

通信ネットワークノードへの  
Linux適用拡大  
(キャリアグレードLinux)

2003年10月30日

日本電気株式会社

モバイルIPネットワーク事業部

エグゼクティブエキスパート 中山 三郎

# 目次

1. 通信システムの要件と事業環境
2. 新プラットフォームの選定
3. CGPFの特徴
4. PICMG概要
5. キャリアグレードLinux概要
6. SAF概要
7. ATCA適用領域
8. CGPF採用システムアーキテクチャー
9. NECの取り組み

# 商標表示

- ◆ PICMG、AdvancedTCAおよび ATCAは、PCI Industrial Computers Manufacturers Groupの商標です。
- ◆ Linux は、Linus Torvalds の米国およびその他の国における登録商標 あるいは商標です。
- ◆ OSDL は、Open Source Development Labs, Inc.の商標です。
- ◆ Service Availability は、Service Availability Forumの商標です。
- ◆ Javaは、米国 Sun Microsystem, Inc. の商標です。
- ◆ CORBAは、オブジェクト・マネジメント・グループの登録商標です。
- ◆ その他会社名およびロゴ、製品名などは該当する各社の商標または登録商標です。

# 1.通信システムの要件と事業環境

## ◆ 通信システムの要件とは

- サービス品質(基準)に関する挙動  
(完了率、接続(応答)遅延、等のふるまい)
- 性能(加わる負荷)に関する挙動  
(負荷変動対応力、高負荷時の安定性、等のふるまい)
- 無停止運転  
(3分間/年のサービス停止基準確保)

➡ プロプラ技術の集大成システム(高コスト、長開発期間)

## ◆ 通信事業環境の急激な変化

=レガシー・プロプラな局用交換機からオープンなIP製品への世代交替=

- 低価格化、小規模化
- 市場規模予測の困難性
  - スモールスタートからの段階的投資のニーズ
  - TimeToMarketのニーズ

➡ 「早く製品を開発できる環境」、「安く製品をつくれる環境」が重要

=Global Standard技術  
(PCサーバ&UNIX系OS)の活用

## 2. 新プラットフォームの選定

- ◆ オープンな技術を活用しつつ、通信キャリアのニーズを満足出来るプラットフォームの適用が急務
- ◆ PICMG、OSDL、SAFが連携して標準化を進めている、通信システムをターゲットとしたプラットフォームが最適解と判断
  - 通信キャリアの要求をこまめに取り込み
  - ハード、OS共にオープンな標準規格品

以降これをキャリアグレードPF (CGPF)と呼ぶ

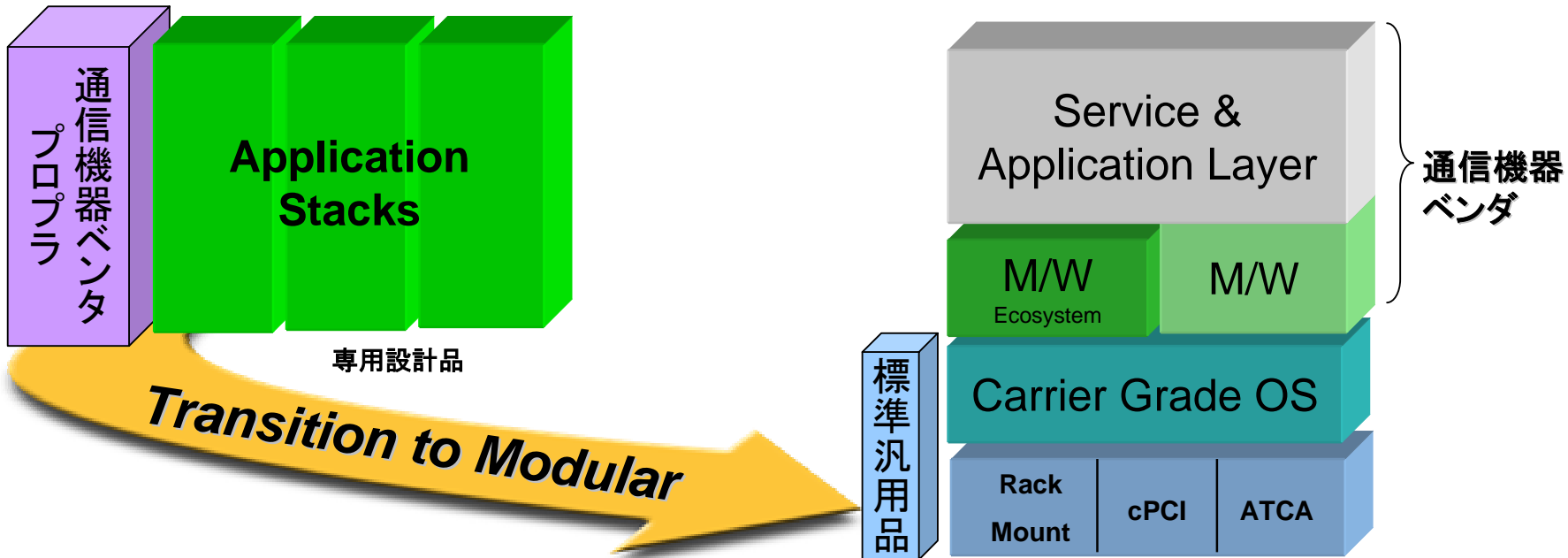
HW・・・PICMG3. X(ATCA)

OS・・・CarrierGradeLinux + SAF準拠ミドルウェア

## 2. CGPFとは

標準規格に基づく、オープンな通信ネットワーク向けプラットフォーム  
Intelはこれを「標準規格ベースのモジュラ型通信サーバー及び通信機器」として発表\*1

高コストパフォーマンス、汎用品による互換性、再利用性を持つHW&SWビルディング  
ブロックであり、開発コストの削減、早期開発を可能とするもの



\*1 2003年10月13日

オープンシステム、オープンソフトウェア、互換性&再  
利用性を持つモジュラー型ビルディングブロック

ITU Telecom World 2003 にて

## 2. CGPF標準化推進組織体制



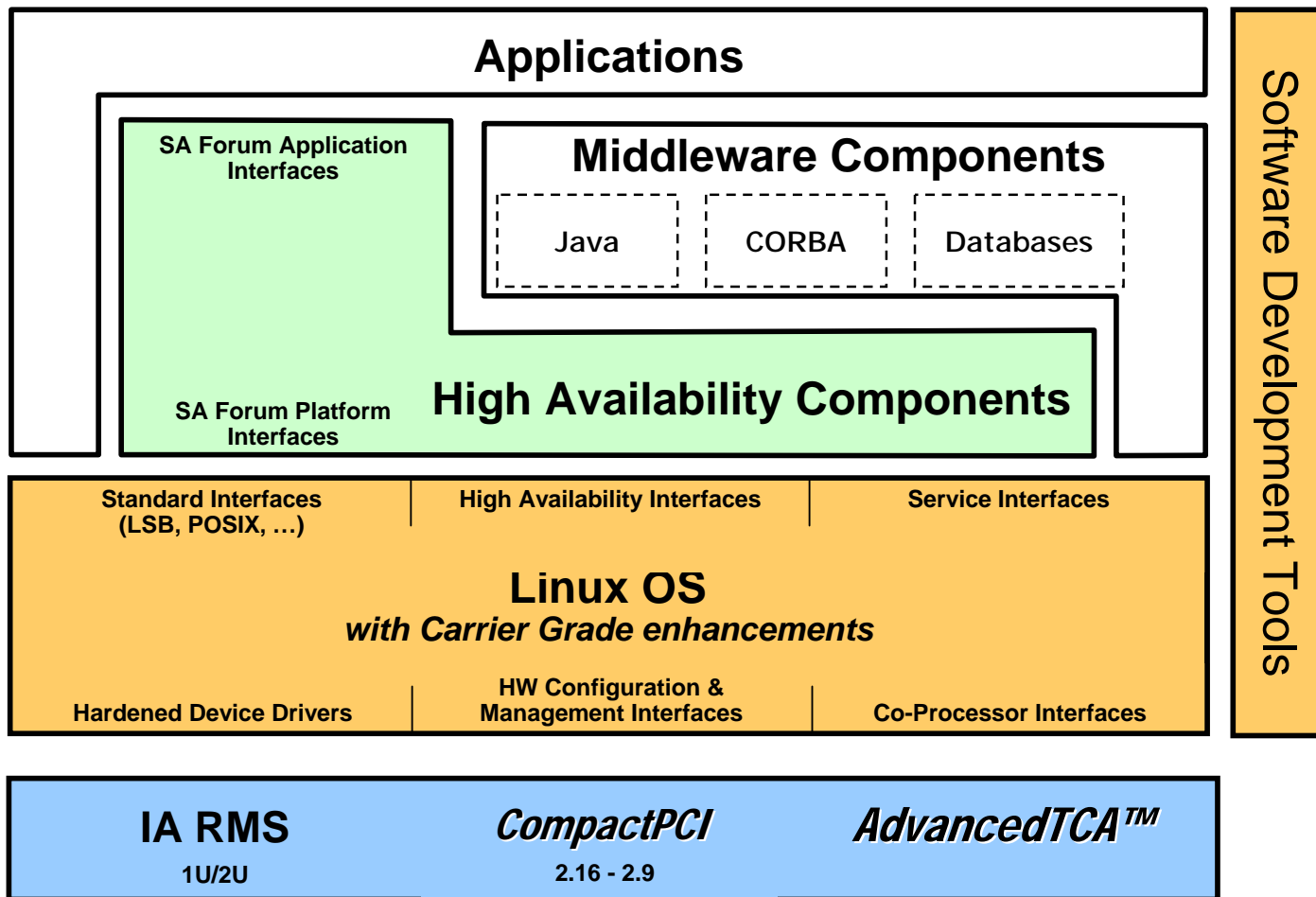
27 companies defining MW API interfaces




29 companies defining OS reqs & architecture



Over 600 companies defining HW reqs & architecture



 Carrier Grade Linux Work Group Scope

## 2. 標準化団体概要

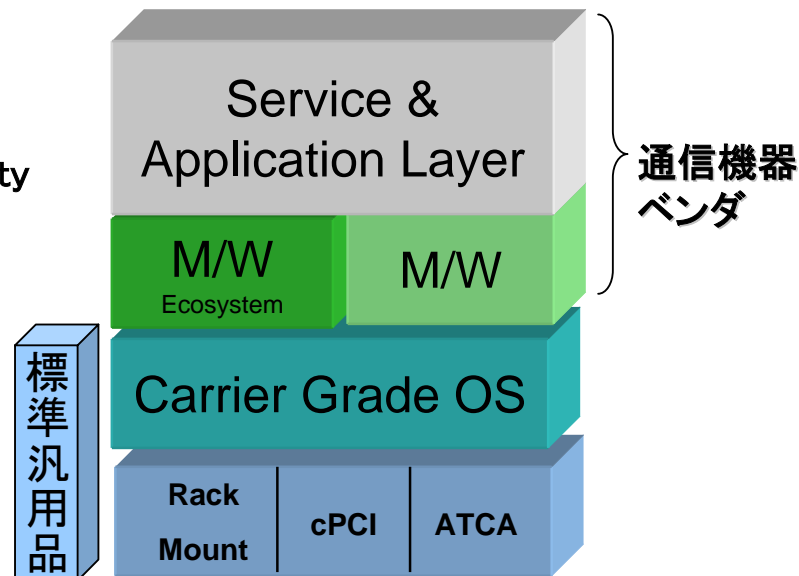
- ◆ **PICMG (PCI Industrial Computer Manufactures Group)**
  - PCIベースの産業用組込ボード、関連製品に適用する規格団体で現在600社以上が加盟
  - Compact PCIボードやシャーシの規格
- ◆ **OSDL (Open Source Development Lab.)**
  - 2000年8月にIntel、IBM、HP、NECの4社が立ち上げ
  - 現在は29社が参加 (Alcatel, Cisco, Dell, Ericsson, Force, Fujitsu, Hitachi, HP, IBM, Intel, Montavista, NEC, Nokia, NTTデータ、RedHat, SuSe, Toshiba, Turbolinuxなど)
  - 2002年1月CGL (キャリアグレードLinux) WGを設立
  - 2002年8月DCL (データセンタLinux) WGを設立
- ◆ **SAF (Service Availability Forum)**
  - ハイアベイラビリティを実現するミドルウェアの標準API、また必要とするプラットフォームのインタフェースを規定する
  - 現在27社が参加 (Fujitsu-Siemense, Goahead, HP, IBM, Intel, Montavista, Motorola, NEC, Nokia, Oracle, Sun, Timestenなど)



### 3. CGPFの特徴(1/2)

#### ■ ATCAの特徴

- ◆ テレコム用途を考慮した世界標準のため保守性良好  
⇒ 全コンポーネントがblade型かつHot Swap対応
- ◆ 世界標準PFの採用で今後のCost, Space, Power, Performance向上のトレンドにのる
- ◆ 約束されたロードマップ(Intelとの協業、日本ではNECが先行して取り組んでいます)  
⇒常に最先端のCPU搭載可能
- ◆ オープンシステム、オープンSW、互換性&再利用性を持つモジュラー型ビルディングブロック  
⇒早期開発が可能  
⇒世代間互換性保証
- ◆ 業界標準に沿ったインプリメントでHigh Availability  
⇒ 豊富な交換機のノウハウを継承



