Building Sakai Tools in Google App Engine

Dr. Charles Severance
University of Michigan
IMS Global Learning Consortium
www.dr-chuck.com
http://drchuck.dyndns.org
/~csev/sakai-workshop/

www.appenginelearn.com
www.pythonlearn.com
www.activestate.com
My Goals for Today

• Talk about use cases and scenarios for non-enterprise “Learning Tools”
• Talk about Google App Engine
• Talk about the IMS Basic LTI Specification
• Show you how to add BasicLTI support to Sakai
• Show you how to build a Learning Tool in App Engine using the Tsugi framework
Background Task: Install App Engine

- Please install AppEngine and get the trivial application running while I am talking
- www.appenginelearn.com
- Helper scripts
- Handouts
- Book Appendices
There are more than 15 applications...

2008 Sakai Usage Data at University of Michigan
Scenarios / Use Cases

- Hobby Application - Support Users and Courses around the world
- Instructor built simple, interactive tools - Like Wisdom of Crowds
- Local Instructional Support Staff - Build one-off simple tool for a particular instructor
Log in with your GMail account

Navigate to the Wisdom of Crowds Application

We will come back to this later in the talk
Collaborative eScience: Evolving Approaches

Charles Severance
Executive Director, Sakai Foundation

Shaping Collaboration 2006
Geneva, Switzerland
December 11-13, 2006
Chuck’s Goals for Sakai

- Bring enterprise-level open source LMS into the Market
- Collect bright worldwide developer community and achieve sufficient adoption for sustainability
- Focus on ease of adding new tools
- Create market to enable the free exchange of tools
- Get teachers writing tools
Learning Tools Interoperability

- Content Integration
  - McGraw-Hill Katana
  - Pearson TPI
  - K12 - www.k12.com
  - Google Documents (SAML)

- LMS Integration
  - Angel
  - Sakai Porlet / Melete
  - Moodle
  - Microsoft MIRLearn
  - Blackboard Building Block
wiscrowd.appspot.com

- Log in with your GMail account
- Navigate to the Wisdom of Crowds Application
- Lets guess some angles
import logging
import pickle
from google.appengine.ext import db
from core.tool import ToolRegistration
from core import learningportlet

class Wisdom(db.Model):
    blob = db.BlobProperty()

def register():
    return ToolRegistration(WisHandler, "Wisdom of Crowds")

class WisHandler(learningportlet.LearningPortlet):

    def doaction(self):
        wisdom = Wisdom.get_or_insert("a", parent=self.context.course)
        if wisdom.blob == None:
            wisdom.blob = pickle.dumps( dict() )
            wisdom.put()
        data = pickle.loads(wisdom.blob)
        name = self.request.get("name")
        if len(name) < 1: name = self.context.user.email
        guess = self.request.get("guess")
        try: guess = int(guess)
        except: guess = -1
import logging
import pickle
from google.appengine.api import memcache

from core.tool import ToolRegistration
from core import learningportlet

def register():
    return ToolRegistration(WisHandler,
                             "Wisdom of Crowds", 
                             """Game""""
                             )

class WisHandler(learningportlet.LearningPortlet):
    def doaction(self):
        wiskey = "WisCrowd-"+str(self.context.course.key())
data = self.getmodel(wiskey)

        name = self.request.get("name")
        if len(name) < 1 : name = self.context.user.email

        guess = self.request.get("guess")
        try: guess = int(guess)
        except: guess = -1
msg = ""
if self.context.isInstructor() and name.lower() == "reset":
    data = dict()
    logging.info("Storing Wis Key=\""+wiskey)  
    memcache.set(wiskey, data, 3600)
    msg = "Data reset"
elif guess < 1:
    msg = "Please enter a valid, numeric guess"
elif len(name) < 1:
    msg = "No Name Found"
elif name in data:
    msg = "You already have answered this"
elif len(data) > 1000:
    msg = "Game only supports 1000 players."
else:
    data[name] = guess
    memcache.set(wiskey, data, 3600)
    logging.info("Storing Wis Key=\""+wiskey)
    data = self.getmodel(wiskey)  
    if data.get(name,None) == guess :  
        msg = "Thank you for your guess"
    else:
        msg = "Unable to store your guess please re-submit"
return msg
def getview(self, info):
    wiskey = "WisCrowd-"+str(self.context.course.key())
    data = self.getmodel(wiskey)
    rendervars = {'context': self.context,
                  'msg' : info}
    if self.context.isInstructor() and len(data) > 0 :
        text = ""
        total = 0
        for (key, val) in data.items():
            text = text + key + "," + str(val) + "\n"
            total = total + val
        count = len(data)
        ave = 0
        if count > 0 : ave = total / count
        rendervars["ave"] = ave
        rendervars["count"] = count
        rendervars["data"] = text
        return self.doRender('index.htm', rendervars)

