Towards Engineering Principles for Human-Computer Interaction (Domestic Energy Planning and Control)

Jeremy Adam Joseph Stork

University College London

Submitted for the Degree of Doctor of Philosophy
at the University of London

June 1999

Abstract

This research identifies a specific and general operational problem of an inability to design human-computer systems effectively. The technical problem is considered a lack of suitable Human-Computer Interaction (HCI) knowledge. The long-term technical solution is considered to be the development of 'engineering principles', as proposed by Dowell and Long (1989). The technical aim of this research is to make progress towards these engineering principles.

'Engineering principles' for HCI are considered the knowledge required by an engineering conception of the discipline of HCI; it is knowledge that offers a guarantee of application. The thesis conceptualises engineering principles and describes a strategy for their development. The strategy involves cycles of development of human-computer systems using current HCI best-practice, which, in this case, includes the application of a structured Method for USability Engineering (MUSE; Lim and Long, 1994).

To assess the strategy, two domestic energy management systems are developed and examples of initial engineering principles, for domestic energy planning and control, acquired. The engineering principles are considered 'initial', since they have not been validated by application. Further, the status of these examples of initial engineering principles is considered 'early', requiring generalisation by further development cycles. The strategy is considered successful, given the early status of the initial engineering principles acquired. This research is concluded to have made significant progress *towards* engineering principles.

Given the success of the strategy, a version of MUSE, termed MUSE/R—MUSE for Research, is presented to support further engineering principle acquisition and validation. Shorter-term research products are identified and further research is outlined.

ascend to the concrete

Luria, The Mind of a Mnemonist

Tiön is surely a labyrinth,
but it is a labyrinth devised by men,
a labyrinth destined to be deciphered by men.

Borges, Labyrinths

Contents

List of Figures

List of Figures in Pocket on Back Cover

Acknowledgements

I acknowledge the support of the Engineering and Physical Sciences Research Council (EPSRC); of Schlumberger Industries, Paris; and of the Ergonomics & HCI Unit, University College London. The research was supported by an EPSRC CASE studentship.

I thank many for their support and patience, in particular: Professor John Long, my supervisor; Sam Stork, my wife; and my family. I also thank my colleagues at the Ergonomics & HCI Unit for their helpful comments both on papers and during seminars.

1. Introduction

This research is primarily concerned with improving Human-Computer Interaction (HCI) design through the development of HCI design knowledge. The requirement to develop design knowledge arose from a specific operational problem with design, raised by the industrial sponsor of the research. The following sections outline this specific problem with HCI design, and show that the problem is more widespread. Essentially, the general operational problem is an inability to design human-computer systems effectively.

The chapter ends with an overview of the thesis by chapter.

Specific Problem with an EMS Design

France has recently changed its electricity tariffs to try to reduce the number of power stations. To take advantage of cheap electricity, you must now accept to have your supply changed dramatically at short notice. For example, the electricity supplier can inform you that, for the next day, electricity will be very expensive.

The sponsor of this research, a French company, designed a prototype energy management system (EMS) for the home. Their prototype EMS aimed to reduce the inconvenience of supply changes at short notice, in order to enable homeowners to take advantage of the cheaper electricity (Appendix A). The prototype EMS was a triumph of hardware and software development. For example, it could control appliances, kettles, heaters, etc. remotely, simply by having them on a special plug with the control signals carried over the standard electricity wires. Further, strong software engineering techniques (object-oriented analysis, design, and implementation) were used in its development.

However, an HCI evaluation of the prototype EMS undertaken during the course of this research (Appendix A) shows that it was unlikely to achieve the original intention, that of enabling home owners to take advantage of the cheaper tariffs. The prototype did not have appropriate usability and functionality, and would have been ineffective—a case of good software and hardware engineering potentially being squandered. HCI input into the design process would probably have improved the effectiveness of the prototype EMS.

However, there was some HCI input into the design process: a late evaluation. Since the developers did not have HCI expertise, they commissioned an evaluation to deliver design recommendations. The evaluation was commissioned late in the design process, after the developers considered the hardware and software nearly complete. None of the design recommendations were incorporated into the prototype. Therefore, to be effective, the HCI input needed to be early in the design process.

Existing HCI input, such as Hierarchical Task Analysis, structured methods, guidelines, etc., was available to be applied early in the design process. However, this existing early HCI input was not effective enough, for the following reasons:

- Early in the design process, the developers had placed an emphasis on hardware and software development, rather than HCI, because they could be *more certain* of solving the hardware and software 'engineering' problems.
- Available early HCI input, while better than no HCI input, might still have resulted in an unacceptably ineffective EMS.
- Available early HCI input can be criticised for involving too much iteration during design, its application being 'trial-and-error' (Draper ,1991).

If the existing HCI input was not effective enough for the prototype EMS, then it will not be effective enough for the future. Even more effective EMS designs will be required in the future. The sponsor is interested in improving the design of its future EMS products as well as their current EMS product.

The specific problem, with an EMS design, is an inability to design a prototype EMS effectively, using existing available HCI input to the design process.

General Problem with HCI Design

The prototype EMS described above is an interactive human-computer system, and its design is within the scope of HCI. Many hardware and software products, whose scope is within that of HCI design, are ineffective, i.e. fail to provide appropriate usability and functionality. Many have been developed using good hardware and software engineering, and little, none, or late HCI input. The standard example is the video recorder, which many cannot program (usability) and thus many fail to time-shift program watching as they wish (functionality), although video recorders rarely suffer hardware or software breakdowns. The situation has not changed since Thimbleby (1991) wrote: 'video cassette recorders (VCRs) have poor user interfaces, and their user interfaces show no improvement over the now considerable period of their development.'

London Ambulance Service

> 'Few will not recall the computer system failure/collapse at the London Ambulance Service in 1992 which received world-wide media attention.' (Tighe, 1996)

In particular, Newman (1994) claims that an important part of this failure was that 'a more radical user interface design was used than might have been appropriate (LAS, 1993)'. Tighe, the project leader after the failure, claimed that user involvement and prototyping were 'to prove critical to future success'. Again, HCI was not involved until proven necessary, and then only in a prototyping rôle.

The challenge for HCI is to demonstrate considerable benefits when included in the design process, and demonstrate them convincingly.

Energy Management Systems in the Future

Brinkworth (1993) argues that there is a world energy crisis, with a massive predicted increase in primary energy consumption, CO2 emissions, and car usage. Energy utilities are seeking to introduce rate changes for industry and the home to address this crisis (Capehart, 1986). Sophisticated software and hardware is proposed for the home following industrial use (Rahman and Bhatnagar, 1986; Benator, 1987). In France, where two-thirds of the domestic energy bill is for heating (Energy, 1990), trials of advanced EMSs are starting (Phillips, 1994). Complex, remote-controlled, meters will be available for homes that,

> 'could reduce the load by turning on heating systems at staggered intervals. "The system could be extended to non-heating appliances," says Nunn [an industry spokesperson]. "Non-essential high energy appliances like washing machines, dish washers, and tumble dryers could run within a time band rather than a specific time." They would be turned on and off by a signal from the electricity supplier.' (Goodwin, 1995)

Capehart (1986) sees the problems for the introduction of advanced EMS technology in the home as 'interest and cost-effectiveness'.

Solving these problems is another HCI challenge.

Corporate Productivity

Attewell (Constant, 1993) conducted three types of studies to gather the evidence for and against claims that Information Technology has improved corporate productivity. He analysed inter-firm differences within an industry, inter-industry differences, and entire economic sectors. He claimed that 'overall the evidence is that improvements [in productivity] have been insignificant or absent'; a claim supported by a panel discussion at a CSCW conference (Constant, 1993). Although, Attewell's definition of 'productivity' could be criticised as being narrow, the result remains of concern.

These concerns are expressed by others, including Newman et al. (1996), who analysed document authoring by economists. This analysis 'shows that benefits gained from word processing appear to be offset by authors' tendency to tinker with documents up to the last minute'. Newman et al. cite further examples:

> 'Recent studies have questioned the widely-held view that interactive computer systems offer productivity gains to service industries such as banking, insurance and health care. For example, a 1992 study of the impact of 15 years of information technology investments at the World Bank could find no evidence of gains in productivity (Katzenstein, 1993). Landauer (1995) has quoted a number of similar cases in his general study of productivity trends in the service industries since the 1970s.'

A survey by Bellotti (1988) showed that corporations had 'no confidence in HCI as a discipline and no perceived need for it'. The situation has changed, but there is still a need to improve the confidence in HCI in order to justify HCI during development (Johnson, 1995).

The challenge for HCI is both to improve the confidence in HCI and to improve corporate productivity.

General Problem

That the challenges are for HCI has been succinctly expressed by Draper (1991):

> 'The fundamental fact about HCI today is that neither computer science or cognitive science have any theories to offer that are adequate for predicting how a given design of user interface will perform.'

The general problem with HCI is an inability to design human-computer systems effectively, using available HCI input to the design process. The primary aim in this research is to address these challenges for HCI, and enable HCI to deliver considerable benefits for human-computer systems.

Overview of the Thesis

This first chapter has identified the specific and general problems with HCI design that the research will address. This 'operational' general problem, outlined above, is given a technical expression, the 'technical problem', in terms of the development of HCI design knowledge. The 'technical solution' aims to solve, at least to a declared extent, this operational problem (see Figure 1). The desired operational solution is more effective HCI design practice, for EMSs in particular and human-computer systems in general.

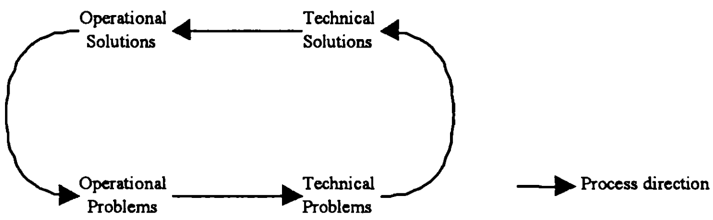


Figure 1. Operational and Technical Problem.

The terms 'conception', 'conceptualise', and 'operationalisation' are used extensively in the thesis. A conception is understood to be a set of concepts, which are abstractions over a class of objects, based on their common aspects, and their relations. Conceptualisation is the process of generating a conception. Operationalisation is the process of instancing a conception to produce an operationalisation. An operationalisation of a conception is a set of less abstract concepts (related to the concepts in the conception) that ultimately reference observables in the 'real' world.

It is convenient at this stage to introduce the concept of 'metrics' and the process of 'metrication'. Metrication is the process of instancing an operationalisation to its limit, to produce metrics. Metrics quantify the less abstract concepts of the operationalisation in an observable relation with the 'real' world.

Chapter 2—Engineering Principles: Technical Solution

The technical problem is identified as the requirement to make HCI knowledge more effective, and specifically to acquire HCI knowledge with a guarantee of application.

Dowell and Long's assessment of the discipline of HCI (Long and Dowell, 1989; and Dowell and Long, 1989) is first described and then applied to identify the technical problem. Dowell and Long propose the development in the longer term of HCI as an engineering discipline, with knowledge that has a guarantee. They term this knowledge 'engineering principles'.

The development of engineering principles for HCI is the long-term technical solution addressed by this research. The technical aim of this research is to make progress towards these engineering principles, through the development of a conception of engineering principles, a strategy for their acquisition, and an implementation and assessment of that strategy.

Chapter 3—Conception of (Substantive) Engineering Principles

Dowell and Long's conception of the general design problem of an engineering discipline of HCI (Long and Dowell, 1989; and Dowell and Long, 1989) is presented. A conception of (substantive) engineering principles is developed for this research from the Dowell and Long conception. *Substantive* engineering principles 'prescribe the features and properties of artefacts' (Dowell and Long). Components of (substantive) engineering principles are conceptualised for this research: general design problems; general design solutions; specific design problems; specific design solutions; partial design problems; and partial design solutions.

A secondary aim of this research is an informal assessment of the Dowell and Long requirement for an engineering discipline of HCI and their conception of an engineering discipline of HCI.

Chapter 4—Strategy for Developing Engineering Principles

A strategy for developing (substantive) engineering principles is developed for this research.

The strategy identifies two stages of engineering principle development. The first stage is the development of 'initial' engineering principles, which have not been validated by application. The second is the validation of those initial engineering principles. The aim of this research is to assess the proposed strategy by implementing it to develop examples of initial engineering principles.

The strategy involves cycles of current HCI best-practice development and operationalisation of that development as specific design problems and solutions, followed by the identification of general relationships, as initial engineering principles, between these operationalisations. Two cycles of development and operationalisation, followed by identification, are proposed for this research to assess the strategy. Current HCI best practice includes MUSE—a Method for USability Engineering (Lim and Long, 1994).

The rationale for scoping this research to 'planning and control' engineering principles is presented. Additional shorter-term benefits of the strategy are identified.

Chapter 5—Conception of Human-Computer Systems

An initial conception of human-computer systems and their costs is developed for this research. The human components of the conception are based on a human mental architecture developed by Timmer and Long (1996 and 1997).

Chapter 6—Operationalising Specific Design Problems and Solutions

Frameworks to support operationalising the effectiveness of human-computer systems have been developed during this research. These frameworks include the layout and scope of diagrams and tables to operationalise and metricate specific design problems and their solutions.

Chapter 7—Conception of Planning and Control

An initial conception of planning and control is developed for this research. The conception is based on previous research into planning and control in HCI, Psychology, and Artificial Intelligence.

Chapter 8—Cycle 1 Best-Practice Development

The Cycle 1 user requirements involve a problem with a domestic heating system that results in the user becoming cold when working at home in the mornings. An artefact was designed by HCI best-practice to solve this problem. The artefact is a modification to the heating system controller, and was prototyped to support evaluation. The evaluation was positive.

The user requirements, application of HCI best-practice, resulting artefact, and the evaluation of the artefact are summarised.

Chapter 9—Cycle 1 Operationalisation

The operationalisation of the specific design problem and solution for Cycle 1 is summarised.

Chapter 10—Cycle 2 Best-Practice Development

The Cycle 2 user requirements involve a problem with a domestic heating system that results in the user becoming overheated during vigorous cooking, to the detriment of the food produced. As for Cycle 1, an artefact was designed by HCI best-practice to solve this problem. The artefact is a new planning aid for cooking, and was prototyped to support evaluation. The evaluation was positive.

The user requirements, application of HCI best-practice, resulting artefact, and the evaluation of the artefact are summarised.

Chapter 11—Cycle 2 Operationalisation

As for Cycle 1, the operationalisation of the specific design problem and solution for Cycle 2 is summarised.

Chapter 12—Initial Engineering Principles

A detailed strategy for identifying initial engineering principles from the two cycles is developed. The detailed strategy includes six means of targeting initial engineering principles. Examples of initial engineering principles are developed for each of these six forms of targeting.

Chapter 13—Strategy Assessment and Discussion

The status of the acquired initial engineering principles is considered 'early', requiring generalisation from further development cycles. The strategy is considered successful at this stage, requiring further cycles and initial engineering principle validation. Future research based on this research is discussed.

Chapter 14—MUSE for Research (MUSE/R)

Methodological support is developed for future research that will apply design knowledge to acquire engineering principles. This research proposes and outlines a version of the MUSE method to support such research, termed MUSE/R—MUSE for Research.

Potential tool support for the methodological support is identified. Tool support is essential for serious progress in the acquisition and application of engineering principles.

Chapter 15—Conclusions

This research is considered to have made significant progress *towards* Dowell and Long's engineering principles, and, therefore, progress towards solving the operational problem for this research of the inability to design human-computer systems effectively. The progress towards engineering principles made by this research is: a conception of engineering principles, a strategy for the acquisition of engineering principles, examples of early initial engineering principles, a positive assessment of the strategy at this juncture, and an outline of further research for the acquisition of engineering principles.

The research informally supports the Dowell and Long requirement for an engineering discipline of HCI and their conception of an engineering discipline of HCI. The research also delivers some additional shorter-term benefits.

2. Engineering Principles: Technical Solution

The previous chapter identified the general operational problem of this research, that of an inability to design human-computer systems effectively, using available HCI input to the design process. This chapter identifies the technical problem of this research and its technical solution.

Dowell and Long's (D&L) assessment of the discipline of HCI (Long and Dowell, 1989; and Dowell and Long, 1989) is described and then applied to identify the technical problem. The technical problem is identified as the requirement to make HCI knowledge more effective, and specifically to acquire HCI knowledge with a guarantee of application.

Dowell and Long propose the development in the longer term of an engineering discipline, with knowledge that has a guarantee. They term this knowledge 'engineering principles'.

The development of engineering principles for HCI is the long-term technical solution addressed by this research. The technical aim of this research is to make progress towards these engineering principles, through the development of a conception of engineering principles, a strategy for their acquisition, and an implementation and assessment of that strategy.

Dowell and Long's Characterisation of the Discipline of HCI

Dowell and Long (Dowell and Long, 1989; and Long and Dowell, 1989) characterise the discipline of HCI as a design discipline rather than as a scientific discipline: HCI design knowledge supports HCI design practice, which is to provide solutions to general HCI design problems. They identify from this characterisation a means of assessing the effectiveness of HCI knowledge: to be effective, HCI knowledge must be conceptualised; operationalisable; generalisable; and testable. These features are used by them to assess the current HCI discipline and propose a more effective discipline.

Current State of the HCI Discipline

D&L characterise the current state of the HCI discipline as that of a 'craft' discipline. Knowledge is implicit and informal, consisting of 'heuristics'; and practice is that of 'implement and test' (and iterate). De Souza et al. (1990) analysed HCI designers and stated that 'their [product] effectiveness is weakened by errors and difficulties experienced by designers in their use [of heuristics]'. Heuristics are poorly, if at all, conceptualised—often only through example practice—, which leads to them being difficult to operationalise, therefore unlikely to be generalisable or testable.

D&L's characterisation of the current state of the HCI discipline supports the general operational problem identified by this research. Currently available HCI knowledge is not effective, and has no guarantee of success of application in practice.

Future Development of the HCI Discipline

D&L compare the current HCI discipline with existing engineering disciplines, and propose that, to be more effective in the future, the HCI discipline should develop as an 'engineering' discipline. D&L propose that the knowledge of the engineering HCI discipline would be more effective, with a guarantee of application similar to that currently enjoyed by the existing 'hard' disciplines such as electrical or mechanical engineering.

D&L propose that engineering HCI knowledge would need to be conceptualised, with explicit, complete, consistent, and formal definitions, to be operationalisable, testable, and generalisable. D&L term this knowledge 'engineering general design principles', which will be referred to in this research as 'engineering principles'. HCI practice would become that of 'specify then implement'. D&L describe engineering principles as 'prescriptions ... which, when implemented, demonstrate a prescribed and assured performance'.

The acquisition of more effective HCI knowledge, particularly with a guarantee of application, would solve the specific and general operational problems, since:

- A guarantee of application would enable developers to place an appropriate emphasis on HCI input to the design process and hardware and software development. They could be convincingly *certain* of solving HCI problems.

- A 'prescribed and assured performance' would result in effective human-computer systems and improved productivity.

- A 'specify then implement' practice would not involve iteration.

Since more effective HCI knowledge would solve the operational problems of this research, its acquisition is taken as its technical problem. The development of engineering principles for HCI is taken as the long-term technical solution of this research. The technical solution is 'long term' because it is anticipated that acquiring engineering principles is difficult. Therefore, the technical aim of this research is to make progress towards the development of these engineering principles.

D&L have developed a conception of the general design problem of the engineering HCI discipline. This conception is relatively well operationalised (for example: Dowell, 1993; Stork and Long, 1994; Hill et al., 1995; Timmer and Long, 1996).

However, D&L have not developed a conception of engineering principles nor a strategy for their acquisition. This research builds on D&L's research, to develop a conception of engineering principles, a strategy for their acquisition, and an assessment of that strategy. The assessment of the strategy involves acquiring examples of engineering principles.

Requirement for Guarantee

There are alternative characterisations of HCI. However, these characterisations do not recognise the need for more effective HCI knowledge, with a guarantee of application. The following paragraphs outline the main alternatives to D&L's characterisation.

Carroll (1989, 1997) sees HCI as a 'design science'. He aims for iterative and historically-based improvement of HCI knowledge, as embodied in 'artefacts' and 'second-order artefacts (prescriptive design models, architectures and genres, tools and environments, interface styles)' (Carroll et al., 1991; Carroll et al., 1992). He does not address the requirement for guarantee or its delivery.

Gaines and Shaw (1986a and b) support D&L in terms of hardware and software engineering:

> 'It is time that we provided foundations of engineering human-computer interaction (HCI) as explicit and well-founded as those for hardware and software engineering.'
>
> 'We believe that the cutting edge of HCI research studies must now move to the provision of deep theories.'

Norman (1989), Diaper (1989), and Storrs (1989) recognise the need for an engineering discipline of HCI. However, they only implicitly recognise the requirement for guarantee. They all address the need for a conception of HCI knowledge. However, D&L's conception of the general design problem is broad enough to cover their conceptions.

Engineering and Craft

Stork (1992) considered that engineering and craft would co-exist in HCI practice, as in other hard engineering disciplines (Figure 2). 'User requirements' and 'artefact specification' are used as the input and output to HCI practice, when that practice includes craft HCI practice. The secondary direction of knowledge application in Figure 2 shows potential iterations during design.

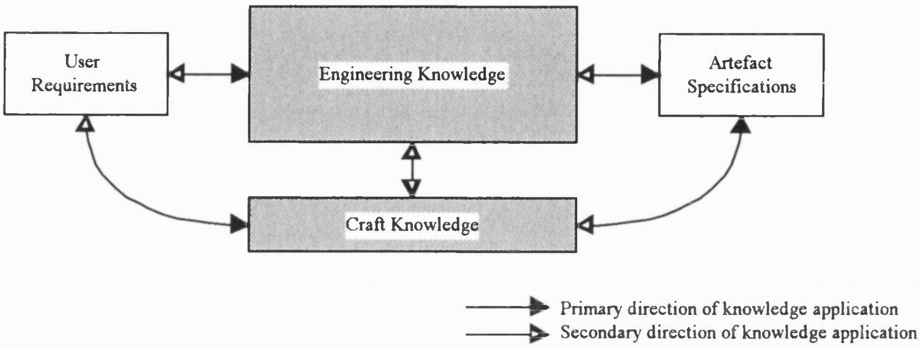


Figure 2. HCI Practice.

Acquiring Engineering Principles

Given the anticipated difficulty of acquiring engineering principles, their scope will be limited for this research to enable strategy implementation and assessment within the timeframe of this research. Their scope will be limited to:

1. Concentration on 'substantive' engineering principles, rather than 'methodological' engineering principles (see Chapter 3).

2. Acquisition of 'initial' engineering principles, which have not been validated by application (see Chapter 4).

3. Concentration on engineering principles for the specific operational problem, the design of domestic energy planning and control. Domestic EMSs are relatively simple systems, and planning and control appears to have potential for engineering principles (see Chapter 4).

3. Conception of (Substantive) Engineering Principles

The previous chapter identified that the technical aim of this research was to make progress towards engineering principles, through the development of a conception of engineering principles, a strategy for their acquisition, and an assessment of that strategy.

Part of D&L's proposal is a conception of the general design problem of an engineering discipline of HCI. This research requires a conception of engineering principles. This chapter presents D&L's conception. A conception of (substantive) engineering principles (meta-knowledge[1]) is developed for this research from the Dowell and Long conception. *Substantive* engineering principles 'prescribe the features and properties of artefacts' (D&L). Components of engineering principles are conceptualised: general design problems; general design solutions; specific design problems; specific design solutions; partial design problems; and partial design solutions.

Dowell and Long's Conception of the General Design Problem of the Discipline of HCI

D&L's conception '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 general design principles.' This conception is presented here, however reference to the original papers by D&L is recommended.

D&L state the general HCI design problem informally as 'the design of interactive worksystems for performance'. They propose a more precise description as follows (slightly amended for typographical considerations):

> 'The design of behaviours constituting a worksystem {S} whose actual performance (Pa) conforms with some desired performance (Pd). 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:

---

[1] 'Meta-level knowledge is knowledge about knowledge.' (Davis and Buchanan, 1977).

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

{U} interacting with {C} = {S} Pa=Pd

where Pd = fn (Qd, Kd)

Qd expresses the desired quality of the products of work within the given domain of application;

Kd expresses acceptable (i.e., desired) costs incurred by the worksystem, i.e. by both human and computer.'

This statement expresses D&L's distinction between the behavioural system that is the interactive worksystem, hereafter 'worksystem', that performs work, and the world of work, the domain of application, within which the work is performed (Figure 3). It follows from their conception that Pa is a function of the actual quality of the products of work within a particular domain of application (Qa) and the actual costs incurred by a particular worksystem (Ka).

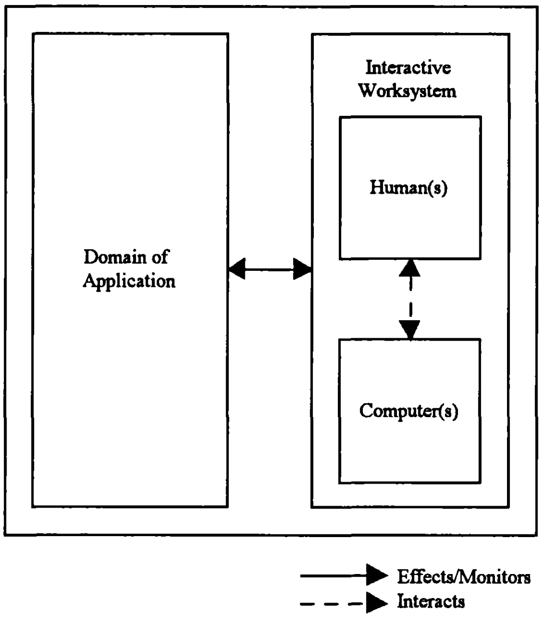


Figure 3. Behavioural System and World of Work Distinction.

Conception of (Substantive) Engineering Principles

The development of engineering principles requires the development of knowledge to support HCI practice. HCI practice is the provision of artefact specifications to user requirements.

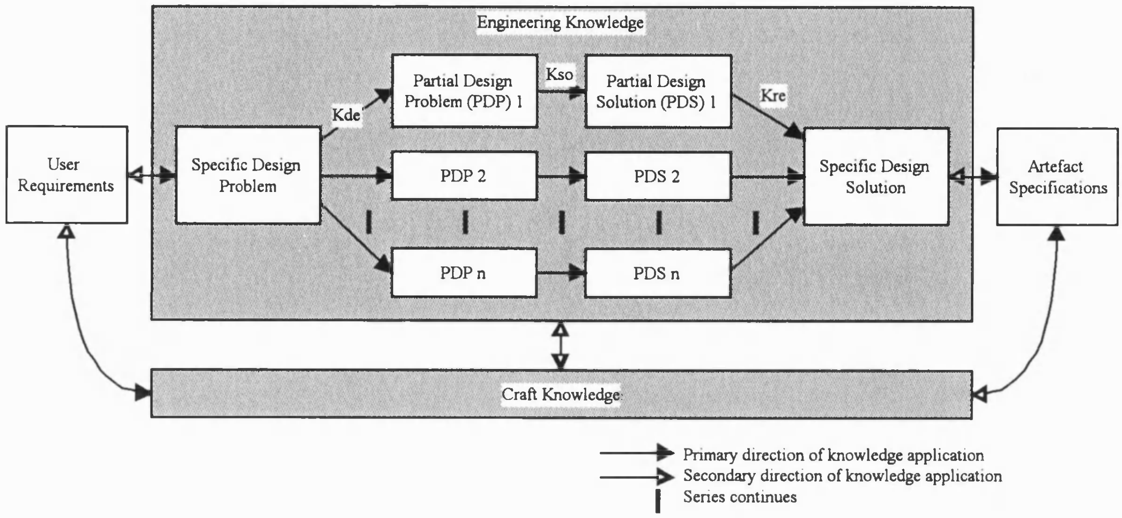


Figure 4. HCI Engineering Practice.

The engineering knowledge applied during practice is conceptualised as producing: a specific design problem operationalisation; partial design problem operationalisations; partial design solution operationalisations; and a specific design solution operationalisation (Figure 4). The partial design problem and solution operationalisations are the instantiations of *a*[2] general design problem and its general design solution (Figure 5). The specific design problem and solution operationalisations represent the scoping of the engineering HCI discipline by comparison with craft HCI. The partial design problem and solution operationalisations represent the application of HCI engineering knowledge. They are 'partial' because they solve only *part* of the specific design problem.

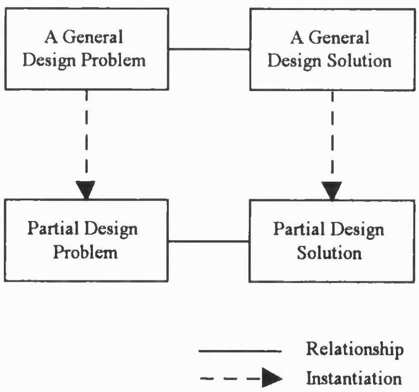


Figure 5. HCI Engineering Knowledge.

[2] *A* general design problem is contrasted here with *the* general design problem.

This research restricts itself to substantive rather than methodological engineering principles. D&L distinguish between substantive and methodological engineering principles (following Checkland, 1981; and Pirsig, 1974):

> 'Methodological principles prescribe the methods for solving a general design problem optimally. ... Substantive principles prescribe the features and properties of artefacts, or systems that will constitute an optimal solution to a general design problem.'

Three types of engineering principles can be identified:

- Decomposition knowledge (Kde) is conceptualised as the means of instantiating a partial design problem from a specific design problem. Kde requires substantive knowledge of the general design problem of which the partial design problem is the instance (*a* general design problem; Figure 5).
- Solution knowledge (Kso) is conceptualised as the means of instantiating a partial design solution from a partial design problem. Kso requires substantive knowledge of the general design solution to the general design problem identified in Kd (*a* general design solution; Figure 5).
- Recomposition knowledge (Kre) is conceptualised as the means of instantiating a specific design solution from partial design solutions. The assured prescription of the substantive Kso implies that recomposition would be prescribed, and so no substantive knowledge is required for Kre.

The specific design problem and solution may not be required for engineering design practice. It may be possible to instantiate a partial design problem from the user requirements using Kde and it may be possible to instantiate part of the artefact specification using Kre. However, it is expected that they are required at least for further research work, as part of research design practice.

The substantive knowledge required—for Kde, Kso, and Kre—is a general design problem and its general design solution, which are conceptualised by a general desired performance and a general actual performance respectively. A general design problem and its general design solution are general over types of user, types of computer and types of domain of application. Desired performance and actual performance are conceptualised further below, following D&L's conception of the general HCI design problem.

Conceptions of the General Design Problem and Solution

The general HCI design problem requires a statement of the desired performance for the desired worksystem, whereas a statement of the general HCI design solution requires a statement of the actual performance for the actual worksystem.

Desired performance and actual performance are conceptualised following D&L's conception of performance. Important occurrences of the concepts are highlighted in bold to aid their identification. The concepts taken from D&L are in italics and the quotations are from D&L (1989).

Conception of Desired Performance

The ***desired performance***, Pd, is conceptualised as a function of the **desired *quality*** of the products of work, Qd, within the domain of application and the acceptable or **desired *costs***, Kd, incurred by the worksystem.

The ***worksystem* boundary criteria** allow statement of the behavioural system which constitutes the ***worksystem***, that system 'whose purpose is to achieve and satisfy ... common goal[s]'. The ***domain* boundary criteria** allow assertion of the world of work that constitutes the domain of application, that world of work which is determined by the requirement to express these common goals.

Conception of Actual Performance

Actual performance, Pa, is conceptualised as a function of the **actual *quality*** of the products of work, Qa, within the given domain of application and the current or **actual *costs***, Ka, incurred by the worksystem.

The ***worksystem* boundary criteria** and ***domain* boundary criteria** are the same as for the conception of desired performance.

Conception of Desired Quality

D&L conceptualise the world of work as consisting of ***objects*** that have ***attributes*** that have a set of possible ***states*** (defining their affordance for change). The **desired *quality*** of the products of work to be achieved by the worksystem are conceptualised as transformations of states of attributes of objects that are desirable, called ***product goals***. These objects and their attributes are conceptualised as ***abstract*** or ***physical,*** and ***related*** or ***unrelated***. The transformations described by a product goal can be identified for each attribute, and these transformations are termed ***task goals***.

D&L describe the difference between abstract and physical attributes of objects as 'abstract attributes of objects are attributes of information and knowledge' and 'physical attributes of objects are attributes of energy and matter'. They also propose that 'different attributes of an object emerge at different levels within an hierarchy of levels of complexity' and, in general, abstract attributes emerge at a higher level than physical attributes. Similarly, 'objects are described at different levels of specification commensurate with their levels of complexity'. Furthermore, attributes of objects are related to attributes of other objects both between and within levels of complexity.

Conception of Actual Quality

The **actual** ***quality*** of the products of work achieved by the worksystem are conceptualised as similar to desired quality, with transformations of states of attributes of objects that are achieved, called ***product*** **achieved** ***goals***, and transformations for each attribute, called ***task*** **achieved** ***goals***.

Conception of Desired Costs

D&L conceptualise the worksystem (the behavioural system) as 'human and computer ***behaviours*** together performing work'. They make a distinction between human behaviour as purposeful and computer behaviour as purposive. They claim that human behaviours correspond with the transformation of objects in a domain and that an expression of them must 'at least be expressed at a level commensurate with the level of description of the transformation of objects in the domain'. These statements would appear to hold for computer and worksystem behaviours.

These behaviours can be ***abstract*** or ***physical***. 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 express abstract behaviours and are 'related in an hierarchy of behaviour types'.

D&L conceptualise the user as having ***cognitive***, ***conative***, and ***affective*** behaviours. 'The cognitive aspects of the user are those of knowing, reasoning and remembering, etc.; the conative aspects are those of acting, trying and persevering, etc.; and the affective aspects are those of being patient, caring, and assuring, etc.'

D&L conceptualise humans and computers as 'having (separable) ***structures*** that support their (separable) behaviours'. Furthermore, 'Human structures may be ***physical*** (neural, biomechanical, and physiological) or ***mental*** (representational schemes and processes)'. Similarly, computer structures may be ***physical*** or ***abstract***.

D&L claim that 'work performed by worksystems always incurs resource costs'. They identify resource costs as behavioural or structural and associated with the human or the computer (separable). These costs can be further associated with abstract (mental) and physical behaviours or structures. Examples of resource costs related to the human are: physical workload for ***human physical behavioural costs***; mental workload for ***human abstract (mental) behavioural costs***; physical development and deterioration for ***human physical structural costs***; and mental development and deterioration for ***human abstract (mental) structural costs***. Examples of resource costs related to the computer are: energy emission and consumption for ***computer physical behavioural costs***; software and functional resource (transaction and access resources) usage for ***computer abstract behavioural costs***; system (hardware) development and degradation for ***computer physical structural costs***; and software and functional development (and degradation) for ***computer abstract structural costs***.

The **desired** ***costs*** are conceptualised as the necessary resource costs of the worksystem to achieve the desired task quality.

Conception of Actual Costs

The **actual** ***costs*** are conceptualised as the actual resource costs of the worksystem to achieve the actual quality.

Conceptions of the Specific Design Problem and Solution

The conceptions of the specific HCI design problem and solution are operationalised from the conceptions of the general HCI design problem and solution. The specific HCI design problem and solution are particular, by definition, to an instance of HCI design, termed a 'design situation'.

The specific desired performance is conceptualised as a function of the desired quality of the products of work within a particular domain of application and the desired costs incurred by a particular worksystem.

The specific actual performance is conceptualised as a function of the actual quality of the products of work within a particular domain of application and the actual costs incurred by a particular worksystem.

Conception of (Substantive) Engineering Principles Revisited

Engineering principles achieve, or exceed, prescribed performance on application (as in D&L's 'Pa = Pd'). The conceptions of a general design problem and its general design solution can be combined to produce a single conception of a substantive engineering principle. Any expression of the domain, actual task quality, and actual costs are not required for a general design solution (or for its partial design solution), since they will be the same as those for its general design problem (or for its partial design problem). Therefore, the only component of the actual performance of a general design solution that is not expressed by the desired performance in its general design problem are those structures and behaviours of the worksystem required to achieve that desired performance. A substantive engineering principle is conceptualised, therefore, as the desired performance of a general design problem and the structures and behaviours of its general design solution.

Informal Assessment of Dowell & Long

The possibility, difficulty, or impossibility of acquiring engineering principles in this research will have implications for D&L's characterisation and conception of the engineering discipline of HCI. For example, the development of HCI engineering principles by this research would support D&L's characterisation and conception.

The chapter has presented a conception of (substantive) engineering principles developed from Dowell and Long's conception of the general design problem of HCI.

4. Strategy for Developing Engineering Principles

The previous chapter presents a conception for substantive engineering principles. This chapter proposes a strategy for developing such engineering principles and compares it with alternative strategies. The proposed strategy is selected and developed in further detail.

This research aims to assess the proposed strategy by developing examples of 'initial' engineering principles, which have not been validated by application. The rationale for scoping this research to 'planning and control' engineering principles is presented.

Additional shorter-term benefits of the strategy are identified.

Strategy for Developing Engineering Principles

One possible means of developing substantive engineering principles is:

- To identify general relationships between specific design problems and their solutions. These general relationships would be considered putative, i.e. requiring validation, and termed 'initial' engineering principles. The identification of general relationships between specific design problems and their solutions requires the operationalisation of specific HCI design problems and their solutions from the conceptions of specific HCI design problems and specific HCI design solutions.

- To validate initial engineering principles by testing.

This research aims to acquire examples of initial engineering principles by implementing this strategy to support its assessment. The research will operationalise two specific design problems and their solutions (Cycle 1 and Cycle 2), as the minimum necessary for generality. Assessment of this strategy consists of:

- Acquisition or not of initial engineering principles.

- Assessment of the status of acquired initial engineering principles.

- Discussion of the strategy and conceptions following acquisition, or not, of initial engineering principles.

D&L claim that for the knowledge to support engineering principles, the operationalisation of the specific design problems and solutions, needs to be explicit and formal. Formal, here, is understood as a representation that has defined rules of syntax and semantics (following Bowers, 1992), and is therefore understandable by some people for some purpose. Formality requires the metrication of the operationalisation of the conceptions of the specific HCI design problem and solution. Metrication was defined in Chapter 1 as "Metrication is the process of instancing an operationalisation to its limit, to produce metrics. Metrics quantify the less abstract concepts of the operationalisation in an observable relation with the 'real' world".

To operationalise specific design solutions to specific design problems, the following methodology is proposed:

- That appropriate 'user requirements' are selected for each Cycle (this chapter).
- That an artefact specification is developed to solve the user requirements for each Cycle using best current HCI practice (Chapter 8 for Cycle 1 and Chapter 10 for Cycle 2).
- That the specific design problem and its solution are operationalised from a design situation based on the user requirements and its artefact specification (Chapter 9 for Cycle 1 and Chapter 11 for Cycle 2).

Engineering principles were conceptualised as the desired performance of a general design problem and the structures and behaviours of its general design solution. The strategy, however, is not to limit the operationalisations to these concepts, in order to provide:

- A check that the specific design solution is a solution to the specific design problem (i.e. by explicit representation of the actual performance to compare with the desired).
- Establishment of the relationships between the specific design solution structures and performance (a check for the solution).
- The availability of the research products for further research, some of which might aim for a lower prescribed performance.

The research has been restricted to the acquisition of *cognitive* engineering principles, rather than *conative* or *affective*, since cognitive processes and representations are relatively well-defined.

Additional Concepts

For the research to operationalise specific design problems and solutions, a conception of human-computer systems costs is required. Human-computer costs are poorly conceptualised in the conception of specific design problems and solutions relative to the conception of task quality. Chapter 5 presents a conception of human-computer costs developed for this research.

Current Solutions

It is considered easier to operationalise existing, installed, specific design solutions, referred to here as 'current solutions'[3], than to operationalise specific design problems. The operationalisation of the current solution can provide a basis for the operationalisation of the specific design problem, because the desired performance of the specific design problem is likely to be similar to the actual performance of the current solution. Each cycle operationalises the current solution before the operationalisation of the specific design problem and its solution (Chapters 9 and 11). The selection of re-design user requirements supports the operationalisation of a current solution.

Tractable user requirements are susceptible to current design knowledge. The selection of tractable user requirements ensures that the differences between the operationalisation of the current solution and the specific design problem be minimal, and that a specific design solution exists.

SuperCraft Design

Best current HCI practice, which could be characterised as 'SuperCraft', for developing design solutions is considered the application of structured methods, current design guidelines, and evaluation. This research is, therefore, using a Human Factors (HF) structured method called MUSE, a Method for USability Engineering[4] (Lim and Long, 1994), applied by an HF designer. The researcher is a qualified and experienced HF designer with expertise in MUSE.

---

[3] Current solutions are not to be confused with the specific design solutions that solve the specific design problem, and that might be installed after the specific design problem.

[4] MUSE was developed to solve the 'too-little, too-late, and unimplementable' contribution of HF to system development.

MUSE is usually complemented by a Software Engineering (SE) method. Cycle 1 used the SE structured method called JSD, which was the first method configured for use with MUSE. However, it was found that the JSD products (Appendix F) were not required for the artefact specification to support the operationalisation of the specific design solution. This finding is probably because the Cycle 1 user requirements are relatively simple and because the researcher is a qualified and experienced Software Engineer. The Cycle 2 user requirements are also relatively simple, so JSD was not used for Cycle 2.

Comparison with Alternative Strategies

The strategy proposed above for developing HCI engineering principles can be characterised as 'bottom-up' and cautious, or sceptical (Stork and Long, 1998). This cautious approach means that steady progress is made towards engineering principles, or the abandonment of the research direction. However, this cautious approach means that engineering principles may not be found during an initial period.

An alternative 'top-down' strategy would be to postulate operationalisable and testable engineering principles from a conception of engineering principles, which are then operationalised and tested. This strategy could be considered bold given our current understanding of the nature of engineering principles. The likelihood of locating an engineering principle would be low, although the effort for each attempt would appear less than with the 'bottom-up' strategy. This strategy is not recommended, because the low likelihood of locating an engineering principle does not sufficiently merit the effort for each attempt.

A further alternative 'middle-out' strategy would be to develop a conception of the general design problem and solution for a simple design 'world', i.e., the design of simple shapes, depending on a small set of requirements. These conceptions could be used to develop 'engineering principles' for this simple world, through either of the above strategies. This route might provide insight into the nature of HCI engineering principles, but scaling up would require the adoption of one of the other two strategies.

Dowell (1993) proposes an additional type of strategy based on an understanding of the discipline of HCI. He suggests that engineering principles could be used for diagnosis in addition to prescription. Such a strategy is based on an understanding that scientific knowledge offers explanation in addition to prediction. This proposal suggests that diagnosis, the formulation of partial design problems based on current solutions, could be used to develop engineering principles. This strategy was rejected for two reasons: current best practice craft HCI knowledge is better at developing artefacts to user requirements than at diagnosis; and it is considered easier to describe design problems when the problem still exists. Interestingly, the example that Dowell uses appears to be similar to the 'bottom-up' strategy adopted in this research, rather than the 'diagnostic' strategy he describes.

Colbert (1994a) has attempted to deliver informal 'immature HCI Engineering'. His strategy can be understood as a form of the 'top-down' strategy combined with the 'bottom-up' strategy. For 'top-down', he develops a general description of the 'domain' of military planning. For 'bottom-up', he postulates, using craft practice, a general menu design for the computer support of military planning. The approach appears positive, with the acquisition of a possible informal general principle, which was tested by specifying two particular menu designs for the computer support of two specific military planning situations. Guarantee is poorly addressed due to the informality, but the strategy could be developed with a view to greater formality. However, the generality of the domain was assumed and that assumption was poorly tested, so it is difficult to assess the 'top-down' part of the strategy.

In addition to the above, several researchers have attempted domain and worksystem descriptions without the aim of engineering discipline or principle development. For example, Hill et al. (1995) describe the domain and worksystem of medical reception and Jenkins et al. (1993) describe the domain of domestic work. These researchers claim that these descriptions support current design practice.

Scoping the Research using the Potential for Planning and Control Engineering Principles

In order to scope the research to the resources available, it has been decided to concentrate on the potential for planning and control engineering principles. Planning and control can be contrasted with alternative potential scoping, such as 'training', 'opportunism', 'coordination' (e.g. Lambie et al., 1998), 'pattern-recognition', etc. Further, planning and control is of interest to the researcher and to the sponsors. There is sufficient craft planning and control practice that there are likely to be planning and control engineering principles. This last claim rests on computer support for the following fields: military planning; aircraft flight planning and control; office administration; project management; business decision making; and clinical decision-making. The following paragraphs identify particular research in these areas.

Computer support for military planning and control has been investigated by Colbert (1994b). He reported that it was possible to develop a generic menu structure for military planning tasks. He claimed that this generic menu structure could be integrated with other generic user interface objects (see Rosenberg and Moran 1982), and then instantiated for two hypothetical specific military planning systems. He claimed that this generic menu structure was problem-dependent, supporting the military planning problem; in contrast to normal generic interface objects, which are claimed to be problem-independent. His claim suggests that general principles may be developed for computer support for military planning.

Analysis of the domain of aircraft flight planning and control by Dowell (1993) allowed him to reason informally about a possible software replacement for the physical flight strips, used in a specific aircraft flight planning and control example. The reasoning included references to aspects of the software replacement that were related to the controller's planning and control. These relationships suggest that general principles may be developed for computer support for aircraft flight planning and control.

Analysis of the planning and control of multiple task work at the reception of a medical practice led Hill et al. (1995) to propose guidelines that they claimed should reduce the cost of behaviours related to planning. These proposals suggest that general principles may be developed for computer support in such multiple task planning and control situations.

Analysis of planning in project management by Pietras and Coury (1994) prompted them to remark that 'one can conclude that theoretical models of planning are relevant to project management and useful in the design of planning systems'. Although the discussion in their paper does not adequately support this conclusion, it does suggest that general principles may be developed for computer support for planning in project management.

Several analyses (Rouse and Greenstein, 1976; Halé, 1986; Surunjan, 1986; Keen, 1987) of computer support for business decision-making, or planning, suggest that successful support has been achieved and that there is further potential. Some of these analyses contain recommendations for computer behaviour that it is claimed support business decision-making. These studies indicate that general principles may be developed for computer support for business decision making.

The claimed success and potential of two alternative approaches to computer support for clinical decision-making—process tracing and probability methods (Dowie and

Elstein, 1988)—suggests that general principles may be developed for computer support in this area.

Analysis of text editing by Robertson and Black (1986) shows that the operation of a text editor requires planning. Experienced users generated better plans, and took less time to edit a document, than novices. This finding suggests that general principles may be developed for computer-supported text editing that requires planning.

Although it is possible that the above all relate to different general principles, it is more likely that, together, they provide sufficient support towards the generation of general engineering principles for planning and control. Therefore, this research is scoped by concentrating on the potential for planning and control engineering principles. This scoping is delivered by operationalising planning and control concepts for the specific design problems and their solutions. These planning and control concepts have been developed for this research and are presented in Chapter 7.

Acquiring Potential Guarantee

Ultimately, the guarantee of engineering principles rests on the effectiveness of their support for practice, i.e. their validation through testing. However, initial engineering principles need to acquire potential guarantee to support validation. Following D&L's properties of effective HCI knowledge—conceptualised, operationalisable, generalisable, and testable—the pre-requisites for acquiring potential guarantee are that initial engineering principles:

- Are conceptualised according to a conception of the discipline of HCI. The conceptions developed for this research are all based on D&L's conception of the general design problem of the discipline of HCI.

- Are operationalisations of those conceptions. The cycle operationalisations will identify the concepts operationalised to support an informal checking.

- Are generalised. The generalisation for this research will be over the two cycles.

- Are tested. The testing for this research will be informal evaluations of the two cycles.

Shorter-Term Research Benefits

The main thrust of this research strategy, and consequent research knowledge, is to develop HCI knowledge for long-term HCI practice. Several shorter-term research benefits may be met in the process. A medium-term research benefit may be the development of a version of MUSE that supports a more complete, coherent, and consistent specification of the design problem and solution. Such a version of MUSE would improve HCI practice (see Chapter 14). A short-term research benefit may be the development of conceptions and operationalisations of design problems and solutions, which could assist practitioners to identify and assess design problems and solutions better.

Further short-term research benefits of MUSE assessment, MUSE applications, example solutions, and guidelines will be possible.

Overview of MUSE

MUSE is a structured analysis and design method for use by human factors engineers. The method aims to improve the practice of HCI practitioners by providing support for the integration of human factors with existing structured methods for software engineering, such as JSD, Yourdon, or SSADM. The output of MUSE is the specification of an interaction artefact. The software engineering method produces the specification of an implementable artefact, which incorporates the interaction artefact.

MUSE supports design in a 'top-down' manner based on information derived 'bottom-up'. Application progresses from the specification of general features of the tasks to be performed by the user, derived from analysis of the user requirements and from existing systems, to the specification of the details of the interaction artefact. The application of MUSE is an iterative process, both overall and internally, supporting the production of the best first-attempt artefact, following the initial complete application.

Figure 6 shows a schematic diagram of MUSE with an unspecified software engineering method. MUSE has three phases: the Information Elicitation and Analysis phase; the Design Synthesis phase; and the Design Specification phase. The Information Elicitation and Analysis phase supports the assessment and re-use of components of extant systems and the maintenance of the consistency of the design with the user requirements. The Design Synthesis phase supports the conceptual design of the interaction artefact and the maintenance of the consistency of the design. The design is kept consistent with the semantics of the domain and a human factors interpretation of the user requirements with respect to the analysis of extant systems. The Design Specification phase supports the detailed design of the interaction artefact. Mandatory checking and exchange of information with the software engineering method occurs to ensure that the interaction artefact is implementable and to support overall design agreement and consistency.

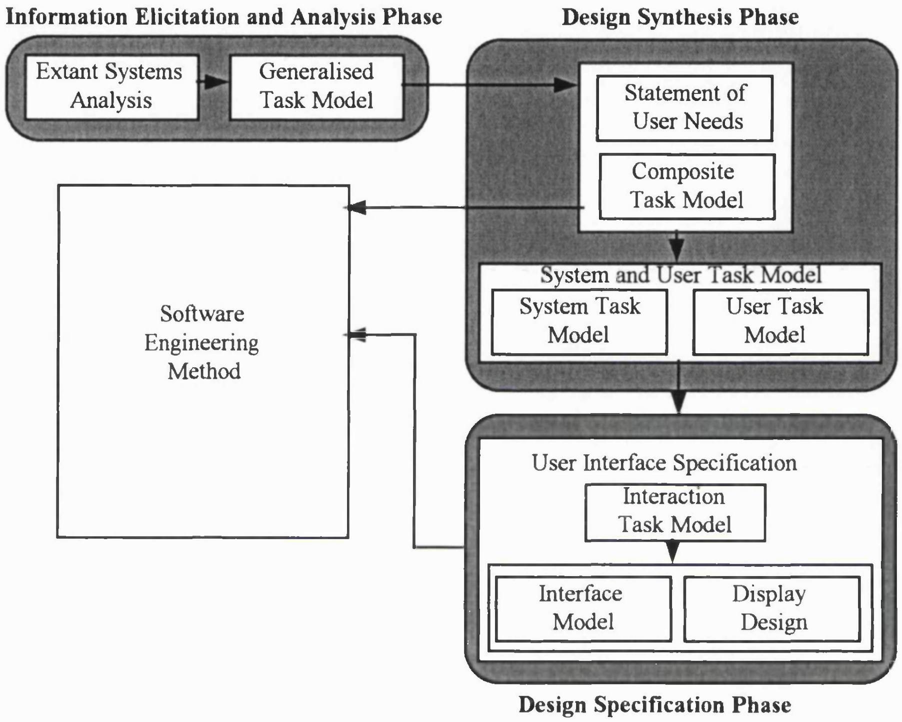


Figure 6. MUSE Overview.

Cycle User Requirements Selection Criteria

Tractable, re-design, and relatively simple user requirements should be selected that offer repetition, access, interest, and generalisation potential. The selection criteria are considered further in the following sections. A questionnaire (Appendix B) was prepared to identify potential user requirements for each cycle.

The user requirements are intended to be operationalisable according to the conception of the specific design problem in order to prevent concentration on the relationship between the user requirements and the operationalisation of the specific design problem.

Tractable Re-Design

In the Current Solutions section above (Page 39), it was decided, as part of the strategy, to select tractable re-design user requirements

Relatively Simple

The conceptions of the specific design problem and solution are expected to be difficult to operationalise. Therefore, relatively simple user requirements should be selected to support relatively simple operationalisations. For Cycle 2, some increase in complexity over Cycle 1 could be appropriate.

Repetition

The user requirements should be selected such that the design situation will be maintained for the design and operationalisation period.

Access

Access to the user requirements situation—before, during, and after the design process—should be as high as possible to permit: cycle selection; best-practice design; evaluation; and the operationalisations.

Interest and Generalisation Potential

The main interest of the sponsors is the development of more effective energy management systems. Planning and control is a feature of most energy management systems. Selection from user requirements for energy management systems should support potential generalisation.

Cycle 1 Selection

Identified User Requirements

A, the researcher, identified the following broad user requirements with his home heating. He lives with *S*. Further background information on *A*, *S*, and the heating system is to be found in Appendix C. The problems are skewed towards being too cold, since most of the observation was performed during the winter, and this fact is taken into account during selection.

- Different tasks can require very different conditions for comfort. Any particular room is not always comfortable because the task being performed in the room can change faster than the conditions in the room. This problem is more noticeable in the rooms downstairs, which do not have individual heating controls.

- The two occupants of the house do not require the same level of comfort in all situations.

- The heating in the morning at weekends is nearly always too hot while the occupants are still in bed and then too cold on rising.

- If the occupants are up late in the evening or friends visit then the house can become cold, unless the heating is switched back on.

- If the occupants are out late in the evening, then the house is cold on their return.

- After either or both of the occupants have returned by bicycle they are usually too hot in the house. Occasionally *S* is too cold on her return by bicycle.

- If *A* leaves after 8 a.m. or stays at home to work, then the house is too cold until he turns the heating back on. If he expects to be at home for a short time, then he often uses the boost facility, which can result in him being too cold if he is at home for longer than expected.

The current gas bill is acceptable for the comfort; an increase could be considered acceptable for greater comfort. A decrease in the gas bill for the same comfort or better would be desirable.

Comparison against the Criteria

All of the identified user requirements are tractable re-design user requirements with repetition, access, interest, and generalisation potential.

The last identified user requirements were selected for Cycle 1, since they are relatively simple, being based primarily on *A*. They have repetition, appearing to be time-invariant, if taken for days of a specific outside condition or worse, with a reasonably constrained set of factors that are not invariant.

Cycle 2 Selection

Four types of situation that might yield user requirements for Cycle 2 were considered:

- A trial site for the sponsor's energy management system.
- Another home whose occupants are known to the researcher, which would be similar but different, improving the potential generality over the cycles.
- *A*'s home. One of the user requirements not selected in Cycle 1 could be selected for Cycle 2. However, none of them were selected for Cycle 1.
- *A*'s car. It was felt that for generalisation with Cycle 1 a home would be more appropriate.

It was beneficial for this research to have user requirements that related to the sponsor, the fourth option. User requirements that related to the sponsors would link this research to state-of-the-art energy management system technology. The sponsor would benefit because this research would offer a shorter-term return to the sponsor. However, access was a problem as the sponsor's trial sites were located in France and Spain.

Therefore, the Cycle 2 user requirements were selected from the second option, those in another home whose occupants are known to the researcher. However, if feasible, consideration should be given to the selection of user requirements comparable with those potentially required for the sponsor's energy management system, to deliver some of the benefits mentioned above. Appendix A contains an analytical evaluation and potential user requirements for a prototype of the sponsor's energy management system.

Identified User Requirements

The following broad user requirements have been identified after discussion with the occupants of the home, *D* and *J*. Further background information on *D*, *J*, and the heating system is to be found in Appendix D. These problems are skewed towards being too cold, since most of the observation was performed during the winter, matching
Cycle 1.

- The main house is too cold if only one of its boilers is started for the early mornings, since the following areas are always accessed in the morning and their radiators are supplied by different boilers: the kitchen; the front porch, accessed for the mail and newspapers; and the downstairs toilet.

- Study 1 and studio (Study 2) are always cold for sedentary working since the radiators are badly located for rooms with external walls. The rooms are comfortable once warmed using fan heaters.

- The sitting room can be cold on winter evenings, particularly if the boiler supplying the radiators in the sitting room has been off during the day.

- The dining room can be too hot when there are many people in it. It is undesirable to open the window, since it faces the prevailing wind.

- The kitchen is a comfortable room with thick walls that retain the heat. However, it can become too hot during cooking, particularly in the summer, but also in winter. The windows are all fitted with security locks.

- *D* can feel cold while working, as she requires a warmer temperature than when she performs other, more physical, tasks (e.g. cooking or housework), and warmer than *J* requires.

- *D* often works in the cottage, a small property attached to the main house, since she can control the heating more easily: it is separate from *J*'s heating requirements; and she finds the controls easier to use. She usually knows in advance that she will be working in the cottage. She has to walk across the garden to turn the heating on, or up, before returning to work after the cottage has warmed up. She normally leaves the heating on in the main building for her return, even if *J* is out.

- *J* tends to turn the heating off if he is going out for the day or longer. *D* tends to leave it on, so that it is warm on her return.

- *J* turns the heating off on April 1st for summer. *D* would prefer it on, since she is sometimes cold in summer.

- The timers are all difficult to adjust, being mechanical, situated separately in cupboards at each end of their home. The occupants felt that the controls required moving and improving, with separate weekend times and digital controls. They have installed the wiring to put the two main house controllers in the lobby.

- Ventilation is very poor throughout the house.

The heating costs seem high, but there are no standards for comparison. Any reduction would be welcome. Any improvement should not cost more than the gas bill reduction.

Comparison against the Criteria

The following user requirements are selected:

> 'The kitchen is a comfortable room with thick walls that retain the heat. However, it can get too hot during cooking particularly in the summer, but also in winter. The windows are all fitted with security locks.'

These are tractable re-design user requirements that have good access, interest, and generalisation potential. They are relatively simple since there are few conflicting needs and they are not based mainly on the technology. They are marginally more complex than the Cycle 1 user requirements. They relate to the sponsor's potential user requirements, since they match approximately one of the potential user requirements, 'too hot during some tasks when heating on', of the sponsor's energy management system.

This chapter has developed a strategy for the development of (substantive) engineering principles that are scoped by the potential for planning and control engineering principles.

The selected strategy involves identifying general relationships between specific design problems and their solutions, based on cycles of current HCI best-practice development and operationalisation of that development as specific design problems and solutions. Two cycles and the acquisition of examples of initial engineering principles are proposed to assess the strategy. Current HCI best practice includes MUSE—a Method for USability Engineering (Lim and Long, 1994). The user requirements for the two cycles have been selected.

Several shorter-term benefits of the strategy have been identified.

5. Conception of Human-Computer Systems

The previous chapter proposed a strategy for developing engineering principles. Part of this strategy is the operationalisation of specific design problems and their solutions from their conceptions. Since human-computer system costs are poorly conceptualised relative to task quality in those conceptions, this chapter proposes an initial conception of human-computer systems and their costs.

According to D&L, costs are conceptualised as associated with the human or the computer, and separable. Separable for this research means that the human and computer costs can be conceptualised separately, but also means that they need to be able to be integrated. This chapter starts with a conception of the worksystem, to support integration, and continues with separate conceptions of the human and the computer.

Interactive Worksystem Costs

Human and computer behavioural costs are conceptualised as arising from each behaviour occurrence. D&L conceptualise human and computer structural costs as initial and ongoing. Initial structural costs arise from the initial processes and representations required and present at the start of the design problem or solution. Initial processes are conceptualised as including the ordering of the behaviours during the design problem or solution. Ongoing structural costs arise from the development or change in state of processes and representations during the design problem or solution.

All costs are initially conceptualised as unitary (following Dowell, 1993) and, so, non-dimensional. Each behaviour occurrence incurs one unit cost. Each initial process and representational structure incurs one unit cost. Each ongoing structural change incurs one unit cost. Further research can consider non-unitary costs; for example, French (1990) describes early assessment of the cost of providing torque by any engine.

Potential Human Cognitive Structures

Timmer and Long (1997) propose an 'operator mental architecture', based on a computational cognitive architecture (Holland et al., 1987). The Timmer and Long architecture has been selected as the human cognitive structures for this research. It was selected because:

- It is relatively simple; for example, the process of 'problem solving' is not conceptualised further.
- It employs some concepts from the general design problem of HCI, including the distinction between domain and worksystem, user ('operator') and computer ('device'), and structure and behaviour.
- It has been employed, with some success, for design diagnosis in Air Traffic Management, particularly to identify potentially ineffective cases of planning (Timmer, 1999).

Others were considered but rejected, including Barnard et al. (1988), which has a 'performance' concept, but no design examples. Timmer and Long describe the architecture as follows:

> 'The … architecture distinguishes four classes of mental structure: storage; process; transducer; and representational. … Three major storage structures are specified: long-term memory; working memory; and a goal store, accommodating a single active goal. Eleven process structures are loosely associated with particular storage structures: 'decay' and 'store' in long-term memory; 'form', 'pop', 'suspend' and 'reactivate', for goal management in the goal store; and higher level processes of 'categorise', 'problem-solve' and 'evaluate' in working memory. A single mental processor is assumed in working memory. An input transducer, with an associated 'encode' process, maps environmental stimuli into a mental code. An output transducer, with an 'execution' process, maps an action specification into physical behaviour.'

Figure 7 shows the cognitive architecture and its relationship with the human physical architecture, which is described in the next section.

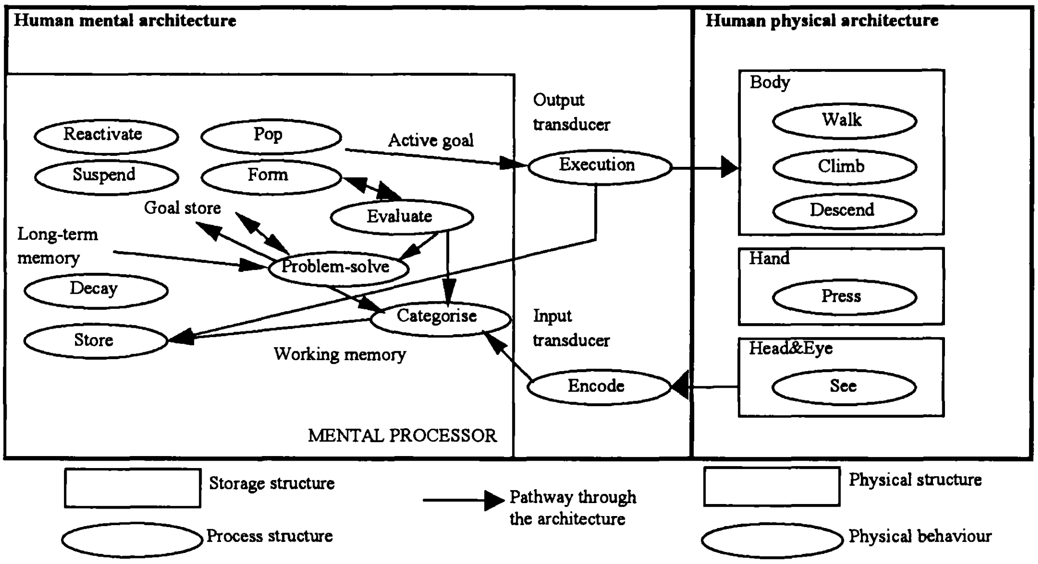


Figure 7. Human Architecture.

Potential Human Physical Structures

The emphasis in this research is on cognitive structures and behaviours. The human physical 'architecture' is conceptualised as any part of the human body, or the body itself, required for operationalising the specific design problem and solution. Figure 7 shows the selection for the first cycle.

Potential Computer Abstract Structures

A computer 'architecture' is conceptualised in a similar manner to the human architecture of the previous section. Figure 8 shows the proposed computer architecture, based on a Von Neumann computer architecture, for the first cycle.

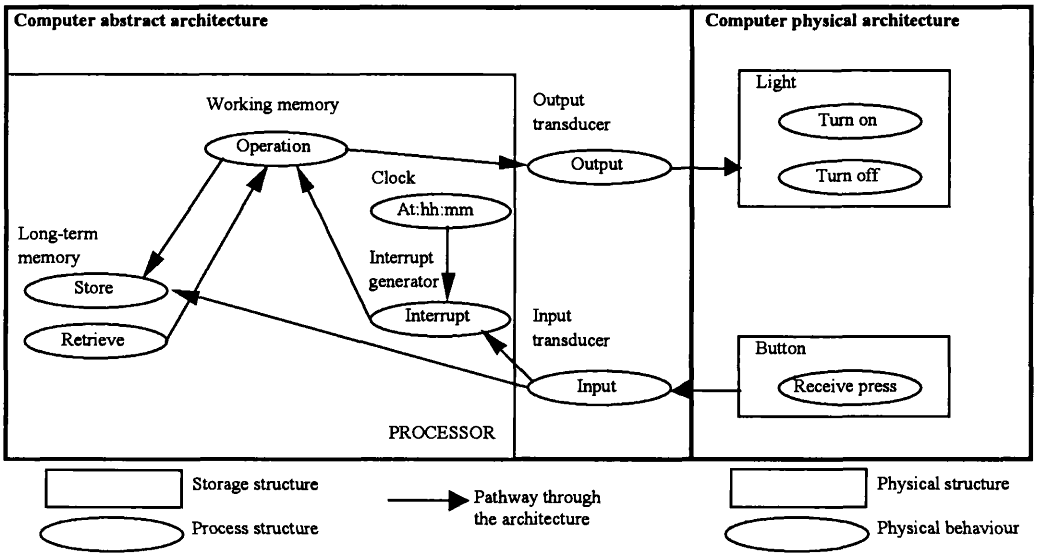


Figure 8. Computer Architecture.

Potential Computer Physical Structures

The computer physical 'architecture' is conceptualised as any device required for operationalising the specific design problem and solution. Figure 8 shows the selection for the first cycle.

This chapter has presented a conception of human-computer system costs. The unitary costs of the human-computer system arise from the occurrence of each behaviour, each initial structure, and each ongoing structural change.

The potential human-computer system structures, or 'architectures', are conceptualised. These structures support the potential human-computer system behaviours. These structures and behaviours are considered 'potential', because they offer an initial view to be validated by engineering principles.

6. Operationalising Specific Design Problems and Solutions

Chapter 3 presented conceptions to support operationalisations of specific design problems and solutions. Chapter 5 presented a conception of human-computer systems and their costs. This chapter presents frameworks developed for this research to enable operationalisations of these conceptions. These frameworks include the layout and scope of diagrams and tables for the metrication of these operationalisations.

This chapter presents a framework for task quality, including the domain, and a framework for worksystem costs, including their human-computer structures and behaviours. Composite structures are defined as groups of processes that occur repeatedly. The composite structures developed during this research are presented in this chapter.

Framework for Task Quality

The states of the task quality, product goals, and task goals are conceptualised by numerical or Boolean values over time. The relationships (between and within the hierarchy of complexity) are conceptualised by formulae.

The domain concepts from the D&L conception of the general design problem are considered sufficiently comprehensive to permit operationalisation. A diagram of the domain is a suitable representation of the objects and attributes and their relationships. Figure 9 shows the key for domain diagrams.

