

GoldFish: JavaScript と Android NFC による実世界 GUI フレームワーク

橋本 翔[†] 増井 俊之^{††}

本論文では Android 携帯電話と NFC リーダーを用いて実世界 GUI を実装するための JavaScript フレームワークを提案する。NFC リーダーと加速度センサーを使うと、ユーザーは画面上とマウスでのメニューやスクロールバーのような様々な操作を NFC タグ上で行える。全てのプログラムは Web 上に配置された JavaScript で実装され、ユーザーは Android 携帯電話にたくさんのプログラムをインストールする必要なく、それらを利用できる。

GoldFish: Real-world GUI Framework using JavaScript and Android NFC

SHO HASHIMOTO[†] and TOSHIYUKI MASUI^{††}

We introduce a JavaScript framework for developing “Real-World GUI” using an Android phone equipped with an NFC reader. Using the NFC reader and motion sensors at the same time, users can control various parameters by moving the phone on NFC tags, just like using a mouse for controlling menus and scroll bars on a computer display. All the control programs are written in JavaScript and put on the Web as JavaScript programs, and users do not have to install different programs on their Android phones.

1. 実世界 GUI

本研究の目的は、実世界 GUI⁴⁾ を日常的に使えるようにする事である。実世界 GUI は、コンピュータのグラフィカルユーザインタフェース (GUI) に似たくみを実世界で使うという概念である。

塚田らの Ubi-Finger²⁾ は、対象を指さす事で選択し、ジェスチャーで操作するための手袋型装置を開発した。また暦本は複数のコンピュータ間でマウスのドラッグアンドドロップの様にデータをやりとりする手法³⁾を提案した。椎尾らは、マウスとバーコードリーダーを一体化させることであらゆる物に対してマウスで操作できるようにした。¹⁾

GUI には、コンピュータの画面上にゴミ箱や窓などの現実世界のメタファーを提示してユーザーに理解しやすくしている部分と、データを扱うためにドロップダウンメニューやスクロールバーなどのディスプレイならではの新しいユーザインタフェースとが組み合わせられてきている。我々の生活にはたくさんのコン

ピュータが埋め込まれているが、電子錠つきの自動ドアをキーパッドで操作して開けていたり、ラップトップ PC から目の前にあるプリンターに印刷させるのに画面上でプリンタの名前を指定して送信していたりと、操作と効果の対応づけがわかりにくく効率が悪い事がある。そうではなくむしろ、実世界でも GUI のようにメタファーによるインタフェースとデータ操作インタフェースが混在している方が使い勝手が良いのではないかと考えた。

既存の実世界 GUI の研究から、1. 対象の指定 2. ジェスチャーや GUI による操作 3. ユーザーとコンテキストの判別 4. 他のシステムとの通信 の4つの要素が重要であると考えられる。これらを利用するアプリケーションを簡単に実装でき、また日常的に使えるような仕組みを作ることが本研究の目的である。

2. GoldFish

GoldFish は実世界 GUI を実装するためのフレームワークである。Android NFC を使い、実世界の様々な場所に貼った NFC タグ (RFID タグ) を読む事でそれぞれ個別のアプリケーションを起動させる事ができる。GoldFish 上でのアプリケーションは通常の Android アプリケーションの様に Java と XML で実装して個々の端末にインストールするのではなく、JavaScript と HTML で実装して Web 上に公開し、

[†] 慶應義塾大学 政策メディア研究科

Keio University, Graduate School of Media and Governance

^{††} 慶應義塾大学 環境情報学部

Keio University, Faculty of Environment and Information Studies

GoldFish の Web サイトにその URL と NFC タグの組み合わせを登録する事で各 Android 端末から呼び出される。一般にプログラミングというものは初心者には敷居が高く、ましてや実世界を操作する電子回路や機器と連動したプログラムを書く事などは初心者には難しいが、プログラミング初心者に GoldFish を使わせたところ半日で研究室の電子錠ドアをジェスチャー入力で開閉するシステムが実装できた。

GoldFish は ubif.org⁵⁾ で公開されており、ソースコードは github で公開されている。⁷⁾

2.1 GoldFish アプリケーションの例

GoldFish アプリケーションの例として、実世界コピペを実装した。コンピュータ同士の間で Android を用いてコピーアンドペーストができるアプリケーションである。あらかじめコンピュータに NFC タグを貼り付けておき、そこに Android 端末で触れると金魚掬いの画面が表示される。(図：1) この画面で Android 端末を右に掬うと、現在最前面に表示している web ページが Android にコピーされる。別のコンピュータに触れてから左に流し込むとペーストが行われる。

