

Google Search Appliance

Administrative API Developer's Guide: Protocol

Google Search Appliance software version 7.2 and later



Google, Inc.
1600 Amphitheatre Parkway
Mountain View, CA 94043
www.google.com

GSA-API_200.01
March 2015

© Copyright 2015 Google, Inc. All rights reserved.

Google and the Google logo are, registered trademarks or service marks of Google, Inc. All other trademarks are the property of their respective owners.

Use of any Google solution is governed by the license agreement included in your original contract. Any intellectual property rights relating to the Google services are and shall remain the exclusive property of Google, Inc. and/or its subsidiaries ("Google"). You may not attempt to decipher, decompile, or develop source code for any Google product or service offering, or knowingly allow others to do so.

Google documentation may not be sold, resold, licensed or sublicensed and may not be transferred without the prior written consent of Google. Your right to copy this manual is limited by copyright law. Making copies, adaptations, or compilation works, without prior written authorization of Google, is prohibited by law and constitutes a punishable violation of the law. No part of this manual may be reproduced in whole or in part without the express written consent of Google. Copyright © by Google, Inc.

Contents

Administrative API Developer's Guide: Protocol	5
Introduction	5
API Operations	5
Authenticating Your Google Search Appliance Account	6
How the API Works	7
XML Element Definitions	7
Content Sources	11
Crawl URLs	12
Data Source Feed	13
Feeds Trusted IP Addresses	16
Crawl Schedule	17
Crawler Access Rules	19
Host Load Schedule	22
Freshness Tuning	23
Recrawl URL Patterns	24
Connector Managers	25
OneBox Settings	27
OneBox Modules	28
Crawl Status	30
Document Status	31
Index	31
Collections	32
Index Diagnostics	34
Content Statistics	40
Reset Index	42
Search	43
Front Ends, Remove URLs, and Relative OneBoxes	44
Output Format XSLT Stylesheet	46
KeyMatch	48
Related Queries	51
Query Suggestion	54
Search Status	55
Reports	56
Search Reports	56
Search Logs	61

GSA Unification	65
Configuring a GSA Unification Network	66
Adding a GSA Unification Node	66
Retrieving a Node Configuration	67
Retrieving All Node Configurations	68
Updating a Node Configuration	69
Deleting a Node	69
Administration	69
License Information	69
Import and Export	71
Event Log	72
System Status	73
Shut Down and Reboot	74
Index	75

Administrative API Developer's Guide: Protocol

Introduction

The Google Search Appliance Administration API enables administrators to configure a search appliance programmatically. This API provides functions for creating, retrieving, updating, and deleting search appliance configuration settings.

The Google Search Appliance Administration API follows the principles of the Google Data APIs. Google Data APIs are based on both the Atom 1.0 and RSS 2.0 syndication formats in addition to the Atom Publishing Protocol.

The audience for this guide are XML programmers who have access to a Google Search Appliance. The user name and password for the Admin Console are required to obtain the authentication token necessary to run applications for this API.

This guide consists of the following sections:

- "Content Sources" on page 11
- "Index" on page 31
- "Search" on page 43
- "Reports" on page 56
- "GSA Unification" on page 65
- "Administration" on page 69

API Operations

To use this API, you can send HTTP requests to a search appliance to instruct the search appliance to perform a create, retrieve, update, or delete configuration information in the search appliance.

Note: The sections that follow indicate the corresponding Java client library methods. Parallel methods are also available in other client libraries.

This section explains the different types of operations that the API supports. See also "How the API Works" on page 7, which identifies the URL that corresponds to each API operation.

The operations are as follows:

- **Create**—Operations to add a new object, such as a collection or front end. To perform any of these operations, issue an HTTP `POST` request with the appropriate URL. The body of the `POST` request is an XML document that contains information about a resource to create.
- **Retrieve**—Operations to request and obtain information about search appliance features. For information on the Google Data API retrieval operations, see the *Google Search Appliance Administrative API Developer's Guide: Java* and *Google Search Appliance Administrative API Developer's Guide: .NET*. To retrieve information about a resource, issue an HTTP `GET` request to the appropriate URL that identifies a resource to retrieve.
- **Update**—Operations to modify information about search appliance. To update the information, issue an HTTP `PUT` request to the appropriate URL. The body of the `PUT` request is an XML document that contains information about a resource to update.
- **Delete**—Operations to delete objects such as a collection or a front end. To perform any of these operations, issue an HTTP `DELETE` request to the appropriate URL. The URL contains information that identifies a resource to delete.

The search appliance verifies that all create and update requests contain valid XML, include all required data fields, and meet authentication requirements.

Authenticating Your Google Search Appliance Account

You can send API requests over HTTPS or HTTP.

Specify an authentication token with each API request. The search appliance uses the token to authorize access to the operation that you request. Authentication tokens are available only to users who have administrative rights to the search appliance, and the tokens authorize operations only within a search appliance.

To obtain an authentication token, submit an HTTP `POST` request to port 8443 on a search appliance as shown in the following URL:

```
https://Search_Appliance:8443/accounts/ClientLogin
```

The following guidelines apply to the request:

- Include in the `POST` body a string in the following format:

```
&Email=username&Passwd=password
```

Make the following changes to this string:

- Replace *username* with a user name that has an Admin Console administrator account.
 - Replace *password* with the password for the Admin Console account.
- The user name and password values must be URL-encoded. For example, the URL-encoded form of the `AcQ.87@` password is the `AcQ%2E87%40` value.
 - The `POST` request must specify the value `application/x-www-form-urlencoded` for the `Content-Type` header.

The search appliance returns a response containing your authentication token in response to a `POST` request. The authentication token is the `Auth` value on that page, and you need to extract the token from the page. When you submit an API request, you must set the `Content-Type` and authorization headers as follows:

```
Content-type: application/atom+xml
Authorization: GoogleLogin auth=your-authentication-token
```

Note: Authentication tokens expire after 24 hours or 30 minutes when not in use. Submit a request to the URL at least once again. We recommend that you keep the token in memory rather than writing the token to a file.

How the API Works

To execute an operation using the API, submit an HTTP `POST`, `GET`, `PUT`, or `DELETE` request to the URL that corresponds to the operation that you wish to perform. Each URL includes variables that identify the resource that you are creating, retrieving, updating or deleting. The URL pattern is as follows:

```
http://Search_Appliance:8000/feeds/Collection_Name/Entry
```

The `Collection_Name` and `Entry` values indicate a search appliance configuration. Note that all create and update requests (`POST` and `PUT` requests) also require that you submit an XML document that contains the information you need to fulfill the request. Send the content using the `application/atom+xml` content type. The section “XML Request Formats” on page 10 explains the XML structures.

XML Element Definitions

The following XML elements can be used in a reporting API request. The elements are listed in the order that they appear in an API request.

Note: In API requests, the ampersand (&) character must be XML-escaped as `&` when used in `<gsa:content ...>` values. For example:

```
<gsa:content name='followURLs'>^http://my.domain.com/index.php?a=2&amp;b=1</gsa:content>
```

atom:feed

Definition

The `<atom:feed>` element encapsulates an API response to a request to retrieve all the information in one configuration collect.

Example

```
<atom:feed xmlns="http://www.w3.org/2005/Atom"
xmlns:openSearch="http://a9.com/-/spec/opensearchrss/1.0/">
```

Child Elements

atom:id, atom:link, atom:entry

Content Format

Container

atom:entry

Definition

The <atom:entry> encapsulates an API request or an API Atom response.

Child Elements

atom:id, gsa:content, atom:link

Content Format

Container

atom:id

Definition

The <atom:id> element's value identifies a permanent, unique identifier for a feed. This element is included in API responses.

Example

```
<atom:id>https://gsa/feeds/config/crawlURLs</atom:id>
```

Child Element

atom:entry

Content Format

String (IRI)

atom:link

Definition

The <atom:link> tag provides an RFC 3987 IRI reference (<http://www.ietf.org/rfc/rfc3987.txt>) related to an API results feed or a resource in the feed.

Attributes

Name	Format	Description
<code>rel</code>	Text	<p>The <code>rel</code> attribute identifies the relationship of the link to the API response feed.</p> <ul style="list-style-type: none">If the value of the <code>rel</code> attribute is <code>self</code>, then the <code>href</code> attribute value is a link to the URL you use to request the feed.If the value of the <code>rel</code> attribute is <code>edit</code>, then the <code>href</code> attribute value is the URL that you use to retrieve, update, or delete the resource. <p>Note: Use an HTTP <code>GET</code> request to retrieve a resource, an HTTP <code>PUT</code> request to update a resource, and an HTTP <code>DELETE</code> request to delete a resource.</p>
<code>href</code>	Text	The <code>href</code> attribute contains an URI reference that indicates how to retrieve or edit the information in an API response.

Example

```
<atom:link rel="edit" type="application/atom+xml"
href="https://gsa/feeds/config/crawlURLs"/>
```

Parent Element

atom:entry

Content Format

Empty

atom:updated

Definition

The `<atom:updated>` tag identifies the date and time that an entry in an Atom feed was updated.

Example

```
<atom:updated>1970-01- 01T00:00:00.000Z</atom:updated>
```

Parent Elements

atom:feed, atom:entry

Content Format

Date

gsa:content

Definition

The `<gsa:content>` tag specifies properties of the search appliance Admin Console settings. The `<entry>` must contain at least one `<gsa:content>`. The attribute `name` specifies the name of property and the value for the property should be put in `content`.

Example

```
<gsa:content name='crawlURLs'>http://yourdomain.com/</gsa:content>
```

Parent Element

atom:entry

Content Format

Complex

XML Request Formats

For API requests to create or update information (HTTP `POST` and `PUT` requests), the body of a request must be an XML document that provides the data necessary to complete a request.

For API requests to retrieve or delete information (HTTP `GET` and `DELETE` requests), the URL and HTTP request type specify all of the information that the search appliance needs to fulfill the request. Put all necessary information in the `<gsa:content>` XML tag.

The following example updates the crawl URLs in a search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://ent1:8000/feeds/config/crawlURLs</id>
  <gsa:content name='crawlURLs'>http://yourdomain.com/</gsa:content>
  <gsa:content name='startURLs'>http://yourdomain.com/</gsa:content>
  <gsa:content name='doNotCrawlURLs'>
    http://yourdomain.com/not_allow
  </gsa:content>
</entry>
```

XML Response Formats

Depending on the API request, the search appliance Administrative API returns XML responses. The XML response is a Google Data Atom entry. The `<entry>` must contain at least one `<gsa:content>`. All the search appliance related information are put in `<gsa:content>` XML tag. For example, the following list defines a `GSAEntry` response as an XML document that contains information about the crawl URLs. The client libraries convert this XML response into a `GSAEntry` object.

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://ent1:8000/feeds/config/crawlURLs</id>
  <updated>2008-12-08T20:11:58.342Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/crawlURLs' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/crawlURLs' />
  <gsa:content name='entryID'>crawlURLs</gsa:content>
  <gsa:content name='crawlURLs'>http://yourdomain.com/</gsa:content>
  <gsa:content name='startURLs'>http://yourdomain.com/</gsa:content>
  <gsa:content name='doNotCrawlURLs'>
    http://yourdomain.com/not_allow
  </gsa:content>
</entry>
```

Content Sources

The sections that follow describe how to configure the **Content Sources** features of the Admin Console:

- “Crawl URLs” on page 12
- “Data Source Feed” on page 13
- “Feeds Trusted IP Addresses” on page 16
- “Crawl Schedule” on page 17
- “Crawler Access Rules” on page 19
- “Host Load Schedule” on page 22
- “Freshness Tuning” on page 23
- “Recrawl URL Patterns” on page 24
- “Connector Managers” on page 25
- “OneBox Settings” on page 27
- “OneBox Modules” on page 28
- “Crawl Status” on page 30
- “Document Status” on page 31

Crawl URLs

Retrieve and update crawl URLs for a search appliance using the `crawlURLs` entry of the `config` feed.

Property	Description
<code>doNotCrawlURLs</code>	Do not crawl URLs with the following URL patterns.
<code>followURLs</code>	Follow and crawl only URLs with the following URL patterns.
<code>startURLs</code>	Start crawling from the following URL patterns.

Retrieving Crawl URLs

To get the crawl URLs information for a search appliance, send an authenticated `GET` request to the `config` feed URL:

```
http://Search_Appliance:8000/feeds/config/crawlURLs
```

The following example requests the current crawl URLs values from a search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/config/crawlURLs</id>
  <updated>2008-12-12T07:49:32.957Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/crawlURLs' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/crawlURLs' />
  <gsa:content name='entryID'>crawlURLs</gsa:content>
  <gsa:content name='startURLs'>http://www.example.com/</gsa:content>
  <gsa:content name='doNotCrawlURLs'>.xls$</gsa:content>
  <gsa:content name='followURLs'>http://www.example.com/</gsa:content>
</entry>
```

Updating Crawl URLs

To update Crawl URLs information for a search appliance, send an authenticated `PUT` request to the `config` feed URL:

```
http://Search_Appliance:8000/feeds/config/crawlURLs
```

The following example overwrites the crawl URLs specified in the entry to update:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/config/crawlURLs</id>
  <gsa:content name='entryID'>crawlURLs</gsa:content>
  <gsa:content name='startURLs'>http://www.example.com/</gsa:content>
  <gsa:content name='doNotCrawlURLs'>.xls$</gsa:content>
  <gsa:content name='followURLs'>http://www.example.com/</gsa:content>
</entry>
```

Data Source Feed

Retrieve, delete, and destroy data source feed information for a search appliance using the `feed` feed. The Google Search Appliance supports an interface known as the “feeds interface,” which is different from a Google Data API feed. To differentiate between these terms, the feeds interface on the search appliance is referred to as a data source feed. For more information on data source feeds, see the *Feeds Protocol Developer's Guide*.

Parameter	Description
<code>query</code>	To get all feed information, this parameter is the feed data source (<code>feedDataSource</code>). To get information on a single feed, this parameter is the query string. Each log line contains a query string to retrieve.
<code>startLine</code>	The first log line to retrieve. The default value is line 1.
<code>maxLines</code>	The maximum number of log lines to retrieve. The default value is 50 lines.

The following properties provide data source feed information.

Property	Description
<code>errorRecords</code>	The number of documents that had errors and were not added to the data source feed.
<code>feedDataSource</code>	The name of the data source feed.
<code>feedState</code>	Feed state: <code>ACCEPTED:0</code> , <code>IN_PROGRESS:1</code> , <code>COMPLETED:2</code> , <code>COMPLETED_WITH_ERROR:3</code> , <code>FAILED_IN_ERROR:4</code>
<code>feedTime</code>	The timestamp for the search appliance at the start of each stage (in milliseconds).
<code>feedType</code>	Feed type, <code>FULL_FEED:0</code> , <code>INCREMENTAL:1</code> , <code>DELETED:2</code> , <code>METADATA_AND_URL:3</code>
<code>fromLine</code>	The starting line of the log.
<code>logContent</code>	The log content.
<code>successRecords</code>	The number of documents that have completed indexing.
<code>toLine</code>	The end line of the log.
<code>totalLines</code>	The total lines in the log.
<code>updateMethod</code>	The command sent to a search appliance to delete a data source feed. The value can only be <code>delete</code> .

