

ISSN 1346-7328

国総研資料 第410号

平成 19 年 7 月

# 国土技術政策総合研究所資料

TECHNICAL NOTE of  
National Institute for Land and Infrastructure Management

No.410

July 2007

## 水理・水文・水質シミュレーションモデル・ ソフトウェアの開発戦略に関する調査報告書

藤田光一・小路剛志・吉谷純一

Report of the Development Strategy of Software and Simulation Models of  
Hydraulics, Hydrology and Water quality cycles for River and Watershed Management

Koh-ichi FUJITA, Takeshi ORO, and Jyunichi YOSHITANI

国土交通省 国土技術政策総合研究所

National Institute for Land and Infrastructure Management  
Ministry of Land, Infrastructure and Transport, Japan

水理・水文・水質シミュレーションモデル・ソフトウェアの開発戦略に関する調査報告書

藤田 光一\*・小路 剛志\*・吉谷 純一\*\*

Report of the Development Strategy of Software and Simulation Models of  
Hydraulics, Hydrology and Water quality cycles for River and Watershed Management

Koh-ichi FUJITA\*, Takeshi ORO\* and Jyunichi YOSHITANI\*\*

概要

河川・流域管理のための水理・水文・水質シミュレーションモデルとソフトウェアの開発状況について、我が国がかかえる課題を認識した上で、諸外国の動向を把握し、今後とるべき方向性について考察を行った。その過程で得られた種々の情報等を調査記録としてとりまとめた。

キーワード： 水理，水文，水質，シミュレーションモデル，ソフトウェア，フレームワーク，共通基盤，インターフェース，河川，流域，管理

Synopsis

The authors examine the present state of developed software and simulation models of hydraulics, hydrology and water quality for river and basin management in japan in reference to other counties. Subsequently, the authors discuss issues and directions for building a firm but flexible foundations of future development of a kind. This technical note also attaches detailed report on HEC software development.

Key Words :

Hydraulics, hydrology, water quality cycle, simulation model,  
software, framework, platform, Interface, river, watershed, management

---

河川環境研究室\*

River Enviroment Division\*

土木研究所 水災害・リスクマネジメント国際センター\*\*

The International Centre for Water Hazard and Risk Management, Public Works Research Institute\*\*

# 目 次

はじめに	1
1 . 国総研環境研究部におけるこれまでの啓発活動	
1.1 水のモデリングとインターフェイス 2 0 0 3	1-1
1.2 流域水物質循環モデル・ソフトウェア博覧会 2 0 0 4	1-4
1.3 流域水物質循環モデル・ソフトウェア博覧会 2 0 0 5	1-12
2 . シミュレーションモデル・ソフトウェアの共通基盤の構築に向けて	
2.1 共通基盤 - 「汎用型シミュレーションモデル」とは	2-1
2.2 データ構造の標準化	2-3
2.3 インターフェイスの標準化	2-7
2.4 プログラム構造の標準化	2-9
2.5 標準フレームワークの開発言語	2-12
2.6 既存のフレームワークの調査	2-16
2.7 標準フレームワークの開発に向けて	2-27
3 . 海外の水理・水文・水質シミュレーションモデルの開発・運用体制	
3.1 欧州における水理・水文・水質シミュレーションモデルの概要	3-1
3.2 米国における水理・水文・水質シミュレーションモデルの開発体制等に関する現地調査	3-17
4 . HEC ソフトウェアの開発の経緯と現状に関する講演録	
4.1 講演開催の経緯	4-1
4.2 講演録	4-2
参考文献	
謝辞	
研究協力者	
添付 CD-ROM の内容	

## はじめに

日本は亜熱帯、モンスーン地帯、亜寒帯の幅広い気候帯に属し、大きな季節変動を伴う降雨と、急峻な地形、沖積平野に資産の大部分が存在するという国土条件の中で緻密な国土管理を必要とする。このため、水の適正な管理が継続的に実施されることが重要であり、そのような認識の下、これまでも水管理行政が精力的に進められて来た。ダム、堤防などの防災のための施設整備に加え、今後は、これまでに整備してきた水管理のためのインフラの資産を適切に維持・管理するとともに、一層合理的かつ有効に運営することも重要になってきている。

河川・流域管理に関わるニーズの多様化、高度化に応じて、あるいは気候変動などの新たな状況に対応するために、技術政策の立案過程においては、様々な知見や情報の統合・集約化や、それらに基づく将来予測等を適切かつ効率的に行い、理に適った議論に資する情報を生産・編集・提供することの必要性が益々高まっている。これに伴い、このような作業を支援し、政策検討に係わる労力の合理化に資するシミュレーションモデル・ソフトウェア（以下、「モデル・ソフト」と略記する）の開発・普及が望まれている。また、住民参加による政策説明・検討の場での効果的な情報提供の手段も求められている。

水理、水文、水質現象にかかわるモデル・ソフトは、研究機関、コンサルタントが個別に作成し、これを行政機関が、技術政策検討、意思決定や住民への情報提供に利用しているのが実態であり、現時点において、個々のモデル・ソフトの評価や体系化が統合的に行われている状況にはない。また、標準的モデル・ソフトの開発・利用も必ずしも広範に行われている状況にはない。

水文・流出現象について見ると、降雨、蒸発散、表面流、河川流、地下水流など多くの素過程が関係しており、それぞれが複雑な関係を持ち、水収支関係を形成する。我が国ではこれまで、個別の現象に対する解析モデルが数多く開発されてきており、高度あるいは高精度な解析モデルも多数存在する。しかし、機関によって開発環境や稼働環境などが異なり、プログラミング言語や解析手法、データの入出力フォーマットなどもそれぞれ異なり、開発者の数だけモデル・ソフトの種類が存在するとも言える状況にある。したがって、他機関で開発されたモデル・ソフトを取り入れて新たなモデル・ソフトを構築しようと考えたとき、他機関が作成したソースコードを読み下し、理解する必要があることが一般的である。そうした作業は非常に煩雑なものになるため、現実的には実行されることはあまりなく、新たにモデル・ソフトを一から再構築せざるを得ない状況もしばしば見られる。以上について、課題を認識・整理するとともに、今後の展開の方向を具体的に検討していくことが大切である。

一方、欧州のいくつかの国や米国においては、当該分野のモデル・ソフトの開発と維持・管理・更新、利活用について、相当程度戦略的あるいは長期的な取り組みが、かなり前からなされてきており、そのような動向を把握することは、我が国の取り組み方を考える上でも不可欠と言える。デンマーク水理研究所の MIKE シリーズや陸軍工兵隊水文工学センターの HEC シリーズはこの代表例である。

以上のような認識は、1990年代前半以降、実は、既に多くの人の間で醸成されてきている。たとえば筆者の1人も、米国・欧州3カ国の河川・水理関係研究所の戦略を比較分析し、ソフトの開発・運用の戦略が（当時の我が国に比べ）高いプライオリティを持っていることを指摘している<sup>1)</sup>。また、我が国においても、このような認識を背景にした具体的に取り組みが始まっており、90年代半ばの「河道計画シミュレータ」の開発着手から発展した「河川計画シミュレータ」（（財）国土技術研究センターが著作権を保有するフリーソフトウェア）はその例と言える。京都大学の椎葉研究室で開発されたオブジェクト指向型水文モデリングシステム OHyMoS(次章以降で詳述する)は、その開発が1990年代初頭から始まっており、上記の認識に基づくと言うよりも、それに先駆けた日本独自の取り組みと位置づけられる。こうした具体的な取り組みは別にしても、1990年代後半以降は、モデル・ソフトのあり方に高い関心を持つ、あるいは我が国の状況に懸念を持つ専門家等の間で度々議論がなされ、今日に至っている。こうした動きが、しかし、統合的かつ継続的に発展する状況にまでは至っていない。

国土技術政策総合研究所環境研究部（担当研究室は河川環境研究室）では、過去の取り組みや議論を踏まえ、課題の認識・整理を越えて、具体的にどのような取り組みを今後行っていくべきかの全体像を検討することが今最も求められていると考え、平成14年度後半から～平成18年度にかけて、土木研究所ユネスコセンター防災チームと一部連携しつつ、水理、水文、水質に関するモデル・ソフトを対象として、政策検討への活用という観点から、優れたモデル・ソフトを構築するための体制、開発仕様等の立案や、既存モデル・ソフトの適切な利用促進のための方策の立案を試みた。さらに、モデル・ソフト開発にかかわる汎用的仕様に関する技術的検討として、モデル・ソフトの構造や、プログラムインタフェース、データインタフェースなどの枠組みの検討を行った。本資料はその現時点における結果を、検討に際して収集・整理した主要な情報とあわせ、まとめたものである。

言うまでもなく、本課題は、一研究部の力だけで対応できるものではなく、本課題に関する高い見識や高度な専門的知見を有する、あるいは地道な取り組みを続けてきた技術者・研究者等、さらにはユーザーやクライアントである河川管理者等が持つ知見の結集が必要である。以下の内容は、それらの方々の協力・参画により中味が作られたものといえ、本資料の著者は、そうした一連の取り組み成果の報告役と位置づけるのがより適切である。そのような意味を込めて、協力・参画いただいた方々については、本資料において、その具体的形と合わせ、内容に即してその都度記述している。

1章では、流域管理における水理・水文・水質モデルに関する研究、開発、利用等に関する現状、将来のあり方について、これまで国総研環境研究部がコーディネートしつつ取り組んできた活動について概説する。この過程で、多くの貴重な意見・情報や方向性について重要な示唆をいただき、また、検討を進める上での人的ネットワークを作ることができた。この成果は、直接的ではないが、3章、4章の内容と並んで、論点や課題の構図を明確化し、2章の内容を作る上でのバックボーンとなった。

2章では、水理・水文・水質モデル・ソフトの開発における共通の基盤づくりに向けて、必要な条件を(1)データ構造の標準化、(2)インターフェースの標準化、(3)プログラム構造の標

準化の3点に整理する。その中でプログラム構造の標準化を中心に議論の進め、モデル・ソフトを構成するコアエンジンのシステム部分であるモデルフレームワークをオブジェクト指向型言語で記述することを前提として必要な仕様の要件について提示する。ここが、本資料の提案としての中核部分となる。

3章では、欧米における水理・水文・水質モデル・ソフトの開発・運用体制に関する調査結果を取りまとめる。はじめに欧州における代表的な商業的シミュレーションモデル・ソフトについて取り上げるとともに、欧州の主要なシミュレーションモデル・ソフトを統合するためのフレームワークを構築する最近のプロジェクトについて整理する。次に米国におけるシミュレーションモデル・ソフトの開発体制等に関する現地調査結果(平成17年11月実施)を取りまとめる。

最後に4章では、平成17年3月に開催された「HEC 開発及び維持管理体制に関する講演会」の講演録を掲載する。この講演は、著者の一人である吉谷(独)土木研究所ユネスコセンター設立推進本部(当時)が主導的に企画したものであり、一連の検討にとって重要な情報になるという考えから、講演の運営を国総研環境研究部がサポートした。

なお、本資料の内容にかかわる検討は、平成14~16年度については、国土技術政策総合研究所プロジェクト研究「自然共生型流域圏・都市の再生」の一環として、平成17年度は河川事業調査費・河川総合開発事業調査費課題「河川・流域管理を支援するシミュレーションモデルの開発と普及に関する研究」、18年度については、河川事業調査費・河川総合開発事業調査費課題「水理・水文・水質シミュレーションモデルの開発戦略に関する検討」の取り組みとして行ったものである。

## 1 国総研環境研究部におけるこれまでの啓発活動

### 1.1 水のモデリングとインターフェイス 2003

#### 1.1.1 開催趣旨

我が国では、流域管理における水理水文に関するモデルについて様々な研究がなされてきた。しかしながら、その研究成果は、施策の立案や住民を含めた合意形成等に広く活用されている欧米と比較すると、民間や行政に十分活用されているとは言い難いのが現状である。一方、水環境に対する国民の意識が向上しつつある中、行政においても流域の総合的な管理の重要性が高まっている。

そこで、水文・水資源学会（企画事業委員会）と国土交通省国土技術政策総合研究所（環境研究部河川環境研究室）を事務局として、平成15年3月26日（水）に中央大学後楽園キャンパスにてシンポジウム「水のモデリングとインターフェイス2003～水理水文水質ソフト～」が開催された（主催 シンポジウム「水のモデリングとインターフェイス2003」実行委員会）。図1-1に示すプログラムのように、流域管理における水理水文水質ソフトに関する研究、開発、利用等の現状について、産官学各分野の専門家の方々を招いて広く情報交換し、将来のあり方について意見交換を行う場として企画された。なお、本章における講演者等の肩書きは開催当時のものである。

（シンポジウム「水のモデリングとインターフェイス2003」実行委員会事務局）

山田 正	中央大学工学部教授
椎葉 充晴	京都大学工学部教授
平野 廣和	中央大学総合政策学部教授
立川 康人	京都大学防災研究所助教授
吉川 勝秀	国土交通省国土技術政策総合研究所環境研究部長
安田 佳哉	国土交通省国土技術政策総合研究所環境研究部河川環境研究室長

#### 1.1.2 講演概要

基調講演の概要を下記に示す。

水理水文水質ソフトの構造とインターフェイスについて（京都大学教授 椎葉充晴）：ソフトの統合のための仕様を決めた方がよい。また、利用者間で情報の交換をする組織づくりが必要である。

統合型水理水文水質ソフト MIKE シリーズの新しい水のモデリングと意志決定支援への取り組み（デンマーク水理・環境研究所水資源部門担当重役 Karsten Havn）：エンジンの開発よりもグラフィカル・ユーザー・インターフェイス（Graphical User Interface: GUI）の開発に時間がかかるようになってきた。GUIの開発、GISとのカップリング、モデル間のカップリングを進めるためには大規模な開発努力が必要であり、国際協調が必要である。

時 間	議 事 内 容
13:00	<b>開会の辞</b> シンポジウムの開催にあたって 水文・水資源学会会長（九州大学 教授） 楠田 哲也
13:05	<b>趣旨説明</b> 「なぜ今、水理水文水質ソフトか」 中央大学 教授 山田 正
13:20 ~ 14:50	<b>基調講演</b> 「水理水文水質ソフトの構造とインターフェイスについて」 京都大学 教授 椎葉 充晴  「統合型水理水文水質ソフト MIKE シリーズの新しい水のモデリングと意思決定支援への取り組み」 DHI（デンマーク水理研究所 水資源部門担当重役（取締役） Karsten Havn  「米国の水理水文学的モデリングと水文工学センターの展望」 US Army Corps of Engineers - HEC（陸軍工兵隊水文工学センター） 水理水文技術部 部長 Arlen D. Feldman
14:50	<b>休 憩</b> （10分間）
15:00~	<b>講 演</b> 「利用者からみた水解析モデルのあり方」 独立行政法人土木研究所 水工研究グループ 水理水文チーム 上席研究員 吉谷 純一  「自然共生型流域圏・都市再生における水物質循環のモデリングについて」 国土技術政策総合研究所 河川環境研究室長 安田 佳哉  「オープンソースとしての公開型水理水文水質ソフトのあり方」 京都大学 防災研究所 助教授 立川 康人  「データの標準化から見た水理水文水質ソフトのあり方」 国土交通省 河川局 河川情報対策室長 佐藤 宏明
16:20	<b>休 憩</b> （10分間）
16:30~	<b>パネルディスカッション</b> 「水理水文水質ソフトの現状と今後のあり方」 ・コーディネーター 山田 正（中央大学 教授） 吉川 勝秀（国土技術政策総合研究所 環境研究部長） ・パネリスト 椎葉 充晴（京都大学 教授） 吉谷 純一（独立行政法人土木研究所 水工研究グループ水理水文チーム 上席研究員） 佐藤 宏明（国土交通省 河川局 河川情報対策室長） 都丸 徳治（建設コンサルタンツ協会 常任委員長） Karsten Havn（デンマーク水理研究所 水資源部門担当重役（取締役））
17:30~	<b>質疑応答</b> （20分間）
17:50~	<b>総 括</b> 京都大学 防災研究所 助教授 立川 康人
18:00	<b>閉 会</b>

図 1-1 シンポジウムプログラム

米国の水理水文学的モデリングと水文工学センターの展望( 米国陸軍工兵隊水文工学センター水理水文技術部長 Arlen D. Feldman ) : 当センターではソフトを開発、管理、配付しているが、サポートは民間のエンジニアリング・ベンダーに対してのみ行っている。ユーザーは質問があればベンダーに質問するが、ベンダーが対応できない場合はセンターがベンダーをサポートしている。

次に、4人の講演の概要を示す。

利用者からみた水解析モデルのあり方( 独立行政法人土木研究所水工研究グループ水理水文チーム上席研究員 吉谷純一 ) : 多くのモデルの中からどのモデルを使った方がいいかというクライテリアを示す必要がある。

自然共生型流域圏・都市の再生における水物質循環のモデリングについて( 国土交通省国土技術政策総合研究所環境研究部河川環境研究室長 安田佳哉 ) : 国総研のプロジェクト研究「自然共生型流域圏・都市の再生( H14～16 )」で取り組んでいる水物質循環モデルの開発に関する取り組みについて紹介した。

オープンソースとしての公開型水理水文水質ソフトのあり方( 京都大学防災研究所助教授立川康人 ) : 開発者はソースコードを公開し、利用者はサブルーチン、クラスを提供し、ウェブを通じたコンソーシアムのようなものを形成できるとよい。

日本の河川情報の現状と今後～データ構造の標準化とソフトについて～( 国土交通省河川局河川情報対策室長 佐藤宏明 ) : 河川局におけるデータ構造の標準化に向けた取り組みについて紹介した。

## 1.2 流域水物質循環モデル・ソフトウェア博覧会 2004

### 1.2.1 開催趣旨

水・物質循環モデル・ソフトウェアは、流域における水・物質循環現象の総合的な理解、各主体による活動や施策の評価、住民をはじめとする参加主体間の合意形成等を支援するものであり、政策の検討と推進等において、今後ますます重要になっていくものと考えられる。その一方で、モデル・ソフトウェアについては、これらが現地に適用された場合の信頼性・適用性等が十分明らかにされていないこと等から、十分な活用が図られていないのが現状である。また、大学、国の研究機関、民間等様々な主体により、エンジンおよび付随するビジュアルな表現技術、データの蓄積・保存技術等の開発がなされてきたものの、それら技術的ノウハウが開発主体間で共有化、標準化されておらず、国内におけるモデル・ソフトウェア開発が効率的でないといった指摘もされている。

このような背景から、平成16年4月23日(金)に中央大学駿河台記念館において、この種の企画としては日本初となる流域水物質循環モデル・ソフトウェア博覧会 2004 が開催された。その目的は、モデル・ソフトウェアに係わる様々な技術の研究・開発状況や、メニュー、技術水準、実現象への適用性等の情報を共有することにより、政策担当者、研究者等のモデル・ソフトウェアに対するニーズの掘り起こし、モデル・ソフトウェア開発のインセンティブの形成、開発・維持管理体制を含めたモデル・ソフトウェアのあり方等に関する議論等を行うことである。なお、当日の様子は同日夕方のNHKニュース(関東ローカル)でも取り上げられた。

### 1.2.2 博覧会の実施体制

本博覧会は水文・水資源学会の主催、国土交通省国土技術政策総合研究所の共催、(独)土木研究所、アジア学術会議、(社)土木学会、(社)日本水環境学会、(財)河川情報センター、(社)建設コンサルタンツ協会の後援で開催された。

国総研は国土管理に係わる技術の開発・普及促進の観点から参画した。博覧会の開催にあたっては、中央大学山田教授、京都大学椎葉教授を中心に、主催、共催、後援団体の代表者からなる事務局(流域水物質循環モデル・ソフトウェア博覧会 2004 事務局)が設けられ、博覧会の企画、運営方法や博覧会で議論すべき論点等について検討を進めてきた。その成果が次項に示すアクションプランである。

(流域水物質循環モデル・ソフトウェア博覧会 2004 事務局)

山田 正	中央大学工学部教授
椎葉 充晴	京都大学工学部教授
平野 廣和	中央大学総合政策学部教授
立川 康人	京都大学防災研究所助教授
藤田 光一	国土交通省国土技術政策総合研究所環境研究部河川環境研究室長
末次 忠司	国土交通省国土技術政策総合研究所河川研究部河川研究室長
吉谷 純一	独立行政法人土木研究所ユネスコセンター設立推進本部 水災害・リスクマネジメントチーム上席研究員
中尾 忠彦	財団法人河川情報センター理事



ソフトウェアの評価に関しては、ソフトウェアの根幹をなす理論モデルや数値計算手法等の基本原理にまで踏み込んで評価する必要があり、十分な知識を有する学識者等が代表して評価を行い、その結果を各主体で共有するのが効率的である。

ソフトウェアの適用性や精度に関する評価結果の情報共有は、利用者側にメリットがあるだけでなく、開発者側にとっても開発インセンティブを高めるものになり得る。

公平・公正かつ効率的に評価作業を行うためには、具体的な評価基準、評価手法を設けるとともに、ソフトウェアの検証システム（入力条件と結果の真値）の整備が必要となる。

#### 流域データの共有化

水物質循環の解析には、水文・水理・水質データや地形・地質、人間活動等様々なデータからなる流域データが必要となるが、これらデータは一括管理されておらず、対象流域において、このようなデータを収集することは容易でないのが現状であり、このことがソフトウェアの活用を妨げる要因の一つとなっている。また、一旦取得されたデータは貴重な財産であり、様々な主体により最大限活用されることが望ましい。このため、様々な主体により整備されたデータを共有化し、総合的な流域情報として提供するシステムを構築することは、ソフトウェアの普及とデータの有効活用の点から重要である。

各主体が所得した取得したデータを一括保管・管理し、適宜利用者に提供できるようなシステム（流域データライブラリー）が望まれる。近年、水情報国土のように国土情報のデータベース化が進められており、これらプロジェクトとの連携が重要である。

精度の低いデータを排除し、データの品質を確保するためには、データの評価制度や検証システムが必要である。

また、流域データの集積・管理やソフトウェアによる利用等を効率化するため、データフォーマットの規格化も望まれる。

段階的に、データの所在を明らかにするクリアリングハウス、データとデータの読み方を併せてやりとりするルールづくり等も考えられる。

## （２）開発面

### 合意形成を支援するためのインターフェイスの高度化

国土管理、水管理への住民参加が進む中、特に専門的知識を有しない者による水物質循環現象の正しい理解を図り、各主体間の合意形成を支援することがソフトウェアに期待されている。このため、ソフトウェアの操作性を高めたり、現象の過程や計算結果を分かりやすく表現するインターフェイスの整備・充実を図ることにより、ソフトウェアの合意形成ツールとしての機能を高めることも、ソフトウェアの普及のため重要である。

ソフトウェアの利用者を広げる効果もある。

大学等における学習教材としての活用も考えられる。

### 技術情報の共有化、組み合わせによるソフトウェアの高度化

ソフトウェアの開発はこれまで個々の主体により行われ、他者より優れたソフトウェアを構築することが、各者（企業、研究者）にとっての開発インセンティブとなってきた。このため、開発に係わるノウハウは共有化されることなく、多種多様なモデルが構築されてきた

と言える。汎用性に富み、性能の優れたソフトウェアを提供していくためには、開発に係わる技術情報の共有化による開発効率の向上や、統一基準に基づくソフトウェアの組み合わせ、相互補完も考えられる。

技術情報を公開するためには、プログラム等に係わる知的財産権等保護制度の確立を待つ必要がある。

組み合わせ部分について統一規格を決めておけば、基本ソフトウェアと特徴的な現象に特化した様々なサブソフトウェアの組み合わせが可能になり、信頼性と汎用性の確保が可能となる。

開発者間の議論が必要。

### 3．ソフトウェアの普及・開発のための方策

上記の課題と対応方針を受け、より効果、実現性が高いと考えられるものについて具体的方策を提案する。

#### (1) ソフトウェアライブラリーの構築(案)

##### 目的

国内外の既存のソフトウェアに係わるメニューや特性、機能に関する情報を開発者が提供し、それら情報を一括して登録・公開する制度を設けることにより、利用者による目的に応じた最適なソフトウェアの選定を支援する。また、その一方で、開発された個々のソフトウェアが社会的に認知されること、利用者が求めるソフトウェアの諸特性が明らかになることによる、開発者側の開発インセンティブの高まりを期待する。

##### 方法

#### 1) ソフトウェアライブラリーの運営組織

ソフトウェア情報の登録・公開を運営する組織を設ける。各主体の利害に係わる情報を扱う可能性があるため、中立性・公益性の高い機関に設ける。

#### 2) 登録・公開方法

・開発者があらかじめ定められた様式に基づき、ソフトウェアに係わる(概略及び詳細)情報を有料で登録する。

・登録は情報提供サービスに載せることを意味するものであり、当該組織がそのソフトウェアの品質を保証することを意味しない。

・WEB上で、各ソフトウェアの概要情報を掲示するとともに、利用者からの求めに応じて詳細情報を有料で提供する。

#### 3) 提供する情報の範囲

・概要情報は、ソフトウェアの名称、開発者、諸機能、必要なデータの種類、価格等とし、WEB上で検索が可能なシステムとする。

・詳細情報としては、ソフトウェアの理論モデル、数値計算手法、適用事例、計算結果等(実測値との比較)とする。

・海外も視野に入れ、英語バージョンも作成する。

・ソフトウェアのバージョンアップ、使用実績の追加等に伴い、適宜登録情報を更新する。

・ソースコードに関しては、知的財産権保護に係わる議論の結果を待つが、開発者が希望する場合は、詳細情報に加えことも可能。

#### 課 題

・運営体制の整備、運営資金の確保。  
・より有益な情報提供とするためには、(2)で論じる第三者によるソフトウェアの評価結果があることが望ましい。

### (2) ソフトウェアの評価システムの構築(案)

#### 目 的

ソフトウェアに係わる諸情報の公開に加えて、ソフトウェアの性能評価情報(適用範囲や精度、信頼性等)は、利用者がソフトウェアを選定するに当たってより有益な情報となり得る。このため、学識者等による評価システムを設け、ソフトウェアの適用範囲、計算精度を評価し、ソフトウェア選定の参考情報として提供する。高い評価を受けたソフトウェアは、ユーザーに活用される機会が増えるため、開発者側の開発インセンティブも高まると考えられる。

#### 方 法

##### 1) ソフトウェアの評価委員会の設置

・ソフトウェアに関する十分な知識・経験を有し、中立性の高い学識者等からなる評価委員会を組織。

##### 2) ソフトウェアの評価方法

・開発者がソフトウェアを持ち込み、評価委員会が評価を行う。  
・評価に係わる作業及び諸費用は基本的に開発者が負担。  
・評価だけでなく、改良等に関するアドバイスも行う。

##### 3) 評価結果の公開

・評価結果の公開・非公開は開発者が判断。  
・開発者の要望に基づき、WEB等で公開。

#### 課 題

・評価項目、効率的かつ公正な評価基準、評価方法等の立案。  
・ソフトウェアの検証用データベースの作成

(対象流域の選定、流域水循環(気象、水文、水理、水質等)、地質・土壌、人間活動等のデータの収集とデータベース化。必要に応じて補足調査、一斉観測プロジェクト等)。

### (3) 流域データライブラリーの整備(案)

#### 目 的

水理、水質データ等各主体により取得された流域データを集積、提供することにより、解析に必要なデータの収集や、利用可能なデータの範囲等に応じたソフトウェアの選定を支援する。また、様々な分野における流域データの有効活用も図る。



時 間	内 容
10:00	<b>1. 開会挨拶</b> 「流域水物質循環モデル・ソフトウェア博覧会 2004 の開催にあたって」 水文・水資源学会会長（代理） 實 馨（京都大学教授） 国土交通省国土技術政策総合研究所環境研究部長 福田 晴耕
10:10	<b>2. 趣旨説明</b> 「流域水物質循環モデル・ソフトウェア博覧会 2004 の開催目的」 水文・水資源学会企画事業委員会委員長 山田 正（中央大学教授）
10:30	<b>3. 基調講演</b> 「水物質循環モデル・ソフトウェアの利用・普及のために」 水文・水資源学会副会長 椎葉 充晴（京都大学教授）
11:00	<b>4. 「流域水物質循環モデル・ソフトウェア博覧会」</b> 各ブースでモデル・ソフトウェアの展示 展示モデルの概要を大スクリーンにて紹介
15:00 ～17:00	<b>5. 公開討論</b> 「流域水物質循環モデル・ソフトウェアの今後の展開」  コーディネーター：水文・水資源学会企画事業委員会委員長 山田 正（中央大学 教授） 水文・水資源学会副会長 椎葉 充晴（京都大学 教授）
18:00	<b>6. 閉会</b>

図 1-2 プログラム

また「できることから始める」をキーワードに、ソフトウェアライブラリーの構築、ソフトウェアの評価システムの構築、流域データライブラリーの整備の3つの具体的方策が提案された（前項参照）。

## 2) ブース展示

会場に設けられたブースにおいては、大学、研究機関、財団、海外企業を含む民間企業等計35者から、41ブースを使用してモデル・ソフトウェアや関連する技術の出展が行われた。出展物は、水物質循環モデルを中心に、理論モデルからリモートセンシング等による計測技術、データ作成技術まで幅広い分野の技術から構成され、各ブースにおいてパソコンを使用したシミュレーションの実演やパネルにより説明が行われるとともに、意見交換等が熱心に行われた。



写真 1-2 ブース展示

### 3) 公開討論

ブース展示と並行して、メインホールにおいては、山田教授、椎葉教授をコーディネーターとして、参加者全員で公開討論が行われた。公開討論に先立って、山田教授、椎葉教授から開会式の趣旨説明、基調講演を踏まえて、再度論点が提示され、会場に集まった参加者からは、開発者、利用者等それぞれの立場からの意見が寄せられた。

主な発言要旨は以下の通りである。

モデルの計算結果に対する信頼性の確保と議論の効率性を高めるためには、モデルの汎用化が必要である。ただし、一つのモデルで汎用化を図るには限界があり、要素となる個々のプログラムを繋ぎ合わせて、目的に応じたモデルを構築するというシステムが必要となる。

優れたモデルの開発や、モデルを実現象に当てはめるために適切な諸定数を設定するためには、良質な流域データが必要である。このようなデータの取得、収集は個々の研究者では困難であり、政府機関で行ってほしい。

データの効率的な管理、有効活用のためには、基本データについてのデータ構造の標準化やデータとデータ仕様を併せてやりとりするルールづくりが重要である。

現在の日本のモデルは、エンジン(演算プログラム)では多様で優れている部分があるものの、インターフェイスは、欧米のものに対して遅れており、海外での競争力がない。産官学が連携し、優れた日本版モデルを開発し、アジアモンsoon地域の水物質循環管理をリードすべきである。



写真 1-3 公開討論

## 1.3 流域水物質循環モデル・ソフトウェア博覧会 2005

### 1.3.1 開催目的

2004年に開催された「流域水物質循環モデル・ソフトウェア博覧会2004」に引き続き、エンジンをはじめとするモデル・ソフトウェアに係わる様々な技術の現時点における研究・開発状況や、メニュー、技術的水準、実現象への適用性等の情報を共有することにより、政策担当者、研究者等のモデル・ソフトウェアに対するニーズの掘り起こし、モデル・ソフトウェアの開発インセンティブの形成、開発・維持管理体制を含めた流域水物質循環モデル・ソフトウェアのあり方等に関する議論等を行うために、平成17年12月12日(月)に中央大学駿河台記念館において、「流域水物質循環モデル・ソフトウェア博覧会2005」が開催された(主催:水文・水資源学会、共催:国土交通省国土技術政策総合研究所、日本学術会議大気・水圏科学研究連絡委員会陸水専門委員会、後援:(社)土木学会、(社)日本水環境学会、(独)土木研究所、(財)河川情報センター、(社)建設コンサルタンツ協会)。プログラムを図1-3に示す。

時間	内容
12:30	1. 開催挨拶 「流域水物質循環モデル・ソフトウェア博覧会2005の開催にあたって」 水文・水資源学会会長 池淵 周一(京都大学教授) 国土交通省国土技術政策総合研究所環境研究部長 福田 晴耕
12:40	2. 趣旨説明 「流域水物質循環モデル・ソフトウェア博覧会2005の開催目的」 水文・水資源学会理事 山田 正(中央大学教授)
12:45	3. 「流域水物質循環モデル・ソフトウェア博覧会」 各ブースでモデル・ソフトウェアの展示 展示モデルの概要を大スクリーンにて紹介
16:00 ~17:30	4. 公開討論 「流域水物質循環モデル・ソフトウェアの今後の展開」 コーディネーター:水文・水資源学会理事 山田 正(中央大学教授) 水文・水資源学会副会長 椎葉 充晴(京都大学教授)
17:30	5. 引き続き閉会まで各ブースでモデル・ソフトウェアの展示
18:30	6. 閉会

図 1-3 プログラム

### 1.3.2 博覧会の概要

博覧会では、開会式、ブース展示及び参加者全員による公開討論が行われた。

#### 1) 開会式

開会式においては、水文・水資源学会長の池淵京都大学教授と福田国土技術政策総合研究所環境研究部長による開会挨拶の後、山田中央大学教授による博覧会の趣旨説明があり、モデルの標準化の観点を中心に問題を提起するとともに、河川局の藤山河川情報対策室長からの水理標準ソフトウェアの構築に向けた考えを示したレポートについて紹介した。



写真 1-4 開会の挨拶（左：池淵教授、右：福田部長）

#### 2) ブース展示

会場に設けられたブースにおいては、昨年度と同様に大学、研究機関、財団、海外企業を含む民間企業等計 40 者からモデルや関連する技術の展覧が行われた。各ブースにおいてパソコンを使用したシミュレーションの実演やパネルにより説明が行われるとともに、意見交換等が活発に行われた。当日、展示されたモデル・ソフトウェアの概要の説明したパンフレットについて、翌年 4 月に開催されたアジア学術会議にて英語版小冊子としてアジア諸国の政府系機関および開発系銀行等の諸機関・組織に配布された。



写真 1-5 ブース展示

### 3) 公開討論

昨年度と同様に、山田教授、椎葉京都大学教授をコーディネーターとして、参加者全員で公開討論が行われた。公開討論に先立って、藤田国総研河川環境研究室長から昨年度開催した博覧会で示したアクションプランとその時の討論の概要の説明、中尾河川情報センター理事から「水管理と情報」をテーマにこれまでの水情報国土におけるデータ提供の現状、インターフェース標準化の動向に関する紹介、小川河川情報センター審議役から米国におけるモデルに関する調査の報告(3.2参照)が行われた。その上で山田教授、椎葉教授から標準ソフトの要件やモデリングシステムの必要性などの論点が提示され、会場に集まった参加者からは、開発者、利用者等それぞれの立場からの意見が寄せられた。

主な発言要旨は以下の通りである。

- 国と国民が、科学技術レベルで議論し、理解し合える基礎が必要
- 標準ソフトと共に、基本的なデータが計測され、データベース化され、提供されなければならない
- コンサルタントは、もっと政策レベルの検討、提案が必要になる
- “分布型モデルを 流域に適用しました”～個別のモデルを個別の流域に適用されている～ツールの共通化が必要
- 共有できる部分は、共有して、変えた部分で、議論すべき
- モデルと合わせて、データの標準化も、同じ枠組みで進めて欲しい
- フレームワークだけでは、使われない、進まない可能性があるのではないか
- 標準ソフトを使えば結果は出せるが、基礎理論の習得は同時に行わなければならない



写真 1-6 公開討論



もちろん、概念的にはこのような方式が良いとしても、実現性について具体的な見通しが無ければ議論にとどまる。以下では、「汎用型シミュレーションモデル」を実現する方法を検討した内容を1つの提案として示し、モデル・ソフトの共通基盤づくりに向けた具体的動きにつなげることをねらう。なお、モデリングのシステムの共通化が重要であっても、その下で動く個々のモデル・ソフト（要素モデル）が具体のニーズに直接応えるツールであることには変わりはなく、本章の内容は、個々のモデル・ソフト（要素モデル）の存在を前提する。また、「汎用型シミュレーションモデル」を構築する際に、その下で動くモデル・ソフト（要素モデル）を想定する必要がある、その意味でも、代表的なモデル・ソフト（群）の存在は「汎用型シミュレーションモデル」と車の両輪をなすと言える。さらに、既存のモデル・ソフトの資産を生かすことも重要な検討対象となる。

汎用型シミュレーションモデルの構築に向けて、大きく(1)データ構造の標準化、(2)インターフェースの標準化、(3)プログラム構造の標準化を満たすことが求められる。そのつながりのイメージを図 2-2に示す。①～⑩のそれぞれの要求事項は、「1)流域水循環解析モデルフレームワーク」、「2)データ構造標準化」、「3)河川GIS・アプリケーションインターフェース」、「4)ユーザーインターフェース」、「5)運用、維持管理の枠組み」の中の事項として位置づけられる。なお、要素モデルに関する詳細な説明は 2.6.7にて行う。

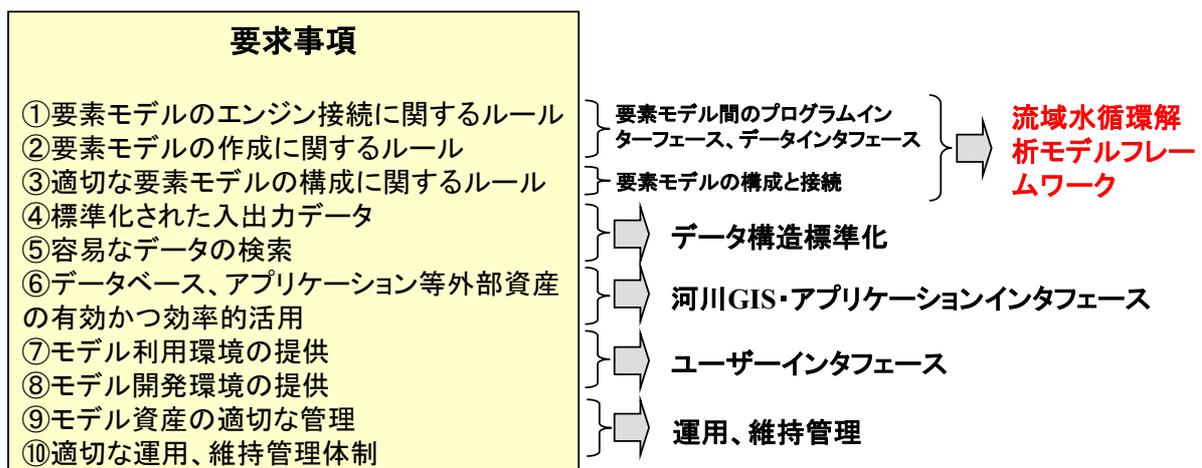


図 2-2 汎用型シミュレーションモデル

## 2.2 データ構造の標準化

現在、日本における河川に関する電子データの標準化状況は、図 2-3に示すようにデータの電子化、公表は進んでいるが（例：水文水質データベース、水路網データガイドライン）、水理・水文・水質シミュレーションに関するデータの公表は進んでおらず、データは存在しているものの、多種多様なデータ構造であり、それらは散在し、統一的に管理されていない。

これらの問題を解決するには、利用者が必要とするデータについて、標準化ガイドラインを策定し、それらを運用するルール、仕組みを構築する必要がある。

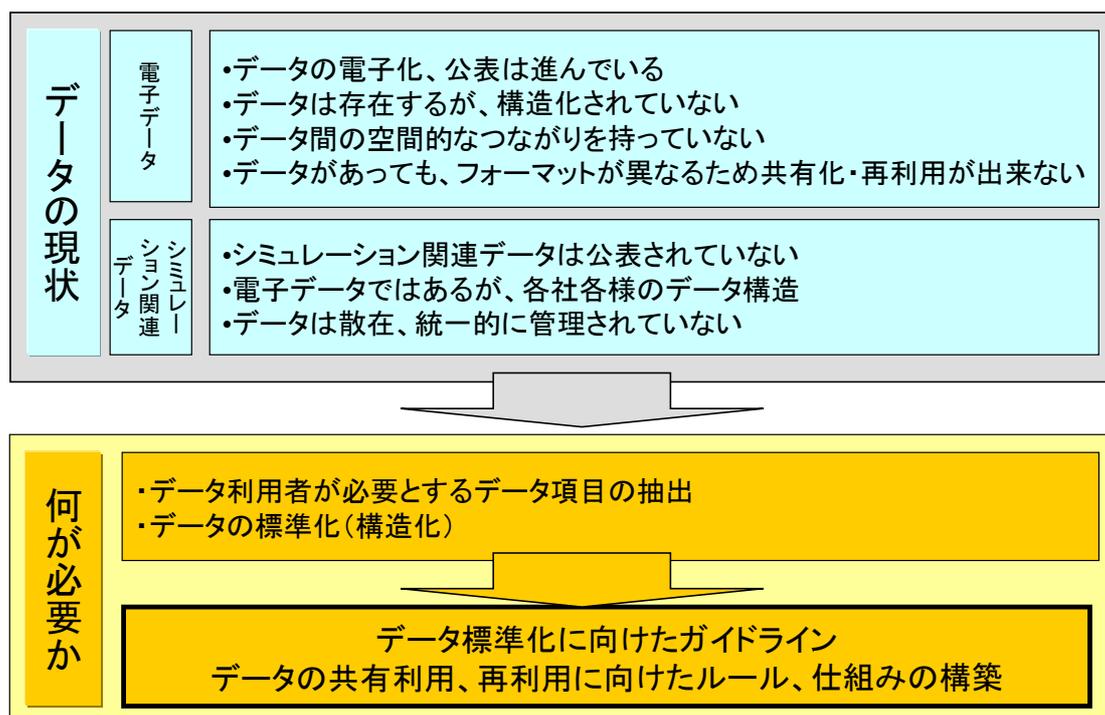


図 2-3 データの現状と標準化の必要性

データの構造化、標準化について、以下のように定義する。

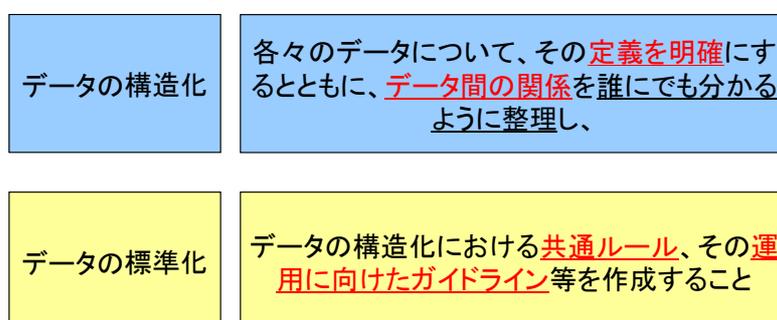


図 2-4 データの構造化と標準化の定義

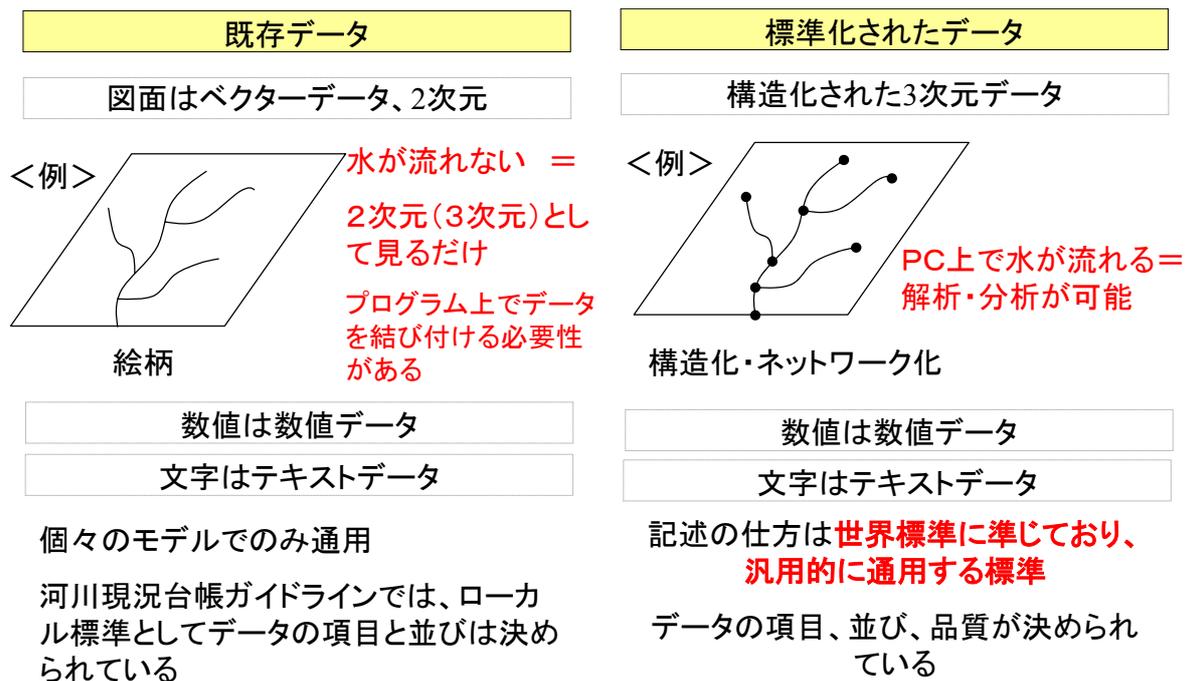


図 2-5 既存データと標準化されたデータの違い

これまで、標準化データとして作成されてきたデータは、例えば、河川を表現する線データがあったとしても、絵として河川が表現されているのみで、“意味”として河川を表現しているわけではなく、また、河川・水路の分合流についても、そこに分流、合流の意味が付与されているわけではないので、既存データを用いて、水理解析を行う際には、改めて、河川・水路としての、あるいは分合流の意味を付与しなければならない。

これでは、標準化データとして定義、整備されたとしても、実際の利用に供する際には様々な加工を施さなければならず、利活用の範囲は限定されてしまう。また、同じデータを利用しても、解析や検討の結果は異なるものになってしまう。

これらの理由から、河川に関するデータの構造化・標準化では、河川・水路の意味、分合流の意味を付与されたデータ、すなわち解析等の実利用に供することが可能なデータを整備する方針で作業を進めており<sup>2)</sup>、具体的には、次図に示す内容を持ったデータを整備している。

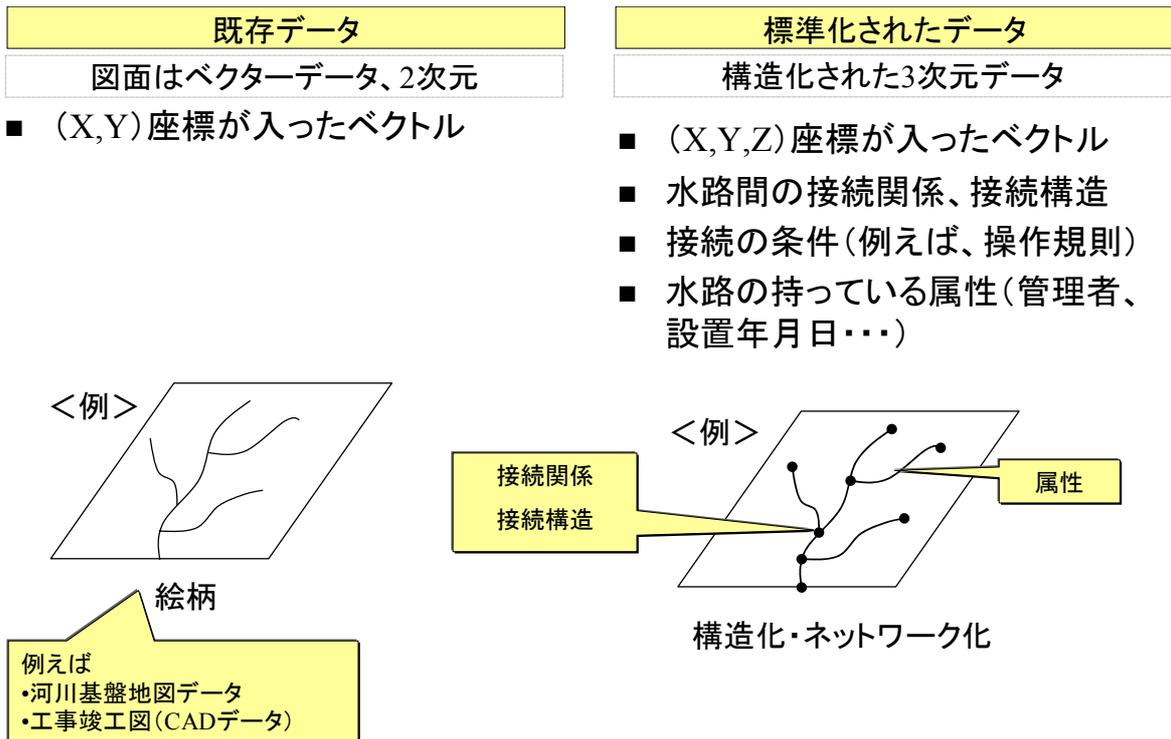


図 2-6 既存データと標準化データの内容

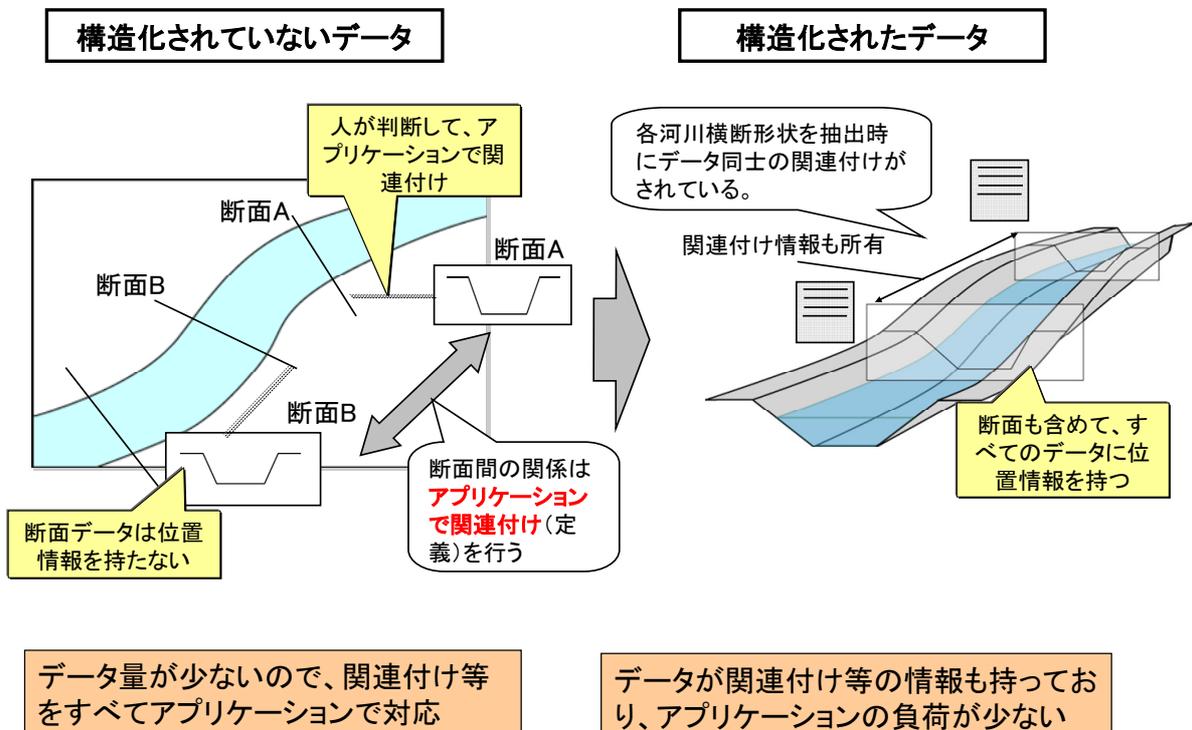


図 2-7 構造化データのイメージ

また、データの標準化を進めることで、同じ解析を行う別のプログラム間でのデータの共有利用が実現し、これまで、大きな時間と作業量を費やしていたデータ作成業務、データ作成・加工作業が減少することが期待されている。

