

Modeling Moving Machines

Peter Bøgh Andersen
Department of Computer Science
Aalborg University
pba@cs.auc.dk

Palle Nowack
Maersk Institute
University of Southern Denmark
nowack@mip.sdu.dk

1 Introduction

In the old days computers didn't move, today they do. This means two things. On the one hand, you want access to the information services you need in whatever place you are. You want to plug your computer into whatever socket is close at hand, receive emails, access files, and watch pornography. And you want the service at any time of the day: banking businesses should be possible at one o'clock in the night if that suits you. On the other hand, you want your car computer to take its point of departure in the exact location you are in when it gives you driving directions. It should not tell you how to get to the ferry in Dover, if you are leaving Cologne. In addition, we want it to be sensitive to the time. Our calendar should alert us to a deadline two days before, not one month after it has passed.

Apparently these are opposite demands: on the one hand, the computer should be immune to its physical and temporal location; on the other hand, it should be sensitive to it.

It is as if the computer is located in two independent spaces. In the *informational* space the requirement is *stability*. We want the computer to maintain its connection to the data we normally need. However, in the *physical* space the requirement is *mobility*. Being the workaholics we are, we want the computer to follow us everywhere and we expect it to produce information that is relevant at our current location and time.

A computer is thus an empty shell, suspended in two spaces from where it gets its only reason of being: the informational space furnishes it with information material, and its space-time coordinates enable it to select those chunks that will please its spoilt owner. The computer is nothing in itself; it is a placeholder, a parasite of the surroundings that provide the goods that prevent its owner from thrashing it. Like postmodern man, the computer is nothing but the associations it contracts to other computers and men. It/I am what others make me.

The life of modern computers (and men) is very different from that of older tools (and men). The affordances of a saw or a hammer are so to speak innate;

a hammer is good for hammering and not for sawing because of its physical shape. The hammer is not supposed to maintain ephemeral relations to do-it-yourself books on hammering, and neither is it required to adapt itself to the particular nail its owner wants to hit, or to the time of the day. A hammer is something in itself because of the way it was born. Relations to other tools are of no importance to it. It couldn't care less.

But modern computers parasite on their surroundings. One may say that for such leeches it is more important which *associations* they can contract, than what they are in themselves. Modern computers are media whose purpose is to pass information from one place to the other. The more information they can pass, the more associations they can contract, the more successful they are, as the sad and immoral history of Microsoft shows. Some theories even claim that you can only have one medium of this kind. This explains why we have only one video standard today, even if several ones competed some decades ago and it explains why Microsoft has acquired monopoly in the market of operative systems. The reason is that the benefits of a medium increases with the number of people that uses it. It is no fun to own the only telephone in the country; the investment is much more rational if there are thousands of people you can phone. Media are immoral in the sense that quality does not matter as long as sufficiently many use it. After some critical number has been reached, one medium wins and the other loses. As Nokia correctly claims, the function of media is *connecting* people, and communication requires standardization.

The gist of these arguments is that good relations to other computers are the most important property of the modern computer. The value of a computer is the sum of the associations it can contract.

From this initial analysis two concepts are important. The first one is the notion of the dual space, the *informational* and the *spatio-temporal* space; the second one is the notion of *relation* or *association*.

We shall use the term "habitat" to characterize the dual space (May et al. 2001, Andersen & Nowack 2002, May & Kristensen, this volume, Kristensen, B.B., May, D.C. & P. Nowack, this volume). The general concept is defined as follows:

- A habitat is a *container*. *Inhabitants* can move into a habitat, live there for a while, and move out again. The *action possibilities* (and as a consequence their *information needs*) of the inhabitants depend upon the habitat they currently inhabit.

We distinguish between *spatio-temporal* and *informational* habitats. This is exemplified in the next section.

The purpose of the rest of the paper is to describe existing habitats in order to get a feeling of their nature and the problems associated with them; then we discuss whether standard analytical techniques in the OOA tradition are able to deal with the phenomenon; finally we propose a few amendments to the OOA tradition that can hopefully become useful in designing mobile technology. In particular, we propose a map annotated with software objects as a possible diagramming technique.

2 The train as a spatio-temporal habitat.

In this section we use the train as an example of a habitat. The data was collected in a one-day trip with the Danish railways.

The train is clearly a spatio-temporal habitat since it is a container you enter and leave. If one observes the passengers in the compartment, two or more distinct behaviors can be seen, including the *travel* and the *office* behavior.

The travel behavior includes *entering, taking a seat, sitting, showing tickets, consulting timetables, rising, and leaving.*

The train information system supports this travel behavior by offering timely information. The main loop of the train information system in the Danish Intercity train is displayed below.

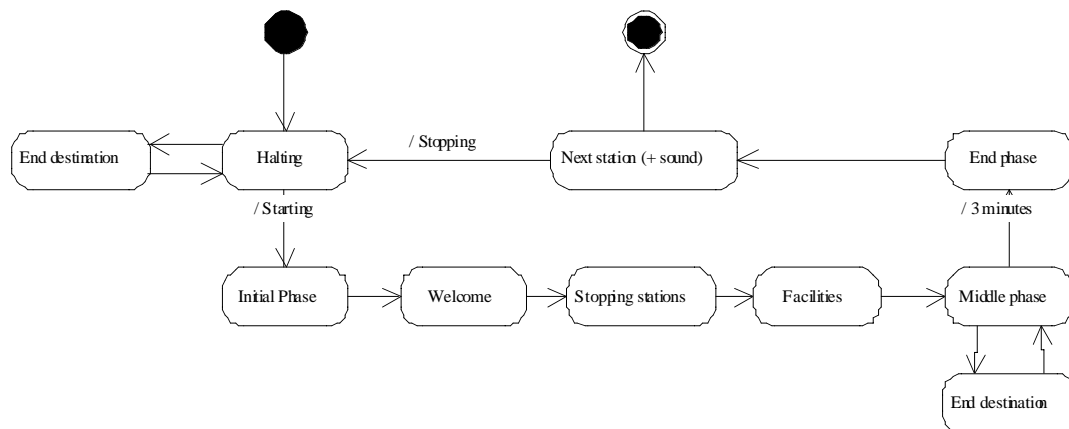


Fig. 2.1. The main loop of the IC train information system. Round rectangles: State. Fat line: train state; thin line: state of display. Arrows: transition from state to state. Labels of arrows: events causing the change of state.

The default display displayed in the middle phase informs the passengers of the end destination of the train: it predicts a potential question from the passengers, *will it bring me to where I want?* and answers it (Fig. 2.2).



Fig. 2.2. The default display

Three minutes before the train stops at the station, information about the station is displayed (answering the question *when should I begin collecting my things?* Fig. 2.3.).



Fig. 2.3. Next station.

This message is also spoken and accompanied by effect sound in order to interrupt whatever the passenger is doing, and return his attention to the travel process: the travel behavior interrupts whatever behavior the passenger indulges in. Speech is also used to inform passengers of connecting trains.

