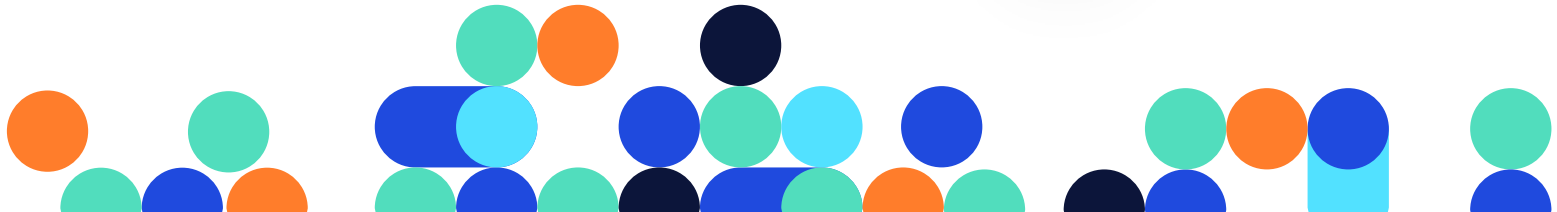
DataStax Developer Day

DATASTAX

DataStax Enterprise Search



What are we doing today?

- Explore the product catalog use case
- Discuss how your use case might be limited using just Cassandra
- Use DSE Search to perform queries on different columns
- Make changes to our DSE Search schema
- Dive into full text searching

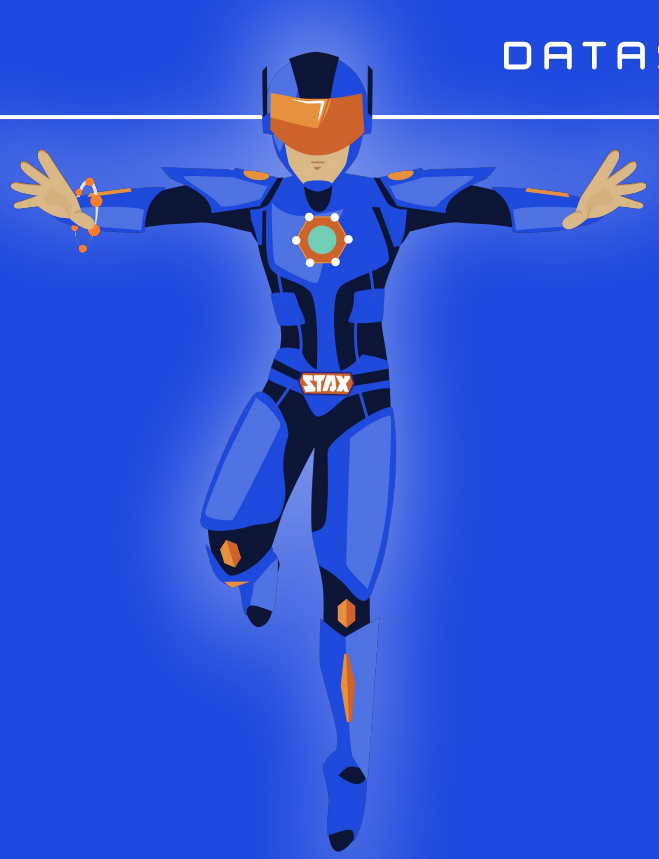
How's your Cassandra?

- We assume that you are somewhat familiar with Cassandra
- If you don't, the reasons to use Search may not make sense

DSE Search

DATASTAX

Product Catalog Use Case



What Functionality Do We Need?

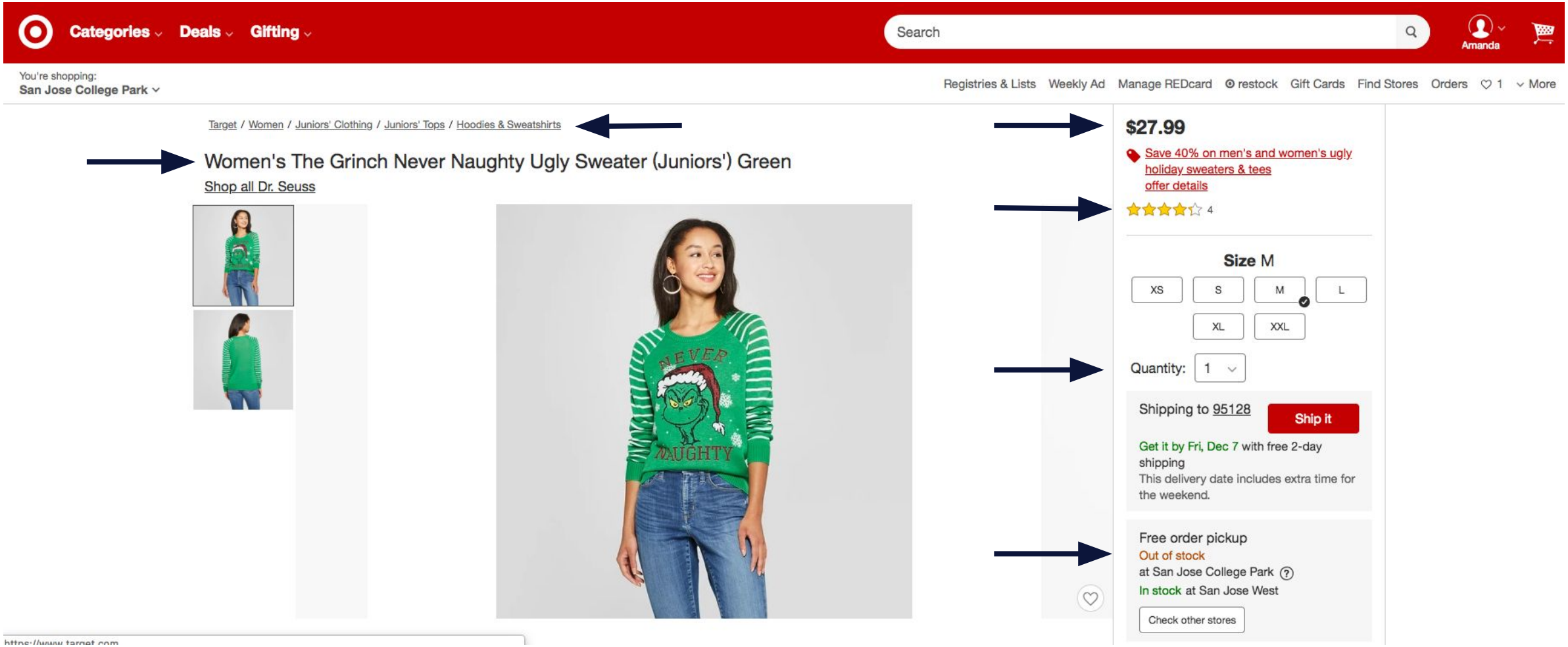
- Querying columns
- Search in text
- Sorting through results
- Counting
- Pagination

Searching

- Cassandra designed to allow on specific columns – partition key columns

- Some additional features to make querying more flexible
  - Secondary Indexes
  - Materialized Views
  - SASI – SSTable attached secondary indexes

