

PC 画面配信システム TreeVNC の NAT への対応

125716B 氏名 伊波立樹 指導教員：河野 真治

Abstract

An implementation of Screen sharing system is presented. TreeVNC supports multicasting high resolution PC screen using tree structured network over PCs. Copying costs are distributed instead of the centralized heavy costs on a VNC server. It is also possible to change screen server dynamically without changing display cables. TreeVNC is useful in a lecture which contains 40-60 students or small seminar. We add several improvements on TreeVNC, such as new connection method for supporting NAT, screen change in a separated thread, multiple screen selection, notification of connection errors. We also show the delay time distribution in real environments

1 画面共有を利用したコミュニケーション

授業やゼミ等で、それぞれが PC 端末を持っている場合では、PC の機能を活かしたコミュニケーションが可能である。

通常の授業では先生の用意した資料、PC 画面を見ながら授業が進むことが多い。ゼミでは発表者を切り替えながら発表を行う。

画面配信システム TreeVNC[1] は発表者の画面を参加者の PC に表示する。そのため、参加者は手元の PC を使用しながら授業を受ける事が可能になる。更に発表者の切り替えの際もケーブルの差し替えずに共有する画面の切替が可能としている。

Tree VNC は VNC[2] を使用した画面配信を行っている。しかし通常の VNC では配信側に全ての参加者が接続するため、多人数の際の処理性能が落ちてしまう。Tree VNC では有線でネットワークに接続した参加者をバイナリツリー状に接続し、配信コストをクライアントに分散させる仕組みになっている。そのため、授業で先生の画面を表示する際、多人数の生徒が参加しても処理性能が下がらない。また、ツリーのルートが参照している VNC サーバーを変更することで、共有する画面の切替が可能となる。

しかし TreeVNC を授業やゼミで使用している中、様々な問題が発生した。NAT を越えたネットワーク接続に対応しておらず、遠隔地などで授業やゼミに参加することが出来ない。また、ゼミの際に、マルチディスプレイを使用して画面配信を行う際、すべての画面が配信され、不必要な画面まで表示されてしまう。

そこで、本研究では上記の問題点を解決し、TreeVNC の有用性を評価することで授業やゼミを円滑に行えることを目標とする。

2 画面配信システム TreeVNC

VNC(Virtual Network Computing) は、RFB プロトコルを用いて遠隔操作を行うリモートデスクトップソフトウェアである。

TreeVNC は Java を用いて作成された TightVNC(Tight Virtual Network Computing)[3] を元に作成されている。

TreeVNC はクライアント同士を接続させ、画面描画のデータを受け取ったクライアントが次のクライアントにデータを流す方式を取っている。また、サーバへ接続しに来たクライアントをバイナリツリー状に接続する(図1)。バイナリツリー状に接続することで、 N 台のクライアントが接続しに来た場合、画面配信の画像データをコピーする回数は従来の VNC ではサーバ側で N 回する必要があるが、TreeVNC では各ノードが 2 回ずつコピーするだけで済む。

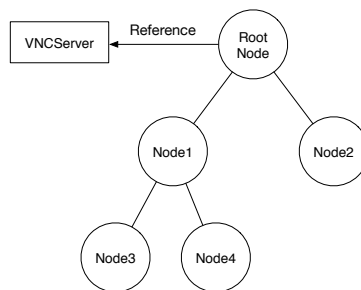


図 1: 構成される木構造

3 NAT への対応

遠隔地からでもゼミや授業に参加できるよう、NAT を越えたネットワークから TreeVNC への接続を可能にした。

図 2 に NAT を越えたネットワークからの接続を示す。別ネットワークから TreeVNC に参加する際、直接配信側のネットワークの Root Node に接続を行う。この接続を Direct Connection と呼ぶ。

Direct Connection した Node はそのネットワークの Root Node になる。そのネットワークの他の Node はそのネットワークの Root Node に接続し、木構造を生成する。

配信側の Root Node は Direct Connection で接続された Root Node に対して Framebuffer Update で 画像データを送信する。Framebuffer Update が送信された Root Node は そのネットワークの Node に対して Framebuffer Update を送信する。

