Accelerating Applications in a massive parallel environment using Windows HPC Server 2008 R2 and Excel 2010

Sandro Cambruzzi
Technical Solution Professional
sandroca@microsoft.com
Microsoft Entry into HPC

Windows Compute Cluster Server 2003

Personal Super Computing

- Built on Windows Server 2003
- Microsoft Entry into HPC
- Addressing Personal And Workgroup Needs
- End User Applications available for Windows
- Parallel and HPC Development Tools
- Ease of Management and Deployment
Microsoft HPC Server Today

Broad Reaching HPC

- Built on Windows Server 2008
- Support Traditional & Emerging HPC
- Larger Cluster support & Top500 Range
- Better integration for Windows-based Users
- Broader Developer support with tools and SOA
- Improved Management and Deployment
Windows HPC Tomorrow

Highly Scalable, Efficient HPC

- Built on Windows Server 2008 R2
- Scalable HPC Infrastructure for 1000+ nodes
- Customizable management elements for superior control
- Evolved SOA support for scale and programmability
- Ease parallel development with VS 2010 & Extensions .NET 4.0
- Distributed Services for Accelerating Microsoft Excel ® 2010
Windows HPC in the Future

Seamless Parallelism

- Tackling scalable applications from workstation to cluster to cloud
- Dynamic and elastic virtualized clusters
- Uniform programming model for scalable applications
- Easily deploy and manage scalable applications
- Capacity without limitations for users of those applications
- Addressing the explosion of data
Dawning 5000A, QC Opteron 1.9 Ghz, Infiniband, Windows HPC 2008, Dawning
Shanghai Supercomputer Center, China

is ranked

No. 10

among the world's TOP500 Supercomputers with
180.6 TFlop/s Linpack Performance

The 32nd TOP500 list was published at SC08 in Austin, TX November 18th, 2008

Congratulations from The TOP500 Editors

Hans Meuer
University of Mannheim

Erich Strohmaier
NERSC/Berkeley Lab

Jack Dongarra
University of Tennessee

Horst Simon
NERSC/Berkeley Lab
Tuning Wizard for Linpack

Single pack with HPL 1.0a + Intel math Kernel Lib
HPCS 2008 R2 Features

Windows HPC Server 2008 R2 dramatically increases the scale of HPC clusters with an out-of-the-box experience to deploy, manage and schedule large scale HPC systems.

**Scalability**
- Reliable deployment at scale
- Admin tool experience at scale
- Support for enterprise databases

**Ease of Use**
- Heat map customization
- Updated node template and software patching wizard
- Filter by node location

**Efficiency**
- iSCSI boot for diskless compute nodes
- Saved views and filtering
- Extensible diagnostics
- Support for custom reporting

**Systems Management**
- High volume job & task scheduling
- Dynamic parametric tasks

**Job Scheduler**
- Job progress tracking
- Node prep and release tasks
- Streamlined job troubleshooting
- Improved password management

**SOA Model**
- Duplex Message Exchange Pattern
- Message persistence
- Automated broker restart or failover

- Fire & Recollect SOA jobs
- Out-of-the-box setup for broker
- Unified WCF configuration
- SOA job monitoring
- HPC Services for Excel 2010

- Enhanced environment diagnostics
- Service list and configuration
- Graceful service shutdown
Parallelizing Excel

• Excel is frequently used as the “programming environment” for computational simulation

• Parallelizing Excel
  ➢ Greatly increases the compute power available to Excel
  ➢ Reduces the time it takes to actualize results and execute a greater number of calculations
  ➢ Benefit with shorter time to results and improved probability or likelihood of the outcomes
Excel 2010 on Windows HPC

Windows HPC Server 2008 R2 Provides 2 new mechanisms to distribute Excel calculations to an HPC Cluster

- **Excel SOA Client**
  - VSTO code in workbook calls out to SOA Service
  - Input and output managed by Excel developer

- **Distributed Excel**
  - Excel runs on the cluster and recalculates workbook
  - Spreadsheet as a service with input and output values
  - Works with Excel or other application as “client” (like above)
  - Requires HPC Server 2008 R2 and Excel 2010

- **Excel UDF on the Cluster**
  - Off-load existing UDF (external functions) to a cluster
  - Sample add-in for connecting to cluster
  - Requires HPC Server 2008 R2 and Excel 2010
Excel Workbooks on the Cluster

[Diagram showing the process of Excel workbooks on a cluster with steps involving request session, broker assignment, task submission, task assignment, and results.]
Parallelizing Excel

