Today's Agenda:

9:00 a.m.  公司产品许可证管理介绍
3:00-3:30 a.m.  大规模数据处理能力
4:00-4:30 a.m.  并行计算功能
5:00-5:30 a.m.  发布功能
6:00-6:30 a.m.  图像用户界面设计
7:00-7:30 a.m.  实时测量分析与发布

MathWorks Product Family Overview

Key Capabilities Drive The MathWorks Business
Engineering Challenges

- Shorter development time and increasing design complexity
- Large amounts of data to analyze
- Limited time to make sense of data
- Developing algorithms and applications in C/C++, Fortran, and VB is time-consuming
- Software tools are incomplete
  - Specialized point-and-click tools
  - Excel
    - Large data sets (row and column limit)
    - Limited file type support (e.g. audio, video, etc.)
    - Limited engineering functions

Today’s Agenda:

9:00 a.m. 11:30 a.m. 11:30 a.m. 11:30 a.m. 公司产品产品产品产品许可证管理介绍许可证管理介绍许可证管理介绍许可证管理介绍

9:30 a.m. 1:00 p.m. MATLAB Products Introduction to MATLAB Products Large Data Handling with MATLAB Parallel Computing with MATLAB

2:00 p.m. 4:00 p.m. MATLAB 大规模数据处理能力 Parallel Computing with MATLAB

3:00 p.m. 5:00 p.m. MATLAB 并行计算功能 Parallel Computing with MATLAB

4:00 p.m. 6:00 p.m. MATLAB 的发布功能 Application Deployment of MATLAB

5:00 p.m. 7:00 p.m. MATLAB 的图像用户界面设计 GUI of MATLAB

6:00 p.m. 8:00 p.m. MATLAB 实时测量分析与发布 Test and Measurement with MATLAB and Deployment

11:30 a.m. 问答回答 & A

Technical Computing Workflow

Access  Explore & Discover  Share

Demostration: Lane Marking Identification

- Description
  - Unintended lane departures are a major cause of road accidents
  - Detecting them is an important feature in luxury cars and trucks

- Approach
  - Camera captures images of the road ahead
  - Develop an algorithm to identify the lane markings in the captured images
Languages Used in Data Analysis

- **Spreadsheets**
  - Excel
- **Specialized and single-purpose tools**
  - Image processing application
  - Speech recognition application
- **General-purpose languages**
  - C, C++, FORTRAN
  - Java, Visual Basic, C#
- **Technical computing languages**
  - MATLAB

Data Access

- **Files**
  - Excel, text, or binary
  - Multimedia, scientific
  - Web, XML
- **Applications and languages**
  - .NET, C/C++, Java, Fortran, DLLs
  - Databases
- **Measurement hardware**
  - Data acquisition hardware for signals or images
  - Stand-alone instruments and devices

Data Analysis and Visualization

- **Data Analysis**
  - Manipulate, preprocess, and manage data
  - Fast, accurate analysis with pre-built math and engineering functions
- **Visualization**
  - Built in graphics functions for engineering and science (2D, 3D, VoI/O)
  - Interactive tools to annotate and customize graphics

What are Large Data Sets?

- 30% of MATLAB users access files > 100 MB
Total System Memory Available

- Storage for all processes =
  - Physical RAM (fast and expensive)
  - Page file on disk (cheap and slow)
- Memory Management Guide [Tech note 1106](#)
- Amount of RAM affects performance; not direct cause of "out of memory" errors

What’s the biggest array in MATLAB under Windows XP?

```matlab
>> a=zeros(? ,1);
>> whos
```

- a) 600MB
- b) 1GB
- c) 1.5GB

What are Large Data Handling Problems

1 Running out of memory
   - Large data sets need lots of memory to store and process
   - Computers have finite memory
   - Data set size > available memory: "Out of memory" errors

Solutions
   - Ensure available memory > required memory
   - *Maximizing available memory* on your system
     - System configuration
   - *Minimize required memory* in MATLAB
     - During access, storage, processing, plotting

Maximizing Total System Memory

- Size
  - Ensure non-zero or system managed page file
  - Max 4 GB
- Performance
  - Add RAM
  - Max 4 GB

