Chapter 3

A Generic Architecture for Human-Aware Ambient Computing

Tibor Bosse, Mark Hoogendoorn, Michel C.A. Klein, and Jan Treur
Department of Artificial Intelligence, VU University Amsterdam

Abstract

A reusable agent-based generic model is presented for a specific class of Ambient Intelligence applications: those cases addressing human wellbeing and functioning from a human-like understanding. The model incorporates ontologies, knowledge and dynamic models from human-directed sciences such as psychology, social science, neuroscience and biomedical sciences. The model has been formally specified, and it is shown how for specific applications it can be instantiated by application-specific elements, thus providing an executable specification that can be used for prototyping. Moreover, it is shown how dynamic properties can be formally specified and verified against generated traces.

3.1 Introduction

The environment in which humans operate has an important influence on their wellbeing and performance. For example, a comfortable workspace or an attentive partner may contribute to good performance or prevention of health problems. Recent developments within Ambient Intelligence provide technological possibilities to contribute to such personal care; cf. [Aarts *et al.* (2003)], [Aarts *et al.* (2001)], [Riva *et al.* (2005)]. For example, our car may warn us when we are falling asleep while driving or when we are too drunk to drive. Such applications can be based on possibilities to acquire sensor information about humans and their functioning, but more substantial applications depend on the availability of adequate knowledge for analysis of information about human functioning. If knowledge about human functioning is represented in a formal and computational format in devices in the environment, these devices can show more human-like understanding, and (re)act accordingly by undertaking actions in a knowledgeable manner that improve the human's wellbeing and performance. As another example, the workspaces of naval officers may include systems that track their gaze and characteristics of stimuli (e.g., airplanes on a radar screen), and use this information in a computational model that is able to estimate where their attention is focussed at; cf. [Bosse *et al.* (2006b)]. When it turns out that an officer neglects parts of a radar screen, such a system can either indicate this to the person (by a warning), or arrange in the background that another person or computer system takes care of this neglected part.

In recent years, human-directed scientific areas such as cognitive science, psychology, neuroscience and biomedical sciences have made substantial progress in providing an increased insight in the various physical and mental aspects involved in human functioning. Although much work still remains to be done, dynamic models have been developed and formalised for a variety of such aspects and the way in which humans (try to) manage or regulate them. From a biomedical angle, examples of such aspects are (management of) heart functioning, diabetes, eating regulation disorders, and HIV-infection; e.g., [Bosse *et al.* (2006a)], [Green (2005)]. From a psychological and social angle, examples are emotion regulation, attention regulation, addiction management, trust management, stress management, and criminal behaviour management; e.g., [Gross (2007)], [Bosse *et al.* (2007)], [Bosse *et al.* (2008c)].

The focus of this paper is on the class of Ambient Intelligence applications as described, where the ambient software has context awareness (see, for example, [Schmidt (2005)], [Schmidt *et al.* (1999)], [Schmidt *et al.* (2001)]) about human behaviours and states, and (re)acts on these accordingly. For this class of applications an agent-based generic model is presented, which has been formally specified. For a specific application, this model can be instantiated by case-specific knowledge to obtain a specific model in the form of executable specifications that can be used for simulation and analysis. In addition to the naval officer case already mentioned, the generic model has been tested on a number of other Ambient Intelligence applications of the class as indicated. Three of these applications are discussed as an illustration, in Section 5, 6 and 7 respectively. Section 3.2 describes the modelling approach. In Section 3.3 the global architecture of the generic model is presented. Section 4 shows the internal structure of an ambient agent in this model. Section 8 shows how overall properties of this type of Ambient Intelligence system can be specified, verified against traces and logically related to properties of the system's subcomponents. Finally, Section 9 is a discussion.

3.2 Modelling Approach

This section briefly introduces the modelling approach used to specify the generic model. To specify the model conceptually and formally, the agent-oriented perspective is a suitable choice. The processes in the generic process model can be performed by different types of agents, some human, some artificial. The modelling approach used is based on the component-based agent design method DESIRE [Brazier *et al.* (2002)], and the language TTL for formal specification and verification of dynamic properties [Bosse *et al.* (2008b)], [Jonker and Treur (2002)].

Process and Information Aspects Processes are modelled as components. A component can either be an active process, namely an agent, or a source that can be consulted or manipulated, which is a world component. In order to enable interaction between components, interaction links between such components are identified and specified. Ontologies specify interfaces for components, but also what interactions can take place between components, and the functionalities of components.

Specification Language In order to execute and verify human-like ambience models, the expressive language TTL is used [Bosse *et al.* (2008b)], [Jonker and Treur (2002)]. This predicate logical language supports formal specification and analysis of dynamic properties, covering both qualitative and quantitative aspects. TTL is built on atoms referring to states, time points and traces. A *state* of a process for (state) ontology Ont is an assignment of truth values to the set of ground atoms in the ontology. The set of all possible states for ontology Ont is denoted by STATES(Ont). To describe sequences of states, a fixed *time frame* T is assumed which is linearly ordered. A *trace* γ over state ontology Ont and time frame T is a mapping $\gamma : T \rightarrow \text{STATES(Ont)}$, i.e., a sequence of states γ_t $(t \in T)$ in STATES(Ont). The set of *dynamic properties* DYNPROP(Ont) is the set of temporal statements that can be formulated with respect to traces based on the state ontology Ont in the following manner. Given a trace γ over state ontology Ont, the state in γ at time point t is denoted by state(γ, t). These states can be related to state properties via the formally defined satisfaction relation $\models$, comparable to the Holds-predicate in the Situation Calculus [Reiter (2001)]: $\text{state}(\gamma, t) \models p$ denotes that state property p holds in trace γ at time t. Based on these statements, dynamic properties can be formulated in a sorted first-order predicate logic, using quantifiers over time and traces and the usual first-order logical connectives such as $\neg$, $\wedge$, $\vee$, $\Rightarrow$, $\forall$, $\exists$. A special software environment has been developed for TTL, featuring both a Property Editor for building and editing TTL properties and a Checking Tool that enables formal verification of such properties against a set of (simulated or empirical) traces.

Executable Format To specify simulation models and to execute these models, the language LEADSTO, an executable sublanguage of TTL, is used. The basic building blocks of this language are causal relations of the format $\alpha \twoheadrightarrow_{e,f,g,h} \beta$, which means:

if state property α holds for a certain time interval with duration g,
then after some delay (between e and f) state property β will hold
for a certain time interval of length h.

where α and β are state properties of the form ‘conjunction of literals’ (where a literal is an atom or the negation of an atom), and e, f, g, h non-negative real numbers.

3.3 Global Structure of the Agent-Based Generic Model

For the global structure of the model, first a distinction is made between those components that are the subject of the system (e.g., a patient to be taken care of), and those that are ambient, supporting components. Moreover, from an agent-based perspective (see, for example, [Brazier *et al.* (2000)], [Brazier *et al.* (2002)]), a distinction is made between active, agent components (human or artificial), and passive, world components (e.g., part of the physical world or a database). Furthermore, within an agent a mind may be distinguished from a physical body. This provides the types of components distinguished shown in Figure 3.1. Here the dotted rectangles depict agents with mind and body distinguished within them, and the other geometrical shapes denote world components. Given the distinctions made between components, interactions between such components are of different types as well. Figure 3.1 depicts a number of possible interactions by the arrows. Table 3.1 shows an overview of the possible interactions.

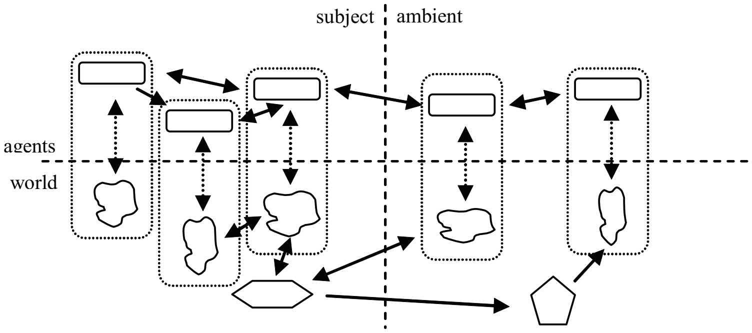

Fig. 3.1 Different types of components and interactions

Interaction Between Agents Interaction between two agents may be communication or bodily interaction, for example, fighting. When within the agent a distinction is made between mind and body, communication can be modelled as information transfer between an agent's mind and another agent's mind. Whether for a given application of the generic model, within agents a mind and a body are distinguished, depends on the assumptions made about the application domain. If it is assumed that communication is independent of and cannot be affected by other processes in the world, then communication can most efficiently be modelled as information transfer between minds. If, in contrast, it is to be modelled how communication is affected by other processes in the world (e.g., effects on the quality of a channel or network), then it is more adequate to model communication as bodily interaction. Obviously, also in cases that it is to be modelled how agents affect each others bodies, as in fighting, the latter is the most adequate option.

Agent-World Interaction Interaction between an agent and a world component can be either observation or action performance; cf. [Brazier *et al.* (2000)]. An action is generated by an agent, and transfers to a world component to have its effect there. An observation has two directions: the observation focus is generated by an agent and transfers to a world component (providing access to a certain aspect of the world), and the provision of the observation result is generated by the world component and transfers to the agent. Combinations of interactions are possible, such as performing an action and observing the effect of the action afterwards. When the agent's body is distinguished from its mind, interaction between agent and world can be modelled as transfer between this body and a world component. In addition, interaction between the agent's mind and its body (the vertical arrows in Figure 1) can be used to model the effect of mental processes (deciding on actions and observations to undertake) on the agent-world interaction and vice versa (incorporating observation results). Also here, whether for a given application of the generic model interaction between an agent and the world is modelled according to the first or the second option, depends on the assumptions made about the application domain. If it is assumed that performance of an intended action generated by the mind has a direct effect on the world and has no relevant effect on an agent's body, then it can most efficiently be modelled according to the first option. If, in contrast, it is to be modelled how actions and observations are also affected by other processes in the body or world, then the second option is more adequate. Also in cases that it is to be modelled how the world affects an agent body, obviously the second option is the most adequate option.

Table 3.1 Different types of interaction

to **from**	*subject* *agent*	*subject* *world component*	*ambient* *agent*	*ambient* *world component*
subject agent	subject communication; subject body interaction	subject observation focus; subject action performance; subject body-world interaction	subject-ambient communication; subject-ambient body interaction	subject-ambient observation focus; subject-ambient action performance; subject-ambient body-world interaction
subject world component	subject observation result; subject world-body interaction	subject world component interaction	subject-ambient observation result; subject-ambient world-body interaction	subject-ambient world component interaction
ambient agent	ambient-subject communication; ambient-subject body interaction	ambient-subject observation focus; ambient-subject action performance; ambient-subject body - world interaction	ambient communication; ambient body interaction	ambient observation focus; ambient action performance; ambient body-world interaction
ambient world component	subject-ambient observation result; ambient-subject world-body interaction	ambient-subject world component interaction	ambient observation result; ambient world-body interaction	ambient world component interaction

The naval officer example Table 3.2 illustrates the different types of components and interactions for a case concerning a naval officer, as briefly explained in the introduction. The officer keeps track of incoming planes on a radar screen, and acts on those ones classified as dangerous.

Generic State Ontologies at the Global Level For the information exchanged between components at the global level, generic ontologies have been specified. This has been done in a universal order-sorted predicate logic format that easily can be translated into more specific ontology languages. Table 3.3 provides an overview of the generic sorts and

Table 3.2 Components and interactions for a naval officer case

subject components	**subject agents**	**subject world components**
	human naval officer	radar screen with moving planes
subject interactions	**observation and action by subject agent**	
	naval officer gaze focuses on radar screen with planes, extracts information from radar screen view, naval officer acts on planes that are dangerous	
ambient components	**ambient agents**	
	dynamic task allocation agent (including an eye tracker), task-specific agent	
ambient interactions	**communication between ambient agents**	
	communication between task allocation agent and task-specific agent on task requests	
interactions between subject and ambient	**communication**	**observation and action**
	task allocation agent communicates over-looked dangerous item to naval officer	ambient agent has observation focus on radar screen and naval officer gaze ambient agent extracts info from views

predicates used in interactions at the global level. Examples of the use of this ontology will be found in the case studies.

Table 3.3 Generic Ontology for Interaction at the Global Level

SORT	**Description**
ACTION	an action
AGENT	an agent
INFO_EL	an information element, possibly complex (e.g., a conjunction of other info elements)
WORLD	a world component

Predicate	**Description**
performing_in(A:ACTION, W:WORLD)	action A is performed in W
observation_focus_in(I:INFO_EL, W:WORLD)	observation focus is I in W
observation_result_from(I:INFO_EL, W:WORLD)	observation result from W is I
communication_from_to(I:INFO_EL, X:AGENT, Y:AGENT)	information I is communicated by X to Y
communicated_from_to(I:INFO_EL,X:AGENT,Y:AGENT)	information I was communicated by X to Y
can_observe_in(X:AGENT, I:INFO_EL, W:WORLD)	agent X can observe I within world W
can_perform_in(X:AGENT, A:ACTION, W:WORLD)	agent X can perform action A within W
can_communicate_with_about(X:AGENT,Y:AGENT,I:INFO_EL)	agent X can communicate with Y about I

Generic Temporal Relations for Interaction at the Global Level Interaction between global level components is defined by the following specifications. Note that in such specifications, for state properties the prefix input, output or internal is used. This is an indexing of the language elements to indicate that it concerns specific variants of them either present at the input, output or internally within the agent.

Action Propagation from Agent to World Component

$\forall X{:}\mathrm{AGENT}\ \forall W{:}\mathrm{WORLD}\ \forall A{:}\mathrm{ACTION}\ \mathrm{output}(X)|\mathrm{performing\_in}(A, W) \wedge \mathrm{can\_perform\_in}(X,A,W) \twoheadrightarrow$
$\mathrm{input}(W)|\mathrm{performing\_in}(A, W)$

Observation Focus Propagation from Agent to World Component

$\forall X{:}\mathrm{AGENT}\ \forall W{:}\mathrm{WORLD}\ \forall I{:}\mathrm{INFO\_EL}\ \mathrm{output}(X)|\mathrm{observation\_focus\_in}(I, W) \wedge \mathrm{can\_observe\_in}(X,I,W) \twoheadrightarrow$
$\mathrm{input}(W)|\mathrm{observation\_focus\_in}(I, W)$

Observation Result Propagation from World to Agent

$\forall X{:}\mathrm{AGENT}\ \forall W{:}\mathrm{WORLD}\ \forall I{:}\mathrm{INFO\_EL}\ \mathrm{output}(W)|\mathrm{observation\_result\_from}(I, W) \wedge \mathrm{can\_observe\_in}(X,I,W) \twoheadrightarrow$

input(X)|observed_result_from(I, W)

Communication Propagation Between Agents

∀X,Y:AGENT ∀I:INFO_EL output(X)|communication_from_to(I,X,Y) ∧ can_communicate_with_about(X,Y,I) ↠ input(Y)|communicated_from_to(I,X,Y)

3.4 Generic Ambient Agent and World Model

This section focuses on the ambient agents within the generic model. As discussed in Section 3, ambient agents can have various types of interactions. Moreover, they are assumed to maintain knowledge about certain aspects of human functioning in the form of internally represented dynamic models, and information about the current state and history of the world and other agents. Based on this knowledge they are able to have a more in-depth understanding of the human processes, and can behave accordingly. This section presents an ambient agent model that incorporates all these.

Components within the Ambient Agent Model In [Brazier *et al.* (2000)] the component-based Generic Agent Model (GAM) is presented, formally specified in DESIRE [Brazier *et al.* (2002)]. The process control model was combined with this agent model GAM. Within GAM the component *World Interaction Management* takes care of interaction with the world, the component *Agent Interaction Management* takes care of communication with other agents. Moreover, the component *Maintenance of World Information* maintains information about the world, and the component *Maintenance of Agent Information* maintains information about other agents. In the component *Agent Specific Task*, specific tasks can be modelled. Adopting this component-based agent model GAM, the Ambient Agent Model has been obtained as a refinement, by incorporating components of the generic process control model described above.

