

協調作業支援のためのデータモデルの一考察

1G-10

中村 竜也、宮田 功治、横山 重俊

NTTデータ通信株式会社

1. はじめに

一人あるいは複数のユーザが異なった観点から対象を同時に操作するような作業環境においては、対象となるモデルがそれぞれの観点に沿った視覚的なあるいは操作するためのインタフェースを提供できなくてはならない。このようなモデルを実現するために、さまざまな試みがなされているが、モデル自体が複雑な構造になるため、モデルの表現が困難であることや、モデルがインタフェースに与える影響が大きいと、実用化されるには至っていない。本稿では、オブジェクト指向データベースの持つ各オブジェクトの識別子(オブジェクト識別子)を用いて依存関係を表現することにより、複数の観点から操作可能なデータベースのデータモデリングについて、建築設計モデルを例に述べる。

2. 背景と目的

産業活動においては、作業規模が大きくなるにしたがい、作業をいくつかの独立性の強い単位に分割することによって、作業単位間の相互干渉を少なくさせることにより作業の効率を維持することが可能であった。これは、ある作業の最適化が、その作業を行うための環境に閉じて可能なことにより、作業相互の干渉を抑えることが可能なためである。このような作業形態においては、他の作業に及ぼす影響が考慮され難いため、作業全体の状況判断の誤りが発生し易く、また特定の作業は効率よく行なえるものの対象の変化に弱い。これに対処するため、ある程度作業の独立性を維持し、全体の状況を判断しながら作業を行なえるような環境の開発が望まれている。そのためには、ある作業環境における情報を、別な作業環境でも理解できるかたちで保存することができ、それらを各作業環境が共有することができれば、各作業単位は他の作業の状況を把握しながら、自己の作業の独立性を維持し作業をすることが可能となる。これによって、ある作業の最適化だけが進み、他の作業がそれに従属して作業の最適化を控えるといったような、全体としてバランスのとれていない状況を抑制することができる。また、作業の対象が変化したとしても、各作業単位がその変化を吸収できる程度であれば、作業システムを変更しなくても作業効率を維持することが可能である[1]。

本研究の目的は、先に述べたような、ある作業環境における情報を別な環境でも理解できるかたちで保存でき、この情報が他の環境においても操作がで

きるような共有情報のためのデータベースのデータモデリングの手法およびそのモデルを表現/操作するためのユーザインタフェースの検討にある。

本稿の目的は、建築設計過程を例に、建築物の設計段階における意匠、構造といった異なった設計作業がおこなわれていることに着目し、異なった観点から操作が可能な建築物のモデルを検討することである。

3. 建築物のモデル化

建築物の基本設計段階においては、以下に示すような設計作業が、異なった観点からなされている[2]。

- (1) 設計 機能および意匠設計
- (2) 構造 耐震性に基づく構造解析

設計に関しては、建築物の内外観、部屋などの機能的な配置、部屋内部の機能設計がおこなわれる。構造に関しては、柱や梁、壁、床などの部材からなる構造体として建築物をモデル化し、構造の計算をおこなう。工法に関しては、対象となる建築物に、どのような工法を適用すれば、工期あるいはコスト的にも見合うものになるか検討される。このように、設計の観点はそれぞれ独立している。これらから考えて、建築物は概念的に、設計が扱うような内外観や部屋の設計に代表されるような、空間的な設計を行なう対象と、構造が扱うような、柱や壁、床といった要素から構成される構造的な設計対象に分けて考えることができる。これを具体的に階層化したものが図1である。

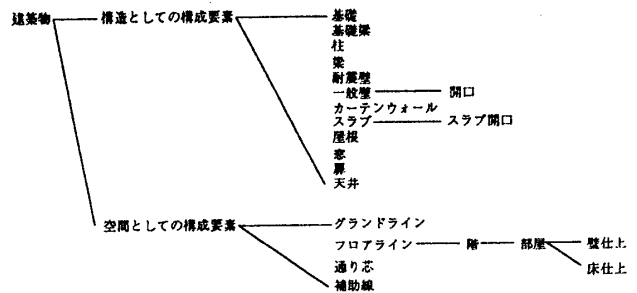


図1 建築物の階層表現

構造としての構成要素は、下位の概念に向かって建築物の各構成要素を具体化している。また、空間としての構成要素は、建築物の各構成要素の相対的な位置関係を決定するための通り芯、補助線、地上線を示すグランドライン、階の高さを示すフロアラインに細分化される。ところで、構造としての構成要素である柱などは、空間としての構成要素である通り芯によって相対的な位置関係や具体的な長さが決定される。また、構造としての構成要素によって、部屋や階といった空間としての構成要素が決定される。このように、構造あるいは空間としての構成要

