

Situations in Conceptual Modeling of Context

Patrícia Dockhorn Costa¹, Giancarlo Guizzardi^{2,3}, João Paulo A. Almeida⁴,
Luís Ferreira Pires¹, Marten van Sinderen¹

¹Centre for Telematics and Information Technology, University of Twente, The Netherlands

²Computer Science Department, Federal University of Espírito Santo (UFES), Brazil

³Laboratory for Applied Ontology (LOA), ISTC-CNR, Trento, Italy

⁴Telematica Instituut, The Netherlands

dockhorn@cs.utwente.nl, joaopaulo.almeida@telin.nl, pires@cs.utwente.nl, guizzardi@loa-cnr.it, sinderen@cs.utwente.nl

Abstract

In previous work, we have defined conceptual foundations that can be beneficially used in context modeling. These conceptual foundations include the separation of entity and context, and the characterization of context as either Intrinsic or Relational. This paper aims at extending this approach by introducing the ontological concept of Situation as means of composing the elements of our ontology (entities, intrinsic and relational contexts) to model particular states of affairs of interest. Our concepts have been inspired by and aligned with conceptual theories from the fields of philosophy and cognitive sciences.

1. Introduction

Context-awareness has emerged as an important and desirable feature in ubiquitous applications. This feature deals with the ability of applications to utilize information about the user's environment (context) in order to tailor services to the user's current situation and needs [3].

In order to address both the users' demands for innovation and the interest of service providers to offer more attractive services, new kinds of context are frequently incorporated into applications. In addition, sophisticated context reasoning is used.

As applications become more complex and interconnected, there is an increasing need for context modeling abstractions that are appropriate to: (i) characterize the application's universe of discourse; to (ii) support common understanding, problem-solving, and communication among the various stakeholders involved in application development [5]; and to (iii) represent context unambiguously.

In [2], we have discussed the need to adequately characterize the application's universe of discourse. As a result of this characterization process, we expect a context model, which is a conceptual model (in the sense of [9]) of context. We argued that the definition of such a context model should precede the detailed design of a context-aware application.

As part of that work, we have proposed basic conceptual foundations for context modeling, which allow designers of context-aware applications to represent relevant elements of a context-aware application's universe of discourse. These conceptual foundations should facilitate the specification of context models that are clearer and easier to understand.

As a basic distinction, we have proposed the separation of the concepts of *entity* and *context*. We have also proposed that context should be characterized as either *intrinsic* or *relational*. We have motivated our concepts by relating them to developments in foundational ontologies [5], which are in line with conceptual theories in the areas of philosophy and cognitive sciences.

The models we have discussed in [2], allow application designers to represent all possible states of affairs of an application's universe of discourse, without discriminating particular situations that may be of interest to applications. For example, while we could model that a person may be near another person and that a person may own a mobile phone, we could not explicitly model particular situations such as "John is near Alice and their mobile phones are available" or "John has a fever and influenza". Since we could not individuate these situations, we were also unable to predicate them.

We aim in this paper at extending our conceptual foundations with the ontological concept of Situation. Situations are genuine ontological entities [1, 6] that

model particular states of affairs which are of interest for applications. In our approach, Situations are composite concepts, whose constituents are the elements of our ontology. We discuss here how application designers can compose Entities, Formal Relations, and Intrinsic and Relational Contexts, to yield Situations. Further, we introduce the concepts of Situation Type and Situation of Situations.

The remainder of this paper is structured as follows: Section 2 identifies relevant characteristics of context and introduces our context categorization scheme. Section 3 discusses the categorization scheme in more detail. Section 4 introduces the concept of Situation and motivates its use in context-awareness; Section 5 discusses related work. Finally section 6 presents our conclusions.

2. Characteristics of Context

Context can be defined as “the interrelated conditions in which something exists” [8]. This definition reveals that context is only meaningful with respect to a thing (that “exists”), which we call here an entity.

The concept of entity is fundamentally different from the concept of context: context is what can be said about an entity, i.e., context does not exist by itself. Examples of entities are persons, computing devices and buildings. The context of an entity can have many constituents (“interrelated conditions”). Examples of some constituents of the context of a person are the person’s location, mental state, and activity. In the remainder of this paper, we use the term context to refer to constituents of the context of an entity. Together, these constituents form the entity’s context.

The process of identifying relevant context consists of determining the “conditions” of entities in the application’s universe of discourse (e.g., a user or its environment) that are relevant for a context-aware application or a family of such applications. The representation of these relevant conditions or circumstances is called here a context model. We define a context model as a conceptual model (in the sense of [9]) of context.

