Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT

Information and Software Technology 46 (2004) 423–431

INFORMATION AND SOFTWARE TECHNOLOGY

www.elsevier.com/locate/infsof

A peer to peer (P2P) architecture for dynamic workflow management

Georgios John Fakas[a,1], Bill Karakostas[b,*]

[a]Department of Computing and Mathematics, Manchester Metropolitan University, Manchester, UK
[b]Centre for HCI Design, School of Informatics, City University, Northampton Square, London, Gtr. London EC1V 0HB, UK

Received 19 February 2003

Abstract

This paper presents the architecture of a novel Peer to Peer (P2P) workflow management system. The proposed P2P architecture is based on concepts such as a Web Workflow Peers Directory (WWPD) and Web Workflow Peer (WWP). The WWPD is an active directory system that maintains a list of all peers (WWPs) that are available to participate in Web workflow processes. Similar to P2P systems such as Napster and Gnutella, it allows peers to register with the system and offer their services and resources to other peers over the Internet. Furthermore, the architecture supports a novel notification mechanism to facilitate distributed workflow administration and management.

Employing P2P principles can potentially simplify the workflow process and provide a more open, scalable process model that is shared by all workflow participants. This would enable for example a WWP to connect directly to another without going through an intermediary, currently represented by the workflow process management server. P2P workflow becomes more efficient as the number of peers performing the same role increases. Available peers can be discovered dynamically from the WWPD.

The few currently existing P2P based workflow systems fail to utilise state of the art Web technologies such as Web Services. In contrast, using the approach described here it is possible to expose interoperable workflow processes over the Internet as services. A medical consultation case study is used to demonstrate the proposed system.

Keywords: Peer to peer; Workflow management; Business process management; Web services; XML; Simple object access protocol; BPEL4WS

1. Introduction

As we move to component and service-based architectures, decentralised coordination and automation of processes in an organization will take on more importance. Efficiencies are gained by integrating internal systems with internal users, partner systems, and external customers in order to create virtual organizations, where users can collaborate in order to produce and provide goods and services.

Peer to Peer (P2P) computing facilitates the sharing of computer resources and services by direct exchange between systems. Users can take advantage of existing desktop computing power and networking connectivity, allowing client computers to leverage their collective power to benefit the entire enterprise. This architecture has achieved considerable attention by mainstream computer users and by the PC industry. The Napster, MP3 music file sharing application, was set up in September 1999, and attracted more than 20 million users by mid-2000. The SETI@home program (2001), which uses distributed processing to analyse radio telescope data, has attracted more than 2.6 million users who have donated over 500,000 years of processor system time to the hunt for extraterrestrial intelligence.

This paper introduces a novel interoperable P2P workflow architecture that has several benefits over centralised ones. Employing P2P can potentially simplify the workflow process and provide a more open process model that is shared by all workflow participants. Such a system is more flexible and scalable as it is easy to redeploy workflow activities by invoking new participating peers, discovered dynamically from a pool of peers. This in turn leads to a significant increase in efficiency and eliminates delays that would otherwise be incurred if a centralised process management component had to coordinate a static set of available participants/resources.

* Corresponding author. Tel.: +20-70408411; fax: +20-70408859.
E-mail addresses: billk@soi.city.ac.uk (B. Karakostas), g.fakas@mmu.ac.uk (G.J. Fakas).
[1] Tel.: +44-161-2473537; fax: +44-161-2471483.

0950-5849/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/j.infsof.2003.09.015

According to the proposed P2P architecture, the process definition is not maintained and coordinated by a server workflow engine (in contrast to the Workflow Management Coalition (WfMC) reference model [27] and to the majority of the existing workflow architectures). Instead the process definition is decentralised and distributed. In this way a dynamic workflow process is achieved without the need to use a server. Each Web Workflow Peer (WWP) encapsulates adequate functionality and knowledge to execute activities and to decide which WWP needs to be activated next in the process chain.

For the development of such a system, a number of P2P specific and more general workflow management issues had to be investigated, namely:

- The specification of a Web Workflow Peer Directory (WWPD);
- The specification of Web Workflow Administrators and Peers (WWP);
- Defining workflow processes as XML documents shared amongst peers, i.e. the Workflow Process Description (WPD) document. Metadata approaches such as BPEL4WS, Web Services Description Language (WSDL) and UDDI were used as a starting point [8,11,24];
- The development of a Notification Mechanism based on XML messaging, used in the administration and management of the workflow process. In a P2P environment, peers need to be able to know the status of processes and notify others when their task is completed. Notifications also involve 'alerts' such as about overdue-tasks.

The rest of this paper is structured as follows. Section 2 surveys related work. Section 3 describes the P2P workflow architecture and its main components. Section 4 illustrates the application of the model to the implementation of a medical consultation workflow process. Finally Section 5 contains a discussion of the research results and plans for further work.

