

# デザインナレッジシート：CADCEUS の知識ベース設計支援機能

## Design Knowledge Sheet

### A New Capability of CADCEUS Supporting Knowledge-based Design

関戸勝己, 宮地恵美, 加藤公一

**要約** 現在のCADシステムはフィーチャ・パラメトリック・モデリング機能を備えることにより、強力な形状処理能力を獲得した。しかしながら、設計意図、設計基準を扱う能力は決して十分なものではなかった。このたび筆者等はDKS (Design Knowledge Sheet) なる機能を開発し、設計意図、設計基準を扱う能力を著しく向上させることに成功した。

DKSは、Microsoft Excelのシートに設計意図、設計基準を蓄積し、それを再利用することで、設計意図、設計基準に照らした試行錯誤モデリング機能、既存ノウハウとの連携機能などを提供する。これにより、設計業務の強力な支援が可能となった。

**Abstract** Modern CAD systems have achieved the strong shape processing capability by providing the feature parametric modeling capability. However their capabilities to handle design intentions or design criteria are still not enough. Now we have developed the new function named "Design Knowledge Sheet" (DKS), which enhances the ability to deal with design intentions and criteria.

The DKS stores intentions and criteria in sheets of Microsoft Excel and reuses them to provide trial-and-error modeling function referring to intentions and criteria. This capability also enables designers to associate intentions and criteria with existing know-how. Therefore, CAD systems with the DKS can support the design activities more effectively.

## 1. はじめに

本稿は、CADCEUS Version 7の新機能、Design Knowledge Sheet (以降、DKS) の開発についての技術報告である。

我が国のモノづくりを支えてきたCAD (Computer Aided Design) は、ワイヤ・フレーム、サーフェス・モデラ、ソリッド・モデラ、更には最新の3次元フィーチャ・パラメトリック・モデラと着実に進化を遂げてきた。これらの進化は、CADシステムがもつ「形状の表現力」、及び「形状を創成/変更する機能」という観点では実に大きな進化ととらえることができる。しかしながら、現場の設計者からは、「あくまでも形状創成・変更のツールでしかない」という指摘を受け続けているのも、また事実であり、Computer Aided Designではなく、Computer Aided Draftである、なる揶揄はあまりに有名である。

この指摘を更に掘り下げると、現在のCADの問題点として、次のようなものを考えることができる。

- ① 設計結果 (What) のみが残されており、設計意図、設計基準 (How, Why) が残されていない
- ② 寸法を矛盾無く、かつ形状生成の順番に従って決めていかなければならない。従って自在な試行錯誤ができない。(「清書」しかできない。)
- ③ 既に手元にある設計ノウハウとの連携が十分でない

このような現状に鑑み、このたびCADを真に設計のツールに昇華させることを目的としたシステム開発を行った。前述した問題点を克服するために、下記の目標を掲げた。

- ① 形状データのみならず、設計意図、設計基準をデータとして残すこと
- ② 自在な試行錯誤が可能なこと
- ③ 既存の設計ノウハウを適用可能なこと

これらの目標を実現するために、今回我々は以下のような方針を設定しシステム構築を行うことにより、ベースCADがもつ形状表現力と設計知識を連携させた。

- 1) ベースとするCAD（形状モデラ）は、3次元フィーチャ・パラメトリック・モデラのCADCEUSとする。  
⇒CADCEUSの形状モデリング能力は非常に強力であるため、これを利用した。
- 2) 設計知識はCAD（形状モデラ）とは分離し、設計知識によって形状を駆動する。  
⇒CADの内部に設計知識を埋め込む方法も考えられるが、設計知識の流用性、更には既存設計ノウハウとのシームレスな連携も考慮し、あえて設計知識をCADと分離する方法をとった。
- 3) CAD（形状モデラ）から分離された「設計知識」はMS-Excelシート（Design Knowledge Sheet）に蓄積し、再利用する。  
⇒設計知識の物理的格納方法にも複数の選択肢が考えられるが、設計者にとっても馴染みのあるツールであり、既存ノウハウの多くがMS-Excelシートに書かれていることを考慮した。

