

平成 12 年度

学士学位論文

# 地域指向疑似 IX における

## WWW サーバ群の連携

How to combine WWW servers on PIX model

1010437 廣瀬 崇夫

指導教員 菊池 豊 助教授

2001 年 2 月 5 日

高知工科大学 情報システム工学科

# 要 旨

## 地域指向疑似 IX における WWW サーバ群の連携

廣瀬 崇夫

地域指向のインターネットトラフィック交換方式に PIX モデルがある。我々は、高知県において PIX モデルの実証実験 KPIX を行っている。PIX モデルでは多くのキャッシュサーバを介してトラフィックを交換する。このキャッシュサーバ間の連携には多くのバリエーションがある。

本稿では PIX モデルにおいて、キャッシュサーバ間の連携を効率よく行う WWW サーバ群の構成を提案する。また、キャッシュサーバとして Squid を用いての具体的な設定例を示す。この手法によると、効率的なリクエストの解決を可能とし、インターネット経路への負荷を軽減可能である。

キーワード PIX、地域 IX、キャッシュサーバ、Squid

# Abstract

## How to combine WWW servers on PIX model

HIROSE Takao

PIX is a region oriented model for exchanging traffic of the Internet. We have been working on KPIX, an implementation of PIX in Kochi prefecture. WWW traffic is exchanged via many proxy and/or cache servers on PIX. Therefore, there are many vary combinations of cooperative proxy servers.

This paper proposes how to combine WWW servers that perform cache exchange efficiently on PIX model. We will show a concrete example of a configuration proposal using Squid as proxy and cache service.

This method provides efficient cache exchange and reduction of the load of connection to the internet.

*key words* PIX, regional IX, cache, Squid

# 目次

第 1 章	はじめに	1
第 2 章	PIX	2
2.1	IX (Internet eXchange)	2
2.2	地域 IX	3
2.3	PIX モデル	4
2.3.1	HAN (Harmonizing Area Network)	4
2.4	KPIX(Kochi PIX)	8
2.4.1	KPIX における下位層の設計と構築	9
2.4.2	KPIX の特徴	11
2.5	KPIX の経路制御	12
2.5.1	PIX モデルの経路制御の問題点	12
2.5.2	KPIX での実装	13
第 3 章	コンテンツの共有	14
3.1	コンテンツの共有	14
3.2	WWW サーバ群の構成	15
3.2.1	参加組織による HAN の構成	15
3.2.2	KPIX における典型的な WWW サーバ群の構成例	16
3.2.3	WWW サービスに対する要求	16
3.2.4	キャッシュサーバ	17
3.2.5	サーバ群の連携パターン	18
3.3	参加組織コンテンツの共有	20
3.3.1	リクエストに対するサーバ間の連携	20

3.3.2	各サーバの動作	23
3.3.3	Squid によるサーバ群の実現	23
3.3.4	考察	26
3.4	参加組織以外のコンテンツの共有	28
3.4.1	リクエストに対するサーバ間の連携	28
3.4.2	各サーバの動作	29
3.4.3	Squid によるサーバ群の実装	31
3.4.4	考察	33
3.5	クライアントの設定	34
3.5.1	WWW ブラウザのプロキシ設定	35
<b>第 4 章</b>	<b>まとめ</b>	<b>37</b>
	<b>謝辞</b>	<b>39</b>
<b>付録 A</b>	<b>KPIX の歴史</b>	<b>40</b>
<b>付録 B</b>	<b>KPIX 活動経緯</b>	<b>44</b>
B.1	Squid 動作モデルの設計	44
B.1.1	設定変更にかかるコスト	45
	キャッシュサーバ連携モデル	45
B.2	参加組織のコンテンツを共有する Squid の設定	46
B.2.1	参加組織による HAN の構成	47
B.2.2	KPIX における典型的な WWW サーバ群の構成例	48
B.2.3	WWW サービスに対する要求	49
B.2.4	サーバ群の連携パターン	49
B.2.5	リクエストに対するサーバ間の連携	51
B.2.6	各サーバの動作	54

B.2.7	Squid によるサーバ群の実現	54
B.3	コンテンツを共有するサーバ間の連携	56
B.3.1	リクエストに対するサーバ間の連携	57
B.4	コンテンツを共有する Squid の設定	60
B.4.1	各サーバの動作	60
B.4.2	Squid によるサーバ群の実装	60
B.5	NTP による時刻同期	63
B.5.1	NTP とは	64
B.5.2	NTP の必要性	65
B.5.3	NTP の構成	65
B.5.4	NTP の設定	65
B.6	MRTG による測定	67
B.7	PIX モデルの諸問題とその解決手法	68
B.7.1	PIX モデル設計時の問題点	69
B.7.2	問題点に対する解決法	69
B.7.3	KPIX での対応	70
B.8	富山国体映像中継	70
B.8.1	JGN を利用したデジタルビデオによる高品質映像の配信実験	70
B.8.2	映像ソース	71
B.8.3	トポロジー	71
	菊池研究室内の液晶テレビに表示	72
	WMT を用いて KPIX へストリーム配信	72
	RealSystem を用いて KPIX へストリーム配信	73
	ASX200 を用いて名古屋と岐阜へ再配信	74
B.8.4	ストリーム配信の設定とスペック	74
B.8.5	KPIX への配信結果	74

B.9	ギガビットシンポジウム 2000 中継 . . . . .	76
B.9.1	映像ソース . . . . .	77
B.9.2	トポロジー . . . . .	77
	菊池研究室内の液晶テレビに表示 . . . . .	78
	WMT を用いて KPIX へストリーム配信 . . . . .	79
	DVTS を用いて富山と山梨に再配信 . . . . .	80
B.9.3	ストリーム配信の設定とスペック . . . . .	80
B.9.4	KPIX への配信結果 . . . . .	80
B.9.5	まとめ . . . . .	81
B.10	pac ファイルの作成 . . . . .	82
B.10.1	pac ファイルとは . . . . .	82
B.10.2	pac ファイルの目的 . . . . .	82
B.10.3	pac ファイルの記述 . . . . .	83
B.10.4	pac ファイルの適用 . . . . .	84
B.11	コストモデル . . . . .	85
	参考文献 . . . . .	86

# 目次

2.1	HAN を構成する典型的な LAN の例 . . . . .	5
2.2	HAN を構成する最も小さい LAN の例 . . . . .	6
2.3	HAN を構成するより大きな LAN の例 . . . . .	7
2.4	4 つの LAN を持つ HAN の例 . . . . .	8
2.5	インターネットから見た HAN . . . . .	9
2.6	地域イントラネットとしての HAN . . . . .	10
2.7	KPIX の無線リンク . . . . .	11
3.1	参加組織による HAN の構成 . . . . .	16
3.2	KPIX における WWW サーバ群の構成 . . . . .	17
3.3	サーバ群の全連携パターン . . . . .	18
3.4	制御ポリシーを満たす連携パターン . . . . .	19
3.5	WWW サービスへの要求を満たす連携パターン . . . . .	19
3.6	最善なサーバ群の連携パターン . . . . .	20
3.7	PIX 上の公開コンテンツに対するリクエスト . . . . .	21
3.8	キャッシュされた公開コンテンツに対する同組織内からのリクエスト . . . . .	22
3.9	キャッシュされた公開コンテンツに対する他組織内からのリクエスト . . . . .	22
3.10	非公開コンテンツに対するリクエスト . . . . .	23
3.11	インターネット上の公開コンテンツに対するリクエスト . . . . .	24
3.12	インターネット上の公開コンテンツに対するリクエスト . . . . .	29
3.13	キャッシュされた公開コンテンツに対するリクエスト . . . . .	30
B.1	KPIX のトポロジー . . . . .	47
B.2	参加組織による HAN の構成 . . . . .	48

B.3 KPIX における WWW サーバ群の構成 . . . . .	49
B.4 サーバ群の全連携パターン . . . . .	50
B.5 制約を満たす連携パターン . . . . .	50
B.6 KPIX におけるサーバ群の連携パターン . . . . .	51
B.7 PIX 上の公開コンテンツに対するリクエスト . . . . .	52
B.8 キャッシュされた公開コンテンツに対する同組織内からのリクエスト . . . . .	52
B.9 キャッシュされた公開コンテンツに対する他組織内からのリクエスト . . . . .	53
B.10 非公開コンテンツに対するリクエスト . . . . .	53
B.11 インターネット上の公開コンテンツに対するリクエスト . . . . .	54
B.12 インターネット上の公開コンテンツに対するリクエスト . . . . .	59
B.13 キャッシュされた公開コンテンツに対するリクエスト . . . . .	60
B.14 日間キャッシュヒット . . . . .	68
B.15 週間キャッシュヒット . . . . .	68
B.16 富山国体中継全体トポロジー . . . . .	71
B.17 富山国体中継トポロジー . . . . .	73
B.18 富山国体中継全体トポロジー . . . . .	78
B.19 ギガビットシンポジウム 2000 トポロジー . . . . .	79
B.20 PIX モデルの月額回線維持費用例 . . . . .	85

# 表目次

2.1	KPIX 実験協議会参加組織	8
2.2	無線 LAN ユニット能力一覧	12
3.1	組織 X 内キャッシュサーバ A の動作	24
3.2	組織 X に対応する HAN 内キャッシュサーバ $\alpha$ の動作	25
3.3	Squid A の設定	26
3.4	PIX Squid $\alpha$ の設定	27
3.5	組織 X 内キャッシュサーバ A の動作	30
3.6	組織 X に対応する HAN 内キャッシュサーバ $\alpha$ の動作	31
3.7	Squid A の設定	32
3.8	PIX Squid $\alpha$ の設定	34
B.1	KPIX トポロジーと設定量	47
B.2	組織 X 内キャッシュサーバ A の動作	55
B.3	組織 X に対応する HAN 内キャッシュサーバ $\alpha$ の動作	56
B.4	Squid A の設定	57
B.5	PIX Squid $\alpha$ の設定	58
B.6	組織 X 内キャッシュサーバ A の動作	61
B.7	組織 X に対応する HAN 内キャッシュサーバ $\alpha$ の動作	62
B.8	Squid A の設定	63
B.9	PIX Squid $\alpha$ の設定	64
B.10	MRTG 測定項目一覧	69
B.11	ストリームサーバの設定	74
B.12	ストリームエンコーダの設定	75

B.13 ストリームサーバの設定 . . . . .	75
B.14 ストリームエンコーダの設定 . . . . .	76
B.15 スペック比較 . . . . .	76
B.16 KPIX での富山国体中継視聴結果 . . . . .	77
B.17 ストリームサーバの設定 . . . . .	81
B.18 ストリームエンコーダの設定 . . . . .	81
B.19 KPIX でのギガビットシンポジウム 2000 中継視聴結果 . . . . .	82

# 第 1 章

## はじめに

著者らはインターネットトラフィック交換モデルである PIX モデルを提唱した [菊池 99]。また高知県において、地域情報化プロジェクトである Kochi 2001 Plan の下に、県内の産官学の共同で「KPIX 実験研究協議会」を 1997 年 7 月に組織し、実験を行っている [菊池 00a][西内 00][正岡 00]。KPIX とは、”Kochi Pseudo Internet eXchange” の略である。

キャッシュサーバの運用については、トラフィック解析によるキャッシュサーバの評価が報告されている [西川 97] [笠原 97] [安田 99]。また、キャッシュサーバの運用技術として透過型キャッシュやリバースプロキシなどの報告がある [鍋島 99] [鍋島 00]。

PIX モデルでは多くのキャッシュサーバを介してトラフィックを交換する。このキャッシュサーバ間の連携には多くのバリエーションがある。

本稿では、キャッシュサーバ間の連携を効率よく行う WWW サーバ群の構成を提案する。以降、第 2 章では、著者らが提唱した PIX モデルについて、その経緯と概要について述べる。また、高知県において行われている KPIX 実験研究協議会について述べる。第 3 章では、PIX モデルにおけるキャッシュサーバ間の連携パターンについて述べる。また、KPIX において Squid を用いて実装した連携パターンについて述べる。最後に第 4 章でまとめを行う。

# 第 2 章

## PIX

本章では、まず、地域 IX や、PIX モデルが要求されるようになった背景についての概要を述べる。次に、地域 IX 解決し得ない問題について述べる。そして、PIX モデルの基本概念について述べたのちに、高知県にて行っている PIX モデルの実証実験 KPIX について述べる。

### 2.1 IX (Internet eXchange)

IX とは ISP (Internet Service Provider) を相互に接続する仕組みである。IX には、OSI 参照モデルの第 2 層であるデータリンク層で交換する仕組みと、第 3 層であるネットワーク層で交換する方式がある。なお、機能する OSI 階層のいかんに関わらずトラフィックを交換する機構を IX と呼ぶ場合がある。これを広義の IX、第 2 層ないしは第 3 層による交換を指す場合を狭義の IX とし、以降は IX の狭義の IX を指すものとする。IX は大都市を中心とした交換を行う。このことは以下の観点から考えると問題である。

- 対象障害

通信経路上に障害が生じた際、別の地域での障害であっても、通信に影響を与える。

- レスポンス・スループット

大都市を中心の通信経路であることにより、地域内での通信のレスポンスやスループットが他地域の通信の混雑状況に影響を受ける。

- 人的資源や経済活動が地域に根付かない

トラフィックが大都市中心に交換されることにより、それを支える技術者や経済活

動も大都市に集中する状況が起こる。大手 IPS や IX に近い場所にデータセンターを置く事が妥当であり、トラフィックやコンテンツも集中する。

- 通信経路に対する主体制・自治

大都市を中心とする IP 交換のトポロジーを持つと、経路制御の主体が大手 ISP や IX になる。上で述べたような問題があることが分かって、さらにそれが地域内のトラフィックであっても、問題は地域内では解決できない。

## 2.2 地域 IX

大都市中心の交換による問題を解決するために、大都市以外に IX を置く動きが活発である。これを地域 IX と呼ぶ。

地域 IX には、IX の持つ問題の一部を解決するものの、IX 技術が持つ問題点をそのまま継承する側面もある。以下では地域 IX の持つ問題点を指摘する。

- ネゴシエーションコスト

ISP などの経路制御を統括する組織との協調に多くの労力が必要となることがある。インターネットの経路制御は BGP4 を用いるルータ間で AS (Autonomous System) 全てに関する経路情報が交換される。このため、以下のような欠点がある。

- 経路制御が複雑になり管理に高い技術が要請される
- 制御の影響を管理単位の組織で閉じることができない

技術者のスキルが十分に高くないと経路制御の混乱を招きやすい上に、一旦混乱を起こした場合には影響がインターネット全体に広がる。このため、地域 IX を計画した場合には、安定な運用を維持したい ISP などの理解と協力を得る事は難しい。

- スケーラビリティ

地域 IX の最も大きな問題は、比較的大きな地域による地域 IX が成功すると仮定しても、さらに小さな単位の組織が同様の展開をするための環境を必ずしも提供しない事にある。地域 IX 導入には高い技術者を必要とするので、地域 IX を導入す

るコミュニティが小さくなればなるほど導入は難しくなる。このため地域 IX を導入する事は、コミュニティに則したネットワーク技術を導入する観点からは、望ましいとはいえない。

- 効率

密なコミュニティであるほど、ユーザが同一コンテンツを参照する機会は多くなる。このような状況では IP データグラムの交換をスムーズにするよりも、コンテンツの繰り返し転送を減少させる方が効率が大きい。コミュニティ構造とネットワーク構造が近いほどアプリケーション層で共有できるコンテンツのトラフィックが増え重複転送を避ける事による効率改善を促進する事が可能になると推測できる。しかし、IX は OSI 第 3 層以下での交換しか行わないので、この状況を活かした交換の効率化には全く寄与する事ができない。

- 人材

地域 IX では、経路制御に高い技術レベルが要求されるため、いかに管理者を確保するかという問題がつかまとう。

## 2.3 PIX モデル

PIX (Pseudo IX) モデルとは、直観的にいえば地域に閉じたイントラネットを構成し、インターネットとイントラネットとのトラフィック交換は全てアプリケーション層で行うようなモデルである [菊池 99]。

### 2.3.1 HAN (Harmonizing Area Network)

PIX モデルの構成要素である HAN について説明する。PIX モデルにおいて経路制御の単位を HAN と呼ぶ。HAN は地域内での IP データグラムの到達可能域であり、1 つ以上の組織 (LAN) から成る。あるネットワークが PIX モデルを用いている場合には、そのネットワークは 1 つ以上の HAN により構成されている。PIX モデルにおける HAN は BGP4

における AS の概念に相当する。HAN を構成する各 LAN はインターネットへの通信路と HAN の通信路との 2 つの通信路を持つ。各 LAN は以下の 3 つの部分から成る。

1. インターネットから IP データグラム到達可能
2. HAN 内で IP データグラム交換可能
3. それ以外

前 2 者の重複している部分を HAN ボーダと呼び、以下の条件を満足するものとする。

- 少なくとも 1 つのホストが存在する
- グローバルアドレスが与えられている
- トランジット (HAN ボーダを越えるような IP データグラムのフォワード) は行わない

HAN を構成する典型的な LAN のトポロジーを図 2.1 に示す。

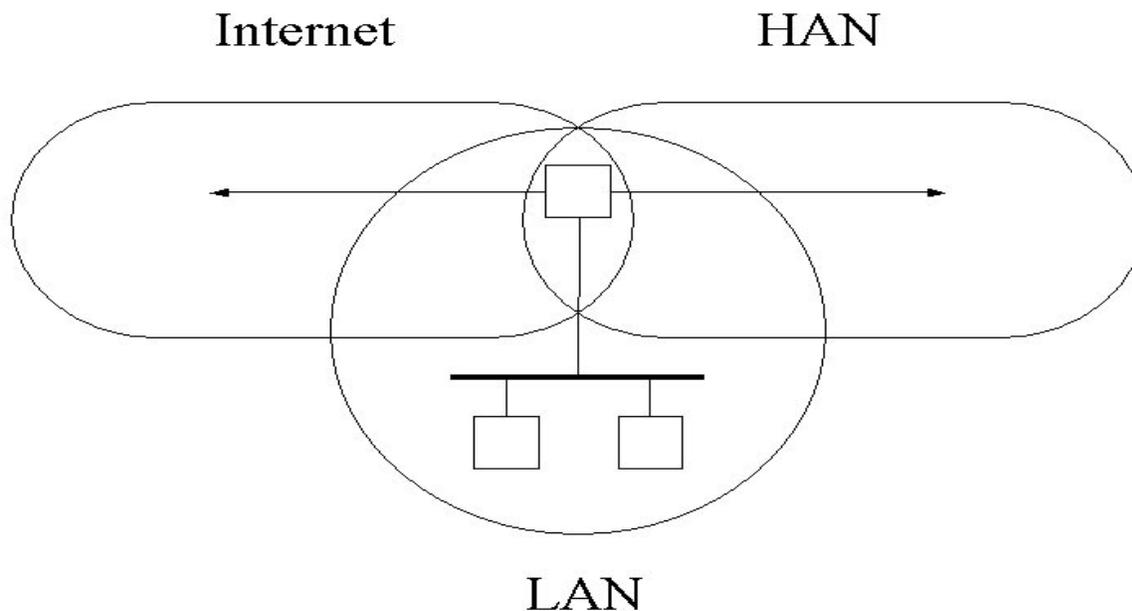


図 2.1 HAN を構成する典型的な LAN の例

四角はホストを示し、円と長円は IP が直接到達可能な領域を示している。左側の長円はインターネットを、右側の長円は HAN を、円は LAN を示している。HAN ボードにファイアウォールを設置し、LAN のその他のホストは HAN からインターネットからも IP 不到達であり、IP のトランジットは行わない。

HAN の一部を構成するような最も簡潔な LAN の形態は、図 2.2 に示すような HAN ボードにただ 1 台のホストが存在する場合である。

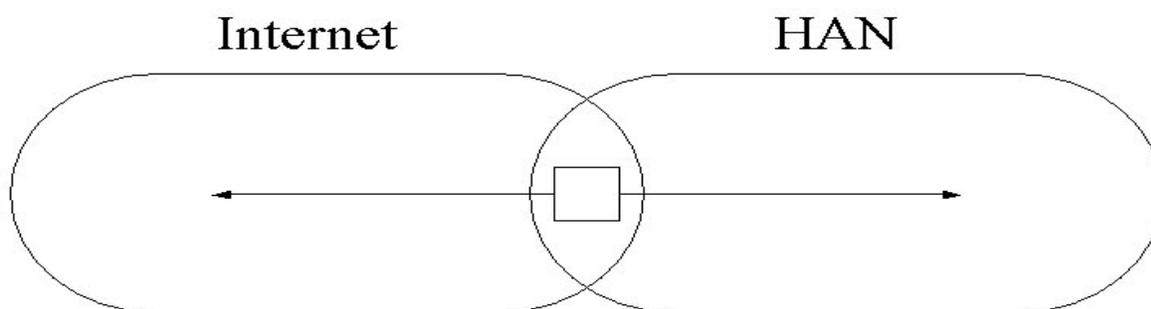


図 2.2 HAN を構成する最も小さい LAN の例

長円の重複する部分がホスト 1 台から構成される LAN である。より複雑な LAN の例は図 2.3 に示す。

これらのトポロジーを持つ LAN が集まって HAN 全体を構成する。図 2.4 は 1 つの HAN の例である。

図 2.4 の場合、LAN は、A、B、C、D の 4 つより構成される。この図でも円や曲線で囲まれた領域が IP の到達可能域を示している。各 LAN の HAN ボードは、ISP1、ISP2、ISP3、ISP4 を経由してインターネットと IP の交換を行う。

