## Multiple Application XBRL Delivery System (MAXDS):

New Features:

* Recommence work on Solr to Elasticsearch conversion, resolve connection issues.
* Work on performance comparison between Elasticsearch and Solr
* Research on search tuning, filtering and other features provided by Elasticsearch engine.
* Created customized Elasticsearch query logic in ReferenceDataService to replace current Solr query for LookupDoc.
* Tested customized Elasticsearch queries in ReferenceDataService.
* Updated pom.xml dependencies to make sure existing MAXDS dependencies work properly with Elasticsearch spring jars.
* Uploaded documents to Elasticsearch deployment for following index (completed):
* Term result
* Processing log
* Accuracy test notes
* Entity
* LookupDoc
* TermRule
* NormalizedFact
* TermMapInformation
* Designed and implemented package and configuration structure in current MAXDS for Elasticsearch module, which will replace the Solr module.
* Created Elasticsearch package with following components implemented:
* Elasticsearch configurations.
* Elasticsearch document object – LookupDoc.
* Elasticsearch repository object – LookupDocRepository.
* Elasticsearch service object – ReferenceDataService.
* Current Solr modules need to be migrated, including document, repository, customized query, service, and configuration packages:
* Lookup References, Normalized Facts, Term Results, AccuracyTestNotes, ProcessingLog, TermMapInformation, TermRule.
* TermRule repository and service module.
* TermMapInformation repository and service module.
* AccuracyTestNotes document, repository, and service modules.
* NormalizedFact document, repository, and service modules.
* ProcessingLog document, repository, and service modules.
* TermResults document, repository, and service modules.
* Elasticsearch queries/APIs to replace corresponding Solr queries for following modules.
* TermRule.
* TermMapInformation.
* AccuracyTestNotes.
* NormalizedFact.
* ProcessingLog.
* TermResults.
* Performed MAXDS local deployment and worked on fixing issue
* Resolved dependency jar files issue with Elasticsearch v7 in local deployment. (Elasticsearch works with newer version of java third party jars whereas current MAXDS references older version. In progress.).
* Merged Elasticsearch configuration with existing MAXDS Solr configurations, and update package settings and component scan settings.
* Implemented test controller for Elasticsearch API.
* Implemented and test Elasticsearch module on MAXDS and resolve issues in following components :
* Elasticsearch query implementation and testing for “Go To Peer Group” (controller, service, and repository classes).
* Elasticsearch query implementation and testing for “Show Resolved” (controller, service and repository classes).
* Converted Solr pivot facet queries to Elasticsearch custom queries (includes converting @Facet, @Pivot, and FacetPage. :
* TermResultDocRepository.
* TermResultService.
* Enabled Elasticsearch module on MAXDS login page and resolved issues in following components
* Elasticsearch entity modules (document, service, and repository classes).
* Elasticsearch TermRule modules (document, service and repository classes).
* Enabled Elasticsearch module on MAXDS Home page and resolved issues in following components:
* Elasticsearch entity modules (document, service, and repository classes).
* Elasticsearch TermRule modules (document, service and repository classes).
* Elasticsearch TermResult modules (document, service and repository classes).

Bug Fixes:

* Log4j2 v2.15.0 migration for MAXDS:
* Updated pom.xml dependencies.
* Modified all java module where log4j old version is used.
* Modified original log4j.properties for new log framework.

Configuration and Software Changes:

* Update LDAP connection string for infrastructure updates.

**Forecasted Items**

## Multiple Application XBRL Delivery System (MAXDS):

Recommence effort on AWS Elasticsearch implementation for MAXDS features.

Support AWS S3 bucket and logstash configuration and setup.

Provide support for MAXDS new features, enhancements, and issue fixes.

Implement corresponding Solr queries in Elasticsearch engine.

Utilize Elasticsearch tuning and analysis tools for query performance analysis.

Perform improvements to UI and business functionality.

Create automated test cases.

Perform upgrade to AWS cloud.

Work on QCTC fixes.