Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

**Research Report**
RR-92-05

# Feature based Integration of CAD and CAPP

**Ansgar Bernardi, Christoph Klauck,
Ralf Legleitner, Michael Schulte,
Rainer Stark**

**February 1992**

**Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH**

Postfach 20 80
D-6750 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11, FRG
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

# Deutsches Forschungszentrum
# für
# Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern und Saarbrücken is a non-profit organization which was founded in 1988. The shareholder companies are Daimler Benz, Fraunhofer Gesellschaft, GMD, IBM, Insiders, Krupp-Atlas, Mannesmann-Kienzle, Philips, Sema Group Systems, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

☐ Intelligent Engineering Systems
☐ Intelligent User Interfaces
☐ Intelligent Communication Networks
☐ Intelligent Cooperative Systems.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Prof. Dr. Gerhard Barth
Director

# Feature based Integration of CAD and CAPP

**Ansgar Bernardi[+], Christoph Klauck[+], Ralf Legleitner,[+]**
**Michael Schulte[*], Rainer Stark[*]**

DFKI-RR-92-05

**[+]: DFKI GmbH**
**[*]: Universität des Saarlandes, Lehrstuhl für Konstruktionstechnik/CAD**

A short version of this paper will be published in the Proceedings of the GI conference "CAD'92: Neue Konzepte zur Realisierung anwendungsorientierter CAD-Systeme", Springer-Verlag.

This work has been supported by a grant from The Federal Ministry for Research and Technology (FKZ ITW-8902 C4).

# Feature based Integration of CAD and CAPP

Ansgar Bernardi, Christoph Klauck, Ralf Legleitner [*]
German Research Center
for Artificial Intelligence Inc.
DFKI GmbH
P. O. Box 2080
D-6750 Kaiserslautern

Michael Schulte, Rainer Stark [†]
Fachbereich Ingenieurwissenschaften
Lehrstuhl für Konstruktionstechnik/CAD
Universität des Saarlandes
6600 Saarbrücken

February 14, 1992

### Abstract

To integrate CAD systems with other applications in the CIM world, two principal approaches are currently under development. The feature based CAD systems provide higher level primitives which support not only the generation of the drawing but also serve as basic input for other CIM components. Another approach enables any CIM component to recognize the higher level entities used in CAD systems out of a lower level data exchange format, which might be the internal representation of such systems as well as some standard data exchange format. In this paper the authors examine both approaches in more detail. First a conceptual model of CAD and - as an example of another CIM component - of CAPP is represented. Comparing these two models the authors investigate the possible integrations on the different levels and provide a concise terminology and advantages and disadvantages of the different approaches.

---

[*] Phone: ++49-631-205-3477, e-mail: klauck@dfki.uni-kl.de
[†] Phone: ++49-681-302-3607, e-mail: stark@olymp.cad.uni-sb.de

# Contents

1 Motivation 3

2 Conceptual Model of CAD 3

3 Design by Features 5

4 Conceptual Model of CAPP 8

5 CAD Features Versus CAPP Features 10

6 What are Features ? 12

7 Possible Integrations 14

8 Conclusion 17

# List of Figures

1 Conceptual model of CAD . . . 4
2 Compound and Single Features . . . 6
3 Design Feature Bearing . . . 7
4 Functional Features . . . 7
5 Conceptual Model of CAPP . . . 9
6 PIM . . . 10
7 Design Features 1 . . . 11
8 Design Features 2 . . . 12
9 Manufacturing Features . . . 13
10 Stand alone Systems . . . 15
11 Minimal Integration . . . 16
12 Close Integration . . . 16

# 1 Motivation

The expected advantages of a close coupling of CAD and CAPP are the same which also apply to any other CIM related connection between different components: The information interchange shall lead to better knowledge transfer, to shorter turnaround times and to improved feedback. In the end, higher flexibility and generally better results are expected.

# 2 Conceptual Model of CAD

The task of design in mechanical engineering is to come from an abstract or logic idea of a technical product or system to a specific solution which obeys certain constraints. In recent years many models have been developed to describe and prescribe this design process (e.g. [Cross89, Pahl88, VDI86, VDI73]). According to the reasonable comprehensive model of Pahl and Beitz [Pahl88] the design process comprises the following stages:

1. *Clarification of the task*
   Design activity of collecting information about the requirements - e.g. functions - to be embodied in the solution and about the constraints

2. *Conceptual design*
   Design activity of establishing function structures, searching for suitable solution principles, and combining them into first concept variants

3. *Embodiment design*
   Design activity of determining the layout and forms of the technical product or system in accordance with technical and economic considerations

4. *Detail design*
   Design activity of laying down the arrangement, form, dimensions, and surface properties of all individual parts; subsequently materials have to be specified, technical and economic feasibilities are re-checked, and all technical drawings and other production documents are produced.

When classifying the conceptual framework of current CAD systems into the above described model of the design process it is only the fourth stage - *detail design* - which represents the predestinated stage of designing with the help of CAD systems. Since the heart of a conventional mechanical CAD system is its geometric modeller the use of current CAD systems is mainly restricted to allowing and supporting the designer to conceive, evolve, and document the design in terms of technical drawings.

According to Koller [Koller89] it is usually not worthwhile extending the conceptual framework of future CAD systems to earlier design stages than *detail design*. The main use of CAD systems should be concentrated on the design tasks which occur most often

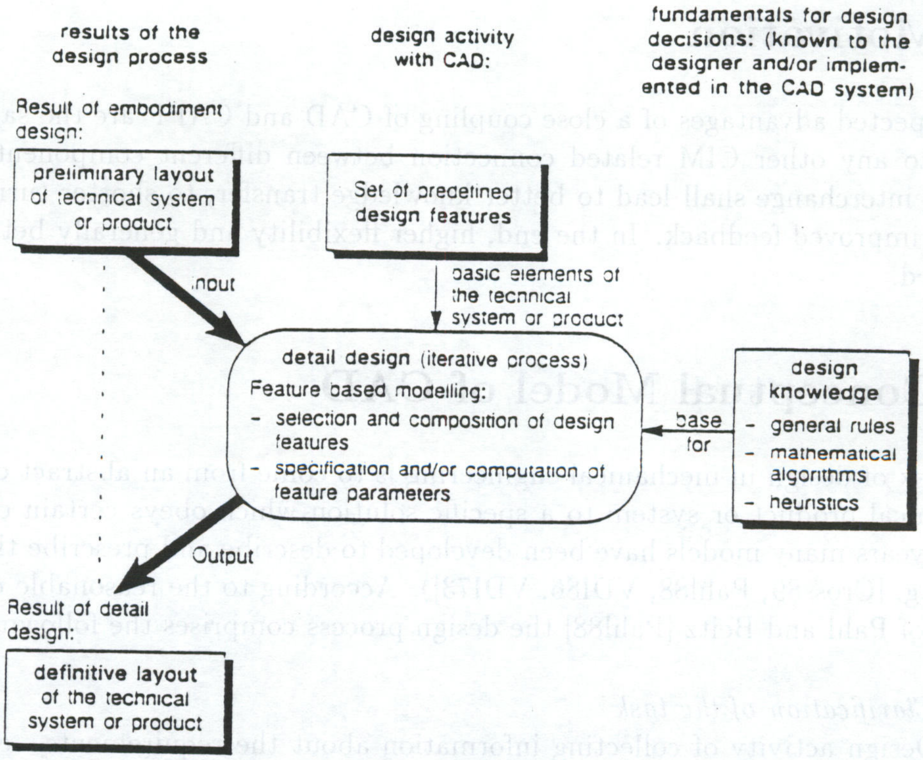

Figure 1: Conceptual model of CAD: feature based modelling at the stage of *detail design*

in design problems: the design and re-design of already - at least in principle - known solutions. But to come to more effective support of the designer CAD systems have to be extended to more intelligent design tools. For instance, it is desirable to develop CAD systems which are able to heed design rules and design interrelationships or to meet requirements of strength.

Besides the domain of geometric modelling which already has obtained a very high, sophisticated degree, CAD systems can be made more intelligent in terms of non-geometric technical information (e.g. functional requirement specifications, material properties, etc.) and administrative information (e.g. standards for common parts, evolution and versions of design, related families of parts, scheduling, inventory, etc.) [Arbab87].

