# Nuances of Human-Centredness in Information Systems Development

Hannakaisa Isomäki
*Department of Research Methodology*
*University of Lapland*
*PO Box 122, 96101 Rovaniemi, Finland*
*hannakaisa.isomaki@ulapland.fi*

Samuli Pekkola
*Department of Computer Science and*
*Information Systems, University of Jyväskylä*
*PO Box 35, 40014 University of Jyväskylä,*
*Finland*
*samuli@cc.jyu.fi*

## Abstract

*Numerous methods, methodologies, approaches, techniques and tools have been developed over the years to ensure successful accomplishment of information system development (ISD) projects in terms of user satisfaction. However, different methodologies and approaches perceive the user differently; sometimes the user is seen as an anonymous 'object' that is going to use the system, or as an evaluator confirming the correctness of the design, or even as a critical contributor along the way to user-friendly information system. Each of these approaches has their own benefits from the ISD point of view but they lack a holistic view of the user. In this paper, we will review the trajectories of ISD approaches and elucidate the nuances of human-centredness in ISD. We aim at offering a holistic picture that illustrates an overview of different understandings of the user in ISD, so that, first, the systems designers' awareness about the user in general is increased, and second, future research directions are portrayed to the researchers.*

## 1. Introduction

Avison and Fitzgerald [2] discussed the needs for new development methods and changes in the perceptions of the role of users in information systems development processes. Development methods have gone through a long evolution from basic technical problem solving approaches to well-defined multi-methodological techniques. Similarly, the interpretation of the user has changed. Early, the user was seen as a person using the given system. Later, with modern era, it is often recognised that the user needs opportunities to influence the development process and to yield the system according to his or her own needs and preferences. The range of these approaches is still very common in different realms of ISD discipline as pointed out by Iivari, Hirschheim and Klein [28] in their dynamic framework of eleven generic approaches for ISD.

When comparing different perceptions of users in ISD with development projects, it is interesting to realise that the projects incorporate only a certain single perception. For example, users are seen as experts in the requirements specification phase [35], or as resources in the design and testing phases [14], or even as integral participant in the whole development process [52]. This kind of congregated focus is evident as each approach has its benefits. For example, participatory approaches, e.g. participatory design, can provide realistic feedback for the developers throughout the development process but reciprocally require resources and intensive involvement from every actor [3, 24]. On the other hand, requirements engineering approaches focusing solely on the requirements specification phase disregard the active user involvement during the whole project. Consequently they require fewer resources and simultaneously minimise the chances for conflicting requirements (c.f. [3]). However, grounding the development of IS on a single type of user perception does not guarantee a 'human-centred' outcome for the project. As Isomäki [30] argued, users possess various roles and inherent characteristics that should be taken into account in ISD, and thus, a holistic view of users needs to be embedded into the process of ISD in order to facilitate a comprehensive analysis of the various ways that people interact with information, technologies, and tasks.

Nowadays, as human-centredness is becoming a hot topic in the information systems community [2], it is necessary to discuss how the users will be perceived in the future. That is to say, IS designers' awareness and perception on the user needs to be increased. This is of utmost importance because the designers' intellectual frameworks, first, determine the operationalisation of human-centredness within a certain ISD methodology or approach (e.g. [15]), and second, are of practical relevance in that the designers' views of users are mediated to practice through the use of ISD methodologies, methods, techniques, and tools [26, 27]. The purpose of our paper is to promote and contribute to the discussions that aim at increasing user awareness within the field of IS. For this we will present a historical

review of different nuances of human-centredness in ISD in order to both set a context to the problem area concerning the topic, and to be able to comprehend how we have come to the current situation. In this way we attempt to draw a holistic picture of the nuances of human-centredness in ISD, and further, to illustrate how the interpretation of users has changed during the evolution of various ISD approaches.

In the following we will walk through a historical development of the perceptions on end-users in ISD. This is followed by a presentation of the nuances of human-centredness in ISD. Finally, the paper is summarised and future prospects are stated.

## 2. History, Perspectives and Issues

The recurrent period of focusing attention on human issues within IS and their development has its origins at the very outset of computing. The trajectories of different ideas concerning IS development methodologies and approaches are ever-increasingly geared towards a deeper understanding of human-beings as users of computerised information systems. To illustrate this, we will present a brief overview of the most significant approaches or ideas aiming at the humanisation of IS since the 1950's. Just like all action involving IS and people, also the strategies of making the IS human-centred have been shaped within the interaction between humans and constantly evolving information technology. Nevertheless, since the study focuses on the human side of the IS-user -relationship, we shall overlook the solely technological aspects and discuss the following approaches, methodologies and actions with respect to their contribution to the human-centred view of ISD. In the following we point out how development methods have gone through a long evolution from basic technical problem solving approaches to well-defined multi-methodological techniques (Figure 1).

