Co-creation in Ambient Narratives

Mark van Doorn[1] Arjen P. de Vries[2]

[1] Philips Research,
Prof. Holstlaan 4,
5656 AA Eindhoven, The Netherlands
mark.van.doorn@philips.com
[2] Centre for Mathematics and Computer Science,
Kruislaan 413,
1098 SJ Amsterdam, The Netherlands
arjen.de.vries@cwi.nl

1 Introduction

Ambient Intelligence [1] aims to improve the quality of people's life by making everyday activities more convenient and enjoyable with digital media. Technically, Ambient Intelligence refers to the presence of a digital environment that is sensitive, adaptive, and responsive to the presence of people. Electronic devices are embedded in furniture, clothing or other parts of the environment; the technology recedes into the background of our everyday lives until only the function (i.e., the user interface) remains visible to people. At the same time, the human moves into the center of attention, in control of the devices around him. These work in concert to support the performance of everyday activities in an intelligent manner.

Producing such smart ubiquitous computing environments on a large scale is problematic however. First, it is technologically not possible in the near foreseeable future to mass produce a product or service that generates Ambient Intelligence, given the current state-of-the-art in machine learning and artificial intelligence. Second, it is economically not feasible to manually design and produce Ambient Intelligence applications for each person individually.

So the question becomes, how can we customize Ambient Intelligence environments on a mass basis to achieve economies of scale as well as scope?

To address this research question we iterate between a top-down and bottom-up approach. There are several reasons for this. Over the past years, many Ambient Intelligence prototypes and scenarios have been developed and written. Unfortunately, there are so many examples and there is so much variety in these examples that it is very hard to generalize from these examples. There is also no clear definition of what Ambient Intelligence is or how it relates to other forms of media. In the first part of this article we therefore examine important social, economical and cultural trends and derive the notion of *ambient narratives* as an organizing concept from this analysis. In the second part, we discuss the design and implementation of an ambient narrative system in detail and report on the results we obtained from experiments.

Part I

2 Disembodied Intelligence

Basically, there are two high-level strategies towards Ambient Intelligence, illustrated in Figure 1. In the Ambient Intelligence inside approach the application intelligence is embedded in a tangible, physical product either in hardware or as embedded software on hardware. This strategy is a product-centric approach: The device generates the experience. The Ambient Intelligence outside approach starts from the desired experience: Devices are the means through which the desired experience is conveyed to the end-user. Furthermore, in this approach the application intelligence is disembodied from the tangible, physical product and offered as an information product or service to the end-user. In both cases the application intelligence can be personalized to the end-user. The differences between the two approaches become clear if we look at the underlying economics.

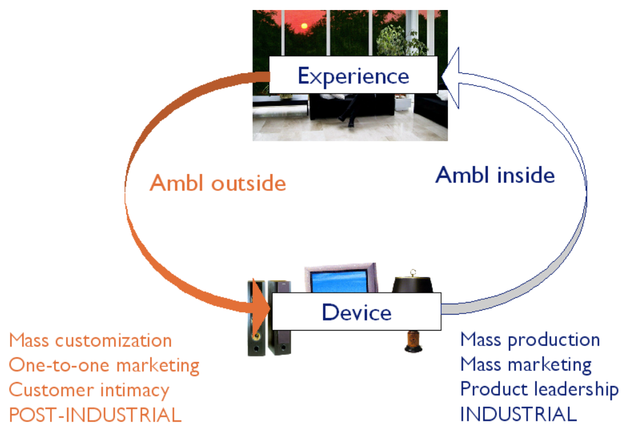

Fig. 1. Two different strategies

The product-centric Ambient Intelligence inside approach is the dominant, industrial strategy towards realizing the ambient intelligence vision. Market studies and surveys are conducted to define products that are targeted to specific customer segments. Products are mass-produced and delivered via large distribution and retail networks to the customer. The focus of companies that follow this Ambient Intelligence inside approach is on achieving product leadership in one or more product categories. With their industrial mindset, these companies are often forced to view Ambient Intelligence as a new and complex physical product or range of cooperating products that are mass produced and brought to the market by means of mass marketing and mass production techniques.

This mode of operation is increasingly under pressure. Over the past decades the needs and wants of individual people have become less and less homogeneous and stable and as a result of high-speed changes in society and technology, the pace of life also greatly increased; our relationships with people, places, things, organizations and ideas have shortened in duration in the past decades [39]. Power is also shifting from companies to individuals, today customers are no longer at the end of the value chain; they are at the beginning. In such a rapid-changing and demand-driven society, customers demand personalized products and services at any time so producers who can customize their products and services in real time will have a decisive advantage over producers who cannot deliver in real time [19]. Physical product manufacturers are constantly trying to reduce time-to-market to deal with this reality, but eventually only information products that have zero mass and do not occupy space can be custom produced and delivered in real time. With products and services becoming commoditized quickly, innovation becomes all the more important. But as the speed of innovation can be much higher for information goods than for physical products, product innovation is easier if the product or service is informational or as much informational as possible (as not all products can be turned into digital information goods).

In this context, the demand-driven, informational Ambient Intelligence outside strategy is a better fit than the supply-driven and product-centric Ambient Intelligence inside strategy. This leads us to the first requirement: *Ambient Intelligence is a personalized information good delivered over a set of networked devices which surround the customer(s). The intelligence that controls the appliances is separated from the ambience and customized on-demand by individual customers.*

3 Experience Co-creation

The rising affluence and growing transience of people, places, things and ideas that created the material basis for todays global, informational networked society led to another, even more profound change. This change is however more subtle and occurs deep inside people's minds but affects the economy at large: People's existential view towards life is shifting from purely external oriented to more and more internal oriented [33]. To explain this difference it is best to give an example. If we are external oriented, we focus on the effect on the outside world. Driving a car to go from home to work is an example. If we act in an internal oriented way, the effect on ourselves, the resulting experience is important: We want a good driving sensation. This situation management with the goal to affect one's own inner life is a crucial change from the past in which most people were only concerned with survival or acquiring higher social (economical) status. Life may seem easy if everything is available in abundance, but the difficulty of biological survival is replaced with the question of "What do I want next?". To support people in this need companies are changing their focus from manufacturing products and delivering services to staging memorable experiences and transformations.

The actors in the experience economy, experience consumers and producers, each have their own rationality type, i.e. a collection of action strategies with constant, repeating goals. The consumer acts in a purely internal oriented way: Experience consumers actively work to regulate the constant stream of experiences they experience. The experience consumer rationality can be summarized by the following principles [33]: Correspondence, abstraction, accumulation, variation and autosuggestion. The correspondence principle states that people do not choose an experience at random, but that they choose experiences that they connect with their own style. Abstraction means that experience consumers do not look for the optimization of a single experience offering but the optimization of a flow of experience offerings. Accumulation refers to the fact that people have the tendency to increase the tempo in which they consume experiences and start to collect, pile experiences. To still feel a nice stimulus after each accumulated experience, people look also for variation. This variation is often restricted to a common frame: e.g. people change bars but not do change the bar scene. Autosuggestion finally refers to the fact that people have a strong need to ask other poeple in the same environment what they felt of a particular experience being offered in the absence of any clear quality criteria to measure the value of an experience. In stark contrast to the internal oriented rationality type of the consumer is the external oriented rationality type of the experience producer. Experience producer try to match the action patterns of the experience consumers as best as possible with their strategies. They have developed a strong (traditional) outside oriented rationality type. Their rationality type can be characterized with the following action strategies: Schematization, profiling, variation and suggestion. Schematization refers to the fact that demand for experience can be categorized in aesthetic schemas (coherent style groups), which are common in large groups of the population and stay stabile over the years. Segmentation of the market according to aesthetic schemas alone is not sufficient, companies must also profile themselves to make themselves known by means of elaborate brand management campaigns (profiling). Since experience consumers demand variety and expect the experience to have something new, something more stimulating, the experience producer must be able to customize the experience on a mass, low-cost basis (variation). Finally, providers of experiences must work hard to raise the suggestion that their product is new or authentic (suggestion).

