LuceneConnectorFrameworkProposal

Lucene Connector Framework

Abstract

Many, many search engines, as well as other applications, have a need to connect with content repositories (SharePoint, CMS, Documentum, etc.) in a standard manner. The Lucene Connector Framework (LCF) is a project aimed at building out these connectors in open source under the Apache brand.

Proposal

The goal of LCF is to create a viable Lucene subproject aimed at delivering a best of breed connector framework under the Apache Lucene name. As a framework, the project will not only provide a way to connect to individual repositories, but also a mechanism for plugging in new connectors or custom connectors in a straightforward manner.

A connector framework is vital for search engines and other tools that need to access data located in corporate repositories. By abstracting the problem into a framework, applications can code to a set of well-defined interfaces instead of having to use a different interface for each connector.

Connector Framework is an extendible incremental crawler, which uses a database to manage configuration and crawl history, and provides reasonably high performance in accessing content in multiple repositories for the main purpose of search engine indexing. Connector Framework also establishes a repository-specific security model which can be used to limit search user access to repository content based on a user's identity. Connector Framework also includes existing connectors and authorities for:

- File system
- Windows shares
- JDBC-supported databases
- RSS feeds
- General websites
- [LiveLink] [from OpenText]
- Documentum [from EMC]
- [SharePoint] [from Microsoft]
- Meridio [from Meridio]
- Memex [from Memex]
- [FileNet] [from IBM]

Key design points for Connector Framework are as follows:

- Extendability - you can add new connectors for new repositories, and new authorities for specific repository security models
- Incrementality - the ability to process only what changed between crawls, in a repository-specific manner
- Restartability - using a database with ACID properties to insure that crawls are safe against process interruption or machine shutdown
- Security - establishing a model of security tokens that allows a search engine to enforce a repository's security model
- Limited footprint - ability to operate reliably within a fixed amount of process memory, regardless of configuration
- Performance - management of connector-specific resources to maximize overall thoughput
- Transparency - ability to generate reports on the activity of all crawls and repository connections

Background

MetaCarta originally approached Grant Ingersoll from the Lucene PMC about donating their existing connector framework to the Lucene PMC. After some discussion about accepting it as a software grant, the PMC decided it would be best to incubate the project first.

Rationale

The Connector Framework fills an often significant gap in the Lucene experience, namely, how to get content locked away in a content repository into Lucene/Solr/Nutch/Mahout/Tika. Naturally, many other tools (search engines and others) will also have this same problem. A Connector Framework would also be useful for someone wishing to migrate between content repositories, too.

Current Status

Connector Framework has been under development and in use in the field for close to five years, deployed on a MetaCarta search appliance. Almost all development of the project has been done by Karl Wright (kwright@metacarta.com). Some individual connectors were developed initially by contractors hired by MetaCarta, Inc., but maintenance and further development is currently handled by the MetaCarta team.

Development of Connector Framework can therefore be viewed as core framework development, plus development of individual connectors. Core framework development is currently not a terribly collaborative process, as there are no maintainers of the core functionality other than Mr. Wright. Development of new connectors has been done in the past in a much more collaborative way by supplying a developer with a "development kit", and then integrating the resulting connector (with whatever changes might have been necessary) into the source tree.

Reasonable efforts have been made to maintain the generality of the code base during the time that MetaCarta has owned it. Nevertheless, certain MetaCarta-specific changes have been made which may require review and modification. The following areas probably need to be addressed in the code before graduation can occur:

1. Branding. The UI brands it as a MetaCarta project.
2. Package names. Package names would have to be changed.
3. How Connector Framework handles document delivery needs to be generalized, at least for a single, configurable target output connector, and perhaps for multiple, independently-configurable targets. Simple example output connectors need to be written. Work in this direction is currently underway at MetaCarta and may or may not be complete at the time of the code handover.

4. Connector Framework-specific dependent package modifications need to be addressed somehow. For instance, the following projects that Connector Framework depends upon have been modified, but the modifications have not been accepted upstream: commons-httpclient NTLMv2 and NTLM2 support [RSS, Web, SharePoint, Meridio, and Livelink connectors]; commons-httpclient custom HTTPS protocol factory support [Web, SharePoint, Meridio, and Livelink connectors]; xerces ability to handle non-legal RSS feeds [RSS and Web connectors]

5. MetaCarta-specific features, like document templates, are explicitly handled by the UI and the infrastructure. These features should be generalized so that they are controlled by the choice of output connector.
6. Some specific hooks, namely support for configuration change notification, and for database maintenance notification, may need to be made more generic.
7. Share Connector has a "fingerprinting" feature, which prefilters documents based on a document type it surmises using a document inspection technique. This feature is only viable at the moment for very basic document types. It should either be removed, or generalized significantly to be much more flexible.
8. Documentation needs to be fleshed out, including javadoc and overall usage documents.
9. Tests need to be written and/or ported from MetaCarta's test suite.

