

Towards a Conception for an Engineering Discipline of Human Factors

John Dowell and John Long

*Ergonomics Unit, University College London,
26, Bedford Way, London. WC1H 0AP.*

abstract

This paper concerns one possible response of Human Factors to the need for better user-interactions of computer-based systems. The paper is in two parts. Part I examines the potential for Human Factors to formulate *engineering principles*. A basic pre-requisite for realising that potential is a conception of the *general design problem* addressed by Human Factors. The problem is expressed informally as: 'to design human interactions with computers for effective working'. A *conception* would provide the set of related concepts which both expressed the general design problem more formally, and which might be embodied in engineering principles. Part II of the paper proposes such a conception and illustrates its concepts. It is offered as an initial and speculative step towards a conception for an engineering discipline of Human Factors.

Part I. Requirement for Human Factors as an Engineering Discipline of Human-Computer Interaction

1.1	Introduction	2
1.2.	Characterisation of the Human Factors Discipline	3
1.3.	State of the Human Factors Art.....	5
1.4.	Human Factors Engineering Principles	7
1.5.	The Requirement for an Engineering Conception for Human Factors	11

1.1. Introduction

Advances in computer technology continue to raise expectations for the effectiveness of its applications. No longer is it sufficient for computer-based systems simply 'to work', but rather, their contribution to the success of the organisations utilising them is now under scrutiny (Didner, 1988). Consequently, views of organisational effectiveness must be extended to take account of the (often unacceptable) demands made on people interacting with computers to perform work, and the needs of those people. Any technical support for such views must be similarly extended (Cooley, 1980).

With recognition of the importance of 'human-computer interactions' as a determinant of effectiveness (Long, Hammond, Barnard, and Morton, 1983), Cognitive Ergonomics is emerging as a new and specialist activity of Ergonomics or Human Factors (HF). Throughout this paper, HF is to be understood as a discipline which includes Cognitive Ergonomics, but only as it addresses human-computer interactions. This usage is contrasted with HF as a discipline which more generally addresses human-machine interactions.

HF seeks to support the development of more effective computer-based systems. However, it has yet to prove itself in this respect, and moreover, the adequacy of the HF response to the need for better human-computer interactions is of concern. For it continues to be the case that interactions result from relatively *ad hoc* design activities to which may be attributed, at least in part, the frequent ineffectiveness of systems (Thimbleby, 1984).

This paper is concerned to develop one possible response of HF to the need for better human-computer interactions. It is in two parts. Part I examines the potential for HF to formulate HF engineering principles for supporting its better response. Pre-requisite to the realisation of that potential, it concludes, is a conception of the *general design problem* it addresses. Part II of the paper is a proposal for such a conception.

The structure of the paper is as follows. Part I first presents a characterisation of HF (Section 1.2) with regard to: the general design problem it addresses; its practices providing solutions to that problem; and its knowledge supporting those practices. The characterisation identifies the relations of HF with Software Engineering (SE) and with the super-ordinate discipline of Human-Computer Interaction (HCI). The characterisation supports both the assessment of contemporary HF and the arguments for the requirement of an engineering HF discipline.

Assessment of contemporary HF (Section 1.3.) concludes that its practices are predominantly those of a craft. Shortcomings of those practices are exposed which indict the absence of support from appropriate formal discipline knowledge. This absence prompts the question as to what might be the

formal knowledge which HF could develop, and what might be the process of its formulation. By comparing the HF general design problem with other, better understood, general design problems, and by identifying the formal knowledge possessed by the corresponding disciplines, the potential for HF engineering principles is suggested (Section 1.4.).

However, a pre-requisite for the formulation of any engineering principle is a conception. A *conception* is a unitary (and consensus) view of a general design problem; its power lies in the coherence and completeness of its definition of the concepts which can express that problem. Engineering principles are articulated in terms of those concepts. Hence, the requirement for a conception for the HF discipline is concluded (Section 1.5.).

If HF is to be a discipline of the superordinate discipline of HCI, then the origin of a 'conception for HF' needs to be in a conception for the discipline of HCI itself. A conception (at least in form) as might be assumed by an engineering HCI discipline has been previously proposed (Dowell and Long, 1988a). It supports the conception for HF as an engineering discipline of HCI presented in Part II.

1.2. Characterisation of the Human Factors Discipline

HF seeks to support systems development through the systematic and reasoned design of human-computer interactions. As an endeavour, however, HF is still in its infancy, seeking to establish its identity and its proper contribution to systems development. For example, there is little consensus on how the role of HF in systems development is, or should be, configured with the role of SE (Walsh, Lim, Long, and Carver, 1988). A characterisation of the HF discipline is needed to clarify our understanding of both its current form and any conceivable future form. A framework supporting such a characterisation is summarised below (following Long and Dowell, 1989).

Most definitions of disciplines assume three primary characteristics: a general problem; practices, providing solutions to that problem; and knowledge, supporting those practices. This characterisation presupposes classes of general problem corresponding with types of discipline. For example, one class of general problem is that of the general *design* problem¹ and includes the design of artefacts (of bridges, for example) and the design of 'states of the world' (of public administration, for example). Engineering and craft disciplines address general design problems.

Further consideration also suggests that any general problem has the necessary property of a scope, delimiting the province of concern of the associated discipline. Hence many disciplines also be distinguished from each other; for example, the engineering disciplines of Electrical and Mechanical Engineering are distinguished by their respective scopes of electrical and mechanical artefacts. So, knowledge possessed by Electrical Engineering supports its practices solving the general design problem of designing electrical artefacts (for example, Kirchoff's Laws would support the analysis of branch currents for a given network design for an amplifier's power supply).

Although rudimentary, this framework can be used to provide a characterisation of the HF discipline. It also allows a distinction to be made between the disciplines of HF and SE. First, however, it is required that the super-ordinate discipline of HCI be postulated. Thus, HCI is a discipline addressing a general design problem expressed informally as:

'to design human-computer interactions for effective working'.

The scope of the HCI general design problem includes: humans, both as individuals, as groups, and as social organisations; computers, both as programmable machines, stand-alone and networked, and as functionally embedded devices within machines; and work, both with regard to individuals and the organisations in which it occurs (Long, 1989). For example, the general design problem of HCI

¹They are to be distinguished from the class of general scientific problem of the explanation and prediction of phenomena.

includes the problems of designing the effective use of navigation systems by aircrew on flight-decks, and the effective use of wordprocessors by secretaries in offices.

The general design problem of HCI can be decomposed into two general design problems, each having a particular scope. Whilst subsumed within the general design problem of HCI, these two general design problems are expressed informally as:

'to design human interactions with computers for effective working'; and

'to design computer interactions with humans for effective working'.

Each general design problem can be associated with a different discipline of the superordinate discipline of HCI. HF addresses the former, SE addresses the latter. With different - though complementary - aims, both disciplines address the design of human-computer interactions for effective working. The HF discipline concerns the physical and mental aspects of the human interacting with the computer. The SE discipline concerns the physical and software aspects of the computer interacting with the human.

The *practices* of HF and SE are the activities providing solutions to their respective general design problems and are supported by their respective *discipline knowledge*. Figure 1 shows schematically this characterisation of HF as a sub-discipline of HCI (following Long and Dowell, 1989). The following section employs the characterisation to evaluate contemporary HF.