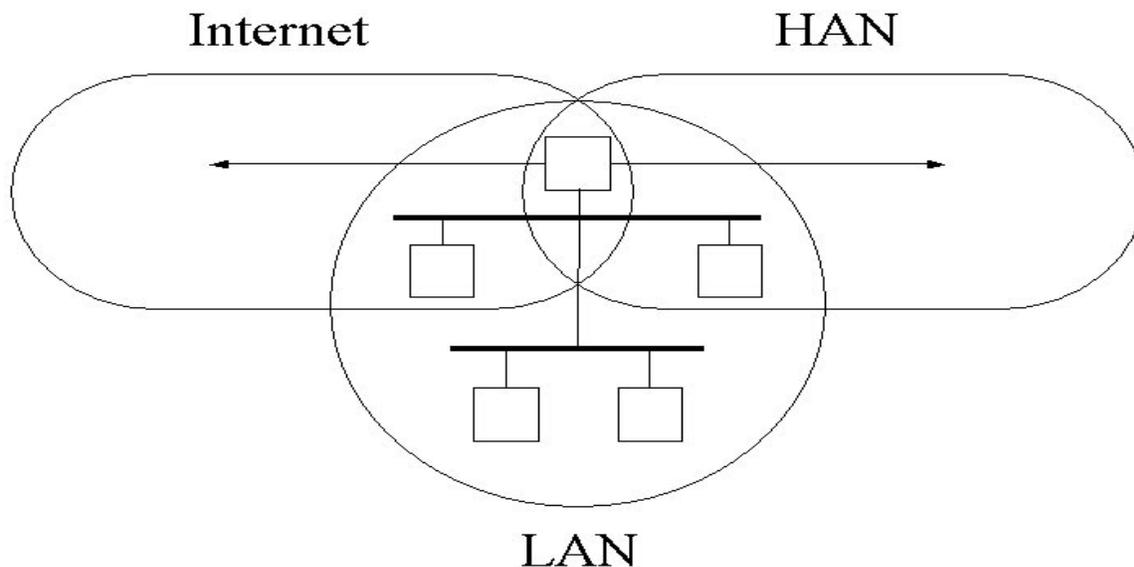


図 2.3 HAN を構成するより大きな LAN の例

インターネットと HAN 内のホストとは HAN ボーダを越えて IP を交換することは無い。したがって、IP データグラムの到達可能性の視点で見ると、インターネット側からはいくつかの LAN が ISP 経由で接続されているようにしか見えない (図 2.5)。

同様に HAN の内側からは、いくつかの LAN が HAN を経由して IP 接続されている地域イントラネットのようにしか見えない (図 2.6)。

IP データグラムの到達保証が HAN の十分条件である。トポロジーを含むデータリンク層以下の構成は、HAN を構成する地域の実情に応じて決定することになる。たとえば、中核となるルータを準備し、そこから各 HAN ボーダに放射状に通信路を張っても構わないし、HAN ボーダ全てを直線上に接続しても構わない。また、データリンク層を構成する技術が Ethernet であっても専用線であってもスペクトラム拡散方式の無線であっても構わない。

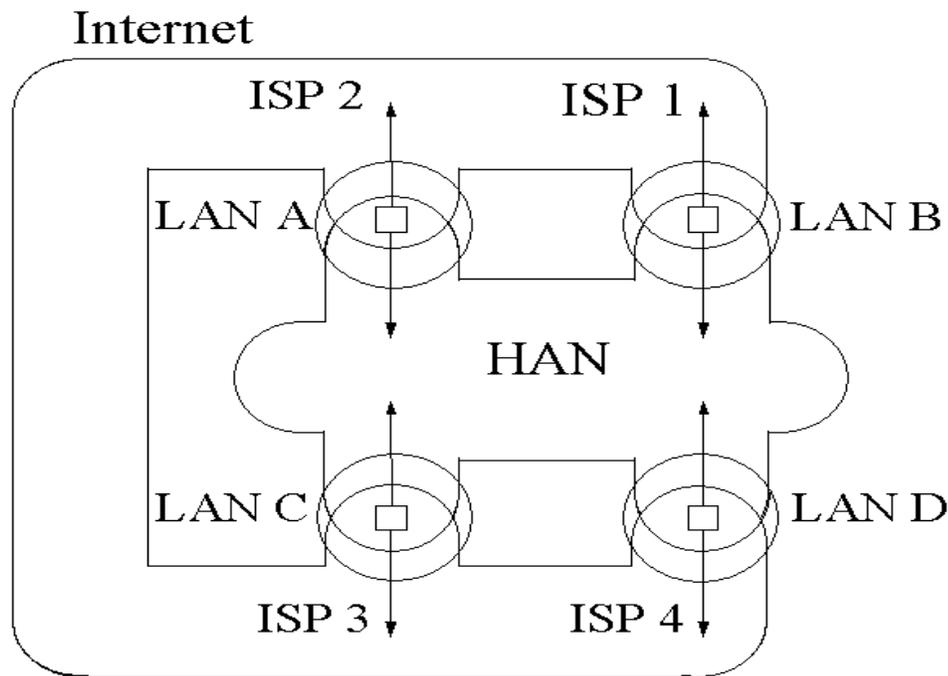


図 2.4 4つのLANを持つHANの例

## 2.4 KPIX(Kochi PIX)

我々は産官学からなる KPIX 実験研究協議会を 1998 年 7 月 13 日に設立し、PIX モデルの実用化に向けた研究活動を行っている。また、この協議会は高知県での地域情報化プロジェクトである KOCHI 2001 PLAN で定める協議会である。そして、事務局を高知県企画振興部情報企画課に置く。本協議会参加メンバーの組織(表 2.1)を中心として、PIX モデルの実証実験を行っている。我々は、KPIX の構築・運営を通して、PIX モデルの構築・運用ノウハウの確立を目的としている。

産業	官庁	学術
高知システムズ	高知県情報企画課	高知工科大学
シティネット	高知県工業技術センター	高知大学
富士通高知システムズ		高知高専

表 2.1 KPIX 実験協議会参加組織

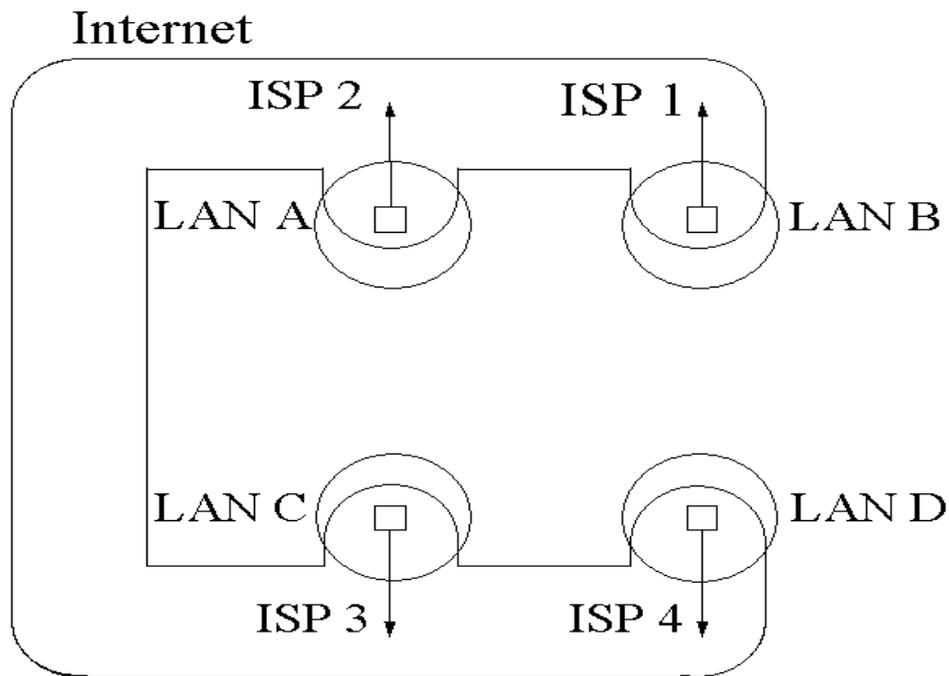


図 2.5 インターネットから見た HAN

### 2.4.1 KPIX における下位層の設計と構築

本節では、KPIX の下位層の設計と構築について述べる。ここで述べる内容は PIX モデルとは直接関係がない。これは、PIX モデルがネットワーク層以上についての概念であるためである。しかし、実際には PIX モデルは地域での応用を目指しているため、地域内でのデータリンクをどのように構築するかという問題は間接的に関係してくる。

KPIX では、HAN を構成するためのインフラに一部 KCAN (Kochi City-size Area Network)[今井 00] を使用している。一部とした理由として、KPIX は独自の専用線として無線 LAN ユニットを使用しているためである。KCAN は高知市・南国市をまたがる東西におよそ 20km、南北におよそ 10km を無線 LAN ユニットを使用し構築している。ネットワークとしては、プライベート IP アドレスを中央基幹セグメント・東部セグメント・西部セグメント・正蓮寺セグメント・土佐希望の家セグメントの 5 つのセグメントに分割し運用している。以下より、KPIX のインフラとして KCAN の無線ネットワークを含むものとする。

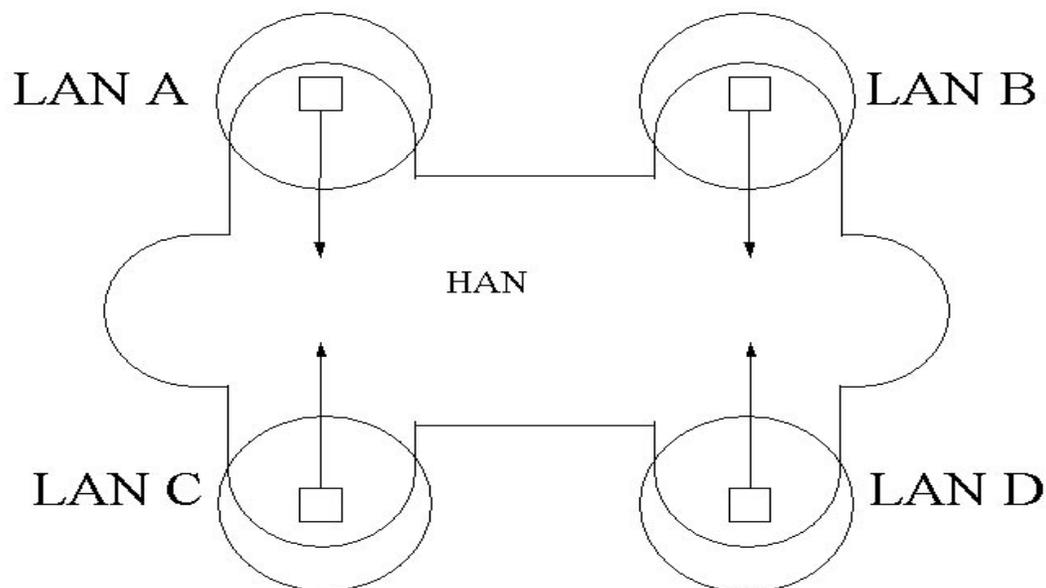


図 2.6 地域イントラネットとしての HAN

KPIX のネットワークを図 2.7 に示す。この図で四角は中継ポイントを、丸は接続ポイントを示している。

無線 LAN をデータリンクとして選択したのは、初期投資を除けばわずかなランニングコストでの運用が可能であるためである。無線 LAN ユニットの、株式会社コーラス社製<sup>\*1</sup> の「LAN Any-where B10」とアイコム株式会社製<sup>\*2</sup> の「BR-200」を選択した。これらの簡単なスペックを図 2.2 に挙げる。KPIX を構成する HAN はこれらの無線 LAN ユニットを組み合わせて構築されている。

これらの無線 LAN ユニットの通信方式には 2.4GHz 帯のスペクトラム拡散方式を使っている。また、これらの無線 LAN ユニットの直接拡散方式 (DirectSequence Spread Spectrum、以下 DSSS とする) を採用している。スペクトラム拡散方式の無線を選択したのは以下の利点があるためである。

\*1 <http://www.callus.co.jp/>

\*2 <http://www.icom.co.jp/>

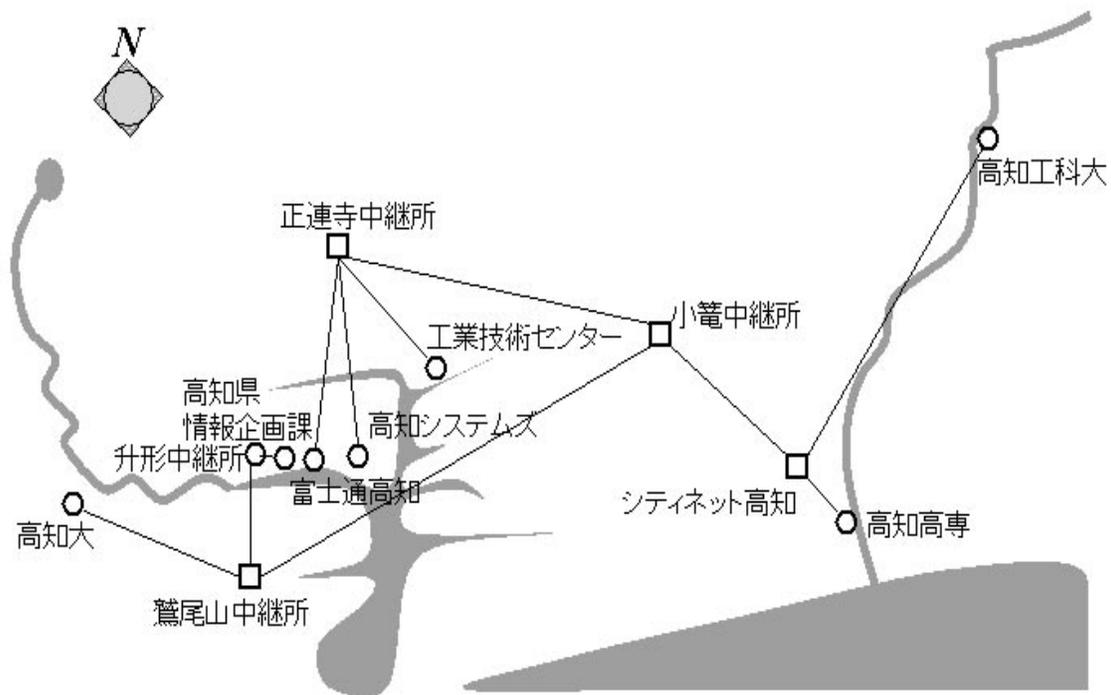


図 2.7 KPIX の無線リンク

- 無線局免許、無線従事者免許が不要である
- 秘話製、秘匿性に優れている
- 天候の変化による影響を受けにくい
- 高速の伝送路としては、初期コスト及び運用コストが低廉である
- 小型、軽量、小消費電力の割に長距離（数 km）の通信が可能である

### 2.4.2 KPIX の特徴

KPIX の特徴を述べる。

KPIX はデータリンクとして無線 LAN を用いている。これにより、HAN を構成する回

	通信速度	通信距離 (カタログ値)	通信方式
B10	11Mbps	約 5km	スペクトラム拡散方式 (DSSS)
BR-200	2Mbps	約 2km	スペクトラム拡散方式 (DSSS)

表 2.2 無線 LAN ユニット能力一覧

線の維持費が安価になっている。

現在、KPIX で交換するアプリケーションプロトコルは HTTP である。HTTP を交換するために、フリーのプロキシキャッシュサーバである Squid を使っている。また Squid の動作連携は、PIX モデルの基本動作連携である 12 通りとは異なる構成を取っている。これは、KPIX 参加組織のポリシーを満たす構成を取ったためである。

KPIX は将来、ストリーム型のデータをキャッシュし、地域にとって有益なコンテンツの増加を目指す。

以降、一般的な PIX モデルのことを「PIX モデル」、高知県で PIX モデルの実証実験として構築しているものを「KPIX」と呼ぶ。

## 2.5 KPIX の経路制御

KPIX では PIX モデルの経路制御設計とは異なった設計、実装を行っている。この章では KPIX の経路制御設計の PIX と異なる点を挙げ、KPIX における設計を示す。

### 2.5.1 PIX モデルの経路制御の問題点

PIX モデルの経路制御ポリシーでは、HAN ボードと HAN とは IP reachable である。すなわち、全ての組織内の HAN ボードは、互いに IP reachable である。そのため、HAN ボードを互いに IP reachable にするために設定を行う必要がある。しかし、PIX で接続される各組織はそれぞれの組織内の経路制御に関するポリシーを持っている場合がある。その

場合、ポリシーによっては HAN ボードを互いに IP reachable にするための設定が行えない場合がある。このような場合は、ゲートウェイを設置する場所が IP reachable でなくなり、PIX モデルが要求するトラフィック交換のモデルを実現できない可能性がある。

## 2.5.2 KPIX での実装

KPIX では以下の理由により、PIX モデルの経路制御ポリシーとは異なったポリシーを採用している。

- 下位層として KCAN のリンクを利用している

KCAN は KPIX とは別のネットワーク組織であるため、KPIX の経路情報が KCAN に流れるのは好ましくない。

- KPIX で利用しているルータが OSPF を実装していない

KPIX 参加組織の多くは、組織内の経路制御を OSPF を用いて行っていたが、KPIX で採用したルータが OSPF に対応しておらず、経路情報の交換が不可能であった。

PIX 参加組織の数が少ない場合は、静的ルーティングを行うことで上記のような問題がある場合でも PIX の設計に基づいた経路制御設計が可能である。しかし、その場合は PIX 参加組織の増減に伴って経路情報の更新が必要となり、管理コストが増加するため PIX のメリットが減少する。

KPIX では PIX モデルにおける HAN ボードを作らず、各組織内 LAN と HAN とは IP reachable ではない。そこで各組織内 LAN と HAN とを繋ぐルータの両側にキャッシュサーバを設置し、その二つのホストの間のみに対する静的経路を設定している。つまり、二台のゲートウェイをセットにして、組織内 LAN、HAN 両方から同様にできるように設定し、あたかも一台の HAN ボードが存在するかのように見せている。

## 第 3 章

# コンテンツの共有

本章では、PIX モデルにおけるコンテンツの共有について述べる。コンテンツの共有には参加組織の WWW コンテンツの共有と参加組織以外の WWW コンテンツの共有の 2 つに大別できる。

ここでは、まず参加組織のコンテンツを共有する場合において具体的なサーバ間の連携を示す。そして、各サーバごとの動作を示し、Squid による具体的な設定例を示す。次に参加組織の WWW コンテンツと参加組織以外の WWW コンテンツとを共有する場合においても同様に述べる。最後にクライアントが KPIX を利用するための設定について述べる。

### 3.1 コンテンツの共有

インターネットトラフィックの大部分は HTTP によって占められている。また、地域情報化の進行により、地方公共団体の提供するコンテンツや、地域指向のコンテンツが充実し、地域の人々の求めるコンテンツは重なる傾向にあると予測される。このような状況ではインターネットトラフィックを効率的に交換するよりも、コンテンツの重複転送を抑制する方が、ネットワークへの負荷の軽減が期待できる。

KPIX では、WWW コンテンツ (以降、コンテンツという) を HTTP で交換する PIX モデルをキャッシュサーバを用いて構築した。キャッシュサーバを用いることにより、KPIX 内での同一コンテンツの重複転送の抑制が可能である。また、コンテンツを交換するために、各組織の HAN ボーダにキャッシュサーバを設置し、必要に応じて各参加組織内にもキャッシュサーバを設置した。これらのキャッシュサーバを互いに連携させることにより、

HAN 内でのコンテンツの共有を実現した。

コンテンツの共有は、参加組織のコンテンツの共有と参加組織以外のコンテンツの共有の2つに大別できる。参加組織のコンテンツの共有は、参加組織の提供するコンテンツを HAN を経由して取得することである。また、HAN を経由することにより、コンテンツはキャッシュされ、HAN 内でのリクエストの解決を可能とすることである。参加組織以外のコンテンツの共有は、参加組織が自組織のインターネット経路を使って取得したコンテンツを HAN 内にキャッシュさせ、以降は HAN 内でそのコンテンツへのリクエストを解決可能とする。

HAN 内でコンテンツの共有を行うことにより、効率的なリクエストの解決を実現し、ネットワークへの負荷の軽減を可能とする。

## 3.2 WWW サーバ群の構成

ここでは、KPIX で構築した WWW サーバ群の構成を示す。

### 3.2.1 参加組織による HAN の構成

PIX では、以下の経路制御ポリシーを満たす HAN を構成する（図 3.1）。

- HAN 内は IP Reachable である。
- HAN と Internet は IP Reachable でない。
- 組織内のコンピュータと、組織に対応する HAN 内のサーバとは IP Reachable である。

円は IP Reachable な領域を示す。HAN と各組織との重なりは互いに IP Reachable な HAN ボードである。HAN は Internet と IP Reachable でないため、各組織のインターネット経路を経由しないと IP の交換は行えない。

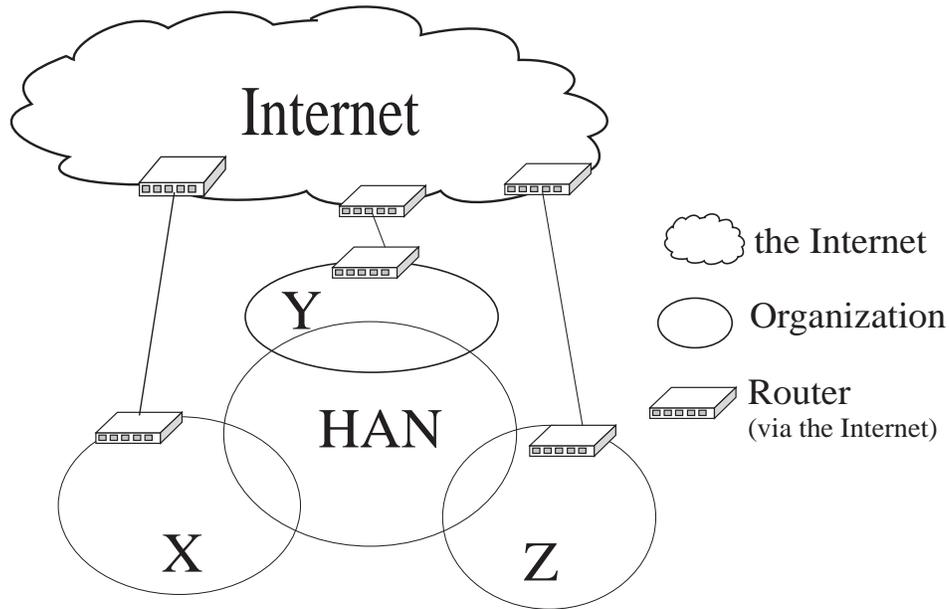


図 3.1 参加組織による HAN の構成

### 3.2.2 KPIX における典型的な WWW サーバ群の構成例

ここでは、以降の説明のために PIX モデルにおける WWW サーバ群の構成例を示す（図 3.2）。

各組織は以下で構成される。

- コンテンツを提供するサーバ (図中 I, II, III)
- 組織内で運用するキャッシュサーバ (図中 A, B, C)
- HAN で各組織に対応するキャッシュサーバ (図中  $\alpha, \beta, \gamma$ )
- インターネット経路
- 各組織のクライアント (図中 1, 2, 3, 4, 5)

