

# Tackling Big Data with MATLAB

**Adam Filion**  
**Application Engineer**  
**MathWorks, Inc.**

# Challenges of Big Data

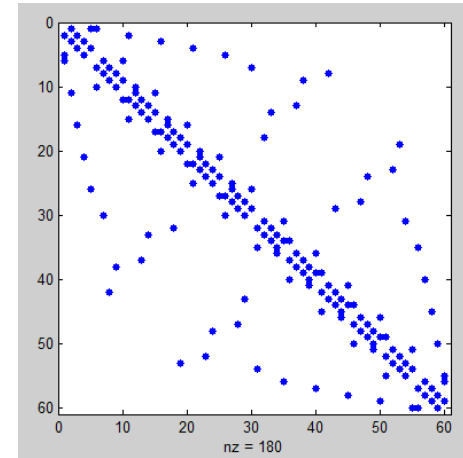
*“Any collection of data sets so large and complex that it becomes difficult to process using ... traditional data processing applications.” (Wikipedia)*

- Getting started
- Rapid data exploration
- Development of scalable algorithms
- Ease of deployment

# MATLAB and Memory

## Best Practices for Memory Usage

- Use 64-bit MATLAB whenever possible
- Use the appropriate data storage
  - Use only the precision your need
  - Sparse Matrices
  - Categorical Arrays
  - Be aware of overhead of cells and structures
- Minimize Data Copies
  - Lazy copy
  - Nested functions
  - In place operations
  - If using objects, consider handle classes



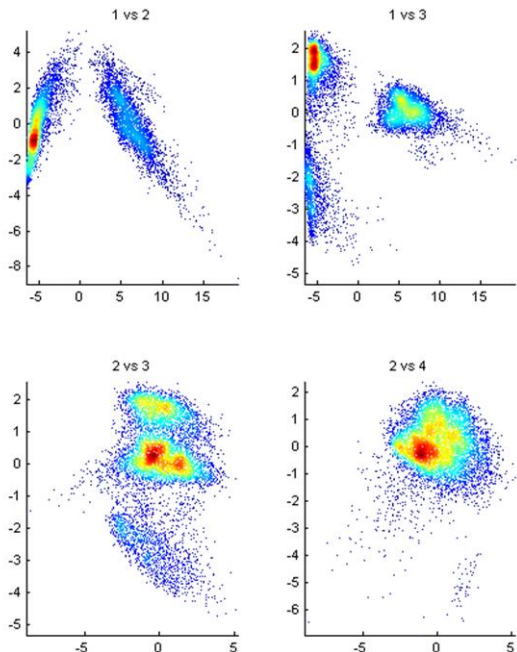
```
function primaryFcn
    x = 1;
    y = x;
    nestedFcn

    function nestedFcn
        x = x + 1;
    end
end
```

# Big Data Capabilities in MATLAB

## Memory and Data Access

- 64-bit processors
- Memory Mapped Variables
- Disk Variables
- Databases
- **Datastores R2014b**



## Programming Constructs

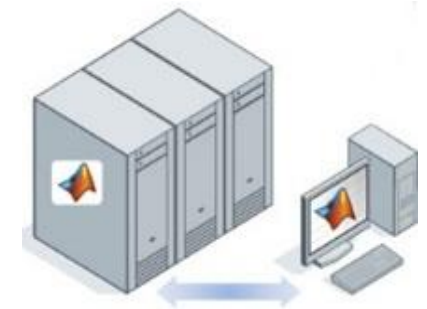
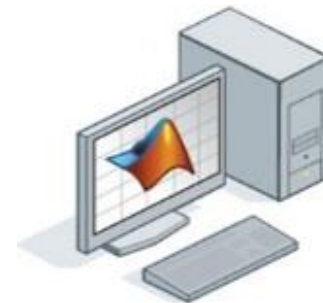
- Streaming
- Block Processing
- **Parallel-for loops**
- GPU Arrays
- SPMD and Distributed Arrays
- **MapReduce R2014b**

## Platforms

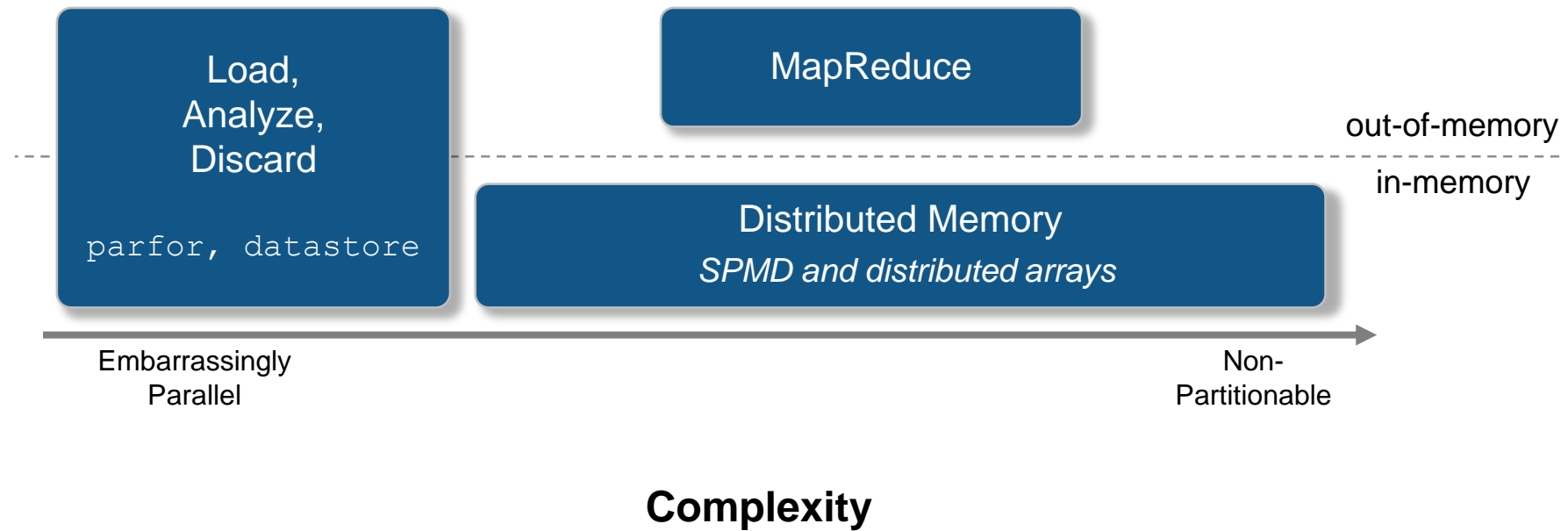
- Desktop (Multicore, GPU)
- Clusters
- Cloud Computing (MDCS on EC2)
- **Hadoop R2014b**

