Leveraging smart meter data for electric utilities: Comparison of Spark SQL with Hive

5/16/2017
Hitachi, Ltd. OSS Solution Center
Yusuke Furuyama
Shogo Kinoshita
Who are we?

◆ Yusuke Furuyama
- Solutions engineer at Hitachi, Ltd.
- Offering and co-creating progressive Hadoop solutions to customers who are going to build enterprise system.

◆ Shogo Kinoshita
- Solutions Engineer at Hitachi, Ltd.
- Focusing on Hadoop eco-system (including Spark, Hive, Impala) and write web-articles, make presentations about evaluation of Hadoop-related OSS.
Contents

1. Leveraging smart meter data [Sample use case for electric utilities]
2. Performance evaluation of MapReduce and Spark 1.6 (using Hive and Spark SQL)
3. Additional evaluation with Spark 2.1
4. Summary
1. Leveraging smart meter data
   [Sample use case for electric utilities]
## 1-1 Hitachi Corporate Profile

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Established</td>
<td>February 1, 1920</td>
</tr>
<tr>
<td>Capital</td>
<td>458.7 billion yen</td>
</tr>
<tr>
<td></td>
<td>(as of end of Mar. 2017)</td>
</tr>
<tr>
<td>Number of Employees</td>
<td>303,887</td>
</tr>
<tr>
<td></td>
<td>(as of end of Mar. 2017)</td>
</tr>
<tr>
<td>Revenues</td>
<td>9,162.2 billion yen</td>
</tr>
<tr>
<td></td>
<td>(FY2016 Consolidated)</td>
</tr>
<tr>
<td>Operating Income</td>
<td>587.3 billion yen</td>
</tr>
<tr>
<td></td>
<td>(FY2016 Consolidated)</td>
</tr>
</tbody>
</table>

Hitachi, Ltd.  
President & CEO  
Toshiaki Higashihara
Revenue by Segments FY2016

Total: 9,162.2 billion yen

- **Information & Telecommunication Systems**: 23%
- **Electronics Systems & Equipment**: 14%
- **Social Infrastructure & Industrial Systems**: 12%
- **Construction Machinery**: 7%
- **High Functional Materials & Components**: 6%
- **Automotive Systems**: 6%
- **Smart Life & Ecofriendly Systems**: 6%
- **Other (Logistics & Other services)**: 2%
- **Financial Services**: 2%

*Figures are on a consolidated basis*
1-3 Hitachi’s Social innovation business approach

Solutions to social issues and social innovation by collaborative creation and open innovation

OT×IT Integrated Services/Cross-industrial Business/Business Ecosystem

- O&M Service
  - Energy Management
- Datahealth
- Fintech
- Total SCM
  - Railway/machinery × IT
  - Power × IT
  - Healthcare × IT
  - Manufacturing × IT
  - Finance × IT

© Hitachi, Ltd. 2017. All rights reserved.
1-4 Situation of electric utilities in Japan and their needs

- Liberalization of the retail Electric Power Market
  - Electric utilities in Japan have to adapt to competitive free market
  - Request for price cut of power transmission fee from government of Japan

- Needs to cut the cost for transmission and distribution equipment
  - Transmission and distribution equipment have been replaced periodically
  - Decide the timing of replacement by the condition of equipment

Maintenance team needs: Obtain the load status of each equipment

- Electric Power generation companies
- Transmission and distribution companies
- Substation
- Distribution lines
- Transmission lines
- Power transmission fee
- Electricity rate
- Electricity retailers
- Home
1–5 Situation of electric utilities in Japan and their needs (future)

- Unstable power supply
  - Decreasing nuclear plant as a stable power supplier
  - Increasing renewable energy supply

- Needs for high level Demand Response
  - Rates by time zone (current demand response)
  - Many and small renewable energy suppliers
  - Near real-time demand response for each distribution system

Planning team needs: Obtain near real-time load status of each equipment

Electric Power Generation Companies

Transmission and Distribution Companies

Transmission Lines

Substation

Distribution Lines

Power transmission fee

Electricity rate

Electricity retailers

Home
1–6 Leveraging big data for electric utilities

Meet the needs of electric utilities

- Power Grid
- Substation (1,000)
- Distribution Lines (5/Substation)
- Switch (20/Distribution Line)
- Transformer (500/Distribution Line)
- Smart Meter (4/Trans)
- Data Analysis System
- Meter Data Management System

Data from smart meters (every 30min.)

