

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 23, 2013

A. Newton
ARIN
S. Hollenbeck
Verisign Labs
September 19, 2012

JSON Responses for the Registry Data Access Protocol (RDAP)
draft-ietf-weirds-json-response-00

Abstract

This document describes responses in the JSON format to the Registry Data Access Protocol (RDAP) queries described in [draft-ietf-weirds-rdap-query](#).

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 23, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	3
2.	Terminology and Definitions	4
3.	Common Data Types	5
4.	The Entity Object Class	6
4.1.	The RIR Entity Object Class	7
4.2.	The DNR Entity Object Class	9
5.	The Nameserver Object Class	11
6.	The Domain Object Class	13
6.1.	The RIR Domain Object Class	13
6.2.	The DNR Domain Object Class	16
7.	The IP Network Object Class	20
8.	Autonomous System Number Entity Object Class	23
9.	Contributing Authors and Acknowledgements	26
10.	References	27
10.1.	Normative References	27
10.2.	Informative References	28
Appendix A.	Suggested Values	29
A.1.	Status	29
A.2.	Roles	29
A.3.	URI Types	30
Appendix B.	Suggested Data Modeling with the Entity Object Class	31
B.1.	Registrants and Contacts	31
B.2.	Registrars	32
Appendix C.	IDN Query and Response Model	34
Appendix D.	Postal Addresses vs Location	35
Appendix E.	Changelog	36
	Authors' Addresses	37

1. Introduction

This document describes responses in the JSON [[RFC4627](#)] format for the RESTful web queries as defined by UNIFIED-RDAP-QUERY [[I-D.ietf-weirds-rdap-query](#)]. The signalling for JSON, JSON naming scheme, JSON namespace and extension mechanism and normalized references to other specifications are documented in RDAP-HTTP [[I-D.ietf-weirds-using-http](#)].

The data model for the responses consists of two major categories: responses returned by Regional Internet Registries (RIRs) for registrations data related to IP addresses, reverse DNS names, and Autonomous System numbers; and responses returned by Domain Name Registries (DNRs) for registration data related to forward DNS names. Where overlap exists between RIR and DNR response object classes, the RIR object classes are a proper subset of the DNR object classes. The current division between RIR and DNR object classes is given to illustrate an expectation of what data may be expected from an RIR vs a DNR. However, implementers should be aware that RIRs are not limited to the data in the RIR object classes (as an example, some RIRs have a notion of "status" for entities as defined in the DNR entity object class and may at some point start publishing that data).

Object classes defined in this document do not represent the full range of data that any registry may wish to publish. RDAP-HTTP [[I-D.ietf-weirds-using-http](#)] defines a JSON extension mechanism that maybe used by registries to insert registry specific data values.

2. Terminology and Definitions

The following list describes terminology and definitions used throughout this document:

DNR:	"Domain Name Registry".
member:	data found with in an object as defined by JSON [RFC4627].
object:	a data structure as defined by JSON [RFC4627].
object class:	the definition of members that may be found in JSON objects described in this document.
object instance:	an instantiation or specific instance of an object class.
RDAP:	"Registry Data Access Protocol".
RIR:	"Regional Internet Registry".

3. Common Data Types

JSON [[RFC4627](#)] defines the data types of a number, character string, boolean, array, object and null. This section describes the semantics and/or syntax reference for data types used in this document derived from the JSON character string.

'handle': DNRs and RIRs have registry-unique identifiers that may be used to specifically reference an object instance. The semantics of this data type as found in this document is to be a registry-unique reference to the closest enclosing object where the value is found. The data type names 'registryId', 'roid', 'nic-handle', 'registrationNo', etc... are terms often synonymous with this data type. In this document, the term 'handle' is used. The term exposed to users by clients is a presentation issue beyond the scope of this document.

IPv4 addresses: The representation of IPv4 addresses in this document uses the dotted-decimal notation described in [[RFC1166](#)]. An example of this textual representation is '192.0.2.0'.

IPv6 addresses: The representation of IPv6 addresses in this document follow the forms outlined in [[RFC5952](#)]. An example of this textual representation is '2001:db8::1:0:0:1'.

country codes: Where the identity of a geopolitical nation or country is needed, these identities are represented with the alpha-2 or 2 character country code designation as defined in [[ISO.3166.1988](#)]. The alpha-2 representation is used because it is freely available whereas the alpha-3 and numeric-3 standards are not.

domain names: Textual representations of DNS names follow the rules set forth in [[RFC4343](#)], specifically the case insensitivity and character escaping rules. Trailing periods are optional for both input and output.

email addresses: Textual representations of email addresses follow the syntax defined in [[RFC5322](#)].

dates and times: The syntax for values denoting dates and times is defined in [[RFC3339](#)].

