

(財)日本建設情報総合センター研究助成事業

建設分野へのRFID(電子タグ)と  
プロダクトモデルの適用 報告書

平成15年9月

室蘭工業大学工学部建設システム工学科

助教授 矢吹 信喜

## 1. はじめに

土木構造物を、長期間安全かつ経済的に供用・利用するためには、建設過程および完成後の点検や補修・更新に関する記録をきちんと残しておき、必要な時に直ぐに引き出せるようにする必要がある。これまでに各種構造物のデータベースが開発されたが、現場において効果的に運用されている例は少ない。それは、現場の特定の部材や機器のデータにアクセスするための、データと現場の部材そのものとの関連付ける有効なデバイスがほとんどないことが一因だと考えられる。製造業や流通では、バーコードやシリアル番号が利用されているが、建設分野では、パイプや鉄筋等の部材は、コンクリートや土の中に埋められるため、維持管理での利用を考えると不適である。

そこで、我々は RFID<sup>1)</sup> (Radio Frequency IDentification, 電子タグ, IC タグ) を部材に取り付け、ID 番号のみならず、施工過程や点検データ等を電子的に蓄え、必要な時に即座に現場において取り出せるシステムの開発を行っている。

一方、構造物のデータをきちんとデータベースとして一元的に管理するためには、標準化されたデータモデルであるプロダクトモデルを定義する必要がある。我々は、土木構造物のプロダクトモデルに関する研究も実施している。

そこで、本研究では、構造物のデータを一元的に管理するとともに、現場で必要なデータに容易にアクセスできるようにし、また、長期間にわたって、構造物や各部材の履歴を記録しながら、管理することを目指して、RFID とプロダクトモデルとを統合化したモデルを開発することとした。

## 2. RFID (電子タグ)

### (1) 電子タグ

データ等の情報を電子的に記憶させたボタン状のものや小型のカード等は、電子式データキャリアと呼ばれ、データの読取・書込み時に装置と機械的に接触して行うか、非接触的に行うかで、それぞれ「接触型」と「非接触型」に分類される。接触型の代表は IC カード (スマートカード) である。接触型の IC カードは、耐久性が高くない点とコンクリート等の中に埋め込むことができないことから、土木構造物の施工や維持管理には不向きだと考えられる。非接触型の電子式データキャリアは、一般に電子タグあるいは RFID システムと呼ばれ、リーダライタと電子タグとの間のデータ交換は電磁界を用いて行われる。電子タグは、データ容量が通常 240 Byte 程度と小さいが、耐久性が高く、コンクリートの中に埋め込むことが可能であることから、土木分野に向いていると考えられる。

電子タグシステムは、電子タグとリーダライタで構成されている。電子タグを点検個所に接着剤で貼り付け、リーダライタとモバイルコンピュータを用いてデータの読み書きを行う。図 - 1 に電子タグシステムの外観を示す。携行する携帯コンピュータとしては、本研究の現場点検支援情報システムでは、PDA (Personal Digital Assistants) をベースとしたものを提案している。以前から使用していたリーダライタは大型で重量も大きいため、本研究では、図 - 2 に示すように、リーダライタ機能を持った小型装置を PDA (オペレーティングシステムは PocketPC 2002) のコンパクトフラッシュ (CF) スロット (TYPE II) に直接挿入したものをを使用することとした。尚、本 CF カードと電子タグとの間の交信周波数は 125 kHz であり、伝送速度は 9600 bps である。

## (2) 電子タグの活用

電子タグの点検業務への活用については、ID 機能、トリガー機能、及び簡易データ保管機能等が考えられる。

- ID 機能は、電子タグに記憶されている ID 番号と読取機器内のインデックス機能により、電子タグが設置されている場所や関連付けられている部材や設備等を判別するのに使用される。これにより、GIS（地理情報システム）や 3 次元 CAD システム等と組合せて、作業ルート等の巡視ナビゲーションやデジタルカメラによる現場写真の管理業務効率化等が図れると考えられる。
- トリガー機能は、電子タグの ID 番号や格納データを読取ると同時に、計算、図化、音声注意喚起、作業手順プログラム等を起動させるのに使用される。
- 簡易データ保管機能は、前回の計測データや管理基準値等のデータや点検上のアドバイスなどの簡単なナレッジを記憶させるのに使用される。

本研究で開発した PDA による電子タグシステムをダム(point)の点検作業に適用させたスクリーンイメージを図 - 3 に示す。図 - 3 (a) はダムの揚圧力のデータを示しており、PDA を電子タグに近づけて、「タグ読込」をスタイラスペンでタップすると前回の揚圧力値が読込まれる。次に、現在の水圧計のデータを入力し、「揚圧計算」をタップすると現在の揚圧力が計算され、設計値と比較して判定される。最後に、PDA を電子タグに再度近づけて「タグ書込」をタップすれば、その日のデータがタグに記録される。図 - 3 (b) は、バルブの操作上の注意をアドバイスするものである。図 - 3 (c) は、コンクリートクラックの幅の推移が写真と共に記録されたものを表示している。



図 - 1 電子タグ及びリーダライタ



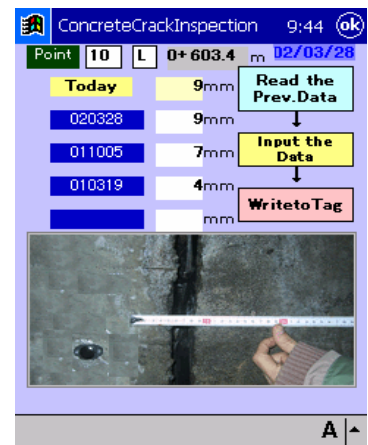
図 - 2 リーダライタ付き PDA



(a)



(b)



(c)

図 - 3 PDA 電子タグシステムのスクリーンイメージ

### (3) データ保管方式

電子タグシステムをオンサイト点検支援システムに活用する際、どのデータをどこに記憶させるか、という点について検討する必要がある。データの保持・取得方法としては、電子タグ内には ID 番号のみを記憶させ、その箇所の点検・計測データ等はネットワークを介して本部のサーバに集結させる「サーバ集中方式」、現場の電子タグに計測データや各種情報を記憶させ、リーダーライターにより読取る「分散メモリ方式」、及び電子タグには ID のみ記憶させ、データは携帯コンピュータに記憶させる「携帯コンピュータ方式」の3つに大別される。サーバ集中方式は、データの一元管理が可能となるが、非常時等にネットワークや通信が途絶えた場合、データにアクセスできなくなるのが欠点である。分散メモリ方式は、非常時でもデータにアクセスできるが、データ容量が小さく、現地へ行かないとデータにアクセスできないのが欠点である。携帯コンピュータ方式は、前記2方式の欠点を補うものであるが、複数の携帯コンピュータがある場合、データの一元管理ができず、逆に1台だけであれば安全性に問題がある。そこで、本研究では、以上3つの方式を組合せた「ハイブリッド方式」を採用することとした。

ハイブリッド方式では、電子タグ内に ID、前回計測データ、簡単なアドバイス等を格納し、携帯コンピュータにはトリガー機能により動作するプログラムや点検の際に参照する図面や仕様書、これまでの計測データやグラフ等を入れ、本部のサーバにおいて、全てのデータを一元管理するものである。この方式は、前記3つの方式の利点を兼ね備えており、安全性も確保される。但し、電子タグ内のデータと本部サーバのデータを一致させるためには、携帯コンピュータとネットワークによるデータ同期等の作業が必要である。

## 3 . RFID システムの既設ダム点検への適用

### (1) 点検支援システムの概要

点検支援システムは電子タグ、リーダーライター、PDA、PC で構成される。点検支援システムの構成を図 - 4 に示す。電子タグは 240bytes のメモリサイズを持つデータキャリアで、点検設備

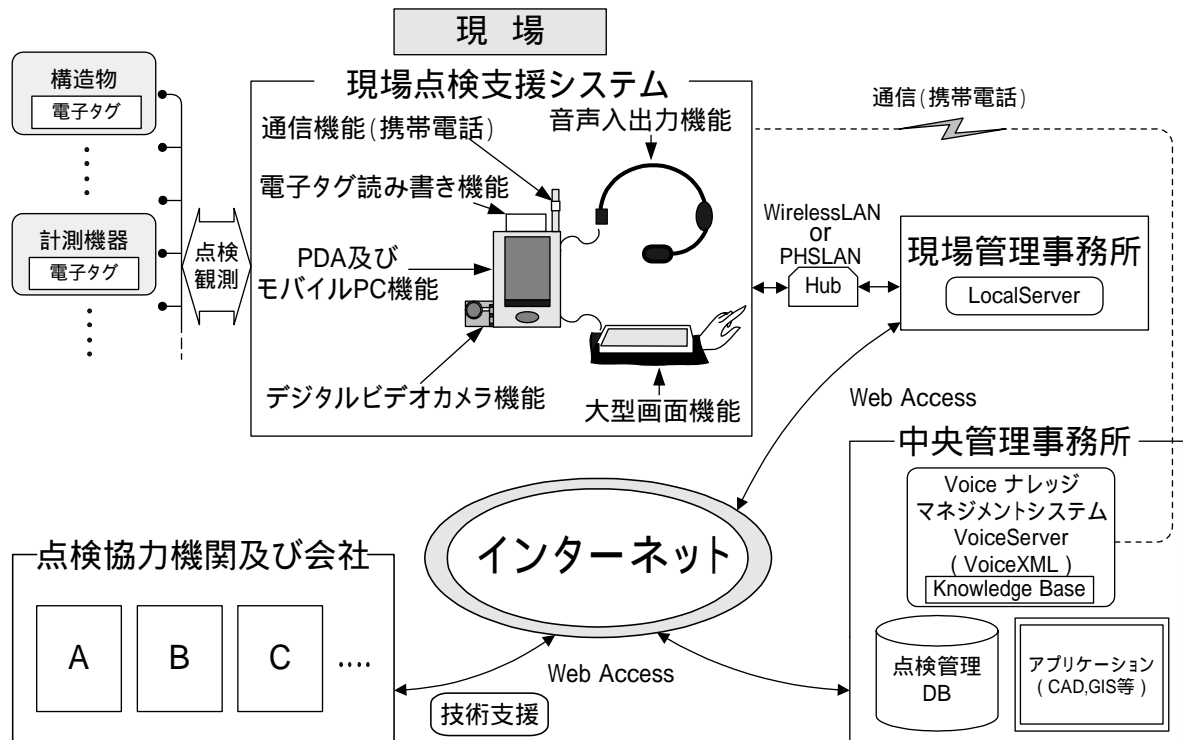


図 - 4 現場点検支援情報システム

に張り付けられる。リーダライタは電子タグの読み書きに使用され、PDA のコンパクトフラッシュスロットに装着される。現場に設置された電子タグには、ID と前回の点検結果が保存されており、点検員はリーダライタ付 PDA で電子タグを読み込むことで前回の点検結果と ID に関連付けられた申し送り事項、画像データを参照することができる。点検結果や申し送り事項を PDA にて入力し、電子タグに書き込む。こうして現場で入力された情報は、事務所に帰還してから PDA と PC を接続し、同期処理を行うことにより、確実に PC 内へデータベース化される。PC は点検結果のデータベースとして機能する他に、システム全体を管理するために用いられる。土木構造物の長いライフサイクルを考えた場合、点検対象設備（電子タグ設置設備）や点検項目が増減することは必至である。そこで、ユーザが自由に点検対象設備と電子タグを関連付ける機能や点検項目を設定できる機能を PC に加え、長期に亘る供用期間内における点検データ管理の一元性を持たせた。

