Scalability: Many KNIME Workflows / Users

• KNIME Server
  – Scheduling
  – Distributed Executors – Scale up/Scale out
    • KNIME Cluster Execution
  – KNIME Server REST API
  – KNIME WebPortal
  – KNIME Model Factory
Performance and Scalability of a KNIME Workflow

Optimize:

- Effectiveness
  - Score
  - Accuracy
  - Model
  - Result

- Efficiency
  - Runtime
  - Resources
  - Costs
  - "Happiness"

Effectiveness

- Collaborate
- Automate
- Manage
- Deploy

Load > Integrate > Transform > Analyze > Visualize

KNIME Analytics Platform

KNIME Integrations

KNIME Extensions

Community Extensions

Partner Extensions

KNIME Server

Open Source

© 2019 KNIME AG. All rights reserved.
KNIME Performance and Scalability Options

- Soft Options
- Hardware Options
- 3rd party Execution Contexts

(KNIME Setup)
(Resources)
(KNIME Integrations....)
Soft Options (KNIME Setup) Highlights

• Memory
  – Heap Space
• Redirecting Workspaces, etc. to SSD
• Set Data Tables in Memory Size

• Data Tables in Memory
• Streaming
• Parallel Chunking
• Workflow Structure

https://www.knime.com/blog/optimizing-knime-workflows-for-performance
Hardware Options (Resources)

- Client
- Server
- Cloud
- Hybrid
- Memory
- SSD
- Cores
- GPUs
- Others
- Windows
- Mac OS X
- Linux

Hosting

Hardware

Operating System
3rd party Execution Contexts  (KNIME integrations....)

- R
- Python
- Jupyter
- Spark
- H20
- Tensorflow
- Keras
- Databases
- Microsoft SQL Server R Services
- Kafka
  Amazon – S3, Connectors, etc.
- Azure – Blobstore, Connectors, etc.

Considerations:
- Environment Instantiation/Startup?
- Data conversion to environment format?
- “Everything in Memory” limits?
- Alternate Environment Controls
  (Memory, GPU, etc.)

- GIT Client
- Google Connectivity
- Call External
- REST

Considerations:
- “Outside” of any sort of KNIME control
### Define your workflow steps

<table>
<thead>
<tr>
<th>Workflow Steps</th>
<th>Initialize</th>
<th>Load</th>
<th>Join</th>
<th>Transform</th>
<th>Feature Prep</th>
<th>Learn</th>
<th>Evaluate</th>
<th>Consolidate</th>
<th>Deploy</th>
<th>Interact</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
</tbody>
</table>

© 2019 KNIME AG. All rights reserved.
# Benchmarking Performance & Scalability in Practice

## Workflow Steps

<table>
<thead>
<tr>
<th>Workflow Steps</th>
<th>Initialize</th>
<th>Load</th>
<th>Join</th>
<th>Transform</th>
<th>Feature Prep</th>
<th>Learn</th>
<th>Evaluate</th>
<th>Consolidate</th>
<th>Deploy</th>
<th>Interact</th>
</tr>
</thead>
</table>

## Identify Relevant capabilities

1. Define your workflow steps

## KNIME Analytics Platform

- KNIME Native Nodes
- KNIME Streaming
- DBMS
- R
- Python
- H2O
- Spark
- Keras
- Tensorflow
- Google API
- Microsoft SQL
- R
- Call External

## KNIME Environment Contexts

- WebPortal
- REST API
Benchmarking Performance & Scalability in Practice

1. Define your workflow steps

2. Identify Relevant capabilities

3. Define possible Scenarios

<table>
<thead>
<tr>
<th>Environment Contexts</th>
<th>KNIME Analytics Platform</th>
<th>Workflow Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBMS</td>
<td>KNIME Native Nodes</td>
<td>Initialize</td>
</tr>
<tr>
<td>R</td>
<td>KNIME Streaming</td>
<td>Load</td>
</tr>
<tr>
<td>Python</td>
<td></td>
<td>Join</td>
</tr>
<tr>
<td>H2O</td>
<td></td>
<td>Transform</td>
</tr>
<tr>
<td>Spark</td>
<td></td>
<td>Feature Prep</td>
</tr>
<tr>
<td>Keras</td>
<td></td>
<td>Learn</td>
</tr>
<tr>
<td>Tensorflow</td>
<td></td>
<td>Evaluate</td>
</tr>
<tr>
<td>Google API</td>
<td></td>
<td>Consolidate</td>
</tr>
<tr>
<td>Microsoft SQL R</td>
<td></td>
<td>Deploy</td>
</tr>
<tr>
<td>Call External</td>
<td></td>
<td>Interact</td>
</tr>
<tr>
<td>WebPortal</td>
<td>Server</td>
<td></td>
</tr>
<tr>
<td>REST API</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Benchmarking Performance & Scalability in Practice

