

regext
Internet-Draft
Intended status: Standards Track
Expires: 8 January 2026

J. Galvin
Identity Digital
M. Bauland
Knipp Medien und Kommunikation GmbH
7 July 2025

Domain variant support for EPP
draft-galvin-regext-epp-variants-02

Abstract

This document defines an EPP extension allowing clients to learn about and manipulate variant groups of domains, ie. groups of domains whose names are equivalent in a registry-defined way and are tied to a single registrant.

Discussion Venues

This note is to be removed before publishing as an RFC.

Discussion of this document takes place on the Registration Protocols Extensions Working Group mailing list (regext@ietf.org), which is archived at <https://mailarchive.ietf.org/arch/browse/regext/>.

Source for this draft and an issue tracker can be found at <https://github.com/arnt/regext-epp-variants>.

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 <https://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 8 January 2026.

Copyright Notice

Copyright (c) 2025 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction	2
2. Requirements Language	4
3. Terms	4
4. EPP Commands	6
4.1. EPP <check> command	6
5. EPP <info> command	9
6. EPP <transfer> command	10
7. EPP <create> command	10
8. EPP <update> command	11
9. EPP <delete> command	12
10. EPP renew command	12
11. EPP <transfer> query command	13
12. Result codes	13
13. Acknowledgements	14
14. Security Considerations	14
15. IANA Considerations	14
16. References	14
16.1. Normative References	14
16.2. Informative References	14
Appendix A. Open issues	15
Authors' Addresses	15

1. Introduction

EPP is defined in [RFC5730]. EPP commands were developed to operate on a single object at a time. This document defines an EPP extension allowing clients to learn about and manipulate variant groups of objects that have a characteristic that makes them equivalent.

Similar to EPP, the principal motivation for this extension was to provide a standard Internet domain name registration extension for use between domain name registrars and domain name registries. This

protocol provides a means of interaction between a registrar's applications and registry applications. It is expected that this protocol will have additional uses beyond domain name registration.

As an example, the problem being considered is that spelling is not necessarily uniform. For example, an 豎 and an e may be regarded as equivalent in some languages, and as different in others.

Some registries plan to support this explicitly, with groups of variant domains that can only be registered by the same registrant. Having the same registrant is most commonly considered essential for equivalence, since if the domains are intended to be equivalent then the responsibility of maintaining that equivalence must be present. This is a specific example of the more general "Same Entity Principle", which in this specification is defined to mean that a variant group MUST be created, managed, and deleted by the same entity. From a registry perspective the same entity would be registrar; from the registrar's perspective the same entity would be the registrant.

This document does not define a variant or a group of variants, i.e., this document does define what makes the domains in the variant group equivalent. A registry policy MUST exist that specifies both that a registry supports variant groups and that defines what domains are eligible to be a member of a variant group. This policy MUST be agreed between a registry and a registrar. The policy and the establishment of the agreement is outside the scope of this specification.

A common policy expression among domain registries and registrars is to define variants in terms of the script or language in use for an Internalized Domain Name (IDN). IDN variants can arise when different characters or sequences of characters in an IDN are considered equivalent in a particular language or script. Standard Label Generation Rules (LGRs) are used to specify the IDN table that establishes the variant relationships. This common policy is presumed and used as an example in this specification.

With this extension, registering a domain creates a variant group and the first domain registered in the group becomes the group's Primary Domain. The creation of the Primary Domain MAY establish rules or guidelines regarding the domains that are eligible to be a member of the group, e.g., an LGR, an IDN table, and a Primary Domain taken together will define a variant group.

Subsequent domains in the same group can only be registered by the same registrar, which asserts that it is acting on behalf of the same registrant. Each domain in a variant group may be the target of any EPP command, with the following restrictions.

- * A <transfer> of any domain in any variant group always acts on the entire group. This is required to ensure that the variant group is always registered by the same registrant and managed via the same registrar. Registry policy MAY be impose additional restrictions.
- * The <delete> of a Primary Domain in any variant group always acts on the entire group. This is required to support the option where the Primary Domain establishes the rules or guidelines for the creation of other domains in the group.