# Considerations for Choosing an Approach

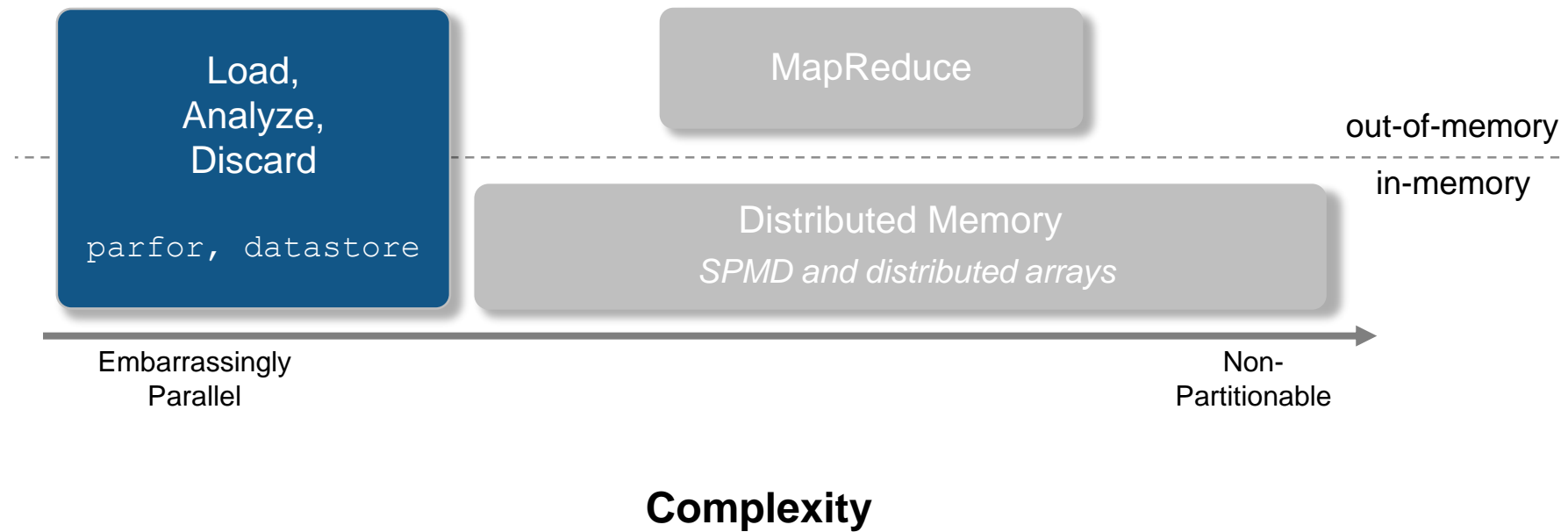
- Data characteristics
  - Size, type and location of your data
- Compute platform
  - Single desktop machine or cluster
- Analysis Characteristics
  - Embarrassingly Parallel
  - Analyze sub-segments of data and aggregate results
  - Operate on entire dataset



