

The Transit Appliance Project

Disruptively low-cost
fixed-location transit arrival
displays

Defining Transit Appliances

- A variety of browser-based low-cost devices to display real-time transit information for a fixed location.
- Make it simple: plug it in and it works.
- Keep operating costs low: leverage existing infrastructure.
 - Agency servers
 - Free or cheap cloud computing (e.g., Google App Engine, CouchOne)
 - Existing WiFi networks

Who's the Audience for Transit Appliances?

- Smartphone users are well served...
- We want to help everyone else:
 - Folks without smartphones
 - Folks who can't be bothered to download an app or open their phone
 - Folks who never thought about it
- People will use more transit if we make it so easy they don't have to think about it.

Use Cases

- Coffee Shops
- Bars
- Building Lobbies
- Storefront Windows near Transit Stops
- Where else?

Hardware Form Factors

- Countertop Displays (8" Chumby Platforms)
 - ~ \$150
- Big-screen (HDTV) display with low-cost processor - ~\$200 plus cost of display
- Wall-mount “picture frame” displays - TBD

Prototypes in the Wild

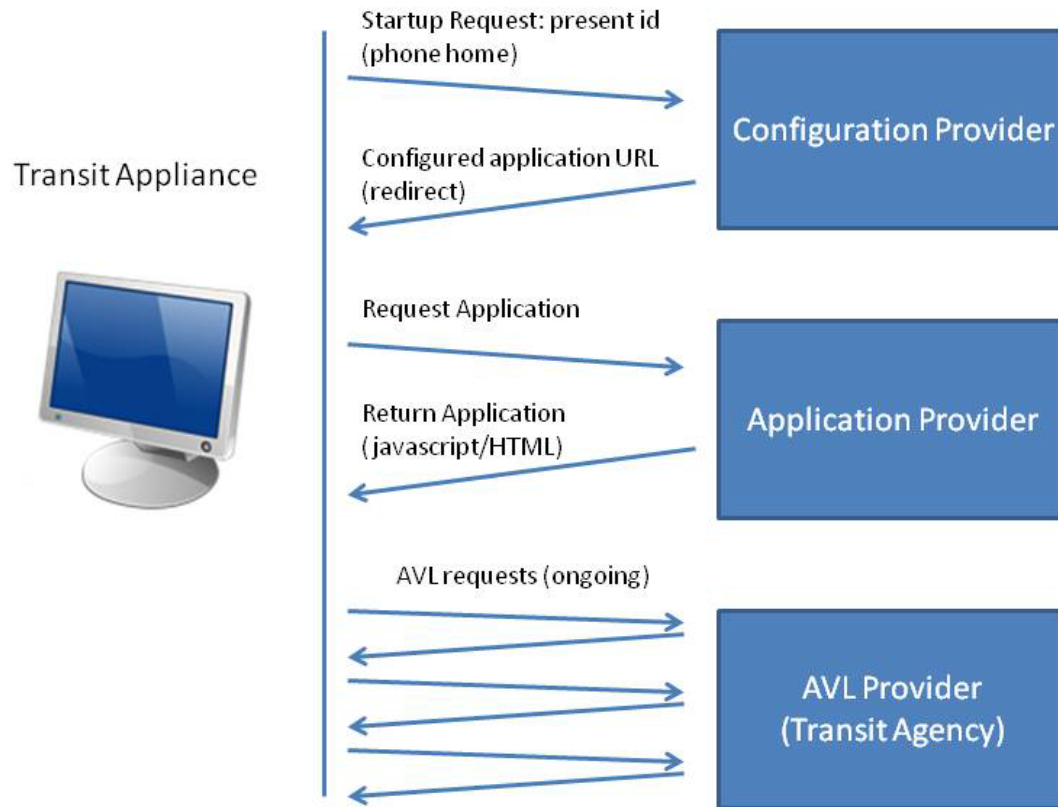
- Portland Building Lobby
- Bailey's Tap Room
 - (SW Broadway & Ankeny in Portland)
- Coming Soon – Large Screen display in lobby of Portland State University Engineering bldg



Introductions

- Team Members
 - Chris Smith (Portland), Architect, lead developer, TriMet/NextBus data/interfaces
 - Matt Conway (SF), 'phone home' loader, MUNI, AC Transit and BART data/interfaces
 - Francis Storr (Portland), UX
 - Scott Garman (Portland), Linux Distro
- Portland Transport
 - Oregon 501(c)(3) with focus on promoting discussion around transportation policies and facilitating tools for transportation information display
 - Home for all project intellectual property

The “Phone Home”Appliance Model



Components of the Project

- Configuration Tool
- Appliance Client
 - Gets configuration, verifies it, loads it
- Multi-agency Transit Stop Web Service
- Standardized JS components for Arrival Displays
- Linux Distro for commodity processors attached to Big Screen Displays

Technologies – JavaScript-centric

- Google App Engine for Configuration Tool
 - JSONEngine storage package
- CouchDB
 - Transit stop database
- YQL
 - Cross-domain JSONP proxy
- jQuery
 - Everywhere!

Open Source

- Stands on the shoulders of MANY open source projects!
- All original code available under Apache 2.0 License

Architecture for JS-based Arrival Displays



An Arrival Object

```
{  
  'arrivalTime' => "1299960275000"  
  'type' => "scheduled"  
  'headsign' => "77 Broadway-Halsey to Troutdale"  
  'stop_id' => "7219"  
  'stop_data' ...  
    [GTFS stop data]  
  'route_id' => "77"  
  'route_data' ...  
    [GTFS Route Data]  
  'agency' => "TriMet"  
  'last_updated' => "1299959306921"  
}
```

The Transit Appliance Ecosystem

- AVL Web Service Providers – Transit Agencies or their vendors
- Configuration Service Providers – Portland Transport will be one, do we need others?
- Service Integration Developers – add new agencies, services to available set (need reduced as AVL services get standardized)
- Application Providers – Many use cases, hopefully many designers
- **Hardware Integrators – who provides hardware and evangelizes, funds or sells it to deployment locations?**

We Need Help! We're Recruiting:

- JavaScript Developers
 - Core display components
 - Agency Adapters
- Java Developers
 - Tweaks to JSONEngine package
- GTFS Gurus
 - Add agencies to stop database
- Code librarians/project managers
 - Setup and manage our source trees
- Linux distro gurus
 - Create custom appliance distro(s)

More Info

- Project Blog
 - <http://transitappliance.org>
- Code Repositories
 - <http://code.google.com/p/transit-appliance-config/>
 - <http://code.google.com/p/transit-appliance-loader/>
 - <https://code.google.com/p/transit-appliance-js-api/>
- Configuration Service
 - <http://service.config.transitappliance.com/>
- chris@chrissmith.us