Product range models supporting design knowledge reuse

C A Costa[1] and **R I M Young**[2]*
[1]Department of Mechanical Engineering, University of Caxias do Sul, Brazil
[2]Department of Manufacturing Engineering, Loughborough University, Loughborough, Leicestershire, UK

Abstract: Redesign, where previous information is recovered in order to be adapted to a new situation, is an area of design where information technology can potentially provide substantial benefits. Information support to product design and manufacturing has been pursued through the use of product and manufacturing models. This paper introduces a new concept of a complementary information model, called a product range model, that aims to support variant and adaptive design activities. The general concept and structure of such an information model is defined in terms of product functions and their respective design solutions. The interactions taking place between particular design solution options are discussed, and methods are proposed for their evaluation against product specifications and design constraints. The concept of knowledge links is introduced to maintain the relationships between solutions within the product range model and the particular model of the product being developed. The work has been explored using injection mould tooling as an appropriate product range and evaluated through the design and implementation of a design support system utilizing an object-oriented database.

Keywords: information modelling, injection mould, variant design, information reuse, interactions, product range model

1 INTRODUCTION

The reuse of information to support design activities is attracting significant attention by the research community [1]. Similarly, information models have been recognized as one of the main elements in integrated computer aided engineering system architectures to support design and manufacturing applications through the product life cycle [2, 3]. This paper argues that, in addition to the traditional concept of a product model as a source and repository for product information, a further model, termed here a product range model, can enhance integrated product development systems by enabling the reuse of past product design information and knowledge.

The application of knowledge in design automation has grown over recent years as vendors have begun to offer 'high-end computer aided design (CAD)' tools that can use knowledge linked with product geometry and data management, such as Parametric Technology's behavioural modelling, Unigraphics's UG/Wave, IBM/ Catia's Knowledgeware, and Knowledge Technologies

International's ICAD [4]. However, these are stand-alone solutions to particular problems. Solutions that support team-based design effectively must operate in a more open information support environment.

A number of computational approaches have been applied to the area of reuse of information and knowledge to support engineering design [5]. Case-based reasoning has had significant attention as a tool to support design reuse [6, 7], where cases are recovered from a library and adapted to the new situation. These, however, also offer stand-alone solutions rather than the integrated solutions that are necessary to support broad-based design and manufacture. Such integrated solutions can be achieved through the use of information models to support design and manufacture applications based on the knowledge-based systems (KBS) approach [8, 9].

The concept of a product range model, identified by Lee and Young [10] as a potential link between design for function and design for manufacture, is explored in this paper in terms of its structure and its relationship with product model structures. The combination of a product range model and a product model aims to provide designers with appropriate support to enable them to make design decisions based on established principles and past experience in adaptive design cases.
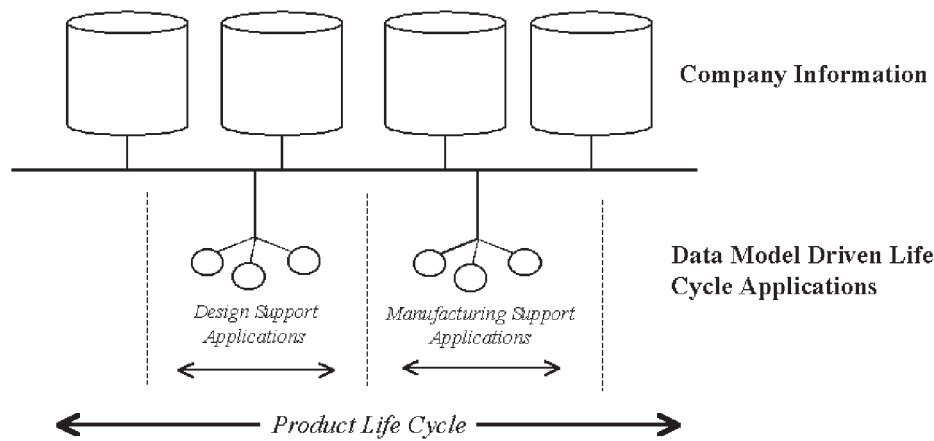
Fig. 1   Integrated information architecture for CAE systems

The application of product models to support product design and development decisions has gained significant attention by the industrial community in recent years [11]. The definition of an integrated product data representation to share and exchange data is also defined in ISO 10303-STEP, particularly in the generic product description resources [12, 13]. The structure of the product data model is critical to enabling software applications to share and store data in the product model [14]. Similarly, the structure of a product range model is significant in the capture and sharing of instance data related to the reuse of previous design solutions.

Recent work on product ranges has focused on the structure of product information on existing designs [15–17]. The ISO initiatives such as AP214 are also starting to recognize and improve representations for product ranges through the concept of product class [18]. These efforts have focused on product structural representations, from a functional perspective of the product, which can provide a better understanding of the product architecture. However, these provide variant data models for existing completed designs and are therefore limited in terms of the support they provide for design reuse in new product development.

To provide support for design reuse, an information model must capture the knowledge acquired over a period of time that is relevant to that product. This should include relationships between function, requirements and means [19]. The potential to capture such information is greater in the case of product ranges where the design concept is in place, but the alternative ways in which these concepts can be embodied and combined can provide new designs.

Product functions are recognized as an efficient way to drive the design process, providing intelligent information retrieval, or reuse, for high-level design support [20, 21]. However, when applied with this meaning, the inclusion of functions in a product model can make the structure of such an information model very complex [22, 23], interfering with the main purpose of this

information model. Fothergill *et al.* [24] provide additional models that can capture information reuse to support variant design activities. However, because their approach uses constraint management mechanisms based on a defined design model, it becomes overdependent on the design process.

Other information models, apart from the product model, have also been recognized as necessary to support integrated life-cycle activities, such as the manufacturing model [25]. Figure 1 depicts the general architecture of a typical data model-driven CAE system to support product design and manufacturing activities, where information is placed in a separate level from the software applications. This general concept, resulting from the MOSES research project [26], has been explored further in the MIM project [27] to which this paper is closely related. In such architectures, software applications share consistent information provided from databases. The databases each provide access to a particular type of information; i.e. a product model offers product information, a manufacturing model offers information concerning manufacturing facilities and a product range model offers information to support design reuse.

The relationships between data, information and knowledge can be confusing. This paper follows the general view of the computer science community [28], which is that data are related simply to words or numbers, the meaning of which has not been defined, information is structured data that have some meaning, and knowledge is information with added value that relates to how it may be used or applied. A product model, which supports integrated information sharing, is an information model, while a product range model combines both information and knowledge.

The product range model provides an information and knowledge repository, separate from the product model, which can be used to store the ways in which a particular product range can, and cannot, be designed. In order to guide the designer, information about product range

functions are associated with design solutions. The knowledge to support the intelligent reuse of design concepts is captured through elements referred to in this paper as interactions. The product range model aims to provide designers with appropriate information, leaving them to make the design decisions. This approach does not impose a particular design process or sequence on the designer and maintains the view of the product data model as an information structure that can be used as a source and repository for information about the product under development.

The work has adopted an object-oriented approach to modelling a product range information structure. The object-oriented database ObjectStore© has been used to realize the information structure, in combination with Visual C++© to realize functional software applications.

To explore the ideas around this product range concept, injection mould design has been chosen and is explored in the next section, followed by an outline of the basic concepts of the product range model in Section

3. The experimental system design and information structures are shown in Sections 4 and 5, followed by a discussion and conclusion section.

