

IT強震計用に開発した Web利用の動的表示技術

- 伊藤貴盛(応用地震計測)・鷹野澄(東京大学地震研究所)・
林能成(名古屋大学災害対策室)

はじめに

最近のネットワークの広帯域化の進展は著しく、かつては不可能であった観測システムの構築が容易に行えるようになってきている。また、それ以上に、観測データの公開に関する自由度が増している。そこで、IT強震計では現在のインターネット利用環境に適応した表示システムについて試作しており、Web上でJavaアプレットを使用するものを積極的に開発している。

Java

サンマイクロシステム社が開発したオブジェクト指向言語と仮想マシンを含めたその環境。

ソースコードをコンパイルした段階で、完全な実行用バイナリではなく中間コード(Javaバイトコード)となり、実行マシン上で動作するJVMという仮想のコンピュータ上でコード実行される。このため、JVMがインストールされた環境であれば、プログラムはどんなマシンでもそのまま使用することができる。

Javaアプリケーション

Java で作成され、バイトコードに変換された実行プログラム。JavaバイトコードをJVMに読み込ませる形になるが、できることは他のアプリケーションプログラムと同様。

Javaアプレット

Java で作成され、バイトコードに変換された実行プログラム。ブラウザにダウンロードされて実行される点がアプリケーションと異なる。Javaアプレットを使うことにより、HTMLで記述された静的なWebページでは実現できない動的な表現が可能となる。IT強震計では、まずこの点に着目した。セキュリティのために、自身が呼び出されたWebサーバ以外とは通信等できない。

従来の観測データ表示(専用アプリケーション)

長所

- ・ 多機能
- ・ 動作が速い
- ・ 観測装置にかかる負荷は低い(自己が観測装置を兼ねる場合は除く)
- ・ 表示の微調整が柔軟に行える

短所

- ・ 実行ソフトのインストールが必要
- ・ 汎用性が低い(低くなる)
- ・ 実行環境が制限され、Windowsでしか動かない場合が多い
- ・ インターネット経由の使用は通常考慮されない

従来の観測データ表示(Web)

長所

- ・ Webブラウザは最近の多くのOS(あるいはディストリビューション)に標準装備される
- ・ もともとインターネットでの情報共有用のため、データ公開が容易
- ・ 一般ユーザが表示を変更するのは容易

