Development of a CORBA Toolkit and its Evaluation

Norihiko Kamikubota, Matthias Clausen*, and Kazuro Furukawa
High Energy Accelerator Research Organization (KEK)
1-1 Oho, Tsukuba, Ibaraki 305, Japan
Deutsches Elektronen-Synchrotron (DESY) *
Notkestrasse 85, 22603 Hamburg, Germany

Abstract

Trials to develop a CORBA toolkit have been made at DESY and KEK. This toolkit provides basic APIs for general-purpose data exchange.

There are two aims of the trials. The first one is to make it easier to convert the existing CORBA applications of older versions (Digital's @aGlance) to those with the new CORBA environment (Object Broker). Since the toolkit includes a complete emulation of the @aGlance functionalities, re-use of the existing sources becomes possible. The second aim is to evaluate the availability of CORBA over different control systems. In order to inspect the availability, the toolkit was introduced into three different control systems at DESY and KEK. As a result, communication availability has been demonstrated over three different platforms.

In addition, when an application is developed with the APIs of the present toolkit, it can communicate with any of the three different control systems. Thus, the present work encourages the sharing of application software between DESY and KEK.

1 Introduction

At KEK, an injector linac of 2.5-GeV electrons/positrons[1] provides beams to the KEKB rings. Since each accelerator (the linac and the rings) was constructed independently, there are no established communication protocols between the two control systems.[2] At DESY, the cryogenic control system (D/3 system [4]) and the vacuum system (DOOCS system [5]) were developed for the HERA ring, but are independent. In both cases, new ideas are needed for cooperative operations between two accelerators.

In recent years, a distributed object-oriented design has become a promising direction for future accelerator control systems. New techniques related to object-oriented ideas, for example cdev[6], CORBA, Java, HORB[7] and so on, become available. Feasibility studies in 1995 suggested that the introduction of cdev and CORBA would be a possible solution to link two accelerators at KEK [8]. In 1994, DESY introduced a toolkit, based on an old version of CORBA, and showed the usefulness of this toolkit (see Section 2.1).

This article describes our experimental trials to introduce recent versions of CORBA to our (both DESY and KEK) accelerator control systems. At first, we developed a toolkit with the recent CORBA product (Section 2). The toolkit was then introduced to the DESY/KEK accelerators. A discussion is given in Section 3. Finally, a conclusion is given in Section 4.

[1] Now upgrading the energy from 2.5 GeV to 8/3.5 GeV for electrons/positrons as a part of the KEK B-physics project (KEKB) [3]. Upgrade will be completed in 1998.

[2] Details of these control systems are given elsewhere [1, 2].

2 Development of the CORBA toolkit

2.1 An early study at DESY-KRYK

The @aGlance is an application developer's toolkit by Digital which is based on the old version of CORBA (CORBA 1.1). Typical application interface functions (APIs) for the client-server model are included. For example, a) 'GetList' to get values of the specified tag-names, b) 'GetTags' to get object tag-names, and c) 'GetHistory' to get history data of a specified time-window.

The cryogenic control group of DESY (DESY-KRYK) started to develop an archiver viewer system in 1994. As shown in Figure 1, the archiver viewer can communicate with any of the archiver servers for the three different control systems. It is worth noting that the @aGlance toolkit made it possible to share high-level application(s) among different control systems. However, this toolkit is available only on VAX machines. Moreover, Digital decided to stop its support for this toolkit.

2.2 A new toolkit with object broker

A new toolkit, based on the latest CORBA product (Object Broker[3] provided by Digital based on CORBA 1.2), has been developed at DESY. The new toolkit includes complete emulation of the @aGlance functions; thus, re-use of the existing sources is possible. In addition, Object Broker has a C-language flavor rather that C++, while most others are dedicated to C++. It is preferable for us, since our control systems are mostly written in C language.

The current version 1.0a (Jan.1997) still has a serious problem in that only one service is allowed at the same time. This problem will be removed in future versions.

2.3 Performance

The present toolkit is expected to work on any platform for which Object Broker is available. So far, the availability was checked for the following platforms:

[3] The latest version of Object Broker is 2.7, while 2.5A and 2.6 were used with the developments and tests in this article.