To better understand what experiences are, we can look at the behavior of actors, consumers and producers in the experience economy and society at large or look at the formation of individual experiences. Eventually, the behavior of the actors in the experience economy at large must emerge from how individuals form experiences in their mind. Basically we can view the individual as a free subject that interacts with its environment or as an object that is exposed to environments that impose meaning. In reality both views are true at the same time as each individual is both subject and object at the same time. Experiences can therefore be said to form in a subject-determined, reflexive and involuntary way. The analysis of how experiences are formed can be approached from two different angles:

- The subject-oriented viewpoint puts the performance of the self in the foreground. Performance is the main object of study.
- The object-oriented viewpoint focuses on the production of signs and meanings that are encoded in for example a written or audio-visual language by an author. The main object of study is text (in the broad sense of the word as we will explain).

3.1 Subject-oriented

The word performing is so commonly used in our language that we forget that performances are pervasive to every culture. Performances are not just seen on in the theater, music, dance and performance arts in general but also in our everyday lives: We perform the role of a father or son in our private lives but maybe also that of a doctor, judge or police agent for example in our professions. Performance studies is an emerging yet wide-ranging interdisciplinary field that takes performance as the organizing concept for the study of a wide range of human behavior. It embraces research in the social sciences from anthropology to theatrical studies. Because performances vary so widely from medium to medium and culture to culture, it is hard to pin down an exact definition for performance. Schechner defines performance as ritualized behavior conditioned/permeated by play or twice-behaved behavior [32]. When people are performing, they show behavior that is at least practiced once before in a similar manner. In traditional performance arts this behavior can be detected easily: Actors in a theater play, opera or movie rehearse their roles off-stage and repeat this behavior when they are on stage. But this twice-behaved behavior can also be seen in a priest conducting a wedding ceremony, a surgeon operating on a patient or a McDonalds service employee behind the counter. Even in our own homes, people show signs of this repeated behavior. This happens for example during everyday rituals, like brushing your teeth in front of a mirror in the morning, watching a soccer match with friends, or, coming home from work in the evening. Note that, here, the sending and receiving party in a performance may be the same. These kinds of everyday life performances where already mentioned by Goffman [14], who wrote in his book the Presentation of Self in Everyday Life about how people follow culturally specified social scripts that interact with each other. These social scripts may differ from culture to culture and from epoch to epoch, but no culture exists without social scripts according to Goffman. People are performing all the time, most of the time without knowing. Each time we perform one of these social scripts, we form a new experience. These new experiences can also be shared with everybody else present as in interactive theatre or live action role playing games. Here the audience actively participates in the performance or play, following certain predefined/rehearsed roles and rules that are combined to create a novel experience each time the play is performed. The experience is co-created by the actors and the audience. This immediacy of the action creates highly immersive and engaging experiences [12].

Viewing life as social theatre is interesting for us for two reasons: First, if people behave according to social scripts, we may succeed in codifying interactive media applications to support people in carrying out these scripts. Just as lighting and sound effects add to the overall drama of a theatre play, Ambient Intelligence may thus be applied to enhance the performance described by these social scripts. Second, positioning Ambient Intelligence in performance theory may open up a well-studied and familiar frame of reference for the design of Ambient Intelligence environments and the underlying technology, as we will discuss later.

Many social scripts are enacted at home. The home people live in can be seen as a stage on which perform everyday rituals, such as brushing your teeth, going to work or watching a soccer match. Figure 2 shows a cartoon projected on a Philips mirror TV (a two-way mirror with an LCD screen behind) that invites the small child standing in front of the mirror to brush his teeth for two minutes. The cartoon carries the child through this daily task. It would be too easy to say that we can create an optimal experience by making a task more effective or more entertaining. A better explanation of what an optimal experiences is, is perhaps provided by psychologist Mihaly Csikszentmihalyi, who argues that happiness is not so much a result of finishing a task but more about being immersed and engaged in the process of performing the task. Only then do we get into a state of 'flow' and optimal experience [8]. In the mirror TV example, the cartoon shifts the attention of the child from achieving the end result to the process of getting there. The cartoon increases the flow of the activity by improving the level of engagement.

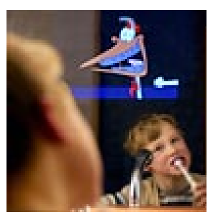

Fig. 2. Enhancing everyday life performances with Ambient Intelligence.

The central role of performances is also reflected in recent business literature about services. Pine and Gillmore [28] talk about an experience economy in which work is theatre and every business a stage. Earlier research on service marketing by Fisk and Grove [13] discusses a theatre framework for service marketing, in which services are seen as performances, front-end service personnel as actors, the service setting as the stage on which the service is performed, products used in the services as props and the business process of the service as the script. Empirical evidence suggests that the 'servicescape', the environment of the service, plays an important role in how people perceive a service encounter. This suggests that Ambient Intelligence can also be applied in professional service encounters to enhance the atmospherics of a service encounter and thereby the perception or performance of a service in a positive way.

Consider for example a medical imaging room in a hospital. Many patients feel frightened by the bulky equipment in the examination room of a hospital. By enhancing this environment with immersive media, e.g. by projecting video clips on the walls and ceiling of the examination room, patients may feel more at ease, as illustrated in Figure 3. Philips Medical Systems demonstrated this concept together with Philips Design in the ambient experience pavilion at the Annual Meeting of the Radiological Society of North America (RSNA) in 2003 (see Figure 3).

Fig. 3. Enhancing a medical examination room with Ambient Intelligence.

3.2 Object-oriented

To understand how experiences are formed, we can also look at how meaning is imposed upon individuals. Literature studies can offer insight in this process [29]. Some schools argue that this meaning is entirely contained in the structure of the text (e.g. structuralism, Russian formalism), others such as Marxism, feminism, gay/lesbian studies argue that the social context is vital to understand this. Other theories of literature such as reader response theory take a much more subject-oriented view and consider the reader in the analysis of the text. Nobody reads the same text. Nobody interprets the meaning in the text in the same way, this depends on the experience of the individual reader.

In most works of literature and therefore literary analysis the text may be read differently by different readers, but the text to which readers are exposed is always the same. In interactive fiction, readers can actively change, affect and alter the plot of the narrative. Although works of interactive fiction and narrative are known from earlier times, interactive fiction became easier and popular with the advent of digital computers. In Hamlet on the Holodeck Janet Murray [27] argues that the computer is not just a tool but as a medium for telling stories. Since Adventure the first text adventure, the computer adventure game genre has evolved and become much more graphical and integrated often in first person action games but the reader that can affect or otherwise alter the plot remains present. The non-linear nature of interactive narratives also attracted the attention of hypertext writers. Michael Joyces Afternoon, a story (1987) and Stuart Moulthrops Victory Garden (1991) are some successful examples of hypertext novels. In interactive narratives, the reader actively transforms in a performer but the performance is changed, affected and altered by the text.

