Collaboration and the Knowledge Economy: Issues, Applications, Case Studies
Paul Cunningham and Miriam Cunningham (Eds)
IOS Press, 2008 Amsterdam
ISBN 978–1–58603–924-0

POPEYE, a Peer-to-Peer Collaborative Working Environment over Mobile Ad-hoc Networks

Paolo GIANROSSI[1], Christian MELCHIORRE[1], Patrizio PELLICCIONE[2]
[1]*Softeco Sismat S.p.A., Via De Marini 1, Genova, 16145, Italy*
Tel: + 39 010 6026 1, Fax: + 39 010 6026 350
Email: christian.melchiorre@softeco.it, paolo.gianrossi@softeco.it
[2]*Università dell'Aquila, Dipartimento di Informatica, via Vetoio, 67100 Italy*
Tel: +39 0862 433734, Fax: +39 0862 433131, Email: pellicci@di.univaq.it

Abstract: Most currently available tools supporting collaboration exploit a rigid client-server paradigm and fixed communication infrastructures (e.g. the Internet). Peer to Peer (P2P) systems represent an alternative, however, despite the growing success of Internet based collaboration application, P2P technology research has given little attention so far to specific issues related to wireless mobile environments. The POPEYE research project brings an answer to fundamental needs of mobile information sharing and collaboration, free from the constraints of relying on fixed infrastructure, in a flexible and simple way while supporting dynamic spontaneous collaborative group working environment services with the appropriate quality (such as persistence, synchronisation, security, etc.).

1. Introduction

Next generation collaborative systems will offer mobile users seamless and natural collaboration amongst a diversity of agents, within distributed, knowledge-rich and virtualized working environments. This ambitious goal faces numerous challenges, from the underlying communication infrastructure through to security, the limited capabilities of mobile handheld devices and the need of high level application services.

Although useful and increasingly put into regular use, current Collaborative Working Environment (CWE) solutions have several limitations and relevant needs for further research. Most systems address either rather traditional and rigid intra-organizational collaboration scenarios or, at the opposite, completely free and unstructured open communities' interactions. Emerging dynamic, more flexible and ad-hoc collaboration schemes are hardly or not supported at all. Explicit representation of the collaboration context, of workers' and team's goals and the semantics of underlying business process are not addressed, and this makes difficult to ensure the context and process awareness, personalisation and collaboration support required in knowledge-rich interactions.

In this setting, the POPEYE project considers P2P over wireless ad hoc groups, where fixed infrastructure is not a prerequisite, where virtual communities can emerge spontaneously and share data with the appropriate quality of service (persistence, synchronisation, security, etc.).

The POPEYE infrastructure is assessed through representative and challenging selected mobility-enabled peer-to-peer e-collaboration applications and two demonstration events during the project lifetime allow the validation of the POPEYE proof-of-concept prototype in realistic situations.

2. Scope and Objectives

Many research, open source and commercial systems have been built to support Collaborative Working Environments (CWEs) i.e., computer and communications based systems designed to facilitate communication, collaboration, and collaborative work by groups, organizations, and societies. The most popular are, on the side of commercial systems, BSCW[i], Microsoft's Sharepoint Services/Portal[ii], KAVI[iii], GROOVE[iv], Lotus Sametime Connect[v], and, on the side of open source systems, PhpGroupware[vi], more.groupware[vii], TikiWiki[viii]. Such applications would typically offer all or part of the following features:

- Document Management and Sharing (group document repository for secure storage and management of documents, comments and change requests collection, etc.);
- Group Communication (email reflector dedicated to each group, email archive, group calendars, notification of a new document, of a poll/ballot, of a calendar event or action item assigned, etc.);
- Polling/balloting System (vote per member or one vote per company, voter eligibility check, polling/balloting rules management for quorum, abstain, etc.);
- Process Controls (action items tracking, members roles and privileges management and control, etc.);
- Virtual meetings (white board, screen sharing, instant mail/chat, audio/video conferencing, moderation feature, etc.).