URIs: The syntax for values denoting a Uniform Resource Identifier (URI) is defined by [[RFC3986](#)].

4. The Entity Object Class

The entity object class appears throughout this document and is an appropriate response for the /entity/XXXX query defined in UNIFIED-RDAP-QUERY [[I-D.ietf-weirds-rdap-query](#)]. This object class represents the information of organizations, corporations, governments, non-profits, clubs, individual persons, and informal groups of people. All of these representations are so similar that it is best to represent them in JSON [[RFC4627](#)] with one construct, the entity object class, to aid in the re-use of code by implementers.

Many of the members of the entity object class are repeated in other object classes described later in this document.

4.1. The RIR Entity Object Class

The following is an example of an RIR entity:

```
{
  "handle" : "XXXX",
  "names": [ "Joe Bob, Inc.", "Bobby Joe Shopping" ],
  "roles" : [ "registrant" ],
  "postalAddress" :
  [
    "123 Maple Ave",
    "Suite 90001",
    "Vancouver",
    "BC",
    "12393"
  ],
  "emails" : [ "joe@bob.com", "bob@joe.com" ],
  "phones" :
  {
    "office" : [ "999-999-999-99", "111-111-111-11" ],
    "fax" : [ "222-222-222-22" ],
    "mobile" : [ "333-333-333-33" ]
  },
  "remarks" :
  [
    "she sells seas shells",
    "down by the seashore"
  ],
  "uris" :
  [
    {
      "type" : "source",
      "uri" : "http://whois-rws.net/contact/xxxx"
    },
    {
      "type" : "held",
      "uri" : "http://example.net/location/xxxx"
    }
  ],
  "registrationDate" : "1990-12-31T23:59:60Z",
  "lastChangedDate" : "1990-12-31T23:59:60Z",
  "lastChangedBy" : "joe@bob.com"
}
```

This object as the following members.

- o handle -- a string representing an registry unique identifier of the entity
- o names -- an array of strings, each signifying the name of the entity
- o roles -- an array of strings, each signifying the relationship an object would have with its closest containing object.
- o postalAddress -- an array of string, each representing a line in a postal address.
- o emails -- an array of strings, each containing an email address for the entity
- o phones -- an object containg telephone information associated with the entity, with the following members:
 - * office -- an array of strings, each being a telephone number
 - * fax -- an array of strings, each being a telephone number
 - * mobile -- an array of strings, each being a telephone number
- o remarks -- an array of strings, each containing comments about the entity
- o uris -- an array of objects, each object having the following members:
 - * type -- a string denoting the application type of the "uri" value
 - * uri -- a string containing a URI [[RFC3986](#)]
- o registrationDate -- a string containing the date the entity was registered
- o lastChangedDate -- a string containing the date of last change made to the entity
- o lastChangedBy -- a string containing an identifier of the party responsible for the last change made to the entity registration

[4.2.](#) The DNR Entity Object Class

The DNR entity object class is a superset of the RIR entity object class ([Section 4.1](#)). It has the following additional members:

- o registrationBy -- a string containing an identifier of the party responsible for the registration of the entity
- o sponsoredBy -- a string containing an identifier of the party through which the registration was made, such as an IANA approved registrar
- o resoldBy -- a string containing an identifier of the party originating the registration of the entity.
- o status -- an array of strings indicating the state of the entity
- o port43 -- a string containing the fully-qualified host name of the WHOIS [[RFC3912](#)] server where the object instance may be found.

The following is an example of a DNR entity:

```
{
  "handle" : "XXXX",
  "names": [ "Joe Bob, Inc.", "Bobby Joe Shopping" ],
  "status" : [ "validated", "locked" ],
  "postalAddress" :
  [
    "123 Maple Ave",
    "Suite 90001",
    "Vancouver",
    "BC",
    "12393"
  ],
  "emails" : [ "joe@bob.com", "bob@joe.com" ],
  "phones" :
  {
    "office" : [ "999-999-999-99", "111-111-111-11" ],
    "fax" : [ "222-222-222-22" ],
    "mobile" : [ "333-333-333-33" ]
  },
  "remarks" :
  [
    "she sells seas shells",
    "down by the seashore"
  ],
  "uris" :
  [
    {
      "type" : "source",
      "uri" : "http://whois-rws.net/entity/xxxx"
    },
    {
      "type" : "held",
      "uri" : "http://example.net/location/xxxx"
    }
  ],
  "port43" : "whois.example.net",
  "registrationDate" : "1990-12-31T23:59:60Z",
  "registrationBy" : "ABC123",
  "lastChangedDate" : "1990-12-31T23:59:60Z",
  "lastChangedBy" : "ABC123",
  "sponsoredBy" : "SponsorXYZ",
  "resoldBy" : "ResellerPDQ"
}
```

5. The Nameserver Object Class

The nameserver object class is used by both RIRs and DNRs. Unlike other object classes used by both registries where the RIR object class is a subset of the DNR object class, a clear delineation is not made with the nameserver object class because some DNRs have the same or a similar registration model as the RIRs. RIRs and some DNRs register or expose nameserver information as an attribute of a domain name, while other DNRs model nameservers as "first class objects".

