FAA-STD-064
May 1, 2009

U.S. Department
of Transportation

**Federal Aviation
Administration**

U.S. Department of Transportation

Federal Aviation Administration

Standard Practice

WEB SERVICE REGISTRATION

FOREWORD

This standard is approved for use by all Departments of the Federal Aviation Administration (FAA).

This standard contains the requirements for registering services, including web services. It specifies the minimum acceptable content for documenting services within the FAA.

This standard is one of several related standards that together define FAA's requirements for describing and registering services. A future Handbook will provide guidance for using these standards in concert with the FAA Enterprise Architecture, FAA Order 1375.1 Information/Data Management, and other relevant FAA initiatives.

This standard has been prepared in accordance with FAA-STD-005F.

Comments, suggestions, or questions on this document shall be addressed to:

Federal Aviation Administration
ATO-P/SE
800 Independence Avenue, SW
Washington, DC 20591

Table of Contents

List of Figures

1 SCOPE

This standard specifies the content and structure of the information needed to describe services registered in an FAA-affiliated service registry. It applies to the formulation of data concepts, including their meanings and their relationships, to be shared among people and machines.

The nature of a service registration process is very broad, as it could be expected to contain everything from a simple version control policy to enterprise architectural assertions. The following is intended to more tightly constrain the in-scope requirements so as to prevent future confusion.

a. This standard does not prescribe or limit the technological solutions for a registry. Rather, it seeks to provide a syntactic basis for the registration process that is flexible enough to accommodate different registry implementations for use across multiple business domains.

b. This standard does not specify any configuration management (CM) or quality assurance (QA) policies, rules or assertions that an FAA service registration process may be subjected to.

c. This standard does not implement or suggest an architectural or semantic schema specific to any FAA organization or Line of Business, but rather seeks to provide an extensible and flexible approach that may fit any FAA business domain.

d. This standard does not specify any governance policies, rules or procedures that an FAA service registry may be subjected to.

1.1 Purpose

The purpose of this standard is to provide a set of instructions for registering service metadata in an FAA-affiliated registry. These instructions are based on widely-used normative industry guidelines and best practices.

Information registered in a registry is used to characterize a service for purposes such as advertisement, discovery, and selection. This information may be published in various kinds of registries, discovered using various tools, and selected using various kinds of matchmaking techniques.

1.2 Applicability

This standard is applicable to all FAA programs responsible for registering information about services in an FAA-affiliated service registry. It is intended to be used by the FAA and associated contractors responsible for specifying service descriptions that will be registered in any FAA-affiliated service registry.

2 APPLICABLE DOCUMENTS

2.1 Government documents

The following form a part of this document to the extent specified herein. Unless otherwise specified, the issues of these documents are those cited in the solicitation or contract.

FAA Order 1375.1d, Information/Data Management, July 2006
http://www.faa.gov/about/office_org/headquarters_offices/aio/library/

FAA-STD-063: XML Namespaces (pending approval)
(Location to be supplied)

United States Postal Service Publication 28 – Postal Addressing Standards
http://pe.usps.gov/text/pub28/welcome.htm

2.2 Non-Government standards and other publications

The following form a part of this document to the extent specified herein. Unless otherwise specified, the issues of these documents are those cited in the solicitation or contract.

UDDI Version 3.0.2: UDDI Spec Technical Committee Draft, Dated 20041019
http://uddi.org/pubs/uddi-v3.0.2-20041019.htm

ebXML Registry Services and Protocols Version 3.0: OASIS Standard, 2 May, 2005
http://docs.oasis-open.org/regrep/v3.0/specs/regrep-rs-3.0-os.pdf

ebXML Registry Information Model Version 3.0: OASIS Standard, 2 May, 2005
http://docs.oasis-open.org/regrep/v3.0/specs/regrep-rim-3.0-os.pdf

RFC 2119: Key words for use in RFCs to Indicate Requirement Levels, March 1997
http://www.ietf.org/rfc/rfc2119.txt

RFC 4122: A Universally Unique Identifier (UUID) URN Namespace, July 2005
http://www.rfc-archive.org/getrfc.php?rfc=4122

ISO/IEC 11179, Information Technology – Metadata Registries (MDR), Parts 1 - 6
http://metadata-standards.org/11179/

RFC 3986: Uniform Resource Identifier (URI): Generic Syntax: Network Working Group, January 2005
http://www.rfc-editor.org/rfc/rfc3986.txt

RFC 3198: Terminology for Policy-Based Management, Network Working Group, November 2001
http://www.ietf.org/rfc/rfc3198.txt

RFC 3935: Internet Engineering Task Force (IETF) Mission Statement, October 2004
http://www.ietf.org/rfc/rfc3935.txt

OASIS Reference Model for Service Oriented Architecture 1.0: Committee Specification 1, 2 August 2006
http://www.oasis-open.org/committees/download.php/19679/soa-rm-cs.pdf

DCMI Glossary, Dublin Core Metadata Initiative, User Guide Committee, 23 April 2004
http://dublincore.org/documents/usageguide/glossary.shtml

Web Services Architecture: W3C Working Group Note 11 February 2004
http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/

