E-Infrastructures
H2020-EINFRA-2014-2015

EINFRA-4-2014: Pan-European High Performance Computing Infrastructure and Services

PRACE-4IP

PRACE Fourth Implementation Phase Project

Grant Agreement Number: EINFRA-653838

D7.6
Best Practice Guides for New and Emerging Architectures

Final

Version: 1.0
Author(s): Volker Weinberg, BADW-LRZ
Alan Gray, EPCC
Ole Widar Saastad, UiO
Date: 23.01.2017

Project and Deliverable Information Sheet

PRACE Project	**Project Ref. №: EINFRA-653838**	
	Project Title: PRACE Fourth Implementation Phase Project	
	Project Web Site: http://www.prace-project.eu	
	Deliverable ID: **D7.6**	
	Deliverable Nature: Report	
	Dissemination Level: PU *	**Contractual Date of Delivery:** 31 / January / 2017
		Actual Date of Delivery: 31 / January / 2016
	EC Project Officer: Leonardo Flores Añover	

* - The dissemination levels are indicated as follows: **PU** – Public, **CO** – Confidential, only for members of the consortium (including the Commission Services) **CL** – Classified, as referred to in Commission Decision 2991/844/EC.

Document Control Sheet

Document	**Title: Best Practice Guides for New and Emerging Architectures**	
	ID: **D7.6**	
	Version: 1.0	**Status:** ***Final***
	Available at: http://www.prace-project.eu	
	Software Tool: Microsoft Word 2016	
	File(s): D7.6	
Authorship	**Written by:**	Volker Weinberg, BADW-LRZ Alan Gray, EPCC Ole Widar Saastad, UiO
	Contributors:	Momme Allalen, BADW-LRZ Emanouil Atanassov, IICT-BAS Vali Codreanu, SURFsara Jörg Hertzer, USTUTT-HLRS Nevena Ilieva-Litova, NCSA Tomáš Karásek, VSB-TUO Christian Morales, BSC Mikael Rännar, SNIC-UmU Jorge Rodriguez, BSC Michael Schliephake, KTH Martin Stachoň, VSB-TUO Janko Strassburg, BSC
	Reviewed by:	Michaela Barth, SNIC/KTH Florian Berberich, JUELICH
	Approved by:	MB/TB

Document Status Sheet

Version	Date	Status	Comments
0.1	16/November/2016	Draft	First Draft
0.2	10/January/2017	Draft	Included updates by coauthors.
1.0	23/January/2017	Final version	Update after internal review.

Document Keywords

Keywords:	PRACE, HPC, Research Infrastructure, Best Practice Guide, Intel® Xeon Phi™, Knights Landing, Haswell, Broadwell, GPGPU

Table of Contents

List of Figures

References and Applicable Documents

[1] http://www.prace-project.eu (identical to http://www.prace-ri.eu)

[2] Emanouil Atanassov, Michaela Barth, Mikko Byckling, Vali Codreanu, Nevena Ilieva, Tomáš Karásek, Jorge Rodriguez, Sami Saarinen, Ole Widar Saastad, Michael Schliephake, Martin Stachoň, Janko Strassburg, and Volker Weinberg (ed.), *Best Practice Guide – Intel® Xeon Phi™ v2.0*, produced by PRACE-4IP, January 2017. Available at the PRACE RI web site [1] as: http://www.prace-ri.eu/IMG/pdf/Best-Practice-Guide-Intel-Xeon-Phi.pdf, http://www.prace-ri.eu/Best-Practice-Guide-Intel-Xeon-Phi-HTML

[3] Vali Codreanu, Jorge Rodriguez, and Ole Widar Saastad (ed.), *Best Practice Guide – Knights Landing*, produced by PRACE-4IP, January 2017. Available at the PRACE RI

web site [1] as:
http://www.prace-ri.eu/IMG/pdf/Best-Practice-Guide-Knights-Landing.pdf,
http://www.prace-ri.eu/Best-Practice-Guide-Knights-Landing-HTML

[4] Vali Codreanu, Joerg Hertzer, Christian Morales, Jorge Rodriguez, Ole Widar Saastad, Martin Stachoň, and Volker Weinberg (ed.), *Best Practice Guide – Haswell/Broadwell*, produced by PRACE-4IP, January 2017. Available at the PRACE RI web site [1]
http://www.prace-ri.eu/IMG/pdf/Best-Practice-Guide-Haswell.pdf,
http://www.prace-ri.eu/Best-Practice-Guide-Haswell-HTML

