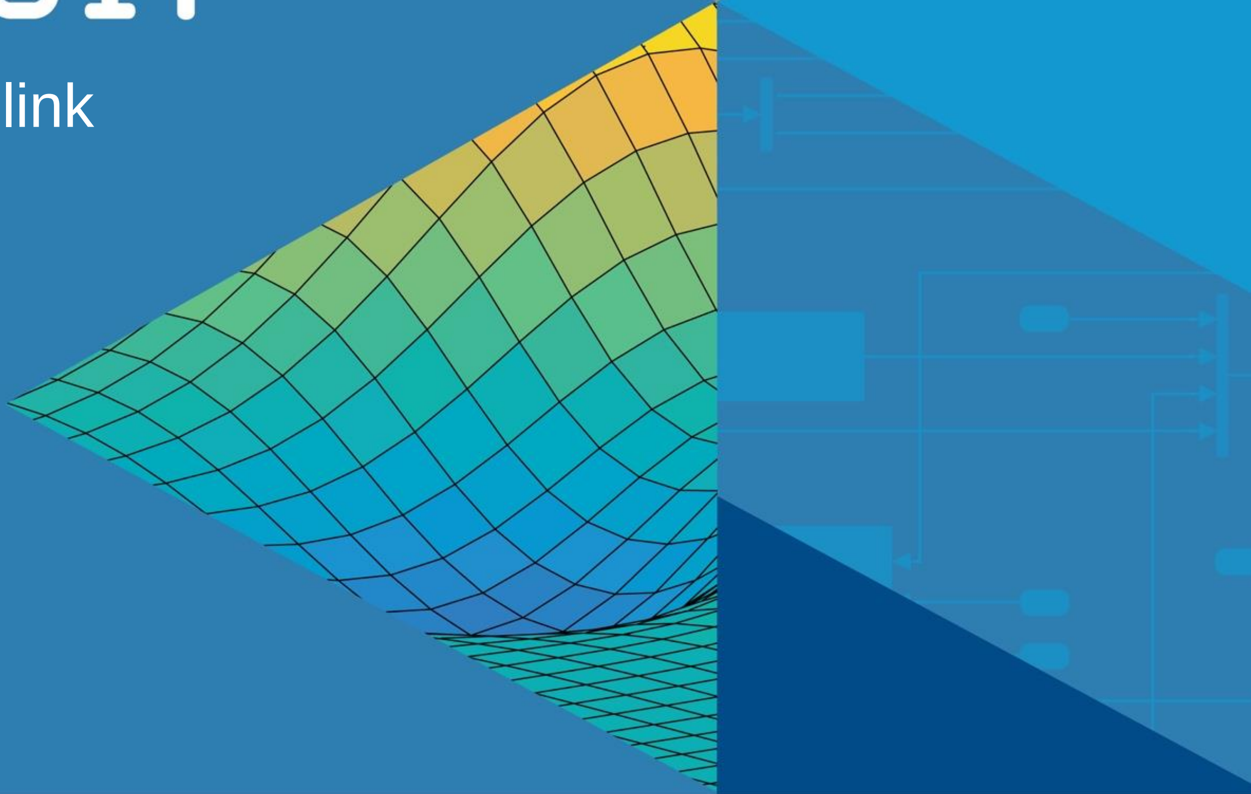
MATLAB EXPO 2017

Parallel Computing with MATLAB and Simulink



Alka Nair
Application Engineer
MathWorks India Private Limited

Why Parallel Computing ?

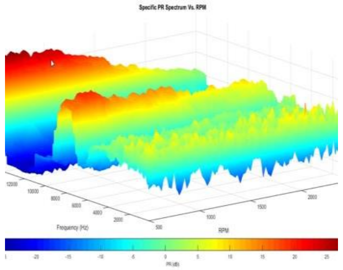
- Size and complexity of analytical problems is growing across industries
- Need faster insight to bring competitive products to market quickly
- Hardware is becoming powerful: Leverage computational power of multicore desktops, GPUs, clusters

Key Takeaways

➢ Overcome challenges with MathWorks Parallel Computing Tools

- Save engineering and research time and focus on results
- Leverage computational power of broadly available hardware with minimal changes to your existing code
  - (Multicore Desktops, GPUs, Clusters)
- Seamlessly scale from your desktop to clusters or the cloud
- Speed-up analysis of Big Data using built-in parallel computing capabilities

Where is Parallel Computing Used?



Bosch Develops Platform for Automotive Test Data Analysis and Visualization
Validation time **reduced by 40-50%**
3-4 months of development time saved

Heart Transplant Studies
4 weeks reduced to 5 days

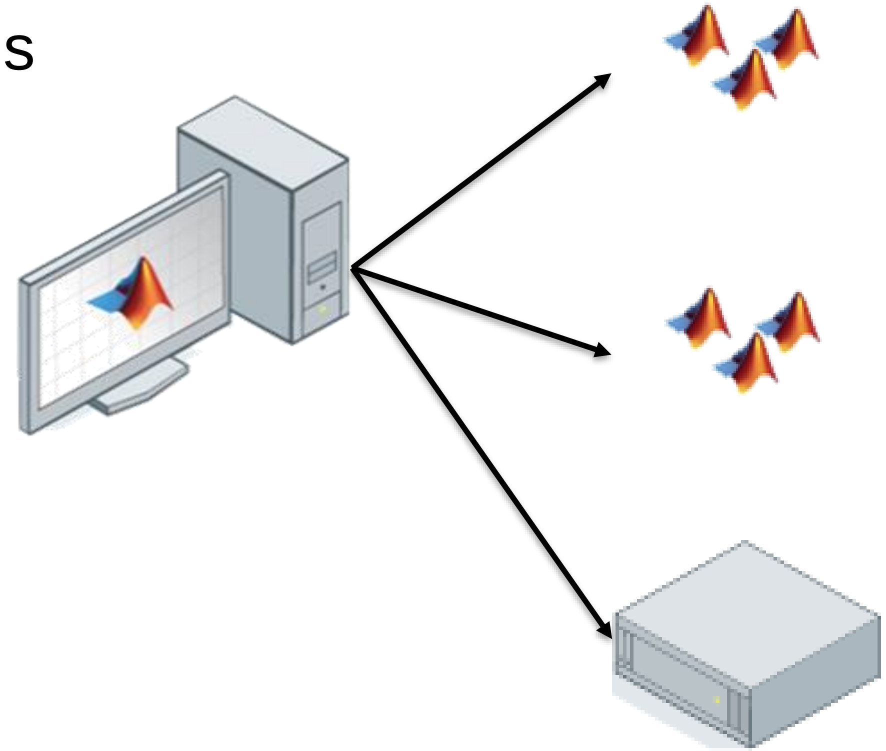
Carnegie Wave Energy Designs and Builds Wave Energy Farm
Sensitivity studies accelerated 12x

Lockheed Martin Builds Discrete-Event Model of Fleet Performance
Simulation time reduced **from months to hours**
20X faster simulation time
Linkage with Neural Network Toolbox

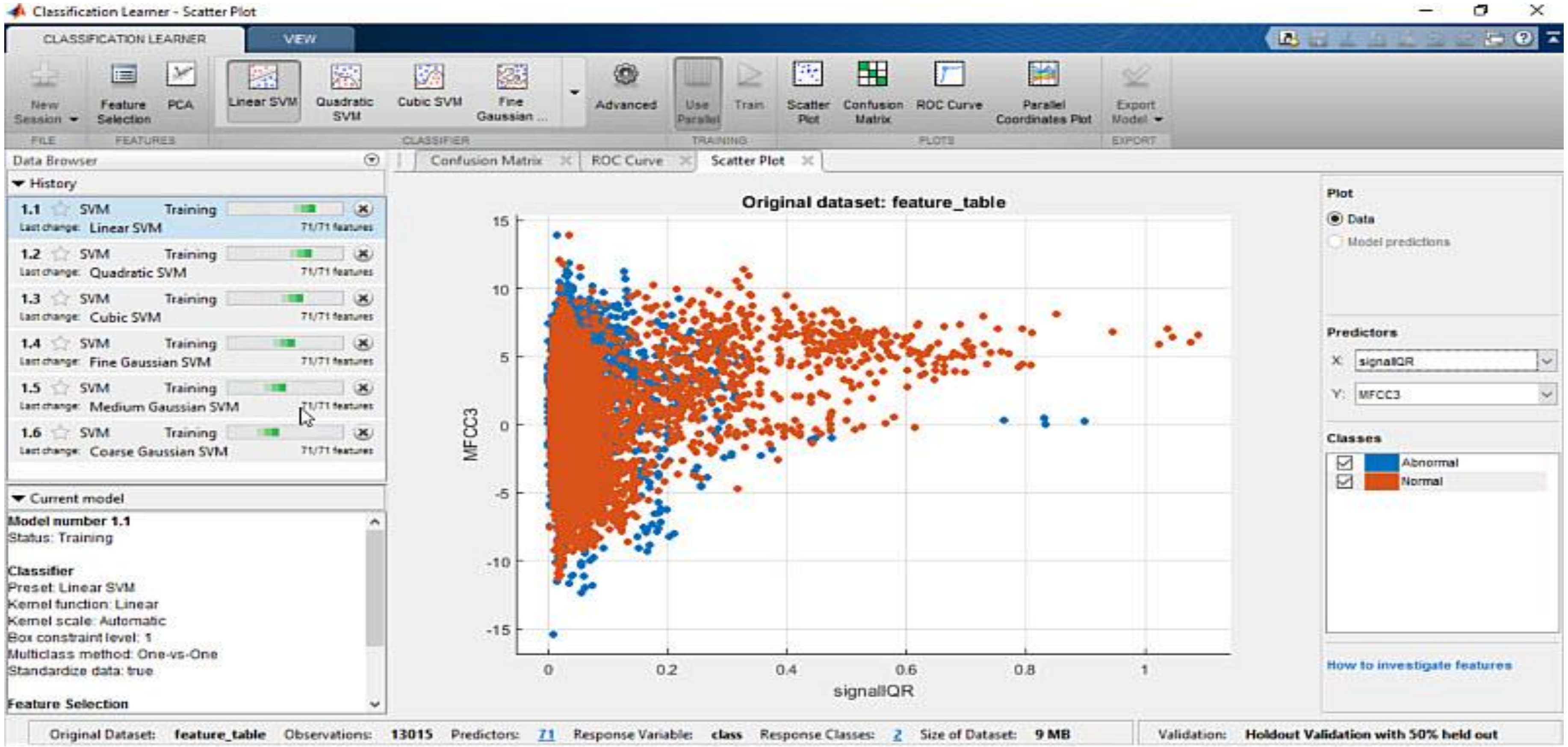
Commerzbank Develops Production Software Systems for Calculating Derived Market Data
Implementation time **reduced by months**
Updates loaded 8X faster

Agenda

- Parallel computing paradigms in MATLAB and Simulink
- Accelerate applications with NVIDIA GPUs
- Scaling to clusters and clouds
- Summary



Classification learner demo



▼ History

1.1 ☆	SVM	Training	⊗
Last change:	Linear SVM		71/71 features
1.2 ☆	SVM	Training	⊗
Last change:	Quadratic SVM		71/71 features
1.3 ☆	SVM	Training	⊗
Last change:	Cubic SVM		71/71 features
1.4 ☆	SVM	Training	⊗
Last change:	Fine Gaussian SVM		71/71 features
1.5 ☆	SVM	Training	⊗
Last change:	Medium Gaussian SVM		71/71 features
1.6 ☆	SVM	Training	⊗
Last change:	Coarse Gaussian SVM		71/71 features

▼ Current model

Model number 1.1
Status: Training

Classifier
Preset: Linear SVM
Kernel function: Linear
Kernel scale: Automatic
Box constraint level: 1
Multiclass method: One-vs-One
Standardize data: true

Feature Selection

● Data
○ Model predictions

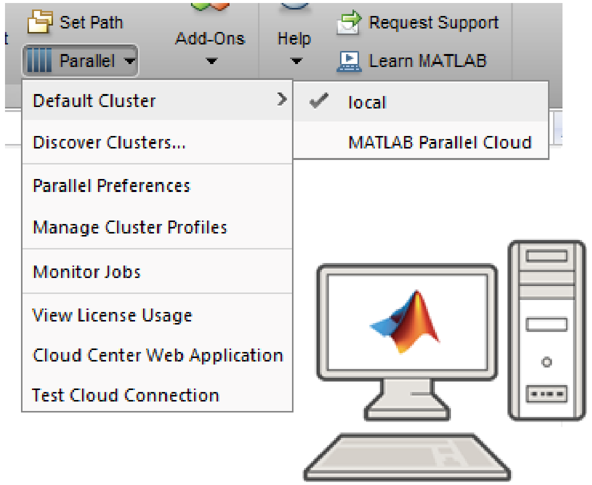
X: signalIQR
Y: MFCC3

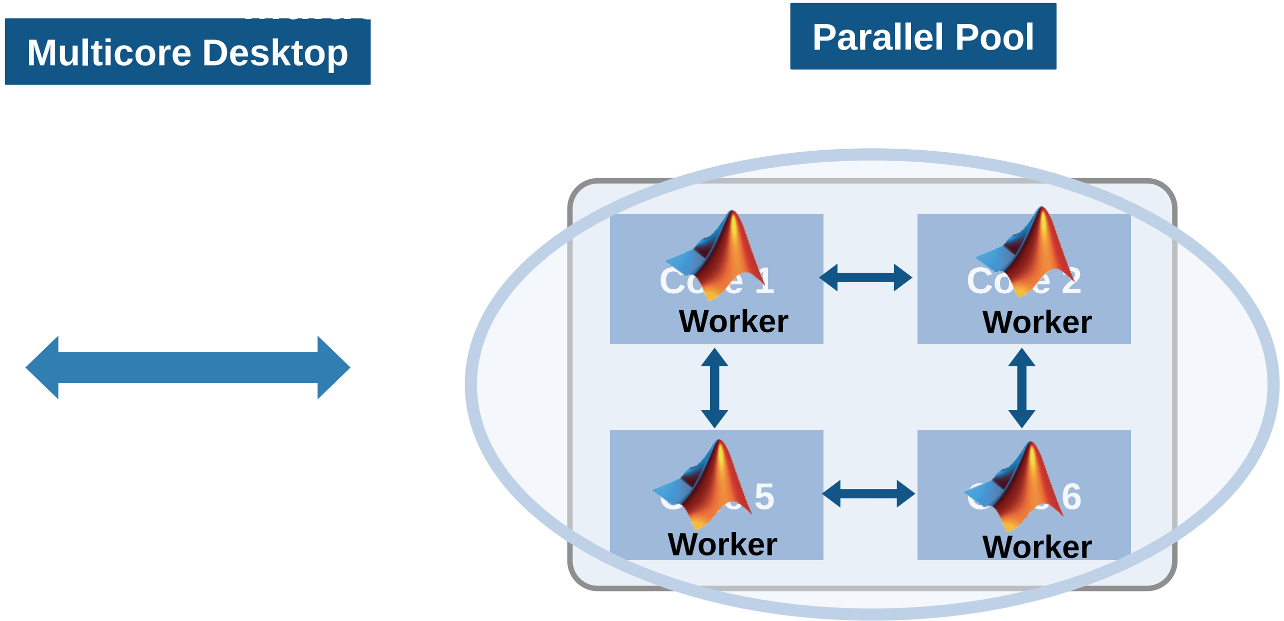
☑ 🟦 Abnormal
☑ 🟥 Normal

Original Dataset: **feature_table** Observations: **13015** Predictors: 71 Response Variable: **class** Response Classes: 2 Size of Dataset: **9 MB** Validation: **Holdout Validation with 50% held out**

