Optimizing Apache Nutch For Domain Specific Crawling at Large Scale

Luis A. Lopez, Ruth Duerr, Siri Jodha Singh Khalsa
luis.lopez@nsidc.org
http://github.com/b-cube

IEEE Big Data 2015, Santa Clara CA.

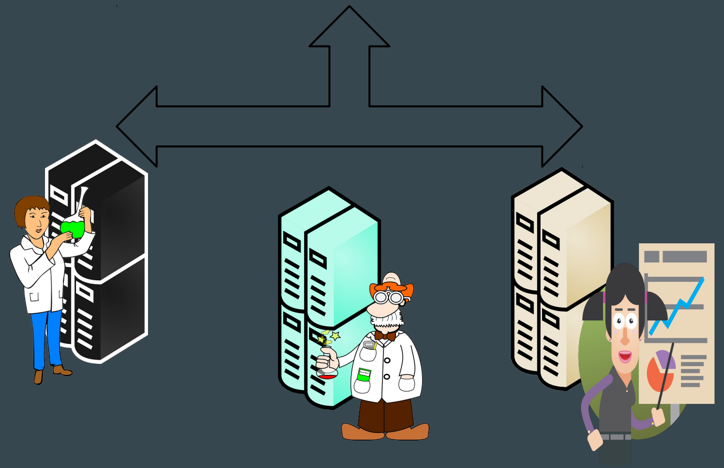
Overview

- BCube is a building block of NSF’ EarthCube, for the past 6 months we’ve been crawling the Internet trying to gather all possible data that’s relevant to the geosciences.
- Our focus is to discover scientific datasets and web services that may contain geolocated data. Mainly structured information (xml, json, csv)

Google

Greenland Aerosol Datasets ✓

Mass Spectrum Analysis in Geothermal Areas ✗



Understanding the problem [Focused Crawling]

Big Search Space With Very Sparse Data Distribution

- Billions of web pages
- Most content is not scientific data
- Scientific data is not well advertised

Solution

A good(enough) scoring algorithm

Acceptable Performance With Limited Resources

- Scalable stack
- Handles TB of data
- Distributed processing
- Fault tolerant
- Uses commodity hardware

Apache Nutch!

… Hard Problems

- Content Duplication
- Semantics
- Robots.txt
- Remote Servers Performance
- Malformed Metadata
- Bad Web Standards Implementation
- Cost

Previous Work and BCube

Previous Work

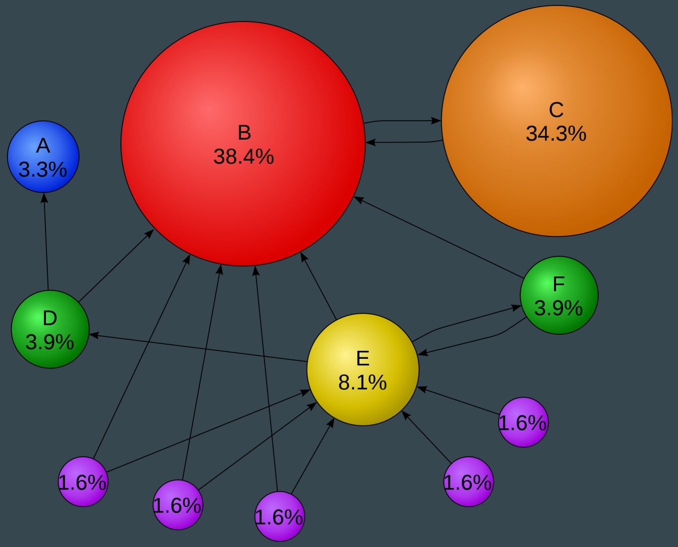
- Finding the scoring algorithm that performs 10% better
- Implementing an in house(not open sourced) crawler
- Focusing on an specific type of data
- Measuring performance on thousands of pages(sometimes just hundreds)

