bonding機能紹介と展望

NECシステムテクノロジー(株)
ネットワークソフトウェア事業部

国本 英悟

目次

- 導入利点と機能
- モード毎の詳細
- 導入時の注意点と設定方法
- 最後に

導入利点と機能

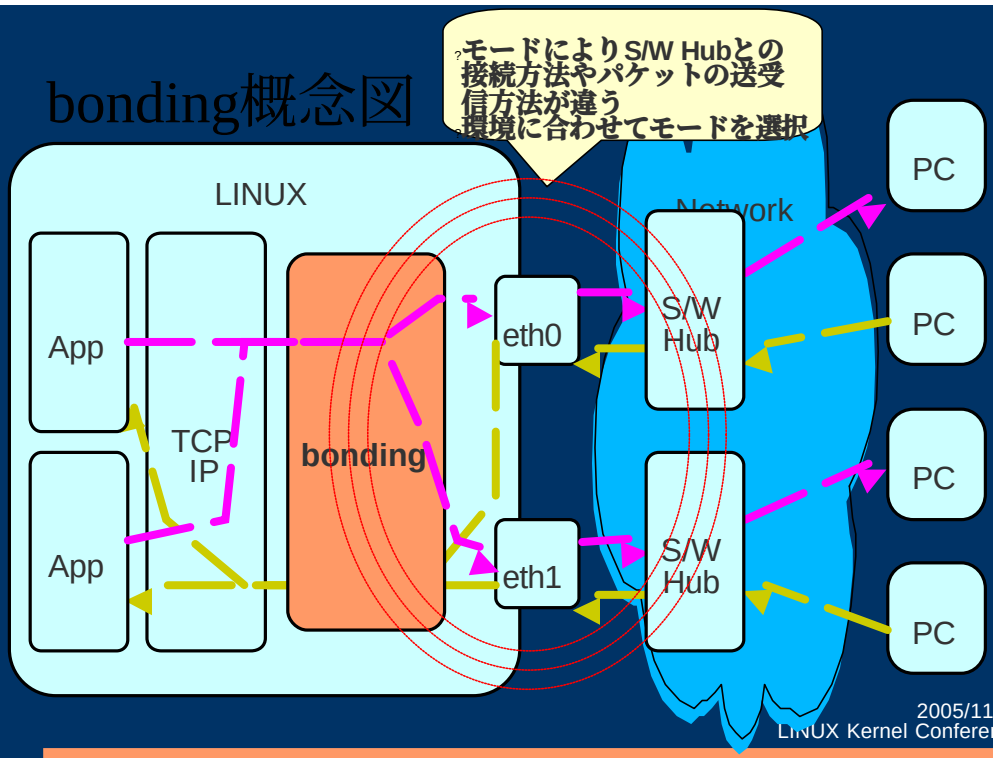
実現機能

- 擬似的なEthernetポートを構成する機能
  - 複数の物理的なEthernetポートを一つの疑似的なポートに集約
- 障害の自動検出と復旧機能
  - 障害を検出した物理的なポートを使用せず、復旧すれば元通り動作
  - 障害発生に伴う作業を無くす
- 送受信に使うポートの振り分け機能
  - MACアドレスやIPアドレスをキーにした分散
  - 通信負荷状況(ポート毎の通信レート)を元にした分散
  - NICの種類と環境に合わせたポートの選択

bonding導入による利点

- 耐障害性の向上
  - 最後の数mでの信頼性確保。直結したケーブルやS/W Hubの障害でもサービスを止めない
- 送受信パケットの負荷分散
  - 全てのNICを利用して送受信、マシンの持つ資源を最大限活用
- 導入し易い
  - 7つのモードから導入する環境に合わせて選択
  - NICベンダーに依存せず機能
  - LINUXにとって'軽い'
    - パケット受信処理に関与しない