4 Ambient Narratives

Interactive narrative and interactive drama cannot be understood from only the object-oriented or subject-oriented lens. The analysis depends on the choices of the subject but at the same time the choices of the subject are constrained, influenced and maybe even unconsciously or consciously directed by the text. This tension between reader interactivity and plot structure is present in all works of interactive narrative and drama. Too much interaction and the reader/performer feels lost. Too little interaction and a reader/performer feels his actions do not matter. Believability is another important factor [5] in interactive drama (as well as in human computer interaction design in general [21]). The characters in the story world may seem alien, but their actions should be believable at all times within the context of the story world in which they live. Murray [27] proposes three categories for the analysis of interactive story experiences: immersion, agency and transformation. Immersion is the feeling of being present in another place and part of the action, it relates strongly to believability. Agency is the feeling of empowerment that comes from being able to take actions that relate to the players intention. Transformation is the effect that happens if a players self is altered by the multitude of possible perspectives offered by an interactive story experience.

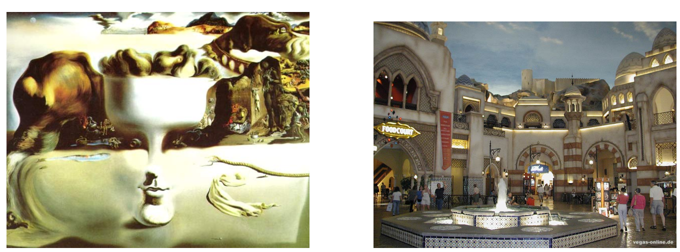

Fig. 4. Different forms of interactive narrative

Forms of interactive narrative are not just found in hypertext fiction or computer games, they can be found in many, perhaps unexpected, places. Painters and sculptures use perspective to tell different stories in one painting or sculpture. Depending on the way you look at the work of art, a different story is revealed to you. Figure 4 (left) shows the famous painting "'Appartition Of A Face & Fruit Dish" by Spanish surrealist painter and artist Salvador Dali. Depending on the way you look at this painting you can see a dog, a cup with hanging fruit, a face and a desert landscape with the sea in the background.

Interactive narrative is also always present in architecture. Depending on the way you walk through a building a different story is told to you. This becomes very clear in museums but also is present in other forms of architecture. Theme parks and themed environments such as the Desert Passage in Las Vegas (see right-hand side of Figure 4) are explicitly designed to support interactive storytelling.

If we 'add' up all such single interactive narratives, we get a single master narrative that might be called an *ambient narrative*. An ambient narrative is superposition of the interactive narrative forms found in architecture, computer games and virtual reality, and works of art. An ambient narrative has the following characteristics:

- Interactive: Readers create their own unique, personal story and experience from a large set of possible choices defined by the author of the interactive narrative.
- Situated in mixed reality: Most forms of interactive narratives and drama are situated in the real world (architecture, improvisational theatre, visual art) or in the virtual world (e.g. virtual reality simulations and adventure games). Ambient narratives have both a real world and a virtual component. 'Text' and 'reading' should therefore be taken broadly: Reading becomes performing in such a mixed reality environment; we may skip a text page as we move from one room into the next. In contrast to augmented reality applications (see e.g. [4]), users do not necessarily need special equipment such as head-mounted displays or see-through glasses to experience these virtual objects. The virtual objects are presented through the individual devices that surround users. Each device generates only a part of the final experience.
- Non-fictional: Ambient narratives can be designed for pure entertainment or infotainment but will be mostly designed to support people in their everyday life activities and rituals (e.g. brushing teeth, cooking etc.).

How does this relate to Ambient Intelligence? By making choices in the ambient narrative we choose our situation (subject-determined), but this situation will affect, alter us as objects to the meaning imposed on us by the ambient narrative in an involuntary way (object-determined). For us as performing readers in the ambient narrative, Ambient Intelligence is a subset of the overall story and the experience formed in our mind. More precisely, Ambient Intelligence is that part of the co-created story that is conveyed through the user interface of the devices surrounding us.

The notion of ambient narratives brings us the organizing concept we are looking for because ambient narratives support the mass customization of ambient intelligent environments. This organizing concept is not built on quicksand. This definition of ambient narrative helps to understand the importance and pervasiveness of interactive texts and performances. We can start to view Ambient Intelligence as something more than a technological phenomenon; we can start to see it as a by-product of a literary form and apply literary theory and performance theory to deepen our understanding. The ambient narrative can be written completely in advance by an experience designer, the author of the ambient narrative like a computer adventure game or only partially, more like a live action role playing game or piece of improvisational theatre. In both cases, the ambient narrative allows readers to make their own choices that affect the plot. In both cases Ambient Intelligence is a by-product of a co-creation process between reader and writer. The difference is that in the latter case, the reader may also alter the plot at a more fundamental level by adding his own material to the ambient narrative: The ambient narrative is not static but dynamically evolving.

We can also derive a more concrete but weaker definition of ambient narratives that helps computer scientists, system architects and software engineers to model media-enhanced performances in the home and in commercial service encounters in a machine understandable way. This definition will be introduced in the second part and forms the basis for the design and implementation of the ambient narrative prototype and simulation tool that is under development. First, we give a scenario of a typically ambient intelligence scenario that will be used as an example throughout the second part.

Part II

5 Example Scenario

Sarah, age 35, married to John arrives at home after a long working day. When she enters the hall, the lights automatically switch on and in the mirror TV in the hall she sees her voice mails and text messages. She quickly browses through them and then leaves to the kitchen to prepare some dinner. When she enters the kitchen, the kitchen portal is shown on the screen in the kitchen. She decides to start an interactive, electronic cooking book application on this screen. After a short while her attention falls on an simple but very nice looking Italian pasta dish which she can prepare with the ingredients present in the house. As soon as she has selected the pasta dish, a voice controlled application guides her through all the necessary steps to prepare the food. Eventually, when the food is ready to be served, she selects a video chat application that opens up a communications channel in each room where a person is. John, who just entered the hall, sees Sarah and responds that he will be right there. Sarah takes the pasta dish to living room and John sets the lights, wall-size displays and ambient music to an Italian restaurant.

6 Ubiquitous physical hypermedia

An ambient narrative can be represented by a hypertext network with nodes and links between nodes. This is illustrated in Figure 5 Each node is a description of an interactive, distributed media application in a ubiquitous, physical hypermedia language that corresponds with a particular everyday performance, for example brushing teeth or a medical examination in a hospital performed by a doctor. Links between nodes model allowed transitions from one media-enhanced performance to another. Transitions should be taken very broadly in physical hypertext; walking from one room to another, changing orientation in a room, touching a lamp can all be seen as events that activate a link. By making choices in the hypertext network, the reader, performer creates a personal story – a personal ambient intelligence experience. Ambient narratives enable the mass customization ambient intelligence. The main difference with mass customization in the traditional sense is that the mass customization process in ambient narratives is continuous as opposed to the assembly of personal computers or design of furniture. When readers start to add and remove their own plot material to the ambient narrative, the mass customization process turns into a co-creation process where readers and writers create the desired experience together.

