

Introduction to the DNS world (DNS, URI, ENUM)

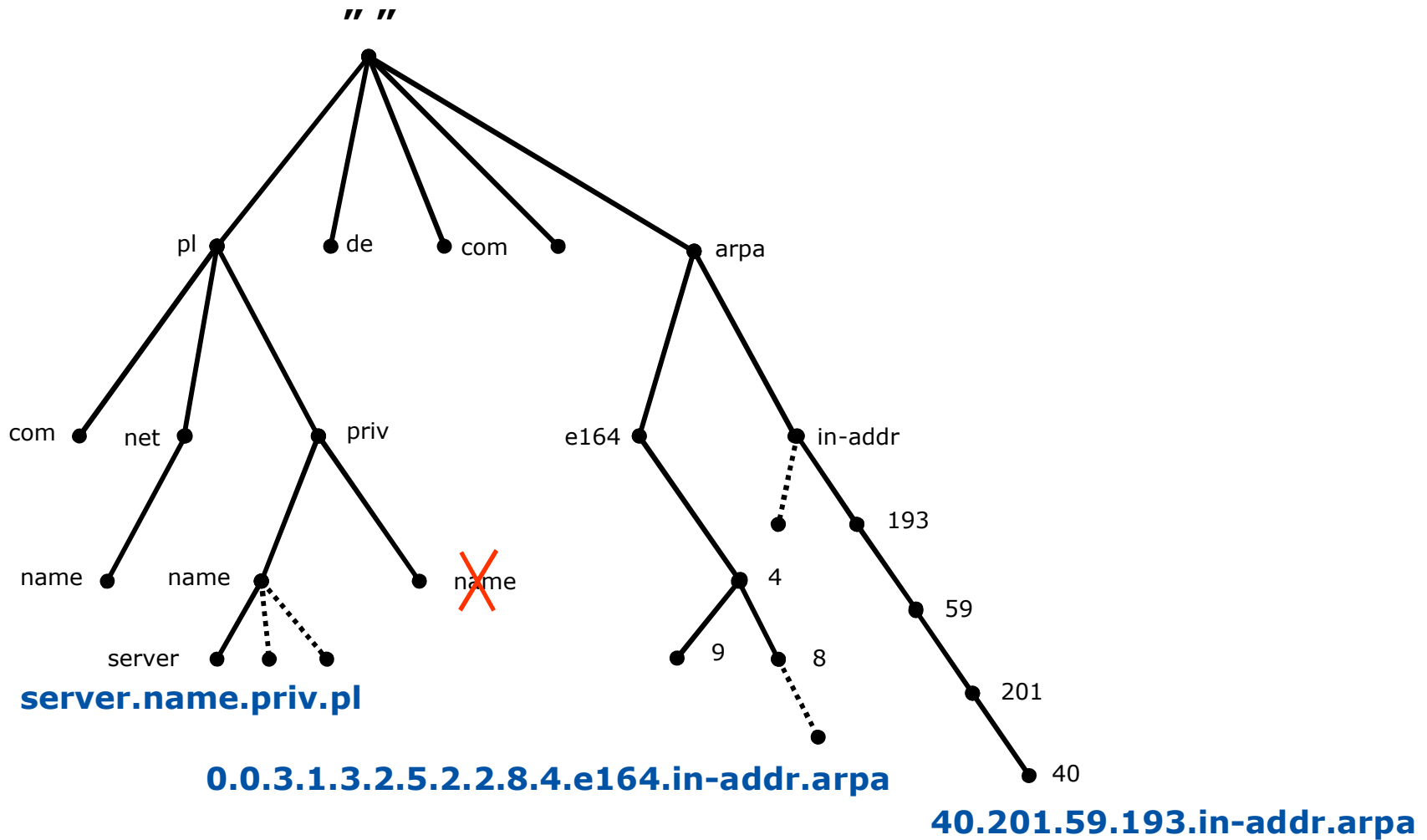
Andrzej Bartosiewicz, Rafał Galiński
Number Portability
For Fixed, Mobile, FMC and VoIP Networks
21st - 22nd March 2006, London

