IT University of Copenhagen

The Activity-Based Computing Project – A Software Architecture for Pervasive Computing

Final Report – The Strategic Research Council, grant no. #2106-04-0019

Jakob E. Bardram, professor, PhD

IT University Technical Report Series **TR-2010-136**

ISSN 1600–6100 **December 2010**

Copyright © 2010, Jakob E. Bardram, professor, PhD

IT University of Copenhagen
All rights reserved.

Reproduction of all or part of this work
is permitted for educational or research use
on condition that this copyright notice is
included in any copy.

ISSN 1600–6100

ISBN 978-87-7949-229-5

Copies may be obtained by contacting:

IT University of Copenhagen
Rued Langgaards Vej 7
DK-2300 Copenhagen S
Denmark

Telephone: +45 72 18 50 00
Telefax: +45 72 18 50 01
Web `www.itu.dk`

Contents

List of Figures

Executive Summary

This report describes the results of the Activity-Based Computing (ABC) project granted by the Danish Strategic Research Council, grant no. #2106-04-0019. In summary, we conclude that the ABC project has been highly successful. Not only has it meet all of its objectives and expected results, but have been able to pull additional resources and move beyond what was originally planned in the project.

From a research perspective, all of the original research objectives of the project have been met and published in 4 journal articles, 13 peer-reviewed conference papers[1], and two book chapters. Special attention should be drawn to publication [25], which gives an overview of the ABC project to the IEEE Pervasive Computing community; the ACM CHI 2006 [19] paper that documents the implementation of the ABC technology; and the ACM ToCHI paper [12], which is the main publication of the project, documenting all of the project's four objectives. All of these publication venues are top-tier journals and conferences within computer science.

From a business perspective, the project had the objective of incorporating relevant parts of the ABC technology into the products of Medical Insight, which has been done. Moreover, partly based on the research done in the ABC project, the company Cetrea A/S[2] has been founded, which incorporate ABC concepts and technologies in its products. The concepts of activity-based computing have also been researched in cooperation with IBM Research, and the ABC project has been widely disseminated to key stakeholder nationally and internationally.

Regarding research education, the original goal was to educate two PhD students as part of the project. Based on the ability to attract additional co-finansing from the University of Aarhus (AU) and the IT University of Copenhagen (ITU), in total of four PhD students have been associated with the ABC project. Three of these students have graduated, and now work at Oxford University and Cetrea A/S. The last student will graduate in 2011.

The project was delayed, partly due to 4-year PhDs instead of 3-years, and partly due to delays in recruitment. This delay has not had any impact on the results obtain; on the contrary. From a research management point-of-view, the project has learned us several lessons, which are being incorporated into the management of current research project at ITU.

The research on the ABC concepts and technologies have caused an interest in continuing to using it in further research. The "Collaborative Mini-Grid" project[3] has adopted the technology and is using it for biologists; the "Trustworthy Pervasive Healthcare Services (TrustCare)"[4] project is investigating how to combine ABC with workflow technologies; and there is an on-going research on ABC technology together with researchers at other universities As such, the ABC project has had a great impact on current and future research at the IT University of Copenhagen and other academic institutions.

[1] To the reader coming from a scientific community outside of computer science, it should be noted that peer-reviewed conference papers are the main venue of publication within computer science, and is often much more competitive than journal publication.

[2] http://www.cetrea.com/

[3] Research grant no. #09-061856 from the Danish Strategic Research Council

[4] Research grant no. #2106-07-0019 from the Danish Strategic Research Council

Chapter 1

Scientific Results

This chapter presents the scientific results of the ABC project in terms of meeting the main objectives, obtained results, and publications.

1.1 Research Objectives and Results

The original project proposal stated 4 research objectives, namely to:

1. develop a conceptual framework for understanding ABC
2. develop a set of design principles and architectural principles for ABC
3. design and implement a proof-of-concept in term of research prototypes
4. evaluate activity-based computing as a software architecture for hospital work

As we shall outline in the following four subsections, all of these objectives have been meet and has been documented both in technical reports as well as in a wide range of publications.

1.1.1 Objective I – Conceptual Framework

The conceptual framework for ABC obviously pivots around the concept of an 'Activity'. Figure 1.1 is a conceptual illustration of an *activity* which is a work-related aggregation of *services* and *data*. We have defined activity-based computing around the following essential principles:

Activity-Centered – A 'Computational Activity' collects in a coherent set a range of services and data needed to support a user carrying out some kind of (work) activity. This principle moves computational support beyond contemporary application-centered desktop computing.

Activity Suspend and Resume – A user participates in several activities and he or she can alternate between these by suspending one activity and resuming another. Resuming an activity will bring forth all the services and data which are part of the user's activity. This principle introduces support for interruptions.

Activity Roaming – An activity is stored in an infrastructure (e.g. a server) and can be distributed across a network. Hence, an activity can be suspended on one work-station and resumed on another in a different place. This principle introduces support for mobility.

Activity Adaptation – An activity adapts to the resources available on the device (i.e. computer) on which it is resumed. Such resources are e.g. the network bandwidth, CPU, or display on a given devices. This principle helps move beyond the contemporary computational model of isolated and homogeneous devices.

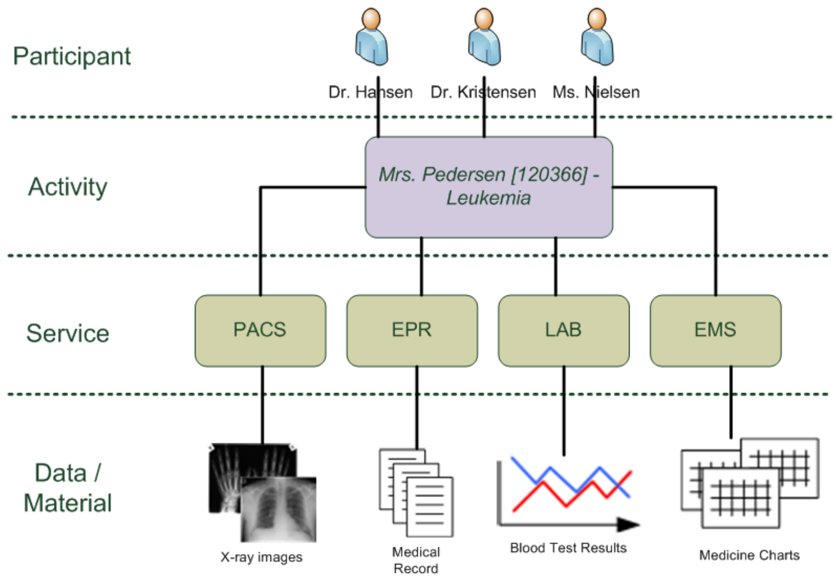


Figure 1.1: A computational activity that aggregates a set of computational services, data resources, and users.

Activity Sharing – An activity is shared among collaborating users. It has a list of participants who can access and manipulate the activity. Consequently, all participants of an activity can resume it and continue the work of another user. Furthermore, if two or more users resume the same activity at the same time on different devices, they will be notified and if their devices support it, they will engage in an on-line, real-time desktop conference. This principle introduces support for collaboration.

Context-awareness – An activity is context-aware, i.e. it is able to adapt and adjust itself according to its usage context. Context-awareness can be used for adapting the user interface according to the user's current work situation – e.g. by showing medical data for the patient currently being treated.

This activity-based computing conceptual framework has proved stable during the whole project period; both in terms of the number of principles as well as their formulation. During the 5 years of working closely with clinicians in different hospitals, these six principles has proved sufficient for creating pervasive computing support for clinical work in hospitals. From a conceptual point-of-view, this is strong result of the ABC project. Details on this conceptual framework and the six ABC principles can be found in [25, 12, 19].

1.1.2 Objective II – Design and Architectural Principles

The *ABC Framework* represents the architectural design of the ABC principles described above. The main goal of the ABC framework is to provide a platform for the *development* and *deployment* of computer applications that support the principles of activity-based computing. The framework consists of three parts: (i) a *runtime infrastructure*, which handles the computational complexities of managing distributed and collaborative activities by adapting to the available services or resources in a specific environment; (ii) a *user-interface* that supports users in activity management according to the five activity-based computing principles discussed above; and (iii) a *developer's framework (API)*, which allow programmers to create ABC-aware applications that can be deployed in the runtime infrastructure.

Figure 1.2 illustrates the ABC architecture. It consists of a range of processes running as part of the underlying infrastructure, a set of client-layer processes (i.e. processes at the level of the particular device) responsible for activity management on that device, and a set of ABC-enabled applications with a user interface in the application layer. The technical details of the ABC Framework have been described in various publications [22, 7, 18].

The ABC Framework also provides the design of the core ABC user-interface software components, such as an

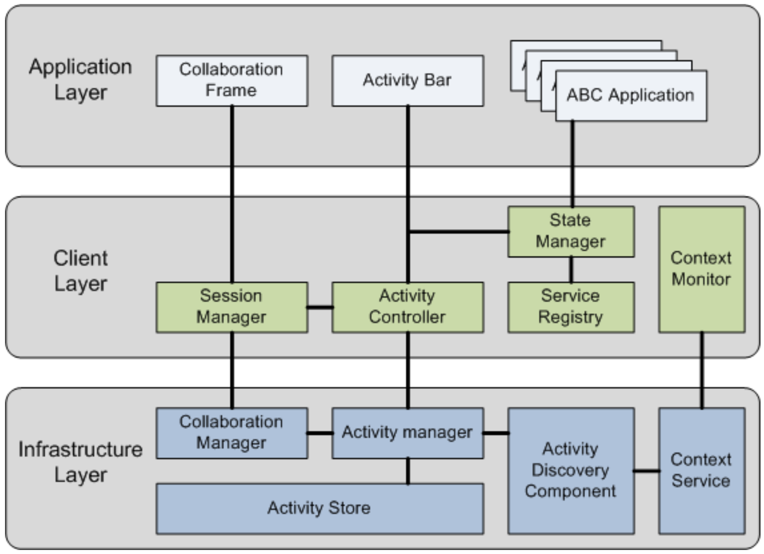


Figure 1.2: The ABC Runtime Infrastructure illustrating both server-side and client-side processes.

