A Brief Introduction to JPRS DNSSEC Implementation Research for TLD

3rd DNS-OARC Workshop

Nov. 3, 2007

Japan Registry Services Co., Ltd. (JPRS)

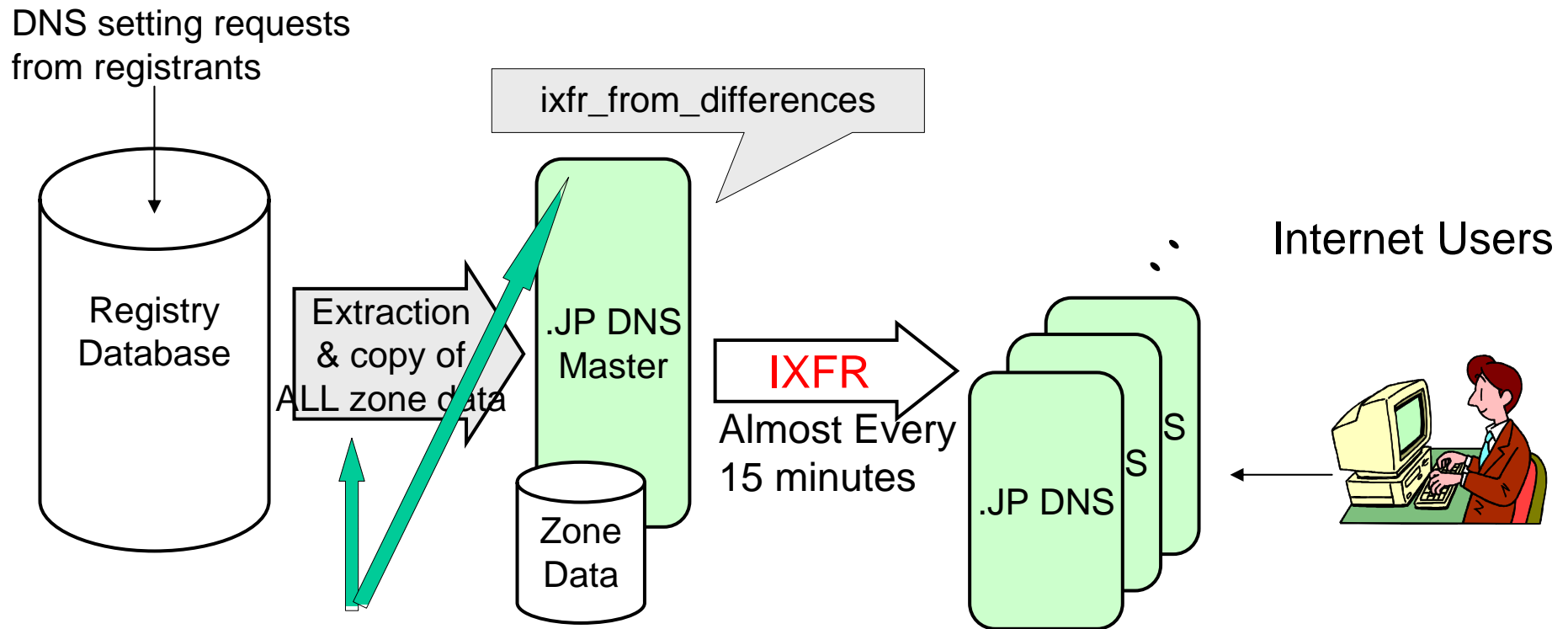Kentaro Mori <kentaro@jprs.co.jp>

Shinta Sato <shinta@jprs.co.jp>

Contents

- Background

- Requirements

- Design concepts

- Implementation

- Performance

- Field Test

Background

- ## Current status of .JP
  - A million of domain names
  - Frequent DNS updates (every 15 minutes)
  - Zone synchronization to 30 DNS servers (including IP anycast & stand-by servers)

- ## Issue regarding to implement DNSSEC
  - Zone data signing with existing tool (e.g. dnssec-signzone) takes longer time than our update interval