DNS

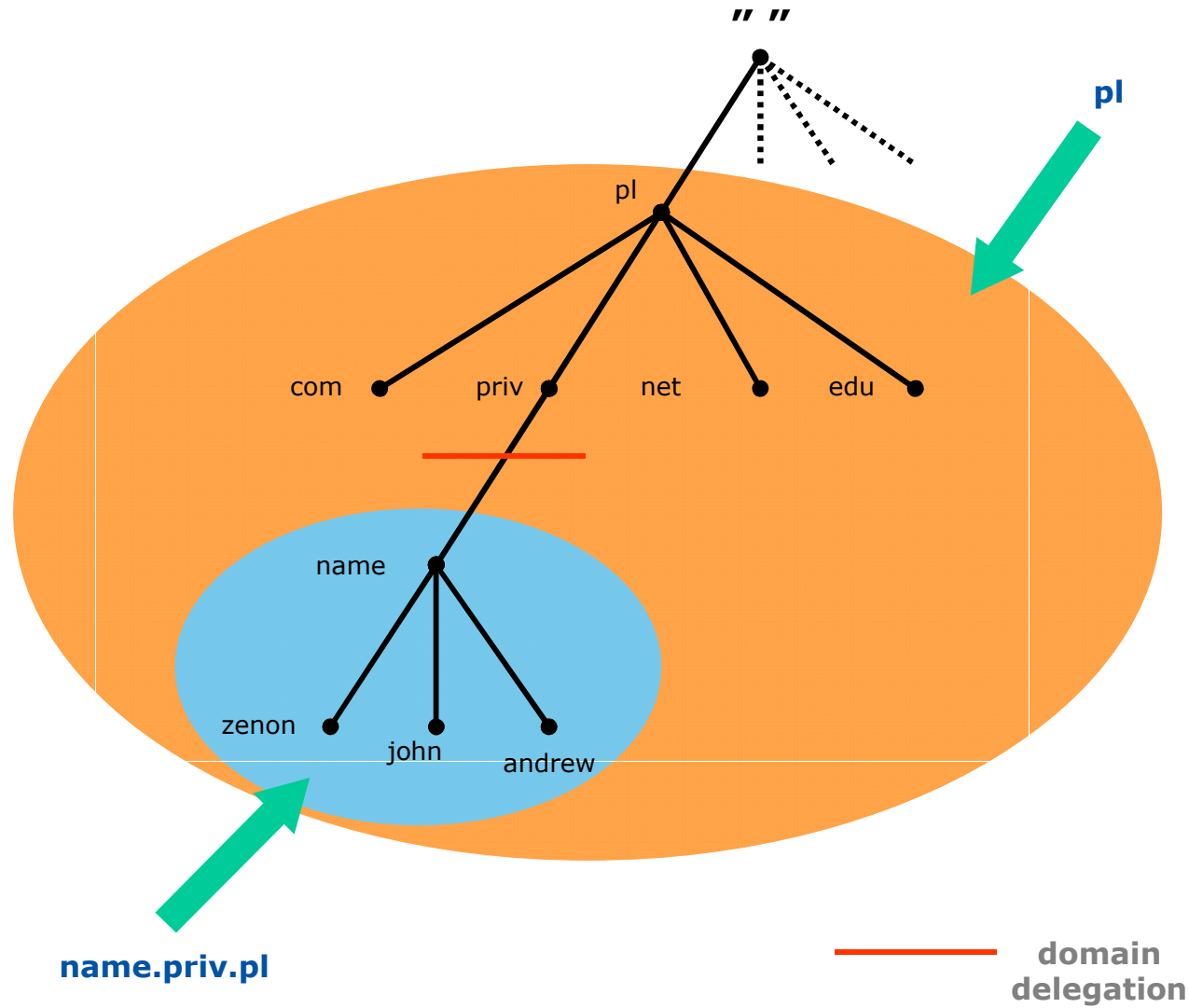
Naming History

- 1970's ARPANet. (host.txt)
- 1980's TCP/IP
- 1987 DNS related in RFCs 1034 (Domain Names - Concepts and Facilities) and 1035 (Domain Names - Implementation and Specification) by P. Mockapetris