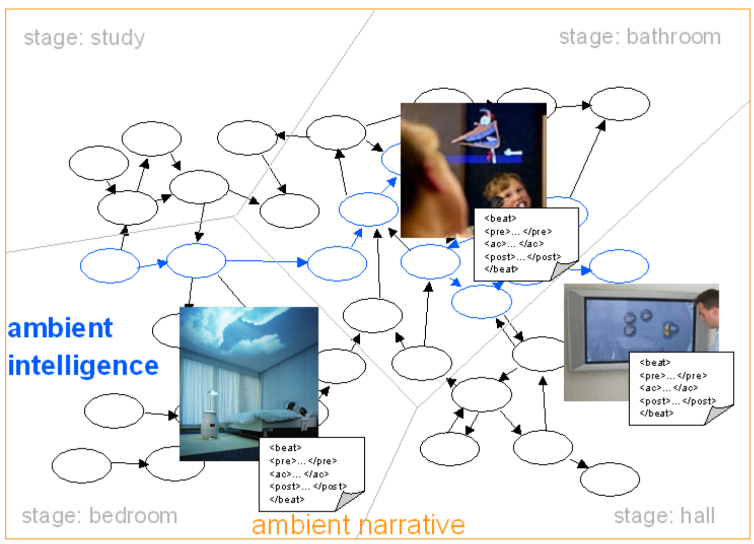

Fig. 5. An ambient narrative as a hypertext network

From a writer's perspective, the ambient narrative describes all possible media-enhanced performances and their interrelationships. In a pure mass customization approach, the ambient narrative is written in advance and enables consumers to create their own personal story, their own Ambient Intelligence from existing plot material. In our example, this means that ambient narrative authors should understand the performances, rituals that take place in Sarah's home. Although this seems a daunting task, there a number of factors that simplify this task: First, many performances are framed by location. In our example, we know beforehand that Sarah is not entering the kitchen to take a shower. Furthermore, even if people may now have more choice than ever before, mass media and popular culture tend to reinforce cultural identity and form new social groups. Authors of ambient narratives could take this into account and design a variety of ambient narratives, each suited for a different life style. Finally, like in movies and drama, people who interact with an ambient narrative may be willing to accept errors in the performance as long as they do not disrupt the overall experience too much. In a co-creation strategy, the ambient narrative plot material is not fully defined in advance but (implicitly) created by the readers of the ambient narrative. This allows end-users to program their own ubiquitous computing environment in exactly the way they see fit. Unless explicitly mentioned, we will look at ambient narratives from the writer's point of view, as our focus is on the production of Ambient Intelligence.

7 Related Work

Ambient narratives can be related to interactive storytelling, hypertext, and ubiquitous computing.

Many interactive storytelling systems designed so far in the interactive fiction domain are based on Artificial Intelligence techniques that control either the plot generation or real-time character behavior, see e.g. [5, 24, 23]. Because AI planning techniques deal with action selection, these techniques lend themselves also well to sequence plot elements into a compelling dramatic story. One of the main issues in interactive drama systems is the balance between player interactivity and plot structure: Too much structure imposed, and users may get the impression that their actions do not matter, because the important choices have been made anyway. Too much focus on interactivity on the other hand causes the impression of feeling 'lost'. Believability is another important factor [5] in interactive drama (as well as in human computer interaction design in general [21]). The characters in the story world may seem alien, but their actions should be believable at all times within the context of the story world in which they live. Murray [27] proposes three categories for the analysis of interactive story experiences: immersion, agency and transformation. Immersion is the feeling of being present in another place and part of the action, it relates strongly to believability. Agency is the feeling of empowerment that comes from being able to take actions that relate to the player's intention. Transformation is the effect that happens if a player's self is altered by the multitude of possible perspectives offered by an interactive story experience.

The non-linear nature of interactive narratives also attracted the attention of both hypertext writers and researchers. Michael Joyce's "Afternoon, a story" (1987) and Stuart Moulthrop's "Victory Garden" (1991) are some successful examples of hypertext novels. During the nineties, hypermedia narratives were also demonstrated, HyperSpeech [3] and HyperCafe [31] to name just a few. Recent publications have expanded models and techniques into mixed reality environments [15, 34, 30]. With respect to interactive storytelling, Weal et al. [40] describe an approach that records the activities of children in an outdoor environment and represent these recorded activities using adaptive hypermedia techniques to children later so they can later relive the experience. Romero and Correia [30] discuss a hypermedia model for mixed reality and describe a mixed reality game that takes place in a gallery environment.

Pervasive or ubiquitous computing plays an important role in ambient narratives. To create the feeling of ambient intelligence, users need to be surrounded by devices that can work in concert on behalf of the user. The term ubiquitous computing was coined by Mark Weiser [41] in the early 90s and many work has been done since. It is outside our scope to discuss this work in detail, see [2] for an overview of applications and research topics such as wireless networking, context-awareness, power management etc.

8 Dexter-based Hypertext Model

The ambient narrative approach is modular, because the narrative itself constitutes the modular parts of all the possible Ambient Intelligence fragments. The user interacts with the narrative engine by the *ambient browser*, that collects user input and context information from multiple devices. The *ambient narrative engine* determines the next episode in the 'story' told, given the user input and context and the current position in the ambient narrative plot. It returns a description of the next episode that is rendered by the ambient browser. Expressed in business terminology we follow the paradigm of mass-customisation [9], where the ambient browser and ambient narrative engine assemble a customized product, Ambient Intelligence, and deliver it to the customer, the reader/performer.

The remainder of this section describes how the high-level ambient narrative system architecture above can be implemented in terms of (extensions of) the Dexter hypertext reference model[16]. Figure 6 shows the mapping of our high-level ambient narrative system architecture on the Dexter model. The following subsections explain how the run-time layer is implemented by the ambient browser, the storage layer by the ambient narrative engine and the within-component layer by the plot material of an ambient narrative. In the following sections, we will discuss the current system architecture and the results of experiments conducted with an ambient narrative simulation tool we have written.

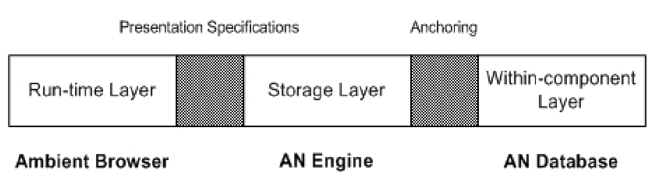

Fig. 6. Mapping the high-level system architecture onto the Dexter reference model.

8.1 Run-time Layer: Ambient Browser

Although hypermedia can be added to the Dexter model as a data type in its storage layer, this approach cannot adequately support the complex temporal relationships among data items, high-level presentation attributes and link context that are important in supporting hypermedia. The Amsterdam Hypermedia Model (AHM) [17] extends the Dexter model by adding these hypermedia requirements. The AHM model inspired the definition of the Synchronized Multimedia Integration Language (SMIL) [37]. AHM and the SMIL language are designed for hypermedia presentations running on a single device and therefore do not mention the issue of timing and synchronization of media elements across multiple devices, characteristic for mixed reality or ubiquitous hypermedia. To support timing and synchronization of media objects within and across devices, we use an in-house developed SMIL interpreter. This SMIL interpreter has the role of a networked service to which the other devices register themselves. The underlying reason for this choice is that we expect 'media' to include also lamps, fans and other output devices in the future. To use this functionality in SMIL, we have extended the toplayout element in the SMIL language with a proprietary 'target' attribute that specifies the rendering (or input) device. The author of the SMIL document can set the target attribute of the toplayout element in the SMIL head part to point to a specific rendering or input device. In the SMIL body part, the author can use (as he would normally do) the id of the toplayout element or the id of one of its region element children in the region attribute of a media element (e.g., an image, or a video fragment), to indicate the device/region on which the media element should be rendered. The advantage of this approach is that we can remain close to the AHM model and do not have to introduce spatial mapping functions outside the SMIL engine as described in [20] for example.

Since every performance can be augmented with a hypermedia document and multiple performances can be going on at the same moment in an ambient intelligent environment, we have extended the run-time layer further to deal with document sets. Users can add, remove or replace documents from the set of documents in the ambient browser and in doing so change parts of the surrounding ambience.