### 3.2.3 WWW サービスに対する要求

ここでは PIX 参加組織からの WWW サービスに対する要求を述べる。

- HAN 内でできるだけ解決する。

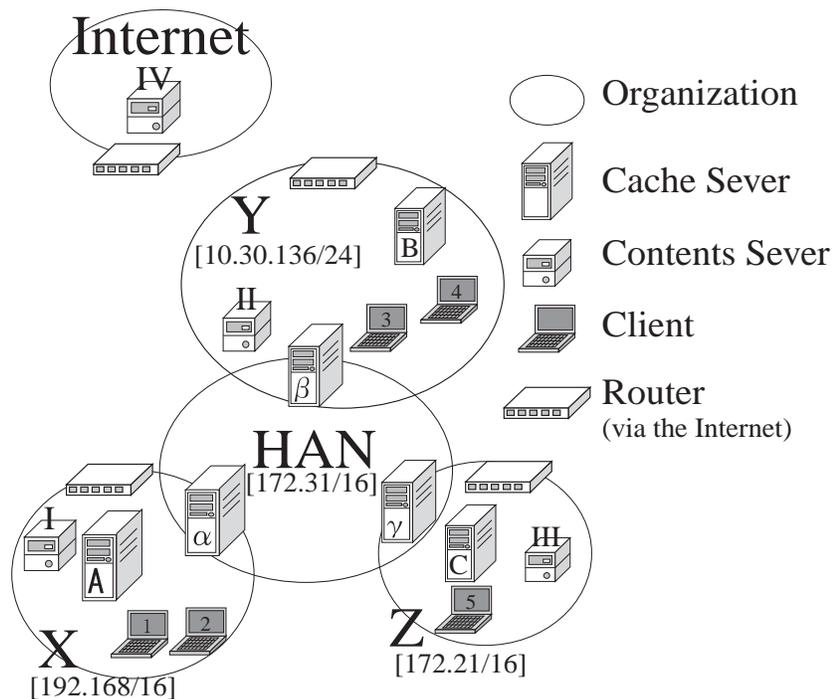


図 3.2 KPIX における WWW サーバ群の構成

- HAN 内で解決できない場合、自組織のインターネット経路により解決する。
- コンテンツの公開・非公開を、要求元の IP アドレスを用いて制御する。

### 3.2.4 キャッシュサーバ

効率的なコンテンツの交換を行うために、同一コンテンツの重複転送を抑制できるキャッシュサーバを用いる。複数のキャッシュサーバがコンテンツの交換・キャッシュを行い、さらにキャッシュサーバを互いに連携させることにより、PIX では HAN 内でのコンテンツの共有を実現した。

KPIX ではキャッシュサーバとして、Squid [矢吹 98]<sup>\*1</sup>を選択した。Squid は、キャッシュサーバとしての以下の機能を備えており、またフリーソフトであり入手しやすいからである。

<sup>\*1</sup> <http://www.squid-cache.org>

- HTTP コンテンツを交換し、交換したコンテンツをキャッシュする。
- クライアントからのリクエストを受け、それに対して応答を返す。
- キャッシュがヒットする場合は、それを応答に用いる。
- 他のサーバにリクエストをリレーする。
- 他のサーバからのリクエストを受け、応答を返す。

また、Squid では、他サーバとの連携の際にキャッシュがヒットしない場合に、Parent と Sibling という動作を選択できる。Parent はリクエストをそのまま他サーバにリレーして、コンテンツを取得してもらい、Sibling は、他サーバのキャッシュを利用し、キャッシュがない場合は自サーバでコンテンツを取得する、という動作を行う。したがって、リクエストに応じて Parent, Sibling を使い分けることにより、効率的なサーバ間の連携を実現可能である。

Squid はこの他に、リクエスト先のホスト名によりサーバの動作を変更できる、リクエスト元によるアクセス制御を行える、といった機能も備えている。

### 3.2.5 サーバ群の連携パターン

ある組織のコンテンツを別の組織のクライアントが閲覧する場合、12 通りの連携パターンが存在する (図 3.3)。

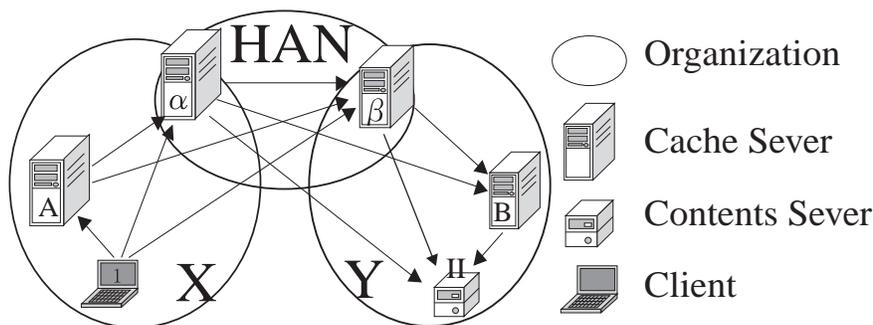


図 3.3 サーバ群の全連携パターン

このトポロジーと、KPIX における制約により、不可能な連携パターンがある。

まず、経路制御ポリシーにより、組織 X とサーバ  $\beta$ 、組織 Y とサーバ  $\alpha$  とは IP Reachable でないため、クライアント 1 とサーバ A からのサーバ  $\beta$  へのリンクと、サーバ  $\alpha$  からサーバ B、サーバ II へのリンクは不可能である。よって、制御ポリシーを満たす連携パターンは図 3.4 である。

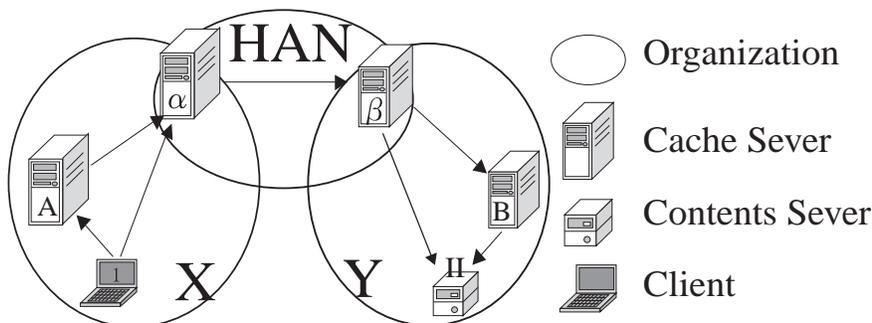


図 3.4 制御ポリシーを満たす連携パターン

また、WWW サービスへの要求により、サーバ B を用いると要求元の IP アドレスによるアクセス制御ができないため、要求を満たす連携パターンは図 3.5 である。

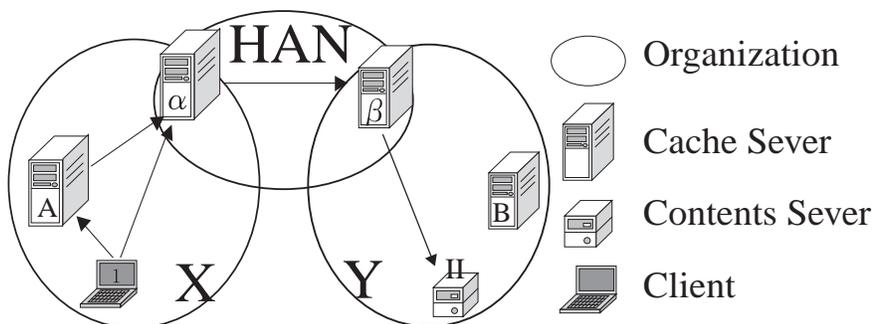


図 3.5 WWW サービスへの要求を満たす連携パターン

図 3.5 の連携パターンは、ネットワークの運用面においてそれぞれの特徴がある。

- クライアント 1 からサーバ A へのリンク

KPIX の導入とネットワーク運用体制とが連携できる場合や、ISP などのようにユーザ数が多くクライアントの管理が一括されていない場合においては、サーバ A を経由

する方が低コストである。これはサーバ A の設定変更のみで済み、クライアントでは一切の変更が不要であることによる。

- クライアント 1 からサーバ  $\alpha$  へのリンク

上述の連携ができない場合や、ユーザ数が少ない場合は、サーバ A を経由しない方が低コストである。この場合クライアントでの設定変更が必要である。これはクライアント設定のような共通ファイルを準備することなどでコストの低減が可能である。

以上のことより、KPIX においてはネットワーク体制との連携ができ、また参加組織が ISP や学校といったユーザが多い組織であることから図 3.6 のパターンが最善である。

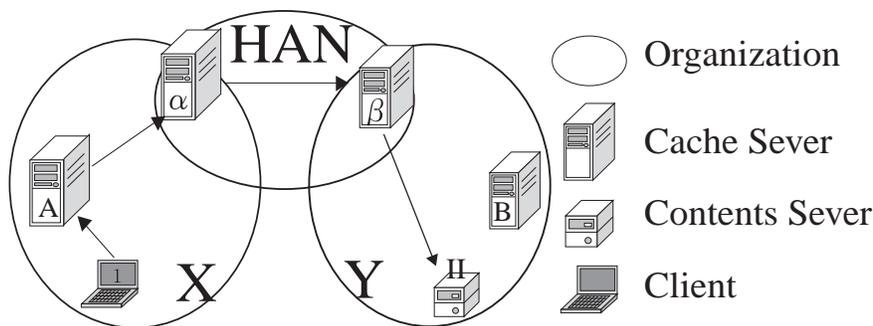


図 3.6 最善なサーバ群の連携パターン

### 3.3 参加組織コンテンツの共有

本節では、参加組織コンテンツを KPIX で共有するサーバ間の連携について述べる。

#### 3.3.1 リクエストに対するサーバ間の連携

ここでは、あるリクエストに対してのサーバ間の連携について具体的な動作を示す。

初期状態としてコンテンツはどのサーバにもキャッシュされていないものとする。図中の矢印はリクエストの流れを、矢印に付した数字は流れの順序を表す。

- 組織 X におけるサーバ I に公開コンテンツ  $i-1$  があり、組織 Y 内にいるクライアント 3

が閲覧しようとした場合（図 3.7）。

クライアント 3 はサーバ B に  $i-1$  をリクエストする ( $\vec{1}$ )。サーバ B は  $i-1$  がキャッシュされていないのでサーバ  $\beta$  に  $i-1$  をリクエストする ( $\vec{2}$ )。サーバ  $\alpha, \beta$  は、リクエストを  $\vec{3}, \vec{4}$  とリレーする。サーバ I はサーバ  $\alpha$  に  $i-1$  を返す ( $\vec{5}$ )。サーバ  $\alpha$  は  $i-1$  をキャッシュし、サーバ  $\beta$  に返す ( $\vec{6}$ )。サーバ  $\beta, B$  はコンテンツを  $\vec{7}, \vec{8}$  とリレーすると同時にキャッシュする。最後にクライアント 3 は  $i-1$  を閲覧できる。

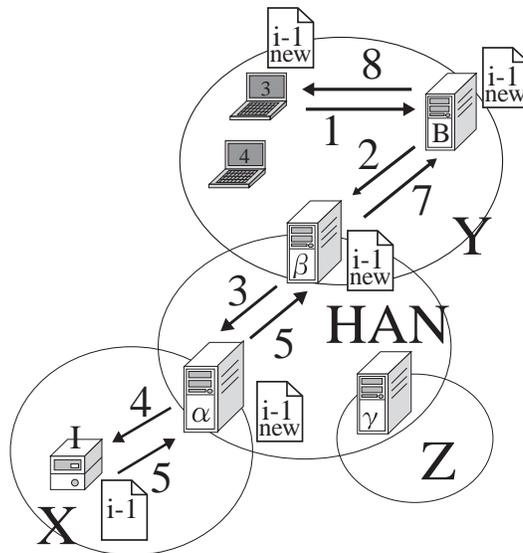


図 3.7 PIX 上の公開コンテンツに対するリクエスト

- その後、組織 Y 内にあるクライアント 4 が公開コンテンツ  $i-1$  を閲覧しようとした場合（図 3.8）。

クライアント 4 はサーバ B に  $i-1$  をリクエストする ( $\vec{1}$ )。サーバ B は  $i-1$  をキャッシュしているため、サーバ  $\beta$  にリクエストする事なくクライアント 4 コンテンツを受け取る ( $\vec{2}$ )。よってクライアント 4 は  $i-1$  を閲覧できる。

- その後、組織 Z 内にあるクライアント 5 が公開コンテンツ  $i-1$  を閲覧しようとした場合（図 3.9）。

クライアント 5 はサーバ C に  $i-1$  をリクエストする ( $\vec{1}$ )。サーバ  $\gamma, \beta$  は、リクエストを  $\vec{2}, \vec{3}$  とリレーする。サーバ  $\alpha$  は  $i-1$  をキャッシュしているため、サーバ I にリクエ

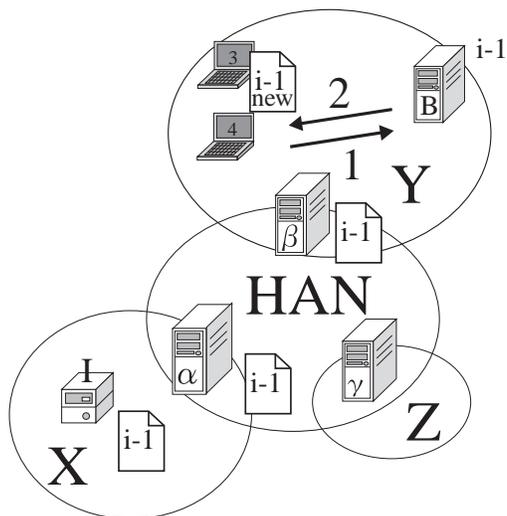


図 3.8 キャッシュされた公開コンテンツに対する同組織内からのリクエスト

トする事なくサーバ  $\beta$  に返す ( $\vec{4}$ )。サーバ  $\gamma$ , C はコンテンツを  $\vec{5}$ ,  $\vec{6}$  とリレーすると同時にキャッシュする。最後にクライアント 5 は  $i-1$  を閲覧できる。

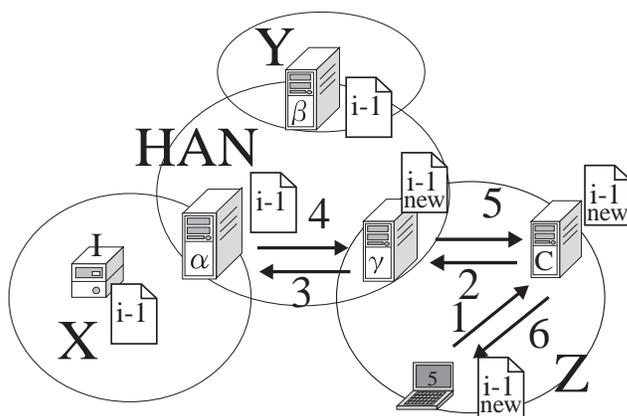


図 3.9 キャッシュされた公開コンテンツに対する他組織内からのリクエスト

- 組織 X におけるサーバ I に非公開コンテンツ  $i-2$  があり、組織 Z 内にいるクライアント 5 が閲覧しようとした場合 (図 3.10)

クライアント 5 からの  $i-2$  へのリクエストは、公開コンテンツに対するリクエストと同様にサーバ C、 $\gamma$ ,  $\alpha$ , I は、リクエストを  $\vec{1}$ ,  $\vec{2}$ ,  $\vec{3}$ ,  $\vec{4}$  とリレーする。サーバ I は、リクエスト元が組織 X 内の IP アドレスを持たないサーバ  $\alpha$  であるため、否定応答を返す

(5) 否定応答は、サーバ  $\alpha$ ,  $\gamma$ , C が  $\vec{6}$ ,  $\vec{7}$ ,  $\vec{8}$  とリレーし、クライアント 5 は i-2 を閲覧できない。

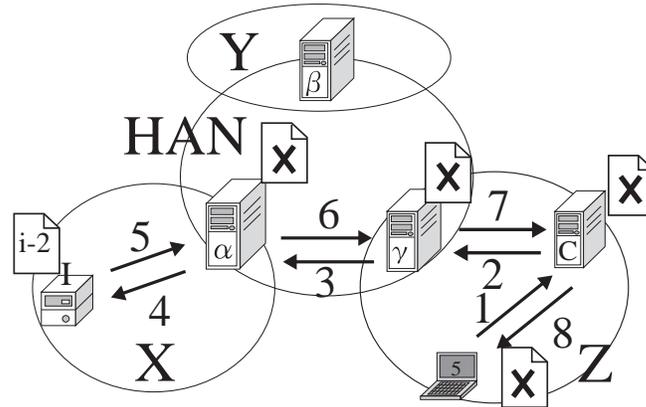


図 3.10 非公開コンテンツに対するリクエスト

- インターネット上にあるサーバ IV に公開コンテンツ i-3 があり、組織 X 内にいるクライアント 2 が閲覧しようとした場合 (図 3.11)

クライアント 2 はサーバ A に i-3 をリクエストする ( $\vec{1}$ )。サーバ A は、サーバ IV にリクエスト  $\vec{2}$  し、コンテンツを受け取る ( $\vec{3}$ )。受け取ったコンテンツをサーバ A はキャッシュして、クライアント 2 に返す ( $\vec{4}$ )。最後にクライアント 2 は閲覧できる。

### 3.3.2 各サーバの動作

ここでは、図 3.6 の連携パターンにおいて、第 3.3.1 節の動作を行うための各サーバごとの動作を示す。表 3.1 と表 3.2 をそれぞれサーバ A とサーバ  $\alpha$  の動作をリクエスト元を行とし、コンテンツのある場所を列として表す。

組織 Y, Z のサーバも同様の動作である。

### 3.3.3 Squid によるサーバ群の実現

表 3.1 と表 3.2 とに示すサーバの動作を Squid を例にして説明する。

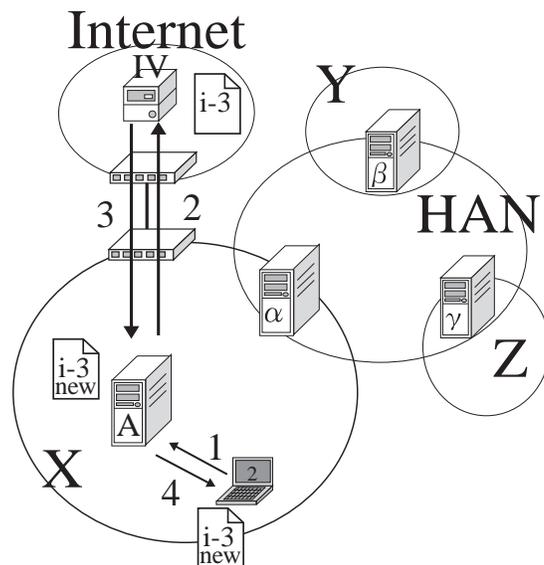


図 3.11 インターネット上の公開コンテンツに対するリクエスト

サーバ A と、サーバ  $\alpha$  をそれぞれ Squid A、Squid  $\alpha$  とし、squid.conf の設定例を表 3.3 と表 3.4 に示す。

ここでは、組織 X に IP アドレス 192.168/16 を割り当て、Squid A を 192.168.8.10 とす

表 3.1 組織 X 内キャッシュサーバ A の動作

リクエスト元	コンテンツのある場所		
	組織 X 内	組織 Y、Z 内	インターネット
X のクライアント	直接 *1	$\alpha$ にリクエスト *1	直接 *1
サーバ $\alpha$ *2	拒否 *2	拒否 *2	拒否 *2
サーバ $\beta, \gamma$ *3	拒否 *3	拒否 *3	拒否 *3
インターネット *4	拒否 *4	拒否 *4	拒否 *4

\*1 キャッシュされていればそれを用いてコンテンツを返す。\*2  $\alpha$  の設定が正しければありえない。対設定ミス。\*3  $\beta, \gamma$  の設定が正しければありえない。対設定ミス。\*4 組織 X のインターネット側ファイアウォールで止めるのが普通である。

表 3.2 組織 X に対応する HAN 内キャッシュサーバ  $\alpha$  の動作

リクエスト元	コンテンツのある場所		
	組織 X 内	組織 Y、Z 内	インターネット
X のクライアント *1	拒否 *1	拒否 *1	拒否 *1
サーバ A	拒否 *2	Y なら $\beta$ にリクエスト *3 Z なら $\gamma$ にリクエスト *3	拒否 *2
サーバ $\beta, \gamma$	直接 *3	否定応答 *3	拒否 *4
インターネット *5	拒否 *5	拒否 *5	拒否 *5

\*1 クライアントの設定が正しければありえない。\*2 A の設定が正しければありえない。対設定ミス。\*3 キャッシュされていればそれを用いてコンテンツを返す。\*4  $\beta, \gamma$  の設定が正しければありえない。対設定ミス。\*5 HAN がプライベートアドレスを用いるならありえない。グローバルアドレスであっても、A のインターネット側ファイアウォールで止めるのが普通である。

る。同様に HAN に 172.31/16 を割り当て、Squid  $\alpha$  を 172.31.8.3 とする。

#### [Squid A の動作]\*2

- 組織 X 内からのリクエストを許可し、それ以外からのリクエストは拒否する (13, 14)。
- KPIX 参加組織コンテンツへのリクエストについては、必ず squid $\alpha$  に Parent でリクエストする (4, 5, 10)。
- 上記以外へのリクエストは、直接リクエストする (11)。

#### [Squid $\alpha$ の動作]\*3

- 組織 X 内にあるサーバ I への HAN 内からのリクエストと、SquidA からのリクエスト

\*2 動作の文末の数値は、表 3.3 の行番号に対応する。

\*3 動作の文末の数値は、表 3.4 の行番号に対応する。

表 3.3 Squid A の設定

```

1  visible_hostname A.X.or.jp
2  http_port 3128
3  icp_port 3130
4  cache_peer  $\alpha$ .X.KPIX.net parent 3128 3130
5  cache_peer_domain  $\alpha$ .X.KPIX.net .Y.or.jp .Z.or.jp
6  acl all src 0.0.0.0/0
7  acl X_SRC src 192.168.0.0/16
8  acl PIX_DST dstdomain .Y.or.jp .Z.or.jp
9  acl X_DST dstdomain .X.or.jp
10 never_direct allow PIX_DST
11 always_direct allow !PIX_DST
12 icp_access allow all
13 http_access allow X_SRC
14 http_access deny all

```

を許可し、それ以外からのリクエストは拒否する (16, 17, 18)。

- 組織 X 内にあるサーバ I へのリクエストは、直接リクエストする (13)。
- KPIX 参加組織のコンテンツへのリクエストは、それぞれ対応する HAN 内 Squid  $\beta$ ,  $\gamma$  に Parent でリクエストする (4, 5, 6, 7)。
- 上記以外のコンテンツへのリクエストは、許可しない (16, 17, 18)。

### 3.3.4 考察

参加組織のコンテンツの共有は、リクエスト先の組織に対応した HAN 内のサーバに、リクエストを Parent でリレーすることにより実現した。

表 3.4 PIX Squid  $\alpha$  の設定

```

1  visible_hostname  $\alpha$ .X.PIX.net
2  http_port 3128
3  icp_port 3130
4  cache_peer  $\beta$ .Y.KPIX.net parent 3128 3130
5  cache_Peer_domain  $\beta$ .Y.KPIX.net .Y.or.jp
6  cache_peer  $\gamma$ .Z.KPIX.net parent 3128 3130
7  cache_peer_domain  $\gamma$ .Z.KPIX.net .Z.or.jp
8  acl all src 0.0.0.0/0
9  acl PIX_SRC src 172.31.0.0/16
10 acl X_A src 192.168.8.10/32
11 acl X_DST dstdomain .X.or.jp
12 acl PIX_DST dstdomain .Y.or.jp .Z.or.jp
13 always_direct allow X_DST
14 never_direct allow all
15 icp_access allow all
16 http_access allow PIX_SRC X_DST
17 http_access allow X_A PIX_DST
18 http_access deny all