One approach of coming to more intelligent design tools within CAD systems is to integrate the concept of design by features (cf. [Cunningham88, Dixon87, Pratt85, Shah91, Shah88]) into a feature based modeller (cf. [Krause88]). Further explanations to the concept of design by features are made in the following chapter of this article.

The conceptual framework for intelligent CAD systems in terms of feature based modelling can be illustrated as figure 1 shows:

According to the described design process the preliminary layout of the technical system or product (result of design stage three: *embodiment design*) serves as input for the *detail design* whereas the definitive layout represents the output. The activity of feature based modelling is characterized by both, the selection/composition process and the specification/computation process. To be able to make decisions in a design process the user - designer - and/or the CAD system need help in the form of specific design knowledge (cf. [Stark91]). If the design problems do not vary to a high degree, the effort of implementing design rules into CAD systems or into other supporting design systems (e.g. knowledge based systems for design) is justified. Since the designer intends to use only a certain number of design features (cf. the following chapter) for specific design problems, it is useful to provide a library of them within the CAD system.

## 3 Design by Features

Current CAD systems usually provide engineering drawings (2D), resp. wire frame models, surface models, solid boundary representation models, or solid constructive geometry models (3D). This implies that a product or assembly unit - sub-assembly or individual part - is represented by sets of points, lines, surfaces, and/or primitive volumes. This type of representation influences the way designers have to work with CAD systems. To a certain extent the designer is forced to translate his (high level feature) conception of the assembly units into sets of points, lines, surfaces, and/or primitive volumes.

The term feature is used to indicate form elements that are described on a higher semantic level than those primitives that can be found in the traditional geometric models. The conviction has been accepted that different classes of features have to be used in different engineering domains. In [Shah88] feature definitions depend on disciplines like engineering design or process planning. In the authors' view features used in engineering design have to be "elements used in generating, analyzing, or evaluating design" and in geometric modelling features are "groupings of geometric and/or topological entities that need to be referenced together".

However, in this paper features for design - from now on called design features - have to meet the following two requirements:

- They are mappable to a generic shape.
- They have an engineering significance resp. a semantic meaning in engineering design.

In design the engineering significance can be seen from four points of view:

1. required function,
2. manufacturing,
3. assembly, and last but not least
4. strength.

To determine the design of an assembly unit the designer first has to create those parts of the geometry that are necessary for realizing the required functions. Furthermore, he has to ensure that the assembly unit can be manufactured and mounted. Finally, the assembly unit has to fulfill the conditions of strength.

In the context of design by features the designer determines the geometry of assembly units by employing design features for all those four points of view. For instance, the

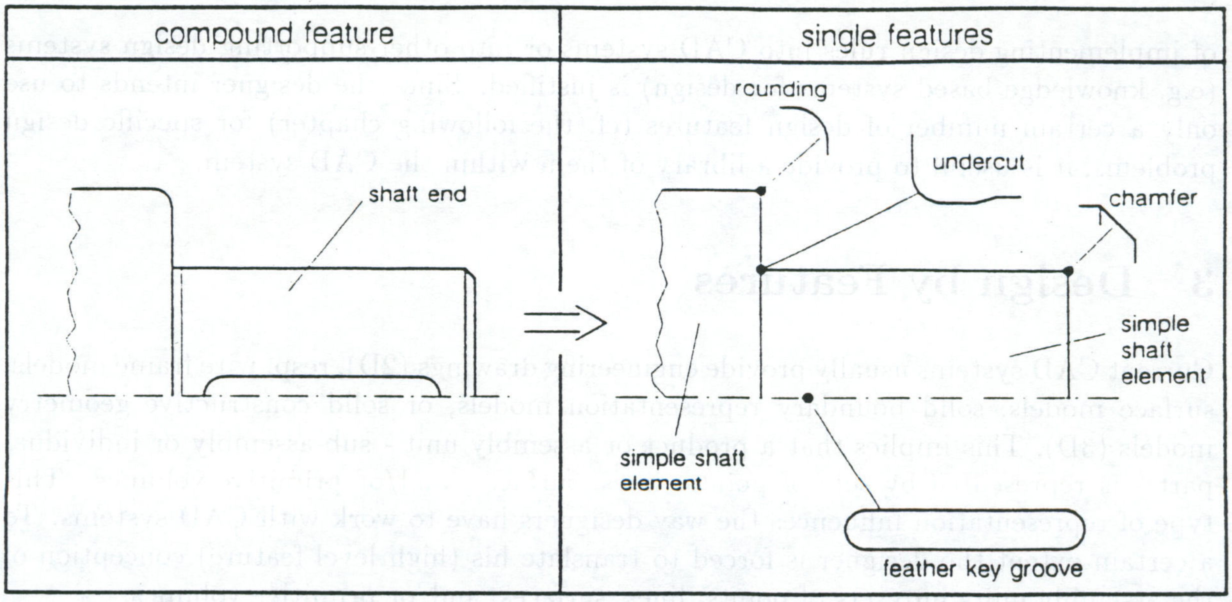

Figure 2: Compound feature shaft end and its single features

design feature *feather key groove* serves for the transmission of torque in connection with a feather key - point of view *function*; the feature *chamfer* makes mounting easier - point of view *assembly*. Some design features influence the design in more than one aspect. The feature *undercut* realizes a space for running out of a tool - point of view *manufacturing* - as well as it helps to minimize the stress concentration at the transition from one shaft section to another one - point of view *strength*.

Furthermore, features can be classified according to their level of complexity. Two terms are used: *single features* and *compound features*. Compound features are a combination of more elementary features which may themselves be compound features. Single features are the lowest order canonical forms supported by a feature based system. The usefulness of the concept of compound features is the generation and manipulation of features at multiple levels. A related group resp. compound feature could be manipulated as a unit rather than working on each single feature individually. The feature *shaft end* in figure 2 can be disintegrated into its sub-features - single features - *shaft shoulder*, *rounding*, *undercut*, *feather key groove*, and *chamfer*. The ability to capture the relationships between the single features in a compound definition is useful, as well.

Using design features two different feature classes can be distinguished: Design features like *bearing application* or *gearing* are not related to a single part. E.g. a feature *bearing application* can have effects on different parts of a housing as well as on different parts of a shaft (cf. figure 3). Furthermore, the feature may include two bearings and parts to fix the bearings on the shaft and on the casing - e.g. spring rings. However, most design features described in the literature are related to single parts. The feature *shaft end* and its single features shown in figure 2 are examples for such a feature type. The above described features refering to a group of parts can be decomposed into several features related to single parts. Figure 4 shows the detail design of the seats of the rolling bearings on the shaft.

At the beginning of the feature based modelling process the designer mostly uses features which are referred to a group of parts. The more the detail design process is advanced the less complex features are used. The design features used at the end of the