This extension is backwards compatible with registrars that do not support variant groups. Specifically, this extension supports registries that do support variant groups interacting with registrars that do not support variant groups. Registrars that do not support variants that attempt to act on a member of a variant group inappropriately will receive a compatible error response with which they can continue to function. The compatible error response may not provide sufficient detail to fully understand the rejection but will be sufficient to ensure continuation of normal operations.

The remainder of this document describes the specific details.

TODO: login exchange of variant-aware

TODO: discussion of reference to EPP Extensibility and Extension Analysis https://docs.google.com/document/d/1WR00oB43XZCDqD0zvRvRajuWAq_9wQ3c0RrFKlGC3So/edit?tab=t.0

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Terms

Allocated variant: A domain that has been created in the registry, and which is tied to an existing primary domain.

Allocatable variant: A domain that has not been allocated but is allocatable (according to the LGR), and which is conceptually tied to an existing primary domain.

Activated variant: An allocated domain that is in the DNS.

Blocked domain: A domain that cannot be allocated due to its disposition value in relation to the primary domain name.

Disposition Value: While a variant relationship is symmetric it has exactly one of two disposition values which are not necessarily symmetric. A variant can be "allocatable" (i.e., available for the same entity) or "blocked" (i.e., not available for anybody).

Exempted domain: A preexisting domain that exists as a stand-alone domain, but would be part of a variant group if it were allocated now. Exempted domains may exist with any registrant at any registrar. The exemption stops as soon as at most 1 allocated domain remains within a variant group.

IDN Table: The combined information about what characters (code points) are available for domain registration as well as the variant relationships between those code points. IDN tables can be defined via RFC3743 or RFC4290 or LGRs (RFC7940). The latter one SHOULD be used as it also allows the formal definition of context rules, which is lacking in the former ones.

Label Generation Rules (LGR): The preferred way of defining IDN tables. Among others, they define the variant relationships as well as their disposition values (blocked or allocatable). The formal definition of LGRs can be found in RFC7940.

Primary domain: The chronologically first domain in a variant group. While the variant relationship is symmetric, its disposition value is not. It can either be blocked or allocatable. The primary domain name therefore partitions the variant group into allocatable variants and blocked variants. In case of variant TLDs, there can be a primary domain per TLD.

Same Entity Principle: A requirement that all domains within a variant group either belong to the same entity (i.e., the same registrant via the same registrar) or are withheld for that entity. No other entity is allowed to activate any domain within the same variant group.

Variant domain: A domain in a variant group which is not a primary domain.

Variant group: An implicit set of domains defined by the Label Generation Rule (LGR) of the registry. The variant relationship is symmetric and transitive. Hence, an arbitrary element of a variant set defines the whole set. This set is not expressed explicitly in EPP, because it can be impractically large. At the time of writing, a domain is registered whose variant set would contain 10¹⁰ variants.

TODO: Do we need to differentiate between allocated and activated?

TODO: Does it make any difference, whether a variant has DNS name servers or not?

4. EPP Commands

4.1. EPP <check> command

TODO: probably better not to have a command extension. For Registrars, it would be difficult to determine the suitable primary domain. Therefore, it is better to not ask them to send it, but rather return the appropriate primary domain name in the response.

This extension defines a new command called the Variant Check Command that defines an additional Primary Domain name element for the EPP <check> command.

The command MAY contain an <extension> element, which MUST contain a <var:check> element. The <var:check> element MUST contain one <var:primary> element containing the intended primary domain.

```
C: <?xml version="1.0" encoding="utf-8" standalone="no"?>
C: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
C:   <command>
C:     <check>
C:       <domain:check
C:         xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
C:           <domain:name>examplev1.com</domain:name>
C:           <domain:name>examplev1.net</domain:name>
C:           <domain:name>examplev1.xyz</domain:name>
C:         </domain:check>
C:       </check>
C:     <extension>
C:       <var:check xmlns:var="urn:ietf:params:xml:ns:epp:variants-1.0">
C:         <var:primary>example.com</var:primary>
C:       </var:check>
C:     </extension>
C:     <clTRID>ABC-12345</clTRID>
C:   </command>
C: </epp>
```

When the server receives a <check> command from a variant-agnostic client and any domain within the checked domain's variant group is allocated (or at least one exempted domain within the variant group exists) the server MUST NOT include an <extension> element. Instead, its response MUST be available = "false". The optional reason MAY be "Unavailable (except as variant)" to tell the registrar it might be available as a variant.

