

## CFD 検証データベースの開発

高木 亮治\*、山本一臣\*、山根敬\*

### Validation Database for CFD

Ryoji TAKAKI\*, Kazuomi YAMAMOTO\* and Takasi YAMANE\*

#### ABSTRACT

This paper presents the development of a validation database for Computational Fluid Dynamics(CFD), which is important to enhance reliability of CFD codes. This database is a kind of a new knowledge sharing system using a data description language called eXtensible Markup Language (XML). One of the key technology of the database is the data structure: defined data and their relations. Defined data in the database are *geometry*, *grid*, *analysis*, *parametric*, *comparison*, *solver*, *derived* and *experiment*, which come from the analysis of flow calculation processes in CFD. The development of the database has been finished and the internal operational test has started. It is necessary for the official operation to discuss the quality, reliability and disclosure policy of registered data and programs.

#### 1. はじめに

宇宙航空研究開発機構総合技術研究本部情報技術開発共同センターでは、前身である航空宇宙技術研究所 CFD 技術開発センターの時代から引き続き CFD 信頼性向上に向けた様々な取組を行なっている<sup>1, 2, 3, 4, 5)</sup>。これら一連の取組の中で、最近注目を集めているデータ記述言語 XML を用いた CFD 検証データベース<sup>6)</sup>の開発を行なっている。本データベースは単に CFD の計算結果のみならず、検証結果の有効な蓄積および一般への公開を目的としている。更には、単に検証データの提供だけに留まらず、ユーザーが自由に利用可能な計算格子及び格子情報、更には UPACS<sup>7)</sup>を基幹とした共有プログラムあるいは比較検証用ソフトウェアの利用を可能とすることで全日本的な CFD 検証プロジェクトが行える環境の構築を目指している。ここで重要なことは、単に計算結果、実験結果を公開するだけではなく、それらを有効に活用できる環境の基でそれらの

情報を共有することである。つまり計算結果を単純に提供するのではなく、計算に用いた全てのデータ、計算コードを初めとして、結果を得るまでの過程を、データの利用者が追体験できるレベルで共有する必要がある。そのため中核となるデータベースには従来以上にユーザーが積極的にデータベースを活用し、データをより共有できる仕組みが必要となる。CFD コードとそのコードの検証データ一式をセットにすることで結果的に CFD の信頼性を確保することができると考えている。

以上の目的を実現するために新しいタイプの情報共有システム基盤である「知見情報プラットフォーム」<sup>8)</sup>の考え方を導入した。「知見情報プラットフォーム」は「知見プラットフォーム推進協議会」で提唱されている科学技術計算における様々な技術情報を共有するための技術的フレームワークである。「知見情報プラットフォーム」では情報共有システム基盤を構築するために XML(EXTensible Markup Language) が用いられ

\*宇宙航空研究開発機構 総合技術研究本部 情報技術開発共同センター

ている。XML は人とアプリケーションプログラムの両方がその目的に沿って理解し処理できる形式でデータを記述する事を目的とした汎用的なデータ記述言語であり、Web 関連の新技术として急速に普及している。XML の特徴はデータに対してそのデータの内容を示すタグを付けて保存する点、そのタグをユーザーが定義できる点である。つまりオブジェクト指向的にデータを中心に考え、そのデータが持つ特性、扱われ方(メソッド)をデータ自身に記述することでデータの再利用性を高めることができる(ここではデータの中にアプリケーションプログラムも含めている)。XML はその汎用性、柔軟性により近年 e-business, e-commerce では標準的なデータ記述言語の地位を確立し、広く一般に使われるようになり、IT 革命の原動力となっている。このように汎用的なデータ記述言語である XML を用いた「知見情報プラットフォーム」を利用する事で従来にない機能を持った CFD 検証データベースの開発を行なった。