The reason is that people tend not to look up from what they are doing, so important messages must be conveyed in an intruding modality. In the trains we visited, predictable speech was computer-generated whereas unpredictable speech was produced by the personnel.

When the train starts again, it informs the new passengers of the end destination, welcomes them, and lists the stopping stations ahead and the facilities offered by the train. Then it returns to its default state. Apart from this loop, relevant messages about e.g. delays can be sent (Fig. 2.4).

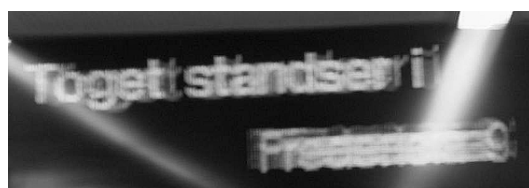


Fig. 2.4. At which stations does the train stop?

It is possible to make a more fine-grained analysis of the train in terms of habitats.

There are different compartments with special behavioral constraints: smokers/non-smokers, ordinary compartments versus “silent” compartments where loud talk is prohibited and where people can work. In night trains, there are special compartments for sleeping. The aisles are for walking and for assembling before disembarkment, whereas the seats are for sitting.

Thus, the same spatio-temporal habitat is decomposed into parts, each with its own behavioral repertoire associated, and some habitats support intertwined different behavioral repertoires.

There are various ways of displaying information relevant to passengers. In the Intercity example above, the bits of information are presented in a temporal sequence where one piece of information replaces the other. On the one hand, this enables the system to display information at the time it is needed, but since the information is not persistent, it does not give the passenger an overview of the travel. This is done in the S-net in Copenhagen (see Fig. 2.5). Here the whole route is displayed simultaneously, and the passenger can all the time estimate whether he is heading in the right direction and how long there is until disembarkment.

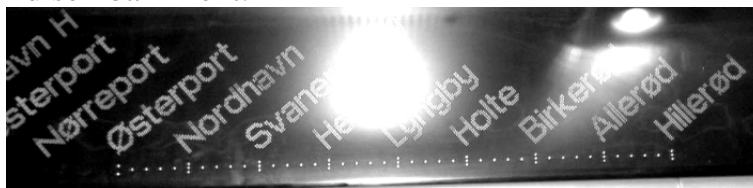


Fig. 2.5. S-net display. *How long has the train got on its scheduled route?*

What this display does not offer is information about connections at the future stations which is relevant if the passenger is to change train. The old-fashioned displays are far better at this (see Fig. 2.6).

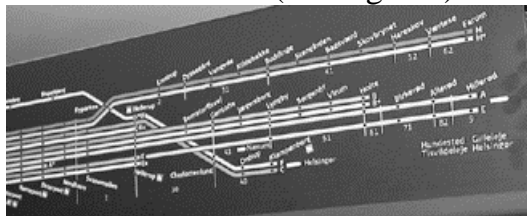


Fig. 2.6. S-net display. Old-fashioned track diagram. *What are the possible connections in the S-net?*

But the drawback of the older display is that it does not provide “runtime”-information about the current position of the train. The passenger has to figure this out himself by looking out the window at the stations. Between the stations, the passenger just has to bite his nails.

The S-net system tries to provide connection information by displaying a ticker-tape in the lower left part of the display before entering a station (sequential information). However, the pace of the ticker-tape was so fast that it was virtually useless (Fig. 2.7.).

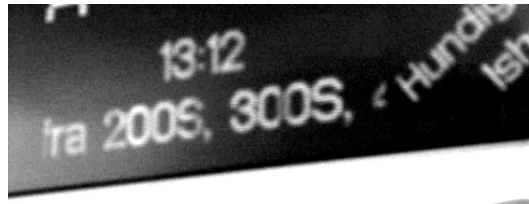


Fig. 2.7. S-net display. Ticker-tape in the S-net

The Intercity system offers a dynamic map showing the position of the train and the future stopping stations by means of diodes.

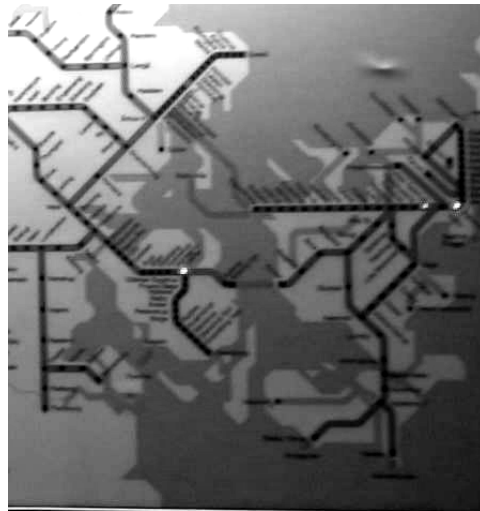


Fig. 2.8. IC-train. Map + diodes.

3 The train as an informational habitat.

A very characteristic feature of the train information system is that it is *self-referential* and *deictic*. It refers to itself both textually (the word “toget”, *the train*, in Fig. 2.4) and graphically (the moving dot in figures 2.5 og 2.8). In many messages “I” or “this” is presupposed. For example, Fig. 3.1 must be interpreted as “I (this wagon) is number 91” and Fig. 2.2 must be understood as “(This train is) going to Copenhagen”.



Fig. 3.1. *This wagon is number 91.*

It is deictic in that it refers to objects in relation to itself, its history, and the time of speaking. The message “Next station” in Fig. 2.3 means “The station immediately *after the one we have just visited*”, and the sign in 3.2 (from a Copenhagen platform) describes departures in terms of “x minutes from *now*”.

Destination	Time	Platform
Ballerup	4 min.	11-12
Køge	6 min.	11-12
Høje Taastrup	7 min.	11-12
Hundige	9 min.	11-12
Frederikssund	11 min.	11-12
Veksø	16 min.	11-12
Høje Taastrup	17 min.	11-12
Køge		

Fig. 3.2. Trains leaving x minutes from *now*

Self-referentiality is a necessary feature of systems whose behavior must be sensitive to the spatio-temporal habitat they are located in. For example, the sign in Fig. 2.3 requires the program to execute a piece of program equivalent to:

```

MyPosition = read PositionSensor
Myspeed = read SpeedSensor
if EstimatedTimeToStop(MyPosition, MySpeed, Schedule) < 3 then
  DisplayText ("Next station" + Nextstation(MyPosition, Schedule))
end if

```

4 Defining the concept of habitats

In Section 1 the general notion of *habitat* was defined as follows

- A habitat is a *container*. *Inhabitants* can move into a habitat, live there for a while, and move out again. The *action possibilities* (and as a consequence their *information needs*) of the inhabitants depend upon the habitat they currently inhabit.

Habitats, like other artifacts, have *affordances* (Gibson 1986) i.e. they afford action possibilities to particular kinds of people, devices and software objects. Thus, a habitat can only be defined with respect to specific types of inhabitants, in the same manner as the Gibsonian concept of affordances is a relation between an environment and a specific type of animal. To the fox, a steep mountainside is an obstacle, to the bird a safe place for nesting. Similarly, a wireless ethernet habitat is only a habitat for devices that can receive and send the signals and that support specific kinds of protocols.