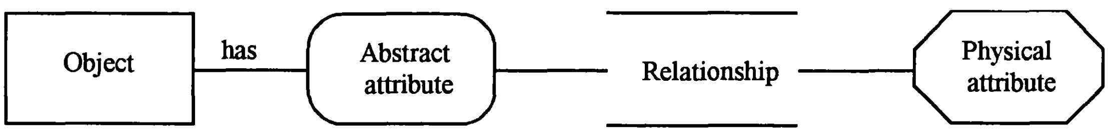


Figure 9. Domain Diagram Key.

The task goals, product goals, and task quality Boolean values will be documented as attributes of the objects, together with their relationship with the other attributes. The relationships are intended to be mathematical, and to include the Boolean logical operations.

The states of the attributes for each instance of the current or actual design will be recorded in a 'state stream table'. Figure 10 shows the headings to be used for the state stream table, with sample entries.

Time	**Event**	**Attrib-ute 1**	**Attrib-ute 2**	**Attrib-ute 3**	**Attrib-ute 4**	**...**	**Task goal 1**	**Task goal 2**	**Product goal 1**	**Task quality**
0:00	1	3.4	10°	TRUE	-62	...	TRUE	FALSE	FALSE	FALSE
...	2									

Figure 10. Domain State Stream Table Key.

The time column shows a suitable progression of time during the design problem or solution. The time interval to select is dependent on the rate of change of the domain and worksystem. The event column shows the ordering of domain changes in the domain, including where those changes occur within the same time frame. It is required for the framework for worksystem costs, presented below. The first row shows the initial states. The state stream table has the advantage of supporting better specification of the relationships between the attributes during the design problem or solution. Further, it can be used to identify a formula for the state over time. If such a formula can be identified, then it can be employed to calculate the state over time.

Boundary of Meta-Assumption

Operationalising the task quality of a worksystem[5] could be an attempt to operationalise a broad purpose, such as man's existence. However, for these operationalisations, only 'local' purposes, such as 'comfort', will be operationalised.

Framework for Interactive Worksystem Costs

The diagram for the worksystem will show the process structures (which support the behaviours) using a MUSE-like[6] notation and the representational structures using the domain key. The potential behaviours supported by the process structures that

---

[5] The goals of users and clients.

[6] The notation is a subset of the MUSE notation, except for the addition of concurrency shown by a • above the sequence, etc. construct.

change the states of the domain will be linked with a line to the domain state that can be changed in the domain diagram.

The worksystem will effect state changes over time. It will often be difficult to identify the time between behaviours, for example, it will be difficult to identify the time between seeing an object and its categorisation. The concept of 'events' is introduced to operationalise the ordering of behaviours, without distinguishing the time. A new event occurs for every behaviour, except when the behaviours are concurrent. The time is recorded against each event.

Once the initial description has been produced, the structures, the behaviours that occur, and their costs will be placed as the headings in a table to match that of the domain table above. This 'structure and behaviour streams' and costs table is shown in Figure 11 with sample entries.

		Abstract beh.s		**Physical beh.s**		**Costs**		**Abstract structures**		**etc.**
Time	**Event**	**Beh. 1**	**Beh. 2**	**Beh. 1**	**Beh. 2**	**Cost 1**	**Cost 2**	**Struct. 1**	**Struct. 2**	...
		Cost contrib. for Beh. 1	Cost contrib. for Beh. 2	Cost contrib. for Beh. 1	Cost contrib. for Beh. 2			Cost contrib. for Struct. 1	Cost contrib. for Struct. 2	...
0:00	1	FALSE	TRUE	FALSE	TRUE	1	3	0	TRUE	...
...	...	...	...	...	...	...	...	...	...	...

Figure 11. Structure and Behaviour Streams and Costs Table Key.

For the structures, the change in the state will be marked against time and event. For the behaviours, the occurrence of the behaviour will be marked against time and event. The cost 'contribution' of the structures and behaviours will be shown in the first rows of the table. This cost contribution is the abstract and physical costs of the structure state change or behaviour occurrence (and development), separated into abstract and physical. The behaviour occurrences, structure state changes, and domain state changes can be related by formulae. The costs columns can then be calculated by formulae.

The time interval to select is now dependent on the rate of change of the worksystem as well as the domain.

Formulae

The primitives for the formulae are in Appendix H.

Composite Structures

Composite structures are conceptualised as groups of processes that occur repeatedly in the same operationalisation or across operationalisations. They may be used as process structures, in place of the repeated process structures. Composite structures can be given parameters. Appendix E gives the composite structures used in the operationalisations for this research.

Composite structures reduce the size of the diagrams and tables for the operationalisations, so improving their readability and development. They represent low-level structural generality within and between the operationalisations. Figure 12 describes the composite structures that were developed during the operationalisations.

The planning and control composite structures (H:StMon, H:StSubPlan, etc.) refer to planning representations (CDc, CDd, CWd, etc.) that are conceptualised in the next chapter.

Composite Structure	Description
H:FP:X	Human forms goal, other behaviours occur, then goal is popped.
H:FS:X	Human forms goal, other behaviours occur, then goal is suspended.
H:RS:X	Human resumes goal, other behaviours occur, then goal is suspended.
H:RP:X	Human resumes goal, other behaviours occur, then goal is popped.
H:FxP:X	Human forms a goal to encode or execute X, encodes or executes X, then pops the goal.
C:IISO:X	Computer inputs X.
C:O:X	Computer outputs X.
H+C:Change gas:X, Change	Human and computer (cooker) change the gas of X (a ring or the oven) by Change amount.
H:StMon:X,Y	Human collects information through sight, updates the CDc planning representation, and decides whether to change the plan.
H:StSubPlan:X,Y	Human updates the CDd planning representation, and then updates the CWd planning representation.
H:StMonA:W,X,Y,Z	Human collects information, updates the CDc planning representation, and decides whether to change the plan.
H:StMonB	Human updates the CWc planning representation.

Composite Structure	**Description**
H:StSubPlanA:W,X,Y	Human updates the CDd planning representation.
H:StSubPlanB	Human updates the CWd planning representation.
H:ShSubPlan:W	Human updates the CWd planning representation by writing.
H:StShSubPlan:W,X,Y	Human updates the CWd planning representation by either writing or mental storage.

Figure 12. Composite Structures.

This chapter has presented several frameworks to support operationalising design problems and solutions. In addition, the concept of composite structures was introduced to improve the readability and development of the operationalisations.

7. Conception of Planning and Control

Chapter 4 presented the rationale for scoping this research to 'planning and control' engineering principles. This chapter presents an initial conception of planning and control that has been developed for this research. This conception supports the operationalisation of planning and control for the specific design problems and their solutions.

Conceptions of planning and control are frequently offered in the HCI, Psychology, and Artificial Intelligence (AI) literature. The conception developed here suits the purposes of this research by giving priority to conceptions for which there are claims for design guidance. Priority will be given to those with stronger claims, which are typically those in the HCI literature.

This chapter starts with planning and control literature with claims for design guidance, and continues with planning and control literature with no claims for design guidance. An initial conception of planning and control is then presented, based on the literature.

Conceptions of Planning and Control with Claims for Design Guidance

The research that contains claims for design guidance can be divided into that which identifies plans as being in the domain and that which identifies plans as being representations in the worksystem. Planning is identified as occurring in the worksystem. Control is identified as either occurring in a different worksystem from the planning worksystem or occurring in the same worksystem that has performed the planning.

Plans in the Domain

Colbert (1994a, 1994b; Colbert and Long, 1996; and Colbert et al., 1995) proposes a design for 'a menu structure for planning systems'. He devised rules to systematically relate the menus to the planning. The general menu structure was instantiated using the rules for two types of planning: the planning of men and equipment off-loading during amphibious operations; and the planning of attacks with surface-to-surface guided weapons. The instantiated menus were evaluated, leading to revisions of the rules and the general menu design. The final version of the menu structure was, therefore, explicit, with rules, and has a strong claim to be design guidance.

Colbert also identified worksystems that produced plans by planning. He described a 'Domain of Plans' that enabled a description of the quality of plans. The military plans that he considered were conceived as 'a representation of desired future states of conflict objects (friends, enemies, etc.), and/or behaviours of a system that controls military operations'. Control, therefore, is the execution of the behaviours defined by the plan and the achievement of the states as defined by a plan.

Plans in the Interactive Worksystem

Dowell (1993) developed design guidance based on the description of a planning and control worksystem that manages air traffic. The description of the worksystem identifies the cognitive representations and processes that support planning and control behaviours. The representations are complex, and include the current and future state of the domain of air traffic management. The processes, which are relatively simple in comparison with the representations, are developed from AI planning.

Dowell separated planning processes from control processes in that 'planning specifies plans for the air traffic, controlling executes those plans'. Dowell considered that the cognitive representations he described were such plans. The plans contain states of the aircraft passing through the sector—the current, projected, planned, and goal states—and planned interventions for the aircraft. The state of the aircraft is a representation of the attributes identified for the domain. Dowell identified the attributes for the air traffic objects in the air traffic management domain as the position, altitude, speed, and heading of the aircraft at a particular time (PASHT). The current state of the aircraft at any particular time is its current PASHT value. The projected state of the aircraft at any particular time is its PASHT value on leaving the sector, if no interventions were made. The planned states of the aircraft are the expected PASHT values given the planned interventions. The goal state of the aircraft is a desired PASHT value on leaving the sector. The interventions are representations of the processes that the worksystem intends to execute to achieve the planned states.

Dowell claimed that air traffic management is a dynamic domain. A dynamic domain has 'intrinsically dynamic processes [which] change state over time even without intervention'. A dynamic domain reduces the length of time available for planning. Dowell adopted AI responses to dynamic domains to characterise the worksystem as 'a reactive planner; interleaving planning and control; ... a hierarchical planner; ... and a non-linear planner'. Reactive planners 'build or change their plans in response to shifting situations at execution time' (Firby, 1987; cited in Dowell, 1993).

Reactive planners are a particular type of interleaved planner, which plan concurrently with controlling. Hierarchical planners plan at a higher level than basic cognitive processes (for example, 'change aircraft height to 790' rather than 'grasp mouse', 'move hand', etc.). Non-linear planners do not necessarily represent the planned processes linearly, the order and the time that they are to be executed may not be directly represented.

At an abstract level, Dowell followed Linney (1991) in identifying the interleaved planning and control processes. These are represented in Figure 13. Dowell identified processes for the worksystem that are less abstract, and relates those processes to the representations. Figure 14 shows these processes and their related representation changes.

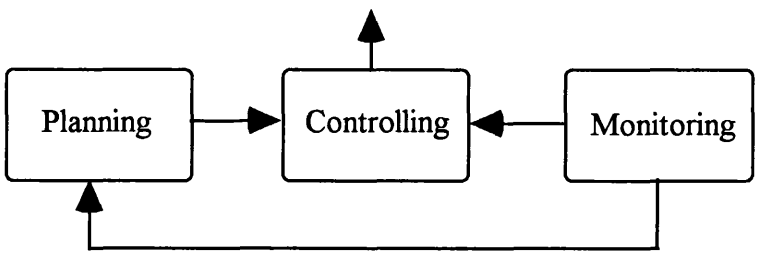


Figure 13. Linney's Abstract Description of the Planning and Control Behaviours of an Interleaved Planner.

Process	Representation
	Monitoring behaviours
generate	current airtraffic event (PASHT attribute values)
generate	current vector (actual and projected task attribute values)
generate	goal vector (goal task attribute values)
evaluate	current vector
	Planning behaviours
generate	planned vector (planned task attribute values)
evaluate	planned vector
generate	planned interventions (PASHT attribute values)
	Controlling behaviour
generate	execution of planned intervention (issue instruction)

Figure 14. Description of Dowell's Planning and Control Processes of an Air Traffic Management Worksystem.

The planning and control design guidance generated by Dowell is only putative and specific to the design of a software flight strip. Its relationship with the planning and control representations and processes is not explicit. It was developed by identifying positive and negative current performance statements (in comparison with some unstated desired statements) and attributing these statements to the current planning processes and plan representations. The following list of design guidance is the result (Dowell, 1993):

- Currently, the planned vector of a plane is not evaluated exhaustively with respect to safety. Improving the ability to evaluate the planned vector of a plane with respect to safety would improve the performance. Therefore, the designer should highlight 'those aircraft with proximal projected vectors' or train the controller in conflict search procedures.

- Currently, the rate of potential plan moves is slow. Improving the rate would improve performance. Therefore, the designer should ensure that the mental representation of the planned vector and the paper representation is closer.

- Currently, the construction of current and goal vectors is adequate. Performance would be reduced if this construction changed. Therefore, the designer should ensure that the aircraft's flight information is differentiable from those of other aircraft by being visually displayed in an arrangement based on the current spatial arrangement of aircraft on the sector, as it is with the current paper flight strips.

- Currently, the evaluation of planned vectors with respect to safety is adequate. Performance would be reduced if this evaluation changed. Therefore, the designer should ensure that the flight information for proximal aircraft within the controller's planning horizon should be displayed together, as it is with the current paper flight strips.

Hill et al. (1995) similarly develop design guidance, but based on planning and control of multiple-task work rather than air traffic management. They concentrate on the cognitive processes of the user that support planning and control. The processes are more complex than Dowell's (described above), but are also developed from AI planning. Like Dowell, they suggest that multiple-task work is a 'dynamic task environment'. They suggest that planning and control in multiple-task work is interleaved, hierarchical, and non-linear (to use Dowell's terms). Hierarchical, in this case, must be extended to include the 'states of the environment' in addition to the 'behaviours'. This extension is analogous to Dowell's plans, which need not necessarily contain interventions, and leads to an emphasis on 'execution' being 'constrained by, rather than specified by, the plan'. They define 'planning ... as specifying the tasks and/or behaviours necessary to carry them [the tasks] out, and control … as selecting behaviours to be carried out'. Control in this sense includes the behaviours of planning and control. They also define 'perception behaviours … as those whereby the system learns about the tasks, and execution behaviours as those which directly effect the task'. Execution behaviours appear to be already included in the definition of control behaviours above. However, by explicitly defining them, it must be assumed that the control behaviours do not include execution behaviours.

The planning and control process (general cognitive control behaviours) is identified by Hill et al. as: 'The cognitive behaviours of perception, planning and execution are carried out in a single fixed sequence of: perception then planning then execution'. Representations are not elaborated further, except in the specific analyses, for example, a 'to do' list is a plan representation in an analysis of secretarial multiple-task work.

The planning and control design guidance generated by Hill et al. (1995) is only putative but is expressed in general terms, probably because they analysed three systems:

- They state that 'for any system there is a potential trade-off between the complexity of planning behaviours and the complexity of control behaviours'.
- They suggest that 'sharing behaviour', the progression of more than one task simultaneously, is 'expeditious', i.e. enhances performance; 'sharing behaviour suggests that planning behaviour is able to take account of some low-level similarities between requirements for different tasks'.
- They identify that planners perform 'opportunistic task switching'.
- They identify that planners perform 'forward information acquisition', the gathering of information that might be relevant to future tasks.
- They claim that 'if plan checking is too infrequent, relative to domain stability, it will not support the maintenance and use of suitable plans. If on the other hand, plan reading is too frequent, relative to domain stability, it will incur unnecessary resource costs'.
- They claim that 'failure to be prepared for opportunities will lead to a reduction in task quality while effort put into preparation for opportunities which never happen will only generate greater resource costs'; they claim that the optimal state of this relationship is dependent on the stability of the domain.

Robertson and Black (1986) take an alternative direction in their analysis of text editing. They state that 'when people learn such a complex skill as computer text editing, they are learning a set of goals and the plans for accomplishing those goals'. A plan is understood as 'a memory structure that indexes subgoals or actions that will achieve a specific overall goal'. They also state that 'for well-learned behaviours, an active goal indexes several potential plans'. They claim that 'people acquire plans by repeated problem solving' and that,

> 'Initially, successful plans are maintained in a declarative representation that allows easy verbal access for use in problem solving and plan restructuring. Frequently used sequences of goals and actions are eventually compiled into a procedural representation.'

From the above statements, the procedure for planning and control is: for experts, one of simply 'indexing' an appropriate 'plan' for a particular goal; for non-experts, developing an appropriate plan for a particular goal. The process of development of an appropriate plan by non-experts is not explicit. However, they state generally that people 'do not typically plan an entire task at the outset' and 'they seem to plan very short sequences of actions at a time'.

The design guidance they offer would be difficult to operationalise given their conceptions. They state their design guidance generally as:

- Long inter-keystroke times were found to be associated with plan boundaries. The longest inter-keystroke times were found between keystrokes separating super-ordinate goals, whereas less significant time increases appeared between keystrokes at subgoal boundaries.

- Experience resulted in a reduction in inter-keystroke times, because plan restructuring occurred.

If planning and control is understood to be a type of problem solving, then Mayhew (1992) supports some of Robertson and Black's claims. She states that:

> 'People make decisions regarding the relative importance of a problem and the relative expense of different strategies and often choose sub-optimal strategies for lower-priority problems.'

> 'There is a natural tendency to learn better strategies with practice.'

However, she does not give a clear indication of the concepts of planning and control. She offers the following design guidance:

- An interactive system that blindly forces a user to execute repeatedly algorithmic procedures will quickly cause boredom and frustration.
- Systems should be flexible and allow shortcuts for experienced users.
- A robust system with good 'help' capabilities will encourage users to experiment.
- An active help system may be useful.
- A good interactive system should not require more effort to learn than is merited by the problem to be solved.

Shneiderman (1992) offers planning and control design guidance, but no explicit understanding of planning and control. Shneiderman states:

- For a given user and task, there is a preferred computer response time. Long response times lead to wasted effort and more errors when a solution plan is reviewed continually. Shorter response times may generate a faster pace in which solution plans are prepared hastily and incompletely.

The above design guidance is supported by several quoted experiments. It suggests that planning and control is a worksystem issue rather than only a user issue, and that worksystem planning and control is interleaved.

Conceptions of Planning and Control with no Claims for Design Guidance

The examination of conceptions for which there is no explicit claim for design guidance, particularly those from Psychology and AI, is included in this chapter since:

- These conceptions often implicitly underlie the design guidance research, particularly when the design guidance research is weak on exposing the conceptions involved.
- The operationalisation of planning and control may require concepts that are not included in the existing design guidance work.

HCI

Norman (1989) states that 'for many everyday tasks, goals and intentions are not well specified: they are opportunistic rather than planned'. Planning, therefore, includes specifying goals and intentions, where intentions appear to lead to action. A plan includes a specification of goals and intentions. He contrasts planning with opportunism, where goals and intentions are not well specified. A plan, therefore, must include well-specified goals and intentions.

Town Planning

Friend and Jessop (1969) identify planning as being 'required for non-trivial action decisions, i.e. prior elaboration of potential actions is required for them to be assessed'. They also note that 'it is [in public planning] exceptionally difficult to formulate strategies in advance which are sufficient to cope with all conceivable contingencies … in these circumstances, planning must become in some degree an adaptive process'.

Psychology and AI

The Hayes-Roths' (1988; and Engelmore et al. (1988) model of planning is intended to be 'computationally feasible and psychologically reasonable'. They define planning as 'the process by which a person or a computer program formulates an intended course of action'. They emphasise that planners may make decisions about the contents of the plan in very varied ways. They may make abstract decisions about the 'gross features of the plan' to guide decisions about the details, or vice-versa.

The Hayes-Roths' planning model contains independent and asynchronous 'specialists'. These specialists propose decisions to be incorporated in a tentative plan. The plan is maintained on a 'blackboard' through which the specialists communicate. The plan indicates the 'actions the planner actually intends to take in the world'. The plan stops being tentative when the planner 'accepts' the overall plan, after 'plan evaluation, the analysis of the likely consequences of hypothesized actions'. Presumably, acceptance occurs when the plan evaluation passes some threshold. A 'meta-plan' orders the execution of: the specialists; a process of 'situation assessment, analysis of the current state of affairs'; and the process of plan evaluation.

Barr et al. (1989) supply a pithy definition: 'a plan is a representation of a course of action'.

AI

Alterman (1988) describes adaptive planning instantiated in a system called PLEXUS: 'the problem of adaptive planning ... is to take a prestored plan ... and apply it to a novel set of circumstances'.

Initial Conception of Planning and Control

Rationale

The conceptions outlined above vary in their explicitness, completeness, coherence, their operationalisation in the design guidance, and in their claims for the design guidance.

Dowell and Colbert's offer the design guidance with the strongest claim, since it is very specific. Hill et al. offer the next best design guidance, since it is explicitly developed from analysis of particular systems. All the other design guidance cited, where no underlying conception of planning and control, have only weak claims.

The explicit conceptions that claim design guidance have their origins in Psychological and AI conceptions, which offers additional value to those conceptions. The general extent of operationalisation of the conceptions above, however, is low. Colbert, Dowell, and Hill et al. probably operationalised their conceptions more than the others.

The conception developed for this research aims to:

- Be inclusive over the above conceptions of planning and control to ensure the widest potential to develop planning and control engineering principles. For example, from all the conceptions evaluated, AI probably offers the best potential for operationalising computing planning and control.

- Decide between alternatives by selecting those in which the design guidance is better operationalised and has stronger claims.

- Relate the planning and control conception to the conception of the general design problem.

Initial Conception

As identified above, descriptions of planning and control are divided into those which consider planning and control to be separate worksystems (e.g. Colbert) and those which do not (Dowell, Hill et al., etc.). The distinction should be considered in terms of the scope of the system to be designed, the knowledge to be acquired, or both. However, this distinction appears to have an additional relationship with the 'planning horizon', the length of time available before control must be performed, and, perhaps, therefore, with the design guidance. The conception offered here attempts to relate the two aspects.

Colbert's conception is an example of the separation of planning and control into separate worksystems, and he presents the relationship as in Figure 15.

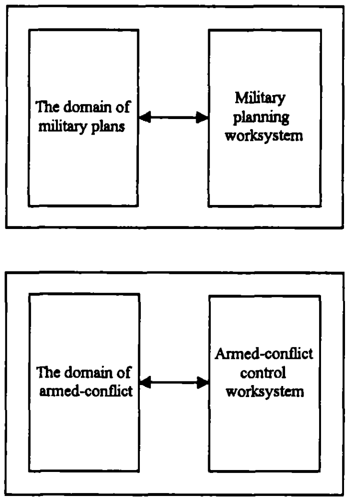


Figure 15. Colbert's Representation of the Domains and Interactive Worksystems of Planning and Control.

Colbert fails to identify explicitly the relationship between the two worksystems. There are several not necessarily mutually exclusive alternatives (shown in Figure 16), assuming that the planning system is being designed. Figure 16a shows one alternative in which the plans specify the desired states of the control domain. Figure 16b shows another alternative in which the plans specify the behaviours of the control worksystem. Figure 16c shows a third alternative in which the plans specify the (perhaps initial) contents of representations: of the desired states of the control domain; and of the planned behaviours of the control worksystem. The first of these alternatives is an analysis that would probably need to be performed during design of the control worksystem. The second is part of the specification to be produced during design of the control worksystem. The third could similarly be part of the specification to be produced during design of the control system or could represent a logical separation of planning and control for the application (and potentially therefore, the acquisition) of design knowledge. Colbert's work does not suggest the latter; the former can be understood as the same as Figure 16a, if the plan representation content is understood to be abstract structures which is the approach adopted here. Colbert does not state which of the alternatives he intends, so Figure 17 contains Colbert's diagram representing these three alternatives. Figure 18 shows a generalisation of Figure 17; this generalisation will be referred to for this research here as 'desired states and structures planning and control' (DSSP&C).

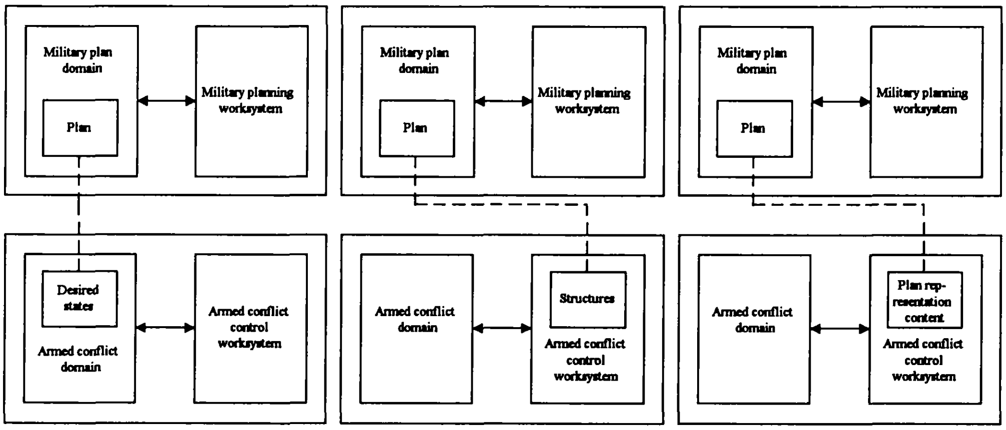


Figure 16 a, b, and c. Alternative Representations of Colbert's Planning and Control.

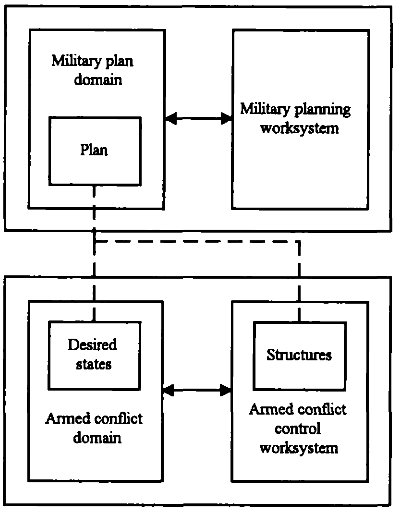


Figure 17. Composite Representation of Colbert's Planning and Control.

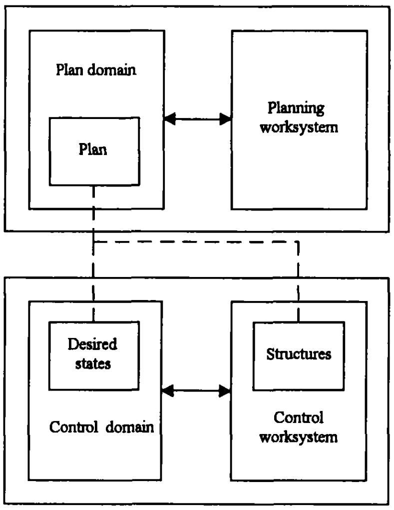


Figure 18. Desired States and Structures Planning and Control (DSSP&C).

Hill et al. and Dowell take a different approach, see Figure 19, (taken from Hill et al. although excluding some detail), and Figure 20 (inferred from Dowell). Their approach is compatible and Figure 21 shows a general version recognising the overall target of planning and control as the control work. This combined version can be represented, albeit in a more decomposed manner, by DSSP&C, see Figure 22.

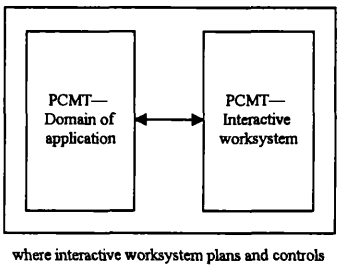


Figure 19. Hill et al. Planning and Control.

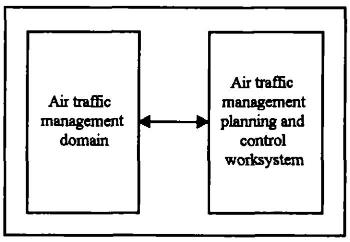


Figure 20. Dowell Planning and Control.

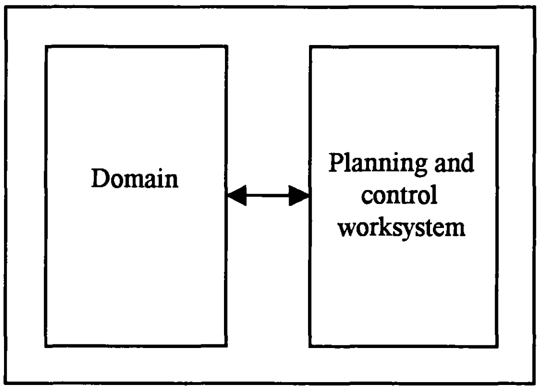


Figure 21. Composite Planning and Control for Hill et al. and Dowell.

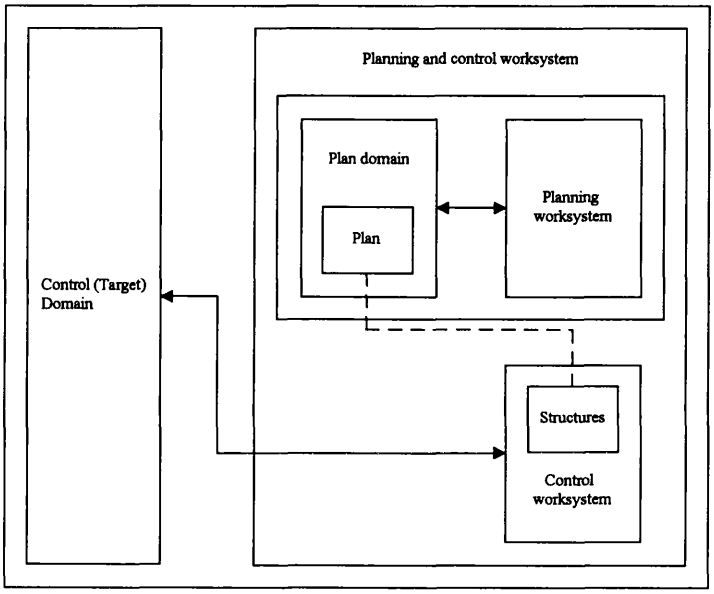


Figure 22. Hill et al. and Dowell Planning and Control Represented in terms of DSSP&C.

Therefore, DSSP&C (Figure 18) is taken as the basis for the planning and control conception for this research. The target concept is generalisable, as demonstrated by Stork et al. (1998) who apply it to training and emergency management.

Control Domain

The control domain is conceptualised for this research in the same manner as the domain in the specific design problem and solution conceptions. Therefore, the control domain contains desired states, in the case of a desired performance operationalisation, and actual states, in the case of an actual performance operationalisation.

Control Worksystem

The control worksystem is conceptualised in the same manner as the worksystem in the specific design problem and solution conceptions. Therefore, the control worksystem contains structures.

Plan domain

Colbert identifies plan and sub-plan objects in the domain of plans for armed-conflict. Plan objects are 'a representation of the goal states of [control][7] domain objects and/or desired future behaviours of a control worksystem'. Sub-plan objects are 'a specification of lower level goal states of [control] domain objects and/or desired future behaviours of a control worksystem'.

He states that plans and sub-plans have attributes of scope, view, and content types. The scope types are: time_scope, 'the period of time to which *content* applies'; object_scope, 'the [control] domain objects to which *content* applies'; and behaviour_scope, the 'control worksystem behaviours to which *content* applies'. The view types are: view_type, 'the type of representation'; view_content_options, 'selections of *content* to be expressed in a representation'; and view_format_options, 'variations in the physical representation of *content*'. The content types are: content, 'the specification of goal states of ... [control] objects and/or the behaviour of ... control worksystems'. The states of the attributes support the representations of the plan and sub-plan objects.

Colbert's scheme will be adopted for this research. The content is redefined as 'the specification of desired states of control objects and/or the behaviours of control worksystems'.

Planning Worksystem

The primary representation is that of the plan requiring a representation of: the potential control behaviours and their effects on the desired control states, and the current and desired control domain and worksystem. Colbert's menus, Dowell's list of representations, Norman's plan, and the Hayes-Roth blackboard model all support the identification of these representations. Figure 23 shows the potential behaviours on these representations.

---

[7] [control] has been added to relate Colbert's planning and control with the DSSP&C.

Process	Representation
	Monitoring
(re-)generate	Current state of control domain (CDc).
(re-)generate	Current state of control worksystem (CWc)
(re-)generate	Potential control structures and their effect on the desired states (CPSEc)
	Planning
(re-)generate	Desired control domain (CDd)
evaluate	desired worksystem domain structures (CWd)

Figure 23. General Planning Behaviours.

The overall ordering is one of monitor→plan→monitor→etc. The behaviours are of the worksystem, humans *and* computers; consistent with Shneiderman's planning and control design guidance above. Adaptive planning (Dowell; Friend and Jessop) is supported by the re-generation of plans.

Following Robertson and Black, both expert and non-expert planning and control behaviours are conceptualised. Non-expert planning and control would require more of the behaviours than expert planning and control. Users of domestic energy management, the focus of this research, could be experts or non-experts at planning and control.

Planning is similar to design, and the representations are in the terms of design. In current HCI terms, the representations might be understood as 'the worksystem's view of the domain and worksystem', rather than the designers'.

Operationalisation

It is proposed that the planning domain and worksystem are operationalised separately from, but related to, the control domain and worksystem. The representation structures will be operationalised in the planning domain and worksystem. The process behaviours will be operationalised by reference to composite planning and control structures.

Concepts such as 'learning', 'pre-planning', 'reflectiveness of planning', and 'meta-planning' are not expected to be operationalised in the cycle designs.

This chapter has presented an initial conception of planning and control for this research. This initial conception is to be used in the operationalisation of the planning and control of the cycle designs. These operationalisations may lead to alteration of this initial concept of planning and control, since it is based on current conceptions of planning and control. Current conceptions are either strong on the design guidance and weak on the conception, or weak on the design guidance and strong on the conception.

8. Cycle 1 Best-Practice Development

Chapter 4 proposed the development of an artefact specification to solve the Cycle 1 user requirements as part of the strategy for acquiring engineering principles. The Cycle 1 user requirements were selected in Chapter 4. This chapter describes in brief the best-practice development for this research of an artefact specification to solve the Cycle 1 user requirements. The artefact specification, the product of the best-practice development, is described before the best-practice development to aid comprehension.

Best-practice development was taken to include the application of MUSE to the user requirements. The full products for the MUSE development for this research appear in Appendix F.

Finally, the chapter describes informal evaluations from this research of the artefact specification against the user requirements. The evaluation was positive, i.e. informally, the artefact specification indeed fulfils the user requirements. This positive evaluation means that this Cycle 1 best-practice development supports the operationalisation of the Cycle 1 specific design problem and its solution, which is presented in the following chapter.

User Requirements

The user requirements for Cycle 1 are restated below:

> 'If *A* leaves after 8 a.m. or stays at home to work, then the house is too cold until he turns the gas-powered central heating back on. If he expects to be at home for a short time, then he often uses the one-hour boost facility on the heating controller to turn the heating back on. However, if he is then at home for more than an hour, he can become cold. *A*'s ability to work is adversely affected by being cold and having to control the heating. The nature of his work means that it is difficult for *A* to plan much in advance whether he will be at home, and if so, for how long. The current gas bill is acceptable and an increase could be tolerated, although a decrease would be desirable.'

Artefact Specification

The artefact specification was developed from the best-practice described in the following sections. It can best be characterised by comparison with the existing heating controller. The existing controller in the home was limited to a pre-set schedule for each day. It was programmed to have two heating 'on-off' periods:

- 'On' early morning at 6:40 a.m. and 'off' at 7:20 a.m.
- 'On' early evening at 6:30 p.m. and 'off' at 10:00 p.m.

This existing controller does not have enough features to meet the user requirements. The best-practice development for this research replaced the existing controller by one that has the facility to:

- Switch on in the morning at 6:40 a.m. and switch off at 10:00 p.m. during the week.
- Turn the heating on again at 6:30 p.m., if turned off during the day.
- Turn the heating on and off as before for the weekends.
- Have an additional remote heating-controller, with an advance button and a bright status light, by the front door.

The occupants of the home will be instructed to use the heating controls as before, except that *A* should press the advance button on either controller if the status light is 'on' just before leaving to go to work during the week. *A* is to be considered the user of the designed artefact.

Best Practice Development

As determined by the research strategy, the MUSE method and HCI guidelines were applied to the user requirements. The resulting artefact specification is described in the previous section. The MUSE products for the development are in Appendix F.

Information Elicitation and Analysis Phase

The current extant system was analysed in detail. Other extant systems were listed but not analysed, since a satisfactory artefact specification was delivered by the first MUSE iteration. Two Task Description MUSE products were produced:

- A task analysis was conducted based on an interview in which *A* introspected about his days (Task Description 1.1).
- *A* was asked to keep a diary for several mornings during which he stayed at home and left for work.

These Task Descriptions were generalised (Generalised Task Model of the extant system MUSE product) to gain an understanding of 'generic' mornings (which the design needs to support). The tables for the products for the extant system detailed valuable observations, design implications, and speculations that arose during this phase. For example, it was observed that *A* appears to plan using an electronic diary and to-do list. The possibility of interfacing this electronic diary and to-do list with the heating control was considered, but dismissed because there was poor correspondence between the departure plan and the electronic diary and to-do list.

The final step of the phase was to develop a task-level conceptual design of the target system (General Task Model of the target system MUSE product) based on: the user requirements; and the design implications and speculations produced by the analysis of the extant system. The task-level conceptual design documented the essential design decision to control the heating on departure.

The initial task-level conceptual design suggested a potential for re-use of more detailed extant system features and it was decided to perform a more detailed analysis of the extant system to support that potential. Accordingly, a range of MUSE products were developed that analysed the extant system from its conceptual to its detailed design, e.g. the Domain of Design Discourse of the extant system; and the System Task Model of the extant system.

Analysis during the Information Elicitation and Analysis phase was the basis of the design in the other phases, so reducing the time spent on these other phases.

Design Synthesis Phase

A textual summary of the human factors concerns was constructed (Statement of User Needs MUSE product) based on the user requirements and the analysis of the extant system. The statement contained:

- Explicit design criteria, such as the need for the artefact cost to be acceptable for the benefits.

- Implicit design criteria, such as the retention of the existing functionality of the controller to support non-weekday-morning tasks.

- Explicit system performance criteria, such as *A* must not be cold.

- Implicit performance criteria, such as *A* must be permitted to leave home when he desires (constraining should not be considered suitable for the artefact specification).

- Relevant human factors knowledge, such as an extension of a guideline by Shneiderman (1992) that 'human action should be eliminated where no [human] judgement is required' to include 'and minimise human action where human judgement is required'. This extended guideline confirmed the essential task-level decision expressed above.

The conceptual design of the conjoint user and computer tasks was advanced (Composite Task Model MUSE product), maintaining consistency with the accepted foundation of the task-level design developed in the previous phase. Important design decisions were rationalised at this stage: the provision of a controller in the same location as the existing one; and the further provision of a controller near the front door.

The design was considered at a lower level of detail by the decomposition of the on-line tasks (System Task Model MUSE product). At this stage, the human factors guidelines of 'transfer of learning', 'feedback', and 'consistency' (Smith and Mosier, 1986) were applied. For example, transfer of learning was supported by porting effective extant tasks to the target system.

Allocation of function between the user and the artefact was considered. It was considered difficult, if not impossible, to allocate the user's leaving plan to the controller, so it was decided that the controller should simply respond to the user's control commands. This allocation corresponds with the human factors guideline that humans are generally better than computers at 'drawing on experience and adapting decisions to situations' (Shneiderman, 1992).

The additional remote heating-controller was justified as reminding *A* to control the heating on leaving.

Design Specification Phase

The interaction-level design was advanced (Interaction Task Model and Interface Model MUSE products). The remote heating-controller was designed with an advance push button to ensure 'consistency' between the two controllers. Substantial porting of the extant design was possible, particularly with the layout of the two heating-controllers (Pictorial Screen Layouts MUSE product).

Evaluation

Three informal analytic assessments of whether the artefact fulfils the user requirements were conducted, apart from the assessment of consistency through the application of MUSE. Firstly, an analytic argument was constructed to show that the introduction of the artefact into the home of *A* and *S* should 'satisfy' the problem. A form of this analytic argument, commensurate with the user requirements, follows:

> 'The proposed artefact should support the work patterns exhibited by *A*, which occasionally requires him to remain at home to work in the mornings, rather than leave earlier with his partner, *S*, to work at his office. If *A* leaves after 8 a.m. in the morning, or stays at home to work, then the house should remain warm without intervention. The design ensured that the gas-powered central heating would remain on rather than turning itself off, which caused *A* to be uncomfortable because the house cooled. Since the heating stays on until adjusted on exit, *A* is not required to adjust the system manually. Therefore, even if *A* expects to be at home for a short time after 8 a.m., he should not need to use the one-hour boost facility.
>
> *A*'s ability to work should no longer be adversely affected by him being cold and having to control the heating, since the house is now warm and the heating does not need controlling until he has finished working.
>
> *A* finds it difficult to plan in advance, whether he is staying at work and, if he stays, how long he will stay to work. The artefact should address this planning difficulty, as the heating should only need controlling to match the time of planning.
>
> The gas bill may increase by a small amount, which *A* and *S* consider acceptable. The cost to *A* in remembering to turn the heating off on

> exit should be low. The cost of the artefact should be low (approximately £40 for a fully functioning prototype version).'

The second informal analytic assessment involved a panel of nine practitioners, five human factors engineers and four software engineers, appraising the artefact specification produced using MUSE. They were all familiar with the method and the user requirements. Although some initial objections were raised, after discussion none of these were considered relevant in terms of the artefact satisfying the user requirements. Some of the objections asserted that the artefact fulfilled more than the user requirements (but not less), while others that the artefact might have embodied alternative design features.

The third, and last, informal analytic assessment was an expert walkthrough of the artefact specification performed by a human factors engineer (other than the researcher). His report contained the following concluding statement:

> 'The likely behaviour of the occupants of the house with respect to the system was estimated with respect to a number of scenarios concerning different types of morning events. It was considered that in the scenario where there was previously a problem (i.e. when *A* remained at home after 8 o'clock), the system would solve the problem by maintaining *A*'s comfort, and that *A* would remember to switch the system off as long as the front door controller was located in a suitably prominent position. In a situation where *A* left the house early, his expectations of the system based on the existing system may initially cause him to forget to switch the heating off, as he is currently not required to take any action if he leaves early in the morning. However, it is to be expected that *A* would soon learn to adapt his morning routine to include the new task of switching the heating off. Similarly, if *A* left the house earlier than *S* at any time, *S* might forget to switch the heating off, as the normal morning routine does not require any action on *S*'s part. However, if the indication of the system status was designed to be sufficiently conspicuous, and the controller was prominently located, these problems would be less likely to occur than if the controller was located in a less visible position. At present, there is no evidence in the user requirements or in the analysis of the existing systems that *A* will ever leave earlier than *S*; further consultation with *A* has revealed that it is very seldom the case that *A* leaves first, and so the problem of *S* having to

remember to operate the system would occur very (and acceptably) infrequently.'

In addition, an empirical assessment has been performed by constructing from this research an interactively faithful prototype (which does not alter the state of the heating) of the remote heating controller and re-programming the existing controller for weekdays. This prototype was placed by the front door in the home (Figure 24) and the occupants given instruction to its use. This assessment confirms the analytic argument, except that an empirical assessment of the gas bill increase was not completed.

Taken together, the analytic and empirical assessments demonstrate, albeit informally, that the artefact specification indeed fulfils the user requirements. Further evaluation (e.g. Karat, 1988) was not considered necessary.

The artefact specification developed for this research and described in this chapter informally fulfils the Cycle 1 user requirements. Therefore, this Cycle 1 best-practice development supports the operationalisation of the Cycle 1 specific design problem and its solution, which is presented in the following chapter.

Figure 24. Front Door Controller Prototype.

9. Cycle 1 Operationalisation

Chapter 4 proposed the operationalisation of the Cycle 1 specific design problem and its solution as part of the strategy for acquiring engineering principles. The Cycle 1 best-practice development for this research was presented in the previous chapter. This chapter describes in brief the operationalisation for this research of the Cycle 1 specific design problem and its solution. The current solution operationalisation is described before the specific design problem and its solution operationalisation.

The operationalisations are of the specific design problem and solution conceptions (Chapter 2), the conception of human-computer systems (Chapter 5), and the conception of planning and control (Chapter 7). The operationalisation was developed from an explicit operationalisation (similar to the brief description in this chapter) to support the formal and metricated operationalisation. The formal and metricated operationalisation, using the frameworks (Chapter 6), is in Appendix G.

Following the strategy, the **conative and affective abstract behaviours and structures** are not operationalised here.

Current Solution Operationalisation

Specific Actual Performance

The planning specific **actual performance** is operationalised as the union of the planning specific actual quality and the planning specific actual costs. The planning **worksystem boundary criteria** are operationalised by the requirement that the constituents of the planning **worksystem** have the common goals of the current (level of) achievement and satisfaction of the planning of the comfort of A and the leaving of A. The planning **domain boundary criteria** are operationalised by the requirement that the constituents of the planning domain of application express the current (level of) achievement and satisfaction of these common goals.

The control specific **actual performance** is operationalised as the union of the specific actual quality and the specific actual costs. The control **worksystem boundary criteria** are operationalised by the requirement that the constituents of the control worksystem have the common goals of the current (level of) achievement and satisfaction of the control of the comfort of A in the home of A using the heating system and the leaving of A. The control **domain boundary criteria** are operationalised by the requirement that the constituents of the control domain of application express the current (level of) achievement and satisfaction of these common goals.

Specific Actual Quality

The planning specific actual domain of application has a main **abstract object** of *A*'s plans, with two **abstract attributes** of leaving plan quality and comfort plan quality. Both of these abstract plan quality attributes have attributes of: time scope; object scope; behaviour scope; view type; view content options; view format options; and content control structures. All of these plan quality attributes is **related** to the plan quality, and each plan quality is related to the overall plan quality of *A*'s plans. For example, when the planning worksystem finalises a leaving plan, the state of the time scope for the leaving plan changes to indicate when the leaving plan is to occur.

The control specific actual domain of application has two main **physical objects**: *A* and the study, where *A* works. *A* has a **physical attribute** of temperature and an **abstract attribute** of comfort. The attribute of comfort is **related** to the attribute of temperature having a range of acceptable temperatures (between 36.5°C and 37.5°C) when *A* is in the house. The second main **physical object** is the study, which has a **physical object** of its radiator and a **physical attribute** of the radiator's temperature. The temperature of the study is **related** to the temperature of *A*—an approximately linear relationship—and the temperature of the radiators—related through convection, u-value of the room, etc. The temperature of the radiator is controlled by the worksystem.

The current **states** of the temperatures of the radiators result in the **state** of the comfort **attribute** of *A* being 'not comfortable', indicated by a 'false' Boolean value, at some times. This state of the comfort attribute is a **task achieved goal** and defines the **product achieved goal** of the **actual quality** by interpretation of the relationships between this attribute and the other attributes in the current actual domain of application.

Specific Actual Costs

There are two main sub-systems in the planning worksystem: the planner (*A*); and the heating controller (a simple two-period time controller). The planner has the **physical behaviour** of feeling the temperature of *A*. The **abstract behaviours** are mainly contained in the composite behaviours (see Appendix E) of: standard monitor (type A); standard sub-plan (type A); standard monitor (type B), standard sub-plan (type B), and standard sub-plan (type 0).

The **abstract structures** of the planning worksystem include: the current and desired comfort of *A*; the current and desired temperature of *A*; the current and desired location of *A*; and the time when the heating controller turns off the heating.

There are two main sub-systems in the control worksystem: the user (*A*) and the heating system (a combination boiler system and the heating controller). The heating system has the following interacting **physical behaviours**: receive press of a one-hour boost button, turn on the LED[8], and turn off the LED. The user has the following interacting **physical behaviour**: perform press of one-hour boost button and see the LED. The non-interacting **physical behaviours** include, as examples: for the heating system, turn the heating on and off; and for the user, walk to and from the location of the heating controller. A further non-interacting **physical behaviour** of the user—and an example of a behaviour that corresponds with the transformation of the attributes of objects in the domain of application—is the closing of the front door, which changes *A*'s 'in the house' attribute state to false.

The **physical structures** can be derived from the physical behaviours, for example the heating controller has a physical structure of a one-hour boost button and the user has a physical structure of a body, including a hand that can press and an eye that can see.

The **abstract behaviours** of the heating system include turning off the heating at 7:20 in the morning, turning off the heating at the end of the boost period, and the computer operation of addition for the boost timer. The abstract behaviours of the user include forming and popping goals to boost the heating, move to the controller, and leave. The **abstract structures** of the heating system are the current boost time and the potential ordering of the heating system behaviours. The abstract structures of the user are the current state of the heating LED and the potential ordering of the user abstract behaviours

The unitary **behavioural and structural costs** as operationalised over the whole period are in Figure 25, for planning, and Figure 26, for control. The **actual costs** are operationalised by the union of these actual resource costs.

---

[8] The heating controller has a Light Emitting Diode (LED) to display the current intended state of the heating.

Main Sub-system	Cost Type	Cost
Planner	Abstract Structural Costs	81
	Physical Structural Costs	1
	Abstract Behavioural Costs	66
	Physical Behavioural Costs	1
Heating System	Abstract Structural Costs	1

Figure 25. Planning Behavioural and Structural Costs for Cycle 1 Current Solution Operationalisation.

Main Sub-system	Cost Type	Cost
User	Abstract Structural Costs	35
	Physical Structural Costs	7
	Abstract Behavioural Costs	41
	Physical Behavioural Costs	11
Heating System	Abstract Structural Costs	16
	Physical Structural Costs	19
	Abstract Behavioural Costs	18
	Physical Behavioural Costs	18

Figure 26. Control Behavioural and Structural Costs for Cycle 1 Current Solution Operationalisation.

Specific Design Problem Operationalisation

The desired operationalisation aims for a minimal expression, which is achieved by using quality and costs statements with respect to the current operationalisation.

Specific Desired Quality

The main **task goal** is to maintain the state of *A*'s comfort attribute as 'comfortable' instead of a task achieved goal of 'not comfortable'. The comfort plan quality should be acceptable. The leaving plan quality should also be acceptable, including permitting *A* to leave when he wishes.

Specific Desired Costs

The **physical structural costs** of the heating system should be within a range that allows for the preferred decrease or an acceptable increase in gas and electricity usage. It is assumed that the heating system can be modified and, therefore, the operationalisation of the **physical and abstract structural costs** of the heating system should be within a range that allows for a different installation and maintenance price. Further, it is expected that a small increase in **physical and abstract behavioural costs** of the heating system would be tolerated and this increase would be reflected in the operationalisation within a range of acceptable costs. It is assumed that the **user costs** either remain the same, or decrease if possible.

Specific Design Solution Operationalisation

Specific Actual Performance

The planning specific **actual performance** is operationalised as the union of the planning specific actual quality and the planning specific actual costs. The planning **worksystem criteria** are operationalised by the requirement that the constituents of the planning worksystem have the common goals of the actual (level of) achievement and satisfaction of the planning of the comfort of *A* and the leaving of *A*. The planning **domain boundary criteria** are operationalised by the requirement that the constituents of the planning domain of application express the actual (level of) achievement and satisfaction of these common goals.

The control specific **actual performance** is operationalised as the union of the control specific actual quality and the control specific actual costs. The control **worksystem criteria** are operationalised by the requirement that the constituents of the planning worksystem have the common goals of the actual (level of) achievement and satisfaction of the control of the comfort of *A* in the home of *A* using the heating system and the leaving of *A*. The control **domain boundary criteria** are operationalised by the requirement that the constituents of the control domain of application express the actual (level of) achievement and satisfaction of these common goals.

Specific Actual Quality

The planning and control domains of application are the same as those in the current operationalisation. The **task achieved goal** is that the state of the comfort attribute of *A* is 'comfortable' (true) for all times, as expected by a solution. This state is achieved through the **state** of the temperature **attribute** of *A* being held between the range of acceptable temperatures for *A*'s comfort. The **state** of the temperature of the study is held relatively constant by the **state** of the temperatures of the radiator. All of these states describe the **product achieved goal**.

Specific Actual Costs

There is one main sub-system in the planning worksystem: the planner (*A*). The planner has the **physical behaviour** of seeing the heating system LED. There are fewer, relative to the current, occurrences of the composite **abstract behaviours**.

The **abstract structures** of the planning worksystem remain the same.

There are two main sub-systems in the control worksystem: the user (*A*) and the heating system (a combination boiler system and a simple two-period time controller with remote advance controller). The heating system has the following interacting **physical behaviours**: receive press of front-door advance button and turn off the LED. The user has the following interacting **physical behaviours**: perform press of front-door advance button and see the LED. Examples of **physical structures** are, for the heating system, a front-door advance button and, for the user, a hand that can press.

The **abstract behaviours** for the heating system include turning off the heating on the advance button. The abstract behaviours of the user include forming and popping goals to leave and advance the heating. The **abstract structures** of the heating system are the current advance state and the potential ordering of the heating system behaviours. The abstract structures of the user are the current state of the heating LED and the potential ordering of the user abstract behaviours.

The **behavioural and structural costs** as operationalised over the whole period are in Figure 27, for planning, and Figure 28, for control. The **actual costs** are operationalised by the union of these actual resource costs.

Main Sub-system	Cost Type	Cost
Planner	Abstract Structural Costs	63
	Physical Structural Costs	1
	Abstract Behavioural Costs	50
	Physical Behavioural Costs	1

Figure 27. Planning Behavioural and Structural Costs for Cycle 1 Specific Design Solution Operationalisation.

Main Sub-system	Cost Type	Cost
User	Abstract Structural Costs	24
	Physical Structural Costs	6
	Abstract Behavioural Costs	24
	Physical Behavioural Costs	7
Heating System	Abstract Structural Costs	14
	Physical Structural Costs	23
	Abstract Behavioural Costs	10
	Physical Behavioural Costs	3

Figure 28. Control Behavioural and Structural Costs for Cycle 1 Specific Design Solution Operationalisation.

This chapter has described the Cycle 1 operationalisation. The formal and metricated operationalisation is in Appendix G. The Appendix G operationalisation is the reference for identifying initial engineering principles.

10. Cycle 2 Best-Practice Development

Chapter 4 proposed the development of an artefact specification to solve the Cycle 2 user requirements as part of the strategy for acquiring engineering principles. The Cycle 2 user requirements were selected in Chapter 4. This chapter describes in brief the best-practice development for this research of an artefact specification to solve the Cycle 2 user requirements. The artefact specification, the product of the best-practice development, is described before the best-practice development to aid comprehension.

As before, best-practice development was taken to include the application of MUSE to the user requirements. The full products for the MUSE development for this research appear in Appendix I.

Finally, the chapter describes informal evaluations from this research of the artefact specification against the user requirements. The evaluation was positive, i.e. informally, the artefact specification indeed fulfils the user requirements. This positive evaluation means that this Cycle 2 best-practice development supports the operationalisation of the Cycle 2 specific design problem and its solution, which is presented in the following chapter.

User Requirements

The user requirements for Cycle 2 are repeated below:

> 'The kitchen is usually a very comfortable room, probably because it has thick walls. However, it can get too hot when *D* is cooking, even in the winter. The room has three radiators that have individual thermostats. These radiators are heated using hot water from a gas-powered combination boiler that is in another room. There is no central thermostat for the boiler, but there is a time-controller and a water temperature controller, neither of which are in the kitchen. The boiler supplies other radiators in the house. There is an extractor fan over the cooker, but it is broken. The windows, which are double-glazed, are difficult to open due to security fittings. An outside door is sometimes opened when the room is too hot. A decrease in the gas bill is desirable.'

Artefact Specification

The artefact specification was developed from the best-practice described in the following sections. It involved the provision of additional cooling[9] and supporting *D*'s cooking and cooling planning and control. A new fan was specified to provide additional cooling, rather than repair the broken extractor-fan, which, even if mended, would not provide significant cooling.

The planning support designed for the cooking activities and the cooling is a pre-printed A3 surface covered in laminated plastic (Appendix I). The surface is written on by *D* with a water-soluble pen so that changes can be made, including starting a new plan. Two pens are available, one for planning and the other for re-planning during cooking. The pre-printing provides prompts and space for an explicit representation of the plans and some of their criteria.

A controller is provided for the door, the fan, and the radiators in the kitchen. The controller permits entry and display of the heating plan as it relates to the cooking time. A pre-printed booklet covered in laminated plastic supports the documentation of previous times of cooking activities to support future cooking planning. Instructions are printed on the front of the booklet.

Best Practice Development

As determined by the research strategy, the MUSE method and HCI guidelines were applied for this research to the user requirements. The resulting artefact specification is described in the previous section. The MUSE products for the development are in Appendix I.

Information Elicitation and Analysis Phase

The current extant system was analysed in detail. Other extant systems were listed but not analysed, since a satisfactory artefact specification was delivered by the first MUSE iteration. Two Task Description MUSE products were produced:

- Three scenarios were elicited by paper-based questioning. All of the scenarios involved *D* cooking meals that resulted in her becoming too hot.

---

[9] To relate to domestic energy management systems, 'heating/cooling' is used rather than 'cooling' in the Cycle 2 MUSE application.

- *D* was observed and questioned concerning the complete range of cooking tasks, including meal planning, shopping, and cooking planning.

These Task Descriptions were generalised (Generalised Task Model of the extant system MUSE product). Valuable observations, design implications, and speculations arose during this phase. For example, it was recognised that the kitchen door provided effective cooling, but that it was not often opened, leading to the implication that the kitchen door should be opened more often to provide cooling. For another example, it was observed that *D* became 'flustered' during cooking, contributing to *D* becoming over heated, implying that a reduction in *D* becoming flustered would lessen the problem.

The initial task-level conceptual design of the target system (General Task Model of the target system MUSE product) documented the essential design decisions for:

- More and earlier planning of the cooking.
- Early planning of the heating.
- Turning off the heating, even in winter if necessary.
- Support for re-planning during cooking.
- Support for improving future planning.

At the initial task-level conceptual design stage, it was evident that there was not a requirement for detailed porting. Therefore, no further MUSE extant systems products were developed.

As in Cycle 1, analysis during the Information Elicitation and Analysis phase was the basis for the design in the other phases, and the time spent on the other phases was relatively short.

Design Synthesis Phase

A textual summary of the human factors concerns (Statement of User Needs MUSE product) detailed:

- Any explicit design criteria, such as the amount of fuel used cannot increase very much and desirably would decrease.
- Any implicit design criteria, such as the artefact cost should be low.

- Any explicit system performance criteria, such as *D* must not be too hot.
- Any implicit performance criteria, such as *D* must be able to cook the meals that she desires when she wishes.
- Any relevant human factors knowledge, such as feedback and consistency guidelines should be followed. In particular, the following guideline was applied: 'Use familiar material, situations, working methods, and relevant analogies to engender good user performance.' (Gardner and Christie, 1987).

The conceptual design of the conjoint user and computer tasks was advanced (Composite Task Model MUSE product). Important design decisions were rationalised at this stage: the provision of an additional fan; when the door should be opened, the fan turned on, and the radiators turned off; and the desired explicitness of the cooking and heating plan. The desired increase in meal and heating planning suggested support for both types of planning. A controller was rationalised to off-load the control during cooking.

The on-line tasks were decomposed (System Task Model MUSE product) to support the ordering of cooking and heating planning. Internal iteration delivered a 'bubbled up' rationale for two devices: one for planning and another for control. Tasks were identified to support these two devices, and the transfer of information from the planning support to the control support.

Design Specification Phase

The interaction-level design was advanced (Interaction Task Model and Interface Model MUSE products). A paper-based planning and memory aid covered in laminated plastic was rationalised to support:

- Being cleaned and amended during use, facilitated by the use of water-soluble markers.
- Being lightweight, it can be carried around, both in the kitchen and out.
- Being readily available and visible, in that it can be stood-up and it is yellow.
- Cleanliness in a food environment.

Computer-supported planning was rejected, since putting a computer in the kitchen would be inconvenient and *D* has a dislike of electronic gadgets. Lines were rationalised on the aid to show the planned length of cooking activities, with their thickness showing the effort.

An electronic controller was rationalised to control the radiators in the kitchen and the fan, and to remind *D* when to open the door. Consistency between the planning aid and the controller was maintained to improve transfer between the devices.

The 'screen' layouts of the planning aid and controller were designed (Display Design MUSE product).

Evaluation

Three informal analytic assessments of whether the artefact fulfils the user requirements were conducted, apart from the assessment of consistency through the application of MUSE. Firstly, an analytic argument was constructed to show that the introduction of the artefact into the kitchen of *D* should remove the problem. A form of this analytic argument, commensurate with the user requirements, follows:

> 'The artefact should support *D* in improved planning of meals, the activities involved in generating meals, and the required heating. Improved planning of the meals and their activities should prevent *D* from becoming flustered during meal preparation. Improved planning of the heating should enable *D* to control the heating so that she will be kept cool at all times during cooking.'

The second informal analytic assessment involved a panel of seven practitioners, five human factors engineers and two software engineers, appraising the artefact specification produced using MUSE. They were all familiar with the method and the user requirements. No objections were maintained such that the artefact was considered to fail to fulfil the user requirements.

The third, and last, informal analytic assessment was an expert walkthrough of the artefact specification performed by a human factors engineer (other than the author). His report contained the following concluding statement:

> 'Based on my examination of the meal planning aid and my discussions with *D*, it is my opinion that use of the planning aid is likely to result in improved meal planning and less heat in the kitchen at busy points during meal preparation. *D* should therefore not become flustered and too hot. Due to the effort involved in planning, I anticipate that the sheet will probably only be required for

> more complicated meals; it is reasonable to expect that use of the sheet on these occasions will result in a 'transfer of training' to simple meals, and *D*'s awareness of the need for ventilation and cooling of the kitchen will be improved as a result of using it. Initially, I was concerned that use of the planner would be abandoned during busy periods in the kitchen, exactly when it is required most. However, *D* appears to be of a very methodical nature, always planning meals well in advance and preparing and using a detailed shopping list. Given *D*'s existing use of lists, and the apparent satisfaction derived from making and executing plans, I would expect *D* to find using the meal planner during busy periods both natural and easy.'

In addition, an empirical assessment has been performed by constructing from this research an interactively faithful prototype of the planning aid and the controller. The prototype was employed in cooking a complicated meal that would normally be expected to cause *D* to become too hot. *D* was less hot, and was not flustered. Minor changes were proposed, for example using numbers instead of lines to represent timing and effort, and implemented (Figure 29). In a second empirical assessment, *D* was not hot. This assessment confirms the analytic argument.

Taken together, the analytic and empirical assessments demonstrate, albeit informally, that the artefact specification indeed fulfils the user requirements.

The artefact specification developed for this research and described in this chapter informally fulfils the Cycle 2 user requirements. Therefore, this Cycle 2 best-practice development supports the operationalisation of the Cycle 2 specific design problem and its solution, which is presented in the following chapter.

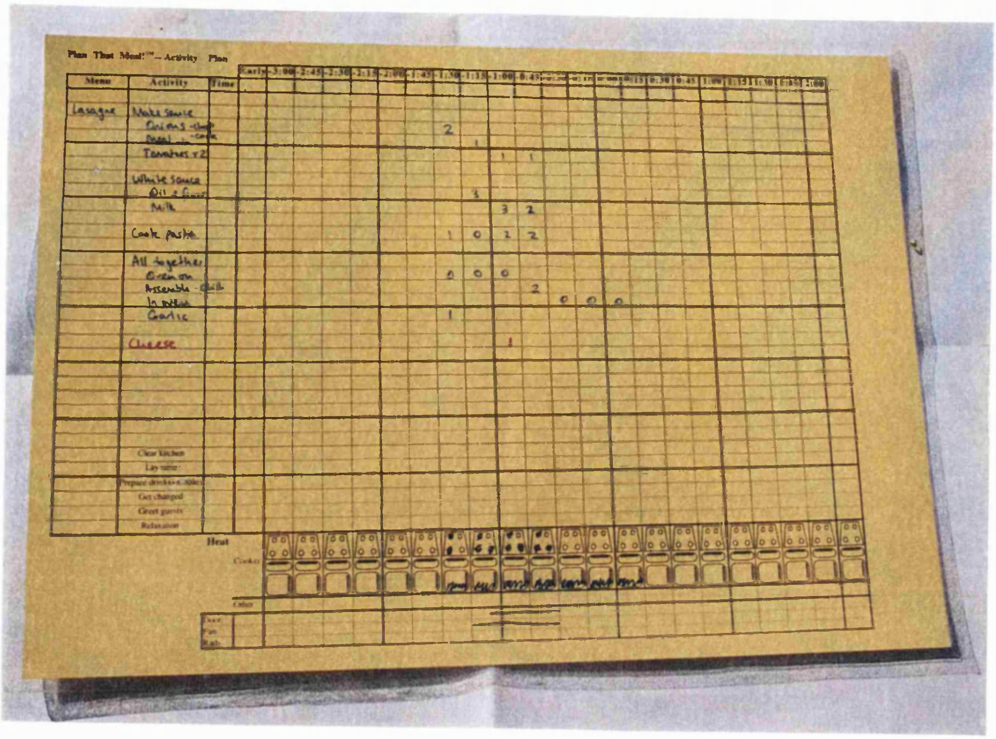


Figure 29. Planning Aid Prototype.

11. Cycle 2 Operationalisation

Chapter 4 proposed the operationalisation of the Cycle 2 specific design problem and its solution as part of the strategy for acquiring engineering principles. The Cycle 2 best-practice development for this research was described in the previous chapter. This chapter describes in brief the operationalisation for this research of the Cycle 2 specific design problem and its solution. The current solution operationalisation is described before the specific design problem and its solution operationalisation.

The operationalisations are of the specific design problem and solution conceptions (Chapter 2), the conception of human-computer systems (Chapter 5), and the conception of planning and control (Chapter 7). The operationalisation was developed by starting with an explicit operationalisation (similar to the brief description in this chapter) to support the formal and metricated operationalisation. The formal and metricated operationalisation, using the frameworks (Chapter 6), is in Appendix J.

Following the strategy, the **conative and affective abstract behaviours and structures** are not operationalised here. Videos, with concurrent verbal protocols, of the cooking and planning were analysed to support the operationalisation.

Generality Concern

Due to unfortunate circumstances, *D* was unable to provide access for the operationalisation[10]. The researcher (*A*) recreated the conditions to support the operationalisation. He cooked one of *D*'s recipes with and without the planning sheet in a similar environment to *D*. He became hot and flustered without the planning sheet, but was not hot or flustered with the planning sheet. The cooking was conducted in the kitchen in his home. The kitchen has a window and a back door. A fan was fitted. The heating controller was not used, since the heating controls are in the kitchen and the fan switch was readily accessible.

Two issues arise due to these circumstances:

- *A* did not 'own' the problem. However, on introduction of the artefact, he did have the problem and it was solved by use of the prototype artefact.

---

[10] Unfortunately, she broke her leg.

- Generalisation over two different users would have been a valuable step in initial engineering principle acquisition.

The last of these issues will need to be addressed by further research.

Current Solution Operationalisation

Specific Actual Performance

The planning specific **actual performance** is operationalised as the union of the planning specific actual quality and the planning specific actual costs. The planning **worksystem boundary criteria** are operationalised by the requirement that the constituents of the planning worksystem have the common goals of the current (level of) achievement and satisfaction of the planning of the cooking of *A* and the heating of *A*. The planning **domain boundary criteria** are operationalised by the requirement that the constituents of the planning domain of application express the current (level of) achievement and satisfaction of these common goals.

The control specific **actual performance** is operationalised as the union of the specific actual quality and the specific actual costs. The control **worksystem boundary criteria** are operationalised by the requirement that the constituents of the control worksystem have the common goals of the current (level of) achievement and satisfaction of the control of the cooking of *A* and the heating of *A* in the kitchen of *A* using the kitchen's cooker and door. The control **domain boundary criteria** are operationalised by the requirement that the constituents of the control domain of application express the current (level of) achievement and satisfaction of these common goals.

