

I-5 IFC に基づいた PC 中空床版橋の 3 次元プロダクトモデルの開発

Development of a 3 Dimensional Product Model for Prestressed Concrete
Hollow Slab Bridges Based on Industry Foundation Classes (IFC)

矢吹 信喜¹ 志谷 倫章²

Nobuyoshi Yabuki, Tomoaki Shitani

【抄録】 PC 中空床版橋を対象として、基本設計から詳細設計、施工設計における各種アプリケーションシステム間のデータ相互運用を可能とするために、3次元プロダクトモデルを開発した。開発に当たっては、モデルの国際標準化を目指して、IAI の IFC に準拠した形になるよう工夫しながら、PC 中空床版橋の特徴を分析し、スラブ、PC 鋼材、ボイド、鉄筋等のクラスを新しく定義した。本モデルの実装のために、複数の開発言語の中から ifcXML を選定し、本プロダクトモデルのスキーマとインスタンスを実装した。本モデルは、各種システム間のデータ相互運用に利用することが可能である。

【Abstract】 We developed a 3D product model for prestressed concrete hollow slab bridges in order to enable interoperability of various application systems throughout preliminary, detailed, and construction design phases. The product model has been developed on the basis of IFC of IAI, aiming an international standard. We analyzed the characteristics of the bridge and defined new classes for slabs, PC cables, voids, rebars, etc. We selected ifcXML for implementation of the product model among several languages, and implemented the schema and an instance, using ifcXML. This model can be used for data transferring and conversion among various systems.

【キーワード】 IAI, IFC, XML, プロダクトモデル, PC 中空床版橋, 3次元 CAD, CALS/EC

【Keywords】 IAI, IFC, XML, product model, prestressed concrete hollow slab bridge, 3D-CAD, CALS/EC

1. はじめに

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)¹⁾ 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)²⁾ 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)³⁾ 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^{4) 5)}。

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-1 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

1 正会員 Ph.D. 室蘭工業大学工学部建設システム工学科 助教授 〒050-8585 室蘭市水元町 27-1
TEL: 0143-46-5219 FAX: 0143-46-5218 Email: yabuki@news3.ce.muroran-it.ac.jp

2 学生員 室蘭工業大学大学院工学研究科建設システム工学専攻

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^{6)~9)}。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。

本研究では、世界 25 カ国のメンバーを擁する IAI が、将来国際標準とすることを目標として、建築のビルディングのプロダクトモデルを構築している IFC に着目し、IFC をベースに、橋梁の上部工としてはかなり複雑な PC 中空床版橋を対象として、3次元プロダクトモデルを開発し、IFC 用に開発された ifcXML を用いて実装することを目的とした。尚、本プロダクトモデルは、基本設計のみならず、詳細設計および施工設計にも利用可能であることを念頭において、開発した。

2. IFC 2x

2.1 IFC 2x 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¹⁰⁾。

IFC の現在のバージョンは、Release 2x (IFC 2x)^{2) 11)} で、オブジェクト指向技術が適用されたクラス群によって構成されている。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-2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。