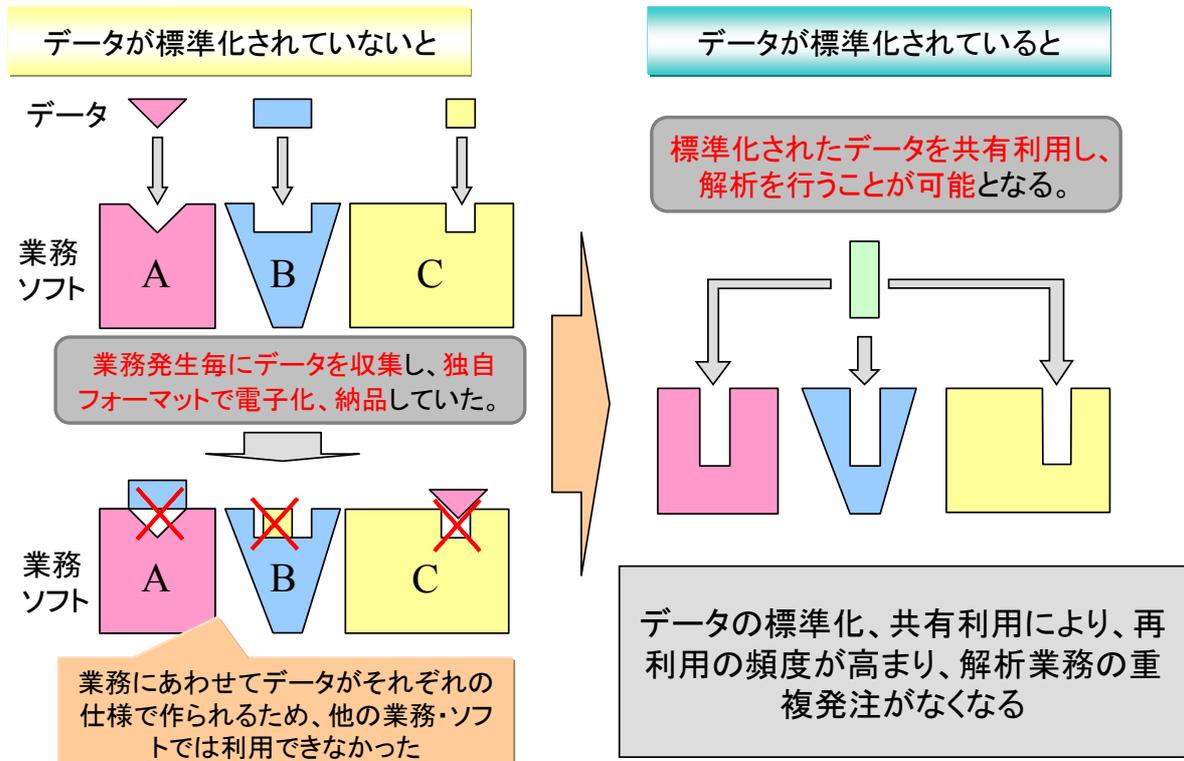


図 2-8 標準化データによるデータの共有利用

## 2.3 インターフェースの標準化

### 2.3.1 インターフェースの標準化とは

ここでのインターフェースとは、下図に示すシミュレーションモデルの全体図において、データベースとアプリケーション間のデータの送受信にかかわるものを指し、降雨予測、流出計算、河道計算等、さまざまな水理・水文・水質シミュレーションモデルと、それらが共通して利用するデータを接続する仕様、関数、ルールを定めることである。

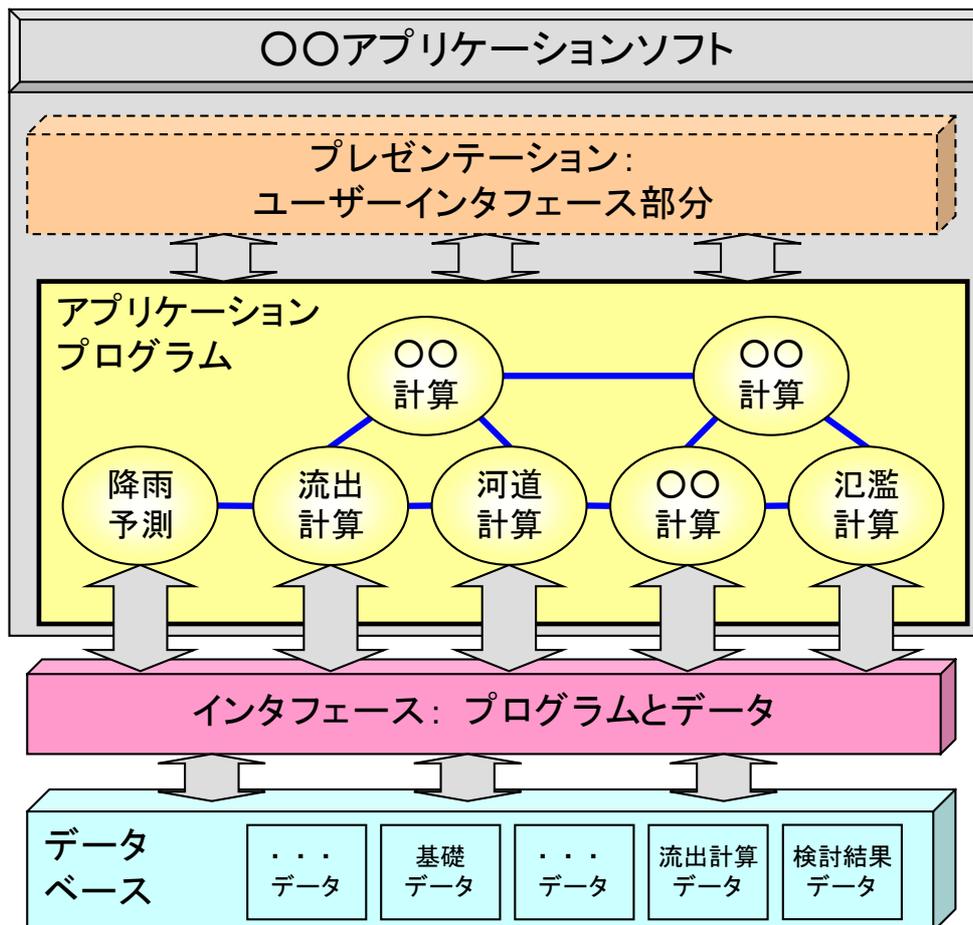


図 2-9 インターフェースの構成

### 2.3.2 インターフェース標準化の方向

河川局では、現在、図 2-10 に示す範囲でのインターフェースの標準化を進めている<sup>2)</sup>。

汎用型シミュレーションモデルにおけるインターフェースにおいても、この仕様を基本に策定を進めるべきであるが、水理・水文・水質シミュレーションモデルに関する必要な関数仕様等については議論が十分でないため、この標準仕様を参考に必要なインターフェースについても、議論・検討していく必要がある。

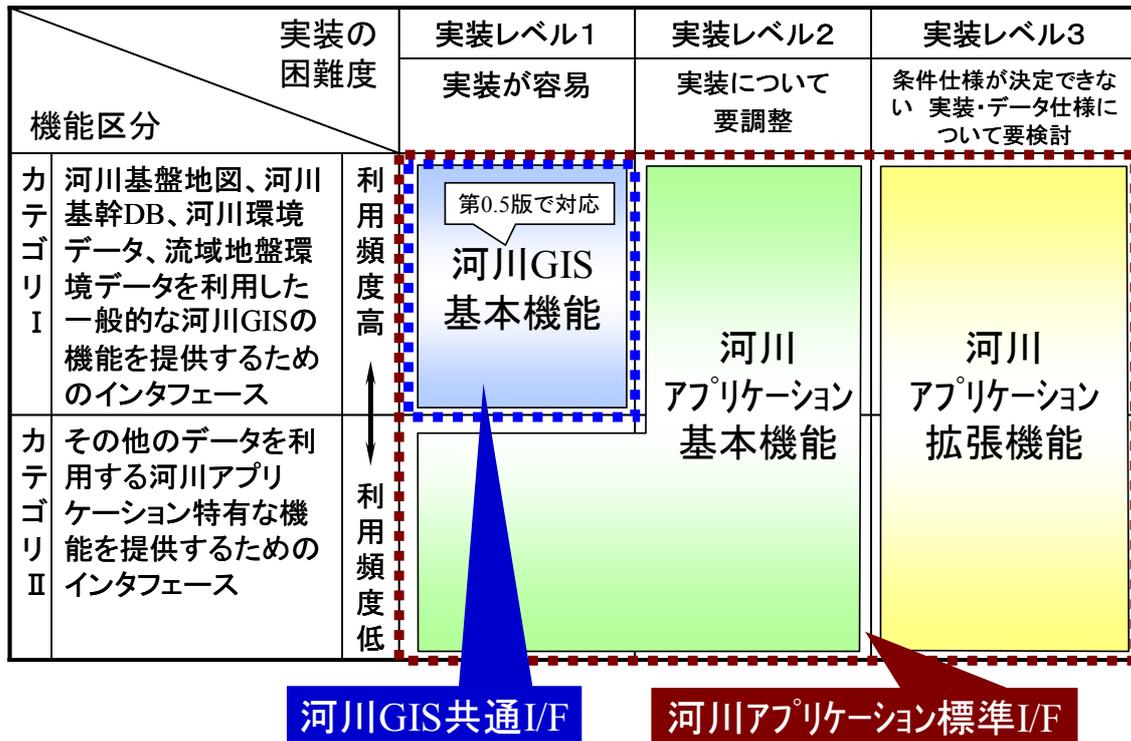


図 2-10 河川 GIS・アプリケーション標準インターフェースの区分

## 2.4 プログラム構造の標準化

### 2.4.1 プログラム構造の標準化とは

シミュレーションモデルの開発はこれまで個々の主体により行われ、他者より優れたソフトウェアを構築することが、研究者や企業にとっての開発インセンティブとなってきた。このため、開発に係わるノウハウは共有化されることなく、多種多様なモデルが構築されてきたと言える。汎用性に富み、性能の優れたソフトウェアを提供していくためには、開発に係わる技術情報の共有化による開発効率の向上や、統一基準に基づく個々のプログラムモジュールの組み合わせ、相互補完が考えられる。

前述のように、汎用型シミュレーションモデルとは、データ構造、インターフェース、アプリケーションプログラムのそれぞれが標準化されたものである。プログラム構造の標準化ならびにデータ構造の標準化、インターフェースの標準化によって、

- ・ 要素モデルが容易に変更できる。
- ・ 要素モデルを組み合わせ、規模の大きいモデルを構築できる。
- ・ 流域水物質循環のシミュレーションモデルの開発の効率が向上する。
- ・ より流域の課題解決に向けたソリューションを提供できる。
- ・ インターフェースの作成に労力をかけずに、要素モデルの開発に注力できる。

プログラム構造の標準化において、重要な概念となるのが次項で説明する標準フレームワークである。

アプリケーションプログラムは、複数のプログラムモジュール（オブジェクト）により構成されている。各モジュール同士は、決められたプログラムインターフェースで通信しており、その手順を共通化することで、モジュールの共通化、再利用が可能となっている。

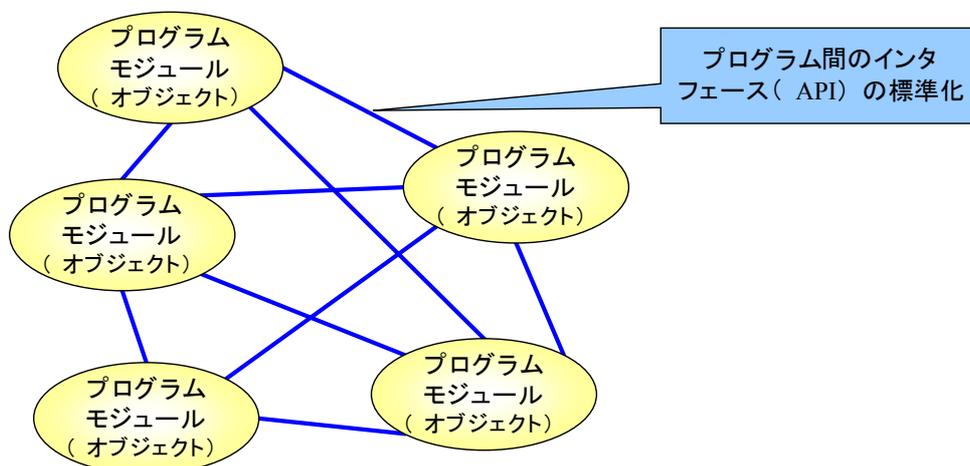


図2-11 プログラム構造の標準化

### 2.4.2 標準フレームワークとは

標準フレームワークとは、汎用型シミュレーションモデルを構成するコアエンジンのシステム部分のことを指し、その機能を下記の3点に要約する。

- ・ 標準フレームワークとは、モデルそのものではなく、モデルを構築するためのシステム（仕組み）のことである。

- 標準フレームワークの仕様に基づいて、種々の要素モデルを構築することが出来る。
- それぞれの要素モデルを自由自在に相互接続し、複合的な物理現象をシミュレートする全体系モデルを構築することが出来る。

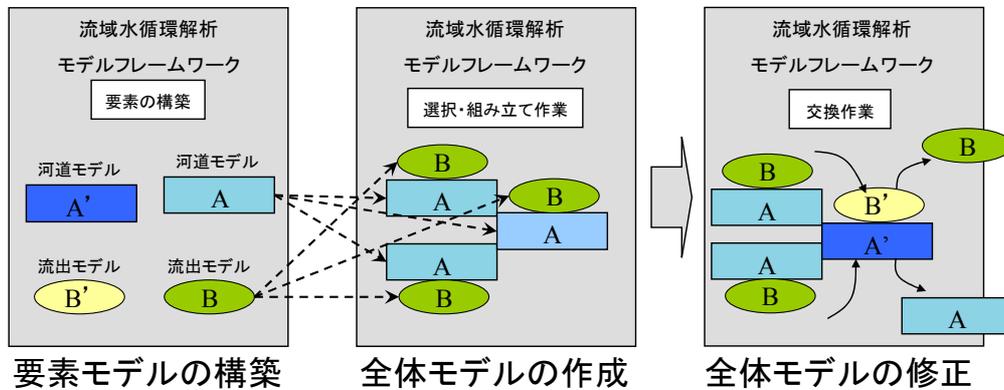


図 2-12 標準フレームワークのイメージ～流域水循環解析モデルを例に～<sup>3)</sup>

標準フレームワークの仕様に基づいて、要素モデルを構築することが出来る仕組みが整っていることで、様々な開発者が流域水循環に関係する要素モデルを多種多様にコーディング（ソースコードを作成すること）することが可能である。それぞれの要素モデルの相互接続の規定（プログラムインターフェース、データインターフェース）も予め定められていることから、自由自在に要素モデルを相互接続し、複合的な物理現象をシミュレートする全体系モデルを構築することも可能である。種々の物理現象を説明する要素モデルが多いほど、その中から取捨選択して、ある流域の水循環解析にカスタマイズした全体系モデルの構築が可能と期待できる。また、ある全体系モデルを構成している要素モデルを別の要素モデルに交換して応答を見てみたいといったニーズに容易に答えることが可能となる。

### 2.4.3 標準フレームワークのコンセプト

“標準フレームワーク”とは、データ交換・連携インターフェースの機能を有し、異なる解析モデルの接続を可能とし、複合的な流域水循環解析を実現するためのモデルフレームワークである。すなわち、この標準フレームワークによって、様々なユーザーが水文・水理現象に関わる種々のモデルを任意に組み合わせて、必要な入力データを効率よく処理しつつ、求める計算を実行できる環境を得ることができる。

また、今後必要となる水工系以外の解析プログラムと連携させる複合的な解析に対応させるべきものでなければならない。

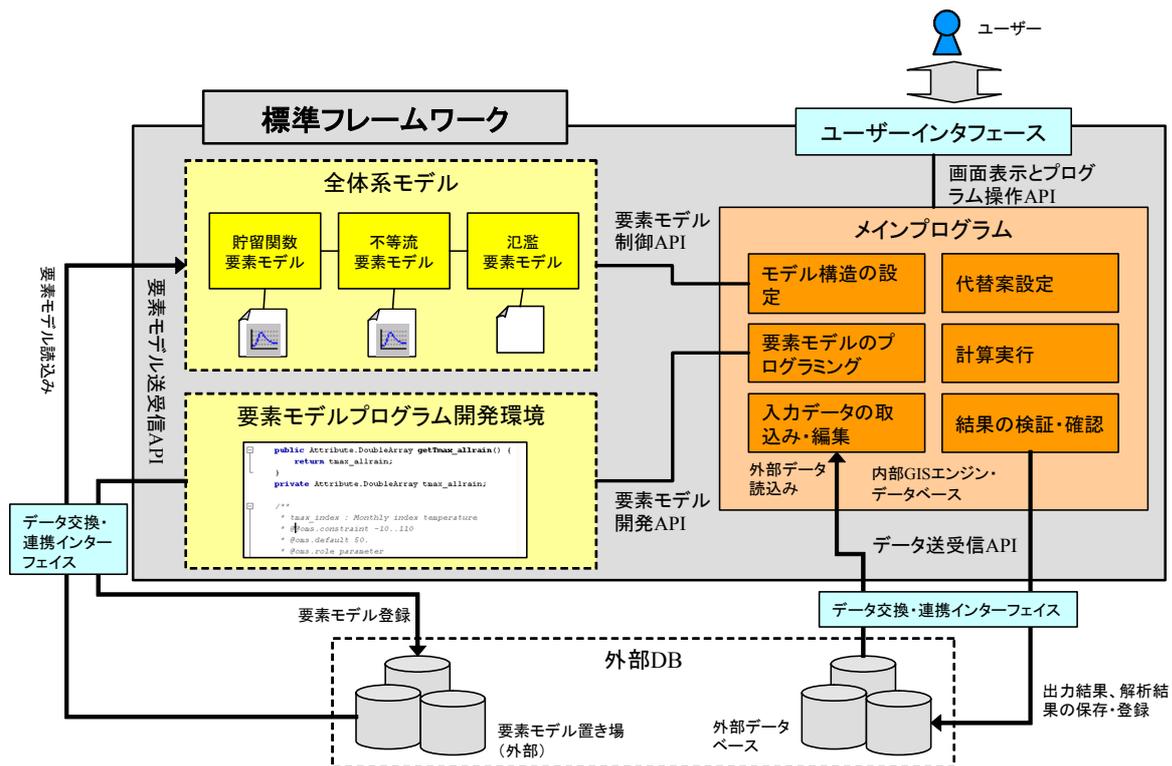


図 2-13 標準フレームワークの機能イメージ図

#### 2.4.4 標準フレームワークの開発効果

標準フレームワークを開発することによって得られると期待される効果の要点をまとめると以下の項目が挙げられる。

- 異なった環境で作成されたソフトが流通し、競争が始まるので
  - 研究が活性化する
  - 水循環・水環境行政が高度化でき、新しい河川や水利用に関する施策が提案される
  - 学生や技術者が直接ソフトを操作できるので、技術力が向上する
- ソフトの重複開発が避けられ、メンテナンスコストが下がり、行政機関とコンサルタンツ、研究機関すべてで、生産性が向上する
- ソフトの品質がオープンになるので、国民間の政策合意形成に寄与される
- アジアの特徴を表現でき、外国との競争力が高まる

以下、標準フレームワークの開発に向けて、プログラミング言語について検討する。

## 2.5 標準フレームワークの開発言語

### 2.5.1 標準フレームワークの開発言語の検討

ここでは、標準フレームワークの開発言語について検討する。これまでも述べてきたように標準フレームワークは水理・水文・水質解析の共通基盤となるフレームワークであり、完成されたモデル・ソフトウェアを開発するわけではない。また、様々な技術レベルのユーザーによる利用を想定したものであり、フレームワークに自ら要素モデルを加えるなどして利用するユーザーもいるだろう。従って、フレームワークを開発するプログラム言語が適しているかについて調査検討しておく必要がある。

### 2.5.2 オブジェクト指向型言語の概要

オブジェクト指向とは、ソースコードをクラスと呼ばれる単位に分割し、それらを統合することによりプログラムを構築することであり、こういった方法でプログラミングする言語のことをオブジェクト指向型言語という。代表的なオブジェクト指向型言語としては、C++、C#、Java が挙げられる。手続き型言語にはないオブジェクト指向型言語の特徴として

- ・プログラムの作成
- ・プログラムの再利用
- ・プログラムの保守

のしやすさが挙げられる。表 2-1 は、オブジェクト指向型言語と手続き型言語の違いを説明したものである。オブジェクト指向型プログラムでは、「あたかも、オブジェクトが自分で何でも知っていて、自分で何でも自律的に行動するようにプログラミングする」ので、「クラスオブジェクトのユーザーは、そのオブジェクトに命令する（メッセージを送る）だけで OK」である。手続き型言語が、「よちよち歩きの子供を手取り足取り世話するイメージ」であるのに対し、オブジェクト指向型言語のクラスオブジェクトは、「自律的に行動できる大人というイメージ」である。

表 2-1 手続き型言語とオブジェクト指向型言語

プログラム技法	説明	主な言語
手続き型言語 (procedural language)	処理を必要最小限の手続き単位に分解・作成し、これを必要とする処理に基づいて組み立てていく手続き指向の言語	FORTRAN C COBOL Pascal etc.
オブジェクト指向型言語 (object oriented language)	プログラムをオブジェクト(物)の単位としてとらえ、オブジェクトの相互的な作用によって処理を行う、オブジェクト指向の言語	Smalltalk C++ Java Visual Basic etc.

### 2.5.3 オブジェクト指向型プログラムの重要な概念

オブジェクト指向型プログラムが、プログラムの作成、プログラムの再利用、プログラムの保守を容易にするのは、先の述べたようにプログラムの要素がオブジェクトとして自律的に行動することにあるが、どのようにしてこのようなプログラミングが実現されるのか。ここではオブジェクト指向型プログラムの重要な概念を説明する。

#### ①クラス、インスタンス

クラスとは共通の特性（属性、振る舞い）を持つオブジェクトの集合である。モデリングシステムにおいては、クラスとはあるデータ群とそれら进行操作する関数群を一つのパッケージにまとめたものを指す。このデータ群をデータメンバ、関数群をメンバ関数と呼ぶ。要素モデルは状態量やパラメータといったデータ群を、水文特性を表現する数式群で操作することによって計算を行うので、状態量やパラメータをデータメンバ、数式群をメンバ関数とすれば、要素モデルをクラスで表現することができる。

クラスは様々なプログラムで共通に利用することができる。例えば、プリントアウト機能は、あらゆるソフトウェアで必要な機能である。こういったとき、プリントアウトとする機能を実装したクラスがあれば、様々なソフトウェアを開発するときにこのクラスを自分でコーディングすることなく、共通して利用すればよい。また、高度なプログラミング技術がなくても、他者が実装した高度なクラスを自分のプログラムに組み込むことも可能で、インターネット上にも無償で高性能なクラスが公開されていることもある。

このように、ソースコードを機能ごとに分割してプログラミングすることにより、一つのプログラムを構築するのに複数のプログラマがそれぞれクラスを作成するという作業分担ができるため、ソフトウェア開発の生産性が向上される。また、別のプログラムに以前作成したクラスを再利用することもできる。

#### ②メッセージ

メッセージとは、オブジェクトに対する指示や設定のことである。すなわちオブジェクトに対して、プロパティを設定したり、メソッドを動かすように指示することをメッセージという。このメッセージを送ることを”メッセージパッシング”という。

図 2-14 の例では、オブジェクトに「ベルを止める」というメッセージパッシングをしているが、プロパティそのものを設定する、たとえば現在時刻を設定する、目覚まし時刻を設定するなどのプロパティの設定ももちろんメッセージパッシングである。ちなみにベルを止めるというメッセージパッシングは、実生活では目覚まし時計のボタンを押すという行為になる。

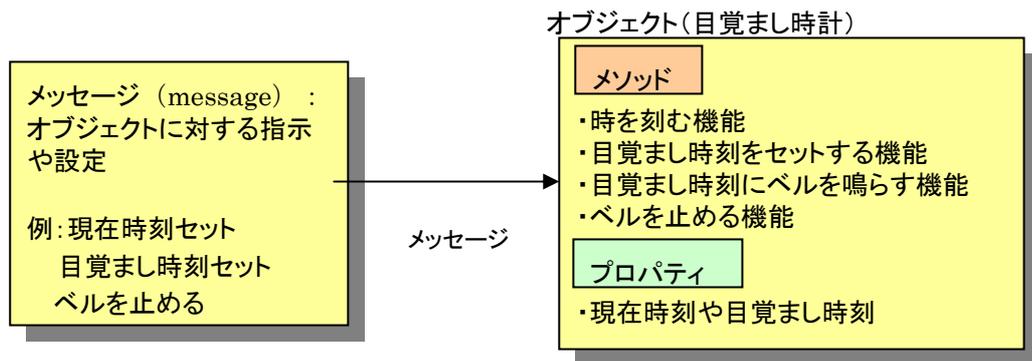


図 2-14 メッセージの概念

### ③継承 (インヘリタンス)

現在、大規模なソフトウェアやプログラムはほぼオブジェクト指向型言語を用いて作成されているといってもよく、プログラミング言語の主役となっている。その理由は、クラスを共通利用できるという点に留まらない。

オブジェクト指向型言語には、「継承」という機能が含まれている。継承とは新たなクラスを作成する際に既存のクラスの機能を引き継ぐことである。例えば、図 2-15 に示すようにある機能を持ったクラス (親クラス) から新たなクラス (子クラス) を作成すると、親クラスが持っている機能は全て子クラスが受け継ぐことになり、追加したい機能だけの子クラスに実装すれば、わざわざ子クラスに必要な全ての機能をコーディングすることなく子クラスは「親クラスの機能+追加した機能」を持つことができる。このように継承機能を有するにより、新たにクラスを作成することも容易になる。

例えば、目覚まし時計、腕時計、大時計という子クラスは、時計という親クラスのメソッドとプロパティを自動的に継承した上で、それぞれの子クラス独自のメソッドとプロパティを定義することになる。

水文モデルで言えば、流域流出モデルという親クラスは、雨から流量を算出するというメソッドを持ち、面積というプロパティを有する。その下の子クラスである貯留関数法では、貯留方程式というメソッドと流出率、パラメータ K,P,TI というプロパティを有することになる。

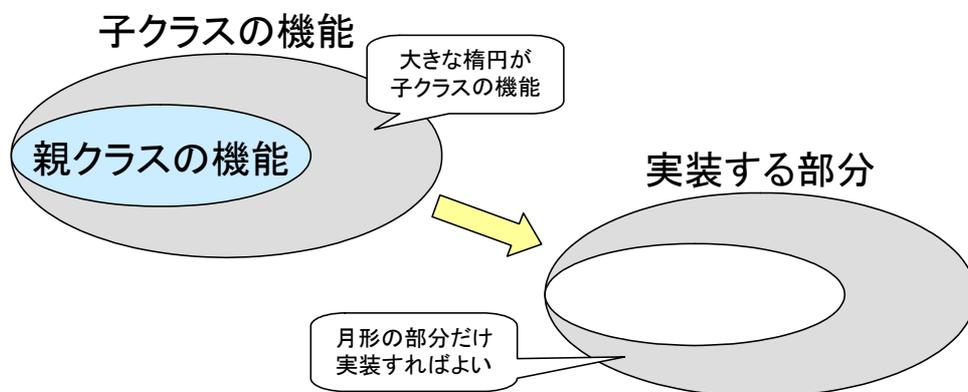


図 2-15 クラスの継承

#### ④多態性（ポリモフィズム）

2つのクラス A, B が全く同じ機能を備える場合には継承を利用すればよいことがわかった。しかし、機能によっては、A, B ともにその機能を備えることがわかっていても、その内容までは同一に定義できないものもある。例えば、全ての要素モデルがその水文特性に応じて数理計算を行う機能を備えるのはわかっているが、その内容は当然ながらそれぞれ異なる。しかし、『水文学的な計算を行う』ということに関しては共通であり、その手続きを統一することで利便性・汎用性を高めることができる。

このように、クラスのメンバ関数の使用方法のみ（メソッド）を規格化し、その操作により実現される内容（メソッドの方法）を個々の派生クラスごとに定義できることを多態性という。下図の例で、スーパークラスの「楽器」で「演奏する」というメソッドが定義されているが、このメソッドは、ピアノクラスではピアノの演奏を、バイオリンクラスではバイオリンの演奏をするという意味を持つ。例えば、流域流出モデルで、雨から流量への変換を行うというメソッドを定め、それを「Calculate」という名前にすれば、貯留関数や kinematic wave のクラスで具体的に計算式を定義すればよいことになる。

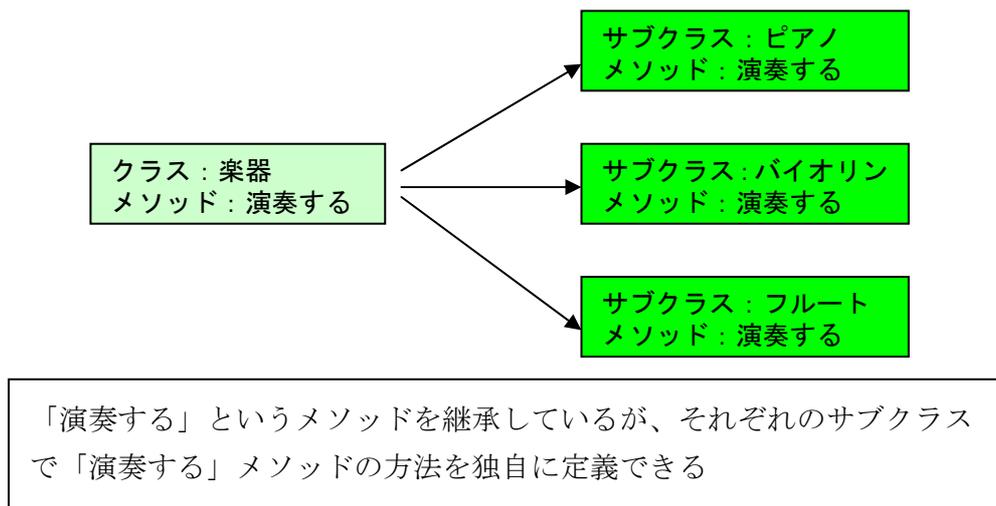


図 2-16 多態性の概念

## 2.6 既存のフレームワークの調査

前節まで説明してきたモデル構造の標準化を意識した水理・水文に関するフレームワークの開発が進められてきている。例えば、「標準フレームワーク」の概念に近いシステムとして、欧米で OpenMI(欧州関係機関が推進する HarmonIT プロジェクト)、MMS (米国地質調査所 (USGS) が推進するプロジェクト)、OMS (米国農務省 (USDA) が推進するプロジェクト) として開発されてきているが (詳細は 3 章で説明)、日本においても京都大学椎葉研究室で開発・保守・管理されている「OHyMoS (Object-oriented Hydrological Modeling System・オハイモス)」がある。本節では、「OHyMoS」の調査結果を整理する。

### 2.6.1 OHyMoS とは

OHyMoS とは、Object-oriented Hydrological Modeling System の略であり、オブジェクト指向型水文モデリングシステムである。その概要は、「水理・水文モデルの共通基盤を開発し、その基盤上でモデルを開発させ、モデル同士の接続を可能にすることにより、流域規模の統合型水文解析を構築するための水理水文モデル構築システム」である (図 2-17 参照)。

OHyMoS のような標準フレームワークや標準的な要素モデルを広く公開し、誰もが利用することができる仕組みがあれば、他機関が開発した要素モデルを取り入れ、新たなモデルを構築することもできるし、自身で作成した要素モデルと結合させることもできる。また、数値計算の技術を持たない者であっても、公開されている要素モデル同士を接続し、解析モデルを構築することができるため、水理・水文計算を行うことができる。

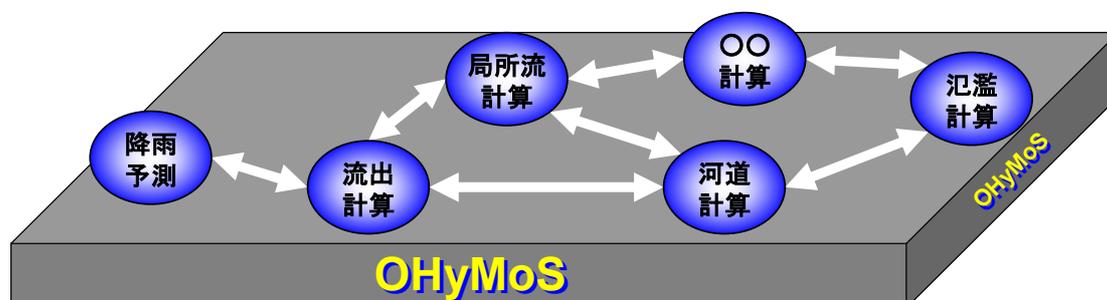


図 2-17 OHyMoS の概念図

### 2.6.2 開発の経緯

OHyMoS は京都大学の椎葉研究室で開発されたものであり、開発の経緯は次のようである。現在、OHyMoS はバージョン 5 まで開発されており、オブジェクト指向型言語である C++ で記述されている。また、web との親和性が高い Java (Java もオブジェクト指向型言語) で記述された OHyMoSJ も開発されており、バージョン 2 まで開発されている。OHyMoS、OHyMoSJ 共に、web サイト (<http://hywr.kuciv.kyoto-u.ac.jp/ohy mos.html>) でソースコードから公開されており、いくつかのサンプルプログラム (シンプルな数値解析モデルが主) と共に OHyMoS および OHyMoSJ をダウンロードすることができる。

- ・ 1991-2 年 : 原型の開発

- ・1993-4年：現在のシステムがほぼ完成
- ・2003年：Java版 OHyMoS-J の開発

### 2.6.3 OHyMoS の基本的な概念

OHyMoS の基本概念は図 2-18 のとおりであり、要素モデルを複数個接続することで、全体系モデルを構成し、個々のモデルは交換可能とする。こうすることで、他の要素モデルのコードまで、解読しなくても、要素モデルの付加や交換が可能になるのである。しかし、このようなモデル構成にするためには、A,B,C,D の各要素モデルが独立していることが必要となる。B の動作に A が必要なら、A を D に入れ替えることができない。このように個々の要素モデルを独立とするためには、FORTRAN などの手続き型言語では難しく、個々の要素モデルをオブジェクト（物）として取り扱うオブジェクト指向型言語を用いることになる。

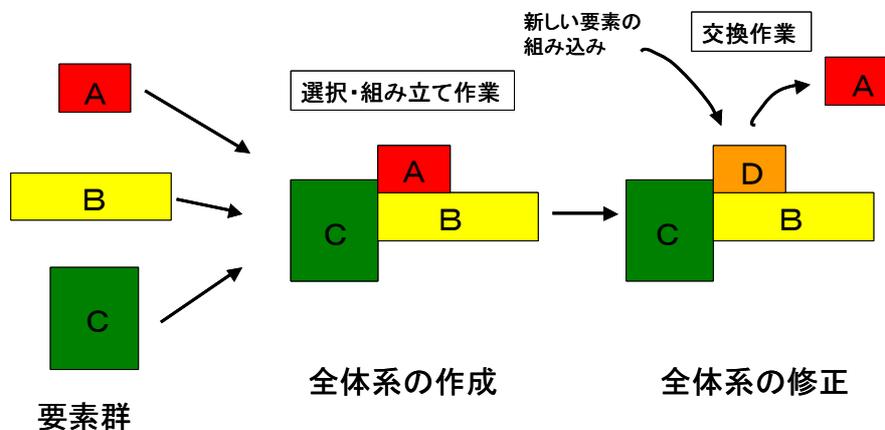


図 2-18 OHyMoS の基本的な概念

### 2.6.4 OHyMoS の代表的なクラス

OHyMoS を一言で表すならば、クラスライブラリー（同じクラスを様々なプロジェクトで利用できるようにしたもの）であると言うことができる。解析モデルに必要なプログラムを予め C++ もしくは Java のクラスとして用意している。OHyMoS が予め用意しているクラスは、モデルのエンジン部分を構成するクラスや、データの交換、ファイルの読み込み、ファイルの吐き出し、要素モデル間のデータの交換などの役割を持つクラスなどである。

OHyMoS のシステムと要素モデルの関係をコンピュータの世界に例えるならば、OS とアプリケーションの関係で例えることができる。コンピュータで稼働させるアプリケーションは共通の OS 上で動かすことを前提に作られているからこそ、アプリケーション間のデータの交換などを行うことができるし、同じアプリケーションを異なるコンピュータで利用することができる。つまり、各々の要素モデルは OHyMoS という仕組みの上で動くように作ることによって、要素モデルの共通利用や、モデル間の情報のやり取りを可能にすることが OHyMoS の基本的な概念の一つである「モデル構成の共通基盤」としての機能である。

また、様々な要素モデルの交換・追加・削除が容易に行うことができることも OHyMoS の特徴の一つであり、他機関が作ったモデル同士であっても、OHyMoS の仕様を満たしてい

る要素モデルであれば、自由に組み合わせることができ、さまざまな水理水文モデルを構築することができる。

要素モデルの新規開発や既往モデルのカスタマイズを行うことも可能で、要素モデルを新規開発する場合であっても、ユーザーは特段プログラミング言語（C++、Java）に精通している必要がないような仕組みが考えられている。

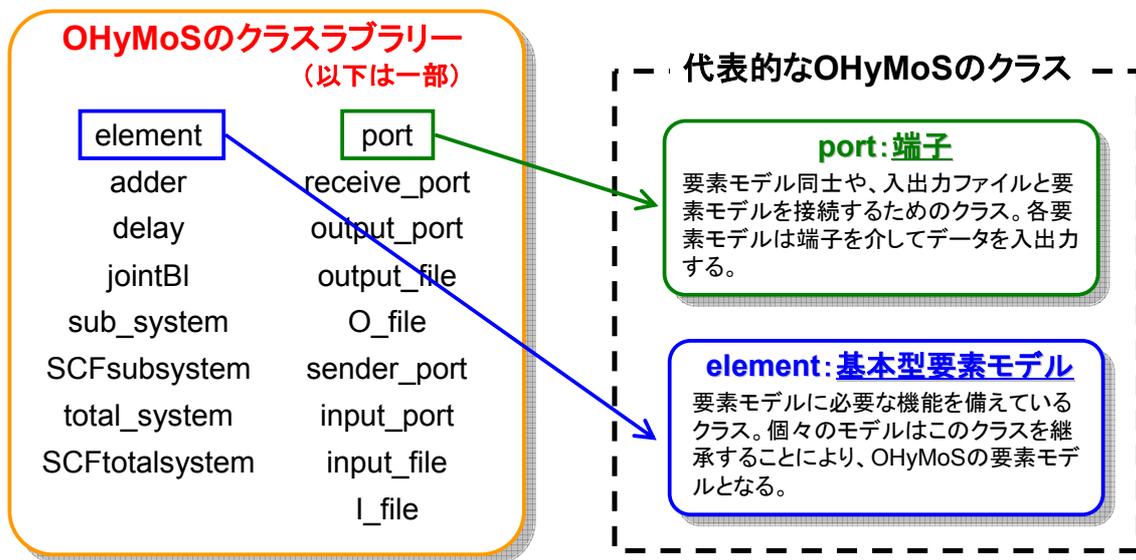


図 2-19 OHyMoS の代表的なクラス

### 2.6.5 OHyMoS の基本構造

上述したように、OHyMoS では要素モデルの機能を共通の機能と固有の機能に分離し、共通の機能を基本型要素モデルとする。すなわち基本型要素モデルをひとつのベースクラスとする。基本型要素とは、パラメータ値の設定、状態量の初期化、計算時間の更新、時系列データ入力機能などである。次に、各要素モデルは、この基本型要素モデルを継承して、独自機能のプロパティとメソッドを定義することにより作成されることになる（図 2-20 参照）。

図 2-21 にモデルの構成イメージを示す。モデルは要素モデル、部分系モデル、全体系モデルという 3 つの概念を持つ。要素モデルは個々の水文要素に対応するモデルである。部分系モデルは、要素モデルを幾つか統合したモデルである。例えば、土中の水理モデルは中間流と地下水流という 2 つの要素モデルで構成されるが、これら 2 つの要素モデルをあらかじめ接続し、部分系モデルとする。全体系モデルは、これら要素モデルと部分系モデルで構成されるものであるが、全体系モデルとしては、ユーザーとの対話作業、ファイルとのデータの授受、要素モデル・部分系モデルに対する計算実行命令という 3 つの機能を持つ。

また、要素モデル間、要素モデルと部分系モデル間、及び部分系モデル間のデータのやりとりは全て端子で行われ、これら端子の基本型要素も作成されている。

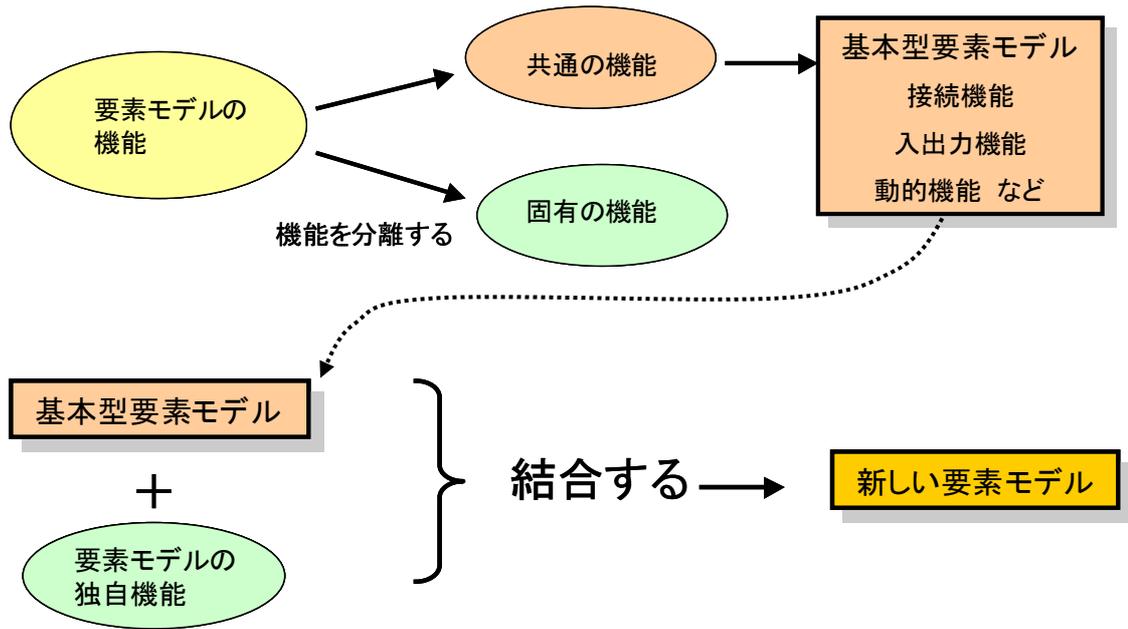


図 2-20 OHyMoS の基本構造

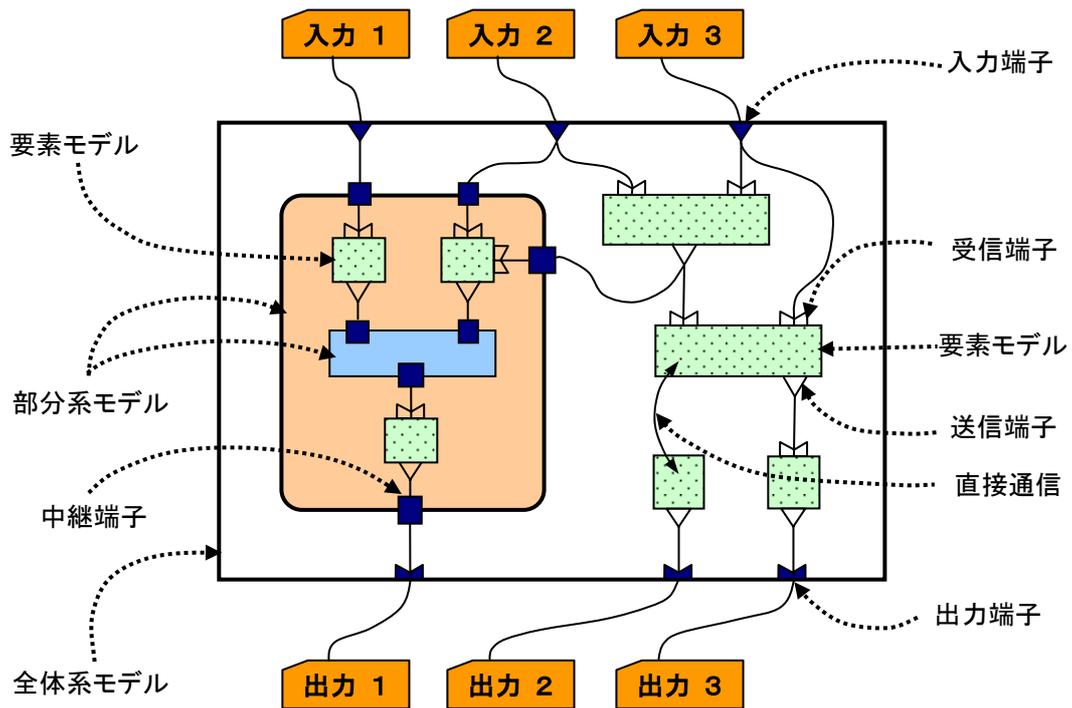


図 2-21 OHyMoS のモデル構成 <sup>3),4)</sup>

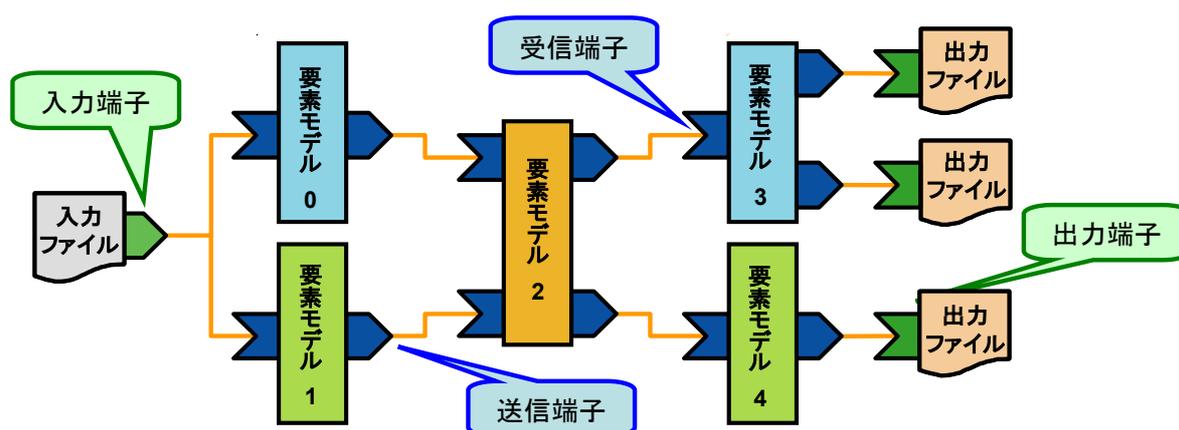
## 2.6.6 データの受け渡し

OHyMoS では、要素モデル間のデータの交換に『端子』という考え方を導入しており（端子のクラスが用意されている）、各要素モデルは端子を介してデータの授受を行う。端子には入力端子、出力端子、送信端子、受信端子の4種類があり（4つのクラスが用意されている）、入力端子は入力ファイルに、出力端子は出力ファイルに、送信端子・受信端子は要素モデルに付属する。

各端子には固有の認識番号が割り当てられ、その認識番号を利用して端子同士のつながり、つまり要素モデルや入出力ファイルのつながりを定義する。後述の構造定義ファイルにおいて、固有の認識記号を用い、端子同士のつながりを定義することにより要素モデルの接続関係を決める。

要素モデルは受信端子からデータを受け取り、送信端子でデータを送信することになる。受信したデータを変数に代入し、ある変数の値を送信端子から送るが、その変数を決めるのは、要素モデルのプログラムでユーザーが決める。端子から送信される情報は必ずしも流量だけではなく、流速、水深といった情報も送ることができる。

図 2-22 は、3種類の要素モデルを全部で5つつなげ、入力ファイルから出力ファイルに至るまでの過程を示したイメージ図である。



1. 入力ファイルの情報が要素モデル 0,1 に送信される
2. 要素モデル 0,1 が計算を行う
3. 要素モデル 0,1 の計算結果がそれぞれ要素モデル 2 に送信される
4. 要素モデル 2 が計算を行う
5. 要素モデル 2 の計算結果がそれぞれ要素モデル 3,4 に送信される
6. 要素モデル 3,4 の計算結果がそれぞれ出力ファイルに送信される

図 2-22 要素モデルと端子のイメージ

## 2.6.7 要素モデル

### (1) 基本型要素モデル

OHyMoS では基本型要素モデルというクラスを用意している。基本型要素モデルは OHyMoS 上で稼働するための必要な機能を予め備えており（必要な関数が実装済みである）、

この基本型要素モデルというクラスを継承することにより、OHyoMoS の要素モデルとしての機能を持たすことができる。概念図を図 2-23 に示す。

要素モデルの構築にあたり、ユーザーが実装しなければならないのは、1 ステップ分の計算プログラムと、送受信端子の定義、パラメータ値・初期値の設定などの関数である。一般的に数値計算モデルを構築しようとする、計算部分とは直接関係のない入出力データに関するプログラミングや、画面表示のプログラミングなどに時間と手間がかかってしまうことが多いが、OHyoMoS では前述の通り、基本的な機能は予め準備してあるので、ユーザーは計算の本質部分のプログラミングに集中することができ、モデル開発にかかる時間や手間を削減することができる。

