Mobile Learning in Context – Context-aware Hypermedia in the Wild

doi:10.3991/ijim.v3i1.766

Frank Allan Hansen and Niels Olof Bouvin
Aarhus University, Århus, Denmark

Abstract—**Modern project-based education requires students to be able to work with digital materials both in and out of the classroom. Field trips are often an integral part of such projects and greatly benefit students' learning by allowing them to engage with real-world environments firsthand. However, the infrastructure for accessing context sensitive information and supporting in-situ authoring by students while in the field is often lacking. In this paper we present the HyCon framework for mobile, context-aware, and multi-platform hypermedia that aims at supporting several aspects of fieldtrips and project-based education. The framework and its applications utilize information about the users' digital and physical context in order to support digital-physical linking of multimedia materials, and it supports browsing, searching, and creation of new materials which can be linked to points in time and space for later inspection. The HyCon framework is based on a conceptual model for handling context in its myriad forms, as well as a extensible software architecture, and we describe in this paper both. We also describe a number of mobile applications developed with the framework and targeted especially towards mobile learning. Based on a number of field tests with students and teachers from local schools, we evaluate the mobile applications, discuss the experiences gained during the development and evaluations, the requirements for mobile learning tools and the implications for leaning.**

Index Terms—**Nomadic Learning, Mobile learning, Context-awareness, Context-aware Hypermedia, Geospatial Hypermedia, Physical linking, Tagging.**

I. INTRODUCTION

It is becoming didactically desirable as well as technically possible to move learning outside of the classroom and take advantage of the rich sources of information available beyond books and computer screens. It is e.g., possible to gain basic knowledge of what constitutes working at a construction site by reading a book, but the book cannot convey how work is coordinated, how noisy the environment is, how safety is ensured through the action of the workers etc. Taking a field trip to a construction site is a much richer source of information if we wish to proper grasp working conditions. Schools, however, are ill-equipped in supporting this kind of project-based education. In particular, the process of *collecting*, *producing* and *presenting* information from heterogeneous sources and working with this information in and out of the classroom; most schools have dedicated computer rooms and Internet connection, but cannot move the digital information outside the computer rooms.

However, the development in pervasive and mobile technology makes it possible to combine digital information with the physical environment and enables the students to carry a digital context around with them. We see great potential in using knowledge about location and purpose to frame the information made available at a given point in time and space and to let teachers and students play an active part in building the digital context made available on site and sharing it with others.

In this paper we focus on the technical challenges and techniques required to support mobile information work that make sense in a project-based education scenario, where information needs to be understood in context. Based on an empirical study and extensive work with teachers and students from several elementary schools in the Århus area in Denmark, we identify and describe a number of requirements for project based education taking place in- and outside the classroom. We focus on three aspects of project-based education that support learning outside of the classroom through context-aware hypermedia and contextualization of information:

Browsing with your feet is the activity of searching and browsing in context and taking advantage of knowledge about time, place, and purpose when investigating information in the field. Searching and browsing on-site can be dramatically improved by using mobile and context-aware technology. When students are out in the field, they will often need information about the place, they are moving through or are standing in front of. By using knowledge about the user's location, search and browsing can be focused to the context and drastically improve the number and relevance of the search results.

Annotating the world refers to the activity of producing information and writing "digital graffiti" i.e., leaving information in the world tied to the physical place of creation. Annotations are meant to support the students' project activities outside the classroom by enabling them to produce material on-site tied to the location. The students can leave traces of their ongoing activities in the environment that can be revisited later or "bumped into" by others as they travel through the same zone.

Overview at a glance is the ability to quickly provide an overview of the available digital material based on contextual meta-data. Knowing what you have done and where you have been are important aspects of most project presentations. The collection and preservation of the contextual information can be used to create an overview of the process, and allow the children to retrospectively trace their journey, project material, and knowledge building both spatially, temporally, and conceptually.

> *Peter is a biology teacher in Århus. He has been teaching his 7th graders about animals and their habitats and as part of the project, the class is now going to an area near the school to investigate a number of different biotopes. The area includes a small wood, a waterhole, and a patch of wild plants. Peter inspects a map of the area in the hypermedia application on his laptop and links the Web pages he has used in his classes to points on the map. He then writes assignments and questions related to each of the three biotopes and create a guided tour through the area, which the students should follow. The next day, the children begin the field trip equipped with Smartphones connected to RFID readers and GPS receivers that supply sensor data to the applications on the phones. As they arrive at the wood, the mobile hypermedia application on the phones presents them with a map of the area with Peter's links and assignment overlaid on top of the map. The students begin answering the question by taking pictures of plants and animals and writing short notes. At the waterhole they collect insects and place them in jars marked with RFID tags. Once back in the classroom, the students can browse the material on Web pages generated by the hypermedia services. They create multimedia presentations with facts about the three habitats illustrated with the notes, pictures, and videos produced in the field and with the material Peter provided. While they present their work to each other and to Peter, they bring out the jars with insects and plant samples and places them on a RFID reader in the classroom. The pictures they took at the waterhole appear on a Smartboard in the classroom and they explain where the different insects were found and how they live. Both the children and Peter are satisfied with the project and Peter decides that he will use the material, the children created and linked to locations in the three biotopes, in next years assignments to give the coming students an understanding of the change of the animals and their habitats over time.*

Figure 1. Scenario: Context-aware Hypermedia for Mobile Learning

To see how these activities can be supported by context-aware hypermedia, consider the scenario described in Figure 1. This scenario illustrates several important requirements. First of all, the scenario applies hypermedia to mobile computing and suggests hypermedia systems for Smartphones to support access to information in the field. Through a set of simple sensors the system gets information about the physical environment (in this case information about the location from the GPS and identity of the nearby objects, the children work with from the RFID reader) and uses this context information to find and present structures and documents that are potential interesting for the user in the current situation. In the scenario, the students also create new material with their Smartphones, and use (implicitly) the gathered context information to link documents to physical places and to the plants and insects they collected. Hence, the classical hypermedia support for browsing and structuring has been applied to physical phenomena and augmented with context-aware capabilities. However, the scenario also hints at requirements for the infrastructure needed to run this kind of system. Information is produced and accessed from a number of different devices (laptops, Smartphones, Smartboards, etc.) in a number of different places (Peter's office, the classroom, and in the field). This is a typical example of a ubiquitous computer environment where a number of heterogeneous devices with diverse capabilities participate across users' activities and utilize the same data and infrastructure. Both the hypermedia infrastructure and the components that utilize it must be flexible enough to integrate into this technology web where devices come and go and new devices, sensors, services, and data types are introduced over time.

We present in this paper a conceptual framework for context-aware hypermedia that support mobile learning. We discuss the design and evaluation of an implementation of the conceptual model, namely the *HyperContext* (HyCon) framework and some of its applications, which explore different aspects of our general model, and discuss the challenges in developing these kinds of systems both technical and conceptual. The HyCon applications have been evaluated in an educational setting with sixth to seventh grade classes. The framework and its applications have previously been described in [4][5][22][24].

The rest of the paper is structured as follows: Section II discusses the notion of context and context-aware computing and presents our conceptual model for context, which is used as the foundation for our implementations. Section III describes our data model for context-aware hypermedia and the HyCon framework architecture. Section IV discusses the requirements for mobile learning, learning activities and the applications we have built to support these activities. Section V describes the technical implementation of the prototype applications. Section VI discusses evaluations and lessons learned and Section VII presents a number of related efforts. Finally, Section VIII concludes the paper.

II. CONCEPTUAL FRAMEWORK

The notion of computers responding according to their users' implicit stated context is an intriguing and challenging one. In this section we describe our conceptual model for context-aware hypermedia and in Section III we describe two different implementations of this conceptual model. However, before presenting the model we discuss a number of previous efforts to define and model context.

A. *Context-aware Computing*

Context-aware computing refers to software systems that can adapt their behavior, interface and structures according to a user's context. Information about the context can be made available to the system either explicitly by the user (e.g., by entering a login name) or implicitly through associated sensory systems. Adding context-aware capabilities to systems is desirable for several reasons: firstly, interaction with applications can be greatly improved if an application adapts to a user's situation and only provides information relevant to that situation. For example, filtering information about a city based on a user's location can make it much easier for users to gain an overview, especially given the limitations of browser user interfaces on small, mobile devices [24]. Secondly, knowledge about the physical environment surrounding a user can be used to create new powerful applications. Hypermedia, for instance, has traditionally been concerned with linking digital resources stored on computers, but the ability to identify physical resources through tagging or sensor input enables hypermedia systems to also support linking to locations and physical objects.

B. *Previous Definitions of Context*

The idea of utilizing location in computing systems was advocated by Mark Weiser in his seminal 1991 paper on ubiquitous computing [44]. Some of the early research on location- and context-aware computing was performed on the Olivetti Active Badge systems in 1992 [41]. The term "context-aware computing" was however not defined until 1994, where Schilit et al. listed three important aspects of context: where you are, who you are with, and what resources are nearby [37][38]. Schilit et al. defined context as a location and its dynamic collection of nearby people, hosts, and devices. A number of similar definitions exist that all try to define context by enumerating examples of context elements [6][14][36]. However, as discussed by Dey [15], these types of definitions may be too specific and hard to apply when trying to define general support for context-awareness: e.g., how is it decided whether a type of information can be regarded as context if it is not listed in the definition, and how does a system handle new types of contexts if its design is based on a fixed set of types? Later, definitions of context became more general. Schmidt et al. [39] use the following definition: "[Context is defined as] knowledge about the user's and IT device's state, including surroundings, situation, and to a less extent location." Chen and Kotz [9] define context as "the set of environmental states and settings that either determines an application's behavior or in which an application event occurs and is interesting to the user." Similarly, Dey [15] provides the following definition: "Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves." These latter definitions are easier to apply than the former since information can be categorized as context on a per application basis, and they can be used as guidelines for designing general support for context-awareness.