Analyze the data from smart meters to grasp the load status of equipment

Concern about performance needed for near real-time processing

Total 10,000,000
Data Analysis System

1-7 System Component

Planner (Equipment/Demand response)

Data processing platform (Hadoop, Spark)

Data visualization tool (Pentaho)

Target of this session

Data from smart meters (every 30min.)

Raw Data → Processed Data

Meter Data Management System

Data Analysis System

Hitachi Demonstration for electric power company

Dashboard → Planning → Setting

System Component

1-7

- Raw Data
- Processed Data

- Data from smart meters (every 30min.)

- Planner (Equipment/Demand response)

- Data visualization tool (Pentaho)

- Data processing platform (Hadoop, Spark)
2. Performance evaluation of MapReduce and Spark 1.6 (using Hive and Spark SQL)
2-1 Use Case for electric utilities

- **Needs of electric utilities (recap)**
  - Analyze the data from smart meters to grasp the load status of equipment
  - Near real-time

- **Points of analysis**

<table>
<thead>
<tr>
<th>Needs</th>
<th>Point of analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Find the equipment that needs to be replaced</td>
<td>Find the equipment that has heavy workload</td>
</tr>
<tr>
<td>Estimate the timing for replacement</td>
<td>Check the trend of load status</td>
</tr>
<tr>
<td>Select the proper capacity of new equipment</td>
<td>Extract the peak of the load</td>
</tr>
</tbody>
</table>
2-2 Contents of performance evaluation

◆ Point of evaluation for near real-time processing

- Concern about performance for processing 10,000,000 meters
- Data comes from each smart meter every 30min (48/Day, spec of smart meter)

Check if MapReduce and Spark can process the data in 30min.

◆ Items for evaluation

<table>
<thead>
<tr>
<th>Point of analysis</th>
<th>Aggregate per</th>
<th>Items for evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Find the equipment has heavy workload</td>
<td>Equipment</td>
<td>• Distribution system • Substation • Switch</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Transformer • Meter</td>
</tr>
<tr>
<td>Check the trend of load status</td>
<td>Term</td>
<td>• 1day • 1month(30days) • 1year(365days)</td>
</tr>
<tr>
<td>Extract the peak of the load</td>
<td>Time zone</td>
<td>• Specific 30min of each day • 24h</td>
</tr>
</tbody>
</table>
2-3 Target of performance evaluation

Target of evaluation

Time from start of aggregation batch for meter data to end of the batch

Data visualization tool (Pentaho)

Start

Aggregation batch (Hive / Spark SQL)

Data Processing Platform (MapReduce, Spark 1.6)

End

Target

Data Analysis System

Meter Data Management System
2-4 Evaluation environment

System Configuration

1 Master Node (Virtual Machine)
- CPU Core: 2
- Memory: 8 GB
- Capacity of disk: 80 GB
- # of disk: 1

4 Slave Nodes (Physical Machines)
- CPU Core: 16
- Memory: 128 GB
- Capacity of disk: 900 GB
- # of disk: 6

Spec

<table>
<thead>
<tr>
<th></th>
<th>Master Node</th>
<th>Per slave node</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Core</td>
<td>2</td>
<td>16</td>
<td>64</td>
</tr>
<tr>
<td>Memory</td>
<td>8 GB</td>
<td>128 GB</td>
<td>512 GB</td>
</tr>
<tr>
<td>Capacity of disk</td>
<td>80 GB</td>
<td>900 GB</td>
<td>-</td>
</tr>
<tr>
<td># of disk</td>
<td>1</td>
<td>6</td>
<td>24</td>
</tr>
<tr>
<td>Total capacity of disks</td>
<td>5.4 TB (5,400 GB)</td>
<td>21.6 TB (21,600 GB)</td>
<td></td>
</tr>
</tbody>
</table>
2-5 Dataset