'ActivityBar', a 'CollaborationWindow', and other ABC widgets. These user interface components have been designed and implemented in two main versions as part of this project; version 3 and 4.

Version 3 is aimed at the use for medical applications in a hospital setting. Figure 1.3 shows the design of the ABC user interface. This design is targeted for shared interactive displays within a hospital, such as wall-mounted touch screens, tablet PCs, and bedside displays. The interface takes up the whole screen real-estate. Thus the ABC client is not an application but is viewed as the only interface to the device. The user interface from the underlying operating system is hidden, including e.g. the Windows Taskbar. This is intended to emphasize that the primary entry point to the computer is through activities rather than through applications or files. Figure 1.3 shows the main UI components of the ABC client: the *Activity Bar* for accessing and managing activities, the *Collaboration Window* showing a list of participants in the activity, the *Activity Chat* used for instant messaging inside an activity, the *Activity Menu* containing a list of all ABC-aware applications available on the given device, and the *Telepointers*, which are used during synchronous activity sharing. In addition, figure 1.3 shows a range of *ABC-aware Applications*, such as the EPR patient toolbar, which lists all patients, the medicine chart, which shows the medicine prescribed for the selected patient, the medicine ordination wizard, and the x-ray viewer for showing radiology images.

Version 4 of the ABC Framework was targeted 'normal' personal computing usage, in the sense that we wanted to extend the contemporary support for application- and desktop-based interaction with activity-based computing. For this reason, we extended the Windows XP operating system with activity-based computing support, using the ABC Framework. Compared to the previous Java-based version, version 4 is a Microsoft .NET-based implementation, which had two main goals: to create a platform and language independent architecture, and to create an ABC user interface client which is tightly integrated into the Windows XP operating system. Version 4 is documented in the ACM CHI 2006 paper [19].

The ABC user interface for Windows XP is shown is figure 1.4. In Windows XP, each service is mapped to an application window. Thus, an activity can be made up of a range of windows, including child windows to an application, where the main window is not part of the activity. In figure 1.4 the activity labeled 'ABC CHI Paper' is resumed and contains windows from different applications like Adobe Reader, Firefox, and an open mail in a child window from Thunderbird. The main user interface component is the Activity Bar illustrated in figure 1.4. This bar replaces the Windows XP Taskbar since activities – and not applications – are the main focus in ABC. In order to facilitate an intuitive understanding of how the bar works, the activity bar is deliberately designed to resemble the Windows Taskbar. The 'Activities' button is used to list the current user's activities. The action buttons are used to: (i) Create a new activity; (ii) suspend the current activity; (iii) invite other participants; (iv) save the activity locally; (v) zoom out the activity; (vi) show the ABC control panel; and (vii) to show the radar view. Frequently used activities

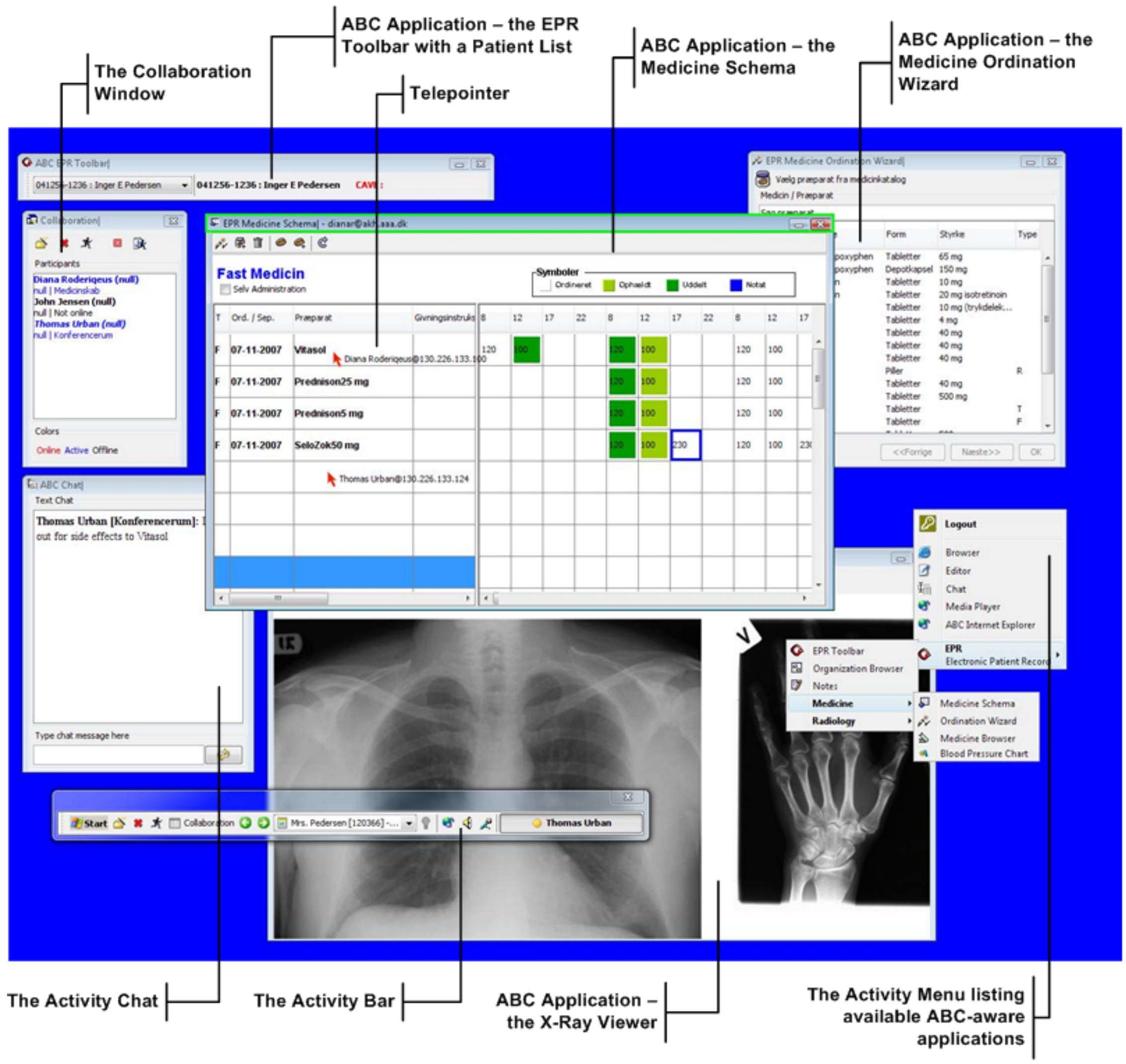


Figure 1.3: Version 3 of the ABC User-interface with (i) the Activity Bar, (ii) the Collaboration Window showing all participants in this activity, (iii) the Activity Chat for chatting inside an activity, (iv) four ABC Applications which implement the 'EPR Toolbar', the 'Medicine Chart', the 'Medicine Ordination Wizard', and the 'Radiology Image' services, (iv) the Activity Menu, and (v) a Telepointer. This picture shows a typical activity in the medical treatment of a patient (Mrs. Pedersen) by three participating clinicians.

are shown in the middle part of the bar, and the status icons on the left reveal the collaborative status for the current user: (i) Other online participants; (ii) tele-pointers on/off; (iii) voice-link on/off; (iv) and server online status.

1.1.3 Objective III – Implementations

The ABC Framework has been implemented in 4 different versions as part of this project, and a 5th version is currently being developed for use in other projects (se more on the future use of the ABC research in section 6). Table 1.1 provides an overview of the different versions.

Version 1 and 2 were preliminary prototypes that actually was done prior to the funding of this project, and as such actually formed the background for the original project proposal. These two version were focused on initial exploration

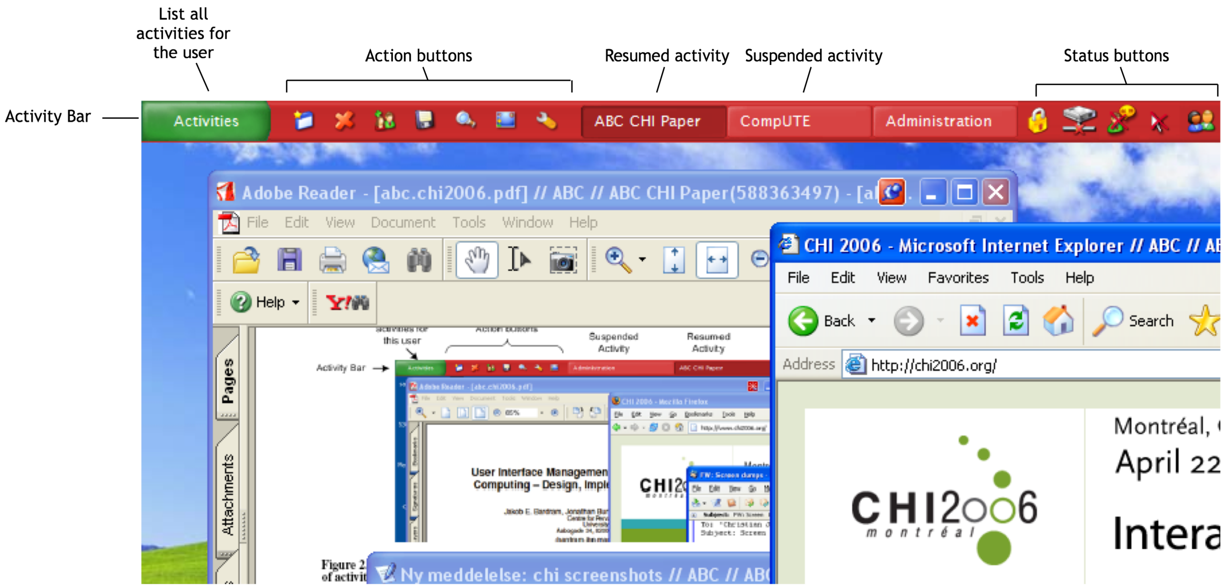


Figure 1.4: An overall view of the ABC user interface for Windows XP, including the Activity Bar, the current user's list of activities, and different application windows that are part of the resumed activity labeled 'ABC CHI Paper'.

of the principle of context-awareness and activity roaming. As explained above, the two main implementation as part of this project has been version 3 and 4.

