

SRS

Shared Registry System

Presentation to APTLD Kuala Lumpur
24 May 2008

Nick Griffin
General Manager



Outline

- ④ Shared Registry System (SRS) History
- ④ Design Goals
- ④ Business Requirements
- ④ Technical Solution
- ④ The Results
- ④ Looking forward – SRS Investment

Shared Registry System (SRS) History

- ④ 2001 - Requirements defined
- ④ 2002 - Developed
 - ④ Operational October 2002
- ④ Continuous improvement programme
 - ④ To increase reliability and stability
 - ④ Support changing registrar needs
- ④ Currently serves
 - ④ over 330,000 names
 - ④ 67 Registrars (1/3 Foreign)

Design Goals

- Support the .nz policies
- Availability
- Reliability and redundancy
- Thick registry
- Registrants rights to be protected
- Non repudiation
- All registrars are treated the same
- Open Source
- Secure data transfer - HTTPS
- Dedicated protocol and server environment.

Business Requirements

- ④ Based on .nz policy including:
 - ④ Roles and Responsibilities
 - ④ Create, Renew and Cancel
 - ④ Whois
 - ④ Domain transfer token
 - ④ Unique Domain Authentication Identifier (UDAI)
 - ④ Protects registrant
 - ④ Registry configurable elements
 - ④ All registrars are treated the same
