# Role of the user in information systems development

# Role of the User in Information Systems Development

H. Jaakkola\*, B. Thalheim \*\*, J. Henno\*\*\*, J. Mäkelä\*\*\*\* and H.Keto\*

\* Tampere University of Technology/Pori, Finland
\*\* Christian Albrechts University at Kiel, Kiel, Germany;
\*\*\* Tallinn University of Technology, Tallinn, Estonia;
\*\*\*\* University of Lapland, Rovaniemi, Finland;
hannu.jaakkola@iki.fi; thalheim@is.informatik.uni-kiel.de; jaak.henno@ttu.ee; jukka.makela@ulapland.fi;
harri.keto@tut.fi

*Abstract* - **We have focused our paper on the aspects important in adapting an Information System (IS) to the user's cultural background. We are interested both in the factors related to IS development and in the use of IS. Increasingly, ISs are being developed and used in a global context. We have perceived differences in expectations of functionalities, architecture, structural properties, information search practices, web-based system properties, and user interfaces. One conclusion would be that a high quality IS reflects user behavior in its use context. In that case, the system has to model its user one way or another. Until now, the topic has been handled without meaningful effort to model user behavior. Current publications cover a wide variety of rules on how to take into account cultural differences in the IS context. In this paper, our aim is to study the current state-of-the-art of user modeling – modeling the human being as an IS user. We start with general aspects related to the role of the user in IS development and alternatives to adaptable systems. The findings are applicable in the educational context as well. More and more, the use of computers and ISs is becoming an essential part of studies: the use of MOOCs (Massively Open Online Courses) as a part or replacement for traditional face-to-face classes; flipped learning methodology emphasizing the significance of self-learning; and blended learning, including quite often computerized study content. Our focus is on the global context, in which students represent different cultures and the IS is globally available.**

*Keywords – user; information system; adaption; context; information systems development; requirements engineering; user modeling; user adaptable; human behavior modeling.*

## I.    INTRODUCTION

The role of the user in information system (IS) development is twofold: First, user expectations are embedded in the system's structure and functionality during the development process and secondly, an IS has to be built that is able to adapt to the needs of different users during its use. In addition to the *users,* there is a wide variety of other stakeholders (client, customer, management, government, legal, etc.), whose wishes and requirements must be taken into account in IS development. In this paper, to simplify the text, we will use the term "user" to cover all of the above, because ultimately an IS is developed for users to meet their needs as a part of real business processes.

In our paper [1] we have pointed out that increasingly, an IS is developed for "faceless users" by "faceless developers". Developers and users do not know each other, and the development work is done by globally distributed teams. As a result, the traditional (face-to-face) way to execute development tasks is not possible, nor is traditional (face-to-face) teamwork. Problems in culture-sensitive parts of the development life cycle may appear, as well as in the culture-sensitive features of the final artifact (software). *Formulation of concepts* is also culture-sensitive: people in different cultures have a different approach to handling abstract structures of concepts and concepts may have different semantics. Therefore, the same conceptual model does not meet the expectations of all interest groups. Paper [2] handles this topic from the point of view of the database schema.

In this paper, we first discuss the two above-mentioned approaches to taking user expectations in IS into account and answer the following two questions:

1. How to embed user expectations into the system's structure and functionality during the development process (Section 2);

2. How to build such flexibility in the system that it adapts to the needs of different users, especially taking into account their cultural differences (Section 3).

Although the approach is general, the discussion also answers the question "What must be taken into consideration in the acquisition of educational software and what kind of properties must it have?"

This leads us to the central *research problem* of this paper: *How to model the user of an IS, both in general, but especially in a multicultural context.*

The role of user modeling in IS is a culmination of Sections 2 and 3. In Section 4 we introduce briefly how to analyze cultural differences. Section 5 covers a review of alternative user modeling approaches. This is based on a light mapping study (heuristic, non-systematic) without

aspiring to the perfection. Section 6 summarizes the paper. 2018.

## II. THE USER AND THE DEVELOPMENT PROCESS

Several standards and process manuals handle software processes having high capability and maturity. The series of ISO 15504 standards [3] and CMMI [4] represent models of the *plan (process) oriented* approach in software development. The role of interest groups is clearly defined as a part of process specifications. Nowadays, *agile development* (see the principles of e.g., the SCRUM development method in [5]) has been widely adopted, even in the development of large ISs. The main principles of SCRUM cover the user joining the software development team as an essential member, partitioning the system to allow incremental / iterative development, and immediate integration of the increments into the system in a sequence of releases.