2 EXPLORATION OF THE INJECTION MOULD PRODUCT RANGE

Injection moulds can be considered to be a type of product range. The basic idea, or concept, behind an injection mould is well understood, although there are many variations in the detail design of a mould, depending on how a designer chooses to meet the design requirements. Besides geometric aspects of the plastic component impression, the injection mould has basic requirements such as the number of impressions and the type of mould, e.g. a two- or three-plate construction. A mould design also has more specific requirements such as those related to cooling, ejection, feeding, etc. Such requirements can be met by sets of different possible design solutions, where each set aims to satisfy the basic functions of the injection
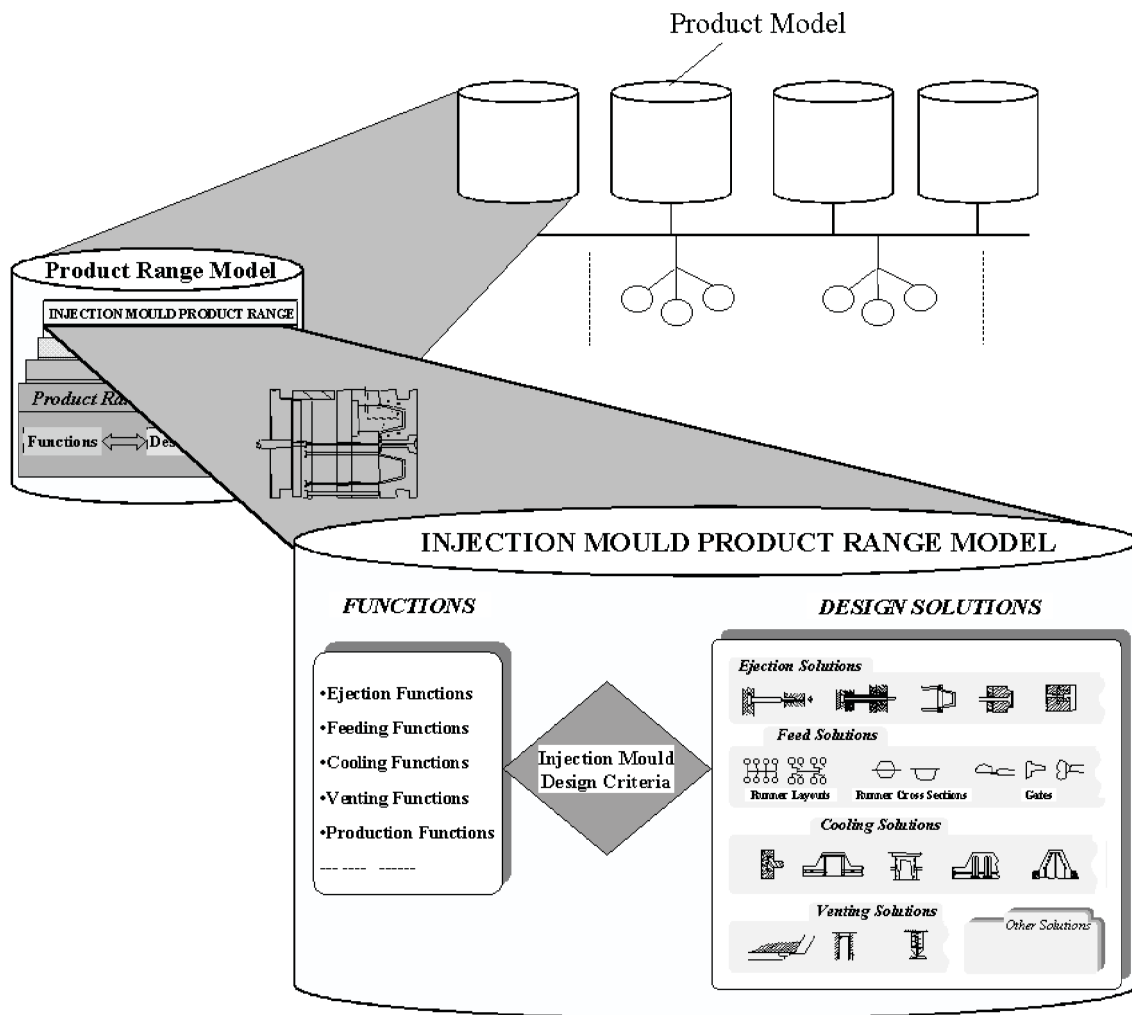

Fig. 2 Injection mould product range concept

mould product, e.g. eject product, feed impression, cool core, etc. [**29**, **30**]. Sebastian [**31**] and Catic and Raos [**32**] have stated the importance of such functions driving the injection mould design process.

Figure 2 shows a general representation of the injection mould product range model, illustrating the association between design functions and design solutions for injection moulds. Although most of the detailed design solutions and the reasons for their applications (mouldability, functionality, manufacture, constraints, etc.) are well understood, the choice of the best set of solutions is not a straightforward process, as a range of design criteria influences these choices.

Research in injection mould design systems has focused both on particular systems of the mould, e.g. feeding [**33**, **34**] and ejection [**35**], and on general design environments, where different systems and design decisions are considered together [**36–38**]. However, one of the main issues that still remain to be resolved is related to the interactions between different injection mould design decisions [**39**]. There is also a need to provide injection mould design systems with better functionality and knowledge support [**40**, **41**].

Figure 3 depicts an overview of the injection mould design process, highlighting the relevance of interactions through the decision process. In the early stages of injection mould design, different variables must be evaluated, such as injection mould specifications, in which the requirements of the plastic component are included. Decisions made at this stage, e.g. the type of mould configuration, the number of cavities or the type of runner, impose constraints on the design process and

may influence later design decisions. For example, the selection of ejection pins as an ejection technique is likely to affect the cooling techniques that can be used. The process of enabling with these interactions during the design process is essential if a final balanced design is to be achieved. This work therefore explores how the association between functions and design solutions should be captured to provide design support, and also explores the kind of mechanisms required to guide the acceptability of particular design solutions throughout the design process.

3 PRODUCT RANGE MODEL CONCEPT

The product range model (PRM) is an information model that stores not only the relationships between functions and design solutions but also the design criteria that must be fulfilled for the applicability of each design solution. These design criteria, in turn, are defined through the potential interactions of each design solution with other product information and with decisions already made by the designer.

Three main issues that influence the structure of a product range model are addressed in the following subsections. These are:

1. What are the set of functions and their design solutions?
2. How do individual design solutions interact?
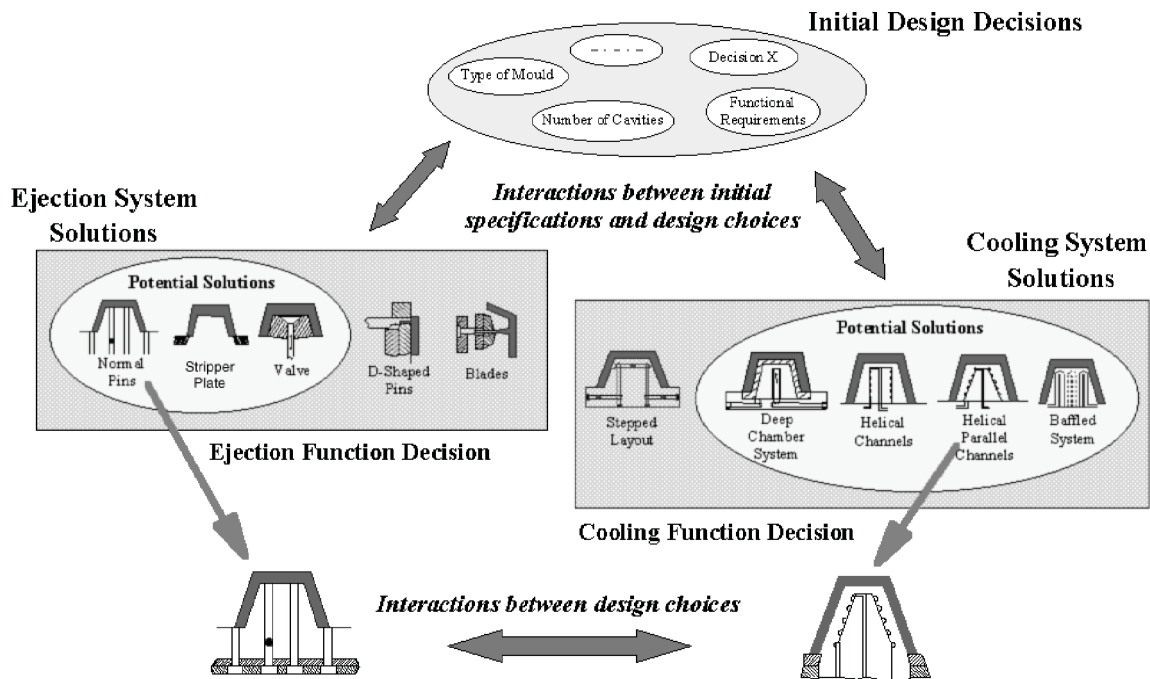3. What is the relationship between a PRM and a product model?

