Dynamical Exascale Entry Platform

Grant Agreement Number: 287530

D7.2
Concepts for improving energy and cooling efficiency

Approved

Version: 2.0
Author(s): N. Meyer (UniReg), T. Wettig (UniReg), S. Solbrig (UniReg), A. Auweter (BADW-LRZ), M. Ott (BADW-LRZ), D. Tafani (BADW-LRZ)
Date: 25.02.2013
**Project and Deliverable Information Sheet**

<table>
<thead>
<tr>
<th>DEEP Project</th>
<th>Project Ref. №: 287530</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Project Title: Dynamical Exascale Entry Platform</td>
</tr>
<tr>
<td></td>
<td>Project Web Site: <a href="http://www.deep-project.eu">http://www.deep-project.eu</a></td>
</tr>
<tr>
<td></td>
<td>Deliverable ID: D7.2</td>
</tr>
<tr>
<td></td>
<td>Deliverable Nature: Report</td>
</tr>
<tr>
<td></td>
<td>Deliverable Level: PU*</td>
</tr>
<tr>
<td></td>
<td>Contractual Date of Delivery: 30 / November / 2012</td>
</tr>
<tr>
<td></td>
<td>Actual Date of Delivery: 30 / November / 2012</td>
</tr>
</tbody>
</table>

**EC Project Officer:** Luis Carlos Busquets Pérez

* - The dissemination level are indicated as follows: **PU** – Public, **PP** – Restricted to other participants (including the Commission Services), **RE** – Restricted to a group specified by the consortium (including the Commission Services). **CO** – Confidential, only for members of the consortium (including the Commission Services).

**Document Control Sheet**

<table>
<thead>
<tr>
<th>Document</th>
<th><strong>Title:</strong> Concepts for improving energy and cooling efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>ID:</strong> D7.2</td>
</tr>
<tr>
<td></td>
<td><strong>Version:</strong> 2.0</td>
</tr>
<tr>
<td></td>
<td><strong>Available at:</strong> <a href="http://www.deep-project.eu">http://www.deep-project.eu</a></td>
</tr>
<tr>
<td></td>
<td><strong>Software Tool:</strong> Microsoft Word</td>
</tr>
<tr>
<td></td>
<td><strong>File(s):</strong> DEEP_D7.2_Concepts_for_improving_energy_and_cooling_efficiency_v2.0-ECapproved</td>
</tr>
</tbody>
</table>

**Authorship**

| Written by: | N. Meyer (UniReg), T. Wettig(UniReg), S. Solbrig(UniReg), A. Auweter (BADW-LRZ), M. Ott(BADW-LRZ), D. Tafani (BADW-LRZ) |

**Contributors:**

**Reviewed by:** F. Affinito (CINECA), W. Gürich (JUELICH)

**Approved by:** BoP/PMT
## Document Status Sheet

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Status</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>09/November/2012</td>
<td>Draft</td>
<td>Submitted to internal review</td>
</tr>
<tr>
<td>0.4</td>
<td>12/November/2012</td>
<td>Draft</td>
<td>Updated references and acronyms</td>
</tr>
<tr>
<td>0.5</td>
<td>19/November/2012</td>
<td>Draft</td>
<td>Integrated recommendations from internal review</td>
</tr>
<tr>
<td>0.6</td>
<td>28/November/2012</td>
<td>Final</td>
<td>Integrated recommendations from PMT review</td>
</tr>
<tr>
<td>1.0</td>
<td>30/November/2012</td>
<td>Final</td>
<td>EC submission</td>
</tr>
<tr>
<td>2.0</td>
<td>25/February/2013</td>
<td>Approved</td>
<td>Approved by EC</td>
</tr>
</tbody>
</table>
Concepts for improving energy and cooling efficiency

Document Keywords

<table>
<thead>
<tr>
<th>Keywords</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEEP, HPC, Exascale, Energy Efficiency, Liquid Cooling, System Monitoring</td>
</tr>
</tbody>
</table>

Copyright notices

© 2011 -2012 DEEP Consortium Partners. All rights reserved. This document is a project document of the DEEP project. All contents are reserved by default and may not be disclosed to third parties without the written consent of the DEEP partners, except as mandated by the European Commission contract 287530 for reviewing and dissemination purposes. All trademarks and other rights on third party products mentioned in this document are acknowledged as own by the respective holders.
Table of Contents

Project and Deliverable Information Sheet ........................................................... i
Document Control Sheet ...................................................................................... i
Document Status Sheet ....................................................................................... ii
Document Keywords ........................................................................................... iii
Table of Contents ................................................................................................. iv
List of Figures ....................................................................................................... v
List of Tables ......................................................................................................... v
Executive Summary .............................................................................................. 6

1 Outline .............................................................................................................. 7

2 Liquid Cooled Node Designs ............................................................................ 8
   2.1 Motivation .................................................................................................... 8
   2.2 Types of Cooling Solutions ....................................................................... 8
       2.2.1 Indirect Coldplate ............................................................................. 8
       2.2.2 Direct Coldplate .............................................................................. 10
       2.2.3 Cooling Pipeline .............................................................................. 13
   2.3 Material Considerations ............................................................................ 15
   2.4 Ranking Matrix .......................................................................................... 18
   2.5 Conclusions ................................................................................................. 20

3 System Monitoring for Energy Efficiency ...................................................... 21
   3.1 Motivation .................................................................................................. 21
   3.2 Available Sensors in the DEEP System .................................................... 22
   3.3 Sensor Acquisition .................................................................................... 23
   3.4 Scalability of the System Management Framework .................................. 24

4 System Management for Energy Efficiency .................................................. 27
   4.1 Motivation .................................................................................................. 27
   4.2 Voltage and Frequency Scaling ................................................................ 27
   4.3 Concurrency Throttling ............................................................................ 31
   4.4 Thread Pinning ........................................................................................... 31

References and Applicable Documents ............................................................. 32
List of Acronyms and Abbreviations ................................................................. 34
List of Figures

Figure 1: QPACE Node Design ................................................................. 10
Figure 2: Aurora Node Design ................................................................. 12
Figure 3: iDataCool and SuperMUC Node Designs ..................................... 13
Figure 4: Liquid Cooled Heat Bridge .......................................................... 16
Figure 5: General Concept of a Sensor Management Framework .................. 24
Figure 6: Scalable Sensor Management Framework through Hierarchical Ordering .................................................. 25
Figure 7: Scalable Sensor Management Framework using a Distributed Data Store ........................................... 26
Figure 8: Time-to-Solution of HYDRO Benchmark on CoolMUC .................. 29
Figure 9: Energy-to-Solution of HYDRO Benchmark on CoolMUC .................. 29
Figure 10: Time-to-Solution of HYDRO Benchmark on SuperMUC ................. 30
Figure 11: Energy-to-Solution of HYDRO Benchmark on SuperMUC ............... 30

List of Tables

Table 1: Comparison of Liquid Cooled Node Designs ................................. 20
Table 2: List of DEEP Building Infrastructure Sensors .................................. 22
Table 3: List of DEEP System Sensors ....................................................... 23
Executive Summary

This deliverable describes concepts for improved energy and cooling efficiency of supercomputers such as DEEP. Our previous deliverable D7.1 targeted HPC centre infrastructures, meaning the data centre building, power supply, and cooling system. In contrast to this, the following document focuses on energy efficiency for the machine itself. Similarly to the site infrastructures, the cooling solution for compute nodes contributes to the overall energy efficiency. For this, we have analysed three solutions that are available in the market with respect to cost, weight, serviceability, engineering effort, robustness and performance. We conclude that the DEEP direct coldplate approach performs best under the named criteria.

To get a good overview of the operating condition of the DEEP System, thorough monitoring is indispensable. To ensure this, a large number of sensors will be available in DEEP. Although sensor information will be gathered through various protocols, all sensor information will be available from within a single system monitoring framework. This framework will also correlate power and resource management information to calculate the Energy-to-Solution of each application. The Energy-to-Solution of an application is defined as the energy consumed by all the hardware components necessary to run the application during the application’s execution time.

