

Parallel Computing @Microsoft

Sérgio Martinho
sergioma@microsoft.com
Microsoft Portugal

Challenges?



Free Lunch Is Over For Traditional Software

Free Lunch
for traditional software

Operations per second for serial code

24 GHz
1 Core

12 GHz
1 Core

6 GHz
1 Core

3 GHz
1 Cor 3 GHz
1 Cores

3 GHz
2 Cores

3 GHz
4 Cores

3 GHz
8 Cores

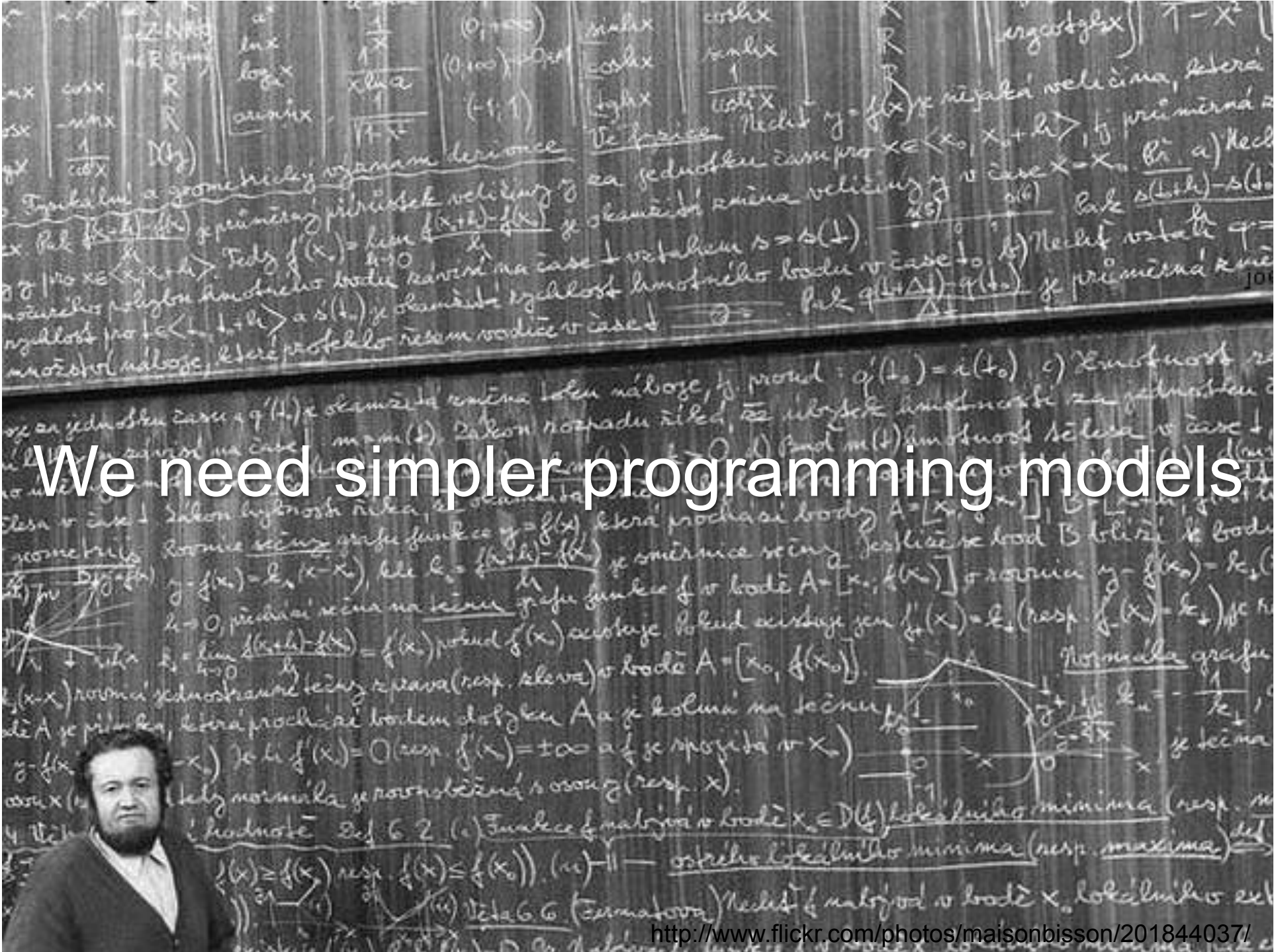
Additional operations per second if code can take advantage of concurrency



No Free Lunch for traditional software
(Without highly concurrent software it won't get any faster!)

