Enterprise-Scale MATLAB Applications

Sylvain Lacaze
Rory Adams
Enterprise Integration

Access and Explore Data → Preprocess Data → Develop Predictive Models → Integrate Analytics with Systems

Databases
- DynamoDB
- MongoDB
- SQL Server
- Cassandra
- Cosmos DB

Cloud Storage
- Azure Blob

Big Data / OT
- cloudera
- Hortonworks
- OSIsoft

Streaming
- AWS Kinesis
- Azure IoT Hub

OT Platforms
- OSIsoft PI System

Dashboards
- Qlik Sense
- Microsoft Power BI
- Spotfire

MATLAB

MDCS

Request Broker

MATLAB Production Server
MATLAB at Scale

Scale with increasing

- Access
- Computational Complexity
- Data Volume and Velocity
Key Takeaways

1. Share applications and algorithms with anyone

2. Integrate MATLAB functions into existing workflows and development platforms.

3. Deploy MATLAB applications within enterprise systems in a scalable manner
Write Once Then Share To Different Targets

MATLAB

MATLAB Compiler

MATLAB Compiler SDK

MATLAB Coder

GPU Coder

Apps
Files
Custom Toolbox

Standalone Application
Hadoop
Excel Add-in
Web Apps

C/C++
Python
MATLAB Production Server

Java
.NET

.c,cpp

CUDA
Easily share apps with your team using Web Apps

- Use apps in browser
- Easy deployment
- No required knowledge of web technologies
Integrate MATLAB-based Components With Your Own Software

- Royalty-free Sharing
- IP Protection via Encryption
Customer examples: Financial customer advisory service

- Saved **€ 2 million annually** for an external system
- Quicker implementation of adjustments in source code by the quantitative analysts
- Knowledge + MATLAB = Build your own systems

Global financial institution with European HQ
Scaling up: Asset Allocation Demo

MATLAB Production Server(s)

HTML
XML
Java Script

Web Server(s)

Asset Allocation Results

Efficient Frontier

Portfolio Composition & Metrics

ALLOCATION

METRICS

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expected Return</td>
<td>10.26%</td>
</tr>
<tr>
<td>Expected Std. Deviation</td>
<td>8.68%</td>
</tr>
<tr>
<td>Expected Sharpe Ratio</td>
<td>1.04%</td>
</tr>
<tr>
<td>Single-period 99% VaR</td>
<td>0.94%</td>
</tr>
<tr>
<td>Single-period 99% CVaR</td>
<td>0.79%</td>
</tr>
<tr>
<td>Historic Semi-deviation</td>
<td>0.29%</td>
</tr>
<tr>
<td>Historic Max Drawdown</td>
<td>4.44%</td>
</tr>
</tbody>
</table>

Click on a portfolio on the frontier to see its composition, metrics, historical, and future performance data.
MATLAB Production Server with Visualization Platforms

MATLAB analytics for use with desktop, browser, and mobile visualization dashboards

- **Tableau**
  Access models published on MATLAB Production Server inside Tableau calculated fields

- **Spotfire**
  Access models published on MATLAB Production Server inside Spotfire workbooks
Example: Travelling Salesman Problem with Tableau
MATLAB and MATLAB Production Server

- The easiest and most productive environment to *take your enterprise analytics solution* from *idea* to a *scalable production* solution
Production Deployment Workflow

Development

MATLAB Developer

- Initial Test Application
- Debug Algorithm
- MATLAB Algorithm
- MATLAB Compiler SDK
- Deployable Archive

Enterprise Application Developer

Web Application

Function Call

Production

Web Application

Function Calls

...
Scale Up with MATLAB Production Server™

- Scalable and reliable
  - Service large numbers of concurrent requests
  - Add capacity or redundancy with additional servers

- Directly deploy MATLAB programs into production
  - Automatically deploy updates without server restarts
  - Most efficient path for creating enterprise applications
MATLAB Production Server
Enterprise Class Framework For Running Packaged MATLAB Programs

- **Server software**
  - Manages packaged MATLAB programs and worker pool

- **MATLAB Runtime libraries**
  - Single server can use runtimes from different releases

- **RESTful JSON interface and lightweight client library**
  - Isolates the MATLAB processing
  - Access using native data types
MATLAB at Scale

Scale with increasing

- Access
- Computational Complexity
- Data Volumes
Key Takeaways

1. Leverage parallel computing

2. Seamlessly scale to clusters or the cloud
Commerzbank

Compute a variety of derived market data from raw market data

“… can complete urgent change requests ourselves with MATLAB, often on the same day… Testing time has also been reduced, … load data 8 times faster than we could do before.”