Parallel Computing Paradigm - Hardware

Multicore Desktops

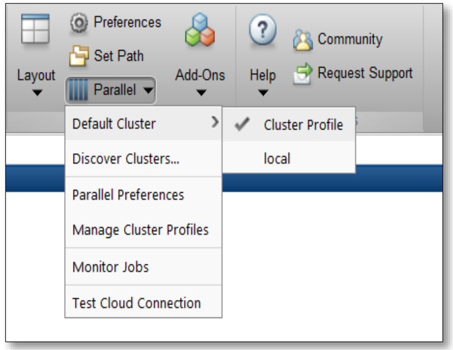


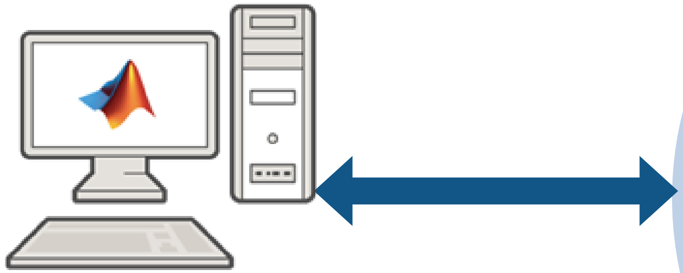


MATLAB multicore

Parallel Computing Paradigm - Hardware

Cluster Hardware

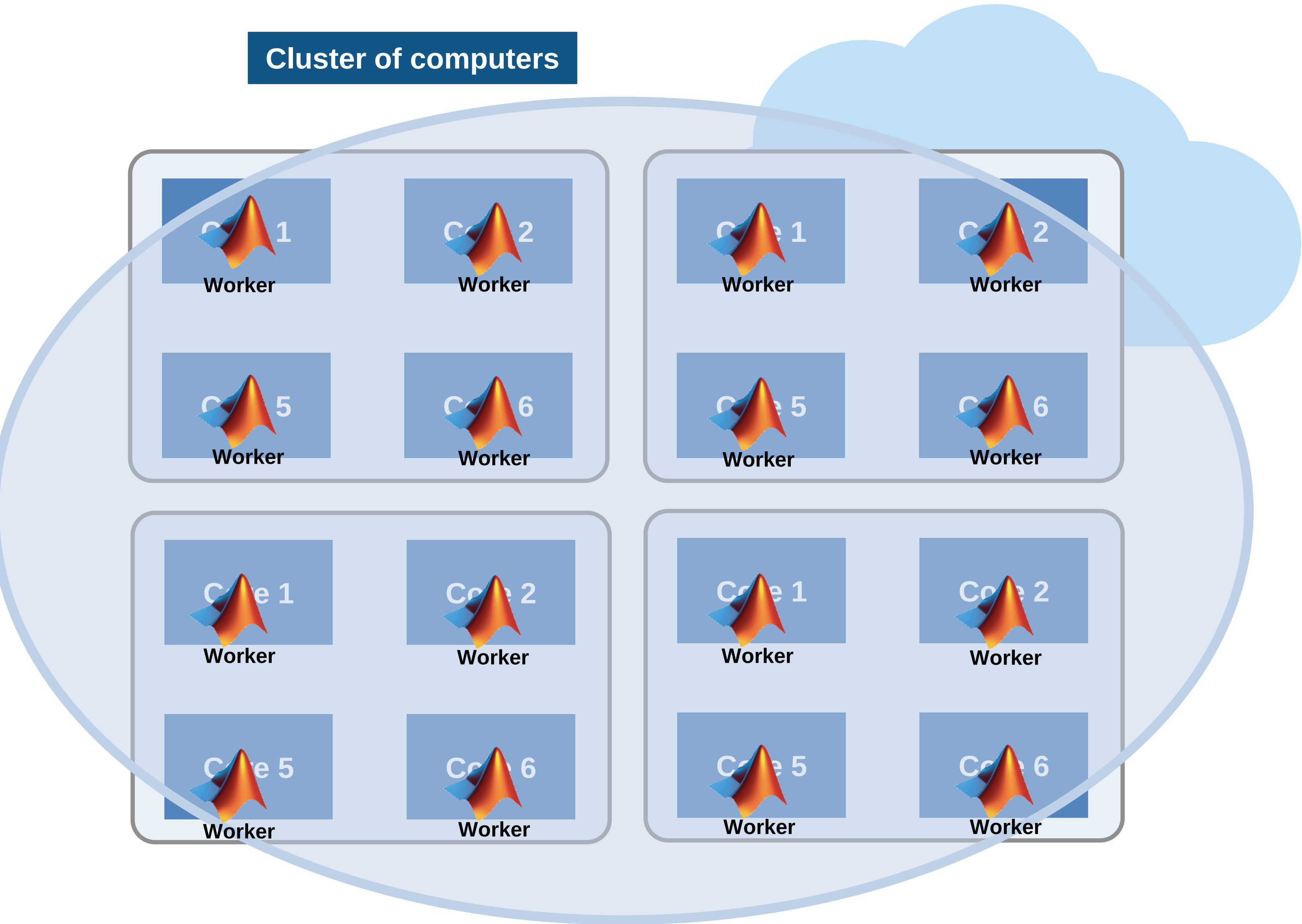




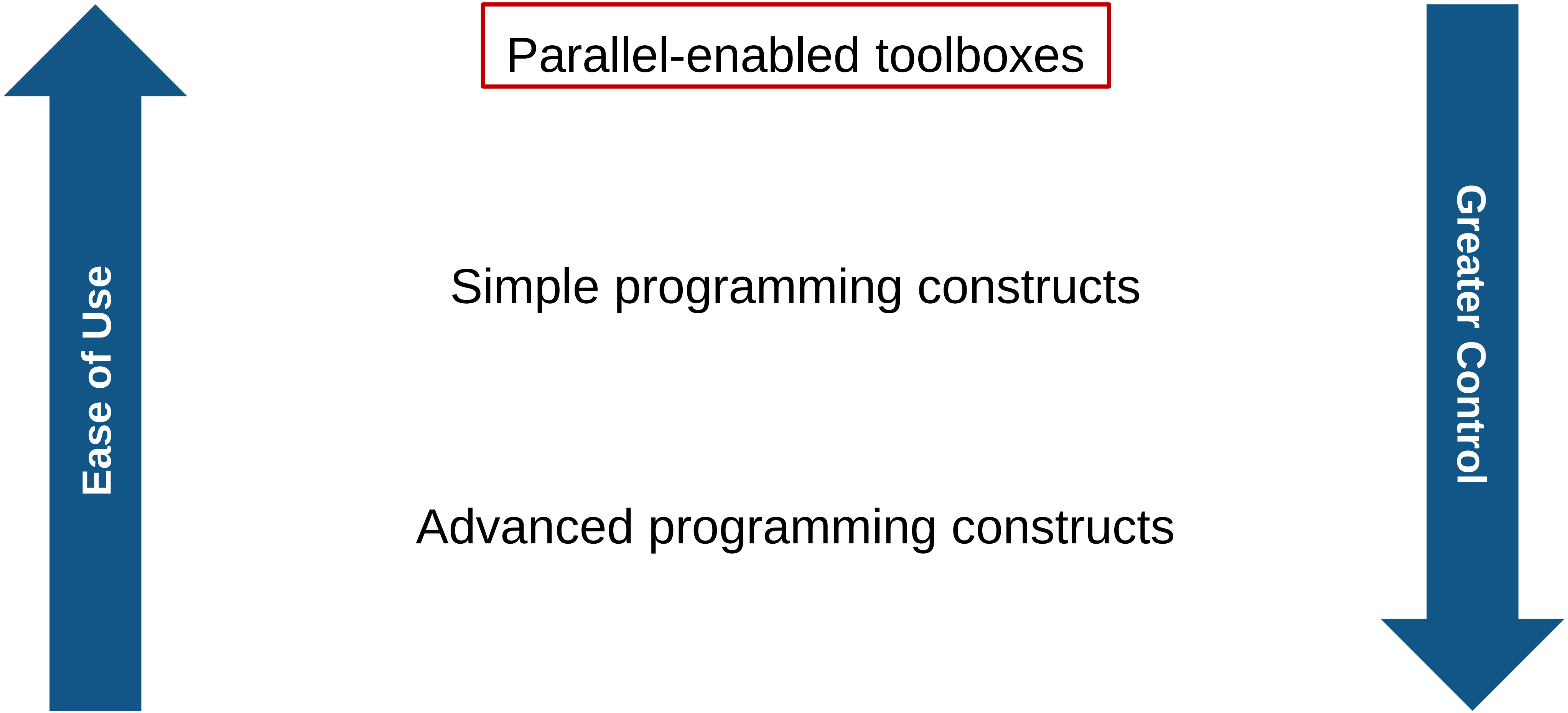
MATLAB Desktop
(client)

Cloud

Cluster of computers



Programming Parallel Applications – Programming Constructs



Ease of Use

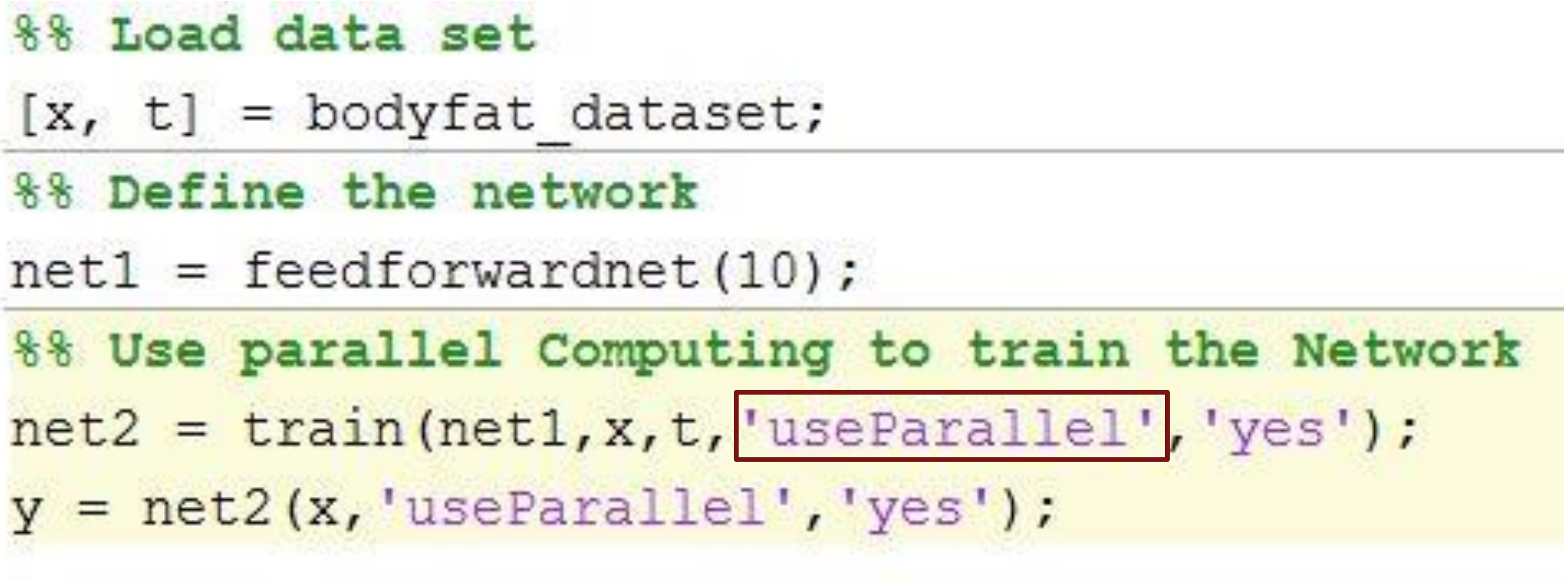
Parallel-enabled toolboxes

Simple programming constructs

Advanced programming constructs

Greater Control

Parallel Computing: Neural Network Toolbox



```
%% Load data set
[x, t] = bodyfat_dataset;
%% Define the network
net1 = feedforwardnet(10);
%% Use parallel Computing to train the Network
net2 = train(net1,x,t,'useParallel','yes');
y = net2(x,'useParallel','yes');
```

Parallel-enabled Toolboxes (MATLAB® Product Family)

Enable parallel computing support by setting a flag or preference

Image Processing

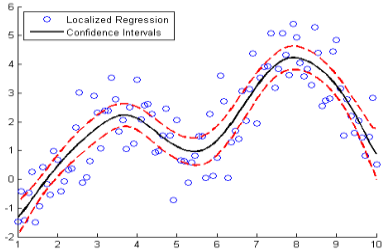
Batch Image Processor, Block Processing, GPU-enabled functions

Original Image of Peppers

Recolored Image of Peppers

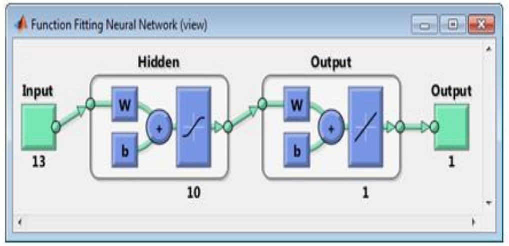
Statistics and Machine Learning

Resampling Methods, k-Means clustering, GPU-enabled functions



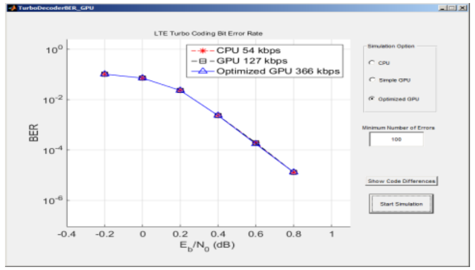
Neural Networks

Deep Learning, Neural Network training and simulation



Signal Processing and Communications

GPU-enabled FFT filtering, cross correlation, BER

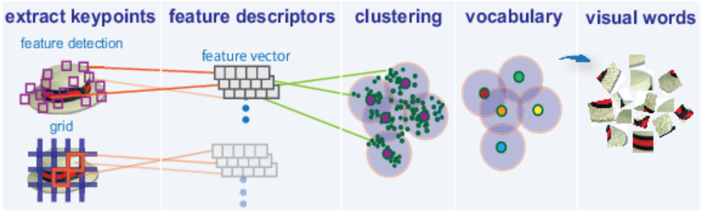


Simulation Option: ○ CPU ○ Simple GPU ● Optimized GPU

Minimum Number of Errors: 100

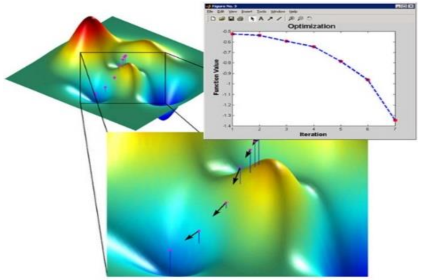
Computer Vision

Parallel-enabled functions in bag-of-words workflow



Optimization

Parallel estimation of gradients



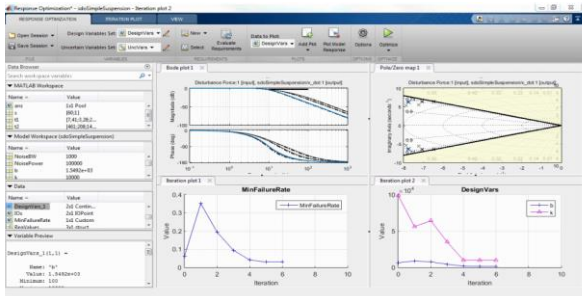
Other Parallel-enabled Toolboxes

Parallel-enabled Toolboxes (Simulink® Product Family)

