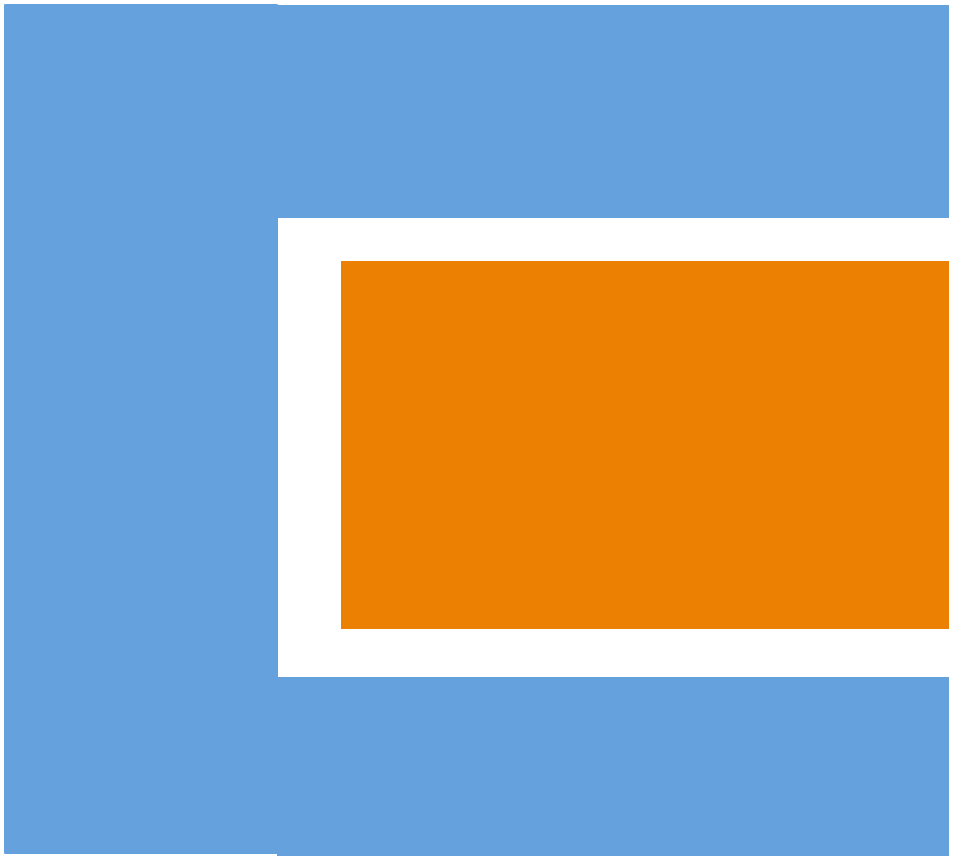
A RESTful Web Service for Whois

Andy Newton
Chief Engineer, ARIN

My Background on Whois

- Prototyped an LDAP alternative to Whois (RFC 3663)
- Principal author of CRISP (IRIS) documents
  - RFC 3707, RFC 3981, RFC 3983, RFC 4698, RFC 4991, RFC 4992, RFC 4993, RFC 5144
- Worked with principal authors of Rwhois and Whois++ at VeriSign
- Most recently – driver behind ARIN's Whois RESTful Web Service

