

## WWW3D によるユーザビリティ評価システムの試作

## —状態遷移仕様の実装に関する考察—

北海学園大学 ○菊地慶仁, 北海道大学 金井 理, 岸浪建史  
要 旨

電子機器の開発初期におけるユーザインターフェース仕様の検証をソフトウェアモックアップとして行うために Web3D を用いた動作検証システムの開発を試みた。今回用いた Web3D では、状態遷移機械を作りこむことが不可能なことが判明したため、JavaScript で状態遷移機械を別途開発し、その下で Web3D モデルを実行するなど問題点の検討と対策を行った。

## 1. はじめに

電子機器類の設計初期段階では、(1)意匠的な確認が重要視され、ユーザインターフェース仕様の論理的な検証などは行われていない、(2)下流のファームウェア開発工程で設計仕様が再利用されていないため、設計初期段階で仕様を確定させることが難しい、といった問題が発生しており、その克服を目的としている。

本報告では、UML (Unified Modeling Language)[1] / Statecharts / XMI (XML Metadata Interchange)[2] を状態遷移機械の仕様に用い、Web3D によるデジタルモックアップとして実際に開発し、次の 2 点を探ることを目的として行った内容について報告する。

- 1) デジタルモックアップの作成で、自動化可能な作業、及び自動化できず手入力が必要な作業の明確化
- 2) デジタルモックアップ開発における課題の抽出

## 2. 関連技術と課題

### 2.1 利用可能性検討の進め方

今回の研究では、設計対象の機器(サンプルモデル)として本プロジェクトで開発した VWC (Visual Wireless Communicator) の仕様を用いた。また Web3D として 3DX[3]を用いた。

### 2.2 判明した課題と対策

デジタルモックアップ作成での課題として、3DX では変数を用い

たプログラマブルなスクリプティングが出来ないことが開発中に判明した。従って 3DX のモデル中に状態遷移機械を作り込むことが不可能ではあるが、予め用意しておいたアニメーションを JavaScript 中から動作させることは可能なので、まずイベントを発生させるボタンを HTML で生成し、そのボタンの押し下げ(イベント発生)によって状態遷移を行う JavaScript を生成し、その配下で 3DX にアニメーション動作をさせることで対応した。