Enable parallel computing support by setting a flag or preference

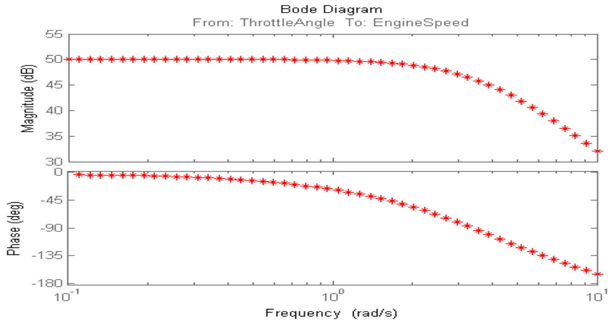
Simulink Design Optimization

Response optimization, sensitivity analysis, parameter estimation



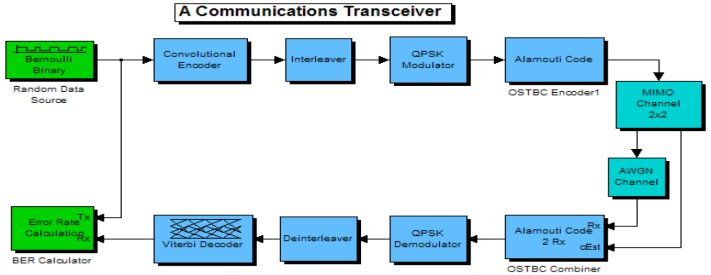
Simulink Control Design

Frequency response estimation



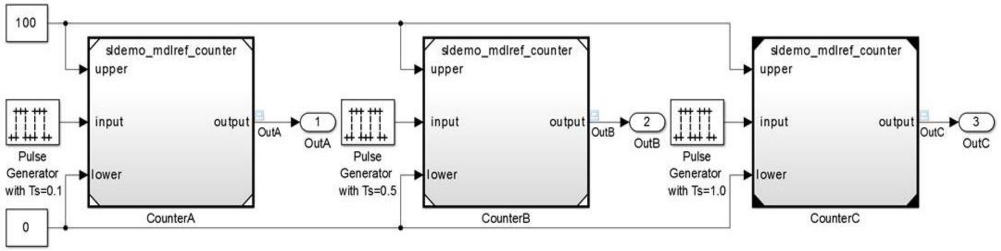
Communication Systems Toolbox

GPU-based System objects for Simulation Acceleration



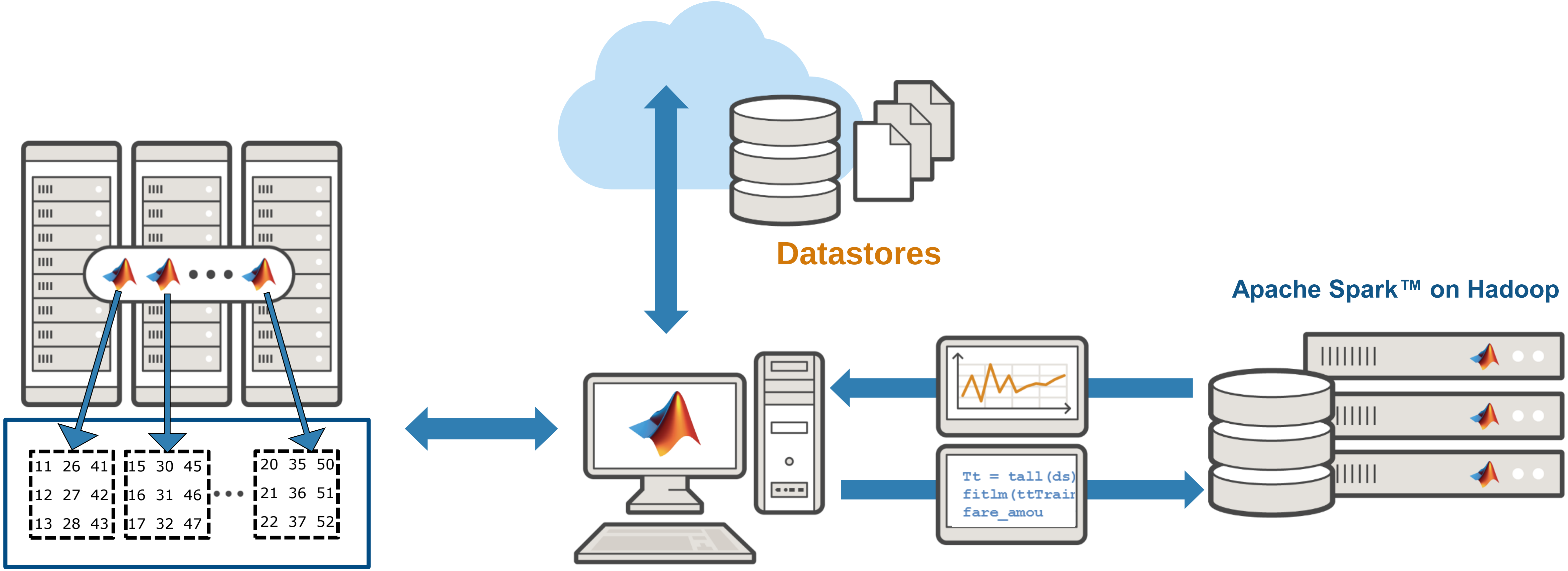
Simulink/Embedded Coder

Generating and building code



Other Parallel-enabled Toolboxes

Data Intensive: Big Data support in MATLAB



Datastores

Apache Spark™ on Hadoop

```
11 26 41    15 30 45  •••  20 35 50
12 27 42    16 31 46       21 36 51
13 28 43    17 32 47       22 37 52
```

```
Tt = tall(ds)
fitlm(ttTrain
fare_amou
```

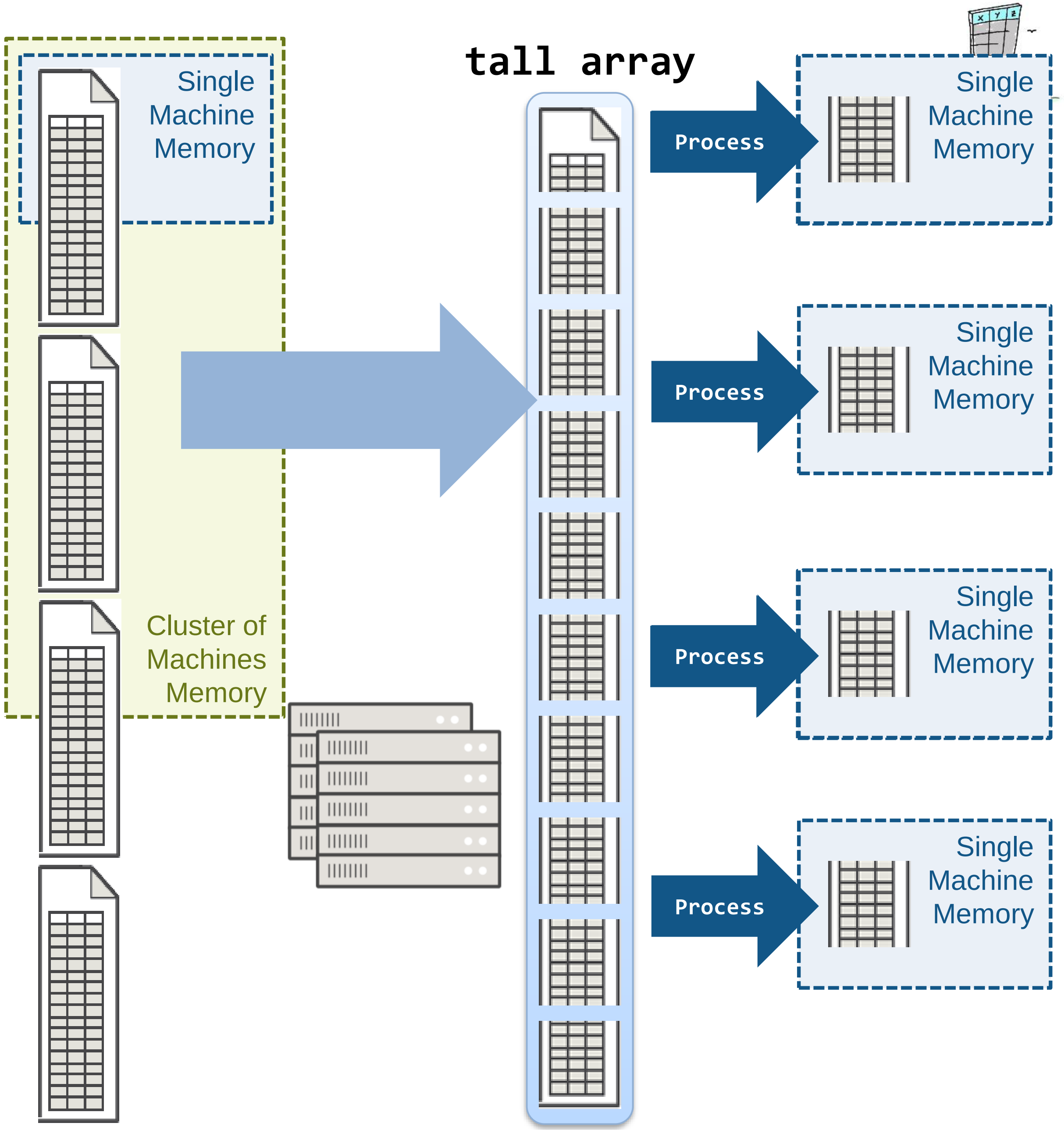
Distributed Arrays

Tall Arrays

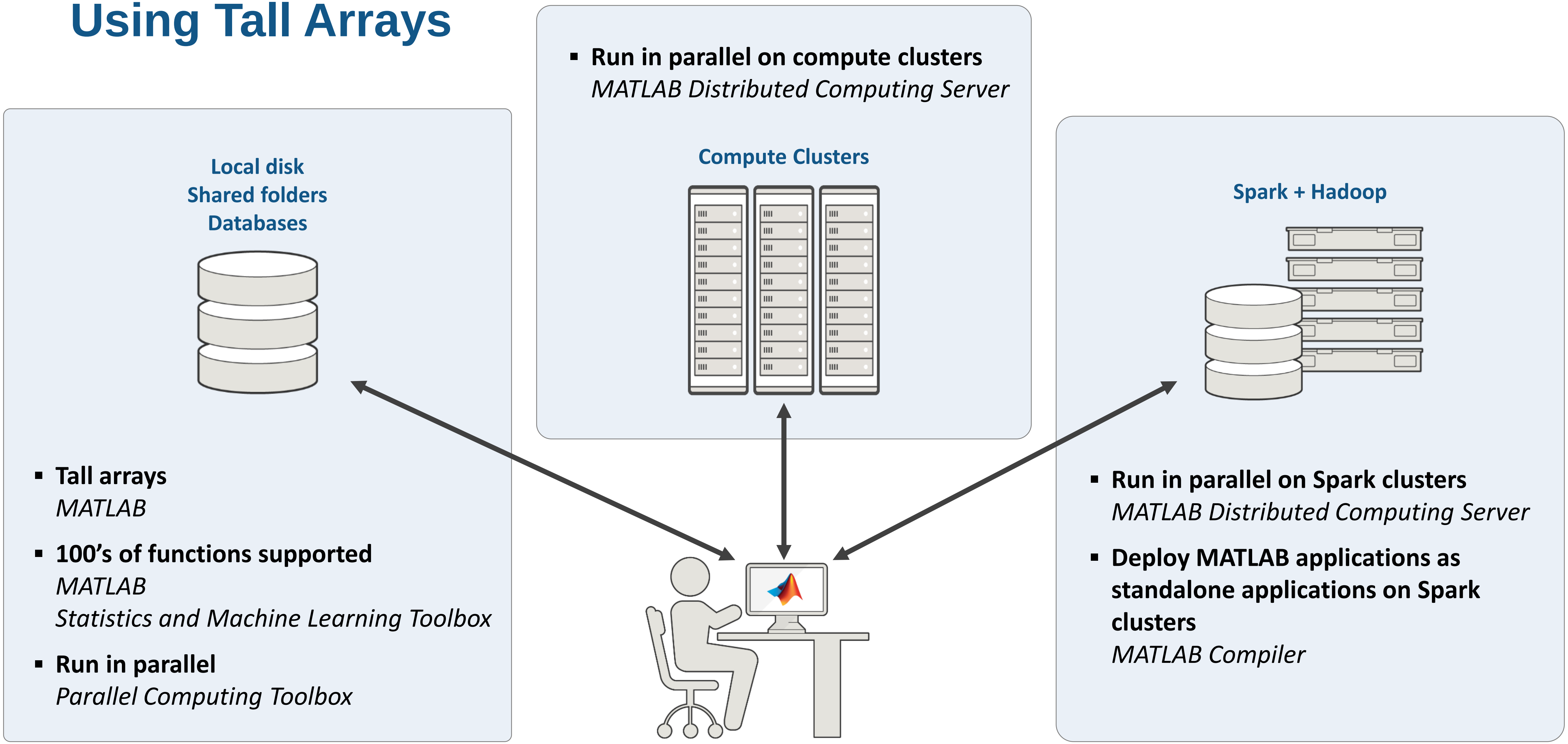
tall arrays R2016b

- With Parallel Computing Toolbox, process several “chunks” at once

- Can scale up to clusters with MATLAB Distributed Computing Server



Using Tall Arrays



Local disk
Shared folders
Databases

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

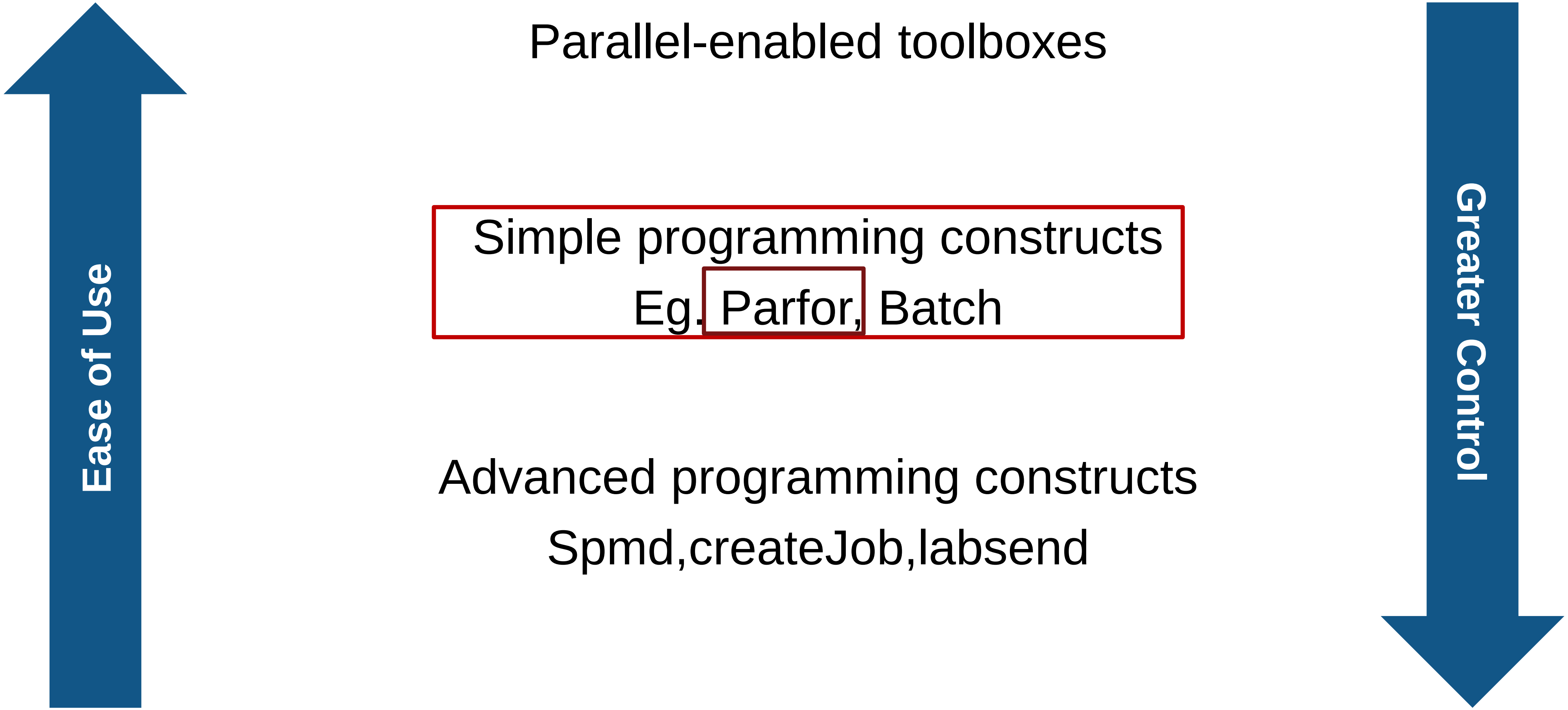
- **Run in parallel on compute clusters**
  MATLAB Distributed Computing Server

