

平成14年度

# 並列分散システム論

## Part III

---

計算機システム講座

計算機アーキテクチャ分野

森眞一郎

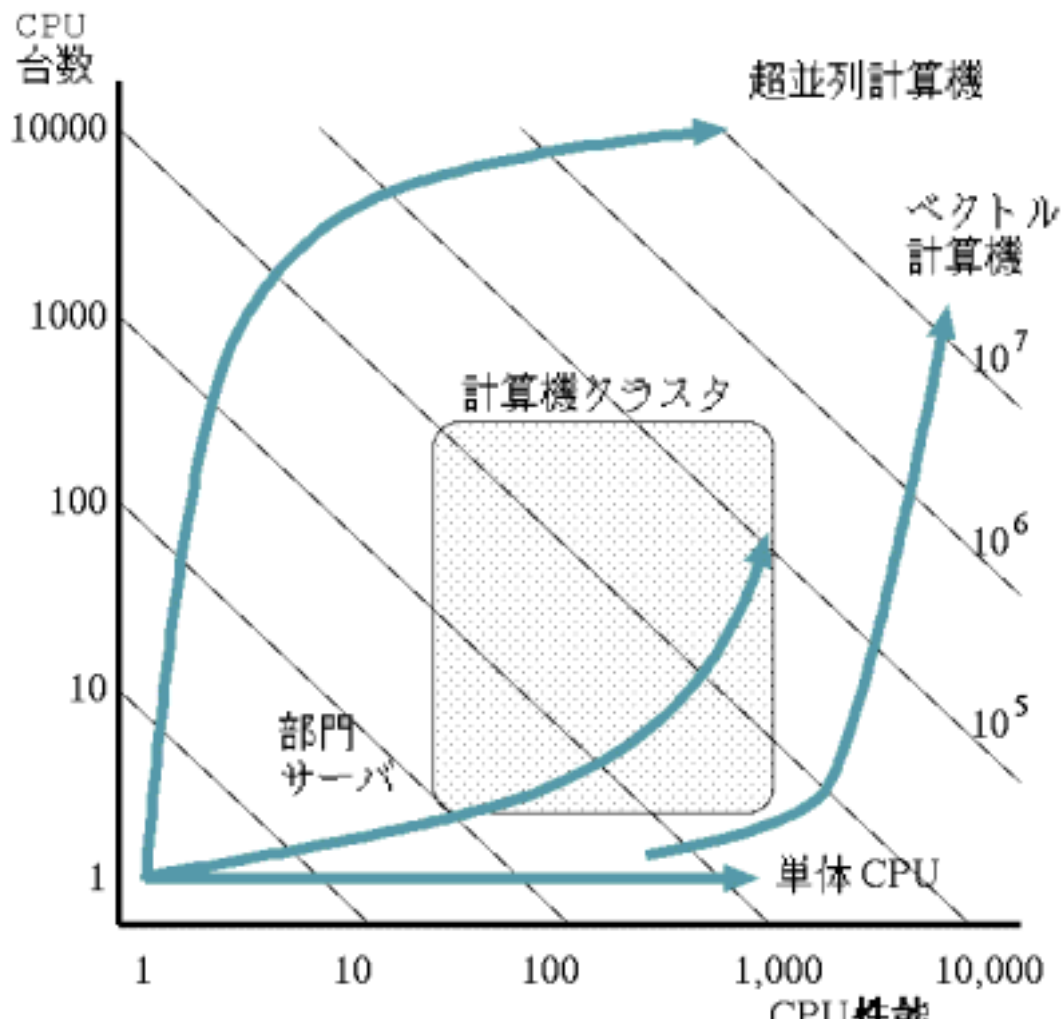


# 並列・分散システムの実現

---

- 計算機クラスタ
- 分散共有メモリシステム
- Grid (??)

