G E N I

Global Environment for Network Innovations

Milestone S2.a
Unified Measurement Framework:
NetFPGA Cube Prototype

Document ID: ERM_S2a_Dec09

December 03, 2009

Prepared by:
C.P. Lai, M.S. Wang, and K. Bergman
Dept. of Electrical Engineering, Columbia University New York,
500 W. 120th Street,
New York City, NY 10027
under Project Nr. 1631
“Embedding real-time measurements for cross-layer communications”

TABLE OF CONTENTS

1 Document Scope

This section describes this document's purpose, its context within the overall GENI project, the set of related documents, and this document's revision history.

1.1 Executive Summary

This technical note presents the outcome of the work package Milestone S2.a of Project Nr. 1631, "Embedding real-time substrate measurements for cross-layer communications." This milestone document comprises the first deliverable of ERM in Spiral 2 and involves the design and development of the NetFPGA Cube implementation of the proposed unified measurement framework (UMF).

In the scope of Spiral 1, ERM assessed the GENI requirements for real-time measurements. In order to appropriately limit the overhead and complexity associated with gathering and accessing measurement data, we proposed the realization of a unified measurement framework (UMF). The UMF serves as a means for gathering physical layer measurements and conveying the data to the GENI researchers in an aggregated, unified way. We also evaluated the corresponding networking protocol and management languages that will be essential in our implementation of the UMF. As a first step toward GENI integration, we have joined the ORCA Cluster (D) and plan to integrate our concept of the UMF within the ORCA-BEN aggregate. This will enable real-time measurement experimentation using the existing networking elements (NEs) embedded in the BEN metro-area optical network.

In the scope of Spiral 2, the first major task is to design an implementation of the UMF that can be integrated within the ORCA Cluster. Further, this design should be general enough such that it can be easily extended to other GENI control frameworks in the future. This milestone document discusses this design and development of the UMF by means of a NetFPGA Cube. The NetFPGA Cube [netfpga_1] is an integrated system composed of a general purpose processor, in addition to the proprietary NetFPGA hardware [netfpga_2]. The UMF comprises of both a software component (run on the general purpose processor), as well as a hardware component (run on NetFPGA card). As explained in this milestone report, each component has a defined role in facilitating the UMF to access the networking elements and its measurement data.

1.2 Related Documents

The following documents are related to this document, and provide background information, requirements, etc., that are important for this document.

1.2.1 GENI Documents

Document ID	Document Title and Issue Date
GENI_QR_ERM_Jul09	2Q09 Status Report
GENI-QR-ERM-Oct09_rev2	3Q09 Status Report
GENI-MS1-ERM-March09-v1.1	Technical Note 1, Milestone 1

GENI-MS2-ERM-March09-v1.0	Technical Note 2, Milestone 2
GENI_MS4_ERM_June09_v1-0	Technical Note 4, Milestone 4
GEC4_Columbia_ERM	GEC4 Presentation

1.3 Document Revision History

The following table provides the revision history for this document, summarizing the date at which it was revised, who revised it, and a brief summary of the changes. This list is maintained in chronological order so the earliest version comes first in the list.

Revision	Date	Revised By	Summary of Changes
1.0	02 Dec 09	C. P. Lai	Initial draft
1.1	02 Dec 09	M. S. Wang	Revised draft
1.2	03 Dec 09	C. P.Lai	Revised draft

2 UMF Overview

To evaluate the GENI requirements for real-time user access to measurement data, we have previously assessed the capabilities of GENI's future infrastructure with respect to real-time measurements [erm09_1]. As referenced in [erm09_2], interfacing available NEs that have embedded performance monitoring capabilities to the control framework and to the access point and user tools of the GENI researcher in a straightforward way may lead to the some of the following obstacles:

- Depending on the number of available NEs, the number of required interfaces might become very large.
- Designing the interfaces requires detailed, vendor-specific knowledge about the control mechanisms of the NEs and of how measurement data is exported. Thus, some level of abstraction would be highly desirable.
- Some amount of editing and preparation of the measured data prior to the delivery to the researcher could be desirable.
- Extensibility of the measurement framework and manageability may be difficult if each NE has to be controlled individually.