Compute Clusters

Spark + Hadoop

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

Programming Parallel Applications – Programming Constructs



Parallel-enabled toolboxes

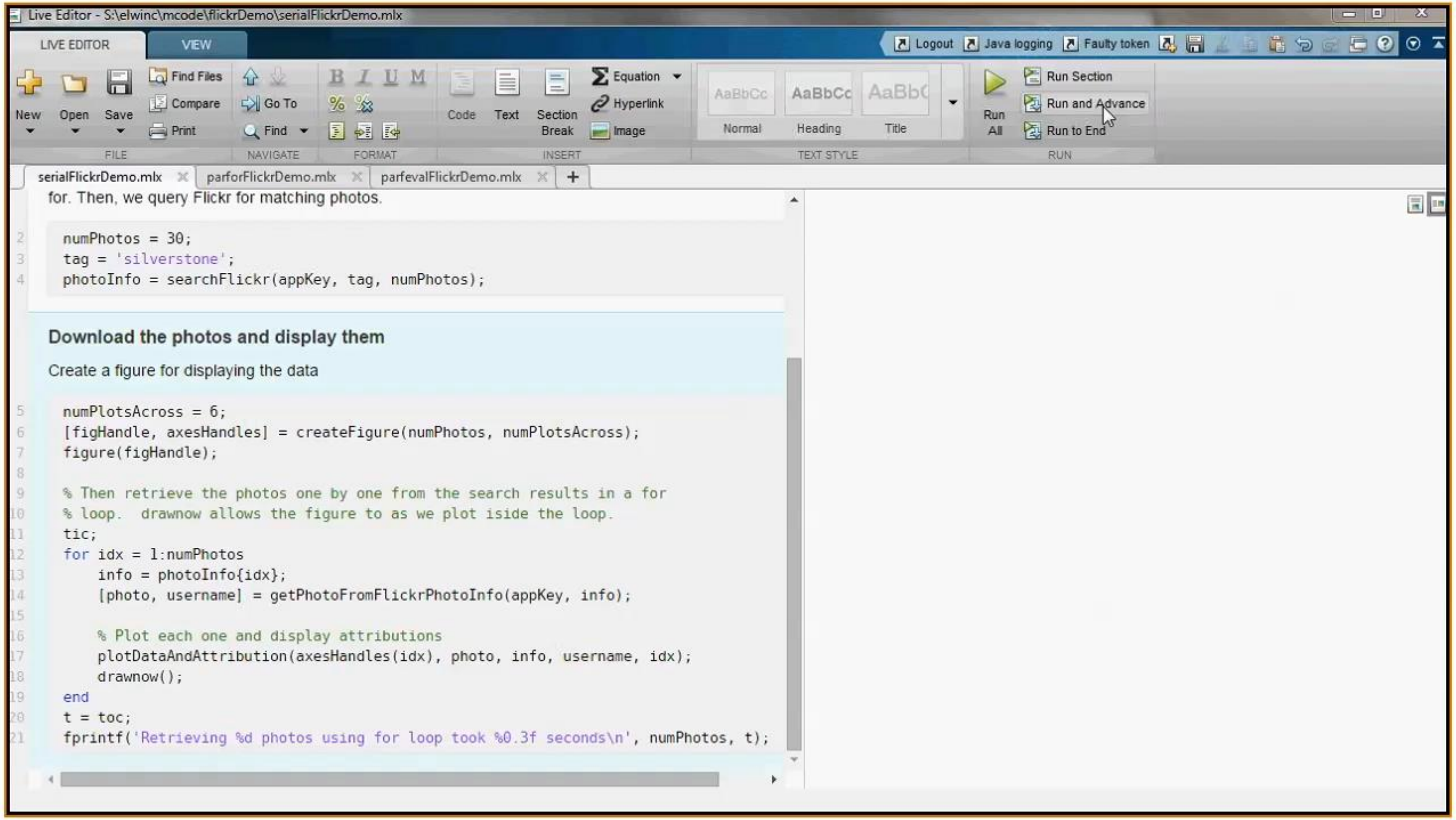
Simple programming constructs

Eg. Parfor, Batch

Advanced programming constructs

Spmd,createJob,labsend

Demo: Getting Data from a Web API



for. Then, we query Flickr for matching photos.

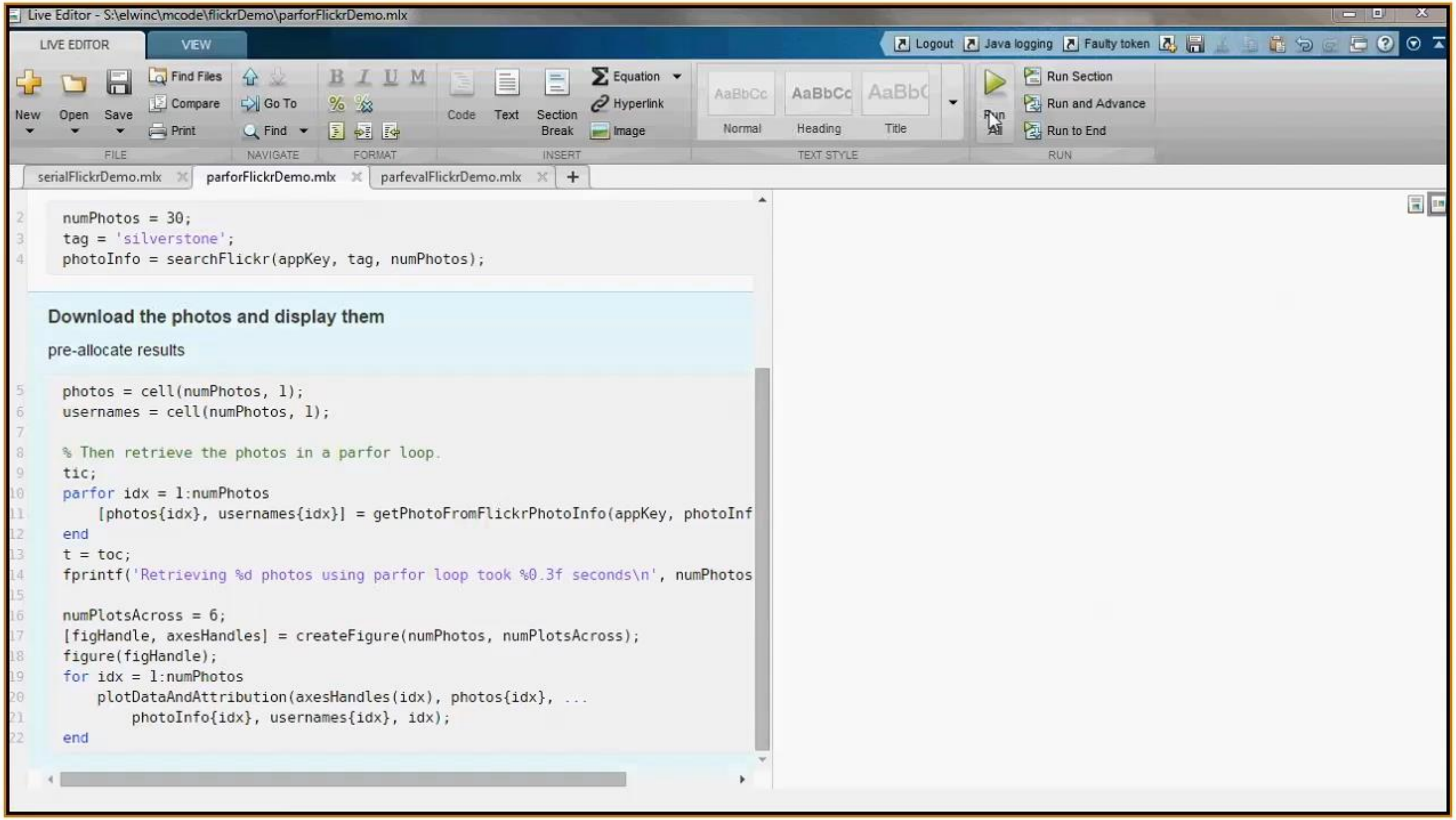
```
numPhotos = 30;
tag = 'silverstone';
photoInfo = searchFlickr(appKey, tag, numPhotos);
```

Download the photos and display them

Create a figure for displaying the data

```
numPlotsAcross = 6;
[figHandle, axesHandles] = createFigure(numPhotos, numPlotsAcross);
figure(figHandle);

% Then retrieve the photos one by one from the search results in a for
% loop.  drawnow allows the figure to as we plot iside the loop.
tic;
for idx = 1:numPhotos
    info = photoInfo{idx};
    [photo, username] = getPhotoFromFlickrPhotoInfo(appKey, info);

    % Plot each one and display attributions
    plotDataAndAttribution(axesHandles(idx), photo, info, username, idx);
    drawnow();
end
t = toc;
fprintf('Retrieving %d photos using for loop took %0.3f seconds\n', numPhotos, t);
```

Demo: Getting Data from a Web API using `parfor`



```
numPhotos = 30;
tag = 'silverstone';
photoInfo = searchFlickr(appKey, tag, numPhotos);
```

Download the photos and display them

pre-allocate results

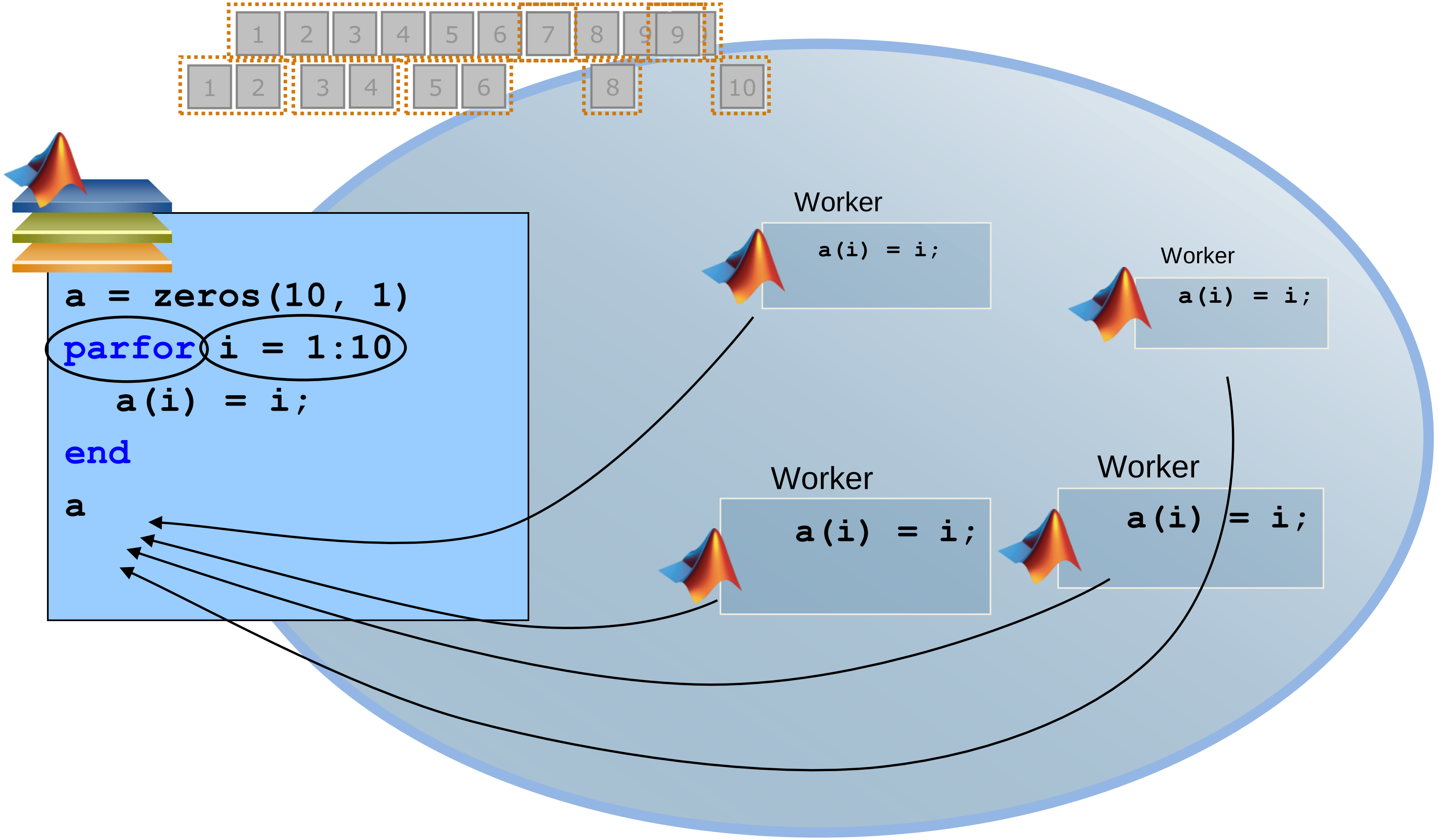
```
photos = cell(numPhotos, 1);
usernames = cell(numPhotos, 1);

% Then retrieve the photos in a parfor loop.
tic;
parfor idx = 1:numPhotos
    [photos{idx}, usernames{idx}] = getPhotoFromFlickrPhotoInfo(appKey, photoInf
end
t = toc;
fprintf('Retrieving %d photos using parfor loop took %0.3f seconds\n', numPhotos

numPlotsAcross = 6;
[figHandle, axesHandles] = createFigure(numPhotos, numPlotsAcross);
figure(figHandle);
for idx = 1:numPhotos
    plotDataAndAttribution(axesHandles(idx), photos{idx}, ...
        photoInfo{idx}, usernames{idx}, idx);
end
```