- ## JPRS decided to create a prototype implementation to solve this issue

Frequent DNS updates : Current .JP System

DNS setting requests
from registrants

ixfr_from_differences

Registry
Database

Extraction
& copy of
ALL zone data

.JP DNS
Master

Zone
Data

IXFR
Almost Every
15 minutes

Internet Users

.JP DNS
S
S
S

**Bottle-necks against
further update frequency &
creating DNSSEC related RRs**

Major Requirements Defined in R&D

- **Large zone** administration
  - 10 million-domain class

- **Rapid updates**
  - Data updates by every minute (on service operation)
  - 100 domains update is performed within 10 seconds

- **Reliable zone data synchronization**
  - Checking sync. delay time
  - Checking data integrity

- **DNSSEC capability**
  - Compliance to RFC4033-RFC4035
  - Compliance to both of NSEC & NSEC3
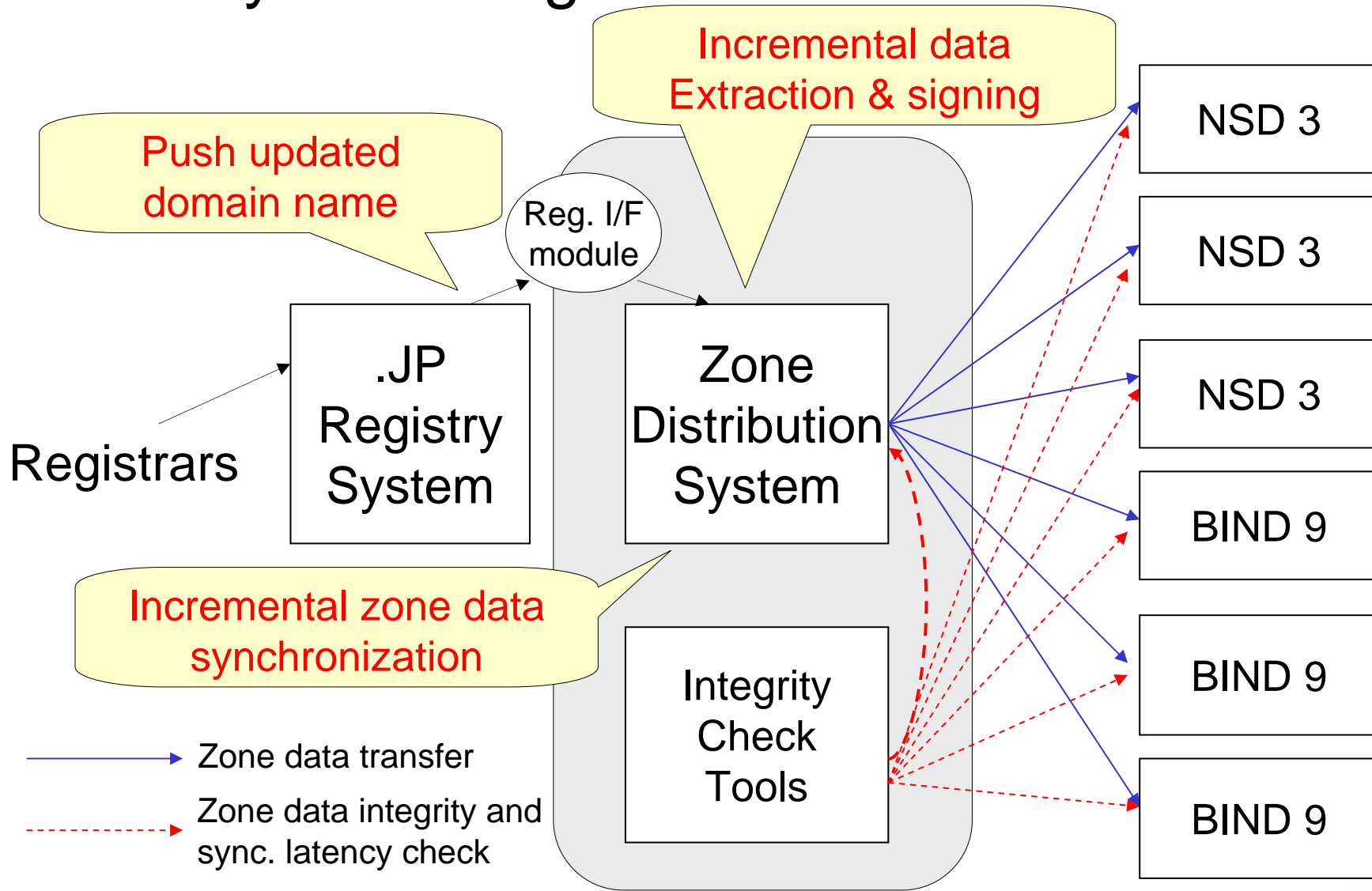 - Key management (generation/rollover)

