3 METHODOLOGY

3.1 Introduction

Chapter 2 developed a number of hypotheses and research question issues. Chapter 3 describes the methodologies used to answer these research issues, and aims to show that the decisions that have been taken are supported by either authorities, evidence, or logic.

Chapter 3 is also the starting point of a new CAPP software project called SACAPP (South African CAPP), that considered throughout its development that the organizational and managerial aspects of software development were at least as important as its technical aspects (van Vliet, 2000) (Figure 3.1).

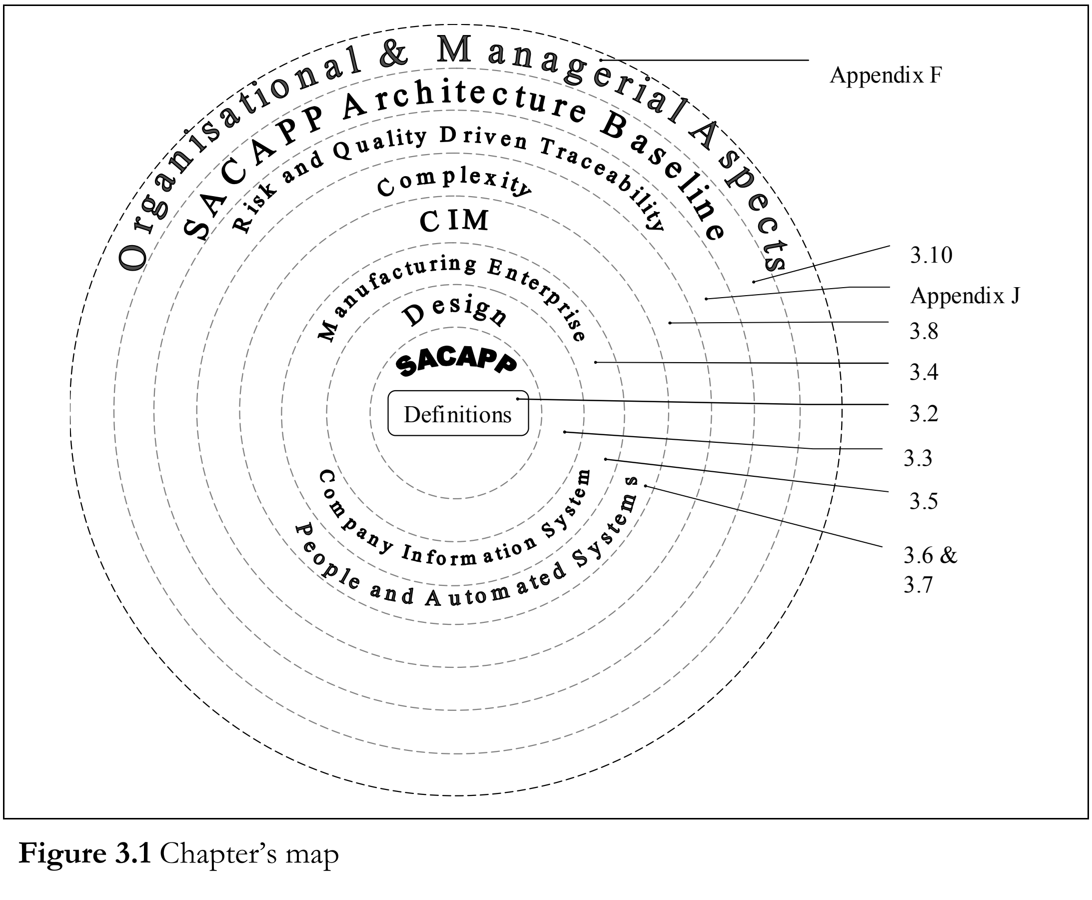


Figure 3.1 Chapter's map

The chapter starts with the SACAPP roadmap that represents the skeleton of the whole project (Appendix F), followed by the project vision for the new system (Appendix G), and business modeling (Appendix H). Then, in order to cover all their pertaining aspects and keep their traceability, the hypothesis and research questions developed in Chapter 2 were broken down into smaller manageable pieces (Table 3.1), and theoretical answers given in sections 3.2 to 3.10. The chapter finally ends with section 3.11., conclusions.

Table 3.1 Hypothesis and research question analysis

	The broken down hypothesis and research questions	Traceability
Research Question	**How to develop the architecture of a new CAPP software system that takes into consideration the specific application requirements, the enabling technologies, the existing company resources and culture, and also minimises the software development risks and time.**	3.10
First hypothesis,	**A CAD system that uses common designs and manufacturing objects and preserves most of the actual design representations will**	3.3
	enhance CAD/CAPP communication, and	3.3
	lead to the development and implementation of a better CAPP software system.	
Second hypothesis,	**Decomposing CAPP's complex problems into smaller more manageable sub-problems,**	3.5
	keeping human in the systems loop, and	3.6 & 3.7
	better alignment of the architecture of a new modular CAPP software system with the organizational structure of the engineering company, its characteristics, manufacturing concepts used in practice, new technologies, business practices, manufacturing processes,	3.4

	the need for information, new technologies, and the new trends in the IT infrastructure,	3.5
	will lead to the development and implementation of a better aggregate CAPP software system.	
Third hypothesis,	**Simplifying information complexity,**	3.5
	use of automation principles and strategies,	3.6 & 3.7
	and the inclusion of CAD, CAPP, and other categories of data in the communication part of CIM	3.3
	will lead to the development and implementation of a better CAPP software system.	

3.2 Definitions Considered in this Thesis

In this thesis, process planning was defined (see Chapter 1) as:

"Process planning" is a multi-perspective technical activity with a scientific and engineering background that establishes the process that transforms a starting material into a final product. The effectiveness of this manufacturing process is largely decided by the technical product requirements complemented by the customer specific requirements, and a combination of the company's technical resources, available skills, prevalent culture, strategies, procedures, organisational structure, and technological systems.

The definition was based on a number of previous process planning definitions (Kiritsis 1995, Rozenfeld and Kerry 1999, Law et al. 2001, Twigg 2002, Scallan 2003, Usher 2003), but it also brought new and multi-faced aspects. For example, apart from the engineering drawing, it considered new aspects such as the managerial decisions, local practices and knowledge, sales, estimation, and customer specific data. Also, as opposed to Dorf and Kusiak (1994), it considered unnecessary the engineering drawing details, and, as opposed to Kiritsis (1995), it considered process planning a technical specification that could not create alone an overall effective production unit because the production was governed by other specific activities such as production planning and production scheduling.

Also, in this thesis, the term process planning was used only to represent the machining process because, it has been the most demanding researched area (Cay and Chassapis 1997, Law and Tam 2000, Ciurana et al. 2003), still caused a major bottleneck to CAPP implementation (Ahmad et al., 2001), and had always generated the need for the development and introduction of new computer-based applications (Maropoulos 1995, Wright 2001).

Furthermore, based on a combination of definitions from ElMaraghy (1993), Sormaz and Khoshnevis (1997), and Hollingum (1999), *the process was considered the flow of products from one worker to another, within given constrains and optimisation criteria; the operation was considered a manufacturing discrete stage, performed on a single machine with a single set-up, at which a worker may add value to a product.* Finally, the relationship between process and operation was considered as production networks that focused on the process that only could become efficient through radical process improvements supplemented by operation improvements (Hollingum, 1999).

3.3 SACAPP and Design

In this thesis, the "high-level" creative phase of design was viewed as a conceptual map between the functional requirements (FRs) and the design parameters (DPs), extended then to a conceptual map between the DPs and the process planning parameters (PPs), which also was extended to a conceptual map between the PPs and the company's facilities (CFs) and rules (CRs) parameters, therefore creating high-level conceptual maps that linked the design, process planning, and production (Figure 3.2).

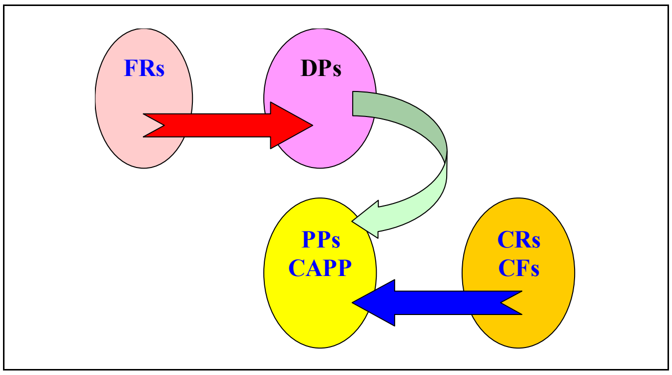


Figure 3.2 High-level design, process planning, and production conceptual maps

So, by pursuing the goal to establish a meaningful relationship between the design and the process plan, the example in Figure 3.3 was given, and some first relationships were established (Table 3.2).

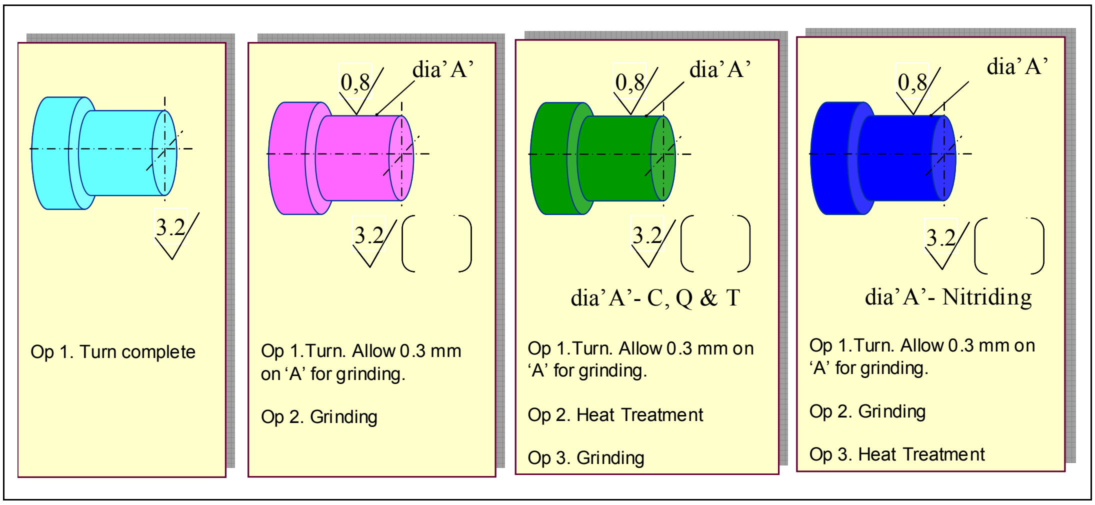


Figure 3.3 Four simple and similar drawings with no dimensions and no tolerances, where it was still possible to develop correct process plans by only considering, and combining the surface finish, and the heat treatments drawing instructions

Following this, a very complex item for an aerospace engine was considered using the same style as in the previous example (Figure 3.4). It was again observed that, although the drawing had no dimensions and no tolerances, and no specific vertices, surface, or volumes were used, it was still possible to develop the process plan (Figure 3.5).