Smart meter data

<table>
<thead>
<tr>
<th>Smart meter data /day</th>
<th>48 columns (every 30min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>meter1</td>
<td>0:00-0:30 power usage</td>
</tr>
<tr>
<td></td>
<td>0:30-1:00 power usage</td>
</tr>
<tr>
<td>meter2</td>
<td>0:00-0:30 power usage</td>
</tr>
<tr>
<td></td>
<td>0:30-1:00 power usage</td>
</tr>
<tr>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>meter 10,000,000</td>
<td>0:00-0:30 power usage</td>
</tr>
<tr>
<td></td>
<td>0:30-1:00 power usage</td>
</tr>
<tr>
<td></td>
<td>23:30-0:00 power usage</td>
</tr>
<tr>
<td></td>
<td>Meter mgmt. info</td>
</tr>
<tr>
<td></td>
<td>23:30-0:00 power usage</td>
</tr>
<tr>
<td></td>
<td>Meter mgmt. info</td>
</tr>
</tbody>
</table>

Data size

<table>
<thead>
<tr>
<th>Term</th>
<th># of records</th>
<th>Size (CSV)</th>
<th>Size (ORCFile)</th>
</tr>
</thead>
<tbody>
<tr>
<td>365days (1 year)</td>
<td>3,650 million</td>
<td>2.475 TB</td>
<td>1.325 TB</td>
</tr>
<tr>
<td>30days (1 month)</td>
<td>300 million</td>
<td>0.205 TB</td>
<td>0.158 TB</td>
</tr>
<tr>
<td>1 day</td>
<td>10 million</td>
<td>0.007 TB</td>
<td>0.005 TB</td>
</tr>
</tbody>
</table>
## 2-6 Contents of performance evaluation (recap)

### Point of evaluation

Check if MapReduce and Spark can process the data in 30min.

### Target of evaluation

Time from start of aggregation batch for meter data to end of the batch.

### Items for evaluation

<table>
<thead>
<tr>
<th>Point of analysis</th>
<th>Aggregate per</th>
<th>Target of evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Find the equipment has heavy workload</td>
<td>Equipment</td>
<td>• Distribution system • Substation • Switch • Transformer • Meter</td>
</tr>
<tr>
<td>Check the trend of load status</td>
<td>Term</td>
<td>• 1day • 1month(30days) • 1year(365days)</td>
</tr>
<tr>
<td>Extract the peak of the load</td>
<td>Time zone</td>
<td>• Specific 30min of each day • 24h</td>
</tr>
</tbody>
</table>
2-6 Contents of performance evaluation (recap)

◆ Point of evaluation

Check if MapReduce and Spark can process the data in 30min.

◆ Target of evaluation

Time from start of aggregation batch for meter data to end of the batch

◆ Items for evaluation

<table>
<thead>
<tr>
<th>Point of analysis</th>
<th>Aggregate per</th>
</tr>
</thead>
<tbody>
<tr>
<td>Find the equipment has heavy workload</td>
<td>Equipment</td>
</tr>
<tr>
<td>Check the trend of load status</td>
<td>Term</td>
</tr>
<tr>
<td>Extract the peak of the load</td>
<td>Time zone</td>
</tr>
</tbody>
</table>

+ File type
- Text (CSV)
- ORCFile (Column-based)

- Distribution system
- Substation
- Switch
- Transformer
- Meter

- 1day
- 1month (30days)
- 1year (365days)

- Specific 30min of each day
- 24h
2-7 Comparison of txt with ORCFile (MapReduce)

- Aggregate meter data of entire Distribution System

- Couldn’t finish processing in 30min
- Performance improvement by ORCFile
2-8 Comparison of txt with ORCFile (Spark 1.6)

- Aggregate meter data of entire Distribution System

- Could finish processing in 30min (1,800s)
- Performance improvement by ORCFile
2-9 Review of the results

◆ Why the processing was fast with ORCFile

Results
- Processing big data with ORCFile was more effective than processing small data
- Processing specific 0.5h data with ORCFile was more effective than processing 24h data

