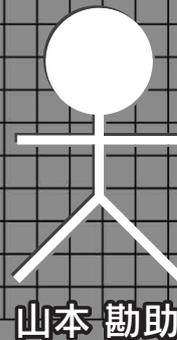


UMLはExcelで書く？ 専用ツールで書く？



山本 勘助

読者の皆さんの多くは、ワープロ・ソフトウェアのWordや表計算ソフトウェアのExcelを日常的に利用しているかと思います。これらのツールは高機能ということもあり、単なる文書作成や表計算のためのツールとしてだけでなく、さまざまな用途に利用されているのではないかと思います。

汎用ツール vs. 専用ツール

これらのツールを利用すると、図形なども含めて、表現したい内容を自由に書くことができます。特に企業における業務では広く利用されています。「誰でも持っていて、誰でも見ることができる」といっても過言ではありません。つまり、作成した内容を「見ることができない」という問題がありません。

その一方で、特別な内容を記述するための専用の“お絵描き”ツールが存在します。こうしたツールは、記述する対象に応じて専門化・最適化されています。例えば、建築物の設計のために利用されるCAD(Computer Aided Design)ツールや、複雑な計算を実行することで自然現象などを分析できる科学技術計算のツールなどがあります。

このような専用ツールは、代用することのできないものがある一方で、WordやExcelである程度代用可能と思われるものもあります。その一つの例が、UML(Unified Modeling Language)で定義された図形を描画できる「UMLツール」です。UMLそのものは徐々に広まっていますが、このUMLの記述にあたっては、WordやExcelを利用している例が少なからずあると聞いています。

ここで疑問に思うことの一つとして、「WordやExcelで代用可能と思われる専用ツールは本当に必要なのか？」ということです。CADツールや科学技術計算のツールなど、代用不可能なツールの場合は、そのツールを使わないという選択肢はありません。しかし、そうではない場合、その専用ツールを購入し利用することのメリットが分からないこともあるでしょう。特にUMLについては、ちょっとした図を描くだけであれば、WordやExcelのような汎用ツールでも十分ではないかと思われるます。

そこで、「専用ツールを利用すると何が便利なのか?」、「専用ツールを使う利点は何か?」、「価格に見合う価値はあるのか?」ということについて、具体的にツールを利用した結果やツールを利用するときの手順を比較し、検討してみます。

ここでは、ExcelとUMLツールを利用してUMLの図の作成・編集の過程を比較します。そして「UMLツールはどのよう

な場合に使うべきか?」について考えます。UMLツールとしては、ここではSparx Systems社の「Enterprise Architect」を利用しました(当然、ツールによって機能や操作が異なる。本稿の説明がすべてのUMLツールに当てはまるわけではないことに注意されたい)。

なお、比較する際の条件として、WordやExcelでも可能な表現についてUMLツールを使うとどのような差が出てくるのか、という点を中心に確認しました。例えば、ソース・コードの入出力やドキュメントの生成機能など、UMLツールならではの機能を利用するのではなく、汎用ツールでも可能な機能を対象とします。つまり、「とりあえずUMLのモデルを書いてみるなら」というコンセプトです。この点は、UMLツール本来の目的とは異なる部分もありますが、ご了承ください。

UMLについておさらい

具体的な比較に進む前に、今回利用するUMLの概要についておさらいしておきましょう。

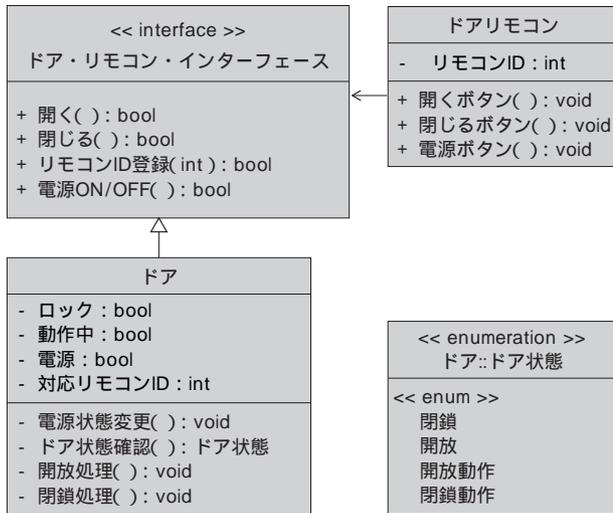
UMLはOMG(Object Management Group)が定義する、世界共通の設計内容の表現方法です。主にオブジェクト指向の設計開発で利用するための多くの図が定義されています。最新版のUML 2.1では、全部で13種類の図が定義されています。

このUMLは、ここ2、3年の間にソフトウェアの設計開発で本格的に利用され始めています。以前にもUMLが「銀の弾丸」のようにもてはやされたときもありましたが、最近ではUMLを設計の表記方法としてプロセスに採用し利用するという、しっかりした考え方で利用されるケースが増えてきました。