Therefore, rather than interfacing every performance monitoring device embedded within the substrate directly with the control framework and the experimenter, we recommended the design and use of the UMF [erm09_2, erm09_3]. The UMF represents a universal measurement platform which can be accessed by the control framework and the researcher via a limited number of well-defined interfaces. The main tasks and functionalities of the UMF are:

- Acquiring measurement data from the NEs within the optical substrate,
- abstraction of measurement capabilities and equipment from several manufacturers,
- provide a single point of access,
- basic processing of the measurement data, e.g., to extrapolate signaling data for the researcher,
- provisioning of some storage capacity for non-time-sensitive measurements,
- interfacing to the researcher via a unified measurement interface which allows requesting and controlling certain measurements and delivering the measured data, and
- interfacing with the control framework so that the measurement framework (or a subset of NEs) can be allocated to a requesting GENI researcher and reconfiguration of the slice is made possible.

As such, the UMF presents a uniform view and an abstraction of the measurement capabilities within the substrate and makes them accessible/sliceable to a control framework.

In Spiral 1, the concept and high level architectural design of the UMF was proposed [erm09_4]. As the first step in Spiral 2, this milestone document details a more specific prototype of the UMF. In Section 3, we outline a NetFPGA Cube-based implementation of the UMF that can be integrated within the ORCA Cluster [orca_1], while maintaining a certain level of generality such that the design can be easily extended to other GENI control frameworks in the future.

3 NetFPGA Cube Implementation

In this section, we examine a NetFPGA Cube-based implementation of the UMF. In Section 3.1, we present an architectural overview of this implementation, and then look at the software component and hardware component in Sections 3.2 and 3.3, respectively.

3.1 Overview

The main goal of the UMF is to present a uniform view and an abstraction of the measurement capabilities within a substrate and make them accessible/sliceable to a control framework. As such, the UMF is required to interface with both the GENI control framework, as well as to a set of NEs within the GENI network substrate. Figure 3-1 shows an architectural flowchart of how the UMF interfaces specifically to the ORCA control framework and its NEs. The green dotted lines depict the flow of measurement commands, such as the signal monitoring commands, downstream from the ORCA control framework, through the UMF, down to the underlying NEs. The blue solid lines show the flow of retrieved measurement data from the NEs, up to the UMF and the ORCA control framework to be processed or stored.

The UMF interfaces with the ORCA control framework via the integrated measurement framework (IMF). The ORCA control framework is in charge of managing the network resources and allocating slices of the available resources to the GENI researchers. ORCA receives the measurement data from the IMF and can choose to store it locally, or send it to the GENI users' external tools for further processing, or storage.

The IMF is further responsible for interfacing the UMF to the Services Integration, controL, and Optimization (SILO) framework, which is an infrastructure for a non-layered internetworking architecture in which complex communication tasks are accomplished by combining functional blocks in a configurable manner [silo09_1]. The cross-layered experimentation capability of SILO can be combined with the unified measurement capability of UMF to enable substrate measurement as a service in a custom protocol stack (i.e. a silo) [imf_1].

The UMF also interfaces to a set of NEs in the substrate, including the Polatis switch [polatis_1] and Infinera Digital Transport Node (DTN) [infinera_1]. Since different NEs have different, vendor-specific control mechanisms, the UMF will have to provide different interfaces for each NE. Specifically, the software for interfacing the Infinera DTN to the IMF is called the Measurement Handler Software (MHS) and is developed in conjunction with the GENI LEARN project [learn_1].

As a first hardware realization, we implement the UMF using the NetFPGA Cube, which is an integrated system composed of a general purpose processor, in addition to the proprietary NetFPGA hardware. The UMF comprises of both a software component (which runs on the general purpose processor), as well as a hardware component (which runs on the NetFPGA card). We outline the functionalities of software and hardware components in the following sections.

