InfiniBand INTERCONNECTS FOR HIGH-THROUGHPUT DATA ACQUISITION IN A TANGO ENVIRONMENT

T. Dritschler*, S. Chilingaryan*, T. Farago†, A. Kopmann*, M. Vogelgesang*
*Institute for Data Processing and Electronics
†Institute for Photon Science and Synchrotron Radiation
Karlsruhe Institute of Technology, Germany

Abstract

Advances in computational performance allow for fast image-based control. To realize efficient control loops in a distributed experiment setup, large amounts of data need to be transferred, requiring high-throughput networks with low latencies. In the European synchrotron community, TANGO has become one of the prevalent tools to remotely control hardware and processes. In order to improve the data bandwidth and latency in a TANGO network, we realized a secondary data channel based on native InfiniBand communication. This data channel is implemented as part of a TANGO device and by itself is independent of the main TANGO network communication. TANGO mechanisms are used for configuration, thus the data channel can be used by any TANGO-based software that implements the corresponding interfaces. First results show, that we can achieve a maximum bandwidth of 30 Gb/s which is close to the theoretical maximum of 32 Gb/s, possible with our 4xQDR InfiniBand test network, with average latencies as low as 6 µs. This means that we are able to surpass the limitations of standard TCP/IP networks while retaining the TANGO control schemes, enabling high data throughput in a TANGO environment.

INTRODUCTION

PC-based systems have made significant advances in stability and computational power and have become viable for usage as control systems even in large installations. Driven by this influx of 'off-the-shelf' PC systems, the ESRF started to develop its own remote control system called TANGO [1] in 2003. TANGO provides transparent and uniform access to all devices on the control network. It uses the CORBA communication layer to send function calls to any remote device that provides a TANGO interface.

Since TANGO was created mainly as a control system for the synchrotron community and is distributed as an open source system, it found broad acceptance across the European synchrotron community and is now actively being developed by a large consortium, lead by ESRF. Eventually, the ANKA synchrotron at KIT has also decided to extensively use TANGO as one of the control system for their beamlines.

The new IMAGE beamline is being constructed at ANKA to allow ultra fast 3D tomography with near real-time monitoring and fast image-based control loops. The beamline development aims to permit investigations of the internal morphology and structural changes in small living organisms in 4D (3D + time) with micrometer spatial resolution and sub-second time resolution [2]. The required resolution is achieved with high-speed cameras providing over a thousand frames per second and with streaming bandwidth ranging from a few hundred Megabytes up to multiple Gigabytes per second. The image-based control is made feasible by the UFO-framework [3]. Running on GPU-based computational servers, it is able to process a few Gigabytes of tomographic data per second. The efficient delivery of this data to the computation nodes, though, is a challenge.

To comply with ANKA standards, the control system for the IMAGE beamline [4] is based on TANGO to communicate with the beamline hardware and the controls for the pixel sensors/cameras are exposed using TANGO modules. Since TANGO transports all its data using CORBA with standard TCP/IP communication, its bandwidth is not only limited by the speed of the network adapter, but also by the performance of the TCP/IP stack, especially in terms of latency [5]. Also, the effective bandwidth is further limited by the particular implementation of CORBA (omniORB [6]) used by TANGO.

In this paper, we present our approach to extend the already existing TANGO infrastructure with a secondary high-bandwidth data channel. We use InfiniBand interconnects to avoid latency and bandwidth limitations of standard TCP/IP over Ethernet. A TANGO enabled camera driver was developed that allows to transport camera data through a secondary data channel based on InfiniBand RDMA communication. With this new data channel and the TANGO control system, coupled together, a control interface was created that replicates the remote TANGO camera interface into a local system while using InfiniBand as transfer for the camera data. This effectively combines high throughput data with the flexibility of TANGO while feeling and behaving like a purely local camera system.

DATA TRANSPORT PERFORMANCE

We evaluated the performance of TANGO and omniORB under certain scenarios. For this, a TANGO device server was written that generates pictures of 512x512 pixels (265 KB data) with a static gradient pattern and a region of random noise data in the center. These pictures can then be pulled from the server over the standard TANGO interface. We connect to this device server with a device proxy client that continuously pulls the picture attribute from the device server for a certain number of iterations and creates a mean value of the observed throughput by dividing the transfered

ISBN 978-3-95450-146-5

amount of data by the elapsed time. The amount of iterations were chosen to ensure that at least 1 GB of data gets transfered in total.