短所

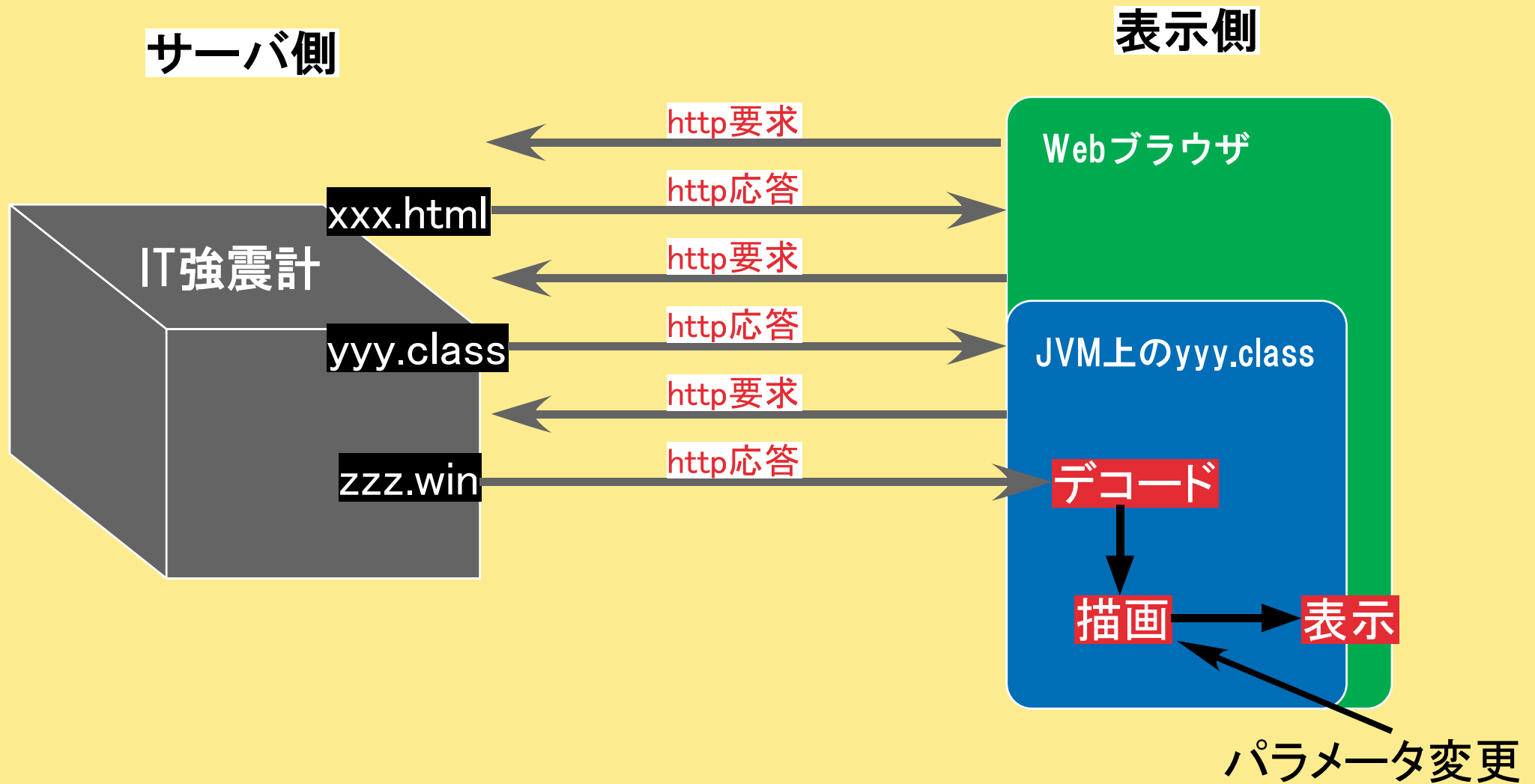
- ・ 表示される画面は、通常動かない
- ・ 表示側で、表示の微調整を行うことは難しい
- ・ CGI等の併用で機能は増やせるが、サーバの負荷が高くなる
- ・ 高度な表示を行うと、サーバだけでなくネットワークの負荷も高くなる

Javaアプレット表示

Webブラウザが、htmlファイルに従ってJavaアプレットをWebサーバから取得し、実行する。Javaアプレットは、指定された方法で指定されたデータをサーバから取得して、表示を行う。Javaアプレット自体はプログラムなので表示アプリケーション高度な描画も可能であるが、インストールは必要ない。

1. アニメーション表示も容易に作れる
2. ユーザによる描画の微調整が容易
3. サーバの負荷は小さい
4. ネットワークトラフィックも必要最低限
5. htmlから各種パラメータ指定可能
6. Webサーバ無しのローカル動作も可能(イベント・ファイル表示系)