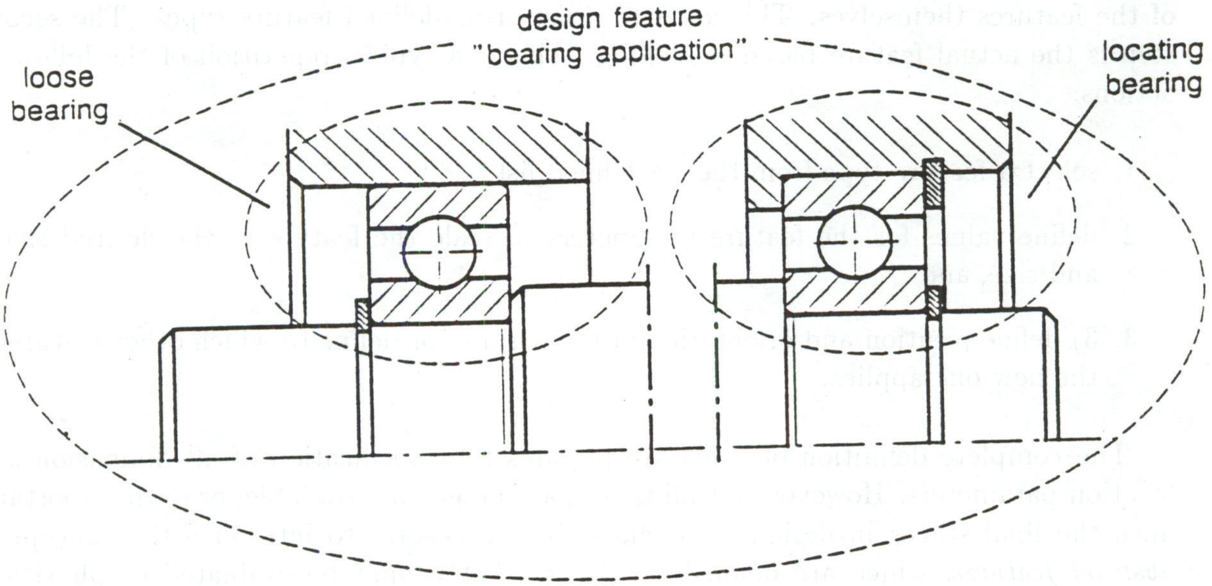

Figure 3: Design feature bearing application

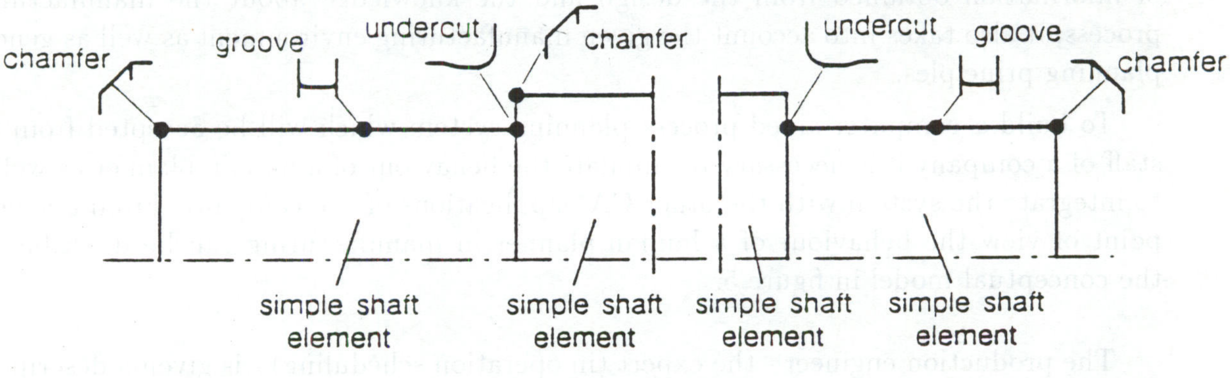

Figure 4: Seats of the rolling bearings on a shaft

detail design process are just referenced to single parts.

Design by features can be seen as a two-step process. The first step is the modelling of the features themselves. This leads to a list of predefined feature types. The second step is the actual feature based modelling. This is a typical repetition of the following actions:

1. select a feature type from the predefined list,
2. define values for the feature parameters to scale the feature to the desired shape and size, and
3. 3) define position and orientation of the feature, or define to which other feature(s) the new one applies.

The complete definition of a feature requires the specification of all dimension and location parameters. However, not all these parameters are available, or even important, until the final stages in design. Therefore, it is necessary to introduce the concept of *abstract features*, which are defined as entities that cannot be evaluated or physically realized until all variables have been specified (cf. [Shah91]).

It depends on the definition of the term design feature if besides geometry characteristics such as type of material, surface finish, dimensional or shape tolerances, and relations between design features - e.g. parallelism and mating surfaces - can be expressed as well. Otherwise, property features, precision features, and/or assembly features have to be used in addition to the design features.

# 4 Conceptual Model of CAPP

The task of process planning is the generation of a sequence of actions which must be performed in order to manufacture a given workpiece. The generation starts on the basis of information obtained from the design and the knowledge about the manufacturing process; it also takes into account the given manufacturing environment as well as general planning principles.

To build a computer aided process planning system which will be accepted from the staff of a company it is necessary to simulate the behaviour of a human planner as well as to integrate the system with the other CA* applications of the company. From a general point of view the behaviour of a human planner in manufacturing can be described by the conceptual model in figure 5.

The production engineer - the expert (in operation scheduling) - is given a description of the workpiece. This description consists of all geometrical and technological data which are necessary for the generation of a process plan. In this description the expert identifies characteristical parts or areas which are related to special information about the manufacturing process, e.g. an insertion. This parts are the so-called *features* (for manufacturing) and can be seen as an abstraction of the manufacturing task to build a structure of the manufacturing problem. According to these features, the expert selects

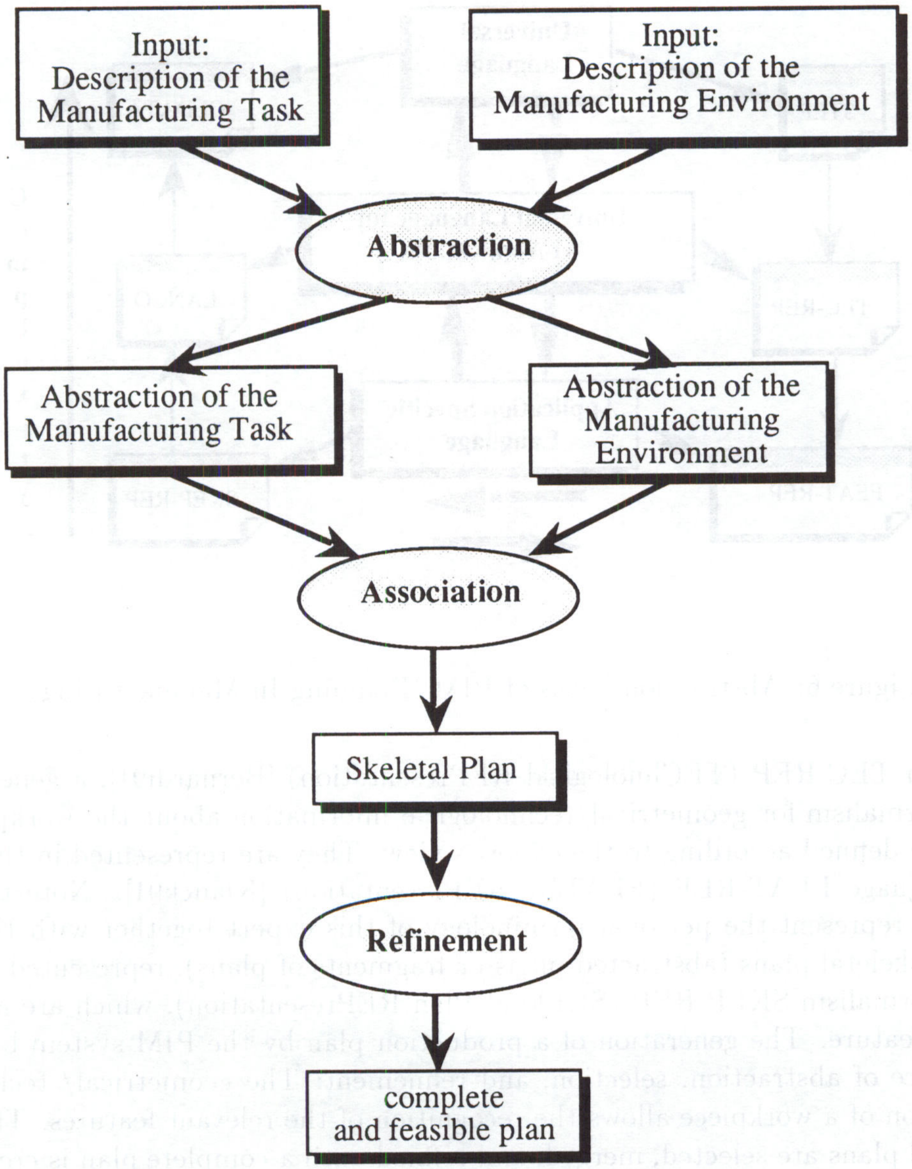

Figure 5: Conceptual Model of Process Planning

out of his memory (or out of existing plan libraries) generalized plan fragments, the so-called *skeletal plans.* By combining these skeletal plans according to the feature structure and by adapting them to the concrete workpiece, a complete production plan can be created. This conceptual model of an expert's way of process planning was simulated by the implementation of the prototypical system PIM [Legleitner92].

The different abstraction and refinement steps the expert performs lead to the definition of suitable domain specific higher level representation languages which allow the adequate representation of the expert's terminology and know-how. The integration of the PIM system into the CIM area was realized by connections to existing interfaces from the CAD and CAM world like STEP [1] and CLDATA[2]. The main idea of this higher level language system is shown in figure 6.