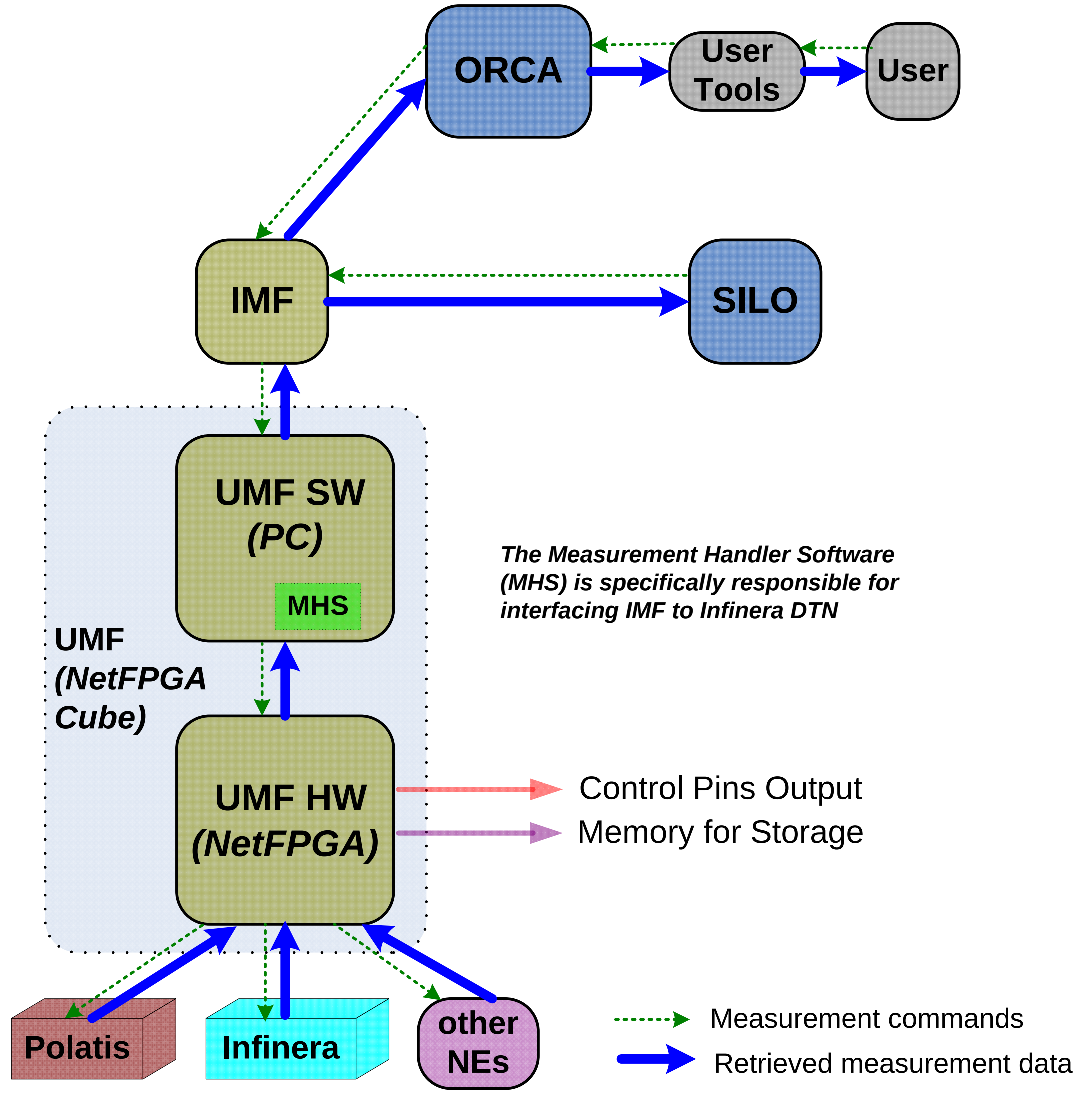

Figure 3-1: Architectural Flowchart of how UMF fits within ORCA Control Framework

3.2 Software Component

The software component of the UMF is responsible for interfacing with the IMF. The IMF sends the following measurement command information to the UMF SW:

- Slice allocation information
- ID of NE to make the measurement (Polatis, Infinera, or others)
- Measurement metric (such as power, bit-error rate (BER), etc.)
- Measurement metric update rates (e.g. how often to make the measurement)
- Final destination of the retrieved measurement data for processing or storage (ORCA, SILO, User tools, or UMF)

Upon receiving this information, the UMF SW issues the following measurement command information to the UMF HW:

- Vendor-specific command to retrieve specified measurement metric from specified NE (uses TL1, SNMP, SCIP, etc. depending on vendor)
- Measurement metric update rates (e.g. how often to make the measurement). Could be:
  - Issue once per application-specified time interval
  - Issue once per user-specified time interval
  - Issue only once
- Direction on where to send the retrieved measurement data for processing or storage (ORCA, SILO, User tools, or UMF)

One should note the different roles played by the IMF and UMF. The IMF is in charge of interfacing with the ORCA and SILO frameworks, and the applications running on them. It then communicates with the UMF which measurement metric the higher level applications wish to retrieve and what NE to use. Then, the UMF is responsible for actually interfacing with all available NEs, by sending vendor-specific commands to them and receiving measurements from them. Once the UMF SW receives the measurement information from the NEs, it can store it locally or forward it up to the ORCA, SILO, or User tools for processing and/or storage.

