D CAPP SYSTEMS EXAMPLES

In this appendix, a number of CAPP systems as found in the review of the literature are briefly introduced. The alphabetical presentation of systems mainly according to the author's name, and secondary by the system's name, provides details about developer, date, system approach, part shape, programming language, and some system characteristics. For some systems complete information was not available. Furthermore, it should be emphasized that the list below cannot be considered as an exhaustive list.

CAM-I CAPP: Developed by McDonnell Douglas Automation Company in 1976, CAM-I was a variant approach system based on group technology.

Chang and Chang, 2000: The Integrated Artificial Intelligent system for dynamic CAPP (IAI-CAPP) simulated the experienced human planners who recognize a design specification by important features (Figure D.1). Written using Visual Basic, it integrated both variant and generative CAPP approaches. The system integrated an adaptive-learning recognition mechanism consisting of fuzzy logical back-propagation neural networks (BPNs) for evaluation of manufacturability of important features characterized by seven indices namely tool-shape, intricacy, symmetry, quantitative estimate of the accessibility, number of feasible tool approaching directions, dimensions, and technical requirements. These indices were also used to reduce the solution searching space of process planning.

Chung and Peng, 2004: Computer-aided tool selection (CATS) was considered a vital element in CAPP systems. The Web-based tools and the machine selection system (WTMSS) were implemented for machining rotational parts as an affordable solution for small companies with limited capital investment, and simplify the sharing of production knowledge through the Internet. The system required the sequence of operations from process planning and part information from design. WTMSS mainly consisted of object-oriented databases (OODB), java applets, and a virtual reality modeling language browser (Figure D.2).

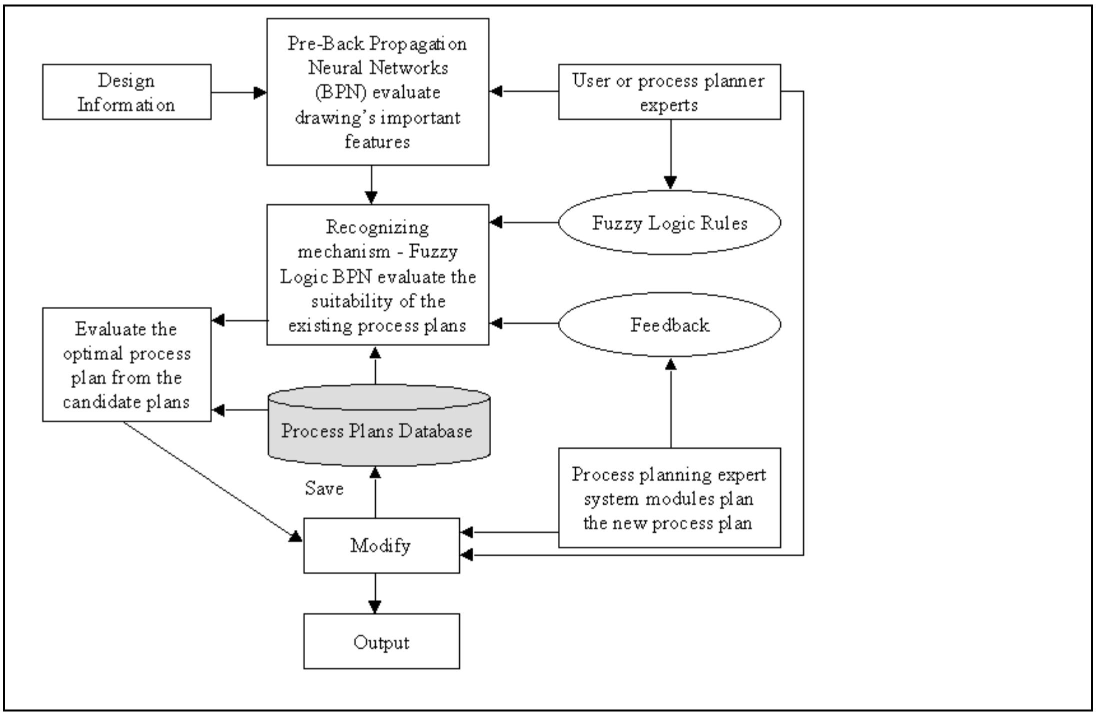

Figure D.1 IAI-CAPP architecture (Chang and Chang, 2000)

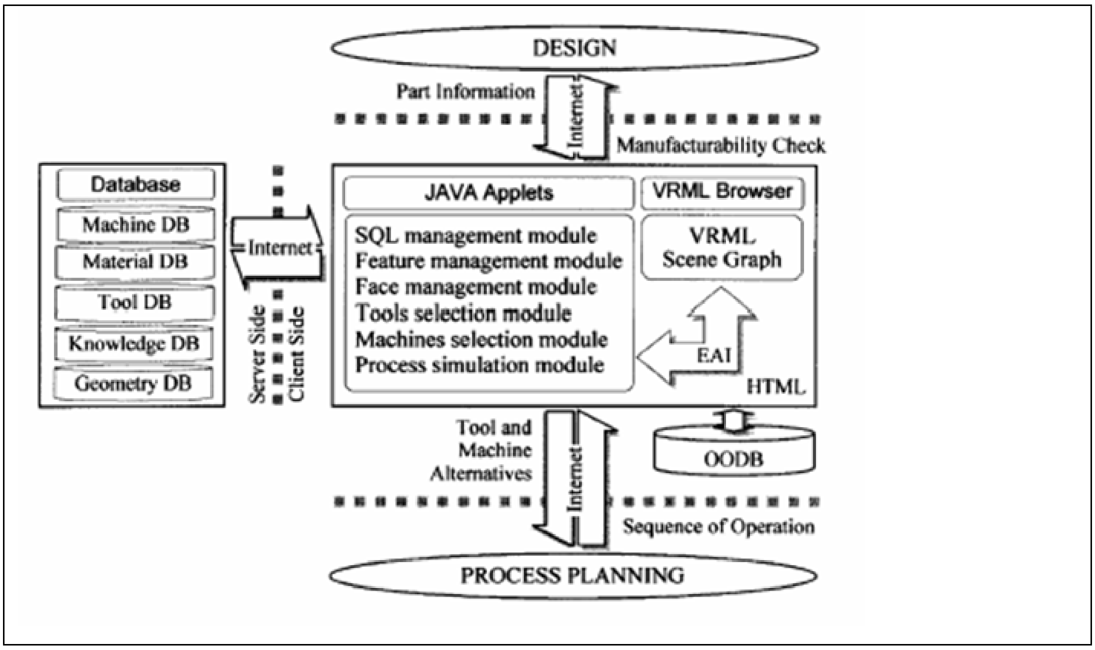

Figure D.2 WTMSS architecture (Chung and Peng, 2004)

COMPLAN, 1995: It was a generative and variant CAPP developed in C++ that provided either manual or automatic planning functions for process planning and scheduling, so the user retained at all times full control over the system output.

Feng and Zhang, 1998: The modular architecture for rapid development of CAPP systems considered technology, management, and socio-cultural manufacturing enterprise factors (Figure D.3). The information infrastructure combined the Internet as software delivery vehicle with the Standard for the Exchange of Product Data (STEP), and the Common Object Request Broker Architecture (CORBA) standard for interoperation of software components.

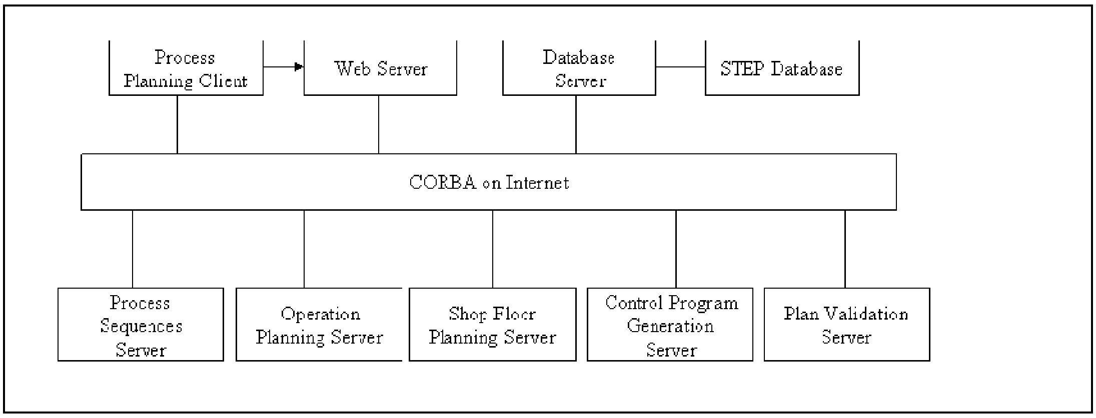

Figure D.3 Distributed CAPP system (Feng and Zhang, 1998)

Feng and Song, 2000a: They suggested the diagrams in Figure D.4 and Figure D.5 for design and process planning integration.