---

[1] STandard for the Exchange of Product Model Data, ISO TC 184/SC 4, NAM 96.4

[2] Cutter Location DATA (DIN 66025)

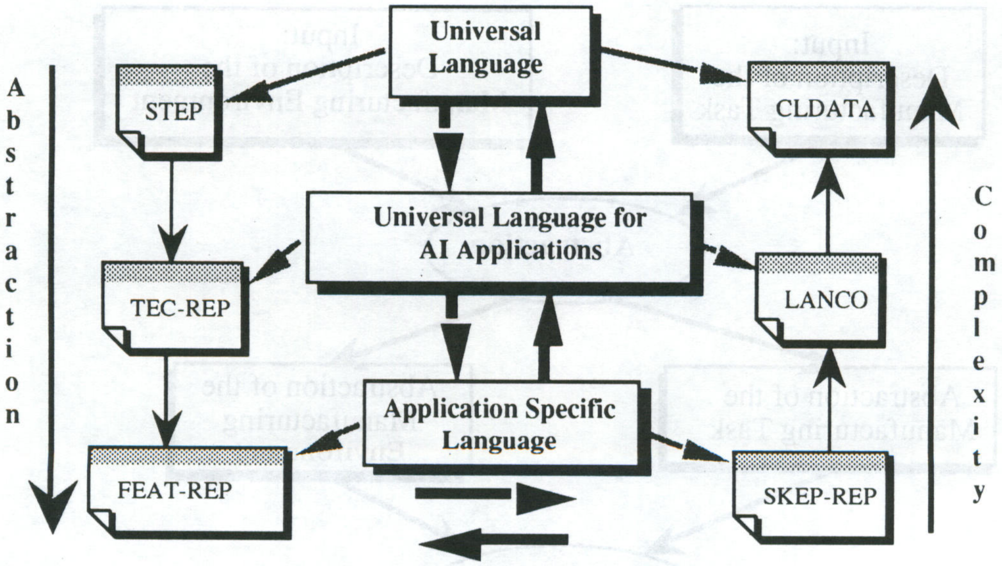

Figure 6: Abstraction levels of PIM (Planning In Manufacturing)

Based on TEC-REP (TEChnological REPresentation) [Bernardi91], a general representation formalism for geometrical/technological information about the workpiece, features can be defined according to the expert's view. They are represented in the domain specific language FEAT-REP (FEATure REPresentation) [Klauck91]. Note that these descriptions represent the personal terminology of this expert together with the sets of alternative skeletal plans (abstracted plans or fragments of plans), represented in the hierarchical formalism SKEP-REP (SKEletal Plan REPresentation), which are associated with every feature. The generation of a production plan by the PIM system boils down to a sequence of abstraction, selection, and refinement: The geometrical/ technological representation of a workpiece allows the recognition of the relevant features. The associated skeletal plans are selected, merged, and refined until a complete plan is created. (cf. [Becker91])

## 5 CAD Features Versus CAPP Features

The principal description of the CAD process and the conceptual model of CAPP employed the term *features* to denote some higher level entities which represent knowledge of the respective experts. Using the example presented in figure 3 the authors now compare the features used in the different areas.

The CAD expert thinks in terms of functionality which results in an appropriate feature structure of the workpiece under construction. This feature structure of an example is illustrated in figure 7. The functional entity *bearing* consists of a loose bearing seat and a locating bearing seat; the loose bearing seat is build upon the single design features groove, cylinder and undercut. These geometric entities are the direct results of the functional description. The other points of view mentioned in section 3 lead to more details

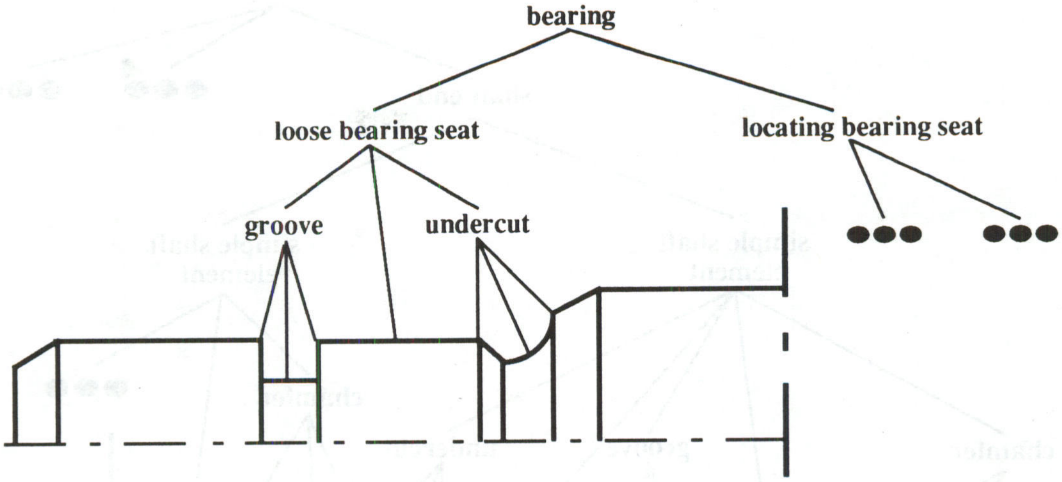

Figure 7: Design Features - Functional View

of the realization, e.g. the functional design feature *loose bearing seat* is associated to the geometrical design feature *shaft end.* The final result is shown in figure 8, where those geometrical design features are presented which describe the geometrical and technological phenotype of the designed shaft.

The expert in process planning starts from the geometrical/technological description of the workpiece which is the result of the CAD process. Based on this information the expert recognizes his own features. In the example this may lead to a feature structure as illustrated in figure 9: The basic surfaces as primitive elements are aggregated to manufacturing dependent higher level entities which the expert can associate with fragments of the process plan, the skeletal plans.

Comparing the features of the different experts it has to be noted that

1. Some basic features arise in both feature structures, e.g. the groove. While the described entities are the same in both areas, the experts associate different information with them.
2. The domain specific view of the experts may result in different names for the same geometric entity: The *chamfer* of the CAD expert is a functionality related term - point of view assembly -, whereas the process planner calls the same entity *trunnion* because of manufacturing aspects.
3. On higher levels the feature structures differ: The lower level entities are aggregated in different ways. This is the direct result of the domain dependent point of view of the experts. While the *bearing* as a higher level functional feature unites several surfaces forming the two bearing seats, these surfaces do not form a single manufacturing feature, since these surfaces appear in an ascending part and a descending part of the shaft and therefore in general cannot be manufactured without tool change or chucking change.

