Web Infrastructure for Collaborative Applications

edited by
Robert Tolksdorf* *tolk@cs.tu-berlin.de*
Gustaf Neumann† *neumann@wi-inf.uni-essen.de*
Wolfram Conen‡ *conen@wi-inf.uni-essen.de*
Peter Bertok§ *p_bertok@rmit.edu.au*
Mathew Fuchs¶ *matt@wdi.disney.com*

Abstract

The attendees of the workshop "Web Infrastructure for Collaborative Applications" tried to develop a joint opinion how the Web and its infrastructure should develop to become a productive environment for collaborative applications. This paper sketches the different approaches presented in the workshop and summarizes the consolidated results of the discussions.

1 Introduction

The workshop "*Web Infrastructure for Collaborative Applications*" at WETICE'96 brought together 37 attendees from academia and industry. 14 selected papers were presented in the sessions named "*Case Studies*", "*Tools, Methods, Techniques*", "*Application Requirements*", and "*Collaborative Engineering*". A final discussion session and the collaborative effort of all participants to produce a summary of the workshop results finished the sessions.

The call for papers focused on the following main question: "*Can the Web serve as an infrastructure for both developing and implementing business applications in a possibly global, distributed, and collaborative business environment?*" This main topic was broken down into three topics:

1. What are the requirements imposed by the application area, what has to be supported?
2. What are the requirements from the implementation point of view, what tools and methods are needed?
3. Which are the infra-structural consequences of the identified requirements, what do we need in the basic environment?

*TU Berlin, FB 13, Informatik, KIT/FLP, Germany
†Universität Essen, Wirtschaftsinformatik, Germany
‡Universität Essen, Wirtschaftsinformatik, Germany
§Royal Melbourne Institute of Technology, Victoria, Australia
¶Walt Disney Imagineering, Glandale CA, U.S.A

During the discussions among the participants three interest-groups developed: the "*analyze-problem-task!*" group, the "*use-tools!*" group, and the "*develop-technology!*" group. The discussions in the workshop demonstrated clearly that it is both difficult and necessary to integrate these points of view to obtain answers to the questions mentioned above. Some of the answers advocated in the joint discussions are documented below.

2 Collaborative Applications Requirements

The World Wide Web provides the most globally available and accepted environment for accessing distributed information sources. This makes it the leading contender for the integration of existing and future business applications. It was suggested that the primary success factors for the acceptance of the Web are

- its platform independence,
- the use of the Internet, and
- its easy to use hypertext-based interface.

The development of the basic Web concepts followed a relatively general vision of "*What is wanted?*" and a relatively precise notion of "*What is technologically possible?*". A rigid analysis of general application requirements has to precede any discussion or evaluation of the usability of the Web as an infrastructure for developing and implementing collaborative application. In the following section elements of such an analysis are provided in this and the following section. The step of relating these requirements to recent and future Web concepts is discussed in the sections 4 and 5.

2.1 Applications Characteristics

Collaboration itself may serve as an example for an unprecisely defined term. According to the Webster's dictionary, to collaborate has the following meanings:

1. to work jointly with others or together esp. in an intellectual endeavor,

2. to cooperate with or willingly assist an enemy of one's country and esp. an occupying force,

3. to cooperate with an agency or instrumentality with which one is not immediately connected.

If the first definition is adopted, there is still enough room to interpret: what does it mean to *"work jointly together"* exactly? What is the notion of vicinity in time, space, awareness, or, more general, what context is required to qualify an act of exchanging matter, energy, or information (interaction) as collaborative? Clearly, some coordination of the interaction of the participants in the collaboration is required, and this in turn requires the use of communication. In addition there is an implicit necessity of matching individual goals to allow the emergence of collaboration in an application context.

2.2 Problem Analysis in Collaborative Applications

Proponents of the *"analyze-problem-task!"* group concentrated on the problem how complex problem tasks can be decomposed and managed in a collaborative environment.

The overall objective of a collaborative application is to solve a problem where several agents have to collaborate. Examples for such applications are: asynchronous design by engineers, joint work on reports, project management, etc. Every effort to support collaborative applications can be positioned at a specific location in the layers of Figure 1.

Collaboration can be direct or indirect. Generic forms of direct collaboration are meetings, working groups, synchronous and asynchronous work patterns, delegation etc. Indirect collaboration occurs when results of others can be reused without explicit coordination with the original problem solver.

Coordination is necessary to perform a directly collaborative task. It requires the use of communication and planning. Coordination structures can be influenced by the problem requirements, organizational structure, environmental constraints, policies, and human preferences. Coordination can be achieved by directives, discourse, market mechanisms, voting etc.

Along the dimension of communication, several communication patterns can be observed. These include peer-to-peer, multicast, etc. We have not specified the nature of the collaborating entities. Both human and computational agents can be participants in a collaboration and our technological choices must support them both.

