

センサネットワークにおける大規模な可視化システムの開発

中根 傑[†] 江田 政聡[†] 横山 昌平[†] 福田 直樹[†] 石川 博[†]

[†] 静岡大学情報学部情報科学科 〒 432-8011 静岡県浜松市中区城北 3-5-1

E-mail: †{cs06062,cs06015}@s.inf.shizuoka.ac.jp, ††{yokoyama,fukuta,ishikawa}@.inf.shizuoka.ac.jp

あらまし 近年，家庭などの消費電力を可視化することによって省電力化につながるということが言われている．本研究では，センサネットワークを利用することにより，消費電力だけではなく人間行動も可視化し，無駄を発見し易くすることで，より省電力化につなげるための可視化システムを開発する．また，その可視化システムは，例えば建物内のセンサネットワークを可視化する際に，各階のセンシングデータを閲覧できるだけでなく全ての階のセンシングデータを一度に閲覧できるようにし，人が全体のセンシングデータを俯瞰できるような大規模な可視化システムを目指す．

キーワード 可視化，センシングデータ

Development of a Large-scale Visualization System Based on Sensor Network

Takashi NAKANE[†], Masaaki EDA[†], Shohei YOKOYAMA[†], Naoki FUKUTA[†], and Hiroshi ISHIKAWA[†]

[†] Department of Computer Science, Faculty of Informatics, Shizuoka University Johoku 4-5-6, Nakaku, Hamamatsu-shi, Shizuoka, 432-8011 Japan

E-mail: †{cs06062,cs06015}@s.inf.shizuoka.ac.jp, ††{yokoyama,fukuta,ishikawa}@.inf.shizuoka.ac.jp

Abstract Recently, it is said to connect with the power saving by making power consumption at home etc. visible. In the resaerch, the visualization system to tie to the power saving or more is developed by making not only power consumption but also the man action visible, and making uselessness easy to discover by using the sensor network. Moreover, the visualization system ,for example when sensor network in building visualized,not only browsing sensing data with other floor but also browsing all floor at a time,and to aim at a large-scale visualization system by which the person can look down upon the entire Sensing data

Key words Visualization , Sensing Data

1. はじめに

近年，さまざまなセンサが普及し，家電製品などでも用いられるようになってきた．センサの例としては，モーションセンサや照度センサ，扉開閉センサなどがあり，それらのセンサからセンシングデータを取得するための環境としてセンサネットワークの研究が盛んに行われている．センサネットワークは，複数のセンサを無線ネットワークに繋げて，センシングデータの取得や，センサ制御などを行う．

また，ネット回線のインフラなどが整ったことで，インターネットの通信速度が年々速くなってきている．そのため，Google Maps [1] のような Ajax (Asynchronous JavaScript + XML) [2] を用いて Web ブラウザだけで使えるアプリケーションが年々増加の傾向にある．Ajax とは，Web ブラウザ内での非同期通信とそれを用いてのインターフェイスの構築を行うことの総称である．

また，Ajax とともによく挙げられるものが JSON (JavaScript Object Notation) [3] である．JSON とは近年 Web 上でよく使われる軽量な半構造データ記述フォーマットである．JSON は JavaScript の文法に従っているため Ajax とともによく使用される．

本研究では，センサネットワークと Web に着目し，センサネットワークから得られるセンシングデータを Web ブラウザから利用できるようにすることで，インターネットにアクセスできる誰もが手軽にセンサの状態やセンサ間の関係を見出し，利用するための可視化手法を提案する．

また可視化するセンシングデータの取得方法には 2 種類あり、一つ目は DBMS に格納しておいたセンシングデータに PHP や CGI などによって Web ページ側からアクセスする方法で、もう一方はセンシングデータを直接ファイルに書き出し、Web ページがそれを読み込む方法である．この 2 種類の方法を適切に使