2. Related work

There are a number of different standards, technologies and research work relevant to our research objectives which are discussed below.

2.1. Workflow process description standards

The main proposers of the workflow standards described below are the WfMC, and the World Wide Web Consortium W3C [5,25,27].

- The XML Process Definition Language (XPDL) is a meta-model for describing workflow process definitions and also a grammar for the interchange of process definitions [28]. The Wf-XML Binding (WfXML) standard describes a language that is independent from any particular implementation mechanism that supports the exchange of messages between workflows [29]. The WfMC Workflow Interoperability standard defines an abstract protocol for interaction of workflow enactment services across business domain boundaries [7,26].
- The Web Services Flow Language (WSFL), proposed by IBM, is an XML language for defining workflow processes within the framework of the Web services architecture composition [9].
- XLANG is a notation developed by Microsoft for the specification of message exchange behaviour among participating web services supporting especially the automation of business processes [18].
- BPEL4WS replaces the existing IBM WSFL and Microsoft XLANG efforts by combining and extending the functions of these previous foundation technologies. At the core of the BPEL4WS process model is the notion of P2P interaction between services described in WSDL; both the process and its partners are modelled as WSDL service [8].

2.2. The peer to peer framework JXTA

The JXTA project is an effort by Sun Microsystems to standardise P2P development. It is an open source development effort that uses XML to encode and expose the availability of resources on P2P networks that use this framework. Using the JXTA shell and components, developers can create a number of what are emerging as standard P2P services, including instant messaging, collaboration, and content management. JXTA peers create a virtual network where any peer can interact with other peers and resources directly even when some of the peers and resources are behind firewalls or on different network transports. The JXTA code (which is available on www.jxta.org) is still an early release but it is considered to be usable [10,19,22].

2.3. Web services

A Web Service is a self-describing, self-contained, modular application accessible over the web. It exposes an XML interface, is registered and can be located through a Web Service registry. Programs can invoke a Web service using XML messages over standard Web protocols [23]. Web services standards include WSDL and Simple Object Access Protocol (SOAP) both standardised by the W3C. WSDL is an XML grammar for describing network services as a set of endpoints capable of exchanging messages, and SOAP is a protocol specification for invoking methods on servers, services, components and objects. SOAP codifies the existing practice of using XML and HTTP as a method invocation mechanism [30–32].

2.4. Related work in peer to peer workflow systems

The few existing P2P-based workflow systems are based on older protocols and technologies such as CORBA. In contrast the approach described here was influenced by new standards such as Web services and associated protocols (UDDI, WSDL, SOAP). The system described here builds upon previous work by the authors to base workflow technologies on XML and Internet architectures [3,4,12,13]

The BBN Service and Contract (S + C) workflow approach discovers, assembles, invokes, and adapts the application of distributed services (components and gauges) within a dynamic operating environment. The S + C infrastructure assumes a subscribe/publish distributed blackboard infrastructure provided by a Java agent framework. Services are described in a Resource Description Format [2].

Consilient was one of the most ambitious products in P2P that facilitate programmers to set up a workflow and users to change the course of the workflow 'on the fly'. It supported concurrent processing, and let users see their completed work. It was developed in Java and JSP [20].

RainMan is a distributed web-based workflow system developed in Java. It is a scalable workflow infrastructure that supports both flexibility in workflow participation and interoperability between heterogeneous workflow system components. RainMan is a loosely coupled collection of independent services that cooperate with each other rather than a monolithic system [21].

The ORBWork-METEOR2 system intends to reliably support the coordination of user and automated tasks in real-world multi-enterprise heterogeneous computing environments [17].

3. A P2P workflow management architecture

The proposed P2P architecture is based on the concepts of the Web Workflow Peer Directory (WWPD) and the WWP. WWPD is a directory system that maintains a list of all peers (WWPs) that are available to participate in Web workflow processes. Similar to systems like Napster, it allows peers to register with the system and offer their services and resources to other peers. During the execution of workflow processes, the WWPD assists WWPs to locate other WWPs and use their services and resources.

Workflow process administration is achieved by employing a notification mechanism. For instance, at the completion of an activity the WWP notifies the Administrator (so that an updated status of the process instance is maintained). Similarly, upon expiration of an activity deadline, the Administrator notifies the WWP responsible for the expired activity.

The proposed system departs from the WfMC reference model, which is based on a client–server architecture, with a server being responsible for the workflow engine, process enactment, definition, and management. In the proposed system, server functionality and data are distributed among the WWPs. The architecture is completely decentralised as no central workflow engine is employed to coordinate the process execution. The WWP encapsulates the necessary knowledge to perform the activities that it was assigned and also to delegate some of the process execution to other WWPs.