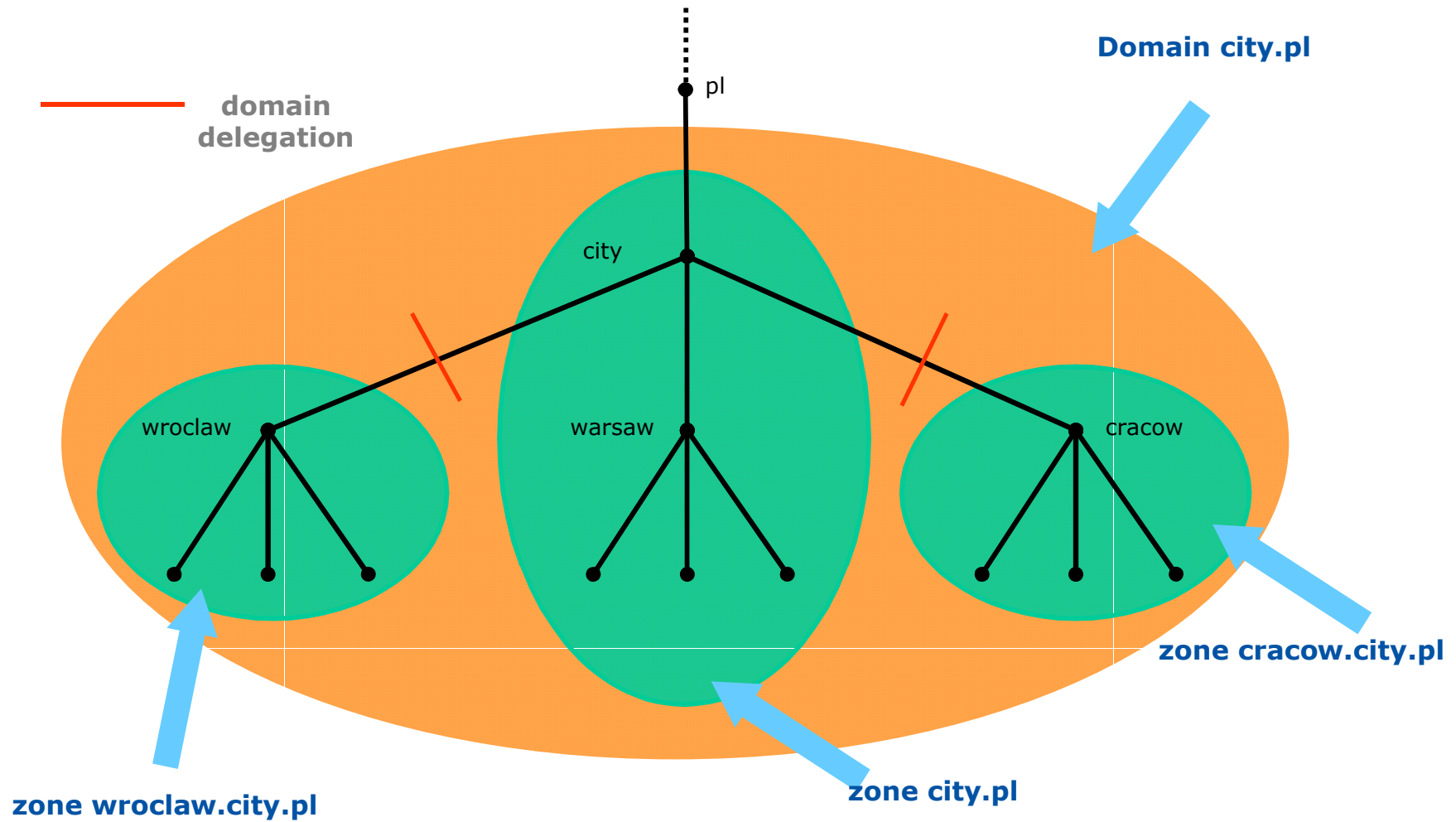
Structure of the DNS name space