- Excel is frequently used as the “programming environment” for computational simulation
- Parallelizing Excel
  - Greatly increases the compute power available to Excel
  - Reduces the time it takes to actualize results and execute a greater number of calculations
  - Benefit with shorter time to results and improved probability or likelihood of the outcomes
Excel 2010 on Windows HPC

Windows HPC Server 2008 R2 Provides 2 new mechanisms to distribute Excel calculations to an HPC Cluster

Excel SOA Client
- VSTO code in workbook calls out to SOA Service
- Input and output managed by Excel developer

Distributed Excel
- Excel runs on the cluster and recalculates workbook
- Spreadsheet as a service with input and output values
- Works with Excel or other application as “client” (like above)
- Requires HPC Server 2008 R2 and Excel 2010

Excel UDF on the Cluster
- Off-load existing UDF (external functions) to a cluster
- Sample add-in for connecting to cluster
- Requires HPC Server 2008 R2 and Excel 2010
HPC Services for Excel
• Ability to execute entire spreadsheets on an HPC cluster
• Excel 2010 supports the use of HPC Services for Excel to control remote spreadsheets in a cluster

What is UDF Offloading?
• A mechanism to run User Defined Functions in parallel on a cluster
• Excel 2010 includes a new API and options for cluster computing
  – Support for .XLL files developed through Excel SDK
• Easy to develop on a single system and then deploy to a cluster
HPC Services for Excel

- Complex, or spreadsheet-based calculations.
  - multiple inputs to produce multiple outputs
- Wherever Excel is used as the calculation engine and not simply as a data UI
- Provides a lightweight and nondestructive way to port workbooks to a cluster
  - No need to develop a new platform
  - Users know how to work with Excel, don’t need to learn C, C#
  - Users can maintain their models themselves
- Supports blocking pop-up dialogs
- Scales: limited by SOA, not Excel

Remote UDFs

- Discrete calculation which is encapsulated in a function
  - For example a custom .XLL
- Calculation is complex or time-consuming
  - Should take longer than the network roundtrip
- The function changes infrequently
  - Recompilation requires deployment
- Calculation must be independent of other cells
  - Callback into the Spreadsheet breaks the architecture

Usage Guidance
Excel Workbooks on the Cluster
Offloading UDFs – Concept

- Microsoft Excel 2010
- Spreadsheet
- Offloading UDFs – Concept
- Windows HPC Server 2008 R2
- Request Session
- Broker Assignment
- Job Submission
- Results
- Head Node
- UDF
- Compute Nodes
- Windows HPC Server 2008 R2
- UDF
- UDF
- UDF
- UDF
- UDF
- UDF
Configuring Excel for UDF Offload

In the Excel Options dialog:
- Under Formulas:
  - Change options related to formula calculation, performance, and error handling.
  - Workbook Calculation:
    - Automatic
    - Automatic except for data tables
    - Manual
  - Recalculate workbook before saving
  - Enable iterative calculation
  - Maximum Iterations: 100
  - Maximum Change: 0.001

- Allow User Defined XLL functions to run on a compute cluster
  - Cluster type: Microsoft HPC Server 2008 x64
  - Options...

- Working with formulas:
  - Formula AutoComplete
  - Use table names in formulas
  - Use GetPivotData functions for PivotTable references

- Error Checking:
  - Enable background error checking
  - Reset Ignored Errors

- Error checking rules:
  - Cells containing formulas that result in an error
  - Inconsistent calculated column formula in tables
  - Cells containing years represented as 2 digits
  - Numbers formatted as text or preceded by an apostrophe
  - Formulas inconsistent with other formulas in the region
  - Formulas which omit cells in a region
  - Unlocked cells containing formulas
  - Formulas referring to empty cells
  - Data entered in a table is invalid
Visual Basic for Application

Public Function HasMoreWork() As Boolean
    ' first time:
    If iRowIndex = 0 And iColIndex = 0 Then
        iNumCol = 1
        While iNumCol < MaxColumn + 1 And CDB1.Range("Table1!").Offset(iNumCol - 1, 0).Value <> 0
            iNumCol = iNumCol + 1
        Wend
        iNumRow = 1
        While iNumRow < MaxRow + 1 And CDB1.Range("Table1!").Offset(iNumRow - 1, iNumCol).Value <> 0 And CDB1.Range("Table1!").Offset(iNumRow - 1, 0).Value <> CDB1.Range("Table1!").Offset(1, 0).Value
            iNumRow = iNumRow + 1
        Wend
        ' update globals
        iRowCount = iNumRow
        iColCount = iNumCol
        Range("Sheet1!").ClearContents 'This needs to occur first!!!
    End If