## 2. CFD 検証データベース

本システムでは一般のデータベース(データの管理、検索)とは異なり以下の特長を有する。まず、データ構造に関しては、CFD における解析作業を分析および整理し、それに適合するデータ構造を策定し、データベースシステムに反映した。その際に CFD に限らず科学技術計算で用いられるデータへの汎用性もある程度考慮した。また、入力支援機能および自動化機能を実現した。これらの特長についてそれぞれ紹介していく。

### 2.1 システム構成

本データベースのシステム構成を図1に示す。全体的にフリーソフトを活用したシステム構成となっている。本システムは Linux マシン上で構築され Web サーバーは tomcat, XML データベースには Xindice を用いている。XML でデータ構造を表現する場合、XML スキーマや DTD を用いるのが一般的であるが、本データベースシステムでは、その役割をシステム自体および、後述するマッピングファイルで実現している。

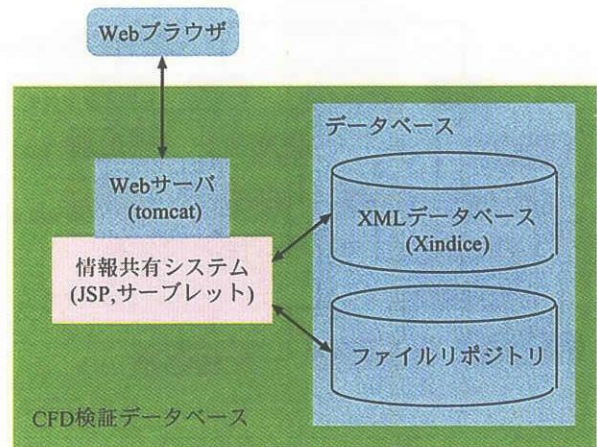


図1: システム構成

### 2.2 データ構造

本データベースの開発ではまず CFD を用いた解析作業をデータとその流れもしくは関係という観点で整理し、データベースのデータ構造の検討を行なった。その結果、定義されたデータおよびその関係を表1および図2に示す。

表1: CFD 検証データベースにおけるデータ

構造名	内容	参照先
geometry	形状データ	(ルート)
grid	格子データ	geometry
analysis	計算結果	geometry, solver
parametric	計算結果群	複数の analysis
solver	CFD コード	(ルート)
experiment	実験データ	geometry
derived	比較データ	analysis, experiment, parametric
comparison	比較	derived

今回開発を行なった CFD 検証データベースは知見プラットフォーム<sup>8)</sup>をベースにしている。知見プラットフォームではある情報の集まりを XML を用いて記述したときにそれらを文書と呼んでいる。ここでは様々な情報をひとまとめにして XML で記述したときに、物理的に一つの XML ファイルで記述されたデータの集まりを文書と呼ぶことにする。



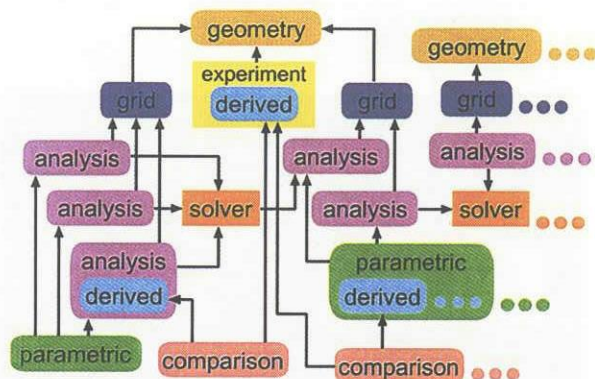


図 2: データと参照関係

**geometry** 計算対象となる物体形状に関するデータで、独立した XML 文書で定義される。具体的には CATIA データ等形状を定義する情報が格納される。また geometry 文書はデータ構造の Root(幹)と位置付けられている。