[5] Momme Allalen, Alan Gray (ed.), Nevena Ilieva-Litova, Anders Sjöström, Vali Codreanu, and Volker Weinberg, *Best Practice Guide – GPGPU,* produced by PRACE-4IP, January 2017. Available at the PRACE RI web site [1] as:
http://www.prace-ri.eu/IMG/pdf/Best-Practice-Guide-GPGPU.pdf ,
http://www.prace-ri.eu/Best-Practice-Guide-GPGPU-HTML

[6] Michaela Barth, Mikko Byckling, Nevena Ilieva, Sami Saarinen, Michael Schliephake, and Volker Weinberg (ed.), *Best Practice Guide – Intel Xeon Phi*, produced by PRACE-3IP, December 2013. Peplaced by [2] at the PRACE RI web site [1].

[7] Alan Gray, Anders Sjöström, Nevena Ilieva-Litova, and Maciej Szpindler (ed.), *Best Practice Guide – GPGPU*, PRACE-2IP D7.3, May 2013. Replaced by [7] at the PRACE RI web site [1].

[8] Jacques David, Jeroen Engelberts, Xu Guo, Florian Janetzko, and Walter Lioen, *Petascaling and Optimisation Guides for PRACE Systems*, PRACE-1IP D7.3, June 2012. Available at the PRACE RI web site [1] as:
http://www.prace-ri.eu/IMG/pdf/d7.3_1ip.pdf.

[9] Vegard Eide, Walter Lioen, Maciej Szpindler, and Volker Weinberg, *Petascaling and Optimisation for Tier-1 Architectures*, PRACE-2IP D7.3, May 2013. Available at the PRACE RI web site [1] as:
http://www.prace-ri.eu/IMG/pdf/d7.3.pdf.

[10] Nikos Anastopoulos, Maciej Cytowski, Mark Filipiak, Walter Lioen, Philippe Wautelet, and Volker Weinberg, *Best Practice Guides for New and Emerging Architectures*, PRACE-3IP D7.3.1, December 2013. Available at the PRACE RI web site [1] as:
http://www.prace-ri.eu/IMG/pdf/d7.3.1_3ip.pdf

[11] http://www.docbook.org/

[12] http://tldp.org/LDP/LDP-Author-Guide/html/docbook-why.html

List of Acronyms and Abbreviations

aisbl	Association International Sans But Lucratif (legal form of the PRACE-RI)
BCO	Benchmark Code Owner
Byte	8 bits (Octet)
CoE	Center of Excellence
CPU	Central Processing Unit
CUDA	Compute Unified Device Architecture (NVIDIA)
DARPA	Defense Advanced Research Projects Agency
DEISA	Distributed European Infrastructure for Supercomputing Applications EU project by leading national HPC centres
DoA	Description of Action (formerly known as DoW)
EC	European Commission
EESI	European Exascale Software Initiative

EoI	Expression of Interest
ESFRI	European Strategy Forum on Research Infrastructures
GiB	Gibi (= 2^{30}) Bytes
GB	Giga (= 2^{30} ~ 10^9) Bytes (= 8 bits), also GByte
Gb/s	Giga (= 10^9) bits per second, also Gbit/s
GB/s	Giga (= 10^9) Bytes (= 8 bits) per second, also GByte/s
GiB/s	Gibi (= 2^{30}) Bytes per second
GÉANT	Collaboration between National Research and Education Networks to build a multi-gigabit pan-European network. The current EC-funded project as of 2015 is GN4.
GFlop/s	Giga (= 10^9) Floating point operations (usually in 64-bit floating point, i.e. DP) per second, also GF/s
GHz	Giga (= 10^9) Hertz, frequency =10^9 periods or clock cycles per second
GPU	Graphic Processing Unit
HET	High Performance Computing in Europe Taskforce. Taskforce by representatives from European HPC community to shape the European HPC Research Infrastructure. Produced the scientific case and valuable groundwork for the PRACE project.
HMM	Hidden Markov Model
HPC	High Performance Computing; Computing at a high performance level at any given time; often used synonym with Supercomputing
HPL	High Performance LINPACK, the top500 test.
ISC	International Supercomputing Conference; European equivalent to the US based SCxx conference. Held annually in Germany.
KB	Kilo (= 2^{10} ~ 10^3) Bytes (= 8 bits), also kbyte
kiB	Kibibyte (= 2^{10}) Bytes (Octets)
LINPACK	Software library for Linear Algebra
MB	Management Board (highest decision making body of the project)
MiB	MebiByte (= 2^{20}) Bytes (Octets)
MB	Mega (= 2^{20} ~ 10^6) Bytes (= 8 bits), also MByte
MB/s	Mega (= 10^6) Bytes (= 8 bits) per second, also MByte/s
MiB/s	Mebi (= 2^{20}) Bytes (Octets) per second
MFlop/s	Mega (= 10^6) Floating point operations (usually in 64-bit floating point, i.e. DP) per second, also MF/s
MIC	Multi Integrated Cores
MOOC	Massively Open Online Course
MoU	Memorandum of Understanding.
MPI	Message Passing Interface
NDA	Non-Disclosure Agreement. Typically signed between vendors and customers working together on products prior to their general availability or announcement.
PA	Preparatory Access (to PRACE resources)
PATC	PRACE Advanced Training Centres
PRACE	Partnership for Advanced Computing in Europe; Project Acronym
PRACE 2	The upcoming next phase of the PRACE Research Infrastructure following the initial five year period.
PRIDE	Project Information and Dissemination Event
RI	Research Infrastructure
SHAPE	SME HPC Adoption Programme in Europe
SME	Small and Medium Enterprises
TB	Technical Board (group of Work Package leaders)