## (2) RFID システムの適用

本研究で開発した点検情報支援システムを電源開発株式会社奥只見発電所の奥只見ダムの巡視点検へ適用した。奥只見ダムは、使用開始から 40 年以上が経過している堤長 480m、高さ 157 m の重力式コンクリートダムである。また、発電設備も点検対象に含まれることから、点検箇所が多岐にわたり、巡視点検に 2 日以上も要する。そのため、システムの開発に当っては、点検効率を低下させることのないよう、特に PDA への入力等の操作性に留意した。

本システムを利用した点検状況を図 - 5 に示す。PDA という小型端末を使用したため、サイズ、重量等の面からも携帯性に優れ点検現場での機動性を損なうことはなかった。ただし、用途を考慮して比較的大型の耐衝撃・防滴型 PDA を用いたため、点検員の意見としては、より軽く、小さくという意見が寄せられた。これについては、PDA ハードウェアに依存するものの、耐衝撃・防滴型以外の一般的小型軽量 PDA にストラップや防水カバー等を付加させることにより、軽量化、小型化を図ることができる。点検箇所に設置された電子タグを読み込んだ場合、PDA の画面上には図 - 6 に示すとおり、点検対象構造物、点検項目、前回の点検結果が表示される。電子タグをトリガーにして画面表示するため、階層構造の展開操作やスクロール操作が不要で、必要情報をオンサイトでタイムリーに得ることができる。点検結果の入力にあたっては、チェックボックスを採用し、入力作業の簡素化を図っている。入力操作は、画面の対象箇所をタップするだけで入力できるため、非常に簡便である。ただし、画面が小さいため、入力操作の簡便性と表示情報量の確保のためには、プルダウン形式によるものも考えている。

図 - 6 右上のメモをタップすると図 - 7 が表示され、点検で得られた気付きや次回点検への申し送り事項等を記入することができる。この機能により次回点検者への確実な申し送りが可能となる。このメモは、電子タグに関連付けられた情報として PDA に保存され、事務所における PC との同期作業により PC 内にデータベース化される。メモの入力作業は、PDA 付属機能を用いているため、入力し難く、誤入力も多い。今後は、音声によるメモ機能も付加する必要があると考えている。さらに、図 - 6 メモ左横の File をタップす



図 - 5 点検状況





図 - 6 基本操作

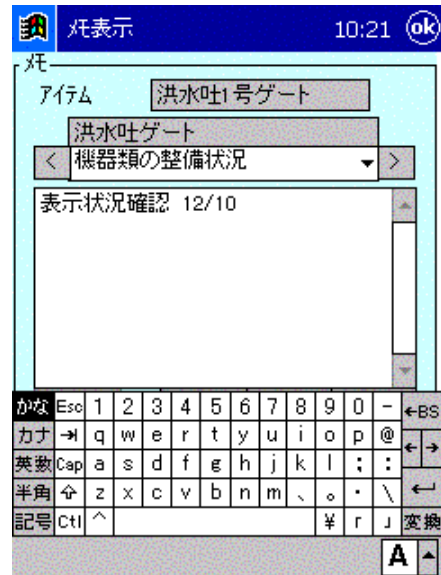


図 - 7 メモ入力画面

ると予め PC で電子タグと関連付けられた写真ファイルを参照することもできる．参照できる写真は PC に電子タグ毎にデータベース化されており，このデータを PDA 内に転送しておくことにより参照できる．これまで，写真整理の煩雑さから点検日毎の写真をまとめて保存・管理してきたが，点検対象設備（電子タグ）毎に時系列で管理されれば設備の変状履歴を把握していく上で非常に有用であると考えられる．現在，電子タグと写真ファイルの関連付けは PC で行っているが，現場の PDA にて電子タグと関連付けられた写真ファイルがデータベース化される過程で自動的に仕分けされれば，より利便性を向上できるものと考えている．

## 4．プロダクトモデルの概要

### (1) プロダクトモデルとは

各種構造物や製品のライフサイクルの中で，異なるアプリケーションシステム間において，形状や材料等に関するデータの相互運用を図るために，プロダクトモデルの開発や研究が行われている．プロダクトモデルの開発に関しては，国際標準として ISO10303 の STEP (Standard for the Exchange of Product model data)<sup>2)</sup>の中に Application Protocol (AP) があり，機械関連分野については，規定化が進んでいる．建築の業界標準としては，IAI (International Alliance for Interoperability) において IFC (Industry Foundation Classes)<sup>3)</sup>が策定され，将来は STEP の一部として取り込まれることを目指して，モデルの仕様等が開発されている．土木分野においては，日本では建設 CALS/EC (Continuous Acquisition and Life-cycle Support / Electronic Commerce)<sup>4)</sup>の中で，2次元 CAD 図面データの互換運用ができるよう，共通フォーマットが取り決められている．しかし，2次元 CAD データでは，CAD 以外のアプリケーションシステムに渡しても，部材そのもののデータが伝達されないことから，3次元のプロダクトモデルに関する研究や開発が関係各方面で行われるようになった<sup>5)6)</sup>．

我々は，これまでに鋼骨組構造，PC 中空床版橋，水圧鉄管等を対象として，プロダクトモデルを構築し，XML (Extensible Markup Language) を用いてコンピュータに実装して，図 - 8 に示す 3次元 CAD，設計照査，数量計算・積算，工程管理，維持管理等の各種アプリケーションシステム間でデータ相互運用を図ることにより，ライフサイクルにおいて全体的な効率化をもた

らすことが可能であることを示した<sup>7)~10)</sup>。しかし、プロダクトモデルのモデリングに関する標準的な方法論といったものは未だ確立されておらず、各研究者や組織等により、様々なアプローチがなされているのが現状である。また、コンピュータへの実装には一般に XML が使用されているが、種々の XML スキーマが存在している。従って、今後プロダクトモデルを実用化させていくためには、モデリングと実装に関して、標準的なアプローチを構築していくことが必要だと考えられる。

## (2) IFC

### 1) IFC の構成要素

IAI は「建設業における企画・設計・施工・保守管理等を通して、様々な業種によるコンピュータ上のデータ共有を実現するため、3次元 CAD を中心としたデータの仕様の定義を行い、各業種での利用の推進とその広報活動を行う」ことを目的として、1995年に米国の提案で結成された国際コンソーシアムである。IAI が提案するデータ仕様 IFC の中心は、建物及び建設産業に関する情報をオブジェクト指向の考え方で表現した情報モデルである<sup>11)</sup>。

IFC の現在のバージョンは、Release 2x(IFC 2x)<sup>3)12)</sup>である。(最近、IFC 2x の 2nd edition である IFC 2x2 が提案され、各国で検討中である。) 部材の形状表現、弾性係数等の属性値の定義、他部材との関係の定義等は、全て別々のクラスで定義される。つまり、図 - 9 に示すような、IfcObject、IfcPropertyDefinition、IfcRelationship、及び IfcRepresentationItem の 4 つのクラスの相互関係から定義される。

#### a) IfcObject

IfcObject クラスとそのサブクラスでは、物理的に存在する「もの」を定義する。形状を表現することは必須条件ではない。オブジェクトクラスの例として梁 (Beam)、柱 (Column) 等がイメージしやすいが、空間 (Space)、場所 (Site) 等も一つのオブジェクトとなる。柱を一つのオブジェクトとして考えた場合、長さや太さ等の形状の表現が必要不可欠となるが、形状の表現は、IfcObject クラスのサブクラス IfcProduct が、属性として保持している。つまり、形状を有する部材の定義は、IfcProduct クラスのサブクラスとして定義されなければならない。

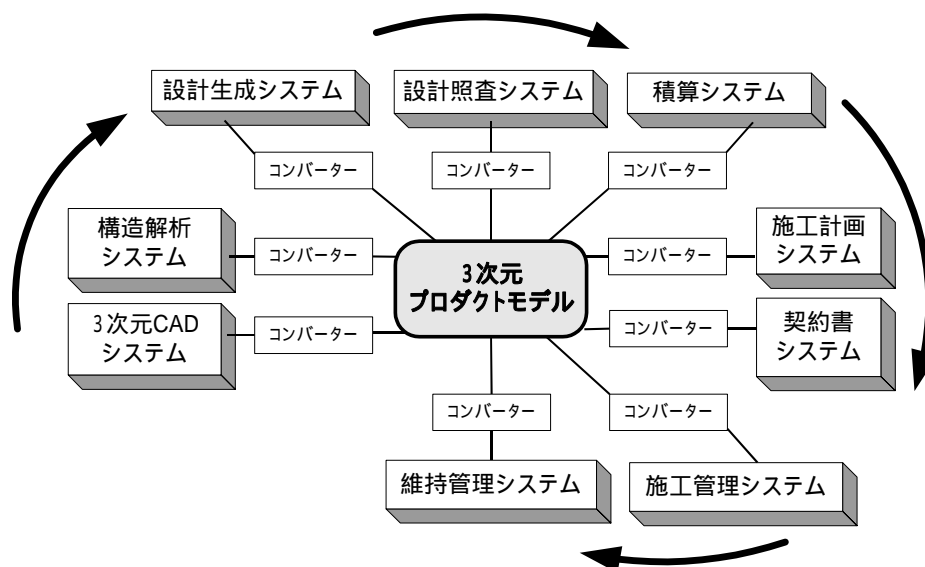


図 - 8 プロダクトモデルによるデータ相互運用

## b) IfcPropertyDefinition

IfcPropertyDefinition とそのサブクラスでは IfcObject 及びそのサブクラスで継承されない，その他の付加的な情報を格納する．例えば，柱を考えた場合，弾性係数  $E$  や降伏応力  $\sigma_y$  等の力学的パラメータがそれに相当する． $E$ ， $\sigma_y$  といった個々のデータは個々のクラスでそれぞれ定義され，プロパティセットとして，一つのまとまったデータベースとしての形態と機能を持つ．このように，オブジェクトとそのプロパティを分離して定義し実装させることで，複数のオブジェクトが同一のプロパティを共有する事が可能になり，インスタンスファイルの肥大化を抑制する事が可能となる．