Table 3.2 Relationships - design/ process planning rules and instructions

Structural fact: each final item needs a starting material
Action triggering: each starting material requires a receiving inspection
Inference: if the item is round and general surface finish equals 3.2 microns: "Turn complete"; if the item is round, general surface finish equals 3.2 microns, and at least one surface requires 0.8 microns: "Turn. Allow 0.3 mm on the surface that requires 0.8 microns. Grind the journal that requires 0.8 microns"; if the item is round, general surface finish equals 3.2 microns, at least one surface requires 0.8 microns, and the heat treatment of C, Q, & T (Carburising, Quench and Tempering) is required: "Turn. Allow 0.3 mm on the surfaces that requires 0.8 microns. Heat-treat according to C, Q, & T process. Grind the surfaces that require 0.8 microns."

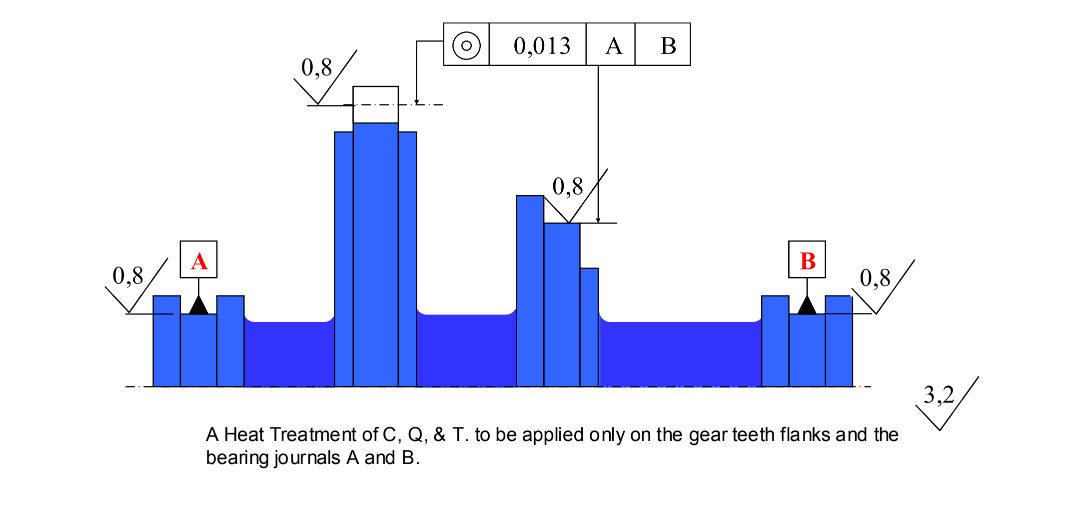


Figure 3.4 Typical gear-shaft for an aerospace engine (the condition of concentricity of 0.013 mm could be implemented through a proper process plan sequence)

Consequently, a high-level conceptual map that linked a limited number of design elements with well established manufacturing knowledge was developed (Figure 3.5) without the need to recognise the very detailed design information (Feng et al. 1999, Chang and Chang 2000, Feng and Song, 2000b) which was not necessary in process planning (McMahon et al., 1997), and was a waste of time, cost, and efficiency (Chang and Chang, 2000). Also, in order to bring together the design and manufacturing data, SACAPP coded the data from Figure 2.3 and Table 3.3, and so, used a simple methodology to transfer the tolerance specifications to the surface finish specifications, and avoided the risks associated with the tolerance representation and interpretation (AAAI, 1997).

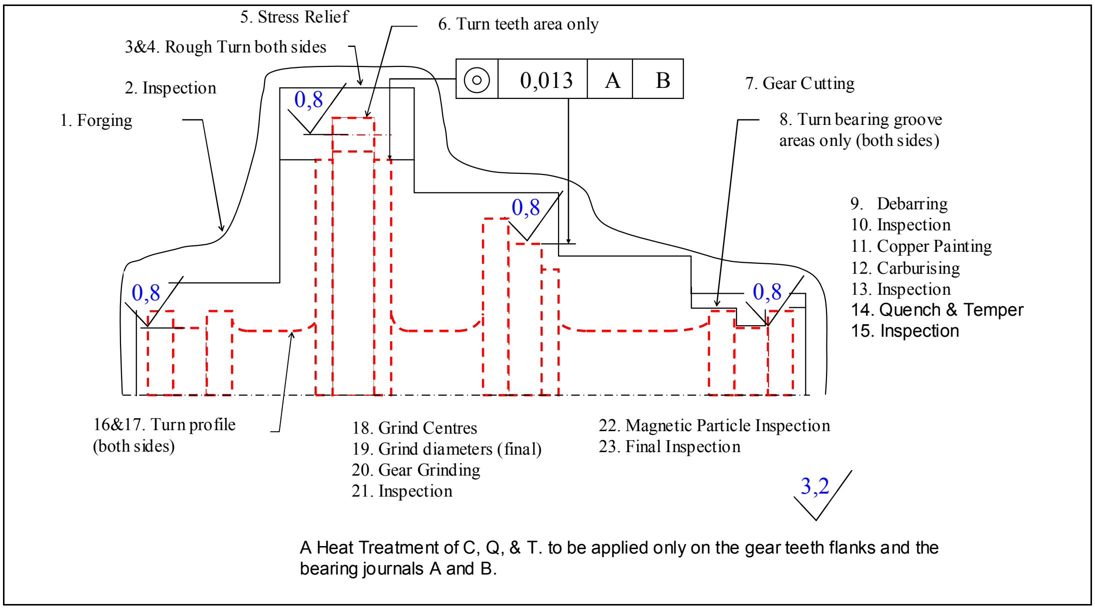
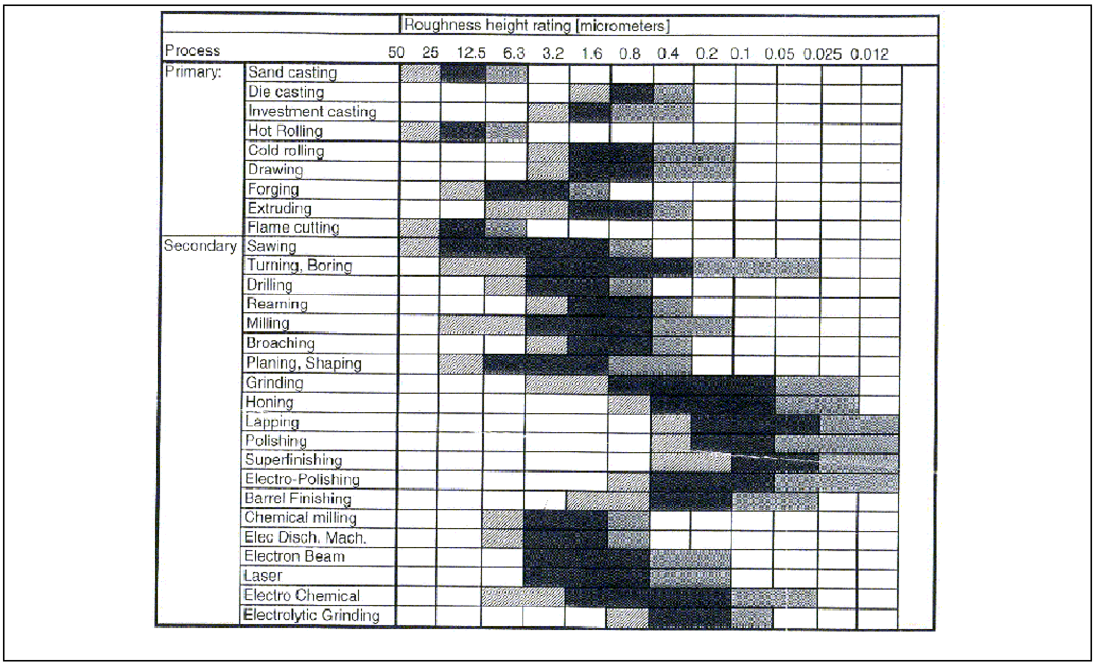


Figure 3.5 The final item (dotted lines) and the process planning operations

Table 3.3 Process-surface roughness link (economical range in solid colour) (DeGarmo et al., 2003)



Furthermore, in the conditions when the vast majority of engineering data was non-geometric in nature (AAAI, 1997), SACAPP considered that the drafting symbols and text represented the conscious knowledge explicitly represented, examined, and manipulated (Bigus and Bigus, 2001) that could provide the designer with an added flexibility in design (Kuric and Janec, 1998), yield enough input to determine manufacturing process (Feng et al., 1999, Feng and Song, 2000a), and that could be represented inside a computer by character strings or by numbers.

The fact that SACAPP considered the drafting symbols and text for communication was not surprising. Symbols have been used as communication for centuries in arts, mathematics, mechanical engineering designs, or electrical circuits' designs. As Leonardo de Vinci realised, the power of using images and associations in order to unleash problem solving and see new creative pathways (Buzan, 2002), in modern times, the UML standard modeling language as the best current software development practices (Reed, 2002), has been used as the graphical notations for visualizing, specifying, constructing, and documenting all the software blueprints (Jacobson et al., 1999) (Note: The readers not familiar with UML and RUP are invited to see Appendix M – Introduction to the UML).

Therefore, just as the UML has been the answer to the two key challenges in software development, namely system's complexity and communication, the thesis used the same approach in CAD/CAPP communication because engineering design's complexity could, as opposed to individual feature extraction, be "understood" better when displayed as drafting symbols and text with meaning behind them for visualizing, specifying, constructing, and documenting part of the design artifacts (Figure 3.6). Consequently, this thesis, proposes and develops a prototype CAD system called SADwO (South African Design with Objects), that used machining features with relevant technological information (Kumar and Roman 1997, Feng et al. 1999, Kang et at. 2003).