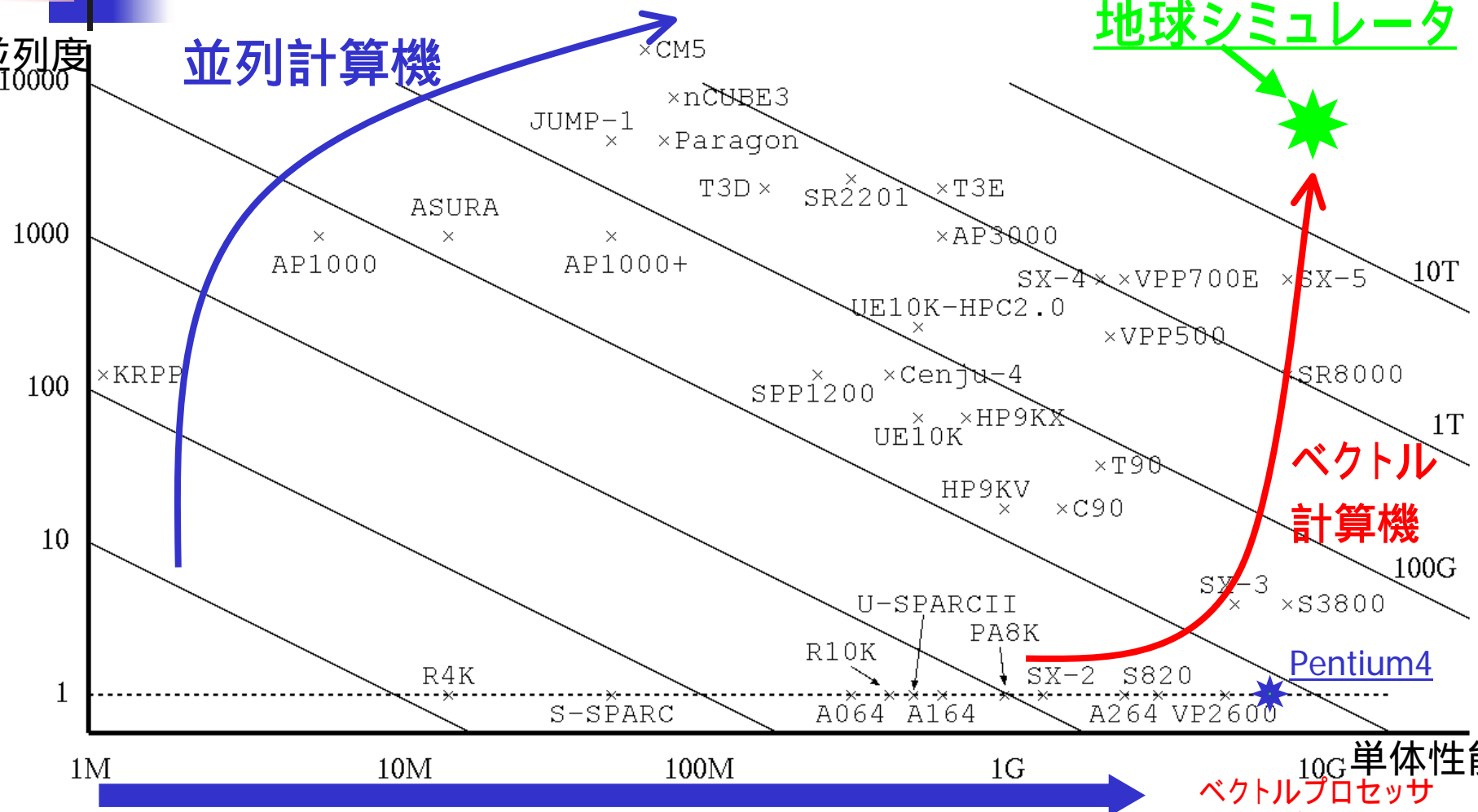
# 計算機クラスタの位置付け



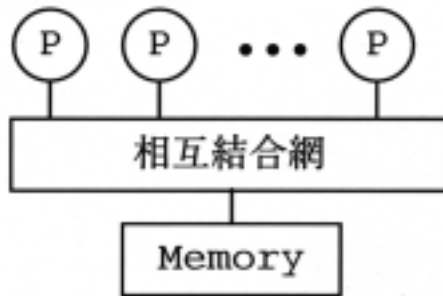
# 何が「計算機クラスタ」を今Hotにしているのか？

- CPUの高速化/低価格化..... Alpha/Pentium (PCの追い上げ)
  - 8GFlopsの壁???.....Alpha(612MHz)の6倍, UltraSPARC(336MHz)の12倍  
Pentium4(3.06GHz)の1.5倍
- 通信媒体の高速化
  - 2Gpbsの壁??? ....多重化(波長,位相,振幅), 並列化, 光のWave Pipeline
- 標準通信ライブラリの普及
  - Posix Thread, OpenMP, HPF, MPI, AM, FM, PM, .....
- 米国(だけ??)での並列計算機メーカーの不振.....(最近は??)
  - 地球シミュレータ..... 5120プロセッサ, 40TF (2002.03@海洋科学技術センタ 横浜)
  - Cray Inc. .... 単体性能12.8GFlopsのベクトルプロセッサを発表(2002.11)
    - 3.2GFlops のスカラプロセッサ4台 + ベクトル処理向けに強化した同期機構  
仮想ベクトル処理
    - 4096プロセッサで52.4TFlops (未稼働)、 2010年までに 1 PetaFlops を目指す
  - Blue Planet .... 2005年を目処にIBM Power5(10GF) x 16,384 = 150TFlops

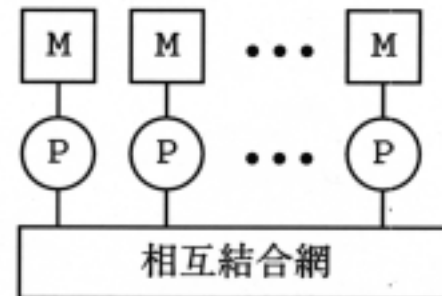
# いつまで続くCPUの高速化



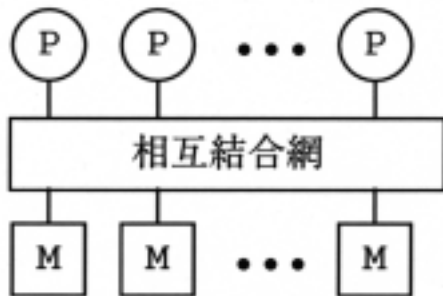
# 並列計算機の種類



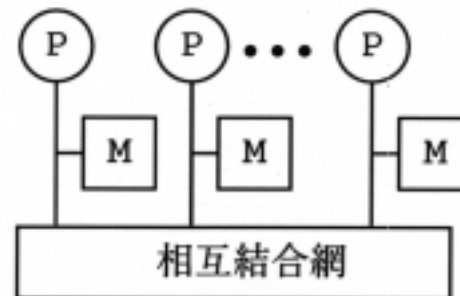
(a) 集中メモリ型  
(UMA or SMP)