Julian Zenglein
Commerzbank

Aberdeen Asset Management

Improve asset allocation strategies with machine learning techniques

“… can develop prototypes to test machine learning techniques quickly… get rapid, reliable results by running the algorithms with large financial data sets on a distributed computing cluster.”

Emilio Llorente-Cano
Aberdeen Asset Management
Accelerating MATLAB Applications

Parallel-enabled toolboxes

Simple programming constructs

Advanced programming constructs

Ease of Use

Greater Control
Parallel-enabled Toolboxes (MATLAB® Product Family)
Enable acceleration by setting a flag or preference

Statistics and Machine Learning
Resampling Methods, k-Means clustering, GPU-enabled functions

Optimization
Estimation of gradients

Neural Networks
Deep Learning, Neural Network training and simulation

Signal Processing and Communications
GPU-enabled FFT filtering, cross correlation, BER simulations

Computer Vision
Bag-of-words workflow

Image Processing
Batch Image Processor, Block Processing, GPU-enabled functions

Other Parallel-enabled Toolboxes
Independent Tasks or Iterations
Simple programming constructs: `parfor`, `parfeval`

- Examples: parameter sweeps, Monte Carlo simulations
- No dependencies or communications between tasks
Parallel Computing
Multicore Desktops
Parallel Computing – Scaling Up
Clusters/Cloud

Parallel Computing Toolbox

MATLAB Distributed Computing Server
## Summary - Scale your applications beyond the desktop

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Explicit desktop scaling</td>
<td>Scale to clusters</td>
<td>Scale to EC2 with some customization</td>
<td>Scale to custom cloud</td>
<td>Scale to custom cloud</td>
</tr>
<tr>
<td>Maximum workers</td>
<td>No limit</td>
<td>No limit</td>
<td>256</td>
<td>No limit</td>
<td>No limit</td>
</tr>
<tr>
<td>Hardware</td>
<td>Desktop</td>
<td>Any</td>
<td>Amazon EC2</td>
<td>Amazon EC2, Microsoft Azure, Others</td>
<td>Hadoop + Spark</td>
</tr>
<tr>
<td>Availability</td>
<td>Worldwide</td>
<td>Worldwide</td>
<td>United States, Canada and other select countries in Europe</td>
<td>Worldwide</td>
<td>Worldwide</td>
</tr>
</tbody>
</table>

Learn More: [Parallel Computing on the Cloud](#)
MATLAB at Scale

Scale with increasing

- Access
- Computational Complexity
- Data Volume and Velocity
Big Data – Batch vs Stream Processing

- **Batch Processing** applies computation to a finite sized historical data set that was acquired in the past.

- **Stream Processing** applies computation to an unbounded data set that is produced continuously.
Large Data Options

Data fits in memory of pool
- Distributed arrays
  - Look like normal MATLAB variables

Data does not fit in memory (Big Data)
- Tall arrays
  - Looks like normal MATLAB variables
- Custom map-reduce functions
  - Can be painful to learn
Typical Workflow – Big Data

1. Access and Explore Data
2. Preprocess Data
3. Develop Predictive Models
4. Integrate Analytics with Systems

Data

“Package” MATLAB code & run as automated batch

MATLAB Developer

HDFS

Spark or MapReduce job
tall arrays

- Data doesn’t fit into memory
- Lots of observations - “tall”
- Looks like a normal MATLAB array
  - Numeric types, tables, datetimes, strings, etc…
  - Basic math, stats, indexing, etc.
  - Statistics and Machine Learning Toolbox
    (clustering, classification, etc.)
Example: Taxi Data

**tall Arrays for Big Data in MATLAB**

**Predict Cost of Taxi Ride in New York City**

This example explores NYC taxi data and predicts the fare based on distance and the time of day.

The data come from .csv files containing taxi trip information, separated by month. The data set is freely available from the City of New York.