3.1. The WWP Directory (WWPD)

The WWPD is the only centralised feature of the proposed P2P workflow architecture. It is important to note here that totally distributed P2P solutions would probably be unfeasible for purposes like workflow management. Most P2P architectures employ some centralisation concepts (called for example 'super-peers'). Peers can register with the directory and advertise the workflow services they provide. Similar to the business entities and business services of UDDI [24], WWPD maintains a list of registered WWPs and their profile (IP address, list of tasks provided and administration data). In contrast to UDDI, however, WWPD is an active directory in the sense that it maintains live information about peer availability. To achieve that, it continuously interrogates registered peers, ensuring that they are active (e.g. by 'pinging' their IP addresses) and retrieves their current status ('busy', 'available'). By averaging the speed of the peer response and availability over a time period, WWPD maintains information about the quality of the service provided by the peer, speed of the connection, packet loss etc. [14,16].

3.2. Anatomy of the Web Workflow Peer (WWP)

A WWP Peer is, similarly to a Web service, a processing capacity with an interface that is exposed on the Web and which can be accessed using Internet protocols. Its interface describes different types of processing capabilities, each corresponding to a workflow activity. Combined, such activities form a workflow process. A WWP that initiates and administers the process is called the Administrator Peer. Other WWPs delegated to carry out workflow activities are called the Participating Peers (Fig. 1). Although conceptually there is a difference between 'administrator' and 'participating' peers, in practice all peers are capable of exhibiting the same functionality i.e. of becoming administrators in different workflow process instances.

1. The Administrator WWP initiates and administers the workflow process by supervising and controlling its execution. To do so, the Administrator may have to reallocate or cancel activities or issue deadline alert notifications to Participating Peers. This peer exhibits the following functionality:
   - *Process initiation*: The Administrator WWP initiates the process and invokes the set of peers that will