以下、次のような章立てで順次詳細を説明していく。まず、2章では、ベースとなる最新のCADシステムの特徴、及び適用状況について述べる。続く3章では、最新CADの設計業務への支援が不十分である事例を挙げ、その分析を行う。第4章では、DKSのコンセプト、アーキテクチャについて解説する。第5章ではDKSの適用例、及び期待できる効果について述べる。最終章では、今後の課題について述べる。

## 2. 最新CADの状況

本章では、まず現状認識を行うため、CADCEUSをはじめとする最新のCADシステムの特徴と適用状況について述べる。

### 2.1 機能特徴

CADCEUSの機能の特徴としては、次の点が挙げられる。

- ① ソリッド、サーフェス、ワイヤの自在な混在が可能なハイブリッド・モデラ  
CADCEUSの形状表現は非常に柔軟性に富む。ソリッド、サーフェス、ワイヤの混在はもとより、サーフェスを繋ぎ合せてソリッドとする。あるいはソリッドをサーフェスで切断し、サーフェスとして取り扱うなどのモデリングが可能である。
- ② ワイヤ形状に手順に依存しない拘束条件を付加できる対話的スケッチ・モデラ  
2次元ワイヤモデリングを強力に支援する機能である。寸法拘束はもとより、角度拘束など多様な拘束条件を与えると、これらの拘束条件を変更することによるパラメトリックな変形が可能である。
- ③ 履歴ベース・パラメトリック・モデラ

ハイブリッド・モデラに対する操作履歴をシステム内に蓄えておき、設計変更などある履歴の変更が必要となった場合、その履歴を編集し、更にその履歴と連想性をもつ履歴を再実行（再生）することができる。

## 2.2 適用状況

CADCEUSの機能特徴を使った実際の設計手法は、次のように分類される。

### ① 新規設計時のトライアル&エラー

もっとも基本的なパラメトリック機能の応用である。モデリング手順をパラメトリック・モデラ上に作り込んでいき、評価の結果が望ましくなかった場合は、ある手順の寸法を変更し、その変化に追随した全体の形状変更をパラメトリック・エンジンに委ねる手法である。この形状変更の自動化により、設計変更の工数削減が期待できる。

### ② 雛形設計，流用設計

過去に作成されたパラメトリック・モデルの手順の全て、あるいはあるまとまった一部分を切り出し、「ライブラリ」として登録しておき、類似品のモデリングにあたり、これを再利用する設計手法である。これも大幅なモデリング工数削減の効果が期待できる。

### ③ 最適設計

パラメトリック・モデラと最適化エンジンを連携させ、パラメトリック・モデルの諸元（寸法）を最適化エンジンの指示に従って摂動させながら、慣性モーメントなどの目的量を自動的に最適値に近づけていく方法である。これはモデリング、モデル評価、モデル変更の全てを自動化することで、工数削減と質の高い設計を狙った手法である。

## 3. 最新のCADの問題点

最新のフィーチャ・パラメトリック・モデラは、モデルが頑健で、思考錯誤がない設計手順の場合には、工数削減、設計対象の高品質化に寄与している。しかしながら、モデルが複雑な場合には現システムの実装方法では効率や頑健性に問題があり、工数削減、品質向上につながらないこともある。

また、実装方法が改善されてシステムの頑健性が向上したとしても次のような課題が残っている。

- 1) 「ライブラリ」とはいつても、形状作成手順の並びであり、そこには「設計意図」「設計基準」が存在しない。そのため、再利用を続けると、「ライブラリ」が「ブラック・ボックス」と化してしまう。単純な再利用をしている限りは問題がないが、「ライブラリ」の内部を改修する必要が発生した場合のその改修作業は非常に困難な作業となる。
- 2) 実際の設計では、単純な最適設計では済まない場面が多々ある。そこで評価基準はむしろ「全体のバランス」、あるいは「設計者のイメージ、更には嗜好」といった、数値化が困難なものが重要となることが多い。このような場合、設計者は試行錯誤をしつつ設計の完成度を高めていく。より具体的には、ある寸法を決めておいて、他の寸法を振らせる、といった具合に試行錯誤を行うわけだが、現在のパラメトリック・モデラは、寸法間の整合がとれた状態で、操作手順に従った形状を表現することしかできない。ましてや、設計ルールをあえて踏み越えたトライアル、あるいは設計ルールの優先度の考慮やその変更は