bonding概念図



モード一覧

モード	(値)	用途	負荷分散		S/W障害時の耐障害性向上	S/W接続台数	必要なS/W機能	障害監視		送信元MACアドレス	備考
			送信	受信				MII	ARP		
balance-rr	0	高速送信	Round Robin	S/W依存	S/W依存	一台	トランキング	?	?	全ポート同じ	パケット到着順序の入れ替わり
active-backup	1	冗長化最優先の環境	×	×	?	複数	-	?	?	全ポートプライマリーポートと同じ	重複受信
					S/W依存	一台					
balance-xor	2	直結のネットワーク上のホストとの通信向け	MACアドレスのXOR	S/W依存	S/W依存	一台	トランキング	?	?	全ポート同じ	送信ポートの偏り
broadcast	3	※特殊用途向け	×	×	×(S/W間接続不可)	複数	-	?	?	全ポート同じ	特殊用途向け
			×(負荷を増長)	S/W依存	(?)	一台	トランキング				
802.3ad	4	直結のネットワーク上のホストとの通信向け	MACアドレスのXOR	S/W依存	S/W依存	一台	IEEE 802.3ad	MIIとLACP		全ポート同じ	送信ポートの偏り
balance-tlb	5	IPv4で運用	ポート毎の送信速度(IPv4)	×	?	複数	-	?	×	送信ポート別	フラッディング重複受信(Bcast)
					S/W依存	一台					
balance-alb	6	IPv4で運用	ポート毎の送信速度(IPv4)	ポートを通知(IPv4)	?	複数	-	?	×	送信ポート別	フラッディング重複受信(Bcast)ネットワーク監視
					S/W依存	一台					

導入の際に

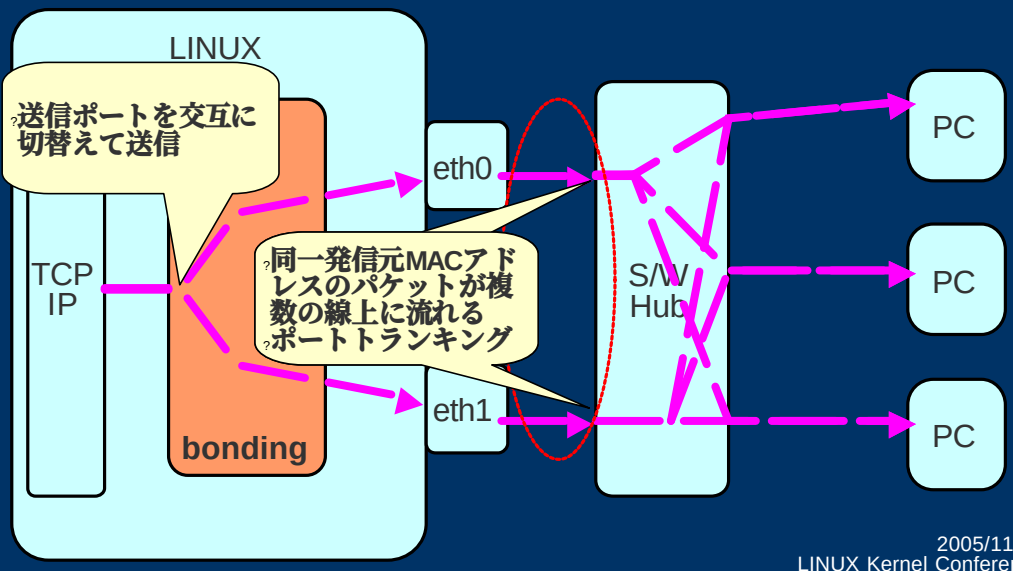
- モードの選択
  - 全モード耐障害性を実現
  - ネットワークの構成、通信相手、通信方法によりモードを選ぶ
- 導入時のモード選択キーワード
  - 通信負荷分散の必要性
  - ネットワークトポロジー
  - 接続するS/W Hubの機能
  - 通信負荷分散アルゴリズム
  - ネットワーク管理システム

モード毎の詳細

balance-rr(Round Robin)

- 送信するべきパケットをポートへ順々に振り分ける
  - 全NICを使用した最大の送信速度
- 通信上の特徴
  - 通信相手へのパケット到着順序が送信順序と入れ替わる
    - パケットの到着順に合わせて制御する様なAPは動かない
- S/W Hubの機能
  - ポートトランキング