Version 3 is build entirely using Java technology. The server is implemented using J2SE; the client using Java Swing; the distribution is using Java RMI; and the programming API is Java-based. Version 3 was a proof-of-concept implementation of all the six ABC principles, with a special emphasis on support for activity sharing.

Version 4 was implemented by reusing the Java-based server from version 3, but shifting to MS .NET technology for the client. Focus was on integrating the ABC technology in the Windows XP operating system. In order to support cross-platform communication, the special-purpose, stateless ABC communication protocol over TCP/IP (ABCP) and Activity Markup Language (AML) was designed and implemented. In effect we implemented what today corresponds to a RESTful communication protocol. In the client-side implementation, special-purpose hooks into the Windows OS enabled us to tap into window and file management on the clients.

1.1.4 Objective IV – Evaluations

Both version 3 and 4 of the ABC Framework have been evaluated. Based on their different foci, version 3 in a hospital setup and version 4 in an office environment. Version 3 – focusing on ABC support for hospital work – was designed by creating a suite of clinical applications for hospital use and then inviting a set of clinicians to evaluate the resulting setup. Thus the focus was on end-user (clinical) evaluation of the runtime infrastructure, the ABC-aware clinical applications, and the ABC user-interface, including the activity bar and the collaboration window. The ABC Framework was designed in close cooperation with a set of clinicians from a hematology department. After the period of co-design and evaluation, this group of clinicians felt confident about the end result, and we then conducted four whole-day evaluation workshops with a group of clinicians who had never seen the ABC Framework before or been introduced to the concept of activity-based computing. This group of 3 physicians and 3 nurses worked in the department of plastic surgery in another hospital. In these workshops, clinicians were asked to role-play a number of clinical scenarios in which they would use the ABC Framework and its clinical applications. Figure 1.5 and 1.6 show pictures from the workshops.

The overall findings from our experiments are that the principles of activity-based computing, and their implementation in version 3 of the ABC Framework, have proved rather robust throughout the design and evaluation workshops.

Ver.	Impl.	Dist.	Focus and Features	Pub.
1	Java	N/A	Activity Discovery.	[13, 20]
			Activity as an event.	[35]
2	Java	Java RMI	Activity as an object.	[33]
			Client/Server, management, storage.	
			Activity as a singleton object.	
3	Java	Java RMI	Activity Sharing	[7, 12]
			Activity as col. session mgmt. on the server.	[25]
			New ABC client (bar, telepointers, voicelink)	[22]
			Java-based API for development of ABC collaborative applications.	
4	Java	AML/	Integration with Windows XP	[10, 19]
	.NET	ABCP	Robust vrt. network failure / works disconnected	[32]
			Client/Server architecture	[18]
			Reuse of v. 3 server from; new XML-based ABC Protocol (ABCP)	
			Wraps native Win XP applications	
			New "Activity Bar" to replace the Windows Taskbar.	
			New ABC User Interface for zoom, collaboration, etc.	
			C#-based API for development of ABC applications.	
5	Java	P2P	Extended Activity Model	[16]
	Air/Flex	REST	Robust event-based Peer-to-Peer infrastructure, discovery.	
			Novel interaction design and UI supporting non-desktop devices.	
			Adaptation of activities based on the usage context.	
			Discovery of services/devices that are relevant for an activity.	
			Support for temporal workflows of activities.	

Table 1.1: The different versions of the Activity-Based Computing infrastructure. Ver. = Version; Impl. = Underlying implementation technology; Dist. = Underlying distribution technology; Pub. = Core publications.

The ABC system unifies computational support for handling multiple applications running on different client machines; it supports mobility through activity-roaming between devices; it supports advanced management of parallel work tasks and interruptions; it directly supports both asynchronous and synchronous cooperation, and it makes it possible to link physical artifacts with digital material. This evaluation and these results are further detailed in [12].

Version 4 – focusing on ABC support for desktop computing – was evaluated by investigating whether support for activity-based computing can be part of a contemporary OS, like XP. This raise the following usability questions:

- **Usefulness** – Our primary concern was to establish whether the whole idea of activity-based computing as embedded in the Windows XP operating system (OS) is useful, i.e. can ABC be part of a contemporary OS? Can activities exist in an OS or does it clash with the desktop metaphor? Is ABC useful in other, non-clinical, domains?

- **Ease-of-Use** – Our secondary concern was to establish if the ABC user interface for XP was easy to use. Our goal was that experienced Windows XP users would be able to use the ABC extended version with no prior training.

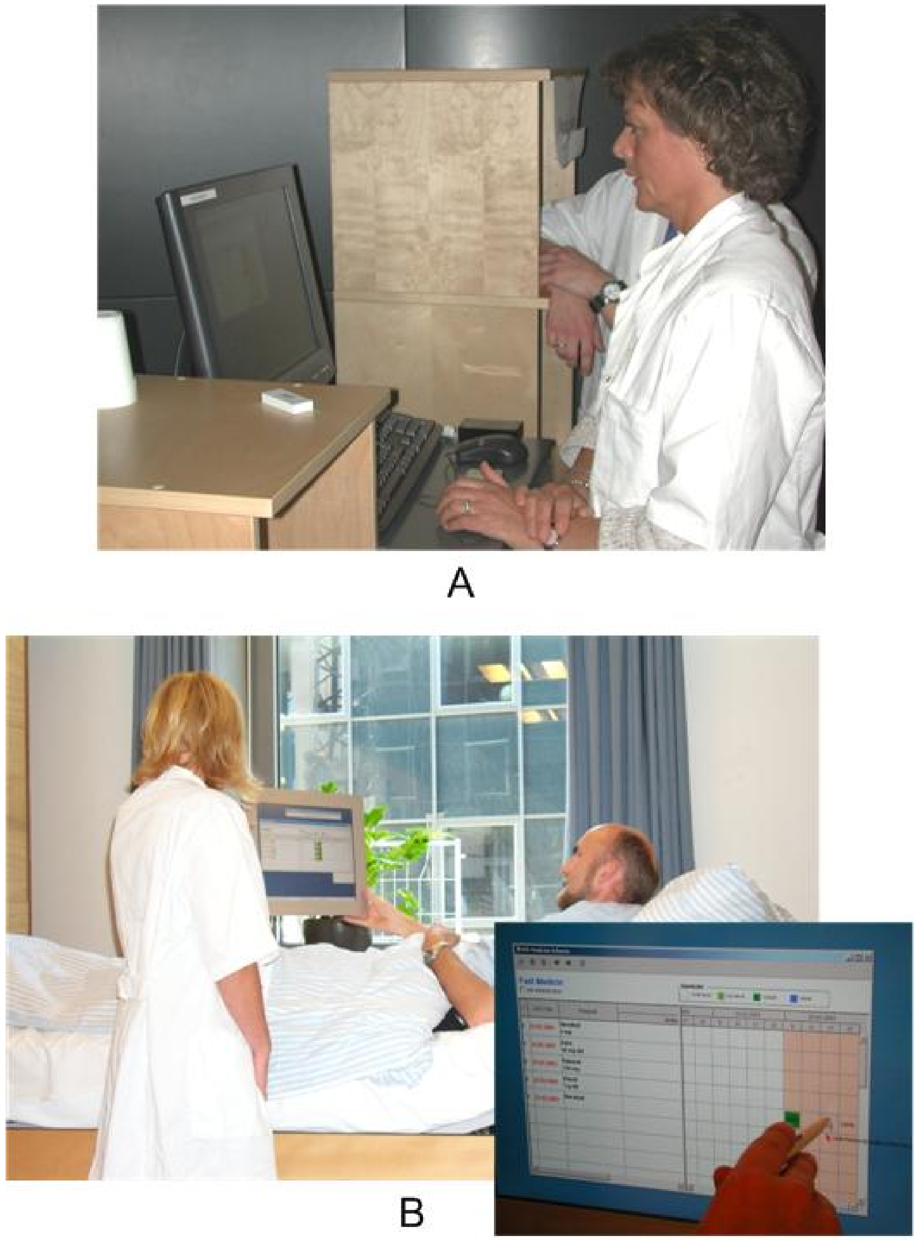


Figure 1.5: Evaluation of 'Activity Roaming' scenarios. **A** : The nurse is standing in the 'Medicine Room' using the computer to prepare the medicine. **B** : Another participant in the same activity uses the display attached to the hospital bed to activate the activity and document the patient's medicine intake (picture inserted).