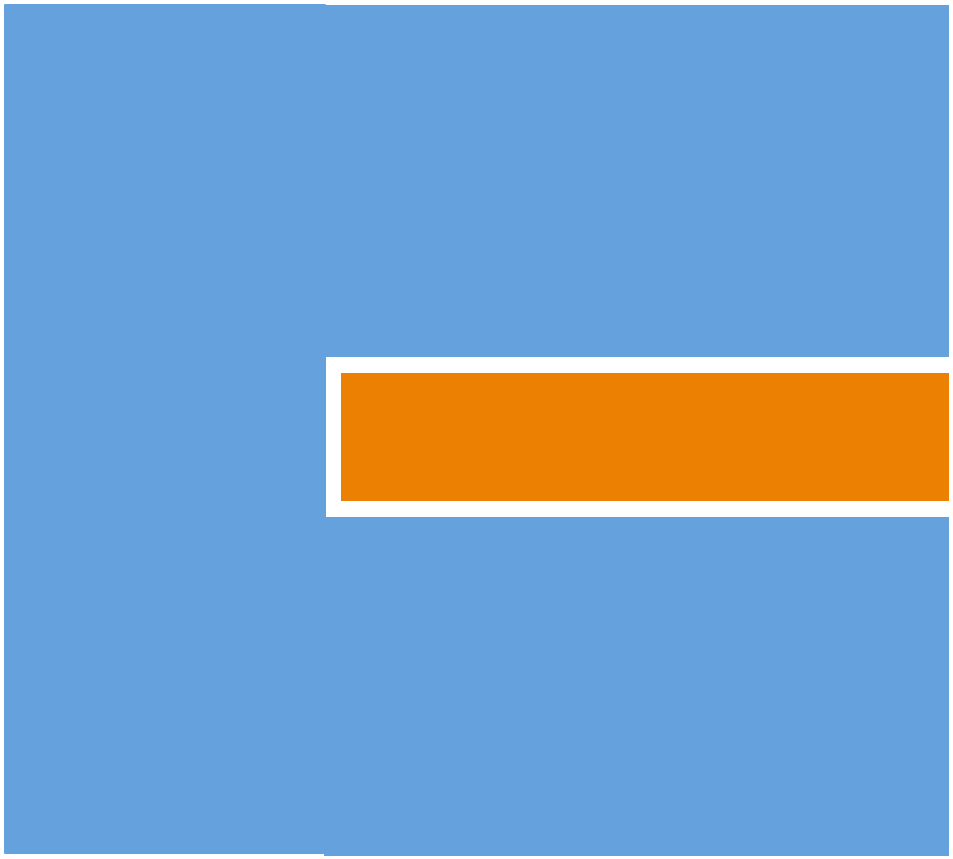
Basics of All Protocols



Control
Framing, Security, etc…

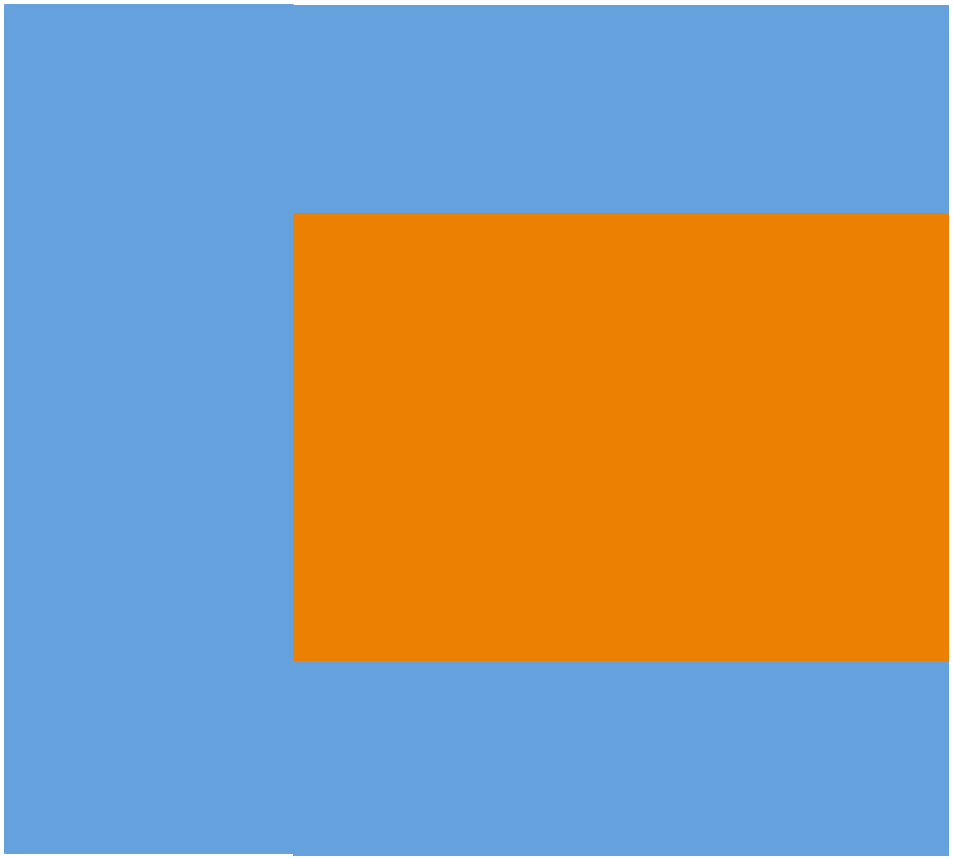
Data
What Users See

LDAP



The Control part heavily dictates the Data part.

