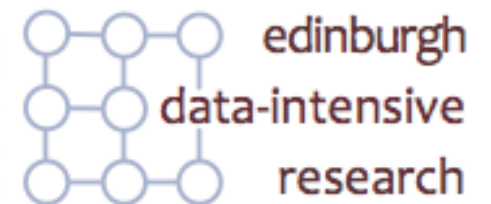


PRACtICaL-MPI: **P**ortable **A**daptive **C**ompression Library **MPI**

Author: Rosa Filgueira Vicente
University Carlos III
University of Edinburgh



Summary

2

1. Problem description
2. Main Objective
3. Strategy for improve the performance communication operations
4. Evaluations

Summary

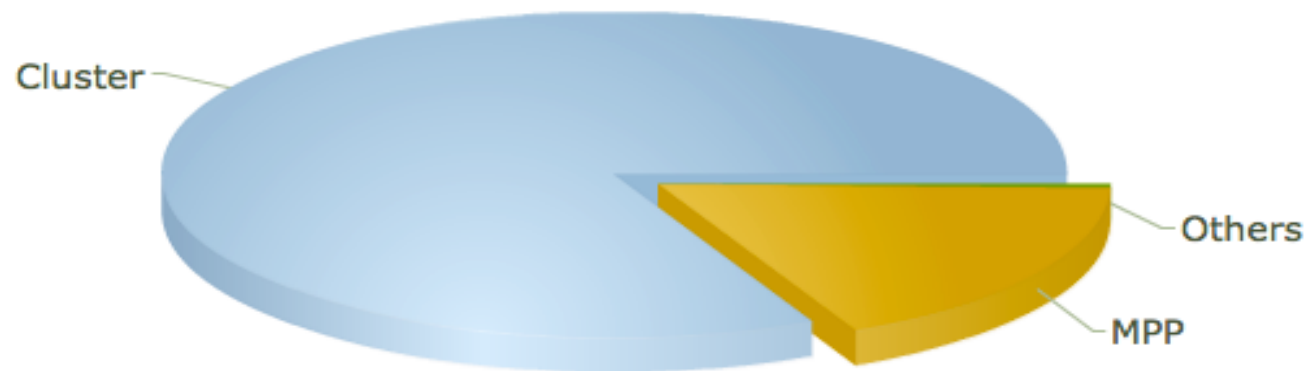
3

1. Problem description
2. Main Objective
3. Strategy for improve the performance communication operations
4. Evaluations

Problem description

4

- Parallel computation on cluster has become the most common solution for HPC application.

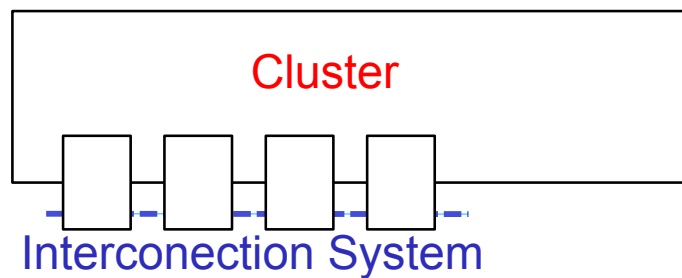


TOP 500
June 2011

Problem description

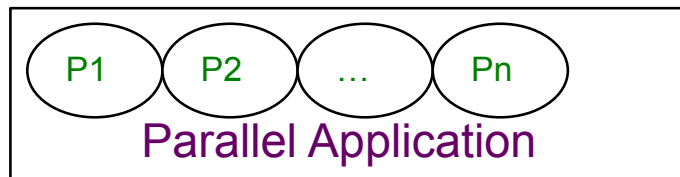
5

- **Cluster** is a group of linked computers, working together closely thus in many respects forming a single computer.

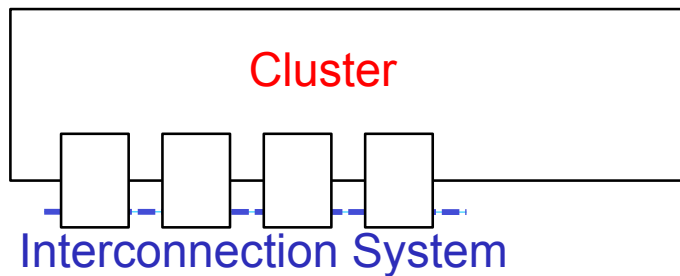


Problem description

6

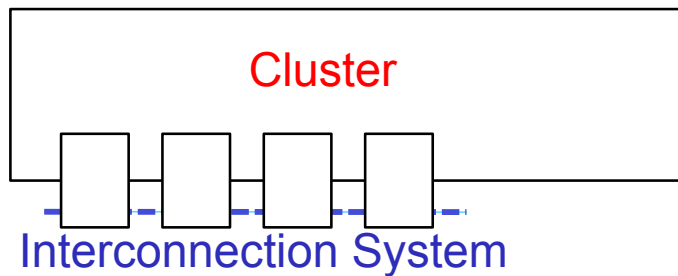
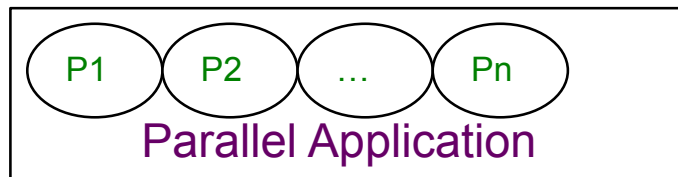


- On a cluster, one/many parallel applications could be running.



Problem description

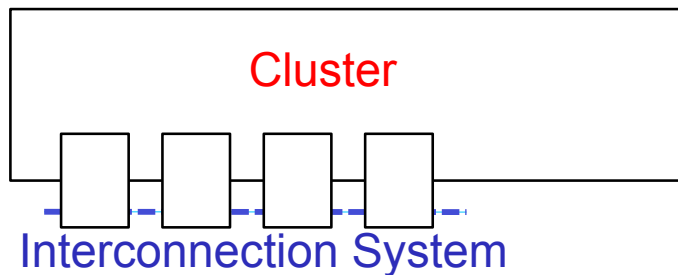
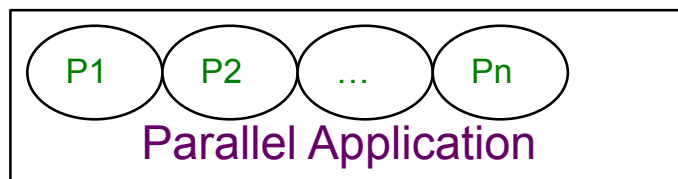
7



- One of most communication middleware used is the standard **MPI** (Message Passing Interface).

Problem description

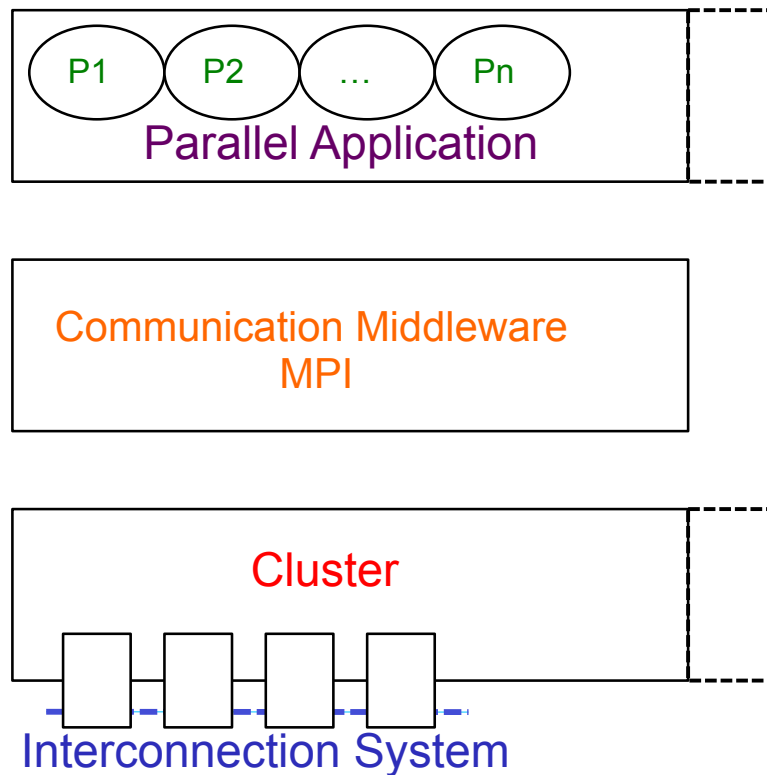
8



- One of most communication middleware used is the standard **MPI** (Message Passing Interface).
- Different implementations:
 - **MPICH**
 - OpenMPI
 - LAM
 - CHIMP-MPI ...