Maximizing The MATLAB Process’s Virtual Memory

- Choose OS with largest process memory (in order):
  - 64-bit Linux, (future Win64)
  - 32-bit UNIX/Linux/MAC
  - Windows XP with /3gb
  - Window 2000, Windows XP (by default)
Checking The Virtual Memory Limit

```matlab
>> sys_unique_memstats
```

Increasing Process Limit on XP to 3G: 3GB Switch

- Right-click Properties > Advanced > Startup and Recovery > Edit
- Make copy of [Operating system line], change comment and add 3gb
- Reboot, select new OS option, check memstats
Please select the operating system to start:

- Microsoft Windows XP Professional
- Microsoft Windows XP Professional with SP2 switch
- DOS Prompt

Use the up and down arrow keys to move the highlight to your choice. Press ENTER to choose.
Seconds until highlighted choice will be started automatically: 5

For troubleshooting and advanced startup options for Windows, press F8.
The MATLAB Process’s Virtual Memory Limit with 3GB Switch

```matlab
>> system_dependent memstats
```

Minimizing Memory Requirements

- Data access
- Data storage
- Processing
- Plotting
How much temporary memory does data=data+1 require in an M-File?

Run M-file containing:
```matlab
data=zeros(1e6,1);
data=data+1;
>> monitormatlab
```
a) 1 MB  
b) 8 MB  
c) 16 MB

What is the fastest way to process MATLAB matrices with for loops?

a) Down the columns  
b) Along the rows  
c) Doesn’t matter
- Vectorized !!  
- Static!!

What are Large Data Handling Problems

- **Slowness**
  - Large data sets need lots of operations to process, and access
  - Today’s CPUs have limited speed  
  - Slowness due to page file use

- **Solutions**
  - Distributed computing
    - N Machines — N x Memory  
    - Subset of all applications (data parallel)

Solving Big Technical Problems

<table>
<thead>
<tr>
<th>Problem</th>
<th>You could...</th>
<th>Solutions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Long running</td>
<td>Fail</td>
<td>Tasks parallel</td>
</tr>
<tr>
<td>Computation</td>
<td>Reduce size of problem</td>
<td>Data parallel</td>
</tr>
<tr>
<td>Intensive</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


Parallel for loops

```matlab
parfor (i = 1 : n)
    % do something with i
end
```

- Mix task parallel and serial code in the same function
- Run loops on a pool of MATLAB resources
- Iterations must be order-independent
- M-Lint analysis helps to identify if existing for loops can be changed to `parfor`
Distributed Arrays, Parallel Algorithms

- Distributed arrays
  - Store segments of data across participating workers
  - Examples: doubles, sparse, logical, cell arrays, and arrays of structs

- Parallel algorithms for distributed arrays
  - Matrix manipulation operations
    - Examples: indexing, data type conversion, and transpose
  - Parallel linear algebra functions such as `svd`
  - Data distribution
    - Automatic, specify your own, or change at any time

MPI-based Functions in Distributed Computing Toolbox

Use when a high degree of control over parallel algorithm is required.

- High-level abstractions of MPI functions
  - `labSendReceive`, `labBroadcast`, and others

- Send, receive, and broadcast any MATLAB data type

- Automatic bookkeeping
  - Set-up: communication, ranks, etc.
  - Error detection: deadlocks and miscommunication

- Pluggable
  - Use any MPI implementation that is compatible with MPICH-2

Run four local workers with a DCT license

- Easy to experiment with explicit parallelism on multi-core machines
- Rapidly develop parallel applications on local computer
- Take full advantage of desktop power
- No separate compute cluster required

Scale up to cluster configuration with no code changes
Licensing: MATLAB Distributed Computing Engine

- One key required per worker
  - Packs of 8, 16, 32, 64, 128, etc.
  - Worker is a MATLAB session, not a processor

- All-product install
  - No code generation or deployment products

Dynamic Licensing
Dynamic Licensing

Supported on all MATLAB platforms

Versatile Modes of Access

Flexible Hardware
Third Party Schedulers

Extended support for 3rd-party schedulers

Open API for generic schedulers

MATLAB Distributed Computing Engine

Client Machine

Computer cluster

Third Party Schedulers

MATLAB Distributed Computing Engine

Client Machine

Function results = main(var1, var2)

results = postprocessing(out);

MATLAB Distributed Computing Engine

Client Machine

Function results = main(var1, var2)

results = postprocessing(out);
**Demo: Face Recognition**

- Do we recognise this person?
- Compare this image against a database.
- Images in database represented using 6 principal eigenfaces (component images).
- Image set must be handled in one bite.