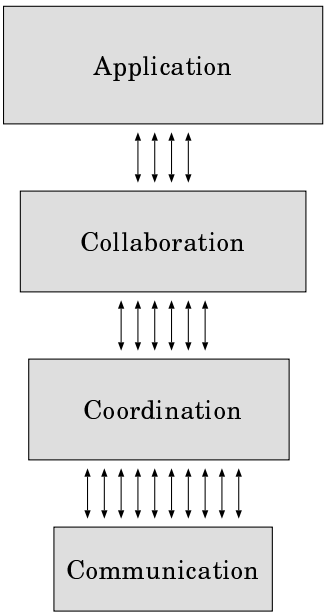

Figure 1. Infrastructural Requirements of Collaborative Applications

2.3 From Tools to Application Programs

Proponents of the *"use-tools!"* group view an application as a set of tools providing the functionality of the various application aspects. We can distinguish application specific tools from generic tools. Application specific tools are developed for a single application such as for interacting with special devices like laboratory equipment.

Generic tools are developed for solving common and generic applications aspects. Examples are chat and editing facilities, white-boards, audio- and video conferencing, object request brokers, interface repositories, naming services, etc.

An application program is a collection of generic or specific tools which are integrated to solve the problem task. The Web may provide an environment in which such a collection of tools can be integrated into an application which is easily accessible, simple to use, and expandable with respect to new media types and tools.

2.4 Requirements for Collaborative Application Programs

Proponents of the *"develop-technology!"* camp analyzed the requirements of collaborative applications from a technological point of view. Collaborative application programs should:

- **Support different domains of trust.** Users and applications impose the need for differentiated domains of trust. Workspaces of various forms ranging from private over group-wise to public have to be supported. Their properties have to be adjustable by users.
- **Account for users capabilities.** Applications should take technical factors of the system and personal abilities of the user into account. Examples for technical factors on the Web are the available network bandwidth and the available interaction technology. An adjustment of form and content should be possible according to the environment of the users and their capabilities.
- **Speak the language of the applications domain.** Any application should "speak" the language of the application domain, not that of the underlying implementation. This supports collaboration among tools and allows applications to be adjusted for users, as mentioned above. Thus, functionality should be expressed in terms of the applications domain, not in "tech-speak".

3 Development Tools, Methods and Models for Collaborative Applications

The call for papers identified several key issues in collaborative Web based development, such as development of business applications, component based development of complex applications, coordination and integration aspects. The papers presented at the workshop addressed many of these issues, and proposed solutions for complex problem solving, modular design tools, and coordination technology. One **design tool** was presented that allowed to model the access structures and - up to a certain degree - the presentation of Web sites.

In the discussions during and after the presentation sessions several problem and deficiency areas concerning the **basic mechanisms** available in the Web were identified. In the final discussion session the following concrete problem areas were identified as important especially for collaborative applications on the Web:

Area	Problem
Time	Low level of support for synchronous collaboration
Location	No virtual URL or replication mechanism exists
Technology	Extensions lacking in robustness
Models	Need for a general framework that includes process models, development models, domain models, testing models

Another promising field which is currently in a very early state is the field of development models for collaborative applications. One approach towards a general frameworks for applications is based on **reference models** which can be instantiated to produce application specific models. Models for various areas of

- business administration,
- public administration, and
- personal/private information systems

can be developed that address common collaborative problems in these application domains. The presentations at the workshop addressed only problems of business administration, but other application domains are emerging with significant shares as well.

Once these models are available the need for appropriate testing methods and a **test suite** will emerge that can define and validate the provided services at different levels, e.g. minimum, optionally full-blown services. The testbed should aim at core services as well as other components, such as different interfaces (API, etc.).

The discussions concluded with the general opinion that collaborative development tools are very promising applications of Web technology, but at its current level the technology does not offer all the methods and components that users would need.

4 Infrastructure

Several talks of the workshop emphasized the potential of the Web for the development of collaborative applications. Todays core technologies of the Web are

- HTML for the textual representation,
- HTTP as the access protocol, and
- CGI to generate dynamic pages.

Vendors and research institutions experiment with more or less established extensions, which are partly on the client-side (such as plug-ins, or Java) or on the server side (such as for example database interfaces). Standards for virtual reality are being developed (e.g. VRML). Figure 2 depicts these components.

However, these components do not provide enough support for collaborative applications.

4.1 Deficiencies of the Current Web

Statelessness

HTTP as it is defined today is a stateless protocol. However, many applications on the Web (such as Internet shopping

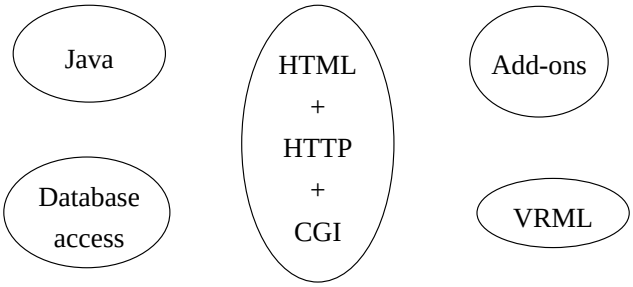

Figure 2. The Current Web

with shopping bags, search engines, etc.) need to maintain some state within and across sessions. Today, state has to be maintained using a number of mechanisms that are rather tricks than integral concepts of HTTP.

