Intelligence Community and Department of Defense

Content Discovery & Retrieval Integrated Project Team

IC-DoD SOAP Interface Encoding Specification for CDR Search

V2.0-20111214

14 December 2011
**Revision History**

<table>
<thead>
<tr>
<th>Revision</th>
<th>Revised By</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td></td>
<td>20 November 2009</td>
<td>Initial draft for subgroup review.</td>
</tr>
<tr>
<td>0.2</td>
<td></td>
<td>18 December 2009</td>
<td>All sections updated.</td>
</tr>
<tr>
<td>0.4.1</td>
<td>CDR IPT</td>
<td>01 January 2010</td>
<td>All sections updated. Changed the revision number to reflect the associated RA and Specification Framework documents, respectively.</td>
</tr>
<tr>
<td>0.4.4</td>
<td></td>
<td>15 January 2010</td>
<td>All sections updated.</td>
</tr>
<tr>
<td>0.9.0</td>
<td>CDR IPT</td>
<td>22 January 2010</td>
<td>First draft release for CDR SOAP Working Group review.</td>
</tr>
<tr>
<td>0.9.1</td>
<td>CDR IPT</td>
<td>28 January 2010</td>
<td>All sections updated.</td>
</tr>
<tr>
<td>0.9.2</td>
<td>CDR IPT</td>
<td>04 February 2010</td>
<td>All sections updated.</td>
</tr>
<tr>
<td>0.9.3</td>
<td>CDR IPT</td>
<td>16 February 2010</td>
<td>Minor updates.</td>
</tr>
<tr>
<td>0.9.4</td>
<td>CDR IPT</td>
<td>19 February 2010</td>
<td>Minor updates.</td>
</tr>
<tr>
<td>0.9.5</td>
<td>CDR IPT</td>
<td>26 February 2010</td>
<td>Minor updates.</td>
</tr>
<tr>
<td>1.0-Milestone 1</td>
<td>CDR IPT</td>
<td>09 March 2010</td>
<td>Minor updates.</td>
</tr>
<tr>
<td>1.0-Milestone 1</td>
<td>Farley</td>
<td>23 April 2010</td>
<td>Tech Edit</td>
</tr>
<tr>
<td>2.0 Draft 1</td>
<td>McKnight</td>
<td>15 July 2011</td>
<td>Harmonization Update- incomplete</td>
</tr>
<tr>
<td>2.0 Draft 2</td>
<td>McKnight</td>
<td>22 July 2011</td>
<td>Harmonization Update- complete</td>
</tr>
<tr>
<td>2.0 Draft 3</td>
<td>McKnight</td>
<td>29 July 2011</td>
<td>Added references and completeness</td>
</tr>
<tr>
<td>2.0 Draft 4</td>
<td>McKnight</td>
<td>9 August 2011</td>
<td>Modified to accommodate feedback on draft 3 and consistency with schema and WSDL</td>
</tr>
<tr>
<td>2.0 Draft 5</td>
<td>McKnight</td>
<td>08 September 2011</td>
<td>Modified to accommodate changes from adjudication review</td>
</tr>
<tr>
<td>2.0 Draft 6</td>
<td>McKnight</td>
<td>20 September 2011</td>
<td>Modified to accommodate changes from adjudication review</td>
</tr>
<tr>
<td>2.0 Draft 7</td>
<td>McKnight</td>
<td>26 September 2011</td>
<td>Modified to accommodate changes from adjudication review</td>
</tr>
<tr>
<td>2.0 Draft 8</td>
<td>McKnight</td>
<td>06 October 2011</td>
<td>Modified for adjusted view of Query Language and QueryId</td>
</tr>
<tr>
<td>2.0 Draft 9</td>
<td>McKnight</td>
<td>12 October 2011</td>
<td>Modified for ODNI style guide concerns</td>
</tr>
<tr>
<td>2.0 Draft 10</td>
<td>McKnight</td>
<td>02 November 2011</td>
<td>Modified for CDR IPT comments</td>
</tr>
<tr>
<td>2.0 Draft 11</td>
<td>McKnight</td>
<td>28 November 2011</td>
<td>Modified for SCA comments</td>
</tr>
<tr>
<td>2.0 Draft 12</td>
<td>Wigglesworth</td>
<td>14 December 2011</td>
<td>Modified for ASEC comments</td>
</tr>
</tbody>
</table>
Table of Contents

1 Introduction .......................................................................................................................... 3
  1.1 Service Overview ............................................................................................................ 5
  1.2 Artifact Overview .......................................................................................................... 5
  1.3 Notational Convention ................................................................................................... 7
  1.4 Conformance .................................................................................................................. 7
  1.5 Namespaces .................................................................................................................... 7
  1.6 Query Language URIs .................................................................................................... 8
  1.7 Security .......................................................................................................................... 8