Feng and Song, 2000b: The functional design that generates a product's major functions was decomposed into detailed functions, thereafter mapping detailed functions into behaviour models (Figure D.6).

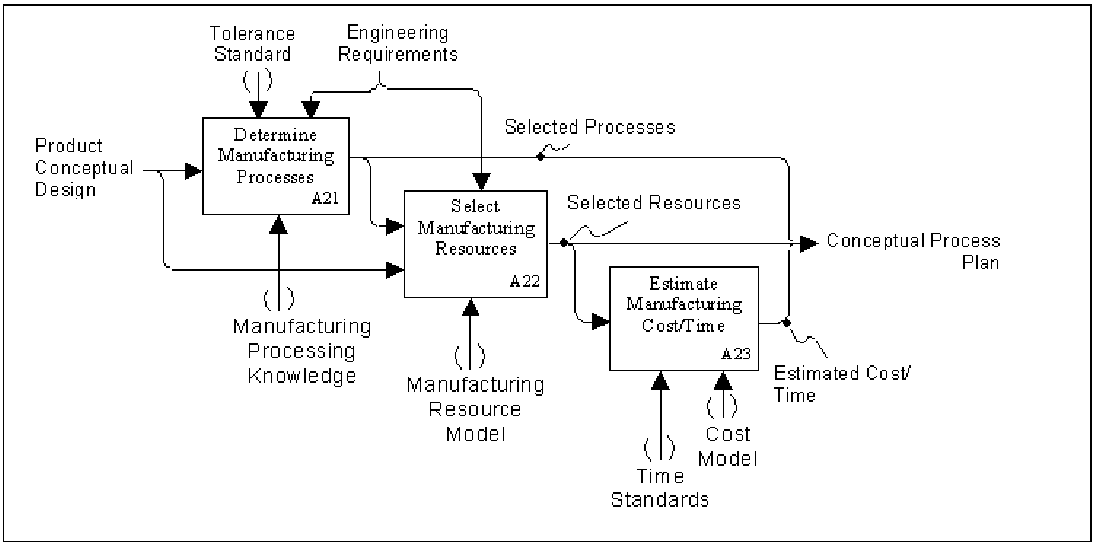

Figure D.4 Functional decomposition of conceptual process planning

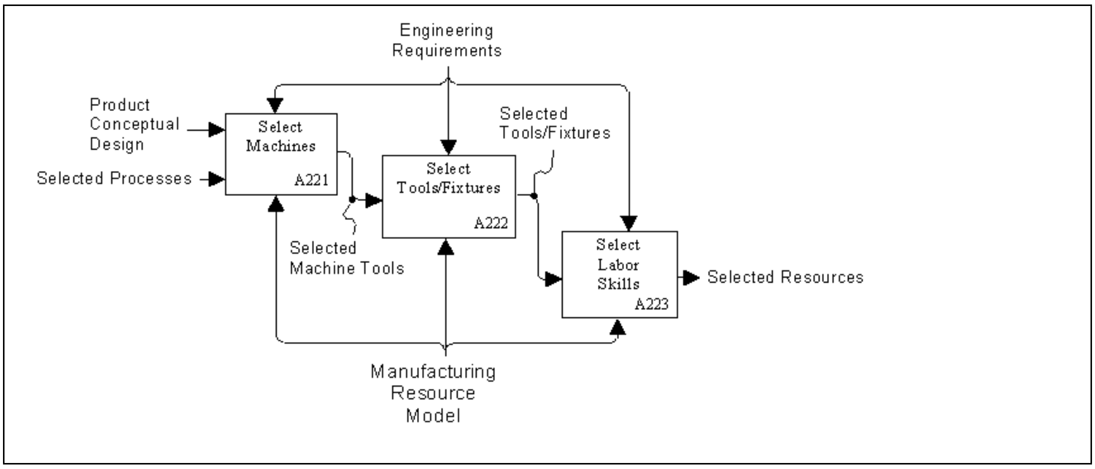

Figure D.5 Manufacturing resource selection

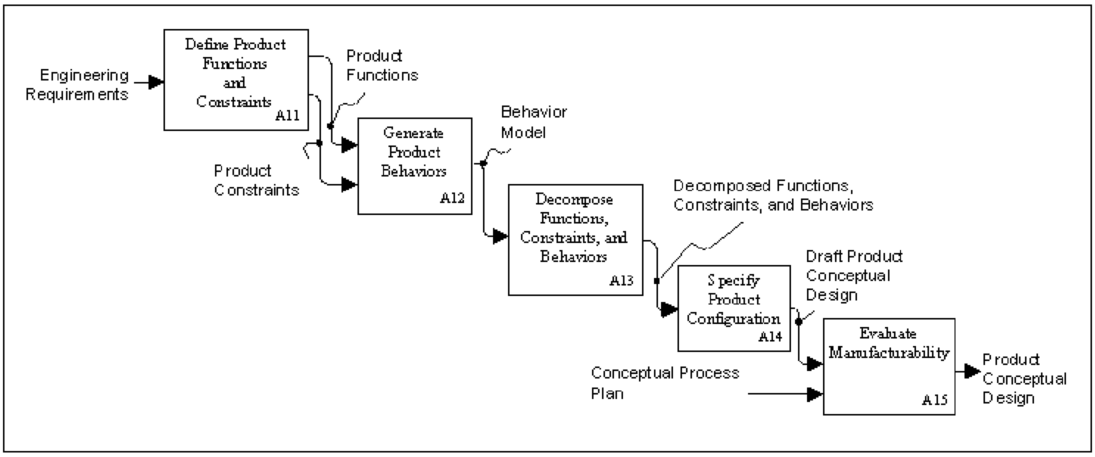

Figure D.6 Functional decomposition of conceptual design

GARI: Developed in 1981 at the Grenoble University, France, GARI was the first AI based generative CAPP system to be reported in the literature (Hugh 1994, Kiritis 1995). The knowledge was represented by production rules dealing with technological and economical preferences weighted according to their importance. The manufacturing rules were of the type "IF-THEN", with the input as symbolic representation of the part, features, and relations between them in the form of LISP rules.

Gonzalez and Rosado, 2003/2004: GF-CAPP was designed to give the CAPP system autonomy from the manufacturing environment (Figure D.7), so, its internal database contained specific information in machining domain, with the external database considering data for production resources. The model also proposed a low-level information extraction geared to a structuring of knowledge in CAPP.

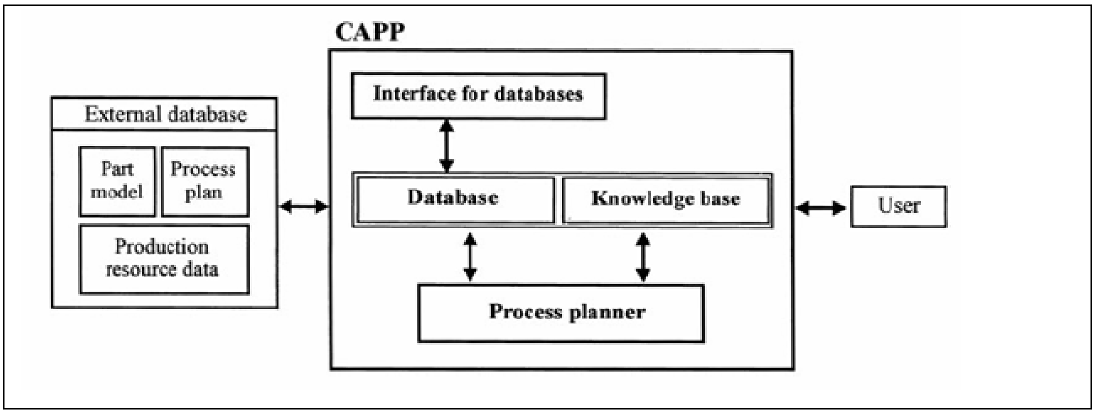

Figure D.7 GR-CAPP architecture

Jang et al., 2003: The process plans were generated through two steps, namely elaboration and validation (Figure D.8).

Joo et al., 2001: The adaptive and dynamic process plans based on shop floor status were constructed as NNs embedded into each process feature. When executed by the shop floor controller, the machine, cutting tools, cutting parameters, tool paths, and NC codes were determined before the associated process plan feature were machined (Figure D.9).