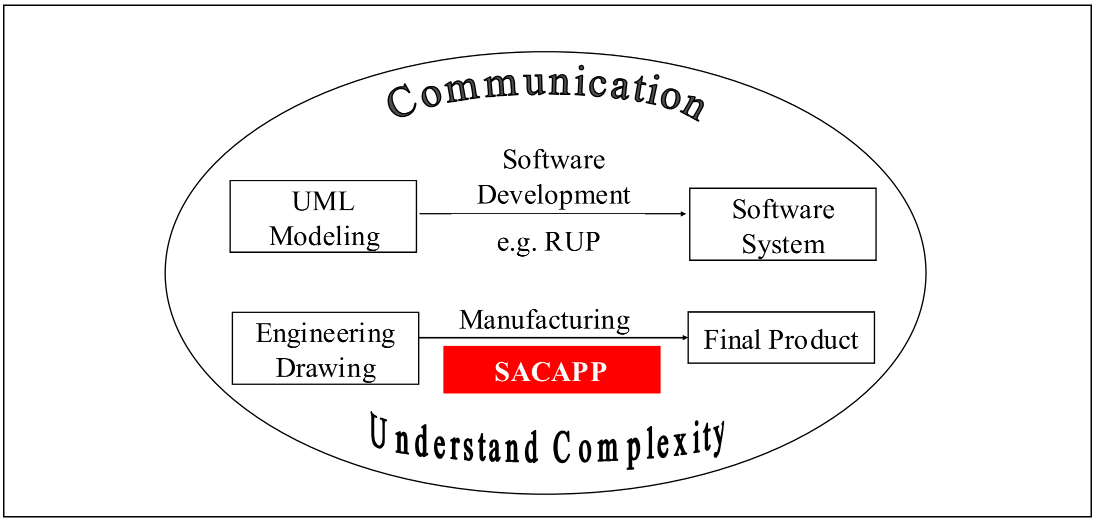


Figure 3.6 UML, design, RUP, manufacturing relationships

So, in order to develop the SADwO system, the example from Figure 3.3 was revised, then, a number of entity, boundary, and control objects were identified (Figures 3.7 and 3.8), and after that, it is shown how these objects worked step-by-step together in order to implement one of the flows through the functionality in the use-case example (Figures 3.9 and 3.10).

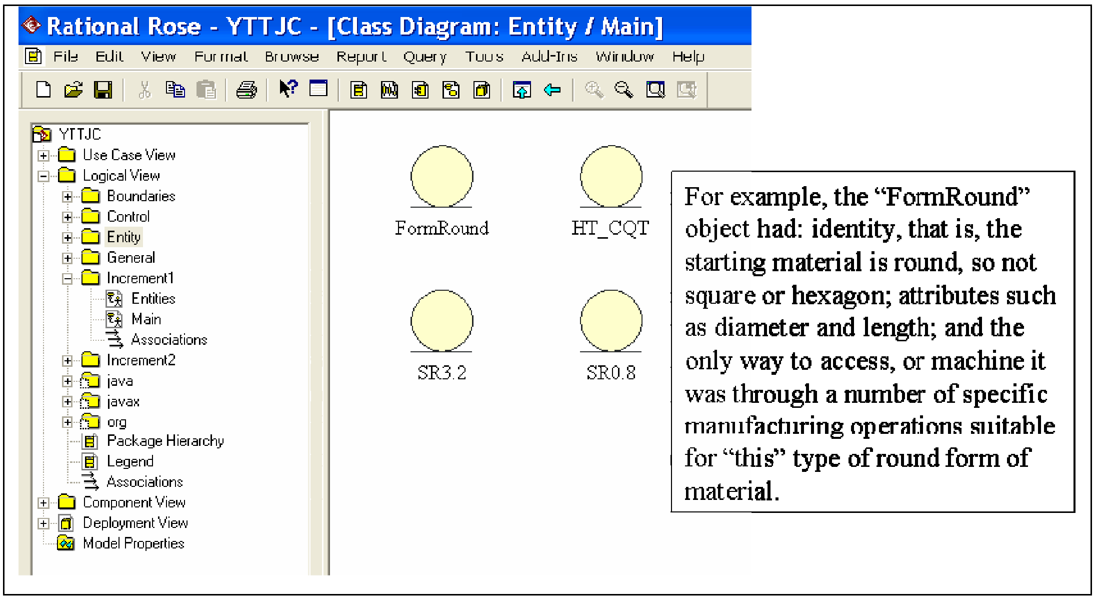
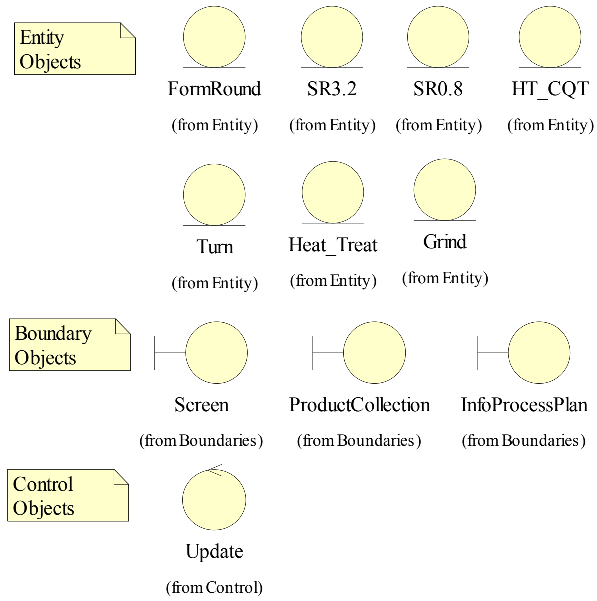


Figure 3.7 UML Class diagram example (first iteration)



For example, the "FormRound" object had: identity, that is, the starting material is round, so not square or hexagon; attributes such as diameter and length; and the only way to access, or machine it was through a number of specific manufacturing operations suitable for "this" type of round form of material.

Figure 3.8 UML Class diagram example (second iteration)

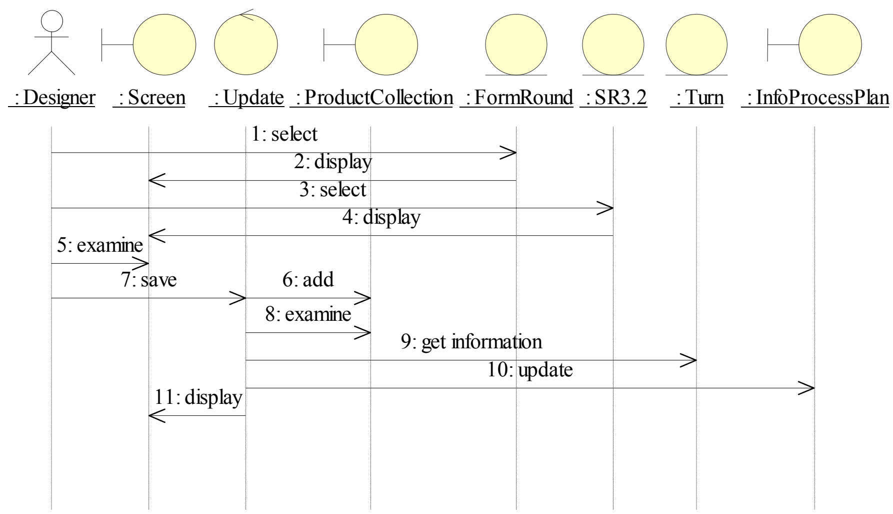
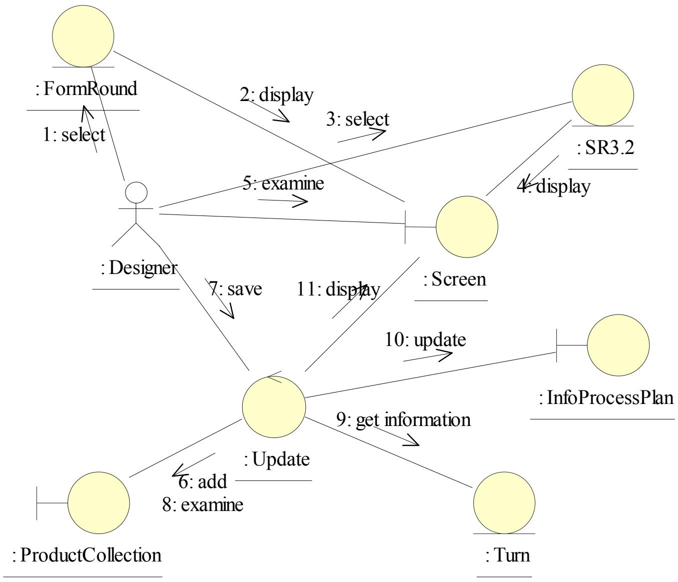


Figure 3.9 The UML sequence diagram, organised by time, showing: one of the flows through the example; the actor which initiated the whole flow; the objects that are needed for the flow; and the messages the objects sent to each other to assign responsibilities (each message will later become an operation name for the class).



The objects that directly communicate with each other were shown with a line drawn between them so, the absence of a line means that no communication occurs directly between objects.

Figure 3.10 The UML collaboration diagram, showing exactly the same information as the UML Sequence diagram but without reference to the time.

In addition, when dealing with a greater number of objects such as in Figure 3.5, SACAPP used centres of process gravity, which used the same principles similar to the group technology and modularity manufacturing concepts (Marshall and Leaney, 1999), also similar to the modular software principles to make its architecture work in the real world (Kazman and Bass, 2002); and also similar to the automation and robotics recommendations, which required the determination on all sites a central focus or series of central focuses about which all other activities revolved (Cusack 1994, Kazman and Bass 2002).

Consequently, in order to develop a technique to be used for constructing process planning sequence of operations, the operations indicated in Figure 3.5 were broken down into smaller, independent, self-contained, and co-operative groups of operations, and doing so, transformed the process plan of a complex item into a sequence of nine modules (Table 3.4). Then, the nine modules have been placed into a graphical format, and observed that they resembled an array of objects (Figure 3.11). Finally, by going through each of the array's elements and writing their contents, the process plan's sequence of operations was possible to be constructed, and so achieving the most critical activities of the process planner's activity (Wong and Siu, 1995) because it involved knowledge about facts, procedures, "if-then" rules (Wong and Siu, 1995), machine-tools availability, manufacturing time (Kumara and Rajotia, 2003), and human-resident experience or established local practice (Esawi and Ashby, 1998).