def getmodel(self, wiskey):
    data = memcache.get(wiskey)
    if data == None or not isinstance(data, dict):
        data = dict()
    return data
Hello: {{ context.getUserName }}
{% if context.isInstructor %}
(Instructor)
{% endif %}
from {{ context.getCourseName }}</p>
{% if msg %}
<p>{{ msg }}</p>
{% endif %}
<form method="post" action="{{ context.getPostPath }}">
{{ context.getFormFields }}
<p>Enter Guess: <input type="text" name="guess"/></p>
<p>Enter Name (opt): <input type="text" name="name"/></p>
<p><input type="submit"></p>
</form>
{% if ave %}
<p>Average: {{ ave }} Count: {{count}}</p>
{% endif %}
{% if data %}
<pre>
{{ data }}
</pre>
{% endif %}
CloudCollab Goals

• Easy to use tool building and hosting environment
• Support IMS Learning Tools Interoperability and other IMS specs
• Written in Python, Free, Open Source
• Hostable on Google for Free
• Teachers writing tools - students writing tools (Thousands)

www.cloudcollab.org
Google App Engine
Google App Engine Intro

- Why App Engine?
- Make the web better
- More developers
- It is nice to outsource production issues

http://www.youtube.com/watch?v=tcbpTQXNwac
Simple Application Video

- Brett Slatkin builds a Google App Engine application in 10 minutes
- Basic structure - GET / POST
- Form Input
- Templating
- Database

http://www.youtube.com/watch?v=tcbpTQXNwac
Google App Engine

- When you write a Google Application Engine Application - you are running in the Google Cloud
- Just like you were a Google Developer
- You don’t know where you are running or if one copy of a thousand copies of you are running
- Google hosts small applications for *free* - larger applications pay by usage
<table>
<thead>
<tr>
<th>Resource</th>
<th>Free Default Quota</th>
<th>Billing Enabled Quota</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Daily Limit</td>
<td>Maximum Rate</td>
</tr>
<tr>
<td>Requests</td>
<td>1,300,000 requests</td>
<td>7,400 requests/minute</td>
</tr>
<tr>
<td>Outgoing Bandwidth (billable, includes HTTPS)</td>
<td>1 gigabyte</td>
<td>56 megabytes/minute</td>
</tr>
<tr>
<td>Incoming Bandwidth (billable, includes HTTPS)</td>
<td>1 gigabyte</td>
<td>56 megabytes/minute</td>
</tr>
<tr>
<td>CPU Time (billable)</td>
<td>6.5 CPU-hours</td>
<td>15 CPU-minute/minute</td>
</tr>
</tbody>
</table>
Download the Google App Engine SDK

Before downloading, please read the Terms that govern your use of the App Engine SDK.

Please note: The App Engine SDK is under active development, please keep this in mind as you explore its capabilities. See the SDK Release Notes for the information on the most recent changes to the App Engine SDK. If you discover any issues, please feel free to notify us via our Issue Tracker.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Version</th>
<th>Package</th>
<th>Size</th>
<th>SHA1 Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows</td>
<td>1.1.9 - 02/09/09</td>
<td>GoogleAppEngine_1.1.9.msi</td>
<td>2.7 MB</td>
<td>c0a09f23950de606a4c4933687335582c18bd3cd</td>
</tr>
<tr>
<td>Mac OS X</td>
<td>1.1.9 - 02/09/09</td>
<td>GoogleAppEngineLauncher-1.1.9.dmg</td>
<td>3.8 MB</td>
<td>d8ca6afa091f0a1e48e2132454a71bf7ecbfed27</td>
</tr>
<tr>
<td>Linux/Other Platforms</td>
<td>1.1.9 - 02/09/09</td>
<td>google_appengine_1.1.9.zip</td>
<td>2.8 MB</td>
<td>aa11e2e6940071cf61c3092b2365605a01d739e7</td>
</tr>
</tbody>
</table>

http://code.google.com/appengine/downloads.html
app.yaml - Tells Google about the application
index.py - the code for our application
```yaml
application: ae-01-trivial.
version: 1.
runtime: python.
api_version: 1.

handlers:
  - url: /*.
    script: index.py.
```

```python
print "Hello There Chuck",
```
Helper Scripts