実世界コピペの実装では、GoldFish アプリケーションは各コンピュータと直接通信していない。GoldFish アプリケーションおよび各コンピュータはインターネット上のクリップボードサーバーと通信する。各コンピュータにはクリップボードクライアントがインストールされていて、それぞれ貼り付けられた NFC タグの ID が実行引数に与えられている。クリップボードサーバーはサーバーアプリケーションに Ruby の Sinatra と EventMachine を、データストアに Mem-Cached を使用して実装した。クリップボードクライアントは web ブラウザ用を Google Chrome Extension で、Mac 用アプリケーションを JRuby でそれぞれ実装した。

Android 上で動作する GoldFish アプリケーションは、JavaScript の setInterval と goldfish.accelerometer 関数を用いて 10 ミリ秒毎に加速度センサーを監視する。1G 以上の加速度が 100 ミリ秒以上連続で左右どちらにかかったかによって、Android 端末をコピーとペーストどちらにジェスチャー入力したかを判別する。ジェスチャー入力値と端末 ID と NFC タグの ID は Ajax でクリップボードサーバーに送信される。goldfish.id 関数、goldfish.tag 関数がそれぞれ端末 ID と NFC タグの ID 取得に使える。各コンピュータにインストールされたクリップボードクライアントは、comet を用いてクリップボードサーバーから通信があるまで待機する。URL が送られてきた時は web ページを開き、

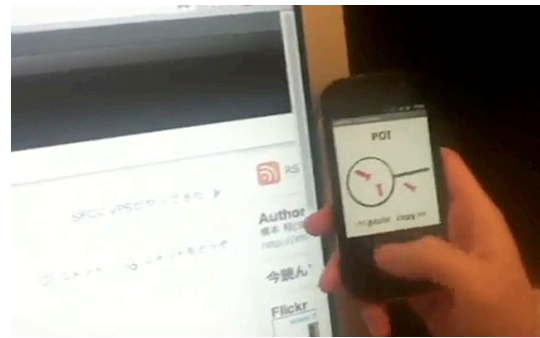


図 1 実世界コピペ

またコピー命令が送られてきた時は現在開いている URL を返信する。

実世界コピペを使うと、これまでコンピュータ間での通信では機器同士が隣にあるにも関わらずデータ送信先の名前を入力したりアイコンをクリックしたりしていた操作を、直接指示でデータ送信できるようになる。他にもプリンターにデータを送信するのではなく、データを手で掴んで投げ込むと印刷されるといった理解しやすいユーザインタフェースが実装できる。

2.2 GoldFish フレームワークの実装

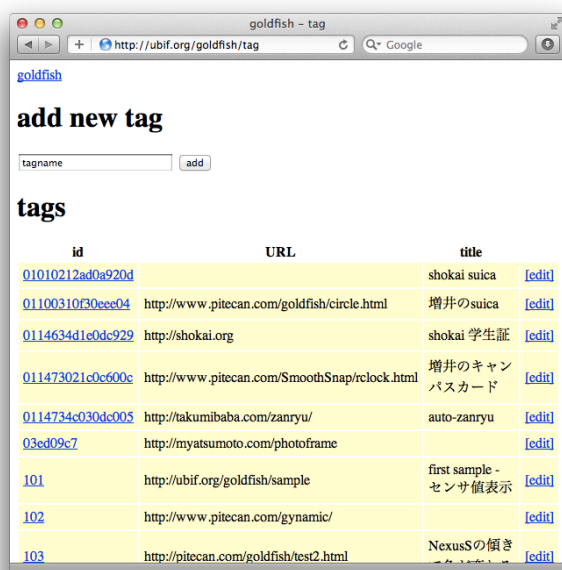
GoldFish では、アプリケーションの実装に Android ネイティブの Java ではなく JavaScript を採用している。Java はプログラム言語自体が複雑で難しく、またセンサーの値を監視しつつ画面を更新しつつ通信も行うなどの並行処理の記述がシンプルに記述できない。JavaScript はシンプルなプログラム言語で、初心者にもよく薦められている。そして関数がファーストクラスオブジェクトなのでタイマーを用いた並行処理の記述も容易である。

GoldFish は Java で実装したネイティブアプリに、WebView コンポーネントで web ページを表示している。WebView 内の JavaScript と Android ネイティブの Java が通信する事で、JavaScript からセンサーなどの機能呼び出せる。

GoldFish は NFC タグを読んだ際に、あらかじめタグに対して登録されている URL を読み込み、WebView に表示する。NFC タグは GoldFish の web サイトで登録できる。(図：2)