In [2], we have drawn a parallel between the concepts proposed here for context and those defined elsewhere for foundational conceptual models [5, 11, 10]. We summarize our results in the following sections.

3. Ontological Foundations

Universals and *individuals* are fundamental categories that have been considered in our modeling abstractions. Universals are predicative terms that can possibly be applied to a multitude of individuals [5]. Intuitively, individuals refer to instances, while universals refer to types. We focus here on context models that capture the general aspects of context, and therefore, we only represent universals. We define a universal for entities and a universal for context, namely, *Entity* and *Context*, respectively. For example, the *Entity* type *Person* and the *Context* type *Location* are universals, while John and his actual location are individuals (instances of these universals), respectively.

Universals can be categorized as *substantial* or *moment* [10]. A moment is an individual that *existentially depends* on other individuals, named its *bearers*. In addition, a moment should also *inhere* on its bearer(s), the way mood inheres in a person and a smile on a face. Substantials are universals that do not inhere in other universals, i.e., which are not moments. Inherence is much stronger than a one-to-one relationship, since it implies existential dependence between individuals. Figure 1 summarizes these concepts.



Figure 1 - Fragment of foundational concepts

Considering the fundamental categories mentioned above, we argue that *Entity* and *Context* types should be classified into *substantial universal* and *moment universal*, respectively. Since entities do not inhere in other entities, they cannot be moments, and therefore they should be classified as substantials. On the contrary, contexts always inhere in other entities, and therefore, they should be classified as moments.

Figure 2 depicts the relationship between the concepts of Context and Entity. In this paper, we represent context models as UML class diagrams because of UML’s widespread adoption. We use the stereotypes `<<SubstantialUniversal>>` and `<<MomentUniversal>>` to denote explicitly that *Entity* and *Context* are categorized as substantial and moment universals, respectively. Context types are depicted as shaded rectangles to facilitate readability.

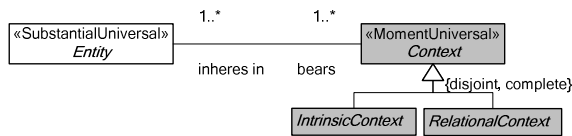


Figure 2 – Basic context modeling concepts

We distinguish two categories of context, namely intrinsic context (*IntrinsicContext*) and relational context (*RelationalContext*).

3.1 Intrinsic Context

Intrinsic context defines a type of context that belongs to the essential nature of a single entity and does not depend on the relationship with other entities. Figure 3 depicts examples of intrinsic context types. Geographic location (*GeoLocation*) is context that inheres in all spatial entities. Spatial entities are *bearers* of *GeoLocation*. Similarly, battery power (*BatteryPower*) inheres in a device. Analogous reasoning can be applied to other context types depicted in this figure.

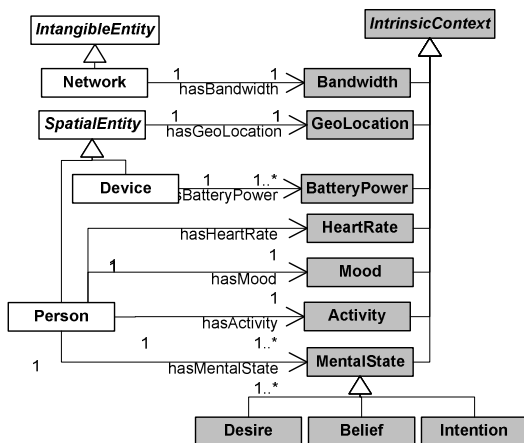


Figure 3 - Intrinsic Context Types

Intrinsic context types discussed in this paper are classified as the ontological notion of *quality universal*. Quality is an intrinsic moment that can be mapped to a value (*quale*) in a quality dimension [5]. A quality dimension defines the possible set of values a quality type might be associated with. The geographical location of an entity is an example of quality, whose quality dimension is defined by all possible values in a geographical coordinate system.

The quality of an entity is an intrinsic objectified property of that entity, thus, even if two entities are co-located, they do not necessarily have the same location quality in the strong sense. Co-location depends on the

granularity of associated quality dimension. For instance, take two different quality dimensions Q , Q' associated with the quality universal location such that $Q = \{list\ of\ names\ of\ civil\ locations\}$, $Q' = \{precise\ GPS\ location\ value\ space\}$. Under these circumstances, we can have that two entities are considered co-located in the quality space Q but not in Q' . In other words, the accuracy of our comparisons of entities' intrinsic properties depends on the precision of our quality dimensions.