The availability of Energy-to-Solution measurements will allow us to investigate possible approaches to lower the Energy-to-Solution. For this, we have identified three promising approaches: voltage and frequency scaling, concurrency throttling and thread pinning. Experiments with voltage and frequency scaling on our test platforms have shown that significant savings in energy can be achieved. Yet, the effectiveness of each approach varies with different systems. Intensive testing will therefore be required on the final DEEP hardware to devise the most efficient solution.
1 Outline

Energy efficiency remains a key challenge on the way to Exascale computing. In deliverable D7.1 we have outlined the necessity of both local and global optimisations on the data centre infrastructure stack. In this document we focus on optimisations on the machine level and node level of the DEEP architecture.

Liquid cooling is by far the most efficient way of cooling high-performance computers. It enables free cooling year-round and facilitates energy reuse. In directly liquid-cooled designs the coolant is brought as close as possible to the components on the node. Heat is directly dissipated to the coolant without the need for any airflow. We discuss and compare implementations of direct liquid-cooling on the node level in the first part of this document and give recommendation for cooling of the DEEP Booster.

Following the philosophy of “you can only optimise what you can measure”, in the second part we stress the importance of system monitoring of DEEP and its connection to energy-efficient system operation. We summarise the Jülich data centre infrastructure sensors and all sensors available within the DEEP Cluster and the DEEP Booster. We discuss scalable system monitoring frameworks capable of efficiently collecting, storing and handling all sensor information.

Finally, the third part of this document addresses opportunities for reduction of energy consumption of the DEEP architecture. Since no DEEP prototype is available yet, we have performed benchmarks on existing supercomputer architectures, measuring Energy-to-Solution and runtime. The results widely differ between the architectures. It is not yet predictable which of the proposed techniques will have the most impact on the energy efficiency of the DEEP System. Still, with the monitoring capabilities of the DEEP System, we will be able to start the evaluation of the proposed techniques as soon as the first DEEP hardware becomes available in order to conclude with the best energy saving approaches for the final DEEP System in the deliverables D7.4 and D7.5.
2 Liquid Cooled Node Designs

2.1 Motivation

In Deliverable D7.1 we have evaluated different types of cooling solutions typically found in today’s data centres. We have identified three main classes: air-cooled, indirectly liquid-cooled and directly liquid-cooled systems. The primary medium for heat removal in both air-cooled and indirectly liquid-cooled systems is air blown over the system components. Indirectly liquid-cooled systems allow for removal of waste heat to a liquid cooling circuit using air-to-liquid heat exchangers. For directly liquid-cooled systems no airflow is required at all. Removal of heat is based on the cooling liquid being in thermal contact with components generating heat. In directly liquid-cooled systems the coolant is brought as close as possible to the components. In this section we exclusively focus on solutions for direct liquid-cooling of the compute nodes. We have identified three main types of cooling solutions for direct liquid cooling which we discuss in detail. We give examples for the implementation of the solutions found in recent HPC systems. An important aspect of direct liquid cooling is the material choice for the cooling design. Copper has optimal material properties with respect to cooling performance. However, the choice of copper comes with a series of drawbacks. Another material commonly used is aluminium. We compare both materials and illustrate both the advantages and disadvantages of copper or aluminium. Finally we discuss general aspects of the three types of direct liquid cooling. We contrast different properties of the solutions in a simple ranking system and conclude with a summary and suggestions for high power-density systems.

2.2 Types of Cooling Solutions

2.2.1 Indirect Coldplate

One method to remove the heat from the compute node is a two-component solution consisting of a thermal box in combination with a coldplate. Each node is attached to a thermal box which serves as a heat sink. The surface of the box is shaped with a Manhattan skyline profile that mimics the height and the power signature of the devices on the node. The box is designed to maximise the heat flow from the devices to one of its six surfaces. This ‘hot’ surface is attached to a coldplate, which is connected to the cooling circuit (see left picture of Figure 1). Channels inside the coldplate guide the flow of the coolant. Thus the heat is transferred from the node to the surface of the thermal box to the coldplate and finally to the coolant\(^1\). We refer to this kind of cooling as “indirect coldplate”, since the thermal box is never exposed to any liquid.

There is a variety of options for mounting nodes to the coldplate, depending on the dimension and power signature of the node, and also for the thermal design of the thermal box and coldplate. Thermal conductivity is proportional to the interface area, thus the heat flow from the box to the coldplate is largest if the interface area is maximised. This is the case if the largest surface of the box is mounted on the coldplate, allowing for double-sided population with at least two nodes. Due to the large thermal interface this approach in principle is feasible for very high power-density systems with several CPUs per node. Another, more economical, option is to mount several thermal boxes side-by-side perpendicular to the

---

\(^1\) For this cooling solution to be efficient the interface area between thermal box and coldplate has to be sufficiently smooth. Thermal conductivity between the thermal box and the coldplate is typically increased by thermal grease or silicon oil.
coldplate. In this approach the thermal interface is much smaller. However, the packaging density is increased significantly.

There are well-known examples for high performance systems based on this cooling solution. One is the IBM Blue Gene/Q, the leader of both the Top 500 and Green 500 lists in June 2012 [1], [2]. The largest installation of Blue Gene/Q has a peak performance of more than 20 PetaFlops and an energy signature of 2100 MegaFlops per Watt in the Linpack benchmark. In the nomenclature of IBM a single Blue Gene node card comprises 32 compute cards [3]. The Blue Gene processor is a system-on-a-chip ASIC design, i.e. the compute cores, memory controllers, network interface and I/O interface are embedded in a single device. Each compute card hosts one ASIC and 16 GB off-chip memory, which is soldered on the board. Each compute card dissipates its heat by a heat sink that is mounted to the card by bolts and clamped to the coldplate which removes the heat. Each coldplate allows for cooling of up to 32 compute cards. According to the official documentation water temperatures up to 25°C above the dew point are possible [3].

The Blue Gene/Q architecture is truly remarkable not only with respect to the cooling design. However, we like to draw attention to another implementation of the indirect coldplate cooling solution. In the following we discuss the cooling solution for the QPACE supercomputer. QPACE ranked first place in the Green500 list both in November 2009 and June 2010, achieving 773 MegaFlops per Watt in the Linpack benchmark [4]. The special-purpose architecture focuses on the requirements of applications in lattice QCD. QPACE is designed by a collaboration of academic and industrial partners, amongst them the University of Regensburg and IBM Research and Development, Böblingen, Germany. Two installations of QPACE have been in operation since 2009, one at the Jülich Supercomputer Centre and one at the Bergische Universität Wuppertal, Germany. Each installation consists of four racks. The aggregate compute performance is 200 TeraFlops in double precision. We refer to [5] and [6] for further details on QPACE.

The QPACE node is based on the IBM PowerXCell 8i, which is an enhanced version of the IBM Cell Broadband Engine designed for the PlayStation 3 gaming platform. Besides the PowerXCell 8i each node hosts one Xilinx FPGA, seven Ethernet PHYs, a board management controller, CPLD and 4 GB of surface-mounted memory. The FPGA implements a custom-designed I/O fabric. The nodes communicate by a three-dimensional nearest-neighbour torus network based on 10 Gbit Ethernet physical layer. A switched Gigabit Ethernet network is used for I/O operations. High speed torus network signals are routed by backplanes, which are connected to each other by cables. Thirty-two nodes are attached to one backplane. Four backplanes are mounted to each side of the rack, resulting in a packaging density of 256 nodes per rack.

The outer dimensions of the node are about 32x15 cm². All devices are mounted single-sided on the Printed Circuit Board (PCB). On the edge of the PCB there are power and signal connectors for attachment to the backplane. The peak power consumption per node has been determined by a synthetic benchmark to be about 130 W. With a contribution of more than 50% to the total heat production the PowerXCell 8i is the heat-critical component on the PCB. Other components that need cooling are the FPGA, the memory, the physical network interface chips (PHY) and the voltage converters.

---

2 Each QPACE node has undergone individual tuning of both processor core and main memory voltage.
Figure 1: QPACE Node Design
The left panel shows the concept of the QPACE cooling solution. The QPACE nodes, each embedded in a thermal box, are mounted on the coldplate. Heat is conducted from the thermal box to the coldplate at the interface surface (indicated by the red stripe). The right panel shows the coldplate fully populated with 32 nodes and two administration cards. Manifolds are mounted on the left side of the rack and connect each coldplate to the cooling circuit. Power supplies are attached to the right side of the rack.