Specific Actual Quality

The planning specific actual domain of application has a main **abstract object** of *A*'s plans, with two **abstract attributes** of cooking plan quality and heating plan quality. These two plan quality attributes both have attributes of: time scope; object scope; behaviour scope; view type; view content options; view format options; and content control structures. Each of these plan quality attributes is **related** to the plan quality, and each plan quality is related to the overall plan quality of *A*'s plans. For example, when the planning worksystem finalises a cooking plan the state of content structure changes to reflect the next ingredient required for the cooking.

The control specific actual domain of application has two main **physical objects**: *A* and the kitchen. There is one main **abstract object** of the meal. *A* has **physical attributes** of temperature and activity, which are **related** to the abstract attributes of comfort and agitation. The kitchen has **physical objects** of the cooker, the radiators, and the door. The **physical attributes** of the temperature of the cooker, the temperature of the radiators, and the airflow of the door are related to the abstract attribute of the temperature of the kitchen, which is related to *A*'s temperature. The temperature of the cooker and door are controlled by the worksystem. The meal has an abstract attribute of quality, which is related to its physical attributes of flavour, presentation, and location.

The current **states** of the door's airflow and the temperature of the cooker result in the **state** of the comfort **attribute** of *A* being 'not comfortable' (false), the **state** of the agitation **attribute** of *A* being 'agitated' (a high percentage), and the **state** of the quality **attribute** of the meal being 'poor', with a value of 7.3, at some times (as against a possible 10 for 'excellent'),. These states are **task achieved goals** and define the **product achieved goal** of the **actual quality** by interpretation of the relationships between this attribute and the other attributes in the actual domain of application.

Specific Actual Costs

There is one main sub-system in the planning worksystem: the planner (*A*). The planner has the **physical behaviour** of seeing the current ingredients used in the cooking. The **abstract behaviours** are contained in the composite behaviours of standard monitor (Type 0) and standard sub-plan (Type 0).

The abstract structures of the planning worksystem include: the current and desired ingredients of the meal; and the current and desired temperature of *A*.

There are two main sub-systems in the control worksystem: the user (*A*) and the cooker. The cooker has the interacting **physical behaviour** of change the level of the gas ring or the oven (a composite behaviour). Correspondingly, the user has the interacting **physical behaviours** of change the level of the gas ring and the oven.

The **physical structures** can be derived from the physical behaviours, for example the cooker has a physical structure of a gas ring, and the user has a physical structure of a hand (that can change the level of the gas of a ring).

The **abstract behaviours** of the cooker include increasing the ring/oven heat on a clockwise turn of a knob. The abstract behaviours of the user include forming and popping goals to make lasagne, cook onions, collect pasta from the cupboard, and assemble lasagne.

The **behavioural and structural costs** as operationalised over the whole period are in Figure 30, for planning, and Figure 31, for control. The **actual costs** are operationalised by the union of these actual resource costs.

Main Sub-system	Cost Type	Cost
Planner	Abstract Structural Costs	91
	Physical Structural Costs	2
	Abstract Behavioural Costs	214
	Physical Behavioural Costs	5

Figure 30. Planning Behavioural and Structural Costs for Cycle 2 Current Solution Operationalisation.

Main Sub-system	Cost Type	Cost
User	Abstract Structural Costs	256
	Physical Structural Costs	65
	Abstract Behavioural Costs	382
	Physical Behavioural Costs	70
Heating System	Abstract Structural Costs	90
	Physical Structural Costs	33.4825
	Abstract Behavioural Costs	114
	Physical Behavioural Costs	38

Figure 31. Control Behavioural and Structural Costs for Cycle 2 Current Solution Operationalisation.

Specific Design Problem Operationalisation

The desired operationalisation aims for a minimal expression, which is achieved by using quality and costs statements with respect to the current operationalisation.

Specific Desired Quality

The main **task goal** is to maintain the state of *A*'s comfort attribute as 'comfortable', *A*'s agitation attribute as 'not agitated', and the meal's quality attribute as 'good'.

Specific Desired Costs

The **physical structural costs** of the heating system should be within a range that allows for a desirable decrease or acceptable increase in gas and electricity usage. It is assumed that the heating system can be modified and, therefore, the operationalisation of the **physical and abstract structural costs** of the heating system should be within a range that allows for a different installation and maintenance price. Further, it is expected that a small increase in **physical and abstract behavioural costs** of the heating system would be tolerated and this increase would be reflected in the operationalisation within a range of acceptable costs. It is assumed that the **user costs** either remain the same or decrease if possible.

Specific Design Solution Operationalisation

Specific Actual Performance

The planning specific **actual performance** is operationalised as the union of the planning specific actual quality and the planning specific actual costs. The planning **worksystem criteria** are operationalised by the requirement that the constituents of the planning worksystem have the common goals of the actual (level of) achievement and satisfaction of the planning of the cooking of *A* and the heating of *A*. The planning **domain boundary criteria** are operationalised by the requirement that the constituents of the planning domain of application express the actual (level of) achievement and satisfaction of these common goals.

The control specific **actual performance** is operationalised as the union of the control specific actual quality and the control specific actual costs. The control **worksystem criteria** are operationalised by the requirement that the constituents of the planning worksystem have the common goals of the actual (level of) achievement and satisfaction of the control of the cooking of *A* and the heating of *A* in the kitchen of *A* using the kitchen's cooker, door, and fan. The control **domain boundary criteria** are operationalised by the requirement that the constituents of the control domain of application express the actual (level of) achievement and satisfaction of these common goals.

Specific Actual Quality

The planning and control domains of application are the same as those in the current operationalisation. During development, an iteration was required to ensure that the current (and, therefore, problem) operationalisation domain of application was the same as that for the solution application.

The specific actual quality has a **task achieved goal** that such that the state of *A*'s comfort attribute is 'comfortable', *A*'s agitation attribute 'not agitated', and the meal's quality attribute 'good'. The states of these attributes are achieved by: the state of *A*'s temperature attribute being held between the range of acceptable temperatures for *A*'s comfort; the rate of change in *A*'s activity being low; and the state of the meal's flavour, presentation, and location being tasty, well presented, and on the table respectively. All of these states describe the **product achieved goal**.

Specific Actual Costs

There are two main sub-systems in the planning worksystem: the planner (*A*) and the planning-aid. The planner has the **physical behaviour** of, as before, seeing the current ingredients, and, for the solution, seeing and writing on the planning-aid. The planning-aid has physical behaviours of displaying and accepting writing. The **abstract behaviours** of the planner are contained in the composite behaviours of: sheet sub-plan (Type 0), standard monitor (Type 0), and standard sub-plan (Type 0).

The abstract structures of the planning worksystem include: the current and desired ingredients of the meal; and the current and desired temperature of *A*.

There are four main sub-systems in the control worksystem: the user (*A*), the cooker, the door, and the fan. The fan has the interacting **physical behaviour** of accept button press to turn on. Correspondingly, the user has the interacting **physical behaviours** of press button to turn on the fan.

The **behavioural and structural costs** as operationalised over the whole period are in Figure 32, for planning, and Figure 33, for control. The **actual costs** are operationalised by the union of these actual resource costs.

Main Sub-system	**Cost Type**	**Cost**
Planner	Abstract Structural Costs	160
	Physical Structural Costs	5
	Abstract Behavioural Costs	1232
	Physical Behavioural Costs	51
Planning Sheet	Abstract Structural Costs	3
	Physical Structural Costs	4
	Abstract Behavioural Costs	35
	Physical Behavioural Costs	35

Figure 32. Planning Behavioural and Structural Costs for Cycle 2 Specific Design Solution Operationalisation.

Main Sub-system	**Cost Type**	**Cost**
User	Abstract Structural Costs	300
	Physical Structural Costs	75
	Abstract Behavioural Costs	488
	Physical Behavioural Costs	125
Heating System	Abstract Structural Costs	90
	Physical Structural Costs	33
	Abstract Behavioural Costs	102
	Physical Behavioural Costs	34

Figure 33. Control Behavioural and Structural Costs for Cycle 2 Specific Design Solution Operationalisation.

This chapter has described the Cycle 2 operationalisation. The formal and metricated, operationalisation is in Appendix J. The Appendix J operationalisation is the reference for identifying initial engineering principles.

12. Initial Engineering Principles

Chapter 4 describes a strategy for acquiring engineering principles. The research aim is to implement and assess this strategy by acquiring examples of initial engineering principles.

These examples of initial engineering principles are acquired by 'identifying general relationships between specific design problems and their solutions' (Chapter 4). Two specific design problems and their solutions have been operationalised (Chapters 9 and 11), based on best-practice development (Chapters 8 and 10). This chapter reports examples of initial engineering principles acquired from the formal and metricated operationalisations (Appendices G and J).

The chapter starts by presenting a detailed strategy developed for this research. This strategy clarifies the generality of the relationships and the 'identify' process. The generality of the relationships depends on: commonalities, which include composite structures, parameters, and null concepts; and types of user, computer, and domain. The 'identify' process is specified as targeting relationships that are more likely to be general, and six means of targeting are proposed. General relationships, either within or between operationalisations, constitute initial engineering principles.

The rest of the chapter details each means of targeting, with example initial engineering principles for each. The initial engineering principles are presented in a formal notation. The notation is employed for precision, however, initially and where suitable, an equivalent in words is also provided. Even in the simplest cases, the word equivalent is not as precise as the notation, and in cases that are more complex, the word equivalent is unwieldy. The notation is introduced as required during the chapter.

Consideration of the status of the acquired initial engineering principles and the strategy assessment is in the following Strategy Assessment and Discussion chapter.

Detailed Strategy

Specific Design Problem

The operationalisation of the current solution was included in the research strategy to support operationalisation of the specific design problem. However, the specific design problem operationalisations are minimalist, requiring the contrast of the current solution, and not supportive of generalisation. Therefore, the operationalisations of the current solution are considered included with the operationalisations of the specific design problem for this detailed strategy. This inclusion could be interpreted as design practice requiring diagnosis to support prescription.

Generality of the Relationships

The research strategy stated that initial engineering principles were 'general relationships between specific design problems and their solutions'. Therefore, initial engineering principles need to contain parts of the operationalisations of the specific design problems and their solutions. They need to have generality between these operationalisations to operationalise a general design problem and its solution. The initial engineering principles can have generality through:

- Commonalities, including composite structures (Chapter 6 and Appendix E), parameters, and null concepts.
- Cycle types, which arise from the earlier (Chapter 4) statement that: 'a general design problem and its general design solution are general over types of user, types of computer and types of domain of application'.

Commonalities

The following concepts can be common between the operationalisations, for both the specific design problem and its solution:

- Structures, including composite structures, and their state changes
- Behaviours, and their ordering.
- Domain objects, attributes, states, and the state changes.

Parameterisation, as employed in the composite structures, can be used to recognise generality at a higher level of description, when there is no generality at a lower level of description. If a concept is null, i.e. not operationalised, in an initial engineering principle, then the initial engineering principles is general over those concepts. For example, if the desired users costs are not operationalised, then the initial engineering principle is general for all desired user costs.

Identify

There are many potential generalities between the operationalisations. However, identifying them is difficult. Six means of targeting are proposed to support identification of generality, particularly generality to support design. These six means are:

- *Initial engineering principles identified during operationalisation(s).* Generalisation occurs during examination of the operationalisations.
- *Initial assumption assessment from operationalisation(s).* The initial assumptions—the underlying conceptions—can be assessed, since they are intended to be general.
- *Inspirational initial engineering principles from operationalisation(s).* During operationalisation and investigation, potential initial engineering principles were noticed and noted.
- *Initial engineering principles from general guidelines.* Guidelines are general design knowledge.
- *Initial engineering principles from MUSE guidelines.* Guidelines that are specific to the Cycles were acquired during MUSE application.
- *Initial engineering principles from MUSE tasks.* The MUSE task diagrams also contain guidelines that are specific to the Cycles.

The six means of targeting are detailed later, with examples from the cycle operationalisations.

Inter- and Intra-Initial Engineering Principles

Following the operationalisations for this research, it is suggested that commonalities can be within an operationalisation as well as between operationalisations. Those within will be termed 'intra-initial engineering principles' while those between 'inter-initial engineering principles'.

Intra-initial engineering principles could be interpreted as general relationships between specific design 'sub-problems' and their solutions. However, with this understanding, their sub-problem basis probably rests on their being part of an overall problem.

Both inter- and intra-initial engineering principles are exemplified here.

Cycle 1 and 2 User Types

The user in both the cycles was *A*. The types of user for *A* include:

- Researcher.
- Male.
- Aged 32.
- Postgraduate.
- Etc.

These lists of types could be developed widely, as shown by the 'Etc.'. For example, for the types of user list above, Neale and Liebert (1980) suggest further types, or 'external validity': 'population validity, geographic areas validity, temporal validity, and [designer] validity'.

Cycle 1 Heating Controller Types

The types of heating controller in Cycle 1 include:

- Simple controller.
- Two-period controller.
- Heating controller.
- Domestic heating controller.
- Domestic energy management system.
- Energy management system.
- Etc.

Cycle 1 Heating System Types

The types of heating system in Cycle 1 include:

- Combination boiler heating system.
- Gas-powered heating system.
- Energy delivery system.
- Etc.

Cycle 2 Cooker Types

The types of cooker in Cycle 2 include:

- Upright cooker.

- Gas cooker.
- Domestic cooker.
- Etc.

Cycle 1 Domain Types

The types of the domain in Cycle 1 include:

- Comfort planning and control.
- Leaving planning and control
- Domestic energy management
- Energy management
- Late comfort planning
- Late leaving planning
- Etc.

Cycle 2 Domain Types

The types of the domain in Cycle 2 include:

- Comfort planning and control.
- Cooking planning and control
- Domestic energy management
- Energy management
- Late comfort planning
- Minimal cooking planning
- Etc.

Generalisation Over Types

Inter-initial engineering principles will require generalisation over the above types. Generalisation can occur in two ways:

- Types that are common to both cycles are carried forward to those of the inter-initial engineering principle.

- The power set of types that are not common to both cycles are carried forward to those of the inter-initial engineering principle.

Terminology

In the rest of this chapter, 'Op1' refers to the Cycle 1 operationalisation and 'Op2' to the Cycle 2 operationalisation.

Initial Engineering Principles Identified during Operationalisation(s)

Strategy

The means of targeting that is closest to the original strategy is to 'identify' initial engineering principles, both within operationalisations (intra-initial engineering principles) and between operationalisations (inter-initial engineering principles). These initial engineering principles may be in the formulae or in the values of the operationalisation tables. 'Identify' in this case requires an iterative search.

The initial engineering principles may be across, down, or both the values of the tables. Across relates to the initial engineering principle between the changes of behaviours, structures, and states of the domain. Down relates to the changes for a particular behaviour, structure, or state of the domain.

Examples

Examples 1 to 8 are examples of intra-initial engineering principle identification within Op1 and Op2.

Example 9 is an example of an inter-initial engineering principle identified between Op1 and Op2.

Example 1—Within Op1

In the current solution, StMonA is always followed directly by StSubPlanA. StMonA and StSubPlanA are composite structures, and are described in Chapter 6. The formulae show this outcome as the same true value being taken from the previous behaviour and the previous event, for example:

F11	A:StMonA:FP, FeelTemp, Temp, Comfort	
G11	A:StSubPlanA:RP, In house, Comfort	=F10

This outcome is common to many of the formulae. To represent this outcome across the occurrences of the behaviours in the current solution requires parameters. This intra-initial engineering principle can be shown in a notation:

$$A : \text{StMonA} : X, Y, Z \xrightarrow[1, e=1]{} A : \text{StSubPlanA} : P, Q, R$$
$$\Downarrow$$

Where:

- X, Y, Z and P, Q, R are parameters.
- The horizontal arrow (→) shows a 'followed by' relationship.
- The first equation (1) under the horizontal arrow shows the likelihood, i.e. probability, of the 'followed by' relationship. In this case, it is 'always follows'.
- The second equation (e=1) under the horizontal arrow shows the number of event ticks in the 'followed by' relationship. In this case, it is one event tick.
- The double down arrow (⇓) shows the direction of design. In this initial engineering principle, the solution concepts are null.

This initial engineering principle can be expressed in words as:

> 'Within Op1 in the problem component, the Type A standard monitoring is always followed after one event tick by Type A standard sub-planning'.

Example 2—Within Op1

There is a similar intra-initial engineering principle to that in Example 1 within Op1:

$$A : \text{StMonB} \xrightarrow[1, e=1]{} A : \text{StSubPlanB}$$
$$\Downarrow$$

This initial engineering principle can be expressed in words as:

> 'Within Op1 in the problem component, the Type B standard monitoring is always followed after one event tick by Type B standard sub-planning'.

Example 3—Within Op1

In the specific design solution, there is an intra-initial engineering principle similar to those in Examples 1 and 2:

$$\Downarrow$$
$$\text{A : StMonA : X, Y, Z} \xrightarrow[1,e=1]{} \text{A : StSubPlan : P, Q}$$

Note the location of the double down arrow, to indicate that the problem concepts are null.

Example 4—Within Op1

Combining the intra-initial engineering principle from Example 3 with that from Example 1, results in a further following initial engineering principle:

$$\text{A : StMonA : X, Y, Z} \xrightarrow[1,e=1]{} \text{A : StSubPlanA : P, Q, R}$$
$$\Downarrow$$
$$\text{A : StMonA : A, B, C} \xrightarrow[1,e=1]{} \text{A : StSubPlan : M, N}$$

This initial engineering principle has problem concepts and solution concepts that are not null.

It is possible to relate the parameters in this initial engineering principle, where they represent increased generality:

$$\text{A : StMonA : X, Y, Z} \xrightarrow[1,e=1]{} \text{A : StSubPlanA : P, Q, Z}$$
$$\Downarrow$$
$$\text{A : StMonA : X, B, Z} \xrightarrow[1,e=1]{} \text{A : StSubPlan : P, Z}$$

This initial engineering principle can be expressed in words as:

> 'Within Op1: in the problem component, the Type A standard monitoring is always followed after one event tick by Type A standard sub-planning; and in the solution component, the Type A standard monitoring is always followed after one event tick by standard sub-planning'.

Example 5—Within Op1

An initial engineering principle can include structure states. The following intra-initial engineering principle includes a structure state.

$$\Downarrow$$
$$\begin{array}{c} \text{A : StSubPlan : RP, Comfort} \\ \downarrow 1, e = 0 \\ \text{A : CDd : Desired Comfort} = \text{TRUE} \end{array}$$

Where:

- The single down arrow ($\downarrow$) shows the structure state that 'follows' the behaviour occurrence.
- The first equation (1) beside the single down arrow shows the likelihood, i.e. probability, of the structure's state. In these cases, they are 'always follow'.
- The second equation beside the single down arrow (e=0) shows the number of event ticks in the 'follow' relationship. In these cases, they occur in the same event.

This initial engineering principle can be expressed in words as:

> 'Within Op1 in the solution component, the standard sub-planning of Comfort always results in the control domain planning structure's desired comfort attribute state being true'.

Example 6—Within Op1

An initial engineering principle can also include domain states. The following intra-initial engineering principle, based on Example 5, includes a domain state:

$$\Downarrow$$
$$\begin{array}{l} \quad \text{A : StSubPlan : RP, Comfort} \\ \qquad\quad \downarrow 1, e = 0 \\ \text{A : CDd : Desired Comfort} = \text{TRUE} \\ \underset{1,e=0}{\mapsto} \textit{Control Comfort} = \textit{TRUE} \end{array}$$

Where:

- The horizontal bar arrow ($\mapsto$) shows the domain state that 'follows' the behaviour occurrence.
- The first equation (1) under the horizontal bar arrow shows the likelihood, i.e. probability, of the domain's state. In this case, it is 'always follow'.
- The second equation (e=22) under the horizontal bar arrow shows the number of event ticks in the 'follow' relationship. In this case, it occurs after no event ticks.

This initial engineering principle expressed in words as:

> 'Within Op1 in the solution component, the standard sub-planning of Comfort always results in the control domain planning structure's desired comfort attribute state being true and the Control domain Comfort state being true after no event ticks'.

Example 7—Within Op1

Following Example 5, this example presents a more complex initial engineering principle:

$$\begin{array}{ccc} \mathrm{A : StMonA : X,Y,Z} & & \mathrm{A : StSubPlanA : P,Q,Z} \\ \downarrow_{1,0} & \xrightarrow[1,e=1]{} & \downarrow_{1,0} \\ \mathrm{A : CDc : Current\ Z = FALSE} & & \mathrm{A : CDd : Desired\ Z = TRUE} \end{array}$$

$$\Downarrow$$

$$\begin{array}{ccc} \mathrm{A : StMonA : Z,B,Z} & & \mathrm{A : StSubPlan : P,Z} \\ \downarrow_{1,0} & \xrightarrow[1,e=1]{} & \downarrow_{1,0} \\ \mathrm{A : CDc : Current\ Z = TRUE} & & \mathrm{A : CDd : Desired\ Z = TRUE} \end{array}$$

This initial engineering principle can be expressed in words as:

> 'Within Op1: in the problem component, the type A standard monitoring for Z, which always results in the control domain's planning structure's current Z state changing to false, is always followed after one event tick by Type A standard sub planning for Z, which always results in the control domains planning structure's desired Z state changing to true;
>
> in the solution component, the type A standard monitoring for Z, which always results in the control domain's planning structure's

current Z state changing to true, is always followed after one event tick by Type A standard sub planning for Z, which always results in the control domains planning structure's desired Z state changing to true'.

Example 8—Within Op2

The following intra-initial engineering principle holds within Op2:

$$\Downarrow$$
$$\text{A : ShSubPlan : RP, Cooker heat}$$
$$\underset{1,e=0}{\mapsto} \textit{Control Comfort} = \textit{TRUE}$$

This initial engineering principle can be expressed in words as:

> 'Within Op2 in the solution component, the sheet sub-planning of the Cooker heat always results in the Control domain Comfort state being true after no event ticks.'

Example 9—Between Op1 and Op2

Identifying a commonality between two intra-initial engineering principles will identify an inter-initial engineering principle. For example, An inter-initial engineering principle from Examples 5 and 8 can be based on a composite structure. Generalising the StSubPlan and ShSubPlan composites to a StShSubPlan: X, Y, Z, where Z can be Store or Write (see Chapter 4 and Appendix E) and Y can be 'Comfort' or 'Cooker heat', leads to an inter-initial engineering principle between Op1 and Op2:

$$\Downarrow$$
$$\text{A : StShSubPlan : RP, Y, Z}$$
$$\underset{1,e=0}{\mapsto} \textit{Control Comfort} = \textit{TRUE}$$

This initial engineering principle can be expressed in words as:

> 'Between Op1 and Op2 in the solution component, the standard sheet sub-planning of the Comfort or the Cooker heat always results in the Control domain Comfort state being true after no event ticks.'

Initial Assumption Assessment from Operationalisation(s)

Strategy

The initial assumptions—the underlying conceptions—can be assessed, given that they have been operationalised successfully. Since these initial assumptions are intended to be general, they provide a basis for generality. For example, the 'monitor→plan→monitor' conception of planning and control is operationalised in both Cycle 1 and Cycle 2 operationalisations, and therefore it is general over both.

Example

The assumption that 'the monitor→plan→monitor conception of planning and control is operationalised in both operationalisations, and therefore is general over both, can be assessed.

'Monitor' can be considered to be a StMon or an StMonA behaviour, and 'plan' to be an StSubPlanA, an StSubPlanB, an StSubPlan, or a ShSubPlan behaviour. The following corollaries from the assumption might result in generalities:

1. If there is a monitor behaviour, then it will be followed by a plan behaviour.

$$\text{Monitor} \xrightarrow[1,e\geq 1]{} \text{Plan}$$
$$\Downarrow$$
$$\text{Monitor} \xrightarrow[1,e\geq 1]{} \text{Plan}$$

   This condition holds in Op1 and Op2, so it is an inter-initial engineering principle between Op1 and Op2.

2. If there is a plan behaviour, then it will always be followed by a monitor behaviour.

$$\text{Plan} \xrightarrow[1,e\geq 1]{} \text{Monitor}$$
$$\Downarrow$$
$$\text{Plan} \xrightarrow[1,e\geq 1]{} \text{Monitor}$$

   This condition is violated in both the Op1 and the Op2 current solutions and specific design solutions. It is probably violated because the assumption does not anticipate an end to planning and monitoring.

3. There will not be a monitor behaviour directly followed by a monitor behaviour.

$$\text{not (Monitor} \xrightarrow[1,e=1]{} \text{Monitor)}$$

This condition holds in Op1 and Op2, so is an inter-initial engineering principle between Op1 and Op2.

4. There will not be a plan behaviour directly followed by a plan behaviour.

$$\text{not (Plan} \xrightarrow[1,e=1]{} \text{Plan)}$$

This condition is violated in the Op1 current solution and the Op2 specific design solution.

Inspirational Initial Engineering Principle from Operationalisation(s)

Strategy

During operationalisation, potential initial engineering principles were noticed and noted. Further potential initial engineering principles become apparent while investigating other initial engineering principles, and these initial engineering principles were noted. These 'theories' are worthy of investigation. For example, it appeared during operationalisation that 'To achieve comfort with energy management systems in the home, prescribe 'late' control of the heating system by the user'.

Example 1—To achieve comfort with energy management systems in the home, prescribe 'late' control of the heating system by the user

This theory was developed from Op1, and 'late' can be understood with respect to Op1. In Op1, the control of the heating was moved to the end of the behaviours. In notation, an intra-initial engineering principle within Op1 is:

$$\text{Plan : ...Comfort} \xrightarrow[1,e\geq 1]{} \text{Plan : ...In house}$$

$$\Downarrow$$

$$\text{Plan : ...In house} \xrightarrow[1,e\geq 1]{} \text{Plan : ...Comfort}$$

A more general inter-initial engineering principle based on the above can be sought in Op2:

$$\text{Plan} : \ldots \text{X} \xrightarrow[1,e\geq 1]{} \text{Plan} : \ldots \text{P}$$
$$\Downarrow$$
$$\text{Plan} : \ldots \text{P} \xrightarrow[1,e\geq 1]{} \text{Plan} : \ldots \text{X}$$

Unfortunately, there is no obvious case. This type of case can be identified as a *counter-principle*—where it is certain that there is no such initial engineering principle within or between operationalisations. Further operationalisations would enable more detailed generality.

Example 2—More specific monitor→plan

Analysis of the first initial assumption corollary, above, led to the theory that there might be a more specific initial engineering principle, based on the monitoring and planning parameters. For example, there is an intra-initial engineering principle within Op1 of:

$$\text{Monitor} : \ldots \text{Comfort} \xrightarrow[1,e\geq 1]{} \text{Plan} : \ldots \text{Comfort}$$
$$\Downarrow$$
$$\text{Monitor} : \ldots \text{Comfort} \xrightarrow[1,e\geq 1]{} \text{Plan} : \ldots \text{Comfort}$$

There is an inter-initial engineering principle between Op1 and Op2 for the more general form:

$$\text{Monitor} : \ldots \text{X} \xrightarrow[1,e\geq 1]{} \text{Plan} : \ldots \text{X}$$
$$\Downarrow$$
$$\text{Monitor} : \ldots \text{X} \xrightarrow[1,e\geq 1]{} \text{Plan} : \ldots \text{X}$$

Example 3—Planning takes longer overall, is more effort overall, but provides the benefits.

If the planning effort is considered to equate to the structural and behavioural costs in planning, then the operationalisations can be compared. In terms of the time, the actual time taken for planning could be used, but this time would be difficult to measure, so the event ticks are used instead, which is the same[11] as the behavioural costs.

A table notation is employed for the comparison of these costs:

Op1 Planning

Current structs		Actual structs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
82	1	63	1	-19	0

Current behs		Actual behs		Difference	
Abstract		Abstract	Physical	Abstract	Physical
66	1	50	1	-16	0

Op2 Planning

Current structs		Actual structs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
91	2	163	9	72	7

Current behs		Actual behs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
214	5	1267	86	1053	81

The headings show the operationalisation (Op1 or Op2), the types of costs—including whether for the current solution ('current') or the specific design solution ('actual'), and the difference of the costs. The figures are the costs or their differences. Further research could seek to combine the table notation and the equation notation developed in this research.

The analysis clearly shows the differences between the two operationalisations. It is not generally the case that planning takes more effort overall (nor takes longer), giving another counter-principle.

Example 4—Control effort is decreased and the benefits are provided

Following from Example 3 above, perhaps the control effort is more important in the prescription of solutions. A similar analysis to Example 3 can be performed by inspecting the control costs.

[11] A difference would be due to concurrent events, of which there are none in the planning parts of these operationalisations.

Op1 Control

Current structs		Actual structs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
51	26	38	29	-13	3

Current behs		Actual behs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
59	29	34	10	-25	-19

Op2 Control

Current structs		Actual structs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
346	99	390	108	44	9

Current behs		Actual behs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
496	108	590	159	94	51

Therefore, there is an inter-initial engineering principle in 'an increase in the physical structural costs'.

The human costs can be separated from the computer costs:

Op1 Control (H only)

Current structs		Actual structs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
35	7	24	6	-11	-1

Current behs		Actual behs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
41	11	24	7	-17	-4

Op2 Control

Current structs		Actual structs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
256	65	300	75	44	10

Current behs		Actual behs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
382	70	488	125	106	55

There are no generalities (except counter-principles).

Op1 Control (C only)

Current structs		Actual structs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
16	18.9	14	22.63	-2	3.73

Current behs		Actual behs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
18	18	10	3	-8	-15

Op2 Control (C only)

Current structs		Actual structs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
90	33.48	90	33.26	0	-0.22

Current behs		Actual behs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
114	38	102	34	-12	-4

Therefore, there is an inter-initial engineering principle of 'a reduction in the computer control costs'.

Initial Engineering Principles from General Guidelines

Strategy

Guidelines are design knowledge, and therefore might provide a suitable basis for initial engineering principles, particularly if they are claimed or demonstrated to work for craft practice.

It is expected that general guidelines, such as 'feedback' and 'consistency' (Smith and Mosier, 1986), can be identified within the operationalisation(s). The difficulty would be expected to be in delimiting and defining the investigated guideline, but any generality within the operationalisation(s) would support that delimiting and defining.

Example 1—Feedback

Feedback might be further described as the provision of a (relatively rapid) response by the computer after input by the human. This outcome can be represented as:

$$\mathrm{H:Input} \xrightarrow[1,t\leq 4s]{} \mathrm{C:Response}$$

$$\Downarrow$$

Where:

- $t \leq 4s$ means that the time between the input and the response should be less than 4 seconds.

In the Op2 specific design solution[12], there is an intra-initial engineering principle of:

$$\Downarrow$$
$$\text{A : FxP : Turn} \xrightarrow[1, t<1s]{} \text{C : O : Gas}$$

However, a more human factors (HF) view of feedback might be:

$$\Downarrow$$
$$\text{H : Input} \xrightarrow[1, t\leq 4s]{} \text{H : Encode response}$$

There is an intra-initial engineering principle in Op2 of:

$$\Downarrow$$
$$\text{A : FxP : Turn} \xrightarrow[1, t<1s]{} \text{A : Encode : Gas}$$

The relationship is also general over Op1, if H:Input is taken as H:FxP:Press or H:FxP:Turn, and H:Encode response is taken as H:Encode: LED off and H:Encode:Gas. There is an inter-initial engineering principle between Op1 and Op2 based on feedback of:

$$\Downarrow$$
$$\text{H : Input} \xrightarrow[1, t\leq 1s]{} \text{H : Encode response}$$

Example 2—Consistency

The possibility that there is consistency between the current and actual systems for goals with two standard following behaviours will be considered. This possibility might be represented as:

$$\text{H : FP : X} \xrightarrow[1, e=1]{} \text{Y} \xrightarrow[1, e=1]{} Z$$
$$\Downarrow$$
$$\text{H : FP : X} \xrightarrow[1, e=1]{} \text{Y} \xrightarrow[1, e=1]{} Z$$

---

[12] The time in the operationalisation tables does not show seconds, but the seconds were assessed independently.

This representation could be generalised for any number of standard following behaviours with an additional construct. However, this representation of consistency is violated in Op1 and Op2.

Initial Engineering Principles from MUSE Guidelines

Strategy

MUSE supports the development, during extant system analysis, and expression, during design, of design knowledge that is specific to the system. In part, these 'specific guidelines' are documented in the design recommendations and speculation columns of the MUSE tables. Initially, it is sensible to concentrate on specific guidelines that were used in the solution.

Example—Reduce later remembering

The following specific guidelines from the Cycles suggest possible generalisation, since improvement of planning might reduce later 'remembering':

> 'Avoid having A remember to turn the heating on or off if possible.' Cycle 1 MUSE application.
>
> 'Improvement in planning activities (particularly start time) should reduce flustering and therefore overheating.' Cycle 2 MUSE application.

'Reduction in remembering' might be understood as a reduction in the costs of planning behaviours that leads, or intends to lead, directly to control. In Op1, these behaviours are all of the planning behaviours except for A:FP:Plan. In Op2 current planning, they are all of the planning behaviours except for A:FP:Plan. In Op2 actual planning, they are all of the planning behaviours except for A:ShSubPlan and A:FP:Plan.

Op1 Planning

Current behs		Actual behs		Difference	
Abstract		Abstract	Physical	Abstract	Physical
64	1	48	1	-16	0

Op2 Planning

Current behs		Actual behs		Difference	
Abstract	Physical	Abstract	Physical	Abstract	Physical
212	5	662	16	450	11

The generality does not hold.

Initial Engineering Principles from MUSE Tasks

Strategy

The MUSE task diagrams also contain some of the above 'specific guidelines'. It is likely that the generalised products have more potential to be general 'specific guidelines', termed 'MUSE task guidelines'. The generalised products are the (x) and the (y) products.

To support intra-initial engineering principles, generalisation over the MUSE (x) and (y) products for the design for an operationalisation would be useful, to produce products that might be termed (xy) products. Included in these products would be selection constructs that indicate a task change from the (x) situation to the (y) situation. It is suggested that the selection entries are marked with (x) or (y), to support the direction of design operator (⇓).

To support inter-initial engineering principles, generalisation over these (xy) products would be useful, to produce products that might be termed (xy^n) products.

Example—Cycle 1 CTM(xy)

Figure 34 shows an (xy) product between the GTM(x) and CTM(y) products in the Cycle 1 MUSE application.

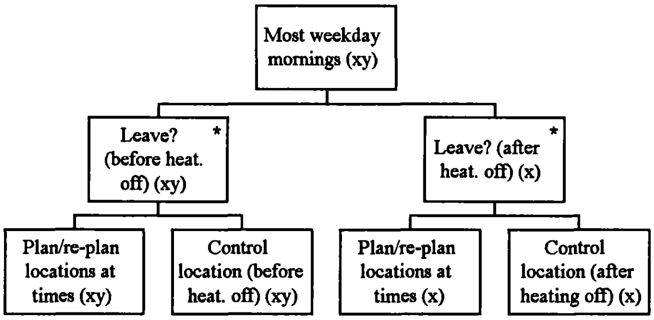


Figure 34. Cycle 1 CTM(xy) Product.

It is unlikely that the generality will hold over Op1, because Op1 does not cover a current design situation that includes leaving before the heating goes off. However, it does show an (xy) product. This example shows that generalisation over design situations with the same current artefact, user requirements, and artefact would be possible. (Analogous to MUSE TD analyses.)

The chapter details the strategy for acquiring initial engineering principles. Examples of initial engineering principles from the operationalisations are reported for each of the detailed strategies. The concept of 'counter-principle' is introduced, as an initial engineering principle that is not general between operationalisations.

Consideration of the status of the acquired initial engineering principles and the strategy assessment follows in the Strategy Assessment and Discussion chapter.

13. Strategy Assessment and Discussion

This chapter assesses and discusses the research strategy for developing engineering principles (introduced in Chapter 4 and developed in Chapter 12) following the acquisition of initial engineering principles, described in the previous chapter. The steps required to acquire engineering principles from initial engineering principles are discussed.

The assessment of the strategy rests on the status of the acquired initial engineering principles.

Strategy and Conception Changes

The research strategy was described as a bottom-up strategy. It might be claimed that the strategy used is closer to the top-down strategy:

- The architecture conceptions and the planning and control conception directly influence the content of the initial engineering principles.
- The detailed strategy based the identification of initial engineering principles on craft substantive design knowledge.

There remains a contrast with a top-down strategy, however. Stork et al. (1998) and Lambie et al (1998) describe a project that attempts the top-down strategy. They start with an informal statement of craft substantive design knowledge and then attempt to operationalise it as an initial engineering principle. Accordingly, Stork and Long (1998) propose that there might be a continuum of strategies between the bottom-up and the top-down strategies, along a continuum of the expected initial generality[13]. Therefore, the research strategy used is closer to the top-down strategy than originally anticipated, although it can still be distinguished from the top-down strategy. Application of the alternative strategies outlined would be beneficial to confirm strategy selection.

---

[13] The initial expected generality in this research is the general design problem conception, the architecture conceptions, and the planning and control conception; in order of decreasing expected generality.

Identifying initial expected generality in the strategy raises a concern for the conception of engineering principles. If the specific design problem and its solution conception contains concepts that relate to engineering principle acquisition, then are they required for engineering practice? Possibly yes, although a specific design problem and its solution conception might need to encompass alternative such general conceptions to match the partial design problem and solution operationalisations (potentially to be applied). However, it seems more likely that it will be possible to operationalise partial design problems and solutions directly from the user requirements[14]. A similar strategy that operationalised initial expected generality—the Hill et al. (1995) model of the planning and control of multiple tasks—for design is described in Stork et al. (in preparation) and briefly in the following chapter.

Status of Initial Engineering Principles

Initial engineering principles have been acquired. These initial engineering principles have the pre-requisites for acquiring potential guarantee:

- They are conceptualised according to a conception of the discipline of HCI.
- They are operationalisations of conceptions based on that conception.
- They are generalised over or within the two cycles.
- They are tested by successful evaluations of the two cycles[15].

The generality is a remaining concern for the initial engineering principles. In particular, two or fewer cycles can be considered poor generality, indicated by the difficulty of selecting appropriate general cycle types and commonalities. A further concern is that the expression of the initial engineering principles might not be appropriate for application. These concerns indicate that the initial engineering principles should be considered 'early'.

---

[14] This situation is analogous to alternative representations available in Software Engineering. For example, a SE design may be specified using both data flow diagrams and entity relationship diagrams.

[15] In the case of Cycle 2, *A* had the same problem as *D* and it was solved by use of the prototype artefact. The evaluations of *D* are considered to support the case for *A*.

Strategy Assessment

Early initial engineering principles have been acquired. The strategy can be considered successful at this stage. Further cycles and validation are the next steps for assessing the strategy.

Further Research

Further Early Initial Engineering Principles

The research offers examples of early initial engineering principles. Further early initial engineering principles can be identified from the products of this research.

Further Cycles

As noted above, further cycles are required to move from early initial engineering principles to initial engineering principles. More complex design situations could be addressed.

It was intended to operationalise a third cycle for this research. A third set of user requirements were selected that were more complex than the Cycle 1 and Cycle 2 user requirements. An artefact specification was developed, prototyped, and evaluated positively. An explicit operationalisation was developed. Unfortunately, time constraints prevented formal and metricated operationalisation during this research. However, these products are available for further research.

Selection of the user requirements for these cycles is important. The selected cycle user requirements for this research did not support potential generalisation as well as expected. In particular, the type of planning and control for each cycle artefact was different. The Cycle 1 artefact attempted to minimise re-planning, whereas the Cycle 2 artefact attempted to maximise pre-planning. A possible improvement to the strategy might be to have a more rapid design phase before selection, perhaps encompassing:

- The MUSE Information Elicitation and Analysis Phase, analysing the current extant system to the TD(Current) and GTM(Current) products.
- The MUSE Design Synthesis Phase to the CTM(y) product.

Validation of Initial Engineering Principles

Validation of the initial engineering principles involves re-expression as required and testing by application to a design situation. Methodological engineering principles are required for application. The guarantee of engineering principles validated by application would be based on:

- The initial engineering principle guarantee.
- The operationalisations.
- The (known) generality.

Testing is a challenge, however, since the effect of a particular engineering principle needs to be identified. The alternatives appear to be:

- To control the designs to include or exclude the engineering principle application.
- To 'trace' the engineering principle application and its contribution to effectiveness. Simulation may support this tracing.

Metrification of the guarantee of engineering principles could be considered at this stage.

Method and Tool Support

This research has highlighted a requirement for methodological and tool support for the research strategy. The strategy products could have been integrated with a method (see next chapter). Tool support could usefully support:

- MUSE application. (A diagram editor was used.)
- Operationalisation.
- Detailed strategy: the identification of relationships. (For example, a tool could have supported the extension of 'consistency' to e<=n and any number of following behaviours.)
- Initial engineering principle validation.

Strategy Discussion

Current, Problem, and Solution

The specific design problem conception was found to require the current solution conception.

The operationalisation of these conceptions required several iterations. In particular, the specific design problem domain operationalisation requires iteration. The first version was derived from the current solution domain operationalisation. However, the second version was derived also from the specific design solution domain operationalisation, which is based on the craft design. The craft design might imply different goals from that initially expected.

Ergonomics Discipline

The operationalisations might be considered closer to the discipline of Ergonomics rather than that of HCI. However, Long (1995) relates the two disciplines:

> 'To a first approximation, Ergonomics (and Human Factors) can be considered the discipline of Human-Machine Interaction (HMI). HMI can be assumed to include Human-Computer Interaction (HCI), if computers are conceived as a sub-set of machines.'

Since the research emphasis is the acquisition of HCI engineering principles, then the research is still relevant for the HCI discipline.

Unitary HCI Discipline

The early initial engineering principles subsume both HF concerns and SE concerns, supporting a suggestion by Long (1995) that there may be 'a unitary discipline of HCI (rather than HF & SE) in the longer term'.

Craft and Scientific Knowledge

The strategy provides a means of potentially incorporating craft and (applied) ccientific knowledge into initial engineering principles. Craft knowledge has been incorporated by (Super-)Craft design. Scientific knowledge has been incorporated through the conceptions (for example, the cognitive architecture conception) and the operationalisation (for example, the formulae).

Dowell (1993; after Long, 1986) claimed that engineering 'principles could be validated through utility or through explanation by science'. (Scientific disciplines have a general science problem of the explanation and prediction of phenomena.) It seems likely that the potential guarantee of the early initial engineering principles is improved by their craft and scientific knowledge basis. However, the status of the craft and scientific knowledge for design is not known, therefore determining that improvement is not possible.

Further research is required into the relationship of engineering knowledge with craft and scientific knowledge.

Formal Methods

The early initial engineering principles are formal in that they are operationalised to the level of metrics. These metrics enable mathematical techniques to be employed.

> 'Expressed as simply as possible, the goal of Formal Methods is to base the software development process squarely upon a workable set of mathematical techniques.' (Gerhart, 1991)

Formal methods is an approach that is being followed by SE (Hoare, 1969; Gerhart, 1990) to solve the SE 'software crisis' (Pressman, 1982). There has been significant analysis of the issues surrounding formal methods in SE (for example, Fetzer, 1988; Cantwell-Smith, 1985) and some in HF (Stork, 1992; Bauer, 1995). Stork (1992) identifies that their use in HF is restricted mainly to the formal description of interfaces (for example, Alexander, 1985 and 1987; Anderson, 1987). There are several concerns about formal methods with implications for their use in informal engineering principles: the Fetzer 'gap', the complexity of systems; the purposiveness of animates; and the executability of specifications.

Fetzer (1988), argues against formal verification[16]. He accepts that there may be a formal path between requirements specification and a solution specification, but claims that there will never be an unbroken path from the formal solution specification to a solution implementation. The initial engineering principles distinguish between a general design solution and its artefact specification.

---

[16] Boehm (1981) describes the difference between validation and verification in SE as: Validation—'Are we building the right product?'; and Verification—'Are we building the product right?'.

Cantwell-Smith (1985) claims that the increasing complexity of computer programs will prevent them from being proven correct. The analogy here might be that engineering principles might be too complex to be formally applicable. However, the initial engineering principles are not complex. Also, in high risk situations (for example, safety critical systems) complexity may deliver valuable results and so be worth significant effort in formal application.

Becker (1975) underlines the value of mathematics: 'Mathematics arises from man's attempt to find concepts that allow a wide range of phenomena to be described in similar terms, and thereby understood in a coherent manner.' He claims that mathematics cannot be used for the phenomena of behaviours: 'It is quite possible that animate behavioural systems are organized in ways (e.g., ultra-high parallelism) that are not compatible with our traditional habits of induction and part-whole analysis.'; and 'A "behavior" is not a well-defined thing like a numerical quantity; rather it is a selective account of an event that is defined only by certain decisions on the part of us, its describers.' However, initial engineering principles are not concerned with a scientifically-correct description of the behaviour, only the utility of the initial engineering principle.

Hoare (1989) dismisses the requirement for formal specifications to be executable: 'To require a specification or design of a program to be executable is hardly less absurd than requiring the specification of a building to be habitable or the blue-prints of a car to be driveable.' However, tool support may be required to apply engineering principles. Further, the formalism may need to be altered to enable tool support (Breuer and Bowen, 1994).

The main concern for a formalism 'is whether it is useful, and I would add usable to that'. (Stork, 1992; following Milner, 1989, and Dix, 1991). The known guarantee of engineering principles ensures usefulness. Usability of engineering principles relates to the requirement for that known guarantee (e.g. safety critical systems may be worth significant effort).

The initial engineering principles are considered to have the pre-requisites for acquiring potential guarantee. However, concerns are raised over their generality and their expression for application, leading to them being termed 'early' initial engineering principles. Given the 'early' status of the initial engineering principles, the research strategy is assessed as successful.

14. MUSE for Research (MUSE/R)

A MUSE application involves the construction of products that have a well-defined scope, process, and notation, for example, the MUSE applications of Chapters 8 and 10. Similarly, the application of the research strategy (Chapters 4 and 12) has involved the construction of products that have a well-defined scope and notation (Chapters 9 and 11). The scope and notations of the MUSE products and the research products can be related to each other, to propose a version of MUSE to support research similar to the present research. This chapter proposes and outlines such a version of MUSE, termed MUSE for Research (MUSE/R), that has been developed for this research.

Scope and Notation

The research strategy has four main products: current solution operationalisations; specific design problem operationalisations; specific design solution operationalisations; and initial engineering principles. The scope of the products is expressed in the product name. The notations for the current and specific design solution are: domain, structure, and behaviour diagrams; and quality, structure changes, and behaviour formulae values. The notations for the specific design problem and initial engineering principles are logical/mathematical expressions.

MUSE has potential for supporting the research strategy because:

- Most of the products are operationalisations of specific design situations: either systems being analysed or the system being designed.

- The products are explicit with a well-defined and explicit scope.

The scope, processes, and notations of the MUSE products need to be enhanced to accomodate the research conceptions and to support the research strategy.

MUSE has three phases: the information elicitation and analysis (IEA) phase; the conceptual design (CD) phase; and the detailed design (DD) phase. The first stages of the IEA phase involve the analysis of extant systems: including operationalisations of the 'tasks' and 'domains of discourse' of the systems; and generalisations of the tasks. For MUSE/R, these stages are re-scoped to operationalise: the specific structures and behaviours of the extant worksystems; their structural and behavioural costs; their domain; the quality of their work; and appropriate generalisations of these operationalisations. Thus, the first stages of the IEA phase operationalise current solutions according to the conceptions for this research.

The final stages of the IEA phase and the first stages of the CD phase in MUSE involve the specification of a human factors statement of the user needs. For MUSE/R, these stages are re-scoped to operationalise the specific design problem. The final stages of the CD phase and the DD phase in MUSE involve the specification of the interaction artefact and the documentation of the design rationale. For MUSE/R, these stages are re-scoped to operationalise the specific design solution and the previously-acquired HF knowledge applied to develop the specific design solution.

The notations for the re-scoping are those employed by this research strategy, which are similar or additional to the MUSE notations no longer employed.

HCI design knowledge, including MUSE/R, also has potential for application to more general research strategies, including validation of HCI knowledge. Stork and Long (1997) outline the more general case for HCI research and development.

Process

The redefinition of scope of MUSE/R suggests the process changes. Figure 35 shows the overview of the processes of MUSE/R. The redefinition demonstrates potential for reduction in research effort relative to this research.

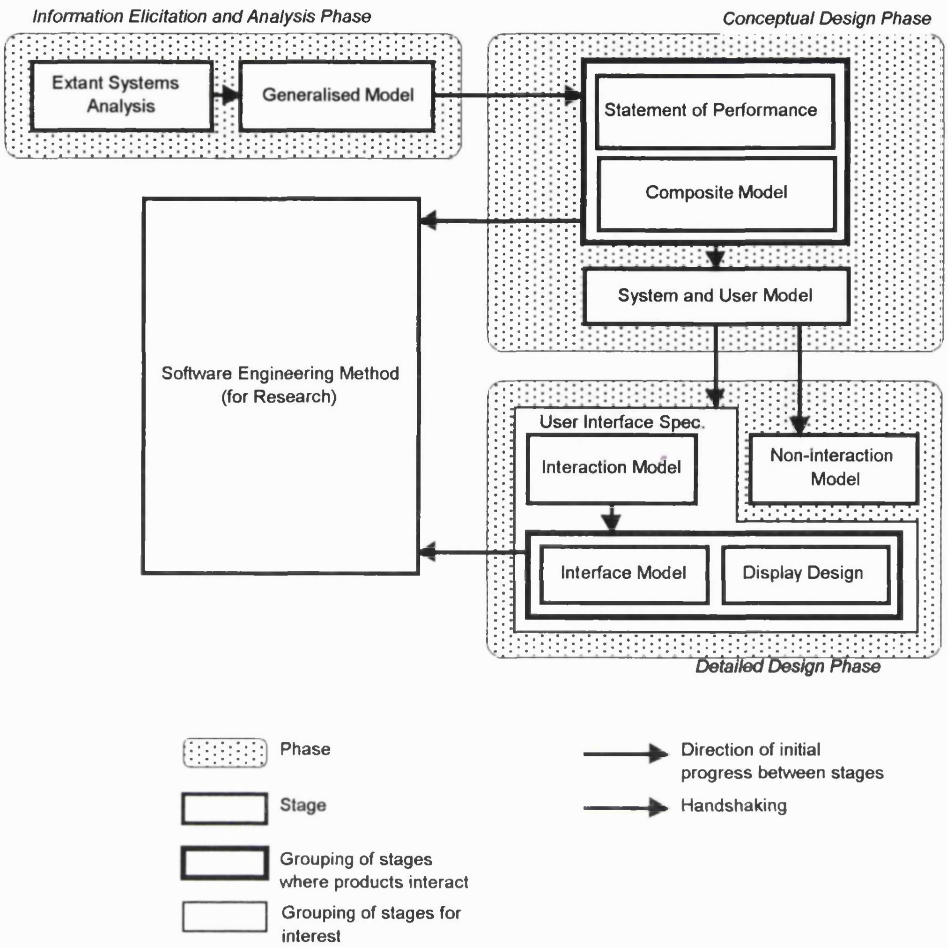


Figure 35. MUSE/R Overview.

Support for Design

MUSE/R needs to support HCI practice at least as well as MUSE. The accomodation of the research conceptions suggests that the MUSE/R products will be more complete and coherent (relative to the general design problem conception of HCI) than the MUSE products, suggesting improved support for design. (See earlier claims for non-engineering applications of the D&L general design problem conception.)

Görner (1994) claims that 'a significant relationship [exists] between the amount of goal characteristics analysed and the quality of resulting design solution'. MUSE/R emphasises the operationalisation of task quality relative to MUSE, so potentially increasing 'the amount of goal characteristics analysed' (Görner, 1994). Further, Gomaa et al. (1992) claim that domain model analysis supports design (including different views of the domain, as implied by the specific design conceptions of this research). MUSE/R emphasises domain analysis relative to MUSE. Chung et al.

(1995) employ similar diagrams to those proposed for MUSE/R for design. Ege and Stary (1992) successfully employ a combined task and object approach to designing interfaces, being similar to the behaviours and domain operationalisations delivered by MUSE/R.

MUSE/R has been applied to the design of an in-car newspaper-information device (Stork et al., in preparation). The scope, process, and notation of the products were defined further and MUSE/R applied informally. A different research strategy was employed: the Hill et al. (1995) model of planning and control of multiple tasks was operationalised for the products, to attempt informal validation of the model for design. A successful design was produced.

Further Research

Although an initial application of MUSE/R has been conducted, the scope, process, and notation of the products need to be defined further to support application. Further research and design case-studies need to be performed to assess the support MUSE/R offers for research and design. The apparently strong relationship between research and design in HCI also needs further investigation.

The evaluation component of the strategy could be incorporated into the MUSE/R method (and remain consistent with design, Hacker, 1997). Tool support designed for MUSE/R is essential, since both the research and design parts of this research would have been significantly assisted by such support. Automatic generation of prototype interfaces would be advantageous (for example, the DIANE method provides automatic help generation, Barthet, 1995).

A version of MUSE developed for this research, termed MUSE for Research (MUSE/R), is proposed to support HCI research similar to the present research.

15. Conclusions

Research is the acquisition of knowledge to support a purpose. This research has acquired: a conception of engineering principles, a strategy for the acquisition of engineering principles and examples of early initial engineering principles. A positive assessment of the strategy to this juncture has been made. Further research towards the acquisition of engineering principles has been outlined.

This research has made significant progress towards solving the operational problem of the inability to design human-computer systems effectively. Recognising the targets of the research has been an important aim in appropriately focussing the research and acquiring knowledge towards solving this operational problem.

In addition, shorter-term benefits can be identified.

Early Initial Engineering Principles

Technical Solution

This research has acquired examples of early initial engineering principles to support the effective design of human-computer systems. The acquisition of these early initial engineering principles supports the strategy developed for this research and takes HCI substantially *towards* engineering principles.

The status of the early initial engineering principles acquired by this research and further research towards engineering principles are discussed in Chapter 13.

Operational Solution

Taking HCI towards engineering principles offers the potential for more effective HCI practice, as required to solve the operational problem, which was the inability to design effectively human-computer systems.

Strategy

The explicit strategy and implementation for the research, and its discussion, provides a basis for other HCI researchers to consider their acquisition of HCI knowledge. The MUSE/R method provides a basis for other researchers and designers to apply the strategy.

Informal Assessment of Dowell and Long

The D&L characterisation of the discipline of HCI as an engineering discipline and their conception of the general design problem of HCI have enabled the research to acquire early initial engineering principles. The research informally supports D&L's characterisation and conception.

Further Research

Several next steps towards engineering principles are identified in Chapter 13. They are, in brief:

- Identification of further early initial engineering principles from the products of this research.
- Acquisition of initial engineering principles from further cycles.
- Acquisition of methodological principles.
- Application of alternative strategies, to assess further the merits of different strategies.
- Validation of initial engineering principles.
- Further method support for the strategy and validation.
- Tool support for the strategy and validation.

Shorter-Term Research Benefits

Several shorter-term research goals have been met by the research.

A version of MUSE has been developed, MUSE/R, that supports more complete, coherent, and consistent specification of the design problem and solution. It is expected that MUSE/R has the potential to improve HCI practice in the medium- and longer-terms.

Conceptions and operationalisations of design problems and solutions from this research are available to assist practitioners in the short term to better identify their design problems and solutions, and to better assess whether their solutions solve their design problems.

A short-term goal of MUSE assessment has been achieved (Stork et al., 1995). MUSE applications have been conducted that can be used to support MUSE training and application. Two example artefacts to HCI user requirements have been developed. Guidelines from the development of the artefacts have been acquired.

References

Alexander, H. (1985). *Two Techniques for Specifying and Prototyping Interactive Software*. University of Stirling internal report.

Alexander, H. (1987). *Executable Specifications as an Aid to Dialogue Design.* Formal Methods and HCI 1987. IEE Computing & Control Division Colloquium.

Alterman, R. (1988). *Adaptive Planning.* Cognitive Science 12, pp. 393-421.

Anderson, S. (1987). *Analysis and Synthesis in HCI.* Formal Methods and HCI 1987. IEE Computing & Control Division Colloquium.

Baecker, R. and Buxton, W. (1987). *Readings in Human Computer Interaction: A Multidisciplinary Approach.* Morgan Kaufmann Publishers.

Barnard, P., Wilson, M., and Maclean, A. (1988). *Approximate Modelling of Cognitive Activity with an Expert System: A Theory-Based Strategy for Developing an Interactive Design Tool.* The Computer Journal. Vol. 31. No. 5. pp 445-456.

Barr, A., Cohen, P., and Feigenbaum, E. (1989). *Handbook of Artificial Intelligence*. Addison-Wesley.

Barthet, M-F. (1995). *L'integration de l'ergonomie aux methodes informatiques de conception de logiciels*. Proc. IHM'95. Toulouse. Cépaduès-Editions.

Bauer, B. (1995). *Proving the Correctness of Formal User Interface Specifications.* Design, Specification and Verification of Interactive Systems '95. P. Palanque and R. Bastide (eds.). Springer. pp. 224-241.

Becker, J. (1975). *Reflections on the Formal Description of Behavior.* Representation and Understanding. Bobrow and Collins (eds.). Academic Press. pp. 83-102.

Bellotti, V. (1988). *Implications of Current Design Practice for the use of HCI Techniques.* People and Computers IV: Proceedings of HCI'88. D. Jones and R. Winder (eds.). Cambridge University Press. pp. 13-34.

Benator, B. I. (1987). *Microcomputer applications in energy management.* Journal of Microcomputer Applications. 10. pp. 1-9.

Bowers, J., (1992). *The Politics of Formalism*. Context of Computer-Mediated Communication. M. Lea (ed.). Hassocks: Harvester.

Breuer, P. and Bowen, J. (1994). *Towards Correct Executable Semantics for Z.* Proc. 8th Z Users Workshop. J. Bowen and J. Hall (eds.). Springer-Verlag.

Brinkworth, B. (1993). *The great energy crisis—could it have been averted?* Energy World. September 1993. pp. i-viii.

Cantwell-Smith, B. (1985). *The Limits of Correctness.* Paper prepared for the Symposium on Unintentional Nuclear War, Fifth Congress of the International Physicians for the Prevention of Nuclear War, Budapest, Hungary.

Capehart, B. (1986). *Emerging Microcomputer Technology for Electrical Energy Management*. Journal of Microcomputer Applications. 9. pp. 253-260.

Carroll, J. (1989). *Artifacts as Psychological Theories: the Case of Human-Computer Interaction*. Behaviour and Information Technology. Vol. 8. No. 4. pp. 247-256.

Carroll, J. (1997). *Human-Computer Interaction: Psychology as a Science of Design*. Int. J. Human-Computer Studies. 46. pp. 501-522.

Carroll, J., Kellog, W., and Rosson M.B. (1991). *The Task-Artifact Cycle*. Designing Interaction: Psychology at the Human-Computer Interface. J. Carroll (ed.). Cambridge University Press. pp. 74-102.

Carroll, J., Singley, M., and Rosson, M. B. (1992). *Integrating Theory Development with Design Evaluation*. Vol. 11. No. 5. pp. 247-255.

Checkland, P. (1981). *System Thinking, Systems Practice*. John Wiley & Sons.

Chung, L., Nixon, B., and Yu, E. (1995). *Using Non-Functional Requirements to Systematically Support Change*. Proc. of the 2nd IEEE International Symposium on Requirements Engineering. pp. 132-139.

Colbert, M. (1994a). *Carry Forward in the Development of Military Planning Systems*. PhD thesis. University College London.

Colbert, M. (1994b). *Towards Generic User Interface Components: A Partial Menu Structure for Planning Systems*. Internal Ergonomics and HCI Unit paper.

Colbert, M. and Long, J. (1996). *Towards the Development of Classes of Interaction: Initial Illustration with Reference to Off-Load Planning*. Behaviour & Information Technology. Vol. 15. No. 3. pp. 149-181.

Colbert, M., Long, J., and Dowell, J. (1995). *Integrating Human Factors with Software Engineering: An Illustration with Reference to a Military Planning System*. Proc. Pan Pacific Conference on Information Systems (PACIS). National University of Singapore. pp. 357-364.

Constant, D. (1993). *The Productivity Paradox; Why Hasn't Information Technology Fulfilled Its Promise?* A Panel Report from CSCW'92. SIGCHI Bulletin. October 1993. Vol. 25. No. 4. pp. 42-44.

Davis, R., and Buchanan, B. (1977). *Meta-Level Knowledge: Overview and Applications*. Proc. 5th IJCAI. pp. 920-927.

De Souza, F., Long, J., and Bevan, N. (1990). *Types of Error and Difficulty in using Human-Factors Guidelines: The Case of Interface Menu Design*. Contemporary Ergonomics 1990. E. Lovesey (ed.). Taylor and Francis. pp. 340-346.

Diaper, D. (1989). *The Discipline of HCI*. Interacting with Computers. Vol. 1. No. 1. pp. 3-5.

Dix, A. (1991). *Formal Methods for Interactive Systems*. Academic Press. Cambridge.

Dowell, J. (1993). *Cognitive Engineering and the Rationalisation of the Flight Strip*. PhD thesis. University College London.

Dowell, J. and Long, J. (1989). *Towards a conception for an engineering discipline of human factors*. Ergonomics. 32(11). pp. 1513-1535.

Dowie, J. and Elstein, A (1988) (eds.). *Professional judgement: a reader in clinical decision making*. Cambridge University Press. 1988.

Draper, S. (1991). *User Interface Performance Measurement*. JCI Summer School August 1991.

Ege, R., and Stary, C. (1992). *Designing Maintainable, Reusable Interfaces*. IEEE Software. November 1992. pp. 24-32.

Energy (1990). *Energy Facts and Figures.* Agence Française Pour la Maîtrise de L'Energie.

Engelmore, R., Morgan, A., and Nii, H. (1988). *Introduction.* Blackboard Systems. R. Engelmore and A. Morgan (eds.). Addison-Wesley.

Fetzer, J. (1988). *Program Verification: The Very Idea.* Communications of the ACM. Vol. 31. No. 9. pp. 1048-1063.

Firby, R. (1987). *An Investigation into Reactive Planning in Complex Domains.* AAAI 87. pp. 202-206.

French, M. (1990). *Function Costing: A Potential Aid to Designers.* Journal of Engineering Design. Vol. 1. No 1. pp. 47-53.

Friend, J. and Jessop, W. (1969). *Local Government and Strategic Choice: an Operational Research Approach to the Processes of Public Planning.* Tavistock Publications.

Gaines, B. and Shaw, M. (1986a). *From Timesharing to the Sixth Generation: the Development of Human-Computer Interaction. Part 1.* International Journal of Man-Machine Studies. 24. pp. 1-27.

Gaines, B. and Shaw, M. (1986b). *Foundations of Dialog Engineering: the Development of Human-Computer Interaction. Part II.* International Journal of Man-Machine Studies. 25. pp. 101-112.

Gardner, M. and Christie, B. (1987). *Applying Cognitive Psychology to User Interface Design.* John Wiley & Sons.

Gerhart, S. (1990). *Applications of Formal methods: Developing Virtuoso Software.* IEEE Software. September 1990. pp. 7-10.

Gerhart, S. (1991). *Formal Methods: An International Perspective.* 13 Int. Conf. on Software Engineering. Austin Texas. pp. 36-37.

Gomaa, H., Kerschberg, L., and Sugumaran, V. (1992). *A Knowledge-Based Approach to Generating Target System Specifications from a Domain Model.* Information Processing 92. Vol. 1. pp. 252-258.

Goodwin, B. (1995). *Reading the Future for Meters.* Profesional Engineering. 22 March 1995. pp. 24-25.

Görner, R. (1994). *Zur Psychologischen Ayalyse von Konstrukteur- und Entwurfstätigkeiten. Die Handlungsregulationstheorie.* Von der Praxis einer Theorie. B. Bergmann and P. Richter (eds.). Hogrefe. pp. 233-241.

Hacker, W. (1997). *Improving Engineering Design—Contributions of Cognitive Ergonomics.* Ergonomics. Vol. 40. No. 10. pp. 1088-1096.

Halé, J. (1986). *Goal oriented decision support.* Computer Assisted Decision Making. G. Mitra (ed.). Elsevier Science Publishers. pp. 35-48.

Hayes-Roth, B., Hayes-Roth, F., Rosenschein, S., and Cammarata, S. (1988). *Modelling Planning as an Incremental, Opportunistic Process.* Blackboard Systems. R. Engelmore and A. Morgan (eds.). Addison-Wesley. pp. 231-583.

Hill, B., Long, J., Smith, W., and Whitefield, A. (1995). *A Model of Medical Reception—the Planning and Control of Multiple Task Work.* Applied Cognitive Psychology. Vol. 9. pp. S81-S114.

Hoare, C. (1969). *An Axiomatic Basis for Computer Programming.* Communications of the ACM. Vol. 12. No 10. pp. 576-583.

Hoare, C. (1989). *Formal Methods in Computer System Design.* Computer Physics Communications 57. pp. 206-210.

Holland, J., Holyoak, K., Nisbett, R., and Thagard, P. (1987). *Induction, Processes of Inference, Learning and Discovery.* MIT Press.

Jenkins, V., Lim. K. Y., Long, J. (1993). *A Classification of Domestic Tasks to Support Device Procurement and Design.* Proc. Ergonomics Society's Annual Conference, Edinburgh. E. Lovesey (ed.). Taylor and Francis. pp. 123-128.

Johnson, C. (1995). *The Economics of Interface Development.* Proc. of Interact '95. pp. 19-25.

Karat, J. (1988). *Software Evaluation Methodologies.* Handbook of HCI. Elsevier Science Publishers. pp. 891-903.

Katzenstein, W. (1993). *Information Technology and Productivity in the World Bank.* Planning and Budgeting Working Paper. World Bank. Washington DC.

Keen, P. (1987). *Decision Support Systems: The Next Decade.* Decision Support Systems. 3. pp. 253-265.

Lambie T., Stork A., and Long, J. (1998). *The Coordination Mechanism and Cooperative Work.* Proc. of the Ninth European Conference on Cognitive Ergonomics. T. Green, L. Bannon, C. Warren, and J. Buckley (eds.). pp. 163-166.

Landauer, T. (1995). *The Trouble with Computers.* MIT Press.

LAS (1993). *Report on the Inquiry into the London Ambulance Service.* S. W. Thames Regional Health Authority, 40 Eastbourne Terrace, London W2 3QR.

Lim, K. Y., and Long, J. (1994). *The MUSE Method for Usability Engineering.* Cambridge University Press.

Linney, J. (1991). *Review of Reactive Planning Literature.* QMW Dept. of Computer Science Technical Report No. 560.

Long J. & Dowell J. (1989). *Conceptions for the discipline of HCI: Craft, Applied Science and Engineering.* Proceedings of the Fifth Conference of the BCS HCI SIG. A. Sutcliffe and L. Macaulay (eds.). Cambridge University Press. pp. 9-32.

Long, J. (1995). *Specifying and Solving Ergonomic Design Problems.* IEA World Conference, Brazil.

Mayhew, D. (1992). *Principles and Guidelines in Software User Interface Design.* Prentice Hall.

McDonald, J. and Schvaneveldt, R. (1988). *The Application of User Knowledge to Interface Design.* Cognitive Science and its Applications for Human-Computer Interaction. R. Guindon (ed.). Lawrence Erlbaum. pp. 289-338.

Milner, R. (1989). *Communication and Concurrency.* Prentice-Hall.

Neale J. and Liebert, R. (1980). *External Validity.* Chapter 12, Science and Behaviour—An Introduction to Methods of Research. Prentice-Hall. pp. 254-274.

