6

Envisioning Improved Work Practices and Associated Technology Requirements in the Context of the Broader Socio-technical System

Joan Cahill
School of Psychology, Trinity College Dublin
Ireland

1. Introduction

Work practice and technology innovation presents a number of challenges for Human Computer Interaction (HCI) designers. Chief among them is the question of devising suitable HCI methods, for future process and task envisionment and related technology design. Methods must facilitate work practice re-engineering/envisionment and the development of user friendly work tools. Despite the future oriented nature of this activity, and its associated outputs, research must be predicated on a clear model of existing processes, task practices and tools usage.

HCI research in both commercial and technology research settings, is undertaken in the context of the broader software development process. As such, HCI methods must deliver clear user requirements for use by Software Developers. Nonetheless, HCI resources may be limited, or the research subject to time constraints - impacting on the scope of HCI research. As such, a valid research design which delivers on the core research brief, while taking into account project constraints, is required.

HCI design methodologies are used at different points in the software development lifecycle to design new technologies or re-design existing technologies, in the context of both open and closed systems. Typically, open systems involve the performance of a series of work processes requiring both individual and/or group task activities. Usually, these activities require operator interaction with a range of technical (e.g. IT systems) and human agents. Further, such interactions are subject to external influences. In contrast, closed systems are characterized by one to one user interaction with simple software packages in office or home computing settings. These interactions are unaffected by external influences.

This chapter focuses on the use of HCI methods in the context of open systems (or socio-technical systems). Specifically, it investigates methodologies for the envisionment of new or improved task practices and associated technology requirements, taking into account the broader socio-technical context for human machine interaction. First, an overview of the methodological implications of a range of conceptual frameworks, relevant to an understanding of human interaction with computer systems in socio-technical settings is provided. Following this, a summary of the software development process and the different requirements distinguished in this process, is presented. An introduction to Human Factors and HCI is then provided. Proposed HCI methodological requirements are then specified. Following this, specific HCI and work analysis methodologies are reviewed, as part of identifying an overall integrated HCI design approach. This is followed by an examination of certain practical challenges facing HCI practioners. In so doing, the author will consider the application of best practices in a real world setting, where HCI research is subject to commercial, technical and organisational constraints.

The HCI methodology outlined in this chapter may be of interest to HCI researchers or practitioners tasked with process and technology envisionment, and/or investigating the links between HCI theory and methods. The specific HCI research methodology proposed and related discussion of practical issues is also relevant to HCI researchers working with limited resources in both commercial software development and/or technology research settings. Further, the specific user requirements gathering methods examined, may be of interest to Software Developers and/or Business Analysts.

2. Conceptual Frameworks and Methodological Implications

2.1 Background

It is well established in the HCI literature that technology systems either fail, or do not perform as well as they might do, because they are not optimised from a user task perspective (Norman, 1988, 1993 and Preece, 2002). Perhaps this seems an obvious point. However, defining the nature of the task, and envisioning new or improved work practices and associated tool requirements, is not a straightforward activity. The question 'what is the task' must be explored on a number of levels. This links to certain theoretical models concerning the relationship between process, task and technology design, and specifically, the relationship between operator task performance and tools and information flow design. Importantly, an investigation of these models suggests certain methodological requirements for HCI design.

2.2 Introduction to Socio-technical Systems

In order to understand the methodological requirements for technology design in socio-technical settings, we must first understand the nature of socio-technical systems and how they perform. A 'socio-technical system' is defined as any instantiation of socio and technical elements engaged in goal directed behaviour. In place of a formal definition, engineering psychologists have proposed a range of characteristics to describe these systems. Characteristics include: large problem spaces, social, heterogeneous perspectives, distributed, dynamic, potentially high hazards, many coupled subsystems, automated, uncertain data, mediated interaction via computers and disturbances (Perrow, 1984, Vicente, 1999).

2.3 Basic Concepts Socio-technical Systems

The definition of a number of basic concepts in socio-technical systems helps illuminate certain aspects of the HCI design problem, which should be considered by HCI professionals. Before discussing socio-technical systems theories, a brief explanation of a number of basic socio-technical concepts is provided.

The operational goal refers to the purpose of the operation or the state of affairs to be achieved (e.g. safe and on time flight). This is associated with a series of operational states necessary to the achievement of the goal and a specific 'end state' which marks the successful accomplishment of the operational goal. The operational process defines the logic or structure of work, so that the operational objective is achieved. This includes the distribution of work or task activities between different human and technical agents and the overall timeline for this (e.g. sequencing of tasks). An operational process can be divided into a number of sub processes. Typically, this includes a planning process and the active operation. The active operation requires certain prior work to be accomplished (e.g. all technical and human resources in place). This work is undertaken in the planning phase. In the active operation, the planned work is executed. The operational process can be conceptualized in relation to a series of process gates (or critical points in the operational process). At each process gate, work must be accomplished by different operational agents, so that the process can move forward. Overall, the collective accomplishment of work at each of these process gates, results in the achievement of the operational goal. The process state refers to the status of the process at any point in time, in relation to the achievement of specific operator tasks. A process dependency refers to a relationship between two different parts of a process or two sub-processes. For example, the relationship between the planning sub process and the active operations sub process. Process dependencies also include dependencies between two related but different processes. In terms of a flight operation, this could be the relationship between the active flight operation, and the line maintenance process. Underlying these process dependencies are specific task dependencies. The operational plan describes how the operational goal will be achieved from an organisational perspective. This includes a definition of what human and technical resources will be used at different points in the operation. It also includes any regulatory requirements to be adhered to. Certain background organisational processes are required to ensure that the operational objective is achieved in a safe, efficient and legal manner. This includes the management of a range of organisational functions such as procedures design, documentation management, training, human resources, safety management and risk management.