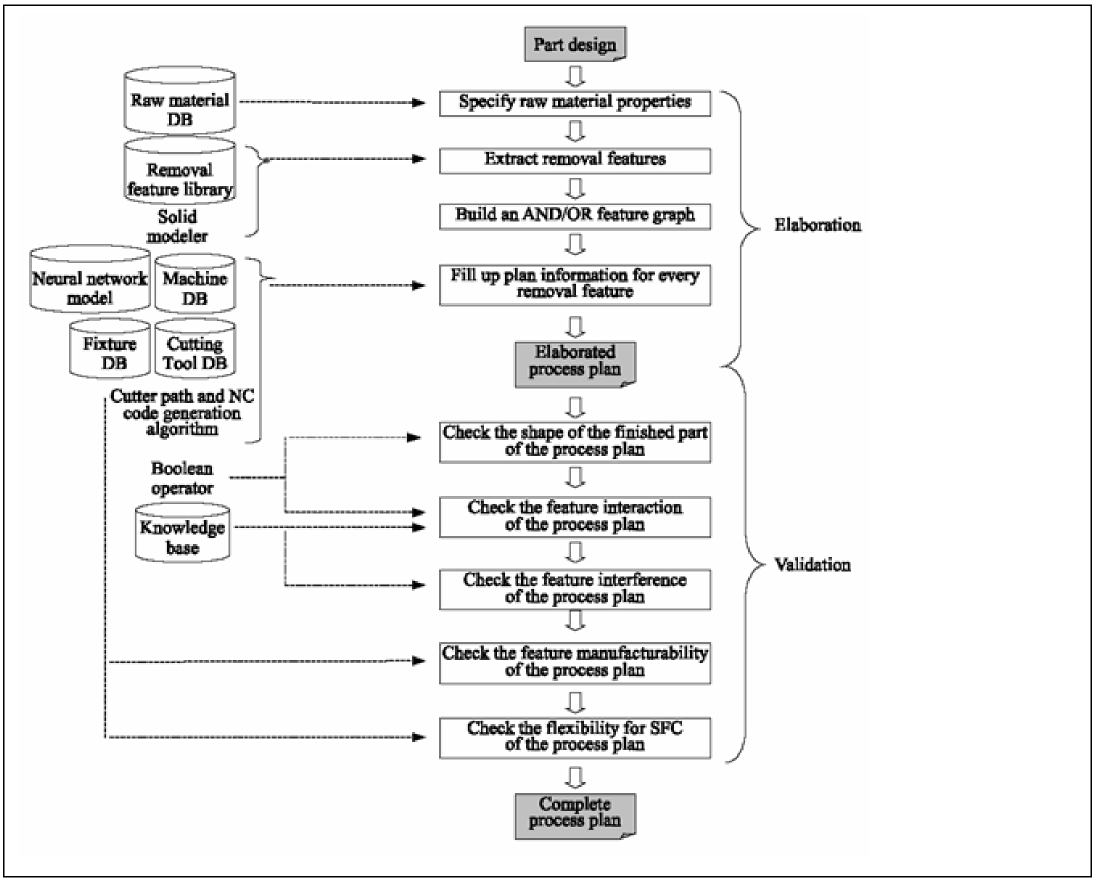

Figure D.8 Framework for the elaboration and validation of a process plan

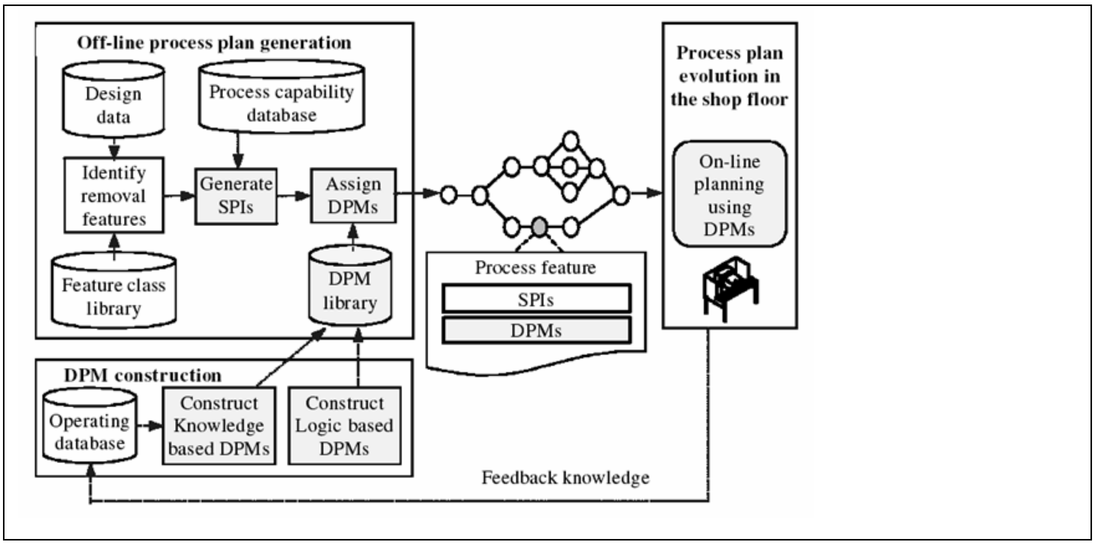

Figure D.9 System architecture for adaptive and dynamic process planning

Sormaz and Khoshnevis, 1997, and Khoshnevis et al., 1999: The 3I-PP (Integrated, Incremental, and Intelligent Process Planning) system for prismatic parts consisted of a number of knowledge base, feature finder (which is then augmented by the user in order to include explicit feature dimensions, tolerances and surface finish, material, batch size, etc), process selection (performed independently for each feature), and process sequencing (implemented using frames) modules. Developed in LISP, the hybrid rule-based and frame-based system viewed the process planning as a set of interrelated objects, which exchange messages. The relations were of the following forms: is-a, may-be-machined-by, may-be–used-for, is-machined-by, do-after, preferred-to, may-use-tool, make-process-time (Figure D.10).

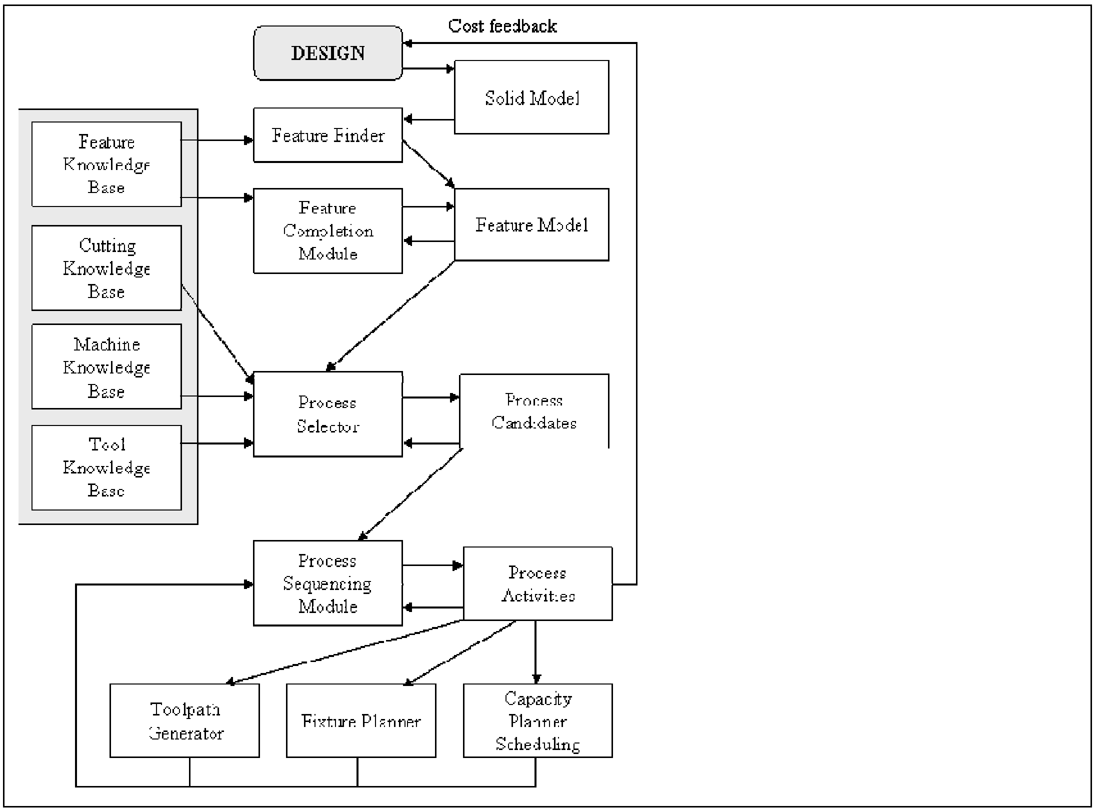

Figure D.10 3I-PP system organization (Khshnevis et al., 1999)

Kryssanov et al., 1998: The CAPPES (CAPP Expert Systems) was focussed on finding multiple knowledge types and planning strategies within one flexible facility (Figure D.11).