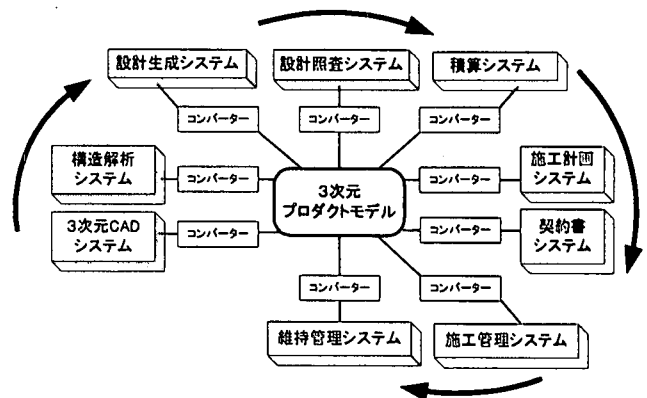


図-1 プロダクトモデルによるデータ相互運用

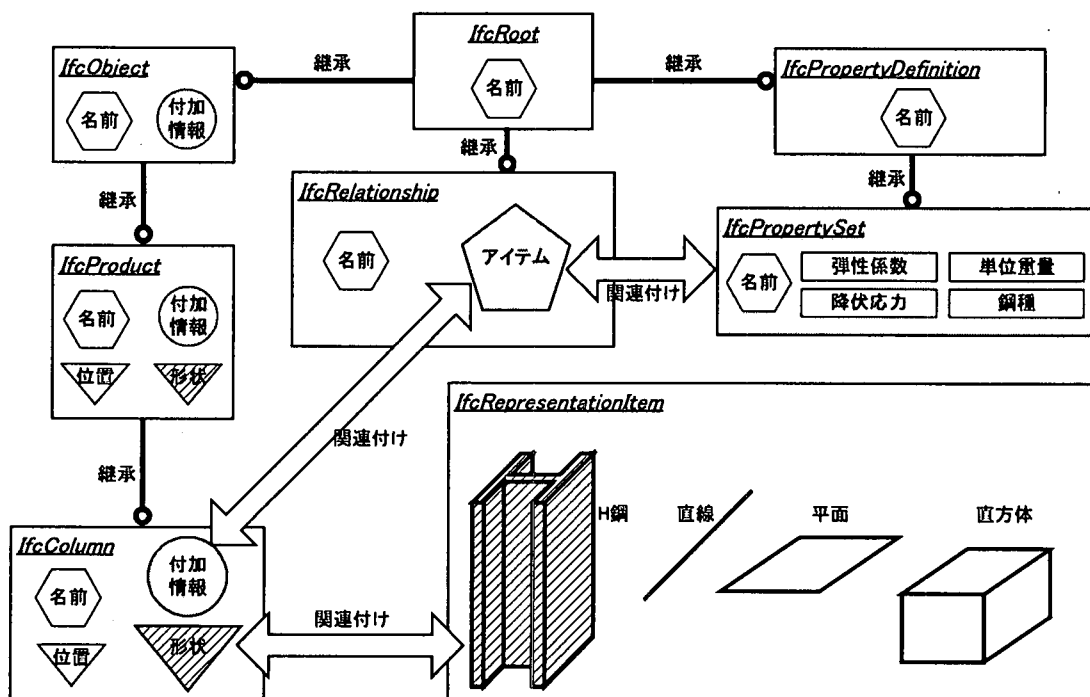


図-2 クラスの相互関係を表す概念図

(1) 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。

(2) 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 σ_y 等の力学的パラメータがそれに相当する。 E 、 σ_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。

(3) 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。

(4) IfcRepresentationItem

図-2 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、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-3 に