Web Services Glossary, W3C Working Group Note, 11 February 2004
http://www.w3.org/TR/2004/NOTE-ws-gloss-20040211/

ISO/IEC 6523-1:1998 Structure for the identification of organizations and organization parts
http://www.iso.org/iso/catalogue_detail?csnumber=25773

NAICS: North American Industry Classification System
http://www.census.gov/epcd/www/naics.html

ISO 3166: Codes for the Representation of Names of Countries and their Subdivisions - Part 2: Country Subdivision Code
http://www.iso.org/iso/en/prods-services/iso3166ma/index.html

UNSPSC: The United Nations Standard Products and Services Code®
http://www.unspsc.org/

Web Service Description Requirements, W3C Working Draft, J. Schlimmer, October 2002
http://www.w3.org/TR/2002/WD-ws-desc-reqs-20021028/

2.3 Order of precedence

In the event of a conflict between the text of this document and the references cited herein, the text of this document takes precedence. Nothing in this document, however, supersedes applicable laws and regulations unless a specific exemption has been obtained.

3 DEFINITIONS

3.1 Key words

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. See Appendix, Key words for indicating requirement levels, for definitions.

3.2 Terms

Administrator See Registry administrator.

Agent See Software agent.

Binding An association between an interface, a concrete protocol and a data format. A binding specifies the protocol and data format to be used in transmitting messages defined by the associated interface.[WS Req]

Consumer Any client who retrieves Web services-related resource descriptions from a registry.

Discovery service A service that enables agents to retrieve Web services-related resource descriptions. [WS Arch]

Domain A collection of elements and services, administered in a coordinated fashion. [RFC 3198]

Entity An instance of a persistent data structure inside a registry that represents a particular and discrete unit of information.

FAA-affiliated registry An FAA registry that implements sharing of information with other registries within the FAA. The subject of this standard.

Internet A large, heterogeneous collection of interconnected systems that can be used for communication of many different types between any interested parties connected to it. The term includes both the "core Internet" (internet service provider networks) and "edge Internet" (corporate and private networks, often connected via firewalls, network address translation boxes, application layer gateways and similar devices). [RFC 3935]

Key A unique token used to identify and refer to an entity stored in a registry. [UDDI 3]

Lower camel case A naming convention in which a name is formed of multiple words or acronyms that are joined together as a single word, with the first letter of each of the multiple words or acronyms (except the first one) capitalized within the new word that forms the name. Example: “anFaaStandard”.

Metadata Data that defines or describes other data. [ISO 11179-1:2004]

Namespace A collection of names, identified by a URI reference [RFC 3986], that are used in XML documents as element types and attribute names. The use of XML namespaces to uniquely identify metadata terms allows those terms to be unambiguously used across applications, promoting the possibility of shared semantics. [DCMI Glossary]

Normative Of, relating to, or prescribing a norm or standard. [American Heritage Dictionary, 4th Edition]

Organization A unique framework of authority within which a person or persons act, or are designated to act, towards some purpose. [ISO/IEC 6523-1:1998, 3.10] Any department, service or other entity within an organization which needs to be identified for information exchange. [ISO/IEC 6523-1:1998, 3.2]

Publisher Any client who publishes in a registry.

Publish The act of placing one or more entities in a registry by invoking one of the publication APIs.

Publisher-assigned key Any key created by a publisher. A publisher may propose a key for a new entity at the time it is first published. If the proposed key meets the registry’s policy for publisher-assigned keys and the publish operation succeeds, the key proposed by the publisher becomes a publisher-assigned key. [UDDI 3]

Repository A collection of resources accessible over an internet.

Resource An object of information that is available on an internet and identified by a unique Uniform Resource Identifier.

Registry An enabling infrastructure that uses a formal registration process to store, catalog, and manage metadata relevant to the services. A registry supports the search, identification, and understanding of resources, as well as query capabilities.

Registry administrator A person or group who sets the policies for a registry and is responsible for service registration functions and registry operations.

Registry-assigned key Any key created by a registry. If no key is proposed for an entity at the time it is first published, the registry assigns a key. Such keys are called registry-assigned keys. [UDDI 3]

Service An implementation-independent reusable operational function that may be discovered as self-describing interfaces, and invoked using open standard protocols across networks. See also Web service.

Service-Oriented Architecture A paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. It provides a uniform means to offer, discover, interact with and use capabilities to produce desired effects consistent with measurable preconditions and expectations. [OASIS SOA RM]

Service registry See Registry.

Software agent A running program that *drives* Web services, both to implement them and to access them. [WS Arch]

Taxonomy A controlled list of values by which to categorize or classify registered entities.