Explicit Parallelism: Independent Tasks or Iterations

Simple programming constructs using `parfor`, `parfeval`

Examples: parameter sweeps, Monte Carlo simulations

No dependencies or communications between tasks

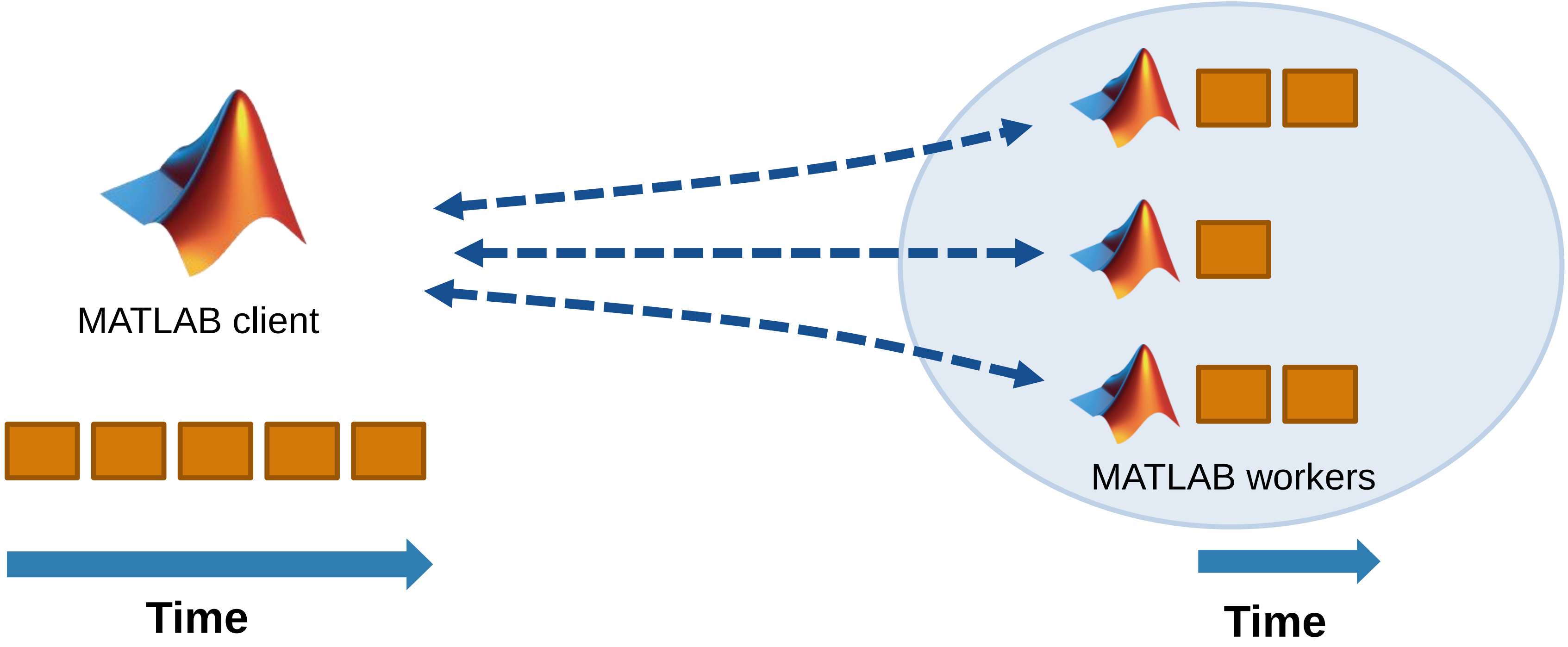


```
a = zeros(10, 1)
parfor i = 1:10
   a(i) = i;
end
a
```

Independent Tasks or Iterations

Simple programming constructs using `parfor`, `parfeval`

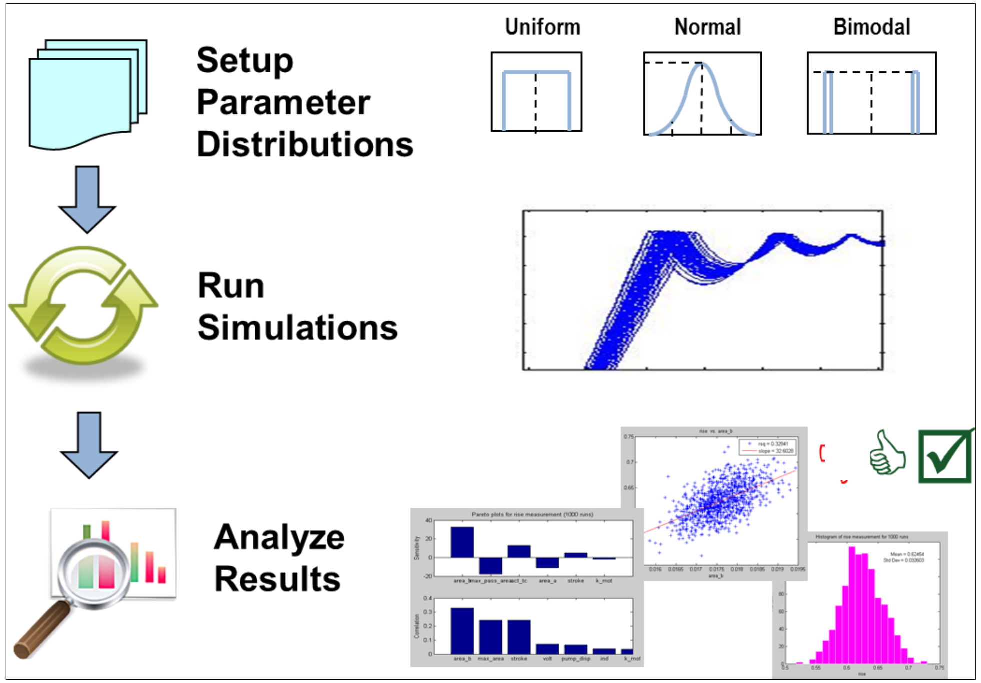
- Examples: parameter sweeps, Monte Carlo simulations
- No dependencies or communications between tasks



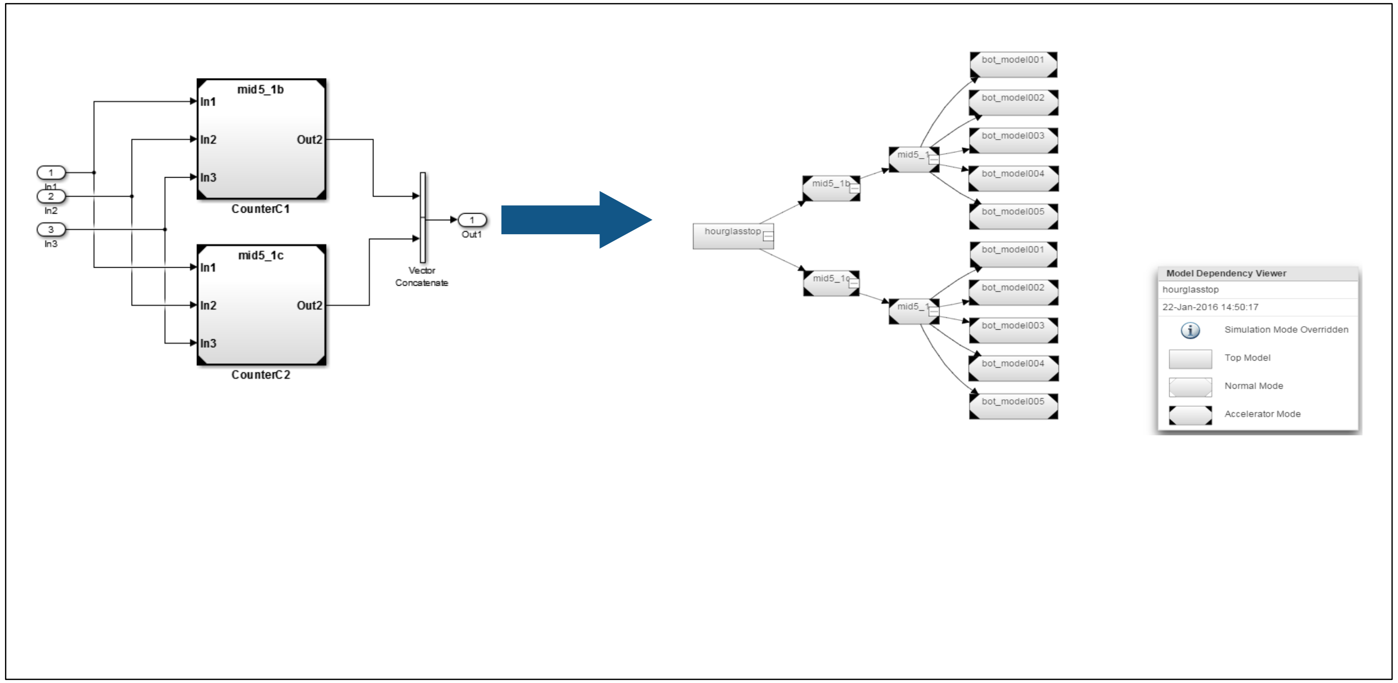
Leverage Parallel Computing for Simulink

- Reduce the total amount of time it takes to…

•Run multiple independent simulations (E.g. Parameter sweeps, Monte Carlo Analysis)

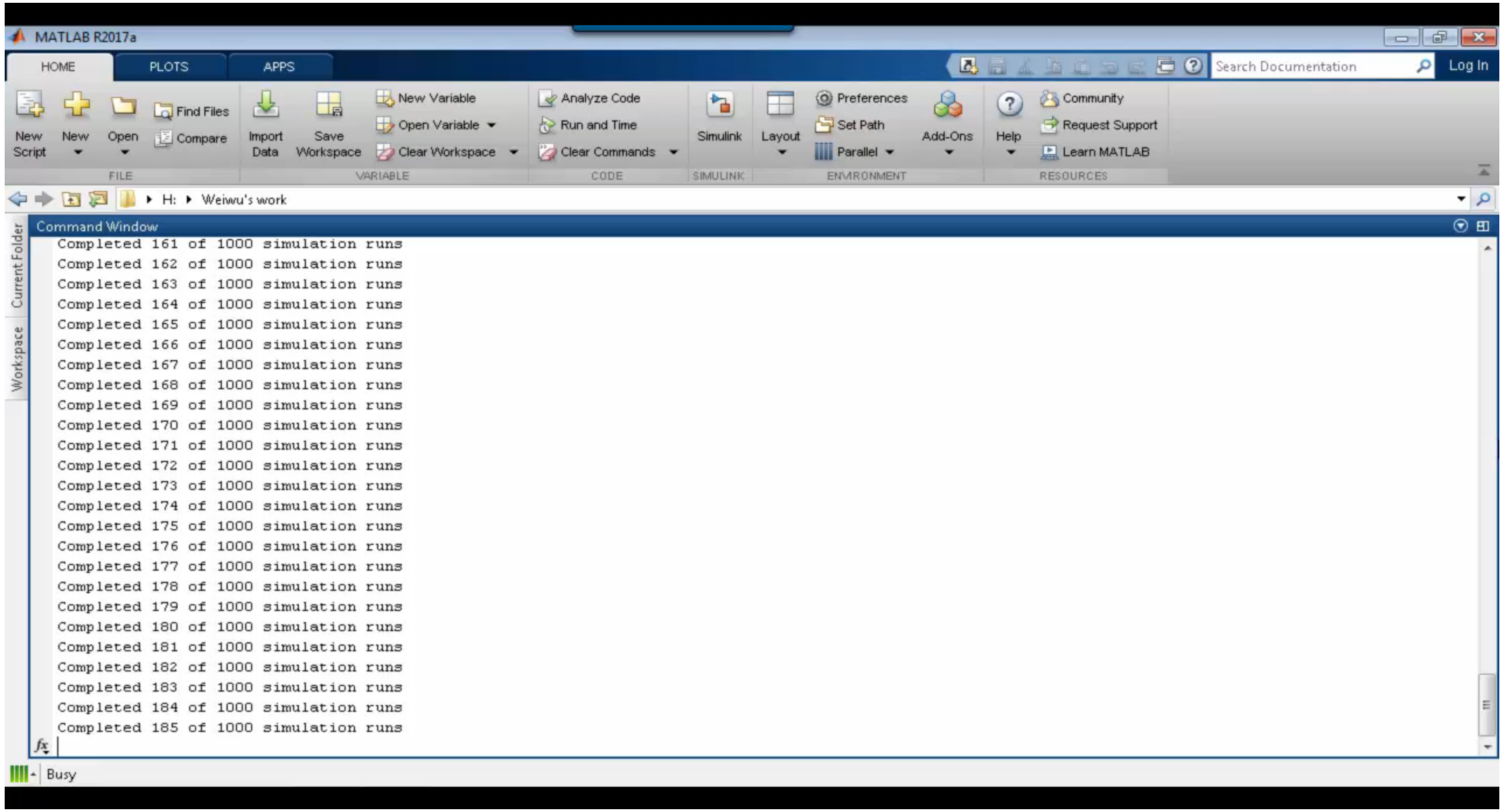


Update models containing large model reference hierarchies

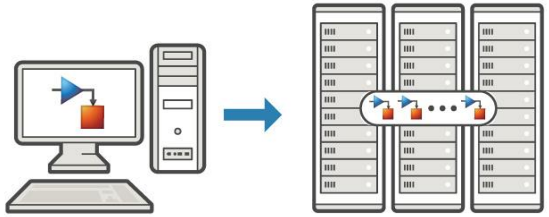


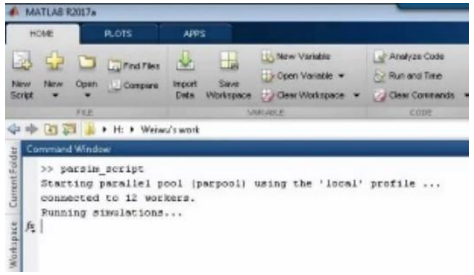
Parallel Simulations using Simulink and Parsim

Directly run multiple parallel simulations from the parsim command



```
Completed 161 of 1000 simulation runs
Completed 162 of 1000 simulation runs
Completed 163 of 1000 simulation runs
Completed 164 of 1000 simulation runs
Completed 165 of 1000 simulation runs
Completed 166 of 1000 simulation runs
Completed 167 of 1000 simulation runs
Completed 168 of 1000 simulation runs
Completed 169 of 1000 simulation runs
Completed 170 of 1000 simulation runs
Completed 171 of 1000 simulation runs
Completed 172 of 1000 simulation runs
Completed 173 of 1000 simulation runs
Completed 174 of 1000 simulation runs
Completed 175 of 1000 simulation runs
Completed 176 of 1000 simulation runs
Completed 177 of 1000 simulation runs
Completed 178 of 1000 simulation runs
Completed 179 of 1000 simulation runs
Completed 180 of 1000 simulation runs
Completed 181 of 1000 simulation runs
Completed 182 of 1000 simulation runs
Completed 183 of 1000 simulation runs
Completed 184 of 1000 simulation runs
Completed 185 of 1000 simulation runs
```



