

# 分散フレームワーク Christie による画像配信システム TreeVNC の実装

## Implementation of screen delivery software TreeVNC by distributed framework Christie

165729B 氏名 安田亮 指導教員：河野 真治

### 1 研究目的

現代の講義やプレゼンなどでは、PC 画面で用意した資料を見ながら進行することが多い。参加者はプロジェクトに注視する必要があり、手元の PC と交互に見る必要がある場合両方に集中を割く必要があり、負担になる可能性がある。また、発表者が交代する際にケーブルの差し替えを行う必要があり、PC と接続するアダプターによっては正常に PC 画面を表示できない場合がある。

当研究室で開発している TreeVNC[?] は発表者の PC 画面を参加者の PC 画面に表示する画面配信ソフトウェアである。しかし既存の TreeVNC では画面共有する際の送信するデータ量が多いため、有線 LAN での使用に限られている。広く使われている無線 LAN に対応するために Multicast でのデータ通信の実装やデータの分割・圧縮を行い、TreeVNC の Multicast の可能性を評価する。

また、当研究室で開発を行なっている分散フレームワーク Christie で TreeVNC を再実装を行う。Christie を用いることで TreeVNC における長期間開発してきたことに伴うアルゴリズムの複雑さの解消および、メッセージ伝達速度の維持・向上の検証を行う。

### 2 VNC

VNC(Virtual Network Computing)[1] は、サーバ側とクライアント(ビューワー)側からなるリモートデスクトップソフトウェアである。遠隔操作にはサーバを起動し、クライアント側がサーバ側に接続を行うことで可能としている。また、動作には RFB プロトコル [2] を用いている。TreeVNC は VNC を利用した画面配信を行なっている。しかし通常の VNC では配信側の PC に、全ての参加者への配信を行う負担がかかってしまう。

RFB(Remote Frame Buffer) プロトコルとは、自身の PC 画面を Framebuffer と呼ばれるメモリ上に置かれた画像データをネットワーク上に送信し、他者の PC 画面に表示を行うプロトコルである。

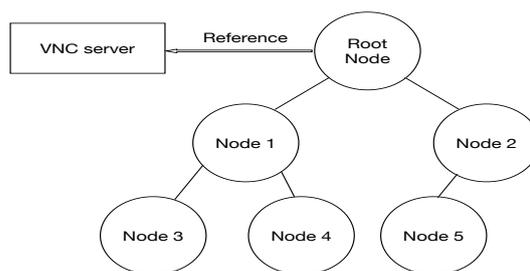


図 1: TreeVNC の接続方法

### 3 TreeVNC

TreeVNC はサーバに接続してきたクライアントをバイナリツリー状に接続する。接続してきたクライアントをノードとし、その接続したノードの下に新たなノード 2 つを接続していく。これにより、人数分のコピーと送信の手間を分散することが可能である(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てきた場合、従来の VNC ではサーバ側が N 回コピーを行なって配信する必要があるが、TreeVNC では各ノードが 2 回ずつコピーをするだけで配信が可能となる。バイナリツリールートのノードを Root Node と呼び、そこに接続されるノードを Node と呼ぶ。Root Node は子 Node にデータを渡す機能、各 Node の管理、VNC サーバから送られてきたデータの管理を行なっている。各 Node は、親 Node から送られてきたデータを自身の子 Node に渡す機能、子 Node から送られてきたデータを親 Node に渡す機能がある。

### 4 Multicast

画面配信のデータ量は膨大なため、現在の TreeVNC で VNCServer に無線 LAN 接続を行なった場合、画面配信の遅延が大きくなってしまふ。バイナリツリーを構成している中に無線 LAN で接続を行なっている Node が存在すると、その Node に接続されている全ての Node もその遅延の影響を受けてしまふ。ここで Wifi の Multicast の機能を用いることで、配信側にも Wifi を使用することが可能であると考