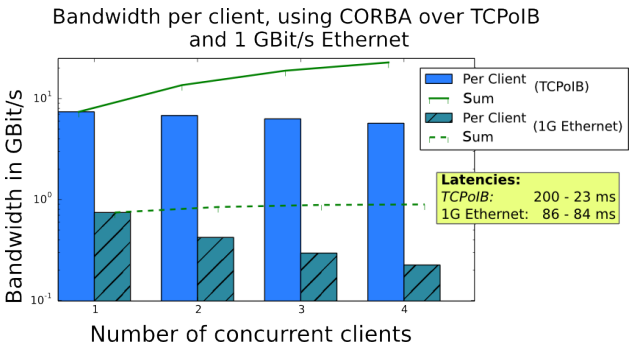

Figure 1: TANGO with omniORB performance using TCPoIB and 1 Gbit/s Ethernet.

We ran the test on a standard 1 Gbit/s Ethernet network with one switch. Our results showed, that omniORB efficiently uses the available bandwidth in case of a single client with up to 75% (750 Mbit/s) of the theoretical bandwidth and with up to 90% in sum (900 Mbits/s) in case of multiple clients. However, since standard Gigabit Ethernet does not provide enough bandwidth in order to transport our camera data with the required data rate of several Gigabytes per second, we need faster nework adapters. Also, with latencies in the order of 85 ms for the standard Ethernet TCP/IP communication of omniORB, high frequency data transfer for high-framerate cameras is impossible.

Since InfiniBand interconnects are already widely used by the IMAGE beamline infrastructure we also investigated the performance of omniORB over InfiniBand interconnects. Modern InfiniBand adapters can easily reach bandwidths of up to 32 Gbit/s (4xQDR) with latencies as low as 2 µs when using *native* InfiniBand communication protocols. However, since omniORB is not able to use native InfiniBand, we ran a performance test with a 4xQDR network using a TCP over InfiniBand stack (TCPoIB) instead. We present the results of our performance tests in Fig. 1. Here we can see that the efficiency of the data transmission is severely limited by the performance of TCPoIB, reaching only 7.4 Gbits/s out of the possible 32 Gbits/s (23%) with latencies ranging from 200 ms to 23 ms, in case of single client access. Per client efficiency is further limited with increasing amounts of concurrent clients. These results suggest, that *native* InfiniBand should be used as a dedicated data channel instead, to alleviate the problem of both bandwidth and latency.

SOFTWARE ARCHITECTURE

In order to separate between data acquisition and processing, and thus allow for easily interchangeable acquisition systems, the UFO infrastructure follows a distributed design. This makes it necessary to remote control the acquisition system over the network. To solve this problem, two pieces of software are already existing:

libuca

libuca is a unified camera control library with a generic interface to control different types of cameras on a local system. The specific implementations for communication and control of each camera type are encapsulated in the form of plugins. Each plugin is derived from a libuca camera base class that provides a well defined interface to perform the most common operations of any camera, such as 'start/stop recording' and 'start/stop readout' (in case of a cameras with an internal buffer). Besides this basic functional interface, each plugin provides a set of *properties* that can be queried and listed from the calling application. These properties are used to represent internal states and settings of each camera, like exposure time, sensor dimensions, region of interest, etc.

UcaDevice

A TANGO Server wrapper around libuca, called *UcaDevice*, was created to integrate supported cameras into the TANGO environment. In order to do so, UcaDevice creates an instance of the selected Uca camera plugin. It enumerates all properties published by the plugins and for each property a TANGO attribute is dynamically created and exposed into the TANGO network. A set of operations mirroring the basic Uca camera interface is published as well.

With these two pieces of software it is possible to remotely control the data acquisition over a TANGO network. However, we can not use the same TANGO communication scheme to also transport the acquired camera data due to the performance problems shown in the *Data transport performance* section. In order to extend the infrastructure by a high-throughput data channel that is able to handle the required data rates and low latencies we have extended UcaDevice with a dedicated data transport channel based on *native* InfiniBand communication. Our solution employs RDMA (Remote Direct Memory Access) and sustains throughput in the order of several Gigabytes per second. RDMA allows to transfer data directly into the destination machines RAM without additional I/O operations, which reduces latency significantly. To achieve this, we have created an InfiniBand communication library, called *KIRO*, and integrated it into to existing software infrastructure.

KIRO Library

The KIRO library is a networking library that provides an InfiniBand server and client class. The server locks a preallocated memory block for RDMA-Read access. Each connecting client, once access is granted, can read the locked memory region at any time without further involvement of the server. The RDMA communication is based on InfiniBand communication primitives and has virtually no transport or protocol overhead.