```

また、WWW サービスへの要求により組織内のサーバと HAN 内のサーバが互いに連携するのではなく、自組織のクライアントからの参加組織のコンテンツへのリクエストの場合だけ連携を行う。この動作は表 3.1 と表 3.2 を参照するとよく分かる。組織内サーバ A、HAN 内サーバ  $\alpha$  に以下の特徴があることが分かった。サーバ A が処理するリクエストは、すべて自組織 X のクライアントからのリクエストであり、すべてのコンテンツをキャッシュしている。サーバ  $\alpha$  が処理するリクエストは、HAN 内の他のサーバまたはサーバ A からだ

けであり、キャッシュするコンテンツは参加組織のコンテンツだけである。

したがって、たとえ他の組織に対応する HAN 内のサーバの設定が間違っている場合、自組織のサーバさえ正しく設定しておくことにより、他組織のクライアントからのリクエストの解決に自組織のインターネット経路を利用されることや、非公開コンテンツが公開されるといった問題は起こらない。

## 3.4 参加組織以外のコンテンツの共有

本節では、第 3.3 節の連携を行い、なおかつ参加組織以外のコンテンツを共有するサーバ間の連携について述べる。

### 3.4.1 リクエストに対するサーバ間の連携

ここでは、参加組織以外のコンテンツへのリクエストとに対してのサーバ間の連携について具体的な動作を示す。

初期状態としてコンテンツはどのサーバにもキャッシュされていないものとする。図中の矢印はリクエストの流れを、矢印に付した数字は流れの順序を表す。

- インターネット上にあるサーバ IV に公開コンテンツ  $i-3$  があり、組織 X 内にいるクライアント 2 が閲覧しようとした場合 (図 3.12)。

クライアント 2 はサーバ A に  $i-3$  をリクエストする ( $\vec{1}$ )。サーバ A は、サーバ  $\alpha$  にリクエスト  $\vec{2}$  を出す。サーバ  $\alpha$  は他の HAN 内キャッシュサーバ  $\beta, \gamma$  にキャッシュされているか問い合わせる ( $\vec{3}, \vec{3'}$ )。各キャッシュサーバにはキャッシュされていないので、サーバ  $\alpha$  に否定応答を返す ( $\vec{4}, \vec{4'}$ )。サーバ  $\alpha$  は問い合わせたサーバすべてから否定応答をもらった後、サーバ A に  $i-3$  のリクエスト  $\vec{5}$  を出す。サーバ A はサーバ IV にリクエスト  $\vec{6}$  を出す。以降、 $i-3$  は  $\vec{7}, \vec{8}, \vec{9}, \vec{10}$  と返り、クライアント 2 は  $i-3$  を閲覧できる。3.13 では、リクエストをできるだけ HAN 内で解決するために、サーバ  $\alpha$  は、サーバ A からのリクエストを HAN 内のサーバ  $\beta, \gamma$  に問い合わせている。また、HAN 内で

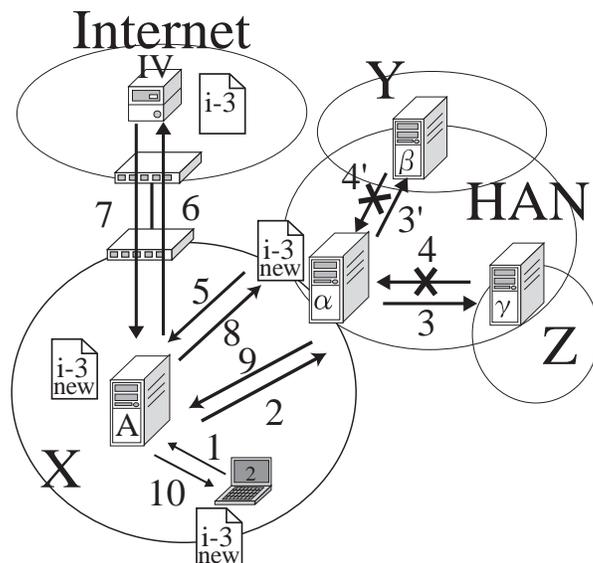


図 3.12 インターネット上の公開コンテンツに対するリクエスト

解決できない場合、自組織におけるインターネット経路で解決する。さらにこのとき、HAN 内でのキャッシュコンテンツを増やすため、サーバ A がサーバ  $\alpha$  にコンテンツをキャッシュさせている。

- その後、組織 Y 内にいるクライアント 4 が公開コンテンツ i-3 を閲覧しようとした場合 (図 3.13)。

クライアント 4 はサーバ B に i-3 をリクエストする ( $\vec{1}$ )。サーバ B は、サーバ  $\beta$  にリクエスト  $\vec{2}$  を出す。サーバ  $\beta$  は他の HAN 内キャッシュサーバ  $\alpha$ ,  $\gamma$  にキャッシュされているか問い合わせる ( $\vec{3}$ ,  $\vec{3'}$ )。サーバ  $\gamma$  は i-3 キャッシュしていないので、サーバ  $\beta$  に否定応答を返す ( $\vec{4'}$ )。サーバ  $\alpha$  は i-3 をキャッシュしているので、キャッシュをサーバ  $\beta$  に返す ( $\vec{4}$ )。サーバ  $\beta$  は、受け取ったキャッシュコンテンツをキャッシュし、クライアント 4 に返す ( $\vec{2}$ )。よってクライアント 4 は i-3 を閲覧できる。

### 3.4.2 各サーバの動作

ここでは、参加組織のコンテンツを共有する 3.3.1 の動作を行い、なおかつ、第 3.4.1 節の動作を行うための各サーバごとの動作を示す。表 3.5 と表 3.6 とは、それぞれサーバ A と

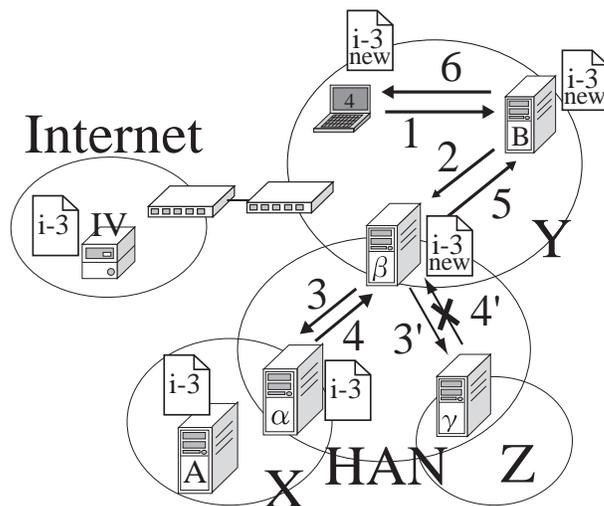


図 3.13 キャッシュされた公開コンテンツに対するリクエスト

サーバ  $\alpha$  との動作を示す。行にリクエスト元を、列にコンテンツのある場所をとっている。  
Y と Z のサーバも同様である。

表 3.5 組織 X 内キャッシュサーバ A の動作

リクエスト元	コンテンツのある場所		
	組織 X 内	組織 Y、Z 内	インターネット
X のクライアント	直接 *1	$\alpha$ にリクエスト *1	$\alpha$ にリクエスト *1
サーバ $\alpha$	直接 *1	拒否	直接 *1
サーバ $\beta, \gamma$ *2	拒否 *2	拒否 *2	拒否 *2
インターネット *3	拒否 *3	拒否 *3	拒否 *3

\*1 キャッシュされていればそれを用いてコンテンツを返す。

\*2 PIX のルーティングの設計で止める事も可能である。

\*3 組織 X のインターネット側ファイアウォールで止めるのが普通である。

表 3.6 組織 X に対応する HAN 内キャッシュサーバ  $\alpha$  の動作

リクエスト元	コンテンツのある場所		
	組織 X 内	組織 Y、Z 内	インターネット
X のクライアント *1	拒否 *1	拒否 *1	拒否 *1
サーバ A	拒否 *2	Y なら $\beta$ にリクエスト *3	$\beta, \gamma$ にリクエスト *3
サーバ $\beta, \gamma$	直接	Z なら $\gamma$ にリクエスト *3 否定応答 *3	無ければ A にリクエスト 否定応答 *3
インターネット *4	拒否 *4	拒否 *4	拒否 *4

\*1 クライアントの設定が正しければありえない。

\*2 A の設定が正しければありえない。対設定ミス。

\*3 キャッシュされていればそれを用いてコンテンツを返す。

\*4 HAN がプライベートアドレスを用いるならありえない。グローバルアドレスであっても、A のインターネット側ファイアウォールで止めるのが普通である。

### 3.4.3 Squid によるサーバ群の実装

表 3.5 と表 3.6 とに示すサーバの動作を Squid を例にして説明する。

サーバ A と、サーバ  $\alpha$  をそれぞれ Squid A、Squid  $\alpha$  とし、squid.conf の設定例を表 3.7 と表 3.8 に示す。

ここでは、組織 X に IP アドレス 192.168/16 を割り当て、Squid A を 192.168.8.10 とする。同様に HAN に 172.31/16 を割り当て、Squid  $\alpha$  を 172.31.8.3 とする。

[Squid A の動作]\*4

- 組織 X 内からと Squid  $\alpha$  からのリクエストを許可し、それ以外からのリクエストは拒否する (13, 14, 15)。

\*4 動作の文末の数値は、表 3.7 の行番号に対応する。

- 組織 X 内のコンテンツサーバ I へのリクエストは、直接リクエストする (9)。
- 上記以外へのリクエストについては、必ず Squid  $\alpha$  に Parent でリクエストする (4, 11)。
- Squid  $\alpha$  からのリクエストは、直接リクエストする (10)。

表 3.7 Squid A の設定

```

1  visible_hostname A.X.or.jp
2  http_port 3128
3  icp_port 3130
4  cache_peer  $\alpha$ .X.KPIX.net parent 3128 3130
5  acl all src 0.0.0.0/0
6  acl X_SRC src 192.168.0.0/16
7  acl PIX_ $\alpha$  src 172.31.8.3/32
8  acl X_DST dstdomain .X.or.jp
9  always_direct allow X_DST
10 always_direct allow PIX_ $\alpha$ 
11 never_direct allow all
12 icp_access allow all
13 http_access allow X_SRC
14 http_access allow PIX_ $\alpha$ 
15 http_access deny all

```

[Squid  $\alpha$  の動作]\*<sup>5</sup>

- HAN 内からのリクエストと、Squid A からのリクエストを許可し、それ以外からのリクエストは拒否する (16, 17, 18)。

---

\*<sup>5</sup> 動作の文末の数値は、表 3.8 の行番号に対応する。

- 組織 X 内にあるサーバ I へのリクエストは、直接リクエストする (13)。
- 上記以外のすべてのリクエストは他のキャッシュ・プロキシを使ってリクエストする (14)。
- KPIX 参加組織のコンテンツへのリクエストは、それぞれ対応する HAN 内 Squid  $\beta$ ,  $\gamma$  に Parent でリクエストする (5, 7)。
- 上記以外のコンテンツへのリクエストは、HAN 内 Squid  $\beta$ ,  $\gamma$  に Sibling でリクエストする (4, 6, 16)。
- Sibling でキャッシュの無かったリクエストについては、Squid A に Parent でリクエストする (8)。

#### 3.4.4 考察

参加組織以外のコンテンツを共有するサーバ間の連携は、KPIX において実装を始め、現在動作の検証を行っている。

参加組織以外のコンテンツの共有は、組織内のサーバと自組織に対応する HAN 内のサーバを複雑に連携させることにより実現している。これは、HAN 内のキャッシュコンテンツを増やし、HAN 内でのリクエストの解決の割合を増やすために行っている。HAN 内でのコンテンツの共有を行うために、HAN 内のサーバにリレーされた参加組織以外のリクエストを他のサーバに Sibling で問い合わせている。この際に、Sibling によるキャッシュのヒットを行うために、HAN 内のサーバからのリクエストをすべて許可して。しかし、そのために HAN 内の他のサーバから Parent でリクエストされたときに、自組織内のサーバに Parent でリクエストをリレーしてしまう。自組織内のサーバは、自組織対応する HAN 内からのリクエストを無条件にインターネット経路を利用して解決しようとするので、自組織以外のクライアントからのリクエストの解決に自組織のインターネット経路を利用されてしまう問題がある。この問題は、参加組織の各サーバが正しく設定されていない場合に起こり得る。現在のところ対処法は、Squid の残すアクセスログを監視し、不正なアクセスに対

表 3.8 PIX Squid  $\alpha$  の設定

```
1 visible_hostname  $\alpha$ .X.PIX.net
2 http_port 3128
3 icp_port 3130
4 cache_peer  $\beta$ .Y.KPIX.net sibling 3128 3130
5 neighbor_type_domain  $\beta$ .Y.KPIX.net parent .Y.or.jp
6 cache_peer  $\gamma$ .Z.KPIX.net sibling 3128 3130
7 neighbor_type_domain  $\gamma$ .Z.KPIX.net parent .Z.or.jp
8 cache_peer  $\alpha$ .X.or.jp parent 3128 3130
9 acl all src 0.0.0.0/0
10 acl PIX_SRC src 172.31.0.0/16
11 acl X_A src 192.168.8.10/32
12 acl X_DST dstdomain .X.or.jp
13 always_direct allow X_DST
14 never_direct allow all
15 icp_access allow all
16 http_access allow PIX_SRC
17 http_access allow X_A
18 http_access deny all
```

してアクセス拒否の設定をする以外に方法はない。

## 3.5 クライアントの設定

本節では、各クライアントが KPIX を利用するための設定について述べる。

組織として PIX を構築する場合、または、既に構築されている PIX に加入及び脱退する

場合において、クライアントが PIX を利用するための設定が必要である。必要な設定方法として、自組織のキャッシュサーバの設定を変更させる方法と、各クライアントの設定を変更する方法とがある。

- 自組織のキャッシュサーバの設定変更

KPIX の導入とネットワーク運用体制とが連携できる場合や、ISP のようにユーザ数が多く、クライアントの設定がまちまちである場合、自組織のキャッシュサーバを経由する方が低コストである。これは、サーバの変更のみで済み、クライアントでは一切の変更が不要である。

- クライアントの設定変更

KPIX に導入とネットワーク運用体制とが連携できない場合や、ユーザ数が少ない場合、ワークステーションのようにクライアントの管理が一括で行える場合は、クライアントの設定を変更する方が低コストである。

以降は、クライアントの設定として、WWW ブラウザでの設定について述べる。

#### 3.5.1 WWW ブラウザのプロキシ設定

クライアントの WWW ブラウザのプロキシ設定として、手動設定と自動設定がある。

- WWW ブラウザのプロキシ手動設定

大抵の WWW ブラウザではプロキシサーバを設定できる。しかし、プロキシサーバの変更ごとに設定のクライアントの設定変更を必要とする。

- WWW ブラウザのプロキシ自動設定

ブラウザのプロキシ設定を自動化する設定ファイルに PAC(Proxy Auto Configuration) ファイルがある。PAC ファイルは、JavaScript によって書かれたテキストファイルであり、現在は、Netscape Navigator<sup>\*6</sup>、Microsoft Internet Explorer<sup>\*7</sup>で利用で

---

<sup>\*6</sup> <http://home.netscape.com/eng/mozilla/2.0/relnotes/demo/proxy-live.html>

<sup>\*7</sup> <http://microsoft.com/japan/ie/ieak/>

きる。

PAC ファイルによる設定を一度してしまえば、以降、プロキシの変更があっても、PAC ファイルを書き換えるだけで済み、クライアントの変更を一切必要としないため、作業コストを抑えることができる。また、ブラウザの動作についても、URL や、ホスト名、IP アドレスなどにより細かく設定できるため、柔軟なネットワーク設計を可能とする。PAC ファイルは、ブラウザのプロキシの設定で URL を指定することで有効となる。一度読み込ませれば、PAC ファイルが更新されても自動的に再読み込みを行うブラウザが一般的である。

また、WPAD(Web Proxy Auto-Discovery protocol) によって、PAC ファイルを自動検出を行う方法もあるが、現在では、Microsoft Internet Explorer だけでしか使用できない。

PAC ファイルを利用することにより、クライアントの設定コストの抑制ばかりでなく、PIX モデルのサーバ群の連携にクライアントを取り込むことができ、より柔軟なネットワークの設計を可能とする。

## 第 4 章

# まとめ

WWW コンテンツを HTTP で交換する PIX モデルにおいて、効率的なサーバ間の連携を行えるサーバ群の構成を提案した。

参加組織のコンテンツの共有では、各サーバが他組織のコンテンツをリクエストする以外に連携しないため、たとえ他の HAN 内のサーバの設定が間違っている場合、自組織のサーバさえ正しく設定しておけば、他組織のクライアントからのリクエストの解決に自組織のインターネット経路を利用されることや、非公開コンテンツが公開されるといった問題は起こらない。

参加組織以外のコンテンツの共有は、HAN 内のキャッシュコンテンツを増やし、できるだけ HAN 内でリクエストを解決するために、複雑に連携している。この連携パターンでは、参加組織の各サーバが正しく設定されていない場合に、自組織以外のクライアントからのリクエストの解決に自組織のインターネット経路を利用されてしまう問題がある。現在のところ対処法は、Squid の残すアクセスログを監視し、不正なアクセスに対してアクセス拒否の設定をする以外に方法はない。

参加組織以外のコンテンツの共有は、現在 KPIX において実装を始め、検証を行っている。しかし、現在の KPIX において KPIX を利用するユーザ数が少ないため、リクエスト数も少なく、キャッシュのヒット率も低い場合、検証データが少ない。また、サーバ間の連携が複雑なため、サーバの設定も難しく、思うように設定ができないためといった問題がある。PIX モデルへの移行を促すためには、設定ツールは不可欠である。

地域コミュニティの活性化にとともに、さまざまなコミュニティに対応できる柔軟なネットワークが不可欠である。今回は、PIX モデルの一つの連携パターンでしか、サーバ間の連

携効率よく行うサーバ群の構成を提案できなかった。しかし、選択できなかった連携パターンにおいての最適なサーバ群の構成の検証も必要である。

また、今後、インターネットトラフィックの主流となるであろうストリーム型コンテンツを交換・共有する手法の検討も行っていきたい。

# 謝辞

本研究におこなうにあたり、高知大の菊池時夫助教授には、サーバ間の連携から Squid の設定方法までご指導いただきました。また、高知工業高等専門学校は今井一雅助教授、高知工業技術センターの武市統さん、今西孝也さん、富士通高知システムエンジニアリングの濱崎哲一さん、平田圭一さん、高知システムズの松本浩明さん、山本幸生さん、シティネットの澤本一哲さん、高知県情報企画課の大庭 孝之さん、西岡輝幸さんほか KPIX 実験研究協会のみなさまにはサーバの立ち上げから Squid の設定までいろいろとご協力いただきました。

また、同研究室においては角谷俊介さん、田淵理恵さん、戸梶 桃さん、舟橋 稔仁さん、長尾 美保さんに、KPIX を利用してもらいデータ収集に協力してもらいました。同じテーマで研究を行った杉山道子さん、西内一馬さん、正岡元さんには、本学でのサーバの運営や検証の際にさまざまなご協力をいただきました。

指導教員の菊池豊助教授には、本研究活動のすべてにおいて御指導、御協力いただきました。ここに感謝の意を表します。

ほか本研究に関わったすべての人に感謝の意を表します。

また、本研究は、通信・放送機構の平成 11 年度地域提案型研究開発制度（研究開発名「インターネットにおける地域指向型トラフィック交換モデル」と、科学研究費補助金（課題番号 11450153）との助成を受けています。

# 付録 A

## KPIX の歴史

この付録では KPIX の歴史を時系列にそって示す。

- 【1996 年未明】地域 IX の誕生

IX を地方に構築しようとする活動が活発におこなわれ始める。これは、地域内でのトラフィックが当該地域外の IX を経由することを、インターネットを基盤とした地域情報化を推進する場合の障害になると考えるようになってきたためである。

- 【1997 年】PIX モデル誕生

応用層によるトラフィック交換モデルが日本ソフトウェア科学会において発表される [菊池 97]。

- 【1997 年】DNS の構成

DNS の構成が話し合わせ、おおまかな DNS の構成方法が決定する。

- 【1998/07/13】第 1 回 KPIX 会議

県庁情報企画課（電気ビル内）において会議を行う。本会において KPIX 実験研究協議会が発足した。

- 【1998/07/13】KPIX の誕生

PIX モデルの実証実験の場として KPIX が組織される。KPIX の構築・運営を通して、PIX モデルの構築・運用ノウハウの確立を目指す。

- 【1999/08/02】第 2 回 KPIX 会議

KOCHI 2001 PLAN 会議室（森連会館 6 階）において会議を行う。本会では、KPIX の実験計画について討議された。

- 【1999/12/02】第3回 KPIX 会議
 

高知大 菊地研究室において会議を行う。本会では、KPIX のルーティングについて討議した。
- 【1999/12/02】KPIX の経路制御
 

第3回 KPIX 会議において、以下の事項を確認した。

  - KPIX では HAN ボードを置かない
  - 組織内 LAN と HAN との両方にキャッシュサーバをおく
  - ルーティングは静的に行う
- 【2000/02/10】第4回 KPIX 会議
 

高知大 会議室において会議を行う。本会では、Squid のインストール法、設定例、そして PIX モデルにおけるキャッシュサーバの連携の提案をし討議した。
- 【2000/02/10】Squid の動作の提案
 

PIX モデルにおけるキャッシュサーバの連携が 12 通りあることを示した。この連携は、設定変更にかかるコストについて考えられている。
- 【2000/】旧 Squid の設定考案
 

参加組織のコンテンツを HAN 内で共有する Squid の設定を考案し、示した。また、KPIX において実装し、検証を開始した。
- 【2000/04/17】KPIX 会議番外編
 

シティネット（大そね）において会議を行う。本会では、KPIX がデータリンクとして使用している IP ネットワークが IP reachable であってよいものを討議した。
- 【2000/06】NTP の設定
 