Newman, W. (1994). *A Preliminary Analysis of the Products of HCI Research, Using Pro Forma Abstracts.* CHI'94. pp. 278-284.

Newman, W., Eldridge, M., and Harper, R. (1996). *Modelling Last-Minute Authoring: Does Technology Add Value or Encourage Tinkering?* CHI'96 Short Papers. pp. 221-222.

Norman, D. (1989). *The Psychology of Everyday Things.* Basic Books.

Phillips, T. (1994). *Welcome to the Computerised Home*. The Guardian Computer Section. March 10, 1994. p. 19.

Pietras, C. and Coury, G. (1994). *The development of cognitive models of planning for use in the design of project management systems*. Int. J. Human-Computer Studies 40. pp. 5-30.

Pirsig, R. (1974). *Zen and the Art of Motorcycle Maintenance*. Bodley Head.

Pressman, R. S. (1982). *Software Engineering: a practitioners approach*. McGraw-Hill.

Rahman, S. and Bhatnagar, R. (1986). *Computerised energy management systems—why and how*. Journal of Microcomputer Applications 9. pp. 261-270.

Robertson, S. and Black, J. (1986). *Structure and Development of Plans in Computer Text Editing*. Human-Computer Interaction. Vol. 2. pp. 201-226.

Rouse, W. and Greenstein, J. (1976). *A model of human decision making in multi-task situations: implications for computer aiding*. Proc. of IEEE Int. Conf. on Cybernetics and Society. p. 425-433.

Shneiderman, B. (1992). *Designing the User Interface: Strategies of Effective Human-Computer Interaction*. Second edition. Addison-Wesley Publishing Company.

Smith, S. and Mosier, J. (1986). *Guidelines for Designing User Interface Software*. The MITRE Corporation.

Stork, A. (1992). *A Formal Description of Worksystem Behaviours and Interactions*. MSc (Ergonomics) Thesis. University of London.

Stork, A. and Long, J. (1994). *A Planning and Control Design Problem in the Home: Rationale and a Case Study*. Proc. International Working Conference on Home-Oriented Informatics, Telematics and Automation. K. Bjerg and K. Borreby (eds.). University of Copenhagen. pp. 419-428.

Stork, A. and Long, J. (1997). *Structured Methods for Human Factors Research and Development*. Proc. HCI International '97. G. Salvendy, M. Smith, and R. Koubek (eds.). Elsevier Science. Vol. 2. pp. 531-534.

Stork, A., and Long, J. (1998). *Strategies for Developing Substantive Engineering Principles*. HCI '98 Conference Companion. J. May, J. Siddiqi, and J. Wilkinson (eds.). pp. 36-37.

Stork, A., Hill, B., and Life, A. (in preparation). *Engineering Research and Development (In-Car Newspaper Information)*.

Stork, A., Lambie, T., and Long, J. (1998). *Cognitive Engineering Coordination in Emergency Management Training*. HCI '98 Conference Companion. J. May, J. Siddiqi, and J. Wilkinson (eds.). pp. 36-37.

Storrs, G. (1989). *A Conceptual Model of Human-Computer Interaction?* Behaviour and Information Technology. Vol. 8. No. 5. pp. 323-334.

Surunjan, D. (1986). *Providing Effective Decision Support: Modelling Users and Their Requirements*. Decision Support Systems. 2. pp. 309-319.

Thimbleby, H. (1991). *Can Humans Think?* The Ergonomics Society Lecture 1991. Ergonomics. Vol. 34. No 10. pp. 1269-1287.

Tighe, I. (1996). *All Systems Go!* Computer Bulletin. September 1996. pp. 20-21.

Timmer, P. (1999). *Expression of Operator Planning Horizons: A Cognitive Engineering Approach*. PhD thesis. University College London.

Timmer, P. and Long, J. (1996). *Integrating Domain and Worksystem Models: an Illustration from Air Traffic Management*. Domain Knowledge for Interactive System Design. A. Sutcliffe, D. Benyon, and F. van Assche (eds.). Chapman & Hall. pp. 194-207.

Timmer, P. and Long, J. (1997). *Separating User Knowledge of Domain and Device: A Framework*. People and Computers XII. Proceedings of HCI'97. H. Thimbleby, B. O'Conaill, and P. Thomas (eds.). Springer.

Appendix A. The OpEnMan Heating Control Interface

A description and evaluation of the interface for heating control offered by the OpEnMan controller.

The OpEnMan Controller

The OpEnMan controller has been developed by an Esprit project (7061) consortium which includes the sponsors of this research, Schlumberger Industries, as well as Electricité de France (EdF), Iberdrola—the Spanish electricity providers—, and KEON—a software house.

The OpEnMan controller is designed to work with tariff systems in France and Spain. It aims to control heating, water-heating, and other electrical appliances. The result is a complicated situation and device, so this appendix concentrates on the French situation, and not Spain, and the OpEnMan heating control interface. The French situation is selected because the sponsors' consider that the French situation, the tariff and other factors, is more complex, so more likely to be problematic, than the Spanish situation. The heating control interface is selected because one of the primary concerns of this research is energy management.

Relevant Electricity Tariff in France

The OpEnMan controller is designed to work with a recently introduced tariff in France called 'Tempo.' Each day has a different tariff type designated by a colour: blue, white, or red. Each of these tariff types has an on-peak and an off-peak period, the times of which depends on the region. The periods for an example region are: off-peak period for blue and white days from midnight to 6am and from 10pm to midnight; off-peak period for red days from midnight to 6am; and on-peak periods at all other times.

The blue tariff offers cheap electricity for both off-peak and on-peak, the white tariff is about 2.5 times more expensive than the blue, but still reasonably cheap, and the red tariff is about 9.5 times more expensive than the blue. Red is considered very expensive. The red tariff will only on occur 22 days or less in a year and not on two consecutive days. The white tariff will only occur on 43 days or less in a year. The blue tariff occurs the rest of the time. The future tariff is only known one day in advance. The Tempo meter has an LCD (display) that can show the current day's tariff colour and the next day's tariff colour.

The domestic properties using Tempo have a contract for electricity that stipulates the maximum current that can be supplied. A contract for more current is more expensive than a contract for less current. If this maximum current is exceeded then the circuit breaker for the property will trip. This circuit breaker can be reset by the owner if the maximum current is no longer exceeded.

Existing Electricity Controllers for the Tempo Tariff

To date the Tempo tariff is barely used (less than 1700 users). It is recommended for residential primary and secondary homes that have a secondary heating supply as well as electrical heating, for example, coal or gas. The owners of the secondary homes can decide to risk that their visit to the home occurs during a red tariff period in return for the overall savings that they might expect. A simple controller is included in the meter to switch on and off a circuit depending on the current tariff and time. This simple controller could be used to have the electrical heating on during off-peak blue tariff periods. This simple controller requires additional wiring to the controlled appliances. Additional such simple controllers, external to the meter, are also available.

A more complex controller, the StarBox controller, is available for the Tempo tariff. This controller offers similar functionality to the OpEnMan controller, although the low-level heating control algorithms are not as sophisticated as those in the OpEnMan controller.

Overview of the OpEnMan Controller

The OpEnMan controller has currently been developed as a complete prototype. The prototype will be used to assess whether this controller in particular, and pre-configuration controllers like this controller in general, allow better management of the Tempo tariff and maximum current limit with easier installation than the existing Tempo controllers. These features aim to create a market for Tempo to enable the energy supplier to reduce their energy production costs, by flattening and reducing the load curve. The controller, if produced, will be pre-configured by installers using information about the occupants' lifestyles and the property gathered using a questionnaire. The user can then alter some of the settings while the system is in use at the property.

The system is installed by plugging the controller into the electricity supply and the meter. Each appliance to be controlled has a smart plug that connects between the electricity supply and the appliance. The main controller can direct the smart plug to turn on or off; it can also determine the ambient temperature at the smart plug, as well as at the controller. There is a local override on the smart plug to allow it to be switched on or off independently from the control of the main controller. Figure 36 shows a schematic of a typical OpEnMan installation. Figure 37 shows the displays and buttons on the OpEnMan controller.

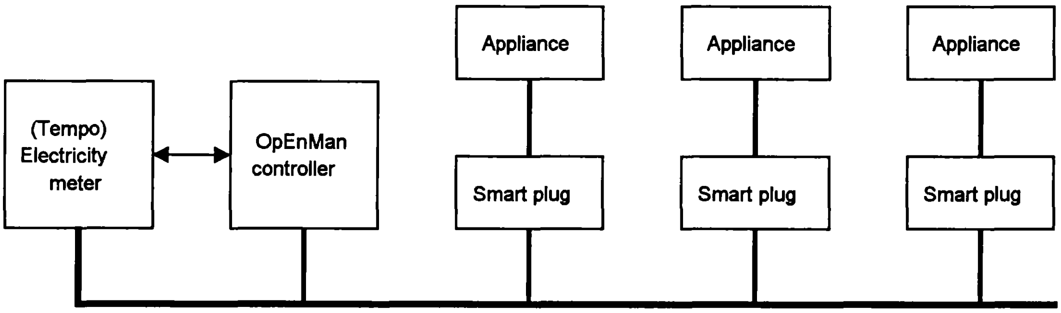


Figure 36. OpEnMan Installation Schematic.

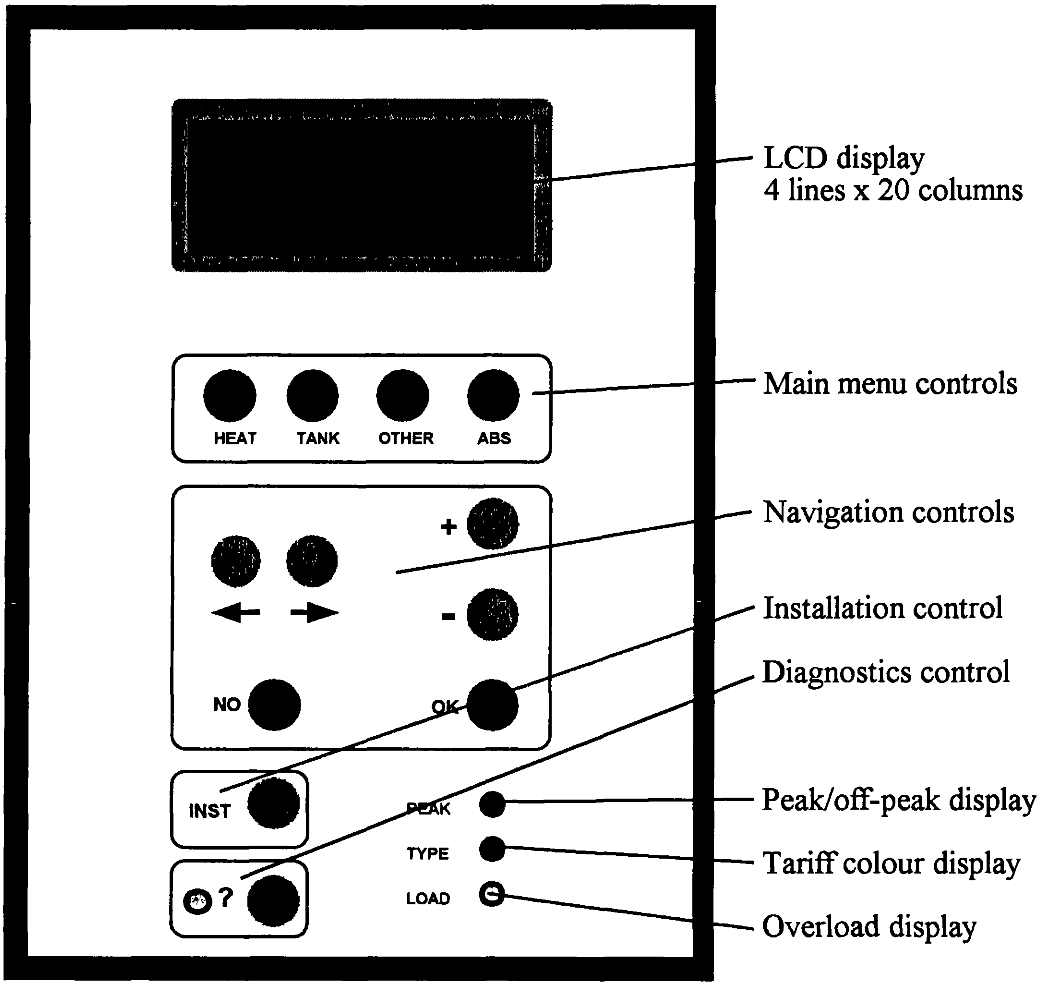


Figure 37. OpEnMan Displays and Buttons.

Normally the controller displays the 'main display' on the LCD display. The 'main display' contains a welcome message, the current date and time, the current external temperature, and the current total current being drawn. The right arrow button will scroll through 'general information menu' screens which allow the date and time to be set and the heating control to be started or stopped. Pressing one of the 'heat', 'tank', 'other', or 'absence' buttons gives a different menu: the 'space heating menu', the 'water heating menu', the 'electric appliances menu', or the 'absence menu'. The relevant parts of these menus will be described in further detail below.

The navigation controls allow: the screens of the menu to be selected using the arrow buttons; selection of an option using the OK button; switching on or off an option or selection from a list of options using the + and - buttons; movement within a screen using the arrow buttons; and cancellation of the current screen and return to the main display using the NO button.

The installation and diagnostic controls are not used by the occupants of the home.

The peak/off-peak display is lit if the current time is within a peak period and unlit otherwise. The tariff colour display is blue, white, or red depending on the tariff of the day. The overload display is lit if the current being drawn is equal or greater than the maximum current permitted and unlit otherwise. If the controller is preventing an appliance from being used, to ensure that the maximum current is not exceeded, then the overload display blinks.

Controlling the Heating using the OpEnMan Controller

The user can select whether the heating control of the controller is 'active' or 'stopped': active means that the controller does control the heating, stopped means that the heating (and cooling) is off. This selection is made from the third screen on the 'general information menu'.

OpEnMan does not control cooling or ventilation appliances.

Zones

The space to heat is divided into 'zones', each of which is controlled separately from the other zones. The first screen of the 'space heating menu' allows the user to select the zone for which they wish to view and alter the settings by pressing the 'heat' button to scroll through the zones. This screen shows information about the zone and allows the user to change the current temperature in the zone.

Patterns

The controller comes pre-configured with a set of template patterns which have names like 'weekends', 'Wednesdays', and 'weekdays' (depending on the configuration). The configuration is covered in its own section below.

A template pattern determines the heating to be applied over a 24 hour period. The pattern allows the heating state, 'high' or 'low', to be set with a granularity of fifteen minutes. Each day of the week is assigned a pattern (during configuration), termed the 'default' pattern. At the beginning of each day, the default pattern for that day is made the current pattern for that day. When the heating is 'high', the temperature is at a 'comfort' or an 'economy' setting as defined during configuration. When the heating is 'low', the temperature is at a 'reduced' or 'night-reduced' setting as defined during configuration.

The configuration defines a temperature for the 'comfort', 'economy', 'reduced', and 'night-reduced' settings. It also defines which of these settings is to apply depending on the tariff, on/off-peak, and zone; for example: the controller could be configured to apply the economy setting on red days during on-peak and the comfort setting at all other times for a zone, when the pattern is 'high'.

User Override of the Current Pattern

The user can alter the current pattern for the day using the second screen on the 'space heating menu' of the controller. Altering this pattern does not alter the default pattern. The altered pattern is, by default, based on the current pattern but, if desired, can be based on any of the default patterns.

The user can also change the currently desired temperature using the first screen on the 'space heating menu' of the controller. An altered temperature will be maintained until the temperature is altered again, a transition occurs in the current pattern, the tariff changes, or midnight arrives.

Absence Override

The user can also indicate, using the first screen of the 'absence menu' on the controller, that they have left the property. The heating is set to an 'absence' setting in all zones, and all patterns, until the user returns and presses a key on the controller. The temperature for the absence setting is determined during configuration. The user can set, using the second screen of the 'absence menu' on the controller, that they have left the property and will return on a particular date at a particular time. In this case, the heating will resume in order to achieve the appropriate pattern for that date at that time.

Further Heating Controls

The third screen on the 'space heating menu' on the controller shows, for information: the current desired temperature, according to the current pattern, etc. or the overriden temperature desired by the user; the current actual temperature of the zone; and the current setting for the heating (comfort, economy, etc.).

Demand Control

The controller can ensure that the current drawn is maintained at less than the maximum current permitted under the electricity supply contract. The controller will switch off appliances, including heating appliances, according to a priority list if the current approaches the maximum. The appliances are switched on again in the reverse order if the current falls. Appliances of the same priority are cycled if necessary. The priority list is determined during configuration.

Configuration of the OpEnMan Controller

The OpEnMan controller is intended to be configured at an agency before installation. The configuration can be updated by a member of the agency visiting the property. The configuration is generated from a questionnaire. The relationship between the questionnaire and the configuration is not explicit, but can be inferred to a reasonable extent.

The questionnaire asks two types of questions: questions to determine whether Tempo and the OpEnMan controller are suitable; and questions to determine the configuration of the controller.

According to the questionnaire, Tempo is considered suitable if the property is the occupant(s) principal home and the property will have electrical heating.

For configuration, the questionnaire asks 'direct' questions about a subset of the configurable options of the controller. The contrast with direct questions would be questions about the life-style of the occupants of the home. In other words, the responder to the questionnaire is asked to 'configure' a subset of the controller using the questionnaire. The configuration requires that devices remain in fixed locations.

Potential User Requirements for the Sponsor's Energy Management System

A list of potential (stereotypical) scenario installations—homes, users, and configurations—of the OpEnMan energy management system is shown in Figure 38. A list of potential user requirements, inappropriate usability and functionality, that might occur in such scenarios is shown in Figure 39.

User	Configuration	Can Use
old lady: at home mostly, pop out to the shops, gardening, go away for holidays	comfort 18°C, economy 16°C, heating on all day, off at night	no

User	Configuration	Can Use
retired couple: at home mostly, one pops out at any time, go away for holidays together	comfort 18°C, economy 16°C, heating on all day, off at night	no
semi-retired couple: reasonably separate lives (very varied), both out often, one home more than two	comfort 18°C, economy 16°C, heating on in morning and late evening, off during day and night	no
middle-aged working couple (working man + working woman): both out during work times, both in during evenings	comfort 19°C, economy 17°C, heating on in morning and late evening, off during day except at weekends, off at night	no
middle-aged couple (working man + housewife): working man in during evenings with woman, woman pops out at any time	comfort 19°C, economy 17°C, heating on all day, off at night	no
middle-aged working older family (working couple + older children at school): woman returns earlier for children	comfort 19°C, economy 17°C, heating on in morning and early evening, off during day except at weekends, off at night	yes
middle-aged working younger family (working couple + younger children at day school): woman does part time morning job	comfort 19°C, economy 17°C, heating on in morning and early afternoon, off during morning except at weekends, off at night	no
middle-aged working younger family with nanny (working couple + younger children at day school + nanny)	comfort 19°C, economy 17°C, heating on all day, off at night	no
middle-aged couple older family (couple + older children at school): woman definitely at home for when children return from school	comfort 19°C, economy 17°C, heating on all day, off at night	yes
middle-aged couple younger family (couple + younger children at day school): woman takes and collects children from day school	comfort 19°C, economy 17°C, heating on all day, off at night	no
young adult couple (working couple + out a lot): at home some evenings only, back late other evenings	comfort 19°C, economy 17°C, heating on in morning and late evening, off during day except at weekends, off at night	yes
young adult working family (working couple + younger children at home + nanny)	comfort 19°C, economy 17°C, heating on all day, off at night	yes
young adult family (couple + younger children at home): woman at home a lot	comfort 19°C, economy 17°C, heating on all day, off at night	yes
young single adult (working + out a lot): at home some evenings only, back late other evenings	comfort 19°C, economy 17°C, heating on in morning and late evening, off during day except at weekends, off at night	yes

Figure 38. OpEnMan Installation Scenarios.

	old lady	retired couple	semi-retired couple	middle-aged working couple	middle-aged couple	middle-aged working older family	middle-aged working younger family	middle-aged working younger family with nanny	middle-aged couple older family	middle-aged couple younger family	young adult couple	young adult working family	young adult family	young adult single
TASK QUALITY														
too cold during day			!	!	!	!	!	!	!	!				
too cold in really cold weather	!	!	!	!	!	!	!	!	!	!				
too cold during illness	!	!	!	!	!	!	!	!	!	!	!	!	!	!
too cold on economy	!	!	!	!	!	!	!	!	!	!	!	!	!	!
too cold late some nights											!			!
too cold early some evenings											!			!
too hot/too cold if heating on/off in summer	!	!	!	!	!	!	!	!	!	!	!	!	!	!
too hot during some tasks when heating on (e.g. cleaning, DIY)	!	!	!	!	!	!	!	!	!	!	!	!	!	!
HUMAN BEHAVIOURAL COSTS														
excessive time &/ effort using controller						!			!		!	!	!	!
potential ill-feeling from being forced into Tempo tariff	!	!	!	!	!	!	!	!	!	!	!	!	!	!
HUMAN STRUCTURAL COSTS														
excessive time &/ effort learning to use controller						!			!		!	!	!	!
COMPUTER BEHAVIOURAL COSTS														
COMPUTER STRUCTURAL COSTS														
initial expense too high	!	!	!											
high running expenses during holidays	!	!	!	!	!		!	!		!				
high running expenses during weekends				!	!		!	!		!				
high running expenses when popping out	!				!		!	!	!	!				

Figure 39: List of Potential User Requirements for an Installation of the OpEnMan system.

Appendix B. Questionnaire for User Requirement Selection

Questionnaire

- Are you ever too cold or too warm...
  - when moving from room to room?
  - due to a change of task?
  - due to a difference of opinion between the occupants of the house?
  - due to visitors?
  - due to unusual events (e.g. staying up late to watch a film, going out)?
  - due to problems programming the heating controller?
- Also...
  - do you ever have difficulty programming the heating controller?
  - is your heating too expensive?

Notes

Questions are only prompts and are not intended to result in non-overlapping answers.

Supplemental Questions for Cycle 2

1. What are the current settings for the timers and boilers? When do you change them? When do you use any of the other controls?
2. Why are the study and studio always cold for working?
3. Why do you want the ability to set the timers with different times for weekends?

Appendix C. Background Information for Cycle 1

Relevant background information about the Cycle 1 user requirements.

Background Information

The house is occupied by the author (Adam) and his girlfriend (Sam). He is a researcher who bicycles to his college most weekday mornings at a varying time (usually between 7.30 o'clock and 10 o'clock) and returns in the weekday evenings at a varying time (usually between 5.30 o'clock and 7.30 o'clock). Some days he works at home during the weekdays.

Sam is an Architect who bicycles to Liverpool Street station to catch a train to work in Chelmsford every weekday morning. She leaves at either 7.30 o'clock or 8 o'clock. Sometimes she will go directly to her site by car, in which case she leaves at 8.30 o'clock. She returns from work at 7 o'clock (either by car or by bicycle).

The weekday evenings are usually spent cleaning, cooking, eating, and chatting. Occasionally Adam does some work in the evenings, they visit the pub after eating, friends visit, they go out for the evening, or they do some DIY. They usually go to bed at 10.30 o'clock except when out or friends visit.

They usually spend the weekends together, either essentially at the house with the aim of doing DIY on the house, going out on Saturday and/or Sunday evening, and perhaps going out for half of a day; or away from the house visiting other people's houses. Occasionally one or both of them will work for a day: either at college/work or at home. They usually go to bed at 11 o'clock except when out or friends visit.

The heating system was recently installed. It has a gas powered combination boiler that supplies hot water for the radiators and the taps. The temperature of hot water supply is set within the boiler. At the moment it is set to very hot (about 80°C). The time of potential supply of hot water is set using a 'timer' (or time controller). The timer allows from no *on-off* periods to up to two *on-off* periods to be set per day. It can be 'boosted' to give an *on* period for an hour from the moment the boost button is pressed and it can be 'advanced' to toggle the current *on-off* state until the next programmed state toggle change. The supply of timed hot water to the radiators is controlled by a hall-mounted thermostat and radiator thermostats for each of the rooms not on the ground floor.

Each room in the house has at least one opening window. Most of the rooms have doors. The roof space is insulated.

The normal setting for the thermostat in the hall is 18°C. The room thermostats are set at maximum (i.e. no restriction of water flow) except in the spare bedroom where they are set to about half-heat.

The timer is set to have two on-off periods: *on* at 6.40 am and *off* at 7.20 am; and *on* at 6.30 pm and *off* at 10 pm.

Appendix D. Background Information for Cycle 2

Relevant background information about the Cycle 2 user requirements.

Background Information

The home is occupied by *J* and *D* who live in Hampshire. He is a businessman who has interests in several different business concerns located in London, Cambridge, and abroad. He has a major leisure pursuit of a cruising boat, which is moored near their house. She is a part-time masters degree student at a nearby University and spends the rest of her time looking after the house, gardening, and performing local neighbourhood committee functions.

J normally works at home for most of the day (in the dining room or study 1) or leaves early in the morning to work in London (leaves between 7 and 11am, returns between 6 and 11pm). If he is at home, then he may go out to attend to his boat. He sails at weekends and on longer trips. These trips are usually planned in advance.

D spends most days at home: part working and part attending to the house. She leaves the house to go to University and to the shops for short periods during the day. Sometimes she will go sailing with J; and sometimes she will go away on a trip related to her field of study. These trips are usually planned in advance.

Breakfast is usually taken in the kitchen. Lunch and supper are usually taken at a small round table in the sitting room unless there are visitors when the dining room is used.

They both visit London and other locations to meet friends and relations regularly. This can mean being away from home for the night or returning late at night.

The heating system was installed about 10 years ago. There are two gas powered combination boilers that supply hot water for the radiators and the taps in the main house: one of the boilers is in the dining room and the other in the lobby. The boiler in the dining room supplies radiators in the: sitting room 1; dining room; cloakroom; front porch; laundry room; landing; bedroom 2; study 1; and bathroom 2. The boiler in the lobby supplies radiators in the: kitchen 1; bedroom 1; bathroom 1; and studio (study 2). Both of the boilers has a timer beside the boiler. There is no central thermostat for the radiators in the main house; they all have individual thermostats which are rarely used. There is a gas powered boiler in a separate cottage that supplies hot water to a tank and radiators in the cottage: sitting room 2 (study 3);

bedroom 3; bedroom 4; kitchen 2; inner hall; and bathroom 3. The cottage has a central thermostat in the sitting room and a timer in the kitchen. Electric fires are often used in study 1 and 2.

Each room in the home has at least one opening window, but they are difficult to open as they are fitted with security locks. The home is well insulated but very exposed to the elements.

Appendix E. Composite Structures

The composite structures were developed during the operationalisations by identifying groups of processes that occur repeatedly. Their development involved identification within and across operationalisations. Once identified, the operationalisations were re-expressed with the composite structure. For each composite structure, the unitary costs are summed to give the composite structure's costs. These costs are shown in the operationalisations.

Goals

The goal store processes repeatedly occur in certain orders. For example, a goal is formed, other behaviours occur, then the goal is popped. Figure 40 shows the composite structures relating to goals. The stubs permit expression of behaviours occurring within the composite structure.

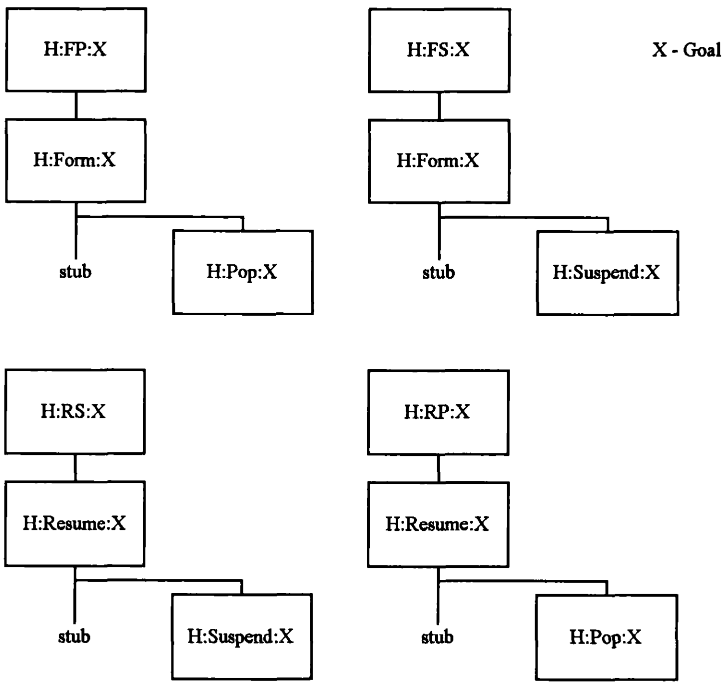


Figure 40. Composite Structures Relating to Goals.

Human Physical Execution

Figure 41 shows the composite structure relating to human physical execution.

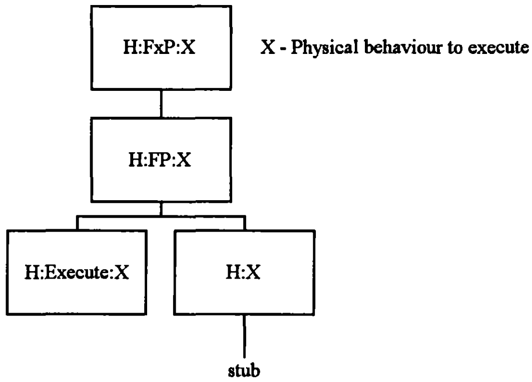


Figure 41. Composite Structures Relating to Human Physical Execution.

Computer Input and Output

Figure 42 shows the composite structure relating to computer input and output.

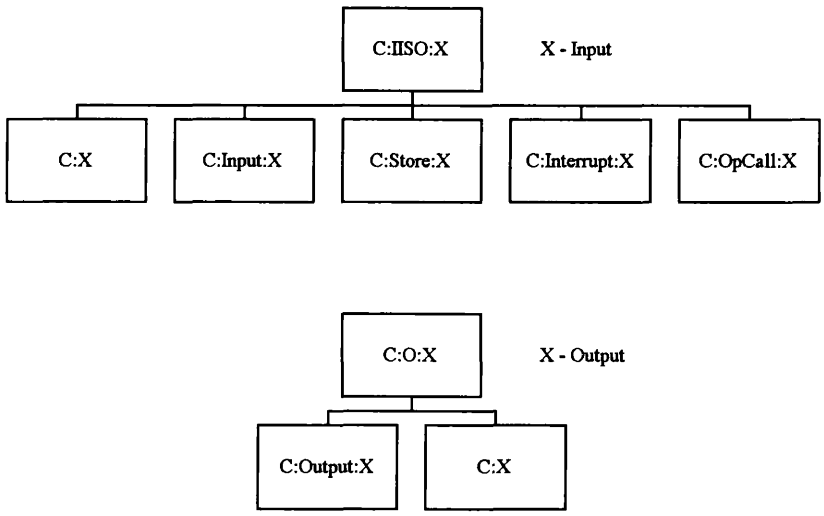


Figure 42. Composite Structures Relating to Computer Input and Output.

Cooking

A composite structure was developed from the operationalisations of cooking: the change in level of the gas for the cooker. Figure 43 shows the composite structures relating to cooking. The domain connector, ●, indicates the connection of any domain connector for the composite structure.

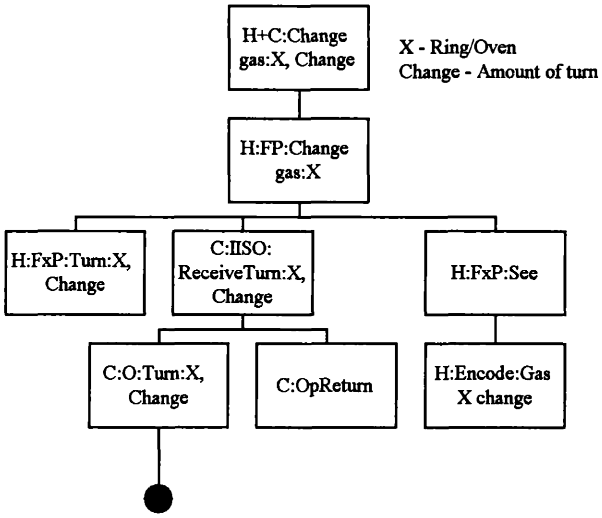


Figure 43. Composite Structures Relating to Cooking.

Planning Composite Structures

Figure 44 and 45 show initial composite structures that relate to monitoring and planning developed during the Cycle 2 current operationalisation. These initial composite structures, shown in Figures 46 and 47, were developed further to relate to the Cycle 1 and Cycle 2 actual operationalisations. Figure 47 also shows a composite structure developed for initial engineering principle acquisition, the StShSubPlan composite structure.

These Figures show some parameters beside or underneath some constructs. These parameters refer to structure state changes and were used during operationalisation. They have been retained in some of the operationalisation diagrams to support comprehension.

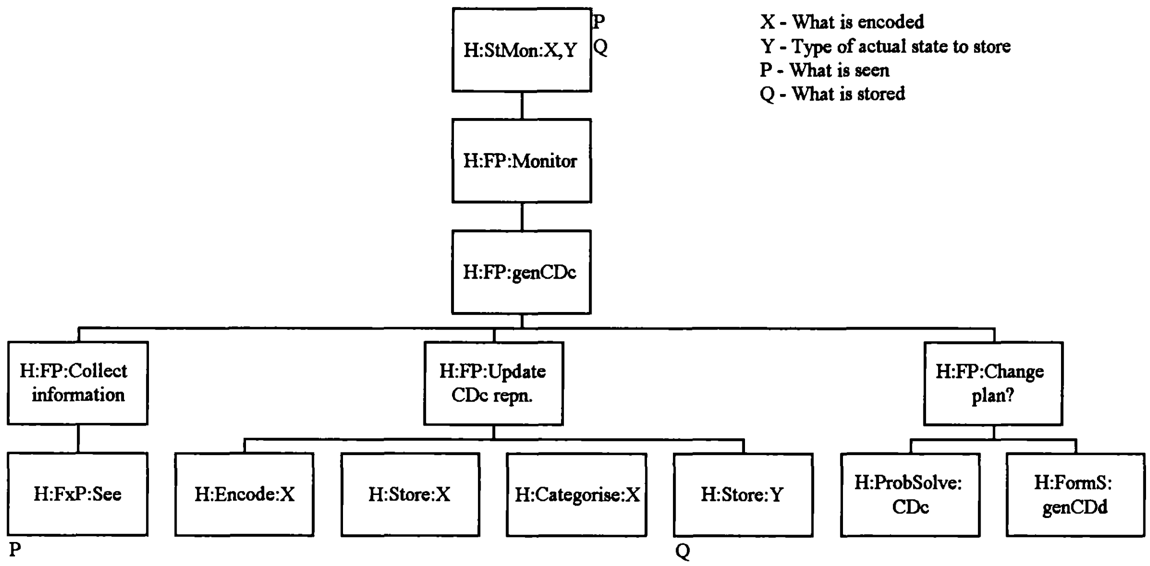


Figure 44. Initial Composite Structures Relating to Monitoring.

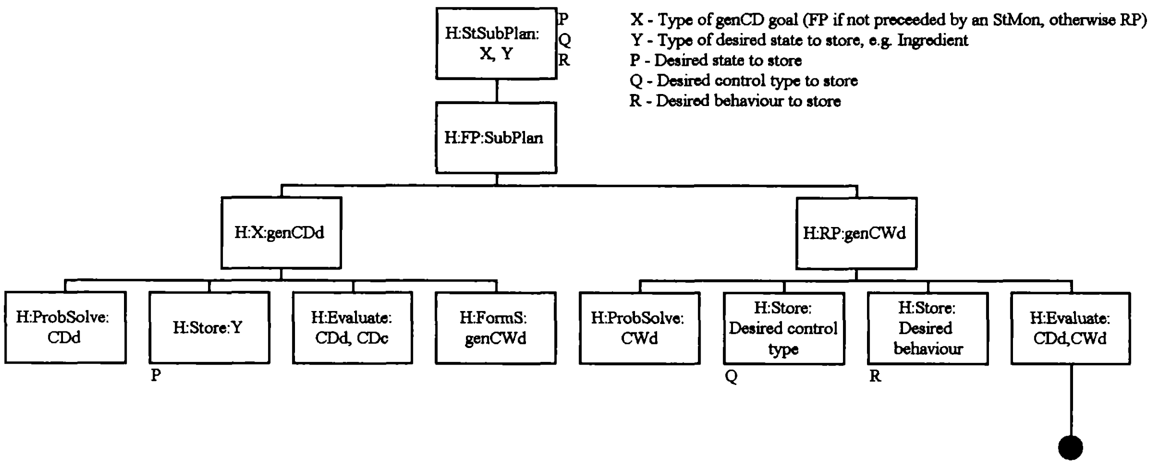


Figure 45. Initial Composite Structures Relating to Planning.

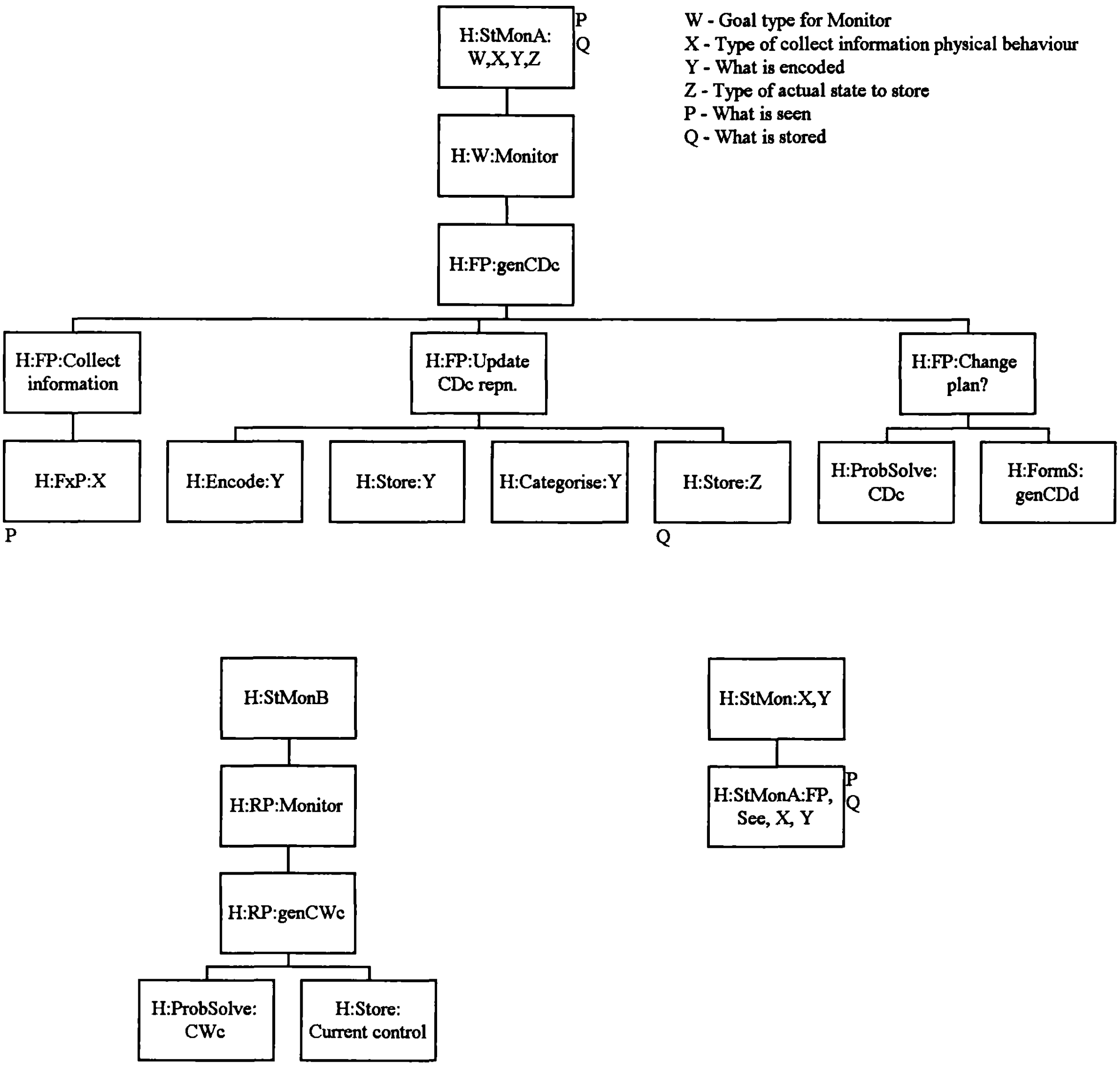


Figure 46. Final Composite Structures Relating to Monitoring.

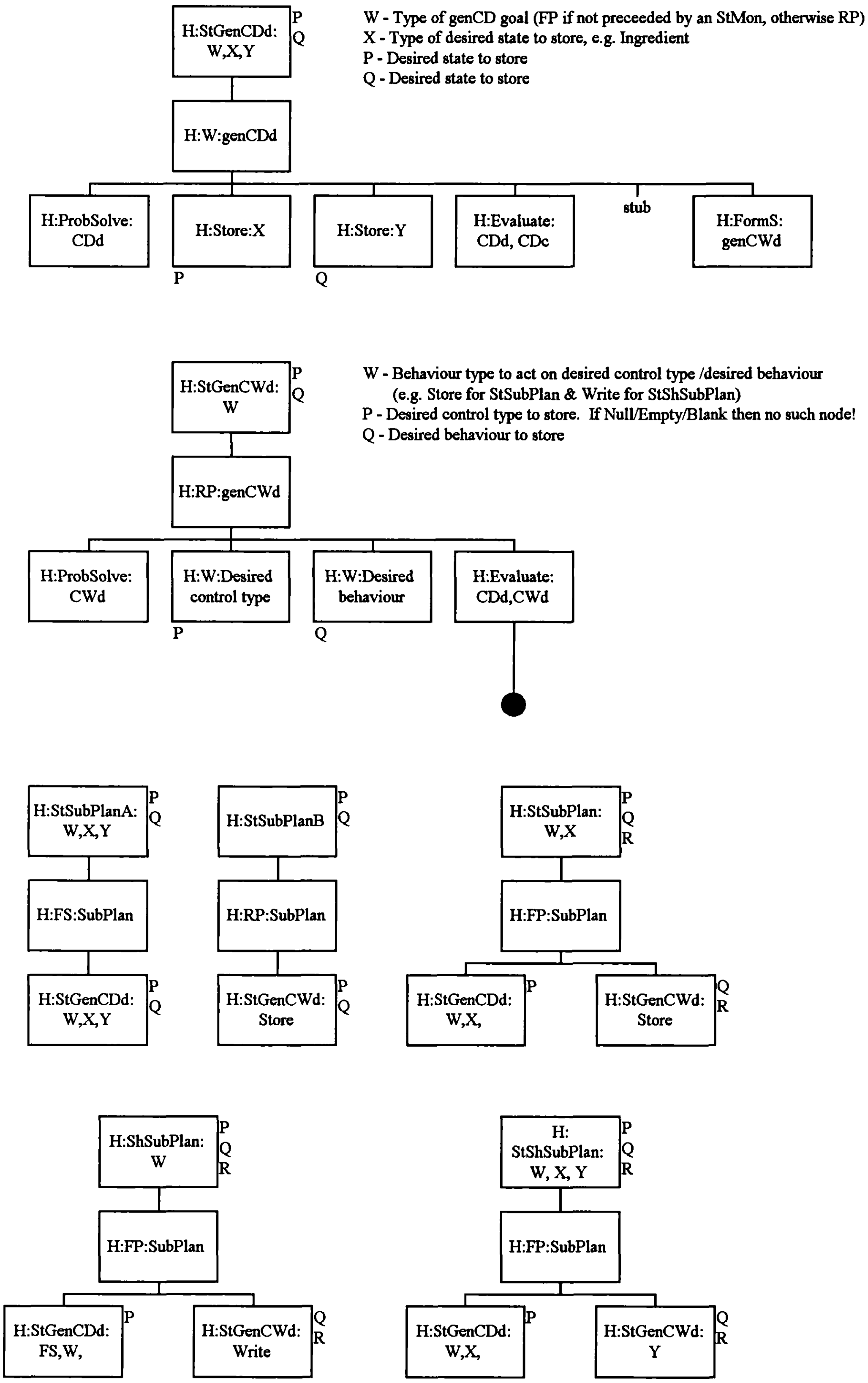


Figure 47. Final Composite Structures Relating to Planning.

Appendix F. Complete MUSE Application for Cycle 1

Initial SUR(y)

The statement of user requirements of the target system—SUR(y)—for this design is to solve the following identified user requirements:

> 'If *A* leaves after 8 a.m. or stays at home to work, then the house is too cold until he turns the gas-powered central heating back on. If he expects to be at home for a short time, then he often uses the one-hour boost facility on the heating controller to turn the heating back on. However, if he is then at home for more than an hour, he can become cold. *A*'s ability to work is adversely affected by being cold and having to control the heating. The nature of his work means that it is difficult for *A* to plan much in advance whether he will be at home, and if so, for how long.'

The following constraint must also be fulfilled:

> 'The current gas bill is acceptable for the comfort; an increase could be considered acceptable for greater comfort. A decrease for the same comfort or better would be desirable.'

There is additional detail given in Appendix C concerning the house, the occupants, the occupants' lives, the heating system, and the current settings of the heating system.

MUSE Elements of SUR(y)

Domain of Application

The domain of application[17] is that of Home Heating Management, a sub-domain of the domain of Home Energy Management, which is in turn a sub-domain of the domains of Home Management and Energy Management, which are in turn both sub-domains of the domain of Management (or Planning and Control).

[17] The MUSE concept of 'Domain of Application' is not the same as that of Dowell and Long.

Technological Constraints

The technology is only constrained by what is available and acceptably priced for its benefits.

Client-Specified Task Constraints

The task is constrained such that there must remain the flexibility for *A* to remain at the house (i.e. the solution must not specify that he must leave at 8 o'clock).

System Performance Criteria

There are no specific system (controller) performance criteria.

(End-) User Characteristics

The end-users are the occupants of the house. *A* has wide and deep experience of computer systems and *S* deep experience of one computer system (Architectural CAD designing). They lead busy lives, however, and do not wish to spend much time using a new system or learning to use a new system.

Environmental Factors

There are no specific environmental factors.

Extant systems analysis stage

Identification of Extant Systems for Analysis

The Current System

The current system is installed in the author's house. It is a fairly normal house heating controller and is described in Appendix N.

1. A description of the weekday uses of the system.

2. A description of the weekend and holiday uses of the system.

Related Systems

3. Other home heating management systems.
    a. Other researchers may have different heating management systems and routines.
    b. Other people with similar jobs may have different heating management systems.
    c. Other people with different jobs may have different heating management systems.
4. Other home energy management systems.
    a. Hot water provision.
    b. Electric heating energy use.
    c. Electric car re-charging from home.

Partially related Systems

5. Other home management systems.
    a. Alarm clock setting.
    b. Food purchasing and cooking scheduling.
    c. Cleaning scheduling and performing.
6. Other energy management systems.
    a. Apartment-wide energy control.
    b. Office energy control.
    c. Industrial plant energy control.
7. Other management systems.
    a. Security.
    b. Decision Support Systems.
    c. Personal Diaries.

The current system will be analysed initially and others selected afterwards if appropriate.

Extant System System's Analysis of the Current System

This section contains the analysis of the current extant system. The task descriptions for this current system were developed by introspection and by the use of a diary.

First Task Description Of Current System—TD1(Current)

This product is not generalised over more than one scenario. It is based on an introspection of the generalised task (generated directly into a structured diagram) by *A*. The structured diagram is in Figure P1 and the table in Figure 48.

Name	Description	Observation	Design Implication	Speculation
Most weekday mornings	On weekends and holidays, the problem does not occur. On days that *A* goes to college in the morning the problem does not occur.	Problem does not occur at weekends and holidays because cold on waking and easy to turn on heating when visiting kitchen to make tea.	Do not change the behaviour and location of the current controller to affect the weekends and holidays.	Different timer settings for the weekend and weekday.
Heating on	The current controller is programmed to turn the heating on at 6:40am.	Controller is programmable and can have two different on-off periods. Not programmable for different (types of) days.	Re-programming the controller is cheap (within the limits of re-programming allowed).	Could upgrade to a more sophisticated controller (more periods, better day control) quite cheaply. Could use existing computer in the house as a controller.
Early morning tasks	*A* has a (fairly) standard routine in the morning: wake-up, get dressed, and make sandwiches.			
Leave (before heating off)	*A* maintains a mental plan containing the amount of work to be done and the most desirable location for doing the work. This is derived from the diary and to-do list.	The mental plan can change rapidly (usually a half-hour granularity).	Do not rely on stated plan of more than half an hour.	
Heating off	The current controller is programmed to turn the heating off at 7.20am.			

Name	Description	Observation	Design Implication	Speculation
Leave (after heating off)	Similar to leaving before heating off, except heating must be controlled by *A* for him to remain comfortable.	Discomfort due to the cold impacts on *A*'s ability to maintain his mental plan.	Prevent the cold.	
Get ready to go				
Leave house				

Figure 48. TD1(Current) Table.

Second Task Description Of The Current System—TD2(Current)

The structured diagrams and the tables were elicited through *A* maintaining a diary. The diary entries are reproduced here. Figures P2, 49, and P3 show the structured diagrams for the TD2.1(Current), TD2.2(Current), and TD2.3(Current). Figures P4 and 50 show the structured diagram and table for TD2(Current).

TD2.1—Day 1

Time	Activity
7:00	Got out of bed and started to get dressed.
7:09	In bathroom: went to the toilet; and cleaned teeth.
7:15	Made a cup of tea and a glass of ribena. Drank ribena and took vitamin pill.
7:17	Made sandwiches.
7:30	Examined to do list. Said goodbye to *S*. Went to study, tidied up, and started to work.
8:20	Felt cold.
8:30	Went downstairs and put heating on one-hour boost.
9:30	Felt cold, examined to do list, and decided to leave.
9:40	Collected and filled bag.
9:45	Put on coat
9:47	Opened front door and left.

TD2.2—Day 2

Time	Activity
7:00	Got out of bed and got dressed.
7:07	Bathroom.
7:13	Made sandwiches.
7:17	Made tea, drank ribena, eat toast.
7:31	Examined diary & left for work with *S*.

Cycle 1
TD2.2(c)

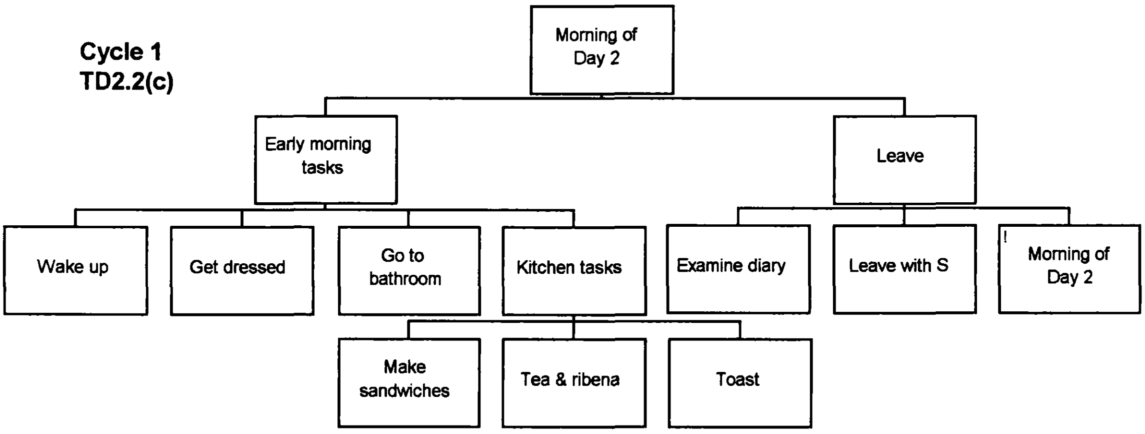


Figure 49. TD2.2(Current) Structured Diagram.

TD2.3—Day 3

Time	Activity
7:00	Got up, ironed shirt, dressed.
7:11	In bathroom: went to the toilet; washed; cleaned teeth.
7:14	Made a cup of tea and a glass of ribena. Drank ribena and took vitamin pill.
7:16	Made sandwiches.
7:32	Examined diary. Said goodbye to *S*. Went to study and started to work.
8:22	Felt cold.
8:30	Went downstairs and advanced heating. Returned upstairs to study.
11:00	Examined diary. Went downstairs and advanced heating. Collected and filled bag.
11:07	Got coat, opened front door, left, closed front door.

Name	Description	Observation	Design Implication	Speculation
Most weekday mornings	This is a generalisation of three weekday mornings	These appear to represent the three different types of workday mornings that *A* has.	The other days should not be affected as they are not part of the problem.	Retain the existing system to some extent. Useful to assess other days for confirmation and other options.
Early morning tasks		*A* has a fairly standard routine on most weekday mornings		Take advantage of any advance planning and his location near the controller? Remind to adjust heating if plan sufficiently advanced.
Leave? (before heating off)	Generalisation of the activities that can occur before the heating goes off: work or leave.	*A* appears to plan using some information sources: a diary and a to-do list. These are stored electronically.		Perhaps interface with the diary/to-do list (but diary/to-do-list plan does not predict reality very well)? Adaptive/predictive controller: Probably not because 1) does not seem to be a predictable task 2) technology not very advanced.

Name	Description	Observation	Design Implication	Speculation
Leave? (after heating on)	Generalisation of the activities that can occur after the heating goes off: work, control heating, and leave.	Planning as above. This is where the problem with comfort occurs: the heating goes off, some time passes, *A* feels cold but does not adjust the controller for some time in order not to break concentration during the work (i.e. gets colder).	Try to avoid the first time that discomfort occurs as well as the later ones.	Perhaps turn the heating off on leaving?
Get ready to go	*A* performs certain tasks before leaving: getting and filling his bag; and putting his coat on.	These tasks are done about 5-15 minutes before leaving. Location of bag and coat varies.		Use the bag (or coat) to trigger heating off as turning it off before leaving will save money?
Leave house	*A* performs certain tasks on leaving: moving to the front door, opening the front door, stepping out, and leaving.	Deadlock on the door always locked when nobody in the house. If the heating has been advanced earlier then *A* must remember to advance it again on leaving. He finds this tricky to do.	Avoid having *A* to remember to turn the heating on or off if possible.	Perhaps use the deadlock (or other trigger at this point) to switch off the heating for morning time period.
Control heating	*A* has to move to the heating controller from his current location, press one of the buttons, and then return to the controller.	Usually upstairs when cold and needing to adjust the controller which is downstairs.	Ensure that controller is near *A* when controlling is necessary.	

Figure 50. TD2(Current) Table.

Domain of Design Discourse of the Current System—DoDD(Current)

Figures 51 and 52 show the structured diagram and table for DoDD(Current).

Cycle 1
DoDD(c)

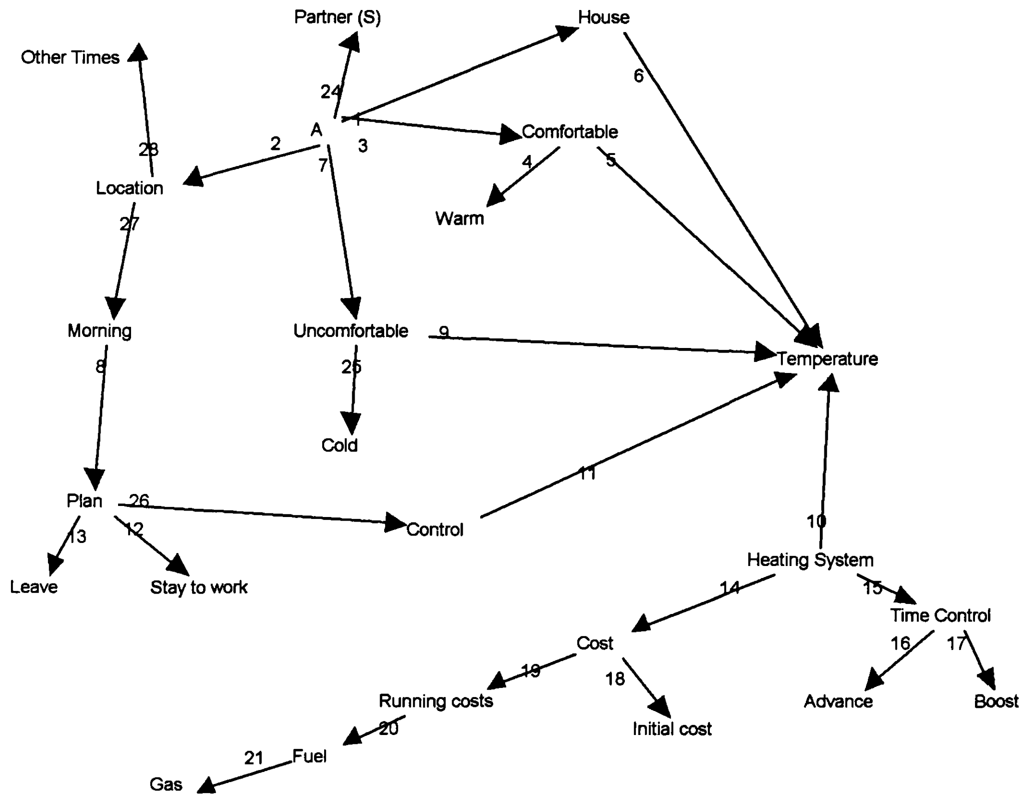


Figure 51. DoDD(Current) Diagram.

Number	Relationship
1	lives in
2	has a
3	desires
4	means
5	caused by
6	has a
7	does not desire
8	component of
9	caused by
10	controls
11	by *A* using heating system
12	component is
13	component is
14	causes
15	controlled by
16	with exception control
17	with exception control
18	type of
19	type of
20	from
21	from
22	has a
23	means
24	results in

Number	Relationship
25	in the
26	at

Figure 52. DoDD(Current) Table.

System Task Model of the Current System—STM(Current)

The diagram (Figure 53) shows a further decomposition of the 'Adjust heating' component of the TD1(current). None of the rest of the TD1(current) is repeated. Figure 54 shows the table.

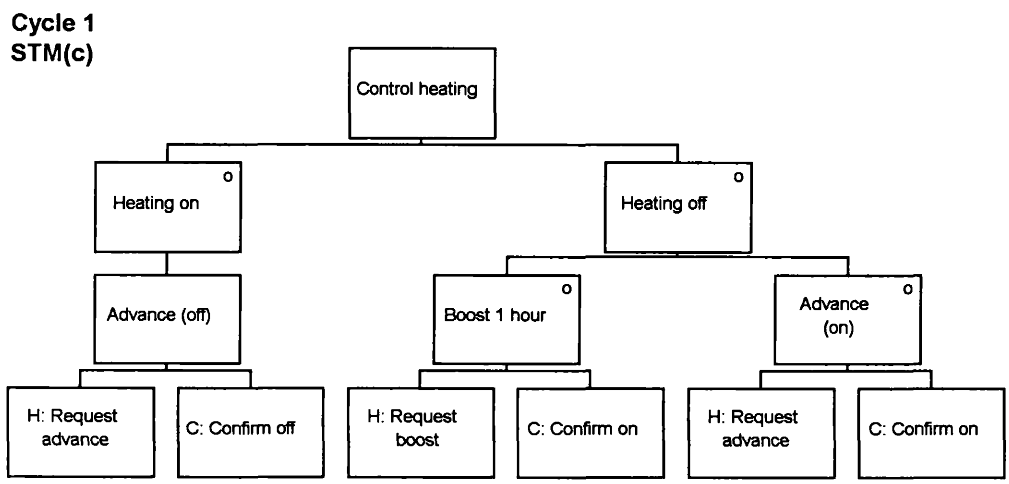


Figure 53. STM(Current) Structured Diagram.

Name	Description	Design Speculation
Adjust heating	Heating can be boosted for an hour or advanced.	These facilities should be retained for compatibility with times not considered.
Heating on	The action of the controls is different if the heating is on or off.	The user appears to be able to cope with this complexity.
Advance (off)	If the heating is off, then advancing the heating will turn it on. The advance button works like a toggle of the current state.	
Heating off	See Heating on description.	
Boost 1 hour	Boosting the heating will turn it on for an hour.	
Advance (on)	If the heating is on, then advancing the heating will turn it off.	
Request advance	The user requests that the heating is advanced.	Current manner of advancing works well for non-problem times (weekends, evenings).
Confirm off	Heating system confirms that the heating has been turned off.	Immediate feedback should be retained in target system.
Request boost	The user requests that the heating is boosted.	Current manner of boosting works well for non-problem times (weekends, evenings).
Confirm on	Heating system confirms that the heating has been turned on.	

Name	Description	Design Speculation
Request advance	The user requests that the heating is advanced. This occurs in the same manner as advance when the heating is on.	Difference between on and off behaviour of advance appears to be understood by and acceptable to user.
Confirm on	Heating system confirms that the heating has been turned on. This happens in the same manner as when the boost is confirmed.	Difference between boost and advance confirmation appears to be understood by and acceptable to user.

Figure 54. STM(Current) Table.

User Task Model of the Current System—UTM(Current)

The diagram are those parts of the TD1(current) that are called 'Plan/re-plan locations at times' and 'Early morning tasks'. They are not separate (or re-drawn) because they were provided by the original analysis that delivered TD1(current).

Interaction Task Model of the Current System—ITM(Current)

The current user interface environment is a hard-wired device in a plastic container.

The diagram (Figure 55) only shows a decomposition from the STM(current) and does not repeat the contents of the STM(current). The screen construct shows the activation point of screens and not the consumption point. Figure 56 shows the table.

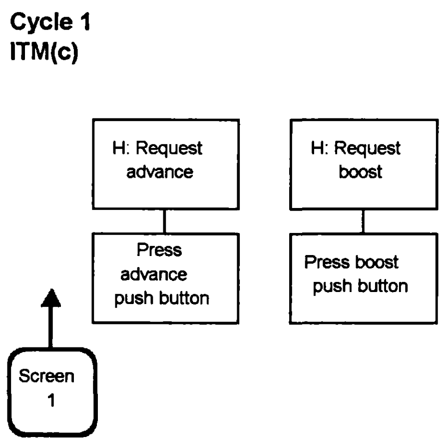


Figure 55. ITM(Current) Structured Diagram.

Name	Description	Design Speculation
Request advance	Request that the heating is advanced.	
Press advance push button	*A* press of the advance button will toggle the state of the heating: if it is on, it will be turned off; if off, turned on.	Push button considered acceptable means of requesting advance.
Request boost	Request that the heating is boosted.	

Press boost push button	A press of the boost button will turn the heating on for an hour.	Push button considered acceptable means of requesting boost.

Figure 56. ITM(Current) Table.

Interface Model of the Current System—IM(Current)

Figure 57 shows the structured diagram.

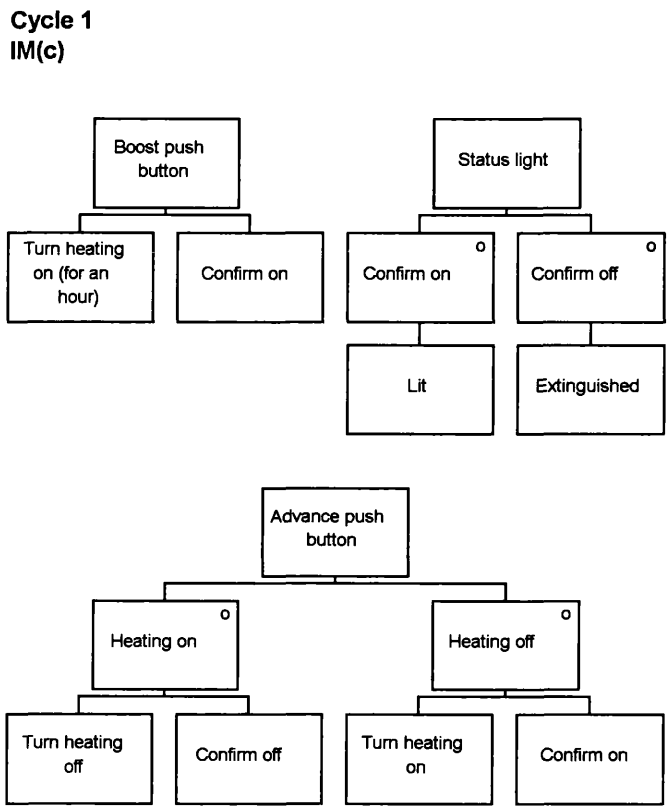


Figure 57. IM(Current) Structured Diagram.

Display Design of the Current System—DD(Current)

Figure 58 shows the diagram for the PSL(Current).

Cycle 1
DD(c)
PSL(c)

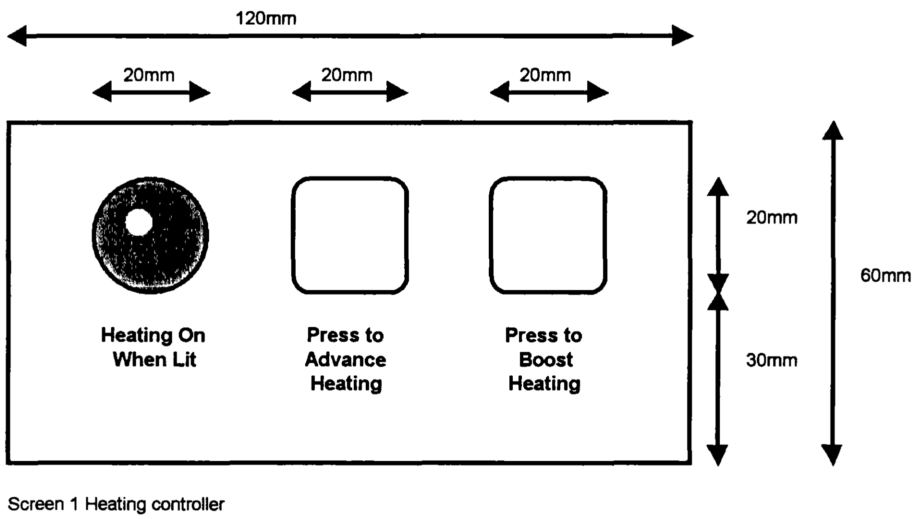


Figure 58. PSL(Current) Structured Diagram.

Figure 59 shows the dictionary of screen objects for screen 1 (the heating controller).

Screen Object	Description	Design Speculation
Boost push button	A tip-of-finger-sized button that can be depressed and springs back to its previous position.	The same style of button could be retained as it has shown to be effective for pressing.
Advance push button	A tip-of-finger-sized button that can be depressed and springs back to its previous position.	
LED for status light.	A small high-intensity focussed light. Displays the current state of the heating.	Works well when lit: not too bright but status can be determined.

Figure 59. PSL(Current) Structured Diagram.

There are no error messages, so no DET(Current). No screens are consumed, so there is no DITaSAD(Current).

General Task Model of the Current System—GTM(Current)

The GTM(Current) is developed from the TD1(current) and the TD2(current). Those elements that are appropriate for porting are in the diagram (Figure P5).

The elements of the TD1(Current) and TD2(Current) tables are not repeated in a table for GTM(Current).

Generalised Task Model Stage

Generalised Task Model for an Extant System Composite—GTM(x)

GTM(x) is the same as GTM(Current) as only one extant system has been analysed currently.

Generalised Task Model of the Target System—GTM(y)

Task Information from SUR(y)

- *A* stays at home to work or leaves.
- *A* needs to control the heating to ensure that he is not cold. This requires informing the heating system of his plan to stay or leave.

