Journal of Universal Computer Science, vol. 8, no. 6 (2002), 644-651
submitted: 4/2/02, accepted: 3/5/02, appeared: 28/6/02 © J.UCS

Shark – a System for Management, Synchronization and Exchange of Knowledge in Mobile User Groups

Thomas Schwotzer
Intelligent Networks and Management of Distributed Systems[1]
TU Berlin, Germany
thsc@ivs.tu-berlin.de

Kurt Geihs
Intelligent Networks and Management of Distributed Systems
TU Berlin, Germany
geihs@ivs.tu-berlin.de

Abstract: New wireless protocols like W-LAN and Bluetooth allow establishing spontaneous networks and peer-to-peer exchange of information. At the same time standards like Semantic Web and Topic Maps gain acceptance that add semantics to information. This paper introduces Shark. Shark is an acronym and stands for "Mobile Shared Knowledge". Shark organizes knowledge with help of Topic Maps, synchronizes knowledge inside closed user groups but also enables a peer-to-peer exchange of knowledge by means of Bluetooth. This paper gives an overview of the system and its communication protocols.

Keywords: knowledge management, information systems, knowledge work processes
Categories: C.2.4., H.1.2., H.4.3.

1 Introduction

New wireless protocols and mobile devices make the Internet mobile. Protocols such as Bluetooth and W-LAN facilitate a peer to peer exchange of arbitrary information in ad hoc networks where communication links may be created temporarily in a spontaneous manner. On the other hand, standards like Topic Maps [TM, XTM] and Semantic Web [SW] gain acceptance that add semantics to Internet documents. Adding such semantic information makes it easier to organize and share knowledge in a distributed environment.

This paper introduces Shark. Shark is an acronym and stands for "Mobile Shared Knowledge". It is a distributed system which supports organization, synchronization and exchange of knowledge for mobile users within a given user group or between users that are members of different user groups.

This paper gives an overview of the design of Shark and its communication protocols. Knowledge exchange requires extraction of parts of a knowledge base and incorporation of knowledge into an existing knowledge base. In order to handle these

---

[1] The research group iVS is sponsored by Deutsche Telekom AG.

knowledge management tasks, Shark uses on a new concept, called *knowledge ports*. We present a prototype implementation of the system using XML Topic Maps (XTM 1.0) to organize, SyncML [SyncML] to synchronize and KQML [KQML] to exchange knowledge.

2 Shark Overview

The 3GPP organisation [3GPP] has defined requirements for the 3rd generation of mobile networks. In these standards, e.g. MExE [MEXE], mobile devices are called *mobile stations*. We adopt their terminology. Therefore the mobile component of Shark is called *Shark Mobile Station*. The other components are *Shark Central Station* and *Shark Local Station*. Remember that we assume a mobile, spontaneous networking environment with a mixture of mobile and stationary computing devices.

Shark Mobile Station (MS) is software which runs on mobile devices that offer a J2ME environment. The Shark MS contains a part of the Shark-wide knowledge base. It can synchronize its knowledge with a Shark Central Station with the help of SyncML [SyncML]. A Shark MS exchanges knowledge using KQML [KQML] in an ad hoc network.

Shark Central Station (CS) is server software. It contains the complete knowledge base. Currently the knowledge base is based on XML Topic Maps [XTM]. The Shark CS offers tools to organize knowledge and to manage the synchronization. New knowledge enters a Shark CS during synchronization. This kind of knowledge is called *unconfirmed* knowledge. After an evaluation process this knowledge will be dropped or published Shark-wide, i.e. becomes *confirmed* knowledge.

Shark Local Station (LS) is a special kind of Shark MS. In contrast to Shark MS it runs at a fixed location. Shark LS is designed to offer location-based knowledge. Otherwise, it synchronizes and exchanges knowledge like a Shark MS.

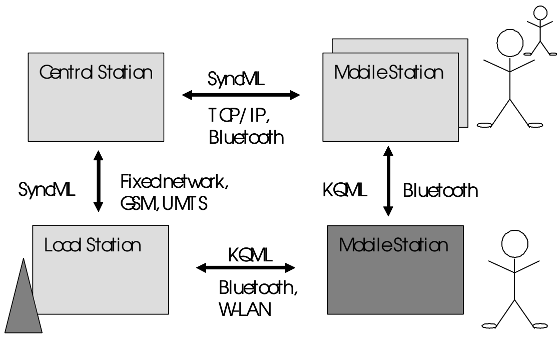

Figure 1: Shark Components and Interactions

Figure 1 illustrates the Shark components and their interactions. In an enterprise environment with mobile staff and stationary offices, all Shark components would be used. Shark CS contains the enterprise wide knowledge base. Shark MS runs on mobile devices of the mobile employees. It provides parts of the knowledge base, e.g. product descriptions, price list, white papers etc. These collaterals are organized by topics. Now, mobile employees can exchange (confirmed) knowledge with interested people. A Shark LS is situated, for instance, in a shop or a subsidiary of the enterprise. Like Shark MS it can provide knowledge to passers-by or visitors. It offers location based knowledge because a Shark LS runs at a dedicated place.