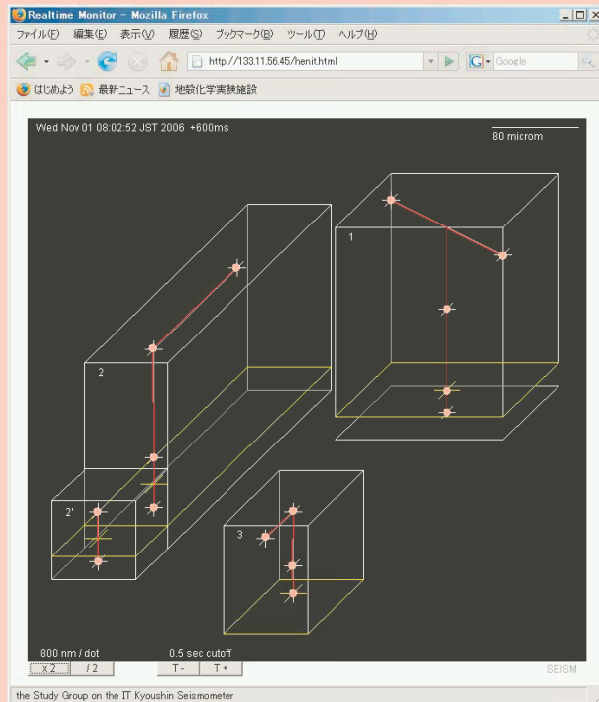
表示までの流れ



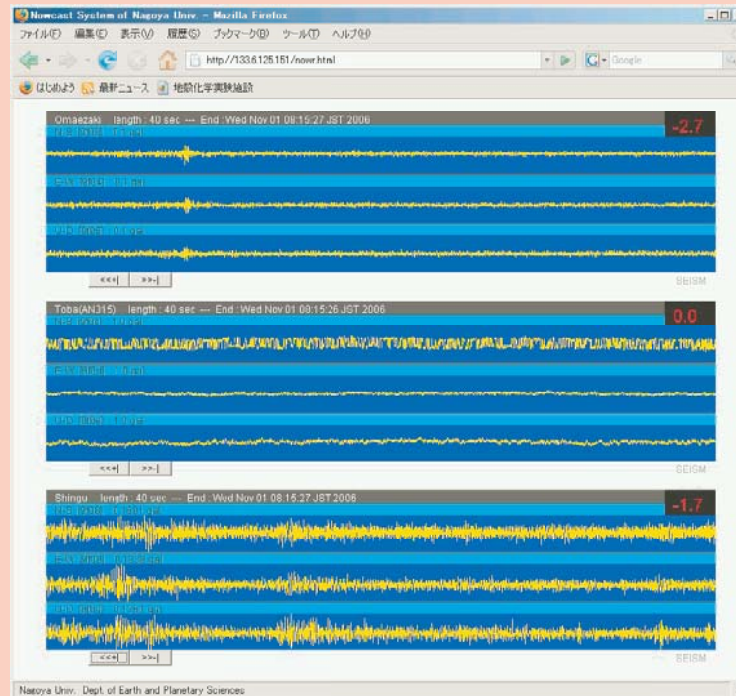
製作したJavaアプレット

a. リアルタイム表示系

2004年秋季大会にて発表済(その後も改良は続行)。基本的に、1秒単位のデータをUDPパケット交換によってサーバから得て表示を行う。データを小さくするため、(視覚化したときに不都合が無い範囲で)不可逆の対数圧縮を施す場合がある。



リアルタイム構造物変形



リアルタイム波形

スペアナ(開発中)

ナウキャスト(開発中)