The realisation of the operational goal requires the accomplishment of work or tasks by different members of the operational team. In socio-technical systems, work is realized by a number of operational agents or resources. This includes human and technical resources. Human resources refer to the people in the system. Technical resources denotes both the tools used by operator to perform their tasks (e.g. procedures, paper tools, IT systems), and all relevant technology (e.g. machines or systems) required to achieve the operational objective. In socio-technical systems, the work environment is distributed in space. As such, both human and technical resources can be situated in similar or remote locations.

Individual units of work are described in terms of tasks. As defined by Kirwan and Ainsworth, a task is 'a set pattern of operations, which alone, or together with other tasks, may be used to achieve the goal' (1992). Task performance is the enactment of the relevant operational task in time and space. The literature distinguishes between the performance of technical and non technical tasks. Technical tasks refer to specific physical tasks undertaken in order to achieve the operational goal. Typically these tasks are described in company standard operating procedures documentation. Non technical tasks denote the cognitive and social aspects tasks that underlie technical task performance. This includes situation assessment, decision making, task management, communication and co-ordination. Often these are not defined in company documentation. Further, it should be noted that task practice does not necessarily follow the task descriptions provided in company SOP. As such, SOP task descriptions should not be read as definitive.

Depending on the work requirements, operators may perform individual tasks in a sequence, or a number of tasks may be performed in parallel. Typically, tasks are analysed in terms of a hierarchy (e.g. task, sub-task and actions). Depending on the complexity of the task, the task might be grouped into a number of smaller steps or sub-tasks. A sub-task reflects a grouping of related actions, which form an overall step in a task. An action refers to the smallest unit of activity. Actions are associated with human roles, machines/tools and technologies. In relation to human performance, this includes technical activity (e.g. selecting a control on an information display or panel) and non technical activity. Non technical activity includes a range of cognitive (e.g. attending to information on a display, decision making) and social functions (e.g. communicating or co-ordinating with other operators in relation to work activities).

Task dependencies refers to relationships between tasks (both technical and non technical tasks) performed by individual operators or by a group of operators (collaborating on the same task, or producing outputs relevant to each others tasks) at different points in time, throughout the process. Two types of task dependencies can be distinguished. This includes prior or sequential dependencies and parallel dependencies. Prior dependencies refer to task activities and associated task outputs performed by the same or other operators, which are inputs to next phase of work. Critically, there are two aspects to this. Firstly, task performance must be considered in terms of task completion. The task needs to be completed, so that the process can continue. In the example of a flight operation, the Captain must obtain technical signoff of aircraft, before proceeding to close the doors and commencing aircraft push-back. Certain tasks can span a number of process gates or not. However, at a certain point in the operational timeline, tasks become mandatory from a process stability perspective. Also, the quality of task performance must be considered. Tasks may be performed, but the quality of task performance may be weak. For example, poor briefing or situation awareness at one point in flight can have a knock on effect on task performance at a later point in flight. Parallel dependencies concern work undertaken in parallel by other agents, which is an input to the operator's task.

In socio-technical systems, human actors are assigned a role. This corresponds to a set of functions or tasks that they are required to perform in relation to the achievement of the operational goal. Certain actors may have the same overall role, but perform different tasks based on their rank or seniority. Further, in team work situations, a number of actors may collaborate in the performance of the same task or different tasks, either in sequence or in parallel. These actors might have similar roles and ranks, or similar roles and different ranks, or different roles. Consequently, for each task we must distinguish the (1) active role (directly involved in the task) and the (2) supporting roles (contributes or provides inputs but is not directly responsible for the task). The supporting role might include actors with a similar role to the active role, or with different roles. Importantly, the supporting role may or may not be involved in the performance of other tasks at the same time. As such, we must consider how the actions of other agents relate to primary role actions.

