

# Web ブラウザで動作する ネットワークエミュレータ用 GUI の実装

2011SE046 平松 剛

指導教員 後藤 邦夫

## 1 はじめに

Web サービスやネットワークを利用するアプリケーションの開発には様々な障害を想定して検証する必要がある。そこで用いられるのがソフトウェアネットワークエミュレータである。ソフトウェアネットワークエミュレータはホスト内に仮想ネットワークを構築し、実際にパケットの送受信をすることができる。Common Open Research Emulator(以降, CORE[2]) と後藤研究室で開発中の Goto's IP Network Emulator(以降, GINE[1]) はソフトウェアネットワークエミュレータである。しかし上記のソフトウェアは, Linux カーネルの Network Name Space 機能や同様の機能が存在する FreeBSD(CORE のみ対応) といった特定の環境でしか動作させることができない。そこで, この研究では Web ブラウザ上に GUI を実装し, インターネットを介してデーモンが動作する Web サーバに接続することでデータをやりとりし, 上記のネットワークエミュレータを動作環境外である Windows や MacOS から GUI 操作可能にするのが目的である。

## 2 ネットワークエミュレータの概要

本節では, 研究のベースにするネットワークエミュレータについて述べる。

### 2.1 CORE の概要

CORE の GUI は簡潔でとても操作しやすくなっている。オブジェクトをキャンバス上へ配置し, ドラッグ&ドロップで移動することができる。また, ダブルクリックすることでそのオブジェクトの IP アドレス等, ネットワークの設定ができる。

### 2.2 GINE の概要

GINE を使用することで, 多数のルータやホストで構築された大規模なネットワークをエミュレートすることができる。上り下りの通信経路に遅延やパケットロスの設定も可能でさらにバンド幅も自由に設定できる。これらの機能によってネットワーク障害を現実に近い環境で再現できる。また, GINE は C++ で記述されたプログラムであり, 研究によって改良が続けられている。

現在, GINE を研究や学習で使用する場合, エミュレーションするネットワークごとに C++ でプログラムを作成し, それを CUI で実行している。しかし, この方法ではプログラムを熟知していないユーザは使用することが困難である。そこで 2010 年南山大学浅野修士論文 [3] によって, オープンソースで配布されている Qt を使用して構築されたのが GINE-GUI である。2011 年南山大学太田, 今川卒業論文 [4] によって仮想ネットワークの部分的削除機能の追加がなされている。

## 3 システムの概要

本研究で構築していくシステムの構想を記述する。検討の結果, GUI 部分では直感的操作が可能で非常に使いやすい CORE の GUI システムを目指し, CORE より多機能で詳細な設定が可能である GINE をエミュレーションに用いることとした。システム全体の構成は図 1 の通りである。図 1 の番号 1, 2 番はクライアントサイドを構成

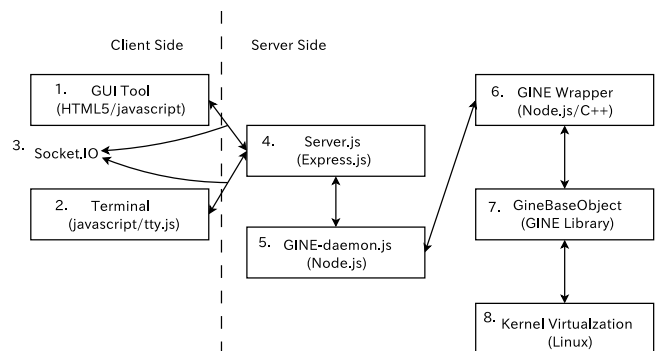


図 1 全体の構成図

するアプリケーションであり, Web ブラウザ上で動作する。複数の OS で動作させるため環境に依存させないよう, GUI-Tool, Terminal 共に HTML と JavaScript を主体として記述している。番号 4 から 7 番はサーバサイドを構成するアプリケーションであり, 4 番の Server.js によって簡易な HTTP サーバを作成し, クライアントからアクセスがあると HTML ファイルを送信する。Node.js は Google の V8 JavaScript エンジンで動作し, V8 が C++ で記述されているために, C++ で書かれたアプリケーションの一部として動作させることができる。GINE ライブラリは C++ によって記述されているので, Node.js は Wrapper ファイルを記述することで GINE ライブラリと繋げることができるので, サーバサイドアプリケーションとして適していると考えたため, Node.js を使用することにした。

## 4 システムの実装

本研究で実装した部分について図 1 中の番号に沿って説明する。

- 番号 1 の GUI Tool については, 操作しやすい環境を目指すために, JavaScript 独自のデータ形式である JSON ファイルから描画データを動的に更新することができる D3.js というライブラリを用いて GUI Tool を実装した。また, Twitter など用いられている Bootstrap.js で仮想ノードの設定画面を設計した。