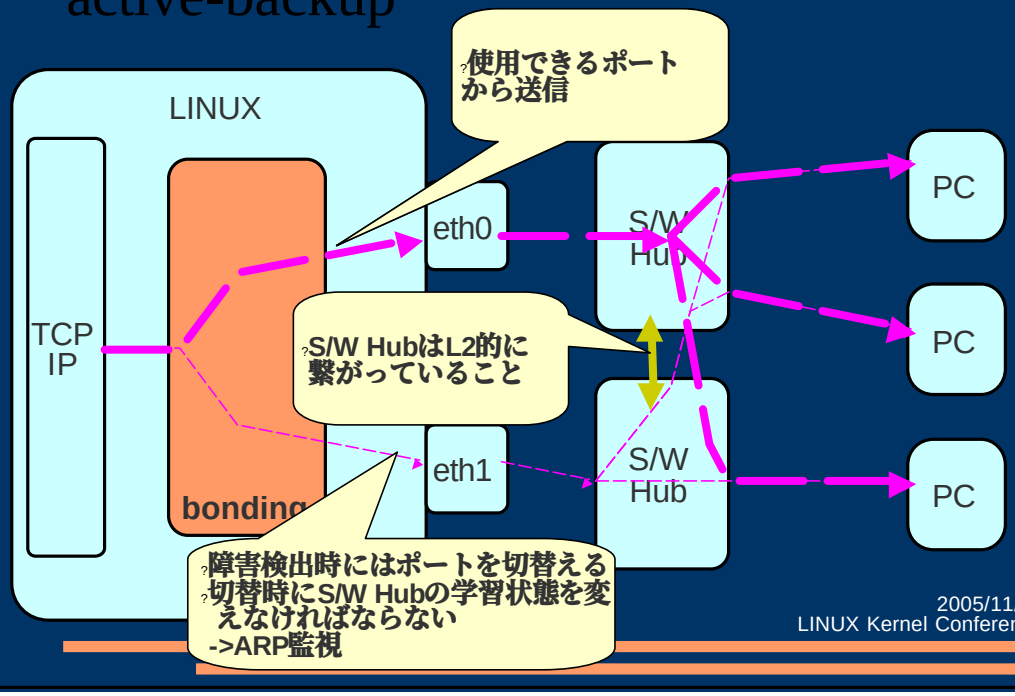
balance-rr



active-backup

- プライマリーに設定したポートで送受信
  - 耐障害性向上を重視するケース
  - どれか一本のEthernetの速度が速いケース
- 通信上の特徴
  - 通常の運用にはプライマリーポートだけで送受信するので速度を上げる様なメリット無し
  - 重複受信問題
- S/W Hubの機能
  - S/W Hubであれば良い
  - 複数のS/W Hubに跨って接続可能
    - S/W Hub自体の障害でも自動的なポート切替機能が動作
    - S/W HubはL2(MAC層)的に繋がっていること

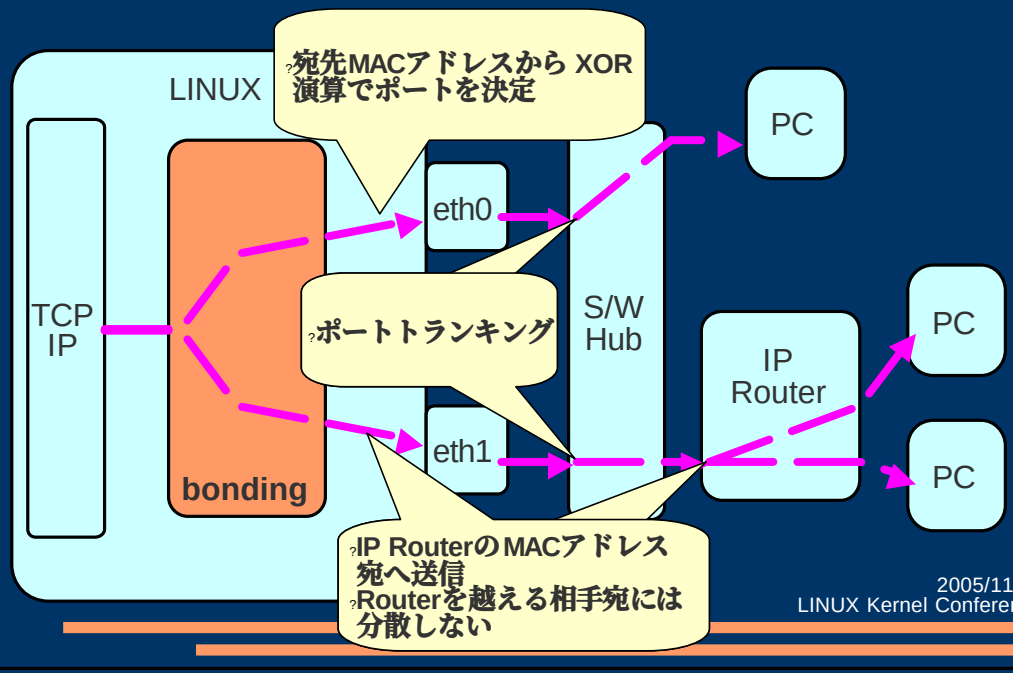
active-backup



balance-xor

- 宛先MACアドレスから使うポートをXOR演算で決める
  - 直結した(ルーターを介さない)ネットワークに多数の通信相手を接続したケース
- 通信上の特徴
  - 宛先MACアドレスをキーにしてポートを決める為、一つの通信相手とは1ポート分の性能
  - 使用するポートに偏り
- S/W Hubの機能
  - ポートトランキング