Table 3.4 Groups of operations and their centres of gravity (CG) classification in "common sense" (CS), e.g. the start and the end of a process plan; "critical", a central focus about which all other activities revolve; and "technological" (Tech), that represent the process planner's specific knowledge

No	Content of the module	Classification
0	1. Forging, 2. Inspection	CS
1	3&4. Rough Turn, 5. Stress Relief	Tech
2	6. Turn	CS
3	7. Gear Cut	Critical
4	8. Turn	Tech
5	9. Debarring, 10. Inspection, 11. Copper Painting, 12. Carburising, 13. Inspection, 14. Quench & Temper, 15. Inspection	Critical
6	16. & 17. Turn profile both sides	CS
7	18. Grind Centres, 19. Grind diameters, 20. Gear Grinding, 21. Inspection, 22. Magnetic Particle Inspection	CS
8	23. Final Inspection	CS

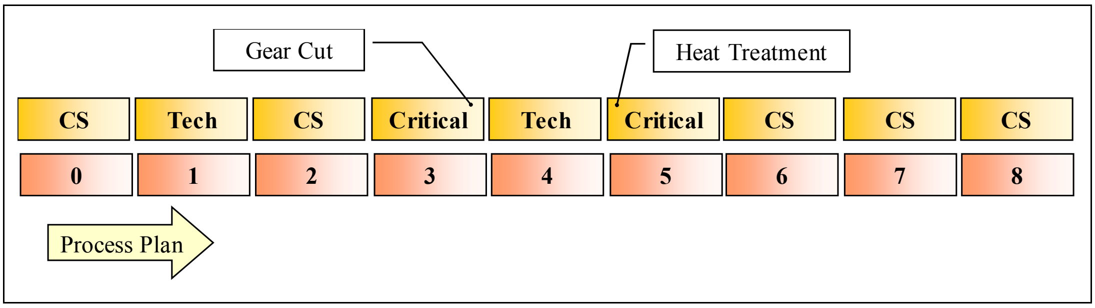


Figure 3.11 List of the centres of gravities

Furthermore, because features were considered central to design, process planning, and manufacturing tasks integration (Chep and Tricarico 1999, Tseng 1999, Ciurana et al. 2003), the different schools of thoughts about features in design and manufacturing had to be analysed in iterations (Table 3.5).

Table 3.5 Analysis of feature definitions in design (D), manufacturing (M), and their influence (I) in design (I_D) or manufacturing (I_M) (1st Iteration)

Group	Characteristics, with the reference is shown in (author, year, code) format
1	Part geometry (Shen and Norrie, 1999, M)
2	Form, shape (Liang and O'Grady, 1998, D), (Timings, 1998, I_M), (McMahon et al., 1997, M)
3	Shapes serving certain functions (Chep and Tricarico, 1999, I_D)
4	Functional meaning (Liang and O'Grady, 1998, I_D), (Martino et al., 1998, I_D), (Ciurana et al., 2003, I_D)
5	Physical properties (Liang and O'Grady, 1998, I_D)
6	Material quality (Timings, 1998, I_M), (McMahon et al., 1997, M)
7	Raw material (Shen and Norrie, 1999, I_M)
8	Heat treatment process (Timings, 1998, I_M), (McMahon et al., 1997, M)
9	Accuracy, surface finish (Timings, 1998, I_M), (McMahon et al., 1997, M)
10	Quantity (Timings, 1998, I_M)
11	Equipment process capability (Timings, 1998, I_M)
12	Cost (Timings, 1998, I_M), (McMahon et al., 1997, M)
13	Physical element with eng. meaning (Rozenfeld and Kerry, 1999, M)
14	Surfaces formed by machining processes (Chep and Tricarico, 1999, M)
15	Individual characteristic e.g. cylindrical surface, screw thread, slot (ISO TC184/SC4, 1993, D & M)
16	A unit of form with semantic meaning (Abdalla and Knight, 1994, D)
17	A set of information (Chep and Tricarico, 1999, D)
18	Functional, geometric, as well non-geometric job characteristics that can facilitate any form of computer decision making (Kunigahalli, 1998, I_M)
19	A simple feature frequently machined (Rho and Lee, 1996, M)

20	Natural association of knowledge between domains (Faraj, 2003, D & M)
21	Important or critical feature (Chang and Chang, 2000, I_D & I_M)
22	A natural way to describe a work piece (Davies, 1997, M)
23	Schematic descriptions of parts that can be task-based at all levels (Ciurana et al., 2003, D)
24	A visual image of the part with a high-level information and never the computerized raw data (Yuen et al., 2003, I_D)
25	Feature has three perspectives: shape (e.g. hole), structure (e.g. stepped hole), and information support (e.g. size, tolerance, surface finish, heat treatment) (Devireddy and Chosh, 1999, I_D)
26	The drawing features' interpretation should help identify the critical processing factors (Scallan, 2003, I_M)
27	Geometrical features that have a major influence on the selection of manufacturing processes (Scallan, 2003, I_M)
28	Available production facilities, (McMahon et al., 1997, M)

Then, during the 2nd iteration, the contents of Table 3.5 have been re-analysed by considering that each object had to be built out, according to the object-orientated programming, of (Identity, Variables, Methods), or (Identity, State, Operations) (van Vliet, 2000). The result of this analysis (Table 3.6) retained only those objects that represented a natural association between both design and process planning. After that, the 3rd final iteration established the list of design and manufacturing objects decided to be used not only in SADwO system, but also for SADwO/SACAPP communication (Table 3.7). Finally, the SADwO/SACAPP object feature definition was established, and then verified (Figures 3.12 and 3.13).

Table 3.6 Feature analysis (Iteration 2)

Group	Name	A few Examples
1	Form, Shape	Round, Square
2	Material quality	655M13, 826M40
3	Raw material	Bar, Forging

4	Surface finish	3.2 or 0.8
5	A unit of form with semantic meaning	Keyway, Hole, Gear Teeth
6	A visual image with high-level information	Surface roughness symbol

Table 3.7 SADwO objects' list (Iteration3, Final)

Group	Name	Example	Notation Example
1	Item's general shape	Round, Prismatic	Round, Prismatic
2	Material quality	655M13, 826M40	655M13, 826M40
3	Starting material	Bar, Forging	Bar, Forging
4	Heat treatment	C, Q, & T	HT_CQT
5	Surface roughness	3.2 or 0.8	SR3.2 or SR0.8
6	Keyway, Slot	Keyway, Slot	Keyway
7	Gear teeth	Gear teeth	GearTeeth
8	Hole (other that e.g. the central hole of the hollow item)	Holes	DrillHoles
9	Other textual notes	Balancing	Balancing

A SADwO/SACAPP object feature is a unitary, consistent, homogeneous, and conscious conceptual entity *model of some part of the real world of both designer and process planner which has identity, information and representation independence, syntax behind it, and meaning, that is semantics, and used for visualizing, specifying, constructing, and documenting part of their artifacts.*

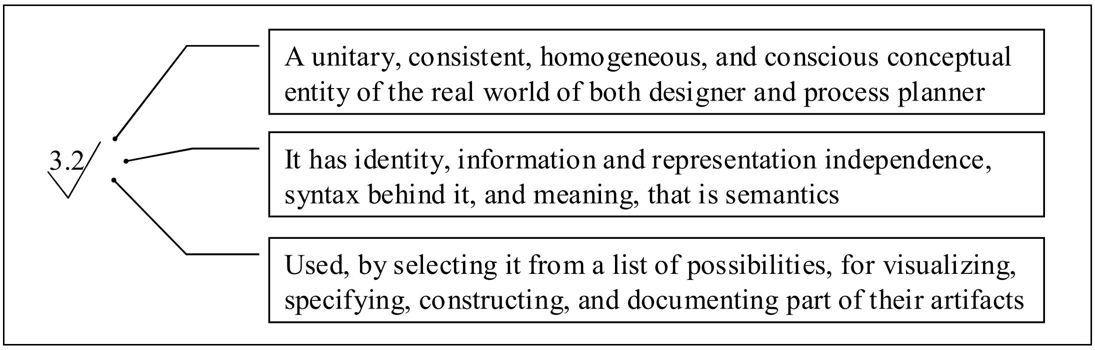


Figure 3.12 SADwO/SACAPP feature example 1

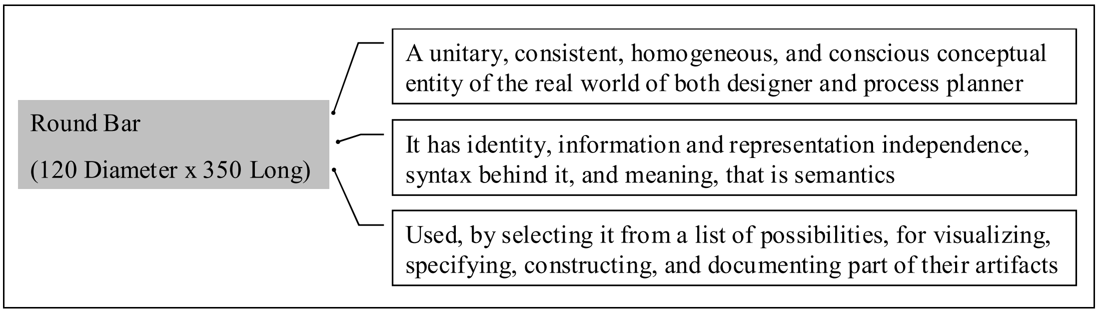


Figure 3.13 SADwO/SACAPP feature example 2

3.4 SACAPP and Manufacturing Systems

The research literature showed that both actual CAPP and manufacturing systems have had fundamental structural defects that represented important impediments within the production facility (OMG 1996, Yan et al. 2001) (Figure 3.14).