C. Previous Models of Context

The different notions of context have led to a number of varied approaches to modeling and handling context in computer systems. Context defined as a fixed set of attributes can be represented by a hard-coded, optimized context model, whereas a more general definition will lead to more generic and flexible models. Typically, data structures include key/value pairs, tagged encodings, object-oriented models, and logic-based models [9]. In hypermedia, however, the prevalent way to model context has typically been as either composites serving as partitioning mechanisms on the global network or as key/value pairs—associated with links, nodes and other hypermedia objects—that describe parameters of the context the hypermedia objects belong to. Context modeled as composites is often a purely structural partitioning concept constraining browsing and linking to some kind of context given by the user explicitly or implicitly. Key/value pairs associated with objects are typically used to describe in which context objects are visible. Neptune's Contexts [13], Intermedia's Webs [46], and the Webvise Context composites [18] belong to the former category while FOHM's context objects [29] and Storyspace's guard fields [3] belong to the latter. Key/value pairs have also been used to model context in other context-aware systems e.g., Schilit's and Theimer's located objects [37] and the context-aware Web browser Mobisaic [40].

D. Conceptual Context Model

We view the definitions of context discussed above as a good starting point for a common context definition for hypermedia. When Chen and Kotz [9] describe context as "[a] set of environmental states and settings" and Dey [15] uses the term "any relevant information" to define context, it sets some requirements for the design of a hypermedia framework. With context defined in general terms rather than specific entities, our framework must be able to handle context data in a very general way. Context must necessarily be modeled and defined as specific entities at some point, but this decision should generally be deferred to the design of concrete applications or perhaps even to run-time, where users should be free to specify the nature and format of context objects. In order to ease the categorization of context information and to facilitate designing for context, we adopt the terminology developed by Brynskov et al. [8] to help distinguish the use of "context" as a concept at different levels of system development. Context is classified into three domains: *physical*, *digital*, and *conceptual*. These domains ease the transition from traditional digital-only hypermedia models to models that also encompass a notion of physical objects and contexts:

Physical context includes the physical surroundings of an entity. This includes physical location, physical objects, physical interaction, absolute time and space, and other physical measurements. Computer systems may be aware of the physical context by using sensors.

Digital context includes computer models, infrastructure, protocols, devices, resources and services, logs, and relative time and space. Specifically, traditional hypermedia structures such as links, collections, guided tours, etc. correspond to objects that can be used to represent relationships in the digital context.

Conceptual context describes user activity, intention, focus, and understanding of surroundings.

Most entities may have both a physical and a digital representation, i.e., the physical phenomenon is modeled in the computer system. Ideally, the relationship between the physical entity and the digital model should correspond to the user's understanding of his or her current context (represented as the conceptual model or user model). The representation of physical entities in the digital model should thus reflect the conceptual context. As an example, consider a user using a Smartphone with a context-aware hypermedia system. Picture the user located in front of a building, and the system being aware of the user's position through the Smartphone's integrated GPS receiver. Based on this sensed data from the physical context, the hypermedia system can search its database (i.e., the digital context) for information pertinent to this particular physical context. In our example, the system finds a resource describing the building and presents it to the user. Furthermore, since additional resources annotate the displayed document, links to these resources are presented in the interface allowing the user to further explore the document's digital context. In this example, the user will have no problem coupling the physical building with the document describing it in the hypermedia system, and the conceptual context therefore corresponds to the relationship between the physical and digital contexts.

The main motive for adopting these three domains is not to enumerate a definitive list of parameters essential to the understanding of context, but rather to partition the design space. This allows developers to focus on the aspects they wish to support in their context-driven applications. The digital context corresponds to the classical problem domain of computer science and can be handled by well-known techniques from this field. Specifically, we can apply hypermedia models and structuring mechanisms to represent the relationships between digital entities. The physical context has traditionally not been a concern in computer science, but handled by other fields such as architecture, engineering, and (industrial) design. However, much work in the ubiquitous computing, pervasive computing, and context-aware computing fields has focused on capturing information about the physical world through sensory systems and making this information available to computer applications. Thus, we can leverage these techniques and apply them to hypermedia in order to handle the physical context. The conceptual context is the human factors—the internal context, which cannot easily be captured by sensors. It includes the user's task, goals, intent, and the user's internal model of the context. Again, this is something outside the computer, but research in educational systems and adaptive hypermedia systems [7] have developed techniques that try to build profiles of the user (user models), which can be used to adapt information and application behavior. These three domains and their supporting techniques are depicted in Figure 2.

III. DATA MODEL AND SOFTWARE ARCHITECTURE

Based on our conceptual framework, we now present our data model and our software architecture. Our notion of context is extensible by definition, and our data model must follow suit. We also discuss in this section how sensors for physical data are integrated in our architecture and how the data models handle the heterogeneous set of sensor data.

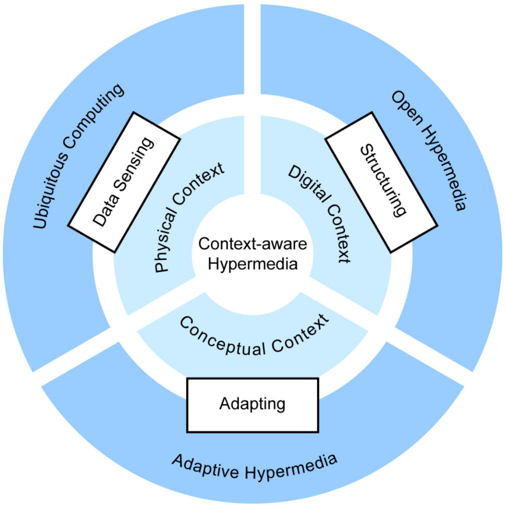


Figure 2. Three context domains: physical-, digital-, and conceptual context, and techniques that support them

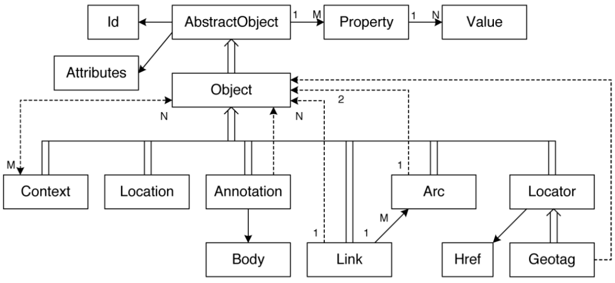


Figure 3. Overview of the Object oriented HyCon data model.

A. *Context Model Implementations*

In our work with the HyCon platform [5][24], we experimented with several ways to represent context. HyCon provides an object-oriented data model (shown in Figure 3) with context data that can be described by sub-classing an abstract data object.

These specialized objects can be associated with other data objects via a dedicated context composite. In addition to this approach, we also allowed key/value pairs to be associated with data objects to support the simple context tagging mechanism described above. During our evaluation of the HyCon framework [21] we found that while the composite mechanism is useful for modeling certain kinds of physical contexts, especially containment—such as locations where one location (a room) may be part of another location (a building)—this mechanism was seldom used in practice. And in addition, the object-oriented approach often resulted in minor changes to the existing framework services, so programmers preferred the simpler key/value pair model when new applications were created or old classes changed.

This observation led us to generalize the data model and develop the data model depicted in Figure 4. In this data model we use tagging as the fundamental paradigm for modeling data objects *and* their context. All resources are modeled as tag-able `BaseObjects`. If a resource has a certain property, we simply "tag" the resource with this property (as a typed property object). This simple mechanism allows us to model normal data objects, e.g, objects which have properties such as "`name`" or "`id`", but also their contexts e.g., adding GPS coordinates or a 2D barcode URL as properties of the object. The advantages of this structuring model is that we get a completely uniform data model, where it is just as easy to find an object with a given GPS coordinate as it is to find objects with a certain name or type, as all this information is represented by tagged properties.

This model also makes it very easy to handle heterogeneous sensor data. As depicted in Figure 4, a number of sensor inputs can be associated with a given resource and afterwards data from any of these sensors may be used to create a link to the resource, e.g., if a resource is tagged with both a 2D-barcode, an RFID tag, and a GPS position, the resource may be retrieved with data from any of these three sensor types. This mechanism is quite flexible, as it does not require special purpose structures for representing different kinds of physical contexts, but simply relies on object tagging and queries over the data to represent the digital-physical relationships.

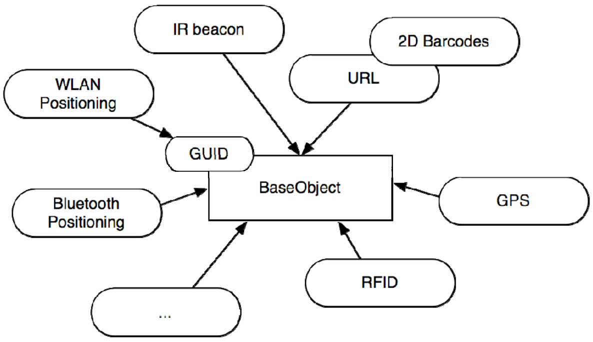


Figure 4. Context-aware hypermedia model as an extension of the notion of Web 2.0 tagging.

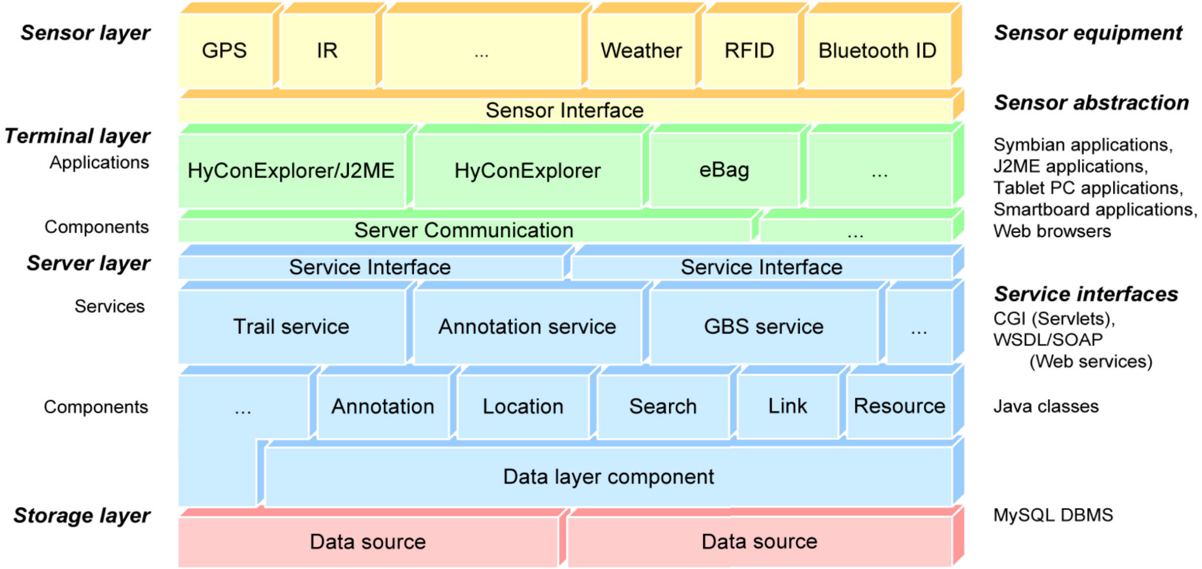


Figure 5. The HyCon service framework architecture.