Independently of the development approach (plan-driven / agile), the system (including every increment) is developed in a sequence of development phases (Figure 1). The aim of the first development steps is to elicit and analyze user expectations carefully to build the foundation for the IS.
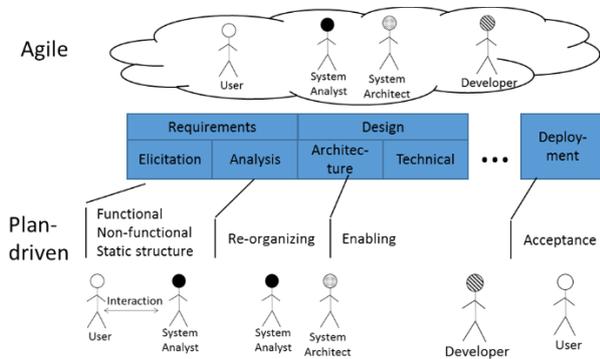


Figure 1.   The user's role in the IS development "life cycle"

The *requirements baseline* covers functional and non-functional requirements and the static structure of the system in a well-organized and analyzed form, as agreed in the interaction between the system analyst and the key interest groups. *Functional requirements* define the functionality (operations) of the system. *Non-functional* system properties are *quality aspects* that provide guidelines, restrictions, and presumptions for the development work; one of these is *adaptability* in the heterogeneous set of unknown users. The *static structure* of the system (in the requirements phase) indicates the *conceptual structure* of the system as seen by the key interest groups, i.e., the *user's internal conceptual model* of the system as documented by the system analysts. All this points out the role of system analyst as an *interpreter, organizer, digger, and bookkeeper* of the requirements identified, and the role of the user as a master and owner of the system.

The role of design is to transfer the requirements baseline to *system design*. Architectural design *embeds* (a part of) the *non-functional features* into the system functionality and structure; some of these features may also

indicate requirements (e.g., traceability) for the development process.

In the case of adaptation requirements, *architecture design is an enabler* that builds adaptability into the system structure and functionality. In practice, this means features that allow the system to modify itself in different use cases, its learning capability, its capability to recognize its users, etc. From the technical point of view, it is a question of following the principles of suitable *architectural styles* [6] and the use of suitable *design patterns* [7]. These provide the means for the loose binding and changeability of system components even in the runtime, in addition to the flexibility built in the system's static structure. One of the technical solutions is to embed a *user model* into the system; this is the focus of the rest of our paper.

## III. THE ADAPTATION CAPABILITY OF THE SYSTEM

There are several opportunities to include user-based adaptability in the IS. We separate three alternative approaches to implementing adaptability:

1. Systems that are adapted by the user to fulfill their expectations (*user adaptable systems*);

2. Systems that recognize the users and adapt their behavior (*user recognition systems*);

3. Intelligent (dynamic, learning) systems that recognize the use context and the user (*context recognition system*).

The classification is analogous to the classification of Fisher [8], which separates two main approaches – adaptable and adaptive. The two latter of our approaches /Approaches 2 and 3 belong to the category of adaptive systems (systems adapt themselves dynamically).
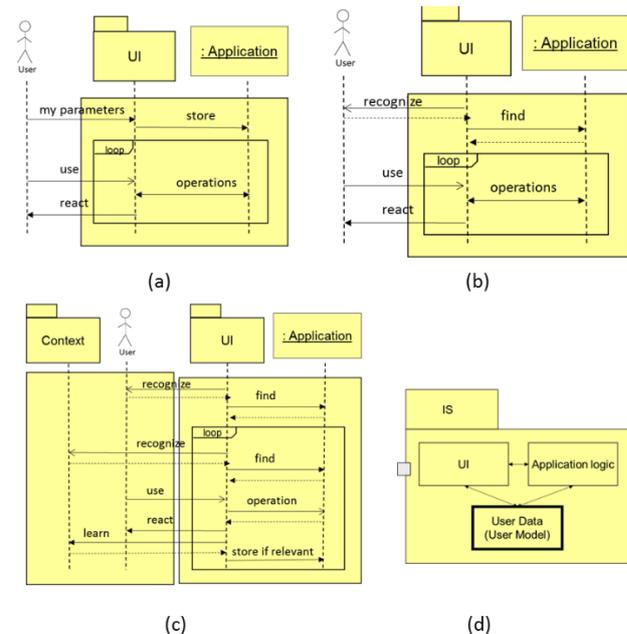


Figure 2.   Adaptive ISs – three basic models and IS structure

In all three cases, the system must be able to store *user related data*, to manipulate it (learning capability), and to use it to modify the system's runtime behavior according to

the expectations of an individual user. Figure 2 illustrates the rough structure of the three solutions and its user interaction in the form of a UML sequence and package diagram.

The simplest form of adaptive system – the *user adaptable system* - is based on parameterization. The system provides an interface in which the user defines values for the parameters. The values are interpreted by the system, which modifies its behavior according to the parameter values. This approach is static: the modification opportunities are limited to predefined properties only, and the alternative parameter values must be predefined (Figure 2a). In this case, the user interaction follows the same procedure during the whole use session and the *user model is simply the data structure*.