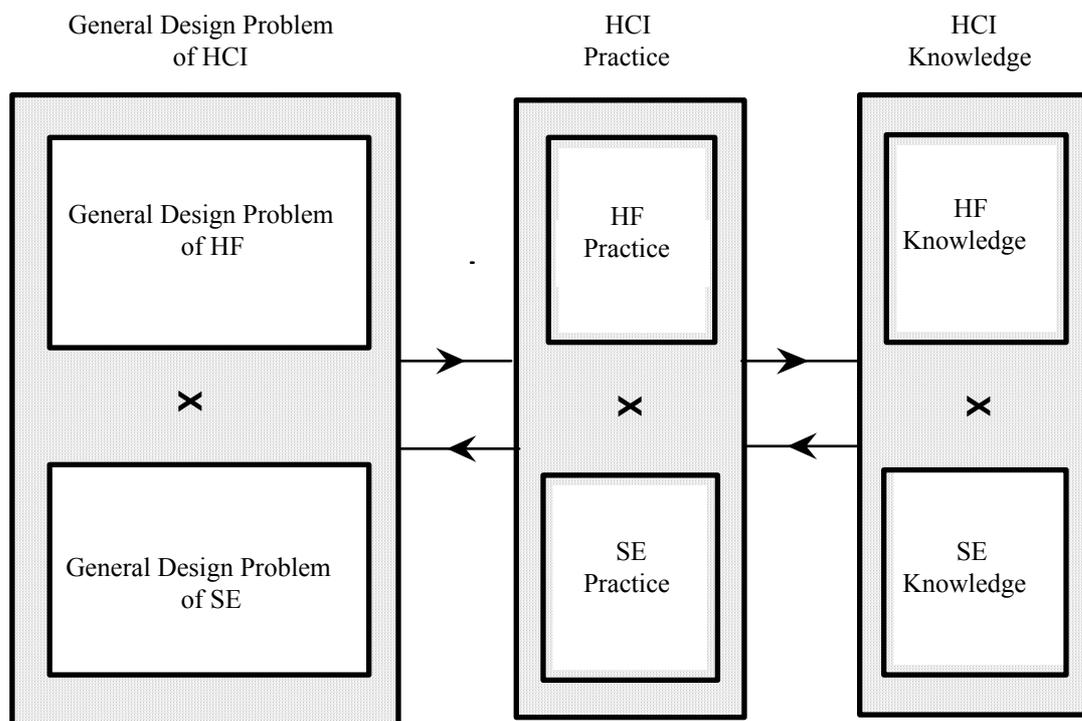


Figure 1. A characterisation of HF as a sub-discipline of HCI (following Long and Dowell, 1989).

1.3. State of the Human Factors Art

It would be difficult to reject the claim that the contemporary HF discipline has the character of a craft (at times even of a technocratic art). Its practices can justifiably be described as a highly refined form of design by 'trial and error' (Long and Dowell, 1989). Characteristic of a craft, the execution and success of its practices in systems development depends principally on the expertise, guided intuition and accumulated experience which the practitioner brings to bear on the design problem¹.

It is also claimed that HF will always be a craft: that ultimately only the mind itself has the capability for reasoning about mental states, and for solving the under-specified and complex problem of designing user-interactions (see Carey, 1989); that only the designer's mind can usefully infer the motivations underlying purposeful human behaviour, or make subjective assessments of the elegance or aesthetics of a computer interface (Bornat and Thimbleby, 1989).

The dogma of HF as necessarily a craft whose knowledge may only be the accrued experience of its practitioners, is nowhere presented rationally. Notions of the indeterminism, or the un-predictability of human behaviour are raised simply as a gesture. Since the dogma has support, it needs to be challenged to establish the extent to which it is correct, or to which it compels a misguided and counter-productive doctrine (see also, Carroll and Campbell, 1986).

Current HF practices exhibit four primary deficiencies which prompt the need to identify alternative forms for HF. First, HF practices are in general poorly *integrated* into systems development practices, nullifying the influence they might otherwise exert. Developers make implicit and explicit decisions with implications for user-interactions throughout the development process, typically without involving HF specialists. At an early stage of design, HF may offer only advice - advice which may all too easily be ignored and so not implemented. Its main contribution to the development of user-interactive systems is the evaluations it provides. Yet these are too often relegated to the closing stages of development programmes, where they can only suggest minor enhancements to completed designs because of the prohibitive costs of even modest re-implementations (Walsh et al,1988).

Second, HF practices have a suspect *efficacy*. Their contribution to improving product quality in any instance remains highly variable. Because there is no guarantee that experience of one development programme is appropriate or complete in its recruitment to another, re-application of that experience cannot be assured of repeated success (Long and Dowell, 1989).

Third, HF practices are *inefficient*. Each development of a system requires the solving of new problems by implementation then testing. There is no formal structure within which experience accumulated in the successful development of previous systems can be recruited to support solutions to the new problems, except through the memory and intuitions of the designer. These may not be shared by others, except indirectly (for example, through the formulation of heuristics), and so experience may be lost and may have to be re-acquired (Long and Dowell, 1989).

¹The claimed craft status of HF practice remains unaffected by the counterclaim that science and, in particular, psychology, offers guidance to the designer. The guidance may be direct - by the designer's familiarity with psychological theory and practice, or may be indirect by means of guidelines derived from psychological findings. In both cases, the guidance can offer only advice which must be implemented then tested to assess its effectiveness. Since the general scientific problem is the explanation and prediction of phenomena, and not the design of artifacts, the guidance cannot be directly embodied in design specifications which offer a guarantee with respect to the effectiveness of the implemented design. It is not being claimed here that the application of psychology directly or indirectly cannot contribute to better practice or to better designs, only that a practice supported in such a manner remains a craft, because its practice is by implementation then test, that is, by trial and error (see also Long and Dowell, 1989).

Fourth, there are insufficient signs of systematic and intentional *progress* which will alleviate the three deficiencies of HF practices cited above. The lack of progress is particularly noticeable when HF is compared with the similarly nascent discipline of SE (Gries, 1981; Morgan, Shorter and Tainsh, 1988).

These four deficiencies are endemic to the craft nature of contemporary HF practice. They indict the tacit HF discipline knowledge consisting of accumulated experience embodied in procedures, even where that experience has been influenced by guidance offered by the science of psychology (see earlier footnote). Because the knowledge is tacit (i.e., implicit or informal), it cannot be operationalised, and hence the role of HF in systems development cannot be planned as would be necessary for the proper integration of the knowledge. Without being operationalised, its knowledge cannot be *tested*, and so the efficacy of the practices it supports cannot be guaranteed. Without being tested, its knowledge cannot be *generalised* for new applications and so the practices it can support will be inefficient. Without being operationalised, testable, and general, the knowledge cannot be *developed* in any structured way as required for supporting the systematic and intentional progress of the HF discipline.

It would be incorrect to assume the current absence of formality of HF knowledge to be a necessary response to the indeterminism of human behaviour. Both tacit discipline knowledge and 'trial and error' practices may simply be symptomatic of the early stage of development of the discipline¹. The extent to which human behaviour is deterministic for the purposes of designing interactive computer-based systems needs to be independently established. Only then might it be known if HF discipline knowledge could be formal. Section 1.4. considers what form that knowledge might take, and Section 1.5. considers what might be the process of its formulation.

1.4. Human Factors Engineering Principles

HF has been viewed earlier (Section 1.2.) as comparable to other disciplines which address general design problems: for example, Civil Engineering and Health Administration. The nature of the formal knowledge of a future HF discipline might, then, be suggested by examining such disciplines. The general design problems of different disciplines, however, must first be related to their characteristic practices, in order to relate the knowledge supporting those practices. The establishment of this relationship follows.

The 'design' disciplines are ranged according to the 'hardness' or 'softness' of their respective general design problems. 'Hard' and 'soft' may have various meanings in this context. For example, hard design problems may be understood as those which include criteria for their 'optimal' solution (Checkland, 1981). In contrast, soft design problems are those which do not include such criteria. Any solution is assessed as 'better or worse' relative to other solutions. Alternatively, the hardness of a problem may be distinguished by its level of description, or the formality of the knowledge available for its specification (Carroll and Campbell, 1986). However, here hard and soft problems will be generally distinguished by their *determinism for the purpose*, that is, by the need for design solutions to be determinate. In this distinction between problems is implicated: the proliferation of variables expressed in a problem and their relations; the changes of variables and their relations, both with regard to their values and their number; and more generally, complexity, where it includes factors other than those identified. The variables implicated in the HF general design problem are principally those of human behaviours and structures.

A discipline's practices construct solutions to its general design problem. Consideration of disciplines indicates much variation in their use of specification as a practice in constructing solutions.

¹ Such was the history of many disciplines: the origin of modern day Production Engineering, for example, was a nineteenth century set of craft practices and tacit knowledge.