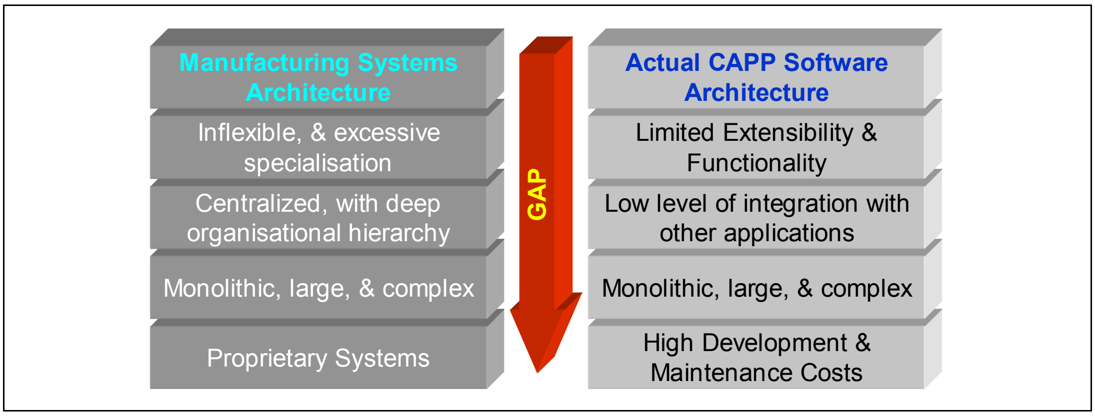


Figure 3.14 Fundamental structural defects for CAPP and manufacturing systems (Rodd et al. 1992, Brandon 1993, Feng and Zhang 1998, Law and Tam 2000, Groover 2001, Wang et al. 2002, Lee 2003 - reorganised)

Therefore, in order to understand their complexity, Figure 3.15 was constructed and found that large runs are associated with continuous or mass production while small runs, where CAPP was most required, were associated with discrete or batch production where 75 per cent of parts were made in lots of 50 or less (DeGarmo et al, 2003), and where the flexible equipment could accommodate a great variety of operations sequences for different part configurations. Also,

Figure 3.16 showed that, in order to cope with the wide product variety and complexity of the small runs, enterprises employed highly skilled labour to use the general purpose equipment arranged into a process layout for maximum workshop flexibility.

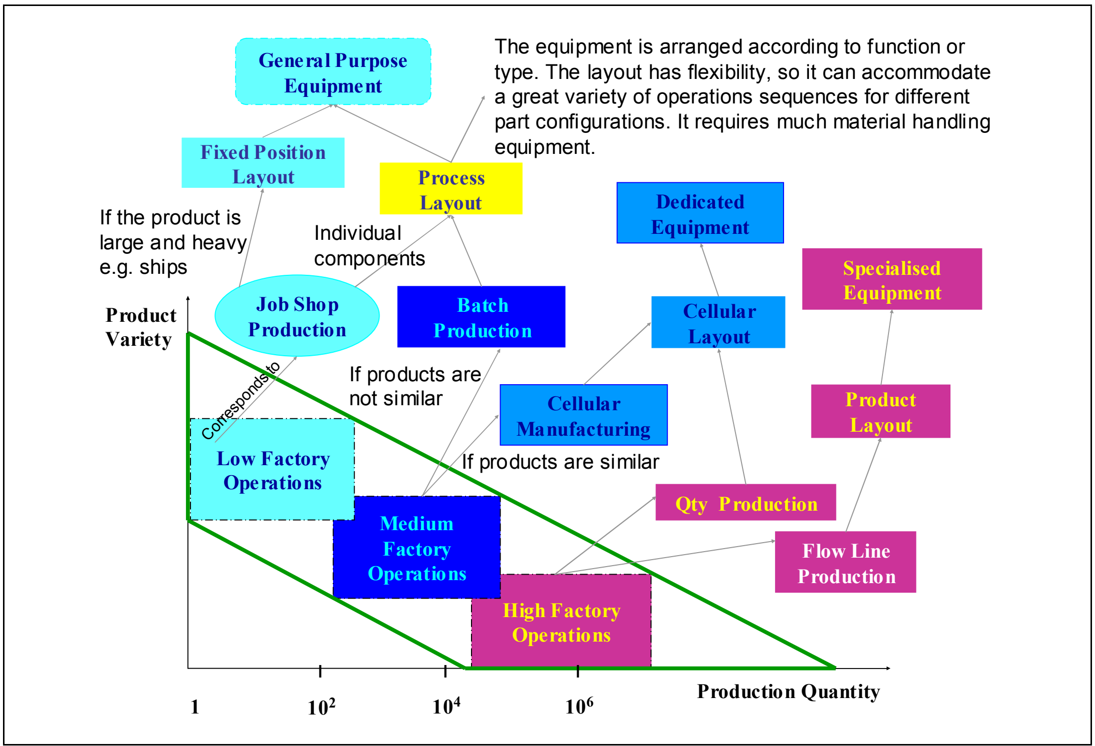


Figure 3.15 Manufacturing systems relationships

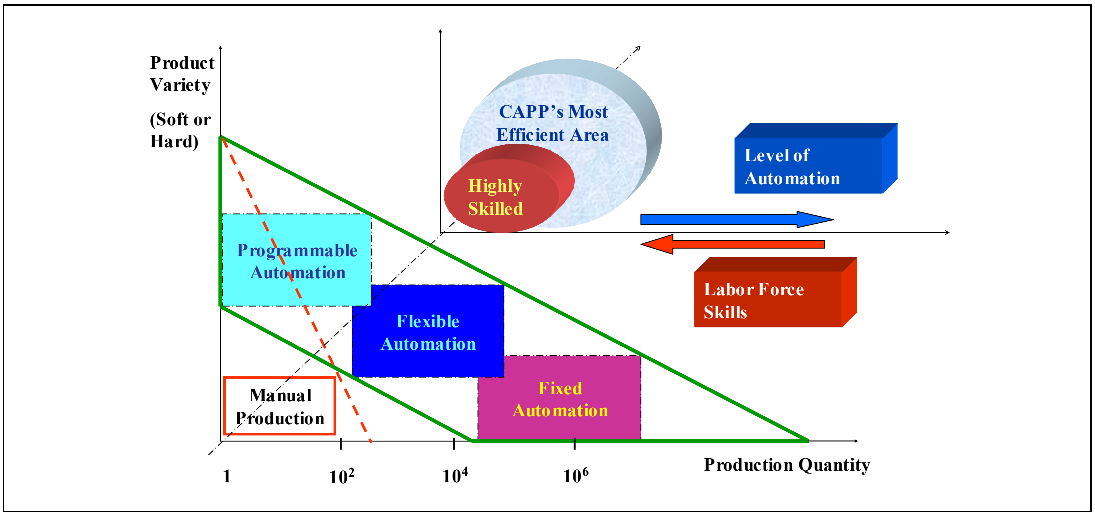


Figure 3.16 Manufacturing enterprise automation and people skills

As a result, in SACAPP, as opposed to Zhang et al. (1996), Chen and Kumara (1998), Fernandes and Raja (2000), and Huang and Xu (2003) which intended to include the detailed process parameters in the process plans, it was considered more important to support the skilled machine-tool operator, which referred to nothing but catalogues and handbooks (Whybrew and Britton 1997, Yan et al. 2001), with new tools capable to provide and capture specific data including the detailed process parameters (Figure 3.17).

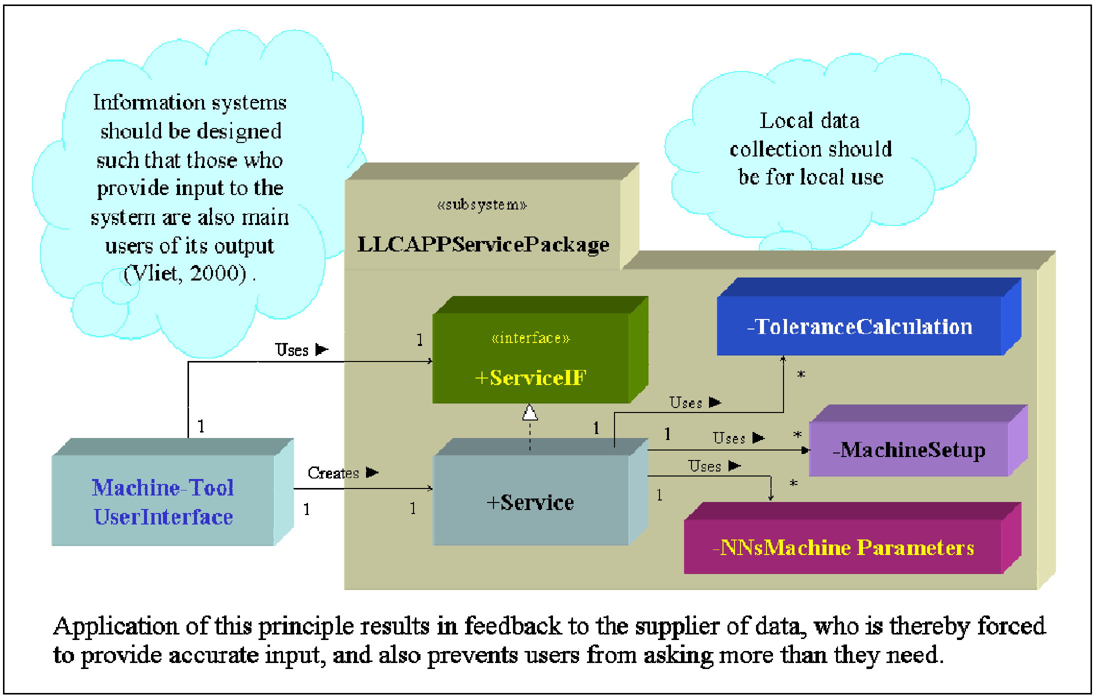


Figure 3.17 Closed loop principle applied at machine-tool level that also made the process plans generation simpler

Therefore, by adopting this solution, SACAPP, not only avoided the risks associated with the actual CAPP systems such as the high-maintenance costs and the incorrect generated data discarded by the highly skilled operators, but it also achieved the right balance between the manual and automated tasks (Maropoulos, 1995), reduced CAPP system's costs (Marri et al., 1998), improved functionalities at the various company levels (Zhang 1994, Shin 1998), and considered the cultural concepts of the society in which the system will operate (Petrarolo 1995, Turner, 1998).

Consequently, SACAPP was based on realistic approaches (Shin, 1998), used carefully crafted computational approaches (Kusiak, 2000), explicitly incorporated the human at the design stage (Bomba 1998, Sreeram and Chawdhry 1999), and considered how people, machines, and information technology could work together beneficially (Price 1998, Shafto and Hoffman 2002) and collectively at various stages of the product development (Shen and Norrie, 1999).

Finally, similar to manufacturing systems that needed to cope with their own complexity and dynamic operations, SACAPP used manufacturing concepts (Figure 3.18), applied manufacturing business rules (Figure 3.19), and was organised and managed under appropriate architecture (Wang et al., 2002) represented by an integrated set of sub-systems built around the main functions of the system, linked according to the way processing advanced, and some means of controlling the subsystems and the overall system (Scallan, 2003).