*The user recognition system* (Figure 2b) recognizes its user automatically. Alternatives for implementing user recognition vary from hidden (cookie type) to active (login). In this case, the user parameters are stored in the system in advance – either stored by the user (as above) or recognized by the system automatically. These may also be changeable (learning capability), but the value space is predefined; as a result, the adaptability extent is static as above. In this case the *user model is represented by the data structure and related functionality*.

*Context recognition systems* are dynamic: in addition to recognizing its users, the system also takes into account the (dynamic) use context (Figure 2c). Therefore, the system must have access to context-related data to implement the adaptability. Because the context changes during or between the use sessions, the system needs access to the context constantly. A good example of context is the location data in mobile applications, in which the context is based on the sensor values of the mobile terminal. Context data may be distributed – either stored in the client application or the server side. For example, some of the Huawei mobile phones have an application processor (currently Kirin 970) that implements context analysis in real time. In this case, *the user model has a distributed (or not) data structure*, whose contents are changed according to the changes in the use context, and applied in changing IS behavior.

Figure 2d illustrates the IS structure as a UML package diagram. The system consists of the user interface (UI), application logic (and the data repositories needed by it), and *user data*, which we call the *User Model*. In the simplest case it is only a data table in the application, whereas in more complex cases it may be a *dynamic structure* representing the *IS's understanding of the dynamics of the user*. The rest of this paper focuses on this topic. From the software structure point of view, we would like to point out the importance of separating the application logic from the user model; it provides high level flexibility in the IS architecture via software interfaces applying design patterns, see e.g. [7] as an integration solution.

IV.    RECOGNIZING CULTURAL DIFFERENCES

There is a wide variety of models for analyzing culture-based differences in human behavior and expectations. In our studies, we have applied the models of Hofstede [9], Lewis [10], and Hall [11; 12] as the main sources.

The model of Hofstede is based on the analysis of six cultural dimensions [10]:

- Power Distance (PDI): the extent to which power differences are accepted.

- Individualism / Collectivism (IDV): the extent to which a society emphasizes the individual or the group.

- Masculinity / Femininity (MAS): refers to the general values in the society - hard/ soft values.

- Uncertainty avoidance (UAI): refers to the extent that individuals in a culture are comfortable (or uncomfortable) with unstructured situations.

- Long-term / Short term orientation (LTO): refers to the extent to which the delayed gratification of material, social, and emotional needs are accepted.

- Indulgence / Restraint (IVR): acceptance of enjoying life and having fun vs. controlling life through strict social norms.

Cultures have different index (dimension) values that define their cultural stereotype. The Trompenaars model [13] follows the structure of Hofstede, with partially different dimensions in analysis.

*Lewis* divides cultures into three basic stereotypes: *linear-active, multi-active, and reactive*. Some of the national cultures are representatives of these three basic types, others lie somewhere between (in a triangular structure). Quite interesting, especially from the multicultural communication point of view, is the work of *Hall* [11; 12]. His model divides national cultures according to *context dependency in communication;* he divides cultures into high- and low-context cultures according to the level of "hidden information" and exactness of messages.

A brief overview of the above-mentioned models is included in a previous paper of ours [14]. There, we have also *combined Hofstede's cultural stereotypes, Lewis' communication-related aspects,* and used *Hall's high/low context classification* in one multidimensional model. In another paper [15], we have handled the concept of *cultural distance* as a mathematical multidimensional structure to be able to explain the difficulty level in collaboration between different cultures (a longer distance indicates the need for more careful focus on organizing multicultural collaboration).

In her studies (see e.g. [16]), Reinecke has clearly contributed to the idea of human modeling in IS design. She has applied Hofstede's dimensions in implementing a test tool for measuring cultural differences in the use of IS. The MOCCA tool has close to 40,000 user interface variations available for different user preferences, especially based on national cultures. The idea of MOCCA is to find a *good match between the user model included in the IS and the mental structure of the real IS user*. This idea is easy to generalize even outside the original scope (cultural

differences): a good quality IS models the behavior of its user; the better the fit between the user model built into the IS and the user's real mental model in the use case, the higher the user's satisfaction.

## V. USER MODELING IN INFORMATION SYSTEMS

### User model

In Section 3 we defined the concept of the user model as a component of IS having the following features:

- It includes the data storage that is able to store data values representing the properties of an individual user;

- It supports dynamics (changes in the data values) based on the user's behavior and the use context (adaptability in new situations, learning capability, etc.);

- There may be dependencies between the data fields of the model that are based on functional interaction between the model's data elements.

The two first properties are implementable as a data structure and its interaction with application logic. The third property requires dynamics that are implementable, e.g., in the form of a class (conceptual, object) model defining dependencies and functionalities between the elements; some of these may be external to the system.