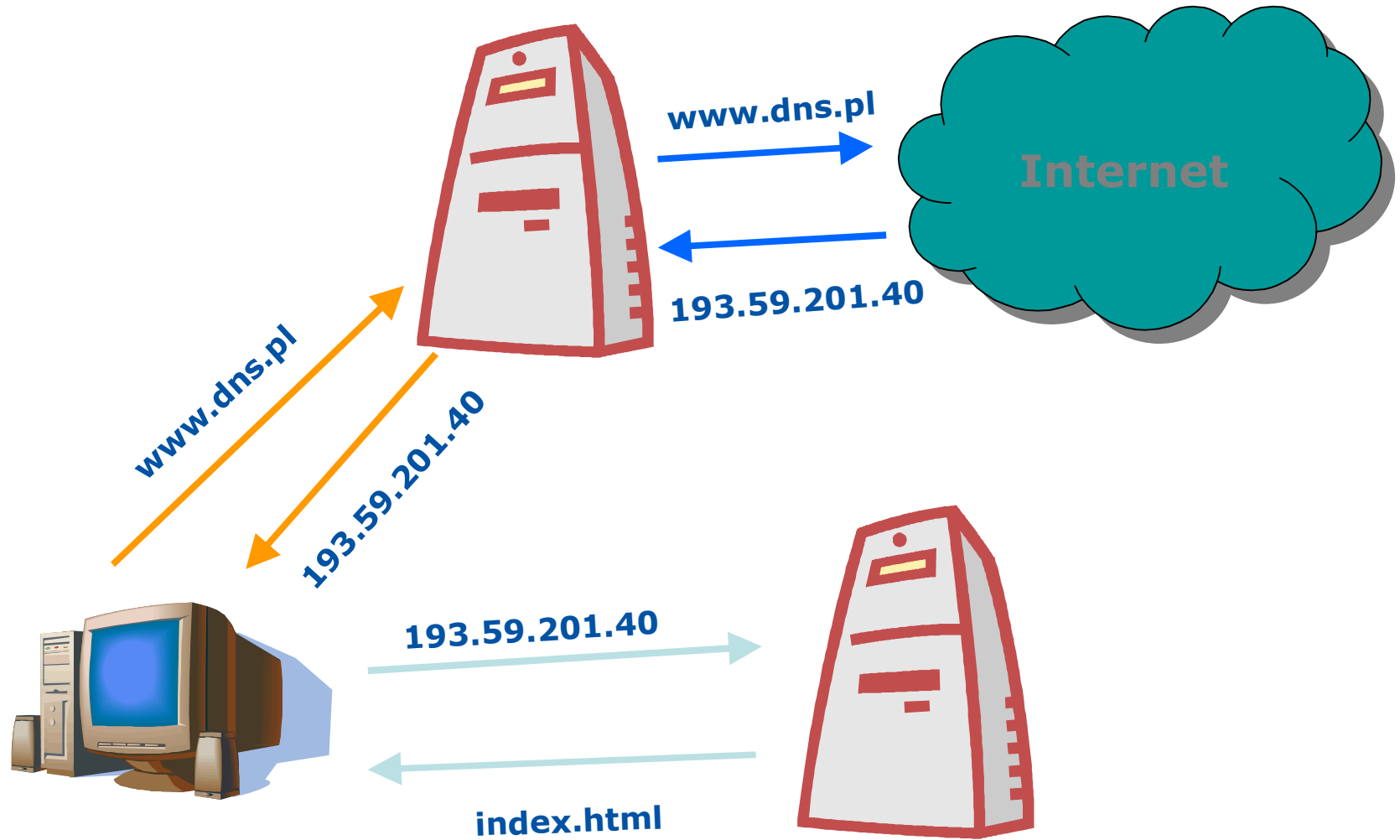
Name servers and Zones...