The nameserver object class accomodates both models and degrees of variation in between.

The following is an example of a nameserver object.

```
{
  "handle" : "XXXX",
  "name" : "ns1.example.com",
  "status" : "active",
  "ipAddresses" : [ "192.0.2.0", "192.0.2.1" ],
  "remarks" :
  [
    "she sells seas shells",
    "down by the seashore"
  ],
  "uris" :
  [
    {
      "type" : "source",
      "uri" : "http://example.net/nameserver/xxxx"
    }
  ],
  "port43" : "whois.example.net",
  "registrationDate" : "1990-12-31T23:59:60Z",
  "registrationBy" : "ABC123",
  "lastChangedDate" : "1990-12-31T23:59:60Z",
  "lastChangedBy" : "ABC123",
  "sponsoredBy" : "SponsorXYZ",
  "resoldBy" : "ResellerPDQ"
}
```

Figure 1

Figure 1 is an example of a nameserver object with all values given. Registries using a first-class nameserver data model would embed this in domain objects as well as allowing references to it with the

/nameserver query type (all depending on the registry operators policy). Other registries may pare back the information as needed. Figure 2 is an example of a nameserver object as would be found in RIRs and some DNRs, while Figure 3 is an example of a nameserver object as would be found in other DNRs.

The following is an example of the simplest nameserver object.

```
{
  "name" : "ns1.example.com"
}
```

Figure 2

The following is an example of a simple nameserver object that might be commonly used by DNRs.

```
{
  "name" : "ns1.example.com",
  "ipAddresses" : [ "192.0.2.0", "192.0.2.1" ]
}
```

Figure 3

The nameserver object class has the following members:

- o handle -- a string representing an registry unique identifier of the nameserver
- o name -- a string containing the DNS name of the nameserver
- o ipAddresses -- an array of strings containing IPv4 and/or IPv6 addresses of the nameserver

The members "status", "remarks", "uris", "port43", "sponsoredBy", "resoldBy", "registrationBy", "registrationDate", "lastChangedDate", and "lastChangedBy" take the same form of the members of the same name of the entity object ([Section 4](#)).

[6.](#) The Domain Object Class

The domain object class represents a DNS name and point of delegation. For RIRs these delegation points are in the reverse DNS tree, whereas for DNRs these delegation points are in the forward DNS tree. The RIR domain object class is a subset of the DNR object class.

In both cases, the high level structure of the domain object class consists of information about the domain registration, nameserver information related to the domain name, and entities related to the domain name (e.g. registrant information, contacts, etc...).

The following is an elided example of the domain object showing the high level structure.

```
{
  "handle" : "XXX",
  "name" : "blah.example.com",
  ...
  "nameServers" :
  [
    ...
  ],
  ...
  "entities" :
  [
    ...
  ]
}
```

[6.1.](#) The RIR Domain Object Class

The following is an example of a JSON object representing a reverse DNS delegation point or the RIR domain object class.

```
{
  "handle" : "XXXX",
  "name" : "192.in-addr.arpa",
  "nameServers" :
  [
    { "name" : "ns1.rir.net" },
    { "name" : "ns2.rir.net" }
  ],
  "delegationKeys" :
```

```
[
  {
    "algorithm": 7,
    "digest" : "E68C017BD813B9AE2F4DD28E61AD014F859ED44C",
    "digestType" : 1,
    "keyTag" : 53814
  }
],
"remarks" :
[
  "she sells seas shells",
  "down by the seashore"
],
"uris" :
[
  {
    "type" : "source",
    "uri" : "http://whois-rws.net/network/xxxx"
  },
  {
    "type" : "parent",
    "uri" : "http://whois-rws.net/network/yyyy"
  },
  {
    "type" : "held",
    "uri" : "http://example.net/location/xxxx"
  }
],
"registrationDate" : "1990-12-31T23:59:60Z",
"lastChangedDate" : "1990-12-31T23:59:60Z",
"lastChangedBy" : "joe@bob.com",
"entities" :
[
  {
    "handle" : "XXXX",
    "names": [ "Joe Bob, Inc.", "Bobby Joe Shopping" ],
    "roles" : [ "registrant" ],
    "postalAddress" :
    [
      "123 Maple Ave",
      "Suite 90001",
      "Vancouver",
      "BC",
      "12393"
    ],
    "emails" : [ "joe@bob.com", "bob@joe.com" ],
    "phones" :
    {
```

```
    "office" : [ "999-999-999-99", "111-111-111-11" ],
    "fax" : [ "222-222-222-22" ],
    "mobile" : [ "333-333-333-33" ]
  },
  "remarks" :
  [
    "she sells seas shells",
    "down by the seashore"
  ],
  "uris" :
  [
    {
      "type" : "source",
      "uri" : "http://whois-rws.net/entity/xxxx"
    },
    {
      "type" : "held",
      "uri" : "http://example.net/location/xxxx"
    }
  ],
  "registrationDate" : "1990-12-31T23:59:60Z",
  "lastChangedDate" : "1990-12-31T23:59:60Z",
  "lastChangedBy" : "joe@bob.com"
}
]
```