不可能である。

- 3) CAD システムの外部にある知識をシームレスに活用することができない。活用するためにはCAD システム上でのマクロ作成といったプログラミングが必要になる。

## 4. DKS 解説

### 4.1 機能概略

前章で述べた最新 CAD の問題を解決するために、以下のような機能を開発した。

- ① 設計意図，設計基準の記述機能  
設計パラメタを設計者の用語を使って簡単に会話型で記述定義する機能
- ② 試行錯誤の支援  
設計時の試行錯誤を可能とするため，設計者の意図を汲み取りつつ，寸法間の矛盾を自動的に解決し，それを形状に反映させる機能
- ③ CAD システム外部に存在する知識との連携  
DKS と CAD システム外部に存在する知識とを連携させる機能

### 4.2 機能詳細

本節では，前節で述べた機能について更に詳細に説明を行う。

#### 4.2.1 設計意図，設計基準記述機能

- 1) CAD データと設計者用語との対応付け機能  
本機能は，「ルール・エディタ」なるモジュールにより提供される。ルール・エディタにより，CAD 図形要素と設計者用語の対応付けがなされ，この設計者用語によって DKS に設計ルールを記述することができる。
- 2) 対話的操作インタフェース作成機能  
DKS では，「変数」以外にも後述する様々な定義要素を用いて設計意図，設計基準を記述することができる。しかしながら，実際の設計行為にあたって使用するインタフェースには，その「全て」が見えている必要はなく，むしろ「興味のある」要素のみが，設計者の嗜好に合わせてレイアウトされていることが望ましい。「GUI エディタ」なるモジュールにより，この機能が提供される。この「GUI エディタ」の操作性は非常に簡便なものであり，ダイアログ構成要素（コントロール）をドラッグ&ドロップしていだけで，ダイアログが作成可能である。ここで作成されるダイアログを DKI (Design Knowledge Interface) と呼ぶ。
- 3) 対話的操作機能  
DKI のダイアログを操作することにより，CAD 形状を対話的に操作することが可能である。

#### 4.2.2 ルールによる CAD 駆動機能

DKS に次のような要素を定義することで，寸法間に矛盾が発生した場合でも，それをシステムが自動的に整合をとり，CAD 図形に反映させることが可能である。

## 1) 寸法（パラメタ）間関係記述（制約式）

「ボルトの長さ=板厚+ナットの厚さ」

のような記述を「制約式」と呼ぶ。多くのCADでも寸法による式定義が可能であるが、それらは「代入」である。つまり、左辺の値は右辺の値から決まる値であり、左辺のパラメタを直接変更することはできない。しかしながら、DKSでは、制約式の左辺に記述されたパラメタと右辺に記述されたパラメタのいずれの値も変更することが可能である。

## 2) 寸法（パラメタ）間の「強弱関係」の記述と動的制御

一般に設計対象の寸法（パラメタ）はそれぞれが独立、かつ平等であるわけではない。何らかの従属関係、そして強弱関係がある場合が多い。非常に単純な例を図1に示す。

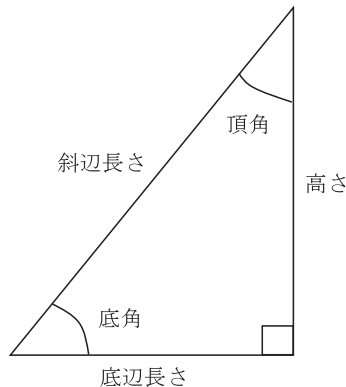


図1 寸法間の競合の例

図1に現れる五つの寸法は独立ではない。例えば、「底辺長さ」「斜辺長さ」を固定し、「底角」を変化させると、それに従って「頂角」「高さ」も変化する。この場合、「頂角」「高さ」は「底角」に従属していると考えることができる。あるいは、設計者の言葉を用いるならば、「底角」を指示すると、「頂角」「高さ」が成り行きで決まる、とも表現できる。このような寸法間関係を「強弱関係」と呼ぶことにする。これは設計の場面で非常に重要な概念である。

