Building Search Based Applications

opencrowd

Introduction

Sushil Prabhu
Chief Executive Officer,
OpenCrowd

Stephen Coursen
Director of Engineering,
OpenCrowd

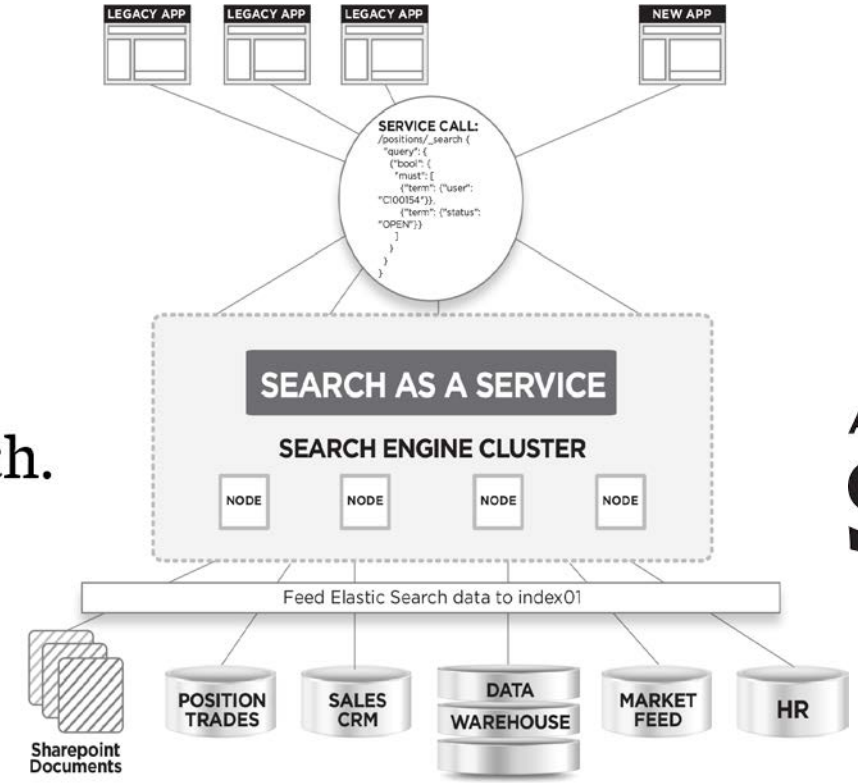
opencrowd

Agenda

- What are Search Based Applications?
- How can you benefit?
- How do you build one?
- Demo

What are Search Based Applications?

Search Based Applications are a new breed of systems that leverage search as a backbone for data retrieval



elasticsearch.

