DSpace Statistics

DSpace 1.6 and newer versions uses the Apache SOLR application underlying the statistics. SOLR enables performant searching and adding to vast amounts of (usage) data.
Unlike previous versions, enabling statistics in DSpace does not require additional installation or customization. All the necessary software is included.

1 What is exactly being logged ?
- 1.1 Common stored fields for all usage events
- 1.2 Unique stored fields for bitstream downloads
- 1.3 Unique stored fields for search queries
- 1.4 Unique stored fields for workflow events

2 Web User Interface Elements
- 2.1 Pageview and Download statistics
  - 2.1.1 Home page
  - 2.1.2 Community home page
  - 2.1.3 Collection home page
  - 2.1.4 Item home page
- 2.2 Search Query Statistics
- 2.3 Workflow Event Statistics

3 Architecture
4 Configuration settings for Statistics
- 4.1 Pre-1.6 Statistics settings

5 Upgrade Process for Statistics
6 Statistics Administration
- 6.1 Converting older DSpace logs into SOLR usage data
- 6.2 Statistics Client Utility

7 Statistics differences between DSpace 1.7.x and 1.8.0
- 7.1 Displayed file statistics bundle configurable

8 Statistics differences between DSpace 1.6.x and 1.7.0
- 8.1 SOLR optimization added
- 8.2 SOLR Autocommit

9 Web UI Statistics Modification (XMLUI Only)
- 9.1 Modifying the number of months, for which statistics are displayed

10 Custom Reporting - Querying SOLR Directly
- 10.1 Resources
- 10.2 Examples
  - 10.2.1 Top downloaded items by a specific user

11 Manually Installing/Updating GeoLite Database File

What is exactly being logged ?

DSpace 1.6 and newer

After the introduction of the SOLR Statistics logging in DSpace 1.6, every pageview and file download is logged in a dedicated SOLR statistics core.

DSpace 3.0 and newer

In addition to the already existing logging of pageviews and downloads, DSpace 3.0 now also logs search queries users enter in the DSpace search dialog and workflow events.

JSP UI Search Query logging

Due to the very recent addition of Discovery for search & faceted browsing in JSPUI, these search queries are **not** yet logged. Regular (non-discovery) search queries **are** being logged in JSP UI.
Workflow Events logging

Only workflow events, initiated and executed by a physical user are being logged. Automated workflow steps or ingest procedures are currently **not** being logged by the workflow events logger.

The logging happens at the server side, and doesn't require a javascript like Google Analytics does, to provide usage data. Definition of which fields are to be stored happens in the file **dspace/solr/statistics/conf/schema.xml**.

Although they are stored in the same index, the stored fields for views, search queries and workflow events are different. A new field, statistics_type determines which kind of a usage event you are dealing with. The three possible values for this field are **view, search** and **workflow.**

```
<field name="statistics_type" type="string" indexed="true" stored="true" required="true" />
```

Common stored fields for all usage events

```
<field name="type" type="integer" indexed="true" stored="true" required="true" />
<field name="id" type="integer" indexed="true" stored="true" required="true" />
<field name="ip" type="string" indexed="true" stored="true" required="false" />
<field name="time" type="date" indexed="true" stored="true" required="true" />
<field name="epersonid" type="integer" indexed="true" stored="true" required="false" />
<field name="continent" type="string" indexed="true" stored="true" required="false"/>
<field name="country" type="string" indexed="true" stored="true" required="false"/>
<field name="countryCode" type="string" indexed="true" stored="true" required="false"/>
<field name="city" type="string" indexed="true" stored="true" required="false"/>
<field name="longitude" type="float" indexed="true" stored="true" required="false"/>
<field name="latitude" type="float" indexed="true" stored="true" required="false"/>
<field name="owningComm" type="integer" indexed="true" stored="true" required="false" multiValued="true"/>
<field name="owningColl" type="integer" indexed="true" stored="true" required="false" multiValued="true"/>
<field name="owningItem" type="integer" indexed="true" stored="true" required="false" multiValued="true"/>
<field name="dns" type="string" indexed="true" stored="true" required="false"/>
<field name="userAgent" type="string" indexed="true" stored="true" required="false"/>
<field name="isBot" type="boolean" indexed="true" stored="true" required="false"/>
<field name="referrer" type="string" indexed="true" stored="true" required="false"/>
<field name="uid" type="uuid" indexed="true" stored="true" default="NEW" />
<field name="statistics_type" type="string" indexed="true" stored="true" required="true" default="view" />
```

The combination of type and id determines which resource (either community, collection, item page or file download) has been requested.