Device Independent Summary of Tasks

- *A* stays at home to work or leaves.

Figures 60 and 61 show the structured diagram and table.

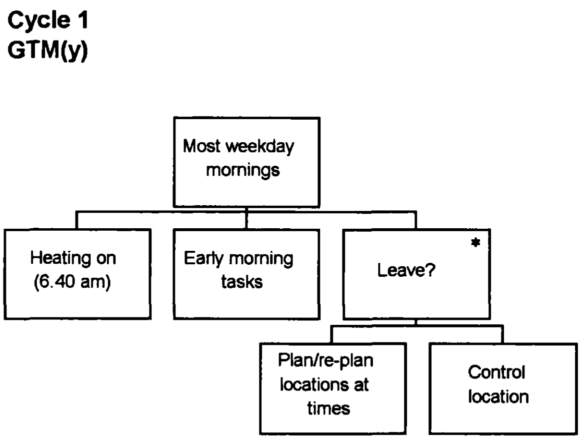


Figure 60. GTM(y) Structured Diagram.

Name	Description	Design Rationale
Most weekday mornings	The problem does not occur at weekends and on holidays	from SUR(y)
Heating on	The controller turns the heating on at 6.40am.	from SUR(y) & extant system: this does not cause any problems.
Early morning tasks	*A* performs a standard set of tasks in the morning.	from SUR(y) & extant system.

Leave?	Allowing for planning and control of staying at home and working or not. The iteration is to cater for the period of re-planning and controlling. On leaving or just before the heating must be turned off. *A* quit construct (for the diagram) must be included in this body to allow for the leaving, otherwise *A* is assumed to stay (end of the diagram at the end of the morning).	Discomfort due to cold avoided by having no cold. Risk of not turning system off must be catered for within this design.
Plan/re-plan locations at times	*A* develops a mental plan for staying at home or leaving.	from extant system.
Control location	*A* replans, decides to stay at home to work based on plan, or decides to leave based on plan.	from extant system.

Figure 61. GTM(y) Table.

Statement of User Needs Stage

Statement of User Needs—SUN(y)

- See SUR(y) for the target system requirements.
- Explicit design constraints from SUR(y): the amount of fuel used cannot increase very much.
- Implicit design constraints from SUR(y): the cost of the new system should not be very high.
- Explicit system performance criteria from SUR(y): *A* must not be cold.
- Implicit system performance criteria from SUR(y): *A* must be permitted to stay/leave the house as desired (i.e. his ability/inability to plan should remain as is). Therefore, that part of the TD1(current) relating to planning and re-planning can remain. The frequency of planning that is essential for this task could still be reduced.
- Existing system results in *A* being cold and has resulted in this requirement for redesign.
- Existing system supports other uses and must not have functionality removed (or over complicated).
- Schneiderman (1992) suggests a guideline: 'eliminate human action when no [human] judgement is required' . This should be augmented to include 'and minimise human action when human judgement is required'. This suggests that

the solution should not require *A* to think regularly about whether the heating is about to go off and perform some action. It also suggests that the amount of action that should be performed by *A* to ensure that the heating stays on should be minimal.

- Feedback on all commands should be provided.
- The system should have a consistent interface.
- Transfer of learning from the extant current system (and other extant systems) should be used where possible.
- Interface Environment: PC Windows (because there is such a system in the house) and Maplin Electronics.

Domain of Design Discourse for the Target System —DoDD(y)

In this case, the DoDD(y) is the same as the DoDD(current), i.e. the DoDD for the current extant system.

Composite Task Model Stage

Composite Task Model—CTM(y)

Figures P6 and 62 show the structured diagram and table. Consult the GTM(y) table for entries that correspond with entries in the GTM(y) table.

Name	Description	Design Rationale
Examine leaving plan	*A* maintains a mental leaving plan that he uses to control his current location for working.	from extant system.
Plan action	Perform the next action indicated by the leaving plan: stay to work or leave.	from extant system.
Stay	In order to work, *A* stays.	from extant system.
Leave	To change work location, *A* leaves.	from extant system.
Perform work	If *A* stays then he works.	from extant system.
Get ready to go	Before leaving the house, *A* gets ready to go.	from extant system.
Leave house	*A* performs tasks before leaving the house.	from extant system.
Get and fill bag	If *A* is near either heating controller while collecting his bag or filling his bag then he should use it to turn off the heating.	The bag collection and filling always happen 5-15 minutes before leaving. The plan is sufficiently formed to ensure that leaving is inevitable. It is cost effective to turn the heating system off before leaving, as long as it is not turned off forty minutes before leaving as this affects comfort. The location of the front door controller (near the base of the stairs) and the main controller is near the location that the bag and some contents are often kept.
Near a heating controller?	Either of the two controllers, the kitchen controller and the front door controller, should remind *A* to turn the heating off and allow him to do so. This design will require a new front door controller and an enhanced main controller that can be programmed differently for the weekends as from the weekdays.	To upgrade to a new, slightly more sophisticated, main controller will not cost very much (approx. £30). The front-door controller can be built cheaply (£10). This will transfer the ability to use the existing controller; maintain the functionality for the other days; and provide the desired level of comfort. The risk is that the heating is left on: the location beside the front door will try to prevent this but a further reminder should be built into the controllers.

Figure 62. CTM(y) Table.

Event List

Figure 63 shows the event list.

Event	Summary	Attributes	Instances
Heating	The heating has a current state	- state of heating	'Heating on', 'Heating remind & off at mc', 'Heating remind & off at fdc'
Location	*A* has a current location	- spatial co-ordinates - functional role	'Stay', 'Leave'
Leaving plan	*A* maintains a leaving plan representing his current intentions for leaving	- criteria - projected locations	'Examine leaving plan', 'Plan action'

Figure 63. Event List.

Functions List

Figure 64 shows the functions list.

Functions	Trigger	End result	Performance
Timed heating on	at 6.40am	Heat radiators	Should match warm-up time for current heating system at the same or less cost
Heating remind & off at mc	*A* leaving and near main controller	No heat to radiators	Should match or better warm-down time for current heating system at the same or less cost
Heating remind & off at fdc	*A* leaving and near front door controller	No heat to radiators	Should match or better warm-down time for current heating system at the same or less cost

Figure 64. Functions List.

System and User Task Model Stage

System Task Model of the Target System—STM(y)

The diagram (Figure 65) only shows a decomposition from the CTM(y) and does not repeat the contents of the CTM(y). Figure 66 shows the table.

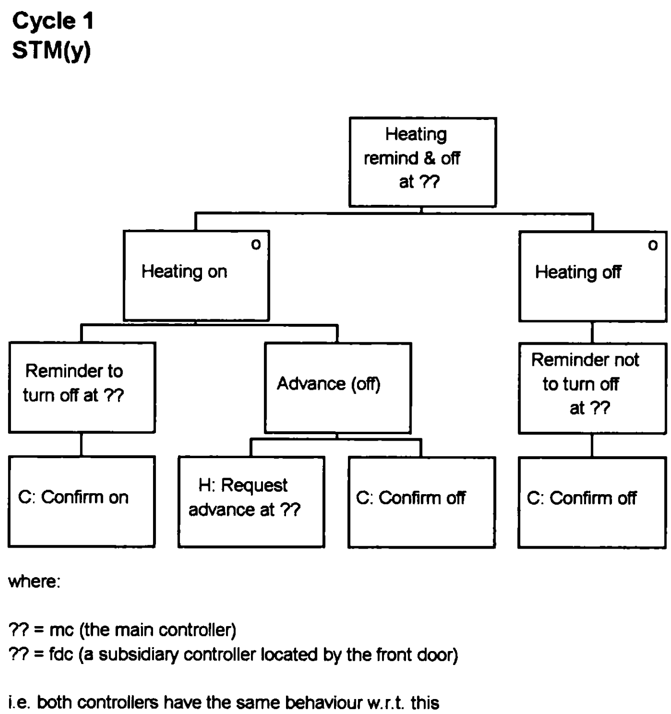


Figure 65. STM(y) Structured Diagram.

Name	Description	Design Rationale
Heating on	Remind and control heating	Only necessary to remind and turn the heating off if it is already on.
Heating off	Remind not to turn off	The state of the heating should be displayed and this will act as a reminder not to turn the heating off (as it is already off).
Remind to turn off at ??	Remind to turn off at either the front door controller or the main controller.	*A* may be in either location when he needs to be reminded to turn the heating off.
Reminder not to turn off at ??	Remind not to turn off at either the front door controller or the main controller.	*A* may be in either location when he needs to be reminded not to turn the heating off.
Advance (off)	The advance facility should be used to control the heating.	Use of the advance facility to turn the heating off transfers the understood behaviour of the existing system. Advance (off) behaviour ported from extant system STM(current).
Confirm on	Confirmation that heating is on.	Feedback on commands ported from extant current system, STM(current). Consistency maintained for remind to turn off.
Confirm off	Confirmation that heating is off.	Feedback and consistency (see above).

Figure 66. STM(y) Table.

User Task Model of the target System—UTM(y)

The diagram (Figure 67) only shows a decomposition from the CTM(y) and does not repeat the contents of the CTM(y). These descriptions are mostly ported from the extant current system, UTM(current). The UTM(y) does not offer any content, format, or mode of presentation issues for the STM(y), ITM(y), etc.

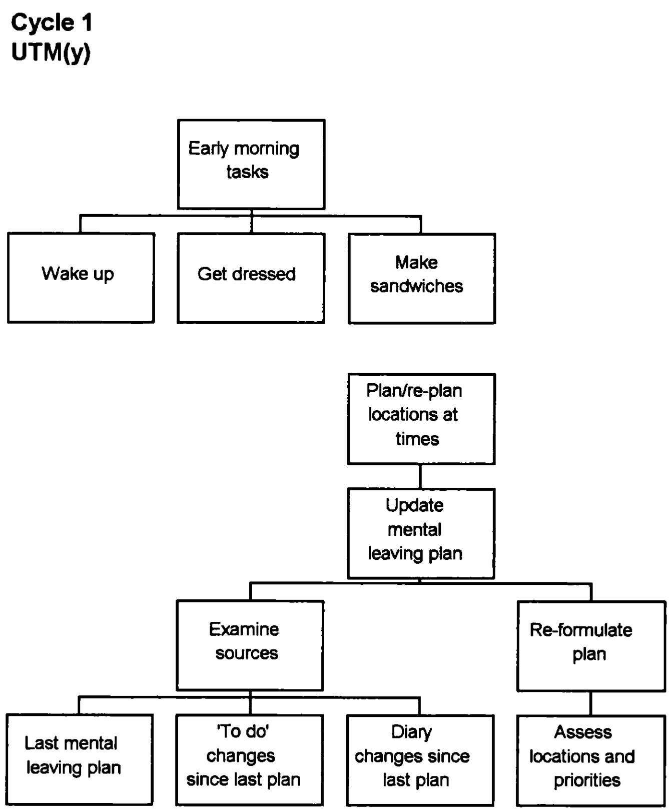


Figure 67. UTM(y) Structured Diagram.

Interaction Task Model Stage

Interaction Task Model of the Target System—ITM(y)

The target user interface environment will be a hard-wired device in a plastic container; this is consistent with the current system. The set of interface objects will be those defined in a Maplin catalogue that are intended for manipulation by humans.

The diagram (Figure 68) only shows a decomposition from the STM(y) and does not repeat the contents of the STM(y). The screen construct shows the activation point of screens and not the consumption point. Figure 69 shows the table.

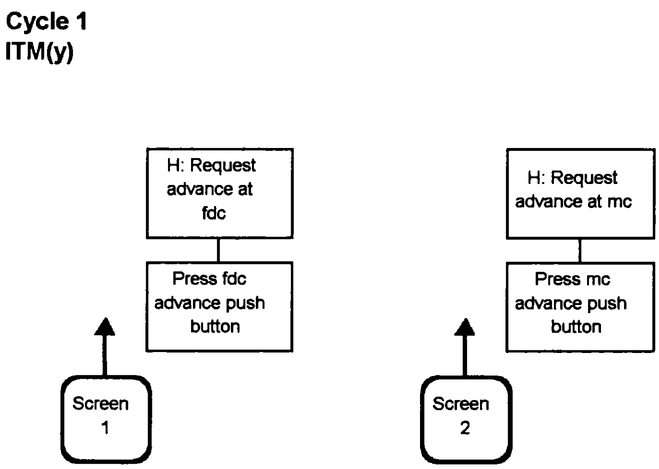


Figure 68. ITM(y) Structured Diagram.

Name	Description	Design Rationale
Request advance at fdc	Advance the heating at the front door controller.	Porting of extant current advance control behaviour for switching off the heating.
Press fdc advance push button	*A* press of the fdc advance button (if the heating is on) will turn the heating off.	Extant current system has an advance button on the main controller: use ported to front door controller in target system.
Request advance at mc	Advance the heating at the main controller.	Porting of extant current advance control behaviour for switching off the heating.
Press mc advance push button	*A* press of the mc advance button (if the heating is on) will turn the heating off.	Extant current system has an advance button on the main controller: use ported to new main controller.

Figure 69. ITM(y) Table.

Interface Model Stage

Interface Model of the Target System—IM(y)

Figures 70 and 71 shows the structured diagram and table.

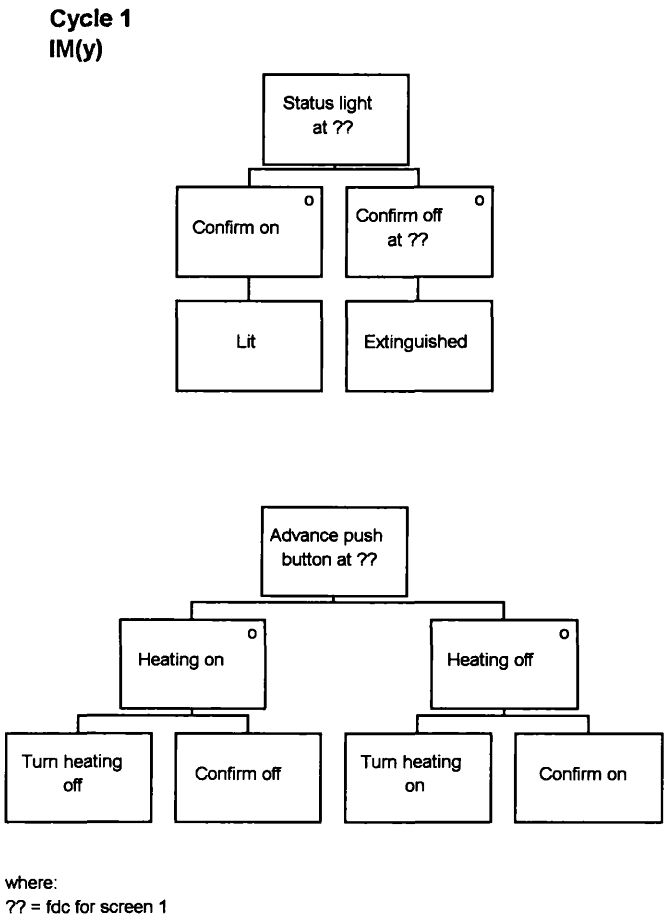


Figure 70. IM(y) Structured Diagram.

Name	Description	Design Rationale
Status light at ??	A red LED light on the front door controller and the main controller.	Red to be noticeable and remind *A* of heating state. Behaviour ported from extant current system. Consistency of behaviour between controllers.
Lit	The LED is bright.	Bright to be noticeable and remind *A* that the heating is on.
Extinguished	The LED is dark.	Dark to be unnoticeable and not remind *A* that the heating is on.
Advance push button at ??	*A* push button on the front door controller and the main controller.	The behaviour for this button is ported from the extant current system. Consistency of behaviour between controllers.

Figure 71. IM(y) Table.

Display Design Stage

Pictorial Screen Layout of the Target System—DD(y)-PSL(y)

See Figure 72. Porting from extant current system for compatibility with user's existing task and device knowledge and cost.

Boost button (and behaviour) retained from extant current system to ensure compatibility with other tasks. Not included in the rest of this design specification (see extant current system description) because it is not part of the solution to this problem.

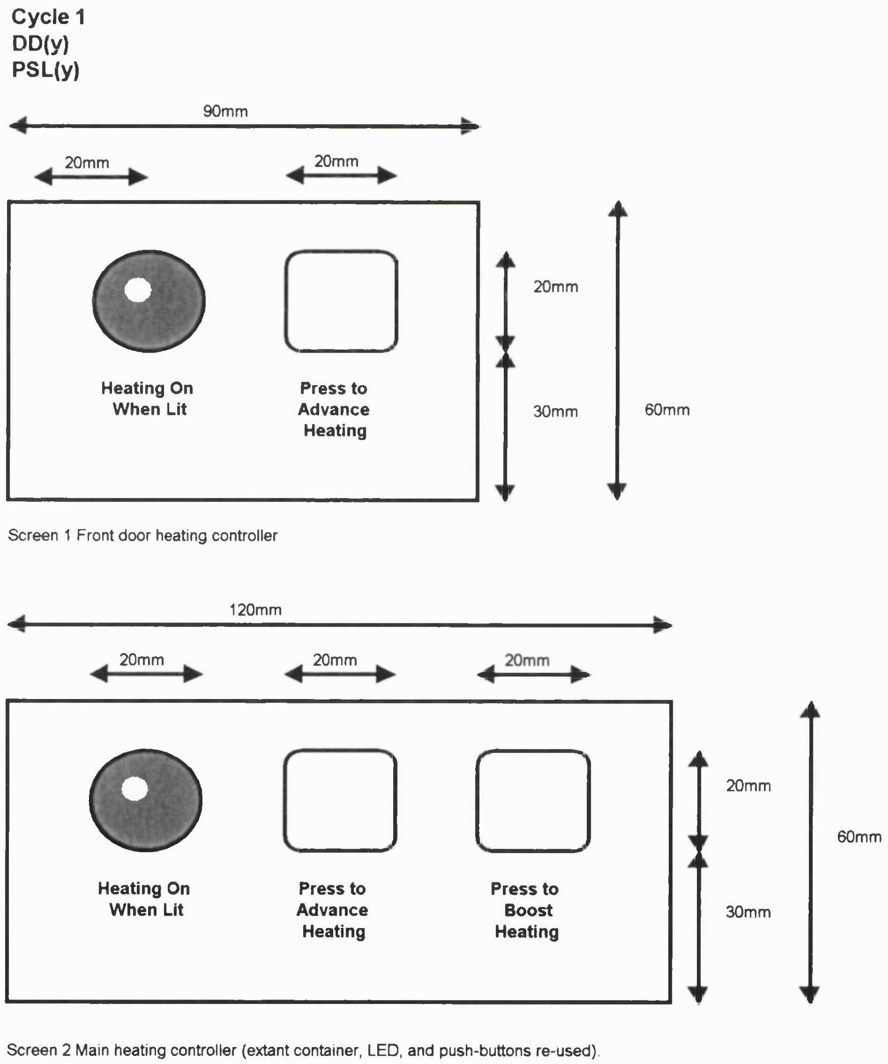


Figure 72. PSL(y) Diagram.

Dictionary of Screen Objects of the Target System—DD(y)-DSO(y)

Figure 73 shows screen 1 (the front door controller) and Figure 74 shows Screen 2—The main controller.

Screen object	Description	Design attributes
Push button for fd advance (Maplin QCA9202403 or equivalent).	A tip-of-finger-sized button that can be depressed and springs back to its previous position. Advances the state of the heating at the front door controller.	A push toggles the current state of something. Advances the state of the heating.
LED for fd display (Maplin UK20W or equivalent).	A small high-intensity focussed red light. Displays the state of the heating at the front door controller.	Lit or extinguished showing the current state of something. Displays the state of the heating.

Figure 73. DSO(y) Screen 1 Table.

Screen Object	Description	Design Attributes
Push button for mc advance (use of extant current system button).	A tip-of-finger-sized button that can be depressed and springs back to its previous position. Advances the state of the heating at the main controller.	A push toggles the current state of something. Advances the state of the heating.
LED for mc display (use of extant current system button).	A small high-intensity focussed red light. Displays the state of the heating at the main controller.	Lit or extinguished showing the current state of something. Displays the state of the heating.

Figure 74. DSO(y) Screen 2 Table.

Dialogue and Error Table of the Target System—DD(y)-DET(y)

There are no error messages.

Display and Inter-Task Screen Actuation Diagram—DD(y)-DITaSAD(y)

Not considered necessary for this design as the screens are not consumed.

JSD Entity and Action Step

Potential Entities and Actions

A: wake-up, leave, stay, return in evening, boost, advance, program time-control, work, plan leaving, view status, be comfortable, be un-comfortable, be warm, be cold, go to bed.

S: wake-up, leave, return in evening, go to bed

Home: occupied, unoccupied

Gas bill: increase, decrease

Heating controller: boost, advance, time-control (heating plan), show status

Heating system: on, off

Assessment of Entities

A is accepted as an entity.

S was initially accepted as an entity. However, she has now been rejected since *A* (nearly) always: wakes at the same time as or after she does; leaves the house at the same time as or after, if at all, she does; returns to the house at the same time as or before she does; and goes to bed at the same time as she does.

Home is rejected as an entity because it would be expensive to recognise the actions occupied or unoccupied. Any resulting changes to the functions should be assessed.

Gas bill is rejected as an entity because its actions are not discrete.

Heating controller is rejected as an entity because it is part of the system itself.

Heating system is rejected as an entity because its actions are outputs of the system.

Assessment of Actions

A: wake-up, leave, boost, advance, return in evening, and go to bed are actions of the real world and are therefore accepted; view status is an output of the system and is therefore rejected; program time control is not an action to be supported by this design and is therefore rejected; work, plan leaving, be comfortable, be un-comfortable, be warm, and be cold are not discrete actions and are therefore rejected.

Result

A: wake-up, leave, boost, advance, return in evening, go to bed

JSD Entity Structure Step

Figure 75 shows the diagram.

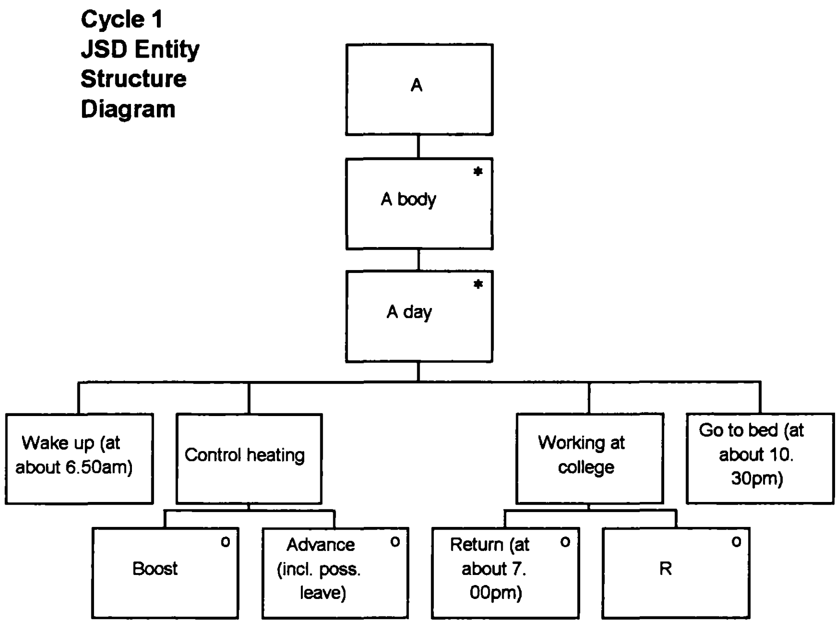
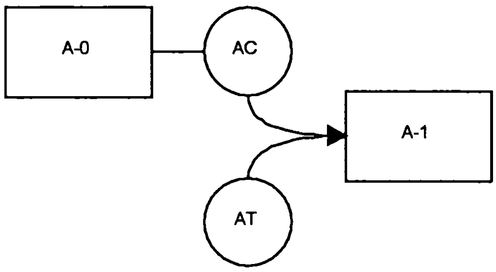


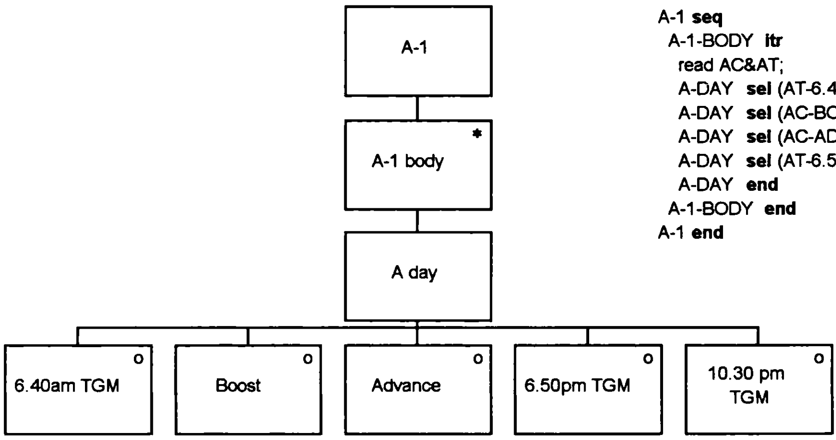
Figure 75. Entity Structure Diagram.

JSD Initial Model Step

Figure 76 shows the diagram.

Cycle 1
JSD Initial
Model
Diagram



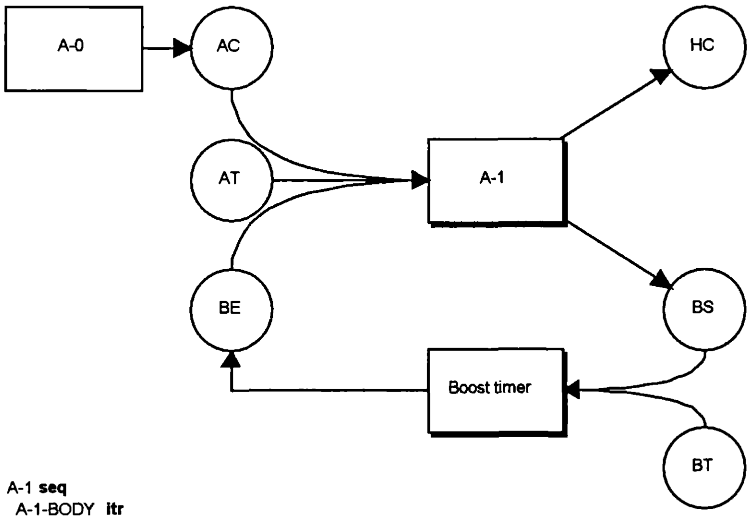



```
A-1 seq
 A-1-BODY itr
  read AC&AT;
  A-DAY  sel (AT-6.40am)
  A-DAY  sel (AC-BOOST)
  A-DAY  sel (AC-ADVANCE)
  A-DAY  sel (AT-6.50pm)
  A-DAY  end
 A-1-BODY  end
A-1 end
```

Figure 76. Initial Model Diagram.

JSD Function Step

Cycle 1
JSD
Functions




```
A-1 seq
 A-1-BODY itr
  send HC-OFF;
  heatingState = off;
  boostState = off;
  read AC&AT&BE;
  A-DAY sel (AT-6.40am & heatingState = off)
   send HC-ON;
   heatingState = on;
   boostState = off;
  A-DAY alt (AC-BOOST & heatingState = off)
   send HC-ON;
   send BS-START;
   boostState = on;
  A-DAY alt (AC-BOOST & boostState = on)
   send HC-OFF;
   boostState = off;
  A-DAY alt (BE-END & boostState = on & heatingState = off)
   send HC-OFF;
   boostState = off;
  A-DAY alt (AC-LEAVE & boostState = off & heatingState = on)
   send HC-OFF;
   heatingState = off;
  A-DAY alt (AC-LEAVE & boostState = off & heatingState = off)
   send HC-ON;
   heatingState = on;
  A-DAY alt (AT-6.50pm & heatingState = off)
   send HC-ON;
   heatingState = on;
   boostState = off;
  A-DAY alt (AT-10.30pm & heatingState = on)
   send HC-OFF;
   heatingState = off;
   boostState = off;
  A-DAY end
 A-1-BODY end
A-1 end
```

```
BOOST-TIMER seq
 count = 0;
 BT-BODY itr
  read BS&BT;
  BT-GROUP sel (BT-TICK) // TGM every second
   count = count - 1
   CHECK-END sel (count = 0)
    send BE-END;
   CHECK-END end
  BT-GROUP alt (BS-START)
   count = 3600; // secs in an hour
  BT-GROUP end
 BT-BODY end
BOOST-TIMER end
```

Figure 77. Function Step Diagram.

System must be switched on overnight only.

JSD Timing Step

The HC-ON & HC-OFF outputs must occur within 1/2 second of AC and AT inputs (to drive status display).

JSD Implementation Step

Implementation on a single processor: interrupt driven communication links.

Appendix G. Full Cycle 1 Operationalisation

The appendix contains the formal and metricated Cycle 1 operationalisation.

Current Solution Operationalisation

Specific Actual Performance

The planning specific **actual performance** is operationalised as the union of the planning specific actual quality and the planning specific actual costs. The planning **worksystem boundary criteria** are operationalised by the requirement that the constituents of the planning **worksystem** have the common goals of the current (level of) achievement and satisfaction of the planning of the comfort of *A* and the leaving of *A*. The planning **domain boundary criteria** are operationalised by the requirement that the constituents of the planning domain of application express the current (level of) achievement and satisfaction of these common goals.

The control specific **actual performance** is operationalised as the union of the specific actual quality and the specific actual costs. The control **worksystem boundary criteria** are operationalised by the requirement that the constituents of the control worksystem have the common goals of the current (level of) achievement and satisfaction of the control of the comfort of *A* in the home of *A* using the heating system and the leaving of *A*. The control **domain boundary criteria** are operationalised by the requirement that the constituents of the control domain of application express the current (level of) achievement and satisfaction of these common goals.

Specific Actual Quality and Costs

There are two main sub-systems in the planning worksystem: the planner (*A*); and the heating controller (a simple two-period time controller). There are two main sub-systems in the control worksystem: the user (*A*) and the heating system (a combination boiler system and the heating controller).

Figure 78 shows the figures that contain the current operationalisation. The **actual costs** are operationalised by the union of the actual resource costs in the tables.

Figure	Contents
P7	Planning domain and worksystem diagram.
P8	Control domain and worksystem diagram
P9	Table (containing the values) for the planning and control quality and costs.
P10	Formulae for the table.

Figure 78. Figures for the Current Solution Operationalisation ('Current').

Specific Design Problem Operationalisation

The specific design problem operationalisation is aiming for a minimal expression, which is achieved by using quality and costs statements that are with respect to the current operationalisation.

Specific Desired Quality

The main task goal is to maintain the state of *A*'s comfort attribute as 'comfortable' instead of a task achieved goal of 'not comfortable'. The leaving and comfort plan quality should be acceptable. In addition, *A* is allowed to leave when desired.

$$\forall e(ComfortTq \neq FALSE)$$

$$\forall e(PlanQuality \neq FALSE)$$
$$\forall e(LPBehaviourScope \in \{A; Form; Leave, -\})$$

$$\forall e(InHouse = TRUE \cap (@e(InHouse = FALSE) > @e(LPBehaviourScope = A; Form; Leave))$$

@e(x) is the event tick of the expression x.

Specific Desired Costs

The physical structural costs of the heating system should be within a range that allows for this desirable decrease or acceptable increase in gas and electricity usage. It is assumed that the heating system can be upgraded and, therefore, the operationalisation of the physical and abstract structural costs of the heating system should be within a range that allows for a different installation and maintenance price. Further, it is expected that a small increase in physical and abstract behavioural costs of the heating system would be tolerated and this would be reflected in the operationalisation within a range.

Physical structural costs of the heating system:

$\forall e(CC : PhysStruct \leq 28))$

Abstract structural costs of the heating system:

$\forall e(PC : AbstStruct \leq 1.5))$
$\forall e(CC : AbstStruct \leq 24))$

Physical and abstract behavioural costs of the heating system:

$\forall e(CC : AbstBeh \leq 21)$
$\forall e(CC : PhysBeh \leq 21)$

It is assumed that the user costs either remain the same, and be operationalised in the same manner—for example the user physical structural costs—, or decrease if possible, and be operationalised to be within a range—for example the user physical behavioural costs.

$\forall e(PA : AbstStruct \leq 81)$
$\forall e(PA : PhysStruct \leq 1)$
$\forall e(PA : AbstBeh \leq 66)$
$\forall e(PA : PhysBeh \leq 1)$
$\forall e(CA : AbstStruct \leq 35)$
$\forall e(CA : PhysStruct \leq 7)$
$\forall e(CA : AbstBeh \leq 41)$
$\forall e(CA : PhysBeh \leq 11)$

Specific Design Solution Operationalisation

Specific Actual Performance

The planning specific **actual performance** is operationalised as the union of the planning specific actual quality and the planning specific actual costs. The planning **worksystem criteria** are operationalised by the requirement that the constituents of the planning worksystem have the common goals of the actual (level of) achievement and satisfaction of the planning of the comfort of A and the leaving of A. The planning **domain boundary criteria** are operationalised by the requirement that the constituents of the planning domain of application express the actual (level of) achievement and satisfaction of these common goals.

The control specific **actual performance** is operationalised as the union of the control specific actual quality and the control specific actual costs. The control **worksystem criteria** are operationalised by the requirement that the constituents of the planning worksystem have the common goals of the actual (level of) achievement and satisfaction of the control of the comfort of *A* in the home of *A* using the heating system and the leaving of *A*. The control **domain boundary criteria** are operationalised by the requirement that the constituents of the control domain of application express the actual (level of) achievement and satisfaction of these common goals.

The Specific Actual Quality and Costs

There is one main sub-system in the planning worksystem: the planner (*A*). There are two main sub-systems in the control worksystem: the user (*A*) and the heating system (a combination boiler system and a simple two-period time controller with remote advance controller).

Figure 79 shows the figures that contain the solution operationalisation. The **actual costs** are operationalised by the union of the actual resource costs in the tables.

Figure	Contents
P11	Planning domain and worksystem diagram.
P12	Control domain and worksystem diagram.
P13	Table (containing the values) for the planning and control quality and costs
P14	Formulae for the table.

Figure 79. Figures for the Specific Design Solution Operationalisation ('Actual').

Appendix H. Operationalisation Formulae

The operationalisations require additional formulae over the built-in Excel primitives. These formulae are in this appendix.

Function GetForm(Reference As String) As String

```
' GETFORM Function to return formula of a text reference
GetForm = Range(Reference).Formula
End Function
```

Function CH(Prev As Range, Curr As Range) As Integer

```
' CH  Function to return 1 if unassigned (—) and changed.
' Used to determine the additional cost of adding,
' but not updating, a structure.
If Prev = "—" And Prev <> Curr Then CH = 1 Else CH = 0
End Function
```

Function CM(Main As Range, Mult As Range) As Integer

```
' CM  Function to add up all values in Mult for which there are
' True values in Main.
CM = 0
I = 1
For Each Col In Main.Columns
  If Col.Value Then CM = CM + Mult.Value(1, I)
  I = I + 1
Next Col
End Function
```

Appendix I. Complete MUSE Application for Cycle 2

Initial SUR(y)

> 'The kitchen is usually a very comfortable room, probably because it has thick walls. However, it can get too hot when *D* is cooking, even in the winter. The room has three radiators that have individual thermostats. These radiators are heated using hot water from a gas-powered combination boiler that is in another room. There is no central thermostat for the boiler, but there is a time-controller and a water temperature controller, neither of which are in the kitchen. The boiler supplies other radiators in the house. There is an extractor fan over the cooker, but it is broken. The windows, which are double-glazed, are difficult to open due to security fittings. An outside door is sometimes opened when the room is too hot. A decrease in the gas bill is desirable.'

The statement of user requirements of the target system are for a bespoke artefact to solve the above problem at a reasonable cost for the envisaged benefits. The user is expected to be *D*, but there is another occupant of the house, *J*. It is not expected that the user will be prepared to spend much time using or learning to use the new artefact. There are other energy management problems in this home, but this application of MUSE does not address them.

There is additional detail given in Appendix D concerning the house, the occupants, the occupants' lives, the heating system, and the current settings of the heating system.

R Construct

The Cycle 2 MUSE products employ the R construct to show no task as the alternative for a selection.

Client Questions and *Answers*

The following questions were put to *D*. Her answers are shown in italics.

First Set of Questions (Before Design)

(Cover sheet included the following unsolicited answer, other than answers to questions)

The heat sources are relatively close together and most of my more complex meals require the use of both, at the same time, towards the end of the cooking which perhaps produces the problems. (Apart from the windows being locked shut!)

1. Is it acceptable to constrain the times of cooking? (Client constraints in SUR(y).)

It is not acceptable to constrain the times of cooking.

2. Under what conditions do you become too hot when cooking?

a. Perhaps when you are cooking something intricate?

Not necessarily. Intricate dishes require activity in preparation, careful timing and planning, and different levels of use of heat sources in the kitchen. Meals which involve four people or more, with two or more courses and use of oven and gas hob generate more heat from the activity and external heat sources than simple cold meals or use of ready-prepared microwaveable dishes for a few people.

b. Wearing particular clothing.

Choice of clothing is dictated by the weather and normal temperature of the environment or by the occasion (e.g. dinner parties) when less suitable clothes are perhaps chosen, to impress or make a statement!

c. At a particular time of the day? During the day or in the evening?

During the day and evening when cooking a main meal for more than few people if the external temperature is hot. During the winter in evening, or perhaps daytime again if preparing a large meal.

d. When cooking for a dinner party?

Often. Complex menus, with several courses requiring precise timing, demand bursts of activity in an environment warmed by the prolonged heating of the oven or use of hob for the main course.

e. When it is warm outside?

Yes, unless preparing simple salad meals, or working slowly preparing meals in advance of a future occasion.

f. Anything else...?

Not to labour the point the chief elements of becoming hot when cooking are:-

- external heat produced by prolonged oven use for roasting, braising, etc.

- external heat from prolonged use of gas rings for stewing, use of pressure cooking, etc.

- bodily heat from activity. beating sauces, stir-frying, for example produce more heat than chopping vegetables.

- bodily heat produced by need for careful timing when preparing complex meals, i.e. several activities may be concurrent, checking oven, stirring sauce, chopping parsley, etc.

- more people = more activity within a defined time period.

3. Under what conditions are you not too hot when cooking?

a) when preparing simple meals at leisure, particularly when requiring only one hot dish.

b) when very cold outside, then kitchen heat is welcome!

c) when little activity is necessary to produce the meal.

d) in summer with door open and easterly breeze.

4. What is the order of events when you cook?

a. When you become too hot?

<u>Roast meal</u>
Light oven
Prepare food to be placed inside
After c. 1/2 hr put food in oven
Leave kitchen
Return to warm kitchen, prepare rest of meal - <u>get hot</u>
Cook rest of food - getting hotter
Serve food and leave kitchen.

<u>Lasagne</u>
Light gas and place three pans on stove
Make pasta sauce, cook pasta and make white sauce
<u>Get hot</u> with activity working over stove.
Light oven for final cooking - <u>get hotter</u>.

<u>Dinner party</u>
- Usually planned as cold starter and probably cold dessert to solve

problem of long pause between courses when hostess disappears!
- Main dish and vegetables hot unless in summer.
- Start early in day, cook starter if necessary—make soup/cook fish for mousse or salad, wash lettuce, etc. Also plan and prepare dessert, sliced oranges, apple pie, check cheeses, etc.
- Start cooking main dish c. 1-2 hrs before arrival of guests. Prepare vegetables, get out cheeses to warm to room temperature, which rises as main dish cooking.
- Dress, check table, return to hot kitchen, check all is well there.
- Greet guests, chat, drink and leave them 1/4 hr before mealtime.
- Rush round in hot kitchen finalising starters, cooking vegetables, checking desserts all ready, etc. Get very hot. Serve starters in dining room. Remove plates, back to cooler kitchen (oven down, everything keeping warm), make gravy, final sauces get hot again, take all to dining room having turned oven and gases off.

Most of the problem I have is that I like everything freshly cooked, not waiting or frozen in advance, which means at the end I am always working in a very hot environment with a need for considerable rapid activity.

b. When you don't become too hot?

- Cooking a meal with only one hot dish e.g. soup followed by cold meat/salad; grilled fish/meat plus salad or jacket potato & salad. Relatively short use of external heat—kitchen stays cool whatever the ancillary cooking activity.

- Cooking meals which have prolonged use of oven or hob at a low heat e.g. casserole dishes, slow stews or meals in one pot.

- Cooking with the microwave generates no external heat in the kitchen.

In all these cases, the food is prepared, perhaps meat is seared or sealed at high heat, then slow cooking produces relatively less heat with time to attend to the rest of the preparation.

Second Set of Questions (During Design)

1. Do you adjust the heating before/during/after cooking?

   No—I do not adjust the heating before, during, or after cooking. Two of the methods of adjustment are outside the room (the thermostat in the bedroom upstairs and overall controller in the porch), which are at some distance, and the thought of changing all the radiator settings does not occur when I am concentrating on cooking.

2. Do you think that you would produce better (or worse) food if you did not become hot while cooking? If so, would you want to produce better food? If so, again, in what ways would you want the food to be better?

It is very possible that the food would be better if I did not become hot while cooking. The physical discomfort possibly leads to taking short cuts or omitting some of the details which might be an improvement. As an enthusiastic cook, I always want to produce the best results which I can, with the materials available. It is difficult to determine the ways in which I would want it to be better—probably best identified as an apparently effortless process without major disasters!

3. Apart from physically being too hot, are you affected in other ways (e.g. mentally or emotionally) when you are too hot while cooking? If so, would you not want to be so affected?

   The effect of the heat induces tension to the process of cooking. Not that I want to leave the kitchen, but that the tension plus the heat complicates the activity. Some of the tension is no doubt generated by uncertainty in trying or experimenting with new forms of cooking in certain circumstances. However, I would prefer not to have the added stress of being too hot.

4. You plan your meals in order to ensure the quality of food. Do you plan your meals to ensure that you do not overheat? Do you plan the heating/cooling consequently (for example, planning to open the back door just before cooking the vegetables)?

   The meal planning is dependent upon the availability of fresh produce, on the numbers to feed and the formality of the occasion. The resulting heat in the kitchen is never considered as part of the planning process—it does not enter my head as a factor to be considered. During the process of cooking, my concentration is usually such that I am much too hot before I realise the fact. As a result, I only open the door during a pause when I become aware that it is very hot, or when I am reminded by others. Opening the door is never part of the planning, it is a consequence of realisation of the heat if I become aware of it.

After First Prototype Evaluation

Changes where incorporated into the design after the first prototype evaluation. The MUSE products presented here are after that evaluation. *D* suggested the following modifications after the first prototype evaluation. Comments by the designer follow each suggested modification.

1. Button on computer to accelerate/retard cooling systems because out of time/guests late.

   Already present [the client did not have a prototype of the cooling controller].

2. Could use section at bottom for other heat sources, e.g. dishwasher which may influence the heat/washing machines in some kitchens/sunlight.

   The dishwasher was run during the evaluation and, perhaps in combination with the lack of the cooling controller prototype, resulted in a brief period of overheating. Will be incorporated into next prototype.

3. 'Warm plates' could be removed—not a major activity. 'Clear kitchen/wash up pans' could be added and 'relaxation'—periods of no activity.

   The order of the activities could be modified as follows:

   Lay table
   Prepare drinks & nibbles
   Get changed
   Greet guests

   The activities need to be grouped together and placed at the bottom of the plan.

   The plates were successfully warmed during evaluation. However, it is suggested that they are removed from the version b prototype, but a check should be made during evaluation that the plates are successfully warmed.

   'Clear kitchen/wash up pans' (shortened to 'Clear kitchen' if insufficient space since wash up pans would be included) and 'Relaxation' will be added.

   The activity plan appears to be ordered by menu (as expected), however the overlay of ordering by activity time does not occur and these standard items do not fit within the menu. They will be moved to the bottom as suggested.

4. Headings might be useful:

   Starter
   Main course
   Vegetables
   Salad
   Dessert
   Cheese
   Coffee

   } for each subsection & would fit layout.

   Activity could be divided into preparation & cooking/assembly.

It is considered that headings and activity division would interfere (as did the standard items) with the 'flow' of the menu and activity plan: the plan would not support a sufficiently wide range of menus and activities. The amount of writing time saved by pre-printing would be low (and not worthwhile by comparison with the reduction in 'flow'). The benefits of the prompting provided by the pre-printing can be had, to some extent, by including the headings in the instructions.

5. An activity level/scale might be useful to determine the programming. This can be assessed for work load.

   The instructions suggest (and the design supports) that the activity line is drawn thicker to indicate greater effort. Discussion with the client suggests that a number to indicate effort would be more visible and easier to produce. THe next prototype will incorporate this suggestion.

8. The activity list notes need to be visible during planning.

   Version a traded the visibility of the activity list during planning against the overall convenience of the meal planner. Evidently, the trade-off was incorrect. The next prototype should determine a different trade-off that makes the activity list notes visible during planning.

It was also noted during the meal preparation that the door was not opened and closed at the planned times, potentially the cause of the brief overheating in the evaluation. The prompt suggested in version a, the post-it notes, was not employed. Version b needs to provide an improved means of prompting.

In general, however, the client: was less hot during the production of the meal; started to make the meal a lot earlier; was less flustered during the production of the meal; was more relaxed at the end of the meal; and was able to re-plan around several meal-changes during production of the meal. The meal tasted good.

MUSE Elements of SUR(y)

The Domain of Application

The domain of application is that of Home Heating Management, a sub-domain of the domain of Home Energy Management, which is in turn a sub-domain of the domains of Home Management and Energy Management, which are in turn both sub-domains of the domain of Management (or Planning and Control).

Technological Constraints

The technology is only constrained by what is available and acceptably priced for its benefits.

Client-Specified Task Constraints

The task is constrained such that *D* must be permitted to cook when she wants to cook.

System Performance Criteria

There are no specific system (controller) performance criteria.

(End-) User Characteristics

The end-users are the occupants of the house. Both are competent users of computers (mainly word-processing). They lead busy lives, however, and do not wish to spend much time using a new system or learning to use a new system.

Environmental Factors

There are no specific environmental factors.

Extant Systems Analysis Stage

Identification of Extant Systems for Analysis

The Current System

The current system is installed in the house of *J* and *D*. The heating system is unusual in that there are two boilers. However, only one of the boilers serves the kitchen, and the user requirements suggest that this single boiler and its controls should be taken as the computer part of the system.

1. A description of the daily use of the system.

Related Systems

2. Other home heating management systems.

   a. The heating system described in SODP1.

   b. Other kitchen users who have different heating systems.

   c. Others who are too hot or too cold when performing a task.

3. Other home energy management systems.

   a. Hot water provision.

   b. Electric heating energy use.

   c. Electric car re-charging from home.

Partially Related Systems

4. Other home management systems.

   a. Alarm clock setting.

   b. Food purchasing and cooking scheduling.

   c. Cleaning scheduling and performing.

5. Other energy management systems.

   a. Apartment-wide energy control.

   b. Office energy control.

   c. Industrial plant energy control.

6. Other management systems.

   a. Security.

   b. Decision Support Systems.

   c. Personal Diaries.

The current system will be analysed initially and others selected afterwards if appropriate.

Extant System System's Analysis of the Current System

Overview

This document contains the analysis of the current extant system. The task descriptions for this current system were developed by questioning and observation.

First Task Description of Current System—TD1(Current)

This product is generalised over three scenarios that were elicited by paper-based questioning. The first scenario (TD1.1(Current); Figure P15) is the preparation of a roast meal, the second (TD1.2(Current); Figure P16) a lasagne, and the third (TD1.3(Current); Figure P17) a dinner party. Figures P18 and 80 show the TD1(Current) diagram and table.

Name	**Description**	**Observation**	**Design Implication**	**Speculation**
'Too hot' meal	Not all meals are too hot. All of the scenarios involved getting too hot.	Some meals are very straightforward or cold and do not involve getting too hot. *D* has a mental plan of early and main cooking activities that is prepared before cooking.	Do not change the current system since doing so may affect meals which are not 'too hot'. Analyse meal planning (see TD2).	Special controls for 'too hot' meals.
Heating is on (in winter)	In winter the heating is controlled by a timer. It is typically on when cooking 'too hot' meals in winter.	Probably the heating in winter and the general heat in summer result in being 'too hot'.	In winter, turn off the heating to prevent overheating. In summer need additional 'cooling'.	Air conditioning would be too expensive.
Early cooking	Many activities are performed before the main period of cooking starts.	No overheating occurs during the early cooking period, due to lesser time pressure or fewer sources of heat.	No need to prevent overheating during early cooking period.	Comparability of activities (at task level) for early cooking and main cooking implies that a solution to overheating for main cooking overheating will provide a solution for early cooking overheating.

Name	Description	Observation	Design Implication	Speculation
Prepare food	Get ingredients ready, chop, slice, etc.	Physical activity generates heat during main cooking period.	'Cooling' should coincide with physical activity.	Difficult to predict. Perhaps planning support with direct control or immediate switch (but time lag?).
Control cooker	Turn cooker on and off	Cooker has four rings, a small oven/grill, and a large oven. Cooker heat plays an important part in the overheating during main cooking period (perhaps from long-term early cooking).	'Cooling' should coincide with cooker heat.	Automatic cooling on cooker switched on? (But, time lag in cooling not equal to time lag in heating.)
Cook food	Grill, fry, etc.	Physical and direct cooker heat in many situations during main cooking period.	Problems of both 'prepare food' and 'control cooker'.	Multiple forms of cooling required?
Cook	Main cooking period that also involves other interleaved tasks, e.g. changing to greet guests.	Overheating occurs during main cooking period.	Need to provide 'cooling' during main heating period.	
Check food	Ensure that food is cooking correctly.	One of many additional non-cooking activities that result in increased pressure, feeling flustered (or stressed), and resulting over-heating. Others are guests, table, serving food...	Ensure sufficiently early start to allow for checking, greeting guests, etc.	Planning aid (both over and within meals?)
Control heating to help reduce overheating	Kitchen door to the garden is opened and closed during main cooking.	The kitchen door is used to regulate the heating in the kitchen. Appears not to be used sufficiently frequently.	Improve use of kitchen door to regulate heating since it is obviously effective. Consider connecting to thermostat with an automatic opener?	Too severe in winter? Prefer regulation of heating in winter? An automatic opener would be expensive (particularly if it needed to maintain security).
Guests invited	With guests there is a need to change and greet the guests.	See check food.		
Check table	Ensure that table is being laid correctly.	Table is usually laid by someone else, however the check increases flustering.		
Serve food	Ensure that the food is finished properly ('extras') and serve.	'Extras' are often missed due to overheating. Resultant dissatisfaction with food.	Elimination of overheating will hopefully reduce dissatisfaction.	

Figure 80. TD1(Current) Table.

Second Task Description of the Current System—TD2(Current)

This product was elicited by observation and questioning (both paper-based and by telephone). Figures P19 and 81 show the diagram and the table.

Name	Description	Observation	Design Implication	Speculation
Meal planning		Meals are planned at different levels of detail: the menu, the shopping list, and the activities.	Improvement in planning activities (particularly start time) should reduce flustering and therefore overheating.	Explicit plan could support direct control of 'cooling' devices: radiator, fan?, door?, ... Computerised meal planning aid? (There is a computer in the home.)
Plan menu and shop	Consider possible foods, the length of time to prepare, and the potential for preparation. Shop with an initial shopping list.	Menu considered at odd moments during the day (not a particular session). Strong familiarity with paper lists. The (fresh) products in the shop will often influence the menu.	Difficult to support menu and shopping list planning.	
Plan cooking	Identify activities to achieve menu and plan their timing and effort.	Mental plan (not made explicit). Wishes could be better (particularly earlier start times).	Support planning cooking. The timing of the activities are critical to reducing flustering, spreading effort, and identifying 'cooling' requirements.	Introduce activity planning stage. Probably not computer based unless no simpler solution since: poor location, lack of familiarity, and a distrust of 'gadgets'. Relate activity plan to cooling requirements.
Cooking	Early and main cooking	Replanning occurs during these periods.	The planning aid must support replanning.	
Post-planning	A certain amount of learning occurs.	Learns: menus which work and the length of time activities take.	Currently inadequate activity planning suggests support requirement. Particularly in terms of starting early enough. i.e. support remembering activity times.	An activity time reference?

Figure 81. TD2(Current) Table.

Domain of Design Discourse of the Current System—DoDD(Current)

Figures P20 and 82 show the diagram and table.

Number	Relationship
1	has a
2	cooks in
3	desires
4	has a
5	does (and can become flustered during)
6	is related to
7	means
8	means not
9	is related to
10	can be
11	can be
12	means
13	means
14	preferably
15	preferably
16	can involve
17	can be
18	can be
19	requires a
20	supporting
21	can be
22	by
23	by
24	controlled by
25	has a
26	has a
27	has
28	related to
29	related to
30	drives
31	related to
32	has a
33	Has
34	has a
35	Changes
36	related to
37	can be
38	can be

Figure 82. DoDD(Current) Table.

General Task Model of the Current System—GTM(Current)

The GTM(Current) (Figure P21) is developed from the TD1(Current) and the TD2(Current). Those elements that are appropriate for porting are in the diagram.

The elements of the TD1(Current) and TD2(Current) tables are not repeated in a table for GTM(Current).

Generalised Task Model Stage

Generalised Task Model for the Extant System Composite—GTM(x)

GTM(x) is the same as GTM(Current) as only one extant system has been analysed currently.

Generalised Task Model of the Target System—GTM(y)

Task Information from SUR(y)

- *D* cooks.
- The heating (and cooling) needs to be controlled (to reduce the overheating) in line with the cooking plan.

Device Independent Summary of Tasks

- *D* cooks.

Figures 83 and 84 show the structured diagram and table.

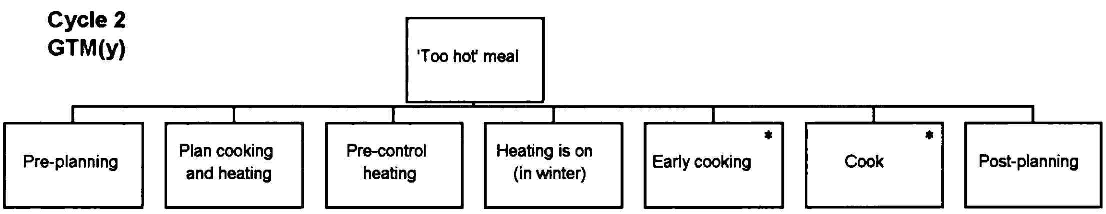


Figure 83. GTM(y) Structured Diagram.

Name	Description	Design Rationale
'Too hot' meals	The problem does not occur for all meals	From SUR(y)
Pre-planning	Plan menu and shopping.	From extant system.
Plan cooking and heating.	The heating (and cooling) needs to be controlled (to reduce the overheating) in line with the cooking plan.	From SUR(y) and extant system.
Pre-control heating	The heating (and cooling) needs to be controlled.	From SUR(y) and extant system. Control the heating in advance of cooking if possible.

Name	Description	Design Rationale
Heating is on (in winter)	The heating is typically on when cooking 'too hot' meals in winter.	From extant system. Needs to be turned off in the kitchen at appropriate times to prevent overheating. Turning off is to be considered in 'Control heating'.
Early cooking	Activities before main period of cooking starts.	From extant system.
Cooking	Main cooking period.	From SUR(y) and extant system. Must support replanning and heating-control if necessary.
Post-planning	Learning.	From extant system.

Figure 84. GTM(y) Table.

Statement of User Needs Stage

Statement of User Needs—SUN(y)

- See SUR(y) for the target system requirements.
- Explicit design constraints from SUR(y): the amount of fuel used cannot increase very much and should decrease.
- Implicit design constraints from SUR(y): the cost of the new system should not be very high.
- Explicit system performance criteria from SUR(y): *D* must not be too hot ('overheat').
- Implicit system performance criteria from SUR(y): *D* must be able to cook the meals that she desires when she desires.
- Existing system results in *D* overheating when cooking some meals and has resulted in this requirement for redesign.
- Existing system supports other uses (for example, cooking meals that do not result in *D* becoming too-hot or others cooking meals) and must not have functionality removed (or overcomplicated).
- Feedback on all commands should be provided.
- Relevant guidelines include: 'In complex or unfamiliar contexts, remind or prompt the user to think about the kinds of model or plan that will be useful.' (Gardner and Christie, 1987); and 'The technique implemented is a powerful one, giving the user many degrees of freedom and control.' (Baecker and Buxton, 1987).

- The system should have a consistent interface.
- Transfer of learning from the extant current system (and other extant systems) should be used where possible.
- Interface Environment: PC Windows (because there is such a system in the house) and Maplin Electronics.

Domain of Design Discourse for the Target System —DoDD(y)

In this case, the DoDD(y) is the same as the DoDD(current), i.e. the DoDD for the current extant system.

Composite Task Model Stage

Composite Task Model—CTM(y)

Figures P22 and 85 show the structured diagram and table. For table entries that correspond with entries in the GTM(y) please consult the GTM(y) table.

Name	Description	Design Rationale
Plan menu and shop	Consider possible foods, the length of time to prepare, and the potential for preparation. Shop with an initial shopping list.	From extant system. Difficult to support menu planning better than currently supported.
Plan cooking	Identify activities to achieve menu and plan their timing and effort.	From extant system. Support to reduce future flustering and provide a basis for planning heating.
Identify activities	Activities required to produce the menu (e.g. make white sauce, etc.)	From extant system.
Plan activity timing and effort	Ensure that the activities are achievable within the time period.	From extant system. Current is very implicit. Make more explicit to ensure more accurate timing and permit assessment of effort.
Plan heating	Identify when to provide 'cooling' to reduce overheating.	Necessary to plan 'cooling' since cooking activities take priority while cooking (thus preventing later planning).
Identify sources	Identify sources of heat.	Different sources of heat might be overlooked.
Identify activity heat	Identify heat due to effort in performing an activity.	Difficult to assess without an explicit plan of activity heat.
Identify cooker heat	Identify heat due to cooker ring, oven, etc.	Timing might be difficult to assess without an explicit plan of cooker heat (e.g. timing of white sauce making).
Identify other heat sources	Identify heat due to dishwasher, weather, etc.	A catch-all. Requirement identified during evaluation of version a when dishwasher caused additional heating.

Name	Description	Design Rationale
Plan heating to help reduce overheating	Identify when possible 'cooling' should apply.	'Cooling' involves opening the door, providing cooling air to the cooking area, and turning off the radiators (in winter). The cooling air should be a direct fan rather than repairing the hob fan (which only extracts and does not cool).
Door open	Plan when to cool by opening the door.	
Assess heat for door	Assessment of the causes of heat that require the cooling of the door: activity heat, cooker heat, and other heat.	The door should be opened just before high levels of activity, if the cooker is producing a lot of heat, or the weather is hot.
Plan door open times	Make explicit the plan to open the door.	Closed is the normal state of the door, open is not.
Fan on	Plan when to cool by turning on a fan.	The fan tackles the direct heat from the cooker rings.
Assess heat for fan	Assessment of the causes of heat that require the cooling of the fan: cooker ring heat.	The fan should be on when two or more of the cooker rings are in use.
Plan fan on times	Make explicit the plan to have the fan on.	Off is the normal state of the fan, on is not.
Radiators off	Plan when to cool by turning off the radiators.	
Assess heat for radiators	Assessment of the causes of heat that require the cooling of the radiators: activity heat, cooker heat, and other heat.	The radiators should not be on when: the oven is on; more than three rings of the cooker are on; and/or there is a high level of activity.
Plan radiators on times	Make explicit the plan to have the radiators on.	1) 'On' (i.e. under thermostatic control) is the normal state of the radiators (in winter), off is not. 2) 'On' refers to heating and 'off' to cooling for the radiators whereas 'on' refers to cooling and 'off' to not-cooling for the fan. However, to maintain consistency with the normal terminology the plan should make explicit when the radiators are on rather than off.
Pre-control heating	Submit the heating plan to a controller.	Off-load the control of the heating to a device.
Control fan, radiators, and door.	Pre-control the fan, radiators, and door.	The fan and radiators can be electronically controlled more easily (cheaply) than the door. The door is to be manually controlled with supported reminder—a requirement identified during the evaluation of version a when the door was not opened or closed sufficiently according to the plan.
Early cooking	Perform activities before main period of cooking starts.	From extant system.
Replan cooking and heating	The cooking and heating plan can change.	From extant system. The change in the cooking plan identified from the extant system can give rise to a change in the heating plan.
Prepare food	Get ingredients ready, chop, slice, etc.	From extant system.
Control cooker	Turn cooker on and off.	From extant system.
Cook food	Grill, fry, etc.	From extant system.
Check food	Ensure that food is cooking correctly.	From extant system.
Control heating to help reduce overheating	Control kitchen door, fan, and radiator according to the plan.	From extant system. In addition, control of the fan and radiators.
Open and close the door	Open and close kitchen door according to the plan.	From extant system with support to ensure that the door is controlled.
Remind to open or close door	Support to ensure that the door is controlled.	Off-loaded to a controller.
Control fan, radiators, and door	Turn on and off fan and radiators (in winter) according to the plan. Permit changes to the plan.	Off-loaded to a controller.
Guests invited	With guests there is a need to change and greet the guests.	From extant system.
Check table	Ensure that table is being laid correctly.	From extant system.

Name	Description	Design Rationale
Serve food	Ensure that the food is finished properly ('extras') and serve.	From extant system.
Post-planning	A certain amount of learning occurs.	From extant system.
Remember activity times		From extant system. Support.
Forget detailed plans		From extant system. Needs to be supported for explicit plan.

Figure 85. CTM(y) Table.

Event List

Figure 86 shows the event list.

Event	Summary	Attributes	Instances
Heating	The heating has a current state	- state of heating	'Radiators on', 'Radiators off', 'Door open', 'Door closed', 'Fan on', 'Fan off'.
Meal	Meal has various states.	- state of meal	'Plan menu & shop', 'Plan cooking', 'Early cooking—prepare food', etc.
Cooking plan	*D* maintains a cooking plan representing her current intentions for cooking activities	- criteria - projected meal states	'Plan cooking', 'Early cooking', 'Cooking', etc.
Heating plan	*D* maintains a heating plan representing her current intentions for heating control.	- criteria - projected heating states	'Plan heating', 'Pre-control heating', etc.

Figure 86. Event List.

Functions List

Figure 87 shows functions list.

Functions	Trigger	End result	Performance
Represent cooking activity timing and effort	*D* (re-)planning activity timing and effort.	Activity timing and effort represented.	Immediate and low cost.
Represent heating plan	*D* (re-)planning heating, *D* controlling fan and radiators	Heating plan represented.	Immediate and low cost.
Control of fan and radiators	*D* controlling fan and radiators.	Fan and radiators will turn on and off at control times.	Immediate and low cost.
Control of door	*D* controlling door	Reminder to open/close the door	Immediate and low cost.

Figure 87. Functions List.

System and User Task Model Stage

System Task Model of the Target System—STM(y)

The diagram (Figure P23) only shows a decomposition from the CTM(y) and does not repeat the contents of the CTM(y). Figure 88 shows the table.

Name	Description	Design Rationale
Activity plan clear?	Is the activity plan clear (i.e. ready for another use)? The activity plan supports an explicit representation of the activities, their timing, their effort, and their consequences on heating (the heating plan).	Activity plan: explicit representation of plan should reduce fluster, support (and incorporate) explicit heating plan, and therefore reduce overheating. The activity plan is represented on a device. Clearing the activity plan prepares it for another use.
Clear activity plan	Clear the activity plan for another use.	
H: Request activity plan cleared	Request that the activity plan is cleared ready for another use.	*D* knows when about to plan a new meal.
C: Show activity plan cleared	Show that the activity plan is clear and ready for another use.	Provide feedback that the plan is clear.
Note menu	Represent menu in activity plan.	The menu is represented in the activity plan to support the elicitation of the activities. (Re-planning must be supported.)
H: Note menu on activity plan	Input menu to activity plan on device.	*D* knows the menu (or has it on a piece of paper).
C: Show menu on activity plan	Show the menu on the activity plan.	Feedback and explicit representation of menu.
Note activities	Represent the activities in the activity plan.	
H: note activities on activity plan	Input activities to activity plan on device.	*D* can remember and reason suitable activities to prepare the menu.
C: Show activities on activity plan	Show the activities on the activity plan.	Feedback and explicit representation of activities.
Not sure of time? Think in activity list?	*D* may not be able to remember the overall time an activity takes. If so, she should check an activity list.	Activity list: explicit, searchable, representation (aide-memoire) of the normal length of time taken by an activity. It is important to get the length of time as correct as possible to reduce flustering (including by starting sufficiently early) and plan the heating at the correct moments. Currently the normal overall time for an activity is poorly remembered.
C: Show time in activity list	Support searching the activity list and show the time for an activity.	Device supported activity time store and display.
H: View time in activity list	View time searched for.	
H: Note overall time	Input the normal overall time for an activity to the activity plan.	Supports planning of start(s) and duration(s) in cooking plan.
C: Show overall time	Show the normal overall time for an activity in the activity plan.	Feedback and explicit representation to support cooking plan.
H: Note start, duration, and effort	Input the start(s), duration(s), and effort(s) required to perform an activity.	The cooking plan as reasoned by *D* made explicit.
C: Show start, duration, and effort	Show the start(s), duration(s), and effort(s) required to perform an activity.	Feedback and explicit representation of cooking plan.
Assess rings in use/use of oven/use of grill	Determine the heat coming from the use of the cooker.	Cooker provides heat that affects the heating plan.
H: Note rings in use/use of oven/use of grill	Input the use of the cooker and the extent of its heat to the activity plan.	*D* can reason the use of the cooker from the cooking plan and the extent of its heat.

Name	Description	Design Rationale
Assess use of other heat sources	Determine the heat coming from the use of other heat sources.	Other heat sources can affect the overall temperature.
H: Note use of other heat sources	Input the use/provision of other heat sources.	*D* can reason the use of other heat sources, e.g. the dishwasher, or the current weather conditions, etc.
C: Show rings in use/use of oven/use of grill	Show the heat coming from the use of the cooker.	Feedback and explicit representation of planned cooker heat.
C: Show use of other heat sources	Show the heat coming from other heat sources, e.g. dishwasher, weather, etc.	Feedback and explicit representation of use/provision of other heat sources.
Estimate heat according to effort	Determine the heat coming from the amount of *D*'s effort.	*D* generates heat that affects the heating plan.
C: Show effort	Show the effort required to perform an activity.	Already explicitly represented in the activity plan.
H: View effort	View effort in the activity plan.	Difficult to remember the effort for each period of cooking time (with various activities occurring).
H: View rings in use/use of oven/use of grill	View heat coming from the use of the cooker.	Difficult to remember the heat coming from the cooker for each period of cooking time.
H: Note when to open door/set fan to be on/set radiators to be on	Input when to open door/set fan on/set radiator on to the activity plan.	*D* can assess the amount of cooling required given the 'effort' heat, cooker heat, outside temperature, her clothing, etc.
C: Show when to open door/set fan to be on/set radiators to be on	Show when to open door/set fan on/set radiator on.	Feedback and explicit representation of the heating plan.
Set time	The controller must know the current time relative to the cooking time periods to control the heating.	Consistency with representation in activity plan important (cooking time periods).
C: Show cooking times	Show cooking time periods.	Representation of cooking time in the plan.
H: Set current cooking time	Input the current cooking time relative to the cooking plan.	*D* transfers from plan. 'Bubbled up' rationale since two devices for cost and convenience. So need to transfer cooking time from one device with the plan to the 'heating controller'.
C: Show current cooking time	Show the current cooking time relative to the cooking plan.	Feedback and explicit representation of the timing for the heating plan (for control).
Control fan	The controller must know the plan for the fan.	Two devices rationale.
Control radiators (in winter)	The controller must know the plan for the radiators	Two devices rationale.
C: Show when to set fan to be on/radiators to be on/door to be opened/closed	Show the plans for the fan/radiators /door	Two devices rationale.
H: Set fan/radiators/door start and duration	Input the plans for the fan/radiators/doors.	*D* transfers the plans. Two devices rationale.
C: Shown when fan/radiators/door set to be on or opened/closed	Show the plans for the fan/radiators/door.	Feedback and explicit representation of the fan, radiators, and door plans (for control or reminding). Two devices rationale.
C: Remind to open or close door	Remind *D* to open or close the door.	*D* needs reminding. The reminder must not be too insistent or intrusive as to disturb cooking apart from prompting to open or close the door.
Update activity lists	Bring activity lists up to date.	From extant system.
Activity in list?	Check the activity list for the activity item.	Search of the activity list.
H: Adjust activity time if necessary	Input the normal overall time for the activity if different from that in the list.	*D* can assess activity times and compare with entry in list easily.