This variation, however, appears not to be dependent on variations in the hardness of the general design problems. Rather, disciplines appear to differ in the *completeness* with which they specify solutions to their respective general design problems before implementation occurs. At one extreme, some disciplines specify solutions completely before implementation: their practices may be described as 'specify then implement' (an example might be Electrical Engineering). At the other extreme, disciplines appear not to specify their solutions at all before implementing them: their practices may be described as 'implement and test' (an example might be Graphic Design). Other disciplines, such as SE, appear characteristically to specify solutions partially before implementing them: their practices may be described as 'specify *and* implement'. 'Specify then Implement', therefore, and 'implement and test', would appear to represent the extremes of a dimension by which disciplines may be distinguished by their practices. It is a dimension of the completeness with which they specify design solutions.

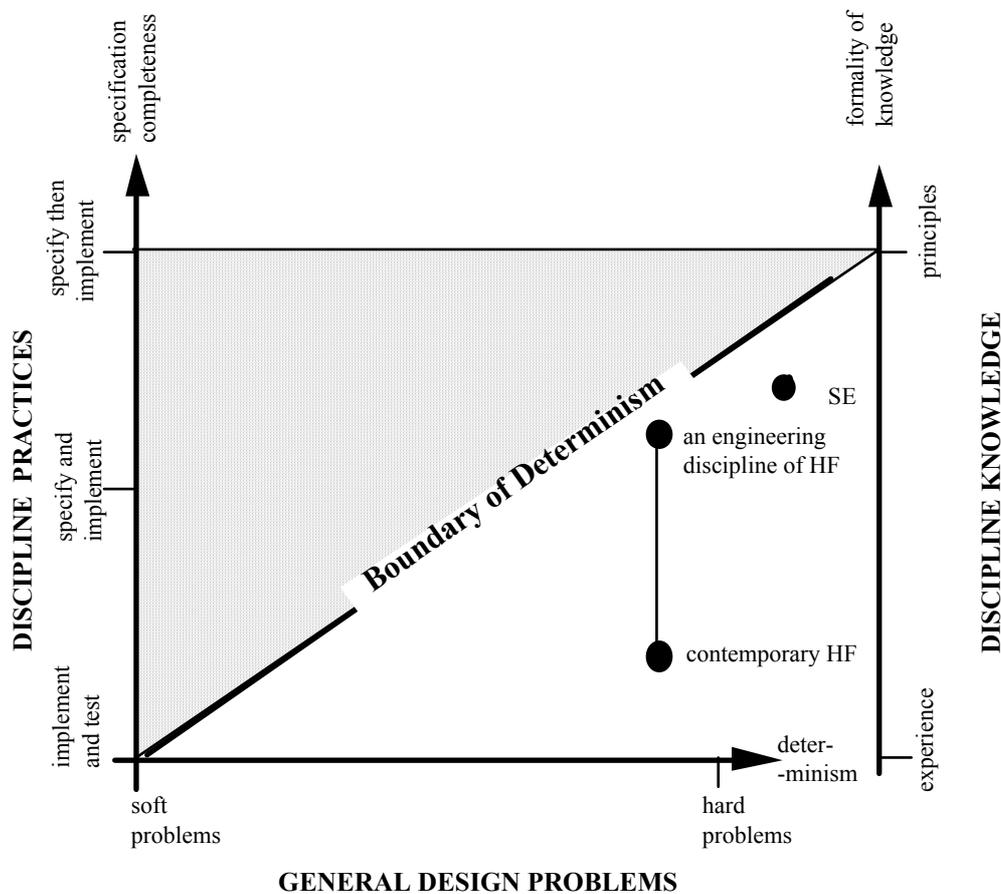


Figure 2. A Classification Space for 'Design' Disciplines

Taken together, the dimension of problem hardness, characterising general design problems, and the dimension of specification completeness, characterising discipline practices, constitute a classification space for design disciplines such as Electrical Engineering and Graphic Design. The space is shown in Figure 2, including for illustrative purposes, the speculative location of SE.

Two conclusions are prompted by Figure 2. First, a general relation may be apparent between the hardness of a general design problem and the *realisable* completeness with which its solutions might be specified. In particular, a boundary condition is likely to be present beyond which more complete solutions could not be specified for a problem of given hardness. The shaded area of Figure 2 is intended to indicate this condition, termed the 'Boundary of Determinism' - because it derives from the determinism of the phenomena implicated in the general design problem. It suggests that whilst

very soft problems may only be solved by 'implement and test' practices, hard problems may be solved by 'specify then implement' practices.

Second, it is concluded from Figure 2 that the *actual* completeness with which solutions to a general design problem are specified, and the realiseable completeness, might be at variance. Accordingly, there may be different possible forms of the same discipline - each form addressing the same problem but with characteristically different practices. With reference to HF then, the contemporary discipline, a craft, will characteristically solve the HF general design problem mainly by 'implementation and testing'. If solutions are specified at all, they will be incomplete before being implemented. Yet depending on the hardness of the HF general design problem, the realiseable completeness of specified solutions may be greater and a future form of the discipline, with practices more characteristically those of 'specify then implement', may be possible. For illustrative purposes, those different forms of the HF discipline are located speculatively in the figure.

Whilst the realiseable completeness with which a discipline may specify design solutions is governed by the hardness of the general design problem, the actual completeness with which it does so is governed by the formality of the knowledge it possesses. Consideration of the traditional engineering disciplines supports this assertion. Their modern-day practices are characteristically those of 'specify then implement', yet historically, their antecedents were 'specify and implement' practices, and earlier still - 'implement and test' practices. For example, the early steam engine preceded formal knowledge of thermodynamics and was constructed by 'implementation and testing'. Yet designs of thermodynamic machines are now relatively completely specified before being implemented, a practice supported by formal knowledge. Such progress then, has been marked by the increasing formality of knowledge. It is also in spite of the increasing complexity of new technology - an increase which might only have served to make the general design problem more soft, and the boundary of determinism more constraining. The dimension of the formality of a discipline's knowledge - ranging from experience to principles, is shown in Figure 2 and completes the classification space for design disciplines.

It should be clear from Figure 2 that there exists no pre-ordained relationship between the formality of a discipline's knowledge and the hardness of its general design problem. In particular, the practices of a (craft) discipline supported by experience - that is, by informal knowledge - may address a hard problem. But also, within the boundary of determinism, that discipline could acquire formal knowledge to support specification as a design practice.

In Section 1.3, four deficiencies of the contemporary HF discipline were identified. The absence of formal discipline knowledge was proposed to account for these deficiencies. The present section has been concerned to examine the potential for HF to develop a more formal discipline knowledge. The potential would appear to be governed by the hardness of the HF general design problem, that is, by the determinism of the human behaviours which it implicates, at least with respect to any solution of that problem. And clearly, human behaviour is, in some respects and to some degree, deterministic. For example, drivers' behaviour on the roads is determined, at least within the limits required by a particular design solution, by traffic system protocols. A training syllabus determines, within the limits required by a particular solution, the behaviour of the trainees - both in terms of learning strategies and the level of training required. Hence, formal HF knowledge is to some degree attainable. At the very least, it cannot be excluded that the model for that formal knowledge is the knowledge possessed by the established engineering disciplines.

Generally, the established engineering disciplines possess formal knowledge: a corpus of operationalised, tested, and generalised principles. Those principles are prescriptive, enabling the complete specification of design solutions before those designs are implemented (see Dowell and Long, 1988b). This theme of *prescription* in design is central to the thesis offered here.

Engineering principles can be substantive or methodological (see Checkland, 1981; Pirsig, 1974). *Methodological Principles* prescribe the methods for solving a general design problem optimally. For example, methodological principles might prescribe the representations of designs specified at a general level of description and procedures for systematically decomposing those representations

until complete specification is possible at a level of description of immediate design implementation (Hubka, Andreason and Eder, 1988). Methodological principles would assure each lower level of specification as being a complete representation of an immediately higher level.

Substantive Principles prescribe the features and properties of artefacts, or systems that will constitute an optimal solution to a general design problem. As a simple example, a substantive principle deriving from Kirchoff's Laws might be one which would specify the physical structure of a network design (sources, resistances and their nodes etc) whose behaviour (e.g., distribution of current) would constitute an optimal solution to a design problem concerning an amplifier's power supply.