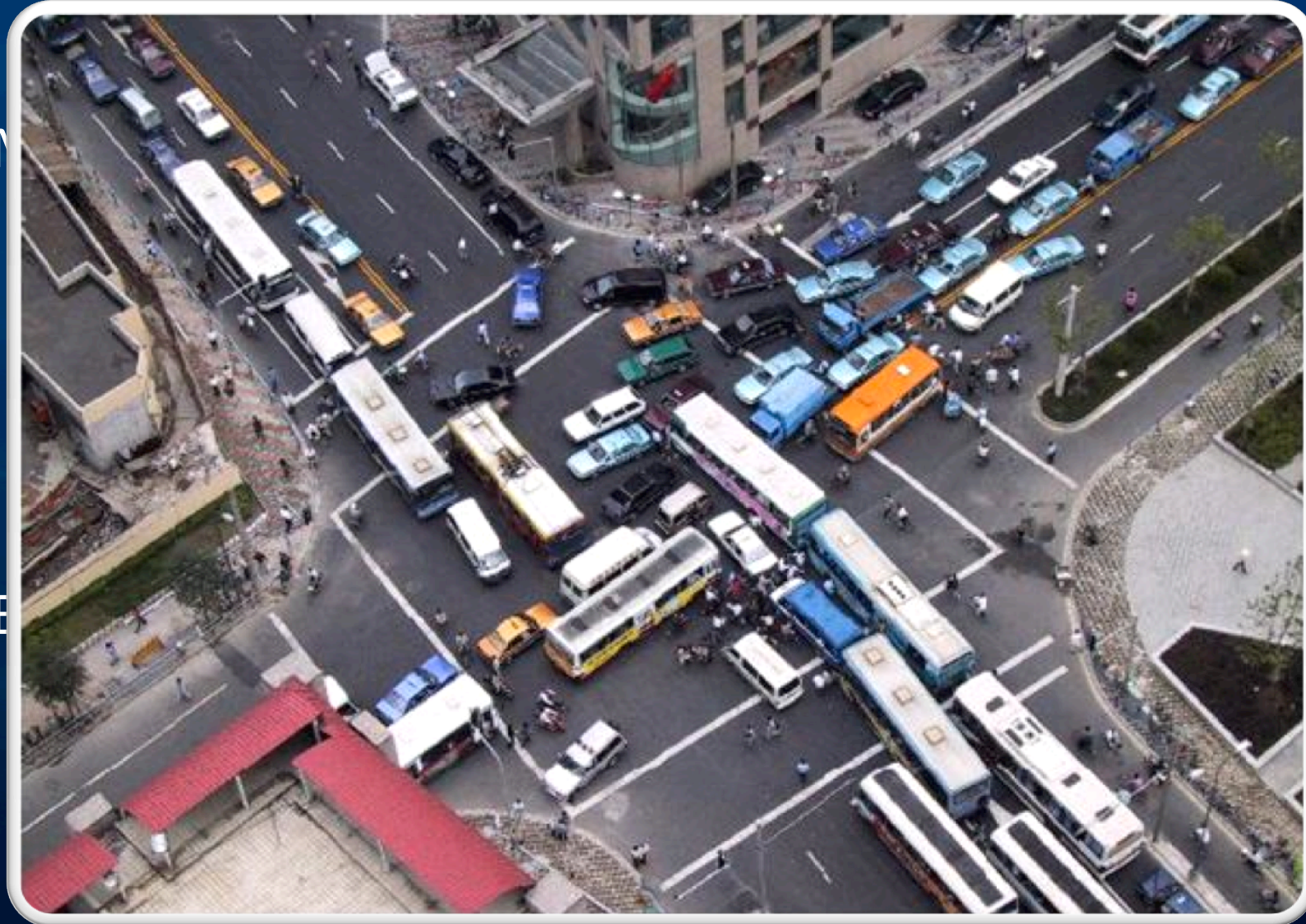
- Scaling distributed systems is hard
- Data sets are increasing
- Programming models are complex





We need simpler programming models

I'm convinced... now what?