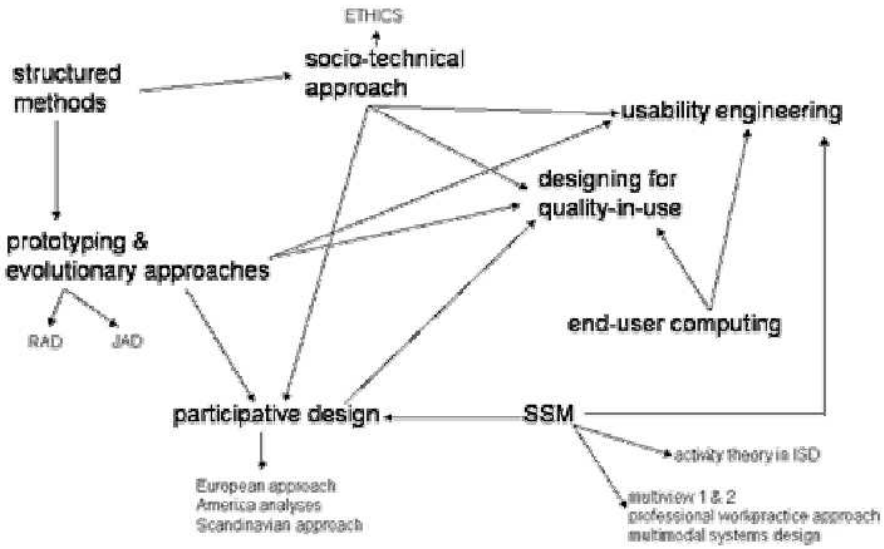


**Figure 1. Accommodation and trajectories of different ideas concerning IS development methodologies and approaches.**

### 2.1. The beginning: structured methods

According to Pain and al. [45], the first ISD methods used in the 1950's consisted mainly of programming, and were accompanied by limited discussions with users about the inputs, outputs and the necessary calculations. At the time, the choices of human-computer interaction were obviously limited in that input was carried out via punch cards, data was stored on magnetic tape, and output was printed on paper. However, as user expectations increased and technology developed, information systems became more complex. The tasks of systems analysis and design, including ascertaining users' requirements, designing data structures and screen layouts, became necessary along with programming. Consequently, methods for controlling and managing the actual ISD process and the numbers of people engaged there became also a necessity. In the 1960's, this instigated several proposals for structured systems development standards, usually referred to as ISD methodologies. They included, for instance, IBM's Vienna development method (VDM) and the British government's Structured systems analysis and design method (SSADM). The idea of design is, first, to characterise the situation in terms of identifiable

objects accompanied by well defined properties; second, to identify general rules that are applicable into the situations in terms of aforementioned objects and properties; and finally, to apply the rules logically to the situation and draw conclusions about what should be done [55]. In a similar sense, structured methods were later developed further to cope with the increased complexity of the analysis and predefined formats for describing and filing the numerous details. Those details are collected during the systems development, beginning with descriptions of the problem and ending with detailed program specifications and user documentation. The most common forms of these methodologies originate in the works of Gane and Sarson, DeMarco and Yourdon [25] and Jackson [22].

The benefits and the deficits of structured methods were soon exposed. A fair amount of criticism has been directed towards this IS approach, which is also often referred to as the rationalistic tradition. The main premise of the criticisms is that the formal objective worldview, that is embedded in the procedures of the rationalistic design tradition, alienates IS designers from the actual nature of the 'object systems'. This occurs particularly with respect to humans [12]. Thus, structured methods are considered to hamper humanised ISD (e.g. [22, 45]). In addition, within these traditional systems development methodologies, the users have little or no role in the design process and thereby had no involvement in the development projects until some training was provided prior to the introduction of the system [53]. Moreover, the use of these methodologies also often resulted in the systems to be sub-optimal independent systems, which were designed for interdependent activities [53].

However, the structured methodologies promoted the human-centred view of ISD. The significance of this tradition in regard to the humanisation of IS lies in that their use made visible the genuine nature of human action. This does not in every respect conform with a formalised and objective world view. In addition, the disappointments and failures related to the use of structured methods triggered new efforts in developing more human-centred methodologies, as is will be point out later.

## 2.2. Prototyping and evolutionary approaches

New user-related problems arose in the late 1970's [25]. Due to the increased pace within business and other organisational action, users could no longer waste years in waiting for their IS to be developed, neither could they wait that long to find out whether the system will meet their needs at all. Another serious problem was that the communication gap between the IS designers and the users continued to widen as computerised information systems geared the IS professionals' attention towards increasingly complex applications. In this situation, new technological tools were applied to ISD in such a way that the users could experiment the system while it was under development so they could get 'hands-on' experience of what the final system would be like. This was the initial human-centred idea behind evolutionary systems development and prototyping. Through prototyping, users could tell much earlier whether the system would meet their needs. The communication between IS designers and users was also seen to be improved. Prototyping allowed users who may previously have had difficulties in formulating and articulating their requirements to better specify their demands. In addition, the flexibility of the prototyping gave IS professionals and users an opportunity to pay more attention to other issues aside just technological ones, for example work design and ergonomic or usability aspects [25].

However, experiences with prototyping also revealed some problems within the approach, as recognised by Friedman and Cornford [21]. First, the impacts of prototyping are, to some extent, limited. Prototyping techniques are usually restricted to the improvements of only a part of users' working environment thus they are potentially ignoring the broader organisational context. The second problem by Friedman and Cornford [ibid.] is quite the opposite to the first one: prototyping alters the balance of power towards the users since in that case they are allowed to make decisions on the design solely from their own point of view. Consequently prototyping may orient the ISD process for the favour of individual users to the detriment of broader organisational issues, such as efficient resource allocation or strategic aims of top managers. Third, prototyping may be misused in terms of manipulating the users to co-operate with the systems whose effects might be unsatisfactory or even deskilling. In spite of these problems, prototyping initiated the transition towards evolutionary and dynamic systems development methods that emphasise user empowerment and participation throughout the ISD process, such as rapid application development [10] or joint application development [56].