(c) 分散メモリ型  
(NORA)

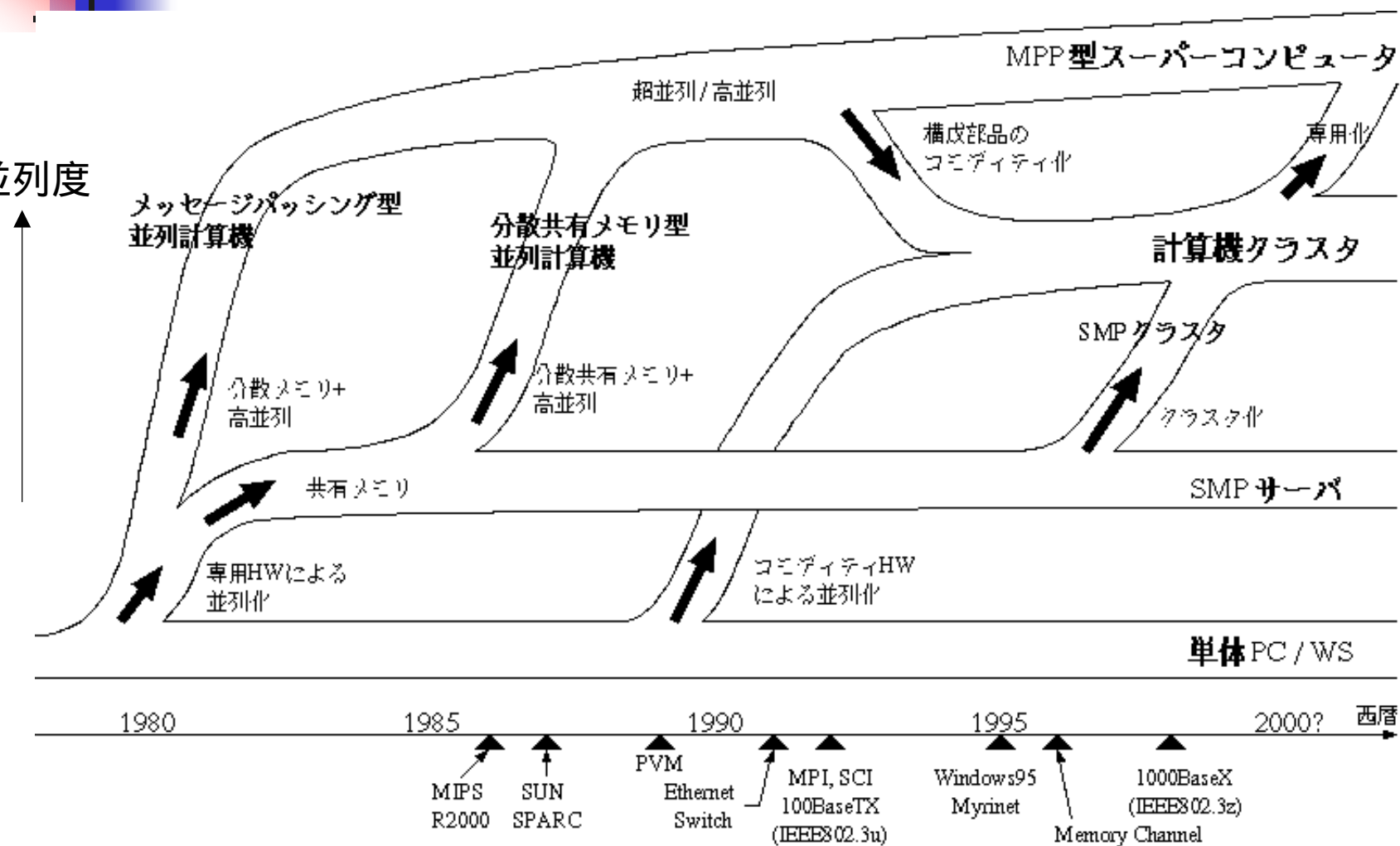


(b) 分散メモリ型  
(UMA/NUMA)