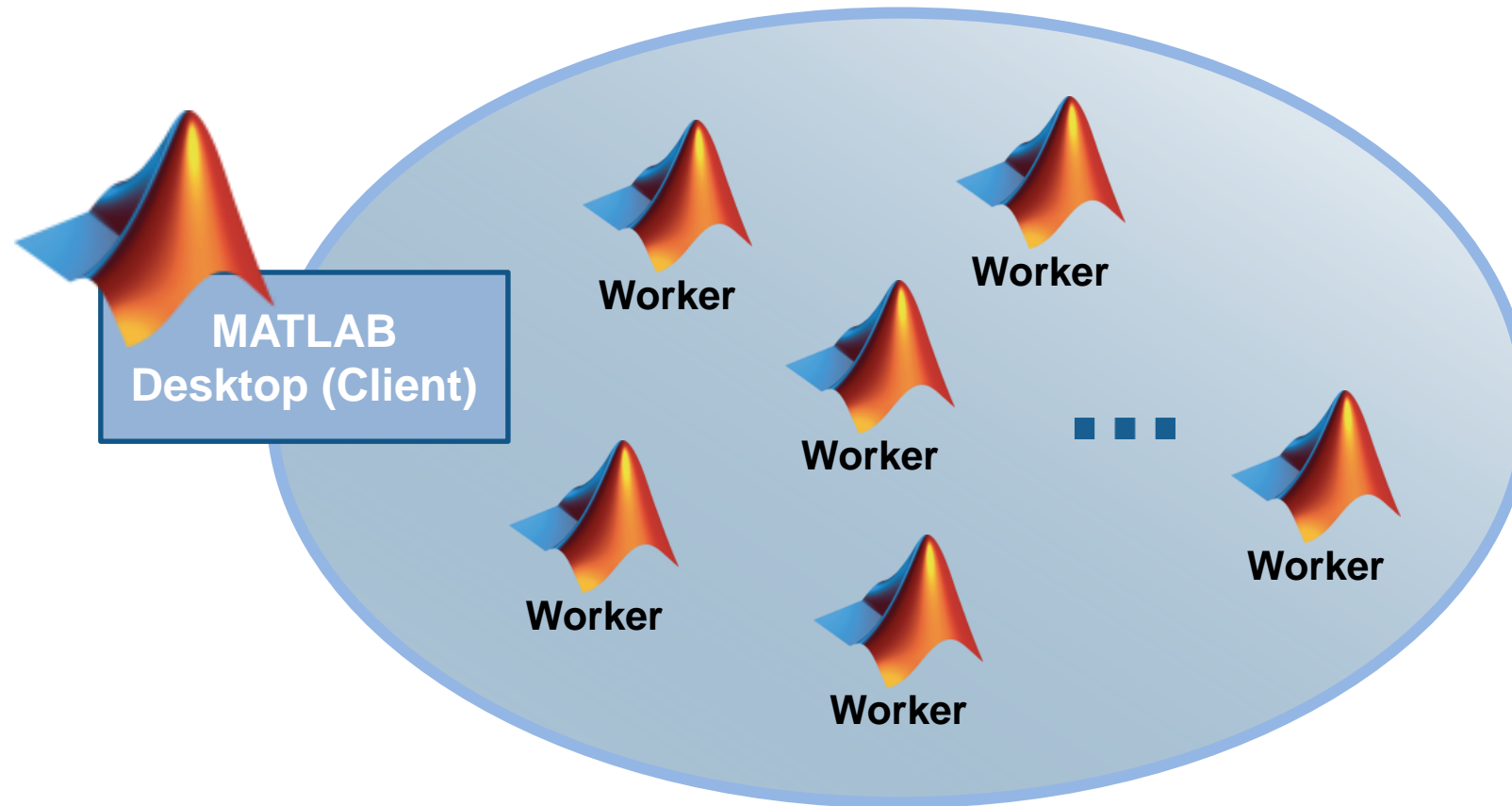
# Techniques for Big Data in MATLAB



# Techniques for Big Data in MATLAB

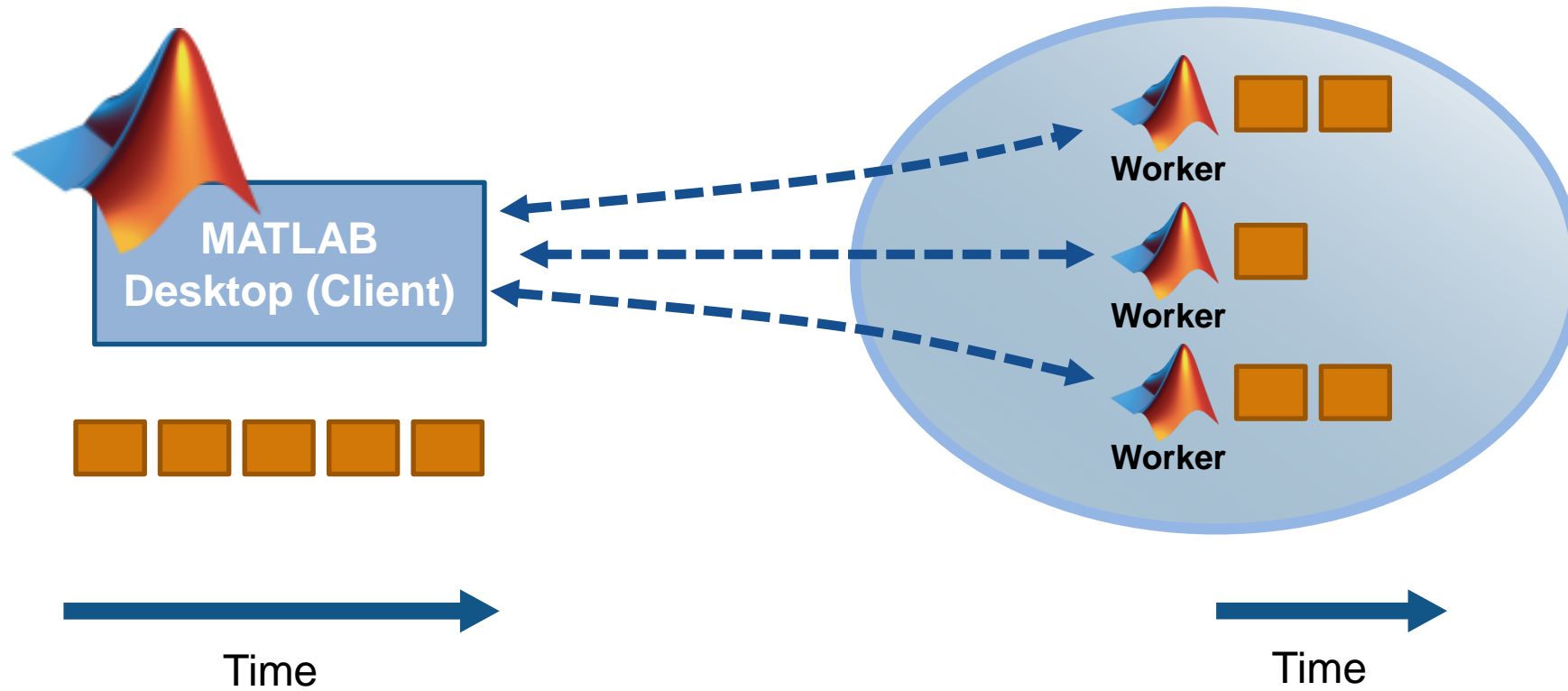


# Parallel Computing with MATLAB





# Speed up Using Simultaneous Workers



# Demo: Determining Land Use

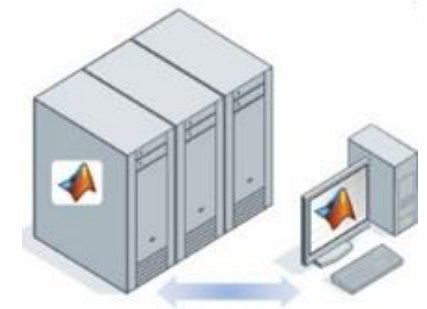
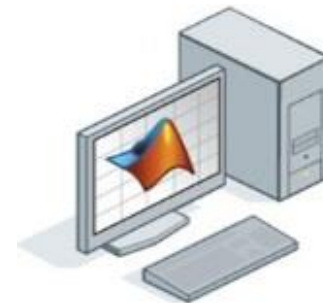
## Using Parallel for-loops (`parfor`)

- Data
  - Arial images of agriculture land
  - 24 TIF files
- Analysis
  - Find and measure irrigation fields
  - Determine which irrigation circles are in use (by color)
  - Calculate area under irrigation

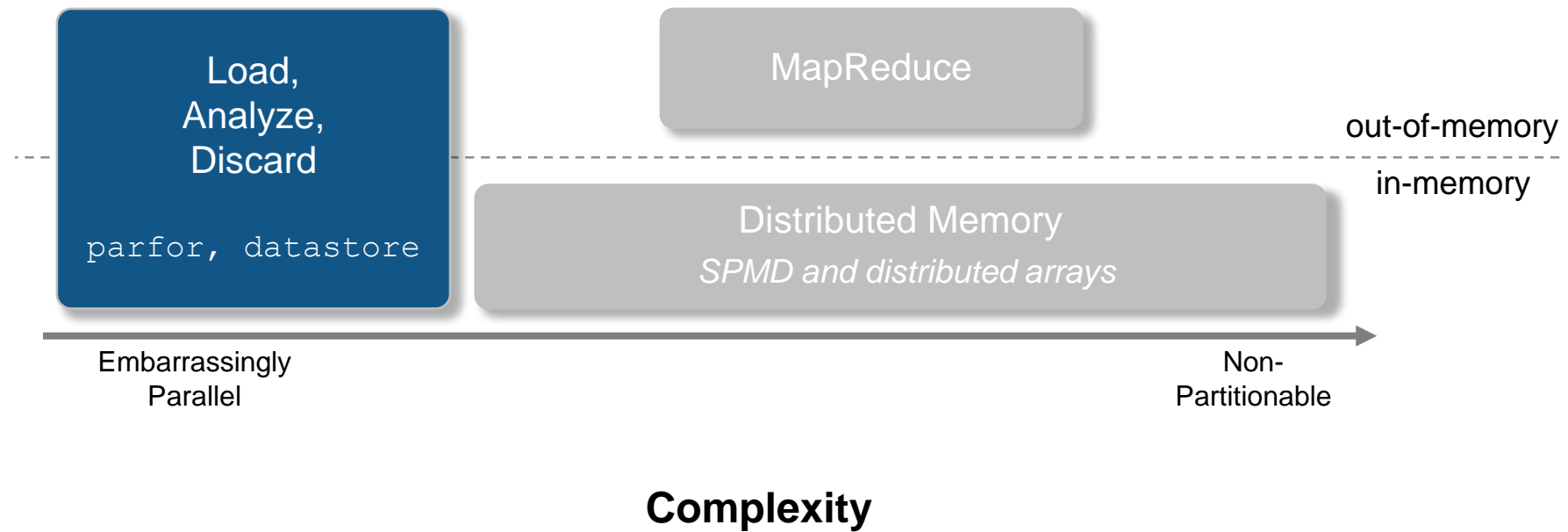


# When to Use `parfor`

- Data Characteristics
  - Can be of any format (i.e. text, images) as long as it can be broken into pieces
  - The data for each iteration must fit in memory
- Compute Platform
  - Desktop (Parallel Computing Toolbox)
  - Cluster (MATLAB Distributed Computing Server)
- Analysis Characteristics
  - Each iteration of your loop must be independent

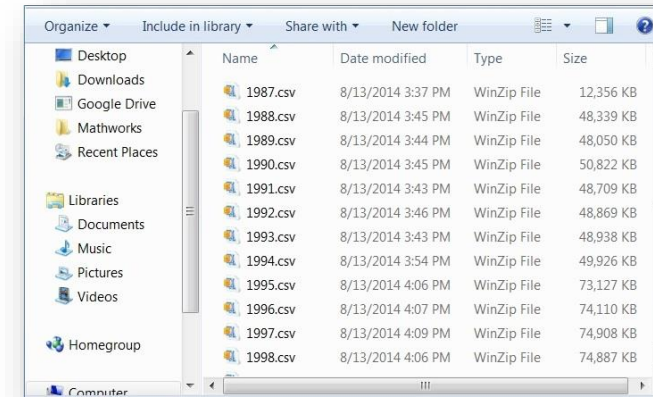


# Techniques for Big Data in MATLAB



# Access Big Data datastore

- Easily specify data set
  - Single text file or collection of text files
  - Data stored on HDFS
- Preview data structure and format
- Select data to import using column names
- Incrementally read subsets of the data



```
>> preview(ds)
ans =
   Year   Month  DayofMonth  DayOfWeek
   _____  _____  _____  _____
   1987     10     21           3
   1987     10     26           1
   1987     10     23           5
   1987     10     23           5
```

```
airdata = datastore('*.*csv');
airdata.SelectedVariables = {'Distance', 'ArrDelay'};

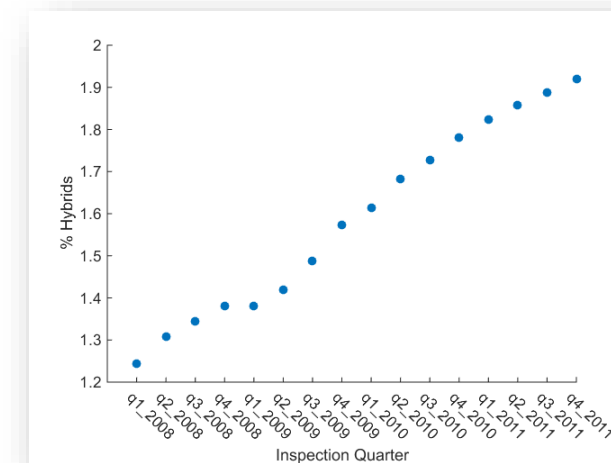
data = read(airdata);
```