Longer term, the project will likely grow into a more distributed crawler, where multiple machines might well be involved in coordinated crawling activity.

Meritocracy

Building the community using a meritocratic approach is very important to the success of LCF. We know many, many people in the search space (and otherwise) have either written their own connectors or are in need of connectors. Thus, we expect a meritocratic community will lead to widespread participation.

Community

Our hope is that our existing code, features and capabilities will attract a large community of both developers and users. We also believe that other organizations will find this project interesting and relevant, and contribute resources.

The user community of LCF would be similar to that of the other Lucene projects, and in many cases they would overlap.

Core Developers

See the initial committer list below.

Alignment

We expect LCF will align quite well with the existing Lucene community and will also provide significant value to other ASF and non-ASF projects as well as many companies and individuals looking to access their content repositories in a programmatic fashion.

Known Risks

Orphaned Products

The Connector Framework is an important piece of any search engine, including MetaCarta's, as it provides the primary mechanism for getting content out of a repository and into the search engine's index. Thus, we don't expect it will be orphaned anytime soon. Once the project is established and the code is available, we expect to attract not only other search companies, but others with similar needs.

Inexperience with Open Source

Grant Ingersoll, Ryan McKinley and Simon Willnauer provide the majority of the experience with Open Source at the ASF, but all of the initial committers are familiar with Open Source and have contributed to other open source projects.

Homogeneous Developers

The current list of committers are mostly members of either the MetaCarta or Lucid Imagination developer team, but several are not. Additionally, we are actively recruiting other developers.

Reliance on Salaried Developers

We have a variety of committers represented. Some are being paid to work on the project and some are not.

Cryptography

Connector Framework itself has no real cryptography component, although it does currently obfuscate passwords it saves to the database or to a configuration file using a proprietary algorithm. The algorithm is present simply to avoid using cleartext and is not secure in any sense other than by obscurity.

Various connectors, such as Share Connector, Web Connector, RSS Connector, SharePoint Connector, LiveLink Connector, and Meridio Connector make use of cryptographic principles via secondary libraries. Specifically, these connectors support NTLM, NTLMv2, and NTLM2 Session authentication via commons-httpclient and jCIFS. The changes to commons-httpclient necessary to support these varieties of Windows protocols have not yet been accepted upstream by the Apache httpclient project.

It is unknown at this time exactly to what degree the Oracle JDBC driver, the jtds JDBC driver, or the Postgresql JDBC driver uses cryptography. Also, the FileNet API class, the Memex API classes, the OpenText LAPI api classes, and the Documentum DFC classes all may or may not use cryptography.

Legal Concerns

Some of the connectors in the existing framework require paid licenses to use. We will need to evaluate each connector to see what can be appropriately included. For those connectors that require a paid license, we will need to determine a plan for including the wrapper code without the underlying bindings in a legal manner. We expect we can provide the wrapper code without the binding and that the code will thus only be compilable by someone who has access to the binding. (This is what Google has done for their individual connectors). Longer term, we expect to demonstrate to the companies with proprietary connectors why it is more valuable for them to open up their specific connector pieces to give broader access to people looking to leverage their content in the repository.

Trademark

The project is being rebranded from a MetaCarta internal name to the Lucene Connector Framework, which will be an ASF mark.

Relationships with Other Apache Products

We expect almost all of the Apache Lucene ecosystem will benefit from having a standard way of connecting to content repositories. Additionally, users of UIMA should also benefit. We also see an especially tight connection with Tika, as much of the content in these types of repositories are "rich" document types which will then need their content extracted.

An Excessive Fascination with the Apache Brand

All of us are familiar with the value that Apache brings to a project in building out a community. We also are all significant users of Apache Lucene and related tools (Solr, Nutch, Mahout, Tika) and expect a close relationship with those projects will help significantly grow the LCF community.

Documentation

MetaCarta has end-user documentation for Lucene Connector Framework, which might function as the core the open-source end-user documentation. The documentation is in LaTeX form, and thus usable sources can readily be extracted. Research as to any ownership issues for the documentation as it stands still needs to be examined.