Name	Description	Design Rationale
C: Show any adjustment in activity time	Show normal overall time for the activity.	Feedback.
H: Note activity and time in list	Input the activity and its time.	
C: Show activity and time in list	Show normal overall time for the activity.	Feedback.

Figure 88. STM(y) Table.

User Task Model of the Target System—UTM(y)

The structured diagram (Figure P24) only shows a decomposition from the CTM(y) and does not repeat the contents of the CTM(y). These descriptions are mostly ported from the extant current system.

The UTM(y) has the following content, format, or mode of presentation issues for the STM(y), ITM(y), etc:

- The noted length of time relates to the overall length of time in the STM(y) activity plan and activity list. The time is usually in hours and minutes and very approximate. The words 'hrs' and 'mins' will usually be used.
- The menu is usually listed in the order of the meal.

Interaction Task Model Stage

Interaction Task Model of the Target System—ITM(y)

Target user interface environment

The computer system is to be composed of two devices. The first is a paper-based planning aid (activity and heating) and memory aid (activity). The second is a hard-wired device in a plastic container; to be consistent with existing controllers and be of low cost.

The set of interface objects for the first device are those shapes that can be drawn and printed relatively easily using a drawing editor and pens to write and draw on paper.

The set of interface objects for the second device will be those intended for manipulation by humans that are defined in a Maplin catalogue.

The structured diagram (Figure P25) only shows a decomposition from the STM(y) and does not repeat the contents of the STM(y). The screen construct shows the activation point of screens and not the consumption point. Figure 89 shows the table.

Name	Description	Design Rationale
Wipe activity plan with damp cloth	Wipe the plasticated activity plan clear using a damp cloth.	Plastic covering and thin water soluble markers permits cleanability, for re-planning and cleanliness in a food environment.
Write menu items	Write the menu into the activity plan using the pens	Using stiff paper for the activity plan enables it to be: carried around, both in the kitchen and out; and stood-up in the kitchen. It is lightweight and available. The possibility of computer-supporting the planning was rejected due to: having a computer in the kitchen where planning mainly occurs (currently a computer is located elsewhere in the house, but it is needed there); and *D*'s dislike of 'gadgets'.
Mistake? (Move or remove)	It is possible to make mistakes on the activity plan, either of the menu item or of the position of the menu item.	(Later re-planning can be supported with a different colour pen to ensure that changes are clear.)
Wipe out menu item	Wipe the menu item(s) from the plasticated plan using a damp cloth.	Plasticated plan & water soluble pens permit changing of plan.
Write activity items	Write the activity items necessary to achieve the menu on the activity plan.	
Mistake? (Move or remove)	It is possible to make mistakes, either of the activity item or the position of the activity item.	
Wipe out activity item	Wipe the activity item(s) from the plasticated plan using a damp cloth	See above.
Locate activity in lists	Look through the lists of activities to find the activity.	The activity list should be handy for the activity planner. A good way of keeping them together is to have the activity list on the back of the activity plan (since both need to be of a reasonable size, probably A3). Probably a mistake to ensure sorting since a linear search will be quite quick and sorting might limit the number of items and be more untidy. An open space does still allow sorting if desired. If on the back, then will be plasticated which is required for changing the activity times.
Look at activity time	Read the activity time beside the activity.	
Write figures corresponding to overall time	Write the activity time beside the activity.	
Position pen at start	Place pen at start time of activity.	A line can indicate the start and duration of the activity very clearly. However, during the evaluation of version a the effort was not adequately conveyed by the use of a thicker line to indicate more effort. Version b uses numbers along the row to indicate start, duration, and effort. The effort is a number between 1 and 9.
For each duration column	Move across the columns for the duration.	
Write figures corresponding to effort	Write a number between 1 and 9 to indicate the effort involved in performing the activity at that time.	The number means that the total effort at any time can easily be assessed by looking down a time column.
Move pen for duration	Draw a line to indicate the duration of the activity.	

Name	Description	Design Rationale
For each ring in use	Up to four rings can be in use at any time.	
Position pen over ring icon	The activity plan contains an icon for each ring. Place the pen over this icon.	
Fill in ring icon	Draw over the icon. Shade more heavily for more heat.	The amount of heat coming from the rings can easily be assessed by looking at the ring icons.
Position pen over oven icon	The activity plan contains an icon for the oven. Place the pen over this icon.	
Fill in oven icon	Draw over the icon. Shade more heavily for more heat.	The amount of heat coming from the oven can easily be assessed by looking at the oven icon.
Position pen over grill/small oven icon	The activity plan contains an icon for the grill/small oven. Place the pen over this icon.	
Fill in grill/small oven icon	Draw over the icon. Shade more heavily for more heat.	The amount of heat coming from the grill/small oven can easily be assessed by looking at the grill/small oven icon.
Look down time column on activity plan	Look down the time column on the activity plan and total the numbers indicating the effort for each activity. A higher total indicates greater effort in any time period.	Easy to assess the effort in a time period.
Look at ring icons	Look at the ring icons to assess the heat from the rings in a time period.	Easy to assess the heat from the rings in a time period.
Look at oven icon	Look at the oven icon to assess the heat from the oven in a time period.	Easy to assess the heat from the oven in a time period.
Look at grill/small oven icon	Look at the grill/small oven icon to assess the heat from the grill/small oven in a time period.	Easy to assess the heat from the grill/small oven in a time period.
Position pen at start	Place pen at start time of opening door/turning on the fan/turning on the radiator.	A line can indicate when to open the door/etc.
Move pen for duration	Draw a line to indicate the duration of the door being open/etc.	
Until time light shows time b4 zero	Until the light above the time list shows a time relative to the zero location.	Permits rapid change in 'plan' shifting once the plan is under way.
Too early	If the time light is currently too early.	
Too late	If the time light is currently too late.	
Press move later button	Press the button that moves the light 'later'.	
Press move earlier button	Press the button that moves the light 'earlier'.	
For each on-off period	The fan/radiators may be turned on and off more than once.	
For each open-close period	The door may be opened and closed more than once.	
Press duration	Put on the duration of the fan/radiators	Important to maintain consistency between the fan and radiators interface, even though radiators usually 'on' (in winter) and fan usually 'off'. Toggle buttons enable changes to be made if plan is updated.
Wipe out activity time	Wipe the activity time in the activity list using a damp cloth.	Enables activity time to be updated.
Write in new activity time	Write the activity time beside the activity in the activity list.	
Write activity in list	Write the activity in the activity list.	
Write activity time in list	Write the activity time beside the activity in the activity list.	

Figure 89. ITM(y) Table.

Interface Model Stage

Interface Model of the Target System—IM(y)

Figures P26 and 90 show the structured diagram and table

Name	Description	Design Rationale
Underlying grid and headings only	Show only the underlying grid and headings.	Clear does not mean totally blank. Only means removal of last written plan. Underlying grid and headings printed onto paper under plasticated surface.
Menu words in 'Menu' column	Show the menu in the 'Menu' column.	Written on by *D*.
Activity words in 'Activity' col.	Show the activity in the 'Activity' column.	Written on by *D*.
Time for each activity in 'Time' col.	Show the time for each activity in the 'Time' column.	Written on by *D*.
Figures indicating effort from start time col. to end time col.	Show the times and effort for activity.	Written on by *D*. After evaluation of version a.
Highlight rings in use	Show the heat from the rings in use.	Drawn on by *D*.
Highlight oven in use	Show the heat from the oven in use.	Drawn on by *D*.
Highlight grill in use	Show the heat from the small oven/grill in use.	Drawn on by *D*.
Write code/descr. for heat source in use	Write a code or description (e.g. DW for dishwasher) for the heat source in use.	Written on by *D*. After evaluation of version a.
Total of figures in time column for activity rows	Show all the figures for the efforts on activies at time.	Written on by *D*.
Write intended cooking time above cols.	Write the target time above the zero time on the sheet, and (optionally) fill in the other times).	Written on by *D*.
Line in time column, 'Door' row	Show the line for door open.	Drawn on by *D*.
Line in time column, 'Fan' row	Show the line for fan on.	Drawn on by *D*.
Line in time column, 'Rads' row	Show the line for radiators on.	Drawn on by *D*.
Time in 'Time' column for activity row	Show overall time for activity in activity list.	Written on by *D*.
Activity in 'Activity' column	Show activity in activity list.	Written on by *D*.
Show time in activity list	Show overall time for activity in activity list.	Written on by *D*.
Light LED indicating current time	The LED above the current (relative) time is lit.	Corresponds to the activity plan.
Light LEDs from start for duration	Light the lamps to indicate when the fan and radiators are on (and off).	Corresponds to the activity plan.
Sound buzzer	Sound a non-intrusive yet noticeable buzzer.	After evaluation of version a.

Figure 90. IM(y) Table.

Display Design Stage

Pictorial Screen Layout of the Target System—DD(y)-PSL(y)

Figures P27, P28, P29 and P30 show the diagrams. The ordering of the prompts and spaces in the activity plan and activity list were influenced by the Gardner and Christie (1989) guideline: 'Present information in the order in which it will be used'.

Dictionary of Screen Objects of the Target System—DD(y)-DSO(y)

Figure 91 shows screen 1 (the activity plan) Figure 92 shows screen 2 (the activity list), Figure 93 shows screen 3 (the instructions), and Figure 94 shows screen 4 (the controller).

Screen object	Description	Design attributes
Activity plan	A plasticated printed plan that can be written on using water-soluble pens.	Supports representing activity plan and heating plan.

Figure 91. DSO(y) Screen 1 Table.

Screen Object	Description	Design Attributes
Activity list	A plasticated printed list that can be written on using water-soluble pens.	Supports representing activities and their normal overall time.

Figure 92. DSO(y) Screen 2 Table.

Screen Object	Description	Design Attributes
Instructions	A plasticated list of instructions for the activity plan and activity list. Can be wiped clean.	Supports the use of the activity plan, activity list, and controller.

Figure 93. DSO(y) Screen 3 Table.

Screen Object	Description	Design Attributes
LED (Maplin UK20W or equivalent).	A small high-intensity focussed red light. Displays the programmed state of the radiators, fan, and door for each time period.	Lit or extinguished showing the programmed state of something.
Toggle button (Maplin XC32 or equivalent).	A small positive push button that toggles states. Changes the programmed state of the radiators or fan for each time period.	Can be pressed to change the programmed state of something.

Screen Object	Description	Design Attributes
Push button (Maplin ?? or equivalent)	A small positive push button. Changes the current time identifier.	Can be pressed to move the current time indicator earlier or later. Shaped like an arrow.
Buzzer (Maplin ?? or equivalent)	A non-intrusive but noticeable buzzer.	Buzzing or not buzzing to indicate whether the door should be opened/closed or not.

Figure 94. DSO(y) Screen 4 Table.

Dialogue and Error Table of the Target System—DD(y)-DET(y)

There are no error messages.

Display and Inter-Task Screen Actuation Diagram—DD(y)-DITaSAD(y)

Not considered necessary for this design as the screens are not consumed.

Appendix J. Full Cycle 2 Operationalisation.

Appendix Introduction

The appendix contains the formal and metricated Cycle 2 operationalisation.

Current Solution Operationalisation

Specific Actual Performance

The planning specific **actual performance** is operationalised as the union of the planning specific actual quality and the planning specific actual costs. The planning **worksystem boundary criteria** are operationalised by the requirement that the constituents of the planning worksystem have the common goals of the current (level of) achievement and satisfaction of the planning of the cooking of *A* and the heating of *A*. The planning **domain boundary criteria** are operationalised by the requirement that the constituents of the planning domain of application express the current (level of) achievement and satisfaction of these common goals.

The control specific **actual performance** is operationalised as the union of the specific actual quality and the specific actual costs. The control **worksystem boundary criteria** are operationalised by the requirement that the constituents of the control worksystem have the common goals of the current (level of) achievement and satisfaction of the control of the cooking of *A* and the heating of *A* in the kitchen of *A* using the kitchen's cooker, radiators, and door. The control **domain boundary criteria** are operationalised by the requirement that the constituents of the control domain of application express the current (level of) achievement and satisfaction of these common goals.

Specific Actual Quality and Costs

There is one main sub-system in the planning worksystem: the planner (*A*). There are two main sub-systems in the control worksystem: the user (*A*) and the cooker.

Figure 95 shows the figures that contain the current operationalisation. The **actual costs** are operationalised by the union of the actual resource costs in the tables.

The link between the planning worksystem and the control domain is an explicit case where the quality of the control (target) is dependent on the planning worksystem.

Figure	Contents
P31	Planning domain and worksystem diagram.
P32	Control domain and worksystem diagram.
P33	Table (containing the values) for the planning and control quality and costs.
P34	Formulae for the table.

Figure 95. Figures for the Current Solution Operationalisation ('Current').

Specific Design Problem Operationalisation

The specific design problem operationalisation is aiming for a minimal expression, which is achieved by using quality and costs statements that are with respect to the current operationalisation.

Specific Desired Quality

The main task goals are to maintain the state of *A*'s comfort attribute as 'comfortable', *A*'s agitation attribute as 'not agitated', and the meal's quality attribute as 'good'."

$\forall e(PlanQuality \neq FALSE)$

$\forall e(ComfortTq \neq FALSE)$
$\forall e(Agitation/time < 35\%)$
$Final(MealQuality) > 8$

Specific Desired Costs

The physical structural costs of the heating system should be within a range that allows for this desirable decrease or acceptable increase in gas and electricity usage. It is assumed that the heating system can be upgraded and, therefore, the operationalisation of the physical and abstract structural costs of the heating system should be within a range that allows for a different installation and maintenance price. Further it is expected that a small increase in physical and abstract behavioural costs of the heating system would be tolerated and this would be reflected in the operationalisation within a range.

Physical structural costs of the heating system:

$\forall e(CC : PhysStruct \leq 38))$

Abstract structural costs of the heating system:

$\forall e(CC : AbstStruct \leq 100))$

Physical and abstract behavioural costs of the heating system:

$\forall e(CC : AbstBeh \leq 120)$
$\forall e(CC : PhysBeh \leq 34)$

It is assumed that the user costs either remain the same, and be operationalised in the same manner—for example the user physical structural costs—, or decrease if possible, and be operationalised to be within a range—for example the user physical behavioural costs.

$\forall e(PA : AbstStruct \leq 91)$
$\forall e(PA : PhysStruct \leq 2)$
$\forall e(PA : AbstBeh \leq 214)$
$\forall e(PA : PhysBeh \leq 5)$
$\forall e(CA : AbstStruct \leq 256)$
$\forall e(CA : PhysStruct \leq 65)$
$\forall e(CA : AbstBeh \leq 382)$
$\forall e(CA : PhysBeh \leq 70)$

Specific Design Solution Operationalisation

Specific Actual Performance

The planning specific **actual performance** is operationalised as the union of the planning specific actual quality and the planning specific actual costs. The planning **worksystem criteria** are operationalised by the requirement that the constituents of the planning worksystem have the common goals of the actual (level of) achievement and satisfaction of the planning of the cooking of A and the heating of A. The planning **domain boundary criteria** are operationalised by the requirement that the constituents of the planning domain of application express the actual (level of) achievement and satisfaction of these common goals.

The control specific **actual performance** is operationalised as the union of the control specific actual quality and the control specific actual costs. The control **worksystem criteria** are operationalised by the requirement that the constituents of the planning worksystem have the common goals of the actual (level of) achievement and satisfaction of the control of the cooking of *A* and the heating of *A* in the kitchen of *A* using the kitchen's cooker, radiators, and door. The control **domain boundary criteria** are operationalised by the requirement that the constituents of the control domain of application express the actual (level of) achievement and satisfaction of these common goals.

The Specific Actual Quality and Costs

There are two main sub-systems in the planning worksystem: the planner (*A*) and the planning-aid. There are four main sub-systems in the control worksystem: the user (*A*), the cooker, the door, and the fan.

Figure **96** shows the figures that contain the solution operationalisation. The **actual costs** are operationalised by the union of the actual resource costs in the tables.

Figure	**Contents**
P35	Planning domain and worksystem diagram.
P36	Control domain and worksystem diagram.
P37	Table (containing the values) for the planning and control quality and costs
P38	Formulae for the table.

Figure 96. **Figures for the Specific Design Solution Operationalisation ('Actual').**

See Volume 11

For Figures on Back Pocket

Towards Engineering Principles for Human-Computer Interaction (Domestic Energy Planning and Control)

Jeremy Adam Joseph Stork

University College London

Submitted for the Degree of Doctor of Philosophy
at the University of London

June 1999

List of Figures in Pocket on Back Cover

Figure P1

Cycle 1 MUSE TD1(Current)
Structured Diagram

Cycle 1
TD1(c)

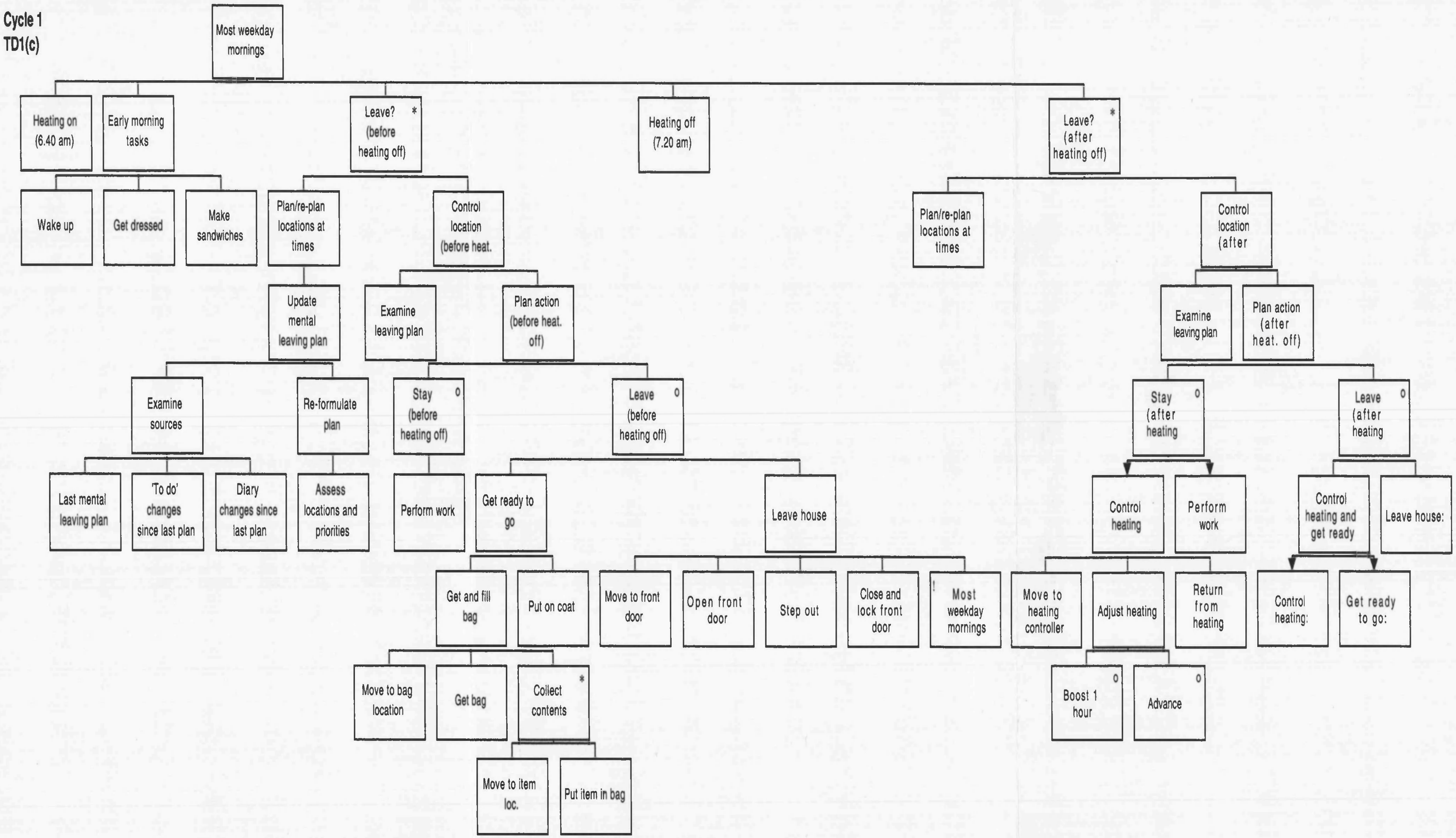

Most weekday mornings
Heating on (6.40 am)
Early morning tasks
Leave? (before heating off) *
Heating off (7.20 am)
Leave? (after heating off) *
Wake up
Get dressed
Make sandwiches
Plan/re-plan locations at times
Control location (before heat.
Plan/re-plan locations at times
Control location (after
Update mental leaving plan
Examine leaving plan
Plan action (before heat. off)
Examine leaving plan
Plan action (after heat. off)
Examine sources
Re-formulate plan
Stay (before heating off) o
Leave (before heating off) o
Stay (after heating o
Leave (after heating o
Last mental leaving plan
'To do' changes since last plan
Diary changes since last plan
Assess locations and priorities
Perform work
Get ready to go
Leave house
Control heating
Perform work
Control heating and get ready
Leave house:
Get and fill bag
Put on coat
Move to front door
Open front door
Step out
Close and lock front door
Most weekday mornings
Move to heating controller
Adjust heating
Return from heating
Control heating:
Get ready to go:
Move to bag location
Get bag
Collect contents *
Boost 1 hour o
Advance o
Move to item loc.
Put item in bag

Figure P2

Cycle 1 MUSE TD2.1(Current)
Structured Diagram

Cycle 1
TD2.1(c)

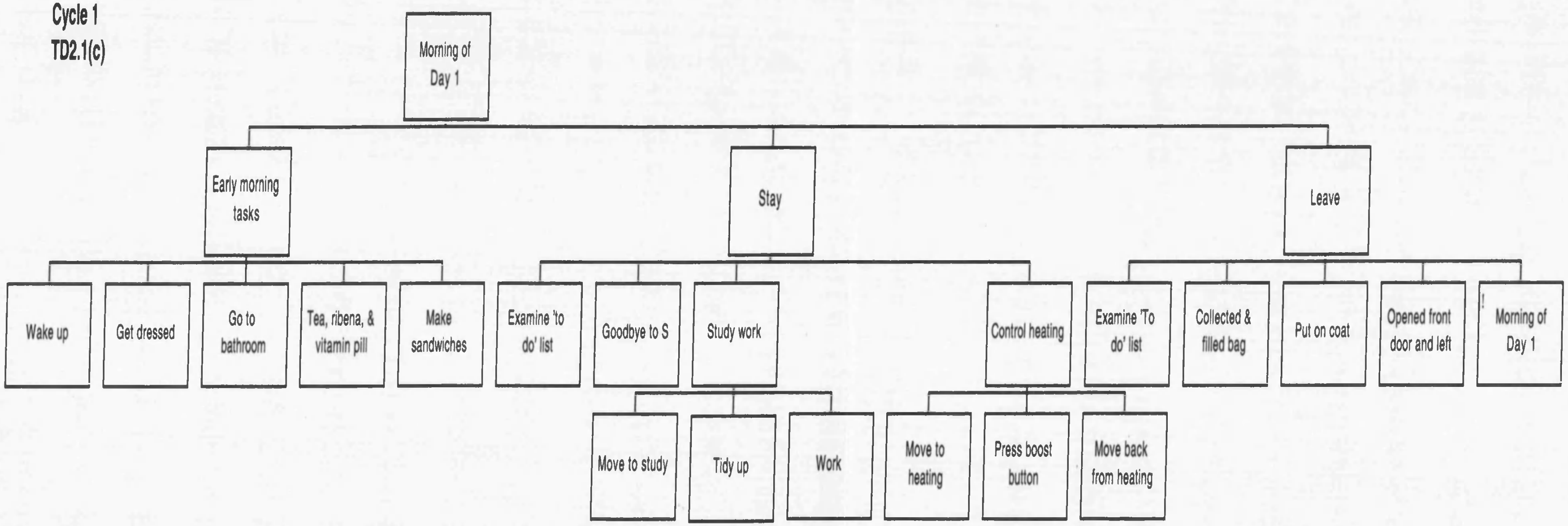


Figure P3

Cycle 1 MUSE TD2.3(Current)
Structured Diagram

Cycle 1
TD2.3(c)

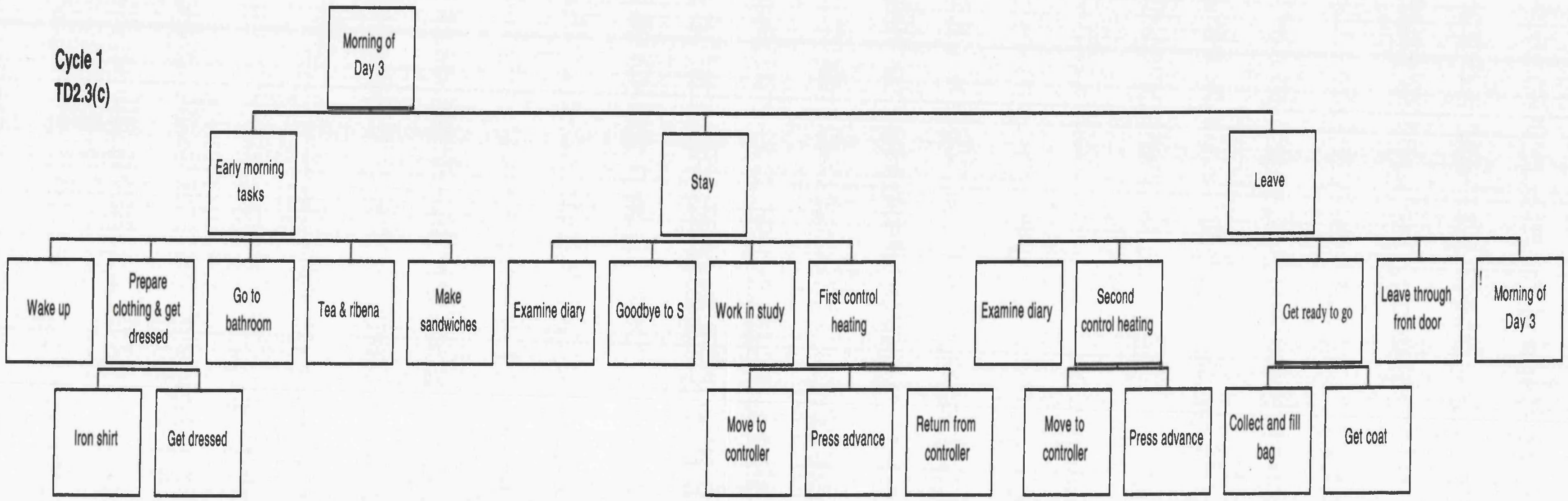


Figure P4
Cycle 1 MUSE TD2(Current)
Structured Diagram

Cycle 1
TD2(c)

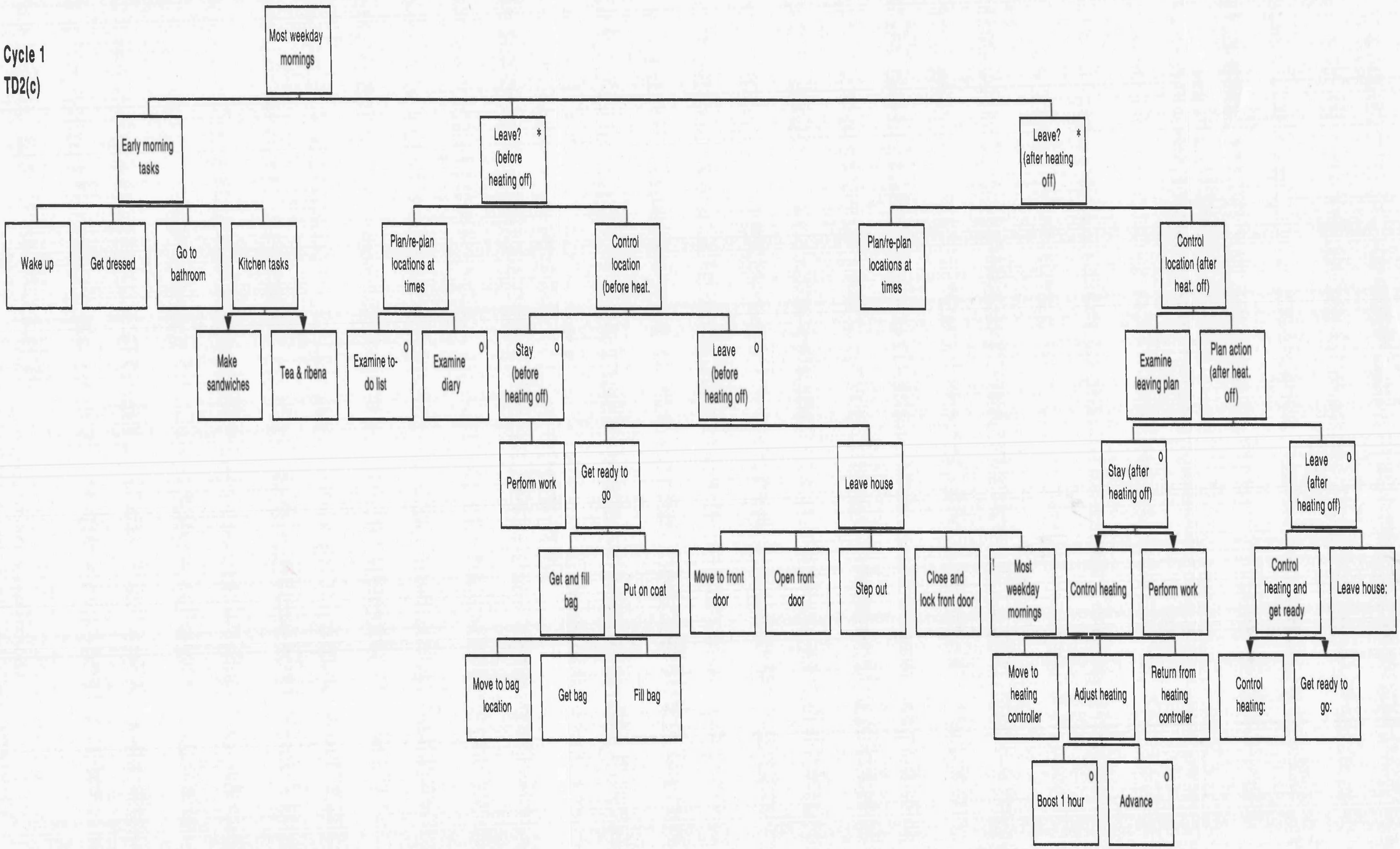
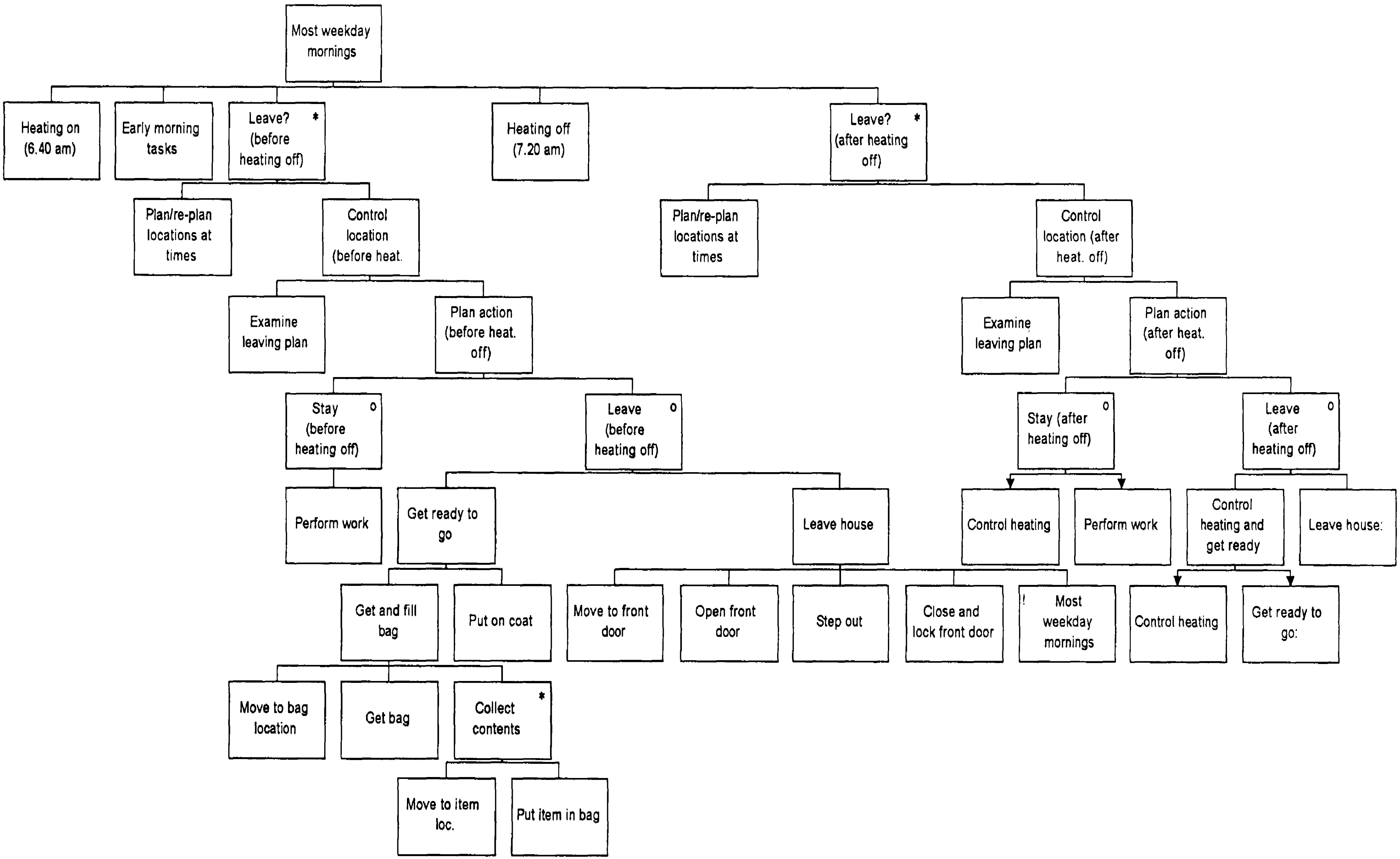

Most weekday mornings
Early morning tasks
Leave? (before heating off) *
Leave? (after heating off) *
Wake up
Get dressed
Go to bathroom
Kitchen tasks
Plan/re-plan locations at times
Control location (before heat.
Plan/re-plan locations at times
Control location (after heat. off)
Make sandwiches
Tea & ribena
Examine to-do list o
Examine diary o
Stay (before heating off) o
Leave (before heating off) o
Examine leaving plan
Plan action (after heat. off)
Perform work
Get ready to go
Leave house
Stay (after heating off) o
Leave (after heating off) o
Get and fill bag
Put on coat
Move to front door
Open front door
Step out
Close and lock front door
! Most weekday mornings
Control heating
Perform work
Control heating and get ready
Leave house:
Move to bag location
Get bag
Fill bag
Move to heating controller
Adjust heating
Return from heating controller
Control heating:
Get ready to go:
Boost 1 hour o
Advance o

Figure P5

Cycle 1 MUSE GTM(Current)
Structured Diagram

Cycle 1
GTM(c)

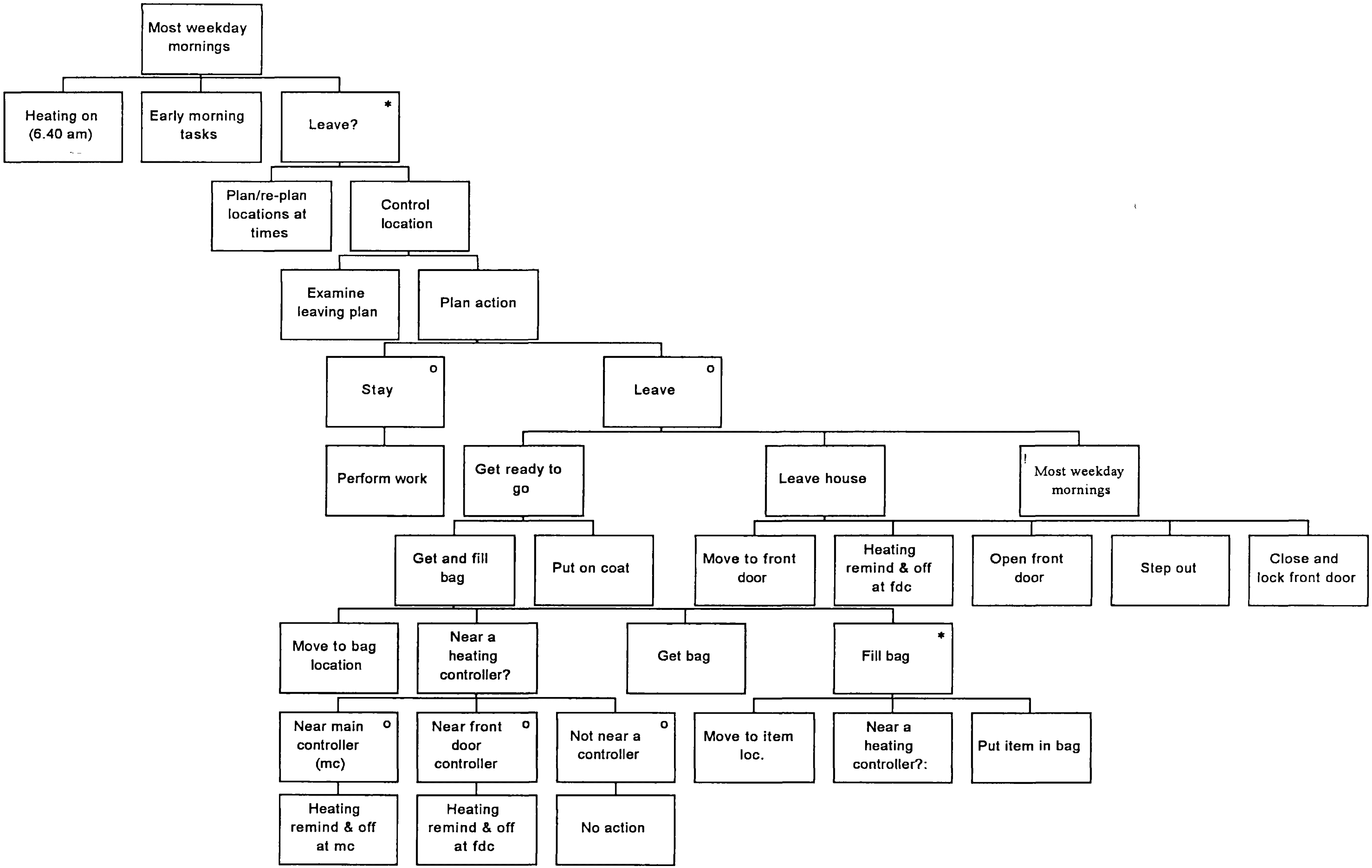


Most weekday mornings
Heating on (6.40 am)
Early morning tasks
Leave? (before heating off) *
Heating off (7.20 am)
Leave? (after heating off) *
Plan/re-plan locations at times
Control location (before heat.
Plan/re-plan locations at times
Control location (after heat. off)
Examine leaving plan
Plan action (before heat. off)
Examine leaving plan
Plan action (after heat. off)
Stay (before heating off) o
Leave (before heating off) o
Stay (after heating off) o
Leave (after heating off) o
Perform work
Get ready to go
Leave house
Control heating
Perform work
Control heating and get ready
Leave house:
Get and fill bag
Put on coat
Move to front door
Open front door
Step out
Close and lock front door
! Most weekday mornings
Control heating
Get ready to go:
Move to bag location
Get bag
Collect contents *
Move to item loc.
Put item in bag

Figure P6

Cycle 1 MUSE CTM(y)
Structured Diagram

Cycle 1
CTM(y)



Most weekday mornings
Heating on (6.40 am)
Early morning tasks
Leave? *
Plan/re-plan locations at times
Control location
Examine leaving plan
Plan action
Stay o
Leave o
Perform work
Get ready to go
Leave house
Most weekday mornings
Get and fill bag
Put on coat
Move to front door
Heating remind & off at fdc
Open front door
Step out
Close and lock front door
Move to bag location
Near a heating controller?
Get bag
Fill bag *
Near main controller (mc) o
Near front door controller o
Not near a controller o
Move to item loc.
Near a heating controller?:
Put item in bag
Heating remind & off at mc
Heating remind & off at fdc
No action

Figure P7

Cycle 1 Operationalisation
Current Planning Diagram

Operationalisation 1
Current Planning

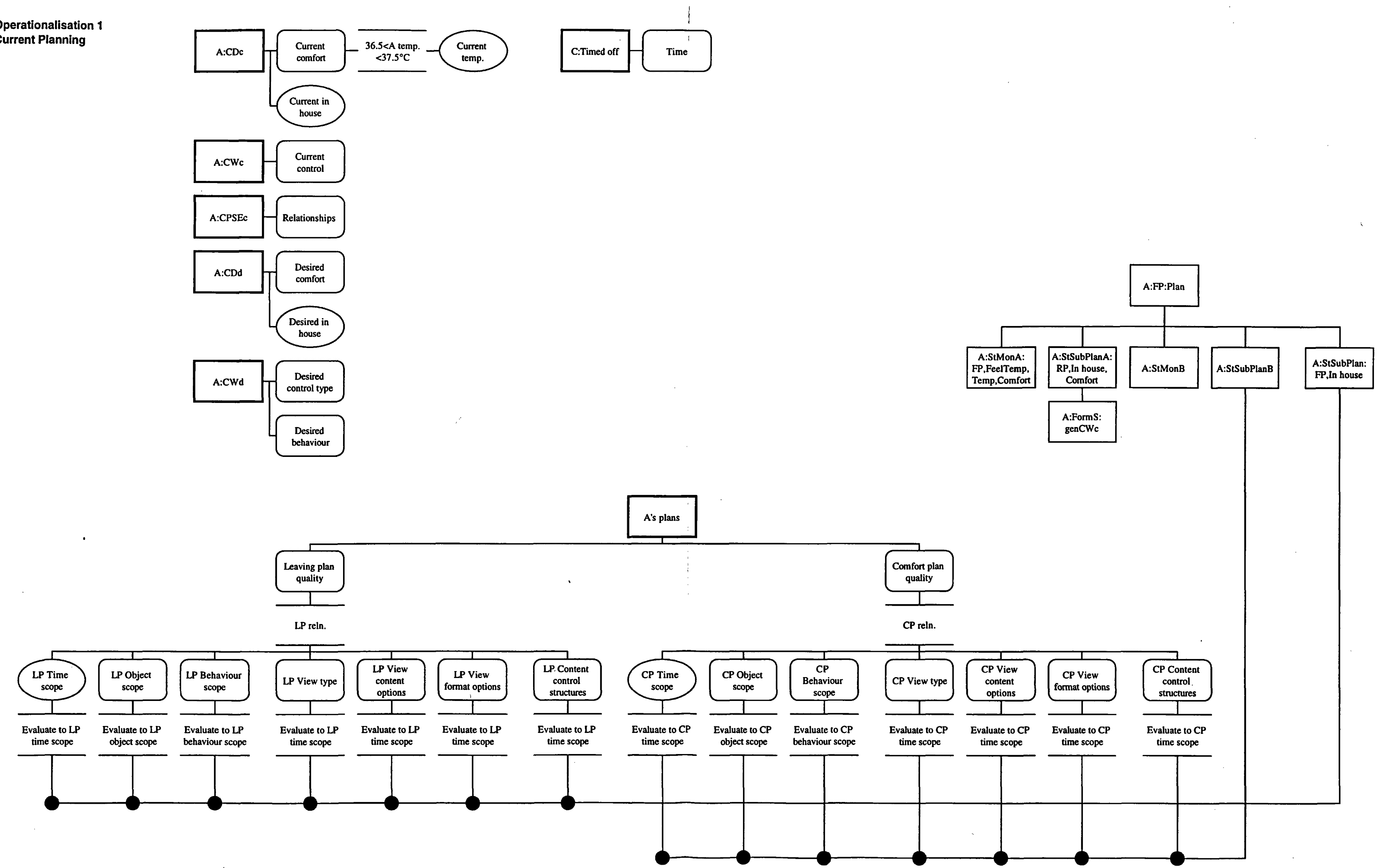


Figure P8

Cycle 1 Operationalisation

Current Control Diagram

Operationalisation 1
Current Control

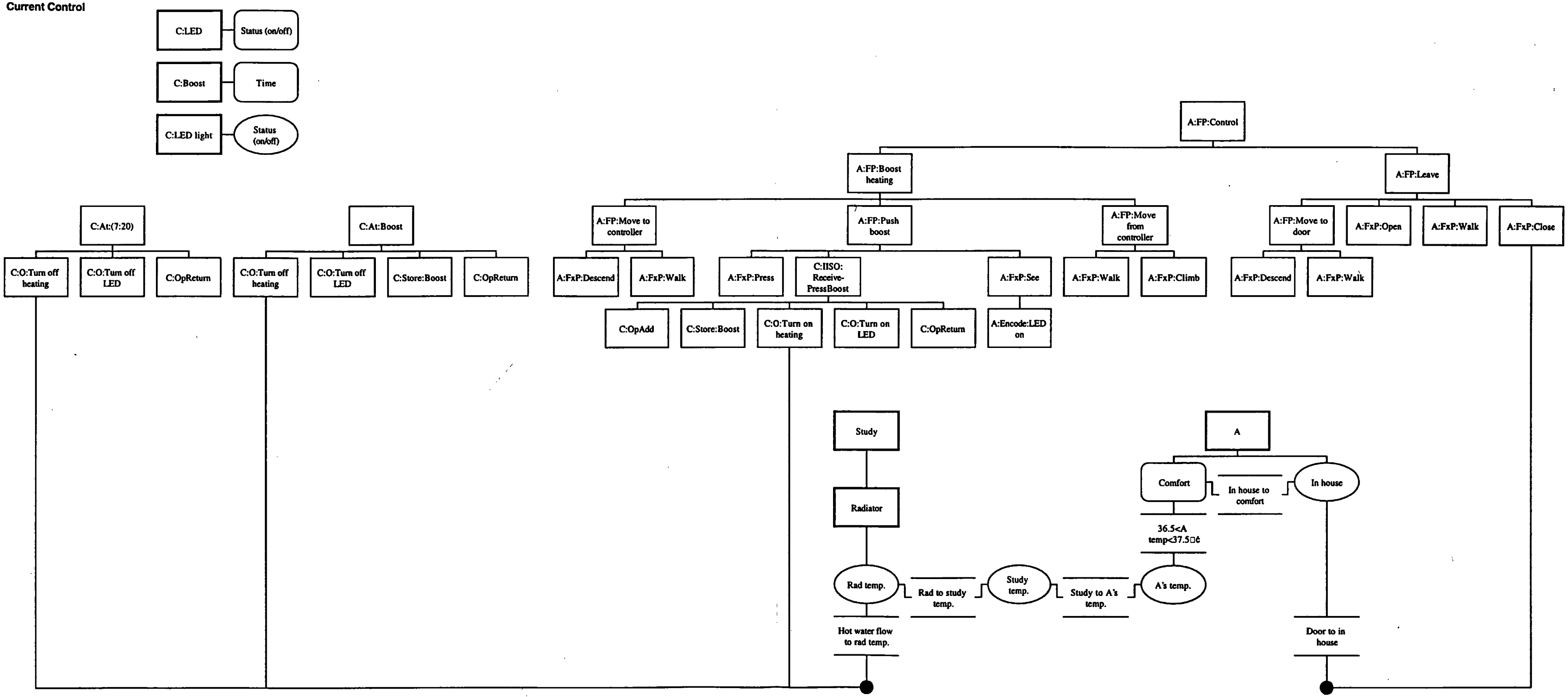


Figure P9
Cycle 1 Operationalisation
Current Planning and Control Table

Cycle 1 Operationalisation Current Table

Event nick	Time	Elapsed time	Diff. time	A-P: Plan	A-SiWork: FP, FeelTemp, Temp, Comfort	A-SiSubPlan: FP, In house, Comfort	A-Sim: genOk	A-SiNoid	A-SiSubPlan	A-SiSubPlan: FP, In house	A-CDe: Current temp.	A-CDe: Current comfort	A-CDe: Current in house	A-CNe: Current control	A-CPSE: Relationships	A-CDe: Desired comfort	A-CDe: Desired in house	A-CNe: Desired control	A-CNe: Desired behaviour	C-Timed off
1	06:20	00:00	00:10	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	6	[illegible]	TRUE	Timed on	Boost heating->Comfort=True; Leave->In house=False	[illegible]	[illegible]	[illegible]	[illegible]	07:20
2	06:30	00:10	00:10	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	6	[illegible]	TRUE	Timed on	Boost heating->Comfort=True; Leave->In house=False	[illegible]	[illegible]	[illegible]	[illegible]	07:20
3	06:40	00:20	00:10	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	6	[illegible]	TRUE	Timed on	Boost heating->Comfort=True; Leave->In house=False	[illegible]	[illegible]	[illegible]	[illegible]	07:20
4	06:50	00:30	00:10	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	6	[illegible]	TRUE	Timed on	Boost heating->Comfort=True; Leave->In house=False	[illegible]	[illegible]	[illegible]	[illegible]	07:20
5	07:00	00:40	00:10	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	6	[illegible]	TRUE	Timed on	Boost heating->Comfort=True; Leave->In house=False	[illegible]	[illegible]	[illegible]	[illegible]	07:20
6	07:10	00:50	00:10	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	6	[illegible]	TRUE	Timed on	Boost heating->Comfort=True; Leave->In house=False	[illegible]	[illegible]	[illegible]	[illegible]	07:20
7	07:20	01:00	00:10	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	6	[illegible]	TRUE	Timed on	Boost heating->Comfort=True; Leave->In house=False	[illegible]	[illegible]	[illegible]	[illegible]	07:20
8	07:30	01:10	00:10	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	FALSE	[illegible]	[illegible]	TRUE	Timed on	Boost heating->Comfort=True; Leave->In house=False	[illegible]	[illegible]	[illegible]	[illegible]	07:20

Figure P10

Cycle 1 Operationalisation
Current Planning and Control
Formulae

Cycle 1 Operationalisation Table Formulae (1/1)

Current.xls

	Loc	Heading	Formula in the 11th row
1	A11		
2	B11		3
3	C11		0.277777777777778
4	D11		=C11-$C$9
5	E11		=C11-C10
6	F11	A:FP: Plan	=AND(HOUR(C11)=8,MINUTE(C11)=20,COUNTIF($F$9:F10,TRUE)=0)
7	G11	A:StMonA: FP, FeelTemp, Temp, Comfor	=F10
8	H11	A:StSubPlanA: RP, In house, Comfort	=G10
9	I11	A:FormS: genCWc	=H10
10	J11	A:StMonB	=I10
11	K11	A:StSubPlanB	=J10
12	L11	A:StSubPlan: FP, In house	=AND(HOUR(C11)=10,MINUTE(C11)=0,COUNTIF($L$9:L10,TRUE)=0)
13	M11	A:CDc: Current temp.	=IF(G11,36.5,M10)
14	N11	A:CDc: Current comfort	=IF(G11,M11<36.5,N10)
15	O11	A:CDc: Current in house	=O10
16	P11	A:CWc: Current control	=IF(J11,"Timed off",P10)
17	Q11	A:CPSEc: Relationships	=Q10
18	R11	A:CDd: Desired comfort	=IF(G11,TRUE,R10)
19	S11	A:CDd: Desired in house	=CHOOSE(COUNTIF($K$10:K11,TRUE)+COUNTIF($L$10:L11,TRUE)+1,"ó",TRUE,FALSE)
20	T11	A:CWd: Desired control	=IF(K11,"Boost",T10)
21	U11	A:CWd: Desired behaviour	=CHOOSE(COUNTIF($K$10:K11,TRUE)+COUNTIF($L$10:L11,TRUE)+1,"ó","Boost heating","Leave")
22	V11	C:Timed off	0.305555555555556
23	W11	A:Abst-struct	=W10+CH(M10,M11)+CH(N10,N11)+CH(R10,R11)+CH(S10,S11)+CH(T10,T11)+CH(U10,U11)
24	X11	A:Phys-struct	1
25	Y11	A:Abst-beh	=Y10+CM(F11:L11,$F$5:$L$5)
26	Z11	A:Phys-beh	=Z10+CM(F11:L11,$F$6:$L$6)
27	AA11	C:Abst struct	=AA10
28	AB11	LP Time scope	=IF(L11,0.2,"ó")
29	AC11	LP Object scope	=IF(L11,"Leaving","ó")
30	AD11	LP Behaviour scope	=IF(L11,"A:Form:Leave","ó")
31	AE11	LP View type	=IF(L11,"Internal","ó")
32	AF11	LP View Content options	=IF(L11,"None","ó")
33	AG11	LP View format options	=IF(L11,"None","ó")
34	AH11	LP Content control structures	=IF(L11,"In house=False; A:Form:Leave","ó")
35	AI11	LP quality	=OR(AND(AB11="ó",AC11="ó",AD11="ó",AE11="ó",AF11="ó",AG11="ó",AH11="ó"),AND(AB11=0.2,AC11="Leaving",AD11="A:Form:L
36	AJ11	CP Time scope	=IF(K11,0.2,"ó")
37	AK11	CP Object scope	=IF(K11,"Comfort","ó")
38	AL11	CP Behaviour scope	=IF(K11,"A:Form:Boost heating","ó")
39	AM11	CP View type	=IF(K11,"Internal","ó")
40	AN11	CP View content options	=IF(K11,"None","ó")
41	AO11	CP View format options	=IF(K11,"None","ó")
42	AP11	CP Content control structures	=IF(K11,"Comfort=True; A:Form:Boost heating","ó")
43	AQ11	CP quality	=OR(AND(AJ11="ó",AK11="ó",AL11="ó",AM11="ó",AN11="ó",AO11="ó",AP11="ó"),AND(AJ11=0.2,AK11="Comfort",AL11="A:Form:B
44	AR11	Plan quality	=AND(AI11,AQ11)
45	AS11	C:At: (7:30)	=AND(HOUR(C11)=7,MINUTE(C11)=30,COUNTIF($AS$9:AS10,TRUE)=0)
46	AT11	C:O:Turn off heating	=OR(AS10,AW10)
47	AU11	C:O:Turn off LED	=AT10
48	AV11	C:OpReturn	=OR(AND(AU10,COUNTIF($AU$9:AU11,TRUE)=1),BI10,AND(AX10,COUNTIF($AX$9:AX11,TRUE)=2))
49	AW11	C:At: Boost	=AND(HOUR(C11)=9,MINUTE(C11)=21,COUNTIF($AW$9:AW10,TRUE)=0)
50	AX11	C:Store Boost	=OR(BG10,AND(AU10,COUNTIF($AU$9:AU11,TRUE)=2))
51	AY11	A:FP: Control	=K10
52	AZ11	A:FP: Boost heating	=AY10
53	BA11	A:FP: Move to controller	=AZ10
54	BB11	A:FxP: Descend	=OR(BA10,BO10)
55	BC11	A:FxP: Walk	=OR(BB10,BL10,BP10)
56	BD11	A:FP: Push boost	=AND(BC10,COUNTIF($BC$9:BC11,TRUE)=1)
57	BE11	A:FxP: Press	=BD10
58	BF11	C:IISO: Receive press boost	=BE10
59	BG11	C:OpAdd	=BF10
60	BH11	C:O: Turn on heating	=AND(AX10,COUNTIF($AX$9:AX11,TRUE)=1)
61	BI11	C:O: Turn on LED	=BH10
62	BJ11	A:FxP: See	=AND(AV10,COUNTIF($AV$9:AV11,TRUE)=2)
63	BK11	A:Encode: LED	=BJ10
64	BL11	A:FP: Move from controller	=BK10
65	BM11	A:FxP: Climb	=AND(BC10,COUNTIF($BC$9:BC11,TRUE)=2)
66	BN11	A:FP: Leave	=L10
67	BO11	A:FP: Move to door	=BN10
68	BP11	A:FxP: Open	=AND(BC10,COUNTIF($BC$9:BC11,TRUE)=3)
69	BQ11	A:FxP: Close	=AND(BC10,COUNTIF($BC$9:BC11,TRUE)=4)
70	BR11	A:LED:Status	=IF(BK11,TRUE,BR10)
71	BS11	(Rad 'on')	=IF(OR(AT11,BH11),AND(NOT(AT11),BH11),BS10)
72	BT11	C:Gas	=0.03*BS11*MINUTE(E11)
73	BU11	C:Boost: Time	=IF(AX11,IF(COUNTIF($AX$9:AX11,TRUE)=1,C11+TIME(1,0,0),"ó"),BU10)
74	BV11	C:LEDlight: Status	=IF(OR(AU11,BI11),AND(NOT(AU11),BI11),BV10)
75	BW11	A:Abst-struct	=BW10+CH(BR10,BR11)
76	BX11	A:Phys-struct	=BX10
77	BY11	A:Abst-beh	=BY10+CM(AS11:BQ11,$AS$5:$BQ$5)
78	BZ11	A:Phys-beh	=BZ10+CM(AS11:BQ11,$AS$6:$BQ$6)
79	CA11	C:Abst-struct	=CA10+CH(BU10,BU11)+CH(BV10,BV11)
80	CB11	C:Phys-struct	=CB10+BT11
81	CC11	C:Abst-beh	=CC10+CM(AS11:BQ11,$AS$7:$BQ$7)
82	CD11	C:Phys-beh	=CC10+CM(AS11:BQ11,$AS$8:$BQ$8)
83	CE11	(Poss. change)	=2*MINUTE(E11)
84	CF11	Rad temp	=IF(BS11,MIN(CF10+CE11,85),MAX(CF10-CE11,10))
85	CG11	(From Rad)	=CG10+0.02*(CF11-CF10)*MINUTE(E11)
86	CH11	Room temp	=IF(CG11>14,CG11,CG11+(4-0.5*(CG11-6)))
87	CI11	A's temp	=35+(CH11*0.1)
88	CJ11	Comfort Tq	=OR(AND(CI11>36.5,CI11<37.5),NOT(CL11))
89	CK11	(Discomfort time)	=IF(CJ11,0,MINUTE(E11)+CK10)
90	CL11	In house	=IF(BQ11,FALSE,CL10)
91	CM11		

Figure P11

Cycle 1 Operationalisation

Actual Planning Diagram

Operationalisation 1
Actual Planning

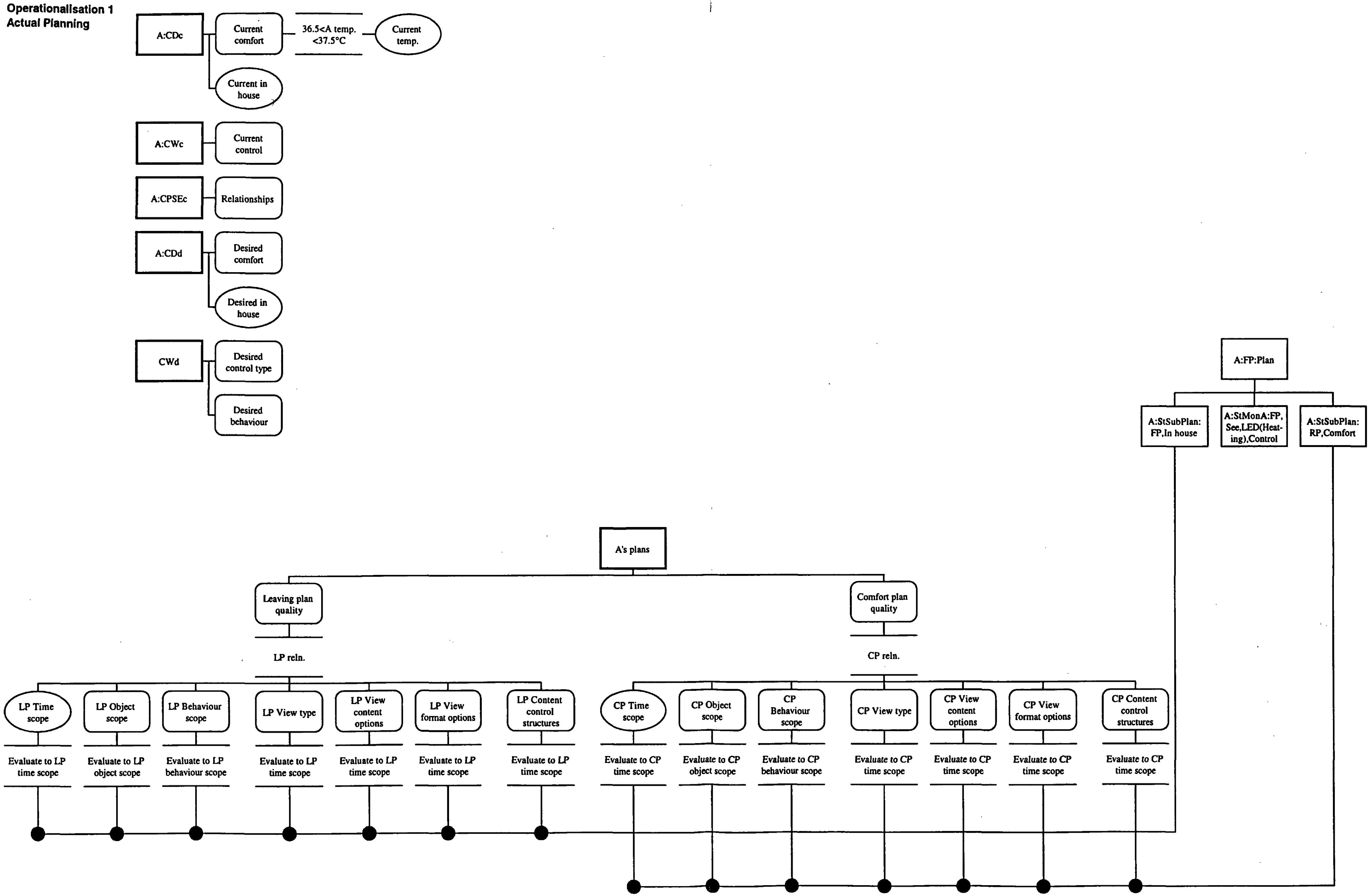


Figure P12

Cycle 1 Operationalisation

Actual Control Diagram

Operationalisation 1
Actual Control

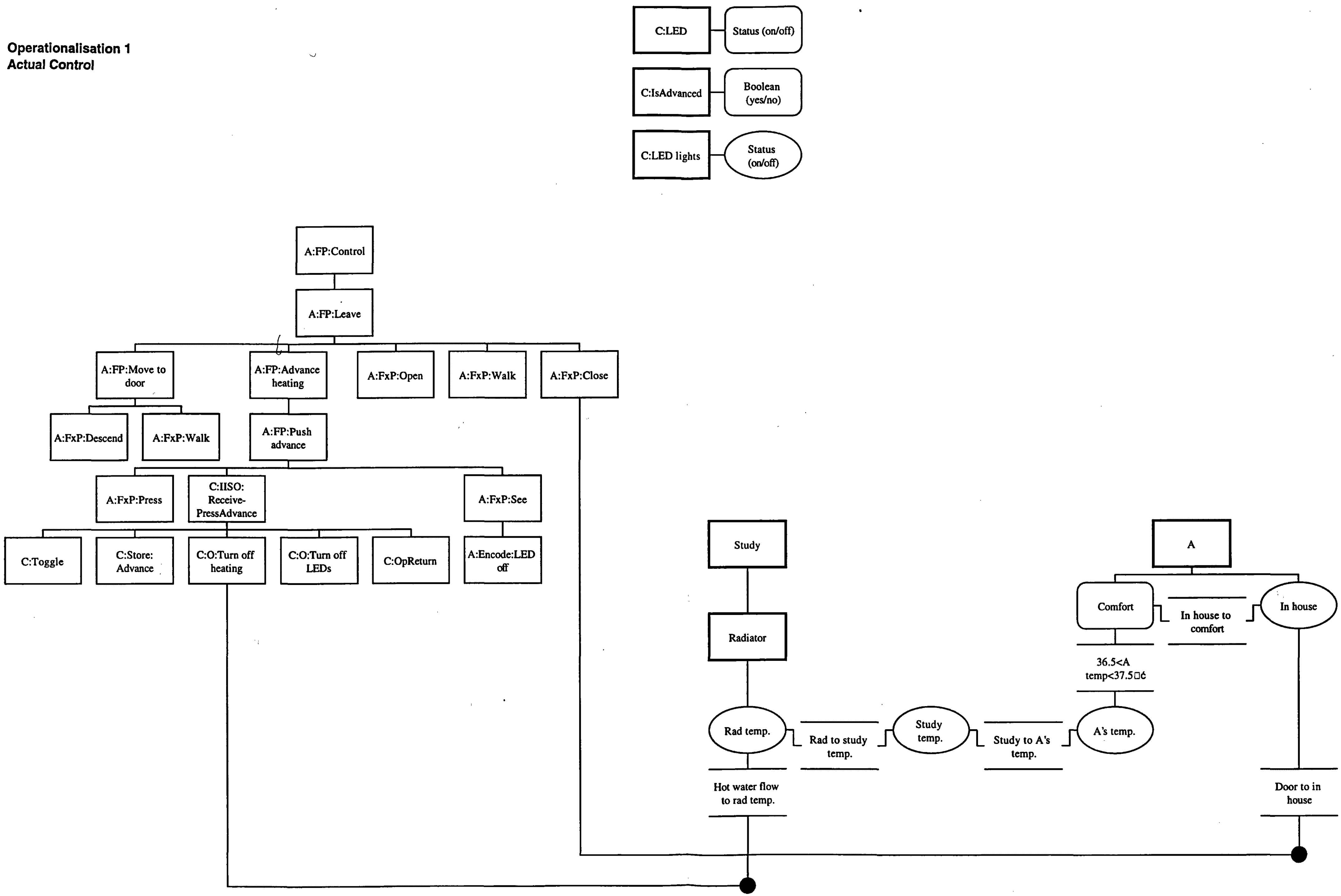


Figure P13
Cycle 1 Operationalisation
Actual Planning and Control Table

Cycle 1 Operationalisation Actual Table

Event No.	Time	Elapsed time	Diff. time	Planning processes				Planning representations								Planning costs				Planning quality	Control processes	Control representations	Control costs	Control quality
				A:P: Plan	M:SolePlan: FP In house	M:SWork: TP See LED/Heat-int,Control	M:SolePlan: FP Comfort	A:CDe: Current comfort	A:CDe: Current In house	A:CDe: Current control	U:CPSE: Relationships	A:CDe: Nested comfort	A:CDe: Desired In house	A:CDe: Nested control	A:CDe: Desired behaviour	U:Abst-struct	M:Phys-struct	U:Abst-beh	U:Phys-beh	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Figure P14