Fig. 3 Interactions between design choices in the injection mould design process

3.1 Function and design solutions

The modelling of functions and design solutions follows the approach that functional enquiries can be linked to sets of possible solutions that meet the design requirement [42]. The general description of injection mould functions and main systems has been addressed in the literature [29], but no work has defined their relationship explicitly. In this paper, such a relationship is defined and used as the basis for the exploration of how appropriate design criteria can be associated with each design solution and used to check the validity of particular design choices.

Figure 4 depicts the general relationship between functions and design solutions within the injection mould product range. Each function is associated with a set of all design solutions that could potentially be applied to achieve it, e.g. for ejecting the plastic component a set of possible ejection techniques is available. Each design solution, in turn, can be applied to one or more function, for instance, ejection pins can be used to eject the plastic product or the runner system. Similarly, a specific configuration of cooling technique could be used to cool the core, cavity or even the bolster of the mould.

The process of searching for valid design solutions starts by 'querying' all possible design solutions that are associated with the function chosen. The selected set of design solutions will be checked against the design conditions, i.e. specifications, requirements and other design decisions, and hence valid solutions are finally offered to the designer.

3.2 Design solution interactions

Each design solution must meet the requirements, established by its design criteria, if it is to provide a valid solution in a particular design situation. These design criteria relate to different kinds of injection moulding information, such as the number of impressions, the mould configuration, the type of feed system, the properties of the plastic component and the techniques chosen for the ejection system, runner system, gate system and cooling system. Thus, capture of the knowledge that represents these design criteria is required.

To capture this design knowledge, a set of elements, termed interaction elements, has been defined. The combination of these elements is based on the application of logic Boolean. Each design solution is associated with a set of interaction elements, which must be evaluated as true in any design case, to provide a valid design solution. In addition, a set of interaction elements must maintain relationships with other interaction element sets, providing a flexible and comprehensive knowledge representation.

This work has identified the critical information aspects of interaction elements as:
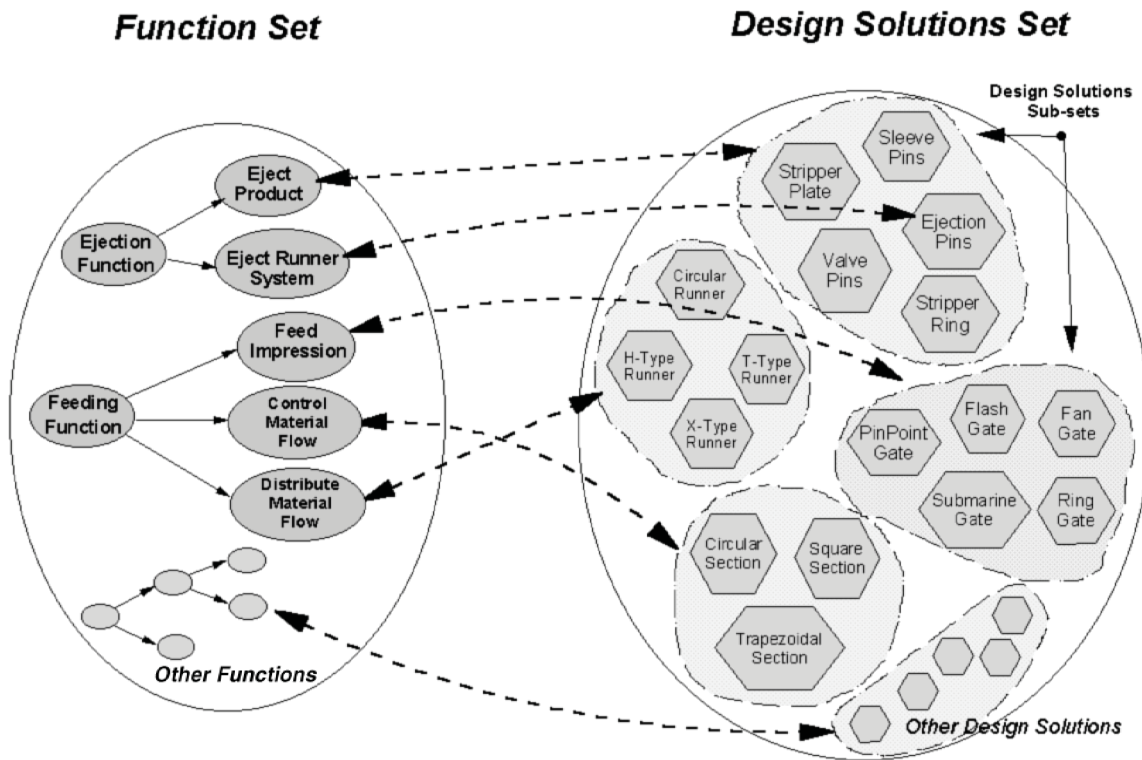

Fig. 4   Relationships between injection mould functions and design solutions
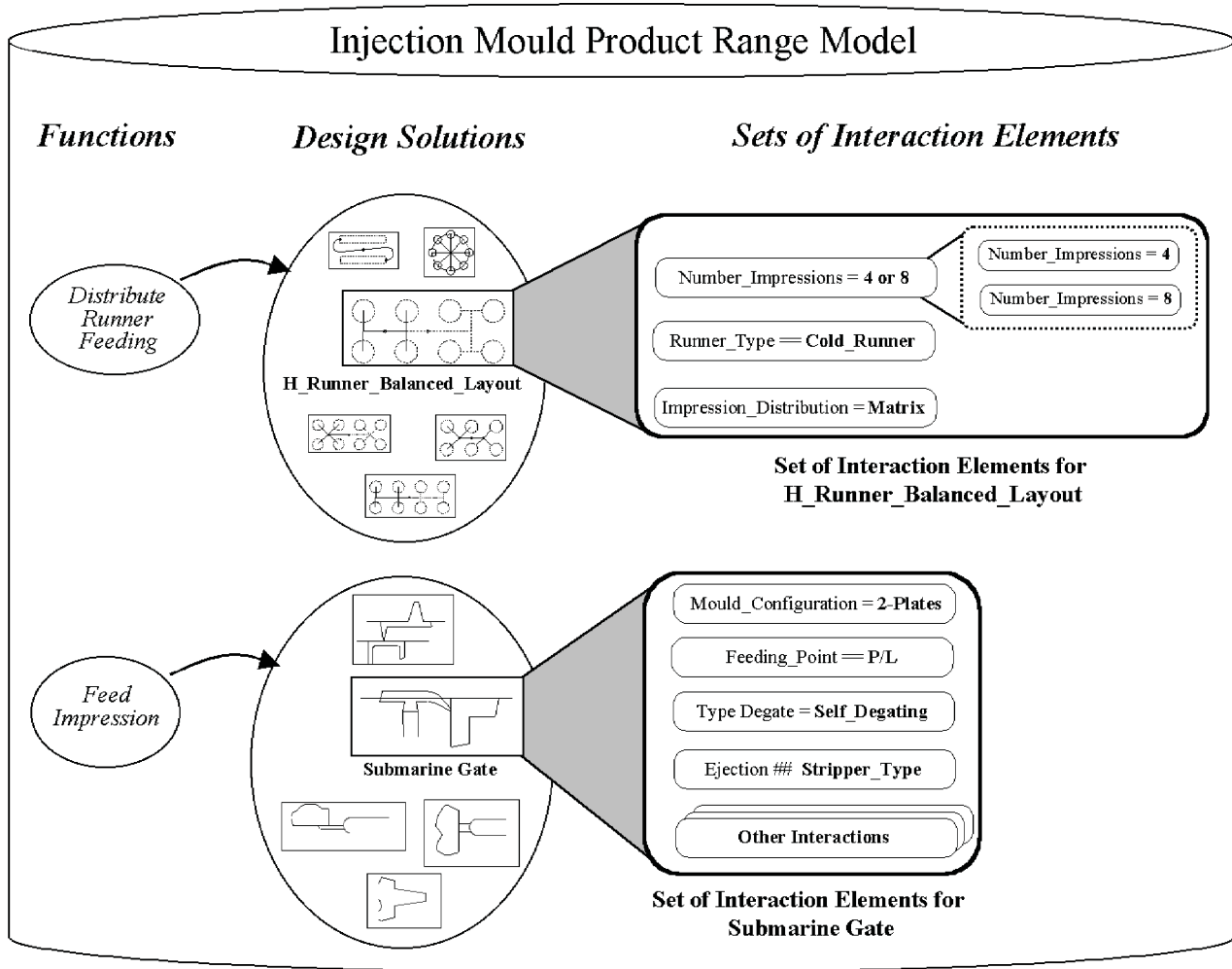
Fig. 5 Example of sets of interaction elements

1. *Information location*, which is concerned with the location of specific information that will be compared within the interaction element. In this respect, information related to the product models, injection mould and plastic component products, and the product range model, temporarily selected design solutions, must be identified. The information structure of each information model must be known in order to provide the specific path for the information retrieving process.