TB	Tera (= 2^{40} ~ 10^{12}) Bytes (= 8 bits), also TByte
TiB	Tebi (= 2^{40}) Bytes (Octets)
TCO	Total Cost of Ownership. Includes recurring costs (e.g. personnel, power, cooling, maintenance) in addition to the purchase cost.
TDP	Thermal Design Power
TFlop/s	Tera (= 10^{12}) Floating-point operations (usually in 64-bit floating point, i.e. DP) per second, also TF/s
Tier-0	Denotes the apex of a conceptual pyramid of HPC systems. In this context the Supercomputing Research Infrastructure would host the Tier-0 systems
Tier-1	National or topical HPC centres in the conceptual pyramid
UNICORE	Uniform Interface to Computing Resources. Grid software for seamless access to distributed resources.
WP	Work Package

List of Project Partner Acronyms

BADW-LRZ	Leibniz-Rechenzentrum der Bayerischen Akademie der Wissenschaften, Germany (3rd Party to GCS)
BILKENT	Bilkent University, Turkey (3rd Party to UYBHM)
BSC	Barcelona Supercomputing Center - Centro Nacional de Supercomputacion, Spain
CaSToRC	Computation-based Science and Technology Research Center, Cyprus
CCSAS	Computing Centre of the Slovak Academy of Sciences, Slovakia
CEA	Commissariat à l'Energie Atomique et aux Energies Alternatives, France (3 rd Party to GENCI)
CESGA	Fundacion Publica Gallega Centro Tecnológico de Supercomputación de Galicia, Spain, (3rd Party to BSC)
CINECA	CINECA Consorzio Interuniversitario, Italy
CINES	Centre Informatique National de l'Enseignement Supérieur, France (3 rd Party to GENCI)
CNRS	Centre National de la Recherche Scientifique, France (3 rd Party to GENCI)
CSC	CSC Scientific Computing Ltd., Finland
CSIC	Spanish Council for Scientific Research (3rd Party to BSC)
CYFRONET	Academic Computing Centre CYFRONET AGH, Poland (3rd party to PNSC)
EPCC	EPCC at The University of Edinburgh, UK
ETHZurich (CSCS)	Eidgenössische Technische Hochschule Zürich – CSCS, Switzerland
FIS	FACULTY OF INFORMATION STUDIES, Slovenia (3rd Party to ULFME)
GCS	Gauss Centre for Supercomputing e.V.
GENCI	Grand Equipement National de Calcul Intensiv, France
GRNET	Greek Research and Technology Network, Greece
INRIA	Institut National de Recherche en Informatique et Automatique, France (3 rd Party to GENCI)
IST	Instituto Superior Técnico, Portugal (3rd Party to UC-LCA)
IUCC	INTER UNIVERSITY COMPUTATION CENTRE, Israel
JKU	Institut fuer Graphische und Parallele Datenverarbeitung der Johannes Kepler Universitaet Linz, Austria
JUELICH	Forschungszentrum Juelich GmbH, Germany
KTH	Royal Institute of Technology, Sweden (3rd Party to SNIC)
LiU	Linkoping University, Sweden (3rd Party to SNIC)
NCSA	NATIONAL CENTRE FOR SUPERCOMPUTING APPLICATIONS, Bulgaria
NIIF	National Information Infrastructure Development Institute, Hungary
NTNU	The Norwegian University of Science and Technology, Norway (3rd Party to SIGMA2)
NUI-Galway	National University of Ireland Galway, Ireland
PRACE	Partnership for Advanced Computing in Europe aisbl, Belgium
PSNC	Poznan Supercomputing and Networking Center, Poland
RISCSW	RISC Software GmbH