Our Work

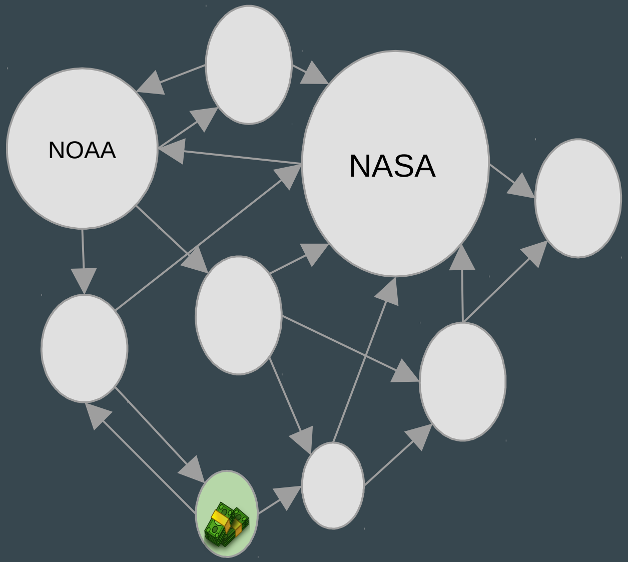
- To understand where the bottlenecks are
- To use an open source project
- To improve fetching times
- To modify the crawler for focused crawls
- To use “the cloud” to lower operational costs
- To verify what happens at large scale