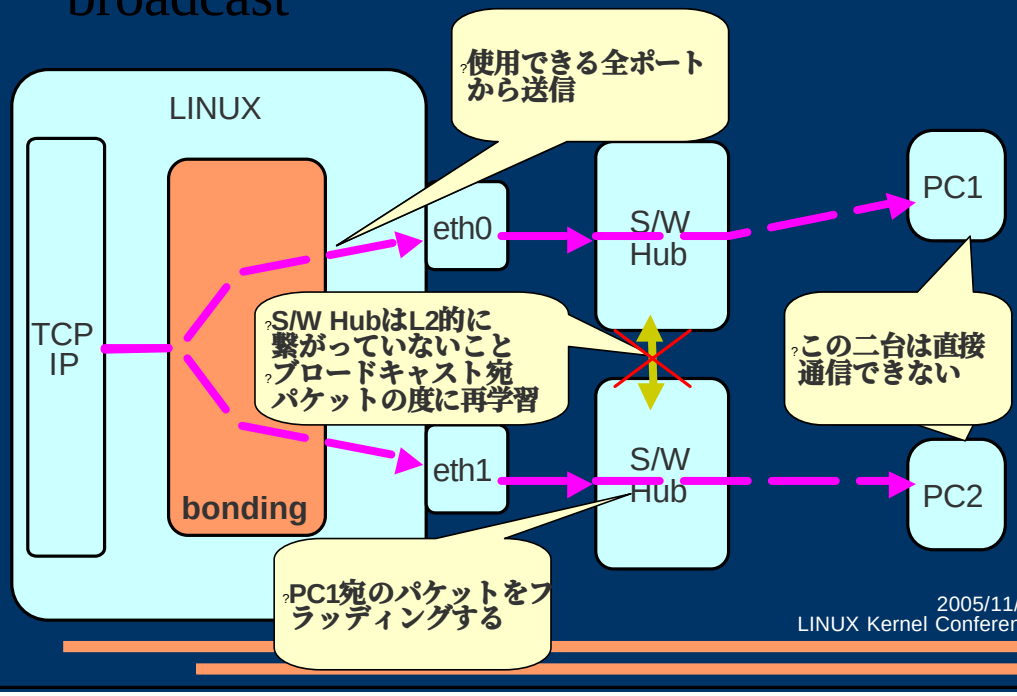
balance-xor



broadcast

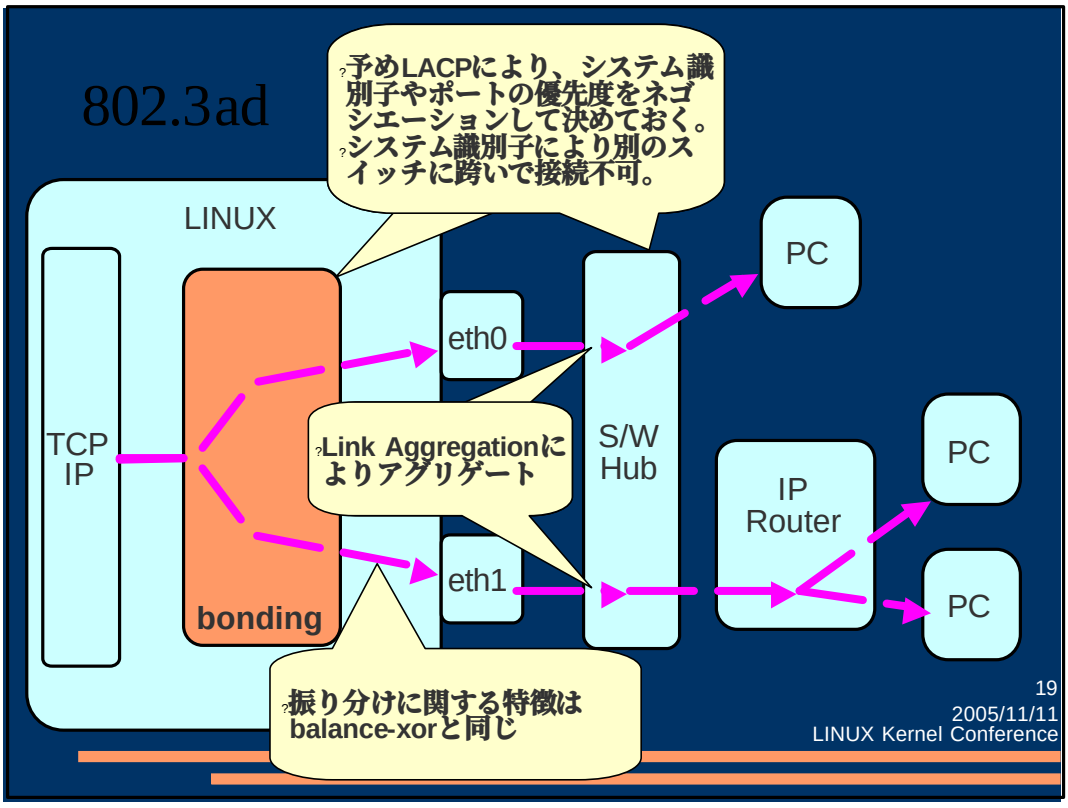
- パケットをコピーしながら全ポートへ送信
  - 複数のスイッチングハブを接続して、且つその複数間の接続の無い、特別ケース
- 通信上の特徴
  - 各NICが繋がったセグメント間の通信不可(複数台のS/W Hubの場合)
  - ブロードキャストストーム
  - NIC数分の同じパケットが通信相手へ届く(1台のS/W Hubの場合)
- S/W Hubの機能
  - 一台のS/W Hubへこのモードで接続する場合にはポートトランキング機能が必要

broadcast