2 Search Service Behavior ...................................................................................................... 8
  2.1 Search ............................................................................................................................ 8
  2.2 Paging ............................................................................................................................ 9
  2.3 Results Presentation ....................................................................................................... 9
  2.4 Relevance ...................................................................................................................... 9

3 Search Service Interface ...................................................................................................... 9
  3.1 Search Function .............................................................................................................. 9
    3.1.1 Preconditions ........................................................................................................... 9
    3.1.2 Input ....................................................................................................................... 10
    3.1.3 Output .................................................................................................................... 12
    3.1.4 Post-Conditions ..................................................................................................... 14
    3.1.5 Fault Conditions .................................................................................................... 14

4 References .......................................................................................................................... 17

Appendix A. Mapping to Specification Framework .................................................................. 17
  A.1. Search Request ............................................................................................................. 17
  A.2. Search Response ........................................................................................................... 18

Appendix B. SOAP-REST Interoperability ......................................................................... 18
  B.1. SOAP-REST Parameter Mapping ................................................................................ 18
    B.1.1 Input ....................................................................................................................... 18
    B.1.2 Output .................................................................................................................... 19

Appendix C. Changes from Prior Version ............................................................................ 19
  C.1. Search Request ............................................................................................................ 19
    C.1.1 Change queryTypeURI to queryLanguage ............................................................. 19
    C.1.2 Change resultsPerPage to count .......................................................................... 19
    C.1.3 Use of Query Identifier ......................................................................................... 20
  C.2. Search Response ........................................................................................................... 20
List of Figures

Figure 1. CDR Architectural Model ........................................................................................................... 6

List of Tables

Table 1. List of Namespaces .......................................................................................................................... 8
Table 2. Query Language URIs ....................................................................................................................... 8
Table 3. Header Elements for Search Requests .............................................................................................. 10
Table 4. Attributes of /cdr:SearchRequest Element ..................................................................................... 10
Table 5. Child Elements of /cdr:SearchRequest Element ............................................................................. 11
Table 6. Required Header Elements for Search Responses ........................................................................... 12
Table 7. Atom Result Set Attributes ............................................................................................................. 13
Table 8. Atom Result Contents ....................................................................................................................... 13
Table 9. List of Faults ..................................................................................................................................... 16
Table 10. Search Request Mapping to Specification Framework ................................................................. 17
Table 11. Search Response Mapping to Specification Framework ............................................................... 18
Table 12. REST to SOAP Mapping ............................................................................................................... 19
1 Introduction

1.1 Service Overview

The Search Component, as defined by the Intelligence Community/Department of Defense (IC/DoD) Content Discovery and Retrieval (CDR) Specification Framework [CDR-SF] serves as the primary content discovery mechanism to expose content collections for discovery and accessibility. This component provides a common service interface and behavioral model for IC and DoD content collections, enabling content consumers to discover relevant content resources from disparate collections across the IC/DoD enterprise.

This specification defines requirements and provides guidance for the realization of the CDR Search Component as a web service using SOAP\(^1\), hereafter termed a Search Service in this document. The content of this specification describes the Search Service’s behavior, interface and other aspects in detail, providing enough information for Search Service providers and consumers to create and use CDR-conformant Search Services.

The Search Service exposes a single Search operation that is responsible for three activities that underpin Content Discovery capabilities: Search, Results Presentation, and Results Paging. As discussed in the CDR Specification Framework, a Search Service’s results are generally resource metadata rather than actual content resources. In the context of Search, resource metadata generally refers to a subset of a resource’s available metadata, not the entire underlying record. Some of the information contained within each Search result may provide the information necessary for a consumer to estimate relevance, retrieve or otherwise use the referenced resource.

1.2 Artifact Overview

This specification is a part of the set of specifications that define the concrete, implementation-specific guidance for the services defined under the auspices of the CDR IPT. The CDR Reference Architecture [CDR-RA] prescribes an abstract-to-concrete model for the development of architecture elements and guidance for content discovery and retrieval. Each layer or tier of the model is intended to provide key aspects of the overall guidance to achieve the goals and objectives for joint DoD/IC content discovery and retrieval. The following graphic, discussed in detail within the CDR Reference Architecture, illustrates this model.

\(^1\) SOAP is a protocol used by web services in the exchange of structured information.
As illustrated in Figure 1, the CDR-SF derives from the CDR-RA and describes behavior in terms of the capabilities, components, and usage patterns defined in the RA. Multiple CDR Service Specifications are derived from the CDR-SF, with separate specifications associated with the components of the architecture (e.g., Search) and, for each service, separate specifications to address Representational State Transfer (REST) and SOAP implementations.