•

N

•

E

Example: “Race Car Drivers”

```
IEnumerable<RaceCarDriver> drivers = ...;
var results = new List<RaceCarDriver>();
foreach(var driver in drivers)
{
    if (driver.Name == queryName &&
        driver.Wins.Count >= queryWinCount)
    {
        results.Add(driver);
    }
}
results.Sort((b1, b2) =>
    b1.Age.CompareTo(b2.Age));
```


Manual Parallel Solution

```
IF... > drivers = ...;  
var ... CarDriver>();  
int pa... ProcessorC...  
int rema... Count;  
var enumer... rator()  
try {  
    using (var ...  
        for(int i...  
            Thread...  
                while...  
                    Rac...  
                    lock...  
                        ) break;  
                    {  
                        result...  
                    }  
                    ked.Decrement(re... done.Set());  
                    );  
                    t((b1, b2) => b1.Age.CompareTo(b2.Age...  
                }  
            }  
finally { if (enumerator is IDisposable) ((IDisposable)enumerator
```

Synchronization Knowledge

Inefficient locking

Lack of foreach simplicity

Manual aggregation

Tricks

Heavy synchronization

Non-parallel sort

LINQ Solution

```
var results = from driver in drivers .AsParallel()  
              where driver.Name == queryName &&  
                  driver.Wins.Count >= queryWinCount  
              orderby driver.Age ascending  
              select driver;
```

Example 2: MPI “Hello World”

```
#include <stdio.h>
#include <mpi.h>

int main (int argc, char *argv[])
{
    int rank, size,
    MPI_Init (&argc, &argv);

    MPI_Comm_rank (MPI_COMM_WORLD, &rank);
    MPI_Comm_size (MPI_COMM_WORLD, &size);

    printf( "Hello world from process %d of %d\n", rank, size );

    MPI_Finalize();

    return 0;
}
```

Example 2: MPI “Hello World”