これにより、NAT を越えたネットワークの画面共有が可能となる。

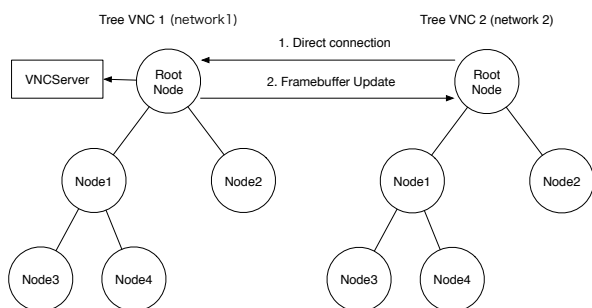


図 2: 遠隔地 Node からの接続

4 マルチディスプレイ対応

画面配信側の PC がマルチディスプレイの場合、VNC サーバーからは複数の画面全体の画像データが送信されてしまう。そこで、画面を共有する際、ディスプレイを選択させ、画面共有を行う機能を追加した [4]。

ディスプレイの情報は個々のクライアントでしか取得ができない。そのため、配信側は画面の切替を行う際に、ディスプレイを選択し、そのディスプレイの左上と右下の座標を取得する。その座標を Root Node への画面切り替えを要求する SERVER_CHANGE_REQUEST メッセージに付加させる。Root Node は 配信側の VNC サーバー に画像データを要求する FRAMEBUFFER_UPDATE_REPLY メッセージに送信された座標を付加する。VNC サーバーは要求された座標内の画像データを FRAMEBUFFER_UPDATE メッセージで Root Node に送信する。これにより、一画面のみの表示が可能となる。

図 3 は Display1 のみを画面共有する例を示している。

5 画面切り替えの安定化

画面切り替えを行う際、配信中の画面を停止した後に画面の切り替え処理を実行していた。しかし、切り替え先の VNC サーバーへの接続に時間がかかったり、切り替え先の設定ミスなどで画面の切替がスムーズに行えない場合があった。

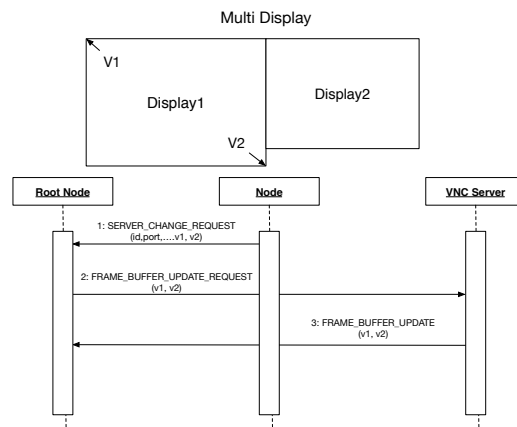


図 3: マルチディスプレイへの対応

そこで、画面切り替えを行う際に新しく切り替え用のスレッドを生成し処理する変更を行った。Root Node は SERVER_CHANGE_REQUEST を受け取ると、配信状態を維持したまま、画面切り替え用に新しくスレッドを生成する。切り替え用のスレッドは切り替えに必要なデータの設定、VNC サーバー への接続など通常の切り替え処理を実行する。切り替えが完了した後に、現在配信中の画面を停止し、画面の切替を行う。切り替え用のスレッドを用意することで、配信状況を維持したままスムーズな画面切り替えが可能になった。

6 クライアントへのエラー通知

TreeVNC には接続しているクライアントへのエラーの通知を行うことが出来なかった。そのため、画面切り替えを行う際に切り替え先が VNC サーバーの共有設定をしていない場合 Root Node に接続エラーのダイアログが表示されるという実装になっており、切り替えを行ったクライアントには通知されていなかった。

この問題を解決するために新しく ERROR_ANNOUNCE というメッセージを追加した。ERROR_ANNOUNCE メッセージは Root Node から木構造を辿りながら末端の Node に通信を行うメッセージで、エラー通知したいクライアントの Node Number と エラー内容の文字列を付加して送信する。付加した Node Number に一致する Node がメッセージを受け取ると、付加されたエラーの文字列をダイアログで表示する。エラー通知用のメッセージを追加することで、各々のクライアントに対して通知を行うことが可能になった。

7 TreeVNC の評価

VNC サーバー から受信する画像データ、TreeVNC で扱われるメッセージ通信は構成された木を伝って伝達される。接続する人数が増える毎に木の段数は増えていく。そ

ここで Root Node から木の末端の Node までの画像データ伝達の遅延を検証する実験を行った。

今回授業を受講している学生が TreeVNC を使用した状態で実験を行った。TreeVNC には最大で 17 名が接続していた。バイナリツリーで木を構成した場合、Node 数が 17 台だと深さが 4 となる。

