

# SPI テスト トレーニングの 안드로이드アプリの試作

著者名(日)	吉永 耕介, 安部 真人
雑誌名	神田外語大学紀要
巻	26
ページ	217-240
発行年	2014-03
URL	<a href="http://id.nii.ac.jp/1092/00001114/">http://id.nii.ac.jp/1092/00001114/</a>

# SPI テスト トレーニングの 안드로이드 アプリの試作

吉永 耕介  
安部 真人

## 1. はじめに

2013 年度から卒論の他に、卒業研究という研究プログラムが新しくでき、4 年次に翻訳、脚本、モノ作りのような作品も卒論と同様の単位が認定されることになった。外語大の学生には、ウェブデザインやプログラミングなどに興味がある学生は多数いる。その才能を発揮すれば思いの外いい作品もできると思い、2012 年度後期の研究演習で、スマートフォンのアプリの試作をしてみた。

卒論を書くときも一種の工程が必要である。まず、テーマを決め、研究資料の収集、本文を「序論、本論、結論」の順に従って構成していかねばならない。構成の順序に沿って、各部分で何を述べたいかを書いていく。そして「目次」、「参考文献」、「索引」を加えて、論文になる。

同様に卒業研究も最初は、何を作りたいのかを考索し、それに基づいて設計、作成、テストなどの工程を踏む。ウェブサイトの制作経験がある学生やない学生、又はスマートフォンに関してもユーザーのみの学生等、も一緒に教室で考え、開発、ティザイン設計をし、画面上で作動させ、不具合が生じた場合、再テストを繰り返し成果がでるまで行うことも学習の過程である。

半年間で行うゼミの活動は、SPI テストの過去問題を解く Android アプリケーションの作成を通じて、プログラミングに関する知識を取得することである。メディア教育センターの安部真人氏の全面的に協力を得て、良い試作品ができた。ゼミ生は途中で作業に戸惑う事もあったが、文科系の学生にとっては、貴重な経験だったと思う。

## 2. 試作品のテーマ

スマートフォンは、「iPhone」と「Android」端末が広くに浸透してきた。Androidとは、Googleが開発する携帯端末向けのOSの事である。その携帯の中で動くアプリケーションを作成する。

就職活動で頭がいっぱいの3、4年生は、SPIテストを練習したいという希望があった。元来SPIは本で勉強するのが正論だと思ったが、学生の発想は、電車の中でも、スマートフォンを見ながら練習できるようにしたいということで、学生の提案の方でテーマが決まった。

SPIはリクルートが提供した適性検査の方法で多くの就職試験に採用されている。現在はSPI3が2013年1月4月から提供されている。また、SPIテストには能力適性検査と性格適性検査がある。今回のSPIアプリケーションでは能力適性検査のうち、非言語能力問題（数字）の対策を行うことにした。

使用したSPIテストの設問および解答は教員が配布し、学生が図表の取り込み（スキャン）および文章の入力を行なった。

## 3. 授業の環境構築

従来プログラミング授業で使ったコンピュータ実習教室（ITLR2）で学習環境を構築した。ITLR2教室では、特定のPCにのみ開発環境を導入することはできない。また、開発を柔軟に行い、統一した環境を提供するために以下のような方針で環境を構築した。

- ・ 授業用サーバを用意しITLR2の各PCから「リモートデスクトップ接続」で授業用サーバに接続する。実行ファイルの生成など開発環境を使用する作業は「リモートデスクトップ接続」で授業用サーバに接続して行う。
- ・ ファイル共有サービス（SMB）を使用し、特別なソフトウェアを使用しないでデータへのアクセスができるようにする。
- ・ ウェブサービスを使用し、アップロードしたファイルにウェブブラウザ

からアクセスできるようにする。

## 4. 授業用サーバの準備

パソコンの Java プログラミングの授業で教員側から、授業用 Linux サーバを用意する。その上に Android 開発環境を追加し、コンピュータ実習室で開発作業ができるようにした。パソコンでスマートフォンのアプリを開発する過程を説明する前に、以下の基本システム構成をみて行く。

### 4.1 授業用サーバの OS

サーバの OS は Linux (Ubuntu Linux 12.10 LTS) を使用した。ホスト名は tom.kanda.kuis.ac.jp とした。学内用 DNS サーバに名前を登録し、学内の PC から tom.kanda.kuis.ac.jp の名前で接続できるようにした。ウィンドマネージャ (Linux 上で GUI 環境を提供するソフトウェア) として、Ubuntu Linux 12.10 標準装備の Unity を使用し、GUI 機能を有効にした。

