Rails and the Apache SOLR Search Engine

By David Keener

dkeener@keenertech.com
david.keener@gd-ais.com
Blog: http://www.keenertech.com

2012
march 23-24, reston, va
rubynation

Agenda

1. Why?
2. Technology Stack
3. Our First App

- Demo

4. Complications
5. Conclusions

Part 1: Why?

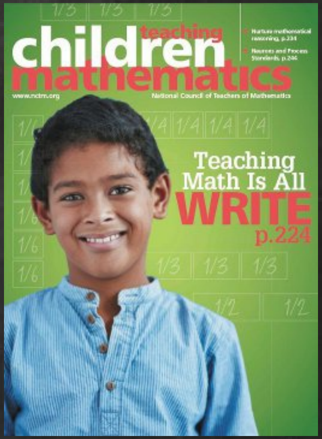
Part 1: Why?

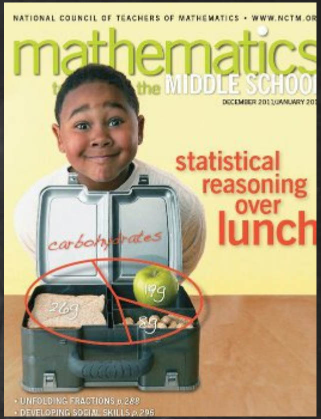
“Why the Lucky Stiff” at a conference in 2006…
- innovative, eccentric, suddenly retired from public life…

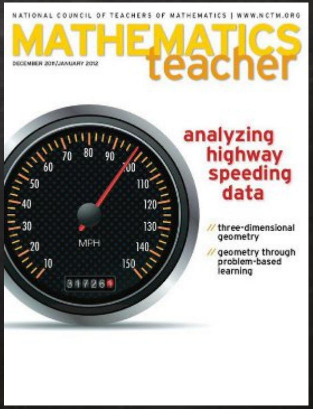
Why Me?

Well, I've been doing Search since 1998…

- **CheckPoint** – Online tax information for CPA's
- **Legislate** – Everything you ever wanted to know about legislation – more than a million docs
- **National Council of Teachers of Mathematics** – Online journals
- **Grab Networks** – Searching news video summaries
- **Pfizer** – Drug documentation


teaching
children mathematics
www.nctm.org
National Council of Teachers of Mathematics
Teaching Math Is All WRITE p.224


NATIONAL COUNCIL OF TEACHERS OF MATHEMATICS • WWW.NCTM.ORG
mathematics teaching in the MIDDLE SCHOOL
statistical reasoning over lunch
carbohydrates
UNFOLDING FRACTIONS p.288
DEVELOPING SOCIAL SKILLS p.296


NATIONAL COUNCIL OF TEACHERS OF MATHEMATICS | WWW.NCTM.ORG
MATHEMATICS teacher
DECEMBER 2011/JANUARY 2012
analyzing highway speeding data
three-dimensional geometry
geometry through problem-based learning
MPH

A Typical Information Site

Why We Need Search

- **“A feature doesn’t exist if users can’t find it.”**

  - Jeff Atwood, co-creator of *Stack Overflow*

  - The same principle applies to content

- **Content costs money**

  - If people can’t find it, your money is wasted

- **The Long Tail**

  - More.content should be == More.traffic

Part 2: Technology Stack

Lucene Apache Solr

Lucene is a Toolbox…

- Indexing
- Searching
- Spell-checking
- Hit Highlighting
- Advanced tokenization