    HasMoreWork = True
    If iRowIndex >= iRowCount Then
        iRowIndex = 0
        iColIndex = iColIndex + 1
    End If
    If iColIndex >= iColCount Then
        HasMoreWork = False
        ' reset for next call
        iRowIndex = 0
        iColIndex = 0
    End If
End Function

Public Function Parse(Optional Data As Variant)
    ' insert the result, first two values
    ' in the array are the address
Application Flow (Original)

1. Read workbook (parameters)
2. Setup initial variables
3. Calculate (Loop)
4. Write results to workbook
Application Flow (Original)

Read workbook (parameters)

Setup initial variables

Calculate (Loop)

Write results to workbook

Calculate one iteration

Update summary statistics

More trials?
Application Flow (Cluster)

1. Read workbook (parameters)
2. Setup initial variables
3. Calculate (Asynchronous) "Client code replaces the VBA loop. In this example, the client is a VSTO add-in."
4. Write results to workbook
Application Flow (Cluster)

- Read workbook (parameters)
- Setup initial variables
- Calculate (Asynchronous)
- Write results to workbook

The service runs on compute nodes, and manages an Excel instance on each core.

Load workbook (if necessary)

Calculate one iteration

The VBA code for a single iteration is the same as the original desktop code.
Application Flow (Cluster)

- Read workbook (parameters)
- Setup initial variables
- Calculate (Asynchronous)
- Write results to workbook
- Load workbook (if necessary)
- Load workbook (if necessary)
- Calculate one iteration
- Load workbook (if necessary)
- Load workbook (if necessary)
- Calculate one iteration
- Load workbook (if necessary)
- Load workbook (if necessary)
- Calculate one iteration
- Load workbook (if necessary)
- Load workbook (if necessary)
- Calculate one iteration
Application Design

Workbook: On the Desktop

- Manage Calculation
- Prepare Spreadsheet
- “Clean” Calculation Framework
- Prepare Requests
- Process Results

Workbook: On Compute Nodes

- Run Single Calculation
- “Atomic” Method: call with parameters, return result
- “Reset” Spreadsheet
- No Side-Effects (or, reload)
Implementation

• Macro framework
  – Partition
  – Merge
  – HasMoreWork
  – Execute

• Managing Data
  – Generic data
  – Serializing parameters

• HPCExcelService
  • ExcelDriver
  • ExcelClient
Function RunModel()
    ' set up variables
    NumIterations = Range("C8").Value
    ' ...

    For n = 1 To NumIterations
        CashRate = CashMu + CashSD * Rnd_N01()
        CashInvGain = (Exp(CashRate * dt) - 1) * Cash
        QtrlyTaxableGain = QtrlyTaxableGain + CashInvGain
        Cash = Cash + CashInvGain
        If Cash < MinCashQtrLevel(n) Then MinCashQtrLevel(n) = Cash
        If Cash > MaxCashQtrLevel(n) Then MaxCashQtrLevel(n) = Cash
    ' ...

    Next n
    call UpdateCharts
    ' ...

End Function
Function Calculate()

    CashRate = CashMu + CashSD * Rnd_N01()
    CashInvGain = (Exp(CashRate * dt) - 1) * Cash
    QtrlyTaxableGain = QtrlyTaxableGain + CashInvGain
    Cash = Cash + CashInvGain
    If Cash < MinCashQtrLevel Then MinCashQtrLevel = Cash
    If Cash > MaxCashQtrLevel Then MaxCashQtrLevel = Cash
    ' ...

Dim results(32) As Variant
results(1) = MinCashQtrLevel
results(2) = MaxCashQtrLevel
' ...
Calculate = results

End Function
Cluster of Workstations (CoW)

Expand the capacity of HPC clusters while increasing the return on your existing technology investments by utilizing desktop cycles as part of your over HPC infrastructure.

- **Feature Summary**
  - HPC Server 2008 R2 introduces the ability to add Windows 7 workstations as compute nodes within an HPC Cluster
  - View and monitor workstations the same as dedicate compute nodes
  - Time of day scheduling for Workstation availability

- **Requirements**
  - Windows 7 Professional or Enterprise, 32 or 64-bit (requires Active Directory support)
  - Desktops and cluster in same AD Domain
  - Only supported for network topology 5 where desktops, head nodes, compute nodes, and broker nodes need to be on the same physical, public, network.

- Unlike compute nodes, deployment of desktop OS is not supported