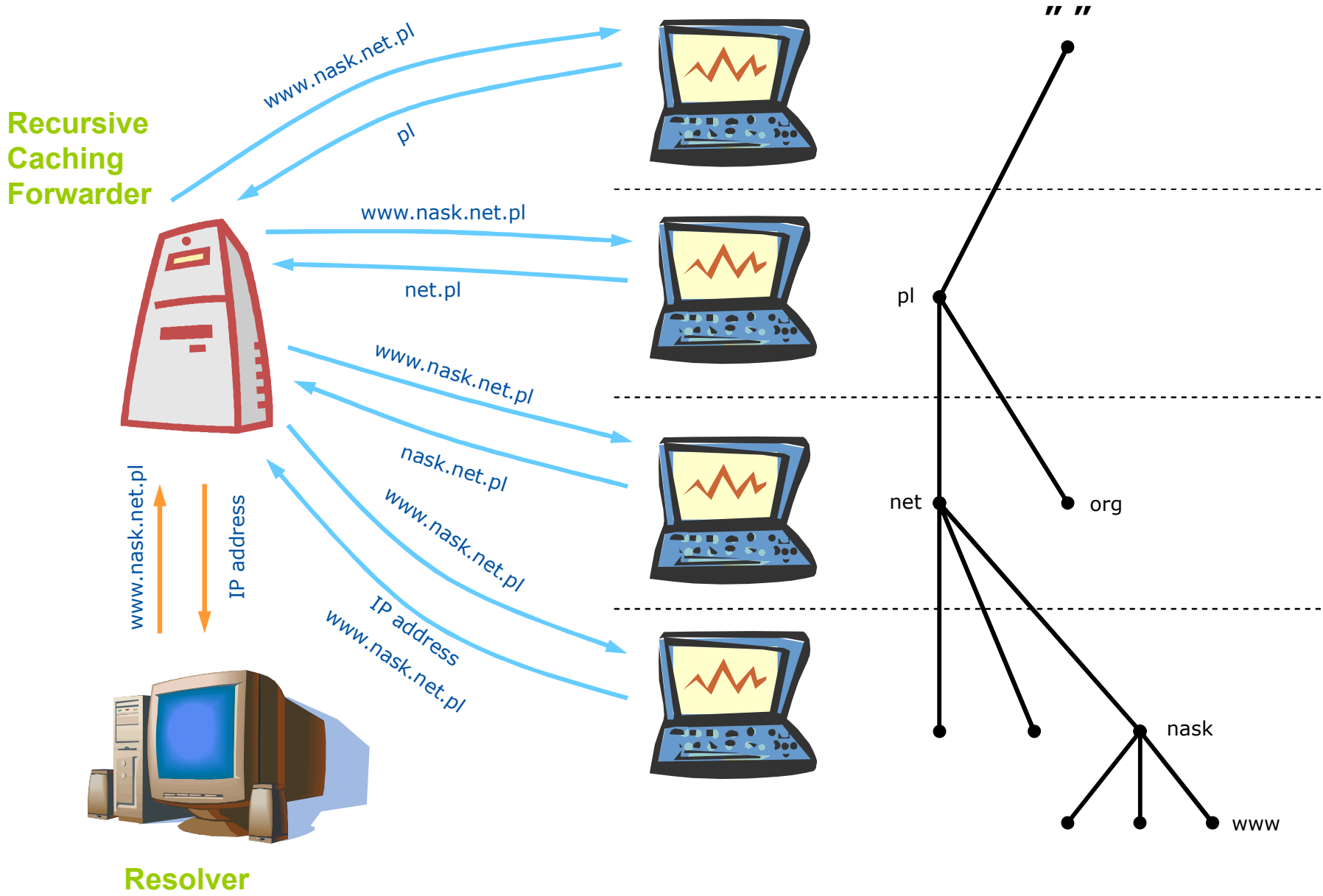
Domain, Zone, Delegation...