The component *Maintenance of Agent Information* has three subcomponents. The subcomponent *Maintenance of a Dynamic Agent Model* maintains the causal and temporal relationships for the subject agent's functioning. For example, this may model the relationship between a naval officer's gaze direction, characteristics of an object at the screen, and the attention level for this object. The subcomponent *Maintenance of an Agent State Model* maintains a snapshot of the (current) state of the agent. As an example, this may model the gaze direction, or the level of attention for a certain object at the screen. The subcomponent *Maintenance of an Agent History Model* maintains the history of the (current) state of the agent. This may for instance model the trajectory of the gaze direction, or the level of attention for a certain object at the screen over time.

Similarly, the component *Maintenance of World Information* has three subcomponents for a dynamic world model, a world state model, and a world history model, respectively. Moreover, the component *Agent Specific Task* has the following three subcomponents, devoted to the agent's process control task. The subcomponent *Simulation Execution* extends the information in the agent state model based on the internally represented dynamic agent model for the subject agent's functioning. For example, this may determine the attention level from a naval officer's gaze direction, and the characteristics of an object at the screen, and his previous attention level. The subcomponent *Process Analysis* assesses the current state of the agent. For instance, this may determine that a dangerous item has a level of attention that is too low. This component may use different generic methods of assessment, among which (what-if) simulations and (model-based) diagnostic methods, based on the dynamic and state models as maintained. The subcomponent *Plan Determination* determines whether action has to be undertaken, and, if so, which ones (e.g. to determine that the dangerous item with low attention from the naval officer has to be handled by another agent).

Finally, as in the model GAM, the components *World Interaction Management* and *Agent Interaction Management* prepare (based on internally generated information) and receive (and internally forward) interaction with the world and other agents. Table 3.4 provides an overview of the different components within the Ambient Agent Model, illustrated for the case of the naval officer.

Table 3.4 Components within the Ambient Agent Model

Maintenance of Agent Information	
maintenance of dynamic models	model relating attention state to human body state and world state
maintenance of state models *subject agent* *subject world component*	model of attention state and gaze state of the naval officer model of state of radar screen with planes
maintenance of history models	model of gaze trajectory and attention of time
Maintenance of World Information (similar to Maintenance of Agent Information)	
Agent Specific Task	
simulation execution	update the naval officer's attention state from gaze and radar screen state
process analysis	determine whether a dangerous item is overlooked
plan determination	determine an option to address overlooked dangerous items (to warn the naval officer, or to allocate another human or ambient agent to this task)
World Interaction Management	processing received observation results of screen and gaze
Agent Interaction Management	preparing a warning to the officer preparing a request to take over a task

Generic State Ontologies within Ambient Agent and World To express the information involved in the agent's internal processes, the ontology shown in Table 3.5 was specified.

Table 3.5 Generic Ontology used within the Ambient Agent Model

Predicate	**Description**
belief(I:INFO_EL)	information I is believed
world_fact(I:INFO_EL)	I is a world fact
has_effect(A:ACTION, I:INFO_EL)	action A has effect I
Function to INFO_EL	**Description**
leads_to_after(I:INFO_EL, J:INFO_EL, D:REAL)	state property I leads to state property J after duration D
at(I:INFO_EL, T:TIME)	state property I holds at time T

As an example belief(leads_to_after(I:INFO_EL, J:INFO_EL, D:REAL)) is an expression based on this ontology which represents that the agent has the knowledge that state property I leads to state property J with a certain time delay specified by D. This can provide enhanced context awareness (in addition to information obtained by sensoring).

Generic Temporal Relations within an Ambient Agent The temporal relations for the functionality within the Ambient Agent are as follows.

Belief Generation based on Observation, Communication and Simulation

∀X:AGENT, I:INFO_EL, W:WORLD input(X)|observed_from(I, W) ∧ internal(X)|belief(is_reliable_for(W, I)) ↠
internal(X)|belief(I)

∀X,Y:AGENT, I:INFO_EL input(X)|communicated_from_to(I,Y,X) ∧ internal(X)|belief(is_reliable_for(X, I)) ↠
internal(X)|belief(I)

∀X:AGENT ∀I,J:INFO_EL ∀D:REAL ∀T:TIME internal(X)|belief(at(I, T)) ∧ internal(X)|belief(leads_to_after(I, J, D)) ↠
internal(X)|belief(at(J, T+D))

Here, the first rule is a generic rule for the component *World Interaction Management*. Similarly, the second rule is a generic rule for the component *Agent Interaction Management*. When the sources are assumed always reliable, the conditions on reliability can be left out of the first two rules. The last generic rule within the agent's component *Simulation Execution* specifies how a dynamic model that is explicitly represented as part of the agent's knowledge (within its component *Maintenance of Dynamic Models*) can be used to perform simulation, thus extending the agent's beliefs about the world state at different points in time. This can be considered an internally represented deductive causal reasoning method. As another option, an abductive causal reasoning method can be internally represented in a simplified form as follows.

Belief Generation based on Simple Abduction

∀X:AGENT ∀I,J:INFO_EL ∀D:REAL ∀T:TIME
internal(X)|belief(at(J, T)) ∧ internal(X)|belief(leads_to_after(I, J, D)) ↠ internal(X)|belief(at(I, T-D))

Generic Temporal Relations within a World For World Components the following specifications indicate the actions' effects and how observations provide their results.

Action Execution and Observation Result Generation in the World

∀W:WORLD_COMP ∀A:ACTION ∀I:INFO_EL input(W)|performing_in(A, W) ∧ internal(W)|has_effect(A,I) ↠
internal(W)|world_fact(I)

∀W:WORLD_COMP ∀I:INFO_EL input(W)|observation_focus_in(I, W) ∧ internal(W)|world_fact(I) ↠
output(W)|observation_result_from(I, W)

∀W:WORLD_COMP ∀I:INFO_EL input(W)|observation_focus_in(I, W) ∧ internal(W)|world_fact(not(I))↠
output(W)|observation_result_from(not(I), W)

3.5 Case Study 1: An Ambient Driver Support System

One of the application cases addressed to evaluate the applicability of the generic model is an ambient driver support system (see Table 3.6 and Figure 3.2). This example was inspired by a system that is currently under development by Toyota. It is a fail-safe system for cars that analyses whether a driver is drunk and in that case automatically shuts the vehicle down. The system uses sensors that analyse sweat on the palms of the driver's hands to assess the blood alcohol level and does not allow the vehicle to be started if the reading is above specified safety limits. The system can also kick in if sensors detect abnormal steering operations, or if a special camera shows that the driver's gaze is not focused. The car is then slowed to a halt. The system makes use of a dynamic model of a driver's functioning expressing that a high alcohol level in the blood leads to measurable alcohol in the sweat, and to observable behaviour showing abnormal steering operation and unfocused gaze.

Table 3.6 Components and Interactions of the Ambient Driver Support System

subject components	**subject agents**	**subject world components**
	human driver	car and environment
subject interactions	**observation and action by subject agent in subject world component**	
	driver observes car and environment, operates car and gaze	
ambient components	**ambient agents**	
	steering, gaze-focus, and alcohol-level sensoring agent; steering, gaze-focus, and alcohol level monitoring agent; driver assessment agent, cruise control agent	
ambient interactions	**communication between ambient agents**	
	steering sensoring agent communicates to steering monitoring agent gaze-focus sensoring agent communicates gaze focus to gaze-focus monitoring agent alcohol-level sensoring agent communicates to alcohol-level monitoring agent alcohol level monitoring agent reports to driver assessment agent alcohol level eye-focus monitoring agent reports to driver assessment agent unfocused gaze steering monitoring agent reports to driver assessment agent abnormal steering driver assessment agent communicates to cruise control agent state of driver	
interactions between subject and ambient	**observation and action by ambient agent in subject world component**	
	steering sensoring agent observes steering wheel operation gaze-focus sensoring agent observes driver body gaze focus alcohol-level sensoring agent measures alcohol level in sweat of driver hand palms cruise control agent slows down car or stops engine	

For the ambient driver support case, several domain specific rules have been identified in addition to the generic rules specified in Section 3.3 and 3.4. Some of the key rules are expressed below. For all domain specific rules, see Appendix 3.10. First of all, within the Driver Assessment Agent an explicit representation is present of a dynamic model of the driver's functioning.

In this model it is represented how a high alcohol level in the blood has physiological and behavioural consequences that can be observed: (1) physiological: a high alcohol level in the sweat, (2) behavioural: abnormal steering operation and an unfocused gaze. The dynamic model is represented by the following beliefs in the component *Maintenance of Dynamic Models*.

```
internal(driver_assessment_agent)|belief(leadsto(alcohol_level_high, driver_assessment(negative), D))
internal(driver_assessment_agent)|belief(leadsto(abnormal_steering_operation ∧ unfocused_gaze,
driver_assessment(negative), D))
```

The Driver Assessment Agent receives this observable information from the various monitoring agents, of which the precise specification has been omitted for the sake of brevity. By the simple abductive reasoning method specified by the generic temporal rule in Section 3.4, when relevant the Driver Assessment Agent can derive that the driver has a high alcohol level, from which the agent concludes that the driver assessment is negative. These

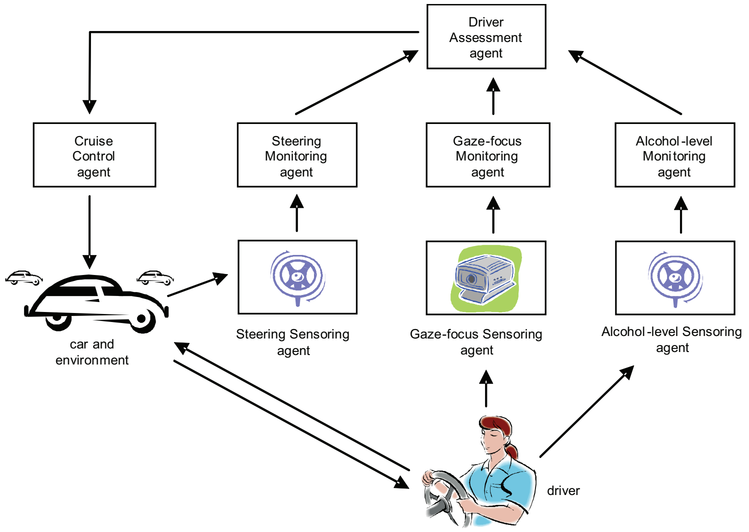

Fig. 3.2 Case Study: Ambient Driver Support System

are stored as beliefs in the component *Maintenance of an Agent State Model* and communicated to the Cruise Control Agent. The Cruise Control Agent takes the appropriate measures. The first temporal rule specifies that if the driver assessment is negative, and the car is not driving, then the ignition of the car is blocked:

internal(cruise_control_agent)|belief(driver_assessment(negative)) ∧
internal(cruise_control_agent)|belief(car_is_not_driving)
↠ output(cruise_control_agent)|performing_in(block_ignition, car_and_environment)

If the car is already driving, whereas the assessment is negative, the car is slowed down.

internal(cruise_control_agent)|belief(driver_assessment(negative)) ∧
internal(cruise_control_agent)|belief(car_is_driving)
↠ output(cruise_control_agent)|performing_in(slow_down_car, car_and_environment)

Based upon such temporal rules, simulation runs of the system have been generated, of which an example trace is shown in Figure 3.3. In the figure, the left side indicates the atoms that occur during the simulation whereas the right side indicates a time line where a dark box indicates the atom is true at that time point and a grey box indicates false.

In the trace, the initial alcohol level in the sweat is 0.4 per mille which is below the maximum allowed level of 0.5 per mille.

internal(alcohol_level_sensoring_agent)|observed_result_from(alcohol_level(0.4), driver)

The driver starts the car and accelerates, resulting in a driving car.

internal(car_and_environment)|world_fact(car_driving)

However, after a while the driver's alcohol level rises to 0.6 per mille, which is classified as high by the Alcohol Level Monitoring Agent, and this is communicated to the Driver

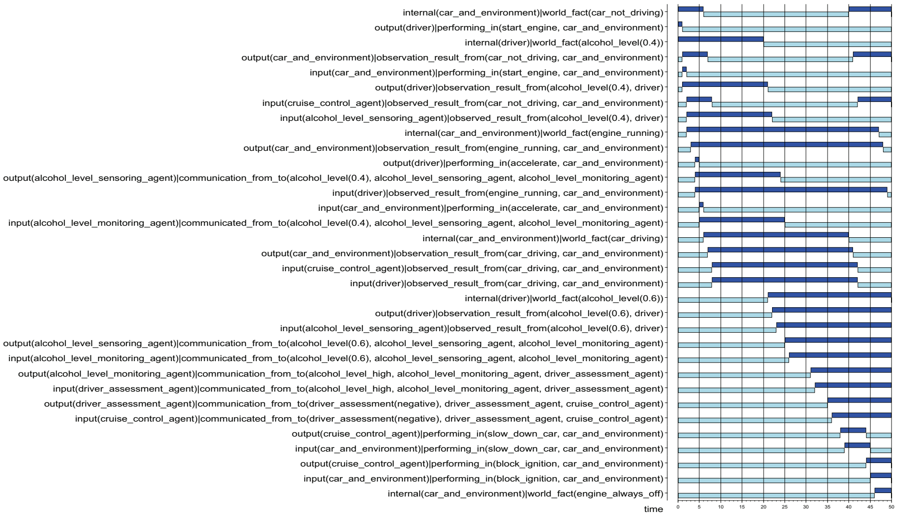

Fig. 3.3 Example simulation trace of the ambient driver support system

Assessment Agent:

input(driver_assessment_agent)|communicated_from_to(alcohol_level_high, alcohol_level_monitoring_agent, driver_assessment_agent)

By the abductive reasoning method this agent assesses the driver as negative, which is communicated to the Cruise Control Agent, which starts to intervene. First it slows down the car, and after it stopped, the agent blocks the ignition:

output(cruise_control_agent)|performing_in(slow_down_car, car_and_environment)
output(cruise_control_agent)|performing_in(block_ignition, car_and_environment)

A more elaborated description of the model and the simulation results can be found in [Bosse *et al.* (2008a)].

3.6 Case Study 2: Ambient Aggression Handling System

This case study is inspired by a system which is operational in the city of Groningen, the Netherlands (see Table 3.7 and Figure 3.4). It makes use of a camera that is equipped with a microphone, and mounted at places where aggression could occur, for example in railway stations or near bars in city centres.

Initially the system only records the sound, which is dynamically analysed by an aggression detection system. As soon as this system detects that the recorded sound is different (more aggressive) from the standard background noise, it turns on the camera and warns the officers at the police station. Subsequently, the police can assess the situation

Table 3.7 Components and Interactions of the Ambient Aggression Handling System

subject components	**subject agents**
	persons in crowd
ambient components	**ambient agents**
	camera, microphone, camera control, and sound analysis agent; police officer at station, police officer at street
ambient interactions	**communication between ambient agents**
	camera control agent communicates to camera agent that inspection is needed camera agent communicates pictures of scene to officer at police station microphone agent communicates sound to sound analysis agent sound analysis agent communicates to camera control agent that inspection is needed sound analysis agent communicates to police officer at station that inspection is needed and the sound police officer at station communicates police officer at street that inspection is needed
interactions between subject and ambient	**observation and action by ambient agent in subject world component**
	camera agent observes persons, microphone agent observes sounds police officer at street stops aggression of persons

remotely using the camera pictures, and if necessary, they can send police officers to the place to stop the aggression. Also for the ambient aggression handling system, a number of domain-specific temporal rules have been established. For a complete overview of all domain specific rules, see Appendix 3.11. First of all, the component *Maintenance of Dynamic Models* within the Sound Analysis Agent contains a representation of a dynamic model of aggression and consequences thereof. A main consequence considered here is that aggression leads to sounds and sound levels that deviate from the normal sounds. This is represented as

internal(sound_analysis_agent)|belief(leads_to(aggression_in_crowd, sound(loud), D))

stating that aggression in the crowd leads to sounds with a certain frequency (for simplicity represented as sound(loud)).