**solver** 計算に用いた CFD プログラムのソースプログラム等で、独立した XML 文書で定義される。solver 文書もデータ構造の Root である。

**grid** 計算対象の形状に対して作成された計算格子に関するデータで、独立した XML 文書で定義される。計算格子データである座標データ、複合格子法であれば各種接合情報(マルチブロック格子のブロック間接合情報、重合格子の補間インデックスなど)、更にはそれら計算格子を作成した格子生成プログラム関連の情報が格納される。一般には計算条件、例えば流れ場の速度域(超音速か遷音速以下かなど)に応じて適切な計算格子を生成するため、一つの形状に対して複数の計算格子が作成されるが、それらはそれぞれ別の文書として管理され、それぞれ基になった geometry 文書に関連付けられる。図 3 に grid 文書のイメージを示す。ここでは対象となる形状(nexst-1 を想定)に対して超音速流用の計算格子のデータが格納されている。

**analysis** ある計算条件下での計算結果に関するデータで、独立した XML 文書で定義される。計算を行なう際に必要となる境界条件、入力条件、実際の計算結果である計算空間の各種物理量分布、収束履歴等、計算の結果得られるデータで構成される。

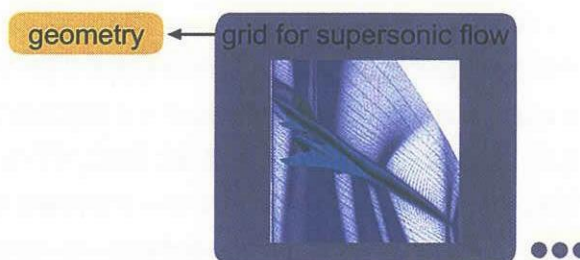


図 3: grid 文書

計算を行なう際に用いた計算格子及び CFD プログラムに関しては独立した XML 文書として analysis 文書外で定義されているので、analysis 文書はそれらへ関連付けられる。図 4 に analysis 文書のイメージを示す。ここでは grid 文書と solver A 文書で定義された計算格子および CFD プログラムを用いてマッハ数  $(M)=2$ 、迎角  $(\text{aoa})=0$  度の計算が行なわれたことを示している。また計算結果の他に、30%、50% スパン位置での  $C_p$  分布が derived データ(後述する)として定義されていることを示している。マッハ数  $(M)=2$ 、迎角  $(\text{aoa})=2$  度の計算の様に solver A を改良した solver A' を利用した場合は analysis 文書内に solver A' のソースプログラムを格納するようになっている。

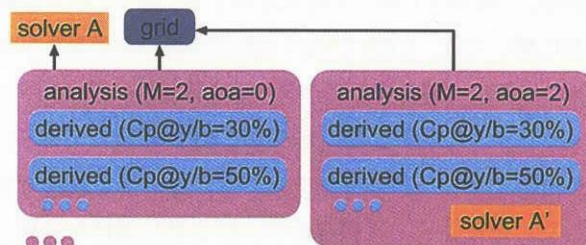


図 4: analysis 文書

**experiment** ある形状に対して実験で計測されたデータで、独立した XML 文書で定義される。CFD と同様に geometry 文書に関連付けられる。また文書中に記述される情報は、例えば空力特性  $(\alpha-C_L$ 、ポーラーカーブ)や一定断面位置での圧力分布という様に CFD と比較する単位で derived データ(後述する)にまとめられる。図 5 に experiment 文書のイメージを示す。この例では geometry 文書で定義された形状に対するマッハ数  $(M)=2$  の計測結果



を示しており、データとしては迎角 ( $\text{aoa}$ )=0 度の時の 30% スパン位置での  $C_p$  分布とポーラーカーブ ( $C_L - C_D$ ) が derived データとして定義されていることを示す。

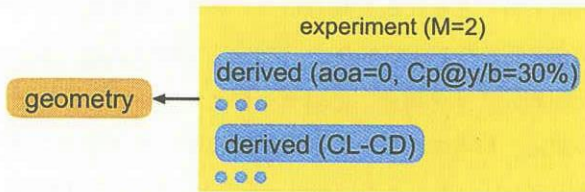


図 5: experiment 文書

**derived** 比較対象となる抽出された分布データ。derived データは独立した XML 文書ではなく、analysis 文書、parametric 文書 (後述する)、experiment 文書内で定義される。図 6 に derived データのイメージを示す。この例ではグラフで示されたある断面 (この場合は 30%、50% スパン位置) での  $C_p$  分布を比較対象のデータとして抽出し、それらを derived データとして analysis 文書内に格納していることを示す。