## c) IfcRelationship

IfcRelationship とそのサブクラスは，オブジェクト間の関係や，オブジェクトとプロパティセットとの関係を定義する．IfcRelationship クラスのサブクラスには，特定の関係を定義するためのクラスが多数用意されている．例えば，柱と梁の接合関係を表現するには，要素間の関係を表現する専用のクラスが，サブクラスで用意されており，これを用いて部材間の関係を表現する．同様に，柱とその柱が所有するプロパティを関連付けするには，オブジェクトとプロパティを結びつける専用のサブクラスがあり，これを用いて関連付けを行う．

## d) IfcRepresentationItem

図 - 9 に示すように，IFC 2x では，IfcObject，IfcPropertyDefinition，IfcRelationship の 3 つのクラスは，IfcRoot クラスのサブクラスとして定義されている．IfcRoot をスーパークラスとするクラスの中で，形状表現を定義するクラスは存在しない．形状表現に関して IFC 2x では，幾何に関するリソースが提供されている．リソース内には，IfcRepresentationItem クラスをルートとして，点や線，立体といった，様々な図形表現を担う多数のクラスが存在する．つまり，

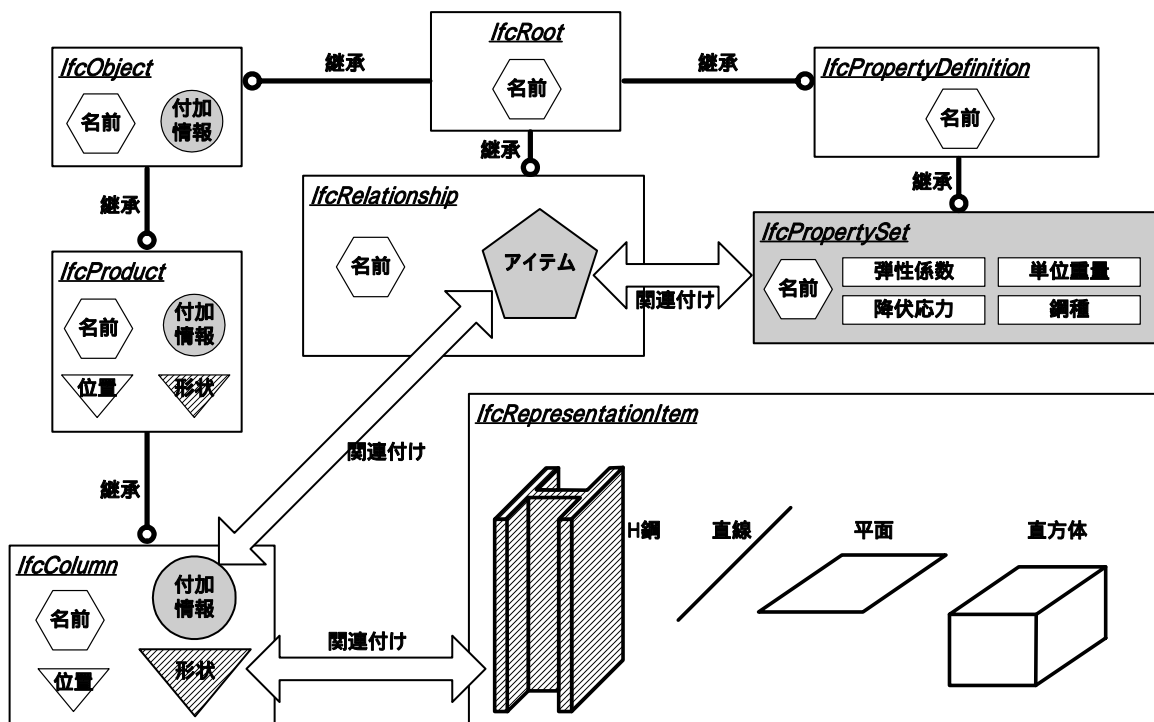


図 - 9 クラスの相互関係を表す概念図



部材がこれらのクラスのインスタンスを，形状を表現する属性の属性値として持つことで，部材の形状を表現することが可能となるのである．図 - 10 に IfcRepresentationItem クラスとそのサブクラスの一部を示す．

## 2) IFC の特長

### a) オブジェクトとプロパティ

IFC 2x では，ビルディングや家屋等の建築構造物を構成する要素として，梁，柱，ドア，窓，天井，床等の部材が，IfcBuildingElement のサブクラスとして用意されている．前述のように，プロパティセットによって，各部材のオブジェクトとプロパティは分離されている．部材オブジェクトのうち，ドアと窓については，既にプロパティセットが IFC の中で提供されており，IfcRelDefinesByProperties クラスによって各部材に関連付けすることができる．但し，梁，柱，天井，床等のプロパティセットは未だ提供されていないが，IfcPropertySet クラスを用いて，独自のプロパティセットを作成することが可能である．この場合も，IfcRelDefinesByProperties クラスによって各部材とプロパティセットを関連付けする．

各部材の位置や図形情報は次のような方法で表現される．各部材は，ObjectPlacement と Representation という名前の属性を持っている．属性 ObjectPlacement は，位置を表現するクラス (IfcAxis2Placement3D) を持ち，属性 Representation は，クラス IfcRepresentation を属性として持つ．さらに，IfcRepresentation クラスがベクター (IfcVector)，ポイント (IfcPoint)，ライン (IfcLine)，曲線 (IfcCurve)，サーフェス (IfcFaceBasedSurfaceModel)，ソリッド (IfcSolid) 等の形状の表現をするクラス (IfcRepresentationItem) を持っている．

### b) 部材の扱い

梁や柱等の部材内に含まれている鉄筋等のデータは，梁や柱のプロパティとして持たせることはできるが，鉄筋や PC ケーブル等を表現するクラスは，IFC 2x には定義されていない．そのた

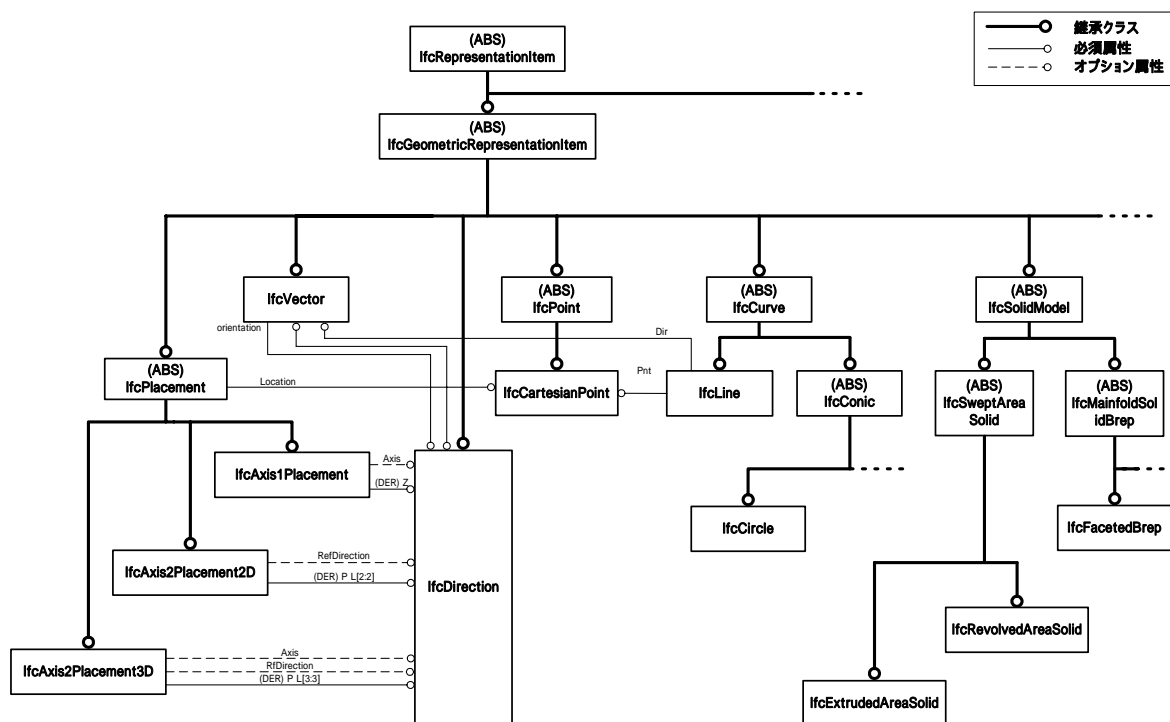


図 - 10 IfcRepresentationItem クラスとサブクラス

め、鉄筋等を IFC のオブジェクトとして、3次元 CAD システム上で図形表現することは極めて困難である。これでは、鉄筋等の干渉チェックやかぶり等の構造細目に関する照査ができず、施工設計に利用することができない。

また、IFC はビルディングや家屋を基本的には対象としているため、梁、柱、天井といった規則的な部材の表現には適しているが、土木構造物に見られるコンクリート躯体のような形状自由度が高い構造部材のクラスは定義されていない。