- 番号2の端末処理では, tty.js という Node.js モジュールの一部を書き換えることで GINE によって生成される仮想ノード端末の入出力と, GUI 側の端末の入出力をつないでいる.
- 番号3の通信部分は Socket.IO.js を使用しており, 数行のコードで簡単にソケット通信が可能になる.

```

  データ送受信のプログラム
  socket.on('message', function(data)//受信部分
  {
    var data = data.message;
    socket.emit('data',data);//送信部分 });
  
```

- 番号4, 5 は Express.js というライブラリによってサーバを実装しており, HTML やクライアント側スクリプトをここから提供している.
- 番号6の GINEWrapper を記述することで GineBaseObject クラスライブラリを Node.js から使用している. GINEWrapper は以下の図2のようにラップされた構造になっている.

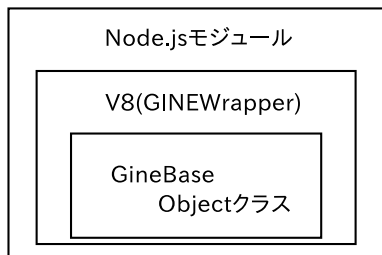


図2 GINEWrapper 構造

また, 今回のシステムで新規に必要な機能として GineBaseObject にクラスの名前一覧を取得する関数の追加とともに, 仮想ノードの作成のための関数である createInstanceOf() について root 権限でしか作成できないノードが権限が無いために作成できない場合のエラー処理について JavaScript でもキャッチできるように変更をした. 更に, NSpace.cpp には仮想ノードが使用している ptyname(/dev/pts/1) などを GineWrapper から直接呼び出す関数を追加した.

## 5 システムの実験

本研究で実装したシステムを用いて複数の OS からエミュレーションすることができるか実験する.

### 5.1 実験方法

今回, サーバ側は Ubuntu 14.04, クライアント側は動作環境外である OS の Windows 7 上で Google Chrome というブラウザを起動して実験した. まず, 右側のメニューから設置するノードの種類を選択して配置する. 次に, ノード同士を接続して IP アドレスの設定をする. 最後に, エミュレートを開始後に端末を起動して ping コマンドにより, 設定が保存されていることや仮想ネットワークが画面に描画されている通りに構築されているか確認する.

### 5.2 実験結果

仮想ネットワークを構築後, 各ノードにおいて ping で疎通確認をしたところ図3のように設定通りの値を表示する期待通りの動作をした. また, サーバ側のデバッグログで仮想ノードが正しく作成されていることや, コマンドの送受信がされていることを確認することができた. また, [Windows 7, Mac OS X, Ubuntu14.04] の三種類の OS で同様の実験をしたところ, Google Chrome, Firefox の二種類のブラウザ上で正常に動作することが確認できた.

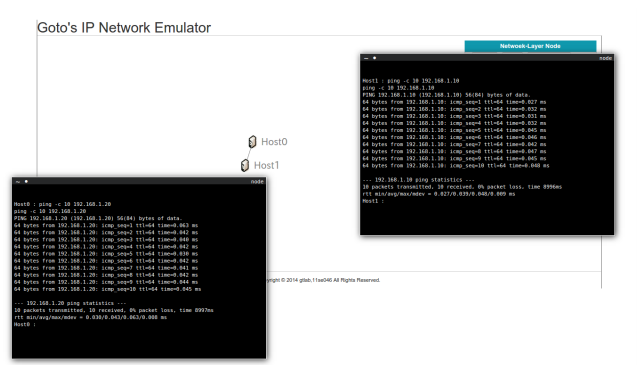


図3 GUI 端末での ping 疎通確認

## 6 おわりに

本研究により, ネットワークエミュレータ GINE の GUI を複数の OS から実行できるようになった. しかし, 現時点では最低限の機能しか実装できていない. 実装時に作成, 変更したプログラム及び HTML の行数は約 930 行であった. 今後の課題を以下に述べる.

1. ノード間を結ぶラインにイベントを設定することで回線遅延や帯域幅の設定を可能にする.
2. 端末の動作をより滑らかにするために, WebShell.js など他のライブラリの利用を検討する.

## 参考文献

- [1] Goto, K.: Network Emulator with Virtual Host and Packet Diversion, *Cyber Journals: Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunication(JSAT)*, Vol. 2, No. 2, pp. 13–20 (2012).
- [2] U.S. Naval Research Laboratory Networks and Communication Systems Branch: Common Open Research Emulator web page (accessed: Jan. 2015). <http://www.nrl.navy.mil/itd/ncs/products/core>.
- [3] 浅野洋介: 大規模ネットワーク構築のための GINE の管理機能の追加, 南山大学大学院数理情報研究科 2009 年度 修士論文 (2010).
- [4] 太田 翔, 今川敬太: GINE の GUI 機能の拡張, 南山大学 数理情報学部 情報通信学科 2010 年度 卒業論文 (2011).