# Demo: Vehicle Registry Analysis

## Using a DataStore

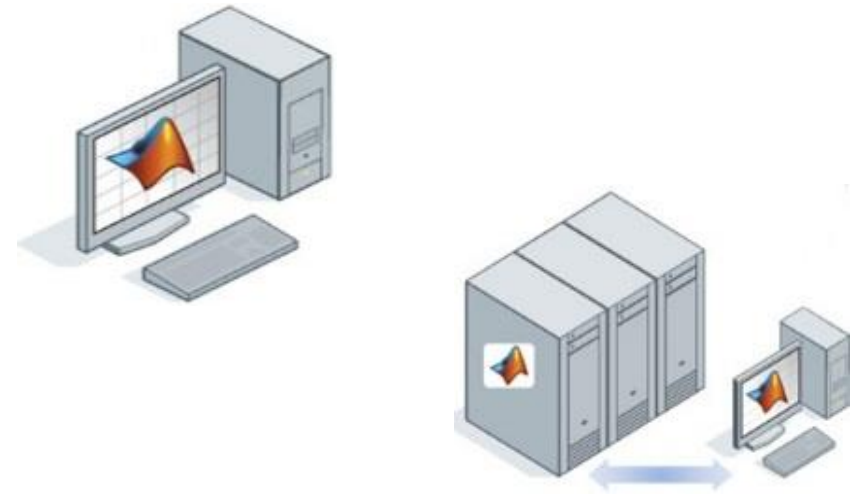
- Data
  - Massachusetts Vehicle Registration Data from 2008-2011
  - 16M records, 45 fields
- Analysis
  - Examine hybrid adoptions
  - Calculate % of hybrids registered by quarter
  - Fit growth to predict further adoption

muni_id	veh_zip	insp_year	model_year	make
325	1089	2011	2008	'Hyundai'
325	1089	2009	2008	'Hyundai'
288	1776	2011	2008	'Acura'
288	1776	2008	2008	'Acura'
145	2364	2011	2005	'Chevrolet'
325	1089	2010	2008	'Hyundai'
325	1089	2011	2008	'Hyundai'
288	1776	2009	2008	'Acura'



# When to Use datastore

- Data Characteristics
  - Text data in files or stored in the Hadoop Distributed File System (HDFS)
- Compute Platform
  - Desktop
- Analysis Characteristics
  - Supports Load, Analyze, Discard workflows
  - Incrementally read chunks of data, process within a **while** loop

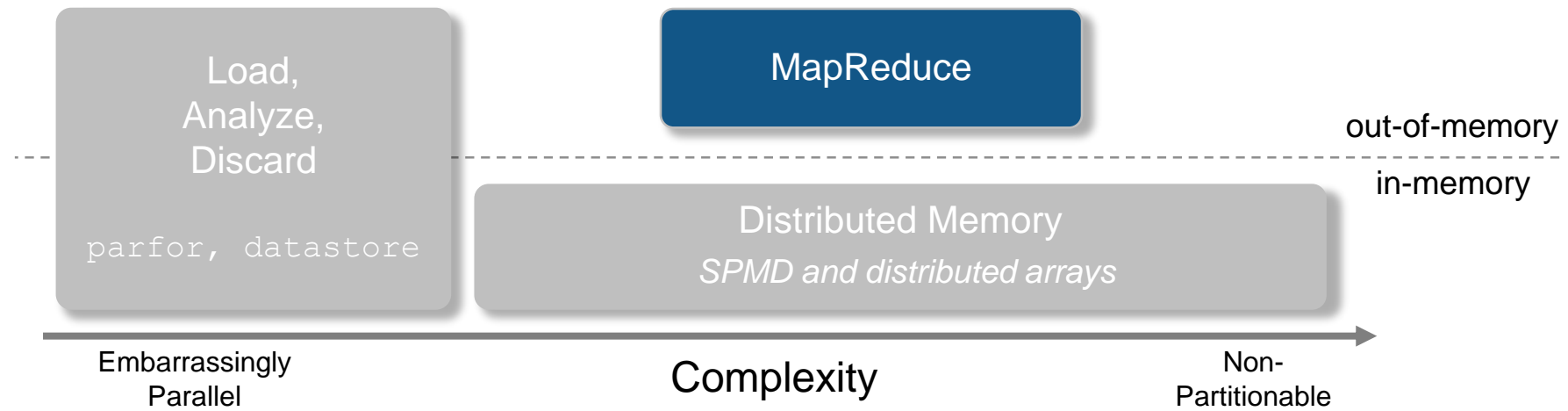


# Reading in Part of a Dataset from Files

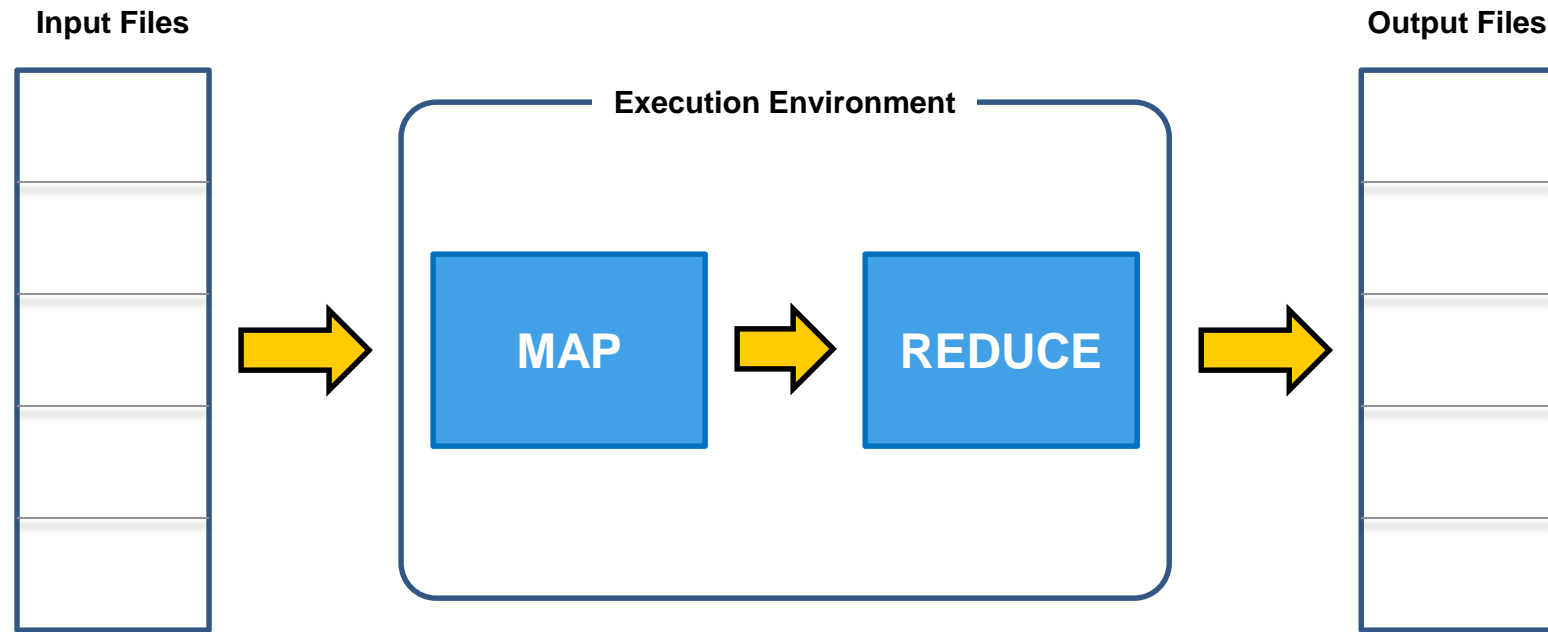
- Text file, ASCII file
  - `datastore`
- MAT file
  - Load and save part of a variable using the `matfile`
- Binary file
  - Read and write directly to/from file using `memmapfile`
  - Maps address space to file
- Databases
  - ODBC and JDBC-compliant (e.g. Oracle, MySQL, Microsoft, SQL Server)



