インターネットワークの経路と接続状況モニタシステムの開発

静岡大学大学院 理工学研究科 システム工学専攻
若林研究室 50930219 反保 良平

1　はじめに

現在，多くの社会システムがコンピュータネットワークに依存しており，コンピュータネットワークの不安はシステムへの不安にも繋がるといえる．そこで，システムの安定性を高めるために，システム管理者がネットワークを監視する必要があるが，大規模なネットワークにおいて，その監視作業を少数で構成される管理者のみで行うことは困難である．

本研究では大規模ネットワークにおいて，管理者以外のユーザが常にネットワークの経路と接続状況をモニタでき，かつ管理者のネットワーク管理作業を助けられるようなシステムを開発する．

2　システムの概要

インターネットワークの状況を監視するには，各ホスト間の経路，各ホストまでの時間の 2 つの情報が必要である．これらは，UNIX 上の traceroute コマンドを用いて得ることができる．しかし，一般ユーザの環境下ではこのコマンドを実行できないことが多く，かつ得られた情報は，実行時刻やトラフィックによる変化があり，このコマンドだけではネットワーク状況が掴みづらい．

そこで本システムでは，これらの情報を一般ユーザと同一ネットワーク上サーバが定期的に取得，統計処理を行い，データベース化する．次に CGI(Common Gateway Interface) を用いてデータベースを参照可能にし，一般ユーザもブラウザ上でネットワーク状況を把握できるようにする．

ここで本システムを, 機能別に 3 部に分けて構成する．まず，ネットワークの情報を定期的に取得する部分 (以下，データ取得部)．次に，得られた情報をデータベース化する部分 (以下，データ加工部)．最後に，データベースからの情報をブラウザ上に表示する部分 (以下，ユーザ対話部)．

また，本システムでは監視対象ネットワークを，最も一般的かつ大規模な WWW(World Wide Web) インターネットワークに想定し，国立大学の WWW サーバを対象ホストとした．

3　データ取得部

```
 [r-tanbo@raicho]:/home/nex1/r-tanbo {16} % traceroute fs1.nc.kyushu-u.ac.jp
traceroute to fs1.nc.kyushu-u.ac.jp (133.5.1.2), 30 hops max, 40 byte packets
 1  133.70.216.250 (133.70.216.250)  0.484 ms  0.393 ms  0.356 ms
 2  133.70.128.250 (133.70.128.250)  1.107 ms  1.353 ms  1.406 ms
 3  gw.ipc.shizuoka.ac.jp (133.70.129.1)  264.543 ms  199.949 ms  239.672 ms
 4  nagoya-10-A7-0-6.sinet.ad.jp (150.99.198.41)  273.778 ms  295.134 ms  273.773 ms
 5  osaka-10-A4-0-2.sinet.ad.jp (150.99.113.4)  82.579 ms  169.410 ms  179.533 ms
 6  kyushu-10-A4-0-1.sinet.ad.jp (150.99.128.6)  223.078 ms  193.401 ms  229.616 ms
 7  kyushu-1-A3-0-1.sinet.ad.jp (150.99.198.101)  218.325 ms  294.613 ms  259.869 ms
 8  cisco0.nc.kyushu-u.ac.jp (133.5.19.1)  279.579 ms  308.163 ms  223.683 ms
 9  fs1.nc.kyushu-u.ac.jp (133.5.1.2)  194.943 ms  183.773 ms  226.524 ms
[r-tanbo@raicho]:/home/nex1/r-tanbo {17} %
```

図 1: traceroute コマンドの出力例

traceroute コマンドは図 1 のような出力になっており，目的ホストまでの経路データと RTT(Round Trip Time) と呼ばれる時間データが得られる．この情報に実行時刻のラベルを付け，1 組のデータとした．UNIX サーバ上で，このデータを得るスクリプト (getData.sh) を cron で定期的に実行する．

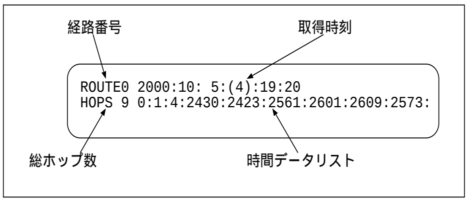
4　データ加工部

データ取得部で得られたデータを開発した加工プログラム (filter.c) で処理し，以下で説明する 2 つのファイルに分け，データベースを構築する．

4.1　経路情報ファイル

プロトタイプモデルによるデータ分析の結果，対象ホスト (WWW サーバ) までの経路は，めったに変化しないことが判った．そこですべてのデータが同じ経路情報を表す文字列を持つ必要が無いと考え，ユニークな経路が得られたときのみ，このファイルは更新するようにした．よってデータは経路情報ファイルに記述された経路番号のみを持てばよい．

4.2 データベースファイル



```
ROUTE0 2000:10: 5:(4):19:20
HOPS 9 0:1:4:2430:2423:2561:2601:2609:2573:
```

図 2: データベースファイルのデータ例

- 経路番号: 経路情報ファイル中のユニークな経路を表す整数．
- 取得時刻: date コマンドの出力を表す文字列．
- 総ホップ数: データの整合性をチェックする指標，目的ホストが何ホップ離れているかを示す整数．
- 時間データリスト 各ホップ毎の 3 つの RTT 値の平均値．リストとして持つ．