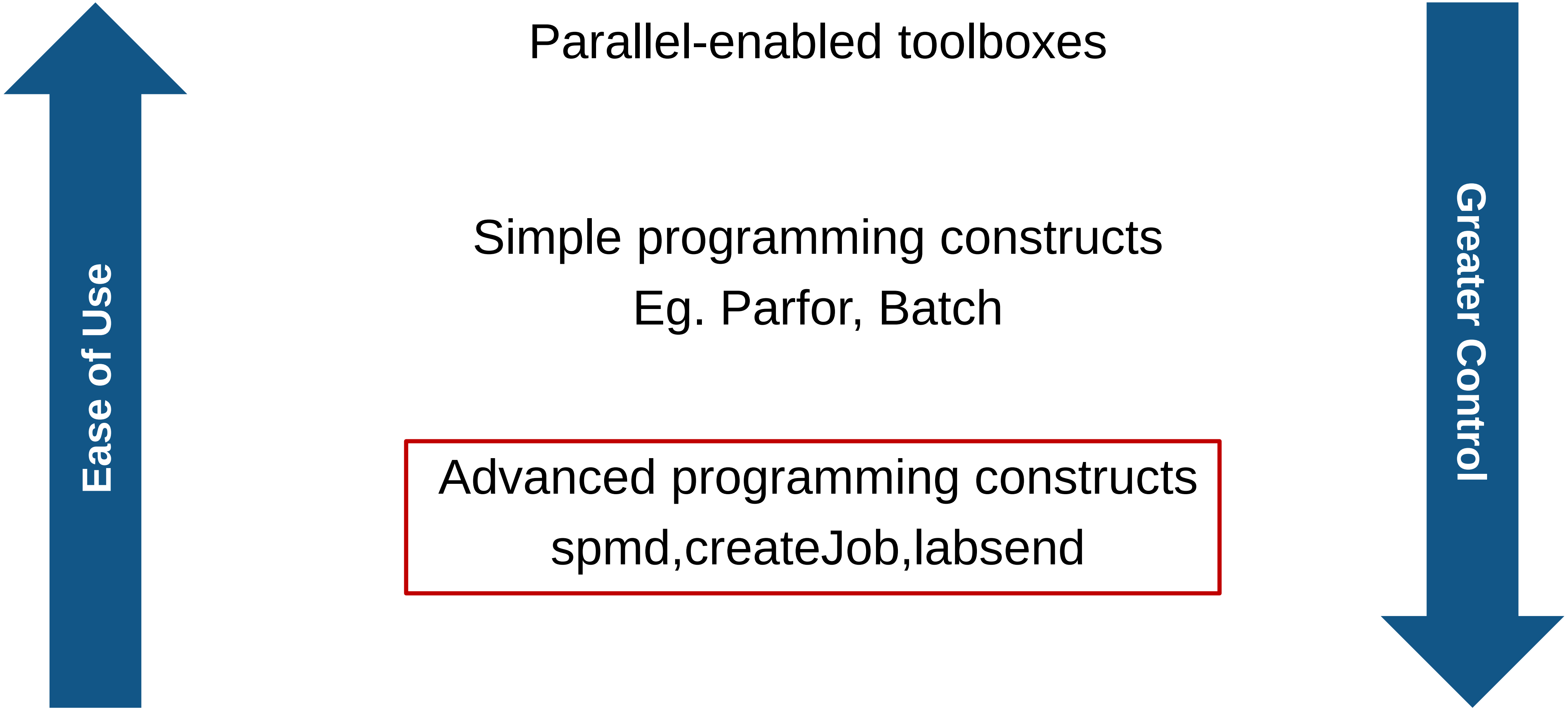


```
>> parsim_script
Starting parallel pool (parpool) using the 'local' profile ...
connected to 12 workers.
Running simulations...
```

We manage the parallel setup so customers can focus on their simulations

- Enables customers to easily use Simulink with parallel computing
- Simplifies customers' large simulation runs and improves their productivity

Programming Parallel Applications – Programming Constructs



Ease of Use

Parallel-enabled toolboxes

Simple programming constructs

Eg. Parfor, Batch

Advanced programming constructs

spmd,createJob,labsend

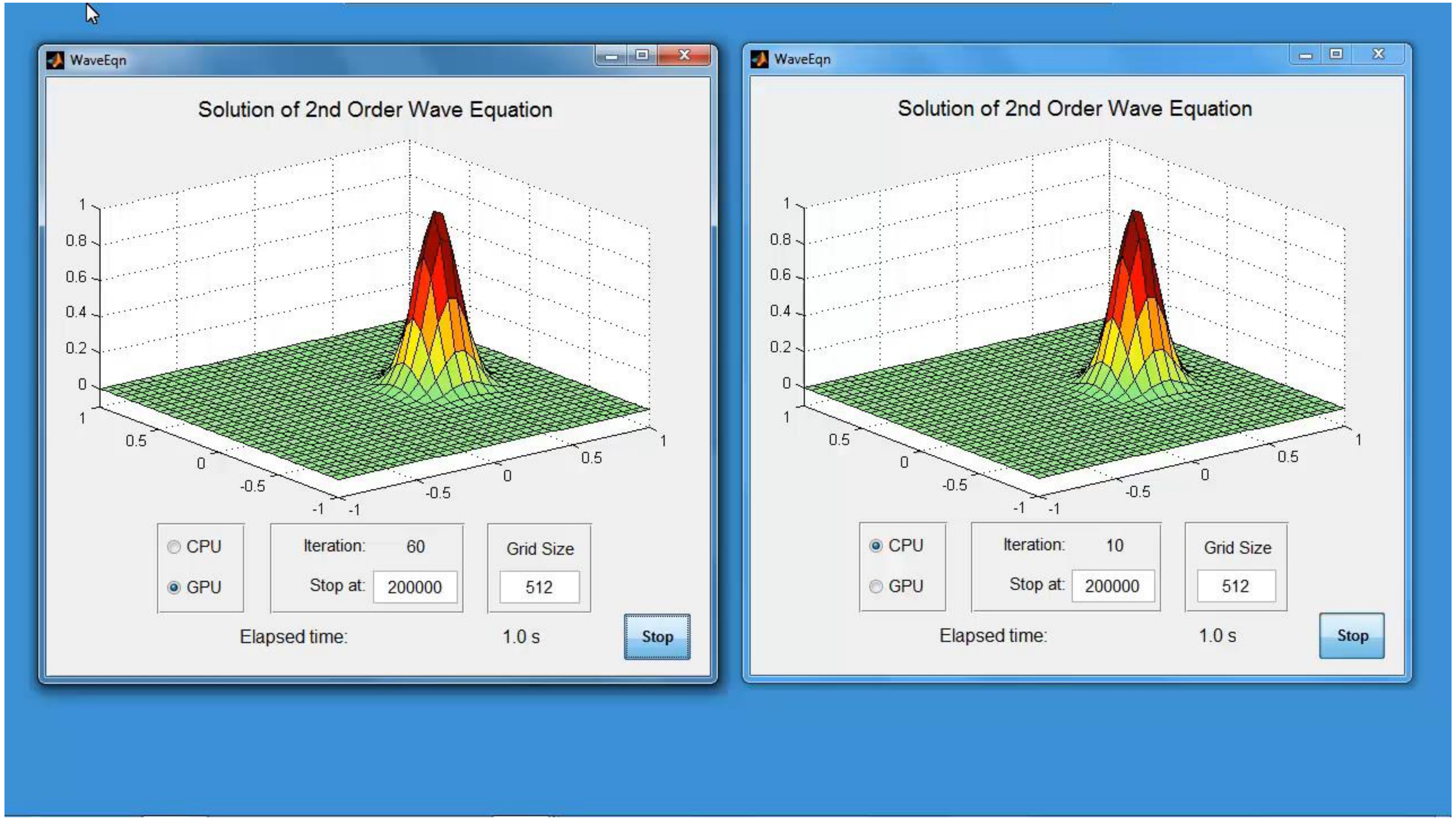
Greater Control

Agenda

- Parallel computing in MATLAB and Simulink
- Accelerate applications with NVIDIA GPUs
- Scaling to clusters and clouds
- Summary

Example 3: Solving 2D Wave Equation

GPU Computing



○ CPU
● GPU

Iteration: 60

Stop at: 200000

Grid Size: 512

Elapsed time: 1.0 s

● CPU
○ GPU

Iteration: 10

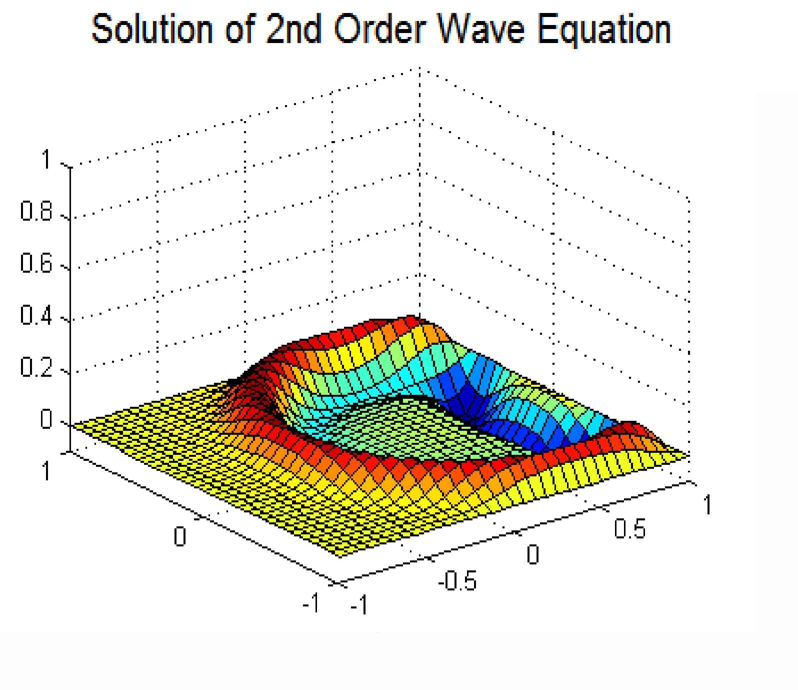
Stop at: 200000

Grid Size: 512

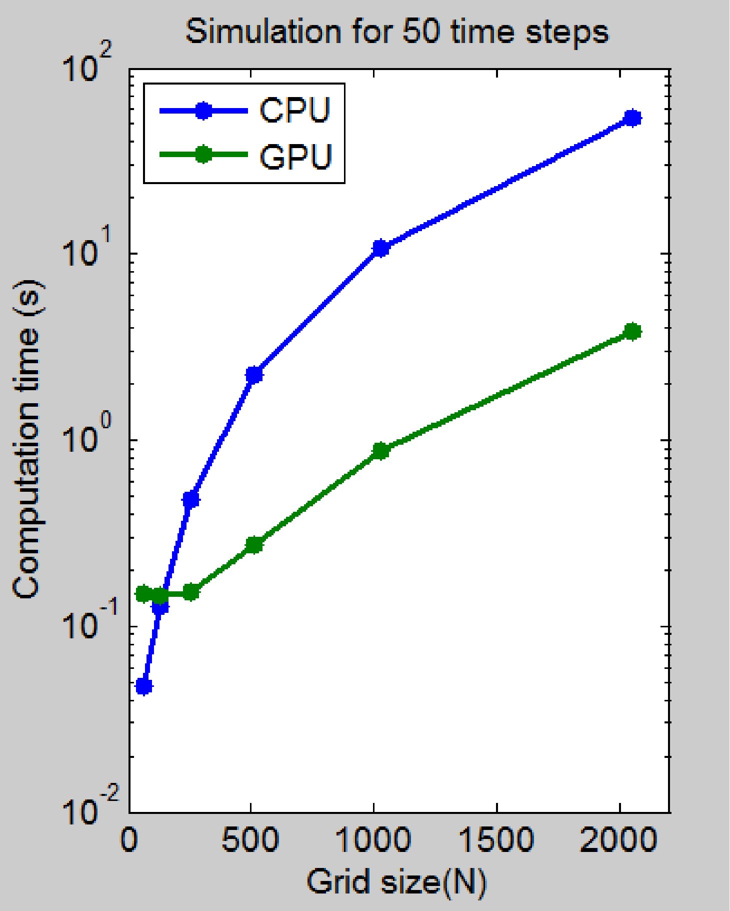
Elapsed time: 1.0 s

Example: Solving 2D Wave Equation

GPU Computing



$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$$

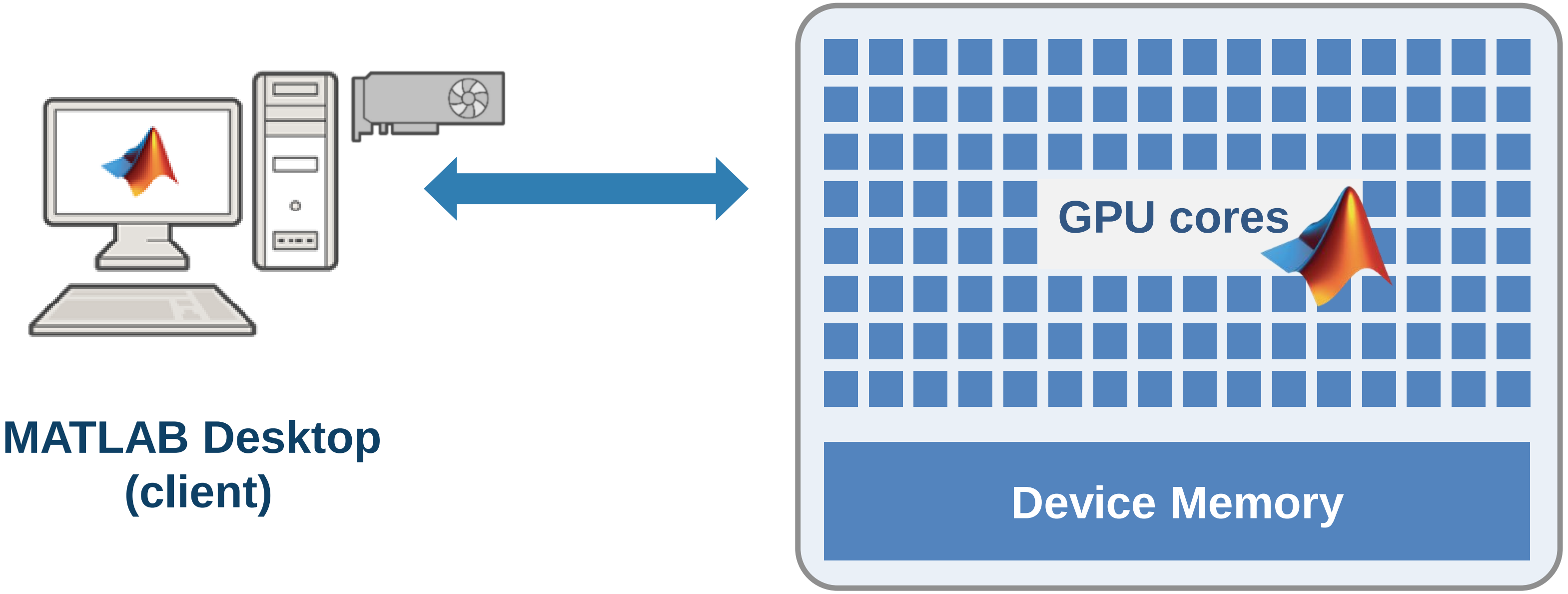


Intel Xeon Processor W3690 (3.47GHz), NVIDIA Tesla K20 GPU

Parallel Computing Paradigm

Going Parallel: GPUs

Using NVIDIA GPUs

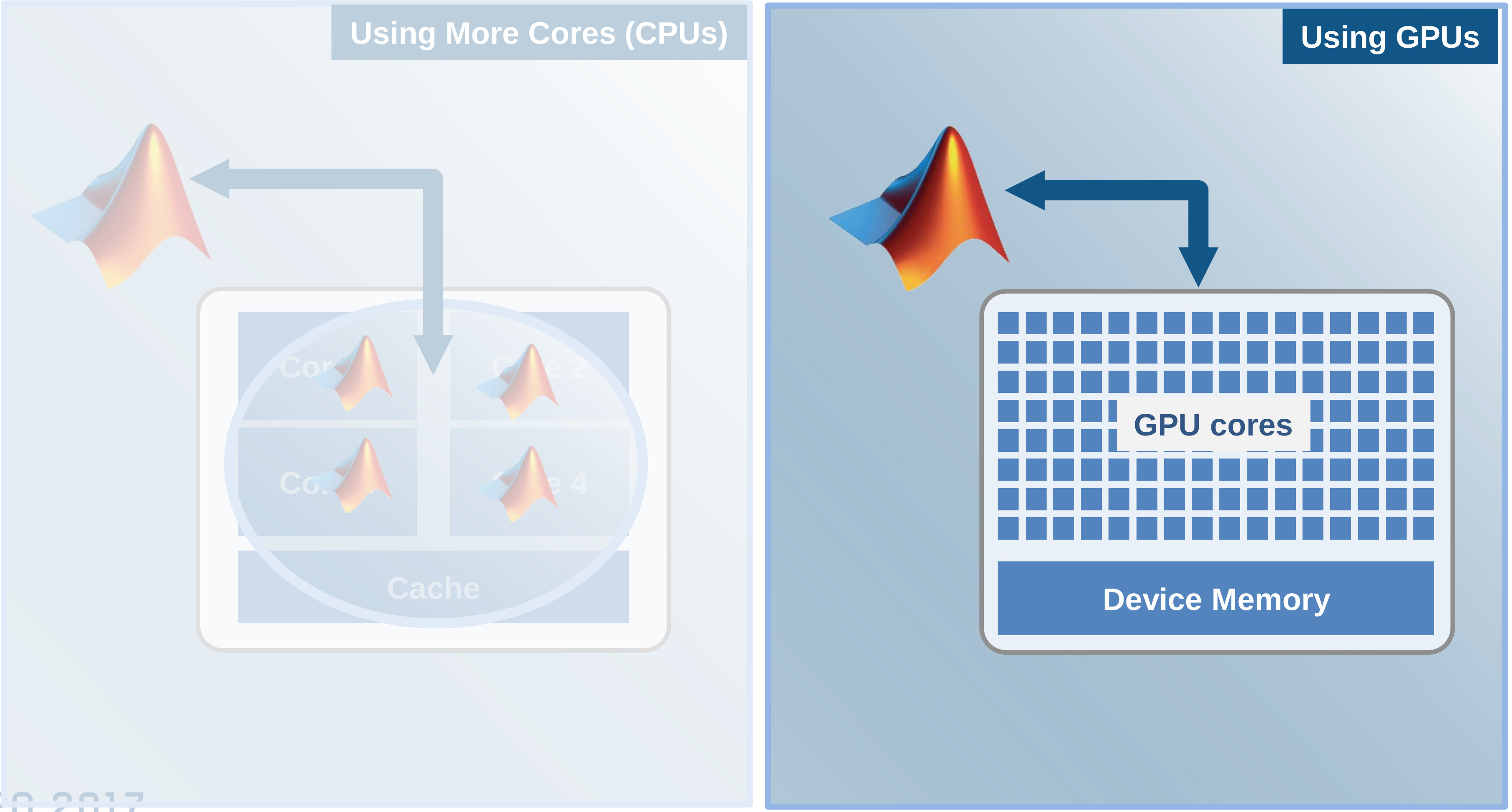


MATLAB Desktop (client)