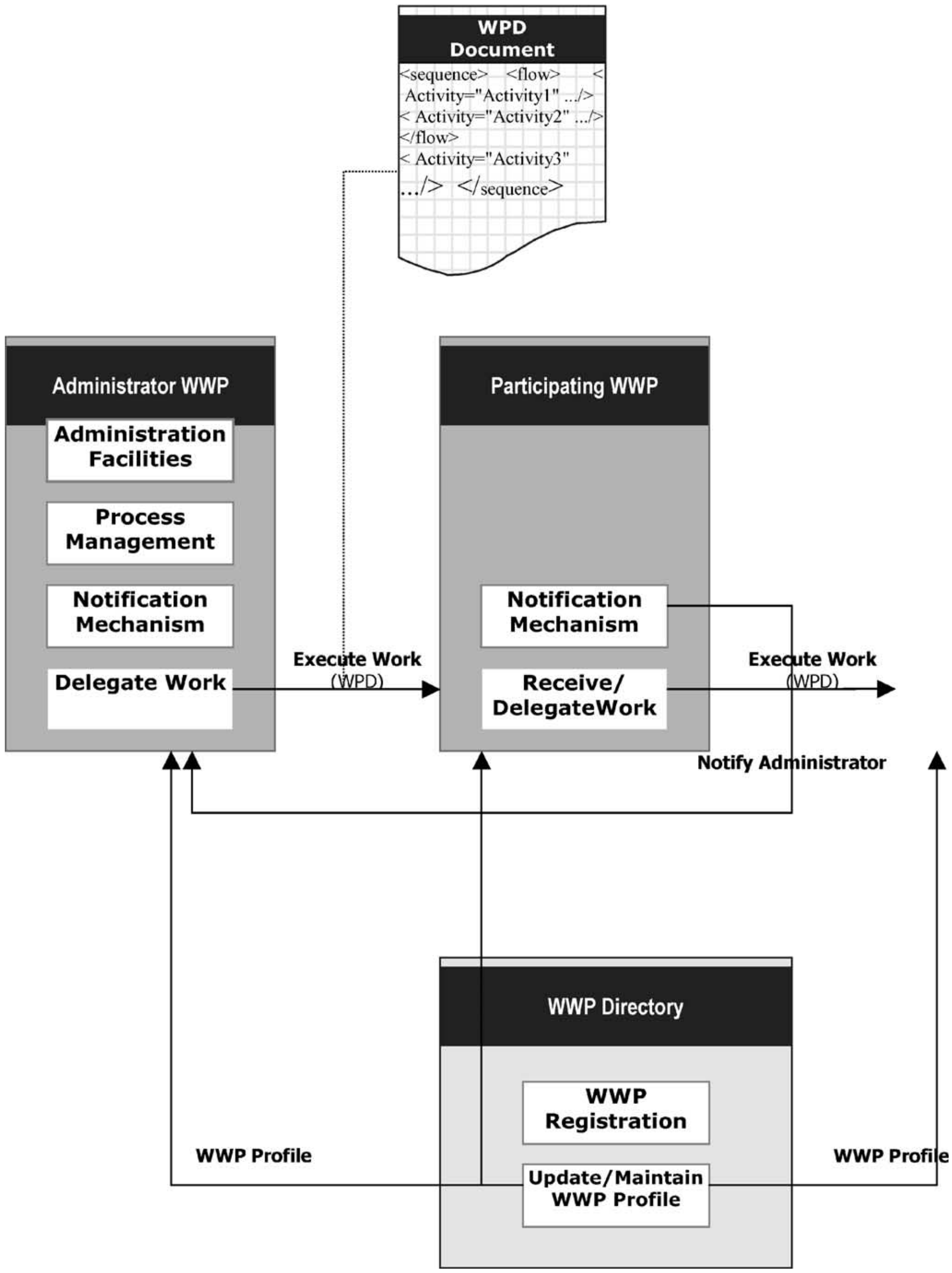

Fig. 1. The P2P workflow architecture.

execute the initial process activities. To identify suitable peers the Administrator employs the services of the WWP directory using selection criteria that consider peer availability, status, quality of service etc.;

- *Reallocation of activity*, in cases where peers become unable to complete their allocated activities on time they notify the Administrator WWP who then reallocates their work to other peers. A reallocation notification is sent to all affected WWPs;
- *Process management*; i.e. maintains the process current state, total running time, current activity and its status (Waiting Time, Deadline, current WWP), WWPs workload etc.;
- *Notification*, i.e. sends and receives notification messages to the peers about deadlines and processes status and WWPs status, etc.

2. The Participating WWP has the following functionality:
   - *Receives* activity to be carried out from the Administrator WWP and from other Participating Peers. This consists of a structured XML message containing a Workflow Process Description document (defined below). Upon successful completion of the activity the participating peer sends a completion notification

to the Administrator Peer to which the updated Workflow Process Description Document (described below) is attached;
- ◦ *Receive/send notification*, to and from other WWPs regarding deadline expiration etc.;
- ◦ *Work delegation*; the WWP is free to delegate work (or parts of it to other peers). In this sense the *Participating Peer* may become an Administrator Peer responsible for sub-processes within the overall workflow process.

3.3. Workflow Process Description (WPD)

The Workflow Process Description (WPD) is a document containing the data and meta-data of a process instance. WPD is defined as an XML document containing structural information about the process (i.e. order in which activities must be executed) and links (such as Universal Resource Identifiers—URIs) to resources (e.g. documents and data) required for their execution and/or produced as a result of their execution. Segments of the WPD document is transmitted from peer to peer attached as a parameter to a message. By attaching the WPD document to exchanged messages, we avoid having to force peers to refer to some central location/server to access workflow related information thus reducing the traffic on the network and improving performance. Similarly having the WPD document pointing to rather than containing the actual required data we avoid increasing the network traffic and overloading the peers with data. A participating peer for example may decide (based on the size and location of the data) to implement a particular fetching policy, i.e. to fetch data locally if and as needed or instead to access it remotely. Additionally, some of the data may be physically replicated across locations to facilitate the performance of the workflow process. Finally, different parts of the WPD document can be encrypted and made available to authorised peers only to ensure confidentiality and privacy of workflow data. The structure of the WPD document is illustrated in the example of Table 1.

Table 1 shows that a process instance document consists of several parts, i.e.:

- Process instance metadata;
- Process instance global data accessible by all WWPs;
- A process description part that describes the activities, their input/output data and the flow of activities as an XML document.

For example, Table 1 shows that Activity1 should use input data called ‘Name1’, available at location uniquely described by URL1. Activity1 can be executed in parallel with Activity2. This is shown by enclosing both activities inside the < flow > tags. After both activities are completed, Activity3 may commence.

Table 1
The WPD document

Process Meta-data: Process ID, Initiation Date/Time, Initiator Peer
Body of the Workflow Process
Global Process Instance data
…
Process Description

```
<sequence>
  <flow>

    <Activity
     Name="Activity1" ...
       <ActivityInputData>
         DataName=Name1
         DataType=DataTYpe1
         DataLocation= URL1
      < /ActivityInputData>
        <ActivityOutputData>
         ....
       />
    <Activity
     Name="Activity2" .../>
…
  </flow>
  < Activity Name="Activity3" .../>
…
 </sequence>
```

As peers execute activities they may alter parts of the WPD document by, for example, updating information about the location of output data from an activity. As a general rule, peers can modify data that are associated with their current activity; everything else is considered to be outside their scope.

3.4. Workflow administration and peer notification

In traditional client–server workflow, process administration and execution is carried out by a centralised workflow engine. In contrast, in a peer workflow architecture administration is distributed amongst peers and therefore a Peer Notification Mechanism is required.

The Administrator Peer needs to know at any time what the state of each activity is. For this reason, WWPs need to be able to notify others when their activity is completed. Notifications need also involve ‘alerts’ such as for overdue-tasks.