### 4.2 開発環境

プログラミング言語 Java の開発環境である Eclipse に Android アプリケーション開発ツールである Android SDK Tools を導入した。Android アプリケーションの開発は、通常 Eclipse に Android 開発用プラグイン (Plugin) ソフトウェア (ADT Plugin) を導入する形で行う。また Android アプリケーションの開発を容易にするため PhoneGap および Chromium を使用した。

### 4.3 ページのマークアップ言語

スマートフォンサイトは PC サイトの制作に近いので、HTML、CSS、JavaScript など編集の知識が必要となる。

マークアップ言語 (Markup Language) とは、文章の構造や見栄えを指定

する文字列 (Tag: タグ) を使用する人工言語である。HTML や XML ではタグを識別する文字列として </> (山括弧) を使用するが、{ } (波括弧) を使用するマークアップ言語が存在する。

- ・ HTML(Hyper Text Markup Language) とはウェブ上のドキュメントを記述するためのマークアップ言語である。最近の HTML では、文章の構造のみ記述し、見栄えについては CSS で指定する事が一般的である。

- ・ CSS (Cascading Style Sheet) とは XML や HTML などのタグで囲まれた部分 (タグ要素) をどのように修飾 (表示) するか指定する仕様である。CSS を使用することでウェブドキュメントの表現と修飾を分離することができる。これによって、複数の HTML ファイル内同一の CSS を指定し、同じ見栄え (修飾) のウェブドキュメントを複数作成する事ができる。

- ・ JavaScript とは主にウェブブラウザで使用されるスクリプト言語の一種である。スクリプト言語とは、プログラミング言語の一種であり、文字列で表現したプログラムをそのままプログラムとして解釈する人工言語である。

JavaScript は Netscape Navigator 2.0 で初めて実装され、その後ウェブブラウザで広く実装されている。JavaScript を使用することで動的なウェブサイトを構築することができる。最近ではウェブサーバ内で JavaScript を使用する (Server Side JavaScript) 事例が増えている。

## 5. おおまかな仕様の検討

まず、学生に SPI テストの過去問題を渡し、作成しようとしている Android アプリケーションで何が必要かを考えさせた。但し、できるだけ簡素にし、最低限必要な機能は何かを考えることを指導した。学生からの回答をまとめると以下のようなものになった。

- ・ 選択肢は常に 4 問
- ・ 回答は常に 1 答

・表示させたい内容は、問題文（画像を含む）、選択肢（4問）、解説である  
学生の回答を基に、どのようなページが必要か学生に回答させた。学生の  
回答をまとめると以下の通り。なお、見易さのためデザイン（参考）を行っ  
たものを表示する。

HTML ファイルは、index.html、start.html、answer\_ok.html、answer\_ng.html、  
end.html の5つのファイルを作成する。次にデザイン管理するファイルであ  
る andriod.css、を1つが作成し、設問データのまとめファイル master.xml を  
1つ作成し、全体動作を制御する javascript ファイルの andriod.js を作成する。

## 5.1 Chrome で表示され画面

### ①スタートページ (index.html)



図 5.1 スタートページの画面

問題を開始する。

「start!」と「終わり」のリンク

### ②設問のページ (qanda.html)

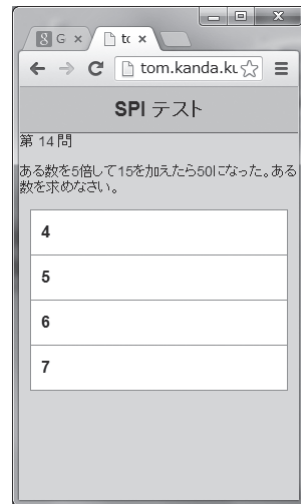


図 5.2 設問のページの画面

設問の内容と4箇所の選択肢

③正解した場合のページ (answer\_ok.html)

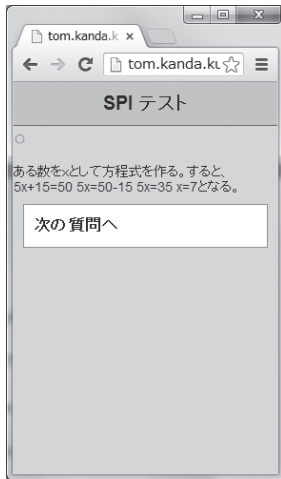


図 5.3 正解のページの画面  
「○」と「次の設問へ」のリンク

④間違えた場合のページ (answer\_ng.html)

