

# Cable Beach and VWRAP

John Hurliman  
Intel Labs

# What is Cable Beach?

- Researching the next steps in virtual world scalability
- How will the introduction of more virtual worlds and larger virtual worlds impact content, identity, and service deployment?
- How can we keep innovating without starting over every few years?



# What was the purpose?

- Many similar worlds with common protocols are appearing, but no interoperability
- The barrier to entry for running a virtual world is too high
- Innovation is difficult without a way to connect different services or introduce new services

# The vision

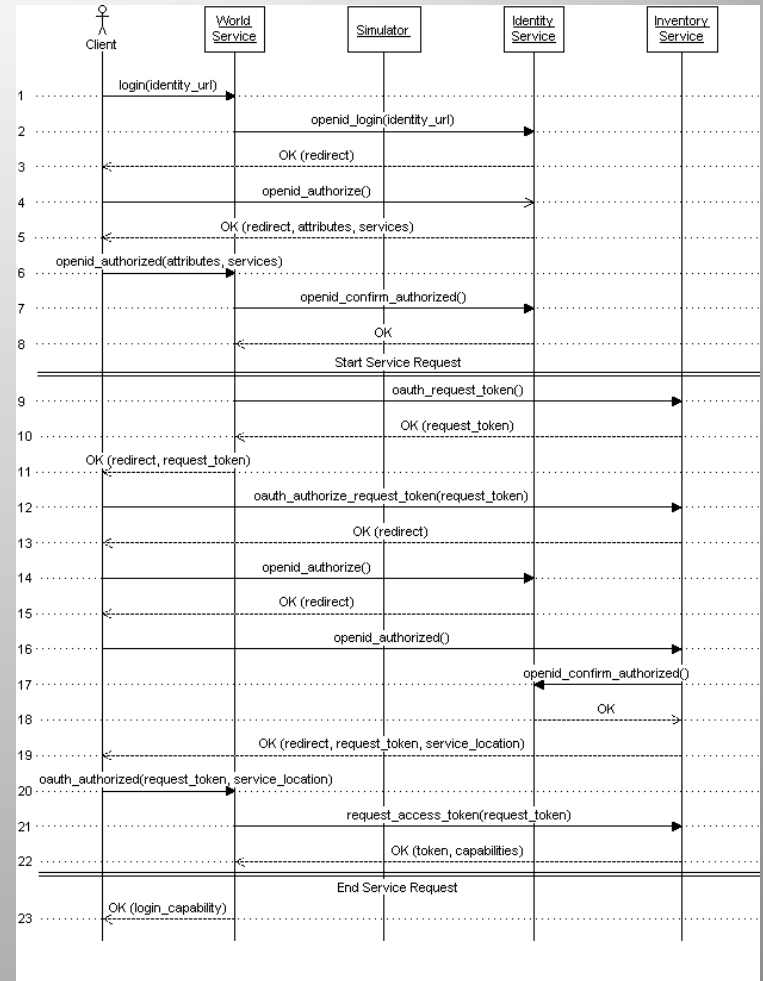
- Millions of independent virtual worlds
  - Many will join common trust domains or subscribe to trust brokers, many will remain completely independent
- Moving between worlds as seamlessly as possible
  - Federated identity, service delegation
- Sharing services between worlds

# Why a separate project?

- Originally unclear whether the projects had common goals
- Keep the research flexible. Provide input to VWRAP without setting the direction
- The opportunity to start from scratch

# The project history

- Cable Beach Asset Server
- Cable Beach Grid Services
- Cable Beach Core 1.0
- realXtend's Naali/Taiga
- MMOX->OGPX->VWRAP



# Lessons learned: assets

- Lots of data, caching is critical, has access control list requirements
- This is a content distribution network, we can leverage existing solutions
- How do we design the rest of the system to work well with existing solutions?