This specification provides guidance for implementing the CDR Search Service as a SOAP Web Service. It is intended to parallel the corresponding REST specification, the IC/DoD Content Discovery & Retrieval Search Service Specification for OpenSearch Implementations [CDR-OS], as closely as possible, to minimize the difficulties in interoperating. Additional CDR Guides, Profile Specifications, or Reference Implementations may provide additional guidance on implementing this specification in a particular context.
1.3 Notational Convention

The key words "MUST," "MUST NOT," "REQUIRED," "SHALL," "SHALL NOT,"
"SHOULD," "SHOULD NOT," "RECOMMENDED," "MAY," and "OPTIONAL" in this
specification are to be interpreted as described in the IETF RFC 2119. When these words are not
capitalized, they are meant in their natural-language sense.

When describing concrete eXtensible Markup Language (XML) schemas and example XML
documents, this specification uses XPath as the notational convention. Each member of an XML
schema is described using an XPath notation (e.g.,
/x:RootElement/x:ChildElement/@Attribute). The use of {any} indicates the
presence of an attribute wildcard (<?xs:anyAttribute>).

Items contained in curly braces ({item}) are meant to indicate template or notional values to
replaced by actual values (without the use of curly braces) when in actual use.

Examples in this text are distinguished by a black border. These are meant to be illustrative and
only one way that the described syntax can be used.

```xml
<atom:entry>
  <atom:title>This is an example.</atom:title>
</atom:entry>
```

1.4 Conformance

This specification defines an interface to a Search Service to which an implementation and a
subsequent deployment MUST conform. A deployment is an instance of an implementation. For
an implementation to conform to this Search specification, it MUST adhere to all mandatory
aspects of the specification.

1.5 Namespaces

Namespaces referenced in this document and the prefixes used to represent them are listed in the
following table. The namespace prefix of any XML Qualified Name (QName) used in any
example in this document should be interpreted using the information below.
Table 1. List of Namespaces

<table>
<thead>
<tr>
<th>Prefix</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>atom</td>
<td><a href="http://www.w3.org/2005/Atom">http://www.w3.org/2005/Atom</a></td>
<td>The atom syndication format</td>
</tr>
<tr>
<td>cdrs</td>
<td>urn:cdr:search:2.0</td>
<td>The CDR IPT Search binding for SOAP implementations</td>
</tr>
<tr>
<td>opensearch</td>
<td><a href="http://a9.com/-/spec/opensearch/1.1/">http://a9.com/-/spec/opensearch/1.1/</a></td>
<td>The OpenSearch specification for search web services</td>
</tr>
<tr>
<td>relevance</td>
<td><a href="http://a9.com/-/opensearch/extensions/relevance/1.0/">http://a9.com/-/opensearch/extensions/relevance/1.0/</a></td>
<td>An OpenSearch extension for relevance.</td>
</tr>
<tr>
<td>soap</td>
<td><a href="http://www.w3.org/2003/05/soap-envelope">http://www.w3.org/2003/05/soap-envelope</a></td>
<td>SOAP 1.2 Envelope</td>
</tr>
<tr>
<td>wsa</td>
<td><a href="http://www.w3.org/2005/08/addressing">http://www.w3.org/2005/08/addressing</a></td>
<td>WS-Addressing</td>
</tr>
<tr>
<td>xs</td>
<td><a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a></td>
<td>XML Schema</td>
</tr>
</tbody>
</table>

1.6 Query Language URIs

The following list provides the URIs for query languages that are currently defined as part of the CDR specification set. Table 2 references the query languages by name and shows the definition uniquely identified by the URIs. These are acceptable values for the queryLanguage attribute defined in Table 4. Additional acceptable query language values may be defined in the future and MUST also be identified by Name and URI. Specification Profiles that describe how to utilize this specification in the context of the use of one of these query languages, or other languages defined in the future, may be added to the CDR collection of artifacts.

Table 2. Query Language URIs

<table>
<thead>
<tr>
<th>Name</th>
<th>URI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Keyword</td>
<td>urn:cdr:1.0:soap:query:keyword</td>
<td>A generic definition for keyword queries</td>
</tr>
<tr>
<td>XQuery</td>
<td>urn:cdr:1.0:soap:query:xquery-1.0</td>
<td>A limited form of XQuery</td>
</tr>
<tr>
<td>OGC Filter</td>
<td>urn:cdr:1.0:soap:query:opengis-filter-1.1</td>
<td>The Open Geospatial Consortium Search filter</td>
</tr>
</tbody>
</table>

1.7 Security

This specification does not directly address security concerns. It will be necessary for any implementation of this specification to address security concerns relevant to the systems with which they interact and the governance bodies. Several aspects of search, to include returning only the results for which the requesting entity is authorized, should be addressed in the detailed security plan of an implementation, but are out of scope for this document. The Web Services Security Working Group is addressing these concerns.