Task performance often requires the use of different types of tools. A number of definitions of tools are provided. Overall, a tool can be defined as a thing (concrete or abstract) that supports task performance - either directly indirectly. From a workplace perspective, the term 'tool' refers to a range of entities - both real and abstract - which are used to perform tasks or to assist in the performance of tasks. This includes paper based information resources (e.g. paper based descriptions of a task or procedure, checklists etc), machines (e.g. mechanical machines, simple computer systems and complex computer systems) and human based information resources (e.g. memory, mental models, expertise, cognitive methodologies and so forth). In this respect, workplace tools can be physical (e.g. paper tools or IT systems) or non physical (e.g. best practice methods or expertise). Critically, workplace tools allow operators to perform tasks that are difficult or impossible given certain physical and/or cognitive limitations. For example, tools can provide a mechanical means to undertake certain physically complex or dangerous tasks. Further, tools enhance our ability to complete difficult cognitive tasks (e.g. processing large amounts of information). In this way tools shape task performance and in particular, extend our cognitive abilities (Norman, 1988, 1993). Certain types of tools are referred to as 'information resources'. An information resource is a tool that provides information relevant to the performance of a task. Information resources include physical resources (e.g. paper tools and IT systems) and human resources (e.g. other operators in the system who provide information to the operator or the operators own memory or expertise). IT systems can provide different levels of information. This ranges from raw data relevant to the performance of a task, to specific decision instruction – depending on the level of automation provided. Depending on the task and tool design, one or more physical tools and/or information resources are used by operators in the performance of task functions. In complex systems (e.g. such as aviation and process control), operators interact with a range of part-task tools, to complete a task. In this instance, the range of part task tools form a system of tools which taken collectively support task performance. From a task performance perspective, integration between different systems or tools is critical (Wickens, 2000).

2.4 Socio-technical Systems Theories & Methodological Implications

What is the nature of socio-technical system performance? How do the elements of a socio-technical system relate? Further, what are the implications of socio-technical system performance theories in terms of HCI research design? A number of theories have been advanced in relation to the overall elements of a socio-technical system. Collectively, these theories build on the basic conception of a socio-technical system as containing three overall elements. This includes the social system, the technical system and the environment. This follows from the socio-technical models of Pasmore (1988) and Trist (1981). Further, it links to frameworks associated with Activity Theory (Leontev, 1974). Overall, socio technical theories emphasize the inter-related nature of the social and technical aspects of a work process. Central to these theories, is the contention that there is a relationship between individual task performance and the design of the overall operational and organisational system (McDonald, 2004, 2006). Specific theories highlight the importance of certain social aspects of organisational performance. This includes the role and organisation of people (e.g. linking to the design of processes and procedures) and the specific social interactions that underlie task performance (e.g. communication and co-ordination). Further, theories point to the gap between formal processes and actual operator task practices. The implication of these theories is that design methodologies should allow for an understanding of the socio-technical context for task performance. Specifically, proposed technology concepts should be embedded in a broader system model. In particular, future technology envisionment must take into account the relationship between task and process. This includes both operational and organisational processes.

2.5 HCI & Information Behaviour Frameworks & Methodological Implications

A range of theoretical frameworks have been proposed to describe human interaction with computer systems, in the context of socio-technical systems. This includes HCI theories and information flow theories. Critically, these theories can be interpreted as suggesting certain methodological requirements for HCI practioners.

HCI theories such as Distributed Cognition (Hutchins, 1992, 1995, Hollan et al, 2000) and Group Supported Co-operative Work (Bannon, 1991), point to the role of tools and information in shaping operator task performance. Further, such theories emphasize the sense in which operator task performance often involves collaboration with other human and technical agents. The implication of these theories is that proposed methodologies should allow the researcher to understand how tools and information shape task performance. Further, methodologies might facilitate the identification of information flow requirements linked to the task performance of all relevant team agents – both human and technical.

In relation to task performance, HCI theories refer to the task problem to be solved or the task objective. Critically, the tools that operators use provide a means to solve the task problem. In this respect, HCI theorists (Norman, Carroll and Bannon) argue the specific design of a tool influences both the nature of the task and how it is performed (e.g. task workflow and level of complexity). Specifically, Norman (1993) uses the term 'cognitive artefacts' to describe those tools that given their design (e.g. task representational qualities), simplify the nature of the task.

As observed by Carroll (2000, 2001), the introduction of new tools can change the overall nature of the task. Further, it can change the nature of the operational process (Mc Donald, 2004). Carroll's (2000, 2001) concept of the task artefact lifecycle is relevant here. New technology requirements cannot be premised on existing task practices and associated problems alone. Carroll argues that we must envisage improved task practices (or future use scenarios) and consider how technology might support this. That is, we must consider how technology might be used to transform the task. In identifying future technology requirements, the researcher must balance two task pictures. This includes the existing task performance picture and the potential new task performance picture, facilitated by the introduction of new or improved technologies.

The literature highlights the necessity of developing tools from the perspective of the full task and not in isolation. As noted by Wickens (2000), individual displays supporting part task functionality cannot be designed in a vacuum. Rather, the wider tools picture must be evaluated. Here HCI designers must consider issues related to design consistency and information integration across the range of tools used by different operators. This is no easy task, but nonetheless requires consideration.

Information behaviour theories (Wilson, 1999) illuminate a range of operator information management processes in socio-technical contexts. This includes processes related to information gathering, information interpretation, information classification and prioritisation, information communication and information use. From a methodological perspective, this suggests that HCI methodologies should allow the research to model human information behaviour and associated information management strategies, so the proposed new technologies are predicated on an appropriate task and information picture, prioritise key user information and facilitate information sharing with all relevant human and technical agents involved in the task activity. Further, it is well documented that the format in which information is presented to the user, impacts on the perception, interpretation and use of this information. Here, we need to consider the HCI aspects of information access and presentation. As such, methods should allow the researcher to properly assess such issues so that user friendly design solutions are advanced.