Figure 3 also presents examples of intrinsic context types of a person, such as the person's current activity, mood and mental state. These context types are quite subjective and difficult to measure. However, one could conceptualize an objective notion for these context types in a context-aware application, by enumerating the possible values (quality dimension) with which each of these types may possibly be associated. For example, we may say that the possible values of a person's mood are: "happy", "sad", "bored", "tired" and "moody"; and the possible values of a person's current activity are: "working", "dancing" or "attending a meeting".

3.2 Relational Context

While intrinsic context information inheres in a single entity, relational context information inheres in a plurality of entities. Figures 4 and 5 show examples of relational context.

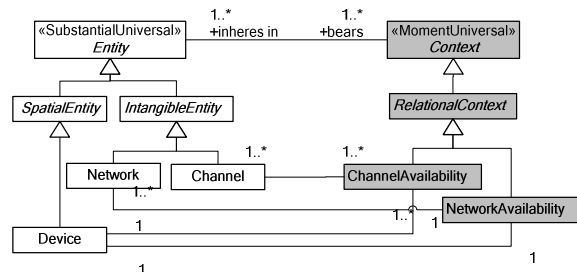


Figure 4 - Relational Context Types

Figure 4 shows that relational context may be used to relate an entity to the collection of entities that play a role in the entity's context. Examples of relational context are *DeviceAvailability*, *NetworkAvailability*, *SocialNetwork* and *ChannelAvailability*. The *DeviceAvailability* relational context relates a person to a collection of devices that are available to that person. *NetworkAvailability* relates a device to a collection of networks that are available through that device, *SocialNetwork* relates a person to the collection of persons interacting with that person by any communication channels, and *ChannelAvailability*

relates a device to a collection of communication channels supported by that device (e.g., e-mail, voice and SMS).

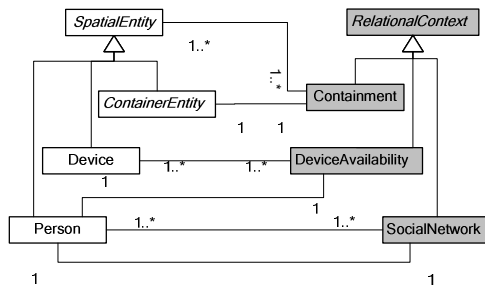


Figure 5 - Relational Context Types

Figure 5 depicts another example of relational context, the *Containment* context, which represents a direct containment relationship among spatial entities. More specifically, a *ContainerEntity* such as a building, a room or a vehicle may be associated with a containment relational context, which may in turn contain a set of spatial entities. A containment chain is created with the condition that every contained entity physically fits in its respective container entity.

Intuitively, relational context allows us to navigate the context model from an entity to the contexts of entities that are related through the relational context, still maintaining the separation of the concerns between entity and context. Consider the following example involving the entity types *Person*, *Device* and *Channel*. Let us suppose that John (of type *Person*) is related to his PDA and phone (of type *Device*) through *DeviceAvailability*. John's PDA is related to e-mail (of type *Channel*) through *ChannelAvailability*, and John's phone is related to a voice channel also through *ChannelAvailability*. Therefore, we can conclude that John is indirectly related to certain e-mail/voice channels.

We regard *RelationalContext* type as a relational moment universal in conceptual modeling. The relation that holds between bearers of a relational moment is called a *material relation*. For example, the relation that holds between devices and channels through *ChannelAvailability* is a material relation.

3.3 Formal Relations

Material relations are not the only means by which one can establish the relation between entities. Conceptual modeling theories also define the notion of *formal relation*. Formal relations hold between two individuals directly, without any intervenient individual. Examples of formal relations are: *greater*

than, *taller than*, *older than* and *subset of*. The immediate relata of such relations are qualities [10], i.e., formal relations are defined in terms of their relata qualities.

Nearness is an example of formal relation useful in context modeling. The truth value of an expression such as "John is near Maria" ("nearness" being defined, for example, as within 1 km range) only depends on the values of John's and Maria's locations, which are qualities (intrinsic context). Another example of formal relation is *distance* ($Distance(x,y,z)$), which can be thought of as a logical construction from the intrinsic context $a = location(x)$, $b = location(y)$, such that $z = |valueof(a)-valueof(b)|$ (Euclidian distance between a and b).

The distinction between material and formal relations are useful in our context models. On one hand, it is possible to derive or infer the truth value of a formal relation solely from the intrinsic context of entities related. On the other hand, direct inference from intrinsic context is not sufficient to determine whether a material relation holds.

4. Situations

Situations define particular states of affairs which are of interest to applications. They are composite concepts whose constituents are the elements of our ontology, i.e., entities, formal relations, intrinsic and relational contexts. Situations are genuine ontological elements that are composed by other elements. Examples of theories that define situations are [1, 6].