DKSでは、このような強弱関係を「コントロール・シート」により記述することが可能である。また、実際の設計では、この「強弱関係」を設計行為の最中に変更する場合も多々ある。このような要件に応えるために、DKSではパラメタの値を入力するときに、動的に強弱関係の切替を行うことができる仕組みをもつ。

## 3) チェック式

寸法（パラメタ）が、設計基準を満たしているかを監視する式である。

## 4.2.3 外部知識連携機能

以下のような外部知識の利用が可能である。

- 1) DKSの「外部」に存在する情報へのリンク（URL）を記述し、DKIから閲覧することが可能
- 2) 他のMS-Excelファイルのセルへのアクセス（I/O）を定義することが可能
- 3) 他のプログラムの起動が可能

### 4.3 アーキテクチャ

本節では、前節で述べた機能を実現するための、システム・アーキテクチャについて説明する。

#### 4.3.1 DSK の二つのユースケース

CAD から分離された設計知識は MS-Excel シート (DKS) に格納される。そして、この DKS を取り扱うモジュール群を総称して、DKS システムと呼ぶ。DKS システムが提供する機能は、大きく分類すると以下の二つに分けられる。ここで、DKS を応用した設計支援機能を DKS アプリケーションと呼ぶ。

- 1) ルールやユーザインタフェースを定義し、DKS アプリケーションを作成する。
  - 2) 1) で定義された DKS アプリケーションを利用して、設計作業を行う。
- これら二つのケースのアーキテクチャを説明する。

#### 4.3.2 DKS アプリケーション作成時のアーキテクチャ

DKS アプリケーション作成時、DKS システムはインターフェースとしてルールエディタと DKI エディタを提供する。

ユーザはルールエディタを用いて DKS を編集する。ルールエディタは、MS-Excel のアドインとして動作するため、MS-Excel の機能をほとんどそのまま利用することが可能である。また、CADCEUS とルールエディタは相互に通信しているため、CADCEUS 上の図形要素のクリックによって、その図形要素を DKS に取り込むなど、直感的な編集操作が可能になっている (図 2)。

GUI エディタは、DKI を編集する機能である。これは Visual Basic (VB) のアドインとして動作し、通常の VB の GUI 設計と同様に、画面に部品 (コンストラクションパーツ) を貼り付けていく直感的な操作でインタフェースを作成できる。DKS システムが提供するコンストラクションパーツは、DKS アプリケーション実行時に、それぞれが後述する「DKI マネージャ」と通信することにより動作する。

CADCEUS、ルールエディタ、GUI エディタはそれぞれ別のプロセスとして動作するので、それぞれ独立に自由な順序で操作が可能である。これを実現するために、CADCEUS 上には他プロセスに対し、図形要素へのアクセス、並びにフィーチャ・パラメトリック機能を提供するための API (Application Programming Interface) を構築した。これを、F-API (Feature API) と呼ぶ。

#### 4.3.3 DKS アプリケーション実行時のアーキテクチャ

DKS アプリケーション実行時、DKS システムはユーザインタフェースとして DKI を提供する。

DKI (バイナリ実行ファイル) はユーザへ「表示」と「入力操作」を提供するのみで、全体の制御は「DKI マネージャ」が行う。DKI マネージャは DKS の読み込みや、入力データ (寸法値など) の解釈を行い、それらのパラメタを与えてパラメタ・ソルバを起動する。パラメタ・ソルバは、パラメタ間の拘束関係、パラメタ間の強弱関係を解決し、整合のとれたパラメタ値を結果として返す。DKI マネージャはパラメタ・ソルバの結果を DKI に反映し、必要に応じ