GPU cores

Device Memory

Performance Gain with More Hardware



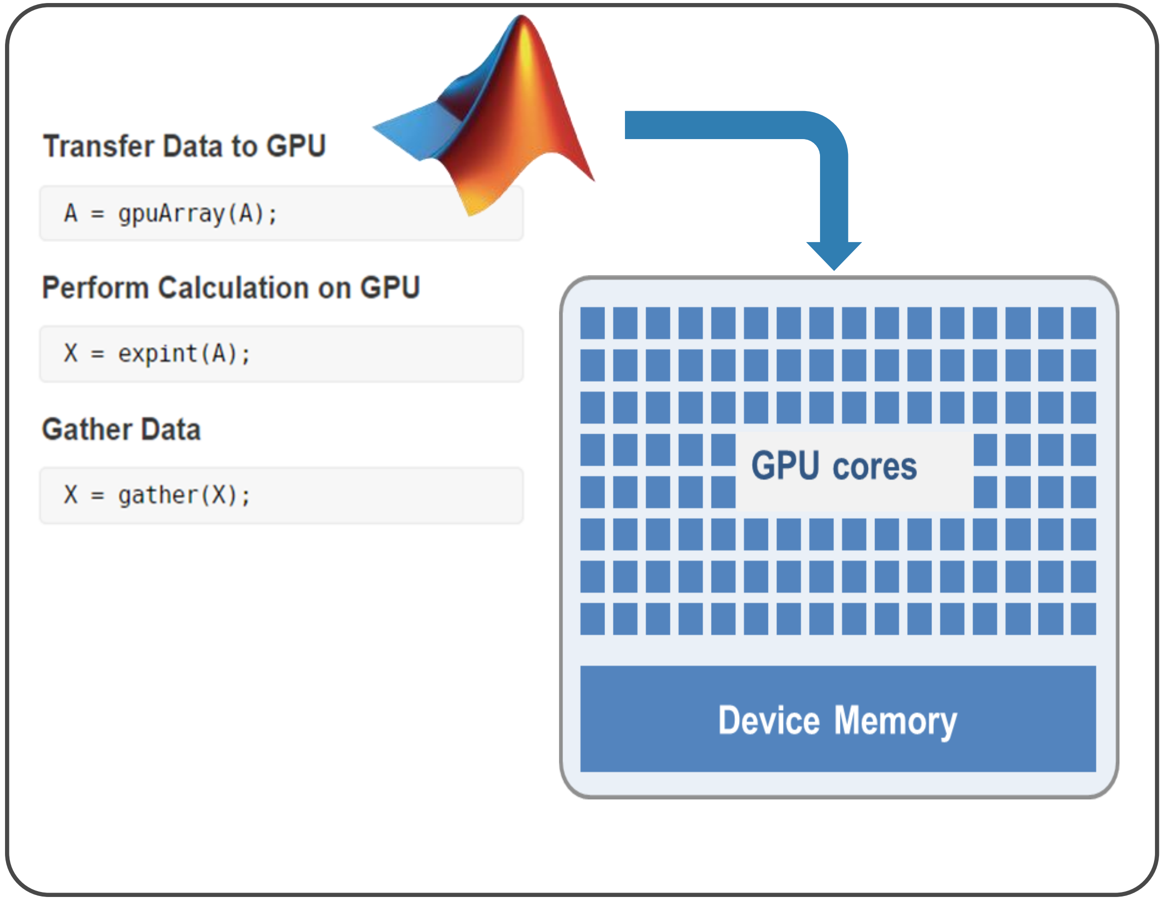
Speed-up using NVIDIA GPUs

- Ideal Problems
  - Massively Parallel and/or Vectorized operations
  - Computationally Intensive
  - Algorithm consists of supported functions

- 300+ GPU-enabled MATLAB functions

- Additional GPU-enabled Toolboxes
  - Neural Networks
  - Image Processing
  - Communications
  - Signal Processing

..... Learn More



Transfer Data to GPU

```
A = gpuArray(A);
```

Perform Calculation on GPU

```
X = expint(A);
```

Gather Data

```
X = gather(X);
```

NVIDIA. TESLA

Signal Processing – Acoustic Data Analysis

NASA Langley Research

Goal: Accelerate the analysis of sound recordings from wind tunnel tests of aircraft components

Learn More

Challenges

- Legacy code took 40 mins to analyze single wind tunnel test data
- Reduce processing time to make on-the-fly decisions and identify hardware problems

Why GPU Computing

- Computations completed 40 times faster.

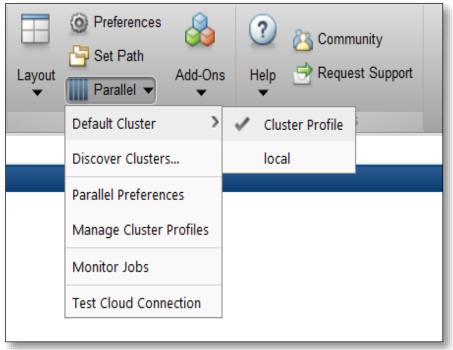
> *"Many operations we perform, including FFTs and matrix multiplication, are **GPU-enabled MATLAB functions** .Once we developed the initial MATLAB code for CPU execution, it took 30 minutes to get our algorithm working on the GPU—**no low-level CUDA programming** was needed. The addition of GPU computing with Parallel Computing Toolbox cut it to **under a minute**, with most of that time spent on data transfer"*

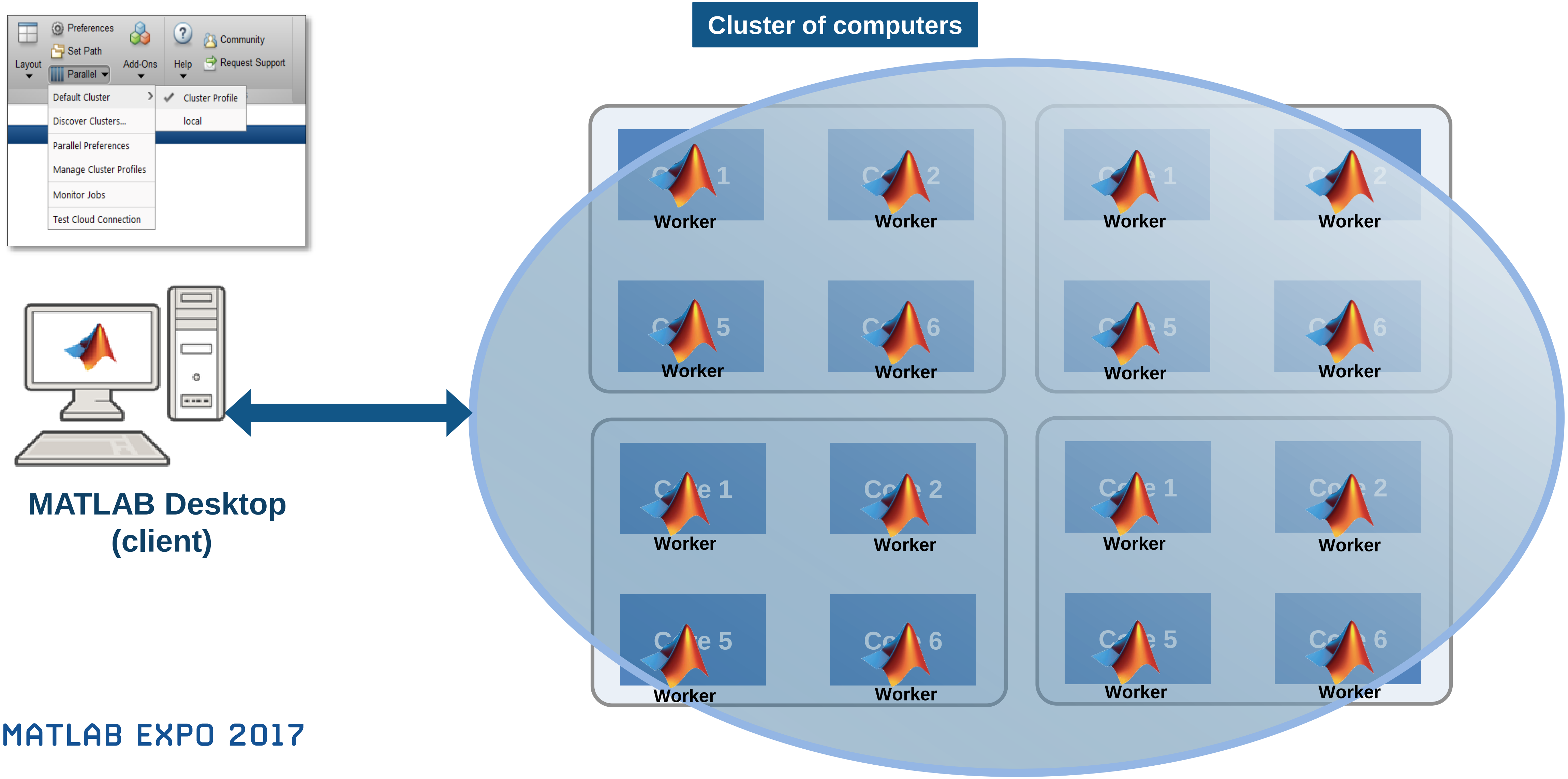
Christopher Bahr, NASA

Agenda

- Parallel computing in MATLAB and Simulink
- Accelerate applications with NVIDIA GPUs
- **Scaling to clusters and clouds**
- Summary

Scaling to a computer cluster

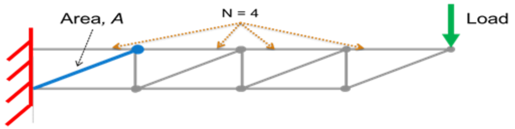




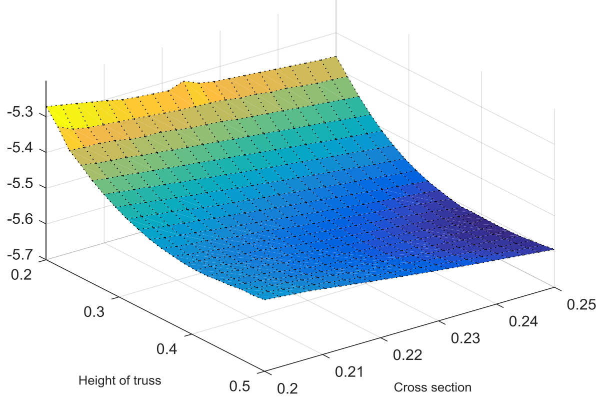
Why parallel computing matters

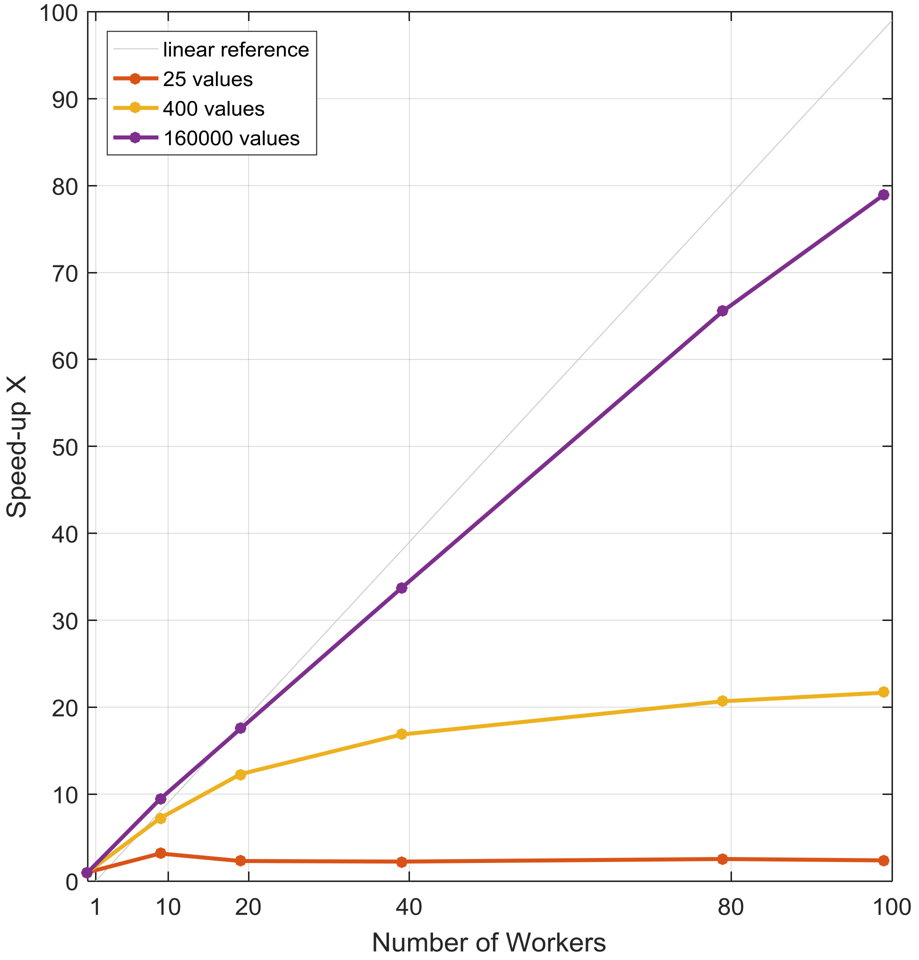
Scaling case study with a compute cluster

$$M\ddot{x} + C\dot{x} + Kx = F$$



Log of Maximum Y Deflection (12 segments)