# Techniques for Big Data in MATLAB



# MapReduce Programming Model



- `mapreducer`
- `datastore`
- `mapreduce`

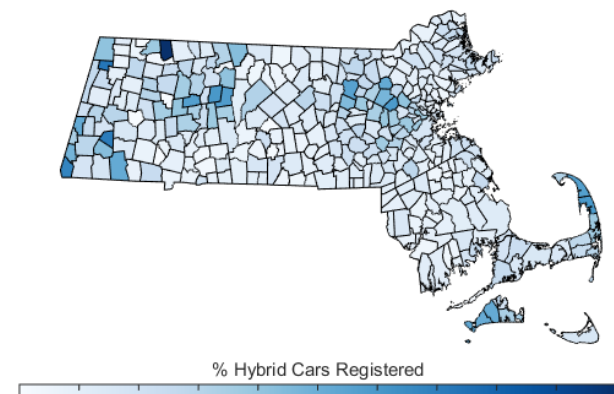
# Demo: Vehicle Registry Analysis

## Using MapReduce

- Data
  - Massachusetts Vehicle Registration Data from 2008-2011
  - 16M records, 45 fields
- Analysis
  - Examine hybrid adoptions
  - Calculate % of hybrids registered by quarter

muni_id	veh_zip	insp_year	model_year	make
325	1089	2011	2008	'Hyundai'
325	1089	2009	2008	'Hyundai'
288	1776	2011	2008	'Acura'
288	1776	2008	2008	'Acura'
145	2364	2011	2005	'Chevrolet'
325	1089	2010	2008	'Hyundai'
325	1089	2011	2008	'Hyundai'
288	1776	2009	2008	'Acura'

Hybrid Usage in Massachusetts Municipalities: q42011



# mapreduce

## Data Store

Veh_typ	Q3_08	Q4_08	Q1_09	Hybrid
Car	1	1	1	0
SUV	0	1	1	1
Car	1	1	1	1
Car	0	0	1	1
Car	0	1	1	1
Car	1	1	1	1
Car	0	0	1	1
SUV	0	1	1	0
Car	1	1	1	0
SUV	1	1	1	1
Car	0	1	1	1
Car	1	0	0	0