The system provides the following administration facilities:

- Process instance report; the system is able to report the current state of a process instance. The Administrator achieves that by processing notifications messages, such as Completion of Activity, Reallocation of Activity etc.;
- Reallocation of activity, the Administrator is able to reallocate work from a peer to another one that is less busy;

- Cancel an activity, the administrator may ask a peer to cancel the execution of an activity;
- Reject an activity, a peer may reject the request to perform an activity;
- Deadline alerts, the administrator notifies the peers when deadlines have expired or some other conditions are met.

Notification messages are structured in XML documents. The basic elements of a Notification message are the Process ID, the Issuer, Date/Time, Description, Notification Type (e.g. Completion of Activity etc.), and a set of recipients.

The system supports the following types of Notification messages:

- *Completion of activity*: A peer that has just completed an activity notifies the administrator that the work has been completed successfully
- *Inability to perform activity*: This is a notification that the peer is unable to continue carrying out an activity he was assigned to;
- *Reallocation of activity*: The Administrator notifies both involved peers about the reallocation;
- *Cancellation of activity*: the Administrator notifies the affected peers that their activities are cancelled;
- *Deadline alert*: the Administrator alerts peers that a deadline is approaching;
- *Work overload alert*: a participating peer may notify the Administrator that work overload might cause him to miss the assigned deadline
- *Rejection of work*: A peer can notify the Administrator that he has rejected the work assigned to him.

3.5. User interface

WWP engines are lightweight components that can run on different machines on the network to give them the capability to participate in workflow. A WWP allows users to initiate new workflow processes, participate in workflow processes, and manage their assigned activities.

Human participants to the workflow can interact with WWPs through interfaces that display information about:

1. The *Work Items* folders:
   - Incoming work, a list of new activities (not yet viewed) assigned to the peer;
   - Pending work, the activities that are currently being processed;
   - Completed work, activities that have been completed;
   - Notifications received by the peer, e.g. deadline expirations etc.

Peers can view work grouped in different ways i.e. by Process, Initiator Peers etc.

The *Web Workflow Peer Profile* includes profile for available WWPs that could participate in the current workflow processes. Peers are grouped by type of activity/service, e.g. in the case of a healthcare scenario (described in the next section) specialist doctors, GPs, hospital services etc. This information is needed for a peer to decide what peers may execute the process. The WWPs profile is obtained from the WWP Directory.

The users may select and view an assigned activity, (placed in their *All Work Items/Incoming or Pending Work* folder) in order to start working on it. For instance, Fig. 2, shows a typical interface of a peer involved in a medical consultation process. Once the user finishes with the current activity, he has to decide which peers may continue with the process. The system provides a list of all peers available to execute the next activity. By clicking the 'select' button peer profiles are downloaded from the WWPD. Once the user selects a peer he can proceed by hitting the 'proceed' button. At this point, the WWP will forward the activity message and will also notify the process administrator peer.

4. Case study: a medical consultation process

The principles of the described P2P workflow architecture can be demonstrated using a medical consultation process. The process begins when a general practitioner (GP) examines one of his/her patients who has recently been experiencing some health problems. The GP ('Administrator Peer') assesses the patient, accesses his current and past medical history (maintained locally on the surgery systems), then examines the patient and makes a preliminary diagnosis, e.g. he suspects a blood disorder problem. The GP then may decide to refer the patient to a hospital for tests, such as blood tests, urine test, X-rays, etc. In the case of a blood disorder, the GP would take a blood sample from the patient and forward it to the hospital labs. As hospital labs are usually very busy and the test results might take several weeks to come back, the GP uses the services of the WWP directory to discover labs with spare capacity i.e. capable of carrying out the tests as soon as possible. The hospital labs (Peer 2) will process the samples and produce the results. The labs may decide to carry out more specialised tests with the given sample (or request additional from the patient through the GP), before returning the results to the patient's GP. For instance, if the results show that the patient is suffering from leukaemia, additional tests would be needed to show the type, e.g. acute lymphatic leukaemia, or chronic myeloid leukaemia.

When the results arrive at the surgery the GP assesses them and meets with the patient. The GP may either treat the patient locally by administering medication (i.e. in the case of deficiency anaemia) or refer him to a specialist such as haematologist (Peer 3) (i.e. in the case of a blood disorder) asking him to carry out a diagnosis. To locate