Although useful and increasingly put into regular use, current CWE solutions have several limitations and relevant needs for further research are generally acknowledged. Most systems address either rather traditional and rigid intra-organizational collaboration scenarios or, at the opposite, completely free and unstructured open communities' interactions. Emerging dynamic, more flexible and ad-hoc collaboration schemes are hardly or not supported at all. When most of the currently available tools supporting collaboration exploit rigid client server architecture and rely on a communication infrastructure like the Internet, POPEYE's ambition is to get collaborative working free from such constraints.

The baseline scenario chosen to illustrate how POPEYE addresses these features in a mobile, peer-to-peer ad-hoc network environment–describes the activities involved in the organisation of a conference or symposium in a location where no network infrastructure is available (see section 4). This scenario was further developed in the project public Whitepaper (see ref [2]).

To address these issues, overall, POPEYE:

- Draws out an integrated overlay networking architecture that combines the stability and performance typical of infrastructure networks with the flexibility and spontaneous character of mobile ad hoc communications,
- Develops a communication platform to provide efficient P2P management and communication primitives, and
- Supplies higher-level context-aware, secure and personalised core services to facilitate application development by allowing the combination of user preferences with ambience information, such as time, location, user activity, and peers' presence.

The POPEYE system is intended to be used as easy as possible from the user's point of view. When POPEYE users want to collaborate they just have to start the POPEYE Environment on their wireless device and login to the POPEYE network. The device connects to a P2P overlay network which is built upon a mobile ad-hoc network (MANET). This happens invisibly to the user. In the next step a user can search for and join existing Workspaces or he creates a new Workspace and invites other available users to join him for collaboration. In POPEYE the Workspace is the term used to designate a group of users and the data and applications they share. The users who joined a Workspace form the Group

that belongs to the Workspace. Sharing of data between all members of the Group is supported by the Shared Space which is associated to each Workspace. The applications the users employ for collaboration (e.g., file sharing, group calendar, whiteboard, etc.) can be plugged into the user's local POPEYE environment at runtime. We call those applications Plug-ins. Their instances are associated to a Workspace and specific configurations are named (Plug-in-) sessions. The POPEYE system is secure and context aware at all times.

3. Methodology

POPEYE is a project funded under EU's 6th Framework Programme, Information Societies Technologies priority (IST). A comprehensive engineering process was applied to develop a proof of concept application prototype out of POPEYE research. The engineering process of POPEYE has been a modification of the Waterfall process in which each different phase of requirements definition, system and software design, implementation and unit testing, integration and system testing, and maintenance has to be completed before moving onto the next phase. In practice these phases overlapped and fed information to each other.

The requirement engineering process comprised end-user scenario definition (e.g. public, open event such as a conference, a workshop, a trade fair, etc.), requirement elicitation from scenarios, function elicitation from requirements, traceability of the developments to the original end-user scenarios, integration, testing and full-scale live demonstrations for validation against the requirements.

In order to match professional integration and validation standards, POPEYE tests and demonstrations relied upon a dedicated POPEYE User Group that has been formed since the early stages of the project and that involves external Collaborative Working Environment experts.

The POPEYE software is built based on available open source tools. An open source version of the POPEYE framework is released not only for further developments and research, but also for use in mobile, collaborative end-users communities[ix].

4. Business Case Description

The most important elements that characterise the issues addressed by POPEYE are summarised by the following baseline scenario, illustrating how the activities involved in the organisation of a conference or symposium, in a location where absolutely no network infrastructure is available, are supported by a simple and reliable computing environments for collaborative work:

"Today is Monday 7th December 2009 and Sandra is early on her way to Château Villette, near Versailles. Sandra is the Organiser of the Annual World Symposium on Sustainable Development that is due to start today. All previous such meetings took place in one or another major conference centres throughout the world, this time Sandra obtained a great bargain in renting a huge Castle in the countryside near Versailles in France.

The main justification for this bargain being the almost complete absence of communication infrastructure, with the exception of a few fixed telephone lines and the relatively good signal strength of the main mobile operators, thanks for the castle being on a small hill. This symposium, like most such meetings these days, heavily relies on modern computing environment and multimedia data but Sandra needed no more than what Château Villette had to offer.