2.3 JavaScript によるアプリケーションの実装

アプリケーションの実装は、GoldFish のサンプルページ⁶⁾を見るとわかりやすい。Web ページを作成し、goldfish.js という JavaScript ライブラリを読み込むと Android ネイティブのセンサーや GoldFish 用の様々な機能が使用できる。1 章で挙げた実世界 GUI の実装に必要な 4 つの機能は、goldfish.tag 関数で操作

図 2 NFC タグ登録画面 <http://ubif.org/goldfish/tag>

対象の指定、goldfish.gyroscope や accelerometer などのセンサーによるジェスチャー入力、goldfish.id 関数によるユーザーの判別、goldfish.tcp や ajax によるほかのシステムとの通信の組み合わせによって実現できる。

3. その他の GoldFish アプリケーションの例

前章の実世界コピペの他にも、いくつか GoldFish を用いた実世界 GUI を紹介する。いずれも実装には 1 日かかっておらず、GoldFish を使うことで実世界 GUI を簡単に素早くプロトタイピングできる事がわかる。

3.1 ドア認証

大学の研究室の電子錠ドアを GoldFish で開閉するシステムを実装した。(図:3) Android アプリケーション側の実装は、プログラムの基礎は学んだもののあまり書いたことのない大学 3 年生でも 6 時間ほどで実装できた。

研究室の電子錠ドアに貼ってある NFC タグに Android 端末で触り、ドアノブをひねるような動きで Android 端末を回転させるとドアの鍵が開くアプリケーションである。JavaScript40 行程度で書かれた GoldFish アプリケーションは、10 ミリ秒毎にジャイロスコープを監視して角度が合計 90 度まわされると Ajax でドアサーバーに開閉のリクエストを送る。ドアサーバーは Phidgets サーボモーターと Ruby の Sinatra アプリケーションで実装されており、HTTP-POST

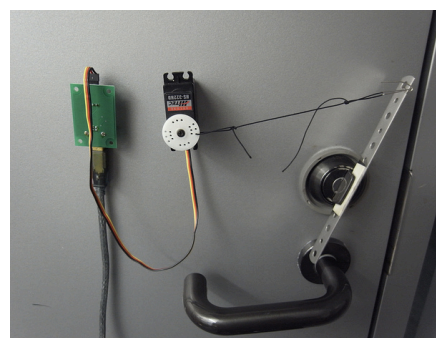


図 3 GoldFish ドアのしくみ

を受信するとドアの鍵を回して開け、5 秒後に閉める。筆者らは日常的にこの仕組みを使っている。

3.2 マウス

GoldFish の UDP/IP 関数を用いて、空中で操作するマウスとトラックパッドを実装した。机に NFC タグを貼り付けておき、その上に Android 端末を乗せると目の前の Mac を操作できるトラックパッドが現れる。Mac では JRuby で実装したマウスサーバーが起動しており、UDP で受信した座標にマウスを移動させたり、クリックさせたりという指示を実行できる。

3.3 写真立て

Android 端末を机の上の写真立てに乗せると、Flickr から取得した動物の写真がスライドショーとして表示されるアプリケーションを実装した。実装には Flickr の JavaScript API を GoldFish の Ajax から使用した。

4. 結 論

GoldFish を使うことで実世界 GUI アプリケーションを簡単に実装できる。GoldFish アプリケーションは JavaScript で記述する事ができ、また操作対象の指定、ジェスチャーや GUI による操作、ユーザーとコンテキストの判別、他のシステムとの通信を簡潔に記述できる機能が JavaScript の関数として実装されている。

参 考 文 献

- 1) 椎尾一郎, 増井俊之, 福地健太郎. FieldMouse による実世界指向インタフェース. コンピュータソフトウェア, Vol.18, No.1, pp. 28-38, January 2001.
- 2) 塚田浩二, 安村通晃: Ubi-Finger: モバイル指向ジェスチャー入力デバイスの研究, 情報処理学会論文誌, Vol.43, No.12, pp.3675-3684 (2002)
- 3) Jun Rekimoto, Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Environments, Proceedings of UIST'97, pp. 31-

39, 1997.

- 4) Toshiyuki Masui, Itiro Siio. Real-World Graphical User Interfaces. In Proceedings of the International Symposium on Handheld and Ubiquitous Computing (HUC2000), pp.72-84, September 2000.
- 5) GoldFish. <http://ubif.org>
- 6) GoldFish App Sample. <http://ubif.org/goldfish/sample>
- 7) ソースコード. <https://github.com/shokai/goldfish>