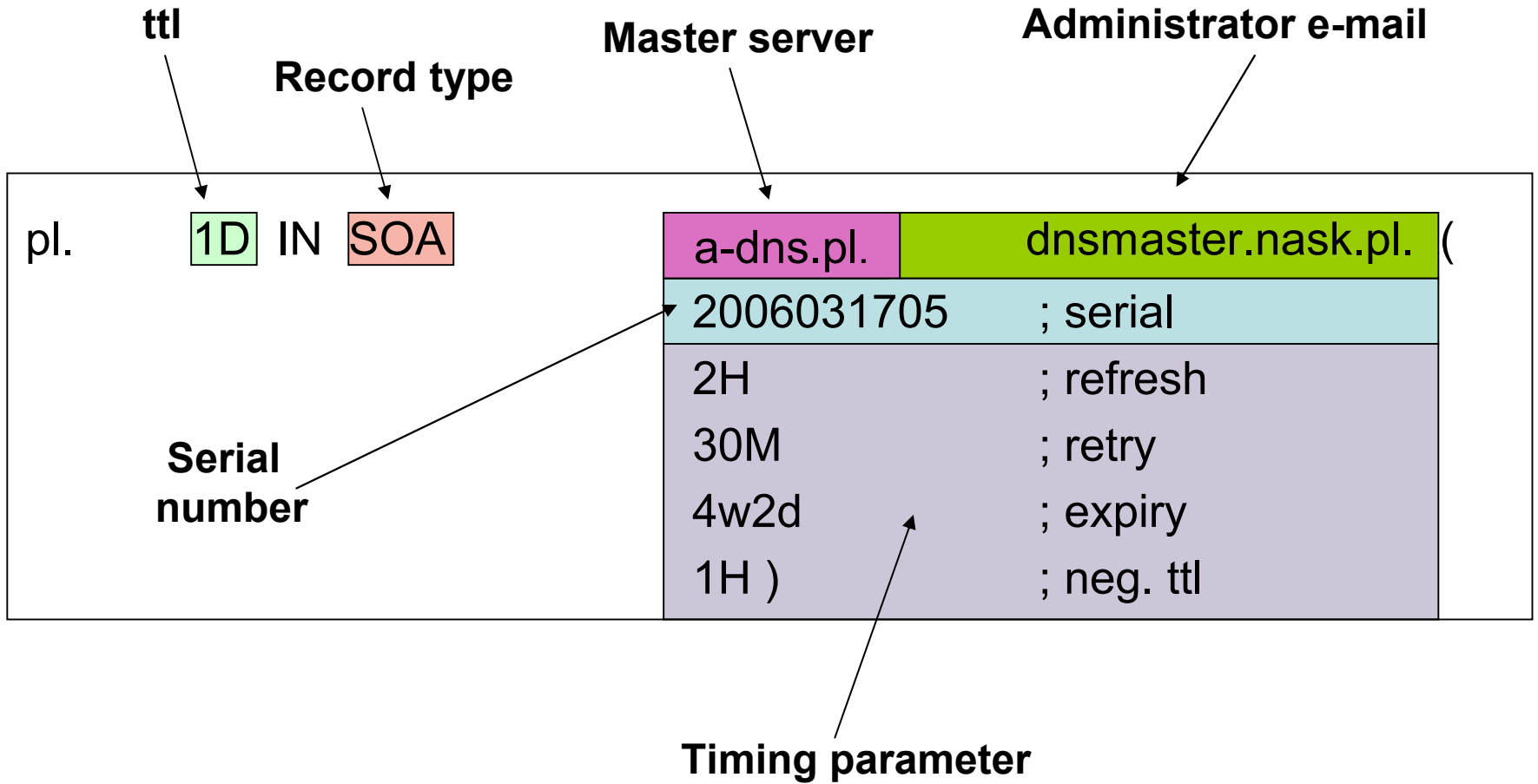
How Does DNS work ?