Note: You can only get information about each data source feed, and whether to delete or destroy a feed. Inserting a new data source feed is not provided in this API.

Retrieving Data Source Feed Information

To retrieve information about all data source feeds for a search appliance, send an authenticated GET request to the `feed` feed URL:

```
http://Search_Appliance:8000/feeds/feed?query=feedDataSource
```

The following example result includes current feeds values for the search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
  xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/feed</id>
  <updated>2008-12-12T12:57:22.970Z</updated>
  <link rel='http://schemas.example.com/g/2005#feed'
    type='application/atom+xml'
    href='http://gsa:8000/feeds/feed' />
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/feed' />
  <generator version='0.5' uri='http://gsa:8000/gsa'>
    Google Search Appliance
  </generator>
  <openSearch:startIndex>1</openSearch:startIndex>

  <entry>
    <id>http://gsa:8000/feeds/feed/Feed_ID</id>
    <updated>2008-12-12T12:57:22.970Z</updated>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/feed' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/feed' />
    <gsa:content name='entryID'>
      sample_feed2_20081212_005647_000000_FULL_FEED_0
    </gsa:content>
    <gsa:content name='errorRecords'>0</gsa:content>
    <gsa:content name='successRecords'>1</gsa:content>
    <gsa:content name='feedType'>0</gsa:content>
    <gsa:content name='feedDataSource'>sample_feed2</gsa:content>
    <gsa:content name='feedState'>2</gsa:content>
    <gsa:content name='feedTime'>1229072207000</gsa:content>
  </entry>

  <entry>
    <id>http://gsa:8000/feeds/feed/Feed_ID</id>
    <updated>2008-12-12T12:57:22.970Z</updated>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/feed' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/feed' />
    <gsa:content name='entryID'>
      sample_feed_20081212_005123_000000_FULL_FEED_0
    </gsa:content>
    <gsa:content name='errorRecords'>1</gsa:content>
    <gsa:content name='successRecords'>0</gsa:content>
    <gsa:content name='feedType'>0</gsa:content>
    <gsa:content name='feedDataSource'>sample_feed</gsa:content>
    <gsa:content name='feedState'>4</gsa:content>
    <gsa:content name='feedTime'>1229071883000</gsa:content>
  </entry>
</feed>
```

Note: To get information about all feeds, specify a query to get the `feedDataSource` value. Alternatively, you can get all the feeds if you do not supply a query. Whether or not you supply a query, you can get information about at most five feeds for each `feedDataSource` value.

To get information about individual feeds from a search appliance, send an authenticated `GET` request to the `feed` URL:

```
http://Search_Appliance:8000/feeds/feed/Feed_File_ID
```

The result is an entry that includes current values for an individual feed:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/feed/Feed_ID</id>
  <updated>2008-12-12T13:03:27.434Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/feed/Feed_ID'>
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/feed/Feed_ID/'>
  <gsa:content name='entryID'>
    sample_feed_20081212_005123_000000_FULL_FEED_0
  </gsa:content>
  <gsa:content name='toLine'>1</gsa:content>
  <gsa:content name='errorRecords'>1</gsa:content>
  <gsa:content name='successRecords'>0</gsa:content>
  <gsa:content name='logContent'>
    ProcessNode: Not match URL patterns, skipping record with URL:
    http://www.sample_feed.com/sample_data.html
  </gsa:content>
  <gsa:content name='feedType'>0</gsa:content>
  <gsa:content name='fromLine'>1</gsa:content>
  <gsa:content name='totalLines'>1</gsa:content>
  <gsa:content name='feedDataSource'>sample_feed</gsa:content>
  <gsa:content name='feedState'>4</gsa:content>
  <gsa:content name='feedTime'>1229071883000</gsa:content>
</entry>
```

Note: The feed log of each data source feed can only be retrieved as an individual feed.

Deleting a Data Source Feed

To delete a data source feed from a search appliance, you must delete one of its individual feed files by sending an authenticated `PUT` request to the `feed` URL:

```
http://Search_Appliance:8000/feeds/feed/Feed_File_ID
```

The `Feed_File_ID` used in this command corresponds to an `entryID`, as shown in “Retrieving Data Source Feed Information.” To delete a data source, you must delete one of its feed files.

Use the following XML for the `PUT` request:

```
<?xml version='1.0' encoding='UTF-8'?>
<atom:entry xmlns:atom="http://www.w3.org/2005/Atom" xmlns:gsa="http://
schemas.google.com/gsa/2007">
  <gsa:content name="updateMethod">delete</gsa:content>
</atom:entry>
```

Note: You can only delete full or incremental feed types. After deleting, the deleted feed name continues to exist, but has a feed type of `DELETED`. To remove a feed from existence use the `destroy` option.

Destroying a Data Source Feed

To destroy a data source feed from a search appliance, send an authenticated `DELETE` request to the feed feed URL:

```
http://Search_Appliance:8000/feeds/feed/Feed_File_ID
```

Note: You can only destroy a data source feed after you delete the feed.

Feeds Trusted IP Addresses

Retrieve and update the trusted IP addresses for feeds for a search appliance using the `feedTrustedIP` entry of the `config` feed.

Property	Description
<code>trustedIPs</code>	Trusted IP addresses: Either a list of IP addresses or <code>all</code> , which means trust all IP addresses. Separate multiple IP addresses with white space.

Retrieving Feeds Trusted IP Addresses

To get the feeds trusted IP address information for a search appliance, send an authenticated `GET` request to the `config` feed URL:

```
http://Search_Appliance:8000/feeds/config/feedTrustedIP
```

The result is an entry that includes current feeds trusted IP values for the search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/config/feedTrustedIP</id>
  <updated>2008-12-12T09:17:20.830Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/feedTrustedIP' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/feedTrustedIP' />
  <gsa:content name='entryID'>feedTrustedIP</gsa:content>
  <gsa:content name='trustedIPs'>all</gsa:content>
</entry>
```


Updating Feeds Trusted IP Addresses

To update feeds trusted IP information for a search appliance, send an authenticated `PUT` request to the `config feed` URL:

```
http://Search_Appliance:8000/feeds/config/feedTrustedIP
```

The following example updates the feeds trusted IP specified in an entry:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/config/feedTrustedIP</id>
  <gsa:content name='entryID'>feedTrustedIP</gsa:content>
  <gsa:content name='trustedIPs'>127.0.0.1</gsa:content>
</entry>
```

Crawl Schedule

Retrieve and update the crawl schedule of a search using the `crawlSchedule` entry of the `config feed`.

Property	Description
<code>isScheduledCrawl</code>	<p>Displays 1 if the search appliance is in scheduled crawl mode or 0 if the search appliance is in continuous crawl mode.</p> <p>You can also change crawl modes by setting 1 for scheduled crawl or 0 for continuous crawl mode.</p>
<code>crawlSchedule</code>	<p>The schedule of crawl, only available in scheduled crawl mode.</p> <p>The <code>crawlSchedule</code> value is in format: <i>Day, Time, Duration</i>.</p> <p>Where:</p> <ul style="list-style-type: none">• <i>Day</i> is a number representation for days of a week,• 0 means Sunday and 1 means Monday.• <i>Time</i> is 24 hour representation of time.• <i>Duration</i> is the representation for time period in minutes and it should not be longer than 1440 which mean 24 hours. <p>A scheduled crawl begins on the <i>Day</i> and <i>Time</i> and continues for the specified <i>Duration</i>.</p>

Retrieving a Crawl Schedule

To check the crawl mode and get the crawl schedule, send an authenticated `GET` request to the following URL:

```
http://Search_Appliance:8000/feeds/config/crawlSchedule
```

The response is as follows:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/config/crawlSchedule</id>
  <updated>2008-12-11T06:29:35.862Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/crawlSchedule' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/crawlSchedule' />
  <gsa:content name='entryID'>crawlSchedule</gsa:content>
  <gsa:content name='isScheduledCrawl'>0</gsa:content>
</entry>
```

Updating a Crawl Schedule

To update the crawl schedule, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/config/crawlSchedule
```

The following example changes the crawl schedule:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='entryID'>crawlSchedule</gsa:content>
  <gsa:content name='isScheduledCrawl'>1</gsa:content>
  <gsa:content name='crawlSchedule'>0,0300,360 2,0000,1200</gsa:content>
</entry>
```

The following example changes crawl mode to continuous crawl:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='entryID'>crawlSchedule</gsa:content>
  <gsa:content name='isScheduledCrawl'>0</gsa:content>
</entry>
```

Crawler Access Rules

Create, retrieve, update, and delete crawler access rules on a search appliance.

Crawler access rules instruct the crawler how to authenticate when crawling protected content, as shown in the following list of properties:

Property	Description
domain	Windows domain (for NTLM) or empty (for HTTP Basic authorization)
isPublic	Indicates whether users can get results on both the public content (normally available to everyone) and the secure (confidential) content. The value can be 1 or 0. For the search appliance, crawler access can let the search appliance index secure content. If <code>isPublic</code> is 1, then the content can be searched by anyone. If <code>isPublic</code> is 0, then content can only be searched by users who can access the secure content.
order	The entries in crawler access rules are sequential rules. The order indicates the sequence. The order is an integer value starting from 1.
password	Password for authentication.
urlPattern	URL pattern that matches files with secure content.
username	User name for authentication.

Inserting a Crawler Access Rule

To insert a new crawl access rule, send an authenticated `POST` request to the following URL:

```
http://Search_Appliance:8000/feeds/crawlAccessNTLM
```

The following example inserts a new crawler access rule:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='entryID'>
    #URL pattern for the new crawler access rule
  </gsa:content>
  <gsa:content name='domain'>domainone</gsa:content>
  <gsa:content name='isPublic'>1</gsa:content>
  <gsa:content name='username'>username</gsa:content>
  <gsa:content name='password'>password</gsa:content>
</entry>
```

Retrieving Crawler Access Rules

To retrieve a list of crawl access rules, send an authenticated `GET` request to the following URL:

```
http://Search_Appliance:8000/feeds/crawlAccessNTLM
```

The following example shows a sample result:

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
  xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/crawlAccessNTLM</id>
  <updated>2009-03-22T06:33:40.471Z</updated>
  <link rel='http://schemas.google.com/g/2005#feed'
    type='application/atom+xml' href='http://gsa:8000/feeds/crawlAccessNTLM' />
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/crawlAccessNTLM' />
  <generator version='0.5' uri='http://gsa:8000/gsa'>
    Google Search Appliance
  </generator>
  <openSearch:startIndex>1</openSearch:startIndex>

  <entry>
    <id>http://gsa:8000/feeds/crawlAccessNTLM/http://example.com/</id>
    <updated>2009-03-22T06:33:40.471Z</updated>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/crawlAccessNTLM' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/crawlAccessNTLM' />
    <gsa:content name='entryID'>http://example.com/</gsa:content>
    <gsa:content name='urlPattern'>http://example.com/</gsa:content>
    <gsa:content name='username'>userone</gsa:content>
    <gsa:content name='order'>1</gsa:content>
    <gsa:content name='domain'>domainone</gsa:content>
    <gsa:content name='isPublic'>0</gsa:content>
  </entry>

  <entry>
    <id>http://gsa:8000/feeds/crawlAccessNTLM/http://example2.com/</id>
    <updated>2009-03-22T06:33:40.471Z</updated>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/crawlAccessNTLM' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/crawlAccessNTLM' />
    <gsa:content name='entryID'>http://example2.com/</gsa:content>
    <gsa:content name='urlPattern'>http://example2.com/</gsa:content>
    <gsa:content name='username'>usertwo</gsa:content>
    <gsa:content name='order'>2</gsa:content>
    <gsa:content name='domain'></gsa:content>
    <gsa:content name='isPublic'>1</gsa:content>
  </entry>
</feed>
```

To retrieve an individual crawler access rule, send an authenticated GET request to the following URL:

```
http://Search_Appliance:8000/feeds/crawlAccessNTLM/urlPattern
```

The following example shows a sample result:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/crawlAccessNTLM/http%3A%2F%2Fexample.com%2F</id>
  <updated>2009-03-23T10:19:55.045Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/crawlAccessNTLM/http%3A%2F%2Fexample.com%2F' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/crawlAccessNTLM/http%3A%2F%2Fexample.com%2F' />
  <gsa:content name='entryID'>http://example.com/</gsa:content>
  <gsa:content name='urlPattern'>http://example.com/</gsa:content>
  <gsa:content name='username'>userone</gsa:content>
  <gsa:content name='order'>1</gsa:content>
  <gsa:content name='domain'>domainone</gsa:content>
  <gsa:content name='isPublic'>0</gsa:content>
</entry>
```

Note: The `password` property is not available when retrieving crawler access rules.

Updating a Crawler Access Rule

To update a crawl access rule, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/crawlAccessNTLM/urlPattern
```

The following example request body shows the result:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='urlPattern'>#new URL pattern</gsa:content>
  <gsa:content name='domain'>newdomain</gsa:content>
  <gsa:content name='isPublic'>0</gsa:content>
  <gsa:content name='order'>2</gsa:content>
  <gsa:content name='username'>newuser</gsa:content>
  <gsa:content name='password'>newpass</gsa:content>
</entry>
```

Deleting a Crawler Access Rule

To delete a crawl access rule, send an authenticated `DELETE` request to the following URL:

```
http://Search_Appliance:8000/feeds/crawlAccessNTLM/urlPattern
```

Host Load Schedule

Retrieve and update the host load schedule for a search appliance using the `hostLoad` entry of the `config feed`.

Property	Description
<code>defaultHostLoad</code>	The default web server host load, a float value.
<code>exceptionHostLoad</code>	Exceptions to the default web server host load. This property consists of one or more lines of text in the following format: <code>hostName startTime endTime loadFactor</code> Where: <ul style="list-style-type: none"><code>hostName</code> is a URL or asterisk (*) to represent all hosts. If a <code>hostName</code> contains multiple load data values, separate the host name into multiple lines with each line containing one load value. The values cannot overlap.<code>startTime</code> and <code>endTime</code> are integer values between 0 and 23 and represent when to start and end crawling.<code>loadFactor</code> is a float value from 0 to 4 that represents the processing load on a search appliance, where 0 is unloaded and 4 is overloaded.
<code>maxURLs</code>	Maximum number of URLs to crawl, an integer value.

Retrieving a Host Load Schedule

To get the host load schedule information for a search appliance, send an authenticated `GET` request to the `config feed` URL:

```
http://Search_Appliance:8000/feeds/config/hostLoad
```

The result is an entry that contains the current host load schedule values for the search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/config/hostLoad</id>
  <updated>2008-12-15T13:28:00.931Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/hostLoad' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/hostLoad' />
  <gsa:content name='entryID'>hostLoad</gsa:content>
  <gsa:content name='defaultHostLoad'>3.6</gsa:content>
  <gsa:content name='exceptionHostLoad'>www.example.com 1 2 2.3</gsa:content>
  <gsa:content name='maxURLs'>2000</gsa:content>
</entry>
```

Updating a Host Load Schedule

To update the host load schedule information for a search appliance, send an authenticated `PUT` request to the `config` feed URL:

```
http://Search_Appliance:8000/feeds/config/hostLoad
```

The following example overwrites a host load schedule:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/config/hostLoad</id>
  <gsa:content name='entryID'>hostLoad</gsa:content>
  <gsa:content name='defaultHostLoad'>2.4</gsa:content>
  <gsa:content name='exceptionHostLoad'>
    * 3 5 1.2 www.example.com 1 6 3.6
  </gsa:content>
  <gsa:content name='maxURLs'>3000</gsa:content>
</entry>
```

Freshness Tuning

Increase or decrease how often a search appliance crawls a URL pattern using the `freshness` entry to the `config` feed.

Property	Description
<code>archiveURLs</code>	URL patterns for pages that contain archival or rarely changing content.
<code>forceURLs</code>	URL patterns for pages to recrawl regardless of their response to <code>If-Modified-Since</code> request headers.
<code>frequentURLs</code>	URL patterns for pages on which content changes often (typically more than once a day).

Retrieving Freshness Tuning Settings

To get the settings for freshness tuning, send an authenticated `GET` request to the following URL:

```
http://Search_Appliance:8000/feeds/config/freshness
```

The response is as follows:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/config/freshness</id>
  <updated>2008-12-11T07:16:26.220Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/freshness' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/freshness' />
  <gsa:content name='entryID'>freshness</gsa:content>
  <gsa:content name='archiveURLs'>http://good/</gsa:content>
  <gsa:content name='frequentURLs'>http://frequent/</gsa:content>
  <gsa:content name='forceURLs'>http://force/</gsa:content>
</entry>
```

Updating Freshness Tuning Settings

To update the settings for freshness tuning, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/config/freshness
```

The following is an example of a request body:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='entryID'>freshness</gsa:content>
  <gsa:content name='archiveURLs'>http://good/</gsa:content>
  <gsa:content name='frequentURLs'>http://frequent/</gsa:content>
  <gsa:content name='forceURLs'>http://force/</gsa:content>
</entry>
```

Recrawl URL Patterns

Recrawl URL patterns using the `recrawlNow` entry to the `command` feed.

If you discover a set of URLs that you want crawled (usually because changes made to the web pages or because of a temporary error or misconfiguration present when the crawler last tried to crawl the URL), you can enter the pattern to inject it quickly into the queue of URLs the search appliance is crawling.

Property	Description
<code>recrawlURLs</code>	URL patterns to be recrawled.

Recrawling URL Patterns

To recrawl URL patterns, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/command/recrawlNow
```

The following is an example of a request body:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='entryID'>recrawlNow</gsa:content>
  <gsa:content name='recrawlURLs'>http://recrawl/page.html</gsa:content>
</entry>
```

The following is an example of a request body with multiple recrawl URLs:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='entryID'>recrawlNow</gsa:content>
  <gsa:content name='recrawlURLs'>http://recrawl/page1.html
    http://recrawl/page2.html
    http://recrawl/page3.html
  </gsa:content>
</entry>
```


Connector Managers

Insert, retrieve, update, and delete connector managers on a search appliance.

Property	Description
description	A description of the connector manager.
status	The status of the connection between a Google Search Appliance and the connector manager deployed on an application server. The value can be <code>Connected</code> or <code>Disconnected</code> . The <code>Disconnected</code> mode can occur if the application server is down or there are problems on the network.
url	The URL of the application server where the connector manager is installed.

Inserting a Connector Manager

To insert a new connector manager, send an authenticated `POST` request to the following URL:

```
http://Search_Appliance:8000/feeds/connectorManager
```

The following example inserts a new connector manager:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='entryID'>ConnectorManagerOne</gsa:content>
  <gsa:content name='description'>Connector Manager One Description</gsa:content>
  <gsa:content name='url'>http://example.com:port</gsa:content>
</entry>
```

Retrieving Connector Managers

To retrieve a list of connector managers, send an authenticated `GET` request to the following URL:

```
http://Search_Appliance:8000/feeds/connectorManager
```

The following example shows a sample result:

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
  xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/connectorManager</id>
  <updated>2009-03-22T06:31:15.357Z</updated>
  <link rel='http://schemas.google.com/g/2005#feed'
    type='application/atom+xml' href='http://gsa:8000/feeds/connectorManager' />
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/connectorManager' />
  <generator version='0.5' uri='http://gsa:8000/gsa'>
    Google Search Appliance
  </generator>
  <openSearch:startIndex>1</openSearch:startIndex>
```

```

<entry>
  <id>http://gsa:8000/feeds/connectorManager/ConnectorManagerOne</id>
  <updated>2009-03-22T06:31:15.357Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/connectorManager' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/connectorManager' />
  <gsa:content name='entryID'>ConnectorManagerOne</gsa:content>
  <gsa:content name='status'>Disconnected</gsa:content>
  <gsa:content name='description'>
    Connector Manager One Description</gsa:content>
  <gsa:content name='url'>http://example.com:port/</gsa:content>
</entry>

<entry>
  <id>http://gsa:8000/feeds/connectorManager/ConnectorManagerTwo</id>
  <updated>2009-03-22T06:31:15.357Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/connectorManager' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/connectorManager' />
  <gsa:content name='entryID'>ConnectorManagerTwo</gsa:content>
  <gsa:content name='status'>Disconnected</gsa:content>
  <gsa:content name='description'>
    Connector Manager Two Description
  </gsa:content>
  <gsa:content name='url'>http://example2.com:port/</gsa:content>
</entry>
</feed>

```

To retrieve an individual connector manager, send an authenticated GET request to the following URL:

```
http://Search_Appliance:8000/feeds/connectorManager/ConnectorManager_Name
```

The following example shows a sample result:

```

<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/connectorManager/ConnectorManagerOne</id>
  <updated>2009-03-22T06:33:26.140Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/connectorManager/ConnectorManagerOne' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/connectorManager/ConnectorManagerOne' />
  <gsa:content name='entryID'>ConnectorManagerOne</gsa:content>
  <gsa:content name='status'>Disconnected</gsa:content>
  <gsa:content name='description'>Connector Manager One Description</gsa:content>
  <gsa:content name='url'>http://example.com:port/</gsa:content>
</entry>

```

Updating a Connector Manager

To update the description and url in a connector manager, send an authenticated PUT request to the following URL:

```
http://Search_Appliance:8000/feeds/connectorManager/ConnectorManager_Name
```

The following example request body shows the result:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='description'>new description</gsa:content>
  <gsa:content name='url'>#new URL</gsa:content>
</entry>
```

Deleting a Connector Manager

To delete a connector manager, send an authenticated `DELETE` request to the following URL:

```
http://Search_Appliance:8000/feeds/connectorManager/ConnectorManager_Name
```

OneBox Settings

Retrieve or update a OneBox setting for a search appliance using the `oneboxSetting` entry of the `config` feed.

Property	Description
<code>maxResults</code>	Maximum number of OneBox results per search.
<code>timeout</code>	OneBox response timeout.

Retrieving OneBox Settings

To get a OneBox setting for a search appliance, send an authenticated `GET` request to the `config` feed URL:

```
http://Search_Appliance:8000/feeds/config/oneboxSetting
```

The following example result is an entry that includes current OneBox setting values for the search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/config/oneboxSetting</id>
  <updated>2008-12-12T09:21:47.477Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/oneboxSetting' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/config/oneboxSetting' />
  <gsa:content name='entryID'>oneboxSetting</gsa:content>
  <gsa:content name='maxResults'>2</gsa:content>
  <gsa:content name='timeout'>1000</gsa:content>
</entry>
```

Updating OneBox Settings

To update the OneBox settings for a search appliance, send an authenticated `PUT` request to the `config` feed URL:

```
http://Search_Appliance:8000/feeds/config/oneboxSetting
```

The following example overwrites the OneBox setting specified in the entry to update:

```
<?xml version='1.0' encoding='UTF-8'?>
  <entry xmlns='http://www.w3.org/2005/Atom'
    xmlns:gsa='http://schemas.google.com/gsa/2007'>
    <id>http://gsa:8000/feeds/config/oneboxSetting</id>
    <gsa:content name='entryID'>oneboxSetting</gsa:content>
    <gsa:content name='maxResults'>3</gsa:content>
    <gsa:content name='timeout'>2000</gsa:content>
  </entry>
```

OneBox Modules

Retrieve the names of and delete OneBox modules from a search appliance using the `onebox` feed.

Note: This API does not support adding, updating, or viewing detailed configuration information for a OneBox module.

Property	Description
logContent	The log content for OneBox logs.

Retrieving OneBox Module Names

To get the OneBox information for a search appliance, send an authenticated `GET` request to the `onebox` feed URL:

```
http://Search_Appliance:8000/feeds/onebox
```

The following example retrieves the current OneBox values for the search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
  xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/onebox</id>
  <updated>2008-12-15T13:37:36.678Z</updated>
  <link rel='http://schemas.example.com/g/2005#feed'
    type='application/atom+xml' href='http://gsa:8000/feeds/onebox' />
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/onebox' />
  <generator version='0.5' uri='http://gsa:8000/gsa'>
    Google Search Appliance
  </generator>
  <openSearch:startIndex>1</openSearch:startIndex>
```

```

<entry>
  <id>http://gsa:8000/feeds/onebox/oneboxone</id>
  <updated>2008-12-15T13:37:36.678Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/onebox' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/onebox' />
  <gsa:content name='entryID'>oneboxone</gsa:content>
</entry>

<entry>
  <id>http://gsa:8000/feeds/onebox/oneboxtwo</id>
  <updated>2008-12-15T13:37:36.678Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/onebox' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/onebox' />
  <gsa:content name='entryID'>oneboxtwo</gsa:content>
</entry>
</feed>

```

Note: Because this API does not support retrieving detailed OneBox configuration information, retrieving the `onebox` feed supplies only the names of each OneBox module.

To view OneBox information for a search appliance, send an authenticated `GET` request to the `onebox` feed URL for a OneBox name:

```
http://Search_Appliance:8000/feeds/onebox/OneBox_Name
```

The result is an entry that includes current individual OneBox values for a search appliance:

```

<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/onebox/oneboxone</id>
  <updated>2008-12-15T13:39:42.895Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/onebox/oneboxone' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/onebox/oneboxone' />
  <gsa:content name='entryID'>oneboxone</gsa:content>
  <gsa:content name='logContent'>onebox logs</gsa:content>
</entry>

```

Note: The logs for each OneBox can only be retrieved by getting separate information for each OneBox.

Deleting a OneBox Module

To delete a OneBox module from a search appliance, send an authenticated `DELETE` request to the `onebox` feed URL:

```
http://Search_Appliance:8000/feeds/onebox/OneBox_Name
```

Crawl Status

Check the crawl status, and also pause or resume crawl using the `pauseCrawl` entry of the `command` feed.

Property	Description
<code>pauseCrawl</code>	<ul style="list-style-type: none">Set to 1 to check to see if crawl on a search appliance is paused. You can also use this property to pause the crawl.Set to 0 to verify that a search appliance is crawling. You can also use this property to start the crawl.

Retrieving the Crawl Status

To check status of crawl, send an authenticated `GET` request to the following URL:

```
http://Search_Appliance:8000/feeds/command/pauseCrawl
```

The response result is as follows:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/command/pauseCrawl</id>
  <updated>2008-12-11T08:55:57.824Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/command/pauseCrawl' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/command/pauseCrawl' />
  <gsa:content name='entryID'>pauseCrawl</gsa:content>
  <gsa:content name='pauseCrawl'>0</gsa:content>
</entry>
```

Pausing or Resuming Crawl

To pause or resume crawl, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/command/pauseCrawl
```

The following is an example of a request to resume crawl:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='entryID'>pauseCrawl</gsa:content>
  <gsa:content name='pauseCrawl'>0</gsa:content>
</entry>
```

Document Status

Retrieve the status of the documents that have been crawled and served using the `documentStatus` entry of the `status` feed. The properties for the document status are:

Property	Description
<code>crawledURLsToday</code>	The number of documents crawled since midnight. (Midnight pertains to the time that is set on the search appliance.)
<code>crawlPagePerSecond</code>	Current crawling rate measured in pages per second.
<code>errorURLsToday</code>	Document errors that occurred since midnight on the search appliance.
<code>filteredBytes</code>	Document bytes that have been filtered by domain, language, file type, or metadata.
<code>foundURLs</code>	The number of URLs found that match crawl patterns.
<code>servedURLs</code>	The number of total documents that have been served.

Retrieving Document Status

To retrieve document status, send an authenticated `GET` request to the following URL:

```
http://Search_Appliance:8000/feeds/status/documentStatus
```

The response result is as follows:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/stats/documentStatus</id>
  <updated>2008-12-11T08:38:05.048Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/status/documentStatus' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/status/documentStatus' />
  <gsa:content name='entryID'>documentStatus</gsa:content>
  <gsa:content name='servedURLs'>0</gsa:content>
  <gsa:content name='crawlPagePerSecond'>0</gsa:content>
  <gsa:content name='crawledURLsToday'>0</gsa:content>
  <gsa:content name='foundURLs'>1</gsa:content>
  <gsa:content name='filteredBytes'>0</gsa:content>
  <gsa:content name='errorURLsToday'>0</gsa:content>
</entry>
```

Index

The sections that follow describe how to configure the **Index** features of the Admin Console:

- “Collections” on page 32
- “Index Diagnostics” on page 34
- “Content Statistics” on page 40
- “Reset Index” on page 42

Collections

Create, retrieve, update, and delete collections on a search appliance.

A collection is a group of URL patterns that can be searched separately from other URL patterns.

Property	Description
collectionName	The name of a collection to create (only required when creating a new collection).
doNotCrawlURLs	The URL patterns to exclude from this collection.
followURLs	The URL patterns to include in this collection.
importData	The collection settings exported from the Admin Console. Only required when creating a new collection by the <code>import</code> method.
insertMethod	The method of creating (only required when creating a new collection). Possible values: <code>default</code> , <code>customize</code> , and <code>import</code> .