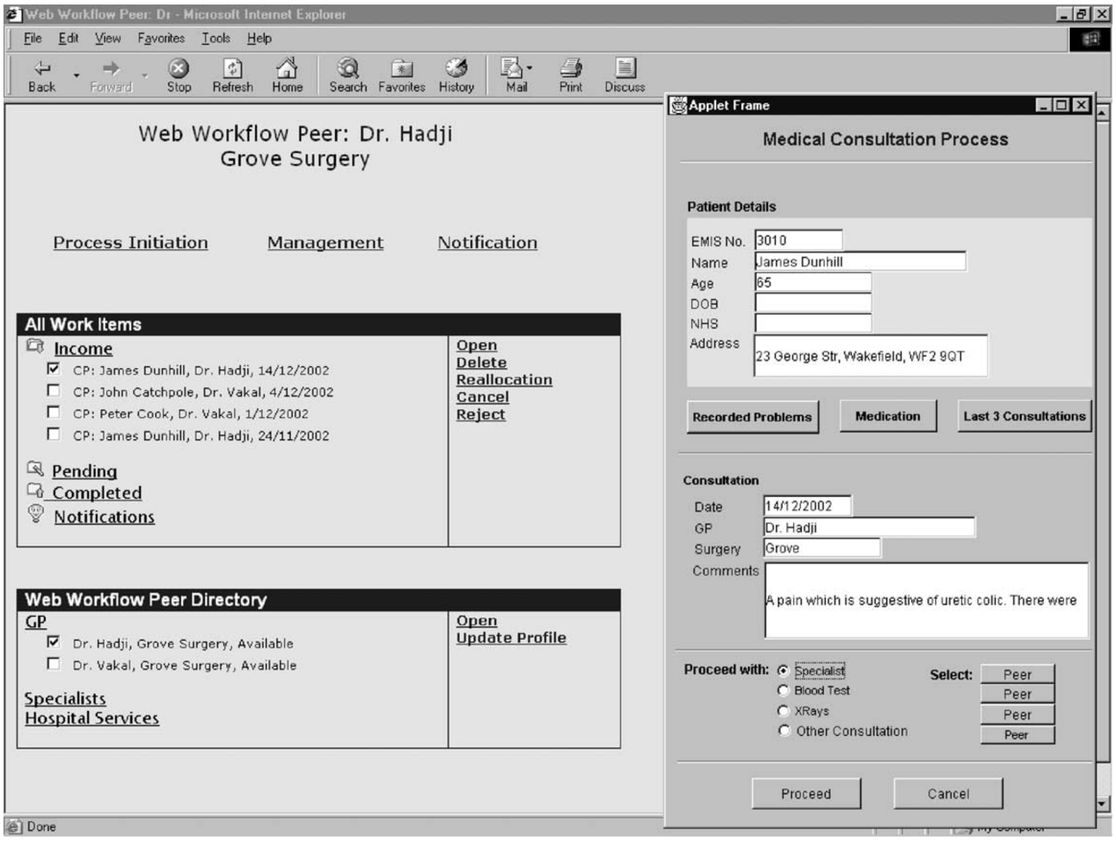

Fig. 2. The system user interface.

a haematologist the GP will again use the services of the WWP directory. The haematologist assesses the patient and his results and makes his/her diagnosis. The haematologist may further refer the patient to a more specialised unit such as chemotherapy (Peer 4). All specialists notify the predecessor specialists and the patient's GP with their diagnosis.

The above workflow process is shown diagrammatically in Fig. 3 as a peer collaboration diagram. The numbering of messages shows the order in which messages are being sent. When two messages share the same number it means that they are sent in parallel. The diagram does not show replies to messages nor data access information but it is assumed that peers access local data from other peers as and when they are required.

Compared to the conventional medical consultation process, the one described above has several advantages. The P2P-based medical consultation process is more efficient as it always allocates activities to peers (hospital labs, specialist doctors) with spare capacity. In this manner valuable time is saved which would be otherwise wasted waiting for medical results to come back from a busy lab or specialists. This in turn ensures a speedier treatment of patients. In addition the risk of medical records, test results and other data becoming misplaced or lost in transit is drastically reduced. By electronically accessing the data from the location of their origin there is no danger that such data become misplaced, lost or corrupted. This of course assumes a certain level of security being in place that includes access authorisation, encryption and so on.

5. Conclusions and further work

This paper presented a novel, web-based, P2P workflow management architecture. The main elements of the architecture, i.e. the Administrator and Participating Web Workflow Peers (WWPs), the WWP Directory, and the Workflow Process Description (WPD) Document were presented and a medical consultation case study was used to demonstrate the main principles of the approach.

P2P computing is currently expected to be a major revolution in computing, as big as the PC was in 1980s. This is due to the emergence of new technologies that could facilitate effective P2P, such as advances in computing power, network bandwidth, and storage capacity [1]. In addition, Web Services are becoming the main paradigm for distributed system implementation. Together, the two paradigms (P2P and Web Services) can provide the essential infrastructure for executing dynamic workflow processes over the Internet. Such workflows are