ユーザーは実装しなければならない関数の中身を記述することで、要素モデルを開発するが、その関数名が明らかになっているため、どの関数の中身を実装すればよいか明確になっている。例えば、1 ステップ分の計算プログラムを実装する場合、`calculate()` という関数の中身を実装すればよく、計算スキームを記述すべき箇所がユーザーにとって明確になっている。受信端子の定義は受け取ったデータをどの変数に代入するかを決め、送信端子の定義は計算によって求められた値のうち、どの値を接続先の要素モデルへ送信するかを決める。送受信端子の数に制限はなく、一つの要素モデルにいくつの送受信端子を作ってもよい。パラメータ値・初期値の設定はそれぞれ、`set_parameter()`、`set_initialstate()` という関数を実装する。



図 2-23 要素モデルと基本型要素モデルのイメージ

## (2) 要素モデルの独立性とボトムアップコントロール

OHyoMoS の特徴は要素モデルの交換・追加・削除が自由に行えることにあり、要素モデルは他の要素モデルから独立していることが求められる。つまり、A モデルを削除したら B モデルが動かないとか、C モデルを追加したら D モデルが動かないということがないような要素モデルを構築しなければならない。

OHyoMoS は要素モデルの自立性というものを重要視しており、メインルーチンを置かないボトムアップコントロール型のモデリングシステムである。OHyoMoS の基本コンセプトはあくまで水理水文学解析の共通基盤を用意することであり、要素モデルは他のモデルに依存していないこと (他のモデルがないと計算することができないということのない) が求められる。

OHyoMoS では、要素モデル自らが周りの要素モデルの計算状況を踏まえながら計算を進めていくため、ある要素モデルを交換・追加・削除したとしても、その状況に応じて計算を進める。メインルーチンを置かないことのメリットは、モデル構成を変えたとしても、プログ

ラミングの中身を変更する必要がなく、次節で後述する構造定義ファイルというシステムとは外部にあるテキストファイルを書き換えることにより対応できることにある。

手続き型言語で書かれている従来型の数値計算モデルは要素モデルをサブルーチン化しておき、メインルーチンからトップダウン的に、それらの要素モデルをサブルーチンとして計算の手順をコントロールしているため、要素となるコンポーネントを交換・追加・削除した場合、メインルーチンを更新する必要がある。一方、OHyMoSの場合、メインルーチンからの指令によって要素モデルの計算をコントロールするのではなく、それぞれの要素モデルが接続先の要素モデルとの情報のやり取りを通して、要素モデル自身が自らの計算手順をコントロールする。メインルーチンの仕事は、要素モデル間のつながりを定義することだけで、計算手順は要素モデル自身が判断し、計算を進めていく。

OHyMoSでは計算するためのデータを受信している要素モデルから計算を行い、接続先の要素モデルにデータを送信する、という手順を取るため、必ずしも時間ステップごとに各要素モデルが時間的に横並びに計算を進めていくわけではない。例えば、河道網の流出解析をした場合、必然的に上流側の要素モデルから計算が進められ、最後に最下流の要素モデルの計算が行われることになる。上流からの流入量が決まらない限り、下流の要素モデルが計算できないからである。

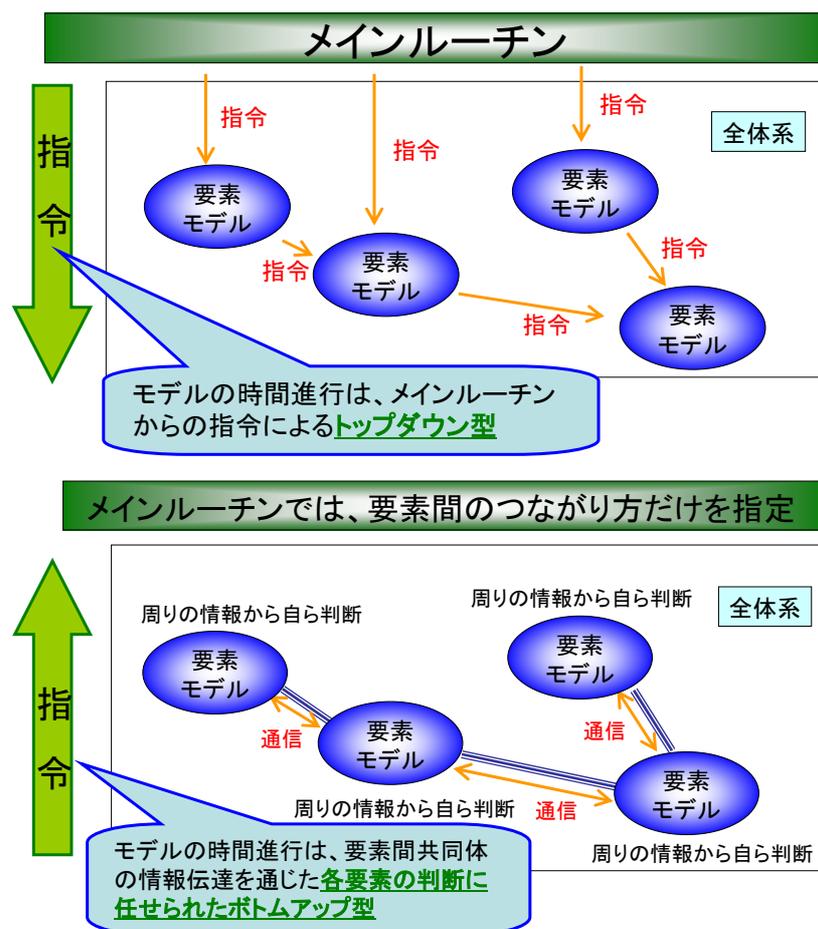


図 2-24 トップダウンコントロールとボトムアップコントロールの比較

### 2.6.8 構造定義ファイル

構造定義ファイル (Structure Configuration File) とは、OHyMoS バージョン 3.0 以上 (OHyMoSJ はすべてのバージョンで対応) で使うことのできるモデルの構成を定義するためのテキストファイルであり、ユーザーはこの構造定義ファイルを作成することにより、解析モデルに組み込む要素モデルの選択、要素モデルの接続関係の定義、入出力ファイルの選択を行い、解析モデルの構成を決定する。従って構造定義ファイルは解析モデルの全容を決定するファイルであり、OHyMoS による解析モデル構築の核となるファイルである。

構造定義ファイルでは、入力ファイル、要素モデル、出力ファイルそれぞれに認識番号を付与して ID 化し、その ID を用いて要素モデルのつながりを定義する。

構造定義ファイルで定義する項目は以下の通りである。

- Number part
  - 入力ファイルの総数
  - 要素モデルの総数
  - 反復計算を行う要素モデルの総数
  - 反復計算を行うセットの総数
  - 出力ファイルの総数
- Input Ports
  - 入力ファイルの登録 (ID を付与)
- Components
  - 要素モデルの登録 (ID を付与)
- Output Ports
  - 出力ファイルの登録 (ID を付与)
- Connect Ports
  - 入力ファイルと要素モデルのつながりを定義 (ID を利用)
  - 要素モデル同士のつながりを定義 (ID を利用)
  - 要素モデルと出力ファイルのつながりを定義 (ID を利用)

構造定義ファイルはプログラミングソースのファイルではないので、ファイル作成にあたり、プログラミング言語に関する知識は必要なく、簡単な記号と数字を打ち込むだけで作成が可能である。よって、プログラミングに関する技術がなく、自ら数値解析モデルの開発を行うことができないユーザーでも、OHyMoS を用いれば数値解析を行うことができる。

また、構造定義ファイルは OHyMoS システムの外側にあるファイル (ソースではない) なので、構造定義ファイルを更新しても、再コンパイルや、システムの再構築などをすることなく、容易に様々な計算パターンを試すことができる。

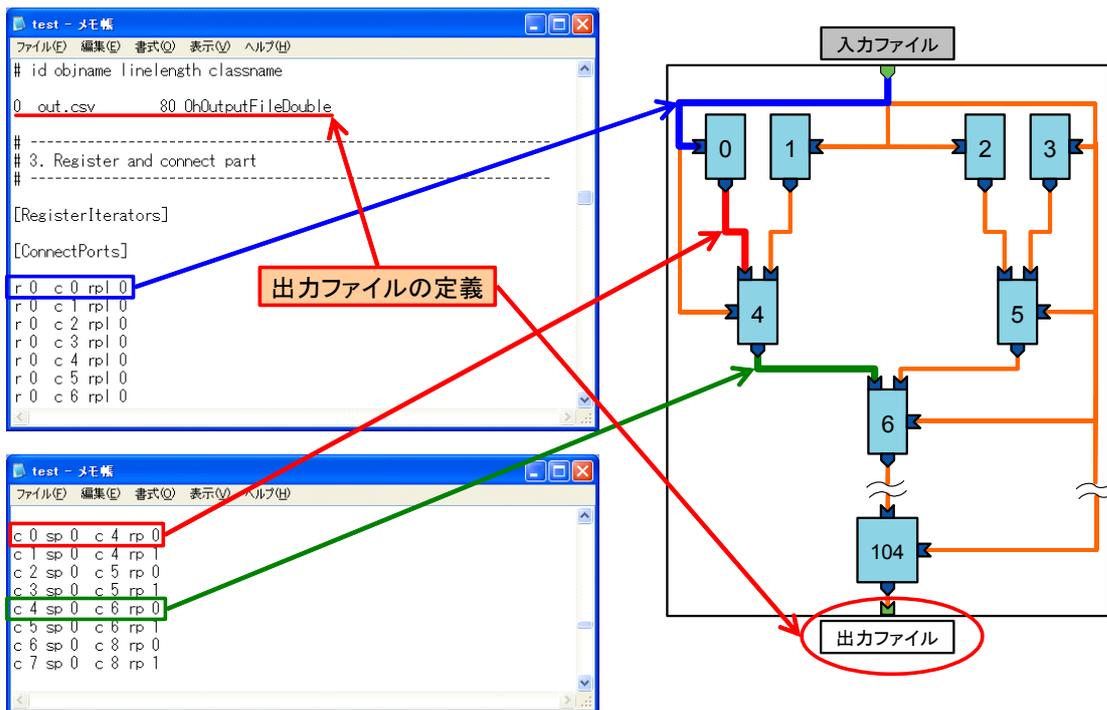
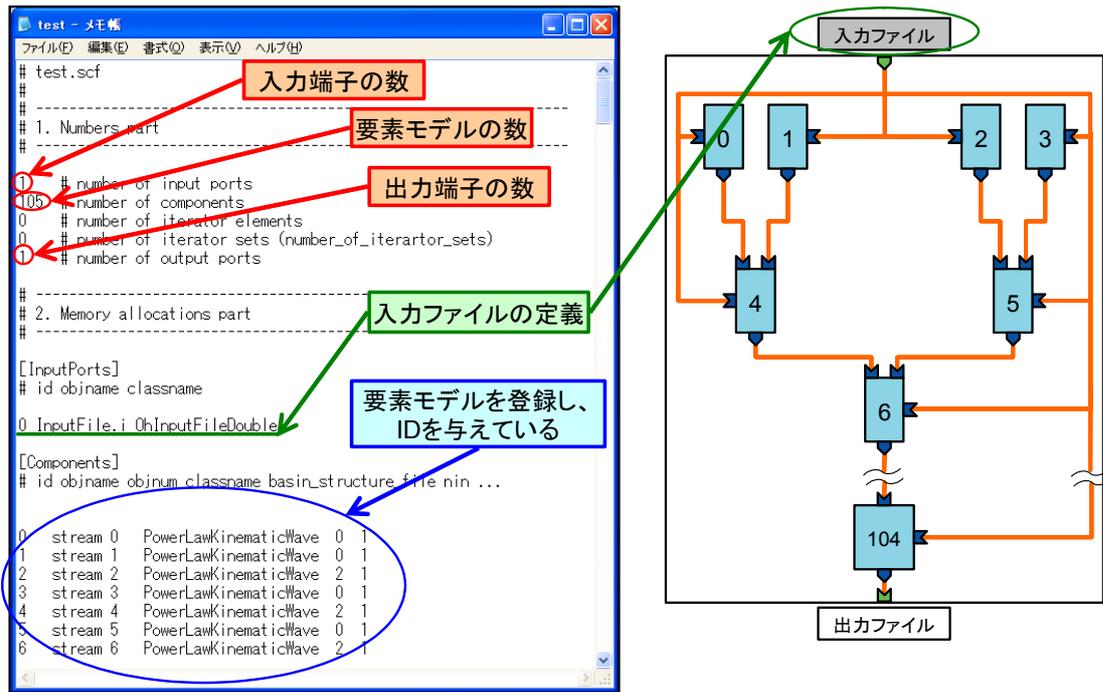


図 2-25 構造定義ファイルの概要

## 2.6.9 コンパイルとクラスライブラリー

OHyMoS ではソースコードをコンパイルし、クラスライブラリーを生成して解析を行うが、OHyMoS と OHyMoSJ では、コンパイル手法が異なり、クラスライブラリーの性質も異なる。ここでは、それぞれのコンパイル法とクラスライブラリーについて説明する。

(1) OHyMoS (C++版)

はじめに、OHyMoS 本体（要素モデルは含まない）のソースコードをコンパイルし、クラスライブラリーのファイル（サンプルでは ohymos.lib、以下同様）を生成する。その後、要素モデルのソースファイルを含むディレクトリで実行ファイル（ohymos.exe）を生成し、この実行ファイルをコマンドツールから実行することにより、OHyMoS を稼働させる。

(2) OHyMoSJ (Java 版)

一方、OHyMoSJ では OHyMoSJ 本体のソースコードと要素モデルのソースコードを同じディレクトリで管理し、OHyMoSJ の本体と要素モデルのソースファイルを一度にコンパイルし、クラスライブラリー化を行う（ohymosj.jar ファイルを生成）。OHyMoSJ のクラスライブラリーは jar ファイルと呼ばれ、中間ファイルであるクラスファイル（.class）を圧縮したファイルであり、OHyMoSJ では jar ファイルをクラスライブラリーと考える。OHyMoSJ 実行の際はこの jar ファイルを呼び出すことによって解析を行う。OHyMoSJ の実行は、OHyMoS 同様、コマンドツールを用いる。

新規要素モデルを追加しようとしたとき、Java のソースファイルを同ディレクトリの中に置き、再コンパイルを行うと、新規要素モデルを含んだクラスライブラリーを生成することができる。この際、新規要素モデルソースファイル以外のものについてはソースファイルである Java ファイルは必要ない。必要なのは class ファイルと呼ばれる抽象クラスのファイルである。class ファイルはコードを見ることができないファイルであるため、要素モデルを公開したとしてもプログラムソースを公開せずにモデルだけ公開することができる。コンパイルし、ライブラリーを生成することに関しては、OHyMoSJ の容易に扱うことができると考えられる。

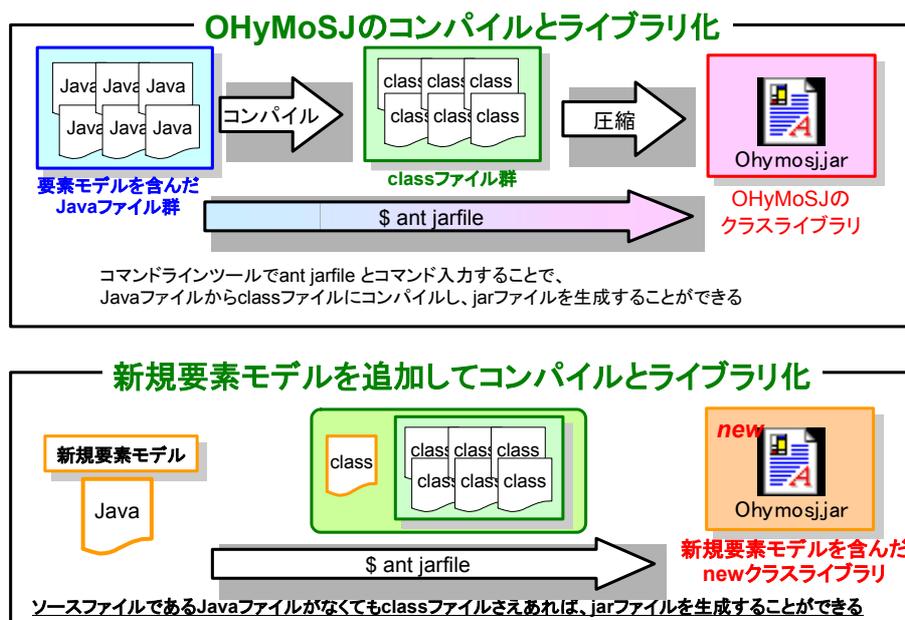


図 2-26 OHyMoSJ のコンパイル

### 2.6.10 ユーザーによるモデルの作成方法<sup>3)</sup>

OHyMoS では、基本型要素モデルのクラスを継承すれば、以下のような方法で要素モデルが作成できる。

- ① 基本型要素モデルを継承する。
- ② 受信端子・送信端子を定義する。
- ③ 受信端子・送信端子を登録する関数を定義する。
- ④ パラメータ・状態量を定義する。
- ⑤ パラメータを設定する関数、状態量を初期化する関数を定義する。
- ⑥ 初期の送信を行う関数を定義する。
- ⑦ 次の計算のためのタイムステップを算出する関数を定義する。
- ⑧ 1ステップ分の計算を実行するか判断する関数を定義する。
- ⑨ 1ステップ分の計算・送信を行う関数を定義する。
- ⑩ 初期化後の一連の計算作業を行う関数を定義する。

### 2.6.11 まとめ

OHyMoS の特徴をまとめると、以下のようなものである。

- ・ 要素モデルを組み合わせて全体系モデルを構成するようになっている。
- ・ 水文系モデルの要素モデルとして普通に要求されるような動作、パラメータの設定、初期値の設定、計算時間の更新などについては標準化されている。データの入出力機能は、新規にユーザーが定義するデータ型に対しても対応できるようになっている。
- ・ 要素モデルは、個々に独自の時間単位で計算を進めることができる。
- ・ ある要素モデルの出力データを他の要素モデルが入力データとして利用するという関係は、端子によるデータ授受の関係としてモデル化されている。
- ・ 端子によるデータ授受では対応できない場合に対応するため、要素モデル同士が直接データを交換する直接通信の機能がある。
- ・ 流出系の最終状態をファイルに書き出し、後で計算を再開することができる。

## 2.7 標準フレームワークの開発に向けて

### 2.7.1 標準フレームワークの目指すもの

以下に、標準フレームワークの開発から究極的な目標到達までの道程のイメージを示す。このような道筋については様々な議論があり、これに拘泥するものではないが、「何を目指して」という部分はとても重要であり、その検討を本格的に行うためにも、こうしたイメージを出してみる事が大事と考えられる。

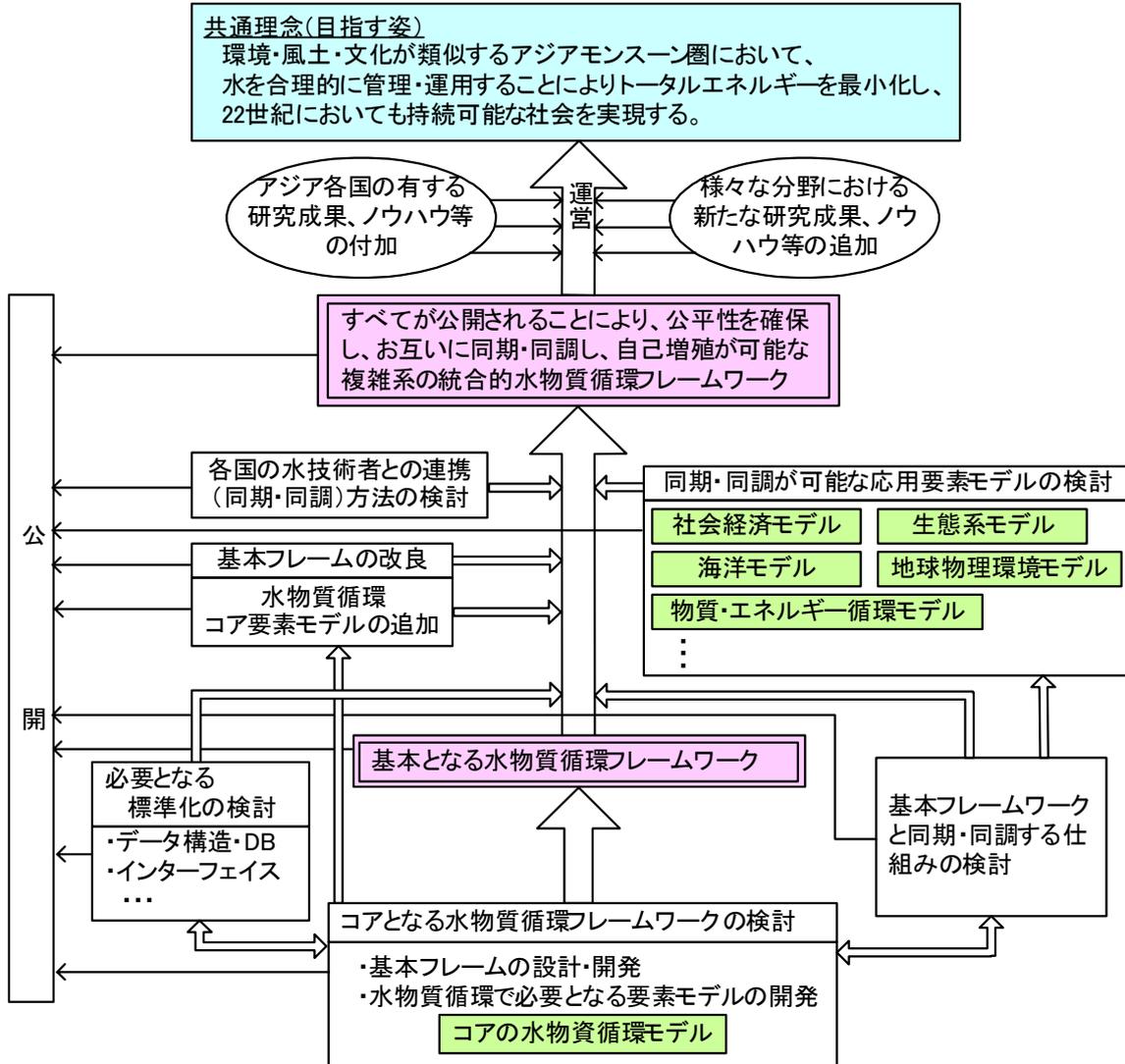


図 2-27 理想形までの道程のイメージ

## 2.7.2 標準フレームワークと既存シミュレーションモデルとの関係

ここでは、標準フレームワークと既存のシミュレーションモデルとの関係について記す。

### (1) 標準フレームワークと既存シミュレーションモデルの共存

標準フレームワークはシミュレーションモデルの共通基盤であり、いわばコンピュータのOSのような位置づけである。標準フレームワークを作成したとしても、既存のシミュレーションモデルやプログラムの存在を否定するものではなく、共存していける関係にある。図 2-28 に示すように、標準フレームワークと既存モデルの機能はお互いに参照・フィードバックしあうことにより、より良いものへと共存共栄することができる。

また、大学などで作成されるモデルに関しては、標準フレームワークの流れに乗らないものも想定されるが、そういったモデルの存在を否定するものではない。

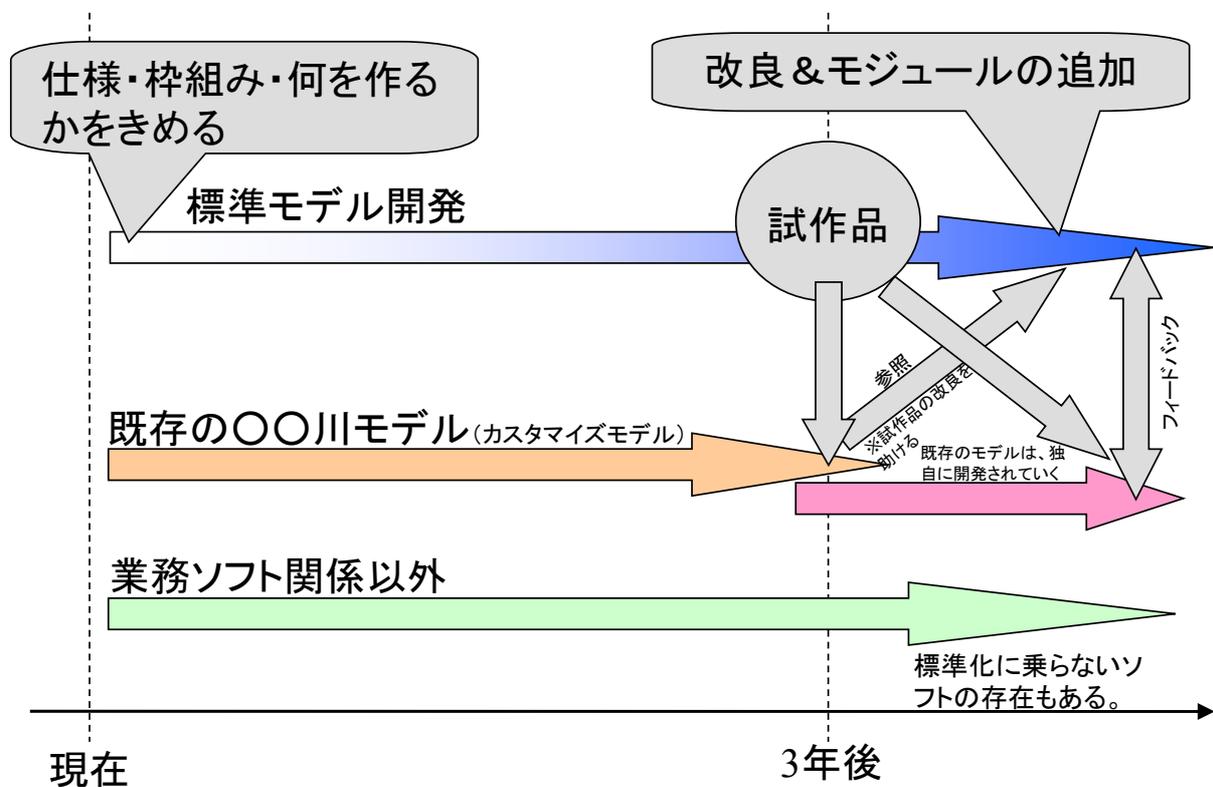


図 2-28 標準フレームワークと既存モデルの関係

### (2) 既存モデルの標準フレームワークへの取り込み

標準フレームワークはシミュレーションモデルの共通基盤であり、シミュレーションモデルそのものではない。従って、開発を目指す標準フレームワークでは、これまで作成してきた解析プログラムも利用できるものとするのが資源の活用という点から重要な点と考えられる。

これまで多くの水理・水文・水質解析プログラムは Fortran で作成されており、これをそのままの形でフレームワーク上に乗せることはできない。フレームワークに対応した言語で作成するかもしくは、ラッピングと呼ばれる技術を導入するかのどちらかと考えられる。

ラッピングに関する技術的な情報は本報では詳細には論じないが、ラッピングとはある言語で書かれたプログラムを他の言語で書かれたプログラムと連動させるための技術である。つまり、ソースコードに触れずに実現できる技術ではなく、ラッピングと呼ばれる処理を実装する必要があり、相応のプログラミング力が要求されると考えられ、容易な作業ではないと想定されるが、新たに再プログラミングするよりは、手間がかからないと言われている。しかし、ラッピングは万能ではなく、相性の悪い言語も存在すると言われている。

### 2.7.3 標準フレームワークの開発言語

ここでは、標準フレームワークの開発言語について検討する。これまでも述べてきたように標準フレームワークは水理・水文・水質解析の共通基盤となるフレームワークであり、完成された解析ソフトウェアを開発するわけではない。また、様々な技術レベルのユーザーによる利用を想定したものであり、フレームワークに自ら要素モデルを加えるなどして利用するユーザーもいるだろう。従って、フレームワークを開発する言語について、その特徴について調査検討しておく必要がある、それぞれ比較しておく必要がある。

2.5 で説明した通り、オブジェクト指向型言語の特徴の中で象徴的なクラスという概念は機能の追加の容易性という利点を持っている。これは標準フレームワークの実現に不可欠な要素である。その理由について以下に記す。

#### (a) 様々な機能を分割して作成

標準フレームワークには解析を実行する以外にも様々な機能が必要となる。例えば、データの入出力する機能、グラフを描画する機能等であるが、これらをクラスの単位で分割して作成しておくことにより、バージョンアップの際に機能を向上させるクラスだけを書き直せばよい。また、Windows で利用するとした場合、既に公開されているクラスをそのまま使えばいいし、データベースと連携するためのクラスなどもあるため、データベースエンジンごとに対応するクラスを持っておけば、様々なデータベースを利用することが可能となる。

#### (b) 要素モデルの作成が容易

標準フレームワーク上で稼働する要素モデルを作成するには、標準フレームワークの仕様に則ってプログラミングする必要がある。つまり要素モデルを実装する場合、最低限必ず実装しなければいけない機能がある。こういったプログラム作成にある程度の縛りがある場合は、継承の機能が有効利用できる。予め要素モデルの親クラスとなるクラスに最低限必ず実装しなければいけない機能を実装しておけば、子クラスは何もしなくてもその機能を持つことができるため、ユーザーは自分が追加する機能、すなわち計算部分だけを実装すればよいということになり、要素モデル実装の生産性が格段に向上する。

この考え方は OHyMoS・OHyoMoSJ ですすでに取り入れられており、基本型要素モデルという親クラスが実装されてある。

## (2) 候補となる言語の比較

ここでは、標準フレームワークの開発言語の候補となるプログラミング言語についてそれぞれの特徴を挙げて比較する。

比較する言語は前述の検討を踏まえオブジェクト指向型言語に絞った。候補として検討する言語は、C++、C#、Java の 3 種類とする。この 3 言語は様々なシーンで使われている言語で、メジャーなオブジェクト指向型言語のトップ 3 といってよい。ユーザー自身も実装することができる標準フレームワークの言語としては極力メジャーなものを利用する方が望ましいことは論を持たない。比較結果を表 2-2 に示す。

表 2-2 言語の比較

比較項目	.NET Framework (C#)	C++	Java
概要	.NETはMicrosoft社が開発したアプリケーション開発環境。プログラミング言語ではない。 C#は.NET対応のプログラミング言語。	C言語をオブジェクト指向型言語として拡張した言語。近年最もよく使われている商用言語の一つといわれている。	Sun Microsystems社が開発したプログラミング言語であり、開発環境でもある。
開発・管理体制	Microsoftの商品であり、1社で開発・管理を行っている。	様々なベンダーがコンパイラや、開発環境のGUIを開発している。最も普及しているのはMicrosoft社の「Visual C++」	開発はSun Microsystems社であるが、その後は様々なユーザーによって言語が成長していった。
特定の組織の影響力	Microsoft社製の商品であるため、Microsoft社の影響を直接受ける。Microsoft社が開発を辞めたら、そこで終了する。	C#ほどではないが、最も普及しているコンパイラがMicrosoft社のVisual C++であるため、Microsoft社の影響を受けるかもしれない。	Sun Microsystems社が開発したが、言語に対して影響力は少ない。既にSunの手を離れてコミュニティによって開発されており、突然なくなることはない。
対応OS	基本的にWindows対応だが、今後Windows外でも利用できるようになる可能性もある。WindowsOSが持っている機能を活用しやすい。	どの環境でも動くが、プログラムを動かすOSごとに合わせたプログラミングが必要。例えば、Linuxで開発したプログラムがWindowsで走らないことがある。	すべてのOSで利用できる。特定のOSで特にパフォーマンスを発揮するということはない。
クラスに関して	Windows上で利用できるGUIなどが初めから備えられており、.NET対応言語であれば共通して利用することができる。	異なる環境(OSやコンパイラ)でコンパイルされたクラスを共通に利用することはできないことがある。	世界中で作成された多種多様のクラスが存在する。無償で手に入るものが多い。
公開されているクラスの品質	Microsoftが管理しているものに関しては、品質は保証されていると考えて良い。	公開されているクラスライブラリが少ないと言われている。また、ヘッダファイルはソース提供しなければならない。	個人作成のものに関しては品質保証されているとは限らない。
過去の資産の活用	過去の資産をそのまま使うことはできない。ラッピング、再プログラミングが必要。	C++で書かれているものであっても、コンパイラなどが違うと、そのまま利用できない可能性がある。	過去の資産をそのまま使うことはできない。ラッピング、再プログラミングが必要。
文法の分かりやすさ	機能が多く、やや煩雑と言われる。	C言語が解る者には分かりやすいと言われるが、全般的に複雑というのが一般的。	シンプルな文法構造になっており、理解しやすいと言われている。
プログラミング	機能が多く、複雑な処理でも比較的簡易に実装できる。	他言語と比べて、プログラミングが複雑といわれ、実装に時間がかかるといわれる。	文法がシンプルな分、複雑な処理には回りにくいプログラミングが必要。
実行環境	.NET framework上で、Windowsで実行できる。	開発時の環境以外では、そのまま実行できないことがある。	Java Virtual Machine上で、OSに依らず実行できる。
DBとの親和性	DBとの親和性は高いと言われている。特にSQL ServerはMicrosoft社の製品であり、親和性は高い。	Microsoft社のコンパイラとDBであれば、親和性は高い。	データベースにアクセスできるAPIを実装することができる(JDBC)
対応言語に関して	.NET対応言語は幾つかある。 ・C#、C++、.NET、VB.NET、など。 C++とC++、.NET、VBとVB.NETは異なる言語	C++そのものが、言語である。	Javaそのものが、言語である。
水理解析に用いられている頻度	現在はほとんど使われていないと考えられるが、C++、VBからの移行がスムーズになれば増加すると考えられる	C#、Javaよりは利用されていると考えられる。C言語ユーザーがC++に移行して利用している可能性がある。	現在はあまり使われていないと考えられるが、今後は徐々に増えていくと考えられる

## 2.7.4 標準フレームワークの仕様の検討

### (1) 要素モデルに関する仕様の検討

#### 1) 要素モデルは独立したソースファイルで作成する

要素モデルは独立したソースファイル（クラス）で作成することが求められる。ユーザーが要素モデルを新規作成する場合、一つのファイルを作成すればよい。

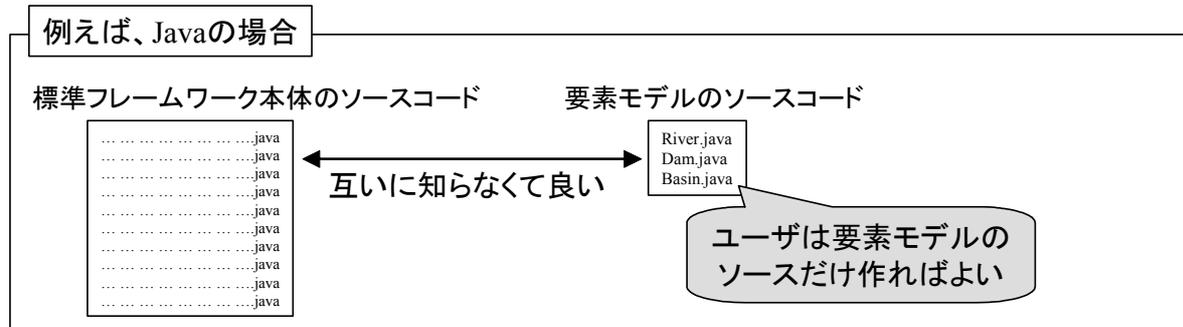


図 2-29 要素モデルは独立したファイルとするイメージ

#### 2) 要素モデルは機能的に独立している

要素モデルは、他の要素モデルから独立していなければならない。接続する相手の要素モデルの中身を知らなくても接続できなければならない。



図 2-30 他のモデルの中身を知らなくて良いイメージ

#### 3) 入力すべきデータ、出力するデータを明らかにする

要素モデルに入力すべきデータ項目、要素モデルが出力するデータ項目を明らかにしなければ、他の要素モデルと接続することができない。

#### 4) パラメータ値はモデル外部から設定可能とする

パラメータ値はソースコード中で設定せずに、ソース外から設定できるようにする。内部で設定すると、条件を変えて解析を行うたびにソースコードを書き換える必要があるためである。

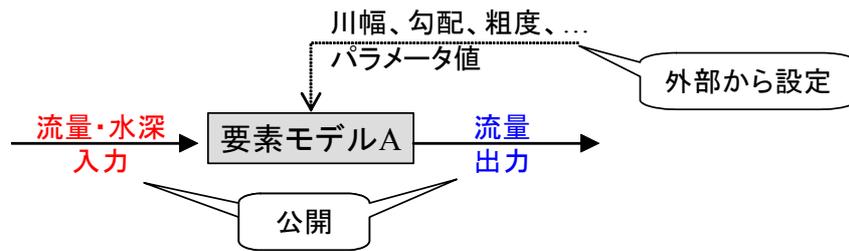


図 2-31 要素モデルの入出力データ項目の公開

### 5) 継承と多態性を持たせる

要素モデルのクラスを継承することにより、新たな要素モデルを作成できる。ある要素モデルをカスタマイズする場合、追加する機能だけを実装すれば、カスタマイズされた新たな要素モデルを容易に作成することができる。

### (2) 全体モデル構築に関する仕様の検討

#### 1) 要素モデルをつなぎ合わせて全体モデルを構築できる

標準フレームワークの仕様に基づく要素モデルをつなぎ合わせることにより、全体モデルを構築できるものとする。

#### 2) 要素モデルの交換・追加・削除ができる

要素モデルの組み合わせはユーザーが自由に設定できるものとし、要素モデルの交換・追加・削除も容易に可能なものとする。

#### 3) 異なる $\Delta t$ の要素モデルを接続することができる

異なる  $\Delta t$  の要素モデルを接続することができれば、要素モデル作成時にその要素モデルに最適な  $\Delta t$  を設定することができる。

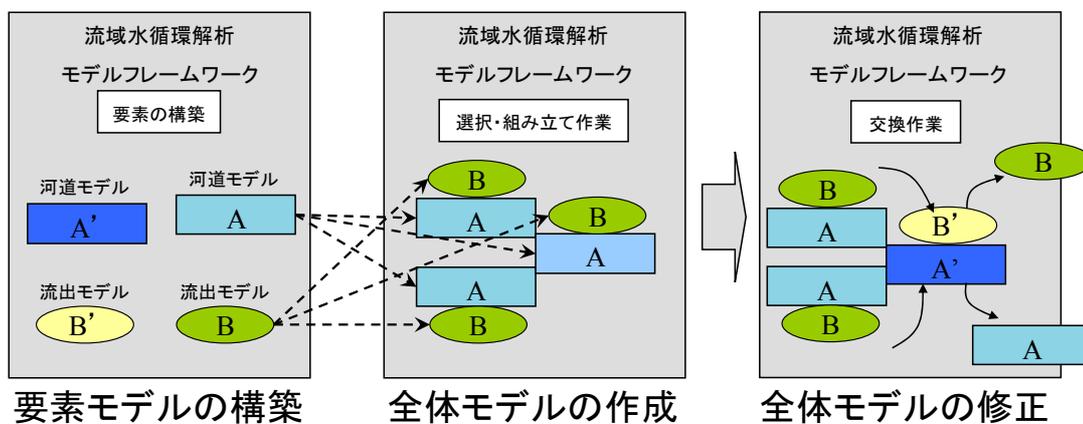


図 2-32 全体モデル構築のイメージ (図 2-12 の再掲)<sup>3)</sup>

### (3) データ交換端子に関する仕様の検討

#### 1) 要素間データの授受、モデルへのデータの入出力は端子で行う

解析における要素モデル間データの授受は端子で行うものとする。解析に用いる入力データ（水理解析でいう外力：雨量、流量など）と、モデルから出力する出力データも端子で行う。

#### 2) 1次元、2次元、3次元の時系列データを取り扱える

水理・水文・水質に関する複雑な現象の解析にも対応する必要がある、多次元の時空間データを取り扱えるだけでなく、異なる次元のモデル同士の接続をも可能とする。また、水理・水文・水質だけでなく環境、生態系、経済などの河川、流域管理のための政策支援に資するための複合的な解析にも今後対応していく必要がある。

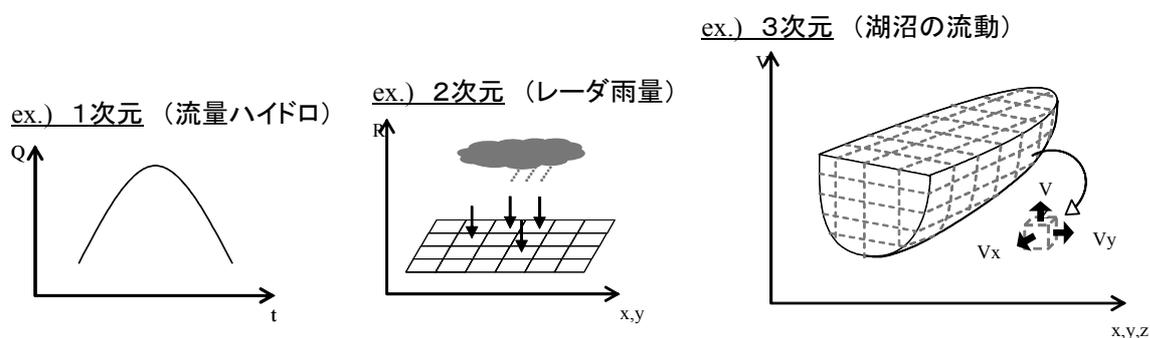


図 2-33 各次元データのイメージ

#### 3) 単位系を規定する

端子を通じて入出力、送受信するデータの単位を規定する必要がある。要素モデルの中身を知ることなく利用できるという標準フレームワークのコンセプトを守るためには単位の統一は不可欠である。

### (4) 計算の進め方に関する仕様の検討

#### 1) 各要素モデルが時間的に平行に進む

接続関係にある要素モデル同士が毎ステップごとにデータの授受を行いながら計算を進めていく。

#### 2) 要素間の反復計算を行うことができる

1次元河道モデルと2次元氾濫解析モデルを相互接続するような水理解析においては、複数の要素の間で情報を交換しながら計算を反復する機能が必須である。

#### 3) 計算の中断再開ができる

計算を途中で中断しても、中断した時点から再開できる機能を持たせることにより、長時間の解析が必要な場合でも、一時中断することができる。

## (5) ユーザーインターフェースに関する仕様の検討

### 1) 様々なレベルのユーザーでも利用できる

水理解析に堪能なユーザーから、水理解析の知識を有さないユーザーまでストレスなく利用できるユーザーインターフェースを作成する必要があり、グラフィカルユーザーインターフェース（以下、GUI と表記）を開発する必要がある。

### 2) 要素モデルの接続が GUI 上で操作できる

要素モデルの接続を GUI 上の操作で設定できる必要がある。要素モデルの交換・追加・削除を容易に行えるものとする。

### 3) 入力データの選択ができる

入力データの設定やどの要素にデータを入力するかの設定について GUI 上の操作で行うことができる。

### 4) 結果をグラフ化できる

ハイドロなどの出力結果をグラフによる視覚化を GUI 上で行うことができる。

### 5) プロジェクト管理ができる

要素モデルを接続し、全体モデルを構築した状態や、パラメータの設定などの状態を記録できるプロジェクトファイルが作成できるものとする。また、プロジェクトファイルは異なる PC 間で互換性のあるものとする。

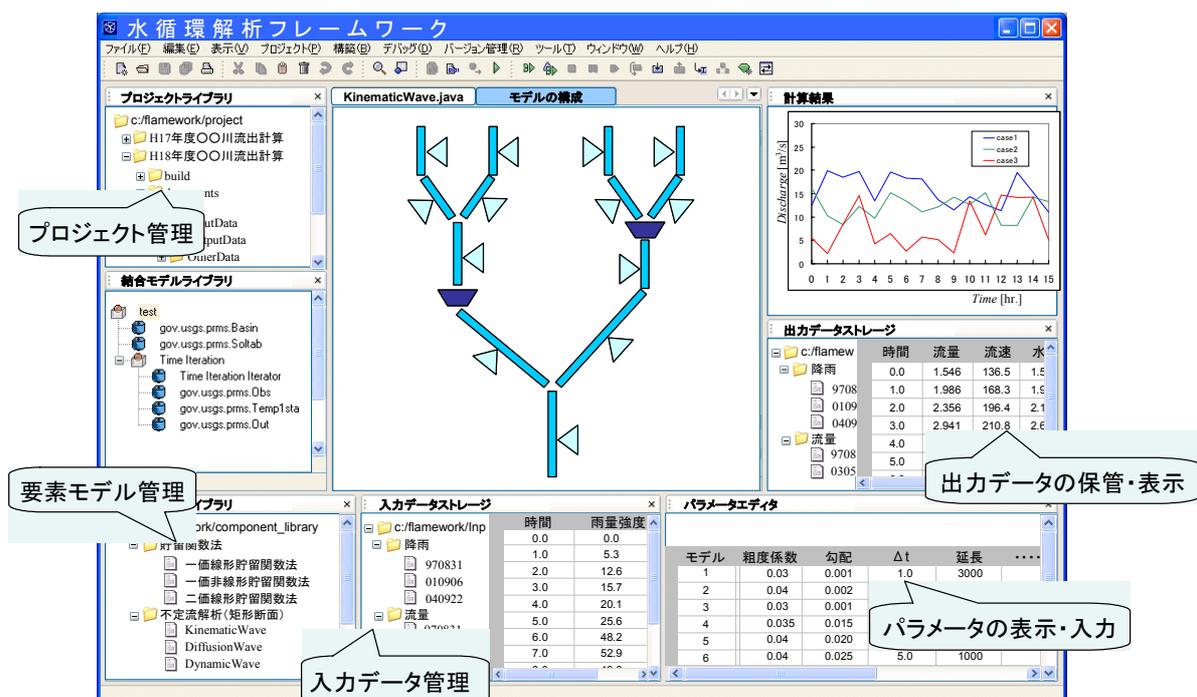


図 2-34 GUI のイメージ

## (6) その他の機能に関する検討

### 1) 汎用型シミュレーションモデルのセットアップが容易にできる

汎用型シミュレーションモデルの PC へのインストールを容易にできる必要がある。また、詳細な PC の設定をすることなく解析を実行できることが望ましい。

### 2) 既存のプログラムを利用することができる

これまで作成されてきた解析モデルのプログラムを標準フレームワーク上で利用できる必要がある。ただし、標準フレームワーク上で動くように加工する必要がある、その一つの手法としてラッピングがある。

以下にラッピングのイメージを示す。

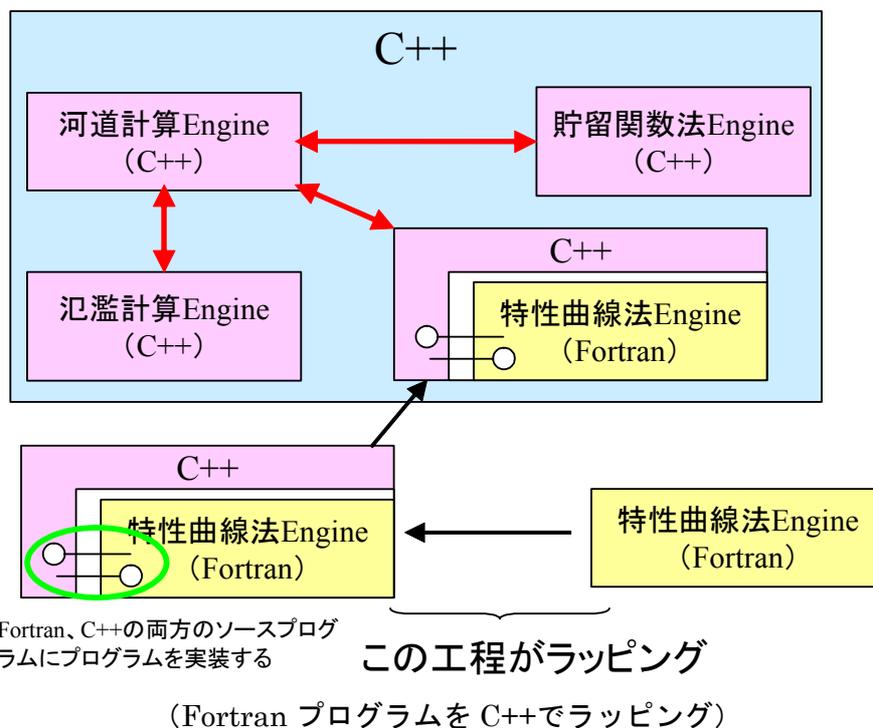


図 2-35 ラッピングのイメージ

### 3) データベースと連携できる

標準フレームワークからデータベースにアクセスできるものとする。その操作は GUI 上でできるものとする。

### 3 海外の水理・水文・水質シミュレーションモデルの開発・運用体制

#### 3.1 欧州における水理・水文・水質シミュレーションモデルの概要

##### 3.1.1 デンマーク水理・環境研究所

デンマークの DHI Water & Environment 社（デンマーク水理・環境研究所、以下 DHI と表記）のインターフェース及びプログラム構成について整理する。

##### (1) DHI ソフトウェアの概要

以下に、DHI の主なソフトウェアを整理する。DHI のソフトウェアは全て有償である。

表 3-1 河川下水道分野における DHI Water & Environment 社の主なソフトウェア

分野	ソフトウェア名称	機能概要
河川	MIKE11	河川を対象とした水文・水理・水質・土砂解析統合ソフト。 水文解析：降雨解析、確率統計解析（EVA:Extreme Value Analysis）、流出解析、洪水予測 水理解析：1次元流れの不等流計算・不定流計算 土砂解析：均一粒径や混合粒径の土砂を対象とした河床変動解析 水質解析：BOD、DO、クロロフィル a、栄養塩、重金属等を対象
	MIKE21	2次元流れの水理・水質・土砂解析統合ソフト
	MIKE3	3次元流れの水理・水質・土砂解析統合ソフト
	MIKE BASIN	流域の利水問題検討ソフト
	MIKE SHE	流域水循環解析統合ソフト（分布型流出解析・地下水解析（2次元又は3次元）、物質の拡散・移流解析）
	MIKE FLOOD	洪水氾濫解析ソフト（1次元と2次元の統合）（MIKE11とMIKE21の統合）
	その他	MIKE12（1次元の二層密度流）、MIKE Reservoir（1次元の多層密度流）、MIKE11 GIS（MIKE11のためGISインターフェース）、MIKE21C（直交曲線座標系での2次元河床変動解析）、FLOOD WATCH（実時間洪水予測）等
雨水排除・下水道	MOUSE	下水道を対象とした水文・水理・水質解析統合ソフト
	MIKE SWMM	同上 米国環境保護庁（USEPA）の開発した SWMM（Storm Water Management Model）と DHI のインターフェースの統合ソフト 機能は MOUSE と同様、相違点は解析エンジン
	その他	MOUSE RTC（リアルタイムコントロール）、MOUSE GIS（MOUSEのためのGISインターフェース）、都市域氾濫計算（MOUSEとMIKE21の統合）

