Development of IBM Watson with UIMA DUCC

Eddie Epstein
eae@apache.org

Apache UIMA
PMC Member and Committer

ApacheCon NA 2015
Presentation Outline

- What is DUCC
- Overview of the IBM-Jeopardy! Question-Answering System
- Interesting development problems
  - Solutions embodied in DUCC
- Fast cruise through DUCC's web interface
What is DUCC

• A Linux-based cluster controller designed specifically for UIMA
• Scales out any UIMA pipeline:
  • for high throughput, or
  • for low latency
• Uses CGroups to partition user processes
• Flexible Resource Management
• Extensive Web, CLI and API interfaces
What DUCC Does

• Collection Processing Jobs
  • Scale out a UIMA pipeline into multiple threads and processes, distribute collection as work items

• Shared Services
  • Manage life cycle of services, supporting dependencies with Jobs or other Services

• Arbitrary Processes
  • Launch arbitrary singleton processes or just provide a container to work
Motivations for DUCC

- Support Ongoing Watson Development
  - Take advantage of game playing hardware
  - Expanding development team
- Bring Functionality to Apache UIMA Community
  - Separate implementation from Watson code
  - Improve quality by targeting wide audience
Example Jeopardy Question

IN 1698, THIS COMET DISCOVERER TOOK A SHIP CALLED THE PARAMOUR PINK ON THE FIRST PURELY SCIENTIFIC SEA VOYAGE

Keywords: 1698, comet, paramour, pink, ...

1. Edmond Halley (0.85)
2. Christiaan Huygens (0.20)
3. Peter Sellers (0.05)
Open Source Software Critical for Watson

Runtime
- Apache UIMA
- Indri Text Search (www.lemurproject.org/indri/)
- Apache Lucene (Text Search)
- Sesame (http://aduna-software.com/technology/sesame)
- Apache ActiveMQ (used by UIMA-AS)

During Development
- Eclipse (https://eclipse.org)
- Weka (http://sourceforge.net/projects/weka/)
- Apache Hadoop
Watson’s Knowledge for Jeopardy!

Watson has analyzed and stored the equivalent of about 1 million books (e.g., encyclopedias, dictionaries, news articles, reference texts, plays, etc).

Watson also uses structured sources such as WordNet and DBpedia.
Watson on UIMA

Aggregate Analysis Engine

- Analysis Engine: Question Analysis
- Analysis Engine: Primary Searches
- Analysis Engine: Candidate Generation
- Analysis Engine: Answer Scoring
- Analysis Engine: Supporting Evidence Search
- Analysis Engine: Deep Evidence Scoring
- Analysis Engine: Final Merger

Flow Controller

CAS
Watson on UIMA – Data Flow

Aggregate Analysis Engine

- Analysis Engine
  - Question Analysis
  - Primary Searches
  - Candidate Generation
  - Answer Scoring
  - Supporting Evidence Search
  - Deep Evidence Scoring
  - Final Merger

Flow Controller

CAS
Problem – One Experiment

- Average 2 hours per question
  - Wide range of times
- 28GB Java Heap on 32GB Machines
  - Large knowledge bases (e.g. Sesame in-memory store)
- ~1000 questions each
  - To get statistically relevant results
Solution – One Experiment

• Run parallel pipelines in multiple threads
  • Share the large in-memory objects
  • Utilize the 8-cores in each machine

• Replicate processes across machines
  • Dynamically feed idle threads next question
BLADE Tool (before DUCC)

UIMA DUCC - Job Model

Collection of Input Data

Raw Data

Analytic Pipeline

Data Ref's

Work Item Generator

Inspect Data

Analysis Results
Job Model – Core UIMA Job

Job Driver

Collection Reader

Job Processes

CM AE CC
CM AE CC
CM AE CC

HTTP

Application Code

Ducc Code
Job Model – UIMA-AS Job

Job Driver

Collection Reader

HTTP

Job Processes

UIMA-AS Service

Application Code

Ducc Code
Job Model – Custom Job

Job Driver

Collection Reader

Job Processes

Java App (Non-UIMA)

HTTP

Application Code

Ducc Code
Job Debugging – all_in_one

All Job code deployed in a single thread in a single process for development & debug
Problem – 15 Researchers