オープンシステム、オープンSW、互換性&再利用性を持つモジュラー型ビルディングブロック

# 3. CGPFの特徴(2/2)

## ■ キャリアグレードLinux の特徴

- ◆ ◎カーネル&ドライバの堅牢化(ハーデニング)
  - ◆ パラメータチェック強化、ハード障害でのメモリ破壊防止、PANIC抑止、イベント通知・ログ収集のAPIサポート、安定した障害処理等の機能強化
- ◆ ◎RAS機能
  - ◆ ホットスワップ、DKミラーリング、クラスタ構成DK管理、DKボリュームのオンライン変更、オンラインファイル更新、冗長NIC、高速BOOT、プロセスの周期監視&リスタート、WatchDogTimer等
- ◆ ◎診断、解析、デバッグ機能
  - ◆ カーネルのダイナミックプローブ、マルチスレッドコアダンプ、カーネルクラッシュダンプ分析ツール、擬似障害発生ツール、カーネルパニックハンドラの拡張、オンライン診断(システム、OS、ハード)
- ◆ その他
  - ◆ ○プリエンティブカーネル、リアルタイムスケジューリング、高精度タイマ
  - ◆ ○ジャーナリングファイルシステム、POSIXイベントログ、リソースモニター
  - ◆ オープンソースであり、確実な保守体制が構築可能

◎: CGL個有機能(無停止運転、安定度向上、障害解析情報収集強化)

○: CGLで機能強化



# 4. PICMG概要

PICMGとは？

=> **PCI** Industrial Computer **M**anufactures **G**roup

PCIベースの産業用組込ボード、関連製品に  
適用する規格団体

**AdvancedTCA™**

PICMG 3.0 Core

PICMG 3.1 GbE

PICMG 3.2 IBA

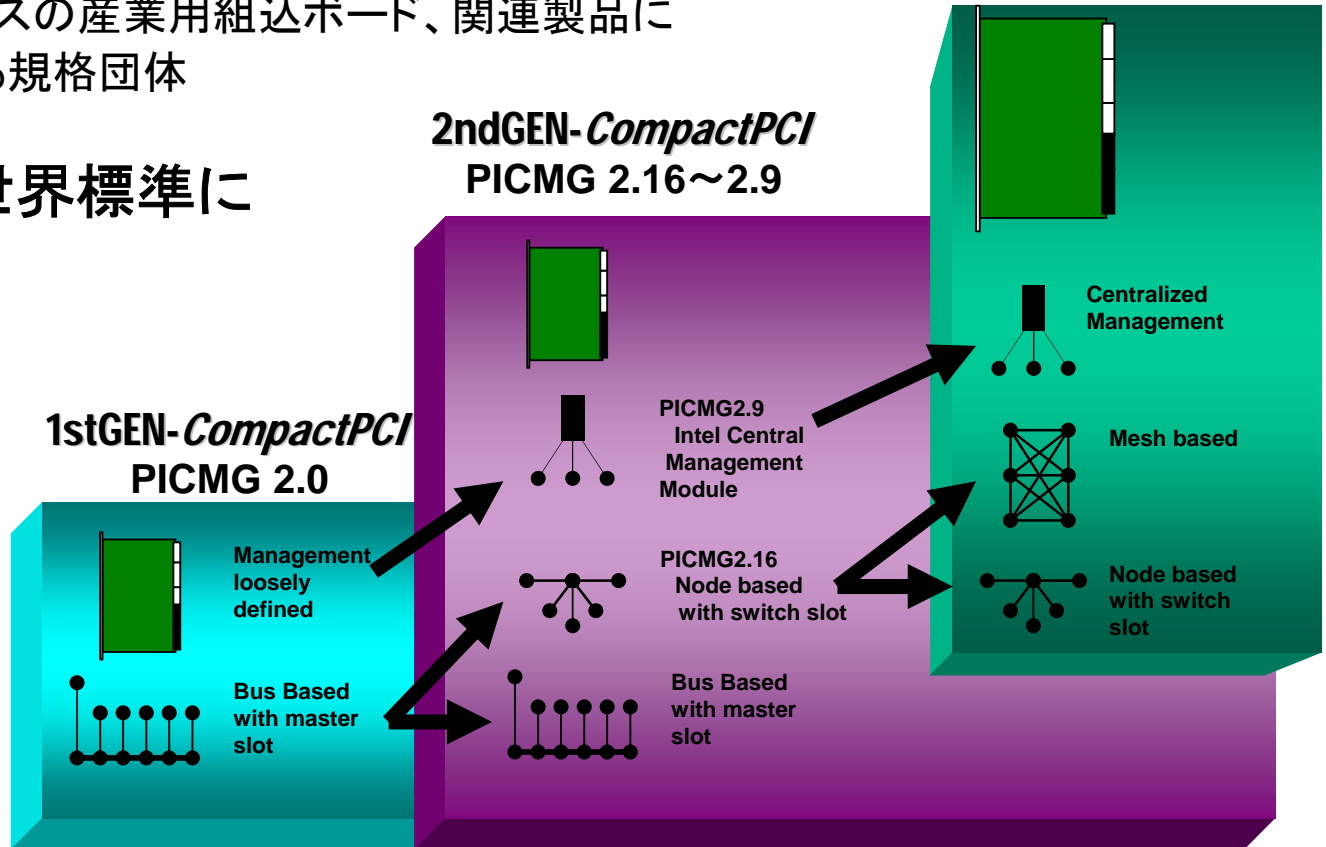
今後の世界標準に

**2ndGEN-CompactPCI**  
PICMG 2.16~2.9

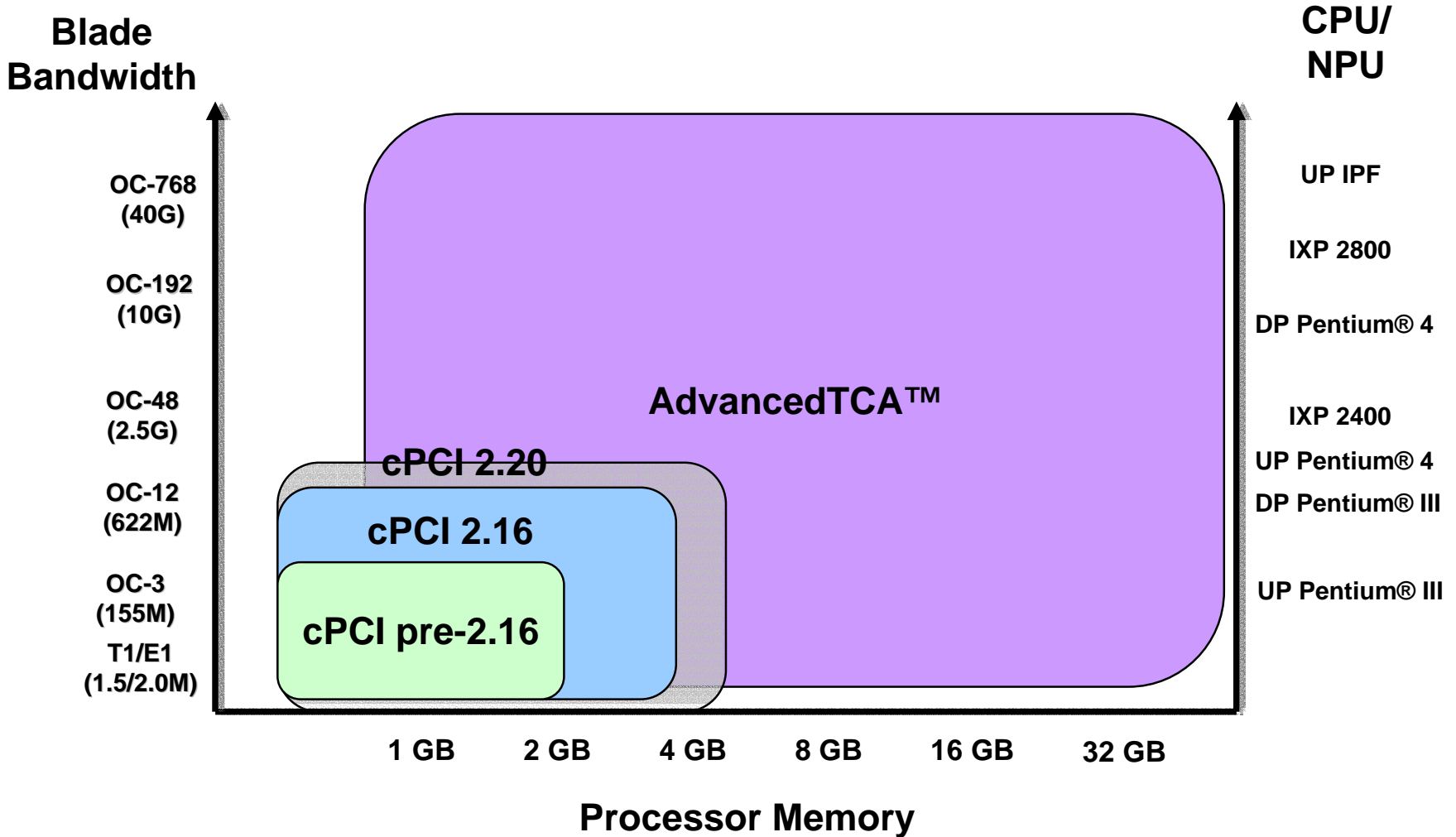
**1stGEN-CompactPCI**  
PICMG 2.0

マネージメント

トポロジ

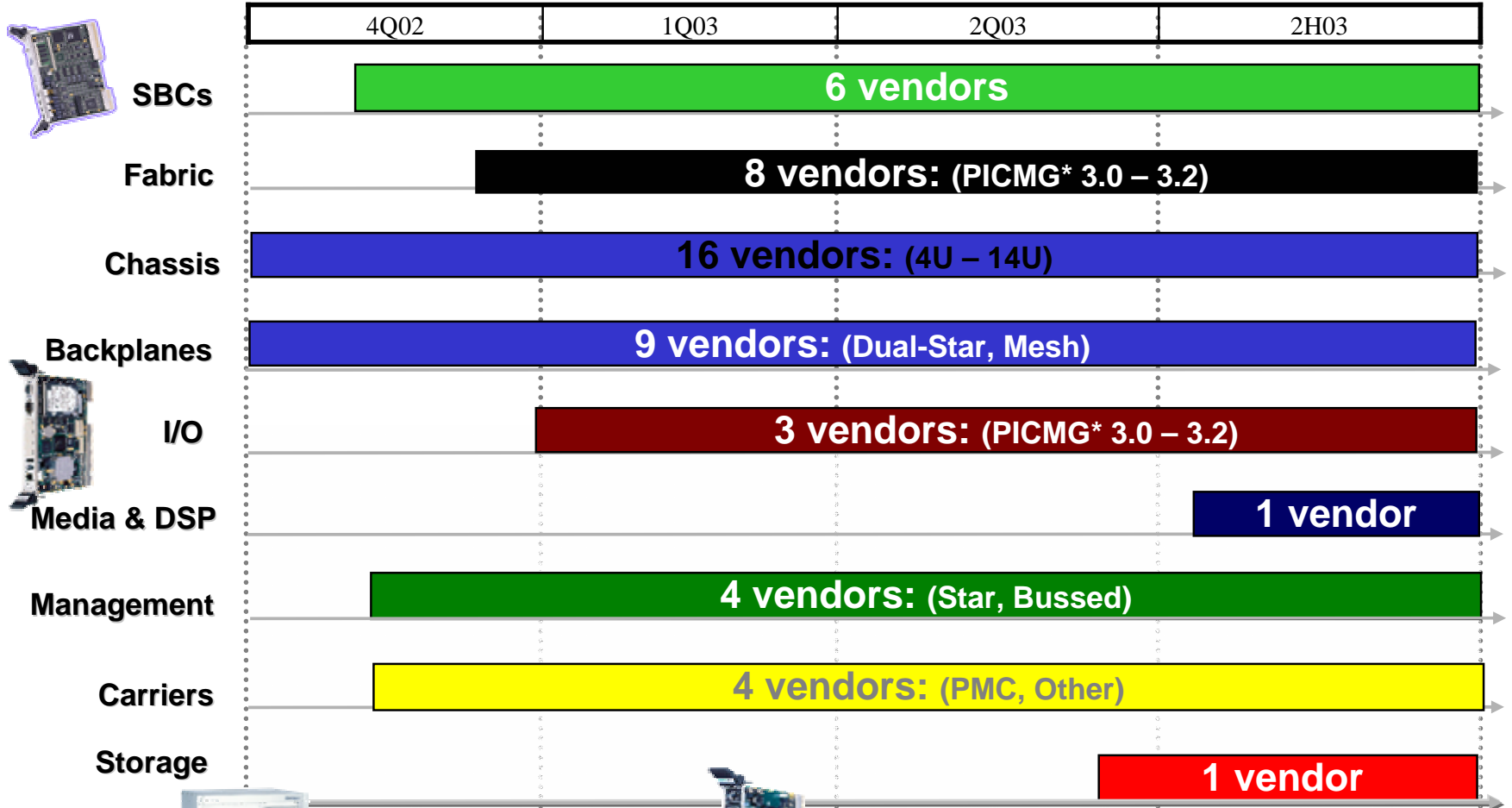


# 4. PICMGボードSpec対比



UP : Uni-Processor  
 DP : Dual Processor

# 4. ATCAロードマップ

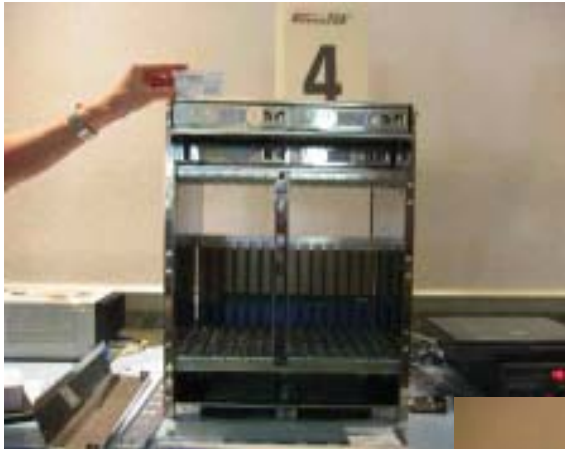


\*Other party brands and marks are the property of their respective owners

## 4. 2003.5. PICMG Plug-Fest 参加企業

- |                    |                          |
|--------------------|--------------------------|
| 1. ZNYX            | スイッチブレード                 |
| 2. PigeonPoint     | シャーシマネジメント               |
| 3. Radisis         | キャリアブレード+メザニンプロセッサ       |
| 4. Kapael (RITTAL) | シャーシ                     |
| 5. FORCE           | シャーシ、マネジメントカード           |
| 6. Tracewell       | 開発用シャーシ                  |
| 7. Asis            | シャーシ(12U、4U)             |
| 8. Intel           | システム(空冷特性・デモ)            |
| 9. Intel           | ネットワークプロセッサブレード          |
| 10. Schroff        | シャーシ(12U、4U)             |
| 11. Carlo Gavazzi  | 開発用シャーシ(6スロット)           |
| 12. NMS            | T1/E1インタフェースカード、DSPカード   |
| 13. Diversifield   | 廉価版サーバブレード               |
| 14. ELMA           | シャーシ(12U、4U)             |
| 15. Motorola       | 10Gbps InfiniBand Fabric |
| 16. GNP            | PICMG2.16キャリアブレード        |