SOLR is a Search Server…

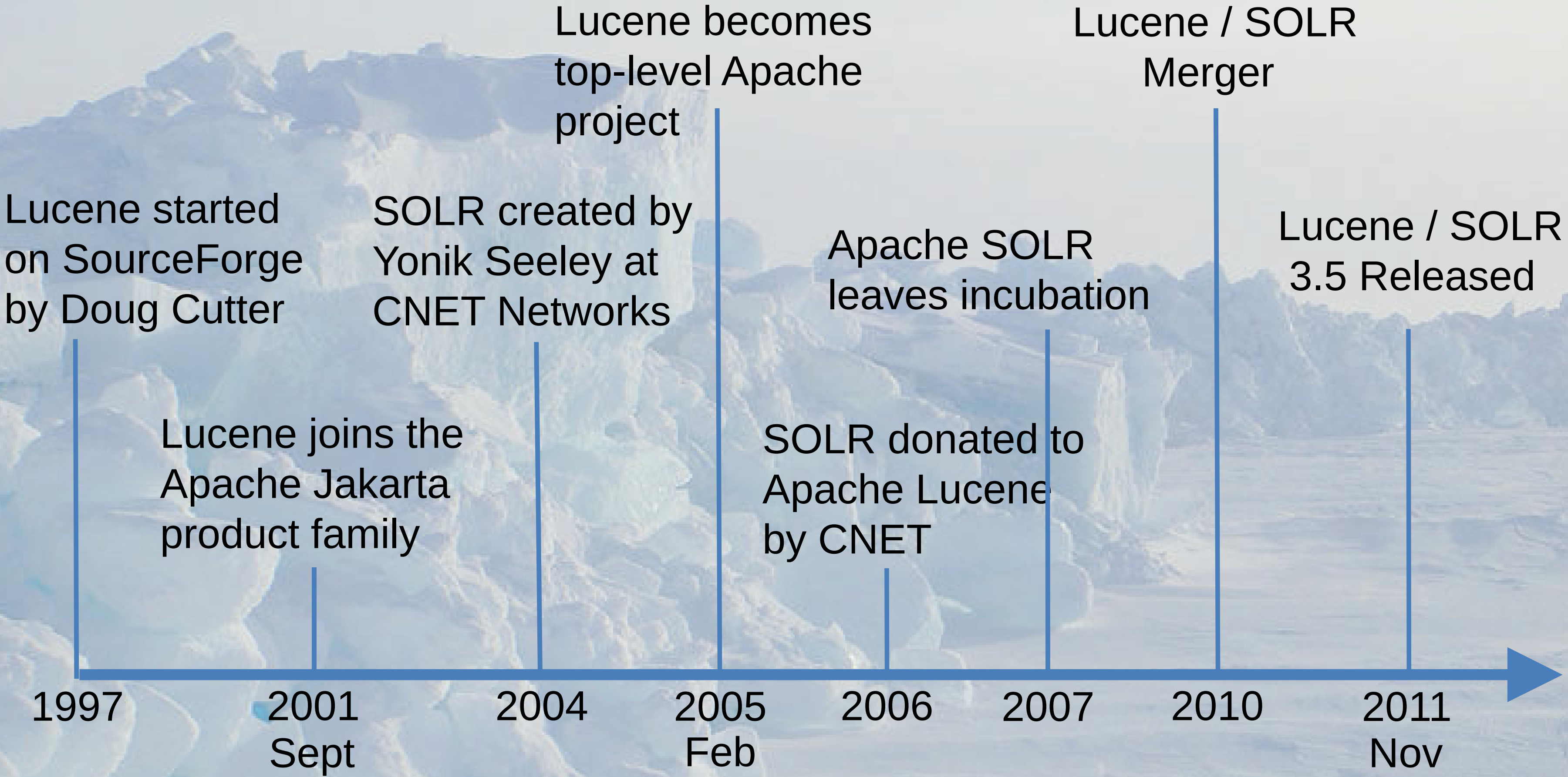
- With API’s
- Hit Highlighting
- Faceted Searches
- Caching
- A Web Admin Interface