From a high-level point of view, the presentation and interaction with SMIL belongs to the run-time layer in the Dexter model. This view differs to some extent from the role of media objects in the AHM, which addresses mostly the storage layer. We consider SMIL documents as basic building blocks, i.e., the components in the storage layer. The AHM model however views individual media objects as the principal components. The advantage of our approach is that we can abstract from the low-level positioning of media objects in space and time and focus on describing how such hypermedia presentations are selected based on context situations. This does not mean that the AHM model is not present; it is just hidden. Inside the SMIL components, the AHM model is revealed. Since this extended SMIL document is viewed as a component, we could replace it by a different hypermedia markup language, or use dedicated applications with new forms of multimodal interaction (e.g., those not supported by the SMIL implementation).

8.2 Storage Layer: Ambient Narrative Navigation

The storage layer in the Dexter model describes the mechanisms by which nodes and links are connected to form networks. The storage layer itself is not specific about the internal structure of the components; components are treated as generic data containers. The components of the ambient narrative are called *beats*. The term beat originates from theater play and film script-writing and is defined [25] as a change in the behavior of a character. The beat is the smallest unit within a scene, that in turn represents a greater change in behavior. Sequences of scenes form acts which culminate into even greater change. This beat (sequencing) approach is also taken in interactive drama, e.g. by Mateas [24], Magerko [23] and Brooks [6]. Since ambient narratives are also enacted, we choose the same terminology. So, an ambient narrative can be viewed equivalent to a hypertext network consisting of beats (nodes) and links between these beats.

An individual Ambient Intelligence experience is represented by a sequence of beats and any specific parameters that may have been used in the instantiation of these beats. Like in Schank's script theory, specific memories (in our case Ambient Intelligence experiences) are represented as pointers to generalized episodes (beats) plus any unique events for a particular episode (represented by story values, as explained shortly).

Beat language Now, we discuss first a language for representing beats, followed by an explanation of how beats are sequenced by a beat sequencing engine or ambient narrative engine. We analyzed dozens of Ambient Intelligence scenarios from literature and past projects in our groups and derived the beat markup language and beat sequencing algorithms from this analysis. Here we will explain the design choices made, illustrated with examples taken from the scenario of section 5.

In mixed reality environments, context and text belong together. Presentation descriptions should only be selected if both the text (position in the narrative) and the context match the current situation (i.e., position in the real-world). Therefore, each beat description consists of two parts: a *preconditions* part to represent the context, and an *action* part to represent the 'text'.

The beat sequencing engine (or ambient narrative engine) that controls the navigation through the ambient narrative checks the *preconditions* part during the selection or search for new beat components. It specifies the conditions that must hold before the beat can be selected; restrictions can be set on the performance (activity), actors (users) that need to be present, stage (location), props (tangible objects and devices present) and/or script (session variables).

The *action* part is executed after the beat has been selected by the engine for sequencing. It is subdivided in an initialization part, a main part and a finalization part. The initialization part is executed when the beat is activated and before the presentation description in the main part is executed. The main part contains the actual presentation description, encoded by a ubiquitous hypermedia document. The finalization part can contain instructions that specify what should happen if the beat is removed.

Because one beat alone will not be perceived as ambient intelligence by readers/performers of the ambient narrative, a requirement for the beat markup language is that it supports links between beats or queries for beats that can be activated by readers. For example, when Sarah decides to go for the Italian dish, she explicitly starts the prepare food application that helps her through the cooking performance.

Clearly, it is too restrictive to support only one activity at a time: At any time there are typically N actors involved in M performances in such an environment. To model multiple activities in the model, it is necessary to support concurrent browsing paths in the model. We looked at a number of approaches discussed in literature. The Trellis system explained in [36] uses a formal Petri net based model to model synchronization of simultaneous traversals of separate paths through a hypertext. The Petri net based model probably works best if the hypertext network is not open-ended, but we would like to support that both authors and end-users can add/remove beats from the narrative as well as allow conditional links (queries) between beats to make room for more variety and surprise during navigation. The spreading activation network introduced in [22] could also be a candidate to model concurrent browsing if we treat a beat as a compentence module of an autonomous agent (the ambient narrative engine) and model the action selection (beat selection) as an emergent property of an activation/inhibition dynamics among these modules. We decided to limit ourselves to a more straightforward solution however, and simply add a 'behavior attribute' to the link that indicates if the target of the link (query on the beat database) should be added to the existing set and/or if the source of the link (query on the active beat set) should be removed from the existing set.

Beat links and queries are traversed immediately as a response to a specific user event (e.g. press of a mouse button, touching a part of the screen, stepping on a floor sensor). In many Ambient Intelligence scenarios, applications are not triggered immediately but as soon as certain context conditions have been fulfilled. We define a context trigger in the beat language as an element that has a preconditions part and a link. When the trigger is set, the system waits for the preconditions specified in the trigger(s) to become true. If all preconditions are valid, the system traverses the link. The situation where Sarah enters the hall and the voice and text messages are presented on the screen in the hall is modelled as a context trigger.

For many Ambient Intelligence applications in the scenarios we investigated it proved too inflexible to allow for only implicit context changes and explicit user feedback. The choice of the next beat can also be dependent on information from the past (e.g. user preferences), present (current narrative state) or future (user goals set). We want to prevent that Sarah can select the food preparing application if she has not selected a recipe first in the electronic recipe book for example. A further requirement for the beat language is therefore a mechanism to test and change variables. These variables that change during the session are called *story values* and are stored in a global *story memory* (see below). The author of an ambient narrative can specify in the precondition part of a beat or trigger which story values must hold. In the action part of a beat the author can define story value changes. These story value changes will be written in the story memory and influence the selection of new beats. At the moment, each story value is a simple key value pair and the story memory is simply the list of all story values that have been set.

Links, triggers and story-value changes elements can be placed in either the initialization, main and/or finalization sections in a beat to describe possible transitions between beats. These elements are called from the run-time layer.

Table 1 summarizes the main beat language features.

Requirement	*Language Construct*
context and text description joined	beat
user-based navigation	link
simultaneous performances	beat sets
context-based navigation	trigger
session variables	story values stored in a story memory

Table 1. Main beat language requirements

The following XML fragment is a typical beat description. It contains a `pre` element that describes the minimal required context conditions and an action part that describes what happens when the beat has been scheduled for execution. For space reasons the presentation markup has been omitted.

```
<!-- OnBrowsingRecipes:

Starts an application that shows electronic recipes the user can choose and customize.
The OnBrowsingRecipes, OnPreparingFood and OnCommunicating beats model a workflow.

-->
<!DOCTYPE beat SYSTEM "http://localhost:8080/beatdb/HomeLab/beat.dtd">
<beat id="Kitchen_OnBrowsingRecipes1">
<!-- preconditions that must hold for this beat to become selected -->
<pre>
<stage id="Kitchen" time="day" location="HomeLab">
<performance id="BrowsingRecipes">
<actor id="housemate"/>
<prop id="screen2" capability="SVGA touchScreen"/>
</performance>
</stage>
</pre>
<!-- what happens when the beat is selected -->
<action>
<init/>
<main preview="http://localhost:8080/beatdb/HomeLab/previews/Kitchen_OnBrowsingRecipes1.jpg">
<!-- presentation markup interleaved with beat instructions (presentation markup deleted) -->
```

```
<!-- set a goal story value and query -->
<story-value action="add" id="setPrepareFood" name="prepareFood" value="true"/>
<link id="prepareFood" behavior="add"
to="http://localhost:8080/beatdb/HomeLab/queries/Kitchen_OnPreparingFood.sql"/>

<!-- kill beat -->
<link id="kill" behavior="delete"
from="http://localhost:8080/beatdb/HomeLab/queries/Kitchen_OnBrowsingRecipes1.sql"/>
</main>
<final/>
</action>
</beat>
```

The design of this beat language has similarities with the interactive drama approaches taken by Mateas and Magerko. Our preconditions part is similar to the selection knowledge stage in Mateas' Facade interactive drama architecture [24] that also uses tests on story memory values (and prior probability on beats). The action stage in Facade roughly corresponds to our action and post-conditions parts in the beat document. Magerko's IDA architecture [23] represents plots at the scene level and consists of five stages: initial state, required events, background knowledge, content constraints and temporal constraints. The initial state sets up the scene and may be similar to the initialization part in our action stage. The required events and background knowledge are comparable with our preconditions stage, while the content constraints that limit the binding of the variables used in the required events are similar to the embedded queries in the main action description part of our ambient narrative beat documents.

