第 3 章

ExpEther

ExpEther[7] は NEC[6] によって開発されたシステムバス仮想化技術である．ExpEther では現状の多くのコンピュータシステムのバスとして利用されている PCI Express を Ethernet に仮想化する．

ExpEther ではシステムバス仮想化において以下の要件を求めた．

- End-to-End のコミュニケーションの低遅延
- リンクのバンド幅のスケーラビリティ
- システムに多数のデバイス接続を実現する高拡張性
- システムバス障害回復・冗長構成対応可能な高信頼性
- 既存資源変更不要でバス管理が容易な運用容易性
- 単一バックプレーンで構成可能なシステム統合
- In-Service で構成変更可能な柔軟性

ExpEther では Ethernet というバックプレーンへのシステムバスの拡張によって，一般的な Ethernet 機器を利用した高拡張性を実現する．また独自の再送・輻輳制御による低遅延化とネットワークの冗長構成のサポート，PCIe 規格のホットプラグ・ホットリムーブのサポートにより要件を満たす．

3.1 概要

図 3.1 に ExpEther を用いたシステム構築例を示す．2 台のサーバに対して 3 つの PCIe エンドポイントが割り当てられている．3 つの PCIe エンドポイントには I/O カードやグラフィックボードなど PCIe の規格に対応したデバイスが置かれる．サーバとデバイスは ExpEther の NIC の仲介を経て Ethernet 越しのコミュニケーションが可能になる．サーバと PCIe エンドポイントを相互結合する Ethernet には既存の Ethernet 機器を使用して構成することが出来る．

サーバへの PCIe エンドポイントの割当は ExpEther NIC が持つ Group ID と呼ばれる値によって決定される．この値の変更によってサーバへのデバイスの割当を容易に変更することが可能となっている．

ExpEther は PCIe ファブリックに Ethernet 上へと拡張することによって接続するハードウェアのスケーラビリティを向上させ冗長構成もまた可能にする．利用されていたデバイスがダウンした場合，Ethernet に接続されてスタンバイしていたモジュールを使用しサービスを継続すること