Problem description: trend

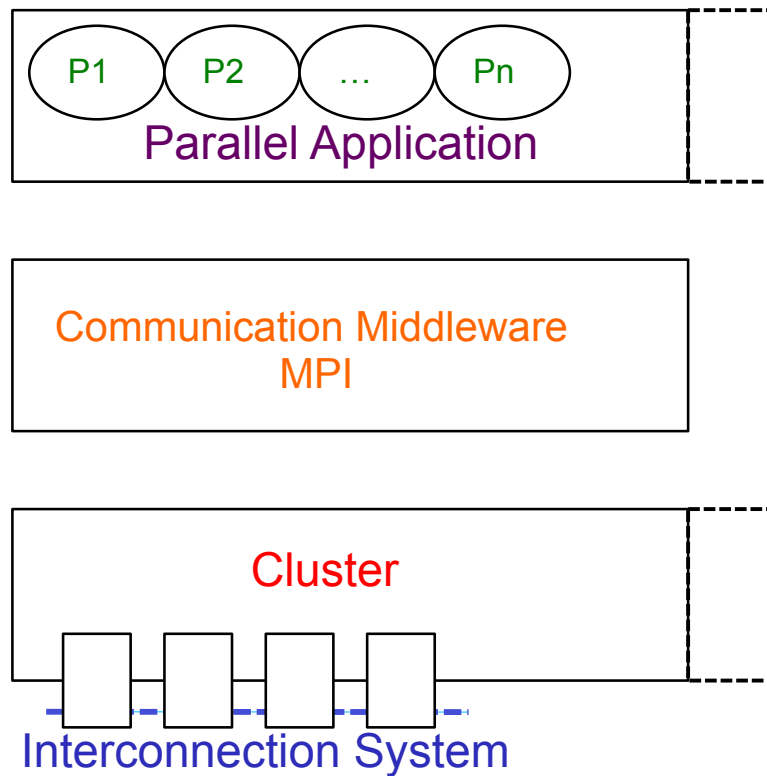
9



- Multicore processors provides a flexible way to **increase the computation capability of cluster**
- System performance may be improved with multicore **but** bottleneck from other components could reduce the scalability.

Problem description: Bottleneck

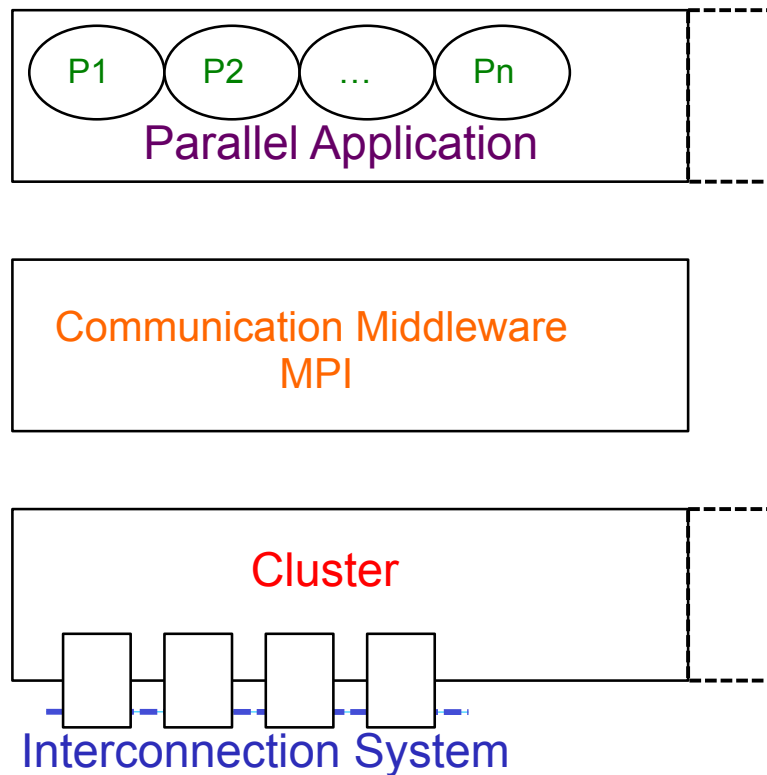
10



- Bottleneck :
 - I/O subsystem
 - Communication subsystem

Problem description: Bottleneck

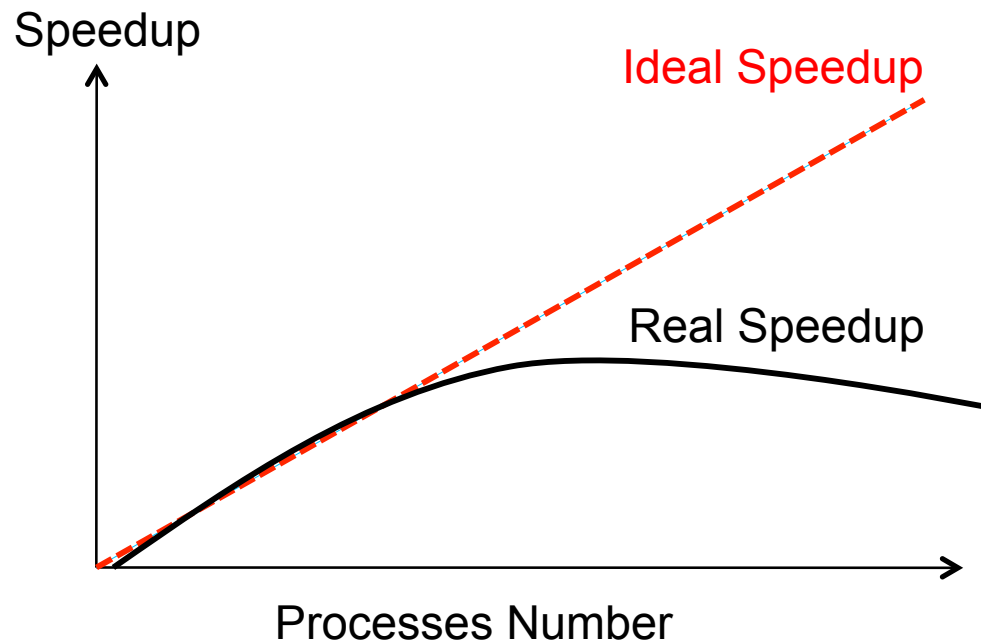
11



- **Communication Bottleneck:**
 - Network used are very fast and low latency.
 - Computational capability in multicore very high.
 - The frequency of message increase a lot → insufficient the increase of the bandwidth and latency

