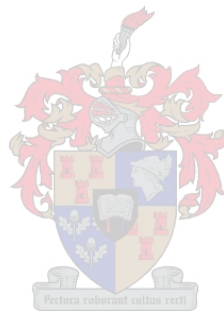


Selecting and Tailoring Design Methodologies in the Form of Roadmaps for a Specific Development Project.

Friedrich Hagen Martin Nieberding



Dissertation presented for the degree of Doctor of Engineering (Industrial)
at the Stellenbosch University

Promoters:

Niek du Preez, Department of Industrial Engineering
Anton Basson, Department of Mechanical Engineering

January 2010

Declaration

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the owner of the copyright thereof (unless to the extent explicitly otherwise stated) and that I have not previously in its entirety or in part submitted it for obtaining any qualification.


Signed


Date

Abstract

This dissertation investigates answers to the question: "How does one decide on the approach to implement when planning, managing and executing a development project?". By examining the prescriptive models of the development process, as proposed by design science, the principle characteristics are identified, that these models try to enforce. The need for tailoring the development process to the context of the project is highlighted by contrasting these prescriptive models to industrial practice and the corresponding descriptive models of the development process. A framework of contextual variables is proposed, which facilitates the tailoring of the approach to the project by defining the project methodology. The usefulness of this contextual framework is verified by means of case studies. Finally the dissertation proposes the use of a roadmap to define the project methodology at the beginning of the project. By packaging the project methodology in the form of a roadmap, implemented in a collaborative computer environment, it can be refined during the execution of the project to suit the information requirements of the project.

Samevatting

Hierdie verhandeling ondersoek antwoorde tot die vraag: "Hoe besluit 'n mens watter benadering om vir die beplanning, bestuur en deurvoering van 'n ontwikkelingsprojek toe te pas?". Deur die voorskriftelike modelle van die ontwikkelingsproses wat deur die ontwikkelingswetenskap voorgestel word, te ondersoek, word die grondbeginsels bepaal wat diïë; modelle poog om af te dwing. Die behoefte om die ontwikkelingsproses by die projek se konteks aan te pas, word duidelik uitgebeeld deur die kontras tussen hierdie voorskriftelike modelle van die ontwikkelingsproses en industriïë;le praktyk, sowel as die ooreenkomstige beskrywende modelle. 'n Raamwerk word voorgestel om die konteks van 'n projek te beskryf. Die aanpassing van die benadering tot die projek word sodoende deur die definisie van 'n projekmetodologie vergemaklik. Gevallestudies bevestig die nuttigheid van hierdie raamwerk. Die verhandeling sluit met die slotsom af dat 'n padkaart gebruik kan word om die projekmetodologie aan die begin van die projek te bepaal. Deur die projekmetodologie in 'n padkaart te verpak wat in 'n rekenaargesteuende spanomgewing gebruik kan word, kan dit gedurende die uitvoering van die projek gewysig word volgens die inligtingsbehoefte van die projek.

Preface

From the time that the author of this dissertation decided to study mechanical engineering, it was clear to him that the focus would be design and development of products. What kind of products was always of secondary importance. The primary focus was, and still is, the process which starts as a desire in the heart, evolves into a picture in the mind and ends up as a physical object, that can be touched and operated. Consequently the author's career has exposed him to a fairly wide range of industries, ranging from electrical switchgear, through a first tier supplier to the major automotive manufacturers, to the development and manufacture in the special automotive industry, which includes military, cash-in-transit and law-enforcement vehicles.

From a perspective of a broad overview, the development process in the various companies was very similar. And yet, it is the differences among these processes that is intriguing. Why are they different? Should or could they be the same? If they are supposed to be different, how does one decide what the process should be like, given a specific project? The need to understand more about design and development, answering some of these questions in the process, was the motivation to start the five year journey through the research that ended with this dissertation.

The first two chapters provide background for the reader to understand the context surrounding this work. The first chapter discusses the research, its aims, its relationship to existing work in the field, and the structure of this document. The second one provides a broad overview of engineering design and development by discussing the typical characteristics of this process.

The second part of the dissertation engages the subject matter in more detail by analysing models of the development process (Chapter 3) that have been proposed over the last sixty years or so. The fourth chapter contrasts these theoretical models against the reality of industrial practise and comes to the conclusion that it is sensible to tailor the development process to the context of each specific development project.

Having come to this conclusion, the next part of the dissertation, provides a means to crystallise the amorphous concept of "project context" into a limited set of variables, which highlight the often conflicting requirements and characteristics of development projects. Chapter 5 discusses each of the variables of this contextual framework and their inter-relationships. To illustrate and verify the framework, the sixth chapter applies the framework to a couple of case studies.

The fourth part of the dissertation (Chapter 7) proposes a way of applying the contextual framework in a practical manner. In accordance with the original intent for this work, the proposed implementation does not involve elaborate IT infrastructure. It can therefore benefit smaller companies as much as large ones.

The last part of the dissertation summarises the work described in the dissertation, by answering the specific research questions posed in Chapter 1, referring to the relevant sections of the dissertation where required.

A final comment regarding the scope of the research described in this dissertation: the words "product development" conjures up different images in the minds of different people. For one it is the legal process of registering patents and trademarks in order to prevent the product being copied. For another, it is the business process of landing a deal, making it viable from a cash-flow perspective, ensuring it is profitable in the long run, and finally executing the project. It is true - all of these are valid perspectives of "product development". However, in research like this one cannot address all aspects of a complex and intricate subject as "product development". The author has therefore decided, with intent, to concentrate on the engineering process, and how it is influenced by the context that surrounds it, including the legal, business, finance, resource and human aspects. In the world of engineering design and development, the engineering process is intrinsically linked to the management of the project. It is for this reason that the dissertation also addresses project management issues, especially in Chapter 7.

The following statement is the author's perspective on engineering design and development and the theme of this research: **"Product development is where design collides with reality"**.

Acknowledgements

Although my passion for and interest in engineering in general and product development in particular has caused me to drive this project from conception to completion, it would not have been possible for me to do so without the assistance of so many other people. Some have played a very active role in the process, others have helped by just being there. It is impossible to mention all of the individuals and institutions that have influenced this work by name, and therefore I ask for understanding and pardon of those which I do not mention explicitly.

First and foremost, I would like to thank my family, my wife, Cora, and my children, Gregor and Nadia, for accepting my pre-occupation with this work for five long years with so much patience and grace. To quote a classic: "I'll be back!".

Also, I thank my parents, Fritz and Doris, who have made this journey possible by laying the foundation of my academic career by affording me the opportunity to achieve my bachelor's and master's degrees in mechanical engineering.

Staying in the realm of academia, I thank Niek du Preez and Anton Basson for their endless patience and guidance. Without their constant support it would not have been possible for me to get through the dark times when the option of giving up seemed a very reasonable alternative. I am also extremely grateful for the assistance I have received from others at the University: Karina Smith, who always arranged my visits to Stellenbosch, Anel Uys, who assisted me with my bursary and related matters, Antje Gildenhuis, who tirelessly obtained reference material from all corners of the globe for me, and Minnaar Pienaar, who was always willing to assist with all the administration related to my being a student at the University of Stellenbosch.

During the process of gathering information for my research, I have been extremely fortunate and grateful to have obtained assistance from all over the world. First and foremost, Eric Lutters of the University of Twente, Netherlands, who was always willing to discuss the various topics of my research and provide valuable insight. Also Bernard Schlegel, a friend of the family, who provided me with access to much of the German literature on the subject of product development.

At times I would come across a reference to a doctoral dissertation, which would be of interest to my work. On the off-chance that I might receive a reply, I would write to the author, asking for an electronic copy of their dissertation. It amazed me that I would always get a positive response - without fail! A special thanks to the following authors for their help: Graham Thompson, Denis Cavallucci, Lucienne Blessing and Crispin Hales.

Moving from academia to industry, there are three organisations that have played major roles in my research: BAE Land Systems OMC, Mechanology and Indutech. For allowing me to use actual industrial projects as case studies for my research, my appreciation goes to the managers I reported to at BAE Land Systems OMC and Mechanology. The owner of Mechanology, Craig Savides, has been particularly supportive and exceptionally patient, especially considering the large amount of study leave required to attend conferences and regular visits to the University in Stellenbosch. Special thanks goes to Niek du Preez and his team at Indutech for all the assistance I have received over the years. Not only for the use of EDENTM to implement the roadmaps I discuss in Chapter 7, but also for the friendship and support that was so crucial as a sounding board for the concepts and ideas generated during the research described in this dissertation. In this respect, my sincere gratitude goes especially to Elna du Preez, Ursula Seconds, Louis Louw, Wilhelm Uys, Riaan Brand, Marc-Allen Johnson and Bernard Katz.

Lastly, but not least, I would like to thank the National Science Foundation for the bursary in support of my studies at the University of Stellenbosch.

Table Of Contents

1	Introduction	1
1.1	Background	2
1.2	Definitions	2
1.3	Aim of the Research Project	3
1.4	Target Environment	4
1.5	Research Questions	4
1.6	Research Taxonomy	4
1.7	How and Where Does this Research Fit into Design Science?	5
1.7.1	Research Categories According to Cross	5
1.7.2	The Taxonomical Framework of Horváth and Vergeest	5
1.7.3	The Research Areas of Design Science by Hubka and Eder	7
1.8	Structure of this Dissertation	8
2	Engineering Design and Development	11
2.1	What is Design in General and Engineering Design in Particular?	12
2.2	Product Life-Cycles And Product Development	16
2.3	Product Life-cycles, Methodologies, and Design Methods	17
3	Theoretical Models of the Development Process	21
3.1	Prescriptive Models	22
3.1.1	Similarities of Prescriptive Models of the Development Process	22
3.1.2	Differences Among Prescriptive Development Process Models	24
3.1.3	Conclusions	26
3.2	Principle Development Rules	28
3.3	Descriptive Models	28
3.3.1	Similarities of Descriptive Models of the Development Process	29
3.3.2	Differences Among Descriptive Models of the Development Process	29
3.4	Prescriptive Models Versus Descriptive Models - Conclusions	29
4	Analysis of the Theoretical Models in the Light of Practical Application	31
4.1	What Happens in Practice?	32
4.2	The Need for Tailoring or Adaptation of Development Models	34
4.3	Project Methodology: Planning, Managing and Executing Development Projects	36
5	Internal and External Influences on the Development Process	39
5.1	Existing Frameworks	40

5.2	Description of Factors Influencing the Development Process	41
5.2.1	Variables Related to the Organisation	41
5.2.2	Variables Related to the Project	46
5.2.3	Variables Related to the Product	50
5.2.4	Variables Related to the Personnel	53
5.3	Framework for the Contextualisation of the Development Life-cycle	55
5.3.1	Variables Related to Size	56
5.3.2	Variables Related to Type	56
5.3.3	Variables Related to Complexity	56
5.3.4	Variables Related to Maturity	58
5.3.5	Variables Related to Capacity and Capability	58
5.4	Role players	59
5.5	Summary and Conclusions	59
6	Case Studies to Evaluate the Framework of Contextual Factors	61
6.1	Case Study A	62
6.1.1	Introduction to the Industry Project and its Context	62
6.1.2	Application of the Framework to the Development Project	63
6.1.3	The Execution of the Project	67
6.1.4	Conclusion and Discussion	70
6.2	Case Study B	71
6.2.1	Introduction to the Industry Project and its Context	71
6.2.2	Values Assigned to the Variables of the Framework	74
6.2.3	The Resulting Project Methodology	76
6.2.4	The Execution of the Project	77
6.2.5	Conclusion and Discussion	78
6.3	Case Study C	79
6.4	Case Study D	80
6.5	Summary and Conclusions	80
7	Generating a Formal Project Methodology	81
7.1	Process Orientation Versus Information Orientation	82
7.1.1	Process Orientated Integration and Control	82
7.1.2	Information Orientated Integration and Control	82
7.1.3	Conclusions and Summary	83
7.2	The Process of Formulating a Formal Project Methodology	84
7.3	Using the Contextual Framework to Tailor the Methodology for a Specific Development Project	84
7.3.1	Project Background	85
7.3.2	Project Enablers	86
7.3.3	Project Structure	86
7.4	Introduction to the Concept of a Roadmap	90
7.5	A Formal Project Methodology as a Roadmap	91
7.6	A Roadmap to Generate a Project Methodology	92
7.7	Conclusions	92

8	Research Findings and Conclusions	97
8.1	Answers to Research Questions	98
8.1.1	Given a Specific Project, with Specific Resources within a Given Environment, How Does One Select and Construct a Design Methodology from a Repository of Templates?	98
8.1.2	Can Benefit be Obtained by Combining Design Methods to Form a Project Methodology?	98
8.1.3	Can Methods be Established to Define a Methodology Optimised for a Specific Project?	99
8.1.4	How do I Specify, Choose and Construct a Development Methodology?	99
8.1.5	How do I Measure the Success or Appropriateness of My Choices?	99
8.1.6	How do I Comprehensively Quantify My Needs with Respect to a Development Methodology?	99
8.1.7	Can a Methodology be Represented in a Roadmap?	100
8.1.8	Can a Roadmap, Implemented in a Design Environment, Facilitate the Engineering Development Process?	100
8.1.9	What Attributes Must a Roadmap Have so That it Can Represent a Design Methodology?	100
8.1.10	What Attributes Must a Design Environment Have so That a Roadmap Can be Applied to a Development Project?	100
8.2	Limitations of This Research	100
8.3	Applications	101
8.4	Suggestions for Future Research	101
8.5	Overall Summary and Conclusions	101
A	Selection of Design Methods	103
A.1	Selection of Design Methods by Project Phase	104
A.2	Selection of Design Methods According to Need	106
B	An Overview of Various Prescriptive Models of the Development Process	115
B.1	Cross's Model of the Design Process	116
B.2	French's Model of the Design Process	116
B.3	Archer's Model of the Design Process	117
B.4	Development Model for Large Infrastructure Projects	118
B.5	Suireg's Model of the Design Process	118
B.6	The Design Process According to Pahl and Beitz	118
B.7	The Model of the Development Process as Defined by VDI 2221	119
B.8	Ullman's Model of the Development Process	121
B.9	The Systems Engineering Approach to the Development Process	123
B.9.1	Requirements Engineering	125
B.10	Concurrent Engineering	125
B.11	Integrated Product Development	126
B.12	Pugh's Core-based Model	127
C	An Overview of Various Descriptive Models of the Development Process	129
C.1	Models of the Design Process by Hillier, Drake and March	130
C.2	Model by Hybs and Gero	130
C.3	Model by Johannes	131
C.4	Model by Reymen	132
C.5	Model by Stempfle and Badke-Schaub	133

C.6 Model by Günther and Ehrlenspiel	134
C.7 Model by Hansen and Ahmed	134
D Characteristics of Companies and Their Development Processes	137
D.1 Influences on a Design Project by Wallace and Hales	138
D.2 Product Properties by Tjalve	138
D.3 Company Classification by Barber and Hollier	138
D.4 The Seven Dimensions of a Company's Product Development by Andreasen et al.	141
D.5 Parameters Affecting the Design Process According to Ehrlenspiel	141
D.6 Field Study of Design by Hykin and Laming	141
D.7 Company Classification by Maffin	142
References	145
Index	151

List of Figures

1.1	Framework of Concepts Addressed in this Research	5
1.2	Contextual Categories for Design Science [66]	6
1.3	Research Areas in Design Science According to Hubka and Eder [68]	8
1.4	The Technical Process According to Hubka and Eder [68]	8
1.5	The Structure of this Dissertation	9
2.1	The Design Process According to Andreasen [9]	12
2.2	Design Freedom and Knowledge During the Development Process [123]	13
2.3	Product Costs as a Function of Time During the Development Process According to Schulz [110]	14
2.4	The Inter-relationship of Synthesis and Analysis During the Development Process as Depicted by Bichlmaier [18]	14
2.5	The Product Life-Cycle Adopted from Krause [80]	16
2.6	The Systems Engineering Product Life-cycle [19]	17
2.7	Ullman's Product Life-cycle [123]	18
3.1	Solution / Problem Hierarchy [125]	23
3.2	Degree of "Structuredness" for Various Prescriptive Models of the Development Process	25
3.3	Matrix Showing Which Phases of the Product Life-cycle are Addressed by Various Prescriptive Models of the Development Process	26
3.4	The Need for a Prescriptive Model of the Development Process Increases as the Level of Novelty, Risk and Complexity Increases	30
4.1	Productive Solution Generation [47]	32
4.2	Corrective Solution Generation [47]	33
4.3	Solution Strategies Adapted From Ehrlenspiel and Dylla [47]	34
4.4	Relationship among Methodology, Methods and Models	37
4.5	Establishing a Design Methodology According to Maffin [86]	38
5.1	The Contextual Framework Adapted from Maffin [86]	41
5.2	Innovation Continuum According to Thomond and Lettice [119]	49
5.3	Level of Innovation According to Frost [53]	50
5.4	Level of Complexity of Various Designed Entities According to Frost [53]	51
5.5	Correlation Between Project/Product Complexity and Design Strategy [86]	52
5.6	Amenability of Some Designed Entities to Decomposition into Sub-systems According to Frost [53]	53
5.7	The Framework of Factors Affecting the Product Life-cycle	56
5.8	Relationship Between Complexity and Structuredness	58
6.1	The Development Process Adopted by Mechanology for this Project	66

6.2	The Original Project Plan	68
6.3	The Project Plan Used for the Execution of the Project	68
6.4	The Process Followed for this Project	77
6.5	The Author's Proposed Development Process	79
7.1	The Development Process	83
7.2	The Process of Formulating a Project Methodology	85
7.3	Categories of Contextual Variables	85
7.4	Roadmap Implemented in EDEN™	91
A.1	Eder's Method Matrix [46]	108
A.2	Methods for Analysis and Goal-setting Recommended by VDI 2221 [125]	109
A.3	Methods for Generation of Ideas Recommended by VDI 2221 [125]	110
A.4	Methods for Cost Estimation Recommended by VDI 2221 [125]	111
A.5	Methods for Assessment and Decision Making Recommended by VDI 2221 [125]	112
A.6	Integrated Methods Recommended by VDI 2221 [125]	113
B.1	The Model of the Development Process by Cross [32]	116
B.2	French's Model of the Product Development Process [32]	117
B.3	Archer's Model of the Product Development Process [32]	117
B.4	Development Process for a Construction Project According to the IDC [36]	118
B.5	Suireg's Model of the Product Development Process [117]	119
B.6	The Product Development Process According to Pahl and Beitz [97]	120
B.7	The Product Development Process According to the VDI [125]	120
B.8	The Mechanical Design Process According to Ullman [123]	122
B.9	The Systems Engineering Product Acquisition Process [19]	124
B.10	The Feedback in the Systems Engineering Process [19]	125
B.11	Process of Integrated Product Development by Andreasen and Hein [9]	126
B.12	The Product Synthesis Process According to Andreasen and Hein [9]	127
B.13	Pugh's Core-based Model [102]	128
C.1	March's Model of the Product Development Process [32]	130
C.2	Model of Design Process According Hybs and Gero [70]	131
C.3	A Design Situation as a State According to Reymen et al. [1]	132
C.4	Transitions from One State to Another According to Reymen et al. [1]	132
C.5	Generic Model of Design Activities by Stempfle and Badke-Schaub [115]	133
C.6	Descriptive Models of the Development Process Presented by Stempfle and Badke-Schaub [115]	134
C.7	Model of the Design Process by Hansen [63]	135
D.1	The Context of a Design Project According to Wallace and Hales [127]	139
D.2	Product Properties According to Tjalve [121]	140
D.3	The Seven Dimensions of a Company's Product Development System by Andreasen et al. [10]	141
D.4	Ehrlenspiel's Parameters Influencing the Design Process [47]	142
D.5	Key Dimensions to Describe a Company by Maffin [87]	143

List of Tables

3.1	Equivalent Terminology of the Various Models of the Development Process	24
3.2	Strengths and Focus Areas of Various Models of the Development Process	27
5.1	Organisational Sizes According to Jones et al. [74]	42
5.2	Levels of Organisational Maturity [74]	45
5.3	Project Sizes According to Jones et al. [74]	46
5.4	Project Phases for Various Project Types [125]	48
6.1	Framework Values for Case Study Project	63
6.2	Framework Values for Case Study Project	74
7.1	The Factors that Influenced the Development Project Discussed in Case Study B, Section 6.2	88
7.2	Numerical Values Assigned to the Various Contextual Variables	89
7.3	The Steps of the Roadmap to Generate a Project Methodology	93
7.4	Requirements for Development Support Facilities [100] and the Level that EDEN™ Meets these Requirements	94
7.5	Requirements for Development Support Facilities [20] and the Level that EDEN™ Meets these Requirements	95
A.1	Selection of Design Method by Project Phase according to Green [58]	104
A.2	Selection of Design Method by Project Phase according to Maffin [86]	105
A.3	Selection of Design Methods According to Matrin [90]	106
A.4	Selection of Design Methods According to Stetter [116]	107
D.1	Factors Influencing the Engineering Design Process by Wallace and Hales [127]	138
D.2	Defining Factors for Key Dimensions [87]	144

Abbreviations

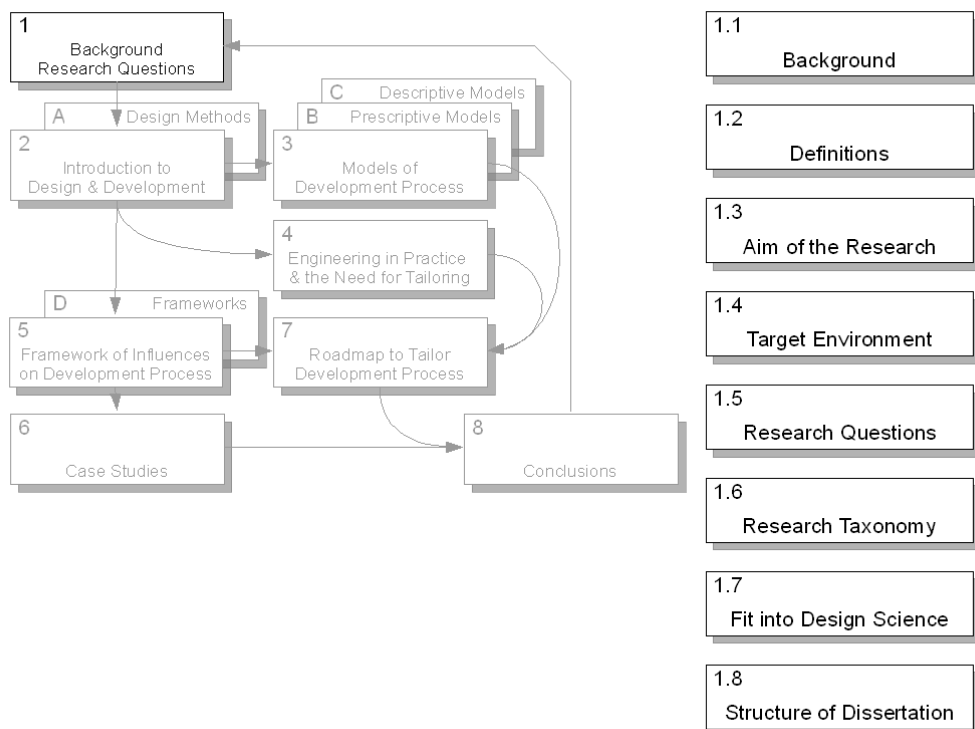
3D	Three (3) Dimensional
BOM	Bill Of Material
CEO	Chief Executive Officer
CAD	Computer Aided Design
COTS	Commercial Off-The-Shelf
DFX	Design For X
FEM	Finite Element Method
FMECA	Failure Mode, Effects & Critically Analysis
FMEA	Failure Mode & Effects Analysis
FIDR	Future Issues In Design Research
IPC	Illustrated Parts Catalogue
IT	Information Technology
MRI	Master Record Index
NCACC	The National Conventional Arms Control Committee
OMM	Operator's Maintenance Manual
PDM	Product Data Management
QFD	Quality Function Deployment
R&D	Research & Development
SANDF	South African National Defence Force
SEMP	Systems Engineering Management Plan
SME	Small and Medium Enterprises
TEMP	Test and Evaluation Master Plan
VDI	Verein Deutscher Ingenieure
WRM	Workshop Repair Manual
WBS	Work Breakdown Structure

Chapter 1

Introduction

Synopsis

This chapter introduces the reader to the research described in the rest of the dissertation by providing the background, describing the purpose, the specific research questions and how this work fits into the landscape of design research in general. For clarity's sake this chapter includes the author's definitions of terminology used in this dissertation. Section 1.8 concludes this chapter with a brief outline of the structure of the rest of the dissertation.



1.1 Background

During the author's career in a number of industries, various companies producing a wide variety of products, he noticed that the development processes were only similar on a very high level of abstraction. The detail differed from project to project.

Wanting to apply development process models, as proposed by design science, to industrial projects, the author realised very quickly that these models could only provide the high level structure for the project. They all came with small print: "adopt to suit the circumstances", but no indication was provided how one would approach this adaptation process.

The author also noticed that there was a correlation between structuring the project and its level of success. Some projects failed due to the lack of structure, while others failed due to micro-management or too much prescription. The most successful ones had the "right" level of structure - somewhere in the middle. How much is enough, and how much is too much? How is the level of structure measured?

Lastly, the author noticed that many of the proposals to support engineering development in practice were based on very elaborate IT infrastructure, both in terms of software as well as hardware. These solutions would only be suitable for large corporations with turnover large enough to be able to afford such overhead. How can one assist SME's? Can the fundamentals be proposed in such a way that they can be implemented in a variety of environments, each with complexity to suit the organisation that it serves?

These are the main motivators for the author to have embarked on the journey of exploration and discovery, which ended with the summary of his thoughts and proposals in this dissertation.

1.2 Definitions

This section is not intended to join the discussion of how specific words in the field of design research should be defined. The development of a dictionary and a taxonomy for design research is a field of research in its own right, and lies outside of the scope of this research. Therefore the purpose of this section is only to define the author's understanding of the vocabulary used in this dissertation.

design (object)	The result of a design process, [11] in the form of a complete, verified description of the artefact to be designed. For the engineering design process the artefact will usually be described by a set of specifications in the form of engineering drawings, material specifications and manufacturing instructions.
design (process)	Design is the part of the development process where user requirements are translated into a hierarchy of components, assemblies, sub-systems and systems as part of the product development process described in detail in Section 2.1.
design method	A step-by-step procedure or recipe describing a way of completing a task within the design process. This includes guidelines, rules of thumb as well as procedures such as quality function deployment (QFD), design for X (DFX), functional analysis and brainstorming.
design science	The branch of science that is concerned with the understanding engineering design and development.
engineering design	Design with particular emphasis on the technical aspects of a product. Includes activities of analysis as well as synthesis [129].
framework	A supporting or underlying structure [2] or a set of assumptions, concepts, values and practices that constitutes a way of viewing reality.
function (of a product)	That which an element (system, part, component, module, feature) of a product actively or passively does in order to contribute to a certain purpose [129].
function (in a process)	That which a resource (individual, team or department) of an organisation contributes to a certain process.

industrial design	The ideation, specification, and development of functions, properties and concepts of industrially manufactured products and systems, mainly regarding aspects of user product interaction, aesthetics and identity, considering a totality of ergonomic, usability, technical, economic, and social factors [129].
life-cycle	The transformation process from need through production to disposal of the product as part of a business (see Figure 2.5). This is what Hein [64] calls a " <i>superior product development model</i> "
methodology	i. The branch of logic dealing with the methods of accurate thinking [21]. ii. " <i>is used for knowledge about practical steps and rules for the development and design of technical systems, based on the findings of design science and of practical experience in various applications.</i> " [17] or iii. " <i>is a collection of procedures, tools and techniques for designers to use when designing.</i> " [49]
model of the development process	A description of the design process, often including flow-charts. This is not to be confused by models of the design (object) used during the design process.
model of the development process - prescriptive	A structured, systematic description of the design process, divided into sub-processes each generating a specific result. Also referred to as a step-orientated model of the development process [94]. See Section 3.1 for more detail.
model of the development process - descriptive	A description of the design process with emphasis on the cognitive processes within the development process, focussing on solving the main problem first, and addressing all peripheral problems accordingly, and solutions are generated early in the development process based on presuppositions. See Section 3.3 for more detail.
product	The physical artefact manufactured according to the design. See "design (object)" above.
process	A sequence of coordinated tasks aimed at achieving a specific goal.
product development (process)	The process to generate a complete, verified description of the artefact to be designed from a design brief or user requirement statement. Verification of the design, usually by means of the construction and evaluation of a prototype, is considered part of the design process. Models of this process are discussed in Chapter 3. The process includes engineering design and industrial design shown as part of the product life-cycle in Figure 2.5. Hein [64] refers to this process as the " <i>product synthesis model</i> ".
product innovation	" <i>Is the introduction of change from a strategic perspective</i> ", including " <i>new products, new processes, new distribution networks, new marketing networks</i> " [91].
roadmap	" <i>a layout of descriptive paths that multidisciplinary teams can use as a guiding framework for collaboration efforts towards a common goal.</i> "[39]
structure	With respect to a process, the term "structure" refers to a prescribed rigid procedure, which is to be followed in order to complete the process.

Although this dissertation contributes to the methodology (in terms of definition i. above) related to engineering design, the word 'methodology' is used in this dissertation with the second meaning in mind. To the author's mind it incorporates more than merely a collection of tools and techniques. It includes an approach or philosophy towards the particular project being executed.

1.3 Aim of the Research Project

The author of this dissertation has come to realise during his involvement with product development for nearly twenty years that no two projects are executed in an identical manner, even if they are executed in the same environment by the same development team. He agrees with Dumas and Whitfield [45] that "*industry, in its relationship with the design function, is not a uniform domain which can be addressed simply and directly; rather, it is segmented with each type exhibiting a unique culture/practice profile. As such, it argues against a*

simplistic approach to the management of design within industry and suggests a greater need for tailor-made solutions". In his article Cooper [30] confirms empirically that no "typical process model" for the development and commercialisation of a new product exists. Even very prescriptive process models such as the one proposed by the Institute for German Engineers [125] suggest that the "standard" process should be adapted to suit the given context.

Unfortunately not much guidance is provided regarding the required type or amount of adaptation for various project situations. The author agrees with Fulcher and Hills [55] that design research should enable design to be practised "*in a context of real people, real organisations and real markets, which are characterised by complexity, time pressure, limited resources and changing circumstances*".

The aim of this research project was to provide the engineer responsible for the execution of a design project a tool to configure his own specific development methodology for his specific circumstances. The word "configure" indicates that the process of establishing the methodology is one of adjusting a template to given circumstances.

1.4 Target Environment

A large amount of research is funded by and aimed at the design processes and environment of large multi-national corporations [34], mostly automotive and electronics manufacturers [87]. SME's in general and smaller suppliers to these industries in particular are generally assumed to be miniature copies of the large corporations. Radcliff et al. [104] confirm that although SME's "*comprise the bulk of the manufacturing industry yet little account is taken of such firms in the literature on design theory and methodology*".

Since the author of this dissertation has worked in SME's, the research is aimed at providing affordable, effective support for product development in small and medium sized enterprises.

1.5 Research Questions

The main question that this research project wanted to answer is: "Given a specific project, with specific resources within a given environment, how does one select and construct a development methodology from a repository of templates?"

Secondary research questions are:

- 1 Can benefit be obtained by combining design methods to form a project methodology?
- 2 Can methods be established to define a methodology optimised for a specific project?
- 3 How do I specify, choose and construct a development methodology?
- 4 How do I measure the success or appropriateness of my choices?
- 5 How do I comprehensively quantify my needs with respect to a development methodology?
- 6 Can a methodology be represented in a roadmap?
- 7 Can a roadmap, implemented in a design environment, facilitate the engineering design process?
- 8 What attributes must a roadmap have so that it can represent a development methodology?
- 9 What attributes must a design environment have so that a roadmap can be applied to a development project?

1.6 Research Taxonomy

An analysis of the most relevant articles used as references in this research, as well as the text of this dissertation, revealed the following framework of concepts. It is presented here to describe the scope of the research by providing the reader with an overview of the topics that are addressed in this dissertation.

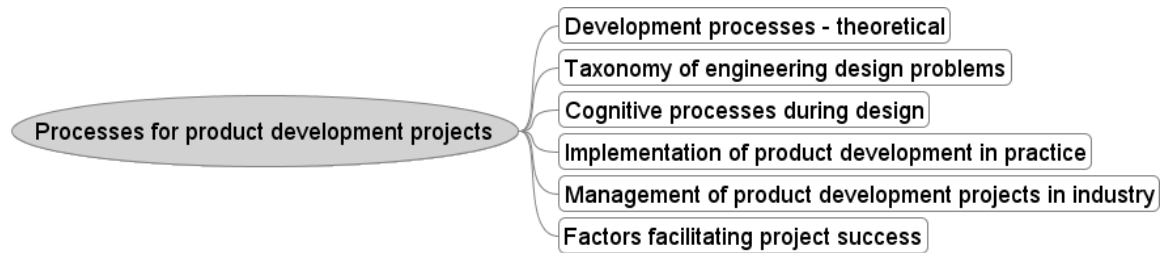


Figure 1.1: Framework of Concepts Addressed in this Research

1.7 How and Where Does this Research Fit into Design Science?

A number of authors have proposed ways of mapping and describing the landscape of design science. This section will describe some of these and indicate where the work of this research project fits into the big picture.

1.7.1 Research Categories According to Cross

Cross [31] categorises the research with respect to engineering design and development under the following five headings:

- 1 The Development of Design Methods: origination and application of systematic methods,
- 2 The Management of Design Process: models and strategies for executing design projects,
- 3 The Structure of Design Problems: theoretical analysis of the nature of design problems,
- 4 The Nature of Design Activity: empirical observations of design practice, and
- 5 The Philosophy of Design Method: philosophical analysis and reflection on design activity.

The idea for the research described in this dissertation was born out of the empirical observation of the author during his practice of design and development engineering that no two projects followed identical processes, i.e. item 4 of the list above. Having investigated the realm regarding the management of the development process (Chapter 3), i.e. item 2, and the author argued that the development process is, can and should be adapted to the boundary conditions surrounding the development project (see Section 4.2). In Chapter 5 the author provided a framework of variables, describing these boundary conditions, which, in his opinion, influence the development process the most. By applying the proposed framework of variables to two case studies in Chapter 6 the author returned to item 4 of the list above. Chapter 7 provides a method of assisting the practising engineer with the tailoring of the development process for his specific project based on the framework of variables, and thereby making a contribution to the realm of item 2.

1.7.2 The Taxonomical Framework of Horváth and Vergeest

Horváth and Vergeest [66] have proposed a taxonomical framework to categorise work being performed within the domain of design science in order to relate the work of the various authors to each other. The authors have defined three contextual categories, as shown in Figure 1.2:

- 1 Source categories provide the "*fundamental mental capacity for engineering design*", while
- 2 Sink categories are concerned with "*the ultimate utilisation of the aggregated knowledge*", and
- 3 Pipeline categories are those categories which link the former to the latter.

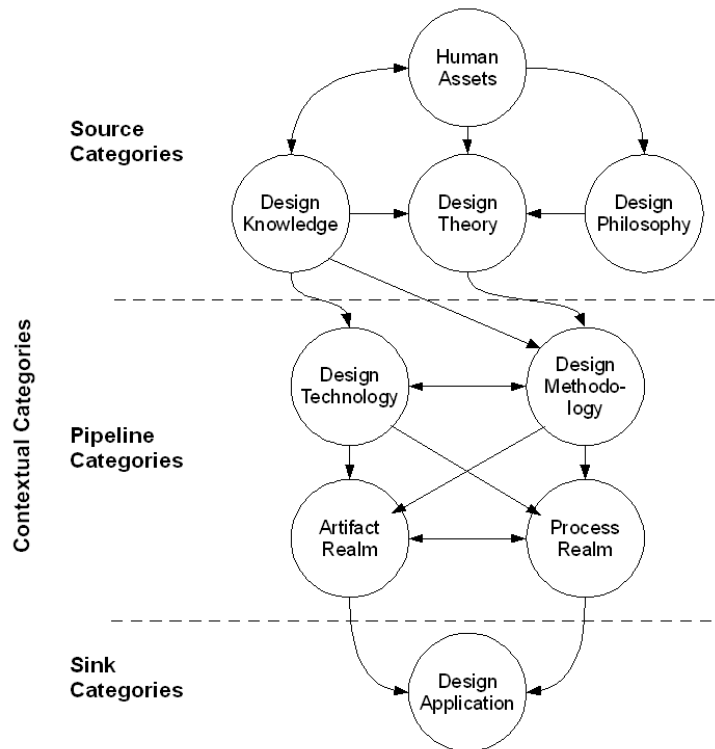


Figure 1.2: Contextual Categories for Design Science [66]

Within each contextual category, they have defined research domains for finer detail. The arrows between the contextual categories indicate the dependence between them.

The following is a quotation from the paper by Horváth and Vergeest [66] describing each of the contextual categories.

1 Human assets

- *originators of universal design knowledge (design philosophers, design scientists and/or theoreticians)*
- *design problem solvers (design methodologists, designers, design system developers)*
- *targeted [beneficiaries] (users, consumers, students)*

2 Design knowledge

- *knowledge gained from natural, social and technical sciences, but also from practice, human involvement*

3 Design philosophy

- *the existence and manifestation of design*
- *the role and position of design in society*
- *the historical evolution of design*
- *the ways of design thinking*

4 Design theory

- *reasoning about the arrangement of design knowledge in a systematic way*
- *interpretation of design semantics*
- *explanation, generalisation and/or abstraction of observed design processes*
- *devising theorems, rules and procedure as a set of instructions for solving design problems*

- *working out approaches to exteriorise design for execution in synthetic environments*

5 Design methodology

- *understanding of design decision making*
- *methodological systemisation of design processes and/or activities*
- *design modelling and representations*
- *design analysis and simulation*

6 Design technology

- *application of specific human and artificial resources to support analysis, design problem solving, synthesis, representation and documentation*

7 Realm of artefacts

- *in their physical manifestation, the ultimate destination of design*

8 Realm of processes

- *understanding the synergy of artefacts and processes (natural, imaginary and artificial)*
- *design-related abstract and concrete processes*

9 Design application

- *putting design into operation*
- *industrial aspects of application and implementation*
- *setting and regulating its performance characteristics*

This dissertation provides arguments for the appropriate configuration of the development process to its environment, and by investigating how the environment surrounding a project influences the development process, it indicates how to adapt the development process to suit given circumstances. With regard to the framework above the focus of this dissertation is clearly in the "realm of processes" by discussing the theory and thought processes required to structure the development process. By providing an indication of how to implement this structuring process in practice, this research also makes a contribution to the category of "design application". Of course all of this can only be done by building on the existing knowledge in the categories of "design theory" and "design methodology".

1.7.3 The Research Areas of Design Science by Hubka and Eder

Hubka and Eder [68] have devised a means of depicting the position of work in the realm of design science relative to other work in the realm by plotting its position on a Cartesian co-ordinate system. The technical systems are plotted on the negative x-axis, while the positive side of the x-axis is used for plotting work on design process. Prescriptive theories are plotted on the positive y-axis, while the negative side is used for the descriptive statements. The diagram is shown in Figure 1.3.

Figure 1.3 also shows where the research described in this document would be plotted. On the one hand the research investigates and makes a contribution to knowledge of design methodology, the "know-how", by proposing a new way of looking at and adapting the development process to the specific context of the design project. This part is depicted by point 1 in the figure. On the other hand this research investigates and makes statements regarding the design process itself, and how to adapt the process according to the context and environment that the design process has to operate in. This part is depicted by point 2 in the figure.

Both points describing this research lie to right of the origin, since the research investigates and makes a contribution to the design process rather than technical systems.

In the same book Hubka and Eder [68] describe the technical process as a process in which an operand (material, energy or information) is transformed from an input state to a output state, as shown in Figure 1.4. The process is influenced by humans, the technical system under consideration and the environment. This research asks the question how the technical process should be adapted to suit the environment, the humans who interact with the technical process and the technical system to be developed.

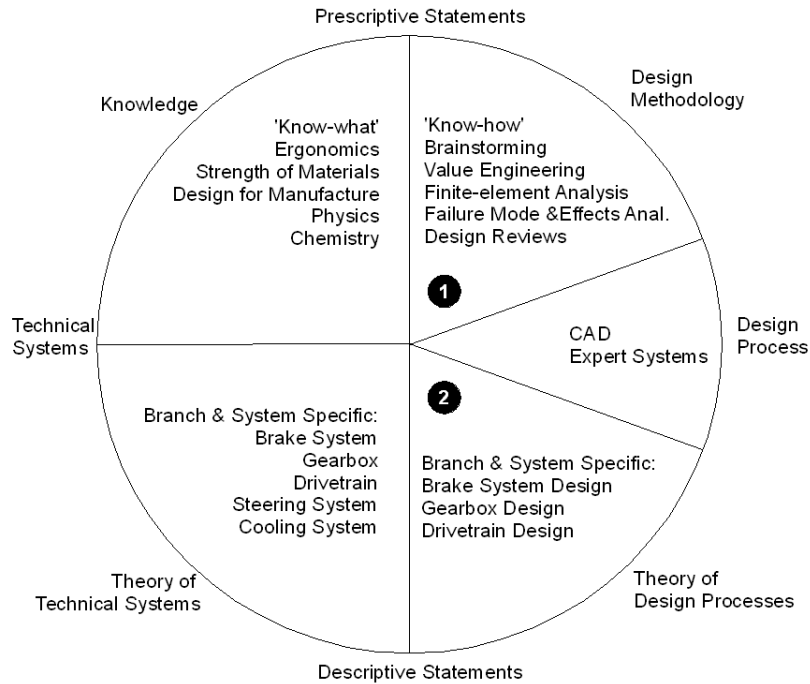


Figure 1.3: Research Areas in Design Science According to Hubka and Eder [68]

1.8 Structure of this Dissertation

Figure 1.5 provides a graphical overview of the structure of this dissertation. This figure is repeated at the beginning of each chapter to remind the reader of the position of the current chapter within the broader context of the document. To emphasise this the block representing the current chapter is highlighted by fading the rest of the structure. For each chapter the breakdown of the particular chapter is provided to help orientate the reader.

Having sketched the background to the research described in this dissertation, as well as the scope, purpose and intent of the work in Chapter 1, the dissertation introduces the reader to engineering design and development in Chapter 2 by briefly describing the general characteristics of the design and development process and how it fits into the life-cycle of a product. By discussing various representations of the product life-cycle, the last part of Chapter 2 expands a little further on this concept, before focussing on the target of this research, the design and development phase in the next chapter.

Numerous prescriptive models of the development process are discussed in Section 3.1, highlighting their similarities and differences. Given the level of structure of these models, specific methods can be allocated to the various phases of these prescriptive models of the development process. These methods and their selection is also discussed briefly. The discussion of the prescriptive models is concluded in Section 3.1.3, while Section 3.2 summarises the author's view of what constitutes the principle rules for the successful

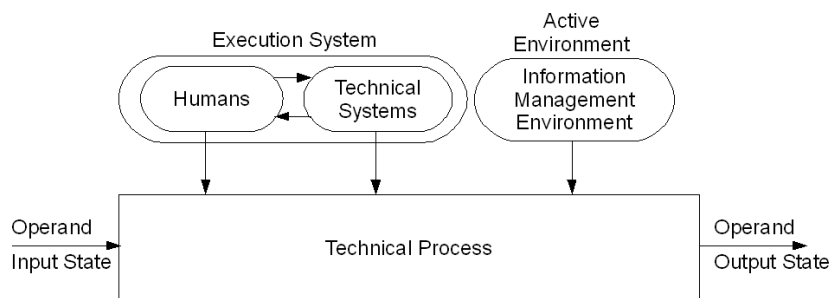


Figure 1.4: The Technical Process According to Hubka and Eder [68]

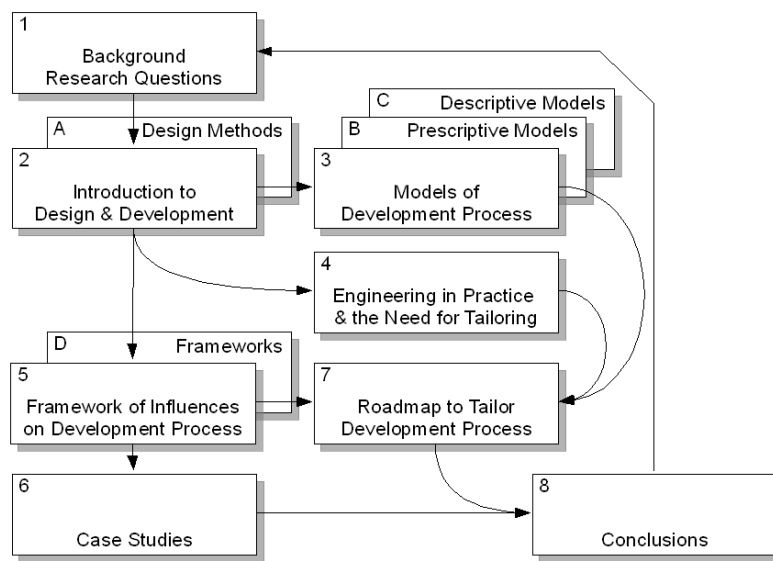


Figure 1.5: The Structure of this Dissertation

execution of a development project that these prescriptive models of the development process attempt to enforce.

The discussion regarding prescriptive models of the development process in Section 3.1 is contrasted by a similar discussion on descriptive models of the development process in Section 3.3. Rather than describing the time-line of a project as it progresses through its various phases, these models focus on the cognitive process that each designer employs during the execution of design tasks [33]. Similarities and differences among the various descriptive models are discussed in Section 3.3.

Given the large number of models for the development process and that not one has emerged as a globally accepted standard, Chapter 4.2 argues for the need to tailor or configure the development process according to the circumstances of the development project, and proposes that the concept of a development methodology, formulated as a roadmap, can facilitate this tailoring process. A methodology in this context is more than just a project plan or a list of design activities to be executed. It addresses the intent and purpose of a project, describes the approach or philosophy to be adopted during the execution of the project, and defines the value system which will determine the degree of success achieved at the end of the project. In this context Section 4.3 discusses the difference between the methodology of a development process, the models of the development process, discussed in Chapter 3, methods utilised during the development process and frameworks, such as the one presented in the next chapter.

With this in mind Chapter 5 starts to investigate the environmental factors that influence the development process, by reviewing existing work on this subject. Since none of these fit the intent and purpose of this research exactly, the author proceeds to generate his own framework of variables that influence the development process, based on the Maffin's [86] contextual framework. These variables and their interrelationships are described in detail in Section 5.2.

In order to illustrate relevance of this framework of variables, Chapter 6 presents two case studies executed within the South African military vehicle manufacturing industry. In both cases the background and environment of the project is described, contextual variables are assigned values, and the project's outcome is discussed with respect to these variables.

Having described the factors that influence the development process in Chapter 5, and their relevance to practical engineering development projects in Chapter 6, Chapter 7 refers to the concept of utilising a formal project methodology to capture the adapted or tailored development process, which was introduced in

Section 4.3. The first section discusses the differences between process orientated and content orientated work-flow, to provide the background to the following sections. Next the reader is introduced to the concept of a roadmap, which the author suggests as a tool to guide the practising design engineer through the process of adapting the development process for his particular development project. The author's implementation of such a roadmap is presented in the last section of this chapter.

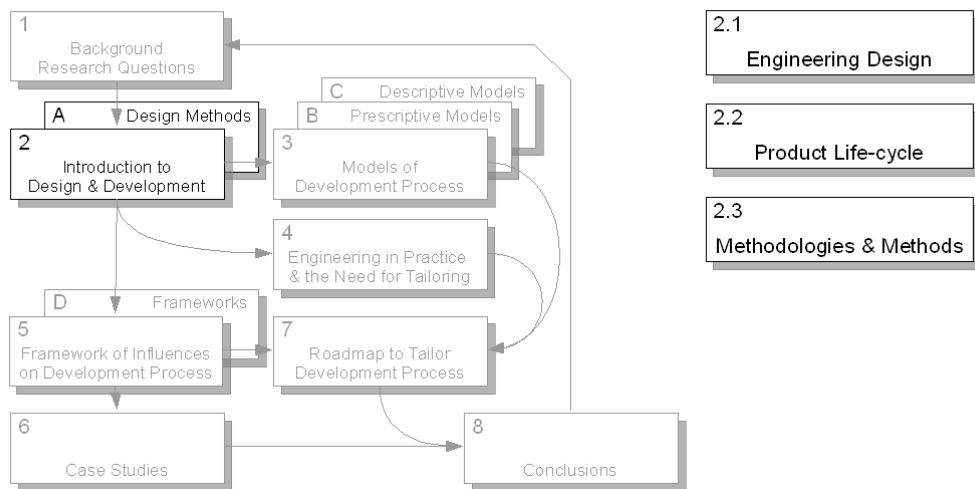
The last chapter of this dissertation answers the research questions posed in Section 1.5 by referring back to the information presented in the previous chapters. The limitations of this research, as well as its application as discussed briefly, before indicating possible research work for the future.

Chapter 2

Engineering Design and Development

Synopsis

Having introduced the reader to this research, including its scope and purpose in Chapter 1, the following chapter describes the characteristic attributes of engineering design and development in Section 2.1. The notion that the development process is embedded in a larger process, namely the product life-cycle, is introduced in Section 2.2. The relationship among the concepts "product life-cycle", "project methodology" and "design methods" are discussed in Section 2.3. Once this background to engineering design and development is provided Chapter 3 can proceed to describe the various models that have been proposed for this process.



2.1 What is Design in General and Engineering Design in Particular?

The need and desire to be creative is an integral part of human nature. Certainly the practice of creating or making artefacts is as old as humanity, but the separation of the 'making' process from the planning process, i.e. designing process is a fairly recent phenomenon, as part and parcel of the industrial revolution. In order to increase the efficiency of the production process, the industrial factory required a description of how the product was to be produced, compared to the traditional craftsman, who knew, often on a gut-feel level, how he would produce his goods.

The word "design" is not just used for the creation of technical systems. For instance, the term "graphic design" is used to describe the process of developing visual art as it is used in the entertainment and advertisement industries. While it would be conceivable to apply the research described in this dissertation to the artistic branches of design, the work described in the following chapters was conducted specifically with the development of technical systems in mind.

Cross [32] points out that designing, in the technical sense, "*is the production of a final description of the artefact in a form that is understandable to those who will make the artefact.*"

Shigley [112] describes engineering design as "*the decision-making processes which mechanical engineers use in the formulation of plans for the physical realization of machines, devices and systems. These decision-making processes are applicable to the entire field of engineering design.*"

Evboumwan et al. defines design as "*the process of establishing requirements based on human needs, transforming them into performance specification and functions, which are then mapped and converted (subject to constraints) into design solutions (using creativity, scientific principles and technical knowledge) that can be economically manufactured and produced.*"

The development process moves from a state of uncertainty to a state which is deterministic and well defined, starting with a problem description that is often vague and incomplete, ending with a complete and verified description of the product to solve the initial problem. This is illustrated in Figure 2.1 by Andreasen [9]. As the knowledge increases during the development process, decisions are made which increases the certainty with regard to the new design. However, each decision made defines the new design a little bit more, reducing the freedom of making decisions in the future. This is shown in Figure 2.2. As the definition of the new product is defined in more and more detail, more and more of the product's cost is fixed as a consequence of the decisions taken during the development process. This is shown in the graph by Schulz [110] et al. in Figure 2.3.

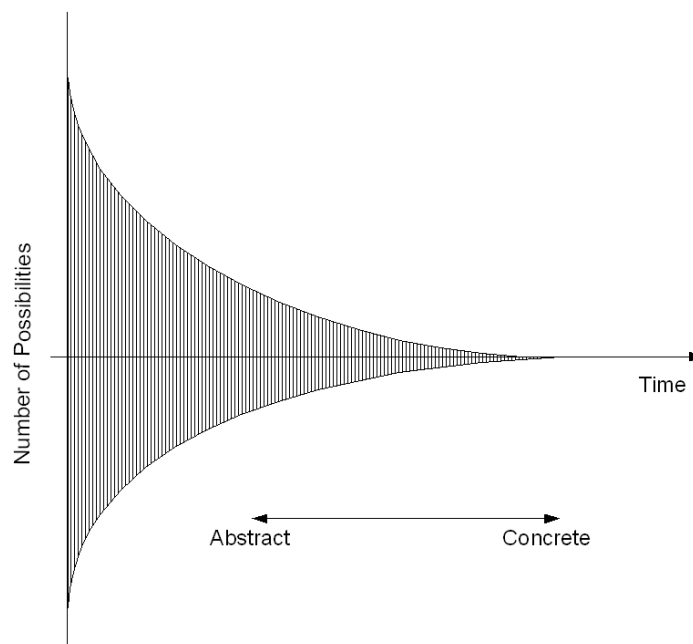


Figure 2.1: The Design Process According to Andreasen [9]

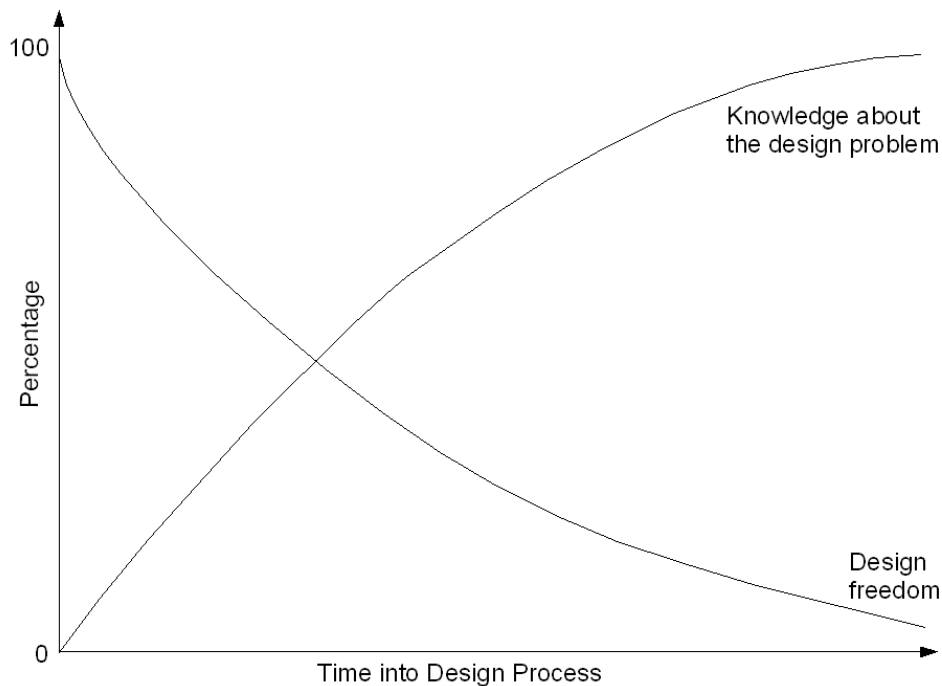


Figure 2.2: Design Freedom and Knowledge During the Development Process [123]

On his web-site "*Methodical Design*" Johannes [73] defines two basic philosophies to this process of design. The first is distinguished by the intuitive, imaginative visualisation of the solution to the design problem in its entirety, and the visual aspects and geometric modelling are given high priority. Once a potential solution has been visualised, it is evaluated in terms of the functional requirements, and adapted to make it fit.