# 4. 2003.5. PICMG Plug-Fest デモ風景

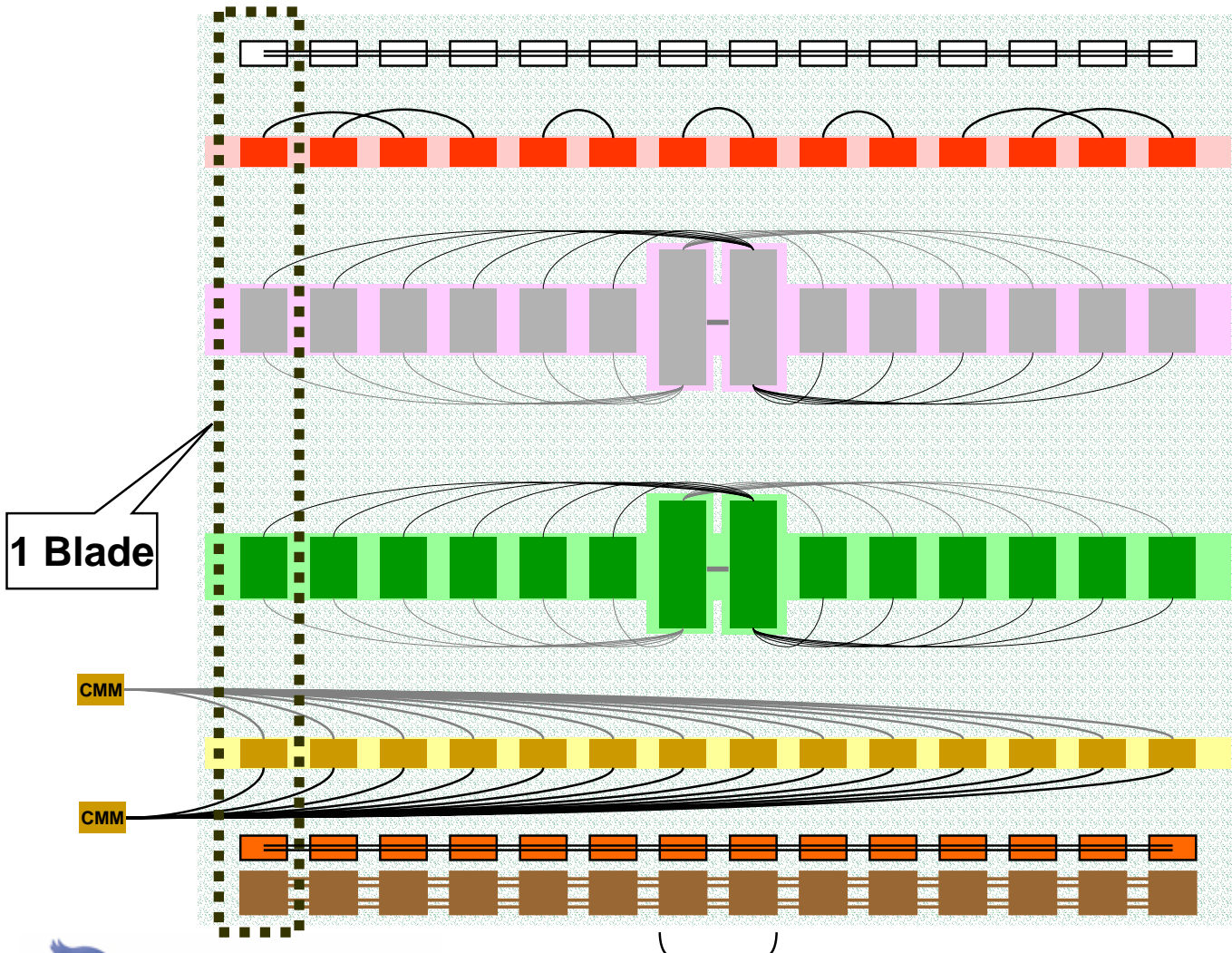


## 4. Intel ATCASBC諸元(PICMG3.0)

CPU		Low Voltage Intel Xeon Processor (PIVベース)
	クロック周波数	2.0GHz
	システムバス	400MHz
	2次キャッシュ	512KB
	搭載可能CPU数 (標準搭載数)	2 (1)
チップセット		Intel E7501
メモリ		DDR266 SDRAM-DIMM
	標準	4GB (1GB x 4)
	最大	8GB (2GB x 4)
内蔵ハードディスク		2.5inch HDD 40GByte(IDE ATA33/66/100対応) 容量は変更可能。 ATA33:266Mbps/ATA66:533Mbps/ATA100:800Mbps
IOインターフェース		10M/100M/1Gb Ether Port x2(Back) Fiber Channel Port x2(Front or Back) Optional ・・・1.06Gbps or 2.12Gbps IPMB x2(Back)、Serial Port(Front)、USB(1.1) Port(Front)、PMC slot
最大消費電力		200W
ボードサイズ		322.25mm x 280mm



# 4. ATCAバックプレーンアーキテクチャ



**Timing Clocks**  
(6 x 1 diff. pair each)

**Update Ports**  
(10 diff. pairs each)

**High-speed Star**  
(8 diff. pairs each)

**Ethernet Star**  
(4 diff. pairs each)

**IPMB Star**  
(2 traces each)

**8 Ring/Test Lines**  
**-48VDC Power**  
(2 feeds of -48V, RTN)

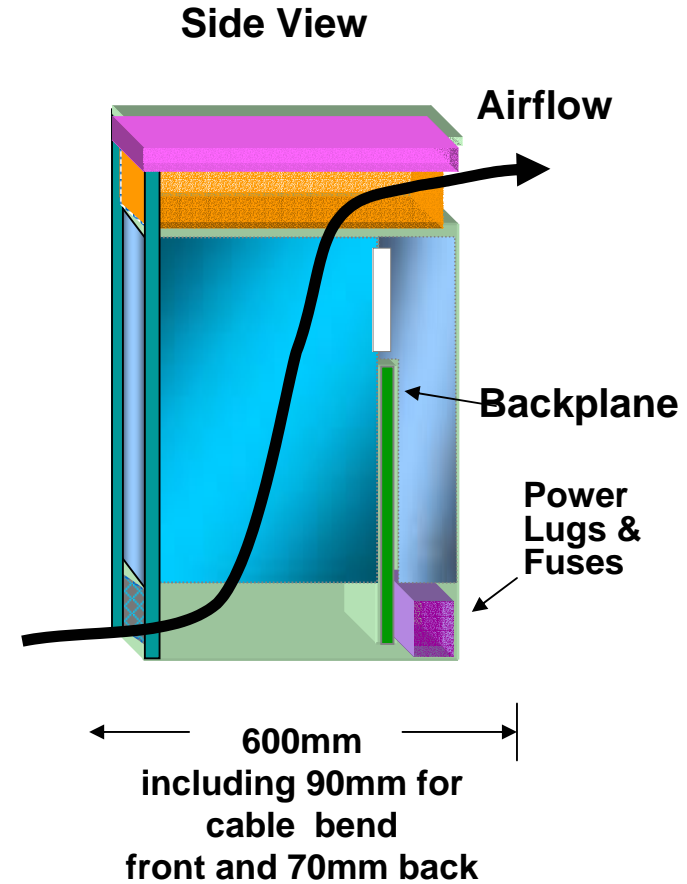
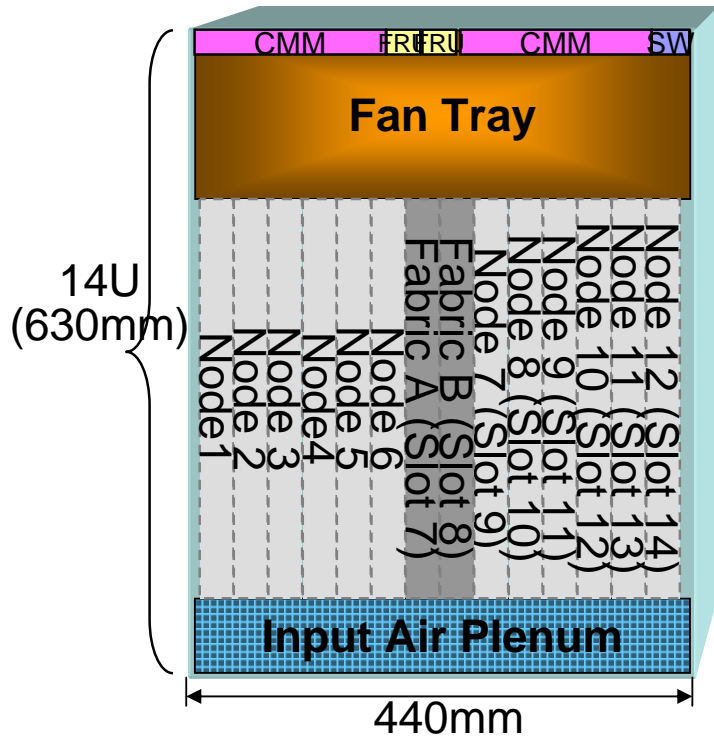
1 Blade

CMM

CMM

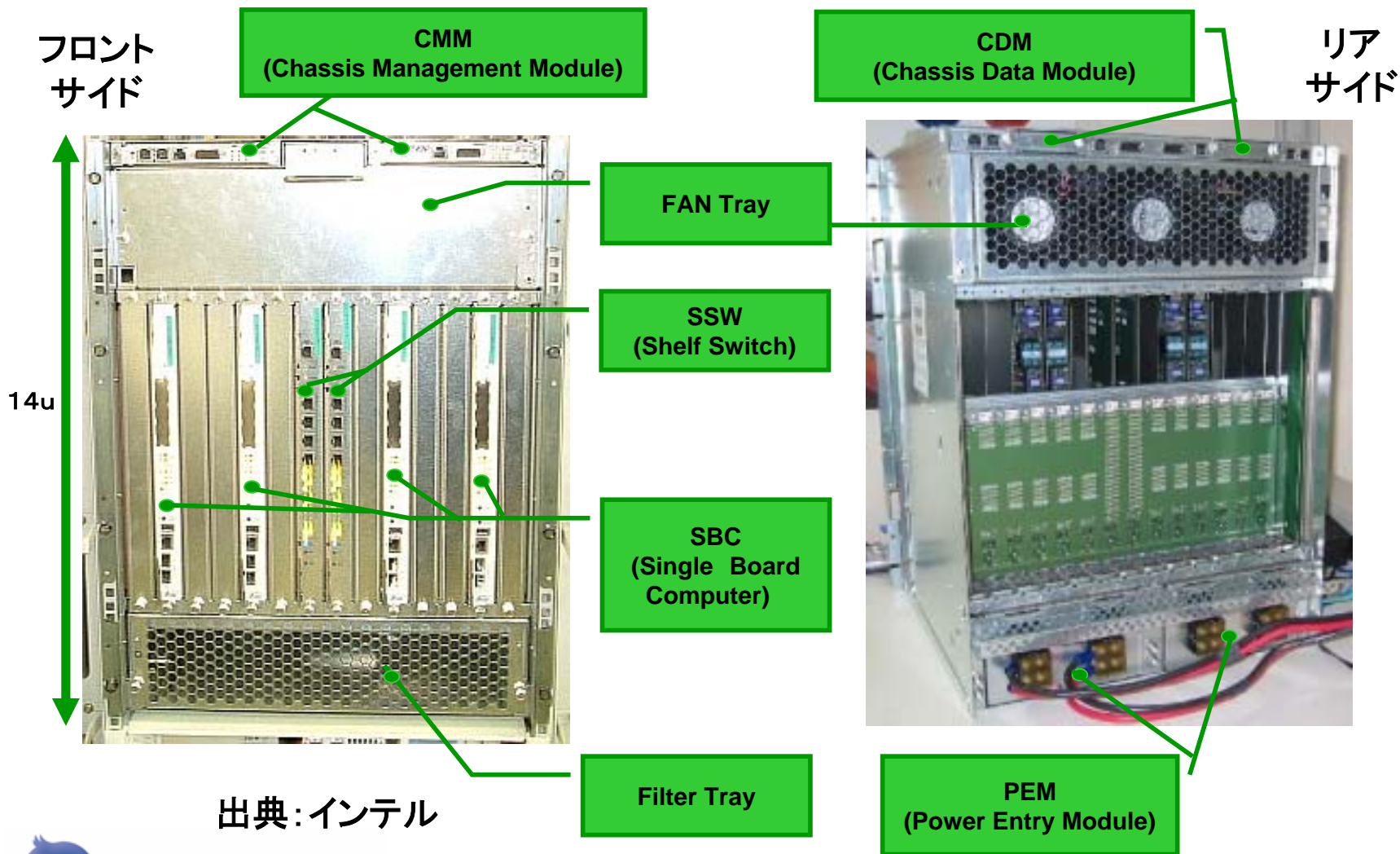
Central Switch  
(Fabric Blades)