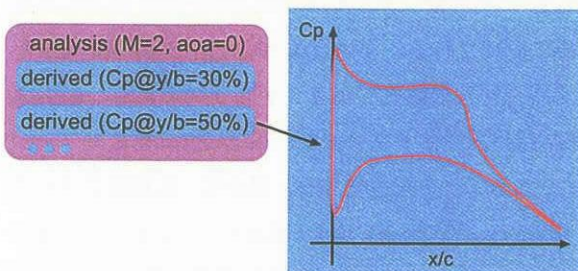


図 6: derived データ

**parametric** 複数の計算結果をまとめたデータで、独立した XML 文書で定義される。例えば、ポーラーカーブの作成の様に、複数の計算結果をまとめて一つのグラフを作成した場合、使用された  $C_L$  と  $C_D$  の値及び作成されたグラフが格納される。parametric 文書は基となる複数の analysis 文書 (正確には analysis 文書内にあるそれぞれの derived データ) と関連付けられる。図 7 に parametric 文書のイメージを示す。ここではマッハ数 ( $M$ )=2 で行なった迎角スイープの計算を一つの parametric 文書に

まとめ、その中でポーラーカーブ特性を derived データとして定義している。

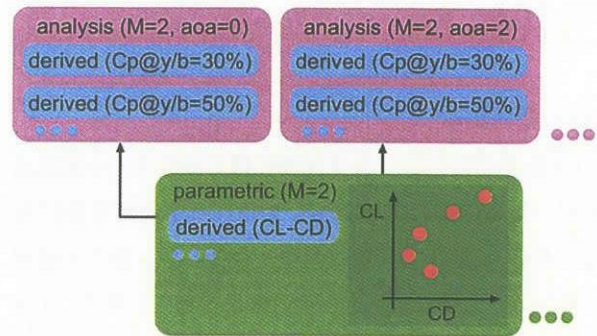


図 7: parametric 文書

**comparison** 各種データ (計算結果、実験データ) を比較したもので、独立した XML 文書で定義される。例えば、ポーラーカーブに関して、計算結果 (analysis 文書内の derived データ) と実験結果 (experiment 文書内の derived データ) を比較したグラフとデータが格納される。comparison 文書は基となる analysis 文書、parametric 文書、experiment 文書 (正確にはそれらの文書内にある derived データ) に関連付けられる。図 8 に comparison 文書のイメージを示しており、ポーラーカーブに関して、マッハ数 ( $M$ )=2 での迎角スイープの計算 (parametric 文書) と実験結果 (experiment 文書) を比較したもの及び、30% スパン位置での  $C_p$  分布を計算と実験で比較したものがそれぞれ comparison 文書で定義されている様子を示す。

