Winning the Big Data Spam Challenge

Erich Nachbar
quantiFind

Stefan Groschupf
Datameer

Florian Leibert
twitter

Spam Types - Email Spam

- What do spammers do?
  - Many domains
  - Cycle through IPs (TOR, bulk blocks)
  - Bulk account creation (increase IP reputation)
    - Break captchas (Mechanical Turk)
    - Common names
      (e.g. http://www.census.gov/genealogy/names/dist.male.first)

Spam Types - Social Media

- Spam Carriers
  - Blog Postings
  - Comments
  - Friend Requests
  - ...
- Spam Generation through
  - Actual User Accounts (Hacked / User Virus)
  - Bot Accounts

Spam Types - Social Media

- Differences
  - Detection is the same
  - Account treatment is different (cancel vs. clean)
- 99% of all Spam contains URLs:
  - Ignore text-only messages.
  - Look at the URL not the text.

Spam Types - Web Spam

- Goal: influence search engine results
  - Link farms
  - Keyword, Meta tag stuffing
  - Hidden or invisible unrelated text
  - Scraper sites
  - Spam blogs

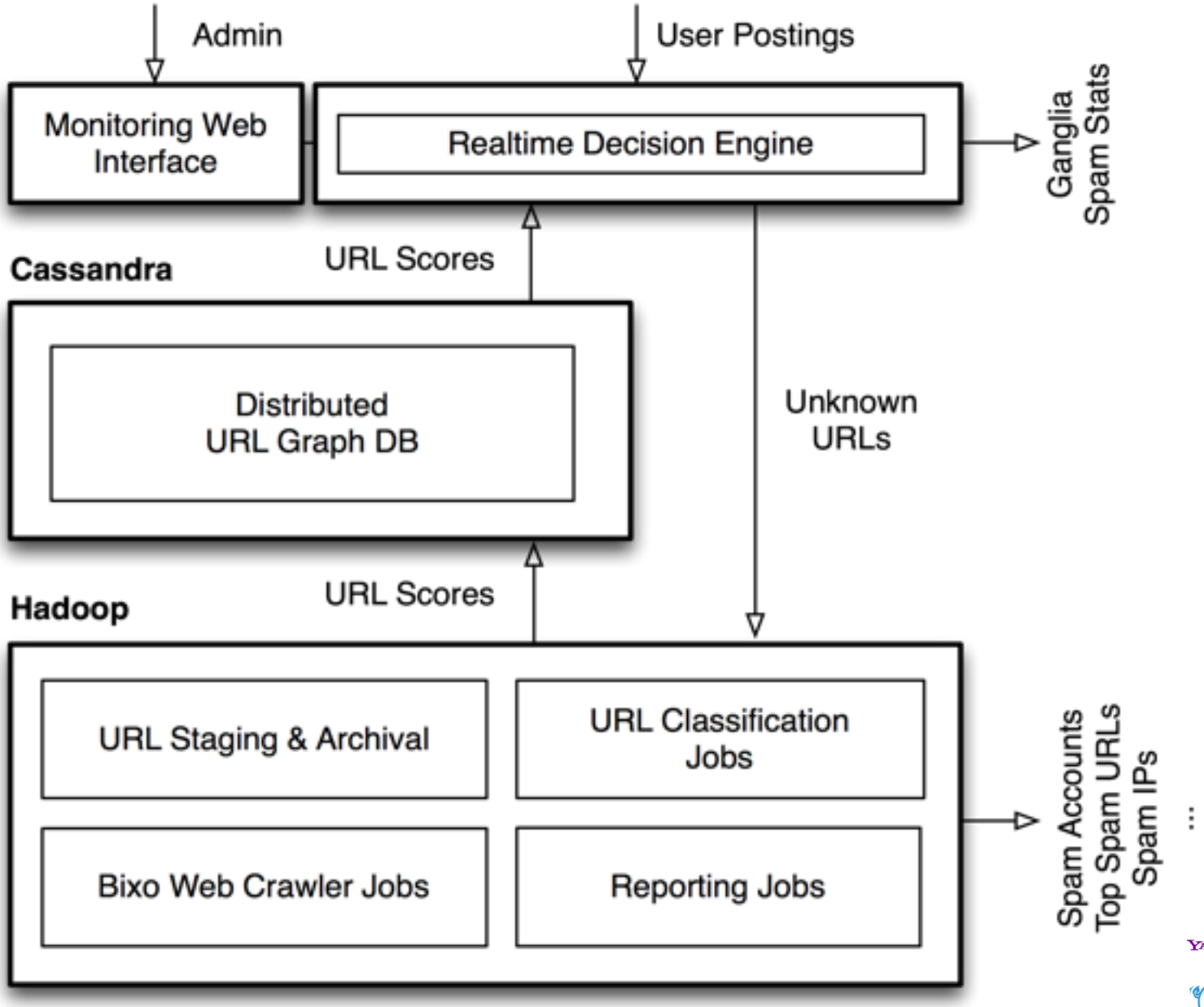
Why process Spam in Hadoop?