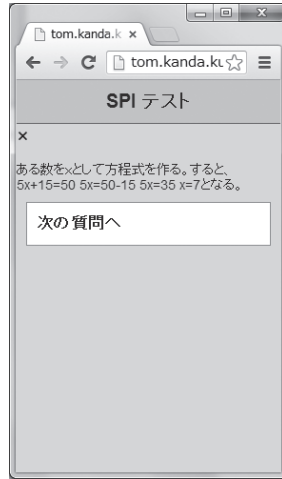


図 5.4 不正解のページの画面  
「×」と解説、「次の設問へ」のリンク

⑤終了のページ (end.html)



図 5.5 終了のページの画面

テストの結果（「点数」、「正解数」「不正解数」と「スタートページに戻る」）がリンク表示された。

学生の回答を基に、操作の流れがわかる最小限度のウェブページファイルを作成できた。

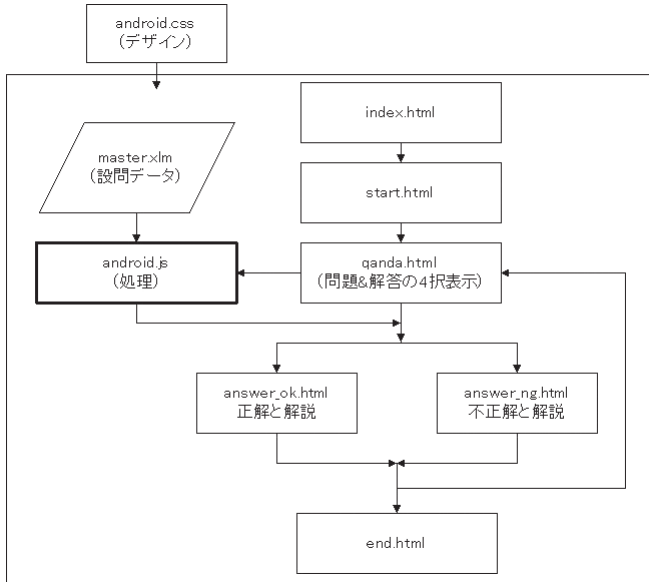
5.2 プログラム全体の動作の流れ<sup>[3]</sup>

図 5.6 プログラムの流れ図

html ファイルの書き方は、一般のウェブサイトと変わらない。最初に index.html からアクセスする。次に、start.html から、スタートか、エンドか 選択画面を表示する。ユーザーによりスタートをクリックする場合、qanda.html で設問を出し、4 択回答を表示し、答えをクリックしたとき、正解した場合は answer\_ok.html へ、不正解の場合は answer\_ng.html へ、解説の表示と次への質問のローテーションの制御は、JavaScript プログラムの android.js (作者:安部真人)で行う。流れは図で表示し、ソースファイルは付録で参考する。



## 6. 設問データの入力についての検討

まず、学生にどのようなデータが必要か意見をだしてもらった。まとめると以下の通りになった。

### 6.1 設問データの構造

- ・ 設問データ全体に関すること
  - 1問正解すると得られる得点の点数 (score)  
各設問に個別の点数を設定するのではなく、全問題共通にした。
- ・ 各設問データに関すること
  - 何番目の問題か (id)
  - 問題文 (question)  
学生がスキャンした画像を含めることができるように配慮する
  - 選択肢 (item)  
選択肢の番号 (id) と 選択肢の文字列 (text) を含む、選択肢の番号は数字のみ指定
  - 正解 (answer)  
正解の選択肢 (id) と 正解の文章 (text) を含む、正解の選択肢は選択肢に指定した番号を指定する。
  - 解説 (explain)  
設問について解説、学生がスキャンした画像を含めることができるようには配慮する。
  - 重み (weight)  
設問の難易度、0 が最も簡単で、数値 (自然数) が上がると難易度が高いことを示す。

問題データの構造を階層化すると以下のようになる

### タグ構造

```

xml
|-- display: 設問データを格納するタグ
    |-- question: 質問の文章
        | |-- {img/sup}: 画像の表示
        |
    |-- item: 選択肢を表示するタグ。4つで固定
        | |-- id: 選択肢の id (数字)
        | |-- text: 各選択肢の文章
        |     |-- {img/sup}: 画像の表示
        |
    |-- answer: 正解の情報を格納するタグ
        | |-- id: 正解の選択肢の番号 :item/id のいずれかの値を指定
        | |-- text: 正解の場合の文章 (一応指定)
        |
    |-- explain: 解説の文章
        | |-- {img/sup}: 画像の表示
        |
    |-- weight: 設問の重み付け。0 が最も簡単。数字
    
```