Uniform Resource Identifier (URI) A compact string of characters for identifying an abstract or physical resource. [RFC 3986]

Uniform Resource Locator (URL) A type of URI that identifies a resource via a representation of its primary access mechanism (e.g., its network "location"), rather than by some other attributes it may have. [RFC 3986]

Web service A self-describing, self-contained, modular unit of software application logic that provides defined business functionality. Web services are consumable software services that typically include some combination of business logic and data. [FAA Order 1375.1d]

3.3 Abbreviations

API	Application Programming Interface
ebRIM	ebXML Registry Information Model
ebXML	Electronic Business using Extensible Markup Language
OASIS	Organization for the Advancement of Structured Information Standards
SOAP	Simple Object Access Protocol
UDDI	Universal Description, Discovery and Integration
WSDL	Web Service Description Language
XML	eXtensible Mark-up Language

4 GENERAL REQUIREMENTS

Registration, in the context of this standard, is the process of creating, storing, and managing information about services in a registry. This information is usually referred as service metadata.

a. Every entity registered in an FAA-affiliated registry SHALL have a globally unique key that is used to identify the entity within the registry. Registry keys are discussed further in section 5.8 of this standard.

b. The service metadata required for registration SHALL include (1) information about the organization responsible for providing a service, (2) a description of the service's business function, (3) service technical details such as locations at which the service can be invoked and reference to a service programmatic interface, i.e. service binding, and (4) available technical information, such as specifications, protocols and transports deployed by a service.

c. Additional metadata required for registration SHALL include categories and classifications chosen from taxonomies maintained in the registry, as well as references to standards and specifications used in the service's implementation.

d. Additional metadata required for registration SHOULD describe relationships with organizations that publish or use the service.

The diagram in Figure 1 depicts the registration process.

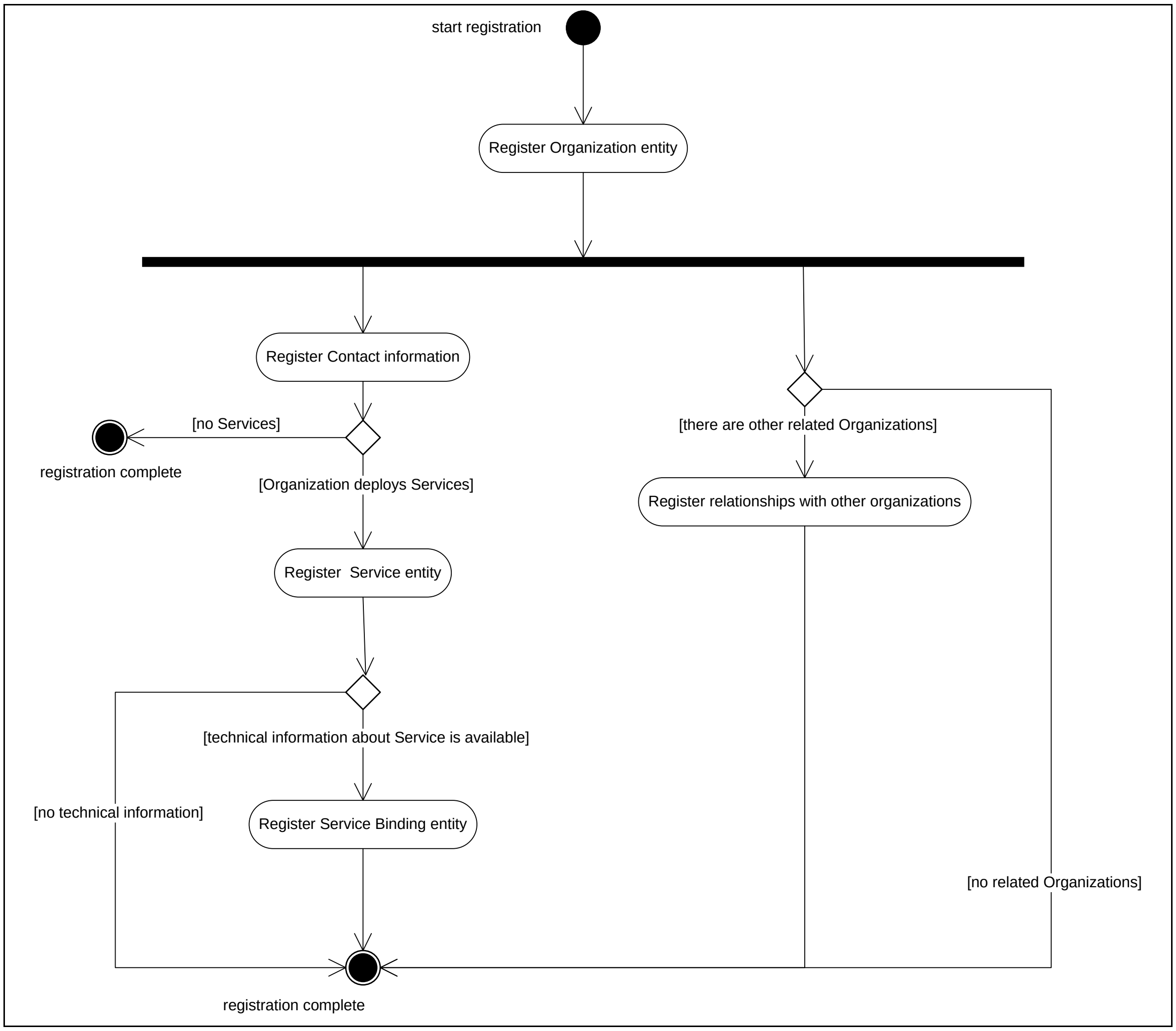
start registration
Register Organization entity
Register Contact information
[no Services]
registration complete
[Organization deploys Services]
Register Service entity
[technical information about Service is available]
[no technical information]
Register Service Binding entity
registration complete
[there are other related Organizations]
Register relationships with other organizations
[no related Organizations]

Figure 1. Registration process activities diagram

5 DETAILED REQUIREMENTS

This section describes detailed requirements for registering information about a service in an FAA-affiliated registry. Note: throughout this section, the word “registry” is used to mean “FAA-affiliated registry”.