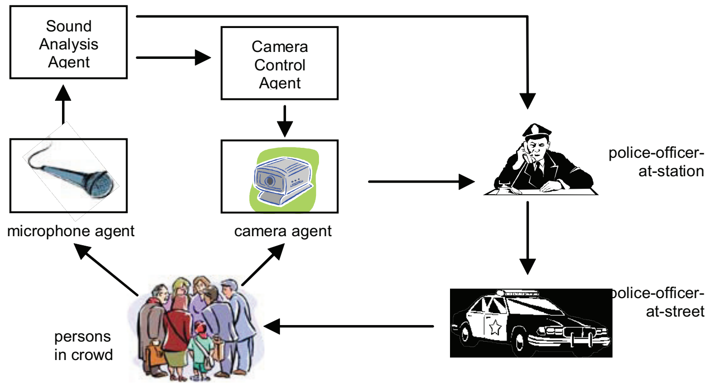

Fig. 3.4 Case Study: Ambient Aggression Handling System

The latter is observable information, so when this comes in, by a simple abductive reasoning method the Sound Analysis Agent concludes a belief that there is aggression in the crowd; this information is transferred to the Camera Control Agent, upon which the latter agent communicates a request for view to the Camera Agent. This is done via the following rule (which is part of the component *Plan Determination* of the Camera Control Agent):

internal(camera_control_agent)|belief(aggression_in_crowd)
↠ output(camera_control_agent)|communication_from_to(inspection_needed, camera_control_agent, camera_agent)

Eventually, when the current sound and the view are perceived, both types of information are transferred to the police officer at the station. For the simulation, this police officer uses the following temporal rule (which is part of the component *Process Analysis* of the police officer) to conclude that there is probably aggression in the crowd:

∀S:SOUND ∀V:VIEW
internal(police_officer_at_station)|belief(inspection_needed) ∧
internal(police_officer_at_station)|belief(sound(S)) ∧
internal(police_officer_at_station)|belief(view(V)) ∧
internal(police_officer_at_station)|belief(sound_view_classification(S, V, aggressive))
↠ internal(police_officer_at_station)|belief(aggression_in_crowd)

If this officer concludes the belief that there is aggression in the crowd, the police officer at the station notifies the police officer at the street that inspection is needed. As a result, this police officer will go to the location of the aggression to observe the actual situation. He will use a similar rule to the one above to conclude that there is indeed aggression, and if this is the case, he will perform the action of stopping the aggression. An example trace that was generated on the basis of these temporal rules is shown in Figure 3.5.

As seen in this trace, from the start of the simulation, there is aggression in the crowd, which is indicated by a loud sound and the view of fighting persons:

internal(persons_in_crowd)|world_fact(sound(loud))
internal(persons_in_crowd)|world_fact(view(fighting_persons))

The Microphone Agent transfers the sound to the Sound Analysis Agent:

output(microphone_agent)|communication_from_to(sound(loud), microphone_agent, sound_analysis_agent)

By simple abductive reasoning the Sound Analysis Agent generates the belief that there aggression, and informs the Camera Control Agent and the police officer at the station:

output(sound_analysis_agent)|communication_from_to(inspection_needed, sound_analysis_agent, camera_control_agent)

output(sound_analysis_agent)|communication_from_to(inspection_needed, sound_analysis_agent, police_officer_at_station)

output(sound_analysis_agent)|communication_from_to(sound(loud), sound_analysis_agent, police_officer_at_station)

Next, the Camera Control Agent informs the Camera Agent that inspection is needed:

output(camera_control_agent)|communication_from_to(inspection_needed, camera_control_agent, camera_agent)

The camera agent observes the fighting persons:

input(camera_agent)|observed_result_from(view(fighting_persons), persons_in_crowd)

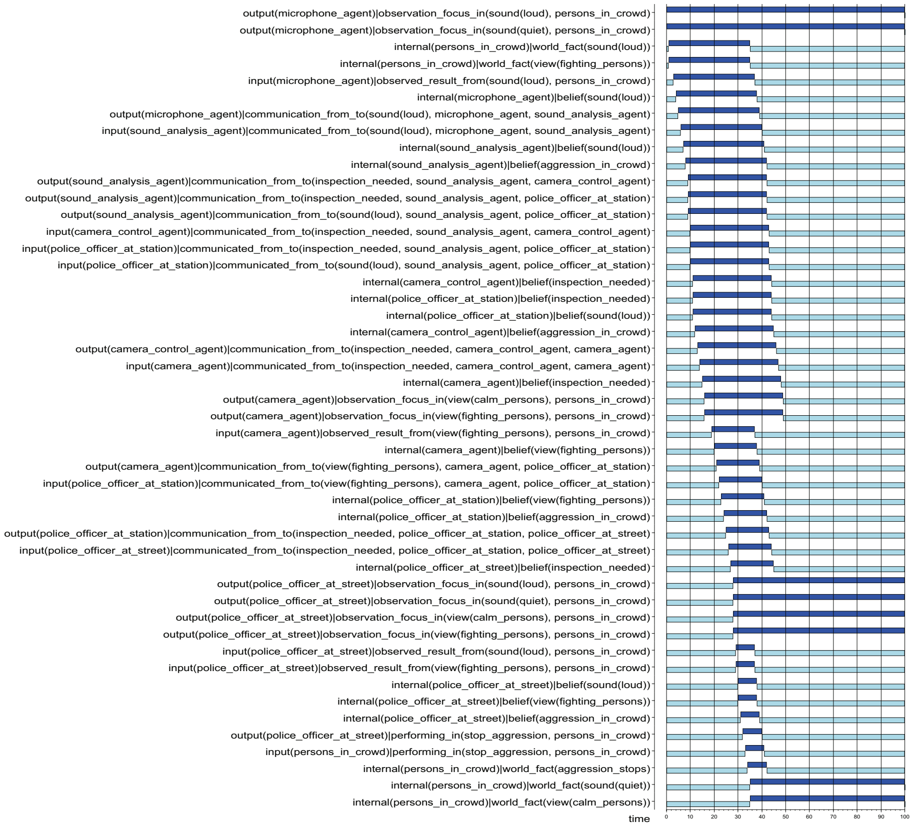

Fig. 3.5 Example simulation trace of the ambient aggression handling system

This information is then transferred to the police officer at the station, who generates the belief that there is aggression in the crowd:

internal(police_officer_at_station)|belief(aggression_in_crowd)

After this, the police officer at the station notifies the police officer at the street that (further) inspection is needed, who confirms that there is indeed aggression in the crowd, and undertakes the action of stopping the aggression (which eventually results in a quiet and calm environment):

output(police_officer_at_street)|performing_in(stop_aggresion, persons_in_crowd)

3.7 Case Study 3: Ambient System for Management of Medicine Usage

Another case used to evaluate the generic model concerns management of medicine usage; e.g., [Green (2005)]. Figure 3.6 presents an overview of the entire system as considered. Two world components are present in this system: the medicine box, and the patient database; the other components are agents. The top right corner shows the patient, who interacts with the medicine box, and communicates with the patient phone.

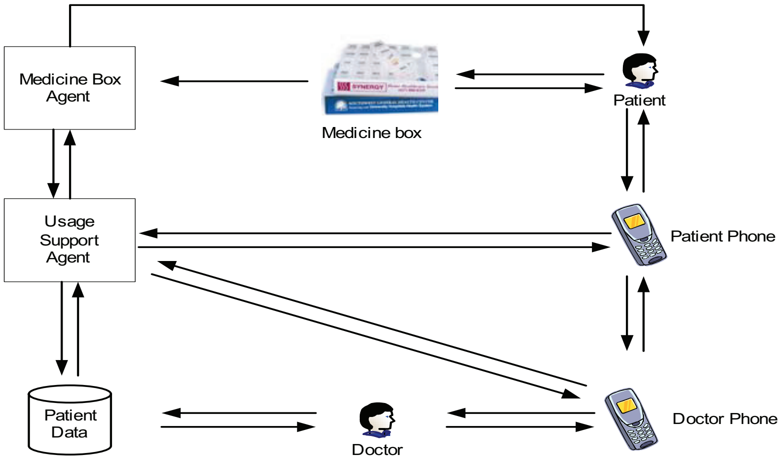

Fig. 3.6 Case Study: Ambient System for Management of Medicine Usage

The (ambient) Usage Support Agent has a dynamic model of the medicine concentration in the patient. This model is used to estimate the current concentration, which is also communicated to the (ambient) Medicine Box Agent. The Medicine Box Agent monitors whether medicine is taken from the box, and the position thereof in the box. In case, for example, the patient intends to take the medicine too soon after the previous dose, it finds out that the medicine should not be taken at the moment (i.e., the sum of the estimated current medicine level plus a new dose is too high), and communicates a warning to the patient by a beep sound. Furthermore, all information obtained by this agent is passed on to the (ambient) Usage Support Agent. All information about medicine usage is stored in the patient database by this agent. If the patient tried to take the medicine too early, a warning SMS with a short explanation is communicated to the cell phone of the patient, in addition to the beep sound already communicated by the Medicine Box Agent. On the other hand, in case the Usage Support Agent finds out that the medicine is not taken on time (i.e., the medicine concentration is estimated too low for the patient and no medicine was taken yet), it can take measures as well. First of all, it can warn the patient by communicating an SMS to the patient cell phone. This is done soon after the patient should have taken the medicine. In case the patient still does not take medicine (for example after a number of hours), the agent can communicate an SMS to the cell phone of the appropriate doctor. The doctor can look into the patient database to see the medicine usage, and in case the doctor feels it is necessary to discuss the state of affairs with the patient, he or she can contact the patient via a call from the doctor cell phone to the patient cell phone. Table 3.8 presents an overview of the various components and their interactions.

The specification of the interaction between the various components within the medicine box case has similarity with the two other cases and has therefore been omitted for the sake of brevity, see Appendix 3.12 for more details. One major difference however is the model the usage support agent has of the patient. The agent maintains a quantitative model of the medicine level of the patient using the following knowledge:

```
internal(usage_support_agent)|belief(leadsto_to (
medicine_level(M, C) ∧ usage_effect(M, E) ∧ decay(M, G),
medicine_level(M, (C+E) - G*(C+E)*D), D)
```

This model basically specifies that a current medicine level C of medicine M and a known usage effect at the current time point of E combined with a decay value of G, leads to a belief of a new medicine level C + E – G * (C + E) * D after duration D. Below, Figure 3.7 shows how the medicine level varies over time when the ambient system support the medicine versus the case where the system is not active. The minimum medicine level required in the patient's blood is 0.3 whereas the maximum allowed medicine level is 1.4. As can be seen, the medicine level using the system does meet these demands whereas without support the level does not.

Table 3.8 Components and interactions of the ambient medicine usage management system

subject components	**subject agents**	**subject world components**
	human patient	medicine box
subject interactions	**observation and action by subject agent in subject world component**	
	patient takes or puts medicine from a particular compartment in medicine box	
ambient components	**ambient agents**	**ambient world components**
	medicine box agent, usage support agent, patient and doctor phone, human doctor	patient database
ambient interactions	**communication between ambient agents**	
	medicine box agent communicates to the medicine usage support agent that a pill has been taken out or added to a compartment of the box medicine usage support agent communicates to the patient cell phone agent a text message that medicine needs to be taken medicine usage support agent communicates to the doctor cell phone agent a text message that a certain patient has not taken his medicine for a certain duration doctor cell phone communicates to patient cell phone (and vice versa) doctor cell phone communicates to the doctor a text message that a certain patient has not take his medicine for a certain duration	
	observation and action by ambient agent in ambient world component	
	usage support agent adds and retrieves info to/from patient database	
interactions between subject and ambient	**communication between ambient agent and subject agent**	**observation by ambient agent in subject world component**
	medicine box agent communicates to patient a warning beep patient communicates with patient phone doctor communicates with doctor phone	medicine box agent focuses on medicine box and receives observation results from medicine box

A more elaborated description of the model and the simulation results can be found in [Hoogendoorn *et al.* (2008)].

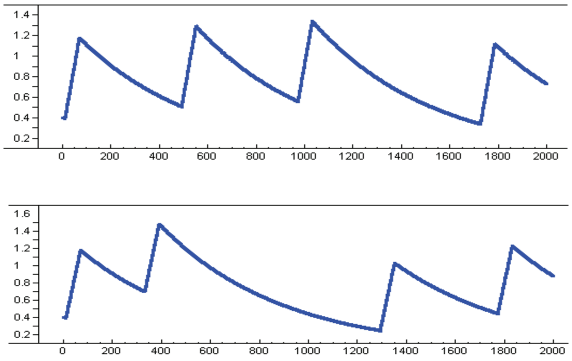

Fig. 3.7 Medicine level with (top figure) and without ambient system usage (x-axis denotes time and y-axis denotes the medicine level, note the different scale)

3.8 Specification and Verification of Dynamic Properties

This section addresses specification and verification of relevant dynamic properties (expressed as formulae in the TTL language) of the cases considered, for example, requirements imposed on these systems. In the future, also the IFIP properties on user interfaces for AmI applications can be tested [Gram and Cockton (1996)].

Properties of the system as a whole A natural property of the Ambient Driver Support System is that a drunken driver cannot continue driving. A driver is considered drunk if the blood alcohol level is above threshold *a*. The global properties (GP) of the presented systems (abbreviated as ADSS, AAHS and AMUMS respectively) are:

GP1(ADSS) No drunken driver
If the driver's blood alcohol level is above threshold *a*, then within 30 seconds the car will not drive and the engine will be off

∀γ:TRACE, t:TIME, R:REAL
state(γ, t, internal(driver)) ⊨ world_fact(alcohol_level(R)) & R > *a*
⇒ ∃t2:TIME < t:TIME + 30 [state(γ, t2, internal(car_and_environment)) ⊨ world_fact(car_not_driving)]

For the Ambient Aggression Handling System a similar property is that aggression is stopped as soon as it occurs. Here for the example a situation is considered aggressive if persons are fighting and there is a high sound level.

GP1(AAHS) No aggression
If the persons in the crowd are fighting and noisy, then within 35 time steps they will be calm and quite

∀γ:TRACE, t:TIME

state(γ, t, internal(persons_in_crowd)) ⊨ world_fact(view(fighting_persons)) &
state(γ, t, internal(persons_in_crowd)) ⊨ world_fact(sound(loud))
⇒ ∃t2:TIME < t:TIME + 35 [state(γ, t, internal(persons_in_crowd)) ⊨ world_fact(view(calm_persons)) &
state(γ, t, internal(persons_in_crowd)) ⊨ world_fact(sound(quiet))]

For the Ambient Medicine Usage Management System, a relevant property is that the medicine concentration is relatively stable, which means that it stays between an upper and lower bound.

GP1(AMUMS) Stable Medicine Concentration
At any time point the medicine concentration is between lower bound M1 and upper bound M2

∀γ:TRACE, t:TIME, R:REAL
state(γ, t, internal(patient)) ⊨ world_fact(medicine_level(R)) ⇒ M1 ≤ R & R ≤ M2

All three of these properties have been automatically verified (using the TTL checker tool [Bosse *et al.* (2008b)]) against the traces shown in the paper. For each of these trace whereby the system is in use, the property GP1 holds.

Interlevel Relations Between Properties at Different Aggregation Levels Following [Jonker and Treur (2002)], dynamic properties can be specified at different aggregation levels. Illustrated for the Driver case, three levels are used: properties of system as a whole, properties of subsystems, and properties of agents and the world within a subsystem. In Table 3.9 it is shown for the Ambient Driver Support System how the property at the highest level relates to properties at the lower levels (see also Figure 2). The lower level properties in the fourth column are described below.

Table 3.9 Properties and their interlevel relations

subsystems and properties		**components and properties**	
sensoring	S1	steering, gaze-focus, alcohol-level sensoring agent	SSA1, GSA1, ASA1
monitoring	M1	steering, gaze-focus, alcohol-level monitoring agent	SMA1, GMA1, AMA1
assessment	A1	driver assessment agent	DAA1, DAA2
plan determination	P1	cruise control agent	CCA1, CCA2
subject process	SP1	driver, car and environment	CE1, CE2

The property GP1 of the system as a whole can be logically related to properties of the subsystems (shown in the second column in the table) by the following inter level relation:

S1 & M1 & A1 & P1 & SP1 ⇒ GP1

This expresses that the system functions well when all of the subsystems for sensoring, monitoring, assessment, plan determination and the subject process function well.

Properties of subsystems The properties characterising correct functioning of each of the subsystems are described below.

S1 Sensoring system
If the sensory system receives observation input from the world and driver concerning alcohol level, gaze focus and steering operation, then it will provide as output this information for the monitoring

system.

M1 Monitoring system
If the monitoring system receives sensor information input concerning alcohol level, gaze-focus and steering operation from the sensoring system, then it will provide as output monitoring information concerning qualification of alcohol-level, gaze-focus and steering operation for the assessment system.