Beat sequencing Beats are sequenced together to create a personalised story in mixed reality, the result of which is what we have called Ambient Intelligence. The beat descriptions and beat sequencing engine or ambient narrative engine can be seen as an adaptive hypertext system. Brusilovsky [7] describes adaptive hypermedia as follows: *"By adaptive hypermedia systems we mean all hypertext and hypermedia systems which reflect some features of the user in the user model and apply this model to adapt various visible aspects of the system to the user. In other words the system should satisfy three criteria: it should be a hypertext or hypermedia system, it should have a user model and it should be able to adapt the hypermedia model using this model."*. The Adaptive Hypermedia Application Model (AHAM) [11] builds further upon this definition and tries to fit adaptive hypertext and hypermedia in the Dexter model. It defines a hypermedia application as consisting of a domain model, user model, teaching model and adaptive engine. The domain model describes how the information is structured in nodes and links. The user model describes the knowledge level of the user and also keeps a record of the nodes visited by the user in the past. The teaching model consists of learning rules (pedagogical rules) which define how the domain model interacts with the user model. The adaptive engine performs the actual adaptation.

The beat descriptions and beat sequencing engine can be quite easily described in terminology of the AHAM model. Beats and their interrelationships form the domain model. The user model is implicit in the story memory of the ambient narrative engine: The story memory contains session knowledge which can contain user preferences as we discussed before. The story memory dynamically evolves out of the continuous interaction between users and the ambient narrative. The teaching model is encoded in the beat descriptions. The action part allows the author to alter story values that can affect the selection of new beats and content items. The adaptive engine is the ambient narrative engine that sequences beats. The ambient narrative engine must implement an action selection mechanism as its main task is to find the next best beat. We implemented the beat sequencing planner using a structured information retrieval approach. Queries for new beats are encoded in the action part of the beat description and may contain both fixed parameters and story values. Beat queries are never explicitly entered by the user, they are selected and filled in by the beat sequencing engine based on user input and information present in the story memory.

The advantage of this approach is that it allows us to introduce adaptation at different levels of the narrative like the Facade [24] and Hyperdoc [26] systems. If we use presentation templates instead of finalized presentation descriptions we can allow for adaptation at the subnode level, i.e., change the media objects in a SMIL fragment in the action part of the beat. Beat queries enable us to describe adaptation at the link level: The choice of a beat can be made context-dependent by using story values or context information in the beat queries. The use of beat preconditions and beat queries also allow us to easily add new beats and content without taking the system down. This way we can defer editing decisions by the narrative engine on both the node and subnode level until the moment they are played out. The same technique is used by the Automatist Storyteller system [10] for content items only. As a result authoring effort is lowered because the author does not have to explicitly sequence story elements into a finished story or rewrite the existing narrative when new material is added. This also provides the writer with the flexibility to add beats for specific devices that can be chosen if these devices are owned by the reader. Furthermore, authoring tools could assist readers in creating their own beats and inserting them in their beat collection.

8.3 Within-component Layer: Plot Material

Depending on whether you look at the narrative level or at the hypermedia level, the presentation document is either an atomic component or a composite component. In the first case the component content in the within-component layer are for example SMIL documents or even dedicated applications, in the second case the component content are media objects, text, images, audio, video and other modalities such as lightscripts. All the component information and content is indexed and stored in an ambient narrative database for fast retrieval by the ambient narrative engine.

9 System Architecture and Algorithms

Figure 7 shows the architecture of the ambient narrative engine. User feedback, contextual data implicitly derived from sensors and the contents of the story memory together determine the direction through the ambient narrative and how the selected beats are customized and sequenced.

First we discuss each component in Figure 7, then we illustrate how the engine works with an example from our scenario.

The *beat database* contains the beat documents that can be chosen. The *content database* stores all media elements and third-party applications that are needed for the beat documents in the beat database. The *context database* maintains a model of the ambient narrative physical context. This information is needed to verify the actor, prop and stage preconditions for beats. The *story memory* is a list of story values that can be set by beats. All the beats that are currently active are placed in the *active beats* set. Context triggers set by beats are stored in the *active triggers* list.

The *internal event server* receives events from the ambient browser (incoming arrow). As can be seen in example beat, the action part contains language elements (link, story-value) that are not meant for presentation by the ambient browser but for navigation through the ambient narrative. When the ambient browser encounters such an element, it forwards the element id and corresponding document id to the internal event server of the ambient narrative engine. This component determines if the element id is a link, trigger or story-value change element. In case of a link type, the element is forwarded to the *beat scheduler*. In case of a trigger or story value change, either the active trigger set or story memory is updated.

The beat scheduler may implement a scheduling and a filtering algorithm to change the order in which queries for beats arrive. In the current implementation the beat scheduler just passes the link information on to the *beat retrieval engine*.

The beat retrieval engine processes and executes the queries contained in the link elements: Each link consists of either a 'to' or 'from' attribute or both. The 'to' attribute contains a query for one or more new beats on the beat database, the 'from' attribute can be used to specify a query for removing beats from the active beat set. When a link element arrives, the beat retrieval engine executes the queries in the 'to' and/or 'from' fields and checks if the retrieved beat preconditions are fulfilled. Beats that are scheduled for removal are forwarded to the *beat set manager*. Beats that need to be added to the *beat set manager* are first processed by the *beat processor*. In our scenario, the example beat is added by the beat set manager as a result of a link that was triggered when Sarah entered the kitchen.

The beat processor parses the beat description and checks if the beat contains open parameters (adaptation at the sub-node level) for example embedded content queries that need to be filled in first before the beat can be send to the ambient browser. At this moment we do not support beat templates and assume beats are finalized at the moment of writing.

The beat set manager updates the active beat set and synchronizes this with the ambient browser to make sure the correct applications are running (those specified in the main part of the beats in the active beat set). The beat set manager also checks for any story value and/or trigger instructions in the initialization and finalization parts.

The *external event server* finally tests the preconditions of the beats in the active set and context trigger list. If a beat in the active beat set is no longer valid because one or more preconditions no longer hold, a number of different strategies are possible: remove the beat, look for a more specific or generic beat or do nothing. If all preconditions hold of one or more of the triggers in the context trigger list, the external event server will send the link that belong to the trigger to the beat scheduler for further processing.