- Personnel evaluated by their contribution to overall accuracy
  - With exceptions, e.g. reduce “stupid answers”
- Wanted their resource “fair share” NOW
Solution – 15 Researchers

- **Preempt running processes**
  - Kill processes with least CPU investment
  - < 10% overhead for lost investment
- **Ramp up after successful initialization**
  - Saved more than preemption loses
- **Allow processes to be non-preemptable**
  - Reserve entire machines
  - Singleton processes (in CGroup containers)
  - Jobs
No, from the start some UIMA components were shared UIMA-AS services
Performance Bottleneck (Development Mode)

- 50 GB Search Index
- NFS Filesystem
- JVM
  - ~30 GB
- File system Buffers
  - JVM
  - ~30 GB
- 32GB Machines
Services Improve Performance

NFS Filesystem

Shared UIMA-AS Service

File system Buffers

Indri Search

File system Buffers

JVM with JNI ~30 GB

48GB Machines

32GB Machines
Problem – Managing Services

- Startup and number of instances manual
- Team had ~3 week sprints
  - Integrate changes and create new baseline
  - New indexes or code meant new services
  - Several baselines active concurrently
DUCC Services

- Service registry
- UIMA-AS or CUSTOM
  - Service “pinger” class required
  - Built-in pinger for UIMA-AS
- Always-on or start-on-demand
- Pinger interface supports autonomous instance management
DUCC Service

Service Manager

Instantiate & query

Service Pinger

Instantiate

Service Process Code

Monitor

Application Code

DUCC Code
DUCC – Node Visualization
DUCC Web Demo
Backup if no Demo
### DUCC Job Page

#### Jobs List

<table>
<thead>
<tr>
<th>Id</th>
<th>Start</th>
<th>Duration</th>
<th>User</th>
<th>Class</th>
<th>State</th>
<th>Reason or extraordinary status</th>
<th>Processes</th>
<th>Init Fails</th>
<th>Run Fails</th>
<th>Pgin</th>
<th>Swap</th>
<th>Size</th>
<th>Total</th>
<th>Done</th>
<th>Error</th>
<th>Dispatch</th>
<th>Retry</th>
<th>Preempt</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>176</td>
<td>2015.04.06 17:23:57 Mon</td>
<td>20:13</td>
<td>jellyfish</td>
<td>normal</td>
<td>Running</td>
<td></td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>30</td>
<td>1001</td>
<td>577</td>
<td>0</td>
<td>16</td>
<td>0</td>
<td>type #2 item count &quot;medium&quot;</td>
</tr>
<tr>
<td></td>
<td>2015.04.06 17:23:06 Mon</td>
<td>21:04</td>
<td>albatross</td>
<td>normal</td>
<td>Running</td>
<td></td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>30</td>
<td>1001</td>
<td>451</td>
<td>0</td>
<td>15</td>
<td>0</td>
<td>type #2 item count &quot;medium&quot;</td>
</tr>
<tr>
<td>179</td>
<td>2015.04.06 17:35:15 Mon</td>
<td>07:16</td>
<td>hummingbird</td>
<td>low</td>
<td>Completed</td>
<td>EndOfJob</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>30</td>
<td>101</td>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>type #1 item count &quot;small&quot;</td>
</tr>
<tr>
<td>177</td>
<td>2015.04.06 17:24:41 Mon</td>
<td>07:19</td>
<td>tapir</td>
<td>normal</td>
<td>Completed</td>
<td>EndOfJob</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>15</td>
<td>101</td>
<td>101</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>type #1 item count &quot;small&quot;</td>
</tr>
</tbody>
</table>
### Job Details - Processes