Indeed, since a little research project called POPEYE showed the way a few years ago, it has become extremely easy to share data in near real time within spontaneous virtual communities without any pre-requisite for a complex infrastructure like Internet or even a simple LAN. As it is, the symposium participants all have access to the each other's data that they decided to share (such as a multimedia presentation or a text document) on their

own PDA or laptop. Later this day, when the participants arrive, they are impressed by the simplicity and reliability of the new collaborative environment offered to them and they are amused to review some of the most impressive documents that they have access to by simply switching on their laptops.

Sandra is confident when she gives her kick-off talk to the Symposium, instantly displaying a few animations to each participant. There are some questions raised about the agenda and everybody can browse through all the comments and suggestions put forward by all participants. Sandra answers some questions directly and her assistant types these answers in the resulting collaborative document. Some amendments to the agenda are necessary and every participant also gets the update in real time.

When joining the parallel working groups, the participants join the corresponding community and instantly are able to collaborate and share data, the results of the working groups are later passed to the general symposium community for everybody to use at will.

When the polling session start on the third day, stronger security is required and each voter has to authenticate himself to join the restricted area of the dedicated working environment. On top of that, with a single connection of her own laptop to the Internet, Sandra (or any participant for that purpose) offers the few major stakeholders that could not make the trip the possibility to join the sessions and even to vote in near real time during the polling sessions."

This conference scenario, further developed in the project public Whitepaper (see ref [2]), covers most of the typical situations where collaborative work is involved. The final story was also elaborated further in order to include the results of the study of several different scenarios during the requirement elicitation process in early phases of the project.

5. Technology Description

Based on stated requirements we iteratively defined the software architecture of POPEYE; the final version obtained at the end of this iterative process is represented in Figure 1.