A. HyCon Framework Architecture

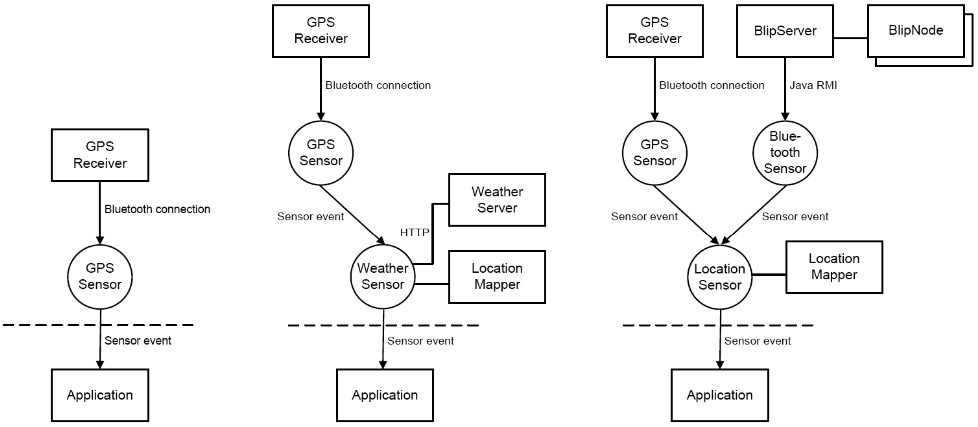
The HyCon framework was developed to provide a general platform suitable for experiments with hypermedia mechanisms in a context-aware and mobile environment. HyCon encompasses an infrastructure for implementing context-aware services and applications and a framework, which can be used by applications programmers to build such services and applications. The HyCon framework was designed from a number of requirements in mind: The framework should support a data model combining both hypermedia structures and context objects and at the same time be extensible enough to support different types of context specified by application programmers. This requirement follows directly from our conceptual model and is realized by both data models described above. Having determined what constitutes context for a specific application, the framework should also support developers in implementing behaviors and features to take advantage of the context information in their applications. Furthermore, in order to support different aspects of project-based education as discussed later, a basic requirement for the design was to allow a large number of heterogeneous clients ranging from simple mobile phones to desktop based application, Smartboards, interactive floors and Web browsers to be used with the framework.

The HyCon service architecture is divided into four logical layers: Storage, Server, Terminal, and Sensor layer as illustrated in Figure 5. This architectural approach is quite similar to earlier hypermedia architectures, such as the Dexter architecture [20], the Open Hypermedia System Working Group's Open Hypermedia Model [34] and Construct [45]. However, key to the HyCon service architecture is the extra layer, the Sensor layer, dedicated to handle sensors and sensor information for terminal layer applications. The bottom layer, the Storage layer, handles persistent storage and retrieval of data, hypermedia structures and context structures produced in the system.

The Server layer is divided into core components implementing functionality for server layer applications i.e., services. The framework includes a number of pre-made components implementing functionality for the location, linking and annotation services. Applications in the Server layer can be developed by using one or several existing components from the framework and implementing specialized functionality, which is not available through the mixture of components e.g., computation on the output from different components. This design provides a mechanism for decoupling the responsibilities of the building blocks and not creating mutual dependencies between individual components in the framework.

Interfaces to the services in the Server layer are provided for communication with applications in the Terminal layer. The Terminal layer is not limited to applications running on a specific platform, but includes applications running on a variety of hardware platforms and software environments (mobile phones, Tablet PCs, laptops, Web browsers etc.). The component based design from the Server layer is also used in the Terminal layer: reusable components are e.g., implementing the communication to the server layer. The applications are implementing functionality and user interfaces appropriate for the terminal device in use. However, Terminal components are not reusable across platforms, but are tied to specific platforms (e.g., Java JME, Windows CE, or Symbian).

An innovation in the HyCon framework is the last layer, the Sensor layer. The purpose of the Sensor layer is to create a logical separation between application specific code and code concerned with acquiring context information for the application. The Sensor layer is implemented as an abstraction that specifies how sensor components integrate with and notify application of changes in the context. While this means that application programmers do have to write some code to handle sensors, it has the advantage that no complex sensor infrastructure need to be deployed to utilize sensors in the Sensor layer. Especially for mobile systems and field trip applications used for mobile learning, this can be a major benefit, as sensor infrastructures are not deployed in all cities or in nature areas. In such cases the sensor equipment can simply be brought along together with the mobile devices and accessed through the Sensor layer abstraction. Furthermore, the decoupling between application code and sensor code in the Sensor layer makes it easy to reuse sensor components across applications running on the same platform.



(a) Context sensor (b) Context interpreter (c) Context aggregator

Figure 6. The HyCon sensor abstraction.

B. *HyCon Sensor Abstraction*

In HyCon, sensor information is accessed through tailored interfaces to components handling the parsing and computation of the raw sensor data. A HyCon sensor is a component that implements the sensor interface. A sensor component must allow applications to access context information and sensor history through both publish-subscribe and traditional poll mechanisms. If an application uses the publish-subscribe interface, it will receive a sensor event each time new context information is available. In this way, context information is made available in a similar fashion as explicit input from keyboards and pointing devices. Sensors can, through the sensor interface, make context information available to applications independently of how this information is sensed. There is no difference between accessing information from a sensor-implementation providing information directly from a physical sensor or from a purely virtual sensor, which aggregates information from a number of other sensor components.

Figure 6 illustrates three different HyCon sensor configurations. Figure 6(a) is a simple sensor that acts as an interface to a Bluetooth GPS receiver. The sensor implementation handles connections to the Bluetooth module and computation on the raw NMEA[1] data to provide context information as location objects delivered through sensor events to the application. Figure 6(b) is an interpreter used in the HyConExplorer to provide weather information and forecasts to the application. The weather sensor aggregates a GPS sensor. The location data from the GPS sensor is mapped to specific areas in the location mapper module, and this information can in turn be used to fetch weather data from a service on the Internet. In this configuration the data from the GPS sensor is interpreted to provide other interesting context information to the user. Figure 6(c) is an aggregated location sensor. The sensor aggregates two other sensors, the GPS sensor and a Bluetooth location system. The Bluetooth location system used here is based on an infrastructure of Bluetooth base stations (BlipNodes[2]) which make information available about nearby Bluetooth devices through a dedicated server (BlipServer). The aggregating sensor uses a mapping between BlipNodes and their (*x*, *y*, *z*)-coordinates in 3D space to provide the application with uniform context information, independently of the sensor system used to acquire the data and thereby supports positioning both outdoors and indoors. As illustrated by the three configurations, the HyCon Sensor layer abstraction is quite flexible. In application scenarios that depend on a fully deployed context infrastructure, the HyCon sensor abstraction can still be used as a facade to this implementation in a similar way as the Blip infrastructure has been integrated in the third example above.

IV. APPLICATIONS FOR MOBILE LEARNING

We present in this section our systems as exemplars of our framework and as illustrations of context-aware computing support for mobile learning. Our systems have ranged in platform from Smartphones and tablet PCs, to stationary Smartboards in the school classrooms, and this has of course had implications for design and usability.

We discuss how browsing, searching, linking, annotation, and tagging can be augmented with context-aware techniques in order to support the three aspects of project-based education we outlined in the introduction. The applications that implement these techniques are not "standalone" applications, but rather an integrated set of applications built on top of our framework to support the different aspects of learning in and out of the classroom and collaboration in these different environments.

Project-based education has become increasingly important in Danish schools. By exploring the multi-faceted and multi-disciplined aspects of a theme chosen by the teacher, the students can apply many aspects of learning to synthesize an understanding.

Projects offer the possibility of combining multi-disciplinary in-class education with real-world experiences. By moving some of the education outside the classroom, lessons can be better learned, as they are experienced rather than just read about, e.g., the noise level of construction plant must be experienced in person in order to be appreciated. When learning moves outside, it becomes essential to bring the experience back into the classroom for further work and discussion. Students customarily bring notebooks and perhaps a camera with them

[1] Link: http://www.kh-gps.de/nmea-faq.htm

[2] The BlipNodes and BlipServer are part of a Bluetooth infrastructure developed by BlipSystems (http://www.blipsystems.com).

to document and record their experiences. Later, these experiences can be compiled, processed, and ultimately presented to the teacher and the rest of the class. Thus, project based learning can be described as a series of steps, involving planning, experience, recording, recall, reflection, and presentation. We have sought to support three main activities tied to project based learning with our system, as outlined in the introduction. These activities are:

Browsing with your feet: Much information is situated, i.e., it can be tied to specific locations and circumstances. By moving about in a real landscape, the student can simultaneously be browsing a virtual information landscape in which data pertinent to e.g. the current location automatically presents itself. This enables the presentation of information about the current location, but also opens the possibilities for on-site tests ("how old is the church that you are standing next to?).

Annotating the world: The information encountered above must be added to the system in some way. We have in our work focused mainly on human-entered data rather than automatically generated content (as seen in e.g., GIS). As the students or users of our system move in the world, they are free to inscribe their surroundings with multimedia annotations. This can be used for both factual information ("this church was built in the twelfth century") or digital graffiti ("Anne & Camilla say hi!").

Overview at a glance: While the two former activities support discovery and recording of information, recall is a crucial step in the learning process. Recall is improved by providing a context for the collected artifacts (e.g., notes, images, or other recordings), and this context is in our system provided by associating the knowledge artifacts with the place of creation, typically in the form of an annotated map.