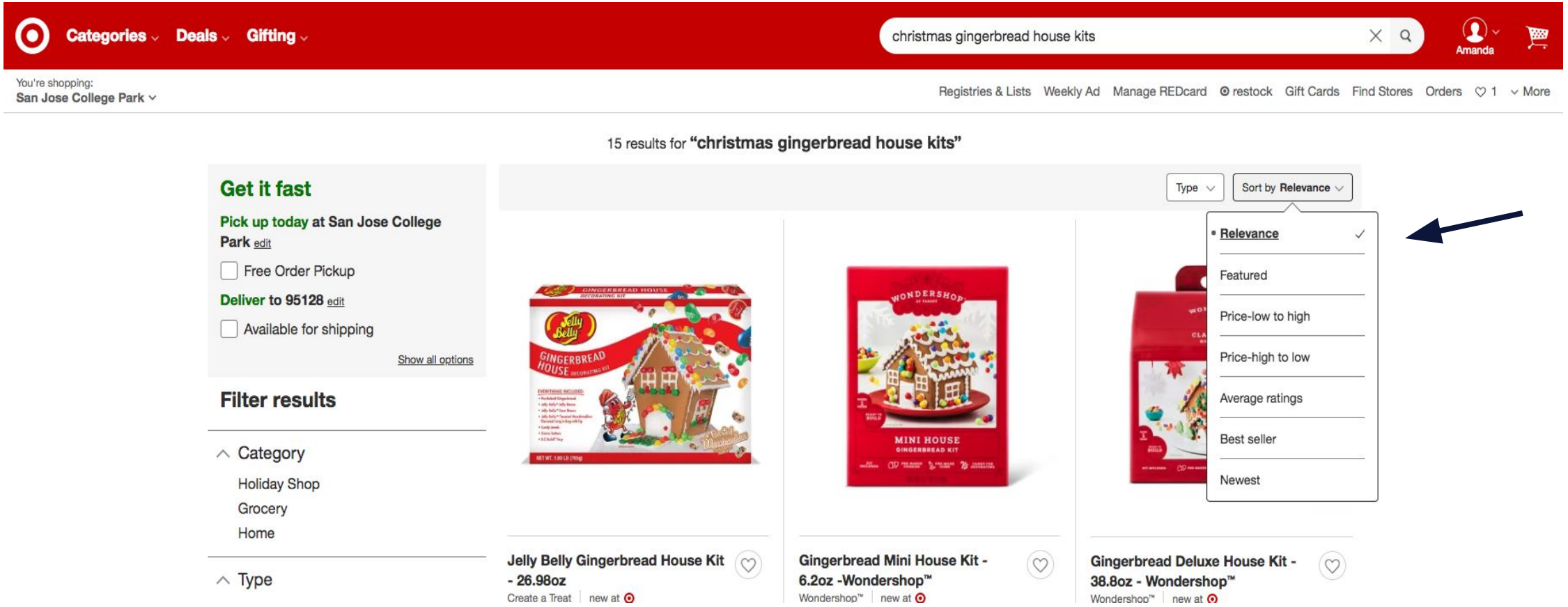
Searching


Categories
Deals
Gifting
Search
Amanda
You're shopping:
San Jose College Park
Registries & Lists
Weekly Ad
Manage REDcard
restock
Gift Cards
Find Stores
Orders
1
More
Target / Women / Juniors' Clothing / Juniors' Tops / Hoodies & Sweatshirts
Women's The Grinch Never Naughty Ugly Sweater (Juniors') Green
Shop all Dr. Seuss
$27.99
Save 40% on men's and women's ugly holiday sweaters & tees
offer details
4
Size M
XS
S
M
L
XL
XXL
Quantity: 1
Shipping to 95128
Ship it
Get it by Fri, Dec 7 with free 2-day shipping
This delivery date includes extra time for the weekend.
Free order pickup
Out of stock
at San Jose College Park
In stock at San Jose West
Check other stores

Sorting

- Limited in Cassandra by how data is stored to disk
- Can only sort within a partition
  - Need to search on partition key
- Only clustering columns are ordered
  - Need to know what columns that can order ahead of time
  - Need to re-create table if new ordering requirements
- Clustering columns sorted in groups following the primary key ordering
- Cannot arbitrarily change clustering column order
  - Depends on the order of the proceeding clustering columns

Sorting


Categories
Deals
Gifting
christmas gingerbread house kits
Amanda
You're shopping:
San Jose College Park
Registries & Lists
Weekly Ad
Manage REDcard
restock
Gift Cards
Find Stores
Orders
1
More
15 results for "christmas gingerbread house kits"
Get it fast
Pick up today at San Jose College Park edit
Free Order Pickup
Deliver to 95128 edit
Available for shipping
Show all options
Filter results
Category
Holiday Shop
Grocery
Home
Type
Type
Sort by Relevance
Relevance
Featured
Price-low to high
Price-high to low
Average ratings
Best seller
Newest
Jelly Belly Gingerbread House Kit - 26.98oz
Create a Treat | new at
Gingerbread Mini House Kit - 6.2oz -Wondershop™
Wondershop™ | new at
Gingerbread Deluxe House Kit - 38.8oz - Wondershop™
Wondershop™ | new at

Sorting

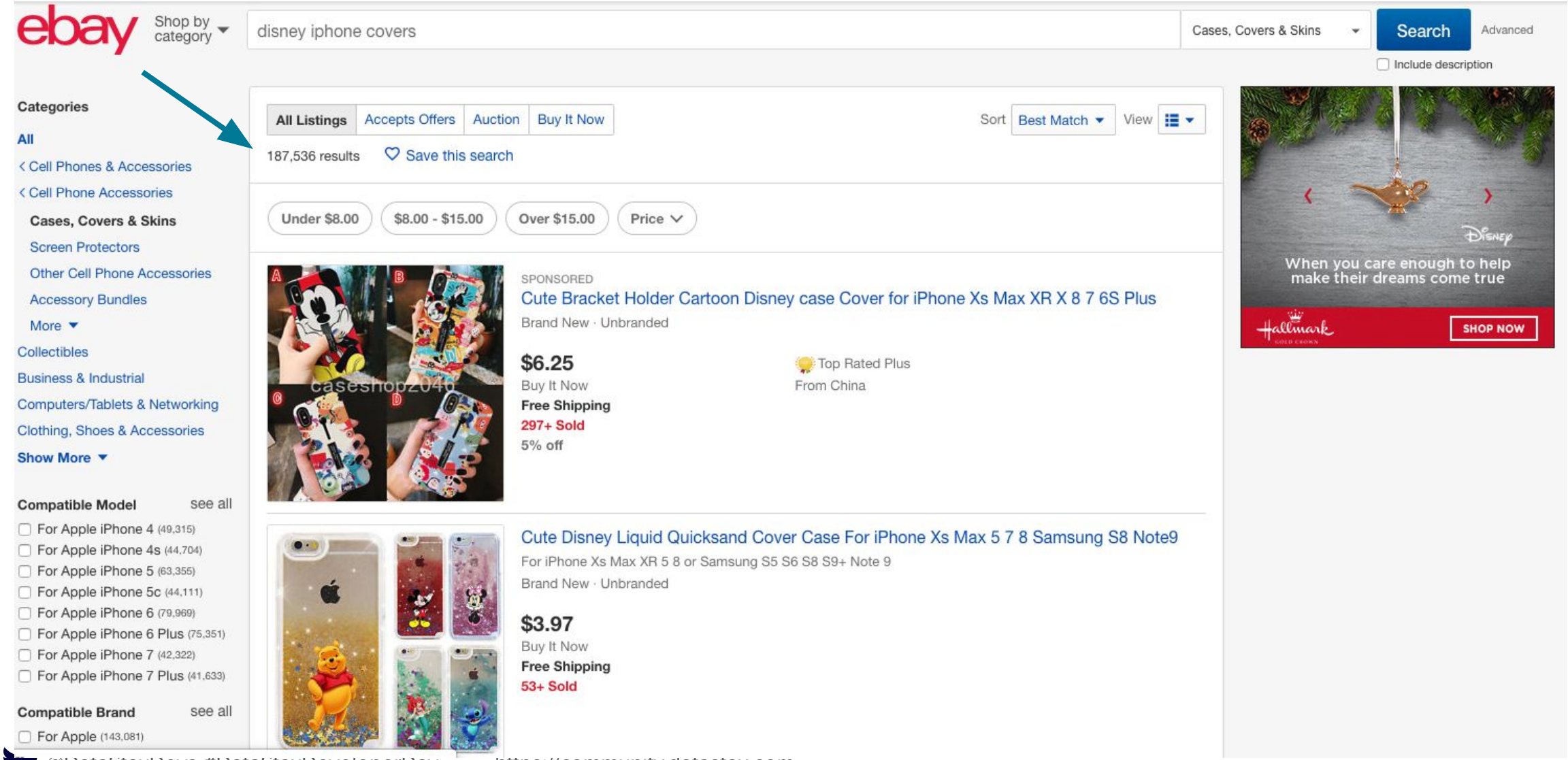
- Limited in Cassandra by how data is stored to disk
- Can only sort within a partition
  - Need to search on partition key
- Only clustering columns are ordered
  - Need to know what columns that can order ahead of time
  - Need to re-create table if new ordering requirements
- Clustering columns sorted in groups following the primary key ordering
- Cannot arbitrarily change clustering column order
  - Depends on the order of the proceeding clustering columns

