

# Update and Delete of Resource Data Description Documents in Learning Registry (V 0.3)

Jim Klo <[jim.klo@sri.com](mailto:jim.klo@sri.com)>, SRI International

## Overview

This document describes how Resource Data Description Documents (RD3), also commonly referred to as “Learning Registry envelopes”, may be updated to newer versions or deleted from Learning Registry in a federated manner. In order to maintain the spirit of the current Learning Registry [Specification](#) and [Implementation](#), updates and deletes are performed by publishing a replacement RD3 for existing RD3 where the contents will determine if new document signals a delete or an update.

## Table of Contents

- [Update and Delete of  
Resource Data Description Documents  
in Learning Registry \(Unversioned DRAFT\)](#)
- [Overview](#)
- [Table of Contents](#)
- [Conventions](#)
- [Resource Data Description Document for Update](#)
- [Resource Data Description Document for Delete](#)
- [Publishing Node Responsibilities for Update and Delete](#)
- [Tombstones](#)
- [Distribute and Tombstones](#)
- [Recommendations](#)
  - [RD3 Identifier Convention](#)
  - [Non Canonical Resource Locators](#)

## Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#).

## Resource Data Description Document for Update

Updates will be signaled to the node in the following manner:

1. A property on the replacement RD3 named “replaces” SHALL exist. This property MUST



This work is licensed under a [Creative Commons Attribution-ShareAlike 3.0 Unported License](#).

be an array containing zero or more RD3 “doc\_ID”s that SHOULD be replaced by the new one.

- a. NB: When “replaces” array contains zero items - no update operation is performed. This is equivalent to not performing any update as no replacement RD3 “doc\_ID” is specified.
  - b. NB: “replaces” MAY contain “doc\_ID”s for all previous revisions. To ensure that all previous revisions are tombstoned it is good practice to include all doc\_ID revisions, however only the most recent version is necessary.
2. The signing key used to sign the replacement RD3:
    - a. SHOULD match the signing key used to sign original RD3 specified in the “replaces” property.
    - b. MAY be signed using an alternate signing key as permitted by node policy.

NB: Each replacement RD3 MUST have a unique doc\_ID; It is RECOMMENDED that RD3 “doc\_ID” follow the convention outlined in this document, however not required.

The data model for signaling RD3 updates SHALL adhere to the following [JSON Schema](#):

```
{
  "name": "Resource Data Description Data Model for Update",
  "extends": "Resource Data Description Data Model",
  "properties": {
    "doc_version": {
      "type": "string",
      "enum": [ "0.24.0" ],
      "required": true,
      "description": "Literal document version for this data model"
    },
    "replaces": {
      "type": "array",
      "required": true,
      "minItems": 0,
      "items": {
        "type": "string"
      },
      "description": "A list of doc_ID that refer to RD3 that this RD3 should
replace"
    },
    "payload_placement": {
      "type": "string",
      "required": true,
      "enum": [ "inline", "linked", "attached" ],

```



“description”: “Fixed vocabulary, inline: resource data is in an object that follows, linked: resource\_data is at the link provided, attached: resource data is an attachment (currently unsupported)”

```
    }  
  }  
}
```

NB: The schema for Resource Data Description Data Model is defined in the [Resource Data Data Model: Learning Registry Technical Specification V RM:0.49.0](#) in a non-JSON Schema format.

### Resource Data Description Document for Delete

Deletes SHALL operate in the same manner as RD3 for update. The published RD3 MUST extend the [Resource Data Description Data Model for Update](#) as follows:

```
{  
  "name": "Resource Data Description Data Model for Delete",  
  "extends": "Resource Data Description Data Model for Update",  
  "properties": {  
    "payload_placement": {  
      "type": "string",  
      "enum": ["none"],  
      "description": "limited vocabulary, none: no resource data follows,  
signaling deletion"  
    },  
    "payload_schema": {  
      "required": false,  
      "description": "since this signals a delete, no payload schema is required."  
    },  
    "resource_locator": {  
      "required": false,  
      "description": "since this signals a delete, no resource_locator is required,  
as resource may no longer exist."  
    }  
  }  
}
```

### Publishing Node Responsibilities for Update and Delete

1. Upon receipt of a replacement RD3 containing a valid “replaces” property, the node MUST:
  - a. Validate that signing key of the replacement RD3 matches original RD3 specified in “replaces” property and undertake the update/delete operation if the



This work is licensed under a [Creative Commons Attribution-ShareAlike 3.0 Unported License](#).

verification match. The node may also optionally support additional verification/authorization for updates for sign-by-proxy or enhanced authorization keys as follows:

- i. In the case where node sign by proxy was used, the replacement document must be published at the same node using a RD3 sign-by-proxy algorithm that the node can validate against the original RD3. The node **MUST** validate the identity of the “submitter” and **SHOULD** reject the replacement RD3 if submitters that do not match the original RD3.
    1. Node has responsibility to perform due diligence for ensuring that replacement is allowed for the RD3 for the submitter.
    2. Nodes providing sign by proxy should mark RD3 in a manner that permits node to match identity at a future point in time
      - a. i.e. Encode the authenticated submitter identity of the document into the RD3’s “identity” property before signing.
  - ii. The node **MAY** grant certain keys a *superuser* status that permits replacement docs signed using the *superuser* key to replace RD3 whose signing key does not match the *superuser* key.
  - iii. The node **MAY** implement other arbitrary key update permission strategies, such as mapping a new key to an older key, where the owner is known to be the same.
2. For each RD3 “doc\_ID” specified in the “replaces” property, the node **SHALL**:
    - a. if RD3 exists locally convert it into a *tombstone* using the same “doc\_ID” as the RD3.
    - b. if RD3 does not exist, create a *tombstone* for the specified “doc\_ID”.

NB: Basic Publish and SWORD Publish services RD3 validation **MUST** be updated to honor the revisions to the data model accordingly.

NB: Federation and processing of trusted *superuser* keys could be done by publishing an assertion into LR containing a list of node trusted super keys signed by the trusting node. Likewise a simpler solution might be to just check the OpenPGP signature chain. Node could countersign trusted keys instead.

## Tombstones

A *tombstone* is node local document that acts a terminal marker for an RD3. It **SHALL** be the only allowed mutability of a RD3. A *tombstone* **MUST NOT** be distributed to other nodes.

The data model for a tombstone **SHALL** adhere to the following [JSON Schema](#):

```
{  
  "name": "Tombstone",
```



This work is licensed under a [Creative Commons Attribution-ShareAlike 3.0 Unported License](#).

```

"properties": {
  "doc_ID": {
    "type": "string",
    "description": "The document id of the RD3 that this tombstone replaces",
    "required": true
  },
  "doc_type": {
    "type": "string",
    "enum": [ "tombstone" ],
    "required": true,
    "description": "The document type to indicate the node such that it if
excluded from default harvest"
  },
  "doc_version": {
    "type": "string",
    "enum": [ "0.24.0" ],
    "required": true,
    "description": "The literal for the version number of this tombstone
document"
  },
  "replaces": {
    "type": "array",
    "required": false,
    "items": {
      "type": "string"
    },
    "description": "The 'replaces' property of the RD3 that this tombstone
replaces."
  },
  "replaced_by": {
    "type": "object"
    "required": true,
    "properties": {
      "doc_ID": {
        "type": "string",
        "required": true,
        "description": "The doc_ID of the first RD3 encountered by
node to signal tombstone creation"
      },
      "public_key_fingerprint": {
        "type": "string",
        "required": true,
        "description": "The OpenPGP fingerprint of the signing key

```



used to verify the first RD3 that signaled tombstone”

```
    },
    "public_key_locations": {
      "type": "array",
      "required": true,
      "minItems": 1,
      "items": {
        "type": "string"
      },
      "description": "A list of URLs that the public_key used for
        verifying the first RD3 that signaled tombstone may be retrieved"
    }
  },
  "create_timestamp": {
    "type": "string",
    "format": "date-time",
    "required": true,
    "description": "timestamp of when tombstone was first created"
  },
  "resource_locator": {
    "type": [ "string", "array" ],
    "items": {
      "type": "string"
    },
    "description": "The resource locator(s) of the RD3 this tombstone marks."
  },
  "do_not_distribute": {
    "type": "boolean",
    "enum": [ true ],
    "required": true,
    "description": "Flag to indicate that this instance MUST not be distributed
to other nodes."
  }
}
```

NB: ALL harvest services SHOULD be updated to exclude tombstones.

NB: Any harvest service MAY be updated to allow explicit request for tombstone by “doc\_ID”. It is RECOMMENDED this feature only be available via Basic Harvest and Obtain.



This work is licensed under a [Creative Commons Attribution-ShareAlike 3.0 Unported License](https://creativecommons.org/licenses/by-sa/3.0/).

## Distribute and Tombstones

When an RD3 is distributed to a node, the following logic should be applied to accommodate Update and Delete RD3 and *tombstones*.

1. If a RD3 arrives via distribution and is an Update or Delete RD3, then honor the previously described validation outlined for Update and Delete.
2. If a *tombstone* pre-exists for an RD3:
  - a. If signing key fingerprint in the tombstone matches the signing key of the RD3, reject the distributed RD3.
  - b. If signing key fingerprint in the tombstone differs from the signing key of the RD3, then:
    - i. if signing key fingerprint is a *superuser key*, then reject the RD3.
    - ii. if signing key fingerprint is an arbitrary key that can be mapped to RD3 signing key using node specific policy, then reject the RD3.
    - iii. if validation fails, replace tombstone with RD3.