1.5. The Requirement for an Engineering Conception for Human Factors

The contemporary HF discipline does not possess either methodological or substantive engineering principles. The heuristics it possesses are either 'rules of thumb' derived from experience or guidelines derived from psychological theories and findings. Neither guidelines nor rules of thumb offer assurance of their efficacy in any given instance, and particularly with regard to the effectiveness of a design. The methods and models of HF (as opposed to methodological and substantive principles) are similarly without such an assurance. Clearly, any evolution of HF as an engineering discipline in the manner proposed here has yet to begin. There is an immediate need then, for a view of how it might begin, and how formulation of engineering principles might be precipitated.

van Gisch and Pipino (1986) have suggested the process by which scientific (as opposed to engineering) disciplines acquire formal knowledge. They characterise the activities of scientific disciplines at a number of levels, the most general being an epistemological enquiry concerning the nature and origin of discipline knowledge. From such an enquiry a paradigm may evolve. Although a paradigm may be considered to subsume all discipline activities (Long, 1987), it must, at the very least, subsume a coherent and complete definition of the concepts which in this case describe the General (Scientific) Problem of a scientific discipline. Those concepts, and their derivatives, are embodied in the explanatory and predictive theories of science and enable the formulation of research problems. For example, Newton's Principia commences with an epistemological enquiry, and a paradigm in which the concept of inertia first occurs. The concept of inertia is embodied in scientific theories of mechanics, as for example, in Newton's Second Law.

Engineering disciplines may be supposed to require an equivalent epistemological enquiry. However, rather than that enquiry producing a paradigm, we may construe its product as a conception. Such a *conception* is a unitary (and consensus) view of the general design problem of a discipline. Its power lies in the coherence and completeness of its definition of concepts which express that problem. Hence, it enables the formulation of engineering principles which embody and instantiate those concepts. A conception (like a paradigm) is always open to rejection and replacement.

HF currently does not possess a conception of its general design problem. Current views of the issue are ill-formed, fragmentary, or implicit (Shneiderman, 1980; Card, Moran and Newell, 1983; Norman and Draper, 1986). The lack of such a shared view is particularly apparent within the HF research literature in which concepts are ambiguous and lacking in coherence; those associated with the 'interface' (eg, 'virtual objects', 'human performance', 'task semantics', 'user error' etc) are particular examples of this failure. It is inconceivable that a formulation of HF engineering principles might occur whilst there is no consensus understanding of the concepts which they would embody. Articulation of a conception must then be a pre-requisite for formulation of engineering principles for HF.

The origin of a conception for the HF discipline must be a conception for the HCI discipline itself, the superordinate discipline incorporating HF. A conception (at least in form) as might be assumed by an engineering HCI discipline has been previously proposed (Dowell and Long, 1988a). It supports the conception for HF as an engineering discipline presented in Part II.

In conclusion, Part I has presented the case for an engineering conception for HF. A proposal for such a conception follows in Part II. The status of the conception, however, should be emphasised. First, the conception at this point in time is speculative. Second, the conception continues to be developed in support of, and supported by, the research of the authors. Third, there is no validation in the conventional sense to be offered for the conception at this time. Validation of the conception for HF will come from its being able to describe the design problems of HF, and from the coherence of its concepts, that is, from the continuity of relations, and agreement, between concepts. Readers may assess these aspects of validity for themselves. Finally, the validity of the conception for HF will also rest in its being a consensus view held by the discipline as a whole and this is currently not the case.

Part II. Conception for an Engineering Discipline of Human Factors

2.1.	Conception of the Human Factors General Design Problem.....	13
2.2.	Conception of Work and the User.....	15
2.3.	Conception of the Interactive Worksystem and the User.....	18
2.4.	Conception of Performance of the Interactive Worksystem and the User.....	24
2.5.	Conclusions and the Prospect for Human Factors Engineering Principles.	26

The potential for HF to become an engineering discipline, and so better to respond to the problem of interactive systems design, was examined in Part I. The possibility of realising this potential through HF engineering principles was suggested - principles which might prescriptively support HF design expressed as 'specify then implement'. It was concluded that a pre-requisite to the development of HF engineering principles, is a conception of the general design problem of HF, which was informally expressed as:

'to design human interactions with computers for effective working'.

Part II proposes a conception for HF. It attempts to establish the set of related concepts which can express the general design problem of HF more formally. Such concepts would be those embodied in HF engineering principles. As indicated in Section 1.1, the conception for HF is supported by a conception for an engineering discipline of HCI earlier proposed by Dowell and Long (1988a). Space precludes re-iteration of the conception for HCI here, other than as required for the derivation of the conception for HF. Part II first asserts a more formal expression of the HF general design problem which an engineering discipline would address. Part II then continues by elaborating and illustrating the concepts and their relations embodied in that expression.

2.1. Conception of the Human Factors General Design Problem.

The conception for the (super-ordinate) engineering discipline of HCI asserts a fundamental distinction between behavioural systems which perform work, and a world in which work originates, is performed and has its consequences. Specifically conceptualised are interactive worksystems consisting of human and computer behaviours together performing work. It is work evidenced in a world of physical and informational objects disclosed as domains of application. The distinction between worksystems and domains of application is represented schematically in Figure 3.

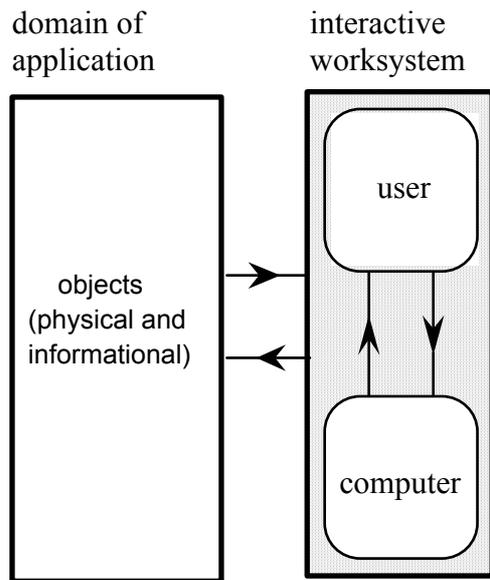


Figure 3. The conception for HCI asserts a fundamental distinction between interactive worksystems and domains (of application). (Adapted from Long and Dowell, 1989).

Effectiveness derives from the relationship of an interactive worksystem with its domain of application - it assimilates both the quality of the work performed by the worksystem, and the costs it incurs. Quality and cost are the primary constituents of the concept of performance through which effectiveness is expressed.

The concern of an engineering HCI discipline would be the design of interactive worksystems for performance. More precisely, its concern would be the design of behaviours constituting a worksystem {S} whose actual performance (P_A) conformed with some desired performance (P_D). And to design {S} would require the design of human behaviours {U} interacting with computer behaviours {C}. Hence, conception of the general design problem of an engineering discipline of HCI is expressed as:

Specify then implement {U} and {C}, such that

$$\{U\} \text{ interacting with } \{C\} = \{S\} \quad \text{as } P_A \longrightarrow P_D$$

where $P_D = f^{\underline{}} \{ Q_D, K_D \}$

Q_D expresses the desired quality of the products of work within the given domain of application, K_D expresses acceptable (i.e., desired) costs incurred by the worksystem, i.e., by both human and computer.

The problem, when expressed as one of to 'specify then implement' designs of interactive worksystems, is equivalent to the general design problems characteristic of other engineering disciplines (see Section 1.4.).

The interactive worksystem can be distinguished as two separate, but interacting sub-systems, that is, a system of human behaviours interacting with a system of computer behaviours. The human behaviours may be treated as a behavioural system in their own right, but one interacting with the system of computer behaviours to perform work. It follows that the general design problem of HCI may be decomposed with regard to its scope (with respect to the human and computer behavioural

sub-systems) giving two related problems. Decomposition with regard to the human behaviours gives the general design problem of the HF¹ discipline as:

Specify then implement {U} such that

$$\{U\} \text{ interacting with } \{C\} = \{S\} \quad \text{as } P_A \longrightarrow P_D$$

The general design problem of HF then, is one of producing implementable specifications of human behaviours {U} which, interacting with computer behaviours {C}, are constituted within a worksystem {S} whose performance conforms with a desired performance (P_D).