Some habitats maintain *boundaries* and allow only specific objects to enter. This is true of both informational and spatio-temporal habitats. In the former case, we have firewalls to protect the habitats from intruders and passwords to restrict the access to specific information. The train itself is a good example of the latter: you must have a ticket to physically enter then train; if not you are thrown out or you have to pay a fine.

Habitats can be active phenomena with *reactive* and/or *proactive* behavior. The former means reacting to external events generated by users and/or external systems. Examples are automatic door openers, automatic light systems or burglar alarms that react to moving bodies. The ATC system uses sensor stations along the tracks to react to moving trains. The train system described in Section 2-3 is a good example of a proactive habitat: it tries to preview the information needs of the passengers and supply the needed information.

Proactive behavior is becoming more interesting with the shift from traditional to pervasive computing: the habitat of a pervasive system can try to guess what the users or external systems in the habitat want to do. However, this is only possible in spaces that already have a well-codified set of behaviors associated to them, such as airports, trains, and other public places. We can make good guesses at the behavior needs of the flight passenger when in the entrance hall, the departure hall, and at the gate.

4.1 Informational habitats

The informational habitat is a subclass of the general habitat class.

- *An informational (or computational) habitat* is a habitat providing a computational context offering services to the applications running in the habitat. Informational habitats can coincide with a spatio-temporal habitat but need not.

Although both spatio-temporal and informational habitats are spatial, the nature of their space is different. In Andersen 2001 spatio-temporal habitats are

said to have an *Euclidean* space, whereas the space of informational habitats is a *connectivity* space.

The train example shows that informational habitats can be *nested* inside each other. On the one hand, the train is itself a container that provides services to its mobile objects. For example, the business compartment may offer internet connections to laptops, and the train's information system offers services to the passengers moving inside the train.

But the train itself moves through other habitats. One technique for updating the information system of the train with respect to booking seats, is to update it when stopping at stations. In this case, the station is a computational (and spatio-temporal) habitat for the train.

The same phenomenon can be observed with ships. Information about whether conditions, prohibited areas, defect buoys etc. are sent via radio and fax and can be received within a certain area. However, in this case, the informational and the spatio-temporal habitats are not congruent; many messages are irrelevant to the captain's actions since they concern locations many nautical miles away. Important information may drown in noise.

In some cases, thinking in terms of informational habitats would improve the information services. For example, the pilot station in Bremerhafen has a very good radar surveillance system monitoring all in- and outbound ships in the river Elb (Andersen, Carstensen & Nielsen 2002). The data goes to the screens at the pilot station where pilots guide the ships via monotone radio messages: "You are exactly on the line", "You are to the right of the line", "you are to the left of the line". If the ship's computer system had access to this information, the captain could use it himself to steer the ship into harbor.

Although computational and spatio-temporal habitats do not necessarily coincide, they do in many cases for security reasons. Employees may have remote access to the mail-servers of the organization but not to any other servers. If they want full access to the organization's facilities, they must connect to it via stationary computers inside the building. Moveable laptops can be connected via special sockets that give only restricted access.

Inside the individual computer, the operative system constitutes the computational habitat for applications by offering access to necessary computational resources and by enforcing standards. Whoever comes to control this informational habitat, controls the *survival conditions* for software, and in this way gains potentially total control over all software. This is a real liability, as the lawsuits against Microsoft clearly shows.

Thus, computational habitats form *the* environment that determines the survival of computational objects.

4.2 Behavioral repertoires and spatio-temporal habitats

The other important habitat subclass is the spatio-temporal variant:

- A *spatio-temporal habitat* is a slice of physical time and space, whose inhabitants tend to exploit a particular *behavioral repertoire* or *set of repertoires*. The habitat offers informational and physical support for these behaviors.

This definition covers the normal biological notion of habitats that support a particular animal species in finding food and reproducing itself. Even in the biological version, a habitat needs to incorporate time as a dimension, since e.g. birds may use the northern hemisphere in the summer for reproducing themselves, and use the equatorial areas in the winter to find food. The temporal coordinate is even more pronounced in human habitats. For example, shops only afford buying and selling in the opening hours.

Behavioral repertoires can sometimes be divided into subsets, each of which can be found in other habitats. In this case, the habitat is said to house many behavioral repertoires. As mentioned above, a train houses at least two main groups of behaviors

- The *travelling* behavior that includes entering the correct train, finding the right seat, moving out in the corridor a couple of minutes before arrival, and leaving the train at the right station.
- But intertwined with this behavior is another set of behaviors. During our train voyage we observed passengers *writing on their PC*, *talking* in mobile phone, *writing on paper*, *reading the newspaper*, *drinking* and *eating*, *sleeping*, *looking* out of the window.

Some of these behaviors occur nearly unchanged outside the traveling habitat, for example working on a PC. Therefore, the most economical analysis of this is to say that two distinct repertoires of behavior are *overlaid* or *intertwined*.

The informational habitat of the carriage must be so designed that it supports both repertoires at the same time. The business behavior for example would need proper email connections to support modern office behavior.

Of two overlaid repertoires, one can be *subordinate* to the other. In the train habitat, the office behavior is subordinate to the travel behavior, since it is the travel that determine when to interrupt the office behavior: you must interrupt working when the attendant comes to check you tickets, and you must stop completely when arriving at your destination station. Interruption in the other direction is not allowed: you cannot ask the attendant to wait five minutes for the tickets because you are deeply engaged in writing a memo.

Orthogonal to the above characterizations of habitat is the distinction between a *paradigmatic* habitat, providing all behavioral elements and constraints simultaneously, but no order among them (the web is good example of a paradigmatic computational habitat) and a *syntagmatic* habitat, imposing a sequence on the behaviors. The IC-train information system is one good example of this; the chains of emergency signs on passenger ships described in May(this volume) is another example. In the latter case, the arrow-signs indicating the direction to the muster stations must be read in sequence as the passenger moves, and each sign must tell him the direction to go next.

5 Habitat crossing

Habitat crossing means that the occupant of a habitat changes information services, physical location, and behavioral patterns. Habitat crossing is a basic human accomplishment we perform every day. I display one behavior when moving around in a public space, but quite another when secure behind the four walls of my house.

Habitat crossing is often carefully prepared. For example, in an airport train waiting for permission to drive on, the following written message was issued: “The train waits for the moment for permission to drive on” followed by a verbal comment: “We wait for permission so that we get to Kastrup airplane in due time”. The train personnel knows that many passengers are anxious to catch a plane, and try to reassure them that they will arrive at the airport in due time.

Crossing from the train to the platform is also prepared: “Disembarkment in the left side”.

Inside the airport the arriving passenger is guided towards busses and trains by means of signposts (Fig. 5.1).



Fig. 5.1. Sign posts in airport referring to busses leaving: “Bus 9, Nordhavn st.. 14.47, Bus stop A”

But change of informational habitat is not always seamless: for example, the GPS signals may be useless in the transition between two sets of satellites, as every maritime officer knows, and owners of mobile telephones have the same experience with the mobile net.