Examples of situations that might be of interest to context-aware applications are "John is running and he has access to his mobile phone", "John is in danger of an eminent epileptic seizure and he is driving", "John is near Alice and both have access to their mobile devices".

Extending the foundational concepts presented in Figure 1, situations are individuals, which are composed by other individuals. Figure 6 depicts situations as foundational concepts.

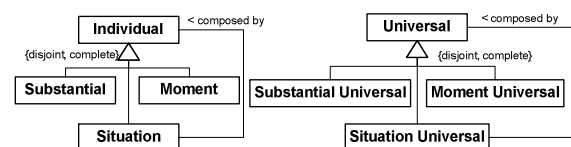


Figure 6 – Foundational concepts

In our approach we define *Situation Universals* (types), which aim at characterizing situations with similar properties. For example, the situation type

“John is within 50 meters from Alice” consists of all situation individuals in which the distance between John’s and Alice’s location values is less than 50 meters. Similarly, the situation type “Person is within 50 meters from another person” consists of all situation individuals in which the distance between any two persons’ location values is less than 50 meters.

A situation exhibits temporal properties, such as the time interval during which the situation holds. This aspect is in line with the ontological definition of situations discussed by [6], which defines a situation as a snapshot view of some part of the world. In this theory, a situation is *framed* by a *chronoid*. Chronoids are ontological entities that define a temporal duration. As an example, consider the situations “John is married to Alice” and “John and Alice are divorced”. From time t_0 to t_{10} (e.g., for 10 years) John has been married to Alice. During this interval, at any time (a snapshot), the situation “John is married to Alice” holds. We can say that the situation “John is married to Alice” is framed by a chronoid that refers to the time interval $[t_0, t_{10}]$. Suppose the situation “John and Alice are divorced” is framed by a chronoid defined by the interval $[t_i, t_f]$. Since the marriage situation is a pre-requisite for the divorce situation, and a couple cannot be married and divorced to each other at the same time, we can explicitly define that $t_i \geq t_{10}$.

The examples used throughout the paper illustrate a range of situation patterns that are relevant for context-aware applications. These patterns involve the different kinds of context (intrinsic and relational), entities, and formal relations, which are the building blocks used to compose situations.

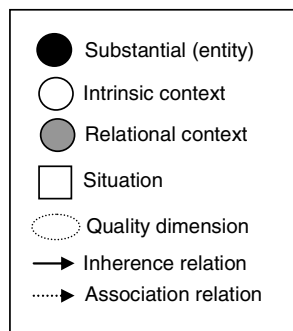


Figure 7 - Notation for situation modeling

We use a simple visual notation to represent situations (see Figure 7): (i) black circle: substantial (entity); (ii) hollow circle: intrinsic context; (iii) grey circle: relational context; (iv) rectangle: situation; (v) spatial inclusion in the plane: parthood; (vi) dashed ellipse: quality dimension; (vii) normal arrow: inherence relation (viii) dashed arrow: association

relation between a quality and its value in a quality dimension.

4.1 Situations involving Intrinsic Context

Situations involving intrinsic context are composed by a unique entity and part of its intrinsic context. An example of such a situation is “John has influenza and he is tired”. Figure 8 depicts this situation using the notation we have just described.

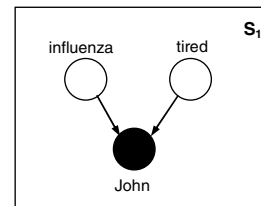


Figure 8 – John with fever and influenza

Other examples of intrinsic situations are “John has temperature 40 degrees” and “a laptop x has battery power at 10%”, which are S_2 and S_3 depicted in Figure 9.

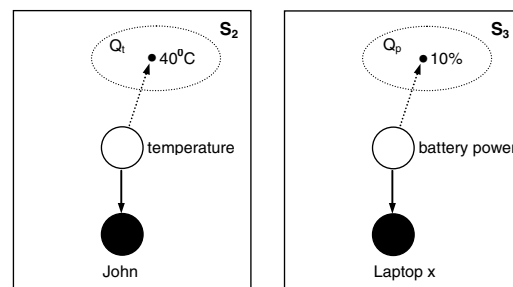


Figure 9 - Examples of situation involving intrinsic context

Q_t and Q_p represent the temperature and percentage quality dimensions, respectively. The points 40°C and 10% are specific values (quaes) of these quality dimensions, as discussed in Section 3.1.