3. Introduction to Human Factors & Human Computer Interaction

3.1 Human Factors

The Human factors discipline arose in relation to understanding the human role in socio-technical systems. The International Ergonomics Association (IEA) defines Ergonomics (or Human Factors) as
'The scientific discipline concerned with the understanding of interactions among humans and other elements of a system, and the profession that applies theory, principles, data and methods to design in order to optimize human well-being and overall system performance' (2000).
The IEA distinguish three domains of specialization within the discipline of ergonomics: Physical ergonomics is concerned with human anatomical, anthropometric, physiological and biomechanical characteristics as they relate to physical activity. Cognitive ergonomics is concerned with mental processes, such as perception, memory, reasoning, and motor response, as they affect interactions among humans and other elements of a system. Organizational ergonomics is concerned with the optimization of socio-technical systems, including their organizational structures, policies and processes.

3.2 Human Computer Interaction

Human computer interaction is a multi-disciplinary subject focused on the design of human friendly technology. Different definitions of HCI have been provided. In certain definitions, HCI is regarded as a subset of Human Factors concentrating on the design and evaluation of technologies, while in others it described in similar terms as Human Factors. This is also complicated by the fact that the term HCI is often used interchangeably with 'Engineering Psychology', 'Cognitive Ergonomics' and 'Human Factors'. In this analysis, HCI can be considered as a sub-set of theory and methodologies within HF, concerned with the design and evaluation of technology for use in the context of both open and closed systems. In terms of the three strands of Human Factors defined above, it can be broadly classified as Cognitive Ergonomics.
According to HCI theorists and practioners, to design human friendly technology which fits the work context, we must adopt a 'user centered design' methodology/process. The HCI literature defines a range of methods for this. Collectively, these methods emphasize: (1) the necessity of involving users in design process, (2) the degree to which design is an iterative process (e.g. designs are prototyped and evaluated and the modified and evaluated again), and (3) the extent to which evaluation provides an empirical basis in which to evaluate/justify designs.

4. Software Development Process

HCI methodologies are adopted in the context of developing software/technologies following project goals and timelines. This process is referred to as the software development process. This process follows a number of high level stages including, (1) specifying user requirements, (2) specifying functional requirements, (3) application development, (4) testing, (5) trials and (6) full implementation. Critically, a range of HCI techniques are used for different purposes (e.g. specifying requirements, designing prototypes, evaluating prototypes etc) at various points in this process.
The software development process involves the specification of a number of different types of requirements. This includes user requirements, system/functional requirements, user interface design requirements and usability requirements. User requirements refer to what the system needs to do from the user's perspective (taking into account that there might be a range of different users). System requirements refer to what the system actually does (e.g. list of functions that the system performs and analysis of each function). User interface design requirements refer to what the user interface will look and feel like, and how users will interact with different system functions. Finally usability requirements refer to the acceptable level of user performance and satisfaction with the system. It is important to note that both user requirements and user interface design requirements can be defined at different levels. Depending on the level of specification (e.g. requirements stated in the form of general guidelines or visual prototypes) more or less design instruction is provided to Interface Designers and Software Developers. Evidently, both user requirements and user interface design requirements must be specified at a sufficiently concrete level so that (1) Software Developers can document the functional specification (e.g. functional requirements) linking to application development and (2) Graphic and Interaction Designers can produce the full interaction and visual design model. According to participatory design advocates, one of the weaknesses of formal HCI methods, is that typical outputs (e.g. lists of users and tasks, task analysis diagrams, task scenarios and so forth), fail to provide sufficient design guidance. This is discussed in more detail, later in this chapter.

5. HCI Design Methodological Challenges

So what should HCI design methodologies achieve? How should the range of HCI methodologies serve HCI design practioners in terms of envisioning new or improved work practices, and associated technologies in socio-technical system settings?
It is argued that HCI design methods should fulfil the following objectives:

1. The development of an appropriate task model
2. Evaluation of existing tools & information resources
3. Envisionment of new tool requirements & associated task practices
4. Understanding broader organisational and technological implications of proposed tool concepts
5. Specification and evaluation of proposed user interface for new or improved tool concepts
6. Facilitation of communication of user requirements and design concepts to Software Developers & Graphic Designers

5.1 The Development of an Appropriate Task Model

Firstly, HCI design methods should allow for the development of an appropriate task model. This involves a consideration of both existing task practice and future task practice. In relation to the former, methodologies should assist the following:

1. Modelling actual task practice, taking into account different operational and environmental contexts (e.g. ecological validity)
2. Modelling operator task activity and information flow (including collaborative work activity)
3. Modelling the relationship between task and process (e.g. design of both operational process and background organizational processes).
4. Modelling the both technical and non technical aspects of task performance
5. Modelling information flow requirements (e.g. information in an out and how this is facilitated by the use of tools and information resources (both human and technical)

In terms of future task practice, methods should allow the researcher to identify how task practice might be improved given new technology possibilities.