Apache Solr

Lucene

opencrowd

A Search Based Application

Auto- Suggest

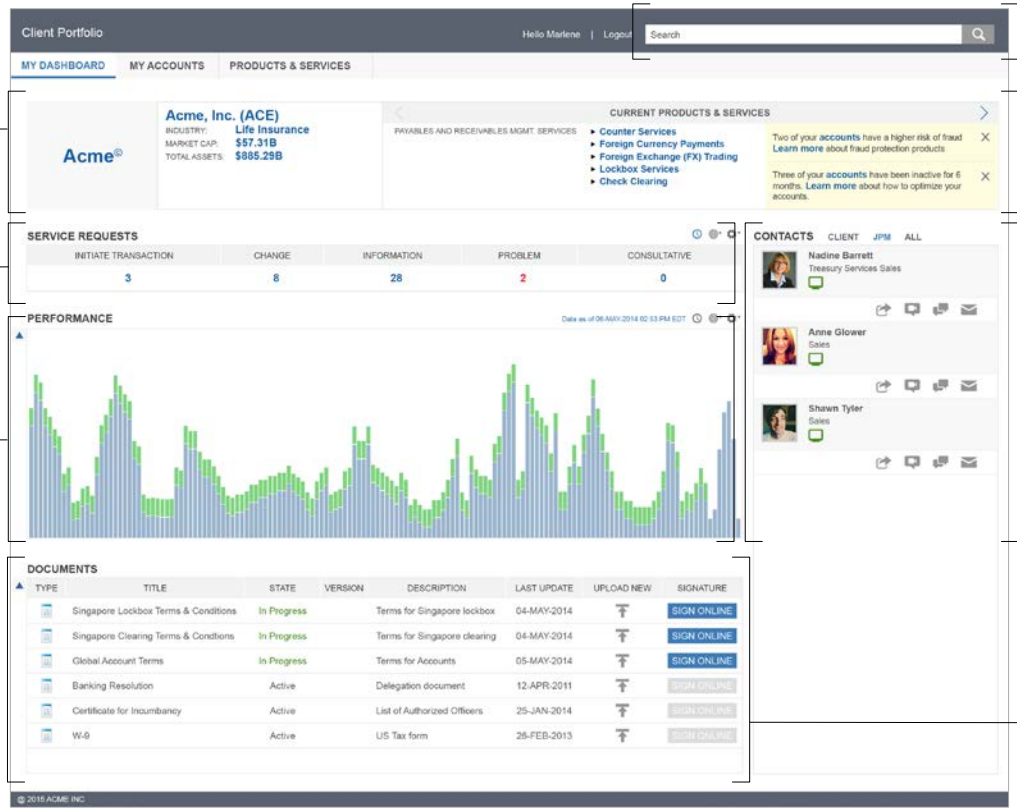
Client Profile

Issue Tickets

Real-Time

Client Managers

Client Contracts



Index

zendesk Confluence SIEBEL Logfiles...

Index

opencrowd

Current challenge in the enterprise

There is a constant need to enrich applications with new sets of data and in many cases these data sets are huge

The issue is that in most applications, data relationships are predefined

Bringing new and a large set of data in an application is a major undertaking

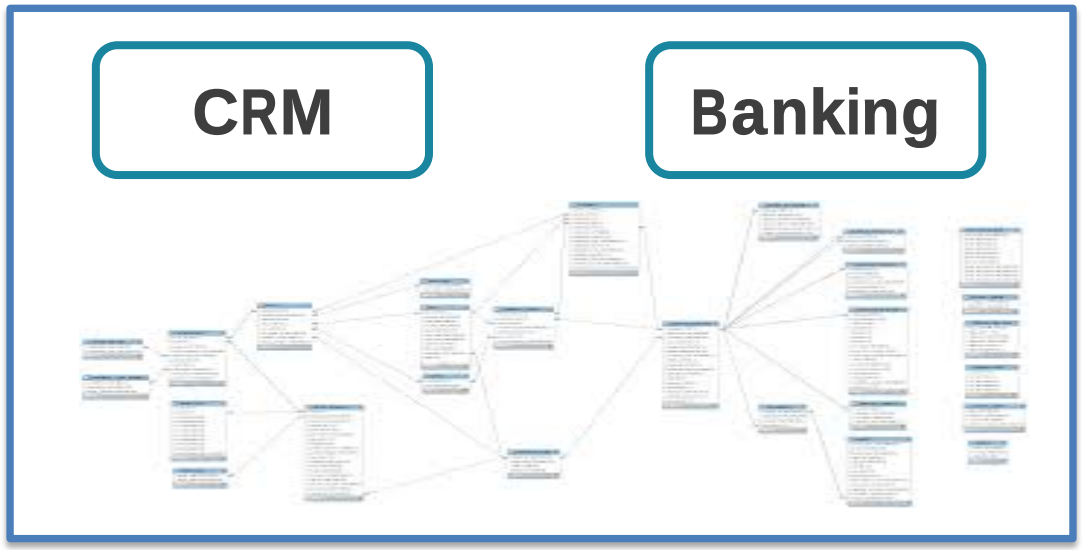
New Data

Twitter feed

Compliance Data

Trade Data

Boxed in

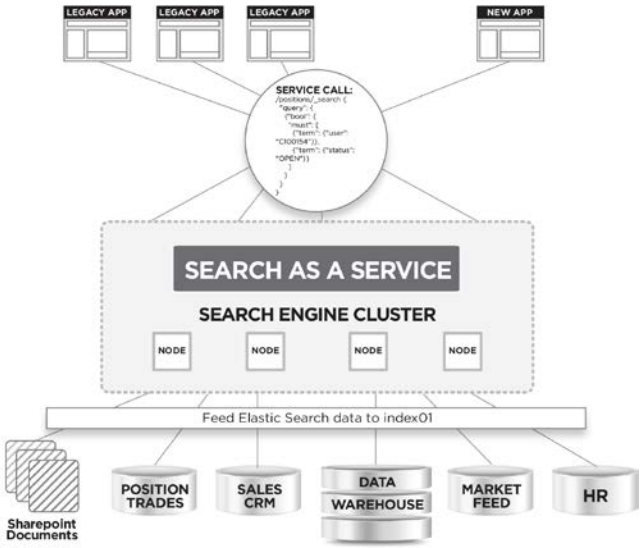


opencrowd

The real need is for systems that can adapt to new data

Search Based Architecture offers four key benefits making this pattern a promising solution for data intensive applications:

- Adaptive to new data
- Split second response time
- Very Intuitive end user experience
- Access to data can be offered as a service