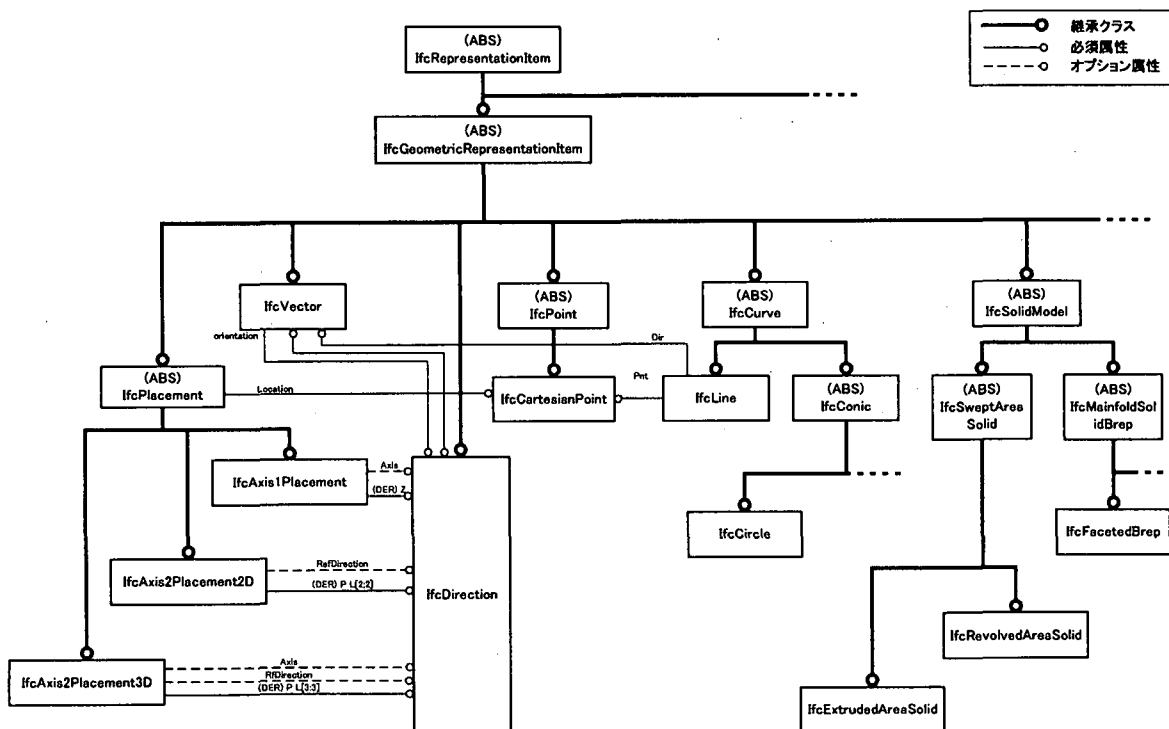


図-3 IfcRepresentationItem クラスとサブクラス

IfcRepresentationItem クラスとそのサブクラスの一部を示す。

2. 2 IFC の特徴

(1) 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、IfcPropetySet 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。

(2) 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。

3. PC 中空床版橋のプロダクトモデル

3. 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観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汎用性を持たせるという基本的な考え方で、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、シーす、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、シーす、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。

3. 2 コンクリート

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、シーす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と、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-4 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-5)。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マテリアル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。

