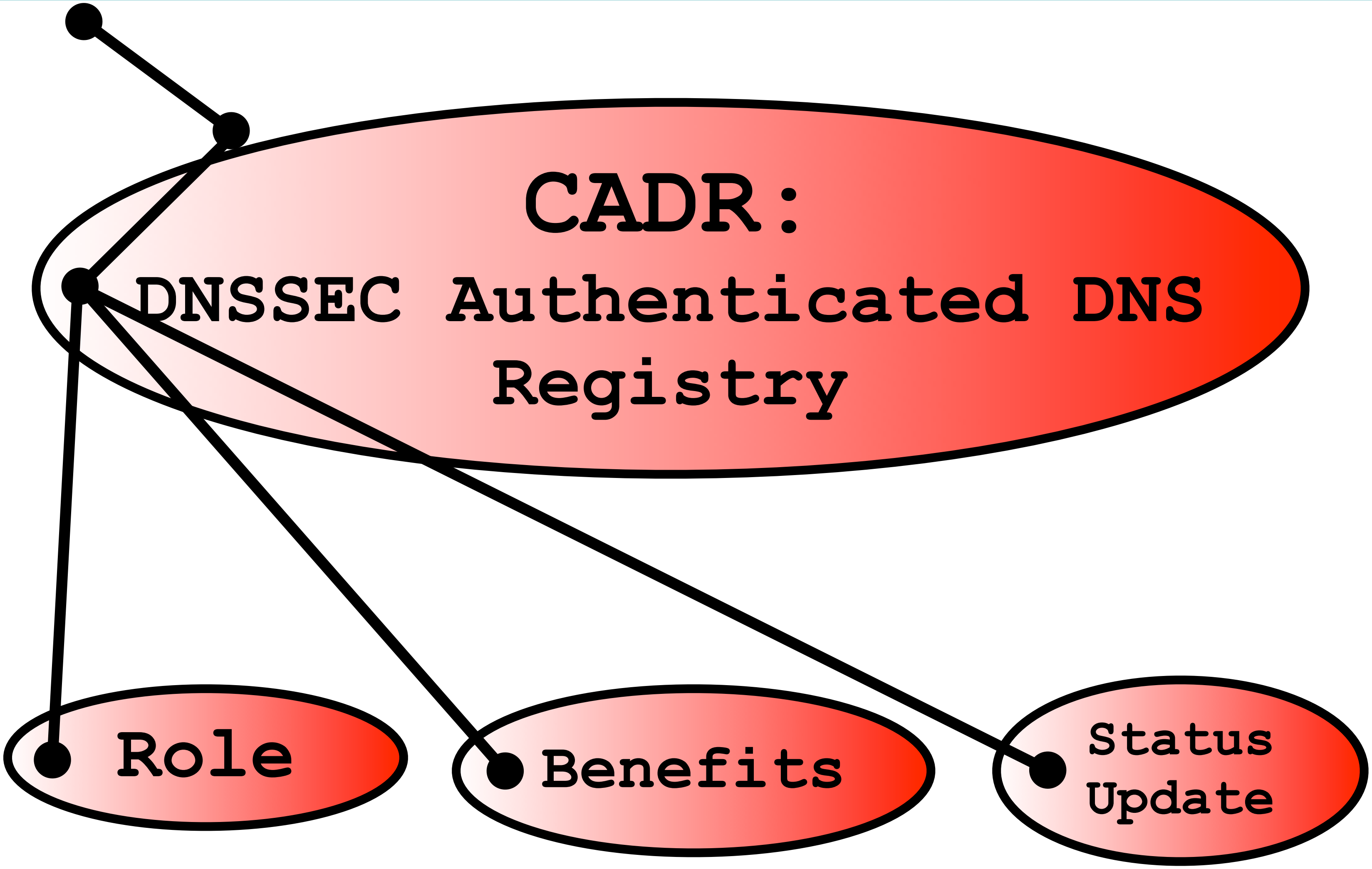


CADR: DNSSEC Authenticated DNS Registry

Role

Benefits