Scoring

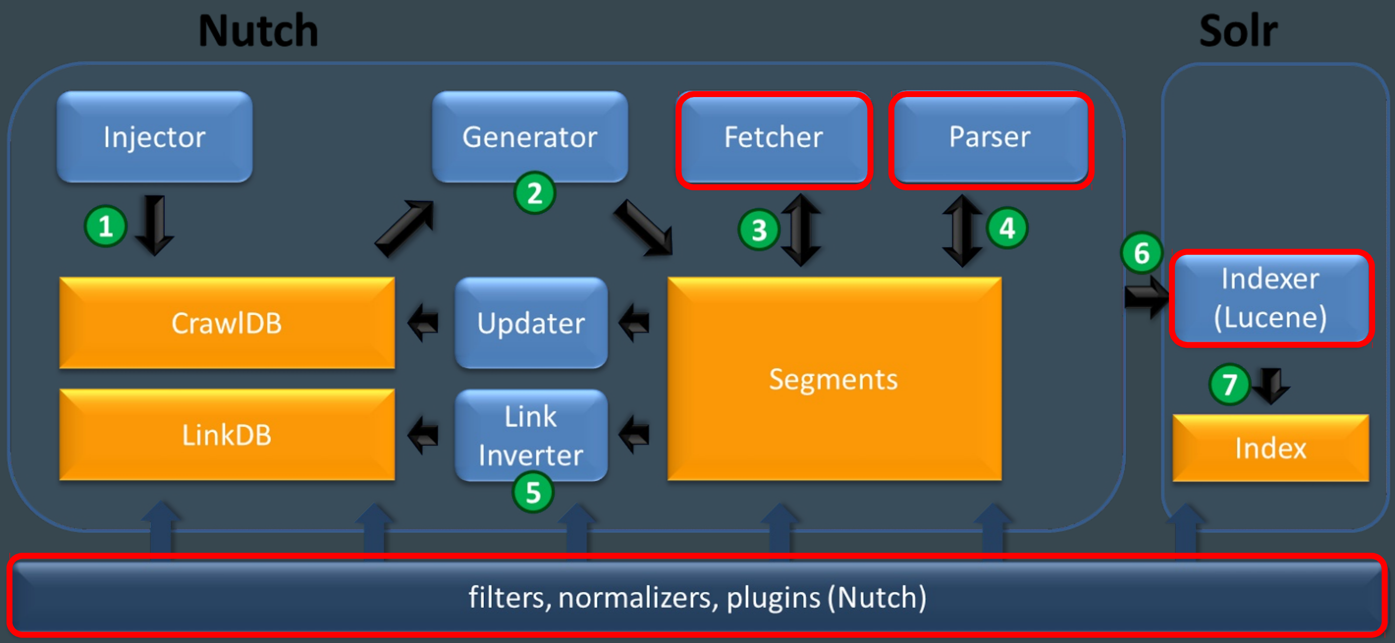
PageRank Like Scoring



Focused Crawl Scoring



BCube Customizations



BCube Plugins

Parse-rawcontent

Indexes the unparsed content of a document

parse-bayes

Scores pages using an online naive-bayes classifier

index-xmlnamespaces

Indexes all the namespaces used in xml documents

index-links

Indexes inlinks and outlinks of a document

index-bcube-extras

Indexes HTTP responses

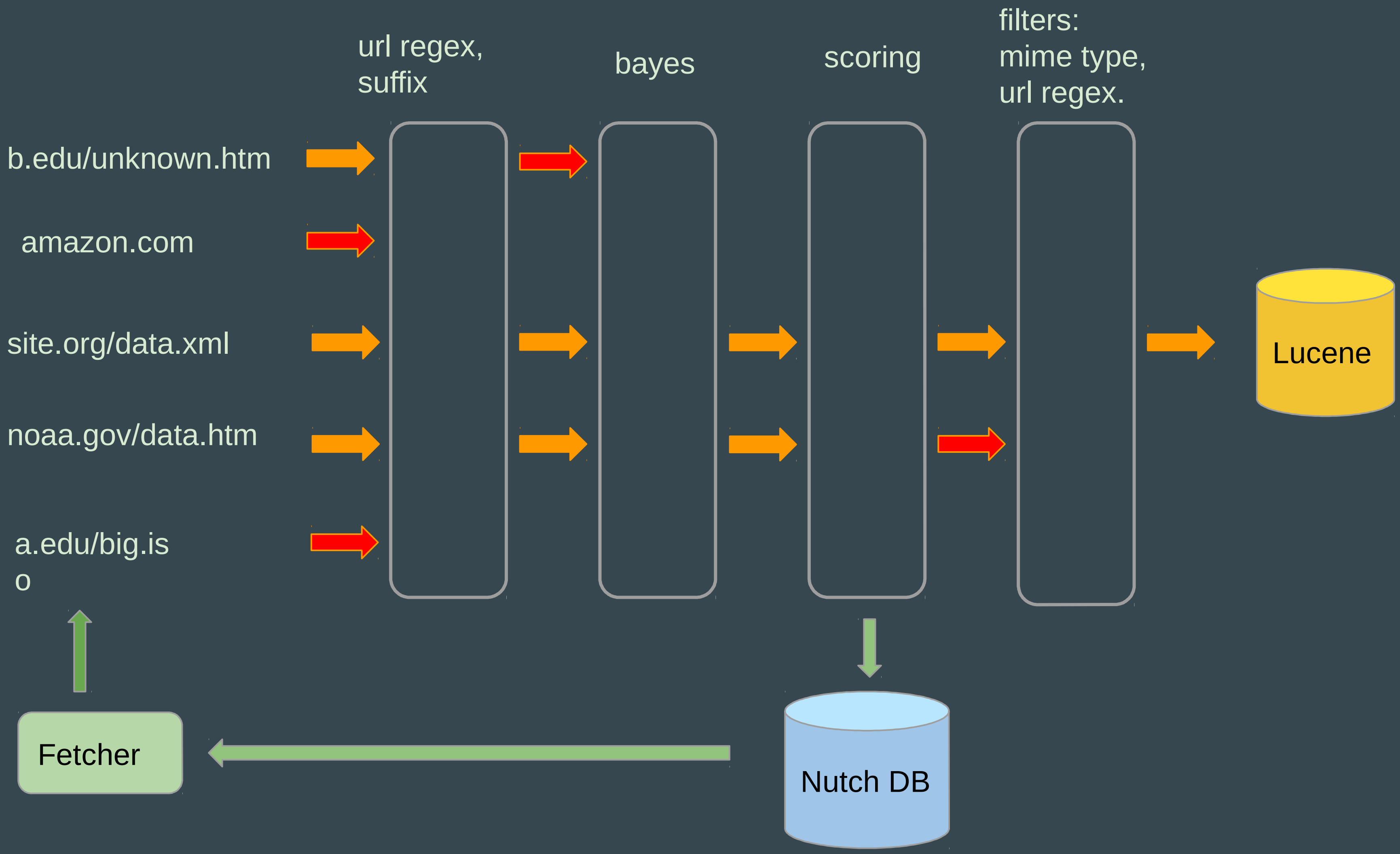