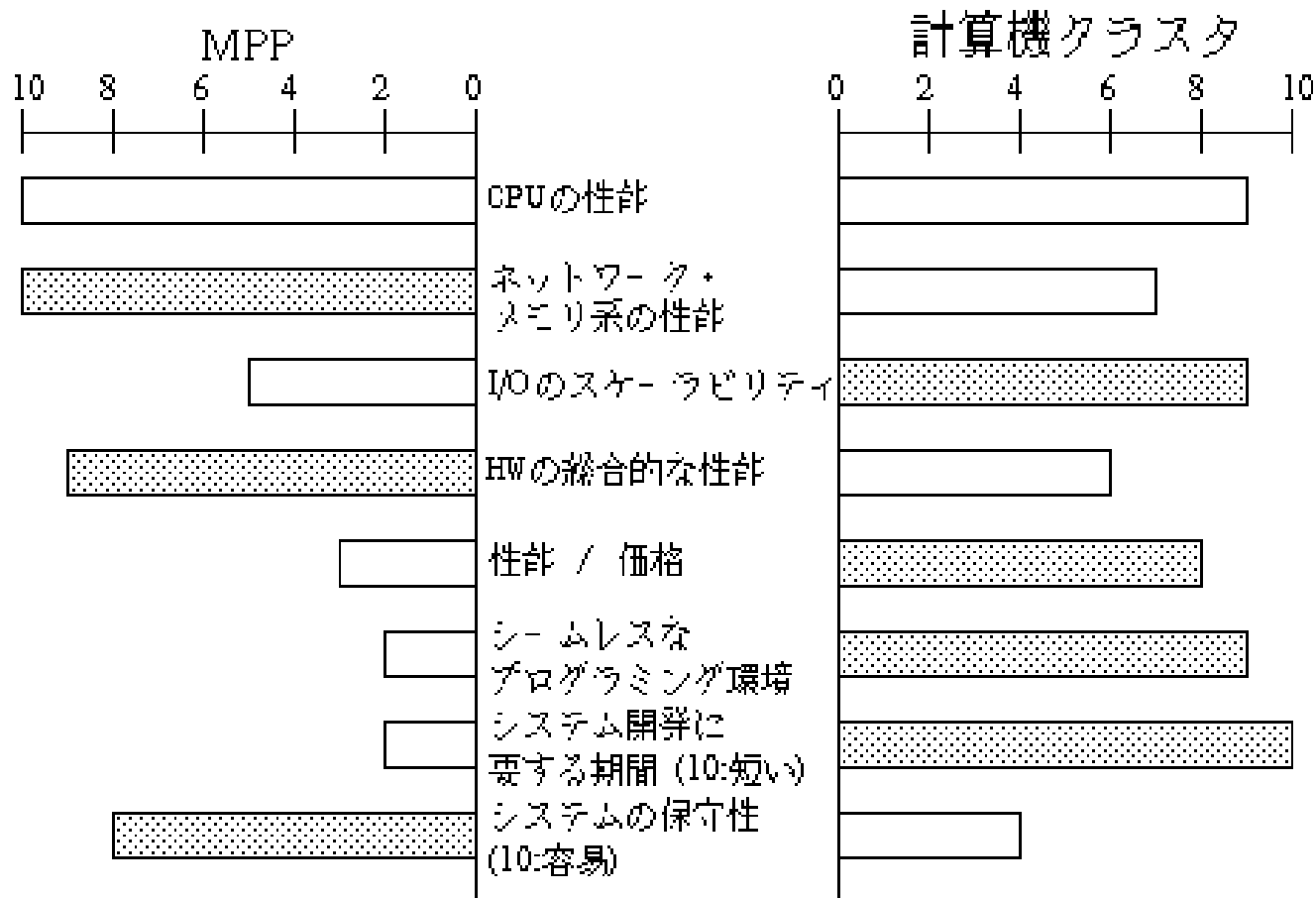


(d) 分散メモリ型  
(NUMA)

# 計算機クラスタへの道程



# MPPと計算機クラスタの比較







# ネットワークの急速な高速化

---

- 通信媒体自体の高速化
- 通信ソフトウェアの高速化

# 通信媒体自体の高速化

## 汎用の通信媒体として

- 10Base2/5 10BaseT 100BaseT Gbit Ether
  - バス型結線 スター型結線 ..... 相互干渉の軽減, 高周波設計の容易さ
  - 電気 光 ..... 物理的な距離に関する制約の緩和
- USB(Universal Serial Bus), IEEE1394バス
  - 家庭内ネットワークコンピューティングへの足掛かりになるか??

## 専用の超高速通信媒体として

- Myrinet(Myricom社)      通信用コプロセッサを搭載
- Memory Channel(DEC)      仮想共有メモリ環境を提供する PCI バス用
- SCI (IEEE1596)      CC-NUMA環境
- Synfinity NUMA(Fujitsu)      CC-NUMA環境 + 1.6Gbps
- QsNet (Quadrix)      VA-to-VA Remote DMA, 340MB/s