Semantics

Applications deal with information that is represented as text, graphics, and other media. HTML as the mark-up language of the Web, however, deals primarily with the visual appearance of and links among the provided information. Thus, currently, there is no uniform mechanism to describe the semantics of information contained in Web pages. Results of this deficiency are phenomena like ”lost in hyperspace”, and the variety of problems finding information on the Web.

Security

Currently, the issue of security on the Web is approached by a variety of solutions. For the support of collaborative applications, a security framework is needed that satisfies the needs of, for example, financial transactions, and supports differentiated access rights as a result of different domains of trust.

Notification

The current Web follows a “pull model” of information gathering: A client actively pulls information by requesting it from the server. Mechanisms such as client-pull provide only a mean to poll a single Web page. In order to support collaborative applications, a “push model” has to be developed that allows applications to notify users of new or changed information or state of the application. This should be part of a move to a general peer-to-peer architectures that is needed to support a wider spectrum of collaboration types.

4.2 Overcoming the Deficiencies of the Web

To overcome the mentioned deficiencies, several directions have been identified in the workshop. Future applications will need to incorporate distributed objects and active components. Currently the trend for distributed objects favors CORBA, which allows programming language independent sharing of data structures across the Internet. Distributed objects will hide many implementation peculiarities (such as establishing session state) and will ease data sharing in distributed applications.

Similarly, applications should be able to work on a homogeneous abstraction of the underlying hardware. Platform independent execution mechanisms (such as virtual machines like the Java VM) will be needed for higher interactivity and for the mobility of code.

In order to ease the semantic access to Web resources the deficiencies of the current text based search engines need improvements. Means for semantic annotations should be provided for this task. Meta grammars like SGML can be seen as simple knowledge representations suitable for semantic annotations while being a conservative extension of current practice. Semantic annotations can be provided automatically for example when databases are accessed or when data is entered via forms.

A key issue in using the Web as the infrastructure for collaborative applications is adherence to Web standards. The World Wide Web Consortium is the organization that promotes and develops those standards. Activities are under way to standardize Web components by the ISO, too. The use of the Web platform should be restricted to standardized features.

4.3 The Ideal Web

Figure 3 shows our view of the “ideal” Web that could be the basis for the infrastructure for collaborative applications.

The core of the Web should be transformed into a set of components of higher abstractions capable of describing and inferring knowledge about information. Collaborative applications could greatly benefit from that knowledge, the results of collaboration could in turn contribute to that knowledge. Notification mechanisms and virtual machines should become integral parts of the Web infrastructure. In order to address information on a more abstract level, generalized smart URLs should be introduced. These could make use of the URN/URC mechanisms currently proposed, or provide means to address agents that furnish information.

The overall guideline for developments should be “evolution instead of revolution”. This guideline led the Web to its current state and will ease the acceptance of future

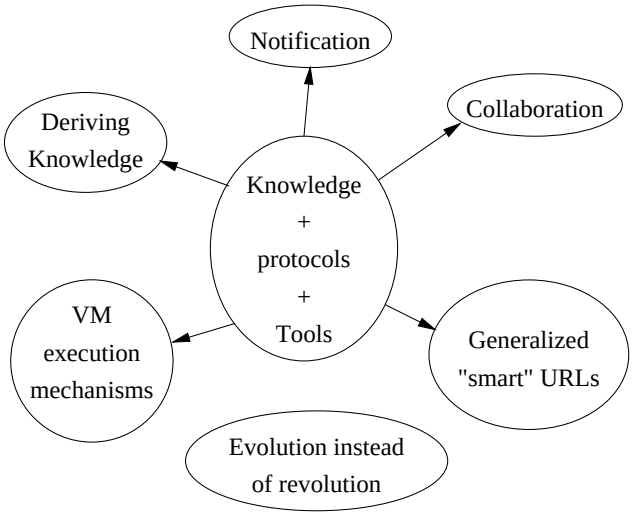

Figure 3. The “Ideal” Web

enhancements.

5 Results and Future Directions

The direction which a constructive and sufficiently complete analysis of an application area may follow: (1) Determine the general characteristics of the area, (2) analyze these characteristics and relevant examples to deduce a set of requirements, (3) check which requirements can be fulfilled with the infrastructure the chosen implementation environment offers, and, finally, (4) specify which enhancements have to be made to this implementation environment to meet the requirements of the application area.

What the discussion about the future of the Web clearly lacks is a precise terminological framework and a reference model allowing to categorize the application areas and to reuse implementation concepts (building blocks) with respect to the requirements imposed by collaboration, coordination and communication needs. The papers presented in this chapter all add new aspects worth consideration in trying to construct such a framework.

The slides containing the results of the workshop are available on the Web at the location `http://www.cs.tu-berlin.de/~tolk/wsslides.html`.