index-bcube-filter

Discards documents using mime types or substring matching

parse-tika

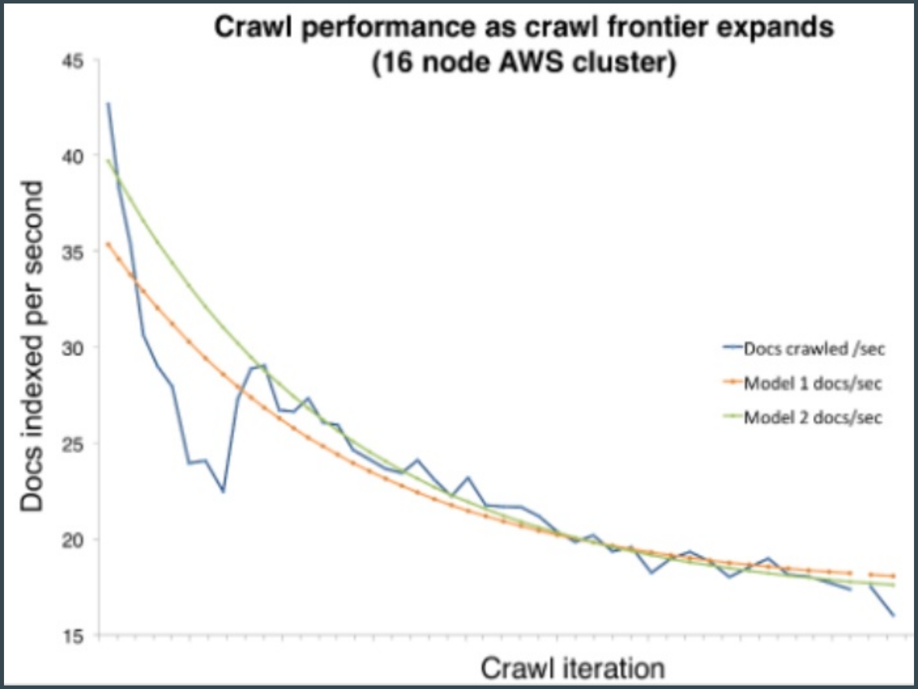
fixed critical bug that blocked us from parsing valid XML files

BCube Filtering



Problems…

Performance Degradation

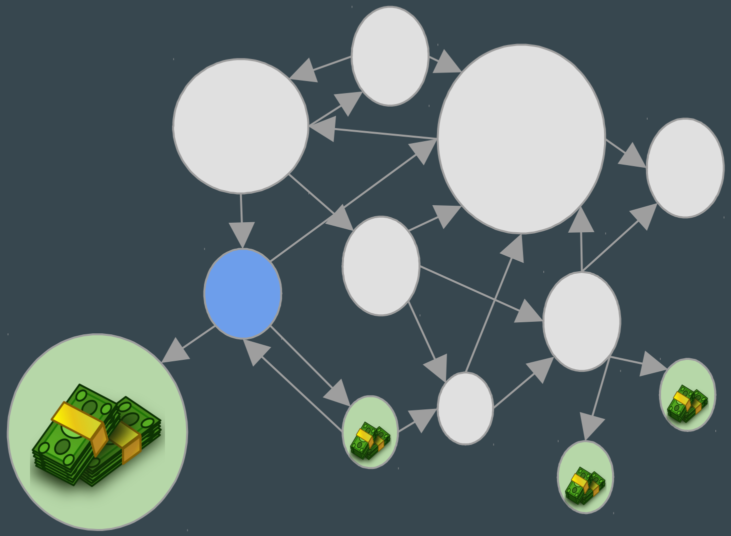


- Crawl-Delay
- Sparse data distribution
- Duplicated content
- Slow servers
- “The tar pits”
- Variable cluster performance in the cloud
- Idle CPU time