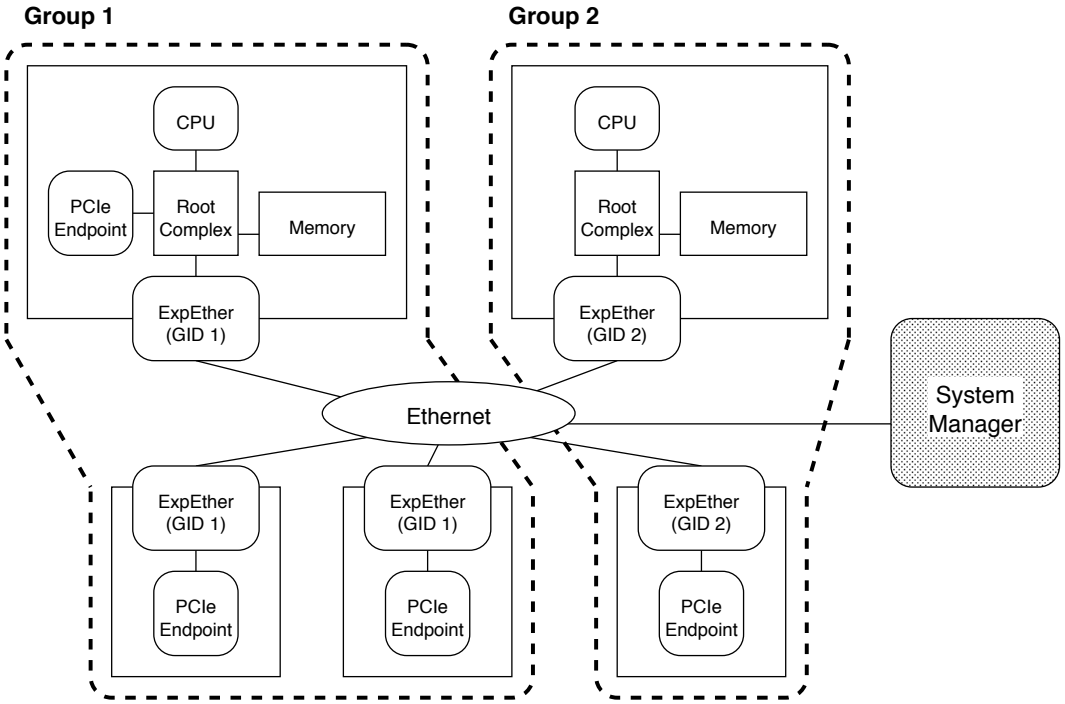

図 3.1: ExpEther を用いたシステム構築例

が出来る．またハードウェアが不足した場合は新たなリソースを Ethernet に接続することによってシステムを拡張することが出来る．

ExpEther の重要な利点としてデバイスを異なるサーバ間で共有できることが挙げられる．サービスが必要としたときだけ，必要な分の I/O をサーバに割り当てることが出来る．これによりシステムにおける I/O リソースの総量を低減し，また I/O リソースの利用率を挙げることができる．In-Service の I/O リソースの交換を可能にする．

3.2 機能

ExpEther がシステムを実現するために提供する機能について示す．

3.2.1 PCI Express over Ethernet

ExpEther によって実現されるシステムにおいて，サーバとデバイス間で行われる PCIe 規格の通信を Ethernet を経由して行うため，PCIe で交換されるデータを Ethernet 上で交換できる形式に変換する必要がある．この機能は PEB (PCI Express-to-Ethernet Bridge) によって実現される．

また既存の OS やデバイスドライバといったソフトウェア資源について変更を行うことなくシステムを構築するために，サーバにはシステムの Ethernet 部が PCIe ファブリックの一部として仮想的に PCI Express スイッチとして認識されるようにする．

PEB

PCI Express でやりとりされるデータを Ethernet を介してやり取りするため，ExpEther NIC では PEB (PCI Express-to-Ethernet Bridge) という機能をもつ．PEB は PCI Express バスと Ethernet の間

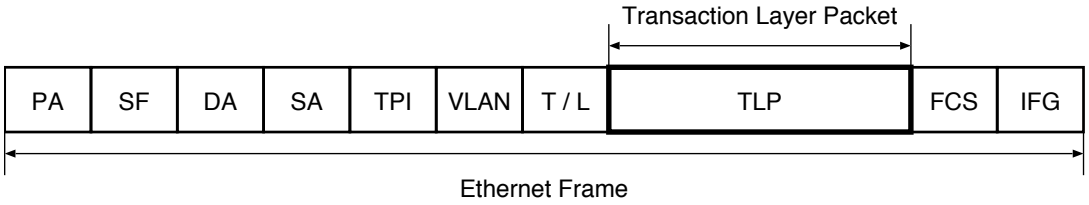

図 3.2: PEB によるカプセリング

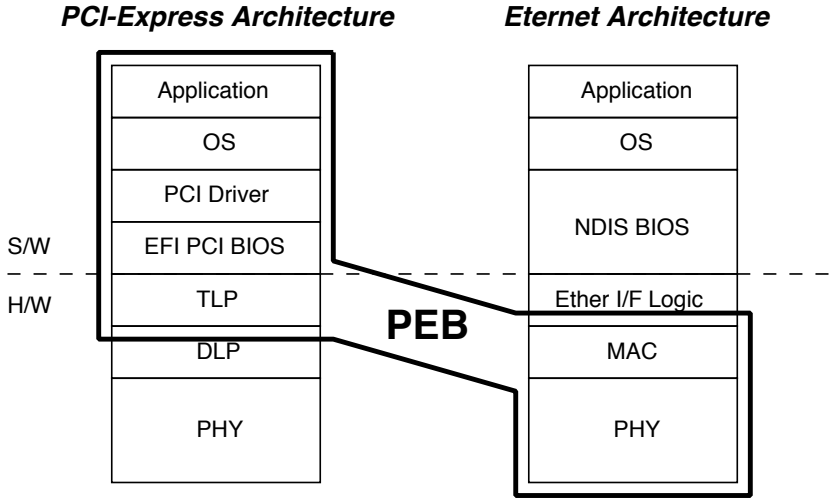

図 3.3: PCI Express と Ethernet のブリッジ

でブリッジの機能を果たす．ブリッジでは PCIe のパケットを Ethernet フレームにカプセリング，またデカプセルの処理を行う．

PCI Express において end-to-end の通信制御の仕様を定めるトランザクションレイヤーにおいて TLP (Transaction Layer Packet) がやり取りされるが，PEB ではこの TLP に対して図 3.2 のような形で Ethernet フレームへのカプセリングを行う．このように PEB は PCI Express の規格におけるトランザクションレイヤーと Ethernet における MAC 層をブリッジする形で実現される．そのため ExpEther のシステムにおいて既存のレイヤ 2 までの Ethernet 機器を使った拡張が可能である．