We implemented a light mapping study [17] of the existing literature. The aim was to deepen our insight of the topic, especially from the point of view of our research problem as specified in Section 1. The search terms used were "*modeling human behavior in IS*" (2.4 M findings in Google Scholar) and "*modeling of human behavior in IS development*" (2 M findings) to cover both of the aspects discussed in Section 2. The findings cover a wide variety of contexts, e.g., crisis management, security-related issues, urban development, simulation of a human being, agent-based implementations, human behavior in certain contexts, artificial intelligence related approaches, etc., which we excluded. For more detailed analysis we selected some of the most interesting studies, which met our *objective* goals, especially the coverage of cultural differences.

### Reinecke's user model

Reinecke (and Bernstein, [16]) has built a test tool, MOCCA, that is used for analyzing the user's national culture based preferences in IS usage. The model has been published as a class diagram that applies Hofstede's dimensions as the key parameters of the user (Person); see Figure 3.

The model takes into account the user's adaptation based on the influence of foreign cultures ("hasFormerResidence"; leads to the definition of the concept "*adapted national culture*") and implements user dynamics as the functional connections between diagram classes. The model also takes into account the nationality of the user's parents (#hasNationalityMother, #hasNationalityFather).
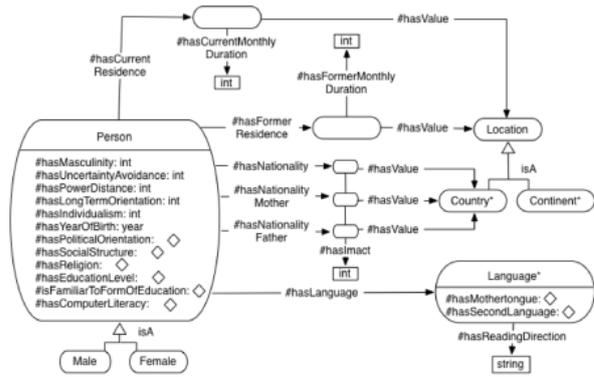


Figure 3.    User model by Reinecke [16]

With reference to Figure 2d in this paper, the class diagram represents its "User Model" component.

### Phaedra and Permanand – a student model

Phaedra and Permanand [18] have introduced a student model that takes into account a student's demographic factor value (Figure 4). The class diagram (student model) describes the student's cultural and demographic properties.
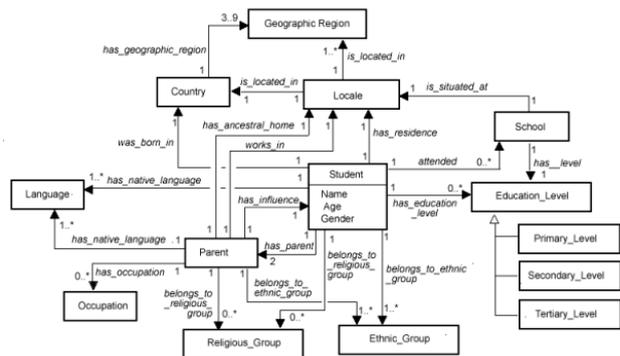


Figure 4.    Student model by Phaedra and Permanand [18]

The model (contextual student model, cultural profile) quantifies the student's expression of socio-cultural group traits and preferences. It was used to indicate the expectations and attitude of students in the field of culturally aware tutoring systems. The key factors cover Locale (residence, country, place of birth), School (and educational level in it), Ethnic and Religious Group, Parent (Locale, Occupation, Language), and student's native language. The class diagram represents the "User Model" component in Figure 2d.

### Dafoulas and Macaulay – cultural differences in virtual software teams

Dafoulas and Macaulay [19] have modeled the dynamics of multicultural virtual software development teams (Figure 5, upper part). The model lists important factors to take into account in managing the team and organizing its work. The model is multidimensional. It recognizes five dimensions of culture that affect the behavior of an individual: national, organizational, professional, functional (tasks), and team (as a subculture inside an organization). The authors point out the