2 Search Service Behavior

2.1 Search

The Search behavior accepts a Search Request\(^2\) from the service consumer, identifies the query and processes it against the collection of information available to the Search Service to build a

\(^2\) Precise definitions of “Search Request”, “Search Results”, “Query” and other search related terminology are included in the CDR-SF.
set of items, called Search Results, which are the Search Services response to the service consumer’s query.

2.2 Paging

Paginating the return of search results can be useful when the number of results is very large or indeterminate. The Search Service may limit the number of Search Results that are returned for a single request for performance and convenience reasons. The Search Service may accept parameters that allow service consumers to request a particular subset of the Search Results. This capability will allow for the management of search requests that generate a very large number of results from overloading the server, network, or client.

It is important to note that the paging mechanism supported by Search Services does not guarantee continuity of search results while switching pages. Consequently, it may not be possible to guarantee consumers will be able to reconstruct the contents of the entire result set at a particular time. Data assets may be added, updated, or removed in the period of time between page requests. Therefore, service consumers SHOULD NOT assume continuity among paged result sets, unless such continuity is explicitly supported.

2.3 Results Presentation

The format, content, and ordering of the Search Results is referred to as Results Presentation. Support for sorting functionality is OPTIONAL; however, Search Services SHOULD by default provide results sorted by relevance, if possible. Implementations MAY add sorting parameters or allow sort order to be expressed in the query expression itself. Support for alternate response data formats MAY be provided by Search Services. If no alternate representations are requested by the consumer, the Service MUST respond with the results format associated with the particular service binding that was invoked.

2.4 Relevance

Result relevance is generally a measure of how well a specific result matched the original query. Providing a result relevance measure allows better matched results to be prioritized relative to other results. A Search Service implementation MAY provide relevance scores for individual Search Results with respect to the particular search with which it is identified.

3 Search Service Interface

3.1 Search Function

3.1.1 Preconditions

The following preconditions MUST be satisfied if the search function is to correctly process input and generate results and post-conditions as described.

1. The requester is authenticated according to applicable policy requirements for auditing search activity and authorizing access.

---

3 The availability and access to the information to be presented MAY also be controlled on the basis of the type of data being represented in the items and the authorization of the requestor to that data.
2. The authenticated requester is authorized to access the Search Service.

3.1.2 Input
The input to the CDR Search Service MUST be a valid SOAP\(^4\) message that meets criteria identified in this section. The input should be directed to the SOAP Endpoint address identified by the implementer.

3.1.2.1 Header
The header of the SOAP message must contain the action element, as defined in WS-Addressing [WS-A]. The purpose of this element is to convey to the service which behavior to invoke. Additional elements, such as other WS-Addressing elements, MAY be added to the SOAP header.

Table 3. Header Elements for Search Requests

<table>
<thead>
<tr>
<th>Element Name Description</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>/wsa:Action</td>
<td>MUST be supported by Service. MUST be provided by consumer with a value of urn:cdr:SearchService:2.0:request</td>
</tr>
</tbody>
</table>

3.1.2.2 Body
The body of the SOAP message must contain a single /cdr:SearchRequest element, as defined in this document. The /cdr:SearchRequest element contains the /cdr:Expression element, which provides the query expression. Table 4 shows the attributes of the /cdr:SearchRequest:\(^5\)

Table 4. Attributes of /cdr:SearchRequest Element

<table>
<thead>
<tr>
<th>Attribute Name Description</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>/cdr:SearchRequest/@startIndex</td>
<td>MUST be supported by Service. MAY be provided by consumer.</td>
</tr>
<tr>
<td>/cdr:SearchRequest/@startPage</td>
<td>MAY be supported by Service. MAY be provided by consumer.</td>
</tr>
<tr>
<td>/cdr:SearchRequest/@count</td>
<td>MUST be supported by Service. MAY be provided by consumer.</td>
</tr>
</tbody>
</table>

\(^4\) Consult the relevant standards registry (such as the ICSR or DISR) to determine the appropriate current version of the SOAP standard to use. Examples in this document use SOAP 1.2.