**Dataset of Faces**

- Single snapshot used to build eigenfaces.
- Dataset also contains same individuals pulling different expressions – used to test recognition algorithm.
- 40 individuals in 10 poses in this dataset.

**Face Recognition Algorithm**

- Sample Faces processed into eigenface components.

  - Compute Mean Face
  - Subtract
  - Compute Eigenvectors (Eigenfaces)
  - Select 6 Principal Eigenfaces
  - Facial Signatures
  - Identification requires only this Reduced Dataset!
Face Recognition Algorithm
- Test image broken into eigenface components and compared with existing database.

Mean Face
Test Image
E Principal
Eigenfaces
Facial Signatures
Reconstruct Match
Reconstruct Test
Fit Metric

Test Image broken into eigenface components and compared with existing database.

Mean Face
Test Image
E Principal
Eigenfaces
Facial Signatures
Reconstruct Match
Reconstruct Test
Fit Metric

Air Traffic Control Radar Simulation
The Challenge
Perform air traffic control radar simulations by calculating the difference between the actual and estimated aircraft position for a variety of radar ranges and rainfall amounts.

The Solution
Use Simulink® and the Distributed Computing Toolbox to perform a parameter sweep of a Simulink Model.

The Results
Computations can run on low-cost computers
Nearly n-times speedup on n CPUs

Massachusetts Institute of Technology Integrates Cancer Research in the Lab and Classroom with MathWorks Tools
The Challenge
To improve diagnostic techniques for cancer by identifying proteins and analyzing their interactions

The Solution
Use MathWorks tools to enable students and researchers to analyze mass spectrometry data, model complex protein interactions, and visualize results

The Results
- Education integrated with research
- Computation time shortened by an order of magnitude
- Research grant won

Technical Computing Workflow
Access
Explore & Discover
Share

MATLAB Application Deployment

- Convert MATLAB applications into self-contained applications and software components
- Share them with end users who do not have MATLAB

Application Deployment Workflow

Integrating with Other Applications

- From MATLAB
  - Call Fortran and C
  - Generic DLL Interface
  - MEX Files
  - Integrate Java classes
  - Integrate COM servers and controls, including ActiveX
- From Fortran and C
  - MATLAB engine
  - Read/write MATLAB data files
- From .NET-based applications
  - MATLAB automation server

C++ Simulation

- Bluetooth and WiFi interference C++ simulation
- NIST Website: w3.antd.nist.gov/wctg/btleoth/bitint.html
Using MATLAB with Excel

- Accessing MATLAB from an Excel spreadsheet
  - MATLAB
  - Excel Link

- Deploying MATLAB as an Excel add-in
  - MATLAB
  - MATLAB® Compiler
  - MATLAB® Builder for Excel®

- Pass data and commands to MATLAB

- MATLAB is required

Automatically create a DLL and a Visual Basic application file that can be imported into Excel

- Convert your MATLAB algorithms into Excel add-ins through an easy-to-use GUI

- Automatically create a DLL and a Visual Basic application file that can be imported into Excel

- Create add-in functions that can run 95% faster than those created with VBA

- MATLAB is not required to use the add-in, but is required to create it

MATLAB® Builder™ Web Application

- Use Excel Link to
  - Pass data and commands to MATLAB
  - Retrieve data and results back into Excel

- MATLAB is required

Web Application

- Excel spreadsheet
- Internet
- Web Server
- .NET
- Java Application Server

Deploying MATLAB as an Excel Add-in

- Provides the functions that connect Excel and MATLAB
- Connection is "live" — requires no intermediate files or inter-process programming

Products Used

- MATLAB® Compiler
- MATLAB® Builder for Excel®
- Excel Link™
- Excel
- Database
Web Application Architecture

Java Servlet and JSP Output

Web Archive File

Web Browser

Who are the target users?
- MATLAB® users writing large applications
- C++ and Java™ prospects
- Spreadsheet users