- ④ Dispute process is outside of the SRS system.

SRS Rules

- ④ All transactions
 - ④ All fields checked for validity
 - ④ Require unique Action ID
- ④ Security
 - ④ Non Repudiation via use of Public / Private Keys (PGP)
 - ④ HTTPS
- ④ History of Names
 - ④ Full history of every update is kept
 - ④ Registrars can view history of names under their management
- ④ Replication
 - ④ Requires updates to be completed on a minimum of 2 servers

Technical Solution

- ④ Developed using Open Source software
 - ④ Debian GNU/Linux
 - ④ Perl
 - ④ PostgreSQL database
 - ④ Apache Webservers
 - ④ XML for registrar/registry interface to the SRS
 - ④ XML transferred over HTTP/HTTPS
 - ④ Commodity Hardware
 - ④ No specialised hardware required
 - ④ Low level servers
 - ④ Low level routers

Configurable items

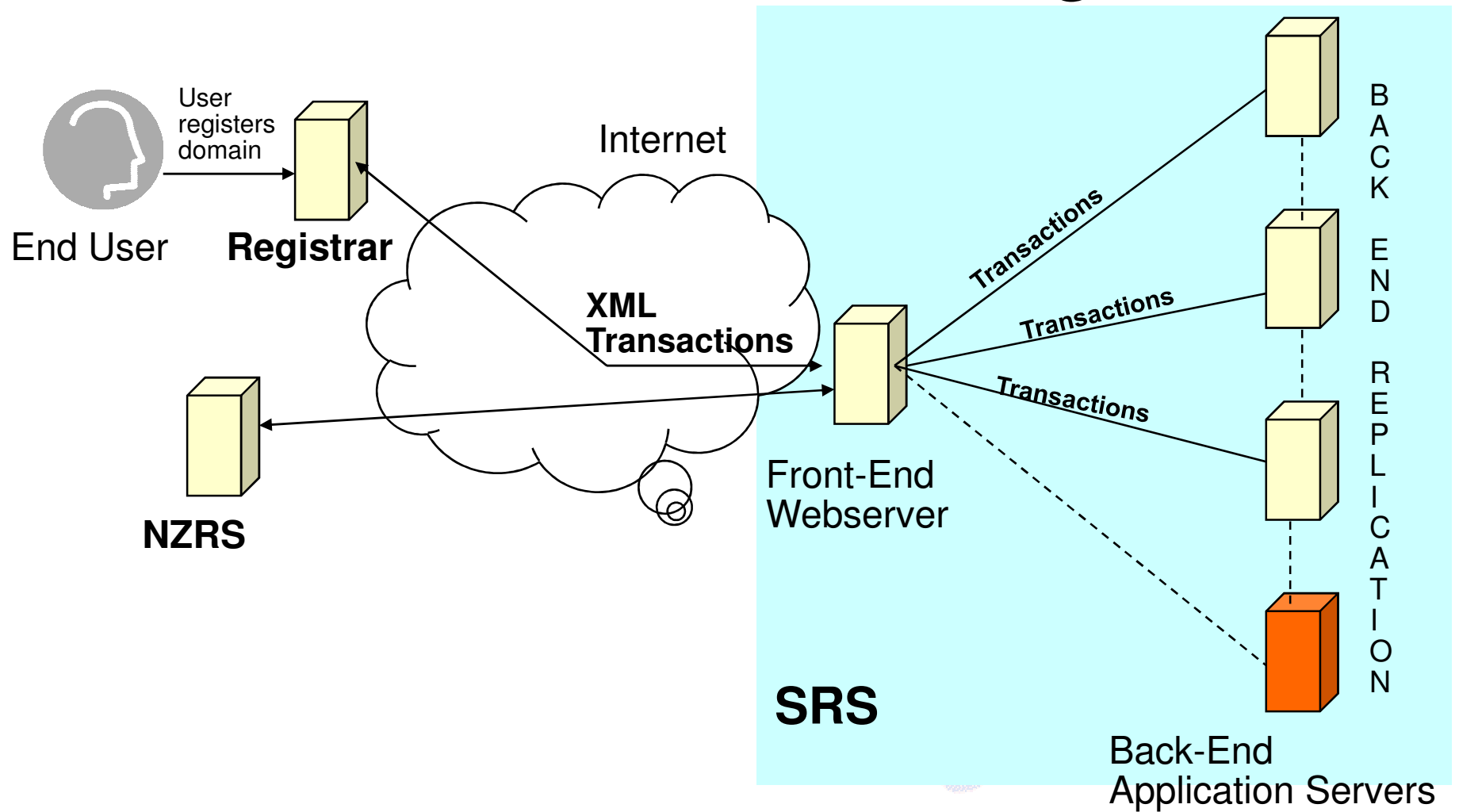
- ④ Grace periods
 - ④ Renewal
 - ④ Registration
 - ④ Pending release period (Redemption grace period)
- ④ Domain name fee
- ④ 2LD's
- ④ Access roles
- ④ Minimum / Maximum billing term
- ④ You can't register the same name > 2 times in 30 days
- ④ The maximum number DNS Glue records for a name
- ④ Maximum response size
- ④ Schedule of batch jobs