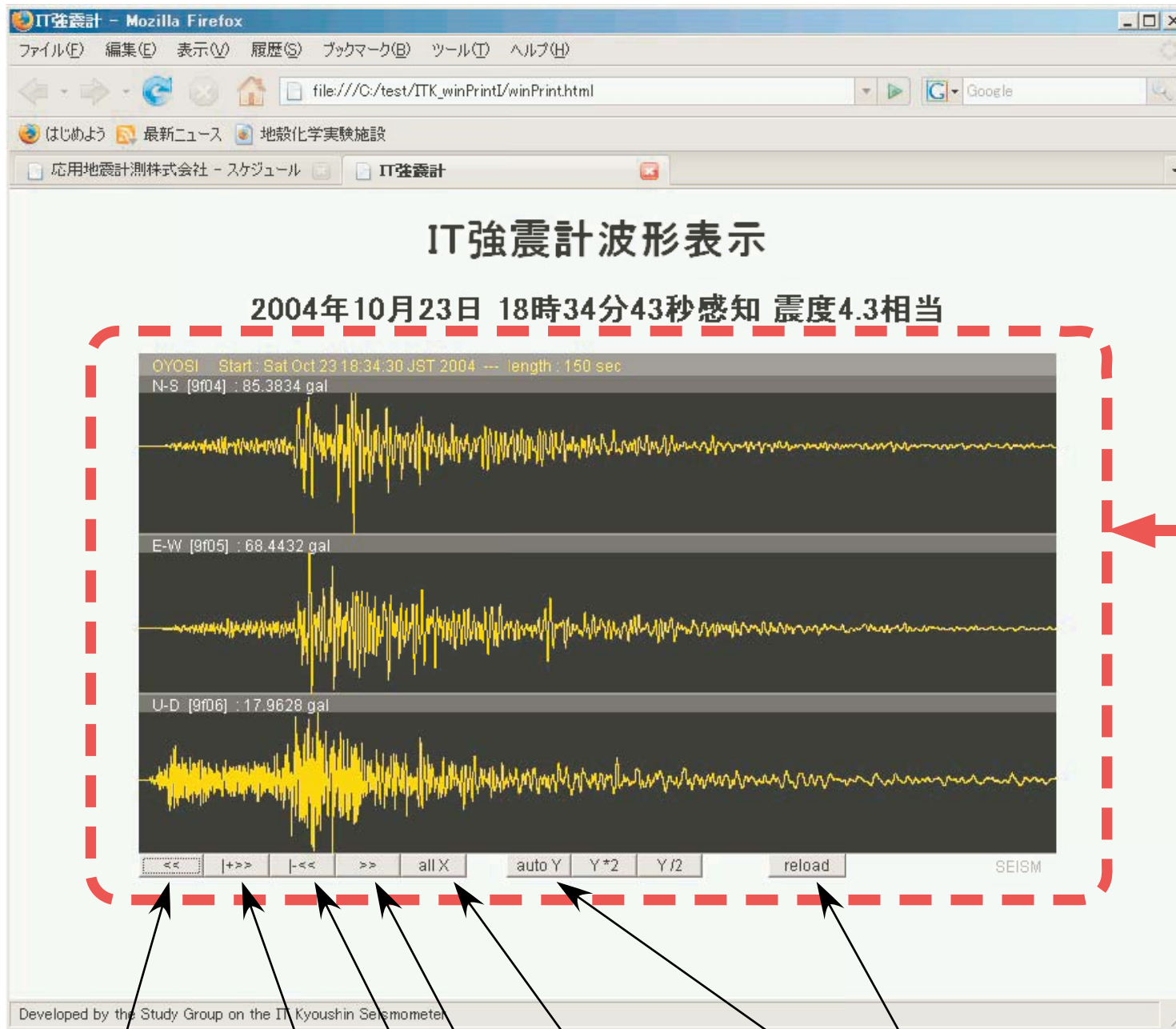


b. イベント(ファイル)表示系

Javaアプレットが、サーバ上の指定されたデータファイルをhttpで読み込んで表示を行う。現在は、winファイルを読み込んでいる。リアルタイム性は弱いですが、http(80/tcp)でのアクセスさえ許可されていれば使用可能。

波形表示

指定したwinファイルを読み込んで波形表示を行う。読み込んだ後で、表示時間範囲の調整やスケール調整が可能。htmlファイルから呼び出すときのパラメータ指定で、表示カラーの調整が可能。CGIと組み合わせることで、連続波形データ(1分ファイル集合)から任意時間の切り出しなども可能であり、工夫次第で用途は多彩。



アプレット外のhtml部分とは
干渉しない

パラメータでの
表示調整が可能

時間方向の表示長さ拡大
時間方向の表示長さ縮小
表示範囲を後ろへ移動
表示範囲を前へ移動
ファイル全体を表示
オートスケール
再読み込み

アプレット(ローカル)の操作で
表示の変更が可能

```

<?xml version="1.0" encoding="EUC-jp"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ja" lang="ja">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=EUC-jp" />
<meta http-equiv="Content-Style-Type" content="text/css" />
<title>IT強震計</title>
<style type="text/css">
.center{ text-align: center }
</style>
</head>
<body>
<h1 class="center">IT強震計波形表示</h1>
<div class="center">
<h2>2004年10月23日 18時34分43秒感知 震度4.3相当</h2>
<applet code="winPrintI.class" width="700" height="400"> <!--実行アプレットと表示サイズ指定-->
  <param name="file" value="0410231834.win" /> <!--表示ファイル指定-->
  <param name="station" value="OYOSI" /> <!--観測点名-->
  <param name="channel" value="9f04" /> <!--表示開始チャンネル. 連続3チャンネル固定-->
  <param name="refresh" value="4096" /> <!--再描画までの時間(秒)-->
  <param name="unit" value="gal" /> <!--単位-->
  <param name="const" value="4280" /> <!--1単位のdigit値-->
  <param name="block" value="1" /> <!--秒ブロックの長さ(秒)-->
  <param name="range" value="0" /> <!--振幅表示指定...0:表示領域の幅. 1:最大値と最小値-->
  <param name="skip" value="30" /> <!--ファイル先頭の非表示部分の長さ-->
  <param name="length" value="150" /> <!--表示長さ(秒)-->
  <param name="channel1" value="N-S" /> <!--1チャンネル目のチャンネル名-->
  <param name="channel2" value="E-W" /> <!--2チャンネル目のチャンネル名-->
  <param name="channel3" value="U-D" /> <!--3チャンネル目のチャンネル名-->
  <param name="linecolor" value="ffef00" /> <!--グラフの線色-->
  <param name="backcolor" value="606060" /> <!--観測点名領域の背景色-->
  <param name="base0color" value="404040" /> <!--チャンネル名領域の背景色-->
  <param name="basecolor" value="000000" /> <!--グラフ領域の背景色-->
  <param name="font1color" value="ffff80" /> <!--観測点名領域の文字色-->
  <param name="font2color" value="ffffff" /> <!--チャンネル名領域の文字色-->
  <!--<param name="comment" value="ITK System" />--> <!--ステータスバーに表示する文字列-->
</applet>
</div>
</body>
</html>