RZG	Max Planck Gesellschaft zur Förderung der Wissenschaften e.V., Germany (3 rd Party to GCS)
SIGMA2	UNINETT Sigma2 AS, Norway
SNIC	Swedish National Infrastructure for Computing (for the Swedish Research Council), Sweden
STFC	Science and Technology Facilities Council, UK (3rd Party to EPSRC)
SURFsara	Dutch national high-performance computing and e-Science support center, part of the SURF cooperative, Netherlands
UC-LCA	Universidade de Coimbra, Labotatório de Computação Avançada, Portugal
UCPH	Københavns Universitet, Denmark
UHEM	Istanbul Technical University, Ayazaga Campus, Turkey
UiO	University of Oslo, Norway (3rd Party to SIGMA2)
ULFME	UNIVERZA V LJUBLJANI, Slovenia
UmU	Umea University, Sweden (3rd Party to SNIC)
UnivEvora	Universidade de Évora, Portugal (3rd Party to UC-LCA)
UPC	Universitat Politècnica de Catalunya, Spain (3rd Party to BSC)
UPM/CeSViMa	Madrid Supercomputing and Visualization Center, Spain (3rd Party to BSC)
USTUTT-HLRS	Universitaet Stuttgart – HLRS, Germany (3rd Party to GCS)
VSB-TUO	VYSOKA SKOLA BANSKA - TECHNICKA UNIVERZITA OSTRAVA, Czech Republic
WCNS	Politechnika Wroclawska, Poland (3rd party to PNSC)

Executive Summary

The Work Package 7 'Application Enabling and Support' provides applications enabling support for HPC applications codes that are important for European academic and/or industrial researchers to ensure that these applications can effectively exploit current and future PRACE systems. Applications are selected for enabling via calls such as PRACE Preparatory Access (for Tier-0 or Tier-1), or SHAPE (SME HPC Adoption Programme in Europe). This applications enabling activity uses the most promising tools, algorithms and standards for optimisation and parallel scaling that have recently been developed through research and experience in PRACE and other projects. Through the applications-enabling work, the Work Package 7 develops specific expertise on most – if not all – of the architectures which make up the European HPC system. Technical results obtained within this Work Package have been disseminated though many technical whitepapers freely available on the PRACE RI web site. In addition to this enabling work on existing systems, the Work Package also progresses the technical work needed to ensure that key applications are able to use future PRACE Exascale systems, and investigate the tools, languages and libraries needed to exploit future PRACE Exascale systems.

One of the main objectives of the Work Package is to support European HPC research communities through the provision of Best Practice Guides, benchmarks, and example parallel codes.

The successful series of Best Practice Guides has been initiated in PRACE-1IP and has been continuously extended since then: PRACE-1IP provided four Best Practice Guides for PRACE Tier-0 systems (JUGENE, Curie, Cray XE and IBM Power) that cover programming techniques, compilers, tools and libraries (cf. [8]). PRACE-2IP added a generic guide about the x86 architecture and a Best Practice Guide for the SuperMUC system, together with a series of seven Best Practice Mini-Guides for other architectures which are important at Tier-1 to allow European researchers to make efficient use of these systems (cf. [9]). PRACE-3IP supplemented these with Best Practice Guides about Blue Gene/Q and IBM Power 775 and provided updates of the Curie and the Cray XE guide, which was renamed into Cray XE/XC. PRACE-4IP Task 7.3.B added new guides about Knights Landing and Haswell/Broadwell and provided updates of the Intel® Xeon Phi™ and the GPGPU Best Practice Guides.

Topics for these Best Practice Guides include: optimal porting of applications (e.g., choice of numerical libraries and compiler options); architecture specific optimisation and scaling techniques; optimal system environment (e.g., tuneable system parameters, job placement and optimised system libraries); debugging tools, performance analysis tools and programming environment.

This report describes the process which led to the Best Practice Guides and the structure of the guides. For the Best Practice Guides itself we refer to the online versions on the PRACE RI web site [1] (cf. [2][3][4][5]).

1 Introduction

Efficient use of PRACE systems requires detailed knowledge of architecture specific factors influencing performance, including compilers, tools and libraries. The main goal of this task is to investigate such issues, collect best practices on how to achieve good performance on the systems, and disseminate this knowledge to users.