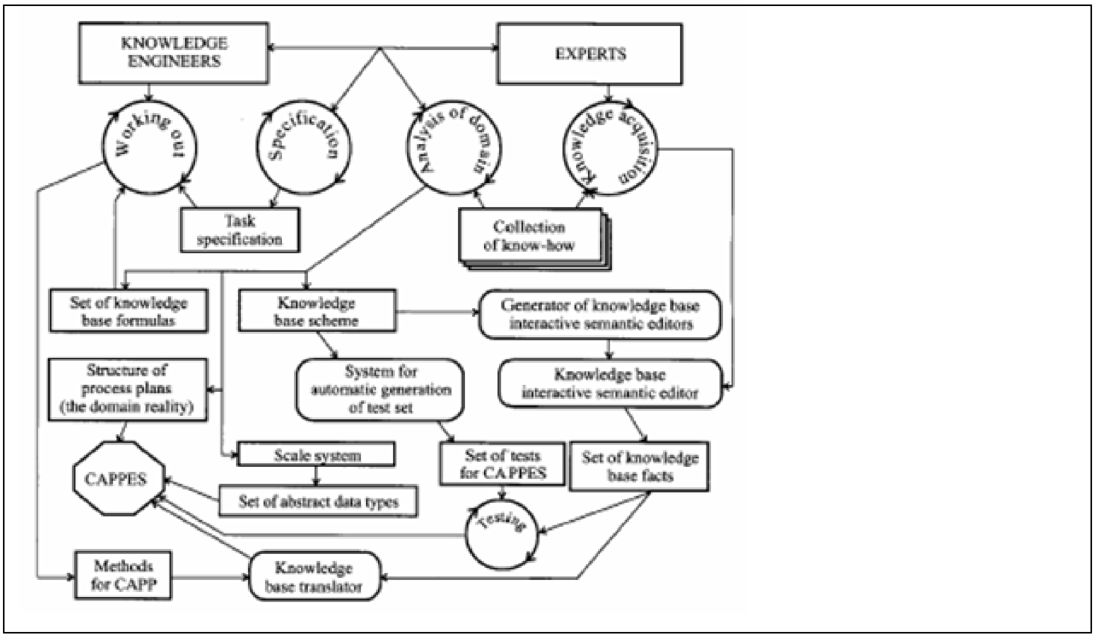

Figure D.11 CAPPES building technology diagram

Law and Tam, 2000: The system used inference procedures that satisfied a set of desired goals under given constraints and resources (Figure D.12).

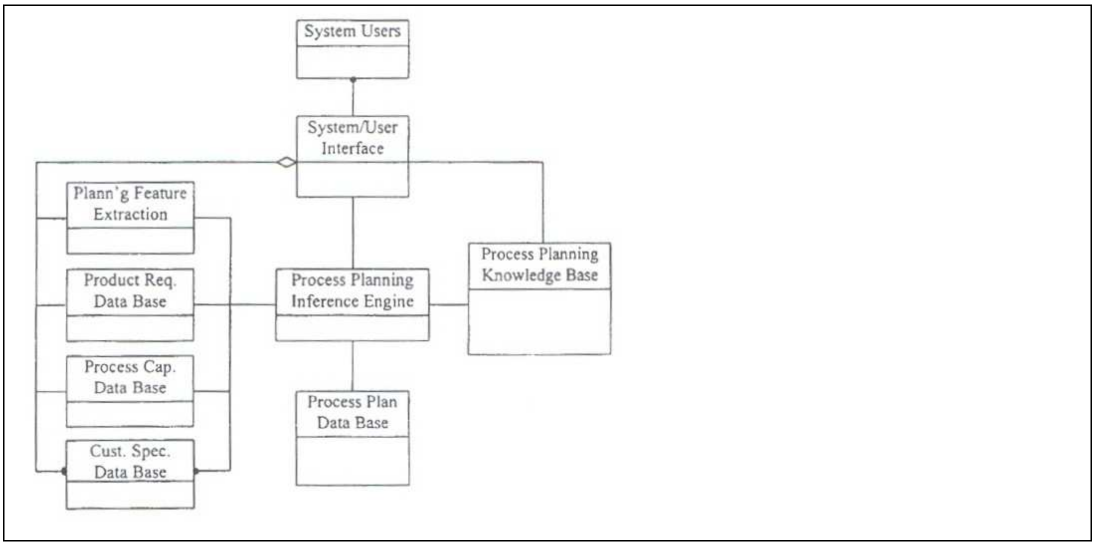

Figure D.12 The highest object model in a CAPP system

Law et al., 2001: Used for double-sided circuit boards, the system intended to find a solution to the problem of structuring planning knowledge, problem constraints, and resource constraints (Figure D.13).

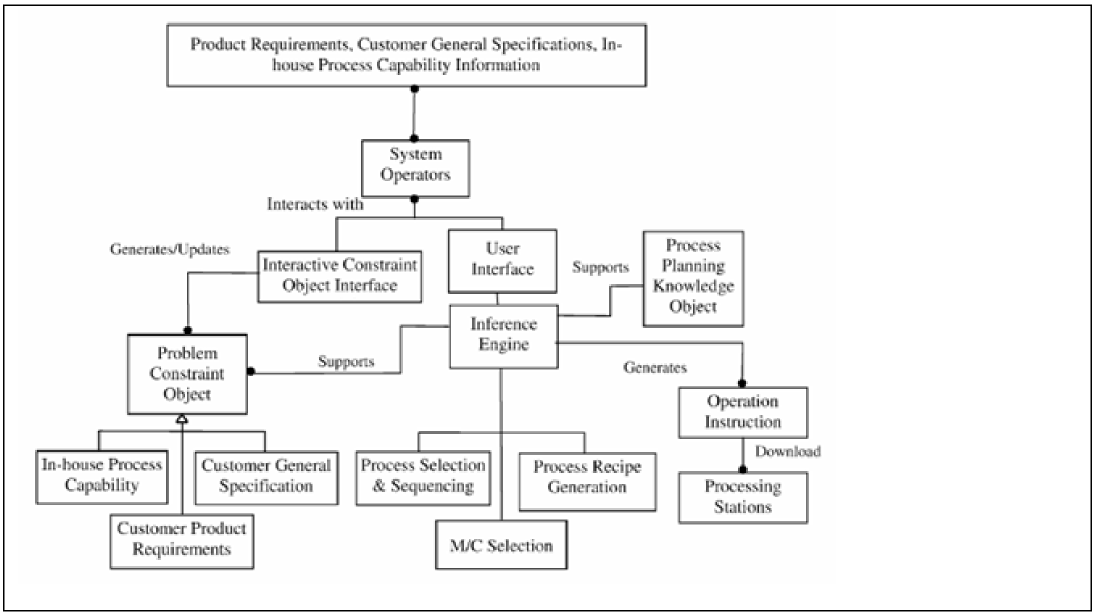

Figure D.13 PCB CAPP system

Love and Barton, 1996: The "Whole Business Simulator" (WBS) was proposed to be built from the elements of CIM system and specialised simulation modules (Figure D.14). Also, the approach considered CAPP an industrial fact.