## 5 . PC 中空床版橋のプロダクトモデル

### (1) モデリングの考え方

IFC は前述のように、建築ビルディングを主対象としていることから、土木構造物に直接応用することは困難であるが、土木用のモデルをゼロから構築するのは非効率的である。さらに、IFC は国際標準にすることを目的として構築されていることから、IFC に準拠する形でモデリングすることは、将来の国際標準化の視点からも有利であると考えた。そこで、IFC の基本構造を壊さず、できるだけ必要最小限の部材クラスを新しく定義し、他の土木構造物の表現にも利用できるような汎用性を持たせるといった基本的な考え方で、モデリングを進めることとした。土木構造物としては、我々が以前にプロダクトモデルを別の考え方で構築した PC 中空床版橋を対象とすることとした。

まず、PC 中空床版橋を構成する部材を整理すると、コンクリート、ボイド（ホロー管）、PC 鋼材、PC 定着装置、シース、鉄筋等が挙げられる。これらの部材どうしの関係に着目すると、コンクリートの中にボイド、PC 鋼材、PC 定着装置、シース、鉄筋が「含まれている」すなわち「内部にある」点が問題となる。また、鉄筋は、形状自由度が高く、属性として、定着長や継手部等があり、特に継手部はどこまで現実に近い形で表現するかが課題となる。以下の節で議論していきたい。

### (2) 各部材

#### 1) コンクリート部材

コンクリートは、H 鋼のような形鋼と異なり、自由な形状であることが多い。また、鉄筋、ボイド、シース等をコンクリート内部に含んでいることから、コンクリートを完全なソリッドモデルとして定義しようとする、コンクリート内部の部材を定義する度に、コンクリートのソリッドから内部の部材を差し引く図形処理が必要となり、煩雑になると考えられる。一方、コンクリートを単なるサーフェスモデルにしてしまうと、面の内部と外部が識別できないので、3次元有限要素解析のためのメッシュ作成や積算のための数量計算等の作業が自動化できなくなってしまう。そこで、コンクリートの形状表現には、面のどちら側に物が詰まっているのかを表すことができるサーフェスに基づいた簡易的なソリッドモデルを使い、鉄筋やボイド等、コンクリート内部にある部材は、コンクリートの内部に「含まれている」ということが明確に表現できるようなモデルとすることが望ましいと考えられる。

こうした観点から IFC を検討してみると、図 - 11 に示すように、IfcRelationship クラスのサブクラスに IfcRelContainedInSpatialStructure があり、このクラスが、建築の梁、柱、壁、床、階段等の IfcBuildingElement が IfcBuilding というビルディングのクラスに「含まれる」関係を表現するのに用いられていることが注目される（図 - 12）。さらに、IfcBuilding の基本形状は IfcFacetedBrep（Brep 若しくは B-Rep とも言う）によって表現されるが、Brep はサーフェスで構成される、閉じた立体であり、内部に関する情報を持つことが可能である。つまり、先

に述べた 2 つの条件を満たすものであることがわかる。

そこで、我々は、PC 中空床版橋のコンクリート部分である床版を、IfcBuilding と同じ性質を持ったオブジェクトと判断し、IfcBuilding と同じ階層に新しく SlabOfBridge という名前のクラスを定義することとした。次に、床版 (SlabOfBridge) に含まれる鉄筋 (Rebar)、PC 鋼材定着装置 (PCFixingEquipment)、ボイド (Void)、PC 鋼材 (PCCable)、シース (Sheath) 等の部材クラスを、柱 (IfcColumn)、梁 (IfcBeam) 等と同じ階層に、新たに定義した。尚、鉄筋は鉄筋の集合体ではなく、一本一本独立したオブジェクトとして定義するものである。

次に、コンクリート、鉄筋、PC 鋼材等の各クラスの材料特性等の属性情報は、それぞれ専用のプロパティセットを定義した。各プロパティセットと各部材を IfcRelDefinesByProperties によって関連付ける。

## 2) 鉄筋

鉄筋は、前述のように鉄筋一本を、一つのオブジェクトとしてとらえることとした。基本的には、鉄筋の形状は、鉄筋径を直径として円を描き、長手方向をベクトルで示して、円をベクトル方向に「押し出す」ことにより表現することとした。但し、湾曲部がある鉄筋の場合、直線部と湾曲部を分離し、直線部には IfcExtrudedAreaSolid (押し出しソリッド) を用い、湾曲部には IfcRevolvedAreaSolid (回転押し出しソリッド) 等を用いて、別々にソリッドを形成して、それらを連結するという方法を用いることとした。

定着部に関しては、形状上の違いはないことから、オブジェクトとして別に扱うことはせず、定着部の位置情報、定着種別、定着長等のデータをプロパティセットとして定義することとした。

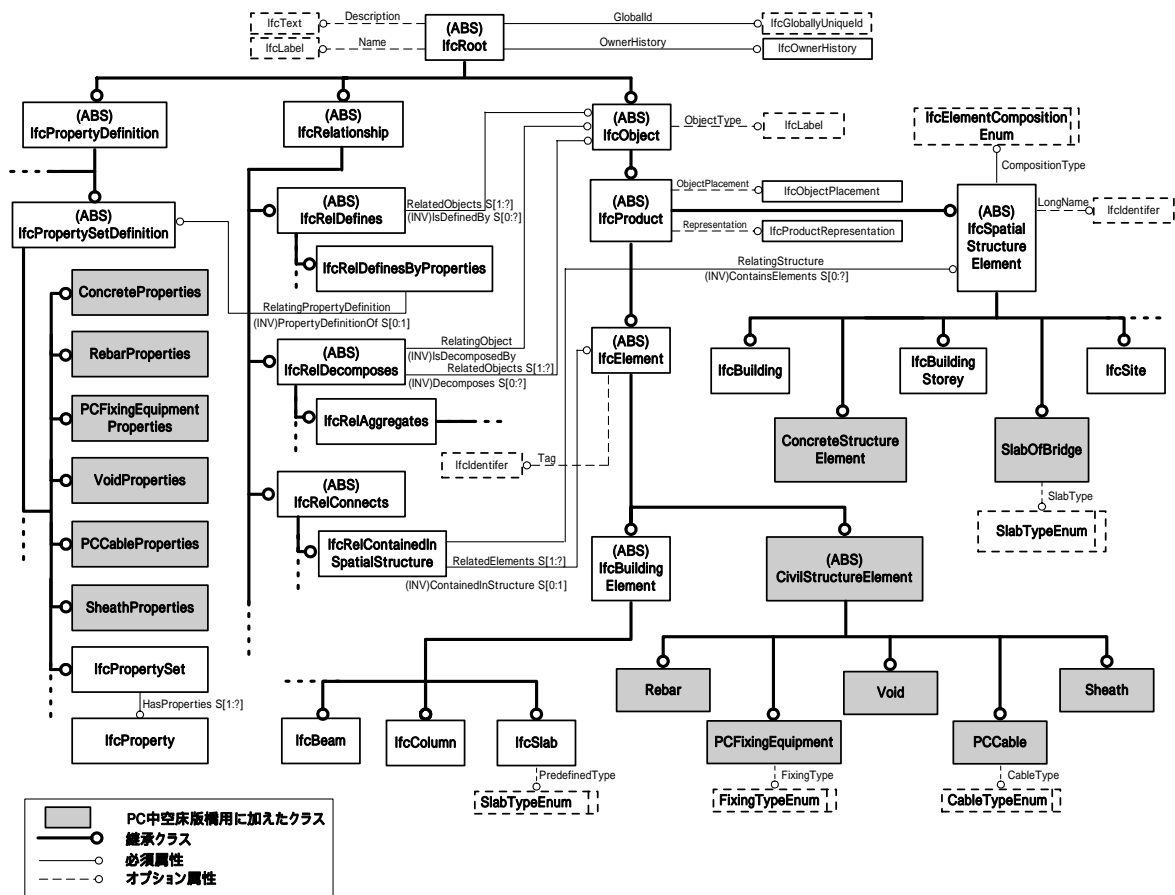


図 - 1 1 本研究で開発したクラスモデル (Express-G)