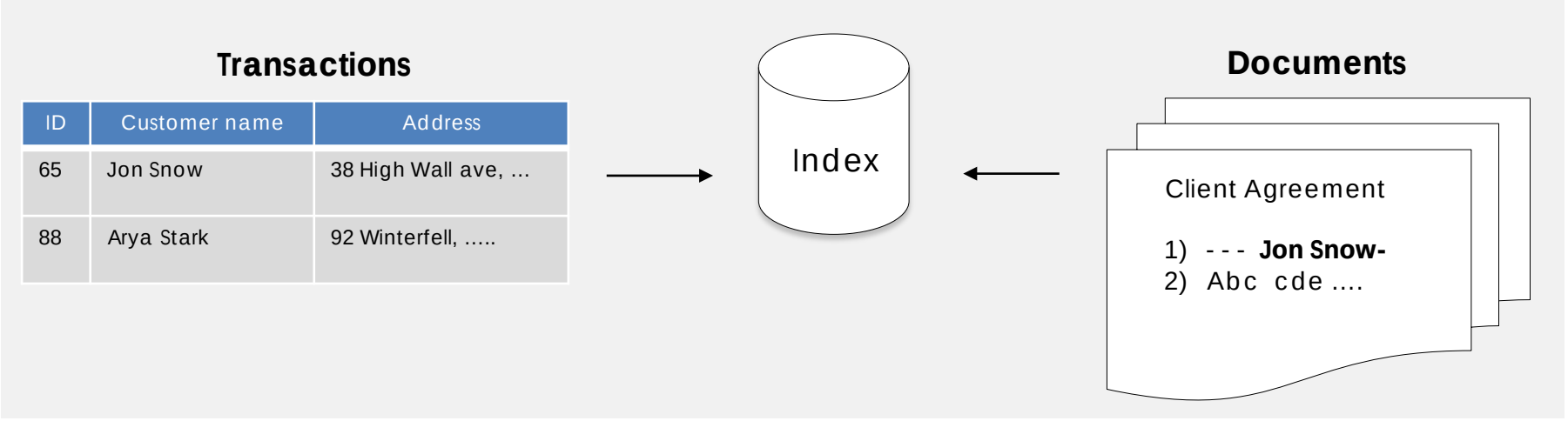


Adaptive to new data – Dynamic structure

The key difference in Search Based Applications is that information is primarily stored and retrieved from an INDEX

- Schema can grow dynamically thus enabling the relationship to evolve and making it easier to bring new data types

- Data types are mapped dynamically



opencrowd

Split second performance to retrieve data

- Search engines are built for extremely fast retrieval of information
- Information is stored in flat de-normalized structure making retrieval a lot less CPU intensive operation

Split second performance to retrieve data

- Most engines perform a wide range of calculations from a simple count to complex aggregations on the fly
- Queries previously deemed complex and resource intensive are a lot easier:

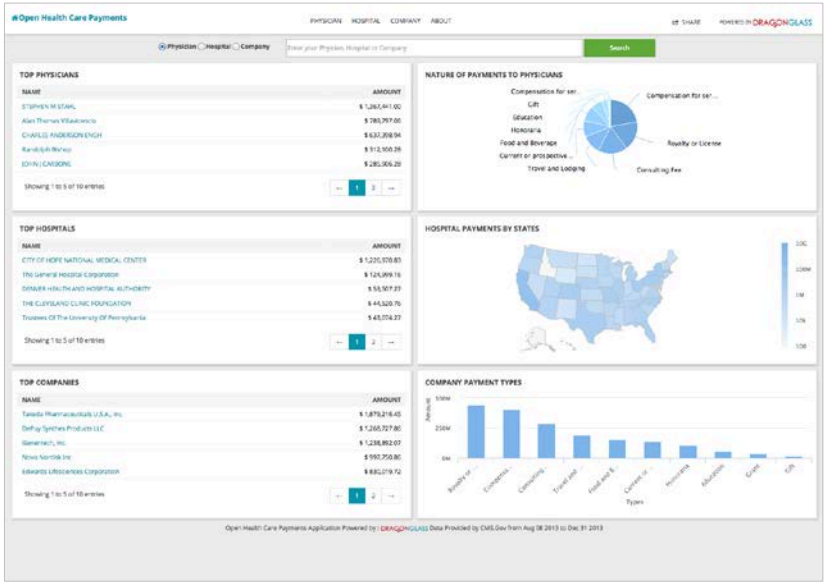
"Give me all client assets grouped by asset classes that have exposure to Greece and were purchased last year with total asset value greater than 100 million and exclude assets that belong in the sin category from this year"

Very Intuitive End User Experience

You can bring Google or Amazon style easy-to-use experience to enterprise applications such as:

- Auto suggest
- Auto complete
- Facets
- Alerts
- Fuzzy search
- Analytics on search results

Improve findability by guiding the user based on behavioral intelligence and trends.