3.3 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

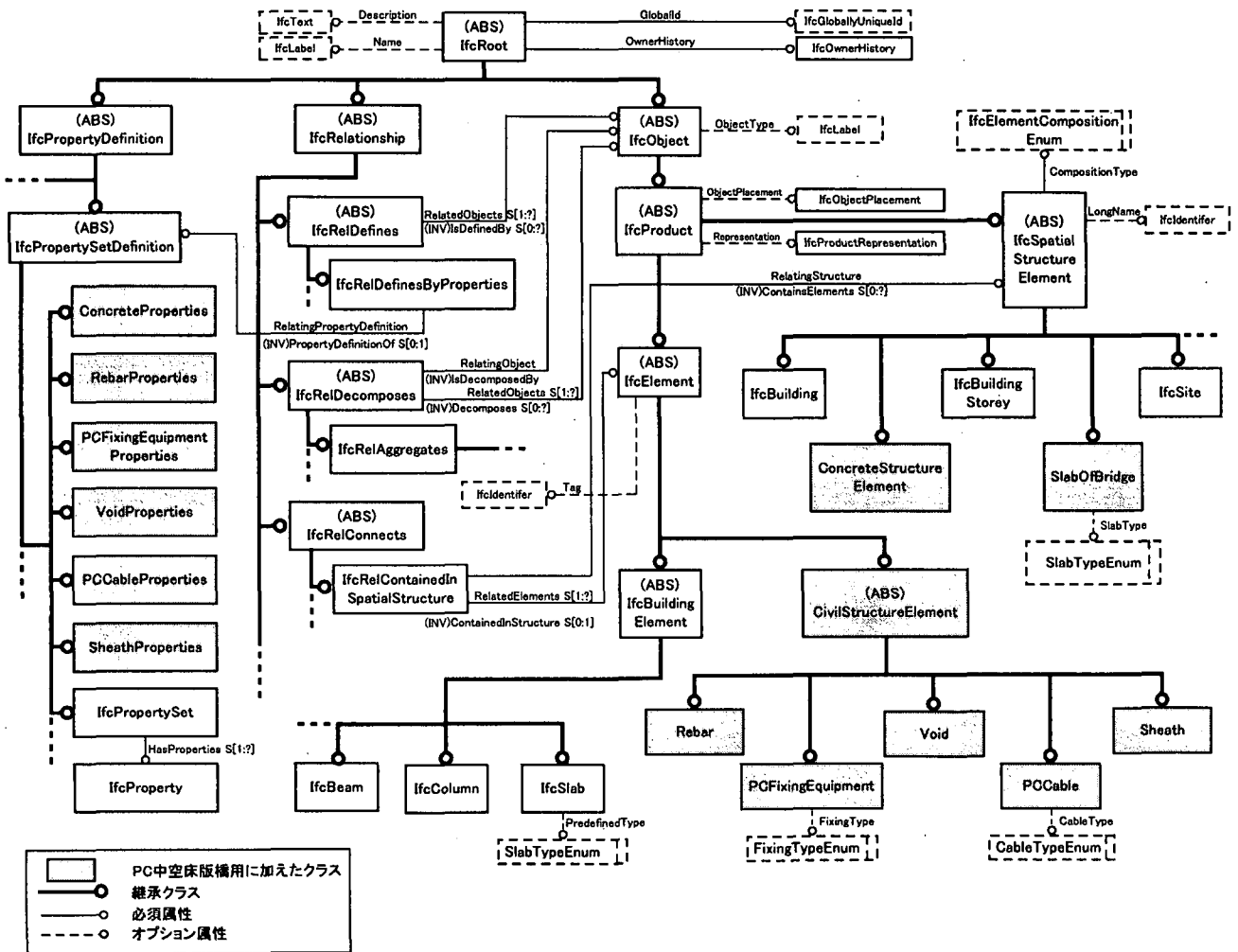


図-4 本研究で開発したクラスモデル (Express-G)

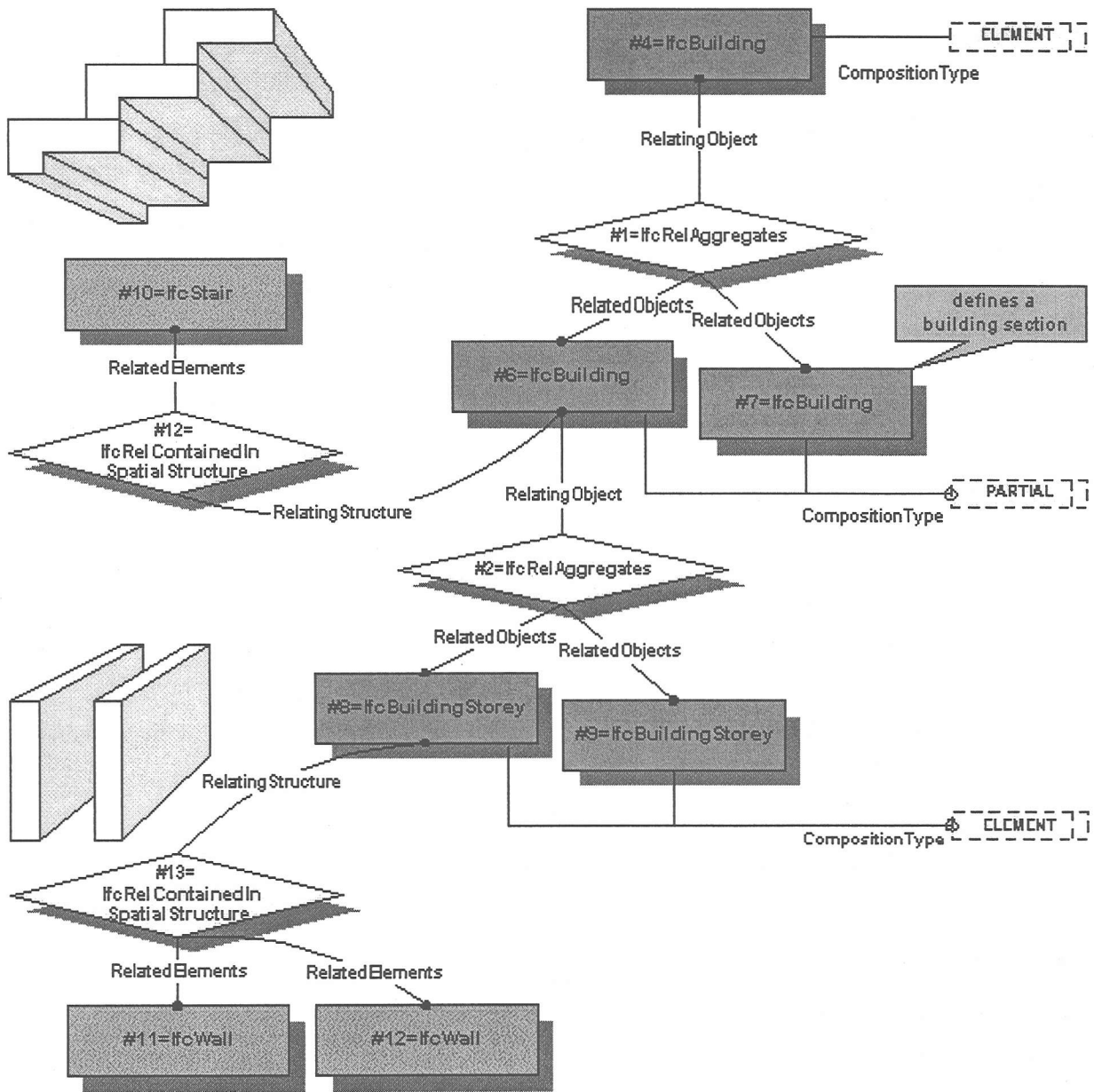


図-5 IfcRelContainedInSpatialStructure

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。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、スタールップ等)、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-6)。