IRIS

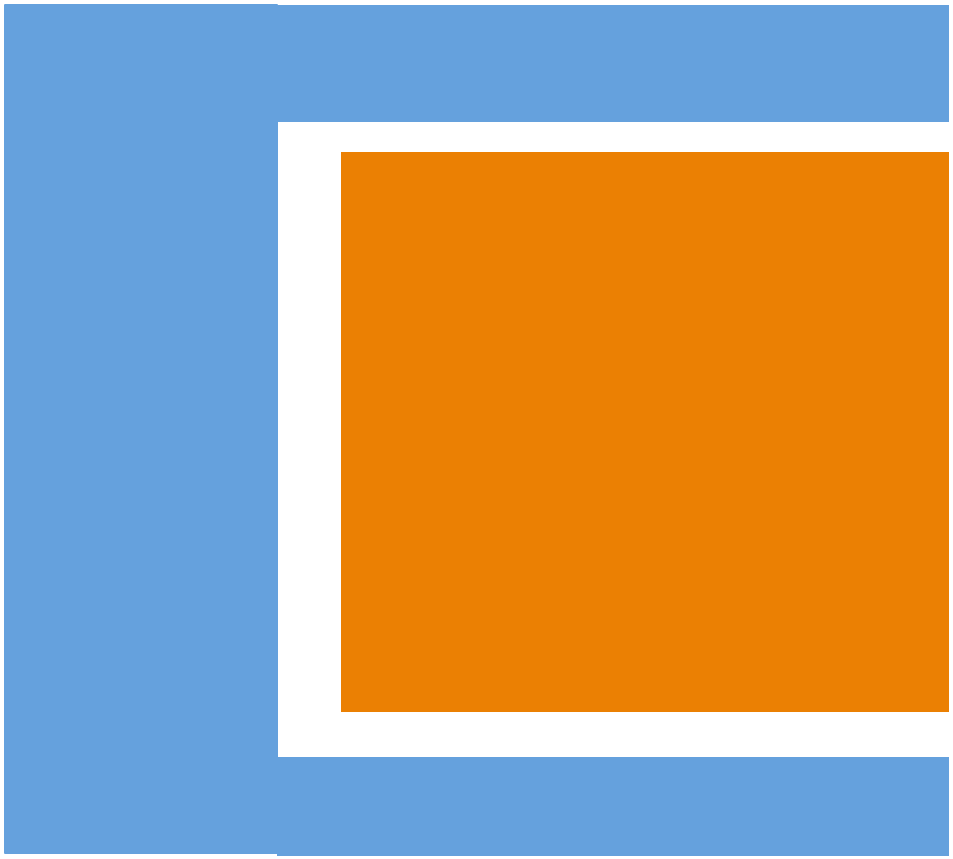


The Control part is specific to the Data part.

WHOIS/NICNAME

The Control part specifies nothing about the Data part.

RESTful Web Services



The Control part enables richness in the Data part.

Why a RESTful Web Service?

- I18N support
- referrals
- security
- … see (RFC 3707)

- Leads to the following conclusion…

We Need to go beyond Port 43

The thin veneer of the NICNAME/WHOIS protocol does not allow for much expansion without a lot of work and complexity.

The Control part specifies nothing about the Data part.

Non-Port 43 Solutions

- Rwhois
  - Problem specific technology
  - Only used by a subset of the ARIN community
- Whois++
  - Focuses on distributed indexes
- LDAP
  - Widely used in Intranets, not the Internet
- IRIS
  - Requirements by lawyers, design by committee
- ***RESTful Web Services (RWS)***
  - ***Simple reuse of web technologies***

What is REST?

- Representation State Transfer
- As applied to web services
  - defines a pattern of usage with HTTP to create, read, update, and delete (CRUD) data
  - "Resources" are addressable in URLs
- Very popular protocol model
  - Amazon S3, Yahoo & Google services, …

How is this Useful to WHOIS?