In the following sections we discuss a number of HyCon prototypes that support these learning activities when used in combination during a project.

A. *The HyConExplorer Prototypes*

The mobile browsing and annotation system *HyConExplorer* was the first prototype built with the HyCon framework. Two versions of HyConExplorer have been developed as HyCon Terminal layer applications: one is designed to run on commercially available tablet PCs and the second is designed for Java enabled Smartphones. The goal of the prototypes were to support project based education in elementary schools and in particular, the process of collecting, producing and presenting information from heterogeneous. In order to support these activities, four different context-aware techniques were developed: *Context-aware browsing*, *Context-aware search*, *Context-aware annotation*, and *Context-aware linking*.

All four functions are derived from classical hypermedia techniques, but combined with context-awareness for use in context-aware hypermedia applications for mobile learning.

Context-aware browsing and search support the first aspect of project-based learning (browsing with your feet), by letting the user explore the digital context by changing parameters in the physical context, such as moving from one location to another. Context-aware annotation and linking support the second aspect (annotating the world), by letting users produce information and structures and automatically tag these resources with available context information (i.e., contextual augmentation). The third aspect (overview at a glance) is supported through the HyCon presentation tools and is discussed in Section IV. F.

B. *Context-aware Browsing*

Navigating information resources by browsing is a core feature of most hypermedia systems and a typical way to quickly gain overview of the information space. By using hypermedia links, authors can contextualize information by linking to related material or other material that supports or contradicts the given information.

HyConExplorer applies context-awareness to browsing to let users navigate information stored in the digital context simply by changing parameters in the physical context (e.g., changing location). The prototypes support both implicit browsing of the digital context, where the user initiates browsing by moving around, as well as explicit user controlled browsing. These two ways of browsing can be described by adopting the terms direct physical navigation and indirect representational navigation from geo-spatial hypermedia [19]. Indirect representational navigation is browsing information related to an area without necessarily being present, e.g., by clicking on a map over a city or studying a list of museums. Direct physical navigation on the other hand is linked to time and space, as the information presented to the user is determined by the user’s whereabouts. This approach has found use in a number of tourist guide systems, such as GUIDE [12], where information presented to the user depends on location (which attractions are nearby) and time (which of these are open). The HyConExplorer browser is shown in Figure 7.

Direct physical navigation lets the user browse information by changing parameters in the physical context. Physically walking or driving from one location to another will affect parameters such as time and location in the physical context, allowing the system to find associated information in the digital context. When engaged in this type of browsing, no direct user intervention is required. The users are “browsing information with their feet”, simply moving about in the world. Behind the scenes, the system gathers contextual cues from the physical environment and maps it to tagged information (annotations, linked documents, and guided tours). This kind of browsing is useful especially while the user is in the field using small mobile devices with limited display capabilities and poor support for browsing through large collections (perhaps through multiple layers of menus or lists) of documents.

On the other hand, it is not desirable to require the user to physically move to a given location in order to access information associated with it. Indirect representational navigation allows the user to navigate information associated with remote locations by specifying a virtual location in the system. This creates a “what if I was there?”-scenario, where the system presents the information associated with the virtual context. This corresponds to investigating what Pascoe calls an imaginary world or a pretend context [30]. Systems such as GeoTags[3] and GeoURL[4] also fall into this category by letting users browse pages by their proximity to a given location (without actually being present at the location). In the HyConExplorer pro-

[3] Link: http://www.geotags.com/

[4] Link: http://www.geourl.org/

totype, the user can override the GPS sensor and simply click on the map to specify a virtual location for investigation e.g., when encountering a guided tour in the field, the user can browse through the tour stops without having to physically walk to the individual locations. Thus, the user can investigate the information before deciding whether following the tour is actually worthwhile.

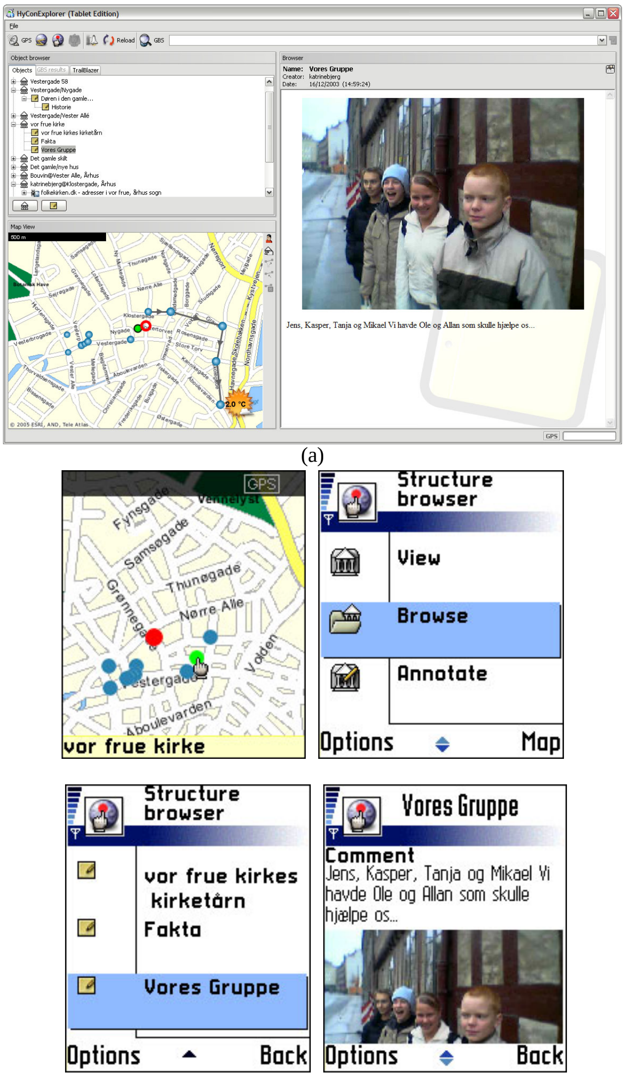
Not all context-aware browsing systems allow both navigation modes e.g., early versions of GUIDE [10] only made information about nearby physical objects available and therefore constrained the users' ability to plan ahead or just browse. Similarly, following links in the Auld Linky server used in the Mack Room experiment [28] required users to actually move from the link source to its destination to follow a link. The HyConExplorer addresses such problems by focusing on (and distinguishing between) the digital and physical context and providing browsing modes for both.

C. Context-aware Search

Context-aware search is based on the general idea of using contextual information as part of the criteria to searches for information. This may help narrowing down the information, so the obtained results are more relevant for the user in the current context. The benefit of adding criteria based on context information is of course highly dependent on the specific search engine in use and how the context data can be formulated as an appropriate criteria.

To mobile users, an inherently interesting part of the context is the location. This single piece of contextual knowledge can be used to locate other resources associated with the location and thus bring location relevant information to the user. Ordinary Web searches based on keywords return results matching the topic of interest, but the pages will typically be completely unrelated to the geographic location of the user. However, many Web sites related to locations (such as landmarks and most businesses) include their postal addresses on their Web pages and this (textual) information is indexed by search engines. Thus, search criteria created as a combination of user supplied keywords and postal addresses can be used to find pages covering a specific topic and are related to a specific location.

The term geocoding refers in GIS systems to the process of assigning a latitude-longitude or (x, y, z)-coordinates to a piece of information, typically an address. Once a coordinate has been assigned, the address can be displayed on a map or otherwise used by the system. The term reverse geocoding refers to the reverse task of getting information about a location given a coordinate. The HyConExplorer prototype supports a novel reverse geocoding technique termed Geo Based Search (GBS). In essence, GBS implements a mechanism to augment search engine queries with information about the user's current location. The goal of GBS is to focus search results to pages covering both a topic of interest (specified by user supplied keywords) and the particular geographical area the user is located in. To our knowledge, GBS was quite novel when it was first implemented in 2002-2003. However, since then our concept has been validated through the success of services such as Google Local Search that have become commonplace. Nevertheless, we will describe the original concept below.



Name: Vores Gruppe
Creator: katrinebjerg
Date: 16/12/2003 (14:59:24)

Jens, Kasper, Tanja og Mikael Vi havde Ole og Allan som skulle hjælpe os...

(a)

Comment
Jens, Kasper, Tanja og Mikael Vi havde Ole og Allan som skulle hjælpe os...

(b)

Figure 7. Browsing trails of annotations and linked Web pages with the HyConExplorer running on the Tablet PC (a) and on the mobile phone (b). Each dot on the map is a link marker to information associated with the physical location.

Raw (x, y, z)-coordinates from GPS sensors are not used directly in GBS, since very few search engines index Web pages by GPS coordinates. Instead, the GBS implementation transforms the GPS coordinates to postal addresses. A database of all public postal addresses in Denmark and their GPS coordinates is freely available from the Danish map provider KMS[5]. Using this database as the basis, mapping raw location information to textual postal addresses provides much more useful input to search engines. The current implementation of the GBS component uses Google as the back-end search engine and utilizes the Google Web APIs. When the user moves from location to location in the physical world the, GPS coordinates are continuously recorded from the GPS sensor. From the coordinates the names of every street within a fifty meters radius are determined and optionally combined with user supplied key words to formulate search strings. The matching Web pages contain the keywords and the postal address printed somewhere in the pages.

[5] KMS: Kort & Matrikelstyrelsen: http://www.kms.dk/

Currently, the first ten search results are returned for each street name with no further filtering. The search results are then plotted onto a map of the searched area to intuitively present the connection between geographical locations and Web pages.

Trying to determine the relationship between a Web page and its geographical location by analyzing the content of the page is what McCurley defines as content-based geo-parsing [27]. This may include determining the language of the page, finding names of geographical sites, telephone numbers, names of events that only occur certain places, names of people and locating them. Geo-parsing the content of Web pages will only be successful if the pages contain location information and can be parsed correctly. As GBS searches for postal addresses printed in the Web pages it naturally falls in the class of such content-based methods. In general, geo-parsing Web pages to determine a geographic association does not provide as much reliability and precision as the approach used by Geotags and GeoURL (mentioned in the previous section), where meta tags defining the precise geographic location are inserted into the Web pages. Including meta data in the resource ensures a very precise and reliable mapping, but relies on the authors of Web pages to embed the meta tags in their documents to enable indexing by the search engine. However, as these standards are currently poorly supported by Web page creators, geo-parsing is a viable alternative and functions especially well in urban areas, where businesses and societies supply the needed information on their Web pages. GBS demonstrated a novel technique of integrating existing unstructured information, which had not in advance been prepared for context-aware applications, into mobile, context-aware system such as HyConExplorer.