Design Concepts of the Prototype Implementation

- **Easy integration** to current Registry system
  - Least changes to Registry system
  - DNSSEC & related features are provided by the Prototype implementation

- RRset extraction from Registry database & DNSSEC signing in an incremental manner
  - For rapid updates of large zone

- Data integrity check of DNS servers without service interruption
  - Under rapid update environment

System Components

1. Registry System (pseudo .JP/real .JP)
2. Zone Distribution System
3. Integrity Checking Tools
4. DNS servers (BIND 9, NSD 3)

System Diagram

Incremental data
Extraction & signing

Push updated
domain name

Reg. I/F
module

Registrars

.JP
Registry
System

Zone
Distribution
System

Incremental zone data
synchronization

Integrity
Check
Tools

NSD 3

NSD 3

NSD 3

BIND 9

BIND 9

BIND 9

Zone data transfer

Zone data integrity and
sync. latency check

8

Zone Distribution System

- 'Intelligent box' between Registry system and DNS servers
  - Incremental data extraction from .JP Registry system
  - Zone data distribution to DNS servers
    - Using IXFR/AXFR for BIND, NSD, etc.
  - Zone data revision management
    - Possible to obtain any #serial of zone data
    - For integrity check without service interruption
  - DNSSEC features (next slide)

- Implementation details
  - Developed from scratch by Java
  - PostgreSQL/Oracle/HSQL as backend DB

DNSSEC Features of Zone Distribution System

- Compliant to NSEC & NSEC3

- ZSK creation & RRset signing for all domains of Registry database on system initialization

- Incremental RRset signing according to the updates on Registry database

- Semi-automatic ZSK rollover & re-signing of RRsets
    - Semi-automatic means KSK private key needs to attach to the system manually for each time by security reason
    - When re-signing, old DNSKEYs / RRSIGs are deleted after appropriate time considering their TTLs
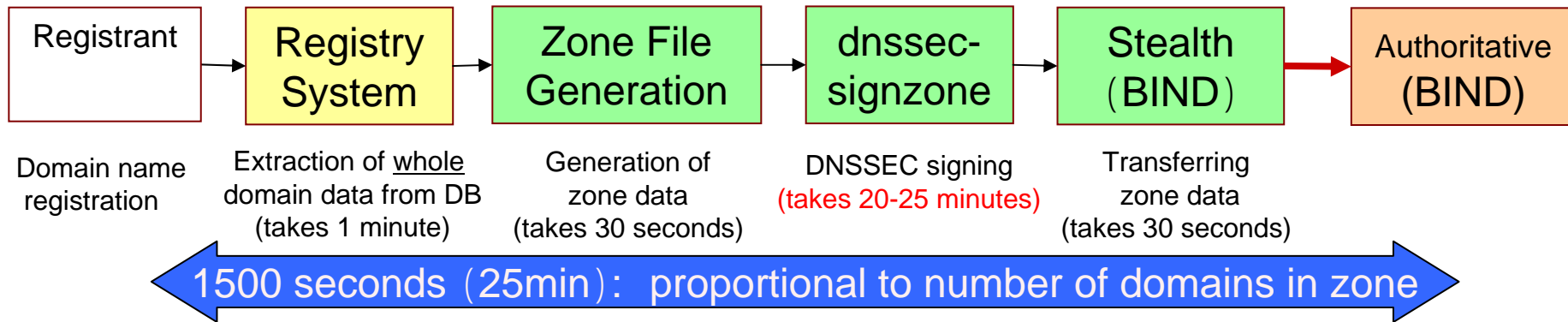
Integrity Checking Tools

- Zone synchronization latency check tool
    - Measures update latency of each DNS server from Zone Distribution System and checks if the latency is in allowable time frame
    - Designed for frequent & light-weight checking