素は互いに依存関係がある。図1の階層表現の建築構造としての各要素の依存関係を図2に示す。

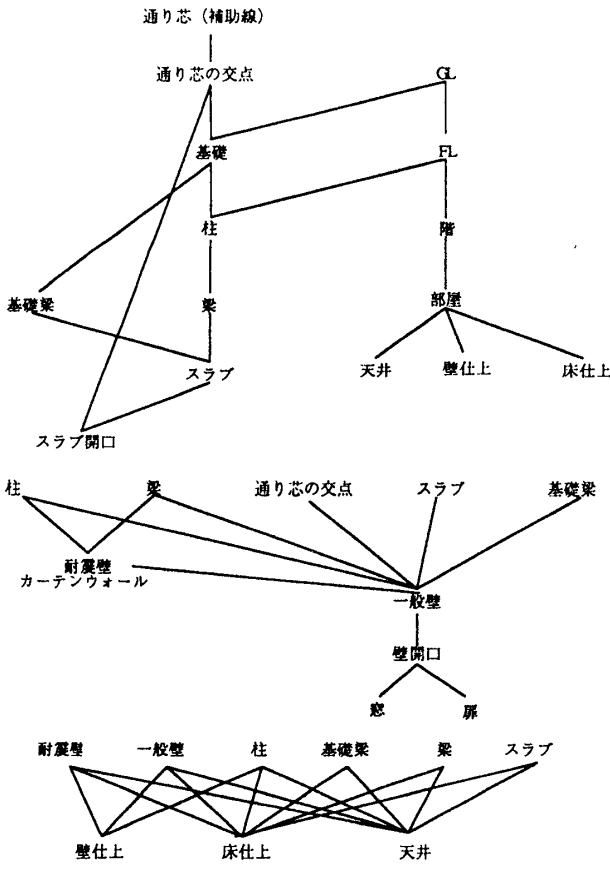


図2 建築構造における依存関係リスト

これは、図1に示したある要素を基点として、その要素を規定するための要素を上、その要素によって規定される要素を下に記述していったものである。この依存関係表現によって、意匠の観点からも構造の観点からも、建築物のモデルを操作することが可能となる。たとえば、壁を移動して部屋としての空間を広くするといった、意匠の観点からの設計変更は、図2の依存関係を辿れば、通り芯を移動することに他ならない。通り芯の変更は、これが規定するすべての構造としての構成要素を変更することであるから、構造の観点からも意匠の観点からなされた設計変更を把握することが可能となる。

4. オブジェクト識別子を用いた依存関係表現

図2の建築構造の依存関係表現を、図3に示すような、依存関係を維持するためのオブジェクト（図中c\_object）を定義し、オブジェクト指向データベースの各オブジェクトが持つオブジェクト識別子(OID: Object ID)を用いた依存関係リストによって実現する。

図1の階層表現をクラス階層として実現し、図2に表現されている各クラスをc\_objectのサブクラスとして実現する。つまり、個々の、建築物の構造あるいは空間としての構成要素は、各要素の定義的な性質と依存関係を維持するための性質を多重継承する。c\_objectには、このc\_objectを規定するためのオブジェクトを親オブジェクトのOIDの集合として、この

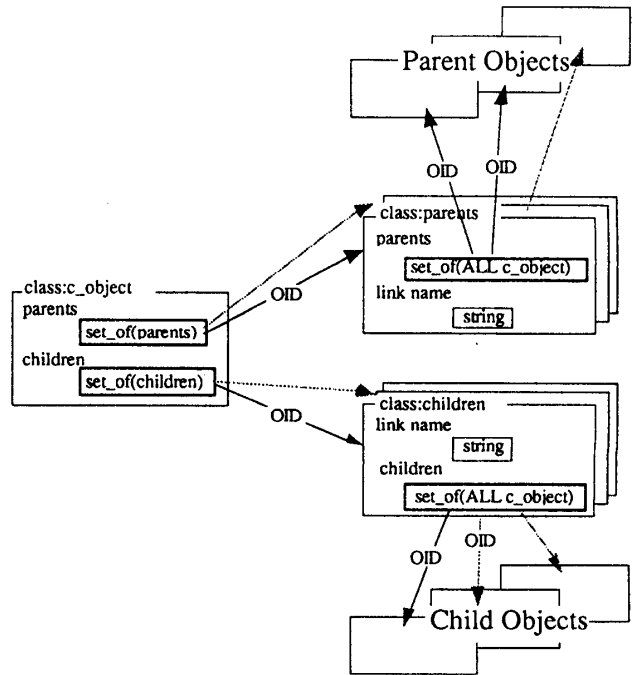


図3 オブジェクト識別子を用いた依存関係リスト

c\_objectによって規定されるオブジェクトを子オブジェクトのOIDとして持つ。親、子オブジェクトは親、子の実体としてのc\_objectのOIDを持つ。オブジェクトに対する操作は、親と子のOIDを辿ることによって伝播する。たとえば、柱を移動することは、柱自身の持つ属性の値を変えることではないので、柱の芯が基礎の芯に一致することから、柱の芯を規定している基礎を移動することになる。また、基礎も相対的な位置に関する情報を持っていないので、柱の芯が基礎の芯に一致することから、柱の芯を規定している基礎を移動することになる。また、基礎も相対的な位置に関する情報を持っていないので、依存関係を辿り、通り芯の交点を変更することになる。通り芯の交点を変更することは、通り芯を移動することに他ならない。このように、空間としての構成要素と構造としての構成要素の間の依存関係にしたがい建築物のモデルを操作することができる。そのため、意匠の観点からも構造の観点からも操作可能なモデルを実現できる。

5. まとめ

オブジェクト識別子を用いた依存関係表現により、複数の観点から操作可能なデータベースのデータモデルを、建築設計を例に述べた。

参考文献

[1] Dunald E. Carter, Barbara S. Baker, "CE Concurrent Engineering, The Product Development Environment for 1990s", Addison-Wesley Publishing Company, Inc.  
 [2] 彰国社編, "建築構造計画チェックリスト", 彰国社