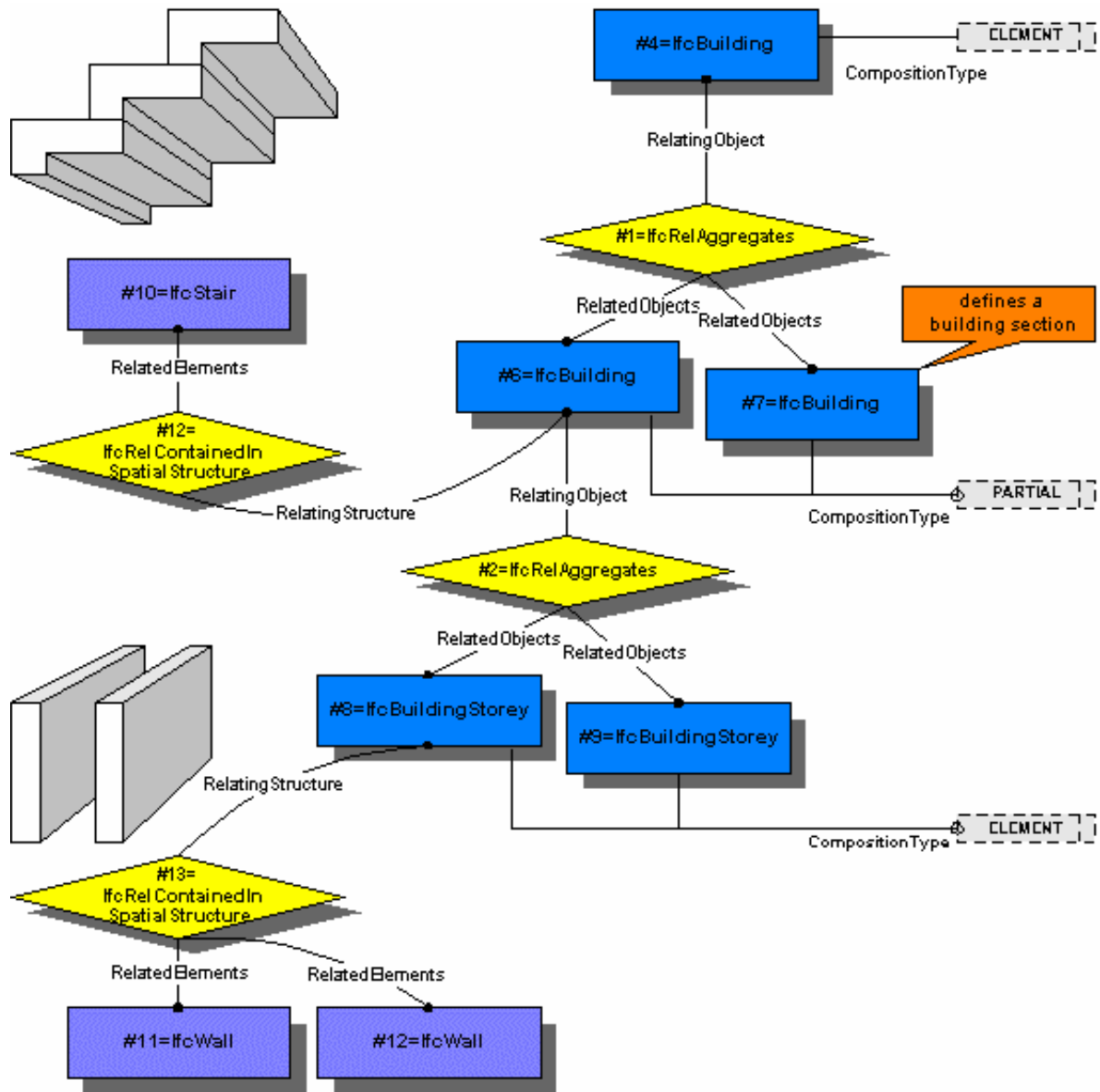


図 - 1 2 IfcRelContainedInSpatialStructure

継手部に関して，特に重ね継手は，異なる鉄筋 2 本を束ねたように表現するのではなく，形状的には 1 本の鉄筋として表現し，その部分が継手部であることを属性として表現することとした．その他，鉄筋の使用種別（主筋，スターラップ等），鋼材規格，呼び名，弾性係数等は，プロパティセットにより定義した（図 - 1 3）．

### 3) その他の部材

PC 鋼材の形状の表現も，鉄筋と同様の手法を用いて定義し，実装することとした．形状以外のデータ（鋼材種別，鋼材規格，呼び名，引張荷重（kN），伸び（%），リラクセーション値（%），公称断面積（mm<sup>2</sup>），単位重量（kg/km）等）の定義に関しては，図 - 1 3 に示す専用のプロパティセット（PCCableProperties）を新たに設けることにより定義した．これにより，実データは PCCableProperties クラスのインスタンスとして表現される．しかし，図 - 1 3 で定義されている全てのデータを，インスタンスが必ず所有する必要があるわけではない．引張荷重や伸び，リラクセーション値，公称断面積，単位重量等の値は，鋼材規格と呼び名がわかれば特定することが可能である．よって，インスタンスは鋼材規格と呼び名のデータのみを所持し，強度や重量

に関するデータが必要になった際には，JIS規格に準ずる PC 鋼材のデータベース等にリンクさせ，値を取得することも可能である．これは，前述の鉄筋の場合にも同様の事が言える．

その他，ボイド，シースのモデリングに関しても，形状はソリッドで表現するために，サーフェスモデルを用いず，IfcExtrudedAreaSolid を使用し，平面内に描かれた厚みのあるドーナツ状の円を，ベクトル方向に「押し出す」ことにより，形状を表現する．また，ボイド，シースに関する形状以外のデータは，それぞれプロパティセット（VoidProperties，SheathProperties）にて定義した（図 - 13）．また，シースのグラウト注入・排出口位置に関しては，その向きも特定出来るように，IfcAxis2Placement3D クラスで定義した．

### (3) プロダクトモデルの実装

#### 1) ifcXML

プロダクトモデルの実装に関しては，リレーショナルモデルやオブジェクト指向モデル等，種々のデータモデルをベースとした開発言語を用いることが可能であるが，ISO の STEP では，スキーマを EXPRESS 言語により定義し，物理ファイルは Part21 ファイルを使用することになっている．IFC も，基本的には STEP に準拠することになっている．しかし，EXPRESS を処理するソフトウェアは一般に高価であり，また自主開発するには相当な労力がかかる．一方，近年，開発や処理がしやすく，無料の XML が広く使用されるようになり，IFC でも実装に ifcXML<sup>13)</sup> を使用している．

ifcXML は IFC で開発されているもので，そのスキーマとしては，XML Schema<sup>14)</sup> が使用されている．XML Schema とは，W3C（World Wide Web Consortium）が 2001 年 5 月に勧告したスキーマ言語であり，文書を対象とした DTD（Document Type Definition）とは異なり，多様なデータ形式をサポートしていることから，今後，各種データ交換の際のフォーマットとなることが予想される．

ifcXML は，EXPRESS へのスキーマ変換が可能であり，インスタンスファイルは Part21 ファイルと等価である．また，階層指向型の構造形式を持ち，クラスを属性型として持つ場合，どの

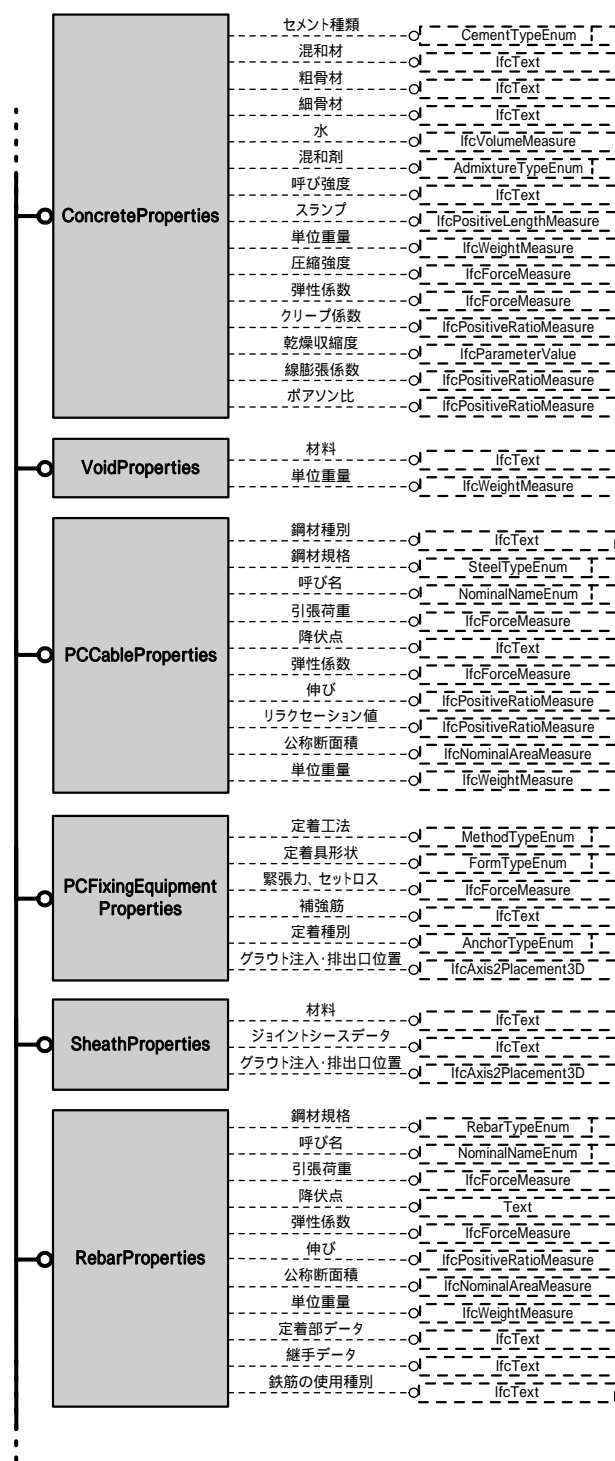


図 - 13 プロパティセット

クラスを持つのが特定可能であり、また、クラスの継承関係を表現することが可能である。さらに、SOAP ( Simple Object Access Protocol )<sup>15)</sup> に実装することができることから、分散処理に容易に適用可能である。

## 2) ifcXML によるプロダクトモデルの実装

我々は ifcXML を使用して、PC 中空床版橋のプロダクトモデルのスキーマを実装した。次に、このスキーマにより、適当な実際の PC 中空床版橋のデータを用いて、インスタンスファイルを作成した。プロダクトモデルのスキーマを図 - 14、インスタンスファイルの一部を図 - 15 と図 - 16 に示す。以前、我々がプロダクトモデルの実装に用いていた BLIS-XML に比べ、各クラスの継承関係や、内部データ構造を図 - 14 から容易に確認することが出来る。図 - 15 は、SlabOfBridge クラスのインスタンスを示しており、図の下部に示される複数の CartesianPoint は、床版の外形を決定するための座標値である。さらに、図 - 16 より IfcRelDefinesByProperties クラスが、床版にコンクリートのプロパティセット ConcreteProperties の関連付けを行い、IfcRelContainedInSpatialStructure クラスが、床版と床版内に含まれる各部材の関連付けを行っているのが確認出来る。さらに、図 - 15 のインスタンスファイルをもとに、3次元 CAD システム ( AutoCAD2002 ) 上で実際に再現した PC 中空床版橋の3次元モデルを図 - 17、18、19、20 に示す。

ifcXML は、クラスの継承機能を表現できるため、スキーマの構造は、BLIS-XML に較べて単純かつ明解になった。また、ifcXML は、階層的な構造をとることで、BLIS-XML に比較し、インスタンスファイルは相当多くの行数になった。しかし、スキーマとインスタンスの整合性の検証が可能のため、モデルデータの信頼性が向上したと考えられる。

## 6. プロダクトモデルによるアプリケーションシステムの統合

本研究では、図 - 21 に示すように、プロダクトモデルを中心として、3次元 CAD システム、PC 上部工の設計計算システム、及び鉄筋かぶりチェックシステムを統合化したシステムモデルを構築した。さらに、モデルの有効性を実証する目的で、プロトタイプシステムを開発した。以下、各システムとプロダクトモデルとの統合化について記す。

### (1) 3つのシステムとの統合

#### 1) 3次元 CAD システムとの統合化

本研究では、3次元 CAD システムとしては、AutoCAD 2002<sup>16)</sup> を使用した。CAD システムとプロダクトモデルデータとの間でデータの相互運用ができるように、データの変換を行うコンバータプログラムを2つ ( コンバータプログラム 及び ) 作成した。

