Designing Redwood City’s Community GIS:
Opportunities & Challenges

by
Stephen de Jong
Joe Marvin
Background
Redwood City

- Located in Silicon Valley
- 76,000 Inhabitants
- City Provides Its Own Water & Sewer Service
- 2 Fulltime GIS Staff + 1 Web Developer
City’s Web-GIS History

- **2005:** MapGuide 6.5  
  (Intranet only)

- **2009:** MapGuide Enterprise

- **2014:** ArcGIS Server for JavaScript API
Redwood City’s 2014 Community GIS
Opportunities
Primary Reasons for Moving to ArcGIS Server

- Only Having to Use Data in SDE
- Modern Map User Interface
- Stable and Supported Platform
- Ability to Consume 3rd Party Basemaps
- Familiar Map Design Interface (ArcMap)
- Large Library of Templates/Widgets
- Technical Support
First Steps

- Chose JavaScript API as the Development Environment
- Chose an existing ESRI Local Government Template as the Starting Point
- Started Customizing
Custom Features:

- 50+ Layers
- City Projects
- Path of History
- Data Downloads
- Custom Geocoder
- Custom Basemap Switch
- Drawing Tools
- Social Media Sharing
Challenges
Architecture Design
Development GIS Architecture

• When we first began developing, the application had a very simple architecture.

• Challenge: Needed to secure the map services before releasing to the public.
Initial Production GIS Architecture

• Same as development, except using a proxy to secure the map services with tokens

• Challenge

Using the proxy introduced a lot of overhead. The map slowed to a crawl impacting user experience. How to increase performance and have some security?
Current GIS Architecture

- Introduced a Web Adaptor for the non-secure services (mainly the aerials)
- Continued to use the proxy for the secure map services
- Map performance improved drastically
- Used different host names to send request to either proxy or web adaptor
The Widgets
Basemap Gallery Widget

• Challenges:
  - Zoom Levels: Esri basemaps and our basemaps have different zoom levels.
  - Projection: All basemaps must be on the same projection.

• Solutions:
  - Customized the Gallery Widget to allow each basemap to have flexible zoom levels.
  - Projected our basemaps into Mercator so they would match with Esri basemaps.
AGSJS TOC Widget

• Challenges:
  - *Link Layers*: Needed the ability to tie two layers together so they toggle on/off in sync.
  - *Ability to hide layers from legend*, but still make them visible on the map.

• Solutions:
  - Customized the Widget to allow these to work.
Geo-Coder Widget

- **Challenges:**
  - *Parcel Highlighting* – When a search result can be narrowed to Parcel, the parcel should be highlighted, otherwise use lat/long.
  - *Control of Results* – Prioritize Redwood City addresses and allow for searches by APN, Intersection, Place, etc..

- **Solutions:**
  - Customized the GeoCoder Widget to allow both lat/long and direct parcels to be returned from the search.
  - Created a custom GeoCoder service that pulled from multiple internal datasets.
Application DEMO
Application Link:

http://webgis.redwoodcity.org/community/
Tools

- **ESRI**
  - ArcMap 10.1
  - ArcGIS Server 10.1

- **Editors**
  - Notepad++
  - Brackets

- **Gulp**
Resources

- ESRI
  https://developers.arcgis.com/javascript/

- Table of Contents Widget
Contact Info

Stephen de Jong
GIS Coordinator - City of Redwood City
sdejong@redwoodcity.org

Joe Marvin
Web Developer - City of Redwood City
jmarvin@redwoodcity.org