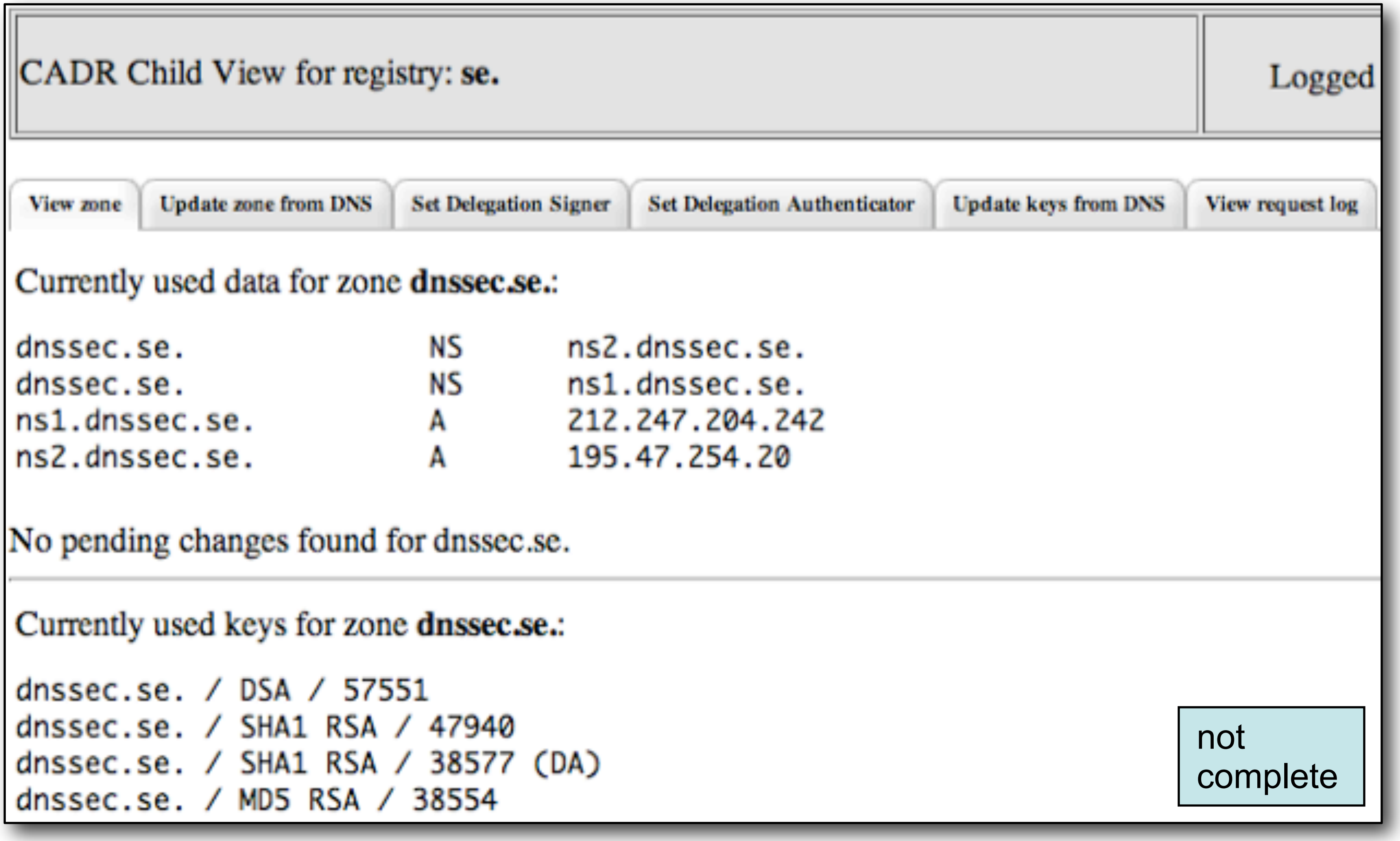
Status Update

johani@axfr.net and **bmanning@ep.net**

What is CADR?