An efficient two-component cooling mechanism is used to remove the heat from the nodes. Each node is packed in a thermal box made of cast aluminium alloy, with 32 such boxes attached to a single water-cooled coldplate. This setup is depicted in the right picture of Figure 1. The thermal box acts as a large heat sink, taking into account the individual height of the devices on the PCB and their individual contribution to heat generation. To avoid distribution of heat from the PowerXCell 8i to other devices on the board a separate aluminium block inside the thermal box is used to cool the processor. The interface between the thermal box and the coldplate is only about 40 cm². Thermal contact is increased by a thin film of synthetic oil.

The footprint of the coldplate is about 34x50 cm². Four water-conducting channels inside the coldplate remove the heat from the surface of the thermal boxes, allowing for heat dissipation of up to 5 kW. Up to 16 nodes each can be mounted to top and bottom side of the coldplate. The water channels have been designed to achieve a homogeneous temperature distribution across the nodes. Each coldplate is connected to the cooling circuit by a single water inlet and outlet. Ball valves attached to the manifold allow for a decoupling of the coldplate from the water circuit. Besides cooling, the coldplate is also used for clamping of the nodes by a spring system.

The temperature difference between the PowerXCell 8i and the coolant is lower than 40°C. The PowerXCell 8i is specified to operate at temperatures up to 95°C. In principle QPACE can thus be operated with warm water at inlet temperatures above 40°C, allowing for free cooling year-round or potential reuse of the waste heat, e.g. in underfloor heating systems.

2.2.2 Direct Coldplate

Instead of a two-component solution with separate thermal box and coldplate, the node can also be directly mounted on, and cooled by, the coldplate. This requires each coldplate to be designed with a Manhattan skyline profile, taking into account both the height and the heat

---

3 To avoid electromagnetic radiation and also accumulation of dust inside the thermal box and on the backside of the node PCB the thermal box is closed by a separate metal lid. There is no requirement for the lid to contribute to the heat flow since all components on the node are mounted single-sided.
contribution of the individual devices on the node. Channels inside the coldplate guide the flow of the coolant from the inlet to the outlet. We refer to this kind of cooling as “direct coldplate”, since the heat is removed from hot spots on the node with almost direct contact to the coolant.

The number of nodes attached to a single coldplate depends on the form factor of the node, its power signature and also on the thermal design and dimension of the coldplate. In principle multiple PCBs can be mounted to a single coldplate if the thermal design is sufficiently capable. The packaging density is nearly doubled if the skyline profile is embossed on both sides of the coldplate.

Examples for high-performance systems based on this type of liquid cooling are the Aurora supercomputers designed by Eurotech [7], [8]. The Aurora series is designed for a wide range of applications. The installation sizes are scalable up to the PetaFlops regime. The recently developed Aurora HPC 10-10 architecture comprises floating-point performance up to 100 TeraFlops per rack. We refer to [8], [9], [10], [11] for further details on Aurora systems and their fields of application. In September 2012 the installation of the DEEP Aurora HPC 10-10 cluster comprising 128 nodes based on Intel Xeon E5 Sandy Bridge processors was finished. The Cluster complements the DEEP Booster, which is still under development.

Nodes of the Aurora series rely on Intel Xeon 5600 Westmere and Intel Xeon E5 Sandy Bridge server processors, with two processor sockets per node. Up to 32 GB DDR3 memory is soldered on the board. Mass storage, e.g., a solid state drive, can be directly mounted on the node. The communication networks are a special feature of the system design. Access to both a switched InfiniBand network and a three-dimensional torus network optimised for nearest-neighbour communication is provided. All compute nodes, control cards, InfiniBand switches and power conversion units are liquid-cooled with support for hot-water cooling at temperatures of the liquid of up to 50°C.

The cooling solution of the Aurora node relies on the direct coldplate approach. The cooling has to dissipate approximately 300 Watts of thermal energy to efficiently cool two Xeon E5s. The coldplate is made out of aluminium and is shown in Figure 2. The heat-critical components on the node are in thermal contact with the Manhattan skyline profile. Mechanical and thermal coupling of the node to the coldplate is realised by a system of springs and bolts. Thermal conductance from the devices to the coldplate is increased by thermal interface material. The coolant flows over the skyline profile guided by channels inside the coldplate. Two quick couplings are used for connection to the water distribution system. The coolant is distributed to each node by liquid distribution bars mounted to the backplane. Each backplane allows for mounting of 16 nodes.
Figure 2: Aurora Node Design
The top picture shows the Aurora coldplate with its Manhattan skyline profile. Thermal interface material optimises the heat flow from hot spots to the coldplate. Quick couplings are positioned on the edge of the coldplate. The bottom picture shows a single node mounted on the coldplate. Pictures are taken from [11].
2.2.3 **Cooling Pipeline**

Yet another approach to heat removal is provided by a “cooling pipeline” that guides the flow of the coolant over the node. While the coolant is flowing over the devices it picks up the heat. There is no need for a massive coldplate. Heat is removed from low power devices such as bridges, memories and networking adapters using heat bridges made of solid blocks of heat-conducting material which is in thermal contact with the pipeline. High power-density devices, such as CPUs, are cooled by heat sinks through which the coolant flows directly. The internal structure of these heat sinks is designed to minimise the temperature difference between silicon and coolant. Thus these heat sinks allow for efficient heat removal directly at the hot spot. Thermal conductance between the devices and the heat sink has to be improved by thermal interface material. Silicon oil can be used to improve the thermal contact between the pipeline and heat bridges.

![Figure 3: iDataCool and SuperMUC Node Designs](image)

The picture on the left shows the iDataCool compute node. The cooling solution consists of a copper pipeline, copper heat sinks and aluminium heat bridges (red), arranged in a serial cooling chain. Water inlet and outlet are attached to the back of the node. The picture on the right shows the SuperMUC compute node. Aluminium heat bridges and heat pipes are attached to a copper pipeline in a parallel cooling chain.

In this approach to liquid cooling each node has to be cooled by a dedicated pipeline, requiring the coolant to be supplied on a per-node basis. In contrast to the coldplate approaches the PCB is only partially insulated from the environment. Additional thermal and, of course, also electromagnetic insulation of the node has to be ensured. A containment for the node is required that is compatible with the cooling design and also provides thermal insulation from the environment.

There are a variety of examples for this cooling solution. We have already introduced two recent HPC clusters based on the cooling pipeline approach in Deliverable D7.1. One is the CooLMUC cluster at the BADW-LRZ in München. The other one is the iDataCool cluster at the University of Regensburg. Both systems have been designed for energy reuse. In the following we discuss the implementation of the cooling solution in the iDataCool system. Afterward we briefly compare the iDataCool solution with one of Germany’s recent Petascale computer clusters, the SuperMUC [12], which is installed at the BADW-LRZ.

iDataCool is an experimentation cluster based on the IBM System x iDataPlex platform [13]. Each iDataPlex node is equipped with either two Intel Xeon E5630 or E5645 Westmere server processors each with four or six cores, respectively. Per node 24 GB shared memory are arranged in six 4 GB DDR3 dual in-line memory modules. QDR InfiniBand is used for interconnection of the nodes. The power signature depends on the processor type and does not
Concepts for improving energy and cooling efficiency

exceed a maximum of 240 W per node. The iDataCool installation consists of three iDataPlex racks with a total of 216 nodes.

The original iDataPlex system is entirely air-cooled. For iDataCool only the nodes have been modified for water cooling, whereas other system components (PSUs and network switches) remain air-cooled. On each node the original heat sinks have been removed and replaced by aluminium heat bridges and copper heat sinks interfacing with a pipeline system made of copper. The water-cooling design is a combined effort of the University of Regensburg and IBM Research and Development, Böblingen, Germany. All modifications have been performed at the University of Regensburg with newly developed parts manufactured at the university’s machine shop. The waste heat of iDataCool is used to drive an adsorption chiller, which in turn cools other compute equipment in the data centre. The efficiency of the chiller scales with the temperature of the cooling water, with a minimum temperature of about 60°C to operate the chiller. For this reason the cooling solution is optimised for operation at temperatures above this threshold.