The goal was to provide objective measurements on the usefulness and usability of our design while, at the same time, investigating the underlying detailed user reaction to the ABC user interface in a more qualitative fashion. The test was set up to simulate well-known tasks for the participants, in this case a programming assignment for a company. This evaluation revealed that users found the user interface easy to use and that activity-based computing support would be useful for them in their work. Due to the multi-method approach, we were also able to establish more precisely what part of the user interface that needed to be improved. These results are being used in our current enhancement

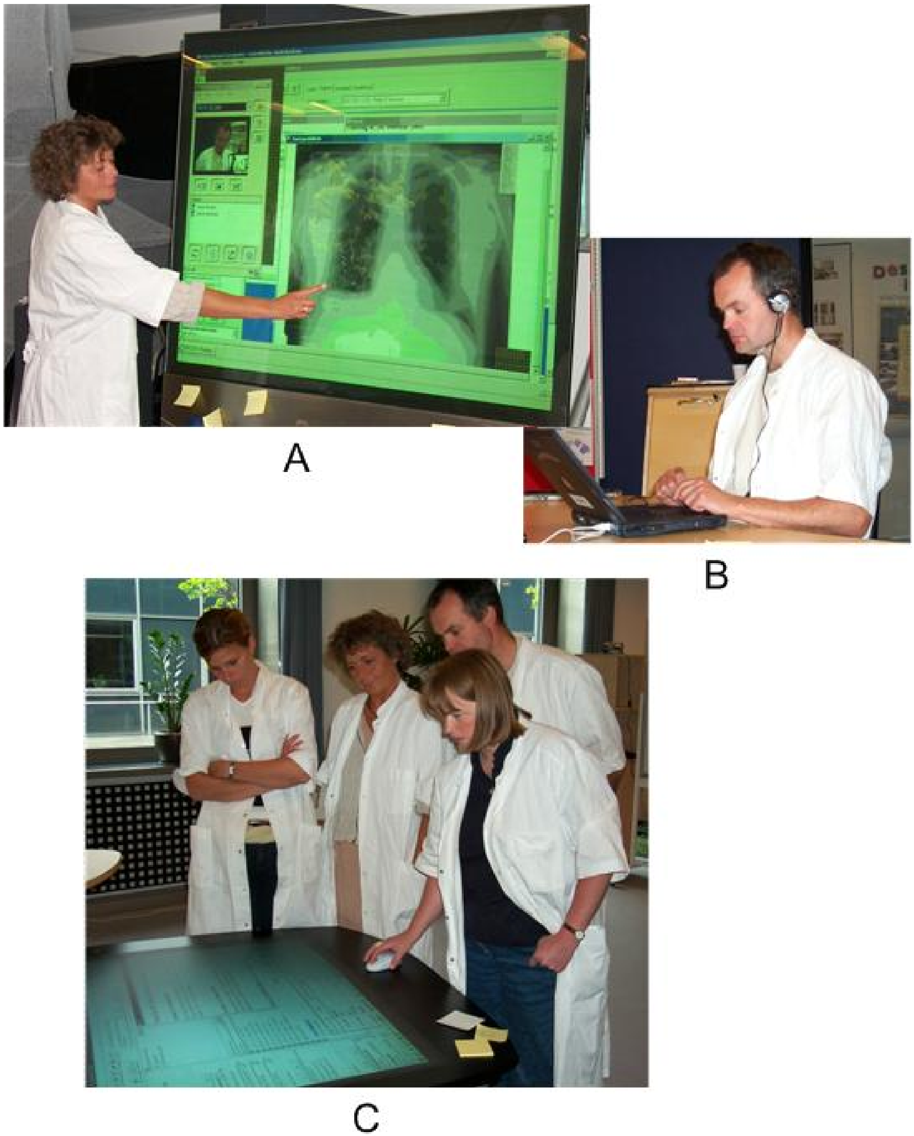


Figure 1.6: Evaluation of ‘Activity Sharing’ scenarios. **A+B** : A nurse and a physician are engaged in a medical conference using synchronous activity- sharing. The nurse is standing in front of the wall-based display in the ‘Team Room’. The physician is in his ‘Office’, which is a separate room. **C** : While still running on the wall-display, the activity is also resumed on the interactive conference table, so that the table and the wall-based display can be used simultaneously.

of the system, which also targets the support for roaming files together with activities. Hence, files belonging to an activity would move between computers as the user roams. The details of this evaluation is published in [19].

1.2 Publications

The scientific work in the ABC project has been published in 4 journal articles, 13 peer-reviewed conference papers, 2 book chapters, and a number of technical reports. All publications related to the ABC project are listed in the bibliography in the end of this report. To the reader coming from a scientific community outside of computer science, it should be noted that peer-reviewed conference papers are the main venue of publication within computer science, and is often much more competitive than journal publication.

Special attention should be drawn to the following publications:

[33] Henrik B. Christensen and Jakob E. Bardram. Supporting Human Activities - Exploring Activity-Centered Computing. In Gaetano Borriello and Lars Erik Holmquist, editors, *Proceedings of Ubicomp 2002: Ubiquitous Computing*, volume 2498 of *Lecture Notes in Computer Science*, pages 107–116, Gothenborg, Sweden, September 2002. Springer Verlag.
This paper was the initial description of the Activity-Based Computing idea, its use in hospitals and the outline of an architecture for the ABC Framework. It was published and presented at the Ubiquitous Computing (Ubicomp) conference, which is the leading conference within ubiquitous and pervasive computing and has an acceptance rate below 20%. Ubicomp is now an ACM sponsored conference. This paper is still one of the most cited ABC paper (115 citations).

[22] Jakob E. Bardram and Henrik B. Christensen. Real-time Collaboration in Activity-based Architectures. In *Proceedings of Fourth Working IEEE/IFIP Conference on Software Architecture (WICSA'04)*, pages 325–329. IEEE Press, 2004.
This paper presents the underlying distributed infrastructure in the ABC Framework, which enables real-time collaboration. The IEEE WICSA conference is the premier conference with software architecture and has an acceptance rate on ca. 25%.

[12] Jakob E. Bardram. Activity-based computing for medical work in hospitals. *ACM Transaction on Computer.-Human Interaction*, 16(2):1–36, 2009.
This journal article can be considered the main ABC publication. As outlined above this article documents the motivation behind the ABC approach, the ABC conceptual framework, the user-interface design of the clinical version 3, and presents the in-depth evaluation of the technology and the results of it. In time, this article will – hopefully – stand as the main reference back to the ABC project for hospital work. ACM Transaction on Human-Computer Interaction (ToCHI) is the premier journal within human-computer interaction and is highly competitive.

[19] Jakob E. Bardram, Jonathan Bunde-Pedersen, and Mads Soegaard. Support for activity-based computing in a personal computing operating system. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 211–220, New York, NY, USA, 2006. ACM Press.
This paper is the main publication for the personal computer (Windows XP) version of the ABC Framework and documents the design of this user-interface software technology as well as a study of its use by programmers in an office setup. The ACM conference on Human-Computer Interaction (CHI) is the premier conference within Human-Computer Interaction and is with its acceptance rate below 25% highly competitive. This paper is one of the most references ABC papers in the CHI community.

[10] Jakob E. Bardram. From Desktop Task Management to Ubiquitous Activity-Based Computing. In Victor Kaptelinin and Mary Czerwinski, editors, *Integrated Digital Work Environments: Beyond the Desktop Metaphor*, pages 49–78. MIT Press, 2007.
This edited book contains descriptions of what is considered to be the best ideas for computing metaphors, which in the future will replace the current desktop metaphor used in personal computers. ABC was chosen to be one of these future metaphors among seven others, and this proves that activity-based computing as a concept plays a central role in contemporary computing research.

[16] Jakob E. Bardram, Jonathan Bunde-Pedersen, Afsaneh Doryab, and Steffen Sørensen. Clinical Surfaces – Activity-Based Computing for Distributed Multi-Display Environments in Hospitals. In Tom Gross, Jan Gulliksen, Paula Kotzé, Lars Oestreicher, Philippe Palanque, Raquel Oliveira Prates, and Marco Winckler, editors, *Human-Computer Interaction – INTERACT 2009*, volume 5727 of *Lecture Notes in Computer Science*, pages 704–717. Springer, 2009.
This paper documents our initial design of version 5 of the ABC Framework and user-interface software technology. This version is radically different from the previous ones both in terms of the underlying infrastructure

(which has completely been re-implemented) an in terms of the user experience. This paper describes how a hospital can be viewed as one large distributed display surface for all clinicians to use, and how the ABC Framework (v. 5) can support such a vision.

[13] Jakob E. Bardram. A novel approach for creating activity-aware applications in a hospital environment. In Tom Gross, Jan Gulliksen, Paula Kotzé, Lars Oestreicher, Philippe Palanque, Raquel Oliveira Prates, and Marco Winckler, editors, *Human-Computer Interaction – INTERACT 2009*, volume 5727 of *Lecture Notes in Computer Science*, pages 731–744. Springer, 2009.
This paper is a highly conceptual paper and builds on an on-going conceptual understanding of the notion of Activity and its relation to other core concepts within Ubicomp, such as context-aware and sensor-based computing. The paper was *awarded best paper* at the INTERACT conference in 2009.

[25] Jakob E. Bardram and Henrik B. Christensen. Pervasive Computing Support for Hospitals: An Overview of the Activity-Based Computing Project. *IEEE Pervasive Computing*, 6(1):44–51, 2007.
As the title indicates, this article provides an overview of the ABC project, its motivation, its technology, and its results. This article is published in the IEEE Pervasive Computing magazine, which is the premier journal-type of publication venue for ubiquitous computing research. As such, this paper makes the ABC project highly visible in it's 'home' scientific community.

Chapter 2

Business and Societal Results

This chapter describes the obtained business and societal results of the ABC project.

2.1 Business Results

The original business goal of the ABC project was to;

- incorporated relevant parts of the ABC technology into the products of Medical Insight.

But since more companies that Medical Insight has been involved, it is relevant to consider all of these, and how parts of the ABC technology has been incorporated into their products.

2.1.1 Medical Insight A/S

The EasyViz PACS[1] system by Medical Insight has been integrated into version 4 of the ABC framework as illustrated in Figure 2.1. This integration was done a proof-of-concept demonstrating that is possible to use the ABC programming API to integrate a 3rd party clinical application like EasyViz Enterprise into the ABC Framework. A simple technical interface was made by Medical Insight allowing the ABC Framework to control which patient and examination session to show in EasyViz Enterprise. In this way, when an activity was resumed in the ABC Framework, which involved a radiology exam, the ABC Framework could open EasyViz Enterprise showing the relevant images for this patient. Later, EasyViz was deployed inside an operating room at Horsens Sygehus, as described in section 4.2 and shown in Figure 4.1.

Medical Insight has used this project for testing new technology outside the normal product development cycle. Concepts and technology design from the ABC project has been incorporated into Medical Insight's enterprise product, and is now used for hosting Multi-Disciplinary Team (MDT) meetings in a project in the North-East and Eastern NHS[2] clusters in the UK. MDT meetings are used to bring experts and specialists together to create treatment plans for difficult patient cases. The concept of 'Activity' and technology design from the ABC project is being used in the EasyViz Enterprise product to enable MDT meeting organizers to prepare cases for the meeting as separate activities. When a meeting is started at a later point in time, the organizer can simply bring back a patient case in the prepared state by resuming the saved activity. This saves valuable specialist time as the organizer does not have to manually find and load relevant prior radiology studies and patient data for each case. Furthermore, the shared session functionality of EasyViz Enterprise allows the MDT meetings to be hosted as a virtual meeting, using the concept of 'Activity Sharing'.