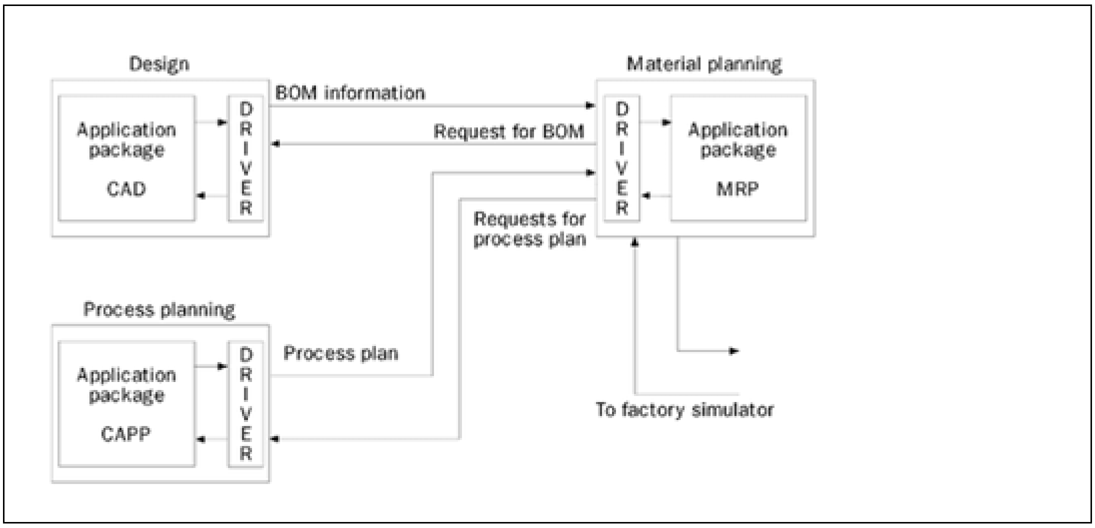

Figure D.14 Encapsulation of application software packages

Li et al., 2004: The generative CAPP system was used to simulate a dynamic workshop case through changing the strategy of cost evaluation and breakdown of machines or tools. Also, it considered that, in order to achieve a global optimal solution, the machines selection, their set-up, and the operations sequencing should be simultaneously determined (Figure D.15).

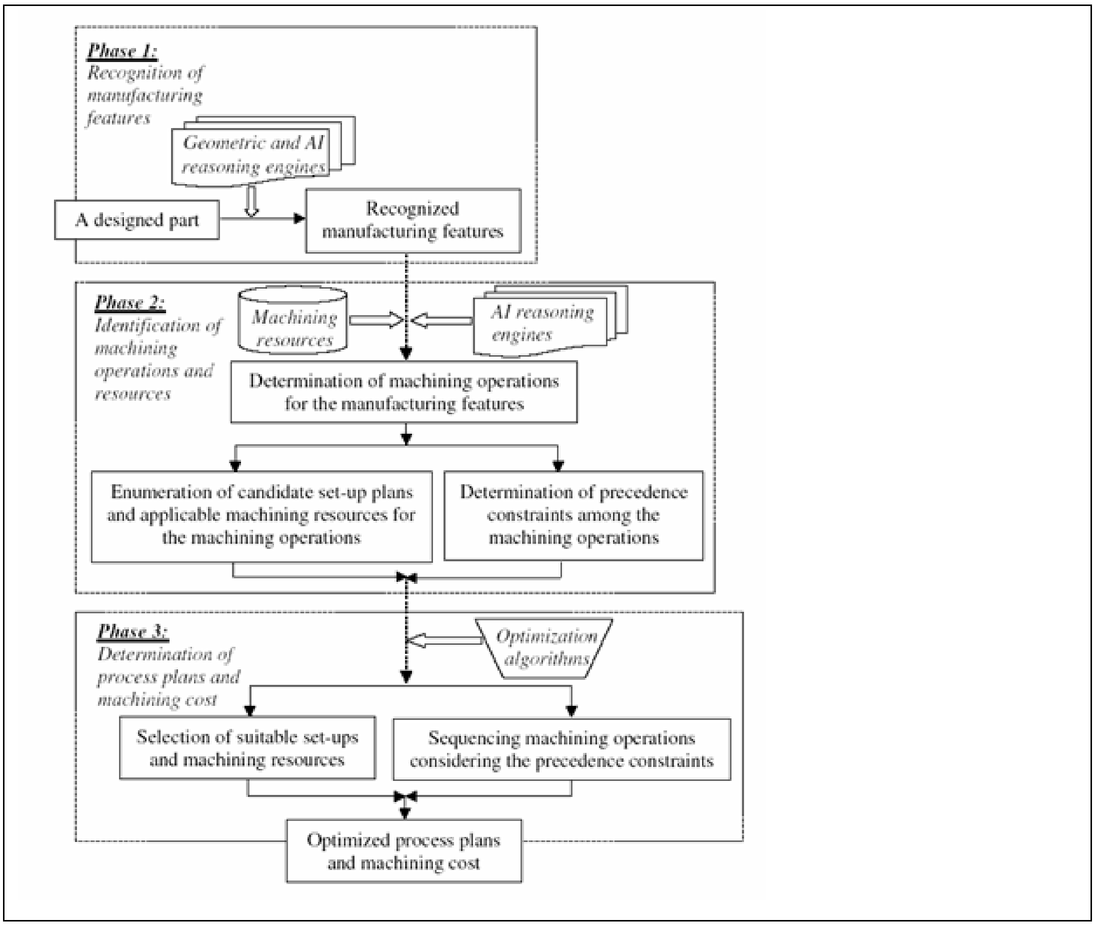

Figure D.15 Workflow of the generative CAPP system

Luo et al., 1997: The Generic Model of Process Planning (GMPP) contained the product information, manufacturing capability, process planning, and synthetic constraints models.

Maropoulos, 1995: The CAPP architecture used three main planning levels, namely: aggregate process planning (that assessed the required technology, costs, development time, quality, make-or-buy decisions, and the targeted costs for suppliers); process planning management (that considered their effect on shop-floor capacity and scheduling), and detailed process planning (Figure D.16).

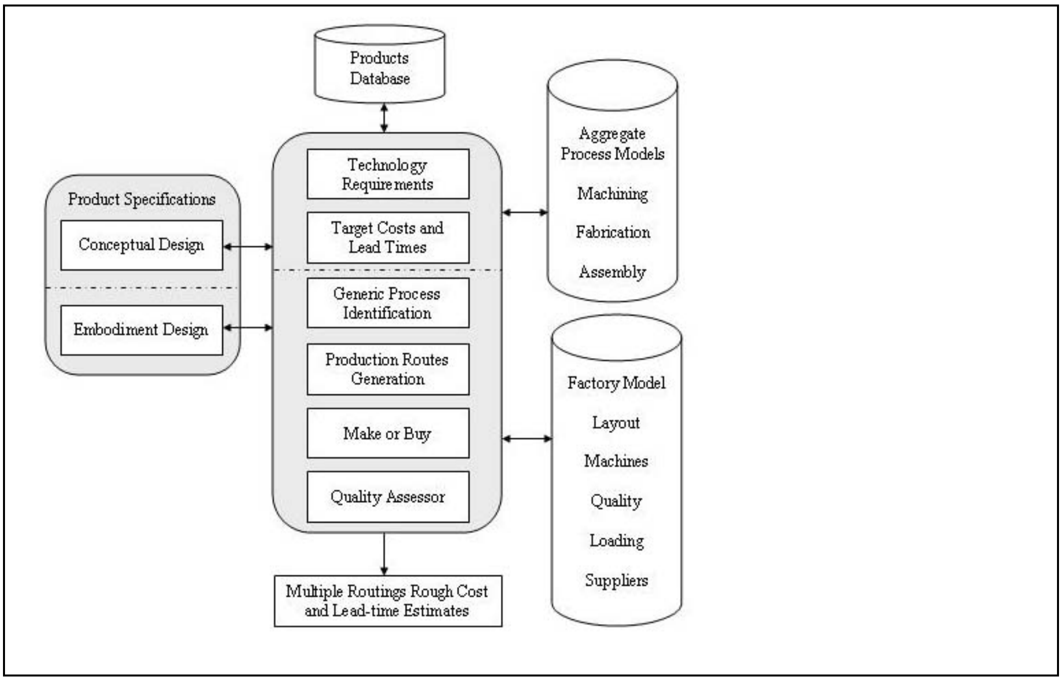

Figure D.16 AMD overall architecture (Maropoulos, 1995)

Marri et al., 1998: The basic CAPP model was considered consisting of a data preparation module (used for obtaining proper product definition for the CAPP system), input, output, database, manufacturing decision-making rules module, and a past processing module that prepares data for production planning and scheduling (Figure D.17).