## Map

Hybrid	Key
0	Key: Q3_08
1	
1	
0	
0	

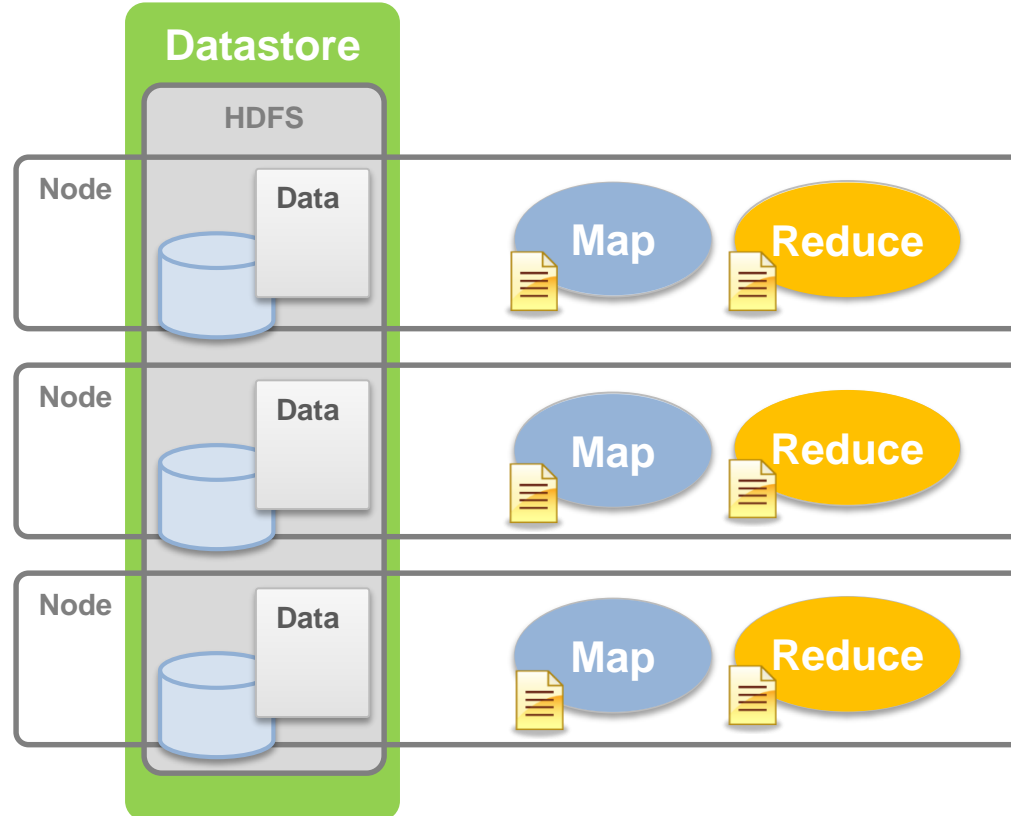
Hybrid	Key
0	Key: Q4_08
1	
1	
1	
0	
1	

Hybrid	Key
0	Key: Q1_09
1	
1	
1	
1	
0	
1	

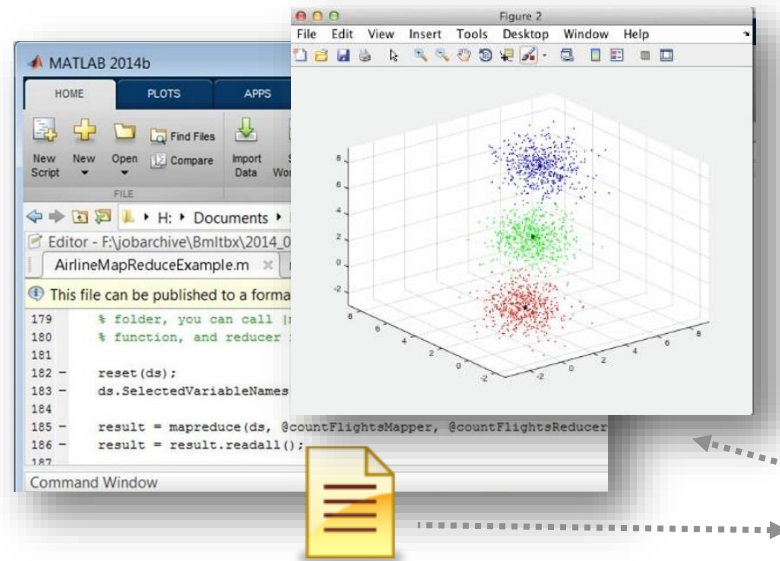
## Reduce

Key	% Hybrid (Value)
Q3_08	40%
Q4_08	67%
Q1_09	71%

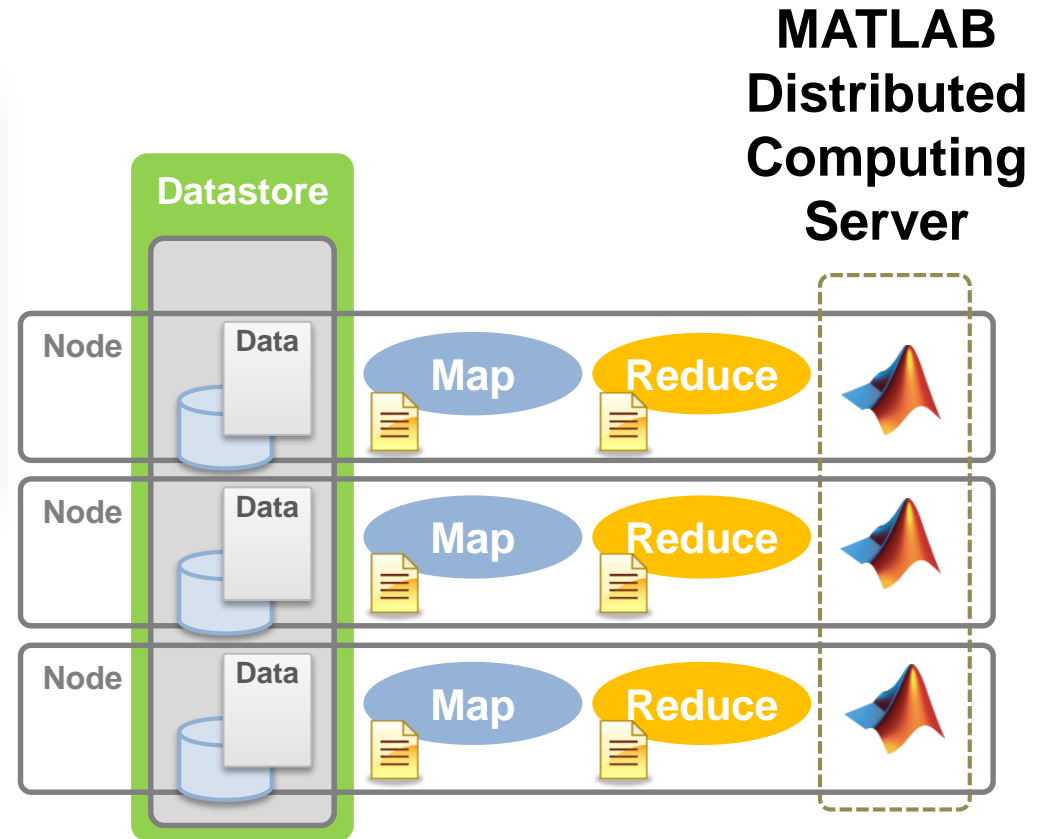
# The Big Data Platform



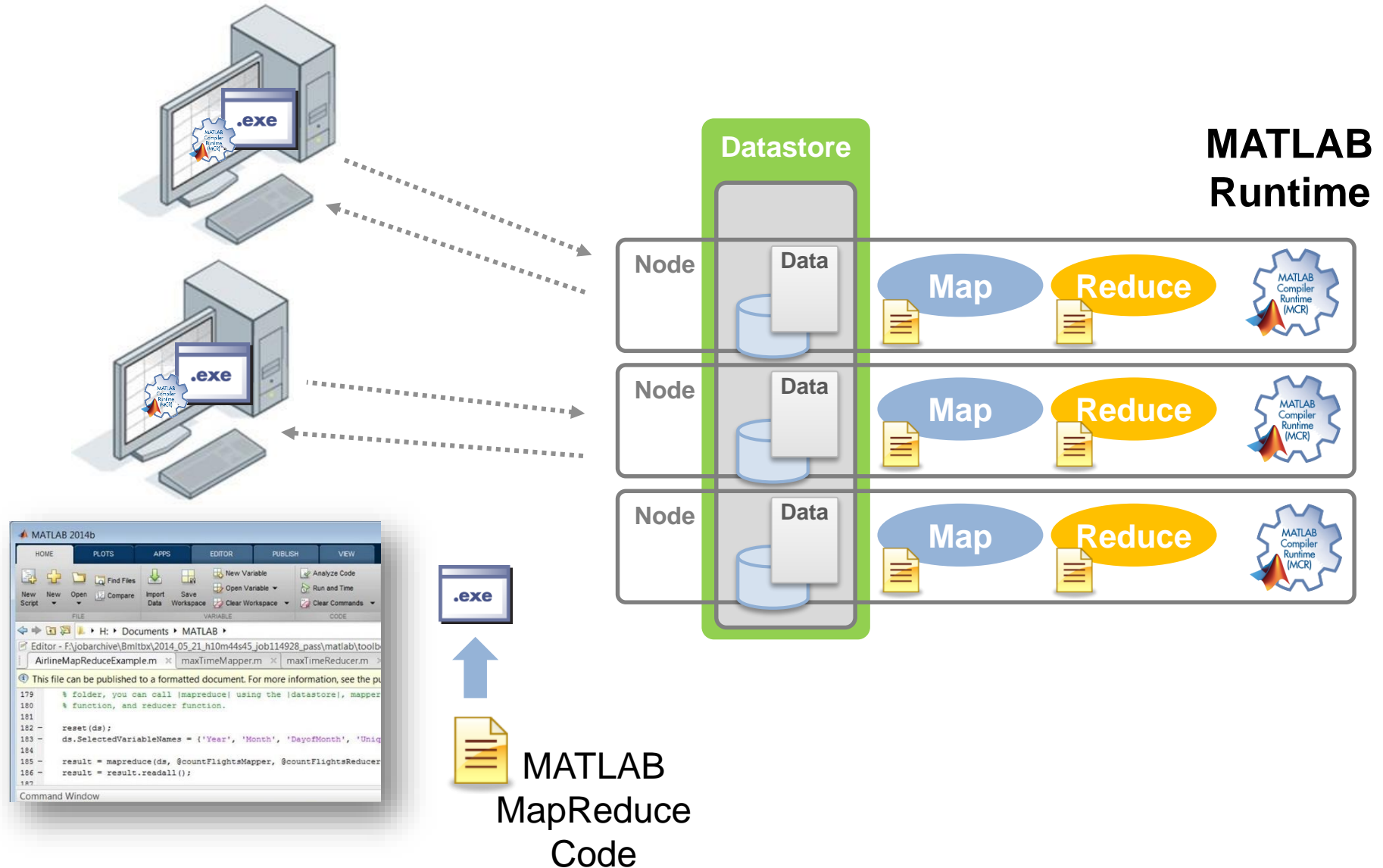
# Explore and Analyze Data on a Cluster



MATLAB  
MapReduce  
Code

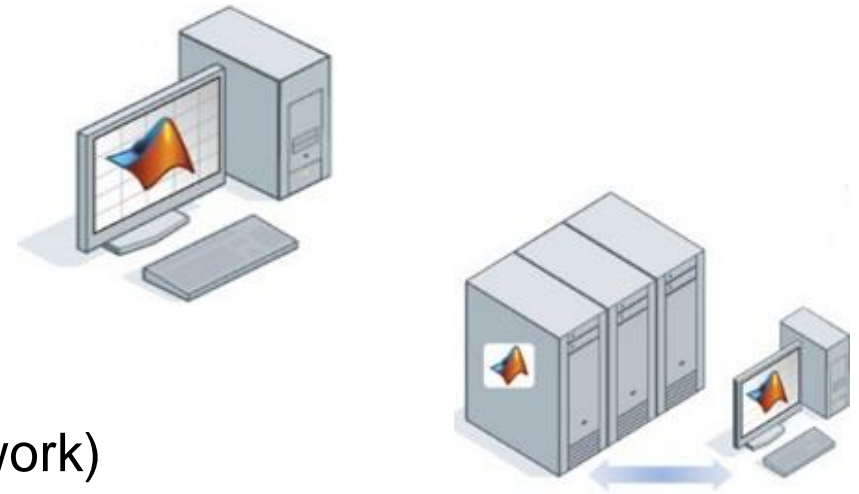


