APIs
  - Weather

Collaborative Apps

- Use W3C Widgets packaging and widget object API with the Google Wave Gadgets API
- Runs in Wookie, no Wave server needed

- APIs:
  - State
  - Participants

State

- State is shared across “sibling” widgets
- State is propagated to related widgets using an event callback
- State is set by submitting deltas (as associative arrays) or single values

State example

```
wave.setStateCallback(stateUpdated);

stateUpdated = function(){
  var keys = wave.getState().getKeys();

  for (var i = 0; i < keys.length; i++) {
    alert(wave.getState().get(keys[i]));
  }
};

wave.getState().submitValue("key", "value");
```

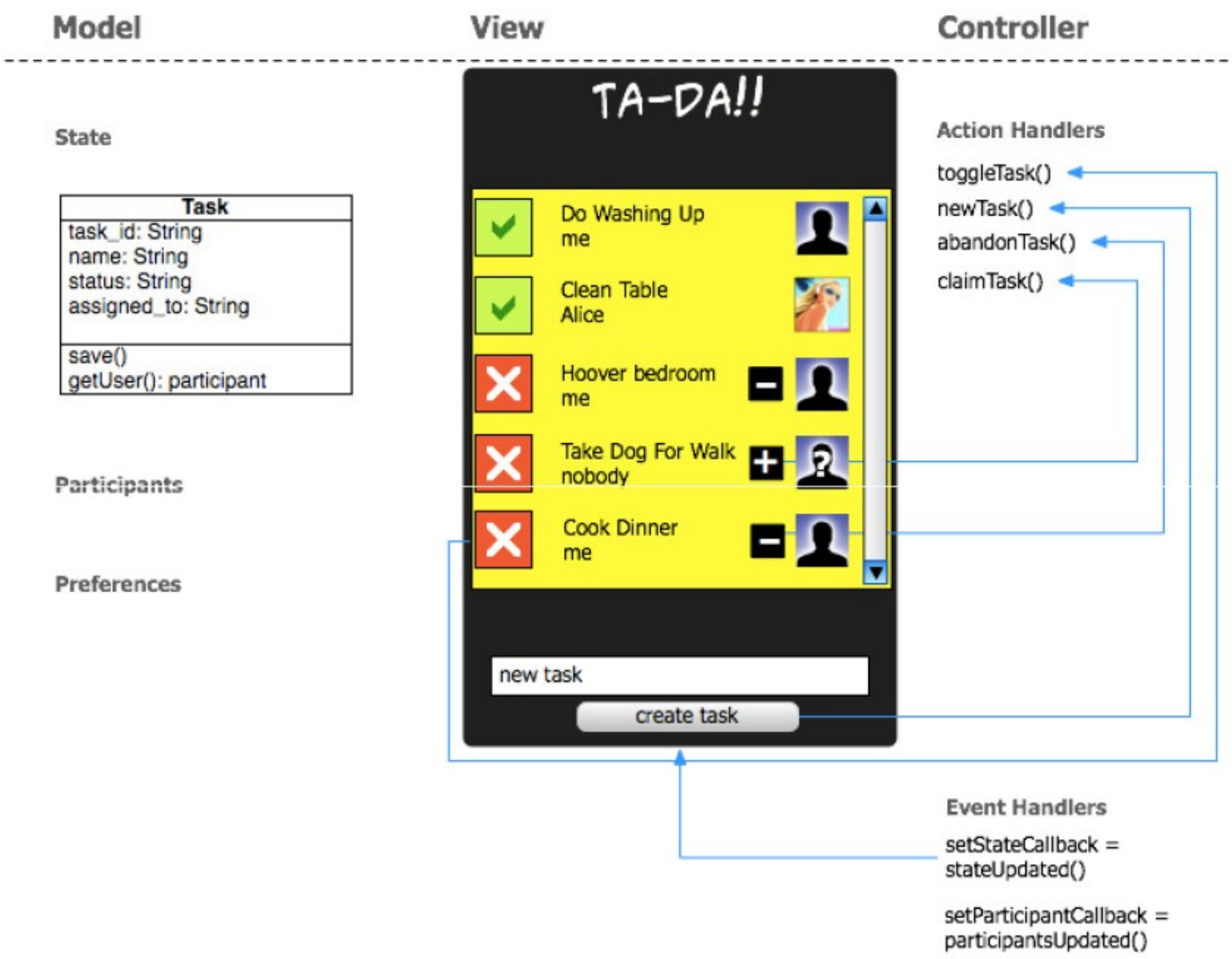
Participants

- Who is using a shared instance
- Use Wookie's participants REST API
- Viewer is the current user object
  - participants is the set of users

Participants example

```
wave.setParticipantCallback(refreshMembers);


refreshMembers: function(){
  var participants = wave.getParticipants();
  var memberList = "";
  for (participant in participants) {
    name = participants[participant].getDisplayName();
    memberList = name + "<br />" + memberList;
  }
  ...
}
```


Model
View
Controller
State
Task
task_id: String
name: String
status: String
assigned_to: String
save()
getUser(): participant
Participants
Preferences
TA-DA!!
Do Washing Up
me
Clean Table
Alice
Hoover bedroom
me
Take Dog For Walk
nobody
Cook Dinner
me
new task
create task
Action Handlers
toggleTask()
newTask()
abandonTask()
claimTask()
Event Handlers
setStateCallback =
stateUpdated()
setParticipantCallback =
participantsUpdated()

More action..

- Collaborating widgets
  - Sharing state
  - Using participants

Apache Rave (Incubating)

- A web and social mashup engine
- Host, serve and aggregate
  - (Open)Social Gadgets using Shinding
  - W3C Widgets using Wookie
- Context-aware personalization and collaboration

Wookie Vs. Rave

- Wookie
  - W3C Widgets
  - Google Gadgets
  - Server only
  - Widget interaction
  - Widget Lifecycle
  - Context aware widgets
- Rave
  - Open Social
  - Widgets and Gadgets
  - Widgetstore
  - UI
  - Context aware portals

Get Involved!

- **http://incubator.apache.org/wookie**
- Create ***widgets*** Wookie can serve up
- Create ***plugins*** to connect Wookie with other platforms
- Create ***features*** that add more runtime capabilities
- Contribute to ***improving*** the server code itself
- Build web based systems with Rave
  - **http://incubator.apache.org/rave**