The following sections elaborate the conceptualisation of human behaviours (the user, or users) with regard to the work they perform, the interactive worksystem in which they are constituted, and performance.

2.2 . Conception of Work and the User

The conception for HF identifies a world in which work originates, is performed and has its consequences. This section presents the concepts by which work and its relations with the user are expressed.

Objects and their attributes

Work occurs in a world consisting of *objects* and arises in the intersection of organisations and (computer) technology. Objects may be both abstract as well as physical, and are characterised by their *attributes*. Abstract attributes of objects are attributes of information and knowledge. Physical attributes are attributes of energy and matter. Letters (i.e., correspondence) are objects; their abstract attributes support the communication of messages etc; their physical attributes support the visual/verbal representation of information via language.

Attributes and levels of complexity

The different attributes of an object may emerge at different levels within a hierarchy of *levels of complexity* (see Checkland, 1981). For example, characters and their configuration on a page are physical attributes of the object 'a letter' which emerge at one level of complexity; the message of the letter is an abstract attribute which emerges at a higher level of complexity.

Objects are described at different levels of description commensurate with their levels of complexity. However, at a high level of description, separate objects may no longer be differentiated. For example, the object 'income tax return' and the object 'personal letter' are both 'correspondence' objects at a higher level of description. Lower levels of description distinguish their respective attributes of content, intended correspondent etc. In this way, attributes of an object described at one level of description completely re-represent those described at a lower level.

Relations between attributes

Attributes of objects are related, and in two ways. First, attributes at different levels of complexity are related. As indicated earlier, those at one level are completely subsumed in those at a higher level. In particular, abstract attributes will occur at higher levels of complexity than physical attributes and will subsume those lower level physical attributes. For example, the abstract attributes of an object 'message' concerning the representation of its content by language subsume the lower level physical attributes, such as the font of the characters expressing the language. As an alternative example, an

¹The General Design Problem of SE would be equivalent and be expressed as 'Specify then implement {C} such that .. etc.

industrial process, such as a steel rolling process in a foundry, is an object whose abstract attributes will include the process's efficiency. Efficiency subsumes physical attributes of the process, - its power consumption, rate of output, dimensions of the output (the rolled steel), etc - emerging at a lower level of complexity.

Second, attributes of objects are related within levels of complexity. There is a dependency between the attributes of an object emerging within the same level of complexity. For example, the attributes of the industrial process of power consumption and rate of output emerge at the same level and are inter-dependent.

Attribute states and affordance

At any point or event in the history of an object, each of its attributes is conceptualised as having a state. Further, those states may change. For example, the content and characters (attributes) of a letter (object) may change state: the content with respect to meaning and grammar etc; its characters with respect to size and font etc. Objects exhibit an affordance for transformation, engendered by their attributes' potential for state change (see Gibson, 1977). Affordance is generally pluralistic in the sense that there may be many, or even, infinite transformations of objects, according to the potential changes of state of their attributes.

Attributes' relations are such that state changes of one attribute may also manifest state changes in related attributes, whether within the same level of complexity, or across different levels of complexity. For example, changing the rate of output of an industrial process (lower level attribute) will change both its power consumption (same level attribute) and its efficiency (higher level attribute).

Organisations, domains (of application), and the requirement for attribute state changes

A *domain of application* may be conceptualised as: 'a class of affordance of a class of objects'. Accordingly, an object may be associated with a number of domains of application ('domains'). The object 'book' may be associated with the domain of typesetting (state changes of its layout attributes) and with the domain of authorship (state changes of its textual content). In principle, a domain may have any level of generality, for example, the writing of letters and the writing of a particular sort of letter.

Organisations are conceptualised as having domains as their operational province and of requiring the realisation of the affordance of objects. It is a requirement satisfied through work. Work is evidenced in the state changes of attributes by which an object is intentionally transformed: it produces *transforms*, that is, objects whose attributes have an intended state. For example, 'completing a tax return' and 'writing to an acquaintance', each have a 'letter' as their transform, where those letters are objects whose attributes (their content, format and status, for example) have an intended state. Further editing of those letters would produce additional state changes, and therein, new transforms.

Goals

Organisations express their requirement for the transformation of objects through specifying goals. A *product goal* specifies a required transform - a required realisation of the affordance of an object. In expressing the required transformation of an object, a product goal will generally suppose necessary state changes of many attributes. The requirement of each attribute state change can be expressed as a *task goal*, deriving from the product goal. So for example, the product goal demanding transformation of a letter making its message more courteous, would be expressed by task goals possibly requiring state changes of semantic attributes of the propositional structure of the text, and of syntactic attributes of the grammatical structure. Hence, a product goal can be re-expressed as a *task goal structure*, a hierarchical structure expressing the relations between task goals, for example, their sequences.

In the case of the computer-controlled steel rolling process, the process is an object whose transformation is required by a foundry organisation and expressed by a product goal. For example, the product goal may specify the elimination of deviations of the process from a desired efficiency. As indicated earlier, efficiency will at least subsume the process's attributes of power consumption, rate of output, dimensions of the output (the rolled steel), etc. As also indicated earlier, those attributes will be inter-dependent such that state changes of one will produce state changes in the others - for example, changes in rate of output will also change the power consumption and the efficiency of the process. In this way, the product goal (of correcting deviations from the desired efficiency) supposes the related task goals (of setting power consumption, rate of output, dimensions of the output etc). Hence, the product goal can be expressed as a task goal structure and task goals within it will be assigned to the operator monitoring the process.

Quality

The transformation of an object demanded by a product goal will generally be of a multiplicity of attribute state changes - both within and across levels of complexity. Consequently, there may be alternative transforms which would satisfy a product goal - letters with different styles, for example - where those different transforms exhibit differing compromises between attribute state changes of the object. By the same measure, there may also be transforms which will be at variance with the product goal. The concept of *quality* (Q) describes the variance of an actual transform with that specified by a product goal. It enables all possible outcomes of work to be equated and evaluated.

Work and the user

Conception of the domain then, is of objects, characterised by their attributes, and exhibiting an affordance arising from the potential changes of state of those attributes. By specifying product goals, organisations express their requirement for transforms - objects with specific attribute states. Transforms are produced through work, which occurs only in the conjunction of objects affording transformation and systems capable of producing a transformation.

From product goals derive a structure of related task goals which can be assigned either to the human or to the computer (or both) within an associated worksystem. The task goals assigned to the human are those which motivate the human's behaviours. The actual state changes (and therein transforms) which those behaviours produce may or may not be those specified by task and product goals, a difference expressed by the concept of quality.

Taken together, the concepts presented in this section support the HF conception's expression of work as relating to the user. The following section presents the concepts expressing the interactive worksystem as relating to the user.

2.3. Conception of the Interactive Worksystem and the User.

The conception for HF identifies *interactive worksystems* consisting of human and computer behaviours together performing work. This section presents the concepts by which interactive worksystems and the user are expressed.

Interactive worksystems

Humans are able to conceptualise goals and their corresponding behaviours are said to be intentional (or purposeful). Computers, and machines more generally, are designed to achieve goals, and their corresponding behaviours are said to be intended (or purposive¹). An *interactive worksystem*

¹ Human behaviour is teleological, machine behaviour is teleonomic (Checkland, 1981).

('worksystem') is a behavioural system distinguished by a boundary enclosing all human and computer behaviours whose purpose is to achieve and satisfy a common goal. For example, the behaviours of a secretary and wordprocessor whose purpose is to produce letters constitute a worksystem. Critically, it is only by identifying that common goal that the boundary of the worksystem can be established: entities, and more so - humans, may exhibit a range of contiguous behaviours, and only by specifying the goals of concern, might the boundary of the worksystem enclosing all relevant behaviours be correctly identified.

Worksystems transform objects by producing state changes in the abstract and physical attributes of those objects (see Section 2.2). The secretary and wordprocessor may transform the object 'correspondence' by changing both the attributes of its meaning and the attributes of its layout. More generally, a worksystem may transform an object through state changes produced in related attributes. An operator monitoring a computer-controlled industrial process may change the efficiency of the process through changing its rate of output.