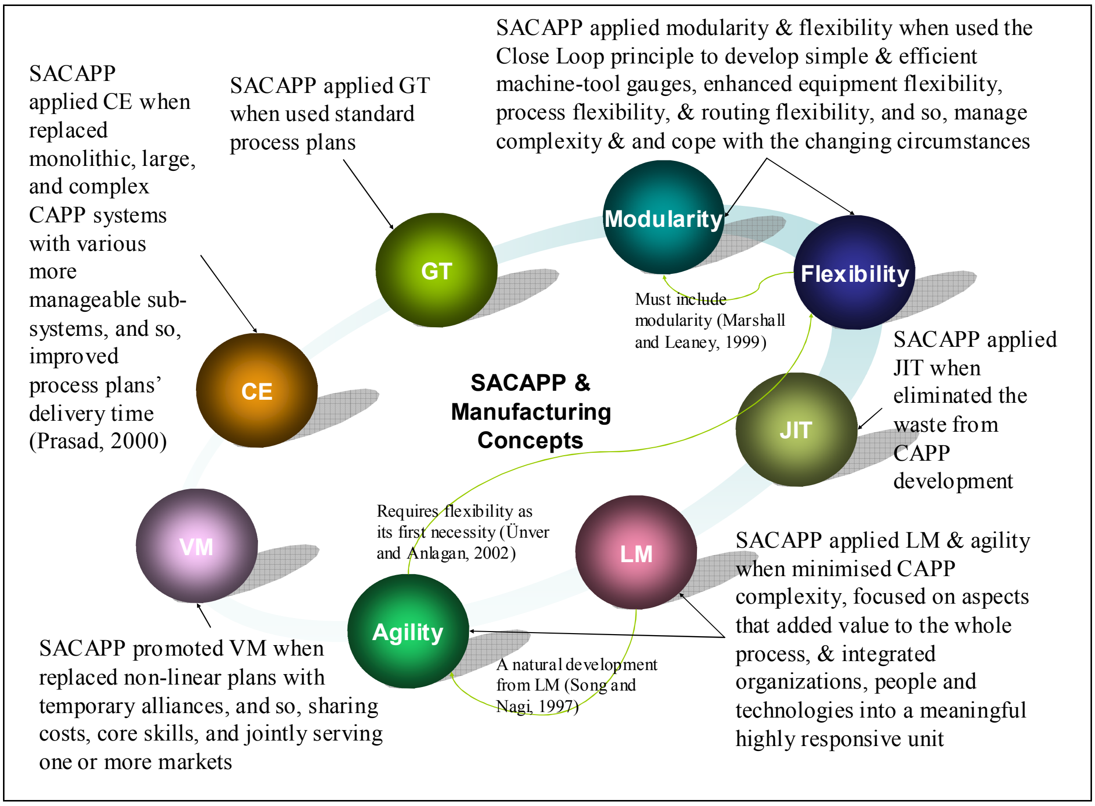


Figure 3.18 SACAPP and manufacturing concepts

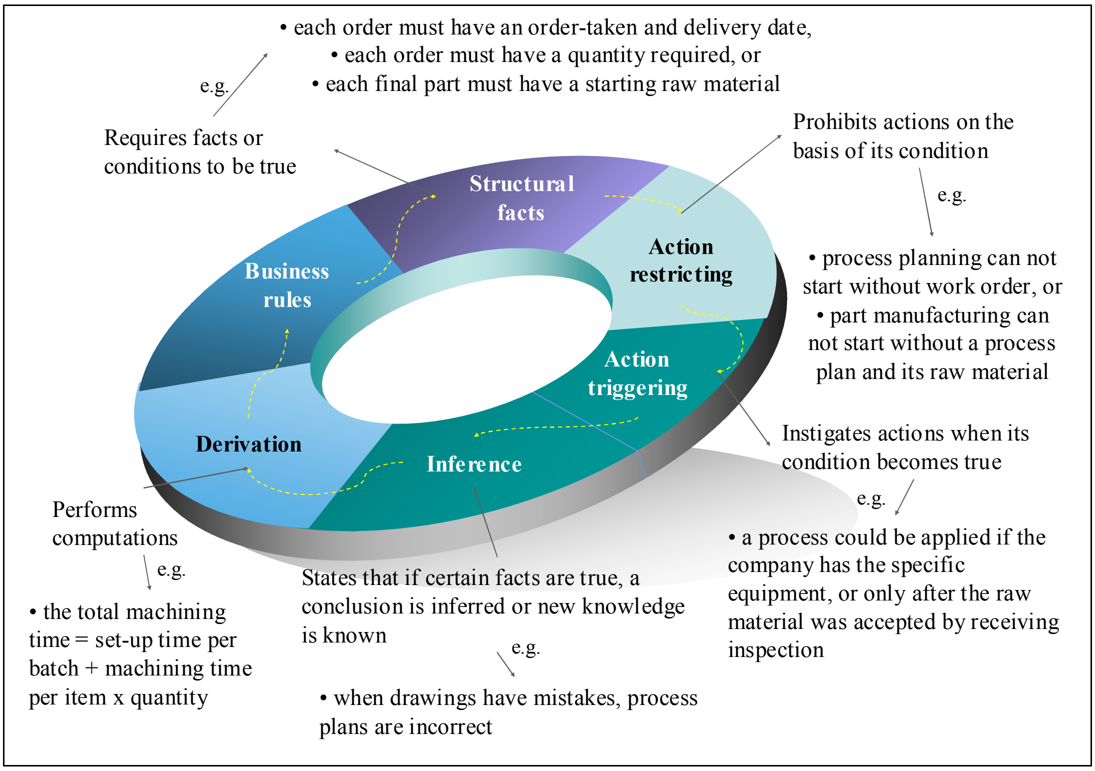


Figure 3.19 Categories of business rules

3.5 SACAPP and Company Information System

The company information system was considered to interconnect and integrate manufacturing process equipment with other systems (Kusiak, 2000). But, as the IT infrastructure evolved towards being process-centric which was based on what the organisation does (Liu, 2002), it was believed that the information should be divided in groups of operations activities linked only by the minimum, most basic, and practical necessary information (Toh and Harding 1999, Ho et al. 2000).

Therefore, the SACAPP, focussed at the high-level on Sales-CAD-CAPP-company resources integration and the generation of the sequence of operations, while at machine-tool level focused on improving human-automata integration, and doing so, was in accord with actual business practices (Song and Nagi, 1997), and supported the performance of the vital business processes (Toh and Harding, 1999).

3.6 SACAPP and CIM

CIM, with its hierarchical control, was considered too inflexible for server-based computing that deployed small business applications for small production batches in a dynamic changing environment (Davies 1997, Volchkov 2002, Ryu and Jung 2003), therefore not suitable for discrete manufacturing environments characterised by a low level of automation, that lead to the conclusion that fully computerised integrated manufacturing system was unlikely to be the main model in the near future (Sun, 2000).

Therefore, the SACAPP considered more appropriate to: handle the knowledge, not just the information or data, through computer techniques which could yield a better CIM (Prasad, 2000) - for example machine-tool gauges that collected and shared the operator's practice knowledge; transform the CAD from a design tool to a data communication tool (Prasad, 2000) - for example SADwO system; and move from a "technology push" situation to a "requirement pull situation", which represented the move from CIM to Intelligent Manufacturing Systems (IMS) (Iung et al., 2001) - for example the way SACAPP collected its data.

3.7 SACAPP and Automation

Although automation reduced the amount of manual and clerical effort in product development, it was not the right answer for: the discrete production of customised products with short life cycle (Groover, 2001); CNC (Computer Numerical Control) machining of non-repetitive orders (Figure 3.20); and the non-linear process plans that had to provide alternative sequences or processes to improve scheduling and shop floor flexibility (Detand 1993, Usher, 2003), but which enormously increased CAPP complexity (Joo et al., 2001) and declined system performance, hence being less important as previously thought (Usher, 2003).

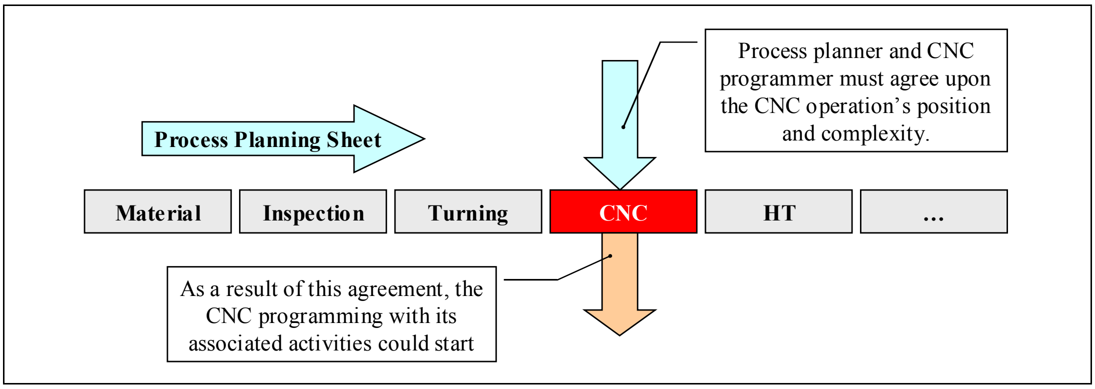


Figure 3.20 Process planning and the CNC operation which, in the first production run was compatible to conventional machining because programming and testing were required

Therefore, before involving automation, SACAPP applied a number of automation principles and strategies (Figure 3.21), and considered the fact that the manual methods offered accuracy and control but lack the efficiency and scalability of automated methods, and the automated methods offered efficiency and scalability but lack the accuracy and control of manual methods.

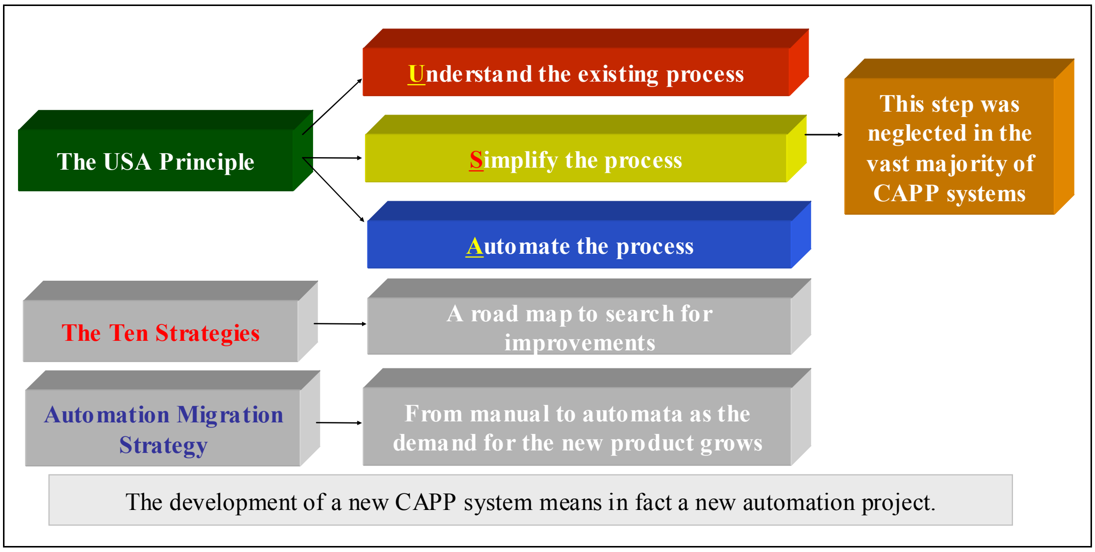


Figure 3.21 Automation principles and strategies (Groover, 2001 - adapted)

Consequently, SACAPP considered that a few good material alternatives would increase the drawing and production flexibility, and because the co-operation was the philosophy behind the agile manufacturing strategy (Song and Nagi, 1997), considered Web applications a powerful communication tool (Ho et al. 2000, Peng 2002) that could expand the traditional plant information systems, eliminate barriers to integration (Cheng et al., 2001, Huang and Mak, 1999) and create new procedures for integration (Chui and Wright 1999, Chung and Peng, 2004) (Figure 3.22).