#### Processes List

<table>
<thead>
<tr>
<th>Id</th>
<th>Log</th>
<th>Size</th>
<th>Host Name</th>
<th>PID</th>
<th>State Scheduler</th>
<th>Reason Scheduler or extraordinary status</th>
<th>State Agent</th>
<th>Exit</th>
<th>Time Init</th>
<th>Time Run</th>
<th>Time GC</th>
<th>PgIn</th>
<th>Swap</th>
<th>%CPU</th>
<th>RSS</th>
<th>Time Avg</th>
<th>Time Max</th>
<th>Time Min</th>
<th>Done</th>
<th>Error</th>
<th>Dispatch</th>
<th>Retry</th>
<th>Preempt</th>
<th>JConsole URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>176-JM-uima-ducc-demo-3-12267.log</td>
<td>5.78</td>
<td>uima-ducc-demo-3</td>
<td>12267</td>
<td>Allocated</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td>00 22:13</td>
<td>00 0 0 0 0.0 0.1</td>
<td>25 60</td>
<td>10 543</td>
<td>0</td>
<td>16</td>
<td>0</td>
<td>service.jmxmli://jndi/rmi://uima-ducc-wm.10014/jmxmli</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>176-JM-uima-ducc-demo-10-13880.log</td>
<td>0.42</td>
<td>uima-ducc-demo-10</td>
<td>13880</td>
<td>Allocated</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td>25 20.48</td>
<td>01 0 0 0 0.0 0.1</td>
<td>35 60</td>
<td>10 272</td>
<td>0</td>
<td>8</td>
<td>0</td>
<td>service.jmxmli://jndi/rmi://uima-ducc-wm.11024/jmxmli</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>176-JM-uima-ducc-demo-9-17754.log</td>
<td>0.39</td>
<td>uima-ducc-demo-9</td>
<td>17754</td>
<td>Allocated</td>
<td>Running</td>
<td></td>
<td></td>
<td></td>
<td>20 18:53</td>
<td>01 0 0 0 0.0 0.1</td>
<td>34 60</td>
<td>10 252</td>
<td>0</td>
<td>8</td>
<td>0</td>
<td>service.jmxmli://jndi/rmi://uima-ducc-wm.11027/jmxmli</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>176-JM-uima-ducc-demo-10-20757.log</td>
<td>0.20</td>
<td>uima-ducc-demo-10</td>
<td>20757</td>
<td>Dealocated Forced</td>
<td>Stopped Deallocated ExitCode=1</td>
<td></td>
<td></td>
<td></td>
<td>08 17:26</td>
<td>00 0 0 0 1.0 0.1</td>
<td>37 59</td>
<td>10 119</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Job Details**

- **Apache UIMA-DUCC**
  - Updated: 2015-04-06 17:46:12
  - Mon
  - Utilization: 80.0%
# Job Details - Performance

**Apache UIMA-DUCC**  
Updated: 2015.04.06 17:46:12 Mon  
Utilization: 80.0%

<table>
<thead>
<tr>
<th>Name</th>
<th>Total</th>
<th>% of Total</th>
<th>Avg</th>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Summary</strong></td>
<td>06:17:43.5</td>
<td>100.0</td>
<td>35.2</td>
<td>05.1</td>
<td>01:00.2</td>
</tr>
<tr>
<td>DefaultAE</td>
<td>06:17:41.8</td>
<td>100.0</td>
<td>35.2</td>
<td>10.0</td>
<td>59.9</td>
</tr>
<tr>
<td>DUCC Job Flow Controller</td>
<td>01.7</td>
<td>0.0</td>
<td>00.0</td>
<td>00.0</td>
<td>00.2</td>
</tr>
</tbody>
</table>
### Services Definitions List