6 Modeling habitats

In the previous sections we have offered empirical observations of several habitats. Although in the past decades computer gurus with severe Platonic inclinations have preached their grand visions of virtual realities and information spaces that allow us access to unlimited resources of information and free us from the confines of our earthly bodies, the fact seems to be that we still chop up information and behavior in small manageable chunks and associate them to likewise human-scale chunks of space and time for the very good reason that we want to lead a comfortable life and avoid going crazy.

In order to account for this phenomenon, we needed the following list of distinctions:

- *Types of habitats*: informational versus spatio-temporal habitats.
- *Properties of individual habitats*: containers, boundaries of containers, and inhabitants of containers; being located, moving into, and move out of.
- *Relations between habitats*: congruence/incongruence between informational and spatio-temporal habitats, nested habitats, habitat crossing.
- *Habitats, inhabitants, and behaviors*: affordances, behavioral repertoires, subordinate and super ordinate behaviors, intertwined behaviors, reactive and proactive habitats; stability versus mobility, survival conditions, and associations.
- *Properties of displays*: paradigmatic versus syntagmatic displays, self-referential and deictic displays.

We claim that these (and other) distinctions are necessary in order to model mobile computer systems adequately. It is therefore relevant to find out whether existing methods support these distinctions, and, if not, how one can improve upon them. We shall limit ourselves to object-oriented methodology, since it is very widespread, and, we must confess, also the only one we know well.

6.1 A brief overview of object oriented modeling

Object-orientation sees the world as consisting of interacting objects, where an object is a stable association of properties and behavior. Objects (signifying phenomena) can be classified into classes of objects, and these classes can be generalized into larger classes sharing properties and behavior. Classes can also be decomposed into parts belonging to different classes.

As a textual genre, object oriented programs belong to the *descriptive* genre, since it is organized around objects, and actions are only described in relation to these objects.

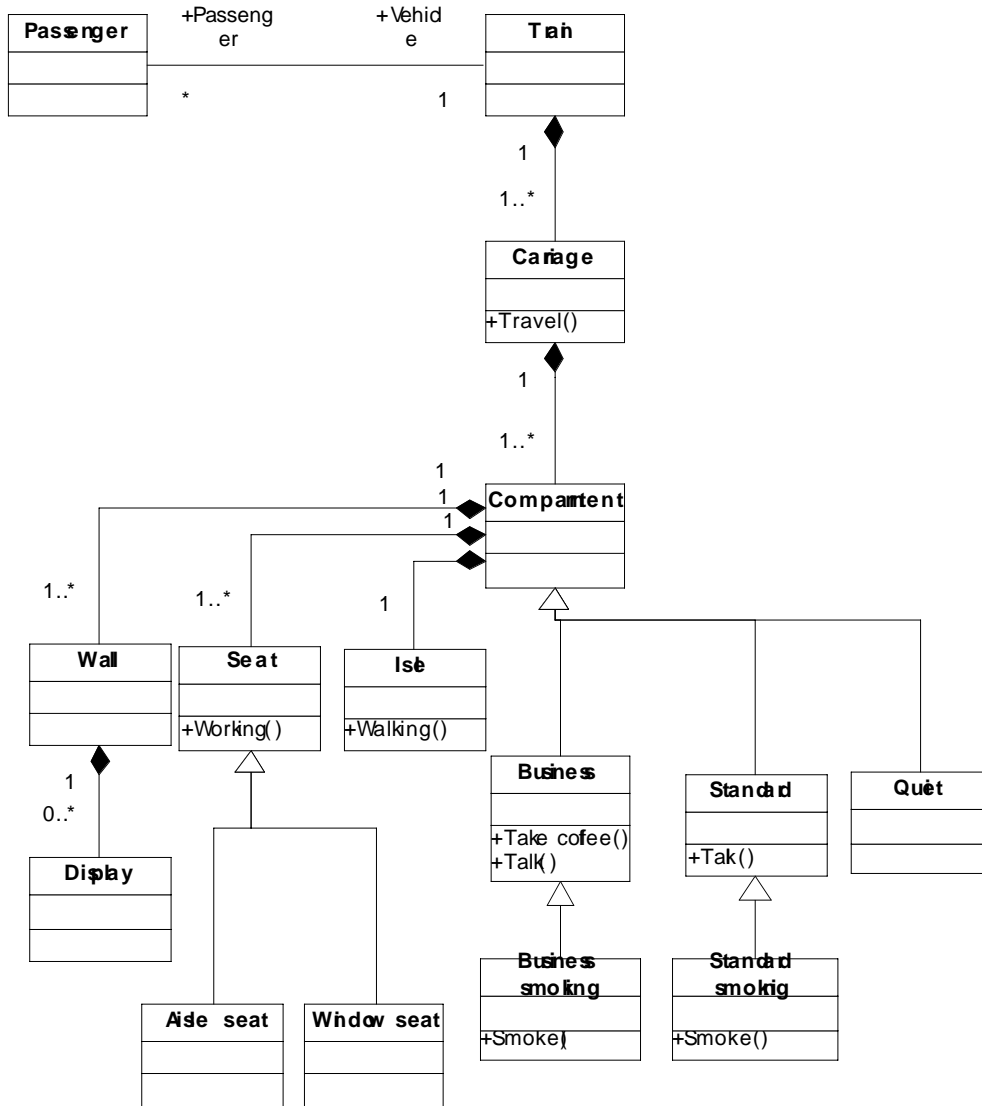


Fig. 6.1. Specialization and decomposition.

An object-oriented model will typically at least include a description of specializations and decompositions, a diagram of the behavior of the individual classes, and a description of the way objects interact with one another by

sending messages. Fig. 6.1 shows an example of classification and decomposition. $\text{---}\blacklozenge$ means *decomposition* (part-whole) and $\text{---}\blacktriangleright$ means *specialization* (class-subclass). The diagram says that trains consist of one or more (1..* means “one or more”) carriages that consist of compartments that consist of walls, seats, and aisles. Walls have displays attached to them. One can exhibit travel behavior everywhere in the carriage, but work behavior is associated to a seat. There are various subclasses of compartments with different behaviors allowed or disallowed.

There exist tools that allow you to draw diagrams like Fig. 6.1 and, in return, will generate code that can run on a computer (Figs 6.2-3).

<pre>class Carriage { public: // methods Void Travel(); End of user code. };</pre>	<pre>Void Carriage::Travel() {}</pre>
--	---------------------------------------

Fig.6.2. C++ code generated by the carriage node

<pre>class BusinessSmoking : public Business { public: // methods Void Smoke(); };</pre>	<pre>Void BusinessSmoking::Smoke() { }</pre>
--	--

Fig.6.3. C++ code generated by the Business Smoking node

The *behavior* can be described by state diagrams that show how the object changes state as a result of events. In Fig. 6.4 we model the behavior of the passenger. The event of entering the train makes the passenger change state to “inside train”, and the event of finding the seat changes his state to “on seat”. In this state three events can happen, the working and travelling events do not change his state, but the “leaving train” event does. Fig. 2.1 uses the same formalism to describe the behavior of the display object.