The behaviours of the human and computer are conceptualised as behavioural sub-systems of the worksystem - sub-systems which interact¹. The human behavioural sub-system is here more appropriately termed the user. Behaviour may be loosely understood as 'what the human does', in contrast with 'what is done' (i.e. attribute state changes in a domain). More precisely the *user* is conceptualised as:

a system of distinct and related human *behaviours*, identifiable as the sequence of states of a person² interacting with a computer to perform work, and corresponding with a purposeful (intentional) transformation of objects in a domain³ (see also Ashby, 1956).

Although possible at many levels, the user must at least be expressed at a level commensurate with the level of description of the transformation of objects in the domain. For example, a secretary interacting with an electronic mailing facility is a user whose behaviours include receiving and replying to messages. An operator interacting with a computer-controlled milling machine is a user whose behaviours include planning the tool path to produce a component of specified geometry and tolerance.

The user as a system of mental and physical human behaviours

The behaviours constituting a worksystem are both physical as well as abstract. Abstract behaviours are generally the acquisition, storage, and transformation of information. They represent and process information at least concerning: domain objects and their attributes, attribute relations and attribute states, and the transformations required by goals. Physical behaviours are related to, and express, abstract behaviours.

Accordingly, the user is conceptualised as a system of both mental (abstract) and overt (physical) behaviours which extend a mutual influence - they are related. In particular, they are related within an assumed hierarchy of behaviour types (and their control) wherein mental behaviours generally determine, and are expressed by, overt behaviours. Mental behaviours may transform (abstract) domain objects represented in cognition, or express through overt behaviour plans for transforming domain objects.

¹ The human behaviours and computer behaviours are separate systems 'coupled' to form a worksystem (see Ashby, 1956)

²Behaviours are conceptualised as being supported and enabled by co-extensive structures. The user, however, is a description of a behavioural system and does not describe the corresponding human structures (see later in Section 2.3.).

³This conception of human behaviour differs from that of behaviourist psychology which generally seeks correlations between observable inputs and outputs of a mental 'blackbox' without reference to any postulated artifacts of the mind or brain.

So for example, the operator working in the control room of the foundry has the product goal required to maintain a desired condition of the computer-controlled steel rolling process. The operator attends to the computer (whose behaviours include the transmission of information about the process). Hence, the operator acquires a representation of the current condition of the process by collating the information displayed by the computer and assessing it by comparison with the condition specified by the product goal. The operator's acquisition, collation and assessment are each distinct mental behaviours, conceptualised as representing and processing information. The operator reasons about the attribute state changes necessary to eliminate any discrepancy between current and desired conditions of the process, that is, the set of related changes which will produce the required transformation of the process. That decision is expressed in the set of instructions issued to the computer through overt behaviour - making keystrokes, for example.

The user is conceptualised as having cognitive, conative and affective aspects. The cognitive aspects of the user are those of their knowing, reasoning and remembering, etc; the conative aspects are those of their acting, trying and persevering, etc; and the affective aspects are those of their being patient, caring, and assured, etc. Both mental and overt human behaviours are conceptualised as having these three aspects.

Human-computer interaction

Although the human and computer behaviours may be treated as separable sub-systems of the worksystem, those sub-systems extend a "mutual influence", or interaction whose configuration principally determines the worksystem (Ashby, 1956).

Interaction is conceptualised as:

the mutual influence of the user (i.e., the human behaviours) and the computer behaviours associated within an interactive worksystem

Hence, the user {U} and computer behaviours {C} constituting a worksystem {S}, were expressed in the general design problem of HF (Section 2.1) as:

$$\{U\} \text{ interacting with } \{C\} = \{S\}$$

Interaction of the human and computer behaviours is the fundamental determinant of the worksystem, rather than their individual behaviours *per se*. For example, the behaviours of an operator interact with the behaviours of a computer-controlled milling machine. The operator's behaviours influence the behaviours of the machine, perhaps in the tool path program - the behaviours of the machine, perhaps the run-out of its tool path, influences the selection behaviour of the operator. The configuration of their interaction - the inspection that the machine allows the operator, the tool path control that the operator allows the machine - determines the worksystem that the operator and machine behaviours constitute in their planning and execution of the machining work.

The assignment of task goals then, to either the human or the computer delimits the user and therein configures the interaction. For example, replacement of a mis-spelled word required in a document is a product goal which can be expressed as a task goal structure of necessary and related attribute state changes. In particular, the text field for the correctly spelled word demands an attribute state change in the text spacing of the document. Specifying that state change may be a task goal assigned to the user, as in interaction with the behaviours of early text editor designs, or it may be a task goal assigned to the computer, as in interaction with the 'wrap-round' behaviours of contemporary wordprocessor designs. The assignment of the task goal of specification configures the interaction of the human and computer behaviours in each case; it delimits the user.

On-line and off-line human behaviours

The user may include both on-line and off-line human behaviours: on-line behaviours are associated with the computer's representation of the domain; offline behaviours are associated with non-computer representations of the domain, or the domain itself.

As an illustration of the distinction, consider the example of an interactive worksystem consisting of behaviours of a secretary and a wordprocessor and required to produce a paper-based copy of a dictated letter stored on audio tape. The product goal of the worksystem here requires the transformation of the physical representation of the letter from one medium to another, that is, from tape to paper. From the product goal derives the task goals relating to required attribute state changes of the letter. Certain of those task goals will be assigned to the secretary. The secretary's off-line behaviours include listening to, and assimilating the dictated letter, so acquiring a representation of the domain directly. By contrast, the secretary's on-line behaviours include specifying the representation by the computer of the transposed content of the letter in a desired visual/verbal format of stored physical symbols.

On-line and off-line human behaviours are a particular case of the 'internal' interactions between a human's behaviours as, for example, when the secretary's typing interacts with memorisations of successive segments of the dictated letter.

Human structures and the user

Conceptualisation of the user as a system of human behaviours needs to be extended to the structures supporting behaviour.

Whereas human behaviours may be loosely understood as 'what the human does', the structures supporting them can be understood as 'how they are able to do what they do' (see Marr, 1982; Wilden, 1980). There is a one to many mapping between a human's structures and the behaviours they might support: the structures may support many different behaviours.

In co-extensively enabling behaviours at each level, structures must exist at commensurate levels. The human structural architecture is both physical and mental, providing the capability for a human's overt and mental behaviours. It provides a representation of domain information as symbols (physical and abstract) and concepts, and the processes available for the transformation of those representations. It provides an abstract structure for expressing information as mental behaviour. It provides a physical structure for expressing information as physical behaviour.

Physical human structure is neural, bio-mechanical and physiological. Mental structure consists of representational schemes and processes. Corresponding with the behaviours it supports and enables, human structure has cognitive, conative and affective aspects. The cognitive aspects of human structures include information and knowledge - that is, symbolic and conceptual representations - of the domain, of the computer and of the person themselves, and it includes the ability to reason. The conative aspects of human structures motivate the implementation of behaviour and its perseverance in pursuing task goals. The affective aspects of human structures include the personality and temperament which respond to and supports behaviour.

To illustrate the conceptualisation of mental structure, consider the example of structure supporting an operator's behaviours in the foundry control room. Physical structure supports perception of the steel rolling process and executing corrective control actions to the process through the computer input devices. Mental structures support the acquisition, memorisation and transformation of information about the steel rolling process. The knowledge which the operator has of the process and of the computer supports the collation, assessment and reasoning about corrective control actions to be executed.

The limits of human structure determine the limits of the behaviours they might support. Such structural limits include those of: intellectual ability; knowledge of the domain and the computer; memory and attentional capacities; patience; perseverance; dexterity; and visual acuity etc. The structural limits on behaviour may become particularly apparent when one part of the structure (a

channel capacity, perhaps) is required to support concurrent behaviours, perhaps simultaneous visual attending and reasoning behaviours. The user then, is 'resource' limited by the co-extensive human structure.