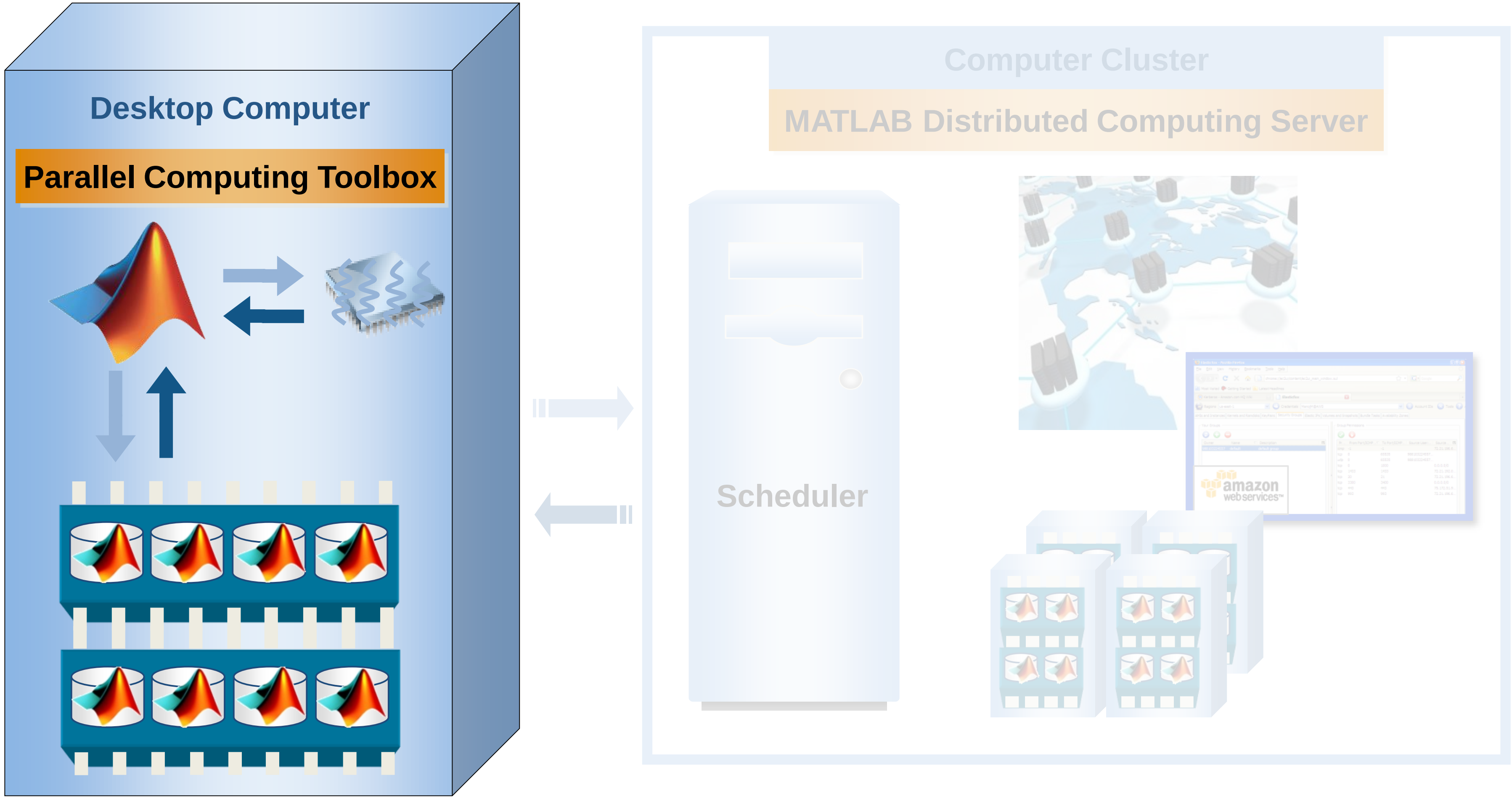




Workers in pool	Compute time (minutes)		
	160e3 values	400 values	25 values
1	140	0.38	0.03
10	15	0.05	0.01
20	8.0	0.03	0.01
40	4.2	0.02	0.01
80	2.1	0.02	0.01
100	1.8	0.02	0.01

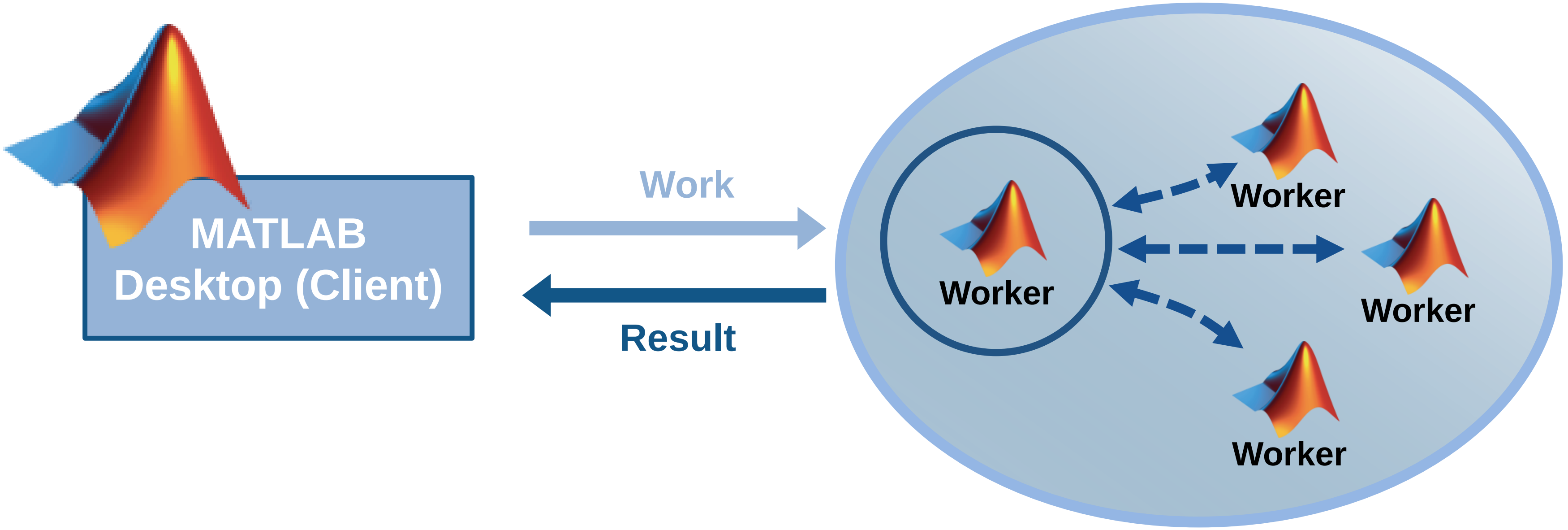
Processor: Intel Xeon E5-class v2
16 physical cores per node
MATLAB R2016a

Scale Up to Clusters, Grids and Clouds


Desktop Computer
Parallel Computing Toolbox
Computer Cluster
MATLAB Distributed Computing Server
Scheduler
amazon webservices

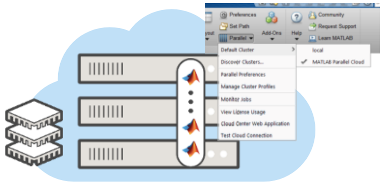
MATLAB EXPO 2017

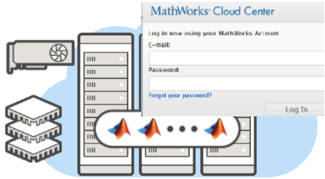
Offload and Scale Computations with `batch`



```
batch(…, 'Pool',…)
```

Scale your applications beyond the desktop





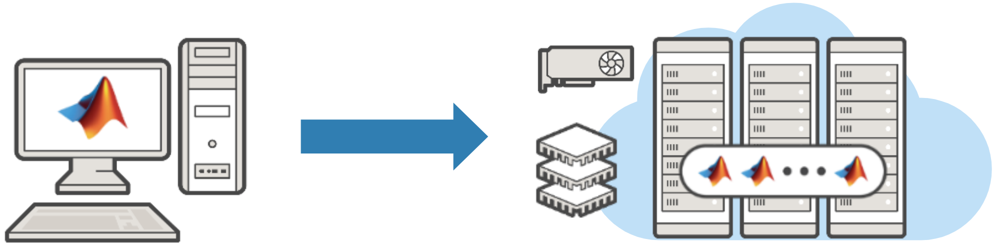
Option	Parallel Computing Toolbox	MATLAB Parallel Cloud	MATLAB Distributed Computing Server *for Amazon EC2*	MATLAB Distributed Computing Server *for Custom Cloud*	MATLAB Distributed Computing Server
Description	Explicit desktop scaling	Single-user, basic scaling to cloud	Scale to EC2 with some customization	Scale to custom cloud	Scale to clusters
Maximum workers	No limit	16	256	No limit	No limit
Hardware	Desktop	MathWorks Compute Cloud	Amazon EC2	Amazon EC2, Microsoft Azure, Others	Any
Availability	Worldwide	United States and Canada	United States, Canada and other select countries in Europe	Worldwide	Worldwide

Learn More: Parallel Computing on the Cloud

Summary and Takeaways

- **Speed up your MATLAB and Simulink applications without being an expert**
  - Reduce computation time by using more cores or accessing Graphical Processing Units
- **Leverage the Parallel Computing Toolbox to**
  - Reduce Computation Time: `parfor, gpuArray, parsim`
  - Offload and Scale Computations: `batch`
- **Speed up Big Data Analytics**
  - Using `datastore` and `tall` or `distributed` arrays
- **Develop parallel applications on the desktop and scale to clusters seamlessly**

What’s new in 16b and 17a?



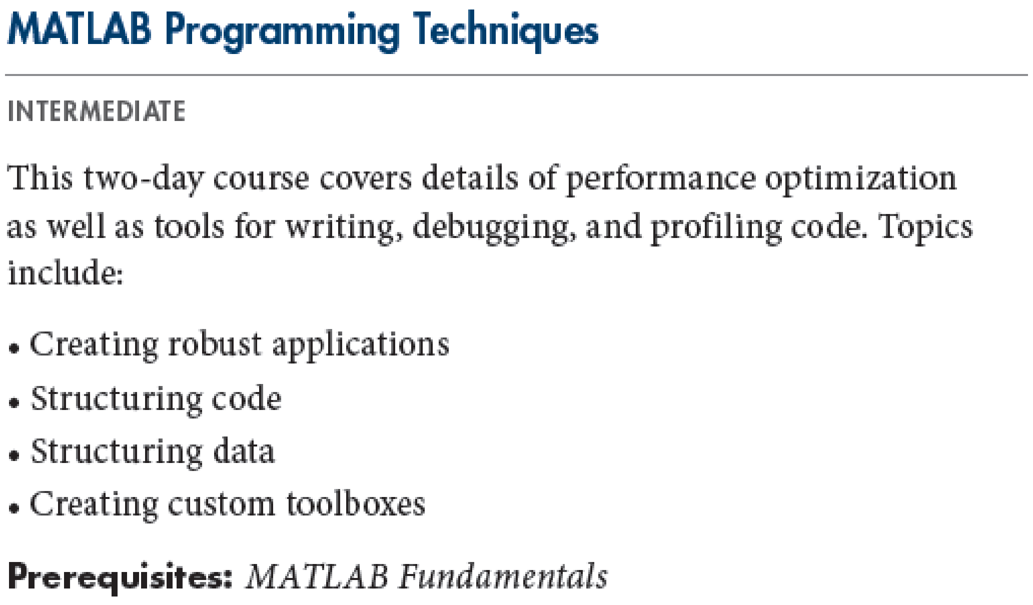
R2016b

- `tall` array support for big data
- Measure data sent to workers using `ticBytes` and `tocBytes`
- Cloud offerings with K80-equipped GPUs

R2017a

- Simplified parallel Simulink simulations using `parsim`
- Send data to client using `DataQueue` and `PollableDataQueue`
- Train a single deep learning network with multiple CPUs or multiple GPUs

MathWorks Training Offerings



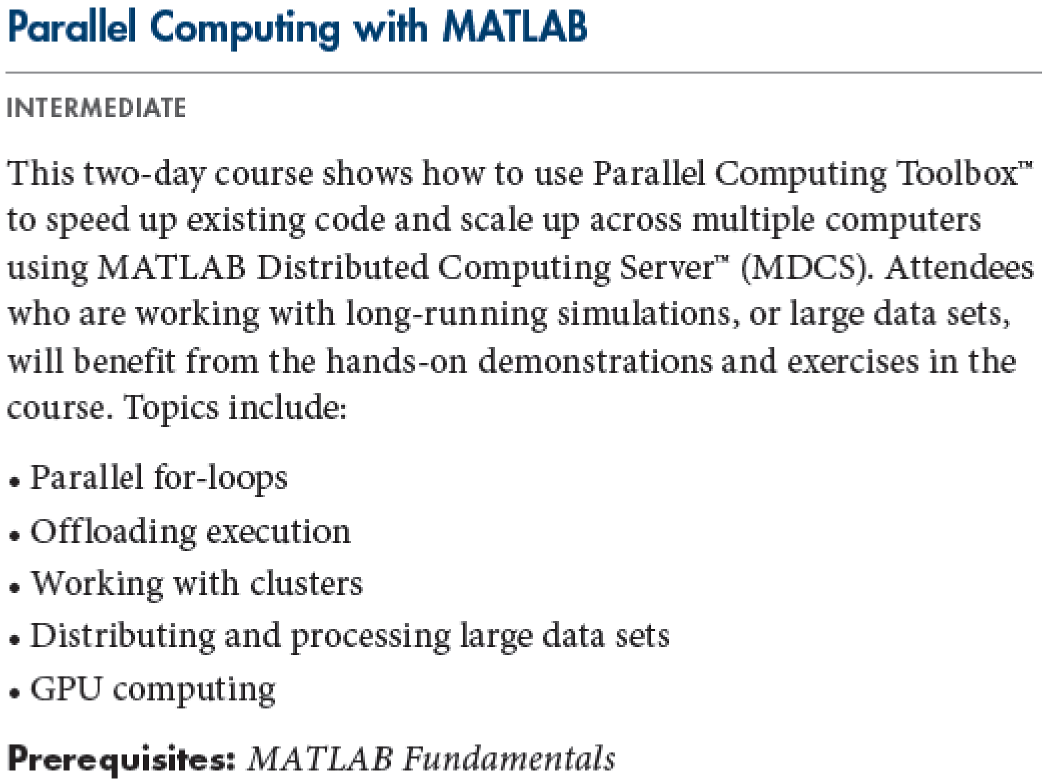
MATLAB Programming Techniques

INTERMEDIATE

This two-day course covers details of performance optimization as well as tools for writing, debugging, and profiling code. Topics include:

- Creating robust applications
- Structuring code
- Structuring data
- Creating custom toolboxes

Prerequisites: *MATLAB Fundamentals*



Parallel Computing with MATLAB

INTERMEDIATE

This two-day course shows how to use Parallel Computing Toolbox™ to speed up existing code and scale up across multiple computers using MATLAB Distributed Computing Server™ (MDCS). Attendees who are working with long-running simulations, or large data sets, will benefit from the hands-on demonstrations and exercises in the course. Topics include:

- Parallel for-loops
- Offloading execution
- Working with clusters
- Distributing and processing large data sets
- GPU computing

Prerequisites: *MATLAB Fundamentals*

http://www.mathworks.com/services/training/

Learn Further

- https://www.mathworks.com/solutions/parallel-computing.html
- https://www.mathworks.com/help/distcomp/
- https://www.mathworks.com/help/distcomp/examples.html

MathWorks®

Accelerating the pace of engineering and science

Speaker Details

Email: Alka.Nair@mathworks.in

Contact MathWorks India

Products/Training Enquiry Booth

Call: 080-6632-6000

Email: info@mathworks.in

Your feedback is valued.

Please complete the feedback form provided to you.

MathWorks®