The existing java doc of the code, while fairly extensive, needs review and perhaps augmentation to insure it meets the needs of an ASF project. Significant attention to maintaining its accuracy was made during MetaCarta's ownership of the code base.

Initial Source

All initial sources will be coming from MetaCarta, Inc., with the goal of folding in changes from others shortly thereafter.

Source and Intellectual Property Submission Plan

Code IP grants need to be made from MetaCarta, Inc. But, in addition, several connectors (notably Documentum, LiveLink, Memex, and FileNet) rely directly on client API's in order to be compiled. Another connector (JDBC) relies on the existence of the Oracle JDBC Driver in the classpath in order to enable crawls against Oracle databases.

It is unlikely that EMC, OpenText, Memex, or IBM would grant Apache-license-compatible use of these client libraries. Thus, the expectation is that users of these connectors obtain the necessary client libraries from the owners prior to building or using the corresponding connector. An alternative would be to undertake a clean-room implementation of the client API's, which may well yield suitable results in some cases (LiveLink, Memex, FileNet), while being out of reach in others (Documentum). Conditional compilation, for the short term, is thus likely to be a necessity.

Other external dependencies, such as jCIFS for the Share Connector, are licensed with LGPL, and thus may need to be treated in a manner similar to the closed API's even though they are open source. These include the postgresql JDBC driver, and JTDS.

The Lucene Connector Framework core and individual connectors are completely separable, and many of the connectors require no third party licenses. Therefore, there is significant utility for this project even in the absence of any third-party software grants, or clean-room engineering.

The software grant will be faxed to the Apache Software Foundation if and when the proposal herein described is accepted. MetaCarta patents are not infringed by this grant. Also, MetaCarta trademarks are not included in this grant.

External Dependencies

The project dependencies, other than on other Apache projects, are as follows:

The ConnectorFramework core currently uses the Bitmechanic JDBC pool driver, which is BSD licensed, and the Postgresql JDBC driver, which is also BSD licensed.

The LiveLink Connector relies on LAPI, which is privately licensed by OpenText. The Documentum Connector relies on DFC, which is privately licensed by EMC. The Share Connector relies on jCIFS, which is LGPL. The Memex Connector relies on privately licensed java libraries from Memex. The FileNet Connector relies on privately licensed java libraries from IBM.

Required Resources

- Mailing lists
  - connectors-private (with moderated subscriptions)
  - connectors-user@
  - connectors-dev@
  - connectors-commit@
- Subversion directory
  - https://svn.apache.org/repos/asf/incubator/connectors
- Website
  - Confluence (CONNECTORS)
- Issue Tracking
  - JIRA (CONNECTORS)

Initial Committers

Names of initial committers with affiliation and current ASF status:

- Karl Wright (kwright at metacarta)
- Josiah Strandberg (jstrandberg at metacarta)
- Ken Baker (bakerkj at metacarta)
- Marc Meadows (mam at metacarta)
- Grant Ingersoll (gsingers@a.o Lucid Imagination, ASF Member)
- Brian Pinkerton (brian.pinkerton at Lucid Imagination)
- Simon Willnauer (simonw at apache org, Committer on Lucene Java and Lucene Open Relevance Project)
- Ryan McKinley (ryan at apache org, Committer on Lucene and Solr)
- Robert Muir (rmuir at apache org, Committer on Lucene and Open Relevance)
- Sami Siren (siren@a.o , Committer on Nutch and Tika)
- Otis Gospodnetic (otis@a.o , Committer on Lucene, Solr, Nutch, Mahout, and Open Relevance Project)
- Shalin Shekhar Mangar (shalin@a.o , AOL, Committer on Apache Solr)
- Noble Paul (noble@a.o , AOL, Committer on Apache Solr)
- George Aroush (george at aroush.net, Committer on Lucene.Net)
- Mark Miller (markrmiller at apache org, Committer on Lucene and Solr, LucidImagination)
- Erik Hatcher (ehatcher, Lucid Imagination, ASF Member, Committer on Lucene and Solr, Lucene PMC)

Sponsors

Champion

- Grant Ingersoll

Nominated Mentors

- Grant Ingersoll
- Jukka Zitting
- Gianugo Rabellino

Sponsoring Entity

- Apache Lucene PMC: Message ID: AF7E5AD4-E208-4866-8D29-6ADE19212F45@gmail.com in private@lucene.a.o