Strathprints Institutional Repository

http://strathprints.strath.ac.uk/

Learning for Design Reuse

Alex H B Duffy and Sandra M Duffy

CAD Centre

Dept. of Design, Manufacture & Engineering Management

University of Strathclyde

75 Montrose Street
Glasgow G1 1XJ
Scotland
Tel: +44 141 552 4400
Fax: +44 141 552 3148
Email: alex@cad.strath.ac.uk

Design Assistance

Over the past decade 'design assistance', i.e. where the computer is viewed as an *Intelligent Design Assistant* (IDA) [MacCallum-etal85], has emerged in knowledge based design support and has formed the basic research strategy for the CAD Centre, University of Strathclyde, since the mid-80s. Within this philosophy, an IDA would act as a colleague to a designer, providing guidance, learning from past design experiences, carrying out semi and fully automated tasks, explaining its reasoning and in essence complementing the designer's own natural skills, and thus leaving the ultimate decision making, control and responsibility with the designer (Figure 1).

Designer:
- Control
- Define
- Direct
- Inquire
- Judge
- Question
- Specify

"Intelligent interface"

IDA:
- Adapt
- Calculate
- Evolve
- Explain
- Guide
- Learn
- Model
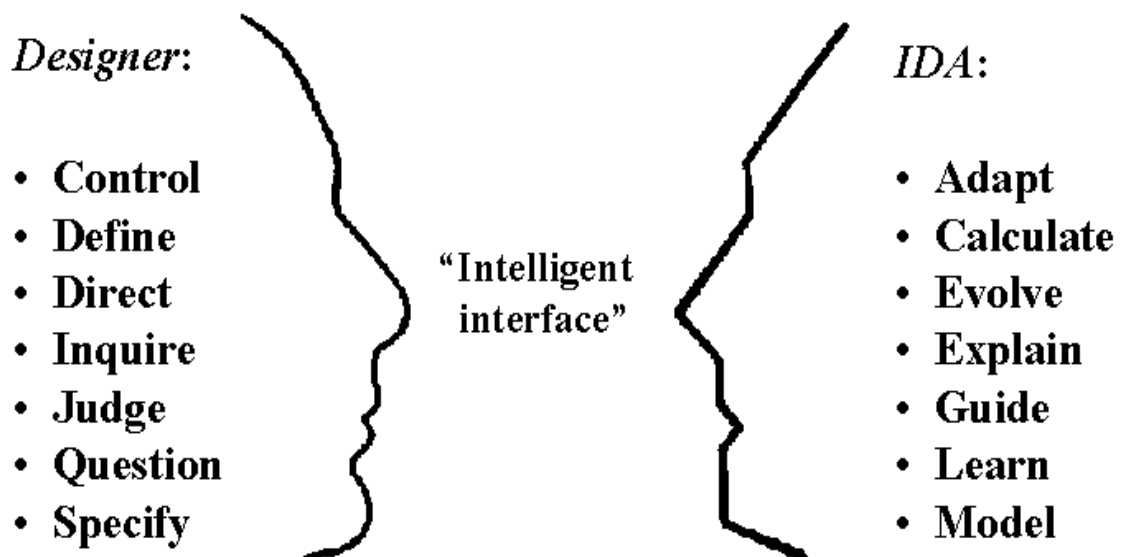
Figure 1: Intelligent Design Assistant

The ability to learn and evolve has been recognised as one of the key components of an IDA in order for it to fully support the designer's activities. Consequently, we have been directing our research effort on two main fronts, formalising our understanding and developing models of learning and reuse in design, and building appropriate software tools to step towards the concept of IDA.

Modelling learning and reuse in design

Learning in design

IDA may be achievable through the development of improved computational models of design. To build appropriate computational models we first of all need to understand what is learning in design. We first attempted this by asking three basic questions [Persidis and Duffy89]: how does learning occur, what knowledge is learned, and when does learning occur ?

How ?
To understand learning it is useful to examine what constitutes a learning event. We consider that learning occurs in three basic ways:

- Acquisition: the process of receiving new knowledge.
- Modification: the process of altering existing knowledge.
- Transformation/Generation: the process of creating new knowledge from existing knowledge.