With respect to the humanisation of IS, the main impact of prototyping is that it led IS designers to be confronted with the consequences of their designs on users [21]. For this reason, perceiving, understanding, analysing and (re)designing the IS-user -relationship from the users' point of view became an essential task of IS designers.

## 2.3. The socio-technical approach

At the turn of the 1970's and 1980's, a significant transition towards human-centred systems development commenced along with new methodologies. Perhaps the best known human-centred design method, termed as ETHICS, was introduced by Enid Mumford [3, 40]. ETHICS, which is based on the socio-technical systems

theory, is often regarded as the foundation and predecessor of current human-centred methodologies. With respect to the humanisation of IS, a significant aspect of ETHICS is that its 'object system' includes both social and technical features. This idea results from Mumford's observations that social requirements could be achieved much better if they are considered before the design is fixed [25]. Consequently, the design team was divided into two parts so that both the social and the technical design objectives were paid explicit attention to. Also, ETHICS emphasised users' participation in the design process so that their job satisfaction is increased. This was regarded as an ultimate goal of ISD.

Nurminen [44] argued that within ETHICS, it is assumed that job satisfaction is met when an employee's own expectations and the demands directed at him or her sufficiently correspond. This correspondence strengthens the commitment to the work situation establishing five different kinds of engagements, 'fits', between the employee and the employer. 'The knowledge fit' means that the employee is prepared to use his or her knowledge and skills for the employer's benefit. 'The psychological fit' signifies the trust the employee experiences in that his or her well-being is regarded and that his or her work is adequately appreciated, challenging and involves responsibility. 'The efficiency fit' denotes that the employee strives to fulfil the productivity and quality demands and accepts the rules and control actions inherent to the work. 'The task-structure fit' indicates that the work task is broad enough and offers employees an opportunity to actualise themselves in conformity with their abilities. The last engagement, 'the ethical fit', refers to the employee's possibility to act and be respected as a valued human-being and that he or she should have adequate social contacts in work. Based on these principles, the procedures employed in ETHICS essentially place importance on the analysis of the needs of business efficiency, effectiveness, job satisfaction and future change. These factors are then moulded into objectives that are addressed by the two components of design: technical and social [53].

However, dividing an IS into two separate systems is also the weakness of the socio-technical approach: if the social part of the whole system is separated, the system is solely technical. Accordingly, the pitfall in the use of ETHICS is that the technical design objectives are the primary concern and the social objectives are neglected [44]. ETHICS is also undeveloped in that it addresses human characteristics in an inexact manner. Although the 'fits' clearly involve several different human characteristics (cognitive, emotional, volitive, social and ethical) they are all termed social features. Also Ehn and Löwgren [17] have criticised the early socio-technical approach for not being truly participative or democratic, but being managerialist. In addition, Pain et al. [45] argued that the early approaches take too simplistic a view of job satisfaction, skill and the impact of technology.

Nevertheless, the socio-technical approach is very significant in relation to the humanisation of IS as it addresses IS as social systems and makes a serious attempt to offer means for building bonds between the social and technical aspects. Moreover, the socio-technical approach obviously broadened IS research and practice intellectually and gave rise to new delineations where information systems are perceived as technical systems with social implications, or oppositely, IS are social systems but which are only technically implemented [25].

## 2.4. Understanding human activity systems

Another major contribution to the human-centred perspective was proposed in the early 1980's by Peter Checkland [15], whose SSM (Soft Systems Methodology) methodology introduced the concepts of human activity systems and multi-perspectives concerning ISD. Checkland contends that the design related to human activity requires cultural analysis concerning human behaviour. Therefore the IS designers should identify both the roles that are either institutionally or behaviourally defined, and the norms that describe expected behaviour and values inherent in the problem situation [53].

Basically the term 'system' is used as a tool to express different views, or holons, of the real world. These holons are turned into a rich picture by the aid of cultural analysis. Cultural analysis is used to study a problem situation wherefrom it attempts to identify roles, norms, and values that are local to a situation and denote organisational performance. Political analysis deals with managing relations between different interests and identifying how power is expressed within the organisation. Within the logic-based analysis, which is parallel to cultural analysis, the holons are first described by identifying a root definition of the problem and then specifying them as conceptual models. These procedures of SSM can be used at the early stages (analysis) of the system's life cycle but they do not apply to systems design [53]. Checkland's method differs from traditional IS methods as it does not prescribe specific tools and techniques but a general problem-formulating approach.

Soft Systems Methodology is remarkable to the humanisation of IS as it provides IS designers a framework, which does not force or lead them to a certain fixed solution but rather assists them to contemplate and understand the problem situation and human activity within it [25]. In this way, the SSM emphasises IS designers' profound understanding of human action within the object system. This initiative has also led to the development of other ISD approaches emphasising IS designers' deep insight and reflection concerning human action, such as Multiview2 [4], the Professional work

practice approach [38], and Multi-modal systems design [5].