importance of the professional culture. Software professionals worldwide belong to the computer subculture, which is stronger than any other culture. For example, "a Russian programmer would be more similar to an American peer than to a Russian marketing manager." In addition to the cultural frameworks mentioned above (Hofstede, Lewis, Hall, Trompenaars), this framework refers to the work of Fukuyama (trust as a factor, [20]) and Lessem & Neubauer [21]. The authors also refer to numerous other theories and surveys included in the model (not handled in detail in the original source).
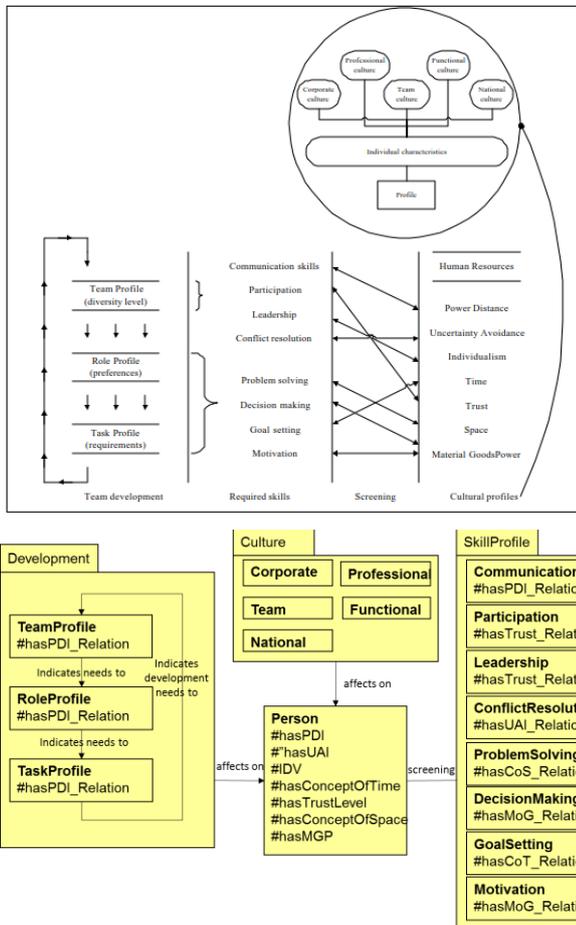


Figure 5.    Framework for using cultural model dimensions in virtual software teams [19; Authors]

The model maps the requirements for team development and required skills to the cultural profile. *Team development* is an iterative process, in which the team (diversity) profile indicates the requirements for the role (preferences) profile and further to the task (requirements) profile. This continuous improvement focuses on the required skills, in which the recognition of cultural profiles plays a central role. This model belongs to the category of "management and leadership models." Unlike previous models it does not represent a conceptual model of a user. For this paper we performed a small "exercise" and transferred the original model to follow the same presentation as the earlier ones (Figure 5 lower). In this form it represents the user model component in Figure 2d.

### Blanchard et al. – intercultural communication conceptual model

The aim of a *conceptual model* is to define the relevant concepts of the target and their relations to each other. One approach is to use the diagram representation (as in Figures 3 and 4), or a kind of dependency diagram (as in the original part of Figure 5). If we are interested only in the relevant concepts and their structural dependencies, a *mind map* is very clear, simple, and widely used for such purposes. Blanchard et al. [22] have introduced a conceptual model of intercultural communication, as illustrated in Figure 6.

Based on a literature review, Blanchard et al. [22] found theoretical evidence that the way people interpret and react to their environment is significantly dependent on their culture. In human-related technologies the Western context dominates; all trials and tests are done in the Western context, independently of the final target. They introduce a simplified *conceptual model of intercultural communication* using the mind map as a presentation tool. The model includes the following elements:

- *Cultural elements*: Basic cultural units of information. It consists of *cognitive cultural elements*, and *cultural non-verbal communication* (communication system shared by a cultural group).

- *Cultural body language (CBLA)*: Behavioral elements associated with meanings: this association results from a sociocultural perspective (i.e., not learned). Some of these are abstract and some concrete.

- *Non-cultural (innate) elements* cover behavioral primitive elements (gestures, postures, facial expressions) and some innate non-verbal communication elements.

- Additional concept classes in the model include the role of context, culture, and cultural group, enculturated individual aspects, cultural group cohesion, and a variety of descriptors (manifestation of a culture in practical situations).