The purpose of this report is to give a description of the process which led to the Best Practice Guides itself and present the structure of the guides.

In Section 2 we describe the selection of the systems, the subtasks, the technology used for creating the Best Practice Guides, and finally, the generic table of contents.

According to the DoW, this deliverable D7.6 "Best Practice Guides for New and Emerging Architectures" should report on the final versions of Best Practice Guides, covering process and structure. As in the past, because of the total size of the Best Practice Guides, we decided not to include them as separate chapters in this report but to refer to the online versions on the PRACE RI web site [1] instead (cf. [2][3][4][5]). Section 3 thus only gives a short summary of the contents of the guides.

The target audience are users and support staff who are developing and enabling applications.

2 Approach to Best Practice Guides

2.1 Selection of Systems

In the DoW we announced to maintain and extend the successful series of Best Practice Guides to new architectures/systems and mentioned Haswell/Broadwell x86 processors and many-core/accelerators (NVIDIA and Intel Knights Landing) as likely architectures.

When PRACE-4IP started in February 2015 we agreed to write the Haswell/Broadwell and Knights Landing guide as new Best Practice Guides from scratch. We also decided to update the Intel® Xeon Phi™ Guide written in PRACE-3IP, as many European Intel® Xeon Phi™ based systems were installed during PRACE-4IP. Finally, an update of the GPGPU mini-guide written within PRACE-2IP was envisaged to include information about the NVIDIA Pascal architecture. The release of the Intel Knights Landing and the NVIDIA Pascal chips was delayed a lot. Intel Knights Landing was introduced to the market during the ISC 2016 in Frankfurt, in June 2016, while NVIDIA Pascal cards were first shipped in August 2016. Due to the delay of the Pascal cards we decided to name the guide "Best Practice Guide - GPGPU" and also concentrated on general updates describing recent advances in technology and programming languages.

2.2 Editors

The whole Best Practice Guide activity 7.3.B was led by Volker Weinberg (LRZ); the respective editors/subactivity leaders were:

- Volker Weinberg, LRZ, for the Intel® Xeon Phi™ and the Haswell/Broadwell Best Practice Guide,
- Ole Widar Saastad, UiO, for the Knights Landing Best Practice Guide,
- Alan Gray, EPCC, for the GPGPU Best Practice Guide.

2.3 Technology

We built on the experience obtained during the corresponding PRACE-1IP, PRACE-2IP and PRACE-3IP tasks (cf. [8], [9], [10]). Although all PRACE deliverables are created using Microsoft Word, this did not seem to be the appropriate technology for creating the Best Practice Guides. It was decided already in PRACE-1IP that high quality HTML versions as well as high quality, fully featured PDF versions would be created and made available. To reach this goal, we use DocBook. DocBook (cf. [11], [12]) is being used by a lot of open source projects amongst others by the Linux Documentation Project. The key feature is having single (XML) source and multiple fully cross-referenced output formats: HTML, PDF and more. To keep track of the source code versions we used the PRACE SVN server.

To keep track of the status of the Best Practice Guides, Wiki pages for the whole activity and subpages for the individual guides have been set up on the PRACE Wiki, including information about status of the individual chapters of the guides, Docbook XML, SVN server access, timetables, generic table of contents, assignment of collaborators to topics and previous work within PRACE related to Best Practice Guides.

Fig. 1 shows the Best Practice Guides available on the PRACE RI web site [1] as of December 2016.