A more recent perspective on human activity systems is the application of activity theory in the study of IS [7, 36]. This perspective offers means to study individuals' actions in a particular context. These actions are not direct but are mediated by various artefacts, such as signs, procedures and instruments. In addition, these mediated activities are a historically developing phenomenon and culturally mediated. This means that the relationships between the main components of an activity system are situation-bound and have developed historically in the course of a particular cultural process. Humans and their activities are examined as a social and cultural process.

## 2.5. End-user computing

Again, in the mid 1980's, a new strategy for user relation problems was promoted. A more accurate fit between humans and IS was pursued by increasing the users' independence and encouraging end-user computing (EUC). According to Friedman and Cornford [21], the general idea was to provide the users with a programming environment, which allows them to tailor a system according to their own needs. The purpose was to increase flexibility in the computer use. With the minimal degree of flexibility, the users are equipped with systems in which the choices are incorporated but they cannot be programmed. Slightly more flexible approach is to implement systems in which choices can be programmed, stored and reused by the users. Even more flexibility can be provided by giving the users the control over the choice of parameters or over operations. The far extreme is to give the users total control over the operations and parameters of the system.

End-user computing, however, did not remove either the IS-user related problem or the need for professional IS staff. While EUC allows users to tailor the IS according to their own preferences, it can lead users to spend more time on developing IS than on doing their actual work through the system. In addition, end-user computing requires good computing skills, which all users necessary do not have. However, the greatest disadvantage of EUC is often seen from the organisational perspective. Uncontrolled end-user computing may lead to wasted resources and result numerous maintenance and compatibility problems [21].

The potential of developing end-user computing as a strategy to improve IS-user relations seems also to have vanished due to the lack of a distinct and commonly accepted definition of the end-users. Although Cotterman and Kumar [16] pointed this out already in the late 1980's when they suggested a taxonomy for understanding and classifying end-users in organisations, nowadays the term EUC is almost a useless due to both its several controversial meanings within the IS literature and consequent inadequacies in specifying its meaning in modern knowledge work environments [20]. The advantage of EUC from the human-centred view point is that it encouraged users to tailor their computer systems according to their own preferences. In so doing, they have also acquired some computing skills while the actual ISD remains still a professional task.

## 2.6. Participative design

Undoubtedly the most noteworthy strategy for humanising IS is user involvement in ISD, i.e. approaches known as co-operative, collaborative, participatory or participative design, which have their origins in the socio-technical approach. Since the 1980's, numerous detailed classifications of user participation in ISD have been proposed. In general, European views are comprised of the distinction between 'weak' consultative participation and 'real' influence over IS design while the American analyses are more focused on personality conflicts and differences in cognitive styles between users and IS designers [21]. A special branch within the European views is the Scandinavian approach [37], which, according to Bjerknes and Bratteteig [11], attempts to, first, improve the knowledge upon which IS are built; second, enable the system's future users to develop realistic expectations and reduce resistance for the change; and third, increase workplace democracy by giving the members of an organisation the right to participate in decisions that are likely to affect their own work. In addition to enhancing workplace and working life democracy inherent to IS, individuals and groups are particularly considered.

It is typical to participatory design the traditional formal systems development techniques were often found to be too abstract thus being inappropriate tools for communication between IS experts and users (e.g. [18]). Hence participative design methods are often termed as design-by-doing where mock-up prototyping is applied and commonly used. Also, co-operation between the parties is also considered crucial. Greenbaum and Kyng [22] argued that user participation should be authentic and full, aiming at enhancing workplace skills rather than degrading or rationalising them. In a similar vein, Bødker and Grønbæk [13] contended that co-operative prototyping is an ongoing mutual learning process involving both IS designers and users.

The emphasis on work situations is prominent in different participative design approaches. This has brought forth different variations of IS methodologies that often rely on ethnographical studies. Ramey and al. [49], for instance, described a practice-oriented application of ethnography in studying users as members of a distinct professional culture. This aims at extracting the actions, the goals of actions and the values that animate them from a 'stream of behaviour'. By iteratively sampling

behaviour and confirming its interpretation with the future users, they build a model of the situation. The advantage in drawing on the ethnography is that it facilitates the capturing of tacit knowledge, which is inherent in human activity.

Understanding humans within mundane work practice is emphasised also in a well-known methodology termed Contextual Design. This method also derives its origins from ethnography but is supplemented by psychological principles concerning, for instance, managing the interpersonal dynamics of an interview and shortening the time needed in observing a long process [8, 9]. Another variant of traditional ethnography suited to swift industrial design is known as rapid ethnography [43]. It is an observational technique for going to the prospective users of a particular product and observing the activities they perform, their interactions, and the subcultural features within their work, learning and play. Rapid ethnography is regarded as critical especially to the invention of new product concepts and classes [43].

Although user involvement is highly regarded within the IS literature, it has not always been found very successful in the practice of ISD [e.g. 31, 32, 34, 41, 48, 54]. For example, already Newman and Noble [41] depicted numerous problems during participative systems development. They include, for instance user resistance, knowledge gap between the IS designers and the users, and the lack of a positive climate of trust. Sutter [54] argued that excessive user involvement slows down the IS effort and often too many user committees just blur the focus and unnecessarily expand the requirements. Moreover, King [34] ascertained that in practice user participation may sometimes be absent. However, there are several contributions made by participative approaches to the humanisation of IS. First, the focus of IS designers' reflection within ISD becomes clearly geared towards humans and their action whereas, for example, in prototyping the focus is on software that is redesigned in accordance with the users' feedback. Second, the nature of human beings was seen in a broader sense than before. Human behaviour is understood in terms of social interactions, e.g. the rituals, ceremonies, norms and symbols both consciously and unconsciously present in everyday life. Third, power relations were explicitly addressed by the Scandinavian approaches, which emphasised that users should be in control of their own work. In this way human action in the context of ISD is reflected in relation to the actions of society.