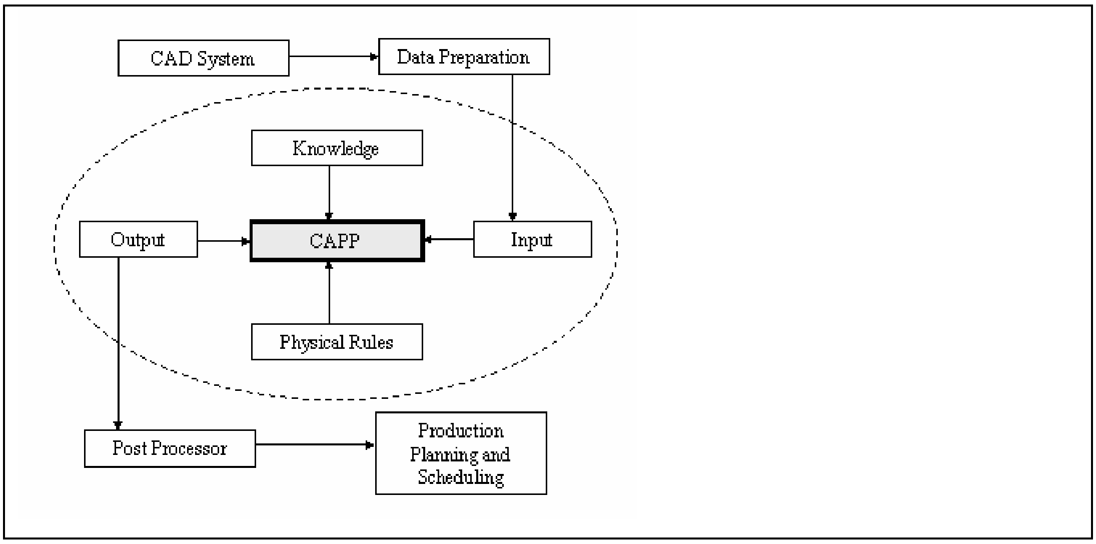

Figure D.17 Basic CAPP model (Marri et al., 1998)

MetCAPP: A knowledge-based process planning and cost estimating system, where the customers could use their own proven manufacturing practices and data.

Ming et al., 1999: The hybrid intelligent inference CAPP system combined the advantages of the expert systems and the neural networks, and stored and managed the explicit knowledge data about the process planning in the form of facts, basic rules, and the implicit knowledge represented in terms of neural network weights (Figure D.18).

Process plan route sheet
Part No.: X6325-2422 Material: 45 steel
Part name: Spline shaft of milling machine
Original: X.G. Ming Date: 02.12.1995 Changes: Date:

Number	Process route	Machine	Setup	Tool	Time
1	Rough-turning	Turn01	Set on out surface	#TURN1	4
2	Semi-finish -turning	Turn04	Set on out surface	#TURN4	6
3	Rough-grinding	Grind01	Set on out surface	#GRIND1	4
4	Semi-finish -grinding	Grind04	Set on out surface	#GRIND4	8
5	Keyway-end -milling	Mill01	Set on out surface	#MILL1	5
6	Keyway-circle -milling	Mill02	Set on out surface	#MILL2	3
7	Spline-milling	Mill05	Set on out surface	#MILL5	12
8	Benchwork				7
9	Inspection				3

Figure D.18 Process plan example

Pande and Desai, 1995: The EXTURN expert system (Figure D.19) was composed of an interactive graphical feature modeller and process panning modules for operation extraction, sequencing, tool selection, and process plan generation. The graphical feature modeller was used to capture geometric and other information from design for the part to be machined.

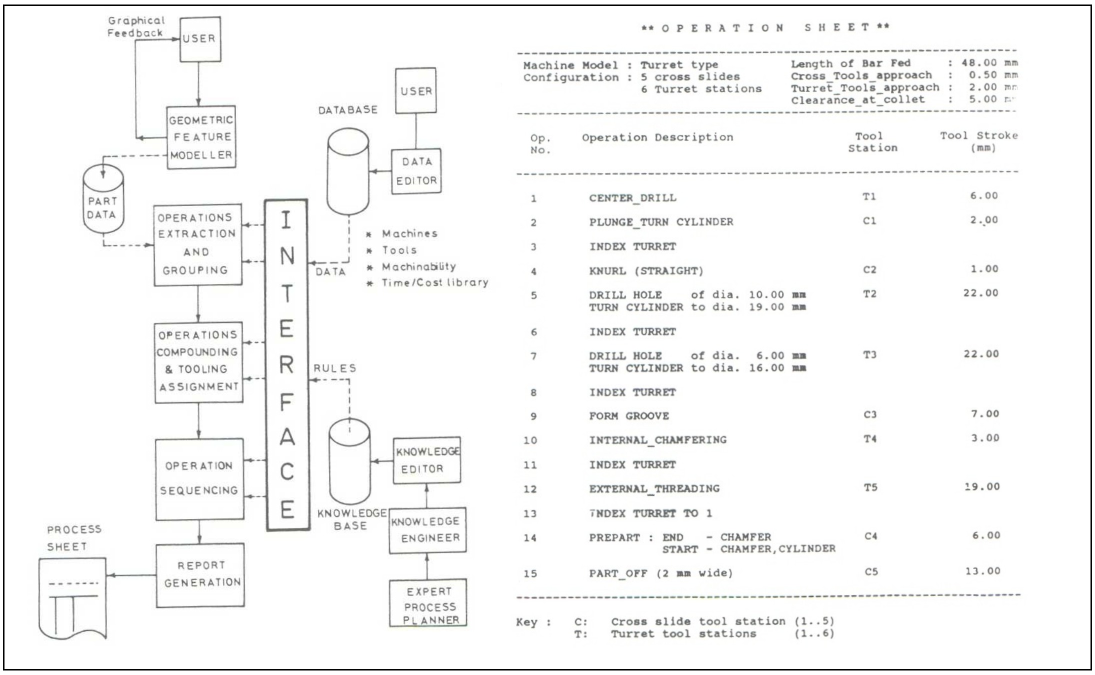

Figure D.19 EXTURN modular structure and its generated process sheet

PART: The generative "Planning of Activities, Resources, and Technology" CAPP systems for 2.5D prismatic components (Figure D.20), was focused on processes like milling, drilling, finishing, and boring. Its functional modules represented sets of related phases, which were independent programs that needed no input from other phases or operator interaction during execution.

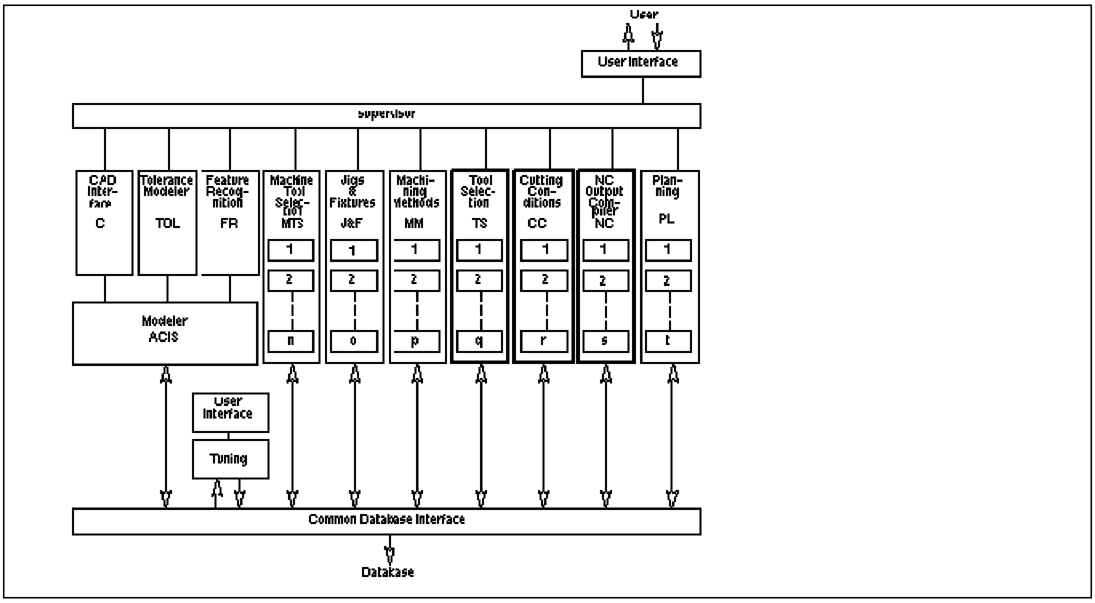

Figure D.20 PART system architecture

The CAD solid model was converted into an internal representation with tolerances not added; then the automatic feature recognition was started. Other activities included: set up selection, machine tool selection, design of jigs and fixtures, cutting tool selection, machining operation sequencing, NC output generation and capacity planning.

PART-S: "Planning of Activities Resources and Technology-Sheet metal" generative CAPP system was inspired by its PART predecessor (Figure D.21). The main processes included in PART-S were laser cutting, nibbling, punching, laser welding and air bending, with their modules subdivided in groups of related phases.

