

Title	整合性を考慮したUMLモデル支援環境の研究
Author(s)	馬場, 茂雄
Citation	
Issue Date	2001-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1461
Rights	
Description	Supervisor:片山 卓也, 情報科学研究科, 修士

整合性を考慮したUMLモデル支援環境の研究

馬場 茂雄

北陸先端科学技術大学院大学 情報科学研究科

2001年2月15日

キーワード: オブジェクト指向, モデル整合性, UML, 関係データベース, 計算機支援.

システムの複雑化, 大規模化に伴い, ソフトウェア開発における有効な方法としてオブジェクト指向方法論が注目されている. オブジェクト指向方法論は, 構造と操作を一体化したオブジェクトを用いて実世界の概念をモデル化する方法である. これまでに, Booch 法, OMT 法, OOSE 法など, 多くのオブジェクト指向方法論が提唱されている. オブジェクト指向方法論では, ソフトウェア開発の過程で開発段階ごとにモデルを作成する. OMG (Object Management Group) は, UML をオブジェクト指向のためのモデル表記法として標準化した. 現在, UML は広く使用されている.

UML には, 9 種類のダイアグラムが用意されている. 実際の開発では, これらのダイアグラムのいくつかを使ってモデルを記述する. このとき, 複数のダイアグラム間において矛盾が生じることがある. 複雑なシステム, および大規模なシステムの分析モデルや設計モデルは大規模で複雑になるので, ダイアグラム間の矛盾の検出はより困難になる. そこで本論文では, モデルの整合性のチェックを行なう手法を提案する. モデルの整合性チェックは, UML に与えた制約規則に基づいて行なわれる. UML の構文を表現する制約として, Well-Formedness Rules が仕様として定められているが, UML に与える制約には, 仕様としては定められていない方法論が定める制約, または, ユーザが定める制約などがある. 既存のツールには, 制約を自由にカスタマイズでき, それをチェックする機構を備えたものはない. 本論文では, 関係データベースシステム上で, モデルの情報や UML への制約記述を扱うことで, 整合性のチェックを行った. 本論文で提案する手法では, 制約を自由に定義でき, それらをチェックすることを可能にする.

UML で作成されたモデルはすべて関係データベースで管理する. モデルを関係データベースで管理するに当たって, テーブルの仕様を定めなくてはならない. UML の構造は, UML メタモデルで定義されており, 関係データベースのテーブルは, このメタモデルの論理的構造を基に定義した. メタモデル中のメタクラスごとにテーブルを用意し, テーブルの属性名はメタクラスのプロパティとする. プロパティは, メタクラスの属性とナビゲー

ション・プロパティから構成される。ナビゲーションとは、関連のあるクラスのプロパティを参照することを指し、参照先の関連終端ロール名を指定する。ナビゲーション可能な関連終端ロール名をナビゲーション・プロパティとする。メタクラス間に汎化関係を持つ場合、子のメタクラスは、親のメタクラスからプロパティを継承する。テーブルは、メタクラスのインスタンスの情報を含む組の集合として成り立つ。メタクラス単位でテーブルを作成することにより、UMLの9種類のダイアグラムに対しての扱いを統一できる。関係データベースを利用することで、関係データベースは広く普及しているため、導入が比較的容易であること、既存のシステムを利用できるので開発する手間を大幅に削減できること、実績のあるシステムを利用するので信頼性が高いシステムを構築できることなどもメリットとして享受することができる。

モデル整合条件は、UMLに対する制約であり、UMLメタモデルの不変条件としてUMLの標準制約記述言語であるOCL (Object Constraint Language) で記述する。OCLで記述された不変条件をSQLクエリーに変換することで、モデル情報が納められているテーブルへアクセスでき、その結果から不整合状態の有無を判定し、不整合箇所を特定する。OCLとSQLのクエリーの両者は、手続き的に記述するものではなく、副作用を持たないステートメントで構成されている。また、両者は、ある要素の集合に対して操作を行なうステートメントを備えている。このように、OCLとSQLとの間には類似している点が多いため、比較的容易に変換できる。本アプローチは、UMLのシンタックスの整合性をチェックする際に有効な方法である。

本研究で提案する環境では、モデル要素間の対応関係などは容易にチェックできる。本論文では、そのような制約の適用例を設定し、不整合検出の過程と結果を示した。