When the server receives a <check> command from a variant-aware client and the checked domain is part of a variant group with at least one allocated variant (or exempted domain), its response MUST contain an <extension> element, which MUST contain a child <var:chkData> element. The <fee:chkData> element MUST contain a <var:cd> element for each object referenced in the client <check> command.

Each <var:cd> (check data) element MUST contain the following child elements:

- * A <var:objID> element, which MUST match an element referenced in the client <check> command.
- * An OPTIONAL <var:primary> element.
- * A <var:status> element, which explains in more detail the availability status of the queried domain.

Example <check> response:

```
S: <?xml version="1.0" encoding="utf-8" standalone="no"?>
S: <epp xmlns="urn:ietf:params:xml:ns:epp-1.0">
S:   <response>
S:     <result code="1000">
S:       <msg>Command completed successfully</msg>
S:     </result>
S:     <resData>
S:       <domain:chkData
S:         xmlns:domain="urn:ietf:params:xml:ns:domain-1.0">
S:           <domain:cd>
S:             <domain:name avail="1">examplev1.com</domain:name>
S:           </domain:cd>
S:           <domain:cd>
S:             <domain:name avail="0">examplev1.net</domain:name>
S:           </domain:cd>
S:           <domain:cd>
S:             <domain:name avail="0">examplev1.tel</domain:name>
S:           </domain:cd>
S:           <domain:cd>
```

```
S:      <domain:name avail="0">examplev1.swiss</domain:name>
S:      </domain:cd>
S:      </domain:chkData>
S:      </resData>
S:      <extension>
S:      <var:chkData
S:          xmlns:var="urn:ietf:params:xml:ns:epp:variants-1.0">
S:          <var:cd avail="1">
S:              <var:objID>example.com</var:objID>
S:              <var:primary>example.com</var:primary>
S:              <var:status>AllocatableVariant</var:status>
S:          </var:cd>
S:          <var:cd avail="0">
S:              <var:objID>example.net</var:objID>
S:              <var:status>NotSameEntity</var:status>
S:          </var:cd>
S:          <var:cd avail="0">
S:              <var:objID>example.tel</var:objID>
S:              <var:status>Blocked</var:status>
S:          </var:cd>
S:          <var:cd avail="0">
S:              <var:objID>example.swiss</var:objID>
S:              <var:status>PendingTransfer</var:status>
S:          </var:cd>
S:      </var:chkData>
S:      </extension>
S:      <trID>
S:          <clTRID>ABC-12345</clTRID>
S:          <svTRID>54322-XYZ</svTRID>
S:      </trID>
S:      </response>
S: </epp>
```

The EPP <check> command may return five new results:

- * The domain cannot be provisioned because it is a variant of a Primary Domain, and the Primary Domain belongs to a different client => NotSameEntity
- * The domain cannot be provisioned because its disposition value is blocked. => Blocked
- * The domain cannot be provisioned because it is a variant of at least one exempted domain. => Exempted
- * The domain cannot be provisioned because it is a variant in a group that is currently being transferred to a different registrar. => PendingTransfer

- * Additional custom value that may be used for server peculiarities.
=> Custom

5. EPP <info> command

For variant-agnostic clients there is no change to the standard behaviour. The response contains the actual data of the domain, independent of the fact whether it is a variant or not, in addition to the following:

- * if the variant-agnostic registrar is inquiring about a non-allocated variant, the response SHOULD be the same as the registrar inquiring about a reserved name. If you don't have a policy, suggest a policy.

TODO: XML example of response?

For variant-aware clients, the EPP <info> command is not extended, but its response is extended if the <info> command concerns a variant domain, i.e., at least two domains within a variant group have been activated. The response then always MUST include all primary domain names across all activated variant TLDs. Optionally the response may include the list of all activated variants (across all variant TLDs).

In case a Primary Domain name is queried in the <info> command, the list of activated variants within the same TLD MUST be returned.

In other words: * If you ask about a primary domain name * you MUST return all primary labels in all variant TLDs * you MUST return all activated variants in that TLD * you MAY return all activated variants in all variant TLDs

- * if you ask about a variant domain
 - you MUST return the primary label for that variant
 - you MUST return all primary labels in all variant TLDs
 - you MAY return all activated variants in that TLD
 - you MAY return all activated variants in all variant TLDs

The main part of the response MUST contain the actual data of the queried domain name (contacts, hosts, status values, etc.)