Creating a Collection

To create a new collection, send an authenticated `POST` request to the following URL:

```
http://Search_Appliance:8000/feeds/collection
```

To create a new collection with a default setting, use the following entry:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='collectionName'>new_collection</gsa:content>
  <gsa:content name='insertMethod'>default</gsa:content>
</entry>
```

To specify the settings for a new collection, send the following entry:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='collectionName'>new_collection</gsa:content>
  <gsa:content name='insertMethod'>customize</gsa:content>
  <gsa:content name='followURLs'>#url in new collection</gsa:content>
  <gsa:content name='doNotCrawlURLs'># url not in new collection</gsa:content>
</entry>
```

Retrieving All Collections

To retrieve a list of collections, send an authenticated `GET` request to the following URL:

```
http://Search_Appliance:8000/feeds/collection
```


The following example shows a sample result:

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
  xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/collection</id>
  <updated>2008-12-11T08:01:21.253Z</updated>
  <link rel='http://schemas.example.com/g/2005#feed'
    type='application/atom+xml'
    href='http://gsa:8000/feeds/collection' />
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/collection' />
  <generator version='0.5' uri='http://gsa:8000/gsa'>
    Google Search Appliance</generator>
  <openSearch:startIndex>1</openSearch:startIndex>

  <entry>
    <id>http://gsa:8000/feeds/collection/default_collection</id>
    <updated>2008-12-11T08:01:21.253Z</updated>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/collection' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/collection' />
    <gsa:content name='entryID'>default_collection</gsa:content>
    <gsa:content name='followURLs'>/</gsa:content>
    <gsa:content name='doNotCrawlURLs'></gsa:content>
  </entry>

  <entry>
    <id>http://gsa:8000/feeds/collection/new2_collection</id>
    <updated>2008-12-11T08:01:21.253Z</updated>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/collection' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/collection' />
    <gsa:content name='entryID'>new_collection</gsa:content>
    <gsa:content name='followURLs'>#urls in new collection</gsa:content>
    <gsa:content name='doNotCrawlURLs'></gsa:content>
  </entry>
</feed>
```

Retrieving a Collection

To retrieve an attribute in a single collection, send an authenticated GET request to the following URL:

```
http://Search_Appliance:8000/feeds/collection/Collection_Name
```

The following example response shows the result:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/collection/default_collection</id>
  <updated>2008-12-11T08:18:04.372Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/collection/default_collection'/>
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/collection/default_collection'/>
  <gsa:content name='entryID'>default_collection</gsa:content>
  <gsa:content name='followURLs'></gsa:content>
  <gsa:content name='doNotCrawlURLs'></gsa:content>
</entry>
```

Updating a Collection

To update an attribute in a collection, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/collection/Collection_Name
```

The following example request body shows the result:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='followURLs'>#updated urls</gsa:content>
  <gsa:content name='doNotCrawlURLs'></gsa:content>
</entry>
```

Deleting a Collection

To delete a collection, send an authenticated `DELETE` request to the following URL:

```
http://Search_Appliance:8000/feeds/collection/Collection_Name
```

Index Diagnostics

List crawled documents and retrieve the status of documents in a search appliance using the `diagnostics` feed.

Document Status Values

The following tables list document status values.

Note: Use `all` to indicate any status value.

Successful Crawl:

Value	Description
1	Crawled from remote server
2	Crawled from cache

Crawl Errors:

Value	Description
7	Redirect with no location header
11	Document not found (404)
12	Other HTTP 400 Errors
14	HTTP 0 error
15	Permanent DNS failure
16	Empty document
17	Image conversion failed
22	Authentication failed
25	Conversion error
32	HTTP 500 error
33	Robots.txt unreachable
35	Temporary DNS failure
36	Connection failed
37	Connection timeout
38	Connection closed
40	Connection refused
41	Connection reset
43	No route to host
50	Other error

Crawl Exclusions:

Value	Description
3	Not in URLs to crawl
4	In URLs not to crawl
5	Off domain redirect
6	Long redirect chain
8	Infinite URL space
9	Unhandled protocol
10	URL too long
13	Robots no-index
18	Rejected by rewrite rules
19	Unknown extension
20	Disallowed by a meta tag
24	Disallowed by robots

Value	Description
26	Unhandled content type
27	No filter for content type
34	Robots.txt forbidden

Listing Crawled Documents

Query parameters:

Parameter	Description
collectionName	Name of the collection that you want to list. The default value is the last used collection.
flatList	<code>false</code> : List the files and directories that directly belong to an indicated URI. <code>true</code> : List all files starting with an indicated URI as a flat list. The default value is <code>false</code> .
negativeState	<code>false</code> : Just return documents with a status that is equal to <code>view</code> . <code>true</code> : Just return documents with a status that is not equal to <code>view</code> . The default value is <code>false</code> .
pageNum	The page you want to view. The files from a URI may be separated into several pages to return. The page number starts from 1. The default value is 1, the first page.
sort	The key field of sorting. <code>host</code> : sort by host name, <code>file</code> : sort by file name, <code>crawled</code> : sort by crawled doc number, <code>errors</code> sort by errors number, <code>excluded</code> sort by excluded doc number. The default value is "".
uriAt	The prefix of the URI of the documents that you want to list. If not blank, it must contain at least <code>http://hostname.domain.com/</code> . The default value is "".
view	A filter of the document status. The values of <code>view</code> are described in the section "Document Status Values" on page 34. The default value is <code>all</code> .

To list documents, send an authenticated `GET` request to root entry of diagnostics feed.

```
http://Search_Appliance:8000/feeds/
diagnostics?uriAt=http%3A%2F%2Fserver.com%2Fsecured%2Ftest1
```

Returns a `description` entry, a set of documents status entries and a set of directories status entries.

Description entry properties:

Property	Description
<Entry Name>	<code>description</code>
numPages	The total number of pages to return.
uriAt	The prefix of the URL taken from the query parameters.

Directory status entry properties:

Property	Description
<Entry Name>	The URL of a directory.
numCrawledURLs	The number of crawled documents in a directory.
numExcludedURLs	The number of excluded URL patterns in a directory.
numRetrievalErrors	The number of retrieval error for documents in a directory.
type	DirectoryContentData OR HostContentData.

Document status entry properties:

Property	Description
<Entry Name>	The URL pattern of a document to check its status.
docState	The status of a document. The values of docState are described in "Document Status Values" on page 34.
isCookieServerError	Indicates if the cookie server encountered an error.
timeStamp	The last time that the search appliance indexed a document.
type	FileContentData

Example:

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
  xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/diagnostics</id>
  <updated>2009-03-26T04:47:40.814Z</updated>
  <link rel='http://schemas.google.com/g/2005#feed'
    type='application/atom+xml' href='http://gsa:8000/feeds/diagnostics' />
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/
    diagnostics?uriAt=http%3A%2F%2Fserver.com%2Fsecured%2Ftest1%2F' />
  <generator version='0.5' uri='http://gsa:8000/gsa'>
    Google Search Appliance
  </generator>
  <openSearch:startIndex>1</openSearch:startIndex>
```

```

<entry>
  <id>http://gsa:8000/feeds/diagnostics/http://server.com/secured/test1/
    level_1_0</id>
  <updated>2009-03-26T04:47:40.813Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T04:47:40.813Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/diagnostics' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/diagnostics' />
  <gsa:content name='entryID'>
    http://server.com/secured/test1/level_1_0
  </gsa:content>
  <gsa:content name='numCrawledURLs'>217</gsa:content>
  <gsa:content name='numExcludedURLs'>0</gsa:content>
  <gsa:content name='type'>DirectoryContentData</gsa:content>
  <gsa:content name='numRetrievalErrors'>0</gsa:content>
</entry>

<entry>
  <id>http://gsa:8000/feeds/diagnostics/http://server.com/secured/test1/
    doc_0_0.html</id>
  <updated>2009-03-26T04:47:40.814Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T04:47:40.814Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/diagnostics' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/diagnostics' />
  <gsa:content name='entryID'>
    http://server.com/secured/test1/doc_0_0.html
  </gsa:content>
  <gsa:content name='isCookieServerError'>0</gsa:content>
  <gsa:content name='timeStamp'>1238042696</gsa:content>
  <gsa:content name='docState'>2</gsa:content>
  <gsa:content name='type'>FileContentData</gsa:content>
</entry>

<entry>
  <id>http://gsa:8000/feeds/diagnostics/description</id>
  <updated>2009-03-26T04:47:40.814Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T04:47:40.814Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/diagnostics' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/diagnostics' />
  <gsa:content name='entryID'>description</gsa:content>
  <gsa:content name='numPages'>1</gsa:content>
  <gsa:content name='uriAt'>http://server.com/secured/test1/</gsa:content>
</entry>
</feed>

```

Getting Crawled Document Status

Get the status for documents that have been crawled for a collection.

Parameter	Description
collectionName	Name of the collection for which you want to list the document status. The default value is the last used collection.

To retrieve detailed information for a document, send an authenticated `GET` request to a document entry of the `diagnostics` feed.

```
http://Search_Appliance:8000/feeds/diagnostics/  
http%3A%2F%2Fserver.com%2Fsecured%2Ftest1%2Fdoc_0_2.html
```

A detailed document status entry is returned with the following properties.

Property	Description
<Entry Name>	The URL of a document.
backwardLinks	The number of backward links for the document.
collectionList	The list of collections that contain the document.
contentSize	The size of the document content.
contentType	The type of the document.
crawlFrequency	The frequency at which the document is being scheduled to crawl, with possible values of <code>seldom</code> , <code>normal</code> , and <code>frequent</code> .
crawlHistory	A multi-line history of the document crawl including the timestamp when the document was crawled, the document status code and description in the following format: <pre>timestamp status_code status_description timestamp status_code status_description</pre> For status code values, see “Document Status Values” on page 34.
currentlyInflight	If the document is currently in process.
date	The date that the document was indexed.
forwardLinks	The number of forward links for the document.
isCached	If a cached page for the document is indexed.
lastModifiedDate	The last modified date of the document.
latestOnDisk	The timestamp of the version being served.

```

<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
<id>http://gsa:8000/feeds/diagnostics/http%3A%2F%2Fexample.com%2Fdoc.html</id>
<updated>2009-03-26T05:41:43.724Z</updated>
<app:edited xmlns:app='http://purl.org/atom/app#'>
  2009-03-26T05:41:43.724Z
</app:edited>
<link rel='self' type='application/atom+xml'
  href='http://gsa:8000/feeds/diagnostics/http%3A%2F%2Fexample.com%2Fdoc.html' />
<link rel='edit' type='application/atom+xml'
  href='http://gsa:8000/feeds/diagnostics/http%3A%2F%2Fexample.com%2Fdoc.html' />
<gsa:content name='entryID'>http://example.com/doc.html</gsa:content>
<gsa:content name='backwardLinks'>0</gsa:content>
<gsa:content name='forwardLinks'>0</gsa:content>
<gsa:content name='isCached'>1</gsa:content>
<gsa:content name='lastModifiedDate'>-1</gsa:content>
<gsa:content name='collectionList'>Default,default_collection</gsa:content>
<gsa:content name='date'>-1</gsa:content>
<gsa:content name='currentlyInFlight'>0</gsa:content>
<gsa:content name='contentSize'>641</gsa:content>
<gsa:content name='contentType'>text/html</gsa:content>
<gsa:content name='crawlFrequency'>normal</gsa:content>
<gsa:content name='crawlHistory'>
1245977534      2      Unchanged.
1245955634      1      Crawled: New Document
1245951054      2      Unchanged.
</gsa:content>
<gsa:content name='latestOnDisk'>1245977534</gsa:content>
</entry>

```

Content Statistics

Get content statistics for each kind of documents using the `contentStatistics` feed.

Common query parameters for all requests:

Parameter	Description
<code>collectionName</code>	Name of the collection which you want to list. The default value is the last used collection.

Content statistics entry properties:

Property	Description
<code><Entry Name></code>	The content type of documents, such as <code>plain/text</code> .
<code>avgSize</code>	The average document size of this content type.
<code>maxSize</code>	The maximal document size of this content type.
<code>minSize</code>	The minimal document size of this content type.
<code>numFiles</code>	The file number of this content type.
<code>totalSize</code>	The total document size of this content type.

Retrieving Content Statistics for All Document Types

To retrieve content statistics for all kinds of document in a search appliance, send an authenticated GET request to the root entry of the `contentStatistics` feed.

```
http://Search_Appliance:8000/feeds/contentStatistics
```

A list of content statistics entries is returned.

```
<?xml version='1.0' encoding='UTF-8'?>
  <feed xmlns='http://www.w3.org/2005/Atom'
        xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
        xmlns:gsa='http://schemas.google.com/gsa/2007'>
    <id>http://gsa:8000/feeds/contentStatistics</id>
    <updated>2009-03-26T05:45:33.701Z</updated>
    <link rel='http://schemas.google.com/g/2005#feed'
          type='application/atom+xml'
          href='http://gsa:8000/feeds/contentStatistics' />
    <link rel='self' type='application/atom+xml'
          href='http://gsa:8000/feeds/contentStatistics' />
    <generator version='0.5' uri='http://gsa:8000/gsa'>
      Google Search Appliance
    </generator>
    <openSearch:startIndex>1</openSearch:startIndex>

    <entry>
      <id>http://gsa:8000/feeds/contentStatistics/text/html</id>
      <updated>2009-03-26T05:45:33.701Z</updated>
      <app:edited xmlns:app='http://purl.org/atom/app#'>
        2009-03-26T05:45:33.701Z
      </app:edited>
      <link rel='self' type='application/atom+xml'
            href='http://gsa:8000/feeds/contentStatistics' />
      <link rel='edit' type='application/atom+xml'
            href='http://gsa:8000/feeds/contentStatistics' />
      <gsa:content name='entryID'>text/html</gsa:content>
      <gsa:content name='numFiles'>1,037</gsa:content>
      <gsa:content name='minSize'>606</gsa:content>
      <gsa:content name='avgSize'>2.5k</gsa:content>
      <gsa:content name='totalSize'>2.5M</gsa:content>
      <gsa:content name='maxSize'>38k</gsa:content>
    </entry>

    <entry>
      <id>http://gsa:8000/feeds/contentStatistics/text/pdf</id>
      <updated>2009-03-26T05:45:33.701Z</updated>
      <app:edited xmlns:app='http://purl.org/atom/app#'>
        2009-03-26T05:45:33.701Z
      </app:edited>
      <link rel='self' type='application/atom+xml'
            href='http://gsa:8000/feeds/contentStatistics' />
      <link rel='edit' type='application/atom+xml'
            href='http://gsa:8000/feeds/contentStatistics' />
      <gsa:content name='entryID'>text/pdf</gsa:content>
      <gsa:content name='numFiles'>3</gsa:content>
      <gsa:content name='minSize'>24k</gsa:content>
      <gsa:content name='avgSize'>136k</gsa:content>
      <gsa:content name='totalSize'>407k</gsa:content>
      <gsa:content name='maxSize'>217k</gsa:content>
    </entry>
```

Retrieving Content Statistics for a Document Type

To retrieve content statistics for a document type in a search appliance, send an authenticated GET request to the content statistics entry of the `contentStatistics` feed.