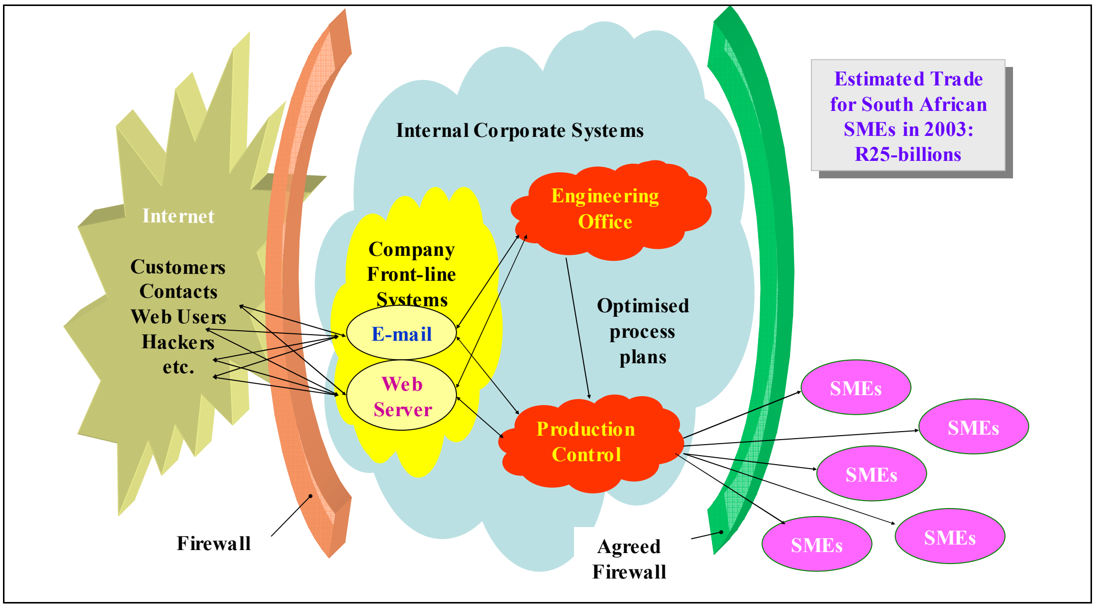


Figure 3.22 Web and the extended enterprise, where the Web-centred technology was considered the key for the SMEs to enter into a meaningful business-to-business electronic commerce, and by doing so, concentrate on their core competencies, increase their flexibility (Peng 2002, Koutsogiannakis and Chang 2002), and being in line with the new distributed and more decentralized process planning systems trend (ElMaraghy 1993, OMG 1996, Joo et al. 2001) that could become capable of fulfilling the requirements of a pull concept based manufacturing system (Shukla and Chen, 1997)

3.8 SACAPP and the Complexity

Manufacturing was considered a complex activity (Morley, 1998) where people practiced complex professional activities (Budd, 2001) using sometimes complex systems (Szyperski et al., 2002) such as the actual CAPP systems (Kryssanov et al. 1998, Ciurana et al. 2003). In these conditions, although sometimes was no solution to the problem as formally stated (Freuder and Wallace, 2000), people pursued system's features and performance that stretched the limits of human understanding and led to system's complexity (Sha 2001, Szyperski et al. 2002).

Therefore, to avoid, mitigate, or monitor the risks associated with complexity, SACAPP's philosophy was to keep it simple (Sha, 2001), use a few simple tools (Budd, 2001), and consider the ways in which the nature, following a few simple rules, enabled individual members to act together in a robust, deterministic, and adaptable way that demonstrated collective intelligence probably more than the sum of its parts (Morley, 1998).

Consequently, in this thesis, CAPP complex problem was decomposed into smaller more manageable sub-problems (Kochikar and Narendran 1998) and, by separating critical requirements from desirable properties (Sha, 2001), emphasizing particular aspects, and relaxing constrains (Freuder and Wallace, 2000), solutions have been found that replaced global optimality in favour of the more realistic goals (Kochikar and Narendran 1998). In addition, by doing so, it was possible to create the conditions for assistance capable of learning, adapting, optimising, and reconfiguring at various levels of the organisation (Kim 1999, Wang et al., 2002), and enable individual sub-systems to act together and demonstrate their collective intelligence (Morley, 1998).

3.9 SACAPP and CAPP Evolution

The SACAPP considered CAPP in a strategic position, and the glue for intertwined activities (ElMaraghy 1993, Davies 1997, Ming et al. 1999, Yan et al. 2001, Kang et at. 2003, Kumara and Rajotia 2003) (Figure 3.23). Therefore, in addition to CAPP's technical aspects usually researched (ElMaraghy, 1993), SACAPP brought in a new three dimensional functionality space where, important social, organizational, managerial, and business implications factors were considered, and so provided better interaction with and support for the human planner (Maropoulos 1995, Kuric and Janac 1999, Kazman and Bass 2002) (Figure 3.24).

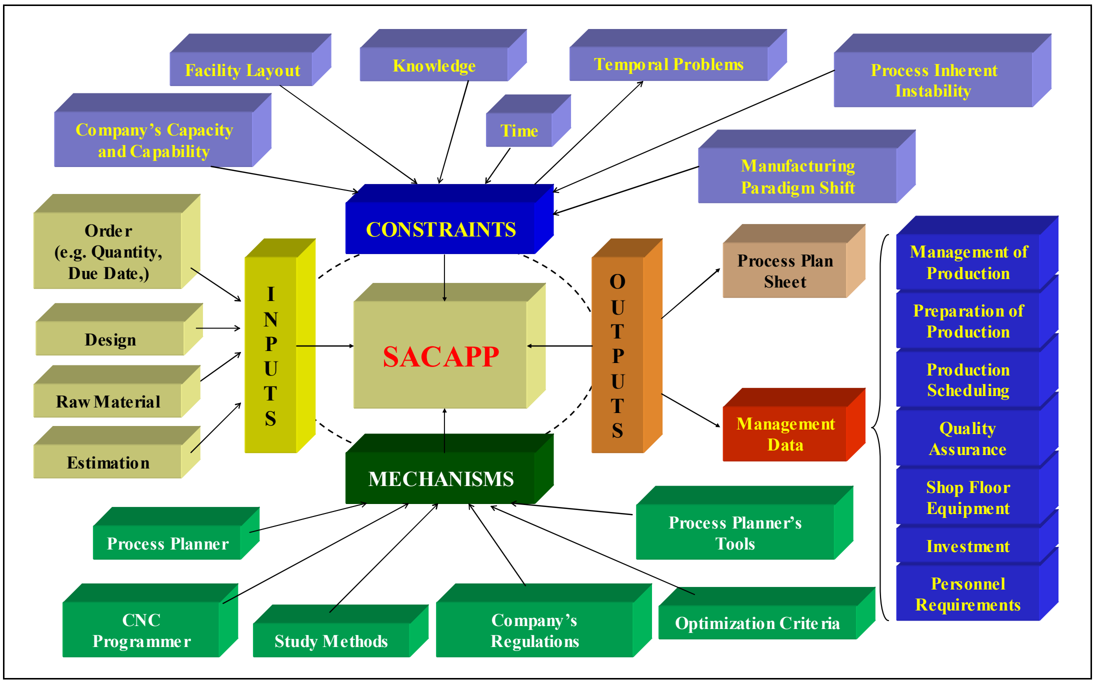


Figure 3.23 SACAPP environment

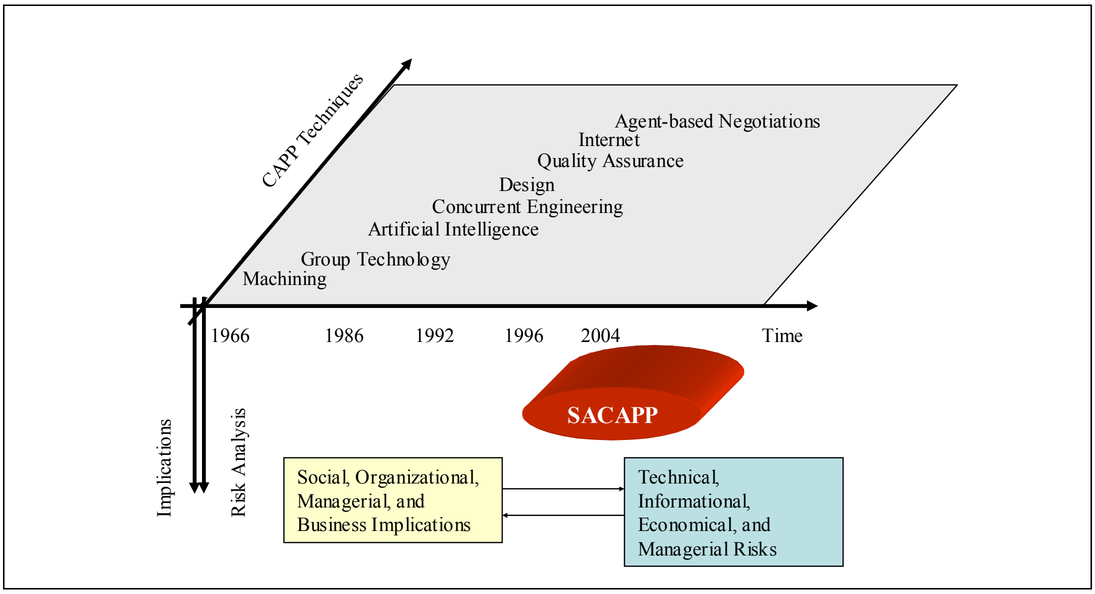


Figure 3.24 SACAPP new dimensional analysis

3.10 SACAPP Architecture Baseline

In the inception phase, the architectural analysis was carried out to the point necessary to establish that a feasible architecture existed. In the elaboration phase, with its Lifecycle Architecture milestone (Jacobson et al. 1999, Reed 2002, Simpson 2002), architectural decisions have been made to extend the preliminary architecture to the architecture baseline which represented the executable early version of the system (Figures 3.25 and 3.26).