い分けることによって、よりリアルタイム性を考慮した可視化システムを開発する。

2. 関連研究

センサネットワークを利用した研究はさまざまな分野で行われている [4][5]。

高田ら [4] は、センサネットワークから得られる時系列センシングデータのピークとボトムを算出することでセンサノードの近接性を発見する手法を提案した。

また、本田ら [5] は、赤外線センサの発火系列を用いて、ID なしで複数人を追跡するシステムを構築した。

Web 上でよく使われるデータ形式として JSON を前述したが、センシングデータを JSON 形式で記述することはあまり実用的ではない。JSON はデータを DOM 木で管理するので、大量のセンサボードから取得するセンシングデータを全て DOM 木で管理する。そのため、アプリケーション側で適切に管理しなければ、Web ブラウザで使用するメモリが爆発的に増加する。そこでメモリ上で展開する JSON の欠点である高メモリ負荷を補完する研究として Freddy[6] が挙げられる。JSON が DOM と類似した手法であることに対して、Freddy は SAX パーサと類似した手法である。また、大量なセンサノードを使用したセンサネットワークから得られるセンシングデータは大量なデータになることが容易に想像できるため、複数のサーバに溜められたセンサネットワークのセンシングデータを扱うことにも適しているデータ形式とも言える。

3. 提案手法

本研究では、センサネットワークから得られたセンシングデータをリアルタイムで Web 上にアップロードし、それを可視化することを試みる。

可視化するにあたって本システムの機能として、リアルタイムの可視化とそれ以外のある時点の可視化の 2 種類を試みる。前者はリアルタイムに更新されるセンシングデータを可視化することで人物行動の把握を目的とし、後者は過去のセンシングデータを用いて可視化をすることで過去の傾向を把握することを目的とする。リアルタイムの可視化にはリアルタイム性が求められるため、センシングデータを直接ファイルに書き出し、Web ページ側でファイルを読み込む方法が適している。また、過去のある時点の可視化にはリアルタイム性がほとんど必要ではないため、センシングデータをいったん DBMS に格納しておき、PHP を介して Web ページ側からアクセスする方法が適している。それぞれ最適な手法を用いて可視化を行い、その上で、Ajax による非同期通信を用いて、サーバに置かれたセンシングデータをリアルタイムで Web 上に可視化する。

3.1 センシングデータの収集方法

本研究で利用するセンサネットワークは、センサボード上に無線モジュール、モーションセンサ、照度センサ、温度センサを有し、乾電池で動作する。1 秒ごとに無線通信でシンクノードへデータを送信し、データベースへデータを記録する。センサボードが送信するデータは、センサに割り当てられた

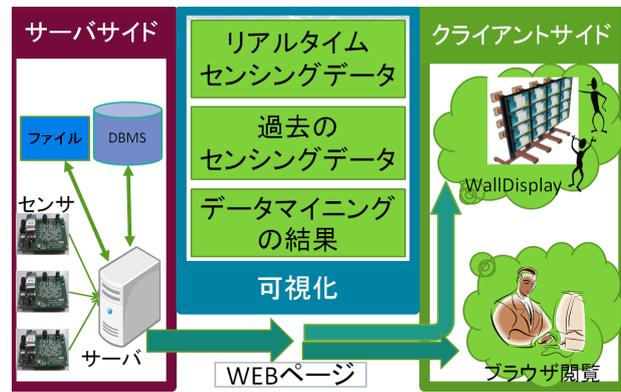


図1 システム構成図

```

- <I>
- <S0>
  <I>39209</I>
  <M>1023</M>
  <L>141</L>
  <T>26.228071</T>
  <B>5.0</B>
  <t>14:55:11</t>
</S0>
- <S1>
  <I>39209</I>
  <M>1023</M>
  <L>147</L>
  <T>26.059357</T>
  <B>5.0</B>
  <t>14:55:12</t>
</S1>
  ⋮
  
```

図2 センシングデータファイルの構造

ID、時刻、センシングしたデータを含む。無線通信の規格には IEEE802.15.4 として規格化されている ZigBee が用いられている。また、拡張ボードを用いてセンサボードに湿度センサなどを拡張することができる。他にも、消費電力特性計測ノードを用いて、交流実効電圧、有効電力、力率を計測することができる。センシングデータの収集は、静岡大学の情報学部棟 1 号館 4 階に構築されたセンサネットワークを利用する。

3.2 システム構成

現在開発しているシステムの構成は図 1 のようになっており、サーバサイドとクライアントサイドに分かれている。サーバサイドは名前の通りサーバの構成要素のことであり、例としては DBMS、YUI2 (Yahoo! User Interface Ver.2) [9] や Ext JS 3.0 [10] などである。また、クライアントサイドは本システムを閲覧するためにユーザ側で用意する必要のあるもので構成される。

3.3 サーバサイド

センサネットワークから取得したセンシングデータはセンサデータファイルと DB に格納される。センサネットワークから取得できるセンシングデータの要素としてセンサボードの ID・モーションセンサの値・照度・温度・バッテリー残量・取得時間が挙げられる。