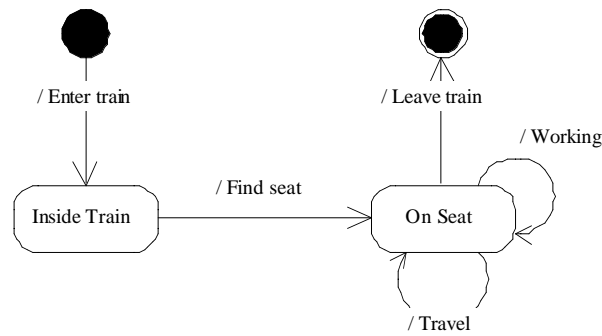


Fig. 6.4. State diagram

The question we want to address is: can this kind of object-oriented methodology describe our findings in a *simple*, *adequate* and *exhaustive* way? The last qualifications are important: a descriptive apparatus can in many cases describe anything, so the interesting question is whether it highlights the features the writer and reader consider important. It is of course possible to write a detective novel in an object-oriented way, but since these novels get their interest from their narrative structure, object-oriented methods will yield extremely boring novels. Compare Text 1 and 2. Text 1 is narrative since the objects are only presented when needed by the action. Text 2 is descriptive since all actions belonging to one object are collected in the same paragraph. Although both texts describe the same events, Text 1 is clearly better than Text 2.

It was typical foggy evening in London. Mr. Sherlock Holmes looked nervously through the yellow mists hanging over the docks, and suddenly discovered a dark furtive shadow moving along the quay. A hundred yards away a slender female shape was standing in the shadow of an old warehouse, anxiously watching the dirty waters of the Thames. The dark shadow stopped, snarled, and was obviously moving towards the female figure. It lingered for a moment, then hurtled itself towards the lonely shape, a faint muffled shriek reached the ears of Holmes, and in the next moment both shapes disappeared in the fog. When he reached the scene he saw that Lady Snodgrass had met a horrible death.

Text 1

London is a town containing a harbor by the river Thames. A yellow dark fog is often hanging over the docks.

Sherlock Holmes is a detective. He follows Lady Snodgrass and watches her. As she is assaulted and killed by her depraved son, he runs to her and discovers the crime.

Lady Snodgrass is a rich widow with a furtive depraved son. She has a slender figure. She was lured by him to the harbor, assaulted at a warehouse, snarled at, and horribly killed.

Text 2

6.2 Types of habitats

An important distinction is that between, on the one hand, the *informational/computational habitat* characterized by the computational services it offers visiting objects, and on the other hand the *spatio-temporal habitat* characterized by the behavioral affordances it offers its inhabitants. The former is a software concept, the latter concerns the application domain. The main thesis is that they are subclasses of the same general class, namely the habitat. The software domain is normally described by class-diagrams like fig. 6.1, whereas the application domain, i.e. the interaction between the tasks of the user and the services offered by the system, is described by so-called use-cases. But class-diagrams and use-cases are not specializations of a common class and OOA therefore cannot see informational and spatio-temporal habitats as subclasses of the same super-class.

One reason for this is that the software models are sharply distinguished from their implementation. There are perfectly good reasons for this. The software models reduce complexity by capturing the essentials of the software and disregarding details of implementation. This means that software models are not located in space and time. But if the location of computational objects in space and time is essential, then it ought not to be relegated to implementation but must have a systematical place in the model.

A similar problem is encountered in Kristensen, May, & Nowack (this volume) that discuss the notion of *tangible objects*, i.e. associations of informational and physical objects. The problem consists in developing good abstractions that enables us to view the tangible object as a unity, not as a split Cartesian personality divided into a spiritual (here: informational) and a material (here: physical) world. The same desideratum is voiced by Kjeldskov (this volume) in the form of a design pattern for augmented reality systems: the relation between physical space and virtual objects (= informational space) should be strengthened, for example by designing interaction methods that makes interaction in physical space a part of the human-computer interface,

It is worth emphasizing that we are concerned with *modeling* the relation between software objects and space; however, a description of *what* the system is to accomplish does not necessarily coincide with *how* it should achieve it. For example, *spatial computing* (Grønbæk et al 2001: 409) aims at using a whole room as the interface to computational objects and supporting movement of information objects from walls to tables to PC's. However, this can be implemented by means of traditional servers keeping track of the location of the display of the objects and does not imply that the objects themselves will move.

6.3 *Properties of individual habitats*

As mentioned in Section 4, a habitat is a container with boundaries within which inhabitants can live. Inhabitants can move in and out and reside inside the container. Some boundaries only tolerate objects with certain properties to pass their boundaries.

OOA has essentially two ways of modeling containers with inhabitants. The *decomposition* (part/whole) relation is used to represent stable “inhabitants” of objects, and the *association* is used to model transient ones. Diagram 6.1 therefore uses decomposition to model the relation between trains and carriages, since trains always have one or more carriages. However, it uses the association to model the relation between passengers and trains, since trains may have no passengers, and passengers may not be on board trains.

Language makes the same distinction; the verb *have* and the genitive denotes both relations, and both comes in two variants: *non-alienable* (parts: “My eyes”) and *alienable* (e.g. possession “My car”). But *have* has a third function, namely to denote general relations described elsewhere in the sentence. The phrases *She has a teacher*, *her teacher* neither means that the person is a part of her nor that she possesses him; it means that a specific relation obtains between them, that of *teaching*: *he teaches her*. This general meaning of “relation” seems to be the basic one, since the other two can be reduced to it: “She has two legs” can be rephrased as “Two legs are parts of her body” or, somewhat morbid of a murderess, “She possesses two legs (from her victims)”.

The term “passenger” is also a relational term at closer scrutiny. Only when I have bought the ticket and entered the train do I assume the role of passenger with the rights and obligations it entails. When my travel has ended, my role as passenger also ends, although I persist as a person. To say that I am a passenger of the train means that the train is obliged to bring me to my destination.

The relation of a habitat to its inhabitants seems in fact to be of the relational kind. The behavior afforded by a habitat is, as Gibson knew, a relation between a particular species of an animal and its surroundings. Similarly, the relation between a passenger and a train is neither part/whole nor possession; rather, it can be described as a kind of protocol specifying the interaction between train personnel and passenger: the passenger is entitled to enter the train; he is entitled to a specific seat if he has a reservation, otherwise he may have to stand up; the attendant has the right to check his ticket; etc., etc. To claim that the concepts of habitat/inhabitant are relational is to claim that we can predict very little about the events taking place if we only inspect the

habitat or inspect the inhabitant. As said in the beginning, they are just end-points of relations.

If this is true, then it seems somewhat inconvenient to make the decomposition relation primitive. Instead we would need a concept of *role* connecting two objects and specifying the protocol for their interaction¹.

