Federated Search Developer Guide

Version 40.0, Summer '17

Note: This release is in preview. Features described in this document don't become generally available until the latest general availability date that Salesforce announces for this release. Before then, and where features are noted as beta, pilot, or developer preview, we can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features.

@salesforcedocs

Last updated: May 3, 2017

CONTENTS

CHAPTER 1 Federated Search Developer Guide

The Federated Search API connects the Salesforce federated search connector to the external search provider so that data from external repositories can be searched and returned within Salesforce. External search providers have partnered with Salesforce to provide a search service that conforms to the Salesforce API.

Important: Use this API only if your company doesn't have an existing search provider, such as Coveo, Docurated, or Swiftype search. If your company manages your own external data source, developers use this API to create a compliant external search provider.

This API follows the OpenSearch specification with more Salesforce extensions. External search providers who conform to this API remain fully compliant to the OpenSearch specification and can be queried by any OpenSearch client at the same URL endpoint.

Note: A Salesforce org trusts the target host only when presented with a certificate signed by a root Certification Authority (CA). For more information, including a list of trusted CAs, see the SSL Certificates Salesforce Supports Knowledge article.

The federated search connector sends an HTTPS GET request to the external search provider URL endpoint. The search request is encoded in UTF-8. The request includes the OAuth token in the request header bearer token in the authorization request header.

This request also sets the values of the URL template parameters, as defined by the OpenSearchDescription. The values include the search terms and the user identity. Other parameters can be part of the URL. The values also set the Salesforce ID, login, and email address of the user who issued the search query.

The search query accepts up to 10,000 characters. An error is returned if the query exceeds the limit.

The external search provider is expected to respond within less than 1 second, up to a maximum of 120 seconds, after which a timeout message is displayed. The admin can change the maximum delay when configuring the external data source.

The external search provider returns results in Atom format encoded in UTF-8, as defined by the OpenSearch and Atom specifications.

More External Resources

- OpenSearch website
- Atom specification website
- Bearer Token specification for the authorization request header field website
- E.123 : Notation for national and international telephone numbers, e-mail addresses and web addresses website

CHAPTER 2 Constraints

Because this API follows the OpenSearch specification, the API requires the same elements as the OpenSearch specification. Constraints specific to this API are also required.

Element	Constraint
`<OpenSearchDescription>`	Required exactly once. The namespace is optional if it is used in the OpenSearchDescription.
`<Url type="application/atom+xml" rel="results">`	Required exactly once. Other `<Url>` elements for other `type` and `rel` values can be included, but you must have exactly one `<Url>` with both `type="application/atom+xml"` and `rel="results"`. API version 1 supports only the HTTPS request GET method.
`<InputEncoding>` and `<OutputEncoding>`	Must declare at least UTF-8, which is the default encoding when those elements are omitted.

CHAPTER 3 Extensions

This API follows the OpenSearch specification with more Salesforce extensions declared. The additional Salesforce extensions support Salesforce-specific search features, and use the Salesforce Federated Search namespace, denoted by the sfdc prefix in attribute names.

Attribute	Description
`<sfdc:Version>`	Optional. Zero or 1 occurrence. Declares that the OpenSearch description and the external search provider are compliant to a specific Salesforce Federated Search API version. First version is version 1. If not present, it is considered version 1.

`<sfdc:RecordTypes>`	Optional. If not present, the search results are not typed and only the standard OpenSearch fields are supported. Zero or more occurrences. Declares the complete list of record types supported by the external search provider. Each record type is declared in an `<sfdc:RecordType>` child element with zero or more occurrences and a case-sensitive name attribute.
	
	The attribute name must be unique and can contain only ASCII alphanumeric characters, spaces (" ", U+0020), or hyphens ("-", U+002D). The maximum length is 80 characters. All `RecordType` names must be distinct.
	
	For example, the following declares that the external search provider restricts the search scope to *Blog Post*, *Medical Record*, or *Supervisor*. Any distinct value can be declared. These record type values are meant to be passed to the extended parameter `{sfdc:recordType}` of the search request URL template.
	
