

Benchmark Statistics project (BEST)



THE AUSTRALIAN NATIONAL UNIVERSITY

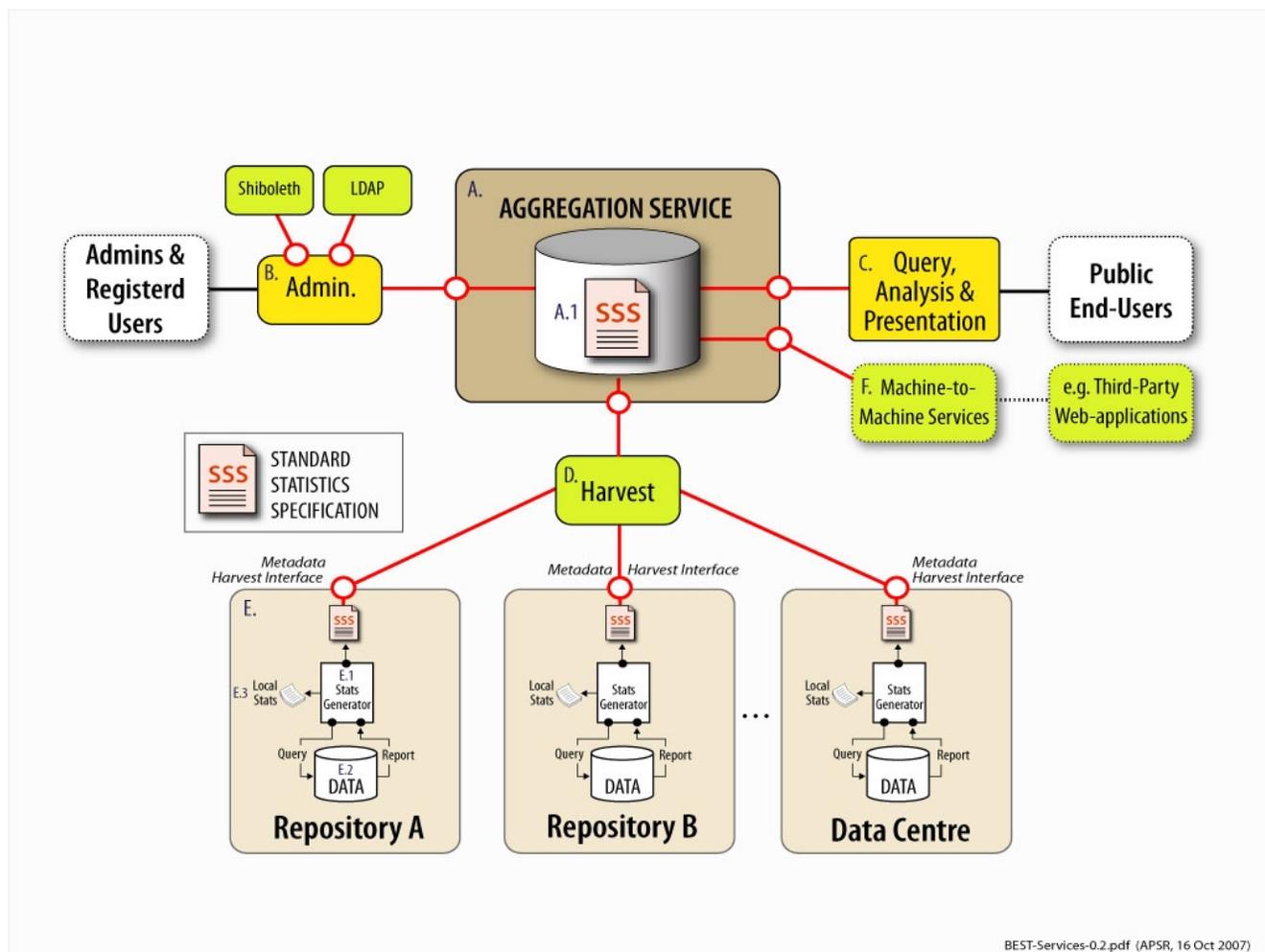
Table of Contents

Benchmark Statistics project (BEST).....	1
Overview.....	2
Functional Specification.....	2
Data-Source Management	4
Report Viewer:.....	4
Reports:.....	5
Technical Overview.....	7
System requirements.....	7
Server components.....	7
File Structure.....	7
Database.....	8
Possible Future improvements/changes:.....	8

Overview

The Benchmark Statistics project (BEST) is developed by APSR. BEST was written to be as a components of the COSI framework. It is a PHP/PostgreSQL web-application. Primary purpose of the application is to facilitate the harvest, aggregation, analysis and presentation of repository content and usage data. It is a pilot project which intended to serve to demonstrate the possibilities a system such as BEST can provide for the repository communities.