6.4 Relations between habitats

As we saw in the maritime example, informational and spatio-temporal habitats may be incongruous. A sensible design objective would therefore to design the two habitats so that former fits the needs of the occupants of the latter. Can we express fitness between habitats systematically in OOA? The answer is *no* if the claim in Section 6.1 is true, namely that we cannot deal systematically with the two phenomena in the same model.

What about *nested habitats*? We need this concept in the transportation domain: the means of transportation is a habitat for the passengers, and is itself transported in and out of other habitats. It turns out that nesting is in fact well supported in OOA, since one can define objects within other objects.

The final important relation between habitats we came across is *habitat crossing*. We need to model the gradual transition of an object from one habitat to another in order to design the information support we observe in the real world (e.g. Fig. 5.1 that supports movement from the airport habitat to the bus habitat). Can we express the notion of something continuously changing role and location? Not immediately, at least.

We need to model the habitat as a spatial structure which the object can move into and out of. But associated to this movement must be a change of the role-relationships that defines the habitat whose strength changes. When we are in the airport it is strong, but as we move out of the airport, the airport's responsibilities weaken, and is finally handed over to the bus company. We gradually cease to be an airport passenger and start becoming a bus-passenger. Therefore we need to associate a strength to the role relationship, a kind of variable *glue* that glues the participants to the role and defines the habitat's *sphere of influence*. The strength of the glue may be calculated on the basis of the distance between the person and spatial habitat, but other definitions may also be useful.

If it stops abruptly as the passenger leaves the habitat, the airport is not invited to think about the passenger's fate once outside the doors of the airport,

¹ This problem has in fact been noted, and a design pattern that circumvents the problem has been invented, namely the *role*-pattern. Kort om dette design pattern.

since the airport's responsibility stops here. The result is missing information and confused passengers that cannot find busses, hotels, taxis and trains.

If instead the glue decreases gradually, and retains some of its strength some hundred meters away from the buildings, the designer will think of this area as still within his responsibility, although to a lesser degree. He will consider it his obligation to support the passenger in his role-shift, from airplane passenger to bus passenger, and may eventually hit upon the idea of designing Fig. 5.1. If the bus-company feels the same, there will be an area of shared responsibility where the object is weakly both an airplane and a bus passenger, and where supporting information facilities will therefore be motivated.

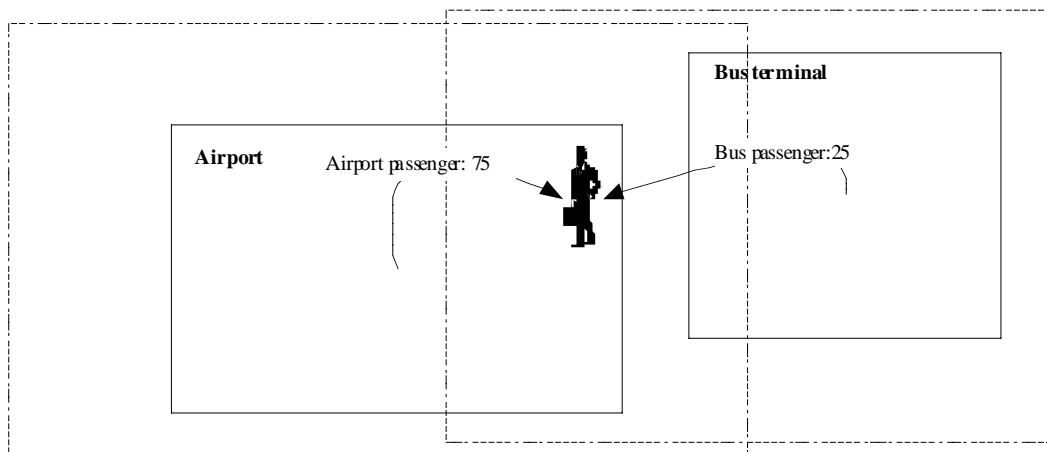


Fig. 6.5. Habitat crossing. Full lines: spatio-temporal habitats; dashed lines: the sphere of influence of informational habitats. Arrows define role-relationships and their strength. They lead from the center of the informational habitat to the passenger. Note the location of the bus-sign,

This means that the informational habitat will be larger than the spatio-temporal one, and that the two informational habitats will overlap. Fig. 6.5 shows one way of diagramming this. Although the spatio-temporal habitats (full lines) are disjunct, the sphere of influence of their informational habitats (dashed lines) are overlapping. The two role relationships and their strength is represented by two labeled arrows anchored inside the two habitats and directed towards the passenger.

6.5 Habitats, inhabitants, and behaviors

The distinction between *reactive* and *proactive* habitats is easy to express. In the former, there is a direct connection between sensor information and the reaction of the habitat. The following piece of code could control the doors in the airport.

```

If something moves in the vicinity then
  open the door
  wait for 20 seconds
  close the door
end if

```

The proactive habitat does not react directly but make statistical guesses that a certain situation obtains. It tries to estimate the future evolvement of this situation and present information that will hinder undesired and enhance a desired outcome. Here is an example:

Situation assessment:

```

If a new PDA enters the airport and
  contains an electronic ticket and
  the flight-number is boarding
then there is x % chance the owner wants to catch the flight and
  y % chance that he will miss it
end if

```

Avoidance of non-desired outcome:

```

If there is x % chance that someone wants to catch a flight and
  y % chance that he will miss it
then alert him
  display the route to the gates on his PDA
  alert the gate that the passenger will arrive late
end if

```

This system tries to avoid that the plane leaves without the passenger by alerting the gate, and supports him in getting quickly to the gate by displaying the route to him. The distinction between reactive and proactive objects are well-known in agent-theory that distinguishes between reactive and deliberative agents.

The notion of *intertwined* behaviors are also well known in programming. Most modern languages support more than one thread of execution; conceptually, these threads run concurrently, and one thread can interrupt another, one acting as super ordinate to another.

6.6 Behavioral repertoires

It is a bit more difficult with the notions of affordances and behavioral repertoires. We have argued that the habitat/inhabitant distinction is really an association between two entities entering a specific role relation. Thus, affordances and behavioral repertoires take the form of protocols regulating the interaction between the two parties. As noted in Section 6.4, these protocols must be sensitive to the strength of the role relations which again depend upon the physical location of the inhabitant. Some (syntagmatic) protocols execute differently depending upon the time of the day and the location of the

inhabitant. Finally, the same person can enter in more than one role relation when located on a particular position.