# Deployed mapreduce



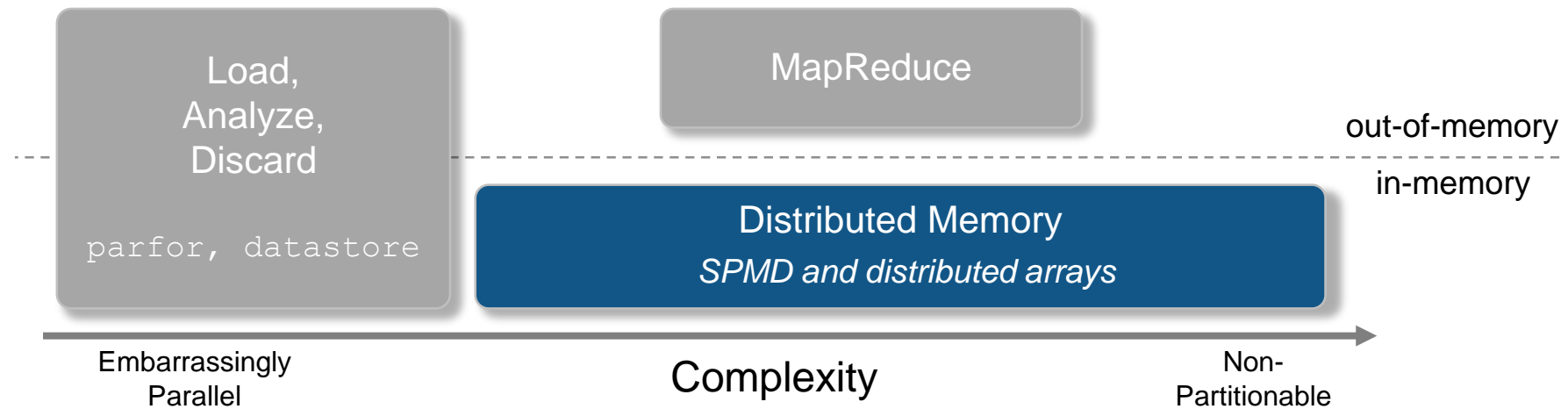
# When to Use `mapreduce`

- Data Characteristics
  - Text data in files or stored in the Hadoop Distributed File System (HDFS)
  - Dataset will not fit into memory
- Compute Platform
  - Desktop
  - Traditional HPC cluster **R2015a**
  - Hadoop cluster (within Hadoop MapReduce framework)
- Analysis Characteristics
  - Must be able to be partitioned into two phases
    1. Map: filter or process sub-segments of data
    2. Reduce: aggregate interim results and calculate final answer