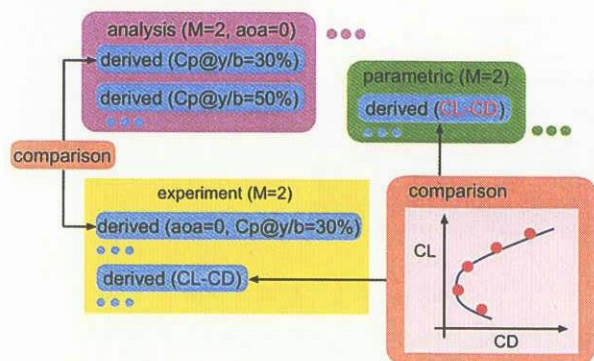


図 8: comparison 文書

これらの文書に関して、個々に格納されるデータ以外



に全文書に共通的に文書の登録者、登録日時、コメント、更には修正履歴(修正者と修正日時、コメント)が記述される。

### 2.3 配列記述

CFDを始めとして科学技術計算で多用される多次元配列の記述に関してもXMLを用いることで柔軟に対応できる。CFDで一般に問題となるのが、多次元配列の持ち方である。3次元の構造格子で考えた場合、3次元の各インデックスI、J、Kを用いて3次元座標(x、y、z)を示すために一般に4次元配列を用いるが、3次元座標(x、y、z)を格納する場所は、ベクトル計算機の場合はインデックスの一番最後に、スカラー計算機の場合は一番最初に設定する。なぜなら、計算機のアーキテクチャ上そうした方が計算性能が高くなるからである。このように様々な理由でフォーマットを統一することは非現実的であり、多次元配列の記述に関しては柔軟に対応する事が必要となる。そのため、図9で示すような多次元配列の汎用的な記述法を設定した。ここでは、構造化を示すインデックスと物理量を示すインデックスをそれぞれ別に記述することで、任意の多次元配列(どのインデックスで、何が定義されているかを記述する)に対応できるようにした。図ではベクトル計算機用データ、スカラー計算機用データの記述例を示しており、両者に対応できることがわかる。

```

<array content="..." type="...">
  <dimension type="index" length="..." name="..." />
  ...
  <dimension type="variable">
    <element name="..." class="..." />
  </dimension>
  <file></file>
</array>

<array content="grid" type="double">
  <dimension type="index" length="81" name="I" />
  <dimension type="index" length="41" name="J" />
  <dimension type="index" length="41" name="K" />
  <dimension type="variable">
    <element name="x" class="nondimensional" />
    <element name="y" class="nondimensional" />
    <element name="z" class="nondimensional" />
  </dimension>
  <file>grid001.fub</file>
  X(I,J,K)
</array>

```

図9: 多次元配列の記述

### 2.4 入力支援機能

データベース内のデータ構造は汎用的な構造となっており、様々なデータを入力する必要があり、入力の手

間が非常に大きいと考えられる。そのため、入力の手間を省くため入力支援機能を作成した。これは、特定のCFDプログラム向けのマッピングファイルを用意することで、入力すべき情報の大部分を自動的にシステム側に入力できる仕組みを用意した。入力支援機能として現在実装されているものは、ファイルの命名規則、多次元配列のデータ構造、必須データとオプションデータの設定、検索キーワードの自動抽出等である。入力支援のマッピングファイルを用いることで、同じCFDプログラムでもバージョンアップに伴う入出力ファイルの変更、入力パラメータの追加等にも柔軟に対応できる。ここではJAXA/ISTAの標準CFDコードUPACS<sup>7)</sup>の場合を例にしてこの記述法を説明する。

ファイルの命名規則ではファイル名が固定のもの、もしくは命名規則に従う場合は正規表現を用いて記述することができる。UPACSではマルチブロックの接合情報ファイル名はconnect.txt、境界条件ファイルはbc.txt、入力パラメータファイルはinput.txtで固定されているためそれらの名前をそのまま記述する。また計算格子はdata/grid???fub(???は3桁の数字でブロック番号を示す)、物理量ファイルはdata/cellPhys???fubと言う風に記述される。多次元配列のデータ構造に関しては4次元配列の前3つがインデックスであり、最後が空間座標(計算格子ファイル)、保存量(計算結果ファイル)であることを記述できる。例えば、計算格子ファイル(data/grid???fub)では4次元のうち、1から3番目までがインデックスで、4番目が空間座標(x,y,z)である、計算結果ファイル(data/cellPhys???fub)では同様に4番目が保存量( $\rho, \rho u_i, E$ )が格納されている、と記述できる。こうすることで、バージョンアップにともなう入出力項目の追加、削除などファイルのフォーマット変更にも柔軟に対応できる。

入力データを必須のものとオプションのものに別けて設定できるため、入力の簡素化や入力洩れを防ぐことができる。また、検索のためのキーワードを手動で入力するのではなく、入力ファイル名とそこにあるキーワードをマッピングファイルに記述することで、予め想定されるキーワードに関して自動抽出する事が可能となっている。