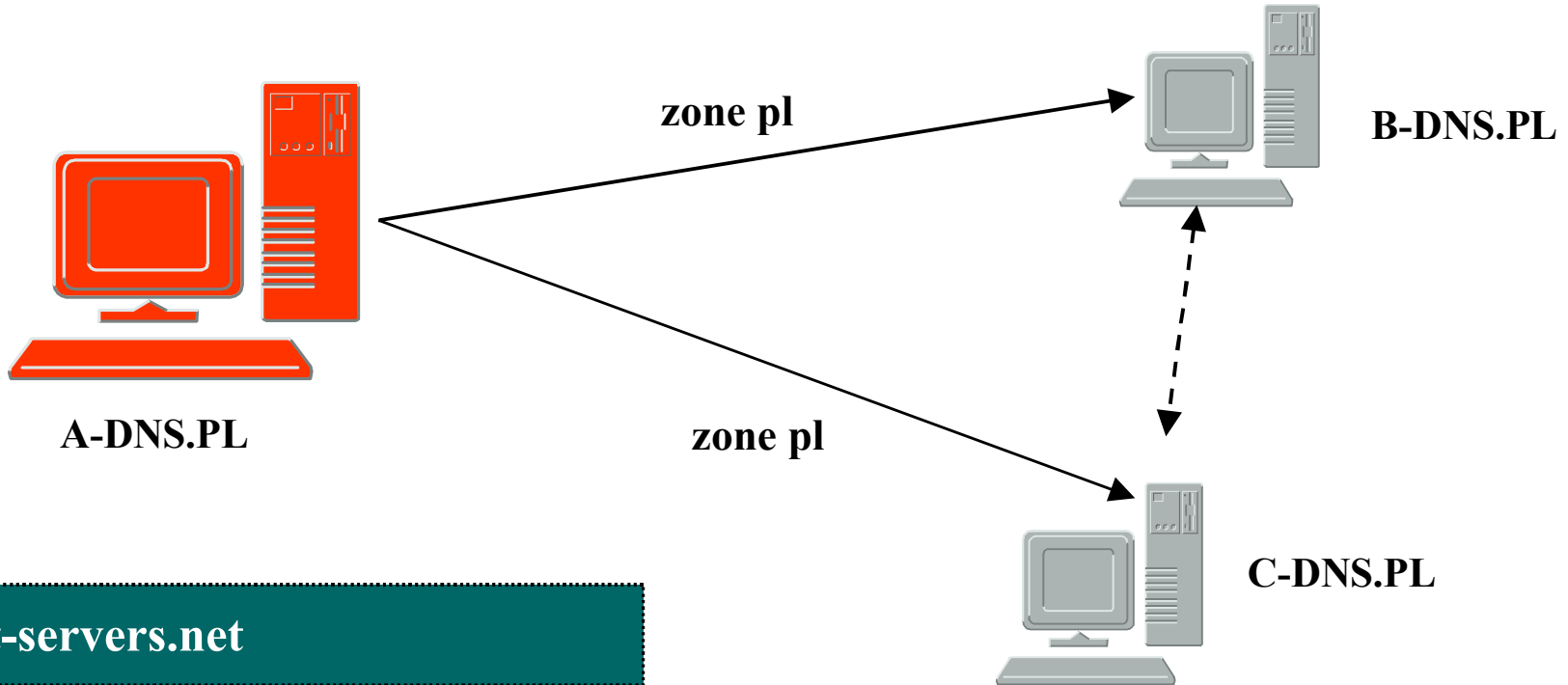
How Does DNS work ?



SOA - Resource Record



[Primary & Secondary Nameservers]



a.root-servers.net

pl. 1D IN NS A-DNS.pl.

pl. 1D IN NS B-DNS.pl.

pl. 1D IN NS C-DNS.pl.

...

Uniform Resource Identifier

The basics of URI

- A Uniform Resource Identifier (URI) is a compact sequence of characters that identifies an abstract or physical resource.
- RFC 3986: „Uniform Resource Identifier (URI): Generic Syntax”

Examples

- `ftp://ftp.is.co.za/rfc/rfc1808.txt`
- `http://www.ietf.org/rfc/rfc2396.txt`
- `ldap://[2001:db8::7]/c=GB?objectClass?one`
- `mailto:John.Doe@example.com`
- `news:comp.infosystems.www.servers.unix`
- `tel:+1-816-555-1212`
- `telnet://192.0.2.16:80/`
- `urn:oasis:names:specification:docbook:dtd:xml:4.1.2`

RFC 3986: „Uniform
Resource Identifier (URI):
Generic Syntax”

Generic URI syntax

- *The generic URI syntax consists of a hierarchical sequence of components referred to as the scheme, authority, path, query, and fragment.*
- *URI = scheme ":" hier-part ["?" query] ["#" fragment] hier-part = "//" authority path-abempty / path-absolute / path-rootless / path-empty*

RFC 3986: „Uniform
Resource Identifier (URI):
Generic Syntax”

[Scheme]

- Each URI begins with a scheme name that refers to a specification for assigning identifiers within that scheme.
- *As such, the URI syntax is a federated and extensible naming system wherein each scheme's specification may further restrict the syntax and semantics of identifiers using that scheme.*

RFC 3986: „Uniform
Resource Identifier (URI):
Generic Syntax”

[Authority]

- Many URI schemes include a hierarchical element for a naming authority so that governance of the name space defined by the remainder of the URI is delegated to that authority
- The generic syntax provides a common means for distinguishing an authority based on a registered name or server address, along with optional port and user information.