Unique stored fields for bitstream downloads

```
<field name="bundleName" type="string" indexed="true" stored="true" required="false" multiValued="true" />
```

Unique stored fields for search queries

```
<field name="query" type="string" indexed="true" stored="true" required="false" multiValued="true"/>
<field name="scopeType" type="integer" indexed="true" stored="true" required="false" />
<field name="scopeId" type="integer" indexed="true" stored="true" required="false" />
<field name="rpp" type="integer" indexed="true" stored="true" required="false" />
<field name="sortBy" type="string" indexed="true" stored="true" required="false" />
<field name="sortOrder" type="string" indexed="true" stored="true" required="false" />
<field name="page" type="integer" indexed="true" stored="true" required="false" />
```

Unique stored fields for workflow events

```
<field name="workflowStep" type="string" indexed="true" stored="true" required="false" multiValued="true"/>
<field name="previousWorkflowStep" type="string" indexed="true" stored="true" required="false" multiValued="
true"/>
<field name="owner" type="string" indexed="true" stored="true" required="false" multiValued="true"/>
<field name="submitter" type="integer" indexed="true" stored="true" required="false" />
<field name="actor" type="integer" indexed="true" stored="true" required="false" />
<field name="workflowItemId" type="integer" indexed="true" stored="true" required="false" />
```

Web User Interface Elements

Pageview and Download statistics

In the XMLUI, pageview and download statistics can be accessed from the lower end of the navigation menu. In the JSPUI, a view statistics button appears on the bottom of pages for which statistics are available.

If you are not seeing these links or buttons, it's likely that they are only enabled for administrators in your installation. Change the configuration parameter "authorization.admin.usage" in usage-statistics.cfg to false in order to make statistics visible for all repository visitors.

Home page

Starting from the repository homepage, the statistics page displays the top 10 most popular items of the entire repository.

Community home page

The following statistics are available for the community home pages:

- Total visits of the current community home page
- Visits of the community home page over a timespan of the last 7 months
- Top 10 country from where the visits originate
- Top 10 cities from where the visits originate

Collection home page

The following statistics are available for the collection home pages:

- Total visits of the current collection home page
- Visits of the collection home over a timespan of the last 7 months
- Top 10 country from where the visits originate
- Top 10 cities from where the visits originate

Item home page

The following statistics are available for the item home pages:

- Total visits of the item
- Total visits for the bitstreams attached to the item
- Visits of the item over a timespan of the last 7 months
- Top 10 country views from where the visits originate
- Top 10 cities from where the visits originate

Search Query Statistics

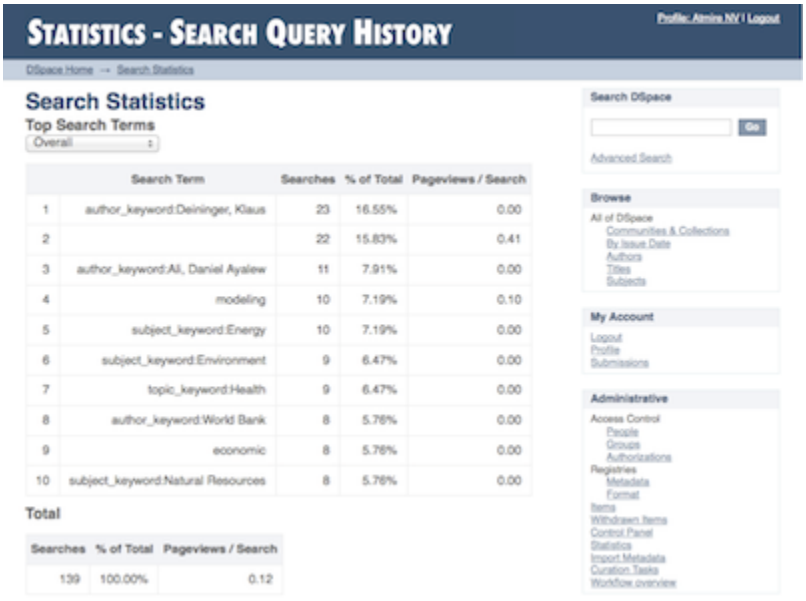
In the XMLUI, search query statistics can be accessed from the lower end of the navigation menu.

If you are not seeing the link labelled "search statistics", it is likely that they are only enabled for administrators in your installation. Change the configuration parameter "authorization.admin.search" in usage-statistics.cfg to false in order to make statistics visible for all repository visitors.

The dropdown on top of the page allows you to modify the time frame for the displayed statistics.