仮想 PCI Express スイッチ over Ethernet

サーバで認識される PCIe ファブリックにおいて PEB を含む Ethernet 部は 1 つの PCIe スイッチとして扱われる．これは PEB があたかも PCIe スイッチにおける PCI-to-PCI ブリッジであるかのように振るまうことで実現される．図 3.4 と図 3.5 は実際の PCIe スイッチと ExpEther における Ethernet 部の扱いを示したものである．

Ethernet が PCIe スイッチとして見られることによって ExpEther を用いた構成はサーバから従来通りの PCIe のトポロジとして認識される．そのため ExpEther は既存のシステムに対して変更を加えることなく導入が可能である．デバイスドライバを新規に導入することなく既存の PCI ドライバや OS などにより利用が可能であり，ソフトウェアへの変更は不要であるまたサーバあたりに接続可能なデバイスの上限についても，TLP においてデバイスを指定するフィールドとして与えられる 8 bit で表しうる ID の数の 256 までの接続を許す．

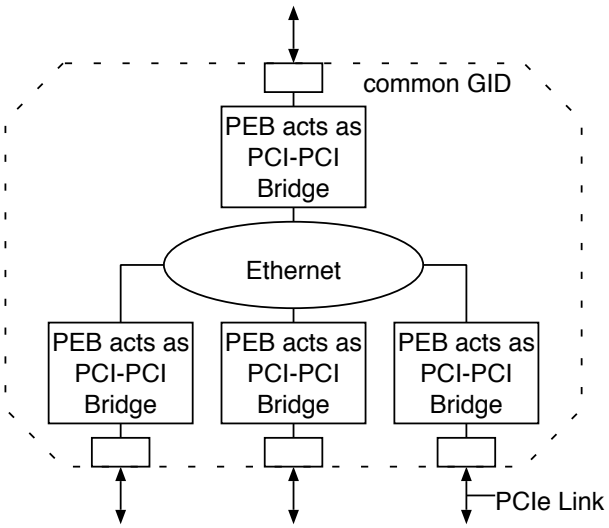

図 3.4: ExpEther における Ethernet

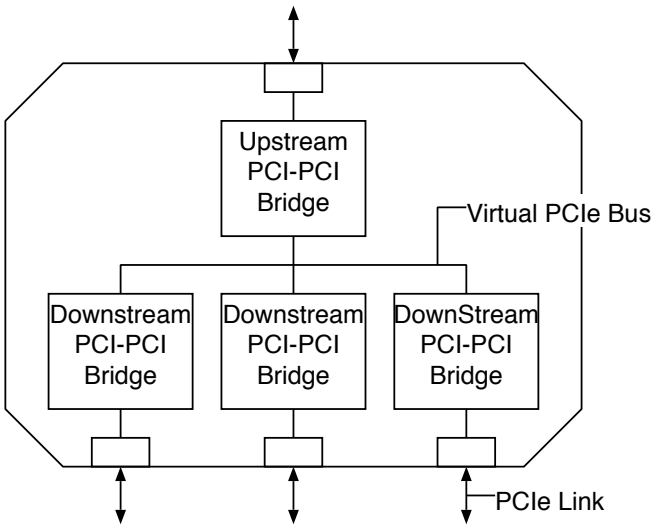

図 3.5: PCI Express スイッチ

また Ethernet 部全体を PCIe スイッチとして振るまわせることは再送タイムアウト時間の短いバス信号を終端させる意味を持つ．

3.2.2 PCIe ファブリックの分離

ExpEther によって実現されるシステムでは 1 つ Ethernet のネットワーク上に複数のサーバの PCIe ファブリックの混在を許容する．このとき GID という値によって混在する PCIe ファブリックの判別を行うことを可能にし，同時に GID の値によってサーバへのデバイスの割当の容易な変更を可能にする．

Group ID

サーバへの デバイスの割当は ExpEther NIC に保持される GID (Group ID) によって定められる．この値は NIC で設定をするか Ethernet 上のシステムマネージャから変更を行うことが出来，GID の値によってサーバへの Endpoint の割当を容易に変更することが可能で，柔軟に PCIe ファブリックの構成を変更することが出来る．

サーバとデバイス間で通信を行う場合は自らの GID の値を Ethernet フレームは IEEE 802.1D VLAN タグ部に記述し転送を行う．これによって交換されるフレームがどのサーバの PCIe ファブリックに所属するものなのかの判別が可能となる．