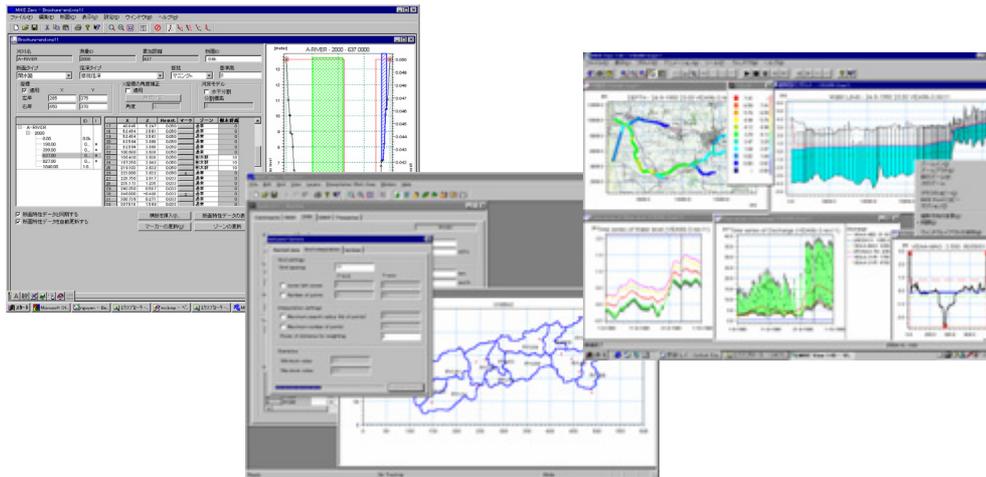


図 3-1 MIKE11 のインターフェースイメージ

また、DHI ソフトウェアの課題として以下の点が挙げられる。

- ・ ソフトウェア購入費用が高い。
- ・ ソースコードは非公開である。
- ・ 商業ベースなので、広く認知される技術しか取り込んでいない。(最先端の技術を取り込んでいくわけではない)
- ・ 平面 2 次元モデルは、直角座標系や直交座標系である。
- ・ 一般座標系や非構造格子には対応していない。

## (2) DHI におけるインターフェースの標準化

DHI のモデリング群の特徴は以下の通りであり、共通のプラットフォーム MIKEZero を有している。

これにより、以下のことが可能となっている。

- ✓ ソフトウェアの品質確保・向上
- ✓ ソフトウェア同士の容易なカップリング  
例：MIKE11+MIKE21、MOUSE+MIKE21、MIKE11+MIKE SHE 等
- ✓ ユーザーが OS に対して多くを知る必要がなく、どのソフトウェアも容易に取り扱うことが可能

以下に、MIKEZero におけるインターフェース標準化の具体例を示す。

### ■MIKE シリーズの共通インターフェース

- ・ 全モデリングパッケージに対し、共通のインターフェースを構築しており、ユーザーは一つのインターフェースから任意のモデル(ソフトウェア)を組み合わせることが可能である。
- ・ また、どのモデルも同形式のモデル作成メニューとなっている。これによりユーザーは一つのモデルの使用方法を理解すれば、容易に他のモデルへも移行が可能である。
- ・ 更に、時系列データ (0 次元、1 次元、2 次元)、グリッドデータなどは、全モデル共通の形式となっており、モデルを超えてこれらを共有化することができる。

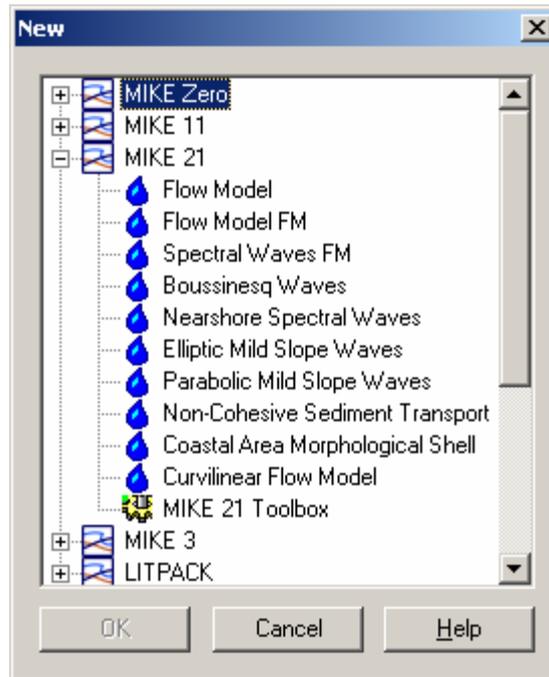


図 3-2 MIKE シリーズの共通インターフェース(1) (モデル選択メニュー)

#### ■MIKE シリーズの共通モデル構成

各モデリングパッケージは、以下に示す 6 つのメニューから構成され、ユーザーは容易にモデル構成を理解、構築することが可能である。

- シミュレーション制御
- 平面情報(ネットワーク等)入力・編集
- 地形データ(横断データ、地盤データ等)入力・編集
- 境界条件入力・編集
- 計算パラメータデータ入力・編集
- 結果表示

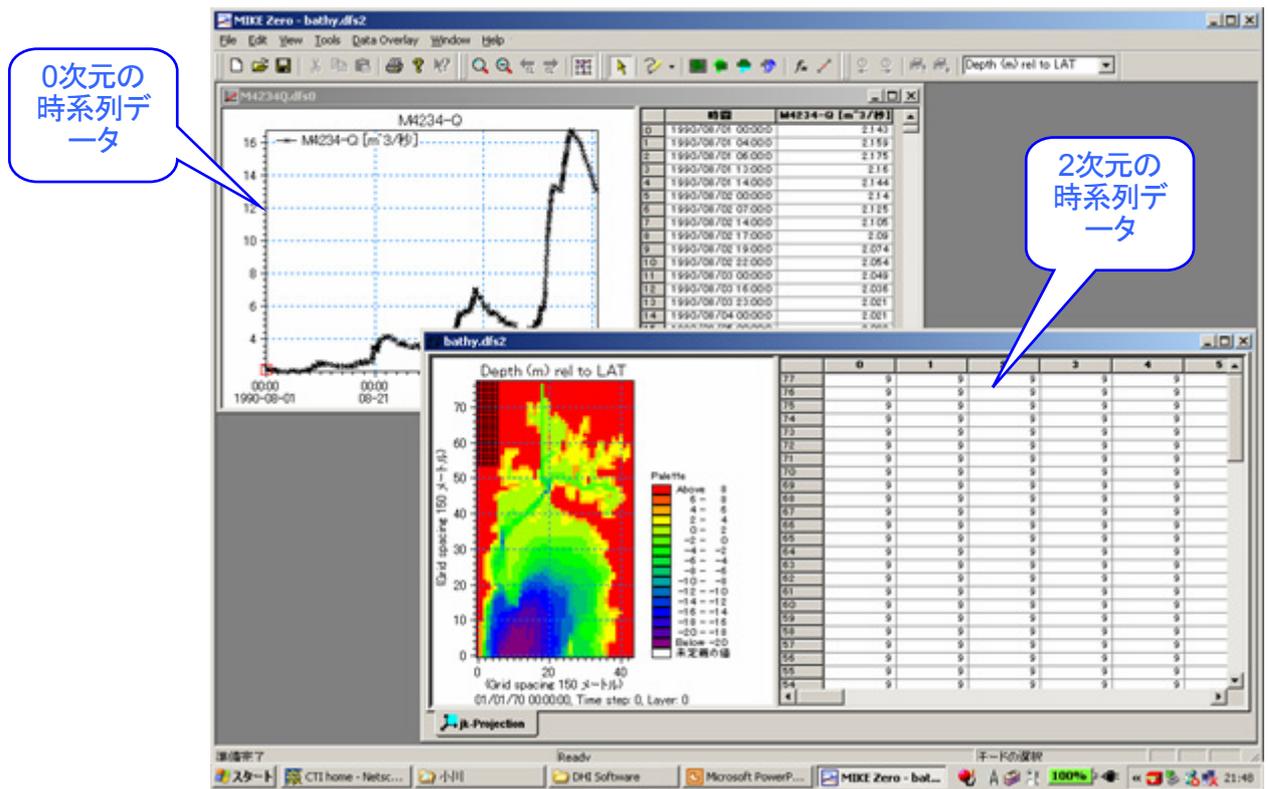
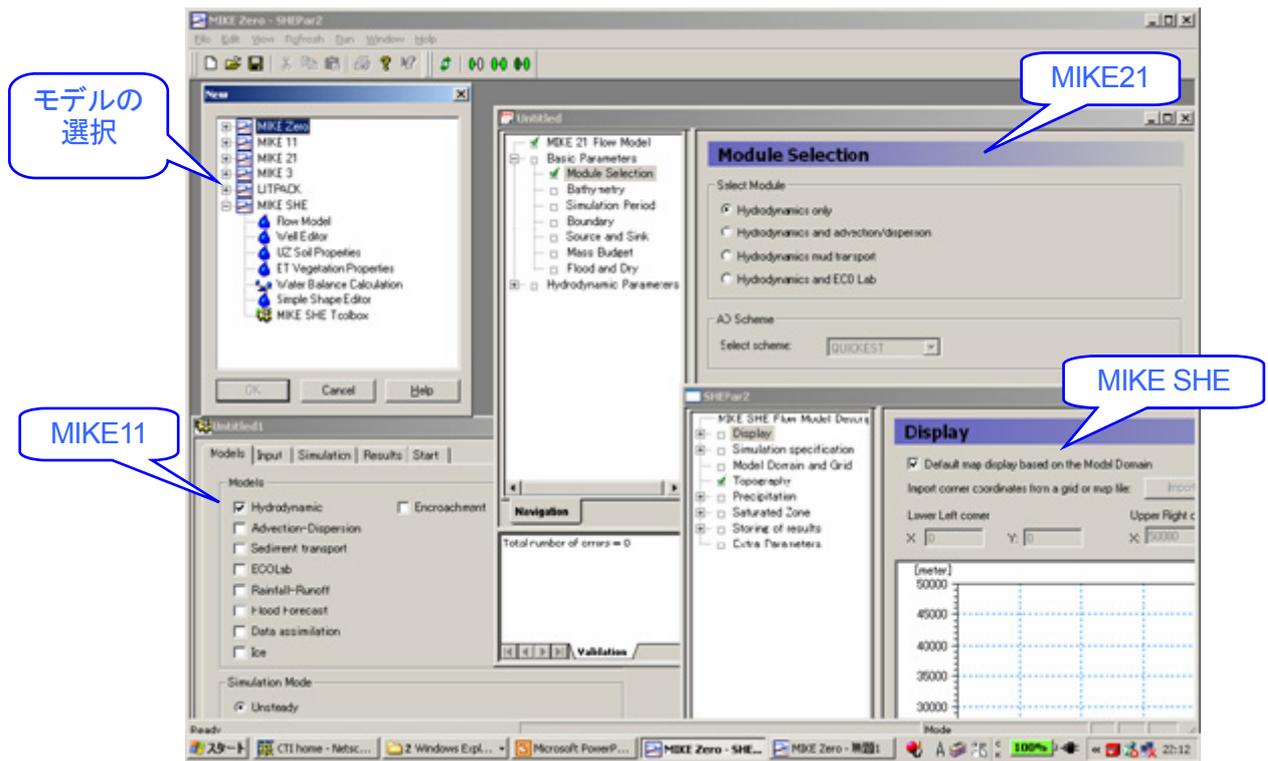


図 3-3 MIKE シリーズの共通インターフェース(2)

### (3) MIKE11 のプログラム構成

流出、水理、水質、河床変動計算など全ての計算が連結しており、同一パッケージ内で任意の処理できるよう構造化されている。

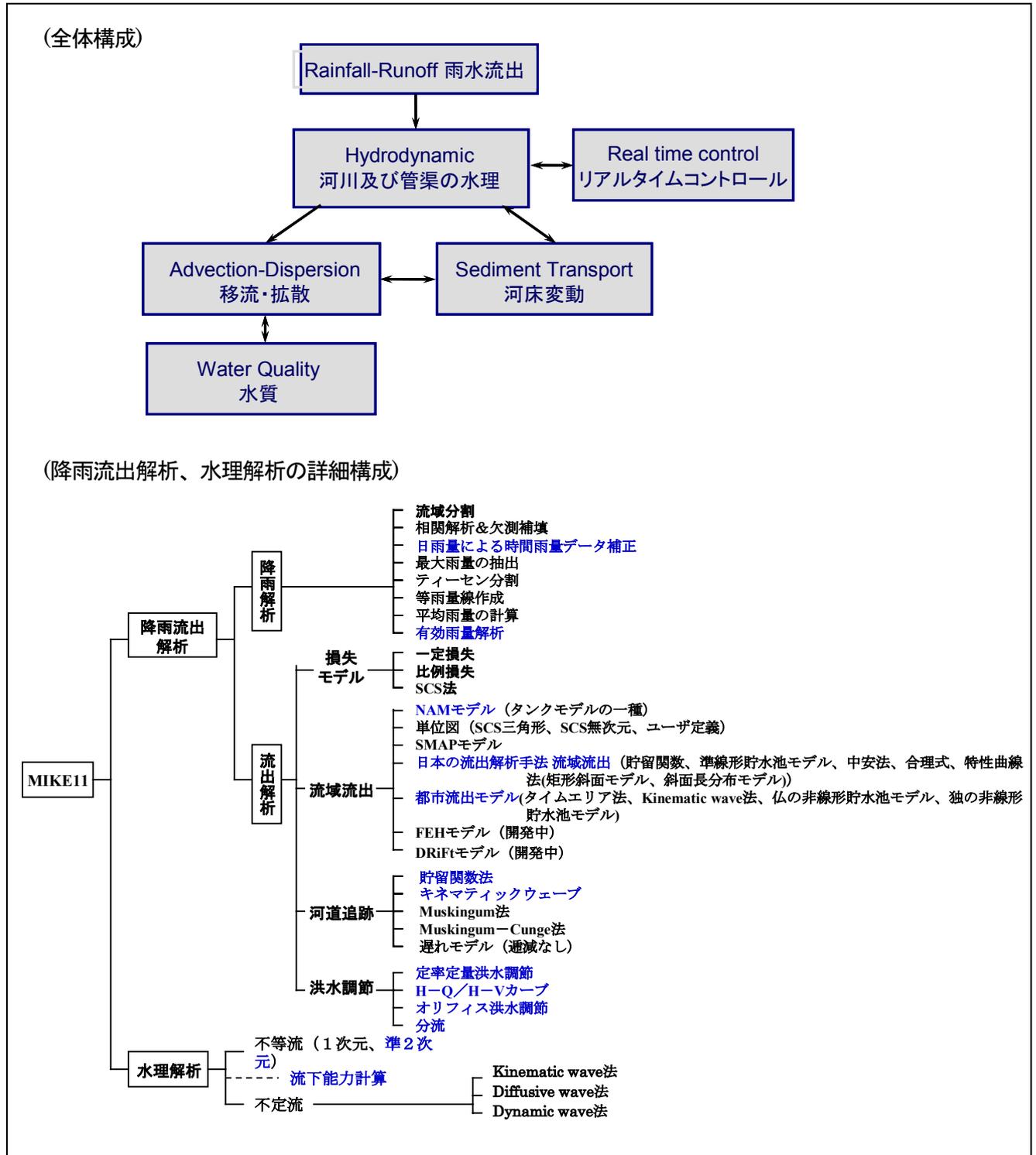


図 3-4 MIKE11 のプログラム構成(1)

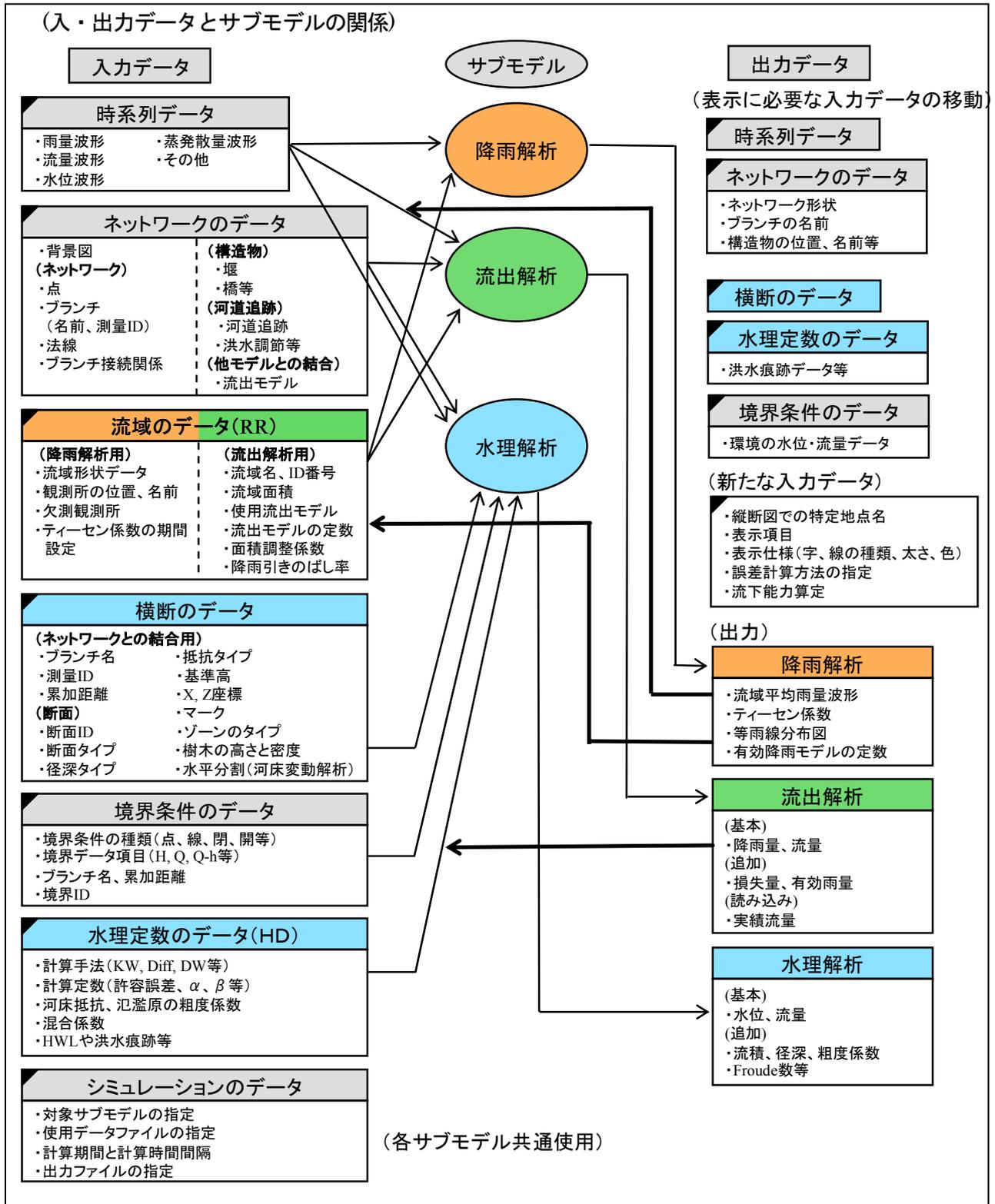


図 3-5 MIKE11 のプログラム構成(2)

### 3.1.2 OpenMI の概要

欧州における水理解析のフレームワークである OpenMI (Open Modeling Interface and Environment) に関して、その概要と機能について、現時点で把握できている内容を整理する。OpenMI とは、ヨーロッパ各国の組織と EU の出資で作成された水工系解析ソフトウェア統合インターフェースである。既存のソフトウェアを OpenMI 対応とすることにより、OpenMI 対応ソフトウェア同士であれば、ソフトウェア相互にデータを連携しながら解析を進めることができるものである。

あくまでデータ交換のためのインターフェースであり、OpenMI そのものに数値解析を行うソフトウェアは持たされていない。

#### (1) OpenMI のコンセプトと目的

OpenMI の目的は、物理・社会経済プロセスのモデルがお互いに他のデータソースと同期するメカニズムを与えることにより、一つのソフトウェアでは実現できない現象をシミュレーションすることである。

- 異なる領域からのモデル (水理学、水文学、環境学、水質、経済学など) と環境 (気象、淡水、海洋、都市、農村など)
- 異なるモデリングの概念に基づくモデル (決定論、統計論など)
- 異なる次元のモデル (0,1,2,3 次元)
- 異なるスケールのモデル (例えば局地的な気象モデルを流域の流出モデルへ)
- 異なる一時的な分解能でのモデル操作 (例えば 1 時間から月単位や年単位まで)
- 異なる空間表現におけるモデルの操作 (例えばネットワーク、グリッド、ポリゴン)
- 異なる計画、単位ユニットや領域に使うモデル
- ほかのデータソースへリンクするモデル (例えばデータベース、ユーザーインターフェース、インストゥルメント)
- 異なるプラットフォームで計算するモデル (例えば Windows、Unix、Linux)
- オープンであること。(仕様がパブリックドメインであること)
- 少なくとも、C 言語、C#、C++、Fortran、Delphi、Java、Visual Basic でコンポーネントが開発できる。

#### (2) OpenMI が必要な理由

既に OpenMI を採用した、もしくは採用を検討している組織が OpenMI を必要としたのは、以下の理由からとされている。

- モデル開発における現存の投資の保護および増進(モデルを OpenMI Compliant にするために完全に書き直す必要はない)
- モデル統合過程の簡易化、モデルプロセスの相互作用能力の向上
- 適切なモデルの組み合わせおよび同じ過程に対する異なるモデルの交換の能力は、感度分析と他社製品を研究して自社製品の性能を高めることを促進する

- 開発時間の短縮、およびそれによる意思決定システムへのコスト縮小
- 異なるソースから‘調和し合う’モデルが使用可能となるようにユーザーの選択肢を増やす
- モデル開発者に対して、フレームワークの開発により、個々のモデルがよりよく売れるような機会を増やす
- 専門部門において **Small and Medium Enterprises (SME)** の創造の機会を増やす
- EU 政策の実現および発展に寄与する機会を増やす
- 例えば GUI や事前事後処理ツール等の **OpenMI Compliant** ツールを購入できるので、モデル開発者に彼らのコアな仕事に集中することができる
- モデル組み込みやリンク、結合されたモデルが動くように観察する為の **OpenMI Environment** ツールがある
- 流域全体を対象としたようなモデルについて既存モデルを再開発するコストと比べて変換コストが少ない
- 他組織の I/O 処理を理解する必要はない
- 結合プロセスやインターフェースに影響を与えないモデルコードの仕様

### (3) OpenMI から提供されるもの

OpenMI からは以下のものが提供されており、すべて OpenMI ホームページから無償で入手することができる。

- The OpenMI standard interface specification
  - ◇ OpenMI の仕様書
- The OpenMI Environment
  - ◇ OpenMI の実行環境。OpenMI のソフトウェアといえる。
- The OpenMI documentation
  - ◇ OpenMI に関する各種のドキュメント
    - A. Scope
      - OpenMI の概要
    - B. Guidelines
    - C. org.OpenMI.Standard interface specification
    - D. org.OpenMI.Backbone technical documentation
    - E. org.OpenMI.Development Support technical documentation
    - F. org.OpenMI.Utilities technical documentation
    - G. org.OpenMI.Configuration technical documentation
    - H. Additional designs for OpenMI tools

### (4) OpenMI 利用のイメージ

OpenMI は既に販売流通している水理解析ソフトウェア同士でデータ連携をしながら解析を進めるためのフレームワークである。図 3-7 にそのイメージ図を示す。

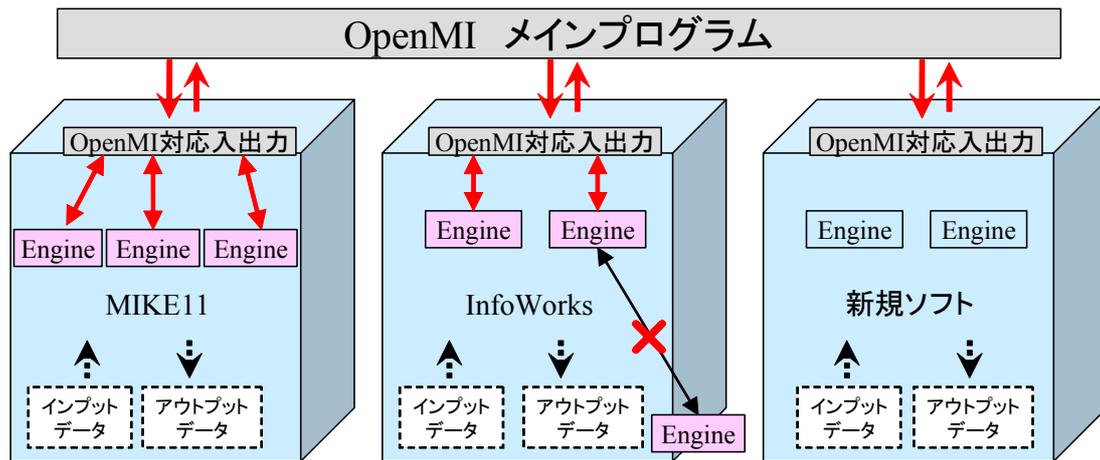


図 3-6 OpenMI のイメージ

上図のように、OpenMI は例えば、DHI の MIKE11 と Wallingford の InfoWorks を接続し、タイムステップごとに同期を取りながら解析をすすめるデータ交換・計算制御インターフェースである。上図に示す MIKE11 や InfoWorks は直接的にデータを授受するのではなく、いったん OpenMI にデータを預け、OpenMI が各ソフトウェアとデータのやりとりを行う。この際、MIKE11 や InfoWorks は OpenMI で利用できるようカスタマイズされている必要がある。

なお、OpenMI そのものは解析ソフトウェアではなく、MIKE11 や InfoWorks が OpenMI の要素モデルに相当する。

## (5) ソフトウェアに OpenMI を実装する

OpenMI は水理解析ソフトウェア間のデータを通信し、計算を制御するプログラムであることは前述の通りだが、OHyMoS のようにプログラムレベルでデータを通信するのではなく、OpenMI を介してソフトウェア間の同期をとり、解析を進めるものである。言ってみれば、データインターフェースであるといえる。ただし、OpenMI を介した解析を行うためには、OpenMI 対応のソフトウェアがなければならない。

### 1) OpenMI Compliant とは

OpenMI Compliant とは、OpenMI で利用できるソフトウェアの総称である。OpenMI Compliant とするには、OpenMI の仕様をソフトウェアに実装しなければならない。その実装仕様はホームページによって公開されているものの、仕様に則り実装することは、水工系の解析者にとっては、容易なことではないと考えられる。

### 2) OpenMI Compliant の実装作業

この実装作業はプログラマかシステムエンジニアレベルの技術が必要である可能性が高いと考えられる。仕様自体は公開されているが、これを C# もしくは Java を用いて実装に落とし込む作業には一定水準以上のプログラミングスキルが必要と考えられる。

従って、あるソフトウェアを OpenMI Compliant にしようと考えたときは、必然的にそのソフトウェアを開発したベンダーが実装することになる。

## (6) OpenMI を利用する

OpenMI から提供されるものは(3)で挙げたが、エンドユーザーが OpenMI を利用する場合は、OpenMI の実行環境である OpenMI Environment をダウンロードし、インストールする必要がある。

### 1) OpenMI の利用制限

OpenMI の利用に関しては、基本的に誰でも制限なく利用することができる。商用利用も可能である。OpenMI Environment は OpenMI のホームページ (<http://www.openmi.org>) からダウンロードすることができる。ダウンロードし、インストールすることにより OpenMI が利用可能となるが、Microsoft が開発したプログラム実行環境である .NET Framework を利用しているため、.NET Framework の環境が整っていない場合は、OpenMI を動かすことはできない。もしくは、Java バージョンを利用する場合は Java の環境を整えておく必要がある。

### 2) GUI を利用する

OpenMI Environment をインストールすると、OpenMI Configuration Editor と呼ばれるアプリケーション (GUI) を立ち上げることができる。OpenMI はこの GUI を利用して、モデル同士の接合や、通信の仕方、計算の制御などを設定することができる。

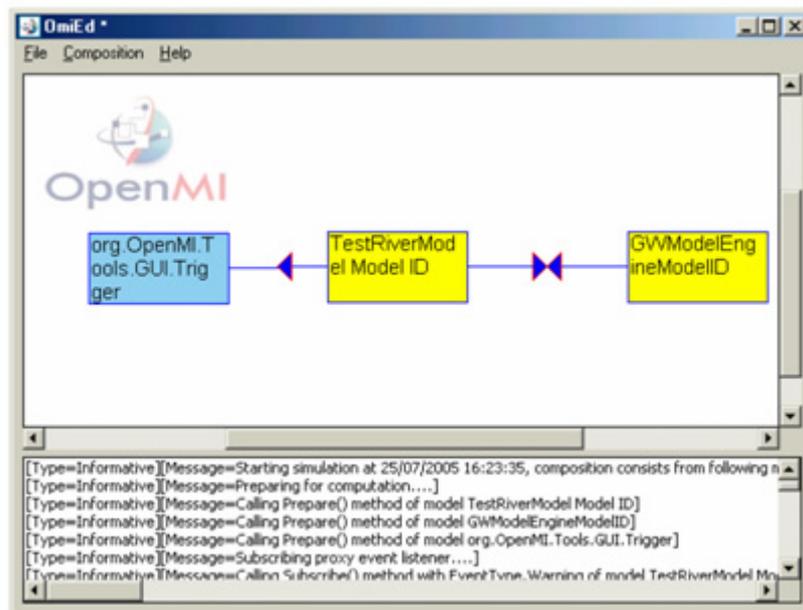


図 3-7 OpenMI Configuration Editor の GUI イメージ

## (7) OpenMI の機能

### 1) モデルの接続

OpenMI では複数のモデルを接続して解析することができ、そのためには各 OpenMI Compliant

の接続関係を定義する必要がある。ここでいう接続関係の定義とは、OpenMI を介してモデル間で交換するデータを関係付けることである。モデルの単位は OHyMoS のようにプログラムではなく、完成されたソフトウェアが必要となる。

あるユーザーが保持している解析モデルと他のユーザーが保持しているプログラムを接続しようとする場合、両方のプログラムを OpenMI Compliant となるよう実装しなければならず、事実上エンドユーザーレベルでは対応することはできない。

## 2) 交換できる要素の単位

前述の通り、OpenMI の要素の単位はソフトウェアとなる。たとえば、ソフトウェア A で考える。ソフトウェア A は OpenMI Compliant として実装されており、OpenMI で利用することは可能である。しかし、ここで計算できる内容はソフトウェア A の機能の範囲内である。言い換えると、ソフトウェア A に含まれている計算エンジンを他の解析方法と交換したいと考えたとしても、ソフトウェア A 自体は単独のソフトウェアであるので、そのソフトウェアそのものの修正はこれまで通り、エンドユーザーではすることができない。

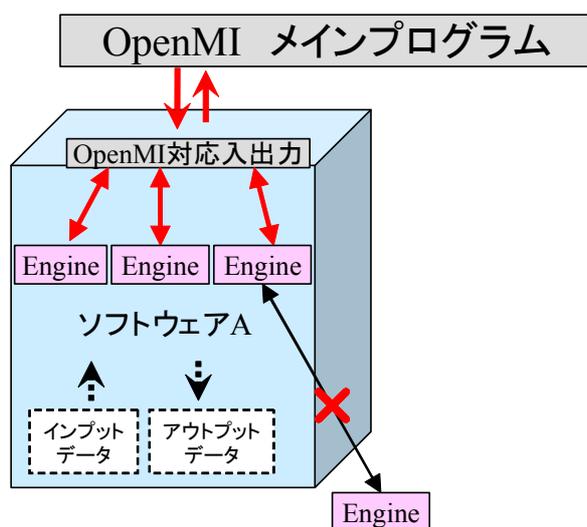


図 3-8 要素の単位

## 3) ラッピングの仕様が含まれている

OpenMI では、ラッピングに関する仕様がマニュアルの中に記述されており、OpenMI の機能としてラッピングを可能にするクラスが実装されている。

## 4) 要素間の反復計算

要素間の反復計算は反復計算コントローラという機能を使うことにより実行することができる。反復計算コントローラとは、反復計算を行う要素間に入り、値の収束などをコントロールする機能である。反復計算の設定などは、OpenMI の GUI を用いて行うことができる。

### 3.1.3 OHyMoS と OpenMI に関する機能の比較

#### (1) アプリケーション実行に関する比較

##### 1) OpenMI

OpenMI Environment をインストールすることにより、OpenMI アプリケーションを実行することができる。また、C#でプログラミングされているため、Microsoft .NET Framework の環境を整える必要がある。Java で実装された OpenMI を利用する場合は Java の環境が必要である。

##### 2) OHyMoS

OHyMoS はソフトウェアではなく、プログラムソースの提供であるため、実行するにはプログラムをコンパイルする必要がある。よって、C++もしくは Java のコンパイラが必要となる。ただし、(新たに要素モデルを加えることはせず) 提供されている実行ファイルを利用するだけなら、コンパイラは必要ない。

#### (2) 要素モデルの利用に関する比較

##### 1) OpenMI

OpenMI Compliant として実装済みのソフトウェアを GUI 上で接続する。そのためには利用するソフトウェアを持っていないなければならない。

##### 2) OHyMoS

プログラムソースに要素モデルの接続関係をコーディングするか、構造定義ファイルを用いて接続関係を定義する。要素モデルはプログラムである。

#### (3) 要素モデルの実装

##### 1) OpenMI

既存のソフトウェアを OpenMI で利用できるようにカスタマイズしなければならない。ソフトウェア内部のソースコードを直接再プログラミングする必要があり、C#もしくは Java を実装する。ただし、元々のエンジン部分のプログラムは書き換えることなく、ラッピングを用いて実装することもできる。実装に必要な技術レベルはソフトウェア開発が可能なプログラマレベルの技術が必要と考えられる。エンドユーザーレベルでは新規に要素モデルを開発することは極めて難しいと考えられる。

##### 2) OHyMoS

要素モデルとなるプログラムソースをユーザーがコーディングする。基本型要素モデルと呼ばれるクラスを継承してプログラミングすればよい。また、既存の要素モデルのソースコードをコピーすることにより、解析の計算式部分、パラメータ・初期状態に関する変数などを書き換えることでも作成できるため、河川技術者のプログラミング技術でも十分作成可能である。他言語で書かれたソースをラッピングするなどそのまま利用することはできないため、C++か Java で実装しなければならない。

#### (4) データの入出力

##### 1) OpenMI

OpenMI では、ソフトウェアそのものが要素モデルとなるので、要素モデルにインプットするデータは、OpenMI を介したデータ入力ではなく、ソフトウェアごとに入力するものと考えられる。

OpenMI の働きはソフトウェア間のデータ通信の仲介者的な役割であり、エンジン部分であるソフトウェアに入力するデータに関しては、OpenMI 側では干渉せず、ソフトウェアごとのデータフォーマットで入力すると予想される。同様に、アウトプットデータはそれぞれのソフトウェアから出力されると考えられ、このフォーマットは各ソフトウェア独自のフォーマットであると予想される。

##### 2) OHyMoS

一方 OHyMoS では、要素モデルにデータを入力する際は、OHyMoS のシステムを介して入力される。要素モデルごとに自由な入力フォーマットではなく、OHyMoS として決められたデータフォーマットで入力データをインプットする必要がある。入力データは OHyMoS の入力端子と呼ばれるオブジェクトの機能で、各要素モデルにデータが入力される。同様に出力データに関しても OHyMoS システムを介してアウトプットされ、データフォーマットは OHyMoS で決められたフォーマットである。また、出力データは出力端子というオブジェクトの機能で出力される。

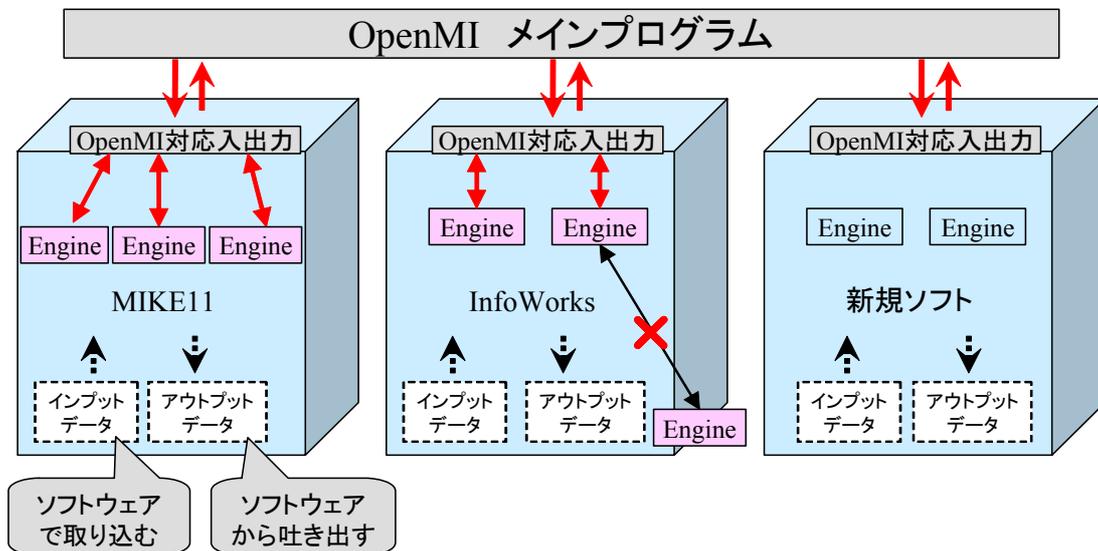


図 3-9 OpenMI のデータ入出カイメージ

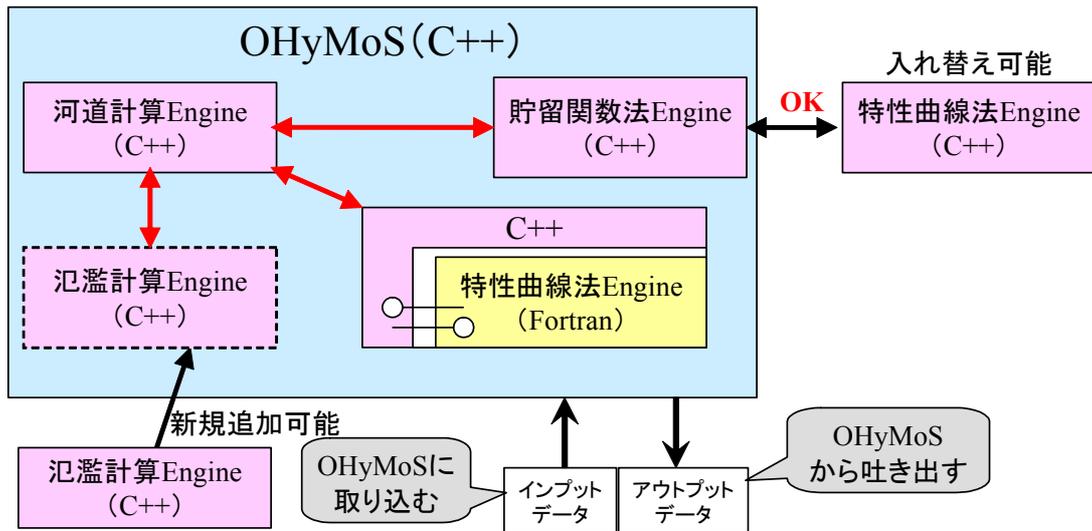


図 3-10 OHyMoS のデータ入出力イメージ

(5) 解析エンジンの単位

1) OpenMI

OpenMI では、ソフトウェアそのものが要素モデルであるので、ユーザーが交換や追加できるエンジンの単位も当然ソフトウェア単位となる。細かい部分でのエンジンの交換や、部分的な要素の追加（例えば、局所流のプログラムを挿入する）などの詳細なモデルの組み替えには向かないと考えられる。

2) OHyMoS

OHyMoS では、プログラムがエンジンの単位となる。また、モデルの組み立て方はユーザーによって自由であり、例えば、図 3-12 の通り、図上方のように河川を地区毎に分断してそれぞれ別のエンジンで計算することもできるし、図下方のように河川は一つのエンジンで解くこともできる。

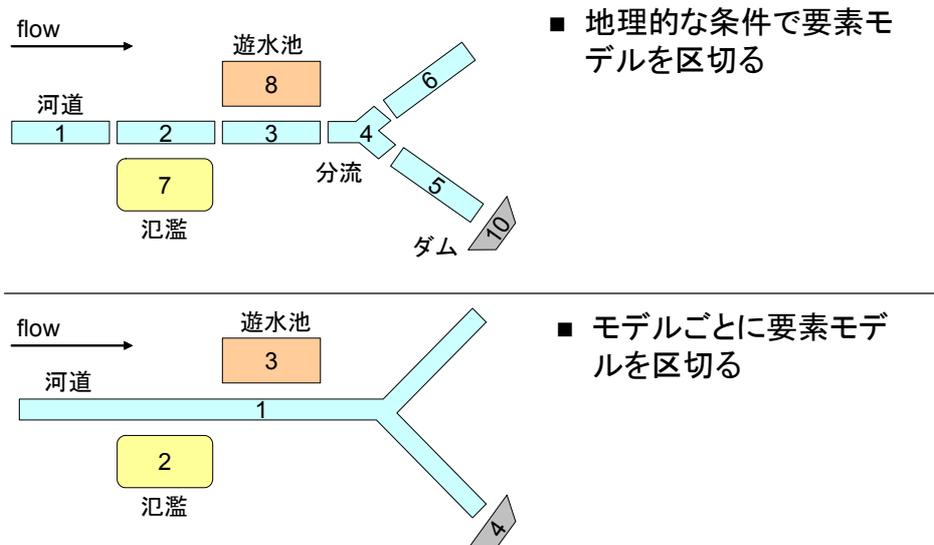


図 3-11 OHyMoS のエンジン単位

## (6) 計算の制御

### 1) OpenMI

計算順序などの計算の制御は **OpenMI** で設定することができる。ただし、エンジンの単位より細かいレベルでの制御はすることができない。タイムステップごとに同時平行で計算を進めることができる。

### 2) OHyMoS

OHyMoS では計算の順序そのもの決めることはできない。基本的に計算できる要素モデルから計算を順次していく。タイムステップごとに同時並行的に計算を進めるためには、その秒数をユーザーが設定する。

## (7) ラッピング

### 1) OpenMI

OpenMI では、前述の通りラッピングの機能がフレームワークの一部として実装されている。具体的には、ラッピングするために必要な機能が実装されたクラスが含まれている。ラッピングの機能はいくつかのクラスで構成されている。

### 2) OHyMoS

OHyMoS では他言語を取り込むラッピングの機能はない。C++もしくはJava で要素モデルを実装しなければならない。

## (8) 反復計算

### 1) OpenMI

反復計算コントローラを用いることにより、GUI 上で反復計算の設定を行うことができる。

### 2) OHyMoS

反復計算を行うことはできるが、要素モデルは独立しているという OHyMoS の基本コンセプトからは若干はずれたプログラミングが必要になる。どの要素モデルでも反復計算ができるというわけではなく、予め反復計算を行うことを前提とし、どの要素モデルと反復計算を行うかも記述したプログラミングが必要となる。収束計算の条件などもプログラミングする必要がある。ただし、接続自体は構造定義ファイルを用いて行うことができる。

### 3.1.4 OHyMoS と欧米のシミュレーションモデルとの比較

図 3-12 および図 3-13 に、OHyMoS と欧米開発のフレームワーク、ソフトウェア、データモデルの関係・位置づけをシミュレーションモデル、データ交換・連携インターフェース、データの構造化という枠組みで整理したイメージを示す。

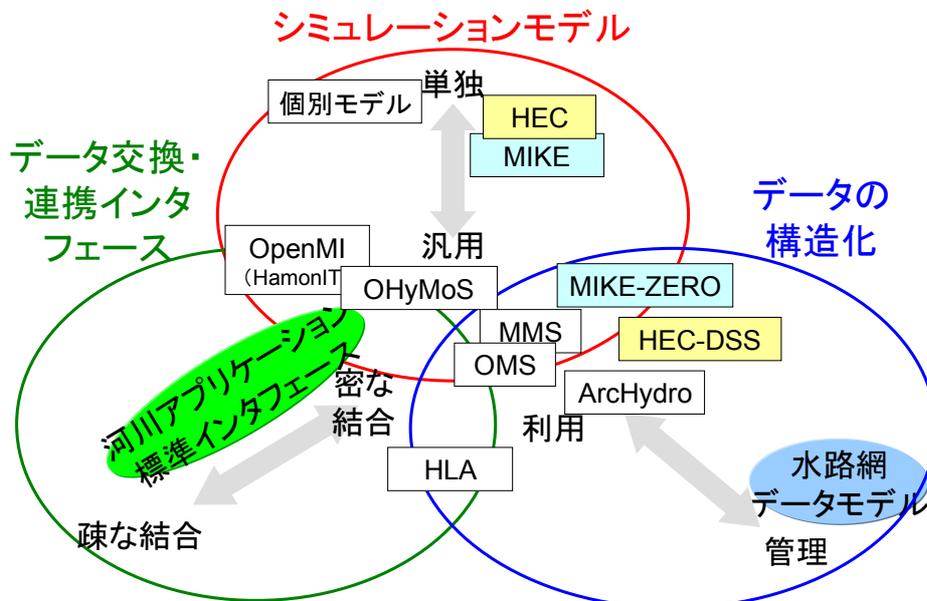


図 3-12 解析モデル、フレームワークの関係

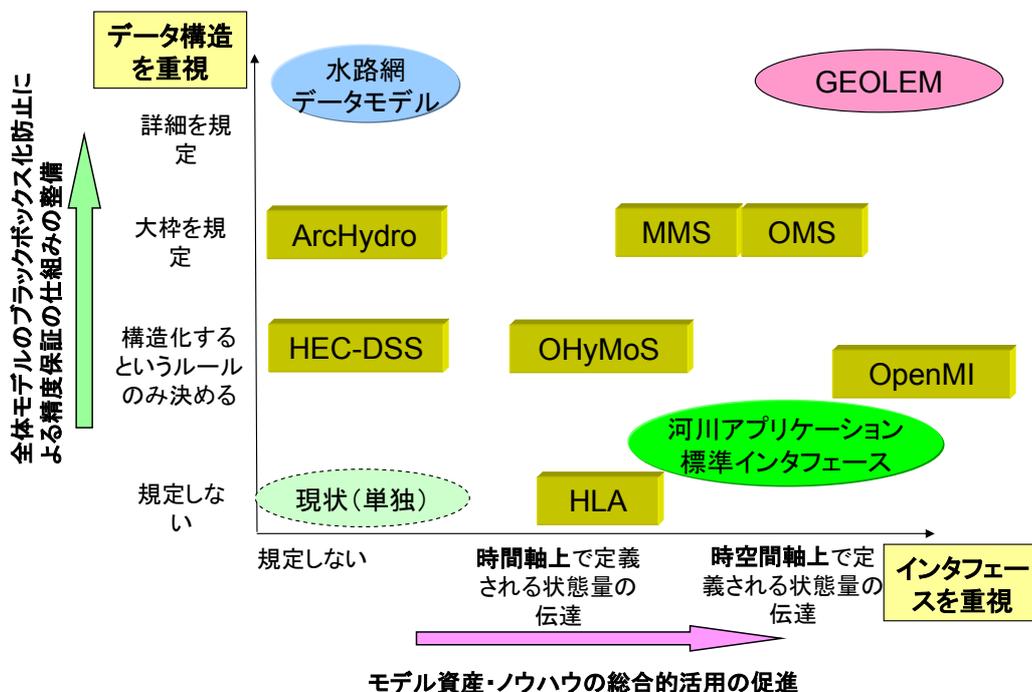


図 3-13 解析モデル、フレームワークの位置づけ

## 3.2 米国における水理・水文・水質シミュレーションモデルの開発体制等に関する現地調査

### 3.2.1 調査の背景と目的

河川局では「水情報国土」に関わる活動として、平成14年（2002）よりデータの提供、データ構造化の推進、インターフェース標準化の推進、ソフト標準化の推進が行ってきた。さらに、それと平行して、関連する海外動向調査が文献調査を中心として行ってきた。

一方で、水文・水資源学会が中心となり、国総研も積極的に参画して、2003年にはシンポジウム「水のモデリングとインターフェイス」、2004年、2005年の2カ年にわたって「流域水物質循環モデル・ソフトウェア博覧会」を開催してきた（1章参照）。さらに、米国陸軍工兵隊水文工学センターで長年にわたってソフトウェア開発に関わってきたアーレン・フェルドマン氏の講演会が2005年3月に実施された（4章参照）。

本調査は、これらの既往の調査とそこで得られた知見を背景とし、河川および流域の水文、水理機構の解明に関する共通利用ソフトとデータに関して、その開発と運営、ならびに利用に関する有効性がどのように保たれているかを米国の先進事例から学ぶことを目的として、実際に米国の関連諸機関を訪問することにより、ソフトウェア開発者・使用者といった種々の立場からみたデータ、ソフトウェアのあり方に関する情報収集と意見交換を行うものである。

### 3.2.2 調査行程

本調査は、平成17年11月13日（日）～19日（土）にわたって実施され、訪問した機関は下記の7機関である。

1)	米国工兵隊 水文工学センター本部	Hydrologic Engineering Center (HEC), USACE
2)	米国工兵隊 サクラメント現地事務所	Sacramento District Office, USACE
3)	統合管理センター	The Joint Operations Center
4)	カルフォルニア州 水資源局	California Department of Water Resources
5)	米国地質調査所 カルフォルニア水科学センター	California Water Science Center, U.S. Geological Survey (USGS)
6)	米国地質調査所 デンバー	U.S. Geological Survey in Denver
7)	米国農務省 農業研究部	Agricultural Research Service, U.S. Department of Agriculture (USDA)

今回訪問した機関は、1)～5)はカルフォルニア州サクラメント周辺、6)、7)についてはコロラド州デンバーおよびフォートコリンズに位置している。

### 3.2.3 訪問機関ごとの調査の着眼点

今回の調査の着眼点は大きく次の3つのグループに分類される。

- HEC（工兵隊水文工学センター）ソフトウェアの実態
  - HEC ソフトウェアは、よく使われている理由。その戦略と実態を知りたい。
  - HEC ソフトウェアの開発と、その運営方法。
  - HEC ソフトウェアの設計思想と、今後の展望。
  - CWMS（工兵隊水管理システム）によるデータの収集、管理、利用の実態と、運営上の課題
  - サクラメント現地事務所、統合管理センター、カルフォルニア州水資源局におけるソフト活用の実態
- USGS（地質調査所）カルフォルニア水科学センターにおけるデータ管理の実態
  - データ収集、集約、公開の実態
  - リアルタイムデータ収集の体制、異常値への対応など
- USGS（地質調査所）、USDA（農務省）における流域水循環解析モデルフレームワーク開発の実態
  - フレームワーク開発の実態（開発の進捗、開発体制、・・・）
  - フレームワーク上で動作する要素モデルの開発の状況
  - 実務への活用の確認（研究開発にとどまっているのか、実務にまで応用されているのか？）