We can define a situation type “John has fever”, which characterizes all situation individuals in which John’s temperature is above 38 degrees (of which S_2 is an instance). Similarly, we can define a situation type “Person has fever” which characterizes all situation individuals of persons (not only John) whose temperature is above 38 degrees.

The situation type “Laptop x has low battery power” can be defined such that it consists of all situation individuals in which Laptop x’s battery power is lower than 20% (of which S_3 is an instance). Analogously, a situation type that applies to any arbitrary laptop can be defined.

4.2 Situations involving Relational Context

Situation involving relational context are constituted by at least two entities and part of their relational contexts. Several examples can be mentioned in this category, as follows.

Figure 10 defines the situation S_4 in which John, Alice and Mary are connected to each other through a social network, such as a friendship. This friendship (of type relational context) inheres in John, Alice and Mary.

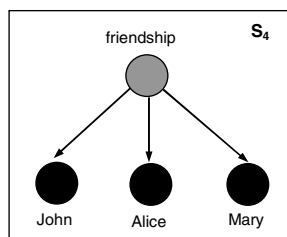


Figure 10 - Friendship situation

Other examples of situations with similar configuration as S_4 include: (i) the case that several people participate in a meeting t . In this case, the relational context of type Meeting inheres in all participants of the meeting; (ii) the case that several medical units participate in the treatment of a patient x . In this case, there is a unique relational context of type Treatment that inheres in the patient plus all participating medical units; (iii) John being married to Mary; (iv) companies a , b , c having a mutual contract (instance of a relational context type) to work on a project.

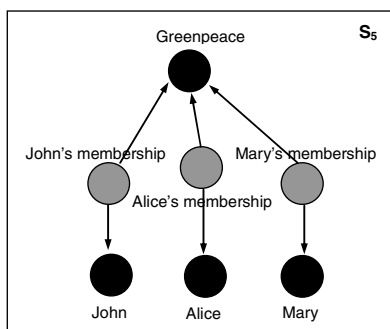


Figure 11 - Membership situation

Figure 11 represents situation S_5 in which John, Alice and Mary are members of a non-governmental organization, such as Greenpeace. The relational context type that connects John, Alice and Mary is a participation relational context, such as membership.

Figure 12 represents situation S_6 in which Laptop x has established a connection (of type relational context)

to each of the three networks, WLAN, Bluetooth and Infrared (entities). By explicitly modeling the connections as relational context, we are able to assign properties to these connections, such as access rights and negotiated QoS.

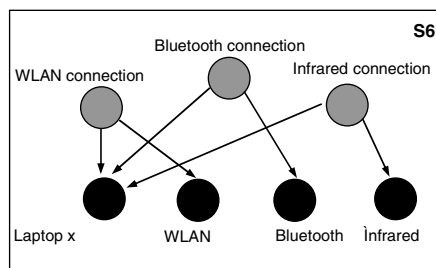


Figure 12 - Network connection situation

Figure 13 represents situation S_7 in which Laptop x has the potential of connecting to WLAN, Bluetooth and Infrared networks. Differently from S_6 , this situation does not imply that laptop x is connected to these networks.

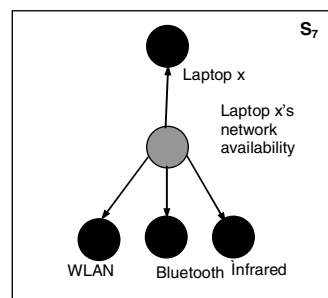


Figure 13 - Network availability situation

Figure 14 represents situation S_8 in which John and Alice are married (through a Marriage relational context), while Alice works for the University of Twente (through a work contract relational context).

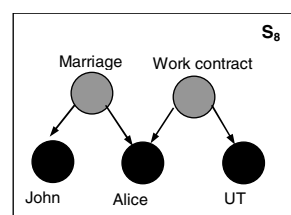


Figure 14 - Marriage and working contract situation

Figure 15 represents situation S_9 in which both John and Alice have access to their mobile phones.

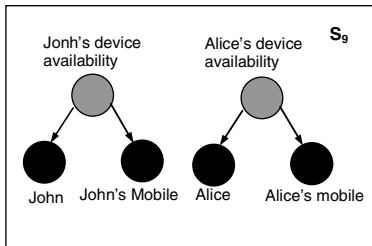


Figure 15 - Device availability situation

4.3 Situations involving Formal Relations

Situations involving Formal Relations consist of at least two entities and at least two or more qualities such that these qualities are comparable. Two qualities are comparable if they are associated to the same quality dimension. An example of such a situation is depicted in Figure 16, where Q_L represents the location quality dimension.