TODO: check whether EPP spec says anything about the alignment of check and info.

6. EPP <transfer> command

If a variant-agnostic client initiates a transfer-in of a variant domain, i.e., at least two domains in a variant group have been activated, the transfer request MUST be denied using 2305 "Object status prohibits operation".

TODO: xml example

If a variant-aware client initiates a transfer-in of a variant domain, i.e., at least two domains in a variant group have been activated, the transfer request MUST include an extension specifying the Primary Domain for the indicated variant domain, including if the Primary Domain is the domain indicated in the transfer-in. If the extension is not present the transfer request MUST be denied using 2003 "Required parameter missing".

The <transfer> response is extended MUST include the complete group of all activated variants formed by combining all variant groups from all variant TLDs.

TODO: It must be ensured that the poll message to the losing registrar also contains the full list of domains that will be transferred together with the primary domain.

TODO: xml example

7. EPP <create> command

The EPP <create> command's standard task is to provision a new domain. If the domain to be created is part of a variant group, the command MUST be rejected as follows.

- * If the client is variant-agnostic, the response SHOULD be the same as if the domain to be created is reserved.
- * If the client is variant-aware, the response MUST indicate that it is an inappropriate use of the command.

The <create> MUST only be used to create the Primary Domain. The task of converting an allocatable domain into an allocated domain MUST be performed using the <update> command.

If a client initiates the creation of a domain that does not exist and is not a member of any variant group, then the following actions are REQUIRED.

- * If the create is otherwise permitted and the domain could be a Primary Domain, the server MUST ensure that all eligible members of the variant group are prevented from creation or allocation until such time as the domain is expressly indicated by the client to be a Primary Domain.
- * The server MUST act on the create and respond to the client as if the domain is a new domain.

__ TODO__: make clear that registering the first domain within any variant group must not be rejected. The rejection only happens if at least one other domain of that variant group already exists.

The EPP <create> command may have five new errors, as described in the <check> section above.

TODO: check alignment of the new error codes

8. EPP <update> command

The EPP <update> command is extended to cover two new tasks:

- * Activating a variant domain in an existing variant group
- * Converting an Exempted Domain into a Primary Domain and optionally converting other Exempted Domains that are eligible to be in the variant group of the stated Primary Domain to be activated domains of the variant group.

This extended <update> is not valid for use by a variant-agnostic client. Any use by a variant-agnostic client MUST be rejected.

This rest of this section specifies behavior when variant-aware servers and clients are interacting.

When a client wishes to provision a new domain in a variant group, it uses the <update> command rather than the <create> command. This informs the server that the client understands that the task is to provision a variant domain rather than a new domain, and asserts that the two domains belong to the same registrant.

The <update> command MUST specify the domain to be activated and include an element specifying the Primary Domain that identifies the correct variant group for the domain.

Note that depending on registry policy, the variant domain may share attributes with the Primary Domain, e.g., nameservers. A registry policy MAY specify rules or guidelines for the set of elements required or permitted for a variant domain according to the Primary Domain.

TODO: xml example

If a client wishes to convert an exempted domain into a member of a variant group as an allocated variant, it issues an <update> command including an element with both the Primary Domain and separately the list of exempted domains, which the client thereby asserts belong to the same registrant. Note that the client MUST include all related exempted domains in the list or the server MUST reject the command. The response MUST include the complete list of exempted domains for the client.

TODO: xml example

9. EPP <delete> command

If a variant-agnostic client issues a <delete> command there is no change from the standard functionality.

If a variant-aware client issues a <delete> command, the command is extended to REQUIRE the client to include an extension indicating the Primary Domain of the domain being deleted, which the client thereby asserts that both domains belong to the same registrant. If a Primary Domain is being deleted then the same domain name MUST be specified in the extension.

If a Primary Domain is to be deleted, the <delete> command is extended to include the deletion of all variant domains in all variant groups in all Variant TLDs. The response MUST include a list of all the allocated domains in all variant groups that were deleted.

TODO: xml example

10. EPP renew command

The EPP renew command is not extended.