# 4. ATCA実装イメージ(14slotシャーシ)

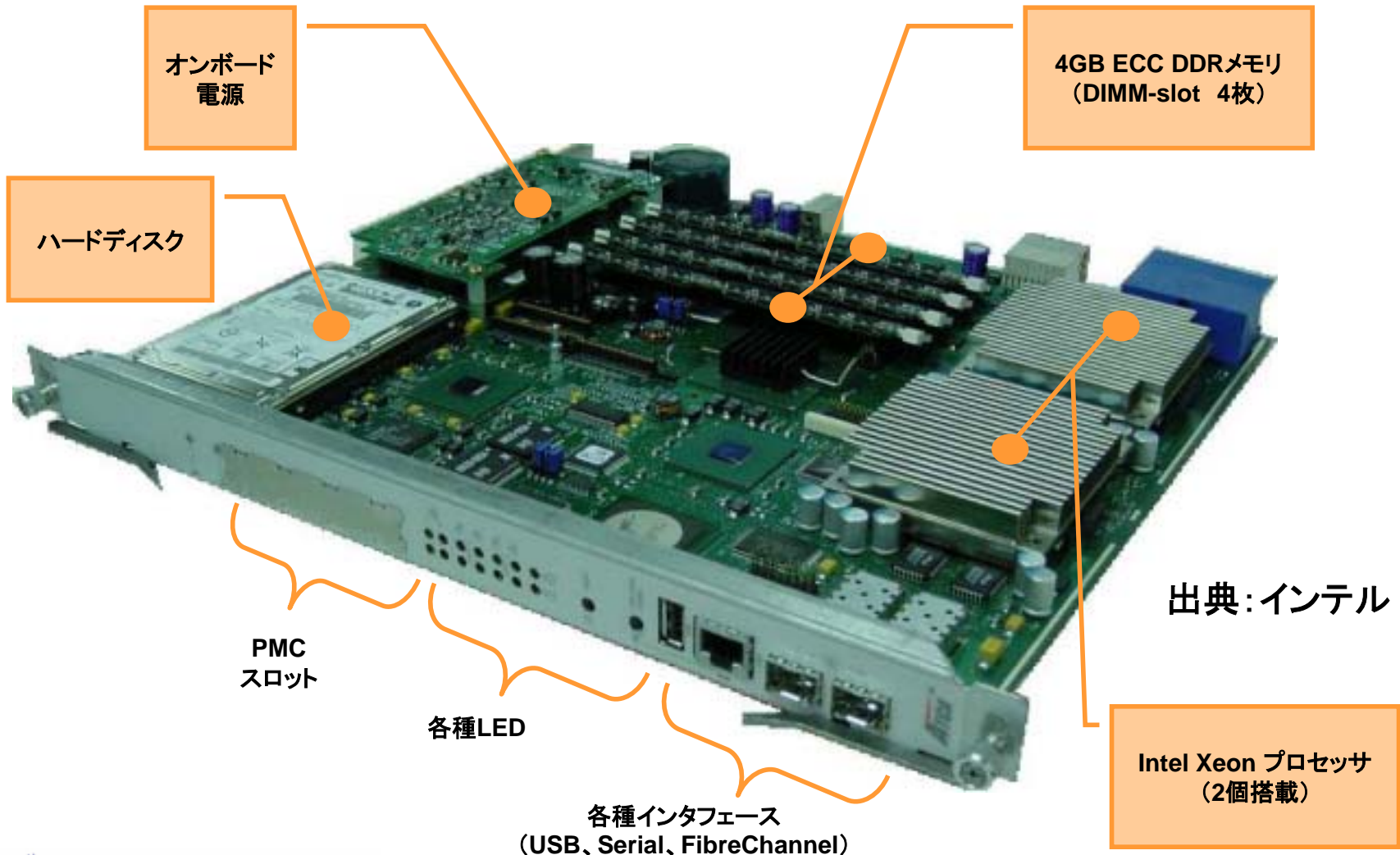


FRU = Chassis Field Replaceable Unit Module  
 CMM = Chassis Management Module  
 SW = Optional Power Switch

# 4. ATCA装置外観



# 4. ATCA Processor Boards : Single Board Computer



## 5. キャリアグレードLinux概要

### OSDL(CGL WG)アクティビティ

2002年

8月 7日 CGL要求定義V1. 0発表

10月31日 CGL要求定義V1. 1発表

2003年

10月 9日 CGL要求定義V2. 0発表

2004年 CGL要求定義V3. 0公開予定

### ディストリビューション(モンタビスタ社の例)

2002年

7月23日 MontaVista Linux Carrier Grade Edition(MV CGE)2. 1リリース

(cPCI2.16,RMS)

2003年

1月21日 MV CGE3. 0リリース

(CGL V1.1対応)

**12月末 MV CGE3. 1リリース予定**

(ATCA対応,SAF対応,CGL V1.1完全対応  
Native POSIX Threads Library (NPTL) )

2004年

**上半期 MV CGE4. 0リリース予定**

(CGL V2.0対応,Kernel 2.6)

モンタビスタ社関連プレス

<http://www.montavista.co.jp/news/2003/03oct08.html>

<http://www.montavista.co.jp/news/2003/03oct15.html>

## 5. キャリアグレードLinuxの生み出す物

- ◆ テレコム／データコムで要求されるハイ・アベイラビリティ、障害管理、リアルタイム性などの機能を提供
- ◆ ネットワーク機器開発メーカーにとり以下のメリットを生み出す
  - 開発期間の短縮
  - 総コストの削減
  - 高可用ミドルを搭載することで真にキャリアサービスに耐えられるプラットフォームの構築を可能とする

## 5. キャリアグレードLinuxの要求定義

### ◆ カテゴリー毎に分類

- Standards, Clustering, Security, Platform, Availability, Performance, Scalability, Tools, Serviceability

### ◆ 優先度を付ける

- Priority 1 — 現状での要求仕様、現行開発仕様にくみこまれるべきもの
- Priority 2 — 次期バージョンに組み込もうと準備しているもの
- Priority 3 — 将来バージョンに組み込もうとしているもの

注) 優先度の定義はV2. 0で変更されました

- 1 この仕様に従いキャリアグレードシステムにおいて必須
- 2 キャリアグレードシステムにおいて望ましい、しかし重要ではありません
- 3 現在は重要であるとみなされないが、しかし今後はそうなるかもしれませぬ。

## 5. キャリアグレードLinuxの要求定義

### ◆ Standards Requirements

- オープン規格と既成コンポーネントに適用される標準化に追従すること
- OSDL CGLでは標準仕様を作成するのではなく、準拠する要求仕様を策定すること
- OSDL V1.1では4カテゴリーで検討されました
  - Linux Standard Base
  - POSIX interfaces
  - IETF RFCs – involving IPv6 and protocols like Stream Transmission Control Protocol
  - Service Availability Forum



## 5. キャリアグレードLinuxの要求定義

### ◆ Platform Requirements

- プラットフォームの性能は、重大な構成要素である
- OSDL CGLでは、プラットフォームやアーキテクチャを規定せず、プラットフォームの機能要件を策定する
  - プラットフォームの機能要件は、特定のベンダーの開発とは独立である
  - 仕様はプラットフォームの本質的な改良モデルを提案するもので、プラットフォームの開発に要求を行うものではない

### ◆ Availability Requirements

- ウォッチドッグタイマー
- ホットスワップ
- RAID DK、クラスタ構成DK管理
- 冗長NIC
- プロセスの周期監視 & リスタート 等

## 5. キャリアグレードLinuxの要求定義

### ◆ Serviceability & Tools Requirements

- カーネルのダイナミックプローブ
- マルチスレッドコアダンプ
- カーネルクラッシュダンプ分析ツール
- 擬似障害発生ツール
- カーネルパニックハンドラの拡張
- イベントログへのリモートアクセス 等

### ◆ Performance Requirements

- キャリア環境でのシステム応答性能
- ソフトウェアによるリアルタイム機能
  - 10msまたはそれ以上のスケジューリング遅延対策
- プリエンプティブカーネル
  - カーネルのプリエンプション機能

## 5. 1 CGL 要求定義 V1.0 の内容

- ◆ Linux Standard Baseに準拠
- ◆ POSIX\* タイマー・インタフェース標準に準拠
- ◆ POSIXシグナル・インタフェース標準に準拠
- ◆ POSIXメッセージキュー・インタフェース標準に準拠
- ◆ POSIXセマフォ・インタフェース標準に準拠
- ◆ イベント・ロギング (POSIX IEEE 1003.25)
- ◆ IPv6, IPSECv6, MIPv6 RFCs
- ◆ SNMPをサポート (V1, V2, V3)
- ◆ POSIXスレッド標準に準拠
- ◆ ホット装置挿入/抜取/識別
- ◆ リモート・ブートのサポート
- ◆ ブート・サイクルの検出
- ◆ 独自開発モジュールのロード
- ◆ ディスクレス・システムのサポート
- ◆ シリアル・コンソール接続のサポート
- ◆ デバイス・ドライバの堅牢化
- ◆ ウォッチドック・タイマーのサポート
- ◆ アプリケーション・ハートビート・モニター
- ◆ イーサネットリンクの集約 / フェイルオーバー
- ◆ RAID 0 / 1 のサポート
- ◆ 回復力に富むファイルシステムのサポート
- ◆ ディスクとボリュームの管理
- ◆ リソースのモニタリング
- ◆ カーネル・ダンプとダンプ解析
- ◆ カーネル・メッセージの構造化
- ◆ ダイナミック・デバッグ / プロブの挿入
- ◆ プラットホーム・シグナルハンドラのサポート
- ◆ イベントログへの遠隔アクセス
- ◆ ユーザレベル・デバッグのサポート
- ◆ カーネル・デバッグ
- ◆ ソフト・リアルタイム性能
- ◆ カーネル・プリエンプション
- ◆ アプリケーション・ローディングの高速化
- ◆ スケーリングの分析
- ◆ 同時実行タイマーのスケーリング動作と報告

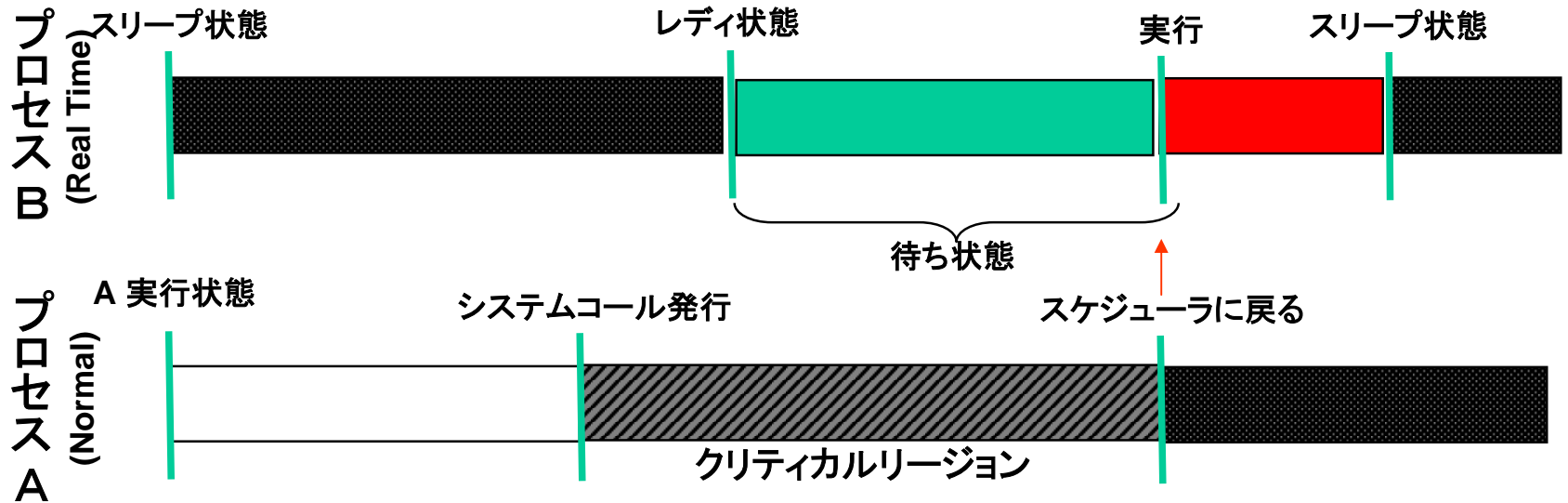


# 5. 1 CGL 要求定義 V1.0 の内容

## 1. リアルタイム&プリエンティブカーネル (モンタビスタ CGEでの独自のインプリ例を紹介します)

- ◆ 高精度タイマーによるリアルタイム性の向上  
→一般的なUNIXのタイマー精度は10msであるが、高精度タイマーにより1  $\mu$ sのタイマー設定が可能  
(従来の交換機のタイマーは、4msである)
- ◆ プリエンティブカーネルによるリアルタイム性の向上  
→一般的にカーネル走行中は優先度の高いプロセス が実行可能な状態になっても待ちが発生する。プリエンティブスケジューラにより、カーネル走行中でもプリエンションが発生し、リアルタイム性を確保する。
- ◆ O/1 (オーダーワン)スケジューラによるリアルタイム性の向上  
→既存のLinuxスケジューラは、スケジューリングキューがシステムで1つしか存在しなかったが、O/1スケジューラでは、プライオリティ毎にキュー管理し、リアルタイム性を向上させている。