D. Context-aware Annotations

The browsing and searching techniques discussed in the previous sections present novel approaches to navigation of the information space based on context filtering. This kind of filtering proves especially useful when accessing information on small handheld devices with limited display and input capabilities such as Smartphones and PDAs. The annotation facilities in HyConExplorer are targeted toward user-created information and positions the prototypes as more than just context-aware browsers or navigation systems, but full context-aware hypermedia systems that support users as active producers of information and structures. Annotations created with the HyConExplorer are associated with context information captured by the client which makes it possible to locate the annotations later by browsing.

Combining mobile devices with built-in cameras and microphones (e.g., mobile phones or our prototype tablet PC) and cheap sensor equipment such as GPS receivers, RFID readers, or 2D barcode scanners (i.e., just the built-in camera in mobile phones) makes it possible to create photos, video, and audio documents and automatically tag them with context information captured by sensors. The annotation facilities in HyConExplorer utilize this combination to support linking of digital annotations to physical locations by linking the annotation documents to location objects.

Annotations can take several forms. A simple annotation consisting of a photo and a link to a given location can be used as a form of "digital graffiti" claiming "I was here" to other visitors later visiting the same location. More elaborate annotations featuring photos, videos, and textual descriptions can be used to document or comment on the location or situation in which they were created. Furthermore, annotations can also be linked to other existing annotations and thus support the creation of entire discussion threads which are anchored in the physical context they are concerned with.

E. Context-aware Linking

The World Wide Web presents a massive corpus of information and much of the information available through the Web may be relevant in a physical context. Thus, supporting links to existing Web pages is an effective means of integrating this information into the system. HyCon implements a link model based on XLink. The model supports simple links (*2*-ary links) linking external documents to objects in the linkbase and extended links (*n*-ary links) linking a collection of objects (typically locations) into a single trail through the objects.

Extended *n*-ary links are used to express trails or paths through a collection of objects. In mobile hypermedia, this typically means trails through a collection of locations, where the link represents a guided tour and the locations represent stops on the tour. So far we have only experimented with static link trails. However, adaptive trails could have some interesting applications as well. Registering the user's movement along the trail in a user model, could be used to gradually present and unfold the trail for the user or present different sub-trails depending on the user's preferences. In a school environment, this could be used when teachers prepare field trips for the children, by creating guided tours through an area the children are to investigate. Each stop on a tour may include a description of the site and additional linked material. This material can be divided into several different layers of information: history, archaeology, politics, environmental info, nature guidance, etc. Upon arriving at a stop the students are presented with one or more of the layers depending on their task and the state of their user model. Furthermore, the students can collect further material at the site and add annotations commenting on the material and on the teacher's predefined material through the linking and annotation mechanisms. At home in the classroom, the trail of information may serve as a way to present the results of the field trip for their classmates. In another scenario, the students may diverge from the predefined trail and create their own trail as they walk along. Other groups of students working simultaneously in the same area may bump into the newly created trail and instantly observe the results of diverging from the teacher's guided tour.

F. HyCon Presentation Tools

The third aspect of project work, *overview at a glance*, is supported by the HyConExplorer presentation module (presented in [4]) and the HyConBoard, HyConFloor, and HyConEditor prototypes illustrated in Figure 8. The HyConBoard and HyConFloor are awareness components designed for the classroom. The prototypes provide real-time overview of online HyCon users and the material they produce. This information is presented on interactive maps on Smartboards (HyConBoard) or projected onto floors (HyConFloor). Users can pan and zoom the map and select material from the map for further investigation. Both prototypes share the same interface (depicted in

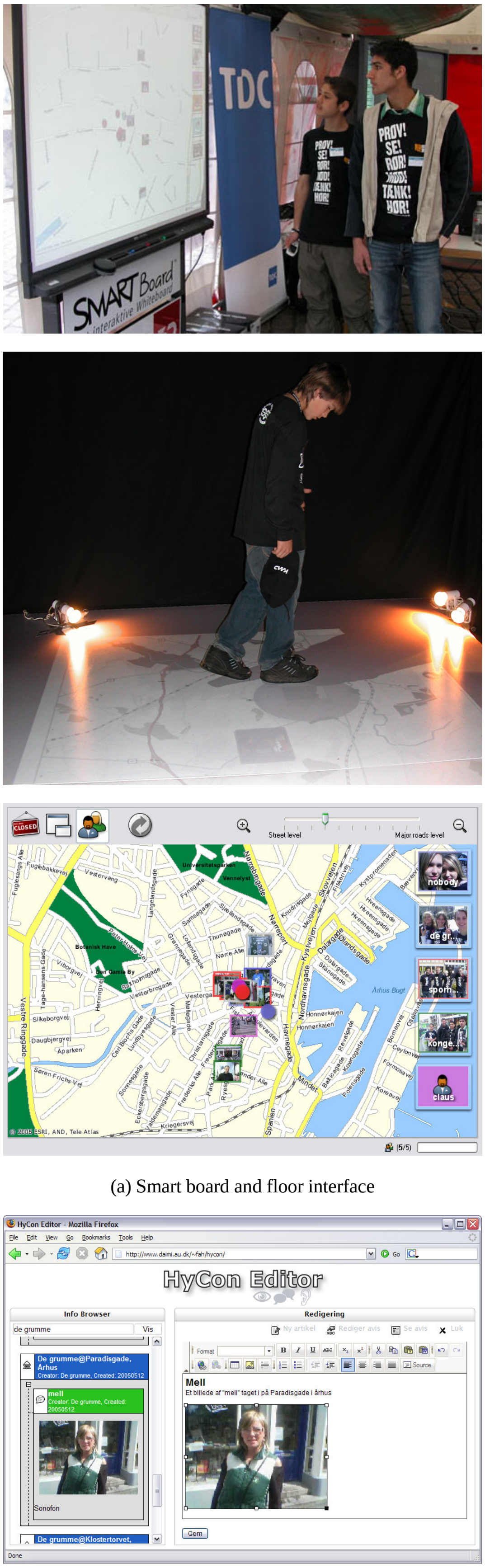
Figure 8. (a)), but while the HyConBoard is controlled through a large, touch sensitive screen, the HyConFloor is controlled by tracking users position on a projected floor with a camera mounted over the floor. The users' positions are indicated by a round, semi-transparent cursor on the map and objects are selected by keeping the same position for a short while. The cursors will then change color to indicate that the user is about to select the object and if she does not move, a mouse click is sent to the object under the cursor. The HyConFloor is developed as part of other "floor interaction" applications in the Center for Interactive Spaces [33].

The HyConBoard and HyConFloor do not rely on sensors but poll the HyCon server for information about online users and their locations. This functionality is provided by the User Tracking Service, implemented atop the location component. The service relies on a number of attributes in the users' profiles: the user's online status, location, and a picture to show on the map. The users' status are assigned a time-to-live (TTL) so users will not appear as online if their location information is not up to date. When users are online, the HyConBoard and HyConFloor queries the annotation service for annotated location and plots available material and the users' position on the map. The HyConExplorer/J2ME provides this information and export the GPS sensor data to the HyCon User Tracking Service.

The awareness components were designed to provide teachers and students in the classroom with an instant, real-time overview of the whereabouts and activities of "field groups". This was used in several evaluations where the students produced online newspapers as part of the projects. Part of the class was appointed to be reporters, and sent out in the field to gather information, while the rest of the class remained in the classroom as part of the editorial team. The editors could follow the activities of field reporters and if dissatisfied with the gathered material they could immediately contact the reporter group and ask them to produce the needed information.

The HyConEditor prototype was also used during these evaluations to manipulate the produced material. HyConEditor is a Web based application supporting the creation of online journals, reports, and newspapers. Journals and journal entries are implemented on top of the annotation service. The HyConEditor implements a simple query interface for annotations and a WYSIWYG HTML editor that supports creation and editing of journal entries (see Figure 8. (b)). Each deployed instance of the editor is associated with a journal and through the interface, students are able to find annotations created by the field groups and write new journal entries based on this material.

Both the awareness components and the editor and journals support overview at a glance, but they differ with respect to temporal presentation: because the HyConBoard and HyConFloor applications are designed as awareness components, other users must be using one of the mobile applications and actively be producing annotations for the board and floor applications to display the users' positions and present newly created annotations. The HyConEditor, on the other hand, functions as a purely virtual browser and editor, that neither requires users nor the author to be co-located with the annotated resources or be working at the same time.



De grumme@Paradisgade, Arhus
Creator: De grumme, Created: 20050512

mell
Creator: De grumme, Created: 20050512

Sonofon

De grumme@Klostertorvet,

Mell

Et billede af "mell" taget i på Paradisgade i århus

(a) Smart board and floor interface

(b) Web authoring interface

Figure 8. Different interfaces for presentation of annotations. All applications are integrated through common services (built on the HyCon framework, in this case) and shared structures.