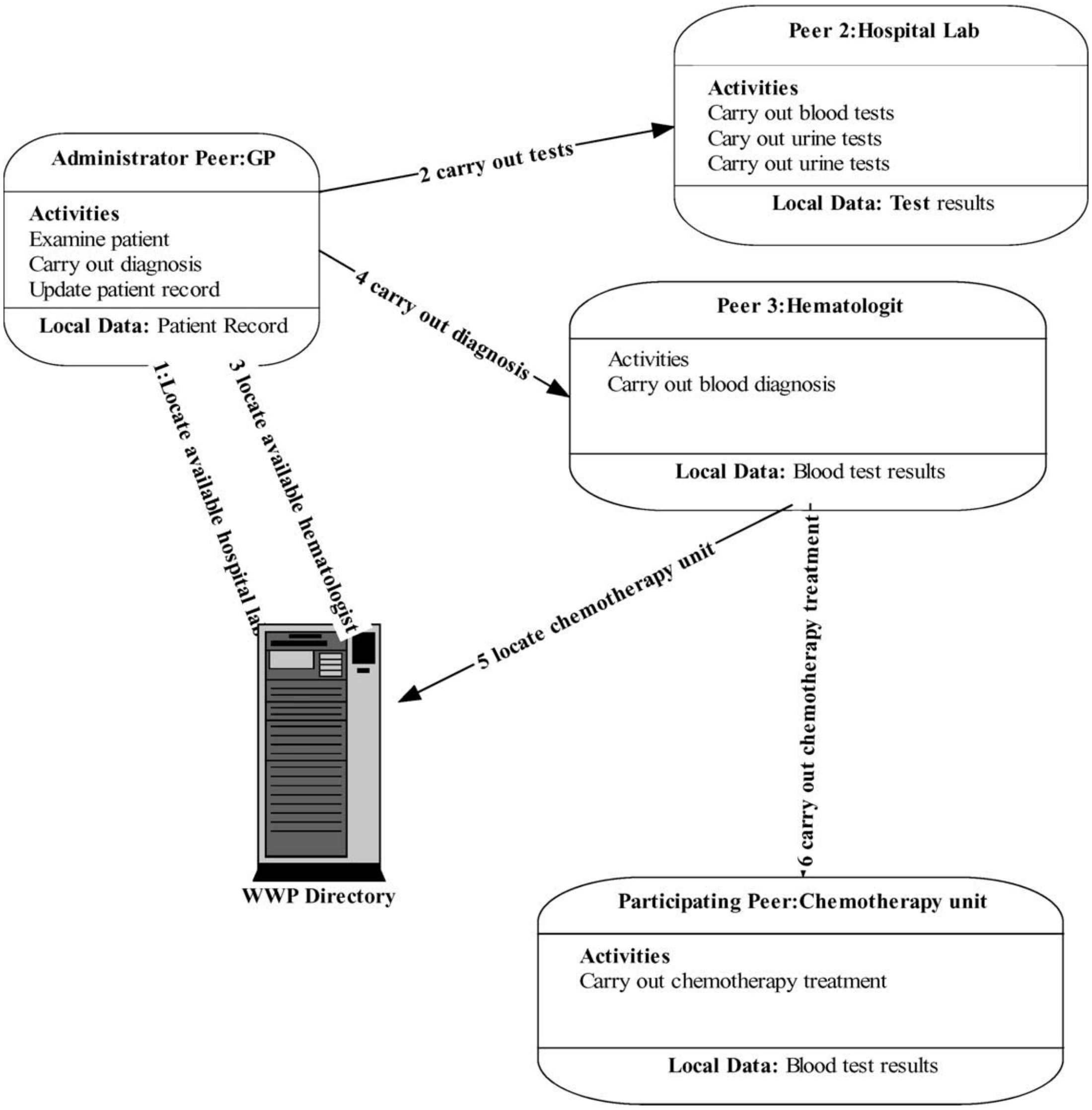

Fig. 3. The medical consultation process.

adaptive (the flow of work is dynamic and is controlled by the peers) and easily scaleable (i.e. by altering the number of participating peers) [6].

Another benefit of the proposed approach is its flexibility: in a conventional static workflow system bottlenecks can occur when one of the process participants becomes unavailable. In contrast, in the P2P approach, work is quickly re-assigned to one of the available peers.

The discussed medical case study where all peers are assumed to be known and trusted, and access to the WWP directory is controlled, does not pose significant problems compared to general business workflow applications where trust, security, integrity and confidentiality are going to be of high importance. Additionally, system performance will be an essential criterion for its commercial acceptability. In public P2P networks overall performance can suffer as a result of slow and/or unreliable connections to some peers, however we expect this not to happen to the same extent in commercial P2P networks where connections are assumed to be fast and reliable. Further research is therefore required in areas such as peer negotiation, quality of service, data encryption, security and integrity of the workflow transactions [15]. Such ongoing research includes XML document encryption to keep confidential information private and digital signatures to provide authenticity, integrity and non-repudiation.

Various competing technologies that could be used to realise the proposed architecture currently exist. One of them is the JXTA framework [10], which although still an early release, contains a lot of useful functionality. In addition, future versions of WfMC and W3C standards such as UDDI and process description languages such as BPEL4WS will influence future realisations of the architecture described here.