\(^5\) Please see Appendix C “Changes From Prior Version” for a disposition of the request parameters from the prior version of this specification.
result set. Must be expressed as an integer greater than zero if used. The default value if not specified is 10. Search clients should anticipate that the value of the "count" attribute may not be honored by the search engine, and should rely exclusively on the contents of the "opensearch:itemsPerPage" response element in calculating actual page size.

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>/cdr:SearchRequest/@queryLanguage</td>
<td>The URI for a query language used by the query. This value is supplied to support the case of a Search Service that supports multiple query languages. A Specification Profile should exist for the Query Language referenced for this URI. If not provided, the service SHOULD interpret the query as being expressed in a default query language.</td>
<td>MUST be supported by Service. MAY be provided by Consumer.</td>
</tr>
<tr>
<td>/cdr:SearchRequest/@responseFormat</td>
<td>The response format desired by the client. If provided, the value MUST be a URI that corresponds to a definition of a search response format defined as part of the CDR specifications.</td>
<td>MAY be supported by Service. MAY be provided by Consumer.</td>
</tr>
<tr>
<td>/cdr:SearchRequest/@timeout</td>
<td>The desired timeout period in milliseconds. If the timeout attribute is supported by the Service and provided by the Consumer, when the response time for the service exceeds the timeout value, the Service MUST respond with either partial results or a Timeout Fault, as described in the Fault Conditions, section 3.1.4. If partial results are returned, this status must be indicated in the metadata of the results.</td>
<td>MAY be supported by Service. MAY be provided by Consumer.</td>
</tr>
</tbody>
</table>

Additional extension attributes MAY be supported by the Search Service and/or provided by the consumer to convey additional Query or Search properties. However, any additional attributes supported by the Search Service MUST be OPTIONAL for the consumer to provide. If the consumer provides extension attributes that the Search Service does not support, these MUST be ignored.

The /cdr:SearchRequest element has one child element, /Expression, which contains the Query Expression.

**Table 5. Child Elements of /cdr:SearchRequest Element**

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>/cdr:SearchRequest/Expression</td>
<td>The element provides the query expression for which the Search Service must return matching Search Results. The format must be consistent with the query language referenced by the @queryLanguage attribute.</td>
<td>MUST be supported by Service. MUST be provided by Consumer.</td>
</tr>
</tbody>
</table>

11
The query language used in the /cdr:SearchRequest/Expression element MUST be in a query language supported by the Search Service. The query language may allow the expression to contain information relating to result set sorting or ordering. Additional extension elements MAY be supported by the Search Service and/or provided by the consumer to convey additional Query or Search properties. However, any additional elements supported by the Search Service MUST be OPTIONAL for the consumer to provide. If the consumer provides additional elements that the Search Service does not support, these MUST be ignored.

### 3.1.2.3 Example

An example of a SOAP message constituting a Search Service request follows:

```xml
<soap:Envelope>
  <soap:Header>
    <wsa:Action>urn:cdr:SearchService:2.0:request</wsa:Action>
  </soap:Header>
  <soap:Body>
    <cdr:SearchRequest startIndex="1" count="10"
      queryLanguage="urn:cdr:queryLanguage:keyword"
      responseFormat="http://www.w3.org/2005/Atom">
      <cdr:Expression>watson ibm</cdr:Expression>
    </cdr:SearchRequest>
  </soap:Body>
</soap:Envelope>
```

This example shows a search for the keywords “watson” and “ibm”, using the Keyword query language, with an Atom response format requested.

### 3.1.3 Output

The output of the CDR Search Service is a page of a Result Set, comprised of Results that describe resources that matched the query provided in the Search Request. The format of the response is determined by the /cdr:SearchRequest/@responseFormat attribute. This section describes a SOAP message format that can be used to encapsulate a pluggable, separately specified response format. It uses the CDR-Atom response format as the basis for the examples in this section. For requests that result in an error, a SOAP fault message will be output.

#### 3.1.3.1 Header

The header of the SOAP message must contain the action element, as defined in WS-Addressing. The purpose of this element is to convey to the receiver which behavior was invoked.

<table>
<thead>
<tr>
<th>Element Name</th>
<th>Description</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>/wsa:Action</td>
<td>This element (content is of type xs:anyURI) conveys the value of the [action] property and indicates to a web service which operation should be invoked.</td>
<td>MUST be provided by the Service with a value of urn:cdr:SearchService:2.0:response</td>
</tr>
</tbody>
</table>

Additional elements, such as other WS-Addressing elements, MAY be added to the SOAP header.
3.1.3.2 Body

The body of the SOAP message MUST consist of a single element, representing a page of the result set. The name and structure of the element is defined by the response format. Table 7 shows this in the context of the Atom 1.0 Result Set Specification, where the single element of the body of the SOAP message is /atom:feed. The attribute definitions this specification adds to that element correspond with those defined in the OpenSearch Specification. These are specified with the required elements and attributes of the Atom Syndication Format. If a different response format is defined and specified by the request, the appropriate information as defined by that response format must be returned.