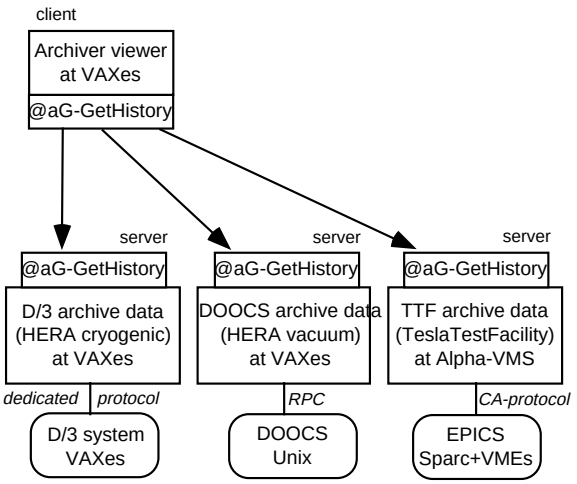

Figure. 1. Relationship between a client (archiver viewer) and servers of archive data for three different control systems.

(a) Open-VMS V6.2 and DEC-C with a MultiNet/TCPIP environment on a DEC Alpha250 4/266;
(b) Sun OS 4.1.3 with an acc compiler on a Sparc Station 10;
(c) Digital Unix v3.2D on a DEC Alpha Server 2100A.

Throughput measurements with functions of the toolkit were carried out on the above three machines. The results are summarized in Table I. The case [1] in Table I has both the client and server processes running on the same machine (the machine (a)), while in case [2] the two processes reside on different machines (the machines (a) and (b)).

It takes 2-12 ms (50-680 ms) to transfer 1 (1000) double value(s) with our environment. The measured data-transfer rate of the toolkit is lower than that of the original Object Broker by the factor 10–40. When the server and the client run in the same machine (case [1]), the limit comes from the CPU power, while the network capacity (10 Mbps in the current test) seems to be the problem when the client runs on a different machine (case [2]).

3 Discussion

3.1 Comparison of CORBA with socket functions

(a) data transfer
A simple test on a workstation (the machine (c) in Section 2.3) shows that the round-trip time needed to exchange 1kB data between two processes is 9.6 ms. The same test with socket functions results in 0.17 ms. In turn, the available maximum size is 1.2-1.4kB with socket functions, while the limit is not specified with CORBA.[4]
(b) programming convenience
For cases where many clients are communicating with one server, programmers need not care about multi-thread handling with CORBA. Another benefit of CORBA is the redundancy of services: many servers providing the same service can reside on distributed nodes. When the server stops, the client disconnects it and searches for another server automatically. In addition, the use of IDL (interface definition language) makes the handling of interface layers easier.

Table I
Throughput of the CORBA toolkit for typical cases.

Function	Arguments (datasize)	case [1]	case [2]
1a) GetList()	1 tag (1 double)	40 ms	180 ms
1b) GetList()	1000 tags (1000 double)	60 ms	250 ms
2a) GetHist()	1double x 1tag (1 double)	130 ms	580 ms
2b) GetHist()	1000double x 1tag (1000 double)	170 ms	680 ms
2c) GetHist()	1000double x 4tags (4000 double)	200 ms	800 ms
3a) GetTags()	5 tags (10strings x 3-5byte)	9 ms	36 ms

3.2 Progress at DESY-KRYK

The new archiver viewer and the server for the TTF archive data were created by using the existing sources and the new toolkit (see the left half of Figure 2). The result is quite successful. The archiver viewer runs in the same way as before without the @aGlance toolkit.

Another important progress is that non-VMS machines can be introduced into the KRYK's control scheme. We are trying to install the same archiver viewer on a Sparc station (the machine (b) in Section 2.3).

3.3 Progress at KEK-Linac

An archiver server of the KEK injector linac (KEK-Linac) was also prepared with the toolkit (see the right half of Figure 2). As a result, an archiver viewer running at DESY can connect with the archiver server of KEK-Linac, as shown in Figure 3.

The present experience suggests the possibility to share high-level applications between DESY and KEK. We are trying to install an archiver viewer at KEK.[5] The present research also shows that CORBA is one of the possible selections for the communication protocol between the injector linac and the KEKB rings.

[4] Transfer of 16kB data was confirmed with Object Broker.

[5] But not completed yet. The problem is the difference of graphic packages.