[1] Picture, Archiving, and Communication System
[2] The National Health Service

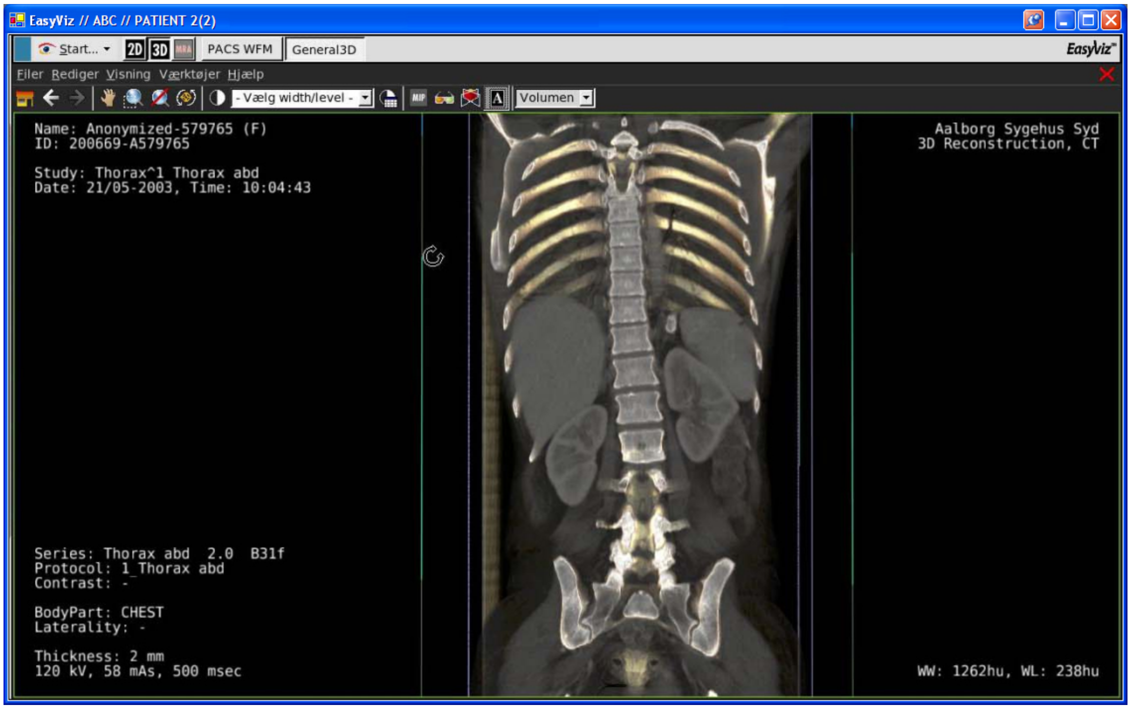


Figure 2.1: The Medical Insight PACS system called EasyViz running as an integrated part of the ABC Framework version in the Windows XP operating system.

2.1.2 Acure /IBM Denmark

Together with Acure / IBM Denmark[3] we integrated the ABC Framework (version 4) with the Acure Eletronic Medical Record (EMR). A simple extension to this EMR made is interoperable with the ABC Framework in a similar way as EasyViz; when resuming an activity in the ABC Framework pointing to a specific patient, then relevant data for this patient is retrieved and shown in the Acure EMR. The result for the doctor using the system is that when he is turning his attention to this patient (by resuming the relevant activity), then the patient data both in the EMR and in the PACS system is shown. This use experience is illustrated in Figure 2.2.

2.1.3 Cetrea A/S

The company Cetrea A/S[4] was founded by researchers involved in the ABC project. Even though the products of Cetrea is a result of another research project[5], the core technology underlying the Cetrea product suite builds directly on research results obtained as part of the ABC project. For example, the distribution mechanism in the Cetrea infrastructure is similar to the ABCP protocol, and uses both client-server communication as well as peer-to-peer communication – the so-called 'hybrid architecture' which was published as part of the ABC project [18].

Moreover, the concept of 'Activity' exists in the Cetrea underlying infrastructure and is used to help clinicians keep track of the unfolding of different clinical activities, which are important to them. For example, an activity can track the status of a blood measurement from it is taken, analyzed, and the result is ready for a doctor to use. This use of the concept of Activity is not exactly as defined in the ABC project, but is clearly inspired from our research.

[3] http://www.acure.dk/
[4] http://www.cetrea.com/
[5] The "Interactive Hospital" project – see http://www.ihospital.dk/

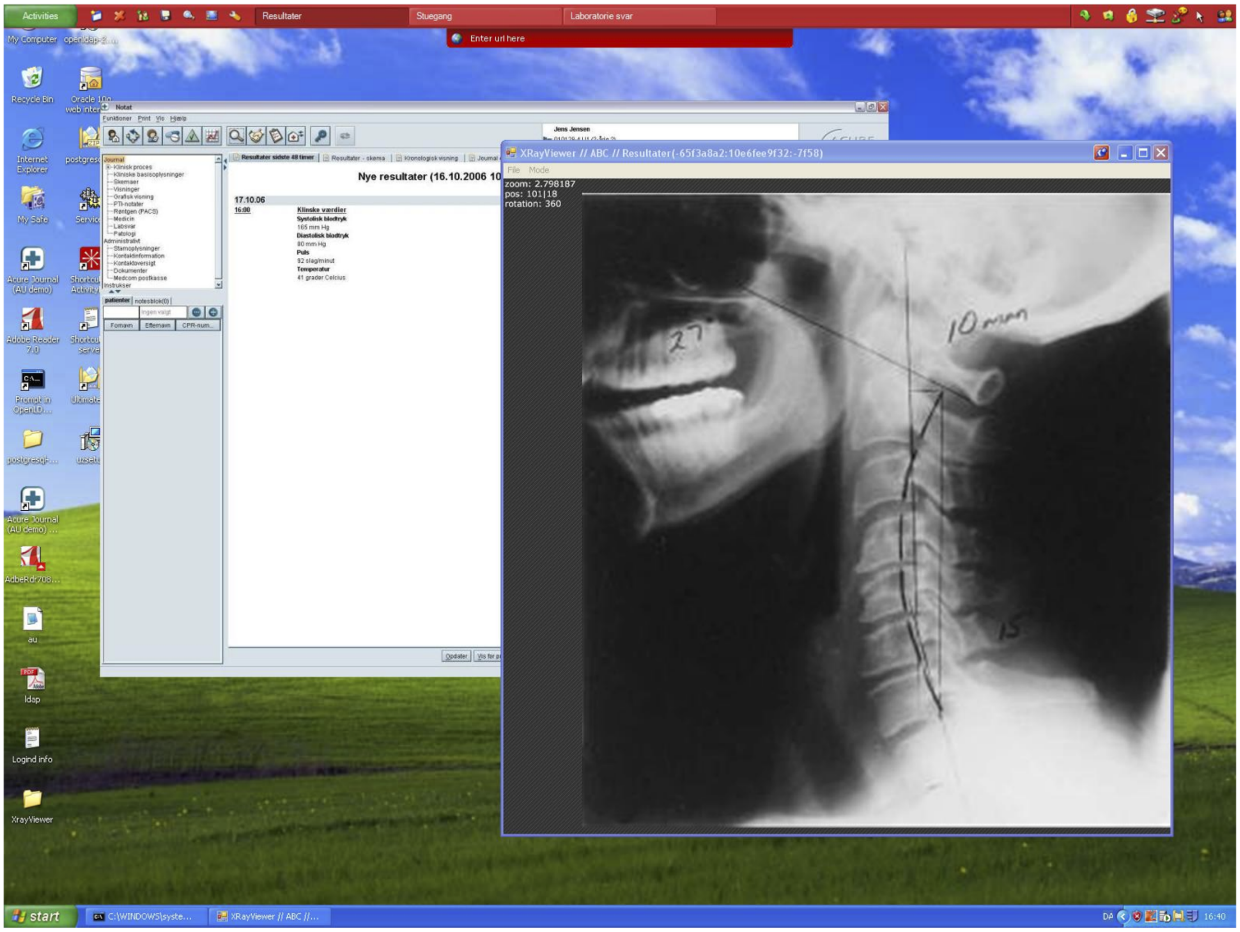


Figure 2.2: The Acure EMR system (in the background) running as an integrated part of the ABC Framework together with a PACS system showing a radiology image. Both application shows patient date that is relevant to the resumed activity (shown in the Activity Bar in the top).

2.2 Societal Impact

The ABC project has been widely disseminated at important venues. Here is a list of the most influential ones.

- In 2005, the ABC idea and the technologies were presented to the present Danish Minister of Health – **Lars Løkke Rasmussen**.
- In 2005, we co-organized the *Workshop on Activity - From a Theoretical to a Computational Construct* at the 2005 ECSCW conference in Paris, France, together with **Tom Moran**, IBM Research.
- In 2006, the paper on Activity-Based Computing for a Personal Operating System [19] was presented at the **ACM CHI 2006 conference**.
- In june 2007, an article about the use of pervasive computing technology in hospitals were printed by the **Danish Medical Association** (Ugeskrift for læger 2007;169(26):2502).
- In July 2007, the ABC project and its results are presented as an invited talk on the New Paradigms for Using Computers (NPUC) conference at the **IBM Research center at Almaden** in Silicon Valley. Following this talk,

Jakob Bardam stayed for one month to work with researchers at IBM Research on activity-based computing. The IBM Lotus Anywhere technology now incorporates the concept of an activity.

- In 2009, the ABC project and the technologies were presented to the Chief Information Officer (CIO) of **Kaiser Permanente**, California.
- In April 2010, the ToCHI paper on Activity-Based Computing [12] was presented at the **ACM CHI 2010 conference**.
- In 2010, the ABC project was presented at an invited talk at the **Danish Symposium for Electronic Health (eHealth)** in Nyborg. This conference is the premier venue for Danish researchers, industry people, and clinicians with electronic healthcare.