The following is a description of the members of this object:

- o handle -- a string representing a registry unique identifier of the domain object instance
- o name -- a string denoting the DNS zone name, which is a domain name
- o nameservers -- an array of nameserver objects as defined by [Section 5](#)
- o delegationKeys -- an array of objects, each with the following members:
 - * algorithm -- an integer as specified by the algorithm field of a DNS DS record as specified by [RFC 4034](#) [[RFC4034](#)] in presentation format
 - * digest -- a string as specified by the digest field of a DNS DS record as specified by [RFC 4034](#) in presentation format

- * digestType -- an integer as specified by the digest type field of a DNS DS record as specified by [RFC 4034](#) in presentation format
- * keyTag -- an integer as specified by the key tag field of a DNS DS record as specified by [RFC 4034](#) in presentation format
- o entities -- an array of entity objects as defined by [Section 4.1](#).

The members "remarks", "uris", "registrationDate", "lastChangedDate", and "lastChangedBy" take the same form of the members of the same name of the entity object ([Section 4](#)).

[6.2](#). The DNR Domain Object Class

The DNR domain object class is a superset of the RIR domain object class ([Section 6.1](#)) and has the following additional members.

- o variants -- an array of strings, each containing a domain name that is a variant of this domain name object instance.
- o expirationDate -- a string containing the date and time this domain name registration will expire
- o registrationBy -- a string containing an identifier of the party responsible for the registration of the domain name
- o sponsoredBy -- a string containing an identifier of the party through which the registration was made, such as an IANA approved registrar
- o resoldBy -- a string containing an identifier of the party originating the registration of the domain name
- o status -- an array of strings indicating the state of the domain name
- o transferDate -- a string containing the date and time this domain name was transferred
- o port43 -- a string containing the fully-qualified host name of the WHOIS [[RFC3912](#)] server where the object instance may be found.

The following is an example of a JSON object representing a forward DNS delegation point or the DNR domain object class.

```
{
```



```
"handle" : "XXXX",
"name" : "blah.example.com",
"variants" : [ "blah2.example.com", "blah3.example.com" ],
"status" : [ "locked", "transferProhibited" ],
"nameServers" :
[
  {
    "handle" : "XXXX",
    "name" : "ns1.example.com",
    "status" : "active",
    "ipAddresses" : [ "192.0.2.0", "192.0.2.1" ],
    "remarks" :
    [
      "she sells seas shells",
      "down by the seashore"
    ],
    "uris" :
    [
      {
        "type" : "source",
        "uri" : "http://example.net/nameserver/xxxx"
      }
    ],
    "registrationDate" : "1990-12-31T23:59:60Z",
    "registrationBy" : "ABC123",
    "lastChangedDate" : "1990-12-31T23:59:60Z",
    "lastChangedBy" : "ABC123",
    "sponsoredBy" : "SponsorXYZ",
    "resoldBy" : "ResellerPDQ"
  },
  {
    "handle" : "XXXX",
    "name" : "ns2.example.com",
    "status" : "active",
    "ipAddresses" : [ "192.0.2.2", "192.0.2.3" ],
    "remarks" :
    [
      "she sells seas shells",
      "down by the seashore"
    ],
    "uris" :
    [
      {
        "type" : "source",
        "uri" : "http://example.net/nameserver/xxxx"
      }
    ],
    "registrationDate" : "1990-12-31T23:59:60Z",
```

```
    "registrationBy" : "ABC123",
    "lastChangedDate" : "1990-12-31T23:59:60Z",
    "lastChangedBy" : "ABC123",
    "sponsoredBy" : "SponsorXYZ",
    "resoldBy" : "ResellerPDQ"
  }
]
"delegationKeys" :
[
  {
    "algorithm": 7,
    "digest" : "E68C017BD813B9AE2F4DD28E61AD014F859ED44C",
    "digestType" : 1,
    "keyTag" : 53814
  }
],
"remarks" :
[
  "she sells seas shells",
  "down by the seashore"
],
"uris" :
[
  {
    "type" : "source",
    "uri" : "http://example.net/domain/xxxx"
  },
  {
    "type" : "held",
    "uri" : "http://example.net/location/xxxx"
  }
],
"port43" : "whois.example.net",
"registrationDate" : "1990-12-31T23:59:60Z",
"registrationBy" : "ABC123",
"lastChangedDate" : "1990-12-31T23:59:60Z",
"lastChangedBy" : "ABC123",
"sponsoredBy" : "SponsorXYZ",
"resoldBy" : "ResellerPDQ",
"expirationDate" : "2016-12-31T23:59:60Z",
"transferDate" : "1990-12-31T23:59:60Z",
"entities" :
[
  {
    "handle" : "XXXX",
    "names": [ "Joe Bob, Inc.", "Bobby Joe Shopping" ],
    "status" : [ "validated", "locked" ],
    "postalAddress" :
```

```
[
  "123 Maple Ave",
  "Suite 90001",
  "Vancouver",
  "BC",
  "12393"
],
"emails" : [ "joe@bob.com", "bob@joe.com" ],
"phones" :
{
  "office" : [ "999-999-999-99", "111-111-111-11" ],
  "fax" : [ "222-222-222-22" ],
  "mobile" : [ "333-333-333-33" ]
},
"remarks" :
[
  "she sells seas shells",
  "down by the seashore"
],
"uris" :
[
  {
    "type" : "source",
    "uri" : "http://whois-rws.net/entity/xxxx"
  },
  {
    "type" : "held",
    "uri" : "http://example.net/location/xxxx"
  }
],
"registrationDate" : "1990-12-31T23:59:60Z",
"registrationBy" : "ABC123",
"lastChangedDate" : "1990-12-31T23:59:60Z",
"lastChangedBy" : "ABC123",
"sponsoredBy" : "SponsorXYZ",
"resoldBy" : "ResellerPDQ"
}
]
```