The iDataCool node is shown in Figure 3. For illustration purposes, all aluminium parts of the cooling solution are highlighted in red. The heat-critical components on the node are the CPUs. Intel’s cooling specifications for the server processors state 80 W of thermal design power for both types of CPUs. Custom-designed heat sinks made from copper are used to satisfy this demand for cooling performance. The heat sinks are attached to the cooling chain in series. For the memory modules copper heat spreaders have been designed. The heat spreaders are in thermal contact with aluminium bars which embrace the pipeline. Other low-power devices such as the InfiniBand daughter card and voltage regulators are cooled by heat bridges made of aluminium. A separate aluminium block embedded into one of the heat bridges is used for cooling of the southbridge. Only two of the four sides of this block are thermally connected to the heat bridge. This allows for directing the flow of heat towards the copper pipe.

In order to ensure both, the necessary mechanical stability as well as good thermal conductance, all components of the cooling design require proper mounting. To achieve this, the heat bridges are aligned to the node by bolts, reusing the original mount points on the PCB. The copper pipe resides inside u-shaped notches incorporated into the bridges. The pipeline is clamped to the bridges using small plates mounted by bolts. A thin film of silicon oil increases the heat flow from the bridges to the copper pipeline. Plates, springs and bolts are used to align the heat sinks to the processors. Clamps press the aluminium heat bridges to the memory modules, allowing for memory replacement in the field without the need to remove other cooling components. To optimise the heat flow, different kinds of thermal interface material are used. Thermal insulation foam attached to the original node chassis reduces heat dissipation into the data centre.

The cooling solution for iDataCool has been tested with water inlet temperatures even above 60°C. Temperature differences between the water inlet and the CPU cores below 25°C can be achieved for the six-core processors and even lower for the four-core processors. The inlet temperature is limited by CPU throttling which sets in at a core temperature of 100°C.

We close this overview with a brief comparison of the cooling pipeline implementation in the SuperMUC cluster and iDataCool. We like to illustrate the design flexibility provided by the pipeline solution. The SuperMUC node is shown in Figure 3. The Sandy Bridge processors, also cooled by heat sinks through which the water flows directly, are connected in series to the cooling chain. Eight pipelines are used in parallel for cooling of the in-line memory modules. Other components such as the southbridge and InfiniBand daughter card are connected to the pipeline by heat bridges and heat pipes.
2.3 Material Considerations

As a prerequisite for the comparison of the three different approaches to cooling it is useful to discuss some general aspects about the materials in use. The liquid cooling circuit for a computer installation consists of several parts including pumps, heat exchangers, and pipelines for transfer of the coolant, distribution of the coolant on rack level and finally the node cooling. We have discussed the importance of the material mix and the problem of water treatment for closed liquid circuits in Deliverable D7.1. In this section we focus on the materials for the node cooling exclusively. There are basically only two elements (and their alloys) whose material properties are suitable for cooling: aluminium and copper. We briefly introduce the basic properties of these two elements and give examples for typical applications.

Aluminium is a metal widely found in the crust of earth. Due to its reactivity with other elements it has to be broken from mineral structures, mostly bauxite, in an energy and resource-consuming process. Pure aluminium is a lightweight material with a density of only 2.7 g/cm³ at room temperature. Amongst the metals its thermal conductivity of about 230 W/mK at 0°C is rather high. Pure aluminium is soft and ductile. Its mechanical strength is rather low. Therefore it is excellent to process, but its mechanical properties render it inappropriate for many applications. When combined with other elements, the properties of this material change significantly, see [14], [15], [16]. For example, aluminium alloys with admixture of zinc have excellent mechanical properties, making these materials especially attractive for applications in aerospace. Due to their thermal properties admixtures with manganese are commonly used in the automotive industry and are also found in kitchenware. Incorporation of magnesium leads to materials with highly improved mechanical stability, good soldering properties and low susceptibility for corrosion. Quality heat sinks for passive and active air-cooling of computer components are typically made of aluminium with admixture of magnesium and silicon [17].

Copper is a metal much rarer than its companion aluminium. It essentially has to be extracted from copper ore by a series of chemical reactions. It has a rather high density of about 8.9 g/cm³ at room temperature and is assigned to the group of heavy metals. The thermal conductivity is about 400 W/mK at 0°C, much higher than most other metals. Pure copper is a soft metal that can easily be processed. The electric properties are excellent and therefore copper is extensively used for cables, wires and electrical contacts. Copper is also used in systems that require good heat conductance, e.g., in heat exchangers. Applications in the cooling of electronic devices are, e.g., heat pipes made of copper. Heat pipes allow for efficient heat transfer using the liquid-to-vapour phase transition of the working fluid, see [18] for an explanation of the underlying mechanism. In recent years liquid-cooled heat sinks based on copper have become well established in the home PC market. There is a large variety of copper alloys with different properties, see [19]. Commonly used are brass (copper and zinc), bronze (copper and other metals) and cupronickels (copper with mainly nickel and a few strengthening elements). Because of their excellent resistance against corrosion copper and its alloys are widely used for pipes, valves and fittings in aqueous environments [20].
Let us discuss the pure materials with respect to their cooling performance. Here the statement is simple: copper is advantageous because its thermal conductivity is almost twice as high as that of aluminium. Although this statement is quite intuitive, let us illustrate the thermal conductivity on an ideal model that does not require complicated computer simulation. For simplicity we assume a single chip to be cooled by some liquid. A cube-shaped heat bridge, with some length and the cross section covering the chip surface, thermally connects the chip to the liquid circuit. We neglect any dissipation of heat into the environment, i.e., all the electrical power of the chip is used for heating of the coolant. This ideal model is depicted in Figure 4. The question of interest is the effect of the heat bridge geometry and material on the temperature difference between the chip and the coolant. Let us give two examples. Assume the chip surface to be 9 cm² and the chip power to be 100 W. As coolant we use water with a specific heat of 4.18 J/gK and a flow rate of say 0.5 litres per minute. Then the temperature increase of the coolant is 2.9°C, independent of the geometry and material of the heat bridge. If the cooling circuit is separated from the chip by 1 cm, the difference between the chip temperature and the average coolant temperature is 4.8°C for the aluminium and 2.8°C for the copper heat bridge. If the separation is doubled to 2 cm, the temperature difference increases to 9.6°C and 5.6°C, respectively. There is a clear performance advantage if the heat bridge is made out of copper. Let us summarise the impact of the material and geometry in this model, assuming constant inlet temperature:

- The chip temperature decreases if the thermal conductivity is increased.
- The chip temperature decreases if the distance between the chip and the coolant is decreased.
- The chip temperature decreases if the power density is decreased, i.e., the chip surface area is increased.

Of course, a real compute node consisting of several chips with different power signatures is far from being an ideal system. Realistic results for temperature distributions and heat transfer can only be achieved by computer simulations.

Irrespective of the choice of aluminium or copper, one can consider the volumetric amount of material required for the cooling design to be comparable. As discussed, copper is favourable in terms of thermal conductivity. However, there is a severe downside coming with the choice

---

4 The difference in thermal conductivity is small if the amount of admixture is rather low. However, any kind of admixture decreases the conductivity of the materials. This effect is more drastic for copper alloys.
for copper. The density of copper is very high, with huge impact on the weight of a fully populated rack. Especially for the direct coldplate approach copper is problematic since it results in a weight of several tons per rack. The nodes, each with an estimated weight on the order of 10 kg, would be difficult to service. The impact of the high density of copper on the indirect coldplate approach can be kept somewhat milder. Here the option is to manufacture only the coldplate out of copper, whereas the thermal boxes are made of aluminium. The greatest flexibility in the choice of the material is provided by the cooling pipeline approach. The materials can be chosen according to the requirements for cooling performance. Cooling of high power-density devices is very effective if heat sinks are made from copper. For low power-density devices aluminium heat bridges are typically sufficient.