Chapter 3

Research Education

The orginal project application applied for two PhD scholarships, but due to the success of attracting additional resource from the University of Aarhus and the IT University of Copenhagen, a total of four PhD students (two male and two female) has been associated to the project, of which three has graduated. The students are;

Jonathan Bunde-Pedersen – *Distributed Interaction for Activity-Based Computing*. University of Aarhus, Computer Science Department
Abstract: As we engage in an increasing number of activities involving computational devices distributed in our environment, organizing and managing these activities becomes harder. This dissertation presents an approach to activity-based computing focus- ing on the distributed interaction between software systems and the individuals using them. The work involves three topics: (i) user-interfaces and interaction techniques for working in, and with, activities, (ii) infrastructures responsible for the runtime distribution of data and events, and (iii) mechanisms to enable application developers to expose their applications to activity-based integration. These three topics are treated each in a chapter in Part I of this dissertation. The first involves how to present activities in user-interfaces such that single individuals can better manage and interact with their activities, but also how groups of individuals are aided in their collaboration around one or more computational activities. The second topic, the software infrastructure, deals with the underlying technical issues of distributing and managing the runtime state of activities, and which type of software architecture to base a well-performing, but also flexible and extensible infrastructure on. The last topic revolves around how to enable external applications to be integrated into activities, and which handles for doing so are provided to developers of these applications. The contributions are within the field of activity-based computing and includes both designs and implementations of prototypical, but nevertheless complete systems which interoperate to form distributed applications, enabling individuals to use the activity-based computing concepts to support e.g. interrupted work and synchronous as well as asynchronous collaboration.

Martin Mogensen *Infrastructure Support for Collaborative Pervasive Computing Systems*. University of Aarhus, Computer Science Department
Abstract: Collaborative Pervasive Computing Systems (CPCS) are currently being deployed to support areas such as clinical work, emergency situations, education, ad-hoc meetings, and other areas involving information sharing and collaboration. These systems allow the users to work together synchronously, but from different places, by sharing information and coordinating activities. Several researchers have shown the value of such distributed collaborative systems. However, building these systems is by no means a trivial task and introduces a lot of yet unanswered questions. The aforementioned areas, are all characterized by unstable, volatile environments, either due to the underlying components changing or the nomadic work habits of users. A major challenge, for the creators of collaborative pervasive computing systems, is the construction of infrastructures supporting the system. Following an experimental computer science approach, where experiments are used both for theory testing and for exploration, we research the area of infrastructure support for collaborative pervasive computing systems. This research leads us to present four major contributions to the area. Firstly, we contribute by building real world Collaborative Pervasive Computing Systems, including the Activity-Based Collaboration

system and the iHospital system, which has been deployed and evaluated. Secondly, we contribute with novel hybrid and fusion Software Architectures. Thirdly, we contribute with peer-to-peer distributed shared objects as a Programming Paradigm and basic distribution mechanism. We have created and evaluated a framework called DOLCLAN as a full-blown implementation of the proposed concepts. Finally, we contribute to the Distribution Mechanisms researching epidemic protocols or gossiping as distribution protocols for pervasive environments.

Doina Bucur *On Context Awareness in Ubiquitous Computing*. University of Aarhus, Computer Science Department
Abstract: This dissertation contributes to the young field of ubiquitous computing, a so-called third wave of computing expected to succeed the current Internet era. The paradigm of ubiquitous computing calls for distributing the computation power among the people and objects in the human environment by embedding tiny, sometimes invisible, networked computers into the items of everyday life, so that (i) some of the interactions we engage in on a quotidian basis get redesigned as interactions with networked data-processing systems available everywhere, and (ii) the resulting networks of computers autonomously monitor and control themselves and the environment. In particular, the concept unifying the contributions of this dissertation is one of the central aspects in ubiquitous computing, context awareness – a paradigm calling for the tiny ubiquitous computers to sense the change in their computational surroundings and dynamically adapt their behavior to this environmental change. The scientific contributions of this dissertation are of both a theoretical and a practical nature, and lie within a number of areas of computing in context: sensing context (specifically, sensing location), discovery of context in large ubiquitous environments (a task known as service discovery), modeling and verification of context-aware systems, and the porting of traditional security techniques (specifically, access-control lists) to work over such mobile, dynamic systems.

Afsaneh Doryab *Activity Awareness in Collaborative Environments*. IT University of Copenhagen.
Abstract: Recent research in ubiquitous computing has focused on creating activity-aware systems, which are able to infer human activity and subsequently offer services that support the user's ongoing task, including recommending actions to take or information to consult. So far, such activity-aware systems has focused on a single person performing one activity at a time. However, in many environments, such as hospitals, activities are collaborative and concurrent i.e., involve several people who perform different activities at the same time. Automatic inferencing such collaborative and concurrent activities has proven difficult and has not been widely investigated. The overall purpose of our research is to address these issues and design recommendation techniques for pervasive systems in collaborative work. This research introduces a design of an activity-based system for surgical procedures in an operating room. Based on sensor input, the system automatically detects and recognizes co-located collaborative and concurrent activities in the operating room and uses this to show information and recommend actions to take, which are relevant to the current situation.

Jonathan Bunde-Pedersen and Martin Mogensen is currently employed at Cetrea A/S, a privately held Danish company. Doina Bucur worked as a post doc at Oxford University after graduating, and is currently a post doc at INCAS in the Netherlands. Afsaneh Doryab is still a PhD student at the IT University of Copenhagen, graduating in 2011.

Chapter 4

Project Management and Cooperation

4.1 Management

The steering group for the project has been:

- Ole Lerhman Madsen, AU – replace by Jørgen Staunstrup, ITU when the project moved to ITU.
- Frits Thomsen, Medical Insight
- Jakob Bardram, AU/ITU

Daily project management has been done by Jakob Bardram and Luuk Thomsen from Medical Insight.

4.2 Project Plan

The original project plan is shown in Figure 4.2. All in all, the project has followed this outline. The different Proof-of-Concepts was developed as part of what is now called version 3 and 4 of the ABC Framework described above. The hospital partner in the project was Horsens Sygehus (task #50), and the hardware and software for running the EasyViz PACS system has been deployed inside an operating room as illustrated in Figure 4.1. This was done by Medical Insight as part of this project (task #54-55).

The pilot phase was planned to run over several weeks (task #56). However, we could not do this for a number of reasons; mainly because the systems that the ABC Framework was integrated with (the Acure EMR and the EasyViz PACS) were not used at Horsens Sygehus. Hence, it was not possible to make a pilot study using real patient data at Horsens Sygehus. Instead we evaluated the ABC Framework using a scenario-based approach as presented in section 1.1.4 and we did a one-day simulation using fake patient data at Horsens Sygehus. The result of this evaluation is published in [16].

The aim of actually deploying an unstable research prototype into real use in a hospital environment can only retrospectively been seen as overly optimistic; deployment of technology – even fully developed and stable – into a hospital is a huge effort, and was clearly not achievable as part of this relatively small project. Despite this, we did manage to do a series of evaluations of the ABC concepts and technologies, which has helped understand how the ABC features and ideas can be implemented in the technologies of Medical Insight, Cetrea, and other companies that might find it useful.

4.3 Financial

The project was originally planned to last 3 years; from 2005–2008. But due to the fact that 4-year phd students became part of the project, and the project was moved to the IT University, the project was subject to a re-planning. The original budget stayed the same, but the project was extended by 2 years. As described above, substantial more

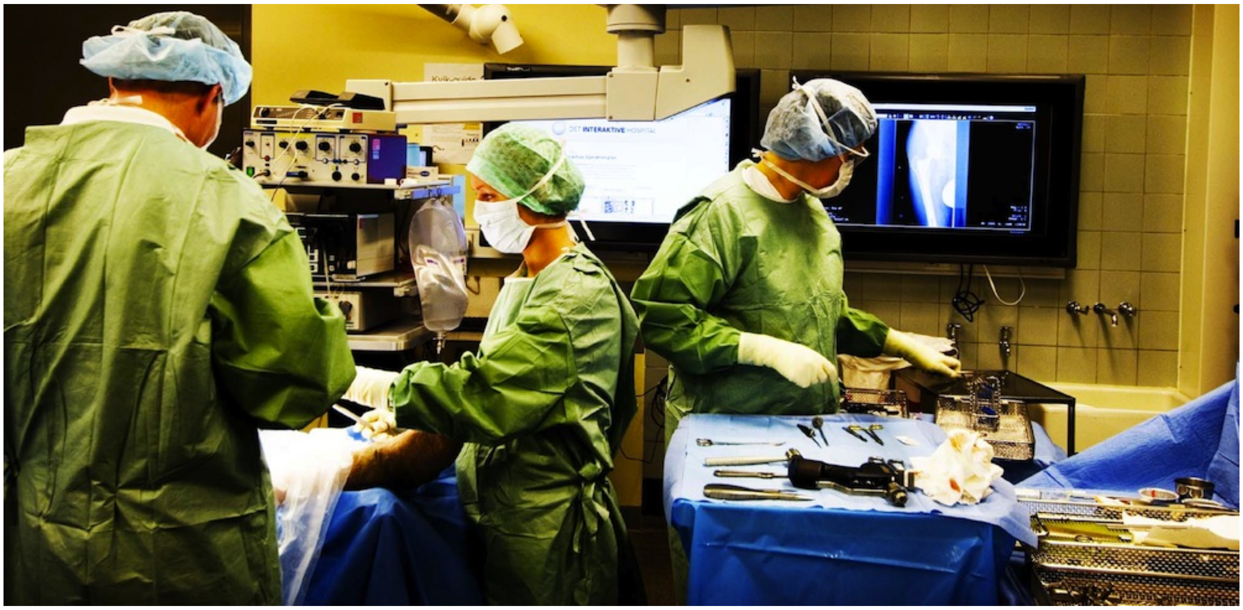
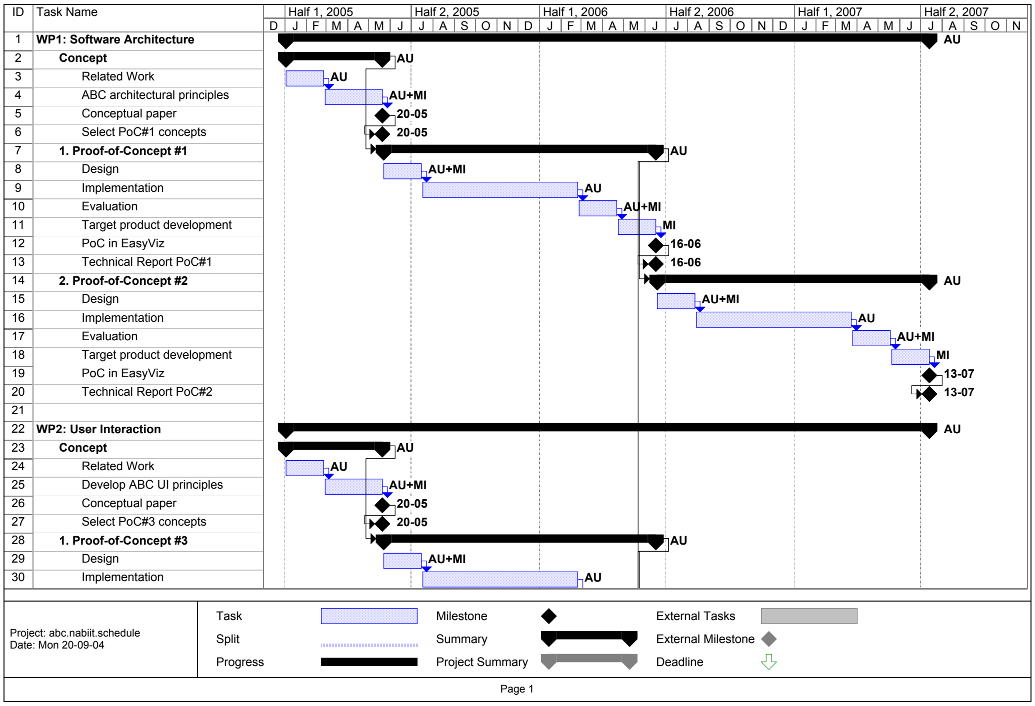