The behavioural limits of the human determined by structure are not only difficult to define with any kind of completeness, they will also be variable because that structure can change, and in a number of respects. A person may have self-determined changes in response to the domain - as expressed in learning phenomena, acquiring new knowledge of the domain, of the computer, and indeed of themselves, to better support behaviour. Also, human structure degrades with the expenditure of resources in behaviour, as evidenced in the phenomena of mental and physical fatigue. It may also change in response to motivating or de-motivating influences of the organisation which maintains the worksystem.

It must be emphasised that the structure supporting the user is independent of the structure supporting the computer behaviours. Neither structure can make any incursion into the other, and neither can directly support the behaviours of the other. (Indeed this separability of structures is a pre-condition for expressing the worksystem as two interacting behavioural sub-systems.) Although the structures may change in response to each other, they are not, unlike the behaviours they support, interactive; they are not included within the worksystem. The combination of structures of both human and computer supporting their interacting behaviours is conceptualised as the *user interface*.

Resource costs of the user

Work performed by interactive worksystems always incurs *resource costs*. Given the separability of the human and the computer behaviours, certain resource costs are associated directly with the user and distinguished as structural human costs and behavioural human costs.

Structural human costs are the costs of the human structures co-extensive with the user. Such costs are incurred in developing and maintaining human skills and knowledge. More specifically, structural human costs are incurred in training and educating people, so developing in them the structures which will enable their behaviours necessary for effective working. Training and educating may augment or modify existing structures, provide the person with entirely novel structures, or perhaps even reduce existing structures. Structural human costs will be incurred in each case and will frequently be borne by the organisation. An example of structural human costs might be the costs of training a secretary in the particular style of layout required for an organisation's correspondence with its clients, and in the operation of the computer by which that layout style can be created.

Structural human costs may be differentiated as cognitive, conative and affective structural costs of the user. Cognitive structural costs express the costs of developing the knowledge and reasoning abilities of people and their ability for formulating and expressing novel plans in their overt behaviour - as necessary for effective working. Conative structural costs express the costs of developing the activity, stamina and persistence of people as necessary for effective working. Affective structural costs express the costs of developing in people their patience, care and assurance as necessary as necessary for effective working.

Behavioural human costs are the resource costs incurred by the user (i.e by human behaviours) in recruiting human structures to perform work. They are both physical and mental resource costs. Physical behavioural costs are the costs of physical behaviours, for example, the costs of making keystrokes on a keyboard and of attending to a screen display; they may be expressed without differentiation as physical workload. Mental behavioural costs are the costs of mental behaviours, for example, the costs of knowing, reasoning, and deciding; they may be expressed without differentiation as mental workload. Mental behavioural costs are ultimately manifest as physical behavioural costs.

When differentiated, mental and physical behavioural costs are conceptualised as the cognitive, conative and affective behavioural costs of the user. Cognitive behavioural costs relate to both the mental representing and processing of information, and the demands made on the individual's extant

knowledge, as well as the physical expression thereof in the formulation and expression of a novel plan. Conative behavioural costs relate to the repeated mental and physical actions and effort required by the formulation and expression of the novel plan. Affective behavioural costs relate to the emotional aspects of the mental and physical behaviours required in the formulation and expression of the novel plan. Behavioural human costs are evidenced in human fatigue, stress and frustration; they are costs borne directly by the individual.

2.4. Conception of Performance of the Interactive Worksystem and the User.

In asserting the general design problem of HF (Section 2.1.), it was reasoned that:

"Effectiveness derives from the relationship of an interactive worksystem with its domain of application - it assimilates both the quality of the work performed by the worksystem, and the costs incurred by it. Quality and cost are the primary constituents of the concept of performance through which effectiveness is expressed. "

This statement followed from the distinction between interactive worksystems performing work, and the work they perform. Subsequent elaboration upon this distinction enables reconsideration of the concept of performance, and examination of its central importance within the conception for HF.

Because the factors which constitute this engineering concept of performance (i.e the quality and costs of work) are determined by behaviour, a concordance is assumed between the behaviours of worksystems and their performance: behaviour determines performance (see Ashby, 1956; Rouse, 1980). The quality of work performed by interactive worksystems is conceptualised as the actual transformation of objects with regard to their transformation demanded by product goals. The costs of work are conceptualised as the resource costs incurred by the worksystem, and are separately attributed to the human and computer. Specifically, the resource costs incurred by the human are differentiated as: structural human costs - the costs of establishing and maintaining the structure supporting behaviour; and behavioural human costs - the costs of the behaviour recruiting structure to its own support. Structural and behavioural human costs were further differentiated as cognitive, conative and affective costs.

A desired performance of an interactive worksystem may be conceptualised. Such a desired performance might either be absolute, or relative as in a comparative performance to be matched or improved upon. Accordingly, *criteria* expressing desired performance, may either specify categorical gross resource costs and quality, or they may specify critical instances of those factors to be matched or improved upon¹.

Discriminating the user's performance within the performance of the interactive worksystem would require the separate assimilation of human resource costs and their achievement of desired attribute state changes demanded by their assigned task goals. Further assertions concerning the user arise from the conceptualisation of worksystem performance. First, the conception of performance is able to distinguish the quality of the transform from the effectiveness of the worksystems which produce them. This distinction is essential as two worksystems might be capable of producing the same transform, yet if one were to incur a greater resource cost than the other, its effectiveness would be the lesser of the two systems.

Second, given the concordance of behaviour with performance, optimal human (and equally, computer) behaviours may be conceived as those which incur a minimum of resource costs in producing a given transform. Optimal human behaviour would minimise the resource costs incurred in producing a transform of given quality (Q). However, that optimality may only be categorically determined with regard to worksystem performance, and the best performance of a worksystem may still be at variance with the performance desired of it (P_D). To be more specific, it is not sufficient for human behaviours simply to be error-free. Although the elimination of errorful human behaviours may contribute to the best performance possible of a given worksystem, that performance may still be

¹See Section 1.4. where the possibility for expressing, by an absolute value, the desired performance of a system or artifact is associated with the hardness of the design problem.

less than desired performance. Conversely, although human behaviours may be errorful, a worksystem may still support a desired performance.

Third, the common measures of human 'performance' - errors and time, are related in this conceptualisation of performance. Errors are behaviours which increase resource costs incurred in producing a given transform, or which reduce the quality of transform, or both. The duration of human behaviours may (very generally) be associated with increases in behavioural user costs.

Fourth, structural and behavioural human costs may be traded-off in performance. More sophisticated human structures supporting the user, that is, the knowledge and skills of experienced and trained people, will incur high (structural) costs to develop, but enable more efficient behaviours - and therein, reduced behavioural costs.

Fifth, resource costs incurred by the human and the computer may be traded-off in performance. A user can sustain a level of performance of the worksystem by optimising behaviours to compensate for the poor behaviours of the computer (and *vice versa*), i.e., behavioural costs of the user and computer are traded-off. This is of particular concern for HF as the ability of humans to adapt their behaviours to compensate for poor computer-based systems often obscures the low effectiveness of worksystems.

This completes the conception for HF. From the initial assertion of the general design problem of HF, the concepts that were invoked in its formal expression have subsequently been defined and elaborated, and their coherence established.

2.5. Conclusions and the Prospect for Human Factors Engineering Principles

Part I of this paper examined the possibility of HF becoming an engineering discipline and specifically, of formulating HF engineering principles. Engineering principles, by definition prescriptive, were seen to offer the opportunity for a significantly more effective discipline, ameliorating the problems which currently beset HF - problems of poor integration, low efficiency, efficacy without guarantee, and slow development.

A conception for HF is a pre-requisite for the formulation of HF engineering principles. It is the concepts and their relations which express the HF general design problem and which would be embodied in HF engineering principles. The form of a conception for HF was proposed in Part II. Originating in a conception for an engineering discipline of HCI (Dowell and Long, 1988a), the conception for HF is postulated as appropriate for supporting the formulation of HF engineering principles.

The conception for HF is a broad view of the HF general design problem. Instances of the general design problem may include the development of a worksystem, or the utilisation of a worksystem within an organisation. Developing worksystems which are effective, and maintaining the effectiveness of worksystems within a changing organisational environment, are both expressed within the problem. In addition, the conception takes the broad view on the research and development activities necessary to solve the general design problem and its instantiations, respectively. HF engineering research practices would seek solutions, in the form of (methodological and substantive) engineering principles, to the general design problem. HF engineering practices in systems development programmes would seek to apply those principles to solve instances of the general design problem, that is, to the design of specific users within specific interactive worksystems. Collaboration of HF and SE specialists and the integration of their practices is assumed.