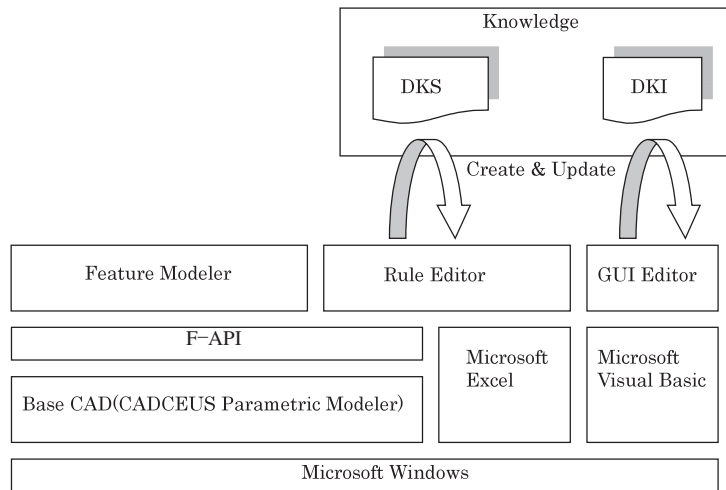


図2 DKS アプリケーション作成時のアーキテクチャ

てエラーメッセージ表示などのイベントを発生させる。

パラメタ・ソルバは、DKS システムの中核となるモジュールであり、前述したパラメタの整合をとる処理のほか、パラメタにより CAD データのパラメトリック変形を駆動する処理も行う。ここでのパラメトリック変形の駆動も、前項で述べた F-API を介して行われる。これらの構造を図式化すると図3のようになる。また、実行時の内部処理の流れは図4のようになる。

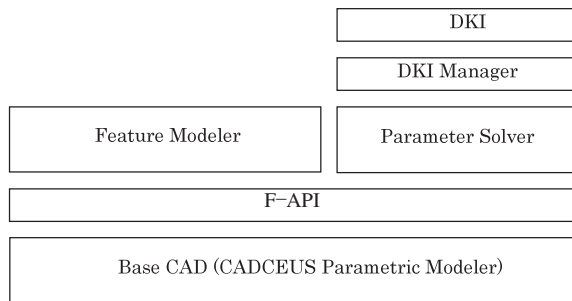


図3 DKS アプリケーション実行時のアーキテクチャ

## 5. 適用事例

本章では、設計行為における DKS の支援事例、および有効性について述べる。

### 5.1 レイアウト構想設計

設計対象の寸法は互いに様々な関係を持っているが、レイアウト構想設計の場面では、特に諸元の競合（せめぎあい）が発生することが多い。つまり、ある寸法は性能を確保すべく可能な限り大きく、しかしながら製品全体の寸法はコンパクトにするために可能な限り小さく、という相反する条件が示される場合がほとんどである。設計者は、この課題を解くために、過去の事例や、自らのバランス感覚を駆使し、最適解を導き出す。つまり、ここでいう最適解とは、最適化エンジンで計算されたような単純なものではない。

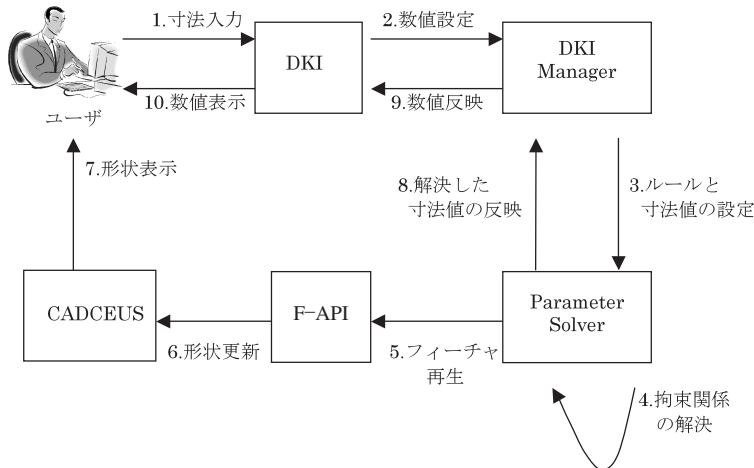


図4 DKSアプリケーション実行時の内部処理

このような場面で、DKSは4章で述べた以下の機能により、設計者の支援を行う。

- ① 設計者が「こうでなくてはならない」と指示した寸法を固定し、その寸法と競合する他の寸法群の整合を自動的にとり、更にCADデータも追随させる機能。
- ② 競合する寸法群の「強弱関係」のパターンをDKSに登録しておき、そのパターンのうち一つを動的に適用させる機能。
- ③ 絶えず図形要素間のクリアランスを監視する機能