◆ Processing 0.5h data

<table>
<thead>
<tr>
<th>0:00</th>
<th>0:30</th>
<th>23:30</th>
<th>0:00</th>
</tr>
</thead>
<tbody>
<tr>
<td>0:00</td>
<td>0:30</td>
<td>23:30</td>
<td>0:00</td>
</tr>
<tr>
<td>0:00</td>
<td>0:30</td>
<td>23:30</td>
<td>0:00</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>....</td>
<td>...</td>
</tr>
</tbody>
</table>

Use 1 column only

Smart meter data

◆ Processing 24h data

<table>
<thead>
<tr>
<th>0:00</th>
<th>0:30</th>
</tr>
</thead>
<tbody>
<tr>
<td>0:00</td>
<td>0:30</td>
</tr>
<tr>
<td>0:00</td>
<td>0:30</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Use all 48 columns

Smart meter data

Features of ORCFile

Compression

Reading specific column

SUM/day

© Hitachi, Ltd. 2017. All rights reserved.
Comparison of MapReduce with Spark 1.6 (ORCFile)

**Result (30days / 24h)**

- **Per equipment (JOIN)**
  - Hive: 145 sec
  - Spark SQL: 224 sec
- **Distribution system (NO JOIN)**
  - Hive: 50 sec
  - Spark SQL: 718 sec

**Result (30days / 0.5h)**

- **Per equipment (JOIN)**
  - Hive: 104 sec
  - Spark SQL: 69 sec
- **Distribution system (NO JOIN)**
  - Hive: 32 sec
  - Spark SQL: 556 sec

**Result (365days / 24h)**

- **Per equipment (JOIN)**
  - Hive: 1359 sec
  - Spark SQL: 3286 sec
- **Distribution system (NO JOIN)**
  - Hive: 993 sec
  - Spark SQL: 4131 sec

- Could finish processing the data from 10,000,000 meter in 30min using Spark 1.6!
- Spark’s good performance with “per equipment” processing

© Hitachi, Ltd. 2017. All rights reserved.
2-11 Review of the results

◆ Why the processing per equipment was more effective than the processing for entire distribution system when using spark?

■ Smart meter data
  - 0:00 - 0:30
  - 23:30
  - ... 23:30

■ Transformer
  - Trans1
  - Trans2
  - Trans3
  - ... 23:30

■ Switch
  - Switch1
  - Switch2
  - ... 23:30

■ Distribution Line
  - Line1
  - ... 23:30

■ Substation
  - Substa.1
  - ... 23:30

◆ Per equipment

◆ For entire distribution system

- Less disk I/O than MapReduce
- Smaller data (including re-distributing data) than total memory of cluster

© Hitachi, Ltd. 2017. All rights reserved.
3. Additional evaluation with Spark 2.1
3-1 Evaluation environment

◆ System Configuration for additional evaluation

1 Master Node (Virtual Machine)
- 1 Gbps LAN
- 10 Gbps SW
- 10 Gbps LAN

4 Slave Nodes (Physical Machines)

◆ Spec

<table>
<thead>
<tr>
<th>Spec</th>
<th>Master Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Core</td>
<td>2</td>
</tr>
<tr>
<td>Memory</td>
<td>8 GB</td>
</tr>
<tr>
<td>Capacity of disk</td>
<td>80 GB</td>
</tr>
<tr>
<td># of disk</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Spec</th>
<th>Per slave node</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Core</td>
<td>16</td>
<td>64</td>
</tr>
<tr>
<td>Memory</td>
<td>128 GB</td>
<td>512 GB</td>
</tr>
<tr>
<td>Capacity of disk</td>
<td>900 GB</td>
<td>-</td>
</tr>
<tr>
<td># of disk</td>
<td>6</td>
<td>24</td>
</tr>
<tr>
<td>Total capacity of disks</td>
<td>5.4 TB (5,400 GB)</td>
<td>21.6 TB (21,600 GB)</td>
</tr>
</tbody>
</table>

10Gbps LAN
Time from start of aggregation batch for meter data to end of the batch