	`<sfdc:RecordTypes>`
	`  <sfdc:RecordType name="Blog Post"/>`
	`  <sfdc:RecordType name="Medical Record"/>`
	
	`  <sfdc:RecordType name="Supervisor"/>`
	`</sfdc:RecordTypes>`

`<sfdc:Field>`	Optional. Zero or more occurrences. Declares more custom fields that can be returned as elements of the search results for the record type. These custom fields can have any distinct name and are in addition to the common fields. When no custom field is declared, search results are expected to contain only common fields. Child element of `<sfdc:RecordType>`, with a case-sensitive name, type and sortable attributes: • `name`—Required. Declares the name of the field. Must only contain ASCII alphanumeric characters, spaces (" ", U+0020), or hyphens ("-",U+002D). The maximum length is limited to 40

Attribute	Description

	characters. All `Field` names must be distinct for a given record type. Custom fields must not have the same name as standard Atom result fields (title, id, link, summary, content, updated, published). Fields with duplicate names are ignored. However, it is possible to declare a field with a standard Atom field name. This action doesn't add a custom field. Rather, it defines whether it is sortable for the parent record type. The type is ignored for a standard Atom field.
	• `type`—Required. Declares the field type. The attribute value can be `string`, `longstring`, `number`, `boolean`, `percent`, `currency`, `email`, `url`, `phone`, or `date`. The phone format follows the E.123 standard. For example: +31 42 1123 4567. The date format follows the Atom date format. For example: 2003-12-13T18:30:02.25+01:00.
	• `sortable`—Optional. Declares whether the field can be used to sort search results. The attribute value can be `true` or `false`. The default is `true`, or sortable. If the field is sortable, its name can be passed as the value of the URL template parameter `{sfdc:sortField}`.
	For example, the following declares that results of record type *Blog Post* can have the additional custom fields *Author*, *Agent*, and *Relevant Tags*, which all type string, and all are sortable. It also declares that the standard Atom field *link* is not sortable for *Blog Post*.
	`<sfdc:RecordTypes>`
	`<sfdc:RecordType name="Blog Post">`
	`<sfdc:Field name="Author" type="string" sortable="true"/>`
	`<sfdc:Field name="Agent" type="string" sortable="true"/>`
	`<sfdc:Field name="Relevant Tags" type="string" sortable="true"/>`
	`<sfdc:Field name="link" type="url" sortable="false"/>`
	`</sfdc:RecordType>`
	`</sfdc:RecordTypes>`

`<sfdc:maxCount>`	Optional. Integer value greater than or equal to 1. Declares that the external search provider serves up to a specified number of results per page. It defines the maximum value for the count parameter of the URL template. The federated search connector doesn't ask for more results than this limit per page. By default, if this attribute is absent, the count parameter has no declared limit. In all cases, the value of the count parameter might not be honored by the external search provider, as stated in the OpenSearch specification. It's a parameter of the `<Url>` element.

Attribute	Description
`<sfdc:maxTotalResult>`	Optional. Integer value greater than or equal to 1. Declares that the external search provider serves up to a specified number of results per search. It defines the maximum value for the (startIndex + count) parameters of the URL template. The federated search connector doesn't ask for more results than this limit per page. By default, if this attribute is absent, there is no declared limit. It's a parameter of the `<Url>` element.

CHAPTER 4 Request: URL Template Parameters

All URL template parameters are URL-encoded and sent via an HTTPS GET method. This list includes all template parameters used by the Salesforce federated search connector.

OpenSpace Namespace

Parameter	Description
`{searchTerms}`	Replaced by the user search query terms as they are entered in the Salesforce search field. They use the SOSL SearchQuery syntax. A SOSL clause is not supported. See the Resources section for a link to the SOSL documentation.
`{count}`	The number of search results per page desired by the search client.
`{startIndex}`	By default, the index of the first result is 1, unless defined otherwise by the indexOffset attribute of the `<Url>` element.
`{inputEncoding}`	If present in the template, this parameter is always set to UTF-8.
`{outputEncoding}`	If present in the template, this parameter is always set to UTF-8.

Salesforce Federated Search Namespace

Parameter	Description
`{sfdc:userId}`	The unique ID of the user who triggered the search. Because OAuth authentication is done with the same bearer token for all users, it authorizes, but does not identify, the user. This parameter gives the identity of the user.
`{sfdc:userLogin}`	The login name of the user who triggered the search.
`{sfdc:userEmail}`	The email address of the user who triggered the search.
`{sfdc:orgId}`	The org ID of the user who triggered the search.