Ex: Open Healthcare Payments Application

openpayments.dragonglass.me

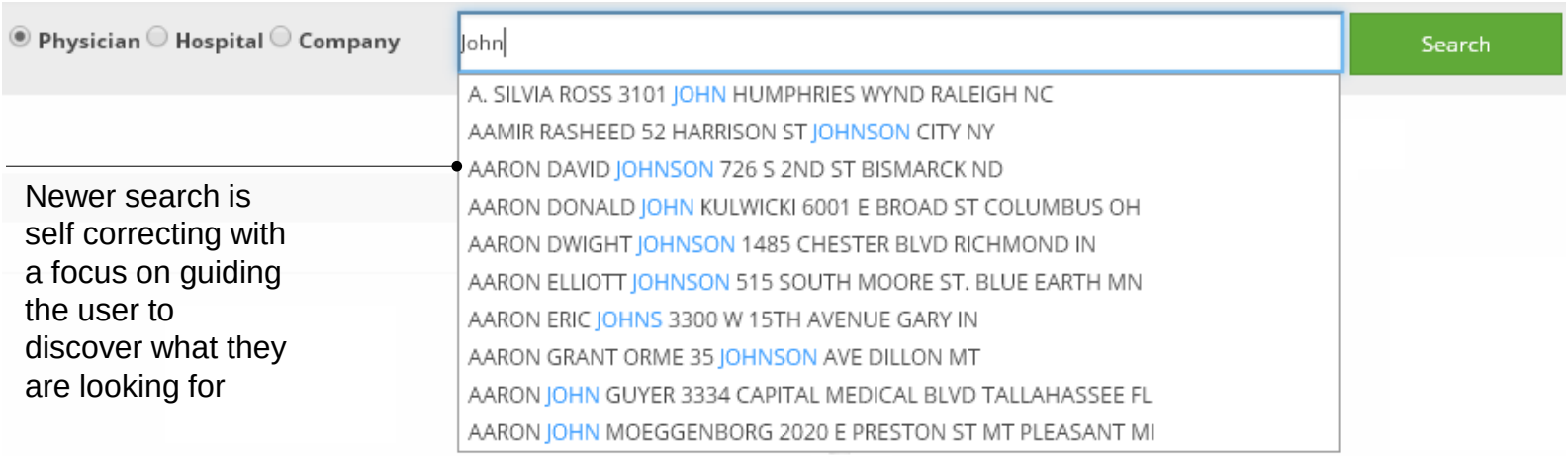
opencrowd

Before and After Search UX

Older form based search:

Older search requires prior knowledge and precise input which can be un-forgiving

Newer search with auto-suggest:

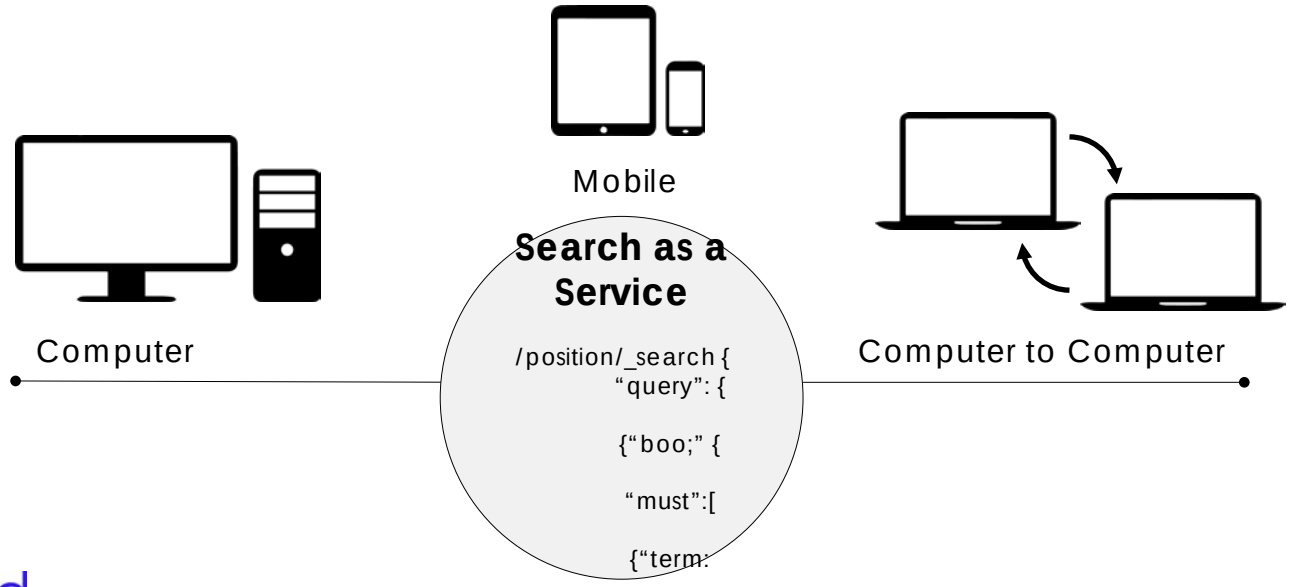


Newer search is self correcting with a focus on guiding the user to discover what they are looking for

opencrowd

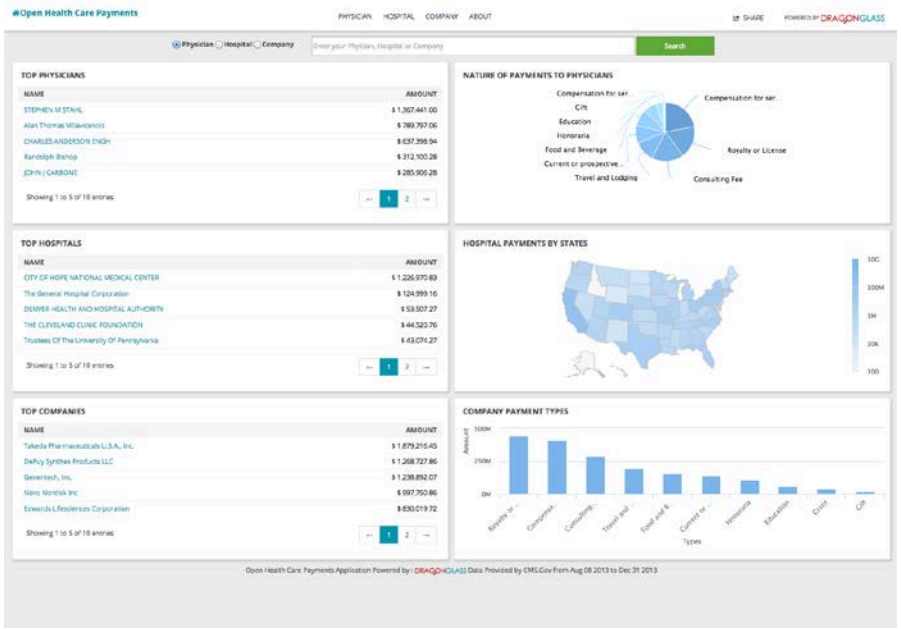
Access to data can be offered as a service

- Most search engines offer simple web services access to the Index, specifically REST
- REST based API makes it very easy to offer “Search as a Service”
- This micro service can be consumed by applications or other systems



opencrowd

A Sample Search Based application



The Open Heath Care Payments application uses DragonGlass to provide insight into payments to physicians and hospitals from medical companies. The application makes it easy to find physicians and hospitals as well as provides analysis of the payments they have received. Raw data for the application is publicly available at: openpaymentsdata.cms.gov site.