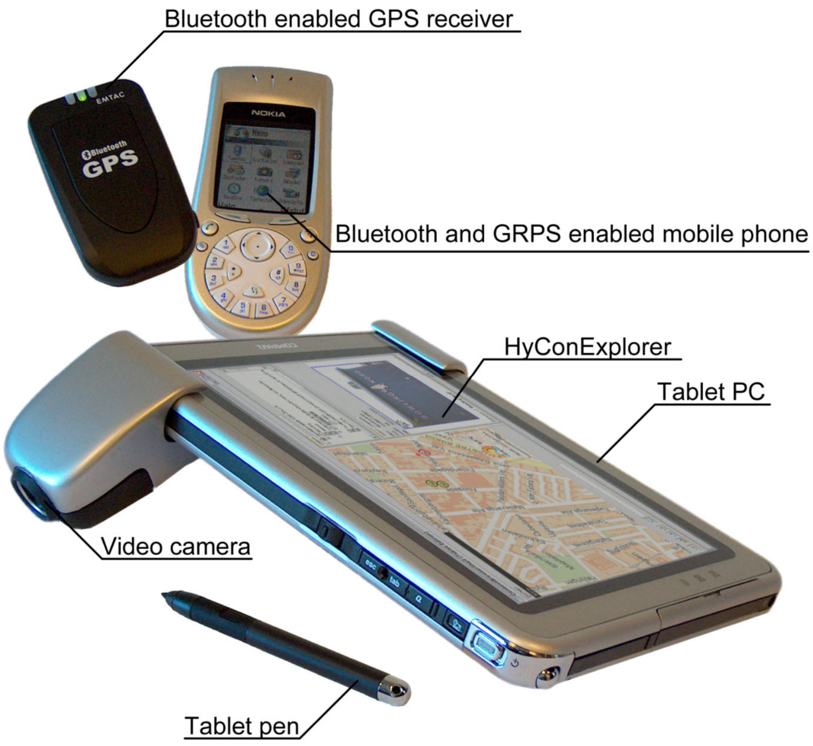

Bluetooth enabled GPS receiver
Bluetooth and GRPS enabled mobile phone
HyConExplorer
Tablet PC
Video camera
Tablet pen

Figure 9. The HyConExplorer hardware setup. All devices are needed for running the Tablet PC version of the system. The J2ME version only requires the Smartphone and GPS receiver.

V. PROTOTYPE IMPLEMENTATION

A. *Requirements for a mobile learning tool*

While supporting the activities described above were the basic requirements for our system, there are also a number of physical requirements and constraints to be considered.

The tools should be portable (i.e., lightweight, sturdy, and battery-powered), usable while standing or walking (eliminating ordinary laptop computers as potential platforms), support browsing (so a display and pointing device is necessary), support text input as well as possible (as text input is cumbersome without a keyboard), handle multimedia input (capturing a photograph or recording sound/video), preferably be connected to the Internet, and able to associate location and data in some fashion.

It became clear to us as we explored these constraints, that it would be difficult to find a single solution to accommodate all needs, as some of these needs are contradicting. High portability implies small size, which restricts screen size and hampers usability. Network connectivity and portability are at odds, as bandwidth will be limited in the field (as well as expensive and battery draining). Cost was also a consideration—ideally, schools should be able equip a number of groups or individual students with the necessary tools.

Since the initial prototypes of 2003, mobile computing has seen a tremendous rate of development, so it is possible today to have small devices that combine reasonable screen size, GPS, 3G connectivity, and full multimedia capabilities. The rate of development makes it likely that future ubiquitous learning applications need not rely on schools handing out specialized expensive mobile handsets to the students, as the students' personal phones will be more than sufficient to handle the requirements.

B. *HyConExplorer Implementation*

The first prototype of HyConExplorer was developed on the Java J2SE platform and was designed to run on commercially available tablet PCs. The second prototype was implemented on the Java micro edition (JME) platform and was designed to run directly on Java enabled Smartphones. The required hardware for both prototypes is illustrated in Figure 9.

The HyConExplorer tablet edition was designed to run on tablet PCs equipped with a mounted camera for capturing low-resolution images, video, and audio, and a Bluetooth enabled GPS unit for recording the user's physical location. The tablet PC prototype implements the full object-oriented HyCon data model. Communication to Server layer services is primarily through the Web Service SOAP interface, with data being sent as XML encoded Java objects. However, upload and download of shared multimedia files are done through the CGI interface. The prototype implements components handling the communication with the server applications, camera data manipulation, and sensor components acquiring location data from Bluetooth GPS receivers. The prototype also implements a map component handling retrieval of bitmap maps from various map providers (e.g., KMS and Esri[6]). Based on the user's position, the chosen map scale, and the geometry of the map view in the application, the map component retrieves maps and displays them in the interface. On top of the map, several layers of hypermedia structures and context information are displayed as link markers and other graphical representations.

[6] Link: http://www.esri.com/software/arcwebservices/

The second generation of HyConExplorer (HyConExplorer/J2ME) was designed to run on a much simpler hardware setup than the tablet PC version, namely directly on Java enabled Smartphones, which communicate with sensor equipment over Bluetooth and features built-in cameras, microphones, etc. The platform chosen for the implementation of HyConExplorer/J2ME was the Java micro edition (JME) with the MIDP 2.0 profile[7]. This platform is well suited for mobile, connected devices and supports network over HTTP, HTTPS, TCP sockets, and serial connections. To take advantage of the Smartphones cameras and Bluetooth implementations, we also required that the actual devices supported J2ME optional packages for Mobile Media[8] (JSR-135), and Bluetooth[9] (JSR-82). A number of devices met these requirements and HyConExplorer/J2ME has been developed and tested on Nokia Series 60 Symbian phones and SonyEricsson K750i phones. At the time of implementation, none of these devices supported Web services (through the J2ME Web services package[10] (JSR-172)) and it was therefore decided simply to communicate with the HyCon Server layer through the servlet CGI interface using a raw XML format instead of the Web service interface used by the tablet PC version. Like the tablet PC version, the HyConExplorer/J2ME also implements the full object-oriented data model, HyCon service components, a map component, and hardware integration components providing access to the Smartphones' camera and Bluetooth communication. As SVG support was quite poor on the J2ME platform, the J2ME implementation of HyConExplorer relies on the native double buffered GameCanvas class included in the MIDP-2.0 Game API.

VI. EVALUATION

We have evaluated our system in an educational setting and we report on our experiences in this section. The evaluation of the educational application HyConExplorer involved a project-based setting as outlined in the introduction.

A. *Field testing*

The HyConExplorer was first evaluated in the context of the NetWorking.Kids research project, headed by Christina Brodersen and Ole Iversen, which ran 2002-2003 [4]. The aim of the project was to enable students at fifth to seventh grade level (11-14 years old) to use technology proactively in the pursuit of project-based learning.

The majority of our tests were performed with a seventh grade class of 20 students from the Katrinebjerg School in central Århus; other schools in Århus were involved to a lesser extend.

We used a number of methods to learn about how teaching was accomplished, including field studies and design workshops with the students. See Iversen's PhD thesis [25] for a thorough discussion of all the aspects of the work with the students.

In order to evaluate how the HyConExplorer matched these requirements, we tested the HyConExplorer with twelve students and three teachers from three school in Århus over a single day. We had prior to this day together with the teachers created a project based on the theme of "Our City", wherein students were to explore parts of Århus, document their findings, and later present them to their class. The students were divided into three groups of four persons. The teachers had, using the tablet based prototype, authored content, questions, and trails for their students to follow. This part of the evaluation went well: the teachers found the supplied tools easy to use and well suited for the task. They used the integrated web browser to locate additional information and further annotate the map and their trails.

With populated trails, we could then turn to the students. All groups had briefly been introduced to the prototype and its use. The groups were each given an area to cover in the city, and were accompanied by one researcher to recording the event and provide tech support, if needed. In order to explore the design space, we supplied one group with the Smartphone HyConExplorer/J2ME, and two groups with the tablet based version.

Using their allotted equipment, the groups proceeded to explore their part of the city. They made annotations, recorded interviews, and took pictures. On average, each group created five annotations. When they returned to the classroom, they used their gathered material to create more comprehensive presentations. In these, they would often expand on brief notes taken in the field that evidently served more as placeholders or reminders. The two groups with tablets continued to use their tablets, while the group with a Smartphone exported their collected material to a PC and used Powerpoint for their presentation. As all gathered information in HyConExplorer is geo-tagged, it was easy for the tablet users to recall the context of each collected data point. While the students found the software part of the HyConExplorer easy to use, there were some concerns over the physical aspects of the prototype. The tablet version of HyConExplorer consists of several separate devices, namely a tablet PC, a mobile phone for internet connectivity, and a GPS receiver for position. In contrast, the Smartphone version has only the mobile phone and the GPS receiver. As the GPS receiver is quite small, it can easily be carried together with e.g. the mobile phone. The students often divided the carrying among themselves, so that one would carry the tablet PC and another would carry the rest. If they moved too far apart, it was no longer possible to maintain a Bluetooth connection, and the Internet connectivity would falter. Additionally, the children found the tablet PC to be heavy and cumbersome, susceptible to rain, awkward to write or type on, and the Internet connection too intermittent. The Smartphone version received far fewer criticisms, as it was light, provided its own Internet connection, and was (to our surprise) not a text entry challenge to the texting savvy youths.

Based on the experiences we gained from the tests we found that the tablet PC software works well as a tool for the teachers in preparing and organizing information for the field trip and as a context-aware presentation tool that can be used by the students when they return back to the classroom. In the classroom, the overview and structuring mechanisms in the HyConExplorer are more important than the size and weight of the hardware; it is vital for the students' ability to assess their collection of project material. There is no doubt that the tablet provided a superior overview compared to the Smartphone's relatively small screen, which was also reflected in the teacher's clear

[7] JME: http://java.sun.com/javame

[8] JSR-135: http://www.jcp.org/en/jsr/detail?id=135

[9] JSR-82: http://www.jcp.org/en/jsr/detail?id=82

[10] JSR-135: http://www.jcp.org/en/jsr/detail?id=172

choice of the tablet over the Smartphone interface and the students' rejection of the Smartphone prototype when preparing their presentation. In the field, however, the simple phone setup turned out to be a much more elegant and usable solution than the tablet PCs. The strength of using a GPS-device is that the prototype automatically supplies the user with information relative to where they are; when the students moved through town and found an interesting spot, the HyConExplorer Smartphone prototype showed them the map and annotations centered around their current location. In the field, location takes priority over overview and the evaluation clearly supports this finding.

B. *Implications for learning*

From a learning perspective, the HyConExplorer prototypes provide both teachers and students with powerful tools for teaching and learning in the real world. The HyConExplorer prototypes provide teachers and students with means for extending the classroom into the real world as well as bringing the real world back into the classroom. Students should be taught in advance of their expedition, so that they can use this knowledge to inform them as they collect evidence and other data in the field, and finally, in the classroom, use their gained knowledge and data to form further hypotheses and present their findings.

