draft-ietf-regext-rdap-sorting-and-paging
Review

Mario Loffredo
IIT-CNR/.it Registry
mario.loffredo@iit.cnr.it
New parameters:
- **count**: allows the user to obtain the total number of results
- **sort**: allows the user to sort the results
- **limit & offset**: allow the user to scroll the results

New properties:
- **sorting_metadata**: includes information about both current and available sort criteria
- **paging_metadata**: includes the total number of results, and paging information

RDAP conformance
- **sorting_level_0**
- **paging_level_0**

Alternative to offset
- **cursor**: an opaque string representing a logical pointer to the first result of the next page
sorting_metadata: sample

```json
{
    "rdapConformance": [ "rdap_level_0", "sorting_level_0" ],
    ...
    "sorting_metadata": {
        "currentSort": "ldhName",
        "availableSorts": [ {
            "property": "registrationDate",
            "jsonPath": "$.domainSearchResults[*].events[?(@.eventAction=="registration")].eventDate",
            "default": false,
            "links": [ {
                "value": "https://example.com/rdap/domains?name=nr.com&sort=ldhName",
                "rel": "alternate",
                "href": "https://example.com/rdap/domains?name=nr.com&sort=registrationDate",
                "title": "Result Ascending Sort Link",
                "type": "application/rdap+json"
            }, ...
            ]
        }, ...
        ]
    }, "domainSearchResults": [ ...
    }
}
```

- **REQUIRED:** property
- **OPTIONAL:** currentSort, availableSorts (at least one must be present)
- **RECOMMENDED:** jsonPath, default, links
<table>
<thead>
<tr>
<th><strong>Offset</strong></th>
<th><strong>Cursor</strong></th>
</tr>
</thead>
</table>
| {  "rdapConformance": [ "rdap_level_0", "paging_level_0" ],  ...
| {  "rdapConformance": [ "rdap_level_0", "paging_level_0" ],  ...
| {  "notices": [  {  "title": "Search query limits",  "type": "result set truncated due to excessive load",  "description": [ "search results are limited to 10" ]  }  ],  "paging_metadata": {  "totalCount": 73,  "pageCount": 10,  "offset": 10,  "nextOffset": 20,  "links": [   {   "value": "https://example.com/rdap/domains?name=*nr.com&count=true",   "rel": "next",   "href": "https://example.com/rdap/domains?name=*nr.com&limit=10&offset=10",   "title": "Result Pagination Link",   "type": "application/rdap+json"   }  ]  },  "domainSearchResults": [  ...  ]  }  |

- **OPTIONAL**: pageCount, totalCount (at least one must be present)
- **RECOMMENDED**: offset, nextOffset, links
According to the online poll, the WG agrees on the solution to provide sorting, paging and counting capabilities in RDAP.

Which default sorting properties should be defined?

Should both sorting and paging information be provided in metadata elements? If yes:
  - Does the WG agree about the proposed structures?
  - Should the metadata elements be included in a more general metadata section together with other contents (e.g. rate limits, information about server, request and response, other metadata)?

Which pagination method should be defined?
  - Only one?
  - Both?
draft-ietf-regext-rdap-partial-response

Review

Mario Loffredo
IIT-CNR/.it Registry
mario.loffredo@iit.cnr.it
The client declares a name identifying a server pre-defined set of data fields instead of declaring explicitly the data fields to get back.