5.2 Evaluation of Existing Tools and Information Resources

Design methods should also allow the researcher to assess the use of existing tools and information resources. For example, task problems associated with existing technology design constraints might be ameliorated by rethinking the current design.

5.3 Envisionment of New Tool Requirements & Associated Task Processes

Proposed HCI design methods should allow for the identification of new tool requirements and associated operational processes based on the task model and evaluation of existing tools. Critically, the resulting tools should represent an improvement on the existing situation. This may involve the envisionment of new operational processes along with new task practices. As such, the design methods should allow for both process and task envisionment. Further, to mitigate problems in relation to the task artefact lifecycle (Carroll, 2000, 2001), such methods might allow the researcher to evaluate the future use situation and associated tool concepts. Importantly, new designs should not inherit the weaknesses of earlier designs, no introduce any new HCI problems.

5.4 Specification and Evaluation of Tool Concepts

Further, design methods should facilitate the prototyping of new tool concepts, thereby bridging the gap between requirements specification and design. Also methods should allow for the evaluation of tool concepts, to ensure that they are optimised in terms of user tasks and conceptual models. Further methods should allow researchers to assess whether human performance and environmental constraints are factored into the proposed HCI design solution.

5.5 Understanding Broader Organisational & Technological Implications of Proposed Tool Concepts & Associated Feasibility Issues

Also, methods should facilitate the assessment of whether or not the proposed technology requirements are achievable. First the researcher must consider feasibility in terms of existing organisational structures and roles, resource capacity and redesign requirements. To this end, methods should facilitate the identification of the task/performance requirements embedded in the proposed technology design, and whether this requires changes to existing work practices and/or resource allocations. There may be organisational barriers to changing existing work practices. For example, new tool concepts may require communication or sharing of information between different roles or departments. Certain departments may not want to share information. Further, there may be data protection issues linked to Union agreements or regulatory rules, which prohibit data sharing. In addition, new tool functionality (e.g. the provision of task support information customised to an operation) may require additional work effort/human resources. Does the organisation have the capacity for this? Can new functions be incorporated in existing roles, or are new employees required? What are the training implications?

The design of technology for use by individual groups of operators in socio-technical contexts often links to the design of technologies used by other roles in the organisation. In particular, the provision of information to specific end users, related to collaborative work activities necessitates information sharing across the different human and technical agents involved in the work activity. As such, methodologies should assist researchers in identifying the relevant information integration requirements inherent in the proposed technology concepts. Further, proposed methods should permit the identification of any additional technology requirements linked to the task performance of other operational roles, which may or may not be supported by existing tools.

5.6 Facilitate Communication of User Requirements and Design Concepts to Software Developers & Graphic Designers

Lastly, it is critical that the analysis and design outputs of the HCI design methodologies adopted, can be utilized by Software Developers, in terms of specifying the functional requirements of the system. Moreover, the outputs need to be instructive in terms of specifying the user interface design requirements for the proposed system, which is managed by the Graphic Design team.

6. Overview of HCI Methods

Do HCI methodologies facilitate the above objectives? Or, are other methods required? The literature distinguishes two high level sets of methods, namely formal and informal HCI methods. In general, formal methods are characterized as being closer to scientific methods. Alternatively, informal methods are strongly linked to design activities and considered to have a more qualitative focus

6.1 Formal HCI Methods

Formal methods in HCI allow for user involvement at specific points in the software development lifecycle (Nielsen, 1993, Constantine et al, 1999, Mayhew, 1999).

First, a task analysis is conducted, to understand how the operator interacts with the existing system and to identify the user requirements for an improved or new system. According to Kirwan and Ainsworth (1992), a task analysis *'is a method of describing what an operator is required to do, in terms of actions or cognitive processes, to achieve a system goal'*. Usually, this occurs at the beginning of a project and takes the form of structured or semi-