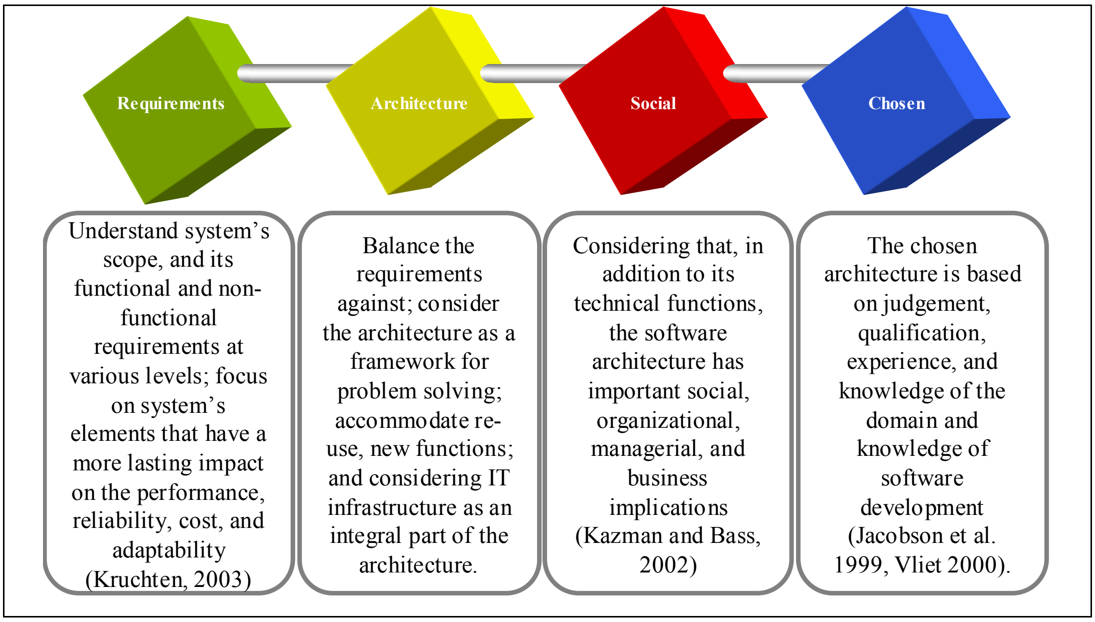


Figure 3.25 Steps in deciding the preliminary baseline architecture which included the fact that the architecture grows out of the needs of the enterprise (van Vliet, 2000)

Furthermore, because the interfaces between the interconnected systems were precisely defined, each system was possible to be treated in its own right. In addition, in order to achieve "plug-able" systems, networks of maps provided the communication between systems, where the GUI data in one or more systems, saved as data file(s), was used to create the GUI map in another system (Figure 3.27), and finally, the physical system architecture in terms of connected nodes was defined (Figure 3.28).

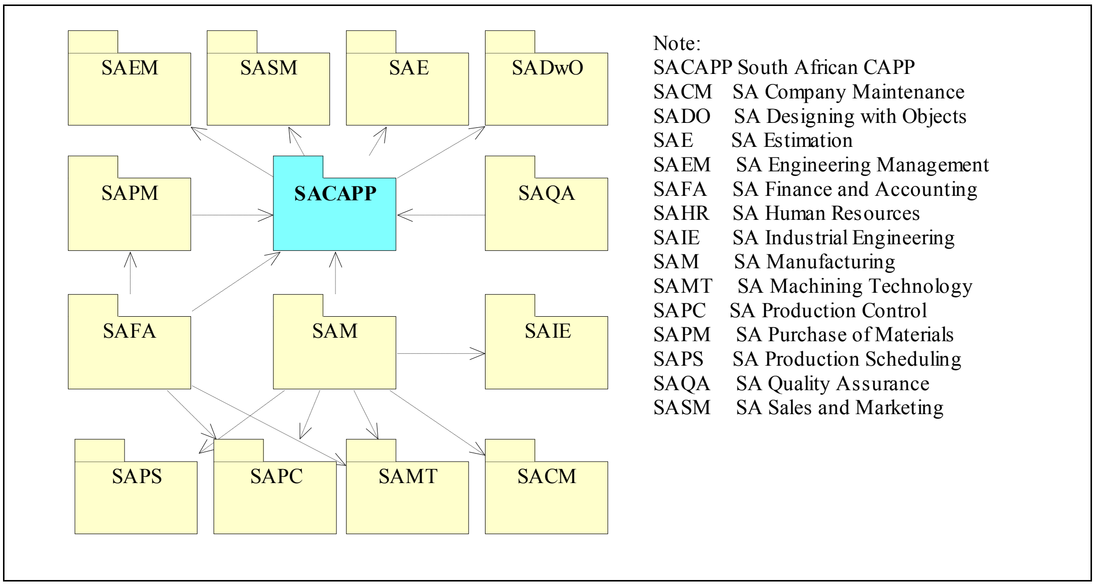


Figure 3.26 A layered and collaborative system of systems, with SACAPP boundary's system dependent of other systems, and so, in line with the normal business practices (Appendix F, Business Modeling)

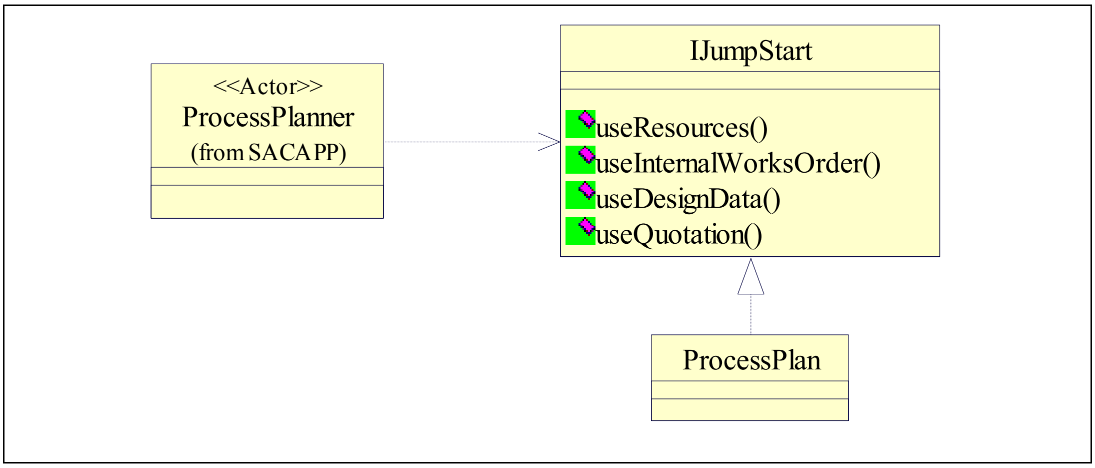


Figure 3.27 The UML SACAPP class diagram interface, where maps created in other systems decides the SACAPP map. The ProcessPlanner class has a dependency association on the interface IJumpStart so: the actor depends on the "useResources" provided by the company's management, the "useInternalWorksOrder", the "useDesignData", and the "useQuotation". Then, the "ProcessPlan" class realises or provides the concrete implementation for the interface "IJumpStart".

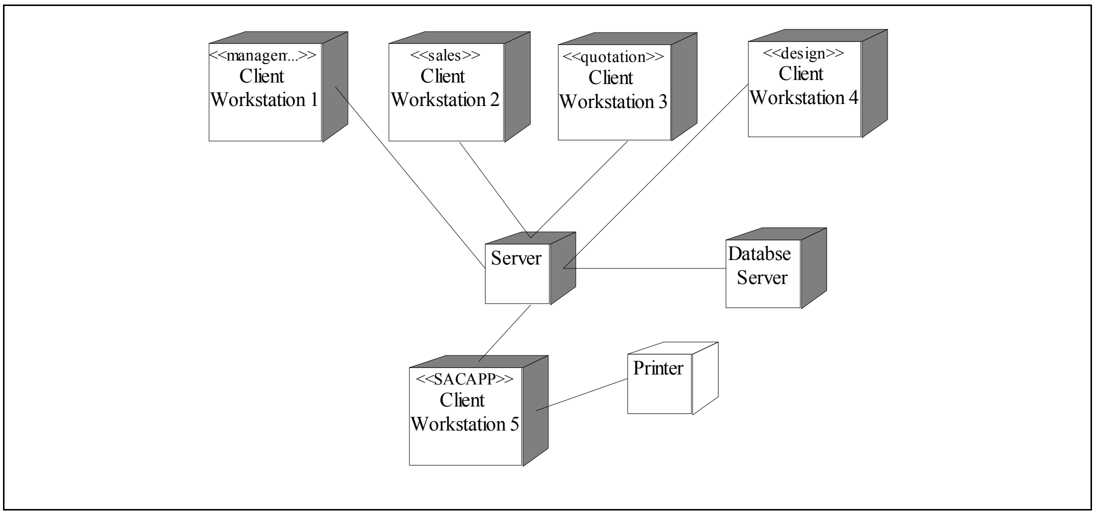


Figure 3.28 SACAPP Deployment diagram

3.11 Conclusions

Chapter 3 focused to find, apply, and clearly support through authorities, evidence, or logic, the methodologies intended to answer to the hypothesis and research questions formulated in Chapter 2. Also, Chapter 3 was the starting point of the SACAPP software project that was spanned by an iterative and risk-driven process that provided the assurance that appropriate procedures were followed, the focus was on the scope of the project, and the risks were minimised. As a result, the Lifecycle Objectives milestone of Inception phase showed that it was both possible and desirable to develop the SACAPP software system, and the Lifecycle Architecture milestone of the Elaboration phase showed that an executable, resilient, and robust architecture baseline which could evolve over the life of the product was created. Consequently, the next chapter will focus on the Construction phase and its Initial Operational Capability milestone.