3 Shark Central Station

Any Shark Central Station comprises a XTM Engine. In Shark 1.0 we use Ontopia Knowledge Suite [OKS]. An XTM Engine comes along with management tools for organizing knowledge. Organizing knowledge covers defining topics, associations, scopes and types and assigning occurrences to topics.

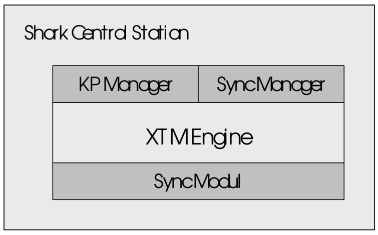

Figure 2: Shark Central Station

Shark facilitates users to copy parts of a topic map to Shark Mobile Stations. These copies will be kept in sync. The Synchronisation Manager allows defining which topics shall be kept in sync whereas the Synchronisation Module executes the synchronisation process, see figure 2. This process will be discussed in a separate chapter.

In Shark, synchronization is the way to spread knowledge in mobile user groups. New knowledge is delivered from a Shark Mobile Station to the Central Station and vice versa. All members of a mobile user group will retrieve new knowledge after a while.

Storing knowledge in a knowledge base does not necessarily mean that one is also interested in delivering or retrieving knowledge. *Knowledge Ports*[Sc02] are used to define interest in knowledge exchange. There are two kinds of Knowledge Ports: Incoming Knowledge Ports (IKP) state an interest in retrieving knowledge. The opposite is an Outgoing Knowledge Port (OKP) which states an interest in delivering knowledge. Knowledge Ports can be assigned to topics and enhance their semantic. Whereas a TCP/IP port is a logical entry and exit of data exchange in the Internet a Knowledge Port is a logical port of knowledge exchange. Detailed discussions on Knowledge Ports and their usage can be found in [Sc02, ScPr02].

Shark 1.0 defines Knowledge Ports as topic types. A topic gets a knowledge port simply by declaring this topic to be of type “Knowledge Port”, e.g.:

```
<topic id="interesting_topic">
  <!—topic has an incoming knowledge port -->
    <instanceOf>
      <topicRef xlink:href="http://www.sharksystem.de/core/1.0/psi.xtm#ikp"/>
    </instanceOf>
  …
</topic>
```

Complete definitions of Knowledge Ports can be found in http://www.sharksystem.de/core/ 1.0/psi.xtm.

Topics can have multiple types. Therefore one topic can have an IKP and an OKP or other types that are not relevant in the context of Shark. Using types to declare Knowledge Ports instead of introducing a XTM extension has an additional advantage: A XML topic map enriched with Knowledge Ports is still a valid XTM topic map. Tools without understanding of Knowledge Ports handle them as ordinary types. Such topic maps can be managed with arbitrary XTM engines without loosing Knowledge Ports.

The Knowledge Port Manager is used to manage Knowledge Ports.

4 Shark Mobile Station

A Shark Mobile Station runs on every system which offers a J2ME environment, like Palm m515 or Nokia 9210. A Mobile Station is assumed to move often, to have less graphical capabilities, to be regularly offline and to have less computational power. For that reason a Mobile Station does not comprise a fully-fledged XTM engine. It only contains XTM documents which are copies of parts of the central knowledge base. Shark 1.0 comprises a reduced XTM editor which allows to browse a topic map and to add topics, associations and occurrences. There is also an adapted version of the Knowledge Port Manager, see figure 3. It has the same tasks as in the central station.

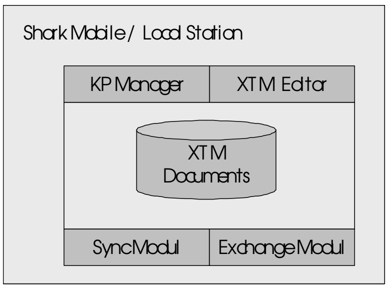

Figure 3: Shark Mobile Station

A user of a Shark Mobile Station cannot manage the synchronization process. The Synchronization Module is the counterpart of the module on the Central Station. Both perform the synchronisation process.

The Exchange Module does not exist at the Central Station. Two Exchange Modules can exchange knowledge, based on their local XTM documents and their Knowledge Ports. Knowledge Exchange will be discussed later.