Counting

- Cassandra does have COUNT, but…
- Need to read through partitions to get the count
- If not restricting to a partition, that means doing a full table scan
- Also no way to count how much each value shows up in a column

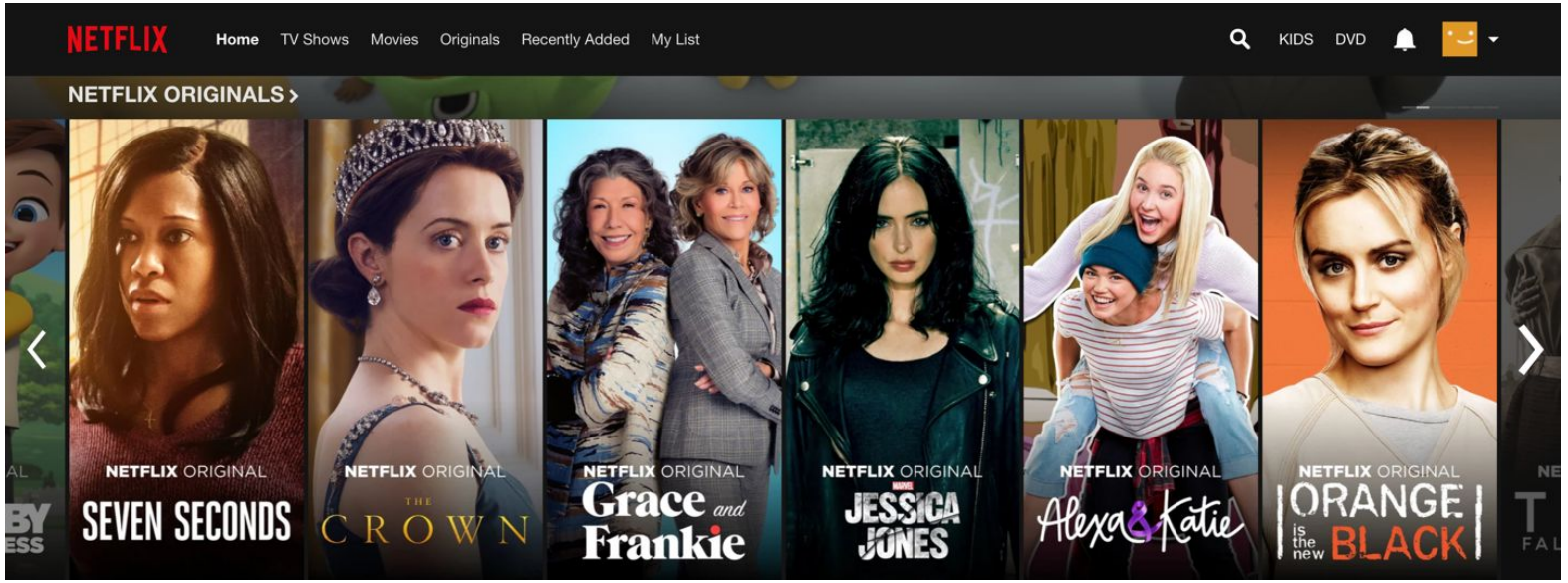
https://academy.datastax.com/support-blog/counting-keys-might-well-be-counting-stars