3.4 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 (%), リラクゼーション値 (%), 公称断面積 (mm²), 単位重量 (kg/km) 等) の定義に関しては、図-6 に示す

専用のプロパティセット (PCableProperties) 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PCablePropertiesクラスのインスタンスとして表現される。しかし、図-6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、リラクゼーション値、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-6)。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。

4. プロダクトモデルの実装

4.1 実装に使用する言語

プロダクトモデルの実装に関しては、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でも実装に各種XMLを用いている。本論では、aecXML¹⁾、BLIS-XML (Building Lifecycle Interoperable Software)¹³⁾、ifcXML¹⁴⁾の3つのXMLを比較して論じたい。

(1) aecXML

aecXMLはIAI北米支部が中心となり、AEC/FM (Architecture Engineering and

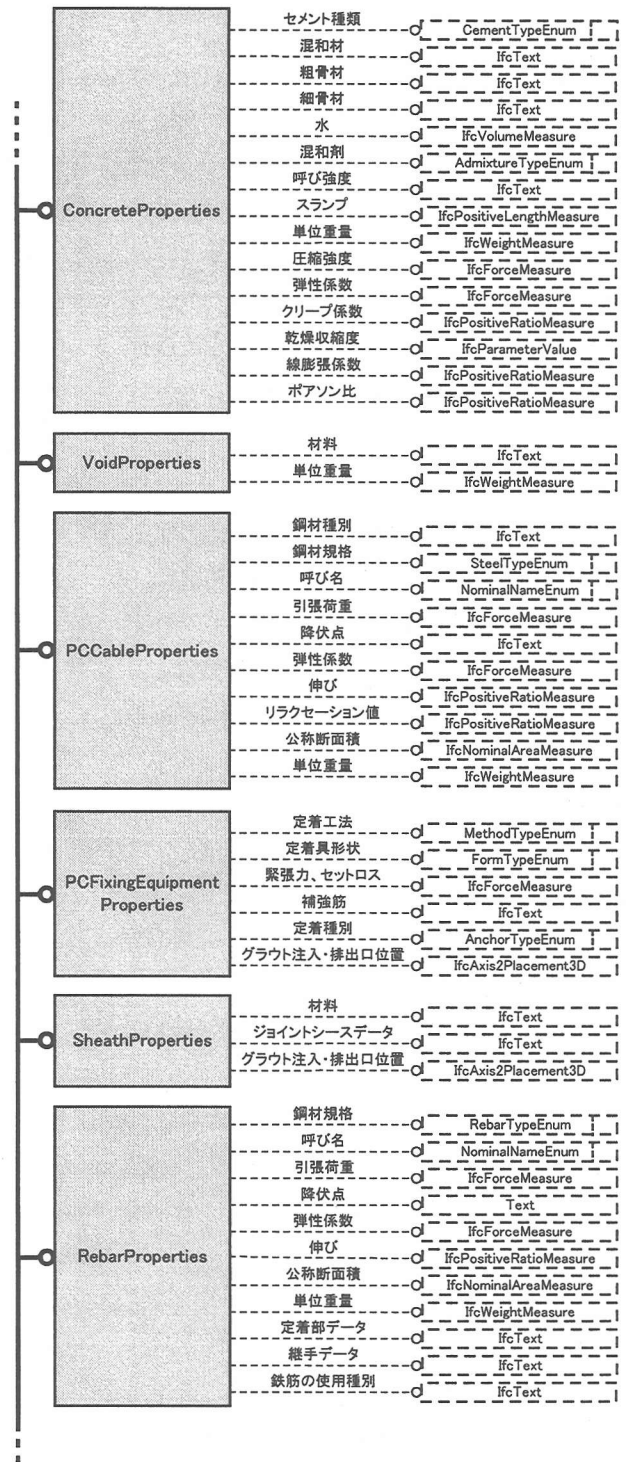


図-6 プロパティセット

Construction/Facility Management) 分野での利用を目指したXMLスキーマであり、AEC分野特有の内容を含んだビジネスメッセージの用語、文法及びレイアウトを定義している。いわば、レポートや書式のためのスキーマと言え、IFCのようなプロダクトモデルのデータ交換のためのデータフォーマットを意図しているものではないようである。但し、IFCのプロパティセットへのデータ取り込みは検討されている。