サーバサイドで書き出されるファイルは、更新ファイルとセンサデータファイルがあり、更新ファイルは現在センシングデータを書き込み中のファイル名が書かれている。また、センサデータファイルは合計で 100 個あり、そのファイルのテキスト

属性	データ型	索引型	説明
sensorid	character varying (5)	primary key	センサボードの個体番号
sensorarea	smallint	foreign key	センサボードの場所
source	character varying (10)		消費電力特性計測ノードの接続先

表 1 sensortable のスキーマ

属性	データ型	索引型	説明
sensorarea	smallint	primary key	センサボードの場所
room	boolean		部屋かどうかの真偽値
areachar	character varying (10)		部屋の名前

表 2 areatable のスキーマ

属性	データ型	索引型	説明
id	integer	primary key	行番号
sensingtime	timestamp		センサが発火した時刻
sensorid	character varying (5)	foreign key	センサボードの個体番号
light	smallint		光センサの値
motion	smallint		モーションセンサの値
temperature	double precision		温度センサの値
moisture	double precision		湿度センサの値
battery	real		センサボードの電池残量
acvrms	double precision		接続先の交流実効電圧
effectivepower	double precision		接続先の有効電力
powerfactor	double precision		接続先の力率

表 3 sensingdatatable のスキーマ

ト形式は XML の仕様に順守しており、図 2 のような構造である。この XML ファイルのルートタグは要素 i であり、その子には要素 S0 や要素 S1 などのセンサタグがある。このセンサタグの子は一つのセンサボードから取得できるセンシングデータをひとつくりにしており、図 2 では要素 I, M, L, T, B, t がそのセンシングデータである。またそれぞれセンサボードの ID, モーションセンサの値, 照度, 温度, バッテリー値, センシングされた時間となる。

センサデータファイルはセンサタグが特定の個数になったら現在のファイルを閉じて次のファイルを書き始める。また、センサデータファイルは環状になっており、100 番目のファイルが書き終わると、最初のファイルに戻ってセンシングデータを書き始める。このセンサタグの個数は可変であるが、本実装では 60 個とした。そして、センサタグの子にはセンサボードから得られる要素があり、現在の要素はセンシングデータの要素と同じである。

3.3.1 DB

DB に格納・管理される要素として、センシングデータの要素やセンサボードが実際に置かれている場所などがある。実際に DB に作成したテーブルは表 1 から表 3 である。表 1 の sensortable ではセンサボードの個体番号、そのセンサボードの置かれている場所（部屋の番号など）、消費電力特性計測ノードの接続先（冷蔵庫やプリンタなど）が格納され、表 2 の areatable では部屋の番号、部屋かどうかの真偽値（部屋であれば 1、廊下などなら 0）、部屋の名前（J1401 など）が格納される。そして表 3 では行番号、センサの発火時刻、センサボードの個体番号、光センサの値、モーションセンサの値、温度セン

サの値、湿度センサの値、センサボードの電池残量、交流実効電圧、有効電力、力率が格納される。本システムでは DBMS に PostgreSQL8.4.1 を用いた。

3.4 クライアントサイド

クライアントサイドは、Web ブラウザを起動し、本システムの URL にアクセスすることで可視化されたセンシングデータを閲覧することができる。また、ユーザが表示機能を選択でき、対応する Web ブラウザは YUI2 に依存するので使用する OS やブラウザによって動作が異なる。

選択できる機能は、グラフ表示や等値線表示などの表示機能・複数のセンサボードのセンシングデータのピックアップ機能・センシングデータ取得時間の指定などを検討している。

また、クライアントサイドでプラグインを用意することで上記の本システムで実装予定以外のことも可視化できるようにプラグインの設計とその試作の実装を検討している。想定しているプラグインは、データマイニングの手法を実装し、その結果を可視化するプラグインである。その中で考えているデータマイニング手法は行動予測や省エネルギー化、異常検知などが挙げられる。

3.5 センシングデータの可視化

センシングデータをどのように Web 上に置くのかは上記で示した。次は、Web サーバ上に置かれたセンシングデータのように可視化するかについて考える。

まず、Web 上に置かれたセンシングデータを Web ページ側で扱えるようにするには JavaScript のクラスである XMLHttpRequest を用いる。