The second basic design philosophy is distinguished by the supposition that the designer ought to derive the solution to the design problem by taking the user requirements into account in a methodical, systematic manner. The geometric form of the design is therefore derived from the user requirements, rather than the other way around.

Clearly design activities very seldom exhibit these basic philosophies in their pure form described above. In reality design is performed with a mixture of these two philosophies, where one or the other may dominate. What is clear however is that the process of design cannot be prescribed to be followed in a mechanistic manner. According to Andreasen [8] the design process is "*more a learning process than an algorithm*", illustrating that the designer learns about the problem he is solving while he generating the solution for the problem. The process is therefore iterative in nature, i.e. while executing the design process, the engineer will gain insight into the problem matter, causing him to re-consider one or more of his previous decisions. This feature, combined with the differences in scope and purpose of each development, makes it impossible to define the process within the boundaries of a rigid, pre-defined procedure.

Design activities can be divided into a number of different categories:

Evbuomwan et al. [49] distinguish between

- routine designs - which are "*derived from a common prototype with the same set of variables or features and the structure does not change*"
- redesigns - involving the modification of "*an existing design to satisfy new requirements*", and
- non-routine - original or new designs.

The design guideline of the society of German engineers (VDI) [125] and Ullman [123] also identify original design as one of the categories. Routine designs, as defined above, are subdivided into parametric, configuration and selection design. Modification of any of these would constitute a redesign.

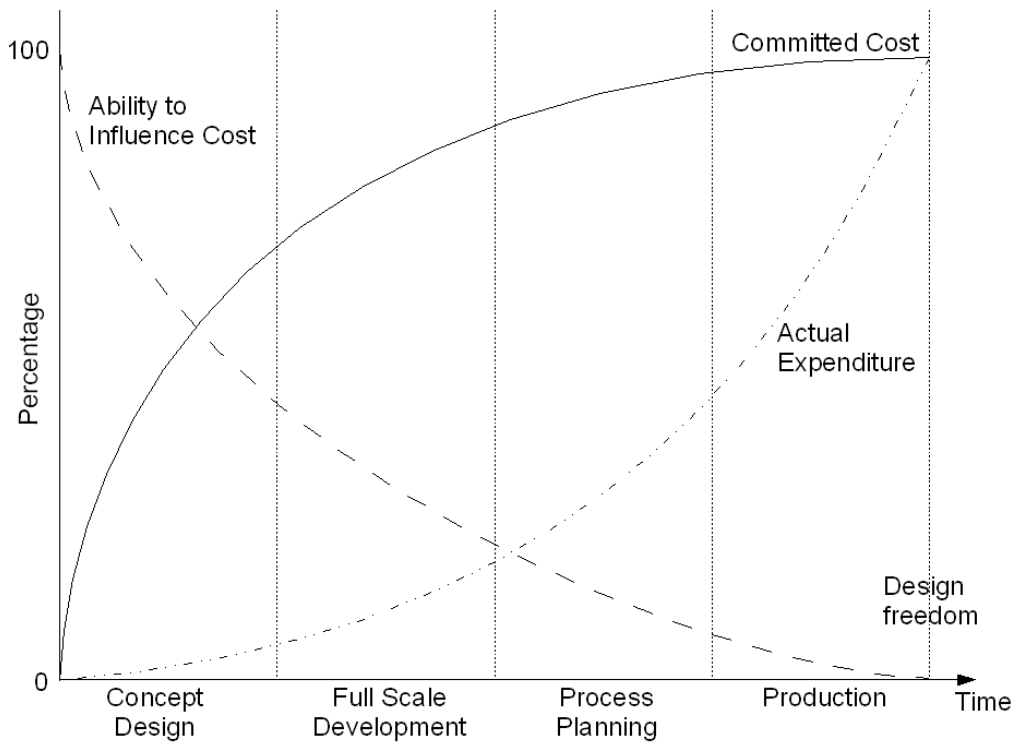


Figure 2.3: Product Costs as a Function of Time During the Development Process According to Schulz [110]

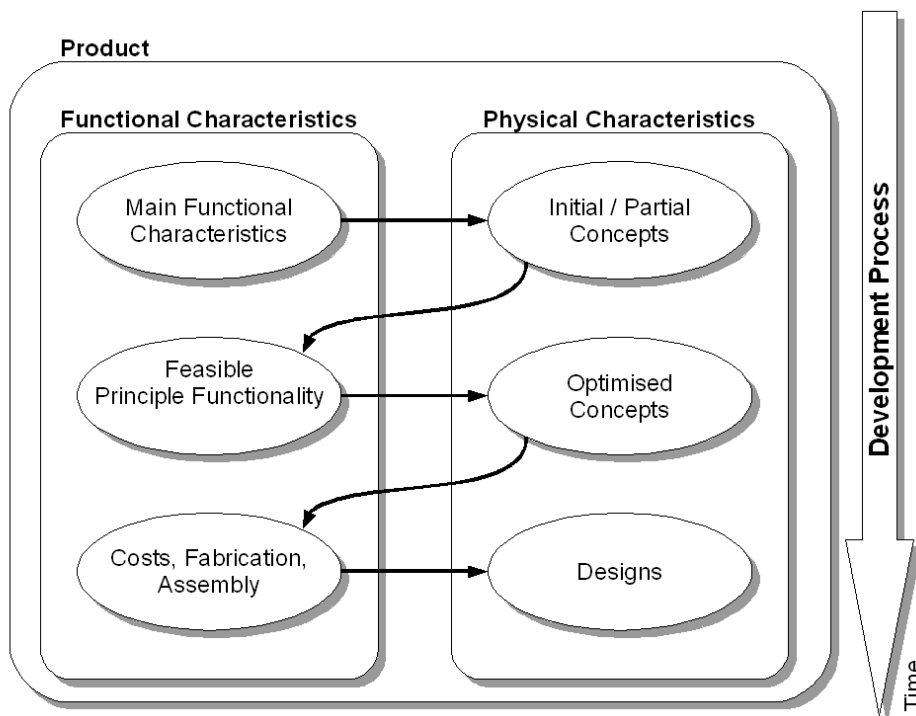


Figure 2.4: The Inter-relationship of Synthesis and Analysis During the Development Process as Depicted by Bichlmaier [18]

Furthermore the process of design generally exhibits the following characteristics:

1 The process involves the following:

- | | | |
|--------------------------|---|---|
| Analytical activity, | - | where understanding of the design problem and its associated solution is generated by the analysis of data. |
| Synthesis, | - | where possible solutions are generated. This is done at various levels of detail and abstraction. |
| Induction or evaluation, | - | where data (including possible solutions) is assessed for the basis of decision making. |
| Decision making, | - | where the next course of action is determined. |

The inter-relationship between synthesis and analysis is illustrated in Figure 2.4 by Bichlmaier [18].

2 As summarised by Juster [76], as well as Evbuomwan [49], the development process is initially divergent during the early phases where the solution space is searched for possible solutions, while the last phases of the process are convergent, once decisions have been made regarding the best solutions to implement:

- | | | |
|----------------|---|---|
| Divergence | - | In this stage emphasis is on extending the design boundary. The design is unstable, ill-defined and no evaluation is performed. |
| Transformation | - | In this stage, the problem becomes bounded, judgements are made, the problem is decomposed, and sub-goals are modified. |
| Convergence | - | In this stage, there is progressive reduction of secondary uncertainties until a single design emerges. |

3 During the development process documentation is generated at various levels for different reasons. Some of the possibilities are as follows:

3.1 The product is documented:

- Since the people who develop the product are often not the same people who manufacture the product, documentation needs to be generated which informs the manufacturer what to make and how to make it.
- Similarly the people who develop the product are not the people who support and maintain the product, therefore information on how to maintain the product needs to be documented.
- The users of the product are usually a third group of people, which need to be informed on how to use the product.

3.2 The development process is documented:

- To make information generated by one part of the development team available to the rest of the team.
- To make information generated for one project available for the next project.

Since the industrial revolution the process of producing products is not just one of planning and making. The process considers all aspects of the product from conception to disposal, and is appropriately called the product life-cycle. A typical product life-cycle is shown in Figure 2.5, as adapted from Krause [80].

In the research phase new technologies (in terms of products as well as the production process) and market opportunities are investigated. The strategy phase will usually use this information to generate a long term plan for a new product, including positioning the product among its competitors in the market place, establishing cost and price expectations and generating a broad description of the new product in terms of features, functionality and aesthetics. In the design phase this information is crystallised into a product specification, which describes the product in more quantitative terms. Next a detailed description of the new product is developed using engineering drawings, material and process descriptions. Having described the new product in detail, the form and fit of all the components that make up the product are verified by the manufacture of a prototype. Once complete, the prototype is used to verify that the functional requirements of the product have been met. The description of the product is updated to include any changes required to improve either the manufacturability of the product or its functional performance. Industrialisation is the phase where the production process and facilities are finalised, to be used to produce the product during the production phase. Distribution and usage phases concentrate on making the product available to customers and supporting it during the period that the customer uses the product. The last phase considers aspects related to the disposal of the product.

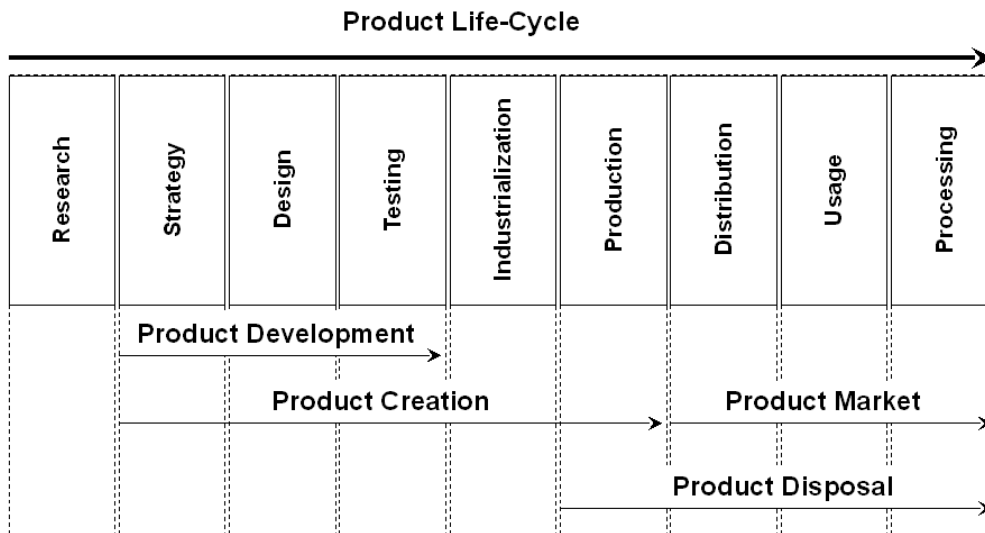


Figure 2.5: The Product Life-Cycle Adopted from Krause [80]

2.2 Product Life-Cycles And Product Development

In Section 1.2 the reader was introduced to the concept of a product life-cycle. This process describes the life of a product from "cradle to grave", i.e. from the identification of a need to the disposal of the product. A graphical representation of this process is shown in Figure 2.5, as adapted from Krause [80]. There is broad consensus in the available literature regarding the basic structure of the product life-cycle.

The process is generally divided into a number of phases. This dissertation concentrates on the phases labelled "strategy", "design" and "testing", collectively called "product development". This is where the development of a product is planned, executed and results verified. This process can be modelled in various ways (discussed in detail in the following sections), but generally it is accepted that the process proceeds from a vague and often brief description of the problem to the detailed description of the solution to the problem via a number of sub-phases. These models of the design process are used as frameworks, which provide the structure required to enable the process to proceed in a logical, methodical and ordered manner. The following paragraphs provide a brief description of various representations of product life-cycle.

Systems Engineering is a school of thought which provides a systematic and holistic approach to the development of complex products, it also includes the concept of life-cycles: a product life-cycle, as well as manufacturing and support life-cycles, shown in Figure 2.6 from Blanchard and Fabrycky [19]. The portion called "product development" in Krause's product life-cycle is identified in the product life-cycle of systems engineering as "conceptual preliminary design" and "detail design and development".

Ullman [123] depicts the product life-cycle as shown in Figure 2.7. Although it is visually structured differently than the ones from Krause[80] and Blanchard above, it essentially consists of the same phases and tasks.

Product life-cycles in general and models of the development process in particular provide the broader context into which the design activities are bedded. Each phase of the design process usually attempts to use the information gathered and developed in the previous phase to clarify and crystallise the description of the product being developed by increasing the level of detail information available.

In order to improve the quality and efficiency of the development process researchers in the field of design research have modelled the life-cycle of a product since the 1960's. As the research described in this dissertation is particularly concerned with the "product development" phase of the product's life-cycle, the section below describes the most common models of the development process. "*Design models are representations of philosophies or strategies proposed to show how design is and may be done*" [49].

Within the engineering fraternity models range from the simple models such as the one by French [52] to very complex models, for instance, the model for systems engineering [19]. The model described in the guideline number 2221 of the Society of German Engineers (VDI) [125], shown in Figure B.7, is a typical example of average complexity.

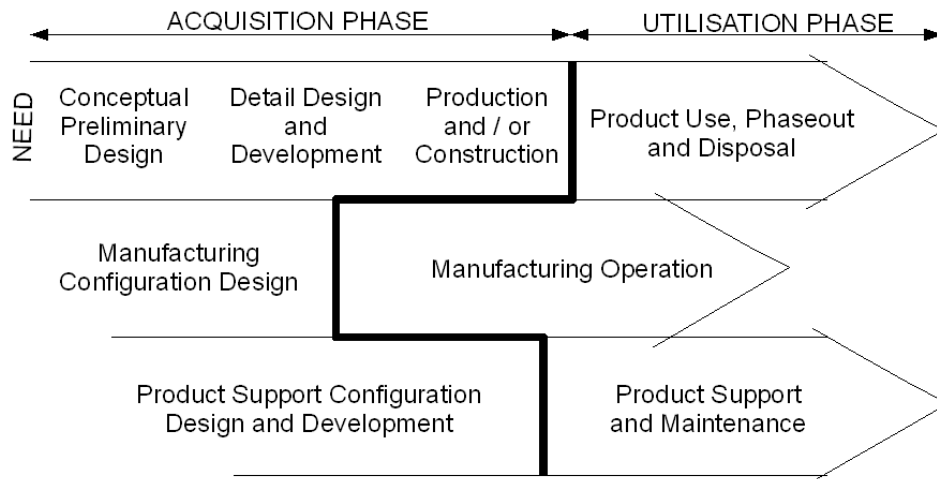


Figure 2.6: The Systems Engineering Product Life-cycle [19]

Although the models proposed by design science for the development process among engineers were initially similar to those in the architectural domain, architects gradually seem to have inclined towards descriptive models [33] such as the one by March [88].

- The prescriptive models (Section 3.1) of the process divide the development process into a set of distinct phases with the following characteristics [33]:
 - prescriptive sub-division of the development process into phases each generating specific intermediate results,
 - sub-division of the main problem into sub-problems, each sub-problem is solved and the main solution is generated by integrating all sub-solutions, and
 - each sub-solution is generated by selecting the most promising one of multiple alternatives, which are based on the analysis of the functional requirements.
- In comparison, the descriptive models (Section 3.3) concentrate on the cognitive and intuitive process required for product development, and are characterised as follows [33]:
 - psychological models describing the cognitive processes employed by designers during the development process [94],
 - main problem is solved by focusing initially on the central problem, and addressing all peripheral problems accordingly [47], and
 - solutions are generated early in the development process based on presuppositions.

This difference in approach towards modelling the development process within the design science fraternity, indicates that there are aspects of the development cycle which are not addressed by the rigid, prescriptive models proposed in the engineering domain.

2.3 Product Life-cycles, Methodologies, and Design Methods

The models of the development process discussed in the previous section, divide the process up into phases, which in turn can be subdivided into tasks and sub-tasks. Common tasks or tasks which are repeated often can be combined into a procedure or method, as defined in section 1.2. This includes guidelines, rules of thumb as well as procedures such as quality function deployment (QFD), design for X (DFX), functional analysis and brainstorming to name a few.

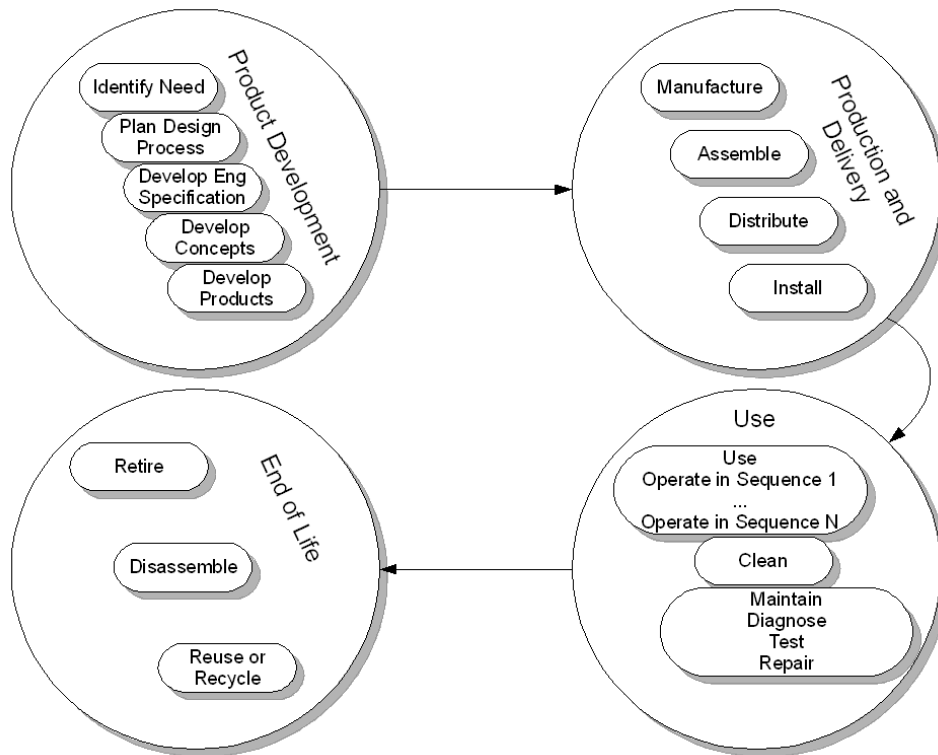


Figure 2.7: Ullman's Product Life-cycle [123]

In contrast to the model of the development process, design methods or tools are procedures to achieve very specific goals, as dictated by the development methodology. For example, assuming that the manufacturing cost is critical for the success of a particular new product, it would be reasonable for the development methodology to require the use of design for manufacturability and assembly (DFMA) methods to ensure an efficient manufacturing process. Another project may require a novel, innovative technical solution for its success, resulting in the use of brainstorming as a design method being used during the concept development. Although it is sometimes possible to prescribe a portion of the methods required for a project during the planning stages of the project, the heuristic nature of the development process, will usually reveal the need for methods during the execution of the project. The relationship among "product life-cycle", "project methodology" and "design methods" is discussed in more detail in Section 4.3 and is illustrated in Figure 4.4. Appendix A provides a fairly comprehensive list of methods arranged for selection:

- by project or development life-cycle phase, and
- by need.

There is a school of thought that maintains that methods and design tools cannot merely be selected and used. They have to be introduced into the development environment of a company in a structured and controlled manner. De Araujo [38] calls this the "*tools acquisition process*". This is especially true for large corporations where a large number of departments and design teams (often distributed across the globe) have to be coordinated and their efforts integrated. It is also true for the acquisition of complex tools within small and medium enterprises, i.e. purchasing and integrating a CAD or PDM system into an existing organisation is a daunting task.

The research described in this dissertation, however, focuses on the planning and execution of engineering development projects, and therefore considers the establishment of resources, such as CAD or PDM systems, an activity within the realm of enterprise management rather than the management or execution of a development project. For large complex methods (and tools) this research therefore assumes that they are either available for use or not. Smaller, easy to use methods, such as DFX and brainstorming, are considered to be simple enough that individuals and design teams can "acquire" them without a formal acquisition process.

It is important to point out that the information summarised in the tables of A.1 and A.2 are quoted from various sources. Each of these has its own definition for certain terminology. For instance, Green [58]

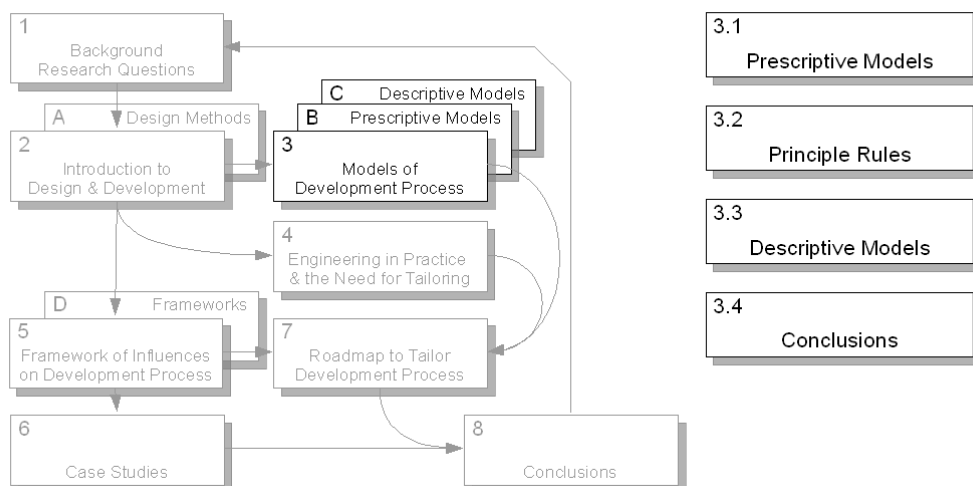
considers design reviews to be a method to be used during the embodiment and detail design phases of the development life-cycle, while in this research they are considered part of certain models of the development processes, e.g. Systems Engineering as shown in Figure [B.9](#).

Chapter 3

Theoretical Models of the Development Process

Synopsis

Having illustrated the landscape of engineering design and development in broad terms in Chapter 2, the following chapter paints a more detailed picture of the engineering development process: by discussing various prescriptive and descriptive models of this process in Sections 3.1 and 3.3 respectively. For both model types, the models are analysed and their differences and similarities discussed. Brief summaries of the most well-known prescriptive and descriptive models are provided in Appendices B and C respectively. In Section 3.2 the author expresses his view on the principles rules that the prescriptive models attempt to impose on the development process, through their degree of structure, in order to increase the probability of executing a development project successfully. Since Chapter 3 concentrated on the theoretical, academic descriptions of the development process, Chapter 4 contrasts this to the actual daily life in the practice of engineering design, identifying the need to adapt or tailor these theoretical models according to the context of the development project to be executed in practice.



3.1 Prescriptive Models

This section briefly discusses various prescriptive models of the development process, with emphasis on their differences and similarities. These models are called prescriptive because they "prescribe" a specific step-by-step process for the development of a product. Other synonymous terms include "systematic" [97][125], "structured", and "step-orientated" [94]. Emphasis is usually placed on finding the solutions to design problems in a structured, systematic manner, in order to be able to select the best solution from a number of alternatives. Brief descriptions of the most well-known prescriptive models of the development process are provided in Appendix B. While earlier models concentrated on the technical process only, later models addressed the need to consider the whole product life-cycle during the development process. The models range from the simplest description of the problem solving process by Cross (see Section B.1) to the most complex, which includes systems engineering (see section B.9) and the German engineering guideline VDI 2221 (see Section B.7). The following sections discuss the differences and similarities of these models, concluding with the principle development rules, that these models attempt to enforce with their structured approach.

3.1.1 Similarities of Prescriptive Models of the Development Process

All the models agree on the basic characteristics of the development process as discussed in Section 2.1: for instance, that the product development process is a process of generating a complete description of the product, starting with a vague description of a need or a design brief. The process therefore progresses from a vague, uncertain starting point with increasing certainty to a complete description of a solution to the original problem.

There is consensus that the prescriptive (step-orientated) models of the development process "*aim to improve the design process by allowing designers to design in a systematic framework, which is based on phases of pre-categorised design activities*" [94]. The following phases are typically part of the model:

- | | |
|--------------------------|---|
| Problem definition phase | - to define the problem statement, including requirements, limitations and constraints. This phase is sometimes included in the concept phase. |
| Concept phase | - to develop the principle solution to the problem at a high level of abstraction. The output of this phase is a description of the concept(s) best suited to solve the problem or need defined in the previous phase. The description will usually address the functional requirements, including the technical performance measures, as well as a general layout of the design. |
| Embodiment phase | - to develop the concept in greater detail. The output of this phase usually includes more detailed information, such as production methods, product cost, product performance and reliability with regard to sub-systems and major assemblies. |
| Detailing phase | - to develop the product documentation for communication with the production and maintenance staff, as well as the end-user. The output of this phase is the complete, verified description of the product, in terms of its production, as well as the logistics, maintenance and support. |

The more detailed models such as the VDI model or the systems engineering model prescribe the output to be generated by each phase. These phases constitute what Pugh [101] calls "structure of the design process" and provides "*a standard for judgement and management of all the different design processes*".

All the prescriptive models of the development process agree that the development process is based on the functional decomposition and analysis of the problem. The results are used to divide the design problem into a hierarchy of sub-problems. To generate these sub-solutions multiple alternative solutions are generated for each sub-problem, evaluated and the best one is chosen. Preconceptions are purposefully set aside (at least in the beginning of generating sub-solutions) in order not to limit the solution space and generate as many varied potential solutions as possible. The solution to the complete design problem is then obtained by re-assembling the sub-solutions. The hierarchy of problems and solutions is illustrated in Figure 3.1.

Most of the more detailed models agree that design reviews are an integral part of the design process. A design review is essentially a formalised communications method. It can be used internally in the design team as part of the peer review process to evaluate the status and quality of the work, as well as to establish a common philosophy and approach towards solving the problem at hand. It can be used to communicate requirements to a subcontractor or supplier in order to integrate this external work with the work being performed by the rest

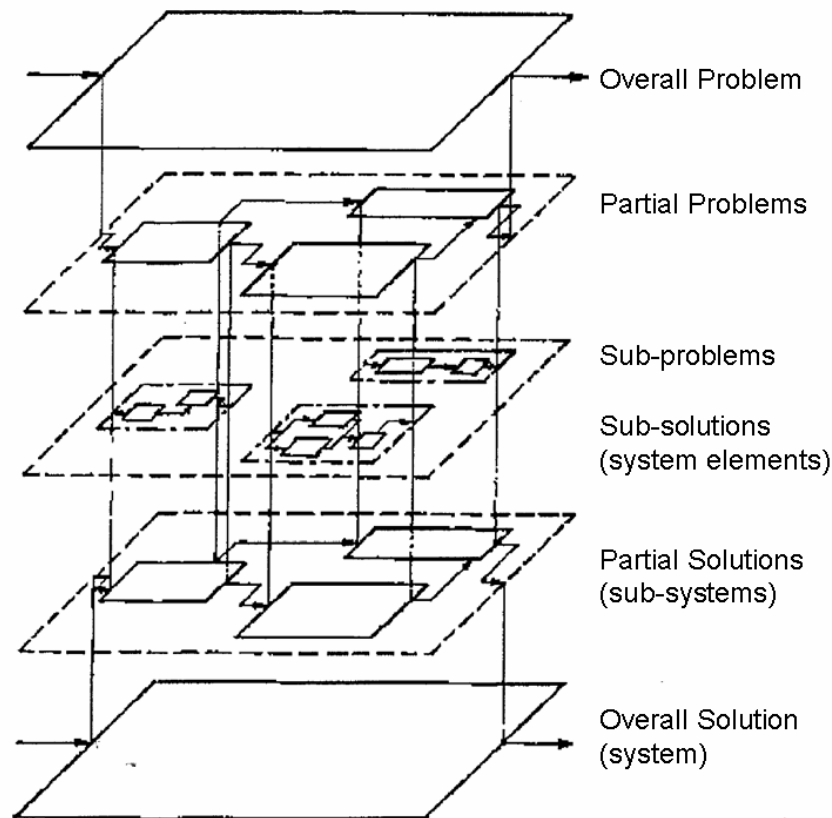


Figure 3.1: Solution / Problem Hierarchy [125]

of the design team. Another option is to use a design review to communicate with the customer to provide feedback with regards to the status of the project, obtain inputs on the customer's requirements and to verify that requirements have been met. The main aims of a design review can therefore be summarised as:

- It is a formalised check of the performance of a system, sub-system or critical assembly relative to its requirements.
- It establishes a common baseline for the project by creating an understanding of the *entire* project by design and support staff. This baseline includes the philosophy and approach to solving the design problem(s), the current status with regard to the quantity and quality of work, as well as resources, unforeseen problems.
- It provides a means of managing interfaces between sub-systems and assemblies developed by different sub-teams or suppliers and subcontractors.
- It is a formalised record of design decisions that influence the course of the project.
- Synergy is achieved by involving team members to evaluate work that they were not directly involved in.

Most of the prescriptive models of the development process state (the simpler models by implication, the more complex by explicit statements) that the process is not to be seen as rigid and should be adjusted to suit each individual project. Pugh [101], for instance, states emphatically that "the structure" of the prescriptive models, "*whilst imparting a discipline of systematic working, should nevertheless allow for variations to that systematic working, whilst retaining discipline and also imparting comprehensiveness. In other words, the system flexibility within a framework must not impose the rigidity and linearity suggested by many models*". Unfortunately, none of the models identify how and under what circumstances the models are to be adjusted.

3.1.2 Differences Among Prescriptive Development Process Models

The models of the development process described in the sections from B.1 to B.11 differ mainly in four aspects from each other:

- in the terminology used,
- in the level of detail and abstraction of the development process model,
- in the boundaries of the development phase within the product life-cycle, and
- in its strengths and focus areas.

Terminology

Since the models of the development process discussed in Appendix B were written over a period of approximately 40 years, in a number of different countries, by authors of varying background, preconceptions and intention, it is not surprising that the terminology used is not identical. Although it is conceivable that the various authors utilised different terminology in order to indicate differences in perspective or connotation, there are similarities in the broad concept being addressed. According to Gill [56] this is one of the reasons for the apparent divergence in models of the development process. The concepts in Table 3.1 can be considered roughly equivalent in their context, although different words were used to describe them. The table compares the terminology of the various models of the development process to terms used by French's model.

Table 3.1: Equivalent Terminology of the Various Models of the Development Process

French	Archer	Suireg	Pahl & Beitz	VDI 2221	Ullman	Systems Engineering
Analysis of problem	Programming, data collection, analysis	Analysis of need and formulation of objective	Clarify and define task	Clarify and define task	Specification definition	Included in conceptual design
Conceptual design	Synthesis	Synthesis of possible alternatives	Develop principle solution	Determine functions and their structures and solution principles	Conceptual design	Conceptual design
Embodiment of schemes	Development	Modelling	Develop & define construction structure	Realisable modules & layout of key modules & complete overall layout	Generate & evaluate product	Preliminary design
Detailing	Communication	Optimisation & implementation	Prepare product documentation	Product documentation	Document & communicate	Detail design & development

Detail and Abstraction of the Model

The second major difference among the various models of the engineering development process concerns the level of detail in which the model prescribes the development process. Cross, for instance, describes the process in very broad terms (Figure B.1), while the systems engineering model (Figure B.9) provides a lot more detail. This level of "structuredness" or "prescriptiveness" of the model is difficult to measure accurately. Therefore, figure 3.2 is a subjective indication of the author's opinion in this regard.

Models such as the ones by Cross (Section B.1) or Archer (Section B.3) provide a broad, basic framework for the technical development process. The exact internal detail of each of the activities are left to the project manager to specify.

On the next level, models such as one by Pahl and Beitz (Section B.7), are more prescriptive by defining the expected outcome of each phase of the development process. The approach to problem solving is also prescribed in detail (Figure 3.1).

The most structured, prescriptive models, such as the Systems Engineering model (Section B.9), specify the complete process in great detail, including the documentation to be generated, and the interfaces with internal and external role players such as marketing, the client, production, the end-user.

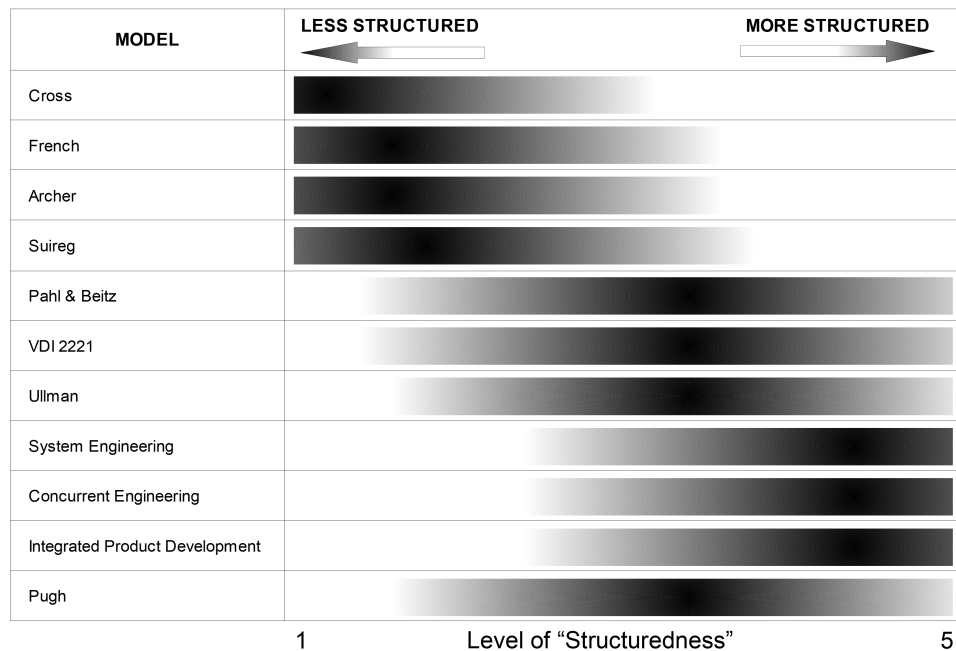


Figure 3.2: Degree of "Structuredness" for Various Prescriptive Models of the Development Process

Although some of the models provide a lot of detail, all authors agree that the product development process needs to be "tailored" to suit the specific project within its context. However, very little is said how this tailoring is to be done.

In Chapter 5.2 the author will discuss the relationship between project complexity and project size in detail, and will indicate how structure in the development process is especially necessary when the design authority can no longer "hold all the important aspects of a problem and all their interrelationships in mind at one and the same time" [108] due to the size and/or complexity of the project. The larger or more complex the project, the more structure is required to assist in the process.

Boundaries of the Development Phase

The third major difference relates to differences in boundaries of the model, i.e. what activities are considered part of "development" rather than "research" or "industrialisation" as illustrated in Figure 2.5. While all the authors agree in general terms on the sequence of the different phases of the development process, they differ on what is considered the start and ending of the "product development" phase of the product life-cycle.

Based on Cavallucci's matrix [26] the author has indicated in Figure 3.3 which phases of the product life-cycle are being addressed by the various prescriptive models of the development process. The description of the product life-cycle, and the definition which phases are combined under headings such as "product development" and "product disposal", for instance, was obtained from Krause [80] (see Section 2.2).

For instance, Ullman (see Section B.8) clearly includes project planning in his model of the development process, while Maffin (see Table A.2) and Green (see Table A.1) include market research and analysis into the first phase of the development process. That some authors have not mentioned these activities specifically, should not be taken as an indication that these authors believe such activities should specifically excluded from the development process. Since "project planning" and "market research", for example, are clearly useful and often necessary activities, one must assume that their omission from certain models of the development process is an indication of a difference in focus rather than an indication of lack of value.

Also what is considered to be the end of the development process differs. The author agrees with Krause [80] that the testing of the product (usually by means of a prototype) is part of the development process, since the evaluation of a physical prototype verifies the design on two levels:

	PRODUCT LIFECYCLE									
	Research	Strategy	Design	Testing	Industrialisation	Production	Distribution	Usage	Processing	
Product Development		■	■	■						
Product Creation		■	■	■	■					
Product Marketing						■	■	■	■	■
Product Disposal					■	■	■	■	■	■
MODEL										
Cross		■	■	■						
French		■	■	■						
Archer		■	■	■						
Suireg		■	■	■						
Pahl & Beitz		■	■	■						
VDI 2221		■	■	■						
Ullman		■	■	■	■	■	■	■	■	■
System Engineering	■	■	■	■	■	■	■	■	■	■
Concurrent Engineering		■	■	■	■	■	■	■		
Integrated Product Development		■	■	■	■	■	■	■		
Pugh		■	■	■	■	■	■	■		

Figure 3.3: Matrix Showing Which Phases of the Product Life-cycle are Addressed by Various Prescriptive Models of the Development Process

- The description of the design in the form of engineering drawings as well as material and manufacturing specifications is verified by the production and assembly of physical components.
- The functionality specified in the design brief or statement of requirements is evaluated and verified by means of tests on the physical prototype.

Many authors, however, end the development process after the detail design has been documented by generating engineering drawings which capture manufacturing and assembly instructions.

Strengths and Focus Areas

Each of the models of the development process described in Section B was originally developed by its author(s) within a specific context: each author had his particular perspective of the development process (how it is executed and how it can be improved) and developed a model to express this perspective. It stands to reason therefore that each model of the development process would have characteristics, strengths and focus areas, which might make it more suitable for a particular situation than another. Table 3.2 summarises the most obvious characteristics and focus areas for each of the models of the development process discussed in this dissertation.

3.1.3 Conclusions

Setting aside differences in terminology, scope and level of aggregation the author agrees with researchers such as Gill [56], as well as Cross and Roozenburg [33], that "*there is much greater consensus than is apparent from the literature*" with respect to prescriptive models of the development process. The structuring of the development into prescribed phases, each of which generate a specific output, was found to be particularly useful as a tool to manage the development process [94], [1].

Model of the Development Process	Strengths / Characteristics
Cross	1. Very basic, general description of the problem solving process.
French	1. Basic description of the development process. 2. Based on mechanical engineering design. 3. Focus on the design phase of the product life-cycle, ending the process with a complete engineering definition of the product. 4. More detail is added to this basic structure in the Pahl & Beitz and VDI models.
Archer	1. Basic description of the development process. 2. Model is generic and not specific to any branch of engineering. 3. Focus on the design phase of the product life-cycle, ending the process with a complete engineering definition of the product.
Large Infrastructure Projects	1. Focus on confirming the technical feasibility and commercial viability of the project due to the large capital investments. 2. Enough design work to determine the technical risks and the project costs. 3. A lot of the detail design work is left for the installation phase to be decided on site.
Suireg	1. Basic description of the development process. 2. Model is generic and not specific to any branch of engineering. 3. Focus on the design and testing phase of the product life-cycle, ending the process with a complete, verified engineering definition of the product.
Pahl & Beitz VDI	1. Very structured, detailed model of the development process. 2. Based mainly on the mechanical engineering process. 3. Advocates the splitting of products into modules. 4. Advocates the division of labour according to this product structure. 5. Focus on the design phase of the product life-cycle, ending the process with a complete engineering definition of the product.
Ullman	1. Focus mainly on the design of mechanical systems. 2. Focus heavily on the product creation portion of the product life-cycle (Figure 2.5) with specific emphasis on the design phase.
Systems Engineering	1. A very comprehensive model, especially suited for the development very complex systems, addressing the complete life-cycle of the product to be developed. 2. The model addresses 2.1 <i>"the technical efforts related to the development, manufacture, verification, deployment, operations, support, disposal of and user training for system products and processes."</i> 2.2 <i>"the definition and management of the system configuration."</i> 2.3 <i>"the translation of the system definition into work breakdown structures."</i> 2.4 <i>"the development of information for management decision making."</i> [19]
Concurrent Engineering Integrated Product Development	1. Focus on the integration of all business activities involved in the development process. 2. Focus on executing these business processes as concurrently as possible. 3. Focuses on the phases of the product life-cycle from strategy to distribution.
Pugh	1. Focus on the integration of all business activities involved in the development process. 2. Model indicates that all these business activities can be of an iterative nature. 3. Focuses on the phases of the product life-cycle from strategy to distribution.

Table 3.2: Strengths and Focus Areas of Various Models of the Development Process

The defining attributes of the prescriptive models are:

- a rigid, broadly linear sequence of phases from problem definition to detail design with an indication of iterative feedback,
- a process structured in phases and therefore lending itself to planning and managing the development process [64], [1],
- concept development based on functional decomposition and analysis, without the use of preconceptions,
- a hierarchical structure of both the problem and the solution,
- resulting in a solution generating process based on disassembly of the problem space, finding solutions for the sub-problems and assembling all these solutions again in the solution space,
- best solution to each of the sub-problems chosen from a number of alternatives.

3.2 Principle Development Rules

When examining the intent of the prescriptive development models described in the previous sections, it becomes evident that they attempt to force the designer or project manager to adhere to the following principles:

- Understand the real need or requirement that the new design is supposed to address.
- Increase participation of those involved in the project by making the development process visible and understandable so that all team members can visualise their contribution in the process. Therefore all team members are aware of the requirements, and everyone's effort is aligned to achieve the desired outcome.
- Consider various ways to address the requirements, so that the best one can be chosen for implementation.
- Consider the whole product life-cycle during the development of a new product, manufacturing, use, maintenance, support, disposal.

3.3 Descriptive Models

In their article, Cross and Roozenburg [33] describe the evolution of models of the development process. They state that "early models of the design process in architecture were often very similar to models of the engineering design process". As examples they refer to similarities of models developed by Marver et al. [92] for architectural design and Asimow [13] for engineering design respectively. Another example cited were models presented by Jones and Thornley at the 1962 Conference on Design Methods [75], which were similar to the Archer's model discussed in Section B.3.

According to Cross and Roozenburg [33], Hillier [65] was one of the first to criticise the engineering model of the development process, and to propose a "conjecture-analysis model" where "*the designer must first generate a solution conjecture which is subjected to analysis and evaluation, rather than analysis preceding synthesis*" as in the prescriptive models of the development process discussed in the previous chapter. Later Drake [42] proposed the "generator-conjecture-analysis model" which states that "*very early in the design process the designer identifies a particular, strong generating concept or a particular set of design objectives*". According to Cross and Roozenburg [33] "*this again contradicts the view that the design process must begin with an exhaustive problem specification, from which solution concepts may then be synthesised*".

Besides the emphasis on "pre-structure" to solve design problems, descriptive models of the development process focus on the cognitive processes employed by the designers during the design work. Therefore the iterative and heuristic nature of the design process is often a central attribute of the model, resulting in a spiral structure in the flowcharts describing the models, e.g. Hillier, Drake and March's model in Figure C.1.

Brief descriptions of the most well-known descriptive models of the development process are provided in Appendix C. The following sections discuss the differences and similarities of these models, concluding the chapter with a brief comparison between the prescriptive and descriptive models.

3.3.1 Similarities of Descriptive Models of the Development Process

According to Cross and Roozenburg [32] the descriptive models of the development process have the following features in common, contrasting them to the prescriptive models described in Section 3.1.

- Descriptive models have "*a spiral structure*" [32].
Describing the structure of the descriptive models as a "spiral structure" seems to the author a little extreme, especially considering models such as the ones by Johannes (Section C.3) and Reymen (Section C.4). However, the descriptive models acknowledge and emphasise the iterative nature of the development process.
- Descriptive models "*assume design problems, by definition, to be ill-defined problems,*" and therefore suggest refinement of "*solution and problem in parallel*" [32].
In the author's opinion, this iterative characteristic of the development process is a consequence of the ill-defined nature (typically) of development problems, and therefore the process of developing solutions to these problems has to be heuristic, where the problem and solution are investigated in parallel. The solution "solidifies" as the understanding of the problem increases. This implies that the development process cannot be specified a priori and remain unchanged through the execution of the process. Having defined an initial process at the start, the process has to be able to change to provide the designer the tools he needs to solve the design problem as his understanding and insight into the design problem increases. Since descriptive models are mostly based on the processes observed in practice, they emphasise this.
- Descriptive models recognise "*the importance of pre-structure, presupposition or protomodels as origins of solution concepts*", in the use of "*the conjecture analysis cycle*" to generate solutions [32].
It is the author's opinion that it is not in human nature to think in terms of abstract concepts. Combining this with the inevitable time pressure in practice, leads the design engineer to use fairly concrete concepts as a proposed solution, which are then refined until they fit the requirements to an acceptable level, rather than an optimal one. Once again this characteristic of practical implementation is reflected in the descriptive models.

3.3.2 Differences Among Descriptive Models of the Development Process

The main differences of the descriptive models of the development process described in Section C are

- | | | |
|---|---|--|
| the level of detail | - | Models by Johannes [73] and Hillier, et al. [65] describe the development process at a fairly high level of abstraction, providing very little detail with regards to the inner workings of the process. On the other hand the models by Hybs, et al. [70] and Reymen [1] discuss the development process in much more detail. |
| the researcher's perspective or intention | - | Each model focuses on the aspects of the development process which the author of each model felt was important for his particular study. For instance, Stempfle and Badke-Schaub [115] wanted to describe the " <i>basic cognitive operations</i> " employed during the development process. Hybs and Gero [70], as another example, were interested what the effect the design environment has on the design process, and concluded that the development process exhibits evolutionary characteristics such as cross-over and mutation, which are normally associated with biology and "genetics-based processes". Reymen [1] established a domain-independent model of the development process by " <i>studying similarities and differences between design processes in three design disciplines</i> " and formalising the common elements. |

3.4 Prescriptive Models Versus Descriptive Models - Conclusions

Both prescriptive and descriptive models of the development process, summarised in Appendices B and C respectively, attempt to describe the process of development. A model, by definition, is an abstraction of

reality to highlight a specific aspect of reality. In order to achieve this a model is usually a simplification of reality, addressing only the aspects it attempts to highlight. When viewed in this light, the two types of models of the development process are just two perspectives of the same reality.

Descriptive models illustrate what happens in practice (this is discussed in more detail in Section 4.1), while prescriptive models describe how the process ought to be executed, according to the author of the particular model [49]. The engineering practitioner has to take both perspectives into account: the former describes the "natural behaviour" while the latter provides the structure for the planning and management of the project. The challenge lies in finding the best balance between the two perspectives.

Archer [12] supports this point of view: "*Systematic methods come into their own, under one or more of three conditions:*

- *when the consequences of being wrong are grave;*
- *when the probability of being wrong is high;*
- *and / or when the number of interacting variables is great."*

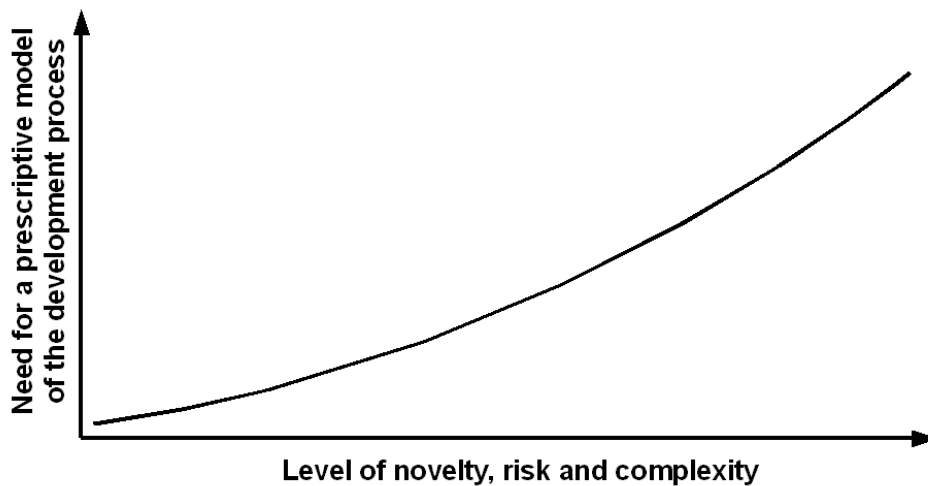


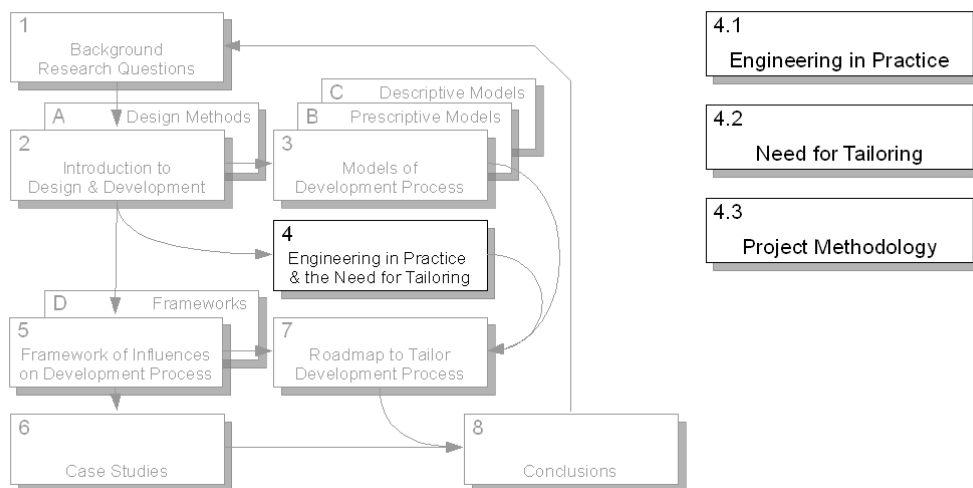
Figure 3.4: The Need for a Prescriptive Model of the Development Process Increases as the Level of Novelty, Risk and Complexity Increases

Chapter 4

Analysis of the Theoretical Models in the Light of Practical Application

Synopsis

Having described various prescriptive and descriptive models of the development process in Chapter 3, Section 4.1 of this chapter starts by contrasting these theoretical descriptions with what happens in daily industrial practice. In Section 4.2 the author presents his case for the need to tailor or adapt the development process to the circumstances surrounding the project to be executed. The Section 4.3 discusses the application of a project methodology in the process of planning, managing and executing a development project. In order to provide a means of determining which changes to make to the development process under specific circumstances Chapter 5 describes the factors that influence a development process.



4.1 What Happens in Practice?

Having discussed prescriptive and descriptive models of the product development process in Sections 3.1 and 3.3 respectively as part of the product life-cycle (section 2.2) the academic and theoretical view of the development process has been painted in reasonable detail. Many researchers agree with Schnelle and Pfeuffer [109] that "*marketable products can only be developed within time and cost constraints if a structured approach is used during the process from need definition to release of production documentation*". To promote these models and to prove their usefulness, researchers have successfully completed development projects in industry which have utilised the prescriptive models as described by design science [62], [130], [109], [40].

However publications such as the final report of the FIDR Workshop by Culley [34], and articles such as the ones by Eder [46] and Frost [54] indicate that there exists a gap between the proposals of design science and the implementations of these proposals by industry. It is interesting to note that this gap is defined by comparing industrial practice with the prescriptive models. Rather than adhering strictly to the prescription of the models of the development process, designers in practice were found to change both the characteristics of the product as well as the development process to suit the context of the project [1], often employing many of the characteristics of the descriptive models of the development process discussed in Section 3.3.

During their investigation, Ehrlenspiel and Dylla [47] found some differences between the methodologies advocated by the prescriptive models of design science and the actual practices in industry:

- Designers in industry tend to search for solutions on a concrete level rather than making use of abstract functional conceptions and structures.
- Design science usually advocates the searching for several solutions and selecting the best alternative, what Ehrlenspiel calls "*productive solution generation*", as illustrated in Figure 4.1. His investigation revealed that this used only 19% of the time. The rest of the time the designers use "*corrective solution generation*", i.e. correction of weaknesses as design progresses, as illustrated in Figure 4.2.
- Similarly design science suggests that the solutions of sub-functions should be found first, to be then combined to generate the optimum solution for whole problem. Quinn [103] found that, in practice, designers generated concepts to "*satisfice*", i.e. rather than finding an optimum solution, designers stopped the process when a solution was found that satisfied all the mandatory requirements. This is supported by Ehrlenspiel and Dylla [47], who found that designers tend to find the solution for the central problem first (focal point), and then solve sub-problems accordingly. This is shown in Figure 4.3.

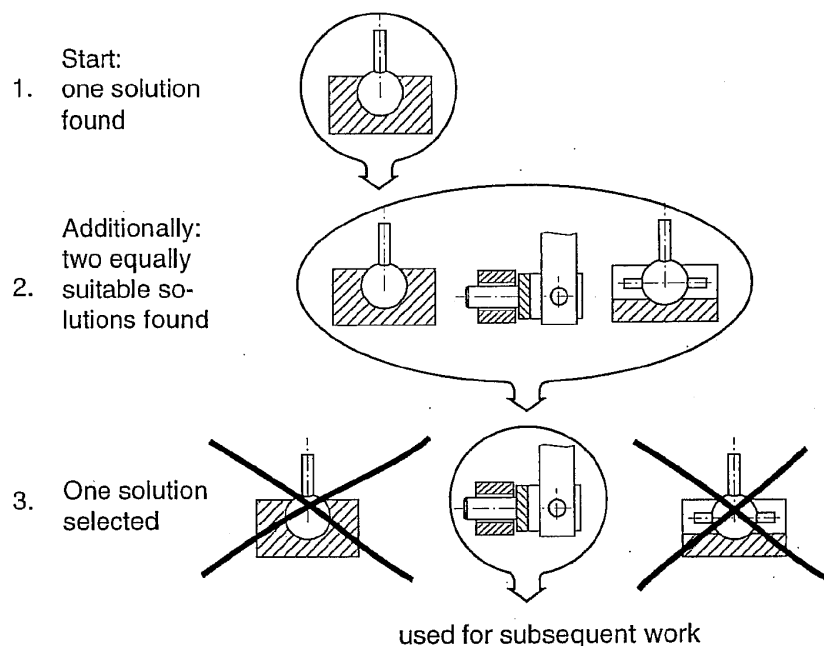


Figure 4.1: Productive Solution Generation [47]

Franke [51] agrees that designers can never truly consider the functional characteristics without the connotation of certain protomodels or pre-suppositions. As an example he suggests that a designer will always visualise a shaft as a cylindrical body with a certain diameter, and a housing as a hollow body with a given wall thickness.