New parameter:
- **fieldSet**: is a string identifying a server pre-defined set of fields

Required field sets:
- **id**: it contains only “the key field (i.e. "handle" and "ldhName")
- **brief**: it contains those elements identified in RFC7485 as “mostly supported” (i.e. by more than one third of contacted Whois services)
- **full**: it contains all the information the server can provide for a particular object

Note:
- The “objectClassName” field is implicitly included in each field set
- Field sets should be provided according to users access levels
- Server MAY add any service information (e.g. notices) and implement additional field sets
- Servers SHOULD also define a "default" field set

New properties:
- **subsetting_metadata**: includes information about both current and available field sets

RDAP conformance
- **subsetting_level_0**
{  
  "rdapConformance": [ "rdap_level_0", "subsetting_level_0" ],
  ...
  "subsetting_metadata": {
    "currentFieldSet": "brief",
    "availableFieldSets": [
      {
        "name": "id",
        "description": "Contains only the key field",
        "default": false,
        "links": [
          {
            "value": "https://example.com/rdap/domains?name=nr.com&fieldSet=brief",
            "rel": "alternate",
            "href": "https://example.com/rdap/domains?name=nr.com&fieldSet=id",
            "title": "Result Subset Link",
            "type": "application/rdap+json"
          }
        ]
      }
    ],
  }
  "domainSearchResults": [
    ...
  ]
}

- **REQUIRED**: name
- **OPTIONAL**: currentFieldSet, availableFieldSets (at least one must be present)
- **RECOMMENDED**: description, default, links
To be discussed

- According to the online poll, the WG agrees on the solution to provide a partial response in RDAP.

- Which field sets should be defined by the draft?
  - Which response elements should they contain?
  - Which ones should be required?
  - Since relationships exist in RDAP, should we define variants according to whether associated objects are returned or not?
    - Variants for brief: brief (i.e. brief-null), brief-id, brief-brief
    - Variants for full: full (i.e. full-null), full-id, full-brief, full-full

- Should the available field sets be provided in a metadata element? If yes:
  - Does the WG agree about the proposed structure?
  - Should the metadata element be included in a more general metadata section together with other contents (e.g. rate limits, information about server, request and response, other metadata)?
draft-ietf-regext-rdap-reverse-search

Review

Mario Loffredo
IIT-CNR/.it Registry
mario.loffredo@iit.cnr.it
New paths:
- **domains?entityHandle**=<reverse search pattern>
- **domains?entityFn**=<reverse search pattern>
- **domains?entityEmail**=<reverse search pattern>
- **domains?entityAddr**=<reverse search pattern>

<reverse search pattern> is a JSON object including two members:
- **value**: it represents the search pattern to be matched by the corresponding entity property. It can be:
  - for the first three paths, a string
  - for the fourth path, a JSON object, in turn, containing the information described in Section 2.4 of RFC 5733
- **role**: it is a string whose possible values are those detailed in Section 10.2.4 of RFC 7483

**Note**: value is REQUIRED, role is OPTIONAL
entityHandle={"value":"CID-40*","role":"registrant"}

entityFn={"value":"Bobby*","role":"registrant"}

entityEmail={"value":"loffredo@example.com","role":"registrant"}

entityAddr={"value":
{"cc":"CA","city":"Sydney"},"role":"registrant"}
To be discussed

- According to the online poll, the WG agrees on the solution to provide a reverse search capability in RDAP

- Which default reverse searches should be defined?

- Should we opt for a unique path, which allows the reverse search on any entity detail?
  - entityDetail={"name":"fn","value":"Bobby*","role":"registrant"}
  - entityDetail={"name":"phone","value":"+39.0503153497","role":"registrant"}

- Is the proposed JSON notation considered suitable?
  - Should we model a reverse search without using JSON?
  - Should it be evaluated given the possibility to submit complex queries in the next future?
    - Es. search all domains where tech’s email matches X AND registrant’s address matches Y
Non technical issues

- Is Privacy Considerations section considered comprehensive or does it need further amendments?
  - In my opinion: YES!
Potential issues

- Due to replacement of jCard
  - Should a name referencing a contact detail (e.g. city, cc) be compliant with the related member’s name of a new JSON contact object?

- Affected drafts
  - draft-ietf-regext-rdap-sorting-and-paging (sorting properties)
  - draft-ietf-regext-rdap-reverse-search (members of the JSON contact)
Thanks for your attention and feedbacks!