### 3.2.4 調査概要

#### (1) 米国工兵隊 水文工学センター本部 (Hydrologic Engineering Center, USACE)

米国工兵隊 水文工学センターの調査結果の要点を下記に示す。

- HEC ソフトウェアは全米で広く使われてきた結果、水計算ソフトウェアのデファクトスタンダードとしての位置を獲得している。その理由は、
  - 計算スキーム・基礎式が確立されたもの。
  - 現場事務所の要望を一般化された形で取り込んでいる。
  - 工兵隊内部からのフィードバック・維持管理の体制が整っている。
  - 息の長いリーダーが存在している。職員も長期間勤務してきている。
- HEC ソフトウェアは“Public Domain”。
- 35年の歴史とユーザーサポートの充実が広く使われている理由。
- 学会等である程度認知されたモデルを導入。制度保証の仕組みがあるわけではない。新モデルの導入にあたっては、従来モデルとの結果比較により検証は実施している。
- HEC の責任はソフトウェアの供給であり、それをどのように使うか（データ入力、モデル選択、パラメータ設定）はユーザーの責任。ただし、モデル選択、パラメータ設定等の参考となるマニュアルは発行（Web サイト）している。
- HEC 単体ソフトウェアの組み合わせ利用は、CWMS（工兵隊水管理システム＝工兵隊独自のインターフェースにより、ソフトを相互接続する仕組み）を活用。CWMS の活用により、

リアルタイム予測等の活用が可能となる。外部データの取り込みや HEC ソフトウェア間の接続は HEC-DSS を介して行う。

- 現在のところ、海外ソフトとの連携は視野には入れていない模様。
- HEC へは、韓国から研修・トレーニングに参加している模様。

### 1) HEC サービスの 3 本柱

HEC における活動の基本としては、技術支援(Technical Assistance)、研究開発(Research and Development)、研修およびサポート(Training and Software Support)となっている。研究開発は、主に HEC ソフトウェアを対象としている。技術支援、研修およびサポートは、HEC ソフトウェアに関し、工兵隊各現地事務所に対して実施している (図 3-14 参照)。

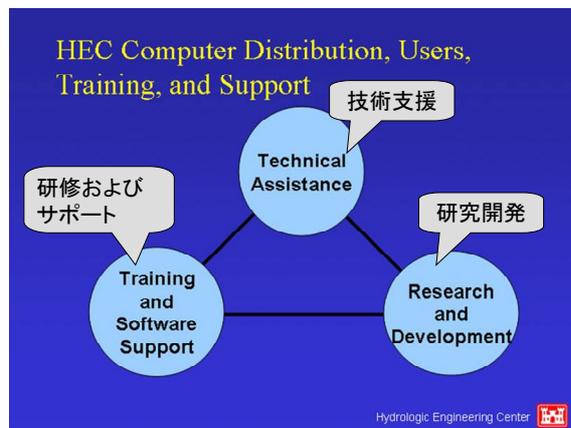


図 3-14 HEC サービスの 3 本柱

### 2) HEC のソフトウェア

米国工兵隊水文工学センター(HEC)のソフトウェアは、原則として無償で公開しており、Website からダウンロードできる主なソフトウェアは下記の通りである。なお、他にも専門家に使用されているソフトは、90 以上にも上る。

表 3-2 : HEC の主なソフトウェア

名 称	概 要
HEC-RAS	1 次元水理解析モデル
HEC-GeoRAS	HEC-RAS の GIS コンポーネント
HEC-HMS	降雨・流出解析 (水文) モデル
HEC-GeoHMS	HEC-HMS の GIS コンポーネント
HEC-ResSim	貯水システム解析モデル
HEC-FDA	洪水被害解析モデル
HEC-DSS	データベースシステム

例えば、近年開発されてきた水文ソフトウェアパッケージ HEC-HMS の計算モジュール構成は下図に示すとおりである。

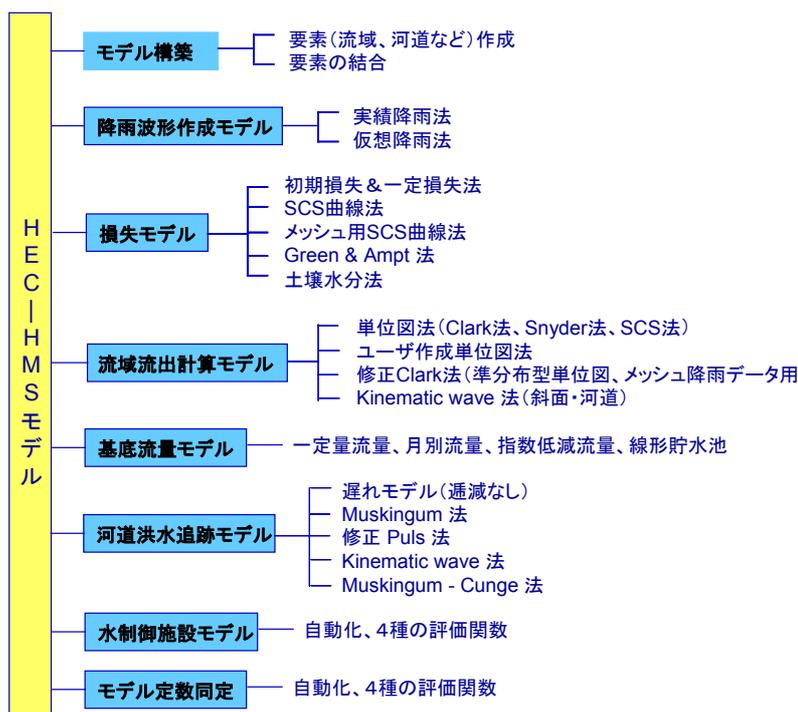


図 3-15 : HEC-HMS のモジュール構成

### 3) HEC ソフトを活用している現地事務所

工兵隊における現地事務所 (Regional Office) の総数は 42 に上る。特に、各事務所で HEC ソフトの使用を義務付けているわけではないが、そのうちほとんどすべての事務所で HEC ソフトが活用されている (図 3-16 参照)。

Voluntary Corps office Maintenance Subscriptions					
RAS UNET GeoRAS	HMS HEC-1 GeoHMS	FDA Risk FIA	FFA STAT	ResSim HEC5 HEC5Q	HEC-DSS DSSVue
河道・水理	水文	洪水被害	洪水頻度	貯水池	データベース
41	40	33	35	30	39

図 3-16 HEC ソフトウェアを活用している事務所の数

### 4) HEC ソフトの活用状況

HEC ソフトウェアは、工兵隊内部のみならず米国内、海外においても幅広く認知され、活用されている。図 3-17 に、過去 1 年間に HEC サイトを閲覧した機関、国を示す。この図から米国内からのアクセスが比較的多いことが理解される。国としては、イタリアからのアクセスの多さが特徴的である。なお、日本からのアクセスは、30,000 前後となっている。

ソフトウェア別のダウンロード状況は、HEC-RAS (約 6,500 ダウンロード/月)、HEC-HMS (約

3,100 ダウンロード/月) となっている。

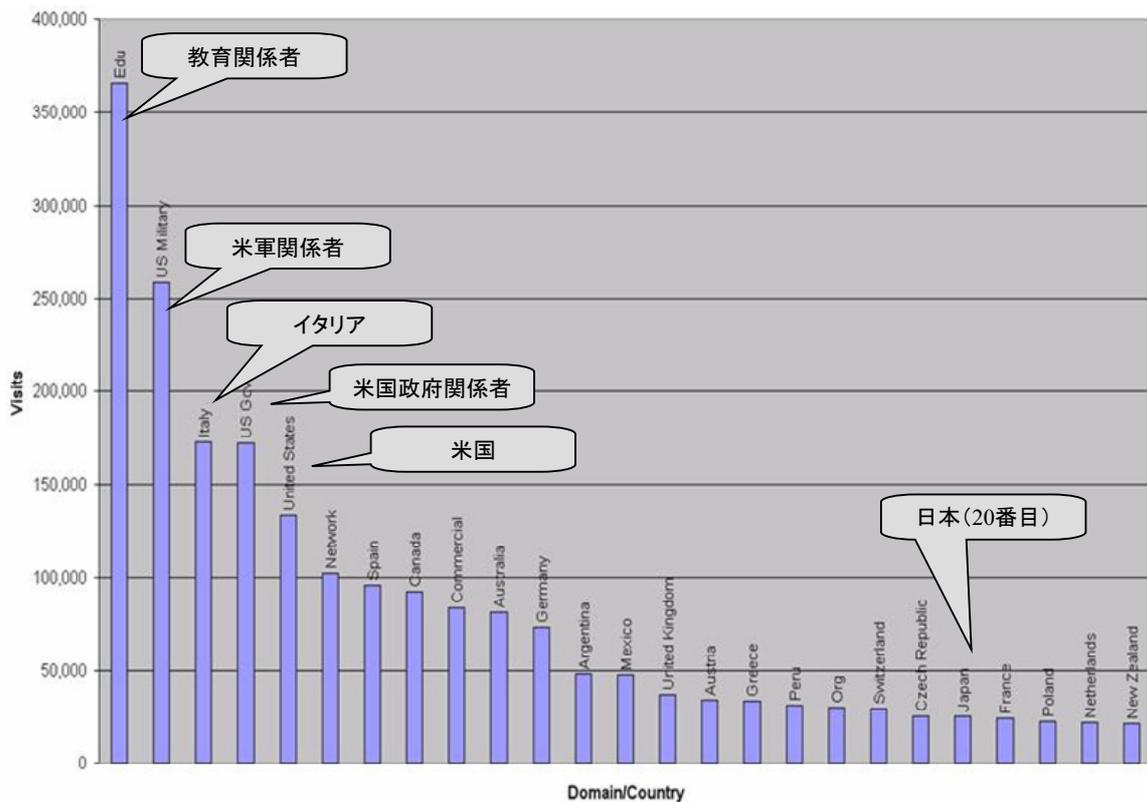


図 3-17 HEC の Web サイトを閲覧した機関および国 (2004 年)

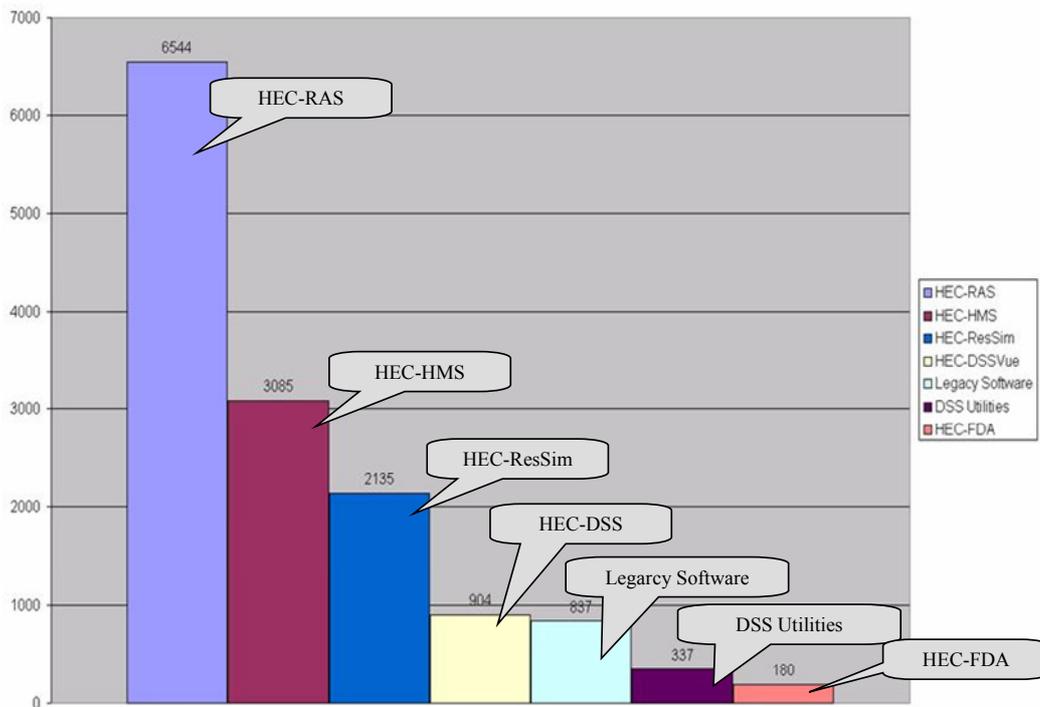


図 3-18 HEC ソフトウェアヶ月当たりのダウンロード数

### 5) HEC ソフトウェアに関わる関係機関 (図 3-19 参照)

HEC では、開発したソフトウェアへのサポートは、工兵隊内部と工兵隊外部に対しては方針が区別されている。工兵隊内部に対しては、ソフトウェアの提供、メンテナンス、各種サポートがなされている。工兵隊ではない一般ユーザーに対しては、HEC の Web サイトからソフトウェア、マニュアルがダウンロードできる以外は、サポートは提供されていない。ただし、ソフトウェアのバグに起因する障害と思われる質問に対しては、対応をしている。また、一般ユーザーに対するサポートに関しては、HEC ソフトウェアのベンダーが実施している。

HEC ソフトウェアの開発に関しては、HEC が中心的な役割を果たしているが、適宜民間企業への外注も行っている模様である。また、大学・研究機関と連携して開発を実施している。

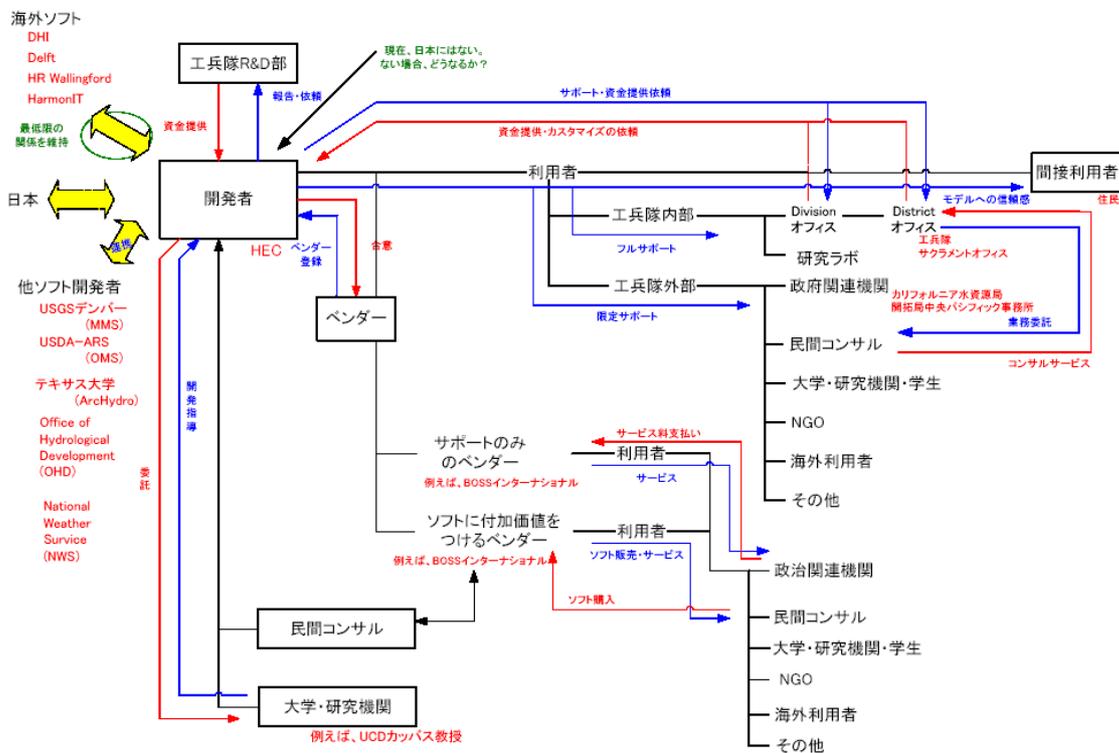


図 3-19 HEC ソフトウェア関係機関の関係図

### 6) HEC-DSS

DSS は、Data Storage System の略称であり、HEC ソフトウェアの入出力データを管理（データ共有、データ操作、視覚化機能等）するための機能がある。HEC ソフトウェア間の入出力データの共有の一例として、HEC-HMS（水文ソフト）と HEC-RAS（1次元水理ソフト）の入出力データの例を紹介する（図 3-20 参照）。

1. 先ず、HEC-HMS のアウトプット（流出量）データを先ず、HEC-DSS に取り込む。
2. 上記、1. の HEC-DSS のデータは、HEC-RAS にインプットデータとして、取り込みをする。
3. HEC-RAS のアウトプットデータ（水位）は、HEC-DSS に取り込みが出来るようになっている。

このようにデータベースを介した形で、複数のソフトウェア間の入出力データの互換性を図っている。このシステムでは、時系列データだけではなく、河道断面などのデータも保存できるようになっている。ただし、OHyMoSのように、異なるソフトウェア間のカップリング（同期をとりながら反復計算）する機能は無い。なお、HEC-DSSには、図 3-22、図 3-23 に示すようにデータの視覚化やデータ同士の加減乗除や複雑な計算をサポートする機能も付加されている。

- HECソフト間のデータ互換性はHEC-DSSを介する。
- DSSは、Data Storage System(データ保管システム)の略
- 例えば、HEC-HMS(水文ソフト)とHEC-RAS(1次元水理ソフト)間のデータ互換性のイメージ

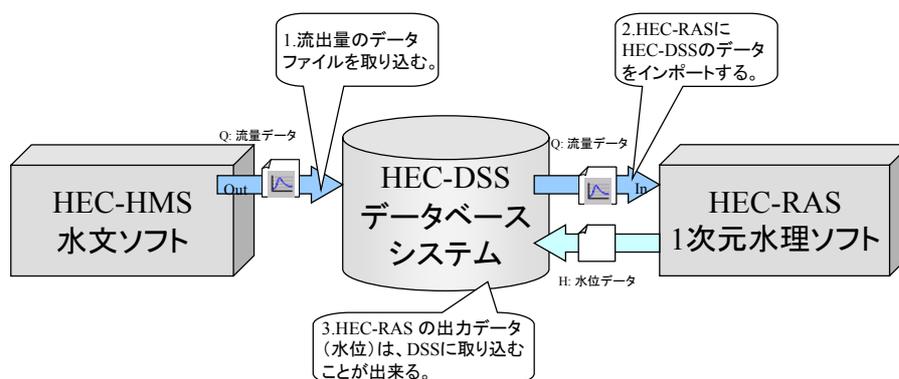


図 3-20 HEC-DSS を介したソフトウェア間のデータ互換

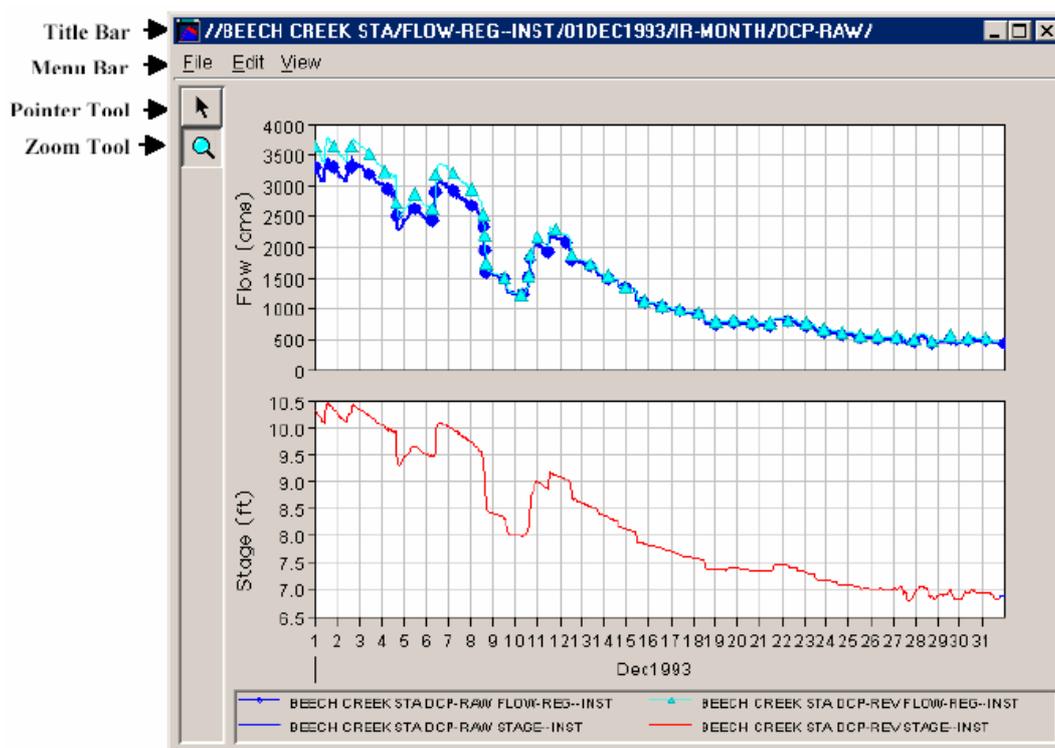


図 3-21 : HEC-DSS のデータ視覚化ツール

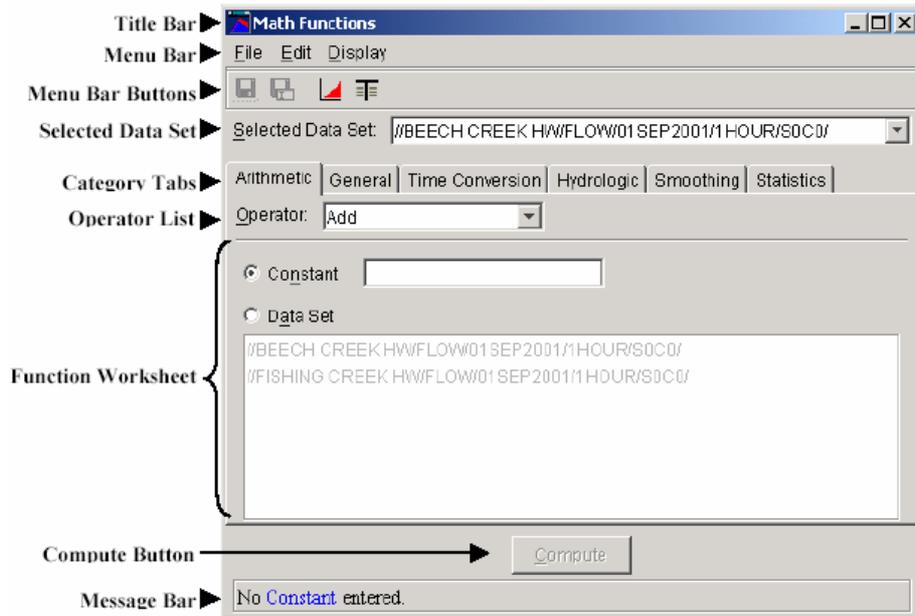


図 3-22 : HEC-DSS の算術計算ツール

### 7) CWMS (工兵隊水管理システム)

HEC では、HEC-RAS、HEC-HMS、HEC-ResSim、HEC-FDA、HEC-DSS の他にも工兵隊の現地事務所の管轄の水管理（統合水資源管理、リアルタイムデータ監視）のため、CWMS (Corps Water Management System: 工兵隊水管理システム) を開発してきている。

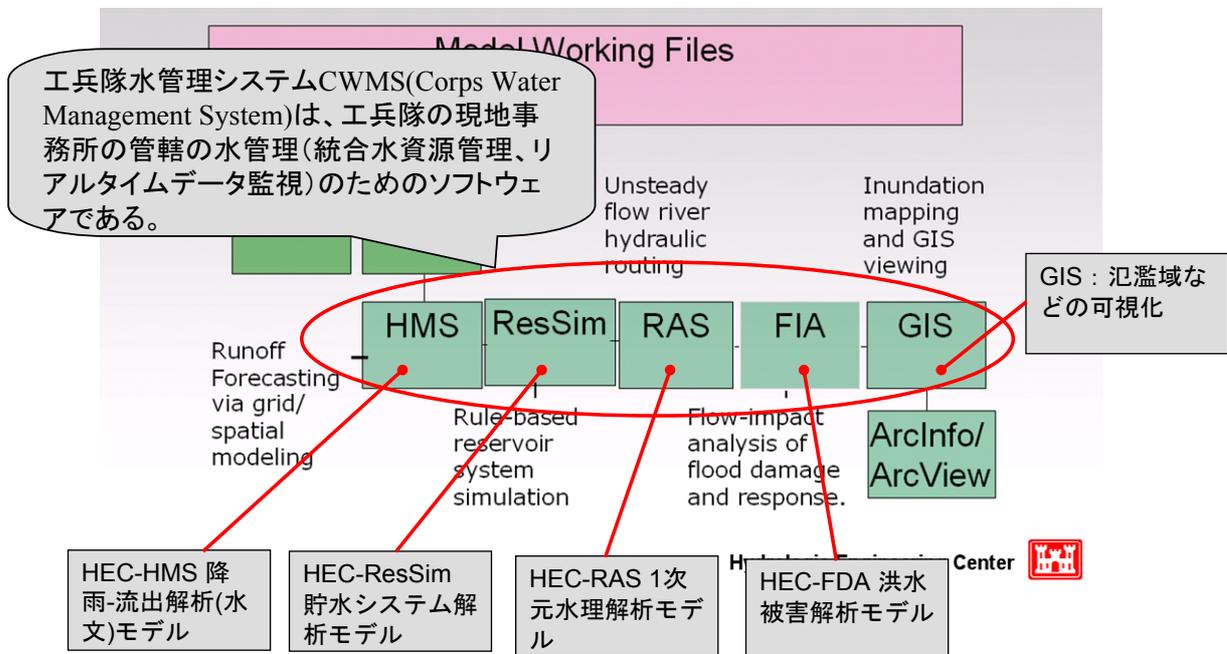


図 3-23 CWMS の概要図

## (2) 米国工兵隊 サクラメント現地事務所(Sacramento District Office, USACE)

米国工兵隊 サクラメント現地事務所の調査結果の要点を下記に示す。

- 一般的な工兵隊の水管理と NWS（気象局）との業務区分
  - 洪水のデータを提供するまでが、NWS の責務となっている。洪水への対処をどうするかまでは、責務ではない。
  - このうち、工兵隊ダムの実管理とダムに関わる洪水警報は、工兵隊が実施する。
  - 一般的には、工兵隊のデータを NWS が利用する仕組みにはなっていない。
- 例外的にユバ川プロジェクトでは、工兵隊と気象局の連携が実現：CWMS（工兵隊水管理システム）を用いた HEC ソフトウェアの活用がなされている。
- HEC ソフトウェアそのものの活用方法
  - 水理、水文解析関連業務は、工兵隊職員が自らプロジェクトをマネジメントし、その中で HEC ソフトウェアを動かしているなど、基本的に全部直営で処理する。
  - 外注はモデル構築業務程度。
- 工兵隊での改善提案（HEC へのフィードバック）は、直営で実施することによる情報把握によるもの

以下、サクラメント現地事務所における HEC ソフトウェア活用の事例について紹介する。

### 1) ユバ川プロジェクト

サクラメント川支流であるユバ川流域には、ニューブラーズバーダムおよびオロビルダムがあり、これらのダムの下流域における治水計画の改善を目的としたユバ川プロジェクトが、カルフォルニア州水資源局、ユバ郡の水局、国立気象局、工兵隊の連携体制のもとに実施されている。

**サクラメント事務所における HEC ソフトウェア活用事例  
～サクラメント川支流のユバ川プロジェクト (1)～**

Forecasted - Coordinated Operation  
Feather - Yuba Joint Project (\$13M)

US Army Corps of Engineers  
Sacramento District  
Forecasted - Coordinated Operation

Yuba County Water Agency  
CA DWR - Flood Management  
CA DWR - State Water Project  
CNRFC - National Weather Service  
OHD - National Weather Service  
SPK - Corps of Engineers  
HEC - Corps of Engineers

サクラメント川支流のユバ川プロジェクトには、**加州水資源局、ユバ郡の水局、国立気象局、工兵隊が参加している。**  
この流域にある**2つのダム(ニューブラーズバーダム、オロビルダム)**の下流域における**治水計画の改善が目的**である。

図 3-24 ユバ川プロジェクト

豪雨の際、貯水池の水を的確に放流するために、HEC ソフトウェア (HEC-ResSim) を活用している。貯水池に流入する流量、水位は気象局 (NWS) の River Forecast Center (河川予測センター) によって、リアルタイムにされている。そのデータを HEC-ResSim に取り込み、そのデータをダムへのゲート操作の検討に活かすこととしている (図 3-25 参照)。

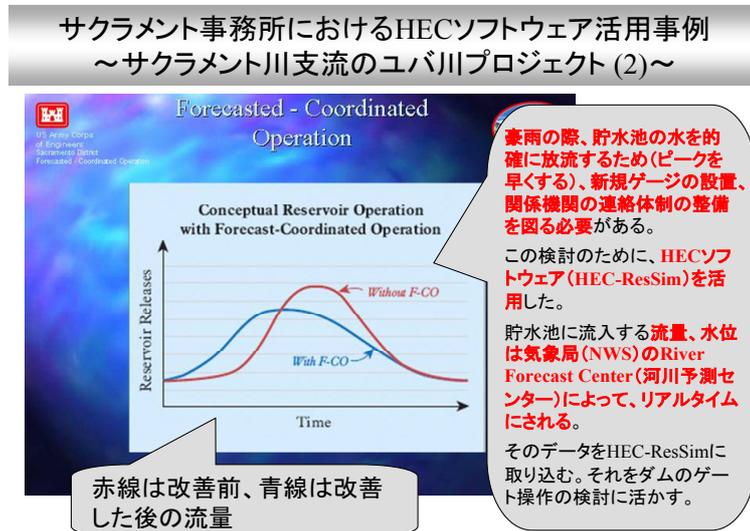


図 3-25 ユバ川プロジェクトの HEC ソフトウェア活用

## 2) トラッキー草地洪水制御プロジェクト

ネバダ州 Pyramid Lake からリノ市に至るエリアにおける洪水被害軽減と河川生態系の回復を主目的にトラッキー草地洪水制御プロジェクトが実施されている。このプロジェクトでは、図 3-29 に示すようにモデル対象地域を 4 区間に分割し、それぞれの区間の目的に応じて HEC ソフトウェアの適用 (組み合わせ) をしている。

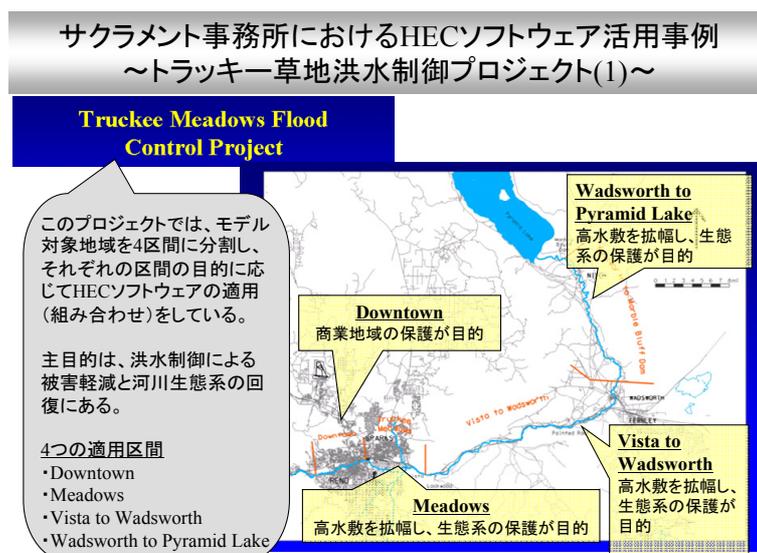


図 3-26 トラッキー草地洪水制御プロジェクトの HEC ソフトウェア活用(1)

例えば、Downtown を対象にしたエリアに対しては、HEC-RAS（河道モデル）、FLO-2D（2次元氾濫モデル）、HEC-FDA（氾濫被害解析）、HEC-DSS（データベース）の組み合わせのモデルを活用している。

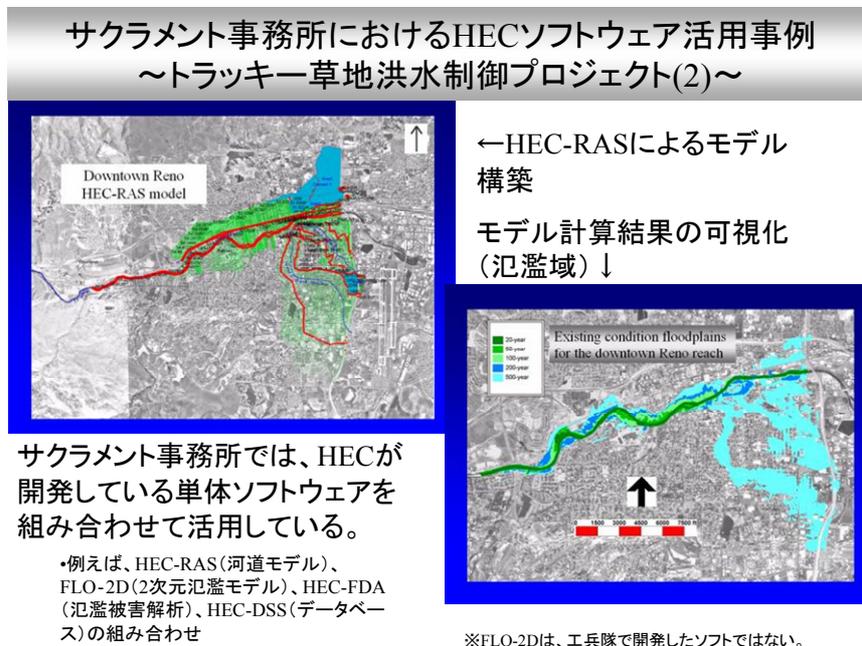


図 3-27 トラッキー草地洪水制御プロジェクトの HEC ソフトウェア活用(2)

### (3) 統合管理センター (The Joint Operations Center)

統合管理センターの調査結果の要点を下記に示す。

- 統合管理センターは、カルフォルニア州内の水管理の関係機関（米国開拓局、米国海洋大気庁等）から構成されている組織である。
- 統合管理センターは、組織的に統合して水管理を実施する組織とはなっていない。技術・研究的には様々なノウハウを共有できる場にはなっている。
  - Community Hydrologic Prediction System（コミュニティ水文予測システム）という関連機関からの情報を統合した水文予報プロジェクトが進行中である。
- 米国開拓局 (BOR : Bureau of Reclamation) では、独自のソフトウェアを使っている。HECソフトウェアの活用はほとんどない。
- 米国海洋大気庁 (NOAA : National Oceanic & Atmospheric Administration) の気象局 (NWS : カルフォルニア州&ネバダ州) では、気象予報とともに、独自の流出解析モデルを用いて水文予報（流量、水位）を行っている。水文予報には、短期（決定論的予報）、中期（アンサンブル平均予報）、長期（統計的予報）がある。
- HEC ソフトウェアは、異なる水文システムで万能に使えるというわけではない（長年、開発に従事してきた OB の感想）。

## 1) カルフォルニアデータ交換センター (California Data Exchange Center :CDEC)

統合管理センター内には、カルフォルニアデータ交換センター (CDEC) が運営されている。これにより、カルフォルニア州内の水文 (河川&貯水池)・気象のリアルタイムデータの共有が可能となっている。CDEC の Web サイトによって、これらのデータが公開されている。なお、サクラメント川に関するデータもほとんどこのサイトから閲覧することが可能である。協力機関は、工兵隊、気象局、開拓局、地質調査所、カルフォルニア州水資源局等から構成される。砂漠研究所、米海軍、カルフォルニア州交通局、リノ市水管理事務所等には、CDEC から直接リアルタイムデータを送信している。



図 3-28 カルフォルニアデータ交換センターの Web サイト

## 2) コミュニティ水文予測システム (Community Hydrologic Prediction System)

統合管理センターにおいては、コミュニティ水文予測システムという関連機関 (連邦政府機関、州政府、地方、大学、民間等) からの情報を統合した水文予測プロジェクトが進行中である。

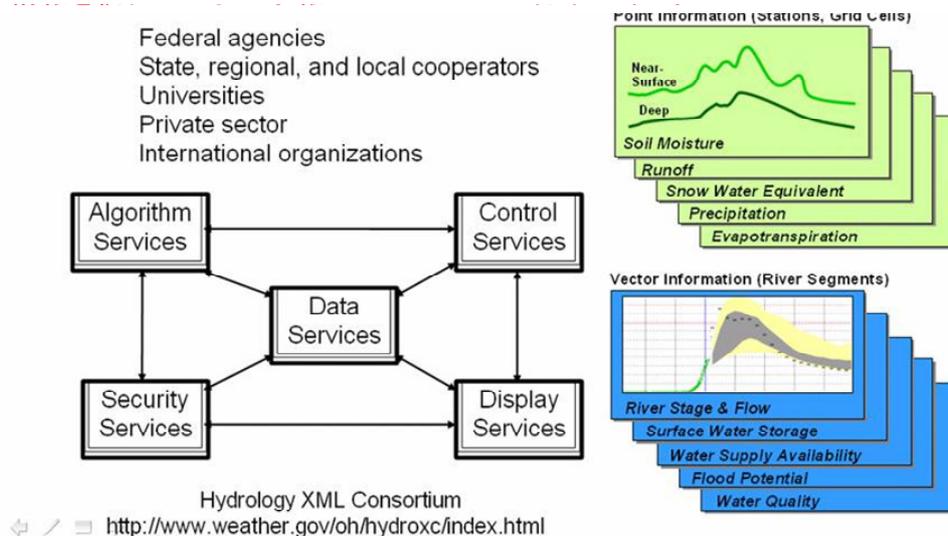


図 3-29 コミュニティ水文予測システム

#### (4) カルフォルニア州 水資源局(California Department of Water Resources)

カルフォルニア州 水資源局の調査結果の要点を下記に示す。

- カルフォルニア州水資源局では、独自の水管理ソフトウェアを開発して、それを活用している。これらのモデルは、カルフォルニア州の必要性に応じて作りこんだ独自のソフトウェアである。(カルフォルニア州としては、HEC ソフトウェアは利用していない。)
  - 州全体の水収支モデル (CALSIM)
    - ◇ カルフォルニア州は、北部の水資源を南部へアクアダクト (用水路) を通じて、水供給している。
  - デルタシミュレーションモデル(DSM2)
    - ◇ 複数のモデルから構成されている。
      - 1次元河川ネットワークモデル(HYDRO)：サクラメントデルタ地区
      - 3次元汽水域モデル(PTM)：魚の動態モデル、温度、食物連鎖もカバー
      - 1次元水質モデル (QUAL)：リン、塩分、塩素、酸素等
    - 地下水・表流水モデル (IGSM2)
- 豊富な資金、人材によるソフト開発を実施している。
  - 水資源局のスタッフは、2500名のうち、技術者は300名。
  - モデル構築の部局 (Modeling Support Branch) には約50名が在籍する。
    - ※ カルフォルニア州は、独立国とすれば世界第5位のGDPに相当する
- データユニットという部署を作り、10年間かけて独自にデータベースを構築し、管理している。データ構造は、特に系統立てて構築しているわけではなく、常に変化している。

##### 1) カルフォルニアシミュレーションモデル (CALSIM: California Simulation Model)

カルフォルニア州水資源局と開拓局により、CALSIM と呼ばれるカルフォルニア州全体の水収支モデルを独自に開発し、州内の水管理 (貯水池の新設検討、貯水容量の増強の検討等) に活用してきている。

##### 2) デルタシミュレーションモデル (DSM2)

DSM2 (Delta Simulation Model2) は、HYDRO (1次元河道モデル：流量、水位、流速、水深)、QUAL (1次元水質モデル：リン、塩分、塩素、酸素等)、PTM (準3次元浮遊性粒子移動モデル：魚類の動態、水温、食物連鎖までカバー) から成り立っており、サクラメント川からサンフランシスコ湾への流入、サンフランシスコ湾の潮汐、ポンプ配水によるカルフォルニア州南部への配水に起因する魚類生態への影響、灌漑用水供給の水質保全等を目的として、適用されてきている (図3-31参照)。

カリフォルニア州全体の水収支モデル (CALSIM) のネットワーク図

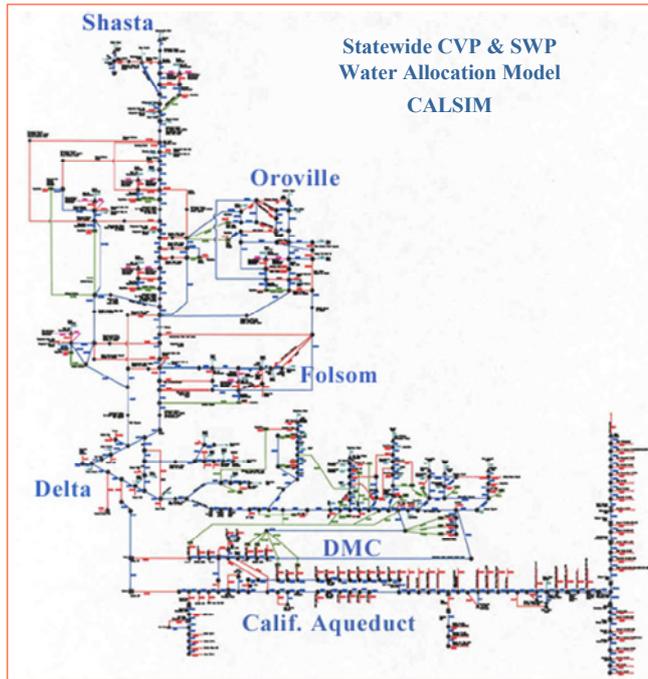


図 3-30 CALSIM ネットワーク図

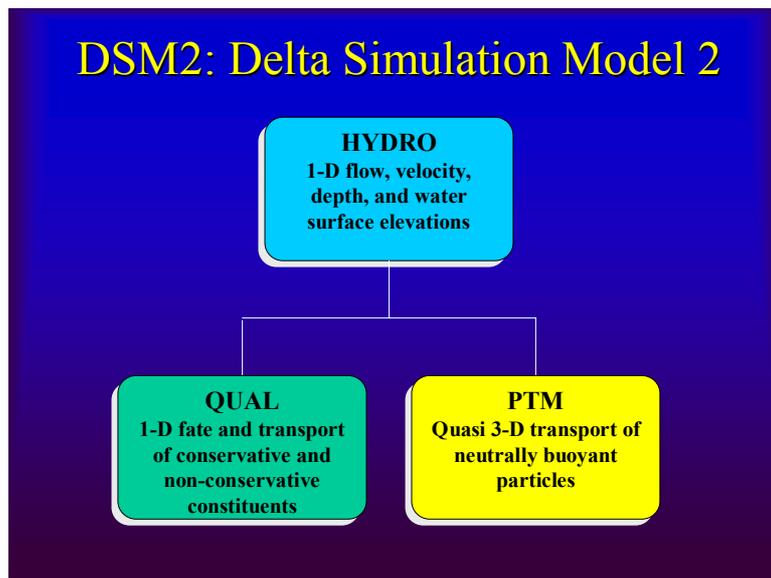


図 3-31 DSM2 の構成

3) 地下水および表流水統合モデル (IGSM2: Integrated Groundwater and Surface water Model2)

地下水および表流水統合モデル (IGSM2) は、準3次元地下水モデルおよび1次元表流水モデルの統合モデルである。基本は、Finite Element モデルである。IGSM2 は、地下水、表流水&河口域のモデリングが可能となっている。このモデルの構築には、マサチューセッツ工科大学、カリフォルニア大学バークレー校およびコーネル大学出身の職員が関わっている。なお、カリフォルニア

州全体の水需要のうち、30%を地下水に頼っている。

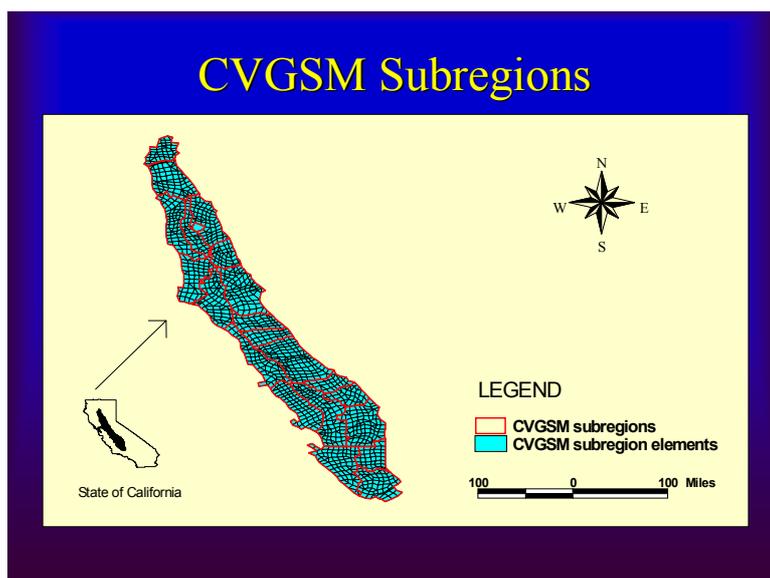


図 3-32 IGSM のイメージ図

(5) 米国地質調査所 カルフォルニア水科学センター(California Water Science Center, U.S. Geological Survey)

米国地質調査所 カルフォルニア水科学センターの調査結果の要点を下記に示す。

- USGS は、NWIS (National Water Information System) と呼ばれる水情報システム、収集貯蔵、アーカイブを保有し、データを収集、公開している。
  - USGS では、自前の観測網から得られる水位、水質などを蓄積データ、リアルタイムデータとして、全米に公開、提供している。
  - USGS のデータを工兵隊やカルフォルニア州水資源局が使用することはあっても、USGS が外部からのデータを入手してそれを使用・公開する仕組みは持っていない。
  - カルフォルニア水科学センターは、州内の NWIS を取り扱っている。各州にも同じような部署がある。
  - 各州の NWIS システムのソフトウェアは、(ヘッドクォーター) USGS 本部から配布される。
  - 州毎に、NWIS システムの中の構造、組織、資金元は異なる。
    - ◇ 以前は、集中管理型の一つのシステムがあったが、地方のサイバー同士の通信に問題があった。
- USGS は、公開している各データに対しては、独自のチェック体制を敷いている。
  - 例えば、水位データは、リアルタイムのデータも公開しているが、履歴データとして公開する際には、スクリーニングをかけ、チェックしてから公開している。
  - USGS が EPA (環境保護局) にデータを提供している。
  - EPA は、SurfYourWatershed で、ポータルサイトの役割を果たしている。が、EPA はデータの妥当性をチェックしていない (データの責任を負っていない)。

■ 予算

- USGS 全体として、活動のカテゴリは下記の 4 つから構成される。
  - ◇ 地質: Geology
  - ◇ マップ: Geography
  - ◇ 生物: Biology
  - ◇ 水: Water
- それぞれのカテゴリによって、資金の入り方が違う。
- 水のカテゴリに関しては、政府機構（地方 (District)、州、連邦など）から拠出されている。
- さらに、ナショナルプログラムの資金枠もあり、その一つにデータマネジメントがある。

NWIS の全体概要を図 3-34 に示す。サイトのデータは、現場まで観測者が出向いて観測を実施する方法、観測所に設置されているパソコン等から自動転送する方法などがある。ADAPS と呼ばれる表流水の時系列データを 15 分間隔でリアルタイムデータを収集している。データを転送する方法には、衛星通信、通常の電話回線、携帯電話回線などがある。州内に 700 の観測所があり、うち 440 箇所は自動化されている。

収集されたデータは、NWIS data server あるいは、NWIS applications servers に転送され、USGS カルフォルニア水科学センターでデータのチェックを実施している。データチェックの後、Web Server に転送し、一般に公開する流れになっている。

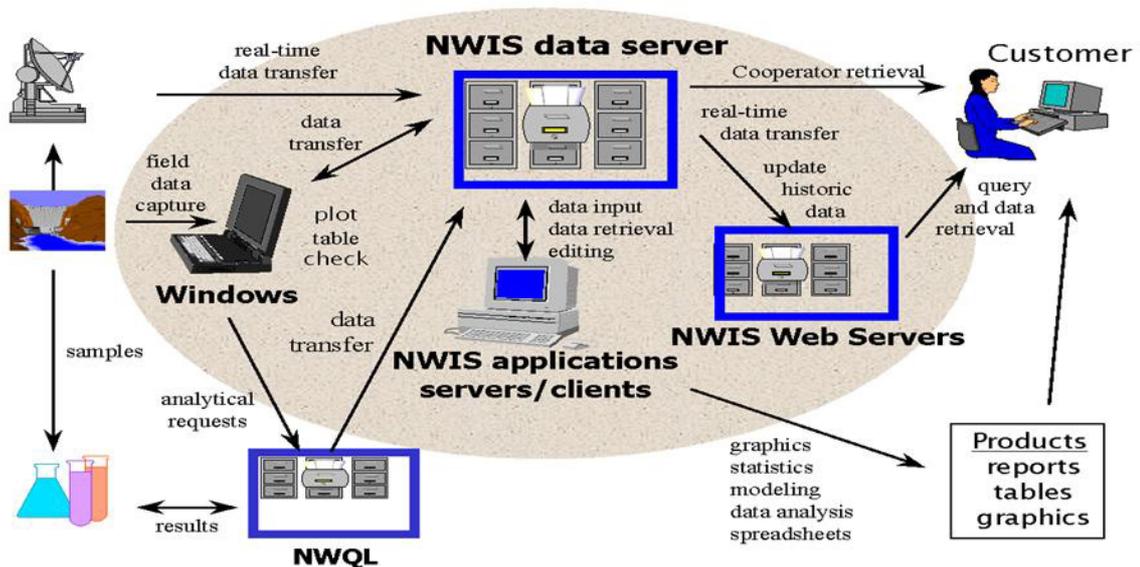


図 3-33 米国地質調査所の水情報システム

上記の NWIS サーバーは、各州に一台ずつ配備されており、それぞれの州の USGS 水科学センターが責任を持って管理している。図 3-34 に示すように、それぞれの州のサーバーが青線で繋がっているが、実はそれぞれのサーバー同士の相互接続はできていない。すなわち、自動的に、他の州のサーバーからデータを自動的に入手することはできない状況である。目下、一般開放している Web サイトの情報しか入手できない状況である。NWIS システム自体は、ソフトウェアが HQ (へ