- Easy to parallelize
  - Bucketization
    - User
    - Date
    - Source
    - etc.
  - Count models (probabilities) are very "hadoopy"

Why process Spam in Hadoop?

- Large data sets
  - More samples ~ better results
- Algorithms require preprocessing
- Existing code
  - e.g. url-parsers, bayes implementations

Sample System Architecture



Heuristics for Spam Detection

- Easy to compute, Group-By & Count
- Captcha solving rates
- Source IP/Email
  - Historic vs. current volume
  - Reputation
- Link
  - Frequency, position, ratio

Heuristics for Spam Detection

- Content
  - Self similarity
  - Hash of media content
  - Keywords

Evaluating content

- Jaccard similarity

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}.$$

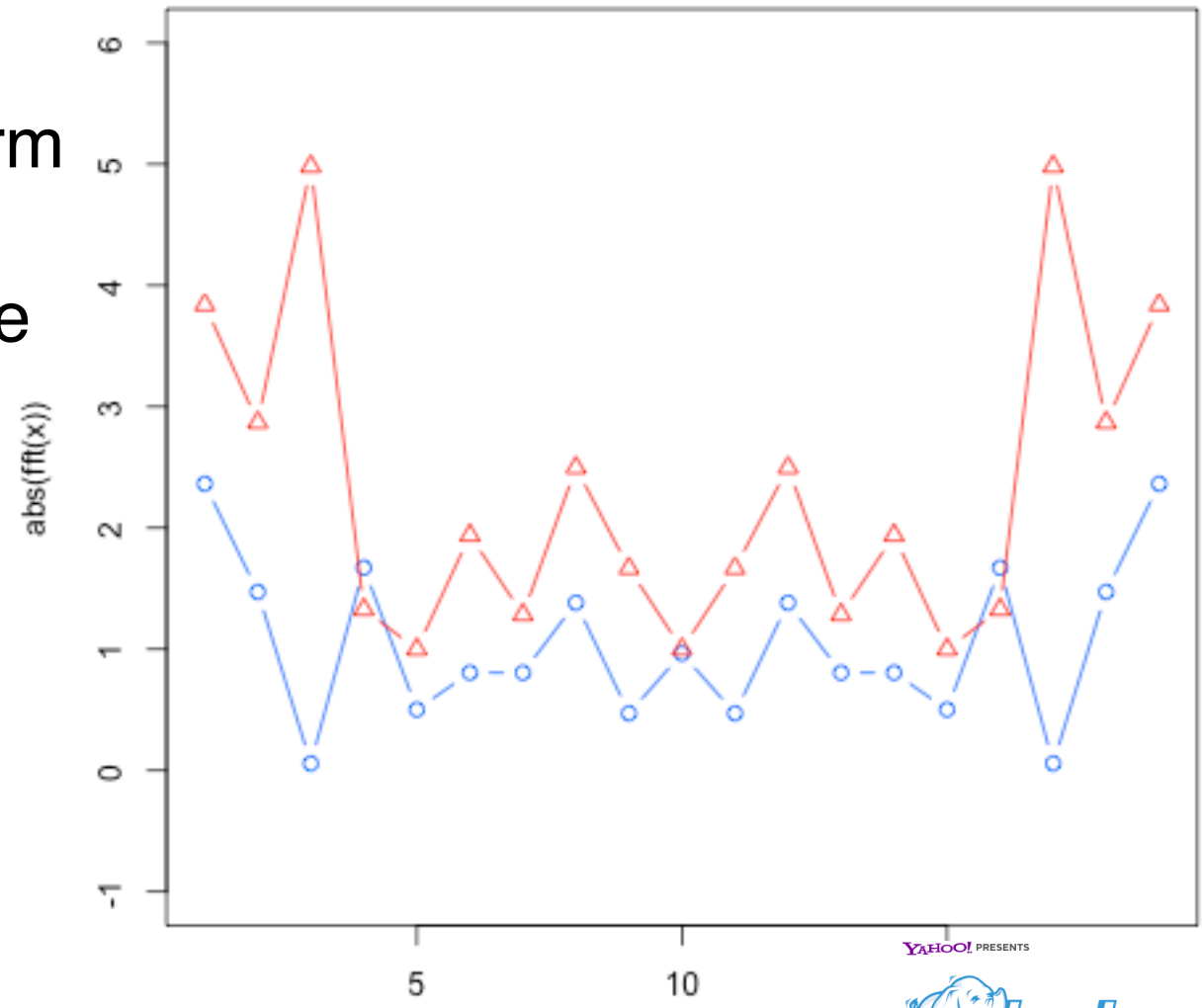
  - Bucketize by user / source email
    - s1 = S(x1,x3,x5), s2 = S(x2,x4,x6)
  - Easy with Hadoop
    - map: emit user_id (K), text (V)
    - reduce:

```
Sets.SetView intersection = Sets.intersection(set1, set2);
Sets.SetView union = Sets.union(set1, set2);
double jaccard = 0;
if (union.size() != 0) {
  jaccard = (double) intersection.size() / (double) union.size();
}
```