- POC, ORG, NET, ASN resources have URLs that you can cut & paste
- Gives a very simple programmatic API into WHOIS data
- Compared to NICNAME TCP/43:
  - Better inputs and queries
  - More meaningful array of outputs
- Uses HTTP infrastructure (e.g. caches)

Where can more information on REST be found?

- *RESTful Web Services*
  - O'Reilly Media
  - Leonard Richardson
  - Sam Ruby

Applicability to ICANN Whois

- This is a "framework" useful to ICANN/ Registries/Registrars
  - Not an out-of-the-box solution
  - Somebody has to decide how it is used
- But…
  - Well within the mainstream of modern Internet communications (i.e. not hard to find programmers who understand it)
  - As the RIRs are showing, it is easy to apply to the Internet Infrastructure space

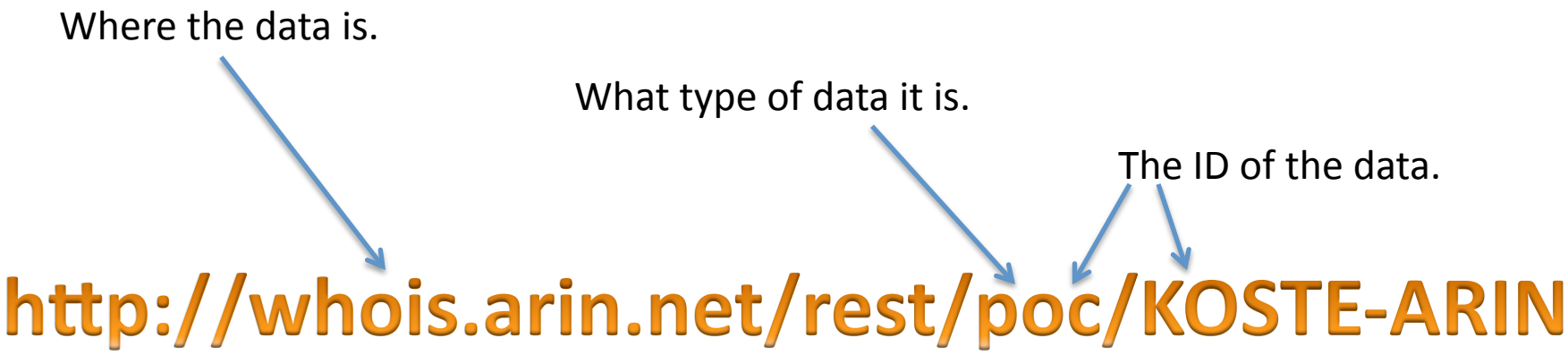
Status of Services

- ARIN
  - Full Production as of July 2010
  - Our RESTful provisioning service goes operational in a few days
- RIPE NCC
  - Announced their RESTful proxy to Whois March 2010
  - Now in production
- APNIC
  - Has been using RESTful services internally for years

The BIG Advantage of REST

- Easily understood
  - Any modern programmer can incorporate it
  - Can look like web pages
- Re-uses HTTP in a simple manner
  - Many, many clients
  - Other HTTP advantages
- This is why it is very, very popular with Google, Amazon, Yahoo, Twitter, Facebook, YouTube, Flickr, …

What does it look like?



It is a standard URL.

Go ahead, put it into your browser.

Addressable URLs

- Mark Kosters
  http://whois.arin.net/rest/poc/KOSTE-ARIN
- ARIN (the organization)
  http://whois.arin.net/rest/org/ARIN
- ARIN's autonomous systems numbers
  http://whois.arin.net/rest/org/ARIN/asns
- ARIN's POCs
  http://whois.arin.net/rest/org/ARIN/pocs
- ARIN-HOSTMASTER's networks
  http://whois.arin.net/rest/poc/ARIN-HOSTMASTER/nets