## 2.7. Integrating issues of human-computer interaction into ISD

During the 1990's the trend on focusing human-centred issues in ISD continued, so several new methodologies and standards appeared. Often those are referred to as human-centred development or usability engineering, which aim at combining knowledge and methods from the field of human-computer interaction (HCI) or computer-supported co-operative work (CSCW) in the ISD process or software engineering. For example, Nielsen's [42] model for usability engineering emphasises the pre-design phase when the designers should get to know the users to be able to define their individual characteristics, current and desired tasks besides performing functional analysis. Based on this pre-design, the actual systems design is carried out as an iterative process employing both heuristic analysis and a variety of participatory design methods.

The usability engineering life-cycle developed by Mayhew [39] follows much the same guidelines. The life-cycle model divides the ISD process into four phases and points out appropriate points for the usability design tasks in relation to the ordinary development tasks. Mayhew's approach stresses that the typical ISD tasks must be supplemented with knowledge concerning users. This knowledge contains information about the user profiles and contextual task analyses, and aims at defining usability goals in the requirements phase. In addition, mock-ups and prototyping are applied within iteratively conducted design process. Quite similar is also the International Standardization Organization's (ISO) standard for human-centred design processes for interactive systems [29]. The standard emphasises active user involvement and a clear understanding of both the user and the task requirements in the early phases of design. There, simulations and user tests are incorporated iteratively within the design. Also other, quite similar usability engineering approaches or methodologies have been developed not only by HCI community but also in ISD and CSCW communities (e.g. [13, 23, 33], c.f. [3]). A common idea underpining these methodologies is that the IS should be considered both from the usability and their utility points of views [42].

Ehn and Löwgren [17] delineated the evolution of HCI and ISD as being consolidated as an approach referred to as Design for quality-in-use. They asserted that the evolution of the usability concept in HCI, and the methodological evolution in the field of ISD have yielded a move from an exclusively rationalistic and objective perspective to the inclusion of interpretive social and subjective aspects. In other words, the traditional rationalistic way of constructing IS, and the tradition of experimental psychology in HCI, have evolved towards a holistic approach that combines methods of Contextual design and Participatory design, and have further developed into interaction design, which requires a particular design ability. This ability refers to the competence to study IS in use from three different standpoints: structure, function and form. The structure of a system is its material or medial aspects, i.e. the technology (hardware and software). The structural aspects are objective as they are inherent in the

construction of the IS, and less dependent on context and human interpretation. The functional aspects of a system concern its actual, contextual purpose and utilisation. Different users have different purposes for and usage of a system. Functional aspects include organisational performance and functions beyond the simple utilities of the system. The form of a system expresses the human experience of using the system. Form is not necessarily a property of the system, but rather a relation between system and user. Designing for quality-in-use emphasises that all the three aspects constitute competence in current ISD [17].

In addition to the above mentioned methodologies and approaches the humanisation of IS has been promoted by discussions reflecting ethical concerns in ISD. The contemporary discussions of computer ethics concern both academic researchers and IS professionals in companies [19]. These discussions assume an official form in the IS professionals' codes of ethics, which indicate norms for performance in the IS designers' professional activity. Two most central manifestations of IS professionals' codes of ethics have been worked on and published by the Association of Computing Machinery (ACM) and the International Federation on Information Processing (IFIP). These codes of ethics bring forward stances widely shared by IS designers and also researchers. Additionally, the majority of the industrial countries have produced their own codes of ethics for ISD (c.f. [6]).

These codes pay a considerable amount of attention to standpoints concerning human well-being. The ACM code of ethics stresses that a fundamental aim of computing professionals is to minimise negative consequences of computing systems, including threats to health and safety. In addition to a safe social environment, human well-being includes a safe natural environment. The values of equality, tolerance and respect for others are looked upon as essential in nature. Especially, discrimination on the basis of race, sex, religion, age, disability, national origin, or other such factors is considered as an explicit violation of ACM policy. In a similar vein, the IFIP code of ethics binds the IT professionals to advance international human welfare and the quality of life for citizens of all nations. These improvements aim at morally desirable goals such as personal development, physical safety, personal dignity and human fulfilment in computerised workplaces. Particular threats to health are poorly designed human-machine interfaces, which are seen to cause stress symptoms. It is also regarded as important that current system users, potential users and other persons whose lives may be affected by a system must have their needs assessed and incorporated in the statement of IS' requirements. In this way IS designers have affirmed their obligation to continually humanise information technology.

As a critical view of the implications for the humanisation of IS it can be stated that the codes of ethics are formal documents that professional organisations themselves produce in order to make known their stance and policy on ethical issues within a profession. As such, they express the desired status of things – 'what ought to be' – but do not offer explicit guidance for achieving the desired goals. Unfortunately, explications of ethically valid intentions do not furnish a guarantee for the actual realisation of human-friendly information technology. In addition, the construction of IS professionals' codes of ethics has been considered as a response to the need for professionalisation – for fulfilling the characteristics of a profession – rather than as a reaction to runaway problems in the field [1].