Two important aspects to discuss are the robustness and the ease of manufacturing of the cooling design. In general the ease of processing, and thus the cost for manufacturing of the final design, depends on the desired shapes and also the material properties, e.g., its mechanical strength and susceptibility to heat. Both aluminium and copper (alloys) are available in a large variety of basic shapes such as sheets, bars, plates, open and closed pipes, etc. If sufficiently thin, both aluminium and copper pipes are relatively easy and especially cheap to form in a cold working process. Connections between pipes, e.g., connections between the cooling pipeline and a liquid CPU cooler, can be created by soldering, brazing or welding. Soldering is a material-friendly process at temperatures lower than 450°C, well below the melting point of the materials. Care must be taken since soldering results in reduced mechanical stability at the soldering joint. Brazing is performed at temperatures above 450°C, whereas the temperature for welding is beyond the melting point. Both techniques result in high mechanical strength, but are more complex than soldering and not applicable to all kinds of aluminium alloys. It is important to note that these connection processes potentially increase the susceptibility to galvanic corrosion. The pipe material and the process agent, e.g., soldering flux, form bimetallic junctions. The larger the gap between the metals on the galvanic series chart the higher is the susceptibility to corrosion. In contact with the coolant the behaviour of corrosion at bimetallic junctions is very different from corrosion in the bulk material. It is considered to be unpredictable and can lead to fatal material degradation if the metals in use are not compatible.

While pipes are relatively easy to process, custom-designed parts such as heat sinks, heat bridges and coldplates require more complex manufacturing. There are two basic options for such parts:

- **Milling**: The final shape is created from a block of raw material. Milling is typically performed by CNC machines. In general milling allows for high-precision parts and allows for wide flexibility in the design. While aluminium can be processed with standard tools, processing of copper typically requires special tools, e.g., special drills. This results in an increased price for manufacturing of copper components. The drawback of milling is the higher cost and the waste of raw material in form of spills that cannot be reused directly.

- **Casting**: The hot liquid material is poured into a mould (the negative for the desired shape) and solidifies. This method offers a wide range of precision and also flexibility in design. There are different methods for metal casting leading to different quality of the components, see [21] for a comparison. Especially for large production quantities and exclusive shapes casting may be cheaper than milling. A drawback of casting is that admixtures are required, e.g., silicon. Pure copper cannot be cast because of structural instability and surface deficiencies. In general the surface structures obtained from material casting are more porous compared to milling.
As a final remark on material considerations let us discuss the prices of the raw materials. The difference in prices of aluminium alloys is rather small. However, the difference in prices between materials based on aluminium and copper is very high. Pure copper is more than three times as expensive as pure aluminium [22]. In this sense copper as a basis for a compute node cooling solution is prohibitively expensive if large amounts of material are required. If, however, the amount of copper is rather small and there are clear advantages for the thermal design of the cooling, then the benefits might outrange the expenses.

2.4 Ranking Matrix

We have introduced three different kinds of solutions for direct liquid cooling in HPC. We have also illustrated their implementations in recent high-performance systems. Obviously every solution comes with a series of pros and cons. In the following we evaluate general aspects of all three types of liquid cooling solutions and display the results in a simple ranking system. We assign a plus (+) if the cooling solution has clear advantages, a minus (−) if there are clear disadvantages and a zero (○) otherwise. The results are summarised in table T1.

One important aspect is the cost of the cooling solution. We attribute the amount of raw material required, the number of connectors used for distribution of the coolant and also the manufacturing costs for custom-designed components to the cost. The cooling solutions only differ at rack-level. Other costs, such as pumps and heat exchangers, are the same for all cooling types. We consider the amount of raw material required and also the manufacturing costs to be comparable for all solutions. However, the amount of connectors required for distribution of the coolant differs. The number of connectors is minimal in the indirect coldplate approach, because multiple nodes are cooled by a single coldplate. Therefore we consider the indirect coldplate approach to be advantageous with respect to costs.

Another important aspect is the weight of the cooling design. Heavy-weight solutions may have impact on the rack design and also on node integration. In contrast, lightweight cooling solutions simplify the node integration. Furthermore structural stability requirements are relaxed, which in turn reduces the necessary material support. The weight of the rack also has an impact on the structural stability requirements for the floor space in the data centre. In terms of serviceability lightweight nodes are also easier to maintain. Since the weight of the cooling solution is proportional to the amount of material used for cooling, the cooling pipeline solution has the lowest impact on the weight of the final system.

Serviceability is the ease of hardware diagnosis, maintenance and replacement. High serviceability effectively reduces downtime and therefore increases the system productivity. We do not see any problems in the integration of sensors into any of the cooling designs. A major difference is the amount of time required to remove and re-mount the cooling from the nodes. This aspect may become important, e.g., if nodes show significantly high temperatures indicating reduced heat conductance. Here the number of parts to be accessed has impact on serviceability. Coldplate solutions are advantageous because of the low part count. The cooling pipeline solution can be more tedious to handle. Heat bridges and heat sinks are mounted to the PCB by potentially difficult to reach bolts. However, the pipeline solution allows for simple exchange of in-line memory modules. This aspect becomes important especially for large-scale installations. If, however, all devices are soldered on the PCB we do not see a major advantage of one solution over the others with respect to serviceability.

Engineering effort has an impact on the design time and thus contributes to the financial expense on the liquid cooling design. Computer simulations as well as tests of prototypes are necessary for all types of cooling. On the rack level the distribution of the coolant can be nontrivial if the number of connectors is large. On this level the indirect coldplate approach is
favourable since the number of connectors is small. On the node level the complexity depends on the number of parts and the design of mechanical and thermal interfaces. In the indirect coldplate approach the flow of heat from the devices to the coldplate is difficult to optimise since the coolant is not brought directly to the hot spots. A proper mounting system has to be designed that allows for optimal thermal conductance between the thermal box and the coldplate. In contrast, the cooling pipeline solution allows for optimal addressing of hot spots. However, this approach requires several distinct parts to be designed, e.g., heat bridges and heat sinks, which have to be thermally attached to the pipeline system. In the direct coldplate approach only a single part has to be designed. The coolant can be brought directly to hot spots. We therefore consider this approach to induce the lowest effort in engineering.

Each of the solutions has different susceptibility to coolant leakage, thus robustness is an aspect worth investigating. All connections within the cooling circuit are subject to the risk of leakage. Bolted connections, e.g., used for mounting of the coolant connectors to the coldplate, have to be adequately sealed. If using bolts is not feasible, e.g., for connections of heat sinks to a cooling pipeline, then soldering, brazing or welding are other options. Soldering is the most material-friendly process, but soldering joints may break even upon weak mechanical stress. Furthermore, bi-metal junctions can be more susceptible to corrosion than bulk material. Therefore we consider the coldplate cooling solutions to be advantageous in terms of robustness since the number of joining pieces is rather small compared to the cooling pipeline.

The final aspect is the performance of the solutions. Implementations of all cooling solutions are found in recent high performance systems. However, the range of applicability widely differs. The cooling pipeline approach allows one to bring the coolant almost directly to any hot spot on the node. The flexibility in the design of heat bridges and mounting options for the pipeline supports virtually any dimension and layout of the PCB. High power-density devices such as CPUs and accelerator cards are efficiently cooled by heat sinks through which the water flows directly. Fine-grained internal channels inside these coolers provide a large interface area for heat exchange and allow for very compact designs. This solution performs best with respect to cooling of hot spots and allows for the lowest temperature difference between the coolant and processing devices.

In the direct coldplate approach a single coldplate interfaces to the devices. The coolant is brought directly to hot spots, which makes this solution also attractive for high power density systems. One difference between the coldplate and the pipeline is that the former is typically made of aluminium, while in the latter case the heat sinks for hot spots can be made of copper. Therefore the latter can be more efficient in the cooling of hot spots.

The last solution to be discussed is the indirect coldplate. In this solution the coolant cannot be brought directly to hot spots. The flow of heat is limited to one single area which interfaces with the thermal box and the coldplate. Therefore this cooling solution is applicable only under certain conditions. Power-intensive chips such as CPUs have to be located close to the coldplate in order to achieve an acceptable temperature difference between the devices and the coolant. Support for high power-density systems is rather complicated or even impossible. If, however, the number of devices on the node is rather small and the overall power signature is sufficiently mild, then the lack of cooling performance is greatly overcome by the high packaging density that can be achieved.
Concepts for improving energy and cooling efficiency

<table>
<thead>
<tr>
<th>Cost incl. manufacturing</th>
<th>Cooling pipeline</th>
<th>Indirect coldplate</th>
<th>Direct coldplate</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>○</td>
<td>+</td>
<td>○</td>
</tr>
<tr>
<td>Weight</td>
<td>+</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>Serviceability</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>Engineering effort</td>
<td>○</td>
<td>○</td>
<td>+</td>
</tr>
<tr>
<td>Robustness</td>
<td>○</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Performance</td>
<td>+</td>
<td>-</td>
<td>+</td>
</tr>
</tbody>
</table>

Table 1: Comparison of Liquid Cooled Node Designs
Summary of the comparison of the cooling pipeline, indirect coldplate and direct coldplate cooling solutions with respect to certain aspects. The plus sign (+) indicates clear advantages, the minus sign (-) indicates clear disadvantages, and zero (○) is assigned otherwise.