SRS Components - Outline

- ④ SRS - Multi Tier Design
 - ④ Front-end Web Server
 - ④ Multiple Back-end Application Servers
 - ④ Multiple Back-end Databases
- ④ Whois Server Component
- ④ Front End Replication
- ④ Scheduled Jobs Daemon
- ④ Registrar Implementation Kit
- ④ Recommended Architecture



SRS - Multi Tier Design



Main SRS Components

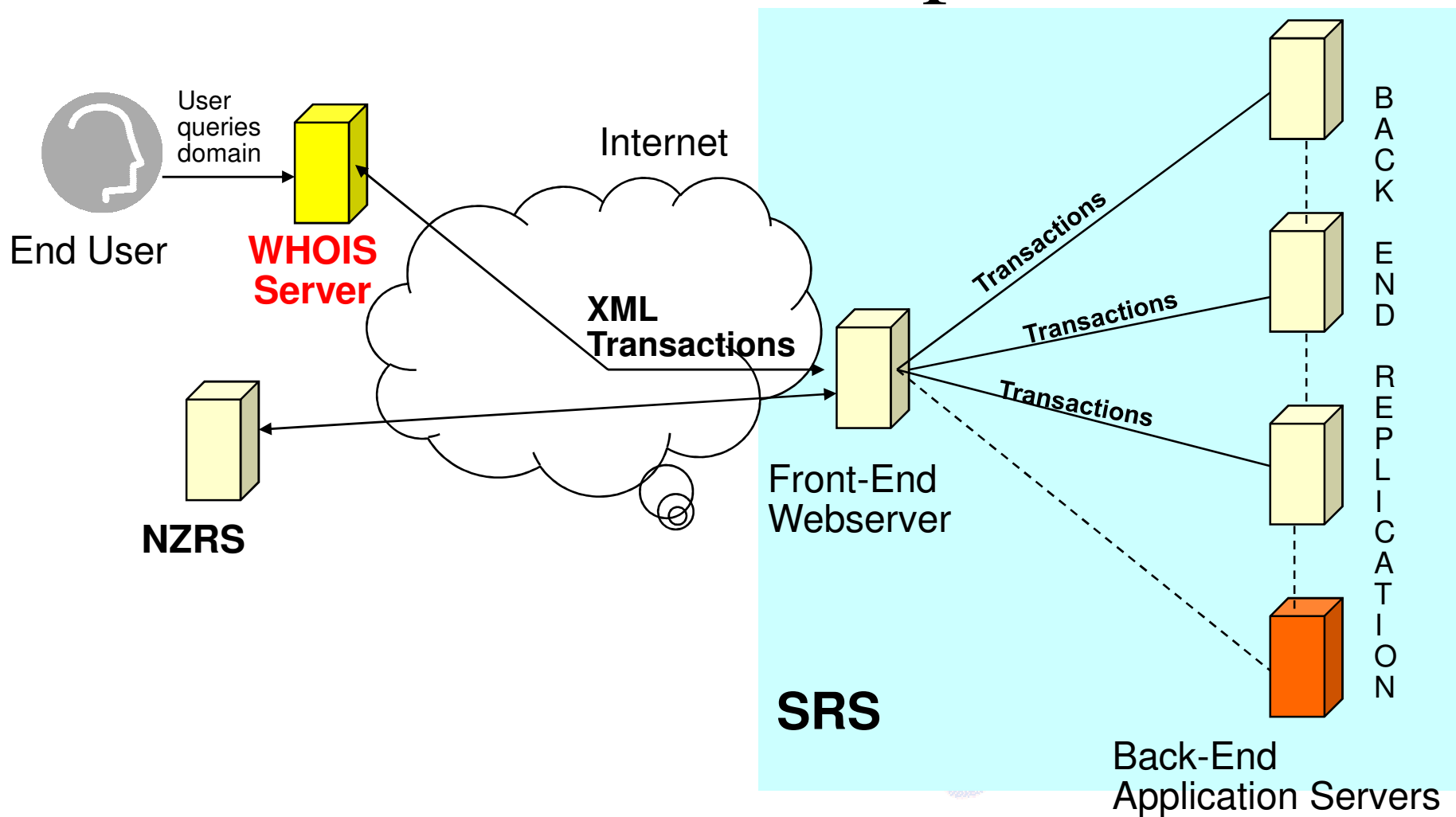
Front-End Webserver

- receives requests from the Internet, performs some basic error checking, and then passes it on to the front-end replication daemon. It then waits for a response to be returned, which it then passes back to the client.

Back-End Application Server

- This server does the actual processing of the request, including full validation, authentication and authorisation, updating the database, and creating a response.
- PostgreSQL Database