えられる。Tree Root は無線 LAN に UpdateRectangle を Multicast で一度だけ送信する。

Wifi の Multicast packet のサイズは 64Kbyte が最大となっている。HD や 4K の大きさの画面更新は  $8MB * 8byte$  となり圧縮前で 64MB 程度となる。これを圧縮しつつ、64Kbyte 毎のパケットに変換して送る必要がある。Wifi の Multicast packet は確実に送られることは保障されていない。通し番号を付けて欠落を検知することはできるが、再送処理は複雑であることが予想される。そこでまず Blocking について考察と実験を行う。

## 5 Blocking

RFB の UpdateRectangle は 1 つの UpdateRectangle の中に複数の Rectangle で構成されており、それぞれの Rectangle には encoding type を持っている。サーバからは ZRLE という、可逆圧縮可能な zlib 形式を encode した Rectangle が送信される。Rectangle には encode されたデータが指定された長さだけ入り、このデータは  $64 \times 64$  の tile にさらに分割されている (図 2)。

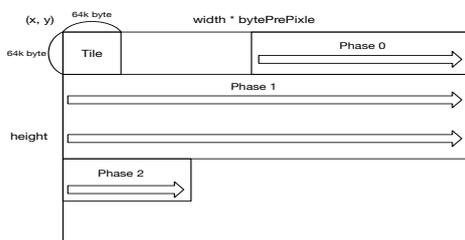


図 2: Rectangle の分類

このパケットを 64Kbyte に収まる 3 つの Rectangle に再構成する。この時に tile 内部は変更する必要はないが、Rectangle の構成は図 2 のように変わる。ZRLE を展開しつつ、パケットを構成する必要がある。

## 6 Christie

Christie[3] は本研究室で開発している分散フレームワークである。Christie は TreeVNC と同様 java で記述されており、以下のような概念を持っている。

- CodeGear(以下 CG)
- DataGear(以下 DG)
- CodeGearManager(以下 CGM)
- DataGearManager(以下 DGM)

CG は java のクラスやスレッドに相当し、DG は変数データに相当している。CGM はノードに相当し、CG, DG, DGM を管理する。DGM は DG を管理および格納を行う。

また、Christie には TopologyManager という機能が実装されている。TopologyManager は Christie 上の CGM(ノード) を Topology として静的・動的に形成するノードである。TopologyManager で TreeVNC の Node の生成 Node 間の接続を行うことで、TreeVNC が持っているアルゴリズムの複雑さの解消を図る。

## 7 まとめ

TreeVNC に Wifi 上の Multicast packet を用いる手法について考察した。Blocking は実装中であり、再圧縮の時間は全画面の場合でも実用的な時間で済むことが予想されている。Multicast で画面配信を行った際にどのくらいパケットロストするか不明であるため、さらなる実験が必要である。また、パケットロストした部分の検知・再送処理は TreeVNC をさらに複雑化させるため行わず、一定期間で画面データを全て送ることによって対処する。

Christie への再実装のタイミングは Blocking を一通り実装したのちに行う予定である。現在の TreeVNC はサーバ側を起動してからクライアント側の接続を行わなくてはならないが、TopologyManager でバイナリツリーの構成を行うことで待ち合わせが可能となり、どちらから起動しても問題なくなる。

その他残っている課題としてはマルチウィンドウの処理部分を行っている Retina の API が java9 以前と以降で変更があったため、java の最新版に対応できるように修正を行う。

## 参考文献

- [1] RICHARDSON, T., STAFFORD-FRASER, Q., WOOD, K. R., AND HOPPER, A. virtual network computing, jan 1998.
- [2] RICHARDSON, T., AND LEVINE, J. The remote framebuffer protocol. rfc 6143, mar 2011.
- [3] 照屋のぞみ. 分散フレームワーク christie の設計, March 2018.
- [4] 立樹伊波, 真治河野. 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予稿集, 第 2016 巻, pp. 29–37, jan 2016.