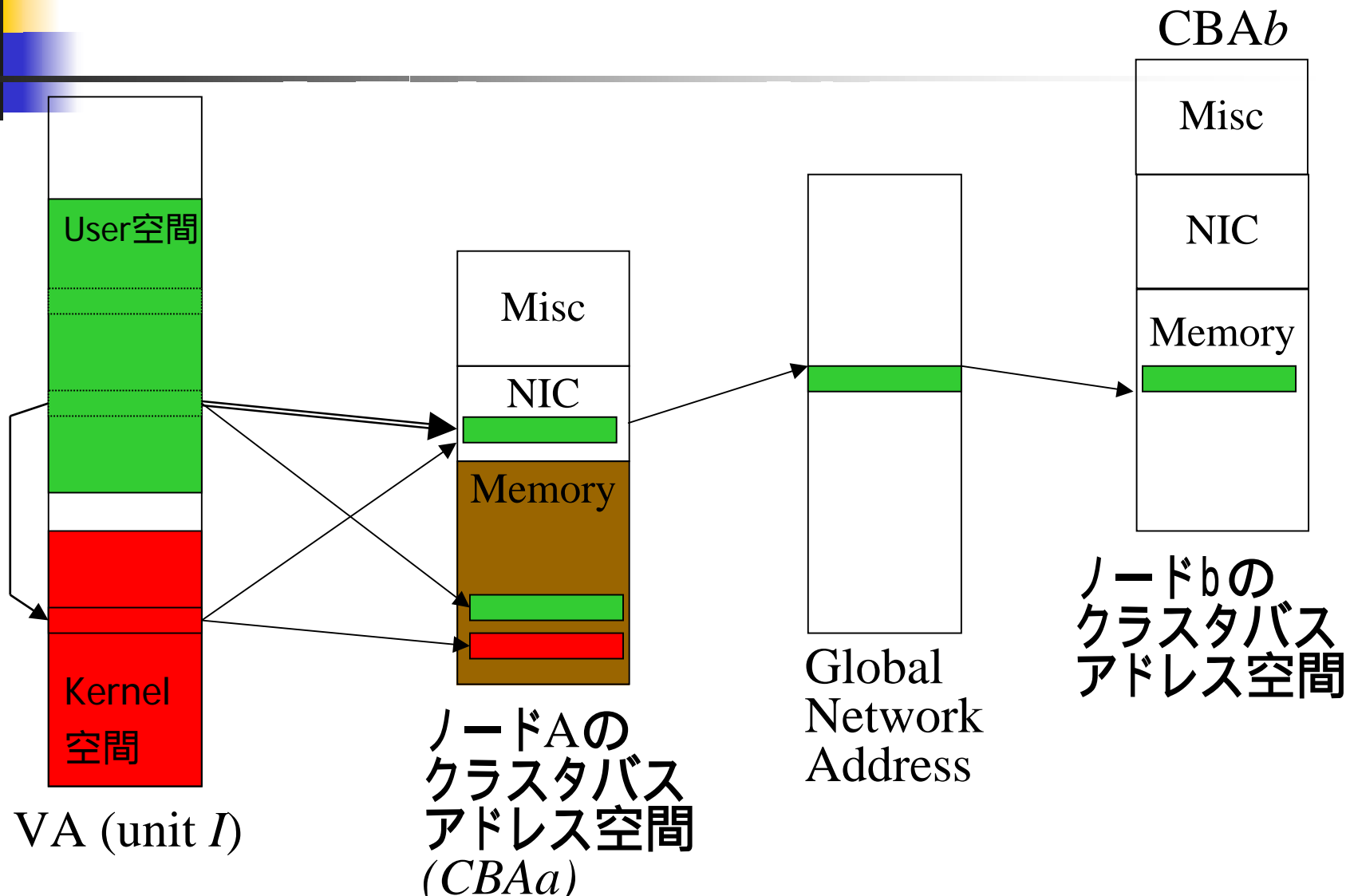


# 通信ソフトウェアの高速化

---

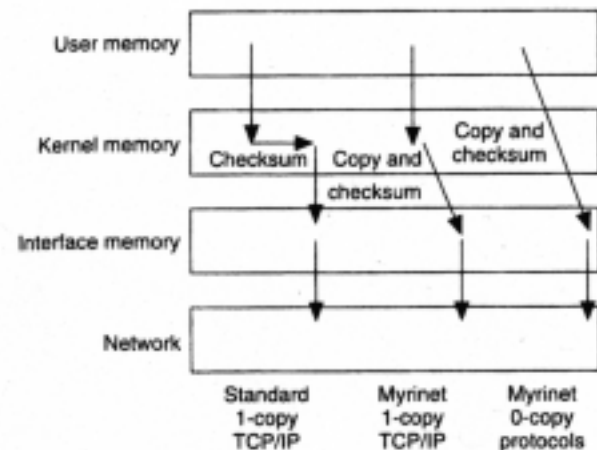
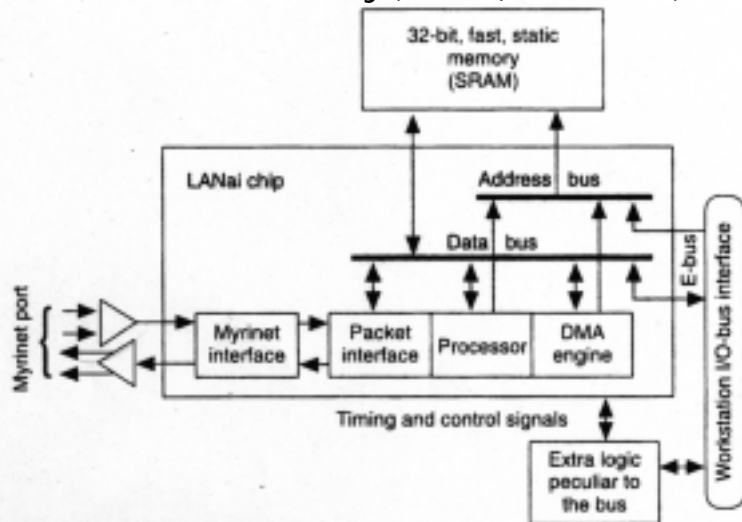
- 0コピー通信
- remote read/write (put/get)
- 軽量化プロトコル