Counting

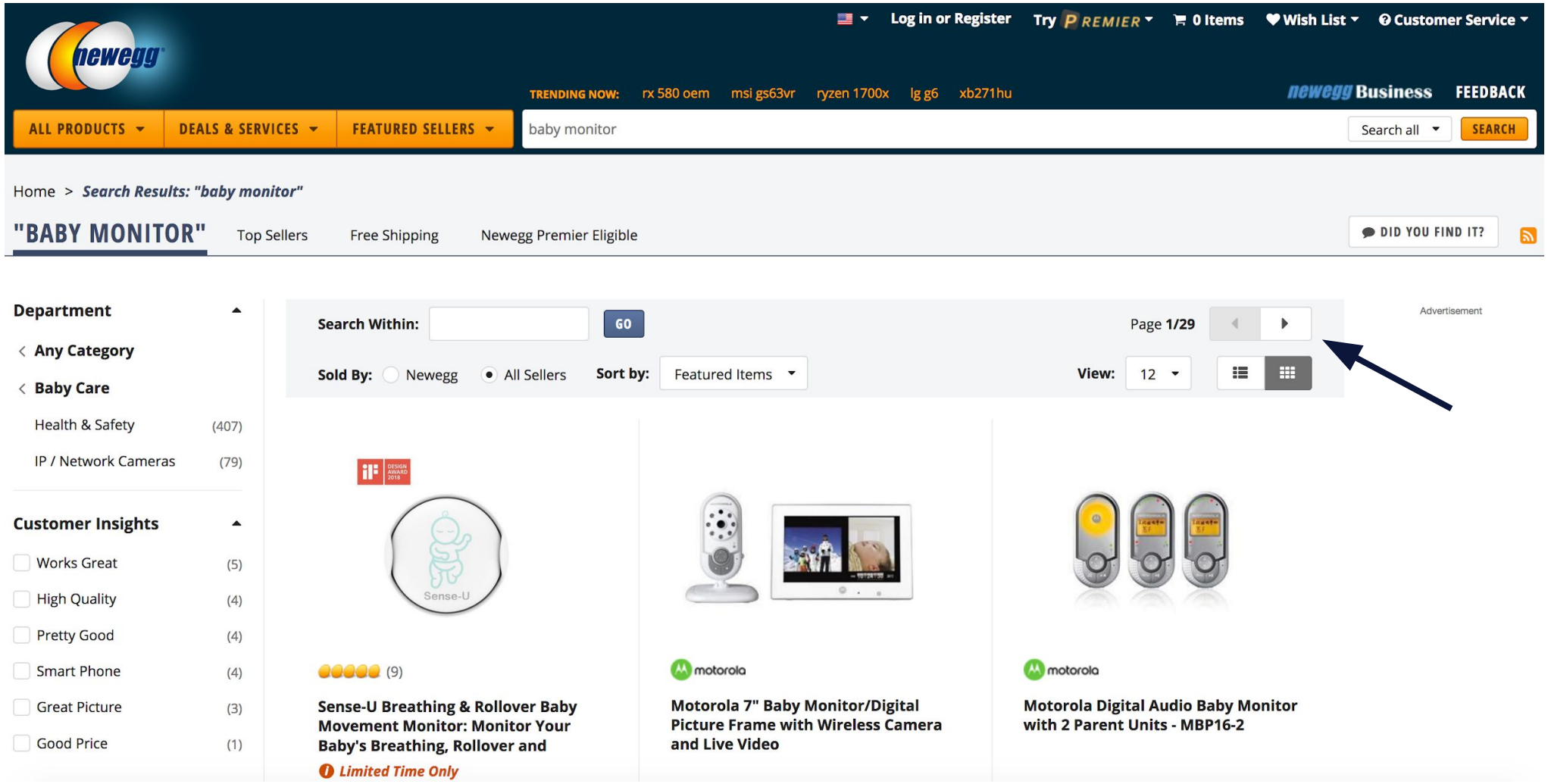

ebay
Shop by category
disney iphone covers
Cases, Covers & Skins
Search
Advanced
Include description
Categories
All
Cell Phones & Accessories
Cell Phone Accessories
Cases, Covers & Skins
Screen Protectors
Other Cell Phone Accessories
Accessory Bundles
More
Collectibles
Business & Industrial
Computers/Tablets & Networking
Clothing, Shoes & Accessories
Show More
Compatible Model see all
For Apple iPhone 4 (49,315)
For Apple iPhone 4s (44,704)
For Apple iPhone 5 (63,355)
For Apple iPhone 5c (44,111)
For Apple iPhone 6 (79,969)
For Apple iPhone 6 Plus (75,351)
For Apple iPhone 7 (42,322)
For Apple iPhone 7 Plus (41,633)
Compatible Brand see all
For Apple (143,081)
All Listings
Accepts Offers
Auction
Buy It Now
Sort Best Match
View
187,536 results
Save this search
Under $8.00
$8.00 - $15.00
Over $15.00
Price
SPONSORED
Cute Bracket Holder Cartoon Disney case Cover for iPhone Xs Max XR X 8 7 6S Plus
Brand New · Unbranded
$6.25
Buy It Now
Free Shipping
297+ Sold
5% off
Top Rated Plus
From China
Cute Disney Liquid Quicksand Cover Case For iPhone Xs Max 5 7 8 Samsung S8 Note9
For iPhone Xs Max XR 5 8 or Samsung S5 S6 S8 S9+ Note 9
Brand New · Unbranded
$3.97
Buy It Now
Free Shipping
53+ Sold
When you care enough to help make their dreams come true
Hallmark
SHOP NOW

Pagination

- Affected by limitations of counting, cannot efficiently do offset paging
- Cassandra driver can do cursor-based paging through results
- Essentially can only go forward or back from the current page



Pagination

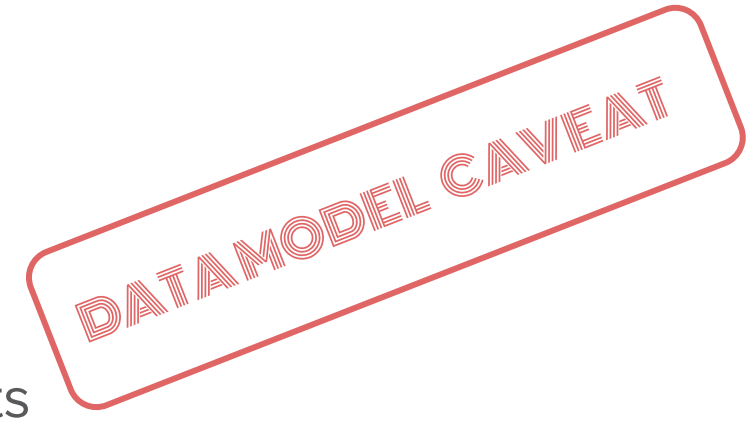

Log in or Register
Try PREMIER
0 Items
Wish List
Customer Service
TRENDING NOW: rx 580 oem msi gs63vr ryzen 1700x lg g6 xb271hu
newegg Business
FEEDBACK
ALL PRODUCTS
DEALS & SERVICES
FEATURED SELLERS
baby monitor
Search all
SEARCH
Home > Search Results: "baby monitor"
"BABY MONITOR"
Top Sellers
Free Shipping
Newegg Premier Eligible
DID YOU FIND IT?
Department
Any Category
Baby Care
Health & Safety (407)
IP / Network Cameras (79)
Customer Insights
Works Great (5)
High Quality (4)
Pretty Good (4)
Smart Phone (4)
Great Picture (3)
Good Price (1)
Search Within:
GO
Page 1/29
Advertisement
Sold By: Newegg All Sellers
Sort by: Featured Items
View: 12
Sense-U
(9)
Sense-U Breathing & Rollover Baby Movement Monitor: Monitor Your Baby's Breathing, Rollover and
Limited Time Only
motorola
Motorola 7" Baby Monitor/Digital Picture Frame with Wireless Camera and Live Video
motorola
Motorola Digital Audio Baby Monitor with 2 Parent Units - MBP16-2

Introducing DSE Search

- Apache Solr™
  - Open-source enterprise search platform
  - Provides tools and an interface for running search queries
- Apache Lucene™
  - Text indexing and search engine library
  - The core of the indexing and search capabilities available with DSE Search

DSE Search

- Rich functionality not available in Cassandra
- More convenient way to access data
  - Doesn't require complex data models or data duplication
  - Less work needed on the application side to format results
- Accessible from CQL
  - Core functionality using pure CQL syntax
  - More features available using the solr_query column
  - Changes to the search index schema or configuration



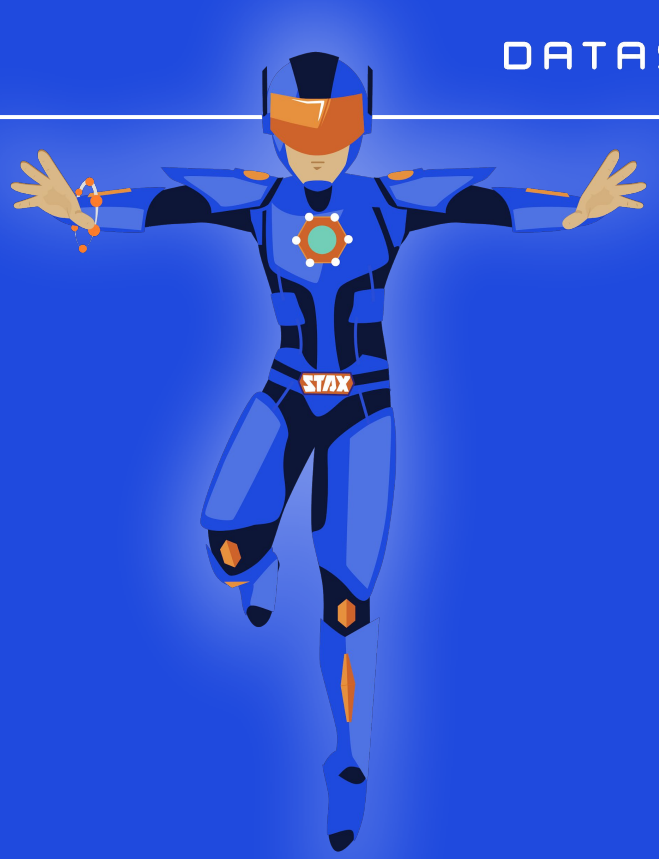
Search features

- CQL enabled search
- SQL “like” syntax
- Filtering / Matching
- Allows indexing on non-primary key columns
- Indexing and Lucene under the hood
- Tight integration with Apache Cassandra in DSE
  - “-s” switch to “dse cassandra”
- CQL auto filters and uses search indexes when combined with DSE Analytics (Spark)
- Geospatial queries