7. The IP Network Object Class

The IP Network object class models IP network registrations found in RIRs and is the expected response for the /ip query as defined by [\[I-D.ietf-weirds-rdap-query\]](#). There is no equivalent object class for DNRs. The high level structure of the IP network object class consists of information about the network registration and entities related to the IP network (e.g. registrant information, contacts, etc...).

The following is an elided example of the IP network object type showing the high level structure.

```
{
  "handle" : "XXX",
  ...
  "entities" :
  [
    ...
  ]
}
```

The following is an example of the JSON object for the network registration information

```
{
  "handle" : "XXXX-RIR",
  "startAddress" : "10.0.0.0",
  "endAddress" : "10.0.0.255",
  "ipVersion" : 4,
  "name": "NET-RTR-1",
  "description" : [ "A network used for routing" ],
  "type" : "DIRECT ALLOCATION",
  "country" : "AU",
  "parentHandle" : "YYYY-RIR",
  "remarks" :
  [
    "she sells seas shells",
    "down by the seashore"
  ],
  "uris" :
  [
    {
      "type" : "source",
      "uri" : "http://whois-rws.net/network/xxxx"
```

```
    },
    {
      "type" : "parent",
      "uri" : "http://whois-rws.net/network/yyyy"
    },
  ],
  "registrationDate" : "20110509",
  "lastChangedDate" : "20110509",
  "lastChangedBy" : "joe@bob.com",
  "entities" :
  [
    {
      "handle" : "XXXX",
      "names": [ "Joe Bob, Inc.", "Bobby Joe Shopping" ],
      "roles" : [ "registrant" ],
      "postalAddress" :
      [
        "123 Maple Ave",
        "Suite 90001",
        "Vancouver",
        "BC",
        "12393"
      ],
      "emails" : [ "joe@bob.com", "bob@joe.com" ],
      "phones" :
      {
        "office" : [ "999-999-999-99", "111-111-111-11" ],
        "fax" : [ "222-222-222-22" ],
        "mobile" : [ "333-333-333-33" ]
      },
      "remarks" :
      [
        "she sells seas shells",
        "down by the seashore"
      ],
      "uris" :
      [
        {
          "type" : "source",
          "uri" : "http://whois-rws.net/contact/xxxx"
        },
        {
          "type" : "held",
          "uri" : "http://example.net/location/xxxx"
        }
      ],
      "registrationDate" : "1990-12-31T23:59:60Z",
      "lastChangedDate" : "1990-12-31T23:59:60Z",
    }
  ]
}
```

```
    "lastChangedBy" : "joe@bob.com"
  }
]
```

The following is a description of the members of this object:

- o handle -- a string representing an RIR unique identifier of the network registration
- o startAddress -- the starting IP address of the network, either IPv4 or IPv6
- o endAddress -- the ending IP address of the network, either IPv4 or IPv6
- o ipVersion -- an integer signifying the IP protocol version of the network: 4 signifying an IPv4 network, 6 signifying an IPv6 network
- o name -- an identifier assigned to the network registration by the registration holder
- o description -- an array of strings containing descriptive text about the network registration
- o type -- a string containing an RIR specific classification of the network
- o country -- a string containing the name of the 2 character country code of the network
- o parentHandle -- a string containing an RIR unique identifier of the parent network of this network registration
- o entities -- an array of entity objects as defined by [Section 4.1](#).