What ?
There are no clear boundaries to the design knowledge we learn, however there are main areas. For example, we learn of: the **environment** in which the design solution must operate and fulfill its functional requirements; the **design/artefact,** which represents a description of the total solution, including necessary life cycle and life phase systems; the **design activity** and its **management** ; and the **domain** in which the design activity/solution belongs.

When ?
Designers learn when they encounter knowledge which is sufficiently different from their present state of knowledge. Learning is a perpetual process which occurs both during and outwith the design activity. It can be actively sought or passively achieved and can occur at any time or any place. The *Design/ Learning Loop* [Duffy and Duffy96] (Figure 2) illustrates how the activities of design and learning are coupled. The lower loop links design and learning activities from the solution to/ from experiential knowledge reflecting the interactive nature of design and learning. That is, where the designer, at various stages of the design process, develops a design solution, learns from that solution and activity, feeds such learned knowledge back to some store of experiential knowledge and reuses this knowledge to aid in the evolution to an acceptable design solution. During design, some of the learned knowledge will transform to longer term experiential knowledge and some only used to help the design process progress. Thus, the experiential knowledge reflects that knowledge which will be reused in new design scenarios whereas the *transient knowledge* will only be used to assist in problem solving and the evolution of the design to a final conclusion.

learning

design   *transient knowledge*   **learning**
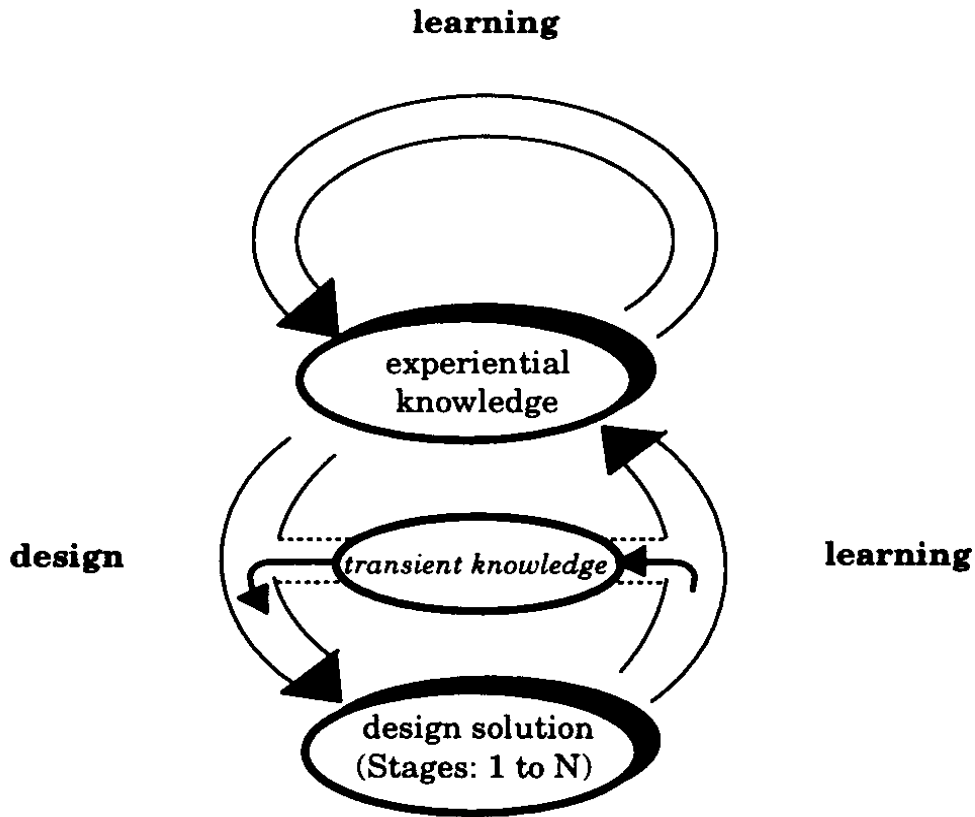
Figure 2: Design/Learning Loop

Some of the more obvious occasions when learning takes place are when evaluating or analysing the design solution; when recovering from errors or mistakes; when asking for advice or actively participating in problem solving; when exploring the domain for appropriate solutions, answers, or new knowledge; and at the completion of the design.