A1 Assessment system
If this system receives monitoring information concerning specific qualifications of alcohol-level, gaze-focus and steering operation, then it will provide as output a qualification of the state.

P1 Plan determination system
If the plan determination system receives an overall qualification of the state, then it will generate as output actions to be undertaken.

SP1 Subject process
If the subject process receives actions to be undertaken, then it will obtain the effects of these actions. If the driver's blood alcohol level is above threshold *a*, then the driver will operate the steering wheel abnormally and the driver's gaze is unfocused.

Properties of components As indicated in Table 3.9 in the fourth column, each property of a subsystem is logically related to properties of the components within the subsystem. For example, the inter level relation

SSA1 & GSA1 & ASA1 ⇒ S1

expresses that the sensoring subsystem functions well when each of the sensoring agents functions well (similarly, for the monitoring subsystem). Examples of properties characterising correct functioning of components are the following. The properties for the other sensoring and monitoring agents (GSA1, ASA1, GMA1, AMA1) are similar.

SSA1 Steering Sensoring agent
If the Steering Sensoring agent receives observation results about steering wheel operation then it will communicate this information to the Steering Monitoring agent observation.

SMA1 Steering Monitoring agent
If the Steering Monitoring agent receives observation results about the steering wheel, and this operation is abnormal, then it will communicate to the Driver Assessment Agent that steering operation is abnormal.

The properties for the Driver Assessment Agent are:

DAA1 Assessment based on alcohol
If the Driver Assessment Agent receives input that the alcohol level is high, then it will generate as output communication to the Cruise Control agent that the driver state is inadequate.

DAA2 Assessment based on behaviour
If the Driver Assessment Agent receives input that steering operation is abnormal and gaze is unfocused, then it will generate as output communication to the Cruise Control agent that the driver state is inadequate.

For the Cruise Control Agent the properties are:

CCA1 Slowing down a driving car

If the Cruise Control agent receives communication to that the driver state is inadequate, and the car is driving, then it will slow down the car.

CCA2 Turning engine off for a non driving car

If the Cruise Control agent receives communication that the driver state is inadequate, and the car is not driving, then it will turn off the engine.

The properties for the Car and Environment are:

CE1 Slowing down stops the car

If the Car and Environment components perform the slowing down action, then within 20 seconds the car will not drive.

CE2 Turning off the engine makes the engine off

If the Car and Environment components perform the turn off engine action, then within 5 seconds the engine will be off.

The Use of Interlevel Relations in Fault Diagnosis Sometimes an error might occur in a component within the system. Therefore, a trace has also been generated whereby the functioning of the various agents is correct with a certain probability. In the resulting trace, the overall property GP1 does not hold. Therefore, the refined properties have been verified to determine the exact cause of this failure, and the results thereof show that the alcohol level monitoring agent does not communicate that the alcohol level is high, whereas the level is in fact too high.

3.9 Discussion

The challenge addressed in this paper is to provide a generic model that covers the class of Ambient Intelligence applications that show human-like understanding and supporting behaviour. Here human-like understanding is defined as understanding in the sense of being able to analyse and estimate what is going on in the human's mind and body (a form of mind/bodyreading). Input for these processes are observed information about the human's physiological and behavioural states and dynamic models for the human's physical and mental processes. For the mental side such a dynamic model is sometimes called a Theory of Mind (e.g., [Dennett (1987a)], [Gaerdenfors (2003)], [Goldman (2006)]) and may cover concepts such as emotion, attention, intention, and belief. This can be extended to integration with the human's physical processes, relating, for example, to skin conditions, heart rates, and levels of blood sugar, insulin, adrenalin, testosterone, serotonin, and specific medication taken. In this class of Ambient Intelligence applications, knowledge from human-directed disciplines is exploited, in order to take care of (and support in a knowledgeable manner) humans in their daily living, in medical, psychological and social respects. Thus, an ambience is created that uses essential knowledge from the human-directed disciplines to provide a more human-like understanding of human functioning, and from this understanding can provide adequate support. This may concern, for example, elderly people, criminals and psychiatric patients, but also humans in highly demanding tasks.

The generic model introduced in this paper is a template for the specific class of Ambient Intelligence applications as described. One of the characteristics of this class is that a high level of human-directed context awareness plays a role; see also [Schmidt (2005)], [Schmidt *et al.* (1999)], [Schmidt *et al.* (2001)]. The ambient software and hardware design is described in an agent-based manner at a conceptual design level and to support context awareness has generic facilities built in to represent human state models and dynamic process models, and methods for model-based simulation and analysis on the basis of such models. For a particular application, biomedical, neurological, psychological and/or social ontologies, knowledge and dynamic models about human functioning can be specified. The generic model includes slots where such application-specific content can be filled in to get an executable design for a working system. This specific content, together with the generic methods to operate on it, enables ambient agents to show human-like understanding of humans and to react on the basis of this understanding in a knowledgeable manner. The model has been positively evaluated in three case studies related to existing Ambient Intelligence applications that already are operational or in a far stage of development.

3.10 Driver Case

Driver Assessment agent: Domain-Specific Temporal Rules

The Driver Assessment agent will believe that the driver's state is assessed as negative either when it believes the alcohol level is high, or when it believes that the driver's gaze is unfocused and his steering operations are abnormal.

internal(driver_assessment_agent)|belief(leadsto(alcohol_level_high, driver_assessment(negative), D))

internal(driver_assessment_agent)|belief(leadsto(abnormal_steering_operation ∧ unfocused_gaze, driver_assessment(negative), D))

If the Driver Assessment agent believes that the driver's state is assessed as negative, then it will communicate this to the Cruise Control agent.

internal(driver_assessment_agent)|belief(driver_assessment(negative))
↠
output(driver_assessment_agent)|communication_from_to(driver_assessment_negative, driver_assessment_agent, cruise_control_agent)

Cruise Control agent: Domain-Specific Temporal Rules

If the Cruise Control agent believes that the driver's state is assessed as negative, and in case the car is not driving it will stop the engine, and in case the car is driving, it will slow down the car.

internal(cruise_control_agent)|belief(driver_assessment(negative)) ∧
internal(cruise_control_agent)|belief(car_is_not_driving)

↠
output(cruise_control_agent)|performing_in(block_ignition, car_and_environment)

internal(cruise_control_agent)|belief(driver_assessment(negative)) ∧
internal(cruise_control_agent)|belief(car_is_driving)
↠
output(cruise_control_agent)|performing_in(slow_down_car, car_and_environment)

Steering Monitoring agent: Domain-Specific Temporal Rules

When the steering operation is classified as abnormal, the Steering Monitoring agent will believe that there is abnormal steering operation.

internal(steering_monitoring_agent)|belief(steering_operation_classification(S, abnormal))
↠
internal(steering_monitoring_agent)|belief(leadsto(steering_operation(S), abnormal_steering_operation, D))

When the Steering Monitoring agent believes that there is abnormal steering operation, it will communicate this to the Driver Assessment agent.

internal(steering_monitoring_agent)|belief(abnormal_steering_operation)
↠
outputl(steering_monitoring_agent)|communication_from_to(abnormal_steering_operation, steering_monitoring_agent, driver_assessment_agent)

Steering Sensoring agent: Domain-Specific Temporal Rules

When the steering sensoring agent observes steering operation, it will communicate this to the steering monitoring agent.

input(steering_sensoring_agent)|observed_result(steering_operation(S), driver_body)
↠
output(steering_sensoring_agent)|communication_from_to(steering_operation(S), camera_agent, gaze_focus_monitoring_agent)

Gaze-focus Monitoring agent: Domain-Specific Temporal Rules

When the gaze focus is classified as unfocused, the Steering Monitoring agent will believe that there is unfocused gaze.

internal(gaze_focus_monitoring_agent)|belief(gaze_classification(G, unfocused))
↠
internal(gaze_focus_monitoring_agent)|belief(leadsto(gaze_focus(G), unfocused_gaze, D))

When the Steering Monitoring agent believes that there is unfocused gaze, it will communicate this to the Driver Assessment agent.

internal(gaze_focus_monitoring_agent)|belief(unfocused_gaze)
↠
output(gaze_focus_monitoring_agent)|communication_from_to(unfocused_gaze, gaze_focus_monitoring_agent, driver_assessment_agent)

Gaze-focus Sensoring agent: Domain-Specific Temporal Rules

When the Gaze focus sensoring agent observes a gaze focus, it will communicate this to the Gaze-Focus Monitoring agent.

input(gaze_focus_sensoring_agent)|observed_result(gaze_focus(G), driver_body)
↠
output(gaze_focus_sensoring_agent)|communication_from_to(gaze_focus(G), gaze_focus_sensoring_agent,
gaze_focus_monitoring_agent)

Alcohol-level Monitoring agent: Domain-Specific Temporal Rules

When the alcohol level is classified as high, the Alcohol-Level Monitoring agent will believe that there is a high alcohol level.

internal(alcohol_level_monitoring_agent)|belief(alcohol_level_classification(A, high))
↠
internal(alcohol_level_monitoring_agent|belief(leadsto(alcohol_level(A), alcohol_level_high, D))

When the Alcohol-Level Monitoring agent believes that there is a high alcohol level, it will communicate this to the Driver Assessment agent.

internal(alcohol_level_monitoring_agent)|belief(alcohol_level_high)
↠
output(alcohol_level_monitoring_agent)|communication_from_to(alcohol_level_high, alcohol_level_monitoring_agent,
driver_assessment_agent)

Alcohol Sensoring agent: Domain-Specific Temporal Rules

When the Alcohol Sensoring agent observes an alcohol level, it will communicate this to the Alcohol-level Monitoring agent.

input(alcohol_sensoring_agent)|observed_result(alcohol_level(A))
↠
output(alcohol_sensoring_agent)|communication_from_to(alcohol_level(A), alcohol_sensoring_agent,
alcohol_level_monitoring_agent)

Driver: Domain-Specific Temporal Rules

The driver is characterised by the steering operations, the gaze focus and the alcohol level.

output(driver_body)|performing_in(steering_operation(S), car_and_environment)
output(driver_body)|performing_in(start_engine, car_and_environment)
output(driver_body)|performing_in(accelerate, car_and_environment)
internal(driver_body)|world_fact(gaze_focus(G))
internal(driver_body)|world_fact(alcohol_level(A))

Car and Environment: Domain-Specific Temporal Rules

Steering operations can be performed upon the car.

input(car_and_environment)|performing_in(steering_operation(S), car_and_environenment)
↠
internal(car_and_environment)|world_fact(steering_operation(S))

The action of slowing down the car has the effect that the car is not driving anymore. The effect of stopping the engine has the effect that the engine is off.

internal(car_and environment)|has_effect(slow_down_car, car_not_driving)
internal(car_and environment)|has_effect(block_ignition, engine_off)
internal(car_and environment)|has_effect(¬block_ignition ∧ start_engine, engine_running)
internal(car_and environment)|has_effect(engine_running ∧ accelerate, car_driving)

3.11 Aggression Handling Case

Sound Analysis agent: Domain-Specific Temporal Rules

The Sound Analysis believes that aggression in the crowd leads to a loud sound (within duration D).

internal(sound_analysis_agent)|belief(leadsto(aggression_in_crowd), sound(loud), D)

When the Sound Analysis agent believes that there is aggression in the crowd, it will communicate this to the camera control agent and to the police officer at the station, together with the sound.

internal(sound_analysis_agent)|belief(aggression_in_crowd ∧ sound(loud))

↠

output(sound_analysis_agent)|communication_from_to(inspection_needed, sound_analysis_agent, camera_control_agent) ∧

output(sound_analysis_agent)|communication_from_to(inspection_needed, sound_analysis_agent, police_officer_at_station) ∧

output(sound_analysis_agent)|communication_from_to(sound(loud), sound_analysis_agent, police_officer_at_station)

Camera Control agent: Domain-Specific Temporal Rules

When the Camera Control agent believes that inspection is needed, it will believe that there is aggression in the crowd.

internal(camera_control_agent)|belief(inspection_needed)

↠

internal(camera_control_agent)|belief(aggression_in_crowd)

When the Camera Control agent believes that there is aggression in the crowd, it will communicate to the Camera agent that inspection is needed.

internal(camera_control_agent)|belief(aggression_in_crowd)

↠

output(camera_control_agent) |communication_from_to(view_needed, camera_control_agent, camera_agent)

Microphone agent: Domain-Specific Temporal Rules

The Microphone can observe the sound in the crowd.

output(microphone_agent)|observation_focus_in(sound(S), persons_in_crowd)

When the Microphone agent believes that there is a certain type of sound, it will communicate this to the Sound Analysis agent.

internal(microphone_agent)|belief(sound(S))

↠

output(microphone_agent)|communication_from_to(sound(S), microphone_agent, sound_analysis_agent)

Camera agent: Domain-Specific Temporal Rules

When the Camera believes that inspection is needed, it will focus its observation for this view.

```
internal(camera_agent)|belief(inspection_needed)
↠
output(camera_agent)|observation_focus_in(view(V), persons_in_crowd)
```

When the Camera agent believes that there is a certain type of view, it will communicate this to the Police Officer at the Station.

```
internal(camera_agent)|belief(view(V))
↠
output(camera_agent)|communication_from_to(view(V), camera_agent, police_officer_at_station)
```

Persons in Crowd: Domain-Specific Temporal Rules

The Persons in the Crowd are characterised by the sound and view (that are generated at random).

```
internal(persons_in_crowd)|world_fact(sound(S))
internal(persons_in_crowd)|world_fact(view(V))
```

The action stop aggression has as effect a quiet sound.

```
internal(persons_in_crowd)|has_effect(stop_aggression, sound(quiet))
```

Police Officer at Station: Domain-Specific Temporal Rules

The Police Officer at the Station believes that a loud sound combined with a view of fighting persons is an indication for aggression.

```
internal(police_officer_ at_station)|belief(sound_view_classification(loud, fighting_persons, aggressive))
```

When Police Officer at the Station believes that inspection is needed, and (s)he classifies the combination of sound and view as aggressive, (s)he will believe that there is aggression.

```
internal(police_officer_ at_station)|belief(inspection_needed) ∧
internal(police_officer_at_station)|belief(sound(S)) ∧
internal(police_officer_at_station)|belief(view(V)) ∧
internal(police_officer_at_station)|belief(sound_view_classification(S, V, aggressive))
↠
internal(police_officer_ at_station)|belief(aggression_in_crowd)
```

When Police Officer at the Station believes that there is aggression, (s)he will communicate to the Police Officer at the street that inspection is needed.

```
internal(police_officer_at_station)|belief(aggression_in_crowd)
↠
output(police_officer_at_station)|communication_from_to(inspection_needed,                police_officer_at_station,
police_officer_at_street)
```

Police Officer at Street: Domain-Specific Temporal Rules

The Police Officer at the Street believes that a loud sound combined with a view of fighting persons is an indication for aggression.

internal(police_officer_at_street)|belief(sound_view_classification(loud, fighting_persons, aggressive))

When the Police Officer at the Street receives communication from the Police Officer at the Station that there is aggression, the will believe that there may be aggression.

input(police_officer_at_street)|communicated_from_to(aggression_in_crowd, police_officer_at_station, police_officer_at_street)

$\twoheadrightarrow$

internal(police_officer_at_street)|belief(inspection_needed)

When Police Officer at the Street believes that inspection is needed, then (s)he will focus on observing the sound and view in the situation.

internal(police_officer_at_street)|belief(inspection_needed)

$\twoheadrightarrow$

output(police_officer_at_street)|observation_focus_in(sound(S), persons_in_crowd) $\wedge$

output(police_officer_at_street)|observation_focus_in(view(V), persons_in_crowd)

When Police Officer at the Street believes that inspection is needed, and (s)he classifies the combination of sound and view as aggressive, (s)he will believe that there is aggression.

internal(police_officer_at_street)|belief(inspection_needed) $\wedge$

internal(police_officer_at_street)|belief(sound(S)) $\wedge$

internal(police_officer_at_street)|belief(view(V)) $\wedge$

internal(police_officer_at_street)|belief(sound_view_classification(S, V, aggressive))

$\twoheadrightarrow$

internal(police_officer_at_street)|belief(aggression_in_crowd)

When Police Officer at the Street believes that there is aggression, (s)he will stop the aggression.

internal(police_officer_at_street)|belief(aggression_in_crowd)

$\twoheadrightarrow$

output(police_officer_at_street)|performing_in(stop_aggression, persons_in_crowd)

3.12 Medicine Usage Case

Medicine Box Agent

The Medicine Box Agent has functionality concerning communication to both the patient and the Usage Support Agent. First of all, the observed usage of medicine is communicated to the Usage Support Agent in case the medicine is not taken too early, as specified in MBA1.