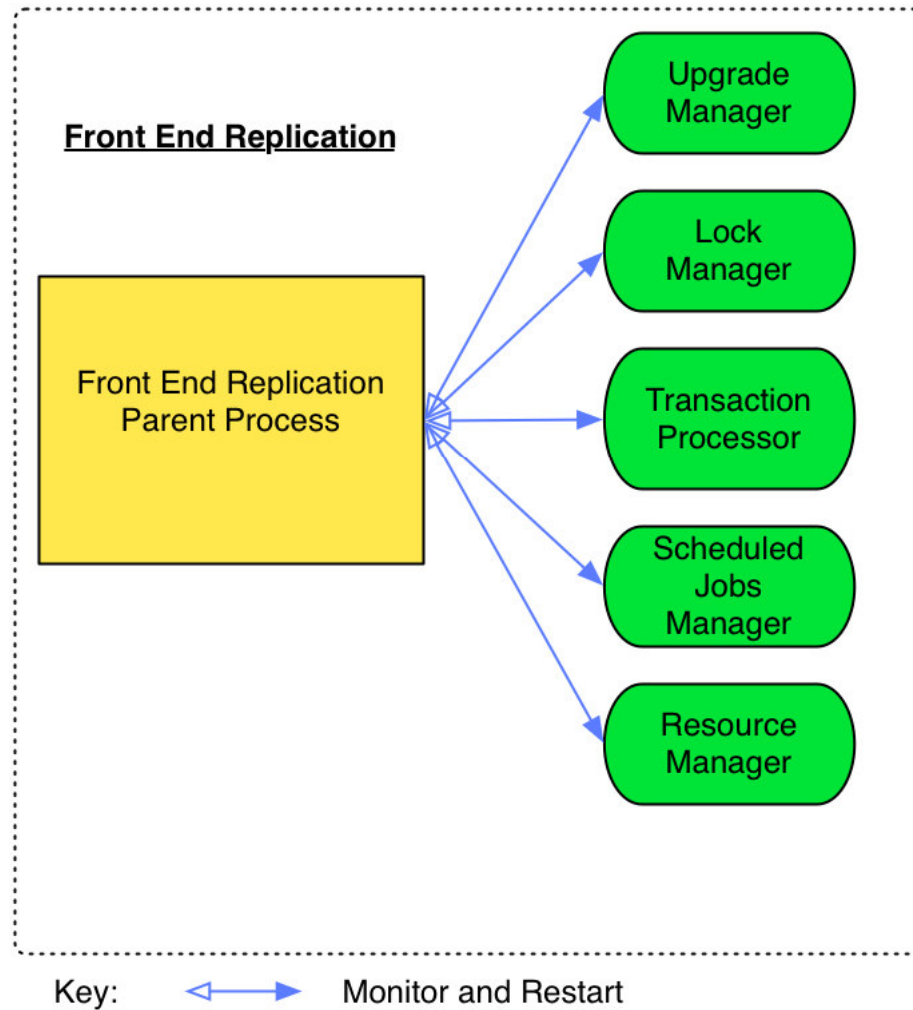
Whois Server Component



Whois Server Component

- ④ Is a pseudo registrar
- ④ Receives standard Whois requests (RFC 954)
- ④ Sends XML transactions to the SRS front-end
- ④ Waits for a response
- ④ Then converts the response and sends back to the client
- ④ Runs as a standalone daemon

Front End Replication

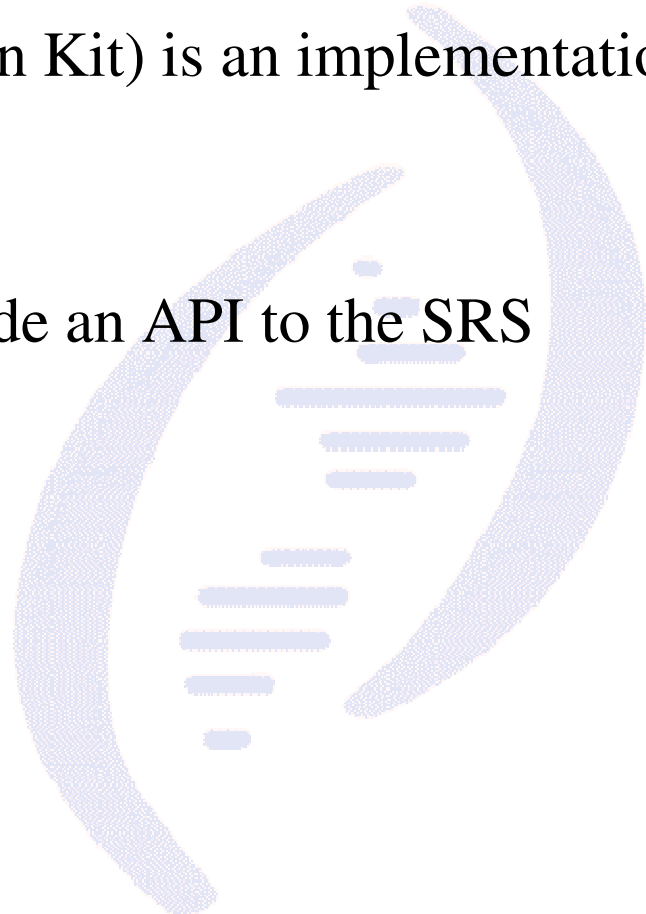


Scheduled Jobs Daemon

- It is a standalone daemon
- It runs the various scheduled jobs, including:
 - Renew Domains
 - Release Domains
 - Build DNS Zone Files
 - Registry reports
- Can run on one or more of the back-end servers