コンバータプログラム は、3次元 CAD システム ( AutoCAD 2002 ) において設計した

```
<!--
+ <xsd:complexType abstract="true" name="Root">
+ <xsd:complexType abstract="true" name="Object">
- <xsd:complexType abstract="true" name="Product">
- <xsd:complexContent>
- <xsd:extension base="Object">
- <xsd:sequence>
+ <xsd:element minOccurs="0" name="objectPlacement">
- <xsd:complexType>
+ <xsd:choice>
+ <xsd:element ref="ObjectPlacement" />
+ <xsd:element ref="ObjectPlacementRef" />
+ </xsd:choice>
+ </xsd:complexType>
+ </xsd:element>
+ <xsd:element minOccurs="0" name="representation">
- <xsd:complexType>
+ <xsd:choice>
+ <xsd:element ref="ProductRepresentation" />
+ <xsd:element ref="ProductRepresentationRef" />
+ </xsd:choice>
+ </xsd:complexType>
+ </xsd:element>
+ </xsd:sequence>
+ </xsd:extensionContent>
+ </xsd:complexType>
- <xsd:complexType abstract="true" name="SpatialStructureElement">
+ <xsd:complexContent>
- <xsd:extension base="Product">
+ <xsd:sequence>
+ <xsd:element type="Identifier" minOccurs="0" name="longName" />
+ <xsd:element type="ElementCompositionEnum" name="compositionType" />
+ <xsd:element name="containsElements" minOccurs="0" maxOccurs="unbounded">
- <xsd:complexType>
+ <xsd:attribute ref="inverse" />
+ </xsd:complexType>
+ </xsd:element>
+ </xsd:sequence>
+ </xsd:extensionContent>
+ </xsd:complexType>
+ <xsd:complexType name="SlabOfBridge">
- <xsd:complexContent>
- <xsd:extension base="SpatialStructureElement">
+ <xsd:sequence>
+ <xsd:element type="SlabTypeEnum" name="slabType" />
-->
```

図 - 14 スキーマ ( SlabOfBridge 部 )



```

<?xml version="1.0" encoding="utf-8" ?>
- <ifcXML schema="IFC 2x Final">
- <SlabOfBridge id="_1001">
  <globalId>|.CY[$]o+Hw=#rH0,xel</globalId>
  + <ownerHistory>
    <name>Original Bridge</name>
    <description />
    <objectType />
    <isDefinedBy inverse="_101" />
    <isDecomposedBy />
    <decomposes />
  + <objectPlacement>
  + <representation>
    <longName />
    <compositionType>COMPLEX</compositionType>
    <slabType>PCHollowSlabBridge</slabType>
    <containsElements inverse="_1000" />
  </SlabOfBridge>
- <CartesianPoint id="_1004">
  <coordinates>0,0,0</coordinates>
</CartesianPoint>
- <CartesianPoint id="_1009">
  <coordinates>0,0,0</coordinates>
</CartesianPoint>
- <CartesianPoint id="_1201">
  <coordinates>0,0,1275</coordinates>
</CartesianPoint>
- <CartesianPoint id="_1202">
  <coordinates>0,0,1525</coordinates>
</CartesianPoint>
- <CartesianPoint id="_1203">
  <coordinates>0,13100,1525</coordinates>
</CartesianPoint>
- <CartesianPoint id="_1204">
  <coordinates>0,13100,1275</coordinates>
</CartesianPoint>
- <CartesianPoint id="_1205">
  <coordinates>0,12450,1175</coordinates>
</CartesianPoint>
- <CartesianPoint id="_1206">
  <coordinates>0,12450,525</coordinates>
</CartesianPoint>
- <CartesianPoint id="_1207">
  <coordinates>0,650,525</coordinates>
</CartesianPoint>
- <CartesianPoint id="_1208">
  <coordinates>0,650,1175</coordinates>

```

図 - 15 インスタンス(SlabOfBridge 部)

```

- <RelDefinesByProperties id="_101">
  - <relatedObjects>
    <SlabOfBridge href="_1001" />
  </relatedObjects>
  - <relatingPropertyDefinition>
    <ConcreteProperties id="_202" />
  </relatingPropertyDefinition>
</RelDefinesByProperties>
- <RelContainedInSpatialStructure id="_1000">
  - <RelatingStructure>
    <SlabOfBridge href="_1001" />
  </RelatingStructure>
  - <RelatedElements>
    <Rebar href="_2001" />
    <Rebar href="_3001" />
    <Rebar href="_4001" />

```

図 - 16 インスタンス  
(リレーションシップ部)

構造物のデータを，ifcXML 形式のプロダクトモデルのインスタンスファイルとして半自動的に生成するものである。開発には VBA (Visual Basic for Application) を用いた。

コンバータプログラム は，ifcXML のインスタンスファイルを一時的に DOM (Document Object

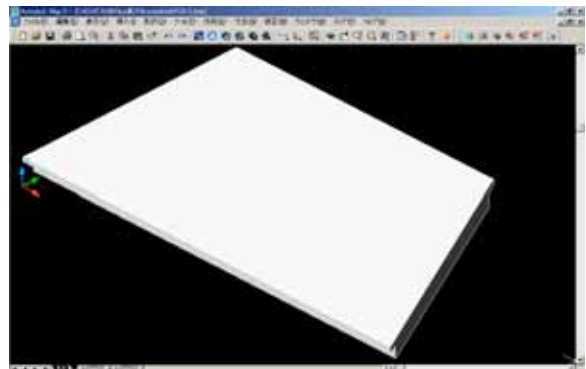


図 - 17 PC 中空床版の 3 次元モデル

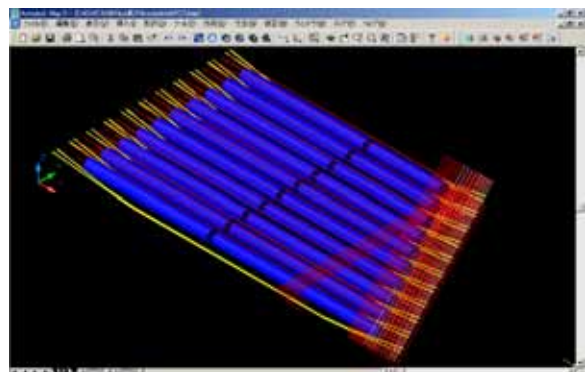


図 - 18 PC 中空床版の内部構造

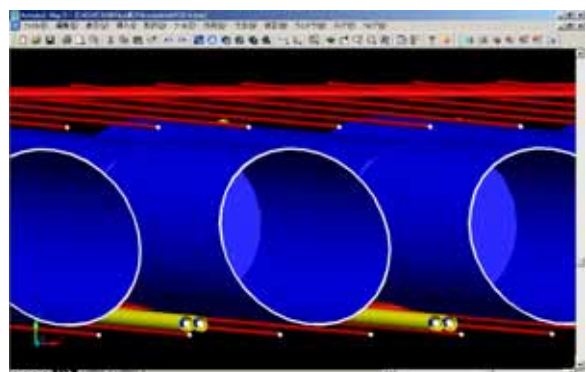


図 - 19 断面の様子

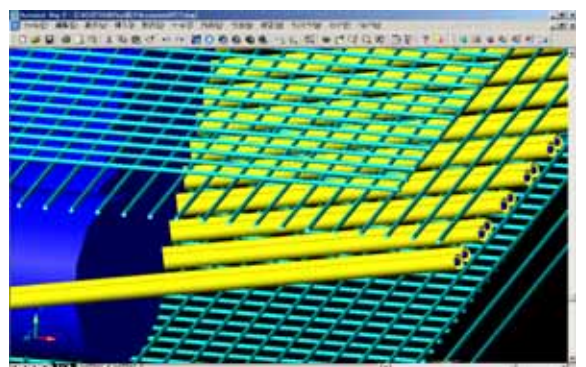


図 - 20 鉄筋とシースの様子

Model)としてコンピュータ上のメモリに展開してからファイル内部のデータを読み取り, 3次元 CAD 上にモデリングを自動実行するものである. XML パーサに MSXML version2.0 を使用し, プログラム本体の開発には VBA を用いた.

## 2) 設計ソフトウェアとの統合化

3次元 CAD において PC 中空床版橋の概略設計を行った後, 力学的な照査を行う必要がある. 本研究では, FORUM8 社製の PC 中空床版の設計計算用ソフト UC-1<sup>17)</sup>を力学的な照査を行うサンプルソフトとして使用することとした. さらに, 照査実行時に必要となるデータを ifcXML インスタンスファイル内から抽出するコンバータプログラムを開発した. コンバータプログラムは, UC-1 に入力すべきデータの一覧を表示し, ユーザの数値入力作業をサポートする. コンバータプログラムと同様, XML パーサに MSXML を搭載し, Visual Basic を用いて開発した.

## 3) 鉄筋かぶり照査システムとの統合化

設計ソフトウェアを用いて力学的な照査を実施した後, 主鉄筋, スターラップ等の配置や形状等を決定し 3次元 CAD システム上で詳細な設計を行う. その後, 鉄筋の定着や鋼材のかぶり等の構造細目に関する照査を行う必要がある. そこで本研究では, 構造細目の中でも特に鋼材のかぶりに着目し, 本プロダクトモデルのデータから最小かぶり値を求めるシステムを Java Servlet と XML for Java Parser 2.0.9 を用いて開発した. 設計基準として, 道路橋示方書<sup>18)</sup>の一部をプログラム化した. 以下に照査の簡単なフローを示す.

まず, ifcXML インスタンスファイルの中から床版の上面, 側面, 底面, さらに鉄筋の中心軸に関するデータを読み取り, それぞれ平面の方程式, 媒介変数を用いた直線の方程式として定式化する. 次に, 直線上の始点から終点間に存在する各点と, 平面との距離を求め, さらに鉄筋の半径の長さを差引き, かぶりを求める. このうち最小値となったものを最小かぶり値とし, 設計基準値との比較を行う.

上記の処理をインスタンスファイル内にある全ての鉄筋について行い 結果を画面に表示する. さらに, 基準を満足しなかった鉄筋に関しては, 修正を促すメッセージとかぶりの値が鉄筋のプロパティセット内に新たに書き込まれ, プロダクトモデルが更新される. さらに, コンバータプログラムにより違反した鉄筋を 3次元 CAD 上で色分けして表示出来るようにした.