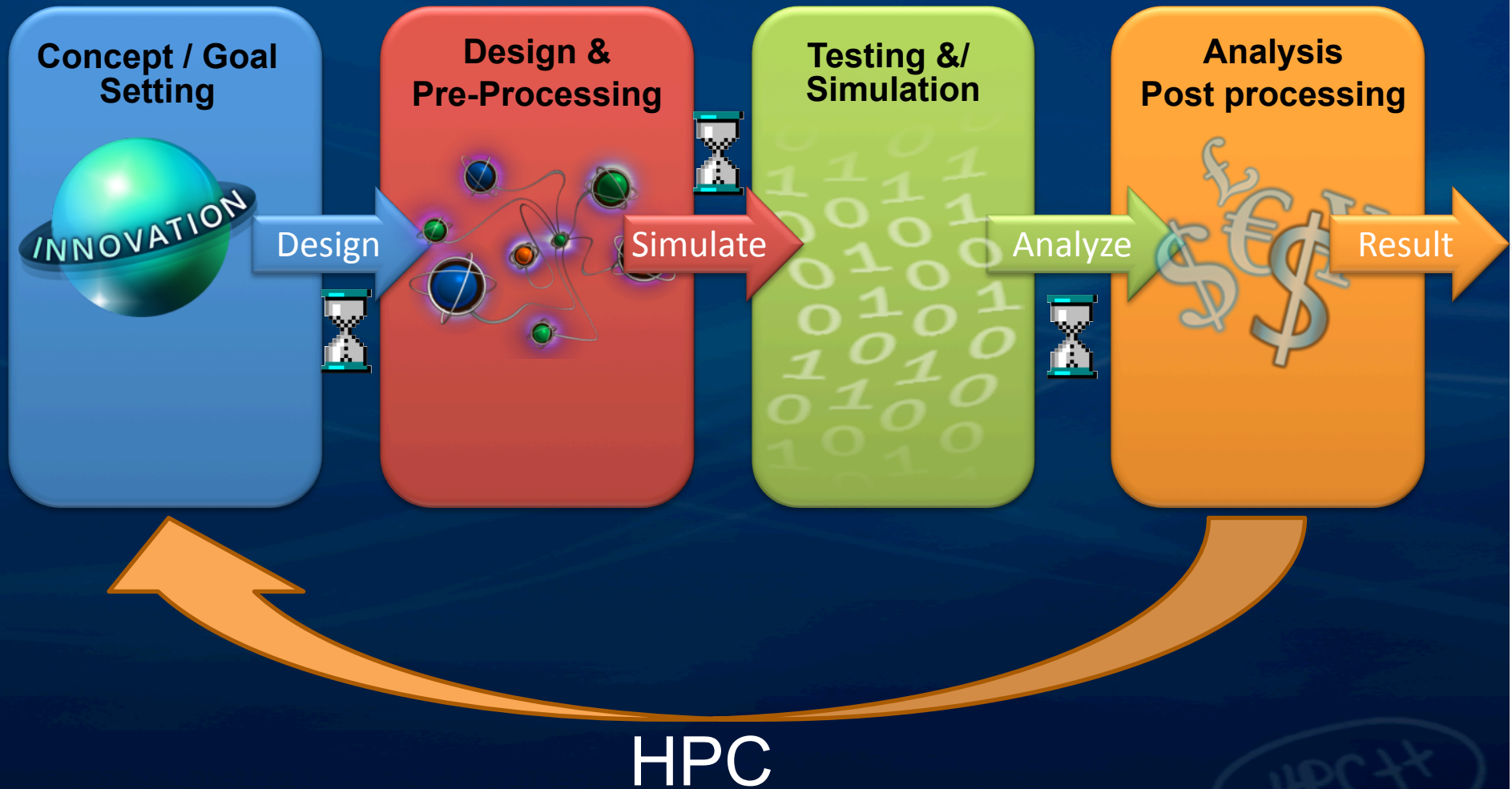
```
using System;
using MPI;

class MPIHello
{
    static void Main(string[] args)
    {
        using (new MPI.Environment(ref args))
        {
            int rank = MPI.Communicator.world.Rank;
            int worldSize = MPI.Communicator.world.Size;

            Console.WriteLine("Hello World from process"
                + rank + " of " + worldSize);
        }
    }
}
```

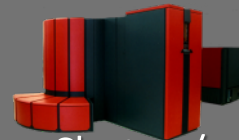

What is HPC?