Project work is supported on several different levels: taking advantage of the rich contextual information that is the real world; providing easy access to information related to the location you are currently in; easy capture of rich media (pictures, video, audio, text) annotations that are tied to your current location; providing an overview of the path you have traveled and the annotations created in the process, and finally the means to reflect on the experience and tie it all together in a presentation for the class. The HyConExplorer prototypes elegantly save collections of material that must ordinarily be retrieved from disparate sources and help the students in tying them together through the context in which they were experienced.

An interesting area, which we have not had the opportunity to explore is the interweaving of generated data over time – i.e., how would the accumulated data from previous years inform the students as they roamed the field? The ease with which digital artifacts can inscribed would lend itself to a continually expanding set of knowledge and experiences – something that would be more difficult where the knowledge artifacts in question physical rather than digital.

VII. RELATED WORK

Many systems have been developed to explore the possibilities of associating location (and more generally, context) to data, user generated or otherwise. Most of these systems have not been explicitly developed for learning purposes, yet they all have characteristics that are relevant and can inform future design of context or location aware systems.

Cyberguide [2][26] was one of the first projects to support both indoor and outdoor information browsing with a mobile, context-aware system. The aim of the Cyberguide was to replicate a human tour guide service through use of mobile and handheld technology and ubiquitous positioning and communication services. Several tour guide prototypes were built for environments within and around the Georgia Tech campus, e.g., guided tours of the laboratories in the Georgia Tech GVU center using an infrared positioning system. Another prototype, called CyBARguide, was running on a Newton MessagePad equipped with a GPS receiver. With the CyBARguide users could locate points of interest that satisfied certain requirements (special offers, free parking, good ambiance, etc.). Cyberguide and CyBARguide were context-aware browsers with no support for user created information. However, the authors acknowledge the need for this kind of functionality and propose future support for a user modifiable information base. Abowd et al. [2] suggest that users should be able to leave comments and impressions for later visitors as "virtual graffiti" when visiting a location. This is similar to creating an annotation on a visited location in the HyConExplorer.

The more recent GUIDE project [12] was similarly developed as a context-aware tour guide and recommender system. Running on a handheld PC, GUIDE provided tourists with context-aware hypermedia information while they were visiting the city of Lancaster in England. Information could be related to the visitors current location or be links to nearby sites that might be of interest. The GUIDE system calculates its position relative to WiFi 802.11 base stations placed at Lancaster attractions. This makes for relatively coarse resolution, but provides high-speed data connection at the hot spots. The GUIDE system has been favorably tested in real use [11], supporting the concept of electronic tourist guides. However, like the Cyberguide, GUIDE only provides browsing capabilities and does not support users to actively create or structure tours and annotations while in the field.

Another project to employ WiFi positioning is Geonotes [17][32]. Geonotes is not designed as a guide system, but as a mass-scale location based annotation system that allows anybody to produce geo-spatial anchored notes. A main concern in the Geonotes project was aspects of social filtering. Realizing that a public annotation space would quickly become crowded and overloaded with information (e.g., spam or commercial advertisement) the designers introduced several collaborative filtering techniques in the system. Each note in the system is assigned metadata describing the popularity of the node, e.g., the number of comments on a note, the number of readings, and the age of the node. This data can be used to provide overview at a glance by only presenting the most popular notes at a given location. In addition, users can invite other users to be "Geonotes friends" to easily share notes among each other. The system supports both pull and push mechanisms for retrieving notes. Text search can be used to retrieve (or pull) notes from the Geonotes server which include specific words or phrases. Searches can also be saved as persistent queries that are continuously performed as the user moves about. When notes match a query, the user is notified with an audible alert. In addition to the WiFi positioning mechanism, Geonotes also allows users to manually specify an extra place label—a user defined location specification that describes the anchor point of the annotation. Even though the place label has to be supplied and interpreted by the user, it gives a much better description of the anchor point than the coarse grained measurement from the WiFi positioning system alone. Geonotes is a fine example of a geo-spatial annotation system with advanced filtering mechanisms. However, the systems services are built around a geographical data model, which may make it difficult to use the system for other context-aware applications that do not solely depend on location.

The Stick-e note architecture [6] is a framework developed at the University of Kent. It is not based on a distributed infrastructure like the previous systems, but aims at providing mechanisms for specifying relationships between application data and context information. The Stick-e note metaphor has been used intensively in mobile annotation systems designed for fieldwork [31]. Two field work applications were built on the Stick-e Notes platform: an application for giraffe observation [31] and a context-aware archaeological assistant [36]. The primary concerns for these applications were robustness of both devices and software while in the field and support for contextual augmentation i.e., automatic gathering and tagging of context information to collected material. Pascoe et al. [31] report that contextual augmentation proved to be almost indispensable during a giraffe observation in Kenya where a researcher recorded data (vegetation surveys and giraffe behavior observations) at a much faster rate than would have been possible using a traditional manual recording medium. Connectivity is also listed as an important aspect of the fieldwork applications. However, since no network infrastructure was readily available on the African savanna and the applications were developed on 1998 PalmPilots with no high-speed network support, data was simply collected on the device and later transferred to a PC back in the research laboratory. Thus, connectivity did not provide real-time collaboration support or sharing of material with co-workers while in the field, as supported by the HyCon framework and applications. But the connection to the PC allowed data to be transferred to a home-base repository that contained all collected material. When the researchers went back into the field, a subset of the home-base repository could be transferred back onto the PalmPilots to a project-dependent repository small enough to fit in the limited memory of the devices. In the field, notes from the project-dependent repository could be triggered and presented when the user's context matched the trigger conditions in the notes.

The Ambient Wood [35][42][43] also focuses on fieldwork, and in particular on supporting school children's education out of the classroom. The project aims at augmenting school children's learning experience in a specific woodland setting, by augmenting the wood with various sensors (RFID readers and location systems based on hidden radio transmitters) and actuators (speakers hidden in the trees). The system is based on a MUD (Multi User Dungeon) with "rooms" corresponding to places in the wood where radio transmitters (pingers) are placed and "players" corresponding to the groups of children in the wood. When the children enter a region in the wood, and are tracked by the pingers, the players position in the MUD is updated to the corresponding room, and the MUD issues an event that sets off audio playback on speakers close to the children. The children also carry moisture and light probes which can be used for taking readings from the environment. Such readings result in presentation of tailored information on the students' handheld computers. All events (the location of the children, probe readings, audio playback, etc.) are recorded in a system log for later reflection. The system also supports the creation of scenarios, which can be played out in the woodland. The scenarios contain pre-authored events that are triggered as the children move about. For example, to provoke further exploration of a site, the children could hear sounds from the hidden speakers and at the same time be presented with information on the handheld computer about specific kinds of animals that live in the grass nearby. After finishing a field trip, the students' exploration can be compiled into field trip journals based on the recorded data. However, unlike the journals created with the HyCon tools, the Ambient Wood journals are created from pre-authored snippets of text that are combined to reflect the specific path the children took through the wood. Thus, rather than trying to support the students as *producers* of material in the field, the Ambient Wood aims at orchestrating an ambient experience environment that provoke discussion and exploration of the surroundings. In evaluation, the authors found that the Ambient Wood system enabled children to connect class room theory with the field experiences. They could form and test hypotheses in the field based on their prior knowledge as well as knowledge imparted to them by the system in the field. Later, in the class room, their journals formed a solid basis for recollection and further work. Thus, the Ambient Wood system demonstrates the possibilities of ubiquitous learning to enhance the students' experience and learning process.

VIII. CONCLUSION

We have herein described the evolution of the HyCon platform, a framework designed to enable the seamless authoring and browsing of contextual information. The framework has explored the meaning of context and provides today a highly flexible and extensible context model as well as a multi-platform set of applications, ranging the very small (mobile phones) to the very large (large floor and wall displays).

The HyCon platform provides teachers and students with an extensible tool set that can support them in the planning, exploration, capture, recall, and presentation of project based education. It has been used in teaching settings on several occasions, and has been shown to work well.

A lesson of our evaluations has been that of choosing the correct tool for the job: when in a mobile setting, portability is key, whereas other settings (such as recall or presentation) favor larger devices. Still, across the various platforms supported by HyCon, the validity of relying on context as an anchoring point for browsing and authoring information has been shown.

The concept of using context as a mechanism for filtering through large data sets has since the conception of the HyCon platform in 2003 become widespread, and Smartphones now commonly feature mechanisms to provide information about the user's immediate vicinity. However, just as the Web moved from being largely a read-only media to today's user contributed "Web 2.0", so geo-based systems should allow its users to add content as they move through the world.

Learning does not occur in a void: it is by its very nature contextualized, and it is by linking and applying new knowledge and experience with the context of what is already known that learning is achieved. The strength of project-based learning is that by broadening the context of learning from one to several disciplines, the lessons learned can illuminate more.

ACKNOWLEDGEMENTS

This work was supported by ISIS Katrinebjerg, Center for Interactive Spaces. The authors wish to thank all our center colleagues. We also wish to thank all our HyCon collaborators over the years, in no alphabetical order: Christina Brodersen, Bent Christensen, Kaj Grønbæk, Ole Iversen, and Torben Pedersen.

REFERENCES

[1] Abowd, G. D. 1999. Software engineering issues for ubiquitous computing. In *Proceedings of the 21st international conference on software engineering* (pp. 75–84). IEEE Computer Society Press.

[2] Abowd, G. D., Atkeson, C. G., Hong, J., Long, S., Kooper, R., and Pinkerton, M. 1997. Cyberguide: a mobile context-aware tour guide. Wireless Networks, 3(5):421–433. (doi:10.1023/A:1019194325861)

[3] Bernstein, M. 2002. Storyspace 1. In Proc. of the 13th ACM Hypertext Conference, pages 172–181, College Park, Maryland, USA, 2002.

[4] Bouvin, N. O, Brodersen, C., Hansen, F. A., Sejer, O. I., and Nørregaard, P. 2005: Tools of Contextualization: Extending the Classroom to the Field. In *Proc. of the 4th Int. Conf. for Interaction Design and Children (IDC 2005)*, June 2005, Boulder, Colorado, USA.