Stempfle and Badke-Schaub [115] found in their investigation that "*natural decision making does not consist of generating many alternatives, the evaluating these alternatives and finally taking a decision*". Rather than striving for "optimum" solutions, humans tend to "satisfice", i.e. choose solutions "*that exceed some (consciously or unconsciously held) threshold*". The natural decision making process therefore "*resembles a sequence of several consecutive screening processes*".

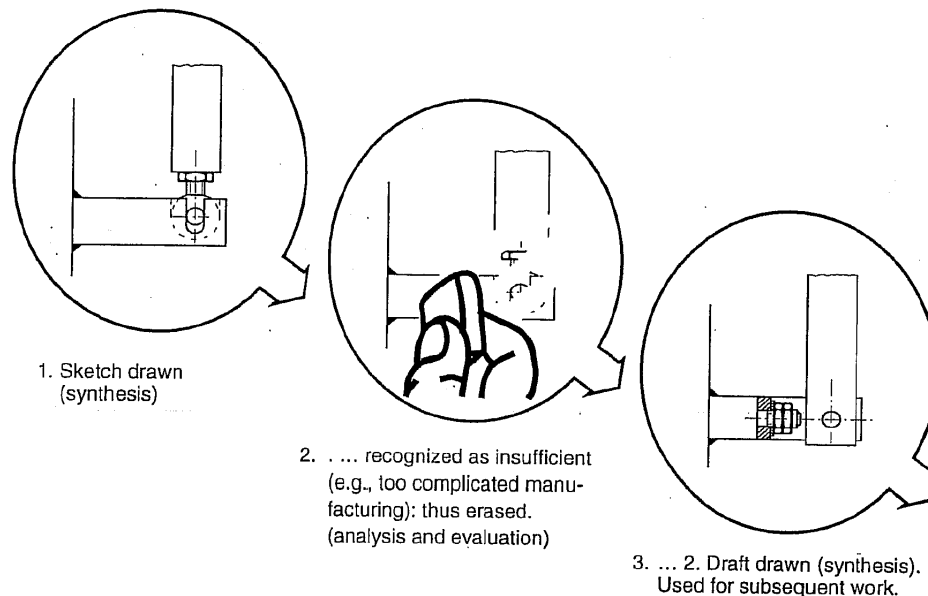


Figure 4.2: Corrective Solution Generation [47]

Von Handenhoven and Trassert [126] also indicate that the prescriptive models of the development process as developed by design science are very seldom implemented in practice, even though every year engineering teams are generating "*products of growing complexity and better quality for lower production costs in shorter periods of time*". They are therefore questioning the premise that "good" products can only be developed using "good" design practice. The argument is put forward that a large percentage of knowledge regarding design is tacit rather than explicit, and an important part of "design skill" is the ability, based on experience, to utilise the different types of knowledge during the development of a product. Prescriptive models of the development process therefore tend to "*interfere with the personal value-system of the engineer*".

These are typical characteristics of the descriptive models of the development process. Hubka and Eder [69] confirm that experienced designers have a "feeling" for the subject matter that allows them to develop good solutions intuitively, without the structured, systematic approach of the prescriptive models of the development process. Andreasen [8] maintains that the design process is "*more a learning process than an algorithm*", indicating that there is more to the design process than the rigid adherence to a prescriptive process.

Reasons for the apparent discrepancy between the prescriptive models of design science and industrial practice are presented by researchers such as Bruseberg et al. [23], Frost [54], Zanker [130], Gouvinhas et al. [57], Von Handenhoven et al. [126] and Maffin [86]. Besides the lack of awareness in industry of available methods and the difficulty of introducing and implementing new methods in an existing environment, the main reasons were cited as follows:

- Design science models and methods are based on the "original design problem", while most industrial design problems are adaptive or variant designs.
- Design science models and methods are not flexible enough to take the constraints and context of a development life-cycle into account.

The former point is clearly supported by Pugh [101], who maintains that a model of the development process should have the following characteristics:

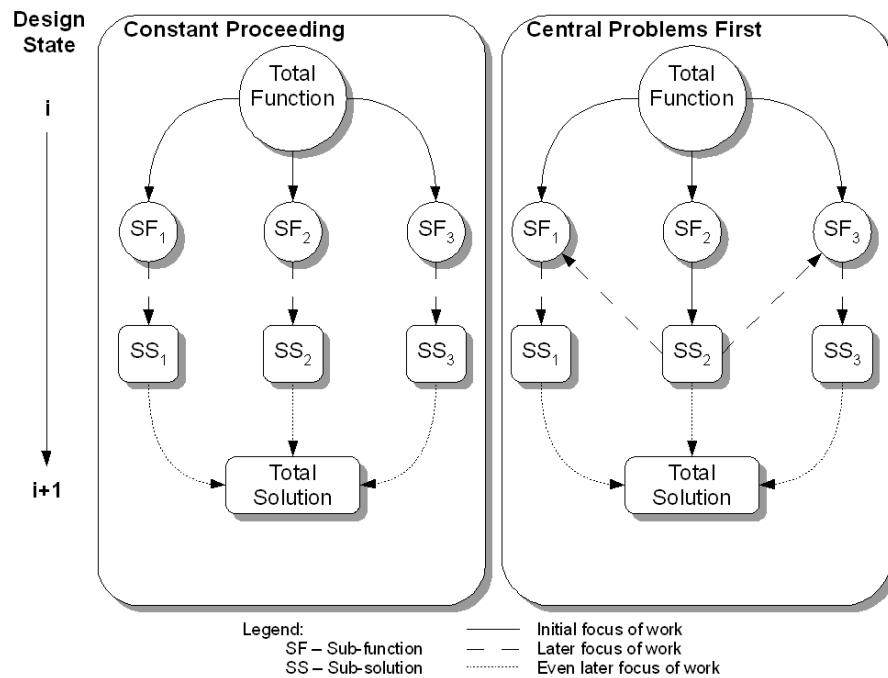


Figure 4.3: Solution Strategies Adapted From Ehrlenspiel and Dylla [47]

- "it must be comprehensive, and"
- "it should preferably have universal application, owing allegiance to neither traditional discipline, industry or product".

Frost [54] explains it as follows:

"Many of the products designed in industry are of an evolutionary nature with incremental improvements during each design cycle. Designers know their own patch of design turf quite intimately; they know what works, what does not work, and how to achieve the pragmatic objectives which are set for their design activities. They have a good intuitive appreciation of the locations of optimal states in the design space, and the paths thereto, and they also have an excellent feel for the 'rates of exchange' associated with the various real compromises which need to made."

Since descriptive models of the development exist in the body of knowledge of design science, as described in Section 3.3, combined with the fact that these descriptive models are actually seen in engineering design practice, as described in Section 4, indicates that development is more than a rigid, pre-determined set of tasks. It is clear to the author that intuition, imagination, the use of presuppositions and the use of "corrective solution generation" [47] and the practice of "satisficing" [103] are all real, legitimate, useful parts of the creative, heuristic process to develop products.

The question is therefore no longer: "Do these characteristics belong in the engineering development process?", since the answer to that question should clearly be an unequivocal "Yes". The question now is: "When and how are these characteristics best implemented?", and "When should these characteristics be avoided?". In the author's opinion the answer to these questions lie in the context of each development project.

4.2 The Need for Tailoring or Adaptation of Development Models

Researchers such as Clarkson and Eckert [29] have indicated that no single model of the development process exists that would be universally acceptable in all circumstances. On the other hand, researchers such

as Frost [54] have indicated that the concepts of design science are actually being applied in industry, the models and methods have just been adapted to suit the context of the specific organisation and project.

In their research regarding user-centred design methods, Bruseberg and McDonagh Philp [23] investigated the use of design methods during product development. She found that designers "*tend not to use the formal methods described in literature. This is mainly due to time constraints. Instead an intuitive approach, using elements of most of the methods, is used.*" The investigations also showed that "*there were considerable differences between individual designers' preferences for methods. This is partly because design projects vary in content and context. Also, each designer seemed to have a personal set of adapted methods, adjusted to particular circumstances and requirements.*"

Eder [46] confirms that each design method has to be adapted to the organisation in general and the development project in particular. Rohatynski [105] is also of the opinion that design methods "*must be adapted in a coherent way to design practice nested in the given field and given company. Introduction of a new method into practice must not ruin the existing mode of engineer-designers proceeding. Evolution is possible and acceptable; revolution shall encounter resistance.*"

Suireg [117] maintains that although the design process "*is not expected to have a rigid set of rules for a heuristic process*" "*there are general guidelines which constitute the basic structure of the design activity*". Unfortunately the article does not discuss how the process could be adapted to suit varying circumstances.

According to Zanker [130] design science provides general, universal models of the development process, while the adaptation of these models to individual, specific context is only inadequately discussed. This leads to the notion that the models presented by design science are experienced as rigid and inflexible. Resistance to the academic theoretical models is due (among other reasons) to assistance being required to derive a "*concrete problem-specific and individual*" development methodology from the general, universal methodologies presented by design science [59] by adaptation to suit the boundary conditions (context) of the specific development problem.

Fulcher and Hills [55] maintain that the development model must cater for projects "*in a context of real people, real organisations and real markets which are characterised by complexity, time pressure, limited resources and changing circumstances*".

Even in the project described by Hales [62], where a special effort was made to follow the prescriptive development model of Pahl and Beitz [97], the development process had to be adapted:

- additional "*design activities*" required accounted for 53% of the engineering design effort, and
- thirteen additional "*methods and aids*" were required, accounting for 74% of the engineering design effort.

In their analysis regarding the difference in development practice when compared with the proposals of design science, the following are some of the issues identified by Gouvinhas and Corbett [57]:

- Models of development process do not "*consider the different cultures of various organisations and the effect that this has on the development process*".
- Development models should be tailored for each project to meet the requirements of each individual project.

Even Franke, who in his article [51] is clearly in favour of a structured, methodical approach to development, concedes that there are instances where a less systematic approach has to be adopted for project under time pressure. He also indicates that for projects where the performance and characteristics of the whole product is very important, it is not advisable to utilise the dis-assembly and assembly approach advocated, for instance, in the VDI guideline [125] (see Section B.7). Under the circumstances it is important to consider the product as a whole rather than an assembly of individual modules.

Günther and Ehrlenspiel [60] have also concluded that there are situations in practice where a structured, methodical approach to development might not be the solution for the success of the development project. Project with small design teams under severe time constraints are examples cited. On the other hand, projects with high novelty content or where an optimum cost-effectiveness must be achieved to be competitive in the market, are projects which would benefit from a more structured, methodical approach.

As discussed above, many of the researchers, who propose models of the development process, support and even advocate the adaptation of their model to the context of the development project. How and under

which circumstances is never clearly indicated. The research described in this dissertation does not intend to provide a prescriptive, rule-based answer to this question. It is the author's opinion that such a rule, if it did exist, would have to be so complex in order to provide for all eventualities, that it would be impractical to implement [30]. The intention of this research is therefore to provide a general guideline, a framework to assist in the decision making process regarding the structuring of the process for each specific development project.

As stated in the previous section, according to the author, the answer lies in the context of each development project. Before the contextual variables of development projects are discussed in detail in Chapter 5, it is necessary to clarify the role of a project methodology in the process of planning, managing and executing a development project.

4.3 Project Methodology: Planning, Managing and Executing Development Projects

The Design Council in the UK has concluded that "*the management of the development process is key to its effectiveness*" [5]. The formalised, structured nature of the prescriptive models make them useful management tools. The previous section, however, clearly indicates that many researchers in the field of design research have come to the conclusion that it is sensible to adapt the models of the development process provided by design science to the context of the development project under consideration. This is confirmed by Pugh [101]: the structured approach, "*whilst imparting a discipline of systematic working, should nevertheless allow for variations to that systematic working, whilst retaining discipline and also imparting comprehensiveness. In other words, the system flexibility within a framework must not impose the rigidity and linearity suggested by many models of the activity*".

By their structured, linear nature the prescriptive models are particularly prone to be seen as a one-size-fits-all solution to development problems. On the other hand descriptive models of the development process are too unstructured and focused on the detail cognitive processes to be useful for the management of a project.

The question now is: "How can this be done?". The author of the dissertation suggests that a development methodology is useful for this purpose.

Section 1.2 cited the definition of a methodology from Pahl and Beitz [17]: "*is used for knowledge about practical steps and rules for the development and design of technical systems, based on the findings of design science and of practical experience in various applications.*" Egbuomwan [49] describes a development methodology as "*a collection of procedures, tools and techniques for designers to use*" when developing a new product.

If a project plan is the definition and allocation of tasks to specific resources for a specific development project, then the project methodology is the philosophy or strategy that was used to create this plan, as depicted in Figure 4.4.

The development methodology is based on a project strategy, which takes into account the project context. The project context, in turn, includes the initial conditions of the project, such as the current state of knowledge, expertise available to the project, the reasons for the project and the background which led to the decision to execute the development project. Boundary conditions of the project such as constraints in terms of lead time, resources available, technical requirements, to name a few, are also included. All projects have this type of information, although it may not be formalised or captured. The project strategy gives rise to the project plan, or the engineering management plan, as it is called in the author's industry.

This engineering management plan defines the next level of detail below the strategy, and it typically consists of the following:

- Specification - Defining the goal, outputs or deliverables of the project
- Budget - Defining the funds available to achieve the project's goals
- Schedule - Defining what is planned to happen when to achieve the goals
- WBS - Defining who does what to achieve the goals

The engineering management plan, and by implication the project methodology, can utilise a model of the development process, particularly one of the prescriptive models, to provide the required structure to the development process. The level of structure required can then be chosen according to the context of the development project, as discussed in more detail in Chapter 7.

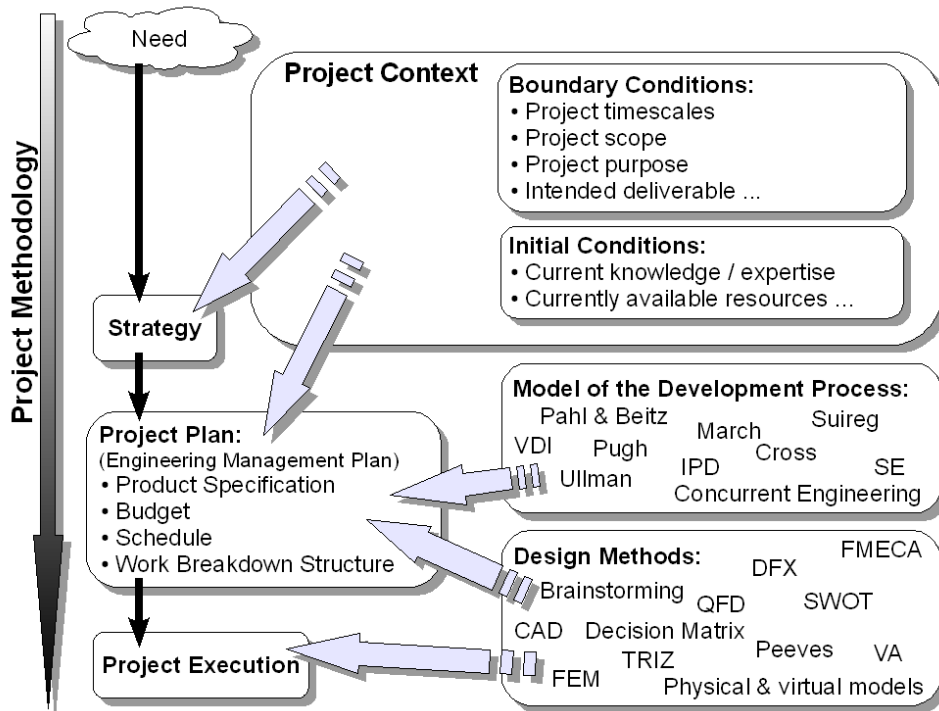


Figure 4.4: Relationship among Methodology, Methods and Models

Why is it important to formalise the methodology?

It is important because the formal methodology is a framework for the project, which, according to Nordlund et al. [95], "is required to facilitate communication by providing a shared vocabulary (requirements, constraints, etc) with a common understanding of their meaning." This is especially true for projects with diverse project teams where high levels of concurrency is required, or for distributed project teams. Even for smaller teams methodologies provide a mental framework "from which the actor will be able to sketch (at intellectual level) immediate and consecutive mini-plans on what and how to do the next action, and also will be aware of why it should be done that way" [37]. A lack of a "common conceptual framework" and misinterpretation of commonly accessible data are some of the typical problems in product development identified by McNeese et al. [93]. Hein [64] agrees that "a common approach to professional procedures and tools" has beneficial effect on product development. This common framework provides a means to structure knowledge in a uniform manner, enabling easy knowledge access, reproduction, recovering and re-use [35]. Therefore formalised, recorded methodologies are not only beneficial for the current project, but also for planning, management and execution of future projects.

The act of establishing a development methodology for a specific project provides another benefit: it forces an understanding, analysis and evaluation of the project on a conceptual level [64]. It is, therefore, important to realise that each project has to be evaluated and the development process models interpreted in the context of their own environment and the problem at hand [86] as illustrated in Figure 4.5. This provides the project manager or chief design engineer with an opportunity to decide how to implement the development principles discussed in Section 3.2 in the most efficient and effective way, given the context of the project.

Rigid application of standard methodologies can be dangerous. Barkan [16] warns: "When specific design rules are handed down from on high they tend to be accepted uncritically, precluding objective thinking by those closest to the problem. Whenever satisfying rules or complying with a methodology becomes the end rather than the means, objective inquiry and critical thought may be cut off, posing a danger to the product development effort. Superficially applied methodologies lead to serious oversights and unanticipated difficulties resulting from overlooking essential considerations or an inadequate grasp of the design rules."

Having established that the standard models of the development process as presented by design science have to be adapted to suit the context of each individual development project, Chapter 5 will investigate the internal and external factors that influence the development process. For development projects the type and scope of the work to be performed, the planning and managing of the project are strongly interrelated. This

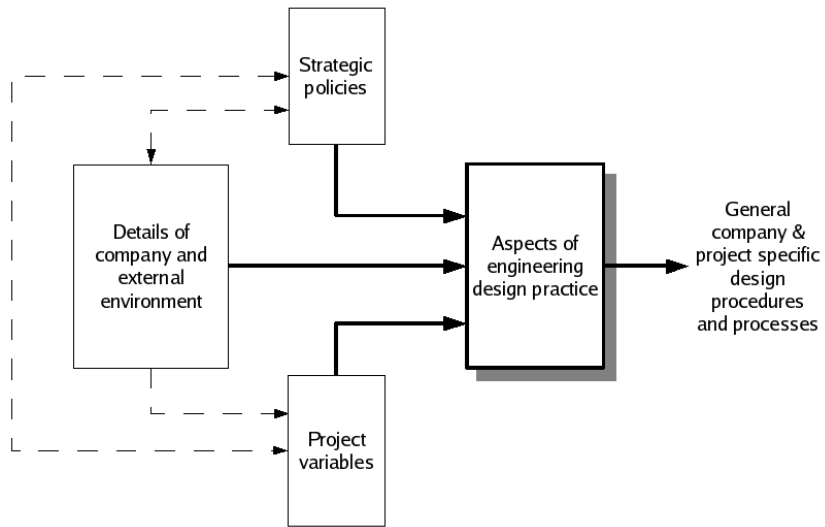


Figure 4.5: Establishing a Design Methodology According to Maffin [86]

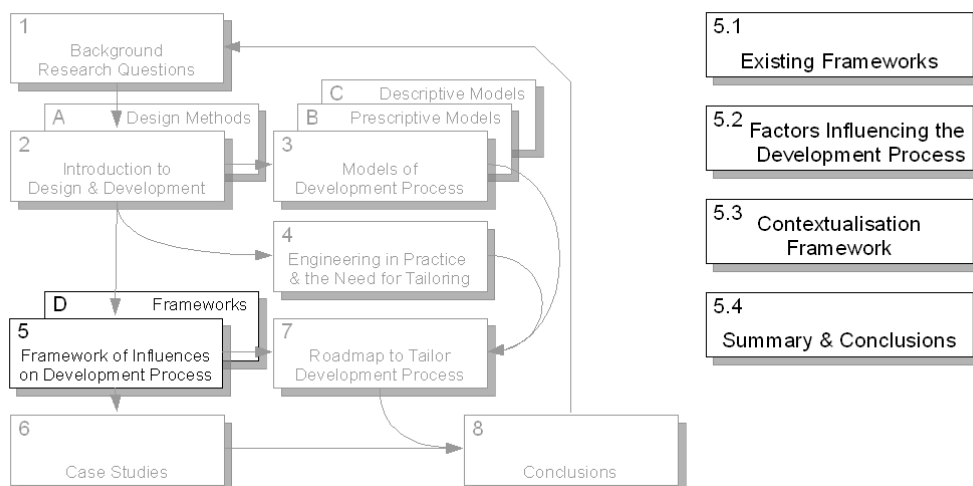
is the reason that many of the aspects discussed in the following sections have a strong project management flavour. Chapter 6 will describe how an analysis of these internal and external factors can be used to structure a development methodology for an industry project in its context.

Chapter 5

Internal and External Influences on the Development Process

Synopsis

In Chapter 4 the author argued that one of the reasons for the differences between the theoretical models and industrial practice could be that the models of the development process have to be adapted or tailored to the context of the development project. The author also proposed the establishment of a formal project methodology as a vehicle to achieve this. Chapter 5 investigates the factors that influence the deployment process of a development project. Section 5.1 discusses the research performed by others to describe and classify the development process, the developed product as well as the companies and their resources which execute the development process. Given this background the author introduces a contextual framework in Section 5.2 to structure the factors that influence the development process, followed by a brief discussion of each of these factors. Chapter 6 evaluates this framework by means of a number of case studies, while Chapter 7 discusses the use of this framework in the process of generating a formal methodology for a specific development project.



5.1 Existing Frameworks

Researchers such as Culley [34], Eder [46] and Frost [54] have indicated that the methods and models generated by design science to facilitate the development process are not being utilised by industry to the expected degree. Among the reasons cited for this phenomenon is that "*methods are not adapted to the given situation*" [82]. The author agrees with Lindemann et al. [82] that "*the designers, the product they have to develop and the conditions they have to work with*" should be the "*central elements*" when considering factors that influence product development projects. The research of Lindemann et al. [82] has led them to the realisation that "*development processes are more strongly influenced by the content and by informal relationships than by officially pre-determined schemes*". Ernst and Young [48] agree that "*best practice in one context can be ineffective in another*". According to Van de Ven [124] et al. "*issues such as domain type, uncertainty, complexity and restrictiveness may play a role in determining whether a best practice is effective*". Reymen et al. [1] that "*the combination of the product being designed, the design process, and the design context was important*".

Appendix D summarises the work of various researchers on parameters that characterise the development process and its context, which includes the product being designed, the designers and the organisation they work for, as well as the circumstances surrounding the development project. Following the Pareto principle, the intent of this research was to focus on the factors with the greatest influence rather than providing a comprehensive list. Also the author intended to provide a practical toll that can be applied with relative ease, even in SME's (refer to Section 1.4). None of the descriptions provided in Appendix D fitted the need for this research exactly. The company classification scheme [87] and the resulting framework of Maffin [86] (Section D.7) seemed the closest to what the author was searching for.

The context model of a design project according to Wallace and Hales[127] (see Figure D.1) is very detailed and comprehensive. In the author's opinion it was the complexity of this model that made it unsuitable for this research.

In comparison to the model by Wallace and Hales, the product properties by Tjalve[121] (see Figure D.2) were too much focussed on the product, i.e. the output of the development process, rather than the process itself. Similarly to the model of Wallace and Hales the list of product properties by Tjalve are comprehensive, and would have made the framework that the author is looking for too complex.

Another complex classification scheme was presented by Barber and Hollier [15]. Although the use of numerical variables would have suited the author's research, this classification scheme was too focused on the management of production in general and batch production specifically. For the purposes of the author's research issues related to the project would have been missing from the model.

The framework by Andreasen also seemed, for the purposes of the author, incomplete: factors related to the production capability and the product to be developed were not included.

The parameters affecting the designer and the design process by Ehrlenspiel (see Figure D.4) focussed too much on factors related to the personnel involved in the design process. Issues such as value system, emotions and motivation are indeed very relevant, but difficult to quantify, and therefore it would be difficult to use these issues as an objective basis for generating a development methodology for a development project. Also, given that these variables are different for each team member, one would have generate an "average" for the whole development team, which is clearly not an easy task.

The field study by Hykin and Laming [71] was very valuable, and confirmed the author's own experience: that the development process is greatly influenced by the intended production quantity of the product. However, in contrast to the comprehensive models and classification schemes described above, factors examined in the field study by Hykin and Laming, did not seem comprehensive enough for the purposes of the author's research.

Given the rationale described above, the author decided to adapt the contextual framework of Maffin [86] as a basis for generating a list of factors that influence the development process the most. The details are described below.

5.2 Description of Factors Influencing the Development Process

Maffin's [86] contextual framework, shown in Figure 4.5, was used as a basis for categorising the variables that influence the development life-cycle. The version as adapted by the author is shown in Figure 5.1. Variables related to the company, its external environment and its strategic policies are considered organisational variables. Any project related variables are summarised under the heading of project variables. Categories for product and personnel variables were added to Maffin's original contextual framework by the author.

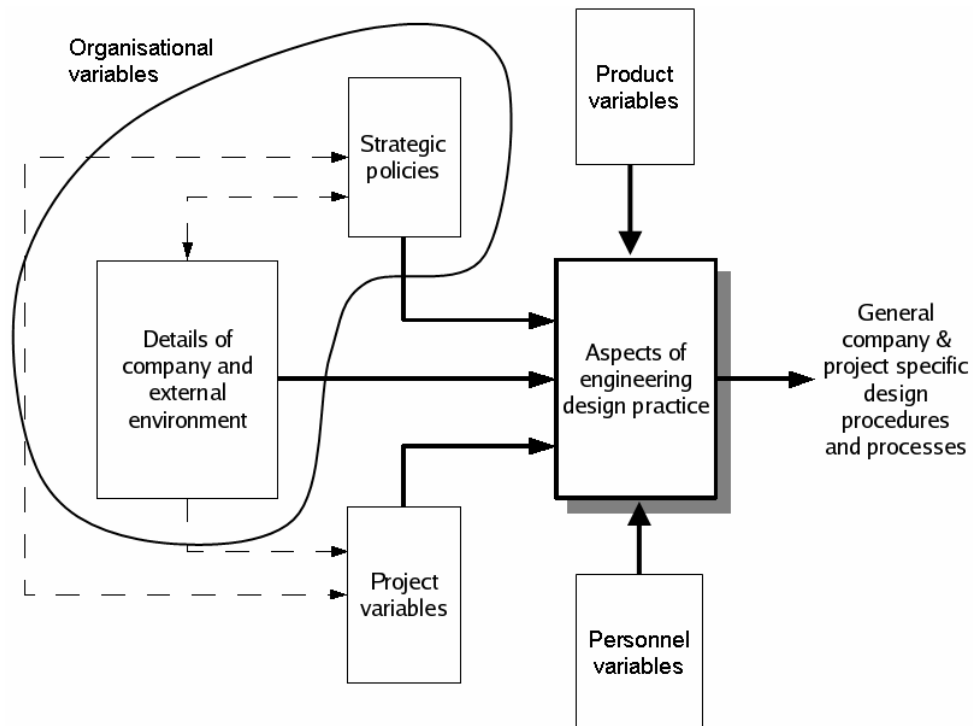


Figure 5.1: The Contextual Framework Adapted from Maffin [86]

5.2.1 Variables Related to the Organisation

Dumas and Whitfield [45] have indicated in their research that "*industry, in its relationship with the design function, is not a uniform domain which can be addressed simply and directly; rather it is segmented with each type exhibiting a unique culture/practice profile. As such, it argues against a simplistic approach to the management of design within industry and suggests a greater need for tailor-made solutions*". Five characteristics of the organisation that influence the product development life-cycle the most were identified by the author. The word organisation in this context represents the environment in which the product is to be developed, including such issues as client and supplier involvement. The factors are

- Organisational size
- Organisational structure
- Type of organisation
- Organisational maturity
- Available design capacity

Organisational Size

The influence of the size of the organisation on the development life-cycle is similar to the effect that the size of the project team has on the development life-cycle: see Section 5.2.2 for more detail. The interrelationship among variables related to size is discussed in Section 5.3.1. In general, the larger the organisation (or project team) the greater the managerial control, and therefore the more formal and prescriptive the development process. Jones et al. [74] quantified typical attributes of an organisation (project) according to its size (Table 5.1).

Attributes	Organisational Size				
	Tiny	Small	Medium	Large	Huge
Professional Employees	3-15	15-300	300-1500	1500-30,000	30,000-150,000
Area	Building	City	Multi-city	Country	World
Structure	Unstructured		Structured		Rigid
Databases	None		Electronic		Elaborate
Communication	Informal		Electronic		Elaborate

Table 5.1: Organisational Sizes According to Jones et al. [74]

As a variable describing the organisation, which will be performing (or at least managing) the development process, organisational size describes in general terms the amount of available resources and implicitly the manner in which the organisation's business in general and product development in particular is approached. This is reflected in Koeller's research [78] which indicates that large firms can afford to be innovative in industries which require "*substantial advertising intensity*" and capital intensive investments. On the other hand small firms find it easier to be innovative in industries where an "*entrepreneurial mode of behaviour*" is favoured. Radcliff et al. [104] have described the working environment of a typical small manufacturing business in Australia, and the pertinent points relating to the product development process are summarised below. The author of this research has worked in South African companies of various sizes, and can confirm through experience that these descriptions are also typical for South African companies.

Essentially, the working philosophy in a small organisation is characterised by the limited availability of resources. They often employ no professional engineers, limiting their exposure to formal design methodologies and methods. Key personnel usually "*perform multiple roles*" within the organisation and therefore have to "*perform a diverse range of tasks without the option of delegation*". This means that development projects have to be fitted in between the daily tasks required for the running of the business. As a consequence "*internal communication tends to be informal and verbal with frequent in-the-corridor meetings*", resulting in very little information being recorded, collated and reported in a formal structured manner. Therefore "*much of the daily working knowledge of the enterprise resides in the memory of the staff*". "*It is an oral tradition with shared memories and stories, poorly supported by documentary evidence, and strongly centred in the moment*". This results in a "*tendency to concentrate attention on short term rather than long term: the current production crisis over-rides the less pressing yet ultimately more important strategic issues*".

The converse is true of large corporations: the availability of resources makes it possible to appoint personnel for a specific task, such as the development of a new product. Strategic issues can be resolved without the interruption of the current production crisis or customer complaint. Those are addressed by the relevant departments. Not only are resources available to record, collate and report information, it is imperative that this is done to ensure effective and efficient communication. The co-ordination and management of a large company cannot be achieved by in-the-corridor meetings.

Roy and Potter [107] confirm that the "*lack of expertise and generally limited resources may be the major constraints affecting small companies, in contrast to the resistance to change and organisational inflexibility that often characterise larger firms*". Rothwell and Zegveld [106] maintain that "*the main advantage of the small and medium sized enterprises in undertaking innovation is good internal communication*". Walsh and Roy [128] determined through a study by the Design Innovation Group of the plastics industry that "*the transition to formalised management structures occurred when a firm reached about 100 employees*" and the commercially successful firms "*were the ones that had successfully made this management transition. Once a firm grows above such a threshold, the small-firm ways of coordinating new product development are likely to become impractical and will inhibit growth*".

Organisational Structure

Dumas et al. [45] have indicated how the existence of the position of design manager can influence the way that product development is implemented. Companies employing a design manager seem to assign greater importance to the design function within the company. The design department is therefore often operated as a profit centre, accepting a high level of accountability for the designs developed. In comparison companies without the position of design manager prefer central control of design projects with very little accountability expected from the designers. Accountability for design is therefore diffused among finance, engineering and marketing.

Maffin [86] investigated the influences of company structure on the development life-cycle. The "*degree of autonomy over the strategy-making process*" was found to be important: "*an independent establishment will have total autonomy over the process*". The word "unit" in this context can refer to organisational structures of various sizes. A department or team of a large organisation can be given authority to plan and execute a project autonomously from the rest of the company. This can also apply to the local branch, division or subsidiary of a national (or multi-national) company. Conversely, if the unit is forced to operate within the constraints of the larger organisation, generally a homogeneous (and therefore prescriptive and static) approach will be followed, that fits the requirements and philosophy of the larger organisation, rather than those of the smaller one.

Autonomy over the development includes the freedom to adapt the development process according to the context and needs of the project. Autonomous units may decide to skip a formal concept stage, since the product has low complexity (see Section 5.2.3) or because the circumstances have made a concept study redundant and the level of knowledge and understanding is such that the project can proceed with the embodiment stage. Large organisations may, on the other hand, require that all project follow a set programme irrespective of the context of the project. On a more detailed level: an autonomous organisation (unit) may decide that a FEM analysis is not required since the achievement of timescales has a higher priority and the associated risks are at an acceptable level. A dependent unit within a larger structure may be procedurally required to perform certain actions (analyses) irrespective of the specific context of the project. Brown and Eisenhardt [22] call this autonomy "*subtle control*", where senior management communicates the vision for the new product and then providing "*enough autonomy to be motivated and creative*". This, obviously, goes hand-in-hand with senior management support, which is required for the provision of both "*financial and political resources*".

Organisational Type

Aspects of the organisational type that influence the development life-cycle are discussed in the following sections.

Purpose

The first aspect is the organisation's main purpose [86]: is it trying to be product leader, i.e. develop products which excel in performance, quality and reliability, or an effective manufacturer, i.e. concentrate on effective manufacturing and cost minimisation? The first can be seen as a product developer with a manufacturing facility, while the second is a manufacturing plant with a design office to handle minor changes to improve efficiency and cost. Typically the former will concentrate on the initial part of the product life-cycle by generating the product definition and documentation, producing prototypes and assisting in the generation of pre-production models. The latter typically obtains the product definition from the former and produces the product in volume (the second half of the product life-cycle).

The industry sector seems to also influence the development process [45]. In the manufacturing sector the design function is dominated by engineering, internal design teams are utilised and accountability is retained by the designers. In the service industry, on the other hand, the design process is dominated by marketing and is reliant on external consultants and design policy documentation.

Therefore the emphasis within the product development life-cycle shifts according to the main purpose of the organisation executing the project. Cooper [30] also describes how the development process is influenced by the way that organisations perceive themselves: is the development of a new product driven by the demands and requirements of the market (market orientation) or opportunities enabled by technological advances (design domination and/or prototype domination). For the former the development process is dominated by marketing activities (close to 75% of the project duration), while the latter is dominated technical and

production issues (up to 79% of the project duration).

The description above obviously shows the extremes of the spectrum in black and white, while in reality there are shades of grey.

Production Size and Inventory Philosophy

The second and third aspects of the type of organisation to influence the development life-cycle (production size and inventory philosophy respectively) are clearly related to the organisations main purpose as discussed above. Production size can either be unit, batch or mass production for discrete products [84]. Although the development of continuous product, such as in the petrochemical industry, for instance, is not considered in this research, the plants to produce the continuous product are discrete products themselves, and therefore considered as part of this research. The inventory philosophy can be one of four options: make-to-inventory, assemble-to-order, make-to-order or engineer-to-order [84]. Production size and inventory philosophy are related: generally products engineered-to-order are produced in unit or batch production, while make-to-inventory products lend themselves to mass production. As a parameter of the organisation, production size influences the development life-cycle by defining the norm for the company. The effect of production size on the development life-cycle as product related parameter is discussed in Section 5.2.3. The point being that a company accustomed to producing a specific product type, for instance engineer-to-order, will have to change the development methodology, as discussed in Section 5.2.2, if it should decide to develop a made-for-inventory product.

Selling Products

The fourth aspect is concerned with the organisations mode of selling products:

- selling standard products through catalogues,
- selling customised products and competing through tenders or
- suppliers operating with the supply chains of a limited number of customers, establishing long term sales agreements.

These aspects are related to the production size and inventory philosophy. Products produced in mass production lend themselves to long term sales agreements, since this agreement lowers the risk for investing in the establishment of capital intensive mass production facilities. Engineered-to-order products, on the other hand, are often customised products sold through tenders, since each customer has different, specific requirements. As a third variant, make-to-inventory, assemble-to-order and make-to-order can be standard products which are sold through catalogues, where the customer can choose a specific configuration made up of a combination of standard components. Just as with production size and inventory philosophy discussed above, the product development methodology is chosen according to the product type (Section 5.2.3).

Organisational Maturity

Organisational maturity relates to the management of processes within the organisation. The various levels as defined by Jones et al. [74] are shown in Table 5.2. Although Jones established this table specifically to indicate an organisation's maturity to manage the process of identifying, analysing, verifying and validating requirements during the development process, the author feels that the maturity models also applies to the management of the development process as a whole.

The maturity model indicates how various key aspects of the development process are executed during a development process. These key aspects are:

- Project management - The process of managing all aspects of the project, i.e. technical integrity and quality, timescales, financial expenditure.
- Documentation - The process of capturing information during the design process for internal communication as well as support to the customer.
- Computer & tool support - The support of the use of computers and design tools during the development process.
- Tool use learning - The process followed to acquire the skills to use new tools effectively and efficiently.
- Selection of tools - The process of deciding which tools are applicable to the specific development process.

These aspects can be implemented during the development process in a number of ways, summarised here in the form of levels. Level one refers to a development process which is managed and executed in an ad hoc fashion. At the second level each team member may implement some of these aspects to some degree, but no overall philosophy is applied. On the third level a common philosophy towards these aspects is accepted and implemented by the whole project team, although this may vary from project to project and from team to team. For the fourth level organisation wide consensus is obtained with regard to the philosophy of managing and executing development projects. The last level indicates that the development process is evaluated on a regular basis to determine how it can be improved.

Levels	Parameters				
	Project management	Documentation	Computer & tool support	Tool use learning	Selection of Tools
1	Not done				
2	Individually implemented				
3	Project wide selection, support & implementation				
4	Organisation-wide procedures and support				
5	Improvement method followed				

Table 5.2: Levels of Organisational Maturity [74]

It is reasonable to assume that an organisation's level of maturity with regard to the management of aspects described above will also be reflected in the planning, management and execution of the development process. A company at maturity level one, may have a very unstructured, uncoordinated approach to the design of their products. In comparison an organisation situated generally at level four or five, will find it easier to have a uniform approach to product development and evaluate this approach on a case-by-case basis to maximise the performance of the development process.

Organisational Design Capacity

The last organisational variable to influence the development life-cycle is concerned with the organisation's available design capacity. Should the organisation's design resource not be sufficient to develop a specific product, the organisation can either employ design staff or sub-contract a portion of the design. For the former provision has to be made in the development process for training and/or mentoring of the new employees. For the latter, the development project should be structured in such a way that the subcontracted portion consists of a definable sub-system or module to facilitate the interface of the sub-contracted work with the rest. Not only is the structure of the development process different, the type of activities and the required capabilities also differ. When a job is subcontracted the activities of the client are mainly focused on formulating requirements and verifying that the subcontractor delivered what was agreed. When a development job is performed in-house the major effort is to generate the know-how to execute the job in such a way that the requirements of the job are met.

The issue of in-house capacity is therefore often linked to in-house capability (Section 5.2.4): the execution of the project involves a portion for which the company does not have the necessary capability, and therefore it sub-contracts that portion of the project.

5.2.2 Variables Related to the Project

Shenhar and Wideman [111] describe a project as a process, "a journey through time", where the "boundaries or limitations imposed on the journey" are "expressed in terms of scope, quality, time and cost". In the case of a product development project the outcome of that journey is the definition of a product and its production process. Project related factors that influence the development life-cycle, include

- Project size
- Project type
- Project complexity
- Project constraints
- Project novelty

Project Size

Project size relates to the number of resources as well as the project and communication structure required for a specific project, as shown in Table 5.3 by Jones et al. [74]. Project size is often strongly related to project complexity which is discussed below. It should be intuitively obvious that it will take a larger project team to develop a complex system, such as a battleship, in a reasonable time, compared to developing a simple products, such as a desk lamp.

Attributes		Project Size				
		Tiny	Small	Medium	Large	Huge
Design Professionals	number	3	15	75	400	2,000
Duration	months	12	24	36	48	60
Area		Building	City	Multi-city	Country	World
Interface Parameters	number	9	90	900	6000	30,000
Product Environment	type	Indoor	in/Outdoor	Outdoor	Global	Space/Ocean
Requirements	pages	15	150	500	2,000	50,000
Requirements	number	150	1,500	5,000	20,000	500,000
Specifications	number	1	3	10	200	1,000
Examples		Electric kitchen mixer	Housing development, TY set	Night vision equipment	Aircraft	Space vehicle, battleship

Table 5.3: Project Sizes According to Jones et al. [74]

Similar to the effect of organisational size, increasing project size requires increasing managerial control and therefore, increasingly, a more formal structure of the development life-cycle. Also the amount of project documentation increases as part of an increased coordination and communications effort. Therefore large projects may be forced to follow all the phases of the development process in a structured manner, as discussed in Section 3.1, not because of the work content, but rather as a part of the co-ordination and communication effort. In other words: to make sure that all team members are singing from the same page of the same hymn book. Due to the informal and efficient communication channels in small organisations (teams), the development process can often be significantly streamlined by reducing (or even removing) some of the phases. This is often referred to as the "skunk works approach". The interrelationship among variables related to size is discussed in Section 5.3.1.

Project Complexity

Project complexity has the same effect as project size on the development life-cycle, since large projects are usually inherently complex. However, even the development of simple projects can have complex projects. If, for instance, the development process requires the involvement of external organisations, e.g. legal experts,

approval authorities or independent test facilities. This is often related to the product complexity (see Section 5.2.3) and the level of product in the system hierarchy (see Section 5.2.3). According to Günther [59] complexity has the following attributes:

- complicacy - a problem with a large number of variables is more complex,
- dynamics - complex problems are often time dependent, making it very important that the intervention is performed at the right time,
- opacity - only a portion of the relationships among project variables are visible at any time,
- interdependency - project variables are interrelated so that unintentional side-effects are created by adjusting project variables with the aim of creating a beneficial effect, and
- multiple objectives - striving for the achievement of multiple, often conflicting goals simultaneously.
(German: Polytelie)

Therefore complex projects may require a very formal and structured development process in order to integrate the various aspects of the project. The interrelationship between project and product complexity is discussed in more detail in Section 5.3.3.

Project Type

Ullman [123] defines the project type as follows:

- "original design - is a new development, which is not based on a previous product or idea,"
- "parametric design - where parameters are adjusted to suit requirements,"
- "configuration design - which requires mainly the packaging of existing modules or components,"
- "selection design - where the correct standard component to suit the requirements is selected from a catalogue and"
- "redesign - which requires modifications to an existing product to meet new requirements."

Evbuomwan et al. [49] distinguish between

- routine design - which is "derived from a common prototype with the same set of variables or features and the structure does not change",
- redesigns - involving the modification of an "existing design to satisfy new requirements", and
- non-routine - original or new designs.

According to Evbuomwan et al. [49] redesigns can be either adaptive or variant designs: the former involves "adapting a known system (solution principle remaining the same) to a changed task" by introducing "improvements and detail refinements", while the latter "follows a extrapolative or interpolative procedure" to generate "further geometrically similar designs of different capabilities".

Selection design typically requires the least amount of engineering effort, is technically quite simple, and this should be evident in the extreme simplicity and reduction of the development life-cycle. Parametric design, configuration design and redesign projects are typically more complex and technically challenging. They also represent the majority of projects in practice. For these projects the concept phase would be limited to establishing the new requirements and a brief investigation into the new aspects of the project. Although the emphasis during the embodiment and detail phases will also be on the new aspects, the original data will have to be considered carefully to ensure alignment with the new requirements. Obviously the original design problem requires the complete development life-cycle from concept phase to detail phase, as described in design literature, such as the VDI 2221 [125] guideline. Table 5.4 shows how the development life-cycle can be adjusted depending on project type, according to the VDI guideline [125].

It should be intuitively clear that project type and project novelty are strongly related. For instance, a non-routine or original design has a high novelty content since it is, by definition, not based on a previous product or idea. However, the relationship is not quite so obvious when redesign, parametric and configuration designs are considered. Although one might usually be inclined to assign a low novelty value to these type of projects, case study B (Section 6.2) presents a case which involved a design of a new product based on an existing one, i.e. redesign. In this case, however, the project novelty was estimated as "medium" since the user requirements forced the design parameters outside of the accepted, normal range for this kind of product.

Project Type		Project Phases		
		Concept	Embodiment	Detail
Original	complete	██████████	██████████	██████████
	research	██████████	██████████	██████
	quotation	██████	██████████	██████
Parametric	complete	████	██████████	██████████
	re-design	████	██████████	██████
	quotation	████	██████████	
Configuration			██████████	██████████
Selection			██████████	

Table 5.4: Project Phases for Various Project Types [125]

Project Constraints

Project constraints that affect the development life-cycle are typically time or project duration, cost and technical risk. These are obviously conflicting variables. To minimise the project duration, additional resources (personnel, subcontractors) are required. Also highly efficient communication and strong project management, with clear goals for each of the resources involved, is necessary. The minimisation of cost is highly dependent on the type, complexity and risk of the product to be developed. Cost could be reduced by favouring computer simulation rather than the construction of prototypes; or modules or sub-systems could be purchased rather than developing them from scratch. All of these will have an effect on the planning, structuring and execution of the development life-cycle as well as the scope of work. The last project constraint, technical risk, is usually strongly linked to the project attribute, "level of novelty", discussed below.

Project Novelty

The word "novelty" refers to the new aspects of the project, the unknown. Novelty does not necessarily mean that it has never been done before; it could merely mean that it has not been done by this specific company or project team before. Frost [53] defines the level of "novelty" or innovation required as follows:

- If the designed entity is already known in stereotype form, and parameter values only (not configuration) need to be determined, then a very low degree of innovation is required.
- If the design is evolutionary, modifying or extending a known stereotype configuration, then a low-to-medium degree of innovation is required.
- If several different known stereotypes need to be adapted and integrated to form a new type of entity, then a medium-to-high level of innovation is required.
- If no viable pre-existing stereotype is available, so that configuration must be fully determined, then the design is revolutionary and a high level of innovation is required.

According to Thomond and Lettice [119]: "innovations can be thought of as falling onto a continuum from evolutionary to revolutionary:

- *Incremental or evolutionary innovations that improve the performance of established products, services or business models "along the dimensions of performance that mainstream customers in major markets have historically valued" [27]*
- *Revolutionary innovations fall onto a continuum ranging from "radical incrementalism" that delivers significant change to the mainstream market which is mostly competence enhancing with low environmental turbulence and low market uncertainty - to totally "disruptive innovations" that deliver transformational change to the mainstream market and its value attributes which, are mostly competence destroying with high environmental turbulence and high market uncertainty."*

This continuum is shown in Figure 5.2.

Design is described as a heuristic process. This means that the exact process to follow and the expected results are unclear at the start of the process. These only clarify as one progresses by learning about the

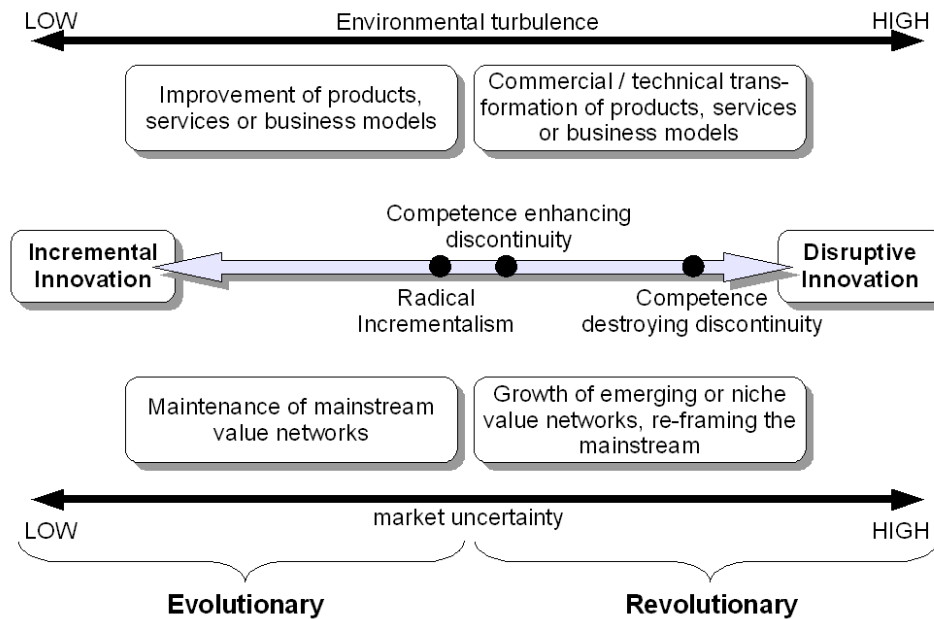


Figure 5.2: Innovation Continuum According to Thomond and Lettice [119]

subject matter, the process and the complete scope of work during the execution of the project. This is especially true for projects with a high novelty content.

It is for this reason that the project plan and the scope of work for projects with a high novelty content are fairly vague at the beginning of the project, becoming more specific and concrete as the projects develops. High levels of novelty therefore require adjustments to the development life-cycle to accommodate the obtaining of information, learning, experimentation, even set-backs and failure. Provision has to be made for "feedback loops", e.g. with reference to the flowchart in Figure B.7, a part of the phase concentrating on solution principles or realisable modules may have to be repeated due to insight gained while laying out the key modules.

The term "project novelty" relates strongly to "technical uncertainty", which, as Shenhar and Wideman [111] point out, requires "*higher levels of process and component integration and testing*" resulting in a "*need for higher and more formal project management*". According to Brown and Eisenhardt [22] products with a higher level of novelty "*more experimental tactics, including frequent iterations of product designs, extensive testing of those designs, and short milestones*" are applicable. On the other hand, "*for stable products*", i.e. fairly low level of novelty or evolutionary innovation, "*product development is a complex task for which tactics such as extensive planning and overlapped development stages are appropriate*". Franke [51] agrees that for products with established technology, such as pumps and internal combustion engines, it is senseless to develop the product geometry from the first principles of the physical effects, because well established, optimised solutions exist for these type of products.

Frost [53] has categorised typical design according to the varying degrees of innovation required, as shown in Figure 5.3. He describes the intent of the figure as follows:

- *items with no asterisk should be regarded as known stereotypes at the time of consideration,*
- *items with one asterisk should be regarded as known stereotypes requiring significant modification or extension at the time of consideration,*
- *items with two asterisks should be regarded as being new syntheses of adapted pre-existing stereotypes or models at the time of consideration,*
- *items with three asterisks should be regarded as having no usable stereotype or model available at the time of consideration.*

If several different known stereotypes need to be adapted and integrated to form a new type of entity, then a medium-to-high level of innovation is required. If no viable pre-existing stereotype is available, so that

configuration must be fully determined, then the design is revolutionary and a high level of innovation is required.

REVOLUTIONARY HIGH LEVEL OF INNOVATION (MORE DIFFICULT)	10	
		Helicopter***
		Watt's Linkage***
	9	Lockstitch Fabric***
		Pitch/Surge Ocean Wave Energy Converter**
	8	Orbital Engine**
		Walschaert's Valve Gear**
	7	Automatic Transmission**
	6	Wave-Following Crane for Offshore Platforms**
		Zoom Lens*
	5	Engine Management Systems*
	4	Radial Piston Engine*
	3	High Speed Passenger Lift
	2	Escalator
STEREOTYPE LOW LEVEL OF INNOVATION (EASIER)	1	
		Centrifugal Pump, Shell & Tube Heat Exchanger

Figure 5.3: Level of Innovation According to Frost [53]

5.2.3 Variables Related to the Product

Product related factors that influence the development life-cycle encompass:

- Product type,
- Product complexity and
- The product's position in the system hierarchy

Product Type

Product type defines the intended production quantity for the product [84], either unit, batch production or mass production. Whereas production size, discussed in Section 5.2.1, is a characteristic of the company, reflected in its business philosophy and processes, the product variable addresses the notion that a product's intended production quantity has an effect on the design of the product. It is therefore seen as a characteristic of the product rather than the entity that produces it. The interrelationships of these issues are discussed in Section 5.3.

Hykin and Laming [71] found that the intended production quantity has a major influence on the development life-cycle: development of mass produced products require emphasis on the first phases of the process (for

reference see Section B.7) to ensure the design of "a nearly right solution for the final design phase and production". As a consequence it requires a "formal, rigidly defined system of control with a well-developed command hierarchy". Also, for mass-produced products the production process is at least as important as the product itself, and therefore the design of the production process requires a similar amount of work as the development of the product. In these circumstances philosophies such as concurrent engineering can contribute greatly to the reduction in development times. In contrast the development of one-off products is "less ordered and constrained" with an organisational structure which is "less formal and more adaptable". The phases of the development life-cycle are "less easily separated" and the emphasis is on the "final design phase and construction of the product". The early phases will attempt to generate as much flexibility as possible so that detail decisions can be delayed until the final phases of the project.

Frost [53] indicates that the intended production quantity introduces "implications for the optimal trade-off between design time and costs, on the one hand, and production or building costs, on the other hand". If the production quantities are large, say above 100 units, the units cost is not greatly affected by the amortised development costs. "Therefore, it also has implications for the trade-off between designing specifically optimal subsystems or components, on the one hand, and selecting readily available proprietary items, which are acceptable but probably not optimal for the particular situation, on the other hand".

Product Complexity

Frost [53] has categorised some typical designed entities in terms of their complexity as shown in Figure 5.4.