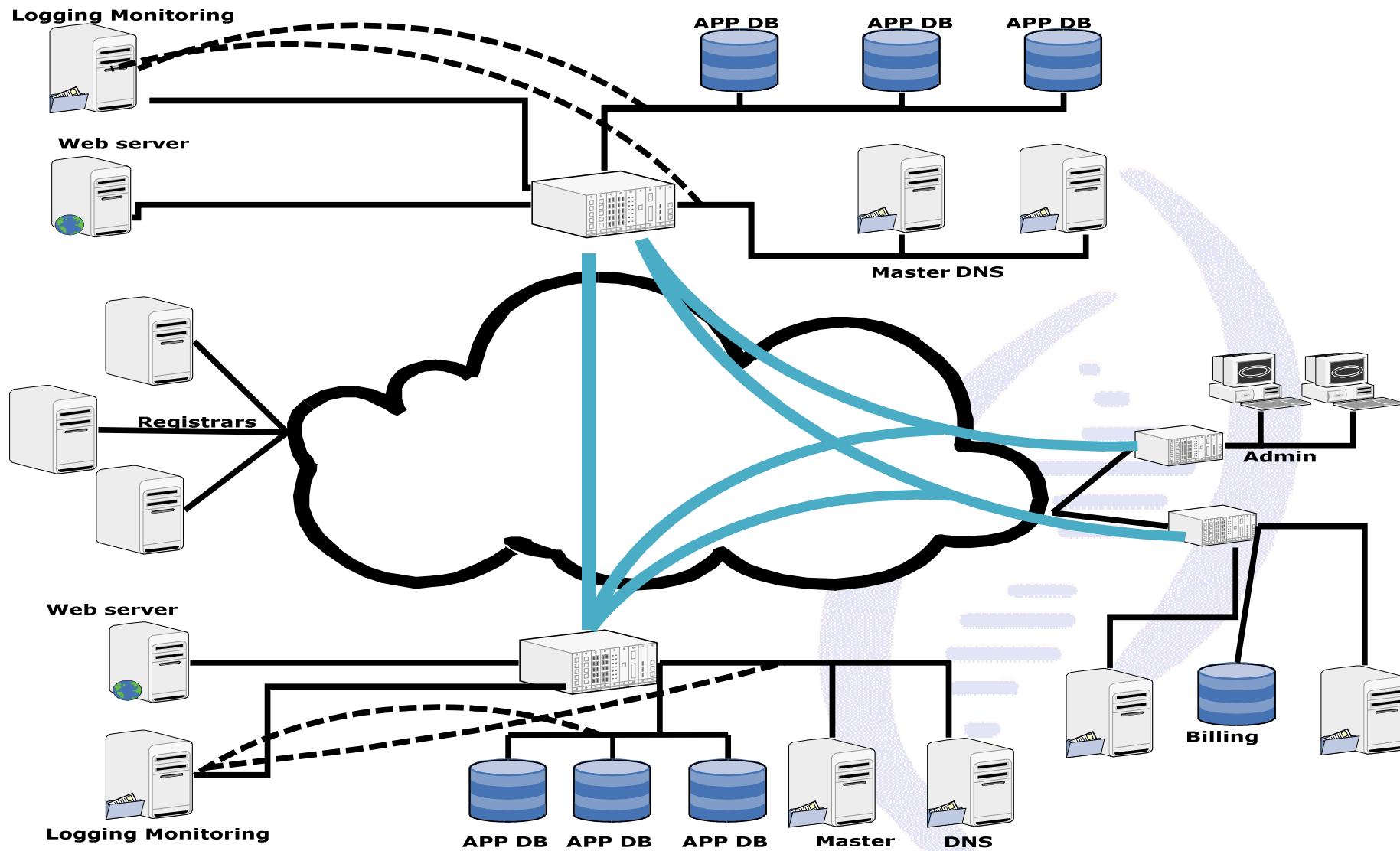
Registrar Implementation Kit

- ④ The RIK (Registrar Implementation Kit) is an implementation of a registrar's client
- ④ Includes;
 - ④ A set of Perl modules that provide an API to the SRS
 - ④ .nz Policies
 - ④ SRS documentation
 - ④ Example XML transactions
- ④ The RIK is distributed to registrars



Recommended Architecture

- ④ One server as a dedicated front-end webserver
- ④ Three or more servers as dedicated back-end application servers
- ④ Whois server run either on the front-end webserver, or on a separate machine with http access to the front-end webserver.
- ④ The back-end application servers be accessible only from the front-end.
- ④ Not possible to run multiple front-ends simultaneously. Due to race conditions occurring when updating transactions for the same domain are sent to two front-ends simultaneously.
- ④ Possible to run additional front-ends that allow only read transactions by running these front-ends in 'maintenance' mode.



24/05/2008

NZ Registry Services

Additional Notes

- ☞ Communications are stateless
 - ☞ You send an XML request and you get a response
 - ☞ No logon required (Authentication via PGP)
- ☞ Transactions are effected within a relatively short timeframe
- ☞ Guarantee of authoritativeness by comparing responses from multiple application servers
- ☞ Replication model is “multi-master” replication
 - ☞ Any back-end application server can disappear without impacting the system
- ☞ Front-end replication passes transactions to all back-end application servers in parallel, with the response passed back to the client only after we have received sufficient responses which are in agreement.

Summary of Statuses

- System status is available via a web page:
 - <http://srs1.srs.net.nz/srs/summary?type=html>

<i>Hosts</i>	
srsapp1	hot spare
srsapp2	active
srsapp4	hot spare
srssapp2	active
srssapp4	inactive
srssapp5	active
srssapp6	hot spare

