Registry Architectures
What's a (ccTLD) registry?

- Publishes one or more zones (think TLD and SLD)
- Manages delegations
- Publishes public (!) information (WHOIS)
- Possibly, receives payment for the service
Data flows

• Inputs
  – Domain Name System requests
  – Creation/deletion/modification requests of domain names
  – Add nameservers (name + IP)
  – Administrative information (registrant, tech contact, billing contact, ...)

Data flows

• Output:
  – Answer DNS queries
  – Zones with delegations (publication)
  – Glue records (for nameservers which are within the zone being delegated)
  – Publication of WHOIS
Architecture

WHOIS

DNS

DNS

DNS

REGISTRY

DB
External interfaces

- **WHOIS client**: WHOIS – TCP port 43
- **resolver**: DNS – TCP/UDP port 53
- **WHOIS**: (Available)
- **DNS**: (Available)
- **DB**: (Available)
- **REGISTRY**: (Available)
- **Registrar**: (Available)
- **EPP, WebService, mail, ...**: (Available)
What operations?

- Add and remove records (redelegation)
- Add/modify/remove nameservers (modification is effectively a redelegation)
- Update of administrative data (whois info)
How complex can it get?

- It can be as simple as a text zone file with comments in it
- Maintained with Ten Finger Interface

```plaintext
; SomeCompany
; contact John Dough, +1 123 123 4567, ;
john@somecompany.mytld

somecompany    NS      ns1.othertld.org.
               NS      ns.somecompany
ns.somecompany  A       1.2.3.4
```

...
Pretty simple operational model

- Add a delegation
  - Creation of domain
- Change a delegation
  - Domain dedelegation
- Remove a delegation
  - Domain destruction

- Every operation can impact delegation entries, glue records, whois data
Terminology

• "Registry" - Institution or organisation which maintains the zone and administrative data
• "Registrant" - Physical or moral person which is responsible for a domain name
• "Registrar" - Organisation managing domain registrations on behalf of registrants
Different models: 2R

• Simple registry models – no registrars
The registrant is in direct contact with the registry. This is also called a "single access" registry.
Different models: 2R

- It remains a single access registry, even though it may or may not allow resellers:
Different models: 3R

- Shared access registry
WHOIS

- Fetch meta-information about a domain, including administrative data (name, address, phone contact, ...)
- RFC 954
  - Not formally specified as a protocol
  - Output from different Registrars and Registries can look different (and often does)
Thick vs. Thin

• Indicates how the WHOIS is placed/distributed

• Depends on where the DB is located
  – Thin: .COM, .NET: administrative data are spread across the various registrars
  – Thick: .INFO – the administrative data are centralized at the Registry
Evolution of a registry

- From most simple
  - Text zone file with comments
  - Domain registration via email
  - No whois, or manually updated
  - No registrars or resellers, 2R

- To most complex
  - Relational database, Transaction, automated billing
  - WHOIS, EPP, Web interface
  - 3R with multiple registrars
  - Anycasting of DNS servers
EPP

- RFC3730
- Supercedes RRP (RFC2832)
- Extensible Provisioning Protocol
- Based on XML
- Used by an increasing number of registries and registrars
- Not all "modern" registries have adopted it yet!
Impact of policy on the registry

• This will be explained at a later stage, but it has to be taken into account that the policy will dictate many operational aspects of your registry:
  – Registration policy – conflict resolution – authorized names – 2R/3R – data privacy – automated renewal or expiration, etc...

• The evolution of the registry if often tied to technical as well as policy aspects