5.1 Registering organization information

Requirements for registering information about an organization in a registry are presented in the following steps. Note: information about an organization is captured in the BusinessEntity element in UDDI-compliant registries. Section 6.1 shows how UDDI entities can be mapped to entities in the ebXML Registry Information Model (ebRIM).

a. The registry administrator SHALL establish a namespace that corresponds to the registered organization.

b. The namespace SHALL comply with requirements and registration procedures as specified in FAA-STD-063, XML Namespaces.

c. All organization entities in FAA-affiliated registries SHALL be globally uniquely identified in a manner prescribed in section 5.8, Registry keys.

d. For an organization entity being registered, a publisher SHALL provide at least one name in English.

e. A publisher MAY provide multiple names.

f. If multiple names in different languages are provided, the publisher SHALL identify the languages. Note: a common method for identifying language is to use the *xml:lang* attribute carrying a language value.

g. For an organization entity being registered, a publisher SHALL provide at least one description in English. Guidance is provided in section 6.2 to help ensure that the description is as informative and understandable as possible

h. A publisher MAY provide multiple descriptions.

i. If multiple descriptions in different languages are provided, the publisher SHALL identify the languages.

j. A publisher SHOULD provide a URL address for the web page associated with that organization.

k. A publisher SHALL categorize and classify the organization using available categorization schemes (taxonomies) as described in section 5.6, Using categorization schemes.

l. When registering an organization entity, a publisher SHALL provide contact information for at least one person or group within the organization, so that someone who finds the information can make human contact for any purpose.

Figure 2 is an example of registering an organization entity in a UDDI v.3 compliant service registry. Note: this example is non-normative.

```
<businessEntity xmlns="urn:uddi-org:api_v3" businessKey="uddi:us:gov:dot:faa:organization">
    <discoveryURLs>
        <discoveryURL>https://www.faa.gov/</discoveryURL>
    </discoveryURLs>
    <name>Federal Aviation Administration</name>
    <name>FAA</name>
    <description lang="en"> FAA as part of the Department of Transportation (DOT) is the
element of the U.S government with primary responsibility for the safety of civil aviation
   </description>
    <description lang="it">La Federal Aviation Administration (FAA) è l'agenzia del Dipartimento
dei Trasporti statunitense incaricata di regolare e sovrintendere ogni aspetto riguardante
l'aviazione civile.</description>
</businessEntity>
```

Figure 2. Registering organization entity in UDDI v.3 compliant registry

5.2 Registering contact information for an organization

To register information about an organization's human points of contact, the following steps should be taken. Requirements for registering information about an organization's human points of contact are presented in the following steps. Note: information about a human point of contact for an organization is captured in the "contact" element in UDDI-compliant registries. Section 6.1 shows how UDDI entities can be mapped to ebRIM entities.

a. For every registered point of contact, a publisher SHALL provide the name of the contact.

b. A point of contact SHOULD be an individual where possible.

c. For every registered point of contact, a publisher SHOULD provide the contact's job title or a description of the contact's responsibilities, in English. Guidance is provided in section 6.2 to help ensure that the description is as informative and understandable as possible.

d. A publisher MAY provide multiple titles or descriptions.

e. If multiple titles or descriptions in different languages are provided, the publisher SHALL identify the languages.

f. For every registered point of contact, a publisher SHALL provide at least one telephone number.

g. For every registered point of contact, a publisher SHALL provide at least one e-mail address.

h. For every registered point of contact, a publisher SHALL provide a postal address.

i. A postal address SHALL be composed in accordance with USPS Publication 28 – Postal Addressing Standards.

j. The country name used in the postal address SHALL be provided in accordance with ISO 3166 Part 2 – Country Subdivision Code.

Figure 3 is an example of registering contact information in a UDDI v.3 compliant service registry. Note: this example is non-normative.

```
<businessEntity xmlns="urn:uddi-org:api_v3" businessKey="uddi:us:gov:dot:faa:organization">
   …
   <contacts>
      <contact useType="Manager">
         <description lang="en">Administrator of Federal Aviation Administration</description>
         <personName>John Doe</personName>
         <phone>202-222-2222</phone>
         <email useType="primary">john.doe@faa.gov</email>
         <address useType="Headquarter" tModelKey="ubr-uddi-org:postaladdress">
            <addressLine keyName="BuildingName" keyValue="80">
              FAA National Headquarters
            </addressLine>
            <addressLine keyName="HouseNumber" keyValue="70"> 800</addressLine>
```

```
                <addressLine keyName="Street" keyValue="60"> Independence
Avenue</addressLine>
                <addressLine keyName="District" keyValue="50">SW</addressLine>
                <addressLine keyName="City" keyValue="40">Washington</addressLine>
                <addressLine keyName="Region" keyValue="30">DC</addressLine>
                <addressLine keyName="CityPostalCode" keyValue="120">20591</addressLine>
                <addressLine keyName="Country" keyValue="20"> US</addressLine>
            </address>
        </contact>
    </contacts>
</businessEntity>
```