<table>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>State</th>
<th>Last Use</th>
<th>Instances</th>
<th>Deployments</th>
<th>Start State</th>
<th>User</th>
<th>Class</th>
<th>Pgin</th>
<th>Swap</th>
<th>Size</th>
<th>Jobs</th>
<th>Services</th>
<th>Reservations</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CUSTOM:localhost:7175</td>
<td>Available</td>
<td></td>
<td>1</td>
<td>1</td>
<td>Autostart degenero</td>
<td>0</td>
<td>0.0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>Custom Service Type auto-start</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>UIMA-AS:FixedSleepAE_4.tcp://localhost:61617</td>
<td>Available</td>
<td></td>
<td>1</td>
<td>1</td>
<td>Reference zebra</td>
<td>fixed</td>
<td>1</td>
<td>0.0</td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>Test Service 4 Type on-demand</td>
</tr>
<tr>
<td>9</td>
<td>UIMA-AS:FixedSleepAE_3.tcp://localhost:61617</td>
<td>Stopped</td>
<td></td>
<td>1</td>
<td>0</td>
<td>Stopped moose</td>
<td>fixed</td>
<td>0</td>
<td>0.0</td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>Test Service 3 Type on-demand</td>
</tr>
<tr>
<td>8</td>
<td>UIMA-AS:FixedSleepAE_2.tcp://localhost:61617</td>
<td>Stopped</td>
<td></td>
<td>1</td>
<td>0</td>
<td>Stopped squirrel</td>
<td>fixed</td>
<td>0</td>
<td>0.0</td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>Test Service 2 Type on-demand</td>
</tr>
<tr>
<td>7</td>
<td>UIMA-AS:FixedSleepAE_1.tcp://localhost:61617</td>
<td>Stopped</td>
<td></td>
<td>1</td>
<td>0</td>
<td>Stopped kiwi</td>
<td>fixed</td>
<td>0</td>
<td>0.0</td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>Test Service 1 Type auto-start</td>
</tr>
<tr>
<td>6</td>
<td>UIMA-AS:FixedSleepAE_0.tcp://localhost:61617</td>
<td>Stopped</td>
<td></td>
<td>1</td>
<td>0</td>
<td>Stopped chinchilla</td>
<td>fixed</td>
<td>0</td>
<td>0.0</td>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td>Test Service 0 Type on-demand</td>
</tr>
</tbody>
</table>
## DUCC Reservation Page

<table>
<thead>
<tr>
<th>Id</th>
<th>Start</th>
<th>Duration</th>
<th>User</th>
<th>Class</th>
<th>Type</th>
<th>State</th>
<th>Reason</th>
<th>Allocation</th>
<th>User Processes</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>174</td>
<td>2015.04.06</td>
<td>17:20:30</td>
<td>Mon</td>
<td></td>
<td></td>
<td>Running</td>
<td></td>
<td></td>
<td>1</td>
<td>15</td>
<td>uima-ducc-demo-1 #2 type &quot;managed&quot;, end by user cancel</td>
</tr>
<tr>
<td>172</td>
<td>2015.04.06</td>
<td>17:17:55</td>
<td>Mon</td>
<td></td>
<td></td>
<td>Assigned</td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
<td>uima-ducc-demo-3 Job Driver</td>
</tr>
<tr>
<td>169</td>
<td>2015.04.06</td>
<td>16:41:57</td>
<td>Mon</td>
<td></td>
<td></td>
<td>CompletedExt</td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
<td>uima-ducc-demo-4 #3 type &quot;managed&quot;, end by program exit</td>
</tr>
<tr>
<td>165</td>
<td>2015.04.06</td>
<td>15:10:09</td>
<td>Mon</td>
<td></td>
<td></td>
<td>CompletedExt</td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
<td>uima-ducc-demo-4 #3 type &quot;managed&quot;, end by program exit</td>
</tr>
<tr>
<td>156</td>
<td>2015.04.06</td>
<td>14:23:06</td>
<td>Mon</td>
<td></td>
<td></td>
<td>CompletedExt</td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
<td>uima-ducc-demo-4 #3 type &quot;managed&quot;, end by program exit</td>
</tr>
<tr>
<td>152</td>
<td>2015.04.06</td>
<td>13:39:33</td>
<td>Mon</td>
<td></td>
<td></td>
<td>CompletedExt</td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
<td>uima-ducc-demo-8 #3 type &quot;managed&quot;, end by program exit</td>
</tr>
<tr>
<td>144</td>
<td>2015.04.06</td>
<td>12:38:59</td>
<td>Mon</td>
<td></td>
<td></td>
<td>CompletedExt</td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
<td>uima-ducc-demo-1 #2 type &quot;managed&quot;, end by user cancel</td>
</tr>
<tr>
<td>132</td>
<td>2015.04.06</td>
<td>11:59:44</td>
<td>Mon</td>
<td></td>
<td></td>
<td>CompletedExt</td>
<td></td>
<td></td>
<td>1</td>
<td>0</td>
<td>uima-ducc-demo-1 #3 type &quot;managed&quot;, end by program exit</td>
</tr>
</tbody>
</table>
Thank You