## 3. Nuances of human-centredness in ISD: summary and future work

As depicted above, taking end-users into account in ISD has been pursued by developing different methodologies and approaches for ISD, with the aid of administrative actions and training. Nevertheless, the issue is still very important. In this paper, this is demonstrated through the lack of a holistic picture of how the users should be studied or considered in the ISD, or what should their role be there. Also, following the discussions on end-user computing, a common understanding of the user in term of terminology or taxonomy is still missing.

Early on in our paper, we asked how the IS designers' general awareness on users-centred ISD approaches can be increased. The requirements for IS designers to understand human characteristics and behaviour can be seen to increase in various contexts within contemporary IS research and practice. We attempt this by distinguishing relations between different approaches. Figure 1 earlier displayed the methodologies and approaches discussed in this paper, and the relations of transitions of ideological ideas and perspectives from one to another. For example, participative design approaches often incorporate prototyping or other evolutionary methods. Also, those approaches usually study both social and technical systems from the organisation cultural viewpoint. Correspondingly, the soft systems methodology origins from socio-cultural studies but its ideas on cultural analysis in a certain context have been utilised both in participative design approaches and in usability engineering.

The comparison between our approach and for example Iivari et al. framework of ISD methods [28] depicts a typical ISD methodology approach; usually the methods do not explicitly focus on humans but technologies instead. If the methods focuses on the user, the approach is commonly managerial or organisational, i.e. the user is seen as a target subject that is going to use

the system as specified by managers or designers. However, as already the early studies on CSCW point out, this is a completely wrong approach – users always find out different ways to (mis)use the system in another way than specified in the design [47, 50, 51]. On a way to IS designers' increased awareness on the user and his or her behaviour in a certain context for which the system is designed, we believe it is important to explicitly list out the approaches and their position towards the user. This is accomplished in Table 1, which summarises earlier chapters.

**Table 1. Nuances of human-centredness in different ISD methods.**

| | **User perception in ISD process** | **User role in organisation information processing** | **Behavioural nuances in ISD** |
|---|---|---|---|
| Structured methods | User is consulted only about the inputs and outputs of the system. | An 'object' whose task is to be supported. | Not considered |
| Prototyping and evolutionary approaches | An evaluator of the design decisions. | Task is considered through professional role in the organisation. | User performs certain tasks. |
| The socio-technical approach | ISD process is divided into two separate systems, social and technical, that are supposed to be integrated (only) at the end. | The needs for organisational information processing are considered comprehensively consisting of both individual and organisational points of views. | In principle, the user is considered as a psychological, emotional and social actor. However, the human-centred issues might be overlooked. |
| Soft systems methodology | ISD process as a whole is disregarded. SSM emphasises only the early (analysis) phase of the development. | Organisational performance is dealt through social norms and power relations. | User is considered as a social, cultural and political actor. |
| End-user computing | ISD process as a whole is disregarded. EUC focuses on the utilisation of the system. | Organisational issues are omitted, individual users' preferences are emphasised. | User is regarded as a computer expert. |
| Participative design | The connection between ISD process and PD methodologies s is weak. | Scandinavian approach attempts to combine user's information utilisation to organisational objectives. In European and American approaches, user role is weaker and only consultative. | User is domain expert that performs certain task. |
| Human-computer interaction in ISD | Process models are many, but their use in in practice is questionable. | Individual users are emphasised in terms of their own information processing functions. | User is seen as a cognitive and collaborating actor. |

Although the evolution of ISD methodologies has geared towards human-centredness, as asked by Avison and Fitzgerald [2], it is still common that the user-oriented methods are not connected with ISD processes. The methods increase designers' understanding about the user, but they do not implicitly and unambiguously provide guidelines to combine that information with the systems development process (c.f. [32]). Why it is so? Following the structuration theoretical approach to conceptualise ISD discipline [46], we argue there is a lack of information exchange between the realms of IS use (i.e. a community studying the workplaces and attempting to make design suggestions for the systems designers) and the scrutiny of ISD methods and practices (i.e. community developing ISD methods). Socially-oriented human-centred methods, that are common in the former community, are not known well enough by the latter community – and vice versa; technical process-oriented ISD methods are not know by the former community. To build adequate user-friendly information systems, we believe it is essential not only to understand the technology development, of which is usually not a problem for an ISD expert, but also to understand the human-being intended to use the system. Hence, we ask for more intensive multidisciplinary cooperation between the 'techics' and 'softies' so that really human-centred ISD methods are to be developed. Our own analysis of the historical development will be continued in the future by analysing the above identified nuances in terms of ISD projects and processes.

## 4. References

[1] Adam, A. 1999. "Computer ethics in a different voice". In Gilson C.H.J., Grugulis I. & H. Willmott, (Eds.) *Proceedings of the 1st critical management studies conference. Information technology and critical theory - stream*, July 14-16, 1999, Manchester, U.K. URL: http://www.mngt.waikato.ac.nz/ejrot/cmsconference/papers_infotech.htm. 16 p.

[2] Avison, D. & Fitzgerald, G. 2003. "Where now for development methodologies?", *Communications of the ACM*, 46(1).