Problem description: Bottleneck

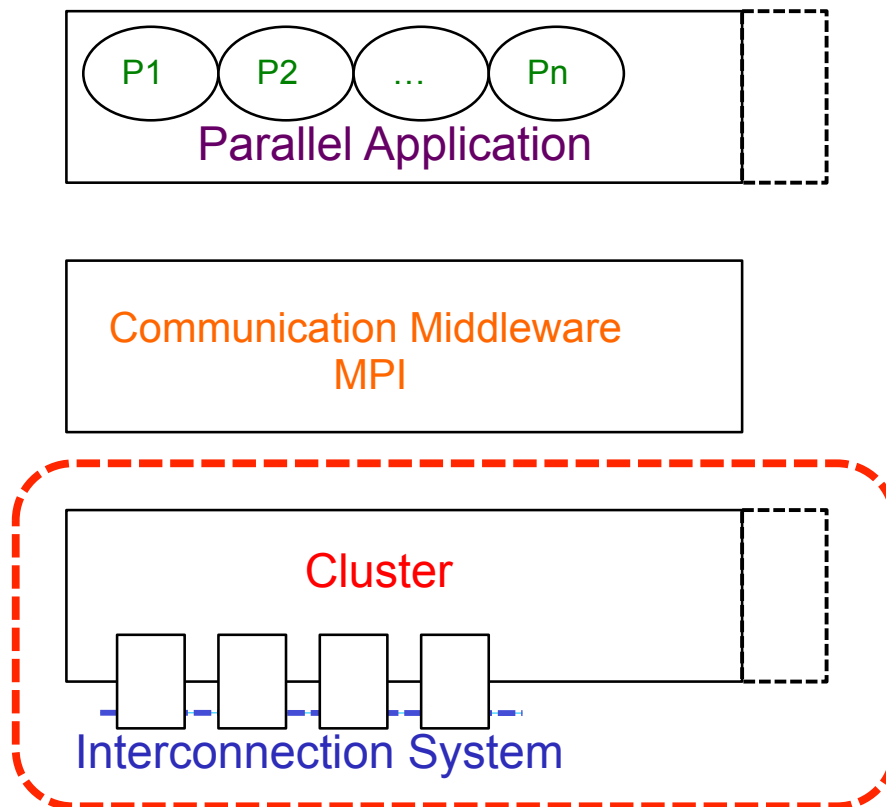
12



- Bottleneck:
 - Scalability problem
 - Performance problem

Problem description: possible solutions

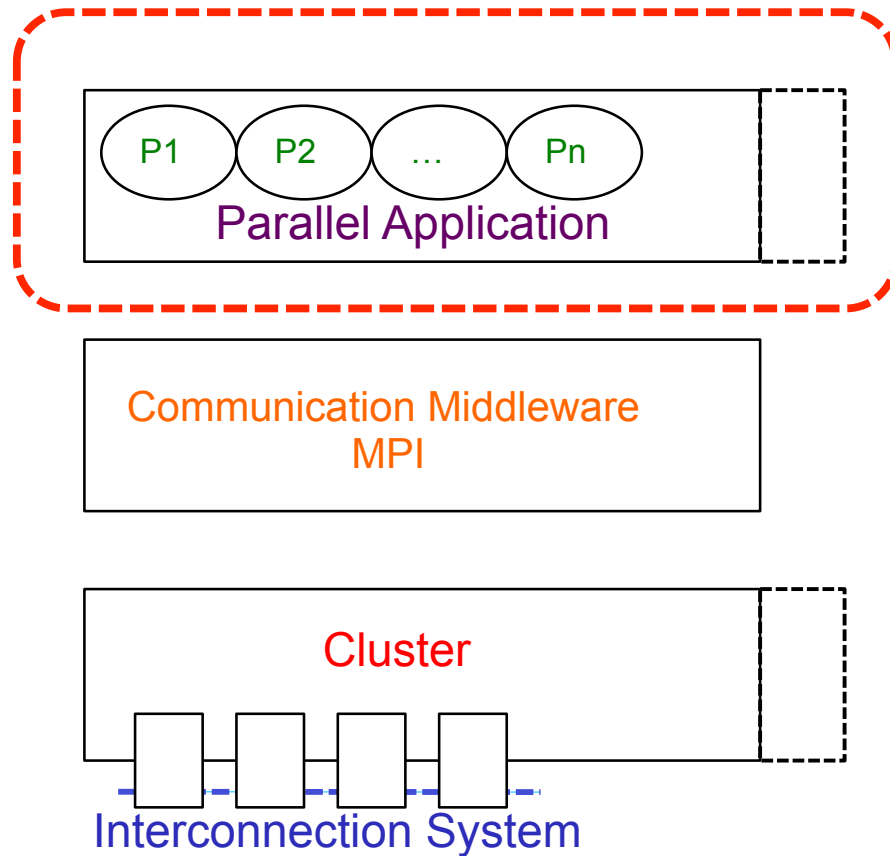
13



- 1) Improve network
 - Expensive.
 - Limited to current technology

Problem description: possible solutions

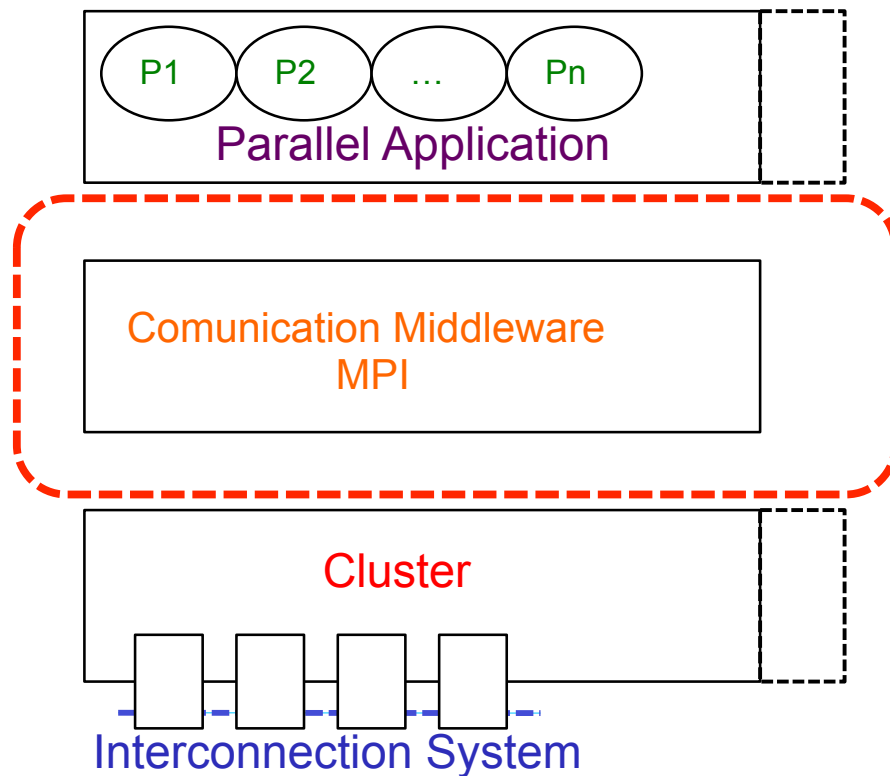
14



- 2) Improve the applications:
 - More effort in the design
 - Not always possible
 - The improvement affects few users

Problem description: possible solutions

15



- 3) Improve the communication middleware
 - Portability
 - Greater user benefited
 - Lower Cost
 - Transparent:
 - Users
 - Applications

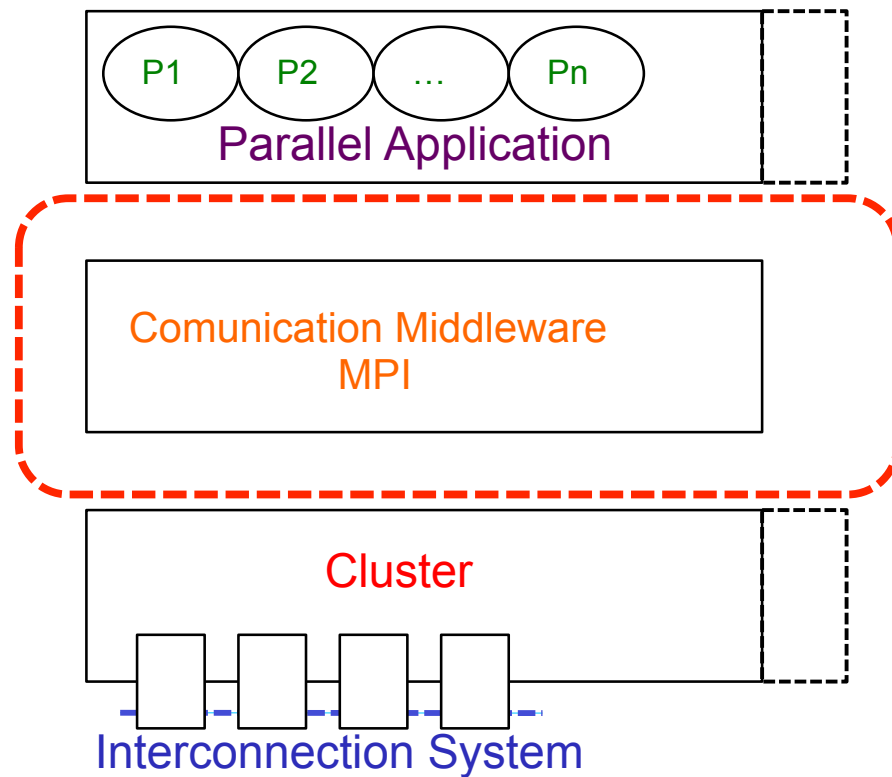
Summary

16

1. Problem description
2. Main Objectives
3. Strategies for improve the performance of collective I/O operation
4. Strategies for improve the performance of communication operations
5. Evaluations

Main objectives

17



- Improve the scalability and the performance of MPI based applications executed on Multicore cluster
- **How?** Improving the Middleware MPI

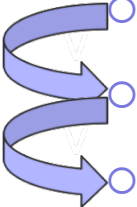
Specific objective

18

- Reduction of transferred data volume in communications:
 - Strategy to reduce the cost of communication in MPI by using lossless compression techniques and passing interface profiling (PMPI)

Phd. Thesis Proposal: Communication Compression

19

- Reduce the cost of communications:
 - By MPI messages compression in **run-time**
- Lossless compressions algorithms
- Compress **all** MPI primitives.
- We have developed three different strategies:
 - ○ **Runtime Compression (RC)**
 - **Runtime Adaptive Compression Strategy (RAS)**
 - **Guided Strategy (GS)**
- Implementation of the strategies by modifying the source code of MPICH1.2

