

Atwater Real-Time Customer Recommendations

with DataStax Enterprise Analytics

DataStax Bootcamp Capstone

Amanda Kay Moran

August 24th, 2018



A woman with dark curly hair, wearing a red headband, glasses, and a blue denim shirt, is smiling while working on a laptop. She is wearing white earbuds. The background is a blurred office setting. The image is framed by a large blue circle on the right side of the slide.

GIVING CUSTOMERS A ENGAGING EXPERIENCE

The Data Model

Customer Transactions, Customer Live, Inventory

Customer Transaction/Live Table

- Customer Transaction Table is added to after each completed transaction and will be used to create our model
- Customer Live Table is a table representing logged-in customers and the contents of their shopping cart. These tables have the same setup and primary key.
(Training/Testing tables)



Customer Transaction/Live Table

- PRIMARY KEY ((state), gender, age, id)
- The partition key is on state as customers are well distributed on the 50 states
- Cluster keys on gender, age, and to give uniqueness to the key we will add the transaction id

CQL Keyspace: demo

```
CREATE TABLE IF NOT EXISTS customer_transactions (id int,  
customer_name text,  
gender text, age int,  
state text, home_store int,  
items list<text>, year int,  
month int, rewards_member text,  
PRIMARY KEY ((state), gender, age, id));
```

Success - No Data Returned

Success. 0 elements returned. Duration: 0.011 s.

Inventory Table

- The inventory table is created to represent over 6 million sku's in stock at any one time
- Our primary key is going to be around the item type (pants, shirts, dress), the location of the items, if it the items is currently available, and to give uniqueness the sku.
- Atwater has over 50 warehouses in each 50 states. All item types are in all warehouses supplying each store but not each unique item is.

Inventory Table

- PRIMARY KEY (item_type, stock_loc, backorder, sku)

CQL Keyspace: demo

```
CREATE TABLE IF NOT EXISTS inventory (sku int,  
    item_name text, item_type text,  
    stock_loc text, num_items int,  
    backorder text,  
    PRIMARY KEY (item_type, stock_loc, backorder, sku));
```

Success - No Data Returned

Success. 0 elements returned. Duration: 0.017 s.

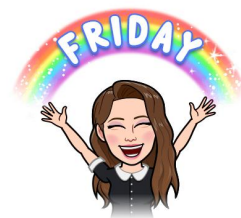


CQL Keyspace: demo

```
INSERT INTO inventory (sku, item_name, item_type, stock_loc, num_items, backorder)  
VALUES (565, 'Boot', 'shoes', 'MI', 141, 'N');
```

Success - No Data Returned

Success. 0 elements returned. Duration: 0.005 s.



High Level Workflow

The Workflow -- and LIVE DEMO

- **Create** `customer_transaction`, `customer_live`, `customer_recommend`, and `inventory` **tables** in DSE
- **Load data** via `cqlsh` to `customer_transaction`, `customer_live`, and `inventory`
 - 1 million records `customer_transaction`
 - 1 thousand records `customer_live`
 - 750 unique sku's records in `inventory`
- Take `customer_transactional` data and live data and transform in Spark Dataframe
- Do some basic visualizations with data (percentage of customers Millennial vs not)
 - Utilizing Spark dataframe to do the WHERE
- Build Two FPGrowth (Frequent Pattern) Models, one for Millennials and one for everyone else
- Test those models on live/shopping cart data
- Load the customer data with a predicted recommendation back into DSE
- For each customer with a recommendation query the `inventory` table to recommend actual products (not just the type) that are in-stock and that are located in their state (so the correct season/style)



DEMO

Thank you