JAVA

JAVA

SOLR and Lucene need Java 1.4 or higher

Timeline



1997 — Lucene started on SourceForge by Doug Cutter

2001 Sept — Lucene joins the Apache Jakarta product family

2004 — SOLR created by Yonik Seeley at CNET Networks

2005 Feb — Lucene becomes top-level Apache project

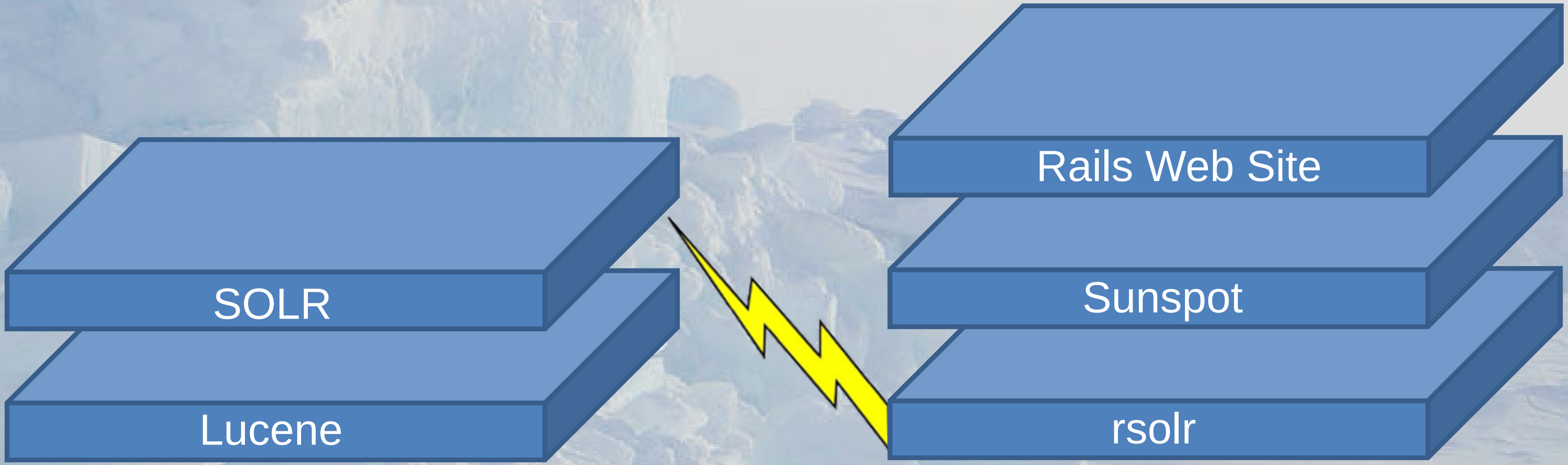
2006 — SOLR donated to Apache Lucene by CNET

2007 — Apache SOLR leaves incubation

2010 — Lucene / SOLR Merger

2011 Nov — Lucene / SOLR 3.5 Released

Search Stack for Our Rails App



Dev/Test: Jetty

Production: Tomcat

Dev/Test: WEBrick

Production: Apache with
Phusion Passenger

Rsolr and Sunspot

Rsolr is a SOLR client…

- By Matt Mitchell
- Originated Sept 2009
- Reached 1.0 Jan 2011
- Now at Version 1.0.7
- Low-level client

Sunspot provides Ruby-style API's…

- By Andy Lindeman, Nick Zadrozny & Mat Brown
- Originated Aug 2009
- Reached 1.0 Mar 2010
- Now at Version 1.3.1
- With Rails or just Ruby
- Drop-in ActiveRecord support

Part 3: Our First App

Another First…

The world’s first amphibious lamborghini

A Simple Blog – With Search

- **Generate a Rails App**

  - rails new demo1
- **Configure Gemfile**

  - bundle install

```
source 'http://rubygems.org'

gem 'rails', '3.0.11'
gem 'sqlite3'                    # Default database
gem 'will_paginate'
gem 'sunspot_rails'              # Sunspot for Rails

group :development do
  gem 'nifty-generators'  # Better scaffolding
  gem 'sunspot_solr'             # Pre-built SOLR
end

group :test do
  gem 'sunspot_solr'             # Pre-built SOLR
  gem "mocha"                    # Testing tool
end
```

Gemfile

A Simple Blog (2)

- **Scaffolding and Database**

```
- rails g nifty:layout
- rails g nifty:scaffold Post title:string body:text featured:boolean
- rake db:migrate
```

- Also, remove public/index.html and point root to posts#index

- **Populate the SQLite3 Database:**

```
- sqlite3 development.sqlite3
      > read data.sql      # Loads 100+ blog entries
      > .exit
```

A Simple Blog (3)

- **SOLR Config File**

  - rails generate sunspot_rails:install

  - Creates config/sunspot.yml

- **Make Post class searchable**

```
class Post < ActiveRecord::Base
  attr_accessible :title, :body, :featured
  self.per_page = 10

  searchable do
    text :title, :body
    integer :id
    boolean: featured
  end

end
```

/app/models/post.rb

```
production:
  solr:
    hostname: localhost
    port: 8983
    log_level: WARNING

development:
  solr:
    hostname: localhost
    port: 8982
    log_level: INFO

test:
  solr:
    hostname: localhost
    port: 8981
    log_level: WARNING
```

/config/sunspot.yml

A Simple Blog (4)

- **Start SOLR**

  - rake sunspot:solr:start

  - Creates SOLR directory tree on first start

- **Index Your Data**

  - rake sunspot:solr:reindex

```
- solr
   - conf
   - data
         - development
         - test
   - pids
         - development
         - test
```

SOLR Directory

```
solr/data
solr/pids
```

.gitignore

The Search UI

- We’ve got a web site
- SOLR is running
- We’ve got indexed content

But…

- We need a Search Form
- We need a Search Results page

The Search Form

```
<%= form_tag(searches_path, :id => 'search-form', :method => :get) do |f| %>
   <span>
     <%= text_field_tag :query, params[:query] %>
     <%= submit_tag 'Search', :id => 'commit' %>
   </span>
<% end %>
```