2. *Information type*, which is concerned with the specific kind of information that is being compared. In this respect, two main kinds of information have been identified, namely numerical, which is related to numerical attributes, and existence, which is related to the existence, or not, of particular product characteristics. This aspect is important to keep the compatibility between the information retrieved from information models and the information stored in the interaction.

3. *Information importance*, which defines the level at which an interaction can or cannot constrain a

design solution. Thus, an interaction can be defined as required, desired, etc.

The first two aspects provide the interaction element with the ability to know 'what' is being checked and 'where' to look for that piece of information. More detail of this is explained in the following section. The last aspect is not addressed in this paper.

Figure 5 depicts an example of a set of interaction elements associated with two design solutions, i.e. *H_Runner_Balanced_Layout* and *Submarine_Gate.* To be valid, these design solutions must have all of their interaction elements evaluated and approved. Different types of information are compared through the interaction elements presented. The number of impressions, the mould configuration and the type of runner system are examples of numerical comparisons, while the type of impression distribution and the ejection technique are examples of existence interactions. In the *H_Runner_Balanced_Layout* example, the interaction element related to the number of impressions is composed of two other alternative interactions, namely *Number_*

of_Impressions = 4 or *Number_of_Impressions = 8*. In this case the approval of either alternative interaction will result in the approval of the 'parent' interaction.

Each interaction element, after the checking process, can assume three possible states: approved, i.e. the condition is true; reproved, i.e. the condition is false; or not evaluated, i.e. the information required for the comparison process was not available in the product model. To be considered valid, a design solution must have all associated interactions approved, or eventually non-evaluated, otherwise it is considered rejected for the particular design situation.

To evaluate each interaction element, the appropriate information must be retrieved from the product model and compared with the interaction condition. This requires that relationships between the information models be defined in terms of information structures as described in the next section.

3.3 INFORMATION MODEL RELATIONSHIPS

The product range model needs to access product information in order to check the validity of potential design solutions in any particular design situation. Relationships to the product data model have been defined through the use of knowledge links, which enable the retrieval of product information. Knowledge links are associated with each interaction element in the PRM and identify where in the product model the required information can be found.

Figure 6 provides an example of checking interactions through knowledge link elements. For each interaction leading to a design solution, a specific attribute is searched and retrieved from an information model and then compared with its reference value. An address will be associated with the data model path required to achieve a specific attribute of an object, or the object itself, in the information models. The information retrieved is then compared with the reference value contained within the interaction element.

In addition to the product range model, two information models are depicted in Fig. 6: the product model, where final design decisions are stored, and a temporary product information model, where temporary design decisions are kept. Information is initially searched and retrieved from the product model and, if no value is met, a search in the temporary product information is performed.

If the result of the information requested by the knowledge link is not found, the checking interaction process cannot be performed and the interaction will assume a 'not evaluated' status. In this situation, if the design solution that holds such an interaction element is selected, the designer will be informed that there is a pending interaction that must be resolved at some stage in the design process.

4 DESIGN OF INFORMATION STRUCTURES AND A USE CASE MODEL

The analysis and design of the information reuse system were carried out using UML notation [43], following the use case process suggested by Texel and Willens [44]. The application of such a notation allows the analysis and design of the system in terms of basic functionality, to provide injection mould design support, and therefore allows the identification of the required information structures (object attributes and relationships) and applications (object methods) that will be part of the information reuse system.

Figure 7 depicts a general representation of the injection moulding product model structure, within which information on two kinds of product, an injection mould and a plastic component, can be stored. The plastic component class enables the storage of information that will be part of the injection mould specification, and the injection mould class maintains relationships with the mould plate and injection mould system (ejection, feeding, cooling, etc.) classes. Each injection mould system instance, in turn, can be composed of one or more solution technique instances, which will be associated with one of the injection mould plates. The solution techniques is the class that will receive the information selected from the product range model, and for this reason compatibility between the PRM structure and the injection mould product model structure is required. In addition to receiving all final design decisions through specific software applications and the designer, the injection mould product model also provides information that will drive the product range model in offering valid sets of design solutions.

Figure 8 shows a UML class diagram of the product range model structure. Using this structure, each specific function instance, e.g. eject impression, cool core, etc., can be associated with one or more instances of the design solution subclasses, i.e. cooling design solutions, ejection design solutions, etc. The design solution classes at the lowest level of the structure are compatible with the product data model solution techniques, allowing instances of any chosen PRM design solutions to be stored in the product model.