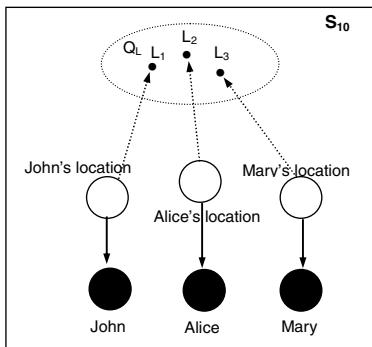


Figure 16 – John, Alice and Mary's locations

In situation S_{10} , the value of John's location is L_1 , the value of Alice's location is L_2 and the value of Mary's location is L_3 .

We can define a situation type "John, Alice and Mary are nearby each other", which consists of all situations in which L_1 , L_2 and L_3 are within 50 meters radius. Similarly, we can define a situation type "Persons are nearby each other" which consists of all situations in which persons within 50 meters radius.

We can also define a situation type "John is closer to Alice than to Mary", which defines all the situations in which the distance between John's and Alice's locations is greater than the distance between John's and Mary's locations. If $|L_1-L_2| > |L_1-L_3|$, situation S_{11} is an instance of type "John is closer to Alice than to Mary".

Situation S_{11} (Figure 17) represents R_1 and R_2 , which are the regions in space occupied by John and the Zilverling building, respectively. Q_R represents the spatial region quality dimension.

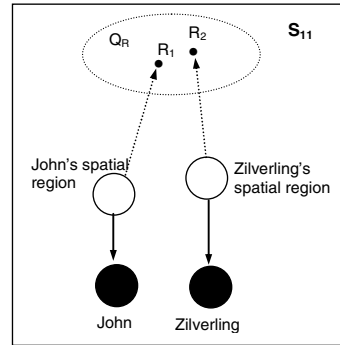


Figure 17 – Containment situation

We can define the situation type "John is contained in the Zilverling building", which consists of all situations in which John's spatial region is within the Zilverling's spatial region. If R_1 is within R_2 , S_{11} is an instance of type "John is contained in the Zilverling building".

Further, we can define a containment situation type between any two spatial entities. This situation type consists of all situations in which an entity's spatial region is within another entity's spatial region.

4.4 Combined Situations

We call combined situations the situations that combine the context types we have defined. Figure 18 represents situation S_{12} in which Alice is being interviewed by John, and Alice has access to her mobile phone (through device availability relational context) and her mobile phone is on. Further, this situation also represents the values of John's, Alice's and Alice's mobile phone locations (L_1 , L_2 and L_3 , respectively).

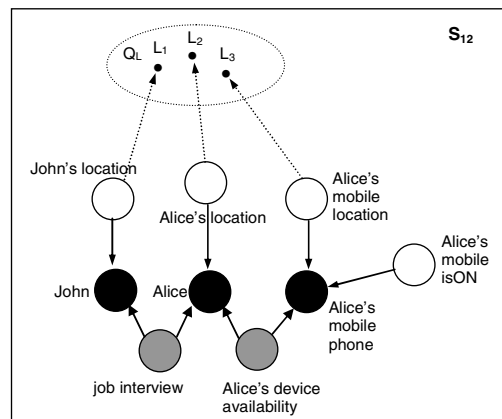


Figure 18 - Example of hybrid situation

Figure 19 represents situation S_{13} in which Laptop x is connected to a WLAN network (through WLAN connection relational context), and laptop x's battery power is 40% and WLAN's bandwidth is 2Mbps.

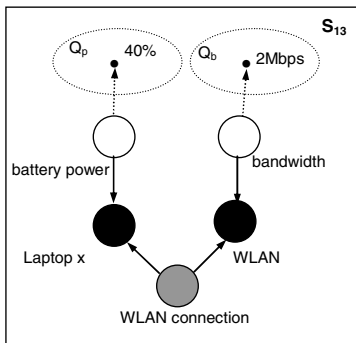


Figure 19 – Example of hybrid situation

4.5 Situations of Situations

Situations themselves exhibit properties (bear moments). Consider the situation type “John is at location L_1 while Alice is at location L_2 , for more than 10 minutes”, which consists of all situations in which John and Alice are located at L_1 and L_2 , respectively, with the duration of more than 10 min, e.g., 11 min, 12 min, 20 min, etc.

Figure 20 depicts a situation of this type (S_{15}), since S_{14} has a duration intrinsic property of 15 minutes.