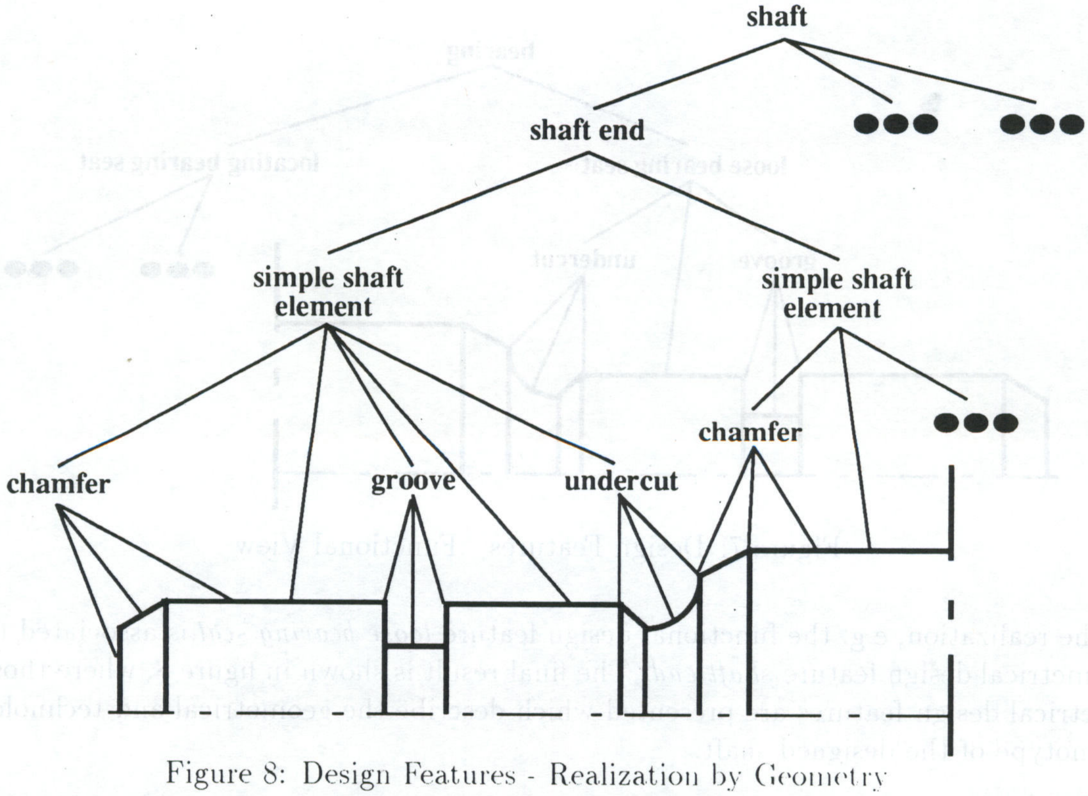

Figure 8: Design Features - Realization by Geometry

In summary, while some features are used in both areas, principal differences in single features as well as in feature structures separate the domains. Higher features which bear more information lead in general to bigger differences. However, the idea of a *feature* as some entity representing expert's knowledge and aggregating lower level information is the same in both areas. This results in certain characteristics of features which the authors investigate in the next section.

## 6 What are Features ?

Currently there is no consensus on a precise definition of the term *feature*. Most researchers working in this area agree that a feature is an abstraction of lower level design or manufacturing information [Dixon89a]. Features that are required for design may differ considerably from those required for manufacturing or assembly, even though they may be based on the same lower level entities. This was discussed in more detail in the previous section.

John R. Dixon and John J. Cunningham have defined a feature as "*any geometric form or entity that is used in reasoning in one or more design or manufacturing activities*" [Cunningham88]. T.-C. Chang has defined a feature in his book [Chang90] as "*a subset of geometry on an engineering part which has a special design or manufacturing characteristic.*". Other similar definitions of features can be found in [Dixon89b].

**Definition** Based on the requirements pointed out in chapter 3 the authors define the term *feature* as a description element based on geometrical and technological data of a product which an expert in a domain associates with certain information.

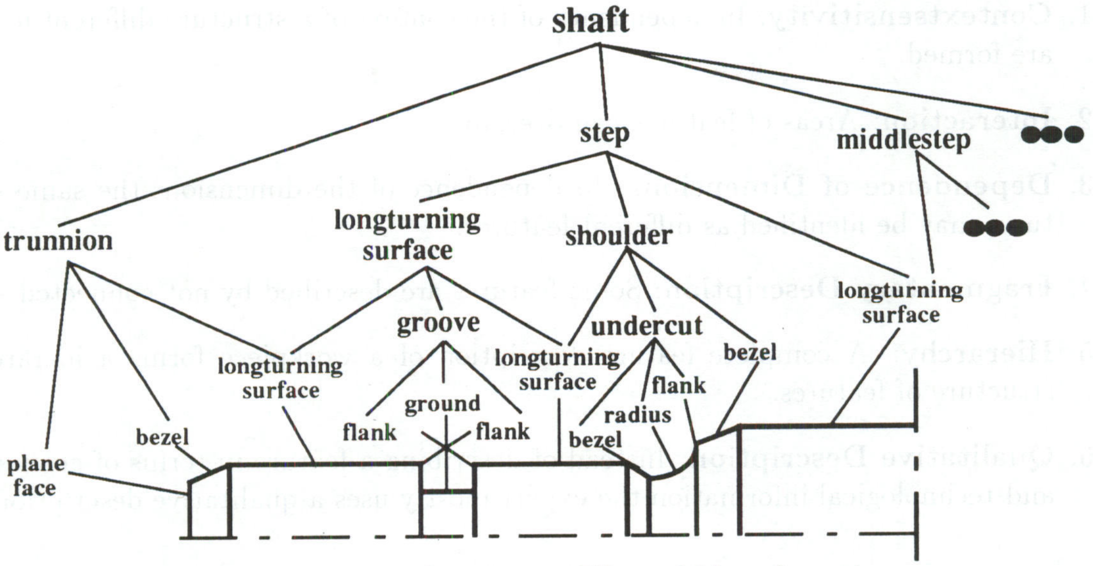

Figure 9: Feature Structure - View of Manufacturing

They are firstly distinguished by their kind as:

1. *functional features*, e.g. seat of the rolling bearing or O-ring groove (cf. [Weber92]),
2. *qualitative features*, e.g. bars or solid workpiece,
3. *geometrical (form) features*, e.g. shoulder, groove or drilled hole,
4. *atomic features*, e.g. toroidal shell, ring, shape tolerance or surface finish.

and they are secondly distinguished by their application as

1. *design features*, e.g. crank or coupler (cf. the more detail classification of design features in section 3),
2. *manufacturing features*:
    - (a) *turning features*, e.g. shoulder or neck,
    - (b) *milling features*, e.g. step or pocket,
    - (c) *drilling features*, e.g. stepped hole or lowering,
    - (d) ...
3. ...

In the area of CAD the design features are additionally distinguished by their associated information as single features and compound features whereas the single features represent the smallest geometrical features with associated information. The compound features are defined via single features or compound features.

It is important to keep in mind that the above mentioned features describe a certain kind of a shape and that they are also related to some information about this shape. So the proposed feature language has a syntax (shape description) and a semantics (description of related information). In [Klauck91] several syntactical characteristics of features have been outlined.

1. **Contextsensitivity**: In dependence of the context of a structure different features are formed.
2. **Interaction**: Areas of features can overlap.
3. **Dependence of Dimensions**: In dependence of the dimensions the same structures may be identified as different features.
4. **Fragmentary Description**: Some features are described by not connected areas.
5. **Hierarchy**: A complete feature description of a workpiece forms a hierarchical structure of features.
6. **Qualitative Description**: Instead of describing a feature in terms of geometrical and technological information the expert mostly uses a qualitative description.

It is obvious that the phenotype of these characteristics differ, if the same workpiece is described in CAD features based on functional requirements or in CAPP features based on the available manufacturing processes, as we have described in the previous section.

The analogue between the feature language and formal language with semantics is explained in [Klauck91]. There it is stated out that the geometrical description in addition to attributes about the context, functionality and technology forms the syntax of a feature. The information associated with the feature forms the semantics of a feature. A designer for example associates functionality and costs with his features whereas a manufacturer associates a set of skeletal plans with his features.

The area of formal languages is a well established field of research and provides a powerful set of methods like parsing and knowledge about problems, their complexity, and the way of how to solve them efficiently. The use of formal language techniques for feature descriptions facilitates the application of these results to the area of feature recognition (in CAPP) and feature expansion (in CAD). So a major component of the CAD/CAM integration can be realized with such techniques. (cf. [Klauck92a, Klauck92b])

## 7 Possible Integrations

To integrate CAD systems with other applications in the CIM world, two principal approaches are currently under development [Chang90]. The feature based CAD systems provide higher level primitives which support not only the generation of the design but also serve as basic input for other CIM components. Another approach enables any CIM component to recognize the higher level entities used in this component out of a lower level data exchange format which might be the internal representation of a CAD system as well as some standard data exchange format.

Feature based CA* systems of the future have to offer both kinds of integration: The former to allow a more efficient integration of feature based CA* systems and the latter to guarantee an integration with other CA* systems and to make the achieved data of a company like technical drawings usable in the systems.