Design reuse

In order to effectively use learned knowledge we have developed the first *Design Reuse Process Model*   [Duffy-etal95a] in engineering design as illustrated in Figure 3.
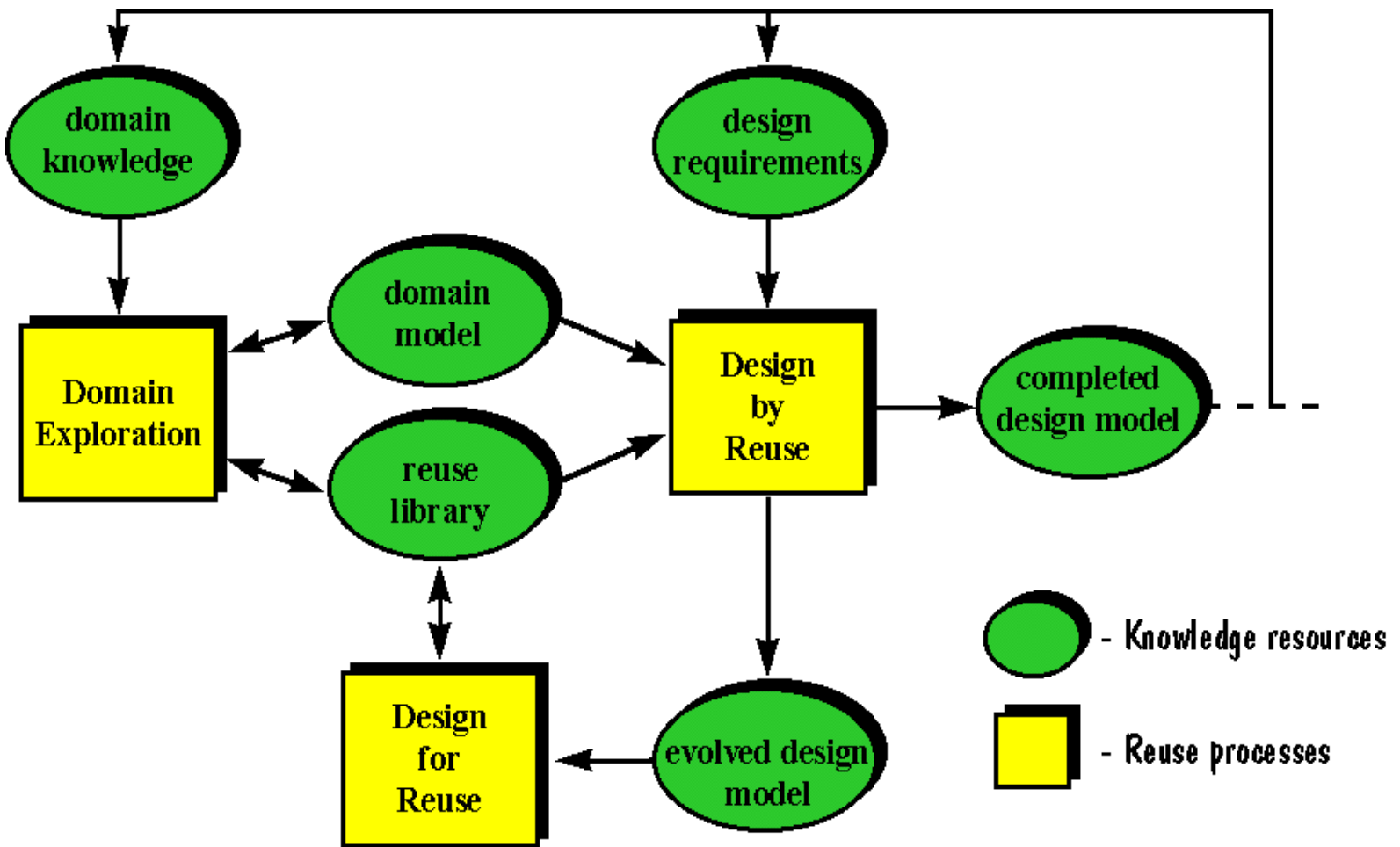
Figure 3: Engineering Design Reuse Process Model