https://wiki.apache.org/nutch/OptimizingCrawls

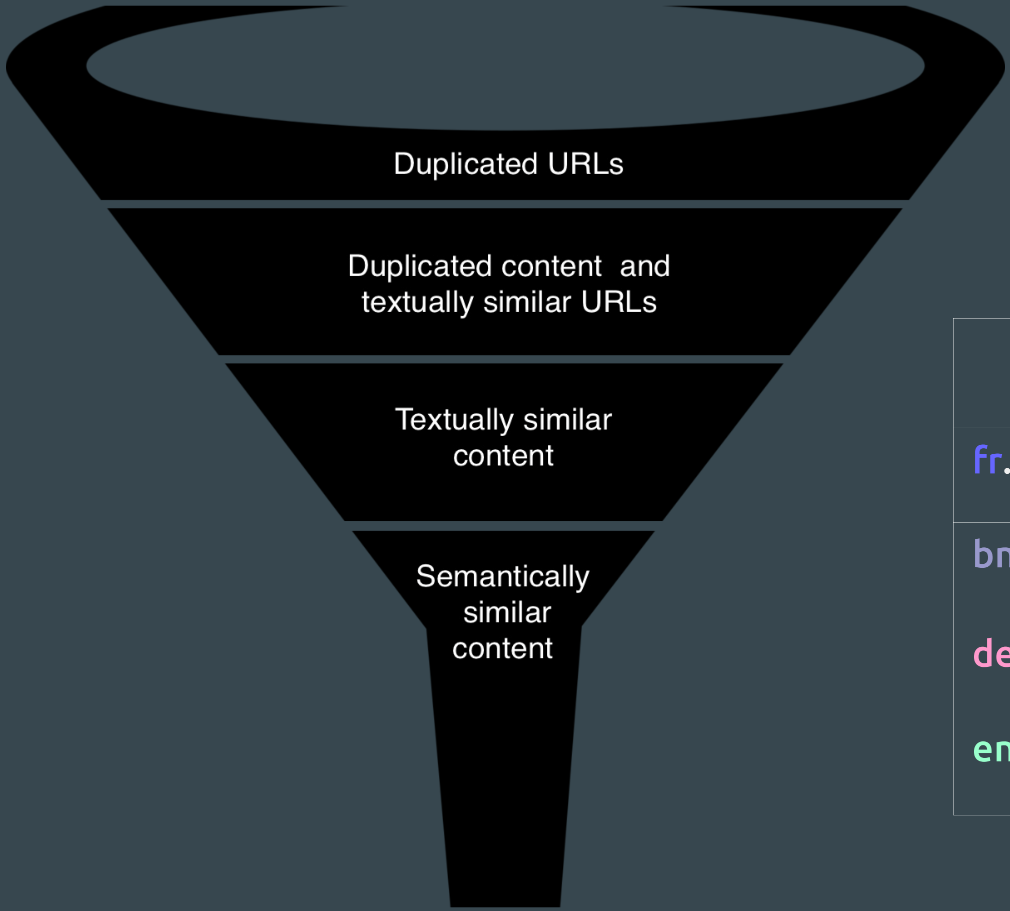
Good Scoring, Let's Celebrate!... not so soon.

The Tar Pit



- Keep indexing relevant documents from the same sites prevented us from discover new ones
- Well scored documents are still relevant and we should index them
- How often these sites are updated should be taken into account.

Content Duplication... at large scale.


Duplicated URLs
Duplicated content and textually similar URLs
Textually similar content
Semantically similar content

Domain	Documents fetched
fr.climate-data.org	212142
bn.climate-data.org	209257
de.climate-data.org	203279
en.climate-data.org	197716