The members "remarks", "uris", "registrationDate", "lastChangedDate", and "lastChangedBy" take the same form of the members of the same name of the entity object ([Section 4.1](#)).

8. Autonomous System Number Entity Object Class

The Autonomous System Number (autnum) object class models Autonomous System Number registrations found in RIRs and represents the expected response to an /autnum query as defined by [\[I-D.ietf-weirds-rdap-query\]](#). There is no equivalent object class for DNRs. The high level structure of the autnum object class consists of information about the network registration and entities related to the autnum registration (e.g. registrant information, contacts, etc...), and is similar to the IP Network entity object class.

The following is an example of a JSON object representing an autnum.

```
{
  "handle" : "XXXX-RIR",
  "startAutnum" : "10",
  "endAutnum" : "15",
  "name": "AS-RTR-1",
  "description" : [ "AS for Exchange" ],
  "type" : "DIRECT ALLOCATION",
  "country": "AU",
  "remarks" :
  [
    "she sells seas shells",
    "down by the seashore"
  ],
  "uris" :
  [
    {
      "type" : "source",
      "uri" : "http://whois-rws.net/autnum/xxxx"
    },
    {
      "type" : "parent",
      "uri" : "http://whois-rws.net/autnum/yyyy"
    },
    {
      "type" : "held",
      "uri" : "http://example.net/location/xxxx"
    }
  ],
  "registrationDate" : "20110509",
  "lastChangedDate" : "20110509",
  "lastChangedBy" : "joe@bob.com",
  "entities" :
  [
```

```
{
  "handle" : "XXXX",
  "names": [ "Joe Bob, Inc.", "Bobby Joe Shopping" ],
  "roles" : [ "registrant" ],
  "postalAddress" :
  [
    "123 Maple Ave",
    "Suite 90001",
    "Vancouver",
    "BC",
    "12393"
  ],
  "emails" : [ "joe@bob.com", "bob@joe.com" ],
  "phones" :
  {
    "office" : [ "999-999-999-99", "111-111-111-11" ],
    "fax" : [ "222-222-222-22" ],
    "mobile" : [ "333-333-333-33" ]
  },
  "remarks" :
  [
    "she sells seas shells",
    "down by the seashore"
  ],
  "uris" :
  [
    {
      "type" : "source",
      "uri" : "http://whois-rws.net/autnum/xxxx"
    }
  ],
  "registrationDate" : "1990-12-31T23:59:60Z",
  "lastChangedDate" : "1990-12-31T23:59:60Z",
  "lastChangedBy" : "joe@bob.com"
}
]
```

The following is a description of the members of this object:

- o handle -- a string representing an RIR unique identifier of the autnum registration
- o startAutnum -- the starting number [[RFC5396](#)] in the block of autonomous system numbers

- o endAutnum -- the ending number [[RFC5396](#)] in the block of autonomous system numbers
- o name -- an identifier assigned to the autnum registration by the registration holder
- o description -- an array of strings containing descriptive text about the autnum registration
- o type -- a string containing an RIR specific classification of the autnum
- o country -- a string containing the name of the 2 character country code of the autnum

The members "remarks", "uris", "registrationDate", "lastChangedDate", and "lastChangedBy" take the same form of the members of the same name of the entity object ([Section 4.1](#)).

9. Contributing Authors and Acknowledgements

This document is derived from original work on RIR response in JSON by Byron J. Ellacott of APNIC, Arturo L. Servin of LACNIC, Kaveh Ranjbar of the RIPE NCC, and Andrew L. Newton of ARIN. Additionally, this document incorporates work on DNR responses in JSON by Ning Kong, Linlin Zhou, Jiagui Xie, and Sean Shen of CNNIC.

The components of the DNR object classes are derived from a categorization of WHOIS response formats created by Ning Kong, Linlin Zhou, and Guangqing Deng of CNNIC, Steve Sheng and Francisco Arias of ICANN, Ray Bellis of Nominet, and Frederico Neves of NIC.BR.