Figure 3. Registering contact information in UDDI v.3 compliant registry

5.3 Registering service information

Requirements for registering information about a service are presented in the following steps. Note: information about a service is captured in the 'BusinessService' element in UDDI-compliant registries. Section 6.1 shows how UDDI entities can be mapped to ebRIM entities.

a. The registry administrator SHALL establish a namespace that corresponds to the registered service.

b. The namespace SHALL comply with requirements and registration procedures as specified in FAA-STD-063, XML Namespaces.

c. All service entities in FAA-affiliated registries SHALL be globally uniquely identified in a manner prescribed in section 5.8, Registry keys.

d. When registering a service, a publisher SHALL provide a least one *name* in English for the registered *service.*

e. The publisher SHOULD consider providing multiple names for the service, for example, in order to specify both the legal name and a known abbreviation for the service.

f. If multiple names in different languages are provided, the publisher SHALL identify the languages.

g. A publisher SHALL provide at least one description in English. Guidance is provided in section 6.2 to help ensure that the description is as informative and understandable as possible

h. A publisher MAY provide multiple descriptions.

i. If multiple descriptions in different languages are provided, the publisher SHALL identify the languages.

j. A publisher SHALL categorize and classify the service using available categorization schemes (taxonomies) as described in section 5.6, Using categorization schemes.

Figure 4 is an example of registering service information in a UDDI v.3 compliant service registry. Note: this example is non-normative.

```
<businessService xmlns="urn:uddi-org:api_v3"
serviceKey="uddi:us:gov:dot:faa:swim:ffm:rvr:service">
   <name>RVR</name>
   <description lang="en">
      Runway Visual Range (RVR) services provide information about the distance a pilot should
be able to see down a runway
   </description>
</businessService>
```

Figure 4. Registering service information in UDDI v.3 compliant registry

5.4 Registering service binding information

Requirements for registering information about a binding template are presented in the following steps. Note: binding information about the service is captured in the 'ServiceBinding' element in UDDI-compliant registries. Section 6.1 shows how UDDI entities can be mapped to ebXML.

a. All service binding entities in FAA-affiliated registries SHALL be globally uniquely identified in a manner prescribed in section 5.8, Registry keys.

b. For a service binding being registered, a publisher SHALL provide at least one description in English. Guidance is provided in section 6.2 to help ensure that the description is as informative and understandable as possible.

c. A publisher MAY provide multiple descriptions.

d. If multiple descriptions in different languages are provided, the publisher SHALL identify the languages.

e. When registering a service binding entity, the publisher SHALL include either the actual service endpoint, i.e. the network address at which the Web service can be invoked, or a reference to an artifact that contains information about the actual service endpoint.

f. A publisher SHALL categorize and classify the service binding using available categorization schemes (taxonomies) as described in section 5.6, Using categorization schemes.

Figure 5 is an example of registering service binding information in a UDDI v.3 compliant service registry. Note: this example is non-normative.

```
<bindingTemplate bindingKey="uddi:us:gov:dot:faa:naco:geospatial:wfs:binding"
                      serviceKey="uddi:us:gov:dot:faa:naco:geospatial:service">
    <description lang="en">
        This binding supports HTTP POST for OGC compliant Web Future Service
    </description>
    <accessPoint useType="endPoint">
         http://172.27.58.110:8080/naco-wfs/wfs.do?
    </accessPoint>
```

```
</bindingTemplate>
```

Figure 5. Registering service binding information in UDDI v.3 compliant registry

5.5 Registering service technical information

The amount of technical information about services is expected to vary significantly within different FAA-affiliated registries.

a. The registry administrator SHALL establish and enforce a policy which prescribes the kind and amount of technical details required for a registered service entity.

b. The registry administrator SHALL provide categorization schemes (taxonomies) which would allow a publisher to describe specifications, protocols and transports used for services design and implementations. See section 5.6, Using categorization schemes, for more details.

c. The publisher SHALL register information about transports, protocols and specifications using categorization schemes (taxonomies), provided by either a vendor or a registry administrator, as described in section 5.6, Using categorization schemes.

Figure 6 is an example of registering technical information in a UDDI v.3 compliant service registry. In this example a service interacts with a client via SOAP and the service is described using WSDL. Note: this example is non-normative.

```
<categoryBag>
   <keyedReference keyName="uddi-org:types:wsdl"
                   keyValue="wsdlSpec"
                   tModelKey="uddi:uddi.org:categorization:types"/>
   <keyedReference keyName="uddi-org:types:xml"
                   keyValue="xmlSpec"
                   tModelKey="uddi:uddi.org:categorization:types"/>
   </categoryBag>
```

Figure 6. Registering technical information in UDDI v.3 compliant registry

5.6 Using categorization schemes

The ability to perform a search based on categorization is one of the most important aspects of a service registry. Supplementing a registry with strongly typed metadata from various categorization schemes (taxonomies) is an essential task of registry administrators and publishers. Note: UDDI registries use “tModels” to represent compliance with taxonomies; ebXML registries use “ClassificationSchemes” to associate semantics with registry items. This section presents guidelines for using categorization and classification schemes in FAA-affiliated registries.

a. The administrator of each FAA-affiliated registry SHALL establish categorization schemes (taxonomies) by which publishers can categorize their services to promote and ensure discoverability.

b. The publisher SHALL categorize all registered entities in accordance with FAA-STD-066 Web Services Taxonomies.