structured interviews focused on understanding and evaluating current work practices and supporting technology (Hackos et al, 1998). This follows ethnographic research methods, advanced in the Social Science field (e.g. Interviews and Observation).
A number of analysis and design steps are then completed by HCI professionals without the participation of end users. These techniques aim to represent the cognition, practice or logic of the task. In addition, they aim to identify user requirements. Typical analysis methods include content analysis, hierarchical task analysis and task workflow analysis.
Design concepts are then modelled with the help of Graphic Designers. This involves mapping user tasks and workflows to a set of interface screens with a defined information structure and presentation logic. Initially, HCI designers might map a high level storyboard. Following this, a more detailed storyboard is modelled. Detailed storyboards include rough sketches of screen layouts and designs that correspond to the use sequence outlined for a detailed level of a task performance by a system. This process is supported by a wealth of advisory information relating to user interface design. This includes International Organisation for Standardisation (ISO) user interface design approaches and standards (ISO, 1995, 1997), and usability design principles/heuristics (Nielsen, 1993, Preece et al, 2002).
Following this, prototypes are modelled. Developing prototypes is a central part of user centred design. A prototype is an experimental or incomplete design. This links to the distinction between specification and implementation. A prototype belongs to specification/design phase, as opposed to the implementation phase. Different kinds of prototyping are appropriate for different stages of design. Once the prototypes are completed, user workflows and interface features/behaviours are evaluated. In HCI design, evaluation is part of the design process. Evaluation is part of the design process. Feedback is obtained about the usability of designs via inspection, testing or enquiry. This is an iterative process. Evaluation occurs at different points in the development process. The goals of evaluation are multiple and varied. Evaluation can be used to investigate what users want, if user requirements are being met and what problems users have. Further, it can be used to test out design ideas/concepts quickly and to assess the usability of a UI and improve the quality of the UI. Two main evaluation methods are used. This includes (1) user testing and (2) heuristic evaluation. User testing involves the assessment of a user interface (UI) by observing representative users performing representative tasks using the UI (Rubin, 1994). This is used to identify any aspects of a design that cause users difficulty, confusion, or misunderstandings. These may lead to errors, delays, or in extreme cases inability to complete the tasks for which the UI is designed. User testing also provides insight into user preferences. In addition, a heuristic evaluation may be conducted. In a Heuristic Evaluation, the UI is examined against a set of agreed usability /user experience principles (the heuristics). This is undertaken by a team of experienced usability professionals (the evaluators). As such, the evaluation does not involve end users. The evaluator or team of evaluators step into the shoes of the prospective end user – taking into account their profile, mental models of the task, typical learning styles and task requirements. Following iterative prototyping and evaluation, high fidelity prototypes are developed by software developers.

6.2 Informal HCI Methods

Formal HCI methods have been the subject of much debate in the HCI literature. Specific challenges have come from the fields of Ethnography and Participatory Design.

Ethnographers argue that classical HCI methods do not take work practice seriously; failing to address the social aspects of work (Hutchins 1995, Vicente 1999). In particular, they argue that user interviews cannot provide actual insight into real work practices. Participatory design theorists have questioned the separation between design and evaluation in formal methods (Bødker and Buur, 2002). Specifically, they have challenged the instructiveness of traditional user and task analysis outputs for design guidance. Also, they argue that user testing provides insufficient information concerning user problems. Further, PD theorists have questioned the usefulness of these methods for the design of both socio-technical systems and ubiquitous technology (Bodker and Buur, 2002).
The field of participatory design originated in Scandinavia in the early 1970s, in response to union mandates that workers should be involved in the design of new workplace technology. This heralded the introduction of new HCI methodologies, many of which were pioneered in the Utopia Project (Bødker, 1985). Central to PD theory is the idea that usability engineers design 'with' end users, as opposed to 'for' them. Accordingly, users are active participants in the design process, and the traditional HCI design team (e.g. Usability Engineers and Graphic Designers) is broadened to include end users (workers and worker organizations), stakeholders and domain experts. Crucially, PD theory stresses the relationship between design and evaluation. PD theorists argue that to design effective work tools, design teams must first experience and evaluate future technology and practices (Bannon, 1991, Muller 2003). As such, PD techniques (such as, the co-creation and evaluation of prototypes and scenario role playing), allow design teams to envision and evaluate future workplace practices and related technologies, without the constraints of current practice. Overall PD techniques have been adapted from Ethnography. This includes concept generation, envisionment exercises, story collecting and story telling (through text, photography and drama), games of analysis and design and the co-creation of descriptive and functional prototypes.
The PD contention that users must be active participants in the design process, (and related argument that Usability Engineers should be receptive to user's own ideas and explanatory frameworks) reflects certain underlying phenomenological conceptions of knowledge. Participants are not objects but partners or 'experts' whose ideas are sought. Thus, it is inappropriate for human factors researchers to formulate design models in advance of collaboration with end users. In this respect, PD theorists argue that there are four dimensions along which participation could be measured. This includes: (1) the directness of interaction with the designers, (2) the length of involvement in the design process, (3) the scope of participation in the overall system being designed and (4) the degree of control over design decisions.
Critical to PD methodology is the envisionment of future work situations. According to PD theorists, users need to have the experience of being in the future use situation, or an approximation of it, in order to be able to comment on the advantages and disadvantages of the proposed system. As argued by Bannon, some form of mock-up or prototype needs to be built in order to let users know what the future use situation might be (1991). This allows users to experience how emerging designs may affect work practice.
Carroll proposes a scenario based design approach (2000, 2001). This links to the development of persona's and task scenarios, used in formal HCI approaches. This approach distinguishes the development of existing task scenarios (describing current practice), and future task scenarios (or future use scenarios). According to Carroll, future

use scenarios are narrative descriptions of a future task state. This relates to the participatory design techniques of imagining future work processes and supporting technology (described above). Further, it relates to Carroll's concept of the task artefact lifecycle. For Carroll, the task artefact cycle is the background pattern in technology development (2000, 2001). Possible courses of design and development must be envisioned and evaluated in terms of their impacts on human activity (before they are pursued). If – If Designers model technology in terms of the existing task practice (e.g. model what is), the technology will be one step behind (Carroll, 2000, 2001).
Further, the application of participant observation methods developed in the Social Science field, have also been proposed. The purpose here is to obtain a picture of real world task practices and associated environmental constraints. This is based on the idea that participant feedback in interviews (used in formal methods) may not provide a true or accurate picture of the actual work reality. These methods have been supported by Hutchins. According to Hutchins, it is through Ethnography that we gain knowledge about how a distributed system actually works (1995).