TS-820 を購入。NTP の設定を行う。
- 【2000/07/25】第5回 KPIX 会議
 

高知高専 今井研において会議を行う。本会では、KPIX の無線リンクと Squid の設置状況と今後について討議した。

- 【2000/08/10】 MRTG を用いたトラフィック測定  
MRTG を用いて、Squid サーバを通過する HTTP コンテンツの監視を始める。
- 【2000/09/29】 DSM 富山  
かねてから問題であった、KPIX のプライベートアドレス枯渇問題について解決案を示した。また、データリンクの相互運用についての解決案も同時に示した。
- 【2000/10/14-19】 富山国体中継  
富山国体の中継画像を JGN-菊池研究室と経由して、Windows Media Technology ストリーミングを KPIX へ配信した。
- 【2000/11/08】 ギガビットシンポジウム 2000 中継  
北九州市で開催されたギガビットシンポジウム 2000 を JGN-菊池研究室と経由して、Windows Media Technology ストリーミングを KPIX へ配信した。
- 【2000/11/20-22】 ITRC 香川  
PIX モデルにおいて効率的なサーバ間の連携の行えるサーバ群の構成を提案し、2000/12/1 に開催された DSM 和歌山にて発表をした。
- 【2000/12/01】 DSM 和歌山  
PIX モデルにおいて効率的なサーバ間の連携の行えるサーバ群の構成を提案し、2000/12/1 に開催された DSM 和歌山にて発表をした。
- 【2000/12/04】 第 6 回 KPIX 会議  
高知工科大 A260 高知工科大学において会議を行う。本会議では、無線リンクの状況、研究発表、次年度の研究計画について討議した。
- 【2000/12/04】 pac 作成  
プロキシ自動設定ファイル pac を作成する。
- 【2000/12/04】 Squid 動作決定  
効率的なサーバ間の連携が行える Squid の動作を決定した。
- 【2000/12/04】 新 Squid の設定考案  
決定した動作に基づき Squid の設定ファイルを示した。

- 【2000/12/04】 Squid 実装開始

新 Squid の設定ファイルを適用し、効率的な連携が行えるか検証を開始した。

- 【2001/01/09】 コストモデル

KPIX 協議会以外の組織が PIX モデルを構築する際の指針となるために、PIX モデルのランニングコストをモデル化し評価した。

- 【2001/01/15】 第 7 回 KPIX 会議

高知工科大 A260 高知工科大学において会議を行う。本会議では、次年度の KPIX 会議運営について討議した。

# 付録 B

## KPIX 活動経緯

本付録を KPIX 活動の一つの資料としてまとめる。そのため、本文と一部重複する部分がある。

### B.1 Squid 動作モデルの設計

我々は、PIX モデルにおける Squid の動作連携を提案した。これにより、PIX モデルでの Squid の連携は 12 通りあることがわかる [菊池 00b]。このモデルでは、PIX モデルに新たに参加する組織が現れた場合の PIX 運営組織・新規 PIX 参加組織の設定変更にかかるコストに重点を置き、順位付けをおこなった (表 B.1)。この Squid 動作連携では、12 通り中どの連携パターンを選択するとキャッシュのヒット率が高くなるかということは考えていない。これは、キャッシュのヒット率が高くなるかということは、コンテンツをアクセスするトラフィックに依存しているため、このモデルでは一般的な解を得ることができないためである。

一般的な PIX モデル構成における Squid 動作連携をすべて示したことで、各組織のユーザ数やネットワーク運用体制の状態によって、最も設定変更にかかるコストが低い Squid の連携を明示することができた。

現在、運用を行っている KPIX では、各組織の運用面でのポリシーにより、基本的な Squid の動作連携は、表 B.1 中の『11』を選択している。この図より、KPIX は現在、設定変更にかかるコストに関しては良い連携を選んでいないことがわかる。

### B.1.1 設定変更にかかるコスト

表 B.1 は PIX モデルでの Squid の連携 12 通りを表している。この表は、後のキャッシュサーバ連携モデルに挙げるモデルを図 B.1 を基に、実際にどのキャッシュサーバを通過させるかを表している。

図 B.1 のパラメータの説明を述べる。最も左の列に表示される数字は、キャッシュサーバ連携の通し番号である。“Z X” が示すものは、組織 Z のクライアントから組織 X へのコンテンツ要求の意味である。“X Z” は、組織 X のクライアントから組織 Z へのコンテンツ要求の意味である。

“新規”・“既存” は PIX モデルに新たに加わる組織・PIX モデルの運営をおこなっている既存の組織が必要な設定量を表す。

“新規”・“既存” は 3 つのパラメータを持つ。このパラメータは左から、PIX 参加組織内キャッシュサーバ、組織に対応する HAN 内キャッシュサーバ、クライアントに必要な設定量を表す。また、その設定量を表す記号として、“-” “x” がある。この記号はそれぞれ左から、設定不要、少量の設定変更が必要、設定ファイルの変更を行うとして 2 行程度の変更が必要、設定ファイルの変更を行うとして 5 行程度の変更が必要、多くの設定変更が必要を表す。それぞれの記号にそれぞれ に 1 点、 に 2 点、 に 3 点、 x に 4 点を付与する。

“合計” は各記号付与されている点の合計を書く。

“rank” は合計を基に各パターンを順位付けした結果である。

“hops” は要求をおこなうコンテンツサーバまでのホップ数を表す。

#### キャッシュサーバ連携モデル

ここでは、PIX モデルのキャッシュサーバ連携モデルを述べる。このモデルは、ネットワーク層の構造と運用面からの制約との条件から導出した。

1. コンテンツを持っている（アクセスされる）側のネットワーク構造は以下のサーバ類が IP reachable であると考える。

- (a) コンテンツを持つサーバ
  - (b) 組織内でキャッシュを行うサーバ
2. PIX バックボーンネットワーク構造は以下のサーバ類が IP reachable であると考え  
る。PIX バックボーンと各組織とは IP 交換可能である。しかし、PIX バックボーン  
は組織間の IP をトランジットしない。なお、ルータは専用であるものと想定している。
- (a) コンテンツを持つ側に設置するキャッシュサーバ
  - (b) コンテンツアクセスする側に設置するキャッシュサーバ
3. コンテンツをアクセスする側のネットワーク構造は以下のサーバ類が IP reachable で  
あると考える。
- (a) コンテンツにアクセスするクライアント
  - (b) 組織内でキャッシュを行うサーバ

このようなネットワークトポロジーモデル上でキャッシュサーバ連携モデルを考察した。

- コンテンツのあるサーバの組織のキャッシュサーバ 1b を用いるか否かの 2 通り。
- PIX バックボーンを通過する HTTP をどのキャッシュサーバ経由にするか。サーバ 2a  
かサーバ 2b か、あるいはその両方かの 3 通り。なお、組織間で IP をトランジットし  
ないので、どのキャッシュサーバも用いないという選択はあり得ない。
- コンテンツにアクセスするクライアント 3a からの HTTP を自組織内のキャッシュサー  
バ 3b 経由にするか否かの 2 通り。

このように、どのキャッシュサーバを連携し通過させるかだけで、12 通りの選択肢が存在  
する。

## B.2 参加組織のコンテンツを共有する Squid の設定

KPIX 参加組織による HAN の構築にともない、「KPIX 参加組織の WWW コンテンツ  
を KPIX を介して取得する」Squid の設定を考案し実装した。

	Z X		Z X		X Z		X Z		1 2 3 4 x	合計	rank	hops
	新規	既存	新規	既存	新規	既存	新規	既存				
1.	6	1 1	--	---	1 3 3	--	--	1 2 0 0	5	3	2	
2.	6	1 1 1	--	---	1 3 C 3	--	--	0 3 0 0	6	4	3	
3.	6	3 1	--	---	1 1 3	---	--	2 0 0 0	2	1	2	
4.	6	3 1 1	-	---	1 1 3 3	--	-	3 2 0 0	7	5	3	
5.	6	3 1 A 1	-	---	1 1 3 C 3	--	-	2 3 0 0	8	7	4	
6.	6	3 A 1	-	---	1 1 C 3	--	-	3 2 0 0	7	5	3	
7.	6	C 1 1	x--	---	1 A 3 3	--	--	1 0 1 1	8	7	3	
8.	6	C 1 A 1	x--	---	1 A 3 C 3	--	--	0 1 1 1	9	10	4	
9.	6	C 3 1	--	---	1 A 1 3	---	---	0 0 1 0	3	2	3	
10.	6	C 3 1 1	-	---	1 A 1 3 3	--	--	1 0 2 1	11	12	4	
11.	6	C 3 1 A 1	-	---	1 A 1 3 C 3	--	--	0 3 1 0	9	10	5	
12.	6	C 3 A 1	-	---	1 A 1 C 3	--	--	1 2 1 0	8	7	4	

表 B.1 KPIX トポロジーと設定量

### B.2.1 参加組織による HAN の構成

PIX では、以下の経路制御ポリシーを満たす HAN を構成する（図 3.1）。

- HAN 内は IP Reachable である。

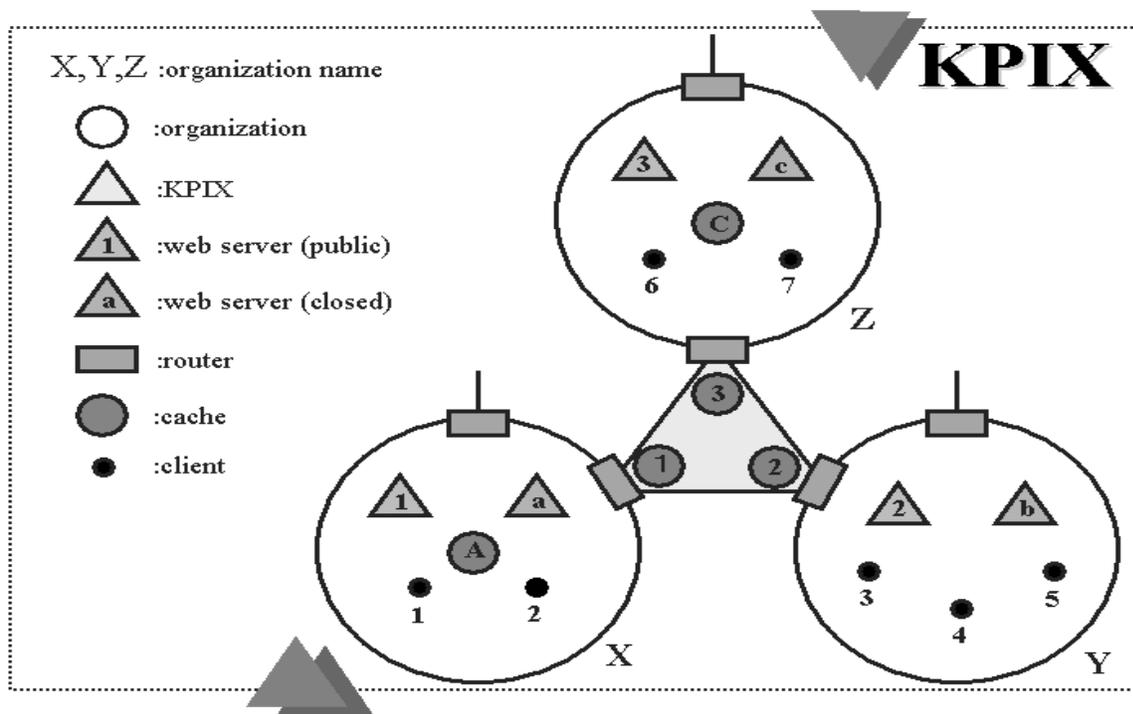


図 B.1 KPIX のトポロジー

- HAN と Internet は IP Reachable でない。
- 組織内のコンピュータと、組織に対応する HAN 内のサーバとは IP Reachable である。

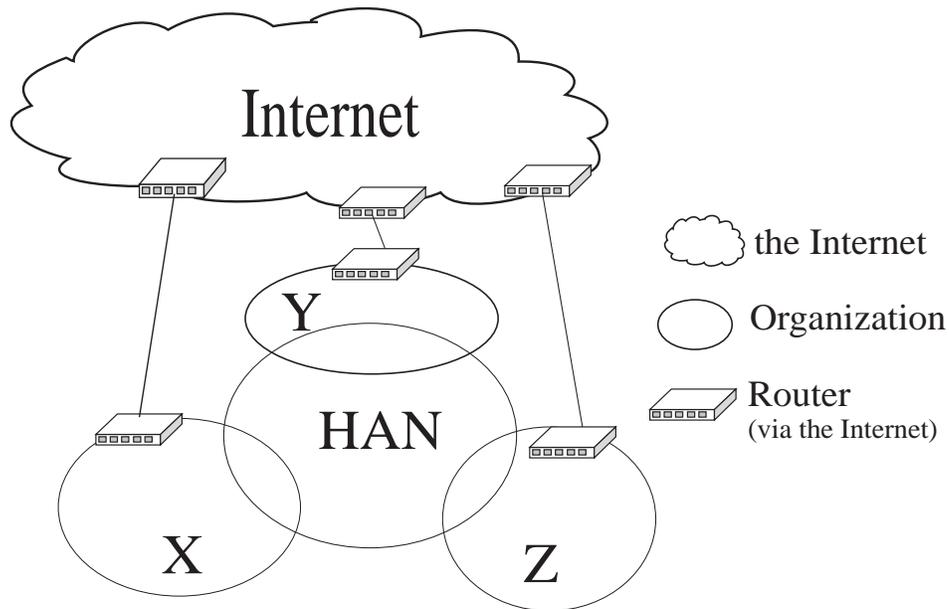


図 B.2 参加組織による HAN の構成

円は IP Reachable な領域を示す。HAN と各組織との重なりは互いに IP Reachable な HAN ボーダである。HAN は Internet と IP Reachable でないため、各組織のインターネット経路を経由しないと IP の交換は行えない。

### B.2.2 KPIX における典型的な WWW サーバ群の構成例

ここでは、以降の説明のために PIX モデルにおける WWW サーバ群の構成例を示す(図 B.3)。各組織は以下で構成される。

- コンテンツを提供するサーバ (図中 I, II, III)
- 組織内で運用するキャッシュサーバ (図中 A, B, C)
- HAN で各組織に対応するキャッシュサーバ (図中  $\alpha, \beta, \gamma$ )
- インターネット経路

- 各組織のクライアント (図中 1, 2, 3, 4, 5)

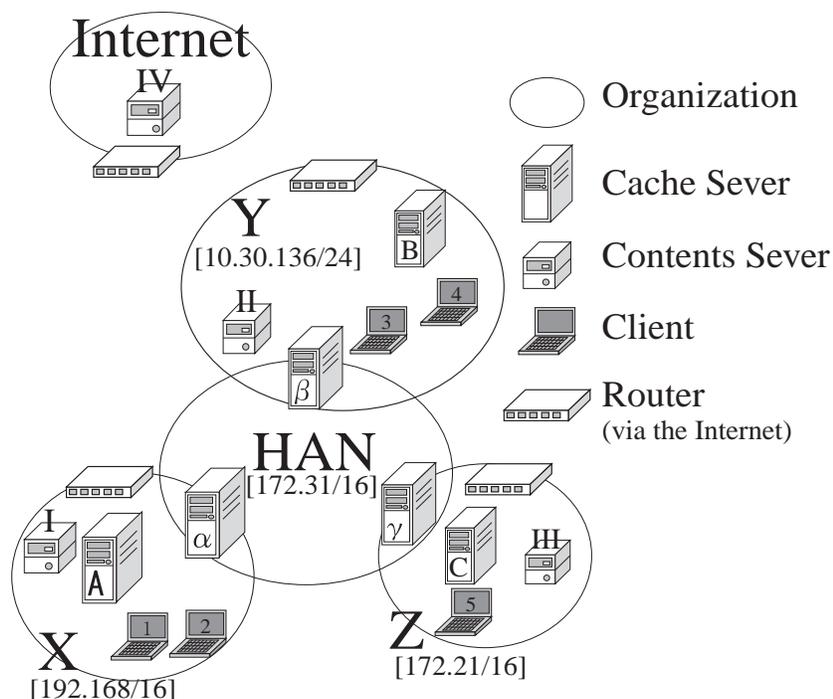


図 B.3 KPIX における WWW サーバ群の構成

### B.2.3 WWW サービスに対する要求

ここでは PIX 参加組織からの WWW サービスに対する要求を述べる。

- HAN 内でできるだけ解決する。
- HAN 内で解決できない場合、自組織のインターネット経路により解決する。
- コンテンツの公開・非公開を、要求元の IP アドレスを用いて制御する。

### B.2.4 サーバ群の連携パターン

ある組織のコンテンツを別の組織のクライアントが閲覧する場合、12 通りの連携パターンが存在する (図 B.4)。

このトポロジーをもとに KPIX の実装では、

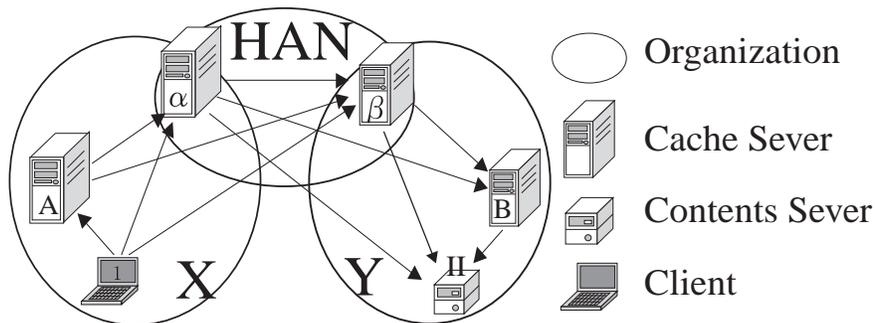


図 B.4 サーバ群の全連携パターン

- 経路制御ポリシーにより、組織 X とサーバ  $\beta$ 、組織 Y とサーバ  $\alpha$  とは IP Reachable でない。
- サーバ B を用いると要求元の IP アドレスによるアクセス制御ができない。

といった制約があり、これを満たすのは図 B.5 の連携パターンである。

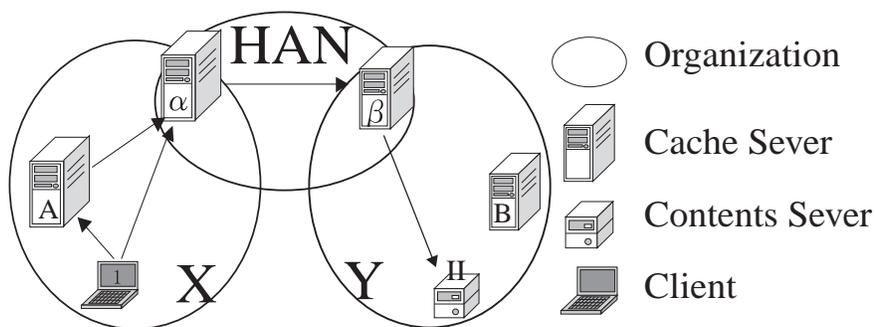


図 B.5 制約を満たす連携パターン

更に、クライアント数はサーバ数よりも圧倒的に多く、クライアントの設定を変更するには大きなコストをとらなう。また、参加組織において、既に運営されているキャッシュサーバがある場合、キャッシュの設定を変更する方が作業コストが小さい。

以上のことより、KPIX においては図 B.6 の連携パターンを選択した。

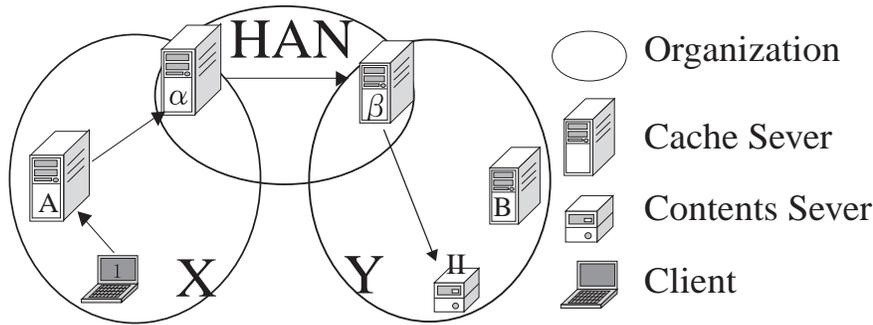


図 B.6 KPIX におけるサーバ群の連携パターン

### B.2.5 リクエストに対するサーバ間の連携

ここでは、あるリクエストに対してのサーバ間の連携について具体的な動作を示す。

初期状態としてコンテンツはどのサーバにもキャッシュされていないものとする。図中の矢印はリクエストの流れを、矢印に付した数字は流れの順序を表す。

- 組織 X におけるサーバ I に公開コンテンツ  $i-1$  があり、組織 Y 内にいるクライアント 3 が閲覧しようとした場合 (図 B.7)。

クライアント 3 はサーバ B に  $i-1$  をリクエストする ( $\vec{1}$ )。サーバ B は  $i-1$  がキャッシュされていないのでサーバ  $\beta$  に  $i-1$  をリクエストする ( $\vec{2}$ )。サーバ  $\alpha, \beta$  は、リクエストを  $\vec{3}, \vec{4}$  とリレーする。サーバ I はサーバ  $\alpha$  に  $i-1$  を返す ( $\vec{5}$ )。サーバ  $\alpha$  は  $i-1$  をキャッシュし、サーバ  $\beta$  に返す ( $\vec{6}$ )。サーバ  $\beta, B$  はコンテンツを  $\vec{7}, \vec{8}$  とリレーすると同時にキャッシュする。最後にクライアント 3 は  $i-1$  を閲覧できる。

- その後、組織 Y 内にいるクライアント 4 が公開コンテンツ  $i-1$  を閲覧しようとした場合 (図 B.8)。

クライアント 4 はサーバ B に  $i-1$  をリクエストする ( $\vec{1}$ )。サーバ B は  $i-1$  をキャッシュしているため、サーバ  $\beta$  にリクエストする事なくクライアント 4 コンテンツを受け取る ( $\vec{2}$ )。よってクライアント 4 は  $i-1$  を閲覧できる。