Transformation of this conceptual model to an IS user model is not straightforward, but it is possible. It includes only the required concept structure: the dynamics (functional connections, learning from external sources) are not specified. However, it gives an excellent list of the relevant aspects to take into account in user modeling. Because of space limitations we have not dealt with the model details, which can be found in the original source. We feel that understanding the basic idea is possible even without the details.
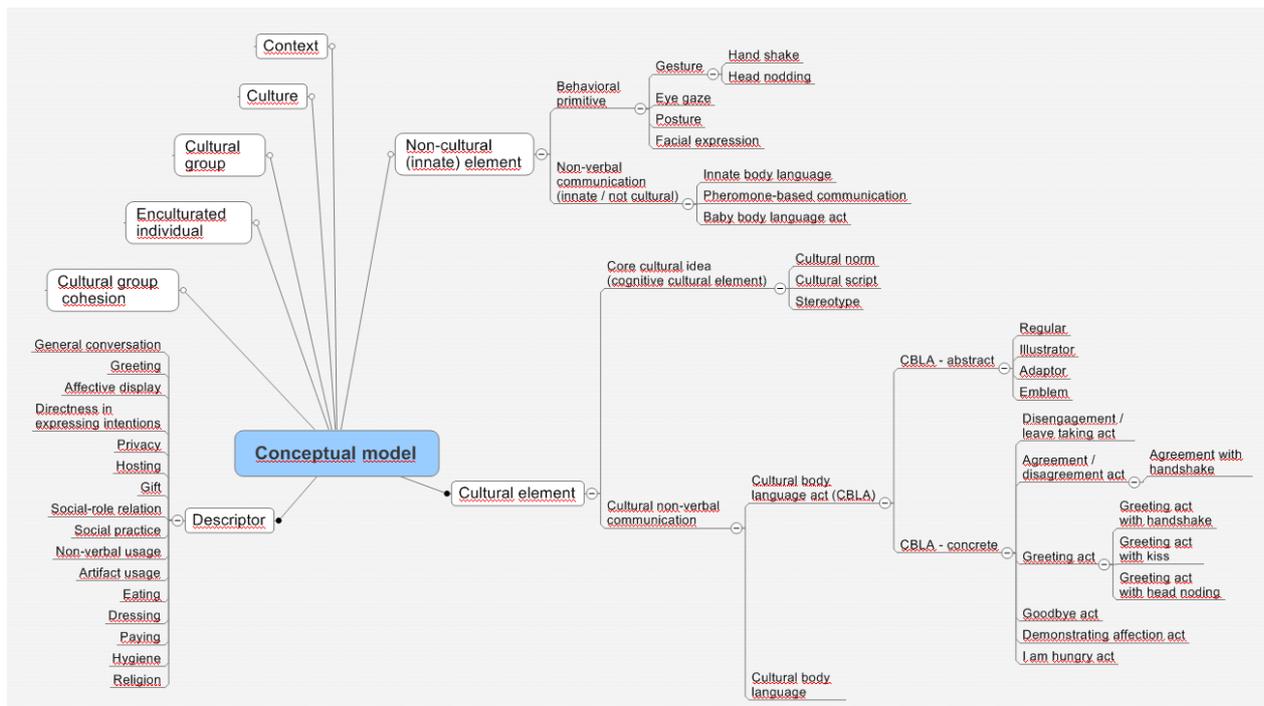
Figure 6.     Intercultural communication conceptual model

**Other findings**

In addition to the few papers introducing a real user model, the literature review resulted in a lot of interesting findings indicating how to take the user's role into account in IS development and in its implementation.

The paper of Damaševičius [23] discusses the organizational, social, cultural, and psychological aspects in the software development process, and reflects the findings in software product properties. The paper introduces a taxonomy of the other disciplines that it is relevant to take into account in software engineering.

Dugdale [24] has published a report that focuses on human behavior modeling in complex socio-technical systems. The work combines *cognitive engineering* and *multi-agent technology* in modeling human behavior (especially human *interaction*). The report itself is a summary of long-term (14 years) research work, with *crisis and emergency management* as its application area. In this field, context- and culture-independent communication is needed to provide means for collaboration between heterogeneous actors.

The paper of Fisher [8] takes a similar starting point as we have in our paper: "*Designers of collaborative human-computer systems face the formidable task of writing software for millions of users (at design time) while making it work as if it were designed for each individual user (only known at use time.*") He approaches the problem from the user modeling point of view (as we do). He distinguishes two basic approaches to make IS adapt to the user's needs: *adaptable systems* are changed by users and under their

control (passive system) and *adaptive systems* are active, based on the dynamic adaptation of the system itself to take into account the user and the environment. The models introduced remain abstract, focusing on human-computer interaction principles (general structure charts), without touching on the user model. The paper of Johnstone and Tate [25] approaches the same interaction problem from the *information contents* point of view; the models introduced are also structural system models without an explicit user model.

From the perspective of psychology and behavioral sciences, the topic is dealt with in several papers. We examined two papers (by Huitt [26] and by Johansen et al. [27]) superficially without finding any exact contribution to this paper. However, this approach would be worthy of more careful examination in the future, especially in building the user's mental structure as a part of the user model. In this case the model already approaches an *artificial intelligence* based solution having elements of (deep) learning and heuristics, which is an approach that we have excluded now but found interesting for future studies.