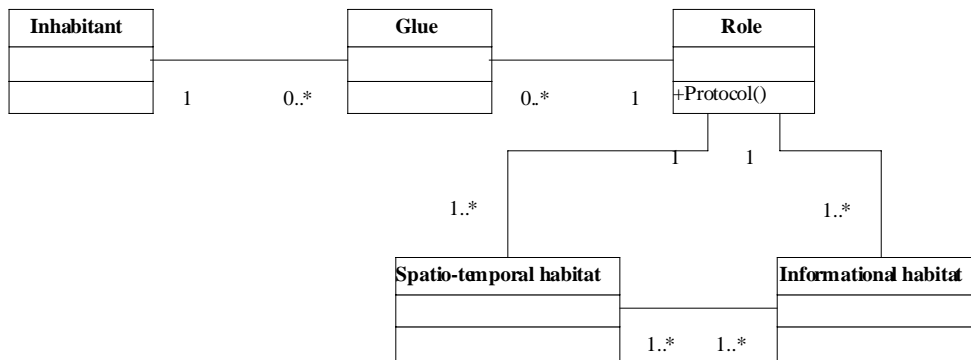


Fig. 6.6.

Clearly these ideas can be described by a normal class diagram as Fig. 6.6. It says that inhabitants can be associated to zero or more glue-objects that are associated to exactly one role and inhabitant each. Roles are associated to zero or more glue-objects, and to one or more spatial/temporal and informational habitat. Finally, informational habitats can be associated to one or more spatio-temporal habitats, and vice versa.

The last requirement can clearly be questioned: it asserts that the train-passenger role is always associated with the informational habitat of trains, and with a location in relation to the train territory. Thus, it asserts a strong association between role, spatial location and informational context.

It is easy to find examples where this association does not exist. For example, you can sit at home as a potential passenger and book a seat via the internet. In this case, you have a role in the informational habitat of the railroad company, but no role in relation to its physical territory. Fig. 6.6 is a false model of this situation, and we have two options for correcting the situation: the first one is to modify the model to fit the situation, e.g. by removing the demand that a role and an informational habitat are obligatorily connected to a spatial habitat. This option is chosen by May & Kristensen, this volume; on the one hand it gives a wider area of application, but, on the other hand, it runs the danger of emptying the habitat concept for content. The other option is to claim that not everything is a habitat. A habitat is a *special* way of organizing social roles, physical surroundings, and information. Some ways of doing this are habitats, others are not. The railroad station and the trains are habitats, the www-services are not.

6.7 Maps annotated by software objects

In this section we shall suggest one way of conceptualizing physical and informational space.

Fig. 6.6 is a traditional diagram that tries to capture the relation between informational and physical space, but it does not give us an immediate impression of the important part of the system, the relation between informational and physical objects. The reason for this is the diagram type. It belongs to the class of *conceptual diagrams* where space is used to denote conceptual relations. However, we have just decided that spatial relations are important in the habitat concept, and would like to have a representation that emphasizes this. One such representation is the old-fashioned *map*. A map is a graphical representation that uses space to represent space, although all maps superimpose additional information called *signatures*: place names, types of roads, icons for types of buildings, bus- and train routes, etc. However, the strong point in a map lies in its iconic relationship to its denotation: what is to the left in the map is also to the left in reality. However, real maps will always have symbolic material attached to it, such as signatures and names. Thus, maps are basically iconic signs with symbolic material superimposed (May & Andersen 2001).

This suggests a solution: in so far as space is important in habitats, maps decorated with pieces of code, parts of protocols, and information needs is a natural model. Thus, software objects, protocols and information needs correspond to the signatures and names in ordinary maps. Fig. 6.7 shows an example. It models a railroad station which is predominantly a paradigmatic habitat and annotates the map with information needs (the balloons) associated to specific behavioral habitats (dashed rectangles). In the main hall, passengers may need information about all trains leaving the station within the next half hour, while they only need to know about trains leaving a particular platform when they are placed at the entrance of the platform. Passengers seated in the cafeteria still need to know when their train leaves even if they are eating. At present, these information needs are fulfilled by stationary monitors, timetables, and menus. The ones shown in the figure can be observed in any railroad station.

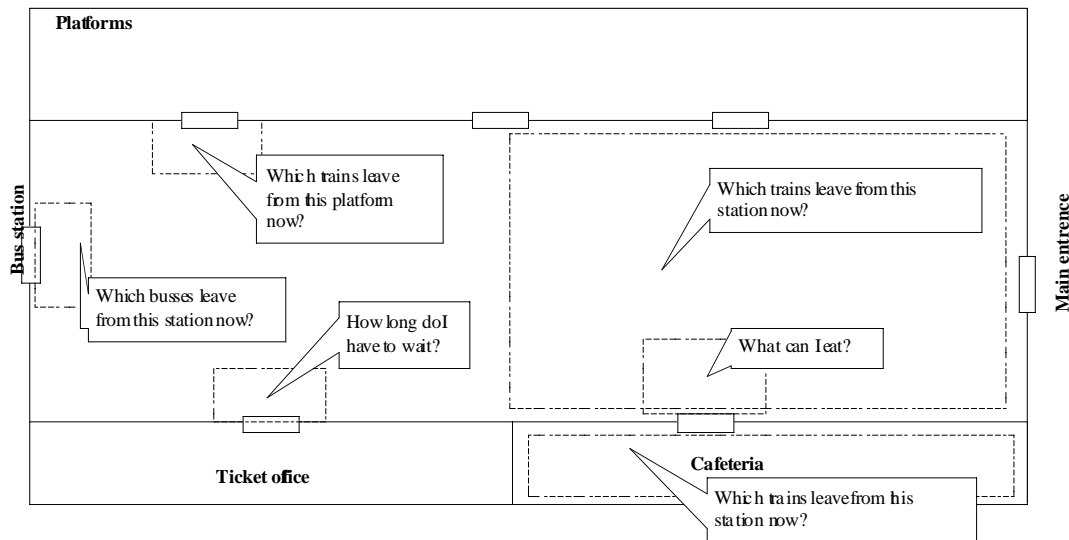


Fig. 6.7. Paradigmatic habitat modeled by a two-dimensional map decorated with information needs in balloons. Informational habitats are dashed.

Another map would show the pieces of protocol that motivate the information needs: to the platform entrance will be associated the bit saying that a the company is obliged to transport a passenger with a correct ticket but is not required to wait for him — it is the passenger’s responsibility to be at the right place at the right time; at the ticket counter belongs the bit saying that the passenger is obliged to buy a ticket, but has to stand in a queue until it is his turn.

Can the same be done with software-models? Fig. 6.8 shows a sketch of a diagram that captures some of the concepts. It is basically a map of an airport that is decorated by informational habitats and inhabitants and by the role relations that motivate the informational habitat. Within an informational habitat, the corresponding service is offered. For example, when a person enters by the rightmost door, arrival and departure information plus warnings are offered. The user can configure his PDA to only accept some of these services. Balloons show the services requested by the user. The topmost PDA, for example, is in the airport to welcome a friend, and therefore only requests arrival information. The rightmost one has requested access to four pieces of software: the Wayfinder, the Liquor Store, the Warnings, and the Departure Information. At the moment only the latter two requests has been fulfilled, shown by the plusses and minuses. If the PDA contains an electronic ticket, it would be possible for the airport system to turn on the Warnings and Departure System automatically, since it can figure out that they will probably be needed by the passenger. If given access to the ticket, the Warning System could inform the individual passenger of the flight he is about to take, since it would know the flight number. Finally, the leftmost PDA is offered a *passen-*

ger relationship to the Wayfinder, and a *customer* relationship to the Liquor Store, which it has in fact accepted. The Warnings and the Departure Information are no longer available in this constructed example.