Table 7. Atom Result Set Attributes

<table>
<thead>
<tr>
<th>Attribute Name Description</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>/atom:feed/openSearch:startIndex</td>
<td>MAY be provided by Service.</td>
</tr>
<tr>
<td>The index into the ordered Search Results of the first Search Result desired by the Consumer. Must be expressed as an integer greater than or equal to zero.</td>
<td></td>
</tr>
<tr>
<td>/atom:feed/openSearch:itemsPerPage</td>
<td>MAY be provided by Service.</td>
</tr>
<tr>
<td>The number of items returned in a single response.</td>
<td></td>
</tr>
<tr>
<td>/atom:feed/openSearch:totalResults</td>
<td>MAY be provided by Service.</td>
</tr>
<tr>
<td>The total number of results that matched the query.</td>
<td></td>
</tr>
</tbody>
</table>

The element containing the result set should contain a series of result elements, each representing one resource that matched the query provided in the Search Request. In the case of the Atom Syndication Format, the element corresponding to a Result is /atom:entry. This Result contains the metadata necessary to identify and retrieve the referenced resource. The contents below are shown relative to an /atom:entry.

Table 8. Atom Result Contents

<table>
<thead>
<tr>
<th>Attribute Name Description</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>/atom:entry/relevance:score</td>
<td>MAY be provided by Service.</td>
</tr>
<tr>
<td>The relevance score for an entry.</td>
<td></td>
</tr>
<tr>
<td>/atom:entry/link</td>
<td>MUST be provided by Service.</td>
</tr>
<tr>
<td>Binding information to a CDR Retrieve Service to retrieve the referenced resource. This may be a wsa:EndpointReference.</td>
<td></td>
</tr>
</tbody>
</table>
3.1.3.3 Output Example

The following represents a Sample Output embedding the Atom Syndication Format style result set into the body of a SOAP message.

```xml
<soap:envelope>
  <soap:header>
    <wsa:action>urn:cdr:search:2.0:response</wsa:action>
  </soap:header>
  <soap:body>
    <atom:feed>
      <atom:id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af6</atom:id>
      <atom:title>Query Results for "watson ibm"</atom:title>
      <atom:author><atom:name>Enterprise Catalog</atom:name></atom:author>
      <opensearch:totalResults>492420</opensearch:totalResults>
      <opensearch:startIndex>1</opensearch:startIndex>
      <opensearch:itemsPerPage>10</opensearch:itemsPerPage>
      <atom:entry>
        <atom:id>urn:uuid:60a76c80-d399-11d9-b93C-0003939e0af7</atom:id>
        <atom:title>IBM - Watson</atom:title>
        <atom:updated>2011-02-21T00:00:00Z</atom:updated>
        <relevance:score>0.97</relevance:score>
      </atom:entry>
    </atom:feed>
  </soap:body>
</soap:envelope>
```

3.1.4 Post-Conditions

The following conditions MUST be met upon successful completion of a search.

1. The results returned to the requester are relevant to the input query.
2. The results are in the correct format.
3. The authenticated requester has been authorized to receive each result in the response.
4. The Search function has been audited according to applicable policy.\(^6\)

3.1.5 Fault Conditions

An implementation of the Search Service MAY provide any of the following faults as a SOAP Fault to the consumer. The SOAP message MUST contain a single SOAP Fault element as the only child of the SOAP Body element.

The SOAP Fault element must contain a mandatory /soap:Fault/Code element and a mandatory /soap:Fault/Reason element. The /soap:Fault/Code/Value element should be used to convey the general type of error condition from the enumeration /soap:faultCodeEnum as described in the SOAP 1.2 specification sections 5.4.1 and 5.4.6. For each fault a unique /soap:Fault/Code/Subcode/Value is provided to support automated processing of CDR specific errors. A /soap:Fault/Reason/Text element

\(^6\) The Search function may be audited according to applicable policy regardless to the success or failure of the function.
should be used to provide a human-readable explanation of the fault as described the SOAP 1.2 specifications section 5.4.2.

