# Fuego: Middleware for Mobile Data Communication and Synchronization

Sasu Tarkoma Jaakko Kangasharju Tancred Lindholm Ramya Sri Kalyanaraman
Kimmo Raatikainen

Helsinki Institute for Information Technology
PO Box 9800, 02015 TKK, Finland
{sasu.tarkoma,jkangash,tancred.lindholm,ramya,kimmo.raatikainen}@hiit.fi

## 1 Introduction

In designing middleware for wireless mobile devices, the environment is notably different from its wired counterpart. Compared to fixed networks and hot-spots, wide-area wireless networks tend to offer lower bandwidth, and higher latency and pricing. The computing capabilities of devices are weaker, and every CPU cycle counts, as energy consumption is a key issue for making a device successful in the marketplace.

Our research project Fuego Core has built a prototype middleware platform for this mobile environment[1]. We have chosen to direct our work based on existing standards as much as possible, so that our middleware will be compatible with existing deployed infrastructure on the fixed network side. In particular, we build everything on top of Internet Protocol (IP).

A distinguishing characteristic of our project is the strong interaction between theory and implementation. As a group with expertise in software engineering, we have put great effort into the implementation, striving for an approach not only theoretically sound but also well thought out from a practical point of view. The contributed set of middleware primitives in our approach are thus more than a set of pieces that work together on the paper: the implemented parts fit together in a manner characteristic of well-engineered software.

## 2 Architecture

The architecture of the current Fuego middleware platform is shown in Figure 1. This figure is divided into APIs on the left, showing what functionality we have considered necessary, and implementations on the right, showing what component of the middleware implements which API. This selection of components is similar to that of other mobile middleware systems [2, 9].

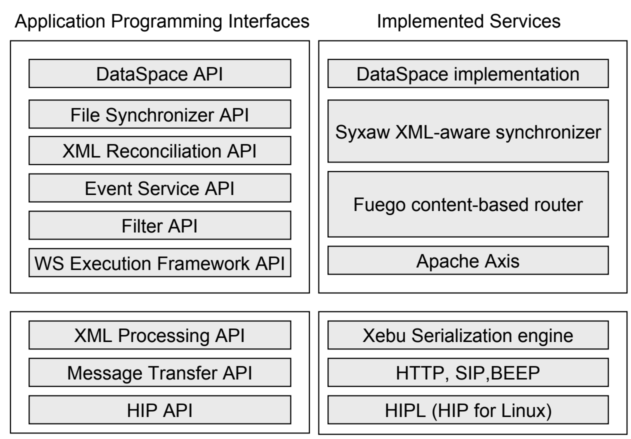


**Figure 1. Fuego middleware architecture**

At the bottom is the Host Identity Protocol (HIP) [5], currently being standardized by the IETF. HIP provides a cryptographic namespace for identities separate from IP addresses, making it possible to identify hosts and connect to them even if their IP addresses change.

As the principal data format of our system we use XML, so selecting SOAP as the messaging protocol was a natural choice. While we use standard application-layer protocols, we also provide additional support for mobility on top of these. XML processing is handled through an API that is better suited for messaging than common APIs [3]. As an option, we also provide a binary serialization format for XML [4], which is much more compact than regular XML, but retains data model compatibility with it.

We see publish/subscribe communication [1] as a crucial enabler of future network applications, due to its asynchronous many-to-many communication paradigm. The Fuego event service is intended as the primary communication ser-

[1] Implementation available at `http://hoslab.cs.helsinki.fi/homepages/fuego-core/`

vice of the middleware, building its broker network on top of the simpler messaging service. The main contributions of the event service are a scalable routing system [11] and explicit mobility support [10].

Session Initiation Protocol (SIP) has been selected as an essential component of the future third generation mobile phone networks. In line with our commitment to standards, we include an interoperability gateway [12] between our event service and the SIP Events package, which also supports gatewaying of event filters.