- Full data integrity check tool
    - Verifies zone data synchronization between Zone Distribution System and each DNS server
        - Obtains complete zone data from each DNS server and compares with corresponding zone data in Zone Distribution System
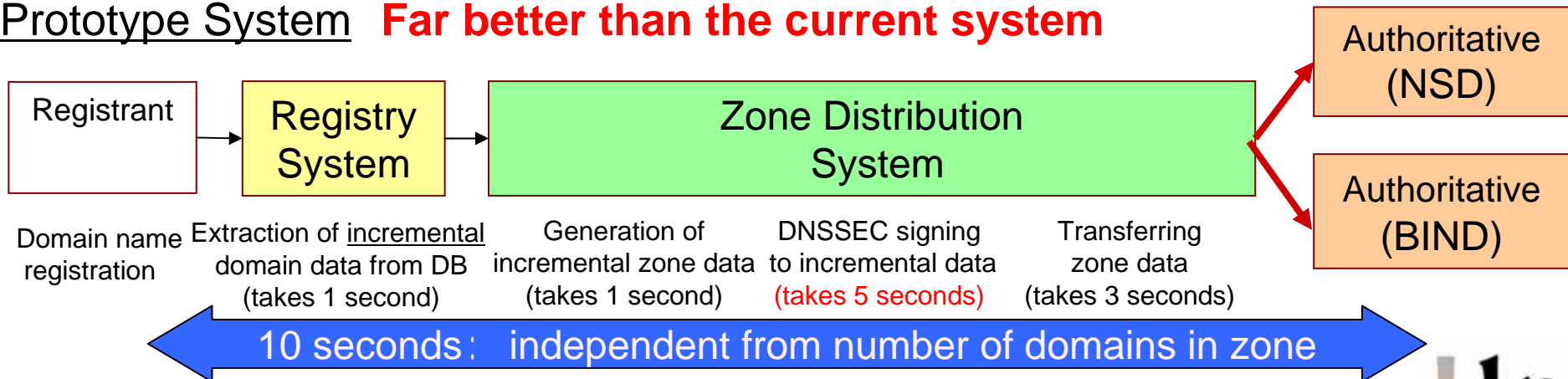   - Designed for periodic (ex. daily/weekly) & comprehensive checking

Update Performance: Comparing with Current System

In case of adding 100 domain to zone of 1 million domains, with DNSSEC signing
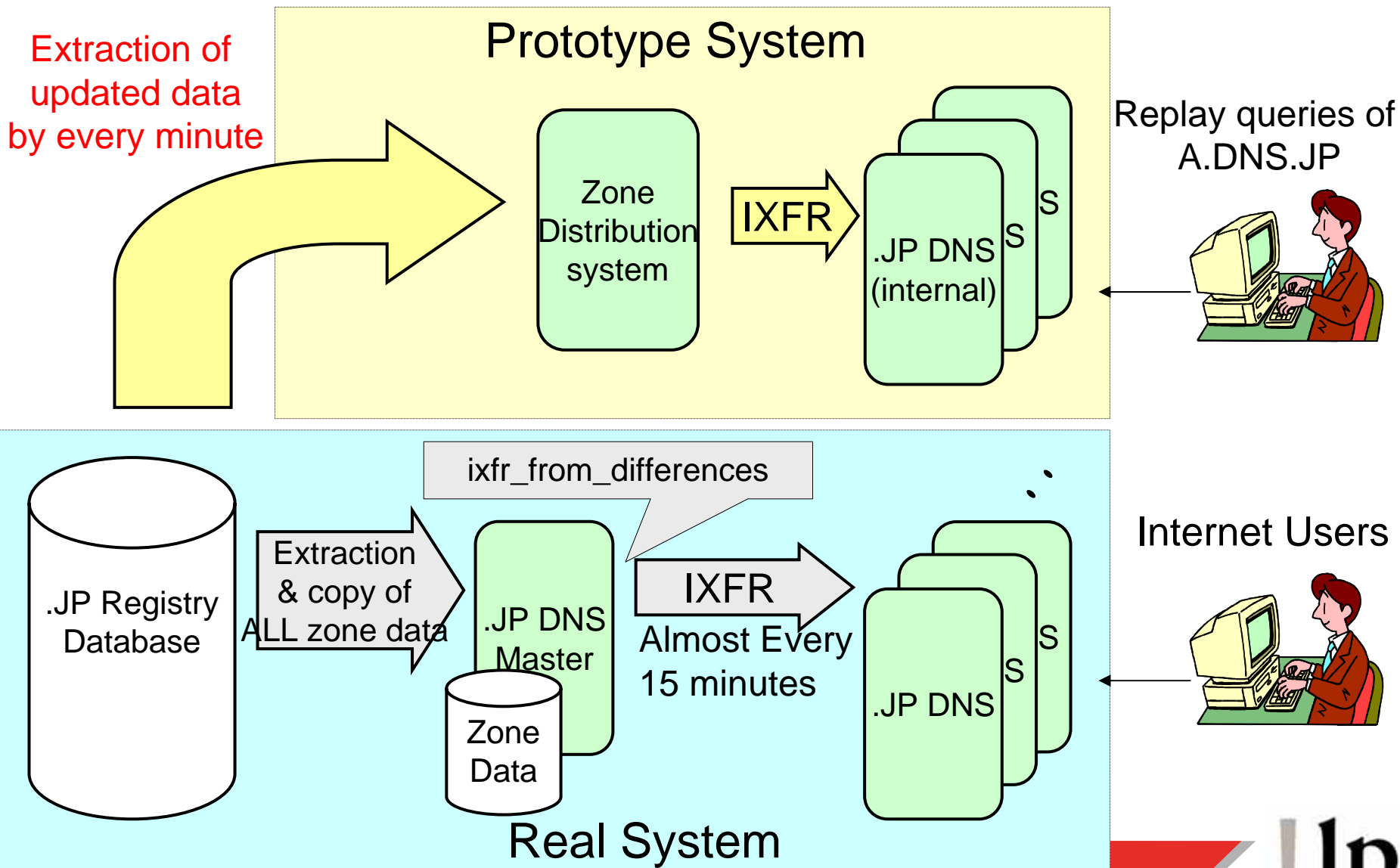
.JP registry System (current+DNSSEC)

