D5.6– Prototype demonstration of performance monitoring tools on a system with multiple ARM boards
Version 1.0

Document Information

Contract Number	288777
Project Website	www.montblanc-project.eu
Contractual Deadline	M24
Dissemintation Level	PU
Nature	Prototype
Coordinator	Alex Ramirez (BSC)
Contributors	Chris Adeniyi-Jones (ARM)
Reviewers	Petar Radojković (BSC)
Keywords	Performance monitoring tools, PAPI, perf, Ganglia

Notices: *The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement n^o 288777*

Change Log

Version	Description of Change
v0.1	Version released to Internal Reviewer
v1.0	Final version sent to EU

Contents

Executive Summary

In this deliverable, we present the current status of the low-level software components required for gathering information about the performance of HPC applications running on ARM-based systems. This work will enable performance monitoring tools to be ported to the Mont-Blanc prototype.

1 Introduction

With its aim to build a large-scale HPC cluster using embedded mobile technology, a substantial part of the Mont-Blanc project consists of porting and tuning applications to the target architecture. Tuning these applications requires detailed performance information and tools to visualize that information. One of the goals of Work package 5 of the Mont-Blanc project is to enable the use of ARM performance monitoring hardware for detailed analysis of performance of HPC applications. This report summarizes our work in this regard.

The rest of this report is organized as follows:

- Section 2 summarizes the status of the Performance Application Programming Interface (PAPI) [pap] for the ARM architecture. PAPI library provides a consistent interface and methodology for using the performance counter hardware found in modern microprocessors.
- Section 3 describes various performance monitoring tools for the ARM architecture.
- Section 4 outlines the plan for a demonstration of performance monitoring tools running on several ARM-based boards.

2 Performance Application Programming Interface

The current release of PAPI is 5.2.0 and supports the ARM architecture using the perf_events driver on Linux kernels 2.6.32 and above. The perf_events interface is built into these kernels and can be used directly requiring no patches.

PAPI provides two interfaces to the underlying counter hardware; a simple, high level interface for the acquisition of simple measurements and a fully programmable, low level interface directed towards users with more sophisticated needs. The low level PAPI interface deals with hardware events in groups called EventSets. EventSets reflect how the counters are most frequently used, such as taking simultaneous measurements of different hardware events and relating them to one another.

3 Performance Monitoring Tools

3.1 perf

The perf [per] userspace tools use the perf_events kernel ABI to provide access to hardware performance counters. They can be used as a basis for profiling applications to trace dynamic control flow and identify hotspots.

3.2 Ganglia

Ganglia [Gan] is a scalable distributed monitoring system for high-performance computing systems such as clusters and Grids. It is based on a hierarchical design targeted at federations of clusters. It leverages widely used technologies such as XML for data representation, XDR for compact, portable data transport, and RRDtool for data storage and visualization. It uses carefully engineered data structures and algorithms to achieve very low per-node overheads and high concurrency. The implementation is robust, has been ported to an extensive set of operating systems and processor architectures, and is currently in use on thousands of clusters around the world. It has been used to link clusters across university campuses and around the world and can scale to handle clusters with 2000 nodes.

We have installed the Ganglia monitoring system on an Arndaleboard and tested that the default metrics can be measured and reported via the Ganglia web-frontend. We have also tested the addition of custom metrics such as CPU temperature and CPU clock frequency.

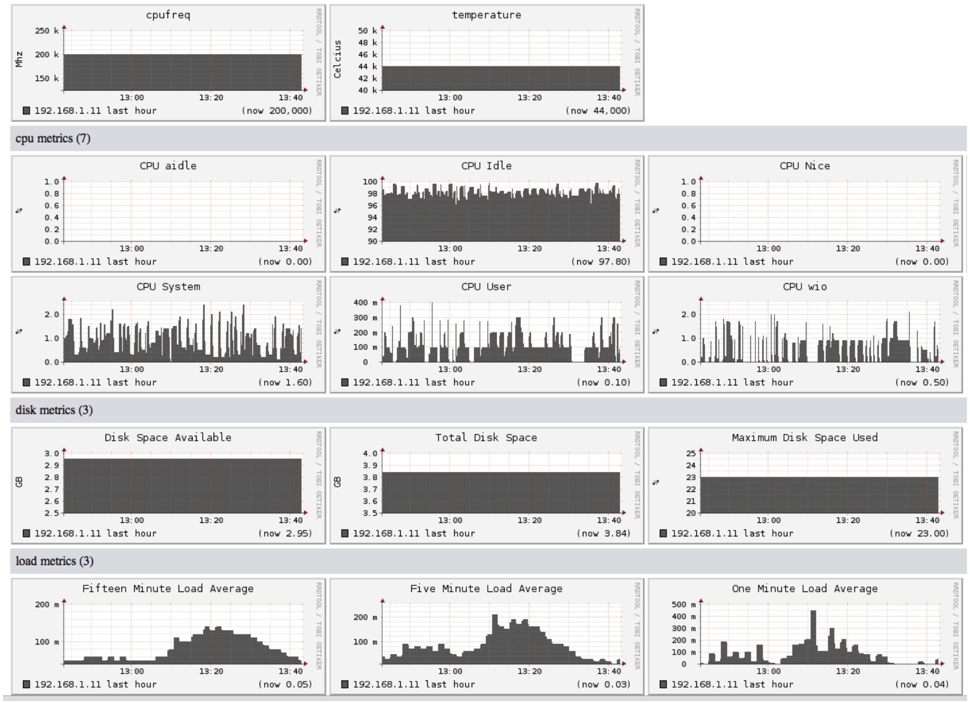

Figure 1: Example output from the Ganglia web-front end monitoring a single Arndaleboard

4 Demonstration

We plan to demonstrate the Ganglia tool being used to monitor several Arndaleboards. We will also show how the userspace perf tools can be used for low-level profiling of applications.

References

[Gan] Ganglia monitoring system. `http://ganglia.sourceforge.net/`.

[pap] Performance Application Programming Interface. `http://icl.cs.utk.edu/papi`.

[per] perf: Linux profiling with performance counters. `http://perf.wiki.kernel.org/index.php/Main_Page`.