```
http://Search_Appliance:8000/feeds/contentStatistics/text%2Fpdf
```

A content statistics entry is returned.

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/contentStatistics/text%2Fpdf</id>
  <updated>2009-03-26T05:51:32.659Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T05:51:32.659Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/contentStatistics/text%2Fpdf' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/contentStatistics/text%2Fpdf' />
  <gsa:content name='entryID'>text/pdf</gsa:content>
  <gsa:content name='numFiles'>3</gsa:content>
  <gsa:content name='minSize'>24k</gsa:content>
  <gsa:content name='avgSize'>136k</gsa:content>
  <gsa:content name='totalSize'>407k</gsa:content>
  <gsa:content name='maxSize'>217k</gsa:content>
</entry>
```

Reset Index

Reset your crawling queues and delete your search index, removing all its contents.

Note: If you reset an index that has a large document corpus, recrawling the index can take many days to complete.

Property	Description
<code>resetIndex</code>	Set to 1 to reset the index or 0 to not reset the index. If viewing, 1 indicates that the index was reset, 0 indicates that the index was not reset.
<code>resetStatusCode</code>	Status code for resetting the index.
<code>resetStatusMessage</code>	Status message. Possible values are <code>ERROR</code> , <code>PROGRESS</code> , or <code>READY</code> .

Retrieving Status After Resetting the Index

To check the status of resetting the index, send an authenticated GET request to the following URL:

```
http://Search_Appliance:8000/feeds/command/resetIndex
```

An example response result is as follows:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/command/resetIndex</id>
  <updated>2008-12-11T09:00:21.907Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/command/resetIndex' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/command/resetIndex' />
  <gsa:content name='entryID'>resetIndex</gsa:content>
  <gsa:content name='resetStatusCode'>2</gsa:content>
  <gsa:content name='resetIndex'>1</gsa:content>
  <gsa:content name='resetStatusMessage'>PROGRESS</gsa:content>
</entry>
```

Resetting the Index

To reset the index, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/command/resetIndex
```

The following is an example of resetting the index:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='resetIndex'>1</gsa:content>
</entry>
```

Search

The sections that follow describe how to configure the **Search** features of the Admin Console:

- “Front Ends, Remove URLs, and Relative OneBoxes” on page 44
- “Output Format XSLT Stylesheet” on page 46
- “KeyMatch” on page 48
- “Related Queries” on page 51
- “Query Suggestion” on page 54
- “Search Status” on page 55

Front Ends, Remove URLs, and Relative OneBoxes

Retrieve, update, and delete front ends, remove URLs, and relative OneBox modules for a search appliance using the `frontend` feed. A relative OneBox is a OneBox module that you assign to work with a front end. Remove URLs are URL patterns that you want to exclude from appearing in an index for a front end.

Property	Description
<code>frontendOnebox</code>	OneBox modules for a front end. Specify a comma-separated list of OneBox module names. The OneBox names display in alphabetic order.
<code>removeUrls</code>	Remove URLs for a front end.

Retrieving Front Ends, Remove URLs, and Relative OneBoxes

To get front end information for a search appliance, send an authenticated `GET` request to the `frontend` feed URL:

```
http://Search_Appliance:8000/feeds/frontend
```

The following result is a feed that includes current front ends values for a search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
  xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/frontend</id>
  <updated>2008-12-15T14:48:14.851Z</updated>
  <link rel='http://schemas.example.com/g/2005#feed' type='application/atom+xml'
    href='http://gsa:8000/feeds/frontend' />
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/frontend' />
  <generator version='0.5' uri='http://gsa:8000/gsa'>
    Google Search Appliance
  </generator>
  <openSearch:startIndex>1</openSearch:startIndex>

  <entry>
    <id>http://gsa:8000/feeds/frontend/default_frontend</id>
    <updated>2008-12-15T14:48:14.851Z</updated>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/frontend' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/frontend' />
    <gsa:content name='entryID'>default_frontend</gsa:content>
    <gsa:content name='frontendOnebox'>oneboxone,oneboxtwo</gsa:content>
    <gsa:content name='removeUrls'>http://www.example.com/</gsa:content>
  </entry>
</feed>
```

To get the individual front end information for a search appliance, send an authenticated `GET` request to the `frontend` feed URL for the front end name:

```
http://Search_Appliance:8000/feeds/frontend/Front_End
```

The following result is an entry that includes current individual front end values for a search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/frontend/default_frontend</id>
  <updated>2008-12-15T16:21:26.012Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/frontend/default_frontend' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/frontend/default_frontend' />
  <gsa:content name='entryID'>default_frontend</gsa:content>
  <gsa:content name='frontendOnebox'>oneboxone,oneboxtwo</gsa:content>
  <gsa:content name='removeUrls'>http://www.example.com/</gsa:content>
</entry>
```

Updating Remove URLs and Relative OneBoxes

To update the remove URLs and relative OneBoxes that are associated with a front end for a search appliance, send an authenticated `PUT` request to the `frontend` feed URL:

```
http://Search_Appliance:8000/feeds/frontend/Front_End
```

The following example updates the values for remove URLs and relative OneBox modules for a front end:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/frontend/default_frontend</id>
  <gsa:content name='entryID'>default_frontend</gsa:content>
  <gsa:content name='frontendOnebox'>oneboxtwo</gsa:content>
  <gsa:content name='removeUrls'>http://www.example2.com/</gsa:content>
</entry>
```

Inserting Remove URLs and Relative OneBoxes

To insert a front end and remove URLs for a search appliance, send an authenticated `POST` request to the `frontend` feed URL:

```
http://Search_Appliance:8000/feeds/frontend
```

The following example specifies a URL pattern to remove from an index for the `frontend_one` front end:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/frontend/frontend_one</id>
  <gsa:content name='entryID'>frontend_one</gsa:content>
  <gsa:content name='removeUrls'>http://www.example3.com/</gsa:content>
</entry>
```

Note: When inserting a new front end, the `frontendOnebox` property is not supported.

Deleting a Front End

To delete a front end from a search appliance, send an authenticated `DELETE` request to the `frontend` feed URL:

```
http://Search_Appliance:8000/feeds/frontend
```

Output Format XSLT Stylesheet

Retrieve and update XSLT template and other output format related properties for each language of each front end using the `frontend` entry of the `outputFormat` feed.

Parameter	Description
<code>language</code>	<p>Specify a language for the output format properties that you want to retrieve. Each front end can contain multiple languages, and each language has its own output format properties. Each front end + language can have its own XSLT stylesheet. The <code>language</code> parameter enables you to retrieve and update a stylesheet for a front end associated with a language.</p> <p>Administrators who use the Admin Console set the language in their browser and the Admin Console then displays in that language (if the Admin Console has been translated into that language). Hence the <code>language</code> parameter for the <code>outputFormat</code> feed is limited to the values to which the Admin Console is translated.</p>

Use the following properties to retrieve an output format stylesheet.

Property	Description
<code>isDefaultLanguage</code>	1 if the designated language is the default language for the specified front end, 0 if not.
<code>isStyleSheetEdited</code>	0 if the style sheet has default values, 1 if the style sheet has been edited.
<code>language</code>	In a retrieving operation, <code>language</code> is determined by the <code>language</code> specified by <code>query</code> parameter. In an updating operation, <code>language</code> is passed as an entry property to specify the language of the output stylesheet.
<code>restoreDefaultFormat</code>	1 if you want to restore a custom-edited XSLT stylesheet to contain default values, a 0 value has no effect.
<code>stylesheetContent</code>	The output format of the XSLT code.

Note: For an update action, the `restoreDefaultFormat` content is mutually exclusive from the `stylesheetContent`. For each update action, you can restore the output format style sheet XSLT back to its original default values, or set the style sheet XSLT to a custom format, or neither, but not both.

Retrieving the Output Format XSLT Stylesheet

To get the output format stylesheet information for a search appliance, send an authenticated GET request to the `outputFormat` feed URL:

```
http://Search_Appliance:8000/feeds/outputFormat/Front_End?language=Language_Code
```

The result is an entry that includes all stylesheet information for the designated `Front_End` and `Language_Code`:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/outputFormat/default_frontend</id>
  <updated>2008-12-09T23:59:51.078Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/outputFormat/default_frontend' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/outputFormat/default_frontend' />
  <gsa:content name='entryID'>default_frontend</gsa:content>
  <gsa:content name='isStyleSheetEdited'>0</gsa:content>
  <gsa:content name='stylesheetContent'>
  &lt;!-- *** START OF STYLESHEET *** --&gt;
  &lt;xsl:stylesheet xmlns:xsl='http://www.w3.org/1999/XSL/
Transform' version='1.0'&gt;
  &lt; ;&gt;
  &lt;xsl:include href='customer-onebox.xsl'&gt; /&gt;
  &lt;xsl:output method='html'&gt; /&gt;
  &lt;xsl:variable name='show_logo'&gt;1&lt;/xsl:variable&gt;
  &lt;xsl:variable name='logo_url'&gt;images/Title_Left.png&lt;/
xsl:variable&gt;
  &lt;xsl:variable name='logo_width'&gt;200&lt;/xsl:variable&gt;
  &lt;xsl:variable name='logo_height'&gt;78&lt;/
xsl:variable&gt;
  &lt;xsl:template match='@*|node()'&gt; /&gt;
  &lt;/xsl:stylesheet&gt;
  &lt;!-- *** END OF STYLESHEET *** --&gt;
  </gsa:content>
  <gsa:content name='isDefaultLanguage'>1</gsa:content>
  <gsa:content name='language'>en</gsa:content>
</entry>
```

Updating the Output Format XSLT Stylesheet

To update the output format stylesheet information for a search appliance, send an authenticated PUT request to the `outputFormat` feed URL:

```
http://Search_Appliance:8000/feeds/outputFormat/Front_End
```

Specify the `language` parameter in the `language` property of the entry to update.

This value overwrites the stylesheet properties specified in the entry to update for the designated *Front_End* and *Language_Code*:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/outputFormat/default_frontend</id>
  <gsa:content name='entryID'>default_frontend</gsa:content>
  <gsa:content name='language'>en</gsa:content>
  <gsa:content name='restoreDefaultFormat'>1</gsa:content>
  <gsa:content name='stylesheetContent'>
  &lt;!-- *** START OF STYLESHEET *** --&gt;
  &lt;!-- XML escaped XSLT code goes here --&gt;
  &lt;!-- *** END OF STYLESHEET *** --&gt;
  </gsa:content>
  <gsa:content name='isDefaultLanguage'>1</gsa:content>
</entry>
```

KeyMatch

Retrieve or update KeyMatch settings on a search appliance using the `keymatch` feed. KeyMatch lets you promote specific web pages on your site. The parameters for this feed are:

Parameter	Description
<code>query</code>	A query string to perform a full-text search. For example, if you specify <code>computer</code> in the <code>query</code> parameter, then you will get all KeyMatch settings that contain the word <code>computer</code> .
<code>startLine</code>	The starting line number of a result, the default value is 0 results.
<code>maxLines</code>	The number of result lines in a response, the default value is 50 lines of results.

The `keymatch` feed has the following properties:

Property	Description
<code>line_number</code>	The <code>line_number</code> of the KeyMatch configuration rule.
<code>newLines</code>	The KeyMatch settings to replace the existing values. You can specify multiple lines of KeyMatch values. The line delimiter is <code>\n</code> .
<code>numLines</code>	The total number of result lines.
<code>originalLines</code>	The original KeyMatch settings to change. You can include multiple lines of KeyMatch values. The line delimiter is <code>\n</code> .
<code>startLine</code>	The starting line number of the KeyMatch configuration to change. The minimum value is 0.
<code>updateMethod</code>	The method to change KeyMatch configurations. Possible values are: <ul style="list-style-type: none"><code>update</code>. Update part of the KeyMatch configuration table to the new configurations. You can also delete KeyMatch configurations using the <code>update</code> method, as shown in “Updating KeyMatch Settings” on page 50.<code>append</code>. Add a new KeyMatch configuration to the end of the KeyMatch configuration table.<code>replace</code>. Delete all rules in the KeyMatch configuration table and then append the new rules that you provide.

A KeyMatch configuration rule is in the following format:

Search_Terms, KeyMatch_Type, URL, Title

The `KeyMatch_Type` is one of the three values, `KeywordMatch`, `PhraseMatch`, and `ExactMatch`. The `Search_Terms` and `URL` fields cannot be empty. The KeyMatch configuration conforms to the CSV format, which uses a comma to separate values.

Retrieving KeyMatch Settings

To get KeyMatch settings, send an authenticated `GET` request to the following URL:

```
http://Search_Appliance:8000/feeds/keymatch/  
Front_End_Name?query=Search_String&startLine=Start_Line&maxLines=Max_Lines;
```

The following example retrieves KeyMatch settings—note that `gsa:content name="2"` (or 0 or 1) shows the use of the `line_number` property:

```
<?xml version="1.0" ?>
<entry xmlns="http://www.w3.org/2005/Atom"
  xmlns:gsa="http://schemas.google.com/gsa/2007">
  <id>http://ent1:8000/feeds/keymatch/default_frontend</id>
  <updated>2008-12-05T03:13:19.806Z</updated>
  <link href="http://ent1:8000/feeds/keymatch/default_frontend"
    rel="self" type="application/atom+xml"/>
  <link href="http://ent1:8000/feeds/keymatch/default_frontend"
    rel="edit" type="application/atom+xml"/>
  <gsa:content name="entryID">default_frontend</gsa:content>
  <gsa:content name="2">
    Google News,ExactMatch,http://news.google.com/,News
  </gsa:content>
  <gsa:content name="numLines">3</gsa:content>
  <gsa:content name="1">
    Google Search,PhraseMatch,http://www.google.com/,I'm Feeling Lucky!
  </gsa:content>
  <gsa:content name="0">
    Python,KeywordMatch,http://www.python.org/,Python Programming Language
  </gsa:content>
</entry>
```

Updating KeyMatch Settings

To change KeyMatch settings, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/keymatch/Front_End
```

The following example appends KeyMatch settings:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='updateMethod'>append</gsa:content>
  <gsa:content name='newLines'>
    image,KeywordMatch,http://images.google.com/,Google Image Search
    video,KeywordMatch,http://www.youtube.com/,Youtube
    rss feed,PhraseMatch,http://www.google.com/reader,Reader
  </gsa:content>
</entry>
```

The following example updates KeyMatch settings:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='updateMethod'>update</gsa:content>
  <gsa:content name='startLine'>0</gsa:content>
  <gsa:content name='originalLines'>
    image,KeywordMatch,http://images.google.com/,Google Image Search
    video,KeywordMatch,http://www.youtube.com/,Youtube rss
    feed,PhraseMatch,http://www.google.com/reader,Reader
  </gsa:content>
  <gsa:content name='newLines'>
    '''
    video,KeywordMatch,http://video.google.com/,Video Search
    rss feed,PhraseMatch,http://www.example.com/,RSS example
  </gsa:content>
</entry>
```

Note: To delete a KeyMatch setting, specify a line as three commas (, , ,).