ッドクォーター) から配布されるが、州ごとに中の構造も異なっている。この NWIS システム以前は、セントラルシステムが整備されていたが、地方のサーバー同士の通信にも、一旦中央を通さなければならないという手間のかかる通信問題が発生したため、各州でサーバーを配備した経緯がある。

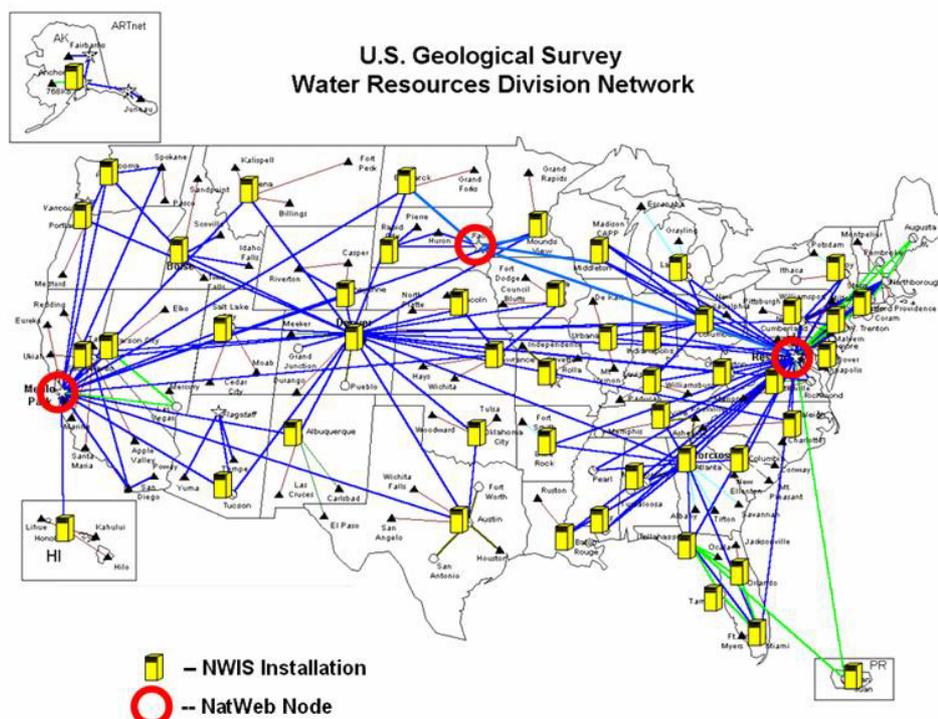


図 3-34 各州に配備された NWIS サーバー

(6) 米国地質調査所 デンバー(U.S. Geological Survey in Denver)

米国地質調査所 デンバーの調査結果の要点を下記に示す。

- 米国地質調査所 デンバーにおいては、MMS と呼ばれるモデルフレームワークを開発している。
  - MMS (Modular Modeling System)とは、
    - ◇ USGS 開発のモデルフレームワークである。(オブジェクト指向ではない)
    - ◇ MMS は、水だけでなく、自然現象、社会現象に幅広く適用できる。
    - ◇ MMS 内部のデータは標準化されている。
    - ◇ モデル構築モジュール (OHyMoS の構造定義ファイルの自動生成機能に近い) によって、各計算モジュールの計算順序、繰り返し計算命令等を自動化する仕組みを有している。
  - 外部データを MMS データ構造に変換するモジュールを作れば、外部データとのやり取りが可能である。
  - 研究者を中心に使用されているが、実務でも活用されている。
    - ◇ PRMS (流出モデル)

- ◇ TOPModel (流出モデル)
- ◇ コロラド州グニソン川流域のリアルタイム水管理システム：開拓局
- ◇ 地下水モデル (MODFLOW) と表面流モデルの結合モデルを開発中
- MMS 自身はある程度完成しており、安定的に動作する。今後 OMS に移行するので、これ以上の機能向上はしない。
- MMS のモジュールにラッパーをかけて、OMS のコンポーネント化を進めている。しかし、この手法では、計算時間がかかるため、今後 10 年程度かけて、MMS のモジュールを JAVA で書き直すことにより OMS のコンポーネント化を図る予定。当面は、MMS と OMS の共存だが、最終的には OMS に移行する。
- Object User Interface (OUI)
  - JAVA で開発した MMS の新しいユーザーインターフェース。
  - 外部データの取り込み、MMS で構築したモデルの実行、解析結果の表示等を行える。
  - OMS 等の他のモデルフレームワークでも使用可能とのこと。
  - 調査時点ではベータ版。マニュアルが整備され次第、一般公開される。

## 1) MMS の開発、サポート体制等

MMS 開発は、1989 年 9 月から始まり、当初は 3 カ年計画であった。Web サイトにおいては、2000 年 7 月に Current Stable Version (安定動作バージョン、動作テスト済み) および Bleeding Edge Version (最先端バージョン) をリリースしている。開発体制は下記の通りである。

- 開発者 (2 機関)
  - ① 米国地質調査所 (USGS)
  - ② コロラド大学水環境システム意思決定支援センター (CADSWES)
- 協力機関 (10 機関)
  - ① 開拓局 (USBR)
  - ② 農務省林野部 (U.S. Forest Service)
  - ③ 航空宇宙局 (NASA)
  - ④ 農業研究事業団 (Agricultural Research Service)
  - ⑤ 陸地生態研究分析研究所(TERRA)
  - ⑥ Deutsche Forschungsgemeinschaft(DFG)
  - ⑦ Rhinish-Westralische Technische Hochschule in Aachen
  - ⑧ University of Bonn
  - ⑨ Friedrich Schiller University
  - ⑩ Potsdam Institute for Climate Impact Analysis
- その他
  - ① MMS の開発・管理スタッフは、現在 5 名。
  - ② サポートとして、ホームページの開設や問い合わせへの回答は行うが、それ専用の部署があるわけではない。
  - ③ MMS は“Public Domain”に属する。

- ④ USGS 以外でも広く使われることを想定している (USGS は他省庁のコンサルタント的な立場との認識)。
- ⑤ ユーザー管理やユーザーによって作成されたモジュール管理は、USGS との共同研究の場合は管理対象とし、MMS 上で動くその他のモジュールについては USGS の管理対象外となる。

## 2) MMS のシステム

図 3-35 に MMS のシステム全体像を示す。MMS は、大きく分けて、1) 前処理コンポーネント、2) モデルコンポーネント、3) 後処理コンポーネントから構成され、それらが MMS 独自のフォーマットで定義されたデータストレージを介して結合している。

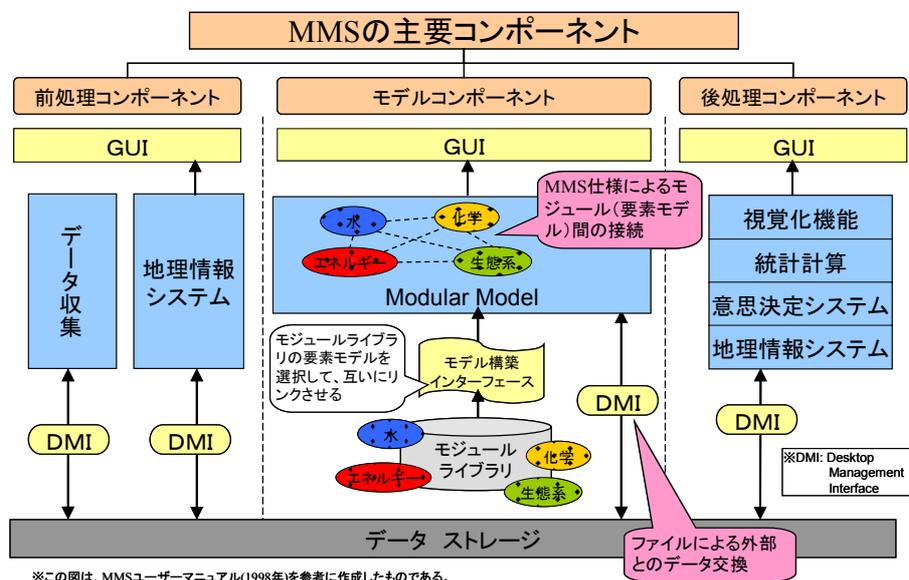


図 3-35 MMS のシステム

図 3-36 に MMS のモデルコンポーネントを構成するモジュール内部の基本構造を示す。モジュールは、5つのサブモジュールから構成される。

- Main Function
  - MMS run controller とモジュールを繋ぐ部分を言い、モジュール内のシステムに命令を与える。
  - MMS からの全ての命令 (Declare, Initialize, Run, Cleanup) は、一度モジュール内の Main 関数に届けられる。
- Declare Function
  - 内部パラメータと変数を宣言する。
- Initialize Function
  - 内部パラメータと変数を初期化する。
- Run Function
  - シミュレーションするプロセスのアルゴリズムを含む。

- シミュレーション計算の際には、タイムステップごとに call される。
- Run 関数は、C か FORTRAN のサブルーチンで構成される。

■ Cleanup Function

- メモリに残っているデータを消去する。

なお、MMS では、トップダウン式に MMS 中央司令塔から各モジュールに命令を出しており、OHyMoS のように各要素モデルが自律的に計算順序を判断して動くわけではない。

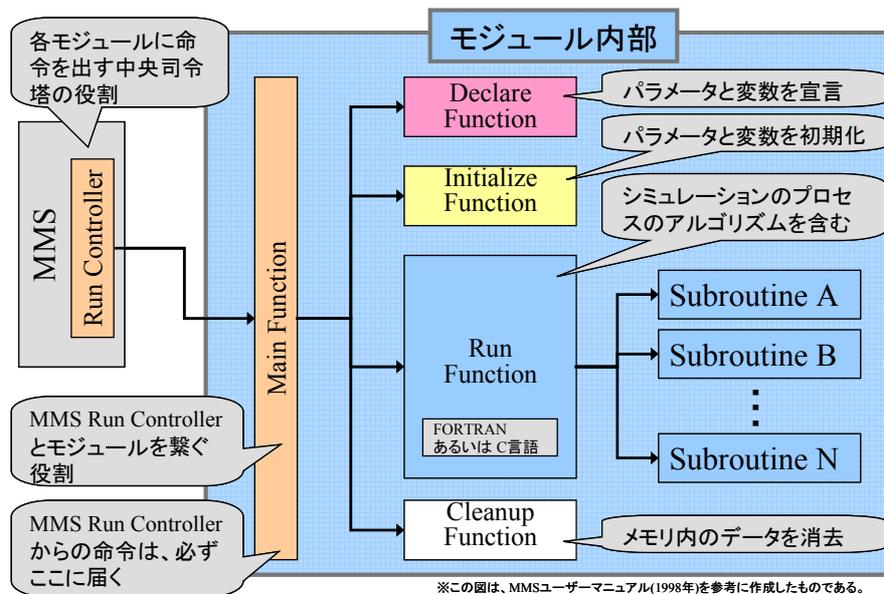


図 3-36 MMS のモジュール内部の基本構造

3) MMS のユーザーインターフェース (OUI)

MMS のツールとしては、Object User Interface (OUI) と呼ばれる JAVA で開発した MMS の新しいユーザーインターフェースが準備されている。

- OUI を用いることにより、容易に外部データの取り込み、MMS で構築したモデルの実行、解析結果の表示等を行える。
- OUI は、OMS 等の他のモデルフレームワークでも使用可能である。
- OUI が現在はベータ版であり、マニュアルが整備され次第一般公開される予定である。調査時点では、MMS のホームページから OUI をダウンロードすることはできない。

(7) 米国農務省 農業研究部(Agricultural Research Service, U.S. Department of Agriculture)

米国農務省(USDA)農業研究部の調査結果の要点を下記に示す。

- 米国農務省 農業研究部においては、OMS と呼ばれるモデルフレームワークを開発している。Object Modeling System (OMS)とは、USDA-ARS で開発中の MMS 後継のオブジェクト指向のモデリングフレームワークである。
- オリジナルは、ドイツのフリードリッヒ大学で開発された。
- JAVA の総合開発環境である NetBeans 上で動作するシステムであり、プロジェクト管理、

モデルコンポーネントの管理（ソースコードの修正、コンパイル、登録等）、モデル構築、モデル実行、解析結果の表示が可能である。

- 過去の資産（Fortran や C）を活かす仕組みがある（JAVA のラッパーの活用）。
- MMS モジュールを OMS 上に移植する作業（ラッピング）が USGS で行われた。
- OHyMoS と異なる事項
  - OMS の各コンポーネントは、内部にモデル接続に関する関数を持たない。
  - コンポーネントの接続関係、受け渡す変数、受け渡しの方向（1 方向もしくは 2 方向）、繰り返し計算の必要性などの情報は別途記述し、その情報に基づき、OMS 上でのモデル構築画面上で、計算順序などを半自動的に決定できる。

## 1) OMS の開発、管理体制等

OMS の開発は、1996 年ドイツの Friedrich Schiller 大学において、開始された。2000 年 10 月に、米国農務省（USDA）、米国地質調査所（USGS）を含んだ省庁間プロジェクトへと展開してきている。開発体制は、下記の通りである。

- 開発機関
  - 米国地質調査所（USGS）
  - 米国農務省（USDA）
  - ドイツ Friedrich Schiller 大学
- 開発のロードマップ
  - Ver.1.2 (Oct 05)
    - ◇ 調査時点（当時）の最新バージョン
  - Ver.2.0 (Jan 06)
    - ◇ NetBeans 5.0 に対応
    - ◇ コンポーネントテストフレームワーク
    - ◇ シナリオ管理
    - ◇ 包括的なエンティティをサポート
    - ◇ モデル接続
  - Ver.2.1 (Apr 06)
    - ◇ 空間的可視化
    - ◇ 自動パラメータ設定
    - ◇ FORTRAN 95 への対応
  - Ver.2.1 完成後
    - ◇ GEOLEM（Geological Object Library For Environment Modeling、環境モデリングのための地理空間オブジェクトライブラリを構築するプロジェクト）の取り込み、データモデリング、データ解析、ClusterRT
  - 外部との共同研究体制
    - ◇ マルチメディア環境モデルに関する関係省庁間の委員会
      - 原子力規制委員会、環境保護局、工兵隊、エネルギー省、地質調査所、農務

省、自然資源保全局、海洋大気庁、開拓局

◇ GEOLEM

◇ 豪州農業生産システム研究ユニット

➤ 中国の地理情報関係機関に OMS 関係者が中国に赴いて説明した模様。

■ その他

➤ OMS 上で動作する数々のコンポーネント（要素モデル）の精度管理や責任のあり方等の管理手法については現在（調査当時）検討中である。

➤ 2005 年 10 月に Ver.1.2 が公開された。2006 年 6 月に Ver2.1 完成を目指している（調査時点における直近の目標）。

➤ USDA の CoLab というプロジェクトトラッキングシステムで管理されている。

➤ CoLab は、ソフトウェアの資産（ソース、ドキュメント、開発者の意見）を皆でより有効に活用していく仕組み。

➤ 結果的には、良いコンポーネントのみが勝ち残っていく仕組み。

➤ OMS 本体の主開発者は 2 名。コンポーネントの開発は外部に多数。

## 2) OMS のシステム概要

OMS に関するホームページを参照して作成した OMS の全体像を図 3-38 に示す。OMS は種々の科学計算、データベースコンポーネントからなるモデルフレームワークであり、GUI モジュール、ユーティリティモジュール（データ検索・ストレージ、GIS、統計処理、可視化パッケージ）が揃っている。また、XML を用いたデータベースマネジメントシステムと Web ベースのモデルリングリソース共有システムを基礎としている。

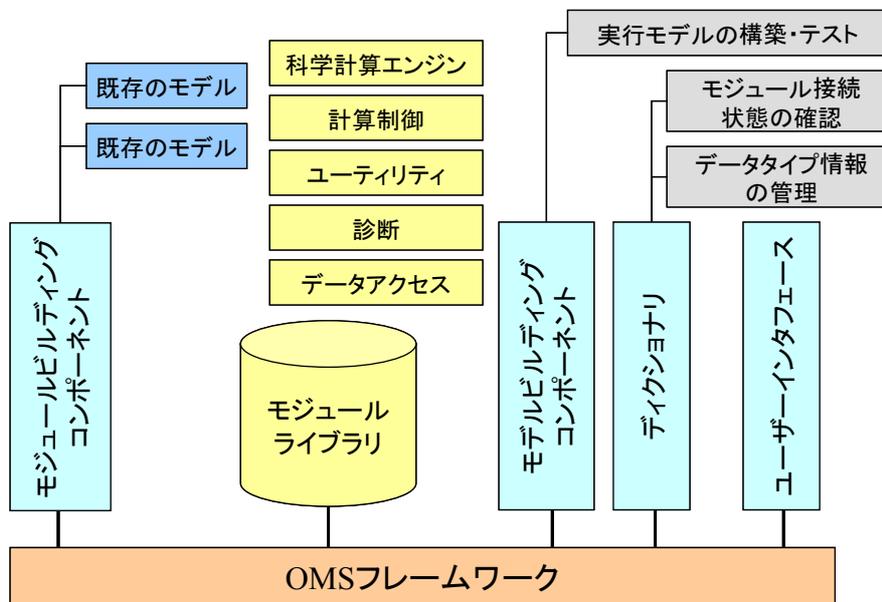


図 3-37 OMS の全体像

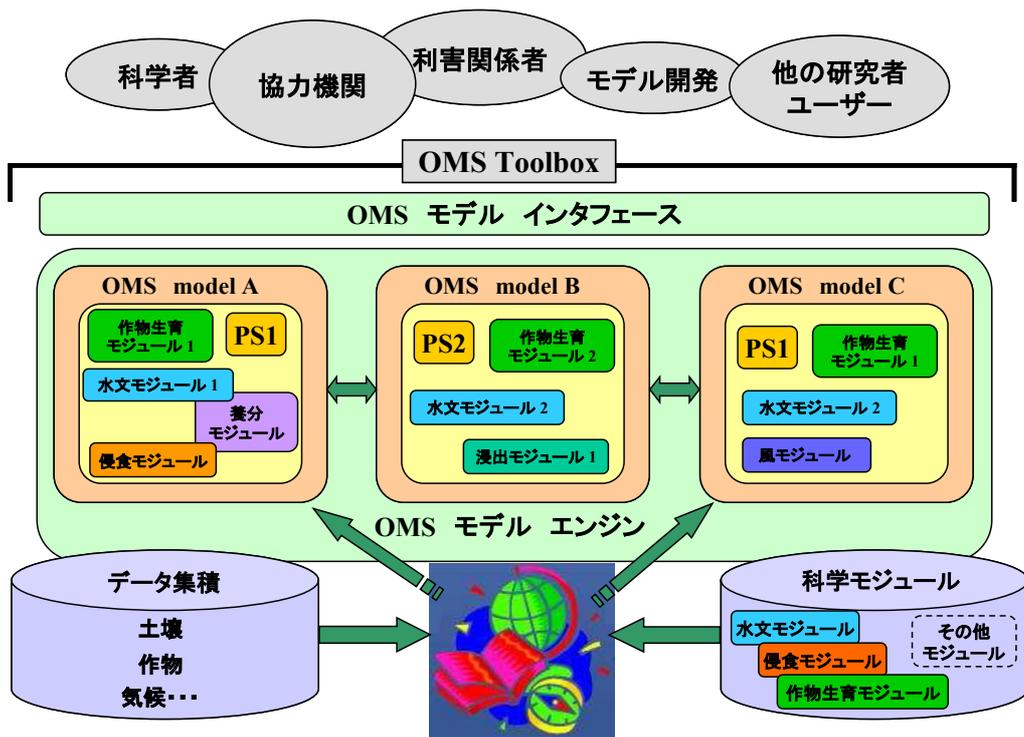


図 3-38 OMS の概要説明図

図 3-39 は OMS のワークフローを示している。コンポーネント開発、開発したコンポーネントライブラリ管理、コンポーネントの統合によるモデル作成と解析実施、さらに解析結果の表示という一連の流れが、OMS の統合開発環境で実施可能である。

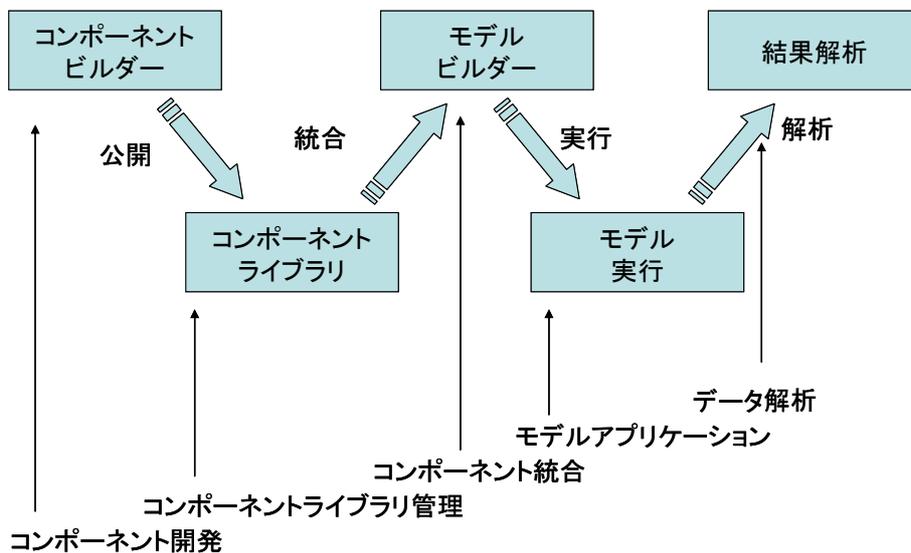


図 3-39 OMS のワークフロー

OMS は、JAVA の統合開発環境である NetBeans 上に構築されているため、要素モデルのソースコードの修正、登録、バージョン管理が容易にできる。また、全体系モデル構築についても、要素モデル自体とは別個に記述されたモデル接続関係をもとに、OMS の GUI 上で

半自動的に構築できる。図 3-40 に OMS の統合開発環境を示す。

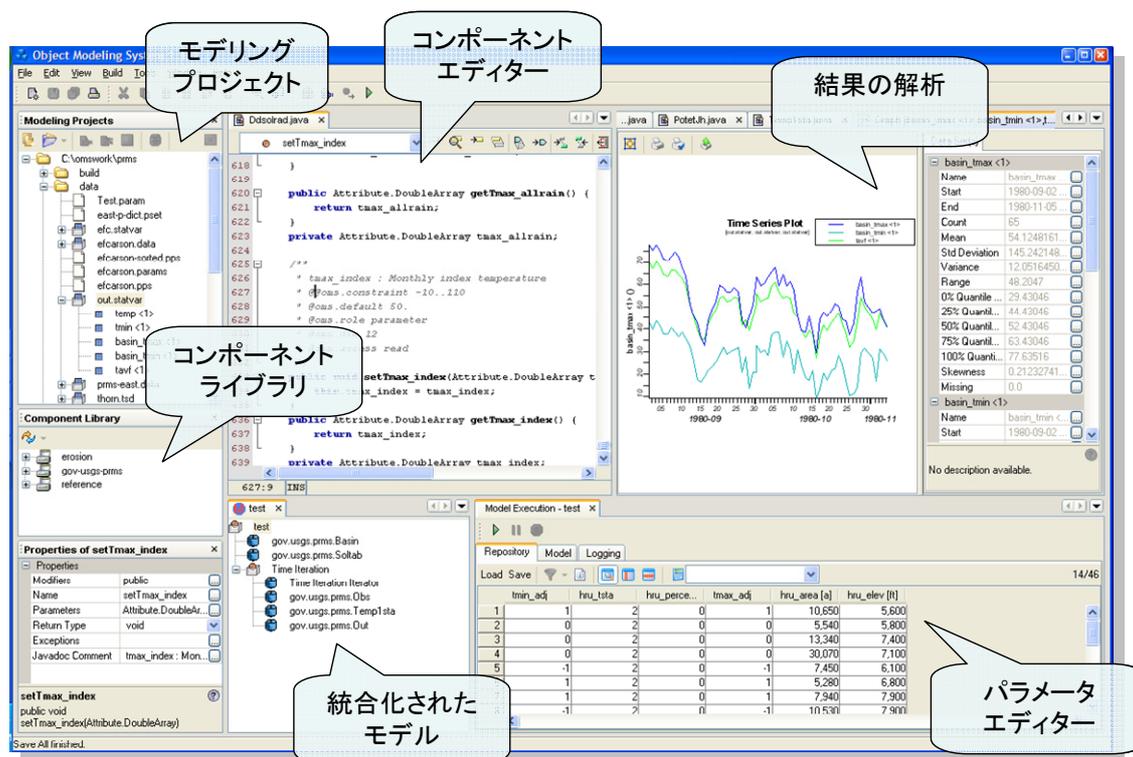


図 3-40 OMS の統合開発環境

### 3) モデリング活動のトラッキングと管理システム (CoLab)

OMS は、USDA が管理している CoLab と呼ばれるモデリング活動のトラッキングと管理システムによって管理されている。CoLab におけるトラッキングおよびバージョン管理システムは次の機能を持つ。

- コードの欠陥から要求にいたる各種問題を追跡、管理する
- プロジェクト管理のためのリアルタイムのタスク、バグレポート
- ドキュメントの共有と版管理
- 議論のためのフォーラム
- ソフトウェアの構築とリリース、ダウンロードに関するレポートを e-mail により得る
- ソースコードの履歴管理と共同開発

図 3-41 に CoLab における OMS 管理画面を示す。なお、この画面は、USDA のユーザーアカウントを得ないと見ることはできない。

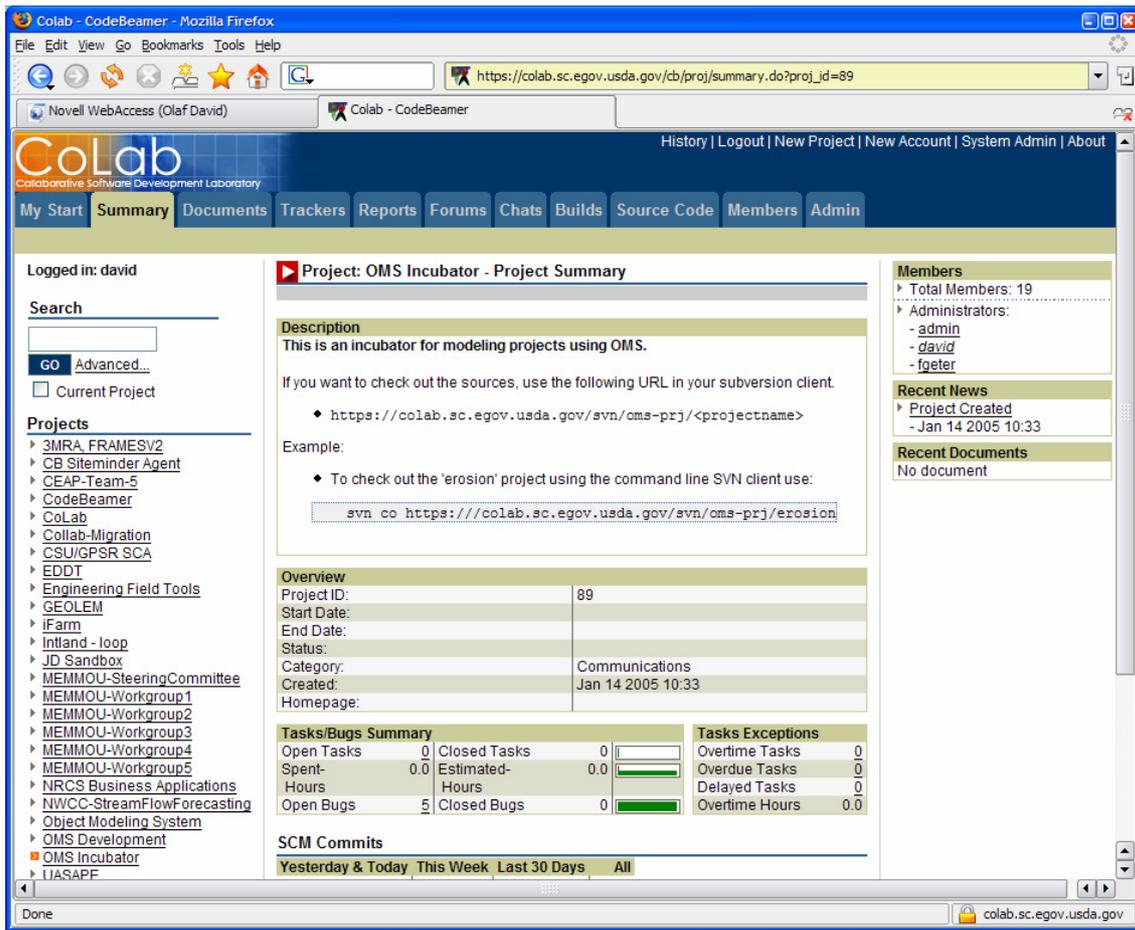


図 3-41 CoLab における OMS 管理画面

## 4 HEC ソフトウェアの開発及び維持管理体制に関する講演録

### 4.1 講演開催の経緯

本講演録は、平成 17 年 3 月 16 日(木)午前 10 時から午後 5 時まで、財団法人河川情報センターにて開催された元米国陸軍工兵隊水文工学センター (HEC) アーレン・フェルドマン氏による「HEC 開発及び維持管理体制に関する講演会」の同時通訳の速記録を基に、吉谷・清水（元土木研究所交流研究員）が編纂したものである。編纂に当たっては、重複した通訳を削除し、誤訳や意味不明部分を講演者の意図を汲んで修正したため、編纂者の主観が含まれている部分があることを断っておく。

アーレン・フェルドマン氏は、20 年以上に亘り HEC の水理水文技術部長として、水理計算モデル HEC-1(洪水予測ソフトウェア)や HEC-HME(流域水文解析ソフトウェア)の開発責任者を務めてきた。フェルドマン氏と日本の交流は、旧土木研究所が行っていたカルフォルニア大学デーヴィス校 Kavvas 教授との共同研究を通して始まった。土木研究所職員が共同研究のためにデーヴィス市を訪問した際に、Kavvas 教授の紹介により HEC を訪問しフェルドマン氏と個人的な意見交換を行っていた。また、フェルドマン氏は、平成 15 年 3 月 26 日に水文水資源学会・国土交通省国土政策技術総合研究所が主体となって開催した「Hydro Modeling and Interface 2003～Software for Hydraulics, Hydrology and Water Quality～」に招聘されている(1.1 参照)。このような経緯から、フェルドマン氏は日本の政府組織や水理・水文・水質ソフトウェアに関する現状についての認識を持つようになったため、最適の講師となったと考えている。

今回の招聘は、世界各地にユーザーを持つ HEC ソフトウェアの維持管理体制と HEC の組織運営を詳しく調べるため、土木研究所ユネスコセンター設立推進本部（当時）が事務局となって行なった。招聘準備段階で、水理・水文・水質シミュレーションモデルの標準化を検討している国土技術政策総合研究所と財団法人河川情報センターに相談し、この関係者向けの講演会も企画することになった。本講演会はこのような経緯で実現した。当講演会の準備として HEC ソフトウェアの開発と維持管理に関する質問を関係者から事前に集約し、それをフェルドマン氏に事前に伝え、その回答に加えて、フェルドマン氏が日本の技術者に有用と判断した内容も解説するレポートを来日前に作成していただいた。講演会での「手元の資料」とある発言は、このレポートのことである。その日本語訳は巻末の CD-R に収録している。

講演会は、日本側が事前に送付した質問をできるだけ網羅するように、しかし、個々の質問に一つ一つ答えるのではなく、できるだけ体系的に日本側に HEC の実情が伝わるよう、フェルドマン氏の判断で講演全体の構成を決めていただいた。これらの交渉は、招聘手続きを行った土木研究所ユネスコセンター設立推進本部（当時）の吉谷が行った。

## 4.2 講演録

### 4.2.1 Corps HEC introduction

#### (1) Water Resources Software Development - HEC's Experience, Observations



HEC は 1 階建ての建物。

<U.S. Army Corps of Engineers(USACE)-Civil Works>

米国陸軍工兵隊 (USACE) は、軍関係と、民間の土木、特に水資源に関する機関の 2 つの使命を帯びている連邦機関です。

当初 USACE は、舟運 (ナビゲーション) の技術開発からスタートしました。次第に洪水防御等のいろいろな工学の研究の要望が出てきて今日に至っています。現在 700 以上のダムを管理し、アメリカ全土にわたる閘門、水門も USACE が管理しています。広大な貯水池から小さな閘門までが対象となっています。他に、洪水の被害軽減、ナビゲーション、水の供給など、活動は多岐にわたっています。

#### (2) USACE Civil Works

アメリカにおいて、USACE は水資源の管理に関しては非常に特異な機関です。陸軍ですが、実際は従業員の 98% が民間人で、軍人は 2% です。ただ 2% の人が与える影響は非常に大きなものがあります。

貯水池の運用、さらに維持、さらに生態系の復旧ということもミッションに入っています。川の流れについていろいろなことをしようと思うと工兵隊から許認可が必要となります。

組織の構成は、本部 (ワシントン D.C.) の下に地域ごとの局があり「division」と呼んでいます。

河川流域の水資源は、このディビジョンが担当します。南太平洋地域のディビジョンは更に幾つかの地域からなり、例えばロサンゼルスであったり、サンフランシスコであったり、サクラメントであったり、ディビジョンがそれぞれのディストリクトごとに分かれています。

ロサンゼルスで洪水の問題があったとすると、ロサンゼルスの政府関係、あるいは市関係、あるいは一般の人が工兵隊に、この洪水の被害があるので何とか助けてほしいというふうに支援を求めてきます。工兵隊はそれぞれの地域の担当が、それぞれの地域で問題があるといろいろな問題解決を図ります。

軍隊ですから、命令指揮系統があります。ディストリクト・オフィスにはカーネルという大佐が

います。将軍がディビジョン・オフィスを管轄しています。さらにその上の位のジェネラルが本部を管轄しているという組織体系です。

### (3) Hydrologic Engineering Center, HEC

HEC は、先ほど述べたそれぞれの地域、ディストリクトにあるオフィスのサポートをするために 1964 年に設立されました。HEC はこの組織を側方からサポートしています。ですから、ヘッドクォーターと一緒にサポートしています。

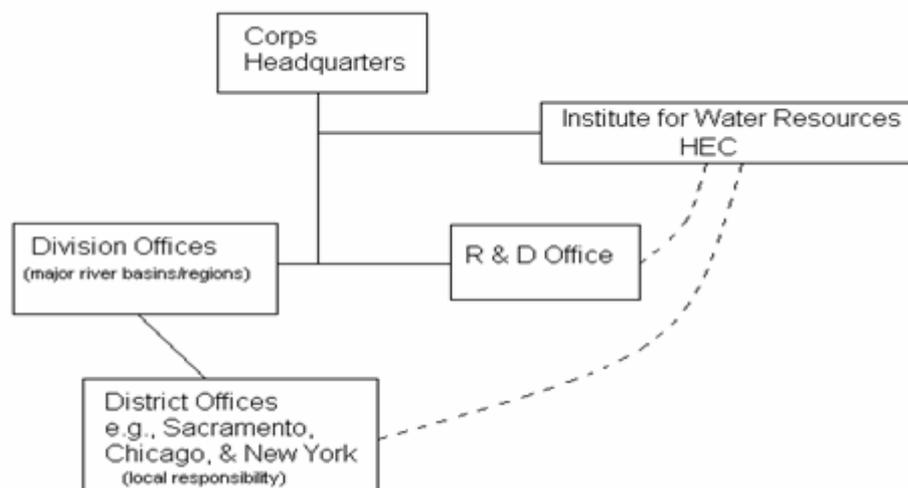


図 4-1 陸軍工兵隊組織における HEC の位置付け

#### <組織>

HEC は組織として小さい方で、30 人ぐらいの専門のエンジニアが働いています。ほとんどのスタッフが大学の修士号を取っていて、5～6 人の博士号取得者もいます。

#### <学生の活用>

HEC はカルフォルニア州のデーヴィス市にオフィスを構えています。そして、カルフォルニア大学デーヴィス校から常に何人かの在校生、あるいは卒業生を受け入れて仕事を一緒にしてもらっています。ですから、我々が教授のように学生を指導するというケースもあります。その学生は、テンポラリー・スチューデント・エンジニアというポジションであり、中には1、2年かけてそれから修士へ進む人もいるし、数年かけて PhD を取る人もいるし、いずれにしても、テンポラリーベースです。正職員として採用するとなるといろいろな査定があります。ただ、テンポラリーの場合はそんなに厳しい査定がないので簡単に採用することが出来るメリットがあります。今も何人かの学生と一緒にやってもらっています。

#### <HEC の活動>

HEC の活動は、Hydrologic and Hydraulic (水理・水文) を基礎として、洪水の被害軽減の計画を立てて、分析をしています。水をどのように管理するかということが大切です。

どのようなツールを使ってその問題解決を図るかということ、一般的に使えるようなソフトウェア開発し、それを使って問題解決に当たっています。実際に研究をしたものをどのようにして使っていかかということで、エンジニア用のマニュアル (EM)、ユーザー用のマニュアルというものを作

って、ガイダンスをしています。

技術支援とかトレーニングを行ったりとか、調査研究を行っています。これについてはまた後で詳しく述べていきます。

水文学においてアメリカの地域のディストリクト・オフィスをサポートしていくことが、我々が焦点を置いている一番大きなミッションです。

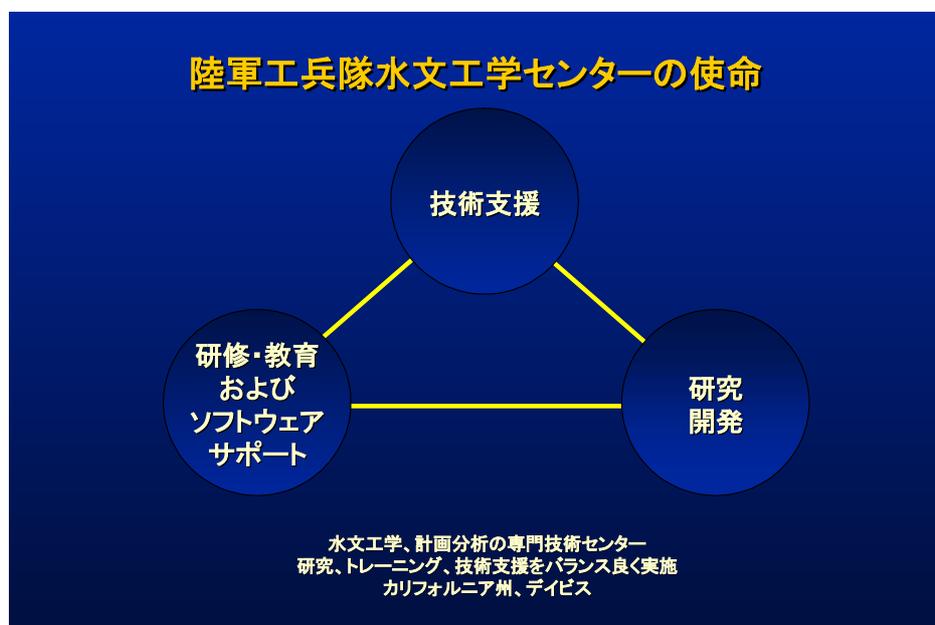


図 4-2 HEC の使命

#### (4) HEC Philosophy

こういった個々のディストリクト・オフィスの問題解決の支援をしていますけれども、一般に適用できるようなやり方で問題解決に当たっていくのが我々HECの大きな理念となっています。

トレーニングも大きな活動目的の1つでありまして、収入はわずかですが時間をかけてトレーニングを提供しています。

#### (5) Technology Development & Transfer

HECの活動がそれぞれどのように相互に関連しているか例を挙げて説明します。

例えば、ロサンゼルス地域で水路の流速が非常に速くなってしまったという問題がありました。どうしたらいいか調査してくださいというふうにHECに要望が来ます。

River Analysis System (RAS) というソフトウェアを使って、流速が速くなったその水路に対して、それを修正する方法を提案しました。ロサンゼルスディストリクト・オフィスから多少お金を出してもらって研究をして、問題解決に当たるといった例です。

また、USACEには研究開発に当たるR&Dオフィスがあります。このR&DオフィスからHECに資金を出してもらったこともあります。いろいろなディストリクト・オフィスから出てきたニーズに基づいてR&Dから要望が出てきます。ですから、このディストリクト・オフィスはR&Dを通して「こんなニーズがあるのだけれども」とHECに言うこともできるし、直接それぞれの現場の

オフィスから HEC に、「こんなニーズがあるのだけれども」というような話がある場合と両方あります。

R&D から調査依頼があるとき、HEC と競合する団体（同じような研究機関）での調査も検討されるので、実行までに時間がかかります。ディストリクト・オフィスから直接 HEC に上がってきた場合はダイレクトに来ますので、短期間で処理できます。

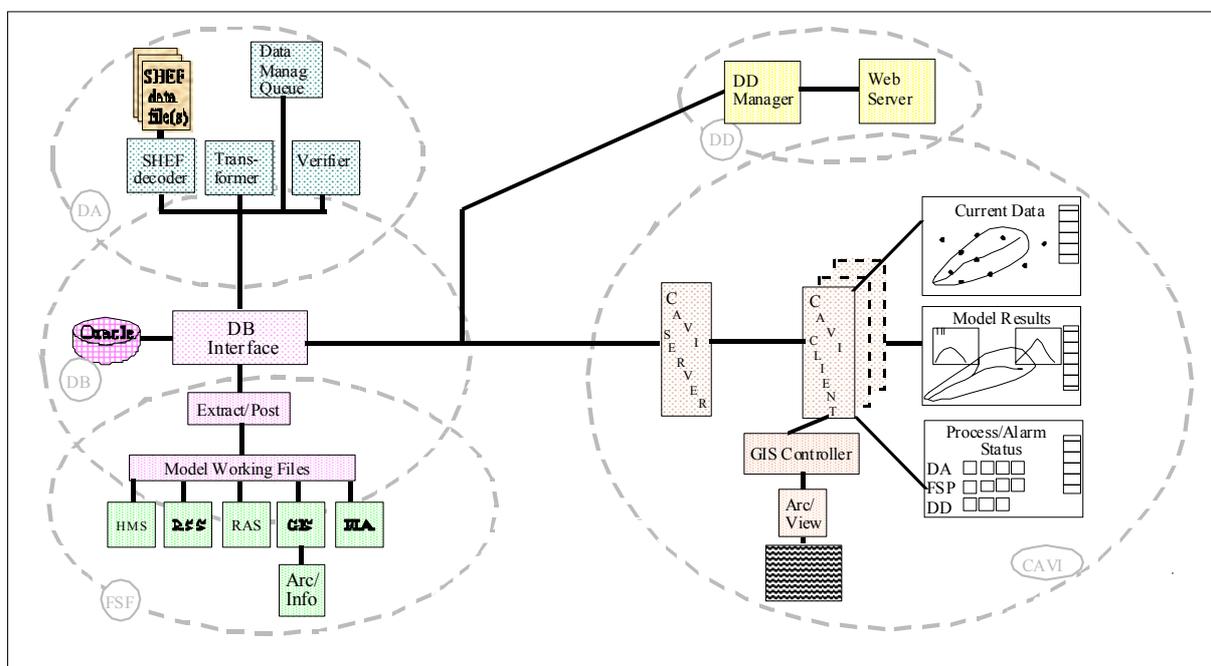
R&D から水理・水文、あるいは洪水軽減、水資源管理について資金を出してもらっていますので、ディストリクト・オフィスからの資金を多少 R&D の資金で補っているという形になっています。

問題解決できた後、そのソフトウェアの使い方、分析のパフォーマンスを、このディストリクト・オフィスの職員の方たちにトレーニングをしていきます。

問題が何か突き止めて、問題を解決して、トレーニングをするという循環で仕事が行っていきま。それが済むとまた新しい問題に取り組むという作業に入っていきます。

<Q： ディストリクト・オフィスの仕事の内容と HEC の関係>

Q： ディストリクト・オフィスの仕事の内容と HEC の関係をお聞きしたいのですが、まずディストリクト・オフィスである問題を抱える、そんなに大きな問題ではなくて通常の水理解析をやるテーマがあるとします。そうすると、ディストリクト・エンジニアリング・オフィスのエンジニア自体がみずから解析計算するパターン、あるいは民間コンサルタントに契約関係を結んで解析をするパターン、それと HEC にサポートをお願いするパターン、の3種類のパターンを仕事の難しさによって使い分けていると考えてよろしいでしょうか？



A： ディストリクト・オフィスは、自分たちで独立しているので、問題解決が自分たちでできるならばディストリクト・オフィス内で問題解決をしていきます。もしディストリクト・オフィス内で問題解決ができなければ、大学に助けを求めてもいいし、HEC に助けを求めてもいいし、ある

いはコンサルタントに何とかしてくれと言ってもいいし、それは問題の性質によって自由に選択していくことです。HEC のソフトウェアを使わなければいけないということではないのです。ですから、我々は使ってもらうように常に良いものをつくらなければいけないわけです。

例えば、ロスアンゼルスの場合、普通は流速の速い水路の分析は大学の教授がやっていたのです。そして、その大学教授が研究したことを今度は HEC に持ってきて、その研究成果を RAS というソフトウェアに盛り込んだという経緯があります。

<Q： 工兵隊は何を管理しているか？>

Q： それぞれのディストリクト・オフィスの管理の範囲はどうなっているのか。例えばロサンゼルスオフィスは何々川と何々ダムを州政府ではなくて、連邦政府が自から管理しているのかを教えてください。

A： 工兵隊が所有しているダムもありますし、開拓局が管理運営しているところもありますし、州がやっているプロジェクトもありますし……。

Q： そのオフィスは工兵隊の管轄内にある施設を自から管理しているという意味ですか。

A： ええ、ちょっと御説明申し上げます。例えば、多目的貯水池の場合、カルフォルニアの州がこの貯水池を管理して水の供給とかに使うわけです。洪水調節のためにこの貯水池の設計に工兵隊が関与してくるという図式です。洪水が起きたときには工兵隊が優先して関与している。洪水がないときには州だけがこの貯水池の運用に関わる。だから、何か問題があると工兵隊が出ていくという図式です。

○司会 これは非常に興味深いお話で、説明に恐らく丸一日必要になるぐらいの問題なので、私の理解している範囲でごく簡単に答えます。日本のような河川管理区域という明確な区域を管理しているという概念ではなくて、その都度にいろいろな組織がプロジェクトを行っています。そのプロジェクトの施設や計画などを担当部局が計画・管理しているという図式になっています。なぜ陸軍工兵隊かという、議会がかなりコントロールしてしまして、最初は舟運のミッションを陸軍工兵隊に与えたことから始まりました。その後、航行可能な区域の構造物建築許可は全部陸軍工兵隊が出すというふうになっています。それは議会が制定した「Clean Water Act」の中で明確に定義されています。そういう関係で、陸軍工兵隊が航行可能な水域に関することは全部所管するという構図になっています。

その辺の趣意は、昔、カルフォルニア大学デーヴィス校の浅野孝先生と一緒にいろいろ報告書を書いてありますので、後で皆さんにお配りしたいと思います。

<Q： コンサルタントとの関係>

Q： ディストリクトからコンサルタントへの委託業務の中で、コンサルタントが問題を抱えたときに、それをサポートする形で HEC が入る仕組みはあるのでしょうか。

A： そういうケースも結構あります。ディストリクト・オフィスが使っているコンサルタントともいろいろやりとりがあります。ディストリクト・オフィスからソフトウェアのアプリケーションをコンサルタントに提供してくれという要望が HEC に来るというケースもあります。サポートしてくださいよという依頼が HEC に来ることもあります。

そこで重要なのは、プログラムがどんなものであるかはディストリクト・オフィスのニーズに基づいているということです。だから、ディストリクト・オフィスから直接要望が来ようが、コンサ

ルタント経由であろうが、ニーズを満たすということがプログラム開発の根底となっています。それが重要な点です。基礎的な研究というよりも応用研究です。実際に実用にするための研究と言えます。

<Q: HEC と同じような組織>

Q: お話の中で、R&D 経由でニーズが来る場合と直に来る場合とあって、R&D 経由で来ると割り振りとかがあるので時間がかかるということなのですが、ということは、HEC と同じような機能を持つ組織がアメリカ国内にあるのでしょうか？

A: 水理実験を行なう、WES という Water Environment System が、これは旧称ですが、ありまして、WES も HEC と同じような、数値的なモデリングも行っています。基礎的な研究に視点を置き、現場に即した活動ではなくて、研究自体に重きを置いています。

#### 4.2.2 HEC products

##### (1) HEC Products

HEC の主な製品は、まずソフトウェアで、次に技術的な方法やガイダンスである「Corps of Engineers Manual」についてのレポートです。インターネットに接続していただいて、工兵隊のウェブサイトに行って「Engineering Manual」を見ていただくことも可能です。

このガイダンスには水理・水文の管理、あるいは洪水の軽減といったことに HEC が責任を持っていますよということが書かれています。

国のダムの安全に関する研究といった特殊な研究も行っています。Federal Emergency Management Agency というアメリカの機関に対しての研究も行っています。

技術移転も大きなウエートを占めています。ウェブサイトを見ていただきますと、ニューズレターとかトレーニングについて書いてあります。

ソフトウェアのダウンロードをするセクションもあります。ウェブサイトにアクセスしていただければ、ここに網羅してある情報は入手可能です。

1 ヶ月に1回、ショートコースを実施しています。3月のショートコースは洪水防御のためのリスク分析でした。来月は川の不定流解析がショートコースとして紹介されています。

このコースは主に工兵隊職員を対象としているのですが、工兵隊外部の方でも受けることは可能です。

##### (2) Major Software Packages

ソフトウェアパッケージについて、幾つか見ていきます。

河川の水理解析、流域の水文、洪水被害、さらに貯水池の分析が、4つの大きな主なパッケージとなっています。

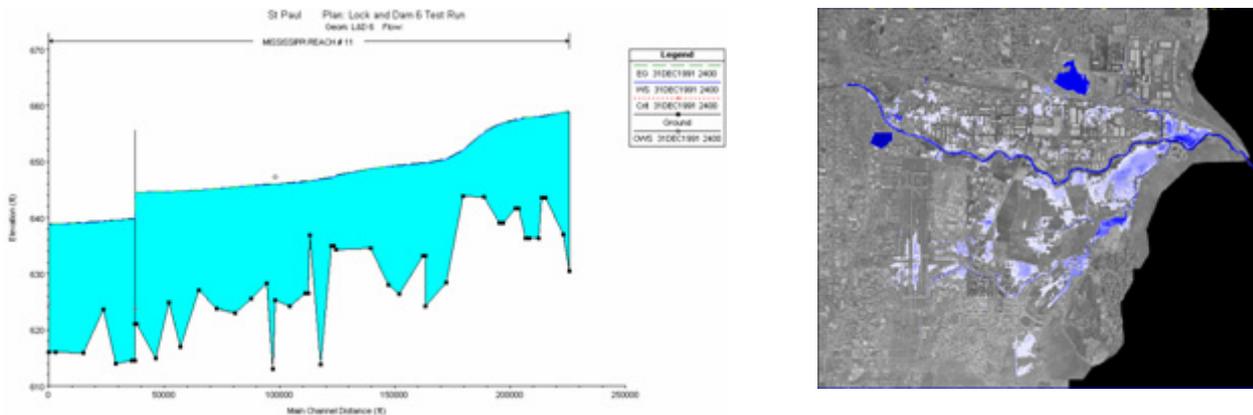
##### (3) Systems Integration, Other

あとはシステム統合を扱っているソフトとしまして、リアルタイムでの水・マネジメント・システムがあります。そして、データ・マネジメント・システムが非常に重要です。時間をずっと追ってデータを見ていくという、データ・マネジメント・システムを開発しました。最近では生態系の