HPC-Programme Proposal

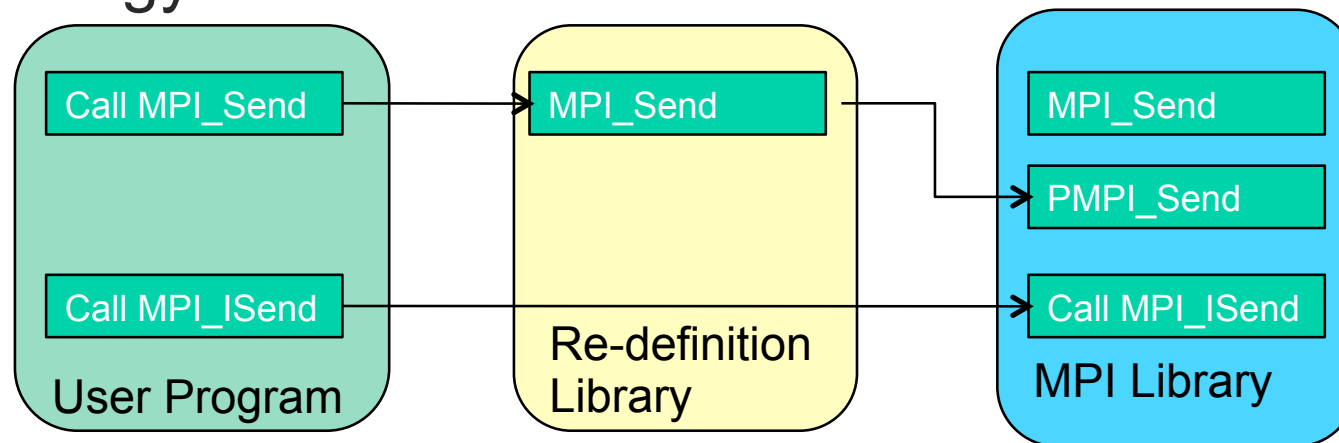
20

- Implement Runtime Adaptive Compression (RAS) by using (New!!) message passing interface profiling (PMPI).
- Why ??
 - HECToR → No install own MPICH
 - DEISA/PRACE Spring School: Tools and Techniques for Extreme Scalability
 - The Scalasca Performance Analysis Toolset → PROFILING!!
 - New idea: Use the MPI standard profiling interface (PMPI) to implement the adaptive compression strategy

PMPI

21

- Each standard MPI function can be called with an MPI_ or PMPI_ prefix.
- PMPI such wrapper functions to customize MPI behavior.
- We are going to use PMPI to implement RAS strategy



Advantages of PMPI

22

- PMPI allows replacement of MPI routines at link time (not need to recompile)
 - No modification of the source code of the MPI implementation
 - No modification of the source code of the application
- Portable, independent of the MPI implementation (XT MPI, MPICH2, OPENMPI ...).

Example of Use of Profiling Interface

23

```
// extern.c
int MPI_Send( void *start, int count, MPI_Datatype datatype,
int dest, int tag, MPI_Comm comm )
{
    printf ("Before send the message to process %d\n",dest);
    return PMPI_send(start, count, datatype, dest, tag, comm);
}
```

```
// my_application.c
if (my_rank==0)
{
    for (i=0;i<5;i++)
        array[i]=i;
    for (j=1;j<num_processes;j++)
        MPI_Send( array,5,MPI_INT,i,tag,MPI_COMM_WORLD);
}
```

```
> mpicc -c extern.c
> mpicc -c my_application.c

> mpicc -g my_application.o extern.o -o executable
```

Example of Use of Profiling Interface

24

```
> mpirun -np 10 ./executable > output.txt  
> cat output.txt
```

```
Before send the message to process 1  
Before send the message to process 2  
Before send the message to process 3  
Before send the message to process 4  
Before send the message to process 5  
Before send the message to process 6  
Before send the message to process 7  
Before send the message to process 8  
Before send the message to process 9
```


RAS: Runtime Adaptive Compression Strategy

25

- Runtime Adaptive Compression Strategy (RAS), per message transferred takes two decision:
 - Turn on and off the compression.
 - Select itself the best compression algorithm:
 - LZO, RLE, HUFFMAN, RICE, FPC.
- Learn in run-time from previous messages
- Decision depending on:
 - Message feature:
 - Datatype and length
 - Network performance:
 - Latency and bandwidth
 - Compression algorithms

RAS Decisions → Speedup

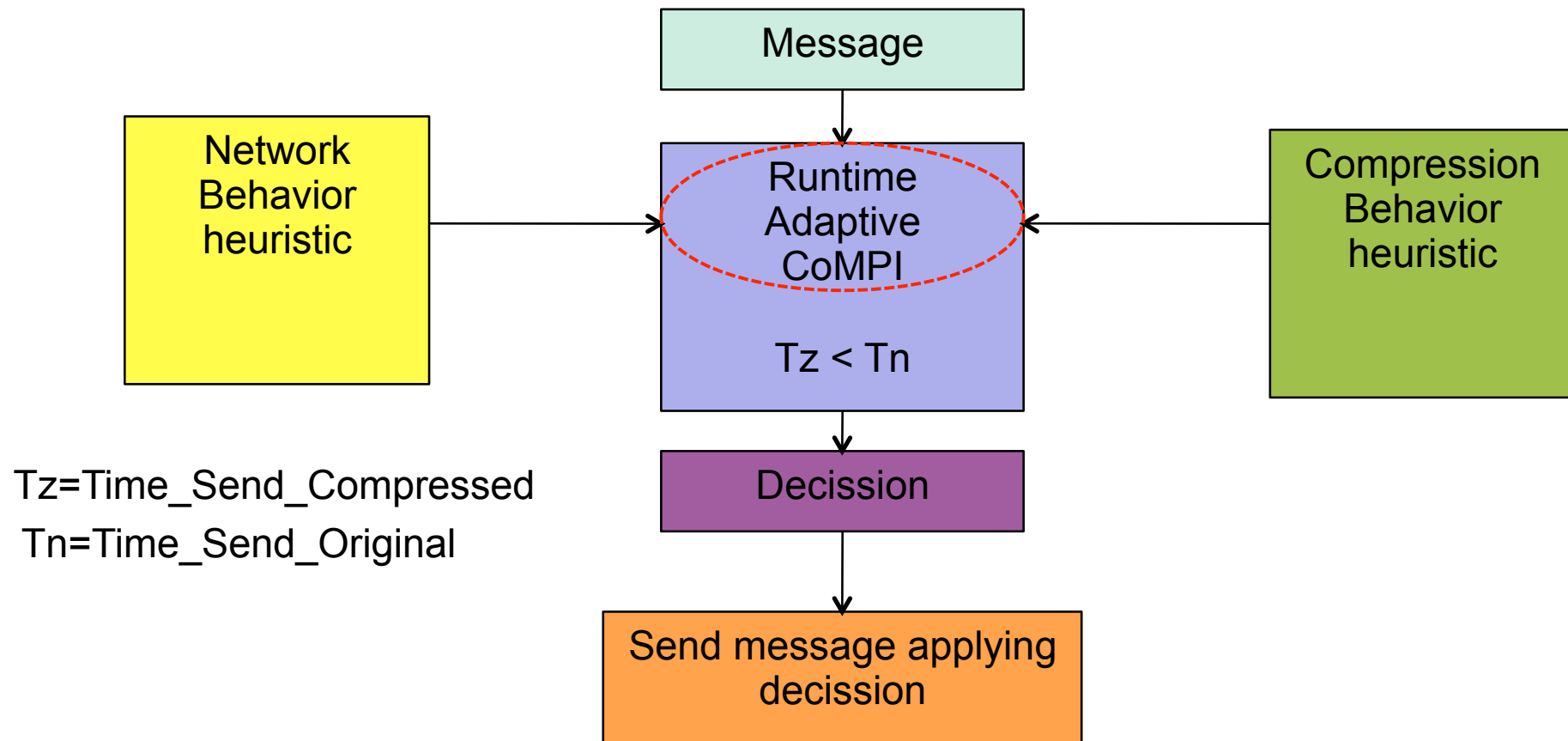
26

- Speedup to decide if send the message with/without compression.
- So the decision depends:
 - Original message transmission time
 - Compressed message transmission time
 - Compression and decompression time

$$Speedup = \frac{Time_Sent_Orig.}{(Time_Sent_Compr.+ time_compress.+ time_decompr.)}$$