図 6.1 タグの構造

設問データは、拡張性を考慮して xml で記述することとした。学生には各自 1 問毎に設問データ (xml ファイル) を作成させた。学生が作らせた設問データに任意の順位番号 (q-xx.html : xx は任意の数字) を付けた。授業

支援者が設問データをまとめて1つのマスターデータ (xml ファイル) を作成するツールを作った。学生がスキャンした画像データは各設問を保存しているフォルダに、関連する問題の番号付けて (question-xx.jpg もしくは xx-question.jpg : xx は任意の数字) 保存した。また、最後に毎回ユーザーがトレーニングをした score (各設問ごとの加点数) が表示される。

## 6.2 設問データのまとめ master.xml

xml では、通常、文字コードは utf-8 を指定する。なお、WindowsXP に付属している「メモ帳」では文字コードとして utf-8 を指定することができない。この場合、ITLR 実習室の各 PC にインストールされている TerPad のような、文字コードを指定できる高性能テキストエディタを使用する必要がある。

学生が作成した約 20 問の設問データ xml ファイルを集めて、master.xml に作成した。解説のため、以下は設問 14 のみのソースファイルを示す。

[master.xml]

---

```
<?xml version='1.0' encoding='UTF-8' standalone='yes' ?>
<?xml-stylesheet type="text/xsl" href="master.xsl" ?>
<xml>
...
<display>
  <answer>
    <id>4</id>
    <text>7</text>
  </answer>
  <explain> ある数を x として方程式を作る。すると、 $5x+15=50$   $5x=50-15$ 
```

5x=35 x=7 となる。 </explain>

<item>

<id>1</id>

<text>4</text>

</item>

<item>

<id>2</id>

<text>5</text>

</item>

<item>

<id>3</id>

<text>6</text>

</item>

<item>

<id>4</id>

<text>7</text>

</item>

<question> ある数を 5 倍して 15 を加えたら 50 になった。ある数を求めなさい。 </question>

<weight>1</weight>

</display>

...

<score>10</score>

</xml>

## 7. おわりに

以上ゼミで作成したプログラムが Eclipse、phonegap、Chrome、または実機で画面を表示するという結果が得られた。あくまで、これはゼミで作り上げた制作のプロセスである。目的は、学生が実際の制作環境を知って、自分はその作成パターンに沿って自分のオリジナル製品ができることを実感するのであって、作った制作品を見せることではない。

このプログラミングの試作を始めた頃は全体を把握する理解力に乏しい学生もいて困難が多かった。最初、プログラム作成、デザイン、資料収集、入力作業といういくつかの組に分けた。ゼミ生の半分以上は4年生、就職活動のため、欠席が多かったので、時間的に、集中して作業ができる学生数が少なく、最初からサーバの構築からの難しい説明は、なかなか理解し難く、後半作業の進行は、なかなか進まない状況であった。

作業中によくある間違いを整理すると、以下のようである。

- ・行のはじめに - (ハイフン) を入力してしまう。

XML ファイルを示させた際に、タグの展開を行うと、行のはじめに - が表示されるため、入力する際にも - を入力してしまうと考えられる。

- ・文字コードが Shift JIS になる。

これは Windows 7 に付属するメモ帳を使用しているためと考えられる。

- ・XML 宣言を記述していない。

間違いではないが、記述するように心かけるべきである。

- ・どのタグもインデントにつけずに記述する。

そのため、タグの構造を間違えることが多い。

- ・「構造化されたデータ構造」というものが理解しにくい。

勿論ゼミ生は全員プログラミング授業を履修したわけでもないが、編集作業は無理である。卒業研究は、このような程度は話ならないとか、あくまで最初の試しである。今回は初めての試みではあったが、今後何回もテスト

を繰り返す必要がある。しかしながらいつか外語大の中から良い卒業研究作品が出るように期待している。

[ 付録 ]<sup>[1],[2],[7],[10],[11],[14]</sup>

[android.js]