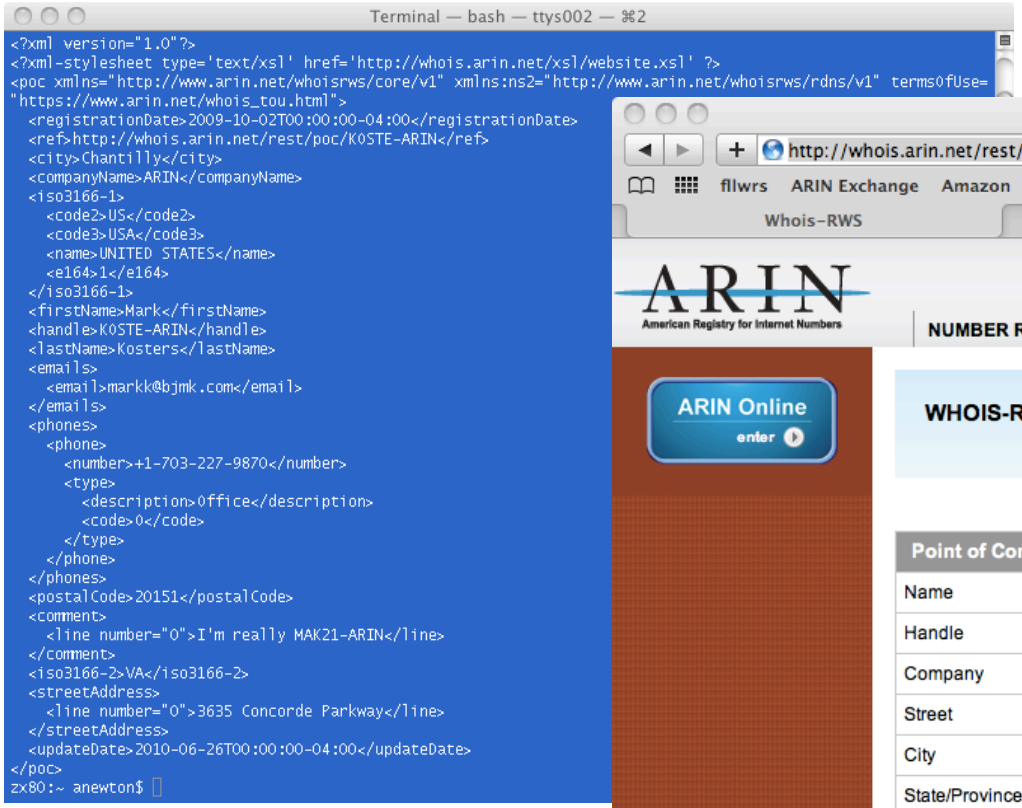
Searches

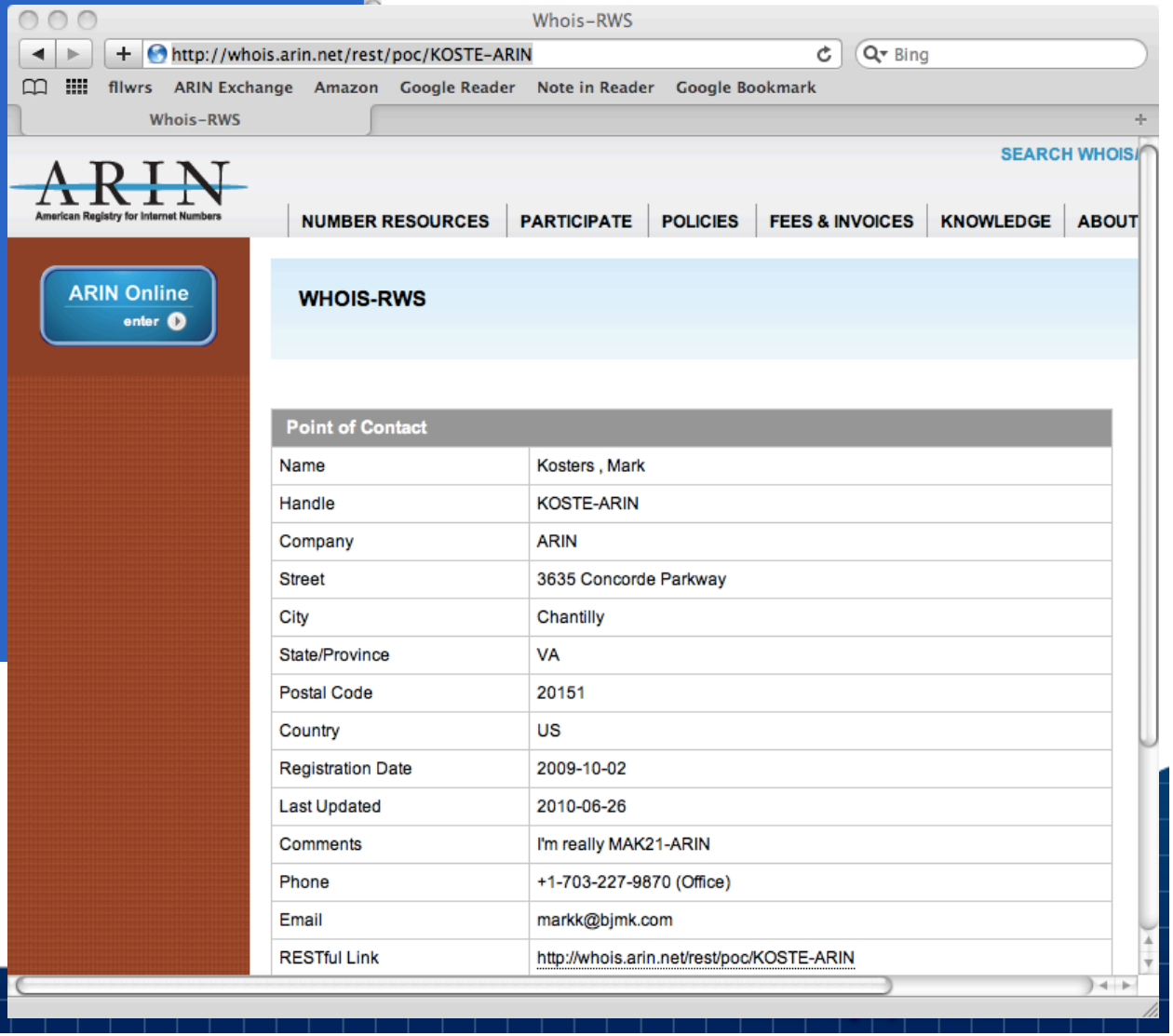
- Same capabilities as port 43, but they can be refined
- Organizations by name
  http://whois.arin.net/rest/orgs;name=ARIN
- Organizations starting with "ARIN"
  http://whois.arin.net/rest/orgs;name=ARIN*
- Mark Kosters by first and last name
  http://whois.arin.net/rest/pocs;first=Mark;last=Kosters

Outputs

- XML
  - Computers can easily digest XML
  - With stylesheets, you can transform XML to pretty, user-friendly web pages
- JSON
- (your choice here)

Machine Readable & Pretty





Clients are Ubiquitous

- One of the problems with a non-port 43 solution is **"Who will write the client software?"**
- With RWS, your web browser is a client.
- Command line clients:
  - Curl, wget, xmllint, etc…
- Embedded clients:
  - Libcurl, libraries for Perl, PHP, Java, etc…

Applying RESTful Web Services to Whois simply re-uses all the web infrastructure we have been using for years.

The Future Enabled: Caching

- Addressable URLs make HTTP caching work with WHOIS data
- Useful for automated security analysis