- その後、組織 Z 内にいるクライアント 5 が公開コンテンツ  $i-1$  を閲覧しようとした場合 (図 B.9)。

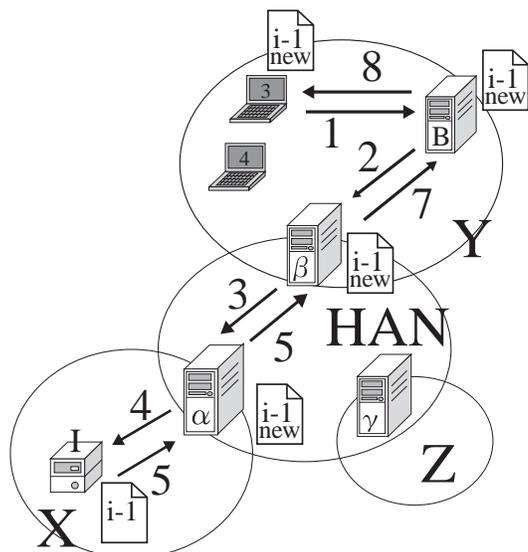


図 B.7 PIX 上の公開コンテンツに対するリクエスト

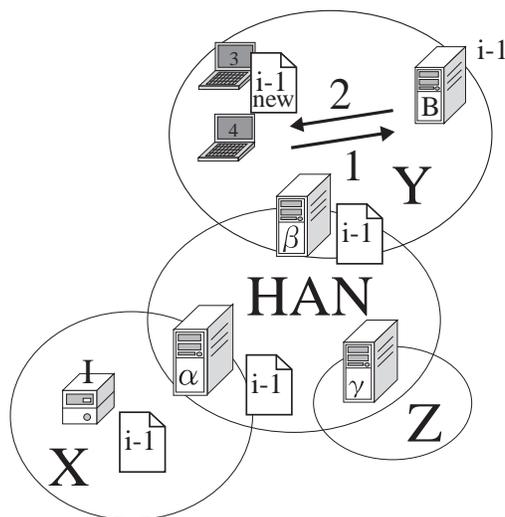


図 B.8 キャッシュされた公開コンテンツに対する同組織内からのリクエスト

クライアント 5 はサーバ C に  $i-1$  をリクエストする ( $\vec{1}$ )。サーバ  $\gamma, \beta$  は、リクエストを  $\vec{2}, \vec{3}$  とリレーする。サーバ  $\alpha$  は  $i-1$  をキャッシュしているので、サーバ I にリクエストする事なくサーバ  $\beta$  に返す ( $\vec{4}$ )。サーバ  $\gamma, C$  はコンテンツを  $\vec{5}, \vec{6}$  とリレーすると同時にキャッシュする。最後にクライアント 5 は  $i-1$  を閲覧できる。

- 組織 X におけるサーバ I に非公開コンテンツ  $i-2$  があり、組織 Z 内にいるクライアント

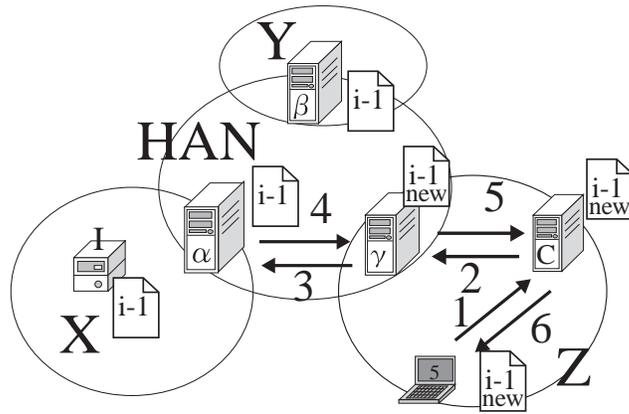


図 B.9 キャッシュされた公開コンテンツに対する他組織内からのリクエスト

5 が閲覧しようとした場合 (図 B.10)。

クライアント 5 からの  $i-2$  へのリクエストは、公開コンテンツに対するリクエストと同様にサーバ C、 $\gamma$ 、 $\alpha$ 、I は、リクエストを  $\bar{1}$ 、 $\bar{2}$ 、 $\bar{3}$ 、 $\bar{4}$  とリレーする。サーバ I は、リクエスト元が組織 X 内の IP アドレスを持たないサーバ  $\alpha$  であるため、否定応答を返す ( $\bar{5}$ )。否定応答は、サーバ  $\alpha$ 、 $\gamma$ 、C が  $\bar{6}$ 、 $\bar{7}$ 、 $\bar{8}$  とリレーし、クライアント 5 は  $i-2$  を閲覧できない。

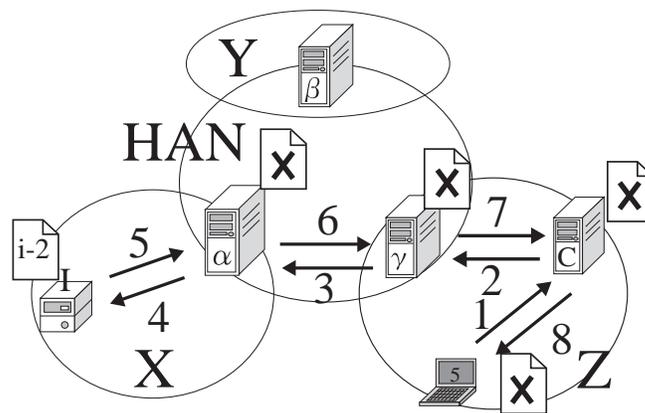


図 B.10 非公開コンテンツに対するリクエスト

- インターネット上にあるサーバ IV に公開コンテンツ  $i-3$  があり、組織 X 内にいるクライアント 2 が閲覧しようとした場合 (図 B.11)。

クライアント 2 はサーバ A に  $i-3$  をリクエストする ( $\bar{1}$ )。サーバ A は、サーバ IV にリ

クエスト<sup>2</sup>し、コンテンツを受け取る(3)。受け取ったコンテンツをサーバ A はキャッシュして、クライアント 2 に返す(4)。最後にクライアント 2 は閲覧できる。

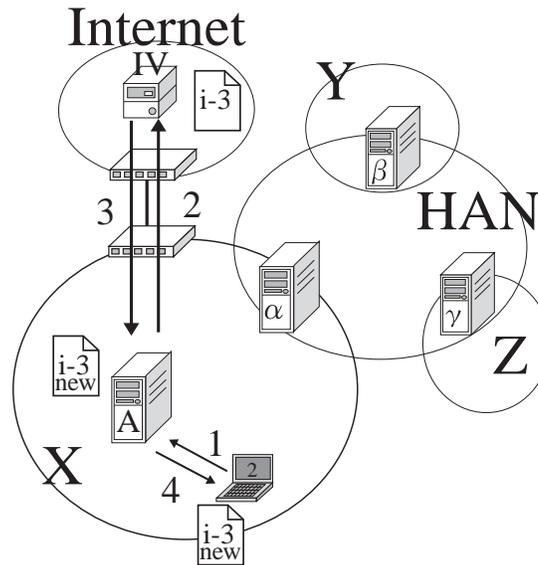


図 B.11 インターネット上の公開コンテンツに対するリクエスト

## B.2.6 各サーバの動作

ここでは、図 B.6 の連携パターンにおいて、第 B.2.5 節の動作を行うための各サーバごとの動作を示す。表 B.2 と表 B.3 とは、それぞれサーバ A とサーバ  $\alpha$  との動作を示す。縦軸にリクエスト元を、横軸にコンテンツのある場所をとっている。Y と Z のサーバも同様である。

## B.2.7 Squid によるサーバ群の実現

表 B.2 と表 B.3 とに示すサーバの動作を Squid を例にして説明する。

サーバ A と、サーバ  $\alpha$  をそれぞれ Squid A、Squid  $\alpha$  とし、squid.conf の設定例を表 B.4 と表 B.5 に示す。

ここでは、組織 X に IP アドレス 192.168/16 を割り当て、Squid A を 192.168.8.10 とす

表 B.2 組織 X 内キャッシュサーバ A の動作

リクエスト元	コンテンツのある場所		
	組織 X 内	組織 Y、Z 内	インターネット
X のクライアント	直接 *1	$\alpha$ にリクエスト *1	直接 *1
サーバ $\alpha$ *2	拒否 *2	拒否 *2	拒否 *2
サーバ $\beta, \gamma$ *3	拒否 *3	拒否 *3	拒否 *3
インターネット *4	拒否 *4	拒否 *4	拒否 *4

\*1 キャッシュされていればそれを用いてコンテンツを返す。

\*2  $\alpha$  の設定が正しければありえない。対設定ミス。

\*3  $\beta, \gamma$  の設定が正しければありえない。対設定ミス。

\*4 組織 X のインターネット側ファイアウォールで止めるのが普通である。

る。同様に HAN に 172.31/16 を割り当て、Squid  $\alpha$  を 172.31.8.3 とする。

#### [Squid A の動作]\*1

- 組織 X 内からのリクエストを許可し、それ以外からのリクエストは拒否する (13, 14)。
- KPIX 参加組織コンテンツへのリクエストについては、必ず squid $\alpha$  に Parent でリクエストする (4, 5, 10)。
- 上記以外へのリクエストは、直接リクエストする (11)。

#### [Squid $\alpha$ の動作]\*2

- 組織 X 内にあるサーバ I への HAN 内からのリクエストと、SquidA からのリクエストを許可し、それ以外からのリクエストは拒否する (16, 17, 18)。

\*1 動作の文末の数値は、表 B.4 の行番号に対応する。

\*2 動作の文末の数値は、表 B.5 の行番号に対応する。

表 B.3 組織 X に対応する HAN 内キャッシュサーバ  $\alpha$  の動作

リクエスト元	コンテンツのある場所		
	組織 X 内	組織 Y、Z 内	インターネット
X のクライアント *1	拒否 *1	拒否 *1	拒否 *1
サーバ A	拒否 *2	Y なら $\beta$ にリクエスト *3	拒否 *2
		Z なら $\gamma$ にリクエスト *3	
サーバ $\beta, \gamma$	直接 *3	否定応答 *3	拒否 *4
インターネット *5	拒否 *5	拒否 *5	拒否 *5

\*1 クライアントの設定が正しければありえない。

\*2 A の設定が正しければありえない。対設定ミス。

\*3 キャッシュされていればそれを用いてコンテンツを返す。

\*4  $\beta, \gamma$  の設定が正しければありえない。対設定ミス。

\*5 HAN がプライベートアドレスを用いるならありえない。グローバルアドレスであっても、A のインターネット側ファイアウォールで止めるのが普通である。

- 組織 X 内にあるサーバ I へのリクエストは、直接リクエストする (13)。
- KPIX 参加組織のコンテンツへのリクエストは、それぞれ対応する HAN 内 Squid  $\beta, \gamma$  に Parent でリクエストする (4, 5, 6, 7)。
- 上記以外のコンテンツへのリクエストは、許可しない (16, 17, 18)。

## B.3 コンテンツを共有するサーバ間の連携

KPIX 内で交換されるすべてのコンテンツを交換・共有するサーバ間の連携パターンを考案した。

具体的には、B.2 の参加組織のコンテンツを共有するとともに、参加組織内でリクエスト

表 B.4 Squid A の設定

```

1  visible_hostname A.X.or.jp
2  http_port 3128
3  icp_port 3130
4  cache_peer α.X.KPIX.net parent 3128 3130
5  cache_peer_domain α.X.KPIX.net .Y.or.jp .Z.or.jp
6  acl all src 0.0.0.0/0
7  acl X_SRC src 192.168.0.0/16
8  acl PIX_DST dstdomain .Y.or.jp .Z.or.jp
9  acl X_DST dstdomain .X.or.jp
10 never_direct allow PIX_DST
11 always_direct allow !PIX_DST
12 icp_access allow all
13 http_access allow X_SRC
14 http_access deny all

```

される参加組織以外のコンテンツを HAN 内でキャッシュして共有する、サーバ間の連携パターンを決定した。

### B.3.1 リクエストに対するサーバ間の連携

ここでは、参加組織以外のコンテンツへのリクエストとに対してのサーバ間の連携について具体的な動作を示す。

初期状態としてコンテンツはどのサーバにもキャッシュされていないものとする。図中の矢印はリクエストの流れを、矢印に付した数字は流れの順序を表す。

- インターネット上にあるサーバ IV に公開コンテンツ i-3 があり、組織 X 内にあるクラ

表 B.5 PIX Squid  $\alpha$  の設定

```

1  visible_hostname  $\alpha$ .X.PIX.net
2  http_port 3128
3  icp_port 3130
4  cache_peer  $\beta$ .Y.KPIX.net parent 3128 3130
5  cache_Peer_domain  $\beta$ .Y.KPIX.net .Y.or.jp
6  cache_peer  $\gamma$ .Z.KPIX.net parent 3128 3130
7  cache_peer_domain  $\gamma$ .Z.KPIX.net .Z.or.jp
8  acl all src 0.0.0.0/0
9  acl PIX_SRC src 172.31.0.0/16
10 acl X_A src 192.168.8.10/32
11 acl X_DST dstdomain .X.or.jp
12 acl PIX_DST dstdomain .Y.or.jp .Z.or.jp
13 always_direct allow X_DST
14 never_direct allow all
15 icp_access allow all
16 http_access allow PIX_SRC X_DST
17 http_access allow X_A PIX_DST
18 http_access deny all

```

クライアント 2 が閲覧しようとした場合 (図 B.12)。

クライアント 2 はサーバ A に  $i-3$  をリクエストする ( $\vec{1}$ )。サーバ A は、サーバ  $\alpha$  にリクエスト  $\vec{2}$  を出す。サーバ  $\alpha$  は他の HAN 内キャッシュサーバ  $\beta, \gamma$  にキャッシュされているか問い合わせる ( $\vec{3}, \vec{3}'$ )。各キャッシュサーバにはキャッシュされていないので、サーバ  $\alpha$  に否定応答を返す ( $\vec{4}, \vec{4}'$ )。サーバ  $\alpha$  は問い合わせたサーバすべてから否定応答をもらったら、サーバ A に  $i-3$  のリクエスト  $\vec{5}$  を出す。サーバ A はサーバ IV にリク



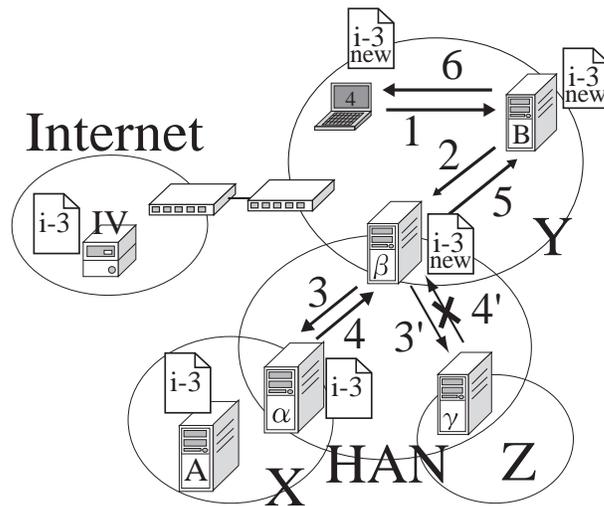


図 B.13 キャッシュされた公開コンテンツに対するリクエスト

## B.4 コンテンツを共有する Squid の設定

第 B.3 節のサーバ間の連携パターンの決定により、参加組織コンテンツを共有し、なおかつ参加組織以外のコンテンツも共有する Squid の設定を行った。

### B.4.1 各サーバの動作

ここでは、参加組織のコンテンツを共有する B.2.5 の動作を行い、なおかつ、B.3.1 の動作を行うための各サーバごとの動作を示す。表 B.6 と表 B.7 とは、それぞれサーバ A とサーバ  $\alpha$  との動作を示す。行にリクエスト元を、列にコンテンツのある場所をとっている。Y と Z のサーバも同様である。

### B.4.2 Squid によるサーバ群の実装

表 B.6 と表 B.7 とに示すサーバの動作を Squid を例にして説明する。

サーバ A と、サーバ  $\alpha$  をそれぞれ Squid A、Squid  $\alpha$  とし、squid.conf の設定例を表 B.8 と表 B.9 に示す。

ここでは、組織 X に IP アドレス 192.168/16 を割り当て、Squid A を 192.168.8.10 とす

表 B.6 組織 X 内キャッシュサーバ A の動作

リクエスト元	コンテンツのある場所		
	組織 X 内	組織 Y、Z 内	インターネット
X のクライアント	直接 * <sup>1</sup>	$\alpha$ にリクエスト * <sup>1</sup>	$\alpha$ にリクエスト * <sup>1</sup>
サーバ $\alpha$	直接 * <sup>1</sup>	拒否	直接 * <sup>1</sup>
サーバ $\beta, \gamma$ * <sup>2</sup>	拒否 * <sup>2</sup>	拒否 * <sup>2</sup>	拒否 * <sup>2</sup>
インターネット * <sup>3</sup>	拒否 * <sup>3</sup>	拒否 * <sup>3</sup>	拒否 * <sup>3</sup>

\*<sup>1</sup> キャッシュされていればそれを用いてコンテンツを返す。 \*<sup>2</sup> PIX のルーティングの設計で止める事も可能である。 \*<sup>3</sup> 組織 X のインターネット側ファイアウォールで止めるのが普通である。

る。同様に HAN に 172.31/16 を割り当て、Squid  $\alpha$  を 172.31.8.3 とする。

#### [Squid A の動作]\*<sup>3</sup>

- 組織 X 内からと Squid  $\alpha$  からのリクエストを許可し、それ以外からのリクエストは拒否する (13, 14, 15)。
- 組織 X 内のコンテンツサーバ I へのリクエストは、直接リクエストする (9)。
- 上記以外へのリクエストについては、必ず Squid  $\alpha$  に Parent でリクエストする (4, 11)。
- Squid  $\alpha$  からのリクエストは、直接リクエストする (10)。

#### [Squid $\alpha$ の動作]\*<sup>4</sup>

- 組織 X 内にあるサーバ I への HAN 内からのリクエストと、Squid A からのリクエストを許可し、それ以外からのリクエストは拒否する (16, 17, 18)。

\*<sup>3</sup> 動作の文末の数値は、表 B.8 の行番号に対応する。

\*<sup>4</sup> 動作の文末の数値は、表 B.9 の行番号に対応する。

表 B.7 組織 X に対応する HAN 内キャッシュサーバ  $\alpha$  の動作

リクエスト元	コンテンツのある場所		
	組織 X 内	組織 Y、Z 内	インターネット
X のクライアント *1	拒否 *1	拒否 *1	拒否 *1
サーバ A	拒否 *2	Y なら $\beta$ にリクエスト *3 Z なら $\gamma$ にリクエスト *3	$\beta, \gamma$ にリクエスト *3 無ければ A にリクエスト
サーバ $\beta, \gamma$	直接	否定応答 *3	否定応答 *3
インターネット *4	拒否 *4	拒否 *4	拒否 *4

\*1 クライアントの設定が正しければありえない。\*2 A の設定が正しければありえない。対設定ミス。\*3 キャッシュされていればそれを用いてコンテンツを返す。\*4 HAN がプライベートアドレスを用いるならありえない。グローバルアドレスであっても、A のインターネット側ファイアウォールで止めるのが普通である。

- 組織 X 内にあるサーバ I へのリクエストは、直接リクエストする (13)。
- 上記以外のすべてのリクエストは他のキャッシュ・プロキシを使ってリクエストする (14)。
- KPIX 参加組織のコンテンツへのリクエストは、それぞれ対応する HAN 内 Squid  $\beta, \gamma$  に Parent でリクエストする (5, 7)。
- 上記以外のコンテンツへのリクエストは、HAN 内 Squid  $\beta, \gamma$  に Sibling でリクエストする (4, 6)。
- Sibling でキャッシュの無かったリクエストについては、Squid A に Parent でリクエストする (8)。

以上のことを、2000 年 11 月 21 日に香川県クアパーク津田にて行われた第 8 回 ITRC 総会・研究会、および、同年 12 月 1 日に和歌山県和歌山大学にて行われた 2000 年度第 4 回

表 B.8 Squid A の設定

```
1 visible_hostname A.X.or.jp
2 http_port 3128
3 icp_port 3130
4 cache_peer  $\alpha$ .X.KPIX.net parent 3128 3130
5 acl all src 0.0.0.0/0
6 acl X_SRC src 192.168.0.0/16
7 acl PIX_ $\alpha$  src 172.31.8.3/32
8 acl X_DST dstdomain .X.or.jp
9 always_direct allow X_DST
10 always_direct allow PIX_ $\alpha$ 
11 never_direct allow all
12 icp_access allow all
13 http_access allow X_SRC
14 http_access allow PIX_ $\alpha$ 
15 http_access deny all
```

DSM 研究会にて発表した。

また、同年 12 月 6 日に高知工科大で行われた「第 6 回 KPIX 会議」において、正式に KPIX のサーバの設定として採用が決定した。

## B.5 NTP による時刻同期

この節では NTP の必要性とその設定について述べる。

表 B.9 PIX Squid  $\alpha$  の設定

```

1  visible_hostname  $\alpha$ .X.PIX.net
2  http_port 3128
3  icp_port 3130
4  cache_peer  $\beta$ .Y.KPIX.net sibling 3128 3130
5  neighbor_type_domain  $\beta$ .Y.KPIX.net parent .Y.or.jp
6  cache_peer  $\gamma$ .Z.KPIX.net sibling 3128 3130
7  neighbor_type_domain  $\gamma$ .Z.KPIX.net parent .Z.or.jp
8  cache_peer  $\alpha$ .X.or.jp parent 3128 3130
9  acl all src 0.0.0.0/0
10 acl PIX_SRC src 172.31.0.0/16
11 acl X_A src 192.168.8.10/32
12 acl X_DST dstdomain .X.or.jp
13 always_direct allow X_DST
14 never_direct allow all
15 icp_access allow all
16 http_access allow PIX_SRC
17 http_access allow X_A
18 http_access deny all

```

### B.5.1 NTP とは

NTP (Network Time Protocol)[Mil92] とはネットワーク上の複数のノードの時刻を同期させるプロトコルである。NTP は client-server 方式をとっており、client は server より時刻情報を受け取り、時刻同期を行う。また、client-server は階層構造をとることが可能であり、階層のどこに位置するかは `stratum` の値でわかる。階層の頂点に位置する server は

stratum1 である。また、同一 stratum のノード同士で互いに相手の時刻情報を用いて同期を行うことも可能である。

### B.5.2 NTP の必要性

PIX ではキャッシュサーバを用いてトラフィックを交換する。そのためキャッシュサーバのログを常に監視し、正常なサービスを継続的に提供する必要がある。また、トラフィックの交換が正しく行われているか確認するためにも複数のキャッシュサーバのログを比較することが必要になる。

### B.5.3 NTP の構成

KPIX では時刻情報を得るためにいくつかの組織に stratum1 の NTP server を設置した。PC を stratum1 の NTP server として使用するために、古野電気株式会社の GPS Time Receiver, TS-820<sup>\*5</sup>を用いた。また、NTP サーバの OS として FreeBSD<sup>\*6</sup>を、サーバソフトとして ntp<sup>\*7</sup>を用いた。TS-820 は GPS から時刻情報を取得して、その情報を 1PPS[MMB+00] として PC に送出する。この 1PPS を PC 側で取得することで極めて正確に PC の時刻あわせを行うことが可能であり、このようにして時刻を設定した PC を stratum1 の NTP server として使用することができる。