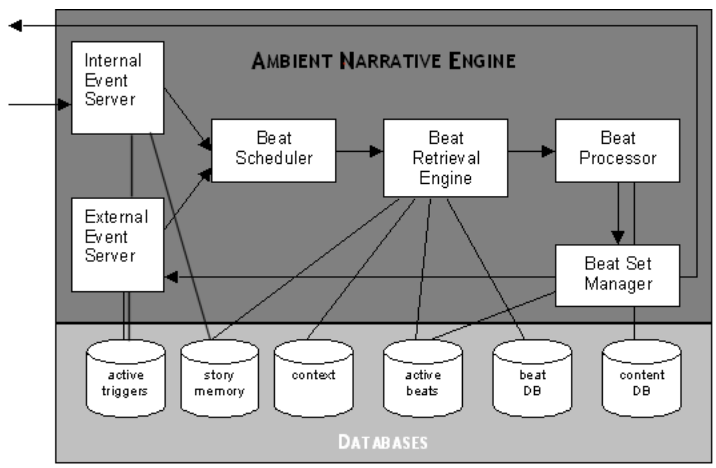

Fig. 7. Ambient narrative engine architecture

When Sarah selects the video chat application to notify other people that the pasta dish is ready, the ambient narrative engine receives the element and document identifiers for two triggers (one for the hall and one for the living) from the ambient browser. The internal event server adds the triggers to the context trigger list and notifies the external event server to recompute the active beat set. The external event server checks the preconditions of the triggers by consulting the context database and story memory and finds that one of the triggers needs to be activated. The external event server then forwards the link to the beat scheduler for further processing. The beat scheduler decides to do nothing and sends the link to the beat retrieval engine which retrieves the beat that describes the video chat application in the hall. The beat retrieval engine forwards the resulting beat to the beat processor which checks if the beat description needs to be processed further, i.e. has any open parameters. The beat set manager adds the beat to the active beat set and notifies the ambient browser to update the running presentation. The result is that John sees a video chat application started in the hall with Sarah asking him to join dinner.

The design of the ambient narrative engine is inspired by the drama manager in the Facade system. This drama manager also uses user input and a story memory to sequence beats into an immersive interactive story. The difference with our approach is that we can not only adapt the link structure based on user interaction but also, to some extent, the nodes; for, we allow embedded queries in the beat description language. Similar concepts are also used by Magerko [23] and Szilas [38]: Margerko uses the word director while Szilas applies the term narrator to refer to a drama manager.

10 Experiments and Results

To verify the ambient narrative system, we developed an ambient narrative simulation tool. The reasons for building a simulation tool instead of a real prototype were two-fold: First, we wanted to reduce risks and development costs. Secondly, experience with building Ambient Intelligence type of applications and environment has taught us that it is hard to quickly prototype and debug interactive media environments. Therefore, we decided to work on a simulation tool first. Similar approaches towards rapid prototyping of context aware applications can be found in i.e. [35, 18].

The simulation tool fully implements the ambient narrative engine (storage layer), but emulates the behavior of the ambient browser (run-time layer) by means of a web interface through which users can manually enter events (user feedback and context changes) for the narrative engine and see the result of their action. The risk of this approach is of course that the simulation tool does not give a realistic experience, but since the ambient browser and narrative engine are relatively independent, we believe the risk is tolerable.

To test the concept, we picked fifteen Ambient Intelligence scenarios situated in the home (hall, kitchen and living) from the set of scenarios (130) we analyzed and wrote an ambient narrative that combined these scenarios. We evaluated the scenarios based on their complexity and difference to other scenarios in the set. This resulted in a network of 33 beats written in our beat markup language. We experimented with the ambient narrative by entering events directly in the simulation tool and viewing the effects on the active beat set. To make the simulation visually more appealing, we added a preview attribute to each beat. This preview attribute links to a picture or movie that illustrates how the beat would have looked like in a real ambient narrative system. During the simulation the ambient browser simulation shows the pictures that belong to the beats in the active beat set (see Figure 8).

Even with this relatively small ambient narrative, we can see interesting results and issues for further study.

Triggers, links and story-values create a highly dynamic system: For example, when a beat that has been launched by a context trigger is removed, it will be

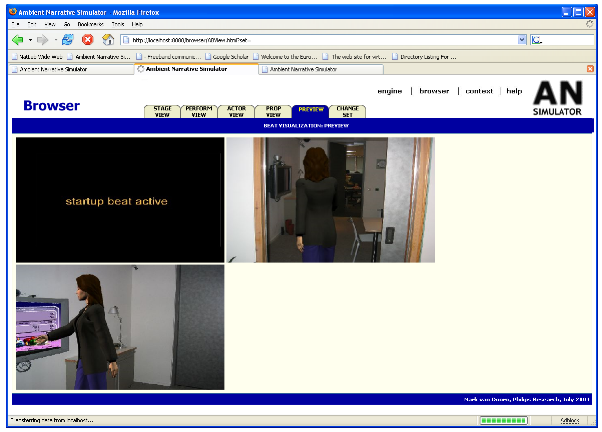

Fig. 8. Screenshot of Ambient Narrative Simulator

immediately reactivated if the context trigger is still set. Although this behavior may seem flawed at first instance, there are scenarios where this behavior is actually desired. For example, when a user logs off a public kiosk in a museum, the application should be restarted. In the experiment we found that we could realise the desired behavior simply by setting a story value that is tested in the precondition of the trigger or by removing the trigger. This same mechanism can be used for links: only traverse the link from the electronic recipe book to the preparing food application if story value 'prepareFood' has been set to true. Although it is possible to write the beats manually without any help of an authoring tool, authoring support is desirable for non-trivial ambient narratives because it can be quite difficult to figure out why a particular beat has been (re)activated.

In designing the ambient narrative documents, we also noticed that very dissimilar application scenarios share the same structure. For example the voice and text messaging application in the hall and electronic recipe book application in the kitchen are structurally similar: On entering the room we set a context triggers that launches a portal type of application in the room from which readers can browse further. Workflow patterns like the selecting and preparation of food are modelled by setting and testing story values. Communication patterns are modelled using triggers: When Sarah selects the video chat application in the kitchen to notify Sarah the food is ready, triggers are set in the other rooms that test for the presence of a housemate. This pattern can be used for any modality and any number of presentations. We found that the function (beat relations) and context of beats is at least as important as the contents (interactive media presentation) of the beat contents themselves to understand the ambient narrative. This structuralist approach to understanding ambient narratives is just one angle of looking at the ambient narrative text. The interaction of users with the ambient narrative also deserves attention because ambient narratives are not really read but performed. The possibility of reusable plot structures for very different scenarios and domains may also simplify the development of an authoring tool, but this requires further research.

Tests with this example ambient narrative also revealed the need for some extensions on the model. At the moment we can model links between beats as a link or a database query. As a result there is no element of surprise or variation which leads to predictable behavior. While this may be sufficient (maybe even required) for functional, rational tasks, it is too restrictive for more emotional, entertaining performances. Another issue we found while experimenting with the simulation tool is that users may want to save (part of) the current state of the ambient narrative session in their user profile. To support this, the model must support both persistent and non-persistent story values and the ability to update story values in the user profiles with the story memory and vice-versa. Furthermore, the localization model allow us to describe which props must be present on a stage but cannot express how props relate to each other: For example, it is possible to describe in our current beat markup language that a room should have a screen, surround loudspeakers, but it not possible to specify that the screen should be in front of the user and the loudspeaker around him at a particular maximum or minimum distance. The last issue we mention here is multiple user support. At the moment we can only describe (as part of the preconditions part of a beat or trigger) which actors must be on a stage and thereby who has access to beats. Relationships between actors and other attributes such as the minimum and maximum number of actors allowed in a performance are also required to model more complicated scenarios.

Next to design issues there are some implementation issues worth mentioning. The simulation tool user interface is adequate for testing all functionality, but requires manual input for all events and context changes which is inconvenient and can easily lead to typing errors. Early on in the implementation we decided to store all context and beat data in relational databases for algorithmic optimization and scalability reasons and develop the ambient narrative engine as a web application deployed on a J2EE-based web application platform.

11 Conclusions

Ambient Intelligence is a vision on the future of consumer electronics that refers to the presence of a digital environment that is sensitive, adaptive and responsive to the presence of people. Since it is technologically not possible to mass produce Ambient Intelligence with the current state of the art in artificial intelligence and economically not feasible to manually develop tailor-made Ambient Intelligence products or services for each customer individually, we believe a different approach is needed to move such type of ubiquitous computing environments out of research laboratories into the real world.