/app/views/layouts/_search.html.erb

- Just a simple view partial…
- That’s rendered by the site’s layout

Search Results

```
resources :searches, :only => [:index]
```

config/routes.rb

```
<% if @posts.present? %>
  <%= will_paginate @posts, :container => false %>

  <table>
    <% for post in @posts %>
      <tr><td"><%= raw(post.title) %></td></tr>
      <tr><td><%= post.body.truncate(300) %></td></tr>
    <% end %>
  </table>

  <%= page_entries_info @posts %>
  <%= will_paginate @posts, :container => false %>

<% else %>
  <p>No search results are available. Please try another search.</p>
<% end %>
```

app/views/searches/index.html.erb

Search Controller

```
class SearchesController < ApplicationController

  def index
    if params[:query].present?
      search = Post.search {
        fulltext params[:query]
        paginate :page => params[:page].present? ? params[:page] : 1,
                        :per_page => 10
      }
      @posts = search.results
    else
      @posts = nil
    end
  end

end
```

ActiveRecord Search Integration

Pagination integrates w/ will_paginate gem

Security: Params[:query] must be scrubbed if it will be re-displayed…

app/controllers/searches_controller.rb

What About New Posts?

How do new posts get indexed?

For any model with searchable fields…

- Sunspot integrates with ActiveRecord
- Indexing happens automatically on save

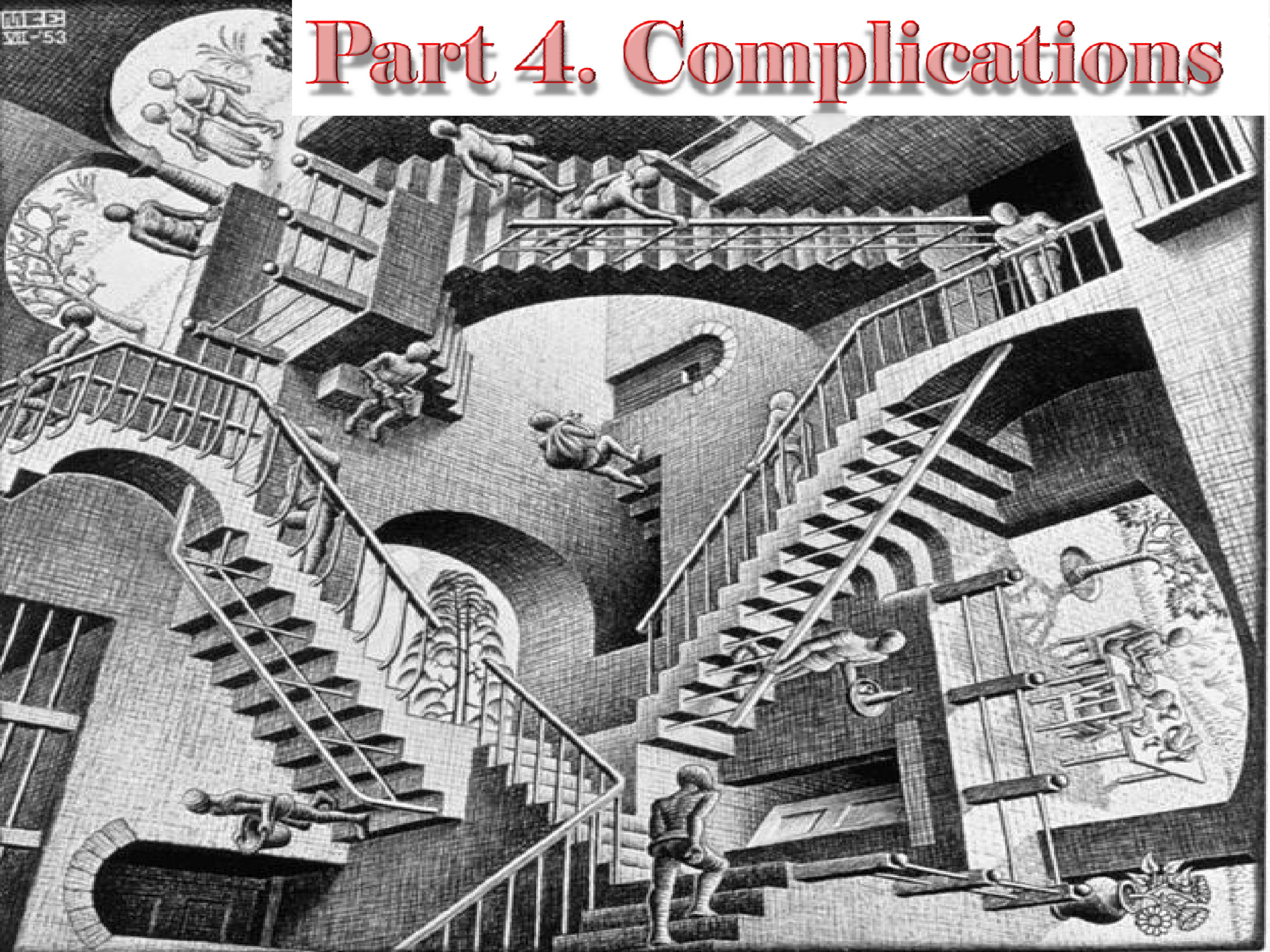
Demo

In 2006, DHH made a big splash with his “15-minute blog with Rails”

This is the…

20-Minute Rails Blog with Search

Part 4. Complications



Tip: A Clean Start

- In development, sometimes you mess up…
- You can hose up the SOLR indices

Solution:

- Don’t be afraid to blow away the *solr* dir tree
- “rake sunspot:solr:start” rebuilds the tree
- Just don’t lose any *solr/conf* changes

Search Weighting

Titles seem more important…can we weight the title higher than the body?

```
search = Post.search {
      fulltext params[:query]