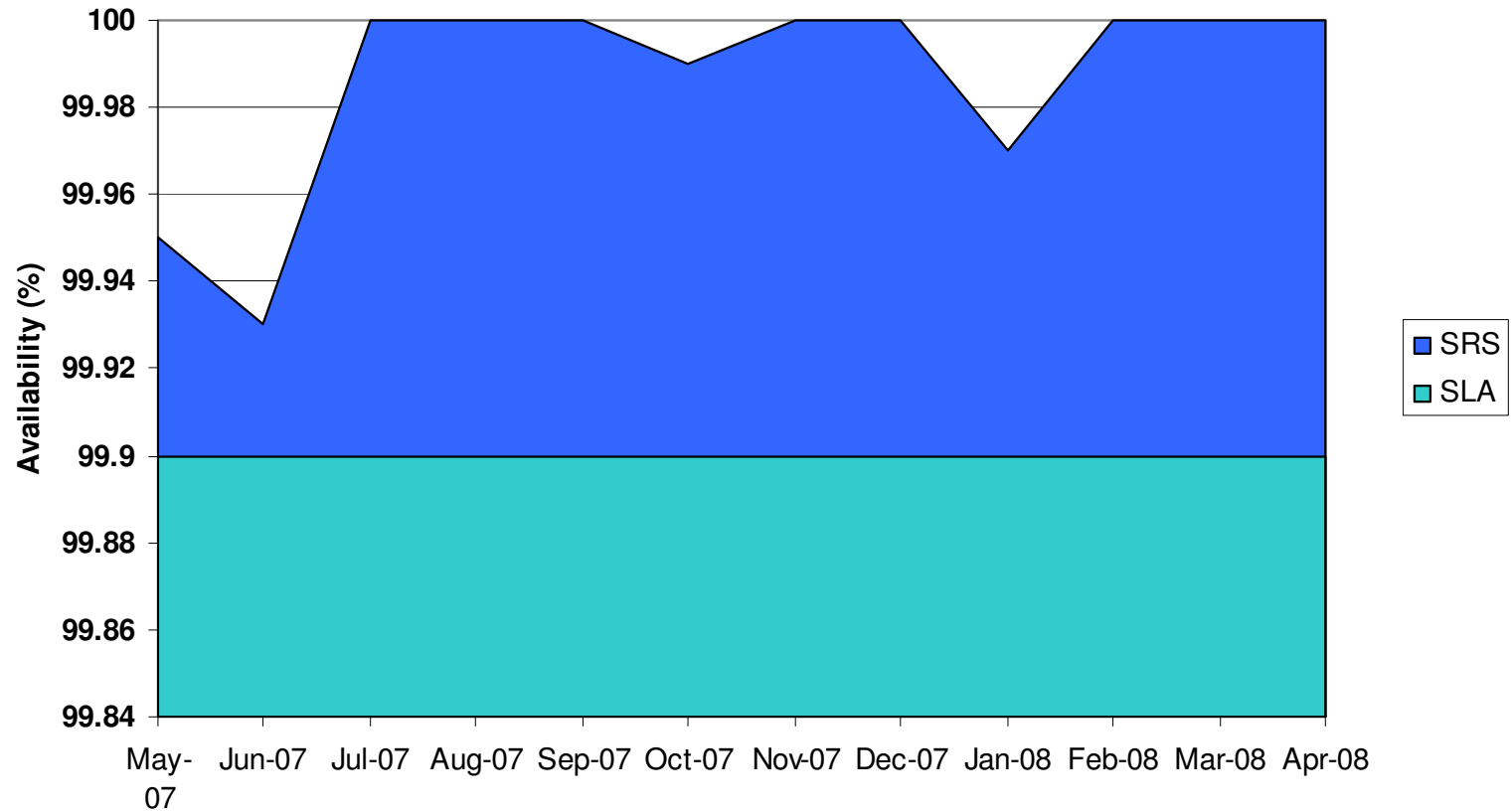
<i>General</i>	
System Mode	writable
Active FE	srssweb1
Page Generated	2008-05-22 10:35:36