While the design solution class has relationships to the manufacturing option and design solution characteristic classes, this paper focuses on the relationship to the interactions class. The set of interaction instances defines the design criteria to be applied to each design solution instance. Two kinds of interaction have been addressed in Section 3.2 and are represented by the simple interaction and composite interaction classes. The simple interaction class has an association with the knowledge link class, which provides the mechanisms for retrieving information from the product model. Composite interaction objects will, eventually, break down to simple elements.
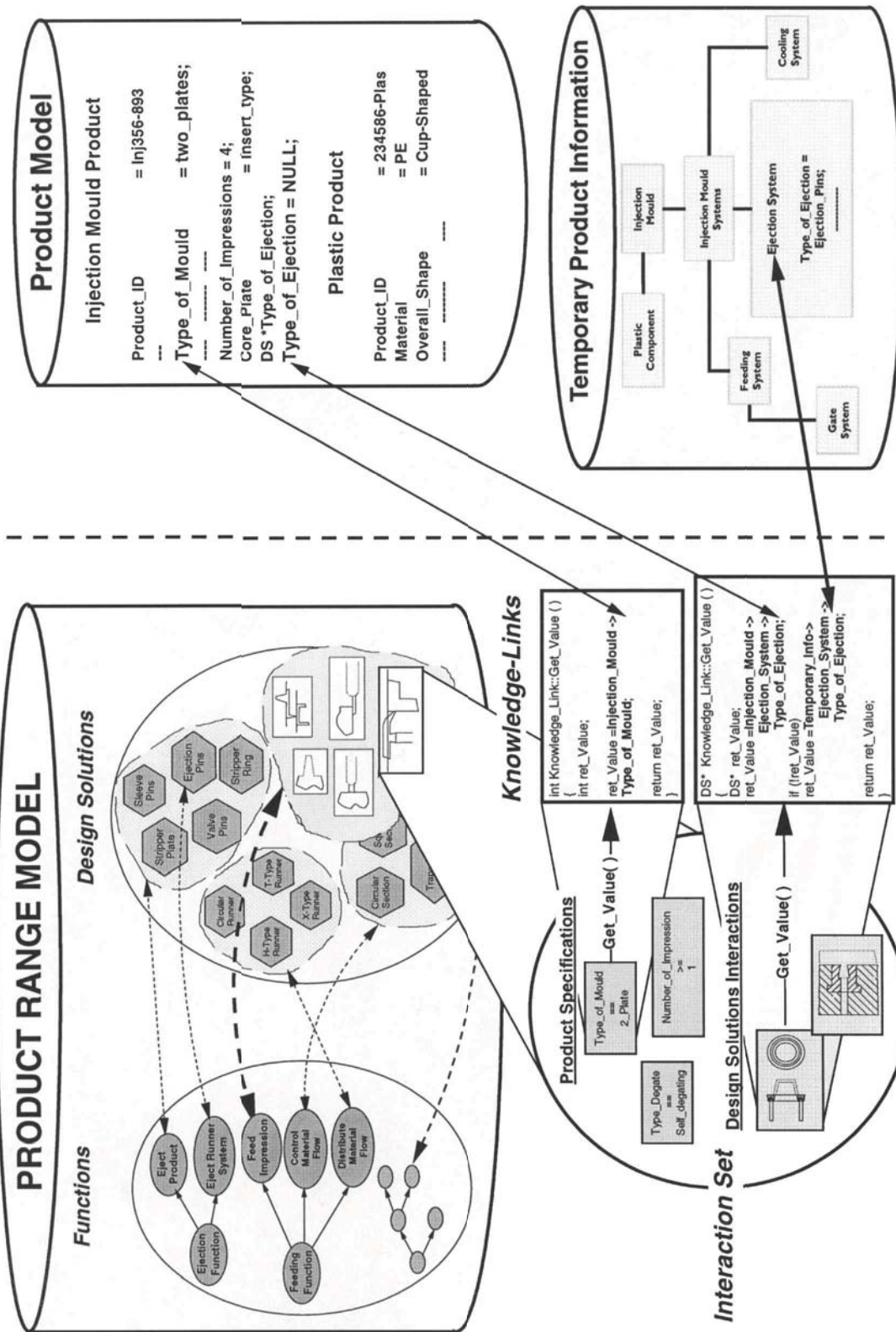
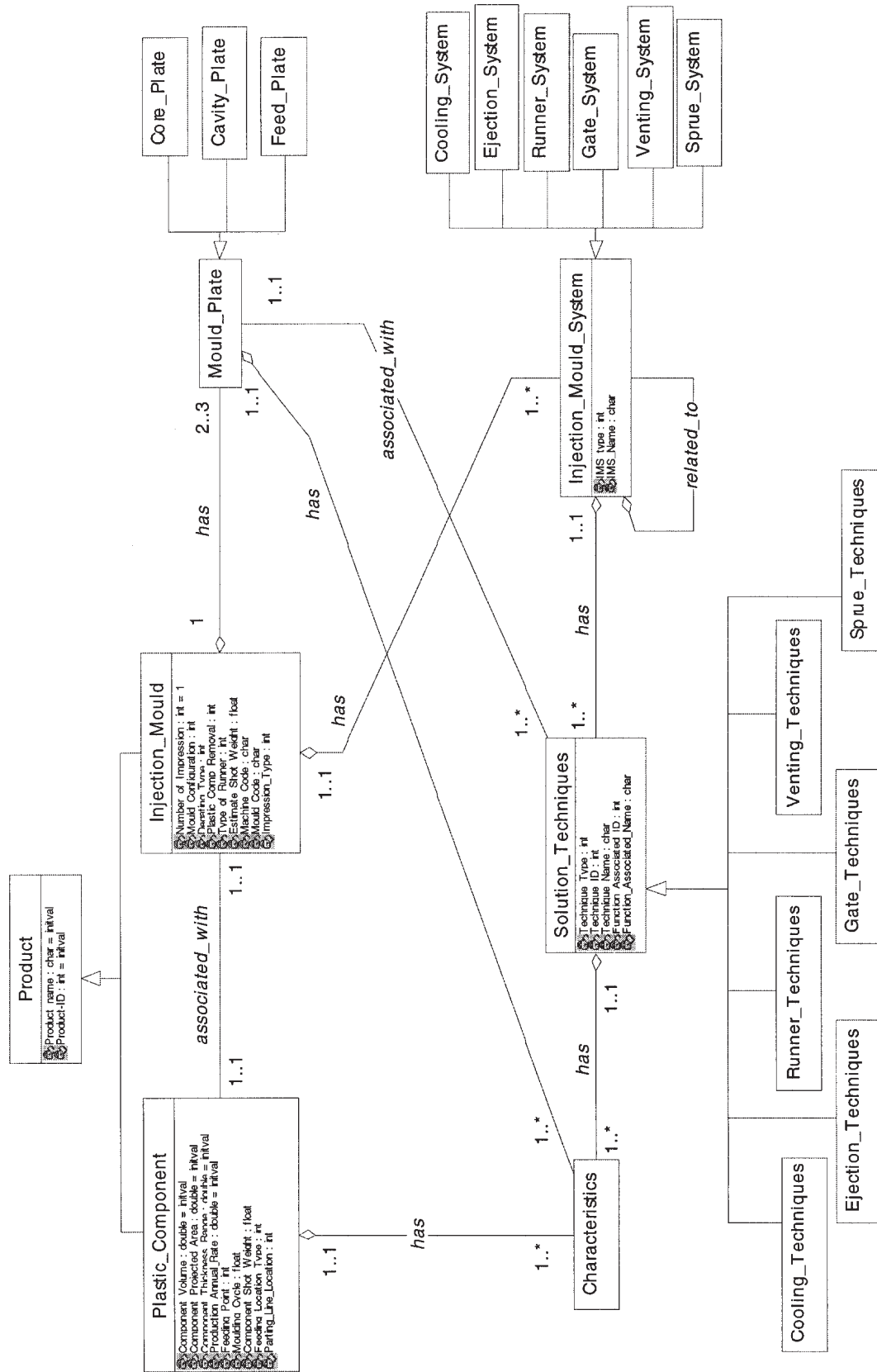**Fig. 6** Example of information location through knowledge links

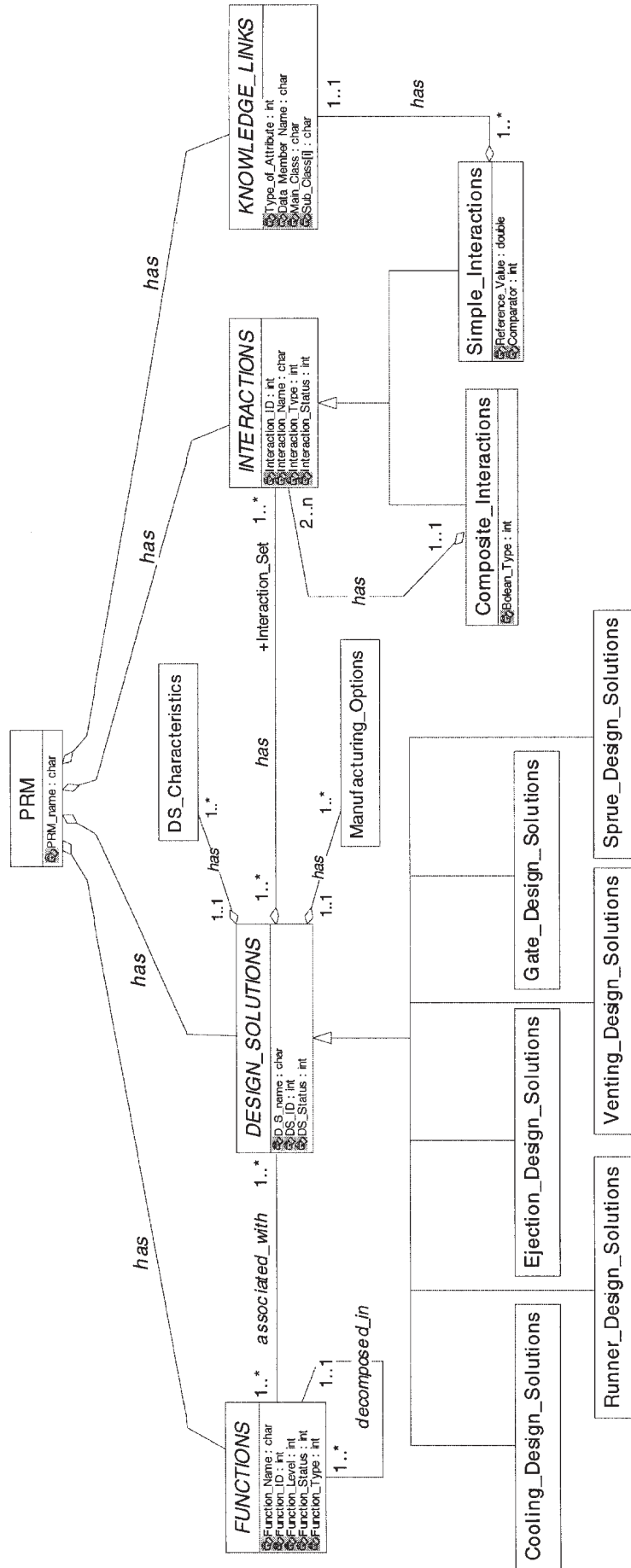**Fig. 7** UML class representation of injection moulding product model structure