authority = [userinfo "@"] host [":" port]

- host subcomponent of authority is identified by an IP literal encapsulated within square brackets, an IPv4 address in dotted- decimal form, or a registered name.

host = IP-literal / IPv4address / reg-name

RFC 3986: „Uniform
Resource Identifier (URI):
Generic Syntax”

[Path]

- *The path component contains data, usually organized in hierarchical form, that, along with data in the non-hierarchical query component, serves to identify a resource within the scope of the URI's scheme and naming authority (if any).*
- *The path is terminated by the first question mark ("?") or number sign ("#") character, or by the end of the URI.*

RFC 3986: „Uniform
Resource Identifier (URI):
Generic Syntax”

ENUM

ENUM (1)

- ENUM is IETF standards track RFC 3761
- *Transformation of International Public Telecommunication Numbers in the international format into DNS names*
- *Use of existing DNS services like delegation through NS records and NAPTR records, one can look up what services are available for a specific E.164 in a decentralized way with distributed management of the different levels in the lookup process*

RFC 3761

The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)

ENUM (2)

- ENUM is only applicable for E.164 numbers.
- ENUM compliant applications **MUST** only query DNS for what it believes is an E.164 number.

How to translate the telephone number into ENUM domain?

- Add to the telephone number the country code number. In our case it is „+48“. The number looks like:
+48 606 24-15-70.
- Remove all characters except digits. The number looks like: 48606241570.
- Add dots between digits: 4.8.6.0.6.2.4.1.5.7.0
- Reverse the order: 0.7.5.1.4.2.6.0.6.8.4
- Add Tier-0 zone - e164.arpa (it may be changed in the future).
- And finally our ENUM domain:
0.7.5.1.4.2.6.0.6.8.4.e164.arpa

NAPTR RR Format

- Order
- Preference
- Flags
- Service
- Regexp
- Replacement

[Order]

- *A 16-bit unsigned integer specifying the order in which the NAPTR records MUST be processed in order to accurately represent the ordered list of Rules.*
- *The ordering is from lowest to highest. If two records have the same order value then they are considered to be the same rule and should be selected based on the combination of the Preference values and Services offered.*

RFC 3403: Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database

[Preference]

- *Although it is called "preference" in deference to DNS terminology, this field is equivalent to the Priority value in the DDDS Algorithm.*
- *It is a 16-bit unsigned integer that specifies the order in which NAPTR records with equal Order values SHOULD be processed, low numbers being processed before high numbers.*
- *This is similar to the preference field in an MX record, and is used so domain administrators can direct clients towards more capable hosts or lighter weight protocols. A client MAY look at records with higher preference values if it has a good reason to do so such as not supporting some protocol or service very well.*

RFC 3403: Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database

[Flags]

- *A <character-string> containing flags to control aspects of the rewriting and interpretation of the fields in the record.*
- *Flags are single characters from the set A-Z and 0-9.*
- *The case of the alphabetic characters is not significant. The field can be empty.*
- *It is up to the Application specifying how it is using this Database to define the Flags in this field. It must define which ones are terminal and which ones are not.*

RFC 3403: Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database

[Services]

- *A <character-string> that specifies the Service Parameters applicable to this this delegation path.*
- *It is up to the Application Specification to specify the values found in this field.*

[Regexp]

- *A <character-string> containing a substitution expression that is applied to the original string held by the client in order to construct the next domain name to lookup.*
- *As stated in the DDDS algorithm, The regular expressions **MUST NOT** be used in a cumulative fashion, that is, they should only be applied to the original string held by the client, never to the domain name produced by a previous NAPTR rewrite.*
- *The latter is tempting in some applications but experience has shown such use to be extremely fault sensitive, very error prone, and extremely difficult to debug.*

RFC 3403: Dynamic Delegation Discovery System (DDDS) Part Three: The Domain Name System (DNS) Database

[Replacement]

- *A <domain-name> which is the next domain-name to query for depending on the potential values found in the flags field. This field is used when the regular expression is a simple replacement operation. Any value in this field **MUST** be a fully qualified domain-name. Name compression is not to be used for this field.*