[3] Avison, D. & Fitzgerald, G. 2003. *Information Systems Development: Methodologies, Techniques and Tools*. Third Edition, McGraw Hill.

[4] Avison, D.E., Wood-Harper, A.T., Vidgen, R.T. & Wood, J.R.G. 1998. "A further exploration into information systems development: The evolution of Multiview2". *Information, Technology and People* 11(2), pp. 124-139.

[5] Bergvall-Kåreborn, B. 2000. "*Using soft systems methodology as a methodology for multi-modal systems design*". Luleå University of Technology. Department of Business Administration and Social Science. 2000:7.

[6] Berleur, J. & Brunnstein, K. (Eds.) 1996. *Ethics of compting. Codes, spaces for discussion and law*. On behalf of IFIP. London: Chapman-Hall.

[7] Bertelsen, O. W. & Bødker, S. 2003. "Activity Theory". In Carroll , J.M. (ed.). *HCI Models, Theories, and Frameworks: Toward an Interdisciplinary Science.* Morgan Kaufman Publishers, pp. 291-324.

[8] Beyer, H. & Holtzblatt, K. 1996. "Contextual design: Principles and practice". Wixon, D. & J. Ramey (Eds.) *Field methods casebook for software design*. N.Y.: Wiley & Sons, pp. 301-330.

[9] Beyer, H. & Holtzblatt, K. 1998. *Contextual design. Defining customer-centered systems*. San Francisco, CA: Morgan Kaufmann.

[10] Beynon-Davies, P., Carne, C., Mackay, H. and D. Tudhope. 1999. "Rapid application development (RAD): An empirical review". *European Journal of Information Systems* 8, pp. 211-223.

[11] Bjerknes, G. & Bratteteig, T. 1995. "User participation and democracy: A discussion of Scandinavian research on system development". *Scandinavian Journal of Information Systems* 7(1), pp. 73-98.

[12] Bødker, S. & Greenbaum, J. 1993. "Design of information systems: Things versus people". E. Green, J. Owen & D. Pain (Eds.) *Gendered by design? Information technology and office systems*. London: Taylor & Francis.

[13] Bødker, S. & Grønbæk, K. 1991. "Cooperative Prototyping: Users and Designers in Mutual Activity". *International Journal of Man-Machine Studies*, 34, Special Issue on CSCW, pp. 453-478.

[14] Boehm, B. W. 1984. "Verifying and Validating Software Requirements and Design Specifications". *IEEE Software*, 1(1), pp.75-88.

[15] Checkland, P. 1981. *Systems thinking, systems practice*. Chichester: Wiley.

[16] Cotterman, W.W. & Kumar, K. 1989. "User cube: A taxonomy of end users". *Communications of the ACM* 32(11), pp. 1313-1320.

[17] Ehn, P. & Löwgren, J. 1997. "Design for quality-in-use: human-computer interaction meets information systems development". In Helander, M., Landauer, T.K. & P. Prabhu (Eds.) *Handbook of human-computer interaction*. Amsterdam: Elsevier, pp. 299-313.

[18] Ehn, P.1988. *Work-oriented design of computer artifacts*. Stockholm: Arbetslivscentrum.

[19] Eriksson, I., Siponen, M.T., & Vartiainen, T. 1999. "Preface". *Proceedings of the first international computer ethics workshop in Finland*, May 28th 1999, Jyväskylä. Technical Reports TR-21, Vol. 2. University of Jyväskylä, Department of Computer Science and Information Systems.

[20] Forsman, L. 1998. *Re-engineering end-user support in distributed organizational computing. Moving from reactive to proactive mode of operation*. Acta Universitatis Tamperensis 640. Tampere: University of Tampere.

[21] Friedman, A.L. & Cornford, D.S. 1989. *Computer systems development. History, organization and implementation*. Chichester: John Wiley & Sons.

[22] Greenbaum, J. & Kyng, M. 1991. "Introduction: Situated design". In Greenbaum, J. & M. Kyng (Eds.), *Design at work: Cooperative design of computer systems*. Hillsdale, N.J.: Lawrence Erlbaum Associates, pp. 1-24.

[23] Grønbæk, K. 1991. *Prototyping and Active User Involvement in System Development: Towards a Cooperative Prototyping Approach.* Computer Science Department, Aarhus University, Ph.D. Thesis.

[24] Grudin, J. 1993. "Obstacles to participatory design in large product development organizations". In D. Schuler ja A. Namioka (Eds.) *Participatory design: Principles and practices*. Hillsdale, NJ: Erlbaum, pp. 99-119.

[25] Hirschheim, R., Klein, H.K. & Lyytinen, K. 1995. *Information systems development and data modelling. Conceptual and philosophical foundations*. Cambridge: Cambridge University Press.

[26] Iivari, J. 1991. "A Paradigmatic analysis of contemporary schools of IS development". *European Journal of Information Systems* 1(4), pp. 249-272.

[27] Iivari, J., Hirschheim, R.A., & Klein, H.K. 1998. "A paradigmatic analysis contrasting information systems development approaches and methodologies". *Information Systems Research* 9(2), pp. 164-193.

[28] Iivari, J., Hirschheim, R., & Klein, H. K. 2001. „A Dynamic Framework for Classifying Information Systems Development Methodologies and Approaches", *Journal of Management Information Systems*, (17)3, pp. 179-218.