(1) Web ページがロードされる時、同時にサーバにある更新ファイルを読み込む。

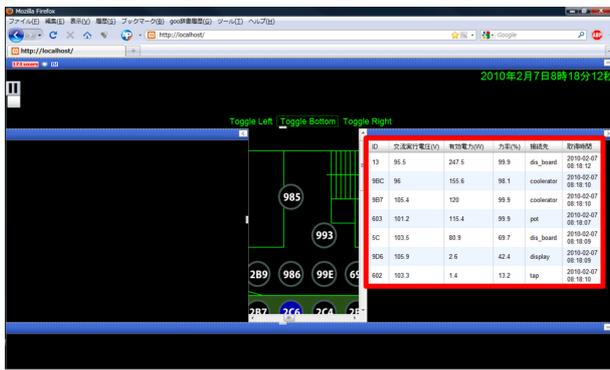


図 6 消費電力特性計測ノードテーブルのスクリーンショット

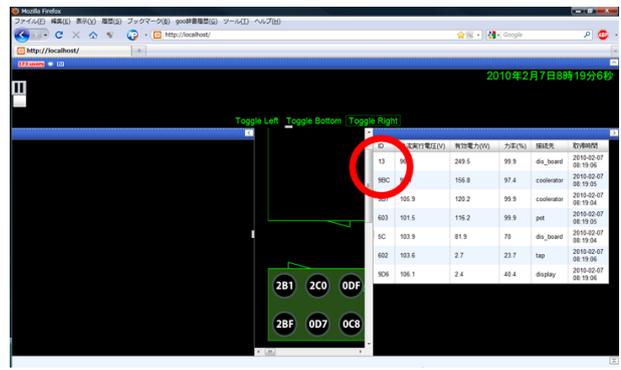


図 7 消費電力特性計測ノードのグラフ表示例

反応から 30 秒たったときに完全に黒へと戻る。

後者はあるセンサボードとその周りのセンサボードのモーションセンサの値が、1 かつ反応した時間に閾値を設けて閾値を下回る場合には、対象となったセンサボード間に図 5 下図の赤で囲んでいる部分のように矢印を描くことで、濃淡を变化させるよりも人の移動が把握できると考える。この矢印を描くために用いたライブラリは JavaScript Vector Graphics Library [8] である。

この 2 つの機能は組み合わせることによって人が移動しているのか、その場所に止まっているかを判別した。例えば矢印が描かれたときは人が移動していることがわかり、矢印は描かれずにある特定の円の色が濃い状態を保っているのなら人がその場所に止まっていることがわかる。

4.2.2 消費電力特性計測ノードテーブル

このテーブルは本システムのレイアウトの Right 部分に表示される機能であり、図 6 の赤で囲んでいる部分のように表示される。テーブルの属性として、センサボードの ID、交流実効電圧 (V)、有効電力 (W)、力率 (%)、センサボードの接続先、センシングデータの取得時間がある。また、このテーブルは 3 秒に一度 Ajax 通信を行うことで最新の状態に更新される。

4.2.3 グラフ化

このグラフ化では消費電力特性計測ノード用の要素を用いたグラフとそれ以外のセンサボードの要素を用いたグラフの 2 種類がある。

前者の要素としては、交流実効電圧、有効電力、力率の各平均値があり、x 軸は時間となっている。また、このとき表示される要素は交流実効電圧は水色、有効電力は黄色、力率は紫色で全て折れ線グラフで表示される。このグラフを表示するには図 7 上図にあるように、消費電力特性計測ノードテーブルにある ID のセルをクリックすることで図 7 下図の赤で囲ってある部分のようにレイアウトの Left にグラフが表示される。

後者の要素としては、モーションセンサの値の積算値・照度の平均値・温度の平均値があり、x 軸は時間となっている。ここでは、モーションセンサの値を時間ごとの積算値として算出し、活動量としてみることで、モーションセンサ付近で人が活発に活動しているかを判断できる。そして、この値を基準とすることで、活動量によって消費電力の無駄を発見できると考えた。例えば、冬の夜に活動量が少ないのに照度や温度が高いグ

ラフであったときは照明や暖房をつけすぎていて、その部分は無駄な消費電力であることがわかる。また、このとき表示される要素のうち、モーションセンサの値の積算値は黄緑色で棒グラフ、照度は黄色、温度は赤色でそれぞれ折れ線グラフで表示される。このグラフを表示するには図 8 上図にあるように、センサボードを表す円を選択すると選択したセンサボードのセンシングデータを用いて図 8 下図の赤で囲ってある部分のようにレイアウトの Left にグラフが表示される。このときグラフを表示するために選択したセンサボードは図 8 下図のように赤に変化する。