2.5 Conclusions

Direct liquid cooling allows for efficient cooling of HPC systems without the need for airflow. There is a variety of cooling solutions on the node level. The cooling pipeline approach has the lowest impact on the weight of the system and the best performance with respect to cooling efficiency of hot spots. The cooling pipeline is very flexible in the design and is the only solution of liquid-cooling with in-line memory modules replaceable in the field. It is suitable even for very high power-density systems. The indirect coldplate approach is cost-efficient and provides outstanding robustness due to the reduced number of joining pieces. The increased space for compute equipment allows for the highest packaging densities achievable with water cooling nowadays. As a drawback this cooling solution is limited to systems with a moderate power signature. The direct coldplate solution combines most of the advantages of the other two solutions. It not only requires the lowest engineering effort but also provides a high level of performance and robustness. Therefore, we conclude that the direct coldplate is the preferred solution for cooling high power-density systems.
3 System Monitoring for Energy Efficiency

3.1 Motivation

Monitoring of high performance computing systems is generally performed for various reasons. Depending on the motivation for system monitoring, requirements for scope, granularity and sampling rates of the sensors contained in the system may vary. In DEEP, four different motivations for system monitoring were identified:

1. Debug and bring-up of the hardware
2. Safety, emergency shutdown and containment of damaged hardware
3. Standard monitoring and control
4. High bandwidth monitoring and control

As the name suggests, item 1 targets the debugging and bringing up of the system. This involves, for example, testing whether all cooling solutions are properly mounted – a potential issue that was already discussed in D7.1.

Item 2 is a low-level feature that prevents the hardware from taking damage in case of system failures. The prime example would be the shutting down of nodes in case of overheating due to a cooling system failure.

For items 3 and 4, we distinguish between standard cluster monitoring and control and more research oriented monitoring and control solutions. The former mainly reflects the features that cluster monitoring solutions already provide today. The latter tries to push sensor readout frequencies and the resulting data transfer bandwidth to their technical limits for gaining additional insight into the behaviour of the system.

Energy efficiency concerns will be addressed under both, items 3 and 4. Extending standard cluster monitoring toolsets with the ability to monitor power consumption already gives additional insight into the system for its operators. Especially when linking per-node power consumption to the resource management system, the Energy-to-Solution of each application can be derived. However, software developers should also have the ability to access the energy consumption information of their applications and libraries at higher granularity to pinpoint code snippets exhibiting particularly high energy consumption and to subsequently optimise their application’s Energy-to-Solution. An example of such technology can be found in the latest Intel SandyBridge CPUs which support hardware performance counters reporting the energy consumed by the CPU package and the associated RAM. This hardware feature integrates with existing performance monitoring solutions like PAPI, so accessing the energy consumption information proves very easy. Unfortunately, no such feature is available on the Intel Xeon® Phi™ (code name “KNC”) device. However, we will try to implement a comparable feature in the DEEP Booster using a special power measurement infrastructure on the Booster Node Card (BNC).

In deliverable D7.1, we reported on the dependency between inlet cooling temperatures and the power consumption of the system which was due to the increase of leakage currents in the semiconductors at higher temperatures. This observation confirmed that a holistic view of the operational state of the machine and its surrounding infrastructure for cooling is necessary. Therefore, sensor information provided by the data centre infrastructure should be included into the monitoring solution as well. Only with full awareness of the operational conditions, we will find those operating points of the system reflecting the most energy-efficient configuration.
3.2 Available Sensors in the DEEP System

Together with WP3 and WP6, thorough instrumentation for both the site infrastructures and the DEEP System was planned. The following tables give an overview of the instrumentation which will be present in the DEEP System.

Table 2 lists the data centre infrastructure sensors available at the Jülich site. Some sensors of minor value to energy efficiency considerations were omitted, e.g. valve positions, leakage sensors:

<table>
<thead>
<tr>
<th>Nº</th>
<th>Description</th>
<th>Unit</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Room air rel. humidity</td>
<td>% rel. humidity</td>
<td>Room air handling</td>
</tr>
<tr>
<td>2</td>
<td>Room air temperature</td>
<td>ºC</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>CRAC water inlet temperature</td>
<td>ºC</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>CRAC water outlet temperature</td>
<td>ºC</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>DEEP Cluster inner loop inlet temperature</td>
<td>ºC</td>
<td>Cluster water cooling</td>
</tr>
<tr>
<td>6-9</td>
<td>DEEP Cluster inner loop outlet temperature</td>
<td>ºC</td>
<td></td>
</tr>
<tr>
<td>10-13</td>
<td>DEEP Cluster inner loop flow rate</td>
<td>l/min</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>DEEP Cluster outer loop inlet temperature</td>
<td>ºC</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>DEEP Cluster outer loop outlet temperature</td>
<td>ºC</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>DEEP Cluster outer loop pressure</td>
<td>Bar</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>DEEP Booster inner loop inlet temperature</td>
<td>ºC</td>
<td>Booster water cooling</td>
</tr>
<tr>
<td>18</td>
<td>DEEP Booster inner loop outlet temperature</td>
<td>ºC</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>DEEP Booster outer loop inlet temperature</td>
<td>ºC</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>DEEP Booster outer loop outlet temperature</td>
<td>ºC</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>DEEP Booster outer loop pressure</td>
<td>Bar</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Dry coolers loop inlet temperature</td>
<td>ºC</td>
<td>Dry coolers</td>
</tr>
<tr>
<td>23</td>
<td>Dry coolers loop outlet temperature</td>
<td>ºC</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>Dry coolers loop pressure</td>
<td>Bar</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: List of DEEP Building Infrastructure Sensors

Table 3 lists the system level sensors available in the DEEP System. They can be divided into the sensors which are integrated in the KNC cards from Intel, other sensors in the DEEP Booster and sensors on the DEEP Cluster nodes.

<table>
<thead>
<tr>
<th>Nº</th>
<th>Description</th>
<th>Measuring Range</th>
<th>Resolution</th>
<th>Accuracy</th>
<th>Max. Sampling Rate</th>
<th>Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>1*</td>
<td>Board temperature (near PCIe connector)</td>
<td>0-100ºC</td>
<td>1ºC</td>
<td>3ºC</td>
<td>~20Hz</td>
<td>KNC Card</td>
</tr>
<tr>
<td>2*</td>
<td>Board temperature (East)</td>
<td>0-100ºC</td>
<td>1ºC</td>
<td>3ºC</td>
<td>~20Hz</td>
<td>Card</td>
</tr>
<tr>
<td>3*</td>
<td>Board temperature (West)</td>
<td>0-100ºC</td>
<td>1ºC</td>
<td>3ºC</td>
<td>~20Hz</td>
<td>Card</td>
</tr>
<tr>
<td>4-12</td>
<td>KNC silicon temperature</td>
<td>30-100ºC</td>
<td>1ºC</td>
<td>Unknown</td>
<td>&gt;1000Hz</td>
<td>Booster Card</td>
</tr>
<tr>
<td>13</td>
<td>KNC1 current</td>
<td>±37.5A</td>
<td>∞, analog</td>
<td>±1.5%</td>
<td>∞, analog</td>
<td>Card</td>
</tr>
<tr>
<td>14</td>
<td>KNC1 voltage</td>
<td>3.3V**</td>
<td>3mV</td>
<td>±1.5%</td>
<td>~100Hz</td>
<td>Node Card</td>
</tr>
<tr>
<td>15</td>
<td>KNC2 current</td>
<td>±37.5A</td>
<td>∞, analog</td>
<td>±1.5%</td>
<td>∞, analog</td>
<td>Card</td>
</tr>
</tbody>
</table>

1 Data centre infrastructure sensors that will be available at the BADW-LRZ site are still under consideration.
Table 3: List of DEEP System Sensors

*: DFF cards will carry only one temperature sensor!
**: Measurement range extended outside sensor to measure on supply rail!