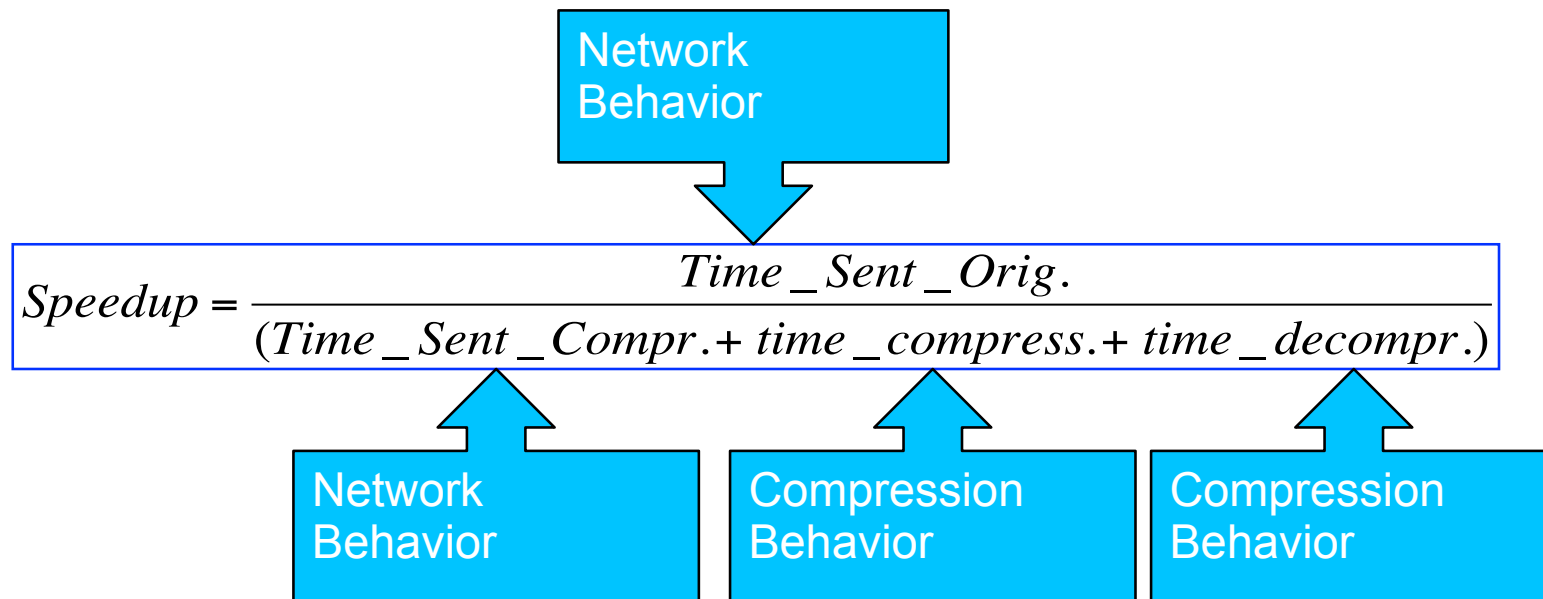
RAS Decisions → Speedup

27



RAS Compression behavior

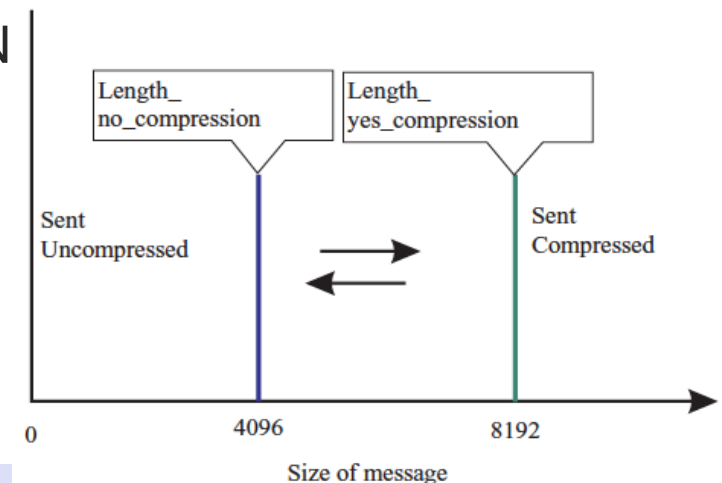
28



RAS Decision Methodology

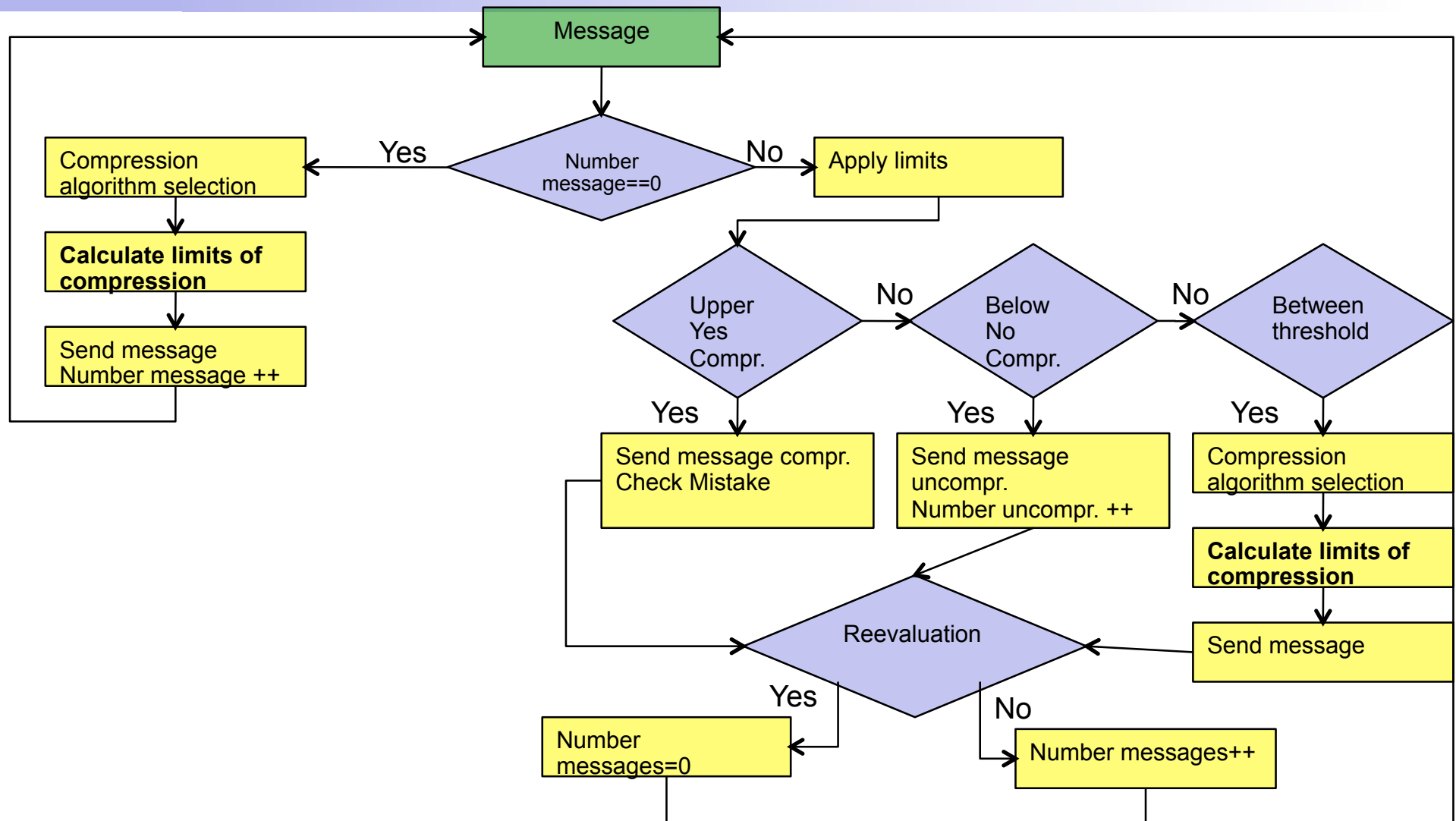
29

- Calculate the speedup per message? No → **high overhead computation time**
- According to Compression Behavior and Network data Behavior, RAS decides:
 - Datatype:
 - Integer y Float → LZO
 - Double → LZO or FPC
 - Others → LZO, RLE, RICE or HUFFMAN
 - Message size → Decision Threshold:
 - Each datatype has its thresholds
 - Length_yes_compression
 - Length_no_compression