Figure 7 is an example of using two common taxonomies, the North American Industry Classification System (NAICS) and the United Nations Standard Product and Services Code (UNSPSC) to classify the organization entry in a UDDI v.3 compliant service registry. Note: this example is non-normative.

```
<businessEntity businessKey=“uddi:us:gov:dot:faa:some:organization”>
 …
 <categoryBag>
   <keyedReference
     keyName="naics:2007:Federal Aviation Administration"
     keyValue="926120"
     tModelKey="uddi:uddi.org:ubr:categorization:naics:2007"/>
<keyedReference
     keyName="Air transportation support systems and equipment”
     keyValue="25191500"
     tModelKey="uddi:uddi.org:ubr:categorization:unspsc"/>
  <keyedReference
```

Figure 7. Using taxonomies in UDDI v.3 compliant registry

5.7 Registering relationships with other organizations

Many organizations both inside and outside the FAA relate to each other in many different ways. The relationships among those organizations may vary from administrative-hierarchical to partnership based on common business goals. Making those relationships visible inside a service registry enriches registry searching capabilities and creates a common view across the set of affiliated registries. Current service registry implementations provide a means to expose and explore those relationships. UDDI v.3 uses the “publisherAssertion” element to establish visible, reciprocal relationships between organizations registered in the registry and ebRIM defines a very similar “Association” structure for the same purpose. Both registry implementations come with taxonomies (“Relationships” and “Association Types” in UDDI v.3 and ebRIM respectively) that are used to describe the type of relationship between organizations.

a. A publisher of an organization within an FAA-affiliated registry SHOULD assert the organization’s relationship with other registered organizations.

b. A publisher who has asserted a relationship with another registered organization SHOULD request the publisher of the other organization to assert that the claimed relationship exists. Only mutually asserted relationships become visible. If a publisher controls both organizations, this step is not required.

c. The administrator of each FAA-affiliated registry SHOULD assert relationships with organizations registered in other FAA-affiliated registries as prescribed above.

Figure 8 is an example of registering a relationship in a UDDI v.3 compliant registry. Note: this example is non-normative.

```
<publisherAssertion>
 <fromKey>uddi:faa.gov:faa</fromKey>
 <toKey>uddi:faa.gov:ato</toKey>
 <keyedReference
   tModelKey="uddi:uddi.org:relationships"
   keyName=“Parent Organization“
   keyValue="parent-child" />
 </publisherAssertion>
```

Figure 8. Registering relationship information in UDDI v.3 compliant registry

5.8 Registry keys

Keys are used to identify entities within a registry. The following section describes requirements for key assignment for entities registered in FAA-affiliated registries.

5.8.1 Registry key assignment

The key assignment process differs for different implementations and even varies from version to version under the same registry specifications. For example, in UDDI versions 1.x and 2.x only registry created keys were available; UDDI version 3.x introduces the ability for publishers to create and assign entity keys in a safe, efficient manner that still supports the prevention of key collisions. The registry keys come in two forms: (1) registry-assigned keys, (2) publisher-assigned keys. Both have their advantages and disadvantages; for instance, keys created by a registry lack readability and have portability issues (though not in an ebXML-compliant registry, which deploys a slightly different approach), while publisher created keys can pose a number of key management challenges. The following requirements ensure that keys do not conflict with one another:

a. Each identifiable entity in the registry SHALL have a unique identifier (key).

b. The registry SHOULD allow an entity’s key to be created and assigned by the publisher.

c. If the publisher does not provide a key for an entity, the registry SHALL assign one.

5.8.2 Naming registry keys

Requirements for naming registry keys are as follows.

a. Each key name in an FAA-affiliated registry SHALL be comprised of UTF-8 characters and SHALL be compliant with URI syntax as specified in RFC 3986, Uniform Resource Identifier (URI).

b. All identifiable parts of a key name SHALL be formatted in lower camel case. (More information on key name syntax appears in the following sub-section).

c. Each key name SHALL be constructed using a namespace identifier to assure uniqueness. Uniqueness of a namespace identifier is achieved by checking it against the FAA Data Registry of previously-assigned namespaces as specified in FAA-STD-063, XML Namespaces.

5.8.3 Generic key syntax

The generic key syntax for FAA-affiliated registry keys consists of a hierarchical sequence of three components referred to as the *scheme*, *Namespace Identifier (NID)*, and *entity type or taxonomy*. The parts are defined in Figure 9.

a. All parts are REQUIRED and SHALL NOT be empty.

b. The syntax assumes that the namespace has been identified using a Uniform Resource Name (URN) form of URI as required by FAA-STD-063. If the namespace uses another form of URI, that URI SHOULD be converted to a URN as described in FAA-STD-063.

```
key = [<scheme>]":"[^"urn:"<NID>]":["<entity type>|<taxonomy>]

    <entity type> = “organization”|”service”|”binding”|<other>

    <taxonomy> = <taxonomy name>[“:categorization”|”:taxonomy”]
```

Where:

<scheme> The scheme is a required part of any URI "that refers to a specification for assigning identifiers within that scheme" [RFC 3986]. For registry keys the scheme SHALL comply with the specification implemented by a registry (e.g. "uddi" scheme for UDDI compliant registries).