MBA1: Medicine usage communication

If the Medicine Box Agent has a belief that the patient has taken medicine from a certain position in the box, and that the particular position contains a certain type of medicine M, and taking the medicine does not result in a too high medicine concentration of medicine M within the patient, then the usage of this type of medicine is communicated to the Usage Support Agent.

internal(medicine_box_agent)|belief(medicine_taken_from_position(x_y_coordinate(X,Y))) ∧
internal(medicine_box_agent)|belief(medicine_at_location(x_y_coordinate(X, Y), M)) ∧
internal(medicine_box_agent)|belief(medicine_level(M, C)) ∧
max_medicine_level(maxB) ∧ dose(P) ∧ C + P ≤ maxB
↠
output(medicine_box_agent)|communication_from_to(
medicine_used(M), medicine_box_agent, usage_support_agent)

In case medicine is taken out of the box too early, a warning is communicated by a beep and the information is forwarded to the Usage Support Agent (MBA2 and MBA3).

MBA2: Too early medicine usage prevention

If the Medicine Box Agent has the belief that the patient has taken medicine from a certain position in the box, that this position contains a certain type of medicine M, and taking the medicine results in a too high medicine concentration of medicine M within the patient, then a warning beep is communicated to the patient.

internal(medicine_box_agent)|belief(medicine_taken_from_position(x_y_coordinate(X,Y))) ∧
internal(medicine_box_agent)|belief(medicine_at_location(x_y_coordinate(X, Y), M)) ∧
internal(medicine_box_agent)|belief(medicine_level(M, C)) ∧
max_medicine_level(maxB) ∧ dose(P) ∧ C + P > maxB
↠
output(medicine_box_agent)|communication_from_to(sound_beep, medicine_box_agent, patient)

MBA3: Early medicine usage communication

If the Medicine Box Agent has a belief that the patient was taking medicine from a certain position in the box, and that the particular position contains a certain type of medicine M, and taking the medicine would result in a too high concentration of medicine M within the patient, then this is communicated to the Usage Support Agent.

internal(medicine_box_agent)|belief(medicine_taken_from_position(x_y_coordinate(X,Y))) ∧
internal(medicine_box_agent)|belief(medicine_at_location(x_y_coordinate(X, Y), M)) ∧
internal(medicine_box_agent)|belief(medicine_level(M, C)) ∧
max_medicine_level(maxB) ∧ dose(P) ∧ C + P > maxB
↠
output(medicine_box_agent)|communication_from_to(
too_early_intake_intention, medicine_box_agent, usage_support_agent)

Usage Support Agent

The Usage Support Agent's functionality is described by three sets of temporal rules. First, the agent maintains a dynamic model for the concentration of medicine in the patient over time in the form of a belief about a leads to relation.

USA1: Maintain dynamic model

The Usage Support Agent believes that if the medicine level for medicine M is C, and the usage effect of the medicine is E, then after duration D the medicine level of medicine M is C+E minus G*(C+E)*D with G the decay value.

internal(usage_support_agent)|belief(leadsto_to_after(medicine_level(M, C) ∧
usage_effect(M, E) ∧ decay(M, G), medicine_level(M, (C+E) - G*(C+E)*D), D)

In order to reason about the usage information, this information is interpreted (USA2), and stored in the database (USA3).

USA2: Interpret usage

If the agent has a belief concerning usage of medicine M and the current time is T, then a belief is generated that this is the last usage of medicine M, and the intention is generated to store this in the patient database.

internal(usage_support_agent)|belief(medicine_used(M)) ∧
internal(usage_support_agent)|belief(current_time(T))
↠
internal(usage_support_agent)|belief(last_recorded_usage(M, T) ∧
internal(usage_support_agent)|intention(store_usage(M, T))

USA3: Store usage in database

If the agent has the intention to store the medicine usage in the patient database, then the agent performs this action.

internal(usage_support_agent)|intention(store_usage(M, T))
↠
output(usage_support_agent)|performing_in(store_usage(M, T), patient_database)

Finally, temporal rules were specified for taking the appropriate measures. Three types of measures are possible. First, in case of early intake, a warning SMS is communicated (USA4). Second, in case the patient is too late with taking medicine, a different SMS is communicated, suggesting to take the medicine (USA5). Finally, when the patient does not respond to such SMSs, the doctor is informed by SMS (USA6).

USA4: Send early warning SMS

If the agent has the belief that an intention was shown by the patient to take medicine too early, then an SMS is communicated to the patient cell phone that the medicine should be put back in the box, and the patient should wait for a new SMS before taking more medicine.

internal(usage_support_agent)|belief(too_early_intake_intention)
↠
output(usage_support_agent)|communication_from_to(put_medicine_back_and_wait_for_signal, usage_support_agent, patient_cell_phone)

USA5: SMS to patient when medicine not taken on time

If the agent has the belief that the level of medicine M is C at the current time point, and the level is considered to be too low, and the last message has been communicated before the last usage, and at the current time point no more medicine will be absorbed by the patient due to previous intake, then an SMS is sent to the patient cell phone to take the medicine M.

internal(usage_support_agent)|belief(current_time(T3)) ∧
internal(usage_support_agent)|belief(at(medicine_level(M, C), T3)) ∧
min_medicine_level(minB) ∧ C < minB ∧
internal(usage_support_agent)|belief(last_recorded_usage(M, T)) ∧
internal(usage_support_agent)|belief(last_recorded_patient_message_sent(M, T2)) ∧
T2 < T ∧ usage_effect_duration(UED) ∧ T3 > T + UED
↠
output(usage_support_agent)|communication_from_to(sms_take_medicine(M), usage_support_agent, patient_cell_phone)

USA6: SMS to doctor when no patient response to SMS

If the agent has the belief that the last SMS to the patient has been communicated at time T, and the last SMS to the doctor has been communicated before this time point, and furthermore, the last recorded usage is before the time point at which the SMS has been sent to the patient, and finally, the current time is later than time T plus a certain delay parameter for informing the doctor, then an SMS is communicated to the cell phone of the doctor that the patient has not taken medicine M.

internal(usage_support_agent)|belief(last_recorded_patient_message_sent(M, T)) ∧
internal(usage_support_agent)|belief(last_recorded_doctor_message_sent(M, T0)) ∧
internal(usage_support_agent)|belief(last_recorded_usage(M, T2)) ∧
internal(usage_support_agent)|belief(current_time(T3)) ∧
T0 < T ∧ T2 < T ∧ max_delay_after_warning(DAW) ∧ T3 > T + DAW
↠
output(usage_support_agent)|communication_from_to(sms_not_taken_medicine(M), usage_support_agent, doctor_cell_phone)

Bibliography

(2000). Agent working group, 2000. agent technology green paper. object management group, `ftp://ftp.omg.org/pub/docs/ec/00-03-01.pdf`.

(2002). Foundation for intelligent physical agents, .

(2008). Agent academy, 2008, `https://sourceforge.net/projects/agentacademy`.

(2008). Jade, `http://jade.tilab.com`.

(2008). Weka, `http://www.cs.waikato.ac.nz/~ml/index.html`.

Aamodt, A. and Plaza, E. (1994). Case-Based Reasoning: Foundational Issues, Methodological Variations, and System, *AI Communications* **7**, 1, pp. 36–59.

Aarts, E. (2004). Ambient intelligence: A multimedia perspective, *IEEE Multimedia* **11**, pp. 12–19.

Aarts, E., Collier, R., van Loenen, E. and de Ruyter, B. (2003). *Ambient Intelligence, Proceedings of the First European Symposium*, *Lecture Notes in Computer Science*, Vol. 2875 (Springer Verlag).

Aarts, E., Harwig, R. and Schuurmans, M. (2001). Ambient intelligence, in P. Denning (ed.), *The Invisible Future* (McGraw Hill, New York), pp. 235–250.

abd K. Shinozawa, J. Y., Brooks, R. and Naya, F. (2003). Human-robot dynamic social interaction, *NTT Technical Review* **1**, 6, pp. 37–43.

Abowd, G. D., Atkeson, C. G., Bobick, A. F., Essa, I. A., MacIntyre, B., Mynatt, E. D. and Starner, T. E. (2000). Living laboratories: the future computing environments group at the georgia institute of technology, in *Proceedings of Conference on Human Factors in Computing Systems (CHI '00): extended abstracts on Human factors in computing systems* (ACM Press, New York, NY, USA), ISBN 1-58113-248-4, pp. 215–216, doi:http://doi.acm.org/10.1145/633292.633416, URL `http://portal.acm.org/citation.cfm?id=633416&dl=ACM&coll=portal#`.

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, pp. 421 – 433, doi:http://dx.doi.org/10.1023/A:1019194325861, URL `http://portal.acm.org/ft_gateway.cfm?id=272199&type=pdf&coll=GUIDE&dl=GUIDE&CFID=31549958&CFTOKEN=48300894`, special issue: mobile computing and networking: selected papers from MobiCom '96.

Akman, V. and Surav, M. (1996). Steps toward formalizing context, *AI Magazine* **17**, 3, pp. 55–72, URL `www.aaai.org/Library/Magazine/Vol17/aimag17-03-003.php`.

Allsopp, J. (2007). *Microformats: Empowering Your Markup for Web 2.0* (Friends of ED).

Amgoud, L. and Parsons, S. (2002). Agent dialogues with conflicting preferences, in *ATAL '01: Revised Papers from the 8th International Workshop on Intelligent Agents VIII* (Springer-Verlag, London, UK), pp. 190–205.

Anabuki, M., Kakuta, H., Yamamoto, H. and Tamura, H. (2000). Welbo: An embodied conversational agent living in mixed reality space, in *Proceeding of the Conference on Human Factors in Computing Systems - CHI 00* (The Hague, The Netherlands), pp. 10–11.

Anagnostopoulos, C. B., Tsounis, A. and Hadjiefthymiades, S. (2007). Context awareness in mobile computing environments, *Wireless Personal Communications: An International Journal* **42**, 3, pp. 445–464, doi:http://dx.doi.org/10.1007/s11277-006-9187-6.

Arranz, A., Cruz, A., Sanz-Bobi, M., Ruiz, P. and Coutino, J. (2008). intelligent system for anomaly detection in a combined cycle gas turbine plant, *Expert Systems with Applications* **34**, pp. 2267–2277.

Athanasopoulou, C., Chatziathanassiou, V., Athanasopoulos, G. and Kerasidis, F. (2008). Reduction of nox emissions by regulating combustion parameters based on models extracted by applying data mining algorithms, in *6th Mediterranean Conference on Power Generation, Transmission and Distribution* (Thessaloniki, Greece).

Athanasopoulou, C., Chatziathanassiou, V. and Petridis, I. (2007). Utilizing data mining algorithms for identification and reconstruction of sensor faults: a thermal power plant case study, in *2007 IEEE Power Engineering Society* (Lausanne, Switzerland).

Baader, F., Calvanese, D., McGuinness, D. L., Nardi, D. and Patel-Schneider, P. F. (eds.) (2003). *The Description Logic Handbook: Theory, Implementation, and Applications* (Cambridge University Press), ISBN 0-521-78176-0.

Barakonyi, I., Psik, T. and Schmalstieg, D. (2004). Agents that talk and hit back: animated agents in augmented reality, in *Proceedings of the 3rd IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 2004)*, pp. 141–150.

Bartneck, C. (2002). *eMuu An Embodied Emotional Character for the Ambient Intelligent Home*, Ph.D. thesis, Eindhoven University of Technology.

Barton, J. and Vijayaraghavan, V. (2002). Ubiwise: A ubiquitous wireless infrastructure simulation environment, Technical Report HPL-2002-303, HP Labs.

Baylor, A. L., Rosenberg-Kima, R. B. and Plant, E. A. (2006). Interface agents as social models: the impact of appearance on females' attitude toward engineering, in *CHI '06: CHI '06 extended abstracts on Human factors in computing systems* (ACM, New York, NY, USA), pp. 526–531.

Bellifemine, F., Caire, G. and Greenwood, D. (2007). *Developing multi-agent systems with JADE* (John Wiley and Sons).

Bellifemine, F., Caire, G. and Rimassa, G. (2002). *JADE Programmers Guide*, TILab and University of Parma, jADE 2.6.

Bellifemine, F., Poggi, A. and Rimassa, G. (2001). Developing multi-agent systems with a fipa-compliant agent framework, *Software-Practice and Experience* **31**, pp. 103–128.

Berners-Lee, T., Hendler, J. and Lassila, O. (2001). The semantic web: Scientific american, *Scientific American* .

Besada, J., Molina, J., García, J., Berlanga, A. and Portillo, J. (2004a). Aircraft identification integrated in an airport surface surveillance video system, *Machine Vision and Applications* .

Besada, J. A., García, J. and Miguel, G. (2004b). A new approach to on-line optimal estimation of multisensor biases, *IEE Proceedings. Radar, Sonar and Navigation* **151**, 1.

Bichindaritz, I. (2006). Memory Organization as the Missing Link Between Case-Based Reasoning and Information Retrieval in Biomedicine, *Computational Intelligence* **22**, 3-4, pp. 148–160.

Biegel, G. and Cahill, V. (2004). A framework for developing mobile, context-aware applications, in *PERCOM '04: Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications (PerCom'04)* (IEEE Computer Society, Washington, DC, USA), ISBN 0-7695-2090-1, p. 361.

Billard, A. and Dautenhahn, K. (1999). Experiments in learning by imitation: Grounding and use of communication in robotic agents, *Adaptive Behaviour* **7**, 3, pp. 411–434.

Billinghurst, M., Kato, H. and Poupyrev, I. (2001). The magicbook: a transitional ar interface, *Computers and Graphics* , pp. 745–753.

Biocca, F. (1997). The cyborg's dilemma: Embodiment in virtual environments, in *CT '97: Proceedings of the 2nd International Conference on Cognitive Technology (CT '97)* (IEEE Computer Society, Washington, DC, USA), p. 12.

Biocca, F. and Nowak, K. (1999a). Communication and progressive embodiment in virtual environments, Paper presented at the International Communication Association, San Francisco.

Biocca, F. and Nowak, K. (1999b). "I feel as if I'm here, inside the computer":Toward a theory of presence in ad- vanced virtual environments, Paper presented at the International Communication Association, San Francisco.

Blum, A. L. and Furst, M. L. (1997). Fast planning through planning graph analysis, *Artificial Intelligence* **90**, 1-2, pp. 281–300, doi:http://dx.doi.org/10.1016/S0004-3702(96)00047-1.

Booth, D. and Liu, C. K. (2007). Web Services Description Language (WSDL) version 2.0 part 0: Primer, `http://www.w3.org/TR/wsdl20-primer/`.

Bosse, T., Delfos, M., Jonker, C. and Treur, J. (2006a). Modelling adaptive dynamical systems to analyze eating regulation disorders, *Simulation Journal: Transactions of the Society for Modeling and Simulation International* **82**, 3, pp. 159–171.

Bosse, T., Gerritsen, C. and Treur, J. (2007). Integration of biological, psychological and social aspects in agent-based simulation of a violent psychopath, in Y. Shi, G. van Albada, J. Dongarra and P. Sloot (eds.), *Proceedings of the Seventh International Conference on Computational Science, ICCS'07* (Springer Verlag), pp. 888–895.

Bosse, T., Hoogendoorn, M., Klein, M. and Treur, J. (2008a). A component-based agent model for assessment of driving behaviour, in F. Sandnes, M. Burgess and C. Rong (eds.), *Proceedings of the Fifth International Conference on Ubiquitous Intelligence and Computing, UIC'08* (Springer Verlag), pp. 229–243.

Bosse, T., Jonker, C., van der Meij, L., Sharpanskykh, A. and Treur, J. (2008b). Specification and verification of dynamics in agent models, *International Journal of Cooperative Information Systems* .

Bosse, T., Schut, M., Treur, J. and Wendt, D. (2008c). Trust-based inter-temporal decision making: Emergence of altruism in a simulated society, in L. Antunes, M. Paolucci and E. Norling (eds.), *Proceedings of the Eighth International Workshop on Multi-Agent-Based Simulation, MABS'07* (Springer Verlag), pp. 96–111.