3.3 Hardware Component

The UMF HW is implemented via the specialized NetFPGA card/hardware. It is composed of a Virtex-II Pro 50 FPGA, SRAM, DDR2 DRAM, multi-gigabit I/O (SATA), and 4 gigabit Ethernet ports. As many of the NEs have an Ethernet interface, the multiple Ethernet ports on the NetFPGA provides additional convenience in interfacing to various NEs simultaneously. Further, the multiple types of I/Os available allow the NetFPGA to be used to control various types of local hardware that require such kinds of specialized I/Os.

We leverage the UMF HW to provide timing-sensitive processing. If the measurement command information sent to the UMF HW by the UMF SW specifies a measurement to be made only once, then there is no time sensitivity. In this case, the vendor-specific measurement command sent from the UMF SW can be directly forwarded to the actual NE. However, if the measurement metric update rate is some fixed time interval, then the UMF HW will be in charge to keeping count of this time interval while repeatedly sending out the measurement command to the actual NE. The UMF HW will also manage (or store) the upward flow of measurement data should the NE be set to stream measurement information.

Upon receiving the retrieved measurement data from the NE, the UMF HW can directly process the data and control/actuate some local hardware using the special I/Os available to the NetFPGA (such as GPIO, Ethernet, Serial, etc). This is only done if it is specified that the UMF HW should perform the processing. Otherwise, the UMF HW forwards the retrieved measurement data upstream for processing or storage.

4 Summary and Conclusions

In this report, we discuss our latest Spiral 2 milestone of designing and developing a hardware implementation of the UMF. We provide an overview of the UMF, as determined from our Spiral 1 work, and subsequently describe the software and hardware components that comprise the UMF. Both of these components are leveraged in accessing the embedded networking elements in order to obtain real-time measurement data. We provide a NetFPGA Cube-based implementation of the UMF that can be integrated within the ORCA Cluster, while maintaining a certain level of generality such that the design can be easily extended to other GENI control frameworks in the future.

5 Bibliography

[1] [netfpga_1] NetFPGA Cube main homepage. [Online]. Available: http://www.accenttechnologyinc.com/netfpga.php?category_id=0&item_id=1
[2] [netfpga_2] NetFPGA website. [Online]. Available: http://www.netfpga.org/
[3] [erm09_1] F. Fidler, C. P. Lai, and K. Bergman, "Technical Note 1: GENI Requirements for Real-Time Measurements (Project Nr. 1631, Milestone 1)," (2009, February) [Online]. Available: http://groups.geni.net/geni/wiki/Embedded%20Real-Time%20Measurements
[4] [erm09_2] C. P. Lai, F. Fidler, and K. Bergman, "Technical Note 2: Specifications and Networking Protocols (Project Nr. 1631, Milestone 2)," (2009, February) [Online]. Available: http://groups.geni.net/geni/wiki/Embedded%20Real-Time%20Measurements
[5] [erm09_3] F. Fidler, "Unified Measurements," 4th GENI Engineering Conference (2009, April) [Online]. Available: http://groups.geni.net/geni/wiki/Embedded%20Real-Time%20Measurements
[6] [erm09_4] F. Fidler, C. P. Lai, and K. Bergman, "Technical Note 4: GENI Real-Time Measurements Software Architecture (Project Nr. 1631, Milestone 4)," (2009, June) [Online]. Available: http://groups.geni.net/geni/wiki/Embedded%20Real-Time%20Measurements
[7] [orca_1] ORCA main homepage. [Online]. Available: http://nicl.cod.cs.duke.edu/orca/
[8] [silo09_1] Net-Silos Team, "SILO Project – Services, Integration, controL and Optimization for the Future Internet," (2009, May) [Online]. Available: http://www.net-silos.net
[9] [imf_1] GENI IMF Wiki. [Online]. Available: http://groups.geni.net/geni/wiki/IMF
[10] [polatis_1] Polatis main homepage. [Online]. Available: http://www.polatis.com/
[11] [infinera_1] Infinera DTN [Online]. Available: http://www.infinera.com/products/DTN.html.
[12] [learn_1] GENI LEARN Wiki. [Online]. Available: http://groups.geni.net/geni/wiki/LEARN