The server MAY reject renewals while a variant group is being transferred.

11. EPP <transfer> query command

The EPP <transfer> query command is not extended.

Note that because variant groups are transferred as a group, the result of the a <transfer> query command is necessarily the same for all domains in a group. Therefore, the result of <transfer> query command for any domain in a variant group applies to all domains in the group.

12. Result codes

The following additional result codes are defined:

23x1: Change impossible because of a transfer in progress.

23x2: Change impossible because something is not a variant.

This error code is used when a change presupposes that two domains belong to the same variant group, but the EPP server's implementation disagrees.

23x3: Change impossible due to invalid primary domain

This error code is used when the primary domain specified in the command is not registered, or is not registered via this registrar.

23x4: Change impossible due to unspecified primary domain

This error code is used when a command needs to specify a primary domain, and does not.

23x5: Specified domain is exempted

This error code is used when a domain specifies a primary domain, and the change is impossible because the specified domain is exempted instead.

23x6: Specified domain is allocatable, but not by you.

This result code is used when a domain is a member of a variant set, and the command did not refer to the primary domain.

13. Acknowledgements

The design of this extension is almost completely based on work done by and decisions made by the [EPDP] committee, which was reviewed by a small technical design team chaired by James Galvin. Members of this team included Dennis Tan, Rick Wilhelm, Edmon Chung, and Jennifer Chung. This text (in RFC format) was initially written by Arnt Gulbrandsen based on a conference presentation by James Galvin.

[YOUR NAME HERE] have reviewed it and provided helpful comments or contributed in other ways.

14. Security Considerations

If two domains are different according to the DNS rules and identical in the eyes of the intended audience, then the audience may be confused. Confusion can always have security-related effects.

This extension expresses the relationships between variants clearly, making it a little more difficult for a would-be impersonator to register a variant of another registrant's existing domain.

15. IANA Considerations

16. References

16.1. Normative References

- [RFC5730] Hollenbeck, S., "Extensible Provisioning Protocol (EPP)", STD 69, RFC 5730, DOI 10.17487/RFC5730, August 2009, <<https://www.rfc-editor.org/rfc/rfc5730>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/rfc/rfc2119>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/rfc/rfc8174>>.

16.2. Informative References

- [RFC7940] Davies, K. and A. Freytag, "Representing Label Generation Rulesets Using XML", RFC 7940, DOI 10.17487/RFC7940, August 2016, <<https://www.rfc-editor.org/rfc/rfc7940>>.

[EPDP] ICANN, "Phase 2 Initial Report of the EPDP on Internationalized Domain Names", 2024, <<https://www.icann.org/en/public-comment/proceeding/phase-2-initial-report-of-the-epdp-on-internationalized-domain-names-11-04-2024>>.

Appendix A. Open issues

Open issue: Assign numbers to the error codes, properly.

Open issue: Not clear that there are any security considerations here 竇 the relationships between the domains may have some, but those exist outside EPP, EPP merely describes them. In Italian, *caffè* and *caff* are variants of the same thing, it's not clear that linking them in a protocol affects security in any way.

Open issue: Check how to insert a DS record in a variant domain.

Open issue: Can a unicode upgrade cause domains to become exempted? Yes, I think, and the terminology covers it, but as of now, it's difficult for the EPP client to understand the situation. Extending the <info> command would help here, perhaps.

Open issue: Creating a primary domain now consists of a two-step process, <create> and then <update>. The first step turns all variants into blocked domains, the second makes them allocatable. It's not clear to me why the two-step process is desirable, compared to a one-step process where a <create> command creates a primary domain if the variant group contains other domains than that one. That needs a couple of sentences of explanation, or else a change.

Open issue: <Delete> now cascades and deletes many domains. Should it instead turn any variant domains into exempted domains?

Open issue: Should the <info> of the primary domain also return the list of allocated variants? Probably not — at the moment, this extension has the client send a set that the server checks, and the server may need to generate a set for checking, but the server does not need to generate a list for returning.

Authors' Addresses

James Galvin
Identity Digital
Bellevue, Washington,
United States of America
Email: jgalvin@identity.digital

Michael Bauland
Knipp Medien und Kommunikation GmbH
Dortmund
Germany
Email: michael.bauland@knipp.de