<table>
<thead>
<tr>
<th>Sensor</th>
<th>Range 1</th>
<th>Range 2</th>
<th>Error 1</th>
<th>Error 2</th>
<th>Measurement Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>KNC2 voltage</td>
<td>3.3V**</td>
<td>3mV</td>
<td>±1.5%</td>
<td>~100Hz</td>
<td></td>
</tr>
<tr>
<td>BNC current</td>
<td>±37.5A</td>
<td>analog</td>
<td>±1.5%</td>
<td>infinite</td>
<td></td>
</tr>
<tr>
<td>BNC voltage</td>
<td>3.3V**</td>
<td>3mV</td>
<td>±1.5%</td>
<td>~100Hz</td>
<td></td>
</tr>
<tr>
<td>BNC temperature</td>
<td>-55-125ºC</td>
<td>±0.0625ºC</td>
<td>2ºC</td>
<td>~10Hz</td>
<td></td>
</tr>
<tr>
<td>BNC humidity</td>
<td>0-100%rH</td>
<td>0.05%rH</td>
<td>±4.5%</td>
<td>~0.1Hz</td>
<td></td>
</tr>
<tr>
<td>Node supply current</td>
<td>±37.5A</td>
<td>analog</td>
<td>±1.5%</td>
<td>infinite</td>
<td></td>
</tr>
<tr>
<td>Node supply voltage</td>
<td>3.3V**</td>
<td>3mV</td>
<td>±1.5%</td>
<td>~100Hz</td>
<td></td>
</tr>
<tr>
<td>Node temperature</td>
<td>-55-125ºC</td>
<td>±0.0625ºC</td>
<td>2ºC</td>
<td>~10Hz</td>
<td></td>
</tr>
<tr>
<td>Node humidity</td>
<td>0-100%rH</td>
<td>0.05%rH</td>
<td>±4.5%</td>
<td>~0.1Hz</td>
<td></td>
</tr>
</tbody>
</table>

3.3 Sensor Acquisition

With all those sensors available in the DEEP System, getting access to the data produced by the sensors becomes an important task.

Information from the site infrastructure sensors will be available through a web interface, a SQL database backend as well as the low-level BacNet protocol [23].

In the DEEP System itself, all sensor values will be forwarded to the respective Board Management Controller (BMC). From there, they can be propagated through a Gigabit Ethernet based management network to a central management framework. For this, standard protocols such as IPMI and SNMP will be used. Some of the sensors are also accessible from outside the node (without interaction from the BMC) to ensure safe operating conditions even before the system is turned on. This feature is a result of the considerations for “Safety, emergency shutdown and containment of damaged hardware” and thus, it is proprietary to the system integrator and goes beyond the scope of this document.

The exact design of the management framework gathering all sensor information is still under discussion among the involved partners of the DEEP project but a baseline concept can already be described.

From an abstract point of view, the sensor management framework consists of 4 different components: a central controller, its associated data store, a list of sensor agents delivering the sensor data to the controller, and a user interface. Figure 5 shows such an abstract sensor management framework and the interaction between the components.

The controller acts as the central management entity. It keeps a list of all the sensors that deliver data to it and an interface to the data store. The controller has control over a list of sensor agents that push sensor data to it. Of course, the sensor agents can run on the same physical machine as the controller to mimic a pull model of data acquisition.

The data store can be implemented in several ways. Round-robin databases are often used in cluster management as the data store. They pre-allocate the required disk space and manage aggregation of older values. Since round robin databases are typically stored as binary files, their performance is typically limited by the underlying file system.

Another option for a data store would be a relational database management system. Relational database management systems are flexible and easy to use. However, since disk space is typically not pre-allocated, care must be taken to delete old values from time to time to avoid full disks. Also, relational database management systems are rather difficult to design when performance is a concern. By ensuring ACID (atomicity, consistency, isolation, durability) properties, most relational database management systems only provide limited scalability.
In-memory databases are an option for system monitoring as their design would allow for keeping the latest values of each sensor in RAM and writing older values to disk. Thus, older values would still be accessible, yet, with higher latency.

An even better option for the data store which would exhibit even better scalability will be presented below. Since the required effort for implementing the different approaches varies depending on the reusability of existing solutions, a final decision on this topic is still pending.

Finally, the system consists of a user interface. This can be graphical (e.g. web based) or accessible through the command line. Control can be exercised by the user through control channels to the controller.

**Figure 5: General Concept of a Sensor Management Framework**

### 3.4 Scalability of the System Management Framework

With the expected size of Exascale systems and the need for thorough monitoring in mind, the system management framework must prove extreme scalability. Two possible approaches to achieve this level of scalability will be presented in this section. Common to both approaches is the introduction of further controllers in order to avoid a single controller becoming a bottleneck with an increasing number of sensors.

The first approach for scalability introduces a hierarchical concept of controllers. By providing an interface for control and an interface for delivering data, the black box view of a controller is similar to the one of a sensor agent. Thus, controllers can be seen as, and act in a similar way as sensor agents. The resulting concept is shown in Figure 6.
While at the lowest level, each controller still has its associated data store and its associated sensors, arbitrary levels of super controllers can collect data from controllers at lower levels. After selecting the most interesting sensors or aggregating data from multiple sensors, the data is forwarded to the next super controller up in the hierarchy or to the user interface. Since super controllers have their own data store to keep values, data can be cached for higher performance.

Although easy to implement, the drawback of this approach is that data is spread over multiple data stores and care has to be taken in the design of the system to ensure that data which is needed frequently will be cached at the super controllers.

The second approach would overcome this problem by relying on a single distributed data store (see Figure 7). A reasonable implementation for such a distributed data store is a key-value store (e.g. Google BigTable, Apache Cassandra) which distributes data to different storage servers according to a predefined mapping function. The distribution onto different storage servers causes an implicit load balancing for both write and read accesses to the data store. This way, even multiple instances of user interfaces can submit queries simultaneously. Scaling up by adding more sensors or increasing the sensor or by increasing the readout frequency is doable by adding more servers as controllers or as data store backend systems and the monitoring solution will keep up with the increased load.
Scalability through distributed data store: All controllers access the same (distributed) data store. If necessary, abstraction is handled at user interface level. Concurrent access from multiple user interfaces is also possible and scales.

Figure 7: Scalable Sensor Management Framework using a Distributed Data Store
4 System Management for Energy Efficiency

4.1 Motivation

With extensive instrumentation at hand, optimisation of the operational parameters of the machine and the infrastructure towards higher energy efficiency is feasible. It is important to note that the biggest improvements in energy efficiency of a supercomputer can be achieved through performance optimisation of the applications. Selecting the best algorithms or using a supercomputer which provides a hardware architecture suitable for the given problem are just two approaches to achieve improvements in energy efficiency that go far beyond the ideas discussed below. However, performance optimisation is a non-trivial task and thus, automation of performance tuning is almost impossible. Therefore, we will focus on approaches that can be applied in a more automated, or at least, semi-automated fashion. This way, the techniques discussed below could become part of a site policy that balances the needs of users who require higher compute performance with the needs of the supercomputing sites who often need to meet power budget constraints.

For the DEEP System, we will evaluate three different approaches for energy savings in parallel applications that have been described in literature: voltage and frequency scaling, concurrency throttling and thread pinning.

4.2 Voltage and Frequency Scaling

Voltage and frequency scaling is a technology that allows modern processors to adapt the CPU frequency and the CPU supply voltage to the actual need. In high performance computing voltage and frequency scaling potentially improves the energy efficiency of applications that are memory bound or which exhibit imbalances in their parallelisation. The savings are achieved by running the processors at frequencies lower than the nominal frequency with the reduction in power consumption of the processor outweighing the increase in runtime.

To assess the impact of voltage and frequency scaling, we conducted experiments on two existing systems at BADW-LRZ: CooLMUC and SuperMUC.

CooLMUC consists of 178 nodes, each of which carries two 8-core AMD 6128 HE processors (MagnyCours) and 16 GB RAM. SuperMUC consists of 9216 nodes. Each node in SuperMUC contains two 8-core Intel Xeon E5-2680 CPUs (Sandy Bridge) and 32 GB RAM.