[5] Bouvin, N. O., Christensen, B. G., Grønbæk, K., & Hansen, F. A. 2003. HyCon: A framework for context-aware mobile hypermedia. *The New Review of Hypermedia and Multimedia*, 9, 59–88. (doi:10.1080/13614560410001725310)

[6] Brown, P., Bovey, J., and Chen, X. 1997. Context-aware applications: from the laboratory to the marketplace. IEEE Personal Communications, 4(5):58–64, Oct. 1997. (doi:10.1109/98.626984)

[7] Brusilovsky, P. 2001. Adaptive hypermedia. User Modeling and User-Adapted Interaction, Ten Year Anniversary Issue 11p:87–110.

[8] Brynskov, M., Kristensen, J. F., Thomsen, B., and Thomsen, L. L. 2003. What is context? Technical report, Department of Computer Science, University of Aarhus. Available from: http://www.daimi.au.dk/brynskov/publications/what-is-context-brynskov-et-al-2003.pdf.

[9] Chen, G. and Kotz, D. 2000. A survey of context-aware mobile computing research. Technical Report TR2000-381, Dept. of Comp. Sci., Dartmouth College, Nov. 2000.

[10] Cheverst, K., Davies, N., Mitchell, K., and Efstratiou, C. (2001). Using context as a crystal ball: Rewards and pitfalls. Personal and Ubiquitous Computing, 5(1):8–11. (doi:10.1007/s007790170020)

[11] Cheverst, K., Davies, N., Mitchell, K., Friday, A., and Efstratiou, C. 2000. Developing a context-aware electronic tourist GUIDE: some issues and experiences. In Proceedings of the SIGCHI conference on Human factors in computing systems, pages 17–24. ACM Press.

[12] Cheverst, K., Mitchell, K., and Davies, N. 2002. The role of adaptive hypermedia in a context-aware tourist GUIDE. Communications of the ACM, 45(5):47–51. (doi:10.1145/506218.506244)

[13] Delisle, N. M. and Schwartz, M. D. 1987. Contexts-a partitioning concept for hypertext. ACM Transactions on Information Systems, 5(2):168–186, 1987. (doi:10.1145/27636.27639)

[14] Dey, A. K. 1998. Context-aware computing: The CyberDesk Project. In Proc. of the AAAI 1998 Spring Symp. on Intelligent Environments (AAAI Tech. Report SS-98-02), pages 51–54, Stanford, CA, 1998.

[15] Dey, A. K. 2001. Understanding and using context. Personal and Ubiquitous Computing, 5(1):4–7, 2001. (doi:10.1007/s007790170019)

[16] Dey, A. K., *et. al* 2001. A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. *Human-Computer Interaction (HCI) Journal*, 16(2-4):97–166., 2001.

[17] *Espinoza, F.*, et al. 2001. Geonotes: Social and navigational aspects of location-based information systems. In *Proc. of the 3rd international conference on Ubiquitous Computing*, pages 2–17, London, UK. Springer-Verlag.

[18] Grønbæk, K., Sloth, L., and Bouvin, N. O. 2000. Open hypermedia as user controlled meta data for the Web. In Proc. of the 9th Intl. World Wide Web Conference, pages 553–566, Amsterdam, Holland, May 2000.

[19] Grønbæk, K., Vestergaard, P. P., and Ørbæk, P. 2002. Towards geo-spatial hypermedia: Concepts and prototype implementation. In Anderson et al. (2002), pages 117–126.

[20] Halasz, F. G. and Schwartz, M. 1994. The Dexter hypertext reference model. Communications of the ACM, 37(2):30–39. (doi:10.1145/175235.175237)

[21] Hansen, F. A. 2005. Representing Context in Hypermedia Data Models. In Proc. of the Intl. Workshop on Context in Mobile HCI, Salzburg, Austria, Sep. 2005. Available from: http://mobilehci.icts.sbg.ac.at/context/papers.htm.

[22] Hansen, F. A. 2006. Ubiquitous annotation systems: technologies and challenges. In *Proceedings of the 17th Hypertext Conference* (Odense, Denmark, August 22 - 25, 2006). ACM Press.

[23] Hansen, F. A. and Grønbæk, K. 2008. Social Web Applications in the City: A Lightweight Infrastructure for Urban Computing. In *Proceedings The Nineteenth ACM Conference on Hypertext and Hypermedia*, Pittsburgh, PA, USA 19 - 21 June, 2008.

[24] Hansen, F. A., et al. 2004. Integrating the Web and the World: Contextual trails on the move. In *Proc. of the 15th ACM Hypertext Conf.*, pages 98–107, Santa Cruz, CA, USA.

[25] Iversen, O.S. 2006: Participatory Design beyond Work Practices - Designing with Children, Ph.D. dissertation, Dept. of Computer Science, University of Aarhus. Available from: http://www.daimi.au.dk/~sejer/Publications%20_files/dissertation.pdf

[26] Long, S., Kooper, R., Abowd, G. D., and Atkeson, C. G. 1996. Rapid prototyping of mobile context-aware applications: the Cyberguide case study. In Proceedings of the 2nd annual international conference on Mobile computing and networking, pages 97–107. ACM Press.

[27] McCurley, K. S. 2001. Geospatial Mapping and Navigation of the Web. In Proceedings of the 10th International World Wide Web Conference, pages 221–229, Hong Kong. W3C, ACM Press.

[28] Millard, D. E., Roure, D. C. D., Michaelides, D. T., Thompson, M. K., and Weal, M. J. 2004. Navigational hypertext models for physical hypermedia environments. In Proceedings of the 15th ACM conference on Hypertext & hypermedia, pages 110– 111. ACM Press.

[29] Millard, D. et. al. 2002. Beyond the traditional domains of hypermedia. In the Workshop on Open Hypermedia Systems, pages 26–32, College Park, MD, 2002.

[30] Pascoe, J. 1997. The stick-e note architecture: extending the interface beyond the user. In Proceedings of the 2nd international conference on Intelligent user interfaces, pages 261–264. ACM Press.

[31] Pascoe, J., Ryan, N. S., and Morse, D. R. 1998. Human Computer Giraffe Interaction: HCI in the Field. In Johnson, C., editor, Workshop on Human Computer Interaction with Mobile Devices, GIST Technical Report G98-1. University of Glasgow. Available from: http://www.cs.kent.ac.uk/pubs/1998/617.

[32] Persson, P., Espinoza, F., and Cacciatore, E. 2001. GeoNotes: social enhancement of physical space. In CHI '01 Extended Abstracts on Human Factors in Computing Systems, pages 43–44. ACM Press.

[33] Petersen, M. G., Krogh, P. G., Ludvigsen, M., and Lykke-Olesen, A. 2005. Floor interaction hci reaching new ground. In CHI '05: CHI '05 extended abstracts on Human factors in computing systems, pages 1717–1720, New York, NY, USA. ACM Press. (doi:10.1145/1056808.1057005)

[34] Reich, S., Wiil, U. K., Nürnberg, P. J., Davis, H. C., Grønbæk, K., Anderson, K. M., Millard, D. E., and Haake, J. 1999. Addressing interoperability in open hypermedia: the design of the open hypermedia protocol. The New Review of Hypermedia and Multimedia, 5:207–248. (doi:10.1080/13614569908914714)

[35] Rogers, Y., Price, S., Randell, C., Fraser, D.S., Weal, M., and Fitzpatrick, G. 2005. Ubi-learning Integrates Indoor and Outdoor Experiences. Communication of the ACM, 48:1:2005, pp. 55-59.

[36] Ryan, N. S., Pascoe, J., and Morse, D. R. 1998. Enhanced reality fieldwork: the context-aware archaeological assistant. In Computer Applications in Archaeology, Oxford, Oct. 1998.

[37] Schilit, B. N. and Theimer, M. 1994. Disseminating active map information to mobile hosts. IEEE Network, 8:22–32, 1994. (doi:10.1109/65.313011)

[38] Schilit, B., Adams, N., and Want, R. 1994. Context-aware computing applications. In Proc. of the Workshop on Mobile Computing Systems and Applications, pages 85–90, Santa Cruz, CA, USA, Dec. 1994.

[39] Schmidt, A., *et. al.* 1999. Advanced interaction in context. LNCS, 1707, 1999.

[40] Voelker, G. M. and Bershad, B. N. 1994. Mobisaic, an information system for a mobile wireless computing environment & engineering. In Proc. of the Workshop on Mobile Computing Systems and Applications, pages 85–90, Santa Cruz, CA, USA, Dec. 1994.

[41] Want, R. *et. al*. The active badge location system. Technical Report 92.1, Olivetti Research Ltd, Cambridge, UK, 1992.

[42] Weal, M. J., Michaelides, D. T., Thompson, M. K., and DeRoure, D. C. 2003a. The AmbientWood Journals: Replaying the Experience. In Proceedings of the 14th ACM conference on Hypertext and hypermedia, pages 20–27, New York, NY, USA. ACM Press.

[43] Weal, M. J., Michaelides, D. T., Thompson, M. K., and Roure, D. C. D. 2003. Hypermedia in the Ambient wood. Hypermedia, 9(1):137–156. (doi:10.1080/13614560410001725347)

[44] Weiser, M. (1991, February). The computer for the 21st century. *Scientific American*, 265(3), 66–75.

[45] Wiil, U. K., Hicks, D. L., and Nürnberg, P. J. 2001. Multiple open services: A new approach to service provision in open hypermedia systems. In Davis et al. (2001), pages 83–92.

[46] Yankelovich, N., Haan, B. J., Meyrowitz, N. K., and Drucker, S. M. 1988. Intermedia: the concept and the construction of a seamless information environment. Computer, pages 81–96, January 1988. (doi:10.1109/2.222120)

AUTHORS

Frank Allan Hansen is Post. Doc. at Aarhus University (Center for InteractiveSpaces, Department of Computer Science, email: fah@cs.au.dk). He received his PhD and MS degrees in computer science from Aarhus University in 2006 and 2003, respectively.

Niels Olof Bouvin is Associate Professor at Aarhus University (Center for InteractiveSpaces, Department of Computer Science, bouvin@cs.au.dk). He received his PhD and MS degrees in computer science from Aarhus University in 2001 and 1996, respectively.

Manuscript received 13 December 2008. Published as submitted by the authors.