The following table outlines the service specific fault conditions that MAY be generated by CDR Search Service implementations.
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Security soap:Sender cdr:search:soap:fault:security</td>
<td>The Consumer is either not authenticated or not authorized to perform the search query.</td>
</tr>
<tr>
<td>Query Property Not Supported soap:Sender cdr:search:soap:fault:property</td>
<td>The Search Component does not support the Query Property, such as queryLanguage, that was specified.</td>
</tr>
<tr>
<td>Invalid Query Syntax soap:Sender cdr:search:soap:fault:syntax</td>
<td>The Search Request syntax is not valid in accordance with the specified Query Language.</td>
</tr>
<tr>
<td>Query Term Not Supported soap:Sender cdr:search:soap:fault:term</td>
<td>The Search Component cannot understand/support one or more elements of the Search Request.</td>
</tr>
<tr>
<td>Query Timeout soap:Receiver cdr:search:soap:fault:timeout</td>
<td>The query cannot be executed in the amount of time specified by the Timeout input parameter.</td>
</tr>
<tr>
<td>Query Metadata Fault soap:Sender cdr:search:soap:fault:metadata</td>
<td>The query metadata is not understood or contains an error. In the case of Query Metadata not being understood, a Search Component MAY choose to continue the execution of the query. In this case, some indication SHOULD be provided in the output’s Result Metadata Properties.</td>
</tr>
<tr>
<td>Invalid Paging Value Fault soap:Sender cdr:search:soap:fault:pagingValue</td>
<td>The Start Index and/or Count values are not valid values (e.g., non-integer, negative).</td>
</tr>
<tr>
<td>Out of Range Fault soap:Sender cdr:search:soap:fault:outOfRange</td>
<td>The Start Index and/or Count values are not in a valid range.</td>
</tr>
<tr>
<td>Result Format Not Supported soap:Sender cdr:search:soap:fault:resultFormat</td>
<td>The Search Component does not support the result format specified by the response Format input parameter.</td>
</tr>
<tr>
<td>Result Sorting Not Supported soap:Sender cdr:search:soap:fault:sorting</td>
<td>The Search Component determined does not support the result sorting mechanism.</td>
</tr>
</tbody>
</table>
3.1.5.1 Fault Message Example

