SAND2015-2723C

Leveraging open source solutions for Enterprise Search

John Herzer
Sandia National Laboratories

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Bio

- John Herzer has served in a number of positions at Sandia Labs over nearly 30 years of service. In prior projects, he helped implement the first workflow system at the lab and managed the migration of over a million documents from WebFileShare to FileNet. John is currently the project lead for the enterprise search team.

Open Source Search Drivers

- Used same proprietary search engine for years
- Adding pages or removing pages from the index was a mysterious process
- Technical support a cyclical process of making config changes and sending a set of logs
- Needed more visibility into the inner workings of the search engine

Comparison of Open source solutions

- Hired consultants to compare Solr and ElasticSearch
- Found that Solr was easier to setup and configure
- ElasticSearch has better metrics, monitoring tools
- Solr is more open and has larger community, but ElasticSearch is growing faster
- Good comparisons at
  - http://thinkbig.teradata.com/solr-vs-elastic-search/
  - http://www.datanami.com/2015/01/22/solr-elasticsearch-question/

Solr Schema definition

- The schema defines the fields that a document can contain and how to handle them
- Default types: text, string, date, long …
- Field options: indexed, stored, termVectors …
- Functionality and performance are dependent on an appropriate schema definition
- Dynamic schemas are a way to add new fields after the schema has already been defined

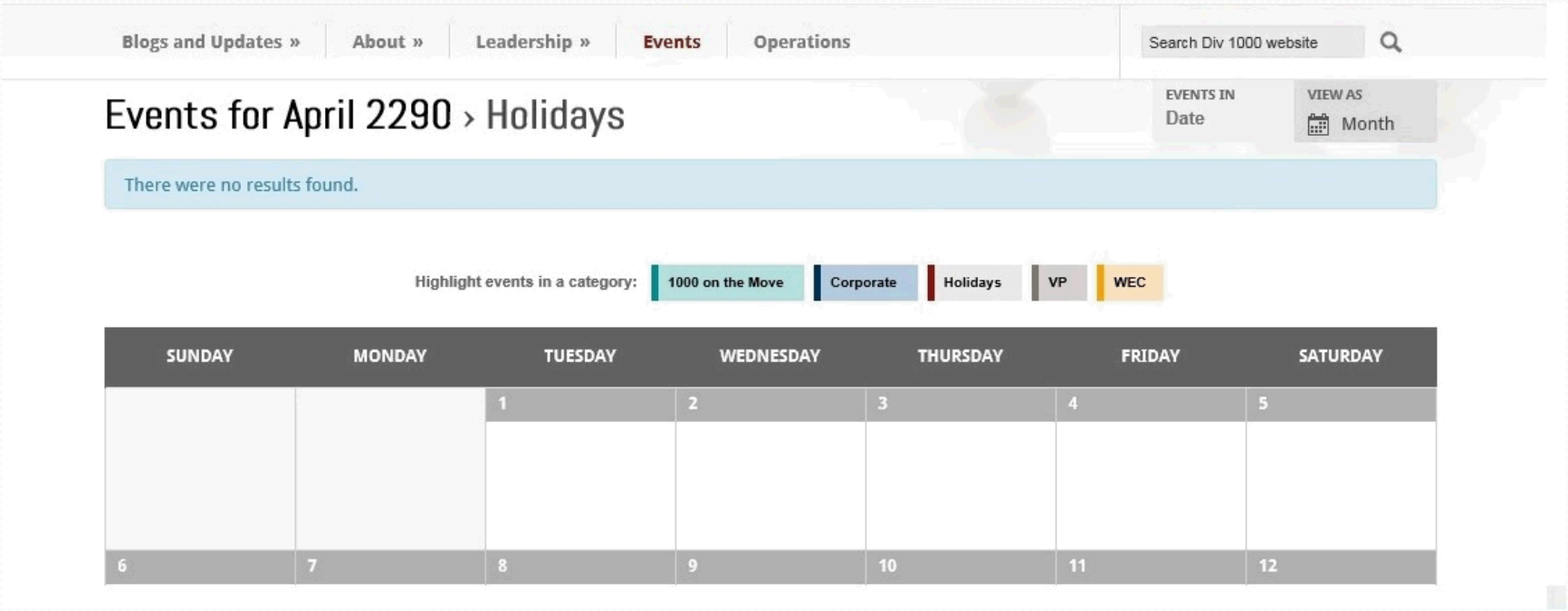
Solr Configuration

- Java Virtual Machine configuration is critical to good performance
- Avoid full garbage collection; use G1GC on Java 7+
- Make use of the Solr caches
- Use load testing to validate configuration changes
- Can scale with SolrCloud if needed

Crawler Creation

- Apache Nutch is commonly used with Solr but we had trouble accessing SharePoint with it
- Ended up writing our own crawler
  - Uses Jsoup and Tika for parsing. Can be configured to crawl different repositories via property files
  - Can specify initial URLs, stop words, forbidden URLs, etc.
  - Stops itself after too many errors or lost Solr connection
  - When stopped manually, it serializes its state so that it can start up at the point where it left off

Crawler Care and feeding



Future enhancements

- Application server cache
- Faceting/filtering
- Ranking by age of document
- More federation
  - Currently federating FileNet and Primo (Journals)
  - Prototype federation for Sharepoint, benefits site
  - Challenges of federation include slow sites, ranking of combined results, document age calculation