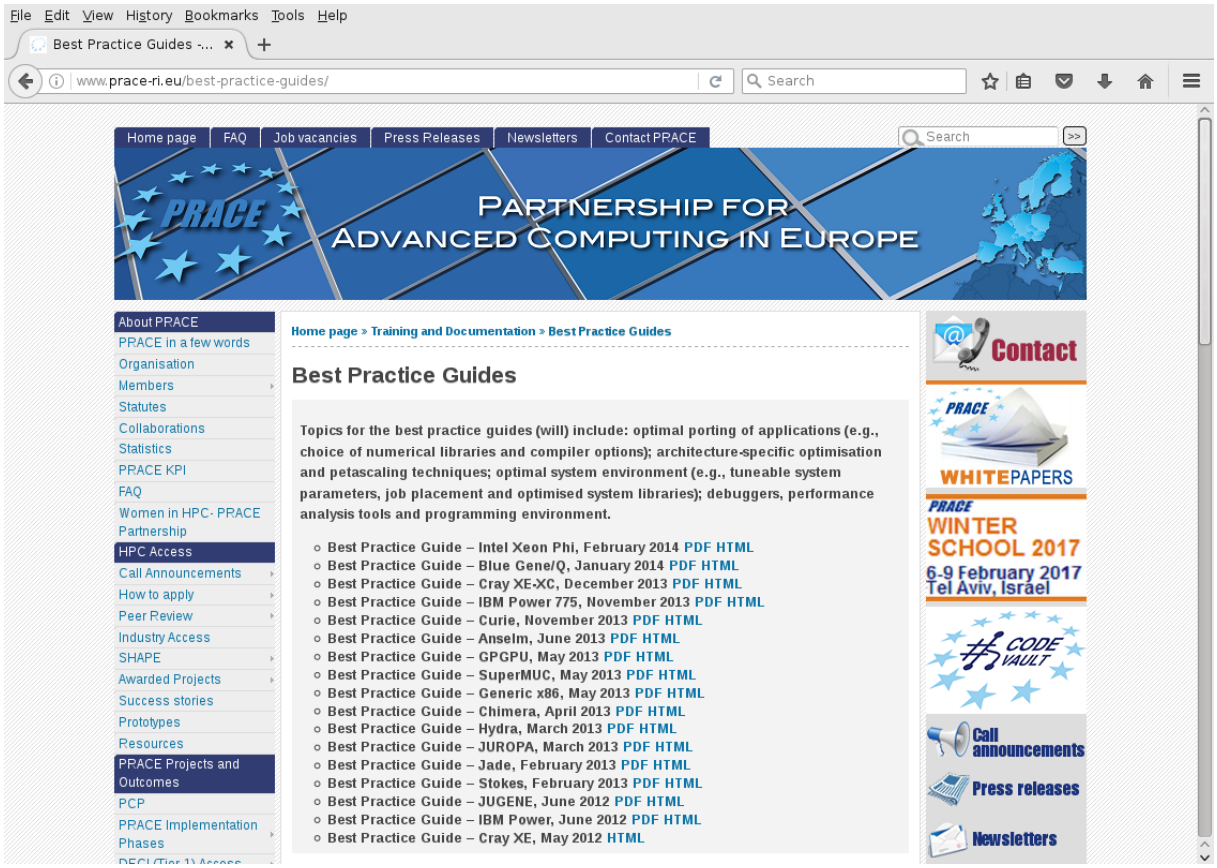

Figure 1: Best Practice Guides on the PRACE RI web site.

2.4 Generic Table of Contents

For PRACE-1IP, all Best Practice Guides were created based on the same generic table of contents. For PRACE-2IP we introduced a generic x86 guide and many system specific mini-guides. For this we decided to move as much information as possible to the generic x86 guide moving a limited number of system specific items from the generic table of contents to the mini-guides. The same approach was used for PRACE-3IP and PRACE-4IP.

The generic table of contents can be found below.

1. Introduction
2. System Architecture / Configuration
    1. Processor Architecture / MCM Architecture (including caches)
    2. Building Block Architecture (node cards, nodes, drawers, supernodes, racks)
    3. Memory Architecture (including NUMA effects)
    4. (Node) Interconnect (including topology, system specific)
    5. I/O Subsystem Architecture (being system specific and not architecture specific!)
    6. Available File Systems
        1. Home, Scratch, Long Time Storage
        2. Performance of File Systems
3. System Access
    1. How to Reach the System (ssh, portals, file transfer, ...)
4. Production Environment
    1. Module Environment
    2. Batch System
    3. Accounting
5. Programming Environment / Basic Porting
    1. Available Compilers
        1. Compiler Flags
    2. Available (Vendor Optimised) Numerical Libraries
    3. Available MPI Implementations
    4. OpenMP
        1. Compiler Flags
    5. Batch System / Job Command Language
6. Performance Analysis
    1. Available Performance Analysis Tools
    2. Hints for Interpreting Results.
7. Tuning
    1. Advanced / Aggressive Compiler Flags
    2. Single Core Optimisation
    3. Advanced MPI usage
        1. Tuning / Environment Variables
        2. Mapping Tasks on Node Topology
        3. Task Affinity

        4. Adapter Affinity
    4. Advanced OpenMP Usage
        1. Tuning / Environment Variables
        2. Thread Affinity
    5. Hybrid Programming
        1. Optimal Tasks / Threads Strategy
    6. Memory Optimisation
        1. Memory Affinity (MPI/OpenMP/Hybrid)
        2. Memory Allocation (malloc) Tuning
        3. Using Huge Pages
    7. I/O Optimisation (Tuning / Scaling of Application I/O)
    8. Advanced Job Command Language (includes defining task topology, affinity, etc.)
    9. Possible Kernel Parameter Tuning (probably less relevant to the 'average' user but possibly relevant for large production runs)
8. Debugging
        1. Available Debuggers
        2. Compiler flags

The actual tables of contents of the individual guides slightly deviate from this generic one, to best reflect system specifics.