MATLAB® Users
OO Programmers
Non-OO Programmers
Spreadsheet users
Overview of Handle Graphics

- Graphics in MATLAB consist of **objects**.
- Every graphics object has a unique handle and a set of **properties** that define its appearance.
- Objects are arranged in terms of a set **hierarchy**.

```
+Root (MATLAB Session)
 | +Figure
 |   | +Axes +Uicontrol +Uimenu +Uicontextmenu
 |   | +Image +Light +Line +Patch +Rectangle +Surface +Text
```

Using Handle Graphics

- **Handles** are MATLAB variables that allow us to identify and refer to graphics objects.

```
>> h_figure = figure;
>> h_axes = axes;
>> Y = rand(1,10);
>> h_line = plot(Y);
```

GUI Control Functions

- MATLAB provides a set of components we can use to build our GUI
  - Push Buttons, Editable Text Box, Pop-up Menu, ... see documentation on `uicontrol`.
  - We can use `uicontrol` to build a GUI programmatically

```
>> h_button = uicontrol('Style', 'pushbutton', ...
  'String', 'Hi David', 'Tag', 'mybutton',...
  'Position', [20 150 100 70], 'Callback', 'cla');
>> set(h(1), 'MenuBar', 'none');
```

Using GUIDE to build GUIs

- Building a GUI programmatically can be **painstakingly laborious**, if you’re working on a larger-scale project.
  - Instead we can use **GUIDE**, a visual aid which simplifies the task of designing the GUI layout.
  - GUIDE generates the underlying M-code automatically.
  - It automatically keeps track of all object handles via the `handles` structure.
Play the GUI

- Execute the GUI by clicking the Run button on the toolbar in either GUIDE or the Editor.

Refine the GUI

- With a little more effort, we can build something that looks like this.

How do we keep track of handles?

- We can easily lose track of graphics object handles when we create more and more.
- The solution is to store each of the object handles in a global handles structure and attach it to the GUI figure.

```matlab
>> handles.mybutton = h_button; % create handles
>> guidata(gcf, handles); % put handles in fig obj
>> handles2 = guidata(gcf) % get handles
```

Summary: MATLAB Advantages

- Data access
  - Applications such as C, C++, and Fortran
  - External devices via serial port
- Supported file formats (e.g., images, audio, video)
- Large data set handling
- Computational speed
- Algorithm development (e.g., GUIDE, programming features)
Why acquire live data in MATLAB?

- MATLAB is the standard tool for data analysis.
- Many people that analyze data also need to acquire data.
- One tool for acquisition and analysis allows problems to be solved and saves time.
- We have documentation to give customers back this up.

What is Measurement?

- Physical Quantity
- Sensor / Transducer
- Voltage
- Computer
- Hardware
- Data
- Measurement Types:
  - Temperature
  - Pressure
  - Flow
  - Acceleration
  - Rotation
T&M Toolboxes (MATLAB and Simulink)

- Instrument Control Toolbox (ICT) – Control and acquire data from instruments such as oscilloscopes, function generators, or signal analyzers, RS-232 serial devices, and remote applications using TCP/IP.
- Image Acquisition Toolbox (IAT) – Control and acquire data from cameras and frame-grabbers.
- Simulink support - All three toolboxes support both MATLAB AND Simulink. All three toolboxes come with Simulink blocks.

Hardware Connectivity

- Data Acquisition Toolbox
- Instrument Control Toolbox
- Image Acquisition Toolbox
- MATLAB Generic DLL
- MEX Interface for communicating with everything

Application Deployment: Soundcard Audio Spectral Analysis

- Acquires live data from measurement hardware
- Uses Data Acquisition Toolbox
- MATLAB Compiler deploys the application
  - Stand-alone

Data Acquisition Toolbox: Supported Hardware

- Acqiris*
- Agilent*
- ADLINK*
- Advantech
- CONTEC*
- Data Translation*
- g.tec*
- IOTech*
- Keithley
- Measurement Computing (MCC)
- National Instruments
- Ono Sokki*
- Parallel Port
- PC sound cards
- United Electronic Industries*
- VXI Technology

* Denotes that the hardware manufacturer made the investment to provide this support

For a full support listing, visit Supported Hardware tab on the DAT product page.