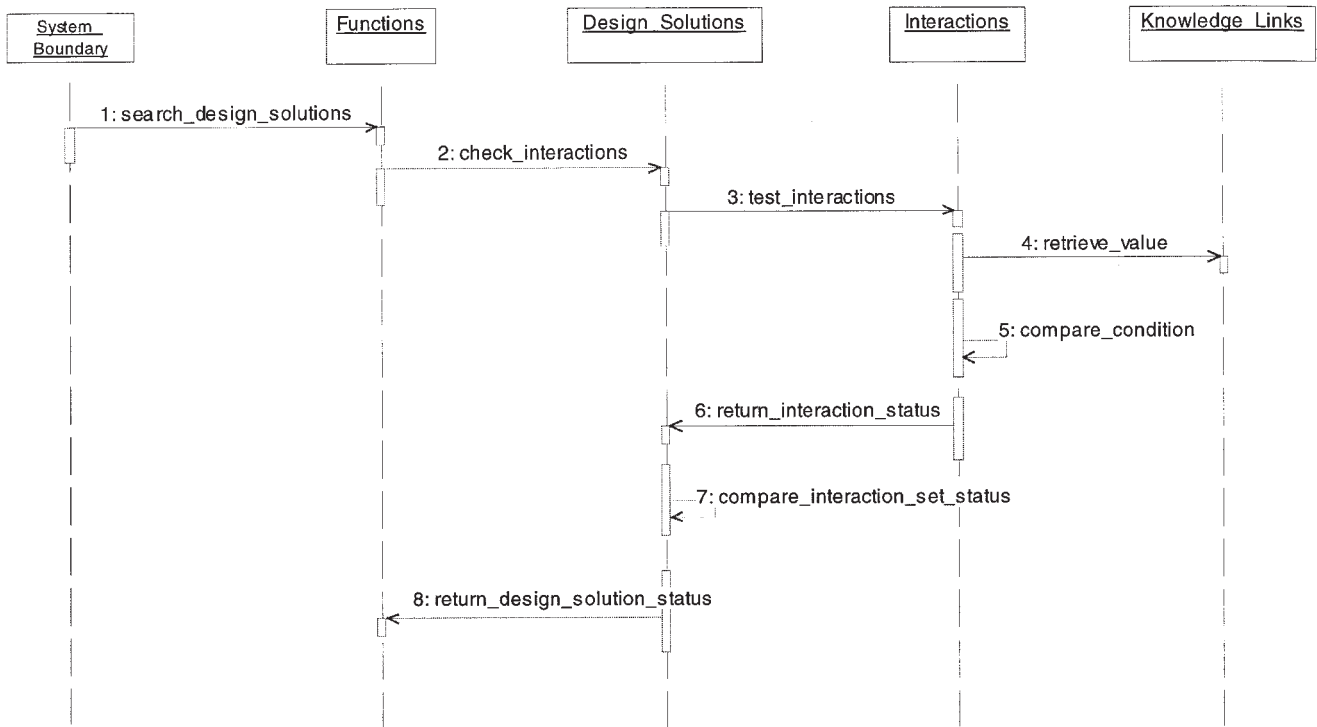**Fig. 8** UML class representation of product range model structure

Fig. 9 UML sequence diagram highlighting class functional relationships

Figure 9 shows a UML sequence diagram highlighting the basic functionality designed within the information reuse system. The main methods called during the process of checking for potential design solutions are depicted. After being selected by the user, a specific function searches for all possible design solutions that can be applied for that specific design situation. For each design solution instance associated with the function selected, the set of interactions associated with it is checked. Each interaction element is tested in turn by comparing the values retrieved by the knowledge links against its particular constraints. Based on the comparison process, a status is assumed by the interaction object, which is returned to its related design solution. The set of interaction states associated with each particular design solution will define the resulting status of the design solution instance, which is returned to the function object under consideration.

5 EXPERIMENTAL DESIGN REUSE SYSTEM

The computational implementation of the design reuse system has used the ObjectStore© database together with the programming environment Visual C++©. While the former has been utilized to create the information structures required, the latter allows the implementation of the system functionality and interfaces with the end user. A range of injection mould functions, design solutions and interactions has been implemented. This section focuses on a particular example to illustrate the ideas presented in the paper. The example used concerns the selection of a gating system, highlighting how interactions are identified and handled by the system.

To achieve the function *Feed Impression*, different gate design solutions must be evaluated against product design information such as the mould configuration, the number of cavities, the feeding point, the kind of degating of the plastic component and the ejection system chosen. Figure 10 depicts an example of the results of a functional enquiry on *Feed Impression*, resulting in sets of accepted and rejected gate design solutions. The interactions associated with the accepted design solution *Tunnel/Submarine Gate* are highlighted. Three interactions have been completely approved, i.e. *Impression >= 1*, *Runner Type = cold* and *Mould Configuration = 2-Plates*, which can be confirmed by the basic product specifications displayed in the right-hand dialogue. The other interactions have been identified as not evaluated, which means that no information could be retrieved from the product model or the temporary information model, to be compared with the interaction reference values.

The not evaluated interactions do not reject the *Tunnel/Submarine Gate* design solution as a possible option to fulfil the *Feed Impression* function. However, in the case of selection of this gate solution, these not evaluated interactions need to be properly resolved, and hence they are displayed to the designer as pending interactions. Figure 11 shows that, after being selected as a temporary design solution, the *Tunnel/Submarine*
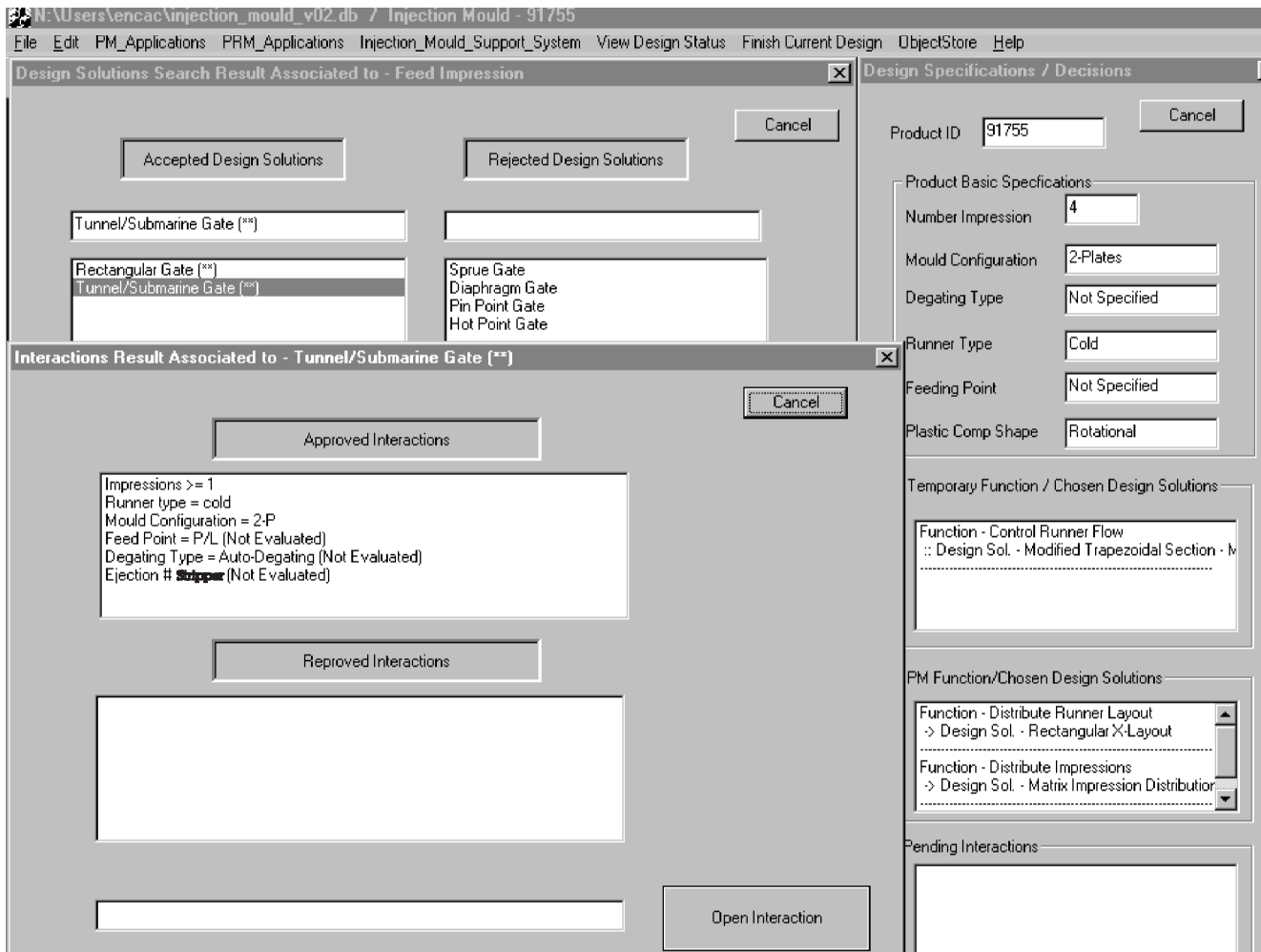
Fig. 10 Example windows of product and PRM interaction instances