分析についても行っています。

川の水利を考えていく上で、HEC は別に生態学者ではないのですけれども、水利・水文を考えていく上でやはり生態系についても考えないといけないということです。

#### (4) HEC-RAS, River Analysis Version 1.0 Release 1995, Now V 3.1



もう少しそれぞれのパッケージを詳しく見ていきます。

「RAS」と言っている河川の水理解析システムです。河川の水利に関して、非常に実際的な面を見ていきます。流れが安定しているとき、安定していないとき、いろいろな橋の影響、あるいは排水溝、実際は一次元的なソフトウェアなのですが、擬似的に二次元的にシミュレーションしていくということが可能なソフトです。

例えばこのアニメーションはネバダ州の Tahoe 湖の近くにある川なのですが、本流がこういうふうに流れ、水がオーバーフローしたときに貯留しておく氾濫原がこの川の流域にあります。そして、この流れがどういうふうに流れていくか、それぞれの貯留氾濫原との結びつきがどうなっていくかということはこの RAS でシミュレーションしていくことができます。



川の水位が上がると、氾濫原へ流れていきます。水位が下がれば、今度は氾濫原から逆に川へ戻ります。それをこの RAS のソフトウェアがシミュレーションして、視覚的に見せてくれます。

ダムが決壊した場合の分析、これは最初のスライドで見ましたね。閘門とかダムというのはある程度水をためておかなければいけないわけです。いろいろなエンジニアの会社がこのような解析ソフトを競合して開発しているのです。DHI が開発しているものもあります。DHI のソフトでもこういう形で見せることができます。これを何とか RAS のソフトウェアに取り込むことができないかというような問い合わせが来たこともありました。

例えば閘門の「hinge pool operation」においてディストリクト・オフィスがどんなニーズを持っているのかということを探ってくわけです。RAS のソフトウェアパッケージにそのニーズも盛り込まれています。ちょっと特殊過ぎるので RAS に入っていないケースもあるかもしれません。パッケージに入れてほしいという要望が工兵隊のそれぞれのディストリクト・オフィスからあれば、当

然それは盛り込んでいく努力はしていきます。

RAS は水理解析においてはかなり完成されたソフトとなっていると言えますが、ただ例外として土砂の移動については扱えるようにはなっていません。RAS パッケージで土砂について取り扱えるようにしようと、今、開発中です。HEC-6 パッケージというのをお耳にされたことがあると思うのですが、この土砂の移動について扱っていた昔のパッケージが HEC-6 というものです。RAS に取り込むだけではなく、最新鋭のものとしてどういうことを考えなければいけないか、例えば土砂についてどういうふうを考えなければいけないかということを鋭意検討しています。

## (5) GIS について

システムにはいろいろな地理学的な要素が入っていますので、GIS については ESRI にサポートをお願いしています。HEC が独自で GIS を開発しているわけではない1つの例が、ESRI に頼んでいるというケースです。Arc GIS を使って HEC-RAS にインプットするデータをつくるようなユーティリティを作っています。

<Q: HEC の業務範囲>

Q: HEC とディストリクトやディビジョンの役割分担について質問します。例えば、氾濫解析のプログラムをつくって提供するまでが HEC なのか、それとも解析してアニメーションまでが HEC の役割なのか。実際に研修プログラムを開発してトレーニングをして使えるようにという話がありましたけれども、どの辺まで HEC がやって、どの辺までが現場の話になるのか。

A: HEC は単に、アドバイスとかコンサルティング業務だけという場合もあるし、細かく開発してディストリクト・オフィスに、どういうふうにしたら分析ができるかということトレーニングするところまで踏み込んで HEC がやっている場合もあります。

例えば、ミシシッピ川のこういった管理上の細かいニーズがあるかということディストリクト・オフィスから吸い上げます。そして、ディストリクト・オフィスの河川管理計画をシミュレーションするようなソフトウェアを開発し、ディストリクト・オフィスにそのソフトウェアの使い方をトレーニングして技術移転し、その後、実際に使ってみたら、これはうまくいったよというようなフィードバックをオフィスから返してもらいます。ディストリクト・オフィスがグラフを使って、例えば地域住民、あるいはほかの機関に河川管理計画を説明するためにグラフを使うケースもあります。フィールド・オフィスからフィードバックをしてもらうのがニーズを吸い上げる上で非常に重要です。

## (6) HEC-HMS, Surface Hydrology Version 1.0 Release 1997,

Now V 2.2

このソフトの対象は降雨です。雨、あるいは雪が降ってきた場合に、その降ってきた雨水がどういうふうに流れるかというお話をします。

RAS はマイクロソフトの FORTRAN で開発されましたが、Java 言語に基づいてこの Hydrologic Modeling System、HMS は開発されています。HEC-1 をさらに発展させたものが HMS です。このソフトウェアは降雨、雪解け、あるいは、降った雨がどういふふうに流れるかといったもろもろの水理・水文現象を解析します。HEC-1 からの大きな変更は、空



間的にグリッドでこの分布を解析する点です。雨水がどういうふうに流れるかというのは空間的に見ていかなければいけないプロセスですから、降雨はレーダからマッピングして解析する必要があります。これは GIS システムを使ってこのモデルシステムをサポートしています。使っているのは Arc GIS システムです。

このユーティリティを開発する場合には ESRI とかなり緊密に連絡をとって進めていきました。ESRI とどのようにして協力的な研究開発協定を結んできたかというのは、後でソフトウェアの開発の話をするときに触れていきます。

<Q： レーダ雨量のメッシュの大きさ>

Q： レーダ雨量の下の図のメッシュの大きさ、1 km×1 km なのか。それと降雨量としての精度管理、降雨量としての精度についてはどう考えているか。

A： 4 km です、周辺が。もっと進んだものと 1 km というものもあります。

<Q： レーダ雨量のキャリブレーション>

Q： 日本では、地上雨量計が一応正しいものとして、レーダ雨量計をそれに合わせて、降雨分布が得られます。米国ではどうしているのか？

A： 地上とレーダと合わせてやるし、気象モデルも加味して予測するという場合もあります。いろいろな手に入る情報を統合してやるのが一番いい評価法です。

#### (7) HEC-FDA, Flood Damage Analysis Version 1.0 Release 1977, Now V 1.2



図 4-3 洪水被害/影響解析 HEC-FDA

FDA は浸水した地域のいろいろな建築物、建物を細かく分析して見ていきます。それぞれの浸水した建物はどれぐらいの程度の被害を受けたか、あるいはどのぐらいの浸水深になったかというよ

うな情報があります。FDA 水理パッケージによれば、水の深さがわかります。そして、水の深さと関連した被害の程度を洪水被害分析ソフトが計算してくれます。そして、どれぐらいの水の深さからどれぐらいの被害があるか、あるいはどれぐらいの深さになりうるのか、そしてどれぐらいの頻度で浸水するかをコンピュータで計算し、1年間当たりの被害を予測計算することができます。

ディストリクト・オフィスはこの被害を軽減することによってどれぐらいのベネフィットがあるかということと、そのためにどれぐらいコストをかけなければいけないかという、そのコストとベネフィットとの相関関係を天秤にはかるということをしています。当然、そのプロジェクトごとに、これはこれぐらい役に立つものだ、ただその分、これぐらいのコストがかかるということをプロジェクトごとに査定していかなければいけないわけです。

そして、GIS からもサポートをしてもらっています。そして、洪水被害を受けた地域の1つ1つの建物の情報をこの GIS によって得ることができます。

#### (8) HEC-ResSim, Reservoir Analysis Version 1.0 Release Spring 2003



図 4-4 貯水池シミュレーションシステム HEC-ResSim

貯水池をうまく運用すれば、洪水被害を削減することも可能になります。そこで、多目的貯水池パッケージ (HEC-ResSim) をつくりました。いろいろな目的を持った貯水池の運用のシミュレーションをしていきます。水力発電、水の供給、あるいは洪水制御、環境的な目的、あるいは灌漑用に下流で水の供給のコントロールもしていくというようなことをやります。

水力発電システムへのこんなことが必要だということの中にこれを加えていくことができます。

この貯水池はある水力発電の目的に合致している、こっこの貯水池はまた別の水力発電の目的に合致しているなどというふうにシミュレーションすることによりわかります。以前の HEC-5 というプログラムでは特殊な、例えばチャネル (水路) のキャパシティだけに特定したものであったり、あるいは貯水池放流に特化した情報しか扱わないというのが HEC-5 でした。

新しいこのパッケージは、下流からのいろいろな要望に基づき、どれぐらいの流量が必要かというようなことも計算できます。この貯水池満水状態であり、別の貯水池は貯水率が低い状況になっ

ている。そして下流で、ある一定のルールがあります。これらの条件全てに基づいてシミュレーションしています。

貯水池、貯水池運用のシステムをかなり包括的に、幅広く取り込んだ最新鋭のソフトウェアパッケージになっています。

工兵隊の水マネジメント・システムについては、非常に重要なので後ほどお話をします。

<Q: HEC システムの対象国>

Q: こういうパッケージは構成上、アメリカ国内のみを対象としているのか、または海外に対して使うような作り方をしているのでしょうか？

A: 工兵隊向けにつくっているのですが、一般でも使っていただけるようには考えていますから、外国でも使えるはずですよ。

<Q: 海外支援での実績>

Q: 例えば、アメリカがどこか海外支援に行ったときにそこでこういうシステムを使うとかいうような具体的な実績はありますか？

A: 例えば、スマトラ沖のああいいう大規模災害があつて、アメリカから支援に行くというように、もちろん使えます。ローカルデータで記述していますが、カルフォルニアの川の状況の分析もできるし、例の津波の被害の分析もできます。

<Q: 精度管理・品質保証>

Q: ソフトウェアの品質保証、ステータスは、あるいは適用条件の明示みたいなものはオープンにしているのでしょうか。

A: それは後でちょっとお話をしますけれども、答えはイエスです。クオリティ・コントロールをどういうふうに行っているかというのは、後でお話をいたします。その点は非常に重要です。世界中で HEC のソフトウェアは使われています。HEC にしてみれば、多くの人が HEC のソフトをテストしてくれているということになります。エラーがあつたらどんどん連絡してくださいというようにお願いしています。

<Q: HEC の人員配置>

Q: HEC の組織としては 30 人ぐらいで、小さなオフィスでやっているとのことですが、ディストリクトの問題解決の支援ソフトの開発と同時にサポートもしている、トレーニングもしている、更に、海外にも目を向けているというのは、物理的に、人数的に足りているのでしょうか？

A: やはりかなり汲々してやっていますというのが現実です。これまでの技術を継続していくと同時に、新しい技術にどれぐらいの HEC の人員を配置していくかというような問題がありますので、HEC のスタッフの能力を広げていく上でも、コンサルタントと契約しています。ソフト開発で、どういうふうに行っているかというお話も後でさせていただきます。

Q: そうすると、平均的に 30 人がコアで、学生さんとかいろいろ入ってきて、客員教授みたいな人もいて、大体平均的に何人ぐらいでやっているのですか。

A: 60~70 人というところですかね、まあ、予算にもよりますが。(笑声)

Q: その 60 人から 70 人の内訳ですが、今、メジャーのソフトが例えば 4 つあつて、それ

それぞれのチームを構成するというのは大体等分に配分されているのでしょうか？

A： パッケージによっては、大体3～4人が大体永久的にそれをずっとやるという人です。チームについては後でまた申し上げますけれども、ESRIとかいったコンサルタントに頼むときもあります。

<Q： HECの雇用形態>

Q： その30人のコアメンバーというのは、結構長く勤務されるのでしょうか。日本の場合だと転勤という制度があって、3年ぐらいで入れ替わるのですが。

A： 離職というのは少なく、非常に定着してずっと長くやっているケースが多いです、喜んで仕事をしてもらっているみたいなので。(笑声) このディストリクト・オフィスの現場から来ている方もいらっしゃいますし、これはという人を現場のフィールド・オフィスから引っ張ってくるというケースもあります。委託先のコンサルタント会社から引っ張ってくるという場合もあります。大学からとか、あるいは政府関係の機関から採用してくるという場合もあります。

例えば今のHEC代表者は、前はコンサルタント会社で働いていた方です。Development and Resources Corporationというコンサルタント会社からHECに来られました。いろいろな経歴を持ったスタッフがHECで働いているというのも1つの利点かと思えますね。大学の教授を招いて仕事をしてもらおうというようなプログラムもあります。あと、ディストリクト・オフィスからHECに来て、担当するプロジェクトに取り組んでもらうというケースもあります。

<Q： ソフトウェアの拡張はその少ないメンバーで進めるのか？>

Q： 例えば今、一次元のモデリングシステムをHEC-RASという形でつくっていると思うのですが、将来的にそれを二次元あるいは三次元のモデルに拡張していくという動きはあるのでしょうか。

A： 徐々にではありますが、動きがあります。川の水利では準二次元というのを今使っていますけれども、土砂の移動については多次元のことを考えています。将来必要となってくると思います。

#### 4.2.3 Software development

##### (1) Early Software Development

それでは、ソフトウェアの開発についてお話をしていきます。

ソフトウェア開発の歴史、これまでどういう経緯でやってきたかお話しします。どんなリソースが必要か、今はそれがちょっと変わってきているので、そのリソースの変化についてもお話をします。

コンピュータは、昔はハードウェア上にいろいろな制約があって、スピードも遅かったし、計算速度も遅かったというようにいろいろな制約がありました。シミュレーションをするときに、私は最初にやっていたのはカードです、メインフレームの大型コンピュータにインプットするというのがコンピュータ・テクノロジー・テクニシャンとしての最初の仕事でした。

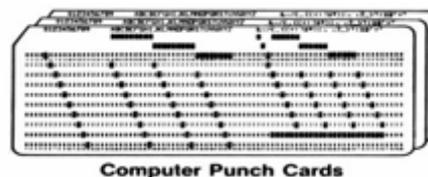
どうしてもサイズが限られていましたから、プロセスそれぞれに別個のプログラムが必要だったわけです。HEC-1ではまず、降雨の解析があって、雨水が地面にどういうふうに流れるかというのはまた別のプログラムがあったわけです。川の流れはどういうふうなルートをたどるかという解析

には別のプログラムが必要でした。

ハードウェアの進歩によって、HEC-1 をパッケージとしてまとめることが可能になりました。

## (2) PC Software Development

現在ではグラフィック・ユーザー・インターフェースが使えるようになりましたから、昔はやっていた“Look and feel”、見て、触るといようなやり方はちょっとばかげたやり方のように、もう昔の話になってしまいましたね。



昔のプログラムは全て大型メイン・フレーム・コンピュータの処理用でした。昔はソースコードというのを配布して、それぞれのユーザーの別個のコンピュータで編集してもらっていたのです。ソフトウェアの維持管理においてはパソコンが出てきてから飛躍的に変わりました。コードをメインフレームから PC に変えていくというのがまず我々がやらなければいけない最初の仕事でした。

同じ OS でどの PC にも使える実行可能なコードをまず配布していきました。

日本はパソコン開発の上で先導的役割を果たしてきましたから、パソコンの能力がいかに早く拡大していったかというのは皆さんよくおわかりのことだと思います。

データをインプットする上で、ユーザーをアシストしていくためにメニュー・プログラムというものをまずつくっていきました。今使っているグラフィック・ユーザー・インターフェースの前準備の段階としての作業でした。

## (3) Modern software development

1990 年ぐらいでメインフレームからパソコンへの転換というのは行われたわけですが、さらにコンピュータ科学の能力というのはどんどん進歩しました。1990 年に HEC では次世代ソフトウェアの開発に取りかかりました。RAS とか HMS だとか、Reservoir System の開発のソフトウェア等がそのさきがけとなってきたのです。

次世代ソフトウェアについてちょっとお話をしてから、1996 年から始まった統合システム（インテグレートッド・システム）についてお話をしていきます。

新しいコンピュータの能力が進歩してきた中で、HEC の内部ではチームをつくってその進歩への取り組みを進めてきました。コンピュータ・プログラミングを行っていく上で文化的に変わってきたと言ってもいいと思いますが、HEC 所長のデーヴィスさんがそれについて触れていました。

昔は1つのプログラムに1人のエンジニアというソフトウェアの開発をしていました。HEC-2 はビルさんという人が全部一人でやっていたので、HEC-2 についてのいろいろな専門的な情報というのは、そのビルさん一人の頭の中に入っていたわけです。これは FORTRAN を使ってプログラミングをしていました。次世代ソフトウェアの開発では新しいコンピュータの進歩に対応していません。

HEC の部署を再編して、いくつかのチームを作りました。工学上のアルゴリズム、十進法の計算法、ハードウェア環境、コード構成と言語、ユーザー・インターフェース、データ管理とソフトウェアの統合をそれぞれ見ていくチームです。チームがそれぞれのアイテムを見ていくという体制で取り組んできました。

技術的な仕事ということもあるのですけれども、HEC のいろいろな部署からの人が集まっているということで、いろいろな新しい考え方が融合して新しい考え方をくり出してきたということが重要だったのです。そして、いろいろなところの部署から集めてきて混合チームをつくって、「う〜ん」というような状況をつくり出してしまったわけです。昔からやっていた人が昔からのやり方を踏襲するということはもうできなくなったわけです。だから、「う〜ん」という状態になってしまったわけです。

#### (4) Software Development Goals

次世代ソフトを開発する上でどういうことを目標、ゴールとしているかということ、我々は水文工学のエンジニアですから、最新鋭の水文工学、あるいは計画の分析を特徴づけていくということがまず注目してやってきたことでした。古いソフトウェアをそのまま新しいものに転換するということではなくて、古いソフトウェアがどういう能力があったのかということ吟味、査定して次につなげていく、新しいものにつなげていくということです。

ソフトウェアを開発していく上で、構成要素としてはコーディング、さらに言語を見ていったわけです。さらに、ユーザー・プラットフォームを見ていきました。ソフトウェアを構成するものがどんどん変わっていつてしまうので、コーディングとかランゲージを見ていくのは非常に大変な仕事でした。

FORTRAN、C++、あるいは Java というように移っていきましたが、やはりコンピュータの科学的な技術が進んでいくのに、それに歩調を合わせていくというのは非常に努力の要ることでした。ソフトウェアを開発する上で先導的な立場をとっているというよりも、もう産みの苦しみを味わうような、そういう大変な作業で取り組んできました。行き止まりに突き当たってしまったって、もう一回研究をやり直したというようなことも多々ありました。

新しいソフトウェアの大きな能力の1つがグラフィカル・ユーザー・インターフェースというのですが、広範にわたってディスプレイができる、あるいはグラフができるというものが大きな特徴です。ソフトウェアが非常に柔軟性を持って、ほかのケーパビリティをも取り込めるように、このソフトウェアのケーパビリティを効果的にどんどん拡大していくというのが大きな努力の目標でした。

1つ以上のプログラムをサポートできるライブラリもつくっていきました。あと、ソフトウェアのメンテナンスにかかるお金をできるだけ削減していくということにも注力しました。そしてもう一つ注力したことは、ソフトウェアを常にパブリックドメインにおくのだということを念頭に置いてやってきました。

アメリカ以外の国は、なぜそのソフトウェアをいわゆるパブリックドメインに置いておかなければいけないのだというのを不思議だなと思われる方がいらっしゃるかと思います。HEC のソフトウェアの開発の資金というのはアメリカの税金で賄われているのです。やはり税金で賄われているからには、当社が開発した製品というのは公共の用に資するべきだというのが基本的な考え方にあるのです。そして、広く使っていただければ、それだけ我々にとっても利点があるというふうに考えています。

## (5) Software Development - continued

これまで、ちょっと前には随分トレーニングを行ってきました。マイクロソフトの Windows と Unix と両方で開発を開始しました。マルチ・プラットフォーム・ケーパビリティというのがありまして、マイクロソフトのパソコンでも Unix でも両方とも使えるという非常に使い勝手のいいものです。それをやるのは非常に大変でした。

最初の River Analysis System はマイクロソフトの環境でしか使えませんでした。マルチ・プラットフォームにするよりも、MS に限ってしまった方が早く、そしてよいソフトウェアが開発できるのでそういう形になったわけです。HMS はマルチ・プラットフォームを Unix でギャラクシーと呼ばれる言語でスタートしました。ギャラクシーは Java の前のバージョンと言えると思います。

## (6) プログラミング言語 : Java オブジェクト指向

HMS はギャラクシーから始まって Java を用いるようになりましたが、Java が発展してきたので、ギャラクシーは余り使い勝手がよくなってしまいました。当時はギャラクシーは最新鋭の技術でしたが、コンピュータ技術が進むにつれて、ギャラクシーは Java にとってかわられてしまったわけです。もうすぐ出る HMS Ver.3 は、すべて Java で作成しました。

グラフィック・ユーザー・インターフェースの開発は非常に難しく、大変でした。ブッシュ大統領もよく言う、「It's hard」、(笑声) 難しい、難しいと言っていたわけです。

FORTRAN から新しいシステムに転換していくというのはなかなか大変だったのです。特に、Unix は扱いが大変でした。大学の人に来てもらって、C++についてまず講義をしてもらったわけです。新しいソフトウェアの開発環境において、いわゆるオブジェクト指向の流儀に慣れる必要がありました。カルチャーショックでした。

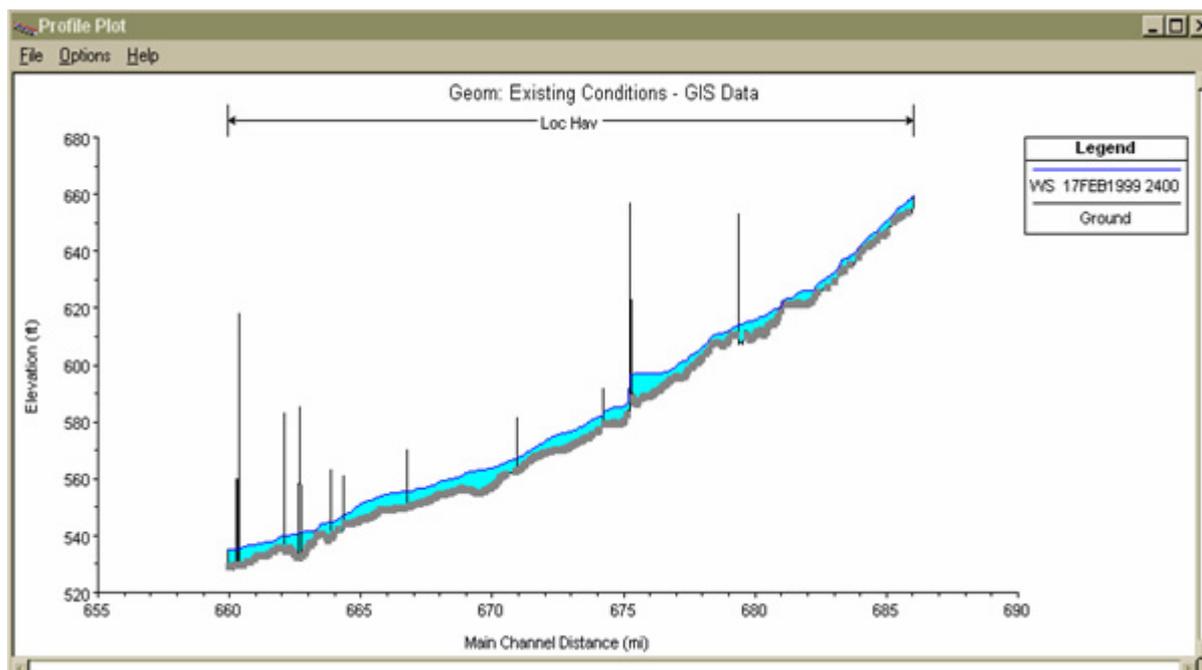
オブジェクト指向のデザインとコーディングというのは、FORTRAN から転換していく上で非常に難しかったのです。「改宗する」ぐらいの大変さがあったわけです。このオブジェクト指向ということを理解して、そして「これだ！」というふうにスタッフがわかったときに、もうこれしかないというように、宗教を広めるみたいな感じでオブジェクト指向 (オブジェクト・オリエンテッド) を説いて回るようになったほどです。

## (7) Guiding Principles

HEC では、実証された最新の技術やコンセプトを水文エンジニアリングに適応しています。それは常に連携をとってディストリクト・オフィスからのニーズの解決に役立っています。そのために、ソフトウェアのデザインをしていく上でディストリクト・オフィスのエンジニアに HEC に来てもらって、取り組んでもらうというような活動もしてきました。

Windows と Unix 両方ということはさっき申し上げました。

オブジェクト指向のプログラミングのメリットの例は、再利用可能なライブラリです。よい例が、グラフィックスです。HEC のグラフィックス・ライブラリを使って下のグラフ等をつくってきたわけです。



#### (8) DHI へのグラフィック・ライブラリーの外注

資金的にも余裕がなかったため、このグラフィック・ライブラリーを開発していく上で、デンマークの会社と契約・ベースで取り組んだことがあります。そして、その契約の中には、HEC が自由にソフトウェアを、実行ファイルで配布できるという条項を盛り込んでおきました。ですから、HEC がそのソフトの使用権を取得すると、ディストリクト・オフィスはその使用権を別個に取得する必要がなかったのです。しかし我々のグラフィック・パッケージを変更するときには、またデンマークに頼まなくてはならないわけです。

デンマークの会社は我々以外にもほかに取引先がありますから、我々がこれと言うパッケージをそのデンマークの会社に頼んでも、すぐに変更してもらおうということはなかなか難しかったのです。当然、いい会社でしたから、我々よりもより優先順位の高いお客さんもいたわけです。まあ、それはしょうがないことだと思います。「OK、もう新しいソフトウェアについてはもう十分わかったので、自分たち独自でパッケージを開発していこう」ということを決めたわけです。

HEC のグラフィック・パッケージは HEC の RAS とか HMS とか ResSim とか Flood Damage Analysis とかのすべてのソフトウェアに使われています。

グラフィック・ユーザー・インターフェースのデザインも大きな仕事でした。コンピュータ・インターフェースを水文のエンジニアとしてどういうふうに設計したらいいのかということ、随分たくさんのエンジニア間で、会議で討議してきました。

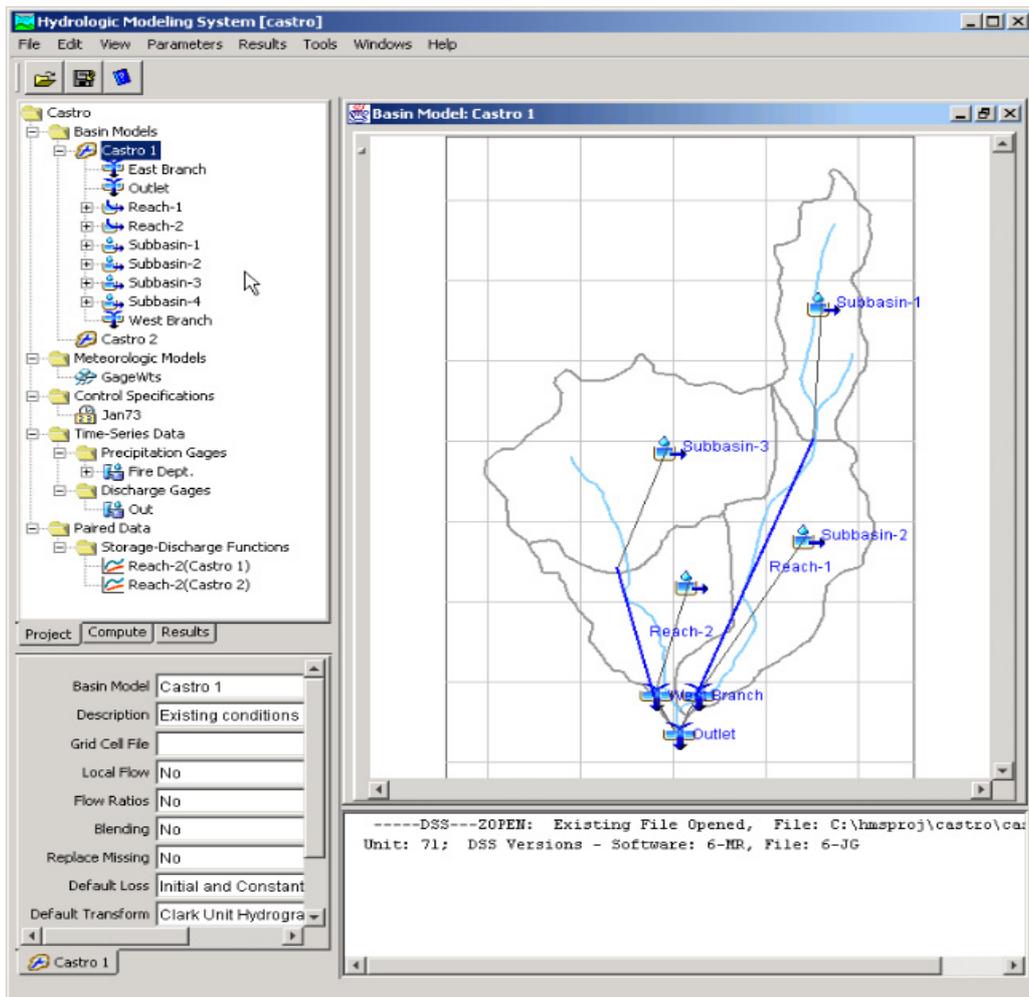
水文エンジニアとコンピュータ技師を協調しながらやっていくためにどうしたらいいかということ、それを述べた文献を文書で出したりということもしてきました。そして、その文献を新しいグラフィック・ユーザー・インターフェースを開発していく上で活用しました。基本的なソフトウェアを開発して、それを正しく適用し、それを使ってもらう。そのいい例としてグラフィック・パッケージと GIS というソフトウェアの開発があります。

## (9) GIS システムの開発

1970年にHECでGISシステムを開発しました。地図データを入れて、分析できる最初のシステムでした。当時としては時代をリードするような非常に新しいものでした。80年代になってESRIの協力があって、やっとGISも一般的に使われるようになってきたわけです。

HECでGISとかArc GISとかARC/INFOを活用したソフトウェアを開発しました。それらを発展させて利用するためにユーティリティをきちんとつくっておくのが重要です。

## (10) HEC-HMS Graphic User Interface



グラフィック・ユーザー・インターフェース (GUI) について、具体的に見ていただきます。

これは水文モデリングシステムです。ウィンドウズのエクスプローラとこのインターフェースとはちょっと似たところがあるとお気づきの方もいらっしゃると思います。GUIには4つコンポーネントがありまして、ネットワークがここに描写、表現されています。ウィンドウズのエクスプローラでもファイルが出てるように、ここに表示されています。これも水文に関する表示です。システムの絵が出ていまして、河川のサブ流域ナンバー1が書いてあって、この流域の一部、サブ流域ナンバー1が示されています。データ編集をする場合には幾つか方法があるのですが、デ

一タのアクセス法は幾つかありまして、ここにウィンドウがあって、ハイライトされたデータのエレメントはこの下に出てきています。最後のところですが、どういう活動が行われるかということについての情報がこの右下のところに出てきています。

これは HMS の新しいバージョンです。ほかのソフトウェアはまた別の考え方を持った人がやっていますから、これとはまたちょっと違ったものになっています。1つの方法だけで、これでやろうというように、全員が納得して合意したというようなものはありません。それぞれ開発チームが検討しながらやっていくということが可能なわけです。HMS の担当の人はこれが一番いいと考えていますし、Reservoir System の担当者は、うん、これはまた Reservoir にも使えるなというように考えるかもしれません。

データを関数を含めて標準化することは、それを編集する上では非常に強力な武器になるものだと思います。流域の状況を記述するだけではなしに、いろいろな水の流れとか、あるいは雨水の動きをつかむこともできます。

雨量と流量は、データが対になっています。ストリーム・システムは、中で流れがどういうふう流れていくかの、ルートを見ていくシステムです。さらに、アウトプットを見ていく上でも使うことができます。

このツリーの中に結果が組み込まれています。このシミュレーションを見たいというときにはそれをハイライトして、クリックしてグラフの上で再現することができます。それぞれのグラフィック・デザインを見たいときにはそこにアクセスしてクリックすれば、その結果が一目瞭然に出てきます。非常に強力なインターフェースだというふうに我々は確信しています。

#### (11) Sources of Funding

では、資金源についてお話しします。

主に3つあります。①研究開発 (R&D : 研究開発に応用するためにこの工兵隊の Research & Development Division) から出ている場合と、②工兵隊内の Automated Information Systems、(「AIS」と呼ばれる)、③ディストリクト・オフィスのそれぞれの要望に基づいて資金が出される場合、の3つです。

HEC がそれぞれの現場のオフィスといろいろ相談して、年間のそれぞれの事業ユニットをつくっています。そしてこのフィールド・オフィスの代表が集まっているいろいろな年間の事業のうち、どれに優先順位をつけて実行していくかということ投票形式で決めています。主に RAS とか HMS とかいったソフトウェアはこの R&D からの資金で賄われたわけですが、一部、「Specific Offices」と呼ばれるディストリクト・オフィスの資金で賄われた部分もありました。

Corps Water Management System は Automated Information System にサポートされていますが、それについては後ほど説明します。

かなり大きなソフトウェアの開発は、AIS が関与しています。工兵隊すべてにわたって使えるような大きなプロジェクトがこの Automated Information System によって行われています。この AIS というのはそれを十分賄うだけの規模があります。

「Life Cycle Management of Information System」という非常に詳細にわたったシステムがあるのですが、AIS システム開発の上でやっている軍としての取り組みのシステムとなっています。まず、非常に圧倒されるようなプロセスなのです。開発工程の全体において、ユーザーのニーズを

インプットしています。それぞれ特定のオフィスについてはさっきお話ししました。

Life Cycle Management System については私のこの文献に詳しく述べてありますのでソフトウェア開発の担当者の役割についてお話をしていこうと思います。

## (12) Participants and Roles

開発に当たってメンバーは、HEC から学会からもユーザーであるディストリクト・オフィスからも、さらに私企業、連邦機関からも参加してもらっています。HEC の責任範囲としては、デザインして、開発して、配布して、それを管轄していく役割を担っています。

開発チームはまず主導的な立場のエンジニア、これは HEC の人間ですけれども、チームには1人か2人の HEC のアシスタントが入ります。さらには、私企業の請負業者が契約ベースで入ります。ユーザーが広くテストをしてもらう、そういう社会ができ上がっています。請負契約については後ほど詳しく申し上げます。

学会、大学からはソフトウェアの活力、あるいは信憑性についての最新鋭のアドバイスをもらっています。そして、学会に頼んで我々が提案するものをそれぞれ独自の立場でレビューを行ってもらっています。

やはり大学となると現場に即した実際的な考え方よりも、将来志向で考えてしまうという傾向があります。ですから、大学と実際的な現場のオフィスとのギャップについて、その橋渡しをする役割を HEC が担っています。ですから、大学では HEC は余りにも実際的、実務的過ぎるのではないか、一方、現場のオフィスでは、HEC というのは余りにも理論的過ぎるのではないかというようなことを言われますが、そういう批判があることもいいことだと思います。

ユーザーは主には工兵隊内の担当者なのですが、そういう人に HEC に来てもらってチームに参加してもらいます。結果をいろいろ査定してもらって、フィードバックしてもらって、そしてソフトウェアの考え方、そしてクオリティ・コントロールの観点においてもこういうフィードバックを外部からしてもらうというのは非常にいいことだと思います。

知的所有権を持った、ソフトウェアの所有権を持ったベンダーについてはまた後で詳しく述べていきます。

私企業、さらには連邦機関のそれはどれぐらいのことができるかということにも関連してきます。技術的に違う見方、あるいは当社の製品の代替物となる製品も出してくれて、いろいろな見方、考え方を交換できます。GIS ソフトウェアの場合はこの知的所有権を使っています。HEC の中で不足する部分があれば、その製品で埋めていくようにしています。

## (13) Software Development Processes Continued

ソフトウェア開発プロセスで Waterfall 方式を採用しているのか、Spiral 方式を採用しているのかという質問がありました。

Waterfall 方式の場合は一気に開発してそれをテストして、テストされて実証されたら、また新たに開発してということ一本道に行っていくのが Waterfall 方式です。

もっと自由にできるのが Spiral 方式で、プロトタイプを作成し、結果をチェックして変更を加える、またテストして変更をする（フィードバックする）ということを自由にできるのが Spiral のやり方です。

理論的には Waterfall 方式でも Spiral 方式でもいいのですが、実際は Waterfall 方式と Spiral 方式、これを併用して作業しています。修正する場合、Waterfall だと 1 回完結して次に行くのですが、Spiral だと開発順序を遡ってもう一度作り直すことができます。

#### (14) 外注比率

HEC が直営でどれぐらいやって、外部にどれぐらい請負契約ベースでやっているかというお話です。業務範囲がいろいろ広範囲にわたっていますので、HEC の職員だけでは全部網羅しきれないので、外注が必要となっています。HEC のスタッフの能力を広めていく上でも外注（契約）は有効です。

別添のレポートの中には、どういう契約書を使っているかというのが、その実例が入っています。具体的には、競争入札をおこなっています。だれでも自由にプロポーザルを提出してもらうことが可能です。

まず、一般的な能力とそのコストがどのぐらいかかるかということだけを考えてます。特殊な、特にこのソフトウェアを開発するのだということについては、最初の契約では触れていません。それぞれの業者はどのようなことができるか、それに対するコストがどれぐらいかということを最初に提案します。

その業者の能力とコストに基づいて、どの業者を使うかということを HEC が決めています。コストが低いかからといってそれで選ばれるとは限らないわけです。技術的な高さとコストとの兼ね合いで契約者を決めていきます。安ければいいというわけではありません。

#### (15) タスクオーダー

次に、契約者が決まったら、どういう仕事をしてもらうかというタスクオーダーを発行します。ソフトウェアのデザインをするのにどれぐらいのケーパビリティがあるかとか、実際どういう開発やメンテナンスをするのか、そういうことが契約のタスクオーダーに入っています。100 にも上るタスクオーダーがあり、通常、契約期間というのは 5 年ぐらいです。

協力的研究開発契約（Cooperative R&D Agreement）を結び、HEC がその契約者にお金を払います。HEC の資金です。

#### (16) ESRI との協定

工兵隊と ESRI との間では、正式な協定が結ばれています。ESRI は実用プログラムを作成する上で、HEC を支援しています。必要な機能を HEC で書面化し、ESRI で作成して貰い、HEC のパッケージにそれを盛り込んでいます。

実際に両者の間でアグリーメントを結ぶ上では、これはいいぞという利点が双方にないと上手くいきません。当社のソフトウェアについて、GIS を開発・利用できるということが HEC にとっては利点です。水理エンジニアリングについて、また新しい使い道があるので、そういう点で ESRI にとっては利点なわけです。また、HEC は広い範囲にわたるユーザーがいますので、こういった ESRI のユーザーの情報も使えるということは、HEC にとっては非常に有利な点です。

## (17) ライブラリの標準化

開発していく上ではコードとかライブラリとかデータをできるだけ標準化していこうという努力をしました。開発者にとってもユーザーにとっても、この標準化は非常に利点がありました。

とはいえ、開発言語については FORTRAN、Java、(今ちょっと Java に傾いていますけれども、)などを

完全に標準化してしまったわけではありません。HMS では FORTRAN のライブラリがあつて、水文のプロセスごとにそれぞれ独立したプログラムになっています。グラフィック・ユーザー・インターフェースのプログラムはすべて Java を使用しています。Java から FORTRAN ライブラリに呼び出して、そして詳細な水文の計算をするようにという指令を出しています。

グラフィックについてのライブラリ、これはもうさっきお話ししました。

インプット、アウトプットのプロシージャについてのライブラリもさっきお話ししました。

データの管理、データの保存のについてのライブラリ、これも 1 つの大きなライブラリです。

## (18) 外国語インターフェース

HMS の最新の開発の中では外国語とのインターフェースというのがあります。ここでは、外国語とのインターフェースについてお話をしていきます。

Java システムは外国語を扱う上で大きな利点があります。HEC では「Java ソフトウェア開発者キット (SDK)」を使っています。誰でも容易に手に入れることができます。Java の開発者は、「ただ開発するだけではなく、ユーザーがどの外国語でも使えるということが大事だ」ということを理解していました。

Unicode というものを Java で使いまして、(90 万の文字を使っているというふうに私は聞いているのですけれども、) どの言葉でも、サンスクリット語でも日本語の漢字でもどれでも使える、と聞いています。実際に今世界で使われているどの言語もこれに入っています。この機能は「Resource Bundle」(リソースの束)と呼ばれています。テーブルを作成し、左には英語で記述し、イコールの右側は他の国の言語で記述します。ソースを直接書き換えなくてもこのテーブルを書き換えることで翻訳が可能なのです。要素名をここに書きなさいというふうにプログラミングして、スクリーンにどういうものが出てくるかという、リストの右側に出てくるものがその指定した場所に出てきます。まだテスト済みではないのですけれども、理論的にはロシア語でも日本語でもこれは対応可能です。イタリアなど数カ国から翻訳してほしいというような要望も HEC に来ています。

ここで、Tool Tip (「TT」と呼んでいます)を紹介します。カーソルを使うと、アイテムのところにカーソルを置くと、風船(バルーン)みたいなものが出て説明がポッと出ます。これを利用すれば、翻訳や注釈を入れることが可能です。スクリーン上のアイテムだけではなく、TT のところにカーソルを置くとバルーンの記述が出てきます。

HMS ではこの Resource Bundle が 3 つか 4 つ使われています。ユーザーは右側を翻訳して、自分の好きな言語に翻訳することができます。HEC が使っているベンダーも非常にこれに興味を持っています。世界でこれを評価してもらえるといいな私は思っています。ソースコードを配布する必要がないというのがこれで非常に重要な点です。

HMS プログラムは、Java の Resource Bundle という形でエグゼキュタブル・コードがまとめられています。ユーザーは HomePage にアクセスして、すぐにプログラムを使うことができます。

ます。

#### (19) EXE ファイルのみの配布・ソースファイルを配布しない理由

HEC ではエクス可タブル・コードだけで、なかなかソースコードを配布しないのはなぜかというような質問があります。HEC のウェブサイトにもそれは書いてありますが、2つの方法をとっています。

古いソフトウェアですでに成熟して (バグ取りが大方終わっている)、もう変更がない (バージョンアップの予定もない) 分についてはソースコードを配布するというのが1つの考え方です。

新しいプログラムで常に新しい変更が行われているプログラムについては、エクス可タブル・コードしか配布したくないというのが基本的な考え方です。なぜなら、自分たちの問題でもう手いっぱいなのに、ユーザーさんがソフト・コードを手に入れて、「あ、こんな問題が起こってしまった」と言われても、なかなかそこまで手が回らないというのが現実だからです。

#### (20) Quality Control パブリックドメイン βテスト

最後に、ソフトウェア開発工程の中で述べたいことは、クオリティ・コントロール、QC の問題です。さっき御質問でもありましたが、ソフトウェアがこれで大丈夫だということをユーザーに確信してもらうことが重要です。HEC の中でも広範囲にわたるテストは行います。標準テストと呼ばれる一連のテストがあって、開発されたソフトウェアについては全てそのテストを行います。それぞれの計算法、アルゴリズムのテストであったり、マスキングのテストをしたりもします。アプリケーション全体のテストを行う場合もあります。

最後のプログラムはいろいろな細かい、微に入り細をうがったたくさんのテストを経ています。プログラムを配布していく前に HEC 内部で行っていくテストとして BETA テストというのがあります。実際にユーザーがソフトウェアをテストする BETA テスティングをおこないます。β版はいろいろな入手方法があるのですけれども、RAS についてはウェブサイトアクセスしてもらえばだれでもそれをテストすることが可能です。工兵隊のオフィスだけではなく、世界中のエンジニアの方にアクセスしてもらってテストしてもらうことが可能です。これは HEC にとって非常に重要なテストです。

プログラムのテストは、これという特定の人に頼みたい場合があります。こんな時でも、頻繁に使ってくれるいいユーザーはだれかということがわかっていれば、いざというときにそのユーザーに頼むということもでき、重宝します。

ダム決壊に興味のある人には RAS とか HMS のプログラムを出して、ダム決壊の結果を査定してもらうということも可能です。

プログラムをパブリックドメインにおくということによって、ユーザーに幅広くテストしてもらって、そのテストの結果をまた教えてもらうという形での公開は、開発側として非常に有用なのです。HEC ではこのような理由でパブリックドメインに公開をしています。

最終版としてリリースしていく前にチェック・リストがありまして、いろいろな項目がこのチェック・リストには盛り込まれています。チェック・リストにはテストとかコードがどういふところで使われたかとか、どこにあるかとかいろいろな情報が網羅されています。

ユーザーからのフィードバックというのは非常に重要です。ソフトウェア・サポートのところ

もお話をしますが、ソフトウェアは無償配布ですが、エラーがあったらどんどん HEC に直接言っ  
てくださいというアナウンスをしています。もしプログラム・エラーを発見したら、制限なしに  
HEC にコンタクトしてもらえようになっています。

ただ、工兵隊以外の方がプログラムを使う上で支援をしてほしいと言われても、それには応じら  
れません。この点については後でソフトウェア・サポートの中で触れます。

ISO9001 では、実際にコンサルティング・カンパニーでソフトウェアが使われる前に QC 的など  
ところをチェックして、実証していかなければいけないということになっています。取得には非常に  
広範囲にわたる手続が必要です。ですから、ソフトウェアのこういうテストをしたからきちんとし  
たものだということを、きちんとした証拠によって実証していく必要があります。

HEC では、この ISO を実際に適用しているというよりも、ISO の考え方を HEC で使っている  
と言えます。最終的には ISO のプロセスを実際には踏襲するというようにはなるとは思いますが、今  
は考え方を踏襲しているという段階です。

HMS の場合ですと、それぞれの計算法にテストのやり方があります。例えば、浸透を考える場  
合には、SES Infiltration Technique の基礎式を、MATHCAD というソフトウェアに入れて、アウ  
トプットを得ています。HMS の結果と MATHCAD よって計算された結果を比較して検証してい  
ます。QC の中でそれも非常に重要な要素を占めています。かなりの作業になりますけれども、降  
雨の計算とか Infiltration のメソッドとか、Unit Hydrograph、Kinematic Wave という方法があ  
るのですけれども、どの計算法を使うにしてもよいのですが、検証は必要と考えています。

川の流れを追跡するプロシージャは、MATHCAD 計算結果と比較検討することで検証できます。  
ISO のプロシージャの中にもそれらの検証は含まれています。

<Q： RAS の開発費用>

Q: HEC の開発に関する予算、経費について、HEC-RAS にまとめるときにどれぐらいかかったか。

A: HMS は R&D から 20 万ドルぐらい出してもらって使いましたが、この資金源とは別のところ  
からもお金が出ています。GIS サポートなどの場合は年間 10 万ドルぐらい。そのうちの 30%、3  
万ドルを HMS が使う。ほかの水管理システムの場合、また別のところから 5 万ドルとか 10 万ド  
ル、HMS がもらってくる。年予算というのは変わるのですけれども、年間、大体 30 万から 40 万  
ドルぐらい、それぞれ HMS と RAS に予算が使われています。予算が少ないときには 20 万ドルぐ  
らいしかもらえない。

Q: 別の言葉にしますと、96 年からということを見ると、8 年間ですから、それ掛ける 8 ぐら  
いの経費が今までにかかったと考えていいのですか。

A: 当然、開発というのはずっと続くものですから、その時々のものではないのだけれども、大体  
100 万ドルぐらいそれにかかっています。3~4 年にわたって年 20 万ドルから 30 万ドルぐらい使  
われて 100 万ドルぐらいになった計算です。HEC-1 との代替、取ってかわるということだけが目  
的ではなかったのですけれども……。

<Q： 予算の内訳>

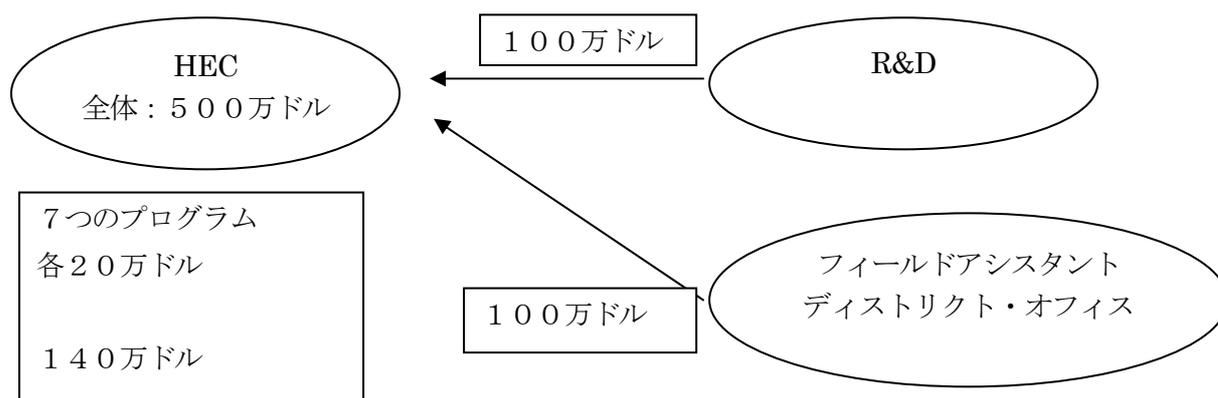
Q: 予算の内訳、外注費、人件費等について、100 万ドルの内訳は？

A: 3 人ぐらいが担当していました、リード・エンジニアというのはビル・グルーナーという人で、  
ニーズによってそれに 1 人とか 2 人加わったりした。

人的経費は1人 12 万ドルぐらいです。ただ、直接の給料だけではなくて、管理費、一切合切含めて1人 12 万ドルぐらい、間接経費を含めてかかっている。直接人件費は年間5万ドルから7万ドルぐらい、もともと HEC にいた人が外へ出てコンサルタントをやっているのですけれども、それぐらいの補助として支払ったということです。

<Q： 今後の開発予算>

Q：ネクストジェネレーションについて今、つくられているようなものがどれぐらい予算を組まれているか。



A：トータルで、R&D からのバジェットが 100 万ドルぐらい、それぞれの主なプログラムに 20 万ドルぐらいというのが1つの予算の見当で、年間で 20 万ドルぐらい1つのプログラムに考えています。水理・水文と貯水池と洪水被害軽減で、それぞれが 20 万ドルぐらい。いつもそれだけ予算がもらえるとは限らないのですけれども、R&D からなかなかお金をもらえないときはもっとフィールド・オフィスに行って、何とか資金援助をしてくださいというふうに頼みに行くこともあります。

HEC の中で R&D が占めている比率というのは 30%~40%ぐらいです。HEC トータルとしては年間 500 万ドルぐらいの予算です。それはトレーニング・プログラムとかソフトウェアのサポートとか、研究開発も含めてです。かなり予算のうちの大きな部分がフィールド・アシスタンス、いわゆるディストリクト・オフィスから出してもらっています。ということは、HEC はコンサルティング会社みたいな感じです。

予算のうちの 30~40%ぐらいが年度の最初にまず割り当てられます。残りの 60~70%は実際にそのプロジェクトが進んでいく中で、ディストリクト・オフィスからその予算年度の中でその都度お金を出してもらっています。30%ぐらいが会計年度の前に来て、その会計年度期間中はディストリクト・オフィスからそれぞれ徐々にもらうという形になっています。

<Q： メインテナンス費用>

Q：メインテナンスでどれぐらいかかるか

A：5年から 10 年ぐらい前にソフトウェアのメインテナンス業務というのがありました。これは、ヘッドクォーターからそれぞれのディストリクト・オフィスに税金が課されるみたいな形で、上納金みたいな形で割り振ったわけです。「嫌だ、税金などは払いたくない」と現場のオフィスは言うの

ですけれども、(笑声) やはり政府関係のユーザーというのはサービスを受けた人が払うべきだというように言うわけです。実際にサービスを受けたら、受けた人が払いなさいよ(受益者負担)というのが政府関係のユーザーの考え方です。

現在では、サポートしてもらったら、毎年、毎年のディストリクト・オフィスから寄附みたいな形で資金を拠出してもらっているのです。

HMS や RAS の 7 つのプログラムについて 1 つの地方オフィスについて年間 3000 ドルぐらいがサポートの予算です。

40 のフィールド・オフィスがあるので、 $3000 \times 40 = \$ 120,000$  / 1 本  
7 本のソフトですから  $120,000 \times 7 = \$ 840,000$  / 年 となります。

<Q: サポートの内容>

Q: サポートの内容は?