DSE Search

DATASTAX

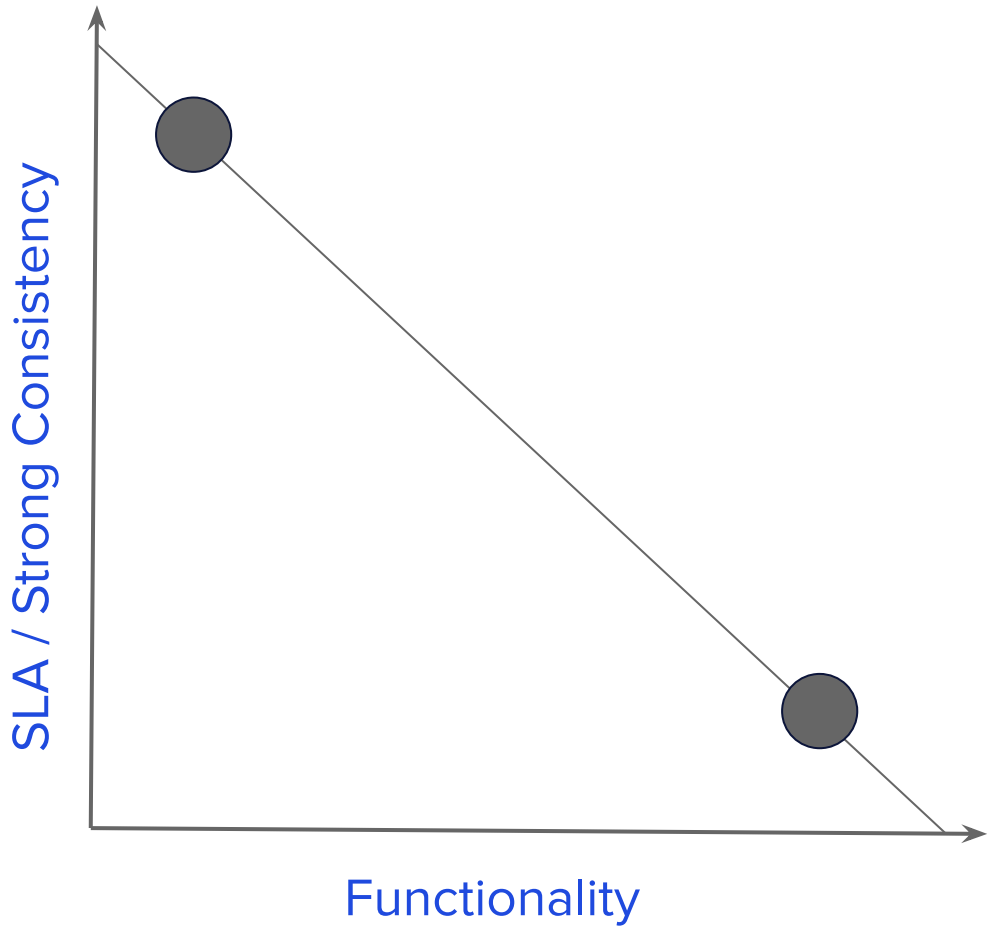
Data model and trade-offs



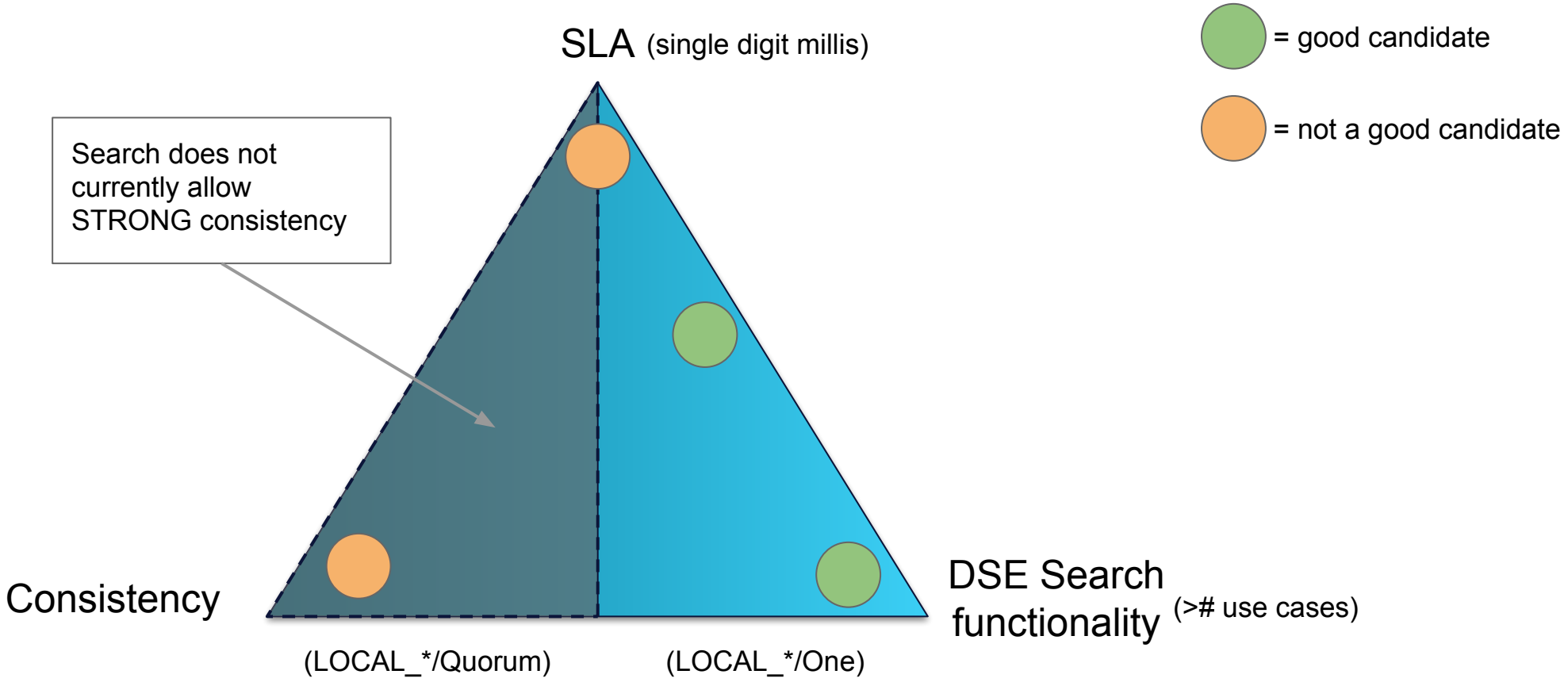
Data model

- DSE Search is not here to bail you out of a bad data model
- Still need to denormalize tables
- There are trade-offs and balance, depends on your needs

Balancing act

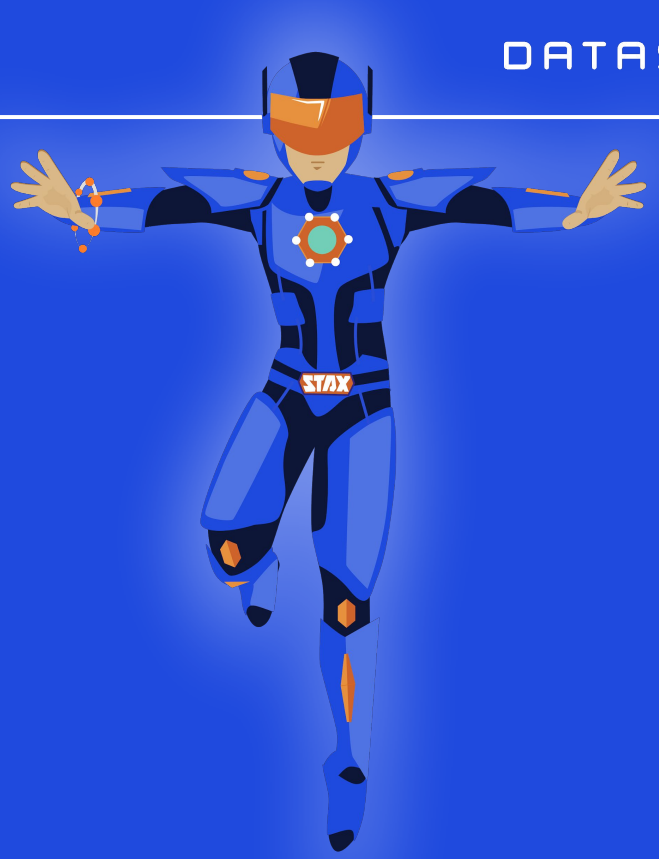


Balancing act



DATASTAX

Getting Started with Search



Creating a search index

- Use CQL to create the search index
  - Command runs on all Search nodes in the datacenter
  - Uses a default search schema and config, which can be altered later
  - Also indexes existing table data
  - New data that you add to the table is automatically indexed

```
// Index all columns in the table
CREATE SEARCH INDEX ON keyspace.table;

// Only index certain columns in the table
CREATE SEARCH INDEX ON keyspace.table WITH COLUMNS column1, column2, ...;
```