## 2.5 自動化機能

自動化機能として自動リンク機能とロボット機能が実装されている。自動リンク機能とは、自動的にリンクを生成する機能であり、データベースの自動組織化の一機能である。例えば、grid 文書を登録する時には計算格子を作成する基になった geometry 文書へのリンクを登録時に手でユーザーが設定する。この時点では grid 文書から geometry 文書へのリンクしか存在しないが、登録後にシステムが geometry 文書から grid 文書へのリンクを自動的に生成する。こうすることで geometry 文書と grid 文書とで双方向のリンクが形成される。これにより高度な検索機能を実装するのではなくインターネット方式つまり芽づる式に次々にデータを引き出してくる方式によるデータ検索を実現している。

一方ロボット機能は自動処理機能(自動リンク機能も自動処理と言えるが)の事で、ある作業に対して、実行コマンドおよびコマンド実行に必要な入出力を設定ファイルに記述することでシステムが自動的にコマンドを実行し、出力データをデータベースに登録していく機能である。例えば、UPACS では analysis 文書の登録後に、物体に働く力を積分するプログラムである calcAeroCharact を実行し、結果( $C_L$ 、 $C_D$  等)を analysis 文書に追加することを行なっている。計算機資源の問題があるが、原理的にはパラメトリック計算を自動的に実行する事も可能である。

## 3. おわりに

開発した CFD 検証データベースについて紹介した。一通りの開発が終了し、当センター内で現在試験運用を行なっている。実際の運用を行なうことで問題点、改善点を把握し、改修および機能拡張を行ないたい。また本データベースでは効率的な知識共有ということで一般に公開することを目標としている。そのためには登録データの信頼性、品質保証、計算コードの公開に関する方針等運用に関する検討が今後必要と考えている。

## 参考文献

- 1) 高橋匡康、中村絹代. 数値データの TV-安定性検証方式の概念検討. In *Proceedings of Aerospace Numerical Simulation Symposium '98*, pp. 313-318. NAL SP-41, February 1999.
- 2) 相曾秀昭. 数値計算における数値的不都合の解析とその解決法について. In *Proceedings of Aerospace Numerical Simulation Symposium 2000*, pp. 319-324. NAL SP-46, December 2000.
- 3) 口石茂, 渡辺重哉, 高木亮治, 山本一臣. CFD/EFD 統合に向けて航技研における取り組み. In *Proceedings of Aerospace Numerical Simulation Symposium 2001*, pp. 103-108. NAL SP-53, January 2002.
- 4) 高木亮治, 山本一臣, 吉田憲司. NEXST-1 周りの流れの CFD 解析結果の比較-第3回 SST-WS でのコンペティションの結果の詳細検討-. 航空宇宙数値シミュレーション技術シンポジウム 2002 論文集, pp. 81-88. NAL SP-57, 2003.
- 5) 高木亮治, 鈴木教雄. NEXST-1 周りの超音速流解析結果の比較-第3回 SST-WS でのコンペティションの結果の詳細検討(その2)-. 航空宇宙数値シミュレーション技術シンポジウム 2003 論文集, pp. 244-249. JAXA-SP-03-002, 2004.
- 6) 高木亮治, 大西楷平. XML を用いたデータベースの構築. In *Proceedings of Aerospace Numerical Simulation Symposium 2001*, pp. 99-102. NAL SP-53, January 2002.
- 7) R. Takaki, K. Yamamoto, T. Yamane, S. Enomoto, and J. Mukai. The Development of the UPACS CFD Environment. In A. Veidenbaum, K. Joe, H. Amano, and H. Aiso, editors, *High Performance Computing, 5th International Symposium, ISHPC 2003, Tokyo-Odaiba, Japan, October 2003. Proceedings*, Vol. 2858 of *Lecture Notes in Computer Science*, pp. 307-319. Springer, 2003.
- 8) 大谷泰昭, 他. XML を活用した知見情報共有システム - 科学技術計算支援の新しいパラダイム. 富士総研技報, Vol. 7, No. 1, pp. 142-151, 2000.