---

```
/*
    android.js
    (c) abe@kanda.kuis.ac.jp
*/

var hist = [];
var cur = 0;
var curLevel = 0;
var atari = 0;
var hazure = 0;
var startUrl = 'start.html';
var questionUrl = 'question.html';
var okUrl = 'answer_ok.html';
var ngUrl = 'answer_ng.html';
var endUrl = 'end.html';
var masterXml = './master.xml';
var objXml;

function loadPage(url, func) {
    if (url === undefined) {url = startUrl;}
}
```

```
        $("body").load(url, null, func);  
    }
```

```
function start_q() {  
    loadPage(questionUrl);  
    $.ajax({  
        type: "GET",  
        async: true,  
        url: masterXml,  
        datatype: "xml",  
        success: setXml,  
        error: logXml,  
        timeout: 5000,  
        isLocal: true  
    });  
}
```

```
/* for start_q() use */  
function setXml(xml) {  
    var str = "";  
    var olddiv;  
    var newdiv;  
    var parentNode;  
    if(xml === undefined) {  
        xml = $(this);  
    }  
}
```

```
objXml = xml;
// get current index
load_cur();
str = $("display:eq(" + cur + ") question", xml).text();
newdiv = document.createElement('div');
newdiv.id = 'question';
newdiv.innerHTML = str;
olddiv = document.getElementById('question');
if(olddiv === null) {
    document.body.appendChild(newdiv);
}
else{
    if(olddiv.parentNode == null){
        document.body.appendChild(newdiv);
    }
    else {
        parentNode = olddiv.parentNode;
        parentNode.replaceChild(newdiv, olddiv);
    }
}

str = ' 第 ' + ( Number(cur) + 1 ) + ' 問 ';
console.log(str);
newdiv = document.createElement('div');
newdiv.id = 'title';
newdiv.innerHTML = str;
```



```

    olddiv = document.getElementById('title');
    if(olddiv === null) {
        console.log("#title not found");
        document.body.appendChild(newdiv);
    }
    else {
        if(olddiv.parentNode === null){
            document.body.appendChild(newdiv);
        }
        else {
            parentNode = olddiv.parentNode;
            parentNode.replaceChild(newdiv, olddiv);
        }
    }
    for(var i=0; i<4; i++){
        str = '<a href="#" onclick="onClick(' + $(("display:eq(" + cur +
") item:eq(" + i + ") id"), xml).text() + ")>'
            + $(("display:eq(" + cur + ") item:eq(" + i + ") text"),
xml).text() + "</a>";
        console.log(str);
        $(("#choice ul li a:eq(" + i + ")"), document.body).
replaceWith(str).end();
    }
}
/* for end_q() use */
function getXml(xml){

```

```
if(xml==undefined){
    xml = $(this);
}
objXml = xml;
}
function logXml(xhr, status, errorThrown) {
    var logstr = "";
    logstr = "load failed : " + status + " / status : " + xhr.status;
    logstr += "(" + xhr.statusText + ")";
    console.log(logstr);
}
function onClick(num) {
    var targetPage = "";
    var right_num = $("display:eq(" + cur + ") answer id"), objXml).text();
    console.log("right answer: " + right_num + " / selected answer : " + num);
    var str = "";
    if(num == right_num){
        targetPage = okUrl;
        curLevel = curLevel + 1;
        atari = atari + 1;
    }
    else {
        targetPage = ngUrl;
        hazure = hazure + 1;
    }
    save_cur();
}
```

```
        loadPage(targetPage, explain);
    }
function explain() {
    var str = "";
    var olddiv;
    var newdiv;
    var parentNode;
    load_cur();
    str = $("display:eq(" + cur + ") explain").objXml().text();
    newdiv = document.createElement('div');
    newdiv.id = 'explain';
    newdiv.innerHTML = str;
    olddiv = document.getElementById('explain');
    if(olddiv === null){
        document.body.appendChild(newdiv);
    }
    else {
        if(olddiv.parentNode === null){
            document.body.appendChild(newdiv);
        }
        else {
            parentNode = olddiv.parentNode;
            parentNode.replaceChild(newdiv, olddiv);
        }
    }
}
```

```
function next_q() {
    var size;
    var targetPage = "";
    cur = cur + 1;
    save_cur();
    size = $("display", objXml).size();
    if(cur >= size){
        loadPage(endUrl);
    }
    else {
        loadPage(questionUrl, (function(){ setXml(objXml); } ));
    }
}
function end_q() {
    if(objXml === undefined) {
        $.ajax({
            type: "GET",
            async: false,
            url: masterXml,
            datatype: "xml",
            success: getXml,
            error: logXml,
            timeout: 5000,
            isLocal: true
        });
    }
}
```

```
var goukei = $('score', objXml).text();
load_cur();
$("#goukei", document.body).text(goukei * atari).end();
$("#atari", document.body).text(atari).end();
$("#hazure", document.body).text(hazure).end();
if( typeof sessionStorage !== 'undefined' ){
    sessionStorage.clear();
}
else {
    console.log("end_q() : sessionStorage cannot use");
}
}
function save_cur(){
    if( typeof sessionStorage !== 'undefined' ){
        if(!sessionStorage.version ){
            sessionStorage.version = '1.0';
            sessionStorage.cur = cur;
            sessionStorage.curLevel = curLevel;
            sessionStorage.atari = atari;
            sessionStorage.hazure = hazure;
        }
        else {
            if( sessionStorage.version === '1.0' ){
                sessionStorage.setItem('cur',cur);
                sessionStorage.setItem('curLevel', curLevel);
                sessionStorage.setItem('atari', atari);
```

```

        sessionStorage.setItem('hazure', hazure);
    }
}
}
else {
    console.log( "save_cur() : sessionStorage cannot use" );
}
console.log( "save_cur() : current " + cur + " / curLevel " + curLevel );
}
function load_cur(){
    if( typeof sessionStorage !== 'undefined' ){
        if(!sessionStorage.version){
            sessionStorage.version = '1.0';
            sessionStorage.cur = 0;
            sessionStorage.curLevel = 0;
            sessionStorage.atari = 0;
            sessionStorage.hazure = 0;
            cur = 0;
            curLevel = 0;
            atari = 0;
            hazure = 0;
        }
        else {
            if( sessionStorage.version === '1.0' ){
                cur = Number(sessionStorage.getItem('cur'));
                curLevel = Number(sessionStorage.