Bosse, T., van Maanen, P. and Treur, J. (2006b). A cognitive model for visual attention and its application, in T. Nishida, M. Klusch, K. Sycara, M. Yokoo, J. Liu, B. Wah, W. Cheung and Y. Cheung (eds.), *Proceedings of the Sixth International Conference on Intelligent Agent Technology, IAT'06* (IEEE Computer Society Press), pp. 255–262.

Bratman, M. (1987). *Intentions, Plans and Practical Reasoning* (Harvard University Press, Cambridge, Massachusetts).

Braubach, L., Pokahr, A. and Lamersdorf, W. (2004). Jadex: A short overview, in *Main Conference Net.ObjectDays 2004*, pp. 195–207.

Brazier, F., Jonker, C. and Treur, J. (2000). Compositional design and reuse of a generic agent model, *Applied Artificial Intelligence Journal* **14**, pp. 491–538.

Brazier, F., Jonker, C. and Treur, J. (2002). Principles of component-based design of intelligent agents, *Data and Knowledge Engineering* **41**, pp. 1–28.

Broch, J., Maltz, D., Johnson, D., Hu, Y. and Jetcheva, J. (1998). A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols, *Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking* , pp. 85–97.

Busetta, P., Bouquet, P., Adami, G., Bonifacio, M., Palmieri, F., Moro, G., Sartori, C. and Singh, M. P. (2004). *K-Trek: a peer-to-peer approach to distribute knowledge in large environments*, Agents and Peer-to-Peer Computing. Second International Workshop, AP2PC 2003. Revised and Invited Papers. (Lecture Notes in Artificial Intelligence Vol.2872) (Springer-Verlag, Berlin, Germany).

Capra, L., Emmerich, W. and Mascolo, C. (2003). CARISMA: Context-aware reflective mIddleware system for mobile applications, *IEEE Transactions on Software Engineering* **29**, 10, pp. 929 – 945, doi:http://doi.ieeecomputersociety.org/10.1109/TSE.2003.1237173.

Carbogim, D. V., Robertson, D. and Lee, J. (2000). Argument-based applications to knowledge engineering, *Knowledge Engineering Review* **15**, 2, pp. 119–149.

Chalmers, R., Scheidt, D., Neighoff, T., Witwicki, S. and Bamberger, R. (2004). Cooperating unmanned vehicles, in *AIAA 1st Intelligent Systems Technical Conference*.

Chen, H. (2004). *An Intelligent Broker Architecture for Pervasive Context-Aware Systems*, Ph.D. thesis, University of Maryland, Baltimore County.

Chen, H., Finin, T. and Joshi, A. (2003). An ontology for context-aware pervasive computing environments, *The Knowledge Engineering Review* **18**, 3, pp. 197–207, doi:http://dx.doi.org/DOI:10.1017/S0269888904000025.

Chen, H., Finin, T. and Joshi, A. (2004a). A context broker for building smart meeting rooms, in C. Schlenoff and M. Uschold (eds.), *Proceedings of the Knowledge Representation and Ontology for Autonomous Systems Symposium, 2004 AAAI Spring Symposium*, AAAI (AAAI Press, Menlo Park, CA, Stanford, California), pp. 53–60.

Chen, H., Finin, T. and Joshi, A. (2005). *Ontologies for Agents: Theory and Experiences*, chap. The SOUPA Ontology for Pervasive Computing, Whitestein Series in Software Agent Technologies (Birkhäuser Basel), pp. 233–258, doi:http://dx.doi.org/10.1007/3-7643-7361-X_10, URL `http://www.springerlink.com/content/k127108k44351226/`.

Chen, H., Finin, T., Joshi, A., Kagal, L., Perich, F. and Chakraborty, D. (2004b). Intelligent agents meet the semantic web in smart spaces, *IEEE Internet Computing* **8**, 6, pp. 69–79, doi:http://dx.doi.org/10.1109/MIC.2004.66.

Chen, H., Perich, F., Chakraborty, D., Finin, T. and Joshi, A. (2004c). Intelligent agents meet semantic web in a smart meeting room, in *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS '04)* (IEEE Computer Society, Los Alamitos, CA, USA), ISBN 1-58113-864-4, pp. 854–861, doi:http://dx.doi.org/10.1109/AAMAS.2004.152.

Chen, H., Perich, F., Finin, T. and Joshi, A. (2004d). SOUPA: Standard ontology for ubiquitous and pervasive applications, in *International Conference on Mobile and Ubiquitous Systems: Networking and Services*, pp. 258–267.

Chiang, F., Braun, R. M., Magrath, S., Markovits, S. and Huang, S. (2005). Autonomic service configuration in telecommunication mass with extended role-based gaia and jadex. ***PONER!!!!*** , pp. 1319–1324.

Chittaro, L., Ieronutti, L. and Rigutti, S. (2005). Supporting presentation techniques based on virtual humans in educational virtual worlds, in *CW '05: Proceedings of the 2005 International Conference on Cyberworlds* (IEEE Computer Society, Washington, DC, USA), pp. 245–252.

Clausen, T. and Jacquet, P. (2003). RFC 3626: Optimized Link State Routing Protocol (OLSR), `http://www.ietf.org/rfc/rfc3626.txt`.

Clement, L., Hately, A., von Riegen, C. and Rogers, T. (2004). UDDI version 3.0.2, `http://uddi.org/pubs/uddi-v3.0.2-20041019.htm`.

Clements, P. C. (2001). From subroutines to subsystems: component-based software development, , pp. 189–198.

Clip, D. (2003). Gnutella Protocol Specification v0. 4, *http//www. clip2. com/GnutellaProtocol04. pdf* .

Coen, M. H. (1997). Building brains for rooms: designing distributed software agents, in *Proceedings of the Conference on Innovative Applications of Artificial Intelligence (IAAI'97)* (AAAI Press), pp. 971–977.

Coen, M. H. (1998). Design principles for intelligent environments, in *Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelli-*

gence (AAAI '98/IAAI '98) (American Association for Artificial Intelligence, Menlo Park, CA, USA), ISBN 0-262-51098-7, pp. 547–554.

Collett, T. and MacDonald, B. (2006). Augmented reality visualisation for player, in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA 2006)*, pp. 3954–3959.

Collier, R. W. (2001). *Agent Factory: A Framework for the Engineering of Agent-Oriented Applications*, Ph.D. thesis, University College Dublin, Dublin, Ireland.

Collins, N. and Baird, C. (1989). Terrain aided passive estimation, in *Proceedings of the IEEE National Aerospace and Electronics Conference*, Vol. 3, pp. 909–916.

Collins, R. T., Lipton, A. J., Fujiyoshi, H. and Kanade, T. (2001). Algorithms for cooperative multisensor surveillance, in *Proceedings of the IEEE*, Vol. 89 (IEEE).

Daily, M., Cho, Y., Martin, K. and Payton, D. (2003). World embedded interfaces for human-robot interaction, *Proceedings of the 36th Annual Hawaii International Conference on System Sciences, 2003* .

DARPA (2008). DARPA Strategic Technology Office, `http://www.darpa.mil/STO/index.html`.

Dasarathy, B. (1991). *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques* (Los Alamitos, CA: IEEE Computer Society Press).

Dautenhahn, K. (1997). I could be you: The phenomenological dimension of social understanding, *Cybernetics and Systems* **28**, pp. 417–453(37).

Dautenhahn, K. (1998). The art of designing socially intelligent agents: Science, fiction, and the human in the loop, *Applied Artificial Intelligence* **12**, 8–9, pp. 573–617.

Dennett, D. (1987a). *The Intentional Stance* (MIT Press, Cambridge Mass.).

Dennett, D. (1987b). *The Intentional Stance* (Bradford Books).

Dey, A. K. (2000). *Providing architectural support for building context-aware applications*, Ph.D. thesis, Georgia Institute of Technology, director-Gregory D. Abowd.

Domingos, P. (1999). The role of occam's razor in knowledge discovery, *Data Mining and Knowledge Discovery* **3**, 4, pp. 409–425.

Doswell, J. T. (2005). It's virtually pedagogical: Pedagogical agents in mixed reality learning environments, in *Proceedings of the Thirtysecond International Conference on Computer Graphics and Interactive Techniques - SIGGRAPH 2005 - Educators Program* (Los Angeles, California), p. 25.

Doyle, J. (1979). A truth maintenance system, *Artificial Intelligence* **12**, pp. 231–272.

Dragone, M. (2007). *SoSAA: An agent-based robot software framework*, Ph.D. thesis, School of Computer Science & Informatics, University College Dublin, Dublin, Ireland, `http://csserver.ucd.ie/~mdragone/pubs/MauroDragonePhdThesis.pdf`.

Dragone, M., Holz, T. and OHare, G. M. P. (2006). Mixing robotic realities, in *Proceedings of the 11th international conference on Intelligent user interfaces (IUI 2006)* (ACM Press, New York, NY, USA), pp. 261–263.

Dragone, M., Holz, T. and OHare, G. M. P. (2007). Using mixed reality agents as social interfaces for robots, in *RO-MAN '07: Proceedings of the 16th IEEE International Workshop on Robot and Human Interactive Communication* (IEEE Press, Jeju Island, Korea).

Duffy, B. R., O'Hare, G. M. P., Martin, A. N., Bradley, J. F. and Schön, B. (2003). Agent chameleons: agent minds and bodies, in *Proceedings of the 16th International Conference on Computer Animation and Social Agents (CASA 2003)*, pp. 118–125.

Dung, P. M. (1995). On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games, *Artificial Intelligence* **77**, 2, pp. 321–358.

Feiner, S., MacIntyre, B., Hollerer, T. and Webster, A. (1997). A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment, in *Proceedings of the 1st IEEE International Symposium on Wearable Computers (ISWC '97)* (IEEE Computer

Society, Washington, DC, USA), p. 74.

Fitzpatrick, A., Biegel, G., Clarke, S. and Cahill, V. (????). Towards a Sentient Object Model, *Workshop on Engineering Context-Aware Object Oriented Systems and Environments (ECOOSE)* .

Flynn, D. (2003). *Thermal Power Plant Simulation and Control* (IEEE Press).

Fogg, B. J. (1999). Persuasive technologies, *Communications of the ACM* **42**, 5, pp. 26–29.

Fong, T., Nourbakhsh, I. and Dautenhahn, K. (2003). A survey of socially interactive robots, *Robotics and Autonomous Systems* **42**, 3–4, pp. 143–166.

Forgy, C. L. (1982). Rete: A fast algorithm for the many pattern/many object pattern match problem, *Artificial Intelligence* **19**, 1, pp. 17 – 37, doi:doi:10.1016/0004-3702(82)90020-0, URL `http://www.sciencedirect.com/science?_ob=ArticleURL&_udi=B6TYF-47X2B1P-4P&_coverDate=09%2F30%2F1982&_alid=484424847&_rdoc=1&_fmt=&_orig=search&_qd=1&_cdi=5617&_sort=d&view=c&_acct=C000051951&_version=1&_urlVersion=0&_userid=1196840&md5=7679fbebbf82cef14bb92cb5c031f2b5`.

Frawley, W., Piatetsky-Shapiro, G. and Matheus, C. (1992). Knowledge discovery in databases: An overview, *AI Magazine* **13**, 3, pp. 57–70.

Friedman, B. (1995). “it’s the computer’s fault”: reasoning about computers as moral agents, in *CHI ’95: Conference companion on Human factors in computing systems* (ACM, New York, NY, USA), pp. 226–227.

Friedman Hill, E. (2003). *Jess in Action: Java Rule-Based Systems* (Manning Publications Co., Greenwich, CT, USA), ISBN 1930110898.

Fuentes, V., Carbó, J. and Molina, J. M. (2006a). Heterogeneous domain ontology for location based information system in a multi-agent framework, in *IDEAL*, pp. 1199–1206.

Fuentes, V., Pi, N. S., Carbó, J. and Molina, J. M. (2006b). Reputation in user profiling for a context-aware multiagent system, in *EUMAS*.

Fujita, Y. (2002). Personal robot PaPeRo, *Journal of Robotics and Mechatronics* **14**, 1, pp. 60–63.

Gaerdenfors, P. (2003). *How Homo Became Sapiens: On The Evolution Of Thinking* (Oxford University Press).

Gandon, F. and Sadeh, N. M. (2003). A semantic e-wallet to reconcile privacy and context awareness, *Lecture Notes in Computer Science: The SemanticWeb - ISWC 2003* **2870**, pp. 385–401, doi:http://dx.doi.org/10.1007/b14287, URL `http://springerlink.metapress.com/content/gc61ckcb8kv4um2l/?p=0c64d88c87ae4dec8e8d70b48c5ee442&pi=3`.

Gandon, F. L. and Sadeh, N. M. (2004). Semantic web technologies to reconcile privacy and context awareness, *Journal of Web Semantics* **1**, 3, pp. 241–260, doi:http://dx.doi.org/10.1016/j.websem.2003.07.008.

García, J., Besada, J. A. and Casar, J. R. (2003). Use of map information for tracking targets on airport surface, *IEEE Transactions on Aerospace and Electronic Systems* **39**, 2, pp. 675–694.

García, J., Besada, J. A., Jiménez, F. J. and Casar, J. R. (2000). A block processing method for on-line multisensor registration with its application to airport surface data fusion, in *Proceedings of the IEEE International Radar Conference* (Washington-DC, EE. UU.).

Garcia-Clemente, F., Martinez, G., Botia, J. A. and Gomez-Skarmeta, A. (2005). On the application of the semantic web rule language in the definition of policies for system security management, in *Workshop AWeSOMe’05*.

Gerkey, B. P., Vaughan, R. T. and Howard, A. (2003). The Player/Stage project: Tools for multi-robot and distributed sensor systems, in *Proceedings of the International Conference on Advanced Robotics (ICAR ’03)* (Coimbra, Portugal), pp. 317–323.

Giesler, B., Salb, T., Steinhaus, P. and Dillmann, R. (2004). Using augmented reality to interact with an autonomous mobile platform, in *Proceedings of the IEEE International Conference on*

Robotics and Automation - ICRA 2004 (New Orleans, Louisiana, USA).

Giunchiglia, F., Mylopoulos, J. and Perini, A. (2002). The tropos software development methodology: Processes, models and diagrams, in *Proceedings of the first international joint conference on autonomous agents and multiagent systems* (ACM press), pp. 63–74.

Goldman, A. (2006). *Simulating Minds: The Philosophy, Psychology and Neuroscience of Mind Reading* (Oxford University Press).

González, C. (2008). *Contribuciones al diseño de sistema tutores inteligentes usando razonamiento basado en casos*, Ph.D. thesis, University of Vigo, Spain.

Gram, C. and Cockton, G. (1996). *Design Principles for Interactive Software* (Chapman & Hall, London).

Green, D. (2005). Realtime compliance management using a wireless realtime pillbottle - a report on the pilot study of simpill, in *Proceedings of the International Conference for eHealth, Telemedicine and Health, Med-e-Tel'05*.

Gross, J. (2007). *Handbook of Emotion Regulation* (Guilford Press, New York).

Gruber, T. R. (1993). A translation approach to portable ontology specifications, *Knowledge Acquisition* **5**, 2, pp. 199–220, doi:http://dx.doi.org/10.1006/knac.1993.1008.

Gu, T., Wang, X. H., Pung, H. K. and Zhang, D. Q. (2004). An ontology-based context model in intelligent environments, in *Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference*, pp. 270–275.

Gutierrez, M., Vexo, F. and Thalmann, D. (2003). Controlling virtual humans using PDAs, in *Proceedings of the 9th International Conference on Multimedia Modelling (MMM'03), Taiwan*.

Ha, Y.-G., Sohn, J.-C., Cho, Y.-J. and Yoon, H. (2005). Towards a ubiquitous robotic companion: Design and implementation of ubiquitous robotic service framework, *ETRI Journal* **27**, 6, pp. 666–676.

Hall, D. L. and Llinas, J. (2001). *Handbook of MultiSensor Data Fusion* (CRC Press, Boca Raton).

Hanssens, N., Kulkarni, A., Tuchida, R. and Horton, T. (2002). Building agent-based intelligent workspaces, in *Proceedings of the International Conference on Internet Computing (IC'2002)* (CSREA Press, Las Vegas, Nevada, USA), pp. 675 – 681.