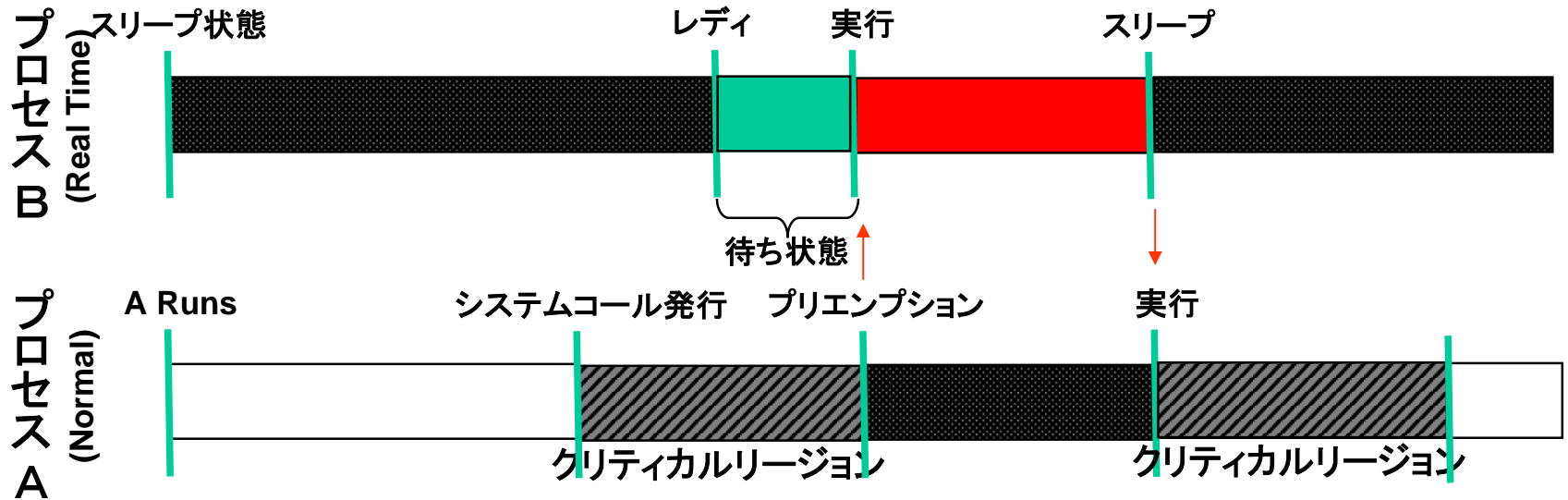
# 1. リアルタイム & プリエンティブカーネル - 既存のプリエンプション -



1. プロセスAがシステムコールを発行後は、カーネル内走行(クリティカルリージョン)
2. その間、優先度の高い(リアルタイム)プロセスがレディーになっても、プロセスAがシステムコールから戻ってくる時点までプロセスBは待ち状態となる。

リアルタイムプロセスであっても待ち時間が発生してしまう問題点がある

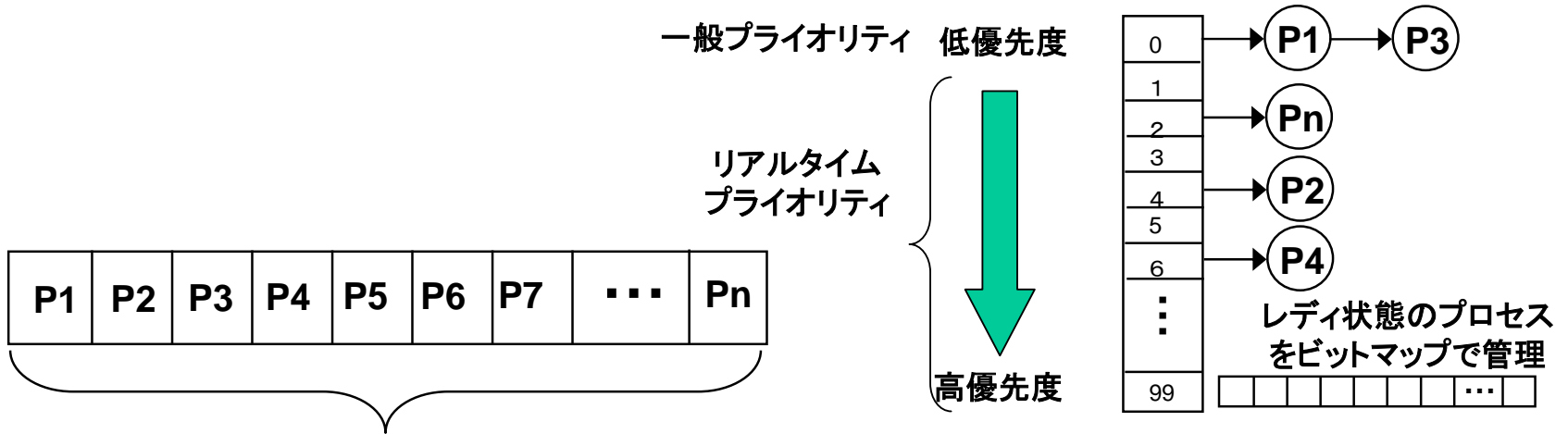
# 1. リアルタイム & プリエンプティブカーネル — カーネルプリエンプション —



1. プロセスAがシステムコールを発行後は、カーネル内走行(クリティカルリージョン)
2. システムコール実行中でも、クリティカルリージョンを抜ければ、プリエンプションが発生し、処理がプロセスAからプロセスBに移る
3. プロセスBがスリープした時点で処理がプロセスAに戻る

リアルタイムプロセスの待ち時間がカーネルプリエンプションにより短縮する

# 1. リアルタイム & プリエンティブカーネル — O/1(オーダーワン)スケジューラ —



既存スケジューラでは、次に実行すべきプロセスを決める場合、全てのプロセスを調査する必要がある

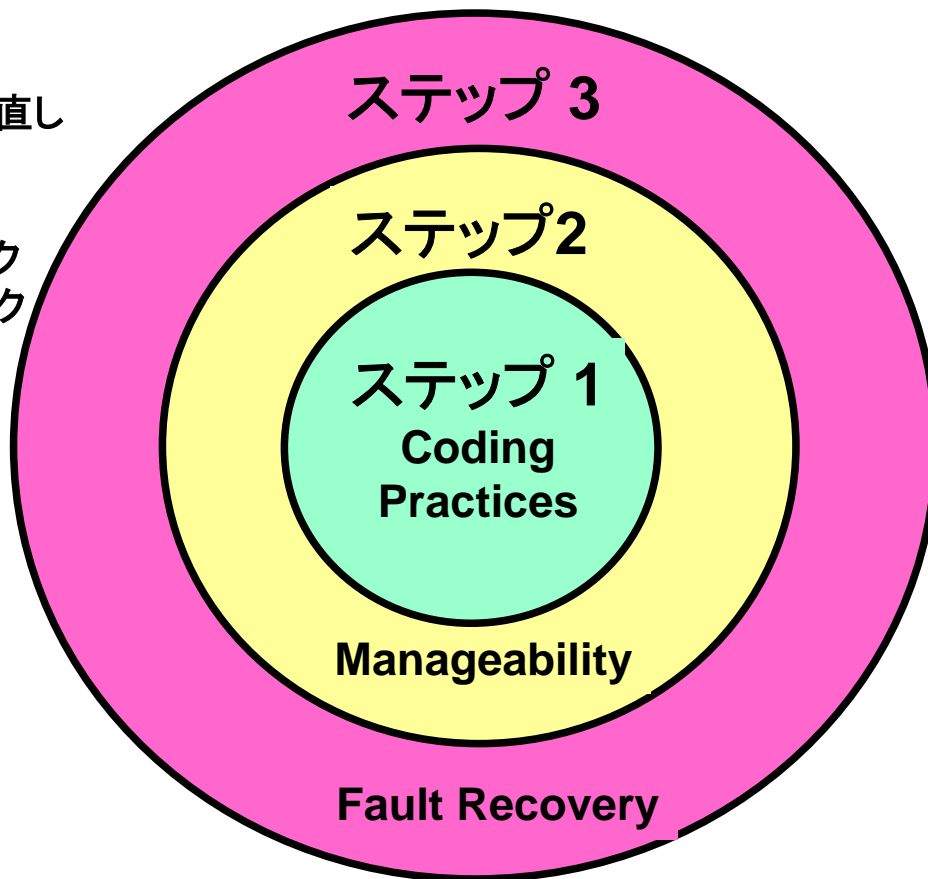
1. 各プライオリティ毎にキュー管理
  2. レディ状態のプロセスをビットマップで管理
  3. プライオリティは最大2047まで拡張可能
- 効率良くスケジューリングが可能