The knowledge components of this diagram are:

- *Design requirements*   : refers to a statement of a
  design need/desire.
- *Domain knowledge*   : knowledge pertaining to a particular
  design domain.
- *Reuse library*   : a storage location holding reusable (compiled)
  design knowledge, information and data.
- *Domain model*   : represents a designer's conceptualisation of a design
  domain which is applicable to the current design problem.
- *Evolved design model*   : a statement of an
  evolved design.
- *Completed design model*   : a statement of
  a fully defined new design which is believed to satisfy the design requirements.

and the processes are:

- *Design by Reuse*   : the process in which knowledge resources
  are searched so that useful knowledge can be identified, retrieved and applied to the new design
  problem.
- *Domain Exploration*   : the process of searching,
  understanding, generalising, and in general `conceptualising' the domain in order to gain an

understanding of the features of that design domain from which reusable fragments of knowledge can be identified, extracted and stored for subsequent use in design.

- *Design for Reuse*   : a process carried out to record knowledge which is generated during design for subsequent reuse.

Software tools

NODES - learns from past designs and supports the creation of new design solutions.

NODES [Persidis89] is a Numerical and Object based DESign system developed to support modelling operations during the early stages of design and provides assistance in the building, manipulation and analysis of a model of the design artefact [Duffy-etal95b]. To fulfill this role, NODES uses its knowledge of a domain which it obtains by accumulating solutions of problems defined within that domain. Thus, although to begin with, the system allows the representation and analysis of designs, it soon acquires, modifies and generates enough knowledge to be able to actively assist with the creation of new design solutions.

The system maintains a designer initiated *reuse library*   by allowing newly evolved concepts (models) to be stored in the library and for the previously stored concepts to be updated as a result of the new concept. That is, the reuse library consists of a set of classification hierarchies (*concept libraries*   ) of commonly used domain concepts; for example, in the domain of car design, commonly used concepts might be the chassis, engine, gear-box, etc. However, a key feature of these hierarchies is that the sub-classes do not inherit their properties from their super-class, as in other typical class hierarchies, but generalise from their sub-class or past design instances/ examples "up" the concept hierarchy. Thus, concept libraries represent the knowledge NODES has acquired of some domain and become relevant when:

- they are being created by the designer - the designer defines a taxonomic hierarchy of similar concepts, where the more specific concepts are subclasses of the more abstract ones.
- a completed design is added to them - when a design is completed, each concept, together with its associated knowledge, is automatically added as a member of its corresponding class.
- the knowledge they contain is used to decompose some model - in designing a *car*   for example, the knowledge which is associated with the class *sports-xx*   in the library *cars*   , can be used to decompose or synthesise the model of the new car design.

When a design has been completed, the evolved model in NODES is used to expand its knowledge by acquiring the knowledge concerned with that new design. In the acquisition stage, the system automatically "breaks down" the design into its constituent concepts, along with appropriate constituent and connective relations, and merges each concept with its corresponding library, that is, *Design for Reuse*   . Alternatively, the designer can select particular components of the design to add to a respective library.

NODES also generalises (`abstracts') knowledge from the most comprehensive concepts within a concept library to the less specific. Numerical parameter ranges and compositional knowledge are generalised to all associated super-classes to ensure that there is no contradiction between a particular concept and its specialisations. This knowledge can then be used in a process of *Design by Reuse*   to not only provide guidance to a designer but also to assemble (synthesise) new concepts in configuration design.

PERSPECT - supports Customised Viewpoints and Shared Learning

This system couples learning and design by supporting *Domain Exploration* and *Design by Reuse* . It is based on the belief that a) within the abundant explicit information of individual past designs, there exists a wealth of implicit knowledge which should be made explicitly available to the designer and b) that designers require different viewpoints from past designs and abstractions in order to facilitate the effective utilisation of past design knowledge [Duffy and Kerr93]. It highlights the need for a dynamic design tool capable of automating the rationalisation of past designs to suit a designer's particular needs [Kerr and Duffy92]. Current approaches to viewpoint support enforces the knowledge engineer's perspectives, do not support the automatic generation of customised design viewpoints, and consequently do not adequately meet the needs of designers. This is partly realised by PERSPECT [Kerr93] which applies clustering and statistical approaches to rationalise past design examples into abstract groups and supports the application of the generated knowledge to develop a new design solution. Consequently, PERSPECT supports preliminary numerical design and is aimed at supporting the effective utilisation of numerical experiential knowledge in design.

