Real-time Interactive Big Data Analysis Using In-Memory Computing

Mike Joyce – Manager Software Engineer, iCrossing
Shawn Nguyen – Lead Software Engineer, iCrossing
**CONNECTED MARKETING PLATFORM (TECHNOLOGY)**

- Bid Management / Trading Desk
- Data Management Platform (Core Audience)

**STRATEGY & PLANNING**
- Market Research
- Analytics
- Strategy & Planning

**PROGRAM DESIGN**
- Media Planning & Buying
- Creative & Experience Design
- Content Creation & Management

**AUDIENCE ENGAGEMENT**
- Search Marketing Programs
- Social Media / Mobile
- Technology & App Development
- Measurement & Optimization
DIGITAL AGENCY INSIDE A CONTENT EMPIRE

Leveraging audience insights:

- 20+ brands
- 30+ TV networks
- 50+ newspapers
- 300+ magazines
Big Data - Cookies!

- Subscribers
- Visitors
- International
- Multiple devices

300+ million unique cookies
DMP Audience Data

Attributes

- Geographic
- Demographic
- Behavioral
- Psychographic

11,000+ Unique Attributes
Cookies + Audience Attributes = Super Big Data!

90M+ Cookies

Male
Age 20 - 35
Sports Enthusiasts
Iowa
High Income
iPad, iPhone
Drives Mini Van
Foodie

Average user
800+ attributes

72B+
Attribute User pairs
Audiences – Targeting vs Discovering

• **Who** you are targeting

• **How** do you connect with them?

• **What** describes them?
Data Scientists

Discovering Audience Attributes

1. Define an audience using attributes
2. Identify all attributes of cookies in audience
3. Calculate highly indexing attributes

Created by Thibault Geffroy from the Noun Project
1) Define the Audience

Population
90M Cookies

Audience
300K Cookies

- Gender: Male
- Age: 20-35
- US > North Dakota
2) Audience Attributes

Attributes of Audience
Cookies in Audience

- Interest: Sports Enthusiast
- Interest: Moose Hunting
- Intent: Auto Purchase >
- US > North Dakota > Fargo
- Pet Supplies > Dog Food
3) Index the Attributes

### Attributes of Cookies in Audience

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Audience</th>
<th>Population</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interest: Sports Enthusiast</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Interest: Moose Hunting</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Intent: Auto Purchase &gt; Truck</td>
<td></td>
<td></td>
</tr>
<tr>
<td>US &gt; North Dakota &gt; Fargo</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pet Supplies &gt; Dog Food</td>
<td>6%</td>
<td>9%</td>
</tr>
</tbody>
</table>
Data Scientists

Development Ask

1. Make it accessible to “normals”
2. Exportable visualizations & calculations
3. Reduce query time from 1 hr to 1 sec
Why is this Hard?

90M+ Cookies

Average user
800+ attributes

Algorithm
1. Check every cookie if it satisfies audience criteria
2. Collect all attributes for every audience cookie
3. Calculate percentages & index

Within 1 sec !!!!!!
The Answer – Audience Discovery Tool

- Audience discovery
  - Cookie Attributes
  - Frequency vs Population
- Built for non-technical users
  - Strategy
  - Sales / Account
  - Anyone
- Flexible
  - Research tool
  - In-meeting, iterative discovery
- Approachable
  - Real-time
  - Results in seconds
  - Simple, elegant interface
  - Multiple export formats

“Making science accessible”
Data Processing R& D
Traditional Relational Databases

- Long load time
- Complex queries resulting in long query times
- Rigid data model
Non Traditional Databases

- Lack of complex query feature
- Large memory footprint requirement
- Aggregation query exceeded by many 10x of seconds
The Low Hanging Fruit

- In memory cache
- Customizable query using Java code
- Relatively low data loading time
The Vertical Problem
Distributed Computing Ecosystem

- Not production ready
- Data import fails without explanation
- Aggregation fails to impress

[Logos for Cloudera, Impala, Shark, and Spark]
Back to Basics

• Pure Java code solution
• Data and logic must exists in same memory space
• Capable of advanced filtering
• Distributed computing, low overhead
• Data locality
• Minimal code migration
The Distributed Solution
The Challenges

• Tedious manual data distribution
• Gar building and deployment issues
• Development challenges
What We Learned

• Indexed data requiring minor calculations -- databases (relational & noSQL) great

• Large non-indexed data -- the data & processing need to live in the same (memory) space