### Workflow Steps

<table>
<thead>
<tr>
<th>Initialize</th>
<th>Load</th>
<th>Join</th>
<th>Transform</th>
<th>Feature Prep</th>
<th>Learn</th>
<th>Evaluate</th>
<th>Consolidate</th>
<th>Deploy</th>
<th>Interact</th>
</tr>
</thead>
</table>

1. **Define your workflow steps**
2. **Identify Relevant capabilities**
3. **Define possible Scenarios**
4. **Deal with Environment contexts**

### KNIME Analytics Platform

<table>
<thead>
<tr>
<th>KNIME Native Nodes</th>
<th>KNIME Streaming</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBMS</td>
<td></td>
</tr>
<tr>
<td>R</td>
<td></td>
</tr>
<tr>
<td>Python</td>
<td></td>
</tr>
<tr>
<td>H2O</td>
<td></td>
</tr>
<tr>
<td>Spark</td>
<td></td>
</tr>
<tr>
<td>Keras</td>
<td></td>
</tr>
<tr>
<td>Tensorflow</td>
<td></td>
</tr>
<tr>
<td>Google API</td>
<td></td>
</tr>
<tr>
<td>Microsoft SQL R</td>
<td></td>
</tr>
<tr>
<td>Call External</td>
<td></td>
</tr>
</tbody>
</table>

### KNIME Server

| WebPortal          | REST API         |

---

© 2019 KNIME AG. All rights reserved.
Benchmarking Performance & Scalability in Practice

1. Define your workflow steps
2. Identify relevant capabilities
3. Define possible scenarios
4. Deal with environment contexts
5. Set environment and execute

**Workflow Steps:**
- Initialize
- Load
- Join
- Transform
- Feature Prep
- Learn
- Evaluate
- Consolidate
- Deploy
- Interact

**KNIME Analytics Platform**
- KNIME Native Nodes
- KNIME Streaming
- DBMS
- R
- Python
- H2O
- Spark
- Keras
- Tensorflow
- Google API
- Microsoft SQL
- R
- Call External

**Environment:**
- Memory:
- Cores:
- Disktype:

**Execution Type:**
- Local / Cloud

**Operating System:**
- Windows
Benchmarking Performance & Scalability in Practice

1. Define your workflow steps
2. Identify Relevant capabilities
3. Define possible Scenarios
4. Deal with Environment contexts
5. Set environment and execute
6. Measure and Compare

**Workflow Steps**
- Initialize
- Load
- Join
- Transform
- Feature Prep
- Learn
- Evaluate
- Consolidate
- Deploy
- Interact

**KNIME Analytics Platform**
- KNIME Native Nodes
- KNIME Streaming
- DBMS
- R
- Python
- H2O
- Spark
- Keras
- Tensorflow
- Google API
- Microsoft SQL
- R
- Call External

**KNIME Server**
- WebPortal
- REST API

**Environment Contexts**
- Environment:
  - Memory:
  - Cores:
  - Disktype:
- Execution Type:
  - Local / Cloud
- Operating System:
  - Windows

**Environment:**
- Memory:
- Cores:
- Disktype:
- Execution Type:
- Operating System:
Measure and Compare

- Measures
  - Model Performance
  - User Satisfaction, etc.
- Total Runtime
  - As well as individual node runtimes
- Environment Settings
  - Memory Usage
  - Cores
  - Disk Types
  - GPUs
  - etc.
- Execution
- Infrastructure Settings

Helpful Tools

- KNIME Measurement MetaNode
- KNIME Measurement Consolidation Workflow
Measure Workflow Resources and Times - MetaNode

- Measured:
  - Total Execution Time
    - Individual Node Executions
  - Maximum Memory Used
    - Heap Space committed
    - Heap Space used
  - Cores Used
  - Environment Settings
    - OS, Version, etc.
  - INI Settings
  - Your own labels and details
    - Local/Cloud, SDD, etc.
Benchmarking: An Example with 3 Alternatives

- KNIME
- KNIME + H2O
- KNIME + Spark
Calling Workflow: Looping
# Example Scenario Report