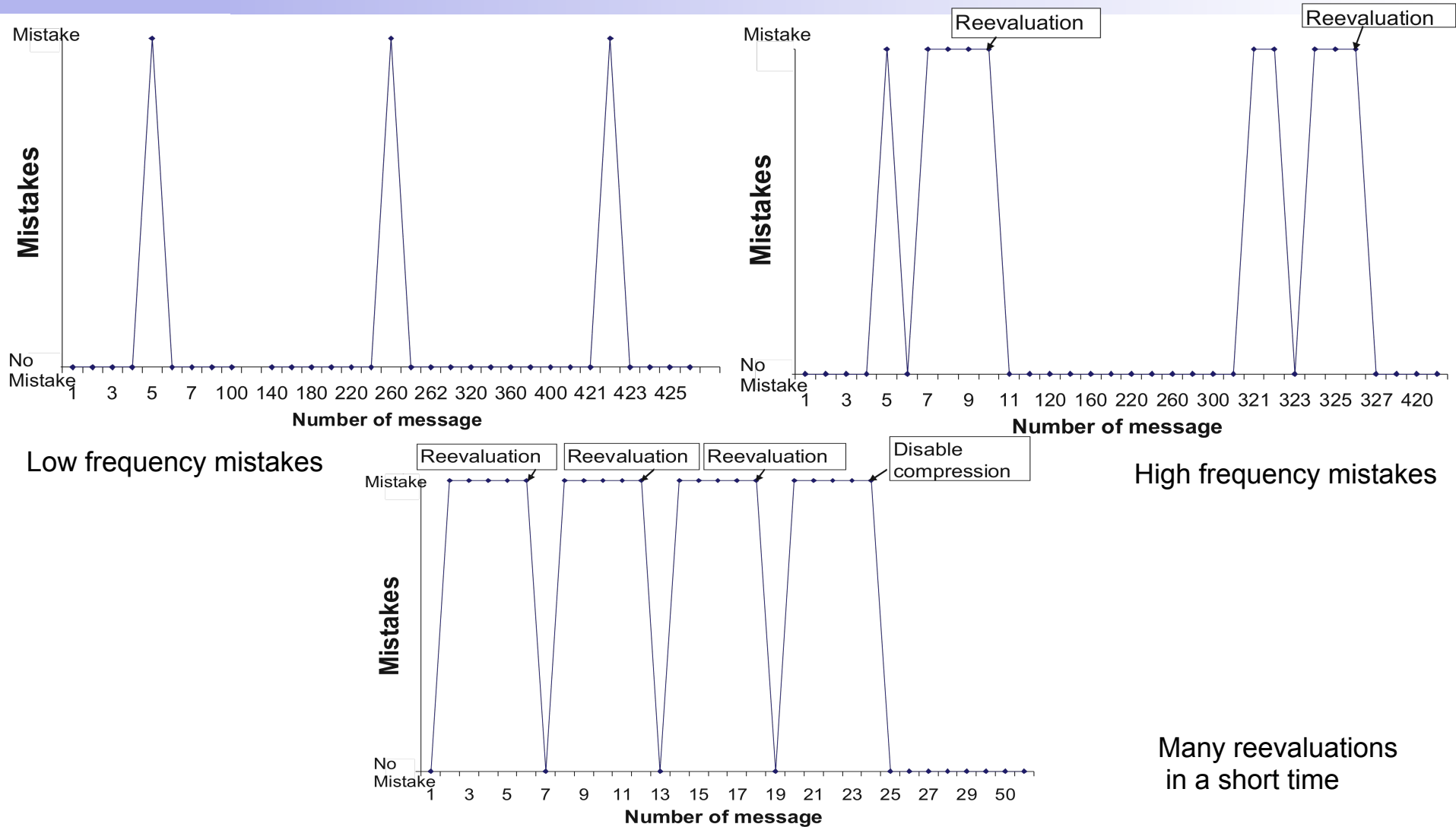
RAS Decision Methodology

30



Different cases of re-evaluation

31



PRAcTICaL-MPI technique

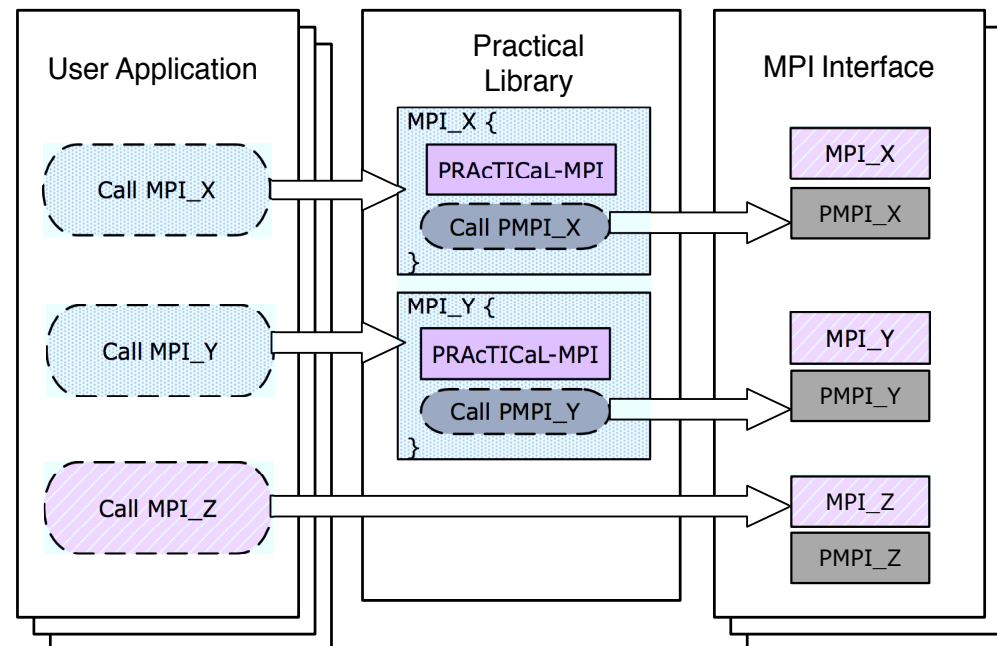
32

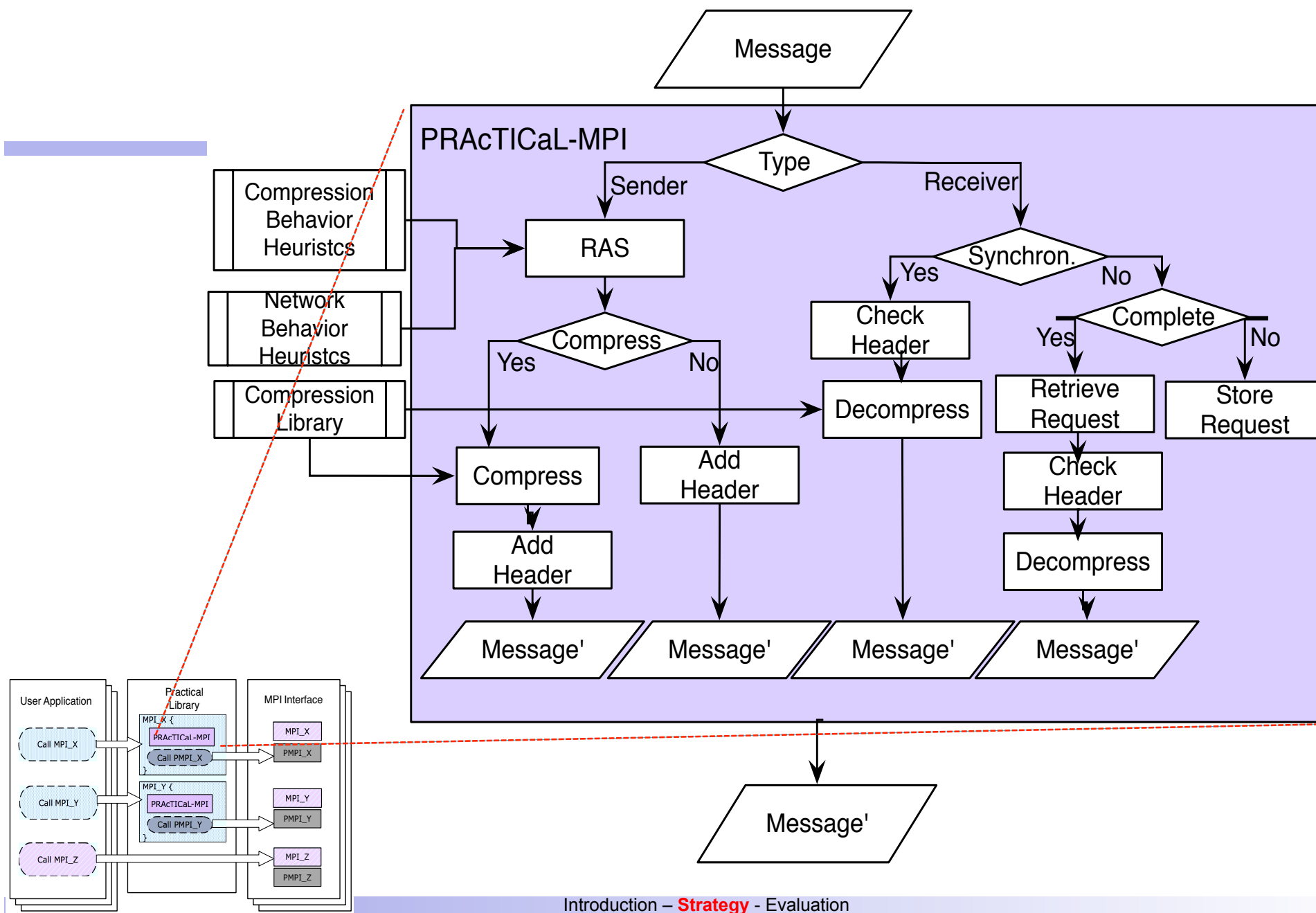
- The PRAcTICaL-MPI technique, is an optimization of MPI communications that exploits PMPI to apply Runtime Adaptive Compression Strategy (RAS) thus reducing the volume of communications.
- PRAcTICaL-MPI is portable: Can be used with any MPI implementation, not just a with a specific MPI implementation.
- PRAcTICaL-MPI is transparent for both applications and MPI implementations, because it can be applied without changing their source code in any way.
- The only requirement:
 - Relink the application with the Practical library to include our adaptive compression functionality.