COMPLEX (MORE DIFFICULT)	10	Major Chemical Plant/Refinery
		Large Ship
		Large Aircraft, Submarine, Power Station
	9	Diesel-electric Locomotive
		Automatic Transmission
	8	Motor Car
	7	Small aircraft, High Speed Passenger Lift
		Manual Transmission, Sewing Machine
	6	Pitch/Surge Ocean Wave Energy Converter
	5	Escalator
	4	Centrifugal Pump
	3	Gate Valve
	2	Lockstitch Fabric
SIMPLE (EASIER)	1	Hose Fitting

Figure 5.4: Level of Complexity of Various Designed Entities According to Frost [53]

Product complexity can be defined similarly to project complexity defined by Günther [59], in terms of complexity, opacity, interdependency and Polytelie (multiple objectives). The factor of dynamics as described in Section 5.2.2 is more relevant to the complexity of projects than products.

Product complexity was found by Maffin [86] to have the following effect on the development life-cycle. The

development of low complexity products showed a short concept phase, no embodiment phase and moved directly into the detail phase. High complexity products, however, required a definite concept phase and a distinct embodiment phase, connecting the concept phase to the detail design phase. Maffin [86] labelled the former design strategy "type A" and the latter "type B". Analysing 10 case studies and 17 answers to his survey, the correlation between project/product complexity and design process philosophy obtained is shown in Figure 5.5. The figure clearly shows that a strong correlation exists between the complexity of the project and/or product and the type of design strategy adopted.

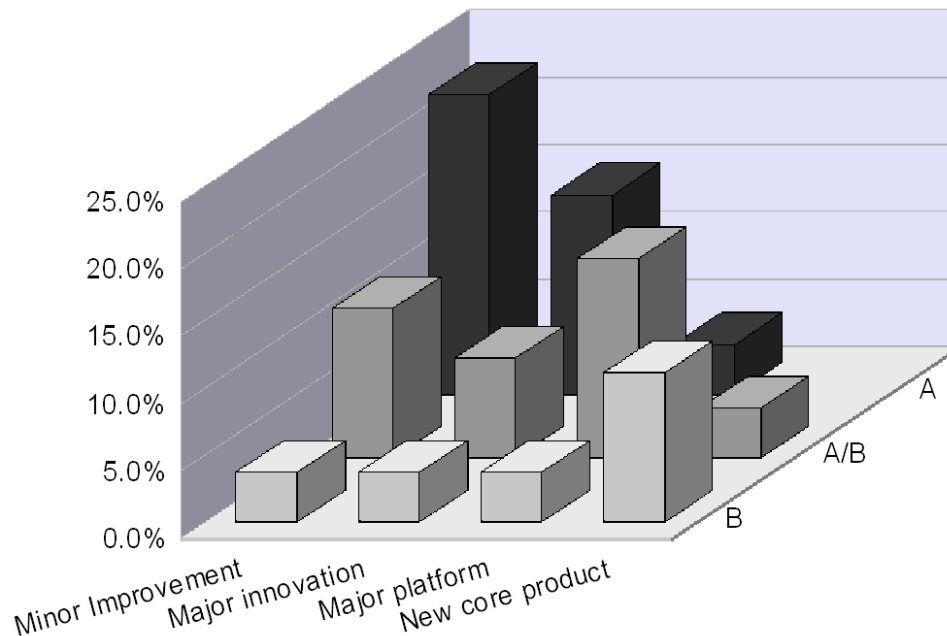


Figure 5.5: Correlation Between Project/Product Complexity and Design Strategy [86]

Maffin also reports a correlation between the complexity of the product and use of a distinct embodiment design phase in the development process. He found that "*embodiment design, which bridges the gap between the overall design concept and the detail requirements, by refining the concept and establishing sub-system requirements, is a distinct phase for complex products, whereas for low complexity products this is achieved by definition in the concept design phase.*" This correlates with the findings of Shenhar and Wideman [111] that increasing complexity also increases "*the need for higher and more formal project management*".

Many of the prescriptive development methodologies (see Section 3.1.1) maintain that design problems can be dissected into sub-problems which are addressed more or less in isolation. The resulting sub-solutions can then be re-assembled to generate the total solution for the original design problem. Although this may be true for the design problems at the extremes of the complexity spectrum, there are design problems in the centre of the spectrum that can be approached differently. For simple design problems the disassembly and re-assembly process is easy to visualise and implement. Very complex design problems are too complex to be handled by a single individual and have to be broken up into modules or sub-systems. This is discussed in more detail in Section 5.3.3, since it applies not only to complex products but also to complex projects.

The product complexity is often related to its level in the system hierarchy, and this aspect is discussed below.

Product Level of Hierarchy

Günther [59] provides a typical level of hierarchy: plant, machine, sub-system, and component.

As far as the development life-cycle is concerned, the effect of the level of hierarchy is the similar to the effect of product complexity discussed in the section above, since the complexity increases with increasing level of hierarchy.

On the other hand, Frost [53] argues that products on high levels of hierarchy, such plants, vehicles, ships, although complex, can typically be conceptually decomposed into subsystems. "*The more this can be done, the easier will be the design, the less holistic the approach need be and the more people will be able to work ef-*

fectively and simultaneously on the design once physical and performance interfaces between systematically adjacent subsystems have been defined." This is illustrated in Figure 5.6.

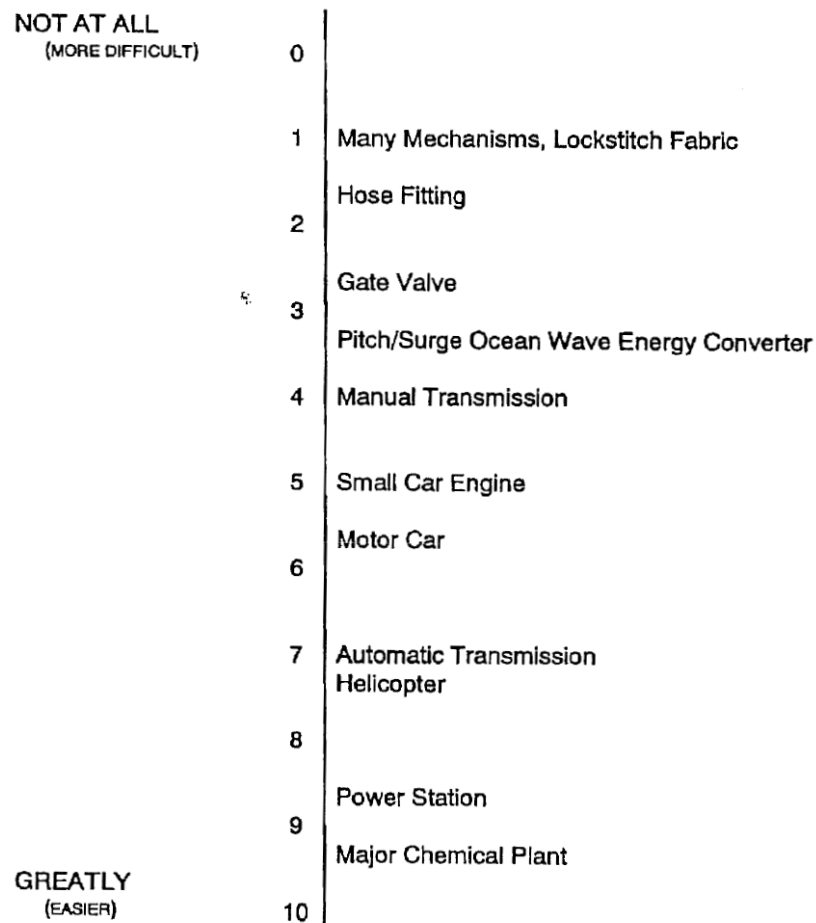


Figure 5.6: Amenability of Some Designed Entities to Decomposition into Sub-systems According to Frost [53]

The development process is also affected by the increase in distance to the end user as the product to be designed moves down the levels of hierarchy. The end user is usually interested to obtain a complete system from one source - the supplier responsible for the system as a whole. This supplier may, and in all probability will, subcontract a portion of the development to others. Therefore for someone designing a sub-system or component, the context of the overall project will be lost, and he will probably not have insight into the overall requirements of the user. While it is favourable for effective communication to be as close to the end user as possible, the sub-contractor has the advantage that he will obtain his requirements already translated from the non-technical user requirements to the technical specification specific for the integration of the sub-system (or component) into the larger system.

5.2.4 Variables Related to the Personnel

The following are personnel related factors that influence the development life-cycle:

- Team size
- Level of maturity
- Level of design capability

Team Size

Team size is strongly related and has the same influence on the development life-cycle as the size of organisation and project size discussed in paragraphs 5.2.1 and 5.2.2 respectively.

These three variables are not necessarily identical, however, and this is discussed in more detail in Section 5.3.1. Organisational size influences the development project by transferring at least some of the management philosophy from the organisation onto the project. While project size defines the resources that the project requires to complete it in a given time-scale, team size defines the resources actually allocated to the project. In the author's experience the projects size usually exceeds the team size.

Level of Maturity

Ehrlenspiel and Dylla [47] defined a list of attributes that can be used to define the level of maturity of a design team: mature designers

- are more precise and spend more time on analysis and formulation of requirements.
- increase the scope during search for solutions - reformulation and summary.
- approach problems strategically from important to less important sub-problems.
- generate variants especially in important problem areas.
- are more accurate during and spend more time on the analysis of solutions.
- have better spatial imagination.
- apply adequate and meaningful strategies for steering the design process based on thorough analysis of demands and solution properties.

These attributes influence the required detail and structure in the planning, management and execution of the development life-cycle. It is the author's experience that mature, experienced personnel will use their experience to determine

- which parts of the design problem constitute the core of the design problem, requiring the main attention, and
- the level of "structuredness" for the design problem, dependent of the level of complexity and novelty (see 5.3.3).

To compensate for their lack of experience less mature designers will require a more structured approach. Any project may seem complex and novel, since they have very little experience to use as comparison. For this reason the possibility exists that sub-problems and the core problem are addressed with the same approach, resulting in inefficiencies and longer development times.

During their investigation with respect to the differences between novice and experienced designers, Christiaans, Dorst and Cross [28] found that increased experience resulted in an "*increased awareness of problem areas*", which probably contributed to the investigation concluding that expert designers spent more time searching for information than they novice counterparts.

Badke-Schaub [14] has investigated the influence of the designer's expertise on the success of the development project. She identified three types of "*critical situations*" in the design process, whose outcome would be influenced by the level of expertise of the designer:

- | | |
|---------------------------------|--|
| elaborations of goals | - clarifying concept or embodiment related requirements and setting goals |
| solution search | - generating proposals and solution ideas |
| solution analysis and decisions | - analysing and evaluating solution ideas, selection of solutions or solution principles |

Her finding was that during goal elaboration experienced designers would tend to "*follow their pre-established knowledge*" increasing the potential of "*insufficient goal elaboration*". In comparison in-experienced designers have very little pre-established knowledge and are usually aware of this fact. Therefore they tend to "*analyse*

goals in detail", increasing the probability of recognising new requirements. For the solution search situation, however, *"the generation of ideas depends largely on the availability of information concerning different solution principles"*. Therefore the experienced designer can draw on a larger pool of information to generate new solution principles. Similarly, the solution elaboration situation *"stand to benefit from routine expertise"*, since the *"designer is able to access his experience base with different types of knowledge"*.

In the context of the current research the findings of Badke-Schaub indicate that experienced, meticulous designers are an asset to the development project especially in the concept and embodiment phases, where the requirements have to be defined and the goals of the development established. Projects benefit more from the expertise of experienced designers during the search and analysis of solutions. Referring the project type described in Section 5.2.2, original designs benefit the most from the designer's experience since it requires the most intensive search and analysis of solutions. Selection and configuration designs can be successfully negotiated by designers with less experience, since it relies more on selecting the correct components or configuration based on the understanding of the clients requirements.

The paragraphs above have mainly focused on the maturity and experience of technical team involved in the development process. Song et al. [114] make the point that the success of product development is also influenced by the maturity, experience and involvement of senior management: important support is obtained from senior management *"by granting authority to the project manager"* and by *"defining the (development) problem strategically"*.

Maturity, capability and experience are clearly inter-related concepts in the context of this research. In the author's mind there is a subtle difference between maturity and capability, even though both of them are very dependant on experience. Maturity is related to general work experience, in the case of this research, in the engineering field. Capability in general and design capability is particular refers to the specific experience required for the specific industry, industry sector, company or project. For instance, a person with much experience in planning and managing projects may provide valuable maturity to the project team. A specialist in brake system design may have no experience in managing projects and would therefore add design capability but little maturity to the project.

Design Capability

The variable "design capability" is related to maturity and is defined by Günther [59] as a combination of epistemic and heuristic competence. The former is defined as technical knowledge, composed of theoretical knowledge (know what) and procedural knowledge (know how). The latter is defined as the capability to develop and implement strategies in new and complex situations, i.e. to determine urgency and importance of problems, to visualise the interdependency of sub-problems, and to plan and execute flexible strategies according to the given context. One would therefore expect that "design capability" increases with increasing "maturity".

Cantamessa [25] describes the capabilities of an organisation *"as a collection of decision rules and routines tied together by the firm's internal language and communication code. Such routines not only describe the organisation, but also constitute the knowledge base of the firm, thus suggesting that what the firm knows is what it does. Part of this knowledge may be "codifiable" and transmissible, whereas another part may be tacit and embedded in individuals; in the same way part of the knowledge may be formal, based on theoretical understanding, whereas another part of the knowledge may be procedural and based on experience"*. This indicates that the organisation's "design capabilities" are strongly related to the maturity or level of experience of its personnel as described above.

5.3 Framework for the Contextualisation of the Development Life-cycle

To clarify the variables discussed in the previous sections, and their interrelationships, the author has arranged them in the form of a framework, as shown in Figure 5.7. The framework is aimed at making the engineer, who has to plan, manage and implement a development project in practice, aware of the main context variables that influence the development life-cycle.

These main context variables have been discussed individually in the preceding sections. However, some of the variables are interrelated, and it is therefore important to consider these variables and the resulting



Figure 5.7: The Framework of Factors Affecting the Product Life-cycle

development methodology in a holistic manner. This is discussed in more detail below.

5.3.1 Variables Related to Size

Organisational size just indicates how many people work for the organisation which is executing the project, while project size determines how much manpower is required to execute the specific project, and the team size describes the number of people actually working on this specific project. In all three cases the required management and control increases as these sizes increase, and therefore the effect on the development methodology is clear as long as the variables are in harmony. However, when a large organisation selects a small team to fast-track a development project, the tendency might be to enforce the philosophy associated with a large organisation on to the small team, increasing the overhead burden unnecessarily. Similarly, when a team from a small company, accustomed to managing small teams and small projects, has obtained an order for a project which necessitates a large team, work is sub-contracted, suppliers are drawn into the development process, new people are employed, and the successful execution of this project requires much more managerial control and formal communication channels than the usual small project. In both these examples the project requirements in terms of size were matched with the team size selected to execute the project. It is sometimes (even often) the case that the required manpower is not available (design capacity) and employing additional people is not feasible due to the short timescales involved in the project. In this case the need for control and formal communication has to be balanced with the notion that the team members are already overburdened due to the lack of resources. In this case effective teamwork is of utmost importance. Easy access among the team members through working in the same location and frequent verbal communication will contribute to the success of the project.

5.3.2 Variables Related to Type

Another set of related context variables are product type and organisation type. As with the examples above, careful balancing of conflicting aspects of the development methodology is required when these variables do not match. For instance, a product, to be manufactured in small batches, is designed by or for an organisation which usually mass produces products. If the development is done by a consultant, the differences in production philosophy and the resulting consequences should be addressed during the development process to prepare the manufacturing organisation properly. If the development is performed in-house, besides addressing the issues mentioned above, the organisation will be unaccustomed to the suitable development process.

5.3.3 Variables Related to Complexity

Project complexity is determined to a large degree by the product complexity. The development of large complex products such as aeroplanes, mining operations or international airports require, by their very nature, complex development processes. The balance to be found here is to make the development process as simple as possible, but as complex as necessary.

Many of the prescriptive development methodologies described briefly in Section 3.1 maintain that design

problems can be dissected into sub-problems which are addressed more or less in isolation. The resulting sub-solutions can then be re-assembled to generate the total solution for the original design problem. It is therefore assumed that design problems can be structured in a hierarchical manner. Alexander [7] describes this type of structure as a tree. Natural processes and complex design problems (such as the design of a city) result in a different structure, namely a lattice structure, since most (or even all) aspects of the design problem are interrelated to most other aspects. If these structures are treated as these prescriptive processes propose, dissecting the problem into sub-problems, some of these interrelationships will be severed, resulting in optimised sub-solutions. Unfortunately the sum of all the optimal sub-solutions is far from optimal. This is referred to as partial optimisation, while the intent of design is global optimisation.

Many of the prescriptive development process models are easy to implement for simple design problems, such as small household appliances, for instance. It is the author's experience that this view does not hold for more complex products such as motor vehicles. However, even more complex products such as aeroplanes and ships, may once more require a structured, "prescriptive" approach.

In the author's opinion, the relationship between complexity of project and product on the one hand and the "structuredness" of the development process is not linear. For "simple" products the structured approach of dissection and re-assembly of the development problem is easily applied since the interrelationships between requirements of the product are relatively few and straight forward. The interrelationships between requirements of more complex products (medium complexity) are too complex to handle easily via the prescriptive structured development process. In the author's experience these type of products are developed more efficiently by employing the approach advocated by the "descriptive" models of the development process: an experienced chief designer will focus the design effort intuitively onto the central design problems first, making the peripheral problems subservient to the solutions of the central problems, without losing sight of the integration of the whole. In his career the author has come across projects where the dissection and re-assembly process of the prescriptive development processes was implemented without careful consideration for the interrelations between the sub-problems. This approach has invariably resulted in a unsatisfactory product. For highly complex projects, the amount of integration required is too much for a single person to manage. There is therefore no choice but to subdivide the whole problem into "chewable" chunks by means of the prescriptive structured approach. Obviously care will still be required in the assembly or integration of all the sub-solutions into the final solution to the development problem.

Rzevski [108] describes the threshold between problems of medium and high complexity as described above as "perceivability": *"the ability to hold all the important aspects of a problem and all their interrelationships in mind at one and the same time"*. According to Rzevski:

- *"People tend to reject, ignore or undermine the importance of systems whose purpose, aims, function or organisation they cannot perceive, and"*
- *"if people are forced to interact with an imperceivable system they are likely to commit an inordinately large number of mistakes"*.

With reference to the author's own experience in the military vehicle industry, Figure 5.8 depicts the relationship between product/project complexity and the "structuredness" of the development process as discussed above. The threshold of "perceivability" lies just beyond 4x4 vehicles, i.e. anything larger or more complex than a 4x4 vehicle is too complex to be handled in an intuitive manner by a single chief designer. The exact position of this threshold depends on a number of variables including the experience of the chief design engineer, and is therefore indicated as a band of probability in Figure 5.8. Structure in a development process provides the members of the development team an external means of visualising the interrelationships among variables, which they can no longer do in their mind. A typical example from the author's experience would be to divide the design into modules or sub-systems and to make a different designer responsible for each sub-system. In such a case, process structure is created to facilitate communication between the design teams working on the various modules, in order to ensure that the modules interface with each other and the product as a whole. It should be clear from the description above that process structure adds an overhead burden to the project, without contributing to the execution of the work directly. It can therefore be argued that unnecessary process structure adds inefficiencies to the development process. In choosing a methodology for a project, the objective is to introduce as little "structuredness" as possible, while maintaining as much as necessary.

In their article Legardeur et al. [81] discusses the differences and relationship between co-ordination and co-operation within a project context. Co-ordination is part of the "structuredness" described above: *"to plan and schedule different tasks, to distribute resources"* [81], while co-ordination is defined as *"an effective and*

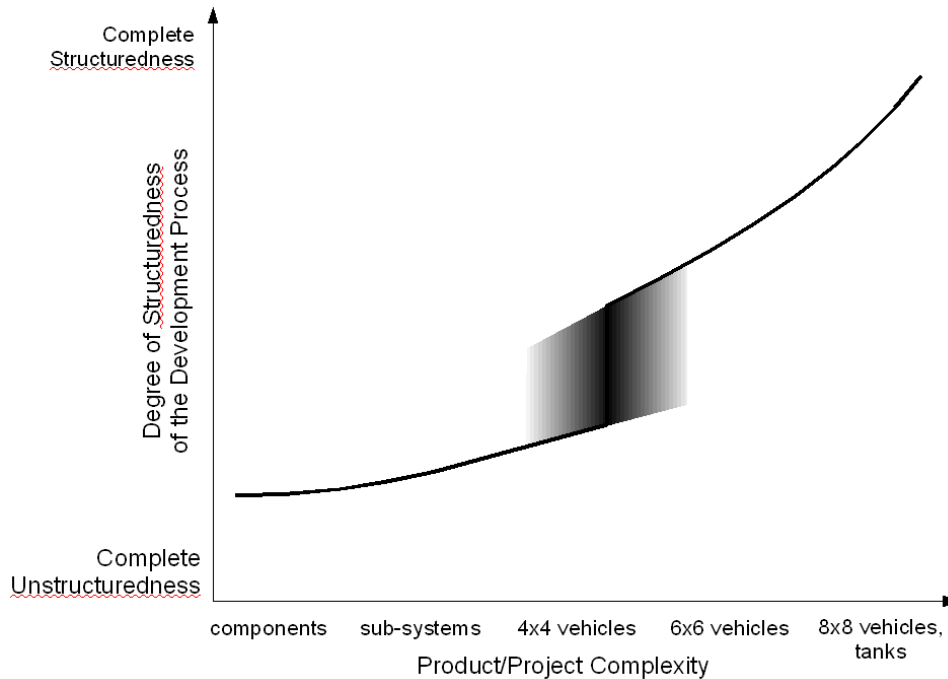


Figure 5.8: Relationship Between Complexity and Structuredness

concrete articulation among designers involved in a collective action (working in practice towards a consensus end)". Co-operation is where designers "develop their shared practice by interacting around problems, solutions and insights, and building a common store of knowledge". This is the part where the creative, innovative part of development happens. The author agrees with Legardeur et al. that unfortunately in practice "top-down co-ordination based on predefined scheduling" often "takes precedence over co-operation mechanisms among designers", to the disadvantage of the project. Therefore it is important that co-ordination is not over emphasised to the detriment of co-operation.

5.3.4 Variables Related to Maturity

Organisational and personnel maturity are not so much conflicting as they are complimentary. Mature experienced design personnel is required to obtain an organisation with a mature development process. Organisational size also plays a role: the process of a small company may seem immature due the lack of necessity for structure and formality, even though all aspects of the development process are managed diligently and all risks are addressed with care. Conversely, the development in a large company may seem mature, while it may actually be rather cumbersome and inefficient.

5.3.5 Variables Related to Capacity and Capability

The last variables that need some discussion are design capacity and design capability. Design capacity is expressed in man-hours available to execute projects and is therefore seen as an attribute of the organisation rather than the individual, i.e. personnel. Design capability is defined as the skill and competences that each individual makes available for the organisation to utilise. It is therefore considered an attribute of the personnel rather than the organisation. There are, of course, capacities of capabilities, e.g. an organisation may have employed no electrical engineers. The capacity for the specific capability to perform electrical engineering work is therefore zero.

5.4 Role players

A number of role players can be identified, who affect the contextual variables described in the sections above. On the highest level of aggregation one would expect the client and the supplier, i.e. the organisation which is contracted by the client to supply the engineering solution. Depending on the context of the project this might not be as clear-cut as it would seem. The client, although the contracting party, might not be the end-user. An example would be the Ministry of Defence in the UK, buying equipment for the Army or the Navy. On the other hand the supplier, although the contracted party might only be a middle-man and not the actual creator of the engineering solution. The middle-man could be a party that funds the contract or just manages (coordinates) the project.

The project contextual variables (Section 5.2.2), constraints and complexity, are determined by the agreement between the client and the supplier, and include issues such as timescales and price. The other project variables, i.e. size, type and novelty are determined by the problem to be solved and the proposed engineering solution. Project complexity might be increased if an external regulator is required due to inherent dangers associated with the proposed engineering solution. An example might be the development of a nuclear powerstation.

The product contextual variables (Section 5.2.3) are determined entirely by the proposed engineering solution as response to the initial problem to be solved. For the problem of electrical power supply, the contextual variables product type and product complexity differ substantially, depending on whether the power supply is intended for a single consumer or an entire nation.

Although most of the contextual variables related to the organisation (Section 5.2.1) are pre-determined given a specific project. However, large organisations can sub-divide to decrease the effective size of the organisation that is involved in and affected by the project. Similarly small organisations can supplement the size by contracting other organisations or individuals as required. The personnel variables (Section 5.2.4) can be addressed similarly. This was discussed in more detail in the sections above.

Clearly, decisions regarding the sub-division of a business, the contracting of other organisations and/or individuals involve the management of the organisation at various levels depending on the exact circumstances. The negotiation process of offering an engineering solution to a specific problem is complex and at times involves a large number of role players (stakeholders) with divergent backgrounds and interests. LU and Jing [83] have investigated this aspect of the product development process extensively for the software industry. This underscores the fact that the human interactions in all the engineering disciplines play a major role and should not be neglected.

5.5 Summary and Conclusions

Chapter 4 discussed the need to tailor or configure the development process for each development project appropriately according to the context and requirements of the project. The chapter also proposed the concept of a "project methodology" to define the approach to and the structure of the process to be followed during the deployment of the project.

What can this configuration process be based on? How does one decide what the newly configured development process should look like?

The author decided that the development process for each project would have to be determined from the project itself and its surrounding context. The preceding sections of this chapter describe the aspects of this context, which the author felt, by virtue of his experience, influenced the development process the most. Section 5.2 discussed each of the factors individually, while Section 5.3 concentrated on the interrelationships among the variables.

By making these influences and their interrelationships visible to the chief design engineer or project manager responsible for planning, managing and executing a development project, the development process can be configured to take these influences into account. This can be seen as a risk management activity, since risk management is defined as "the identification, assessment and prioritisation of risks, followed by coordinated and economical application of resources to minimise, monitor and control the probability and/or impact of unfortunate events" [67]. By aligning the project methodology with the context of the project, in the authors view, that the probability can be reduced of certain "unfortunate events" occurring.

Chapter 6 illustrates this by means of case studies, while Chapter 7 discusses the practical means of imple-

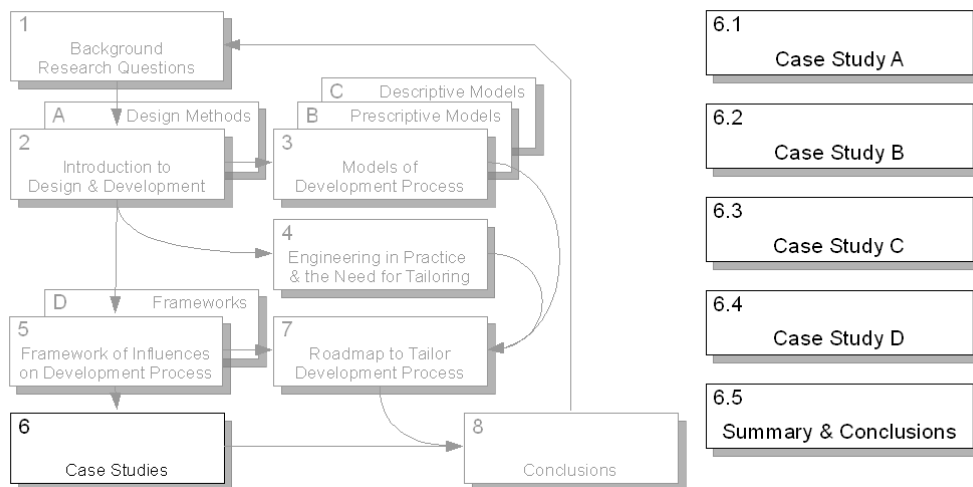
menting these concepts in daily engineering practice.

Chapter 6

Case Studies to Evaluate the Framework of Contextual Factors

Synopsis

This chapter details the context surrounding actual projects, executed in the military vehicle industry in South Africa, utilising the framework of variables that identify and quantify the factors that influence the development process the most, as described in Chapter 5. The development methodology chosen for each of these projects is described and the consequential successes and shortcomings of the projects are analysed. The fit between project context and the chosen methodology for that project is discussed for each of the case studies.



6.1 Case Study A

6.1.1 Introduction to the Industry Project and its Context

Mechanology received an order in the middle of May 2006 to refurbish a number of military vehicles for a client in Africa. These vehicles were made available by the South African National Defence Force for disposal by the material disposal department of Armscor and were consequently bought by Mechanology for the purposes of refurbishment and sale. This process was duly assessed by the National Conventional Arms Control Committee (NCACC).

Since the original engines were no longer produced and spare parts were difficult to obtain, it was decided to integrate a new commercially available engine into the vehicle as part of the refurbishment. This decision necessitated that a number of the related subsystems to be changed to suit:

- The cooling system would have to be redesigned to fit the required cooling capacity for the new engine.
- Since the original vehicle was fitted with a 24Vdc electrical system, and the commercial engines of the required size were only available in 12Vdc, changes to the electrical system would be necessary to accommodate both voltages.
- The mounting frame for the engine would have to change since the new engine would have different mounting arrangements, compared to the original engine.
- The vehicle bonnet and radiator louvre would have to change to accommodate a longer engine and a larger radiator.

Besides the re-engined vehicles, the list of deliverables included the following set of vehicle manuals:

- IPC - to be able to define spares required for maintenance and repair in the future,
- WRM - to be able to execute the required maintenance correctly, and
- OMM - to be able to train operators for the vehicles.

The project therefore consisted of two portions:

- the refurbishment of existing components, and
- the integration of a new powerpack, as well as the sub-systems, updated to fit the new powerpack configuration.

The original proposal to the client had suggested the project plan shown in Figure 6.2 for the latter (i.e. development) portion of the project. The philosophy would have been as follows:

- design the components required to install the new engine into of the vehicle,
- purchase one set of the components,
- use them to build one prototype vehicle,
- evaluate the vehicle,
- make any changes required to the design,
- purchase the components for building the rest of the vehicles, and
- produce the rest of the vehicles to complete the contract.

In order to achieve the first delivery of vehicles at the required time, the order for this project should have been placed by the middle of March. Unfortunately, the order was only placed on Mechanology in May, and, as is usual in the engineering industry, the delivery dates were expected to remain fixed as per the original proposal.

6.1.2 Application of the Framework to the Development Project

In this section the author describes the development methodology that was followed during the planning and execution of the project at Mechanology, described in Section 6.1.1, utilising the framework described in Section 5.3. Section 6.1.2 will discuss the values assigned to the variables of the framework for the particular case study under consideration. This is followed by a discussion regarding the outcome of the project, and how it was affected by the choice of development methodology in Section 6.1.3.

Values Assigned to the Variables of the Framework

Table 6.1 indicates the values for each of the variables that influence the development life-cycle, and these are discussed briefly in the following paragraphs.

Factors influencing the product development life cycle		Values for the various factors for Mechanology in the context of this project
Organisational Factors	Organisational Size	Small - 11 personnel in total
	Organisational Structure	Very flat management structure, no exterior constraints
	Type of Organisation	
	<i>Purpose</i>	Design and development with small batch production facilities
	<i>Production Size</i>	Small batches, usually less than 20.
	<i>Stock philosophy</i>	Engineer-to-order
	<i>Selling mechanism</i>	Competing through tender system
	Organisational Maturity	High (flexible)
Available Design Capacity	1 design engineer, 1 design technician	
Project Factors	Project Size	To make the timescales 4 people would have been required
	Project Complexity	Low
	<i>Complicacy</i>	Low
	<i>Dynamics</i>	Medium
	<i>Opacity</i>	Low
	<i>Interdependency</i>	Medium
	<i>Polytelie</i>	Low
	Project Type	Configuration design / redesign
	Project Constraints	Project duration
Project Level of Novelty	Low	
Product Factors	Product Type	Batch production
	Product Complexity	Low
	<i>Complicacy</i>	Low
	<i>Opacity</i>	Low
	<i>Interdependency</i>	Medium
	<i>Polytelie</i>	Low
Product Position in System Hierarchy	Machine	
Personnel Factors	Team Size	Small - 2 people
	Level of Maturity	High
	Level of Design Capability	High

Table 6.1: Framework Values for Case Study Project

The values assigned to the variables of the framework above are assigned within the context of the South African military vehicle industry in general and the experience of the personnel involved in the project in particular. As an organisation Mechanology was a small company with an extremely flat management structure, owned completely by the CEO. Therefore there was no external constraint on how the projects were to be planned, managed or executed. Internally the engineering staff (2 engineers and 2 technicians at the time)

reported directly to the managing director, and project responsibilities were assigned on a matrix structure. This implied that an engineer would lead one project (his colleagues are reporting to him), while he was concurrently reporting to one of his colleagues on work he was performing on another project. Due to this internal communication was mainly informal and based on the close relationships among the staff. The company's purpose was the design and development of mechanical systems, specialising in military vehicles and turrets. In support of the design and development function a small workshop for building and evaluating prototypes was available. The niche market that Mechanology was addressing, required products in small batches (20 items or less), engineered-to-order. Most orders were obtained through a tender process. Since Mechanology was active in the design and development of military equipment a system was in place to generate and manage (under configuration control) the required documentation. This included inputs to the design process, documents generated during the design process as well as deliverables, such as manuals and catalogues. The system was flexible to enable the scaling up and down to support the differences in requirements from large, complex projects such as the design of complete vehicle systems and turrets, to small jobs such as the design of single components. Therefore organisational maturity was considered to be somewhere between level 3 and 4, as shown in Figure 5.2: although the process of product development was defined on a organisation wide level, specific approaches, philosophies and practices were determined on a project level for each project.

The development part of the project was worth approximately R 1,200,000.00, and could have been executed by an engineer and a technician in the timescales as originally planned. To meet the timescales required by the customer, 4 people would have been required. Unfortunately only 2 people (the author and a technician) were available for the design part of the project, since the other engineer was managing the refurbishment of the vehicles in parallel. The 2nd technician was busy with another project and could not be utilised. It is therefore clear that the project constraint was definitely time (for the resources available). Although sub-contracting was considered, it was decided that the design problem could not be split without generating more work to manage the subcontractor and the interface between the two packages of work than it would have taken to execute the whole job in-house. The design work to be executed placed the project in the configuration design / redesign category - the philosophy was to take as many commercially off-the-shelf (COTS) components as possible and integrate them into a powerpack for the existing vehicle. All the interfaces among the COTS components themselves and the original vehicle would have to be designed. The project complexity was deemed low, for the following reasons:

- Since the project was based on just two objectives, i.e. the refurbishment of the vehicle and the installation of a new powerpack, the level of complicacy for the project was considered low.
- Dynamics was considered to be medium. Although the project requirements were not expected to change during the execution period of the project, timing the various project tasks in order to achieve the required timescales would be both important and quite a challenge.
- Opacity was also considered to be low since Mechanology had previously refurbished and installed a new engine in a similar vehicle. Therefore all project variables were known and visible.
- Interdependency was deemed medium since many of the project tasks were dependent on the timely and successful completion of previous tasks.
- "Polytelie" was also considered low since the main objective was to design and integrate the powerpack inside the required timescales. Given the timescales and the available resources it would be difficult to exceed the budget.

Project novelty was considered low since a similar project, to install a diesel engine into a variant of the vehicle under consideration, had been successfully executed about six months earlier. It was intended to transfer the lessons learnt from the previous project onto the current project. Also both members of the project team had fitted new engines to existing vehicles before.

The product being designed definitely fell into the batch production category.

The product complexity was considered low for the following reasons:

- Since the project scope was limited to a small part of the system hierarchy, i.e. the design and installation of a sub-system, in this case the powerpack, in comparison to developing a complete weapon system, including vehicle and turret, complicacy for this product was deemed low.

- Opacity was also considered to be low since Mechanology had previously installed a new engine in the similar vehicle. Therefore all design parameters were known.
- Interdependency was deemed medium since it was decided to construct a slide mounted powerpack, i.e. the engine with all the auxiliaries as well as the cooling system, including the cooling fan and its drive system were to be mounted on a frame that could slide in and out of the engine bay. This required the integration of these various components and sub-system into a compact unit. Although the interdependency within this unit was considerable, the interface with the rest of the vehicle was prescribed (by existing hardware) and fairly straight-forward.
- "Polytelie" for the product was also considered low. Since the design objective was limited to integrate various components to form the powerpack, and mounting the powerpack, in turn, into the vehicle, compared to the development of a complete vehicle where conflicting variables, such as weight, payload, mobility, size, fuel range and level of protection must be addressed simultaneously.

The team selected for the execution of the design and development part of the project was small, i.e. one engineer and one technician. However, the level of maturity and design capability were considered high, since both team members had been working in the industry for the design of military vehicles for many years and had executed similar projects in the past.

The Resulting Project Methodology

The project, the scope of work, the available resources and the prevailing constraints were discussed internally at Mechanology and it was decided to accept the order for the execution of the project at the timescales dictated by the client. Although the project always had two parts to it, i.e. the refurbishment and the development of the new powerpack, it was decided as a consequence of the extremely short timescales to assign each part to a different engineer to manage. The author was responsible for the development part, while his colleague managed the refurbishment. Therefore the project plan for the design and development portion of the project had to be adjusted as shown in Figure 6.3. It was also decided to draw prospective suppliers into the development process as early as possible by providing a list of components required, the quantities needed and the necessary lead-times, so that they could include this project in their plans.

The development methodology adopted for this project is shown in Figure 6.1. Compared to the conventional product life-cycle, shown in Figure 2.5, and the development process, shown in Figure B.7, the need analysis and concept design phases were excluded from the project plan for this project. The needs analysis had been completed during the quotation phase of the project, in order to finalise the scope of work for costing purposes. The concept design phase was deemed to have been completed given that a different diesel engine had been fitted to a similar vehicle approximately six months before. It was therefore assumed that the concepts could be transferred from the previous project to the current project. It was, however, decided to improve some of the concepts to eliminate problems encountered in the previous project.

The following paragraphs discuss the project as a whole with reference to the various phases, while a detailed description of each phase is provided under separate headings below. The phases embodiment, detail design, prototype build and prototype testing had to be linked sequentially, since each successive process is dependent on the previous process for its input. For a larger team it might have been possible to have more overlap for these phases, since detail drawings for a portion of the design could have been generated while embodiment design was still being completed on another portion. Due to the size of the design team, the technician documenting the embodiment design on 3D CAD, was also generating detail drawings. It was, however, possible to overlap the phases mentioned above with the industrialisation phase and the production phase, as far as the purchasing activities are concerned for these two phases.

It was decided to keep internal communication as simple and as informal as possible, which was acceptable due to the small team and lack of subcontractors for development work. Formal, structured correspondence was only required when communicating with suppliers and the customer. Electronic data, including correspondence, project plans, design calculations, drawings and manuals, would be stored on the local network server, using the standard directory structure. It was considered to change the directory structure to represent a virtual product life-cycle, but it was decided that access time to documentation would be faster if a familiar structure was used.

Although the telephone would be used extensively to communicate with suppliers as well as the client in a efficient manner, emails and faxes would be used to record the pertinent points of important decisions. A

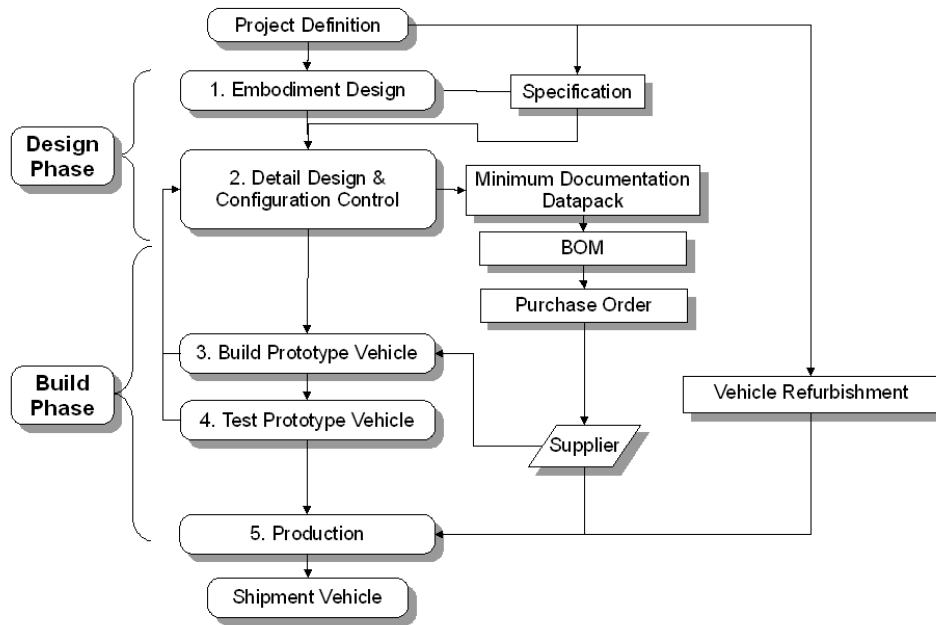


Figure 6.1: The Development Process Adopted by Mechanology for this Project

system of delivery notes would be used to manage the collection and delivery of parts sent to suppliers for repair or rework. In order to keep track of the project's expenses a formal purchasing system would be used to order material for the design as well as the refurbishment portion of the job.

The output of the design phase (besides a prototype vehicle) was decided to be a minimum documentation datapack. This meant that only drawings were to be generated that were essential for the manufacture and assembly of powerpack components, i.e. no drawings would be generated for purchased COTS items such as the engine, the alternator and the vacuum pump. 3D CAD information on these items would, however, be necessary in order to be able to design the parts that interface with these components, such as the mounting bracket for the alternator and the mounting frame for the engine. Also pictures from the CAD system would be used in the compilation of the IPC and WRM manuals.

The scope of work for each of the project phases as shown in Figure 6.3 is discussed in the paragraphs below in more detail.

Concept and Embodiment Phases

As discussed before, the concept phase was dropped as a formal project phase, since a similar project had been completed during the previous six months. However, at the beginning of the project, in this case the embodiment phase, the concept being transferred from the previous project was reviewed in order

- to determine the differences in the projects,
- to define any weaknesses in the previous design and suggest improvements, and
- to define the scope of work for the rest of the development portion of the project.

The embodiment design phase was dedicated to defining and functionally integrating all required COTS components into the powerpack and the vehicle, by generating 3D models on CAD. This meant not just the modelling of bought-in items, but also positioning them in such a manner that they could perform their function as intended. The design of brackets and other means to mount the COTS components was also part of this phase. At the end of the embodiment phase the intended scope of supply (a rough description and the quantities required) was able to be defined and the required lead-times were discussed with the intended suppliers.

Detail Design Phase

The detail design phase involved the generating of detail drawings of all the components that had to be manufactured for this project. Definition of materials, surface finish and surface treatment were finalised, in collaboration with suppliers.

Prototype Build Phase

During the prototype build phase, the powerpack was assembled and fitted to the first vehicle. The supplier of the harnesses for the electrical system fitted the prototype harness, finalising cable lengths in the process. Piping for fuel system, tyre inflation system and brake system were also finalised and documented.

Prototype Test Phase

Evaluation of the prototype vehicle was concerned with testing the integrity of the sub-systems that were changed on the vehicle, such as cooling, fuel supply, electrical and tyre inflation. Automotive performance was only of secondary concern since the new engine was of the same capacity as the original engine. Reliability of the new sub-systems was tested by driving the vehicle a couple of hundred kilometres. The cooling system, however, had to be assessed by means of temperature measurements. The engine and cooling system were instrumented and the data was recorded, using an electronic data logger, for analysis.

Industrialisation Phase

The industrialisation phase overlapped with the detail design phase, since lists of all COTS items with their respective supplier contact details were drawn up during the detail design phase, even though it is essentially an activity belonging to the industrialisation phase. Orders for the long lead items, such as engines, were placed in parallel with the embodiment phase already, in order to have them available for the building of the prototype vehicle. Orders for the rest of the items were placed as soon as the definitions (detail drawings) were complete. This included components for the spares package, which had to be delivered to the client with the vehicles. Tooling requirements for the production, as well as for support of the maintenance at the client, were defined.

Production Phase

The rest of the powerpacks were assembled and installed into the vehicles during the production phase. The refurbished vehicles were considered inputs to this phase of the project. Once the powerpack was installed, and the turret mounted. The vehicle was then to be painted in the colours required by the customer.

6.1.3 The Execution of the Project

Section 6.1.1 described the general context surrounding the project, while Section 6.1.2 utilised the context variables to describe specific details that had an influence on the methodology (discussed in Section 6.1.2) chosen to execute the project. This section discusses the execution of the project as well as the resulting successes and shortcomings.

Successes

On a technical level the project was successful: the vehicles passed the functional and cosmetic factory acceptance tests and they were delivered to the harbour for shipment to the end user. Functional testing included the testing of the cooling system of the prototype vehicle by running the vehicle at full power on the high-speed track at Gerotek while recording the temperatures at all the in- and outlets of the various components of the cooling system. Comparing the measured values with the maximum allowable temperatures for the cooling water and engine oil for the specific engine, and taking the ambient temperature into account, it was established that the cooling system would be functioning adequately in the worst ambient conditions

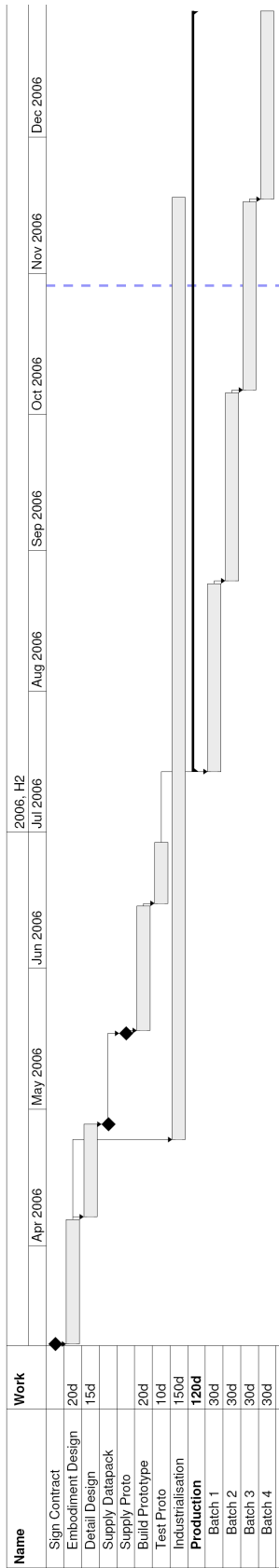


Figure 6.2: The Original Project Plan

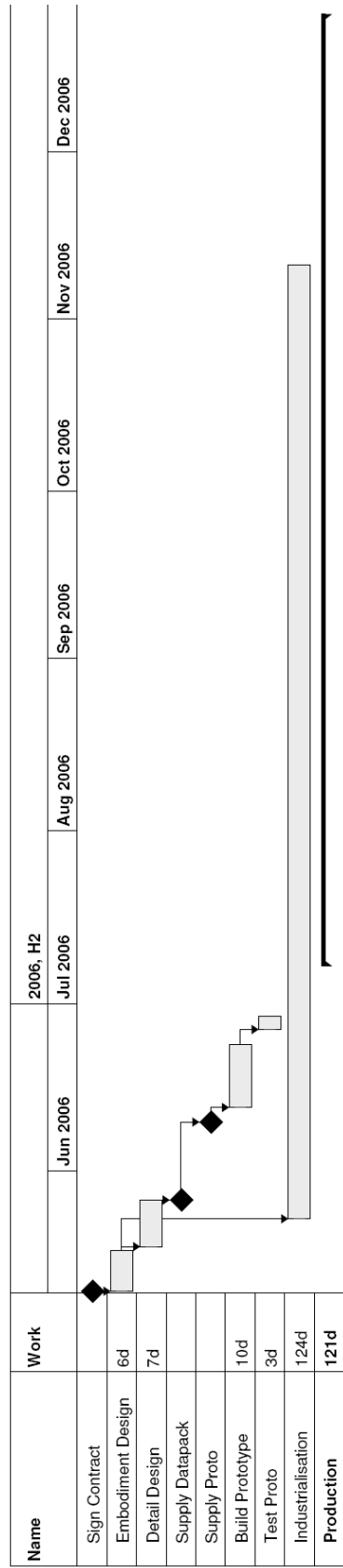


Figure 6.3: The Project Plan Used for the Execution of the Project

expected in the country of the end user. Reliability was confirmed by driving the vehicle for 500 km, while monitoring the condition of the powerpack and its components.

Another positive aspect was the effective internal and external communication:

- Management was informed on a weekly basis of the progress, risks, cost to date and expected cost to completion.
- Finance department were kept up to date regularly on orders placed, invoices received as well as cost to date and expected cost to completion.
- The interface between the refurbishment part and the development part of the project was discussed every second day or so.
- The information required for the generation of the manuals was being captured as it became available.
- Progress on manufactured components at the suppliers was monitored on a daily basis.
- The client was informed on a weekly basis of the progress of the project and any difficulties identified.

Shortcomings

The Delivery of Custom Manufactured Parts

In terms of adherence to the schedule arranged with the client, it was fortunate for the project that the shipping date was postponed by a week due to the late arrival of the ship at the harbour. The biggest problem encountered was the delay in receiving components from suppliers. Most of the COTS components, such as standard air filters, rubber hoses, cooling fans, bearings and pulleys were delivered on time. Delivery problems were encountered with all the components that were specifically designed and manufactured for this project. These included the powerpack frame, the radiator, stainless steel pipes for the cooling system, and all the brackets for mounting components such as alternators. The author and his colleague worked closely with the manufacturers to facilitate the process, ensuring that they had all the information they required and that they understood the requirements and the timescales. Delays were caused by raw material suppliers not supplying the raw material as promised, and by the manufacturers overestimating their own manufacturing capacity, even though the author had attempted to verify the capacity by discussing the scope of work and the timescales with all the manufacturers before placing orders. In one instance it was possible to split the order for components between two manufacturers, but generally it was decided not to switch suppliers once the order was placed. It was argued that it would have taken longer for the second supplier to obtain material and slot this new work into his schedule than it would take the first supplier to finish the job.

These delays had a major impact on the process of refurbishing and assembling the vehicles. The plan had been to complete the mechanical refurbishment of the vehicle, while the powerpack was being assembled. The powerpack would be installed and the refurbishment of the vehicle would be completed by spray painting the vehicle in the colour specified by the end user. The delay in powerpack assembly necessitated that the vehicle was spray painted before the powerpack was fitted, resulting in the need to patch-up the paint work afterwards.

The Transfer of Knowledge from the Previous Project

The second problem encountered was caused by the approach adopted for this project, which was reliant of the work done and understanding generated during a previous project. The design for this project was based on the concept work performed during a previous project six months earlier, where a similar vehicle was also fitted with a new engine in a powerpack arrangement. Unfortunately, neither the author nor his colleagues working on this project had been involved in the previous project, and therefore they had no first-hand knowledge of the design.

Although the previous project was discussed with the colleagues who were involved with it, only general information could be obtained, and therefore, in terms of detailed information, the author could depend only on the information that was stored on the company network server. Since the previous project was considered to have been successfully completed, the CAD data was transferred to the new development. The extremely short timescales and the lack of resources prevented the quality of the data to be determined in detail. Some of the errors in the CAD data were discovered during the embodiment phase and could be corrected without any significant time or cost implication. Others, however, were only discovered during the assembly of the

prototype vehicle when components would not fit as expected. These errors, obviously, had major time and cost implications, and the corresponding re-work contributed to the capacity shortage experienced by the manufacturers.

Summary

This project achieved what it was meant to achieve: the refurbishment of the specific military vehicles, installation of a new powerpack, including the associated sub-systems, and delivery of the complete vehicle to the harbour for delivery to the end user within budget and time constraints of the project. It did, however, experience two shortcomings: the difficulty of obtaining the custom designed parts on time, and problem of transferring tacit knowledge across projects.

A more formal implementation of the "standard" "prescriptive" model of the development process as, for instance, advocated by the VDI guideline [125], would not have shortened the time span between the start of the project and the receipt of the designed parts. No other approach to the project would have prevented this problem from occurring; indicating that not all project related problems can be solved by the choice of the correct methodology. The experience of this particular project has confirmed to the author that the timescales of the design portion of the project can be manipulated fairly easily. For the building and testing of the prototype, timescales are much more rigid: beyond a certain threshold the reduction of timescales for the manipulation of physical material requires an exponentially increasing effort.

In retrospect it is clear that the latter of the two problems experienced during this project, was caused by underestimating the level of project novelty. The reliance on data from a previous project, without any of the role players of this project having been involved in the previous project, should have warranted a stage in the process of the project to check and verify the data.

To illustrate how the approach to the project may have differed under other circumstances the following can be considered. A larger organisation would probably have added resources to the project. Although this would possibly have lessened the time pressure of the project, it would have probably have required a more formal and structured communication structure. Had the product been intended for mass production, more attention would have been given to the ease of assembly, as well as the tooling requirements (jig and fixtures, cranes and pneumatic power tools for instance). If the project had been more complex, it would not have been possible to execute it successfully in the same timescales with the same resources. Depending on the type of complexity the need for extensive sub-contracting could have been required. This would have increased the management overhead of the project.

6.1.4 Conclusion and Discussion

A development project in industry was introduced in Section 6.1.1, and the variables described in general terms in Chapter 5 were discussed with respect to the specific project and its context in Section 6.1.2. Section 6.1.3 described the execution of the project by the author and his colleagues as part of the normal business processes of the company called, Mechanology.

It was shown that the "standard" development process was adapted to suit the specific context surrounding this specific project. In this case the development process could be streamlined by eliminating the needs analysis and concept stages of the process. Also, in order to achieve the required timescales with the extremely limited resources, the industrialisation phase and the detail design phase were conducted in parallel as shown in Figure 6.3. For the design part of the project the "standard" problem solving procedure as proposed by "prescriptive" development process models was not followed. Rather than dissecting the design problem into sub-problems (refer to Figure 3.1), which are evaluated individually by analysis of their functional requirements, followed by the generation of a series of potential solutions of which the best one is chosen, it was decided to generate solutions based on presuppositions, focussing initially on the central problem, and addressing all peripheral problems accordingly, as per the "descriptive" models of the development process, as described in Section 3.3. The only sensible division, that could be implemented on this project, was to split the design portion from the "production" part of the project into different phases. Due to the extreme timescales the phases were intended to overlap.

This adaptation of the development process made it possible to successfully complete this project within extremely short timescales with very limited resources.