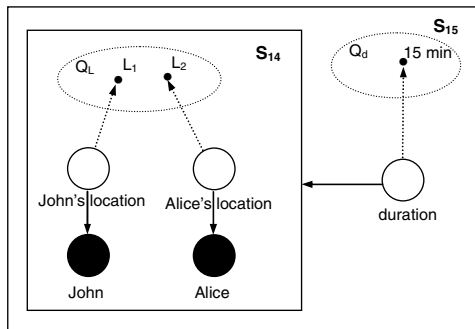


Figure 20 – Example of situation of situation

Consider the example situation in which “John has been married to Alice for 10 years”. This situation is represented by S_{17} in Figure 21.

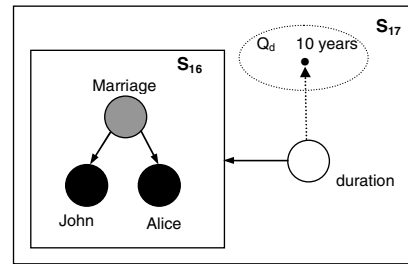


Figure 21 - John and Alice married for 10 years

We might also need to represent temporal relations between situations. For example, we might want to know when laptop x switches from a WLAN connection to a Bluetooth connection in order to set new quality of service parameters.

Figure 22 shows that this situation can be modeled by defining a situation (S_{20}) which is composed of situations (S_{18} and S_{19}) with particular temporal properties.

Situations S_{18} and S_{19} represent the situations “laptop x is connected to WLAN” and “laptop x is connected to Bluetooth”, respectively. Both S_{18} and S_{19} have starting time and final time, which are intrinsic properties of these situations.

S_{20} represents the situation in which initial and final times of S_{18} are T_{18i} and T_{18f} , respectively; and initial and final times of S_{19} are T_{19i} and T_{19f} , respectively. If $T_{19i} > T_{18f}$, and the duration of handover is less than 1 minute, i.e., $T_{19i} - T_{18f} < 1$ minute we say that S_{20} is an instance of the situation type “laptop x switches from WLAN to Bluetooth”.

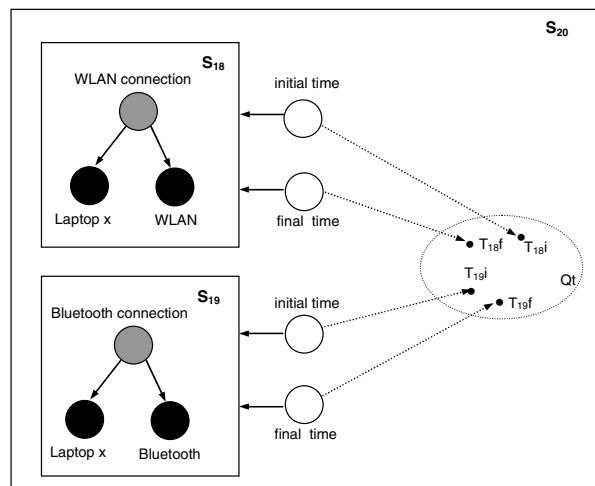


Figure 22 – Example of situation of situations

5. Related Work

Most approaches towards context modeling presented in the literature (e.g., [7, 13, 14]) do not explore the benefits of conceptual modeling as the first phase in the design trajectory. Often these approaches consider technological issues already in the beginning of the design process, giving precedence to computational issues over human understandability. In addition, these approaches do not consider ontologically well-founded theories to support their modeling choices.

The work presented in [7] discusses a situation based theory for context-awareness that allows situations to be defined in terms of basic fact types. Fact types are defined in a ORM (Object-Role Modeling) context model, and situation types are defined using a variant of predicate logic. Since this approach does not discuss a suitable notion of time, temporal aspects, such as duration and precedence of situations, cannot be explicitly defined.

Our approach also differs from that of Barwise [15, 16]. Barwise did not elaborate an ontology of relations (relational moments and formal relations). There are no corresponding elements in his theory to substantial and moments, making it difficult to draw a parallel with context-awareness, since contexts are moments and entities are substantials. In addition, Barwise uses *abstract* situations in order to analyze, describe and classify real situations. Abstract situations are set-theoretical constructions which can capture only limited aspects of the ontology of real situations [6].

6. Conclusions

We have presented in this paper our current efforts towards conceptual modeling of context. We have proposed before an approach that considers as a basic distinction, the separation of the concepts of *entity* and *context*. Further, we characterize context as either *intrinsic* or *relational*. We believe that conceptual modeling of context should precede the detailed design of context-aware applications, in a similar way as analysis should precede detailed design of an information system.

Since conceptual modeling focuses on supporting structuring and inferential facilities that are psychologically grounded [9], the adequacy of our context modeling technique rests on how it contributes to common understanding of context among the stakeholders of a context-aware application (e.g., users and designers). Therefore, we have justified our modeling choices with results from foundational

ontologies [4, 6], which are in line with conceptual theories in philosophy and cognitive sciences.