3.2.3 In-Service での構成変更

ExpEther によって実現されるシステムではサービス起動中のデバイスのホットプラグとホットリムーブを PCIe の規格に準拠した形で可能にする．この時にデバイスの状態を判断するために ExpEther の NIC 間で Device Information Packet がやり取りされる．

Device Information Packet

ExpEther の NIC は Ethernet フレームの形態をとって PEB や接続されるデバイスの情報，GID を付与した Device Information Frame を Ethernet へと定期的ブロードキャストする．受け取ったフレームから NIC は Ethernet 越しに接続される PEB の ID やデバイスの種類といった情報を得ることが出来る．

ホットプラグ・ホットリムーブ

定期的にブロードキャストされる Device Information Frame を利用してホットプラグ及びホットリムーブの機能が実現される．この機能では PCIe の規格に対応した形式がとられており，以下の手順で行われる．

1. 新たに割り当てたいデバイスについて ExpEther NIC にシステムマネージャなどから GID を変更する
2. GID が変更された ExpEther NIC から Device Information Frame がブロードキャストされる
3. 新たなデバイスから受けた Device Information Frame がトリガーとなりサーバ側の NIC が OS に対して割り込みをかける
4. 割り込みにより標準の PCIe 規格に乗っ取った PCIe ホットプラグイベントが実行される

またホットリムーブはデバイス側からの定期的に送られてくる Device Information Frame が途切れることがトリガーとなり，割り込み及びイベントが発生する．

3.2.4 通信の信頼性の保証

PCIe では link-by-link の再送制御とフロー制御を行い，OS に対して通信の信頼性を保証する．ExpEther においても通信の信頼性を維持するために ExpEther NIC 間でも再送制御とフロー制御を行う必要があり，そのために EFE (Ether-Forwarding-Engine) と呼ばれる独自の方式を用いている [21].

Ether-Forwarding-Engine

ExpEther の Ethernet 部の再送制御・輻輳制御である EFE では以下の要件を満たす必要がある．

通信の信頼性の保証**:** PCIe で保証される通信の信頼性を維持する必要がある

広帯域・低レイテンシ**:** サーバとデバイスの通信を低レイテンシかつ広帯域で実現する必要がある

End-to-End での制御方式**:** Ethernet 上のスイッチに特別な機能を要求しないために End-to-End での機能の実現が求められる

ハードウェアコストの低減**:** レイヤ 2 のハードウェアでの機能の実現が求められるため

EFE の輻輳制御に関して一般的な TCP ではロスベース方式を用いているのに対し EFE では遅延ベースの方式がとられている．

遅延ベース方式の輻輳制御では RTT (Round Trip Time) に基づいた送信帯域の決定を行っている．通信の開始時には一定数のプローブパケットを転送しその RTT に基づいて初期転送レートを決定する．その後は転送中のパケットの RTT について，大きくなった場合にはネットワークの混雑度が増していると判断し転送レートを下げ，小さくなった場合にはネットワークの混雑度が低下したと判断し転送レートを上げるといったような，変化に応じた通信帯域の変更を行う．遅延ベース方式の輻輳制御はリンク利用率が低い状況でロスベース方式と比較して早い転送レートの上昇が可能である．しかしながらロスベース方式と競合した場合にスループットが低下するという問題が生じる．

ExpEther ではシステムを構成する Ethernet は LAN であり，異なる輻輳制御方式が混在することはないと考えられ，素早く転送レートを変更し帯域を有効に活用できる遅延ベース方式の輻輳制御を行う．

EFE の再送制御の方式に関しては Go-Back-N を使用する．他に Selective-Repeat の再送制御方式が考えられるが，ハードウェアコストの観点ではより単純な方式である Go-Back-N の方が有利がある．しかしながら再送パケットの数の点では Selective-Repeat の方が有利であり，ハードウェアコストと性能のトレードオフとなっている．しかし ExpEther における通信はは LAN 内で収まるものであり往復伝播遅延はあまり大きくならない環境が想定され再送パケットが大きく増加することは考えにくい．そこで再送パケットの数の不利による影響は小さいと考えられ，Go-Back-N が再送制御方式として選択されている．

またこの再送制御では Ethernet のオートコンフィグレーションの機能を併用し，システムの Ethernet スイッチを冗長に用意することによってネットワークの冗長構成を可能にする．