- CADR is a registry for DNS data
  - i.e. CADR is a tool in the same ball park as registries run by TLDs to manage delegation information
  - or, in some environments, run by registrars to manage delegation information for customers for further propagation to a registry (typically for a TLD)
- CADR differs from other registries by utilizing the in-band authentication of DNS data provided by DNSSEC
  - this enables a new level of simplicity in the management of the parent-child relation at a zone cut (aka a delegation point)
  - i.e. CADR is leveraging from DNSSEC to make the registry **simpler**

A CADR Screenshot



CADR Child View for registry: **se.**

Logged

Currently used data for zone **dnssec.se.**:

```
dnssec.se.          NS      ns2.dnssec.se.
dnssec.se.          NS      ns1.dnssec.se.
ns1.dnssec.se.      A       212.247.204.242
ns2.dnssec.se.      A       195.47.254.20
```

No pending changes found for dnssec.se.

Currently used keys for zone **dnssec.se.**:

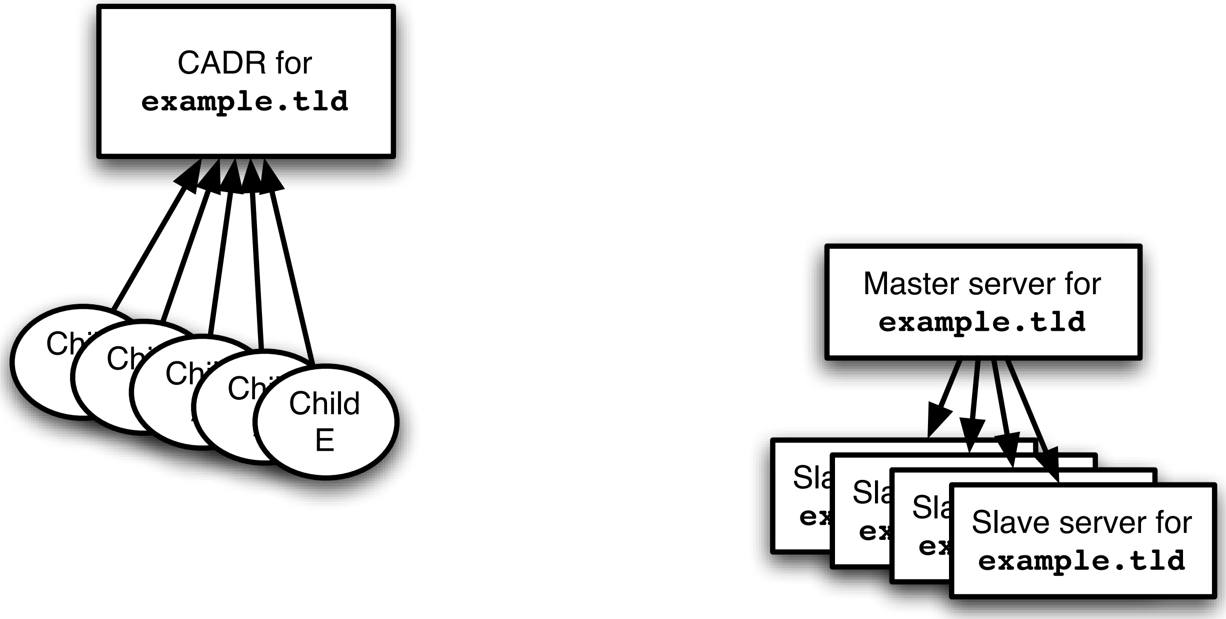
```
dnssec.se. / DSA / 57551
dnssec.se. / SHA1 RSA / 47940
dnssec.se. / SHA1 RSA / 38577 (DA)
dnssec.se. / MD5 RSA / 38554
```

not complete

Why CADR?