Functional Specification



BEST requires the following system components:

- | | |
|---|--|
| | User managements |
| | Access Control |
| COSI framework | Navigation |
| | Database Management |
| Harvester: | Incremental data collection and delivery |
| repositories with Metadata Harvest Interface: | Deliver EIM packets via OAI-PMH protocol |
| | Repository components Statistical data collection and management |

EIM Schema

The EIM (Event Interchange Model) Schema should be used to provide statistical and Metadata information by the repositories. This data should be provided using OAI protocol. The Schema specification can be found at http://www.apsr.edu.au/downloads/eim_schema_guidelines.pdf

Data-Source Management

Datasource management and harvester configuration is identical to the ORCA Registry's

When user is logged on with access privileges BEST_SOURCE_ADMIN user will have the ability to add delete modify data sources in the same fashion as it's described in the ORCA user manual.

The screenshot displays a web interface for managing data sources. On the left is a navigation menu with options like 'Logout', 'Administration', and 'View Data Source'. The main content area is titled 'Data Source' and shows details for a source with 117 records. The details include a key, title, URI, provider type, harvest method, date, frequency, contact information, and creation/modification timestamps. At the bottom, there is an 'Activity Log' section showing a recent successful harvest request.

Logged in as: Leo Mor

Data Source

Records From Source: 117

Key: sts59142.anu.edu.au

Title: Leo's DSpace

URI: http://150.203.59.142:8080/dspace-oai/request

Provider Type: OAI-PMH

Harvest Method: Harvester OAI-PMH

Harvest Date: 2008-06-11 13:15

Harvest Frequency: hourly

OAI-PMH Set:

Contact Name: Leo Monus

Contact E-mail: leo.monus@anu.edu.au

Notes: dgsdgsdg

Record Owner:

Created When: 2008-06-11T10:39:14+1000

Created Who: Leo Monus (leo.monus@anu.edu.au)

Modified When: 2008-06-17T12:58:55+1000

Modified Who: Leo Monus (leo.monus@anu.edu.au)

Harvest Requests: ▶ Status: Scheduled for 2008-06-25T04:15:00Z

Activity Log:

[Plain text Activity Log](#)

```
2008-06-25 13:39:28 130.56.60.111 APSRHarvester/1.0
PUT HARVEST DATA
Request IP: 130.56.60.111
Harvest Request ID: A1EA56211387DF7D4123106410C56AF64AA70748
Harvester IP: 130.56.60.111
Mode: HARVEST
Done: TRUE
Time Taken: 0.086 seconds
>>SUCCESS
```

Report Viewer:

There are 11 specific reports generated by the current system. Each of them were suggestions of the BEST reference group.

These reports can be viewed in 3 formats chart, table or raw XML where appropriate.

The graphical presentation makes the use of the Google chart API service, which was the most suited for BEST's purpose. The service is free and reliable so it doesn't look like an external dependency which may causes problems in the future. If in an unlikely event, that this service goes off-line the users still be able to view their data using the tabular presentation

- Logout
- Change Built-in Passphrase
- ▷ Administration
- ▷ Collections Registry
- ▼ **Benchmark Statistics**
 - ◆ **Reports**
 - Browse
 - Web Services
 - ▼ Administration
 - List Data Sources
 - Add Data Source

BEST PILOT TOP TEN REPORTS

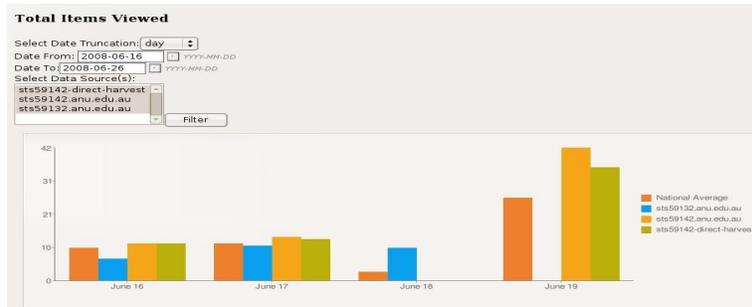
1 Origin Of Visitors (country City)	 XML 
2 Total Unique Visitors	 XML 
3 Total Items Viewed	 XML 
4 Total Items Retrieved	 XML 
5 New Additions by Time	 XML 
6 Top 'N' Viewed Items and Authors	 XML
7 Top 'N' Retrieved Items and Authors	 XML
8 Top 'N' Items and Authors	 XML
9 Total Items by Subject	 XML 
10 Total Items by Resource type	 XML 
11 Total Items by File Format	 XML 