# 5. 1 CGL 要求定義 V1.0 の内容

## 2. カーネル&ドライバのハーデニング

### ステップ1

- コーディング手法の見直し
- 装置障害時の対処
- パニック処理の見直し
- データの妥当性チェック
- リターンコードのチェック



### ステップ2

- 統計情報採取
- イベントログ採取
- 診断
- 障害検出テスト

### ステップ3

- 障害復旧

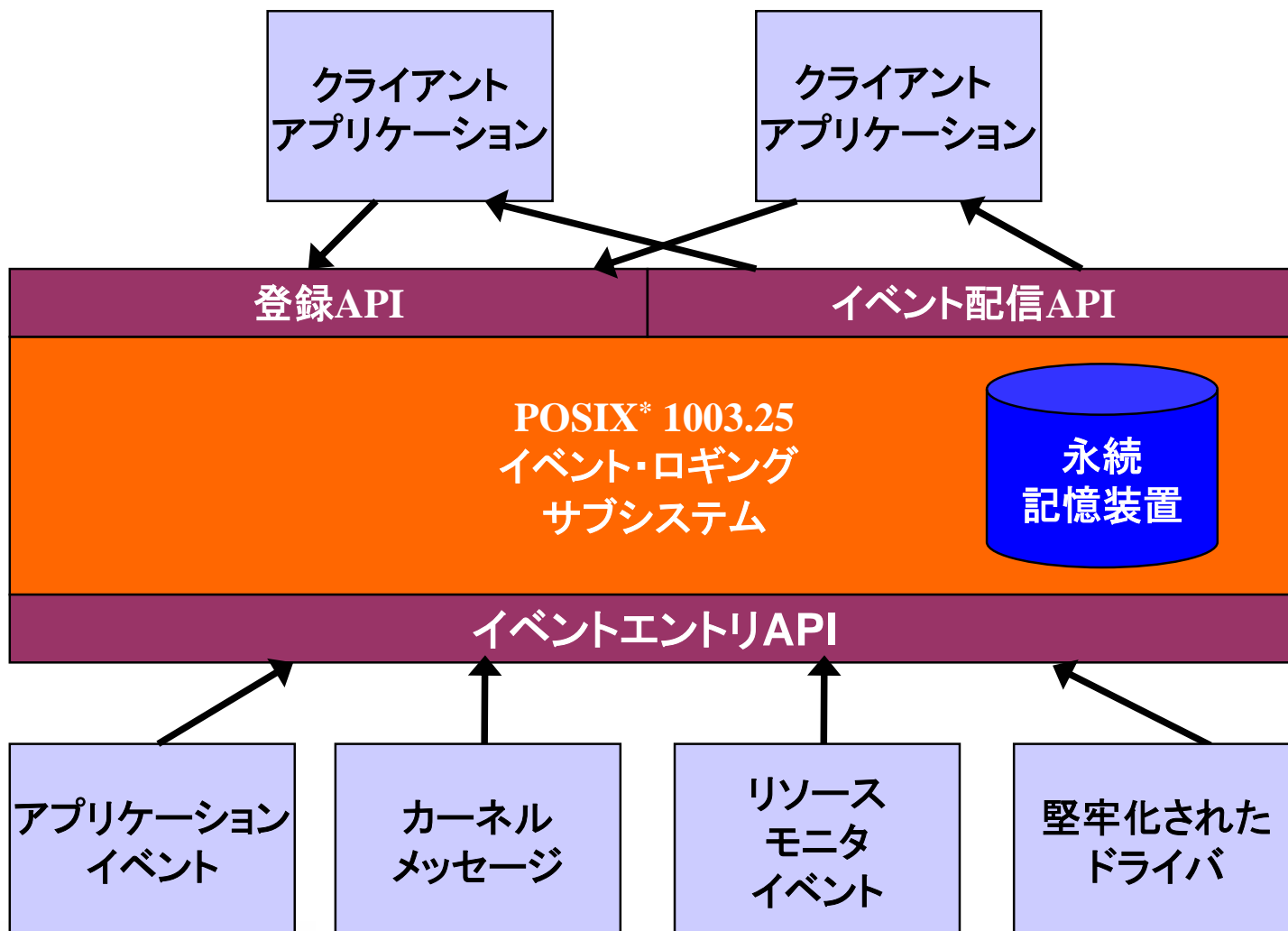


# 5. 1 CGL 要求定義 V1.0 の内容

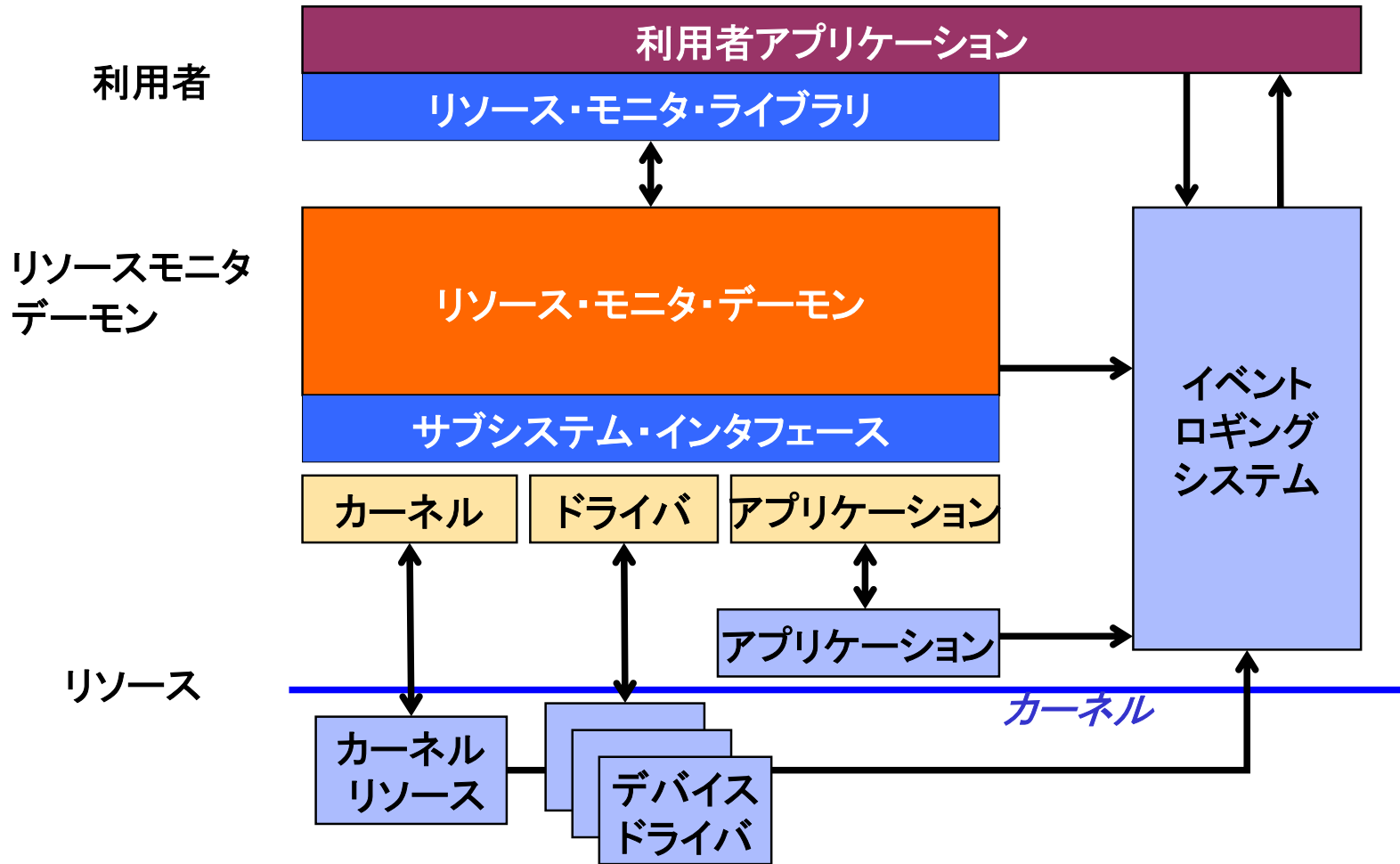
## 3. 情報収集機構

- ・イベントロギング
- ・リソースモニタ
- ・カーネルダンプ
- ・動的プローブ
- ・マルチスレッドコアダンプ
- ・パニックハンドラの拡張

# イベントロギング



# リソースモニタ



# カーネルダンプ

- ・カーネルダンプのターゲット
  - ー ローカルディスクの他に、ネットワーク経由で、遠隔装置に保存することも可能。
  - ー 遠隔保守や、ディスクレスシステムには重要。
- ・概要ダンプ
  - ー 完全なダンプのサブセットのみ保存する。
  - ー サブセットのカスタマイズが可能。
  - ー ダンプサイズを削減できるため、遠隔装置へのダンプには重要。
- ・カーネルダンプ解析
  - ー カーネルやプロセスの状態を解析するツールを提供。
  - ー プロセス空間の解析も可能。

## 動的プローブ

- ・走行中のシステムへのデバッグコードの挿入が可能。
- ・プローブハンドラは、スクリプト言語で、ユーザが記述する。
- ・プローブハンドラは、プローブが実行されたときに呼ばれる。
- ・プローブハンドラで、ハードウェア状態やメモリの情報を収集することも可能。

## マルチスレッドコアダンプ

- ・プロセスの中の全スレッドのレジスタ情報とスタック情報を、コアダンプに保存。
- ・マルチスレッド・アプリケーションのデバッグ効率を高める。

# パニックハンドラの拡張

- ・現状のパニックハンドラは、パニック原因の解析に必要な情報を、十分収集できていない。
- ・以下の機能拡張を実施
  - － システムイベントログにエントリを作成。
  - － IPMI経由でSNMPトラップを送信。
  - － システムメモリダンプの初期設定。
  - － カーネルデバッグ起動のトリガーとする。

## 5.2 CGL要求定義V2.0

Carrier Grade Linux Requirements Definition V2.0はV1.1の定義に対し、以下の3つの大カテゴリーに分類し、クラスタリング、セキュリティという、新しい要求定義項目を追加して2003年10月9日に公開されました。

**一般システム** : Linuxカーネル、コアライブラリ、及びキャリアアーグレードシステムに必要なツールに適合する要件

**クラスタリング** : 冗長のリソースおよび復旧能力によってサービス可用性を高いレベルで提供し、かつ水平方向のスケラビリティアップを支援する要件

**セキュリティ** : 認証、アクセスコントロール、機密性と統合性、監査、セキュリティ管理

## 5. 2 CGL要求定義V2. 0

### 一般システム要件(1/2)

- STD.1.0 Linux Standard Base Compliance
- STD.2 POSIX Compliance
  - STD.2.1 POSIX Core Functionality
  - STD.2.2 Barriers
  - STD.2.3 Process CPU-Time Clocks
  - STD.2.4 Clock Selection
  - STD.2.5 ISO-C Extensions
  - STD.2.6 IPv6
  - STD.2.7 Monotonic Clock
  - STD.2.8 Message Passing
  - STD.2.9 Realtime Signals Extension
  - STD.2.10 Semaphores
  - STD.2.11 Spin Locks
  - STD.2.12 Thread CPU-Time Clocks
  - STD.2.13 Threads
  - STD.2.14 Timeouts
  - STD.2.15 Timers
  - STD.2.16 Thread Priority Inheritance
  - STD.2.17 Thread Priority Protection
  - STD.2.18 Thread Execution Scheduling
  - STD.2.19 Thread Stack Address Attribute
  - STD.2.20 Thread Safe Functions
  - STD.2.21 Thread Process-Shared Synchronization
  - STD.2.22 Thread Stack Address Size
  - STD.2.23 XSI Extensions/ISO C Extensions
  - STD.2.24 Synchronization and Scheduling Support
- STD.3.0 SNMP Support Update
- STD.4.0 Stream Control Transport Protocol (SCTP)
- PLT.1.0 Persistent Device Naming
- PLT.2.0 IPMI 1.5 Support
- PLT.3.0 Service Availability Forum Hardware Platform Interface
- AVL.1.0 Robust Mutexes
- AVL.2.0 Software ECC Support
- AVL.3 Software Live Installation and Upgrade
  - AVL.3.1 Software Remote Update and Installation
  - AVL.3.2 Software Live Upgrade Minimal Reboot
  - AVL.3.3 Software Live Upgrade RPM Version Check
  - AVL.3.4 Software Live Upgrade Log
  - AVL.3.5 Software Live Upgrade per-Node Custom Image and Configuration
  - AVL.3.6 Software Live Upgrade Manual Rollback
  - AVL.3.7 Software Live Upgrade Automatic Rollback
  - AVL.3.8 Software Live Upgrade Application Participation
  - AVL.3.9 Software Live Upgrade Fine-Grain Version Checking
- AVL.4.0 Force Unmount
- AVL.5.0 Linux Panic Handler Enhancement
- AVL.6 Memory Overcommit Actions
  - AVL.6.1 VM Strict Over-commit
  - AVL.6.2 Replaceable OOM Killer



## 5. 2 CGL要求定義V2. 0

### 一般システム要件(2/2)

- SVC.1 SNMP Support Update
  - SVC.1.1 SNMP Support for IPV4
  - SVC.1.2 SNMP Support for IPV6
  - SVC.1.3 SNMP Baseline MIBs
  - SVC.1.4 SNMP IPv6 MIBs
  - SVC.1.5 SNMP IPv6 Kernel Interface - Get
  - SVC.1.6 SNMP IPv6 Kernel Interface - Set
- SCL.1.0 Efficient Low-Level Asynchronous Events
- PRF.1.0 Soft Real Time Support Performance
- PRF.2.0 Managing Transient Data
- PLT.4.0 Boot Cycle Detection
- PLT.5.0 Network Console Operation
- AVL.7.0 Fault Isolation Enabling
- AVL.8 Non-intrusive Monitoring of Processes
  - AVL.8.1 Process-level Non-intrusive Application Monitor
  - AVL.8.2 Kernel-level Non-intrusive Application Monitor
- AVL.9.0 Disk Predictive Analysis
- AVL.10.0 SA Forum HPI Certification
- AVL.11.0 Checkpointing
- AVL.12.0 Multi-Path Access to Storage
- AVL.13.0 NFS Client Protection across Server Failures
- SVC.2.0 CIM
- SVC.3 Kernel Dump Enhancements
  - SVC.3.1 Kernel Dump Targets
  - SVC.3.2 Kernel Summary Dump
- SVC.5 Live Dumping
  - SVC.5.1 Live System Application Dump
  - SVC.5.2 Live System Debugging Kernel Dump
- PRF.3 Application (Pre)loading Capability
  - PRF.3.1 Application (Pre)loading Non-root
  - PRF.3.2 Application (Pre)loading Limits
- TLS.1.0 Kernel Dump Analysis
- TLS.2 Fault Injection
  - TLS.2.1 Fault Injection Framework
  - TLS.2.2 Fault Injection Points
- TLS.3 Kernel Profiling
  - TLS.3.1 Kernel Flat/Graph Execution Profiling
  - TLS.3.2 Kernel Sampling for Profiling
- TLS.4.0 System Tools to Analyze Execution Profiles
- PRF.4.0 ProcessAffinity
- PRF.5.0 Flexible Process Scheduling Policy Framework
- PRF.6.0 Process Priority Inheritance
- PRF.7 Interruptless Ethernet Delivery
  - PRF.7.1 Interruptless Ethernet Delivery Kernel Framework
  - PRF.7.2 Interruptless Ethernet Delivery Driver Support
- PRF.4.0 Page flushing

## 5. 2 CGL要求定義V2. 0

### クラスタリング要件

- CFH.2.0 Cluster Node Membership with Failure Detection
- CCM.1 Cluster Communication Service
  - CCM.1.1 Cluster Communication Service – Logical Addressing
  - CCM.1.2 Cluster Communication Service – Fault Handling
  - CCM.1.3 Cluster Communication Service – Quality of Service
  - CCM.1.4 Cluster Communication Service – Performance
  - CCM.1.5 Cluster Communication Service – Event Notification
- CFH.3.0 Prevent Failed Node From Corrupting Shared Resources
- CFH.4.0 LAN failure / Node Failure Analysis
- CFH.5.0 Application Fail-over Enabling
- CSM.2.0 Storage Network Replication
- CSM.3.0 Cluster-aware Volume Management for Shared Storage
- CSM.4.0 Shared Storage Consistent Access
- CSM.5.0 Shared Storage Mirroring
- CSV.1.0 Cluster-wide Resource Monitor
- CSV.2.0 Cluster-wide Identified Core Dump
- CSV.3.0 Cluster-wide Crash Dump Management
- CSV.4.0 Cluster-wide Log Collection
- CCM.2 Cluster Message Service (SA Forum AIS)
  - CCM.2.1 Cluster Messaging Service
  - CCM.2.2 SA Forum AIS Certification – Messaging Service

## 5. 2 CGL要求定義V2. 0

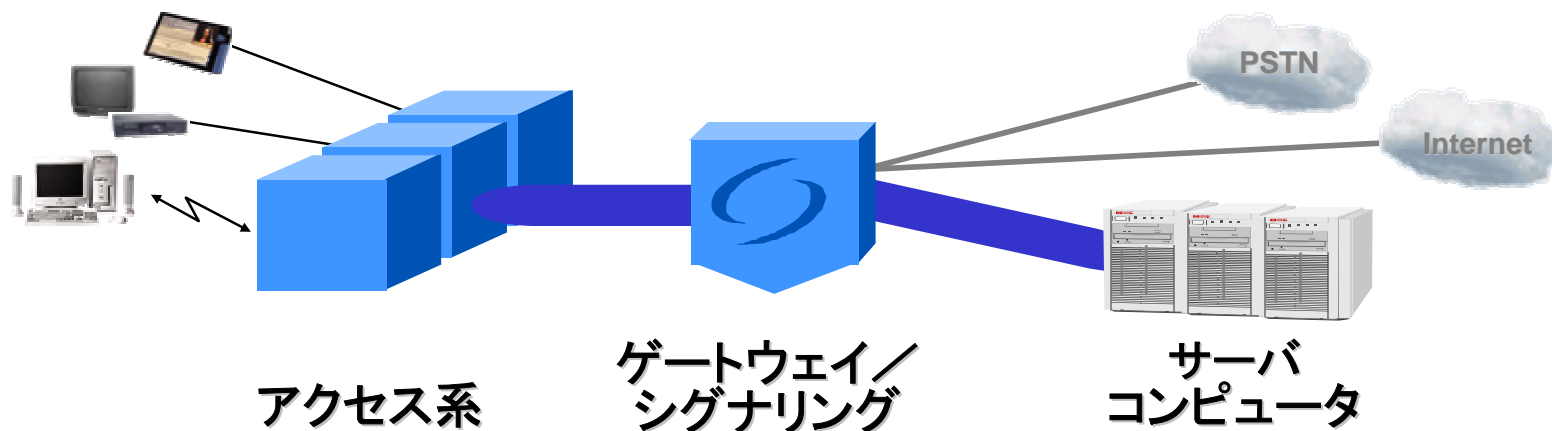
### セキュリティ要件

- AUT.1.0 Password Integrity Checking
- ACC.1.0 Support for Generic Kernel Security Modules
- CON.1.0 IPSec for IPv4
- CON.2.0 Support for IKE
- CON.5.0 PF\_KEY Support
- CON.3.0 Secure Integrity Check At User-level
- AUD.1.0 Logs Integrity And Origin Authentication Of Logs
- AUD.2.0 Confidentiality Of Log Information
- AUD.3.0 Automated Log Analysis
- CON.11.0 PKI CA Support
- AUT.1.0 Support For One Time Passwords
- ACC.2.0 Support Of Capabilities For File Systems
- ACC.3.0 Buffer Overflows Protection Mechanisms
- ACC.4.0 Access Control List Support for File Systems
- CON.6.0 Secure Integrity Verification Of Binaries Before Loading
- CON.7.0 Generic Hardware Crypto Device Support
- CON.8.0 IPSec Hardware Support
- CON.9.0 Secure Boot Signed Kernel Image
- CON.10.0 Secure Core Dumps
- AUD.4.0 Support For Event Logging Mechanisms
- ACC.5 Mandatory Access Control
  - ACC.5.1 MAC Based On Domain-Type Approach
  - ACC.5.2 MAC Based On Role-Based Access Control
  - ACC.5.3 MAC Based On Access Control Lists
  - ACC.5.4 Process Level Granularity
- ACC.6.0 Pre-emptive Security Mechanisms
- CON.12.0 Secure Boot Communications