HPC is computing infrastructure to accelerate innovation



HPC++

Today's HPC Environment



Clusters/
supercomputers



High speed
networking



Storage



Scientists



Engineers



Financial
analysts



Compilers



Specialized
languages



Debuggers

High Productivity Computing Windows HPC Server 2008



HPC and IT data centers merge, streamlined cluster management



Users with broad access to multiple cores and servers



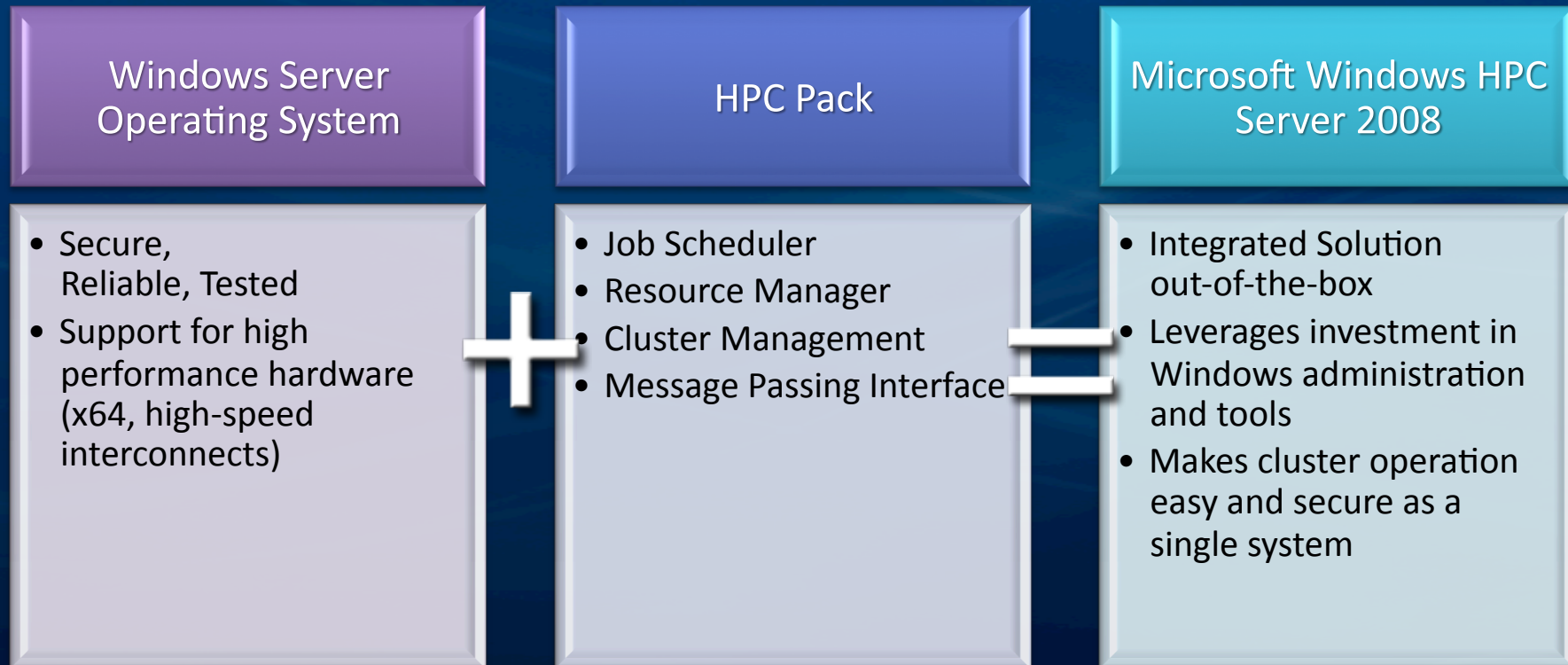
Simplified parallel development



Windows HPC Server 2008



- Complete, integrated platform for computational clustering
- Built on top the proven Windows Server 2008 platform
- Integrated development environment