- Performance improvement 22-27% (including disk I/O)
- More effective with large data
### 3-3 Comparison of Spark 2.1 with Spark 1.6 (Parquet)

**Time from start of aggregation batch for meter data to end of the batch**

#### Result (365days / 0.5h)

<table>
<thead>
<tr>
<th>Equipment (JOIN)</th>
<th>Spark 1.6 + Parquet</th>
<th>Spark 2.1 + Parquet</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 sec</td>
<td>401 sec</td>
<td>227 sec</td>
</tr>
<tr>
<td>43% down</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Result (30days / 0.5h)

<table>
<thead>
<tr>
<th>Equipment (JOIN)</th>
<th>Spark 1.6 + Parquet</th>
<th>Spark 2.1 + Parquet</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 sec</td>
<td>126 sec</td>
<td>82 sec</td>
</tr>
<tr>
<td>35% down</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### Result (1day / 0.5h)

<table>
<thead>
<tr>
<th>Equipment (JOIN)</th>
<th>Spark 1.6 + Parquet</th>
<th>Spark 2.1 + Parquet</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 sec</td>
<td>105 sec</td>
<td>76 sec</td>
</tr>
<tr>
<td>28% down</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Performance improvement 28-43% (including disk I/O)
- More effective with large data
- Better improvement than ORCFile
3–4 Comparison of ORCFile with Parquet (Spark 1.6/2.1)

Time from start of aggregation batch for meter data to end of the batch

**Result (Spark 1.6 / Per equipment (JOIN))**

<table>
<thead>
<tr>
<th>Term</th>
<th>Size (Parquet)</th>
<th>Size (ORCFile)</th>
</tr>
</thead>
<tbody>
<tr>
<td>365days / 24h</td>
<td>1.363 TB</td>
<td>1.328 TB</td>
</tr>
<tr>
<td>365days / 0.5h</td>
<td>0.112 TB</td>
<td>0.109 TB</td>
</tr>
<tr>
<td>1day</td>
<td>3.7 GB</td>
<td>3.6 GB</td>
</tr>
</tbody>
</table>

**Result (Spark 2.1 / Per equipment (JOIN))**

- Basically, performance of Parquet is better than ORCFile.
- Performance of Parquet with small data is worse than ORCFile in some cases.
Demo
Demo: Data aggregation and visualization

Data visualization tool (Pentaho)

1. Show 29 days data
2. Execute aggregation batch (Spark SQL)
3. Show 30 days data
4. Outlier detected!!

Aggregated Data (29 days)

Raw Data (1 day)

Aggregated Data (30 days)
4. Summary
Leveraging data from 10,000,000 smart meters for electric utilities in Japan
- Built data analysis system
- Concern about performance

Evaluate the performance of batch processing
- Spark could process the data from 10,000,000 meters in 30min (4 slave nodes)

Evaluate the performance of Spark 2.1
- Performance improvement 22-27% (compared to 1.6, ORCFile)
- Performance improvement 28-43% (compared to 1.6, Parquet)
Leveraging smart meter data for electric utilities: Comparison of Spark SQL with Hive

5/16/2017
Hitachi, Ltd. OSS Solution Center

Yusuke Furuyama
Shogo Kinoshita
Trademarks

- Hadoop, Spark and Hive are trademarks or registered trademarks of Apache Software Foundation in the United States and other countries.
- Other brand names and product names used in this material are trademarks, registered trademarks or trade names of their respective holders.
HITACHI
Inspire the Next
Appendix. Difficulty with large data to be shuffled

◆ Attempted to aggregate raw (48 columns) meter data per equipment
  - Extremely slow (Spark 2.0) or Job failed (Spark 1.6)
  - Processing: Iteration of JOIN and GROUP BY+SUM
  - Huge data to be shuffled (spilled out from page cache)

Heavy load on a local disk (OS disk) by shuffle

◆ Add HDFS disks as disks for shuffle
  - Performance Improved (365days)
  - Performance degraded (1day/30days)

- Data for Spark (including temporary data) should be smaller than memory.
- Had better to process as a trial to estimate