• http://www.appenginelearn.com/static/aehelpers.zip
charles-severances-macbook-pro:apps csev$ ls -l
total 8
-rw-r--r-- 1 csev staff 742 Nov 21 2008 appengine.sh
drwxr-xr-x 4 csev staff 136 Jul 7 00:06 trivial
charles-severances-macbook-pro:apps csev$ sh appengine.sh trivial/
Checking trivial/
INFO 2009-07-07 04:07:03,924 appcfg.py:322] Checking for updates to the SDK.
INFO 2009-07-07 04:07:04,041 appcfg.py:336] The SDK is up to date.

Hello world
Writing a CloudCollab Portlet

www.cloudcollab.org
CloudCollab Goals

- Easy to use tool building and hosting environment
- Support IMS Learning Tools Interoperability and other IMS specs
- Written in Python, Free, Open Source
- Hostable on Google for Free
- Teachers writing tools - students writing tools (Thousands)

www.cloudcollab.org
A CloudCollab “Portlet”

- Inspired by JSR-168 and JSR-286 “Java Portlet Specification”
- The tool produces a “fragment” to be placed on a screen using a portal
- The enclosing portal decides how to show the context
- There are constraints on the HTML produced by the portlet so the portal can repurpose the Portlet’s fragments
CloudCollab Markup

- Each portlet is also has a “servlet-style” URL and can be placed into an iframe
- A portlet can be told to constrain itself within a named div (this is how CloudCollab’s own internal portal will work)
- A portlet can be told to return Facebook Markup Language instead of HTML
- (Future) a portlet can generate Web Services for Remote Portlets (WSRP)
CloudCollab Authentication

- CloudCollab can establish user identity, organizational identity, and course context in a number of ways
  - IMS Simple LTI
  - IMS Basic LTI
  - Google Accounts
  - Facebook Accounts (A CloudCollab tool is also a Facebook tool)
- CloudCollab is inherently multi-tenancy (i.e. multiple organizations)
CloudCollab Framework (Tsugi)

- Portlets are tolerant of cookies turned off
- Portlets are tolerant of JavaScript turned off
- (Future) Accessibility will be accomplished by using Fluid approaches of generating very basic HTML and using JavaScript to improve the HTML.
Hello: drchuck (Instructor) from Test via google

Enter Guess: 

Enter Name (opt): 

Guess  Reset Game Data

Hello: Charles Severance from Test via facebook

Please enter a valid, numeric guess

Enter Guess: 

Enter Name (opt): 

Guess
Hello: Sakai Administrator (Instructor) from Test via basicli

Enter Guess:

Enter Name (opt):

Hello: Charles Severance (Instructor) from Test via simpleli

Enter Guess:

Enter Name (opt):
Hello: drchuck (Instructor) from Test via google

Enter Guess: 

Enter Name (opt): 

Guess  Reset Game Data
Exercise

http://drchuck.dyndns.org:8080/portal/
Make a new account with full info
Create a Project Site
Add the BasicLTI Proxy Tool to your site
Configure the tool to point to
http://wiscrowd.appspot.com/wiscrowd/12345/
secret is secret
Download CloudCollab

```
svn co http://cloudcollab.googlecode.com/svn/tsugi/trunk wiscrowd
```

Check this out as a directory in your “apps” directory. It is a complete App Engine application.

Once downloaded, start up the wiscrowd (depending on OS)

```
dev_appserver.py wiscrowd
sh appengine.sh wiscrowd
appengine wiscrowd
```
Welcome to the demo of the CloudCollab framework for building applications using IMS Learning Tools Interoperability (Simple LTI and Basic LTI).

This looks like an LMS - but really it is intended to be a collection of tools launched from another LMS using the IMS LTI protocol.

If you want to simulate a Simple LTI Launch click [here].

The IMS Learning Tools Interoperability and Basic LTI are being developed by the [www.imsglobal.org](http://www.imsglobal.org). If you are interested in standards and interoperability for teaching and learning technology you should become involved in IMS.