6.2 Case Study B

6.2.1 Introduction to the Industry Project and its Context

Introduction to BAE Land Systems OMC

BAE Land Systems OMC is part of the international group of companies called BAE Systems. According to the BAE Systems' website [4], they are the "3rd largest global defence company" with "96,000 highly skilled people", "customers in over 100 countries", "annual sales exceed £15 billion" and "annual R&D exceeds £1.3 billion". Businesses within BAE Systems worldwide include:

- | | | |
|--|---|---|
| <i>BAE Systems Australia</i> | - | <i>is a leading supplier of communications, electronic warfare systems, military air support, air defence, mission support systems and intelligence, surveillance and reconnaissance to the Australian Defence Force.</i> |
| <i>CS&S International</i> | - | <i>is responsible for the development of new market opportunities and the sustainment of business in the Middle East region including a long-term presence in Saudi Arabia.</i> |
| <i>Electronic and Integrated Solutions</i> | - | <i>designs, develops and manufactures a wide range of electronic systems and subsystems for both military and commercial applications.</i> |
| <i>Integrated System Technologies</i> | - | <i>is fully equipped to meet the demands of the rapidly evolving modern and future markets in defence, homeland security and complex, mission critical solutions.</i> |
| <i>Land and Armaments</i> | - | <i>is a global leader in the design, development, production, and service support of armoured combat vehicles, major and minor calibre naval guns and missile launchers, canisters, artillery systems, and intelligent munitions.</i> |
| <i>Military Air Solutions</i> | - | <i>focuses on military air solutions to meet the through-life needs of customers.</i> |
| <i>Regional Aircraft</i> | - | <i>is today a leading provider of regional aircraft and support services to regional airlines throughout the world.</i> |
| <i>Submarine Solutions</i> | - | <i>deliver high quality submarines and large naval ships, demonstrating value for money over a full range of contracts, system integration, design and build services.</i> |
| <i>Surface Fleet Solutions</i> | - | <i>is a business in transformation to become the through life surface warship capability partner of choice.</i> |
| <i>Underwater Systems</i> | - | <i>provides the capability to achieve full tactical advantage and dominance in the underwater battlespace.</i> |

BAE Land Systems OMC in South Africa is part of the Land and Armaments business, and specialises in high mobility wheeled armoured vehicles with high levels of protection against ballistic and landmine threats. According to the BAE Land Systems OMC website[3] the business "*is South Africa's primary military vehicle facility, covering all disciplines of the military vehicle spectrum from conceptualisation through design and development, manufacture and production and in-service support*".

The variety of companies that make up BAE Systems, and the fact that they are scattered all over the world, has prompted BAE System to introduce a common procedure to manage the life-cycle of its products uniformly in a formal manner throughout the entire product range. This process is divided into two phases: the business winning and the contract execution phase. As such the process is not specifically aimed at the development of new products or technologies which do not have a definite customer yet. During the process a number of phase and design reviews are conducted to determine the current status of a project and to define the strategy for the next phase. Design reviews concentrate on the technical aspects, the maturity of the product, the technical risks and how well the technical requirements of the customer are met. Phase reviews, on the other hand, concentrate mainly the financial and schedule aspects of the project, taking the technical status into consideration, as well as interactions and communications with the customer.

Introduction to the Development Project

The project was started on the 27th of September 2007. The purpose of this project was to develop and build two almost identical concept demonstrator vehicles. Due to project constraints imposed by the client it was intended to deliver the first vehicle to the client on the 12th of December 2007 after only the absolute minimum of testing and evaluation. The purpose of the second vehicle was to perform in-house testing and evaluation, in order to verify that the functional requirements were met. It was envisioned that the customer would evaluate the concept demonstrator vehicle from January to March 2008, and would then place an order on BAE Land Systems OMC for production in April or May 2008.

The Company Environment at the Time of The Project

At the time of this development project the company was in the process of executing a very large production order for an international client. The client had imposed a very tight delivery schedule, requiring the company to increase the production rate from the typical two vehicle per week to six vehicles per day, i.e. a six-fold increase. Unfortunately, this vehicle had also been developed in very short timescales, necessitating the engineering (development) department to be intensively involved to fine-tune the design and implement design improvements on the production line.

The Technical Requirements

Intelligence from the client indicated that the client required a vehicle with a larger payload (both in terms of volume and mass), and a more powerful engine than the vehicle in production, as described above. The increase of payload by 50% brought the gross vehicle weight of the fully laden vehicle into the abnormal category, making it illegal to travel the roads of South Africa in the laden condition without a special permit. Besides a new hull to increase the under-armor volume of the vehicle, these requirements necessitated the following sub-systems to be changed:

- Powerpack: - more powerful engine, larger cooling pack, larger transmission
- Driveline: - larger transfer gearbox, axles and propshafts
- Running Gear: - larger tires, uprated steering, braking and suspension systems

At the same time it was decided to fit the concept demonstrator vehicles with a digital electrical system rather than the conventional analogue equivalent. These vehicles were the first vehicles to be fitted with such a system at BAE Land Systems OMC.

Although sold to the client as a redesigned, upgraded version of the production model discussed above, the fact that each functional sub-system of the vehicle changed, indicates that this project entailed the development of a new vehicle rather than the redesign of an existing product. The high gross vehicle weight and the digital electrical system introduced a high level of novelty to the project.

Besides sourcing the components for the above mentioned sub-systems, before the vehicle could be designed, most of the suppliers of these sub-systems were also supplying components to the production line. In some instances the supplier did not have enough capacity to produce components for production as well as the once-off components for the prototype.

The Design Process

The production process of an armoured vehicle is divided into fabrication (manufacture of the armoured hull of the vehicle and all unique components that have to be produced by a combination of welding and / or machining) and assembly (the fitting of all components on the hull of the vehicle, including both bought-out and manufactured components). Given the extremely short timescales of the project it was decided not to generate a complete datapack for the building of the first two vehicles. The initial "to-build" datapack would mainly contain component drawings (to be either manufactured or purchased), with very few assembly and weld-assembly drawings. This would mean that design personnel would have to participate and support the fabrication and assembly process.

During the process of building the two concept demonstrator vehicles, the fabrication and assembly information, as well as any changes required to the fabricated components, would be captured by hand, to be incorporated in a "as-built" datapack. Any improvements highlighted through the testing of the concept demonstrator vehicles and changes to the design requested by the customer would also be incorporated into this datapack, which would be verified through the building of a prototype vehicle. Information from the building process of the prototype vehicle would be incorporated into a "to-build" production datapack.

Although it was suggested by the author to divorce this development project completely from the normal (design, purchasing, fabrication and assembly) processes of the company by implementing the "skunk works" process [89], it was decided by management to follow the formal project structures for this project. This process entails the following:

- the engineering department generates the drawing(s) to define a component (to be purchased or manufactured) and books the drawing into the configuration department,
- the configuration department records the component on the MRI and releases the drawing(s) to the material handling department,
- the materials handling department loads the information regarding the component on a BOM and releases the drawing(s) to the purchasing department,
- the purchasing department places orders on suppliers for the components as defined on the drawing(s) and attempts to expedite the delivery process.

This was done for two reasons:

- Management wanted to retain "control" of the development process, and
- It would be easier to generate the "as-built" datapack, if the "to-build" datapack is already under configuration control. This was seen as critical since the production order was expected very quickly after completion of the concept demonstrator vehicles.

The development of this vehicle was led and managed by an engineering manager, focussing on the technical aspects as well as the timely execution of the project. Working on the project were two design engineers, responsible for specific sub-systems, and on average three CAD operators. Due to the high priority of the production project, mentioned at the beginning of this section, as well as other development projects, the same CAD operators were not always available for the duration of the project. The author was responsible for the running gear sub-system, i.e. brakes, steering, suspension and wheels, while a colleague was responsible for the powertrain, i.e. powerpack (cooling pack, engine and transmission packaged together as a module), transfer gearbox, propshafts and axles. All other sub-systems had to be managed by the engineering manager himself.

The Build Process

Once the armour plates had arrived, the fabrication process could start, by placing them into a jig and welding them together to form the hull of the vehicle. Any other weld assemblies, for instance brackets for mounting components such as suspension, powerpack or transfer gearbox were also fabricated at this time. Once complete, the hull and all other components were delivered to the "proto area" where the vehicle was assembled. In the "proto area" sub-assemblies were put together, before mounting them on the vehicle. Piping, hoses and cabling are very difficult to design on CAD accurately enough to fit in the vehicle the first time. Therefore it has always been the practice at BAE Land Systems OMC to reverse engineer these components on the first vehicle and record the design manually.

As mentioned above, the design engineering staff were intensively involved with these processes, since it was decided to generate a minimum of fabrication and assembly documentation.

The Evaluation Process

The second concept demonstrator vehicle was planned to undergo a two-stage evaluation program:

- The critical automotive functions were to be evaluated first: suspension, handling characteristics, brakes, steering and engine cooling.
- Secondly the vehicle was to be evaluated against each line of the client's specification.

Errors or problems with the critical functions were to be corrected on the concept demonstrator vehicle immediately, while recording and correcting less critical problems in the next datapack and verifying the solutions in the next vehicle.

6.2.2 Values Assigned to the Variables of the Framework

Table 6.2 indicates the values for each of the variables that influenced the development life-cycle, and these are discussed briefly in the following paragraphs.

Factors influencing the product development life cycle		Values for the various factors for this project in its context within the environment of BAE Land Systems OMC
Organisational Factors	Organisational Size	Large – in excess of 800 personnel
	Organisational Structure	Very steep management hierarchy with many levels
	Type of Organisation	
	<i>Purpose</i>	Production facility with development department
	<i>Production Size</i>	Large batches, usually 100 vehicles or more
	<i>Stock philosophy</i>	Engineer-to-order
	<i>Selling mechanism</i>	Competing through tender system
	Organisational Maturity	Medium (between levels 3 and 4)
	Available Design Capacity	Total: 13 design engineers, 25 CAD operators Available: 2 design engineers, 3 CAD operators
Project Factors	Project Size	To make the timescales 4 engineers and 6 CAD operators would have been required
	Project Complexity	Medium
	<i>Complicacy</i>	Medium
	<i>Dynamics</i>	Medium
	<i>Opacity</i>	Low
	<i>Interdependency</i>	High
	<i>Polytelie</i>	Low
	Project Type	Redesign
	Project Constraints	Project duration
	Project Level of Novelty	Medium
Product Factors	Product Type	Batch production
	Product Complexity	High
	<i>Complicacy</i>	High
	<i>Opacity</i>	High
	<i>Interdependency</i>	High
	<i>Polytelie</i>	Medium
	Product Position in System Hierarchy	Machine – level 6
Personnel Factors	Team Size	Medium
	Level of Maturity	High
	Level of Design Capability	High

Table 6.2: Framework Values for Case Study Project

The values assigned to the variables of the framework above are assigned within the context of the South African military vehicle industry in general and the experience of the personnel involved in the project in particular.

Organisational Factors

BAE Land Systems OMC, as the South African part of the BAE Systems conglomerate, has always been a large, highly structured company with a rigid hierarchy. Incorporation into the BAE Systems group of compa-

nies has just increased the rigidity of the hierarchy structure of the company. The company philosophy is one of division of labour into respective departments with strict managerial control.

The purpose of BAE Land Systems OMC is to manufacture vehicles, and the function of the engineering department is to feed the factory with definitions of vehicles to produce. Very little emphasis is therefore placed on product innovation and the development of technical know-how, activities commonly labelled as R&D.

BAE Land Systems OMC competes internationally through the tender system, producing vehicles on an engineered-to-order basis, usually in batches of 100 vehicles or more. However, batches as small as three or six vehicles have been produced. These small batches are strictly manufactured-to-order, based on an existing product definition. Orders for larger quantities of vehicles will usually include some engineer-to-order aspects.

The organisational maturity of the company is estimated between level 3 and 4, as discussed in Table 5.2 in Section 5.2.1. Organisation wide procedures and support exists for project management and documentation, while procedures and support for selection, implementation and use of computer tools in engineering is either individually implemented or on a project-by-project basis. These issues related to organisational maturity are strongly related BAE Land Systems OMC' focus on the production of vehicles rather than the development of products, as discussed above.

Although BAE Land Systems OMC has a fairly large engineering office, employing approximately 50 people including section heads and managers, these resources were allocated to a number of projects at the time of the development project described in this case study. Most of the resources were allocated to the current production of vehicles discussed in 6.2.1. Four design engineers were allocated to this project, of which only one, the engineering manager for the project, could dedicate 100% of his time to the project. The three others, including the author could only make about two thirds of their time available. The number of CAD resources allocated to this project varied from two to five, with a average of three CAD operators for the duration of this project.

Project Factors

The most important project factor was the supposition that the concept demonstrator vehicles had to be completed in extremely short timescales. Therefore project duration became the project constraint with the greatest influence on the project. In order to achieve the timescales one of the prerequisites would have been the full-time allocation of the four engineers and six CAD operators to this project.

Project complexity was estimated to be medium. Client involvement would have increased the project complexity substantially. Unfortunately many people of different departments were involved, adding little value to the development process since it was decided to utilise the company's processes (as described in Section 6.2.1) even though these processes are aimed at supporting the production process rather than development projects with extremely tight timescales. This increased the number of project variables to be managed, and therefore the complicacy of the project was estimated to be medium.

Due to the tight timescales of the project, project dynamics was extremely important: technical issues had to be addressed out of sequence in order to define and order components with long lead times, such as engine, transmission, radiator, axles. Often decisions had to be made in terms the definition of these components, without being able to ascertain the complete impact of these decisions on the rest of the design. It was for this reason that the project dynamics was estimated to be medium for this project.

The complicacy of project variables, as discussed above, combined with the critical dynamics of the project, created a high interdependency of project variables for this project: a complicated, involved process had to be managed under tight timescales.

Both opacity and Polytelie were estimated to be low for this project. Once the decision had been made regarding the development process to be followed for this project, all project variables were clearly defined and visible. Due to the time pressure on the project, the primary objective was to define and build the concept demonstrator vehicles as quickly as humanly possible with the constraints of the company and the chosen development process. All other project objectives, including costs were deemed to be secondary.

The level of novelty for this project was estimated to be medium. The development of a high mobility 4x4 armoured vehicle within normal parameters would have been considered a project of low novelty. The requirement for payload increased the gross vehicle mass of the vehicle far above normal values, consequently increasing the novelty factor for this project. The influence of the high vehicle mass on all major automotive

sub-systems as well as the performance of the complete vehicle, could only be estimated by extrapolation, and could only be determined with accuracy during the evaluation of the vehicle. This introduces an increased level of risk to the project. The implementation of a digital electrical system rather than the conventional analogue equivalent, added novelty and risk to the project.

Product Factors

The vehicles would be manufactured in batch production. Given the large batch size and the high production rate required for the current production of vehicles, it would be reasonable to assume that the vehicle under development for this project would have similar production requirements. It would therefore have been reasonable to consider aspects of mass production in the design of the vehicle, to assist with the achievements of this high production rate. Unfortunately, the level of available resources combined with the tight timescales made it impossible to consider these aspects during the design phase.

Product complexity was estimated to be high for this product for the following reasons: a military vehicle has a large number of interdependent and often conflicting user requirements to be met, resulting in a large number of interdependent product variables, and therefore high complicacy and interdependency. Since the design work was allocated to design engineers according to the sub-systems of the vehicle, without providing a means of taking the vehicle as a whole into account, the product variables falling outside the scope of their sub-system became invisible for each these design engineers. Therefore the opacity of the product variables was considered high. Polytelic for this product was considered medium, since the main objective was to design a vehicle that could carry the prescribed gross vehicle mass. Secondary objectives included the achievement of the functional requirements for all the sub-systems under this constraint.

With regards to the level of the system hierarchy, the concept demonstrator vehicles were of the level of a machine according to Günther [59] or level six according to Frost [53], as indicated in Figure 5.4 in Section 5.2.3. The vehicle was developed with the emphasis on automotive performance and payload, both in terms of mass and volume. Integration of client furnished items (CFE's), typically roof mounted gunrings or turrets, personal weapons and communication equipment, were to be considered once the client has evaluated the vehicle.

Personnel Factors

Although the number of personnel involved in the design activities was considered too low for the required timescales (see Section 6.2.2), the broader team involved in the configuration management of information and the purchasing of components was considered too large and cumbersome for this project (see Section 6.2.1 above).

In terms of level of maturity and design capability all members of the development team were very dedicated and experienced.

6.2.3 The Resulting Project Methodology

The development process for this project was based on the process suggested by the VDI [125], shown schematically in Figure B.7. A brief description of the process can be found in Section B.7.

The VDI model of the development process is based on the assumption that both the problem and the solution can be divided into a hierarchical structure of sub-problems and sub-solutions respectively. The design process is therefore one of disassembling the problem (the design objective) into all its sub-problems. This approach and the potential problems associated with it are discussed in more detail in Section 5.3.3.

This disassembly methodology was followed in this project by assigning the principle sub-systems of the vehicle to design engineers with the relevant design experience, as described in 6.2.1. However, due to a lack of available resources no provision was made for the re-assembly of the solution. The integration of vehicle sub-systems is quite complex since sub-systems can have physical interfaces to each other due to being packaged in the vehicle in close proximity to each other, without necessarily requiring functional interfaces. As an example: the steering system and the air-intake system of the powerpack have no functional relationship, and therefore the designer of the steering system would not consider the air-intake system in its design and vice-versa. Since both these sub-systems are usually packaged in the engine bay of a vehicle, each sub-systems has to consider the other during the re-assembly of the solution, i.e. the integration of sub-systems in to the vehicle. Another example would be that the placement of a sub-system within the vehicle can have an

influence on the mass distribution of the vehicle, and therefore by implication on its handling and automotive performance. The scope of work for a designer of a sub-system would not provide him with the insight to make decision on a vehicle level. The re-assembly process is important enough to require the allocation of resources for this function.

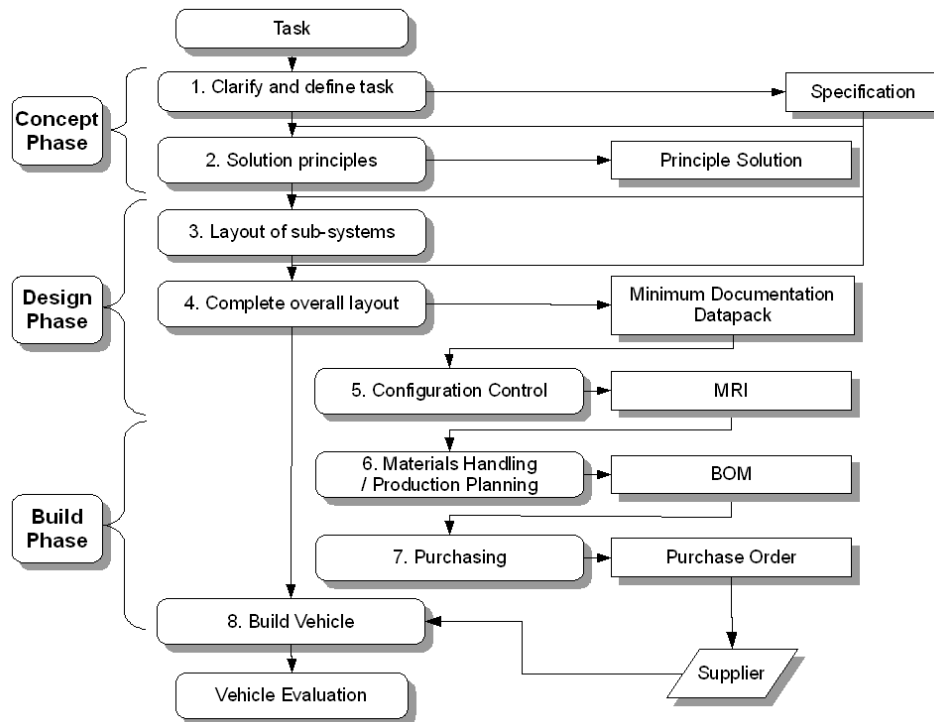


Figure 6.4: The Process Followed for this Project

In terms of the larger process, the VDI process described above, was modified for this project as illustrated in Figure 6.4. As described briefly in Section 6.2.1, the development process was interrupted by configuration control, production planning and purchasing. Once components arrived from the various suppliers, the design team was involved once more with the fabrication and assembly process.

Having built the first vehicle, it was to be evaluated briefly for safety critical automotive performances and then shipped to the client for further evaluation. The second vehicle was then to be extensively evaluated by the test department of BAE Land Systems OMC not only for automotive performance and characteristics, but also against the client's specification.

6.2.4 The Execution of the Project

Section 6.2.1 described the general context surrounding the project, while Section 6.2.2 utilised the context variables to describe specific details that had an influence on the methodology chosen to execute the project (Section 6.2.3).

Successes

From a technical perspective the project was successful: a vehicle was designed and built in extremely short timescales, ultimately meeting the technical requirements described in Section 6.2.1. New major aggregates had been selected and integrated into a vehicle, which stretches the boundary of what is considered the norm in terms of gross vehicle weight and payload, while providing a vehicle with safe handling characteristics and acceptable automotive performance.

Shortcomings

The project had only one major shortcoming: it did not meet the required delivery date of 12 December 2007. The vehicle performed its maiden test run on the test facility the second week in January 2008. A couple of problems with the steering system, power train and electrical system were identified, and the vehicle was brought back to the workshop for corrections. By the first week in February 2008 correction were implemented, the vehicle was back at the test facility and it was verified that the problems had been solved.

The novelty of the digital electrical system introduced a steep learning curve, and therefore caused a large portion of the delay. Since no single point of responsibility existed for the integration of sub-systems in the vehicle, as discussed in Section 6.2.3, an interference resulted between the engine sump (drivetrain assembly) and the front axle (running gear assembly). Fortunately it was an easy problem to solve during the process of building first concept demonstrator vehicle.

By this time it was decided not build the second vehicle, but to keep the first vehicle in South Africa to perform most of the testing originally scheduled for the second vehicle before the end of March.

6.2.5 Conclusion and Discussion

The short-comings of the project described above were caused, in the authors opinion, by the following aspects of the resulting methodology discussed in Section 6.2.3 above.

A Strategic Mismatch

The need for agility and quick responses is often used in an attempt to salvage situations caused by a lack of strategic planning. In this case it has become clear that a mismatch existed between the level of novelty required to meet the customers requirements and the timescales available.

Given that development is a learning process, a couple of aspects of the vehicle could only be addressed properly once the first hardware was available. Therefore the first maiden trip of the vehicle to the test facility was an integral part of the development process rather than a quick quality check before sending the vehicle to the customer as originally intended.

Also, the novelty of using a digital electrical system for the first time at BAE Land Systems OMC, did not correlate well with the urgency dictated by the client.

The Development Process

The discontinuity in the development process described in Section 6.2.1 and illustrated in Figure 6.4, is an attempt to fast-track the development while forcing it into the rigid, procedural world of high volume, continuous production. Since the same personnel in the configuration, materials and purchasing departments were handling the large production order as well as the "fast-track" development project meant that their attention and focus was divided. Tasks for the development project had to slot into gaps in between tasks for the production project.

Since the automotive performance and handling characteristics of the concept demonstrator vehicle were as expected for a four-wheeled with such a high gross vehicle weight, and were well within the prescribed limits for safety, it is clear that the necessary technical skills were available within the organisation and utilised for this project.

Since the first vehicle was intended to be delivered with only a limited amount of testing, indicates that a misconception exists with regard to development in general and novel development such as this one in particular: much of the learning, obtaining of insight and fine-tuning of a design occurs when the first hardware is evaluated. Expecting a novel design to work based only on a paper exercise is folly.

Thirdly, timescales could have been improved if the project methodology were adapted properly to the constraints driving the project, in this case the methodology as shown in Figure 6.5. This would saved a lot of time, since the personnel responsible for defining a component would be directly involved in negotiations with the supplier regarding issues such as leadtimes, quality, manufacturing processes and materials.

Lastly, this project clearly illustrated that the assembly process during the design phase, as discussed in Section 6.2.3, is very important for products where components belonging to different functional sub-systems

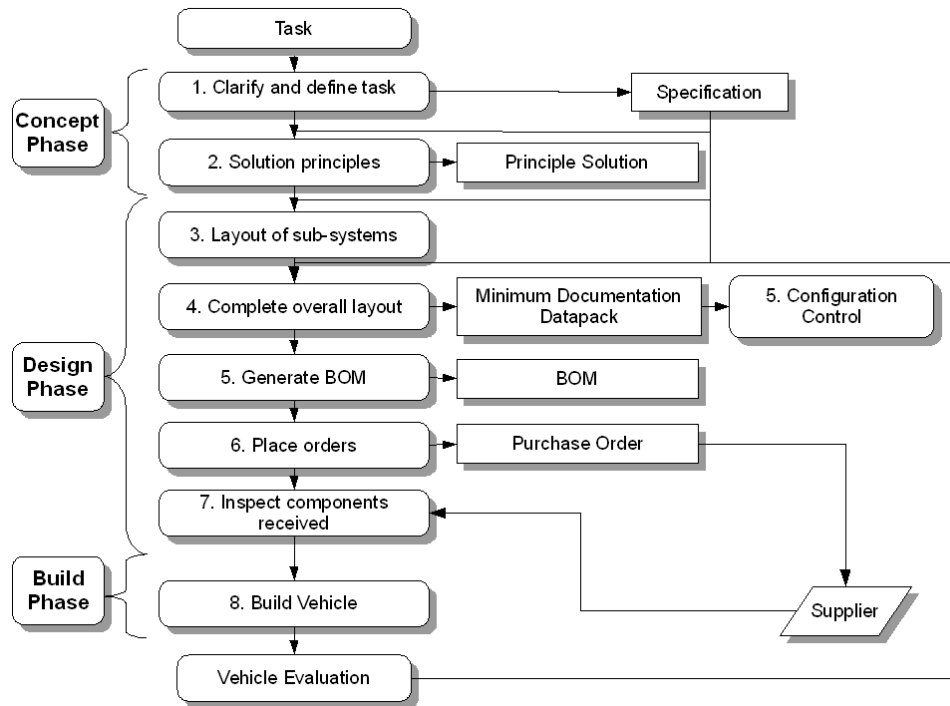


Figure 6.5: The Author's Proposed Development Process

have to be integrated in close proximity to each other. Laptops and aeroplanes are typical products of this nature. The product is clearly more than just the sum of its sub-systems.

6.3 Case Study C

One of the opportunities of evaluating the framework of factors influencing the development process, Figure 5.7, was presenting this framework to students participating in a product development course as part of their curriculum to obtain an engineering degree.

The first case study to evaluate the framework of parameters (figure 1) was conducted as follows: students taking the Management of Product Development course of the Department of Industrial Design at the University of Twente are asked to select a product and describe its development process. In 2006 they were given this task without exposure to the framework described in Section 5.3, while in 2007 the students were presented with the information before they started their project.

Generally, the quality of the articles of 2007 had improved, focussing more on the realities surrounding the product and the organisation that produces it, rather than fairly vague concepts such as designer freedom and employee satisfaction.

In 2006 5 articles of 23 (i.e. 22%) did not consider these factors at all in their proposed development processes, while only 2 articles (or 8%) mentioned whether the development was to be performed in-house or by external sub-contractors. In comparison all students in 2007 made a decision, where the design capability would reside. The percentage of students who decided to adapt a standard process to suit the context of the project increased from 26% in 2006 to 47% in 2007.

Although this case study cannot make a statement regarding the usefulness of the framework of factors in industrial engineering practice, these statistics do indicate that the framework supported the students in their task by increasing their awareness of the context surrounding the specific product they had chosen and its development process.

6.4 Case Study D

When the author of this dissertation presented an article [43], describing the framework of factors that influence the development process, to a fairly small audience at the PICMET '08 conference in Cape Town, a lively discussion resulted among the author and some of the audience. It is interesting to note that the participants in the discussion were academics as well as engineers from industry. This indicates to the author the value of the framework, 5.7, to facilitate thought processes and therefore discussions regarding the development process in practice.

6.5 Summary and Conclusions

Chapter 3 provided the theoretical background with regards to the the engineering development process, while Section 4.1 compared these theories with the daily practice in industry. In light of the differences between the theoretical models and industrial practice, Section 4.2 presents arguments for the adaptation of the theoretical models of the development process to suit the context of each specific development project. The role of methodologies, process models and design methods is described in Section 4.3.

Chapter 5 describes the internal and external factors that, in the author's view, have the greatest influence on the development process. The intention is that an understanding of these factors will lead the project manager, design engineer to construct a project methodology, which will suit the project context, and therefore facilitate the successful execution of the project.

To illustrate how the factors, discussed in Chapter 5, and the resulting project methodology influences a development project, this chapter presented case studies A and B in Sections 6.1 and 6.2 respectively. The former was conducted in a small company, where the freedom existed to adapt the project methodology to the context of the project. The latter was executed in a large company, with very rigid, prescriptive procedures, which govern development projects.

In case study A it was clear that no further improvements to the project could have been achieved through a different approach or strategy (see Section 6.1.4). In case study B, however, areas were identified where the methodology chosen for this project was hindering rather than facilitating the project (see Section 6.2.5).

The research summarised in this dissertation suggests that a better understanding of the relationship between the factors influencing a development project, can lead to the construction of a project methodology which is particularly suited for the circumstances surround the specific project. The prescriptive models of the development project can be utilised as building blocks or guidelines in this construction process.

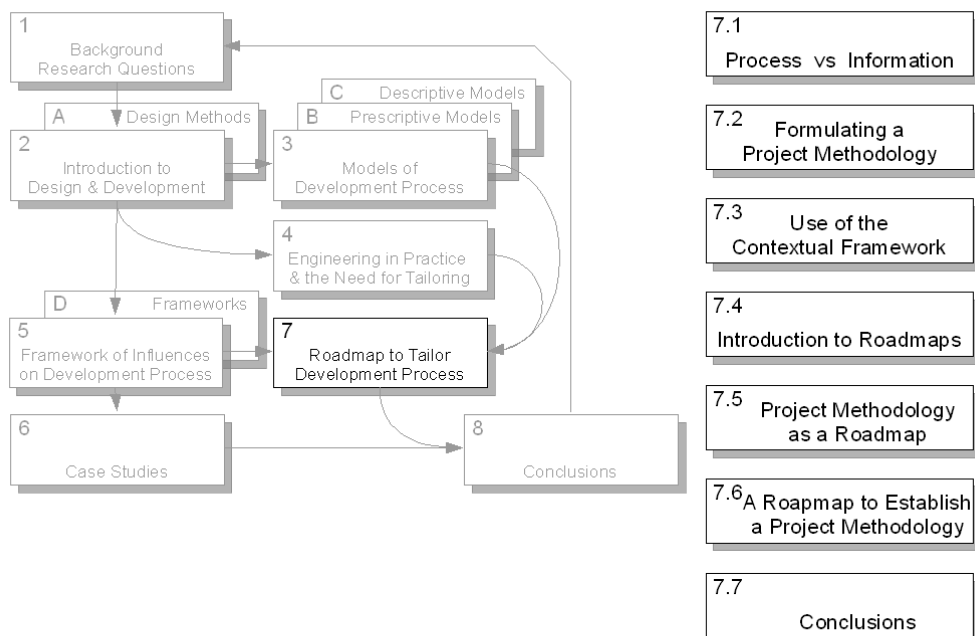
Case studies C and D, in Sections 6.3 and 6.4 respectively, illustrate that the framework proposed in Section 5.3 provides a structure which increases the awareness of a project's context and how it influences the project. In case study C this helped the students to draw up more realistic plans for the proposed products and the companies that would produce them. In case study D the structure provided by the framework facilitated communication among engineering professionals from a wide variety of background by defining the topic of discussion and its context concretely.

Chapter 7

Generating a Formal Project Methodology

Synopsis

While Chapter 3 introduced the reader to various theoretical models of the development process, Chapter 4 identified the need to adapt these standard models to the context of the development project at hand and proposed the use of a formal project methodology to achieve this. In order to understand the relationship between the development process and its context, Chapter 5 investigated the factors that influence the development process and proposed a framework for the contextualisation of the development life-cycle. To illustrate the concepts that this framework is based on, Chapter 6 presented two case studies of industrial development projects executed in South Africa. With this background the following chapter proposes a method of establishing a formal project methodology, referring to the principle development rules identified in Section 3.2, as well as the contextualisation framework developed in Chapter 5.



7.1 Process Orientation Versus Information Orientation

The following chapter discusses the management strategies or methodologies that can be employed during a development process to integrate and control the various aspects of a development project, including the work content in terms of technical quality and performance of resources measured in timescales and expenses. The discussion is largely based on research by Lutters [84], who investigated means of integrating the "manufacturing process" by means of an information orientated control and integration strategy, implemented by a corresponding IT infrastructure. Lutters [84] defined the "manufacturing process" as the part of the product life-cycle ranging from "strategy" to "distribution" as depicted in Figure 2.5. As indicated in Chapter 1, the author's research is focussed on the phases of the product life-cycle called collectively "product creation", particularly in small and medium sized enterprises.

7.1.1 Process Orientated Integration and Control

Before the industrial revolution the craftsman would "design" his product, purchase the required raw materials, plan the production of his product, execute the production plan and sell the finished product himself. By default, this was a harmonious, fully integrated system since it involved only the craftsman himself.

However, after the industrial revolution, organisations attempted (to a degree successfully) to increase the efficiency of this process by division of labour and specialisation. Now all processes could be executed by experts in parallel: the design engineer plans the product, while the process engineer plans and maintains the production facility, the purchaser obtains the raw materials, the production team in the factory produces the products and the sales team sell it to the user. Some of the gains obtained through the increased efficiency is used to integrate and coordinate this fragmented system [84].

Because the principles of "division of labour" and "specialisation" have been successful of the factory floor, the assumption is often made that similar success can be achieved by applying these principles to product development [84]. This assumption resulted in the formulation of prescriptive models of the development process discussed in Section 3.1. As indicated in Chapter 4, when these models fail it is often due to the development process being inherently heuristic and therefore impossible to be defined a priori. In comparison, the production process should be a lot less heuristic, more fully defined and deterministic, if the product development process was executed properly and completely. However, even the production process has heuristic aspects. Generally, when a production process has been established and running for a while, potential improvements become evident, which were not conceivable when the production process was designed.

By pre-defining the development process prescriptively, considerable effort has to be spent in managing and coordinating the process as it unfolds, and the understanding of the development problem as well as its potential solution increases. In contrast, "a design methodology that facilitates the linkage of functions" with emphasis "on the way in which cooperation between functions can be achieved" [84] will utilise the natural flow of the development project and reduce the overhead burden also.

7.1.2 Information Orientated Integration and Control

It is important to realise that the development process, although concerned with the product, does not produce the product (hardware) itself. The purpose of the development process is to produce information. Typically this includes material specifications, definition of components and assemblies in terms of their form and material content, production and assembly instructions as well as operational, maintenance and disposal manuals. The development process can, therefore, be described as a process of evolving information from an idea to the instructions mentioned above. This is discussed in more detail in Section 2.1 where a general overview of design and development is provided.

The management and control of the development process "can shift towards the control of the decision process, in which the control is based on the information content, its evolution and information requirements. This corresponds to the way in which task chains are defined and used" [84]. Utilising the definition of process, functions and tasks provided in 1.2, the development process is depicted in Figure 7.1.

The development process as a whole has an input, which includes a description of the desired output. Depending on how detailed the description of the desired output is, the start of the process usually includes the task to analyse, clarify and understand the desired output. In order to convert the inputs into the final outputs, functions available to the development process utilise resources to perform tasks which evolves the informa-

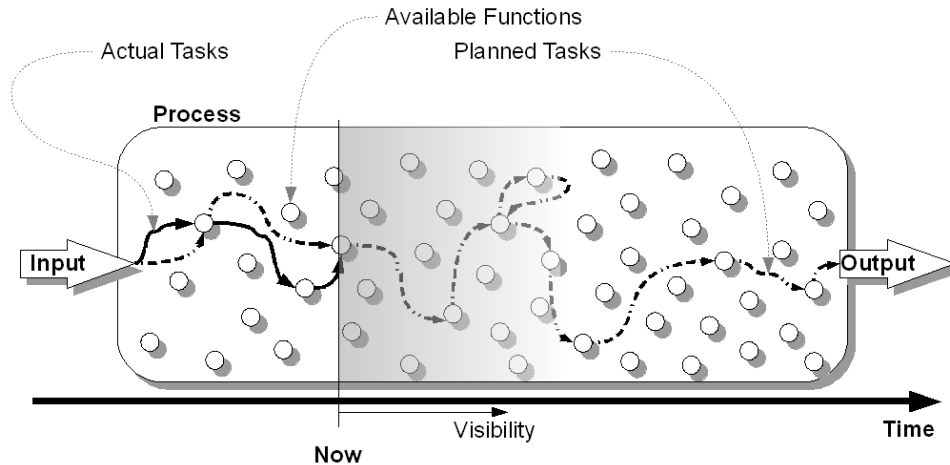


Figure 7.1: The Development Process

tion content of the project. Since development projects in practice are driven by a commercial perspective, the project will start off with a planned task chain, indicated in Figure 7.1 as a dotted line. As information and insight into the development problem becomes available, the actual task chain will probably deviate from the planned one as shown in Figure 7.1. Given the heuristic nature of the process, new information regarding the development problem and its solution becomes available at each point in time during the development process. This implies that only a small part of the task chain can be planned into the future with certainty. This visibility (degree of certainty) of the project variables is indicated in Figure 7.1 by the shaded region. The darker the shading the higher the certainty.

7.1.3 Conclusions and Summary

Processes show, by definition, interdependency among their sub-processes. Structure enables the resolution of conflict among sub-processes by establishing the priority according to the level of importance and urgency associated with each sub-process. This structure can be generic, provided that it "*does not prescribe the possible links*" among sub-processes beforehand [84]. The information orientated approach described above offers precisely such a structure.

Prescriptive models of the development process tend, by their nature, to reduce the human influence on the structure of the development process. By utilising the human intelligence to adapt the process to suit the requirements of the project, the overhead burden of the project can be reduced, the efficiency and effectiveness of the process can be increased by applying resources where and when required, rather than by default in a pre-programmed manner. This is exactly the kind of flexibility in the development process (as well as business processes) that Bichlmaier [18] advocates.

Lutters proposed an information backbone, supported by the corresponding IT infrastructure in terms of both hardware and software, as a means of utilising the information content to control the manufacturing process [84]. Since the research described in this dissertation is conducted with the intent to support SME's (see Section 1.4), which may not be able to afford such investments, the author proposes that the principles discussed in the previous sections are utilised and implemented under human control. Due to their smaller size and less formal internal structure (see Section 5.2.1), SME's are particularly suited for this. The concept of roadmaps, accessible in a collaborative work environment, are just one way to implement these principles.

7.2 The Process of Formulating a Formal Project Methodology

As indicated in Figure 7.2 the planning process for a development project, should generate at least five pieces of information in order to enable a project manager to manage the project:

- | | |
|---------------------------|---|
| Schedule | - A high level timing plan indicating when the major milestones of the project need to be achieved. |
| Scope of Work | - A description of the expected outputs and deliverables to be generated by the project. |
| Work-flow Structure | - A high level description of the major tasks to be performed during the project. |
| Budget | - The expected costs associated with the project. |
| Work Break-down Structure | - Allocation of resources to the work-flow structure. |

The above mentioned documents provide the participants in a development project with a clear indication of the objective to be achieved, the work that is required to achieve the objectives, who should execute the work, as well as how much time and money is available to achieve the objectives. Project management is the process of ensuring that the objectives are achieved within the given constraints. Identification, assessment and strategies for the mitigation of risks should therefore, by definition, be an intrinsic part of project management [72]. It is the author's view that the one of the most fundamental risks to a project would be the misalignment of the the project methodology (as captured in the five documents mentioned above) and the context of the project. For example: executing a project with a high novelty content without the correct level of technical support and opportunities for evaluation and verification, will introduce a probability of failure to the project. The establishment of a project methodology, which acknowledges the context of a project is therefore vital to the success of a project, and a risk management activity in its own right.

It should be intuitively clear that these items are intrinsically linked: e.g. the schedule and the work break-down structure add timing constraints and resources to the work-flow structure respectively. The labour part of the budget is then defined by the hourly rate of these resources for the time spent on the project as per the schedule.

All five items described above are defined by the context of the development project. The schedule is usually dictated by circumstances surrounding the project. For instance, the date of the show to exhibit the next generation product, or the customer's timescales for the deployment of the new product. The scope of work and its structuring depends largely on the information content of the project (see Section 7.1.2 above): i.e. what information (e.g. drawings, material specifications, assembly and maintenance instructions) needs to be generated during the project in order to enable the efficient production of a product that will satisfy the needs of the customer? Lastly, money and personnel resources enable the project: in principle, it is difficult to execute large, complex projects on a shoe-string budget, without people to execute the work.

Having defined the information necessary to manage a project, the next section discusses how the contextual framework can assist in generating this information by increasing the awareness of the project's unique characteristics and constraints.

7.3 Using the Contextual Framework to Tailor the Methodology for a Specific Development Project

Having established in Chapter 4 that no two development projects are ever the same and therefore there exists a need to configure the development process according to the requirement and context of each specific development project, the author discussed the contextual variables that influence the development process in Chapter 5. The intended logic is as follows: if the personnel, tasked with the planning and execution of a development project, are aware of the context of the project, they will be able to structure the project methodology accordingly.

It is therefore necessary to translate the data in the contextual variables (see Tables 6.1 and 6.2 as examples) into one or more measures that will enable the project manager to establish a specific project methodology for the given context. With this in mind the author analysed the contextual variables and found that they fall into three categories: project background, project enablers and project structure, as shown in Figure 7.3.

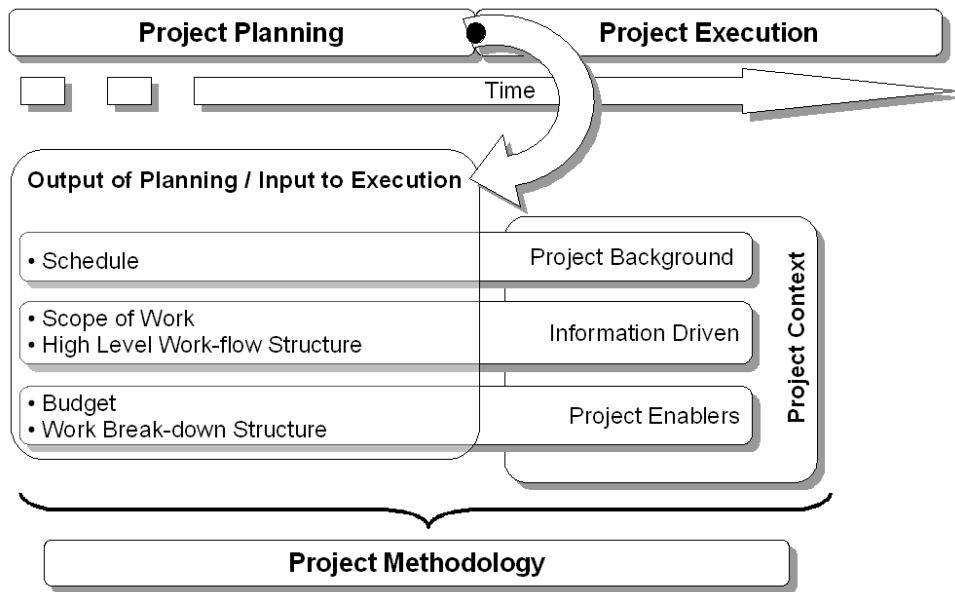


Figure 7.2: The Process of Formulating a Project Methodology

7.3.1 Project Background

The project background consists of all the variables which are given and fixed for this specific project context. These include the structure, type and organisational maturity of the organisation that will execute the project (see Section 5.2.1 for details). For instance a multi-national manufacturing company will (usually) not change its hierarchical structure or its business strategy to suit a specific development project. It is (usually) expected that the project will adapt to suit this type of context. The organisational maturity, which determines a company’s approach to its business processes, cannot be altered to suit each new development project.

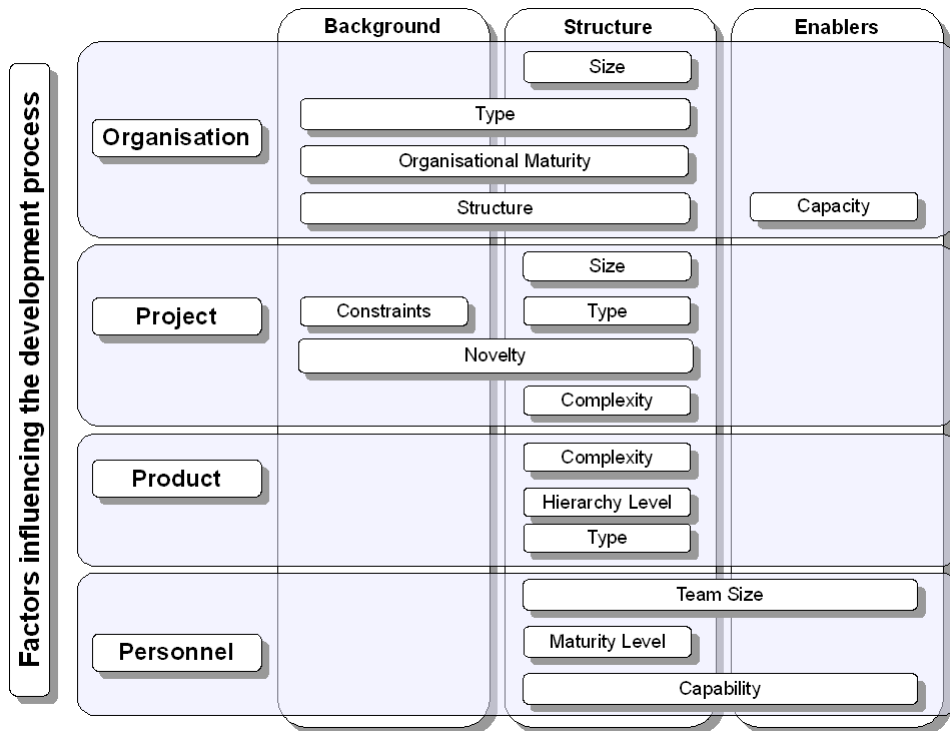


Figure 7.3: Categories of Contextual Variables

Project novelty (see Section 5.2.2 for details) is usually dictated by the project background. For instance, the company needs to develop a radically new product range to enter into a new market segment. Or the user requirements for the new product are such that the traditional approach cannot be applied any more, and therefore components, assemblies or sub-systems have to be replaced with new, innovative concepts.

The other contextual variable that falls in this category describes the project constraints (see page 48 of Section 5.2.2 for details), which typically includes the timescales and budget of a development project. All these variables are fixed (usually) and cannot (easily) be changed or influenced to suit a particular development project. They do, however, influence the project. It stands to reason therefore, that the development project has to take these variables into account in its methodology, to guide and direct the planning and execution of the project accordingly. As indicated in Figure 7.3 these variables also affect the amount of structure the particular project background imposes on the project. This is discussed in more detail in Section 7.3.3.

7.3.2 Project Enablers

In comparison to the variables in the "project background" category discussed above, project enablers can be influenced to a degree to suit the project, and therefore enable the successful planning and execution of the project. These include organisational capacity (see page 45 of Section 5.2.1 for details), as well as project team size and personnel capability (see Section 5.2.4 for details). Organisational capacity and team size can be augmented by sub-contracting and/or hiring of personnel, or making an existing resource in the company available to a project, without employing new personnel. Personnel capability is increased by sub-contracting to or hiring of personnel with the required skills for this project. Similar to the variables in the "project background" category, discussed above, project enablers also affect the amount of structure the project requires. This is discussed in more detail in the next section.

7.3.3 Project Structure

All the variables in this category influence the amount of structure required in the project methodology to execute the project as efficiently and effectively as possible. Chapter 5 discussed the contextual variables and their interrelationship in detail. The paragraphs below summarise the aspects particularly relevant to the required level of "structuredness" for a development project.

Background variables impose structure on the project from context external to the project. For instance: large corporations are characterised by steep hierarchical structures, specific departmental specialisation and highly defined division of labour. The characteristics demand a very structured approach to a development project in order to ensure that all levels of management are informed and all activities in the relevant departments are coordinated and integrated into the larger project plan. Similarly, the core purpose of an organisation creates a mindset which is transferred on to the project, irrespective of whether it benefits the project or not. A company specialising in the mass production of a particular product will usually focus on production costs, rates and efficiencies. In such an organisation, development, with its uncertain nature of product and tooling trials, will usually be relegated to a place of secondary importance, i.e. production comes first! Project constraints have a similar effect on the project: project with tight money and/or time budgets, require intense coordination and integration efforts to achieve the objectives within the required constraints. The correct level and type of structure can assist to achieve this.

In comparison, while project enablers also dictate the "structuredness" of the project to a degree, the project has more influence over the variables. For instance: design capacity and capability can be increased by hiring or sub-contracting. Team size, however, is dictated by the amount of work and the timescales of the project. In principle large, complex projects and/or development of complex products require structure to facilitate the integration of the different components of the project. Due to the size and complexity (of the project, the product or both) the process has to be sub-divided into sub-processes as dictated by the "perceivability" threshold [108] for the particular project context, as discussed in Section 5.3.3.

Assessing the variables of the contextual framework (see 5.7) and their interrelationships, as described above, enables one to determine the level of structure a specific project requires. The project manager is therefore, enabled to choose a prescriptive model as a basis for the adaptation of the development process to suit the requirements of the specific project context by assessing the measure of "structuredness" required through the contextual framework (Figure 7.3), combined with the measure of "structuredness" provided by the various prescriptive models of the development process (see Figure 3.2). As indicated in Section 3.1.3, prescriptive

models provide a good basis for project management and control due to their well-defined structure. It is for this reason that it is sensible to use a prescriptive model of the development process as a basis for a project methodology.

The first step would be to evaluate the variables of the contextual framework, discussed in Section 5.2, resulting in a table such as those generated for case studies A and B (Table 6.1 and 6.2 respectively). As an example, Table 7.1 shows the variables of the contextual framework and their specific values for case study B, arranged in the format of Figure 7.3. In order to convert the values in the structure column to a measure which can be applied to Figure 3.2 to select a prescriptive model of the development process as a basis, the values "low", "medium" and "high" need to be converted into numerical values, as shown in Table 7.2. For each of the variables the value is multiplied by a contribution (weighting) factor, which is a measure of how much the particular variable affects the "structuredness" of the project. The resulting products are summed, and the result can be applied to Figure 3.2. For the example of case study B, the following would have been acceptable choices: the models of Pahl & Beitz, VDI, Ullman and Pugh.

It is important to note when related factors of the framework, as discussed in Section 5.3, exhibit large differences in score. For instance, in Table 7.1, a small project (score = 2) was executed by a small project team (score = 2) in a large organisation (score = 4). Taking into account that the constraint for the project was the time available to complete it, the probability of failure, i.e. not meeting all the project objectives, was increased by not addressing this discrepancy when the project methodology was established. Similarly, the discrepancy between the company focussing on production on the one hand (score = 5), and producing products that are engineered-to-order on the other (score = 2), creates issues in the business process, which spill over into the project. Consistent quality and production efficiency is best achieved by producing large quantities of products with the same configuration, while engineering-to-order implies the production of "specials" in small quantities.

This type of discrepancy is also evident on a higher level: for instance in Table 7.1, the personnel factors require an unstructured project methodology (score = 1.33), due to the small number of capable and mature engineering professionals involved, while the product requires a fairly structured methodology (score = 4.33), due to the large batch size and the high level of product complexity. The contribution factor is a means of finding an acceptable compromise between the variables which demand a high level of structure (scores of 4 or 5) and those which require only little structure (scores of 1 and 2). In the example shown in Table 7.1 it was decided that the demands of the organisational factors were more important than the product factors by applying a contribution factor of 0.3 to the former and only 0.2 to the latter.

There are two other aspects that have to be considered when choosing a prescriptive model for a basis for a project methodology: the model's scope and its characteristics.

The Scope of the Prescriptive Model

As indicated in Figure 3.3 most prescriptive models of the development process start at the "strategy" phase of the product life-cycle (Figure 2.5). Of the prescriptive models considered during this research, only systems engineering addresses "research" as a separate phase, which gathers data, investigates technology and products in order to be able to formulate the strategy in the next phase. Quite a number of the models address only the strategy and design phases of the product life-cycle. Cross, Suireg and Ullman address the next phase where the new product is evaluated. Concurrent Engineering, Integrated Product Development and Pugh address the product life-cycle up to the distribution phase. Of the models considered in this research only Ullman and Systems Engineering include the disposal phase.

It is therefore important to realise which phases of the product life-cycle are required for a particular project, depending on the purpose of the project. The intent to bring a product to market could, for instance, be broken up into a number of projects each lasting a financial year: the first year might address strategy and design, while the next year is dedicated to an extensive evaluation process, before industrialisation and production are addressed in year three.

The Characteristics of the Prescriptive Model

Mixing and matching parts of various models of the development process, depending on their strengths and characteristics should also be considered. Table 3.2 provides a brief summary of the most obvious strengths and focus areas for each of the prescriptive models considered in this research.