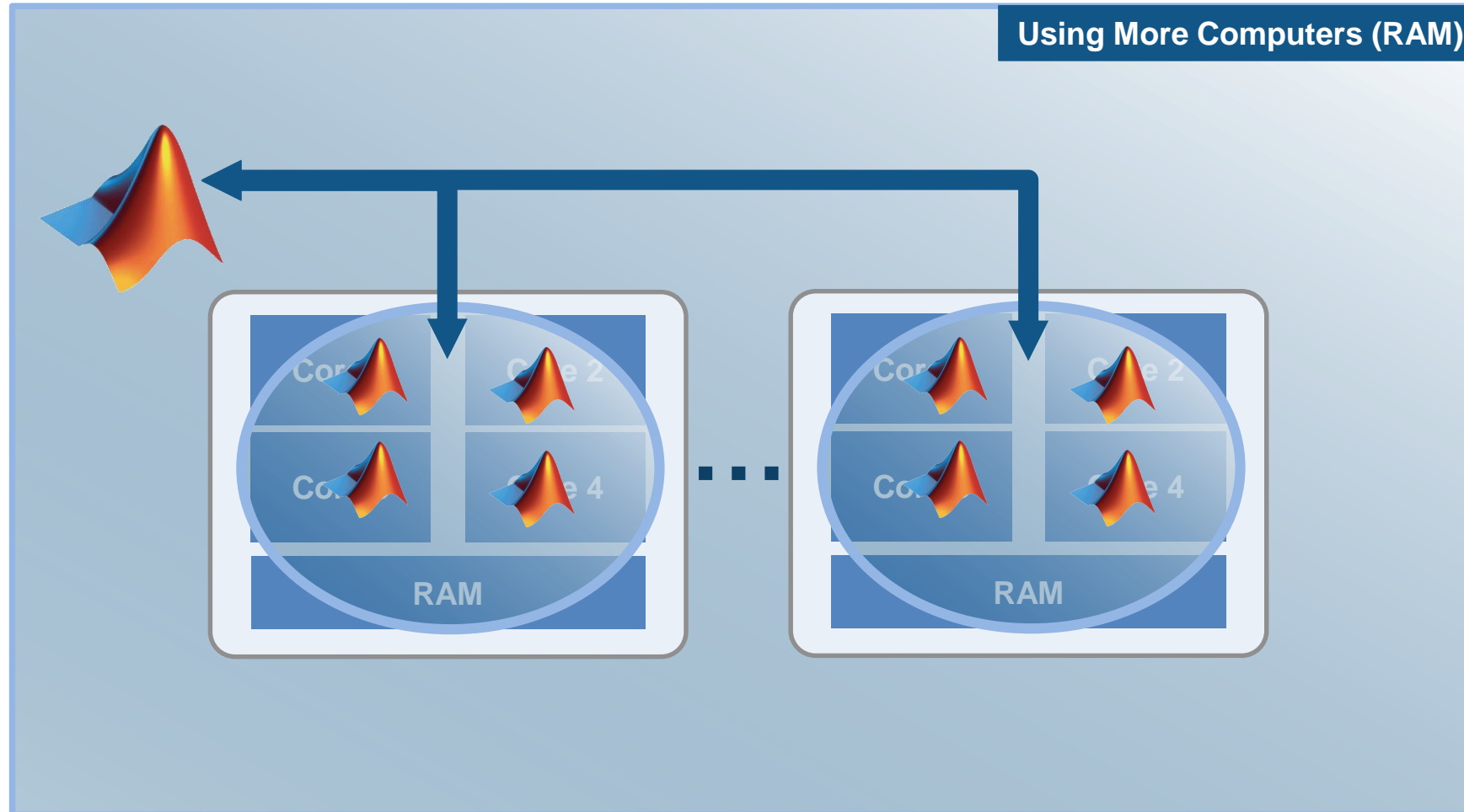




# Techniques for Big Data in MATLAB



# Parallel Computing – Distributed Memory



## spmd blocks

```
spmd
```

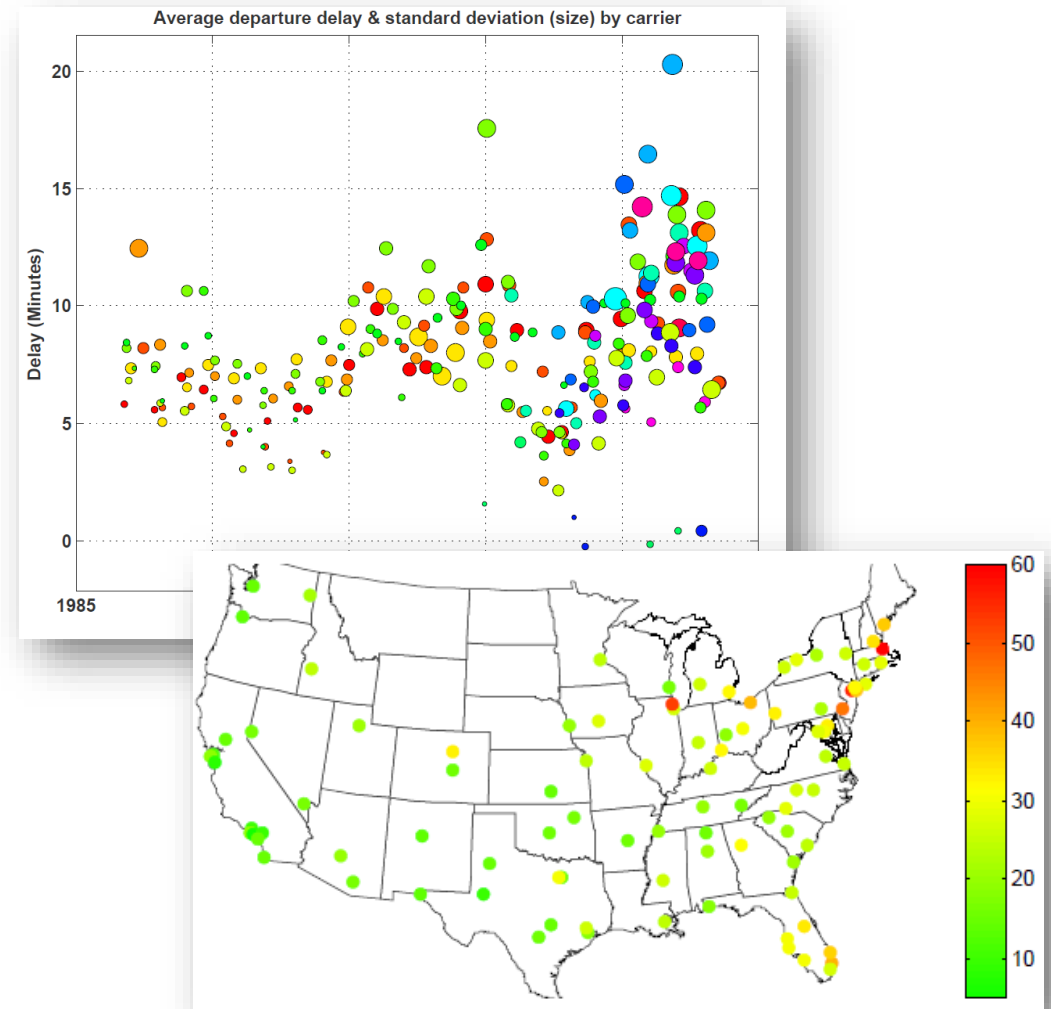
```
    % single program across workers
```

```
end
```

- Mix parallel and serial code in the same function
- Run on a pool of MATLAB resources
- **S**ingle **P**rogram runs simultaneously across workers
- **M**ultiple **D**ata spread across multiple workers

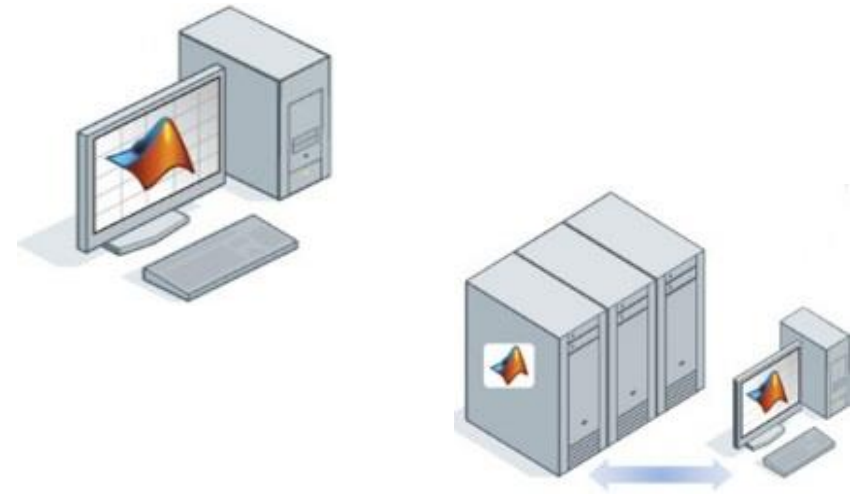
# Example: Airline Delay Analysis

- Data
  - BTS/RITA Airline On-Time Statistics
  - 123.5M records, 29 fields
- Analysis
  - Calculate delay patterns
  - Visualize summaries
  - Estimate & evaluate predictive models



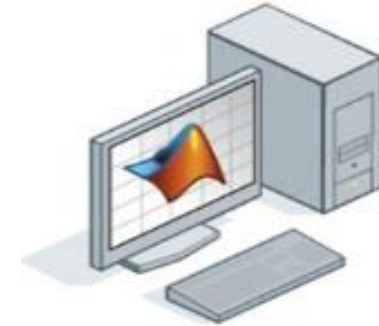
# When to Use Distributed Memory

- Data Characteristics
  - Data must be fit in collective memory across machines
- Compute Platform
  - Prototype (subset of data) on desktop
  - Run on a cluster or cloud
- Analysis Characteristics
  - Consists of:
    - Parts that can be run on data in memory ([spmd](#))
    - Supported functions for distributed arrays



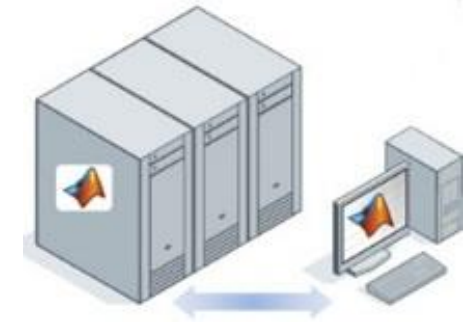
# Big Data on the Desktop

- Expand workspace
  - 64 bit processor support – increased in-memory data set handling
- Access portions of data too big to fit into memory
  - Memory mapped variables – huge binary file
  - Datastore – huge text file or collections of text files
  - Database – query portion of a big database table
- Variety of programming constructs
  - System Objects – analyze streaming data
  - MapReduce – process text files that won't fit into memory
- Increase analysis speed
  - Parallel for-loops with multicore/multi-process machines
  - GPU Arrays

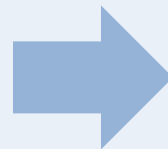


# Further Scaling Big Data Capacity

- MATLAB has a range of programming constructs for clusters
- General compute clusters
  - Parallel for-loops: embarrassingly parallel algorithms
  - SPMD and distributed arrays: distributed memory
  - MapReduce: process big text files
- Hadoop clusters
  - MapReduce: analyze data stored in the HDFS



Use these constructs  
on the desktop to  
develop your algorithms



Migrate to a  
cluster without  
algorithm changes

# Learn More

- MATLAB Documentation
  - Strategies for Efficient Use of Memory
  - Resolving "Out of Memory" Errors
  
- Big Data with MATLAB
  - [www.mathworks.com/discovery/big-data-matlab.html](http://www.mathworks.com/discovery/big-data-matlab.html)
  
- MATLAB MapReduce and Hadoop
  - [www.mathworks.com/discovery/matlab-mapreduce-hadoop.html](http://www.mathworks.com/discovery/matlab-mapreduce-hadoop.html)

**Big Data with MATLAB**

**How to work with huge and fast data sets**

Big data refers to the dramatic increase in the amount and rate of data being created and made available for analysis.

A primary driver of this trend is the ever increasing digitization of information. The number and types of acquisition devices and other data generation mechanisms are growing all the time.

Big data sources include streaming data from instrumentation sensors, satellite and medical imagery, from security cameras, as well as data derived from financial markets and retail operations. Big data sets from these sources can contain gigabytes or terabytes of data, and may grow on the order of megabytes or gigabytes per day.

**MapReduce on the Desktop**

Explore and analyze big data sets on your desktop with the MapReduce programming technique built into MATLAB.

Creating algorithms using MapReduce: max, mean, mean by group, histograms, covariance and related quantities, summary statistics by group, logistic regression, tall skinny QR

- » [Get started with MATLAB MapReduce](#)
- » [MapReduce design patterns](#)
- » [Use MATLAB MapReduce with relational databases](#)

**MapReduce on Hadoop**

Execute MATLAB MapReduce based algorithms within Hadoop MapReduce to explore and analyze data that is stored and managed on Hadoop, using MATLAB Distributed Computing Server.

- » [Run MATLAB MapReduce on Hadoop](#)

Create applications and libraries based upon MATLAB MapReduce for deployment within production instances of Hadoop, using MATLAB Compiler.

- » [Deploy MATLAB MapReduce applications to Hadoop](#)