Notwithstanding the comprehensive view of determinacy developed in Part I, the intention of specification associated with people might be unwelcome to some. Yet, although the requirement for

design and specification of the user is being unequivocally proposed, techniques for implementing those specifications are likely to be more familiar than perhaps expected - and possibly more welcome. Such techniques might include selection tests, aptitude tests, training programmes, manuals and help facilities, or the design of the computer.

A selection test would assess the conformity of a candidates' behaviours with a specification for the user. An aptitude test would assess the potential for a candidates' behaviours to conform with a specification for the user. Selection and aptitude tests might assess candidates either directly or indirectly. A direct test would observe candidates' behaviours in 'hands on' trial periods with the 'real' computer and domain, or with simulations of the computer and domain. An indirect test would examine the knowledge and skills (i.e., the structures) of candidates, and might be in the form of written examinations. A training programme would develop the knowledge and skills of a candidate as necessary for enabling their behaviours to conform with a specification for the user. Such programmes might take the form of either classroom tuition or 'hands on' learning. A manual or on-line help facility would augment the knowledge possessed by a human, enabling their behaviours to conform with a specification for the user. Finally, the design of the computer itself, through the interactions of its behaviours with the user, would enable the implementation of a specification for the user.

To conclude, discussion of the status of the conception for HF must be briefly extended. The contemporary HF discipline was characterised as a craft discipline. Although it may alternatively be claimed as an applied science discipline, such claims must still admit the predominantly craft nature of systems development practices (Long and Dowell, 1989). No instantiations of the HF engineering discipline implied in this paper are visible, and examples of supposed engineering practices may be readily associated with craft or applied science disciplines. There are those, however, who would claim the craft nature of the HF discipline to be dictated by the nature of the problem it addresses. They may maintain that the indeterminism and complexity of the problem of designing human systems (the softness of the problem) precludes the application of formal and prescriptive knowledge. This claim was rejected in Part I on the grounds that it mistakes the current absence of formal discipline knowledge as an essential reflection of the softness of its general design problem. The claim fails to appreciate that this absence may rather be symptomatic of the early stage of the discipline's development. The alternative position taken by this paper is that the softness of the problem needs to be independently established. The general design problem of HF is, to some extent, hard - human behaviour is clearly to some useful degree deterministic - and certainly sufficiently deterministic for the design of certain interactive worksystems. It may accordingly be presumed that HF engineering principles can be formulated to support product quality within a systems development ethos of 'design for performance'.

The extent to which HF engineering principles might be realisable in practice remains to be seen. It is not supposed that the development of effective systems will never require craft skills in some form, and engineering principles are not seen to be incompatible with craft knowledge, particularly with respect to their instantiation (Long and Dowell, 1989). At a minimum, engineering principles might be expected to augment the craft knowledge of HF professionals. Yet the great potential of HF engineering principles for the effectiveness of the discipline demands serious consideration. However, their development would only be by intention, and would be certain to demand a significant research effort. This paper is intended to contribute towards establishing the conception required for the formulation of HF engineering principles.

References

- Ashby W. Ross, (1956), *An Introduction to Cybernetics*. London: Methuen.
- Bornat R. and Thimbleby H., (1989), *The Life and Times of ded, Text Display Editor*. In J.B. Long and A.D. Whitefield (ed.s), *Cognitive Ergonomics and Human Computer Interaction*. Cambridge: Cambridge University Press.
- Card, S. K., Moran, T., and Newell, A., (1983), *The Psychology of Human Computer Interaction*, New Jersey: Lawrence Erlbaum Associates.
- Carey, T., (1989), *Position Paper: The Basic HCI Course For Software Engineers*. SIGCHI Bulletin, Vol. 20, no. 3.
- Carroll J.M., and Campbell R. L., (1986), *Softening up Hard Science: Reply to Newell and Card*. *Human Computer Interaction*, Vol. 2, pp. 227-249.
- Checkland P., (1981), *Systems Thinking, Systems Practice*. Chichester: John Wiley and Sons.
- Cooley M.J.E., (1980), *Architect or Bee? The Human/Technology Relationship*. Slough: Langley Technical Services.
- Didner R.S. *A Value Added Approach to Systems Design*. *Human Factors Society Bulletin*, May 1988.
- Dowell J., and Long J. B., (1988a), *Human-Computer Interaction Engineering*. In N. Heaton and M. Sinclair (ed.s), *Designing End-User Interfaces. A State of the Art Report*. 15:8. Oxford: Pergamon Infotech.
- Dowell, J., and Long, J. B., 1988b, *A Framework for the Specification of Collaborative Research in Human Computer Interaction*, in UK IT 88 Conference Publication 1988, pub. IEE and BCS.
- Gibson J.J., (1977), *The Theory of Affordances*. In R.E. Shaw and J. Branford (ed.s), *Perceiving, Acting and Knowing*. New Jersey: Erlbaum.
- Gries D., (1981), *The Science of Programming*, New York: Springer Verlag.
- Hubka V., Andreason M.M. and Eder W.E., (1988), *Practical Studies in Systematic Design*, London: Butterworths.
- Long J.B., Hammond N., Barnard P. and Morton J., (1983), *Introducing the Interactive Computer at Work: the Users' Views*. *Behaviour And Information Technology*, 2, pp. 39-106.
- Long, J., (1987), *Cognitive Ergonomics and Human Computer Interaction*. In P. Warr (ed.), *Psychology at Work*. England: Penguin.
- Long J.B., (1989), *Cognitive Ergonomics and Human Computer Interaction: an Introduction*. In J.B. Long and A.D. Whitefield (ed.s), *Cognitive Ergonomics and Human Computer Interaction*. Cambridge: Cambridge University Press.
- Long J.B. and Dowell J., (1989), *Conceptions of the Discipline of HCI: Craft, Applied Science, and Engineering*. In Sutcliffe A. and Macaulay L., *Proceedings of the Fifth Conference of the BCS HCI SG*. Cambridge: Cambridge University Press.

Marr D., (1982), *Vision*. New York: Wh Freeman and Co.

Morgan D.G., Shorter D.N. and Tainsh M., (1988), *Systems Engineering. Improved Design and Construction of Complex IT systems*. Available from IED, Kingsgate House, 66-74 Victoria Street, London, SW1.

Norman D.A. and Draper S.W. (eds) (1986): *User Centred System Design*. Hillsdale, New Jersey: Lawrence Erlbaum;

Pirsig R., 1974, *Zen and the Art of Motorcycle Maintenance*. London: Bodley Head.

Rouse W. B., (1980), *Systems Engineering Models of Human Machine Interaction*. New York: Elsevier North Holland.

Shneiderman B. (1980): *Software Psychology: Human Factors in Computer and Information Systems*. Cambridge, Mass.: Winthrop.

Thimbleby H., (1984), *Generative User Engineering Principles for User Interface Design*. In B. Shackel (ed.), *Proceedings of the First IFIP conference on Human-Computer Interaction. Human-Computer Interaction - INTERACT'84*. Amsterdam: Elsevier Science. Vol.2, pp. 102-107.

van Gisch J. P. and Pipino L.L., (1986), *In Search of a Paradigm for the Discipline of Information Systems*, *Future Computing Systems*, 1 (1), pp. 71-89.

Walsh P., Lim K.Y., Long J.B., and Carver M.K., (1988), *Integrating Human Factors with System Development*. In: N. Heaton and M. Sinclair (eds): *Designing End-User Interfaces*. Oxford: Pergamon Infotech.

Wilden A., 1980, *System and Structure; Second Edition*. London: Tavistock Publications.

This paper has greatly benefited from discussion with others and from their criticisms. We would like to thank our colleagues at the Ergonomics Unit, University College London and in particular, Andy Whitefield, Andrew Life and Martin Colbert. We would also like to thank the editors of the special issue for their support and two anonymous referees for their helpful comments. Any remaining infelicities - of specification and implementation - are our own.