The Pageviews/Search column tracks the amount of pages visited after a particular search term. Therefor a zero in this column means that after executing a search for a specific keyword, not a single user has clicked a single result in the list.

If you are using Discovery, note that clicking the facets also counts as a search, because clicking a facet sends a search query to the Discovery index.

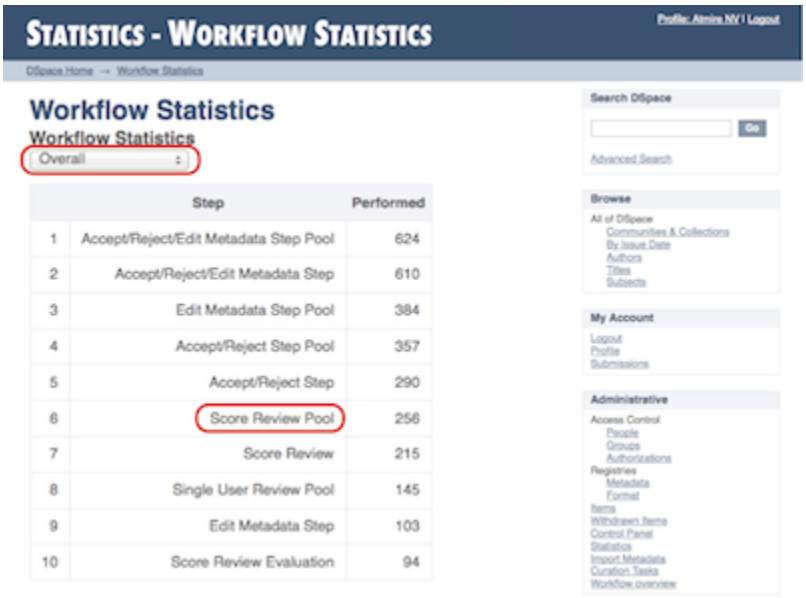


Workflow Event Statistics

In the XMLUI, search query statistics can be accessed from the lower end of the navigation menu.

If you are not seeing the link labelled "Workflow statistics", it is likely that they are only enabled for administrators in your installation. Change the configuration parameter "authorization.admin.workflow" in usage-statistics.cfg to false in order to make statistics visible for all repository visitors.

The dropdown on top of the page allows you to modify the time frame for the displayed statistics.



Architecture

The DSpace Statistics Implementation is a Client/Server architecture based on Solr for collecting usage events in the JSPUI and XMLUI user interface applications of DSpace. Solr runs as a separate webapplication and an instance of Apache Http Client is utilized to allow parallel requests to log statistics events into this Solr instance.

Configuration settings for Statistics

In the {dspace.dir}/config/modules/**solr-statistics**.cfg file review the following fields to make sure they are uncommented:

Property:	**server**
Example Values:	server = http://127.0.0.1/solr/statistics server = ${solr.server}/statistics
Informational Note:	Is used by the SolrLogger Client class to connect to the Solr server over http and perform updates and queries. In most cases, this can (and should) be set to localhost (or 127.0.0.1). To determine the correct path, you can use a tool like `wget` to see where Solr is responding on your server. For example, you'd want to send a query to Solr like the following: `wget http://127.0.0.1/solr/statistics/select?q=*:*` Assuming you get an HTTP 200 OK response, then you should set `solr.log.server` to the '/statistics' URL of 'http://127.0.0.1/solr/statistics' (essentially removing the "/select?q=:" query off the end of the responding URL.)
Property:	**query.filter.bundles**
Example Value:	query.filter.bundles=ORIGINAL
Informational Note:	A comma seperated list that contains the bundles for which the file statistics will be displayed.
Property:	**solr.statistics.query.filter.spiderIp**
Example Value:	solr.statistics.query.filter.spiderIp = false
Informational Note:	If true, statistics queries will filter out spider IPs -- use with caution, as this often results in extremely long query strings.

Property:	**solr.statistics.query.filter.isBot**
Example Value:	solr.statistics.query.filter.isBot = true
Informational Note:	If true, statistics queries will filter out events flagged with the "isBot" field. This is the recommended method of filtering spiders from statistics.
Property:	**spiderips.urls**

Example Value:	spiderips.urls =
	`http://iplists.com/google.txt, \`
	`http://iplists.com/inktomi.txt, \`
	`http://iplists.com/lycos.txt, \`
	`http://iplists.com/infoseek.txt, \`
	`http://iplists.com/altavista.txt, \`
	`http://iplists.com/excite.txt, \`
	`http://iplists.com/misc.txt, \`
	`http://iplists.com/non_engines.txt`