# Zero-copy通信



# Myrinet

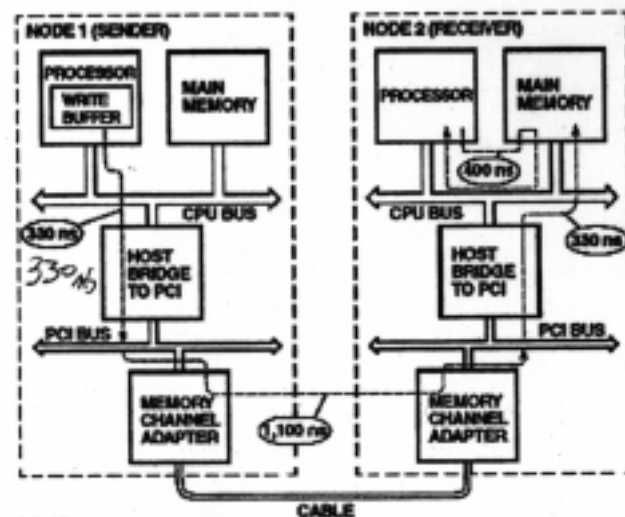
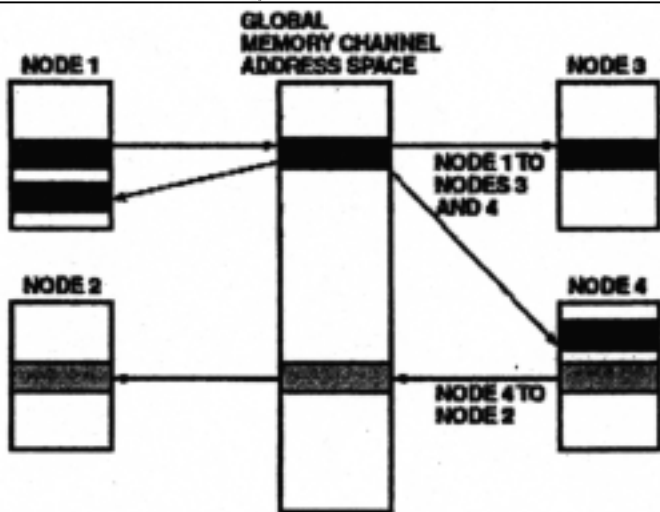
- USC/ISI+Caltecの共同開発(ATOMIC) の commercial version
  - 専用通信プロセッサ LANai
  - 高速パケット交換スイッチ (2Gbps + 2Gbps)@Myrinet2000
- Application Programming Interface や LANai chip 仕様の公開
  - 多くの Hacker達が競って最適化！！
  - UC Berkeley(NOW), Illinois(Fast Messages), RWCP(PM)



# Memory Channel

- 仮想共有メモリ環境を提供する PCI バス用通信ボード
- 異なるノード上のプロセス間通信が, load/store で実現可能

世代	転送速度 (ユーザプロセス間sustained)	遅延 (store to load)	ハブの構成 (cabling)
第一(1996 ~ )	77MB/s (66MB/s)	2.9 $\mu$ s	バス構造(電線4m)
第二(1997 ~ )	97MB/s (88MB/s)	2.2 $\mu$ s	スイッチ(電線10m or 光3Km)



クリストロイドなアドレスマッピング

Latency Contribution

# 汎用ネットワークの追い上げ

## 超並列計算機のネットワーク性能

製品名	メーカー	発表年	Link速度
CM-5	T.M.	1992年	1.06Gbps
Paragon	Intel	1992年	1.6Gbps
T-3D	Cray	1993年	1.2Gbps
T-3E	SGI	1995年	4.8Gbps
SR2201	日立	1996年	2.4Gbps
VPP700E	富士通	1997年	4.8Gbps

製品名	メーカー	発表年	Link速度
SR8000	日立	1998年	1.0GB/s
SX-5	NEC	1998年	8(?)GB/s
Cenju-4	NEC	1998年	0.8GB/s
VPP800	富士通	1999年	1.6GB/s
AsamA	NEC	2002年	12.8GB/s
地球Sim	NEC	2002年	12.3GB/s

## ギガビット級ネットワーク

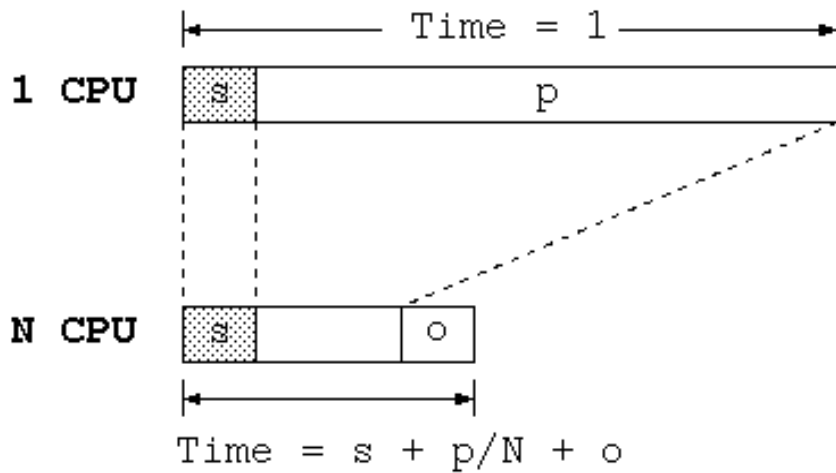
名称	企画/企業	Link速度	通信の信頼性@物理層
Fibre Channel	ANSI X3 T11	1.06Gbps	Reliable
SCI	ANSI/IEEE 1596	1.6Gbps	Reliable
Memory Channel2	HP (DEC)	1.06Gbps	Reliable
Myrinet	Myricom	1.26Gbps	Reliable
Gigabit Ethernet	IEEE 802.3z	1.25Gbps	Unreliable
Synfinity	Fujitsu	1.6Gbps	Reliable