図 4 は深さ 4 のち円の分布を示した散布図である。X 軸はメッセージ伝達にかかった秒数 (ms)、Y 軸は画像データのサイズ (Byte) である。画像データの伝達はほぼ 1 秒以内に収まっているが、容量が小さい場合でも時間がかかる場合がある。それはその送信の前に大容量の画像を送信した後の回線の Delay が残っているためだと考えられる。

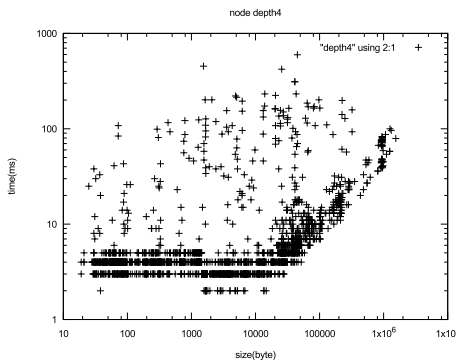


図 4: データサイズと遅延の関係: 深さ 4

また、この実験で 1 つだけ極端に遅い Node が見られた。このような Node を木に配置した場合、その Node の子 Node 以下に影響を及ぼすおそれがある。

8 ボトルネックになっている Node への対処

画像データを受け取る時間が遅い Node をそのまま木構造に配置しているとその子 Node 以下に影響を及ぼす。そのためネックになっている Node への対処が必要である。

ボトルネックになっている Node への対処として CHECK_DELAY_REPLAY メッセージを使用している。図 5 は 6 台の Node が接続している状態で Node1 がネックになっている場合の木の再構成を示している。CHECK_DELAY_REPLAY メッセージを受け取った Root Node はその Node がボトルネックになっているかの判断を行う。ボトルネックになっているなら Root Node の nodeList からその Node を削除する。削除した Node の場所には末端の Node を配置するように CONNECT_TO メッセージを送信する。nodeList から削除された Node は接続されたままなので、データの受信は行すが、木構造には入らないため、ネックになっている Node の下に新しい Node が繋がることはない。そのためネックになっている Node 以下に影響を及ぼすことがない。

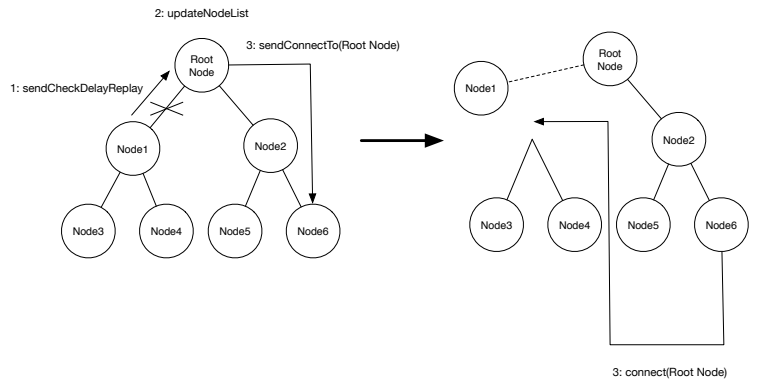


図 5: ボトルネックになっている Node への対処

9 まとめ

本研究では画面配信システム TreeVNC での NAT 対応、マルチディスプレイ対応、共有画面切り替えの安定化、各 Node へのエラー通知、TreeVNC の評価を行った。

NAT を越えに対応した Direct Connection という接続方法を確立し実装した。これにより、NAT を越えた別ネットワークのユーザーが TreeVNC に参加することが可能となった。

マルチディスプレイの場合に画面を選択して配信することが可能になった。

画面切り替えの際に切り替え用のスレッドを用意することでスムーズな画面切り替えが可能。

新しく ERROR_ANNOUNCE メッセージを追加することで各 Node にエラー通知を行えるようになった。

今回の画像データの遅延実験を行い、ボトルネックになっているノードがあることがわかった。また、ボトルネックになっているノードへの対処を行った。

また今後の課題として音声配信機能の実装、マルチディスプレイ時の表示画面サイズの調整を行う

参考文献

- [1] Yu TANINARI and Nobuyasu OSHIRO and Shinji KONO. Vnc を用いた授業用画面共有システムの実装と設計. 日本ソフトウェア科学会第 28 回大会論文集, sep 2011.
- [2] RICHARDSON, T., STAFFORD-FRASER, Q., WOOD, K. R., AND HOPPER,. A. virtual network computing, jan 1998.
- [3] TightVNC Software. <http://www.tightvnc.com>.
- [4] Tatsuki IHA and Shinji KONO. 有線 lan 上の pc 画面配信システム treenvnc の改良. 第 57 回 プログラミング・シンポジウム, jan 2016.