- We believe that with DNSSEC the complexity of managing a zone, especially a zone with children, will be daunting enough that people will move away from the model of "flat text file" over to some sort of DNS management system
  - if there are delegations such systems are usually called "registries"
- I.e. we see a need for "registries" not only on the TLD level (where we already have them), but also further down
  - if we just get the software right then running a registry for "**`example.tld`**" should be **easier** than managing it via a plain text file and an editor

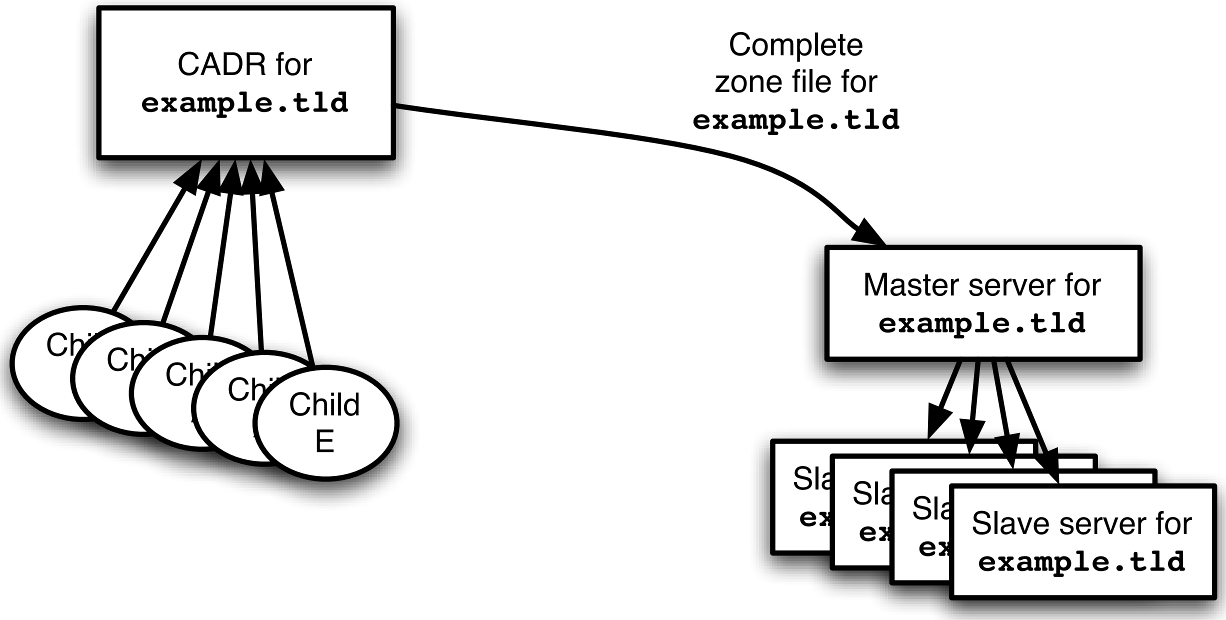
What is the role of CADR?



$Id: cadr-vs-server.graffle,v 1.1 2005/09/04 09:56:52 johani Exp $

- Given that children can update their delegation information in the CADR registry, how should this update be communicated to the parent nameservers?

