2015

Searchblox Project Indexer agent

Tri Nguyen
Virginia Commonwealth University

Tom Vaele
Virginia Commonwealth University

Zachary Johnson
Virginia Commonwealth University

Team Members: Tom Vaele, Tri Nguyen, Zachary Johnson

Faculty Advisor: Dr. Preetam Ghosh

Sponsor: SearchBlox

Sponsor Advisor: Timo Selvaraj

COMPUTER SCIENCE

Searchblox Project
Indexer agent

CAPSTONE DESIGN EXPO 2015

Mission

Problem Definition

- SearchBlox offers a great toolkit to maintain indexes on a local file system of a variety of MIME types but not to external ones.
- Generating the necessary data members for efficient search through indexes.
- How can these indexes of the exterior file systems remain up to date and how can we return to these files?

Core Requirements

- Configurable Crawler
- Parse Meta-Data Fields & Content
- Post to Remote SearchBlox RESTful API
- React to Changes in Native File Systems
- Terabyte Scale
- Concurrent

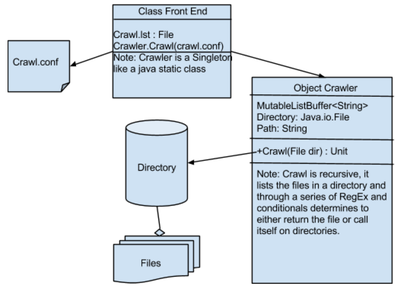
Technical Requirements

- Build on the TypeSafe Activator Platform
- Use the Scala Programming Language
- Configurable & Data Driven
- Documentation & Packaging
- Install Script

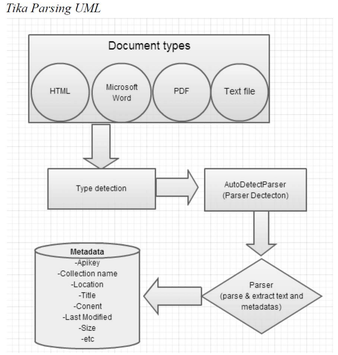
Procedure

7. Module Level Data Flow Diagrams

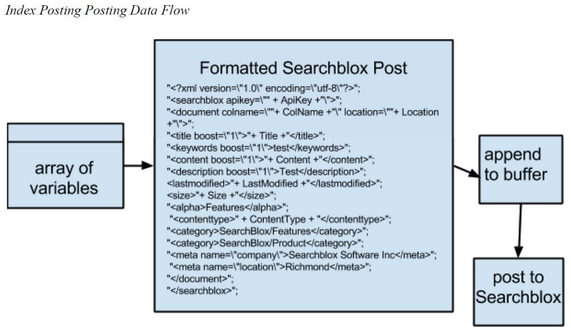
Crawler Data Flow

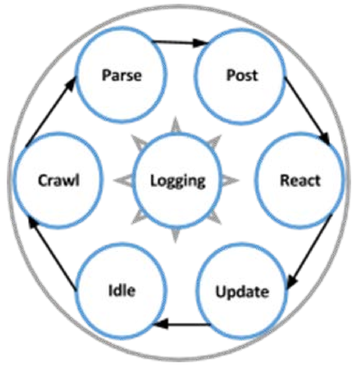


Tika Parsing UML



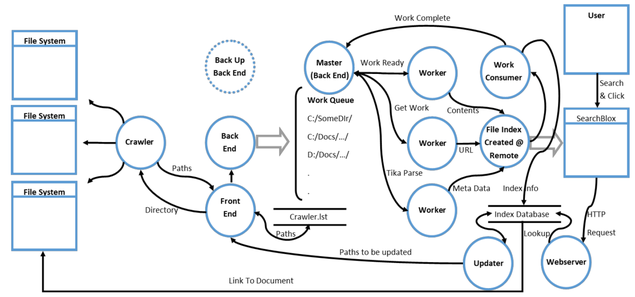
Index Posting Posting Data Flow





Results

Context Level Data Flow Diagram



- Deployed the prototype from the first semester to a non-local server
- We created a file server to respond to http GET requests when trying to re-acess the documents from the SearchBlox server. We did this using Node.js.
- Created our front end to configure the program to choose which documents to select and which server to post to by interfacing with the Node.js RESTful API.
- Currently The bottleneck is in HTTP posts to the indexing server.