より具体的な事例としてエンジン・バルブ・レイアウトを次に示す。

### 5.1.1 エンジン・バルブ・レイアウト

バルブ・レイアウトとは、エンジンの構想設計段階に行われる設計行為である。設計者は図5に示すようなモデルに対し、次のような課題に取り組まなければならない。

「与えられたボア径内に、最大のバルブ径をバランスよくレイアウトし、かつシリンダヘッドをいかにコンパクトに収めるか？」

シリンダ・ヘッドをコンパクトにするために、カム軸ピッチを小さくすると、バルブどうしが干渉するため、バルブ径を小さくせざるを得ない。また、バルブ径を大きくとろうとすると、カム軸ピッチを広げざるを得ない。つまり、「あちらを立てればこちらが立たず」状態に陥る。このようなケースでは、設計者は前述したように「バランス感覚」「過去の経験」を駆使して、試行錯誤しながら、寸法を決めていかなければならない。

このような場面でDKSは、

- ① 競合する寸法間の調整
- ② 寸法の強弱関係の切替
- ③ クリアランス・チェック

を、行うことで、設計者の試行錯誤を強力に支援する。



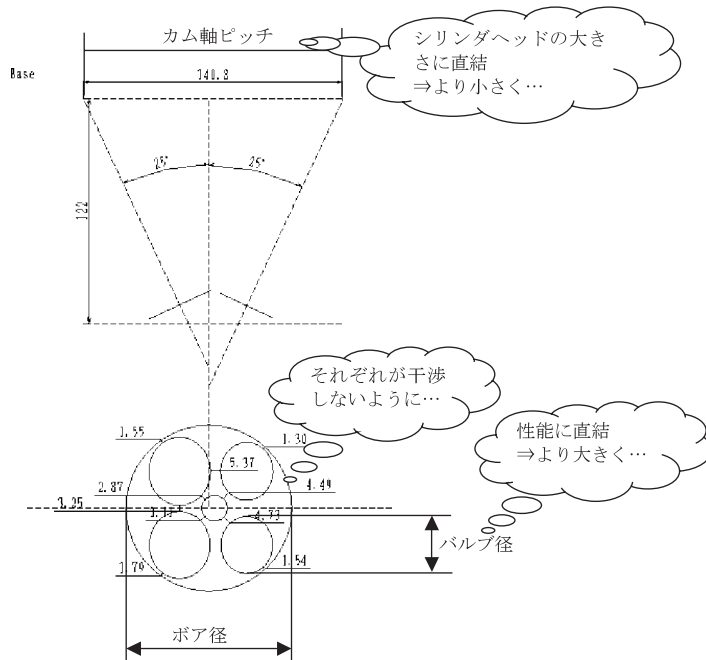


図5 エンジン・バルブ・レイアウト

### 5.1.2 効果

バルブレイアウト設計ではDKSを適用することにより、以下の効果を確認することができた。

- 1) 個々の設計者がもっていた設計ノウハウの「可視化」に成功した。  
より具体的には、
  - ① 設計に必要なすべてのパラメタを列挙することができた。
  - ② これらのパラメタの関係を整理することができた。
  - ③ 競合する寸法間の調整方法をパターン化し、その可視化（コントロール・シート）に成功した。
- 2) 設計作業の品質と効率の向上に成功した。  
より具体的には、
  - ① 形状を操作するのではなく、設計要件に直結するパラメタを直接操作するため、本来の設計行為に近づくことができた。
  - ② パラメタ間の関係と、それに対するチェックが組込み可能なため、自動的な形状変形とそのチェックが可能となった。
  - ③ 競合する寸法間の調整方法を動的に切り替えることが可能となったため、設計に柔軟性が生まれ、より創造的な設計が可能となった。また、この競合する寸法間の調整方法そのものも「知識」として蓄積、再利用可能となった。

### 5.2 既存ノウハウとの連携

設計行為には、必ず「評価」のプロセスが存在する。ほとんどのCADシステムはこの「評価」を支援する機能を備えているが、これらの評価内容は、例えば、ある立体の体積、あるいは