Even now, users often have several devices that they use for network access, and in the future the number of personal devices will only increase. Thus it is vital for users to be able to synchronize their data. Our system provides a file synchronizer [7] based on optimistic reconciliation [8] and explicit support for merging changes to XML files [6].

A notable characteristic of our platform is that mobility of client devices is considered at every layer, with solutions tailored to the specifics of each layer. In our view this is needed, as the issues related to mobility are different depending on the layer, so they cannot be fully solved at any individual layer.

## 3 Future Work

While we consider the current Fuego middleware already a useful system for building applications, there are still several future work items that we have identified during the course of our work.

An important issue will be security on every layer. There needs to be an appropriate security model with requirements on every layer of the middleware, and support for this. Examples include high-granularity encryption for messages and access control for the event service.

Handling of errors is naturally necessary in all components. However, the mobile environment is such that errors in, e.g., communication will be a natural occurrence instead of a rare event, so the middleware needs to treat such error conditions in a similar manner to normal situations.

We view the future computing environment to follow in large parts an ad-hoc peer-to-peer model where intermittently encountered other devices need mutual communication. This kind of environment sets new challenges for the lower-layer messaging components in terms of addressing and for data synchronization by the necessity to support ad-hoc sharing of information.

These two points also imply that the set of devices and services available at each point in time is highly variable. This available set needs to be managed by the middleware so that applications can best take advantage of various services that are offered by the environment while the user moves.

## References


[1] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec. The many faces of publish/subscribe. *ACM Computing Surveys*, 35(2):114–131, 2003.

[2] R. Grimm, J. Davis, E. Lemar, A. MacBeth, S. Swanson, T. Anderson, B. Bershad, G. Borriello, S. Gribble, and D. Wetherall. System support for pervasive applications. *ACM Transactions on Computer Systems*, 22(4):421–486, Nov. 2004.

[3] J. Kangasharju and T. Lindholm. A sequence-based type-aware interface for XML processing. In *Ninth IASTED International Conference on Internet and Multimedia Systems and Applications*, pages 83–88. ACTA Press, Feb. 2005.

[4] J. Kangasharju, S. Tarkoma, and T. Lindholm. Xebu: A binary format with schema-based optimizations for XML data. In *6th International Conference on Web Information Systems Engineering*, volume 3806 of *Lecture Notes in Computer Science*, pages 528–535, New York, USA, Nov. 2005. Springer-Verlag. Short paper.

[5] M. Komu, S. Tarkoma, J. Kangasharju, and A. Gurtov. Applying a cryptographic namespace to applications. In *First International ACM Workshop on Dynamic Interconnection of Networks*, Sept. 2005.

[6] T. Lindholm. A three-way merge for XML documents. In *ACM Symposium on Document Engineering*, pages 1–10, Oct. 2004.

[7] T. Lindholm, J. Kangasharju, and S. Tarkoma. A hybrid approach to optimistic file system directory tree synchronization. In *Fourth International ACM Workshop on Data Engineering for Wireless and Mobile Access*, June 2005.

[8] Y. Saito and M. Shapiro. Optimistic replication. *ACM Computing Surveys*, 37(1):42–81, 2005.

[9] J. P. Sousa and D. Garlan. Aura: an architectural framework for user mobility in ubiquitous computing environments. In *Software Architecture: System Design, Development, and Maintenance (Proceedings of the 3rd Working IEEE/IFIP Conference on Software Architecture)*, pages 29–43. Kluwer Academic Publishers, Aug. 2002.

[10] S. Tarkoma and J. Kangasharju. Mobility and completeness in publish/subscribe topologies. In *IASTED International Conference on Networks and Communication Systems*. ACTA Press, Apr. 2005.

[11] S. Tarkoma and J. Kangasharju. Optimizing content-based routers: Posets and forests. *Distributed Computing*, 2006. To appear.

[12] S. Tarkoma and T. B. Ramya. A gateway for SIP event interworking. In *Third Workshop on Applications of Wireless Communications*, pages 49–59, Aug. 2005.