<NID> The Namespace Identifier defined and registered in the FAA Data Registry in compliance with FAA-STD-063, with the scheme prefix “urn” removed. (The "^" symbol denotes that the following string is being excluded from the resulting string.)

<entity type> Type of the entity being registered (e.g. “organization”)

<taxonomy> A registered taxonomy

<other> An entity not covered by this Standard (e.g. “tModel” in a UDDI compliant registry)

<taxonomy name> Name of the registered taxonomy (e.g. “states”, “metrics”).

Figure 9. Formal syntax definition

Figure 10 depicts the process of constructing registry keys using the syntax described above.

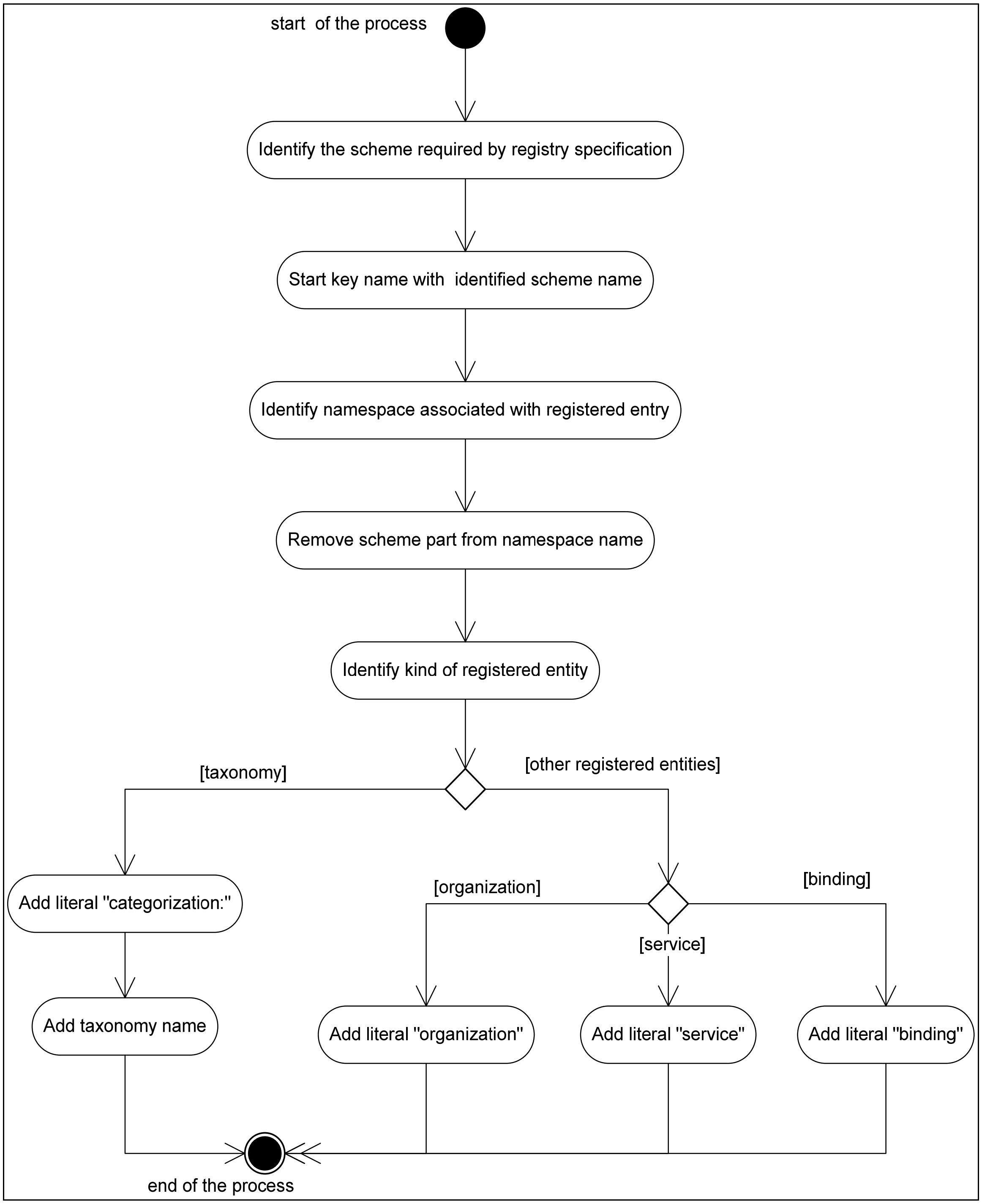
start of the process
Identify the scheme required by registry specification
Start key name with identified scheme name
Identify namespace associated with registered entry
Remove scheme part from namespace name
Identify kind of registered entity
[taxonomy]
[other registered entities]
Add literal "categorization:"
[organization]
[binding]
[service]
Add taxonomy name
Add literal "organization"
Add literal "service"
Add literal "binding"
end of the process

Figure 10. Constructing registry keys

Figure 11 is an example of exercising the process shown in Figure 2 above. For this example we assume that the namespace for the service has already been created in compliance with FAA-STD-063. We also assume the registry we use is UDDI v.3 compliant. Note: this example is non-normative.