## (2) 統合化システムの適用事例

本研究では, 表 - 1 に示す PC 中空床版橋の設計に, 本研究で開発した統合化システムを適用させ, システムの有効性を検討することとした. 以下に, 本システムを用いた設計作業の流れを示す(図 - 21).

まず, 設計者は 3次元 CAD システムを用いて, PC 中空床版橋の概略の設計を行う(図 - 22). この段階では, 鉄筋やスターラップ等はモデリングされず, 鉄筋径とピッチ等のデータのみを決定する.

次に, コンバータプログラムを実行し, の設計データを図 - 14 に示されるスキーマに従って, インスタンスファイル(図 - 15)に変換して保存する. さらに, コンバータプログラムを実行し, 照査に必要なデータの一覧を表示させ(図 - 23)解析モデルを作成し(図 - 24), PC 上部工設計計算システム UC-1 を用いて力学的な照査を行う(図 - 25).



Critical Failure Ratio Summary			
Calculation	Failure Description	Node	Critical Ratio
<b>PC Analysis</b>			
Tension Steel	Not Calculated		0.000
Concrete Shear Stress (Design Loads)	Critical Live Load	G2-12	0.978
Concrete Diagonal Tension Stress	Critical Live Load Neutral Axis	G2-11	0.257
Shear Reinforcing Stress	Critical Snow Load	G2-1	0.000
Fatigue Strength (shear reinforcing)	Not Calculated		0.000
Reinforcing Stress ( PC )	Critical Live Load Top	G2-9	0.254
PC Cable Stress ( PC )	Critical Live Load Cable Centroid	G2-8	0.734
Concrete Stress ( PC )	Critical Construction Top	G2-15	0.706
Average Efficiency Rating			<b>0.366</b>
<b>Strength Analysis ( Ultimate Actions )</b>			
Ultimate Moment Capacity	1.7D + 1.7L <sub>max</sub> + 1.0P	G2-9	0.738
Concrete Shear Stress (Ultimate Loads)	1.7D + 1.7L + 1.0P	G2-14	0.181
Average Efficiency Rating			<b>0.459</b>
<b>PRC Analysis</b>			
Flexural Crack Widths	Critical Live Load Bottom	G2-9	0.393
Fatigue Strength (longitudinal reinforcing)	No Support Settlement Bottom Steel	G2-9	0.440
Reinforcing Stress ( PRC )	Critical Live Load Bottom	G2-9	0.353
PC Cable Stress ( PRC )	Critical Live Load Cable Centroid	G2-9	0.799
Concrete Stress ( PRC )	Critical Snow Load Bottom	G2-4	0.578
Average Efficiency Rating			<b>0.513</b>

図 - 2 5 照査結果画面

照査を満足したため、再度3次元CADシステムに戻り、鉄筋、スターラップを含めた詳細な設計を行う(図-26)。作業終了後、再度コンバータプログラムを起動しインスタンスファイルを更新する。

次いで、構造細目照査システムにおいてかぶり計算を行う。まず、本システムにインスタンスファイルを読み込ませ、計算実行ボタンを押す。図-27は結果表示画面の一部である。スターラップの一つが基準値を満たしておらず、エラーが表示されているのが確認出来る。そこで、同フォーム下部にあるコマンドボタンをクリックし、エラーメッセージとかぶりの値を、エラーが報告されたスターラップのプロパティセットに書込み(図-28)、終了する。

最後に、コンバータプログラムを用いて、先程更新されたインスタンスファイルを読み込ませ、自動的にモデリングを行う(図-29)。同図から、ユーザは、かぶり値にエラーがあったスターラップを容易に確認し、修正することが出来る。

尚、本的用例には通常のパソコン(Windows2000, pentium4, 2.4GHz, Memory, 1024MB)を用いたが、レスポンス時間は特に遅いという感覚は得られなかった。

## 7. 電子タグとプロダクトモデルの統合化

本研究では、電子タグを鉄筋やPCケーブル等の部材や構成要素に貼付け、ID番号のみならず、開発したプロダクトモデルデータの該当する各部材の属性や設計・施工情報を記憶させることにより、電子タグとプロダクトモデルとを統合化したモデルを開発した(図-30)。電子タグは、非接触でデータの読み書きが可能であり、電波はコンクリートもある程度の厚さならば通過できるため、鉄筋やPCケーブル等に貼り付けた電子タグのデータは読み書きすることが可能である。将来、あらゆる部材や構成要素に、製造過程で電子タグが予め取り付けられるようになると予想



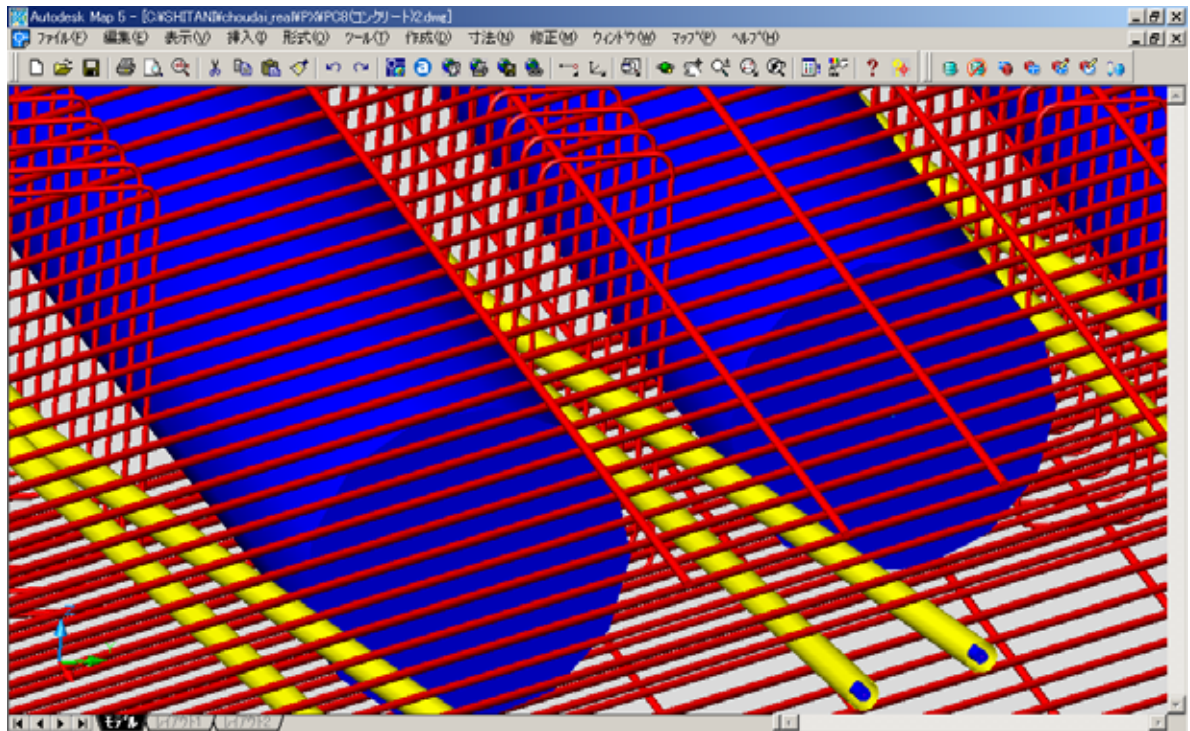


図 - 2 6 PC 中空床版の詳細設計



図 - 2 7 かぶりの照査結果画面



図 - 2 8 新たにエラー情報が書き込まれたプロパティセット

されるが、そこに記録されるデータは、製品番号、製造会社、製造工場、製造年月日等の製造者側から提供されるものに限られよう。本モデルでは、この電子タグあるいは別に貼り付ける新しい電子タグに、構造物における部材や構成要素としてのデータを記憶させるものである。そのデータとしては、部材としての ID 番号、部材名、部材の位置情報、形状データ、材料情報、その他設計者や施工担当者が残しておきたいと考えた情報等があげられる。

電子タグとプロダクトモデルの情報を統合化するためには、両者間でデータの取捨選択、変換を行うコンバータが必要である。プロダクトモデルデータから電子タグへは、ifcXML で記述されているデータをパーサ( Parser )により読取り、必要なデータを別ファイル等に格納してから、電子タグに書き込む。逆の場合も、同様である。