```

```
getItem('curLevel'));

                                atari = Number(sessionStorage.
getItem('atari'));

                                hazure = Number(sessionStorage.
getItem('hazure'));

                                }
                                else {
                                    cur = 0;
                                    curLevel = 0;
                                    atari = 0;
                                    hazure = 0;
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

else {
    console.log( "load_cur() : sessionStorage cannot use" );
    cur = 0;
}

console.log( "load_cur() : current " + cur + " / curLevel " + curLevel );
}

-----
```

## [参考文献]

- [1] 井上誠一郎等：パーフェクト JavaScript, 技術評論社出版,(2011)
- [2] 沖林正紀：jQuery 本格入門, 技術評論社発行,(2012)
- [3] 大滝みや子：大滝みや子先生のアルゴリズム解法 第2版, リックテレコム出版,(2009)
- [4] 猿橋大：詳細 CSS スタイルシート辞典 第2版, 秀和システム出版,(2010)
- [5] 掌田津耶乃：Eclipse ではじめる Android プログラミング入門 —SDK2.3/4.2 対応, 秀和システム出版,(2013)
- [6] たにぐち まこと：iphone+android スマートフォンサイト制作入門, アスキー・メディアワークス出版,(2011)
- [7] 高津戸壮：Web デザイナーのための jQuery 入門, 社技術評論出版,(2011)
- [8] 富田宏昭：HTML5/JavaScript と PhoneGap で作る iPhone アプリ開発入門, 株式会社マイナビ出版,(2012)
- [9] 西畑一馬・鍋坂理恵：jQueryMobile スマートフォンサイトデザイン入門, アスキー・メディアワークス出版,(2012)
- [10] 西畑一馬等：jQuery 逆引きマニュアル, 株式会社インプレスジャパン出版,(2010)
- [11] 古籾一浩：JavaScript 逆引きハンドブック, シーアンドアール研究所出版,(2012)
- [12] Eric van der Vlist：XML Schema (日本語版), オライリー・ジャパン出版,(2003)
- [13] Erik T. Ray：入門 XML (日本語版), オライリー・ジャパン出版,(2004)
- [14] jQuery Community Experts:jQuery クックブック (日本語版), オライリー・ジャパン出版(2010)
- [15] Michael Fitzgerald：XML Hacks (日本語版), オライリー・ジャパン出版,(2005)



- [16] Project KySS, ビスケット株式会社 : XSLT+XPath 実践マスター , ソフトバンクパブリッシング出版 ,(2002)
- [17] Steven Holzner : XSLT 実践ガイド , 株式会社アスキー出版 ,(2002)