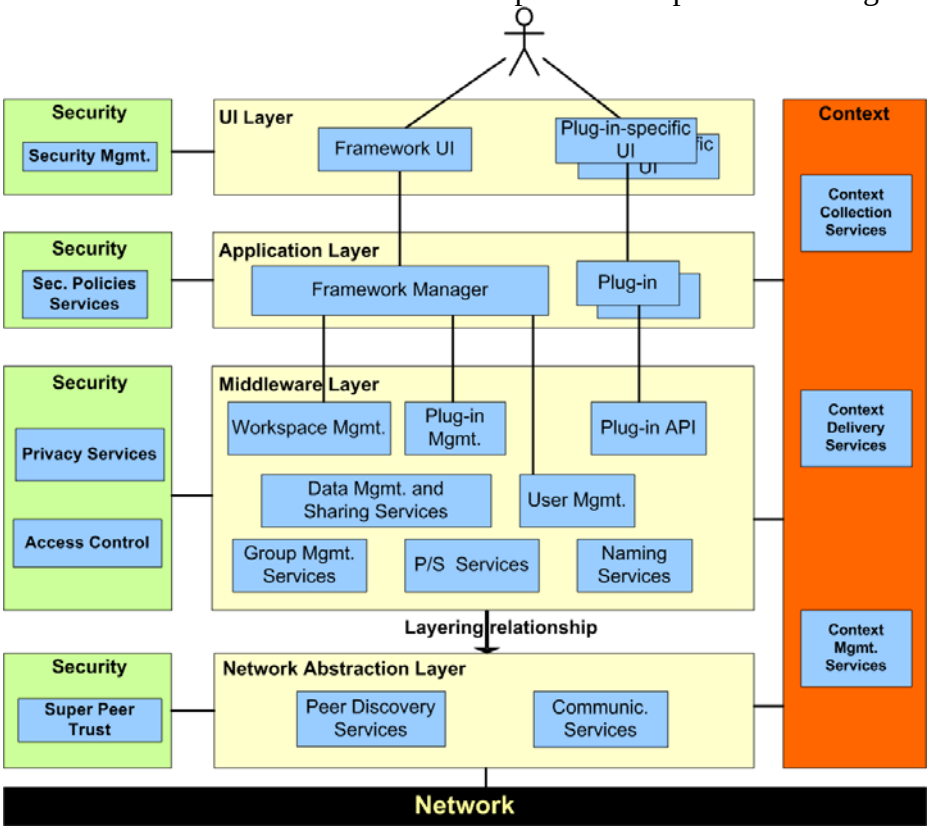

Figure 1: Popeye Software Architecture

The architectural style we followed is the layered one and we identified 22 components distributed in four layers and two macro-components: the User Interface, the Application, the Middleware, and the Network Abstraction layers as well as the physical Network. Security and Context macro-components are orthogonal to the five layers. Each layer and macro-component provides/requires services to/from adjacent layers, through provided/required interfaces. Since each context service is accessed by all POPEYE layers (a real orthogonal context) it is represented by only one component. On the contrary, the security macro-component provides different services to the different POPEYE layers.

5.1 The UI Layer

According to Model-View-Controller (MVC) paradigm [4], the UI Layer offers a standard GUI, offering to the users an access to both the POPEYE framework (Framework UI) and its applications (Plug-in specific UIs).

5.2 The Application Layer

The components in the Application Layer implement services to be built on top of the POPEYE infrastructure. The subcomponent Framework Manager represents the real POPEYE client application core. It constitutes the Controller in a MVC architectural pattern (where the middleware layer constitutes the Model and the User Interface layer acts as the View). The Plug-ins components represent the applications which are used for collaboration. They can easily be plugged into the local POPEYE system at runtime. Plug-ins can register themselves or can be retrieved by using the Framework Manager. Multiple instances of the same Plug-in can run on the local POPEYE system (even in the same workspace). Possible dependencies between plug-ins (a plug-in could need other plug-ins to be already loaded) are managed by the POPEYE plug-in infrastructure. The POPEYE plug-in infrastructure allows plug-ins to communicate each other.

5.3 The Middleware Layer

The Middleware Layer provides the core functionalities of the POPEYE framework architecture. It is organized in eight different modules that make use of the services supplied by the Network layer and Security and Context macro-components. The Data Management and Sharing Services module offers a distributed storage system which allows group members to share data. This system makes use of context information to replicate data in some of the participating devices in order to achieve accessibility from anywhere in the network and the efficient use of the network resources. The User Profile Service stores and delivers information about connected users. It also provides means to search for users along different criteria taking into account also contextual information. The Workspace Management module controls the lifecycle of the existing Workspaces. It handles their profiles including lists of authorised users and the assignment of manager rights for each Workspace. It also controls the associated shared spaces and all instances of applications which we call sessions. Furthermore, this module enables to search for Workspaces following different criteria. First, the Publish/Subscribe Service module offers a fully-decentralized topic-based subscription mechanism, where durable and non durable topic subscriptions are supported. Furthermore, it supports message retransmission of lost messages due to temporary disconnection. The Naming Services module binds a name to a specific resource and at the same time provides event notification whenever naming entries are modified. Built on top of these modules, the Group Management Service functionalities support multiple group creation where members can join several groups at the same time. As described before the POPEYE collaborative tools are implemented as plug-in extensions. The Plug-in Management module in this layer offers mechanisms to download

and install plug-ins that POPEYE users make available to the network. The Plug-in API represents a well-defined interface that encapsulates the set of those middleware functions which are to be accessed by (third party) application plug-ins.