Factors influencing the product development life cycle		Values for each of contextual variables	Score	Sub-Score	Contribution Factor	Score
Organisational Factors	Organisational Size	Large: 400 – 1999	4	4.00	.30	1.20
	Organisational Structure	Steep, nation wide	4			
	Organisational Type					
	Purpose	Production	5			
	Production Size	Large batches > 20	4			
	Stock philosophy	Engineer-to-order	2			
	Organisational Maturity	Organisation-wide procedures and support	4			
	Design Capacity	Huge: 50 >	5			
	Project Size	Small: 5,000 – 50,000 man hours	2			
	Project Complexity					
Project Factors	Complicacy	Medium	3	2.63	.30	.79
	Dynamics	Medium	3			
	Opacity	Low	1			
	Interdependency	High	5			
	Polytelie	Low	1			
	Project Type	Configuration Design / Redesign	3			
	Project Level of Novelty	Medium	3			
	Product Type	Large batches > 20	4			
	Product Complexity					
	Complicacy	High	5			
Product Factors	Opacity	High	5	4.33	.20	.87
	Interdependency	High	5			
	Polytelie	Medium	3			
	Product Position in System Hierarchy	Machine	4			
	Team Size	Small: 4 – 10	2			
	Level of Maturity	High	1			
Personnel Factors	Level of Design Capability	High	1	1.33	.20	.27
						1.00
						3.12

Table 7.1: The Factors that Influenced the Development Project Discussed in Case Study B, Section 6.2

		Score Allocated					
Factors influencing the product development life cycle		1	2	3	4	5	
Organisational Factors	Organisational Size	Tiny: 3 – 14	Small: 15 – 74	Medium: 75 – 399	Large: 400 – 1999	Huge: 2000 >	
	Organisational Structure	Flat, single location	Medium, single location		Steep, nation wide	Very steep, international	
	Organisational Type						
	Purpose	Design & development				Production	
	Production Size	Unit	Small batches < 20		Large batches > 20	Mass production	
	Stock philosophy		Engineer-to-order	Manufacture-to-order	Assemble-to-order	Manufacture-to-inventory	
	Organisational Maturity	Not done	Individually implemented	Project wide selection, support & implementation	Organisation-wide procedures and support	Improvement method followed	
	Design Capacity	Tiny: < 3	Small: 4 – 10	Medium: 11 – 20	Large: 21 – 50	Huge: 50 >	
	Project Size	Tiny: < 5,000 man hours	Small: 5,000 – 50,000 man hours	Medium: 50,000 – 500,000	Large: 500,000 – 5,000,000 man hours	Huge: > 5,000,000 man hours	
	Project Complexity						
Project Factors	Complicacy	Low	Low – Medium	Medium	Medium – High	High	
	Dynamics	Low	Low – Medium	Medium	Medium – High	High	
	Opacity	Low	Low – Medium	Medium	Medium – High	High	
	Interdependency	Low	Low – Medium	Medium	Medium – High	High	
	Polytelie	Low	Low – Medium	Medium	Medium – High	High	
	Project Type	Catalogue Design		Configuration Design / Redesign		Original Design	
	Project Level of Novelty	Low	Low – Medium	Medium	Medium – High	High	
	Product Type	Unit	Small batches < 20		Large batches > 20	Mass production	
	Product Complexity						
	Complicacy	Low	Low – Medium	Medium	Medium – High	High	
Product Factors	Opacity	Low	Low – Medium	Medium	Medium – High	High	
	Interdependency	Low	Low – Medium	Medium	Medium – High	High	
	Polytelie	Low	Low – Medium	Medium	Medium – High	High	
	Product Position in System Hierarchy	Component	Assembly	Sub-system	Machine	Plant	
	Team Size	Tiny: < 3	Small: 4 – 10	Medium: 11 – 20	Large: 21 – 50	Huge: 50 >	
	Level of Maturity	High	Medium – High	Medium	Low – Medium	Low	
	Level of Design Capability	High	Medium – High	Medium	Low – Medium	Low	
	Personnel Factors						

Table 7.2: Numerical Values Assigned to the Various Contextual Variables

For the development of a relatively simple product with low levels of novelty and complexity, French's or Archer's model of the development process may be applicable. A project to develop a product with a specific requirement for modularity might consider utilising the models of Pahl & Beitz or VDI as their framework for the project. Very complex projects, where many organisations collaborate, might require the Systems Engineering approach. In all cases it might be sensible to consider the concurrency aspects advocated by Concurrent Engineering and Integrated Product Development.

How can this be applied to industrial practice? In support of the answer to this question, Section 7.4 introduces the reader to the tool that the author proposes for this task: the concept of a roadmap. The end result, i.e. the project methodology for a specific development project, tailored to suit the project context, is discussed in Section 7.5. Armed with this information, Section 7.6 discusses the means of generating the project methodology in the form of a roadmap.

7.4 Introduction to the Concept of a Roadmap

Frameworks and roadmaps have become popular structures to represent procedures, life-cycles and processes [77], [79], [99]. Various definitions for these terms exist. For the context of the research described in this dissertation, the author prefers the definition provided by the Industrial Engineering Department of the University of Stellenbosch, which defines a roadmap as "*a layout of descriptive paths that multidisciplinary teams can use as a guiding framework for collaboration efforts towards a common goal.*" [39]. The emphasis in this definition is on "guiding framework", describing all aspects of the project, while providing an "*environment with enough freedom to innovate*" [39].

Models of the development process such as those described in Section 3.1 easily deteriorate into prescriptive scripts or programs, since they intend to describe widely applicable, generic situations, rather than the specific project embedded in its context. "*Instead of demarcating the course*" of a project "*by imposing behaviour driven by a design method*" or model of the development process "*roadmaps attempt to facilitate the course of actions. A roadmap is a structured description of all aspects that constitute a project. It is used to guide the team members in executing the project*" [85].

A roadmap should contain the following descriptions:

- | | |
|-------------------------|--|
| "Where" to go | - a structured high-level framework of desired milestones. |
| "Who" is travelling | - an allocation of a person or team to a milestone. |
| "How" to get there | - descriptions and information guiding the user of the roadmap to his destination. |
| "When" to arrive | - a target date for each of the milestones. |
| "What" goals to achieve | - objective to achieve at each milestone. |

In addition the computer environment which houses the roadmap and provides an interface for interaction to the user in order to make it accessible and useful, should provide the following:

- controls in order to manage efficiency and effectiveness of the process of travelling down the path on the roadmap, and
- an information repository to collect all the information relevant to the specific project under consideration.

This information should be able to be re-used by converting the most relevant information, methods, tools, techniques of the current project into the guiding information of the next, similar project. Knowledge management and storage for re-use have been recognised as a requirement to increase the effectiveness and efficiency of engineering development by organisations such as the G-SCOP Laboratory, Institute National Polytechnique de Grenoble (INPG), France and Integrated Manufacturing System Research Center (IMSRC), King Mongkut's Institute of Technology North Bangkok (KMITNB), Thailand [24]. The topic of knowledge management for engineering development is too large and complex to address within the scope of this research project. The author acknowledges the need and has presented a simple solution in keeping with the initial intent of providing solutions suitable for small enterprises.

Figure 7.4 shows the concept of a roadmap implemented in a collaborative software environment called, EDEN™. According to the developers of the software, a company called Indutech in Stellenbosch [6], South Africa, EDEN™ is a "*enterprise-wide innovation management platform*", providing the characteristics of a roadmap described above.

The top left-hand area of the EDENTM interface provides the high-level framework of milestones to be achieved during this project, the "where" of the roadmap. For each step (activity) in this framework the top right-hand area of EDENTM interface provides a definition and description of the task to be performed, as well as the goals to be achieved, including by when and by whom. This provides the "who", the "when" and the "what". The next area, in the middle of the right-hand column, provides material to assist the user with the completion of the task assigned to the specific step of the roadmap, i.e. the "how". This can typically include guidelines, templates, software tools, access or implementations of design methods such as those described in Appendix A. The bottom right-hand area of the EDENTM interface provides access to the information repository. It is here where the information that the project generates is stored. This information is visible to all team members so that each team member can be aware of what the other members of the team are working on, what decisions they have made and how far they have progressed with their assigned tasks.

7.5 A Formal Project Methodology as a Roadmap

Section 7.2 defined the five pieces of information required to build an engineering management plan as the backbone of a project methodology: the schedule, the scope of work, the work-flow structure, the budget and the work break-down structure. The concept of a roadmap discussed in the previous section inherently provides for four of the five aspects of an engineering plan in its structure:

- the schedule - "when", the date for each milestone.
- the work-flow structure - "where", the high-level framework of milestones.
- the scope of work - "what", the goals to achieve for each milestone.
- the work-breakdown structure - "who", the resource responsible for achieving the milestone.

The part of the engineering plan that is not addressed by the roadmap is the budget. This is because even small companies are required to have financial systems to account for income and expenses. If these are run efficiently, it should be easy to draw a statement on a regular basis to establish the current status of labour and material expenses. It is then just up to the project manager to compare these with the status of the work completed on the project to determine the cost required to complete the project.

Table 7.4 shows the requirements for development support facilities, as defined by Pham et al. [100], as well as the means of addressing these requirements through a roadmap in the EDENTM environment. Although all requirements are addressed, they are intentionally addressed in a very general, generic manner, in order

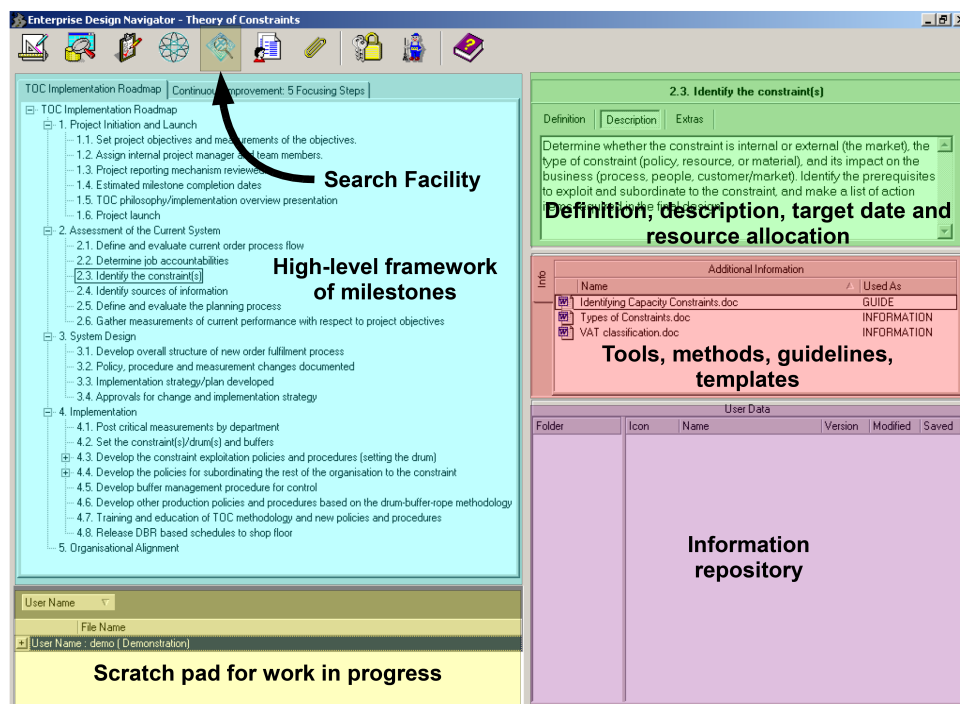


Figure 7.4: Roadmap Implemented in EDENTM

to provide the designer, project manager or chief design engineer the freedom to add as much structure to his project methodology as he deems appropriate for the given circumstances of his project. By contrast, the visualisation of individual design decisions and their effects on the design as a whole, for example, was addressed by Pham et al. [100] in a more formal, structured manner by generating an automated decision tree. Should a project, company or industry require such a structured approach, the facility could be added to the roadmap implemented in the EDEN™ environment by making appropriate software available.

Blessing[20] defined very similar requirements for a design support system. In terms of the interaction between the user and the system in general, the system needs to be "*cooperative, subordinate, flexible and useful*". With regard to supporting the design activities specifically, it needs to be "*comprising, assisting, guiding and structuring*". Table 7.5 provides a description of each term, as provided by Blessing, the means by which a roadmap implemented in a software environment like EDEN™ would address each requirement.

7.6 A Roadmap to Generate a Project Methodology

The author proposes the use of a roadmap to guide the project team, led by the project manager or chief design engineer, through the process of generating a project methodology in the form of a roadmap itself, which would then be used to guide the project team through the execution of the project. It is important to note that the process described below is not intended to be prescriptive. It is intended to provide means for the project team to visualise a large number of (often conflicting) variables, in order to obtain an approach to the project, which represents an acceptable compromise in addressing these variables. The EDEN™ environment will even enable to adaptation of this approach during the execution of the project to take into account new information obtained. Due to the heuristic nature of the design process, this particularly true for original design problems.

Section 7.3 provided a mechanism to indicate the level of structure required for a particular project. It also suggested that a prescriptive model of the development process can be chosen as a basis for tailoring a specific project methodology. Roadmaps for the EDEN™ environment have been generated for a number of these prescriptive models, thereby building a repository of roadmaps that can be used for product development. The facility exists in the EDEN™ environment to build a new roadmap by copying either parts or complete roadmaps from this repository.

The proposed roadmap to generate the project methodology has the steps and sub-steps described in Table 7.3. Once a project has been executed its roadmap can be added to the repository for future reuse.

7.7 Conclusions

While this research has argued for the need to tailor the development process for each project according to its specific context in Section 4.2, Section 5.2 identified and described the factors that influence the development process most. By applying a scale from 0 to 1 to each of these factors, as described in Section 7.3.3, an indication can be obtained what level of structure is required for a given project. Taking the scope and characteristics of the prescriptive models of the development process into account, as discussed in Section 7.3, a model can be chosen as a basis for a project methodology. Finally, Sections 7.4 and 7.5 described how the concept of a roadmap, implemented in a software environment such as EDEN™, can be utilised to formalise a tailored project methodology for a specific project.

The concept of a repository of available models of the development process within a collaborative design environment, that will support object oriented configuration and manipulation of roadmaps, will allow the design team to configure the development life-cycle "on the fly", based on past experience as well as all the requirements and parameters of the specific product, development project, enterprise and project team.

No	Description of Task	Reference
1	High-level structuring of project	
1.1	Assess the characteristics of the project by means of the contextual framework	Sections 6.1.2 and 6.2.2
1.2	Calculate the level of structure required for the project, based on the project characteristics.	Section 7.3.3
1.3	Assess the scope of the project	Section 7.3.3
1.4	Match the characteristics of the project to the characteristics of one or more prescriptive models of the development process	Section 7.3.3
1.5	Generate the high-level structure for the project by choosing either the entire prescriptive model of the development process, or by combining the relevant parts of a number of prescriptive models to suit the project context.	
2	Detail planning of project	
2.1	From the desired outcome (deliverables), establish the scope of work for the project.	
2.2	Assign the appropriate resources to the scope of work, i.e. the work breakdown structure.	
2.3	Establish the work-flow structure by linking the task in a logical fashion, and assigning a duration to each task.	Section 7.2
2.4	Calculate the labour budget for the project by multiplying the cost of each resource with the duration of its input to the project. Estimate material costs and costs to out-source tasks where required.	
2.5	Iterate through steps 2.2 to 2.4 to balance project timescales, available resources and project expenses, taking project cash-flow into account.	

Table 7.3: The Steps of the Roadmap to Generate a Project Methodology

Requirements Identified by Pham et al.		Means of Addressing the Requirement by a Roadmap in the EDEN Environment
Design-methodology-related Requirements	Allow designers to capture their ideas and thoughts.	All team members store their work in the information repository.
	Enable designers to track their ideas and thoughts.	The information repository provides version control for all documents deposited in the repository. Old versions are available for review.
	Enable designers to decide whether to delay their design decision when information is not available.	The high-level framework of milestones provides the designer with insight into the rest of project, enabling him to assess the impact of his work on the rest of project.
	Allow the designers to add, edit or remove their design decisions any time throughout the design process.	Although EDEN does not make provision to capture design decisions specifically, the information repository is intended to store all project related information, including design decisions, if required.
	Indicate the effects of any change in their past design decisions on current and future design decisions.	No Automated system. However, the high-level framework of milestones provides the designer with insight into the rest of project, enabling him to assess the impact of his work on the rest of project.
	Enable designers to reuse ideas and information recorded during the design process in future design problems with similar requirements.	The information repository is intended to enable reuse of gathered information for the next project.
Computational-platform-related Requirements	Saving all records of ideas and requirements at any time for later use.	
	Searching for and visualisation of the designer's design decisions at any stage during the design process.	An Intelligent search facility to enable searching the file names and content of files in the information repository .
	Providing a flexible input interface so that the designer can record his ideas through text, sketches or graphical representation.	Any type of data file can be inserted into the information repository.

Table 7.4: Requirements for Development Support Facilities [100] and the Level that EDEN™ Meets these Requirements

Requirements Identified by Blessing		Means of Addressing the Requirement by a Roadmap in the EDEN Environment	
Requirements with Regard to System-User Interaction in General	cooperative	The system needs to be an " <i>important reasoning component</i> ", either actively (where the system " <i>proposes steps or solutions</i> ") or passively, leaving the " <i>initiative to the user</i> ".	By providing a description of each step of the development process, including the expected outputs of each step, within the context of the whole project, the system enables the designer to evaluate his task and its influence on the complete project.
	subordinate	Even though the system can propose steps or solutions, the user retains the responsibility to decide whether the steps or solutions are implemented or not.	The user is in complete control to create his own project roadmap for the project, and to modify it at any stage if required. Even the roadmap can suggest the use of certain tools, methods or templates to assist with the execution of a task, it is the user's decision whether to use them or not.
	flexible	The system should allow the user the freedom to choose the appropriate approach (methodology) for each development project.	
	useful	This term describes the trade-off between costs and benefits of the system. Costs refer to the financial costs of acquiring and installing the system, as well as the potential additional overhead imposed on the project by requiring that the design team use the system during the execution of the project.	The EDEN software is cost-effective compared to competitors. The intuitive user interface reduces the learning curve for a new user to a minimum.
Design Specific Requirements	comprising	The system should support " <i>the design process from the beginning to the end</i> ".	Since the user can and should generate his own roadmap for each specific project, the system supports the complete process as defined by the user.
	assisting	The system should provide assistance to the designer by suggesting relevant knowledge, methods and tools.	The system provides a means of suggesting methods, tools and templates for each step of the development process. It is the responsibility of the project manager, who controls the roadmap to provide these where necessary and applicable.
	guiding	The system should provide guidance by suggesting the next step in the development process. At the same time it should be flexible enough that the designer can modify the process if required.	By definition the roadmap provides guidance with regard to the next step of the development process and how the current one relates to all the others.
	structuring	The system should provide a " <i>structure to ease documentation and retrieval of project data</i> " " <i>in the current and future projects</i> ".	For each step of the development process roadmap, the system provides a data repository. This structure as well as the sophisticated search facility provides ease of storage and retrieval of data.

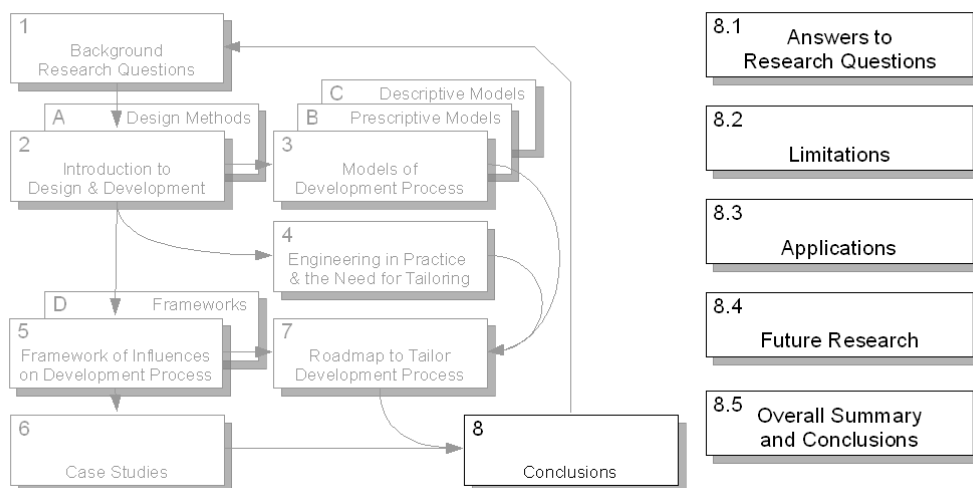
Table 7.5: Requirements for Development Support Facilities [20] and the Level that EDEN™ Meets these Requirements

Chapter 8

Research Findings and Conclusions

Synopsis

This chapter answers the specific research questions posed in Section 1.5 by summarising the relevant information from the previous chapters. The limits and applications of the research summarised in this dissertation are also discussed, as well as a description of future work, that would complement this research.



8.1 Answers to Research Questions

The contrast between the prescriptive models of the development process proposed by design science and industrial practice (see Section 4.1), has prompted the author to ask questions about the nature of the development process and its application in industry. This research project addressed the challenge of how to appropriately configure each design project within a product development life cycle according to a number of framework contexts. In execution of the project a number of significant contributions have been made that lead to significant recognition within the design sciences domain culminating in a proposed solution, described in Chapter 7, which facilitates the configuration of the development process to suit the context of the development project. This results in narrowing the gap between the proposals of design science and industrial practice.

Firstly the landscape of product design methodologies has been documented to provide an extended categorization of a wide range of models and methodologies.

Secondly a framework for configuring a project methodology has been compiled. This framework has been tested in three industrial projects which the author participated in. It was also evaluated in a teaching environment exposing some 120 students in a number of projects over a two year period.

Thirdly the project outcome was presented at two international conferences where significant interaction was initiated. The latter to such an extent that the paper for the design conference in Enschede in April 2008 was selected to be published in the international CIRP addition on Design in 2008.

The subsequent EDENTM roadmap that supports the proposed methodology also makes available the results of this research to a wider design community and will contribute to further developments in the field of design.

At the onset of the project a number of research questions were identified (see Section 1.5), to be answered during the course of the research. All these questions have been answered indirectly in the preceding chapters of this dissertation. For the sake of completeness and clarity the answers are summarised below with references to the parts of the dissertation that discuss the particular topic.

8.1.1 Given a Specific Project, with Specific Resources within a Given Environment, How Does One Select and Construct a Design Methodology from a Repository of Templates?

In addressing the preceding question, as described in more detail in Section 7.6, the author proposes the following:

- Determine the characteristics of the project and its context by assigning values to each of the factors of the contextual framework, as shown in Table 7.1, to indicate the level of structure required for the project. By comparing the final score obtained with the level of "structuredness" of various prescriptive models of the development process, as shown in Figure 3.2, a first-order selection can be made of the prescriptive model to use as a basis for the project methodology.
- By comparing the scope of the prescriptive model (Figure 3.3), as well as the strengths and characteristics (Table 3.2) with the project's requirements, a final choice of prescriptive model can be made. It is conceivable that one might choose the concept phase from one model and the detail design phase from another, etc. depending on what suits the project the best.
- Utilising a software environment such as EDENTM and a repository of roadmaps for the various models of the development process, one can generate a specific roadmap, tailored to suit the needs and characteristics of a specific development project. The roadmap would then be the project methodology expressed in a form that is beneficial to the execution of a project. Prototypes of such roadmaps are available at <http://indutech.cust.iaf.nl/roadmaps>. For access username and password need to be requested via email from support@indutech.co.za.

8.1.2 Can Benefit be Obtained by Combining Design Methods to Form a Project Methodology?

No. Design methods, as defined in Section 1.2, are rigid procedures aimed at obtaining specific results (outputs) from specific inputs. By merely stringing design methods together, even if they are relevant to the

design problem, will not equate to a project methodology. The relationship between the project methodology, the models of the development process and available design methods and tools, is illustrated in Section 4.3. A project methodology needs the flexibility to take the context of the project into account, which cannot be comprehensively accommodated by a design method. A project methodology is based on an approach or a strategy, which is more than the design methods it incorporates.

8.1.3 Can Methods be Established to Define a Methodology Optimised for a Specific Project?

With reference to the specific word "method" in this question, the answer is "No". The reason for this, is the implication of a rigid, prescriptive procedure to be followed: provide some parameters, crank the handle and obtain an optimised methodology, ready for consumption. The decisions made during the establishment of a project methodology, is based on the value system of the individual or group who are planning and executing the development project. This value system differs from case to case.

However, if the question is changed to: "Can guidelines be established to define a methodology optimised for a specific project?", the answer would be "Yes". The influencing factors described in Chapter 5 are the basis for such guidelines, and the roadmap described and illustrated in Chapter 7 provides the assistance to generate an sufficient and comprehensive project methodology.

The author's premiss is that it takes understanding of the project context to generate a project methodology, and it takes the understanding of both the context as well as the strategy behind the methodology to execute it properly. The same understanding would have to be generated by the person tasked to execute the project, even if the methodology could be obtained in an automated, mechanistic manner.

8.1.4 How do I Specify, Choose and Construct a Development Methodology?

Chapter 7 provides one way of generating a project (development) methodology. Since humanity has executed many successful development projects, there must be innumerable ways of defining a development methodology. Many of these will be intuitive, based on experience, and not formalised.

8.1.5 How do I Measure the Success or Appropriateness of My Choices?

The answer to this question is best described by the old adage that "the proof of the pudding is in the eating", i.e. during the execution of the project one can establish whether the project progresses according to plan. If not, the situation can be analysed to determine whether a change in the approach to the project, or a part of it, will improve the matter. The two case studies (Sections 6.1.4 and 6.2.5) and the concluding section of Chapter 6 illustrate this.

8.1.6 How do I Comprehensively Quantify My Needs with Respect to a Development Methodology?

It is the author's opinion that any project methodology should, at least, provide the five pieces of information discussed in Section 7.2: schedule, scope of work, work-flow structure, budget and work break-down structure. Furthermore, the level of structure, "prescriptiveness", of the project methodology should be aligned to the needs of the project. An indication of this can be obtained by completing the matrix shown in Table 7.1. One should be aware of and address situations causing large variation in scores for related contextual variables, as discussed in Section 7.3.3.

The scope and characteristics of the prescriptive model of the development process chosen as a basis for a project methodology should also be aligned with the scope and requirements of the project, as discussed in Section 7.3.

The implementation of the project methodology, enabling it to be applied by the project team in the execution of the project should make provision for attributes such as those listed in Table 7.4.

8.1.7 Can a Methodology be Represented in a Roadmap?

Yes, as discussed in Section 7.4 a roadmap can address all the needs of a project methodology. The EDENTM software environment is a tool to implement the concept of a roadmap in a manner that it can be a useful instantiation of a specific development methodology for a specific development project. Prototypes of such roadmaps are available at <http://indutech.cust.iaf.nl/roadmaps>. For access username and password need to be requested via email from support@indutech.co.za.

8.1.8 Can a Roadmap, Implemented in a Design Environment, Facilitate the Engineering Development Process?

Chapter 6 shows that the correct project methodology can facilitate the engineering development process. Therefore the remaining question is: "Will the implementation of the project methodology in a roadmap facilitate the engineering development process?"

The work described in this dissertation does not include a case study where the project methodology was implemented in the form of a roadmap in a collaborative environment such as EDENTM, and therefore this question cannot currently be answered with any certainty. However, the roadmap concept has been utilised very successfully to implement an innovative project in the financial industry [44]: "*Roadmaps are an excellent way of supporting innovation teams, but it is important for the team members to agree on the roadmap structure and to take ownership of the roadmap from the start*". It is not unreasonable to expect similar benefits when the approach is applied to engineering development projects.

8.1.9 What Attributes Must a Roadmap Have so That it Can Represent a Design Methodology?

This question was answered under the related Sections 8.1.6 and 8.1.7.

8.1.10 What Attributes Must a Design Environment Have so That a Roadmap Can be Applied to a Development Project?

The author agrees with Pham [100] and Blessing [20] that a design environment should address the aspects listed in Tables 7.4 and 7.5. The most important of these would be the storage of shared information for re-use and an effective facility for searching through the information by means of keywords, phrases and concepts.

8.2 Limitations of This Research

The research summarised in this dissertation is limited by two aspects:

- the chosen scope for the research, and
- the perspective of the author.

As indicated in Section 7.2 each project needs a definition of its scope of work. For this research the scope of work was limited to the engineering and project management aspects of a development project, with emphasis on the methodology required for planning, management and execution of the project. Although it is clear to the author that other issues, such as the legal, contracting and business aspects related to development project are vitally important to its success, the author intentionally excluded these from the scope of this work.

The latter limitation is due to the context surrounding the author, his work experience, the industry he is currently serving and the country he is living in currently. Although the author has worked in various industries and companies of varying sizes, the work described in this dissertation is strongly influenced by the author's occupation in the special automotive industry, with an emphasis on the military, cash-in-transit and law enforcement sectors of that industry. This is clearly shown in the case studies in Chapter 6.

8.3 Applications

The work summarised in this dissertation is intended to be applicable to any engineering development project, irrespective of size or industry. It is the author's opinion that the broad principles discussed in the dissertation should be generally applicable. However, industry (and perhaps country) specific nuances will probably have to be taken into account when applying these methodologies in practice. It was obviously not possible to test this notion in practice within the scope of the work described in this dissertation.

The intent was also to make the work applicable to small companies as much as for large ones. It is for this reason that the implementation of the principles, as discussed in Chapter 7, was not based on elaborate, complicated and specialised computer systems. The software environment, called EDEN™, was chosen as the implementation vehicle because it is so easy to install and use.

8.4 Suggestions for Future Research

The author can identify the following research questions for future work:

- How much benefit can be expected from the implementation of a project methodology by means of a roadmap in a collaborative design environment such as EDEN™?
- How applicable are the suggestions made in this dissertation in other industries and/or other engineering disciplines?
- How should the contextual framework shown in Figure 5.7 be expanded to include the legal, financial and commercial aspects of projects?

8.5 Overall Summary and Conclusions

The research described in this dissertation has been successful in a number of different ways. First and foremost it has achieved the objectives defined at the onset of the project: all the research questions posed in Chapter 1 have been answered. The preceding sections summarise these answers with reference to the specific parts of the dissertation where the particular topic is discussed in more detail.

Secondly, the research has added another stone to the large mosaic that is design science, making the big picture a bit more complete. By viewing the theoretical world of prescriptive models of the development process through the lenses shaped by years of industrial experience, the author provides a new perspective on the subject matter. Rather than forcing a project into the mould of a pre-defined prescriptive process, or proceeding on a gut-feel, experiential level, the author suggests an alternative where the project requirements determine the approach to the development project (see Section 7.3). The development process therefore expands, including the development of the approach to the project (Section 7.5) as well as the development of the required product. By definition the process of developing the approach needs to precede the process of developing the product by a degree. However, it is important to realise that the heuristic nature of the development process, necessitates that both the approach and the process of developing the product be flexible enough to respond to the requirements of the development process as it is executed.

Thirdly, the research has provided a means of implementing these concepts in a simple, elegant manner, applicable to all levels of industry (see Section 7.5). For very small companies it may be appropriate to implement these concepts via a paper-based system. Very large companies, at the other end of the spectrum, may want to integrate these concepts into their elaborate business management systems.

Lastly, the journey through this research has provided the author with a level of insight and understanding that he could not have obtained by any other means. He intends to apply this insight in his own industrial practice and, should the opportunities present themselves, he would want to assist his colleagues in the product development industry either through informal mentorship or formal educational processes.

Appendix A

Selection of Design Methods

Synopsis

Section 2.3 explains the author's point of view on the relationship between product life-cycles, project methodologies and design methods. The author sees a design method as a tool to achieve a very specific outcome given a very specific input. Design methods can therefore be classified according to the outputs, or the needs that they address. Alternatively, given the structured nature of the prescriptive models of the development process, design methods can also be classified according to the phases of the development process.

A.1 Selection of Design Methods by Project Phase

Given that prescriptive models of the development process consists of specific prescribed phases, each of which are intended to produce a very specific outcome, it is possible to suggest methods, that would assist the process of generating this outcome for each phase.

Table A.1: Selection of Design Method by Project Phase according to Green [58]

	Phase	Method
1	Product planning	Project time plan Literature searches Parametric analysis Matrix analysis Brainstorming Integrated product development Competition analysis - Literature, sales reports, trade fairs and exhibitions - SWOT analysis - Features analysis - Peeves analysis - Reverse engineering Market research analysis - Trend studies Needs analysis (customer requirements) - Market feedback mechanisms - Customer interviews and customer questionnaires - Competition benchmarking - Quality function deployment (QFD) matrices Phase Method
2	Clarification	QFD Matrices - Engineering requirements - Competition benchmarking - Engineering targets - Performance specification method - Specification check-lists and questionnaires
3	Concept Concept generation	- Objectives tree and functional decomposition - Brainstorming - Principle of division of tasks - Design catalogues - Literature and patent search results - Function-concept mapping (morphological charts)
4	Evaluation/embodiment	Concept evaluation - Feasibility judgement (gut feel) - Technology readiness assessment - Go/no-go screening (customer requirements) - Value analysis (VA) - Design for manufacture and assembly (DFMA) - Evaluation matrix (relative or weighted objective) - Graphical or physical mock-ups - Design review QFD Matrices
5	Detailed design	Product generation - Component design specifications - Engineering design standards - Produceability engineering
continued...		

...continued	
Phase	Method
	Product evaluation <ul style="list-style-type: none"> - Evaluation matrix (engineering matrix) - Evaluating performance - Analytical, physical and graphical model development - Evaluating costs - Design review - Rapid prototyping - DFMA - Taguchi/robust design - Failure-mode-effect (FMEA) - Value analysis/engineering (VA/VE) - Functional cost analysis - QFD Matrices - Prototyping and testing

Table A.2: Selection of Design Method by Project Phase according to Maffin [86]

Phase	Method
Market	Literature Searches Parametric Analysis Matrix Analysis Competition Analysis <ul style="list-style-type: none"> - Literature, sales reports, trade fairs & exhibitions - SWOT analysis - Reverse engineering Market Research Analysis Need Analysis (Customer Requirements) <ul style="list-style-type: none"> - Market feedback mechanisms - Customer interview & questionnaires - Competition benchmarking - QFD matrices
Specification	QFD Matrices <ul style="list-style-type: none"> - Engineering requirements - Competition benchmarking - Engineering targets Performance Specification Method Specification check-lists & questionnaires
Concept	Concept Generation <ul style="list-style-type: none"> - Objectives tree & functional decomposition - Principle of division of tasks - Design catalogues - Literature and patent search results - Function-concept mapping (morphological charts) Concept Evaluation <ul style="list-style-type: none"> - Feasibility judgement (gut feel) - Technology readiness assessment - Go/no-go screening (customer requirements) - Evaluation matrix (relative or weighted objective) - Graphical and/or physical mock-ups - Design review QFD Matrices
continued...	

...continued	
Phase	Method
Detail	Product Generation - Component design specification - Engineering design standards - Producibility engineering (materials, form & process) - Evaluation matrix (engineering requirements) - Evaluating performance (analytical, physical or graphical models) - Evaluating cost - Design review DFMEA Taguchi / Robust Design FMEA Value Engineering Functional Cost Analysis QFD Matrices Prototyping & Testing

A.2 Selection of Design Methods According to Need

Rather than selecting the method according to the phase of the project, one can categorise the methods according to the need that they satisfy, i.e. the results they produce. The tables presented in Section A.1 were generated by Martin and Veron [90], Stetter [116], and the VDI [125]. For the VDI matrices the colour codes have the following meaning:

- White - the design method is not expected to have any application in the particular phases of the development life-cycle.
- Grey - there could be some instances where the design method could be applicable to the particular phase of the development process.
- Black - the design method is very applicable for the particular phase of the development process.

Table A.3: Selection of Design Methods According to Matrin [90]

Tools	Functional module	Structural module	Manufacturing module
for creativity	SADT, FAST	brainstorming, TRIZ & DFM, DFA	
for validation	FMEA, value analysis, cost estimation	FEM, strength of materials, kinematics simulations tools, tolerances analysis	Simulation tools for each manufacturing process: machining, forging, casting
for representation	CAD software, sketches, drawings	kinematics diagram	Simulation tools for each manufacturing process: machining, forging, casting

Table A.4: Selection of Design Methods According to Stetter [116]

Function	Explanation
Goal setting	Gaining information about the desired characteristics of a product.
Analysing	Gaining information about the actual characteristics of a product.
Evaluating	Gaining information about the relative performance of systems.
Deciding	Selecting, e.g. Alternative solutions, courses of action.
Forecasting	Gaining information about possible future developments.
Planning	Anticipating future developments and, based upon this, defining future actions.
Representing	Depicting a system, for example, for the purpose of presenting.
Innovating	Inventing fundamentally new systems, with a high degree of innovation.
Developing	Transforming characteristics of systems qualitatively, with a medium degree of innovation.
Optimising	Transforming characteristics of systems qualitatively, with a low degree of innovation.

	Phases of product development					
	Product planning	Problem clarification	Conceptualisation	Embodiment	Detailing	Preparation for production
Trend studies	█					
Market research	█					
Benchmarking			█			
Integrated product development	█	█	█			█
Total quality management (TQM)	█	█				█
Quality function deployment (QFD)	█			█	█	█
Brainstorming	█	█				
TIPS	█		█	█		
Virtual reality		█	█			
Iteration		█	█	█	█	█
Recursive decomposition			█		█	
Dialog method			█	█	█	
Mathematical Analysis		█	█	█	█	
Interpreting mathematical functions			█			
Transformation process		█				
Function structure			█			
Morphological matrix			█	█		
Design catalogues			█	█	█	
Axiomatic design			█	█	█	
Concept selection			█	█	█	
Organ structure			█	█	█	█
Value analysis / engineering	█		█	█	█	█
Cost pre-calculation			█	█	█	█
Concurrent / simultaneous engineering			█	█	█	█
Design of experiments (Taguchi)			█	█	█	█
Critical path planning (CPM, PERT)			█			█
Fault tree analysis			█	█	█	█
FMECA			█	█	█	█
Zero defects			█	█	█	█
Hitachi method				█	█	█
Boothroyd-Dewhurst (DFMA)				█	█	█
Lucas method				█	█	█
Artificial intelligence				█	█	█
Computer aided draughting (CAD)				█	█	█
Computer aided manufacture (CAM)				█	█	█
Rapid prototyping				█	█	█
CIM					█	█
Statistical process control (SPC)					█	█

Figure A.1: Eder's Method Matrix [46]

		Phases of product development							
		Task clarification & definition	Function structures	Solution Principles	Realizable modules	Layouts of key modules	Overall layout	Product documentation	
Methods for Analysis and Goal-setting		Market analysis	■				■	■	
		Prognosis methods	■		■				
		Competitors analysis	■	■	■	■	■	■	
		“Fremderzeugnisanalyse”	■	■	■	■	■	■	
		SWOT analysis of own organisation	■	■	■	■	■	■	
		Product planning	■	■			■	■	
		Problem analysis	■	■	■	■	■	■	
		Goal definition	■	■	■	■	■	■	
	Functional Description		Verbal – noun & verb combination	■	■	■			
			Physical elementary functions	■	■	■			
			Mathematical modelling		■	■	■	■	■
			Symbols or other representations		■	■	■		
	Functional Structuring		Functional Hierarchy (function tree)	■	■	■			
			Function net		■	■	■		
			Logical function path		■	■	■		
	Qualification		Mathematical models		■		■		
			User requirements specification (URS)	■	■	■	■	■	■
		Technical quality					■	■	
		Simulation or “similarity calculations”			■		■	■	
		Finite element method (FEM)					■	■	

Figure A.2: Methods for Analysis and Goal-setting Recommended by VDI 2221 [125]

			Phases of product development						
			Task clarification & definition	Function structures	Solution Principles	Realizable modules	Layouts of key modules	Overall layout	Product documentation
Methods for Generation of Ideas	Heuristic problem solving	Brainstorming		■	■				
		Method 66		■	■				
		Method 635		■	■				
		Provocation		■					
		Idea – Delphi	■						
	Discursive problem solving	Morphological Box		■	■	■	■		
		Simple Morphology		■	■	■	■		
		Property lists	■	■	■				
		Systematic variation		■	■	■	■		
		Design catalogues	■	■	■	■	■		
		Design guidelines		■	■	■	■	■	■
		Modules		■	■	■	■	■	■
		Drawings, Sketches	■	■	■	■	■	■	■
		“Synektik”	■	■	■				
		“Gemilderte Synektik”		■	■	■	■	■	■

Figure A.3: Methods for Generation of Ideas Recommended by VDI 2221 [125]

		Phases of product development						
		Task clarification & definition	Function structures	Solution Principles	Realizable modules	Layouts of key modules	Overall layout	Product documentation
Methods for Cost Estimation	Calculation Methods	Cost Comparisons						
		Complete Costing						
		Recovery Costing						
		Recovery Costing +						
	Methods to estimate product costs early	Minimum Costing						
		Early costs estimates						
		Catalogue of relative costs						
		Short calculations						
		Cost growth calculations						
		Similarity calculations						
		Calculation of costs / value						
		Investment calculation						
		“Bemessungslehre”						

Figure A.4: Methods for Cost Estimation Recommended by VDI 2221 [125]

		Phases of product development						
		Task clarification & definition	Function structures	Solution Principles	Realizable modules	Layouts of key modules	Overall layout	Product documentation
Methods for Assessment and Decision	Rough Categorisation							
	Selection criteria							
	Pro's, Con's & Costs							
	Dual Comparison							
	Calculation of costs / value							
	Value System							
	Technical / Cost Analysis							
	Failure mode analysis							
	Failure tree analysis							
	Expert judgement							
	Operations research							
	Decision criteria matrix							
	Decision tree							

Figure A.5: Methods for Assessment and Decision Making Recommended by VDI 2221 [125]

		Phases of product development						
		Task clarification & definition	Function structures	Solution Principles	Realizable modules	Layouts of key modules	Overall layout	Product documentation
Integrated Methods	System method	■	■	■	■	■	■	■
	Value analysis	■	■	■	■	■	■	■
	Development methods	■	■	■	■	■	■	■
	CAD / CAM / CIM	■	■	■	■	■	■	■
	Computer graphics	■	■	■	■	■	■	■
	6-Level method	■	■	■	■	■	■	■
	Teamwork	■	■	■	■	■	■	■
	Management by ...	■	■	■	■	■	■	■
	Project management	■	■	■	■	■	■	■
	Planning methods	■	■	■	■	■	■	■
	PERT charts	■	■	■	■	■	■	■
	Capacity planning	■	■	■	■	■	■	■
	Investment planning	■	■	■	■	■	■	■
	Optimisation calculations	■	■	■	■	■	■	■
	Industrial Design	■	■	■	■	■	■	■

Figure A.6: Integrated Methods Recommended by VDI 2221 [125]

Appendix B

An Overview of Various Prescriptive Models of the Development Process

Synopsis

In support of Section [3.1](#), which discusses prescriptive models of the development process, the following sections provide a brief summary for each of the more well-known models.

B.1 Cross's Model of the Design Process

Broadly the process starts with a statement of the requirements of the article to be designed, followed by a concept generation phase. Once concepts have been evaluated, a concept is chosen, and the article is defined in detail in the detail design phase. The output is a complete description of the article by means of engineering drawings, material specifications and production process definition. A typical representation of this process is shown in Figure B.1 [32]. Note the feedback loop between the evaluation process and the generation process, indicating that the development process is iterative.

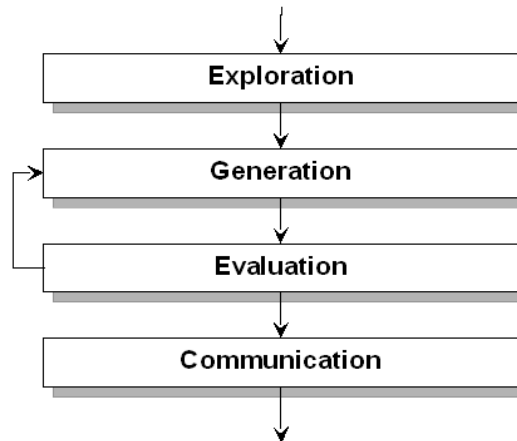


Figure B.1: The Model of the Development Process by Cross [32]

B.2 French's Model of the Design Process

Cross reports that French described the design process in more detail, as shown in Figure B.2. In "Engineering Design Methods" [32], French is quoted as follows:

<i>Problem Analysis</i>	<p><i>The analysis of the problem is a small but important part of the overall process. The output is a statement of the problem, and this can have three elements,</i></p> <ul style="list-style-type: none"> • <i>A statement of the design problem proper.</i> • <i>Limitations placed upon the solution, e.g. codes of practice, statutory requirements, customer standards, date of completion, etc...</i> • <i>The criterion of excellence to be worked to.</i>
<i>Conceptual design</i>	<p><i>This phase ... takes the statement of the problem and generates broad solutions to it in the form of schemes. It is the phase that makes the greatest demands on the designer, and where there is the most scope for striking improvements. It is the phase where engineering, science, practical knowledge, production methods and commercial aspects need to be brought together, and where the most important decisions are taken.</i></p>
<i>Embodiment of schemes</i>	<p><i>In this phase the schemes are worked up in greater detail and, if there is more than one, a final choice between them is made. The end product is usually a set of general arrangement drawings. There is (or should be) a great deal of feedback from this phase to the conceptual design phase.</i></p>
<i>Detailing</i>	<p><i>This is the last phase, in which a very large number of small but essential points remain to be decided. The quality of this work must be good, otherwise delay and expenses or even failure will be incurred; computers are already reducing the drudgery of skilled and patient work, reducing the chance of errors.</i></p>

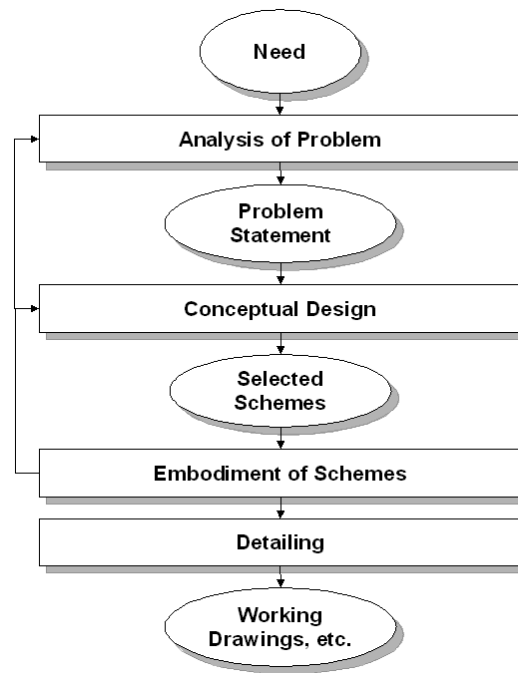


Figure B.2: French's Model of the Product Development Process [32]

B.3 Archer's Model of the Design Process

Archer's model of the design process, shown in Figure B.3, is described by Cross [32] as consisting of the following tasks:

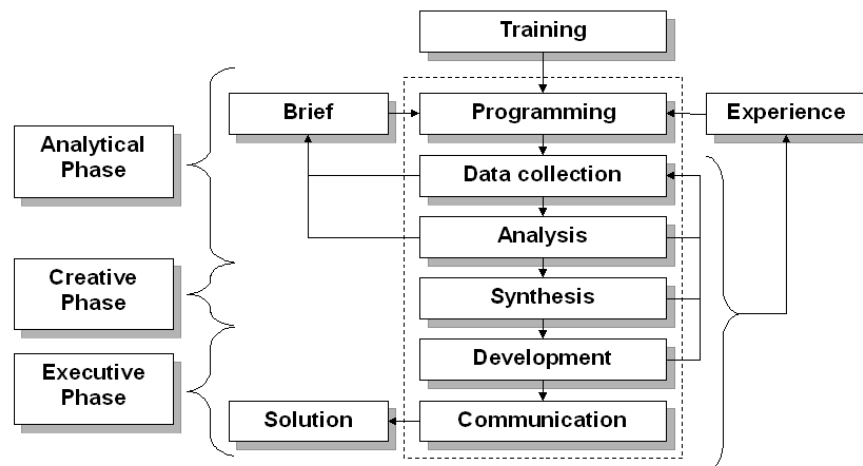


Figure B.3: Archer's Model of the Product Development Process [32]

- | | | |
|-----------------|---|--|
| Programming | - | establish crucial issues; propose a course of action |
| Data collection | - | collect, classify and store data |
| Analysis | - | identify sub-problems; prepare performance specifications; reappraised proposed programme and estimate |
| Synthesis | - | prepare outline design proposals |
| Development | - | develop prototype design(s); prepare and execute validation studies |
| Communication | - | prepare manufacturing documentation |

According to Archer these activities can be summarised into an analytical phase, followed by a creative phase and an executive phase. The model of Archer also includes feedback loops explicitly to indicate the iterative nature of the design process. Archer indicates, in the graphical representation of the process, that training and experience play a vital role in the engineering development.

B.4 Development Model for Large Infrastructure Projects

The company, Instrument Data Communications, has summarised the principles of project management [36], with emphasis on the development of infrastructure projects. Figure B.4 shows the proposed model for construction projects.

The various phases are defined as follows[36]:

- Pre-feasibility* - Identification of needs, and preliminary validation of concept options.
- Feasibility* - Detailed investigation of feasibility, including preliminary brief, project estimate and investment analysis.
- Planning* - Detailed definition of the project with respect to scope, organisation, budget, and schedule, together with definition of all control procedures.
- Implementation* - The execution of the scoped project. The components of this phase will depend upon the nature of the project.
- Handover* - Passing the facility into the control of the principle.
- Close-out* - Archiving of the project records, and dis-establishment of the project organisation.

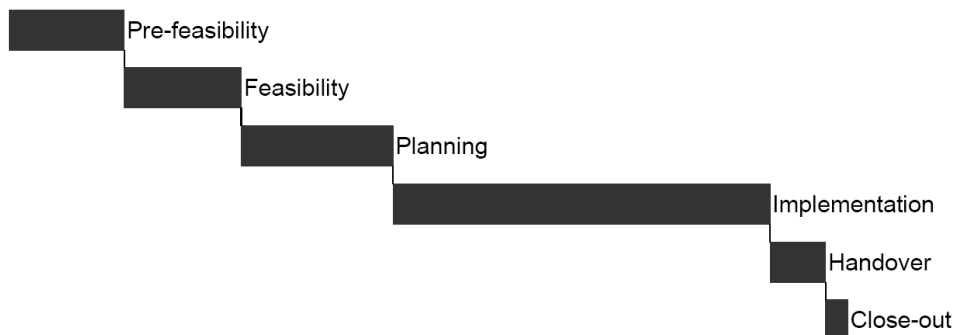


Figure B.4: Development Process for a Construction Project According to the IDC [36]

B.5 Suireg's Model of the Design Process

Figure B.5 shows a graphical representation of Suireg's design process [117]. The first phase is the identification of a need, followed by a thorough functional investigation, resulting in a "formal statement of the design problem and its objective" [117]. The next phase consists of the generation and selection of alternatives. Once the most feasible alternative has been selected, the designer develops a model, to be used for analysis and optimisation. This model can be an analog model (where "the physical phenomenon is replaced by an equivalent, which is more convenient to investigate" [117]), an iconic model (these days it would usually be a CAD model) or mathematical model. The next phase is called system optimisation, where the design parameters are selected, "which produce the best possible design" [117]. Following this phase the design is implemented resulting in a physical object that can be tested and evaluated. Results from the test and evaluation are fed back into the process, initiating a second design iteration, should the testing reveal that the original need has not been met successfully.

B.6 The Design Process According to Pahl and Beitz

In their book [97] Pahl and Beitz, describe in detail their approach to structuring the product development process. The design process requires that facts and relationships amongst information be "consciously anal-

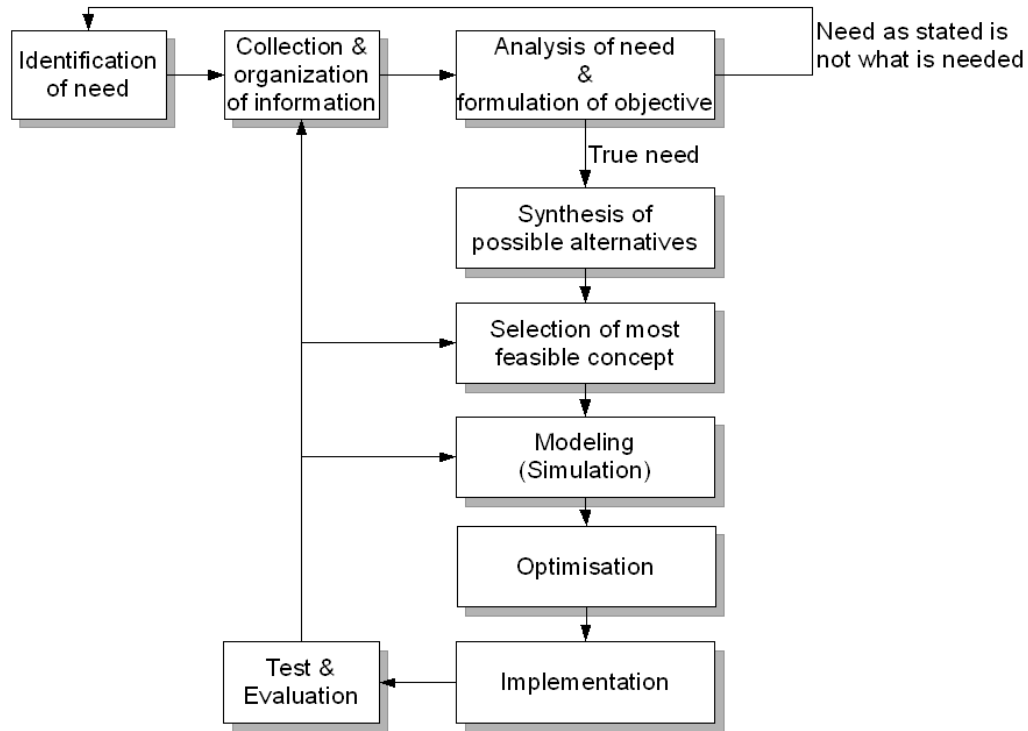


Figure B.5: Suireg's Model of the Product Development Process [117]

used, varied, combined in new ways, checked, rejected and considered further". "For knowledge to be easily retrieved and combined, it is thought that an ordered and logical structure of factual knowledge in the mind of the solver is decisive". To improve the efficiency and effectiveness of these thinking processes during product development it is advisable to transform "aimless and unconscious procedures" as well as "disorderly and fantasy-charged preconscious procedures into a conscious or deliberate approach". This is the aim of the "procedural plans" described by the authors in [97].

Figure B.6 shows the process schematically. The diagram refers to the steps of the process as described in the VDI guideline 2221 [125], "to which the authors [Pahl and Beitz] contributed substantially", and their book [97] provides "an extensive description of [the] flow of work" during the product design process. The author of this dissertation therefore considers the structured approach of Pahl and Beitz to be identical to the VDI guideline, and has used the detail information in [97] to describe the VDI process model in section B.7.

B.7 The Model of the Development Process as Defined by VDI 2221

Figure B.7 shows the design process schematically as described in the VDI guideline 2221 [125].

The first step in the product development process is the clarification of the task at hand. This step documents the context of the project, as well as the scope and purpose of the development to be carried out. The request for a new product can originate from many sources, such as formal product planning process or an order from a customer. Irrespective of where the request originated from, there is a need to clarify the task, its requirements and constraints as well as the desired time-scales. The output of this step of the process is the requirements list (or design specification), which will be updated during the following phases, since the design process can lead to the recognition that new requirements must be added or existing requirements changed. It is usual practice to 'freeze' the requirements list at some stage of the design process, to avoid increasing the development time by changes to the requirements.