5 Shark Local Station

A Shark Local Station comprises the same components as the Mobile Station. But in contrast, a Local Station runs on a PC with a J2SE environment.

6 Knowledge Exchange with Shark

The most powerful feature of Shark is knowledge exchange. Note that knowledge *exchange* takes place between users of *different groups* whereas knowledge *synchronization* keeps *one group* in sync. Technically, knowledge exchange takes place between two knowledge exchange modules of Shark MS and/or Shark LS. It comprises the following steps:

1. find mutual interests
2. test for mutual interest of knowledge exchange
3. test for right of knowledge exchange
4. knowledge exchange

The XTM merging rules (see [XTM]) define under which circumstances two topics are considered to be identical. These definitions are sufficient to identify mutual interest (step 1).

It should be noted that this approach is not sufficient in all scenarios. Imagine two mobile users like to exchange knowledge. One is interested in retrieving everything about “knowledge management” whereas another one has something to deliver about “topic maps”. The topic “topic map” may be seen as a subtopic of “knowledge management”. Based on the merging rules we won’t find any mutual interest because both topics are obviously not the same. On the other hand side, the interested person would appreciate to learn something about topic maps because he is interested in knowledge management in general. Additional relations like child/parent-relations must be used beside tests on identical topics. A simple algorithm for that scenario is proposed in [ScPr02]. This algorithm is not yet implemented in Shark 1.0.

An interest in knowledge exchange (step 2) exists if corresponding Knowledge Ports are assigned to matching topics. Corresponding Knowledge Ports are an IKP/OKP pair.

Knowledge Ports are also used to declare access rights (step 3).

Knowledge exchange is performed by exchanging XTM documents (step 4).

Newly arrived knowledge gets automatically the type “unconfirmed knowledge”. A human user is required to withdraw that type. See the next chapter for a more detailed discussion about the handling of unconfirmed knowledge.

The knowledge exchange protocol is base on KQML [KQML] and runs over Bluetooth. Therefore knowledge exchange can take place in a spontaneous manner, e.g. when a Palm and a mobile phone meet and establish a Bluetooth connection. No central server instance is required. Knowledge exchange is peer-to-peer exchange.

7 Knowledge Synchronisation with Shark

A Shark Mobile Station contains a set of copies of topics, associations and occurrences. In Shark 1.0 only the knowledge located at the Central Station is assumed to be the original one. Furthermore, in Shark 1.0 a user cannot delete or rename topics, associations at the Mobile Station. A Mobile Station can only be used for receiving new knowledge either during knowledge synchronization and exchange or by adding new knowledge manually.

This looks very restrictively at the first glimpse. In fact, later versions of Shark will support less limiting policies. On the other hand, these restrictions fit well to an enterprise with a centralized knowledge management policy. With such an approach only dedicated knowledge officers are allowed to manage a knowledge base (at the Central Station), whereas others can only make proposals for enhancing it. Beside this, a Shark Mobile Station runs at a smartphone or another device with limited resources. These devices are well suited to gather knowledge but not to offer a knowledge base editor. From this perspective, these assumptions are still restrictive but useful in a working environment in the real world.

The synchronization process comprises two steps. During the first step new knowledge at Mobile Stations side is sent the Central Station. The sent knowledge is marked and won't be delivered during next synchronization. At Central Stations side a topic map merge takes place. After this, all changes at Central Stations side are bundled in one SyncML synchronization request. If knowledge was deleted or renamed theses changes will also be announced to mobile devices.

Depending on the policy unconfirmed knowledge can be distributed in the group during synchronization or not. One strategy is to synchronize unconfirmed knowledge from Mobile and Local Stations to their Central Station but not vice versa. A knowledge officer assesses unconfirmed knowledge and decides whether to reject or to keep it in the knowledge base. With such strategy mobile users and local subsidiaries gather knowledge whereas a headquarter is in charge to control its quality. This is the standard strategy of Shark 1.0.

Note, in any case, unconfirmed knowledge will be kept at the mobile device until it is explicitly deleted. Without that rule a human user might be confused. Imagine a policy of an enterprise that does not allow spreading unconfirmed knowledge. In this case newly gathered knowledge – which is unconfirmed by definition – would be visible at Mobile Stations until next synchronization. The Central Station wouldn't deliver this knowledge until it became part of the knowledge base. Consequently, a Mobile Station would have to discard this knowledge. From user's perspective, knowledge would be available after receiving it during knowledge exchange. After the next synchronization process it would get lost. After a while it could come back as confirmed knowledge. With the rule above, any retrieved knowledge is kept in a local XTM document until a knowledge officer has made a decision.