Questions and comments to [Dr. Chuck](mailto:dr.chuck@charlesseverance.com)

[Simple Learning Tools Interoperability](http://imslearningtools.org)
[IMS Learning Tools Interoperability v2.0 Working Group](http://imslearningtools.org/v2)
© 2009 Charles Severance All Rights Reserved.
Exercise

- Login in to CloudCollab on http://localhost:8080
- Navigate to Wisdom of Crowds tool in the CloudCollab portal and also look at the fragment
- Go back to http://drchuck.dyndns.org:8080/portal
- Configure your BasicLTI Tool in Sakai to point to
  - http://localhost:8080/wiscrowd/12345/
- You should see the Wisdom of Crowds tool inside of Sakai
A New Portlet (Handout)

- Go into `wiscrowd/mod`
- Make new folder called “trivial”
- Create `__init__.py` and `index.py` per handout instructions
- Exit and restart your `wiscrowd` server to have it rescan the modules directory
- Verify that the Trivial Application appears
from core import tool
from core import learningportlet

def register():
    return tool.ToolRegistration(TrivialHandler, 'Trivial Tool', 'This tool is very very trivial')

class TrivialHandler(learningportlet.LearningPortlet):
    def doaction(self):
        return None

    def getview(self, info):
        return 'Hello from getview()'
Action and View

- Servlets think in terms of “GET” and “POST”
- Portlets think in terms of “handle an action” and “produce a view”
- You are only supposed to change state in the “handle an action” phase
- GET and POST map to ACTION and VIEW differently depending on who is requesting the markup (browsers, Ajax, Facebook, etc)
def handlepost(self):
    info = self.doaction()
    if (not isinstance(self.div, str)) and self.redirectafterpost is True:
        self.session["info"] = info
        redirecturl = self.getGetUrl(action="view")
        self.redirect(redirecturl)
        return None
    else:
        output = self.getview(info)
        self.session.delete_item("info")
        return output

def handleget(self):
    if self.action == "view":  
        info = self.session.get("info", None)
    else:  
        info = self.doaction()
    self.action = "view"
    self.session.delete_item("info")
    output = self.getview(info)
    return output
Scenario Sequences

Simple GET: getview()
GET with Action: doaction() getview()
Normal POST: doaction() redirect getview()
AJAX POST: doaction() getview()
FaceBook POST with no action: getview()
FaceBook POST with action: doaction() getview()

It is a bit complex when all the choices are contemplated.
HTML Helpers

• In order to generate the proper HTML for AJAX / nonAJAX, Facebook, etc destinations of the markup, we have some helpers which generate HTML for us

• The helpers are an extension to the templating language
Exercise

- Go through the “Add a Form to the Application” steps
def doaction(self):
    logging.info("doaction action=%s" % self.action)
    guess = self.request.get('guess')
    try:
        iguess = int(guess)
    except:
        return 'Please enter a valid guess'

    if iguess == 42:
        return 'Congratulations!'
    elif iguess < 42:
        return 'Your guess %s is too low' % iguess
    else:
        return 'Your guess %s is too high' % iguess
def getview(self, info):
    rendervars = dict()

    if isinstance(info,str):
        rendervars['msg'] = info

    return self.doRender('index.htm', rendervars)
Welcome to the Guessing Game
{% if msg %}
  <p>{{msg}}</p>
{% endif %}
{% form_tag %}
  <input type="text" name="guess" size="40">
  {! form_submit('Guess') !}
{% endform %}
Exercise

- Go through the “Using a Session” steps
def doaction(self):
    guess = self.request.get('guess')
    try:
        iguess = int(guess)
    except:
        return 'Please enter a valid guess'

    self.session['guesses'] = self.session.get('guesses', 0) + 1

    if iguess == 42:
        return 'Congratulations!'
    elif iguess < 42:
        return 'Your guess %s is too low' % iguess
    else:
        return 'Your guess %s is too high' % iguess
def getview(self, info):
    rendervars = dict()

    if not info is None:
        rendervars['msg'] = info
        rendervars['guesses'] = self.session.get('guesses', 0)

    return self.doRender('index.htm', rendervars)
Welcome to the Guessing Game

{% if msg %}
<p>{{msg}}</p>
{% endif %}

<p>Guess Count: {{ guesses }}</p>

<! form_tag !>
<input type="text" name="guess" size="40">
<! form_submit('Guess') !>

</form>
Exercise

- Go through the “Reset button” steps
def doaction(self):
    logging.info("doaction action=%s" % self.action)
    if self.action == 'reset':
        self.session.delete_item('guesses')
        return "Guess count reset!"