Informational Note:	List of URLs to download spiders files into [dspace]/config/spiders. These files contain lists of known spider IPs and are utilized by the SolrLogger to flag usage events with an "isBot" field, or ignore them entirely. The "stats-util" command can be used to force an update of spider files, regenerate "isBot" fields on indexed events, and delete spiders from the index. For usage, run: `dspace stats-util -h` from your [dspace]/bin directory

In the {dspace.dir}/config/modules/**usage-statistics**.cfg file review the following fields to make sure they are uncommented:

Property:	**dbfile**
Example Value:	dbfile = ${dspace.dir}/config/GeoLiteCity.dat
Informational Note:	The following referes to the GeoLiteCity database file utilized by the LocationUtils to calculate the location of client requests based on IP address. During the Ant build process (both fresh_install and update) this file will be downloaded from http://www.maxmind.com/app/geolitecity if a new version has been published or it is absent from your [dspace]/config directory.
Property:	**resolver.timeout**
Example Value:	resolver.timeout = 200
Informational Note:	Timeout in milliseconds for DNS resolution of origin hosts/IPs. Setting this value too high may result in solr exhausting your connection pool.
Property:	**useProxies**
Example Value:	useProxies = true

Informational Note:	Will cause Statistics logging to look for X-Forward URI to detect clients IP that have accessed it through a Proxy service (e.g. the Apache mod_proxy). Allows detection of client IP when accessing DSpace. [Note: This setting is found in the DSpace Logging section of dspace.cfg]
Property:	**authorization.admin.usage**
Example Value:	authorization.admin.usage = true
Informational Note:	When set to true, only general administrators, collection and community administrators are able to access the pageview and download statistics from the web user interface. As a result, the links to access statistics are hidden for non logged-in admin users. Setting this property to "false" will display the links to access statistics to anyone, making them publicly available.
Property:	**authorization.admin.search**
Example Value:	authorization.admin.search = true
Informational Note:	When set to true, only system, collection or community administrators are able to access statistics on search queries.
Property:	**authorization.admin.workflow**
Example Value:	authorization.admin.workflow = true
Informational Note:	When set to true, only system, collection or community administrators are able to access statistics on workflow events.
Property:	**logBots**
Example Value:	logBots = true
Informational Note:	When this property is set to false, and IP is detected as a spider, the event is not logged. When this property is set to true, the event will be logged with the "isBot" field set to true. (see solr.statistics.query.filter.* for query filter options)

Pre-1.6 Statistics settings

Older versions of DSpace featured static reports generated from the log files. They still persist in DSpace today but are completely independent from the SOLR based statistics.
The following configuration parameters applicable to these reports can be found in dspace.cfg.

```
###### Statistical Report Configuration Settings ######

 # should the stats be publicly available?  should be set to false if you only
 # want administrators to access the stats, or you do not intend to generate
 # any
 report.public = false

 # directory where live reports are stored
 report.dir = ${dspace.dir}/reports/
```

These fields are not used by the new 1.6 Statistics, but are only related to the Statistics from previous DSpace releases

Upgrade Process for Statistics

Example of rebuild and redeploy DSpace (only if you have configured your distribution in this manner)

First approach the traditional DSpace build process for updating

```
cd [dspace-source]/dspace
 mvn package
 cd [dspace-source]/dspace/target/dspace-<version>-build.dir
 ant -Dconfig=[dspace]/config/dspace.cfg update
 cp -R [dspace]/webapps/* [TOMCAT]/webapps
```

The last step is only used if you do not follow the recommended practice of configuring *[dspace]/webapps* as location for webapps in your servlet container (Tomcat, Resin or Jetty). If you only need to build the statistics, and don't make any changes to other web applications, you can replace the copy step above with:

```
cp -R dspace/webapps/solr TOMCAT/webapps
```

Again, only if you are not mounting [dspace]/webapps directly into your Tomcat, Resin or Jetty host (the recommended practice)

Restart your webapps (Tomcat/Jetty/Resin)

Statistics Administration

Converting older DSpace logs into SOLR usage data

If you have upgraded from a previous version of DSpace, converting older log files ensures that you carry over older usage stats from before the upgrade.

Statistics Client Utility

The command line interface (CLI) scripts can be used to clean the usage database from additional spider traffic and other maintenance tasks. In DSpace 3.0, a script has been added to split up the monolithic SOLR core into individual cores each containing a year of statistics.