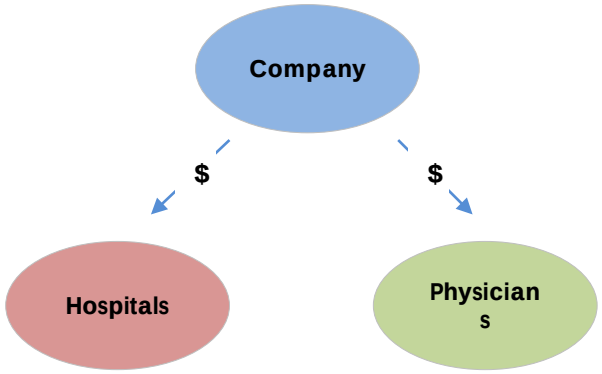
openpayments.dragonglass.me

opencrowd

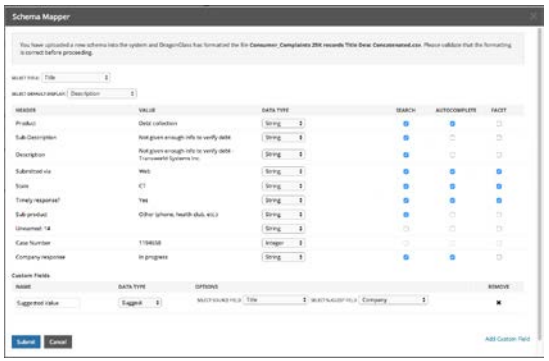
Open Healthcare Payment Application

opencrowd

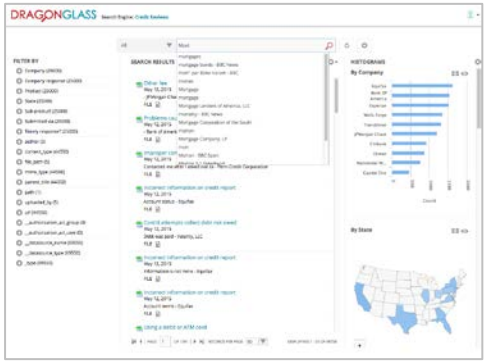
Building a Search Based Application



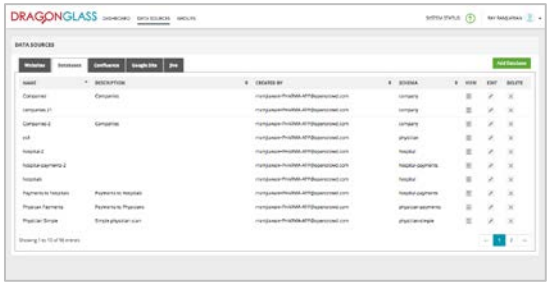
1. Model the Information



2. Create Index & Upload Data



3. Query the Information



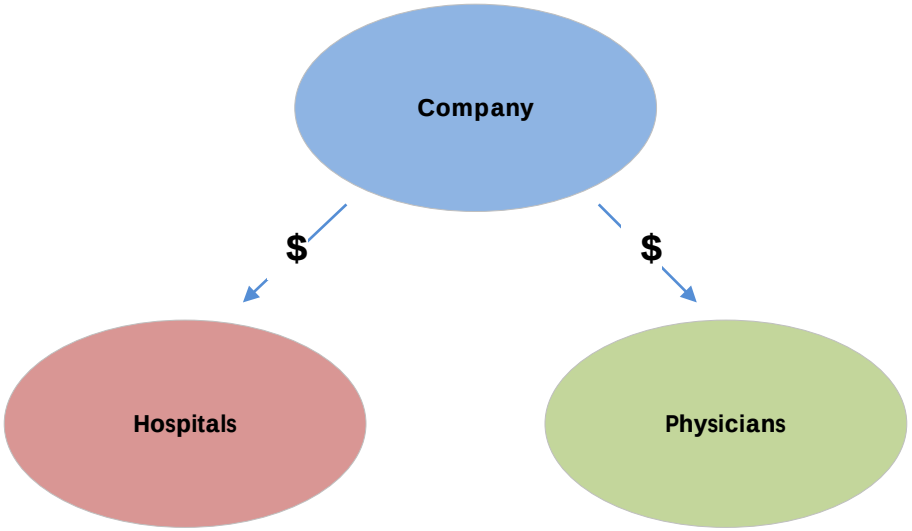
4. Enhancing the System with New Data

Note: In this exercise we will use Elastic Search Engine

opencrowd

Model your base entities

Just like with any other app, the data you use will impact the answers you can get. Since we wanted physician, hospital, and company data, we broke the documents up into those 3 entities:



Physician {"first name": "*STEPHEN*", "last name": *"BURKHART*", "state": *"TX"*, "zip code": *"78258"*, "specialty": *"Allopathic & Osteopathic Physicians*", "total_payment_amount": 7356275.69, ...

opencrowd

Create an index & upload the data

- You can either create a index with a schema or you can decide to use dynamic schema which is created as one uploads the data
- The data from cms.gov site is available as a giant three million record csv. To make it easier we loaded the data in a MySQL database.
- We wrote SQL queries to model the base entities and we set these queries as data sources in DragonGlass (basically REST API calls to Elastic Search)
- As the files from MySQL were uploaded/indexed in Elastic Search the data was ready to be searched

You can now query the information in multiple ways

- As a simple REST call from your browser
  http://openpayments.dragonglass.me:8080/
- From a programmable widget
- A instant data discovery search result page with facets, auto suggest, etc.
- A Micro service to be consumed by other systems

Search Discovery App

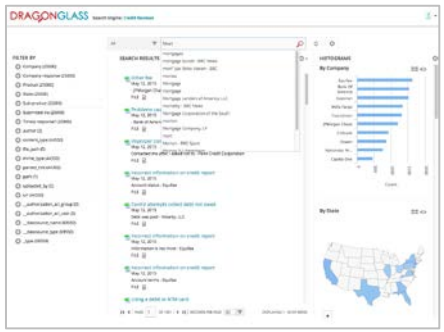


Chart widget

opencrowd

How do I bring new data in my current app?