(2) BLIS-XML

BLIS-XML のスキーマには、XDR (XML Data Reduced) ¹⁵⁾ が使用されている。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。属性指向型の構造形式であり、プログラム処理が容易であることから、以前、我々は、梁、柱、接合部等の鋼部材を対象として構築したプロダクトモデルの実装に BLIS-XML を使用した。

しかし、例えば IfcProduct というクラスが持つ位置情報等の属性データは一つのクラスとして表現されるが、そのクラスの名称が IfcProduct の属性の中で表現されず、idref といった表現にとどまっている。すなわち、クラスを属性型として持つ場合、そのクラスを特定出来ないことになり、属性の型に関して、スキーマとインスタンスとの間で厳密な検証ができないという問題点がある。また、オブジェクト指向によるクラスの継承関係を表現することができないことから、クラス毎に全ての属性を再度、宣言する必要があり、煩雑である。

(3) ifcXML

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 ¹⁶⁾ が使用されている。XML Schema とは、W3C (World Wide Web Consortium) が 2001 年 5 月に勧告した比較的新しい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、BLIS-XML と同様、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、BLIS-XML とは異なり、クラスを属性型として持つ場合、どのクラスを持つのか特定可能であり、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) ¹⁷⁾ 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。

以上の理由から、本研究では、ifcXML をプロダクトモデルの実装に使用することとした。

4. 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-7、インスタンスファイルの一部を図-8と図-9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-7から容易に確認することが出来る。図-8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-9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-8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-10, 11, 12, 13に示す。

```

+ <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:extension>
- </xsd:complexContent>
- </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:extension>
- </xsd:complexContent>
- </xsd:complexType>
- <xsd:complexType name="SlabOfBridge">
- <xsd:complexContent>
- <xsd:extension base="SpatialStructureElement">
- <xsd:sequence>
- <xsd:element type="SlabTypeEnum" name="slabType" />