# プロセッサ間通信の遅延

NIC	Latency	RATIO	I/F
DIMMnet-1	225ns(??)	1	PC133DIMM
ELAN	2us	10	66M64bPCI
Memory Channel	2.2us	10	33M32bPCI
SCI	2.3us	10	66M64bPCI
VIA on CLAN1000	3.5us	16	33M64bPCI
PM on Myrinet	7.5us	33	33M32bPCI
GM on Myrinet2000	7.6us	33	66M64bPCI
PM on GbE	24.1us	107	33M32bPCI
AsamAカスタムLSI	300ns	1	ItaniumII bus

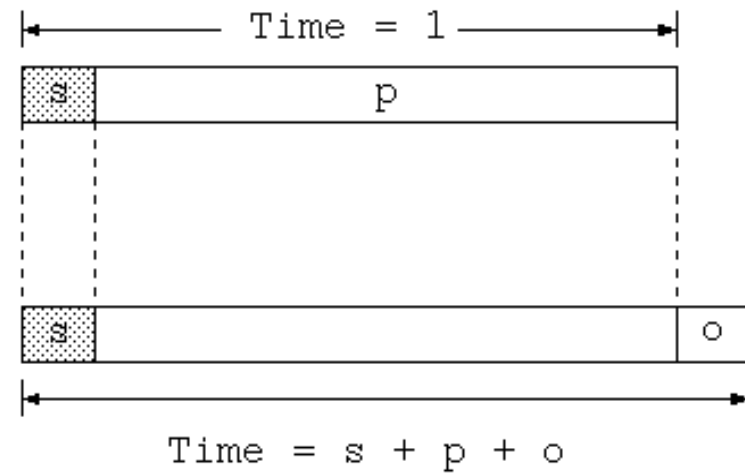


# スピードアップに関する2つの視点



$$\text{Speedup} = \frac{1}{s + p/N + o}$$