We have extended our models with the ontological concept of Situation. Situations are composed of individuals that are instances of entity and context universals, and can be considered as elements of interest on their own. This allows us to identify and discriminate properties of situations such as those we have exemplified throughout this paper. By taking this approach we are able to model relevant changes in the state of affairs of a context-aware application's universe of discourse. We believe this is the key to include temporal aspects of context in a comprehensive ontology for context modeling, since situations can be related to suitable notions of time using the concept of *chronoid*. As future work, we will further explore temporal aspects for the modeling of context including *events* and their (temporal ordering and causal) relations. We anticipate that the notion of event can be used to capture transitions between situations.

Situations have been represented using an ad-hoc graphical notation. It has not been our intention to provide a comprehensive notation, but to use this notation to convey the examples intuitively. As part of our future work, we intend to investigate suitable languages to specify situations and situation types.

We would also like to provide support for bridging the gap between conceptual context models such as those proposed here and *context information models*. In the scope of context information models, we should refer to *context information* as opposed to *context*. Context information refers to the representation of (constituents of) context in an application, such that this representation can be manipulated and exchanged. Situations are in this case, detected based on context information. Issues that become relevant for context information models relate to: (i) how context is sensed; (ii) how context information is produced, learned, inferred and used, and (iii) the validity and quality of context information.

7. Acknowledgements

This work is part of the Freeband AWARENESS and A-MUSE projects (<http://awareness.freeband.nl> and <http://amuse.freeband.nl>). Freeband is sponsored by the Dutch government under contract BSIK 03025.

8. References

- [1] K. Devlin, *Logic and Information*, Cambridge University Press, 1995.

- [2] P. Dockhorn Costa, J.P.A. Almeida, L. Ferreira Pires, G. Guizzardi, M. van Sinderen, "Towards Conceptual Foundations for Context-Aware Applications", *Proc. of the Third International Workshop on Modeling and Retrieval of Context (MRC06)*, Boston, USA.
- [3] P. Dockhorn Costa, L. Ferreira Pires, M. van Sinderen, "Designing a Configurable Services Platform for Mobile Context-Aware Applications", *Int'l Journal of Pervasive Computing and Communications (JPCC)*, 1(1):13-25, Troubador Publishing, 2005.
- [4] G. Guizzardi, H. Herre, G. Wagner, "On the General Ontological Foundations of Conceptual Modeling", *Proc. of the 21st Int'l Conference on Conceptual Modeling (ER-2002)*, LNCS 2503, 2002.
- [5] G. Guizzardi, *Ontological Foundations for Structural Conceptual Models*, PhD Thesis, University of Twente, The Netherlands. TI-FRS No. 15, 2005.
- [6] B. Heller, H. Herre, *Ontological Categories in GOL*, *Axiomathes* 14:71-90 Kluwer Academic Publishers, 2004.
- [7] K. Henriksen, J. Indulska, "A software engineering framework for context-aware pervasive computing", *Proc. of the 2nd IEEE Conf. on Pervasive Computing and Communications (Percom2004)*, USA, IEEE CS Press, 2004.
- [8] Merriam-Webster, Inc., Merriam-Webster Online; <http://m-w.com>
- [9] J. Mylopoulos, "Conceptual modeling and Telos", P. Loucopoulos and R. Zicari, eds, *Conceptual modeling, databases, and CASE*, chap. 2, pages 49-68. Wiley.
- [10] K. Mulligan, B. Smith, A Relational theory of the Act. *Topoi* (5/2), 115-30, 1986.
- [11] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, A. Oltramari, *Ontology Library*, WonderWeb D18, 2003.
- [12] J.F. Sowa, *Conceptual Structures: Information Processing in Mind and Machine*, Addison-Wesley, NY, 1984.
- [13] H. Chen, T. Finin, A. Joshi, "An ontology for context-aware pervasive computing environments", *Knowledge Engineering Review*, Special Issue on Ontologies for Distributed System, 2003.
- [14] X. Hang Wang, D. Qing Zhang, T. Gu, H. Keng Pung, "Ontology-Based Context Modeling and Reasoning Using OWL", *Proc. of the 2nd IEEE Conf. on Pervasive Computing and Communications Workshop (PERCOMW04)*, USA, 2004.
- [15] J. Barwise, J. Perry, *Situations and Attitudes*, Cambridge, MA: The MIT Press, 1984.
- [16] J. Barwise, *Situations in Logic*, CSLI No. 17, 1989.