Cycle 1 Operationalisation
Actual Planning and Control
Formulae

Cycle 1 Operationalisation Table Formulae (1/1)

Actual.xls

	Loc	*Heading*	*Formula in the 11th row*
1	A11		
2	B11		3
3	C11		0.277777777777778
4	D11		=C11-$C$9
5	E11		=C11-C10
6	F11	A:FP: Plan	=AND(HOUR(C11)=10,MINUTE(C11)=0,COUNTIF($F$9:F10,TRUE)=0)
7	G11	A:StSubPlan: FP,In house	=F10
8	H11	A:StMonA: FP,See,LED(Heat- int),Contro	=AND(AQ10,COUNTIF($AQ$9:AQ10,TRUE)=1)
9	I11	A:StSubPlan: RP,Comfort	=H10
10	J11	A:CDc: Current comfort	TRUE
11	K11	A:CDc: Current in house	TRUE
12	L11	A:CWc: Current control	=IF(H11,"Timed on",L10)
13	M11	A:CPSEc: Relationships	Advance heating->Comfort=True; Leave->In house=False
14	N11	A:CDd: Desired comfort	=IF(I11,TRUE,N10)
15	O11	A:CDd: Desired in house	=IF(G11,FALSE,O10)
16	P11	A:CWd: Desired control	=IF(I11,"Advance",P10)
17	Q11	A:CWd: Desired behaviour	=CHOOSE(COUNTIF($G$10:G11,TRUE)+COUNTIF($I$10:I11,TRUE)+1,"ó","Leave","Advance heating")
18	R11	A:Abst-struct	=R10+CH(L10,L11)+CH(N10,N11)+CH(O10,O11)+CH(P10,P11)+CH(Q10,Q11)
19	S11	A:Phys-struct	=S10
20	T11	A:Abst-beh	=T10+CM(F11:I11,$F$5:$I$5)
21	U11	A:Phys-beh	=U10+CM(F11:I11,$F$6:$I$6)
22	V11	LP Time scope	=IF(G11,0.2,"ó")
23	W11	LP Object scope	=IF(G11,"Leaving","ó")
24	X11	LP Behaviour scope	=IF(G11,"A:Form:Leave","ó")
25	Y11	LP View type	=IF(G11,"Internal","ó")
26	Z11	LP View Content options	=IF(G11,"None","ó")
27	AA11	LP View format options	=IF(G11,"None","ó")
28	AB11	LP Content control structures	=IF(G11,"In house=False; A:Form:Leave","ó")
29	AC11	LP quality	=OR(AND(V11="ó",W11="ó",X11="ó",Y11="ó",Z11="ó",AA11="ó",AB11="ó"),AND(V11=0.2,W11="Leaving",X11="A:Form:Leave",Y11="Internal",Z11="None",AA11="None",AB11="In house=False; A:Form:Leave"))
30	AD11	CP Time scope	=IF(I11,0.2,"ó")
31	AE11	CP Object scope	=IF(I11,"Comfort","ó")
32	AF11	CP Behaviour scope	=IF(I11,"A:Form:Advance heating","ó")
33	AG11	CP View type	=IF(I11,"Internal","ó")
34	AH11	CP View content options	=IF(I11,"None","ó")
35	AI11	CP View format options	=IF(I11,"None","ó")
36	AJ11	CP Content control structures	=IF(I11,"Comfort=True; A:Form:Advance heating","ó")
37	AK11	CP quality	=OR(AND(AD11="ó",AE11="ó",AF11="ó",AG11="ó",AH11="ó",AI11="ó",AJ11="ó"),AND(AD11=0.2,AE11="Comfort",AF11="A:Form:Advance heating",AG11="Internal",AH11="None",AI11="None",AJ11="Comfort=True; A:Form:Advance heating"))
38	AL11	Plan quality	=AND(AC11,AK11)
39	AM11	A:FP: Control	=G10
40	AN11	A:FP: Leave	=AM10
41	AO11	A:FP: Move to door	=AN10
42	AP11	A:FxP: Descend	=AO10
43	AQ11	A:FxP: Walk	=OR(AP10,BC10)
44	AR11	A:FP: Advance heating	=I10
45	AS11	A:FP: Push advance	=AR10
46	AT11	A:FxP: Press	=AS10
47	AU11	C:IISO: Receive press advance	=AT10
48	AV11	C:Toggle	=AU10
49	AW11	C:Store: Advance	=AV10
50	AX11	C:O: Turn off heating	=AW10
51	AY11	C:O: Turn off LEDs	=AX10
52	AZ11	C:OpReturn	=AY10
53	BA11	A:FxP: See	=AZ10
54	BB11	A:Encode: LED off	=BA10
55	BC11	A:FxP: Open	=BB10
56	BD11	A:FxP: Close	=AND(AQ10,COUNTIF($AQ$9:AQ11,TRUE)=2)
57	BE11	A:LED:Status	=IF(H11,TRUE,IF(BB11,FALSE,BE10))
58	BF11	(Rad 'on')	=IF(AX11,NOT(BF10),BF10)
59	BG11	C:Gas	=0.03*BF11*MINUTE(E11)
60	BH11	C:Is- Advanced	=IF(AW11,TRUE,BH10)
61	BI11	C:LEDlights Status	=IF(AY11,FALSE,BI10)
62	BJ11	A:Abst-struct	=BJ10+CH(BE10,BE11)
63	BK11	A:Phys-struct	=BK10
64	BL11	A:Abst-beh	=BL10+CM(AM11:BD11,$AM$5:$BD$5)
65	BM11	A:Phys-beh	=BM10+CM(AM11:BD11,$AM$6:$BD$6)
66	BN11	C:Abst-struct	=BN10+CH(BH10,BH11)+CH(BI10,BI11)
67	BO11	C:Phys-struct	=BO10+BG11
68	BP11	C:Abst-beh	=BP10+CM(AM11:BD11,$AM$7:$BD$7)
69	BQ11	C:Phys-beh	=BQ10+CM(AM11:BD11,$AM$8:$BD$8)
70	BR11	(Poss. change)	=2*MINUTE(E11)
71	BS11	Rad temp	=IF(BF11,MIN(BS10+BR11,85),MAX(BS10-BR11,10))
72	BT11	(From Rad)	=BT10+0.02*(BS11-BS10)*MINUTE(E11)
73	BU11	Room temp	=IF(BT11>14,BT11,BT11+(4-0.5*(BT11-6)))
74	BV11	A's temp	=35+(BU11*0.1)
75	BW11	Comfort Tq	=OR(AND(BV11>36.5,BV11<37.5),NOT(BY11))
76	BX11	(Discomfort time)	=IF(BW11,0,MINUTE(E11)+BX10)
77	BY11	In house	=IF(BD11,FALSE,BY10)
78	BZ11		
79	CA11		
80	CB11		

Figure P15

Cycle 2 MUSE TD1.1(Current)
Structured Diagram

Cycle 2
TD1.1(c)

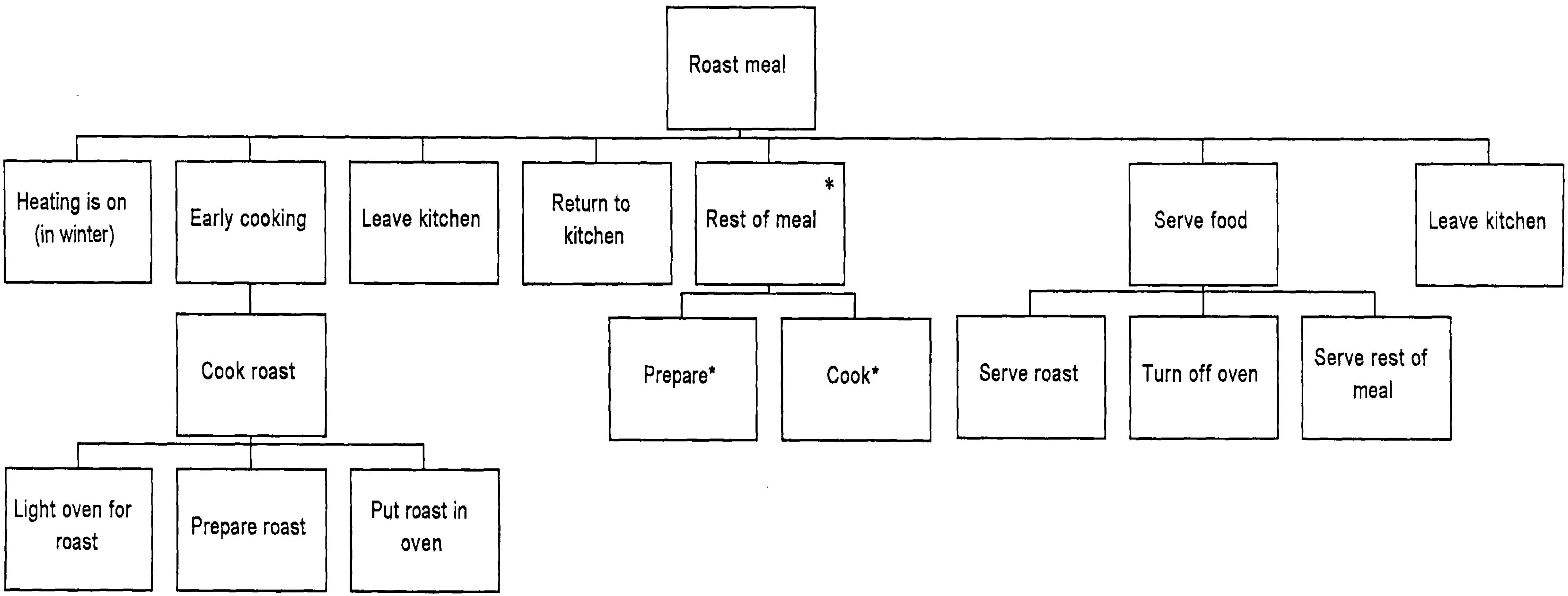


Figure P16

Cycle 2 MUSE TD1.2(Current)
Structured Diagram

Cycle 2
TD1.2(c)

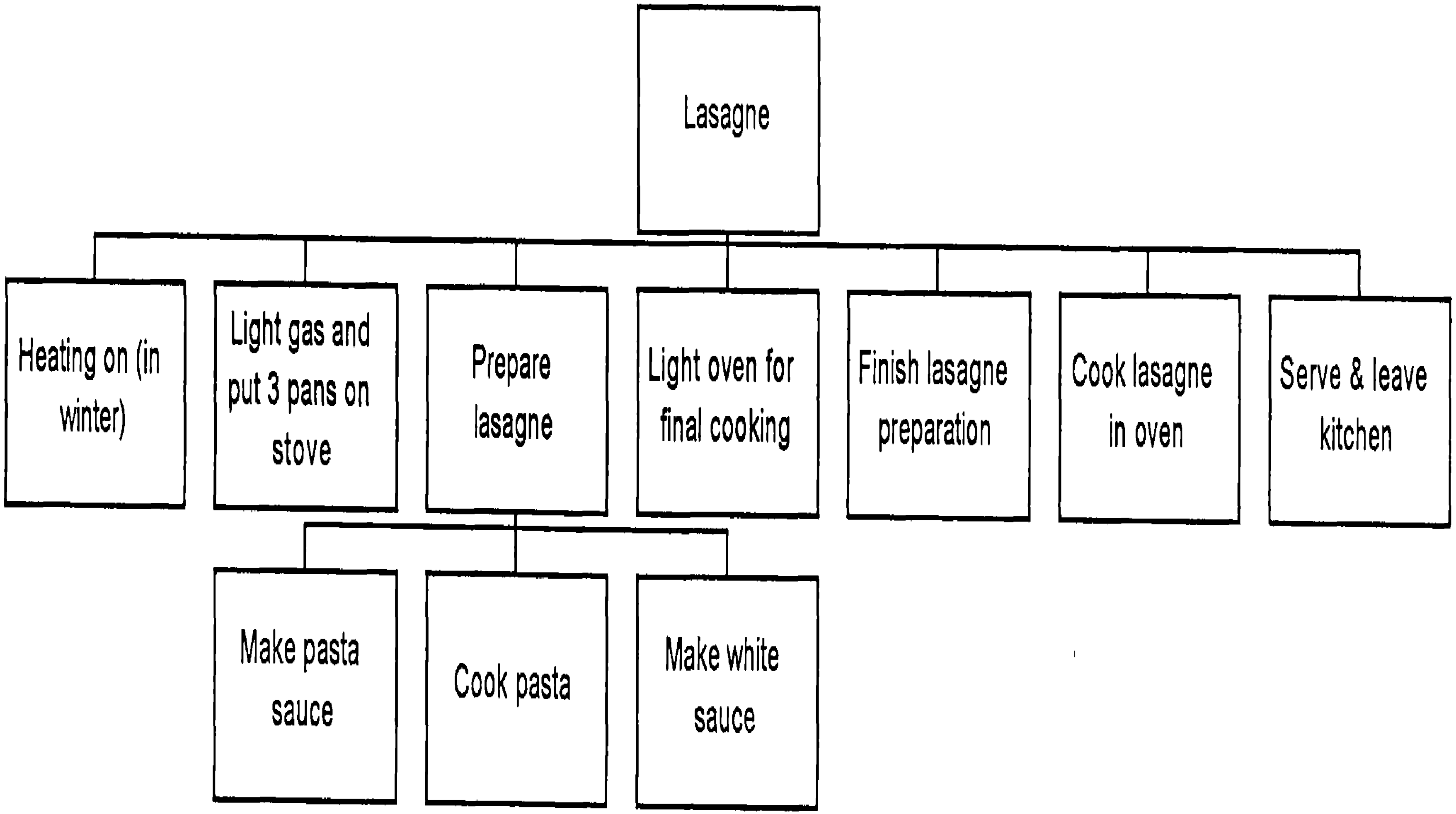


Figure P17

Cycle 2 MUSE TD1.3(Current)
Structured Diagram

Cycle 2
TD1.3(c)

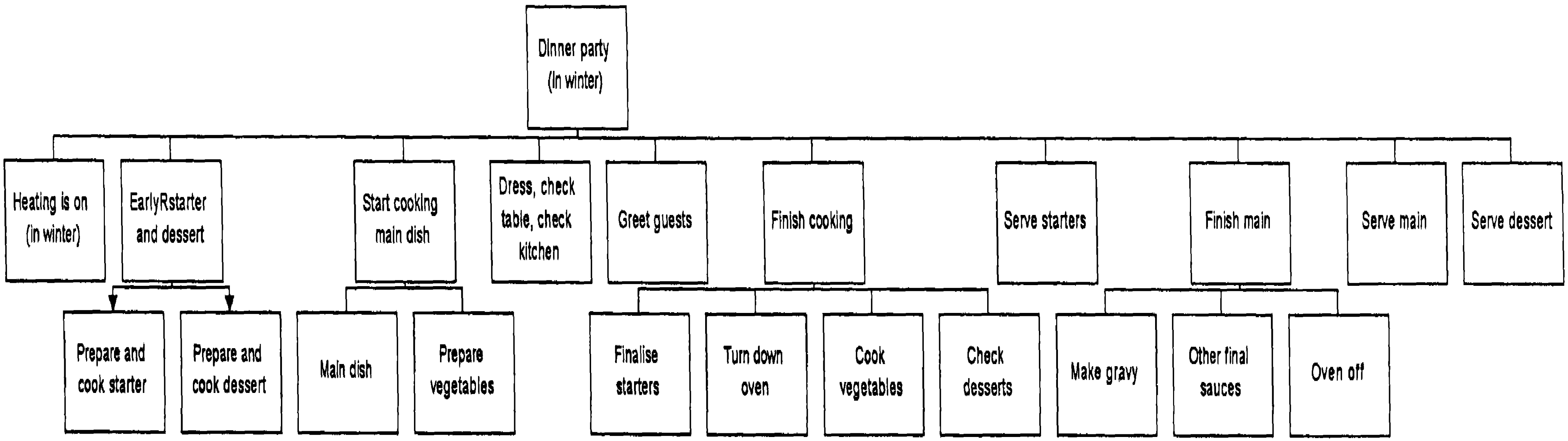


Figure P18

Cycle 2 MUSE TD1(Current)

Structured Diagram

Cycle 2
TD1(c)

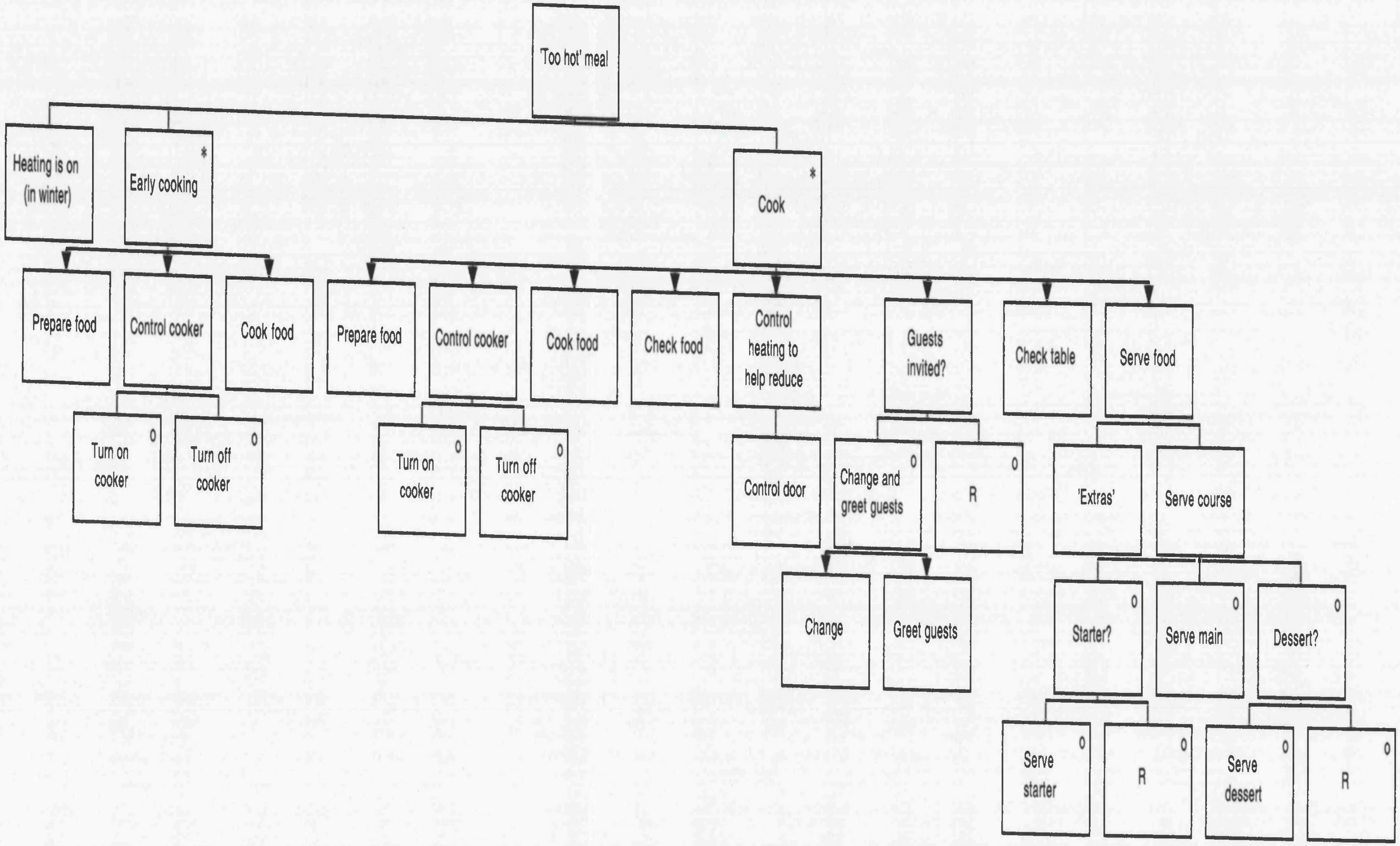

'Too hot' meal
Heating is on (in winter)
Early cooking *
Cook *
Prepare food
Control cooker
Cook food
Prepare food
Control cooker
Cook food
Check food
Control heating to help reduce
Guests invited?
Check table
Serve food
Turn on cooker o
Turn off cooker o
Turn on cooker o
Turn off cooker o
Control door
Change and greet guests o
R o
'Extras'
Serve course
Change
Greet guests
Starter? o
Serve main o
Dessert? o
Serve starter o
R o
Serve dessert o
R o

Figure P19

Cycle 2 MUSE TD2(Current)
Structured Diagram

Cycle 2
TD2(c)

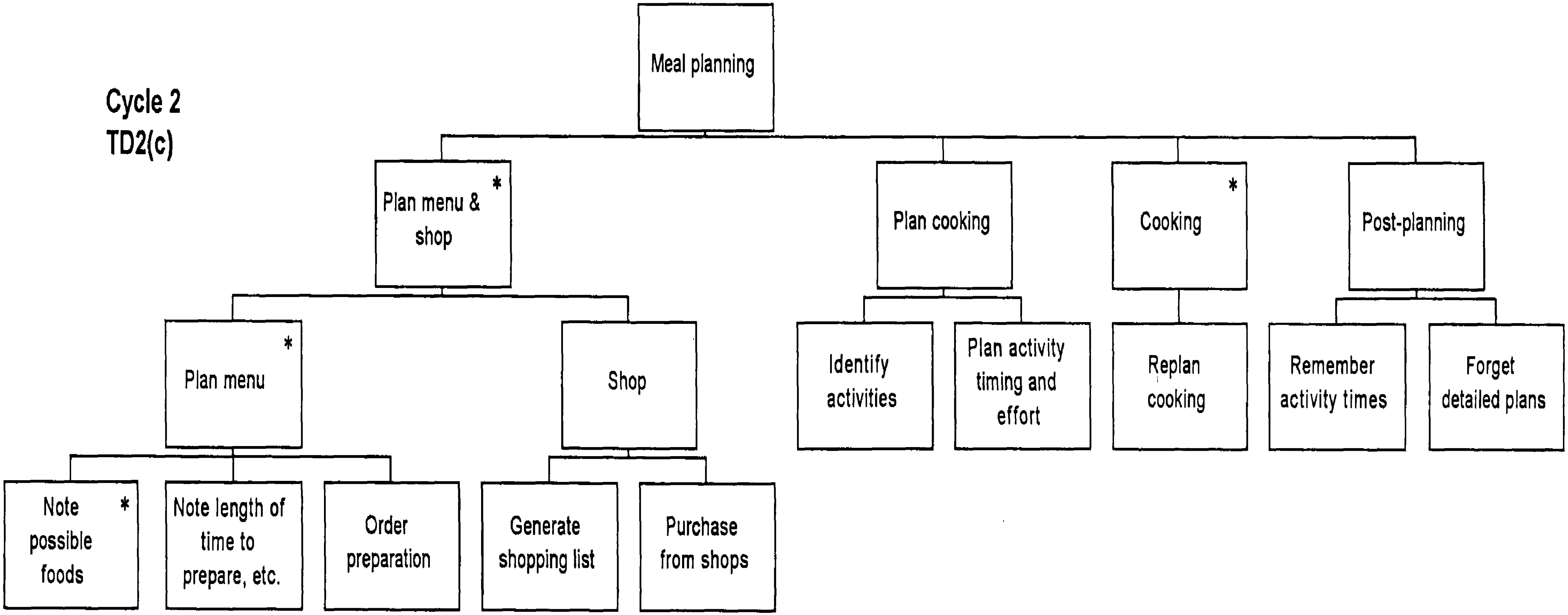

Meal planning
Plan menu & shop *
Plan cooking
Cooking *
Post-planning
Plan menu *
Shop
Identify activities
Plan activity timing and effort
Replan cooking
Remember activity times
Forget detailed plans
Note possible foods *
Note length of time to prepare, etc.
Order preparation
Generate shopping list
Purchase from shops

Figure P20

Cycle 2 MUSE DoDD(Current) Diagram

Cycle 2
DoDD(c)

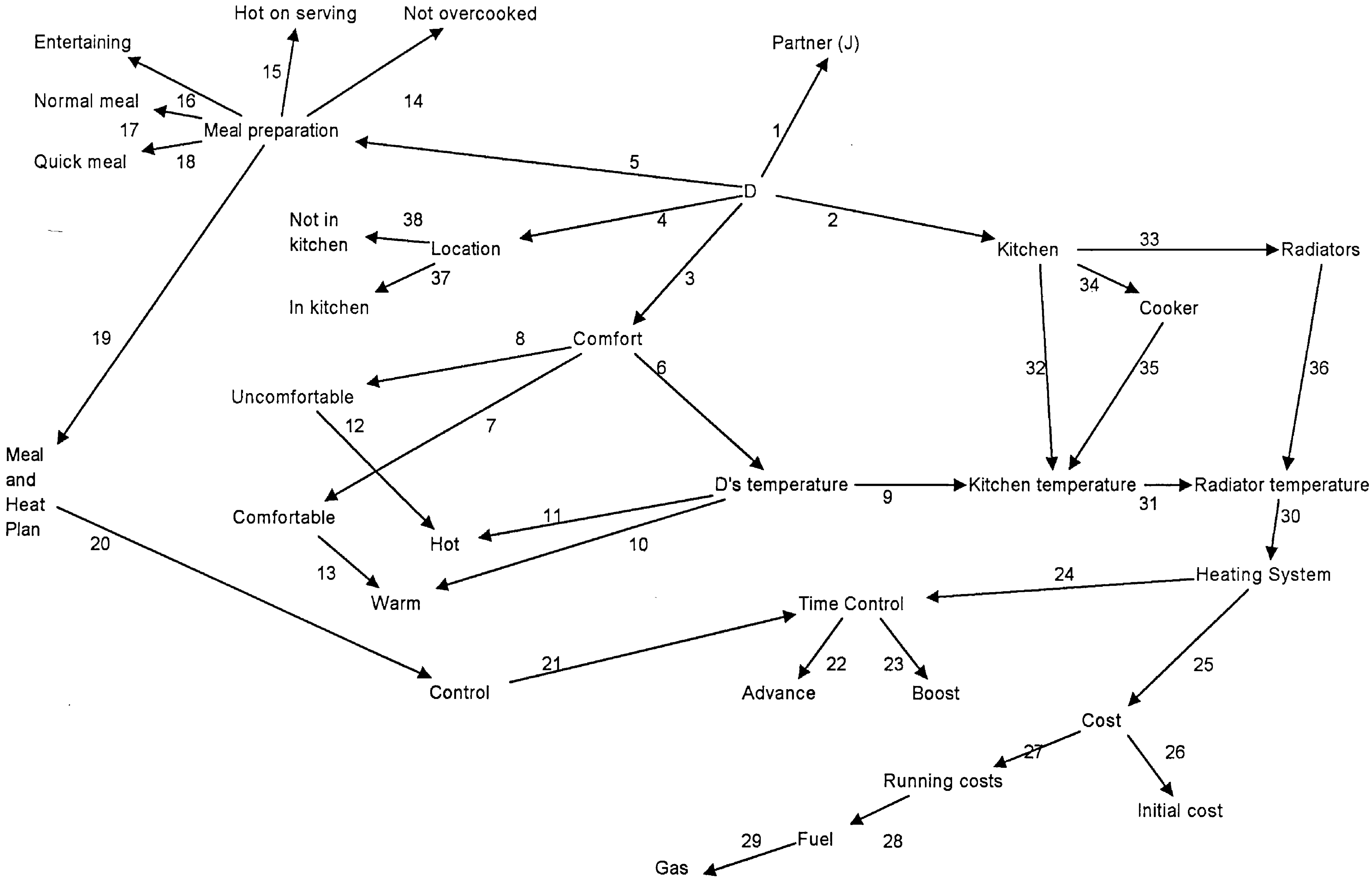


Figure P21
Cycle 2 MUSE GTM(Current)
Structured Diagram

Cycle 2
GTM(c)

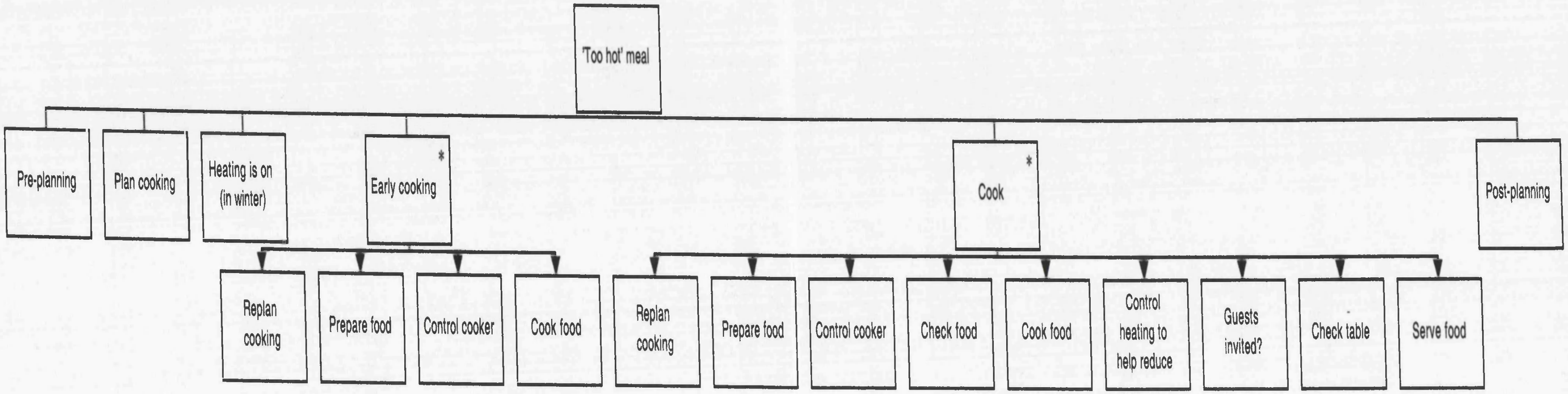
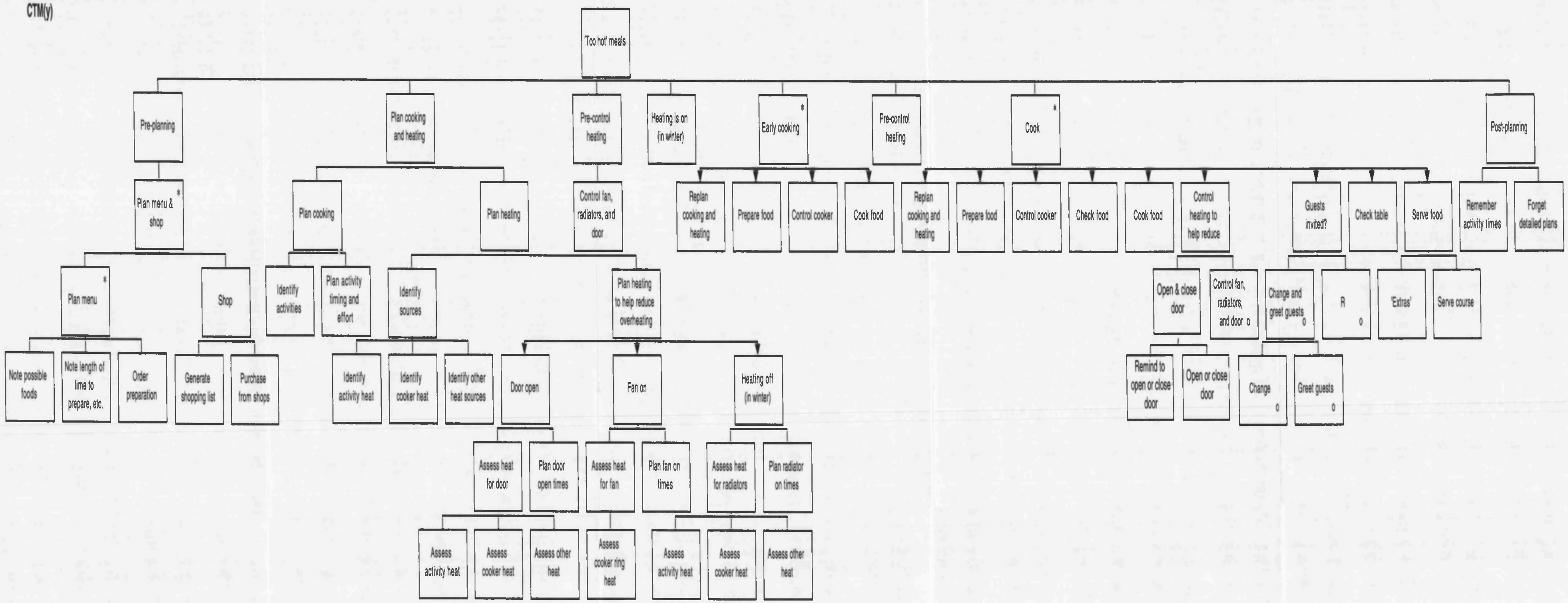

'Too hot' meal
Pre-planning
Plan cooking
Heating is on (in winter)
Early cooking *
Cook *
Post-planning
Replan cooking
Prepare food
Control cooker
Cook food
Replan cooking
Prepare food
Control cooker
Check food
Cook food
Control heating to help reduce
Guests invited?
Check table
Serve food

Figure P22

Cycle 2 MUSE CTM(y)

Structured Diagram

Cycle 2
CTM(y)



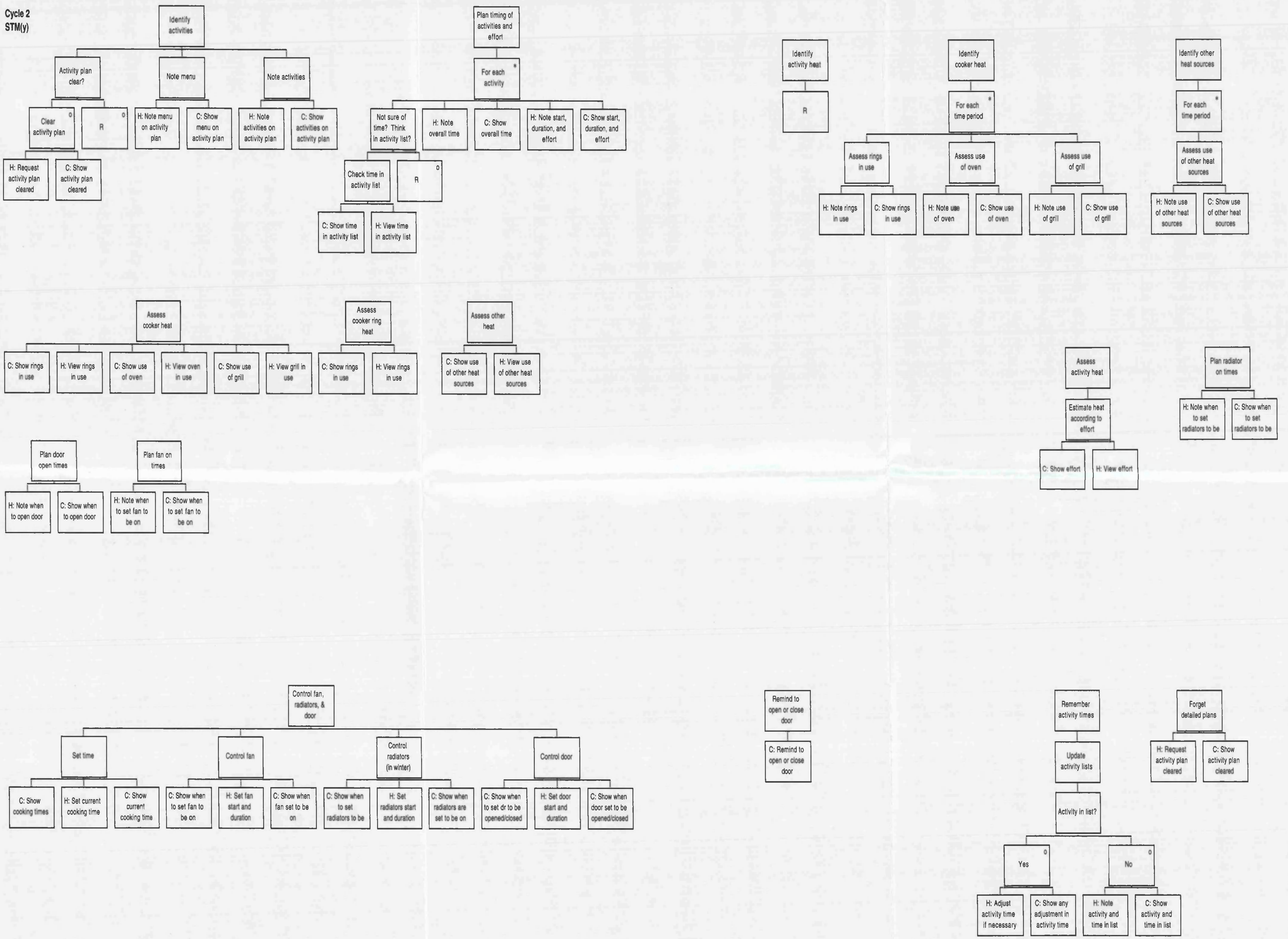
'Too hot' meals
Pre-planning
Plan cooking and heating
Pre-control heating
Heating is on (in winter)
Early cooking *
Pre-control heating
Cook *
Post-planning
Plan menu & shop *
Plan menu *
Shop
Note possible foods
Note length of time to prepare, etc.
Order preparation
Generate shopping list
Purchase from shops
Plan cooking
Plan heating
Identify activities
Plan activity timing and effort
Identify sources
Plan heating to help reduce overheating
Identify activity heat
Identify cooker heat
Identify other heat sources
Door open
Fan on
Heating off (in winter)
Assess heat for door
Plan door open times
Assess heat for fan
Plan fan on times
Assess heat for radiators
Plan radiator on times
Assess activity heat
Assess cooker heat
Assess other heat
Assess cooker ring heat
Assess activity heat
Assess cooker heat
Assess other heat
Control fan, radiators, and door
Replan cooking and heating
Prepare food
Control cooker
Cook food
Replan cooking and heating
Prepare food
Control cooker
Check food
Cook food
Control heating to help reduce
Guests invited?
Check table
Serve food
Remember activity times
Forget detailed plans
Open & close door
Control fan, radiators, and door o
Change and greet guests o
R o
'Extras'
Serve course
Remind to open or close door
Open or close door
Change o
Greet guests o

Figure P23

Cycle 2 MUSE STM(y)

Structured Diagram

Cycle 2
STM(y)



Identify activities
Activity plan clear?
Note menu
Note activities
Clear activity plan
R
H: Note menu on activity plan
C: Show menu on activity plan
H: Note activities on activity plan
C: Show activities on activity plan
H: Request activity plan cleared
C: Show activity plan cleared
Plan timing of activities and effort
For each activity
Not sure of time? Think in activity list?
H: Note overall time
C: Show overall time
H: Note start, duration, and effort
C: Show start, duration, and effort
Check time in activity list
R
C: Show time in activity list
H: View time in activity list
Identify activity heat
R
Identify cooker heat
For each time period
Assess rings in use
Assess use of oven
Assess use of grill
H: Note rings in use
C: Show rings in use
H: Note use of oven
C: Show use of oven
H: Note use of grill
C: Show use of grill
Identify other heat sources
For each time period
Assess use of other heat sources
H: Note use of other heat sources
C: Show use of other heat sources
Assess cooker heat
C: Show rings in use
H: View rings in use
C: Show use of oven
H: View oven in use
C: Show use of grill
H: View grill in use
Assess cooker ring heat
C: Show rings in use
H: View rings in use
Assess other heat
C: Show use of other heat sources
H: View use of other heat sources
Assess activity heat
Estimate heat according to effort
C: Show effort
H: View effort
Plan radiator on times
H: Note when to set radiators to be
C: Show when to set radiators to be
Plan door open times
H: Note when to open door
C: Show when to open door
Plan fan on times
H: Note when to set fan to be on
C: Show when to set fan to be on
Control fan, radiators, & door
Set time
Control fan
Control radiators (in winter)
Control door
C: Show cooking times
H: Set current cooking time
C: Show current cooking time
C: Show when to set fan to be on
H: Set fan start and duration
C: Show when fan set to be on
C: Show when to set radiators to be
H: Set radiators start and duration
C: Show when radiators are set to be on
C: Show when to set dr to be opened/closed
H: Set door start and duration
C: Show when door set to be opened/closed
Remind to open or close door
C: Remind to open or close door
Remember activity times
Update activity lists
Activity in list?
Yes
No
H: Adjust activity time if necessary
C: Show any adjustment in activity time
H: Note activity and time in list
C: Show activity and time in list
Forget detailed plans
H: Request activity plan cleared
C: Show activity plan cleared

Figure P24
Cycle 2 MUSE UTM(y)
Structured Diagram

Cycle 2
UTM(y)

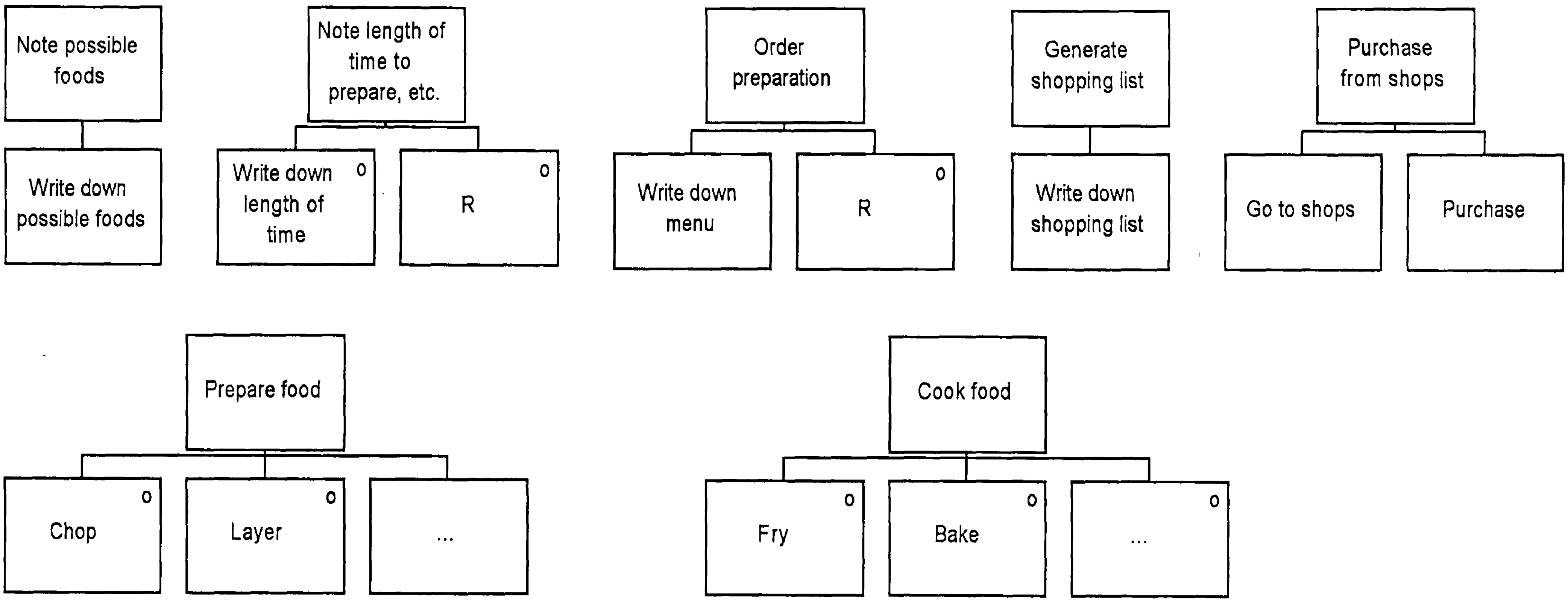
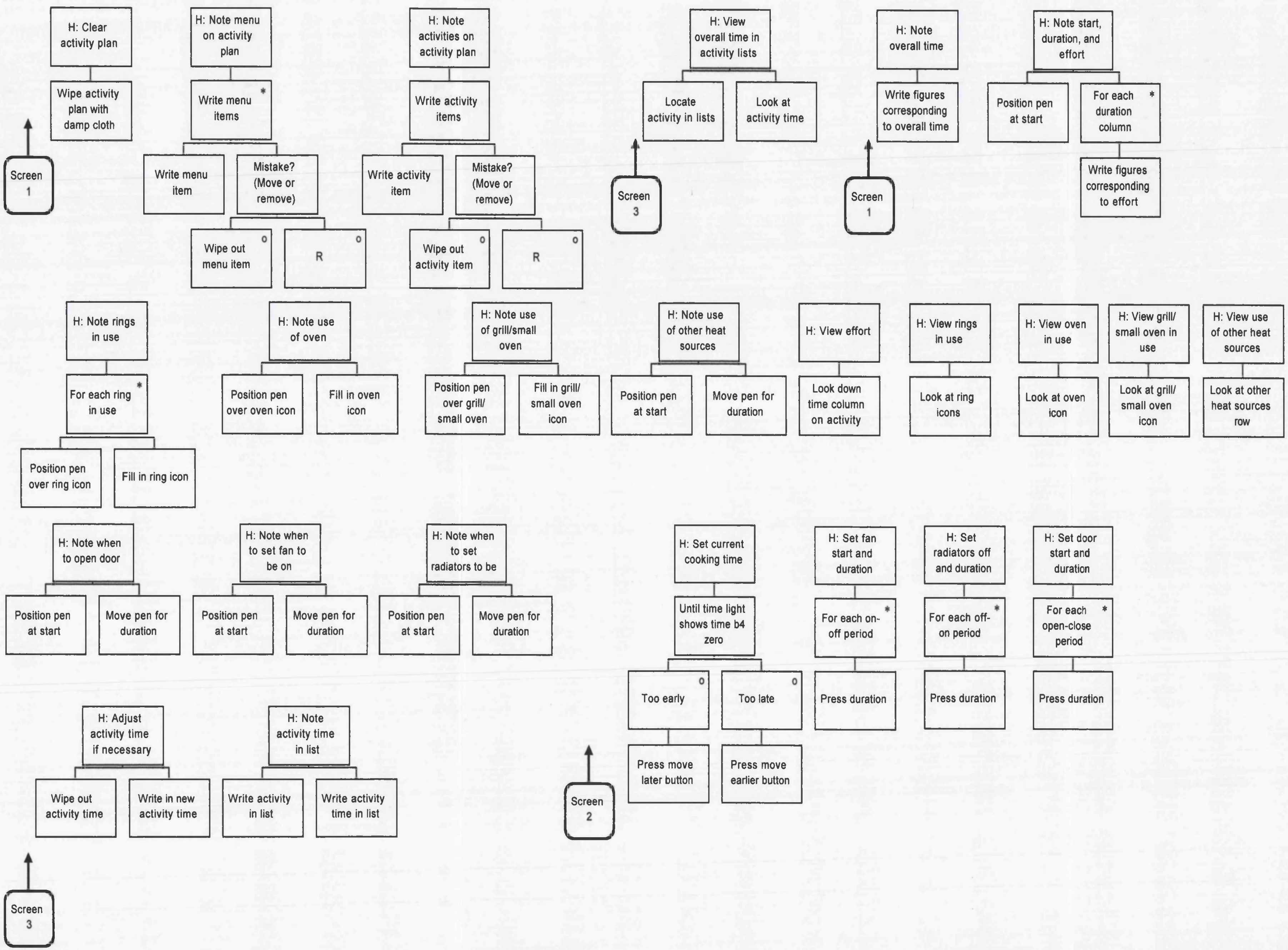

Note possible foods
Write down possible foods
Note length of time to prepare, etc.
Write down length of time
o
R
o
Order preparation
Write down menu
R
o
Generate shopping list
Write down shopping list
Purchase from shops
Go to shops
Purchase
Prepare food
Chop
o
Layer
o
...
Cook food
Fry
o
Bake
o
...
o

Figure P25

Cycle 2 MUSE ITM(y)

Structured Diagram

Cycle 2
ITM(y)



H: Clear activity plan
Wipe activity plan with damp cloth
Screen 1
H: Note menu on activity plan
Write menu items *
Write menu item
Mistake? (Move or remove)
Wipe out menu item o
R o
H: Note activities on activity plan
Write activity items
Write activity item
Mistake? (Move or remove)
Wipe out activity item o
R o
H: View overall time in activity lists
Locate activity in lists
Look at activity time
Screen 3
H: Note overall time
Write figures corresponding to overall time
Screen 1
H: Note start, duration, and effort
Position pen at start
For each duration column *
Write figures corresponding to effort
H: Note rings in use
For each ring in use *
Position pen over ring icon
Fill in ring icon
H: Note use of oven
Position pen over oven icon
Fill in oven icon
H: Note use of grill/small oven
Position pen over grill/ small oven
Fill in grill/ small oven icon
H: Note use of other heat sources
Position pen at start
Move pen for duration
H: View effort
Look down time column on activity
H: View rings in use
Look at ring icons
H: View oven in use
Look at oven icon
H: View grill/ small oven in use
Look at grill/ small oven icon
H: View use of other heat sources
Look at other heat sources row
H: Note when to open door
Position pen at start
Move pen for duration
H: Note when to set fan to be on
Position pen at start
Move pen for duration
H: Note when to set radiators to be
Position pen at start
Move pen for duration
H: Set current cooking time
Until time light shows time b4 zero
Too early o
Too late o
Press move later button
Press move earlier button
Screen 2
H: Set fan start and duration
For each on-off period *
Press duration
H: Set radiators off and duration
For each off-on period *
Press duration
H: Set door start and duration
For each open-close period *
Press duration
H: Adjust activity time if necessary
Wipe out activity time
Write in new activity time
H: Note activity time in list
Write activity in list
Write activity time in list
Screen 3

Figure P26

Cycle 2 MUSE IM(y)
Structured Diagram

Cycle 2
IM(y)

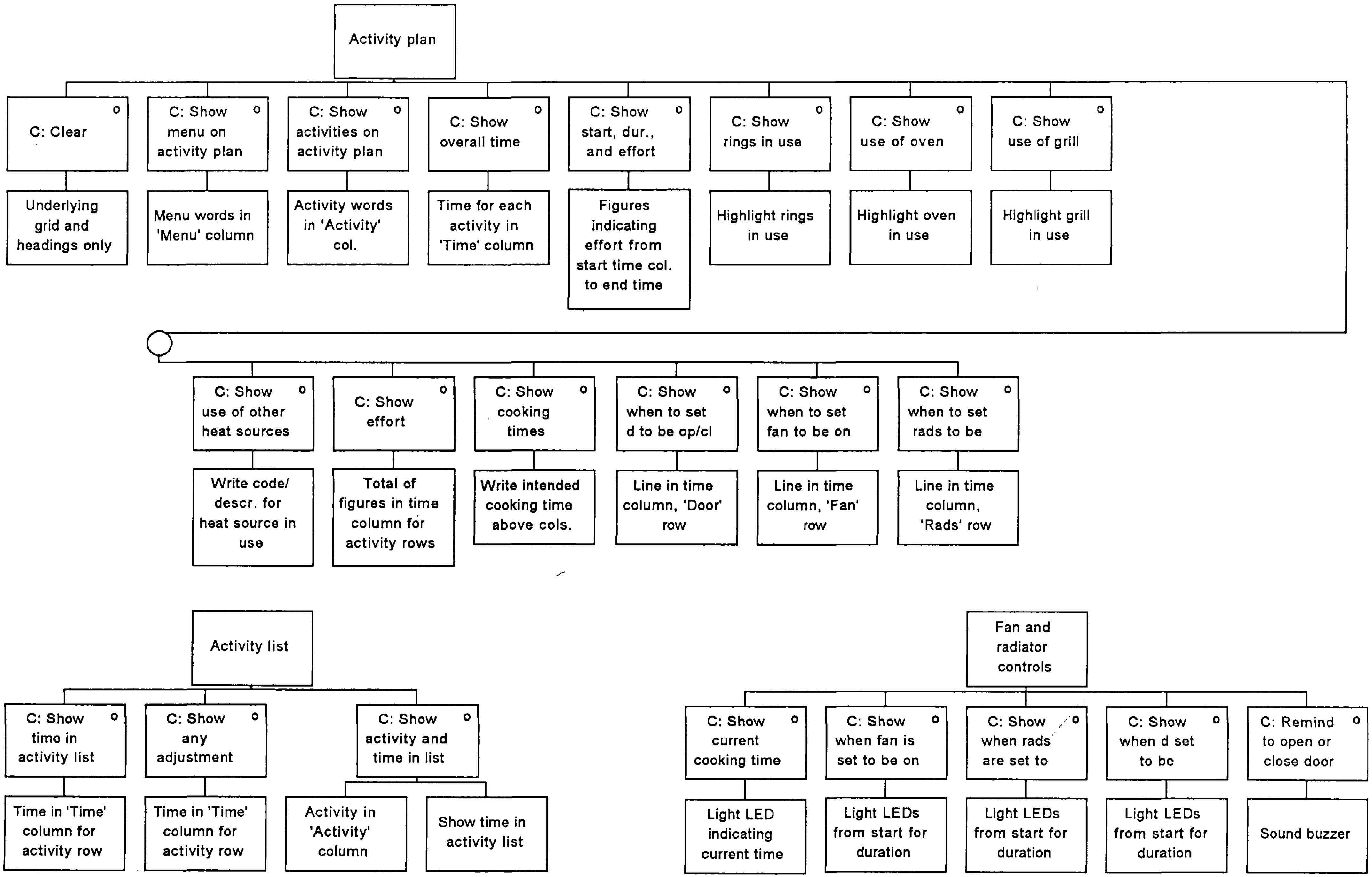

Activity plan
C: Clear
C: Show menu on activity plan
C: Show activities on activity plan
C: Show overall time
C: Show start, dur., and effort
C: Show rings in use
C: Show use of oven
C: Show use of grill
Underlying grid and headings only
Menu words in 'Menu' column
Activity words in 'Activity' col.
Time for each activity in 'Time' column
Figures indicating effort from start time col. to end time
Highlight rings in use
Highlight oven in use
Highlight grill in use
C: Show use of other heat sources
C: Show effort
C: Show cooking times
C: Show when to set d to be op/cl
C: Show when to set fan to be on
C: Show when to set rads to be
Write code/ descr. for heat source in use
Total of figures in time column for activity rows
Write intended cooking time above cols.
Line in time column, 'Door' row
Line in time column, 'Fan' row
Line in time column, 'Rads' row
Activity list
C: Show time in activity list
C: Show any adjustment
C: Show activity and time in list
Time in 'Time' column for activity row
Time in 'Time' column for activity row
Activity in 'Activity' column
Show time in activity list
Fan and radiator controls
C: Show current cooking time
C: Show when fan is set to be on
C: Show when rads are set to
C: Show when d set to be
C: Remind to open or close door
Light LED indicating current time
Light LEDs from start for duration
Light LEDs from start for duration
Light LEDs from start for duration
Sound buzzer

Figure P27
Cycle 2 MUSE PSL1(y)
Diagram

Plan That Meal!™—Activity Plan

Menu	Activity	Time	Early	-3:00	-2:45	-2:30	-2:15	-2:00	-1:45	-1:30	-1:15	-1:00	-0:45	-0:30	-0:15	0:00	0:15	0:30	0:45	1:00	1:15	1:30	1:45	2:00
	Clear kitchen																							
	Lay table																							
	Prepare drinks+nibbles																							
	Get changed																							
	Greet guests																							
	Relaxation																							
		Heat	Cooker																					
			Other																					
		Door																						
		Fan																						
		Rads																						

Figure P28
Cycle 2 MUSE PSL2(y)
Diagram

Plan That Meal!™—Activity Lists

Activity	Time	Notes
Lay table		
Prepare drinks+nibbles		
Get changed		
Greet guests		
Warm plates		

Starters | Mains—Meat | Mains—Fish | Mains—Other | Vegetables | Salads | Desserts | Others

Figure P29

Cycle 2 MUSE PSL3(y)
Diagram

Plan That Meal!™—Full Instructions

Your Plan That Meal!TM is designed to make complicated meals easier—it helps you to plan your cooking so that you will not be flustered and too hot. It also helps you to plan when to open the kitchen door and the settings for your fan and radiator controls.

Use the red pen when planning. The red pen is water soluble so that you can change the plan using a damp cloth or tissue. (Use the green pen if you want to change the plan while cooking, since it will be visible over the red.)

Planning. Use the Activity Plan.

Menu	Fill in the menu.
Critical Times	Put critical times under the plan times. For example, you could start by putting the time the guests arrive under 0:00. The early column is for activities that you may need to do before the planning period.
Activities	Fill in the activities required to produce the menu.
Activity Times	Estimate how long each activity will take.
Plan	Plan when each activity will be done. Write a number between 1 and 9 to indicate the effort involved under each plan time.
Cooker heat	In the heat planner section, shade the parts of the cooker in use at any time.
Kitchen door	Draw lines to indicate when the kitchen door should be open. In general, plan to open the door just before high levels of activity, particularly if the cooker is producing a lot of heat or the weather is hot. You can identify high levels of activity by looking down a time column on the plan. When you have finished the plan, set the door controller to remind you when to open and close the door.
Fan control	Draw lines to indicate when the fan should be on. In general, the fan should be when two or more rings of the cooker are in use. When you have finished the plan, set the fan controller to turn on and off the fan at the planned times.
Radiator control	Draw lines to indicate when the radiator should be on. In general, the radiator should not be on when: the oven is on; more than three rings of the cooker are on; and/or there is a high level of activity. When you have finished the plan, set the radiator controller to turn on and off the radiators at the planned times.

After cooking

Update lists	Update the Activity Lists (over) with the actual time taken for each activity. The lists will help you to plan future meals.
Wipe clean	Wipe off the Activity Plan ready for next time.