他にも、レイアウトの Center 部分で表示されているセンサボードのセンシングデータをグラフ化するときには、個別のセンサボードだけではなく図 9 のように複数のセンサボードのセンシングデータを組み合わせることができる。このように複数のセンサボードのセンシングデータを一つのグラフにまとめることができるため、部屋ごとや建物ごとといった全体をグラフ表示することが可能である。

また、消費電力特性ノードとそれ以外のセンサボードのセンシングデータを用いて作成されるグラフは 2 種類あり、一方は 1 時間前から現在までの間に取得されたセンシングデータのグラフで、もう一方は現在から 24 時間の間に取得されたセンシングデータのグラフである。1 時間前から現在までのグラフにおいては、1 秒ごとにセンシングデータの差分を取得するため、リアルタイムの変化を見ることが可能である。そのため、現在の状態が閲覧したいときには前者のグラフを用いて、一日の傾向が知りたいときには後者のグラフを用いることで使い分ける。

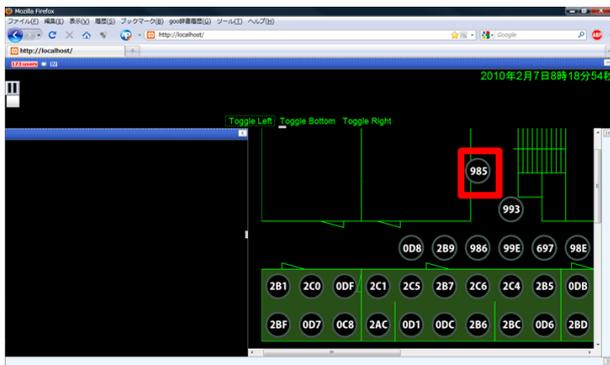


図 8 センサボードのグラフ表示例

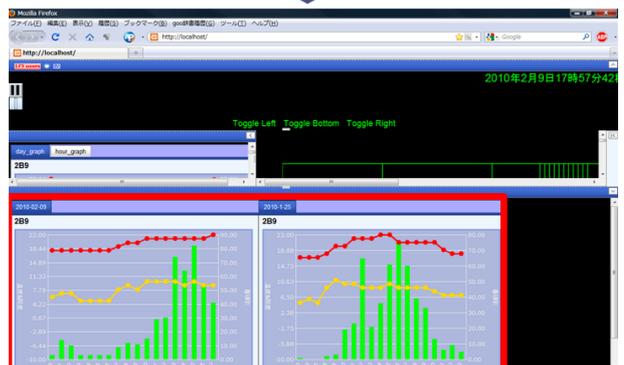
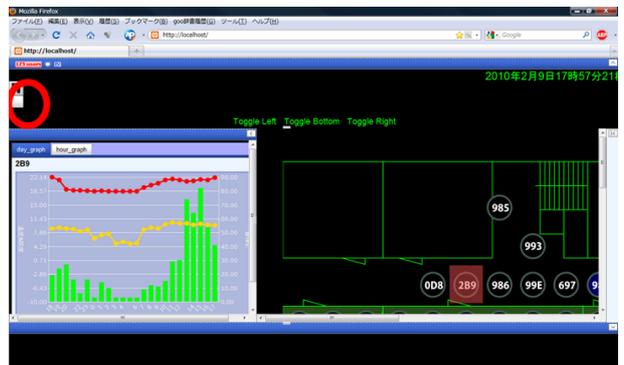


図 10 活動量類似グラフの表示例



図 9 複数のセンサボードのセンシングデータを組み合わせたグラフ表示例

また、この 2 種類のグラフはタブを用いて切り替えることが可能である。このグラフとタブを作成するために用いたライブラリは YUI2 である。

4.2.4 活動量類似グラフ

この機能は現在グラフを表示しているセンサボードの一時間の活動量の変化量と一番類似している日のセンシングデータをグラフ化する機能である。この機能により、現在と一番似ている日の温度や照度を比較をすることができる。

この機能は活動量を要素として持つグラフを表示しているときに図 10 上図の赤い円で囲んであるボタンを押すことで使用することができる。このボタンを押すことによって、現在グラフを表示しているセンサボードの ID と現在の時刻を引数に Ajax 通信を行う。PHP 側で現在の活動量の変化量とそれ以外の日の活動量の変化量を求めて、ユークリッド距離により類似度計算

