

UNDERSTANDING SUPERCOMPUTING

UNDERSTANDING SUPERCOMPUTING with Marenostrium Supercomputer

**Supercomputer Architecture course -
Master in Innovation and Research
in Informatics (MIRI) - Specialization
High Performance Computing**

Jordi Torres i Viñals

Universitat Politècnica de Catalunya (UPC Barcelona Tech)
Barcelona Supercomputing Center (BSC-CNS)

Fall 2016

WATCH THIS SPACE

WATCH THIS SPACE COLLECTION - BARCELONA

Copyright ©2016 by Lulu Press, Inc.

Lulu Press, Inc.
Raleigh, North Carolina, United States
<http://www.lulu.com>

WATCH THIS SPACE Collection
Cover illustration: Supercomputer Marenstrum - Torre Girona chapel

Citation:
UNDERSTANDING SUPERCOMPUTING
with Marenstrum Supercomputer in Barcelona
Jordi Torres
Lulu Press, Inc, September 2016
ISBN 978-1-365-37682-5

First english edition: September 2016.

Author:
Jordi Torres
Universitat Politècnica de Catalunya - UPC Barcelona Tech
www.JordiTorres.Barcelona

This book is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License (CC BY-NC-SA 3.0). In short: Jordi Torres retains the Copyright but you are free to reproduce, re-blog, remix and modify the content only under the same license to this one. You may not use this work for commercial purposes but permission to use this material in nonprofit teaching is still granted, provided the authorship and licensing information here is displayed.

*To the students of the
SA-MIRI master course at
UPC Barcelona Tech, for
inspiring me to develop
the material for this book.*

CONTENTS IN BRIEF

1	Supercomputing Building Blocks: Marenstrum Visit	1
2	Getting Started with Supercomputing	15
3	Getting Started with Parallel Programming Models	31
4	Getting Started with Parallel Performance Metrics	47
5	Getting Started With Parallel Performance Models	65
6	Getting Started with Performance Analysis Tools	95

CONTENTS

List of Figures	xiii
List of Tables	xvii
Preface	xxi
Introduction	xxiii
I.1 Supercomputing: hierarchical structure of abstraction view	xxiii
I.2 Parallel Programming Motivation	xxiv
I.3 Other types of Parallel Computing	xxvii
I.4 How to measure the supercomputing power	xxix
I.5 Supercomputing and its challenges	xxx
1 Supercomputing Building Blocks: Marenstrum Visit	1
1.1 Hardware components overview	1
1.2 Compute nodes	3
1.3 Interconnection networks for Supercomputers	6
1.4 High Performance Storage Systems	8
1.5 System Software Stack	10
1.6 Power and Cooling System	12
	ix

2	Getting Started with Supercomputing	15
2.1	How to access MareNostrum III	16
2.1.1	How to get an account	16
2.1.2	Connecting to MareNostrum III	16
2.2	MareNostrum File Systems	19
2.2.1	MareNostrum File Systems	19
2.2.2	Data Transfer Machine	20
2.2.3	Mid-long term storage filesystem	20
2.3	Transferring files in MareNostrum III	22
2.4	Serial compilation and execution in C language	23
2.5	Batch Scheduling System	24
2.5.1	LSF	24
2.5.2	Basic Commands	24
2.5.3	Additional BSC Commands	25
2.5.4	Job Directives	25
2.5.5	LSF Hello World	27
3	Getting Started with Parallel Programming Models	31
3.1	Parallel Programming Models	32
3.2	Message-Passing Interface: MPI	32
3.2.1	MPI module environment	33
3.2.2	Getting started with MPI	35
3.3	Getting started with OpenMP	37
3.3.1	OpenMP Programming Model	37
3.3.2	Compilers for OpenMP	38
3.4	Case Study: Trapezoidal Rule	39
3.4.1	Trapezoidal Rule	39
3.4.2	MPI version	42
3.4.3	OpenMP version	44
4	Getting Started with Parallel Performance Metrics	47
4.1	Performance Metrics	48
4.1.1	Speedup	48
4.1.2	Efficiency	49
4.1.3	Scalability	52
4.2	Taking timing	54
4.2.1	What time	54
4.2.2	gettimeofday POSIX function	55

4.2.3	Variability in timing	57
4.3	Case Study: Trapezoidal Rule	58
4.3.1	OpenMP parallelism	58
4.3.2	MPI parallelism	59
4.3.3	Speedup and Efficiency of MPI version	60
4.3.4	Speedup and Efficiency of OpenMP version	62
5	Getting Started With Parallel Performance Models	65
5.1	Performace model	65
5.1.1	Parallel Overhead	66
5.1.2	Speedup	67
5.1.3	Efficiency	67
5.2	Advanced MPI parallelism	68
5.2.1	Programming Collective Communication	68
5.2.2	Case Study: MPI Matrix-vector product	68
5.2.3	Assumptions in this code	71
5.3	Amdahl's Law	72
5.3.1	Amdahl's Law equation	72
5.3.2	Asimptotic speedup	74
5.3.3	Amdahl's Law limitations	74
5.4	Gustafson-Barsis's Law	76
5.4.1	Problem size is not constant	76
5.4.2	Gustafson-Barsis's Law equation	78
5.5	Parallel overhead	79
5.6	Case Study: Average matrix-vector product	81
5.6.1	A balance between computation and communication	81
5.6.2	Sequential version	81
5.6.3	Parallel version	83
5.6.4	Building a testbed	85
5.7	Evaluation	87
5.7.1	Taking Timing	87
5.7.2	Performance Metrics	90
5.7.3	Gustafson-Barsis's Law	92
5.7.4	Cost of sharing resources in the same node	94
6	Getting Started with Performance Analysis Tools	95
6.1	Performance of a Compiled Program	96
6.2	Basic linux command line tools	96

6.2.1	gettimeofday() limitations	96
6.2.2	top	97
6.2.3	vmstat	98
6.2.4	perf	99
6.3	Analysing Workflow models	101
6.4	BSC Advanced Performance Analysis Tools	102
6.5	PARAVER	104
6.5.1	Overview	104
6.5.2	Configuration files	104
6.5.3	Trace files	104
6.5.4	Tool Structure	105
6.5.5	Install the Paraver binaries in your laptop	105
6.5.6	Configure the Paraver package	106
6.6	Load traces	106
6.6.1	Timeline Display	106
6.6.2	Main Paraver Window	108
6.7	Obtaining useful information	108
6.7.1	Instructions per cycle	108
6.7.2	Cache misses ratio	111
6.7.3	Duration of computation and message size	111
6.8	Statistics Display	112
6.8.1	2D Analyzer	112
6.8.2	Data Window	113
6.9	Histograms	114
6.10	Other components	115
6.11	Extrae	116
6.11.1	Extrae in Marenostum	116
6.11.2	JSF scripts	117
6.11.3	Quick step by step trace generation recipe	118
6.12	Case Study: Averaged Matrix-Vector product	122
6.12.1	Trace Generation	122
6.12.2	Paraver Analysis	123
	Acknowledgments	133
	References	135
	Index	137