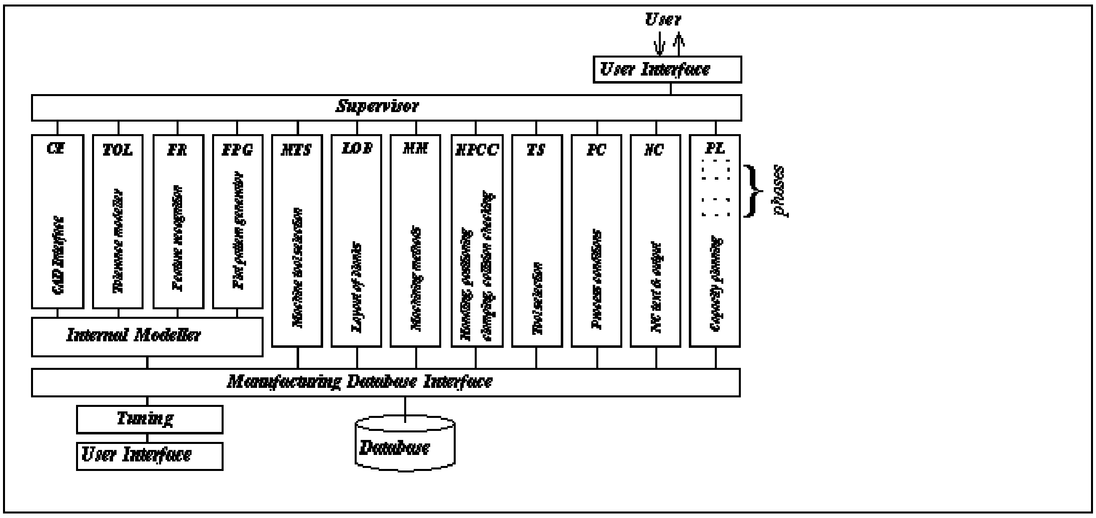

Figure D.21 PART-S system architecture

Pham and Ji, 1999: Geometrical and technological DAD data was extracted to generate manufacturing information. During the design stage, the designer could access manufacturing information to evaluate the design for manufacturability (Figure D.22).

Rozenfeld and Kerry, 1999: The core system for parametric parts consisted of a resource management system and a process plan editor which worked with the user's predefined standards. The operations, and sub-operations, were seen as single resource classes with their attributes stored in the attribute dictionary.

Tables and formulas were stored in databases and associated with object classes, their attributes, and the defined constraints (Figure D.23).

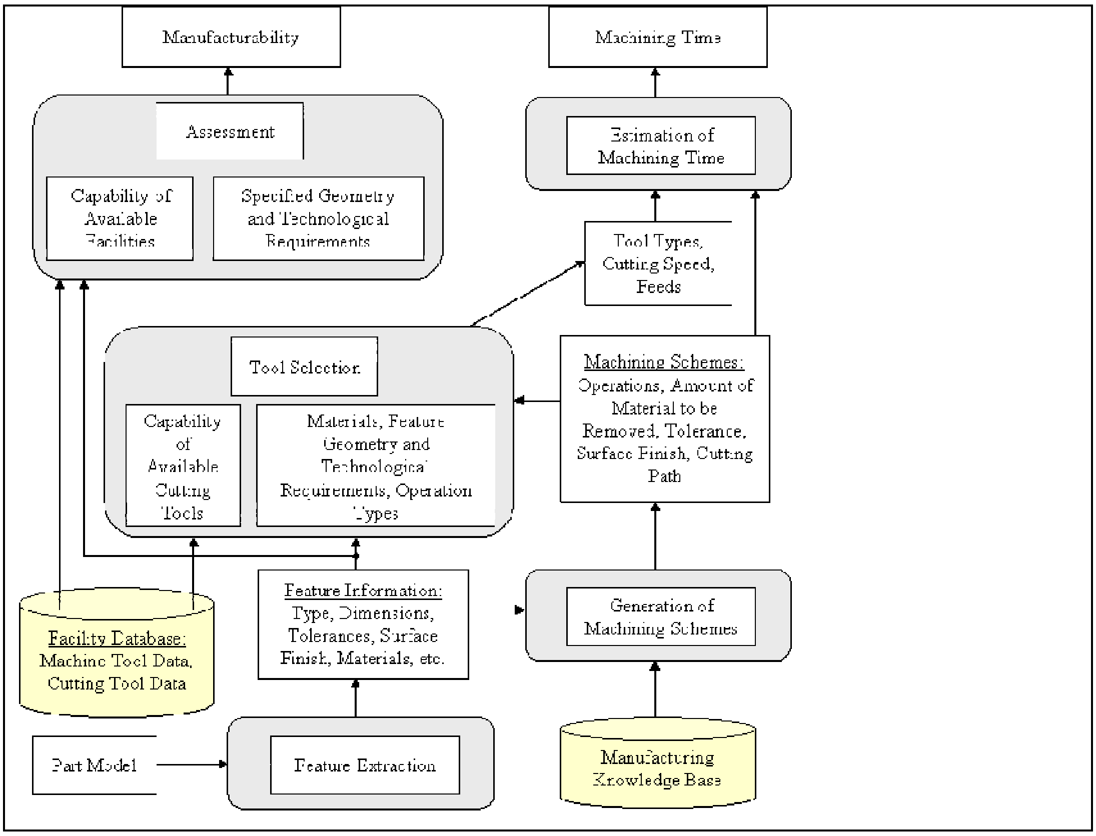

Figure D.22 System architecture (Pham and Ji, 1999)

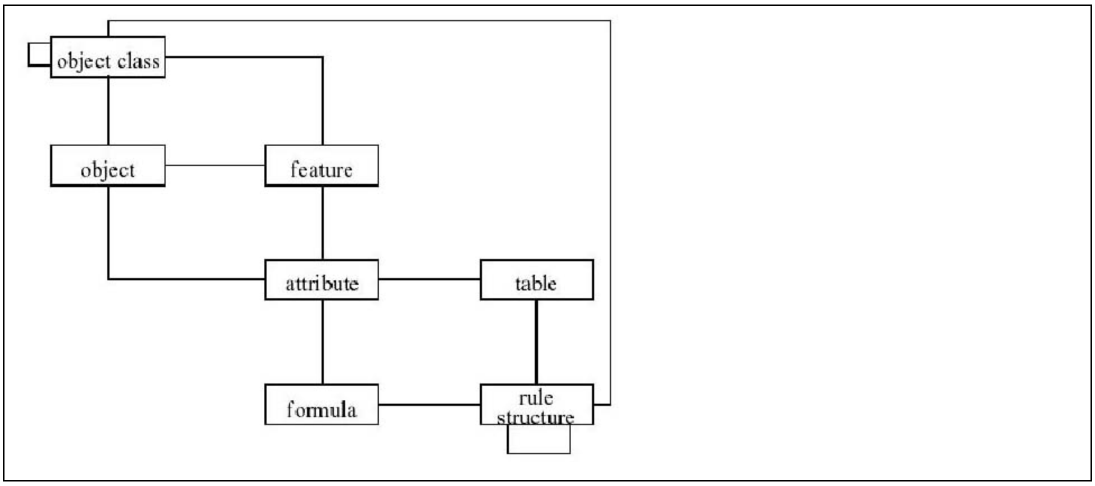

Figure D.23 CAPP environment (Rozenfeld and Kerry, 1999)

Sheu, 1998: A GT code system was used to classify the part geometry together with a weighting method from 1 to 100 to arrange the operation sequencing from the last to the first priority. Planning rules were modified by modifying their weights and the weight of the processes were varied from material to material and from product to product. If a zero weight was given, it meant never generate this process automatically. Also, there were five digits in the primary GT code where the first digit represents the ratio of the product length to the maximum diameter.

SIPP: The Semi Intelligent Process Planner (SIPP) used a frame-based knowledge representation, and a knowledge representation technique called hierarchical knowledge clustering where the knowledge was divided in static knowledge (internally stored by objects) and problem-solving knowledge.

STEP: The STandard for the Exchange of Product (STEP) model data ISO 10303 was aimed to define a neutral standard file format that includes all information necessary to describe a product (OMG, 1996) (Figure D.24).