      paginate :page => params[:page].present? ? params[:page] : 1,
                    :per_page => 10
 }
```

BEFORE

```
search = Post.search {
      fulltext params[:query] do
          boost_fields :title => 2.0
      end
      paginate :page => params[:page].present? ? params[:page] : 1,
                    :per_page => 10
    }
```

AFTER

Is There a Better Way?

The “boost” can be done at index time…

```
class Post < ActiveRecord::Base
  searchable do
    text :title, :boost => 2.0
    text :body
    integer :id
    boolean: featured
  end
end
```

/app/models/post.rb

What About Related Data?

```
text :comments do
  comments.map { |comment| comment.body }
end
```

/app/models/post.rb

- Could have used “acts_as_commentable” gem
- Your “document” is virtual
- You define it
- You can reference attributes, methods, etc.

Filtering

```
search = Post.search {
   fulltext params[:query] do
      boost_fields :title => 2.0
   end
   with(:featured, true)
}
```

/app/controllers/searches_controller.rb

- Text fields are searched
- The boolean “featured” attribute is filtered
- Search returns only featured posts that match the full-text search criteria

Hit Highlighting

```
class Post < ActiveRecord::Base
  searchable do
    ...
    text :body, :stored => true
  end
end
```

/app/models/post.rb

```
@search = Post.search {
   fulltext params[:query] do
     highlight :body
   end
}
```

/app/controllers/searches_controller.rb

```
@search.hits.each do |hit|
  puts "Post ##{hit.primary_key}"
  hit.highlights(:body).each do |highlight|
    puts " " + highlight.format { |word| "*#{word}*" }
  end
end
```

/app/views/searches/index.html.erb

```
Post #1
  I use *git* on my project
Post #2
  the *git* utility is cool
```

OUTPUT

Authorization

Just because content is indexed doesn’t mean a particular user is allowed to see it…

- **Search-enforced access control**

  - Access control data stored in index

  - Can be used to filter results

(No code, but something to think about…)