## VI.   CONCLUSIONS

To conclude this paper we introduce our previously published contribution [2] to the discussion as a starting point for this paper. It includes a simple user conceptual model as a mind map presentation (Figure 7). This model indicates the important factors of an individual to be taken into account in developing an adaptive IS.
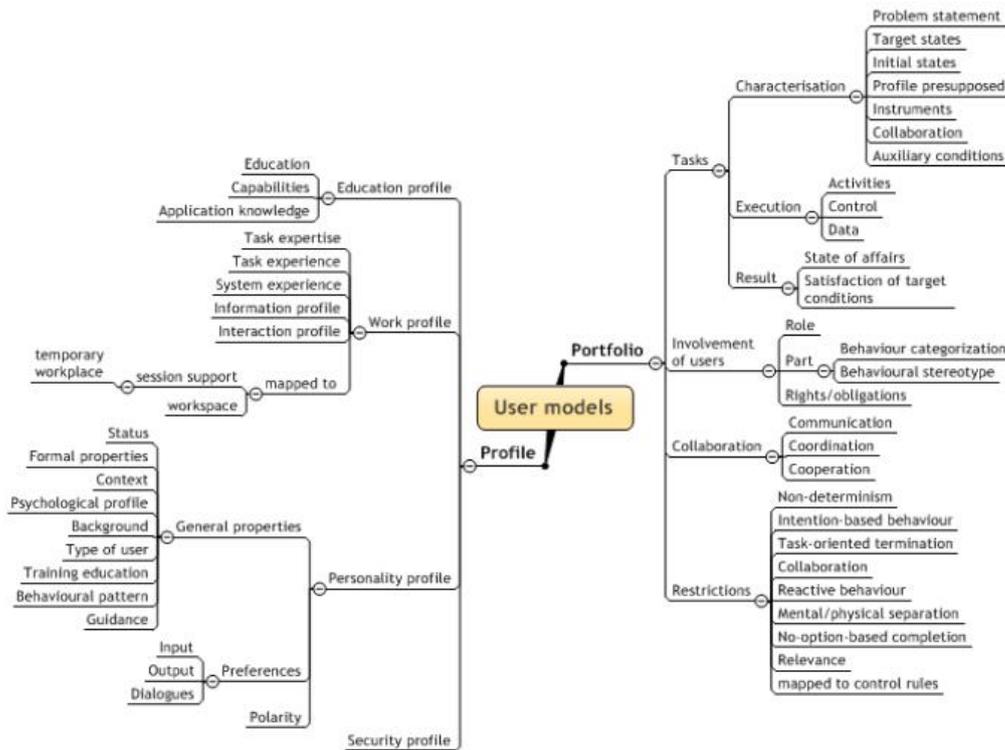
Figure 7. User modeling in adaptive IS

The *personal properties* category of the model covers a personality profile (nine general factors and three dialog preference related factors), a work profile (six factors), and an education profile (three factors). The *portfolio category* includes task-related parameters, a user involvement description, and the type of collaboration and restrictions to be taken into account. A high quality IS implements a good fit between these user profiles and usage portfolio.

Based on this model, we have developed more complex structures for other purposes - human-computer interaction with input, output, and dialog interchange as its components, a user profile specific properties model combining user abilities, HCI preferences, collaboration forms, personality and global behavior in a single mind map.

In this paper we have tried to find means for structuring user properties as an embedded part of the IS itself. Our approach provides understanding about the important factors to be taken into account in user modeling. The studies introduced in this paper (Section 5) and the discussion related to the adaptation structure of IS (Section 3) provide a good starting point for our future work in merging our approach based user models IS?? IS user models based on our approach?. The results of this paper are generic. The relevance to the conference theme comes from two directions: (1) in adaptive systems it is a question of (computer) learning, and (2) every educator should be prepared to validate the educational systems to fulfill the expectations of the students using them. The main principle must be that the system adapts, rather than its user.

REFERENCES

[1] H. Jaakkola, J. Henno, B. Thalheim and J. Mäkelä, "Collaboration, Distribution and Culture – Challenges for Communication," In Proceedings of the MIPRO 2015 Conference, P. Biljanovic, Ed. Opatija: Mipro and IEEE, 2015, pp. 758-765.

[2] H. Jaakkola and B. Thalheim, "Multicultural Adaptive Systems," In Information Modelling and Knowledge Bases XXVI, Thalheim B., Jaakkola H. and Kiyoki Y., Eds. Amsterdam: IOS Press, 2014, pp. 172-191.

[3] ISO/IEC, ISO/IEC 15504-1:2004-2011, Information technology — Process assessment (Parts 1-10), ISO/IEC, 2004-2011.

[4] Carnegie Mellon University Software Engineering Institute, CMMI for Development, Version 1.3. CMMI-DEV (Version 1.3, November 2010), Available at https://www.sei.cmu.edu/reports/10tr033.pdf, Retrieved January 10, 2018, 2010.

[5] Schwaber K. and Sutherland J., 2016. The Scrum Guide, Scrum.org, Available at https://www.scrum.org/resources/scrum-guide, Retrieved January 10, 2018, 2016.

[6] D. Budgen, Software Design, Addison-Wesley Publishing, 2003.

[7] E. Gamma, R. Helm, R. Johnson and J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley Publishing, 1995.

[8] G. Fischer, "User Modeling in Human-Computer Interaction," User Modeling and User-Adapted Interaction vol. 11 Issue 1-2, pp. 65-86, Available at https://doi.org/10.1023/A:1011145532042, Retrieved January 10, 2018.