7. Overview of Methods Used in Organisational Ergonomics Fields

HCI methods are influenced and/or have much in common with specific work analysis methods used in the organisational ergonomics domain. This includes Process Mapping and Cognitive Work Analysis.

7.1 Process Mapping

The objective of process mapping is to model the current process and identify process re-design requirements for the purpose of improving safety, or productivity. This relates to business process modelling (e.g. modelling 'as is' and 'future processes'), with the objective of improving efficiency and quality. Process mapping involves the production of a diagrammatic representation of the overall process, and associated sub-processes. Specifically it represents the sequential and parallel task activities of both human and technical agents, which collectively result in the achievement of the operational goal. This approach originates in the research of Gilbreth and Gilbreth (1921). Underlying this visual map is an analysis of the process as a functional system (e.g. transformation of inputs into outputs, process dependencies), as a social system (e.g. team performance requirements, co-ordination and communication mechanisms) and as an information system (e.g. transformation of information across different technical and human resources). Typically process mapping is conducted in a workshop format involving all relevant stakeholders involved in the operational process. First, the researcher reviews the high level process and then drills down to chart the related task activities of different roles. As part of this, there is usually some form of trouble-shooting related to identifying existing process problems and redesign solutions.

7.2 Cognitive Work Analysis

Vicente argues that in dynamic work settings, there are many factors outside the individual affecting their interactions with computer systems and these factors must be considered in the design of such systems (1999). In this regard, Vicente contends that to understand work demands both cognitive and environmental constraints must be considered. Vicente methodology is based on Rasmussen's argument that the work environment determines to a large extent the operator constraints and the ability of the operator to choose his/her own strategy. In Vicente's view, environmental constraints come first (Vicente, 1999). To this end, Vicente (1999) proposes a cognitive work analysis (CWA) methodology to analyse work. This includes both task and work domain analysis. CWA consists of five concepts and corresponding analysis. This includes, (1) an analysis of the boundaries and restrictions of the work domain, (2) an analysis of the information processing parts of the task, (3) an analysis of the process and associated task performance, (4) an analysis of social organisational and co-ordination and (5), an analysis of worker competencies. This methodology has been applied to diverse work situations involving varying degrees of process control/automation.

8. Analysis

Operator work in socio-technical contexts can be quite complex. Often it involves the performance of collaborative activities with a range of human and technical agents. As such, task activity and human computer interaction in open systems is more demanding than in closed systems. It is argued that (1) the modelling of task activity and (2) the envisionment, design and evaluation of improved task support tools in socio-technical contexts, necessitates the application of a range of design methods, above and beyond what is outlined in the HCI literature (e.g. both formal and informal HCI methods). Taking into account the methodological requirements outlined earlier, it is suggested that HCI researchers adopt a mix of methodologies associated with two of the three Human Factors fields, namely Cognitive Ergonomics and Organisational Ergonomics. Specifically, an integration of formal HCI methods, informal HCI methods and both process mapping and cognitive work analysis methods is proposed. Typically, HCI practioners working in socio-technical settings use a range of both formal and informal HCI methods. Further, certain work analysis techniques such as Cognitive Work Analysis have been applied by HCI practioners. Other methods such as process mapping methods have not been used.
Existing HCI methods do not support an analysis of the relationship between task, process and technology requirements. Specifically, to design 'operational' technologies, HCI researchers must understand how existing and future technologies relate to the design of the existing process and/or future process. The introduction of new technologies has implications for broader task practice (e.g. task practice of other agents) and the design of the operational process. As such, we cannot just think of technology from the perspective of the task performance of one role, or in isolation from the broader process design. To this end, in analysing task performance, we must distinguish two perspectives on task activity – (1) the specific user perspective and (2) the broader system perspective (e.g. takes into account the broader operational and organisational aspects of task performance). Insofar as both perspectives relate, this is not a real distinction. However, this distinction is useful from an analytic perspective. The individual perspective focuses on task performance in terms of unique roles. Here we consider the overall task picture, how tasks relate, actual task workflows (e.g. difference between SOP and actual practice), task information requirements, use of tools and environmental constraints. Critically, this perspective prioritises the task requirements of the individual operator. The system perspective investigates task performance on two other levels – the operational level and the

organisational level. The operational level takes into account collaboration with other roles and associated task information inputs and outputs. As such, it reflects a process perspective on task activity – factoring team collaboration requirements into task models. This links to the computer supported cooperative work frameworks proposed by Bannon (1991) and others. The organisational level examines task performance in terms of those processes in the organisation that support task performance. For example, training, safety management and procedures design. Process mapping workshops can be used to model the existing operational process and envisage future processes and associated re-design requirements. Also, interviews and observations can be used to map relevant work processes.