Parameter	Description
`{sfdc:searchSyntax}`	Declares the syntax of the {searchTerms} parameter. Its value is always SOSL (uppercase). The search provider can use the parameter to determine precisely which syntax parser to use.
`{sfdc:searchById}`	When the value of this boolean parameter is `true`, it restricts the search to a single record by its ID. At the same time, `{searchTerms}` is equal to the record ID. This parameter is used by the federated search connector to retrieve specific record data. If the OpenSearch description supports this parameter, the federated search connector sets this parameter to `true` and the search is restricted to the single record corresponding to the ID specified in `{searchTerms}`. If the OpenSearch description does not support this parameter, the federated search connector searches the record with the first few terms of the record title and selects from the results only the record with the expected ID. This type of search is referred to as a fallback search, and can be less accurate than searching directly on the record ID.
`{sfdc:recordType}`	Restricts the scope of the search request to a specific record type. Its value is the name of a declared `<sfdc:RecordType>`. The supported record types are declared with the `<sfdc:RecordTypes>` element.
`{sfdc:sortField}`	Defines the name of the field by which search results are sorted. If it is sortable, the field can be one of the common fields of the Atom format (title, id, link, summary, content, updated, published). Otherwise, set a declared sortable custom field by using `<sfdc:RecordType><sfdc:Field sortable="true">`. By default, this parameter is not present, and the search results are sorted by relevance. The API intentionally does not follow the OpenSearch Community extension SRU because the Salesforce implementation is limited to sorting on a single field, without a sort schema.
`{sfdc:sortDirection}`	Defines the sort direction with a value of either ascending or descending. Can be added only if the

Parameter	Description
	`sfdc:sortField` parameter is also present. By default, the search results are sorted in ascending order. This parameter isn't used for relevance sorting.

CHAPTER 5 Response: Atom Elements

The external search provider returns results to the federated search connector in Atom format. However, the federated search connector extracts only the specified elements.

Feed Elements

These fields are defined in the OpenSearch standard. The federated search connector extracts the value from these elements.

- `<opensearch:totalResults>`
- `<opensearch:startIndex>`

Entry Elements: Common Fields

These fields are common to all record types.

Field	Description
`<title>`	Required. Title of the search result. Limited to 255 characters, beyond which the text is truncated.
`<id>`	Required. External ID of the search result. The ID can be reused to search for the corresponding specific record. Maximum length is 255 characters; otherwise, the entry is skipped. See the `{sfdc:searchById}` parameter in the URL Template request parameters for more details.
`<link>`	Optional. Clickable, well-formatted URL link to open the external document. The federated search connector encodes this URL. The external search provider can proxy it to track clicks. Maximum length is 1,000 characters; otherwise, the entry is skipped.
`<summary>`	Optional. Snippet for the search result. Can be replaced by `<content>`. Maximum length is 500 characters, beyond which the text is truncated.
`<content>`	Optional. Snippet for the search result for text. Can be replaced by `<summary>`. Maximum length is 500 characters, beyond which the text is truncated.

Field	Description
`<updated>`	Required. Date of the last modification of the entry. Follows the Atom date format. For example: `2003-12-13T18:30:02.25+01:00`.
`<published>`	Optional. Date of the creation of the entry. Follows the Atom date format. For example: `2003-12-13T18:30:02.25+01:00`.
`<sfdc:recordType>`	Required if the external search provider uses Salesforce extensions; otherwise optional. Type of the result document. This element also includes more fields provided for the result. It must have a single value listed by the `<sfdc:RecordTypes>` element of the OpenSearchDescription. Provided as text element, such as `<sfdc:recordType>BlogPost</sfdc:recordType>`.
`<sfdc:link>`	Required if the external search provider uses the Salesforce extension; otherwise optional. If the external search provider uses Salesforce extensions, the URL must point to the app installed by the admin. This app then renders the external document inside Salesforce. This field represents a clickable link to open the external document within Salesforce. Might not be a well-formed URL. It isn't URL-encoded by the federated search connector. Maximum length is 1,000 characters; otherwise, the entry is skipped.

Entry Elements: Custom Fields

These fields are custom and have been declared in the `OpenSearchDescription` with the `<sfdc:Field>` child element of `<sfdc:RecordType>`. A custom field value is provided as a text element.

If a custom field has been declared with a name containing spaces, such as *Relevant Tags*, it can be referred to by replacing all spaces with underscores, such as `<sfdc:Relevant_Tags>`.

Custom field maximum lengths depend on the type of field. A custom field is skipped if it exceeds the limit. Here are the limits for field types. The phone field type follows the E.123 standard, and the date field follows the Atom date format.

Field Type	Character Limit
string	255
longstring	500

Field Type	Character Limit
number	20
boolean	10
percent	20
currency	20
email	500
url	1,000

CHAPTER 6 Error Codes

Codes from the federated search connector.