Gate requires that stripper ejection techniques be avoided to eject the plastic component. Another two pending interactions are also displayed in Fig. 11, emphasizing some interactions related to the injection moulding product specifications. In this case, if these specifications had been detailed at the beginning of the design process, the set of accepted design solutions would be more restricted. However, their absence does not interfere in providing valid design solution options, which offers relative flexibility to the designer in terms of choosing valid options.

6   CONCLUSION

The use of information model structures to support design reuse has been discussed and a new type of information model proposed, called a product range model, to support the reuse of design information in variant design cases. The general structure of a product range model has been defined in terms of design functions linked with sets of design solutions, interactions between potential solutions and knowledge links which capture

relationships between potential solutions and the constraints that are to be applied to their use.

The integration of a product range model into a general product model-based information environment requires a close relationship between the low-level data structures captured in the product range model and the low-level data structures defined in a product data model. It is therefore important that data models to support such information systems are developed and defined in parallel and not considered as totally separate information sources.

The approach taken allows each design solution to be an individual object that holds its own knowledge and defined relationships with other product information. This concept allows the evolution of a heuristic acquired through time and hence has the capability to be maintained and updated as new design ideas become accepted. Further work is required to investigate this capability. Further work is also needed to investigate the product range model relationship to manufacturing alternatives and to explore the applicability of the approach to the generation of new concept designs that go beyond adaptive or variant design cases.
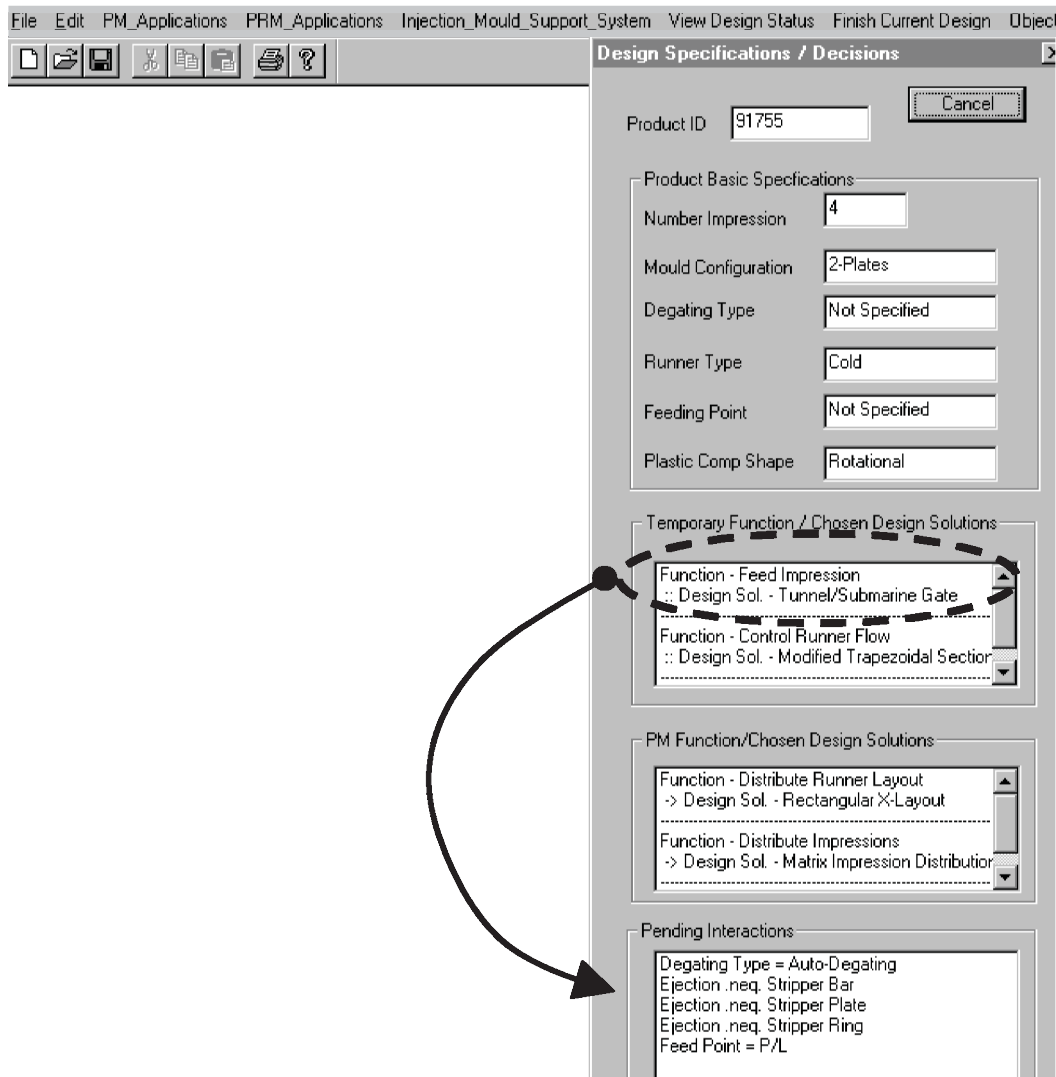
Fig. 11 Support for an incomplete design

This work has shown that the product range model concept can offer companies the ability to reuse design information within integrated systems, providing a more efficient environment to support the design team decision-making process. The ability for design reuse to fit within an integrated information environment is particularly important if future decision support systems are to offer team-based aids when working on new product development.

ACKNOWLEDGEMENTS

REFERENCES

1 **Sivaloganathan, S.** and **Shahin, T. M. M.** Design reuse: an overview. *Proc. Instn Mech. Engrs, Part B, Journal of Engineering Manufacture*, 1999, **213**(B7), 641–654.

2 **Krause, F.-L., Kimura, F., Kjelberg, T.** and **Lu, S. C.-Y.** Product modelling. *Ann. CIRP*, 1993, **42**(2), 695–705.

3 **Jo, H., Parsaei, H.** and **Sullivan, W.** Principles of concurrent engineering. In *Concurrent Engineering: Contemporary Issues and Modern Design Tools* (Eds H. R. Parsaei and W. G. Sullivan), 1993, pp. 3–23 (Chapman and Hall, Cambridge).