[29] ISO 1999. "International standard". *Human-centered design process for interactive systems*. ISA/FDIS 13407.

[30] Isomäki, H. 2002. *The prevailing conceptions of the human being in information systems development: Systems designers' reflections*. University of Tampere, Department of Computer and Information Sciences, Finland. A-2002-6.

[31] Järvi, T. & Reijonen, P. (Eds.) 2003. *People and Computers: Twenty-one ways of looking at information systems*. Turku Centre for Computer Science, TUCS General Publication no 26.

[32] Kaarilahti, N., Pekkola, S., & Pohjola, P. forthcoming. "Formalising End-User Participation in Information Systems Development Process: Bridging the gap between participatory design and ISD methodologies". *Submitted.*

[33] Ketola, P. 2000. "Concurrent Usability Engineering". In McDonald S., Waern Y. and Cockton G. (Eds.) *People and computers XIV - Usability or else. Proceedings of HCI 2000*. Sunderland, U.K.: Springer-Verlag, pp. 149-161.

[34] King, J. 1995. "Sketchy plans, politics stall software development". *Computerworld*, June 19, p. 81.

[35] Kotonya, G. & Sommerville, I. 1998. *Requirements Engineering: Processes and Techniques*. John Wiley & Sons.

[36] Kuutti, K. 1997. "Activity theory as a potential framework for human-computer interaction research". In Nardi, B. (Ed.) *Context and consciousness. Activity theory and human-computer interaction*. Cambridge, MA: The MIT Press, pp. 17-44.

[37] Kyng, M. 1994. "Scandinavian Design: Users in Product Development". *Proceedings of the Conference on Computer-Human Interaction (CHI'94)*, ACM Press, 3-9.

[38] Matthiassen, L. 1998. "Reflective systems development". *Scandinavian Journal of Information Systems* 10(1&2), pp. 67-118.

[39] Mayhew, D.J. 1999. *The usability engineering lifecycle: A practitioner's handbook for user interface design*. San Francisco: Morgan Kaufmann Publishers.

[40] Mumford, E. 1983. *Designing human systems - The ETHICS method*. Manchester: Manchester Business School.

[41] Newman, M. & Noble, F. 1990. "User Involvement in an Interaction Process: A case study". *Information Systems Research* 1(1), 89-110.

[42] Nielsen, J. 1993. *Usability engineering*. Cambridge, MA: Academic Press.

[43] Norman, D.A. 1998. *The invisible computer*. Cambridge, MA: MIT Press.

[44] Nurminen, M.I. 1988. *People or computers: Three ways of looking at information systems*. Lund: Studentlitteratur.

[45] Pain, D., Owen, J., Franklin, I. & Green, E. 1993. "Human-centred systems design: A review of trends within the broader systems development context". In Green, E., Owen, J. and D. Pain (Eds.) *Gendered by design? Information technology and office systems*. London: Taylor & Francis.

[46] Päivärinta, T. & Pekkola, S. 2003. "Structurational Approach to Studying Research and Practice of Information Systems Development". *The JAIS sponsored theory development workshop.* After ICIS'2003. December 14-17, 2003.

[47] Pekkola, S. 2003. "Designed for Unanticipated Use: Common artefacts as design principle for CSCW applications". M. Pendergast, K. Schmidt, C. Simone and M. Tremaine (Eds.) *Proceedings of the 2003 International ACM SIGGROUP Conference on Supporting Group Work (GROUP'03)*. ACM Press, pp. 359-368.

[48] Prinz, W., Mark, G., & Pankoke-Babatz, U. 1998. "Designing Groupware for Congruency in Use". *Proceedings of computer-supported cooperative work (CSCW'98)*. ACM Press. pp. 373-382.

[49] Ramey, J., Rowberg, A.H. & Robinson, C. 1996. "Adaptation of an ethnographical method for investigation of the task domain in diagnostic radiology". Wixon, D. & J. Ramey (Eds.) *Field Methods Casebook for Software Design*. N.Y.: John Wiley & Sons, pp. 1-195.

[50] Robinson, M. 1993. "Design for unanticipated use ...". G. de Michelis, C. Simone, and K. Schmidt (Eds.) *Proceedings of the Third European Conference on Computer Supported Cooperative Work (ECSCW'93)*. Kluwer Academic Publishers, pp. 187-202.

[51] Schmidt, K. and Bannon, L. 1992. "Taking CSCW Seriously: Supporting Articulation Work". *Computer Supported Cooperative Work: The Journal of Collaborative Computing*, 1(1), pp. 7-40.

[52] Schuler, D. & Namioka, A. (Eds.) 1993. *Participatory design: Principles and practices*. Hillsdale, NJ: Erlbaum.

[53] Smith, A. 1997. *Human computer factors: A study of users and information systems*. London: McGraw-Hill.

[54] Sutter, J. 1999. "Letters to the editor: The case of/for the missing user". *Communications of the AIS* 2(1), November 1999. <URL: http://cais.aisnet.org/letters/2-1/article.htm>. 8 p.

[55] Winograd, T. & Flores, F. 1986. *Understanding computers and cognition: A new foundation for design*. New Jersey: Ablex.

[56] Wood, J. & Silver, D. *Joint Application Development*. John Wiley and Sons. 1995.