### B.5.4 NTP の設定

まず、FreeBSD が PPS 信号を受け取るように kernel を再構築する。kernel config ファイルに

- options PPS\_SYNC

---

<sup>\*5</sup> TS-820 <http://www.furuno.co.jp/gps/prod6.htm>

<sup>\*6</sup> FreeBSD 友の会 <http://www.jp.FreeBSD.org/>

<sup>\*7</sup> TimeServer <http://www.eecis.udel.edu/ntp/>

の一行を書き加えて kernel の再構築を行う。

TS-820 には癖があってそのままでは ntp が使用できないため、ソースを展開し、NetBSD での設定方法<sup>\*8</sup>を参考にして refclock\_nmea.c を編集する。

```
gps_send(pp->io.fd, "$PMOTG,RMC,0000*1D\r\n", peer);
```

という行が 2 ヶ所にあるので、はじめを

```
gps_send(pp->io.fd, "$PFEC,GPInt,ZDA00,GGA00,GSV00,VTG00,RMC01\r\n", peer);
```

に修正し、2 番目を消す。

その後ドキュメントに書かれている通りに configure, make, make install を行う。

TS-820 をシリアルポートに接続し、/dev で以下のコマンドを実行する。

```
# ln -s cuaa0 gps1
```

/usr/local/etc/ntp.conf を以下のようにする。

```
server 127.127.20.1
fudge 127.127.20.1 time1 -1.0
driftfile /usr/local/etc/ntp.drift
```

以下のコマンドを実行して様子を見る。

```
/usr/local/bin/ntpd -c /usr/local/bin/ntp.conf
```

動作を確認して、上記コマンドが起動時に実行されるようにする。

動作しない場合は、TS-820 の 1PPS ランプが点滅していることを確認する。点滅していない場合は、アンテナの設置場所が悪いなどの理由により衛星から時刻情報を取得できていない状態である。アンテナを、全天が見渡せる場所に設置することで解決できる。全天が見渡せる場所が確保できない場合も、可能な限り広く空が見渡せる場所に設置する。

---

<sup>\*8</sup> GPS/NetBSD/SS1 <http://lips.is.kochi-u.ac.jp/NTP/>

また、TS-820 が GPS の信号を受信していても NTP サーバがその信号を受信していない可能性がある。これを確認するために、

```
# tip cuaa0c
```

を実行して、TS-820 から来る信号を監視する。時刻・位置情報が来ていればよい。

以上の確認をしたら、利用できるいくつかの NTP サーバと時刻を比較して正しい時刻になっているか確認する。正しい時刻を示していない場合は、以上の作業のいずれかに失敗している可能性がある。再度手順を確認して作業を行う。

以上の手順で stratum1 の NTP server を利用することができるようになる。

## B.6 MRTG による測定

MRTG (Multi Router Traffic Grapher) とは、Tobias Oetiker 氏が 1994 年、インターネット回線の負荷状況を WEB 上で監視するために作成されたフリーのソフトウェアである。MRTG は SNMP (Simple Network Management Protocol) により値を受け取り、ネットワーク等の状況をグラフ化する。SNMP を使用するため、SNMP が実装された機器であれば MRTG による監視が可能である。例えば、SNMP を実装した測定器の測定結果である外界の温度、湿度、風力、 / 線などを定期的にグラフ化し WEB 上で監視することもできる。

MRTG は以前、WEB 上のグラフを GIF フォーマットで作成していた。しかし、近年の GIF ライセンス問題<sup>\*9</sup> により PNG を使用するようになった。

この MRTG を用いて KPIX に接続しているある一つの Squid サーバのトラフィック状況等を測定した (図 B.14、図 B.15)。この Squid サーバは、NTP client になるための設定をおこなっているため、極めて正確な時刻を得ている。そのため、正しい時刻のトラフィック状況の監視が可能になっている。

また、KPIX のデータリンク層を構成している KCAN では、無線リンクの状況を ping コ

---

<sup>\*9</sup> <http://www.unisys.com/unisys/lzw/lzw-license.asp>

マンドと MRTG を用いて監視しておいる。このため、無線リンクの切断などのトラブルに対する素早い対応を可能にしている。

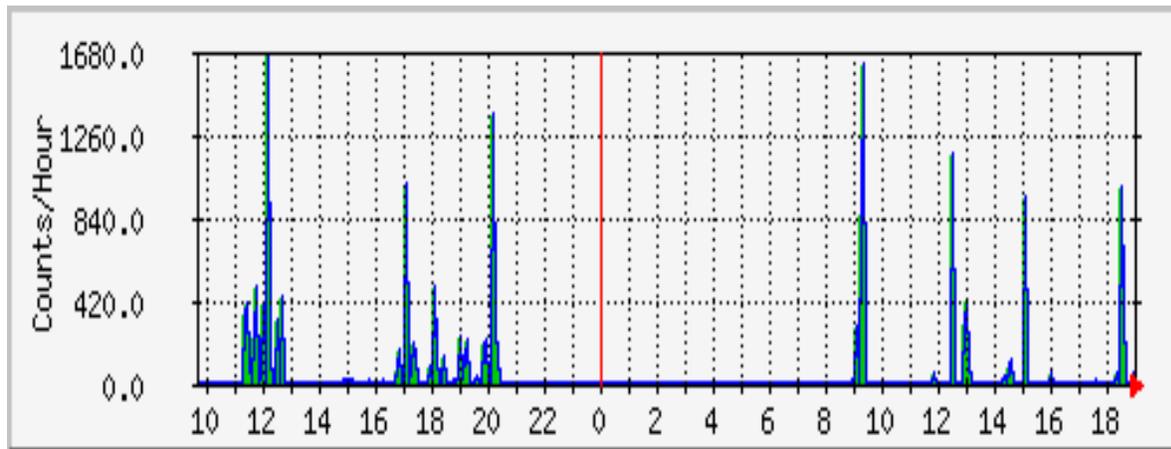


図 B.14 日間キャッシュヒット

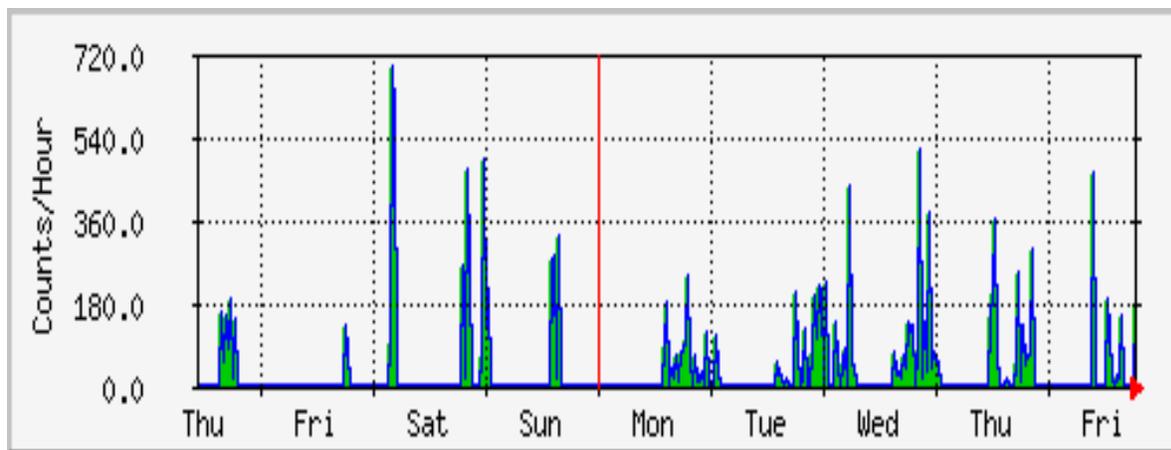


図 B.15 週間キャッシュヒット

これらのデータを含め、表 B.10 に表す値についても測定をおこなっている。

## B.7 PIX モデルの諸問題とその解決手法

PIX モデルの実証実験の場である KPIX の設計の際の 2 つの問題とその解決法を述べる [西内 00]。

cacheServerRequests	cacheServerErrors	cacheServerInKb
cacheServerOutKb	cacheClientHttpRequests	cacheHttpHits
cacheHttpErrors	cacheIcpPktsSent	cacheIcpPktsRecv
cacheIcpKbSent	cacheIcpKbRecv	cacheHttpInKb
cacheHttpOutKb	cacheCurrentSwapSize	cacheNumObjCount
cacheCpuUsage	cacheMemUsage	cacheSysPageFault

表 B.10 MRTG 測定項目一覧

### B.7.1 PIX モデル設計時の問題点

問題の一つは、プライベートアドレスの衝突、枯渇である。PIX モデルはプライベートアドレスを用いて各参加組織をつなぐ。そのため、全参加組織が使用していないプライベートアドレスを見つけだすことが必要となる。全参加組織が使用していないプライベートアドレスがある場合、そのプライベートアドレスを使用すれば問題なく PIX モデルに移行できる。しかし、プライベートアドレスは各組織毎、自由に使用しているため、参加組織が使用しているプライベートアドレスが重複する場合がでてくる。また、重複していない場合でも、空きがいわゆる「虫食い」状態になっていて必要な大きさの空間を確保できない場合がある。

もう一つの問題は、既存のネットワークの上に別のアドレス空間をもつ PIX モデルを導入する際への手法である。これは、運営している IP ネットワークがあり、そのネットワークを PIX モデルで使用したとする。この時、IP ネットワークのポリシーが異なるなどの理由で管理上互いに独立のネットワークにしたい場合がある。

### B.7.2 問題点に対する解決法

以上の問題解決の方法として、前者では NAT を使用する解決法を提案した。各 PIX 参加組織が使用していないプライベート IP アドレスと PIX 用のプライベート IP アドレスをルータの NAT 機能を使用して「1 対 1」静的対応させることで、PIX 内からはある一つの

プライベート IP アドレスを使用しているかのように見せることができる。

後者では IP トンネリングを用いた解決法を提案した。IPsec の技術を使用することでパケットをカプセル化するため、既存のネットワークを意識せずに PIX ネットワークとして使用する事ができる。

### B.7.3 KPIX での対応

これら 2 つの問題は、実際 KPIX でも問題となったことである。しかし、これら 2 つの実証実験を小規模な実験ネットワークを構築し検証した直後、KCAN では IP アドレスのリナンバリングがおこない、NAT を使用することなく IP アドレスの衝突を回避した。

IP トンネリングについても、KCAN 管理・運用者より不満の声があがらなかったため、使用する機会がなくなった。

しかしこれら 2 つの問題は、一般的な PIX モデルに付いても十分当てはまる問題であると考え、我々はこれらの解決法を利用して問題を回避できると考える。

## B.8 富山国体映像中継

平成 12 年 10 月 14 日から 19 日に富山県にて 2000 年とやま国体秋季大会が開催された。開催と同時に、RIBB では、富山国体における高度情報通信技術を利用したマルチメディア配信実験を実施した。

### B.8.1 JGN を利用したデジタルビデオによる高品質映像の配信実験

菊池研究室では JGN を利用したデジタルビデオによる高品質映像の配信実験に参加した。この配信実験は、JGN による超高速通信を利用し、デジタルビデオによる高品質映像を富山県内外の接続拠点に配信し、大型スクリーンなどで上映し、超高速ネットワークを使った映像配信に関するノウハウを取得することを目標とする。

## B.8.2 映像ソース

富山県にて開催される 2000 年とやま国体秋季大会の様子を CATV が撮影して、富山総合情報センターで 1 つにまとめた映像をソースとした。

## B.8.3 トポロジー

富山国体中継の全体トポロジーを図 B.16 に示す。富山で撮影された映像は、JGN を用いて全国 8 都県（福島県、宮城県、山梨県、東京都、岐阜県、富山県、高知県、福岡県）へ配信される。配信先の全国 8 都県では、富山国体の模様を視聴することが可能である。また、インターネットによる配信も行ったため、インターネット経由の視聴が可能である。

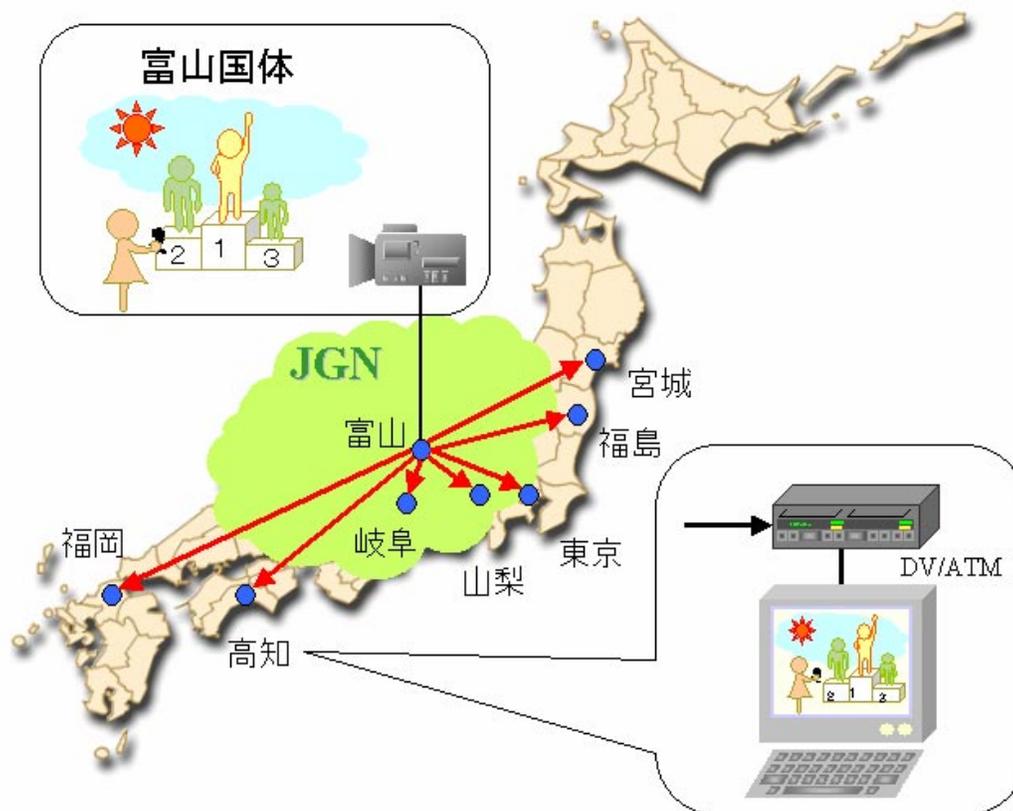


図 B.16 富山国体中継全体トポロジー

図 B.17 に、本研究で携わった部分のトポロジーを示す。富山国体の映像は富山県の撮

影地点から富山総合情報センターへ送られる。そして、JGN を経由して TAO 高知通信トラヒックリサーチセンターに送られ、高知工科大学菊池研究室の ATM スイッチである ASX200 まで送られる。また、富山総合情報センターから菊池研究室の ASX200 までの映像はすべて DV over ATM であり、1 本の PVC (Permanent Virtual Circuit) である。

ASX200 の設定を出力が 2 方向に出力されるように設定した。TAO 高知通信トラヒックリサーチセンターを経由して名古屋と岐阜へ再配信する流れと、菊池研究室の丹箱へ出力する流れの 2 方向である。さらに、菊池研究室の丹箱へ出力された映像を 3 方向に分ける。1 本目は、菊池研究室内のビデオデッキを用いて液晶テレビに表示する。2 本目は、WMT を用いて KPIX ヘストリーム配信する。3 本目は、RealSystem を用いて KPIX ヘストリーム配信する。よって、ASX200 以降の映像の流れは全部で 4 方向となる。以下では、この 4 方向を詳しく説明する。

#### 菊池研究室内の液晶テレビに表示

菊池研究室の ASX200 まで送られてきた映像を、DV over ATM で LinkUnit (図 B.17 中の丹箱。以下、丹箱と呼ぶ。)へ転送する。丹箱へ転送された映像を、DV over IEEE1394 でビデオデッキへ転送する。ビデオデッキへ転送された映像を、NTSC に変換し液晶テレビに出力する。

#### WMT を用いて KPIX ヘストリーム配信

菊池研究室の ASX200 まで送られてきた映像を、DV over ATM で丹箱へ転送する。丹箱へ転送された映像を、DV over IEEE1394 で DVMC (DV Media Converter) へ転送する。DVMC へ転送された映像を、NTSC (National Television Standards Committee) に変換し、ストリームエンコーダであり、ストリームサーバである phage に転送する。phage に転送された映像を、Windows Media Encoder でエンコードし、Windows Media Service で KPIX ヘストリーム配信する。

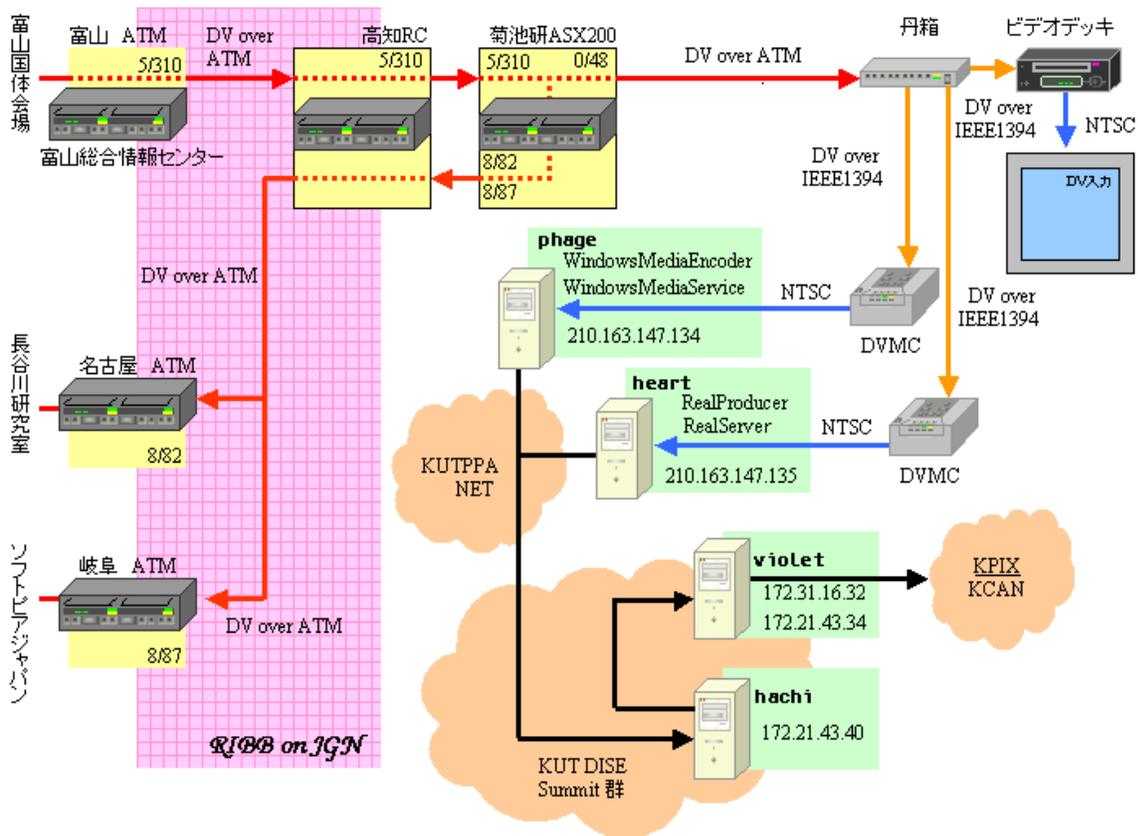


図 B.17 富山国体中継トポロジー

phage からストリーム配信される映像は、高知工科大学のネットワーク（図 B.17 中の KUTPPA NET）と高知工科大学情報システム工学科のネットワーク（図 B.17 中の KUT DISE Summit 群）を經由し、hachi まで転送される。hachi から violet に転送され、violet は KPIX へ転送する。hachi と violet はプロキシサーバである。hachi は自組織キャッシュサーバであり、violet は自組織 KPIX 側キャッシュサーバである。

### RealSystem を用いて KPIX へストリーム配信

WMT を用いて KPIX へストリーム配信と同様である。ただし、以下の部分が異なる。

ストリームエンコーダであり、ストリームサーバである heart に転送する。heart に転送された映像を、Real Producer でエンコードし、Real Server で KPIX へストリーム配信

する。

#### ASX200 を用いて名古屋と岐阜へ再配信

ASX200 まで転送された映像を、TAO 高知通信トラヒックリサーチセンターを介し、JGN を経由して名古屋大学長谷川研究室と岐阜ソフトピアジャパンへ再配信する。また、ASX200 から名古屋大学長谷川研究室と岐阜ソフトピアジャパンまでの映像はすべて DV over ATM であり、1 本の PVC である。

#### B.8.4 ストリーム配信の設定とスペック

今回の KPIX へのストリーム配信にはストリームエンコーダとストリームサーバを 1 台のマシンで行った。表 B.11、表 B.12、表 B.13、表 B.14 に WMT (phage) と RealSystem (heart) の設定を示し、phage と heart のスペックは表 B.15 に示す。

WindowsMediaService (phage)	
エイリアス	ToyamaKokutai
パスの種類	WindowsMedia エンコーダ
URL	http://210.163.147.134
ポート	8080
クライアント数	無制限

表 B.11 ストリームサーバの設定

#### B.8.5 KPIX への配信結果

14 日から 19 日の 6 日間、のべ約 60 時間に及ぶ全国 8 都県（福島県、宮城県、山梨県、東京都、岐阜県、富山県、高知県、福岡県）への中継は、パケットロスもほとんどなく十分な品質の動画を提供した。KPIX へのストリーム配信は、プロキシサーバを介さなかった場合

WindowsMediaEncoder (phage)	
最大ビットレート	512Kbps
オーディオ codec	Windows Media Audio V7
オーディオ形式	64kbps 44KHz stereo
ビデオ codec	Windows Media Video V7
ビデオサイズ	352 × 288 (CIF)
フレームレート	30fps
キーフレームの間隔	4 秒
画像の品質	0

表 B.12 ストリームエンコーダの設定

RealServer (heart)	
URL	http://172.31.16.40:8080/ramgen/encoder/live.rm
ポート	8080
クライアント数	25

表 B.13 ストリームサーバの設定

は視聴可能、プロキシサーバを介した場合は視聴不能であった。

以下に KPIX での富山国体中継視聴結果を示す (表 B.16)。KPIX 経由とは、KPIX 内の squid を経由した場合を指し、プロキシを介してストリーム配信サーバにアクセスする場合をいう。KCAN 経由とは、KPIX 内の squid を経由しない場合を指し、直接ストリーム配信サーバにアクセスした場合をいう。インターネット経由とは、KPIX 経由でもなく、KCAN 経由でもなく、インターネットを経由してアクセスした場合を指す。KPIX 経由ではアクセス不能で、KCAN 経由とインターネット経由ではアクセス可能であった。

RealProducer (heart)	
Target Audience	56K Modem Corporate LAN (150Kbps)
RealMedia Settings	Multi-rate SureStream
Audio Format	Voice Only
Video Quality	Smoothest Motion Video