4 Designers do the knowledge. *The Engineer*, February 2000, 17–19.

5 **Duffy, A. H. B., Smith, J. S.** and **Duffy, S. M.** Design reuse research: a computational perspective. In *Engineering Design Conference '98—Design Reuse*, Brunel, 1998, pp. 43–56 (Professional Engineering Publishing Limited).

6 **Maher, M. L.** and **Gomez de Silva Garza, A.** Case-based reasoning in design. *IEEE Expert*, March–April 1997, 34–41.

7 Watson, I. and **Perera, S.** Case-based design: a review and analysis of building design applications. *Artif. Intell. for Engng Des.*, *Analysis and Mfg*, 1997, **11**, 59–87.

8 Grabowski, H., Lossack, R.-S. and **Weis, C.** Supporting the design process by an integrated knowledge based design system. In *Advances in Formal Design Methods for CAD* (Ed. J. Gero), 1996, pp. 209–229 (Chapman and Hall, London).

9 Dixon, J. R. Knowledge-based systems for design. *Trans. ASME*, 1995, **117**, 11–16.

10 Lee, R. J. V. and **Young, R. I. M.** Information supported design and manufacture of injection-moulded rotational products. *Int. J. Prod. Res.*, 1998, **36**(12), 3347–3366.

11 Anderl, R. Trends in product modelling. In 11th International Conference on *Engineering Design—ICED'97*, Tampere, Finland, August 1997, pp. 113–120 (Tampere University of Technology).

12 Gu, P. and **Chan, K.** Product modelling using STEP. *Computer-Aided Des.*, 1995, **27**(3), 163–179.

13 Ashworth, M., Bloor, M. S., McKay, A. and **Owen, J.** Adopting STEP for in-service configuration control. *Computers in Industry*, 1996, (31), 235–253.

14 McKay, A., Bloor, M. S. and **de Pennington, A.** A framework for product data. *IEEE Trans. Knowledge and Data Engng*, 1996, **8**(5), 825–837.

15 McKay, A., Erens, F. and **Bloor, M. S.** Relating product definition and product variety. *Res. Engng Des.*, 1996, **2**, 63–80.

16 Erens, F. and **Verhulst, K.** Architectures for product families. *Computers in Industry*, 1997, **33**, 165–178.

17 Gorti, S. R., Gupta, A., Kim, G. J., Sriram, R. D. and **Wong, A.** An object-oriented representation for product and design process. *Computer-Aided Des.*, 1998, **30**(7), 489–501.

18 Mannisto, T., Peltonen, H., Martio, A. and **Sulonen, R.** Modelling generic product structures in STEP. *Computer-Aided Des.*, 1998, **30**(14), 1111–1118.

19 Shahin, T. M. M., Andrews, P. T. J. and **Sivaloganathan, S.** A design reuse system. *Proc. Instn Mech. Engrs, Part B, Journal of Engineering Manufacture*, 1999, **213**(B6), 621–627.

20 Beng, T., Britton, G., Chandrashekar, M. and **Wee, N.** Functional design. In *Integrated Product and Process Development: Methods, Tools, and Technologies* (Eds J. Urher, U. Roy and H. Parsaei), 1998, pp. 29–58 (John Wiley and Sons, New York).

21 Al Hamando, M. M. S. and **Kumura, S. R. T.** Models of conceptual design for concurrent engineering. In *Intelligent Systems in Design and Manufacturing* (Eds C. H. Dagli and A. Kusiak), 1994, pp. 61–88 (ASME Press, New York).

22 Baxter, J. E., Juster, N. P. and **de Pennington, A.** A functional data model for assemblies used to verify product design specifications. *Proc. Instn Mech. Engrs, Part B, Journal of Engineering Manufacture*, 1994, **208**(B4), 235–244.

23 Henderson, M. R. Representing functionality and design intent in product models. In Proceedings of 2nd Symposium on *Solid Modeling and Applications*, Montreal, 1993, pp. 387–396 (ACM).

24 Fothergill, P., Arana, I. and **Forster, J.** Constraint management and design models in supporting re-design. In 6th International Conference on *Flexible Automation and Intelligent Manufacturing*, Atlanta, May 1996, pp. 363–372 (Begell House Inc.).

25 Molina, A., Ellis, T. I. A., Young, R. I. M. and **Bell, R.** Modelling manufacturing capability to support concurrent engineering. *Concurrent Engng: Res. Applic.*, 1995, **3**(1), 29–42.

26 Ellis, T. I. A., Molina, A., Young, R. I. M. and **Bell, R.** An information sharing platform for concurrent engineering. In *Integrated Manufacturing Systems Engineering* (Eds P. Ladet and F. Vernadat), 1995, pp. 262–275 (Chapman and Hall).

27 Young, R., Canciglieri Jr, O. and **Costa, C.** Manufacturing information interactions in data model driven design. *Proc. Instn Mech. Engrs, Part B, Journal of Engineering Manufacture*, 1999, **213**(B5), 527–532.

28 Harding, J. A. A knowledge representation model to support concurrent engineering team working. PhD thesis, Department of Manufacturing Engineering, Loughborough University, Loughborough, 1996.

29 Rosato, D. V. and **Rosato, D. V.** *Injection Molding Handbook: the Complete Molding Operation Technology, Performance, Economics*, 1995 (Chapman and Hall, New York).

30 Menges, G. and **Mohren, P.** *How to Make Injection Molds*, 2nd edition, 1993 (Carl Hanser Verlag).

31 Sebastian, D. H. Function based design for injection molding. In ANTEC'93, 1993, pp. 1114–1119 (SPE).

32 Catic, I. and **Raos, P.** Theoretical approach to injection mould design using partial functions and a morphological matrix. *Plastics and Rubber Processing and Applic.*, 1989, **11**(3), 151–157.

33 Ong, S. K., Prombanpong, S. and **Lee, K. S.** An object-oriented approach to computer-aided design of a plastic injection mould. *J. Intell. Mfg*, 1995, **6**, 1–10.

34 Irani, R. K., Kim, B. H. and **Dixon, J. R.** Towards automated design of feed system of injection molds by integrating CAE, iterative redesign and features. *Trans. ASME*, February 1995, **117**, 72–77.

35 Wang, Z., Lee, K. S., Fuh, J. Y. H., Li, Z., Zhang, Y. F., Nee, A. Y. C. and **Yang, D. C. H.** Optimum ejector system design for plastic injection moulds. *Int. J. Computer Applic. Technol.*, 1996, **9**(4), 211–218.

36 Mok, C. K., Chin, K. S. and **Cheung, E. H. M.** A knowledge-based system for plastic injection mold design. In 3rd International Conference on *Automation, Robotics and Computer Vision (ICARV'94)*, Singapore, November 1994, WP8.4, pp. 353–358.

37 Kruth, J. P. and **Willems, R.** Intelligent support system for the design of injection moulds. *J. Engng Des.*, 1994, **5**(4), 339–351.

38 Chin, K.-S. and **Wong, T. N.** Knowledge-based evaluation for the conceptual design development of injection molding parts. *Engng Applic. Artif. Intell.*, 1996, **9**(4), 359–376.

39 Lee, R.-S., Chen, Y.-M. and **Lee, C.-Z.** Development of a concurrent mold design system: a knowledge-based approach. *Computer-Integrated Mfg Syst.*, 1997, **10**(4), 287–307.

40 Kruth, J. P., Willems, R. and **Lecluse, D.** A design support system using high level mould objects. In International Conference and Exhibition on *Design and Production of Dies and Molds*, Turkey, 1997, pp. 39–44 (CIRP).

41 Lee, K. S., Li, Z., Fuh, J. Y. H., Zhang, Y. F. and **Nee, A. Y. C.** Knowledge-based injection mold design system. In International Conference and Exhibition on *Design and Production of Dies and Molds*, Turkey, 1997, pp. 45–50 (CIRP).

42 Suh, N. P. Axiomatic design of mechanical systems. *Trans. ASME*, 1995, **117**, 2–10.

43 Booch, G., Rumbaugh, J. and **Jacobson, I.** *The Unified Modeling Language User Guide*, 1999 (Addison-Wesley Longman).

44 Texel, P. and **Williams, C.** *Use Cases Combined with BOOCH/OMT/UML: Process and Products*, 1997 (Prentice-Hall).