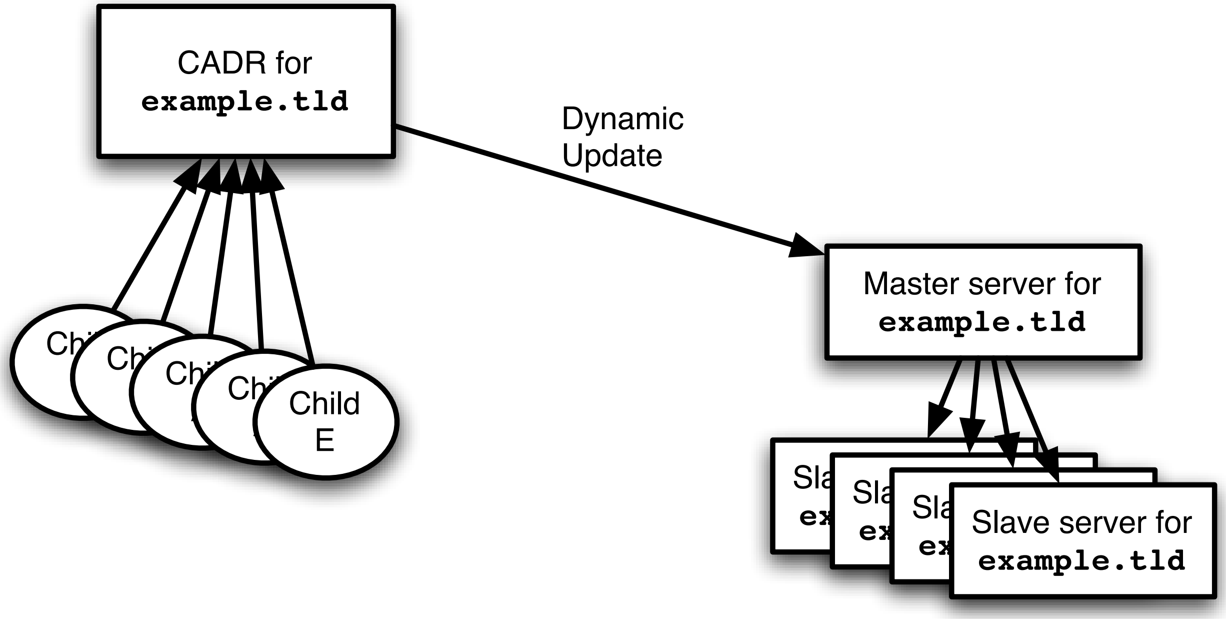
What is the role of CADR?



$Id: cadr-vs-server.graffle,v 1.1 2005/09/04 09:56:52 johani Exp $

- One alternative (the most obvious one perhaps) is to just export the entire zone file.

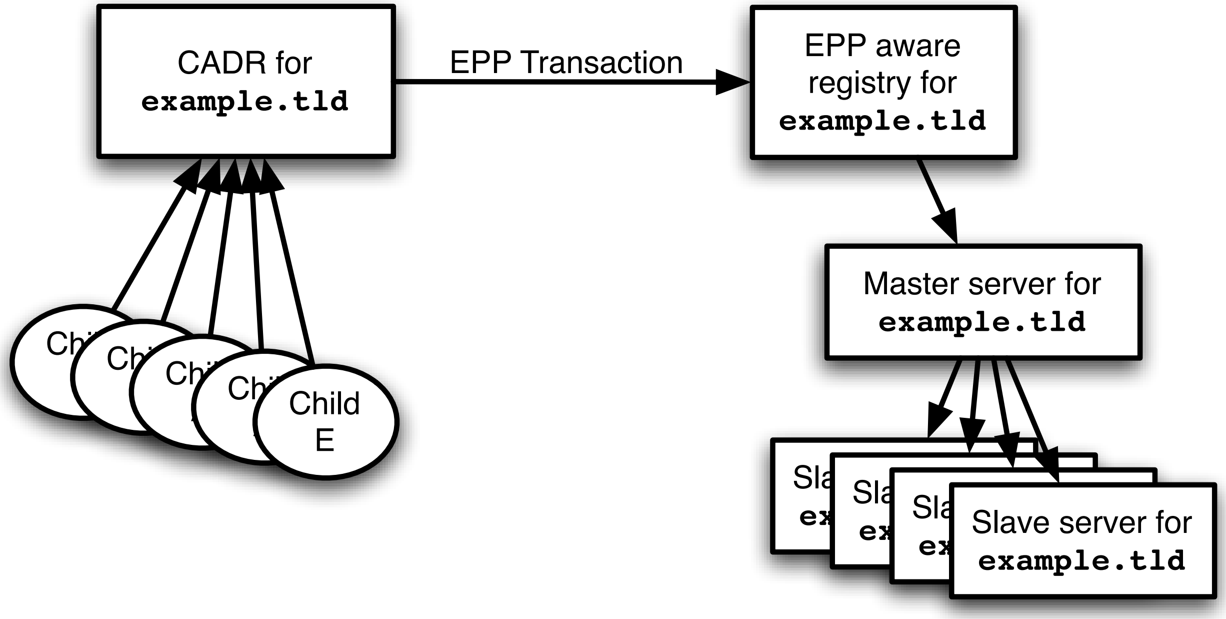
What is the role of CADR?