802.3ad

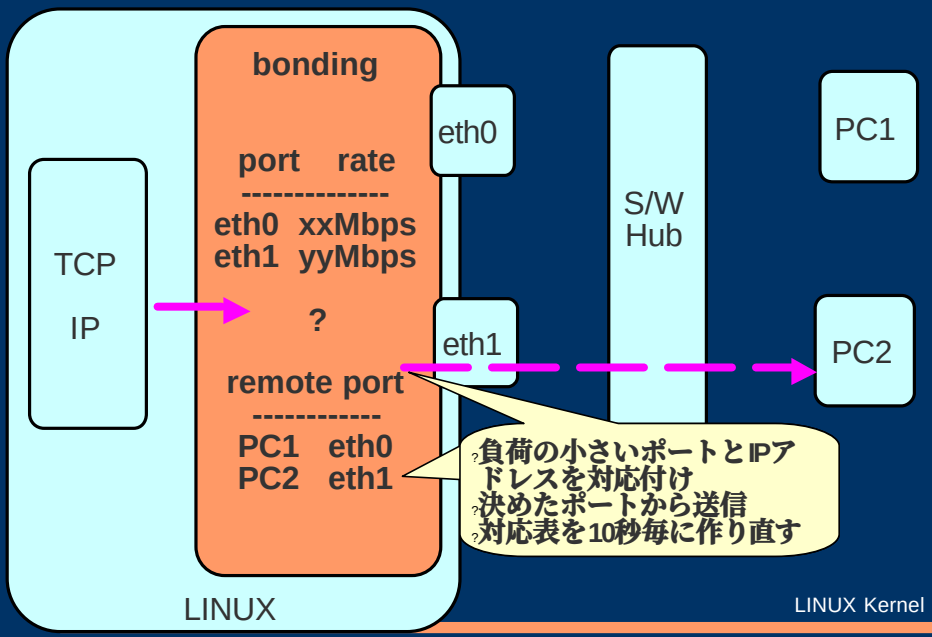
- IEEE802.3ad標準のプロトコル
  - 標準に合わせたシステムを組み上げるケース
  - 直結した(ルーターを介さない)ネットワークに多数の通信相手を接続したケース
- 通信上の特徴
  - Link Aggregation Control Protocol(LACP)
    - システム識別子やポート数、優先度、リンクの状態等を交換し、使用するポートを決める
    - システム識別子の異なる複数のS/W Hubへ跨げない
- S/W Hubの機能
  - IEEE802.3ad Link Aggregation機能(Dynamic Link Aggregationとも言う)