```

図-7 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>
  </CartesianPoint>

```

図-8 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-9 インスタンス(リレーションシップ部)

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。

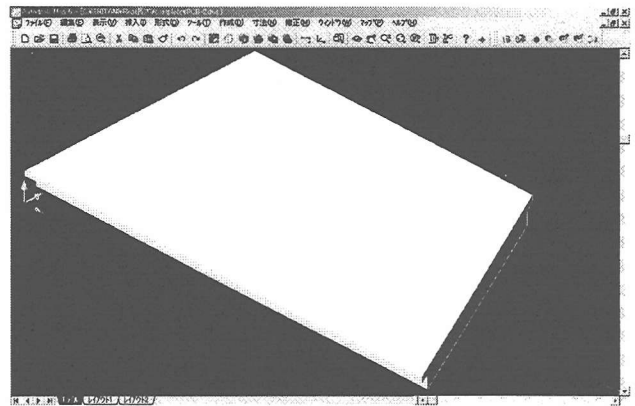


図-10 PC中空床版の3次元モデル

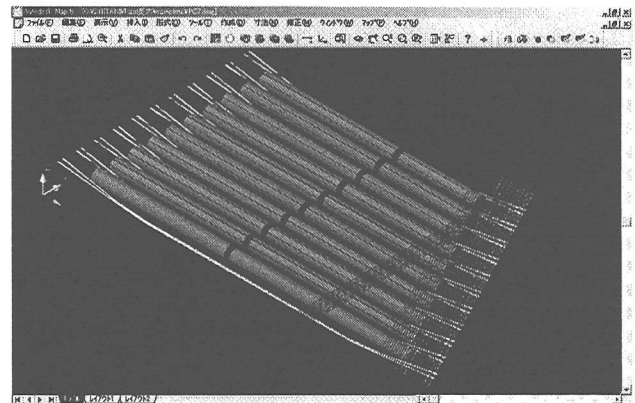


図-11 PC中空床版の内部構造

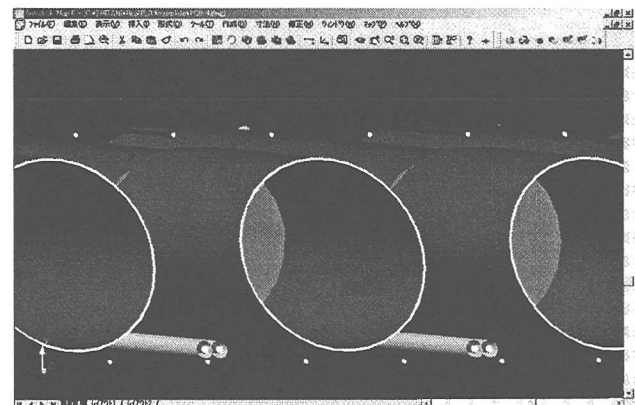


図-12 断面の様子

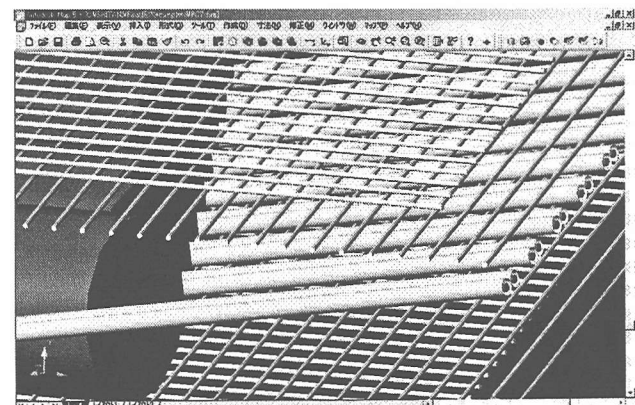


図-13 鉄筋とシースの様子

5. おわりに

本研究では、土木構造物の基本設計から詳細設計、施工設計において、3次元CAD、構造解析、設計照査、数量計算、施工計画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を目的として、PC中空床版橋を対象とした3次元プロダクトモデルを開発した。モデリングに当たっては、将来国際標準にすることを視野に入れて、IAIのIFCに着目して、IFCに準拠した形になるよう工夫した。さらに、構築したプロダクトモデルの実装のために、複数の開発言語の中からifcXMLを選定し、ifcXMLを用いて、本プロダクトモデルのスキーマとインスタンスを実装した。

本研究で開発したプロダクトモデルは、次のような特徴があると考えられる。

- ① 鉄筋、PC鋼材、ポイド等の複雑な部材を3次元で表現し、かつ、材料、強度、重量等の幅広い属性データをカバーしている。従って、3次元構造解析や照査等に対応出来るものと考えられる。
- ② 従来の、クラスの中に全ての属性データを含ませるやや古典的なモデリング手法ではなく、プロパティセットを用いることにより、モデリングが柔軟に出来るようになった。
- ③ 本プロダクトモデルは、XMLによって実装されているため、特定のアプリケーションに依存せず、他のシステムとの統合化が容易である。

今後は、3次元CADシステムと連動させるとともに、鉄筋の「かぶり」等の構造細目に関する照査システムを開発し、プロダクトモデルによるシステムの統合化に関する研究をしていきたいと考えている。

謝辞：

本研究を遂行するに当たり、社団法人プレストレスト・コンクリート建設業協会の性能発注に関する検討委員会の皆様には多大なるご協力を頂きました。また、国土交通省国土技術政策総合研究所総合技術政策研究センター建設マネジメント技術研究室の関係者、及びIAI日本支部構造分科会の皆様からは適切なご助言を頂きました。ここに感謝の意を表します。

参考文献

- 1) ISO10303-1 : Industrial Automation Systems and Integration-Product Data Representation and Exchange, Part 1:Overview and Fundamental Principles, 1994.
- 2) IAI : <http://iaiweb.lbl.gov/>
- 3) 国土交通省 : <http://www.mlit.go.jp/tec/cals/>
- 4) 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.
- 5) 三上市蔵, 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.
- 6) 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.
- 7) 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.
- 8) 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.
- 9) 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.
- 10) 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.
- 11) IFC 2x : http://cic.vtt.fi/iai/technical/ifc_2x/
- 12) aecXML : <http://www.iai-na.com/domains/aecxml.htm>
- 13) BLIS-XML : <http://www.blis-project.org/>
- 14) IfcXML : http://www.iai-international.org/iai_international/Technical_Documents/IFCXML.L.htm
- 15) XDR : <http://www.ltg.ed.ac.uk/~ht/XMLData-Reduced.htm>
- 16) XML Schema : <http://www.w3.org/>
- 17) SOAP : <http://www.w3.org/2002/ws/>