The Results

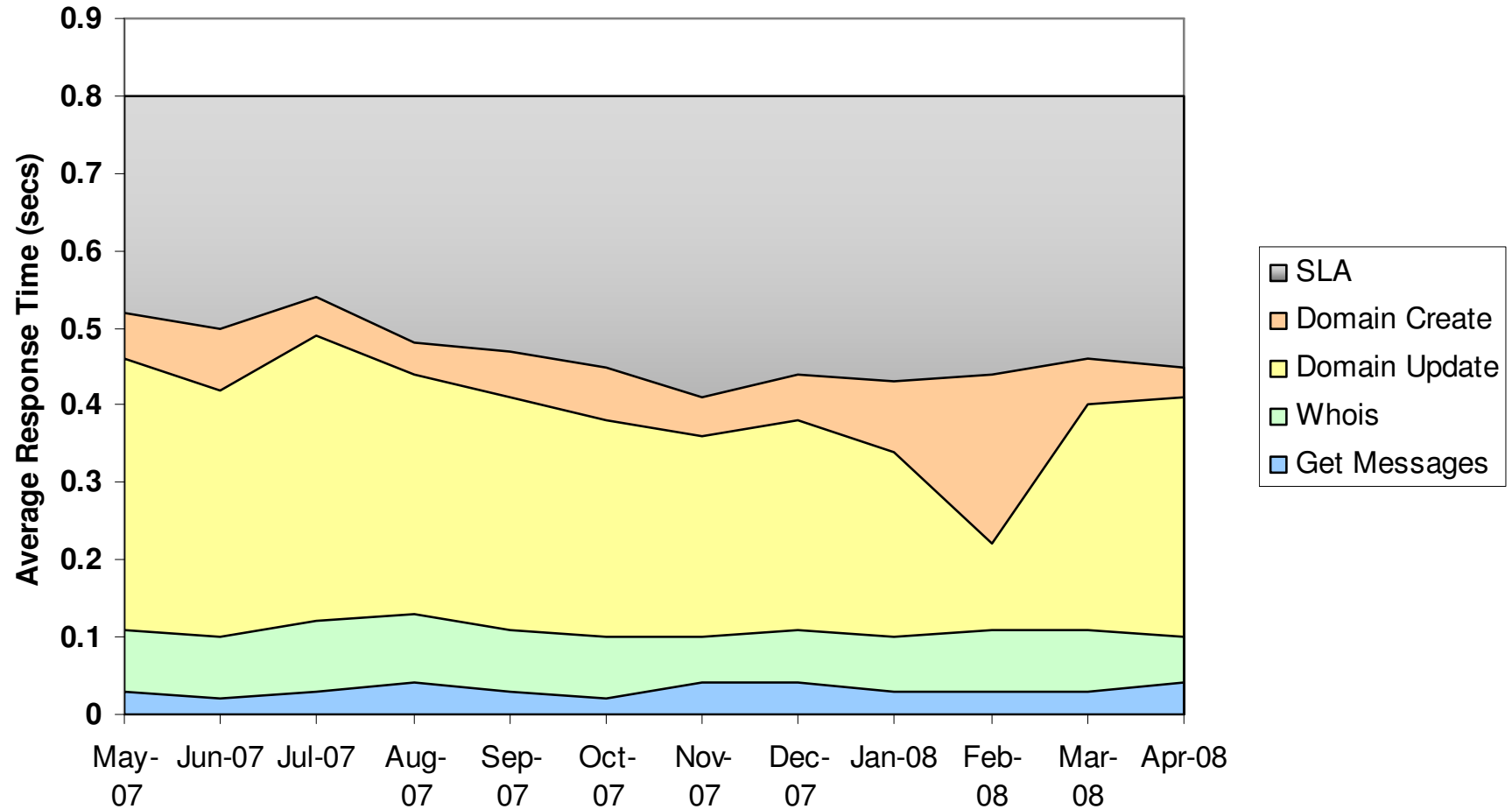
- ④ What have we achieved?
 - ④ 67 registrars
 - ④ Approx 1/3 are foreign registrars
 - ④ Registrar satisfaction of 94%
 - ④ High System Availability
 - ④ Low Systems Response times
 - ④ High Data Integrity



System Availability



System Response Times



System Availability and Response Times

	SLA %	Mar 08	Feb 08	Jan 08	Dec 07	Nov 07	Oct 07	Sep 07	Aug 07	Jul 07	Jun 07	May 07	Apr 07
SRS	99.90%	100	100	99.97	100	100	99.99	100	100	100	99.93	99.95	99.94
Whois	99.90%	100	99.99	99.97	100	100	100	100	100	99.96	99.93	99.97	99.99

Average Response time (Seconds)	SLA	Mar 08	Feb 08	Jan 08	Dec 07	Nov 07	Oct 07	Sep 07	Aug 07	July 07	Jun 07	May 07	Apr 07
Domain Details	<1.50	0.03	0.03	0.02	0.03	0.03	0.02	0.03	0.03	0.03	0.03	0.03	0.03
Domain Update	<0.80	0.40	0.22	0.34	0.38	0.36	0.38	0.41	0.44	0.49	0.42	0.46	0.40
Domain Create	<0.80	0.46	0.44	0.43	0.44	0.41	0.45	0.47	0.48	0.54	0.50	0.52	0.50
Get Messages	<0.80	0.03	0.03	0.03	0.04	0.04	0.02	0.03	0.04	0.03	0.02	0.03	0.03
Whois - SLA	<0.80	0.11	0.11	0.10	0.11	0.10	0.10	0.11	0.13	0.12	0.10	0.11	0.09
UDAI ValidQry	<0.80	0.14	0.13	0.13	0.14	0.14	0.14	0.15	0.15	0.14	0.13	0.15	0.14

Looking forward – SRS Investment

- ④ NZRS has applied for the SRS Protocol to be an RFC standard
- ④ Addition of an EPP Proxy Interface
- ④ Capacity load test of over 1 million names
- ④ DNSSEC
- ④ IDN
- ④ Enum

SRS - RFC Project

Phase one - Completed

- Produced an SRS Standards document
- Submitted to the Internet drafts process.