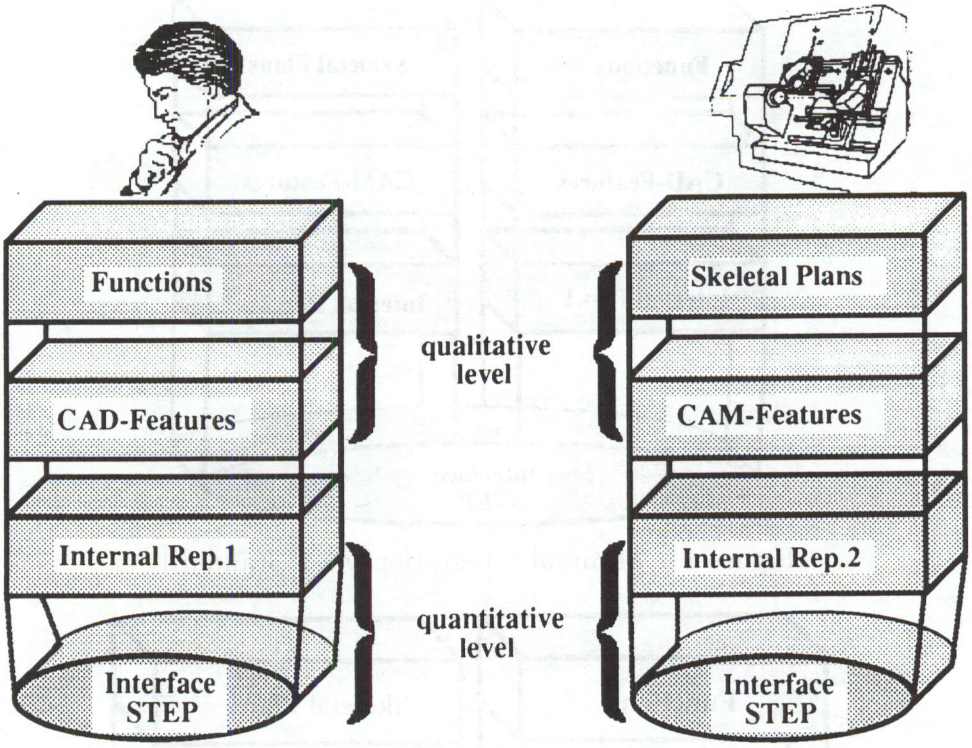

Figure 10: Principle view of CAD and CAPP

The basic functionality of the different systems and their principle design is not affected by the selected kind of integration. Both CAD systems and CAPP systems rely on some internal representation, which is tailored to their respective needs (cf. figure 10). This quantitative data level is connected to some external interface, ideally a standard like STEP. In special cases this standard can be used as the internal representation of the system; in general a well behaving transformation procedure can be used.

The qualitative levels are based on the quantitative information. The features used on these levels bridge the gap between the quantitative level and the expert's way of thinking. The transformation between the quantitative and the qualitative level is complex and expensive. The expert's knowledge must be represented and used in suitable tools in order to realize this step. Nevertheless this transition is feasible, which is proved by the analogue to formal languages where such transformations are already realized, e.g. described in [Legleitner92].

According to this model, a self-evident solution is the integration on the quantitative level via a standard like STEP, illustrated in figure 11. Note that the standard contains no qualitative information. As outlined in the previous sections a standardization of the qualitative information like the feature definitions is in general impossible, even though the underlying principles of the feature definitions and the resulting representation languages are similar (cf. [Klauck91]).

The more interesting integration of feature based CAD and CAPP systems is illustrated in figure 12. The integration will be realized by a so-called *integration function* **INT**. This function INT is defined according to three cases:

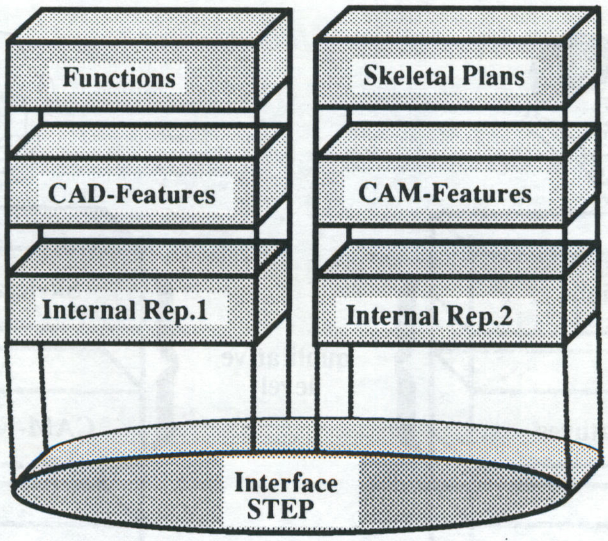

Figure 11: Minimal Integration via a Standard

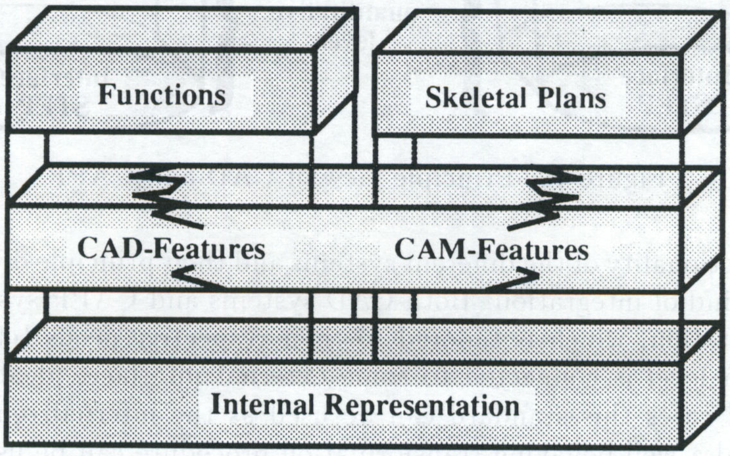

Figure 12: Close Integration on the qualitative level

1. The CAD feature have the same structure (syntax) as a CAM feature: INT is the bijective identity function.
2. The CAD feature (respectively CAM feature) have the same structure as a set of complex CAM features: The CAD feature is reproduced by INT to the set of complex CAM features.
3. The CAD feature (respectively CAM feature) is only describable by a set of atomic CAM features: The CAD feature is reproduced by INT to the set of atomic CAM features.

In the last two cases a feature recognition process is necessary to build the complete CAM feature structure out of the result of INT. This is necessary because the feature structure of a workpiece always forms a hierarchical structure as mentioned in section 6. Because all possible surfaces are contained in the set of atomic CAM features respectively CAD features INT can reproduce every CAD feature to a set of CAM features and vice versa.

The advantage of the integration model of figure 12 lies in the distinction between the time-complexity of the integration function INT and the feature recognition process. The former realize a hard link between two structures in a constant time whereas the latter realize a generation of a structure in a maybe non polynomial time. Independent of the kind of integration the user's view of the CAD or CAPP system is always the same: The system can be tailored to the domain experts terminology, the selected kind of integration may influence only the efficiency, not the terminology.

## 8 Conclusion

Features in CAD and CAPP represent specific experts knowledge; interesting features - which represent more knowledge - are very domain specific. Because of this fact a standardization of features in general and especially on a higher level seems impossible.

Nevertheless, an integration on the qualitative level is at least possible. Due to similarities in the structure of the feature definitions an integration function INT can be found. Using a standardized representation formalism for the feature definitions, this function can be generated automatically.

## References

[Arbab87] Arbab, F.: A Paradigm for Intelligent CAD. Eurographics Workshops on Intelligent CAD Systems (1st: 1987: Noordwijkerhout, Netherlands). Intelligent CAD Systems I: theoretical and methodological aspects/ edited by P.J.W. ten Hagen and T. Tomiyama - (Eurographic seminars); pp. 20-39; 1987.

[Becker91] Becker, A.: Analyse der Planungsverfahren der KI im Hinblick auf ihre Eignung für die Arbeitsplanung. D-91-17, DFKI GmbH, 1991.

[Bernardi91] Bernardi, A.; Klauck, Ch.; Legleitner, R.: TEC-REP: Repräsentation von Geometrie- und Technologieinformationen. D-91-07, DFKI GmbH, 1991.

[Chang90] Chang, T.-C.: Expert Process Planning for Manufacturing. Addison-Wesley, 1990.

[Cross89] Cross, N.: Engineering Design Methods. Chichester, New York, ···: John Wiley & Sons Ltd., 1989.

[Cunningham88] Cunningham, J.J.; Dixon, J.R.: Designing With Features: The Origin of Features. MDA Technical Report 3-88. Proceedings of the ASME Computers in Engineering Conference, San Francisco, CA, July 31 - August 3, 1988

[Dixon87] Dixon, J.R.; Cunningham, J.J.; Simmons, M.K.: Research in Designing With Features. MDA Technical Report 4-87. Proceedings of the IFIP WG 5.2 Workshop on Intelligent CAD, Cambridge, MA, October 6-7, 1987.

[Dixon89a] Dixon, J.R.; Finger, S.: A Review of Research in Mechanical Engineering Design. Part I: Descriptive, Prescriptive, and Computer-Based Models of Design Processes. Engineering Design, Springer-Verlag, 1989, pp. 51-67.