We added a KIRO server component to the UcaDevice wrapper and added new TANGO attributes to its interface which describe the address information of the KIRO server. The data acquisition mechanism of UcaDevice was changed

ISBN 978-3-95450-146-5

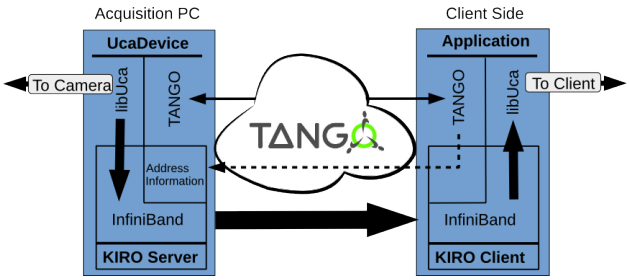

Figure 2: Overview of the software architecture for the InfiniBand data channel.

so it not only provides camera data over the TANGO interface but also via the KIRO server.

In addition, a libuca plugin for any generic TANGO/KIRO-enabled camera was created. The plugin connects to a given TANGO server and, once a connection is established, inquires the remote TANGO attributes to check for an available KIRO InfiniBand server address and port attribute and tries to establish this connection as well. All available attributes are queried from the TANGO server and are translated into internal libuca properties. This efficiently clones the discovered TANGO interface into the local libuca environment and provides a transparent remote interface for the libuca application. The remote camera then feels and behaves as if the camera was connected locally. If the KIRO connection setup was also successful, the plugin proceeds to use the established InfiniBand connection to pull data from the camera, while retaining the TANGO connection to remotely control the connected camera. Otherwise, the standard TANGO attribute for data transmission is used.

The architecture of the overall system is shown in Fig. 2. With this software architecture, we were able to establish a secondary data channel based on native InfiniBand communication between the camera and the processing PC. The UcaDevice server is used to generate a TANGO interface for the locally connected camera which is controlled via an appropriate libuca plugin. It also provides an instance of the KIRO Server to transport camera data.

On the receiving side, our client application loads libuca with the aforementioned KIRO-plugin. It gets directed to our UcaDevice TANGO server and clones the TANGO interface back into a local libuca interface while also establishing a KIRO InfiniBand connection for data transfer.

PERFORMANCE COMPARISON AND CONCLUSION

With this architecture we were able to achieve a bandwidth of 30 Gigabit/s out of the 32 Gigabit/s that are theoretically possible with our 4xQDR InfiniBand test network and average latencies of 6 µs (see Fig. 3).

This shows that, in case of single client access, InfiniBand efficiency surpasses that of TCP/IP easily while also retaining a significantly smaller latency. It does, though, also suffer from some limitations in case of concurrent client access. However, since the data channel using the InfiniBand connection is designed to be a dedicated data channel only for data transfer and is independent of the control communication, concurrent access becomes less of an issue for both communication channels.

This concludes that we are able to use the full efficiency of modern InfiniBand interconnects in a fully TANGO controlled environment while still retaining standard control schemes over TANGO with high-throughput and very low latencies.

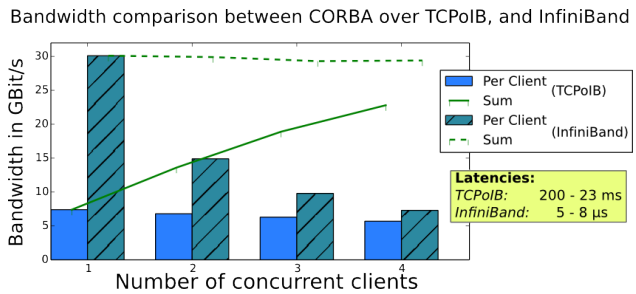

Figure 3: Comparing performance of omniORB over TCPoIB with InfiniBand.

REFERENCES

[1] A. Götz et al., "TANGO is a CORBA based Control System", In Proc. ICALEPCS2003, Gyeongju, Korea, MP705, (2003).

[2] Tomy dos Santos Rolo et al., "In vivo X-ray cine-tomography for tracking morphological dynamics", In Proceedings of the National Academy of Sciences of the United States of America (2014), pp. 3921–3926, doi://10.1073/pnas.1308650111, http://www.pnas.org/content/111/11/3921.full.pdf + html, http://www.pnas.org/content/111/11/3921

[3] M. Vogelgesang et al., "UFO: A Scalable GPU-based Image Processing Framework for On-line Monitoring", In Proceedings of the 14th IEEE International Conference on High Performance Computing and Communications (HPCC-2012) & The 9th IEEE International Conference on Embedded Software and Systems (ICESS-2012) pp. 824–829, doi:10.1109/HPCC.2012.116

[4] M. Vogelgesang et al., "When Hardware and Software Work in Concert", In Proc. ICALEPCS2013, San Francisco, CA, USA, TUPPC044, (2013).

[5] Mellanox Technologies Inc., "InfiniBand and TCP in the Data Center", Document Number 2008WP, (2008).

[6] D. Grisby et al., omniORB: Free CORBA ORB., http://omniorb.sourceforge.net/

ISBN 978-3-95450-146-5