The following example replaces a KeyMatch setting:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='updateMethod'>replace</gsa:content>
  <gsa:content name='newLines'>
    image,KeywordMatch,http://images.google.com/,Google Image Search
    video,KeywordMatch,http://www.youtube.com/,Youtube
    rss feed,PhraseMatch,http://www.google.com/reader,Reader
  </gsa:content>
</entry>
```

Related Queries

Retrieve or update related queries on a search appliance using the `synonym` feed. (Related queries are also known as synonyms.)

Use related queries to associate alternative words or phrases with specified search terms.

Parameter	Description
<code>query</code>	A query string to perform a full-text search. For example, if you specify <code>computer</code> in the <code>query</code> parameter, then you can view all related query settings that contain the word <code>computer</code> .
<code>startLine</code>	The starting line number of the results, the default value is 0 lines.
<code>maxLines</code>	The number of result lines in a response, the default value is 50 lines.

Use the following properties:

Property	Description
<i>line_number</i>	The <i>line_number</i> of a related query configuration rule in the list of rules.
<i>newLines</i>	The new related query configuration to change. You can include multiple lines of related query values. The line delimiter is \n.
<i>numLines</i>	The number of total result lines.
<i>originalLines</i>	The original related query configurations to change. You can include multiple lines of related query values. The line delimiter is \n.
<i>startLine</i>	The starting line number of the related query configuration to change. The minimum value is 0.
<i>updateMethod</i>	The method to change related query configurations. Possible values are: <ul style="list-style-type: none">• <i>update</i>. Update part of the related query configuration table to the new configurations. You can also delete related query configurations using the <i>update</i> method example in "Updating Related Queries" on page 53.• <i>append</i>. Add a new related query configuration to the end of the related query configuration table.• <i>replace</i>. Delete all rules in the related query configuration table and then append a new rule that you provide.

A related queries configuration rule is in the following format:

```
Search_Terms,Related_Queries
```

The *Search_Terms* and the *Related_Queries* values cannot be empty. The related queries configuration conforms to the CSV format, which uses a comma to separate values.

Retrieving Related Queries

To get related queries, send an authenticated GET request to the following URL (wrapped for readability):

```
http://Search_Appliance:8000/feeds/synonym/  
Front_End?query=Search_String&startLine=Start_Line&maxLines=Max_Lines
```

The following example retrieves related queries:

```
<entry xmlns="http://www.w3.org/2005/Atom"  
  xmlns:gsa="http://schemas.google.com/gsa/2007">  
  <id>http://ent1:8000/feeds/synonym/default_frontend</id>  
  <updated>2008-12-15T06:41:20.954Z</updated>  
  <link href="http://sa42.example.com:8000/feeds/synonym/default_frontend"  
    rel="self" type="application/atom+xml"/>  
  <link href="http://sa42.example.com:8000/feeds/synonym/default_frontend"  
    rel="edit" type="application/atom+xml"/>  
  <gsa:content name="entryID">default_frontend</gsa:content>  
  <gsa:content name="2">stock, security</gsa:content>  
  <gsa:content name="numLines">3</gsa:content>  
  <gsa:content name="1">google, googol</gsa:content>  
  <gsa:content name="0">airplane, aircraft</gsa:content>  
</entry>
```

Updating Related Queries

To change related queries, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/synonym/Front_End
```

The following example appends related queries:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='updateMethod'>append</gsa:content>
  <gsa:content name='newLines'>
    airplane,aircraft
    google,googol
    stock,security
  </gsa:content>
</entry>
```

The following example updates related queries:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='updateMethod'>update</gsa:content>
  <gsa:content name='startLine'>0</gsa:content>
  <gsa:content name='originalLines'>
    airplane,aircraft
    google,googol
  </gsa:content>
  <gsa:content name='newLines'>
    airplane,helicopter
  </gsa:content>
</entry>
```

Note: To delete an existing setting, specify a line as a single comma (,).

The following example replaces all related queries:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='updateMethod'>replace</gsa:content>
  <gsa:content name='newLines'>
    airplane,aircraft
    google,googol
    stock,security
  </gsa:content>
</entry>
```

Query Suggestion

There are two features for working with query suggestions:

- “Query Suggestion Blacklist” on page 54
- “Query Suggestion Refresh” on page 55

Query Suggestion Blacklist

The query suggestion blacklist supports the `/suggest` feature described in the “Query Suggestion Service `/suggest` Protocol” chapter of the *Search Protocol Reference*. This feature uses the `suggest` feed to retrieve and update the query suggestion blacklist entries.

Property	Description
<code>suggestBlacklist</code>	Content of the suggest blacklist file.

The query suggestion blacklist supports the regular expressions in the re2 library (<http://code.google.com/p/re2/wiki/Syntax>). If you want specify an exact match, you need to use the following syntax:

```
^the_word_to_match$
```

Retrieving Query Suggestion Blacklist Information

Retrieve query suggestion blacklist information as follows:

```
GET request
URL: http://Search_Appliance:8000/feeds/suggest/suggestBlacklist
```

Updating Query Suggestion Blacklist Entries

Update query suggestion blacklist entries as follows:

```
PUT request
URL: http://Search_Appliance:8000/feeds/suggest/suggestBlacklist
<?xml version='1.0' encoding='UTF-8'?>
<atom:entry xmlns:atom='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'
  xmlns:apps='http://schemas.google.com/apps/2006'>
  <gsa:content name='suggestBlacklist'>
    bad_word_3
    ^bad_word_1$
    car[0-9]{4}.*
  </gsa:content>
</atom:entry>
```

Query Suggestion Refresh

The query suggestion refresh supports the `/suggest` feature described in the “Query Suggestion Service `/suggest` Protocol” chapter of the *Search Protocol Reference*. This feature uses the `suggest` feed to refresh the query suggestion database.

Property	Description
<code>suggestRefresh</code>	Triggers a query suggestion refresh.

Refresh query suggestions as follows:

```
PUT request
URL: http://Search_Appliance:8000/feeds/suggest/suggestRefresh
<?xml version='1.0' encoding='UTF-8'?>
<atom:entry xmlns:atom='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'
  xmlns:apps='http://schemas.google.com/apps/2006'>
<gsa:content name='suggestRefresh'>1</gsa:content>
</atom:entry>
```

Search Status

Retrieve serving status for a search appliance using the `servingStatus` entry of the `status` feed.

Property	Description
<code>queriesPerMinute</code>	Average queries per minute recently served on the search appliance.
<code>searchLatency</code>	Recent search latency in seconds.

Retrieving the Serving Status Entry

To get the current search appliance serving status, send an authenticated `GET` request to the `status` feed URL:

```
http://Search_Appliance:8002/feeds/status/servingStatus
```

The following result is an entry that includes the current serving status values for the search appliance:

```
<?xml version="1.0" encoding="UTF-8"?>
<entry xmlns="http://www.w3.org/2005/Atom"
  xmlns:gsa="http://schemas.google.com/gsa/2007">
  <id>http://gsa:8002/feeds/status/servingStatus</id>
  <updated>2014-03-14T16:05:56.668Z</updated>
  <link rel="self" type="application/atom+xml" href="http://gsa:8002/feeds/status/servingStatus"/>
  <link rel="edit" type="application/atom+xml" href="http://gsa:8002/feeds/status/servingStatus"/>
  <gsa:content name="entryID">servingStatus</gsa:content>
  <gsa:content name="searchLatency">0.07</gsa:content>
  <gsa:content name="queriesPerMinute">0.6</gsa:content>
</entry>
```

Reports

The sections that follow describe how to configure the **Reports** features of the Admin Console:

- “Search Reports” on page 56
- “Search Logs” on page 61

Search Reports

Generate, update and delete search log using the `searchReport` feed and the following properties.

Property	Description
<code><Entry Name></code>	<code><Search_Report_Name>@<Collection_Name></code>
<code>collectionName</code>	(Write only) The collection name, which is only needed when creating a search report.
<code>diagnosticTerms</code>	Terms to exclude when running scripts that create diagnostic data from test queries. All the specified terms in a search query are removed from the report. Use commas to separate multiple terms.
<code>isFinal</code>	(Read only) Indicates if the search report contains the final result. If so, it means the last update date is later than <code>reportDate</code> .
<code>reportContent</code>	(Read only) The search report content, which is only returned when get search report content and content is ready.
<code>reportCreationDate</code>	(Read only) The creation date of the search report.
<code>reportDate</code>	The dates of the queries that are collected in the search report.
<code>reportName</code>	(Write only) The report name, which is only needed when creating a search report.
<code>reportState</code>	(Read only) The status of the search report. 0: Initialized; 1: Report in progress; 2: Report competed; 3: Non-final complete report is being generated; 4: Last report generation failed.
<code>topCount</code>	The number of top queries to be generated.
<code>withResults</code>	Indicates if a search has results. The default value is <code>false</code> .

Listing a Search Report

List a search report using the following query parameters:

Parameter	Description
collectionName	Collection name for the search report. The default value is <code>all.collections</code> .

To list search report entries, send an authenticated `GET` request to the `root` entry of the `searchReport` feed.

```
http://Search_Appliance:8000/feeds/searchReport/
```

A list of search report entries are returned.

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
      xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
      xmlns:gsa='http://schemas.google.com/gsa/2007'>
<id>http://gsa:8000/feeds/searchReport</id>
<updated>2009-03-26T07:26:55.991Z</updated>
<link rel='http://schemas.google.com/g/2005#feed'
      type='application/atom+xml' href='http://gsa:8000/feeds/searchReport' />
<link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/searchReport' />
<generator version='0.5' uri='http://gsa:8000/gsa'>
  Google Search Appliance
</generator>
<openSearch:startIndex>1</openSearch:startIndex>

<entry>
  <id>http://gsa:8000/feeds/searchReport/aaa@default_collection</id>
  <updated>2009-03-26T07:26:55.991Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T07:26:55.991Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchReport' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchReport' />
  <gsa:content name='entryID'>aaa@default_collection</gsa:content>
  <gsa:content name='diagnosticTerms'>comments</gsa:content>
  <gsa:content name='reportState'>2</gsa:content>
  <gsa:content name='reportCreationDate'>
    March 26, 2009 12:14:14 AM PDT
  </gsa:content>
  <gsa:content name='reportDate'>month_3_2009</gsa:content>
  <gsa:content name='withResults'>>true</gsa:content>
  <gsa:content name='topCount'>100</gsa:content>
  <gsa:content name='isFinal'>>false</gsa:content>
</entry>
```

```

<entry>
  <id>http://gsa:8000/feeds/searchReport/bbb@default_collection</id>
  <updated>2009-03-26T07:26:55.991Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T07:26:55.991Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchReport' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchReport' />
  <gsa:content name='entryID'>bbb@default_collection</gsa:content>
  <gsa:content name='diagnosticTerms'></gsa:content>
  <gsa:content name='reportState'>2</gsa:content>
  <gsa:content name='reportCreationDate'>
    March 26, 2009 12:24:16 AM PDT
  </gsa:content>
  <gsa:content name='reportDate'>month_3_2009</gsa:content>
  <gsa:content name='withResults'>true</gsa:content>
  <gsa:content name='topCount'>100</gsa:content>
  <gsa:content name='isFinal'>false</gsa:content>
</entry>
</feed>

```

Creating a Search Report

Create a new search report entry by sending an authenticated POST request to the `root` entry of the `searchReport` feed.

```
http://Search_Appliance:8000/feeds/searchReport/
```

The possible date formats for reports are as follows.

Purpose	Format
Date	date _month_day_year
Month	month _month_year
Year	year _year
Date range	range _month_day_year_month_day_year

An example request with content is:

```

<?xml version='1.0' encoding='UTF-8'?>
  <entry xmlns='http://www.w3.org/2005/Atom'
    xmlns:gsa='http://schemas.google.com/gsa/2007'>
    <gsa:content name='reportName'>bbb</gsa:content>
    <gsa:content name='collectionName'>default_collection</gsa:content>
    <gsa:content name='reportDate'>month_3_2009</gsa:content>
    <gsa:content name='withResults'>true</gsa:content>
    <gsa:content name='topCount'>100</gsa:content>
  </entry>

```

A new search report entry is generated and returned:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/searchReport</id>
  <updated>2009-03-26T07:22:25.162Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T07:22:25.162Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchReport' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchReport' />
  <gsa:content name='entryID'>bbb@default_collection</gsa:content>
  <gsa:content name='diagnosticTerms'></gsa:content>
  <gsa:content name='reportState'>1</gsa:content>
  <gsa:content name='reportCreationDate'>
    March 26, 2009 12:22:25 AM PDT
  </gsa:content>
  <gsa:content name='reportDate'>month_3_2009</gsa:content>
  <gsa:content name='withResults'>>true</gsa:content>
  <gsa:content name='topCount'>100</gsa:content>
  <gsa:content name='isFinal'>>false</gsa:content>
</entry>
```

Retrieving a Search Report

To check search report status and retrieve search log content, send an authenticated GET request to a search report entry of the `searchReport` feed.

```
http://Search_Appliance:8000/feeds/searchReport/aaa@default_collection
```

The following is a returned search report entry that contains log content (if the content is ready):

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/searchReport/aaa%40default_collection</id>
  <updated>2009-03-26T07:14:56.343Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T07:14:56.343Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchReport/aaa%40default_collection' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchReport/aaa%40default_collection' />
  <gsa:content name='entryID'>aaa@default_collection</gsa:content>
  <gsa:content name='diagnosticTerms'>comments</gsa:content>
  <gsa:content name='reportState'>2</gsa:content>
  <gsa:content name='reportContent'>*****Report Content*****</gsa:content>
  <gsa:content name='reportCreationDate'>
    March 26, 2009 12:14:14 AM PDT
  </gsa:content>
  <gsa:content name='reportDate'>month_3_2009</gsa:content>
  <gsa:content name='withResults'>>true</gsa:content>
  <gsa:content name='topCount'>100</gsa:content>
  <gsa:content name='isFinal'>>false</gsa:content>
</entry>
```

Updating a Search Report

Update the search report status and get search report content by sending an authenticated `PUT` request to a search report entry of the `searchReport` feed. There are no properties for this feed.

```
http://Search_Appliance:8000/feeds/searchReport/bbb@default_collection
```

An example request with content is:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
</entry>
```

A search log entry is returned:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/searchReport/bbb%40default_collection</id>
  <updated>2009-03-26T07:24:16.099Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T07:24:16.099Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchReport/bbb%40default_collection' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchReport/bbb%40default_collection' />
  <gsa:content name='entryID'>bbb@default_collection</gsa:content>
  <gsa:content name='diagnosticTerms'></gsa:content>
  <gsa:content name='reportState'>3</gsa:content>
  <gsa:content name='reportCreationDate'>
    March 26, 2009 12:22:25 AM PDT
  </gsa:content>
  <gsa:content name='reportDate'>month_3_2009</gsa:content>
  <gsa:content name='withResults'>>true</gsa:content>
  <gsa:content name='topCount'>100</gsa:content>
  <gsa:content name='isFinal'>>false</gsa:content>
</entry>
```

Deleting a Search Report

To update the search report status and get search log content, send an authenticated `DELETE` request to a search report entry of the `searchReport` feed.

```
http://Search_Appliance:8000/feeds/searchReport/bbb@default_collection
```

A search report entry will be deleted.