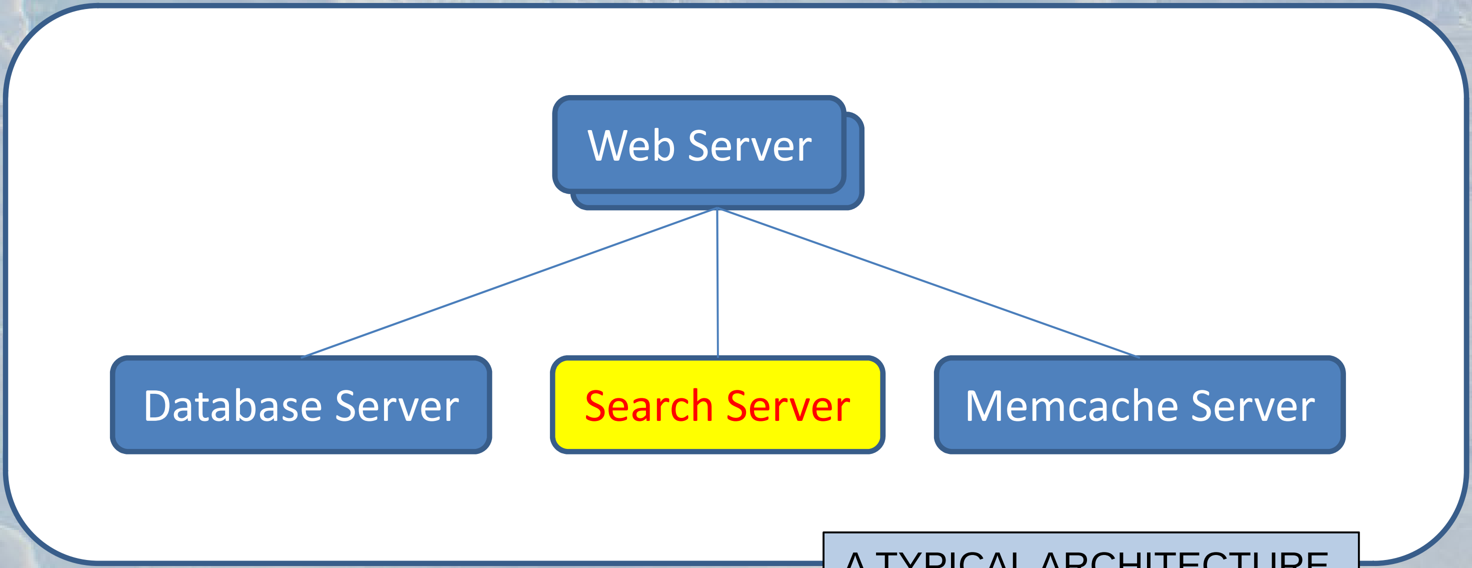
Part 5: Conclusions

Apache Solr Is Highly Customizable

Apache Solr Is Easily Added to Any Project

Apache Solr Helps You Leverage Your Content

Apache Solr Is Designed to be an Enterprise Component



Web Server

Database Server

Search Server

Memcache Server

A TYPICAL ARCHITECTURE

Apache Solr Is Well Supported

Questions?

Get the Code: https://github.com/dkeener/rails_solr_demo

Get the Slides: http://www.keenertech.com/presentations/rails_and_solr

I'm **David Keener** and you can find me at:

- **Blog:** http://www.keenertech.com
- **Facebook:** http://www.facebook.com/keenertech
- **Twitter:** dkeener2010
- **Email:** dkeener@keenertech.com
  david.keener@gd-ais.com

David Keener

From the Washington DC area.

I'm a software architect for **General Dynamics Advanced Information Systems**, a large defense contractor in the DC area now using Ruby, Rails and other open source technologies on *various projects*.

A founder of the RubyNation and DevIgnition Conferences.

Frequent speaker at conferences and user groups, including SunnyConf, Scotland on Rails, RubyNation, DevIgnition, etc.

GENERAL DYNAMICS

NCTM

gab NETWORKS

AOL

CareerBank.com

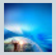
NEXTEL

PSINet

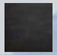
THOMSON

NASDAQ

Credits


http://wwwimagebase.davidniblack.com/templates/

http://www.nctm.org – Journal thumbnails


http://www.sciencewallpaper.com/blackboard-wallpaper/

http://www.flickr.com/photos/pragdave/173649119/lightbox/

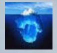
http://brooklynlabschool.wordpress.com/2010/02/09/community-resources-february-2010/

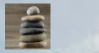
http://www.gadgetlite.com/wp-content/uploads/2009/03/amphibious-lamborghini-mod.jpg

David Keener speaking at AOL; photo by Edmund Joe

http://www.sciencepoles.org/articles/article_detail/paul_mayewski_climate_variability_abrupt_change_and_civilization/

No known copyright; received via spam


Ubiquitous iceberg picture on the Internet; no known copyright


http://www.123rf.com/photo_4155512_stack-of-stones.html

http://topwalls.net/water-drop-splash-art/

http://en.wikipedia.org/wiki/File:Escher%27s_Relativity.jpg

http://www.graphicsfuel.com/2011/02/3d-target-dart-psd-icons/

NCTM PSINet CareerBank.com GENERAL DYNAMICS gab NETWORKS AOL NASDAQ

All logos are trademarks of
their associated corporations

THOMSON NEXTEL