- Even simpler:
  - # links / user
  - # complaints, spam tags / user
  - etc.

Looking at arrival times

- Inter-arrival times
- Fast Fourier Transform
  - Timestamps
    -> frequency space



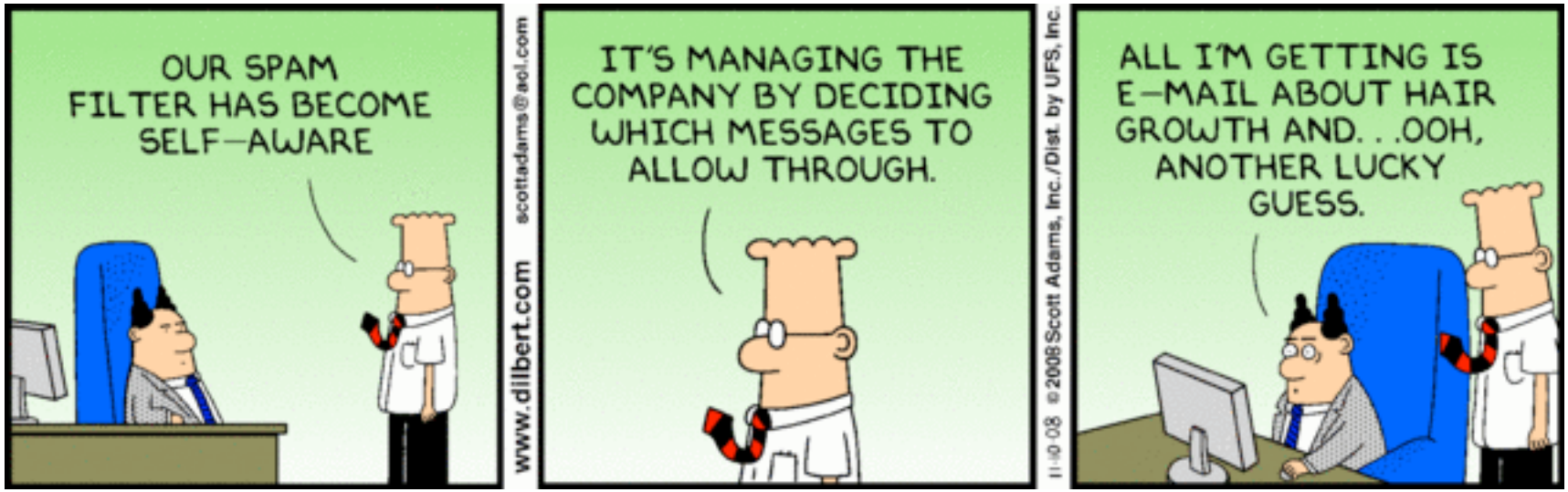
Solutions - Pay Level Domain

- Requires payment at Top Level Domain
- Simple heuristic
- Much simpler than Trust-Rank, Page-Rank, etc.

Demo

Take Aways

- Spam Reports are important
- Rolling real-time Ham & Spam Samples
- Have Knobs to turn (e.g. over JMX)
- Simple solutions can get you pretty far, the easy 80%
- Spammers adapt very fast, stay agile
- Try to break your own system



Thank you!

erich@quantifind.com, @enachb

sg@datameer.com, @datameer

flo@leibert.de, @floleibert

YAHOO! PRESENTS
hadoop
SUMMIT 2010