<table>
<thead>
<tr>
<th>Method</th>
<th>Hardware Environment</th>
<th>Number Rows</th>
<th>num Processors</th>
<th>XMXinG</th>
<th>heap Committed MB</th>
<th>heap Usaged MB</th>
<th>Duration of Execution seconds</th>
<th>Accuracy MS per record</th>
<th>Runtime Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Native KNIME</td>
<td>Laptop</td>
<td>10000</td>
<td>8</td>
<td>24</td>
<td>5020</td>
<td>582</td>
<td>6</td>
<td>0.605</td>
<td>0.000</td>
</tr>
<tr>
<td>Native KNIME</td>
<td>Laptop</td>
<td>100000</td>
<td>8</td>
<td>24</td>
<td>8429</td>
<td>1314</td>
<td>13</td>
<td>0.614</td>
<td>0.000</td>
</tr>
<tr>
<td>Native KNIME</td>
<td>Laptop</td>
<td>1000000</td>
<td>8</td>
<td>24</td>
<td>15702</td>
<td>6652</td>
<td>160</td>
<td>0.613</td>
<td>0.000</td>
</tr>
<tr>
<td>Native KNIME</td>
<td>AWS</td>
<td>10000</td>
<td>16</td>
<td>50</td>
<td>15657</td>
<td>945</td>
<td>6</td>
<td>0.606</td>
<td>0.124</td>
</tr>
<tr>
<td>Native KNIME</td>
<td>AWS</td>
<td>10000</td>
<td>16</td>
<td>50</td>
<td>18208</td>
<td>1373</td>
<td>13</td>
<td>0.609</td>
<td>0.262</td>
</tr>
<tr>
<td>Native KNIME</td>
<td>AWS</td>
<td>1000000</td>
<td>16</td>
<td>50</td>
<td>24759</td>
<td>6058</td>
<td>86</td>
<td>0.615</td>
<td>0.09</td>
</tr>
<tr>
<td>H2O in KNIME</td>
<td>Laptop</td>
<td>10000</td>
<td>8</td>
<td>24</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>H2O in KNIME</td>
<td>Laptop</td>
<td>100000</td>
<td>8</td>
<td>24</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>H2O in KNIME</td>
<td>Laptop</td>
<td>1000000</td>
<td>8</td>
<td>24</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>H2O in KNIME</td>
<td>AWS</td>
<td>10000</td>
<td>16</td>
<td>60</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>H2O in KNIME</td>
<td>AWS</td>
<td>10000</td>
<td>16</td>
<td>60</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>H2O in KNIME</td>
<td>AWS</td>
<td>10000</td>
<td>16</td>
<td>60</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>Apache Spark</td>
<td>Laptop</td>
<td>10000</td>
<td>8</td>
<td>24</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>Apache Spark</td>
<td>Laptop</td>
<td>100000</td>
<td>8</td>
<td>24</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>Apache Spark</td>
<td>Laptop</td>
<td>1000000</td>
<td>8</td>
<td>24</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>Apache Spark</td>
<td>AWS</td>
<td>10000</td>
<td>16</td>
<td>60</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>Apache Spark</td>
<td>AWS</td>
<td>10000</td>
<td>16</td>
<td>60</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>Apache Spark</td>
<td>AWS</td>
<td>10000</td>
<td>16</td>
<td>60</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
</tbody>
</table>
## Example Comparison Report and Drill Down

<table>
<thead>
<tr>
<th>Method</th>
<th>Number Rows</th>
<th>Laptop</th>
<th>AWS</th>
<th>Speedup in MS</th>
<th>% Speedup</th>
<th>Cost of Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Native KNIME</td>
<td>10000</td>
<td>6383</td>
<td>6185</td>
<td>197</td>
<td>3%</td>
<td>$0.12</td>
</tr>
<tr>
<td>Native KNIME</td>
<td>100000</td>
<td>13190</td>
<td>13116</td>
<td>74</td>
<td>1%</td>
<td>$0.26</td>
</tr>
<tr>
<td>Native KNIME</td>
<td>1000000</td>
<td>160103</td>
<td>85516</td>
<td>74587</td>
<td>47%</td>
<td>$1.71</td>
</tr>
<tr>
<td>H2O in KNIME</td>
<td>10000</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
</tr>
<tr>
<td>H2O in KNIME</td>
<td>100000</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
</tr>
<tr>
<td>H2O in KNIME</td>
<td>1000000</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
</tr>
<tr>
<td>Apache Spark</td>
<td>10000</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
</tr>
<tr>
<td>Apache Spark</td>
<td>100000</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
</tr>
<tr>
<td>Apache Spark</td>
<td>1000000</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
<td>xxx</td>
</tr>
</tbody>
</table>
Benchmarking Performance & Scalability in Practice

• Public EXAMPLES Server:
  - 53_Performance_and_Scalability
    - 01_workflows
      - 00_Scalable_Data_Generation
      - 01_KNIME_for_Performance
      - 02_H2O_for_Performance
      - 03_Spark_for_Performance
    - 02_data
      - data.csv
      - Measurements.csv
    - 03_Metanode_Templates
      - Measure Workflow Resources and Times
      - 04_Control_Workflow_for_Performance_and_Scalability_Measurements

• This PPT

• Blog and Whitepaper to follow
KNIME Champions

KNIME Champions

Table

© 2019 KNIME AG. All rights reserved.
The KNIME® trademark and logo and OPEN FOR INNOVATION® trademark are used by KNIME AG under license from KNIME GmbH, and are registered in the United States. KNIME® is also registered in Germany.