A: 電話のホットラインのサポートとか、ある程度限られてはいるのですけれども、コンサルティング業務とかそういった支援的な業務です。

<Q: GUI の比率>

Q: GUI のグラフィカル・ユーザー・インターフェースの開発というのは非常に困難だったというふうに言われていたのですけれども、今の割合として、それに注力するマンパワーは、プログラム本体と比べるとどのぐらいの割合を占めているのでしょうか?

A: RAS の場合はマイクロソフトを使っていたので比較的少なかった。HMS は C++ とか Java を使って開発したので大変でした。HMS については 1 人がもうほとんど専任、張り付きで GUI の開発について取り組み、担当しています。

<Q: インターフェースの共通化>

Q: インターフェースを共有化するという意味合いで同一のコンセプトのもとで開発しているのか、それとも RAS は RAS、HMS は HMS で、別個、別個でインターフェースをつくっているのかという質問ですが。

A: Data Storage System (DSS) を扱っているチームが 1 つあります。データ保存システムです。この DSS というのは HEC のプログラムがそれぞれお互いにコミュニケーションするためにつくられたものです。時間と対になるデータ、時系列データのためにこのチームはつくられました。標準的なデータベースはこの時系列データには余り使い勝手がよくなかったので、DSS がつくられました。例えば DSS が雨水の流量データを受取り、川、あるいは貯水池の水理にこの流量の情報を使うというように使っています。

Q: もう一つ、流量を Data Storage に渡して、そこから別に計算に行くというお話が今ありました。その流量は Data Storage に渡すということだけを考えて渡しているのか、それとも次に何をしてほしいということも含めて渡しているのか、要は全体のコントロールをどうしているのかを聞きたいのですけれども。Data Storage System を使ったデータのやりとりのコントロールをどうしているのか。

A: DSS のデータは、HEC のソフトですべて読み込めます。それがスタンダードになっているのです。

アプリケーション間は DSS のフォーマットのファイル渡しでおこなっています。

また、付加データとして XML が使えないかというのは今検討中です。

GIS のデータ・フォーマットを決めたときは、RAS とか GIS インターフェース、いろいろなチーム、フィールド・オフィスやら政府機関やら ESRI、ベントリー、大学関係者、コンサルタントから HEC に来てもらって、GIS のデータを RAS と GIS とどういうふうに共有するか、そのフォーマットについて、検討しています。フォーマットができますと、今度は ESRI とかベントリーが自社で GIS のフォーマットをつくるという形をとっています。このフォーマットというのはもう公に出していますから、ユーザーでどんなプロシージャを使われているかというのはすぐわかると思います。

具体的な作業は、GIS の水理・水文についての専門家、ディビット・メイドメント教授にヒアリングを行ないました。この教授の博士号研究の 2 人の学生が HEC に来て仕事をしています。

逆に、Arc Hydro の開発については HEC のスタッフもいろいろ関与しています。

GIS については大学とのコネクションで技術を吸収しています。

## (21) XML について

2000 年の話ですけれども、アメリカのいろいろな機関で、これについて盛んな議論が行われていました。そしてアウトプット・レポートをつくるときに、この HMS の中で XML を使っています。テンプレートを当初、HEC から出して、そしてユーザーが自分のアウトプット・レポートをそのテンプレートに基づいて修正していくことができます。サブ・スタンダードのアウトプット・レポートというのはもちろん我々から提供しますが、そうすればユーザーで非常に柔軟性を持って使ってもらえます。連邦の機関ではデータというのは共有したいのですけれども、なかなか予算が取りにくいのが現状です。ですから、プロジェクトのひも付きにして、そういうことを進めていく、そして予算を取るという形で進めています。

Q: 今、XML が外部に出すときに使われていると言っていましたけれども、Data Storage System でやりとりをするときにはそれが使われているのか、使われていないとすれば何でやっているのか。

A: アウトプット・レポートを構成するときにだけ使っています。DSS でのやりとりは使っていません。DSS ではファイルで渡しています。

<Q: Data Storage System について>

Q: その DSS の中にももちろん時系列データもあるし、あるいは GIS データというものもあると思うのですが、その GIS データについてはもう Arc Hydro そのものになっているのでしょうか。

A: 空間的なデータ、空間的な配置というものを DSS の中でやっています。

<Q: 学会との関係>

Q: 先ほど開発したソフトの精度の管理の件でいろいろなテストのランドマーク等をお聞きしたのですけれども、アメリカの土木学会 (ASCE) とプログラム自体の精度をチェックする等、そういう関係は持たれているかどうか。

A: 精度管理は直接はやっていません。アメリカの土木学会は HEC が出したガイダンス・ドキュメントそういった書類やガイダンスをレビューする形で関係しています。確かにソフトウェアの

レビュー自体はもう本当に業界の人、あるいはユーザーがやります。

直接は関与していません。ドキュメントのレビューだけです。

(補足)

「Hydraulic Engineering Manual」というのがありますね。あれを ASCE に頼んでレビューしてもらっている。今の話では、ソフトウェアの精度管理は頼んでいないけれども、一般ユーザーがいろいろ言ってくれるから直せるということだと思いますね。

<Q： 著作権について>

Q： 日本では実際にプログラム・コードを書いた人は著作権者人格権を持ち、その人の許可なしにコードを書き換えることはできないということになります。そうすると、アメリカでもそういう同じような著作権があるのか。もしあったらそれはソフトウェアを後で違う人が書き換えるときには、書き換えていいというような契約をしているのかどうか。

A： 著作権の使用についてはいろいろ調べました。法的ないろいろな難しい点があります。まずソフトウェアを仕上げてしまって、コピーライトを確立して、一たん完成してしまったものはコピーライトが確立します。何かまた変更したときはまた別の著作権が必要になってきます。開発途中でいろいろ変更があるとコピーライトについても非常にややこしい点があります。

ソースコードなりエクスキュータブル・コードを変更しようと思ったら、これは工兵隊からの変更ですよというように証明しないと問題になります。

ユーザーがソースコードをいろいろ変更したときに対応した経験がありますが、どれだけ大変な思いをして対応したかというのは、本当にもう口では言えないぐらいのものでした。メイン・フレーム・コンピュータを使っていた時代にソースコードを一般に出していたのですけれども、プログラムの中身についてのサポートをしていたことになり、予想以上に大変でした。

#### 4.2.4 Software support

##### (1) Software support

ソフトウェアがちゃんと生命を持って生き生きと活動しているための2つの重要な点があります。これといったメンテナンス法が確立しているわけではなく、常に変化していますから非常に取り扱いが大変です。

まず制度上のサポートが必要です。工兵隊、さらには HEC からの制度上のサポートが重要です。これは大事な製品ですから、ちゃんと人や資金などを確保して、継続的なサポートをしていかなければいけないというふうに制度的に決めてサポートしていかなければいけません。

資金は常に変わりますけれども、サポート自体は常に継続的に行っていかなければいけないのです。

制度的なサポートが継続的に得られた次は、技術的な能力の高さが必要になってきます。ソフトウェア開発におけるあらゆる領域で、高い技術力を保持していかなければいけないということが HEC の理念です。やはり DHI とか USGS といったどんなソフトウェアの開発者に聞いても、この2つの問題がお金を投資していく上で、ソフトウェアのサポート上非常に重要なことと、同じことをほかの開発者も言っています。

ソフトウェアの配布と、さらにドキュメントを出していくということもまた別の意味で重要な点です。Executables source codeの話は先ほどしましたが、書類、文献を出していくことも重要です。昔、緑色をしたマニュアルが HEC にあったのですけれども、ユーザーズ・マニュアルとして非常に有名でした。あとはウェブサイトからダウンロードもしていただけます。

RAS の新しいプログラムが出ますと、1 ヶ月に 6,000~7,000 ぐらいのダウンロードのアクセスがあります、非常に人気のあるソフトウェアなので。工兵隊から、アメリカのコンサルタント会社、あるいは世界中からたくさんのウェブサイトへのアクセスがあります。これも1つのすばらしいソフトの配布の方法ではあると思います。当然、このウェブサイトを運営していく上では資金がかかります。

このウェブサイトではウェブマスターとか、あるいは HEC の RAS とか HMS を扱っているチームのリーダーにコンタクトしてもらえることができます。名前は公表できませんが、技術的な、指導的な立場にいる人たちともアクセスは可能です。

ユーザーサポートについては、工兵隊と工兵隊外部に対してはこのサポートの仕方について非常に大きな違いがあります。

この点についてはウェブサイトでも説明を載せています。

ソフトウェアを講入してもらっている工兵隊のオフィスには、直接的にサポートしています。工兵隊外部へのサポートは我々ベンダーという呼び方をしているのですけれども、そういった業者を通じてサポートを行っています。

例えば、エラーがあったよという報告があれば、それはもう制限なく受け付けています。エラーがあったら、もうどんどん HEC に言ってきてくださいというようにしています。これは、世界中で HEC のソフトウェアがテストされているということになるのです。

いつアップデートするのですか、ニューバージョンはいつ出るのですかという質問があったかと思うのですが、ディストリクト・オフィスからこれというニーズがあった場合にアップデートを行っています。そして、アップデートがディストリクト・オフィスからいろいろ一定数出てきて、さらにそれに対するバグを解決した場合に新しいバージョンが出されます。

## (2) Vendors of HEC Software

業界とのいろいろな双方向のやりとりをする中で、ベンダーの働きというのは非常に重要な働きをしています。HEC が一般の人との交流を持つ上での大きな役割を果たすのがベンダーです。HEC から一般にサポートがされなくなってしまうと、小さなソフト会社とかそういったところがサポートをするようになると思います。そして、工兵隊以外の方から HEC に例えば電話で問い合わせが来ると、我々はベンダーにそれをつなぎます。土木研究所もベンダーの1つになることは可能です。

ベンダーになるのに、特にこれという制限とか規制とかはありません。いろいろなサービスをベンダーが提供しているのですが、ソフトウェアとかドキュメンテーションを配布するだけだったり、あるいは電話でのホットライン・サポートをベンダーがする場合もあります。エンジニアリング的なサービスをしてもらえるというのも非常にいい点です。

また、このベンダーが新しい顧客を開拓してくれているのです。トレーニングコースのビデオテープなども配布していますし、トレーニングをしてくれるベンダーもいます。よいベンダーというのはソフトウェアを配布するだけではなくて、エンジニアに関する支援をしたり、あるいはトレー

ニングをしてくれるというのが本当の意味でいいベンダーだと思います。

さっきショートコースについてお話をしましたけれども、ASCEといった大学関係者・政府関係、あるいは民間企業もそういうショートコースを担当してくれています。

1プログラムのベンダーになるには、200ドル払ってもらっています。お金を払うことによって、安易にベンダーになりたいというのを防いでいます。200ドル払うからには、やはり真剣にベンダーになってもらうことを考えてもらうようにしています。

ベンダーを使うことによって、HECは工兵隊以外の人たちとも交流を幅広く持てるようになっています。ベンダーはHECから直接サポートを受けることができます。優秀なベンダーだったら質問のほとんどをベンダー自体でその答えを見つけることができます。どういうふうにソフトウェアを使ったらいいのかということがわからない場合は、ベンダーはHECに問い合わせてきます。

ベンダーは10、あるいは時に100ぐらいのユーザーがいて、その手助けをしています。皆さんからの御質問もあるので、HEC内部でも、どうやってそのベンダーをコントロールしていくのだという問題が1つあります。ベンダーがどういうことをしているか、ベンダーの活動についてのクオリティ・コントロールというのができたらいいのですけれども、いろいろな難しい法律的な問題も絡んできていますので、公的な意味でのベンダーのクオリティ・コントロールは、実際はできていません。

ソフトウェアのサポート、あるいはベンダーを使うことについて何か御質問はありますか。

<Q： ベンダーのクライアント>

Q： ベンダーのクライアントというのはどういう人なのでしょうか。

A： コンサルティング会社であることもあるし、個人の場合もあるし、別のエンジニアリング会社であることもあります。例えば、州政府がベンダーになるということもあります。HECのソフトウェアをイリノイの水資源局ではよく使ってくれていますから、イリノイ州の水資源局が手助けする部分もあります。連邦政府でもベンダーとは言えないのですけれども、同じような役割を果たしています。

National Resource Conservation Service（自然保護局）、Soil Conservation Service（土壤保全局）と同じようなやり方で、こういった組織がHECに直接資金を出してくれています。ディストリクト・オフィスが出しているように。

その政府機関に5人から10人ぐらいのエンジニアがいて、彼らはその管轄のオフィスをサポートしている。1つのグループにHECのサポートを集中すれば、このグループからまたそれぞれの管轄のオフィスをサポートするというやり方をしています。

<Q： ユーザーのニーズを把握する際のベンダーの役割>

Q： ユーザーのニーズを把握するときに、ベンダーというのは有効に機能していますか、それはディストリクト・オフィスにテクニカル・サポートをするのと比べて有効に機能していると思いますか。

A： いいエンジニアリング会社は素晴らしいサービスをし、中には最低限のサービスしかしないところもあるし、やはりいいところからはちゃんとニーズをつかめるし、ベンダーの質にもよることです。HECの職員よりも優秀なベンダーもいるそうです。

ベンダーはサポートしている相手に様々なエンジニアリング・サービスを提供することで、コストをもらっています。ユーザーとベンダー間での契約については、HEC はタッチしません。サービスの善し悪しによってどのくらいお金をもらえるかが決まってきます。

#### 4.2.5 Anatomy

##### (1) Anatomy of a Software Development Project

次のセクションは、ソフトウェア開発プロジェクトの実例についてです。

これは新しい **Reservoir System** のプロジェクトの開発です。ソフトウェアが完了した4～5年前にこのニーズをつかんでソフトウェアの開発に取りかかりました。新しい技術やデータが出てきていますから、昔のソフトウェアはどんどん古い、時代遅れになってしまいます。このようなニーズ応えるためには、5年ぐらい前に開発が始まっていないといけない。実際のソフトウェアが世に出されるのは5年後ということです。

HEC の内部に「チャンピオン」と言われるような人がいて、工兵隊内部でのソフトウェアの開発の推進力となっています。一つのソフトを開発するのに大体4年ぐらいの期間が要ります。技術がどういうものかを見極めるだけではなしに、実際にかかる資金を確保することもこのチャンピオンの大きな役割です。

資金が集まった、そうしたら今度開発に取りかかっていると、次はどのようなニーズがあるかを突き止めていかなければいけないわけです。この **Reservoir System** の場合は、現場のフィールド・オフィスの人にどういうことが要求されるかという **Requirement Document**、要求事項というものをつくってもらいました。

ある程度 **R&D** からもお金が出ますけれども、実際はフィールド・オフィスからかなりお金が出ています。開発の実作業は、3年ぐらいかかります。その内訳は、デザインが1年でコードデベロップが1年です。さらに、リリースの前に、いろいろとテストしたり、書類化したり、あるいは **BETA** テストというのを積極的に行います。

次はメンテナンスで、これは限りがありません。大がかりなソフトになりますと、開発して、さらに改良、さらに新しいバージョンを出すとなると10年以上の年月を要していると予想しています。

その10年の間にまた新しいコンピュータ・サイエンスが出てきて、またさらに改善しなければいけないというケースも有ります。

##### (2) HEC Res Sim, Reservoir Analysis Version 1.0 Release Fall 2002

「Res Sim パッケージ」と呼んでいますが、開発に当たって、いろいろなニーズをフィールド・オフィスの人と相談して確認していきました。

前のソフトウェアではこういったことはできませんでした。Corps Water Management System、工兵隊水管理システムの中で作動することだけを考えていましたから。CWMS については後でちょっとお話をしますが、どういうことが必要かをこうやって長々とリストにしてあります。

これは今、一番新しいHECのモデルです。最新の機能（GUIであり、コードであり、グラフィクス）を備えています。

### (3) Anatomy of HEC-Res Sim Development

まず、HEC-5 に取って代わるものが必要でした。新しい運営ルール、オペレーション・ルールというものも出てきましたし、新しいインターフェースとか、マルチ・プラットフォーム、ケーパビリティに対応していかなければいけないというニーズがあったのです。それが1996年あたりです。1997年にサポートを始めて、大体資金も97年ぐらいに調達しました。97年、98年に細かいニーズを見ていって、要求事項を文書化していきました。

コードが1998年から2001年にかけて開発されました。2001年から2002年にかけてソフトウェアのテスト、文書化、さらにはBETAテストといったことが行われました。

まず、工兵隊のオフィスに2002年に配布を始めました。工兵隊からのフィードバックに基づいていろいろ改定してから、実際に一般用に公開していきました。プロジェクト・エンジニアとかHECの人とか、さらにはITの専門家更に、2、3人いつも学生も開発に参加しました。開発の多くの部分は外注を使いました。このソフトはマルチ・プラットフォームだけではなくて、クライアント・サーバー・テクノロジーも応用されています。その意味では、非常にハイテクな技術を持った外注が必要だったわけです。

コストは100万ドルほどかかりました。作業分担はHECと外注で半々ぐらいです。

Res Simの開発について、どういうタイムラインで行われたかを今ざっと見ていただきました。

### (4) The Real World Today

次に、どういう問題が今あるかというのを見ていきます。

Waterfallのように直線的に工程がつながっていく手法で開発をすることは、理想ではあるのです。しかし技術がどんどん変化していきますから、なかなか実際はそうはいかないようです。開発の上でそのような技術の変化が一番大きな問題でした。オペレーティングシステムとか、言語が変わっています。ソフトウェア開発の環境自体が変わってしまうわけです。直線的に行けばいいのですが、周期的にグルグル、開発した、試してみた、だめだったらまた再開発した、試してみた、まただめだったというふうにグルグル回る、スパイラル方式での開発になっています。

開発の中でどうしても問題が出てくるのはつきものですので、やはりバグというのもあります。

もうこれで開発は終わり、そして今ソフトウェアを世に出していくという決断をするのは、マネジメントの責任です。

資金が尽きてしまったからもうストップという場合もあるでしょうし、(笑声)時間が随分開発にかかってしまって、ユーザーから早く出してくれということで、もうここで開発終了という場合も考えられます。

2、3年前の中央大学での講演で、DHIともお話をしたのですがけれども、昔はソフトウェアの開発というのは8割方がテクニカルなもので、20%ぐらいがアウトプットとかディスプレイの問題でした。ところが、この比率が今ではもう逆転しています。いろいろなグラフィックスとかデータベースの技術が要求されてきていますから。水文のテクニカルな部分の要求されるものは15~20%ぐらい、コンピュータ・サイエンスに関する部分がもう80~58%を占めています。

コンピュータ能力が上がっていますけれども、それに伴ってソフトウェアへ要求されるものもどんどん大きくなってきています。

#### 4.2.6 Integrated system

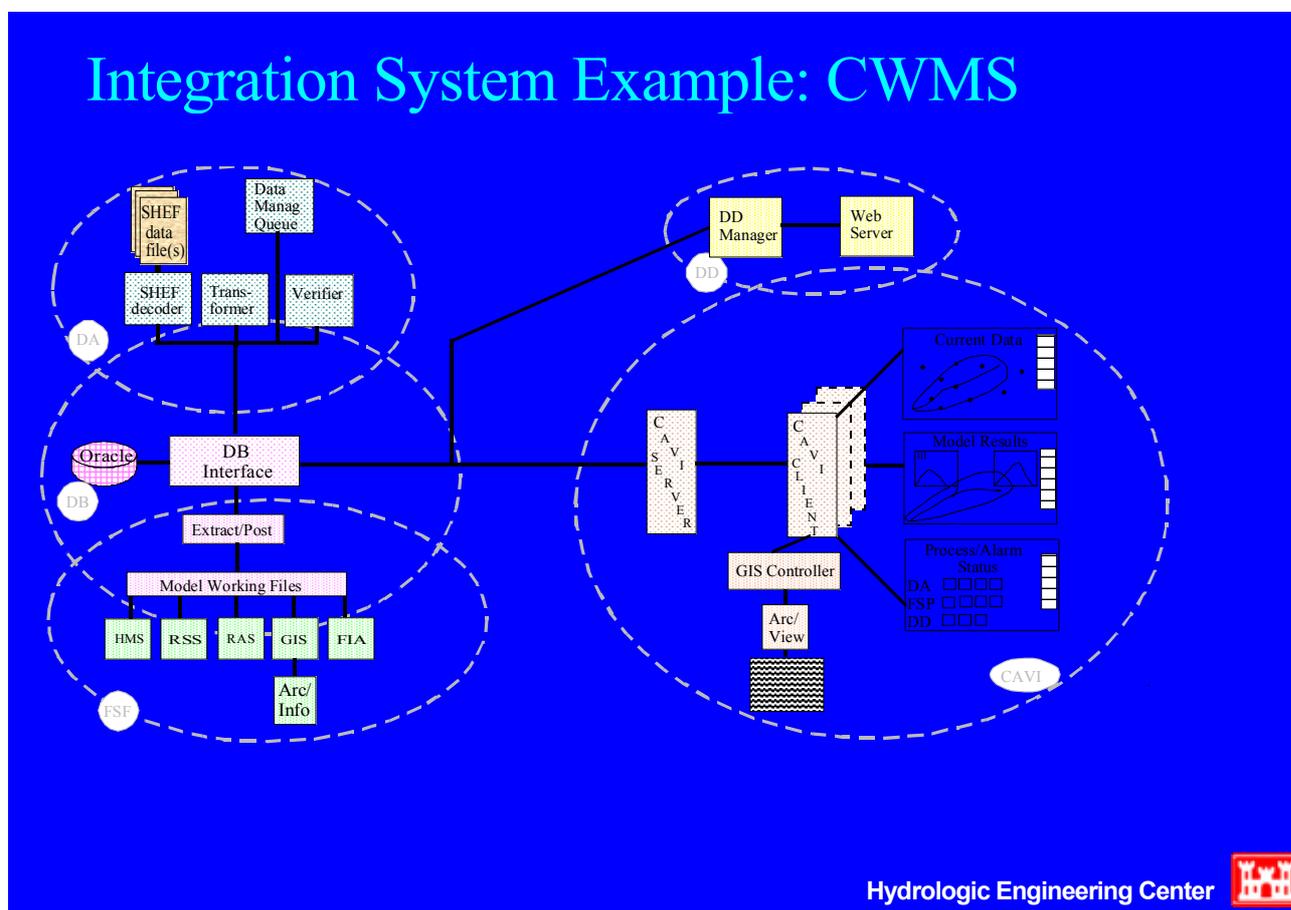
##### (1) Issue of the Day/Near Future: 'System Integration'

水文ソフトウェアを開発していく上で、次に考えていかなければいけないのはシステム統合です。継ぎ目のない環境の中でプログラムとデータが相互にコミュニケーションをできるようにしていかなければいけないわけです。

パズルみたいですが、1つの例としてここに挙げました。データ収集があって、データベースのところへ行って、データベースに入れて、データを視覚画面で見られるようにして、モデリング、あるいは意思決定を行って、そして出てきた結果としての情報を一般に公開するということです。

HECの最初のころの開発については、Corps Water Management Systemでした。ウェブサイトをのぞいていただければ、CWMSについてはいろいろなことが記載されています。

##### (2) Integration System Example : CWMS



Corps Water Management System は1つの例です。CWMSの目的は、貯水池を運用していく、あるいは洪水の際のコントロール、堤防の保全、貯水池の管理が目的です。

CWMSはまずデータ収集から始まっています。衛星通信によってデータを集めてきます。

Telephone landlines Communication、データをいろいろな方法でこのシステムに集めてきます。ナショナル・ウェザー・サービスと協力して SHEF (Standard Hydrologic Engineering Data

Format)、水文工学データ・フォーマットの協定が連邦機関の中でむすばれています。

降雨の観測、あるいは天気予報、USGS の水位観測というのも来ているかもしれませんが、流量観測もあるかもしれません。工兵隊では独自のゲージをいろいろ使っている場合もあると思います。

データの格納の際は Data Verification、クオリティ・コントロールをかけます。データは自動的に入ってきて、それが一定、安定しているかどうかをチェックしてデータベースに格納します。

データベースは、Oracle によって管理されています。ここから HEC の Data Storage System へと運ばれてきます。DSS からそれぞれのモデルにデータを渡していきます。

コントロールセンターは、頭脳に当たるのですが、ここでプロセスを管理しています。入ってくるデータをオペレーターが見て、データベースの中のデータも見て、どのプログラムに入力するかをオペレーターが決めて、入力データをそれぞれ割り振りしています。「Control and Visualization Interface」と呼んでいます。

こういう情報がインプットされると、今度は意思決定が行われます。これが継続的にバックグラウンドの中で行われるのが理想ですけれども、水理水文、あるいは Reservoir のシミュレーションは継続的に行われていきます。

問題がなければエンジニアは、ただモニターしていればいいのです。例えば洪水が起こると、オペレーターはそれぞれのプロセスの細かい点をやりくりして調整していかなければいけません。

今、例えば嵐が来ている場合、あるいは洪水になった場合の情報ははっきりと正確に出すためにいろいろシミュレーションしたり、キャリブレーションしたりということがオペレーターとしては必要になってきます。

この集まった情報に基づいてどういうふうプロジェクトを運営していくか、あるいはその資金、リソースを使っていくかということに対する意思決定が行われます。

あとはデータを一般に広めていく、データの公開です。ウェブサーバーを利用しています。

これは工兵隊内部で行われていますが、実際には一般にも公開することも可能ではありません。

これが 1996 年以後の HEC の主なプロジェクトです。大きなプロジェクトの 1 つです。これによって新しい水文・水理のテクニックが開発されたというわけではありません。この分析をする中では、先ほどからお話ししている HMS とか RSS とか、RAS とか、GIS とかいったソフトウェアを使っているだけなのです。

理想を言いますと、貯水池を管理している人は水路の通水能力ではなくて、実際にこれから起こりそうな洪水の状況に応じて意思決定をできるということが理想ではあります。「Flood Impact」と呼んでいますけれども、洪水がその地域に与える影響です。環境的なものとか、そういうことを突き止めるということです。

新しいプロジェクトとしてはこのアイデアを取り上げて、プロジェクトを計画する上でその考えを使っていくということが新しい動きとしてあります。「Watershed Analysis Package」と呼んでいます。

### (3) CWMS Decision - support Modeling

これは、マネジメントの側からデータを見たもので、「Model Working Files」と呼んでいます。これは Data Storage System の活用方法の一つです。このシステムは、ナショナル・ウェザー・サービスの天気予報のプロセッサにデータを出していきます。さらに、水文予測前処理にデータを出

していきます。また、Hydrologic Modeling System でシミュレーションをするための気象的あるいは水文的なデータ情報を含みます。また、貯水池予測システムへの情報として活用しています。

#### (4) CWMS, Water Management Version 1.0 Deployed 2002, Now 1.1

これは軍のオートメーテッド・インフォメーション・システムとして考えていただけるといいと思います。これはリアルタイムで、年中無休でやっています。

水資源のネットワーク上で意思決定をしていく上で、手助けとなるようなシミュレーションをしてくれるシステムです。

ネットワーク・ベースのクライアント・サーバー・システムです。非常にお金がかかっています。コンピュータ・サイエンスとしても、サーバー・システムはかなり複雑なシステムかと思います。97年に始まったこの開発されたシステムは、今いろいろな工兵隊の事務所で使われています。

<Q: CWMS について>

Q: CWMS は、事務所ごとに動いているのですか。そうだとすると、どのぐらいの事務所でこれを使っているのか、全国でもう共通して工兵隊の中では使っているということでしょうか。例えば、フィールド・オフィスが 40 数カ所とありましたけれども、すべてでもう使っているのでしょうか。

A: イエス。すべての Reservoir について実施されているわけではないのですけれども、それぞれのフィールド・オフィスで、少なくとも 1 つのプロジェクトについて実施されています。5年で総計 700 万ドルぐらい使われています。データの収集管理、さらに処理にほとんどその予算が使われました。

<Q: 外注とのコミュニケーション>

Q: 私も外注を使って非常に簡単なソフトウェアをつくったことがあります、大失敗しました。それは、こちらの要望が全然理解されないからです。契約者はコンピュータ・プログラマーでして、例えばいろいろな色を選べるようにするとか、字の大きさを自在に変えられるとか、我々がそんな重視していないところにたくさん時間をかけてそこを充実して、肝心の、例えば早く計算してほしいとか、そういうのが全然できませんでした。そういうコミュニケーションが全然うまくいかなかったのですが、そういうような経験はありますでしょうか。

A: それはよく起こることです。それに近い経験はあって、そこまでひどくはなかったみたいです。契約書の要求文言の中に HEC と緊密に連絡をとってやってくれということが入っていますので、必要ならば毎日 HEC のスタッフとコンストラクターとが協働できるようにということを契約に明記しています。チェックポイントが規定されていまして、ある 1 つが完了するとその時点でレビューして、これでいいから次へ行っていいよ、あるいはこれはだめだからもう一回元へ戻せということをチェックする規定が契約書の中に明記されています。HEC に契約としてはこういう能力がありますよということを出してもらって、そして HEC では 10 日かけてその契約者のできること、能力を査定します。

製品が完成したらチェックをおこない、上手くできていない場合、10 日から 15 日かけてきちんと HEC からの要求に基づいたものにするために修正する、要するに、要求どおりに上がっていなかったら 10 日から 15 日以内に HEC の要求どおり修正してくださいよということが契約の中に規

定されています。すべての契約の中にはチェックポイントが明記されていますから、契約者との作業はきちんとチェックされるようになっていきます。

Q： ソフトウェアの開発のやり方で、スパイラルで開発した場合、契約者にじゃあプログラムをつくってもらったが、ちゃんと契約書どおりにはつくったのだけれども、実際は使えるソフトではないというようなことがあったら、その契約は全くむだになるということがあると思うのですけれども、そういうようなことも承知で契約に出すということはあるのでしょうか。

A： ちゃんとソフトウェア、RAS であれ、何かのソフトウェアの中できちんと作動するということを実証することを契約者への要求項目の中できちんと明記することにはしています。それで結果をレビューして、必要ならば追加事項を入れます。やはり要求項目の中にきちんとそういうものを入れておけばそういうケースは起こらないということです。

### (5) Public Domain

Public Domain についてはいろいろ先ほどからお話をしましたが、連邦政府のお金が出ていますので、当然、それは一般に公開されるべきだというのが基本的な考え方です。例えば、知的所有権などという問題に突き当たった場合、大学を巻き込んでやっていかなければいけない、大学が知的所有権を所有している場合は、工兵隊にはこれは出してほしくないというようなケースもあります。

ソフトウェアの権利がきちんと確立されないと、その開発には HEC でも随分時間がかかってしまいます。非常に難しい問題、質問ではあるのですけれども、ソフトウェアが広く一般に使われれば、その開発においてもより改善点が出てくる、避けられないということを HEC は確信しているのです。

### (6) Challenges

我々にはいろいろな課題がありますが、一つはパブリックドメインでの開発です。これはいろいろ逼迫した状態になっています。例えば DHI のソフトウェア開発などで、民間でソフトウェアを開発するようにと推奨する動きがあるのです。ちょっと変な言葉かもしれないのですけれども欧州化ですね。デンマーク政府などでも、民間機関で開発することを奨励しています。実際にお願ひしたときには、非常にいい作業をしてくれます。

議会からのサポートも減っているので余り予算もとれない、それだけソフトウェアをいいものにしていく上での資金が足りなくなっているという課題があります。

これはさっき言っていた制度上のサポート、あるいは制度の能力ということに関連しています。ですから、知的所有権ということに対して人々の認識が高まっているので、なかなかソフトウェアをただで配布するということが難しくなっています。

政府が補助金を出して、ヨーロッパのいろいろな機関が、どんどん攻勢をしかけてきています。

フロリダの「Everglades」と呼ばれる生態系復旧プログラムなどは非常に大きな予算を使っているので、ほかのプログラムに対する予算面での圧迫となっています。

やはり資金源の確保ということが非常に大きな課題となっています。

だれもがそのソフトは使いたいと考えるのですけれども、なかなかそれをサポートしてくれる資金が得にくいのが現状です。

## 4.2.7 Summary

### (1) Conclusions

次に、こういう課題に対して結論としてはどういうものかと言いますと、水文工学は当然引き続き進めていかなければいけないわけです。新しいニーズがいつも水文工学においては生まれています。ですから、HECとしてはそれぞれのプログラム、RASであるとか、HMSだとか、Reservoir System だとかいったそれぞれのプログラムのリソースを常に確保しておかなければいけないということです。コンピュータ・サイエンスがどんどん進化していますから、それに歩調を合わせていくということが非常に大変になっています。

どんなコンピュータ能力が必要かということのを頭の中で描くことは簡単なのですが、実際にそれを構築していくのは非常に難しいことです。制度的なサポートを得るためには、何か新しいことをまた見つけていかなければいけないと思います。

工兵隊内部では内部的なサポートシステム、工兵隊外部にはベンダーのサポートシステムも存在しています。これらが引き続き行われていけば、こういった課題に対する解決策となると期待しています。

ただ、ベンダーのシステムが余りにも大きくなり過ぎて、負担が大き過ぎるという危険性もあるかと思っています。

ソフトウェアを開発していく工程の中で、強い統括力を発揮するマネージャーが必要になってきます。コンピュータ・サイエンスということがその作業の中でかなり大きな位置を占めるようになってきました。

### (2) HEC Contacts/Information

HECにコンタクトしたい場合の情報をここに書いてあります。HECのウェブサイトもありますし、「software and report」という欄があって、そこにもアクセスしていただけます。今、どんな活動がされているか、あるいはニューズレターにもアクセスいただけます。トレーニングについての情報も得られます。工兵隊外部の方たちにもアシストが得られるようにベンダー紹介の欄もあります。

あとウェブマスター、つまりソフトウェアに関与しているグループにもコンタクトしていただけます。それぞれRASとかHMS、Reservoir System、それぞれのソフトウェアごとにグループにコンタクトしていただけます。

HECが水文工学においてどういう役割を果たしているか、そしてソフト開発がどのように行われているか、どういう問題があって、それに対してどういう対処をしているかということについて、これでいろいろとおわかりいただけたかと思っています。

ソフトウェアに対するニーズというのがありまして、できたら皆さんにベンダーになっていただいて、日本におけるそういうニーズを汲み上げていただいて、我々とコンタクトしていただけると非常にありがたいと思っています。

直接HECと関係をもつところは、陸軍でするのでなかなか難しいかと思いますが、個人的に活動しているHECの職員もいますし、コンサルタント会社などもありますので、接触はしていただけるとと思います。

<Q： 国費を投入したソフトを海外に配布することの問題>

Q： 資金源はアメリカのタックスペイヤーで、アメリカの国民に対してフリーということだと思うのですが、そういうものを海外に配ることというのは制限・問題はないのですか。

A： 業界で技術の必要性は増大しているので、「ノー」と言う人はいないと思います。ソフトというのは単なるツールであって、それを使い続けるためにはエンジニアリングのアシストが必要ですから、DHIの有償のソフトを買ったはいいけれども、後でメンテナンスとか、サポートとか、トレーニングを受けようと思ったら、またお金を出してそういうものを受けなければいけないですよ。ですから、ソフトだけがあっても実用にならないので、そのような問題は起こらないと思います。

<Q： ソースは公開しない理由>

Q： ソースは公開しないで、エグゼファイルしか出さないという話がありましたが、ベンダーに対してもソースファイルは公開していないのですか。

A： 当然、ソフトの開発中にはいろいろな変更がある可能性がありますから、どこに対してもソースコードは一切出さないことになっています。コンピュータ・サイエンスが発達してくると、ソースコード自体もどんどん、どんどん複雑化してきているので、ベンダーに対してもソースコードは出しません。

ベンダーとして修正してソフトウェアの付加価値を高めたい場合はあります。

Java ではソース無しでも、ユーザーが水文のアルゴリズム、計算法を開発して、日本の **Infiltration Technique** を使って、それを **HMS** に使う。**HMS** にほかのライブラリにコンタクトして、そしてその計算法を見つけなさいというふうなインプットをすることもできる。ベンダー独自の方法を開発して、**HMS** がプロセッサ、データ処理のプロセッサとして機能できるようにするという方法もある。まあ、そういうやり方は問題を起こしているかもしれないけれども、そういうやり方もあるということです。

<Q： Java>

Q： Java を使っておられますが、それはもうずっと Java を使い続けられるということで間違いないのでしょうか。

A： そうです。将来的にも、新しい Java が間違いなくまたアップして……。

Q： Java がアップデートし、新しいバージョンが出たら、新しいバージョンの Java を使っていくのですか。

A： 使うつもりです。古いものとの互換性があればですが。(笑声) 古いシステムが変わるとちょっと問題が起きるものなのですが、でも将来的にも使い続ける意向です。Java の場合、非常にいい開発環境を持っていると思うので。

<Q： リバー・マネジメント>

Q： ちょっと観点の違う質問をさせていただきます。予算がだんだん厳しくなっているという話がありましたが、それは例えば、**Everglades** のようなところに持っていかれたり、民間に開発を委ねたというところがあるのはわかるのですが、リバー・マネジメントの重要性だとか、それからそれのためのこういうプログラム開発そのものの必要性というものの評価が落ちてきてい

るというようなことはないのでしょうか。

A： 公がどういうプロジェクトを重要視するかどうかは、これは時によって変わってしまうものです。環境面に余り注力しないときもありましたが、今は環境的な側面は非常に重要視されています。例えば環境分析をする場合、水理・水文のエンジニアリングの知識も要るし、あと RAMS とかデータ・マネジメントなども必要になっていますから、それなりに水理水文、あるいはソフトウェアを使った解析、分析というものが要求されてきます。

Q： 一般的に、例えば政府内でリバー・マネジメントに対する重要性の認識が下がってくる可能性がないかということについては。

A： 有ると思います。やはりああいいうスマトラ沖地震とか津波とかがあると、どうしても関心は地震予知の分析に行ってしまう。水文・水理、貯水池システム、それぞれに対して核となるマネジメントの意識をずっと高い領域に留めておくのはなかなか難しい。新しいニーズが出てくる、例えばティートンダムが決壊したという場合には、やはり新しいダム管理の手法が要求されてきます。ニーズが出てくると、それに対する新しいものも要求されてきます。

ただ、基本的な川に関する水文学がきちんと運用されていなかったら、例えばダムが決壊して、それに対するシミュレーションが必要になったときも、迅速に対応するということはできません。だから基礎的な部分を今からしっかりやっておかないと、何か起きたときにすぐ対応できないという問題はあると思います。

<Q：ベンダー契約費>

Q： ベンダーがいろいろ外部のユーザーサポートをやってくれる、そのときに 200 ドル払えばベンダーになれるというお話ですが、HEC の予算が非常に苦しいということでしたら、ベンダーのエンジニアリング・コンサルティングの売り上げに比例する料金を徴収するとか、そういう仕組みは考えなかったのでしょうか。

A： やはり連邦機関だから、普通のプロセスの中でなかなか外部からお金をいただくということに制限があるのです。その 200 ドルはアメリカの国庫に入ってしまうのです。HEC としては、お金はもらっていないそうです。なかなか生かさず、殺さずでやられているみたいです、連邦政府から。(笑声)

<Q：他の類似ソフトとの関係>

Q： 例えば Bureau of Reclamation のように、アメリカの中のほかの組織でも同じようなソフトウェアを開発していますけれども、それに対して何か意識をしているのでしょうか。例えば、競争相手と思っているのか、あるいはいつか重複しているのだからどちらかの予算を削れと言われるのではないかと、そういうような意識は何かあるのでしょうか。

A： Bureau of Reclamation だけではなくて、Soil Conservation Services、USGS とか、ほかのアメリカ政府機関にとってもやはり問題になることです。NASA などでもそうなのですが、水文モデルとかそういうもの……、やはりその 1 つ 1 つのものをまとめるようにして、リソース自体を浪費しないようにという動きがありますね。それぞれの開発に正当化された意図がありますが、水文学自体は同じですから、もっと共同して一緒になってやっていくということは可能だとは思っています。水資源のワークショップを開いて議論している例がありますけれども……。

政府機関の中で Bureau of Reclamation のように、同じような水文モデルの開発をやっていると

ころもあります。政府機関内でのいろいろなデータ管理とか、ほかの政府機関にもそれを使えるようにというように HEC はしているし、政府機関ごとのもっと交流というものを活発にしなければいけないと思います。Rio Grande の工兵隊のオフィスも Bureau of Reclamation のシステムを使っています。CWMS に Bureau of Reclamation のソフトウェアを統合しているのです。インターフェースにしておいて、CWMS から Bureau of Reclamation に行けるようにしています。天然資源のことについていろいろな政治家が討議しているらしいのですけれども、工兵隊とか Bureau of Reclamation とか EPA からのいろいろな機関がいろいろな水文を統合して1つにまとめてしまったらどうだという議会での論議があるそうです。それは政治的には難しいことだけれども。

<Q: HEM HM の実習用に HEC のモデルは使っているのか? >

Q: 工兵隊では「Hydraulic Engineering Manual」とか「Hydrology Manual」とか、非常に程度の高いすぐれたマニュアルを出版されていますね。それで、そういうマニュアルの内容を末端のエンジニアも理解し、使っていくというためには、実際の応用問題を解いて練習する必要もあると思いますが、そのときに HEC のこういうシミュレーションモデルとか、そういうものが役に立つのではないかと、あるいは役立つためにつくっているのかどうか、その辺のところをお尋ねしたいと思います。

A: 水理・水文分析の非常に基本的な考え方、原則がこのマニュアルには書いてあります。工兵隊のその担当のテクニカル・エキスパートがそのマニュアルを読むわけです。洪水防御とか洪水予測とか、その Reservoir System について HEC もそのマニュアルの中に記載しています。いろいろな問題の実際的な例とか、アプリケーションの例などについてもその書類の中では触れています。エンジニアリング規制についても書いてある書類もありまして、どのようにしてそのプロジェクトを始めていくかというようなことについても書いてあります。エンジニアリング・マニュアルには技術的な明細が書いてあるし、次はユーザーズ・マニュアル、あるいはアプリケーション・マニュアルがあります。アプリケーション・マニュアルと HEC マニュアルの間には関連性があります。直接的な結びつきはないにしても、それぞれ参照できる部分があります。

Q: それで、こういう HEC-RAS とか、そういうようなものがそういうエンジニアリング・マニュアルの考え方や知識を職員が、エンジニアが理解する、そういうツールになるのではないかと、するつもりはないのか、そういうつもりであるのかということなのですか。

A: アプリケーション・マニュアルからエンジニアリング・マニュアルに参照してもらって、研究するためにどういうことが必要かということはアプリケーション・マニュアルを見てもらえばわかります。そういう必要なことを満たしていくためにはどういうふうにして HMS が役に立つかというようなことを参照していただけるようになっています。

ですから、御指摘のことを、普通の公的なエンジニアリング・マニュアルに対しても HEC のモデルでつなげていけるようにできないかなというようなことを考えています。

Q: 私も個人的に考えているのですけれども、日本でも非常に多くの事業をやっておりますので、マニュアルは相当いろいろなものをつくっております。ただ、それを伝える手法としてこういうソフトウェアをつくって練習するということによってトレーニングするのがいいのではないかなと思ったものですから、そういう質問をしたわけです。

A: 昔のアプリケーション・マニュアルは工兵隊のプロシージャ、一般的なアプリケーションではなくて、実際に工兵隊のテクニカルな観点からの問題があって、工兵隊の実際の例とは結びつけ

られないものだったのです。新しいアプリケーション・マニュアルではプログラムを、どうやってエンジニアリング・マニュアルと参照しながらこのプログラムを使っていったらいいかということ  
を新しいアプリケーション・マニュアルには書いてあります。いろいろなテクニカルな調査をして  
いく上では、やはりソフトウェアは非常に有用なツールになっていくと考えています。

○司会 では、もう時間になりましたので、これで今日の講演会を終わりたいと思います。

大変長い時間、講演をしていただきまして、ありがとうございます。拍手をもってお礼をしたい  
と思います。(拍手)

どうもありがとうございました。

## 参考文献

- 1) 藤田光一：米国・欧州3カ国の河川・水理関係研究所の戦略，土木技術資料，第37巻11号，pp13-15,1995.
- 2) 江幡禎則，小川鶴蔵，松島哲弥，五十嵐孝浩，石井邦宙：水情報国土データ管理センターの現状と課題，河川情報シンポジウム資料,2005.

[www.river.or.jp/hppy/05/img/report\\_08.pdf](http://www.river.or.jp/hppy/05/img/report_08.pdf)

- 3) 高棹琢馬、椎葉充晴、市川温：構造的モデリングシステムを用いた流出シミュレーション，水工学論文集 第39巻, pp141-146, 1995.2
- 4) 高棹琢馬、椎葉充晴、堀智晴、立川康人、市川温：流出系の構造的モデル化システムについて，京都大学防災研究所年報 第38号 B-2, pp395-406, 1995.4

## 謝 辞

中央大学 山田正先生には、流域水物質循環モデル・ソフトウェア博覧会やシンポジウムの実施に向けて積極的にご先導下さると同時に、米国におけるモデルの開発体制に関する現地調査においても、調査団の代表として様々な御指導を頂きました。ここに深甚なる謝意を表します。

また、京都大学 椎葉充晴先生、市川温先生には OHyMoS に関する技術的検討について細部にわたって御指導下さいました。モデルフレームワークを 1990 年代始めから具体化を進めてきた先見の明に深く敬意を表するとともに、篤く御礼申し上げます。

流域水物質循環モデル・ソフトウェア博覧会やシンポジウムの実施にあたり、山田先生、椎葉先生その他、京都大学 立川康人先生、中央大学 平野廣和先生には、アクションプランの提案および啓発活動において多大なご協力を賜りました。また、中央大学 岡田将治先生および山田先生の研究室の学生諸氏の方々には博覧会の実施に向けた準備においてご尽力下さいました。ここに記して感謝申し上げます。

## 研究協力者

「はじめに」で述べたとおり、高い見識、高度な専門的知識を有する技術者、研究者、行政関係者の協力、貢献の上で本研究は進められてきた。以下に、上記の先生方と著者を除く本研究の遂行に関わったメンバーを示す（順不同、敬称略、肩書きは当時）。

藤山 秀章 河川局河川計画課河川情報対策室長  
氏家 清彦 河川局河川計画課河川情報対策室長  
佐藤 宏明 河川局河川計画課河川情報対策室長  
中尾 忠彦 財団法人河川情報センター理事  
小川 鶴蔵 財団法人河川情報センター審議役  
五十嵐孝浩 財団法人河川情報センター企画調整部副参事  
村田 和夫 社団法人日本建設コンサルタンツ協会河川計画専門委員長  
加藤健一郎 パシフィックコンサルタンツ(株)流域計画部  
河上 展久 パシフィックコンサルタンツ(株)流域計画部  
北村 忠紀 パシフィックコンサルタンツ(株)流域計画部  
藤原 直樹 (株)建設技術研究所水システム部  
荒木 千博 (株)建設技術研究所水システム部

## 添付 CD - ROM の内容

CD-ROM には、4 章に関連する資料を納めています。なお、本文の電子データ (PDF 形式) については国土技術政策総合研究所のホームページよりダウンロードして下さい。  
( <http://www.nilim.go.jp/lab/bcg/siryou/tnn/tnn0410.htm> )

フォルダ・ファイル名	内容
「4.2HEC ソフト講演録」	HEC ソフトウェアに関する講演に関連する資料 ( 質問事項、レポート、パワーポイント )

### 注意

本 CD-R の内容を転載、複写する場合は、必ず国土交通省国土技術政策総合研究所の許可を得てください。本 CD-R を使用する際に生じた、いかなる障害および損害に対しても、国土交通省国土技術政策総合研究所では一切の責任を負いませんので、あらかじめご了承ください。

Windows および Internet Explorer は米国 Microsoft Corporation の米国およびその他の国における登録商標です。

Acrobat および Acrobat Reader は、Adobe Systems Incorporated の登録商標です。

その他の会社名、製品名はそれぞれ各社の登録商標または商標です。

---

国土技術政策総合研究所  
TECHNICAL NOTE of N I L I M  
No . 410            July    2 0 0 7

編集・発行 ©国土技術政策総合研究所

---

本資料の転載・複写の問い合わせは  
〒305 0804 茨城県つくば市旭1番地  
企画部研究評価・推進課 TEL 029 - 864 - 2675