| Registrant | Registry System | Zone File Generation | dnssec-signzone | Stealth BIND | Authoritative (BIND) |

Domain name registration

Extraction of whole domain data from DB (takes 1 minute)

Generation of zone data (takes 30 seconds)

DNSSEC signing (takes 20-25 minutes)

Transferring zone data (takes 30 seconds)

⬅️➡️ 1500 seconds 25min : proportional to number of domains in zone

Prototype System **Far better than the current system**

| Registrant | Registry System | Zone Distribution System | Authoritative (NSD) / Authoritative (BIND) |

Domain name registration

Extraction of incremental domain data from DB (takes 1 second)

Generation of incremental zone data (takes 1 second)

DNSSEC signing to incremental data (takes 5 seconds)

Transferring zone data (takes 3 seconds)

⬅️➡️ 10 seconds   independent from number of domains in zone

Field Test of Prototype System

- **Check following performances**
  - Data update latency from .JP Registry database change
  - Data integrity of DNS servers after long term running
  - Without DNSSEC signing
    - Due to implementation problem at that time..

- **Connected Zone Distribution System to real .JP Registry system for 1 month**
  - Configured to pull out updated data every minute from .JP Registry database

- **DNS servers were for internal test only**
  - Placed in Tokyo & New York
  - Simulate queries from A.DNS.JP query log so that they had the same load as real .JP DNS servers

Field Test Scheme

Prototype System

Extraction of
updated data
by every minute

Zone
Distribution
system

IXFR

.JP DNS
(internal)

S
S
S

Replay queries of
A.DNS.JP

Real System

.JP Registry
Database

Extraction
& copy of
ALL zone data

.JP DNS
Master

Zone
Data

ixfr_from_differences

IXFR
Almost Every
15 minutes

.JP DNS

S
S
S

Internet Users

System Integration needed for Field Test

- ## Modification to real .JP Registry System
  - Only added RDBMS trigger to .JP Registry database which generates updated domain names list

- ## Data import Java class of Zone Distribution System
  - To obtain incremental data from .JP Registry database according to the list

- ## Easy to integrate!
  - Relatively..

Field Test Results

- ## Average within 2 seconds of DNS data update latency from .JP Registry database change

  - ### If DNSSEC signing were done, it would be took 5 seconds or so?

- ## NO data inconsistency after 1 month running

Conclusions/Summary of the R&D

- ## JPRS Developed
  - Incremental data processing mechanism of .JP Registry updates, including DNSSEC signing/key management features
    - As Zone Distribution System

- ## The field test results satisfied
  - JP TLD requirements of large zone, rapid updates, reliable synchronization

Thank you!