Search Logs

Generate, update, and delete search logs using the `searchLog` feed.

Search log entry properties:

Property	Description
<code><Entry Name></code>	<code><Search_Log_Name>@<Collection_Name></code>
<code>collectionName</code>	(Write only) The collection name, which is only needed when creating a search log.
<code>fromLine</code>	(Read only) The starting line of a search log that returns in <code>logContent</code> . This property is only returned when getting search log content and the content is ready.
<code>isFinal</code>	(Read only) Indicates if the search log contains the final result. If so, it means the last update date is later than <code>reportDate</code> .
<code>logContent</code>	(Read only) A part of the search log content that is returned when getting search log content and the content is ready.
<code>reportCreationDate</code>	(Read only) The creation date of a search log.
<code>reportDate</code>	The dates for the queries that are collected in the search log.
<code>reportName</code>	(Write only) The report name, which is only needed when creating a search log.
<code>reportState</code>	(Read only) The status of the search log: 0: Initialized; 1: Report is in progress; 2: Report competed; 3: Non-final complete report is in progress; 4: Last report generation failed.
<code>toLine</code>	(Read only) The ending line of the search log that is returned in <code>logContent</code> . This property is only returned when getting search log content and the content is ready.
<code>totalLines</code>	(Read only) The number of lines in the search log that are returned in <code>logContent</code> . This property is only returned when getting search log content and the content is ready.

Listing a Search Log

List the entries in a search log using the following query parameters:

Parameter	Description
<code>collectionName</code>	Collection Name of a search log. The default value is <code>all.collections</code> .

To list search log entries, send an authenticated GET request to `root` entry of the `searchLog` feed.

`http://Search_Appliance:8000/feeds/searchLog/`

A list of search log entries is returned:

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
  xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/searchLog</id>
  <updated>2009-03-26T06:44:31.094Z</updated>
  <link rel='http://schemas.google.com/g/2005#feed'
    type='application/atom+xml' href='http://gsa:8000/feeds/searchLog' />
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchLog' />
  <generator version='0.5' uri='http://gsa:8000/gsa'>
    Google Search Appliance
  </generator>
  <openSearch:startIndex>1</openSearch:startIndex>

  <entry>
    <id>http://gsa:8000/feeds/searchLog/aaa@default_collection</id>
    <updated>2009-03-26T06:44:31.094Z</updated>
    <app:edited xmlns:app='http://purl.org/atom/app#'>
      2009-03-26T06:44:31.094Z
    </app:edited>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/searchLog' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/searchLog' />
    <gsa:content name='entryID'>aaa@default_collection</gsa:content>
    <gsa:content name='reportState'>2</gsa:content>
    <gsa:content name='reportCreationDate'>
      March 25, 2009 11:20:20 PM PDT
    </gsa:content>
    <gsa:content name='reportDate'>date_3_25_2009</gsa:content>
    <gsa:content name='isFinal'>>false</gsa:content>
  </entry>

  <entry>
    <id>http://gsa:8000/feeds/searchLog/bbb@default_collection</id>
    <updated>2009-03-26T06:44:31.094Z</updated>
    <app:edited xmlns:app='http://purl.org/atom/app#'>
      2009-03-26T06:44:31.094Z
    </app:edited>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/searchLog' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/searchLog' />
    <gsa:content name='entryID'>bbb@default_collection</gsa:content>
    <gsa:content name='reportState'>2</gsa:content>
    <gsa:content name='reportCreationDate'>
      March 25, 2009 11:42:28 PM PDT
    </gsa:content>
    <gsa:content name='reportDate'>date_3_25_2009</gsa:content>
    <gsa:content name='isFinal'>>false</gsa:content>
  </entry>
</feed>
```

Creating a Search Log

To create a new search log entry, send an authenticated `POST` request to the `root` entry of the `searchLog` feed:

```
http://Search_Appliance:8000/feeds/searchLog/
```

A request with content is as follows:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='reportName'>bbb</gsa:content>
  <gsa:content name='collectionName'>default_collection</gsa:content>
  <gsa:content name='reportDate'>date_3_25_2009</gsa:content>
</entry>
```

A new search log entry generates and returns:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/searchLog/</id>
  <updated>2009-03-26T06:42:28.742Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T06:42:28.742Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchLog/'>
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchLog/'>
  <gsa:content name='entryID'>bbb@default_collection</gsa:content>
  <gsa:content name='reportState'>1</gsa:content>
  <gsa:content name='reportCreationDate'>
    March 25, 2009 11:42:28 PM PDT
  </gsa:content>
  <gsa:content name='reportDate'>date_3_25_2009</gsa:content>
  <gsa:content name='isFinal'>>false</gsa:content>
</entry>
```

Retrieving Search Log Content

To check the search log status and get search log content, send an authenticated `GET` request to a search

log entry of the `searchLog` feed using the following parameters.

Parameter	Description
<code>query</code>	Query string for the <code>logContent</code> . The <code>logContent</code> contains many lines of logs. The query string applies to each line and only lines that contain the query string are returned.
<code>maxLines</code>	The maximum <code>logContent</code> lines to retrieve. The default value is 50 lines.
<code>startLine</code>	The first <code>logContent</code> lines to retrieve. The default value is 1 line.

Example:

```
http://Search_Appliance:8000/feeds/searchLog/
aaa@default_collection?query=document
```

A search log entry with logContent (if content is ready) returns:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/searchLog/aaa%40default_collection</id>
  <updated>2009-03-26T06:22:41.416Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T06:22:41.416Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchLog/aaa%40default_collection'/>
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchLog/aaa%40default_collection'/>
  <gsa:content name='entryID'>aaa@default_collection</gsa:content>
  <gsa:content name='toLine'>2</gsa:content>
  <gsa:content name='logContent'>
127.0.0.2!127.0.0.1 - - [25/Mar/2009:23:18:43 -0800] &quot;GET
 /search?q=document&amp;btnG=Google+Search&amp;access=p&amp;
 client=default_frontend&amp;output=xml_no_dtd&amp;
 proxystylesheet=default_frontend&amp;sort=date%3AD%3AL%3Ad1&amp;
 entqr=0&amp;oe=UTF-8&amp;ie=UTF-8&amp;ud=1&amp;site=default_collection&amp;
 ip=172.30.120.197 HTTP/1.1&quot; 200 2432 3 0.02
127.0.0.2!127.0.0.1 - - [25/Mar/2009:23:18:14 -0800] &quot;GET
 /search?q=document&amp;btnG=Google+Search&amp;access=p&amp;
 client=default_frontend&amp;output=xml_no_dtd&amp;
 proxystylesheet=default_frontend&amp;sort=date%3AD%3AL%3Ad1&amp;
 entqr=0&amp;oe=UTF-8&amp;ie=UTF-8&amp;ud=1&amp;site=default_collection&amp;
 ip=172.30.120.197 HTTP/1.1&quot; 200 2432 3 0.02
  </gsa:content>
  <gsa:content name='reportState'>2</gsa:content>
  <gsa:content name='fromLine'>1</gsa:content>
  <gsa:content name='totalLines'>2</gsa:content>
  <gsa:content name='reportCreationDate'>
    March 25, 2009 11:20:20 PM PDT
  </gsa:content>
  <gsa:content name='reportDate'>date_3_25_2009</gsa:content>
  <gsa:content name='isFinal'>>false</gsa:content>
</entry>
```

Updating a Search Log

To update the search log status and get search log content, send an authenticated PUT request to a search log entry of the searchLog feed. There are no properties for this use of the searchLog feed:

```
http://Search_Appliance:8000/feeds/searchLog/bbb@default_collection
```

Specify a request with content:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
</entry>
```


A search log entry returns:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/searchLog/bbb%40default_collection</id>
  <updated>2009-03-26T06:50:05.928Z</updated>
  <app:edited xmlns:app='http://purl.org/atom/app#'>
    2009-03-26T06:50:05.928Z
  </app:edited>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchLog/bbb%40default_collection' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/searchLog/bbb%40default_collection' />
  <gsa:content name='entryID'>bbb@default_collection</gsa:content>
  <gsa:content name='reportState'>3</gsa:content>
  <gsa:content name='reportCreationDate'>
    March 25, 2009 11:42:28 PM PDT
  </gsa:content>
  <gsa:content name='reportDate'>date_3_25_2009</gsa:content>
  <gsa:content name='isFinal'>false</gsa:content>
</entry>
```

Deleting a Search Log

To update the search log status and get search log content, send an authenticated `DELETE` request to a search log entry of the `searchLog` feed.

```
http://Search_Appliance:8000/feeds/searchLog/bbb@default_collection
```

A search log entry will be deleted.

GSA Unification

The sections that follow describe how to configure the GSA Unification features of the Admin Console:

- “Configuring a GSA Unification Network” on page 66
- “Adding a GSA Unification Node” on page 66
- “Retrieving a Node Configuration” on page 67
- “Retrieving All Node Configurations” on page 68
- “Updating a Node Configuration” on page 69
- “Deleting a Node” on page 69

GSA Unification is also known as dynamic scalability. GSA Unification features are provided by the `federation` feed.

Configuring a GSA Unification Network

Retrieve, update, create, or delete the GSA Unification node configuration and retrieve the node configuration of all nodes in the network on the Google Search Appliance.

Property	Description																																	
applianceId	The ID of the search appliance, required to identify the node in node operations.																																	
federationNetworkIP	The private tunnel IP address (virtual address) for the node. This address must be an RFC 1918 address. Note: A GSA Unification works best when the IP addresses of the nodes are numerically near, such as 10.1.1.1, 10.1.1.2, 10.1.1.3, and so on. The search appliance disallows a GSA Unification for nodes that are not in the same /16 subnet. This is a problem only if there are more than 65534 nodes in a GSA Unification network. GSA Unification nodes communicate on TCP port 10999.																																	
hostname	The host name of the search appliance.																																	
nodeType	The type of search appliance. Possible values: <ul style="list-style-type: none">PRIMARY: The node merges results from other nodes.SECONDARY: The node serves results to the other nodes.PRIMARY_AND_SECONDARY: The node acts as both a Primary and Secondary node.																																	
scoringBias	The scoring bias value for this node. Valid values are integers between -99 and 99. The scoring bias value reflects the weighting to be given to results from this node. A higher value means a higher weighting. The values and their equivalent in the Admin Console are: <table border="1"><thead><tr><th colspan="3">Less influence</th><th colspan="4">No influence</th><th colspan="4">More influence</th></tr></thead><tbody><tr><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td></tr><tr><td>-99</td><td>-80</td><td>-60</td><td>-40</td><td>-20</td><td>0</td><td>20</td><td>40</td><td>60</td><td>80</td><td>99</td></tr></tbody></table>	Less influence			No influence				More influence				<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	-99	-80	-60	-40	-20	0	20	40	60	80	99
Less influence			No influence				More influence																											
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>																								
-99	-80	-60	-40	-20	0	20	40	60	80	99																								
secretToken	The secret token that you use to establish a connection to this node. This token can be any non-empty string. The remote search appliance needs this token for the connection handshake.																																	

Adding a GSA Unification Node

To add a GSA Unification node, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/federation
```

The following is an example of a request body:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='entryID'>S4-JAX9N2PQ4GNAB</gsa:content>
  <gsa:content name='nodeType'>SECONDARY</gsa:content>
  <gsa:content name='federationNetworkIP'>10.0.0.2</gsa:content>
  <gsa:content name='secretToken'>token</gsa:content>
  <gsa:content name='hostname'>host1.domain.com</gsa:content>
  <gsa:content name='scoringBias'>20</gsa:content>
</entry>
```

Retrieving a Node Configuration

To retrieve the configuration information about a GSA Unification node, send an authenticated GET request to the following URL:

```
http://Search_Appliance:8000/feeds/federation/Appliance_Id
```

The following example shows a sample result for a secondary node:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/federation/S4-JAX9N2PQ4GNAB</id>
  <updated>2008-12-11T08:18:04.372Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/federation/S4-JAX9N2PQ4GNAB' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/federation/S4-JAX9N2PQ4GNAB' />
  <gsa:content name='entryID'>S4-JAX9N2PQ4GNAB</gsa:content>
  <gsa:content name='nodeType'>SECONDARY</gsa:content>
  <gsa:content name='federationNetworkIP'>10.0.0.2</gsa:content>
  <gsa:content name='secretToken'>token</gsa:content>
  <gsa:content name='hostname'>host1.domain.com</gsa:content>
  <gsa:content name='scoringBias'>20</gsa:content>
  <gsa:content name='remoteFrontend'>remoteFrontend</gsa:content>
  <gsa:content name='slaveTimeout'>100</gsa:content>
</entry>
```

The following example shows a sample result for a primary node:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/federation/S4-JAX9N2PQ4GNAB</id>
  <updated>2008-12-11T08:18:04.372Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/federation/S4-JAX9N2PQ4GNAB' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/federation/S4-JAX9N2PQ4GNAB' />
  <gsa:content name='entryID'>S4-JAX9N2PQ4GNAB</gsa:content>
  <gsa:content name='nodeType'>PRIMARY</gsa:content>
  <gsa:content name='federationNetworkIP'>10.0.0.2</gsa:content>
  <gsa:content name='secretToken'>token</gsa:content>
  <gsa:content name='hostname'>host1.domain.com</gsa:content>
  <gsa:content name='secondaryNodes'>Appliance_ID1, Appliance_ID2</gsa:content>
</entry>
```

Retrieving All Node Configurations

To retrieve information on all GSA Unification nodes, send an authenticated GET request to the following URL:

```
http://Search_Appliance:8000/feeds/federation
```