[9] G. Hofstede, G.J. Hofstede and M. Minkow, Cultures and Organizations: Software of the Mind: Inter-cultural Cooperation and Its Importance for Survival, New York: McGraw-Hill, 2010.

[10] R.D. Lewis, When Cultures Collide. Managing Successfully Across Cultures, 3rd Edition, London: Nicholas Brealey, 2011.

[11] E.T. Hall. Beyond Culture, volume 222, New York: Anchor Books, New York, 1976.

[12] Hall E. T. and Hall M. R., Understanding Cultural Differences, Intercultural Press, 1990.

[13] F. Trompenaars and P. Woolliams, Business across cultures: John Wiley & Sons, 2004.

[14] H. Jaakkola, B. Thalheim, "Web Information Systems for High and Low Context Cultures," Information modelling and Knowledge Bases XXIX, Vol. 301, Amsterdam: IOS Press, 2018, pp. 299-320.

[15] H. Jaakkola and B. Thalheim, "Modelling Cultures," Submitted to the 28th International Conference on Information Modelling and Knowledge Bases, EJC 2018, Not published, 2018a.

[16] K. Reinecke and A. Bernstein, "Knowing what a user likes: A Design Science approach to interfaces that automatically adapt to culture" MIS Quarterly Vol. 37, Issue, 2013, pp. 427-453.

[17] K. Petersen, S. Vakkalanka and L. Kuzniarz, "Guidelines for conducting systematic mapping studies in software engineering: An update", Information and Software Technology Vol. 64, 2015, pp. 1–18, Available at DOI:https://doi.org/10.1016/j.infsof.2015.03.007, Retrieved January 10, 2018.

[18] M. Phaedra and M. Permanand, "Contextualised Student Modelling for Enculturated Systems", In AIED 2013 Workshops Proceedings, Volume 5, The 4th International Workshop on Culturally-Aware Tutoring Systems (CATS 2013), E.Walker and C.-K. Looi , Eds., CEUR Workshop Proceedings Vol-1009. urn:nbn:de:0074-1009-5, 2013, pp. 20-29, Available at http://ceur-ws.org/Vol-1009/aied2013ws_volume5.pdf, Retrieved January 10, 2017.

[19] G. Dafoulas and L. Macaulay, "Investigating Cultural Differences in Virtual Software Teams," The Electronic Journal on Information Systems in Developing Countries Vol. 7 Issue 4, 2001, pp. 1-14, Available at http://www.ejisdc.org, Retrieved January 10, 2018.

[20] F. Fukuyama, Trust: The Social Virtues and the Creation of Prosperity, Hamish Hamilton, 1995.

[21] R. Lessem and F. Neubauer, European Management Systems, McGraw-Hill, 1994.

[22] E. G. Blanchard, S. Karanasios and V. Dimitrova, "A Conceptual Model of Intercultural Communication: Challenges, Development Method and Achievements", The 4th International Workshop on Culturally-Aware Tutoring Systems (CATS 2013), E.Walker and C.-K. Looi , Eds., CEUR Workshop Proceedings Vol-1009. urn:nbn:de:0074-1009-5, 2013, pp. 1-10, Available at http://ceur-ws.org/Vol-1009/aied2013ws_volume5.pdf, Retrieved January 10, 2017.

[23] R. Damaševičius, "On The Human, Organizational, and Technical Aspects of Software Development and Analysis," Information Systems Development G. Papadopoulos, W. Wojtkowski, G. Wojtkowski, S. Wrycza and Zupancic J., Eds., Boston MA: Springer, 2009, pp 11-19, Available at DOI https://doi.org/10.1007/b137171_2. 2009

[24] J. Dugdale, Human behaviour modelling in complex socio-technical systems: an agent based approach. Multiagent Systems [cs.MA], Université Joseph-Fourier - Grenoble I, HAL Id: tel-01006225. 2013, Available at https://tel.archives-ouvertes.fr/tel-01006225, Retrieved January 10, 2018.

[25] D. Johnstone and M. Tate (2004), "Bringing human information behaviour into information systems research: an application of systems modelling," Information Research Vol. 9, Issue 4, July 2004, 22 pages, Available at http://www.informationr.net/ir/9-4/paper191.html, Retrieved January 10, 2018.

[26] W. Huitt, "A systems approach to the study of human behavior," Educational Psychology Interactive, Valdosta,GA: Valdosta State University. 2012, Available at http://www.edpsycinteractive.org/materials/sysmdlo.html, Retrieved January 10, 2018.

[27] C. Johansen, T. Pedersen and A. Jøsang, Reflections on Behavioural Computer Science, University of Oslo, Department of Informatics, Research report 452, April 2012. ISBN 978-82-7368-417-2, Available at http://folk.uio.no/cristi/papers/TR452.pdf, Retrieved January 10, 2018.