PERSPECT's functionality has been used to explore the possibility of realising `learning' assistance in IDA by introducing a new concept called *Shared Learning* .
Shared Learning is proposed to empower CAD tools with more useful learning capabilities than that currently available, and thereby provide a stronger interaction of learning between a designer and a computer. `Controlled' computational learning is proposed as a means whereby the *Shared Learning* concept can be realised. Thus, within the *Shared Learning* concept the designer defines the requirements for knowledge, directs and controls the IDA's learning capabilities, makes enquiries about the knowledge IDA presents, makes judgements about this knowledge and is able to over-ride any knowledge presented by IDA. That is, the designer uses the system to learn about a design domain (i.e. *Domain Exploration*) and the design solution currently under development. In contrast, the IDA system should adapt its knowledge to meet the needs of the designer, carry out learning activities when requested and (in some instances) automatically present generated knowledge, continually maintain (i.e. update and evolve) its knowledge, provide explanations about learned knowledge and provide suggestions which may help guide the designer when exploring the design domain or solving particular design problems. Given PERSPECT's ability to support *Domain Exploration* , this system has been used to assess the viablity of the new *Shared Learning* concept [Duffy and Duffy96].

References:
[Duffy86] A H B Duffy: *Computer Modelling of Early Stage Numerical Ship Design Knowledge and Expertise,* Ph.D Thesis, Dept. of Ship and Marine Technology, University of Strathclyde, Glasgow, Scotland.

[Duffy and Kerr93] A H B Duffy and S M Kerr: *Customised perspectives of past designs from automated group rationalisations,* Artificial Intelligence in Engineering, Vol.8, Elsevier Science Publishers Ltd, 1993, pp183-200.

[Duffy and Duffy96] S M Duffy and A H B Duffy: *Sharing the learning activity using Intelligent CAD',* Jrnl. of Artificial Intelligence for

Engineering Design Analysis and Manufacturing (AI EDAM), v.10(2), special section on Machine Learning in Design, Cambridge University Press, New York, 1996.

[Duffy-etal95a] S M Duffy, A H B Duffy and K J MacCallum: *A Design Reuse Model*, International Conference on Engineering Design (ICED'95), Vol.2, Praha, 22nd-24th August 1995, pp 490-495.

[Duffy-etal95b] A H B Duffy, A Persidis and K J MacCallum: *NODES: A numerical and object based modelling system for conceptual engineering design,* Knowledge Based Systems, Elsevier Science Publishers Ltd, 1995 (to be published).

[Kerr and Duffy92]: *Dynamic memory by automating rationalisation of past designs,* Machine Learning in Design Workshop, 2nd Int. Conf. Artificial Intelligence in Design, Carnegie-Mellon University, Pittsburgh, USA, June'92.

[Kerr93] S M Kerr: *Customised Viewpoint Support for the Utilisation of Experiential Knowledge in Design,* Ph.D Thesis, CAD Centre, University of Strathclyde, 75 Montrose Street, Glasgow G1 1XJ, Scotland, 1993.

[MacCallum-etal85] K J MacCallum, A H B Duffy and S Green: *An Intelligent Concept Design Assistant,* Design Theory for CAD, IFIP WG5.2 Workshop 3 on Intelligent CAD, October 1985, Elsevier Science B V (North-Holland) pbs, 1987, pp301-317.

[Persidis89] A Persidis: *Modelling of abstractions for Computer Aided Design,* Ph.D Thesis, CAD Centre, University of Strathclyde, 75 Montrose Street, Glasgow G1 1XJ, Scotland, 1898.

[Persidis and Duffy89] A Persidis and A H B Duffy: *Learning in Engineering Design,* 3rd Int. Workshop on Intelligent CAD, IFIP WG5.2, Osaka, Japan, September 1989, H Yoshikawa, F Arbab and T Tomiyama eds, North-Holland pubs, 1991, p251-272.