[Dixon89b] Dixon, J. R.; Finger, S.: A Review of Research in Mechanical Engineering Design. Part II: Representations, Analysis, and Design for the Life Cycle. Engineering Design, Springer-Verlag, 1989, pp. 121-137.

[Klauck91] Klauck, Ch.; Bernardi, A.; Legleitner, R.: FEAT-REP: Representing Features in CAD/CAM. IV International Symposium on Artificial Intelligence: Applications in Informatics, 1991.

[Klauck92a] Klauck, Ch.; Mauss, J.: A Heuristic Driven Parser for Attributed Node Labeled Graph Grammars and its Application to Feature Recognition in CIM. *forthcomin*, 1992.

[Klauck92b] Klauck, Ch.; Schwagereit, J.: Integration of Graph Gramnmars andTaxonomies for Feature Representation in CAD/CAM. *forthcoming*, 1992.

[Koller89] Koller, R.: CAD: automatisches Zeichnen, Darstellen und Konstruieren. Berlin, Springer-Verlag, 1989.

[Krause88] Krause, F.-L.; Vosgerau, F.H.; Yaramanoglu, N.: Implementation of Technical Rules in a Feature Based Modeller. Eurographics Workshops on Intelligent CAD Systems (2nd: 1988: Veldhoven, Netherlands). Intelligent CAD Systems II: implemental issues/ edited by V. Akman, P.J.W. ten Hagen and P.J. Veerkamp - (Eurographic seminars); pp. 195-208; 1988.

[Legleitner92] Legleitner, R.; Bernardi, A.; Klauck, Ch.: PIM: Skeletal Plan based CAPP. International Conference on Manufacturing Automation, forthcoming, 1992.

[Pahl88] Pahl, G.; Beitz, W.: Engineering Design, a systematic approach. Berlin, Springer-Verlag, 1988.

[Pratt85] Pratt, M.J.; Wilson, P.R.: Requirements for Support of Form Features in a Solid Modelling System. R-85-ASPP-01, CAM-I, Arlington, Texas, 1985.

[Shah88] Shah, J.J. at all: Current Status of Feature Technology. R-88-GM-04.1, CAM-I, Arlington, Texas, 1988.

[Shah91] Shah, J.J.: Conceptual Development of Form Features Modelers. Research in Engineering Design, Vol. Feb. 1991, pp. 93-108.

[Stark91] Stark, R.; Weber, C.: Wissensbasierte Systeme für die Konstruktion - Grundlagen aus konstruktionsmethodischer Sicht. International Conference on Engineering Design, 1991 (ICED 91). Schriftenreihe "Workshop-Design-Konstruktion", WDK 20: Proceedings of ICED 91, Vol. 2, pp. 1151-1161. Heurista-Verlag, Zürich, 1991.

[VDI73] Guideline VDI 2222, Sheet 1: Konstruktionsmethodik, Konzipieren technischer Produkte. Düsseldorf: VDI-Verlag, 1973.

[VDI86] Guideline VDI 2221: Methodik zum Entwickeln und Konstruieren technischer Systeme und Produkte. Düsseldorf: VDI-Verlag, 1986.

[Weber92] Weber, C.; Schulte, M.; Stark, R.: Feature-based Modeling for Design: Functional Features for Design in Mechanical Engineering. Proceedings of the 8th International Conference on CAD/CAM, Robotics and Factories of the Future, Metz, France, 1992.

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
D-6750 Kaiserslautern
FRG

## DFKI Publikationen

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse bezogen werden.
Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

## DFKI Publications

The following DFKI publications or the list of all publisched papers so far can be ordered from the above address.
The reports are distributed free of charge except if otherwise indicated.

---

### DFKI Research Reports

**RR-90-15**
*Harald Trost:* The Application of Two-level Morphology to Non-concatenative German Morphology
13 pages

**RR-90-16**
*Franz Baader, Werner Nutt:* Adding Homomorphisms to Commutative/Monoidal Theories, or: How Algebra Can Help in Equational Unification
25 pages

**RR-90-17**
*Stephan Busemann:*
Generalisierte Phasenstrukturgrammatiken und ihre Verwendung zur maschinellen Sprachverarbeitung
114 Seiten

**RR-91-01**
*Franz Baader, Hans-Jürgen Bürckert, Bernhard Nebel, Werner Nutt, Gert Smolka:* On the Expressivity of Feature Logics with Negation, Functional Uncertainty, and Sort Equations
20 pages

**RR-91-02**
*Francesco Donini, Bernhard Hollunder, Maurizio Lenzerini, Alberto Marchetti Spaccamela, Daniele Nardi, Werner Nutt:* The Complexity of Existential Quantification in Concept Languages
22 pages

**RR-91-03**
*B.Hollunder, Franz Baader:* Qualifying Number Restrictions in Concept Languages
34 pages

**RR-91-04**
*Harald Trost:* X2MORF: A Morphological Component Based on Augmented Two-Level Morphology
19 pages

**RR-91-05**
*Wolfgang Wahlster, Elisabeth André, Winfried Graf, Thomas Rist:* Designing Illustrated Texts: How Language Production is Influenced by Graphics Generation.
17 pages

**RR-91-06**
*Elisabeth André, Thomas Rist:* Synthesizing Illustrated Documents: A Plan-Based Approach
11 pages

**RR-91-07**
*Günter Neumann, Wolfgang Finkler:* A Head-Driven Approach to Incremental and Parallel Generation of Syntactic Structures
13 pages

**RR-91-08**
*Wolfgang Wahlster, Elisabeth André, Som Bandyopadhyay, Winfried Graf, Thomas Rist:* WIP: The Coordinated Generation of Multimodal Presentations from a Common Representation
23 pages

**RR-91-09**
*Hans-Jürgen Bürckert, Jürgen Müller, Achim Schupeta:* RATMAN and its Relation to Other Multi-Agent Testbeds
31 pages

**RR-91-10**
*Franz Baader, Philipp Hanschke:* A Scheme for Integrating Concrete Domains into Concept Languages
31 pages

**RR-91-11**
*Bernhard Nebel:* Belief Revision and Default Reasoning: Syntax-Based Approaches
37 pages

**RR-91-12**
*J.Mark Gawron, John Nerbonne, Stanley Peters:*
The Absorption Principle and E-Type Anaphora
33 pages

**RR-91-13**
*Gert Smolka:* Residuation and Guarded Rules for
Constraint Logic Programming
17 pages

**RR-91-14**
*Peter Breuer, Jürgen Müller:* A Two Level
Representation for Spatial Relations, Part I
27 pages

**RR-91-15**
*Bernhard Nebel, Gert Smolka:*
Attributive Description Formalisms ... and the Rest
of the World
20 pages

**RR-91-16**
*Stephan Busemann:* Using Pattern-Action Rules for
the Generation of GPSG Structures from Separate
Semantic Representations
18 pages

**RR-91-17**
*Andreas Dengel, Nelson M. Mattos:*
The Use of Abstraction Concepts for Representing
and Structuring Documents
17 pages

**RR-91-18**
*John Nerbonne, Klaus Netter, Abdel Kader Diagne,*
*Ludwig Dickmann, Judith Klein:*
A Diagnostic Tool for German Syntax
20 pages

**RR-91-19**
*Munindar P. Singh:* On the Commitments and
Precommitments of Limited Agents
15 pages

**RR-91-20**
*Christoph Klauck, Ansgar Bernardi, Ralf Legleitner*
FEAT-Rep: Representing Features in CAD/CAM
48 pages

**RR-91-21**
*Klaus Netter:* Clause Union and Verb Raising
Phenomena in German
38 pages

**RR-91-22**
*Andreas Dengel:* Self-Adapting Structuring and
Representation of Space
27 pages

**RR-91-23**
*Michael Richter, Ansgar Bernardi, Christoph*
*Klauck, Ralf Legleitner:* Akquisition und
Repräsentation von technischem Wissen für
Planungsaufgaben im Bereich der Fertigungstechnik
24 Seiten

**RR-91-24**
*Jochen Heinsohn:* A Hybrid Approach for
Modeling Uncertainty in Terminological Logics
22 pages

**RR-91-25**
*Karin Harbusch, Wolfgang Finkler, Anne Schauder:*
Incremental Syntax Generation with Tree Adjoining
Grammars
16 pages