```

htmlファイル

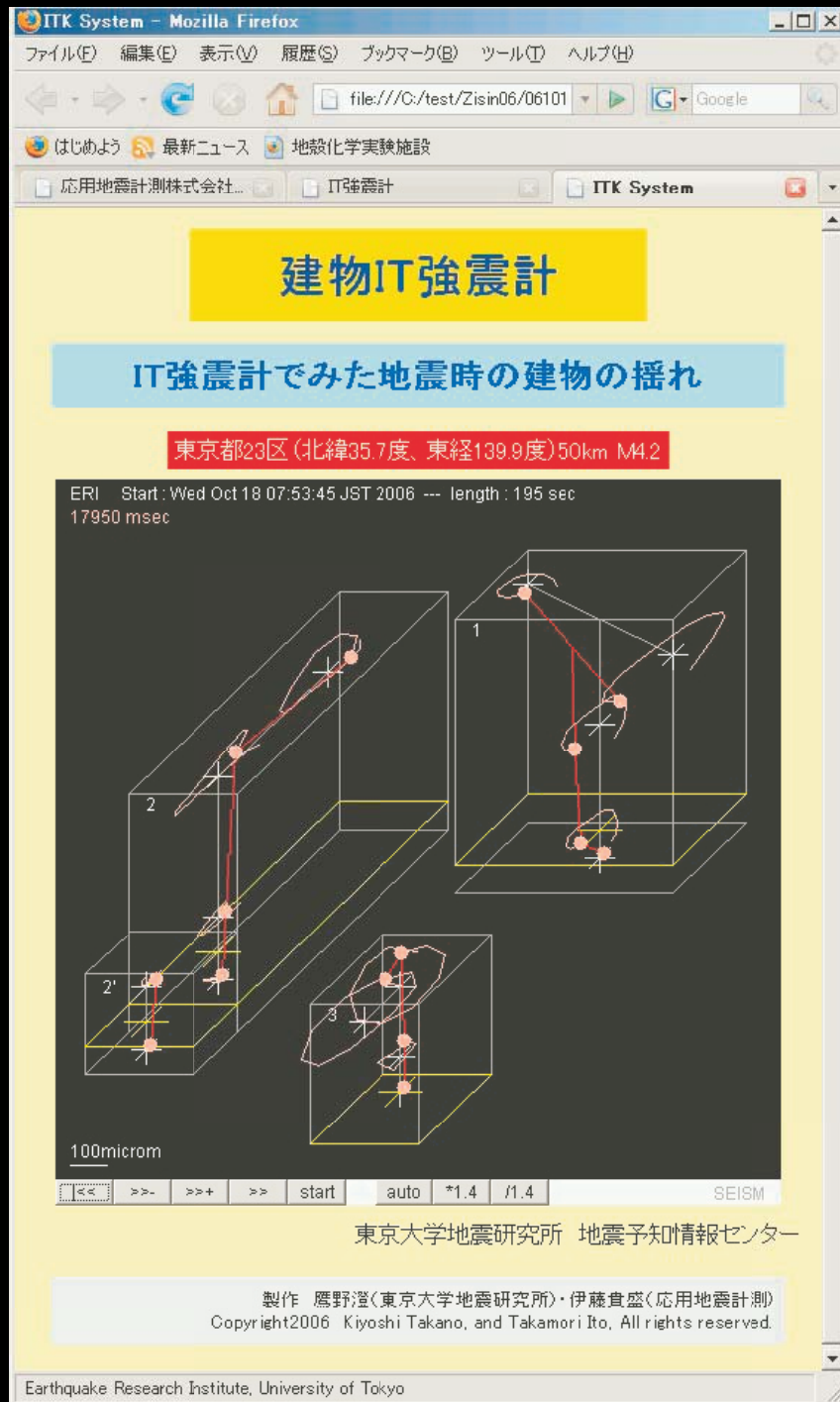
この部分が
アプレット表示に

構造物変形表示

あらかじめサーバ内でイベント波形を積分して変位波形データを作っておき、それを読み込むことで表示を行う。基本はパーティクルモーション表示であり、それを複数組み合わせることで構造物の変形を表現する。波形を見ただけではよく分からない建物の揺れの特徴を視覚的にとらえることができるようになる。

積分処理

表示アプレットは、winフォーマットで格納された変位波形ならば、積分方法等に関係なく表示できる。現在、積分処理は、時間領域での処理を行っている。**オフセット除去→ローカットフィルタ→積分して速度を求める→積分して変位を求める→ローカットフィルタ**という一連の処理を行う。ローカットフィルタでオフセット除去が可能なのにわざわざオフセット除去処理を行っているのは、IT強震計のイベント波形はディジタイザ出力そのままの連続波形を指定部分切り出したもので大きなオフセットがのっている場合があるためである。



表示アプレットは、ファイルから変位データ入手して、それに基づいて順に描画を繰り返すことでアニメーション表示を実現している。描画間隔を調整することにより、再生速度の変更も可能である。現在は、変位波形を20Hzのwinファイルとしておき、50ms間隔の描画を行うことで等速再生を行うものをデフォルトとしている。

パーティクルモーション表示を複数組あわせて、点どうしを線で結ぶ形で作成したため、各点の動きの軌跡が表示される。振動履歴が画面上に残ることになるため、これも揺れの特徴を視覚的にとらえるのに一役買っている。

再生速度調整、拡大率変更、再描画等が、サーバに依存せずに可能なのは、アプレット利用による特長である。

ファイル表示系のアプレットの場合、サーバは必要なく、デモ機のようにローカルのPC単体で表示が可能となることも特徴。