Formal HCI design methods do not support the envisionment of future work practices and associated technology requirements. To this end, informal HCI methods are required. It is argued that participatory design methods facilitate technology envisionment and provide concrete design instruction. Collaborative prototyping of proposed tool concepts with end users allows both the researcher and participants to envision future use scenarios and associated technology requirements. Further, these techniques enable practioners to elicit feedback relating to the usability of future technology concepts - thereby circumventing the task artefact lifecycle. Crucially, the application of these methods results in the advancement of meaningful requirements. User requirements and associated interface concepts are translated into actual user interface features and behaviours. Prototypes can be used as a basis for exploring, evaluating and communicating design ideas. Indeed, it is difficult for participants to fully envisage and evaluate design ideas, without such prototypes. Essentially, techniques allow both users and designers to experiment with different visual/interactive affordances (e.g. menu structures, icons, presentation of form fields) until a design consensus is reached. Further, certain visual and interaction issues require 'hands-on' problem solving. In this way, research does not stop short of concrete solutions. However, as a stand-alone methodology, participatory research methods are insufficient. To design tools that improve upon current practice, we must start from current practice. To interpret and weight participant opinions related to specific design solutions, the researcher must be familiar with the existing problem space. As such, naturalistic research methods (e.g. interviews and observations) are a necessary precursor to PD methods.

Both HCI methods and organisational analysis methods do not facilitate the identification of the broader organisational and technological implications of new tool concepts. It is suggested that process workshop methods are adapted to this purpose. The specific methodology for this is outlined in subsequent sections.

9. Proposed Methodological Approach & Case Study Examples

The proposed integrated HCI design methodology can be grouped into a series of design steps at different points in the user centred design process. The specific steps proposed relate to HCI research only. Some of these steps are required, while others are optional. Further, certain steps depend on the project context. It is recommended that practioners adopt this methodology for their own purpose, taking into account relevant project considerations. Other work, relevant to the performance requirements of the wider software development team is alluded to in terms of dependencies with HCI work, but not described in terms of actual steps. This includes the production of the graphic design, the specification of functional requirements, software development, software testing, software and hardware integration and testing, trial implementation of proposed systems and full implementation of proposed systems.

Step	Description	Required, Context Dependent Or Optional	Methods	Output
1	Literature Review	Optional	Review literature available – comparative tools, known problems	Report
2	Identifying the process context underlying operator task performance	Required	Process mapping workshops (existing process) Follow up observation of work practice, or interviews with different stakeholders	Process map of existing process Process analysis templates Role/task descriptions
3	Modelling existing task practice and tool usage	Required	Observation of work practice Interviews with different operational roles User testing User and task analysis	User/task matrices Task scenarios Procedural workflow diagrams User testing report
4	Specification of preliminary user requirements	Depends on project context	Advancement of future use scenarios and associated technology brief Analysis and documentation of requirements	Future use scenarios Preliminary user requirements specification Prototypes (optional)
Management review and decisions				
5	Envisioning new work practices and associated user requirements for new or improved technologies	Required	Process workshops (future process) Technology envisionment exercises Role play Collaborative prototyping	Future / To be process map Future use scenarios High level tool concepts High level paper or MS Visio prototypes
6	Prototyping and evaluating of proposed tool concepts	Required	Mix of individual and collaborative prototyping User testing	Prototypes (MS Visio Prototypes)
7	Dry run implementation of proposed tool concepts to assess organisational and technological implications	Required	Review of proposed scenarios and prototypes, as part of an implementation workshop	Prototypes Implementation Report

Management review and decisions				
Step	**Description**	**Required, Context Dependent Or Optional**	**Methods**	**Output**
8	Further prototyping and evaluation	Depends on previous – scope of changes	Further prototyping and evaluation (if required)	Prototypes
9	Overall Research Analysis, Further Prototyping & Specification of User Requirements and User Interface Design	Required	Analysis and weighting of all feedback Further prototyping Specification of user requirements	User requirement specification User interface design specification Prototypes Updated process map
10	Handover to Software Developers & Graphic Designers	Required	In person review session – review proposed tool prototypes and relevant documentation	User requirement specification Prototypes
Production of graphic design				
Definition of functional specification				
Initial software development				
Handover of graphic design to software developers				
Further software development				
11	Review Software Prototypes	Required	In person review session (ongoing)	Updated software prototypes
12	Evaluation of High Fidelity Prototypes	Optional	User testing Heuristic Evaluation	Updated software prototypes
13	Tool Certification (ongoing once tool concepts defined)	Depends on project context	Review of regulatory guidance Evaluation with authorities	Certification report
Software testing				
Integration with other software systems and hardware				
Integration Testing				
Trial implementation of new systems in organisation				
Full implementation of new systems in organisation				
14	Ongoing feedback and improvements (after go live)	Optional	Observation of work practice using new tools Interviews with different operational roles Surveys	Implementation report

Table 1. Summary of Proposed HCI Methodology & Associated Steps, Methods and Outputs

Such method triangulation has been used in two different studies conducted by the author. Each of these studies has involved the application of some or most of the design steps outlined above. It should be noted that one of these studies is complete while the other is ongoing. Before presenting the proposed the design steps for the integrated HCI design