[10.](#) References

[10.1.](#) Normative References

- [RFC0791] Postel, J., "Internet Protocol", STD 5, [RFC 791](#), September 1981.
- [RFC1166] Kirkpatrick, S., Stahl, M., and M. Recker, "Internet numbers", [RFC 1166](#), July 1990.
- [RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", [RFC 2616](#), June 1999.
- [RFC3339] Klyne, G., Ed. and C. Newman, "Date and Time on the Internet: Timestamps", [RFC 3339](#), July 2002.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, [RFC 3986](#), January 2005.
- [RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", [RFC 4034](#), March 2005.
- [RFC4343] Eastlake, D., "Domain Name System (DNS) Case Insensitivity Clarification", [RFC 4343](#), January 2006.
- [RFC4627] Crockford, D., "The application/json Media Type for JavaScript Object Notation (JSON)", [RFC 4627](#), July 2006.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", [RFC 5322](#), October 2008.
- [RFC5396] Huston, G. and G. Michaelson, "Textual Representation of Autonomous System (AS) Numbers", [RFC 5396](#), December 2008.
- [RFC5952] Kawamura, S. and M. Kawashima, "A Recommendation for IPv6 Address Text Representation", [RFC 5952](#), August 2010.
- [ISO.3166.1988]
International Organization for Standardization, "Codes for the representation of names of countries, 3rd edition", ISO Standard 3166, August 1988.
- [I-D.ietf-weirds-rdap-query]
Newton, A. and S. Hollenbeck, "RDAP Query Format", [draft-ietf-weirds-rdap-query-00](#) (work in progress),

September 2011.

[I-D.ietf-weirds-using-http]

Newton, A., Ellacott, B., and N. Kong, "Using HTTP for RESTful Whois Services by Internet Registries", [draft-ietf-weirds-using-http-01](#) (work in progress), May 2012.

[10.2.](#) Informative References

[RFC3912] Daigle, L., "WHOIS Protocol Specification", [RFC 3912](#), September 2004.

[Appendix A.](#) Suggested Values

Due to the wide variation between the hundreds of registry operators and the on-going policy refinement by registry communities, values of some data cannot be formally standardized. This section lists suggested values for such data but is not nor will ever be a complete list of values and their meanings.

[A.1.](#) Status

Many of the object classes have a member named 'status'. This member is an array of strings, with each string denoting a status associated with the containing object. The following is a list of suggested values to use in the 'status' array:

- o 'validated' -- Signifies that the data of the object instance has been found to be accurate. This type of status is usually found on entity object instances to note the validity of identifying contact information.
- o 'update prohibited' -- Updates to the object instance are forbidden.
- o 'transfer prohibited' -- Transfers of the registration from one registrar to another are forbidden. This type of status normally applies to DNR domain names.
- o 'delete prohibited' -- Deletion of the registration of the object instance is forbidden. This type of status normally applies to DNR domain names.

[A.2.](#) Roles

Entity object classes have a member named 'roles'. This member is an array of strings, with each string indicating the role or relationship the entity object instance has with a containing object, such as a domain name or IP network. An entity object instance can have more than one type of relationship with a containing object. The following is a list of suggested values to use in the 'roles' array:

- o 'registrant' -- The entity object instance is the registrant of the registration.
- o 'tech' -- The entity object instance is a technical contact for the registration.

- o 'admin' -- The entity object instance is an administrative contact for the registrant.
- o 'abuse' -- The entity object instance handles network abuse issues on behalf of the registrant of the registration.
- o 'billing' -- The entity object instance handles payment and billing issues on behalf of the registrant of the registration.
- o 'registrar' -- The entity object instance represents the authority responsible for the registration in the registry.

[A.3.](#) URI Types

Many of the object classes have a member named 'uris', which is an array of objects with each object containing a 'type' string and a 'uri' string. These URIs represent a location to find information about the containing object. The following is a list of suggested values to use as a 'type' string for an associated URI.

- o 'self' -- Indicates the URI is a direct reference to the object instance via an RDAP query.
- o 'parent' -- Indicates a direct reference to an object instance's parent via an RDAP query. Such a reference might be used in an IP network to reference the next IP network up the tree of IP address hierarchies.
- o 'source' -- Signifies a direct reference to the registration in another RESTful protocol. As an example, both ARIN and the RIPE NCC have RESTful web services for WHOIS data that pre-exists RDAP. This type of URI would be used to reference the same data in such a system.
- o 'web' -- Signifies the web site of the containing object instance. The expectation is that a web browser would be needed to use this URI.
- o 'rdap' -- Signifies the RDAP service where RDAP queries can be issued.
- o 'held' -- Signifies a HELD location service for the registered object.
- o 'geo' -- Signifies that the URI is a geo URI related to the location of the registration.

[Appendix B](#). Suggested Data Modeling with the Entity Object Class

[B.1](#). Registrants and Contacts

This document does not provide specific object classes for registrants and contacts. Instead the entity object class may be used to represent a registrant or contact. When the entity object is embedded inside a containing object such as a domain name or IP network, the 'roles' string array can be used to signify the relationship. It is recommended that the values from [Appendix A.2](#) be used.