Statistics differences between DSpace 1.7.x and 1.8.0

Displayed file statistics bundle configurable

In DSpace 1.6.x & 1.7.x the file download statistics were generated without regard to the bundle in which the file was located. In DSpace 1.8.0 it is possible to configure the bundles for which the file statistics are to be shown by using the **query.filter.bundles** property. If required the old file statistics can also be upgraded to include the bundle name so that the old file statistics are fixed.

Backup Your statistics data first

Applying this change will involve dumping all the old file statistics into a file and re uploading these. Therefore it is wise to create a backup of the {dspace.dir}/solr/statistics/data directory. It is best to create this backup when the Tomcat/Jetty/Resin server program isn't running.

When a backup has been made start the Tomcat/Jetty/Resin server program.
The update script has one optional command which will if given not only update the broken file statistics but also delete file statistics for files that where removed from the system (if this option isn't active these statistics will receive the "BITSTREAM_DELETED" bundle name).

```
#The -r is optional
[dspace]/bin/dspace stats-util -b -r
```

Statistics differences between DSpace 1.6.x and 1.7.0

SOLR optimization added

If required, the solr server can be optimized by running

```
{dspace.dir}/bin/stats-util -o
```

More information on how these solr server optimizations work can be found here: http://wiki.apache.org/solr/SolrPerformanceFactors#Optimization_Considerations.

SOLR Autocommit

In DSpace 1.6.x, each solr event was committed to the solr server individually. For high load DSpace installations, this would result in a huge load of small solr commits resulting in a very high load on the solr server.
This has been resolved in dspace 1.7 by only committing usage events to the solr server every 15 minutes. This will result in a delay of the storage of a usage event of maximum 15 minutes. If required, this value can be altered by changing the maxTime property in the

```
{dspace.dir}/solr/statistics/conf/solrconfig.xml
```

Web UI Statistics Modification (XMLUI Only)

Modifying the number of months, for which statistics are displayed

Modify line 205 in the StatisticsTransformer.java file

https://github.com/DSpace/DSpace/blob/dspace-3_x/dspace-xmlui/src/main/java/org/dspace/app/xmlui/aspect/statistics/StatisticsTransformer.java#L205

-6 is the default setting, displaying the past 6 months of statistics. When reducing this to a smaller natural number, less months are being displayed.

Related: DatasetTimeGenerator Javadoc

Custom Reporting - Querying SOLR Directly

When the web user interface does not offer you the statistics you need, you can greatly expand the reports by querying the SOLR index directly.

Resources

- http://www.lucidimagination.com/Community/Hear-from-the-Experts/Articles/Faceted-Search-Solr
- http://my.safaribooksonline.com/9781847195883/Cover

Examples

Top downloaded items by a specific user

Query:

```
http://localhost:8080/solr/statistics/select?indent=on&version=2.2&start=0&rows=10&fl=*%
2Cscore&qt=standard&wt=standard&explainOther=&hl.fl=&facet=true&facet.field=epersonid&q=type:0
```

Explained:

facet.field=epersonid — You want to group by epersonid, which is the user id.
type:0 — Interested in bitstreams only

```
<lst name="facet_counts">
    <lst name="facet_fields">
        <lst name="epersonid">
            <int name="66">1167</int>

<int name="117">251</int>

<int name="52">42</int>

<int name="19">36</int>

<int name="88">20</int>

<int name="112">18</int>

<int name="110">9</int>

<int name="96">0</int>

</lst>
    </lst>
</lst>
```

Manually Installing/Updating GeoLite Database File

The GeoLite Database file (at [dspace]/config/GeoLiteCity.dat) is used by the Statistics engine to generate location/country based reports. (*Note: If you are not using DSpace Statistics, this file is not needed.*)

In most cases, this file is installed automatically when you run `ant fresh_install`. However, if the file cannot be downloaded & installed automatically, you may need to manually install it.

As this file is also sometimes updated by MaxMind.com, you may also wish to update it on occasion.

You have two options to install/update this file:

1. Attempt to re-run the automatic installer from your DSpace Source Directory ([dspace-source]). This will attempt to automatically download the database file, unzip it and install it into the proper location:

   ```
   ant update_geolite
   ```

   - NOTE: If the location of the GeoLite Database file is known to have changed, you can also run this auto-installer by passing it the new URL of the GeoLite Database File: `ant -Dgeolite=[full-URL-of-geolite] update_geolite`
2. OR, you can manually install the file by performing these steps yourself:
   - First, download the latest GeoLite Database file from http://geolite.maxmind.com/download/geoip/database/GeoLiteCity.dat.gz
   - Next, unzip that file to create a file named GeoLiteCity.dat
   - Finally, move or copy that file to your DSpace installation, so that it is located at `[dspace]/config/GeoLiteCity.dat`.