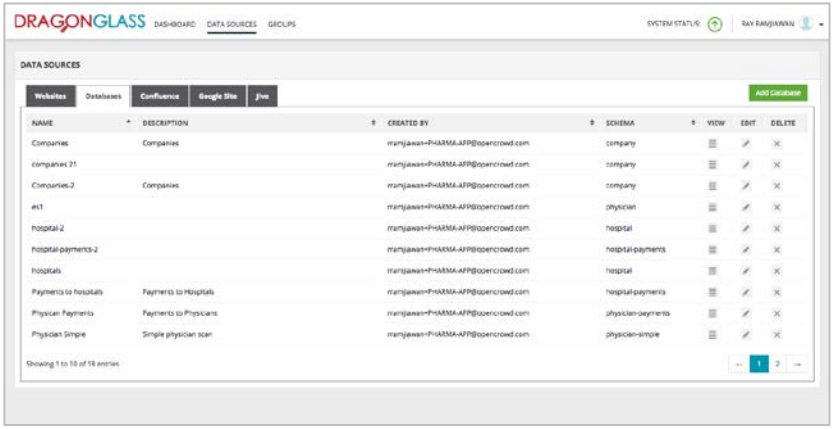
- For example: Associating state wide Medicare & Medicaid payment information with the OpenCMS data
- As all the base entities have a “state” field the initial association will happen automatically
- Once the new data is loaded all existing search queries, aggregations…will automatically include the new data in the result set

Physician {"first name": "*STEPHEN*", "last name": *"BURKHART*“, **"state": *"TX",*** "zip code": *"78258", ……..*

Medicare { "hospital_id": 5323, "physician_id": 223436, **"state": "TX"**, "city": "Houston", "payment_amount": 324.73……..