Windows HPC Server 2008

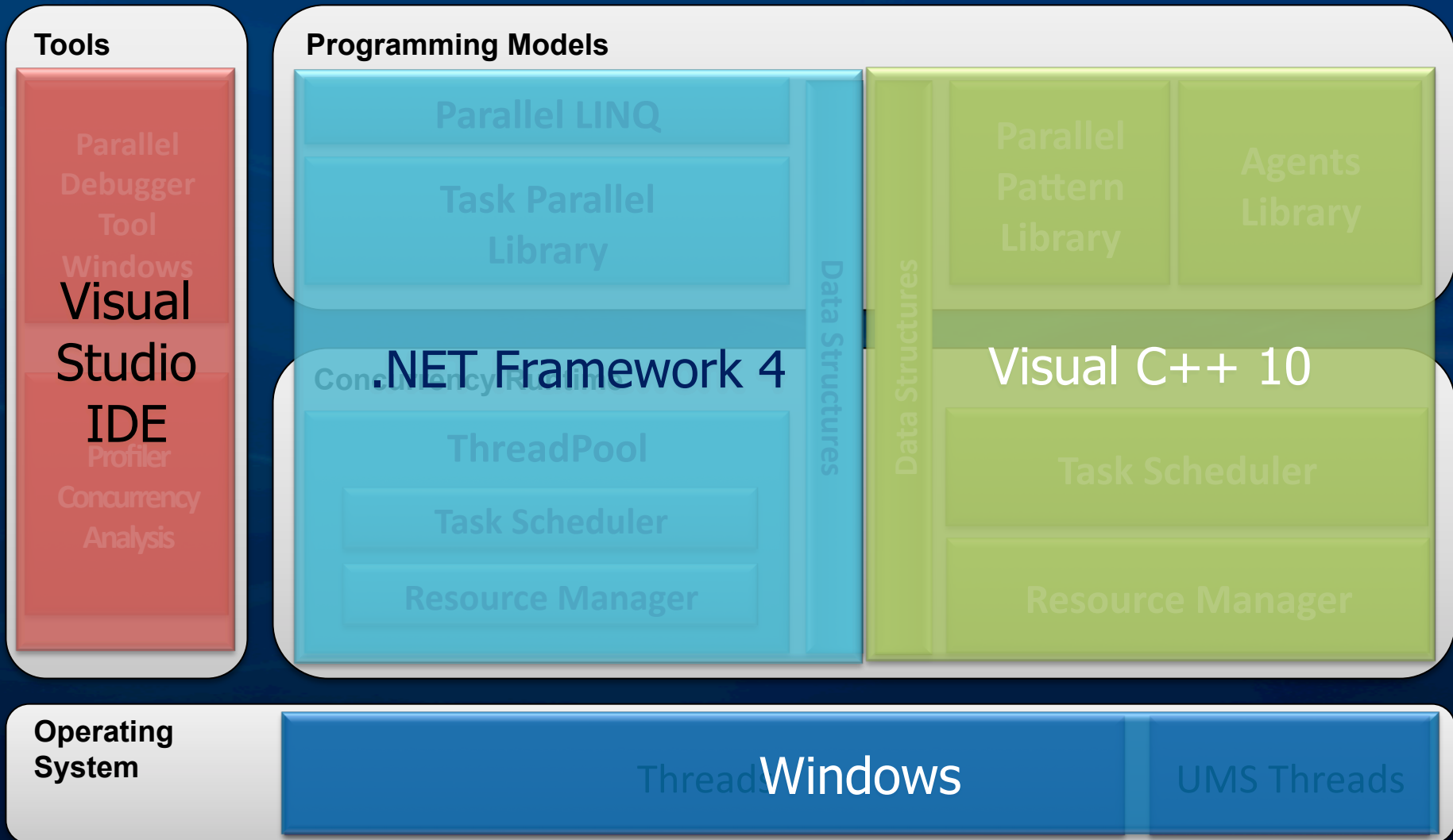


- **Simple to setup and manage**
 - Simplify system and application deployment
 - Base images, patches, drivers, applications
- **Focus on ease of management**
 - Comprehensive diagnostics , troubleshooting and monitoring
 - Familiar, flexible and “pivotal” management interface
 - Equivalent command line support for unattended management
- **Increased Scale**
 - Scale deployment, administration, infrastructure
 - Cluster usage reporting
- **Better integration**
 - Patch Management
 - System Center Operations Management
 - PowerShell
 - Windows 2008 high Availability Services