    guess = self.request.get('guess')

    try:
        iguess = int(guess)
    except:
        return ' Please enter a valid guess'
    ....
<p>Welcome to the Guessing Game</p>
{% if msg %}
    <p>{{msg}}</p>
{% endif %}
<p>Guess Count: {{ guesses }}</p>
<! form_tag !>
<input type="text" name="guess" size="40">
<! form_submit('Guess') !>
<! form_button('Reset Game Data', action='reset') !>
</form>
form_tag(attributes={}, params={}, action=False, resource=False, controller=False, context_id=False)

form_submit(text, attributes={})

form_button(text, params={}, attributes={}, action=False, resource=False, controller=False, context_id=False)

link_to(text, params={}, attributes={}, action=False, resource=False, controller=False, context_id=False)

ajax_url(params={}, action=False, resource=False, controller=False, context_id=False)

HTML Helpers
HTML Helper Examples

{! link_to('Click Me', attributes={'class' : 'selected' },
    action='act-anchor') !}

{! form_button('Cancel', attributes={'class' : 'selected' },
    action='act-cancel') !}
Next Steps

- Read the “Using Google App Engine” book
- Learn more about Python, Templates, Datastore, etc...
- Send URLs and secrets to your friends :)
- Give me feedback on these ideas
Introducing IMS Basic Learning Tools Interoperability

In PowerPoint...
IMS Basic Learning Tools
Interoperability Support in Sakai
SimpleLTI Support

- SimpleLTI Portlet available
- https://source.sakaiproject.org/contrib/sakai-portlets/trunk/
- Running in production at UM since Fall 2008
- Can be a tool in a worksite or a virtual tool
- Suported in Melete 2.6.1 - Authoring / Import / Export
The important thing is not to stop questioning.

-- Einstein

Testing Tools Interoperability
Wisdom of Crowds

Hello: csev (instructor) from DEMO CSEV 101 DEV

Enter Guess: [blank]

Enter Name (opt): [blank]

Submit
SimpleLTI Virtual Tool

- Allows the system administrators to register fake tools that are really Simple LTI Tools with pre-set URLs and secrets
- These tools show up in Site Setup and can be added to a site by the teacher - the teachers often never even know these tools use SimpleLTI
### Omnibus II Score Summary

(Note: [My %] = [My Total] / [Set Total] * [Set %])

<table>
<thead>
<tr>
<th>Set</th>
<th>Date</th>
<th>Short Name</th>
<th>Description</th>
<th>My Total</th>
<th>Set Total</th>
<th>Set %</th>
<th>My %</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>07/01/2008</td>
<td>celSph</td>
<td>Celestial Sphere</td>
<td>9.5</td>
<td>10</td>
<td>10</td>
<td>9.50</td>
</tr>
<tr>
<td>2</td>
<td>07/08/2008</td>
<td>spect</td>
<td>Spectroscopy</td>
<td>8</td>
<td>10</td>
<td>10</td>
<td>8.00</td>
</tr>
<tr>
<td>3</td>
<td>07/15/2008</td>
<td>HR</td>
<td>HR Diagram</td>
<td>7</td>
<td>10</td>
<td>10</td>
<td>7.00</td>
</tr>
<tr>
<td>4</td>
<td>07/22/2008</td>
<td>Var</td>
<td>Cepheid Variables</td>
<td>9</td>
<td>10</td>
<td>10</td>
<td>9.00</td>
</tr>
<tr>
<td>5</td>
<td>07/29/2008</td>
<td>Age</td>
<td>Age of the Universe</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10.00</td>
</tr>
</tbody>
</table>

Omnibus II Score Achievement: 43.5 out of 50 (87.00 %)

Any questions about the Omnibus II scores? [Request a score correction](#).
BasicLTI in Sakai

- BasicLTI Portlet is complete which supports BasicLTI virtual tools
- https://source.sakaiproject.org/contrib/csev/trunk/basiclti/
- LTI Placements go through /access
- Coming: Support in Melete will be part of a bundle that includes Common Cartridge Import (hopefully 4Q2009)
- Contemplating - Adding “Add New Resource” in Resources Tool
IMS Basic Learning Tool Interoperability

**Required Information**

- **Remote Tool Url:**
  

- **Remote Tool Password:** ******  
  (Must re-enter every time)

**Display Information**

- **Set Page Title:** BasicLTI  
  (Button text)

- **Set Tool Title:**  
  (Above the tool)

**Optional Launch Information**

- **iFrame Height:**  
  *(What's this?)*

- **Debug Launch:**  
  *(What's this?)*
Thank you

www.tsugiproject.org
www.appenginelearn.com
www.pythonlearn.com
www.dr-chuck.com