Figure 4.1: The Medical Insigh EasyViz PACS system deployed inside an operating room at Horsens Sygehus (on the display on the right).

co-financing from both University of Aarhus and the IT University of Copenhagen was secured to the project, with the main consequence that 4 rather than 2 phd students has been associated with the project.

All of the non-personal equipment bought by the project is now a part of the PIT laboratory[1] at the IT University of Copenhagen and is hence used in other research projects, including the ones mentioned in section 6.

[1] http://pit.itu.dk/



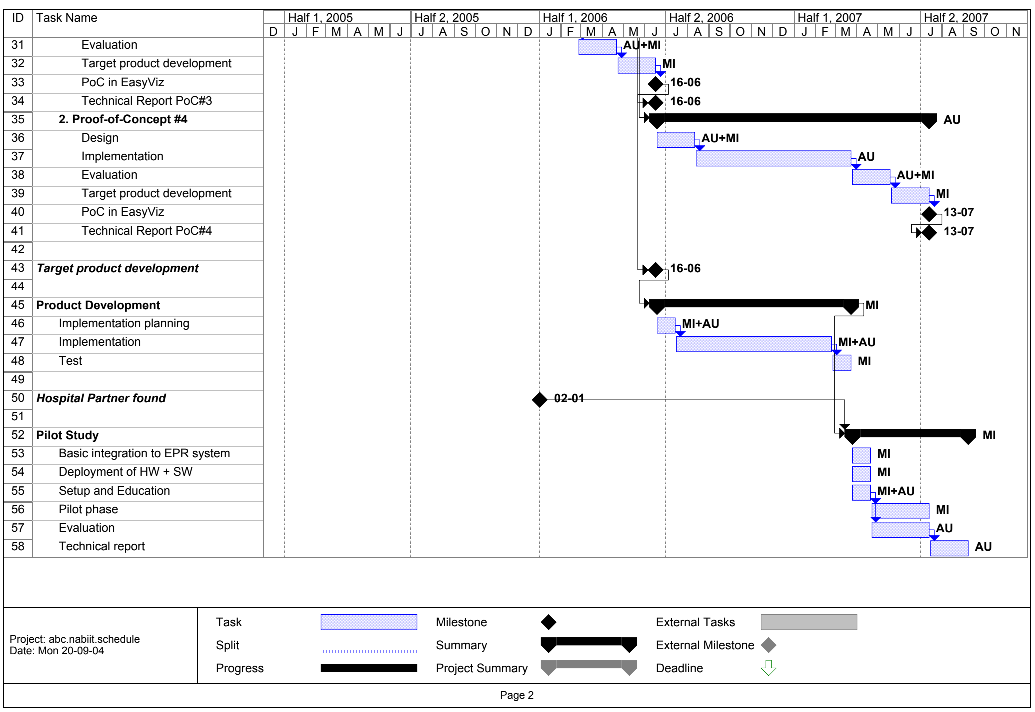


Figure 4.2: The original project plan for the ABC project, spanning 3 years, two main work packages, and a pilot study.

Chapter 5

Overall Reflections

In conclusion, we would like to point out the following reflections;

- We consider the ABC project to be highly successful seen from a scientific point-of-view. It has meet all of its scientific goals; has achieved considerable conceptual and technical new knowledge; and has published a wide range of conference and journal papers that make contributions in many different areas, including pervasive computing, context-aware computing, software architectures, human-computer interaction, and computer supported cooperative work.
- The business objective of the ABC project has been achieved, and the concepts and technologies developed as part of the ABC project has had an important impact on the technologies of the companies of Medical Insight A/S and Cetrea A/S.
- Joint collaboration with universities – like the ABC project – is important for knowledge based companies like Medical Insight, where sourcing of technologies and competences is key to success in business.
- The project has been the 'home' for several PhD projects, which has benefitted from the project in terms of scientific problems, directions, and concepts. Moreover, these students have been able to work together in the overall ABC project, while still leaving space for them to excel in their own research foci and projects.
- Financially, the project has stayed within budget, but only because additional financing from both the University of Aarhus and the IT University of Copenhagen were secured. The original budget was too low. Also, more resources for student programmers were needed.
- The initial goal of deploying the technology for a pilot trial in a hospital turned out to be overly optimistic. Deploying what can be characterized as a new operating system for pervasive computing inside a hospital based on a research proof-of-concept implementation is simply not doable. In future hospital-based project, one should seriously consider exactly how the technology is to be evaluated, and if a pilot trial is planned sufficient resources are needed in order to create stable technologies.
- More regular project meetings involving all the partners in the project should have been planned in order to secure more tight coordination between the research at the university and development in the companies.
- Moving the project from one host institution (University of Aarhus) to another (IT University of Copenhagen) proved to be unproblematic and caused no substantial problems beside a small delay. Resources were secure at AU to fund the work of the PhD students remaining there, and ITU have been very professional in moving the administration of the project from AU to ITU. Two out of the three PhD students at AU remained to be supervised by Jakob Bardram, after moving to ITU. Henrik B. Christensen at AU became the co-supervisor. The third PhD student changed to having Mogens Nielsen as the main supervisor, and Jakob Bardram as a co-supervisor. All students handed in their thesis on time.

Chapter 6

Ongoing Work

The concept of activity-based computing, the principles, and the technologies developed as part of this project is now being used in other research projects:

Trustworthy Pervasive Healthcare Services (TrustCare) – The overall goal of TrustCare is to address the challenges of creating trustworthy pervasive healthcare technologies. This is done in a strategic and interdisciplinary research effort aimed at innovation of effective and trustworthy it-support for pervasive healthcare services in collaboration with the industrial partner, as well as innovation in research across areas in experimental and theoretical research in computer science. The ABC concepts and technologies forms the basis for the design of technologies to be used in hospitals, and the funding in this project is used to further develop the ABC technology base in version 5. The TrustCare project is funded by the Danish Strategic Research Agency, Grant #2106-07-0019.

Collaborative Mini-Grids for Prediction of Viral RNA Structure and Evolution (Mini-Grid) – This project aims at designing a peer-to-peer volunteering infrastructure for bioinformatics algorithms as well as a set of novel interactive technologies for the biology laboratory, which makes research into RNA-based diseases like HIV, SARS, and bird flu more efficient than with current approaches. The project is interdisciplinary and involves researchers from computer science, bioinformatics, molecular biology, and nanotechnology. The partners involve the IT University of Copenhagen, the Department of Molecular Biology, at the interdisciplinary nano-science centre (iNANO) at Aarhus University (AU), and CLC Bio A/S. The ABC concepts and technologies are used as the basic infrastructure in the development of a co-called "iLabBench", i.e. an interactive laboratory bench with a wide range of embedded sensor and displays technologies. This project is funded by the Danish Agency for Science, Technology, and Innovation, grant #09-061856.

Next Generation Technology for Global Software Development (GSD) – This project aims at providing knowledge and tools for organizations to excel in software development on a global scale. The two core contributions of this project are (i) to view cultural diversity as an opportunity for increased innovation, and (ii) to build technologies that help practitioners to move from an outsourcing model to a collaborative model of GSD. One core hypothesis in this project is that Activity-Based Computing (ABC) can be used to create a technological platform for such integrated and continuous coordination in GSD. This project has just been granted from the Danish Strategic Research Agency

Bibliography

[1] Jakob Bardram and Adrian Friday. Ubiquitous computing systems. In John Krumm, editor, *Ubiquitous Computing Fundamentals*, pages 37–94. CRC Press, 2009.

[2] Jakob E. Bardram. Hospitals of the Future - Ubiquitous Computing support for Medical Work in Hospitals. In Jakob E. Bardram, Ilkka Korhonen, Alex Mihailidis, and Dadong Wan, editors, *UbiHealth 2003: The 2nd International Workshop on Ubiquitous Computing for Pervasive Healthcare Applications*. Seattle, WA, USA, October 2003.