Step 1. Identify the scheme (specification) under which the registry is implemented - `"uddi"`

Step 2. Look in the FAA Data Registry to find the name of the appropriate official namespace - `"urn:us:gov:dot:faa:swim:aim:sua"`

Step 3. Remove the scheme portion from the namespace's name - `"us:gov:dot:faa:swim:aim:sua"`

Step 4. Prefix the resulting string with the scheme identified in Step 1 - `"uddi:us:gov:dot:faa:swim:aim:sua"`

Step 5. Identify the kind of entity being registered (in this example, a service) - `"service"`

Step 6. Put it all together - `"uddi:us:gov:dot:faa:swim:aim:sua:service"`

Figure 11. Creating key for a service entry in UDDI v.3 compliant registry

Figure 12 contains some examples of key names as they might appear in a UDDI v.3 compliant registry. Note: this example is non-normative.

Registry key for organization entity

```
   uddi:us:gov:dot:faa:ats:organization
```

Registry key for service entity

```
   uddi:us:gov:dot:faa:swim:eram:flightPlan:service
```

Registry key for service binding

```
   uddi:us:gov:dot:faa:aim:xNOTAM:soap:binding
```

Registry key for taxonomy

```
   uddi:us:gov:dot:faa:swim:categorization:serviceCategory
```

Figure 12. Key names for entities in UDDI v.3 compliant registry

6 NOTES

(This section contains information of a general or explanatory nature that may be helpful, but is not mandatory.)

6.1 Entity mapping between UDDI and ebRIM

As of this standard's approval date, industry has defined two competing specifications for a general-purpose registry service: UDDI and ebXML. Both specifications are owned by the Organization for the Advancement of Structured Information Standards (OASIS). This standard makes no provision for registry implementation. Rather, it addresses the kinds of information needed to support registration in any registry implementation.

The following table provides a mapping between UDDI and ebRIM entities. This mapping is based on JAXR 1.0 Specification. Note: this mapping is not normative.

UDDI Entity	ebRIM Entity
businessEntity	Organization
businessService	Service
bindingTemplate	ServiceBinding
tModel (fingerprint)	Concept
tModel (namespace)	ClassificationScheme
publisherAssertion	Association
discoveryURL	ExternalLink
contact	User
identifierBag	Collection of ExternalIdentifier instances
categoryBag	Collection of Classification instances
address	PostalAddress
overviewDoc	ExternalLink
keyedReference (in categoryBag)	Classification
keyedReference (in identifierBag)	ExternalIdentifier
taxonomy	ClassificationScheme

6.2 Writing good definitions

The purpose of a definition is to define a concept with words or phrases that describe, explain, or make definite and clear its meaning. Precise and unambiguous definitions are one of the most critical aspects of ensuring data shareability. When two or more parties exchange data, it is essential that all be in explicit agreement on the meaning of that data.

ISO/IEC 11179 Part 4 provides a guide for writing good data definitions. The requirements and recommendations described in this section are adapted from ISO/IEC 11179 Part 4, Sections 4 and 5. (Note the difference between requirements and recommendations: compliance with the requirements can be objectively tested, whereas compliance with the recommendations can only be evaluated subjectively.) Although ISO/IEC 11179-4 requirements and recommendations pertain to data elements and other administered items like object classes and properties, they can also be applied when writing definitions or descriptions for any registered artifacts. The requirements and recommendations are as follows. Note: these are not normative.

Requirements – A data definition shall:

- Be stated in the singular
- State what the concept is, not only what it is not
- Be stated as a descriptive phrase or sentence(s)
- Contain only commonly understood abbreviations
- Be expressed without embedding definitions of other data or underlying concepts

Recommendations – A data definition should:

- State the essential meaning of the concept
- Be precise and unambiguous
- Be concise
- Be able to stand alone
- Be expressed without embedding rationale, functional usage, or procedural information
- Avoid circular reasoning
- Use the same terminology and consistent logical structure for related definitions
- Be appropriate for the type of metadata item being defined

APPENDIX

Key words for indicating requirement levels

The following information is obtained from the Internet Engineering Task Force (IETF) RFC 2119: Key words for use in RFCs to Indicate Requirement Levels, March 1997.

MUST This word, or the terms "REQUIRED" or "SHALL", means that the definition is an absolute requirement of the specification.

MUST NOT This phrase, or the phrase "SHALL NOT", means that the definition is an absolute prohibition of the specification.

SHALL See MUST.

SHALL NOT See MUST NOT.

SHOULD This word, or the adjective "RECOMMENDED", means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

SHOULD NOT This phrase, or the phrase "NOT RECOMMENDED", means that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.

MAY This word, or the adjective "OPTIONAL", means that an item is truly optional.