5.4 The Network Abstraction Layer

The Network Abstraction Layer provides a suitable abstraction of the network to the other layers. Mobile ad hoc Networks (MANETs) are mainly characterized by the mobility of the nodes, and the scarce bandwidth. The mobility of the nodes is the responsible of the continuous partitions of the network. Furthermore the scarce bandwidth requires ad hoc routing protocols to create efficient routes between nodes, maintain and reconstruct them whenever they break without generating too many control messages. POPEYE makes use of DYMO (Dynamic MANET On-demand Routing Protocol) [5] to manage the unicast communication and of MMARP (Multicast MANET Routing Protocol) [6] to manage multicast communication. DYMO is specifically designed to work with heterogeneous devices in terms of capacity and computation power and can scale very well to hundreds of nodes, and it has a very low control overhead. On the other side MMARP provides efficient multicast routing inside the MANET and incorporates additional functionalities to deal with the complexity of interoperating smoothly with fixed IP networks. In order to support scalability the POPEYE software architecture in a scalable manner, we group the nodes into clusters. In each cluster, exactly one distinguished node - the superpeer (SP) - is responsible for establishing and organizing the cluster. The SPs are responsible for sending SP Info messages in their clusters, containing administrative information for the cluster members. They also form an overlay which allows the communication with adjacent clusters. Clustering also affects the routing that is consequently divided into two parts: routing within a cluster (intra-cluster) and routing between different clusters (inter-clusters). Communication between two clusters is always routed via SPs. This approach reduces significantly control messages in the whole MANET.

5.5 The Security Macro-Component

The Security macro-Component provides security services to the four POPEYE layers. The main security object in POPEYE is the certificate. This certificate is exchanged between peers in every operation because it is the only object that represents the peer. Thus, the only way to trust each peer in POPEYE is by signing every message exchanged between them. First of all a POPEYE user must have a POPEYE certificate which identifies him into the MANET. This certificate is released by a certification authority. Security mechanisms are also defined to the level of groups and workspaces.

5.6 The Context Macro-Component

The Context macro-Component manages the contexts coming from the environment that can be generated by several sources, such as devices, users, sensors, software modules and even external applications. Each entity capable of generating context is called context source. Users have a profile with contextual information (such as the people they know, the topics they are interested in and the social relationships with other users) and sensors deliver data periodically. The main goal of the context source abstraction is to wrap the entity generating context in a common semantic framework, so the generated context can be readable and understandable by the rest of the system.

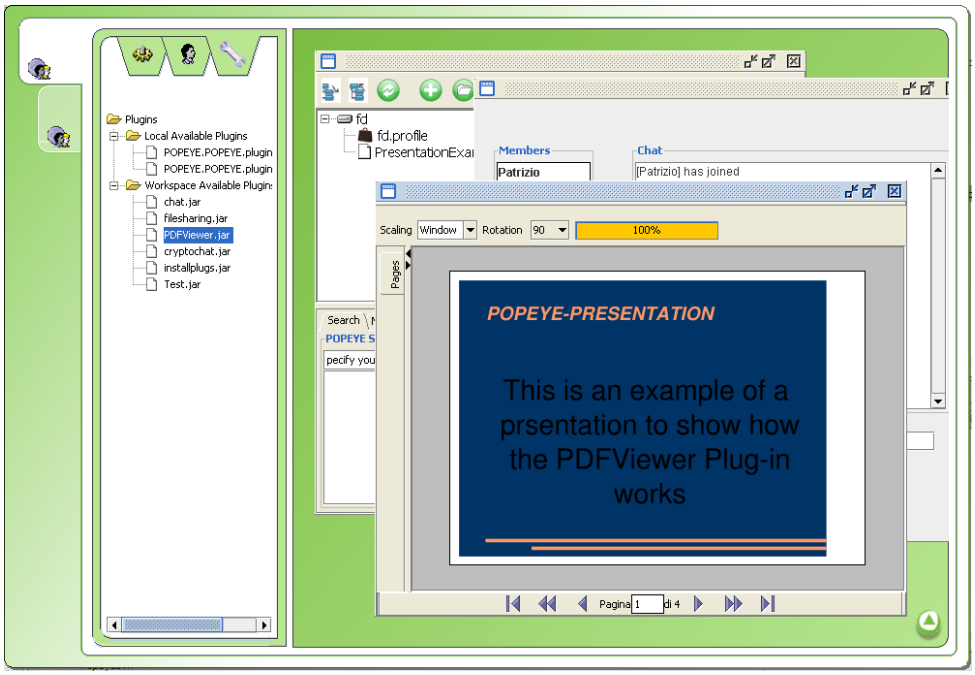

Figure 2: Popeye Tool

This architecture, refined during the entire development process, has been the blueprint for the implementation of POPEYE. In order to enhance maintainability and reuse each layer and macro component became a package containing packages and classes according to the subcomponents of the layers and the macro-components. Figure 2 shows the POPEYE tool with the PDFViewer, Chat and FileSharing plugins. At the left side the reader can see the plugins locally available and the plugins available in the workspace.