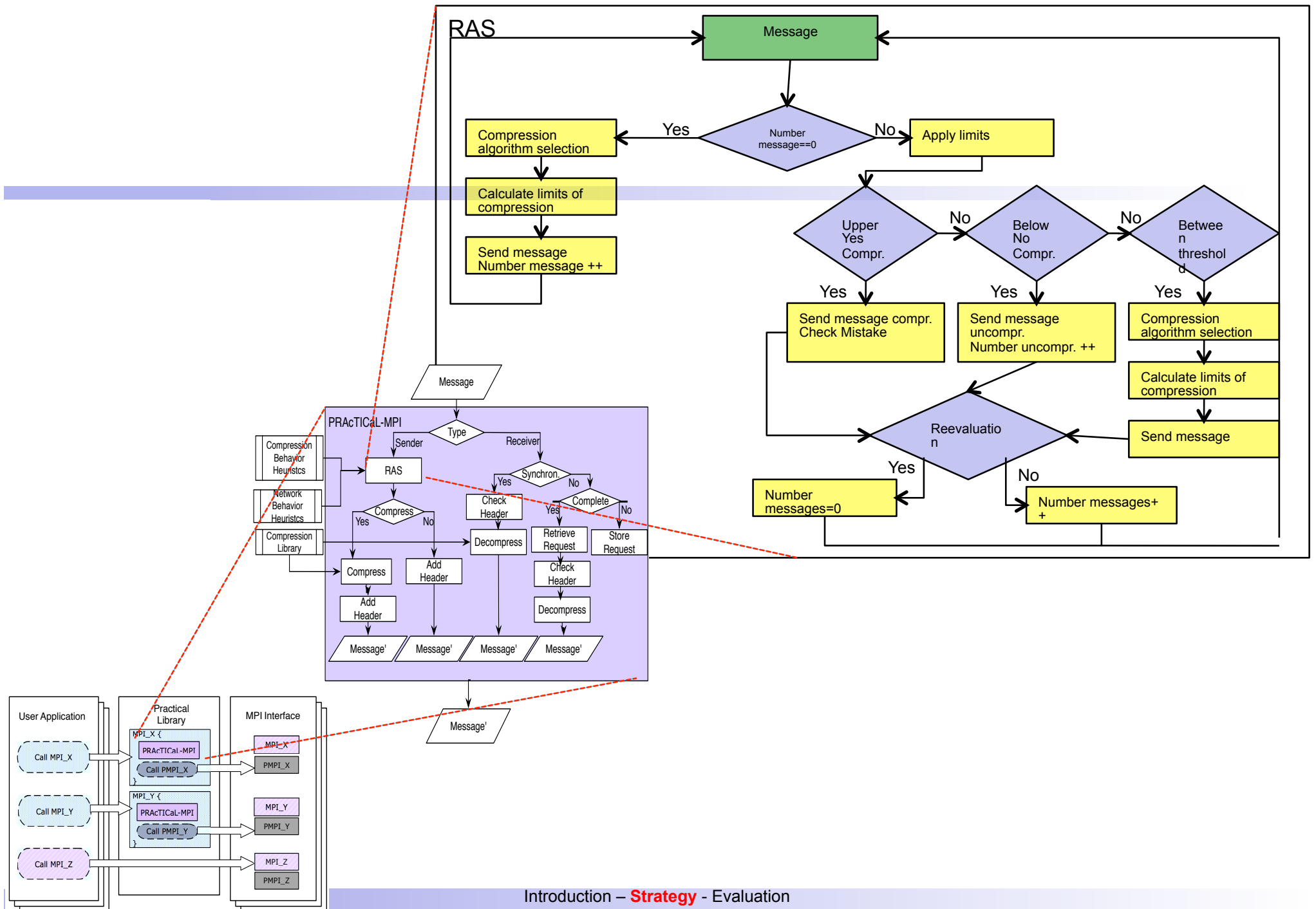
PRAcTICaL-MPI architecture

33

- PMPI intercepts the MPI calls and wraps the PRAcTICaL- MPI technique around the actual MPI library invocation.
- Practical Library:
 - The most common routines of point-to-point and collective communications are wrapped.

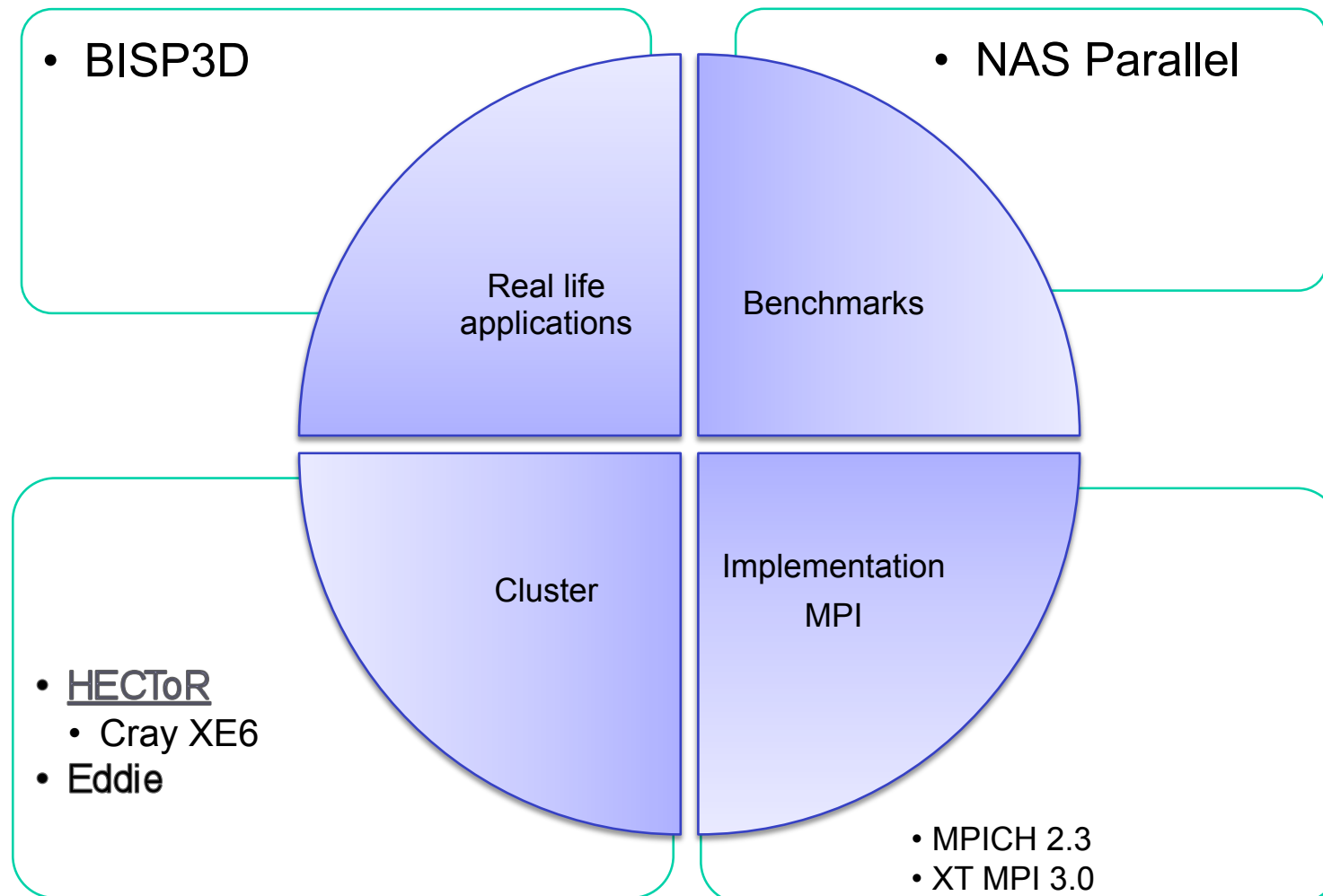






HPC Programme

36



HECToR

37

High End Computing Terascale Resource

“HECToR is the UK's high-end computing resource, funded by the UK Research Councils. It is available for use by academia and industry in the UK and Europe.”