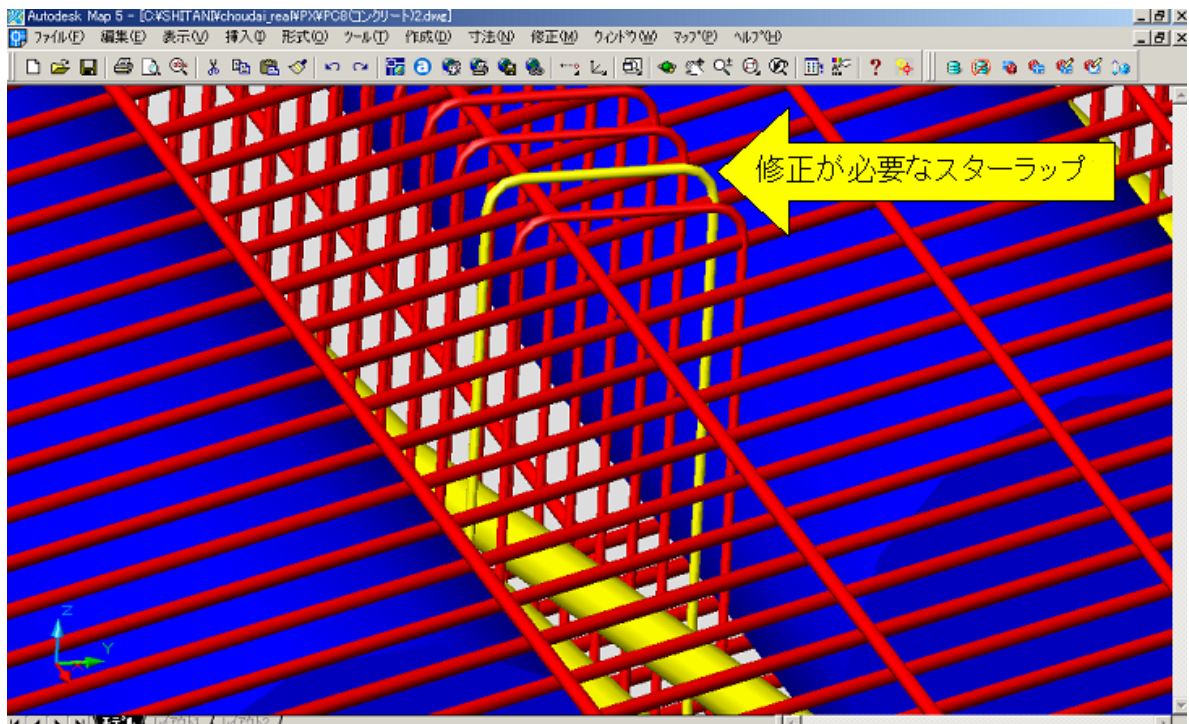


図 - 2 9 かぶりが不足した鉄筋の表示

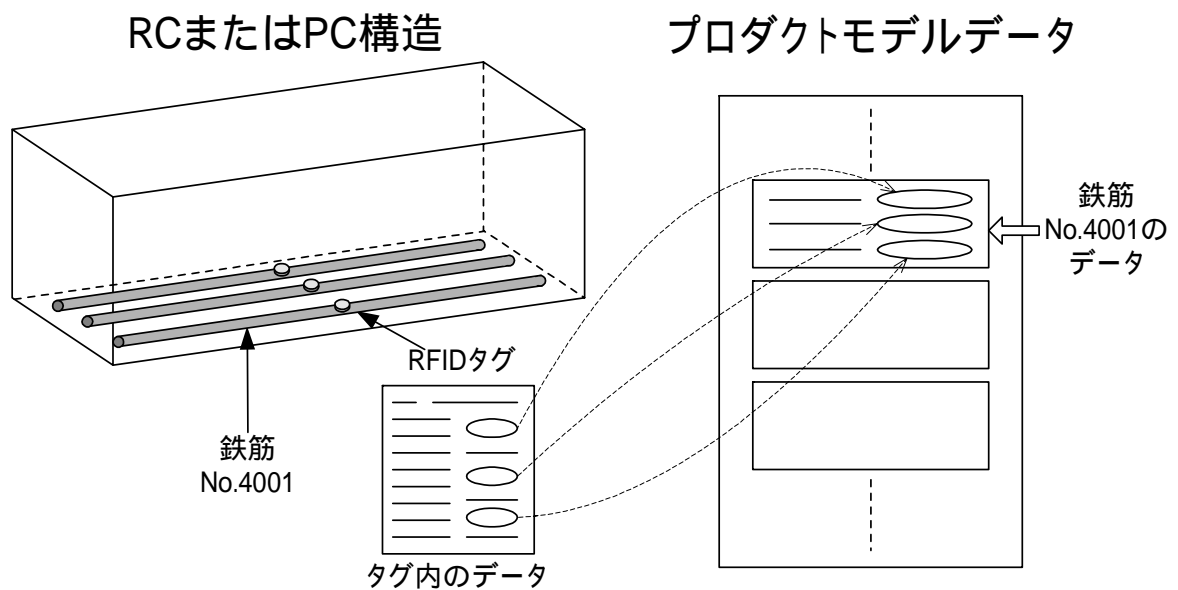


図 - 3 0 RFID タグとプロダクトモデルを統合化したモデル

適用事例として、PC 中空床版橋のプロダクトモデルのインスタンスファイルにおけるある鉄筋 (No. 4001) に関する部分を図 - 3 1 に示す。このデータから必要な箇所を読み取り、電子タグに記憶させるファイルを図 - 3 2 に示す。これにより、施工現場においても、鉄筋が図面通りに配置されているかをチェックすることが出来るし、維持管理の段階においては、内部にどのような鉄筋がどのように配置されているかをデータレベルで確認することが可能となる。

## 8 . おわりに

本研究では、土木構造物の建設および維持管理に、電子タグ、プロダクトモデル、及び両者を統合化したモデルを適用させることにより、現場において必要なデータを即座に読み書きができ





図 - 3 1 鉄筋 No. 4001 のインスタンスモデルの一部

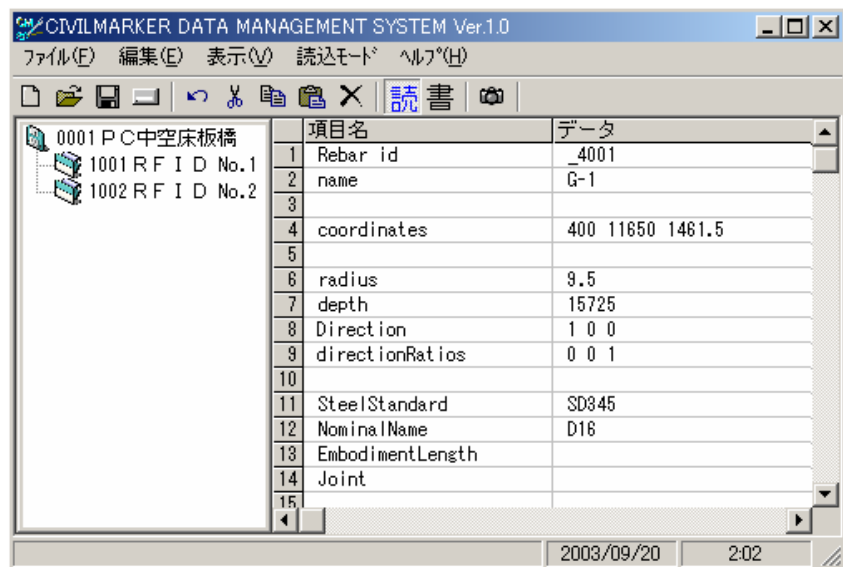


図 - 3 2 鉄筋 No. 4001 に取り付けられた RFID 内のデータ

るシステムモデルを提案し、実際の現場に一部適用し、検証を行った。以下、研究成果を列挙する。

- 電子タグの活用方法として、単に ID 情報を記憶させるにとどまらず、プログラムを起動させるトリガー機能及び点検データやアドバイス等のナレッジを保管する簡易データ保管機能を提案した。
- 電子タグによるデータの保管方式として、サーバ集中方式、分散メモリ方式、携帯コンピュータ方式の各方式について論じ、3つの方式を組合せたハイブリッド方式を提案した。
- 電子タグシステムを PDA とパソコン等を使用して開発し、既設のダム点検業務へ適用し、使用性などを検証した。
- 土木構造物の内、RC および PC 構造物及び内部にある鉄筋、PC ケーブル等のプロダクトモデルを、IFC を基本にしながら、開発した。特に、内部に構成要素を含むコンクリート部材のモデル化においては、工夫を施した。
- PC 中空床版橋の上部工のプロダクトモデルを開発し、ifcXML を用いて、コンピュータに実装した。
- PCI 中空床版橋のプロダクトモデルを使用して、異なる3個のアプリケーションシステムの統合化を実施した。
- 電子タグとプロダクトモデルの統合化の方法を提案し、適用事例を示した。

今後の課題としては、実際の建設途中の RC あるいは PC 構造物の鉄筋や PC ケーブルに電子タグを貼り付け、プロダクトモデルを作成して、統合化する実験を実施したいと考えている。

最後に、本研究を遂行するにあたり、電源開発株式会社の嶋田善多氏、坂田智己氏、室蘭工業大学大学院生の志谷倫章君、植田国彦君、小谷隼君から多大な協力を頂いた。ここに深謝に意を表します。

#### 【参考文献】

- 1) AIM : <http://www.aimglobal.org/technologies/rfid/>
- 2) ISO10303-1 : Industrial Automation Systems and Integration-Product Data Representation and Exchange, Part 1:Overview and Fundamental Principles, 1994.
- 3) IAI : <http://iaiweb.lbl.gov/>
- 4) 国土交通省 : <http://www.mlit.go.jp/tec/cals/>
- 5) 山崎元也, 本郷延悦, 千葉洋一郎: Japan Highway Data Model 構築の基礎研究, 土木情報システム論文集, Vol.10, pp.33-42, 2001.
- 6) 三上市藏, 田中成典, 石井由美子, 奥裕子: 鋼道路橋維持管理業務支援のための三次元モデルライブラリシステムの開発, 土木情報システム論文集, Vol.10, pp.129-136, 2001.
- 7) 矢吹信喜, 志谷倫章, 宮島良将, 岸徳光: 統合化された鋼構造接合部の設計システムに関する研究, 土木情報システム論文集, Vol.10, pp.175-184, 2001.
- 8) 矢吹信喜, 志谷倫章, 小谷隼: 鋼部材3次元プロダクトモデルの開発に関する研究, 土木学会北海道支部論文報告集, 第58号, pp.76-79, 2002.
- 9) 矢吹信喜, 古川将也, 加藤佳孝, 横田勉, 小西哲司: プロダクトモデルによる PC 中空床版橋の設計照査と概略積算の統合化, 土木情報システム論文集, Vol.10, pp.213-220, 2001.
- 10) 矢吹信喜, 齊藤大輔: 3次元プロダクトモデルと電子タグによる水圧鉄管の点検情報システム, 土木情報システム論文集, Vol.10, pp.113-120, 2001.

- 11) 古川暁，宇佐美祐人，鹿島孝，片岡裕美，千葉貴史，八坂文子，吉江慶祐，陸野康之：IFC 鉄筋コンクリート構造モデル仕様の提案の概要，第 23 回情報システム利用技術シンポジウム 論文集，pp.217-222, 2000.
- 12) IFC 2x : [http://cic.vtt.fi/niai/technical/ifc\\_2x/](http://cic.vtt.fi/niai/technical/ifc_2x/)
- 13) IfcXML : [http://www.iai-international.org/iai\\_international/Technical\\_Documents/IFC XML.htm](http://www.iai-international.org/iai_international/Technical_Documents/IFC_XML.htm)
- 14) XML Schema : <http://www.w3.org/>
- 15) SOAP : <http://www.w3.org/2002/ws/>
- 16) AutoCAD 2002 ユーザガイド，オートデスク，2001.
- 17) UC-Win/UC-1 シリーズ 導入の手引き，フォーラムエイト，2002.
- 18) 道路橋示方書（ 共通編・ コンクリート橋編 ）・ 同解説，日本道路協会，2002.

#### 助成研究者紹介

やぶき のぶよし

矢吹 信喜

現職：室蘭工業大学工学部助教授（Ph.D.）

主な著書：土木工学におけるコンピュータの高度利用技術（土木学会，平成 10 年）