**RR-91-26**
*M. Bauer, S. Biundo, D. Dengler, M. Hecking,*
*J. Koehler, G. Merziger:*
Integrated Plan Generation and Recognition
- A Logic-Based Approach -
17 pages

**RR-91-27**
*A. Bernardi, H. Boley, Ph. Hanschke,*
*K. Hinkelmann, Ch. Klauck, O. Kühn,*
*R. Legleitner, M. Meyer, M. M. Richter,*
*F. Schmalhofer, G. Schmidt, W. Sommer:*
ARC-TEC: Acquisition, Representation and
Compilation of Technical Knowledge
18 pages

**RR-91-28**
*Rolf Backofen, Harald Trost, Hans Uszkoreit:*
Linking Typed Feature Formalisms and
Terminological Knowledge Representation
Languages in Natural Language Front-Ends
11 pages

**RR-91-29**
*Hans Uszkoreit:* Strategies for Adding Control
Information to Declarative Grammars
17 pages

**RR-91-30**
*Dan Flickinger, John Nerbonne:*
Inheritance and Complementation: A Case Study of
*Easy* Adjectives and Related Nouns
39 pages

**RR-91-31**
*H.-U. Krieger, J. Nerbonne:*
Feature-Based Inheritance Networks for
Computational Lexicons
11 pages

**RR-91-32**
*Rolf Backofen, Lutz Euler, Günther Görz:*
Towards the Integration of Functions, Relations and
Types in an AI Programming Language
14 pages

**RR-91-33**
*Franz Baader, Klaus Schulz:*
Unification in the Union of Disjoint Equational Theories: Combining Decision Procedures
33 pages

**RR-91-34**
*Bernhard Nebel, Christer Bäckström:*
On the Computational Complexity of Temporal Projection and some related Problems
35 pages

**RR-91-35**
*Winfried Graf, Wolfgang Maaß:* Constraint-basierte Verarbeitung graphischen Wissens
14 Seiten

**RR-92-03**
*Harold Boley:*
Extended Logic-plus-Functional Programming
28 pages

**RR-92-04**
*John Nerbonne:* Feature-Based Lexicons: An Example and a Comparison to DATR
15 pages

**RR-92-05**
*Ansgar Bernardi, Christoph Klauck, Ralf Legleitner, Michael Schulte, Rainer Stark:*
Feature based Integration of CAD and CAPP
19 pages

---

## DFKI Technical Memos

**TM-91-01**
*Jana Köhler:* Approaches to the Reuse of Plan Schemata in Planning Formalisms
52 pages

**TM-91-02**
*Knut Hinkelmann:* Bidirectional Reasoning of Horn Clause Programs: Transformation and Compilation
20 pages

**TM-91-03**
*Otto Kühn, Marc Linster, Gabriele Schmidt:*
Clamping, COKAM, KADS, and OMOS: The Construction and Operationalization of a KADS Conceptual Model
20 pages

**TM-91-04**
*Harold Boley (Ed.):*
A sampler of Relational/Functional Definitions
12 pages

**TM-91-05**
*Jay C. Weber, Andreas Dengel, Rainer Bleisinger:*
Theoretical Consideration of Goal Recognition Aspects for Understanding Information in Business Letters
10 pages

**TM-91-06**
*Johannes Stein:* Aspects of Cooperating Agents
22 pages

**TM-91-08**
*Munindar P. Singh:* Social and Psychological Commitments in Multiagent Systems
11 pages

**TM-91-09**
*Munindar P. Singh:* On the Semantics of Protocols Among Distributed Intelligent Agents
18 pages

**TM-91-10**
*Béla Buschauer, Peter Poller, Anne Schauder, Karin Harbusch:* Tree Adjoining Grammars mit Unifikation
149 pages

**TM-91-11**
*Peter Wazinski:* Generating Spatial Descriptions for Cross-modal References
21 pages

**TM-91-12**
*Klaus Becker, Christoph Klauck, Johannes Schwagereit:* FEAT-PATR: Eine Erweiterung des D-PATR zur Feature-Erkennung in CAD/CAM
33 Seiten

**TM-91-13**
*Knut Hinkelmann:*
Forward Logic Evaluation: Developing a Compiler from a Partially Evaluated Meta Interpreter
16 pages

**TM-91-14**
*Rainer Bleisinger, Rainer Hoch, Andreas Dengel:*
ODA-based modeling for document analysis
14 pages

**TM-91-15**
*Stefan Bussmann:* Prototypical Concept Formation
An Alternative Approach to Knowledge Representation
28 pages

**TM-92-01**
*Lijuan Zhang:*
Entwurf und Implementierung eines Compilers zur Transformation von Werkstückrepräsentationen
34 Seiten

## DFKI Documents

**D-91-01**
*Werner Stein , Michael Sintek:* Relfun/X - An Experimental Prolog Implementation of Relfun
48 pages

**D-91-02**
*Jörg P. Müller:* Design and Implementation of a Finite Domain Constraint Logic Programming System based on PROLOG with Coroutining
127 pages

**D-91-03**
*Harold Boley, Klaus Elsbernd, Hans-Günther Hein, Thomas Krause:* RFM Manual: Compiling RELFUN into the Relational/Functional Machine
43 pages

**D-91-04**
DFKI Wissenschaftlich-Technischer Jahresbericht 1990
93 Seiten

**D-91-06**
*Gerd Kamp:* Entwurf, vergleichende Beschreibung und Integration eines Arbeitsplanerstellungssystems für Drehteile
130 Seiten

**D-91-07**
*Ansgar Bernardi, Christoph Klauck, Ralf Legleitner*
TEC-REP: Repräsentation von Geometrie- und Technologieinformationen
70 Seiten

**D-91-08**
*Thomas Krause:* Globale Datenflußanalyse und horizontale Compilation der relational-funktionalen Sprache RELFUN
137 Seiten

**D-91-09**
*David Powers, Lary Reeker (Eds.):*
Proceedings MLNLO'91 - Machine Learning of Natural Language and Ontology
211 pages
**Note:** This document is available only for a nominal charge of 25 DM (or 15 US-$).

**D-91-10**
*Donald R. Steiner, Jürgen Müller (Eds.):*
MAAMAW'91: Pre-Proceedings of the 3rd European Workshop on „Modeling Autonomous Agents and Multi-Agent Worlds“
246 pages
**Note:** This document is available only for a nominal charge of 25 DM (or 15 US-$).

**D-91-11**
*Thilo C. Horstmann:*Distributed Truth Maintenance
61 pages

**D-91-12**
*Bernd Bachmann:*
$\mathrm{H}^{\mathrm{iera}}\mathrm{C}_{\mathrm{on}}$ - a Knowledge Representation System with Typed Hierarchies and Constraints
75 pages

**D-91-13**
International Workshop on Terminological Logics
*Organizers: Bernhard Nebel, Christof Peltason, Kai von Luck*
131 pages

**D-91-14**
*Erich Achilles, Bernhard Hollunder, Armin Laux, Jörg-Peter Mohren: KRIS :* Knowledge Representation and Inference System
- Benutzerhandbuch -
28 Seiten

**D-91-15**
*Harold Boley, Philipp Hanschke, Martin Harm, Knut Hinkelmann, Thomas Labisch, Manfred Meyer, Jörg Müller, Thomas Oltzen, Michael Sintek, Werner Stein, Frank Steinle:*
µCAD2NC: A Declarative Lathe-Worplanning Model Transforming CAD-like Geometries into Abstract NC Programs
100 pages

**D-91-16**
*Jörg Thoben, Franz Schmalhofer, Thomas Reinartz:* Wiederholungs-, Varianten- und Neuplanung bei der Fertigung rotationssymmetrischer Drehteile
134 Seiten

**D-91-17**
*Andreas Becker:*
Analyse der Planungsverfahren der KI im Hinblick auf ihre Eignung für die Abeitsplanung
86 Seiten

**D-91-18**
*Thomas Reinartz:* Definition von Problemklassen im Maschinenbau als eine Begriffsbildungsaufgabe
107 Seiten

**D-91-19**
*Peter Wazinski:* Objektlokalisation in graphischen Darstellungen
110 Seiten

**Ansgar Bernardi, Christoph Klauck, Ralf Legleitner, Michael Schulte, Rainer Stark**

Research Report