Error Code	Description
400	Missing or invalid parameter
401	Invalid authentication
403	Unrecognized user login
414	Request-URI too long

Optional Codes

Error Code	Description
480	Unidentified user or unrecognized user login
481	Missing required parameter
482	Invalid parameter value

CHAPTER 7 Examples

The provided example uses a search for blog posts and medical records. A basic understanding of the OpenSearch specification is required.

The OpenSearch protocol is composed of two parts:

- The OpenSearch XML description, which describes how to search the external search engine and which features it supports. The description is provided by the search engine itself at a public URL. The main element of the description is the `<Url>`, in which the template attribute provides the URL to call and the supported parameters with semantics.
- The search results returned by the external search engine, in Atom format, as the response of the HTTP search request.

OpenSearch Description

```
<?xml version="1.0" encoding="UTF-8"?>
<OpenSearchDescription xmlns="http://a9.com/-/spec/opensearch/1.1/"
xmlns:sfdc="http://salesforce.com/2016/federatedsearch/1.0">
  <ShortName>Document Search</ShortName>
  <Contact>support@example.com</Contact>
  <Url type="application/atom+xml" sfdc:maxTotalResults="500"
template="https://example.com/search?q={searchTerms}&amp;
byId={sfdc:searchById?}&amp;type={sfdc:recordType}&amp;
user={sfdc:userEmail}&amp;start={startIndex?}&amp;num={count?}&amp;
sf={sfdc:sortField?}&amp;sd={sfdc:sortDirection?}"/>
  <sfdc:RecordTypes>
    <sfdc:RecordType name="Blog Post">
      <sfdc:Field name="Author" type="string" sortable="true"/>
     <sfdc:Field name="Expiration date" type="date" sortable="true"/>

    </sfdc:RecordType>
    <sfdc:RecordType name="Medical Record">
      <sfdc:Field name="Main author" type="string" sortable="true"/>

      <sfdc:Field name="Reviewed" type="boolean" sortable="false"/>
    </sfdc:RecordType>
  </sfdc:RecordTypes>
  <Attribution>
    Copyright 2016, All Rights Reserved
  </Attribution>
  <InputEncoding>UTF-8</InputEncoding>
  <OutputEncoding>UTF-8</OutputEncoding>
  <sfdc:Version>1</sfdc:Version>
</OpenSearchDescription>
```

Search Results

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns="http://www.w3.org/2005/Atom"
xmlns:opensearch="http://a9.com/-/spec/opensearch/1.1/"
xmlns:sfdc="http://salesforce.com/2016/federatedsearch/1.0">
  <title>Example Search: test</title>
  <link>http://example.com</link>
  <updated>2016-11-21T10:19:37.644Z</updated>
  <author>
    <name>Example</name>
  </author>
  <id>urn:uuid:6b63a81d-0a77-48e3-8a74-5d5f34b9da98</id>
  <opensearch:totalResults>10</opensearch:totalResults>
  <opensearch:startIndex>0</opensearch:startIndex>
  <opensearch:itemsPerPage>2</opensearch:itemsPerPage>
  <opensearch:Query role="request" searchTerms="test" startIndex="0"
 count="2"/>
  <entry>
    <title>Unit testing framework</title>
    <id>https://example.com/id:GhITRcjf5go_JLero45</id>
    <link href="https://www.example.com/library/unittest.html"/>

<sfdc:link>/apex/ExampleApp_view?id=GhITRcjf5go_JLero45&amp;q=test</sfdc:link>

    <sfdc:recordType>Blog Post</sfdc:recordType>
    <summary>The unit testing framework is ...</summary>
    <updated>2016-07-13T15:24:57.000Z</updated>
    <published>2016-07-13T15:24:57.000Z</published>
    <sfdc:Author>Maker</sfdc:Author>

<sfdc:Expiration_date>2017-07-13T15:24:57.000Z</sfdc:Expiration_date>

  </entry>
  <entry>
    <title>Cancer tests | Cancer Research</title>
    <id>https://example.com/id:UCAuUUnT6oDeKwE6v1</id>
    <link href="https://www.example.com/library/cancertest.html"/>

<sfdc:link>/apex/ExampleApp_view?id=UCAuUUnT6oDeKwE6v1&amp;q=test</sfdc:link>

    <sfdc:recordType>Medical Record</sfdc:recordType>
    <summary>A short overview about current cancer research
...</summary>
    <updated>2012-08-15T15:20:54.000Z</updated>
    <published>2012-08-15T15:20:54.000Z</published>
    <sfdc:Main_author>Maker</sfdc:Main_author>
    <sfdc:Reviewed>true</sfdc:Reviewed>
  </entry>
</feed>
```

INDEX