は、ある図形とある図形が干渉するか否か、といった非常にプリミティブなものである。しかしながら、設計行為における評価では、さらに高度なものが必要とされることも多い。この場合、なんらかの特化したツールが使われることが多い。これらのツールは、簡易的なものでは、Excel の計算式/マクロ、そして本格的なものとしては、CAE が挙げられる。

DKS は、MS-Excel とのスムーズな連携を実現することで、設計者の手元にある既存の評価ノウハウをダイレクトに CAD データに適用することができる。

具体的な事例として、次にモールド金型水管設計を示す。

### 5.2.1 モールド金型水管設計の事例

モールド金型水管設計とは、モールド・ベース構想設計において、金型を冷却するための冷却水管の諸元(長さ、太さ、本数)および配置を決定する行為である。諸元計算は樹脂の種類、成形品の大きさ、サイクルタイムなどをもとに、Excel の計算式/マクロによって行われる。この Excel は CAD とは独立したデータである。従来は、この Excel で諸元を決定し、それをもとに設計者がモデリングを行っていた。DKS では、この Excel と連動する「水管配置ルール」を用意しておくことで、Excel の計算式をバックグラウンドで動作させつつ、ダイレクトに CAD モデルを操作することが可能となる(図 6)。

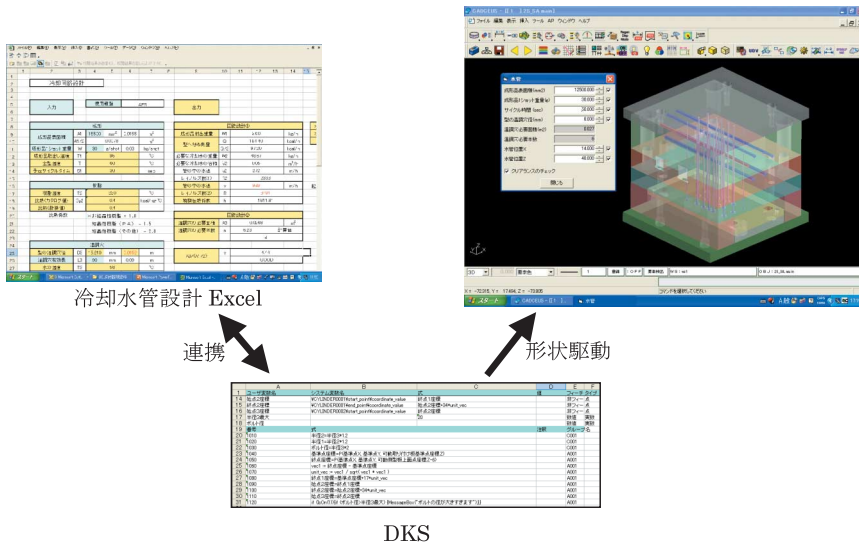


図 6 モールド・ベース構想設計における DKS/外部ノウハウ/CAD の連携

### 5.2.2 効果と今後の展望

バルブレイアウト設計の場合と同様、水管設計の場合も、個々の設計者が持っていた既存ノウハウの可視化と再利用性の向上に成功した。この可視化のプロセスで同時に可視化されたパラメータを、外部の情報、プログラムと連携させることが可能となったため、(専用マクロ開発などの)準備工数、および実際のモデリング工数の削減が確認できた。今回は評価用 Excel シートとの連携を実現したが、流体解析などの CAE と連携させることにより、更に質の高い設計が可能になると考えられる。

## 6. おわりに

これまで述べてきたようなシステムの構築により、更に「設計のための道具」としてのCADに近づくことができたと考える。そして実際に5章で述べたような効果も確認することができた。しかしながら、依然として次のような課題が残されている。