balance-tlb

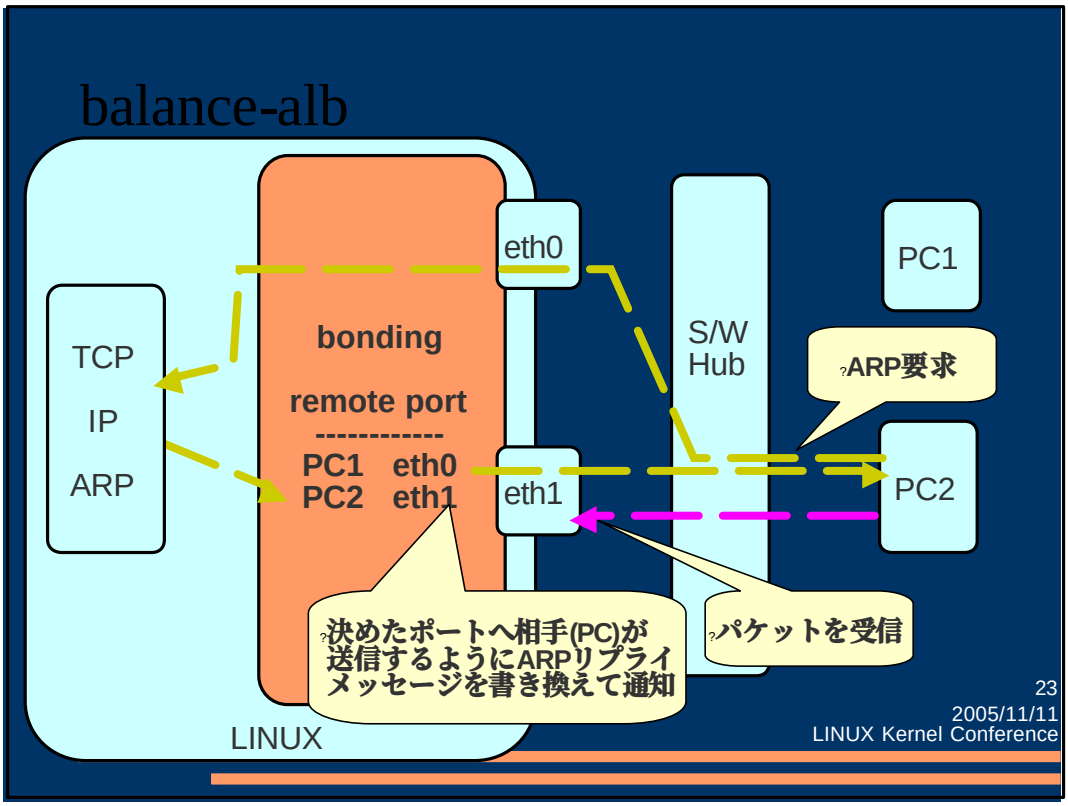
- ポートの送信負荷状態をみて、送信ポート切替
  - 宛先IPアドレスと送信ポート対応表を作成
  - 10秒毎にポートの負荷状態(byte単位の送信速度)を見て再作成
  - IPv4で運用しているケース
- 通信上の特徴
  - IPアドレスで振り分けるので、IPルーターを越えた通信相手にも負荷分散
  - 送信と受信では別の表を使って分散するので、行き帰りの経路が違うため、フラッディングすることがある
- S/W Hubの機能
  - S/W Hubであれば良い

balance-tlb



balance-alb

- TLBの機能に加え、受信ポートの状態をみて、受信ポート切替
  - ARP要求を受信した時に受信ポートの状態を見て、ARPメッセージの自MACアドレスを書き換えて応答
  - IPv4で運用しているケース
- 通信上の特徴
  - 送信負荷分散はTLBと同じ
  - ARPにより受信する(相手が送りつける)ポートを指示する為、LINUXからみて都合の良い分散
  - 障害検出時の受信ポート切替の為、切替える全マシン宛てにARPメッセージを送信する
  - IPアドレスとMACアドレスが一対一に対応しない為、ネットワーク管理システムが重複IPアドレスのアラーム
- S/W Hubの機能
  - S/W Hubであれば良い