CQL Search Query

- Accessing a query using search index can be done through CQL
- Will execute query just using Cassandra, if possible
- Otherwise will use the search index

```
SELECT * FROM keyspace.table WHERE predicate1 AND predicate2 AND ...;

SELECT col1, col2, ... FROM keyspace.table WHERE predicate1 AND predicate2 AND ...;

SELECT COUNT(*) FROM keyspace.table WHERE predicate1 AND predicate2 AND ...;

SELECT * FROM keyspace.table WHERE predicate1 AND predicate2 AND ... LIMIT #rows;
```

CQL Query Predicates

```
CREATE TABLE killrvideo.users (
    userid UUID,
    created_date TIMESTAMP,
    email TEXT,
    firstname TEXT,
    lastname TEXT,
    phone_number SET<TEXT>
    PRIMARY KEY ((userid))
);
```

- Equality and Inequality

```
SELECT * FROM killrvideo.users
  WHERE email = 'eboyeri5@aol.com';
SELECT * FROM killrvideo.users
  WHERE email != 'eboyeri5@aol.com';
```

- Range

```
SELECT * FROM killrvideo.users
  WHERE created_date >= '2018-04-01'
  AND created_date < '2018-05-01';
```

- Contains

```
SELECT * FROM killrvideo.users WHERE
  phone_number CONTAINS '650-389-6000';
```

- In

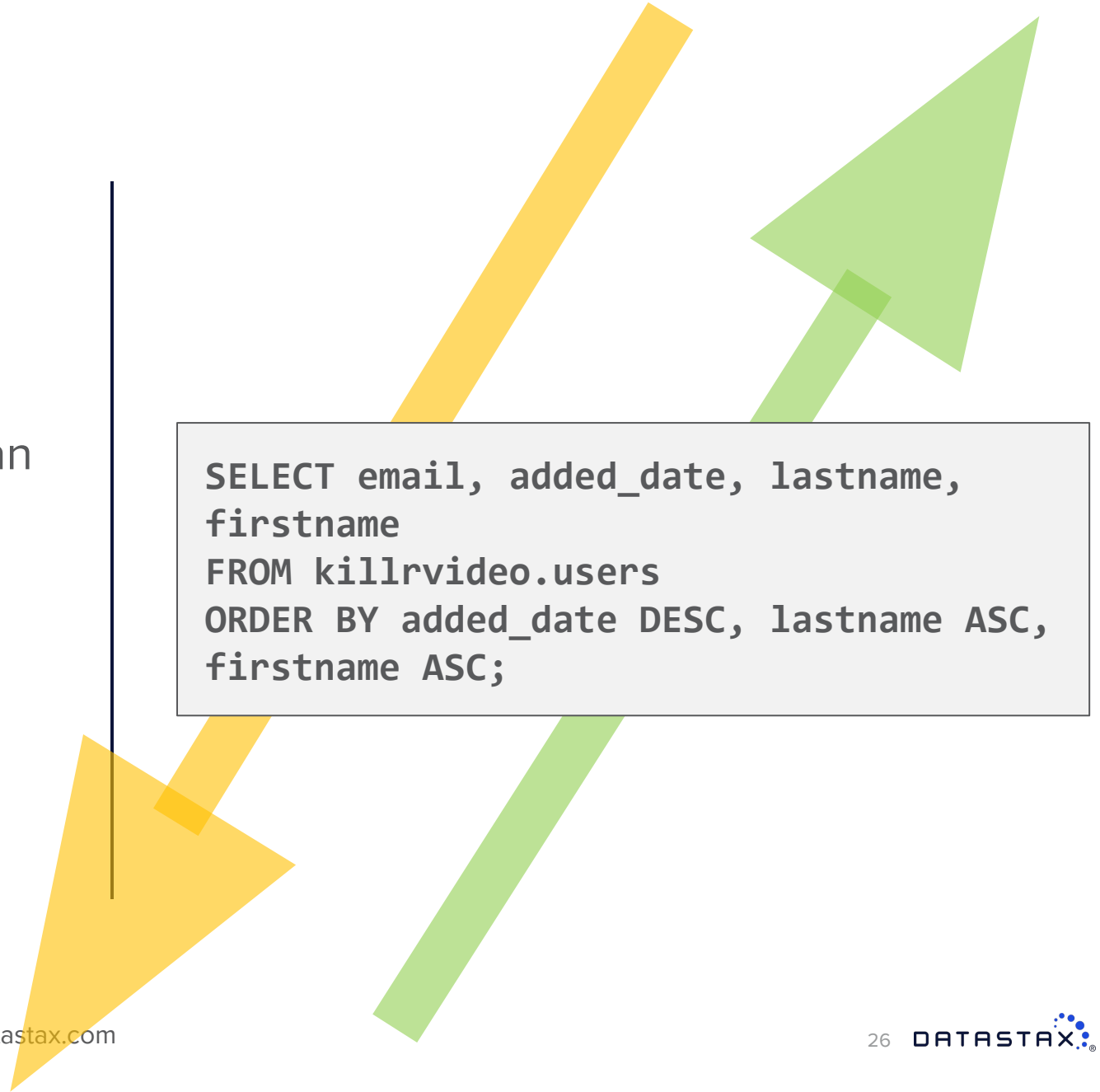
```
SELECT * FROM killrvideo.users
WHERE firstname IN ('Beauregard','Muffin');
```

- Like

```
SELECT * FROM killrvideo.users
WHERE lastname LIKE 'McD%';
```

CQL Query

- Sort by any column
- Text can't be sorted by default, but can be changed in the search index schema
- Ascending (default) or descending order

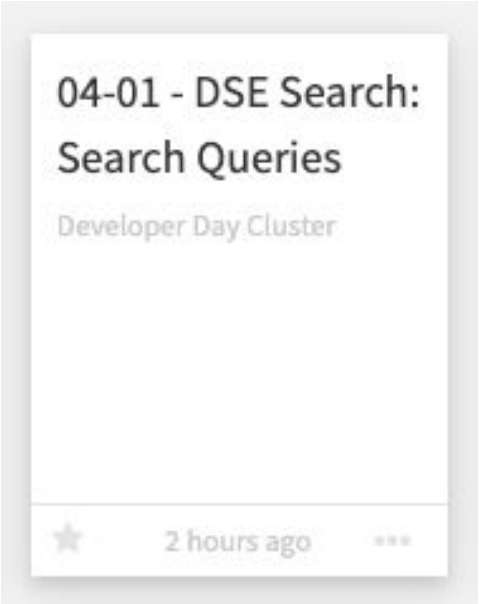


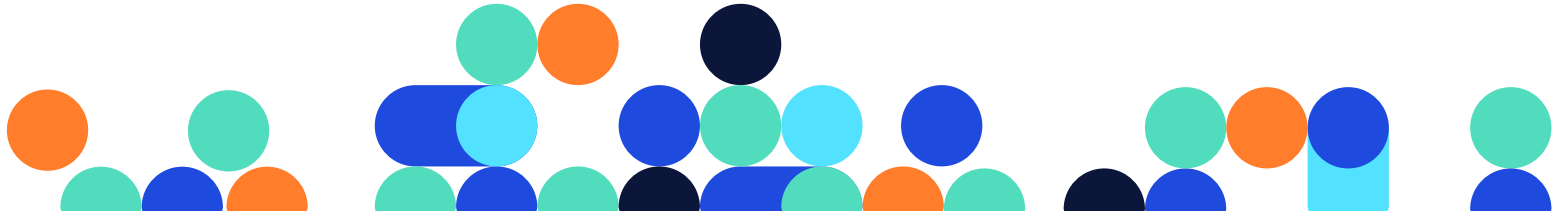
```
SELECT email, added_date, lastname,
firstname
FROM killrvideo.users
ORDER BY added_date DESC, lastname ASC,
firstname ASC;
```

Time for an exercise!