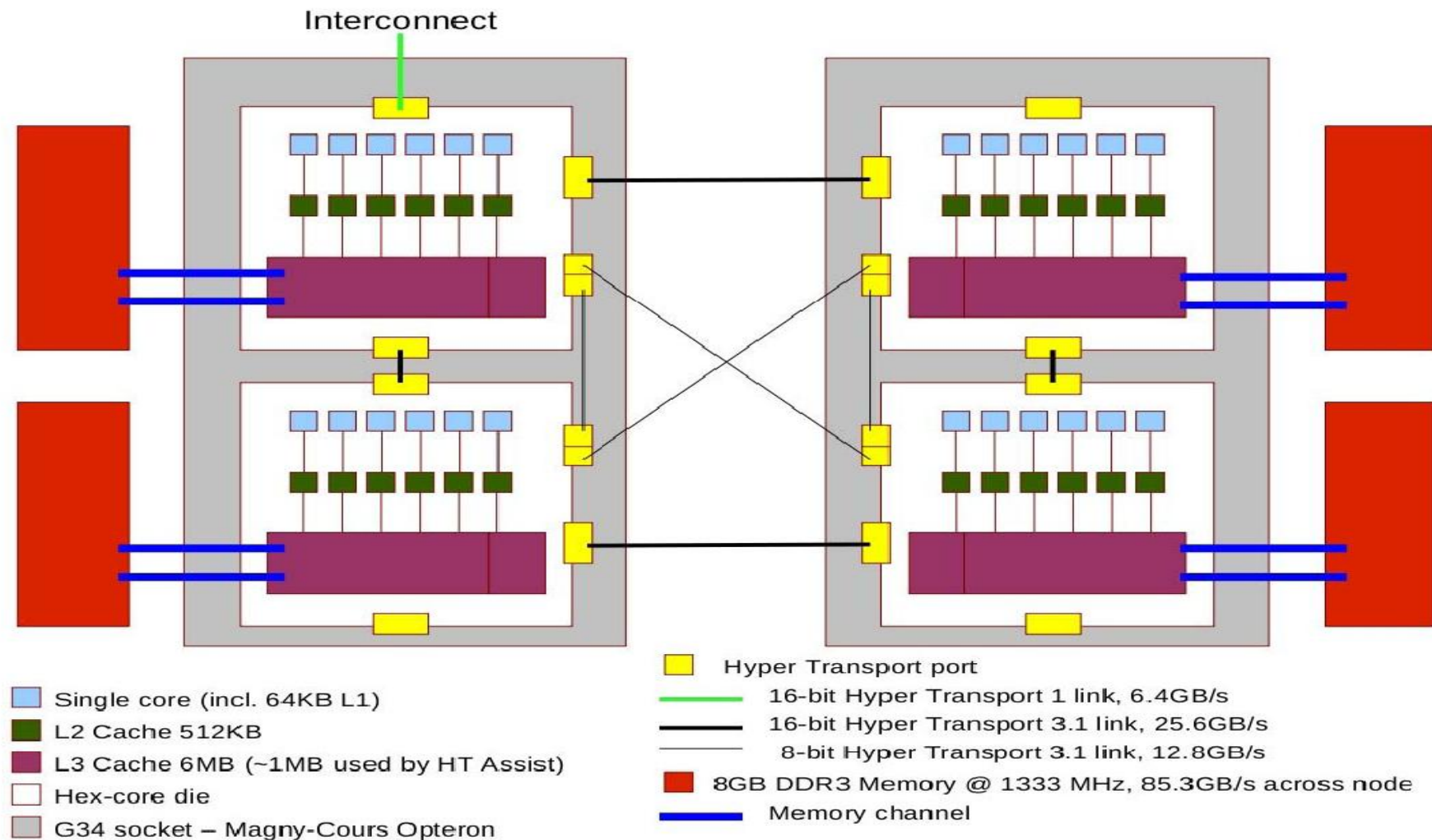
Features on HECToR Phase 2b

38

- 1856 compute nodes which contain two AMD 2.1 GHz 12-core Opteron processors => 44,544 cores
- Theoretical peak performance of 373 Tflops
- 32 GB main memory per processor, shared between 24 cores => total memory of 58 TB
- Gemini interconnect
- 12 IO nodes

XE6 24-core Magny Cours node

39



BISP3D

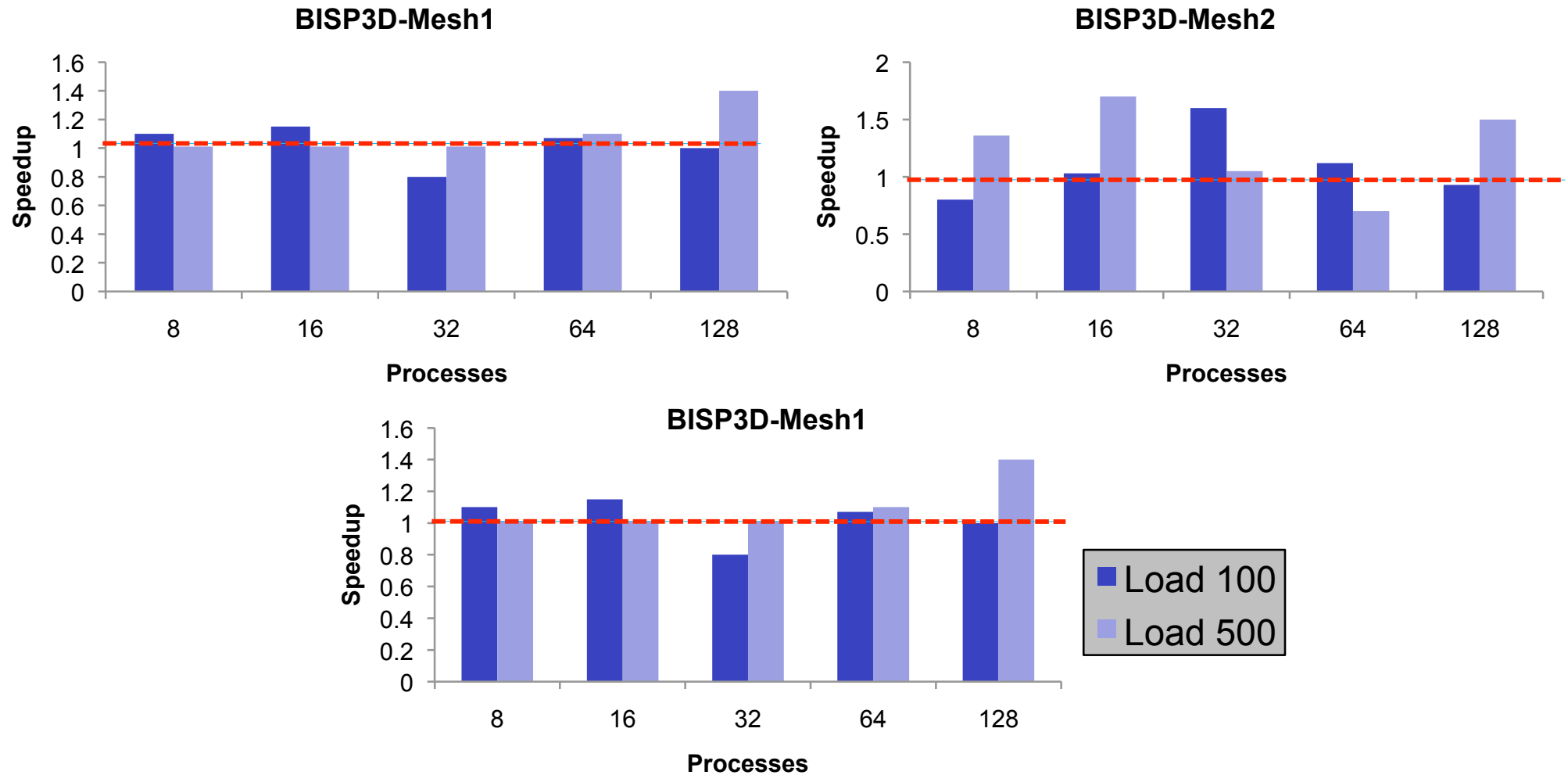
40

- 3-Dimensional simulator of BJT and HBT bipolar devices:
 - The goal is to relate electrical characteristics of the device with its physical and geometrical parameters.

- We have use 3 different different devices
 - Each bipolar device it is represented by a mesh.
 - Load represent the number of elements per node (in a mesh).

Evaluations of PRAcTICAL-MPI

41



Conclusions

42

- The speedups achieved in 90% of the scenarios are greater than or equal to one due to :
 - PRAcTICaL-MPI applying run-time compression to reduce the volume of the messages with the best algorithm per message, thus reducing execution time.
 - PRAcTICaL-MPI deactivates the compression when it is not worthwhile applying any compression.
- Scalability is enhanced with PRAcTICaL-MPI.

Outcomings of HPC-Europa 2

43

- Publication in EuroPar-2012 conference:
 - An adaptive, scalable, and portable technique for speeding up MPI-based applications

- Post-Doctoral position in DIR group, University of Edinburgh:
 - Hazard forecasting in real time: from controlled laboratory tests to volcanoes and earthquakes

PRAcTICaL-MPI: **P**ortable **A**daptive **C**ompression Library **MPI**

Thanks to HPC-Europa 2

Author: Rosa Filgueira Vicente
University of Edinburgh
rosa.filgueira@ed.ac.uk