表 B.14 ストリームエンコーダの設定

	WMT (phage)	RealSystem (heart)
HDD	30G	20G
CPU	PenIII866MHz	PenIII600MHz
メモリ	512MB	640MB
ビデオキャプチャボード	Osprey-100	Osprey-1000
OS	Windows2000 Server	WindowsNT Server 4.0
Server	WindowsMediaService (Windows2000 Server に付属)	RealServer 7.0 Basic (free)
Encoder	WindowsMediaEncoder 7 (free)	RealProducer 8 Basic (free)
Player	WindowsMediaPlayer 7 (free)	RealPlayer 8 Basic (free)

表 B.15 スペック比較

## B.9 ギガビットシンポジウム 2000 中継

平成 12 年 11 月 8 日に北九州国際会議場をメイン会場とし、「ギガビットネットワーク・シンポジウム 2000」が開催された。RIBB では、ギガビットネットワーク・シンポジウム 2000 の模様を JGN を用いて中継した。菊池研究室ではこのギガビットネットワーク・シンポジウム 2000 の映像を DVTS (DV Transfer System) を用いて DV over IEEE1394 から

経由	プロキシの有無	視聴
KPIX 経由	有	不能
KCAN 経由	無	可能
インターネット経由	無	可能

表 B.16 KPIX での富山国体中継視聴結果

DV over IP に変換し、IP Multicast で RIBB 他組織へ再配信すると同時に、WMT を用いて KPIX ヘストリーム配信した。

### B.9.1 映像ソース

北九州国際会議場で開催される、「ギガビットネットワーク・シンポジウム 2000」の様子 (DV) をソースとする。

### B.9.2 トポロジー

ギガビットシンポジウム 2000 中継の全体トポロジーを図 B.18 に示す。北九州国際会議場の映像は、JGN を用いて全国 4 県 (愛知県、高知県、富山県、山梨県) へ配信される。配信先の全国 4 県では、ギガビットシンポジウム 2000 の模様を視聴することが可能である。また、インターネットによる配信も行ったため、インターネット経由の視聴が可能である。

図 B.17 に、本研究で携わった部分のトポロジーを示す。北九州国際会議場の映像は、JGN を経由して名古屋大学大型計算機センターへ送られる。そして、JGN を経由して TAO 高知通信トラヒックリサーチセンターに送られ、菊池研究室の ASX200 まで送られる。ASX200 へ送られた映像は、丹箱へ送られる。また、名古屋大学大型計算機センターから菊池研究室の丹箱までの映像はすべて DV over ATM であり、1 本の PVC である。

ASX200 から送られてきた映像を丹箱で、3 方向に分ける。1 本目は、菊池研究室内のビデオデッキを用いて液晶テレビに表示する。2 本目は、WMT を用いて KPIX ヘストリー

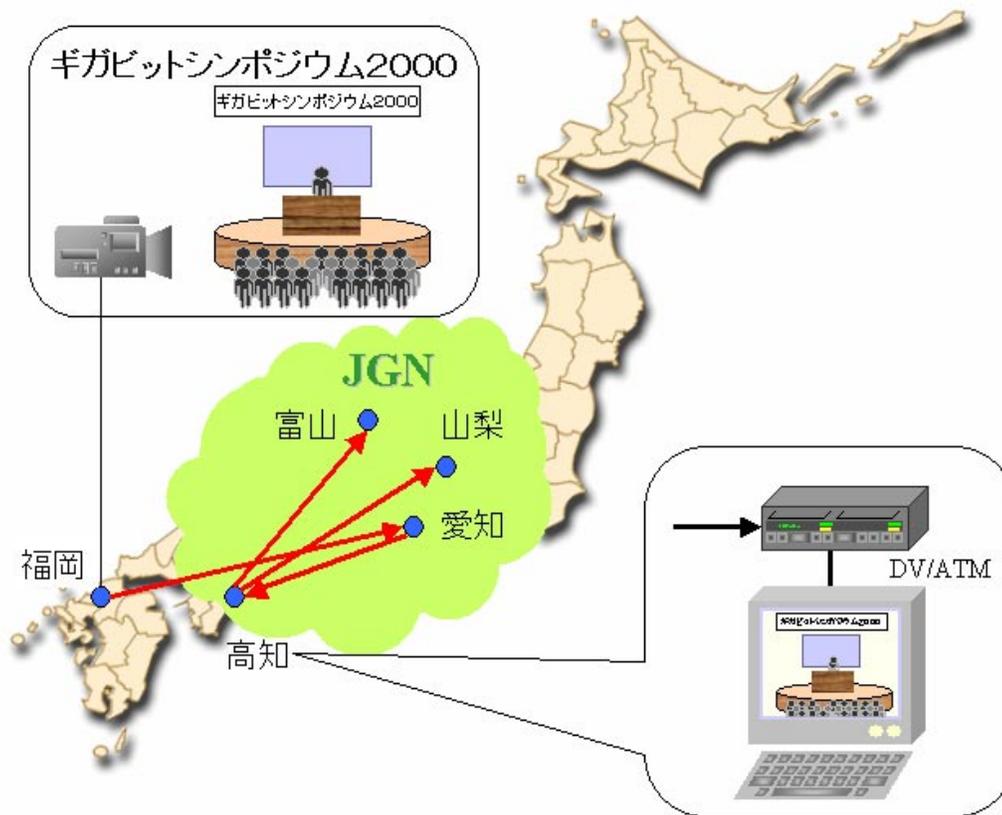


図 B.18 富山国体中継全体トポロジー

△配信する。3本目は、DVTSを用いてJGNを経由して富山と山梨に再配信する。以下では、この3方向を詳しく説明する。

菊池研究室内の液晶テレビに表示

富山国体中継の際と同様である。菊池研究室のASX200まで送られてきた映像を、DV over ATMで丹箱へ転送する。丹箱へ転送された映像を、DV over IEEE1394でビデオデッキへ転送する。ビデオデッキへ転送された映像を、NTSCに変換し液晶テレビに出力する。

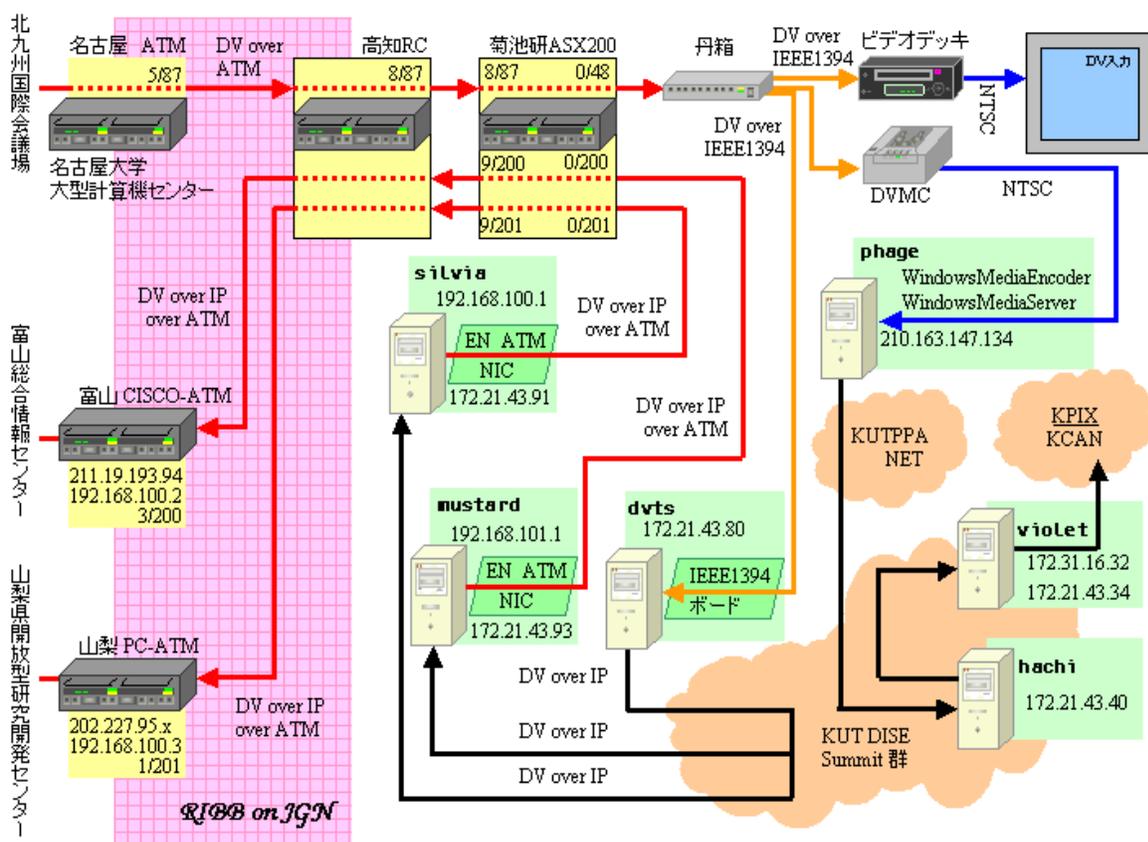


図 B.19 ギガビットシンポジウム 2000 トポロジー

## WMT を用いて KPIX ヘストリーム配信

富山国体中継の際と同様である。菊池研究室の ASX200 まで送られてきた映像を、DV over ATM で丹箱へ転送する。丹箱へ転送された映像を、DV over IEEE1394 で DVMC へ転送する。DVMC へ転送された映像を、NTSC に変換し、ストリームエンコーダであり、ストリームサーバである phage に転送する。phage に転送された映像を、Windows Media Encoder でエンコードし、Windows Media Service で KPIX ヘストリーム配信する。

phage からストリーム配信される映像は、高知工科大学のネットワーク（図 B.19 中の KUTPPA NET）と高知工科大学情報システム工学科のネットワーク（図 B.19 中の KUT DISE Summit 群）を経由し、hachi まで転送される。hachi から violet に転送され、violet は KPIX へ転送する。hachi と violet はプロキシサーバである。hachi は自組織キャッシュ

サーバであり、violet は自組織 KPIX 側キャッシュサーバである。

### DVTS を用いて富山と山梨に再配信

丹箱へ転送された映像を、DV over IEEE1394 で DVST (図 B.19 中の dvts) へ転送する。dvts で DV over IEEE1394 から DV over IP に変換する。dvts で DV over IP に変換した映像を IP multicast で、ATM ルータである silvia と mustard へ転送する。silvia と mustard では、DV over IP から DV over IP over ATM へ変換する。そして、DV over IP over ATM 変換した映像を ASX200 へ再配信する。

ASX200 では、以下のように設定する。silvia からの入力は TAO 高知通信トラヒックリサーチセンターを介して JGN を経由し山梨の山梨県開放型研究開発センターへ配信する。mustard からの入力は TAO 高知通信トラヒックリサーチセンターを介して JGN を経由して富山の富山総合情報センターへ配信する。また、ASX200 から山梨間と ASX200 から富山間の映像はすべて DV over IP over ATM であり、1 本の PVC である。

### B.9.3 ストリーム配信の設定とスペック

今回の KPIX へのストリーム配信にはストリームエンコーダとストリームサーバを 1 台のマシンで行った。表 B.17、表 B.18 に WMT (phage) の設定を示す。また、phage のスペックは富山国体中継の際と同じである (表 B.15 を参照)。

### B.9.4 KPIX への配信結果

のべ約 8 時間に及ぶ全国 4 県 (愛知県、高知県、富山県、山梨県) への中継は、20 ~ 30 秒毎に 2 回程度の配信元が原因らしきノイズが観測されたものの、視聴者にとっては問題なく視聴出来る品質であった。KPIX へのストリーム配信は、報告が得られず視聴結果は不明であった。

以下に KPIX でのギガビットシンポジウム 2000 中継視聴結果を示す (表 B.19)。また、

WindowsMediaService (phage)	
エイリアス	gb-sympo
パスの種類	WindowsMedia エンコーダ
URL	http://210.163.147.134
ポート	8080
クライアント数	無制限

表 B.17 ストリームサーバの設定

WindowsMediaEncoder (phage)	
最大ビットレート	512Kbps
オーディオ codec	Windows Media Audio V7
オーディオ形式	64kbps 44KHz stereo
ビデオ codec	Windows Media Video V7
ビデオサイズ	352 × 288 (CIF)
フレームレート	30fps
キーフレームの間隔	4 秒
画像の品質	0

表 B.18 ストリームエンコーダの設定

表の見方は富山国体の際と同様である。

### B.9.5 まとめ

KPIX では、PIX モデルへのストリーム配信が可能か実験するためにギガビットシンポジウム 2000 中継に参加し、ギガビットシンポジウム 2000 中継を KPIX へストリーム配信した。

経由	プロキシの有無	視聴
KPIX 経由	有	不明
KCAN 経由	無	不明
インターネット経由	無	可能

表 B.19 KPIX でのギガビットシンポジウム 2000 中継視聴結果

配信結果より、現状の KPIX の設定でのストリーム配信が可能かは分からなかった。今後の課題には、squid.conf の考察やプロキシを介した KPIX へのストリーム配信の可能・不可能の調査がある。

## B.10 pac ファイルの作成

この節では pac ファイルの作成方法、およびブラウザへの設定方法などについて述べる。

### B.10.1 pac ファイルとは

pac (Proxy Auto Configuration) ファイル<sup>\*10\*11</sup>とは、ブラウザのプロキシ設定を自動化する設定ファイルである。pac ファイルは JavaScript 風の文法によって記述されたテキストファイルである。現在 pac ファイルを使用できるのは Netscape Navigator と Microsoft Internet Explorer である。

### B.10.2 pac ファイルの目的

pac ファイルを使用してプロキシの自動設定を行うことによって多数あるクライアントのブラウザの設定の変更をそれぞれに対して行う必要がなくなり、もとの pac ファイルを変更するのみで全体の設定を変更することが可能となる。また、ブラウザの動作についても細か

<sup>\*10</sup> Netscape の資料 <http://home.netscape.com/eng/mozilla/2.0/relnotes/demo/proxy-live.html>

<sup>\*11</sup> Internet Explorer 開発キット <http://www.microsoft.com/japan/ie/ieak/>

く規定することが可能であり、pac ファイルを使用してブラウザも PIX を動作に組み込むことでより柔軟なネットワーク設計を可能とする。

### B.10.3 pac ファイルの記述

pac ファイルは JavaScript に似た文法で記述される。ブラウザはユーザから URL へのリクエストを受け取ると、

```
function FindProxyForURL (url, host){}
```

という関数を呼び出す。引数として url はリクエストされた url 全体が、host には url に含まれる hostname が渡される。この関数は受け取った url, host をもとに、答えを返す。この関数が返すのは、以下の3つの値のどれかである。

- DIRECT: プロキシを使用せずに直接接続する
- PROXY: 使用するプロキシサーバとポート番号を返す
- SOCKS: 使用する SOCKS サーバとポート番号を返す

FindProxyForURL 関数の中では、一般的な JavaScript の関数の他にいくつかの定義済み関数を使用することができる。定義済み関数はいくつかのグループに分かれている。グループは以下の通りである。

- ホスト名による条件式
- IP アドレス、DNS に関する関数
- URL/hostname による条件式
- 時間による条件式
- 結合された配列

これらの関数を組み合わせることで柔軟なプロキシ設定を実現することが可能となる。

## B.10.4 pac ファイルの適用

pac ファイルはブラウザに読み込まれることで有効になる。そのため、何らかの方法でブラウザに読み込ませる必要がある。ただし、一度読み込ませれば pac ファイルが更新された場合の再読み込みは自動化されているのが一般的であるため、pac ファイルの読み込みの設定は最初の一度行えばよい。

pac ファイルをブラウザに読み込ませるには以下のようにいくつかの方法がある。

- ブラウザに pac ファイルの URL を設定する
- WPAD[CGC+00] によって自動検出を行う
  - DHCP を用いる
  - SLP を用いる
  - DNS を用いる
    - \* A lookup を用いる
    - \* SRV lookup を用いる
    - \* TXT lookup を用いる

WPAD による方法は現在では Microsoft Internet Explorer のみで使用することができる。

WPAD を使用するには、Internet Explorer のプロキシ設定の項目で

- 設定を自動的に検出する

にチェックを入れておくだけでよい。

これらの方法は上記のリストの順に優先順位が高い。すなわち、上から順に pac ファイルが検出され、見つかった時点でそれを適用する。現在の実装の状況から、一般的にはブラウザに設定する、あるいは DHCP を用いるか、DNS の A lookup を用いることが多い。

## B.11 コストモデル

PIX モデルでの回線維持費を調査することにより、PIX におけるコストモデルを考察した。

PIX モデルに参加している組織は、インターネットへの専用線と PIX 内への接続回線を持つ。この時、総トラフィックが一定だと仮定し、総トラフィックを満たす場合の月額回線維持費用を求めた（例：図 B.20）。

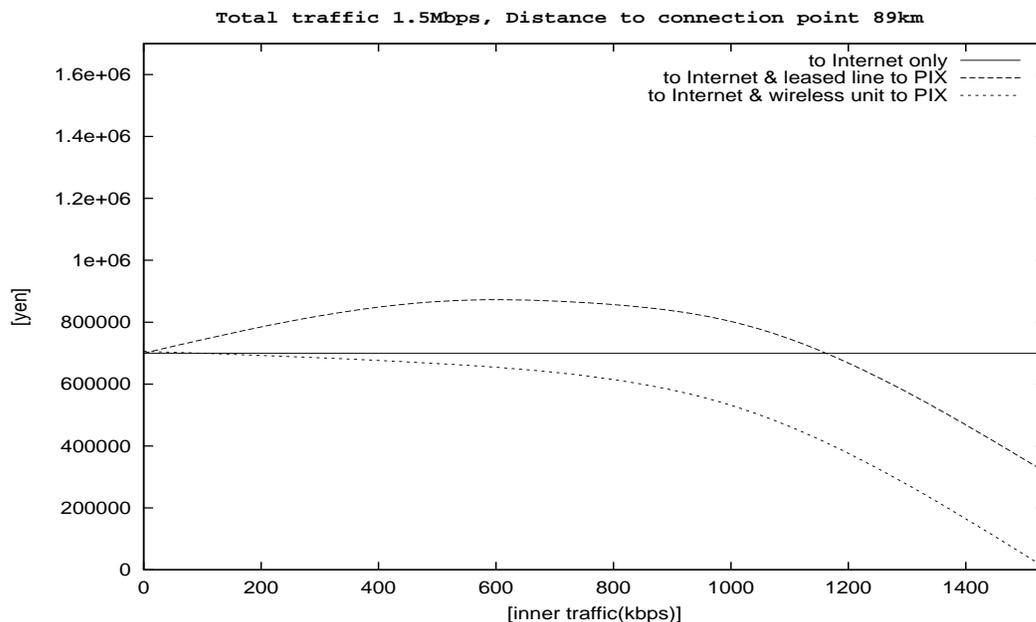


図 B.20 PIX モデルの月額回線維持費用例

このグラフを求めたことで、総トラフィックが大きい程、また地域内トラフィックの割合が大きい程月額回線維持費用は PIX が有利なことがわかった。また無線ユニットによる接続では、トラフィック状況にほぼ無関係に PIX が有利である。

# 参考文献

- [CGC<sup>+</sup>00] I. Cooper, P. Gauthier, J. Cohen, M. Dunsmuir, and C. Perkins. Web proxy auto-discovery protocol, November 2000. draft-cooper-webi-wpad-00.txt.
- [Mil92] David L. Mills. Network time protocol version 3, March 1992. RFC 1305.
- [MMB<sup>+</sup>00] J. Mogul, D. Mills, J. Brittonson, J. Stone, and U. Windl. Pulse-per-second api for unix-like operating systems, version 1.0, March 2000. RFC 2738.
- [安田 99] 安田豊, 中山雅哉. 日蝕中継における WWW 分散サーバ群の構築とその有効性. 情報処理学会研究報告, pp. 19–24. 分散システム運用技術研究会, July 1999. ISSN0919-6072.
- [笠原 97] 笠原義晃, 石田慶樹, 吉川善吾. 九州大学における WWW キャッシュサーバの運用と評価. 情報処理学会研究報告, pp. 49–54. 分散システム運用技術研究会, July 1997. ISSN0919-6072.
- [菊池 97] 菊池豊, 菊地時夫. PIX: 応用層によるトラフィック交換モデル. インターネットコンファレンス'97 論文集, pp. 159–162. 日本ソフトウェア科学会, December 1997. ISSN1341-870X.
- [菊池 99] 菊池豊, 菊地時夫. 応用層によるインターネットトラフィック交換モデル. コンピュータソフトウェア, Vol. 16, No. 4, pp. 46–58, July 1999.
- [菊池 00a] 菊池豊, 菊地時夫ほか. 高知応用層交換所の構築. 情報処理学会研究報告, pp. 49–54. 分散システム運用技術研究会, May 2000. ISSN0919-6072.
- [菊池 00b] 菊池豊, 菊地時夫, 今井一雅, 松本浩明, 濱崎哲一, 武市統, 今西孝也, 澤本一哲, 杉山道子, 西内一馬, 廣瀬崇夫, 正岡元, 蒲原浩, 寺田浩詔. 高知応用層交換所の構築. 情報処理学会研究報告, pp. 49–54. DSM 研究会, May 2000. ISSN0919-6072.
- [今井 00] 今井一雅, 澤本一哲, 矢野漣, 菊地時夫, 菊池豊. 高速無線 LAN システムによる

- 地域情報化ネットワークの構築と運用. 信学技報, pp. 39–44. 電子情報通信学会, January 2000.
- [正岡 00] 正岡元, 杉山道子, 西内一馬, 廣瀬崇夫, 菊池豊, 菊地時夫. 地域指向ネットワークモデル PIX における WWW サーバ群の構成について. 情報処理学会研究報告. インターネット技術研究委員会, November 2000.
- [西川 97] 西川記史, 細川貴史, 辻洋, 森靖英, 吉田健一. WWW トラフィック分析と分散キャッシュ. 情報処理学会研究報告, pp. 7–12. 分散システム運用技術研究会, May 1997. ISSN0919-6072.
- [西内 00] 西内一馬, 杉山道子, 廣瀬崇夫, 正岡元, 菊池豊. 高知地域指向疑似 IX の課題と解決法. 情報処理学会研究報告, pp. 25–30. 分散システム運用技術研究会, October 2000. ISSN0919-6072.
- [鍋島 99] 鍋島公章. キャッシュサーバの運用技術, dec 1999. Internet Week '99, Tutorial.
- [鍋島 00] 鍋島公章. キャッシュサーバの運用技術, dec 2000. Internet Week 2000, Tutorial.
- [矢吹 98] 矢吹 道郎監修, 長岡 秀行著. Squid プロキシサーバの設定と運用. テクノプレス, September 1998.