DATASTAX

Search Queries





DSE Search

DATASTAX

Text Search

Text Search

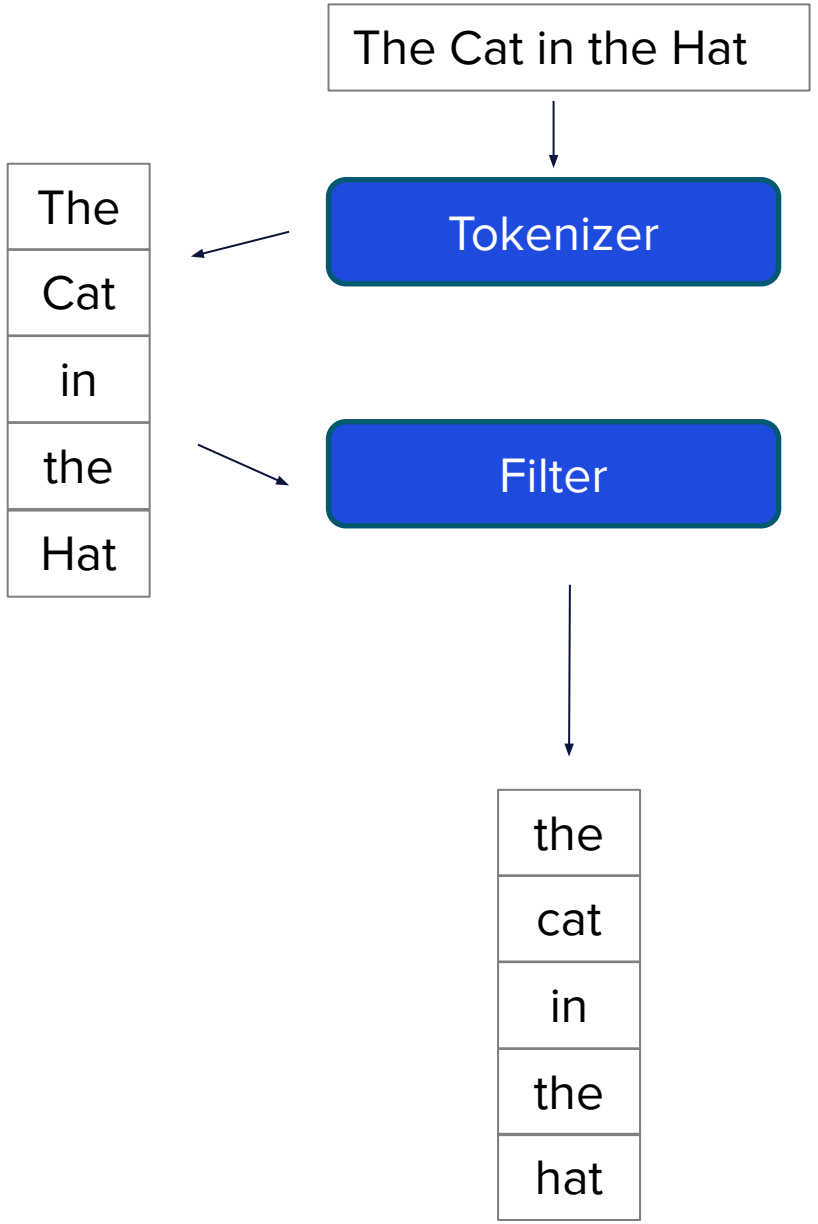
- One of DSE Search's strengths is in full-text search
- Retrieves results based on how well the text matches the search parameters
  - Calculates a relevancy score
  - Only includes the rows with the highest score are included in the search results
- Uses the more expressive Lucene query syntax instead of just CQL

Field Types

- The Search equivalent to data types found in the search index schema
- Cassandra data types map to a corresponding field type
- Some data types may have several compatible field types
  - For the Cassandra TEXT data type, you can use:
    - StrField (default)
    - TextField (text search capabilities)
- Power users can even create fully customizable field types

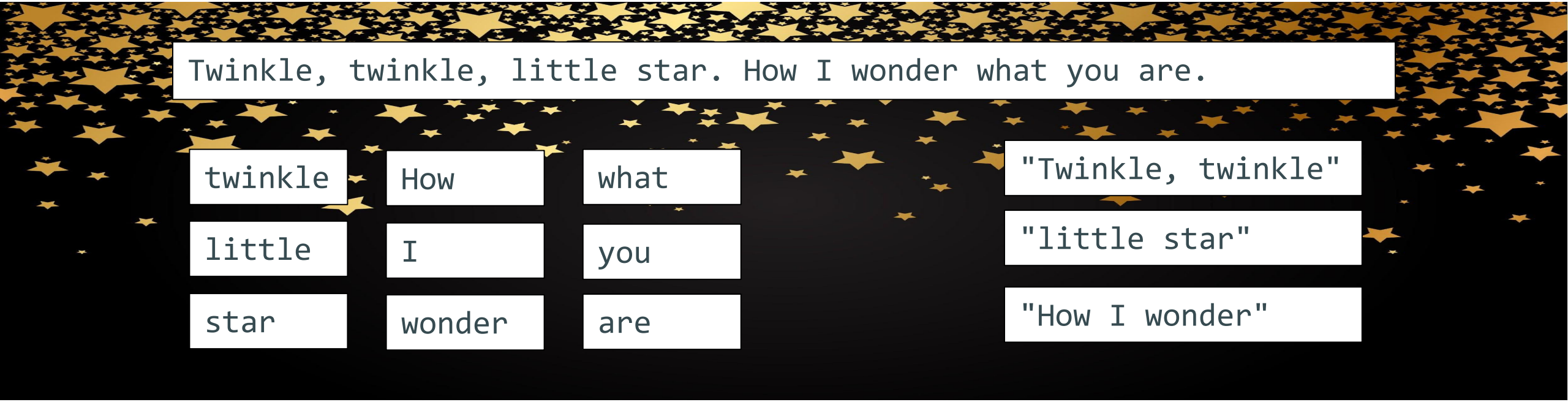
TextField

- TextField is processed as it is indexed
  - Analysis Chain
  - Tokenizer – breaks up text into tokens
  - Filter – performs some sort of processing
  - Resulting terms is what is indexed
- Search parameters also go through analysis chain
  - Compared against the indexed terms
  - Matching rows included in the results



Terms and Phrases

- Term – Tokenized data, or words, from text analysis or search input
- Phrase – Terms that are positioned in a certain order



Search index schema

- Written and stored as a XML file
- Can edit using CQL, or by uploading the new schema XML file

```
DESCRIBE ACTIVE SEARCH INDEX SCHEMA ON keyspace.table;   // CQLSH only
DESCRIBE PENDING SEARCH INDEX SCHEMA ON keyspace.table;  // CQLSH only
```

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<schema name="autoSolrSchema" version="1.5">
  <types>
    <fieldType class="org.apache.solr.schema.StrField" name="StrField"/>
    :
  </types>
  <fields>
    <field indexed="true" multiValued="false" name="title" type="StrField"/>
    :
  </fields>
  <uniqueKey>video_id</uniqueKey>
</schema>
```

Making changes to the schema

- We need to define our TextField and change the field type for our fields
- Schema with changes that haven't been applied yet is PENDING
- Currently running schema is ACTIVE

```
ALTER SEARCH INDEX SCHEMA ON keyspace.table
ADD fieldType[@name='TextField', @class='solr.TextField']
WITH '{"analyzer": [{
          "tokenizer": {"class": "solr.StandardTokenizerFactory"},
          "filter": {"class":"solr.LowerCaseFilterFactory"}  }]}';