The following example shows a sample result for a secondary node:

```
<?xml version='1.0' encoding='UTF-8'?>
<feed xmlns='http://www.w3.org/2005/Atom'
  xmlns:openSearch='http://a9.com/-/spec/opensearchrss/1.0/'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/federation</id>
  <updated>2008-12-11T08:01:21.253Z</updated>
  <link rel='http://schemas.example.com/g/2005#feed'
    type='application/atom+xml' href='http://gsa:8000/feeds/federation' />
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/federation' />
  <generator version='0.5' uri='http://gsa:8000/gsa'>
    Google Search Appliance
  </generator>
  <openSearch:startIndex>1</openSearch:startIndex>

  <entry>
    <id>http://gsa:8000/feeds/federation/ApplianceId1</id>
    <updated>2008-12-11T08:01:21.253Z</updated>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/federation' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/federation' />
    <gsa:content name='entryID'>Appliance_Id1</gsa:content>
    <gsa:content name='nodeType'>SECONDARY</gsa:content>
    <gsa:content name='federationNetworkIP'>10.0.0.2</gsa:content>
    <gsa:content name='secretToken'>token</gsa:content>
    <gsa:content name='hostname'>host1.domain.com</gsa:content>
    <gsa:content name='scoringBias'>20</gsa:content>
    <gsa:content name='remoteFrontend'>remoteFrontend</gsa:content>
    <gsa:content name='slaveTimeout'>100</gsa:content>
  </entry>

  <entry>
    <id>http://gsa:8000/feeds/collection/new2_collection</id>
    <updated>2008-12-11T08:01:21.253Z</updated>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/federation' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/federation' />
    <gsa:content name='entryID'>Appliance_Id</gsa:content>
    <gsa:content name='nodeType'>PRIMARY</gsa:content>
    <gsa:content name='federationNetworkIP'>10.0.0.3</gsa:content>
    <gsa:content name='secretToken'>token1</gsa:content>
    <gsa:content name='hostname'>host2.domain.com</gsa:content>
    <gsa:content name='scoringBias'>40</gsa:content>
    <gsa:content name='secondaryNodes'></gsa:content>
  </entry>
</feed>
```

Updating a Node Configuration

To update the configuration of a node in the GSA Unification network, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/collection/Appliance_Id
```

Note: Changing the Appliance Id isn't possible in an update request. In this case the search appliance should be deleted from the network and added again.

The following example request body shows the result:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='entryID'>Appliance_Id</gsa:content>
  <gsa:content name='nodeType'>SECONDARY</gsa:content>
  <gsa:content name='federationNetworkIP'>10.0.0.5</gsa:content>
  <gsa:content name='secretToken'>token2</gsa:content>
  <gsa:content name='hostname'>host5.domain.com</gsa:content>
  <gsa:content name='scoringBias'>40</gsa:content>
</entry>
```

Deleting a Node

To delete a node from the GSA Unification network, send an authenticated `DELETE` request to the following URL:

```
http://Search_Appliance:8000/feeds/federation/Appliance_Id
```

Administration

The sections that follow describe how to configure the **Administration** features of the Admin Console:

- “License Information” on page 69
- “Import and Export” on page 71
- “Event Log” on page 72
- “System Status” on page 73
- “Shut Down and Reboot” on page 74

License Information

Retrieve license information for a search appliance using the `licenseInfo` entry of the `info` feed.

Note: You can only view license information with this API, installing a new license is not supported.

Property	Description
applianceID	Provides the identification value for the Google Search Appliance software. This value is also known as the serial number for the software.
licenseID	Provides the unique license identification value.
licenseValidUntil	Identifies when the search appliance software license will expire.
maxCollections	Indicates the maximum number of collections. Configure collections at the Crawl and Index > Collections page.
maxFrontends	Indicates the maximum number of front ends. Configure front ends at the Serving > Front Ends page.
maxPages	Maximum number of content items that you can index with this product. Content items include documents, images, and content from the feeds interface.

Retrieving License Information

To get the license information for a search appliance, send an authenticated GET request to the `info` feed URL:

```
http://Search_Appliance:8000/feeds/info/licenseInfo
```

The following example result is an entry that includes current license information values for the search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/info/licenseInfo</id>
  <updated>2008-12-12T09:11:42.455Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/info/licenseInfo' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/info/licenseInfo' />
  <gsa:content name='entryID'>licenseInfo</gsa:content>
  <gsa:content name='maxFrontends'>unlimited</gsa:content>
  <gsa:content name='licenseID'>
    license_S5-QJBPL6N3H8JJA_20081211_220512
  </gsa:content>
  <gsa:content name='maxPages'>unlimited</gsa:content>
  <gsa:content name='maxCollections'>unlimited</gsa:content>
  <gsa:content name='licenseValidUntil'>March 7, 9009</gsa:content>
  <gsa:content name='applianceID'>S5-QJBPL6N3H8JJA</gsa:content>
</entry>
```

Import and Export

Import or export a search appliance configuration using the `importExport` entry of the `config` feed.

Common query parameters for all requests:

Parameter	Description
<code>password</code>	The password of the exported configuration

The `importExport` entry properties:

Property	Description
<code>xmlData</code>	The content of exported configuration
<code>password</code>	The password for generating configuration file

Exporting a Configuration

To export a search appliance configuration, send an authenticated `GET` request to the `importExport` entry of the `config` feed:

```
http://Search_Appliance:8000/feeds/config/importExport?password=12345678
```

An `importExport` entry returns:

```
<?xml version='1.0' encoding='UTF-8'?>
  <entry xmlns='http://www.w3.org/2005/Atom'
    xmlns:gsa='http://schemas.google.com/gsa/2007'>
    <id>http://gsa:8000/feeds/config/importExport</id>
    <updated>2009-03-26T05:56:23.092Z</updated>
    <app:edited xmlns:app='http://purl.org/atom/app#'>
      2009-03-26T05:56:23.092Z
    </app:edited>
    <link rel='self' type='application/atom+xml'
      href='http://gsa:8000/feeds/config/importExport' />
    <link rel='edit' type='application/atom+xml'
      href='http://gsa:8000/feeds/config/importExport' />
    <gsa:content name='entryID'>importExport</gsa:content>
    <gsa:content name='xmlData'>
      *****configuration content*****
    </gsa:content>
  </entry>
```

Import a Configuration

To import a search appliance configuration, send an authenticated `PUT` request to the `importExport` entry of the `config` feed:

```
http://Search_Appliance:8000/feeds/config/importExport
```

The following example shows an `importExport` entry with content:

```
<?xml version='1.0' encoding='UTF-8'?>
  <entry xmlns='http://www.w3.org/2005/Atom'
    xmlns:gsa='http://schemas.google.com/gsa/2007'>
    <gsa:content name='password'>12345678</gsa:content>
    <gsa:content name='xmlData'>
      *****configuration content*****
    </gsa:content>
  </entry>
```

Event Log

Retrieve the event log for a search appliance using the `eventLog` entry of the `logs` feed.

Parameter	Description
<code>query</code>	Query string for the <code>logContent</code> . The <code>logContent</code> contains many lines of logs. The query string applies to each line and only lines that contain the query string are returned.
<code>startLine</code>	The first <code>logContent</code> lines to retrieve. The default value is 1 line.
<code>maxLines</code>	The maximum <code>logContent</code> lines to retrieve. The default value is 50 lines.

The following properties enable access to log content.

Property	Description
<code>fromLine</code>	The starting line of the <code>logContent</code> .
<code>logContent</code>	The log content.
<code>toLine</code>	The ending line of the <code>logContent</code> .
<code>totalLines</code>	Total lines of the <code>logContent</code> .

Retrieving the Event Log

Retrieve the event log information for a search appliance by sending an authenticated `GET` request to the `eventLog` feed URL (wrapped for readability):

```
http://Search_Appliance:8000/feeds/logs/eventLog?
query=User&startLine=Starting_Line&maxLines=Max_Lines
```


The result is an entry that includes the current event log values for the search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/logs/eventLog</id>
  <updated>2008-12-12T09:03:37.294Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/logs/eventLog' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/logs/eventLog' />
  <gsa:content name='entryID'>eventLog</gsa:content>
  <gsa:content name='toLine'>11</gsa:content>
  <gsa:content name='logContent'>
    @ 2008/12/11 23:39:40: User logged in: [admin logged in from 172.30.123.69
    at 2008_12_11_23_39_40_PST]
    @ 2008/12/11 23:39:38: User logged in: [admin logged in from 172.30.123.69
    at 2008_12_11_23_39_38_PST]
  </gsa:content>
  <gsa:content name='fromLine'>10</gsa:content>
  <gsa:content name='totalLines'>67</gsa:content>
</entry>
```

System Status

Retrieve the system status for a search appliance using the `systemStatus` entry of the `status` feed.

Property	Description
<code>cpuTemperature</code>	Temperature of the CPU: 0 if okay, 1 if caution, 2 if critical.
<code>diskCapacity</code>	Remaining disk capacity of the search appliance: 0 if okay, 1 if caution, 2 if critical.
<code>machineHealth</code>	Health of the internal system components: 0 if okay, 1 if caution, 2 if critical.
<code>overallHealth</code>	Overall health of the entire search appliance: 0 if okay, 1 if caution, 2 if critical.
<code>raidHealth</code>	Health of the raid array: 0 if okay, 1 if caution, 2 if critical.

Note: Health properties differ by versions of the search appliance.

Retrieving a System Status Entry

To get the current search appliance system status, send an authenticated `GET` request to the `status` feed URL:

```
http://Search_Appliance:8000/feeds/status/systemStatus
```

The following result is an entry that includes current system status values for the search appliance:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <id>http://gsa:8000/feeds/status/systemStatus</id>
  <updated>2008-12-09T23:53:14.288Z</updated>
  <link rel='self' type='application/atom+xml'
    href='http://gsa:8000/feeds/status/systemStatus' />
  <link rel='edit' type='application/atom+xml'
    href='http://gsa:8000/feeds/status/systemStatus' />
  <gsa:content name='entryID'>systemStatus</gsa:content>
  <gsa:content name='overallHealth'>0</gsa:content>
  <gsa:content name='diskCapacity'>0</gsa:content>
  <gsa:content name='raidHealth'>0</gsa:content>
  <gsa:content name='cpuTemperature'>0</gsa:content>
  <gsa:content name='machineHealth'>0</gsa:content>
</entry>
```

Shut Down and Reboot

Shut down or reboot the search appliance.

Property	Description
command	Command sent to the search appliance. The command can be <code>shutdown</code> or <code>reboot</code> .
runningStatus	Indicates the search appliance status: <ul style="list-style-type: none">• <code>shuttingDown</code> if you sent the <code>shutdown</code> command.• <code>rebooting</code> if you sent the <code>reboot</code> command.• <code>running</code> if the search appliance is operating normally.

Shutting Down or Rebooting a Search Appliance

To shut down or reboot a search appliance, send an authenticated `PUT` request to the following URL:

```
http://Search_Appliance:8000/feeds/command/shutdown
```

The following example request body shows the result:

```
<?xml version='1.0' encoding='UTF-8'?>
<entry xmlns='http://www.w3.org/2005/Atom'
  xmlns:gsa='http://schemas.google.com/gsa/2007'>
  <gsa:content name='command'>reboot</gsa:content>
</entry>
```

Index

A

- Administration 69–74
- atom:entry element 8
- atom:feed element 7
- atom:id element 8
- atom:link element 8
- atom:updated element 9
- authentication 6

C

- collections
 - create 32
 - delete 34
 - retrieve 32
 - update 34
- command feed 24, 30
- config feed 12, 16, 17, 22, 23, 27, 71
- connector managers
 - delete 27
 - insert 25
 - retrieve 25
 - update 26
- contentStatistics feed 40, 41, 42
- crawl access rule
 - update 21
- crawl access rules
 - delete 21
 - insert 19
 - retrieve 19
- crawl and index 11–34
- crawl diagnostics
 - description entry parameters 36
 - document entry properties 37
 - get content statistics 40–42
 - get crawled document status 39
 - query parameters 36
 - status values 34–36
- crawl schedule
 - retrieve 17
 - update 18

- crawl status
 - pause or resume crawl 30
 - retrieve 30
- crawl URLs
 - retrieve 12
 - update 12
- create operations 6

D

- data source feed
 - delete 15
 - destroy 16
 - retrieve 13
- delete operations 6
- diagnostics feed 34, 36, 39
- document status, retrieve 31

E

- elements
 - atom:entry 8
 - atom:feed 7
 - atom:id 8
 - atom:link 8
 - atom:updated 9
 - gsa:content 10, 11
- event log, retrieve 72
- eventLog feed 72
- export configuration 71

F

- federation feed 65
- feed feed 14, 15

- feeds
 - command 24, 30
 - config 12, 16, 17, 22, 23, 27, 71
 - contentStatistics 40, 41, 42
 - data source 13–17
 - diagnostics 34, 36, 39
 - eventLog 72
 - federation 65
 - feed 14, 15
 - frontend 44, 45, 46
 - info 70
 - logs 72
 - onebox 28, 29
 - outputFormat 47
 - searchLog 63, 64, 65
 - searchReport 56, 58, 59, 60
 - status 31, 55, 73
 - suggest 55
 - synonym 51
- freshness tuning settings
 - retrieve 23
 - update 24
- front ends
 - delete 46
 - retrieve 44
- frontend feed 44, 45, 46
- frontendOnebox property 45

G

- GSA Unification 65–69
 - add nodes 66
 - delete nodes 69
 - retrieve nodes 68
 - update nodes 69
- gsa:content element 10, 11
- GSAEntry object 11

H

- host load schedule
 - retrieve 22
 - update 23
- HTTP requests 5, 6, 7, 10

I

- import configuration 71
- index, reset 42
- info feed 70

K

- KeyMatch settings
 - retrieve 49
 - update 50

L

- license information, retrieve 70
- logs feed 72

O

- OneBox
 - delete module 29
 - insert 45
 - retrieve names 28
 - retrieve settings 27
 - update 45
 - update settings 28
- onebox feed 28, 29
- operations 6
- outputFormat feed 46, 47

P

- password 6
- pause crawl 30

Q

- query suggestion
 - refresh 55
 - retrieve blacklist 54
 - update blacklist 54

R

- reboot a search appliance 74
- recrawl URL patterns 24
- related queries
 - retrieve 52
 - update 53
- remove URLs
 - insert 45
 - update 45
- request formats 10
- reset index
 - reset 42
 - retrieve status 42
- response formats 11
- resume crawl 30
- retrieve operations 6

S

- search appliance configuration
 - export 71
 - import 71
- search logs
 - create 63
 - delete 65
 - entry properties 61
 - listing 61
 - retrieve 63
 - update 64
- search report
 - create 58
 - delete 60
 - list 57
 - properties 56
 - retrieve 59
 - update 60
- searchLog feed 61, 63, 64, 65
- searchReport feed 56, 57, 58, 59, 60
- serving 43–55

- serving status, retrieve 55
- shut down a search appliance 74
- status and reports 56–73
- status feed 31, 55, 73
- suggest feed 55
- synonym feed 51
- system status, retrieve 73

T

- token, authentication 6
- trusted IP addresses 16

U

- update operations 6
- URL patterns
 - crawl 12
 - recrawl 24
- user name 6

X

- XML
 - elements 7–11
 - request formats 10
 - response formats 11
- XSLT stylesheet
 - retrieve 47
 - update 47