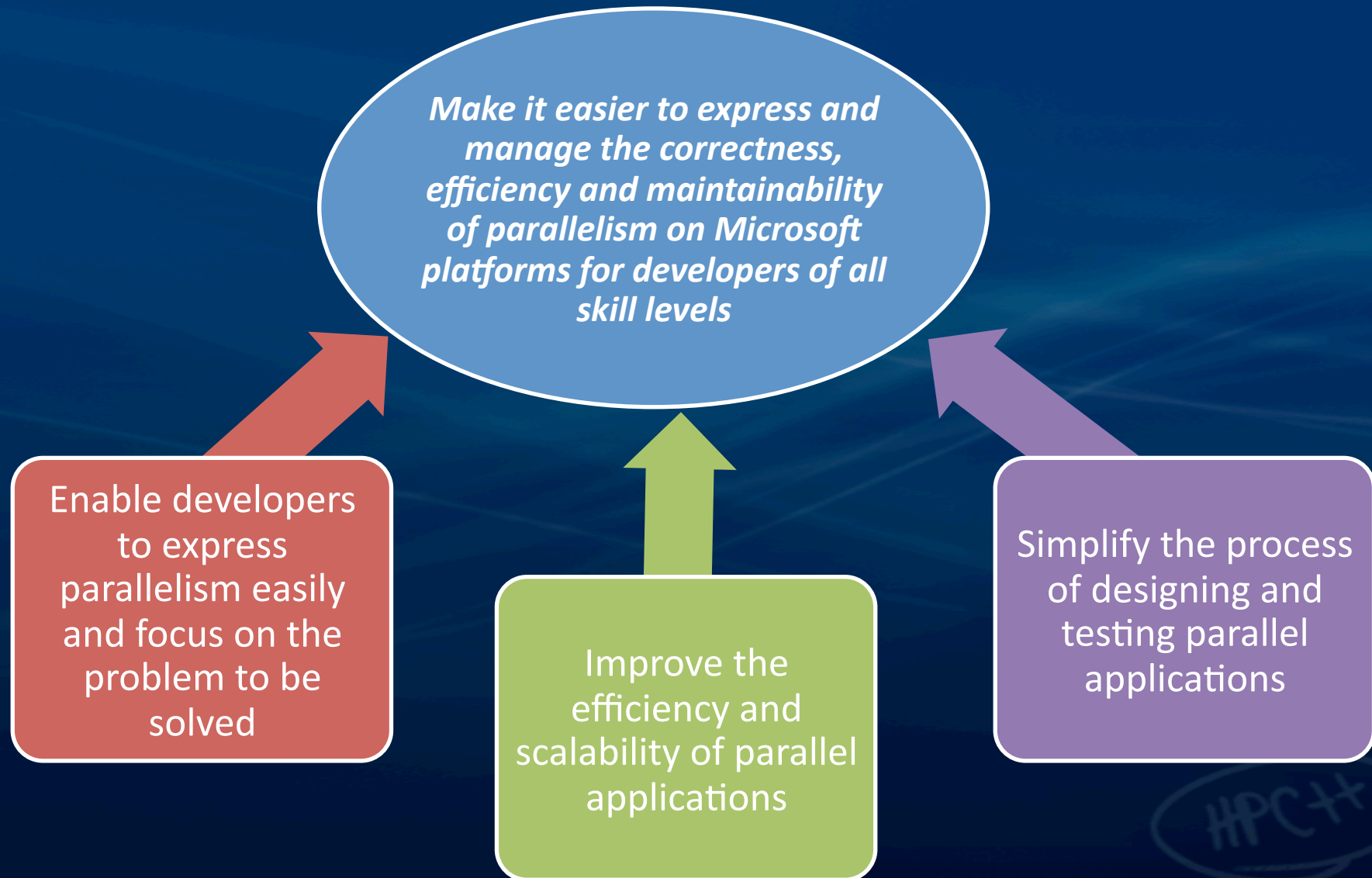
Visual Studio 2010

Tools/Programming Models/Runtimes



Key: Managed Native Tooling

Developer Accessibility



HPC++



Microsoft's vision for HPC

Microsoft's Vision for HPC



“Provide the platform, tools and broad ecosystem to reduce the complexity of HPC by making parallelism more accessible to address future computational needs.”

Reduced Complexity

Ease deployment for larger scale clusters

Simplify management for clusters of all scale

Integrate with existing infrastructure

Mainstream HPC

Address needs of traditional supercomputing

Address emerging cross-industry computation trends

Enable non-technical users to harness the power of HPC

Broad Ecosystem

Increase number of parallel applications and codes

Offer choice of parallel development tools, languages and libraries

Drive larger universe of end-users, developers, and system administrators

Democratize!

Parallel Computing for the Masses

Additional Information

- Microsoft HPC Web site
 - <http://www.microsoft.com/hpc>
- Parallelism at Microsoft
 - <http://msdn.microsoft.com/concurrency>
- Visual Studio 2010 Release Candidate
 - <http://msdn.microsoft.com/visualstudio>
- Windows HPC TechCenter
 - <http://technet.microsoft.com/en-us/hpc/default.aspx>
- HPC on MSDN
 - <http://code.msdn.microsoft.com/hpc>
- To join Microsoft's HPC Newsletter
 - hpcnews@microsoft.com



Questions

- To join Microsoft's HPC Newsletter
 - hpcnews@microsoft.com



HPC++

Microsoft®

Your potential. Our passion.™

Sérgio Martinho
sergioma@microsoft.com
Microsoft Portugal

© 2008 Microsoft Corporation. All rights reserved. Microsoft, Windows, Windows Vista and other product names are or may be registered trademarks and/or trademarks in the U.S. and/or other countries. The information herein is for informational purposes only and represents the current view of Microsoft Corporation as of the date of this presentation. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information provided after the date of this presentation.

MICROSOFT MAKES NO WARRANTIES, EXPRESS,
IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS PRESENTATION.

HPC++