## Recommendations

### *RD3 Identifier Convention*

In order to simplify management of RD3 update and delete documents it is RECOMMENDED that the RD3 “doc\_ID” property SHOULD be a Uniform Resource Name and SHALL comply with [URN Syntax \(RFC 2141\)](#) but MAY NOT adhere to standard usage through the use of officially registered namespaces ( per [RFC 3406](#) ).

Per BNF syntax, URNs are described as:

<URN> ::= “urn:” <NID> “:” <NSS>

rendered as:

urn:<NID>:<NSS>

where <NID> is the namespace identifier, and <NSS> is the namespace-specific string.

It is RECOMMENDED that the namespace identifier be a [domain name](#) that is controlled by the RD3 submitter.

NB: A submitter should choose an identifier that is contextual to the submitter, and not the resource. For example:

- *Aggregator.org curates a collection of resources, some provided by sample.edu.*
- *Sample.edu has a learning resource 1 at: <http://sample.edu/resource1>.*
- *Aggregator.org has information about learning resource 1 stored in it’s db with locally unique id “qwerty123”, and was last updated on Dec 21, 2021 00:00:00 Z*
- *Aggregator.org should use “urn:aggregator.org:qwerty123:2012-12-21T00:00:00Z” as*



*the doc\_ID for the data it submits into Learning Registry as this identifier is contextual to the curator of the metadata, not the resource.*

It is RECOMMENDED that the namespace-specific string is a composite of a domain native identifier and version identifier. Per BNF syntax, NSS described as:

`<NSS> ::= <DNI> ":" <DVI>`

rendered as:

`<DNI>:<DVI>`

where <DNI> is the domain native identifier and <DVI> is the domain version identifier.

Example instances of the recommended convention are as follows:

`urn:anywhere.org:1:2`

`urn:example.com:deadbeef:v1`

`urn:kahnacademy.org:counting-to-ten:2012-11-28T00:00.01234`

NB: The usage of URNs permit publishers to leverage their existing native, domain unique, identifiers and versioning strategies without having to track a unique LR identifier.

NB: Enhanced publish services could be may aware of this solution and be able to locate previous versions of a document automatically. However there is no specification for doing this.

### *Non Canonical Resource Locators*

In order reduce duplication of RD3 due to resources not maintaining canonical resource locators, as well as to simplify the expression and identification of “same as” relationships within Learning Registry by the submitter. It is RECOMMENDED that the schema for Resource Data Description Data Model, defined in the [Resource Data Data Model: Learning Registry Technical Specification V RM:0.49.0](#) in a non-JSON Schema format, be amended as follows in JSON Scheme:

```
{
  "name": "Resource Data Description Data Model",
  "extends": "Resource Data Description Data Model v0.23.0",
  "properties": {
    "doc_version": {
      "type": "string",
      "enum": ["0.24.0"]
    },
    "resource_locator": {
      "type": ["string", "array"],
      "minItems": 1,
      "items": {
        "type": "string"
      }
    }
  }
}
```





```

        "required": true,
        "description": "When resource_locator is a string, it indicates a canonical
locator, if array, one item indicates canonical, more than one indicates
non-canonical"
    }
}
}

```

NB: CouchDB views for harvest would have to be updated to index multiple resource\_locators, however should not change harvest API.

NB: Publish services would need to be updated to permit array data type for resource\_locator property.

NB: The values in a "resource\_locator" array SHOULD only reference locations the resource exists and can be accessed. A "resource\_locator" should not include landing pages or proxies that do not permit interaction or access to the resource.

## Revision History

Version	Date	Author/Editor	Change
N/A	2012/12/06	Jim Klo <jim.klo@sri.com>	Initial internal release for comment.
Draft	2012/12/06	Jim Klo <jim.klo@sri.com>	Misc edits per Steve Midgley.
V 0.2	2012/12/06	Jim Klo <jim.klo@sri.com>	Added section regarding distribution and tombstones. [1] Added "replaces" property to tombstone. [2]
	2012/12/07	Jim Klo <jim.klo@sri.com>	No Spec Changes. Added notes to Non Canonical Resource Locators. [3]
V 0.3	2012/12/12	Jim Klo <jim.klo@sri.com>	Added note regarding best practice for resource_locator[4]. Clarified the "manner" in which RD3 that were submitted as sign-by-proxy can be published [5]. Minor grammar edits.
	2012/14/12	Jim Klo <jim.klo@sri.com>	Reworded note for clarity and intent. [6] Added note regarding best practice for URN selection. [7] Added note regarding federation of superuser keys [8]