6. Conclusions

Although useful and increasingly put into regular use, current Collaborative Working Environment (CWE) solutions have several limitations and relevant needs for further research. Emerging dynamic, more flexible and ad-hoc collaboration schemes are hardly or not supported at all. POPEYE research investigates a number of key issues in mobile, secure, context-aware P2P collaboration over MANETs, delivering a multi-layered, scalable, open framework that enables the development of mobile collaboration services and integrated applications. POPEYE aims to give a solution on how to provide support for ad-hoc cooperation, with the appropriate quality of service, in situations where the fixed network infrastructure is absent or cannot be used. The two-year POPEYE Project has reached its end in April, 2008, delivering a proof-of-concept prototype application that gave very positive feedback both from the point of view of the underlying research work and the future implementation.

The major drawback of the prototype is that we require two MANET transport protocols (DYMO, MMARP) that currently only work in the Linux platform. To solve this portability issue and ease configuration management we began a research line on Application Layer MANET protocols. In this line, we developed a Java implementation of the OLSR protocol (jOLSR) and an optimized overlay multicast over jOLSR named OMOLSR. The overall idea for a future work would be to replace POPEYE's network abstraction layer with this Application layer protocols. This would enable the portability of POPEYE to any platform with a Java Virtual Machine and would also reduce the configuration complexity of the system[x].

Acknowledgements

POPEYE is a Specific Targeted Research Project (STREP) partly founded by the EU under the 6th Framework Programme, IST Priority (project Contract No. IST-2006-034241).

The authors would like to thank all partners and subcontractors participating actively to the project for their continuous and qualified support: Thales Communications, Universidad de Murcia, Università dell'Aquila, Universidad Rovira i Virgili, GET ENST – École nationale supérieure des télécommunications, OFFIS e.V., Softeco Sismat SpA.

References

[1] POPEYE public web site *:* http://www.ist-popeye.eu/

[2] POPEYE Project White paper: http://srvweb01.softeco.it/IST-Popeye/Portals/Rainbow/Documents/POPEYE_whitepaper_3.0.pdf

[3] H. Dung Ha Duong, C. Melchiorre, E. M. Meyer, I. Nieto, G. Paris, P. Pelliccione, F. Tastet-Cherel, "A Software Architecture for Reliable Collaborative Working Environments" (short paper), Proc. of the 5th International Workshop on Distributed and Mobile Collaboration (DMC 2007), IEEE WETICE, Paris, France. June 19th, 2007.

[4] "Model-View-Controller (MVC)," http://java.sun.com/ blueprints/patterns/MVC.html.

[5] Dynamic MANET On-demand Routing Protocol (DYMO). http://moment.cs.ucsb.edu/dymo.

[6] P. Ruiz, A. Gomez-Skarmeta, and I. Groves. Multicast Routing for MANET Extensions to IP Access Networks: The MMARP Protocol. In International Workshop on Mobile IP-based Network Developments., pages 75–81, October 2002.

---

[i] BSCW, Basic Support for Cooperative Work. http://www.bscw.de.

[ii] Microsoft Sharepoint Services-Portal. http://www.microsoft.com/sharepoint

[iii] KAVI website. http://www.kavi.com.

[iv] Groove virtual office web site (Microsoft Corporation). http://www.groove.net

[v] Lotus Sametime Connect web site. http://www.lotus.com/sametime

[vi] PhpGroupware web site. http://www.phpgroupware.org/

[vii] more.groupwarewebsite. http://mgw.k-fish.de/

[viii] TikiWiki website. http://tikiwiki.org/tiki-index.php.

[ix] The home page of the POPEYE project is http://www.ist-popeye.eu while the source code can be found in http://sourceforge.net/projects/popeye-cwe/, released under the GNU General Public License (GPL).

[x] See: http://ast-deim.urv.cat/wiki/OMOLSR for more information.