今後の開発

この技術を使用した表示を行う場合，UDPパケット交換によるリアルタイム表示，ファイル表示ともにサーバとなるIT強震計側の負荷が小さいことが実際に確認されている．IT強震計は，表示の制御だけでなく自らが(連続)波形収集・演算する点が特徴であり，そちらの処理の確実性を保証する上でこれは重要なことである．また，個々のIT強震計と地域センターサーバ等に同じ方法でアクセスできることから，双方を統合した表示システムを作ることが容易である．今後，IT強震計の処理高度化に合わせて各種の表示アプリットを作成していく予定である．

参考

鷹野・伊藤・原2004, IT強震計, 地震学会秋季大会, 2004.10

伊藤・林・鷹野2004, Javaアプレットを用いたリアルタイムモニタリング表示, 地震学会秋季大会, 2004.10

鷹野・伊藤2005, 建物用IT強震計システム, 地震学会秋季大会, 2005.10

鷹野・伊藤2006, IT強震計でみた震度1の建物の揺れ, 本大会

IT強震計HP <http://www.eic.eri.u-tokyo.ac.jp/ITKyoshin/>

同研究会HP <http://www.eic.eri.u-tokyo.ac.jp/ITKyoshin/Kenkyukai/>

本大会の企業展示, 応用地震計測(株)ブースにて, リアルタイム表示アプレットが動作しています. IT強震計用デジタルセンサITK-Sensorと一緒にIT強震計の表示機能をお確かめください.

デモ機では、応用地震計測ブースの展示システムと無線LANで接続して、リアルタイム表示も行っています。ITK-Sensor(IT強震計用に開発した新型のデジタルセンサ)の波形、構造物変形表示のデモ、開発中のスペアナ風表示が可能です。(ブースでも表示)

ローカルで、東京大学地震研究所の建物IT強震計で観測されたデータに基づく、構造物変形表示の連続切替表示を行っています。