We described a mass customization strategy for Ambient Intelligence services offered over a collection of networked devices to customize Ambient Intelligence on a mass basis. In this approach, modular parts of Ambient Intelligence descriptions are assembled based on user feedback and interaction history into a personalized flow of personalized, interactive media presentations in which multiple devices may participate simultaneously. The Ambient Intelligence service thus allows people to create their own Ambient Intelligence within the scope of possibilities set down by the designer or writer of an ambient narrative, an interactive narrative in mixed reality that is designed to support people at home or in a particular service encounter in performing their everyday activities. We explained how an *ambient narrative system* can be implemented in the existing Amsterdam Hypermedia Model (AHM) and Adaptive Hypermedia Application Model (AHAM). We implemented a prototype that simulates the concepts introduced in the setting of the home, and discussed the lessons we learned. Currently, we are working to connect the ambient narrative engine to a real device rendering platform to create a fully working prototype and test the combined system in HomeLab, the usability and feasibility lab at Philips Research.

Still, much work needs to be done and many improvements are possible on both the model and the algorithms used. This is why we decided to write this article in the first place. We hope that by introducing the ambient narrative concept and the hypertext model we have given you something to think about and perhaps even new ideas for AI algorithms and methods or new ways of applying existing AI algorithms and techniques in such ubiquitous computing environments.

References

1. E. Aarts and S. Marzano, editors. *The New Everyday: Views on Ambient Intelligence*. 010 Publishers, 2003.
2. G. Abowd and E. Mynatt. Charting past, present, and future research in ubiquitous computing. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 7(1):29–58, 2000.
3. B. Arons. Hyperspeech: Navigating in Speech-Only Hypermedia. In *UK Conference on Hypertext*, pages 133–146, 1991.
4. R. Azuma. A Survey of Augmented Reality. In *SIGGRAPH'95*, pages 1–38, 1995.
5. J. Bates, A. Loyall, and W. Reilly. Broad Agents. *Sigart Bulletin*, 2(4):38–40, 1991.
6. K. Brooks. *Metalinear Cinematic Narrative: Theory, Process and Tool.* PhD thesis, MIT Media Lab, 1999.
7. P. Brusilovsky. Methods and Techniques of Adaptive Hypermedia. *User Modeling and User-Adapted Interaction*, 6(2-3):87–129, 1996.
8. M. Csikszentmihalyi. *Flow: The Psychology of Optimal Experience* . Perennial, 1991.
9. G. DaSilveira, D. Borenstein, and F. Fogliatto. Mass Customization, Literature Review and Research Directions. *International Journal of Production Economics*, 72(1):1–13, 2001.
10. G. Davenport and M. Murtaugh. Automatist Storyteller Systems and the Shifting Sands of Story . *IBM Systems Journal*, 36(3):446–456, 1997.
11. P. de Bra, G.-J. Houben, and H. Wu. AHAM: A Dexter-Based Reference Model for Adaptive Hypermedia. In *UK Conference on Hypertext*, pages 147–156, 1999.

12. J. Falk. *Funology: From Usability to Enjoyment*, chapter Interfacing the Narrative Experience. Kluwer Academic Publishers, 2003.
13. R. Fisk and S. Grove. The Service Experience as Theater. *Advances in Consumer Research*, 19:455–461, 1992.
14. E. Goffman. *The Presentation of Self in Everyday Life*. Doubleday: Garden City, 1959.
15. K. Gronbaek, P. Vestergaard, and P. Orbaek. Towards Geo-Spatial Hypermedia: Concepts and Prototype Implementation. In *Proceedings of the 13th Conference on Hypertext and Hypermedia*, Maryland, USA, 2002.
16. F. Halasz and M. Schwartz. The Dexter Hypertext Reference Model. *Communications of the ACM*, 37(2):30–39, February 1994.
17. L. Hardman, D. Bulterman, and G. V. Rossum. The Amsterdam Hypermedia Model: Adding Time and Context to the Dexter Model. *Communications of the ACM*, 37(2):50–64, February 1994.
18. R. Hull, B. Clayton, and T. Melamed. Rapid Authoring of Mediascapes. In *Proceedings of the UbiComp Conference*, 2004.
19. G. S. Jr. Time - The Next Source of Competitive Advantage. *Harvard Business Review*, 86(4):41–51, 1988.
20. C. Kray, A. Kruger, and C. Endres. Some Issues on Presentations in Intelligent Environments. In *First European Symposium on Ambient Intelligence (EUSAI)*, 2003.
21. B. Laurel. *Computers as Theatre*. Addison-Wesley, 1993.
22. P. Maes. How To Do The Right Thing. *Connection Science Journal*, 1(3):291–321, 1989.
23. B. Magerko. A Proposal for an Interactive Drama Architecture. In *AAAI 2002 Spring Symposium Series: Artificial Intelligence and Interactive Entertainment*, 2002.
24. M. Mateas. An Oz-Centric Review of Interactive Drama and Believable Agents. Technical report, School of Computer Science, Carnegie Mellon University, Pittsburgh, 1999.
25. R. McKee. *Story: Substance, Structure, Style and The Principles of Screenwriting* . Regan Books, 1997.
26. D. Millard and e.a. Hyperdoc: An Adaptive Narrative System for Dynamic Multimedia Presentations. In *Proceedings of the 14th Conference on Hypertext and Hypermedia*, 2003.
27. J. Murray. *Hamlet on the Holodeck: The Future of Narrative in Cyberspace*. MIT Press, 1998.
28. J. Pine and J. Gillmore. *The Experience Economy*. Harvard Business School Press, 1999.
29. J. Rivkin and M. Ryan. *Literary Theory: An Anthology*. Blackwell Publishers, 1998.
30. L. Romero and N. Correia. HyperReal: A Hypermedia Model for Mixed Reality. In *Proceedings of the 14th ACM Conference on Hypertext and Hypermedia*, Nottingham, UK, 2003. ACM.
31. N. Sawhney, D. Balcom, and I. E. Smith. HyperCafe: Narrative and Aesthetic Properties of Hypervideo. In *UK Conference on Hypertext*, pages 1–10, 1996.
32. R. Schechner. *Performance Studies: An Introduction*. Routledge: New York, 2002.
33. G. Schulze. *Die Erlebnisgesellschaft: Kultursoziologie der Gegenwart*. Campus, 1992.

34. P. Sinclair, K. Martinez, D. Millard, and M. Weal. Links in the Palm of your Hand: Tangible Hypermedia using Augmented Reality. In *Proceedings of the 13th Conference on Hypertext and Hypermedia*, Maryland, USA, 2002.
35. T. Sohn and A. Dey. iCAP: Rapid Prototyping of Context-aware Applications. In *Proceedings of the CHI 2004 Conference on Human Factors in Computing Systems*, 2004.
36. P. Stotts and R. Furuta. Petri-net-based hypertext: document structure with browsing semantics. *ACM Transactions on Information Systems*, 7(1):3–29, 1989.
37. Synchronized Multimedia Integration Language (SMIL) W3C standard. http://www.w3.org/AudioVideo/.
38. N. Szilas. Interactive Drama on the Computer: Beyond Linear Narrative. In *AAAI 1999 Fall Symposium on Narrative Intelligence*, 1999.
39. A. Toffler. *Future Shock*. Bantam Books, 1984.
40. M. Weal, D. Michaelides, M. Tompson, and D. D. Roure. The Ambient Wood Journals - Replaying the Experience . In *Proceedings of Hypertext '03*, Nottingham, UK, 2003.
41. M. Weiser. The Computer of the 21st Century. *Scientific American*, 1991.