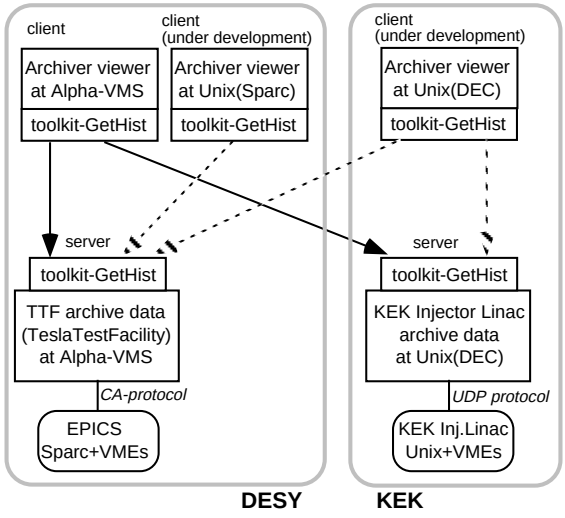

Figure. 2. Relationships with the new CORBA toolkit.

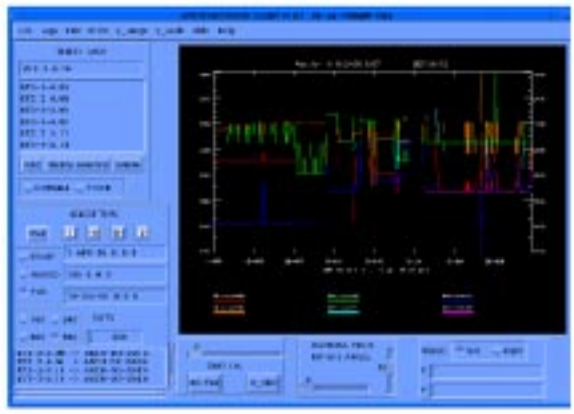

Figure. 3. Archiver viewer running at DESY with histories of six steering magnets of the KEK-Linac.

3.4 Future directions

Since the present toolkit accepts only one service at a time, more development to enable multiple services is needed. In addition, it is important to inspect the inter-operability of the toolkit with other CORBA products.[6]

The present CORBA toolkit can be used as a communication protocol of distributed cdev implementations. The DESY-KRYK group is preparing the cdev environment in order to enable communications between the existing control systems. The toolkit would be used for connections with the D/3 system and the DOOCS system.

[6] For example, Orbix by IONA Technologies Inc.

4 Conclusion

A CORBA toolkit has been developed for accelerators at both DESY and KEK. The data-transfer times with our typical machines are 2-12 ms (50-680 ms) for 1 (1000) double value(s), which are 2 orders slower than socket functions. However, CORBA has benefits over socket functions from the viewpoints of programmers.

The toolkit was installed on three different platforms at DESY and KEK; it was found that the inter-operability was successful. At DESY-KRYK, the toolkit enables the introduction of non-VMS computers into their control systems. At KEK-Linac, it encourages the sharing of application software with DESY.

Acknowledgements

A major part of this work was carried out in 1996 at DESY with the help of the staff of the cryogenic control group. We thank the staff for their kind encouragement as well as for their technical support. One of the authors (n.k.) thanks the Japan Society for the Promotion of Science (JSPS), since the financial support for his stay at DESY in 1996 was provided by a JSPS Fellowship for Research at Centers of Excellence Abroad.

References

[1] N. Kamikubota, K. Furukawa, K. Nakahara and I. Abe, Nucl. Instr. Meth. A352(1994)131

[2] T.Katoh, S-I.Kurokawa, T.T.Nakamura, T.Mimashi and N.Yamamoto, Proc. of the 1995 Particle Accelerator Conference and International Conference on High-Energy Accelerators, Dallas, Texas, May 1995, (IEEE 1996), p.2205

[3] A. Enomoto, Proc. of the XVIII International Linear Accelerator Conference (LINAC96), Geneva, Switzerland, Aug.1996, CERN96-07, p.633

[4] M. Clausen, K.-H. Mess, C. Gerke and S. Rettig, Advances in Cryogenic Engineering, Vol.33 (Plenum Press, New York, 1988) p.1113

[5] K. Rehlich, A distributed Object-oriented control system on PCs & workstations, in the workshop PCaPAC, Hamburg, Germany, Oct.1996

[6] J. Chen, G. Heyes, W. Akers, D. Wu and W.A. Watson III, Proc. 1995 Int'l Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'95), Chicago, Oct.-Nov.1995, FERMILAB-CONF96/069, p.97

[7] S. Hirano, Proc. on France-Japan Workshop on Object-Based Parallel and Distributed Computation (OBPDC'97), Toulouse, France, 1997, in press; or Descriptions at http://ring.etl.go.jp/openlab/horb/

[8] K. Furukawa, N. Kamikubota, K. Nakahara, I. Abe and A. Shirakawa, Proc. 1995 Int'l Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'95), Chicago, Oct.-Nov.1995, FERMILAB-CONF96/069, p.863