2.5 Content

For all guides an inventory of the existing documentation was made that could be used as base material for some of the topics mentioned above. Many topics had to be complemented or even written from scratch. Apart from this, experiences learned during the various enabling or benchmark activities in other tasks were added.

As an internal quality assurance, T7.3B-internal cross-reviews were performed.

3 Best Practice Guides

The Best Practice Guides itself are to be found online on the PRACE RI web site [1] (cf. [2][3][4][5]). The following subsections give a short description of the contents of the guides.

3.1 Best Practice Guide – Intel® Xeon Phi™ (cf. [2])

Figure 2: The Intel® Xeon Phi™ Best Practice Guide.

This Best Practice Guide provides information about Intel's MIC architecture and programming models for the first generation Intel® Xeon Phi™ coprocessor named Knights Corner (KNC) in order to enable programmers to achieve good performance out of their applications.

The guide covers a wide range of topics from the description of the hardware of the Intel® Xeon Phi™ coprocessor through information about the basic programming models as well as information about porting programs up to tools and strategies how to analyse and improve the performance of applications.

The guide was created based on the PRACE-3IP Intel® Xeon Phi™ Best Practice Guide. New is the inclusion of information about European Intel® Xeon Phi™ based systems. The following systems are now described:

- Avitohol @ IICT-BAS, Bulgary,
- MareNostrum @ BSC, Spain,
- Salomon @ IT4Innovations, Czech Republic,
- SuperMIC @ LRZ, Germany.

Furthermore, the following new sections are included:

- OpenMP 4.x Offloading,
- OpenCL,
- Intel Cilk Plus / MYO,
- Information about some selected applications ported to Intel® Xeon Phi™,
- Benchmark results using e.g. the PRACE Accelerator Benchmark Suite.

Several existing sections have been updated to reflect recent changes.

3.2 Best Practice Guide – Knights Landing (cf. [3])

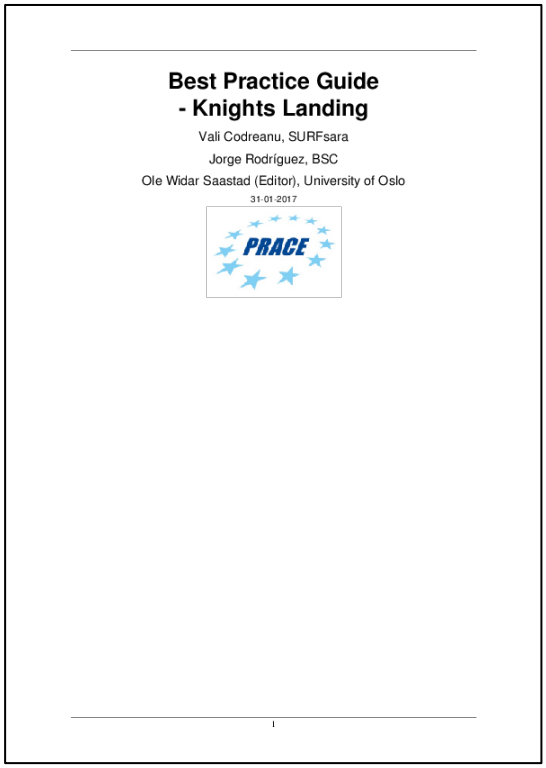

Figure 3: The Knights Landing Best Practice Guide.

This Best Practice Guide written from scratch provides information about Intel's MIC architecture and programming models for the second generation Intel® Xeon Phi™ processor named Knights Landing (KNL) in order to enable programmers to achieve good performance of their applications.

The guide covers a wide range of topics from the description of the hardware of the Intel® Xeon Phi™ processor through information about the basic programming models as well as information about porting programs up to tools and strategies how to analyse and improve the performance of applications.

The Intel Knights Landing (KNL) processor differs from usual Intel processors by its very high core count and the hardware threading architecture. It represents an approach where simple cores are employed in large number as opposed to more sophisticated cores in smaller number. The idea is that a higher fraction of the transistors could be used for arithmetic operations, since the number of flops per transistor have declined over the decades. This new processor design represents a different programming paradigm and forces the programmer to design applications in a different way. The binary compatibility with current processors facilitates such a transition. Porting is in many cases trivial, but achieving high performance and optimised hardware utilisation requires in many cases substantial work.