The next step is conceptual design, which "determines the principle solution" by analysing the required functions of the new product and their interrelationships [96]. From this analysis various solution principles can be developed. Often rough dimensional layouts are used to assess the working structure. The solution variants

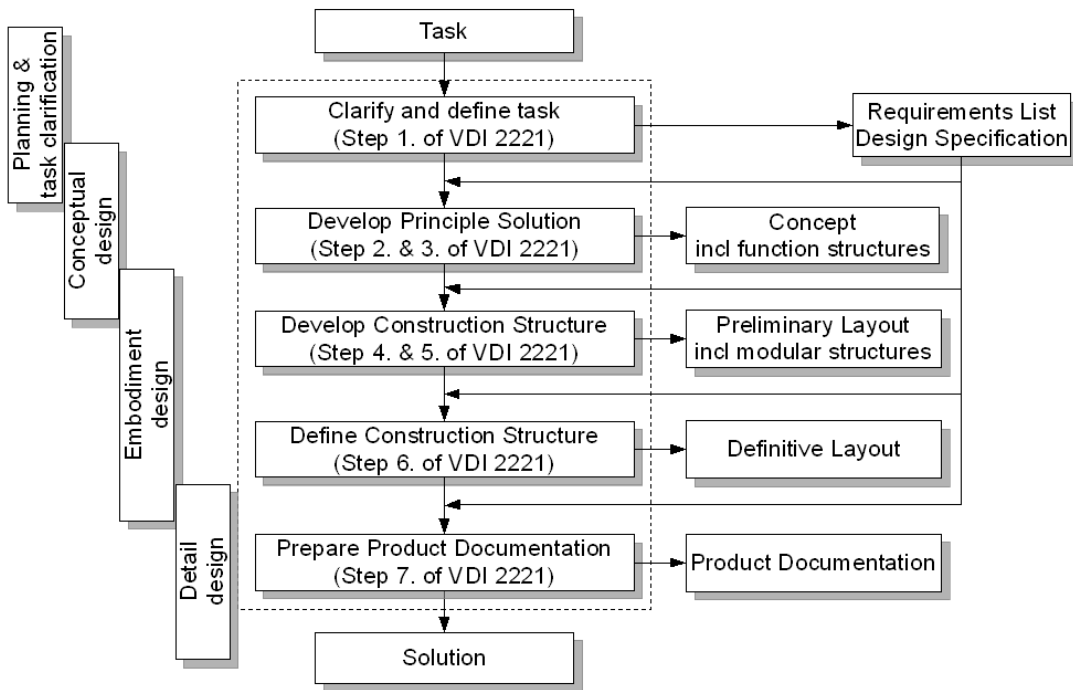


Figure B.6: The Product Development Process According to Pahl and Beitz [97]

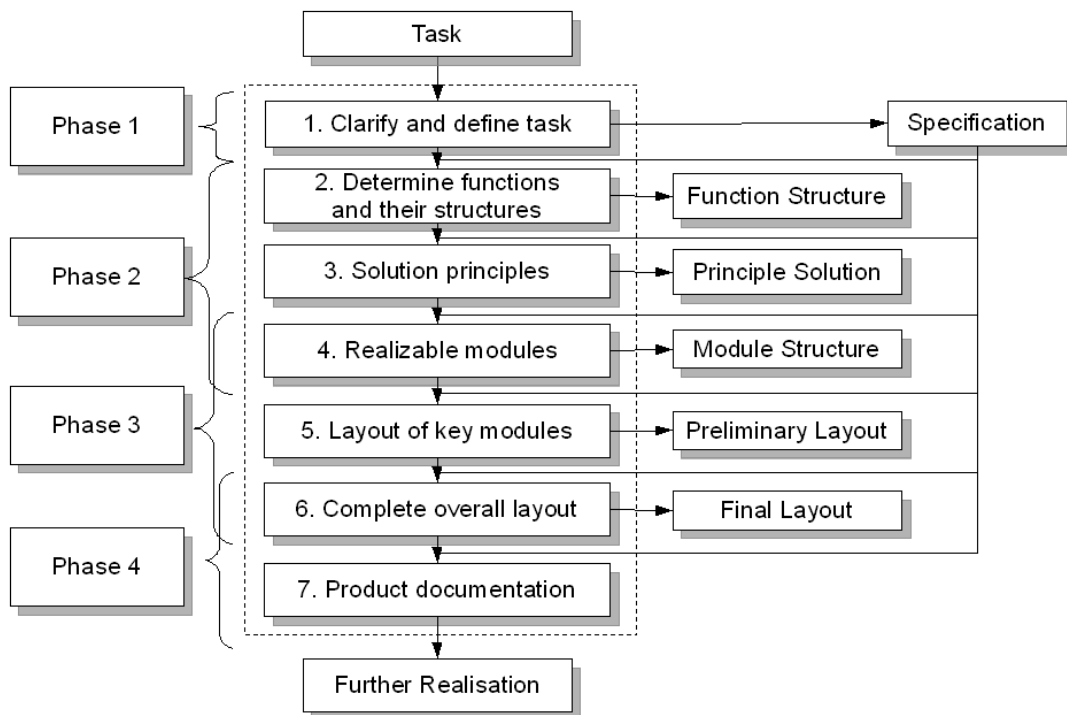


Figure B.7: The Product Development Process According to the VDI [125]

are then evaluated by comparing them to the requirements list, eliminating those that do not conform to the requirements. The rest are then assessed mainly on a technical level, although rough cost criteria should also play a role.

Once the principle solution has been selected, embodiment design can commence. In this phase the principle selected in the previous phase is given form and dimension. In this step the concept can be divided into modules. This is especially useful when complex products are developed by large design teams, since each sub-team can be allocated the task of designing a module or sub-system. Interfaces between modules must be defined to ensure that they can be integrated to form the complete product. Other reasons to divide the product into modules or sub-systems includes the need for re-cycling, re-use of module in other products, ease of manufacturing or service and maintenance. The next step is to develop preliminary layouts which can be evaluated according to technical and economic criteria, and the definitive layout is selected as the output of this phase. The definition of the preliminary layout should include the following [96]:

- the selection of material, at least in board terms,
- the establishment of the main dimensions, determined by the principle solution,
- the checking of the geometric fit of all components,
- the selection of solutions for sub-problems,
- the selection of the manufacturing methods, at least in broad terms, and
- the method of assembly (and disassembly).

Technological and economic aspects will have to be taken into account during the establishment of the above.

Using the definitive layout as a basis, the detail design phase *"is the phase of the design process in which the arrangement, forms, dimensions and surface properties of all the individual parts are finally laid down, the materials specified, production possibilities assessed, costs estimated and all the drawings and other production documents produced."*

One of the underlying principles on which the VDI as well as the Pahl and Beitz models of the development process are based is the assumption that both the problem and the solution can be divided into a hierarchical structure of sub-problems and sub-solutions respectively, as illustrated in Figure 3.1. The design process is therefore one of disassembling the problem (the design objective) into all its sub-problems. Each sub-problem is solved, followed by an assembly process to obtain the solution to the original problem. This process is said to *"facilitate the recognition of sub-problems, the consideration of the complete problem through the identification of the structures and relationships, the systemisation, the development of alternatives, the use of existing and proven sub-solutions and the implementation of rational organisational division of labour."*

B.8 Ullman's Model of the Development Process

In his book, "The Mechanical Design Process", Ullman [123] describes the engineering development process as consisting of five phases as shown in Figure B.8. In the first the project team is assembled and the project is planned. The output of this phase is an *"ideal flow chart"* of the project's activities, including the allocation of the company's resources (personnel, money and equipment) to this process. This is followed by the generation of a product specification in the second phase. The information to be generated in this phase includes the identification of the new product's customers and their requirements, as well as a list of features and performance targets. The third phase is a typical conceptual design phase, where functional decomposition is used to generate concepts, which are evaluated in order to choose the most promising concept for development. Ullman calls the fourth phase "product development" and it incorporates the embodiment and detail design stages of the other models. In this phase the product's form is generated from the function defined in the previous phase. Also material and the manufacturing process is selected. This is done in an iterative manner, while considering the product's requirements as defined in the second phase. The fifth phase concentrates on product support, in production, at the vendors and the customer.

Design reviews are mentioned specifically by Ullman, as evaluation gates at the end of each phase to ensure that previous phase is complete and has achieved what was planned.

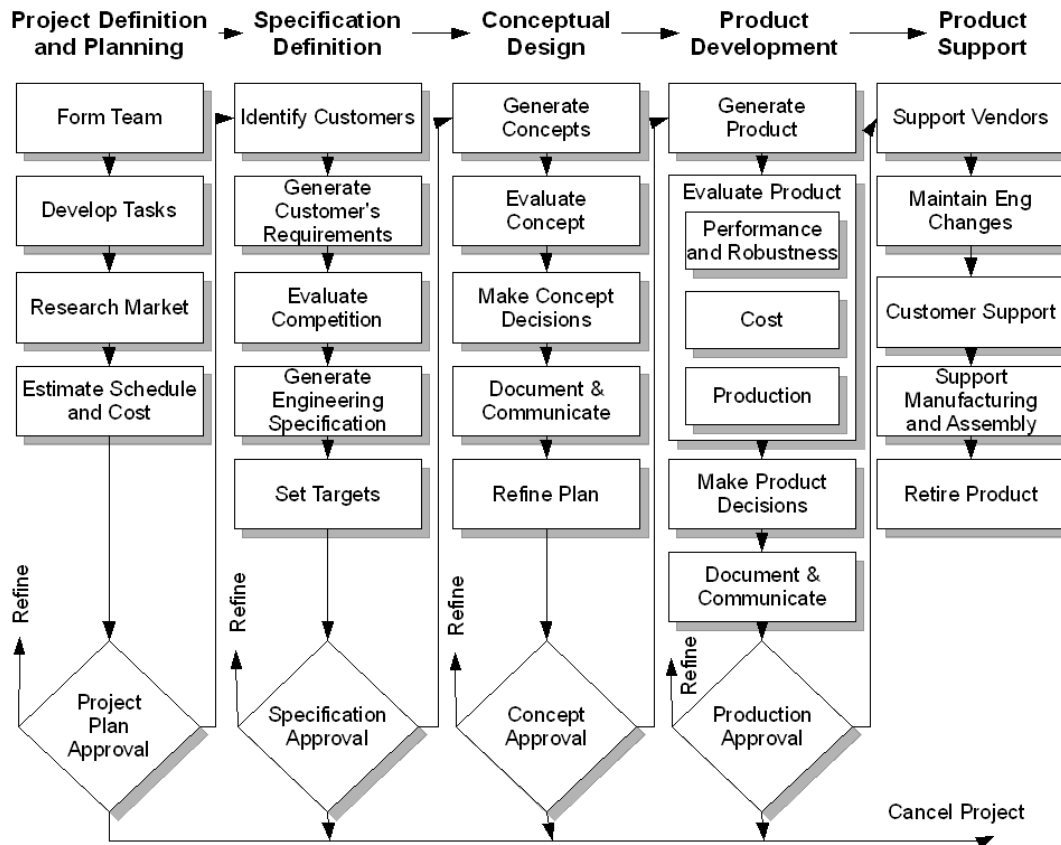


Figure B.8: The Mechanical Design Process According to Ullman [123]

Similar to the models of Pahl and Beitz as well as VDI, Ullman also advocates the use of an hierarchical problem structure to drive the solution process. The functional decomposition is used to define the sub-problems. Components and assemblies are then defined to solve each sub-problem by providing the required functions. The solution of the overall problem is obtained by assembling all these sub-solutions into the complete product.

Ullman [123] also defines different types of design processes:

- Selection design - *"involves the selection of items from a list of similar items"*, such as a catalogue. Example: selection of a rolling contact bearing.
- Configuration design - requires existing components to be assembled (packaged) to make up the completed product. Example: designing a housing for a desk-top PC.
- Parametric design - *"involves finding values for features that characterise the object being studied"*. Example: determining the size of the tank to be mounted on a truck chassis, given restrictions with regard to weight and size.
- Original design - *"requires the development of a process, assembly or component not previously in existence: or that information is not available to the designer."* Example: at the beginning of the 20th century, designing the shape of a wing for an aircraft that is heavier than air.
- Redesign - is the *"modification of an existing product to meet new requirements"*. Example: add a sensing element and an actuator to an electric kettle so that it will switch off automatically when the water is boiling.

B.9 The Systems Engineering Approach to the Development Process

In their book, "Systems Engineering and Analysis", Blanchard and Fabrycky [19] defines systems engineering as follows:

"An interdisciplinary approach encompassing the entire technical effort to evolve and verify an integrated and life-cycle balanced set of system, people, product, and process solutions that satisfy customer needs. System engineering encompasses

- *the technical efforts related to development, manufacturing, verification, deployment, operations, support, disposal of, and user training for, system products and processes;*
- *the definition and management of the system configuration;*
- *the translation of the system definition into work breakdown structures; and*
- *development of information for management decision making."*

With the advent of more and more complex products, such as automobiles, large ships for commercial and warfare purposes, commercial and military aeroplanes to name a few, it was realised that complex systems are more than the sum of their parts, i.e. local optimisation of the components and sub-systems does not provide optimised system. Also the effort required to manage the development process increases with increased product complexity, due to the increasing number of interrelationships between the various components of the system, and also due to the increased number of people of various disciplines required for the development of these complex systems. Blanchard and Fabrycky [19] summarises the focus of system engineering as follows:

- *"A top-down approach that views the system as a whole" providing "the necessary overview and understanding of how these components effectively fit together".*
- *"A life-cycle orientation that addresses all phases to include system design and development, production and/or construction, distribution, operation, maintenance and support, retirement, phaseout and disposal".*
- *"A better more complete effort regarding the initial definition of system requirements, relating these requirements to specific design criteria and the follow-on analysis effort to ensure the effectiveness of early decision making in the design process. The true system requirements need to be well defined and specified, and the traceability of these requirements from the system level downwards needs to be visible".*
- *"An interdisciplinary or team approach throughout the system design and development process to ensure that all design objectives are addressed in an effective and efficient manner".*

The systems engineering model for the development process is shown in Figure B.9. Note that for the purposes of this dissertation, we are interested in the process up to the point where the product baseline has been established, i.e. the product is ready to be industrialised for production.

Note that the model indicates an iterative process within each development phase. This feedback mechanism is described in more detail by Blanchard [19] and Fabrycky, and is shown in Figure B.10.

During the concept phase the need is identified, together with the new product's requirements in terms of operations, maintenance and support (including technical performance measures) to fulfil the need. The requirements are used to generate the functional characteristics of the new product, which are in turn used to generate a systems specification (type A). This constitutes the functional baseline of the new product and will be used to evaluate the design and prototype during the later stages of this process. In terms of planning activities the Systems Engineering Management Plan (SEMP) and the Test and Evaluation Master Plan (TEMP) are developed. The phase ends with the Conceptual Design Review where *"design information is released and reviewed for compliance with the system requirements"* [19].

The preliminary design phase takes the Functional Baseline as defined for the complete system in the concept phase, and extends it to the sub-system, assembly and component levels of the product by analysing the individual requirements and allocating the corresponding specifications (type B or C). A number of concepts

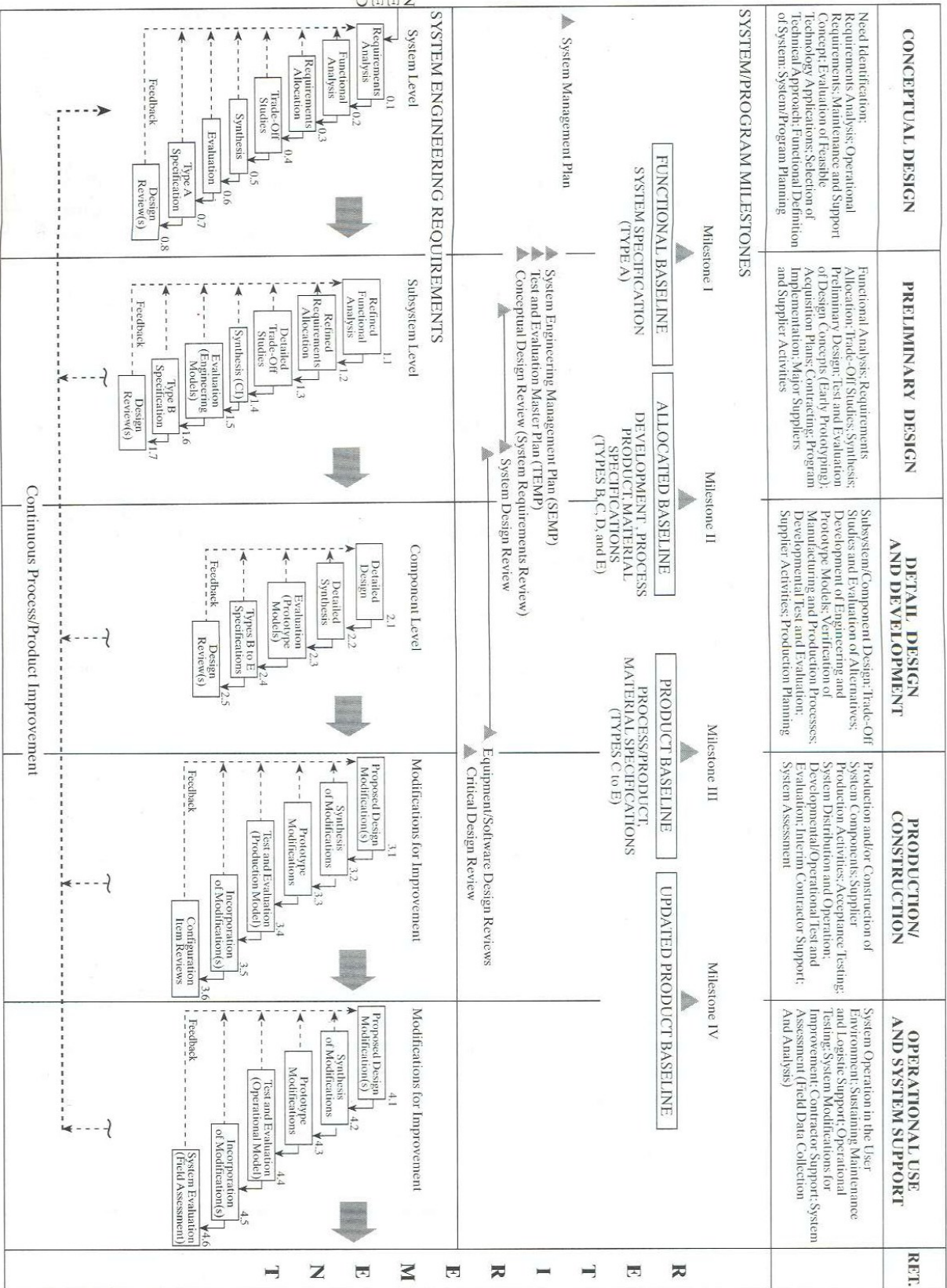


Figure B.9: The Systems Engineering Product Acquisition Process [19]

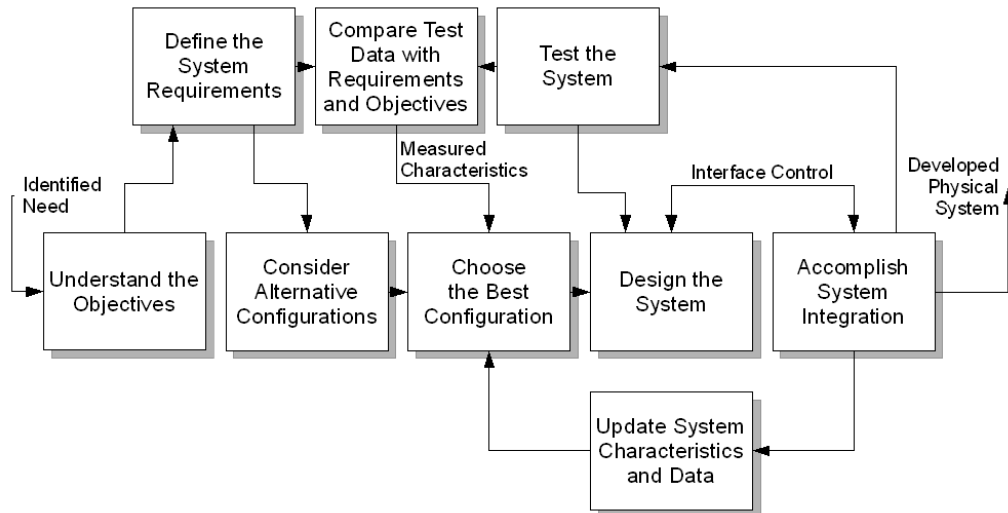


Figure B.10: The Feedback in the Systems Engineering Process [19]

for sub-systems and assemblies are usually generated, and subsequently evaluated by means of trade-off studies. The final concepts and their information is recorded in the Allocated Baseline. The design plan, as well as the test and evaluation plan, are developed in more detail, including strategies for the design, test and evaluation of sub-systems, assemblies, and even critical components. Acquisition plans are drawn up, major suppliers considered and contracted. For complex systems this phase will incorporate a number of design reviews to evaluate the work performed on sub-systems and critical assemblies and components. The final Systems Design Review is the gateway towards the next phase.

The next phase is the detail design phase. Having defined the overall system and the sub-systems and major assemblies in the Functional and Allocated Baselines during the concept and preliminary design phases respectively, one can now proceed to develop the individual components that make up the sub-systems and assemblies. These are described by product specification (type C) for off-the-shelf items, or by process and material specifications (type D and E respectively) for manufactured items. Depending on the risks and complexity evaluation models or prototypes of sub-systems and assemblies are built and evaluated. The phase ends with the building and evaluation of the complete system. The verified definition of the system constitutes the output of the detail design phase: the Product Baseline.

B.9.1 Requirements Engineering

Within the systems engineering fraternity there exists a school of thought which concentrates on the establishment, traceability and validation of design requirements. Thompson [120] describes requirements engineering to consist of two parts:

- Requirements development for the capture, analysis and structuring of requirements, and
- Requirements management for configuration management, including change control and traceability.

In this regard, Sutinen, et al. [118], have concentrated on developing a product modelling system, which allows the designer to quantitatively trace system requirements back to the components that effect the requirement and vice versa.

B.10 Concurrent Engineering

According to Tomiyama [122] concurrent engineering does not present a new model of the development process, but rather "*advocates sub-processes of design and development*" to be "*performed concurrently*" as opposed to sequentially in the conventional development model. To this end Tomiyama identifies the following critical issues:

- "building a cross functional team exclusively focused on a target product,
- facilitating mutual communication among participants of the cross-functional team, and
- bringing up considerations of later stages, such as manufacturing, purchasing, operation, maintenance and recycling to the discussion table."

Ha and Porteus [61] indicate that the interaction between the product design team and the process design team can be facilitated by a number of progress reviews. These reviews provide two major benefits:

- The "review provides information that enables the process designers to work productively on advanced stages of the process design while the product is still evolving", and
- from a quality control perspective, the "review assesses the manufacturability of the current design and determines whether redesign is called for, before getting too far along".

Both of these benefits are aimed at shortening the overall time span of the project.

B.11 Integrated Product Development

In their book, "Integrated Product Development", Andreasen and Hein [9] describe integrated product development as "an idealised model for product development, which is integrated in terms of creation of market, product and production, and which clarifies the integration between project and management". The basic idea is that a business consists of three elements: the market, the product and the production. In order to create and sustain a healthy business it is necessary to manage these three inter-related elements harmoniously.

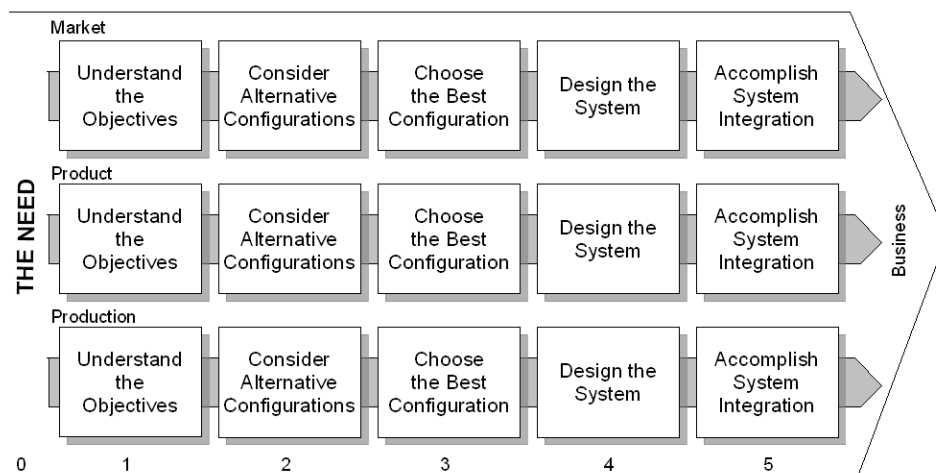


Figure B.11: Process of Integrated Product Development by Andreasen and Hein [9]

The process of integrated product development is divided into six phases as shown in Figure B.11. Phase 0 is used to recognise or find a need that could be filled by a new product, which would be the basis of a new business opportunity. This phase generates "a series of diffuse perceptions of needs". During phase 1 these perceptions are developed a little further. One would investigate how large the potential market would be for the perceived product. The product itself would be described in some more detail by defining who would use, how it would be used to meet the identified need. For the process strand one would investigate potential manufacturing methods. During the next phase, phase 2, the market strand investigates the situation in which the new product will be used, taking the whole life-cycle into account. In the product strand the product's functions are defined, sub-problems identified and the solutions determined in principle. For the production strand the product's structure is determined and principle production methods identified. The market strand in phase 3 clarifies the product specification, determines sales channels, pricing policies and consequential sales volumes. Preliminary product design determines the product's form, including methods of production, the production strand determines manufacturing and assembly methods. During phase 4 everything is prepared for production. The market strand produces detailed plans with regard to sales and advertisements.

For the product and productions strands the workshop costs are finalised, the production system is defined and established. Production readiness is demonstrated by means of a pre-production run. In the execution phase, phase 5, sales and distribution are initiated. In the product strand the product is improved and updated according to the feedback from the market, while the production strand ensures continued production and support of the product.

Andreasen and Hein [9] describe the design process within the integrated product development process illustrated above, as one which moves from the abstract to the concrete, illuminating possibilities as the process progresses, as illustrated in figure 2.1. In detail the design process is structured as shown in Figure B.12.

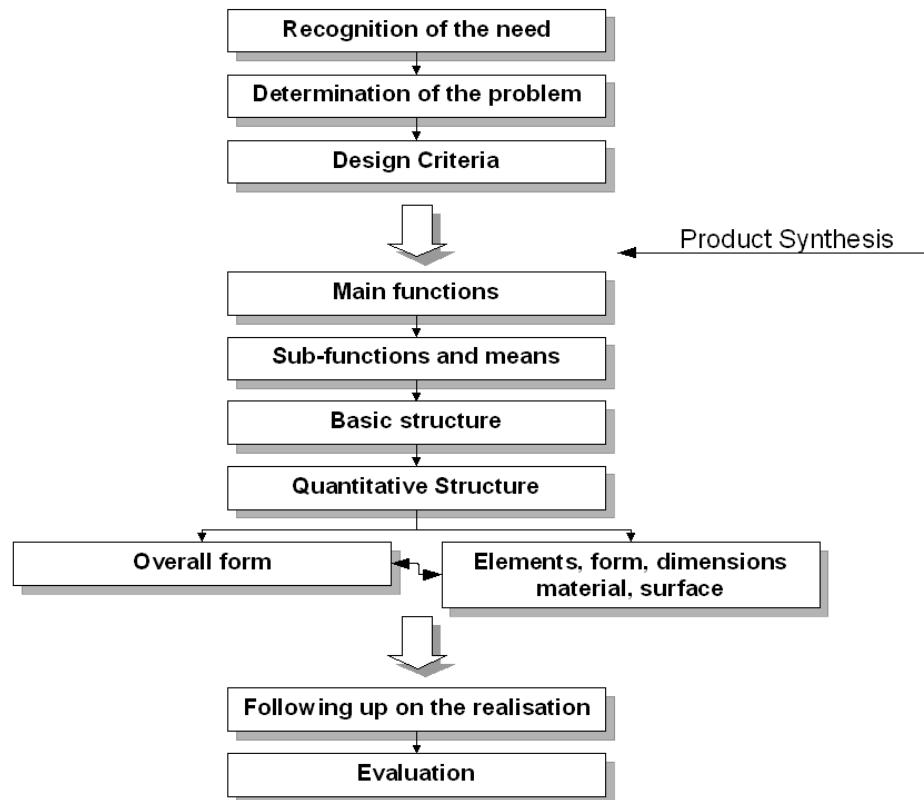


Figure B.12: The Product Synthesis Process According to Andreasen and Hein [9]

Lindemann et al. [82] describes integrated product development as "a holistic approach to overcome the problems that arise in product development because of the division of labour". Integrated product development, although "based in Simultaneous Engineering, goes beyond Simultaneous Engineering with regard to the integration level", since the members of the multi-disciplinary development team "not only consult themselves while they are working simultaneously on their tasks, but exchange interconnected intermediate results in a continuous interplay".

B.12 Pugh's Core-based Model

According to the Design Council of the UK [5] Pugh [102] proposed the core-based model: "a process of iteration, testing and evaluation", as shown on the right hand side in Figure B.13, surrounding "a design core which consists of activities, which he claimed were imperative for any design activity irrespective of domain". Any of the phases of Pugh's core-based model could contain this iteration. According to the Design Council "Pugh focused on a concept called total design which he believes incorporates everything from the early identification of market and user need through to the selling of a product that meets that need. Furthermore, around the core-based model, there would be a plethora of other impacting business and design activities present".

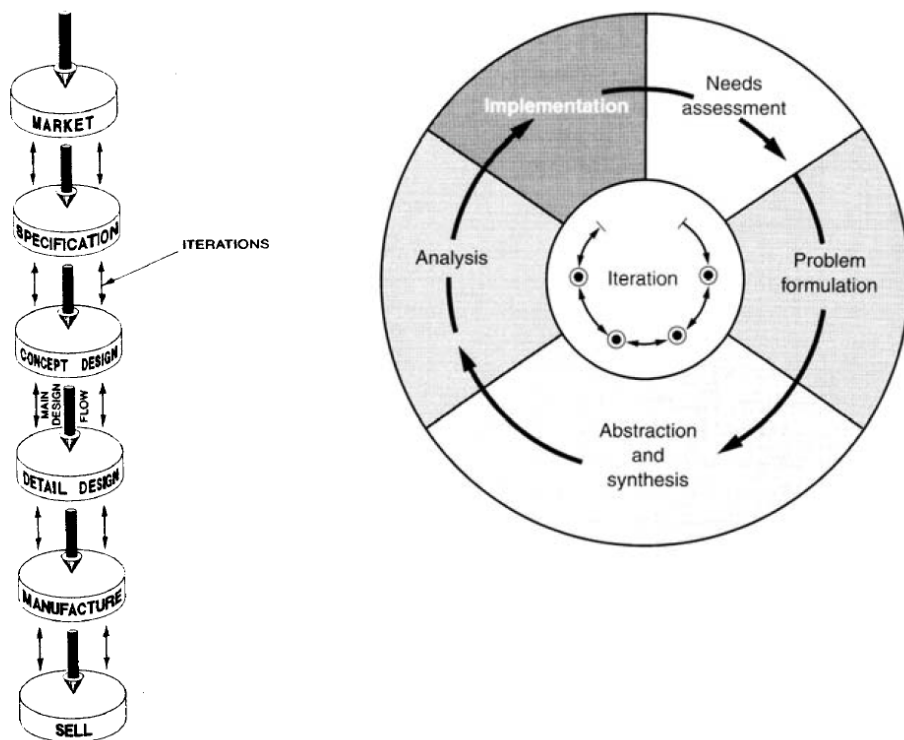


Figure B.13: Pugh's Core-based Model [102]

Appendix C

An Overview of Various Descriptive Models of the Development Process

Synopsis

In support of Section [3.3](#), which discusses descriptive models of the development process, the following sections provide a brief summary for each of the more well-known models.

C.1 Models of the Design Process by Hillier, Drake and March

According to Cross [32] March used the work of C.S. Peirce [98] to describe the concept of abductive reasoning, which "*suggests that something may be*", while "*deduction proves that something must be and induction that something actually is*".

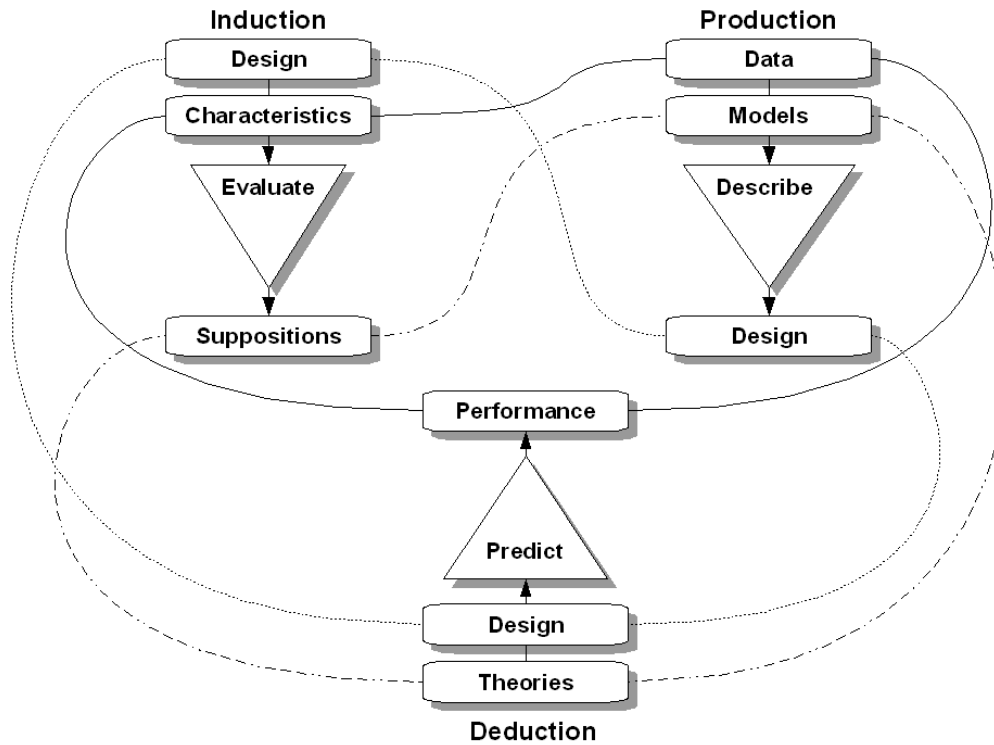


Figure C.1: March's Model of the Product Development Process [32]

During synthesis activities in the design process the designer used abductive reasoning (March prefers to call it "productive" reasoning) to create a design proposal based on presuppositions or protomodels. This proposal is then analysed using deductive reasoning to determine the performance characteristics of the design. The inductive evaluation of the characteristics then leads to further refinements of design and the cycle is repeated as shown in Figure C.1. The process is therefore based on the use of pre-structure as initial solution, which is refined by a conjecture-analysis evolutionary process.

C.2 Model by Hybs and Gero

Hybs and Gero [70] maintain that no design is completely novel, since all design work is based on previous experiences. It could be argued that even an engineering student uses the experiences of his teachers, as well as the knowledge captured in books and lecture material. Therefore their model of the design process, shown in Figure C.2 utilises presuppositions or protomodels as origins of solution concepts.

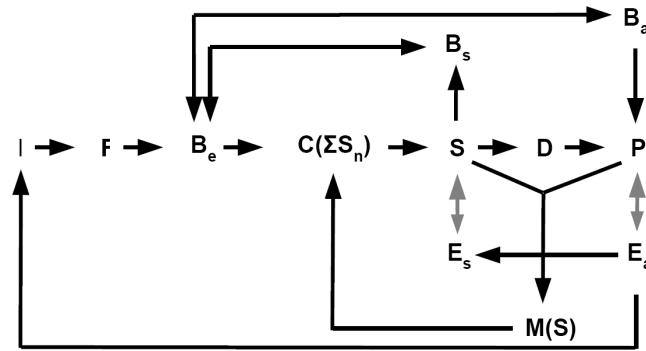


Figure C.2: Model of Design Process According Hybs and Gero [70]

The symbols in the Figure C.2 have the following meaning:

I	the designer's intent
F	function: the product's purpose
B	the product's behaviour
B_e	the product's expected behaviour
B_s	the simulated behaviour of the product
B_a	the actual behaviour of the product
S	structure: "the configuration, arrangement, organisation and form of the product's constituents and their relationships", representing the product "from the point view of the whole rather than that of a single part"
C	cross-over: the "process of combining variables of several structures"
D	design: the "representation of the designer's perception of the product's structure"
P	product: the "physical realisation of the design"
E	environment
E_s	the simulated environment of the product
E_a	the actual environment of the product
M	mutation: the "process of changing variables of the structure"
→	the process of transformation
↔	the process of comparison
↔	the process of indirect transformation

The equation, $I \rightarrow F$, describes the transformation of generating a functional description of the product based on the designer's intent, while the next equation, $F \rightarrow B_e$, uses these functions to generate the description of the product's expected behaviour. The process of synthesis is used to generate the product's structure, S, from the expected behaviour, B_e , utilising structures from previous experiences, S_n , through a process Hybs and Gero have termed "cross-breeding" or "cross-over". This process is repeated until the simulated behaviour, B_s , is close enough to the expected behaviour, B_e . These simulations should take the environment of the product into account (equation $S \leftrightarrow E_s$), by approximating the actual environment E_a by a model of the environment, E_s . The equation, $S \rightarrow D$, describes the process of documenting the design, so that the product, P, can be manufactured. The manufacturing process is described by the equation $D \rightarrow P$. The product, P, interacts with the actual environment, E_a , and exhibits actual behaviour, B_a , which is compared with the expected behaviour, B_e . If the difference between B_e and B_a are unacceptably large, the cycle, $B_e \rightarrow C(\Sigma S_n) \rightarrow S \rightarrow D \rightarrow P$, is repeated until the difference is acceptably small. Obviously both the structure, S, and the product, P, contribute to the population of structures that can be used "cross breed" the next product.

C.3 Model by Johannes

On his web-site "Methodical Design" Johannes [73] defines two basic philosophies to this process of design. One requires the designer to derive the solution to the design problem by taking the user requirements into account in methodical, systematic manner, as per the prescriptive models of the development process described in Section 3.1. The other he calls "imagination" and is distinguished by the intuitive, imaginative visualisation of the solution to the design problem in its entirety. This method is characterised by two aspects:

- the visualised solution is developed without the assistance of any design methods or tools, and
- visual aspects and geometric modelling are given high priority.

Therefore the visualised solution is evaluated in terms of the functional requirements, and adapted to make it fit.

C.4 Model by Reymen

Reymen et al. [1] used the theory of state-transition systems to describe the development process, as "a finite sequence of activities, necessary to obtain the design goal", which is "to create one or more desired representations of the product being designed having a desired state".

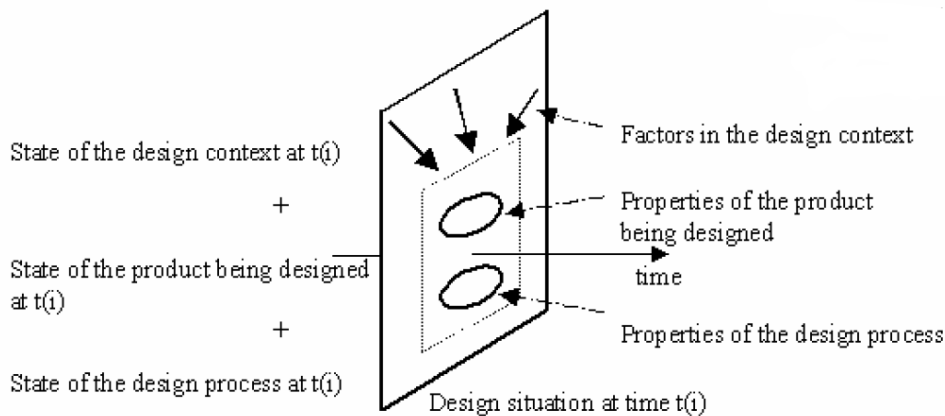


Figure C.3: A Design Situation as a State According to Reymen et al. [1]

As indicated in Figure C.3, Reymen defines "properties", which are "characteristics of the product being designed or the design process" and "factors", "which are external influences on the characteristics of the product being designed or the design process". Properties are determined by the designer, e.g. the mass, length or volume of a component or assembly. The design context is the sum of the all factors that influence the product being designed or the design process, e.g. "the life-cycle of the product being designed, the company's quality handbook, laws, patents".

The current state of the design situation, which includes the current state of the product being designed and the current state of the design process, "can be changed into another design situation by" designers through a design activity, which constitutes a transformation. The current state of the design context can be changed by stakeholders or designers as part of this transformation. This is shown schematically in Figure C.4.

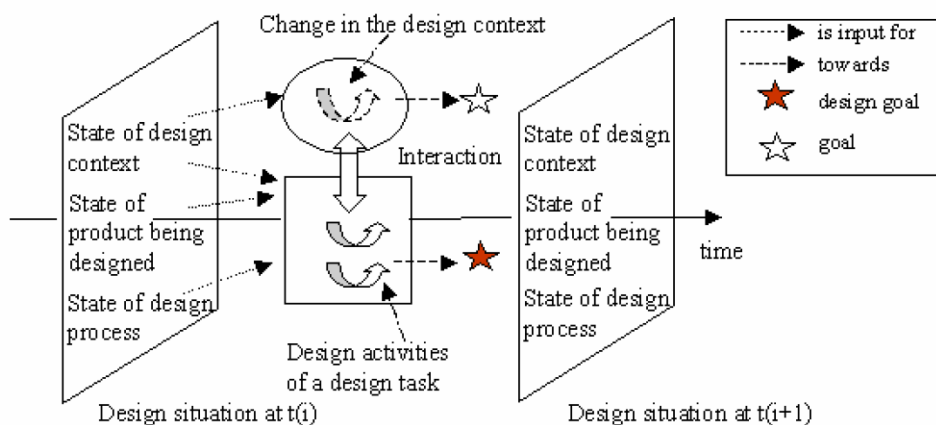


Figure C.4: Transitions from One State to Another According to Reymen et al. [1]

The design activity or task is defined by Reymen et al. [1] as "a task to transform the current state of the product being designed and/or the design process into a desired state, taking into account design context". Each design task can consist of sub-tasks, which can be "executed in parallel or in sequence; iteration between sub-tasks can also occur". Tasks and sub-tasks "can be defined at the beginning or during the project".

C.5 Model by Stempfle and Badke-Schaub

Stempfle and Badke-Schaub [115] analysed the communication within a design team during an experiment where three teams had to complete a complex design tasks. The analysis indicated that "four basic cognitive operations are necessary":

generation	generate a possible solution
exploration	analyse the solution
comparison	compare the results of the analysis to the requirements
selection	select a new course of action depending on the comparison

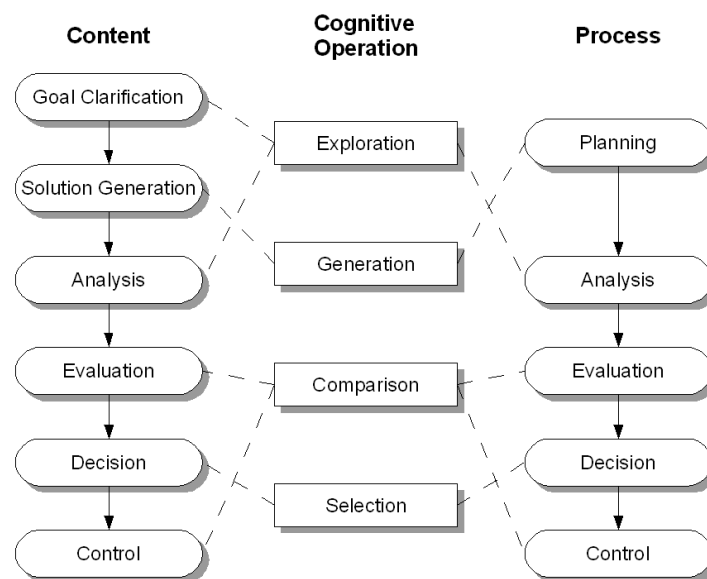


Figure C.5: Generic Model of Design Activities by Stempfle and Badke-Schaub [115]

The first two processes are used to "widen the problem space" while the last two "narrow the problem space". As indicated in Figure C.5 these basic cognitive operations are applicable to both the generation of the solution to the design problem as well as the establishment of the design process to generate that solution. The experiment revealed that in practice the design process was characterised "by a constant interweaving of content-directed sequences with process-directed sequences". Both levels showed a repeated analysis-evaluation loop, enabling the design team to keep the size of the problem space within acceptable limits. Although this model was established through the analysis of communication within teams, the author believes that it is applicable to design in general.

The four basic cognitive operations identified in the experiment by Stempfle and Badke-Schaub [115] are not always employed in the sequence shown. Figure C.6 shows two different sequences found during the analysis of the experiment. "Process 1 is characterised by an immediate evaluation of solution ideas", resulting in a time saving. However, for complex problems, where "questions or misunderstandings as to the nature of the solution idea" are more likely to occur, the likelihood of employing process 2 increases. In Process 2 the generation of a proposed solution is followed by an analysis of the solution before it is evaluated.

In both processes old solutions, previously rejected, were analysed again and re-evaluated, when no other potentially viable solution could be proposed any more. Solutions were accepted on the grounds of exceeding a perceived level of quality (in the sense of meeting most or the most critical requirements), rather than on the grounds of being the optimum solution of an extensive set of potential solutions. This practice is called "satisficing" by Quinn[103] and Simon[113]. The description of the experiments indicates that solutions were generated on a concrete, physical level rather than on an abstract, functional one.

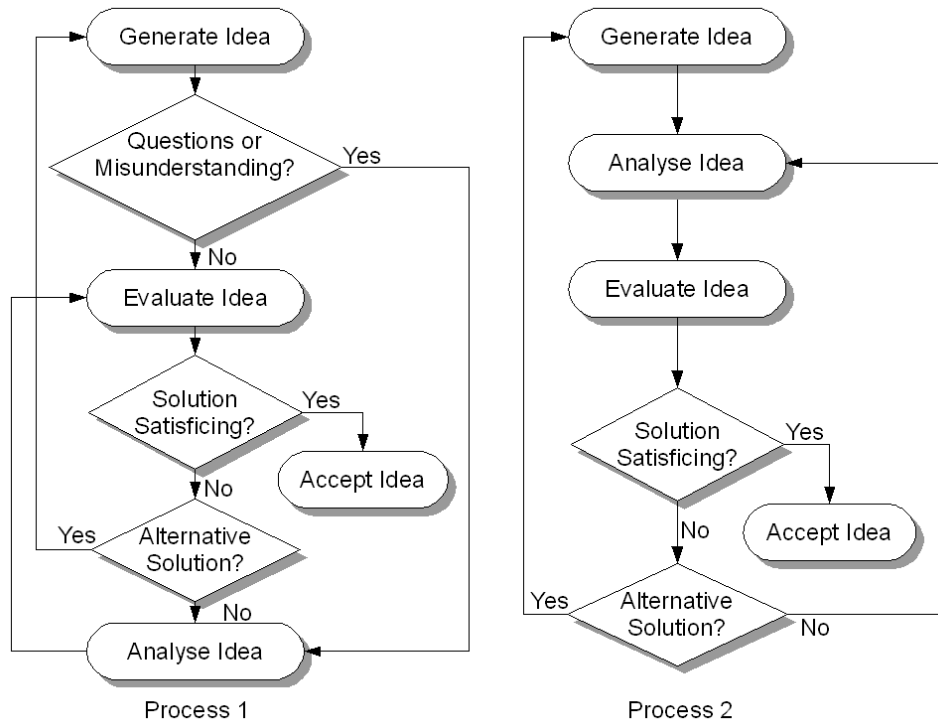


Figure C.6: Descriptive Models of the Development Process Presented by Stempfle and Badke-Schaub [115]

C.6 Model by Günther and Ehrlenspiel

Günther and Ehrlenspiel[60] recorded and analysed the the design process followed by "*experienced designers with no education in design methodology*" (identified as p-designers) as well as "*designers with an education in design methodology at a university*" (identified as m-designers). Although both groups were found to adhere to the basic structural phases of a prescriptive model of the development process as described in Sections B.6 or B.7, p-designers would "*work on one sub-problem in the conceptual, rough and final embodiment design, and then move on to the next sub-problem*", employing a corrective variation procedure to evolve the complete design, as described in Section 4.1. P-designers would also strive for an acceptable solution within acceptable time constraints rather than finding the overall optimum solution.

C.7 Model by Hansen and Ahmed

Hansen and Ahmed [63] evaluated the design process followed by twelve designers, including both experienced as well as novice designers. Based on the analysis of the transcripts the researchers modelled the design process as shown in Figure C.7.

The model consists of a decision map, which is string of decision nodes in a progression along the time axis. The each decision node has three inputs:

- the goals to be achieved in the design process,
- the alternative solutions generated in the previous step, and
- the contextual information, summarised by Hansen as "mindset" and "methods".

Within each decision node the alternative proposed solutions are evaluated against the specified goals. Next the proposed solution is validated by checking it against "*identified product life concerns, e.g. manufacturing, distribution or use*". The influence of the proposed solution on the complete design process is also considered in the "navigating" step. In the "unifying" step the information obtained from the steps above as well as the

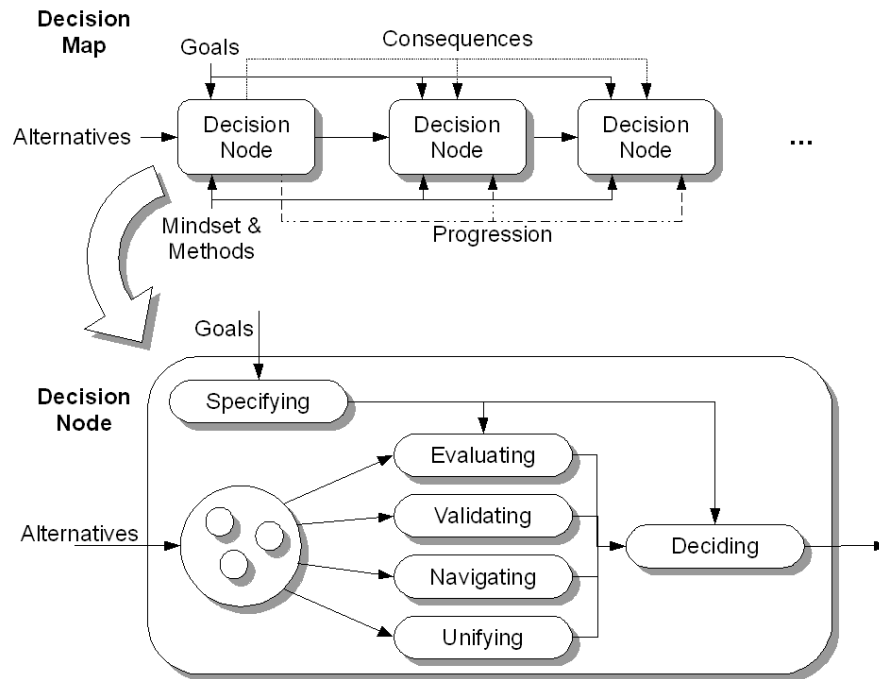


Figure C.7: Model of the Design Process by Hansen [63]

previous decision nodes has to be weighted and balanced in order to achieve the desired result within the project constraints at the end of the complete process. This results in a decision within the design process, which is an input to the next decision mode. Two other outputs of each node, which are also inputs to the subsequent decision nodes are

- the consequences of the decision taken, and
- the evolution or "progression" of the complete design solution along the time axis of this process.

Appendix D

Characteristics of Companies and Their Development Processes

Synopsis

Chapter 5 a framework of variables that describe the context and the characteristics of a development project. In the search for this framework the author came across various classification schemes to describe companies, their products and their business environment. These are described briefly in this appendix.

D.1 Influences on a Design Project by Wallace and Hales

Wallace and Hales [127] analysed data collected during the execution of a design project involving the development of a "high-pressure and high-temperature system for evaluation of materials in a simulated slagging coal gasifier environment". The context of a design project is shown in Figure D.1 below. Given the context model as reference, Wallace and Hales generated a list from literature of 103 factors considered most likely to influence the engineering design process, and divided them into 20 categories of influence at 5 levels of resolution as shown in Table D.1.

Level of Resolution	Influence Category	Examples of Factors
Macroeconomic	External influence	social, political, economic, ecological, legal, ...
Microeconomic	Market	demand, competition, risk, ...
	Resource availability	finance, services, people, ...
	Customer	need, urgency, expectations, ...
Corporate	Corporate structure	size, span, complexity, ...
	Corporate systems	integration, remuneration, ...
	Corporate strategy	objectives, risk-taking, ...
	Shared value	commitment, enthusiasm, ...
	Management style	autocratic, benevolent, ...
	Management skills	coordinating, resource use, ...
Project	Management staff	judgement, confidence, ...
	Design task	magnitude, complexity, risk, novelty, quantity, timing, ...
	Design team	expertise, experience, user, role balance, motivation, ...
	Design techniques	systematic procedures, communicating, motivating, ...
Personal	Design output	productivity, quality, ...
	Personal knowledge	knowledge base, usefulness, ...
	Personal skills	perception, imagination, ...
	Personal attitude	self-discipline, integrity, ...
	Personal motivation	enthusiasm, involvement, ...
	Personal output	productivity, quality, ...

Table D.1: Factors Influencing the Engineering Design Process by Wallace and Hales [127]

D.2 Product Properties by Tjalve

Tjalve [121] organises the factors that influence a product and its development into six categories as shown in Figure D.2. The basic or elementary design properties of the product, listed in the centre of the figure, are the ones that the manipulates directly. The factors of other five categories on the circumference, are dictated by the environment and context of the project. Hubka and Eder [68] add "tolerances" and "manufacturing method" to the list of basic properties. Properties of the other five categories are consequences of the choices that the designer has made with respect to the basic properties.

The factors that this research is interested in are the design factors, i.e. the factors or properties which effect the design or development process.