### 3. デジタルモックアップ作成の流れ

デジタルモックアップの利用可能性検討は、以下手順で行う。図 1 にモックアップ作成の流れを示した。

- 1) 仕様書に従って MagicDraw[4]で Statecharts を描画し XMI 形式で出力する。

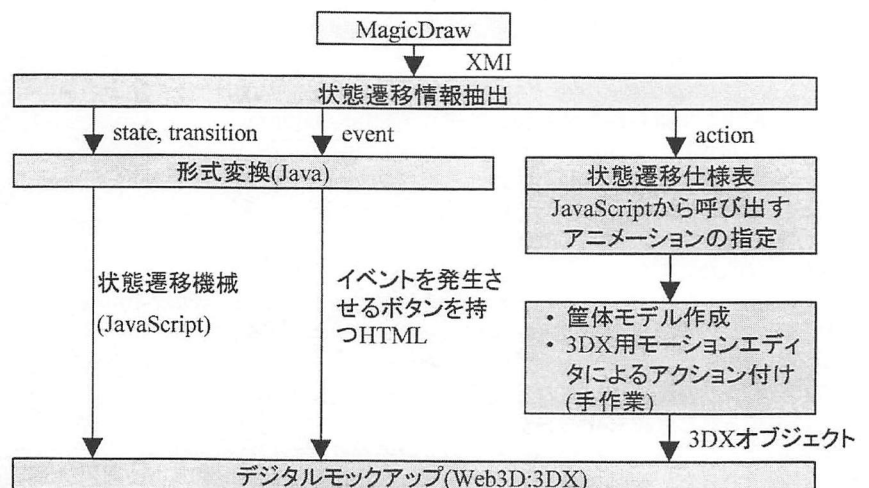


図 1 デジタルモックアップ作成の手順

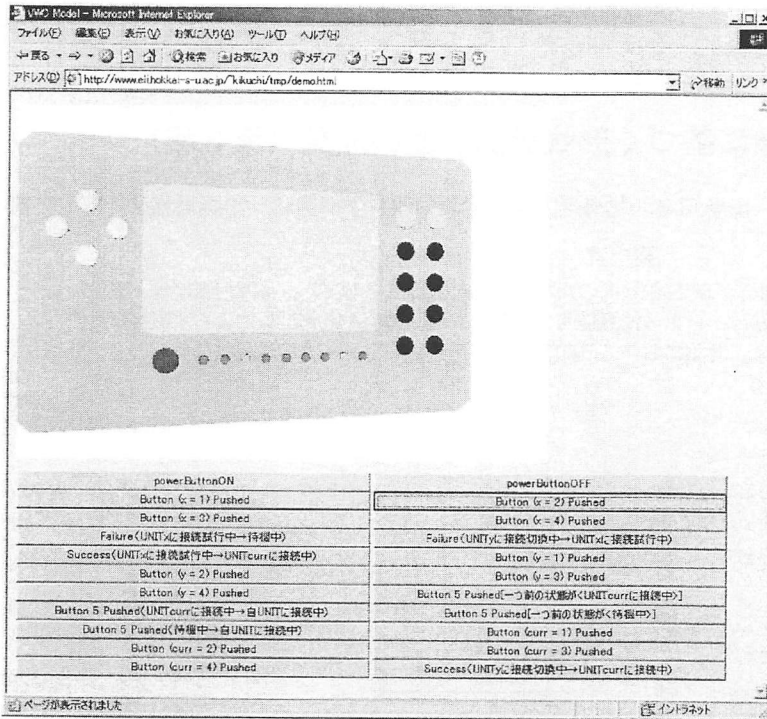


図2 デジタルモックアップの動作状況

2) 状態遷移機械に関する情報(event, state, transition, action)の抽出とアニメーション情報の追加

3) Web3D(3DX)での動作検証

図中の青い四角は自動化された作業. ピンクの四角は自動化が不可能で手作業による部分である.

始めに MagicDraw を使って StateCharts を描画し XMI 形式で出力する. XMI 中には(a)state (b)transition (c)event (d)action の定義情報と StateCharts 中の画面配置情報なども含まれているため, 状態遷移に関するオブジェクトの名称や ID などの情報を XSLT で切り出す.

上述の(c)event からは event 名に対応した名前のボタンの HTML 記述を生成する. (a)state 及び (b)transition の情報からは, 状態遷移機械として動く JavaScript を生成する. このスクリプトは HTML のボタンから渡される event ID を入力として, 内部状態に相当する番号が変化するプログラムとなっている.

(d)の action 情報に関しては, StateCharts 中ではアクション名の指定はできるが, 筐体動作や画像出力など具体的な動作内容については指定できないので, 3DX 用オーサリングソフト中のモーションエディタで別個に作り込む必要がある. また 3DX 中のアニメーション番号と XMI 出力される action の ID を対応させる必要

があるため, それらの対応関係, 及び遷移情報の一覧を仕様表としてまとめ, JavaScript から呼び出すアニメーションを指定しなければならない.

また JavaScript は単体で動作するため, ブラウザ上でアニメーション抜きで状態遷移仕様の検証ができる.

図2に動作中のデジタルモックアップの画面を示す. 画面上の筐体は, マウス操作で位置を変更することが可能であり, またイベントボタンを押すことで筐体の LED に相当する部分の色が変化する.

なお, StateCharts ではガード条件としてイベントの発生時に論理的制約を設けることができ, また並行サブ状態として複数の状態遷移が並行して表現できるが, 今回はまだサポートできていない.

#### 4. まとめ

今回は以下の報告を行った.

- 1) 3DX を対象としてデジタルモックアップの開発を行い JavaScript による状態遷移メカニズムを用いることで対応した.
- 2) StateCharts としての論理的なアクション指定と, 実際のアニメーション動作の指定の間にはギャップがあるため手作業で入力を行った. この点の克服が今後の重要な課題となる.

#### 謝辞

本報告は, 文部科学省「知的クラスター創生事業」"札幌 IT カロッセリア"中の次世代工業デザイン手法研究開発プロジェクトの成果です. ここに関係者各位及び関係者各位に謝意を表します.

#### 参考文献

- [1] Unified Modeling Language: Superstructure version 2.0, The Object Management Group, <http://www.omg.org/>
- [2] XML Metadata Interchange (XMI) Specification version 2.0, The Object Management Group, <http://www.omg.org/>
- [3] <http://www.micronetclub.co.jp/3da/3dx/main.htm>
- [4] <http://www.magicdraw.com/>