opencrowd

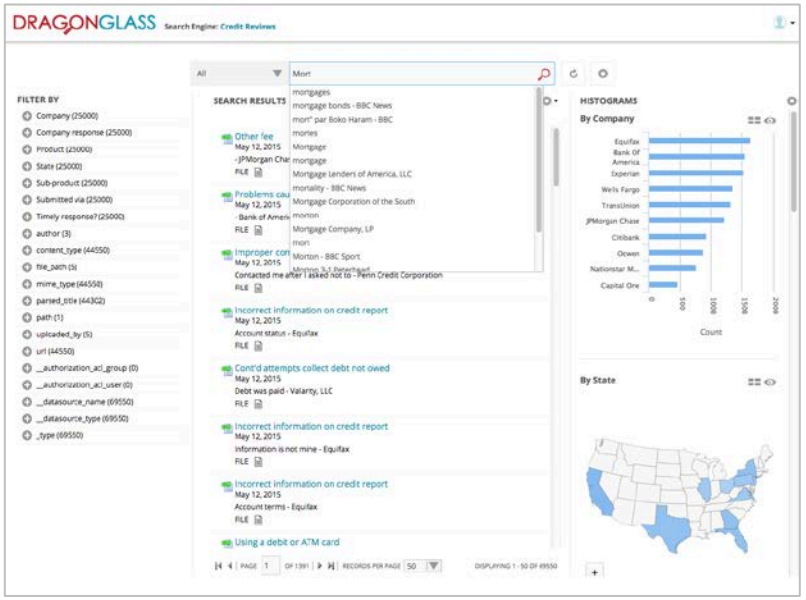
DragonGlass Admin UI – data upload



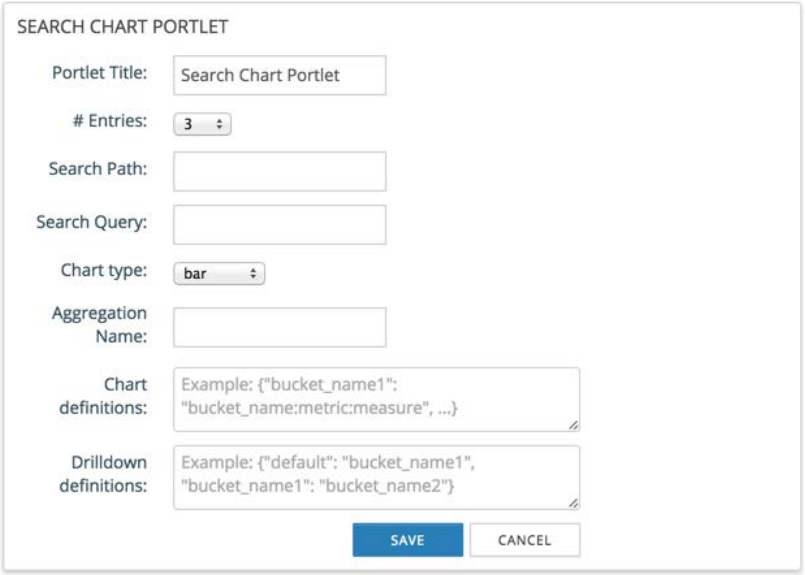
Search as a service REST API

http://openpayments.dragonglass.me:8080/_search/default/physician-simple/_search?aggs=paymaent_state_sort,payment_amount

Data Discovery Application

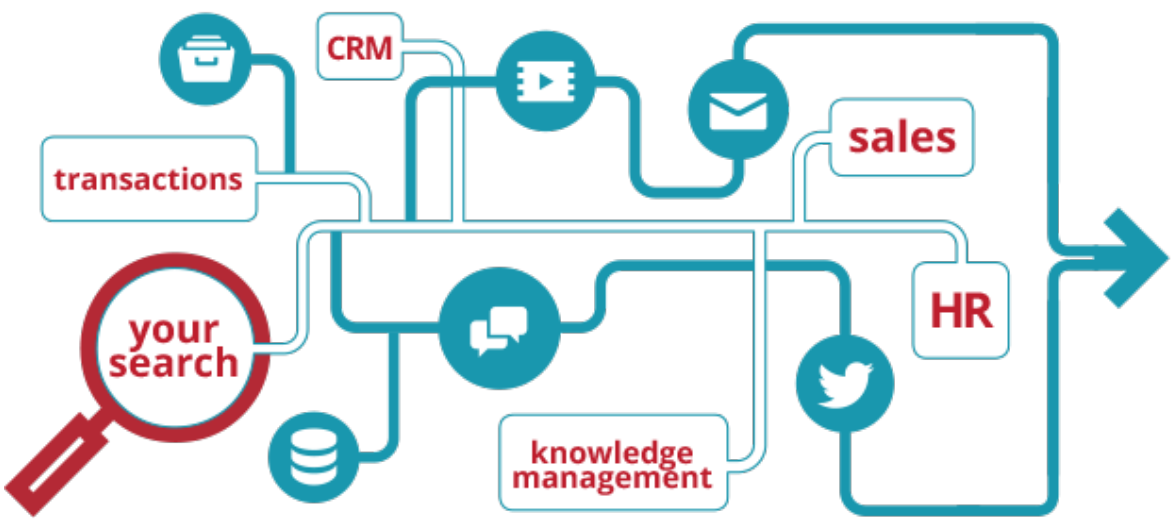


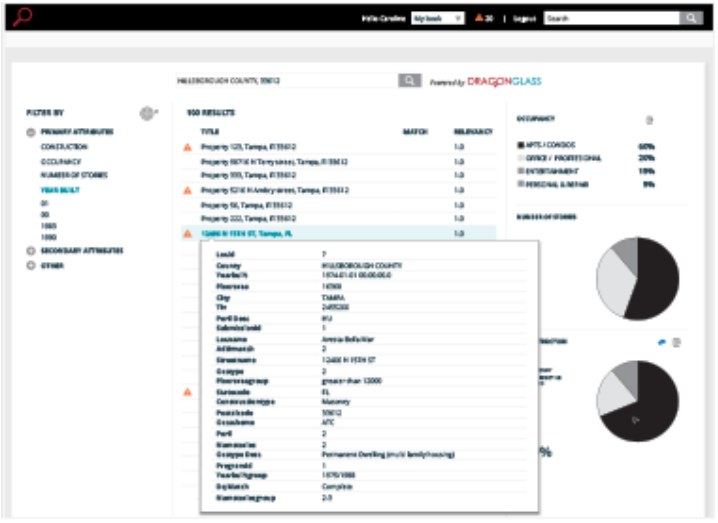
Programmable Widget



opencrowd

DragonGlass, Enabling Search Powered Applications





- Automates data extraction
- Creates a Index store
- Auto generates a data discovery application
- Provides dashboard widgets and API
- Admin services to manage and monitor the platform

opencrowd

Thank You

Visit us at:
DragonGlass.me

41 East 11th St, 11th Floor
New York, NY 10003
www.opencrowd.com

Phone: 917-770-9229
Fax: 212-768-1792
E-mail: info@opencrowd.com

Sushil Prabhu
Phone: 212-768-1730
sprabhu@opencrowd.com

opencrowd