Phase two

- Producing an SRS Implementation document
- uses SRS Standard
- details all the .nz business rules
 - i.e. type of grace periods and specific values used

RFC submission

- ④ System for Managing a Shared Domain Registry
- ④ Submitted and posted to the IETF repository
 - ④ Filename: draft-nzrs-srs-00.txt
 - ④ Creation_date: 2008-05-18
 - ④ WG ID: Independent Submission
- ④ Comments are solicited @:
 - ④ srsstandards-1@nzrs.net.nz

RFC Draft Documents

Available @

<http://www.nzrs.net.nz/downloads/draft-nzrs-srs-00-fo.pdf>

<http://www.nzrs.net.nz/downloads/draft-nzrs-srs-00.pdf>

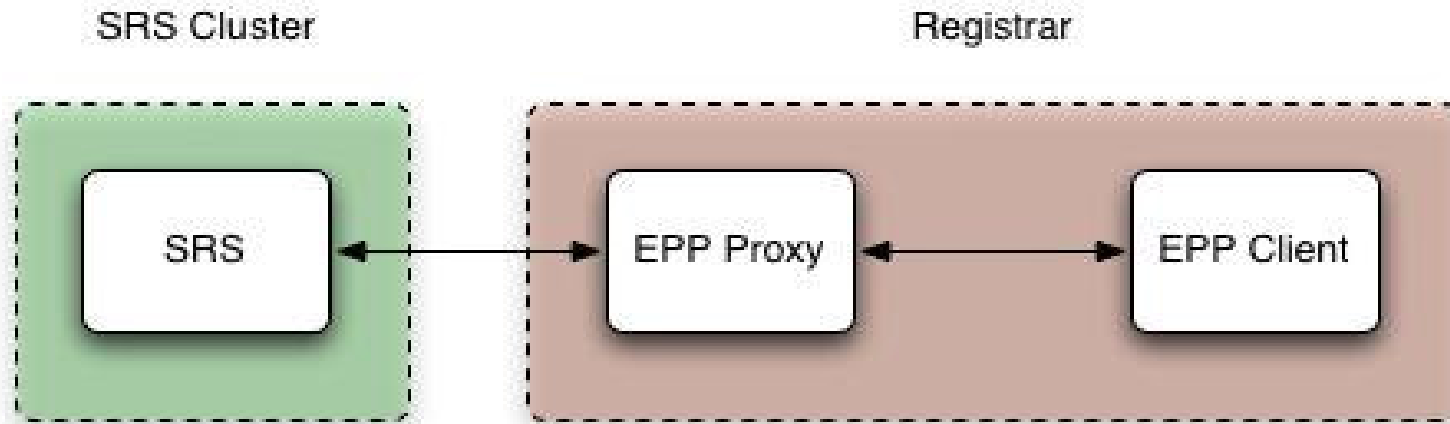
<http://www.nzrs.net.nz/downloads/draft-nzrs-srs-00.txt>

<http://www.ietf.org/internet-drafts/draft-nzrs-srs-00.txt>

EPP

Key difference

- To be implemented as a proxy to the SRS
- Registrars will run their own EPP interface which will communicate natively with the SRS



IDN and Enum

- ⌚ IDN support to be added to the SRS next year
- ⌚ Enum is being considered



Summary

- ④ In addition to Policies Registry Systems need:
 - ④ Security
 - ④ Reliability
 - ④ Reliance
- ④ The SRS provides those elements which resulting in a High Quality of Service

Further Information

- ☞ If you would like further information about the SRS please contact:
- ☞ Nick Griffin
- ☞ General Manager
- ☞ NZRS
- ☞ e nick.griffin@nzrs.net.nz
- ☞ w www.nzrs.net.nz



Thank You

Useful Links

www.nzrs.net.nz

www.dnc.org.nz

www.sourceforge.net/projects/dnrs/

www.ietf.org/internet-drafts/draft-nzrs-srs-00.txt