The guide gives an overview of programming models, MPI, OpenMP threading and hybrid MPI & OpenMP. For processors with such a high core count hybrid models are suggested. Following the trend of the first generation of Intel® Xeon Phi™ even more focus has been placed on vectorisation with the extra-long vectors (512 bits) handled by this processor.

There are extra chapters on benchmarking and applications. Benchmarking is done to assess the performance in different configurations. The complex processor has different ways of setting up the cores and the memory. With a special high bandwidth memory the KNL processor represents a new level of complexity. Application porting, scaling, tuning and

performance assessment sections are also included to show how real applications can be ported and tuned to this novel architecture.

In the tuning context several of the tuning tools provided by Intel are demonstrated and examples are given on how to use command line tools in batch mode. The guide is not an Intel tools guide, but rather a guide on how to collect data using command line tools for this special processor.

As of writing the Best Practice Guide, there were no large systems based on the Intel Knights Landing processor available. The only systems used for testing thus were standalone local nodes.

3.3 Best Practice Guide – Haswell/Broadwell (cf. [4])

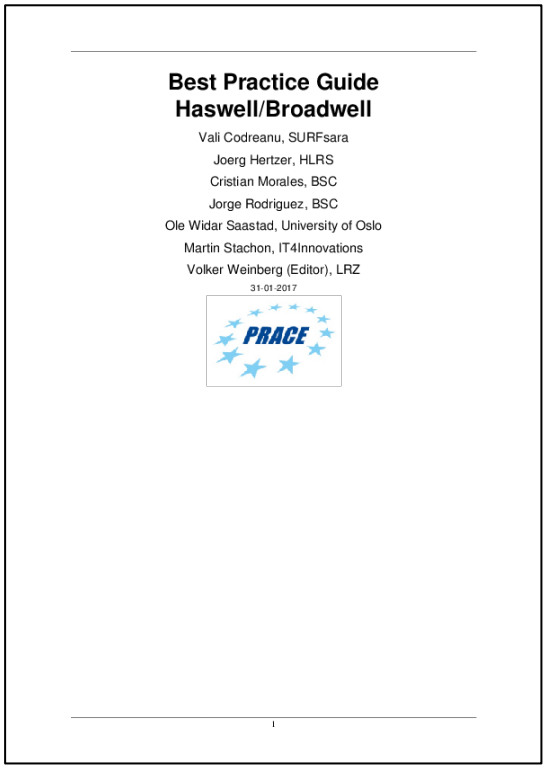

Figure 4: The Haswell/Broadwell Best Practice Guide.

This Best Practice Guide written from scratch provides information about Intel's Haswell/Broadwell architecture in order to enable programmers to achieve good performance of their applications. The guide covers a wide range of topics from the description and comparison of the hardware of the Haswell/Broadwell processor, through information about the compiler usage as well as information about porting programs up to tools and strategies how to analyse and improve the performance of applications.

With the introduction of extra vector instructions with these processors (fused multiply add etc.) stronger focus is placed on vectorisation. In the tuning context several of the tuning tools provided by Intel are demonstrated and examples are given on how to use command line tools to collect data in batch mode.

Furthermore, the guide provides information about the following European Intel Haswell/Broadwell based European systems:

- Hazel Hen @ HLRS, Germany,
- Minotauro @ BSC, Spain,

- Salomon @ IT4Innovations, Czech Republic,
- SuperMUC Phase 2 @ LRZ, Germany.

3.4 Best Practice Guide – GPGPU (cf. [5])

Figure 5: The GPGPU Best Practice Guide.

This Best Practice Guide describes general purpose computation on Graphics Processing Units (GPUs). GPUs were originally developed for computer gaming and other graphical tasks, but for many years have been exploited for general purpose computing across a number of areas. They offer advantages over traditional CPUs because they have greater computational capability and use high-bandwidth memory systems (with memory bandwidth being the main bottleneck for many scientific applications).

The guide includes information on how to get started with programming GPUs, which cannot be used in isolation but only as "accelerators" in conjunction with CPUs, and how to get good performance. Focus is given to NVIDIA GPUs, which are most widespread today.

The GPGPU Best Practice Guide is based on the PRACE-2IP GPGPU Best Practice Mini-Guide. The following new topics were added:

- GPU architecture especially highlighting new features of NVIDIA Pascal,
- OpenCL Programming,
- OpenMP 4.x Offloading.

Several existing sections have been updated to reflect recent changes.