監視モード

- ARP監視
  - 設定時に指定したIPアドレスのマシンへARP要求を送信、このポートからパケットを受信すれば、使えるポートと判断
  - ARP要求のブロードキャストにより、S/W Hubの学習状態を保てるため、出来ればこのモードを使用
  - 指定したマシンが停止していると誤判断するかもしれないので、直結したマシンや24時間運転のマシンを選ぶ
  - balance-rr, -xor, active-backup, broadcastで使用可能
- MII監視
  - NICのH/W的なリンクの状態を見て使えるポートと判断
  - リンクアップの状態を保ったまま、NICが故障したりS/W Hubがこけると誤判断
  - パケットのやりとりが無いのでネットワークに負荷をかけない

お薦めのモード

- active-backup
  - 通信負荷分散より耐障害性を重視するケース
  - 複数実装したNICの うちどれか一つが速いケース
- balance-alb
  - IP運用しかしない条件で、耐障害性と負荷分散を実現したいケース
- balance-xor / 802.3ad
  - bondingしたマシンと同じネットワークへ通信相手が多数繋がったケース
  - S/W Hubが必要な機能を持っているケース

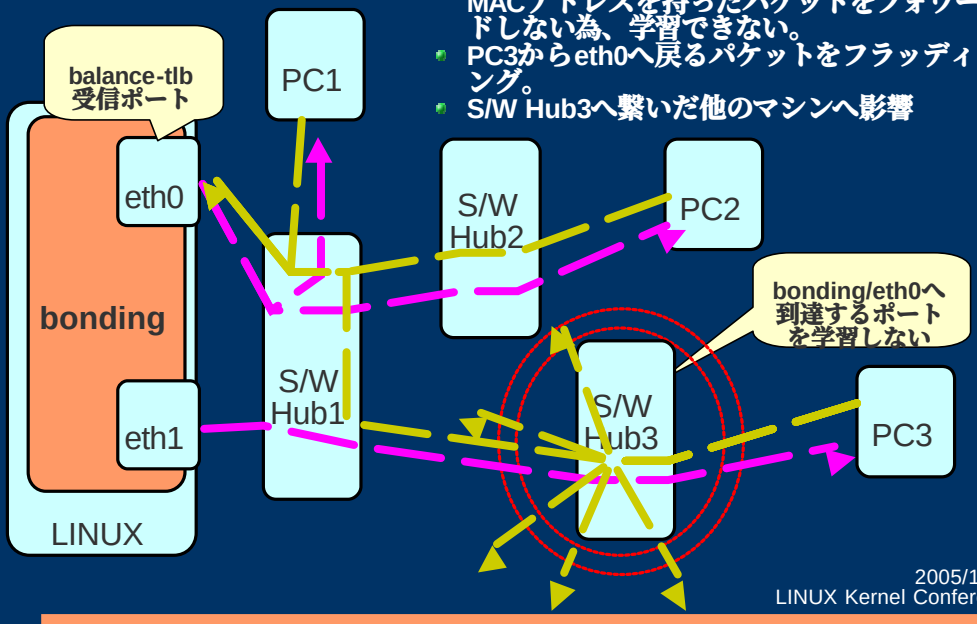
導入時の注意点と設定方法

注意点1 MACアドレス学習

- S/W HubでのMACアドレス学習
  - S/W HubはHubのどのポートの先にどういうMACアドレスのマシンがいるかを学習してフォワード先を決める
  - 未学習時に転送性能の出ないS/W Hubもある
- 学習状態に保つ方法
  - ARP監視を使用
  - balance-tlb, -albでは通信している相手マシンの台数が増えて、全S/W Hubへ全ポートからのパケットが届けばよい

フラッディング

- S/W Hub(下図のS/W Hub3)では、eth0のMACアドレスを持ったパケットをフォワードしない為、学習できない。
- PC3からeth0へ戻るパケットをフラッディング。
- S/W Hub3へ繋いだ他のマシンへ影響



注意点2 IPルータの通過、ネットワーク監視

- IPルータを通過
  - balance-xor, 802.3adモードは適さない
    - MACアドレスをキーにポートを決定
    - IPルータを越えた先にマシンが複数あっても、それらの間での送信負荷分散出来ない
- 一つのIPアドレスに複数MACアドレス
  - balance-tlb, -albモードでは、一つのIPアドレスに対して複数のMACアドレスを割り当て。
    - ネットワーク管理システムで二重IPアドレス使用の異常を検知
    - 管理対象外に設定

設定コマンド1

- Bondingの動作モード、オプション

```
modprobe bonding -obond0 mode=active-backup
¥ arp_interval=60 arp_ip_target=192.168.0.1,192.168.0.3
```