- ① 知識獲得の簡易化
- ② 知識の管理
- ③ 知識の高度化

### 1) 知識獲得の簡易化

「暗黙知」を「形式知」としてシステムで利用できる形式に変換することが知識の獲得であり、その変換過程をいかに簡易に行えるかが、システムが実用に耐えるか否かの分かれ目であるとする。設計知識の獲得に関しては、設計者の設計作業を観察、あるいは設計者自身が解説をしながら設計する様子をビデオ録画し、そこから「知識」の抽出を試みる。設計過程を分析することで、設計対象知識を使う操作を分析して定式化する<sup>[3]</sup>、設計で必要となる標準、規格、加工法、制約条件、過去の失敗例など設計支援に必要なあらゆる情報をデータベース化していく<sup>[4]</sup>、など、様々なアプローチによる研究が行われてきているが、実用で広く利用されるには至っていない。

DKSのコンセプトの基本は、設計者の知識とCADモデルの分離にある。したがって、設計知識を形式化するための、「言葉」（具体的には設計パラメタ）と、これらの「言葉」とCADモデルとの対応付けが、知識獲得の最も重要なプロセスであるとする。よって、この対応付け作業のプロセスを、操作が容易なシステムで支援できれば、「知識獲得の簡易化」の解決方法のひとつになると考えている。

具体的には、設計の言葉とCADモデルとの対応付け機能を行う2つの機能とその言葉を使って設計知識を記述する以下の3つの機能の開発を進めている。

- ① 設計者が設計対象を語るために必要な単語を列挙してDKSに記述、これらの単語に対応するCADデータ（立体、部分、寸法など）をCAD画面上で指示。
- ② CAD画面上でCADデータを指示すると、指示したデータに対応するシステム名がDKSに登録され、設計者が、その名前を設計の言葉に変更する。
- ③ 設計の言葉を使った設計知識の記述をウィザード形式行う。

### 2) 知識の管理

上記の課題がクリアになれば、設計者の数だけ、そして設計行為の数だけの知識が吸い上げられることになる。しかし、知識がただ雑然と蓄積されただけでは、それらの再利用に著しい齟齬を生ずるのは明白である。従って、逐次蓄積されていく知識を再利用しやすいよう管理し、容易に検索が可能となるような仕組みが必要となる。また、自動的に不要な知識を排除する仕組み、あるいは知識を更新する仕組みも必要となろう。これらの仕組みの構築には、DKSがExcelで記述されていることが大きな利点の一つになると考えている。

### 3) 知識の高度化

DKSに記述する対象物は限定されていない。いかなる設計分野での利用も可能であるし、また、設計知識には後工程を考慮した知識が多く含まれるという必然から、CAMや生産の知識を扱うことも可能である。したがって、上記二つの課題がクリアになれば、DKSを利用して最新の材料技術、加工技術、生産技術を俯瞰できる仕組みを構築することも可能

となる。さらに、蓄積された多数の設計知識、および失敗例を組み込むことにより、設計者が、全く新しい高度な設計基準を創出するための強力な支援ツールとして利用することもできる。

- 
- 参考文献** [1] ユニシス技報 71号 2001年11月発刊 Vol. 21 No. 3, 「CADにおける知識の利用形態」 宮地恵美  
[2] 型技術, 2003 Vol. 18 No 8, 「モールド金型設計への「CADCEUS/Design Knowledge Sheet」の適用」, 宮地恵美, 関戸勝己  
[3] 精密工学会誌 vol. 67, No. 9, 2001, 設計対象知識操作論, 吉岡真治, 富山哲男  
[4] 情報処理 2000年7月41巻7号 設計者が欲しい設計支援システムの開発, 畑村洋太郎, 中尾政之

**執筆者紹介** 関戸 勝己 (Katsumi Sekido)

1966年生。1989年東京大学工学部精密機械工学科卒業。  
1990年日本ユニシス(株)入社。CAD/CAMシステムの開発に従事。現在、日本ユニシス・ソフトウェア(株)に出向中。

宮地 恵美 (Emi Miyachi)

1957年生。1982年慶応義塾大学工学系研究科修士課程修了。同年日本ユニシス(株)入社。CAD/CAMシステムの開発に従事。現在、アドバンステクノロジー本部ビジネスプロデュース部所属。情報処理学会員。

加藤 公一 (Kimikazu Kato)

1971年生。1998年東京大学数理科学研究科修士課程修了。同年日本ユニシス(株)入社。CAD/CAMシステムの開発に従事。現在、日本ユニシス・ソフトウェア(株)に出向中。