Reports:

- 1. Origin of Visitors**
the sum of each visitors by country for each data source
also can be presented as a 'heat map'
- 2. Total Unique Visitors**
the number of unique visitors per datasource

3. Total Items Viewed

a time based presentation of items viewed per datasources presented side by side with the average. This query can be refined by using from and to dates and/or modifying the date truncation which is set to weeks by default.



4. Total Items Retrieved

same as the previous report accept this one present the items which were retrieved the graphical presentation used is line chart for no other reason but to show future developers how to use the Google chart API and how to modify chart type for each reports



5. New Additions by Time

same as the previous report accept this one present the items which were retrieved

6. Top N Viewed Items and Authors

displays the most viewed Items and Authors from the selected datasources

7. Top N Retrieved Items and Authors

displays the most retrieved Items and Authors from the selected datasources

8. Top N Items and Authors

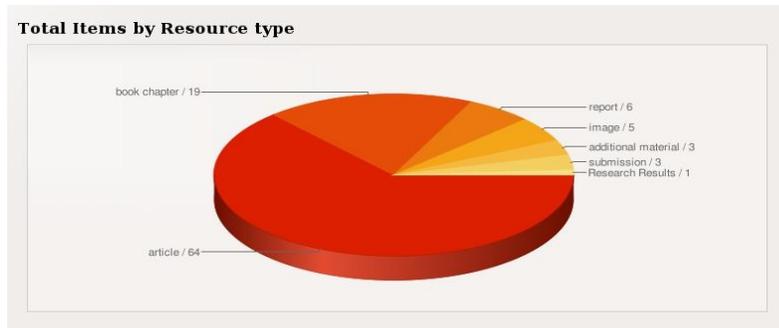
displays the most accessed (practically the aggregated version of the previous two reports) Items and Authors from the selected datasources

9. Total Items by Subject

presents an overview of the type of Items by subject the datasources contain.

10. Total Items by Resource type

presents an overview of the type of Items by Resource Type the datasources contain.



11. Total items by file format

presents an overview of the type of Items by file format the datasources contain.

Technical Overview

System requirements

The BEST is a PHP/PostgreSQL web application built to utilise the COSI-Framework (an install of which is a prerequisite). System and web browser requirements are the same as those for the COSI-Framework (see the COSI-Framework documentation for more information).

Server components

File Structure

The BEST application file structure is similar to that of the COSI-Framework within which it is housed. Activities are defined in files at the application root, and within folders that are not prefixed with an underscore ('_'). Folders that are prefixed with an underscore are used to hold function libraries, help content etc. Function library files are prefixed with 'best' to prevent name clashes with other files included in the COSI-Framework.

The file *best_init.php* at the root of the application contains the environment settings for an installed instance of the ORCA-Registry—consisting of the location of the schemata that are to be used by the instance.

The BEST application defines some Cascading Stylesheet (CSS) styles for its own use in *best.css*. These are included into the response in *best_init.php* via the COSI-Framework API.

The installation process makes changes to the *application_config.php* and *database_env.php* files in the housing COSI-Framework.

Database

The registry database (named *dbs_best*) contains 11 tables, and many user-defined functions. All access to data is via parameterised calls to user-defined functions.

tbl_data_sources	Repository (datasource) information
tbl_data_source_logs	BEST transaction logs
tbl_collections	Item containers
tbl_collections_to_item	Mapping between Item and collection
tbl_item	Item internal and external (eg handles) ids
tbl_metadatavalue	Metadata values
tbl_metadatablefieldregistry	DC or MODS mapping to specific metadata values
tbl_events	Item events EIM
tbl_ip_to_country	Decimal IP range per country table (no City or region data is available as of yet)
tbl_harvest_requests	Harvester information
tbl_keys	Table keys

Possible Future improvements/changes:

- Current system provides anonymous access to all reports might consider controlling access to statistics via shibboleth login
- Since all database access is done by user defined functions adding new or modifying existing reports isn't simple and requires code changes in several places. May provide a generic way to write user defined reports.
- Accumulated stats can grow to very large volume. Might need to summarise them periodically.