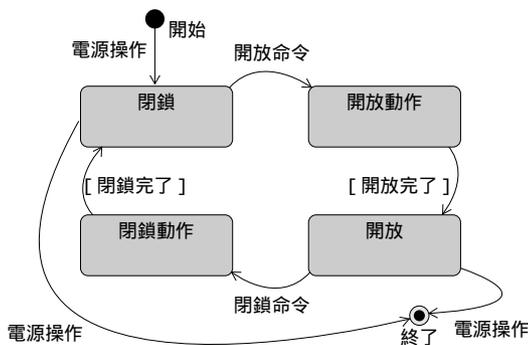
また、大学や専門学校などの授業で教えられるようになったことや、資格試験が提供され、評価の基準としても利用され始めたことなど、標準的な表記方法として使われるための土台もでき上がってきました。さらに、世界共通の表記方法ということで、会社という枠を超えた設計プロジェクトにおいて利用されることも増えてきました。

UMLでは、対象となるシステムやアプリケーションの設計を抽象化し、図として表現します。このように対象のシステムを抽象化して視覚的に表現し設計することを「モデリング」と呼びます。UMLは、「対象のシステムをモデリングするために利用する表記方法」ということになります。設計内容が図として見えるようになるので、昨今話題になっている「見える化(visualization)」の一つとしても広まっています。

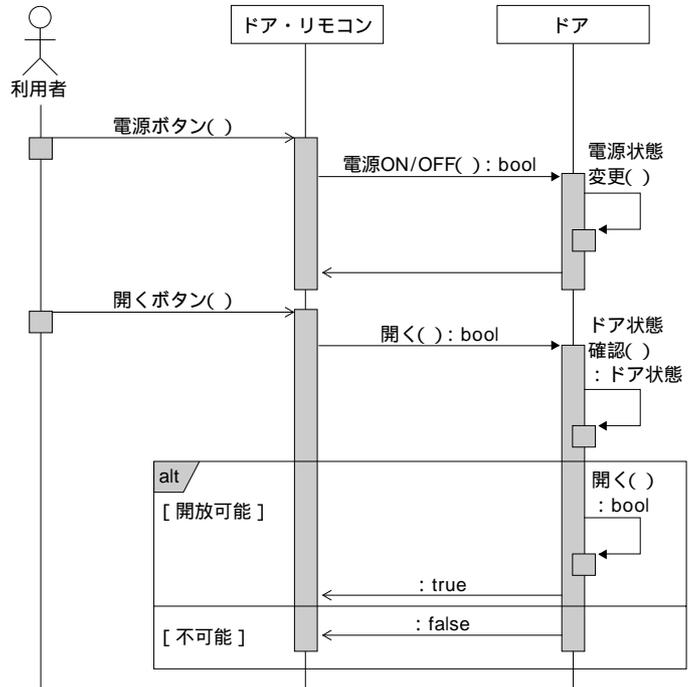
UMLを利用し、設計内容を図として表現する利点はいくつ



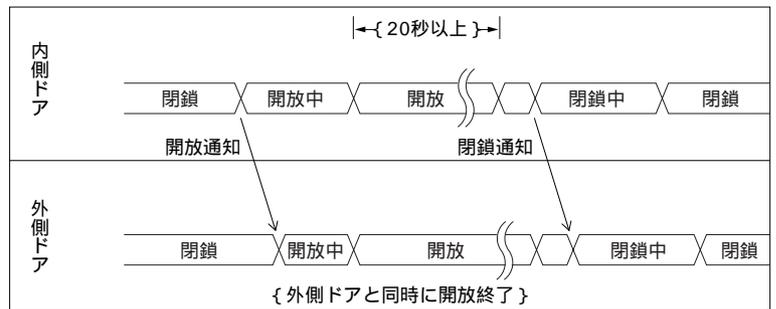
(a) クラス図



(c) ステート・マシン図



(b) シーケンス図



(d) タイミング図

図1 UMLの四つの図

(a)のクラス図では、それぞれの四角がクラスを表す。(b)のシーケンス図では矢印がメッセージであり、相互作用を示す。基本的に上から下へと順番に処理が行われる。(c)のステート・マシン図では、要素間の矢印が遷移を示す。角かっこで囲まれる内容は、遷移するための条件を示す。(d)のタイミング図では、メッセージのやり取りと状態の変化を表現する。

かあります。まず、文章で表現するよりも全体の構成が分かりやすいということです。全体的な構成を広く俯瞰できるほか、注目する対象とほかの項目との関係がすぐに分かることもあり、特に複雑な設計においては文章より理解しやすい形で表現できます。

また、文章よりあいまいな点を減らせることも利点の一つです。例えば、UMLの要素間に何らかの線があれば、何らかの関係があることを示します。逆に線がない場合は、直接の関係がないことが明確に分かります。

そのほか、このような図を仕様の理解のための基盤とすることで、多くの人がかかわる設計開発において誤解なく正しく理解する助けになります。文章で仕様を表現する場合、書き方によってはほかの人が誤読してしまう可能性があります。絵として明確に表現することで、このような誤読を減らすことができ

ます。

このような利点が最大限に生かせるのは、対象となるシステムが複雑で大規模な場合です。その結果、昨今の組み込みソフトウェアの開発において、このような利点が生かせるUMLの利用が徐々に広がっています。

組み込みシステム開発でよく利用される図は4種類、組み込みシステム開発で利用されることの多い図をいくつか紹介します。

図1(a)の「クラス図」は、システムを構成するクラスと、クラス間の関係性を表現します。オブジェクト指向のプログラミング言語を利用する場合、クラスとソース・コードには対応関係があります。例えば、クラスの属性や操作はそのままソース・コードに反映されます。この関係については図2をご覧ください。