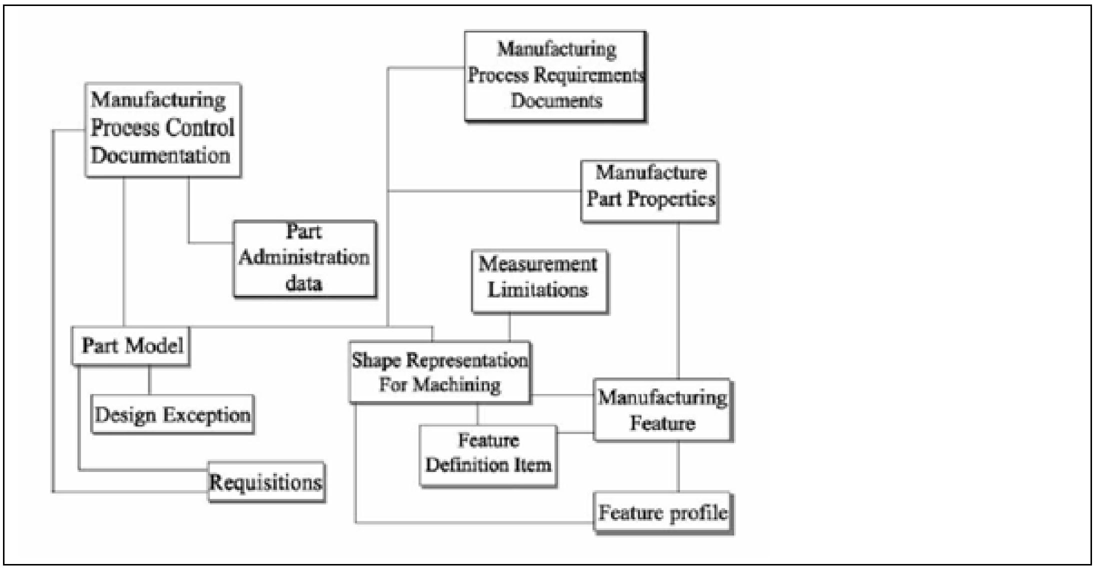

Figure D.24 Basic structure of STEP AP224 schema

For process planning, STEP AP224 was considered to contain all of the information needed to manufacture the required part, but it failed to provide a complete product representation (Yeh and You, 2002) and research dealing with

STEP AP224 was rarely found (Kang et at., 2003) because: it was restricted to a single mechanical part manufactured by a milling or turning process (Kang et at., 2003); tolerances were stored as plain text (Kang et at., 2003); and the relationship between the data and geometric entities was not represented, therefore making the intervention of a process planner still required (Kang et at., 2003).

Wong and Siu, 1995: In ACES (A CAPP Expert System), in order to generate the process sequence, the machined part was viewed as a collection of surfaces mapped, by using transformation algorithms, on a tree of available machining processes (Figure D.25).

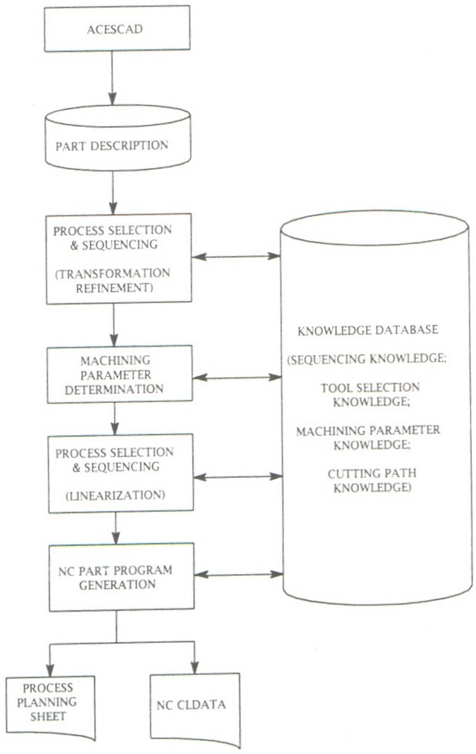

Figure D.25 ACES structure

Wu et al., 2002: Optimal Operation Planning (OOP) used algebraic equations and Petri nets to model the knowledge base for alternative planning operation.

Yan et al., 2001: The Autonomous Machining Process Analyser (AMPA) extracted the milling know-how from Numerical Controlled (NC) programs through reverse engineering, then exported it into a know-how database and finally used to accomplish the automation of operation planning (Figure D.26).

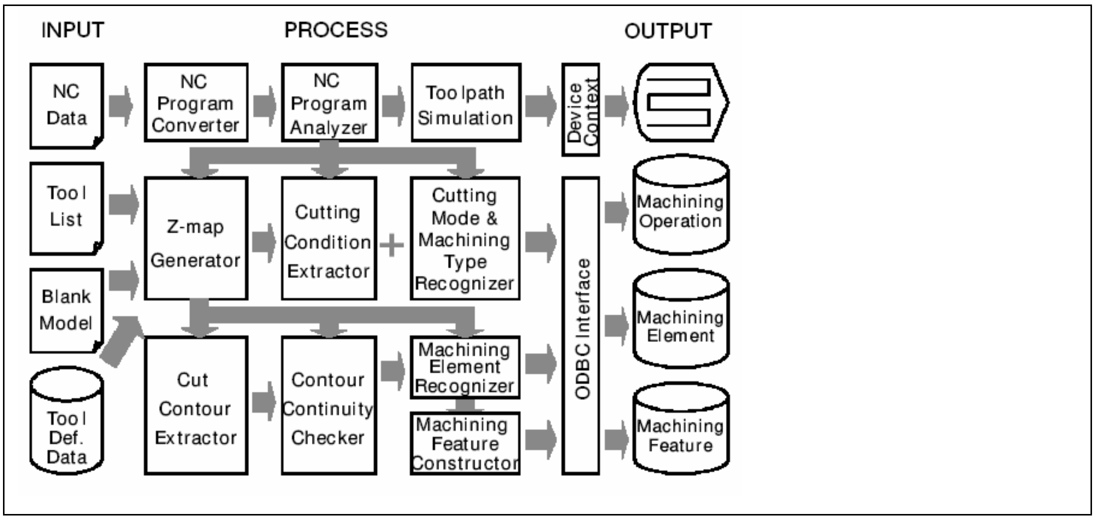

Figure D.26 AMPA architecture (Yan et al., 2001)

Yuen et al., 2003: The CAPP execution was considered a three step process, namely: breaking down the given problem into a set of sub-problems, solve each sub-problem individually, and finally integrate the piecewise solutions to develop the global solution to the complete problem (Figure D.27).

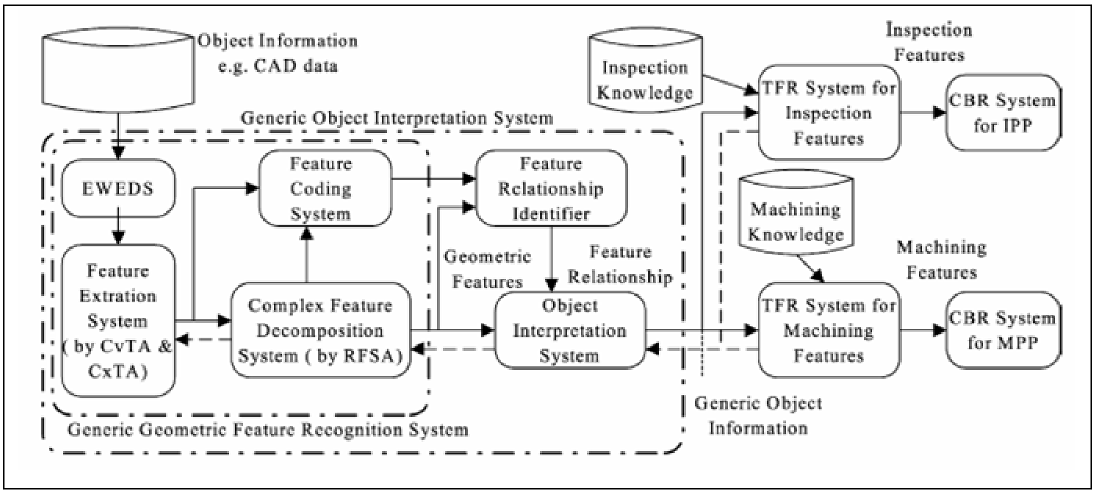

Figure D.27 GCAPPSS structure

Yut and Chang, 1994: The five steps architecture tried to answer questions such as: What are the primary manufacturing objects? What are their functions in planning? How do they interact? Where does manufacturing knowledge reside? How is a control structure implemented among these autonomous objects? Then, a framework was developed to identify top-level classes, the connections between them, and the core messages they used for communicate. The objective in the hierarchical planning was to elaborate an abstract plan by incrementally adding details to a partial plan that had a lower level of abstraction. Also, it was considered that the cutting-tools describe the features and so they own much of the manufacturing knowledge, therefore each tool knows the prerequisite feature it requires for each operation.

Zhang et al., 1999: The Object-Oriented Manufacturing Resources Modelling (OOMRM) was used to describe manufacturing resources capability and capacity. Based on the OOMRM, an Agent-Based Process Planning (ABPP) was used to implement a man-machine process planning platform to support an experienced manufacturing engineer in mapping out a more reasonable and flexible macro machining process, while the computing program determined detailed sequences and cutting conditions at the detailed or micro-process planning stage.