を行う。そして、現在と一番似ている日のデータを JSON 形式で返し、それぞれを図 10 下図で赤く囲んでいる部分のようにグラフ化される。

5. 可視化の応用例

本システムを用いた応用例としては、省エネルギー化やモニタリングなどが挙げられる。例えば、省エネルギー化については、Pacific Northwest National Laboratory の実証実験 [7] で、電力消費量をモニタリングするツールを見せるだけでも人は省エネルギー化を心がけ、平均 10% の省エネルギー化に成功している。そのため、リアルタイムのセンシングデータをグラフで表示すると同時に DB から現在と似た活動量を持っていて最も消費電力が少なかった日を検索し、そのデータも一緒にグラフ表示することによって、より一層消費電力を抑えることができると考えている。

また、本システムに実装予定であるデータマイニング手法を選択できるようなプラグインを用いることによって、最適と思われる省エネ方法などをピックアップさせるような仕組みを検討している。

他にも、モニタリングにおいては CO2 センサや雨量センサなど、用いるセンサの種類を増やすことで環境モニタリングシステムとして使用することが可能であり、他にも人間の行動をモニタリングするシステムやセンサノードの異常検知システムなどとして使用することができると考えられる。

6. おわりに

本研究では、センサネットワークから得られるセンシングデータの可視化の提案と実装を行った。また、可視化するために必要なセンシングデータを Web 上に流す方法についても DB を用いる方法と用いない方法の 2 種類を論じた。

今後の課題としては、リアルタイムでデータマイニングが可能になるようなプラグインの規格化やスケーラビリティの向上を検討している。

本システムではモーションセンサの値によって、表示されているセンサーノードの色の濃淡を変更したり、矢印などを表示することによって人の流れが大まかに把握できるようになった。これにより、センシングデータをより人に見やすい形で可視化できたと考えられる。しかし、より人の行動が簡単に把握できるようになったためプライバシーが露呈する可能性がある。そのためプライバシーの問題も新たに考える必要があると考えられる。

他にも、本システムではグラフを表示することで、そのグラフをみることによってユーザに電力の無駄などを発見することを補助し、その無駄を発見したユーザが無駄を意識していくことで消費電力を抑えることが可能になったと考えられる。しかし、より消費電力を抑えるためにはユーザにより一層働きかえるようなグラフ表示の方法を考えていく必要があると考える。

また、センサネットワークから消費電力を取得することができるようになったため、その消費電力を表示することができるようになった。それだけでも省電力化につながるが、ただ表示するだけでなく電気料金を表示したり、CO2 排出量に換算して表示することで、より一層の省電力化を図りたい。

謝辞 本研究の一部は文部科学省の知的クラスター創成事業(第Ⅱ期)「浜松地域オプトロニクスクラスター構想」、科学研究費補助金基盤研究(B)(課題番号 19300026)、科学研究費補助金基盤研究(B)(課題番号 20300027)の助成による。

文 献

- [1] Google Maps, <http://maps.google.com/>
- [2] J.J.Garrett, "Ajax : A New Approach to Web Applications", <http://www.adaptivepath.com/ideas/essays/archives/000385.php>
- [3] JSON, <http://json.org/>
- [4] Toshihiro Takada, Satoshi Kurihara, Toshio Hirotsu, and Toshiharu Sugawara, "Proximity mining: Finding proximity using sensor data history", In IEEE Workshop on WMCSA, 2003.
- [5] 本田誠一, 福井健一, 森山甲一, 栗原聡, 沼尾正行, "赤外線センサネットワークによる人物追跡", JSAI2006, 2006
- [6] 横山昌平, "Web ブラウザから利用できる SAX パーサ“ Freddy ”の実装と評価", 電子情報通信学会論文誌 D Vol.J91-D No.3 pp.585-594, 2008.
- [7] JavaScript Vector Graphics Library http://www.walterzorn.com/jsgraphics/jsgraphics_e.html
- [8] Yahoo! User Interface Ver.3, <http://developer.yahoo.com/yui/>
- [9] EXT JS 3.0, <http://developer.yahoo.com/yui/>
- [10] Pacific Northwest National Laboratory, "Department of Energy putting power in the hands of consumers through technology", <http://www.pnl.gov/topstory.asp?id=285>