With 16 cores per node, an InfiniBand interconnect and a diskless design, both systems share principal design features. For our test, we used the HYDRO benchmark, which is a computational fluid dynamics code using a Riemann solver on a regular 2D mesh [24]. In contrast to synthetic CPU, memory or I/O intensive benchmarks, HYDRO represents the typical behaviour of applications in the field of computational fluid dynamics. HYDRO has been ported to a large number of platforms and programming models. For our experiments, we used the MPI only version on 16 nodes.

In our test setup we ran the HYDRO benchmark at different CPU frequencies and measured both the Time-to-Solution (Figure 8 and Figure 10) and the Energy-to-Solution (Figure 9 and Figure 11) for a fixed problem size. Similar to the Time-to-Solution, which is the execution time of the application on a given supercomputer for a given problem statement, the Energy-to-Solution of an application is defined as the energy consumed by all the hardware components necessary to run the application during the application’s execution time. For better visibility of the potential savings, the y-axis shows the savings through DVFS relative
to the processor’s nominal frequency. As expected lowering the processor frequency leads to lower compute performance and thus increased Time-to-Solution. CooLMUC shows an unexpected increase in runtime at 1.7 GHz for which we could not find an explanation. In terms of Energy-to-Solution the most energy-efficient configuration for SuperMUC is at 1.8 GHz, whereas on CooLMUC it is at the nominal frequency of 2.0 GHz.

We observe that voltage and frequency scaling can be successfully used on SuperMUC to lower the Energy-to-Solution. Yet, on CooLMUC, lowering the processor frequency does not improve the energy signature. The different results of the two systems indicate that the behaviour is highly dependent on the hardware architecture, with some architectures offering more opportunities for power savings than others. Since the DEEP System is a completely new architecture, we will evaluate the behaviour of the DEEP System with respect to voltage and frequency scaling to see if we can use this technique to lower the Energy-to-Solution.
Concepts for improving energy and cooling efficiency

Figure 8: Time-to-Solution of HYDRO Benchmark on CoolMUC

Figure 9: Energy-to-Solution of HYDRO Benchmark on CoolMUC
Concepts for improving energy and cooling efficiency

Figure 10: Time-to-Solution of HYDRO Benchmark on SuperMUC

Figure 11: Energy-to-Solution of HYDRO Benchmark on SuperMUC


### 4.3 Concurrency Throttling

Concurrency throttling is motivated by the fact that most scientific applications exhibit a non-linear scaling curve. This means that from a given point on, using more cores for the same computation does no longer yield a reasonable benefit. Additional cores do, however, consume power and will thus increase the Energy-to-Solution. Concurrency throttling works by oversubscribing physical cores with more than one thread per core. This leaves the decision which thread to run at what time to the operating system’s scheduler which can select among all runnable threads. In addition, concurrency throttling can also be used to mitigate imbalances in the parallelisation by backfilling threads onto cores that would normally remain idle until the next synchronisation point.

Curtis-Maury et al. have published multiple papers, e.g. [25], on the topic of concurrency throttling for energy efficiency. Once the DEEP System is available, we will conduct experiments to see, in how far concurrency throttling is a successful means of saving energy on the DEEP machine.

### 4.4 Thread Pinning

In a multi-core computer, the operating system’s scheduler is responsible for assigning threads to available CPU cores. This task is typically performed dynamically, taking into account multiple parameters like fairness, system load, process priority, etc. In an HPC environment, however, such dynamic scheduling can have detrimental effect on performance. For example, re-assigning a thread from one CPU core to another is usually undesirable as it introduces unnecessary overhead for the actual transfer of the thread as well as for re-filling the cache. Therefore, it is good practice to pin each thread to its very own core and avoid migration afterwards. Thread pinning also allows for taking application characteristics into account about which the scheduler typically has no information. The communication overhead, for example, can be reduced by placing threads that frequently communicate with each other on neighbouring cores and hence facilitating exploitation of low latency communication via shared caches within a CPU package. Given that a pinning has no influence on the energy consumption of the processor (all cores would be running full-throttle anyway), the right pinning not only reduces the Time-to-Solution but also the Energy-to-Solution.

However, finding the right pinning for a given application can be cumbersome, especially since different phases of the application might even require different pinnings. Yet, this process can be automated by continuously assessing an application’s performance during runtime, e.g. by means of performance counters, with regard to different pinning configurations. For long-running applications a viable approach would be to probe interesting configurations for a few minutes after program start-up and then use the best performing configuration afterwards. Different program phases with different pinning requirements could be detected by observing the average performance in a certain time frame: if it drops significantly, the other pinnings will be re-probed and the program will continue with the best performing configuration. For short running applications, performance data from previous program runs could be considered.
References and Applicable Documents

http://www.engineershandbook.com/Tables/castchart2.htm,


https://github.com/HydroBench/Hydro.

Schulz, “Prediction models for multi-dimensional power-performance optimization on
many cores,” in Proceedings of the 17th international conference on Parallel
architectures and compilation techniques, Toronto, Ontario, Canada, 2008.
List of Acronyms and Abbreviations

A
ASIC: Application Specific Integrated Circuit: Integrated circuit customised for a particular use
Aurora: The name of Eurotech’s cluster systems

B
BADW-LRZ: Leibniz-Rechenzentrum der Bayerischen Akademie der Wissenschaften.Computing Centre, Garching, Germany
BlueGene/Q: Supercomputing architecture developed by IBM, well known for its energy efficiency, massive parallelism, 5D torus network and wide vector units.
BMC: Baseboard Management Controller
BN: Booster Node (functional entity)
BNC: Booster Node Card: A physical instantiation of the BN
Booster System: Hardware subsystem of DEEP comprising of BNC, BIC and Intra-Booster network
BoP: Board of Partners for the DEEP project

C
CNC: Computerised Numerical Control
CooLMUC: Prototype at BADW-LRZ with direct warm water cooling
CPLD: Complex Programmable Logic Device
CPU: Central Processing Unit
CRAC: Computer room air conditioning

D
DEEP: Dynamical Exascale Entry Platform: EU-FP7 Exascale Project led by Forschungszentrum Juelich
DEEP Architecture: Functional architecture of DEEP (e.g. concept of an integrated Cluster Booster Architecture)
DEEP Booster: Booster part of the DEEP System
DEEP System: The production machine based on the DEEP Architecture developed and installed by the DEEP project
DFF: Dense Form Factor

E
EC: European Commission
EU: European Union
Eurotech: Eurotech S.p.A., Amaro, Italy

F
FLOP: Floating point Operation
FPGA: Field-Programmable Gate Array: Integrated circuit to be configured by the customer or designer after manufacturing
G
GREEN500 list: Provides rankings of the 500 top most energy-efficient supercomputers in the world

H
HPC: High Performance Computing

I
ICT: Information and Communication Technologies
Intel: Intel GmbH Braunschweig, Germany
Intel Xeon® Phi™: official product name of the Intel Many Core (MIC) architecture processors. The first available Intel Xeon® Phi™ product is code-named Knights Corner (KNC).
I/O: Input/Output
IPMI: Intelligent Platform Management Interface

J
JUELICH: Forschungszentrum Jülich GmbH, Jülich, Germany

K
KNC: Knights Corner: Code name of a processor based on the MIC architecture. The commercial name of this product is Intel Xeon® Phi™

L
LINPACK: Software library to perform numerical linear algebra calculations used as benchmark

M
MIC: Intel Many Integrated Core architecture
MPI: Message Passing Interface: API specification typically used in parallel programs that allows processes to communicate with one another by sending and receiving messages

P
PAPI: Performance Application Programming Interface
PCB: Printed Circuit Board: Board used in electronic to mechanically support and electrically connect electronic components
PCI: Peripheral Component Interconnect: Computer bus for attaching hardware devices in a computer
PCIe: PCI Express: Standard for peripheral interconnect, developed to replace the old standards PCI, improving their performance
PHY: Physical Network Layer (Chip)
PMT: Project Management Team of the DEEP project
PSU: Power Supply Unit
D7.2 Concepts for improving energy and cooling efficiency

Q
QPACE: Specialised supercomputer for QCD Parallel Computing on CELL processors

S
SNMP: Simple Network Management Protocol

U
UniReg: University of Regensburg, Germany

W
WP: Work Package