## 6. SAF概要

- ◆ 2002年9月25日 Hardware Platform Interface Specification (HPI) 1.0発表
- ◆ 2003年4月14日 Application Interface Specification (AIS) 1.0発表
- ◆ 2003年12月末 モンタビスタはMV CGE3. 1で上記対応のサポート予定
- ◆ SAFに準拠したミドルウェアは、特定の先行メーカーに追随して、これから次々と発表されると思われる (NECも計画中)

# 7. ATCA適用領域



## Equipment types:

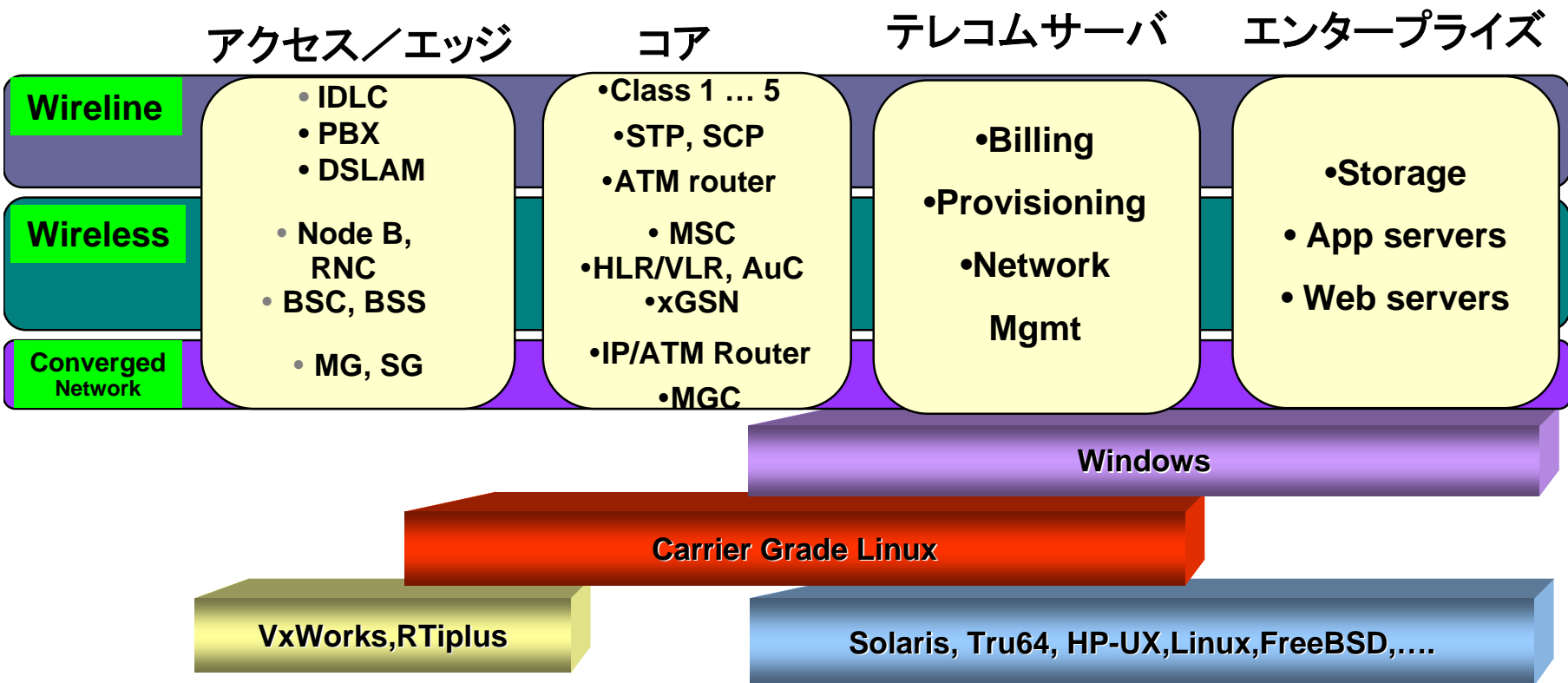
DSLAM  
Access Concentrator  
CMTS  
DLC  
PSTN Switch  
BTS  
PBX

Media Gateway (e.g. VOIP)  
Mobility Gateway  
Signaling GW (SS7)  
SGSN  
GGSN  
Radio Access NW (RAN) GW  
WAP GW  
VPN GW  
Multi-Access Switch

Mobility Servers (Control)  
HLR/VLR (Location Register)  
RNS  
BSC  
SoftSwitch  
Application Server  
Messaging  
Adjunct Processor  
Intelligent Peripheral

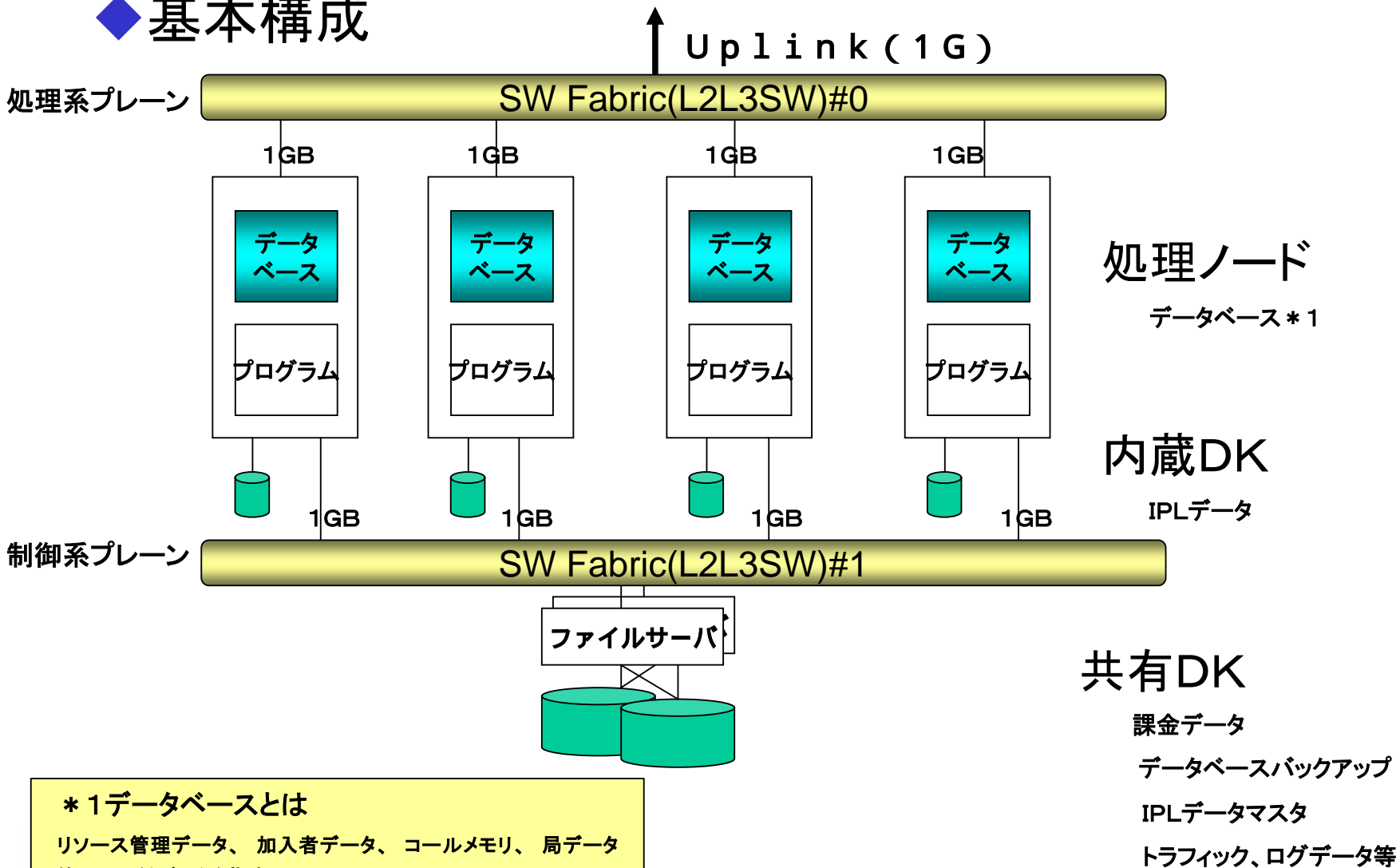
AdvancedTCA\*  
PICMG\* 3.x

# 7. 適用領域と最適カーネル



# 8. CGPF採用システムアーキテクチャー

## ◆基本構成

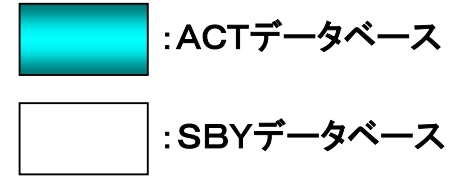


\* 1データベースとは

リソース管理データ、加入者データ、コールメモリ、局データ等の要引継データを指す

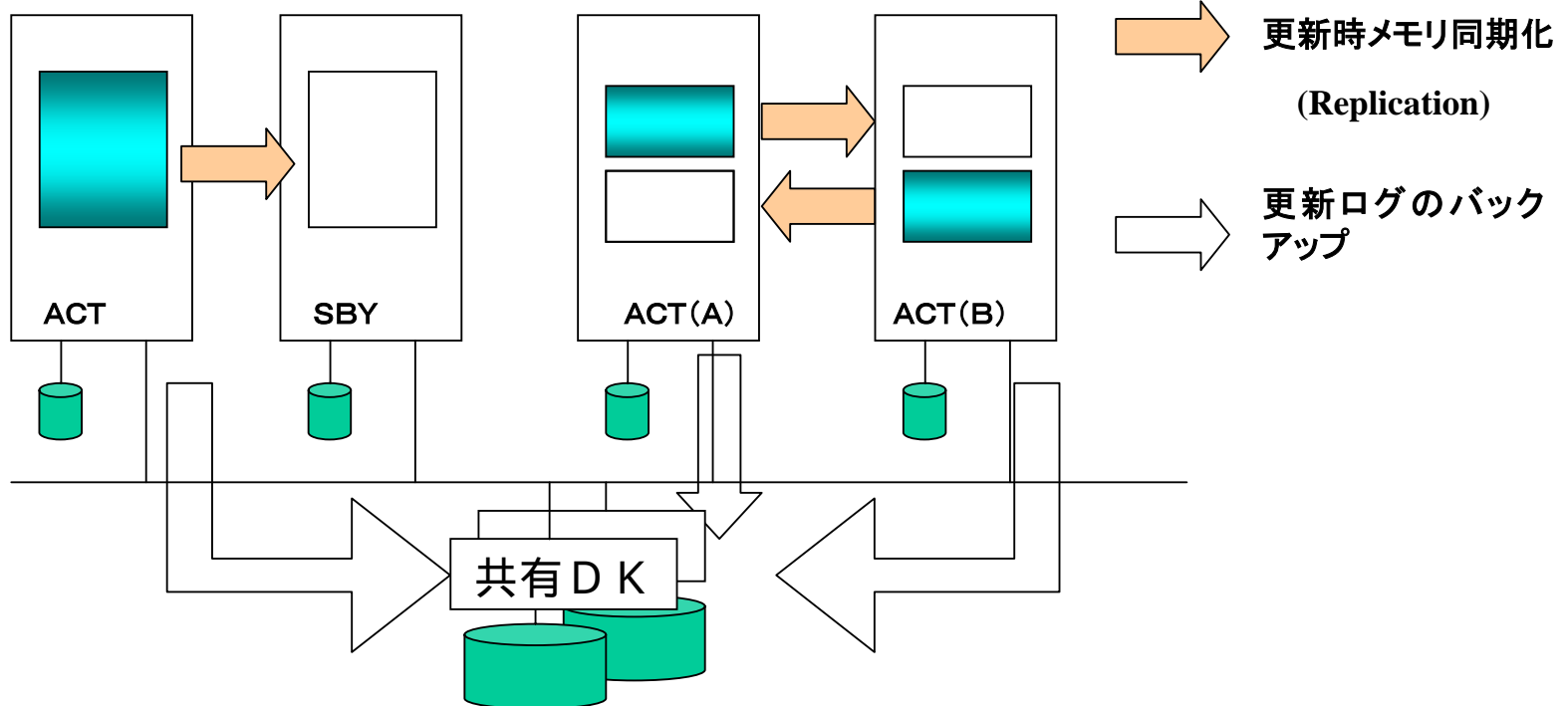
# 8. 冗長構成(1/2)