D.3 Company Classification by Barber and Hollier

To aid companies with the selection of computer-aided production control systems Barber and Hollier [15] analysed 88 companies in the batch manufacturing sector of the engineering industry, and classified them into 6 groups according to the following criteria:

- market and customer environment
- product complexity

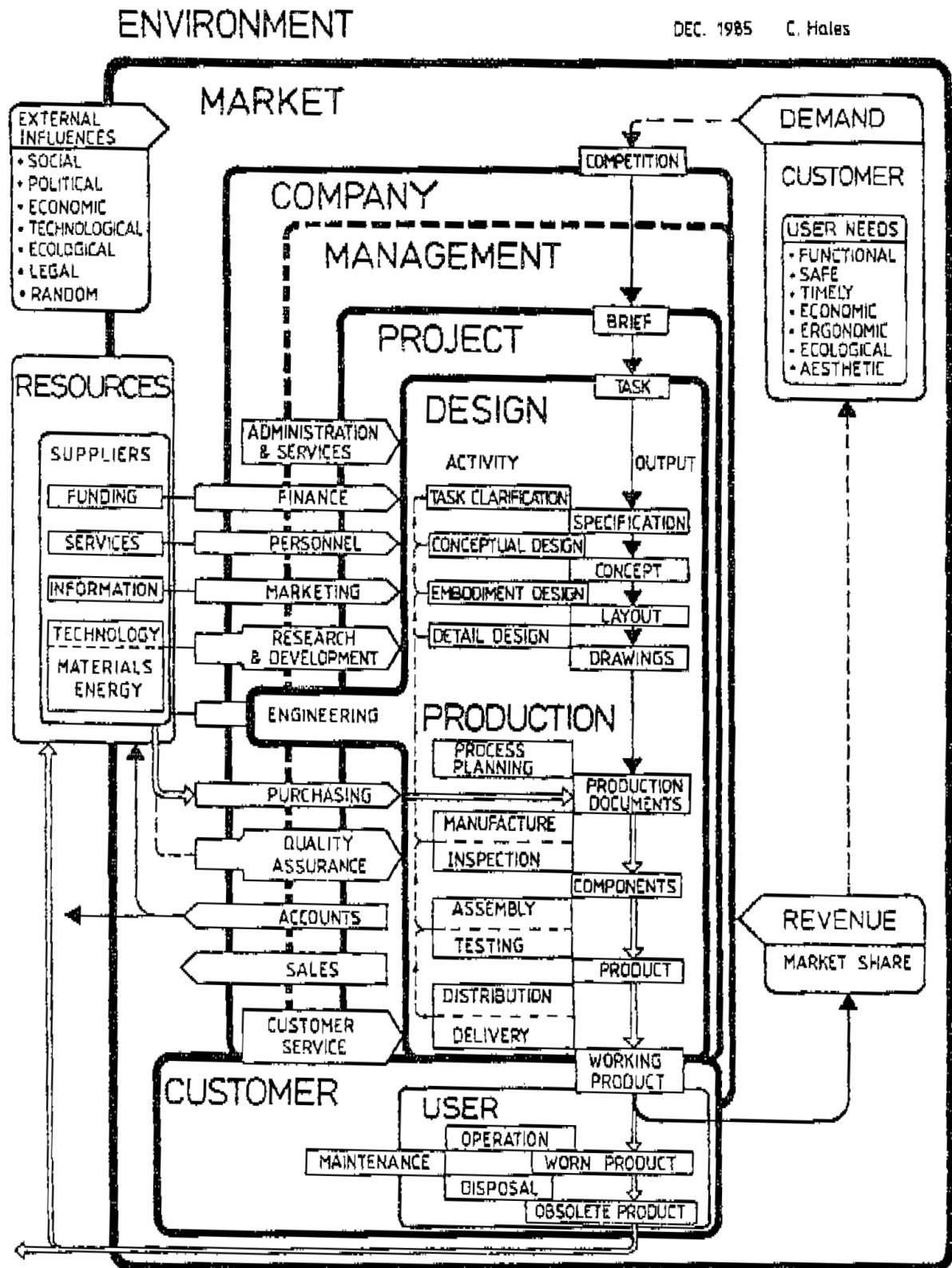


Figure D.1: The Context of a Design Project According to Wallace and Hales [127]

- nature and complexity of manufacturing operations
- supplier environment

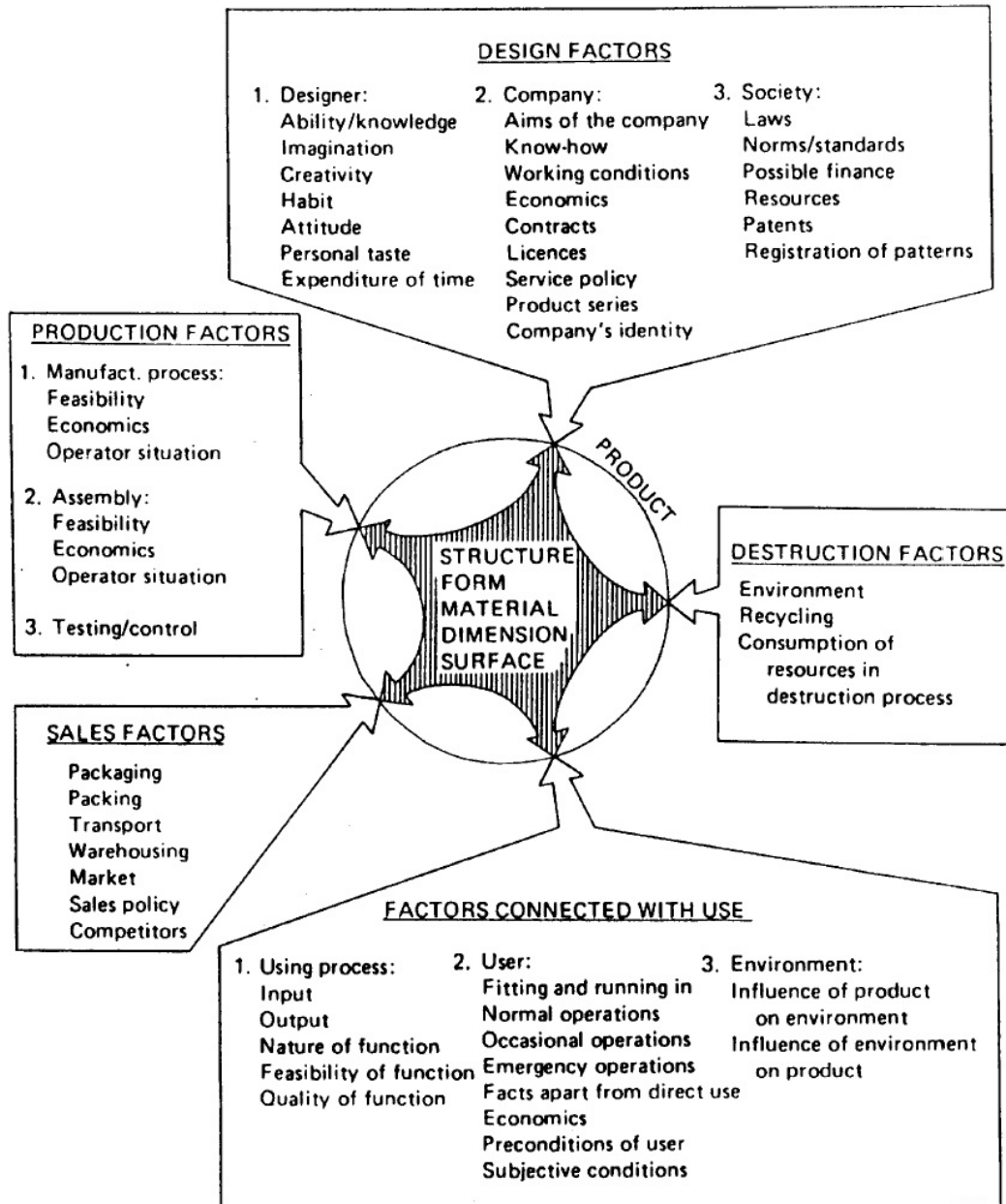


Figure D.2: Product Properties According to Tjalve [121]

- company structure and manufacturing policies

Due to Barber and Hollier wanting to classify the companies utilising numerical analyses, a number of numerical variables were chosen for each of these headings. As an example the market and customer environment was described by the number of distinct products, the percentage of output manufactured for stock, the percentage of output manufactured to order, and the manufacturing lead time. Since the analysis was aimed at production control systems the heading "nature and complexity of manufacturing operations" was characterised by the highest number of variables, and included the number of live shop orders and the number of work stations, for example.

D.4 The Seven Dimensions of a Company's Product Development by Andreasen et al.

Hein [64] quotes an article by Andreasen et al. [10] which identifies seven dimensions which define a company's ability to develop products. A schematic of these dimensions is shown in Figure D.3.

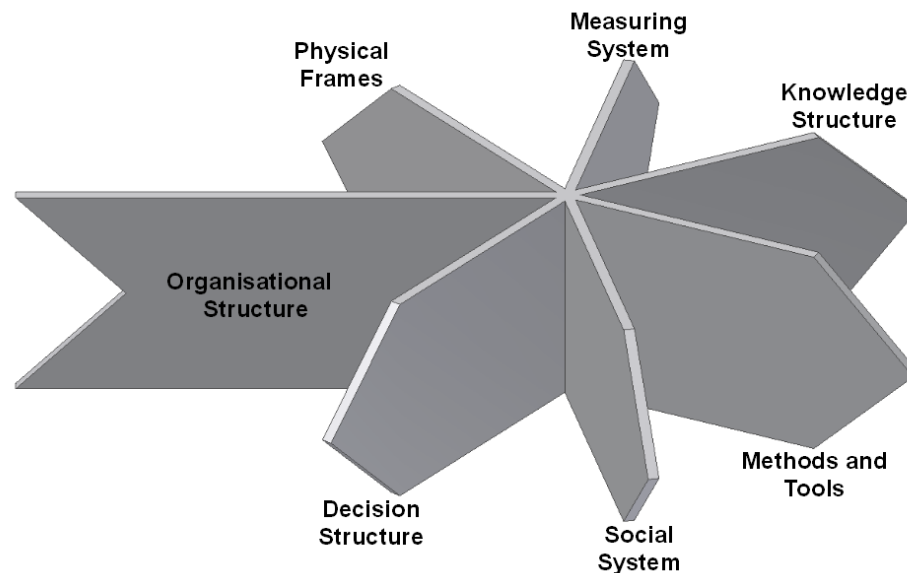


Figure D.3: The Seven Dimensions of a Company's Product Development System by Andreasen et al. [10]

The dimension "Methods and tools" deals with the aspects surrounding the availability and use of methods and tools during the development process. The dimension "measuring system" considers performance measurement in terms of personnel performance (time in attendance, quality of work), but also project performance (keeping of deadlines, development cost, amount of rework), as well as product performance (product quality, meeting of performance and other requirements). "Knowledge structure" is the dimension which considers how knowledge is acquired, processed and communicated. It includes know-how, both on a technical and a methodological level, as well as the employee's or team member's mind set.

D.5 Parameters Affecting the Design Process According to Ehrlenspiel

Research into the "*designer's thinking methods and design procedures*" by Ehrlenspiel et al. [47] has identified parameters that influence the design process from the viewpoint of an individual designer. The parameters are shown in Figure D.4.

D.6 Field Study of Design by Hykin and Laming

Hykin and Laming [71] conducted a field study investigating the design process for 11 different projects in various industries, developing products of various complexities. Their investigation attempted to identify a correlation between the design process and a number of factors including the following constraints on the development process, as well as the technical content of the project, the probability of error (i.e. risk) and the complexity of the product being developed:

- Cost
- Time

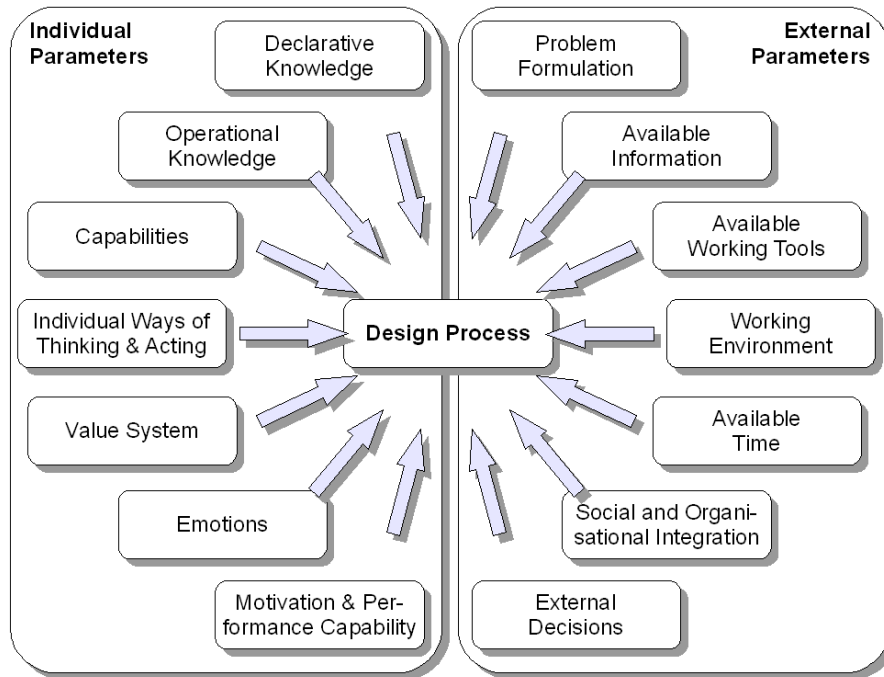


Figure D.4: Ehrlenspiel's Parameters Influencing the Design Process [47]

- Detailed design brief
- Level of innovation
- Production quantity: mass, batch or one-off
- Other

He found that the production quantity had the greatest effect on the development process: products for mass or batch production were handled differently to products being manufactured only once. The development process for products for mass and batch production emphasise the first phases of the development process, *"in order to produce a nearly right solution for the final design phase and production."* In contrast, the development process for one-off products emphasises the final design phase, *"and a prime aim appears to be to build manoeuvrability into the earlier design proposals."*

Another aspect that varied with the production quantity is the management style and the structure of the organisation. Mass production products are developed in a *"well-established hierarchy"* and a *"formal rigidly defined system of control."* In contrast the management and organisational style is less formal and adaptive for one-off products.

D.7 Company Classification by Maffin

Maffin [87] has identified six key dimensions of a company, as shown in Figure D.5. Each of these key dimensions is associated with a set of defining factors, as described in Table D.2.

The structure of a company will usually reflect the role it is expected to perform, i.e. production unit, innovation centre, customer service and distribution centre. Large organisations usually have the resources to employ specialist R&D departments to innovate. Smaller companies, on the other hand, have more informal and efficient communication mechanisms, resulting in a less formal and detailed procedural structure. However, the level of formalisation is not only a function of the company size. This is discussed in more detail in Section 7.3.3.

Price sensitive solutions will tend to emphasise the manufacturing, assembly and production aspects, since high production volumes and emphasis on manufacturing efficiency will make products even with small mark-ups financially viable. Manufacture-to-stock companies usually fall into this category. Manufacture-to-order

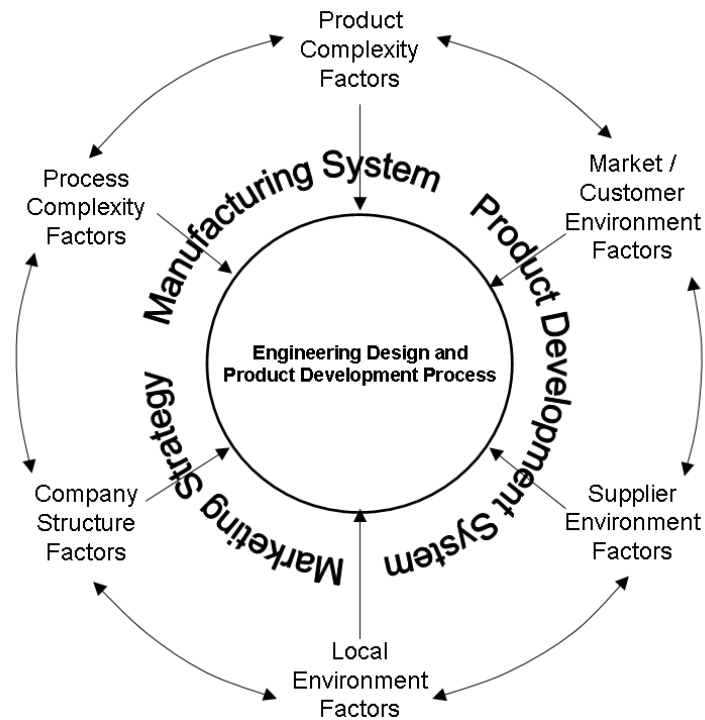


Figure D.5: Key Dimensions to Describe a Company by Maffin [87]

may have products that are too expensive, complex or large to keep in stock. At the other end of the spectrum, products of companies that engineer-to-order are characterised by high design content, often requiring specialist capabilities. Manufacture-to-stock, manufacture-to-order and assemble-to-order companies usually employ product development projects, while engineer-to-order and customised-made-to-order companies are usually orientated towards contract based projects. This has a major influence on the structuring of the phases of the development phases.

Projects which require high levels of innovation will tend to have incomplete specifications, and therefore incorporate project phases concentration on research, market investigations and the identification of suitable technologies. Projects with substantial prototyping and testing costs will tend to concentrate on the 'front-end' of the development process to reduce the risks as much as possible during the concept phases.

The development process is also influenced by the closeness of the designer to his customer. In projects in engineer-to-order and customised-made-to-order companies a direct link to the customer usually exists, while products developed in manufacture-to-stock companies are usually specified by the marketing department and no direct contact with the customer is available.

The complexity of the product also has an effect on the development process. Products of high complexity and/or a deep structure are usually associated with a distinct embodiment phase to refine the concept and establish sub-system requirements. For products of low complexity, this usually achieved in the concept phase. The tendency towards standardisation at the product level is usually found for products of low complexity in the assemble-to-order and manufacture-to-stock companies.

Most of the products developed in industry is concerned with the effective manufacture of the new product using established and existing equipment and processes. Only when the requirement for a new production process is envisioned will the process development be implemented in parallel with the development of the product.

"Make-or-buy" decisions are based on the degree of externalisation of the design and manufacturing process, which in turn depends on what the company consider to be its core competencies. Generally, it is easier for manufacture-to-stock and assemble-to-order companies to develop long-term relationships with suppliers, where the supplier plays an integral part in the product's design. For engineer-to-order companies the scope of work changes too often to make highly integrated supplier relationships financially viable.

Environment factors that can influence the design process fall into the following categories: social, political, economic and legislative. Political and legislative condition, both locally and internationally, may impose

Dimension	Factors
Company Structure	Size, Independence, Centralisation
Process	Complexity, Flexibility, Constraints, Volume, Internal span of process
Supplier	Rationalisation, Degree of control, Collaboration, Locality
Market	Type, Size and share, Complexity, Infrastructure, Exports, Competitors, Competitive criteria
Product	Type, Variety, Complexity, Structure, Status, Innovation rate, Design capability
Environment	Labour market, Skills, Training, Support infrastructure, Financing

Table D.2: Defining Factors for Key Dimensions [87]

additional requirements on the design process and/or the product. The local environment may influence "make-or-buy" decisions when specific technologies or capabilities may not locally available.

References

- [1]
- [2] *The Compact Oxford Dictionary*. Retrieved Jun2 2008.
- [3] Bae systems, land systems omc, website. internet: <http://www.baesystemsomc.co.za/>, September 2007.
- [4] BAE Systems, Website. internet: <http://www.baesystems.com/>, September 2007.
- [5] Desk Research Report: Eleven Lessons: managing design in eleven global companies. Technical report, Design Council, United Kingdom, November 2007.
- [6] Indutech, Website. www.indutech.co.za, 2008.
- [7] C. Alexander. A City Is Not a Tree. *Design*, 206:46–55, 1966.
- [8] M. M. Andreasen. Design Methodology. *Journal of Engineering Design*, 2(4):321–335, 1991.
- [9] M. M. Andreasen and L. Hein. *Integrated Product Development*. IFS Publications Ltd., Springer Verlag, London, 1987.
- [10] M. M. Andreasen, L. Hein, L. Kirkgard, and K. Sant. Product Development - Fundamentals of Innovation. 1989. via [64].
- [11] M. M. Andreasen and N. H. Mortensen. Nature of Design Features, The. In *7. Symposium 'Fertigungsgerechtes Konstruieren'*, 1996.
- [12] L. B. Archer. *Systematic Method For Designers*. 1984. via [32].
- [13] M. Asimow. *Introduction to Design*. 1962.
- [14] P. Badke-Schaub. Strategies of Experts in Engineering Design: Between Innovation and Routine Behaviour. *Journal of Design Research*, 2, 2004.
- [15] K. D. Barber and R. H. Hollier. Use of Numerical Analysis to Classify Companies According to Production Control Complexity, The. *International Journal of Production Research*, 24(1):203 – 222, 1986.
- [16] P. Barkan. Benefits and Limitations of Structured Methodologies. In *Management of Design: Engineering and Management Perspectives*, pages 163–178. Kluwer Academic Publishers, 1994.
- [17] W. Beitz. Design Science - the Need For a Scientific Basis For Engineering Design Methodology. *Journal of Engineering Design*, (2), 1994.
- [18] C. Bichlmaier. *Methoden Zur Flexiblen Gestaltung von Integrierten Entwicklungsprozessen*. PhD thesis, Technische Universität München, 1999.
- [19] B. S. Blanchard and W. J. Fabrycky. *Systems Engineering and Analysis*. Prentice Hall, 3rd edition, 1998.
- [20] L.T.M. Blessing. *A process-based approach to computer-supported engineering design*. May 1994.
- [21] Lesley Brown, editor. *Cassell Concise Dictionary, The*. Nigel Wilcockson, 1962.
- [22] S. L. Brown and Eisenhardt. K. M. Product Development: Past Research, Present Findings and Future Directions. *Academy of Management Review*, 20(2):343–378, 1985.

- [23] A. Bruseberg and D. McDonagh Philp. User-Centred Design Research Methods: The Designer's Perspective. In Childs PRN and Brodhurst E, editors, *Integrating Design Education Beyond 2000 Conference*, pages 179–184. University of Sussex, September 2000.
- [24] S. Butdee, C. Noomtong, F. Vignat, and G. Thomann. Methodology of Knowledge Library Management for Concurrent Cooperative Design. In *Proceedings of the 24 th International Manufacturing Conference, IMC 24*, pages 711–718. Waterford Institute of Technology, Waterford, Ireland, 2007.
- [25] M. Cantamessa. Design Best Practices At Work: an Emperical Research Upon the Effectiveness of Desgin Support Tools. In *Proceedings of the ICED 97, Tampere*, pages 541–546, 1997.
- [26] D. Cavallucci. Intuitive Design Method (IDM), a New Approach On Design Methods Integration. In *Proceedings of ICAD2000 First International Conference On Axiomatic Design*, 2000.
- [27] C.M. Christensen. *The Innovators Dilemma: when new technologies cause great firms to fail*. 1997. via [119].
- [28] H. Christiaans, K. Dorst, and N. Cross. Levels of Competence in Product Designing. In *Proceedings of ICED 93*, pages 368–376. Zuerich: Edition Heurista, August 17-19 1993.
- [29] P.J. Clarkson and C.M. Eckert. *Design process improvement - a review of current practice*. 2005. via [5].
- [30] R. G. Cooper. New Product Process: An Empirical Based Classification Scheme. *R & D Management*, 13:1–13, 1983.
- [31] N. Cross. Science and Design Methodology: A Review. *Research in Engineering Design*, 5:63–69, 1993.
- [32] N. Cross. *Engineering Design Methods, Strategies For Product Design*. John Wiley and Sons, 2nd edition, 1994.
- [33] N. Cross and N. Roozenburg. Modelling the Design Process in Engineering and in Architecture. *Journal of Engineering Design*, 3(4):325–337, 1992.
- [34] S. J. Culley. Final Report. In *Future Issues For Design Research (FIDR) Workshop*, 1999.
- [35] W. Dankwort, J. Ovtcharova, and R. Weidlich. A Concept of Engineering Objects For Collaborative Virtual Engineering: Automotive Case Study. In *ProStep Conference*, 2003.
- [36] Instrument Data Communications. Principles of Project Management. [www.idc-online.com/technical references](http://www.idc-online.com/technical_references), 2000.
- [37] C. S. De Araujo. An Investigation of the Use of Design Methods. In *Proceedings of the 2ND INTERNATIONAL CONGRESS OF INDUSTRIAL ENGINEERING*, 1996.
- [38] C. S. De Araujo. *Acquisition of Product Development Tools in Industry: a Theoretical Contribution*. PhD thesis, Technical University of Denmark, 2000.
- [39] University of Stellenbosch Department of Industrial Engineering. Enterprise Engineering Handbook.
- [40] O. Diederich. Methodik Und Praxis - Erfahrungen in Einem Mittelstaendischen Unternehmen. *VDI Berichte*, 953:237–250, 1992.
- [41] K. Dooley, A. Subra, and J. Anderson. Best Practices in New Product Development: Adoption Rates, Patterns and Impact. *Journal of Operations Management*, 2000.
- [42] J. Drake. *Primary Generator and the Design Process, The*. Chichester, Wiley, 1982.
- [43] N. Du Preez, A. Basson, D. Lutters, and H. Nieberding. Roadmapping methodology for product development. In *Proceedings of PICMET '08*, July 2008.
- [44] N. Du Preez and B.R. Katz. Case study: The implementation of a radical innovation project. In *Proceedings of the International Conference on Competitive Manufacturing*, 2007.
- [45] A. Dumas and A. Whitfield. Why Design Is Difficult to Manage: a Survey of Attitudes and Practices in British Industry. *European Management Journal*, 7(1):50–56, 1989.

- [46] W. E. Eder. Design Modelling - A Design Science Approach (and Why Does Industry Ignore It?). *Journal of Engineering Design*, 9(4):355–371, 1998.
- [47] K. Ehrlenspiel and N. Dylla. Experimental Investigation of Designers' Thinking Methods and Design Procedures. *Journal of Engineering Design*, 4(3):201 – 212, 1993.
- [48] Ernst and Young. International Quality Study. Technical report, 1992. via [41].
- [49] N. F. O. Evbuomwan, S. Sivaloganathan, and A. Jebb. A Survey of Design Philosophies, Models, Methods and Systems. *Proceedings of the Institution of Mechanical Engineers*, pages 301–320, 1996.
- [50] S. Finger and J. R. Dixon. A Review of Research in Mechanical Engineering Design. Part 1: Descriptive, Prescriptive and Computer-Based Models of Design Processes. *Research in Engineering Design*, 1:51–67, 1989.
- [51] H. J. Franke. Konstruktionsmethodik Und Konstruktionspraxis - Eine Kritische Betrachtung. In *Proceedings of the ICED85*, Hamburg, 1985. via [8].
- [52] M. J. French. *Engineering Design - The Conceptual Stage*. Heinemann, 1971.
- [53] R. B. Frost. A Suggested Taxonomy For Engineering Design Problems. *Journal of Engineering Design*, 5(4):399–410, 1994.
- [54] R. B. Frost. Why Does Industry Ignore Design Science? *Journal of Engineering Design*, 10(4):301–304, 1999.
- [55] A. J. Fulcher and P. Hills. Towards a Strategic Framework For Design Research. *Journal of Engineering Design*, 7(2):183–193, 1996.
- [56] H. Gill. Adoption of Design Science By Industry - Why So Slow. *Journal of Engineering Design*, 1(3):289–295, 1990.
- [57] R. P. Gouvinhas and J. Corbett. A Discussion On Why Design Methods Have Not Been Widely Used Within Industry. In *Proceedings of ICED99*, pages 1167 – 1170, August 1999.
- [58] L. N. Green and E. Bonollo. Development of a Suite of Design Methods Appropriate For Teaching Product Design, The. *Global J. of Engng. Educ.*, 6(1):45–51, 2002.
- [59] J. Günther. *Individuelle Einflüsse Auf Den Konstruktionsprozeß*. PhD thesis, Technische Universität München, 1998.
- [60] J. Günther and K Ehrlenspiel. Comparing designers from practice and designers with systematic design education. *Design Studies*, 20:439 – 451, 1999.
- [61] A. Y. Ha and E. L. Porteus. Optimal Timing of Reviews in Concurrent Design For Manufacturability. *Management Science*, 41(9):1431 – 1447, September 1995.
- [62] C. Hales. *Analysis of the Engineering Design Process in an Industrial Context*. PhD thesis, University of Cambridge, 1987.
- [63] C.T. Hansen and S. Ahmed. An Analysis of Design Decision-making in Industrial Practice. In *International Design Conference - Design 2002*, May 14 - 17 2002.
- [64] L. Hein. Design Methodology in Practice. *Journal of Engineering Design*, 5(2):165–181, 1994.
- [65] W. Hillier, J. Musgrave, and P. O'Sullivan. *Knowledge and Design*. Chichester, Wiley, 1984.
- [66] I. Horváth and J. S. M. Vergeest. Engineering Design Research: Anno 2000. In *International Design Conference - Design 2000*, 2000.
- [67] D. Hubbard. *The Failure of Risk Management: Why It's Broken and How to Fix It*. John Wiley & Sons, 2009.
- [68] V. Hubka and E. E. Eder. *Theory of Technical Systems - A Total Concept Theory For Engineering Design*. Springer-Verlag, Berlin Heidelberg, 1988. via [129].

- [69] V. Hubka and W. E. Eder. *Einführung in Die Konstruktionswissenschaft. Übersicht, Modell, Ableitungen*. Springer, Berlin, 1992. via [59].
- [70] I. Hybs and J.S. Gero. An Evolutionary Process Model of Design. *Design Studies*, 13(3):273 – 290, July 1992.
- [71] D. H. W. Hykin and L. C. Laming. Design Case Histories: Report of a Field Study of Design in the United Kingdom Engineering Industry. *Proceedings of the Institution of Mechanical Engineers*, 189(23):203–211, 1975.
- [72] International Organization for Standardization. *ISO31000 Risk Management - Principles and Guidelines on Implementation*, 2009.
- [73] R. Johannes. Methodisches Entwerfen. internet: <http://miless.uni-essen.de/servlets/DerivateServlet/Derivate-10361/umodell.htm>, September 2003.
- [74] D. A. Jones, D. M. York, J. F. Nallon, and J. Simpson. Factors Influencing Requirement Management Toolset Selection. In *Proceedings of the Fifth Annual Symposium of the National Council On Systems Engineering*, volume 2, July 1995.
- [75] J. C. Jones and D. G. Thornley, editors. *Proceedings of the Conference On Design Methods*. Oxford, Pergamon, 1963.
- [76] N. P. Juster. Design Process and Design Methodologies, The. Technical report, University of Leeds, May 1985. in [50].
- [77] T.A. Kappel. Perspectives on Roadmaps: How Organisations Talk about the Future. *Journal of Prod Innovation Management*, 18:39–50, 2001.
- [78] C. T. Koeller. Innovation, Market Structure and Firm Size: a Simultaneous Equations Model. *Managerial and Decision Economics*, 16:259–269, 1995.
- [79] R. Kostoff and R. Schaller. Science and Technology Roadmaps. *IEEE Transactions on Engineering Management*, 48(2), May 2001.
- [80] F. L. Krause. Virtual Product Creation 2013 - Objectives For the Enhancement of Virtual Product Creation. Presentation, September 2003.
- [81] J. Legardeur, J. F. Boujut, and H. Tiger. Innovating in a Routine Design Process - Empirical Study of an Industrial Situation. In *IDMME 2000*, Montreal, May 16-19 2000.
- [82] U. Lindemann and J. Wulf. Action Orientation in Design Methodology. In *Proceedings of ICED01*, August 21-23 2001.
- [83] S.C.Y Lu and N. Jing. A socio-technical negotiation approach for collaborative design in software engineering. *Int. J. Collaborative Engineering*, 1(1/2):185–209, 2009.
- [84] D. Lutters. *MANUFACTURING INTEGRATION BASED ON INFORMATION MANAGEMENT*. PhD thesis, University of Twente, May 2001.
- [85] D. Lutters, M. Johnson, M.E. Toxopeus, and N.D. Du Preez. Introducing the roadmap concept to inexperienced users: a case study. In *International Conference on Competitive Manufacturing*, 2007.
- [86] D. J. B. Maffin. Engineering Design Models: Context, Theory and Practice. *Journal of Engineering Design*, 9(4):315–327, 1998.
- [87] D. J. B. Maffin, N. Alderman, P. Braiden, B. Hills, and A. Thwaites. Company Classification: A New Perspective On Modelling the Engineering Design and Product Development Process. *Journal of Engineering Design*, 6(4):275–289, 1995.
- [88] L. J. March. Logic of Design, The. 1984. via [32].
- [89] Lockheed Martin. Skunk works. website: <http://www.lockheedmartin.com/cgi-bin/pfv.pl>, September 2007.

- [90] P. Martin and P. Veron. Training Experience On Multi-Site Collaborative Product Design. 2003. not certain of the date.
- [91] C. Marxt and F. Hacklin. Design, Product Development, Innovation: All the Same in the End? A Short Discussion On Terminology. *International Design Conference - Design 2004*, May 18-21 2004.
- [92] T. W. Maver. *Appraisal in the Building Design Process*, pages 195–202. Cambridge, MA, MIT Press, 1970.
- [93] M. D. McNeese, B. S. Zaff, C. E. Brown, M. Citera, and A. R. Wellens. The role of a group-centered approach in the development of computer-supported collaborative design technologies. In *Proceedings of the 36th Annual Meeting of the Human Factors Society*, pages 867–871, Santa Monica, CA, 1992. Human Factors Society.
- [94] K.W. Ng. A critical analysis of current engineering design methodologies from a decision making perspective. In *Intelligent Production Machines and Systems, I*PROMS 2006 note=website: conference.iproms.org, retrieved July 2008*. I*PROMS, 2006.
- [95] M. Nordlund, D. Tate, and N. P. Suh. Growth of Axiomatic Design Through Industrial Practice. In *3rd CIRP Workshop On Design and the Implementation of Intelligent Manufacturing Systems*, June 1996.
- [96] G. Pahl. Vorgehen Beim Entwerfen. In *Proceedings of the International Conference On Engineering, ICED '83, Copenhagen*, volume 1, pages 209–219, 1983.
- [97] G. Pahl and W. Beitz. *Engineering Design - A Systematic Approach*. 1988.
- [98] C.S. Peirce. *The Collected Papers of Charles Sanders Peirce*. Cambridge MA: Harvard University Press, 1931 - 1958.
- [99] R. Phaal, C.J.P. Farrukh, and D.R. Probert. Technology roadmapping - a planning framework for evolution and revolution. *Technological Forecasting and Social Change*, 71(1):5–26, January 2004.
- [100] D.T. Pham, S. Dimov, and K.W. Ng. A framework for a novel descriptive product design methodology. In *Innovative Production Machines and Systems, I*PROMS 2008*. I*PROMS, 2008.
- [101] S. Pugh. Design Activity Models: Worldwide Emergence and Convergence. *Design Studies*, 7(3):167–173, July 1986.
- [102] S. Pugh. *Total Design: Integrated Methods For Successful Product Engineering*. Addison-Wesley Publishing Company, 1991.
- [103] J.B. Quinn. *Strategies for change: Logical incrementalism*. R.D. Irwin, Homewood, Illinois, 1980. via [94].
- [104] D. F. Radcliff and P. Harrison. Transforming Design Practice in a Small Manufacturing Enterprise. In *Proceedings of the ASME Conference 1994, Design Theory and Methodology, DTM-'94*, pages 91–98. ASME, 1994.
- [105] R. Rohatynski. Diagnosing the Gap Between Methodology of Engineering Design and Industrial Practice. In *Proceedings of the INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN ICED 01 GLASGOW*, August 2001.
- [106] R. Rothwell and W. Zegveld. *Innovation and the Small and Medium Sized Firm*. Francis Pinter, 1982. via [107].
- [107] R. Roy and S. Potter. Managing Design Projects in Small and Medium-Sized Firms. *Technology Analysis and Strategic Management*, 2(3):321–336, 1990.
- [108] G. Rzevski. On the Design of a Design Methodology. *Design: Science: Method*, pages 6 – 17, 1981.
- [109] E. Schnelle and R. Pfeuffer. Vorteile Und Grenzen der Konstruktionsmethodik Bei der Entwicklung von Produkten der Medizintechnik. *Konstruktion*, 41:395–402, 1989.
- [110] A. Schulz, D. Clausing, H. Negele, and E. Fricke. Shifting the View in Systems Development - Technology Development At the Fuzzy Front End As a Key to Success. In *Proceedings of 1999 ASME DETC, 11th International Conference On Design Theory and Methodology, Las Vegas,, September 1999*.

- [111] A. J. Shenhar and R. M. Wideman. Optimizing Success By Matching Management Style to Project Type. website PMForum, September 2000.
- [112] J. E. Shigley. *Mechanical Engineering Design*. McGraw-Hill Book Company, 1st Metric Edition edition, 1986.
- [113] H.A. Simon. *The science of the artificial*. MIT Press, Cambridge, MA, 1969.
- [114] X. M. Song, Wm. E. Souder, and B. Dyer. A Causal Model of the Impact of Skills, Synergy and Design Sensitivity On New Product Performance. *Journal of Prod Innovation Management*, 14:88–101, 1997.
- [115] J. Stempfle and P. Badke-Schaub. Thinking in design teams - an analysis of team communication. *Design Studies*, pages 473 – 496, 2002.
- [116] R. Stetter. *Method Implementation in Integrated Product Development*. PhD thesis, Technische Universität München, August 2000.
- [117] A. Suireg. Methodology of Mechanical Systems Design. In *Proc. ICED81: WDK-5, Review of Design Methodology*. Heurista, 1981.
- [118] K. Sutinen, L. Almfelt, and J. Malmqvist. Supporting Concept Development Using Quantitative Requirements Traceability. 2001.
- [119] P. Thomond and F. Lettice. Disruptive Innovation Explored. In R. Goncalves, R. Roy, and A. Steiger-Garcia, editors, *Proceedings of 9th ISPE International Conference on Concurrent Engineering: Research and Applications: Advances in Concurrent Engineering*, pages 1021–1025, July 2002.
- [120] G. Thompson. Requirements Engineering - Laying the Foundations For Successful Design. In *Proceedings of the International Conference On Engineering Design*, August 21-23 2001.
- [121] E Tjalve. *A Short Course in Industrial Design*. Newnes-Butterworths, London, 1979. via [129].
- [122] T. Tomiyama. A Note On Research Directions of Design Studies. In *Proceedings of the ICED 97, Tampere*, pages 29–34, 1997.
- [123] D. G. Ullman. *Mechanical Design Process, The*. McGraw-Hill, 3rd edition, 2003.
- [124] A. H. Van de Ven and D. L. Ferry. *Measuring and Assessing Organizations*. John Wiley and Sons, New York, 1980. via [41].
- [125] VDI. *VDI 2221, Systematic Approach to the Development and Design of Technical Systems and Products*, May 1993.
- [126] E. Von Handenhoven and P. Trassnert. Design Knowledge and Design Skills - What Industry Tends to Show Us. In *Proceedings of ICED99*, pages 153–158, August 24-26 1999.
- [127] K. M. Wallace and C. Hales. Detailed Analysis of an Engineering Design Project. *Proceedings of ICED87*, pages 94–101, August 17-20 1987.
- [128] V. Walsh and R. Roy. *Plastics Products: Good Design, Innovation and Business Success*. Technical report, 1983. via [107].
- [129] A. Warell. *Industrial Design Elements: a Theoretical Foundation For Industrial Design Based On a Design Science Perspective*. Department of Mechanical Engineering, Linköpings Universitet, 1999.
- [130] W. Zanker. *Situative Anpassung Und Neukombination von ENTwicklungsmethoden*. PhD thesis, Technische Universität München, 1999.

Index

- Ahmed, S., 134
Alexander, C., 57
analysis, 2, 4, 7, 15, 24, 28, 31, 35, 37, 38, 54, 55, 65, 67, 70, 118, 123, 133
 conjecture, 29, 130
 design, 7
 evaluation, 133
 functional, 2, 17, 22, 28, 119
 market, 25
 numerical, 140
 philosophical, 5
 problem, 116, 117
 requirements, 125
 theoretical, 5
Andreasen, M.M., 2, 12, 13, 33, 40, 126, 127, 141
approach, 3, 4, 9, 17, 22–24, 35, 37, 41, 45, 52, 69, 70, 80, 99, 118, 123
 descriptive, unstructured, 45
 holistic, 127
 information orientated, 83
 prescriptive, structured, systematic, 16, 22, 32, 33, 35, 36, 43, 54, 57, 86, 92, 119
 skunk works, 46
 team, 123
Archer, L.B., 24, 28, 30, 90, 117, 118
Asimow, M., 28

Badke-Schaub, P., 29, 33, 54, 55, 133
BAE
 Land Systems OMC, 71–75, 77, 78
 Systems, 71, 74
Barber, K.D., 40, 138, 140
Barkan, P., 37
Beitz, W., 3, 24, 27, 35, 36, 87, 90, 118, 121, 122
Bichlmaier, C., 14, 15, 83
Blanchard, B.S., 16, 123
Blessing, L.T.M., 92
Brown, S.L., 43, 49
Bruseberg, A., 33, 35
Butdee, S., 90

Cantamessa, M., 55
capability, *see* personnel
capacity, *see* organisation
catalogue, 44, 64
 design, 47, 122
Cavallucci, D., 25
Christensen, C.M., 48
Christiaans, H., 54
Clarkson, P.J., 34

company, 18, 35, 40–45, 48, 50, 55, 56, 58, 63, 64, 69–75, 80, 86, 87, 92, *see* environment, organisation, 121, 132, 140–143
complexity, 2, 4, 40, 47, 56, 57, 86, 90
 design problem, 54, 57
 design process, 16, 40, 56
 manufacturing, production, 139, 140
 product, 33, 43, 46–48, 51, 52, 56, 57, 64, 76, 87, 123, 125, 138, 141, 143
 project, 25, 35, 46, 47, 49, 51, 52, 54, 56, 57, 64, 70, 75
concept, 28, 29, 32, 55
 abstract, 29
 concrete, 29
 design, 24, 52, 65, 116, 121
 development, 18, 28
 phase, 22, 43, 47, 52, 66, 70, 98, 116, 123, 143
 solution, 130
configuration, 44, 48, 50, 87, 131
 control, 64, 73, 76, 77, 125
 design, 13, 47, 55, 64
 system, 27, 123
context, 16, 33, 35, 40, 98, 132
 project, 4, 7, 21, 25, 32, 34–39, 43, 53, 57, 67, 70, 77, 79, 80, 84–86, 90, 92, 99, 119, 137, 138
 variable, 55, 56, 67, 77, 84, 86, 89, 99
Cooper, R.G., 4, 43
Corbett, J., 35
Cross, N., 5, 9, 12, 17, 22, 24, 26, 28, 29, 54, 87, 116, 117, 130
Culley, S.J., 4, 32, 40
customer, 15, 23, 42, 44, 45, 48, 65, 71, 72, 78, 84, 119, 121, 138, 142, 143
 requirement, 44, 64, 67, 78, 116, 121, 123

Dankwort, W., 37
De Araujo, C.S., 18, 37
descriptive, *see* development
design, *see* catalogue, concept, configuration, detail, phase, project
 capability, *see* personnel
 capacity, *see* organisation
 engineering, 2–5, 8, 12, 28, 34, 35, 138
 environment, 4
 industrial, 3, 33
 science, 2, 5, 7, 17, 32, 33, 35, 37, 40
Design Council, UK, 36, 127
designer, 3, 6, 9, 13, 17, 22, 28, 29, 32–36, 40, 43, 54, 55, 57, 58, 76, 79, 92, 116, 118, 122,

- 125, 126, 130–132, 134, 138, 141, 143
- detail
 - design, 26, 27, 65
 - phase, 19, 24, 47, 52, 67, 70, 98, 116, 121, 125
- development, 3, *see* life-cycle, model, *see* phase
 - cost, 51, 141
 - engineering, 2, 5, 8, 34
 - environment, 18
 - integrated product, 87, 90
 - methodology, 4, 9, 18, 35–37, 40, 44, 52, 56, 63, 65, 99
 - practice, 35
 - principle rules, 28
 - problem, 57, 82, 83
 - process, 2–5, 7–10, 12, 15–18, 28, 32–37, 40, 42–47, 52, 53, 55–58, 65, 70, 75, 76, 78–80, 82, 83, 86, 87, 92, 98–100, 138, 141–143
 - descriptive, 3, 17, 28, 29, 33, 34, 129, 131, 132
 - prescriptive, 3, 8, 9, 17, 22–25, 28, 36, 82, 90, 98, 116, 118, 119, 121, 123, 125, 127
 - product, 3, 4, 16, 32, 37, 40, 42, 45, 49, 51, 64, 75, 82
 - project, 4, 5, 9, 18, 32, 34–36, 40, 42, 46, 54–56, 63, 70, 72, 73, 75, 80, 82, 84–86, 92, 98–101, 143
 - support, 91
 - team, 3, 15, 40, 76, 127
- Diederich, O., 32
- Dorst, K., 54
- Drake, J., 28
- Dumas, A., 3, 41, 43
- Dylla, N., 32, 34, 54

- Eckert, C.M., 34
- Eder, E.E., 7, 33
- Eder, W.E., 32, 35, 40, 108, 138
- Ehrlenspiel, K., 17, 32, 34, 35, 40, 54, 134, 141, 142
- Eisenhardt, K.M., 43, 49
- embodiment, *see* phase
- engineer, 5, 10, 12, 13, 29, 30, 33, 35, 37, 42, 55, 63–65, 75, 76, 80, 82, 87, 92
- engineer-to-order, 44, 64, 75, 87, 143
- engineering, 43, 62, 101, *see* design, development, industrial
 - computer, 100
 - concurrent, 51, 87, 90, 125
 - drawings, 15, 26, 116
 - management, 36, 73, 75, 91, 100, 123
 - plan, 91
 - requirements, 125
 - reverse, 73
 - simultaneous, 127
 - system, 16, 19, 22, 24, 87, 90, 123, 125
- environment
 - company, 4, 33, 41, 42, 72, 137
 - computer, 83, 90–92, 98, 100, 101
 - development, 3, 7, 29, 41
 - factor, variable, 9
 - product, 131
 - project, 7, 9
- Ernst, 40
- Evbuomwan, N.F.O., 3, 13, 15, 16, 36, 47

- Fabrycky, W.J., 16, 123
- FIDR Workshop, 4, 32
- framework, 2–5, 16, 22, 23, 36, 37, 39–41, 63, 79–81, 84, 86, 87, 90, 91, 98
 - variable, 55
 - variables, 5, 9, 63, 74, 137
- Franke, H.J., 33, 35, 49
- French, M.J., 16, 24, 90, 116
- Frost, R.B., 32–35, 40, 48–53, 76
- Fulcher, A.J., 4, 35

- Günther, J., 35, 47, 51, 52, 55, 76, 134
- Gero, J.S., 29, 130, 131
- Gill, H., 24, 26
- Gouvinhas, R.P., 33, 35
- Green, L.N., 18, 25, 104

- Ha, A.Y., 126
- Hales, C., 32, 35, 40, 138, 139
- Hansen, C.T., 134
- Hein, L., 3, 28, 37, 126, 127, 141
- Hillier, W., 28, 29
- Hills, P., 4, 35
- Hollier, R.H., 40, 138, 140
- Horváth, I., 5, 6
- Hubbard, D., 59
- Hubka, V., 7, 33, 138
- Hybs, I., 29, 130, 131
- Hykin, D.H.W., 40, 50, 141

- industrial
 - engineering, 79
 - practice, 31–33, 80, 81
 - project, 2
- industrialisation, *see* phase
- Indutech, 90
- information
 - content, 82–84
 - orientated control, 83
 - repository, 90, 91
 - requirements, 82
- information orientated control, 82
- innovation, 48, 49, 75, *see* project, 142–144
 - product, 3
- ISO 31000, 84

- Jing, N., 59
- Johannes, R., 13, 29, 131
- Jones, D.A., 42, 44, 46
- Jones, J.C., 28
- Juster, N.P., 15

- Kappel, T.A., 90
- Koeller, C.T., 42

Kostoff, R., 90
 Krause, F.L., 15, 16, 25

 Laming, L.C., 40, 50, 141
 Legardeur, J., 57
 Lettice, F., 48, 49
 life-cycle, 3, 16, 90, 123, 126
 development, 18, 19, 33, 41–55, 63, 74, 92
 product, 3, 8, 11, 15, 16, 18, 22, 24, 25, 27, 32, 43, 65, 71, 82, 87, 132
 Lindemann, U., 40, 127
 Lockheed Martin, 73
 Lu, S.C.-Y., 59
 Lutters, D., 44, 50, 82, 83, 90

 Maffin, D.J.B., 4, 9, 25, 33, 37, 38, 40, 41, 43, 51, 105, 142–144
 management, *see* engineering
 risk, 59, 84
 manufacture-to-order, 75, 140, 142
 manufacture-to-stock, 142, 143
 manufacturing, 28, *see* production, 123, 126, 134, 142
 company, 42, 56, 85
 industry, 4, 9, 43, 138
 instructions, 2, 26, 117
 life-cycle, 16
 process, 18, 78, 83, 121, 126, 131, 139, 143
 specification, 26
 March, L.J., 17, 28, 130
 Martin, P., 106
 Marver, T.W., 28
 Marxt, C., 3
 maturity, *see* organisation, personnel
 McDonagh Philp, D., 35
 McNeese, M.D., 37
 method
 design, 2, 18, 19, 35, 103, 104, 106
 methodology, 3, *see* development, 134
 project, 4, 9, 36, 57, 70, 77, 78, 80, 84–87, 98, 99
 model
 CAD, 66
 development process, 2–4, 8, 9, 16, 18, 21, 24, 26, 28–30, 33–35, 70, 76, 80, 82, 86, 87, 90, 92, 98, 99, 121, 123, 125, 127, 129–131, 133, 134
 product, 3, 72, 125

 Ng, K.W., 3, 17, 22, 26
 Nordlund, M., 37
 novelty, *see* innovation, project

 organisation
 capability, 55
 design capacity, 45, 56, 58, 86
 maturity, 44, 58, 85
 size, 42, 54, 56
 structure, 43
 type, 43, 56

 Pahl, G., 24, 27, 35, 87, 90, 118, 119, 121, 122
 Peirce, C.S., 130
 personnel, *see* team
 design capability, 55, 58, 86
 maturity, 54, 58
 team size, 54, 56
 Pfeuffer, R., 32
 Phaal, R., 90
 Pham, D.T., 91
 phase, *see* concept, detail, development
 design, 8, 15, 51, 52, 66, 78, 87
 detail, 22, 116
 development, 16, 18, 24, 27, 123
 disposal, 15, 87
 distribution, 15, 87
 embodiment, 22, 52, 55, 65, 66, 69, 116
 industrialisation, 15, 65, 67, 70, 87, 123
 production, 15, 51, 65, 67
 prototype build, 67
 prototype test, 67
 quotation, 65
 research, 15
 strategy, 15, 87
 usage, 15
 Porteus, E.L., 126
 Potter, S., 42
 prescriptive, *see* development
 process, *see* development, manufacturing, model, production
 product, *see* complexity, development, environment, innovation, life-cycle, model
 hierarchy, 52
 type, 50, 56
 production, 26, 40, 73, 75, 87, *see* manufacturing
 batch, 44, 50, 64, 76
 control, 138
 cost, 33, 51, 86
 datapack, 72
 documentation, 32, 82, 121
 mass, 44, 50, 70, 76, 86, 142
 method, 22, 116, 126
 personnel, 22
 process, 12, 15, 46, 51, 72, 75, 82, 116, 143
 quantity, 40, 50, 51, 142
 size, 44, 50
 team, 82
 project, 2, 3, 18, 23, 25, 30, 35, 36, 46, 83–86, 98, 99, *see* complexity, development, environment, industrial, methodology
 complexity, 46
 constraints, 48
 design, 2
 novelty, 48
 size, 46, 56
 type, 47
 Pugh, S., 22, 23, 33, 36, 87, 127

Quinn, J.B., [32](#), [34](#), [133](#)

Radcliff, D.F., [4](#), [42](#)
 repository, *see* information
 requirement, *see* analysis, customer, engineering, information
 research, [5](#), [7](#), [8](#), [16](#), [18](#), [36](#), [40](#), [80](#), [83](#), [100](#), [101](#),
 see phase
 design, [4](#)
 market, [25](#)
 project, [4](#)
 question, [4](#), [98](#)

Reymen, I.M.M.J., [26](#), [28](#), [29](#), [32](#), [40](#), [132](#), [133](#)

Rohatynski, R., [35](#)

Roozenberg, N., [28](#)

Roozenburg, N., [9](#), [17](#), [26](#), [29](#)

Rothwell, R., [42](#)

Roy, R., [42](#)

Rzevski, G., [25](#), [57](#), [86](#)

Schnelle, E., [32](#)

Schulz, A., [12](#)

Shenhar, A.J., [46](#), [49](#), [52](#)

Shigley, J.E., [12](#)

Simon, H.A., [133](#)

size, *see* organisation, personnel, production, project

Song, X.M., [55](#)

Stempfle, J., [29](#), [33](#), [133](#)

Stetter, R., [106](#)

Suireg, A., [35](#), [87](#), [118](#)

Sutinen, K., [125](#)

system, *see* configuration, engineering

team, *see* approach, development, personnel, production

Thomond, P., [48](#), [49](#)

Thompson, G., [125](#)

Thronley, D.G., [28](#)

Tjalve, E., [40](#), [138](#), [140](#)

Tomiyama, T., [125](#)

Trassert, P., [33](#)

type, *see* organisation, product, project

Ullman, D.G., [13](#), [16](#), [25](#), [47](#), [87](#), [121](#), [122](#)

Van de Ven, A.H., [40](#)

variable, *see* context

VDI 2221, [4](#), [13](#), [16](#), [23](#), [35](#), [47](#), [70](#), [76](#), [106](#), [119](#), [120](#)

Vergeest, J.S.M., [5](#), [6](#)

Veron, P., [106](#)

Von Handenhoven, E., [33](#)

Wallace, K.M., [40](#), [138](#), [139](#)

Walsh, V., [42](#)

Warell, A., [2](#), [3](#)

Whitfield, A.D., [3](#), [41](#)

Wideman, A., [52](#)

Wideman, R.M., [46](#), [49](#)

Young, [40](#)

Zanker, W., [32](#), [33](#), [35](#)

Zegveld, W., [42](#)