Harter, A., Hopper, A., Steggles, P., Ward, A. and Webster, P. (2002). The anatomy of a context-aware application, *Wireless Networks* **8**, 2 – 3, pp. 187 – 197.

Hemming, N. (2001). KaZaA, *Web Site-www. kazaa. com* .

Herianto, Sakakibara, T. and Kurabayashi, D. (2007). Artificial pheromone system using RFID for navigation of autonomous robots, *Journal of Bionic Engineering* **4**, 4, pp. 245–253.

Hess, C. K., Román, M. and Campbell, R. H. (2002). Building applications for ubiquitous computing environments, in *Pervasive*, pp. 16–29.

Hobbs, J. R. and Pan, F. (2004). An ontology of time for the semantic web, *ACM Transactions on Asian Language Information Processing (TALIP)* **3**, 1, pp. 66–85, doi:http://doi.acm.org/10.1145/1017068.1017073.

Holz, T., Dragone, M., OHare, G. M. P., Martin, A. and Duffy, B. R. (2006). Mixed reality agents as museum guides, in *ABSHL '06: Agent-Based Systems for Human Learning, AAMAS 2006 Workshop* (ACM Press, New York, NY, USA).

Hoogendoorn, M., Klein, M. and Treur, J. (2008). Formal design and simulation of an ambient multi-agent system model for medicine usage management, in M. Muehlhaeuser, A. Ferscha and E. Aitenbichler (eds.), *Constructing Ambient Intelligence: AmI-07 Workshop Proceedings* (Springer Verlag), pp. 207–217.

Hristova, N., OHare, G. M. P. and Lowen, T. (2003). T.: Agent-based ubiquitous systems: 9 lessons learnt, in *In Workshop on System Support for Ubiquitous Computing (UbiSys03), 5th International Conference on Ubiquitous Computing (UbiComp*.

Huhns, M. and Singh, M. (2005). Service-oriented computing: Key concepts and principles, *IEEE Internet Computing* **9**, 1, pp. 75–81.

Iglesias, C., Garijo, M., Gonzalez, J. and Velasco, J. (1996). A methodological proposal for multiagent systems development extending commonkads, in *Proceedings of the Tenth Knowledge Acquisition for Knowledge-Based Systems Workshop*.

Iglesias, C., Garijo, M., Gonzalez, J. and Velasco, J. (1998). Analysis and design of multiagent systems using mas-commonkads, in *Intelligent agents IV: Agent Theories, Architectures and Languages* (Munchen, Germany), pp. 313–327.

Inc, N. (2002). The Napster Homepage, *Online: http://www. napster. com* .

Intille, S. S. (2002). Designing a home of the future, *IEEE Pervasive Computing* **1**, 2, pp. 76–82, doi:http://dx.doi.org/10.1109/MPRV.2002.1012340.

J., S. and A., H. (2002). Novel interactive control interface for centaur-like service robot, in *Proceedings of the 15th IFAC World Congress on Automatic Control* (Barcelona, Spain).

JARA (2008). Japan robot association, `http://www.jara.jp/`.

Jennings, N. and Bussmann, S. (2003). Agent-based control systems, *IEEE Control Systems Magazine* **23**, 3, pp. 61–74.

Jonker, C. and Treur, J. (2002). Compositional verification of multi-agent systems: a formal analysis of pro-activeness and reactiveness, *International Journal of Cooperative Information Systems* **11**, pp. 51–92.

Kahan, J. and Rapoport., A. (1984). *Theories of Coalition Formation* (Lawrence Erlbaum Associatates Publishers).

Kakas, A. and Moraitis, P. (2003). Argumentation based decision making for autonomous agents, in *AAMAS '03: Proceedings of the second international joint conference on Autonomous agents and multiagent systems* (ACM Press, New York, NY, USA), ISBN 1-58113-683-8, pp. 883–890.

Kaplan, F. (2005). Everyday robotics: robots as everyday objects, in *sOc-EUSAI '05: Proceedings of the 2005 joint conference on Smart objects and ambient intelligence* (ACM, New York, NY, USA), pp. 59–64.

Karniely, H. and Siegelmann, H. T. (2000). Sensor registration using neural networks, *IEEE Transaction on Aerospace and Electronic Systems* **36**, 1, pp. 85–101.

Kato, H. and Billinghurst, M. (1999). Marker tracking and HMD calibration for a video-based augmented reality conferencing system, in *Proceedings of the Second International IEEE Workshop on Augmented Reality - IWAR 99* (San Francisco, California, USA).

Ketchpel, S. (1994). Forming coalitions in the face of uncertain rewards, in *Proceedings of the National Conferece on Artificial Intelligence* (Seattle, WA), pp. 414–419.

Kidd, C. and Breazeal, C. (2005). Sociable robot systems for real-world problems, in *Proceedings of the IEEE International Workshop on Robot and Human Interactive Communication (ROMAN 2005)*, pp. 353–358.

Kim, J.-H. (2005). Ubiquitous robot, in *Computational Intelligence, Theory and Applications*, Vol. 33 (Springer Berlin / Heidelberg).

Kirubarajan, T., Bar-Shalom, Y., Pattipati, K. R. and Kadar, I. (2000). Ground target tracking with variable structure imm estimator, *IEEE Transaction on Aerospace and Electronic Systems* **36**, 1, pp. 392–400.

Klein, J., Moon, Y. and Picarda, R. W. (1999). This computer responds to user frustration, in *CHI '99: CHI '99 extended abstracts on Human factors in computing systems* (ACM, New York, NY, USA), pp. 242–243.

Kopanas, I., Avouris, N. and Daskalaki, S. (2002). The role of domain knowledge in a large scale data mining project, *Lecture Notes in Artificial Intelligence* **2308**, pp. 288–299.

Kraus, S., Nirkhe, M. and Sycara, K. P. (1993). Reaching agreements through argumentation: a logical model (preliminary report), in *Proceedings of the 12th International Workshop on Distributed Artificial Intelligence* (Hidden Valley, Pennsylvania), pp. 233–247.

LaMarca, A., Brunette, W., Koizumi, D., Lease, M., Sigurdsson, S. B., Sikorski, K., Fox, D. and

Borriello, G. (2002). Plantcare: An investigation in practical ubiquitous systems, in *UbiComp '02: Proceedings of the 4th international conference on Ubiquitous Computing* (Springer-Verlag, London, UK), pp. 316–332.

Lech, T. C. and Wienhofen, L. W. M. (2005). Ambieagents: a scalable infrastructure for mobile and context-aware information services, in *AAMAS '05: Proceedings of the fourth international joint conference on Autonomous agents and multiagent systems* (ACM, New York, NY, USA), ISBN 1-59593-093-0, pp. 625–631, doi:http://doi.acm.org/10.1145/1082473.1082568.

Lee, W., Lee, J. and Woo, W. (2005). TARBoard: Tangible augmented reality system for table-top game environment, in *Proceedings of PerGames 2005, 2nd International Workshop on Pervasive Gaming Applications*.

Lesser, V. (1999). Cooperative multiagent systems: A personal view of the state of the art, *IEEE Transactions on Knowledge and Data Engineering* **11**, 1.

Lin, C.-y. and Hsu, J. Y.-j. (2006). IPARS: Intelligent portable activity recognition system via everyday objects, human movements, and activity duration, in *Modeling Others from Observations (MOO 2006): Papers from the 2006 AAAI Workshop* (AAAI Press, Boston, Massachusetts, USA), pp. 44–52, technical Report WS-06-13.

Long, S., Aust, D., Abowd, G. and Atkeson, C. (1996). Cyberguide: prototyping context-aware mobile applications, in *Conference companion on Human factors in computing systems (CHI '96)* (ACM Press, Vancouver, British Columbia, Canada), ISBN 0-89791-832-0, pp. 293 – 294, doi:http://doi.acm.org/10.1145/257089.257328.

Look, G. and Shrobe, H. (2004). A plan-based mission control center for autonomous vehicles, in *IUI '04: Proceedings of the 9th international conference on Intelligent user interfaces* (ACM Press, New York, NY, USA), ISBN 1-58113-815-6, pp. 277–279, doi:http://doi.acm.org/10.1145/964442.964504.

Lupu, E. and Sloman, M. (1999). Conflicts in policy-based distributed systems management, *IEEE Trans. Software Eng.* **25**, 6, pp. 852–869.

Ma, Z., Iman, F., Lu, P., Sears, R., Kong, L., Rokanuzzaman, A., McCollor, D. and Benson, S. (2007). A comprehensive slagging and fouling prediction tool for coal-fired boilers and its validation/application, *Fuel Process. Technol* **88**, pp. 1035–1043.

Malheiro, B. and Oliveira, E. (2000). Solving conflicting beliefs with a distributed belief revision approach, in *IBERAMIA-SBIA '00: Proceedings of the International Joint Conference, 7th Ibero-American Conference on AI* (Springer-Verlag, London, UK), ISBN 3-540-41276-X, pp. 146–155.

Mamei, M., Zambonelli, F., Moro, G., Sartori, C. and Singh, M. P. (2004). *Location-based and content-based information access in mobile peer-to-peer computing: the TOTA approach*, Agents and Peer-to-Peer Computing. Second International Workshop, AP2PC 2003. Revised and Invited Papers. (Lecture Notes in Artificial Intelligence Vol.2872) (Springer-Verlag, Berlin, Germany).

Mangina, E. (2003). Application of intelligent agents in power industry: promises and complex issues, *Lecture Notes in Artificial Intelligence* **2691**, pp. 564–573.

Manyika, J. and Durrant-Whyte, H. (1994). *Data Fusion and Sensor Management a decentralized information-theoretic approach* (Ellis Horwood).

Martin, A., O'Hare, G. M. P., Duffy, B. R., Schön, B. and Bradley, J. F. (2005). Maintaining the identity of dynamically embodied agents, in *Proceedings of the 5th International Working Conference on Intelligent Virtual Agents (IVA 2005)* (Springer-Verlag, London, UK), pp. 454–465.

Martinez, G., Garcia, F. and Gomez-Skarmeta, A. (2006). *Web and Information Security*, chap. Policy based Management of Web Information Systems Security: an Emerging Technology (Idea Group).

McCarthy, J. (1993). Notes on formalizing context, in *Proceedings of the Thirteenth Interna-*

tional Joint Conference on Artificial Intelligence (IJCAI'93), Vol. 1 (Morgan Kaufmann), pp. 555 – 560, URL `http://dli.iiit.ac.in/ijcai/IJCAI-93-VOL1/CONTENT/content.htm`.

McCarthy, J. and Hayes, P. J. (1969). Some philosophical problems from the standpoint of artificial intelligence, *Machine Intelligence* **4**, pp. 463–502.

Micheloni, C., Foresti, G. L. and Snidaro, L. (2003). A co-operative multicamera system for video-surveillance of parking lots, in *Intelligent Surveillance Systems Symp. by the IEE* (London), pp. 21–24.

Milgram, P., Rastogi, A. and Grodski, J. J. (1995). Telerobotic control using augmented reality, in *Proceedings of the 4th IEEE International Workshop on Robot and Human Communication (RO-MAN 1995)*, pp. 21–29.

Mitra, N. and Lafon, Y. (2007). Soap version 1.2 part 0: Primer (second edition), `http://www.w3.org/TR/soap12-part0/`.

Molina, J., Garcia, J., Jimenez, F. and Casar, J. (2002). Surveillance multisensor management with fuzzy evaluation of sensor task priorities, *Engineering Applications of Articial Intelligence* .

Molina, J., García, J., Jiménez, F. and Casar, J. (2003). Cooperative management of a net of intelligent surveillance agent sensors, *International Journal of Intelligent Systems* **3**, 18, pp. 279–307.

Moro, G., Sartori, C. and Singh, M. P. (2004). *Agents and Peer-to-Peer Computing. Second International Workshop, AP2PC 2003. Revised and Invited Papers. (Lecture Notes in Artificial Intelligence Vol.2872)* (Springer-Verlag, Berlin, Germany).

Nabaa, N. and Bishop, R. H. (1999). Solution to a multisensor tracking problem with sensor registration errors, *IEEE Transactions on Aerospace and Electronic Systems* **35**, 1, pp. 354–363.

Nguyen, N. T., Venkatesh, S., West, G. and Bui, H. H. (2003). Multiple camera coordination in a surveillance system, *Acta Automatica Sinica* **29**, 3, pp. 408–421.

Nieto-Carvajal, I., Botía, J. A., Ruiz, P. M. and Gómez-Skarmeta, A. F. (2004). Implementation and evaluation of a location-aware wireless multi-agent system, in *EUC*, pp. 528–537.

Nwana, H. S., Ndumu, D. T., Lee, L. C., Collis, J. C. and Re, I. I. (1999). Zeus: A tool-kit for building distributed multi-agent systems, *Applied Artifical Intelligence Journal* **13**, pp. 129–186.

O'Hare, G. M. P., Collier, R., Conlon, J. and Abbas, S. (1998). Agent factory: An environment for constructing and visualising agent communities, in *Proceedings of the Ninth Irish Conference on Artificial Intelligence and Cognitive Science - AICS 98* (Dublin, Ireland), pp. 249–261.

O'Hare, G. M. P., Duffy, B. R. and Campbell, A. G. (2004). NeXuS: Mixed reality experiments with embodied intentional agents, in *Proceedings of the Seventeenth International Conference on Computer Animation and Social Agents - CASA 2004* (Geneva, Switzerland).

O'Neill, E., Klepal, M., Lewis, D., O'Donnell, T., O'Sullivan, D. and Pesch, D. (2005). A testbed for evaluating human interaction with ubiquitous computing environments, in *Proceedings of the First International Conference on Testbeds and Research Infrastructures for the DEvelopment of NeTworks and COMmunities (TRIDENTCOM'05)* (IEEE Computer Society, Washington, DC, USA), pp. 60–69.

Padgham, L. and Winikoff, M. (2004). *Developing intelligent agent systems: a practical guide* (John Wiley and Sons).

Parsons, S., Sierra, C. and Jennings, N. R. (1998). Agents that reason and negotiate by arguing, *Journal of Logic and Computation* **8**, 3, pp. 261–292.

Parsons, S. D. and Jennings, N. R. (1996). Negotiation through argumentation-A preliminary report, in *Proceedings of the Second International Conference Multi-Agent Systems (ICMAS'96)* (Kyoto, Japan), pp. 267–274.

Patricio, M., Carbo, J., Perez, O., Garcia, J. and Molina, J. (2007). Multi-agent framework in visual sensor networks, *EURASIP Journal on Advances in Signal Processing* , pp. Article ID 98639, 21 pagesDoi:10.1155/2007/98639.

Paulidis, I. and Morellas, V. (????). *Two examples of indoor and outdoor surveillance systems*

(Kluwer Academic Publishers, Boston 2002).

Perich, F., Joshi, A., Finin, T., and Yesha, Y. (2004). On data management in pervasive computing environments, *IEEE Transactions on Knowledge and Data Engineering* .

Perkins, C., Belding-Royer, E. and Das, S. (2003). RFC 3561: Ad hoc on-demand distance vector (AODV) routing, `http://www.ietf.org/rfc/rfc3561.txt`.

Plaza, E., Arcos, J. and Martín, F. (1996). Cooperation Modes among Case-Based Reasoning Agents, *Proc. ECAI* **96**.

Pokahr, A., Braubach, L. and Lamersdorf, W. (2003). Jadex: Implementing a bdi-infrastructure for jade agents, *EXP - in search of innovation (Special Issue on JADE)* **3**, 3, pp. 76–85.

Prakken, H. (2006). Formal systems for persuasion dialogue, *Knowledge Engineering Review* **21**, 2, pp. 163–188.

Prakken, H. and Sartor, G. (1996). A dialectical model of assessing conflicting arguments in legal reasoning, *Artificial Intelligence and Law* **4**, 3-4, pp. 331–368.

Prendingera, H., Dohi, H., Wang, H., Mayer, D. and Ishizuka, M. (2004). Empathic embodied interfaces: Addressing users affective state, in *Proceedings of the Tutorial and Research Workshop on Affective Dialogue Systems* (Springer Verlag), pp. 53–64.

Priest, G. (2002). Paraconsistent Logic, in D. M. Gabbay (ed.), *Handbook of Philosophical Logic Volume 6*, 2nd edn. (Kluwer Academic Pub.).

Raiffa, H. (1984). *The Art and Science of Negotiation* (Harvard Univ. Press, Cambridge, Massachusetts).