## ◆ メモリベースクラスタリングを行う



### 1 . A C T / S B Y方式

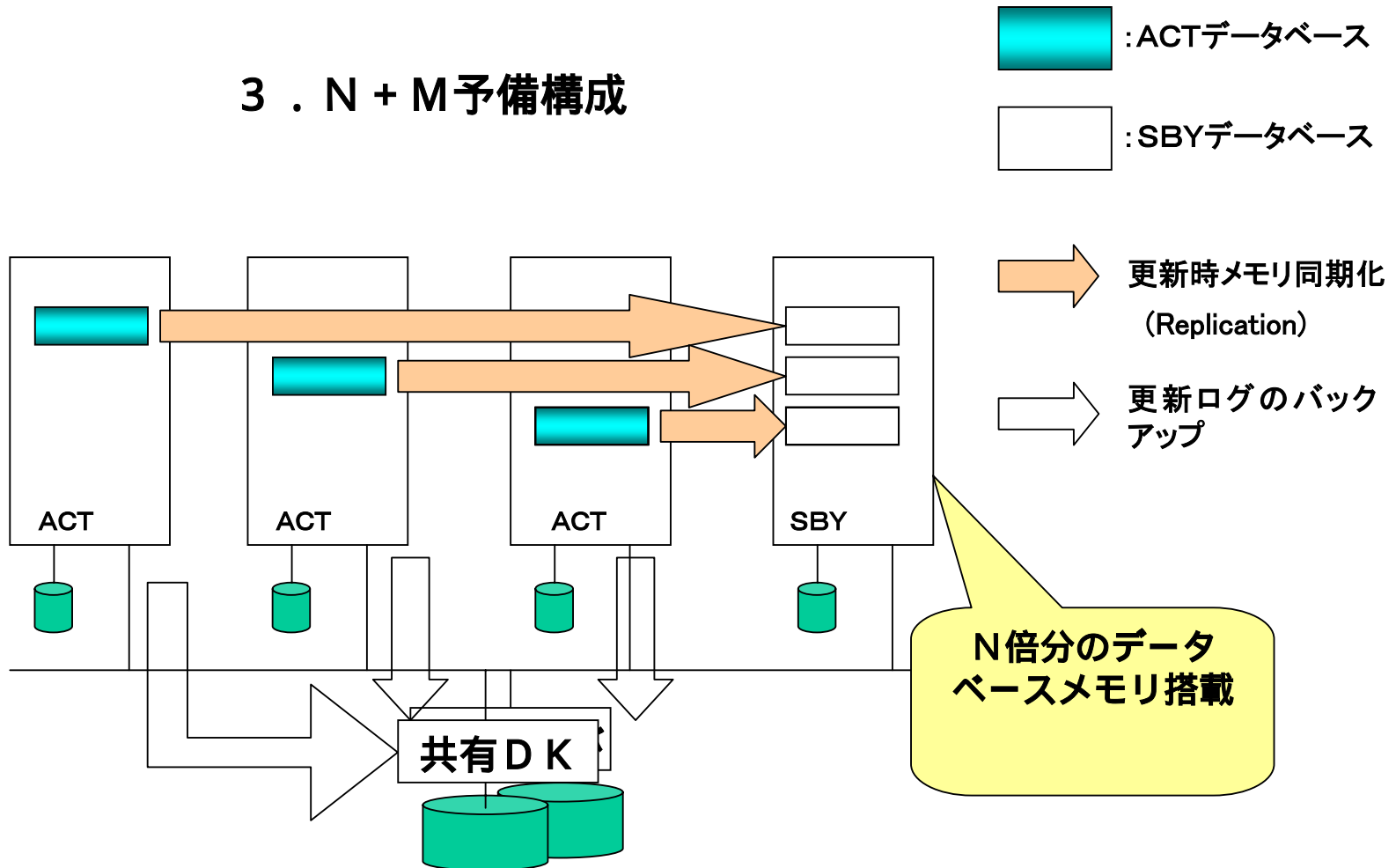
### 2 . 並列処理方式





# 8. 冗長構成(2/2)

## 3 . N + M予備構成



## 9. NECの取り組み(1/2)

- ◆ 以上紹介してきた新キャリアグレードPFにNECの高可用性ミドルウェアを搭載、通信インフラ機器向け高性能プラットフォームとして本年9月に実用化しました
- ◆ 世界初の商品化
  - 具体的製品の第一弾として、今後世界的に導入が本格化する第三代(3G)携帯電話網において、動画像や位置情報関連サービスなど、多様なサービスへの対応力を飛躍的に高め、より高度なサービスを実現できるパケットノード「新SGSN、新GGSN」に新プラットフォームを適用しました
  - 本製品をITU Telecom World 2003にてIntelと共同でデモンストラーションを実施し、関係者の注目を集めました  
<http://www.nec.co.jp/press/ja/0309/1001.html>
- ◆ NECは、CGLinux が通信キャリアにとっての標準OSとなり、さらに最適な運用プラットフォームの最上位に位置づけられる物へと成長すると確信しています

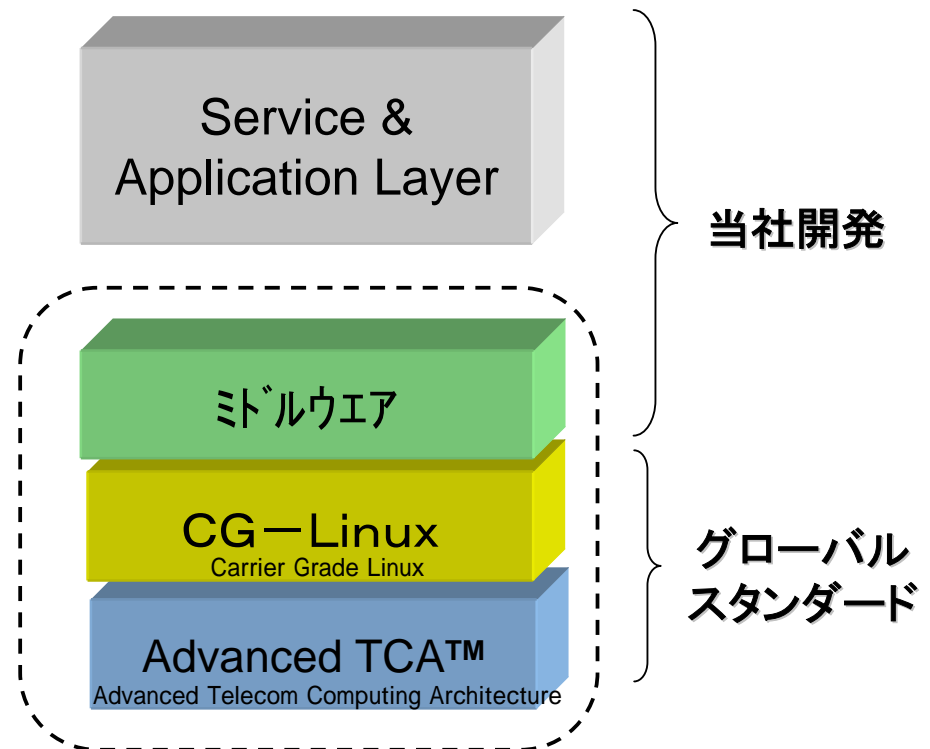
## 9. NECの取り組み(2/2)

- ◆ CGLinuxをキャリアのお客様に安心して使って頂ける体制を構築する為、オールNECの統括組織「Linux推進センター」を本年11月4日付けで設立し、キャリアグレード領域におけるLinux関連システムに対するサポート体制の整備を行います  
<http://www.nec.co.jp/press/ja/0310/1701.html>
- ◆ NECは他社に先がけたフィールド経験を生かし、OSDLへの貢献、CGLの成長に貢献する所存です

2003年9月10日  
発表!

## 9. 世界初、新パケットノードで製品化

3G携帯電話網の高度なサービスとIPパケット通信増大に対応するため  
PICMGフォーラム Advanced TCA™規格に準拠し  
キャリアグレードLinuxを採用



NEC & Linux

新通信プラットフォーム

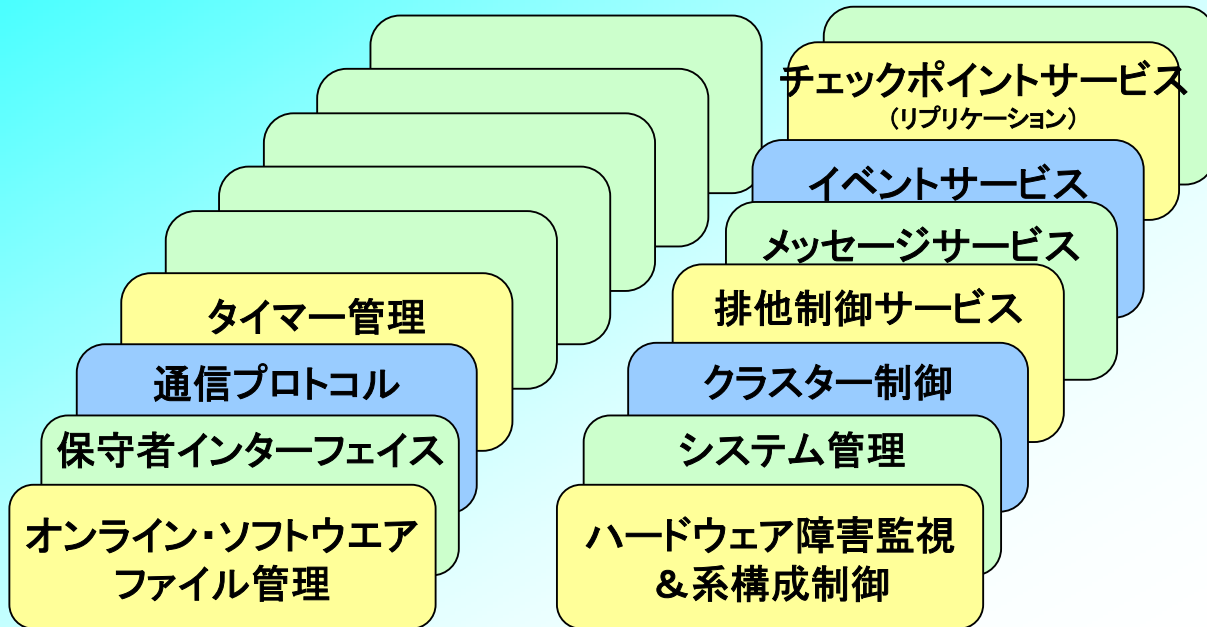
Empowered by Innovation

NEC

# 9. 新通信プラットフォームのミドルウェア構成

## アプリケーション群

### NEC開発ミドルウェア コンポーネント群



### 汎用ミドル群

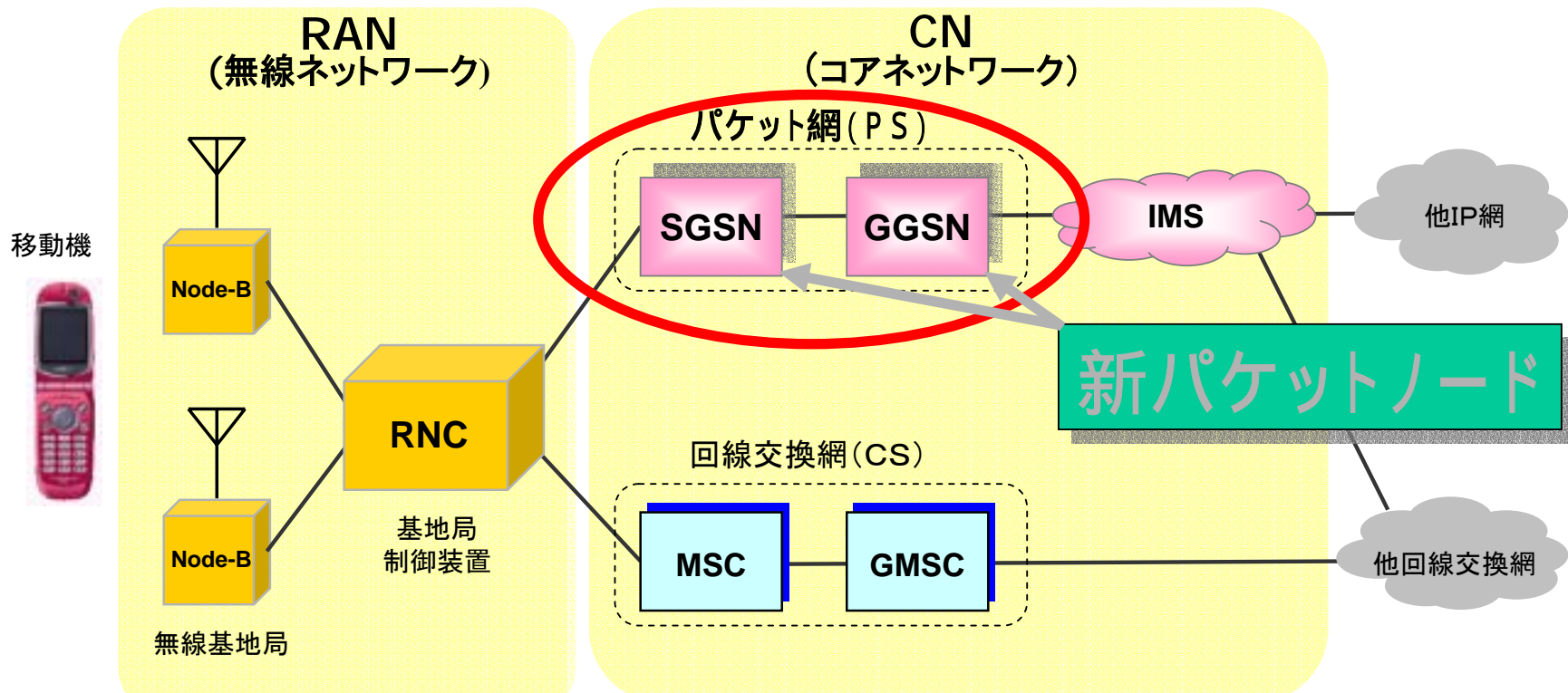
JAVA\_VM

CORBA

インメモリ  
データベース

## Carrier Grade Linux

# 9. 3G携帯電話網でのポジション



MSC : Mobile Switching Center  
GMSC : Gateway Mobile Switching Center  
SGSN: Serving GPRS Support Node  
GGSN: Gateway GPRS Support Node  
IMS: IP Multimedia Subsystem

Linuxがキャリアコアネットワーク領域へ