

draft-gould-regext-rdap-versioning
Versioning in the Registration Data Access
Protocol (RDAP)

Dan Keathley
dkeathley@verisign.com
IETF-116 REGEXT Working Group

Introduction

- RDAP Conformance discussion resulted in [chairs' proposal](#):
 - “Explicit support for version is not an integral part of the extension mechanism”
 - “Certainly, you can choose to include version inside the specification for your extensions, but in the context of the base protocol an extension is either supported or its not, and when supported it simply means there is a shared understanding of what is to be done between the client and server”
- Agreement with the proposal
 - RDAP extension is needed to address versioning in the future
 - RDAP profile extensions can continue to include versioning
 - "icann_rdap_response_profile_0"
 - "icann_rdap_technical_implementation_guide_0"
- RDAP Versioning Extension created based on the agreement

RDAP Extension Versioning Approach

- Facilitates coordination between client and server in order to achieve compatibility with the exchange of RDAP data
- Uses a subset of [Semantic Versioning](#)
 - Only MAJOR and MINOR version components are needed
 - Format: identifier “-” MAJOR “.” MINOR
 - Example: “versioning-1.0”
- Clients can discover supported versions of the server
- Servers may support multiple versions of an extension
- Clients may specify a particular version for the server to use

RDAP Help Response

- Adds a “versioning-help” member to the response of an RDAP help query
 - Lists each extension with versions supported by the server
- Attributes for each version include
 - Version identifier
 - Default (optional)
 - Start and end of support (optional)
 - Documentation links (optional)

RDAP Help Response Example

```
"versioning-help": [  
  {  
    "extension": "versioning",  
    "versions": [  
      {  
        "version": "versioning-0.0"  
      },  
      {  
        "version": "versioning-0.1",  
        "default": true  
      }  
    ]  
  }  
]
```

RDAP Query Response

- Adds a “versioning” member to any response that includes a field provided by an extension
 - Lists each extension and version used

```
"versioning": [  
  {  
    "extension": "versioning",  
    "version": "versioning-0.1"  
  },  
  {  
    "extension": "ext1",  
    "version": "ext1-1.0"  
  },  
  {  
    "extension": "ext2",  
    "version": "ext2-0.1"  
  }  
]
```

RDAP Versioning Query Parameter

- Use of “versioning” RDAP Query Parameter for Client to specify desired versions
- Overrides default version(s) returned in RDAP Response

`https://example.com/rdap/domain/versioning.example?
versioning=ext1-0.1,versioning-0.0`

Conclusion

- Versioning in RDAP standardizes how extensions and their APIs evolve in a safe manner
 - Empowers draft authors to make non-backward compatible changes
 - Allows servers to iterate without disrupting clients
 - Insulates clients from breaking changes and provides a safe upgrade path
- Please review the draft and provide feedback on the mailing list with the goal of adopting it as a working group document