# Lessons learned: inventory

- If assets are blocks of data, the inventory service provides inodes
- By moving all of the mutable metadata for assets into the inventory layer you open the door to immutable content optimizations
- It's difficult to separate the data and the inodes into separate trust domains when there are access restrictions



# Lessons learned: identity

- Virtual world identity is a rich set of data
  - Name and profile
  - Avatar appearance
  - Presence
- Federated identity is only one piece of the puzzle
- Service delegation gets us closer to the goal

# Lessons learned: cross-domain services

- World administrators need policy controls over what services can and can't be used in their worlds
- Users need the ability to bring their own preferred services (if the destination world allows it)
- Tying preferred services to identity is a natural fit for the current direction of VWRAP
- OAuth WRAP was the best match for our requirements

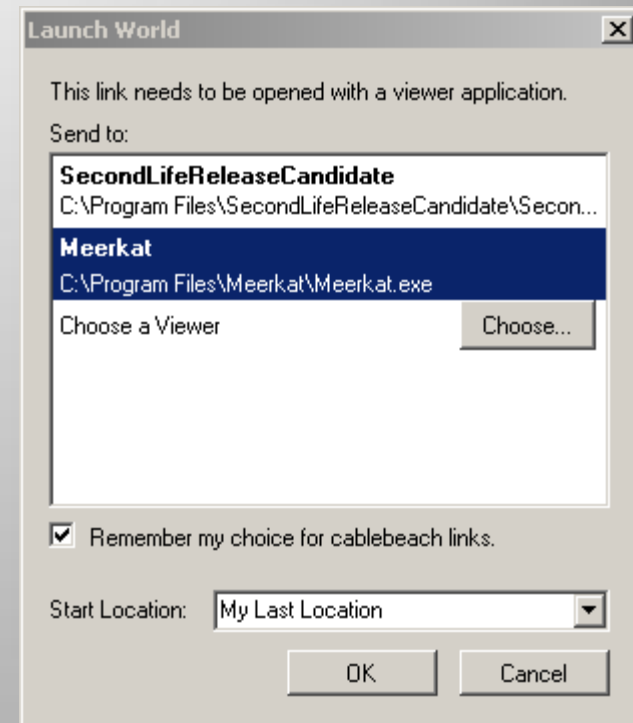


# Implementing LLSD/LLIDL

- LLSD is relatively easy to implement and provides a useful type system on top of loosely structured (JSONish) data
- Pruning out default values and using JSON or binary encoding produces reasonably efficient serializations
- LLIDL seems mostly useful as a human-readable interface description language and possibly in unit testing. Is there a use case where this would be deployed in production services?

# Implementing web authentication

- Moving authentication from the rich client to the web simplifies the implementation
  - Supporting OAuth, CAPTCHAs, Terms of Service agreements, etc. with a rich client is possible, but not preferable for prototyping
- We invented the cablebeach:// URI for testing, the VWRAP launch document replaces this
- Best practices for OAuth logins are still being fleshed out by the web community



# Implementing OGPX (VWRAP)

- I built a region domain in C# to better understand the OGP drafts and test the Snowglobe implementation
- Even with OGP drafts, the spec is not complete enough to implement
- VWRAP documentation needs an overhaul. Aggregate the current docs and prune outdated information

# Merging Cable Beach and VWRAP

- The Cable Beach research project is now entirely folded into VWRAP development efforts
- Working with the Open Metaverse Foundation and OpenSim community to build a new virtual world backend: SimianGrid
- The existing Cable Beach code and region domain implementation are being ported to SimianGrid



# The new platform: SimianGrid

- A set of virtual world backend services written in PHP. Natively supported in the latest OpenSim
- Implements the Second Life / OpenSim model of providing centralized agent domain services to a grid of simulators
- Generic enough to support simulators other than OpenSim (originally tested with Simian)
- Currently in beta grid deployments, ironing out bugs
- A good candidate for a VWRAP region domain implementation



# SimianGrid needs developers!

- Initial development has focused on reducing the barrier to entry for developers
  - Open Metaverse Foundation project. BSD licensed
  - Widely used language (PHP) with few external dependencies
  - Complete API documentation
  - Small but active developer community
- VWRAP is flexible. We need to prioritize which policy controls and deployment patterns are supported first

<http://openmetaverse.googlecode.com/#vwrap> on Freenode