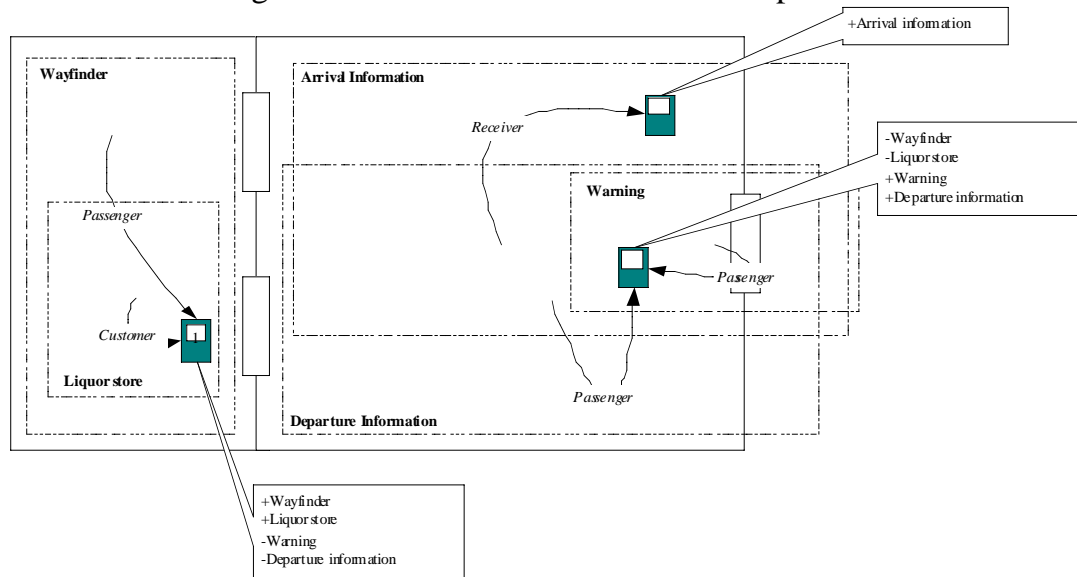


Fig. 6.8. Map with informational habitats and social roles superimposed.

Fig. 6.8 is one way of localizing software models. It uses a kind of market model: inside the sphere of influence of the informational habitat, information services are supplied. If an offer is accepted the two parties enter into a temporary specific role-relationship, e.g. salesman-customer. The supplies are represented by the names of the informational habitats; the demands by the text in the balloons; and the concluded bargains by the minuses and pluses in the balloons.

7 Philosophical implications

We hope to have shown that the habitat concept is a useful one: it can be used to analyze existing usage of spatial information systems and to plan future ones. However, it also has more general implications. It essentially claims that humans are not fixed entities, but change attitudes, action potential and language as they move from one context to another. Humans are post-modern chameleons that change color, dependent upon their surroundings. As usual in the history of technology, technology works as a mirror that highlights unnoticed features of human nature, and, possibly, even creates them?

Let us treat the implication as a scientific theory; are there anomalies in traditional theories it can explain better? One of the authors who is a linguist by education has always been vexed by the following problem: according to dictionaries, most words have multiple senses. For example, Microsoft

Word's thesaurus yields 6 senses for the word *move*: *go*, *reposition*, *change*, *attempt*, *shift*, *cause*. Now, if each word has 6 senses, then a sentence of 10 words will have 6^{10} senses. This is indeed a large number! And in a split second, the listener has to choose the one intended from this huge number.

This is simply not possible, so where did we go wrong? The six senses are a fact, check it yourself. Therefore, we must conclude that we do not ourselves use Bill Gates' lexicon when we understand sentences. But, then, what *do* we do? The riddle can be solved if we hypothesize that language is used in habitats too. A linguistic habitat would be a *situation type*, associated to a specific time and place, with a delimited number of action possibilities, information needs, and *linguistic resources*. Situationally bound linguistic resources are called *registers*, and they are assumed to offer exactly those distinctions needed in the habitat, and none else. In a particular register, *move* may only have 1 or 2 senses, so the problem of disambiguating sentences has been made exponentially easier. The implication is that national languages, as we can see them in lexicons, are a fiction. What exists in reality is registers, or, to use the vocabulary of this volume: *habitats*.

8 References

- Andersen, P. Bøgh & M. May. (2001). Instrument Semiotics. In: *Information, organisation and technology. Studies in organisational semiotics*. (eds. Kecheng Liu, Rodney J. Clarke, Peter Bøgh Andersen, Ronald K. Stamper). Kluwer: Boston/Dordrecht/London. 2001: 271-298.
- Andersen, P. Bøgh & P. Nowack (2002). Tangible Objects: Connecting Informational and Physical space. In: L. Qvortrup (ed.): *Virtual Space: Spatiality of Virtual Inhabited 3d Worlds*. Springer Publishers, London: 190-210.
- Andersen, P. Bøgh (2001). Pervasive computing and space. In: Ronald Stamper, Kecheng Liu & El-Sayed Abou-Zeid (eds.): *Proceedings of IFIP WG8.1 Working Conference. Organizational Semiotics: evolving a science of information systems*. Montreal, Quebec, Concordia University, 2001: 106 – 125.
- Andersen, P. Bøgh, P. Carstensen & M. Nielsen (2002). Means of Coordination. In: K. Liu, R. J. Clarke, P. Bøgh Andersen & R. K. Stamper (Eds.). *Coordination and Communication Using Signs. Studies in Organisational Semiotics*. Kluwer: Boston, 32-58.
- Gibson, J. J. (1986). *The ecological approach to visual perception*. Erlbaum: Hillsdale, New Jersey, London.
- Grønbaek, K., K. Gundersen, P. Mogensen & P. Ørbæk (2001). Interactive room support for complex and distributed design projects. In M. Hirose (ed.), *Human-Computer Interaction. INTERACT '01*. IOS Press: Amsterdam, 407-414.
- Kjeldskov, J. (this volume). Lessons from being there: interface design for mobile augmented reality.

- Kristensen, B.B., May, D.C. & P. Nowack (this volume). Conceptual abstraction in modeling with physical and informational material.
- May, D.C. & B.B. Kristensen (this volume). Habitats for the display of the digitally pervasive world.
- May, D.C., B.B. Kristensen & P. Nowack (2001). TangO: Modeling in Style. *Proceedings of the Second International Conference on Generative Systems in the Electronic Arts*. Melbourne, Australia.
- May, M. & P. Bøgh Andersen (2001). Instrument Semiotics. In: Information, organisation and technology. *Studies in organisational semiotics*. (eds. Kecheng Liu, Rodney J. Clarke, Peter Bøgh Andersen, Ronald K. Stamper). Kluwer: Boston/Dordrecht/London: 271-298.
- May, M. (this volume). Wayfinding, ships, and augmented reality.