Set up execution environment

Use local environment for prototyping. This will later be scaled to run on a Spark-enabled Hadoop cluster.

```matlab
% parpool local;
```

Create a datastore to represent the data
Using Tall Arrays

- Tall arrays
  - MATLAB
- 100’s of functions supported
  - MATLAB
  - Statistics and Machine Learning Toolbox
- Run in parallel
  - Parallel Computing Toolbox

- Run in parallel on compute clusters
  - MATLAB Distributed Computing Server

- Run in parallel on Spark clusters
  - MATLAB Distributed Computing Server
- Deploy MATLAB applications as standalone applications on Spark clusters
  - MATLAB Compiler

Local disk
Shared folders
Databases
HDFS

Spark + Hadoop
Connecting MATLAB Production Server to Kafka

- Kafka client for MATLAB Production Server feeds topics to functions deployed on the server

- Configurable batch of messages passed as a MATLAB Timetable

- Each consumer process feeds one topic to a specified function

- Drive everything from a simple config file
  - No programming outside of MATLAB!
Develop a Stream Processing Function in MATLAB
Streaming Data: Treated as an Unbounded Timetable

### Input Table

<table>
<thead>
<tr>
<th>Event Time</th>
<th>Vehicle</th>
<th>RPM</th>
<th>Torque</th>
<th>Fuel Flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>18:01:10</td>
<td>55a3fd</td>
<td>1975</td>
<td>100</td>
<td>110</td>
</tr>
<tr>
<td>18:10:00</td>
<td>55a3fe</td>
<td>2000</td>
<td>109</td>
<td>115</td>
</tr>
<tr>
<td>18:05:20</td>
<td>55a3fd</td>
<td>1980</td>
<td>105</td>
<td>105</td>
</tr>
<tr>
<td>18:10:45</td>
<td>55a3fd</td>
<td>2100</td>
<td>110</td>
<td>100</td>
</tr>
<tr>
<td>18:30:10</td>
<td>55a419</td>
<td>2000</td>
<td>100</td>
<td>110</td>
</tr>
<tr>
<td>18:35:20</td>
<td>55a419</td>
<td>1960</td>
<td>103</td>
<td>105</td>
</tr>
<tr>
<td>18:20:40</td>
<td>55a3fe</td>
<td>1970</td>
<td>112</td>
<td>104</td>
</tr>
<tr>
<td>18:39:30</td>
<td>55a419</td>
<td>2100</td>
<td>105</td>
<td>110</td>
</tr>
<tr>
<td>18:30:00</td>
<td>55a3fe</td>
<td>1980</td>
<td>110</td>
<td>113</td>
</tr>
<tr>
<td>18:30:50</td>
<td>55a3fe</td>
<td>2000</td>
<td>100</td>
<td>110</td>
</tr>
</tbody>
</table>

### Output Table

<table>
<thead>
<tr>
<th>Time window</th>
<th>Vehicle</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>18:00:00</td>
<td>55a3fd</td>
<td>5</td>
</tr>
<tr>
<td>18:10:00</td>
<td>55a3fe</td>
<td>...</td>
</tr>
<tr>
<td>18:20:00</td>
<td>55a419</td>
<td>...</td>
</tr>
<tr>
<td>18:10:00</td>
<td>55a3fd</td>
<td>7</td>
</tr>
<tr>
<td>18:20:00</td>
<td>55a3fe</td>
<td>3</td>
</tr>
<tr>
<td>18:30:00</td>
<td>55a419</td>
<td>...</td>
</tr>
<tr>
<td>18:30:00</td>
<td>55a3fd</td>
<td>...</td>
</tr>
<tr>
<td>18:40:00</td>
<td>55a3fe</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>55a419</td>
<td>8</td>
</tr>
</tbody>
</table>
**Typical Workflow**

1. **Access and Explore Data**
2. **Preprocess Data**
3. **Develop Predictive Models**
4. **Integrate Analytics with Systems**

**Data Integration and Analytics Tools**
- **Spark or MapReduce job**
- **HDFS**
- **MATLAB Developer**
- **Application Server**:
  - Streaming
  - Ad-hoc analysis via Spotfire, Tableau, …
  - Other…

**Integrate Analytics with Systems**
- “Package” MATLAB code & run as automated batch
- **MATLAB Production Server**
- **MATLAB Analytics**
- **Spark**
- **Kafka**
- **Azure IoT Hub**
- **Other**
Time versus Value in decision making

- Near Real time decisions
- Time critical decisions
- Big Data processing on historical data

Edge Processing with Generated Code, C/C++

Stream Processing
- kafka
- Event Hub
- Kinesis

Batch Processing
- hadoop
- Spark

Value of data to decision making
- Preventive / Predictive
- Actionable
- Reactive
- Historical

Time to React
- Real-Time
- Seconds
- Minutes
- Hours
- Days
- Months