`$Id: cadr-vs-server.graffle,v 1.1 2005/09/04 09:56:52 johani Exp $`

- Another alternative is that the CADR registry sends a (secure) dynamic update to the nameserver infrastructure
  - there are pros and cons of this, but it **is one** of the possibilities

What is the role of CADR?



$Id: cadr-vs-server.graffle,v 1.1 2005/09/04 09:56:52 johani Exp $

- The final alternative is to communicate the update from CADR to an EPP aware registry for the parent zone
  - most relevant for the registrant -- registrar -- registry model of many TLD zones

Benefits of the CADR Model?

- “Traditional DNS” is very often misconfigured in various ways.
    - one of the most common sources of errors is the management of the delegation information for a child zone in the parent zone
    - typical stats indicate that around 15-25% of the delegations are more or less broken in this area
    - major causes of the problems are
        - entry of same information in multiple places (both child zone and parent zone)
        - authentication of child to parent for changes is complicated

“Synchronize Parent!”

- The reason for entering the same information in both parent and child (instead of just copying when needed) is the absence of proof of the integrity of the data
  - i.e. the parent could easily look up the delegation information for the child in the public DNS, but it cannot **trust** the information to be correct
  - this assumption no longer holds true when we deploy DNSSEC
- With DNSSEC it is suddenly possible to prove (to the parent) that the information about the child in the public DNS is authentic and can be depended upon directly
  - this enables us to switch to the new model “synchronize parent” (i.e. in-band copying of delegation data from child to parent)

Goal of this Project

- The goal is to contribute a complete CADR system to the community as open source (hopefully soon after Christmas 2006)
  - we're not quite there yet, more work is needed to polish this off as a piece of software useful to others

Current and Next Steps

- CADR is a powerful registry system with significant functionality that is either complete or underway. Some examples:
  - “Host Records” (i.e. a way to manage out-of-zone glue in a CADR context)

    ONGOING WORK
  - EPP backend

    DEFERRED
  - Dynamic Update backend

    COMPLETE
  - Ability to manage several “parent zones” in one CADR

    COMPLETE
  - Parent key management, including full rollover support and time-scale semantics

    COMPLETE
  - Generation of signed zones

    COMPLETE

Other Features Of Interest

- Dynamic creation of new registries in "same CADR".
  - i.e. if you already manage "**`frobozz.com`**" and "**`gnark.net`**" in CADR then it is just a few buttons to push to add the new parent domain "**`flodhäst.se`**" to the collection

COMPLETE

- Dynamic creation (and destruction) of admin users

COMPLETE

- Access control lists per user and zone.

COMPLETE

- Complete request log for all change requests
  - including back-and-forth between child and parent
  - with ability to add free-form comments in both ends

COMPLETE

- Command line tool to export complete zone file for loading into favourite nameserver
  - for those that don't like Dynamic Update

COMPLETE

Demo Time (if there is time)

- The task is to make an update to a delegation beneath a demo version of "**.SE**".
  - **.SE** is being used because there are a bunch of signed children available to play with
  - i.e. while the parent is a demo version, **the children** are live, public, and real delegations from **.SE**
- CADR's task is to accept the update request, present it to the admin, wait for a commit decision and then publish the result via DDNS.

Questions?

Johan Ihrén --- **johani@axfr.net**

Bill Manning --- **bmanning@ep.net**