  - Bondingドライバーをロード
  - 疑似的なポートの名前をbond0に設定
  - active-backupモードを使用
  - ARP監視を使用し、60ms間隔で検査
  - ARP監視に用いる相手を二つIPアドレスで指定
- 疑似的なポートと物理ポートの関連付け

```
ifenslave bond0 eth0
ifenslave bond0 eth1
```

  - Bond0とeth0, eth1を関連付ける

設定コマンド2

- 疑似的なポートの設定

  ifconfig bond0 192.168.0.10 up

  - 物理的なポートに設定するのと同じ方法

- 参照
  http://prdownloads.sourceforge.net/bonding/bonding.txt?download

設定解除コマンド

- 擬似的なポートの使用終了

  ifconfig bond0 down
  ifenslave -d bond0 eth0
  ifenslave -d bond0 eth1
  rmmod bond0

  - 擬似的なポートの状態をDOWN
  - 擬似的なポートと物理的なポートの関係を解除
  - Bondingドライバーを削除

- 参照
  http://prdownloads.sourceforge.net/bonding/bonding.txt?download

課題

課題1 --動的な設定変更--

- 設定変更時に全bondingを停止
  - 物理ポートを擬似的なポートと関連付けるまたは解除する以外の変更は、動作中に不可
    - ARP監視のターゲットを変更する
    - モードを変える
  - ネットワーク側の変更の影響でサービスを停止しなければならない
- Sysfsを利用した設定方法変更
  - /sys/class/net/bond0/...に値を書くことでモードやオプションを変更可能に

課題2 --重複受信--

- 相手が送信したパケットをコピーして受信
  - 複数の物理ポートが一つのネットワークに繋がっているので、ネットワークに流れた一つのパケットを全物理ポートから拾いあげ、コピーしたように見える
    - active-backupモードで運用した場合に、接続したS/W HubのMACアドレスの学習状態に依存し、S/W Hubがフラッディングすると、重複受信に繋がる
    - active-backup, balance-tlb, -albモードでは、Broadcast / Multicastパケットも重複受信
  - Repeaterを使用不可
  - どこかで正常に動かないケースが出るかも
- 修正提案
  - コミュニティへ修正パッチを提案
    - Jay Vosburgh氏より取込みたい旨リプライ頂き、メールをやりとり中

課題3 --分散アルゴリズム--

- 送信負荷分散
  - 宛先MACアドレスのXORにより送信ポートを決めるだけでは、負荷が分散せず偏る。
  - IPルータを越えた通信は、通過するIPルータのMACアドレスで送信ポートが決まってしまう
- 分散に用いるパラメータを増やす
  - MACアドレス
  - IPアドレス、G/WのIPアドレス
  - TCPやUDPのポート番号、プロトコル番号
  - 分散ポリシー化

他技術との組み合わせ

- Xenでの利用
  - Xenが動くマシンの全NICをbondingすることだが、これは現在でも可能。
  - Xen0とゲストOS間を接続するbridgeの下へ潜らせる様に設定すれば使用出来る。
  - http://thread.gmane.org/gmane.comp.emulators.xen.user/3499
- TOEとの組合せ
  - TCP Offload Engineを実装したNICを用いた場合にbondingの入る余地が無い
  - bondingを適用出来るか、するならどういう方法があるか。

最後に

最後に

- ネットワークは不可欠
  - bondingするとネットワークの事まで手を出さなければなら無いのかと、感じられたかもしれません。
  - でも、bondingしなくてもネットワークの事は必要なのです。
- せっかくだから最高の環境で
  - bondingを使うと、マシンの全NICを耐障害性と負荷分散に用い、最高のネットワーク環境を構築出来ます。

皆様と一緒により素晴らしいLINUXネットワークを構築しましょう。

参考

- Bonding開発
  - mailto:bonding-devel@lists.sourceforge.net
  - http://sourceforge.net/projects/bonding/
- Bonding ドキュメント
  - http://prdownloads.sourceforge.net/bonding/bonding.txt?download
- XenでのBonding設定ノウハウ
  - http://thread.gmane.org/gmane.comp.emulators.xen.user/3499
- 開発者
  - Thomas Davis
  - Willy Tarreau
  - Constantine Gavrilov
  - Chad N. Tindel
  - Janice Girouard
  - Jay Vosburgh
  - その他大勢