Ramanujam, S. and Capretz, M. A. M. (2004). Design of a multi-agent system for autonomous database administration, *International Journal of Intelligent Systems* , pp. 1167–1170.

Ranganathan, A., Al-Muhtadi, J. and Campbell, R. H. (2004). Reasoning about uncertain contexts in pervasive computing environments, *IEEE Pervasive Computing* **3**, 2, pp. 62–70, doi:http://doi.ieeecomputersociety.org/10.1109/MPRV.2004.1316821.

Rao, A. and Georgeff, M. (1995a). Bdi agents: from theory to practice, in *Proceedings of the First International Conference on Multi-Agent Systems (ICMAS'95)* (The MIT Press, Cambridge, MA, USA), pp. 312–319.

Rao, A. S. and Georgeff, M. P. (1995b). Bdi agents: From theory to practice, in *ICMAS*, pp. 312–319.

Reiter, R. (2001). *Knowledge in Action: Logical Foundations for Specifying and Implementing Dynamical Systems* (MIT Press).

Reynolds, V., Cahill, V. and Senart, A. (2006). Requirements for an ubiquitous computing simulation and emulation environment, in *First International Conference on Integrated Internet Ad hoc and Sensor Networks (InterSense 2006)* (OCP Science).

Rickenberg, R. and Reeves, B. (2000). The effects of animated characters on anxiety, task performance, and evaluations of user interfaces, in *CHI '00: Proceedings of the SIGCHI conference on Human factors in computing systems* (ACM, New York, NY, USA), pp. 49–56.

Riva, G., Vatalaro, F., Davide, F. and Alcaniz, M. (2005). *Ambient Intelligence* (IOS Press).

Roman, M., Hess, C., Cerqueira, R., Ranganathan, A., Campbell, R. and Nahrstedt, K. (2002). Gaia: A Middleware Infrastructure to Enable Active Spaces, *IEEE Pervasive Computing* **1**, 4, pp. 74–83.

Román, M., Hess, C., Cerqueira, R., Ranganathan, A., Campbell, R. H. and Nahrstedt, K. (2002). Gaia: a middleware platform for active spaces, *ACM SIGMOBILE Mobile Computing and Communications Review* **6**, 4, pp. 65–67, doi:http://doi.acm.org/10.1145/643550.643558.

Rosenschein, J. S. and Zlotkin, G. (1994). *Rules of Encounter. Designing Conventions for Automated Negotiation among Computers* (MIT Press).

Royer, E. and Toh, C. (1999). A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks, *Personal Communications, IEEE [see also IEEE Wireless Communications]* **6**, 2, pp. 46–55.

Salber, D., Dey, A. K. and Abowd, G. D. (1999a). The context toolkit: aiding the development of context-enabled applications, in *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI'99)* (ACM Press, New York, NY, USA), ISBN 0-201-48559-1, pp. 434–441, doi:http://doi.acm.org/10.1145/302979.303126.

Salber, D., Dey, A. K., Orr, R. J. and Abowd, G. D. (1999b). Designing for ubiquitous computing: A case study in context sensing, Tech. Rep. GIT-GVU-99-29, Georgia Institute of Technology, URL `http://hdl.handle.net/1853/3396`.

Sas, C. and OHare, G. M. P. (2001). Defining and measuring presence in non-virtual environments: An experimental study, in *Proceedings of the 4th International Workshop on Presence* (Philadelphia, PA, USA).

Schiaffinoa, S. and Amandia (2004). A user – interface agent interaction: personalization issues, *Int. J. Human-Computer Studies* **60**, pp. 129–148.

Schilit, B., Adams, N. and Want, R. (1994). Context-aware computing applications, in *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications* (Santa Cruz, CA, US), pp. 85 – 90, doi:http://dx.doi.org/10.1109/MCSA.1994.512740, URL `http://ieeexplore.ieee.org/xpls/abs_all.jsp?isnumber=11297&arnumber=512740`.

Schmidt, A. (2005). Interactive context-aware systems - interacting with ambient intelligence, in G. Riva, F. Vatalaro, F. Davide and M. Alcaniz (eds.), *Ambient Intelligence* (IOS Press), pp. 159–178.

Schmidt, A., Beigl, M. and Gellersen, H. (1999). There is more to context than location, *Computer & Graphics Journal* **23**, 19, pp. 893–902.

Schmidt, A., Kortuem, G., Morse, D. and Dey, A. (2001). Situatied interaction and context-aware computing, *Personal and Ubiquitous Computing* **5**, 1, pp. 1–3.

Scott, C. (1994). Improved gps positioning for motor vehicles through map matching, in *Proceedings of ION GPS-94: 7th International Technical Meeting of the Satellite Division of the Institute of Navigation*, pp. 1391–1400.

Shechory, O. and S.Kraus (1995). Feasible formation of stable coalitions among autonomous agents in general environments, *Computational Intelligence Journal* .

Shohama, Y. and Tennenholtz, M. (1995). On social laws for artificial agent societies: off-line design, *Artificial Intelligence* **73**, pp. 231–252.

Shoji, M., Miura, K. and Konno, A. (2006). U-Tsu-Shi-O-Mi: The virtual humanoid you can reach, in *Proceedings of the Thirtythird International Conference on Computer Graphics and Interactive Techniques - SIGGRAPH 2006* (Boston, Massachusetts, USA).

Shreiber, G., Akkermans, H., Anjewierden, A., de Hoog, R., Shadbolt, N., de Velde, W. V. and Wielinga, B. (2000). *Knowledge Engineering and Management: the CommonKADS methodology* (MIT Press).

Siebel, N. and Maybank, S. (2004). The advisor visual surveillance system, *ECCV 2004 workshop Applications of Computer Vision (ACV)* **1**.

Simmons, R., Goldberg, D., Goode, A., Monetmerlo, M., Roy, N., Sellner, B., Urmson, C., Schultz, A., Abramson, M., Adams, W., Atrash, A., Bugajska, M., Coblenz, M., MacMahon, M., Perzanowski, D., Horswill, I., Zubek, R., Kortenkamp, D., Wolfe, B., Milman, T. and Maxwell, B. (2003). Grace: An autonomous robot for the AAAI robot challenge, *AI Magazine* **24**, pp. 51–72.

Slater, M., Pertaub, D.-P. and Steed, A. (1999). Public speaking in virtual reality: facing an audience of avatars, *Computer Graphics and Applications, IEEE* **19**, 2, pp. 6–9.

Smith, M. K., , Welty, C. and McGuinness, D. L. (2004). OWL web ontology language guide. W3C recommendation, Tech. rep., W3C.

Smithson, A., Moreau, L., Moro, G. and Koubarakis, M. (2003). *Engineering an agent-based peer-to-peer resource discovery system*, Agents and Peer-to-Peer Computing. First International Workshop, AP2PC 2002. Revised and Invited Papers. (Lecture Notes in Artificial Intelligence Vol.2530) (Springer-Verlag, Berlin, Germany).

Soldatos, J., Pandis, I., Stamatis, K., Polymenakos, L. and Crowley, J. (2007). Agent based middleware infrastructure for autonomous context-aware ubiquitous computing services, *Computer Communications* **30**, pp. 577–591.

Stilman, M., Chestnutt, J., Michel, P., Nishiwaki, K., Kagami, S. and Kuffner, J. (2005). Augmented reality for robot development and experimentation, Tech. Rep. CMU-RI-TR-05-55, Robotics Institute, Carnegie Mellon University.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions, *Journal of the Royal Statistical Society* **36**, 2, pp. 577–591.

Strang, T. and Linnhoff-popien, C. (2004). A context modeling survey, in *Workshop on Advanced Context Modelling, Reasoning and Management at The Sixth International Conference on Ubiquitous Computing (UbiComp 2004)* (Nottingham, England).

Strang, T., Linnhoff-popien, C. and Frank, K. (2003). CoOL: A context ontology language to enable contextual interoperability, in *Proceedings of 4th IFIP WG 6.1 International Conference on Distributed Applications and Interoperable Systems (DAIS2003)* (Springer Verlag), pp. 236–247.

Sutherland, I. E. (1998). A head-mounted three dimensional display, , pp. 295–302.

Sycara, K. (1990). Persuasive argumentation in negotiation, *Theory and Decision* **28**, 3, pp. 203–242.

Sycara, K., Paolucci, M., Velsen, M. V. and Giampapa, J. (2003). The retsina mas infrastructure, *Autonomous Agents and Multi-Agent Systems* **7**, 1-2, pp. 29–48.

Syukur, E., Loke, S. and Stanski, P. (2005). Methods for policy conflict detection and resolution in pervasive computing environments, in *Policy Management for Web workshop in conjunction with WWW2005 Conference* (Chiba, Japan).

Tamura, H., Yamamoto, H. and Katayama, A. (2001). Mixed reality: future dreams seen at the border between real and virtual worlds, *Computer Graphics and Applications, IEEE* **21**, 6, pp. 64–70.

Thanh, D. V. and Jorstad, I. (2005). A service-oriented architecture framework for mobile services, in *Telecommunications, 2005. Advanced Industrial Conference on Telecommunications/Service Assurance with Partial and Intermittent Resources Conference/ E-Learning on Telecommunications Workshop. AICT/SAPIR/ELETE 2005. Proceedings*, pp. 65–70.

Thomas, B., Close, B., Donoghue, J., Squires, J., De Bondi, P., Morris, M. and Piekarski, W. (2000). Arquake: an outdoor/indoor augmented reality first person application, *Wearable Computers, 2000. The Fourth International Symposium on* , pp. 139–146.

Tiba, F. and Capretz, M. (2006). An overview of the analysis and design of sigma: Supervisory intelligent multi-agent system architecture, *Information and Communication Technologies, 2006. ICTTA '06. 2nd* **2**, pp. 3052–3057.

Tveit, A. (2001). A survey of agent-oriented software engineering, Proc. of the First NTNU CSGS Conference (http://www.amundt.org), URL `http://www.abiody.com/jfipa/publications/AgentOrientedSoftwareEngineering/`.

Valera, M. and Velastin, S. (2005). Intelligent distributed surveillance systems: a review. , pp. 152:192–204.

van Breemen, A., Yan, X. and Meerbeek, B. (2005). iCat: an animated user-interface robot with personality, in *Proceedings of the 4th international joint conference on Autonomous agents and multiagent systems (AAMAS 2005)* (ACM, New York, NY, USA), pp. 143–144.

View, T. (2002). Reconfigurable context-sensitive middleware for pervasive computing, *Pervasive Computing, IEEE* **1**, 3, pp. 33–40.

Wagner, D., Billinghurst, M. and Schmalstieg, D. (2006). How real should virtual characters be? in *ACE '06: Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology* (ACM, New York, NY, USA), p. 57.

Waltz, E. and Llinas, J. (1990). *Multisensor Data Fusion* (Artech House Inc, Norwood, Massachussets, U.S).

Wang, X. H., Zhang, D. Q., Gu, T. and Pung, H. K. (2004). Ontology based context modeling and reasoning using OWL, in *Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops (PERCOMW '04)* (IEEE Computer Society, Washington, DC, USA), ISBN 0-7695-2106-1, p. 18.

Want, R., Hopper, A., Falcão, V. and Gibbons, J. (1992). The active badge location system, *ACM Transactions on Information Systems (TOIS)* **10**, 1, pp. 91–102, doi:http://doi.acm.org/10.1145/128756.128759, URL `http://portal.acm.org/citation.cfm?doid=128756.128759`.

Want, R. and Pering, T. (2005). System challenges for ubiquitous & pervasive computing, in *Proceedings of the 27th international conference on Software engineering (ICSE '05)* (ACM Press, New York, NY, USA), ISBN 1-59593-963-2, pp. 9 – 14, doi:http://doi.acm.org/10.1145/1062455.1062463.

Want, R., Schilit, B. N., Adams, N. I., Gold, R., Petersen, K., Goldberg, D., Ellis, J. R. and Weiser, M. (1995). An overview of the PARCTAB ubiquitous computing experiment, *Personal Communications* **2**, 6, pp. 28 – 43, doi:10.1109/98.475986.

Weiser, M. (1991). The Computer for the Twenty-First Century, *Scientific American* **265**, 3, pp. 94–104.

Wilensky, U. *et al.* (1999). NetLogo, *Evanston, IL* .

Witten, I. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques* (Morgan Kaufmann Publishers).

Wohltorf, J., Cissee, R., Rieger, A. and Scheunemann, H. (2004). Berlintainment: An agent-based serviceware framework for context-aware services, in *Proceedings of 1st International Symposium on Wireless Communication Systems - ISWCS 2004*.

Wood, M. F. and Deloach, S. A. (2000). An overview of the multiagent systems engineering methodology, in *The First International Workshop on Agent-Oriented software Engineering (AOSE-2000*, pp. 207–221.

Wooldridge, M. (2000). *Reasoning about Rational Agents* (The MIT Press, Cambridge, Massachusetts).

Wooldridge, M. (2002). An Introduction to Multiagent Systems, *West Sussex, England: John Wiley and Sons Ltd* **348**.

Wooldridge, M. and Jennings, N. (1995). Intelligent Agent: Theory and Practice, *Knowledge Engineering Review* **10**, pp. 115–152.

Wooldridge, M., Jennings, N. R. and Kinny, D. (2000). The gaia methodology for agent-oriented analysis and design, *Journal of Autonomous Agents and Multi-Agent Systems* **3**, pp. 285–312.

Wu, H., Siegel, M. and Ablay, S. (2002). Sensor fusion for context understanding, in *Proceedings of IEEE Instrumentation and Measurement Technology Conference* (Anchorage, AK, USA).

Xu, M., Lowey, L. and Orwell, J. (2004). Architecture and algorithms for tracking football players with multiple cameras, in *IEEE Workshop on Intelligent Distributed Surveillance Systems* (London), pp. 51–56.

Yang, B. and Garcia-Molina, H. (2003). Designing a Super-Peer Network, *Proceedings of the 19th International Conference on Data Engineering (ICDE)* , p. 49.

Yang, Y., Hassanein, H. and Mawji, A. (2006). Efficient Service Discovery for Wireless Mobile Ad Hoc Networks, *4th ACS/IEEE International Conference on Computer Systems and Applications* , pp. 571–578.

Yau, S. S. and Karim, F. (2004). A context-sensitive middleware for dynamic integration of mobile devices with network infrastructures, *J. Parallel Distrib. Comput.* **64**, 2, pp. 301–317, doi: http://dx.doi.org/10.1016/j.jpdc.2003.10.007.

You, C.-w., Chen, Y.-C., Chiang, J.-R., Huang, P., Chu, H.-h. and Lau, S.-Y. (2006). Sensor-enhanced mobility prediction for energy-efficient localization, in *Proceedings of Third Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON 2006)*, Vol. 2 (Reston, VA, USA), ISBN 1-58113-793-1, pp. 565–574, doi:

http://doi.acm.org/10.1145/990064.990088.

Young, J. E. and Sharlin, E. (2006). Sharing spaces with robots: An integrated environment for human-robot interaction, in *Proceedings of the First International Symposium on Intelligent Environments - ISIE 06* (Cambridge, England).

Young, J. E., Xin, M. and Sharlin, E. (2007). Robot expressionism through cartooning, in *Proceedings of the 2007 ACM/IEEE International Conference on Human-Robot Interaction* (Arlington, Virginia, USA).

Yuan, X., Sun, Z., Varol, Y. and Bebis, G. (2003). A distributed visual surveillance system, in *IEEE Conf. on Advanced Video and Signal based Surveillance* (Florida), pp. 199–205.

Zambonelli, F., Modena, U., Emilia, R., Jennings, N. R. and Wooldridge, M. (2003). Developing multiagent systems: The gaia methodology, *ACM Transactions on Software Engineering and Methodology* **12**, pp. 317–370.

Zanbaka, C. A., Ulinski, A. C., Goolkasian, P. and Hodges, L. F. (2007). Social responses to virtual humans: implications for future interface design, in *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems* (ACM, New York, NY, USA), pp. 1561–1570.

Zhou, Y., Leung, H. and Chan, K. (1998). A two-step extended kalman filter fusion approach for misaligned sensors, in *Proceedings of the FUSION'98 International Conference* (Las Vegas), pp. 54–58.

Ziemke, T. (2003). What's that thing called embodiment? in *Proceedings of the 25th Annual Meeting of the Cognitive Science Society*.