以上の 4 つのデータ (経路番号，取得時刻，総ホップ数，時間データリスト) が，1 組のデータとしてデータベースファイルに格納される．この例は，経路情報ファイルで経路 0 番として登録されている経路を通り，2000 年 10 月 5 日 (木曜) の 19 時 20 分に取得され，総ホップ数が 9，時間データリストに表される時間データ (RTT) で目的ホストまで辿りついたことを示している．

5 ユーザ対話部

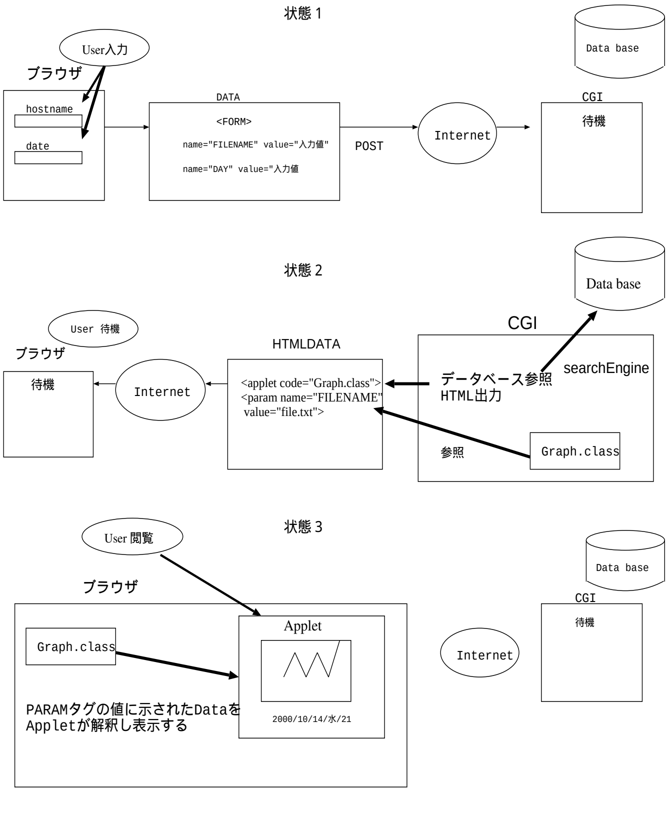


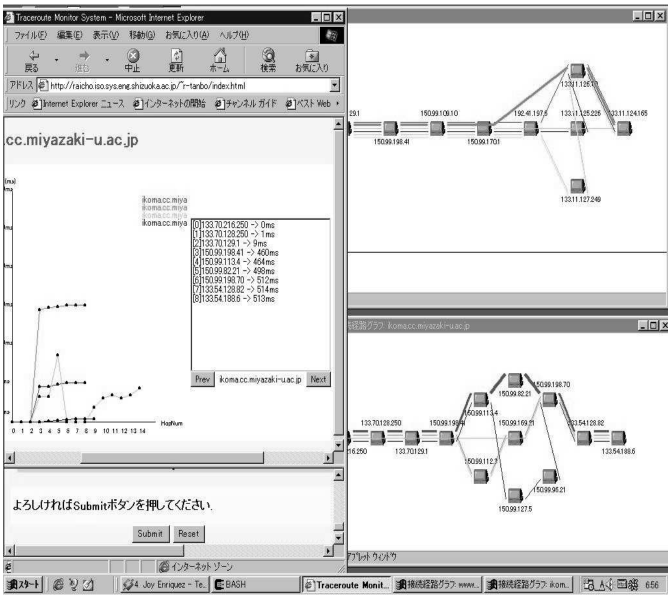
図 3: サーバ側とブラウザ側の状態遷移図

ユーザ対話部は，CGI を用いてブラウザ側から検索キーを受け取り，サーバ側の検索プログラム (searchEngine.c) を実行し，出力結果を HTML ファイル中の $< PARAM >$ タグに埋め込む．$< PARAM >$ タグに記述された値は，ブラウザ上の JavaApplet プログラムの入力となる．ここで，サーバ側とクライアント側の状態遷移を図 3 に示す．

5.1 ユーザインターフェイス

ネットワーク状況を判り易く表現するために，JavaApplet プログラムを用いてユーザインターフェイスを構築した．その表現方法は以下の 3 つである．

- テキスト表現 :テキストベースでの表示で細かな時間データの差を表す．
- グラフ表現 ホップ数毎に時間が変化する様子を表す．
- 階層構造表現 :同じホストを通る経路を判断し，目的ホストまでの経路を階層構造状に表す．



```
[0]133.70.216.250 -> 0ms
[1]133.70.128.250 -> 1ms
[2]133.70.129.1 -> 9ms
[3]150.99.198.41 -> 460ms
[4]150.99.113.4 -> 464ms
[5]150.99.82.21 -> 498ms
[6]150.99.198.70 -> 512ms
[7]133.54.128.82 -> 514ms
[8]133.54.188.6 -> 513ms
```

よろしければSubmitボタンを押してください.

図 4: Applet が起動した画面

6 おわりに

本システムを用いることで，traceroute コマンドだけでは得られない情報，つまり分岐路や時刻による速度の変化，対象ホスト間の関係等の情報をユーザ側に提供できるようになった．これでユーザは対象ホストまでのネットワーク状況を把握したいとき，より詳しい情報が得られることになったと考えられる．このシステムを導入することで，ネットワークの安定性，しいてはシステムの安定性が向上することが望まれる．