References

[1] S. Breidenbach, Feature: peer-to-peer potential, Network World Fusion July (2001).

[2] N. Coombs, Reliable recruitment and assembly of peer-to-peer services and distributed workflow, Working Conference on Complex and Dynamic Systems Architecture, Australia December (2001).

[3] G. Fakas, A. Kakas, C. Schizas, Electronic roads: intelligent navigation through multi-contextual information, Knowledge and Information Systems, International Journal, Springer March (2002) In press.

[4] G. Fakas, B. Karakostas, A workflow management system based on intelligent collaborative objects, Information Systems and Technology (41) (1999) 907–915.

[5] L. Fischer, The Workflow Handbook 2002, The Workflow Management Coalition, March 2002.

[6] G. Flammia, Peer-to-peer is not for everyone, Internet Services, IEEE Intelligent Systems May/June (2001) 78–79.

[7] J. Hayes, E. Peyrovian, S. Sarin, M. Schmidt, K. Swenson, R. Weber, Workflow interoperability standards for the Internet, IEEE Internet Computing May (2000) 37–45.

[8] IBM-BPEL, Business Process Execution Language for Web Services, Version 1.0, http://www-106.ibm.com/developerworks/webservices/library/ws-bpel/, 31 July 2002.

[9] IBM-WSFL, Web Services Flow Language (WSFL), IBM, 2001, http://www-3.ibm.com/software/solutions/webservices/pdf/WSFL.pdf.

[10] JXTA, The Project JXTA web site, http://www.jxta.org/.

[11] B. Karakostas, G. Fakas, Conceptual modelling of workflow using the intelligent business object paradigm, The Journal of Conceptual Modelling December (11) (1999).

[12] B. Karakostas, G. Fakas, Workflow over the Internet using business objects and XML, Innovation Through Electronic Commerce (IeC2000), Third International Conference Manchester, UK (2000).

[13] B. Karakostas, D. Panayiotakis, G. Fakas, Workflow requirements modelling using XML, Requirements Engineering Journal, Springer 7 (3) (2002) 124–138.

[14] K. Kilkki, Differentiated Services for the Internet, MacMillan Technologies Series, 1999.

[15] B. Masters, Peer-to-peer technologies and collaborative work management: the implication of Napster for document management, in: L. Fischer (Ed.), Workflow Handbook 2002, WfMC, 2002.

[16] C. Metz, IP QoS: travelling in first class on the Internet, IEEE Internet Computing April (1999).

[17] J. Miller, D. Palaniswami, A. Sheth, K. Kochut, H. Singh, WebWork: METEOR 2's web-based workflow management system, Journal of Intelligent Information Systems 10 (2) (1998) 185–215.

[18] MS-XLANG, S. Thatte, XLANG, Web Services for Business Process Design, Microsoft Corporation, 2001, http://www.gotdotnet.com/team/xml_wsspecs/xlang-c/default.htm.

[19] A. Oram, Peer-to-Peer, Harnessing the Power of Disruptive Technologies, O'Reilly, 2001, March.

[20] A. Oram, Consilient: Workflow Among Peers, O'Reilly, 2001, March.

[21] S. Paul, E. Park, J. Chaar, RainMan: A workflow system for the Internet, USENIX Symposium on Internet Technologies and Systems (1997).

[22] J. Schneider, Convergence of peer and Web services, O'Reilly's Open Source Convention July (2002).

[23] A. Tsalgatidou, T. Pilioura, An Overview of Standards and Related Technology in Web Services Distributed and Parallel Databases 12 (2002) 135–162.

[24] UDDI, The UDDI Web Site, http://www.uddi.org.

[25] WfMC, The Workflow Management Coalition Web site, http://www.wfmc.org.

[26] WfMC-WI, Workflow Standard—Interoperability Abstract Definition, The Workflow Management Coalition, 1996, www.wfmc.org/standards/docs/TC-1012_10_1996.pdf.

[27] WfMC-RM, The Workflow Reference Model, The Workflow Management Coalition, 1995, http://www.wfmc.org/standards/docs/tc003v11.pdf.

[28] WfMC-XPDL, The Workflow Process Definition Interface—XML Process Definition Language (XPDL), The Workflow Management Coalition, 2002, http://www.wfmc.org/standards/docs/xpdl_010522.pdf.

[29] WfMC-WfXML, The Interoperability Wf-XML Binding, The Workflow Management Coalition, 2001, http://www.wfmc.org/standards/docs/Wf-XML-11.pdf.

[30] W3C-SOAP, Simple Object Access Protocol (SOAP) 1.1, W3C (2000) http://www.w3.org/TR/SOAP/.

[31] W3C-WS, Web Services Activity, W3C (2002) http://www.w3.org/2002/ws/.

[32] W3C-WSDL, Web Services Description Language (WSDL), W3C (2001) http://www.w3.org/TR/wsdl.