The following is an example of an elided containing object with an embedded entity that is both a registrant and admin contact:

```
{
  ...
  "entities" :
  [
    {
      "handle" : "XXXX",
      "names": [ "Joe Bob, Inc.", "Bobby Joe Shopping" ],
      "roles" : [ "registrant", "admin" ],
      "postalAddress" :
      [
        "123 Maple Ave",
        "Suite 90001",
        "Vancouver",
        "BC",
        "12393"
      ],
      "emails" : [ "joe@bob.com", "bob@joe.com" ],
      "phones" :
      {
        "office" : [ "999-999-999-99", "111-111-111-11" ],
        "fax" : [ "222-222-222-22" ],
        "mobile" : [ "333-333-333-33" ]
      },
      "remarks" :
      [
        "she sells seas shells",
        "down by the seashore"
      ],
      "registrationDate" : "1990-12-31T23:59:60Z",
      "lastChangedDate" : "1990-12-31T23:59:60Z",
      "lastChangedBy" : "joe@bob.com"
    }
  ]
}
```

```
]
}
```

B.2. Registrars

This document does not provide a specific object class for registrars, but like registrants and contacts (see [Appendix B.1](#)) the 'roles' string array maybe used. Additionally, a URI type can signify the registrars point of service (see [Appendix A.3](#)).

The following is an example of an elided containing object with an embedded entity that is a registrar:

```
{
  ...
  "entities" :
  [
    {
      "handle" : "XXXX",
      "names": [ "RegistrarsRUS" ],
      "roles" : [ "registrar" ],
      "postalAddress" :
      [
        "1212 Tulip Ave",
        "Suite 1",
        "Marina Del Rey",
        "CA",
        "12393-2193"
      ],
      "emails" : [ "joe@bob.com", "bob@joe.com" ],
      "phones" :
      {
        "office" : [ "999-999-999-99", "111-111-111-11" ],
        "fax" : [ "222-222-222-22" ],
        "mobile" : [ "333-333-333-33" ]
      },
      "remarks" :
      [
        "we registrar for less!"
      ],
      "uris" :
      [
        {
          "type" : "rdap",
          "uri" : "http://rdap.exmaple.com"
        },
        {
          "type" : "web",
          "uri" : "http://www.example.com"
        }
      ]
    }
  ]
}
```

[Appendix C](#). IDN Query and Response Model

Internationalized Domain Names (IDNs) differ from other types of domain names because multiple domain names as would be represented by a name in Master File format (see [[RFC4343](#)]) may be registered by a single IDN. IDNs are based on Unicode, and Unicode can have multiple means for encoding the same word depending on the character set and language being used. And the rules for determining which IDN encoding maps to a "wire-format" domain name vary from DNR to DNR.

When an IDN maps to multiple domain names, the various mappings are called variants. The DNR Domain object class ([Section 6.2](#)) represents the variants using a string array.

The following is an example of an elided DNR domain object with variants.

```
{
  "handle" : "XXXX",
  "name" : "blah.example.com",
  "variants" : [ "blah2.example.com", "blah3.example.com" ],
  ...
}
```

Because IDNs can have multiple targets in a mapping and due to the variance in DNR mapping rules, it is up to the client to reduce an IDN to a domain name in Master File format so as to narrow the lookup of the domain name to the proper subset. A query of a DNR using the IDN itself might map across multiple registrations depending on the mapping rules of the DNR.

[Appendix D](#). Postal Addresses vs Location

The postal address data listed in the entity object class ([Section 4](#)) does not necessarily represent location. The intent of this information is to provide a means to send postal mail to an entity. While in some cases it may also be the location of the entity, there is no guarantee that the two are the same. Accurate representation of location is topic unto itself, and registries wishing to show location of object instances should use the 'geo' or 'held' URI types as mentioned in [Appendix A.3](#).

Additionally, the postal address data represented in this document does not follow any specific standard for postal addresses because many registries do not keep postal address data in an internationalized standard form. Publication of such data in a format that suggests an internationalized standard form when such data is not known to be well-formed for that purpose would be misleading.

[Appendix E.](#) Changelog

Initial -00 Adopted as working group document 2012-September-18.

Authors' Addresses

Andrew Lee Newton
American Registry for Internet Numbers
3635 Concorde Parkway
Chantilly, VA 20151
US

Email: andy@arin.net
URI: <http://www.arin.net>

Scott Hollenbeck
Verisign Labs
12061 Bluemont Way
Reston, VA 20190
US

Email: shollenbeck@verisign.com
URI: <http://www.verisignlabs.com/>