- For ARIN, 99% of WHOIS queries are IP address lookups

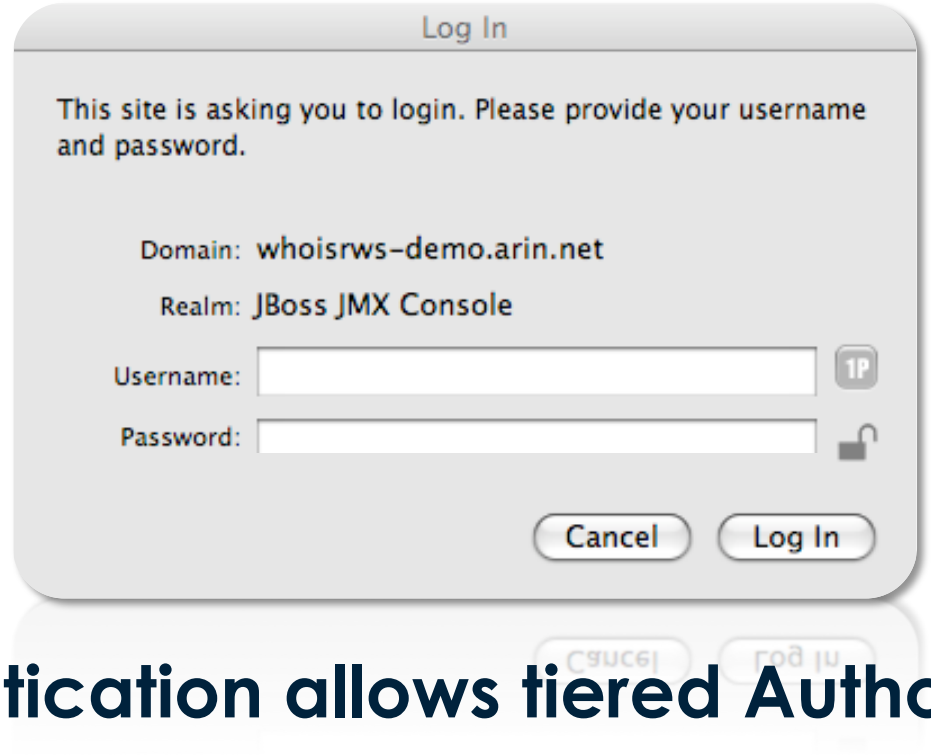
The Future Enabled: Referrals

```
NetName:    ARIN-2610
NetHandle:  NET6-2610-1
NetType:    Allocated to Big Network Provider
RegDate:    2005-11-17
Updated:    2009-09-14
CIDR:       2620::/23
Ref:        http://arin.net/rest/net/NET6-2610-1
HolderRef:  http://example.com/rest/net/NET6-2610-1
```

- Not just for Orgs
  - Nets
  - POCs
  - Etc…

```
NetHandle:  NET6-2610-1
NetType:    Allocated to Customer
CIDR:       2620::/23
Comment:    Problems?
Commnet:    Contact our NOC +1-202-555-1212
```

The Future Enabled: Auth*



- Authentication allows tiered Authorization
  - Policies no longer need to assume all or nothing

The Future Enabled: Versioning

- With standard HTTP headers, we can version our output
  - Changes the data model with as little disruption as possible

```
GET /rest/poc/DUDE1-ARIN HTTP/1.0
Accept: application/arin.whoisrws-v1+xml
```

  - You always get the latest if you don't specify

What Would It Take?

- Just saying "do RWS" is not enough.
- A "standard" is needed
  - Define the proper URL patterns
  - Define extensible output
    - DREG could be used as a starting point
    - Make it more flexible
    - Switch to RelaxNG or other schema language
  - Define pattern for referrals
    - HTTP referrals and/or embedded links

REST is Easy

- Re-use the web technologies
- Define patterns
- Use definitions you already have

- *Done!*

What We Set Out To Do

- ARIN's problem wasn't with Whois.
- We needed a fundamental change in our data model to accommodate better zone delegation management and DNSSEC in the reverse DNS.
  - Our legacy Whois would need to be completely rewritten.
  - So we felt if it had to be completely rewritten, then we should do more than a simple rewrite.

Technical Approach

- We wanted to reuse our new web infrastructure
  - Original RWS technology demonstrator took me a couple of weeks to implement as a side project
- Our Port 43 server would be a proxy into the RWS

Level of Effort

- Once greenlighted
  - Single senior developer for a couple of months
  - Then a small team for a couple of months
- But we had non-protocol problems to solve as well
  - Added in a new near-realtime replication system
  - Developed geometric search system to add CIDR search capabilities
  - Non-trivial amount of time was spent trying to make Port 43 service as backward compatible as possible

Adoption

- Before we even got to production…
  - Several people started using the public pilot to reconcile their records with ours programmatically
  - Somebody wrote a Flash application against our service
- After release…
  - Hard to tell what real adoption of RWS is because right after release our query rates skyrocketed
  - We added a psuedo-resource called "PFT" to help web browser users
    - Our previous stats indicated little use of port 80 Whois
  - Demand now for a RESTful provisioning interface… to be released real soon now

Conclusion

- ARIN's Whois-RWS:
  - http://whois.arin.net
  - Technical questions for all our services can be directed to arin-tech-discuss@arin.net.

- Q&A