The following shows a fault message of type “Invalid Query Syntax”.

```xml
<soap:Envelope>
  <soap:Body>
    <soap:Fault>
      <soap:Code>
        <soap:Value>soap:Sender</soap:Value>
      </soap:Code>
      <soap:Subcode>
        <soap:Value>cdr:search:soap:fault:syntax</soap:Value>
      </soap:Subcode>
      <soap:Reason>
        <soap:Text xml:lang="en">Invalid Query Syntax</soap:Text>
      </soap:Reason>
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

4 References

4. CDR-S 1.0 "IC/DoD CDR Search Specification for SOAP Implementations 1.0". 2010.

Appendix A. Mapping to Specification Framework

This section explicitly ties the items in this specification to the requirements of the CDR-SF. The CDR-SF identifies the requirements for creating specifications, while the implementation details are outlined here.

A.1. Search Request

The CDR Search Request definition in this specification defines a variety of properties that fall under the definition of Query Properties from the CDR-SF.

<table>
<thead>
<tr>
<th>Specification Framework Variable</th>
<th>SOAP Search Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query</td>
<td>/cdr:SearchRequest/Expression</td>
</tr>
<tr>
<td>Query Properties</td>
<td>/cdr:SearchRequest/@queryLanguage</td>
</tr>
<tr>
<td>Search Properties</td>
<td>/cdr:SearchRequest/@startIndex</td>
</tr>
<tr>
<td></td>
<td>/cdr:SearchRequest/@count</td>
</tr>
<tr>
<td></td>
<td>/cdr:SearchRequest/@responseFormat</td>
</tr>
</tbody>
</table>
A.2. Search Response
Details of the search response are contained in the Atom 1.0 Result Set Specification [CDR-Atom] and are not duplicated here. Additional search response formats may be defined and added to the defined set in the future.

Table 11. Search Response Mapping to Specification Framework

<table>
<thead>
<tr>
<th>Specification Framework Variable</th>
<th>SOAP Search Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Result Set</td>
<td>/cdr:SearchResponse/atom:feed</td>
</tr>
</tbody>
</table>

Appendix B. SOAP-REST Interoperability
To date, there are two specifications for the Search Service defined by the CDR Specification Framework version 1.0. The OpenSearch Specification version 1.2 details the RESTful binding. This specification details the SOAP binding. This appendix will address the interoperability of services created with the REST binding with services created with the SOAP binding.

The full scope of interoperability between SOAP and REST web services is outside the scope of this specification. However, by providing a mapping between the URL parameters and returned XML data utilized in the OpenSearch specification and the contents of the SOAP messages provided in this specification, the semantic translation between these two web services can be identified.

B.1. SOAP-REST Parameter Mapping
This section provides a mapping between elements and attributes identified in this specification and the corresponding parameters identified in the OpenSearch Specification version 1.2. The use of this mapping demonstrates the interoperability of the two approaches.

B.1.1 Input
In general, the single element of the body of the SOAP Search Request, the /cdr:SearchRequest element, contains the information which is in the OpenSearch query string template.
Table 12. REST to SOAP Mapping

<table>
<thead>
<tr>
<th>REST Parameter</th>
<th>SOAP Element or Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>opensearch:searchTerms</td>
<td>/cdr:SearchRequest/Expression</td>
</tr>
<tr>
<td>opensearch:count</td>
<td>/cdr:SearchRequest/@count</td>
</tr>
<tr>
<td>opensearch:startIndex</td>
<td>/cdr:SearchRequest/@startIndex</td>
</tr>
<tr>
<td>opensearch:startPage</td>
<td>/cdr:SearchRequest/@startIndex</td>
</tr>
<tr>
<td>Not explicitly defined</td>
<td>/cdr:SearchRequest/@queryLanguage</td>
</tr>
<tr>
<td>Not explicitly defined</td>
<td>/cdr:SearchRequest/@responseFormat</td>
</tr>
</tbody>
</table>

B.1.2 Output

The output of both the REST service and the SOAP service is delegated to the result set specifications provided by the CDR IPT. To date, there is only one recognized Search Results specification, the CDR Atom 1.0 Search Results Specification.

Appendix C. Changes from Prior Version

This section outlines the significant changes that were made from the prior version of the IC/DoD CDR Search Service Specification for SOAP Implementations, Version 1.0-20100309. These changes were made to accommodate changes in the CDR-RA and CDR-SF, to harmonize the content of this specification with that contained in the other specifications produced by this group, and to incorporate feedback on the specification from pilot implementations.

C.1. Search Request

C.1.1 Change queryTypeURI to queryLanguage

The prior version of this specification contained the input variable “query type” to provide a mechanism to specify information to describe and assist in processing the Query Expression. The Specification Framework [ref] example in the definition of query type was “(e.g., keyword, XQuery, etc.)”, indicating a diverse set of possible values and little guidance for what use or behavior the input would support. In addition, CDR-OS did not address query types because there is an implicit assumption that the OpenSearch searchTerms parameter provided keyword targets. The Specification Framework deprecated query type and replaced it with Query Metadata to specifically “enable determination of whether a search capability can process the query”. Query language is given in the Specification Framework definition as an example of Query Properties.

In this version of the SOAP Search Specification, in line with the CDR-SF, query language is recognized as one of the Query Properties that can be provided along with a query. Search Services can indicate via their service descriptions which query languages they support. The flexible Query Properties mechanism can be used to provide information on other aspects of a query, such as the metadata format used in the query expression.

C.1.2 Change resultsPerPage to count

The prior version of this specification used the resultsPerPage parameter to indicate the desired number of results to be returned. This was the incorrect variable name to use, as the correct
corresponding OpenSearch variable is count. This version of the specification and the current version of CDR-OS now use the name “count”.

C.1.3 Use of Query Identifier

The prior version of this specification used the element QueryId in some requests to the server to utilize a temporary identifier for a query expression for purposes of accessing of result set cache in paging operations. The specification for this behavior did not fully describe caching behavior, as it described the behavior when both an expression and a query identifier were provided as indeterminate, but demanded a fault be sent as well. It did not specify a period for which the query identifier would be valid. In addition, it was not supported by the schema file as an option to be used in place of the expression. This specification follows the CDR-OS specification in using the Expression, in conjunction with any security related information, such as the identity of the requestor, used to produce the result set, to serve the purpose of the query identifier.

C.2. Search Response

The current specifications identify a search response binding only for Atom Syndication Format. This specification outlines using the Atom Syndication Format as the body of a SOAP message. This is an extensible mechanism which allows additional search response types to be defined. Version 1.0 of the specification did not specifically address response formats, because none were defined.

Version 1.0 discussed a “Results Metadata Format” as an implicit property of a Search Service. This version of the specification explicitly defines a response metadata format attribute that can be used to select an appropriate format for the response.

The timestamp discussed in the previous version is handled by the /atom:feed/atom:updated element in CDR-Atom, and should be handled by any other result types which are defined.

C.3. Guide for Usage

The guide for usage section, Section 4 of Version 1.0 of this specification, was removed from this document. With the simpler approach to what was called query type, the process of using this specification was simplified. This section did not appear in other documents and may end up in a more general guide document that was not contemplated when Version 1.0 of this specification was created.