Figure P30

Cycle 2 MUSE PSL4(y)
Diagram

Cycle 2
PSL4(y)

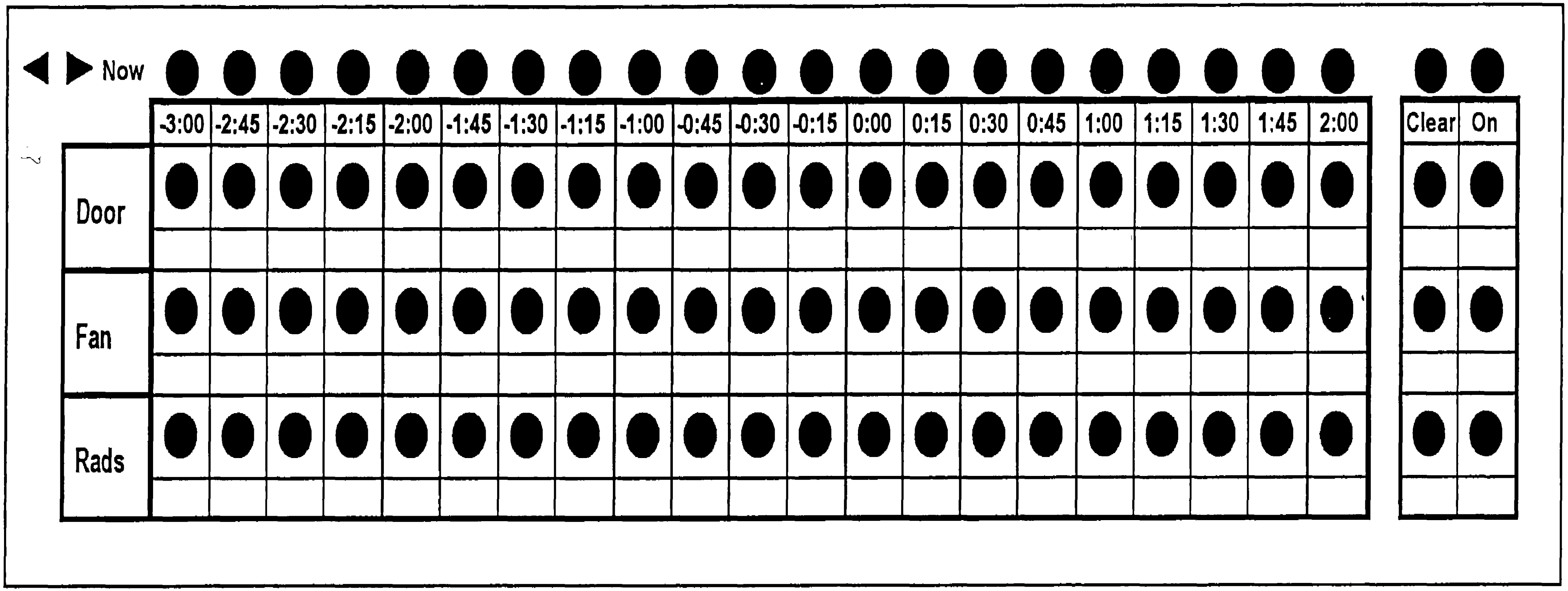


Figure P31

Cycle 2 Operationalisation

Current Planning Diagram

Operationalisation 2
Current Planning

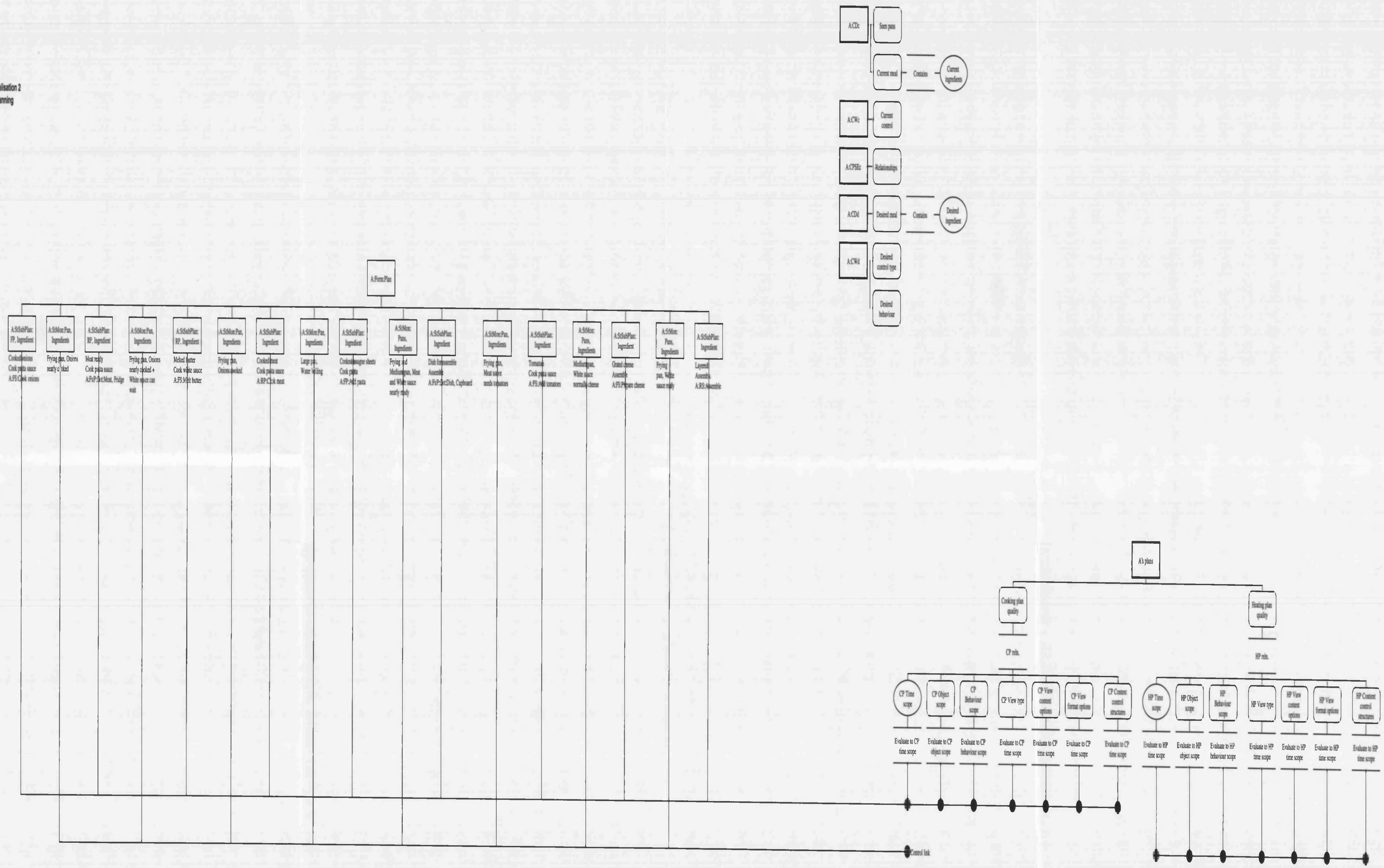
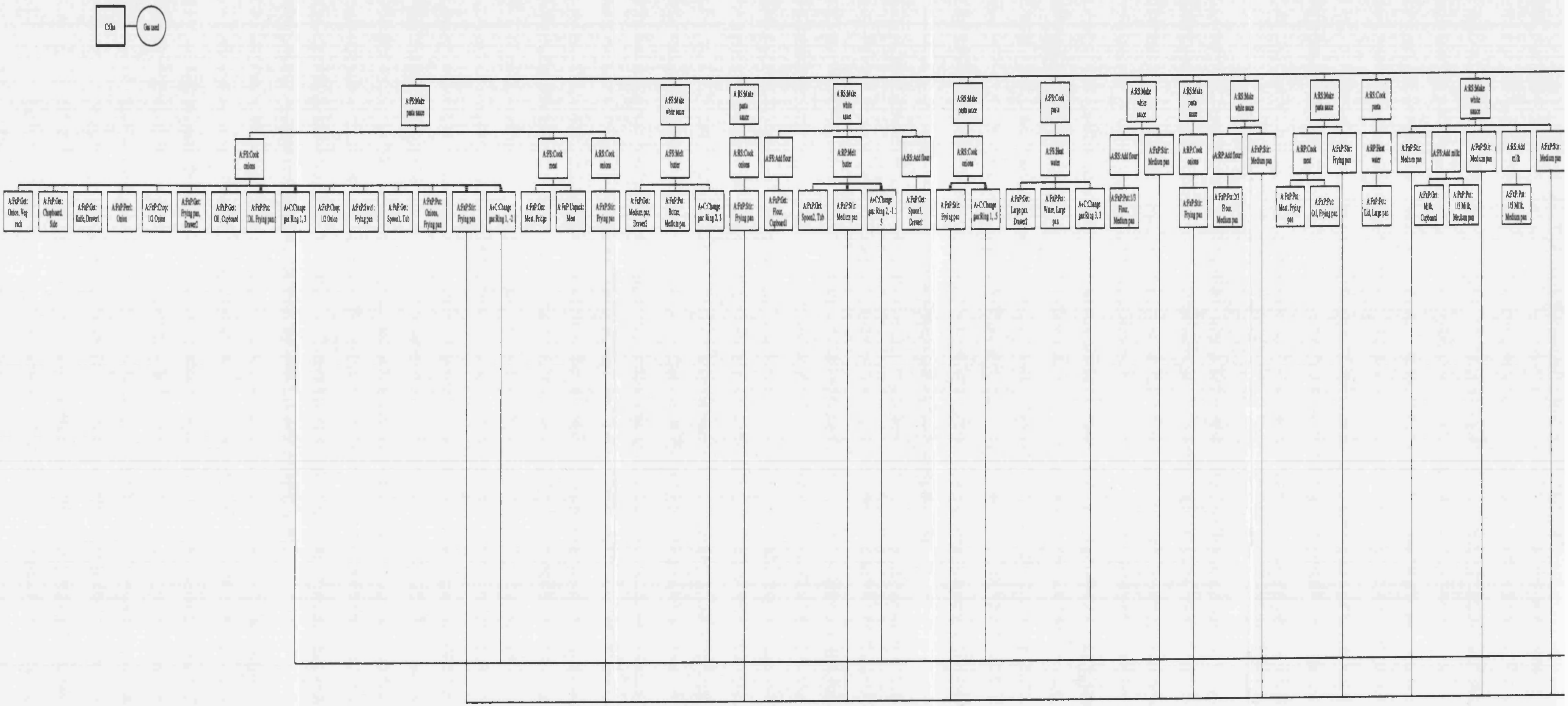


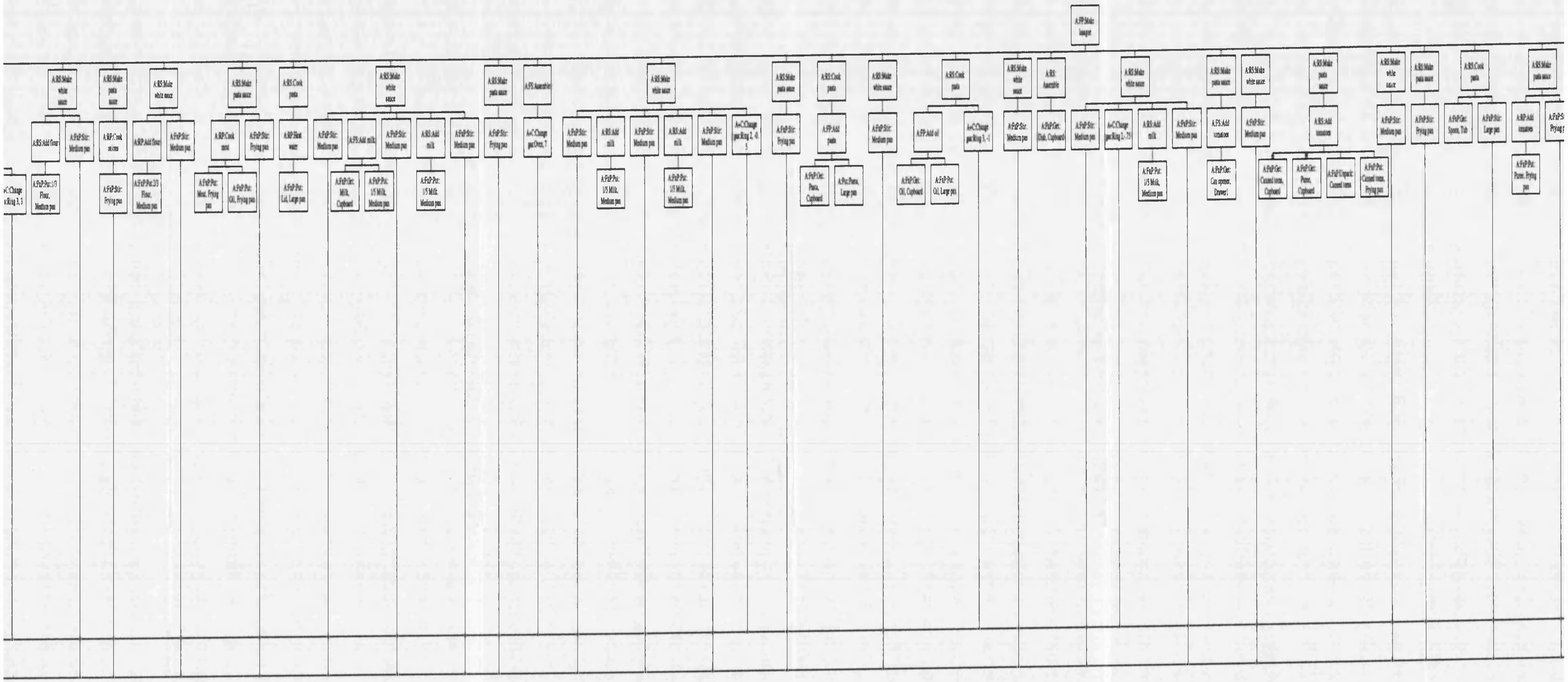
Figure P32

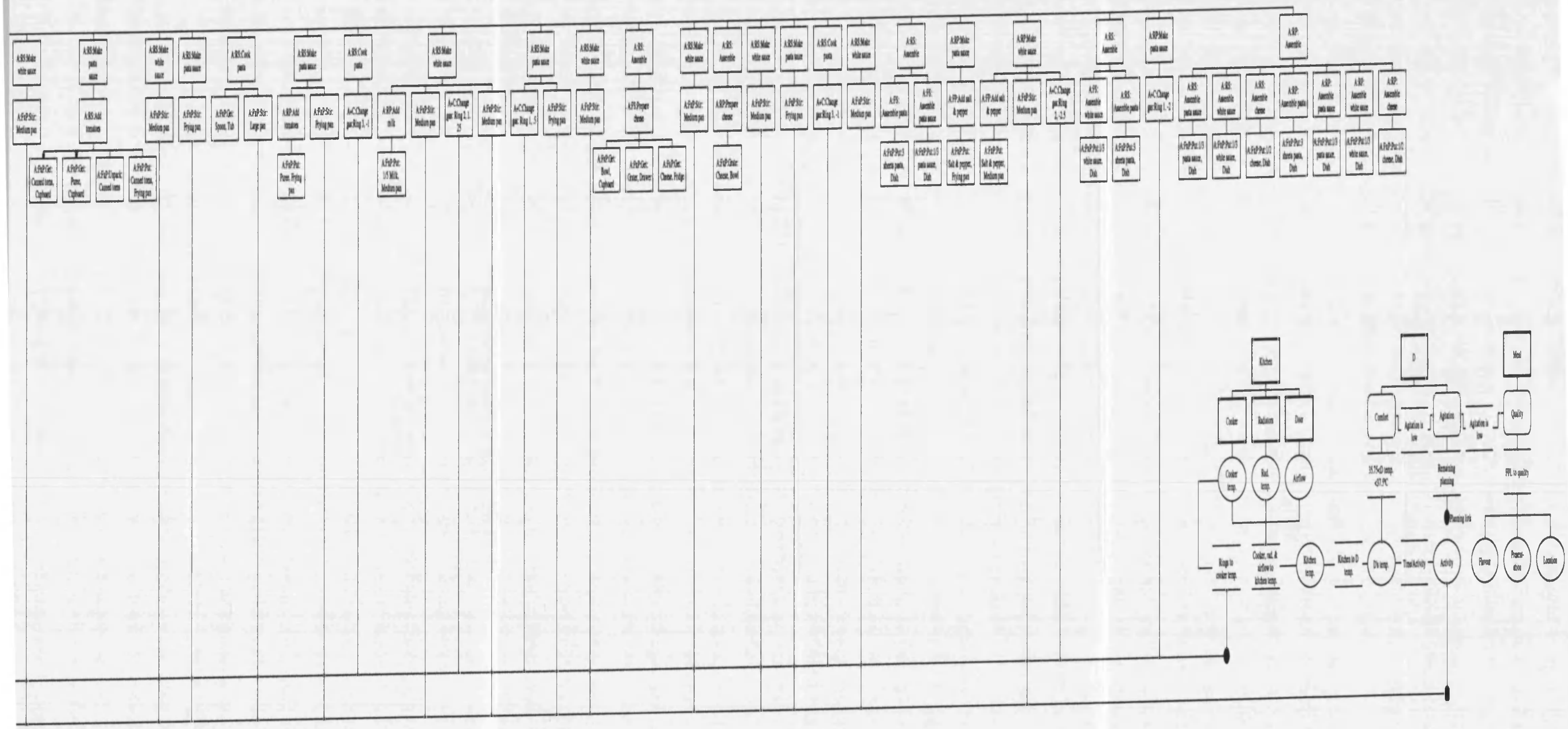
Cycle 2 Operationalisation

Current Control Diagram

Operationalisation 2
Current Control







Meal
Quality
D
Comfort
Agitation
Kitchen
Cooker
Radiators
Door
Cooker temp.
Rad. temp.
Airflow
Kitchen temp.
D's temp.
Activity
Flavour
Presentation
Location
Planning link

Figure P33
Cycle 2 Operationalisation
Current Planning and Control Table

Figure P34

Cycle 2 Operationalisation
Current Planning and Control
Formulae

Cycle 2 Operationalisation Table Formulae (1/1)

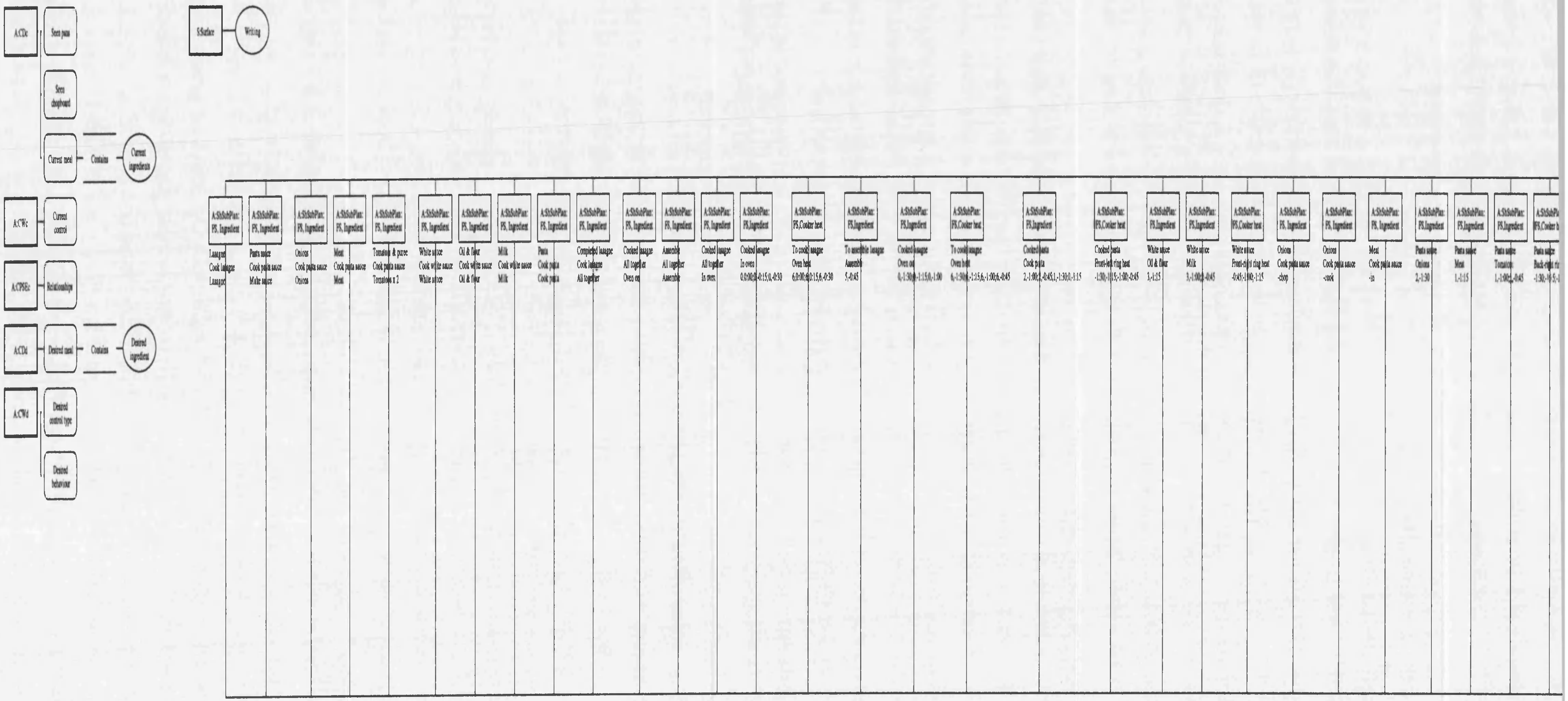
Current.xls

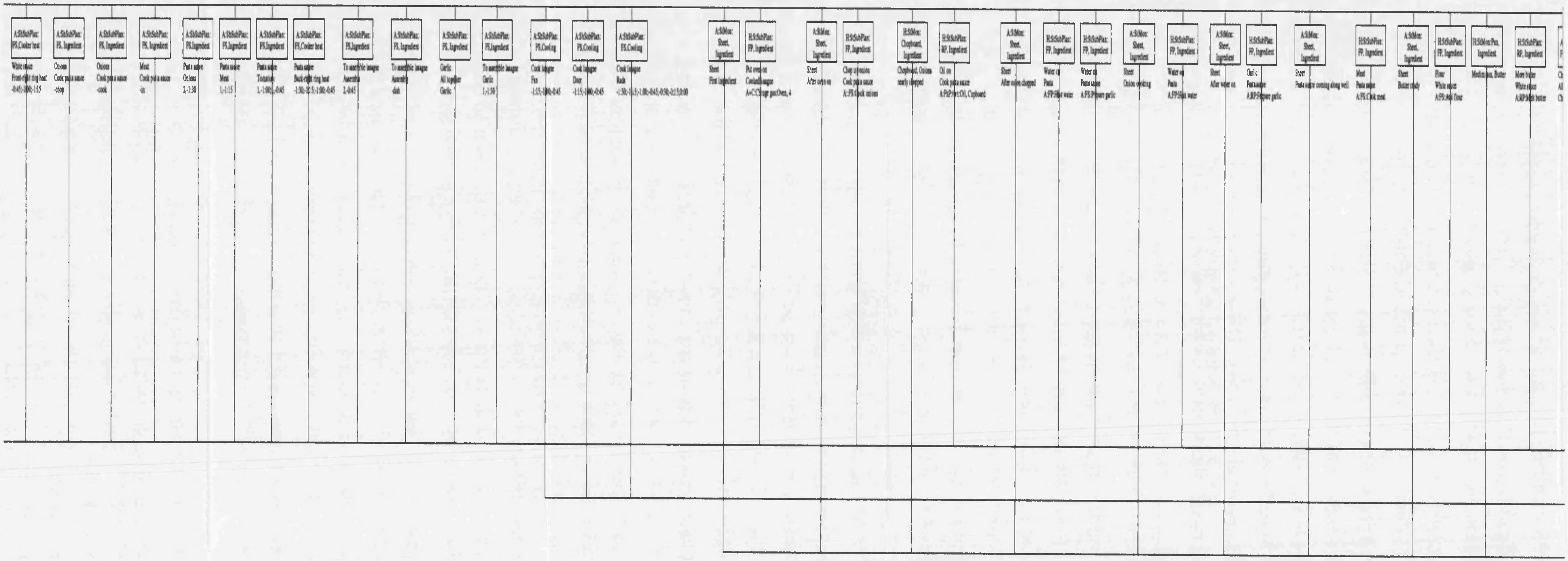
	Loc	Heading	Formula in the 11th row
1	A11		
2	B11		3
3	C11		0.666666666666667
4	D11		=C11-$C$9
5	E11		=C11-C10
6	F11	A:FP: Plan	FALSE
7	G11	A:StSubPlan: FP,Ingredient	=F10
8	H11	A:StMon: Pan, Ingredients	=OR(BB10,AND(BA10,COUNTIF($BA$9:BA10,TRUE)=8),AND(CM10,COUNTIF($CM$9:CM10,TRUE)=5),AND(BR10,COUNTIF($BR$9:BR10,TRUE)=18))
9	I11	A:StSubPlan: RP, Ingredient	=OR(H10, J10)
10	J11	A:StMon: Pans, Ingredients	=AND(CW10,COUNTIF($CW$9:CW10,TRUE)=1)
11	K11	A:CDc: Seen pan(s)	=CHOOSE(COUNTIF($H$9:H11,TRUE)+COUNTIF($J$9:J11,TRUE)+1,"ó", "Frying pan", "Large pan", "Frying and medium pans", "Frying pan", "Medium pan")
12	L11	A:CDc: Ingredients	=CHOOSE(COUNTIF($H$9:H11,TRUE)+COUNTIF($J$9:J11,TRUE)+1,"ó", "Nearly cooked onions", "Water boiling", "Meat and white sauce nearly ready", "Meat sauce needs tomatoes", "Usually make cheese sauce with white sauce")
13	M11	A:CPSEc: Relationships	=M10
14	N11	A:CDd: Desired ingredients	=CHOOSE(COUNTIF($G$9:G11,TRUE)+COUNTIF($I$9:I11,TRUE)+1,"ó", "Cooked onions", "Meat ready, Melted butter", "Cooked lasagne sheets", "Dish for assembled lasagne", "Add tomatoes to pasta sauce", "Grated cheese")
15	O11	A:CWd: Desired control type	=CHOOSE(COUNTIF($G$9:G11,TRUE)+COUNTIF($I$9:I11,TRUE)+1,"ó", "Cook pasta sauce", "Cook pasta sauce, Cook white sauce", "Cook pasta", "Assemble", "Cook pasta sauce", "Assemble")
16	P11	A:CWd: Desired behaviour	=CHOOSE(COUNTIF($G$9:G11,TRUE)+COUNTIF($I$9:I11,TRUE)+1,"ó", "A:FS:Cook onions", "A:FxP:Get:Meat, Fridge; A:FS:Melt butter", "A:FP:Add pasta", "A:FxP:Get:Dish, Cupboard", "A:FS:Add tomatoes", "A:FS:Prepare cheese")
17	Q11	A:Abst-struct	=Q10+CH(K10,K11)+CH(L10,L11)+CH(N10,N11)+CH(O10,O11)+CH(P10,P11)
18	R11	A:Phys-struct	=R10
19	S11	A:Abst-beh	=S10+CM(F11:J11,$F$5:$J$5)
20	T11	A:Phys-beh	=T10+CM(F11:J11,$F$6:$J$6)
21	U11	CP Time scope	=CHOOSE(COUNTIF($G$9:G11,TRUE)+COUNTIF($I$9:I11,TRUE)+1,"ó", 0.2, "0.2, 0.2", 0.2, 0.2, 0.2, 0.2)
22	V11	CP Object scope	=N11
23	W11	CP Behaviour scope	=P11
24	X11	CP View type	Internal
25	Y11	CP View Content options	None
26	Z11	CP View format options	None
27	AA11	CP Content control structures	=O11 & "; " & P11
28	AB11	CP quality	TRUE
29	AC11	HP Time scope	ó
30	AD11	HP Object scope	ó
31	AE11	HP Behaviour scope	ó
32	AF11	HP View type	ó
33	AG11	HP View content options	ó
34	AH11	HP View format options	ó
35	AI11	HP Content control structures	ó
36	AJ11	HP quality	ó
37	AK11	Plan quality	TRUE
38	AL11	A:FP: Control	=G10
39	AM11	A:FP:Make lasagne	=AL10
40	AN11	A:FS: Cook onions	=AM10
41	AO11	A:FxP:Get: Onion, Veg rack	=AN10
42	AP11	A:FxP:Get: Chopboard, Side	=AO10
43	AQ11	A:FxP:Get: Knife, Drawer1	=AP10
44	AR11	A:FxP: Peel: Onion	=AQ10
45	AS11	A:FxP: Chop, 1/2 onion	=OR(AR10,AW10)
46	AT11	A:FxP:Get: Frying pan, Drawer 2	=AND(AS10,COUNTIF($AS$9:AS11,TRUE)=1)
47	AU11	A:FxP:Get: Oil, Cupboard	=AT10
48	AV11	A:FxP:Put: Oil, Frying pan	=OR(AU10,CG10)
49	AW11	A+C: Change ring: 1, 3	=AND(AV11,COUNTIF($AV$9:AV11,TRUE)=1)
50	AX11	A:FxP: Swirl: Frying pan	=AND(AS10,COUNTIF($AS$9:AS11,TRUE)=2)
51	AY11	A:FxP:Get: Spoon1, Tub	=AX10
52	AZ11	A:FxP:Put: Onions, Frying pan	=AY10
53	BA11	A:FxP:Stir: Frying pan	=OR(AZ10,BF10,CC10,AND(AV10,COUNTIF($AV$9:AV10,TRUE)=2),AND(BL10,OR(COUNTIF($BL$9:BL10,TRUE)=5,COUNTIF($BL$9:BL10,TRUE)=6,COUNTIF($BL$9:BL10,TRUE)=9,COUNTIF($BL$9:BL10,TRUE)=12)),DK10,DN10)
54	BB11	A+C: Change ring: 1, -2	=AND(BA10,COUNTIF($BA$9:BA10,TRUE)=1)
55	BC11	A:FS: Cook meat	=I10
56	BD11	A:FxP:Get: Meat, Fridge	=BC10
57	BE11	A:FxP: Unpack: Meat	=BD10
58	BF11	A:RS: Cook onions	=OR(BE10,AND(BL10,COUNTIF($BL$9:BL11,TRUE)=1),AND(BL10,COUNTIF($BL$9:BL11,TRUE)=2))
59	BG11	A:FS:Make white sauce	=AND(BA10,COUNTIF($BA$9:BA10,TRUE)=2)
60	BH11	A:FS: Melt butter	=BG10
61	BI11	A:FxP:Get: Medium pan, Drawer2	=BH10
62	BJ11	A:FxP:Put: Butter, Medium pan	=BI10
63	BK11	A+C: Change ring, 2, 3	=BJ10
64	BL11	A:RS:Make pasta sauce	=OR(BK10,BU10,AND(BR10,COUNTIF($BR$9:BR11,TRUE)=2),AND(BR10,COUNTIF($BR$9:BR11,TRUE)=3),AND(BR10,COUNTIF($BR$9:BR10,TRUE)=5),CQ10,AND(I10,COUNTIF($I$9:I10,TRUE)=4),AND(BR10,COUNTIF($BR$9:BR10,TRUE)=14),AND(BR10,COUNTIF($BR$9:BR10,TRUE)=15),DI10,AND(BR10,COUNTIF($BR$9:BR10,TRUE)=17),AND(BR10,COUNTIF($BR$9:BR10,TRUE)=20))
65	BM11	A:RS:Make white sauce	=OR(AND(BA10,OR(COUNTIF($BA$9:BA10,TRUE)=3,COUNTIF($BA$9:BA10,TRUE)=5,COUNTIF($BA$9:BA10,TRUE)=14)),CA10,CJ10,CP10,CT10,CW10,CY10,DB10,DG10,DR10,DT10,DU10)
66	BN11	A:FS: Add flour	=AND(BM10,COUNTIF($BM$9:BM10,TRUE)=1)
67	BO11	A:FxP:Get: Flour, Cupboard1	=BN10
68	BP11	A:RP:Melt butter	=BO10
69	BQ11	A:FxP:Get: Spoon, Tub	=BP10
70	BR11	A:FxP:Stir: Medium pan	=OR(BQ10,CB10,CE10,AND(BM10,COUNTIF($BM$9:BM10,TRUE)>=4,COUNTIF($BM$9:BM10,TRUE)<=10),CM10,DM10,AND(BM10,COUNTIF($BM$9:BM10, TRUE)>=12,COUNTIF($BM$9:BM10,TRUE)<=15),EE10)
71	BS11	A+C: Change ring: 2, -1.5	=AND(BR11,COUNTIF($BR$9:BR11,TRUE)=1)
72	BT11	A:RS: Add flour	=OR(BS10,AND(BM10,COUNTIF($BM$9:BM11,TRUE)=2))
73	BU11	A:FxP:Get: Spoon3, Drawer1	=AND(BT10,COUNTIF($BT$9:BT11,TRUE)=1)
74	BV11	A+C: Change ring: 1, .5	=AND(BA11,COUNTIF($BA$9:BA11,TRUE)=4)
75	BW11	A:FS: Cook pasta	=BV10
76	BX11	A:FS: Heat water	=BW10
77	BY11	A:FxP:Get: Large pan, Drawer2	=BX10
78	BZ11	A:FxP:Put: Water, Large pan	=BY10
79	CA11	A+C: Change ring: 3, 3	=BZ10
80	CB11	A:FxP:Put 1/3 flour, Medium pan	=AND(BT10,COUNTIF($BT$9:BT10,TRUE)=2)
81	CC11	A:RP: Cook onions	=AND(BL10,COUNTIF($BL$9:BL10,TRUE)=3)
82	CD11	A:RP: Add flour	=AND(BM10,COUNTIF($BM$9:BM10,TRUE)=3)
83	CE11	A:FxP:Put 2/3 flour, Medium pan	=CD10
84	CF11	A:RP: Cook meat	=AND(BL10,COUNTIF($BL$9:BL11,TRUE)=4)
85	CG11	A:FxP:Put: Meat, Frying pan	=CF10
86	CH11	A:RS: Cook pasta	=OR(AND(BA10,COUNTIF($BA$9:BA10,TRUE)=6),AND(I10,COUNTIF($I$9:I10,TRUE)=2),AND(BR10,COUNTIF($BR$9:BR10,TRUE)=10),AND(BA10,COUNTIF($BA$9:BA10,TRUE)=12),AND(BA10,COUNTIF($BA$9:BA10,TRUE)=13),AND(BA10,COUNTIF($BA$9:BA10,TRUE)=16))
87	CI11	A:RP: Heat water	=AND(CH10,COUNTIF($CH$9:CH10,TRUE)=1)
88	CJ11	A:FxP:Put: Lid, Large pan	=CI10
89	CK11	A:FS: Add milk	=AND(BR10,COUNTIF($BR$9:BR10,TRUE)=4)
90	CL11	A:FxP:Get Milk, Cupboard	=CK10
91	CM11	A:FxP:Put: 1/5 Milk, Medium pan	=OR(CL10,CN10,DL10)
92	CN11	A:RS: Add milk	=OR(AND(BR10,OR(COUNTIF($BR$9:BR10,TRUE)=5,COUNTIF($BR$9:BR10,TRUE)=7,COUNTIF($BR$9:BR10,TRUE)=8)),CZ10)
93	CO11	A:FS: Assemble	=AND(BA10,COUNTIF($BA$9:BA10,TRUE)=7)
94	CP11	A+C: Change ring: 0, 7	=CO10
95	CQ11	A+C: Change ring: 2, -0.5	=AND(BR10,COUNTIF($BR$9:BR10,TRUE)=9)
96	CR11	A:FP: Add pasta	=AND(CH10,COUNTIF($CH$9:CH10,TRUE)=2)
97	CS11	A:FxP:Get: Pasta, Cupboard	=CR10
98	CT11	A:FxP:Put: Pasta, Large pan	=CS10
99	CU11	A:FP: Add oil	=AND(CH10,COUNTIF($CH$9:CH10,TRUE)=3)
100	CV11	A:FxP:Put: Oil, Large pan	=CU10
101	CW11	A+C: Change ring: 3, -1	=OR(CV10,AND(CH10,COUNTIF($CH$9:CH10,TRUE)=5))
102	CX11	A:RS: Assemble	=OR(AND(I10,OR(COUNTIF($I$9:I10,TRUE)=3,COUNTIF($I$9:I10,TRUE)=5)),AND(BR10,OR(COUNTIF($BR$9:BR10,TRUE)=19,COUNTIF($BR$9:BR10,TRUE)=21)),EF10)
103	CY11	A:FxP:Get: Dish, Cupboard	=AND(CX10,COUNTIF($CX$9:CX10,TRUE)=1)
104	CZ11	A+C: Change ring: 2, -.75	=AND(BR10,COUNTIF($BR$9:BR10,TRUE)=12)
105	DA11	A:FS: Add tomatoes	=AND(BL10,COUNTIF($BL$9:BL10,TRUE)=7)
106	DB11	A:FxP:Get: Can opener, Drawer1	=DA10
107	DC11	A:RS: Add tomatoes	=AND(BL10,COUNTIF($BL$9:BL10,TRUE)=8)
108	DD11	A:FxP:Get: Canned toms, Cupboard	=DC10
109	DE11	A:FxP:Get: Puree, Cupboard	=DD10
110	DF11	A:FxP: Unpack Canned toms	=DE10
111	DG11	A:FxP:Put: Canned toms Frying pan	=DF10
112	DH11	A:FxP:Get: Spoon, Tub	=AND(CH10,COUNTIF($CH$9:CH10,TRUE)=4)
113	DI11	A:FxP: Stir, Large pan	=DH10
114	DJ11	A:RP: Add tomatoes	=AND(BL10,COUNTIF($BL$9:BL10,TRUE)=10)
115	DK11	A:FxP:Put: Puree, Frying pan	=DJ10
116	DL11	A:RP: Add milk	=AND(BM10,COUNTIF($BM$9:BM10,TRUE)=11)
117	DM11	A+C: Change ring: 2, 1.25	=AND(BR10,COUNTIF($BR$9:BR10,TRUE)=16)
118	DN11	A+C: Change ring: 1, .5	=AND(BL10,COUNTIF($BL$9:BL10,TRUE)=11)
119	DO11	A:FS: Prepare cheese	=AND(CX10,COUNTIF($CX$9:CX10,TRUE)=2)
120	DP11	A:FxP:Get: Bowl, Cupboard	=DO10
121	DQ11	A:FxP:Get: Grater, Drawer	=DP10
122	DR11	A:FxP:Get: Cheese, Fridge	=DQ10
123	DS11	A:RP: Prepare cheese	=AND(CX10,COUNTIF($CX$9:CX10,TRUE)=3)
124	DT11	A:FxP:Grate Cheese, Bowl	=DS10
125	DU11	A+C: Change ring: 3, -1	=AND(CH10,COUNTIF($CH$9:CH10,TRUE)=6)
126	DV11	A:FS: Assemble pasta	=AND(CX10,COUNTIF($CX$9:CX10,TRUE)=4)
127	DW11	A:FxP:Put: 3 sheets pasta, Dish	=OR(DV10,EI10,EQ10)
128	DX11	A:FS: Assemble pasta sauce	=AND(DW10,COUNTIF($DW$9:DW10,TRUE)=1)
129	DY11	A:FxP:Put: 1/3 pasta sauce, Dish	=OR(DX10,EM10,ER10)
130	DZ11	A::RP: Make pasta sauce	=AND(DY10,COUNTIF($DY$9:DY10,TRUE)=1)
131	EA11	A:FP:Add salt and pepper	=DZ10
132	EB11	A:FxP:Put: Salt & pepper, Frying pan	=EA10
133	EC11	A:RP:Make white sauce	=EB10
134	ED11	A:FP:Add salt and pepper	=EC10
135	EE11	A:FxP:Put: Salt & pepper, Medium pan	=ED10
136	EF11	A+C: Change ring: 2, -1.5	=AND(BR10,COUNTIF($BR$9:BR10,TRUE)=22)
137	EG11	A:FS: Assemble white sauce	=AND(CX10,COUNTIF($CX$9:CX10,TRUE)=5)
138	EH11	A:FxP:Put: 1/3 white sauce, Dish	=OR(EG10,EN10,ES10)
139	EI11	A:RS: Assemble pasta	=AND(EH10,COUNTIF($EH$9:EH10,TRUE)=1)
140	EJ11	A:RP:Make pasta sauce	=AND(DW10,COUNTIF($DW$9:DW10,TRUE)=2)
141	EK11	A+C: Change ring: 1, -2	=EJ10
142	EL11	A:RP: Assemble	=EK10
143	EM11	A:RS: Assemble pasta sauce	=EL10
144	EN11	A:RS: Assemble white sauce	=AND(DY10,COUNTIF($DY$9:DY10,TRUE)=2)
145	EO11	A:FS: Assemble cheese	=AND(EH10,COUNTIF($EH$9:EH10,TRUE)=2)
146	EP11	A:FxP:Put: 1/2 cheese, Dish	=OR(EO10,ET10)
147	EQ11	A:RP: Assemble pasta	=AND(EP10,COUNTIF($EP$9:EP10,TRUE)=1)
148	ER11	A:RP: Assemble pasta sauce	=AND(DW10,COUNTIF($DW$9:DW10,TRUE)=3)
149	ES11	A:RP: Assemble white sauce	=AND(DY10,COUNTIF($DY$9:DY10,TRUE)=3)
150	ET11	A:RP: Assemble cheese	=AND(EH10,COUNTIF($EH$9:EH10,TRUE)=3)
151	EU11	(Ring 1 level)	=AW11*3+BB11*-2+BV11*0.5+DN11*0.5+EK11*-2+EU10
152	EV11	(Ring 2 level)	=BK11*3+BS11*-1.5+CQ11*-0.5+CZ11*-0.75+DM11*1.25+EF11*-1.5+EV10
153	EW11	(Ring 3 level)	=CA11*3+CW11*-1+DU11*-1+EW10
154	EX11	(Ring level)	=SUM(EU11:EW11)
155	EY11	(Oven level)	=CP11*7+EY10
156	EZ11	C:Gas	=0.01*(EX11+EY11)*MINUTE(E11)
157	FA11	A:Abst-struct	=FA10
158	FB11	A:Phys-struct	=FB10
159	FC11	A:Abst-beh	=FC10+CM(AL11:ET11,$F$5:$ET$5)
160	FD11	A:Phys-beh	=FD10+CM(AL11:ET11,$F$6:$ET$6)
161	FE11	C:Abst-struct	=FE10
162	FF11	C:Phys-struct	=FF10+EZ11
163	FG11	C:Abst-beh	=FG10+CM(AL11:ET11,$F$7:$ET$7)
164	FH11	C:Phys-beh	=FH10+CM(AL11:ET11,$F$8:$ET$8)
165	FI11	Change in elapsed minutes	=MINUTE(E11)
166	FJ11	Elapsed minutes	=MINUTE(D11)
167	FK11	Poss change	=(3*EX11+3*EY11)*FI11
168	FL11	Cooker temp	=IF(COUNTIF($FI$9:FI11,1)=0,0,SUM($FK$9:FK11)/SUM($FI$9:FI11))
169	FM11	(From cooker)	=FM10+0.2*(FL11-FL10)*FI11
170	FN11	Kitchen temp	=$FN$9+FM11
171	FO11	Activity /event	=DI11+BA11+3*BR11
172	FP11	Activity /time	=IF(FI11=0,FP10,SUMIF($FJ$9:$FJ$210,FJ11,$FO$9:$FO$210))
173	FQ11	(From activity)	=FP11*0.05
174	FR11	A's temp	=35+(FN11*0.1)+FQ11
175	FS11	Activity /event	=3*(AS11+BA11+DI11+DT11)+10*BS11
176	FT11	Activity /time	=IF(FI11=0,FT10,SUMIF($FJ$9:$FJ$210,FJ11,$FS$9:$FS$210))
177	FU11	Comfort	=AND(FR11>36.5,FR11<37.5)
178	FV11	(Discomfort time)	=IF(FU11,0,MINUTE(E11)+FV10)
179	FW11	Agitation	=(COUNTIF(H11:$H$210,TRUE)+COUNTIF(J11:$J$210,TRUE))/(COUNTIF($H$9:$H$210,TRUE)+COUNTIF($J$9:$J$210,TRUE))
180	FX11	Flavour	
181	FY11	Presentation	
182	FZ11	Meal quality	=FZ10-0.01*FV11-0.01*FW11

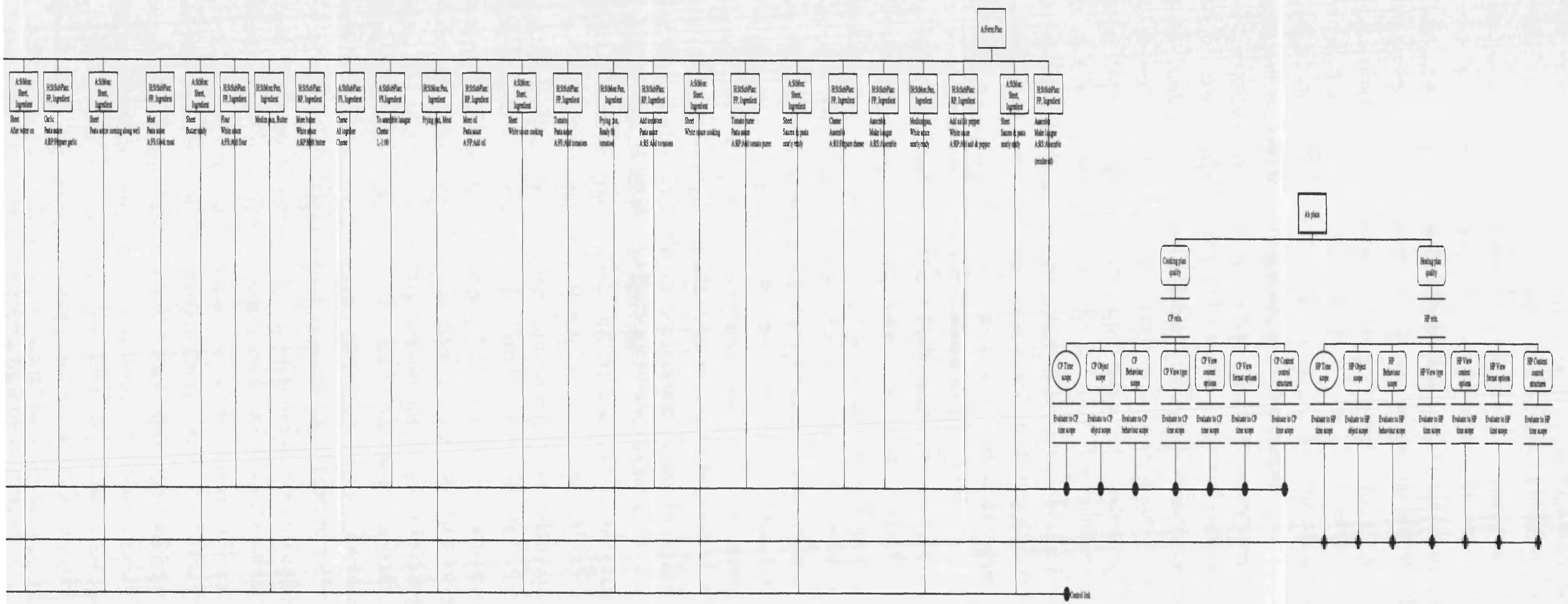
Figure P35

Cycle 2 Operationalisation
Actual Planning Diagram

Operationalisation 2
Actual Planning





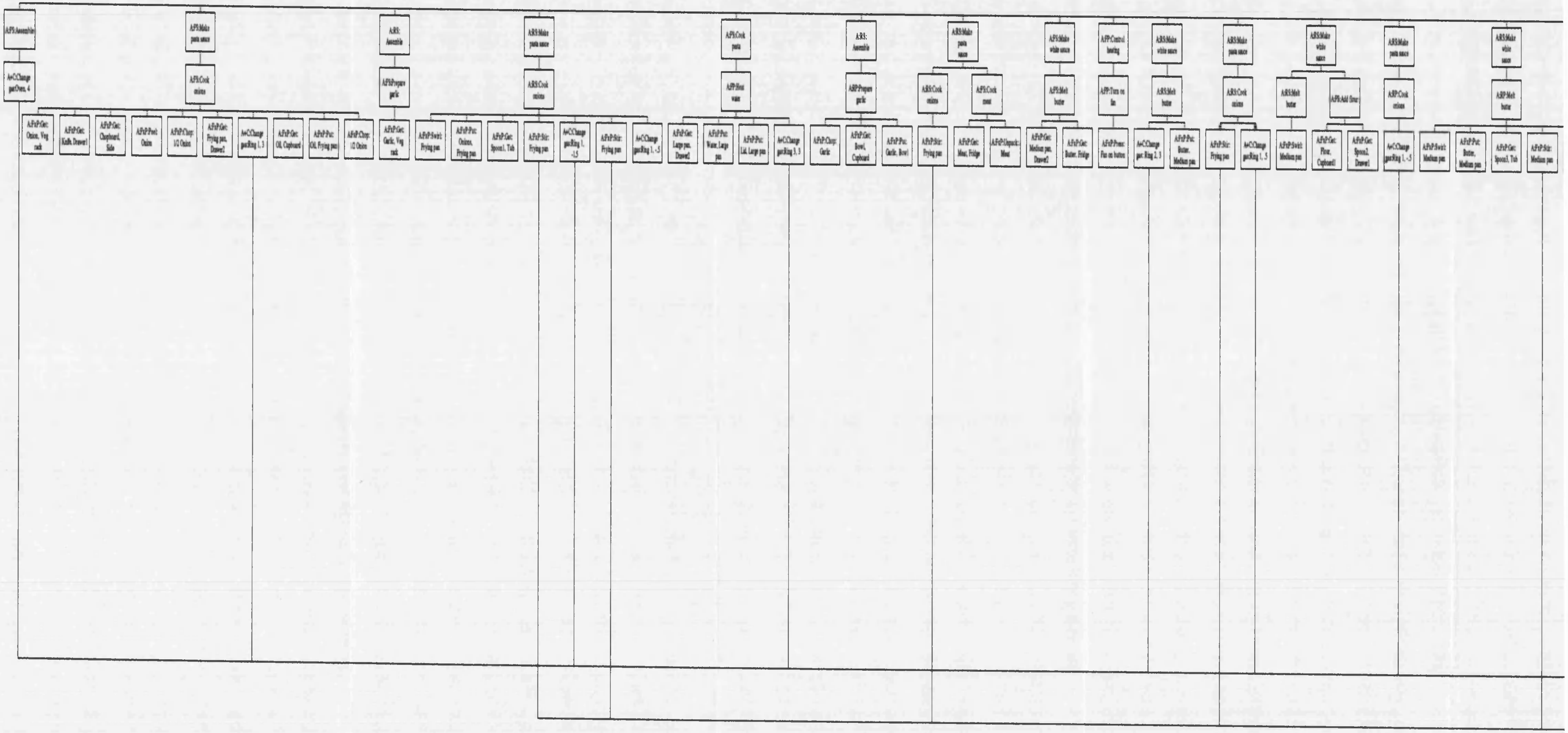


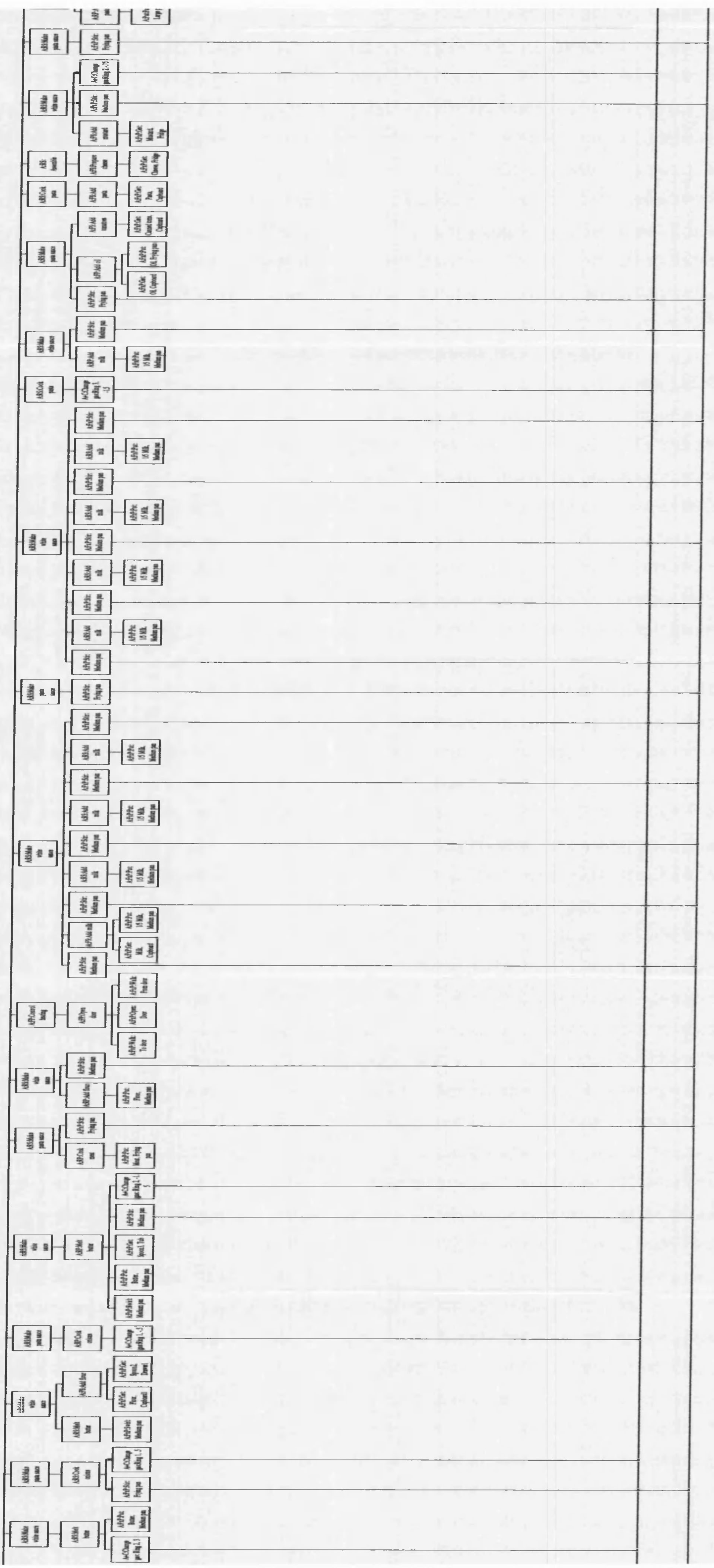
A:Form:Plan
Ab plans
Cooking plan quality
Heating plan quality
CP rela.
HP rela.
CP Time scope
CP Object scope
CP Behaviour scope
CP View type
CP View content options
CP View format options
CP Content control structures
HP Time scope
HP Object scope
HP Behaviour scope
HP View type
HP View content options
HP View format options
HP Content control structures
Evaluate to CP time scope
Evaluate to CP object scope
Evaluate to CP behaviour scope
Evaluate to CP time scope
Evaluate to CP time scope
Evaluate to CP time scope
Evaluate to CP time scope
Evaluate to HP time scope
Evaluate to HP object scope
Evaluate to HP behaviour scope
Evaluate to HP time scope
Evaluate to HP time scope
Evaluate to HP time scope
Evaluate to HP time scope
Control link

Figure P36

Cycle 2 Operationalisation
Actual Control Diagram

Operationalisation 2
Actual Control





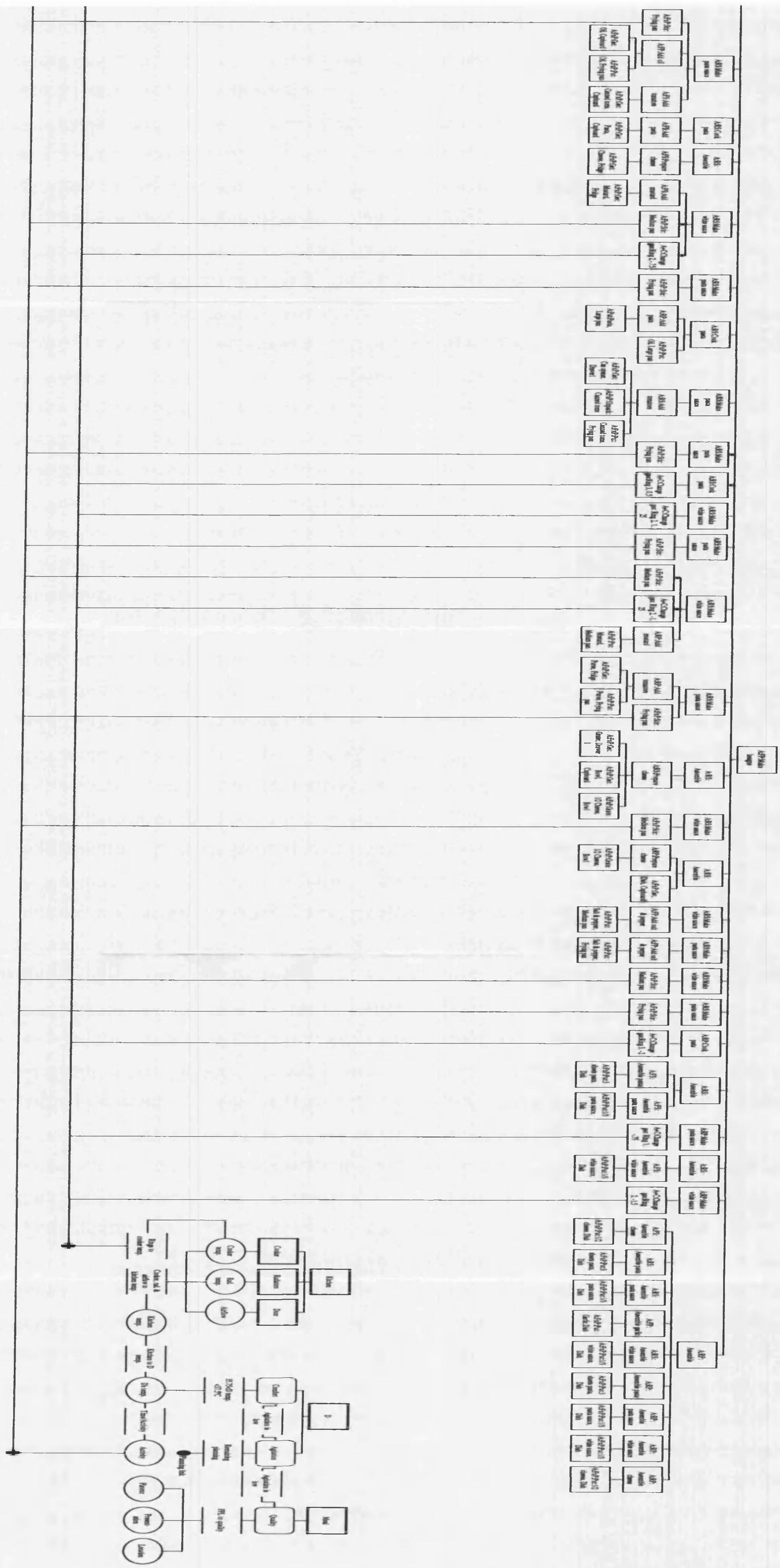


Figure P37

Cycle 2 Operationalisation

Actual Planning and Control Table

Figure P38

Cycle 2 Operationalisation
Actual Planning and Control
Formulae

Cycle 2 Operationalisation Table Formulae (1/1)

Actual.xls

	Loc	Heading	Formula in the 11th row
1	A11		
2	B11		3
3	C11		0.666666666666667
4	D11		=C11-$C$9
5	E11		=C11-C10
6	F11	A:FP: Plan	FALSE
7	G11	A:ShSubPlan: FP,Ingredient	=OR(F10,I10,AND(G10,OR(H10<14,H10=17,AND(H10>=19,H10<=23))),
8	H11	(count)	=G11+H10
9	I11	A:ShSubPlan: FP, Cooker heat	=AND(G10,OR(AND(H10>=14,H10<=16),H10=18,H10=24))
10	J11	A:ShSubPlan: FP, Cooling	=OR(AND(G10,H10=28),AND(J10,COUNTIF($J$9:J10, TRUE)>=1,COUNTIF($J$9:J10, TRUE)<=2))
11	K11	A:StMon: Sheet, Ingredient	=OR(AND(J10,COUNTIF($J$9:J10,TRUE)=3),BG10,AND(BN10,COUNTIF($BN$9:BN10,TRUE)=2),AND(CF10,COUNTIF($CF$9:CF10,TRUE)=1),CL10,AND(CC10,OR(CD10=3,CD10=11,CD10=12)),AND(DH10,COUNTIF($DH$9:DH10,TRUE)=1),AND(DO10,DP10=12),FC10)
12	L11	(Count)	=L10+K11
13	M11	A:StSubPlan: RP, Ingredient	=OR(AND(K10,OR(L10<=2,L10>=4)),O10,P10,Q10,AND(M10,N10=14))
14	N11	(Count)	=N10+M11
15	O11	A:StMon: Chopboard, Ingredient	=BL10
16	P11	A:StSubPlan: RS, Ingredient	=AND(K10,L10=3)
17	Q11	A:StMon: Pan, Ingredient	=OR(DJ10,AND(DO10,OR(DP10=7,DP10=16)),ET10)
18	R11	A:CDc: Seen	=CHOOSE(L11+COUNTIF($O$9:O11,TRUE)+COUNTIF($Q$9:Q11,TRUE)+1,"ó", "Sheet", "Sheet", "Chopboard", "Sheet", "Sheet", "Sheet",
19	S11	A:CDc: Ingredients	=CHOOSE(L11+COUNTIF($O$9:O11,TRUE)+COUNTIF($Q$9:Q11,TRUE)+1,"ó", "First ingredient", "After oven on", "Onions nearly
20	T11	A:CPSEc: Relationships	=T10
21	U11	(Ingred. count)	=COUNTIF($G$9:G11,TRUE)+N11+COUNTIF($P$9:P11,TRUE)+1
22	V11	(Cooker count)	=COUNTIF($I$9:I11,TRUE)+1
23	W11	(Cooling count)	=COUNTIF($J$9:J11,TRUE)+1
24	X11	(All count)	=U11+V11+W11-2
25	Y11	A:CDd: Desired ingredients	=INDEX({"ó","Lasagne","Pasta sauce","Onions","Meat","Tomatoes & puree","White sauce","Oil & flour","Milk","Pasta","Completed
26	Z11	A:CDd: Desired cooker heat	=CHOOSE(V11,"ó", "To cook lasagne", "To cook lasagne", "Cooked pasta", "White sauce", "Pasta sauce")
27	AA11	A:CDd: Desired cooling	=CHOOSE(W11, "ó", "Cook lasagne", "Cook lasagne", "Cook lasagne")
28	AB11	A:CWd: Desired control type	=INDEX({"ó","Cook lasagne","Cook pasta sauce","Cook pasta sauce","Cook pasta sauce","Cook pasta sauce","Cook white sauce","Cook
29	AC11	A:CWd: Desired behaviour	=INDEX({"ó","Lasagne","Make sauce","Onions","Meat","Tomatoes x 2","White sauce","Oil & flour","Milk","Cook pasta","All
30	AD11	S:Writing	=IF(H11<>H10,IF(AD10="ó",AC11,AD10&", "&AC11),AD10)
31	AE11	A:Abst-struct	=AE10+CH(R10,R11)+CH(S10,S11)+CH(Y10,Y11)+CH(Z10,Z11)+CH(AA10,AA11)+CH(AB10,AB11)+CH(AC10,AC11)
32	AF11	A:Phys-struct	=AF10
33	AG11	A:Abst-beh	=AG10+CM(F11:Q11,$F$5:$Q$5)
34	AH11	A:Phys-beh	=AH10+CM(F11:Q11,$F$6:$Q$6)
35	AI11	C:Abst-struc	=AI10+CH(AD10,AD11)
36	AJ11	S:Phys-struc	=AJ10+CH(AD10,AD11)
37	AK11	S:Abst-beh	=AK10+CM(F11:Q11,$F$7:$Q$7)
38	AL11	S:Phys-beh	=AL10+CM(F11:Q11,$F$8:$Q$8)
39	AM11	CP Time scope	0.2
40	AN11	CP Object scope	=IF(U11+V11<>U10+V10,Y11,AN10)
41	AO11	CP Behaviour scope	=IF(U11+V11<>U10+V10,AC11,AO10)
42	AP11	CP View type	=IF(LEFT(AO11,2)="A:", "Internal", "Sheet")
43	AQ11	CP View Content options	None
44	AR11	CP View format options	None
45	AS11	CP Content control structures	=IF(U11+V11<>U10+V10,AB11 & "; " & AC11,AS10)
46	AT11	CP quality	TRUE
47	AU11	HP Time scope	0.2
48	AV11	HP Object scope	=IF(W11<>W10,Y11,AV10)
49	AW11	HP Behaviour scope	=IF(W11<>W10,AC11,AW10)
50	AX11	HP View type	=IF(W11<>W10,"Sheet",AX10)
51	AY11	HP View content options	=IF(W11<>W10,"None",AY10)
52	AZ11	HP View format options	=IF(W11<>W10,"None",AZ10)
53	BA11	HP Content control structures	=IF(W11<>W10,AB11 & "; " & AC11,BA10)
54	BB11	HP quality	=IF(W11<>W10,TRUE,BB10)
55	BC11	Plan quality	TRUE
56	BD11	A:FP: Control	=AND(M10,N10=1)
57	BE11	A:FP:Make lasagne	=BD10
58	BF11	A:FS: Assemble	=BE10
59	BG11	A+C: Change gas: Oven, 6	=BF10
60	BH11	A:FS:Make pasta sauce	=AND(M10,N10=2)
61	BI11	A:FS: Cook onions	=BH10
62	BJ11	A:FxP:Get: Onion, Veg rack	=BI10
63	BK11	A:FxP:Get: Knife, Drawer1	=BJ10
64	BL11	A:FxP:Get: Chopboard, Side	=BK10
65	BM11	A:FxP: Peel: Onion	=AND(M10,N10=3)
66	BN11	A:FxP: Chop, 1/2 onion	=OR(BM10,AND(BR10,COUNTIF($BR$9:BR10,TRUE)=1))
67	BO11	A:FxP:Get: Frying pan, Drawer 2	=AND(BN10,COUNTIF($BN$9:BN11,TRUE)=1)
68	BP11	A+C: Change ring: 1, 3	=BO10
69	BQ11	A:FxP:Get: Oil, Cupboard	=OR(BP10,EH10)
70	BR11	A:FxP:Put: Oil, Frying pan	=BQ10
71	BS11	A:RS: Assemble	=OR(AND(M10,OR(N10=4,N10=15,N10=16)),EL10,FR10,FX10)
72	BT11	(Count)	=BT10+BS11
73	BU11	A:FS: Prepare garlic	=AND(BS10,BT10=1)
74	BV11	A:FxP:Get: Garlic, Veg rack	=BU10
75	BW11	A:RS:Make pasta sauce	=OR(BV10,CQ10,AND(DF10,COUNTIF($DF$9:DF10,TRUE)=1),DK10,DQ10,AND(M10,OR(N10=10,AND(N10>=12,N10<=13))),AND(DO10,
76	BX11	(Count)	=BX10+BW11
77	BY11	A:RS: Cook onions	=AND(BW10,BX10<=3)
78	BZ11	A:FxP: Swirl: Frying pan	=AND(BY10,COUNTIF($BY$9:BY10,TRUE)=1)
79	CA11	A:FxP:Put: Onions, Frying pan	=BZ10
80	CB11	A:FxP:Get: Spoon1, Tub	=CA10
81	CC11	A:FxP:Stir: Frying pan	=OR(CB10,CE10,AND(BY10,OR(COUNTIF($BY$9:BY10,TRUE)=2,COUNTIF($BY$9:BY10,TRUE)=3)),DS10,AND(BW10,OR(AND(BX10>=6,BX
82	CD11	(Count)	=CD10+CC11
83	CE11	A+C: Change ring: 1, -1.5	=AND(CC10,CD10=1)
84	CF11	A+C: Change ring: 1, -.5	=OR(AND(CC10,CD10=2),DL10)
85	CG11	A:FS: Cook pasta	=AND(M10,N10=5)
86	CH11	A:FP: Heat water	=CG10
87	CI11	A:FxP:Get: Large pan, Drawer2	=CH10
88	CJ11	A:FxP:Put: Water, Large pan	=CI10
89	CK11	A:FxP:Put: Lid, Large pan	=CJ10
90	CL11	A+C: Change gas: Ring 3, 3	=CK10
91	CM11	A:RS: Assemble	=AND(M10,N10=6)
92	CN11	A:FS: Prepare garlic	=CM10
93	CO11	A:FxP:Chop: Garlic,	=CN10
94	CP11	A:FxP:Get: Bowl, Cupboard	=CO10
95	CQ11	A:FxP:Put: Garlic, Bowl	=CP10
96	CR11	A:FS: Cook meat	=AND(M10,N10=7)
97	CS11	A:FxP:Get: Meat, Fridge	=CR10
98	CT11	A:FxP: Unpack: Meat	=CS10
99	CU11	A:FS:Make white sauce	=CT10
100	CV11	A:FS: Melt butter	=CU10
101	CW11	A:FxP:Get: Medium pan, Drawer2	=CV10
102	CX11	A:FxP:Get: Butter, Fridge	=CW10
103	CY11	A:FP: Control heating	=CX10
104	CZ11	A:FP: Turn on fan	=CY10
105	DA11	A:FxP: Press fan on button	=CZ10
106	DB11	A:RS:Make white sauce	=OR(DA10,DG10,AND(G10,H10=28),AND(CC10,OR(CD10=5,CD10=6,CD10=10)),DZ10,EF10,EN10,EY10,AND(FJ10,COUNTIF($FJ$9:FJ10,
107	DC11	(Count)	=DC10+DB11
108	DD11	A:RS: Melt butter	=AND(DB10,OR(DC10=1,DC10=2))
109	DE11	A+C: Change ring, 2, 3	=AND(DD10,COUNTIF($DD$9:DD10,TRUE)=1)
110	DF11	A:FxP:Put: Butter, Medium pan	=OR(DE10,AND(DH10,COUNTIF($DH$9:DH10,TRUE)=2))
111	DG11	A+C: Change ring: 1,.5	=AND(CC10,CD10=4)
112	DH11	A:FxP:Swirl: Medium pan	=OR(AND(DD10,COUNTIF($DD$9:DD10,TRUE)=2),DM10)
113	DI11	A:FS: Add flour	=AND(M10,N10=8)
114	DJ11	A:FxP:Get: Flour, Cupboard1	=DI10
115	DK11	A:FxP:Get: Spoon2, Drawer1	=AND(M10,N10=9)
116	DL11	A:RP: Cook onions	=AND(BW10,BX10=4)
117	DM11	A:RP: Melt butter	=AND(DB10,DC10=3)
118	DN11	A:FxP:Get: Spoon3, Tub	=AND(DF10,COUNTIF($DF$9:DF10,TRUE)=2)
119	DO11	A:FxP:Stir: Medium pan	=OR(DN10,DU10,AND(DB10,OR(DC10=5,DC10=6,DC10=10,DC10=11,DC10=13)),EC10,EP10)
120	DP11	(Count)	=DP10+DO11
121	DQ11	A+C: Change ring: 2, -1.5	=AND(DO10,DP10=1)
122	DR11	A:RP: Cook meat	=AND(BW10,BX10=5)
123	DS11	A:FxP:Put: Meat, Frying pan	=DR10
124	DT11	A:RP: Add flour	=AND(DB10,DC10=4)
125	DU11	A:FxP:Put: Flour, Medium pan	=DT10
126	DV11	A:FP: Control heating	=AND(DO10,DP10=2)
127	DW11	A:FP: Open door	=DV10
128	DX11	A:FxP: Walk: To door	=DW10
129	DY11	A:FxP: Open: Door	=DX10
130	DZ11	A:FxP: Walk: From door	=DY10
131	EA11	A:FS: Add milk	=AND(DO10,DP10=3)
132	EB11	A:FxP:Get Milk, Cupboard	=EA10
133	EC11	A:FxP:Put: 1/5 Milk, Medium pan	=OR(EB10,ED10,EG10)
134	ED11	A:RS: Add milk	=AND(DO10,OR(AND(DP10>=4,DP10<=6),AND(DP10>=8,DP10<=11)))
135	EE11	A:RS: Cook pasta	=OR(AND(M10,N10=11),EJ10,AND(CC10,CD10=8),EX10)
136	EF11	A+C: Change ring: 3, -1.5	=AND(EE10,COUNTIF($EE$9:EE10,TRUE)=1)
137	EG11	A:RP: Add milk	=AND(DB10,DC10=7)
138	EH11	A:FP: Add oil	=AND(CC10,CD10=7)
139	EI11	A:FS: Add tomatoes	=AND(BR10,COUNTIF($BR$9:BR10,TRUE)=2)
140	EJ11	A:FxP:Get: Canned toms, Cupboard	=EI10
141	EK11	A:FS: Add pasta	=AND(EE10,COUNTIF($EE$9:EE10,TRUE)=2)
142	EL11	A:FxP:Get: Pasta, Cupboard	=EK10
143	EM11	A:FS: Prepare cheese	=AND(BS10,BT10=2)
144	EN11	A:FxP:Get: Cheese, Fridge	=EM10
145	EO11	A:FS: Add mustard	=AND(DB10,DC10=8)
146	EP11	A:FxP:Get: Mustard, Fridge	=EO10
147	EQ11	A+C: Change ring: 2,-.75	=OR(AND(DO11,DP11=14),GA10)
148	ER11	A:RP: Add pasta	=AND(EE10,COUNTIF($EE$9:EE10,TRUE)=3)
149	ES11	A:FxP:Put: Pasta, Large pan	=ER10
150	ET11	A:FxP:Put: Oil, Large pan	=ES10
151	EU11	A:RS: Add tomatoes	=AND(BW10,BX10=9)
152	EV11	A:FxP:Get: Can opener, Drawer1	=EU10
153	EW11	A:FxP: Unpack Canned toms	=EV10
154	EX11	A:FxP:Put: Canned toms Frying pan	=EW10
155	EY11	A+C: Change ring: 3,.5	=AND(EE10,COUNTIF($EE$9:EE10,TRUE)=4)
156	EZ11	A+C: Change ring: 2, 1.25	=AND(DB10,DC10=9)
157	FA11	A+C: Change ring: 2, -1.25	=AND(DB10,DC10=10)
158	FB11	A:RP: Add mustard	=FA10
159	FC11	A:FxP:Put: Mustard, Frying pan	=FB10
160	FD11	A:RP: Add tomatoes	=AND(BW10,BX10=12)
161	FE11	A:FxP:Get: Puree, Fridge	=FD10
162	FF11	A:FxP:Put: Puree, Frying pan	=FE10
163	FG11	A:RS: Prepare cheese	=AND(BS10,BT10=3)
164	FH11	A:FxP:Get: Grater, Drawer 1	=FG10
165	FI11	A:FxP:Get: Bowl, Cupboard	=FH10
166	FJ11	A:FxP:Grate 1/2 Cheese, Bowl	=OR(FI10,FK10)
167	FK11	A:RP: Prepare cheese	=AND(BS10,BT10=4)
168	FL11	A:FxP:Get: Dish, Cupboard	=AND(FJ10,COUNTIF($FJ$9:FJ10,TRUE)=2)
169	FM11	A:FP:Add salt and pepper	=AND(DB10,DC10=12)
170	FN11	A:FxP:Put: Salt & pepper, Medium pan	=FM10
171	FO11	A:FP:Add salt and pepper	=AND(BW10,BX10=13)
172	FP11	A:FxP:Put: Salt & pepper, Frying pan	=FO10
173	FQ11	A:RP: Cook pasta	=AND(M10,N10=17)
174	FR11	A+C: Change ring: 3, -2	=FQ10
175	FS11	A:FS: Assemble pasta	=AND(BS10,BT10=5)
176	FT11	A:FxP:Put: 3 sheets pasta, Dish	=OR(FS10,GE10,GJ10)
177	FU11	A:FS: Assemble pasta sauce	=AND(FT10,COUNTIF($FT$9:FT10,TRUE)=1)
178	FV11	A:FxP:Put: 1/3 pasta sauce, Dish	=OR(FU10,GF10,GK10)
179	FW11	A::RP: Make pasta sauce	=AND(FV10,COUNTIF($FV$9:FV10,TRUE)=1)
180	FX11	A+C: Change ring: 1, -1	=FW10
181	FY11	A:FS: Assemble white sauce	=AND(BS10,BT10=6)
182	FZ11	A:FxP:Put: 1/3 white sauce, Dish	=OR(FY10,GI10,GL10)