# DNS data representation for use in RPP

IETF123 Madrid, Jul 2025

Christian Simmen, DENIC eG

# DNS Representation: Problems to solve for

- Multiple extensions in EPP
  - DNSSEC
  - TTL
  - Nsset / Keyset
- Use cases outside of EPP
  - Authoritative data
  - Provisioning of DNS data in general
- Evolving DNS landscape
  - DELEG

# DNS resource records as a data model

| NAME | TTL | CLASS | TYPE | RDATA |
|---|---|---|---|---|
| de. | 21600 | IN | NS | a.nic.de. |

```
{
  "@type": "Domain",
  "name": "de",
  "dns": {
    "records": [
      { "name": "@", "type": "ns", "rdata": { "nsdname": "a.nic.de." } }
    ],
    "controls": {
      "ttl": { "ns":  21600 }
    }
  }
}
```

<u>RR definition</u>

- NAME
- TYPE
- RDATA

<u>Operational controls</u>

- TTL

# DNS resource records as a data model

3.3.11. NS RDATA format

```
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
/                     NSDNAME                   /
/                                               /
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

3.4.1. A RDATA format

```
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                    ADDRESS                    |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

3.3.9. MX RDATA format

```
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                  PREFERENCE                   |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
/                   EXCHANGE                    /
/                                               /
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
```

```
{
  "name": "@", "type": "ns",
  "rdata": {
    "nsdname": "ns1.example.net."
  }
}

{
  "name": "@", "type": "a",
  "rdata": {
    "address": "192.0.2.1"
  }
}

{
  "name": "@", "type": "mx",
  "rdata": {
    "preference": "10",
    "exchange"  : "mx1.example.com"
  }
}
```

# Example: A Simple Delegation

```
{
  "@type": "Domain",
  "name": "example.com",
  "dns": {
    "records": [

      { "name": "@", "type": "ns", "rdata": { "nsdname": "ns1.example.net." } },
      { "name": "@", "type": "ns", "rdata": { "nsdname": "ns.example.com." } },

      { "name": "ns.example.com.", "type": "a", "rdata": { "address": "192.0.2.1" } },
      { "name": "ns", "type": "aaaa", "rdata": { "address": "2001:DB8::1" } }

    ]
  }
}
```

- **"@"** Reference to the provisioned object
- **"ns.example.com."** Trailing dot indicates FQDN
- **"ns"** (**ns.example.com.**) Relative names get expanded

# Maintaining compatibility with EPP

- Integrates formerly separated EPP use cases
  - Host attributes (RFC 5731)
  - Host objects (RFC 5732)
  - DNSSEC (RFC 5910)
  - TTL (RFC 9803)

- Unified structure can be used for all of the above use cases

- Flattened structure removes complexity

# Integrating new RR Types

- JSON model allows:
    - Type-flexible **dns[]** list
    - Arbitrary **rdata** objects per RR type

- Guidelines enable early experimental support for new RR types

```
{
  "@type": "Domain",
  "name": "example.com",
  "dns": {
    "records": [
      {
        "name": "@",
        "type": "deleg",
        "rdata": {
          "svcpriority": "direct",
          "targetname": "ns.example.com",
          "svcparams": [
            "ipv4": "192.0.2.1",
            "ipv6": "2001:DB8::1"
          ]
        }
      },
    ]
  }
}
```

# Separation of requirements vs. DNS data model

- This draft does **not define final behavior**

- Requirements must:
    - Be discussed separately (draft-ietf-rpp-requirements)
    - Guide the scope and limits of the JSON model

- Discovery of supported RR types is mandatory

# Summary

- This draft provides:
  - A clean DNS data model
  - Strong alignment with the target system
  - EPP compatibility
- Next steps:
  - Add examples for a (d)nsset
  - IDNA support: A-lables only?
  - ...

# Thank you

**Christian Simmen:**
simmen@denic.de
https://www.linkedin.com/in/christian-simmen-47007b242/

**Pawel Kowalik:**
pawel.kowalik@denic.de
https://www.linkedin.com/in/pawelk/