* The Science of Crawl: Deduplication of Web Content http://bit.ly/1Gg32Hh

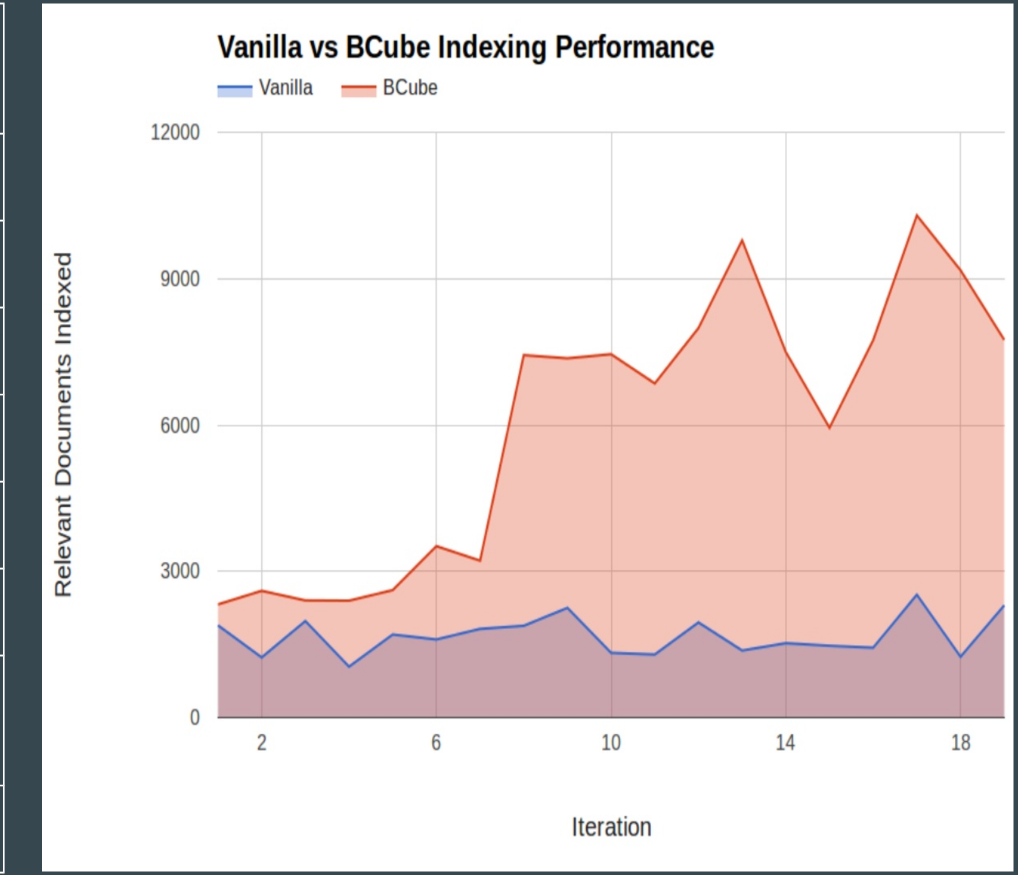
Improving Performance

- Robots.txt
  - Crawl-Delay
- Large files:
  - .ISO .HDF etc.
- SSD vs HDD
- Fetching Strategy

- Crawling at different speeds.
- Filtering out file extensions using Nutch's suffix-regex filter
- SSD instances on AWS
- Generate a limited number of links per host and distribute the fetch on as many nodes as possible.

Nutch + BCube

Property	Default Value	BCube Value
crawldb.url.filters	True	False
db.update.max.inlinks	1000	100
db.injector.overwrite	False	True
generate.max.count	-1	1000
fetcher.server.delay	10	2
fetcher.threads.fetch	10	128
fetcher.threads.per.queue	1	2
fetcher.timelimit.mins	-1	45



Conclusions and Future Work

- There are major issues in focused crawls that can only be reproduced at large scale
- Some issues cannot be addressed by improving the focused crawl alone
- We can implement mitigation techniques effectively to alleviate the problems under our control
- Apache Nutch can scale and be used for focused crawls

- Optimize scoring algorithm using the link graph and content context
- Develop a computationally efficient mechanism for dynamic relevance adjustment
- Automate cost effective cluster deployments
- Use the latest selenium plugins in Nutch for specific use cases
- More…

References

BCube at Github

https://github.com/b-cube

Apache Nutch

https://nutch.apache.org/

Common Crawl Project

https://commoncrawl.org/

The Science of Crawl:
Deduplication of Web Content

http://bit.ly/1Gg32Hh