Therefore the second and final synchronizations step works like this: After receiving the synchronization request the Mobile Stations performs all synchronization commands. Unconfirmed topics are only discarded if their have been explicitly deleted at server’s side. Unconfirmed associations and occurrences are also deleted if their topics have been deleted.

8 Usage Concept: Organizing knowledge flow

Shark is designed as a mobile extension of a knowledge base. Shark 1.0 is a mobile extension of a XML topic map engine. Shark supports mobile user groups with similar interests, e.g. sales representatives of a company, music fans, journalists, scientists and so on. Interests are defined with help of topics. All members of one user group are kept in sync due to synchronization. An exchange of knowledge between different user groups can be performed with help of knowledge exchange.

The definition of Knowledge Ports takes place at the Central Station. They are spread to mobile users during synchronization. Now, mobile devices of the user group have got parts of the central knowledge base including Knowledge Ports. Mobile users benefit from locally available knowledge whereas the group benefits from the fact that any mobile device sends and gathers knowledge on behalf of the whole group.

It should be emphasized that no user interaction is required neither to retrieve nor to send knowledge. Due to this there is even no need that a user understands anything about the exchanged knowledge. A knowledge officer at Central Stations side assesses the incoming knowledge and adjusts Knowledge Ports.

Knowledge is transferred inside a group during synchronization. It is moves between different groups with help of knowledge exchange. We call the overall process *knowledge flow*.

It shall be pointed out: Knowledge bases are used to organize knowledge whereas Shark allows to organize the flow of knowledge.

9 Summary and Outlook

Currently knowledge flow ends in the central knowledge base. A knowledge officer is required to assess incoming knowledge. An increasing amount of incoming knowledge requires smarter solutions. We plan to integrate a workflow management system which is a logical extension of an Incoming Knowledge Port. We use recommendations of the Workflow Management Coalition [WfMC] for that task.

In general, experience with knowledge management projects in enterprises has shown that users need incentives in order to actively offer and transfer knowledge to others. A micro payment system in Shark could be the basis for such an incentive system. In our future work we will develop the technical foundation for such a system. Micro payment can also be used to charge the exchange of knowledge.

There is only a poor security model in Shark. One can only distinguish between users of a group and an external user. This situation is unsatisfying and will be tackled in the near future.

Knowledge management is still a challenge. Emerging standards like Topic Maps and Semantic Web will enrich Internet content with semantics. At the same time

Internet becomes mobile because users of the Internet are already mobile. That means every system which is designed to support user groups faces the task to support mobile users with mobile devices. Mobile devices differ from PCs not only from their limited computational power. Mobile devices cannot be assumed to be usually online. Furthermore, mobile devices have the capability to establish spontaneous connections based e.g. on Bluetooth.

Shark is one solution to support mobile user groups with knowledge. Shark also deploys the capability of spontaneous networks to establish a spontaneous peer-to-peer knowledge exchange. Shark allows to manage the knowledge flow in and between mobile user groups.

References

[3GPP] The 3rd Generation Partnership Project: http://www.3gpp.org

[KQML] Finin, T., Weber J., Wiederhold, G., Genesereth, M. Fritzson,R., McKay, D., McGuire,J., Pelavin, R., Shapiro, S., Beck, C.: *Specification of the KQML Agent-Communication Language - plus example agent policies and Architectures*; June 1993

[MEXE] 3GPP: *Mobile Station Application Execution Environment Stage 1 (document number 02.57).*

[OKS] Ontopia Knowledge Suite: http://www.ontopia.net/solutions/products.html

[Sc02] Schwotzer, T.: *Context Driven Spontaneous Knowledge Exchange*; Proceedings of the 1st German Workshop on Experience Management 2002 (GWEM02), Berlin, March 2002

[ScPr02] Schwotzer, T., Preuss T.: *Knowledge Exchange in Spontaneous Networks - Towards Ubiquitous Knowledge*; Proceedings of E-World@Syria (ET2EB), Damaskus, April 2002

[SW] Berners-Lee, T., Hendler, J., Lassila, O.: *The Semantic Web*; Scientific America, May 2001

[SyncML] SyncML Initiative; http://www.syncml.org/

[TAO] Pepper, S.: *The TAO of Topic Maps*; November 2000

[TM] ISO/IEC 13250: *Topic Maps*; December 1999

[WfMC] The Workflow Management Coalition: http://www.wfmc.org/

[WLAN] IEEE 802.11: *Wireless Local Area Networks*

[XTM] Pepper, S., Moore G.: *XML Topic Maps (XTM) 1.0*; March 2001