[3] Jakob E. Bardram. Supporting Mobility and Collaboration in Ubiquitous Computing. Technical Report CfPC 2003-PB-38, Centre for Pervasive Computing, Aarhus, Denmark, 2003. Available from http://www.pervasive.dk/publications.

[4] Jakob E. Bardram. Activity-Based Service Discovery - An Approach for Service Composition, Orchestration and Context-Aware Service Discovery. Technical Report CfPC Technical Report 2004-PB-67, Centre for Pervasive Computing, Aarhus, Denmark, 2004. Available from http://www.pervasive.dk/publications.

[5] Jakob E. Bardram. Activity-Based Support for Mobility and Collaboration in Ubiquitous Computing. In Luciano Baresi, editor, *Proceedings of the Second International Conference on Ubiquitous Mobile Information and Collaboration Systems (UMICS 2004)*, Lecture Notes in Computer Science, pages 169–184, Riga, Latvia, September 2004. Springer Verlag.

[6] Jakob E. Bardram. Activity-Based Computing - Lessons Learned and Open Issues. In Jakob E. Bardram and Tom Moran, editors, *Proceedings of the ECSCW 2005 Workshop on 'Activity - From a Theoretical to a Computational Construct'*. 2005. http://www.daimi.au.dk/ bardram/ecscw2005/.

[7] Jakob E. Bardram. Activity-Based Computing: Support for Mobility and Collaboration in Ubiquitous Computing. *Personal and Ubiquitous Computing*, 9(5):312–322, July 2005.

[8] Jakob E. Bardram. The Java Context Awareness Framework (JCAF) - A Service Infrastructure and Programming Framework for Context-Aware Applications. In Hans Gellersen, Roy Want, and Albrecht Schmidt, editors, *Proceedings of the 3rd International Conference on Pervasive Computing (Pervasive 2005)*, volume 3468 of *Lecture Notes in Computer Science*, pages 98–115, Munich, Germany, May 2005. Springer Verlag.

[9] Jakob E. Bardram. The Trouble with Login - On Usability and Computer Security in Ubiquitous Computing. *Personal and Ubiquitous Computing*, 9(6):357–367, 2005.

[10] Jakob E. Bardram. From Desktop Task Management to Ubiquitous Activity-Based Computing. In Victor Kaptelinin and Mary Czerwinski, editors, *Integrated Digital Work Environments: Beyond the Desktop Metaphor*, pages 49–78. MIT Press, 2007.

[11] Jakob E. Bardram. Activity-based Computing Support for Agile and Global Software Development. In *CSCW 2008 Workshop on Supporting Distributed Team Work (VRC2008)*, San Diego, CA, November 2008.

[12] Jakob E. Bardram. Activity-based computing for medical work in hospitals. *ACM Trans. Comput.-Hum. Interact.*, 16(2):1–36, 2009.

[13] Jakob E. Bardram. A novel approach for creating activity-aware applications in a hospital environment. In Tom Gross, Jan Gulliksen, Paula Kotzé, Lars Oestreicher, Philippe Palanque, Raquel Oliveira Prates, and Marco Winckler, editors, *Human-Computer Interaction – INTERACT 2009*, volume 5727 of *Lecture Notes in Computer Science*, pages 731–744. Springer, 2009.

[14] Jakob E. Bardram, Heribert Baldus, and Jesus Favela. Pervasive Computing in Hospitals. In Jakob E. Bardram, Alex Mihailidis, and Dadong Wan, editors, *Pervasive Healthcare: Research and Applications of Pervasive Computing in Healthcare*, pages 49–78. CRC Press, 2006.

[15] Jakob E. Bardram and Jonathan Bunde-Pedersen. A2BC - An Agent Approach to Activity-Based Computing. Technical report, Centre for Pervasive Healthcare, Aarhus, Denmark, 2005.

[16] Jakob E. Bardram, Jonathan Bunde-Pedersen, Afsaneh Doryab, and Steffen Sørensen. Clinical surfaces – activity-based computing for distributed multi-display environments in hospitals. In Tom Gross, Jan Gulliksen, Paula Kotzé, Lars Oestreicher, Philippe Palanque, Raquel Oliveira Prates, and Marco Winckler, editors, *Human-Computer Interaction – INTERACT 2009*, volume 5727 of *Lecture Notes in Computer Science*, pages 704–717. Springer, 2009.

[17] Jakob E. Bardram, Jonathan Bunde-Pedersen, Afsaneh Doyrab, and Steffen Sørensen. MPAD - Activity-Based Support for Distributed Multiple Display Environments. Technical Report, IT University of Copenhagen, 2008. 2008.

[18] Jakob E. Bardram, Jonathan Bunde-Pedersen, and Martin Mogensen. Differentiating between Accountable and Ephemeral Events in the ABC Hybrid Architecture for Activity-Based Collaboration. In *Proceedings of the IEEE International Conference on Collaborative Computing (CollaborateCom 2005)*, pages 168–176. IEEE Press, 2005.

[19] Jakob E. Bardram, Jonathan Bunde-Pedersen, and Mads Soegaard. Support for activity-based computing in a personal computing operating system. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 211–220, New York, NY, USA, 2006. ACM Press.

[20] Jakob E. Bardram and Henrik B. Christensen. Middleware for Pervasive Healthcare - A White Paper. In Guruduth Banavar, editor, *Advanced Topic Workshop - Middleware for Mobile Computing*. http://www.cs.arizona.edu/mmc/Program.html, Heidelberg, Germany, November 2001.

[21] Jakob E. Bardram and Henrik B. Christensen. Open Issues in Activity-Based and Task-Level Computing. In Jakob E. Bardram, Henrik B. Christensen, David Garlan, and Joao Sousa, editors, *Proceedings of the First International Workshop on Computer Support for Human Tasks and Activities*, volume PB-2004-60 of *Centre for Pervasive Computing Technical Reports*. University of Aarhus, Aarhus, Vienna, Austria, April 2004.

[22] Jakob E. Bardram and Henrik B. Christensen. Real-time Collaboration in Activity-based Architectures. In *Proceedings of Fourth Working IEEE/IFIP Conference on Software Architecture (WICSA'04)*, pages 325–329. IEEE Press, 2004.

[23] Jakob E. Bardram and Henrik B. Christensen. Real-Time Collaboration in Activity Based Architectures. Technical Report CfPC Technical Report 2004-PB-62, Centre for Pervasive Computing, Aarhus, Denmark, 2004. Available from http://www.pervasive.dk/publications.

[24] Jakob E. Bardram and Henrik B. Christensen. Supporting Pervasive Collaboration in Healthcare - An Activity-Driven Computing Infrastructure. Technical Report CfPC Technical Report 2004-PB-63, Centre for Pervasive Computing, Aarhus, Denmark, 2004. Available from http://www.pervasive.dk/publications.

[25] Jakob E. Bardram and Henrik B. Christensen. Pervasive computing support for hospitals: An overview of the activity-based computing project. *IEEE Pervasive Computing*, 6(1):44–51, 2007.

[26] Jakob E. Bardram, Henrik B. Christensen, David Garlan, and Joao Sousa. WCSHTA 2004: The 1st International Workshop on Computer Support for Human Tasks and Activities. In Alois Ferscha and Friedemann Mattern, editors, *Adjunct Proceedings of Pervasive 2004*, Vienna, Austria, April 2004. Springer Verlag.

[27] Jakob E. Bardram, Henrik B. Christensen, and Anders K. Olesen. Activity-Driven Computing Infrastructure - Pervasive Computing in Healthcare. Technical Report CfPC Technical Report 2004-PB-65, Centre for Pervasive Computing, Aarhus, Denmark, 2004. Available from http://www.pervasive.dk/publications.

[28] Jakob E. Bardram, Rasmus E. Kjær, and Michael Ø. Pedersen. Context-Aware User Authentication - Supporting Proximity-Based Login in Pervasive Computing. In Anind Dey, Joe McCarthy, and Albrecht Schmidt, editors, *Proceedings of UbiComp 2003*, volume 2864 of *Lecture Notes in Computer Science*, pages 107–123, Seattle, Washington, USA, October 2003. Springer Verlag.

[29] Jakob Eyvind Bardram and Jonathan Bunde-Pedersen. Iaso - an activity-based computing platform for wearable computing. In *ICDCSW '05: Proceedings of the Fifth International Workshop on Smart Appliances and Wearable Computing (IWSAWC) (ICDCSW'05)*, pages 484–490, Washington, DC, USA, 2005. IEEE Computer Society.

[30] Doina Bucur and Jakob E. Bardram. Resource Discovery in Activity-Based Sensor Networks. In *Proceedings of 1st International Conference on Pervasive Computing Technologies for Healthcare 2006*, 2006.

[31] Doina Bucur and Jakob E. Bardram. Resource discovery in activity-based sensor networks. *MONET*, 12(2-3):129–142, 2007.

[32] Jonathan Bunde-Pedersen, Martin Mogensen, and Jakob E. Bardram. The abc adaptive fusion architecture. In *MPAC '06: Proceedings of the 4th international workshop on Middleware for Pervasive and Ad-Hoc Computing (MPAC 2006)*, page 1, New York, NY, USA, 2006. ACM Press.

[33] Henrik B. Christensen and Jakob E. Bardram. Supporting Human Activities - Exploring Activity-Centered Computing. In Gaetano Borriello and Lars Erik Holmquist, editors, *Proceedings of Ubicomp 2002: Ubiquitous Computing*, volume 2498 of *Lecture Notes in Computer Science*, pages 107–116, Gothenborg, Sweden, September 2002. Springer Verlag.

[34] Henrik B. Christensen and Jakob E. Bardram. An Initial Comparison between Aura and Activity-Based Computing. Technical Report CfPC Technical Report 2004-PB-56, Centre for Pervasive Computing, Aarhus, Denmark, 2004. Available from http://www.pervasive.dk/publications.

[35] Henrik Bærbak Christensen. Using Logic Programming to Detect Activities in Pervasive Healthcare. In Peter Stuckey, editor, *Proceedings of International Conference on Logic Programming ICLP 2002*, Lecture Notes in Computer Science 2401, Copenhagen, Denmark, July 29–August 1 2002. Springer Verlag.