ALTER SEARCH INDEX SCHEMA ON keyspace.table
SET field[@name='name']@type='TextField';
```

Applying changes to the search index

- Reload the search schema to apply the changes to the schema
  - Afterwards, PENDING schema replaces the ACTIVE schema

- Rebuild the search index to reindex the existing data in the table
  - Current indexes would not match the new schema
  - Not needed if changes only made to the search index config

```
RELOAD SEARCH INDEX ON keyspace.table;
REBUILD SEARCH INDEX ON keyspace.table;
```

Using solr_query in CQL

- solr_query is a pseudo-column created with the search index
- Set a query string to the solr_query column in the WHERE clause

```
SELECT select-clause FROM keyspace.table WHERE solr_query = 'query';
```

  - Passed to DSE Search / Solr to be executed
  - Uses Apache Lucene query syntax
  - Search results return to Cassandra, and then retrieves the actual row and column data

Lucene Query syntax in a nutshell

- Search all the things: *:*
- Term search: field-name:term

```
SELECT * FROM killrvideo.videos WHERE solr_query = 'name:cassandra';
```

- Phrase search: field-name:"term term term"

```
SELECT * FROM killrvideo.videos WHERE solr_query = 'name:"Distributed Data Show"';
```

Lucene Query syntax in a nutshell (continued)

- Multiple terms: field-name:(term OR (term AND term))
  - OR / AND must be capitalized!

```
SELECT * FROM killrvideo.users WHERE solr_query = 'name:(Jack OR Jill)';
```

- Multiple fields: field-name:term AND (field-name:term OR field-name:term)

```
SELECT * FROM killrvideo.videos WHERE solr_query = 'name:something AND tags:cats';
```

- Range search: field-name:(1 TO 0] ( ) - exclusive bounds, [ ] - inclusive bounds

```
SELECT * FROM killrvideo.videos WHERE solr_query = 'year:[2017 TO *]'
```

Levenshtein Distance

- Measure of how similar two strings are
- Based on the number of edits for one string to match the other
- Used by both fuzzy search and proximity search

```
// Distance between the word kitten and sitting is 3

Edit 1:  kitten → sitten (substitution of "s" for "k")

Edit 2:  sitten → sittin (substitution of "i" for "e")

Edit 3:  sittin → sitting (insertion of "g" at the end)
```

Fuzzy Search

- Add ~ at the end of a term
- Degree of similarity to the term is controlled by adding a value after the ~
  - This optional parameter can be 1 or 2
  - Represents the max number of edits to the indexed term

```
SELECT * FROM keyspace.table WHERE solr_query = 'field:term~#';
```

```
seven~1
```



```
The Magnificent Seven
Se7en
The Even Stevens Movie
```

Proximity Search

- Add ~ at the end of a phrase
- Degree of similarity controlled by adding a value after the ~
- Represents the maximum distance that terms in the phrase can be apart

```
SELECT * FROM keyspace.table WHERE solr_query = 'field:"phrase"~#';
```

```
"the road"~3'
```

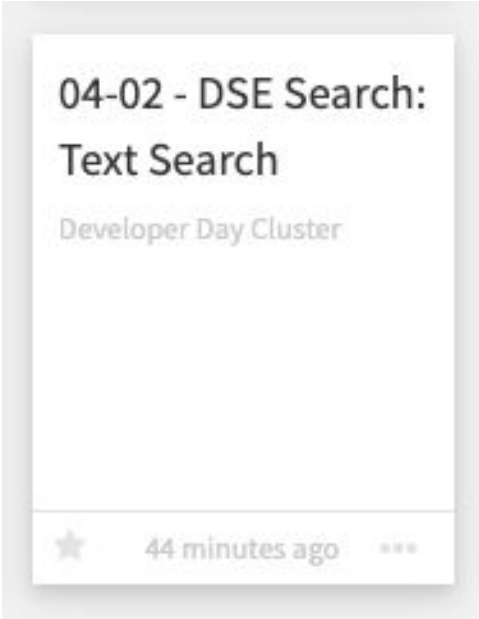


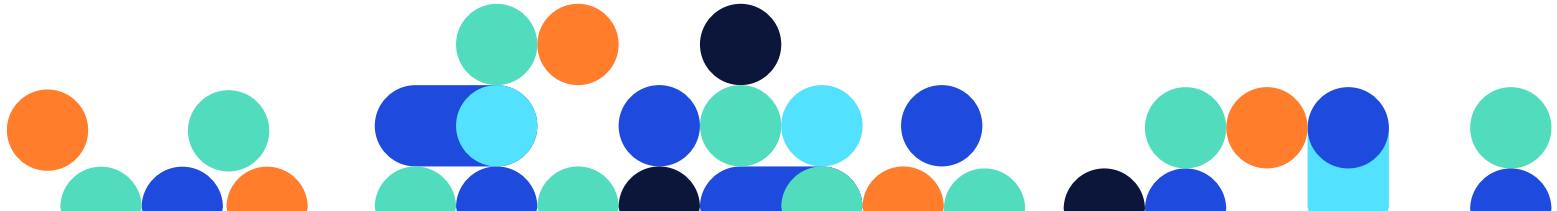
```
Kickboxer 2: The Road Back
The Best of Bray Road
The Black Rider: Revelation Road
```

Time for an exercise!

DATASTAX

Text Search

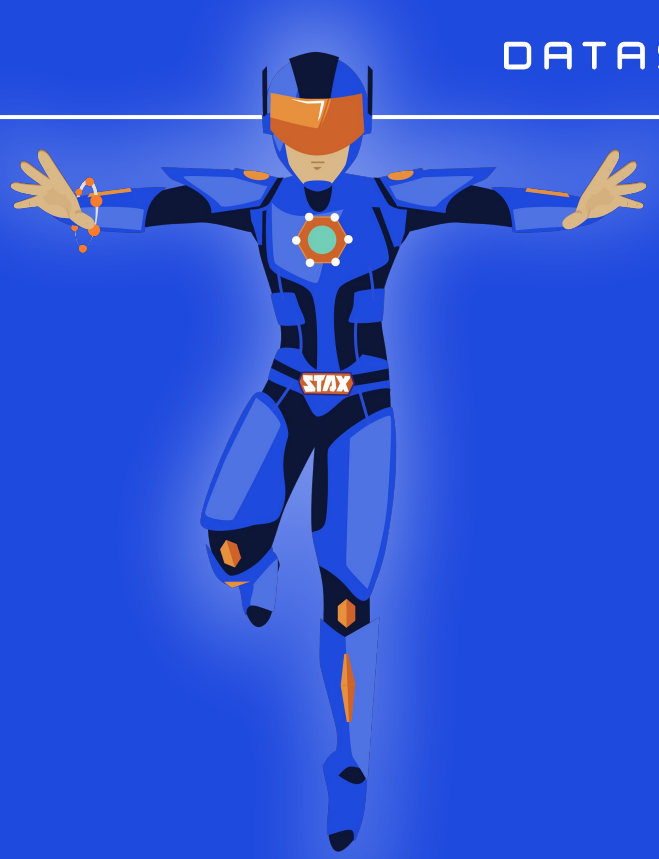




DSE Search

DATASTAX

Wrapping Up



In summary...

- Some use cases can be a challenge to implement with just Cassandra
  - DSE Search makes it much easier
- If you can write CQL, you can search
  - Basic search done using only CQL
  - Use the solr_query column for more complex search and text search
  - Also for managing your schema and config
- Search is great at many things, especially:
  - Text search
  - Counting
  - GeoSpatial

Some resources to learn more

- DataStax Academy
  - https://academy.datastax.com
  - DS310 – DataStax Enterprise Search
- DataStax Community
  - https://community.datastax.com
  - Tags: ”search”, “dse search”
- DataStax Documentation
  - DSE Developer Guide

DATASTAX ACADEMY

DATASTAX®

Thank You