(a) 問題サイズ固定

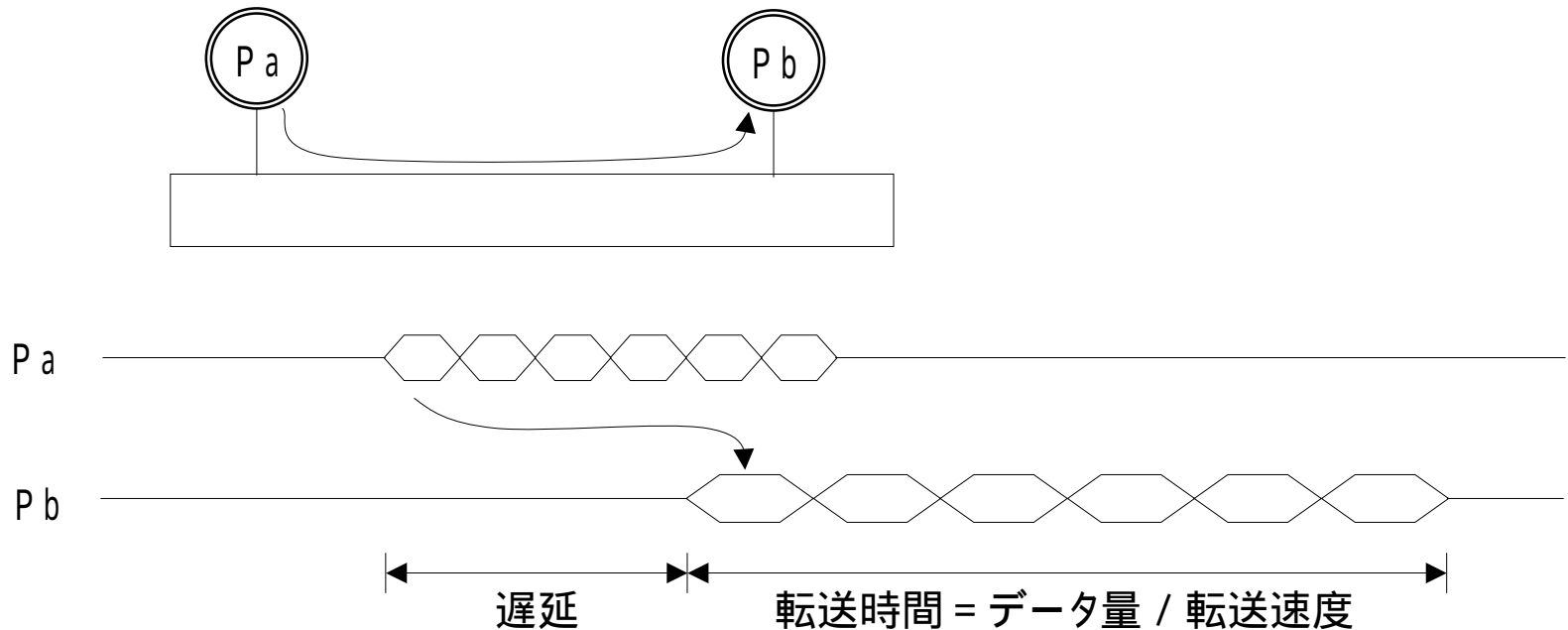


$$\text{Speedup} = \frac{1}{1 + o} \times N$$

(b) 問題サイズ N 倍

# 並列処理オーバヘッド

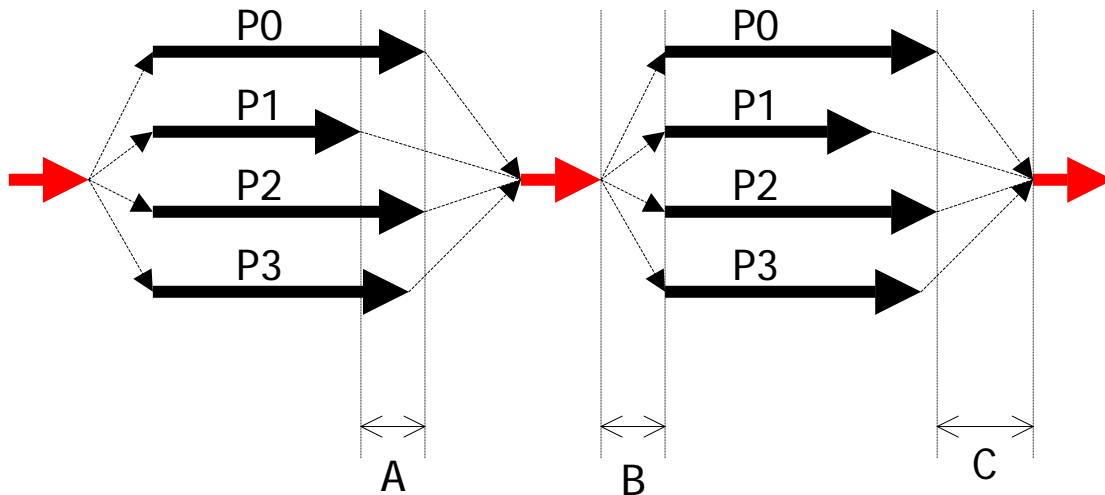
## (1) 通信オーバヘッド



[技術的背景] 転送速度の向上は著しいが、遅延時間の短縮は極めて困難

# 並列処理オーバヘッド

## (2) 負荷のアンバランスと同期オーバヘッド



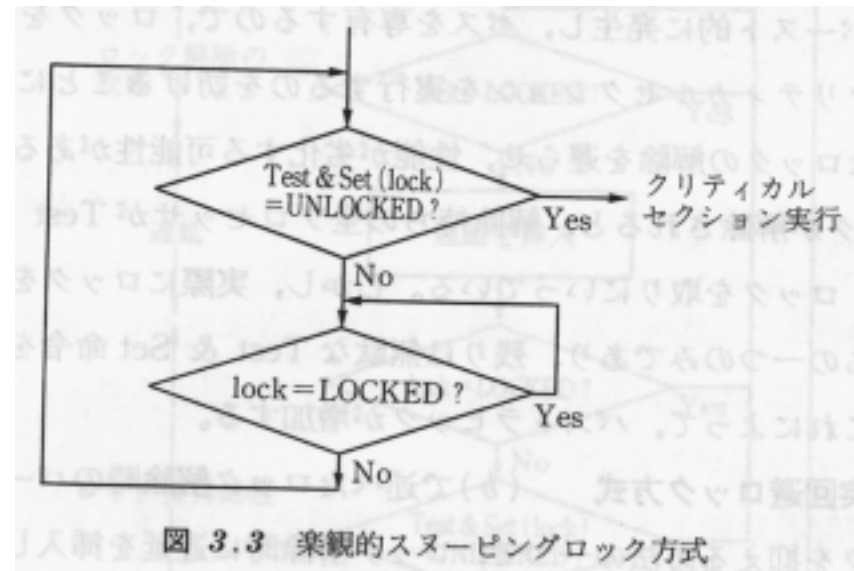
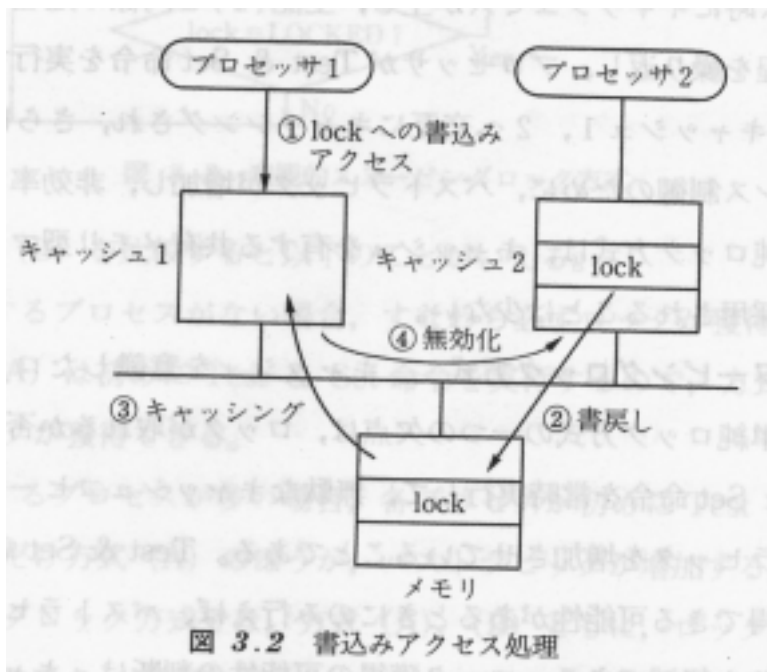
A: 負荷のアンバランスによるオーバヘッド

B + C: 同期オーバヘッドの例

# 同期操作のオーバヘッド削減

## スピンロック

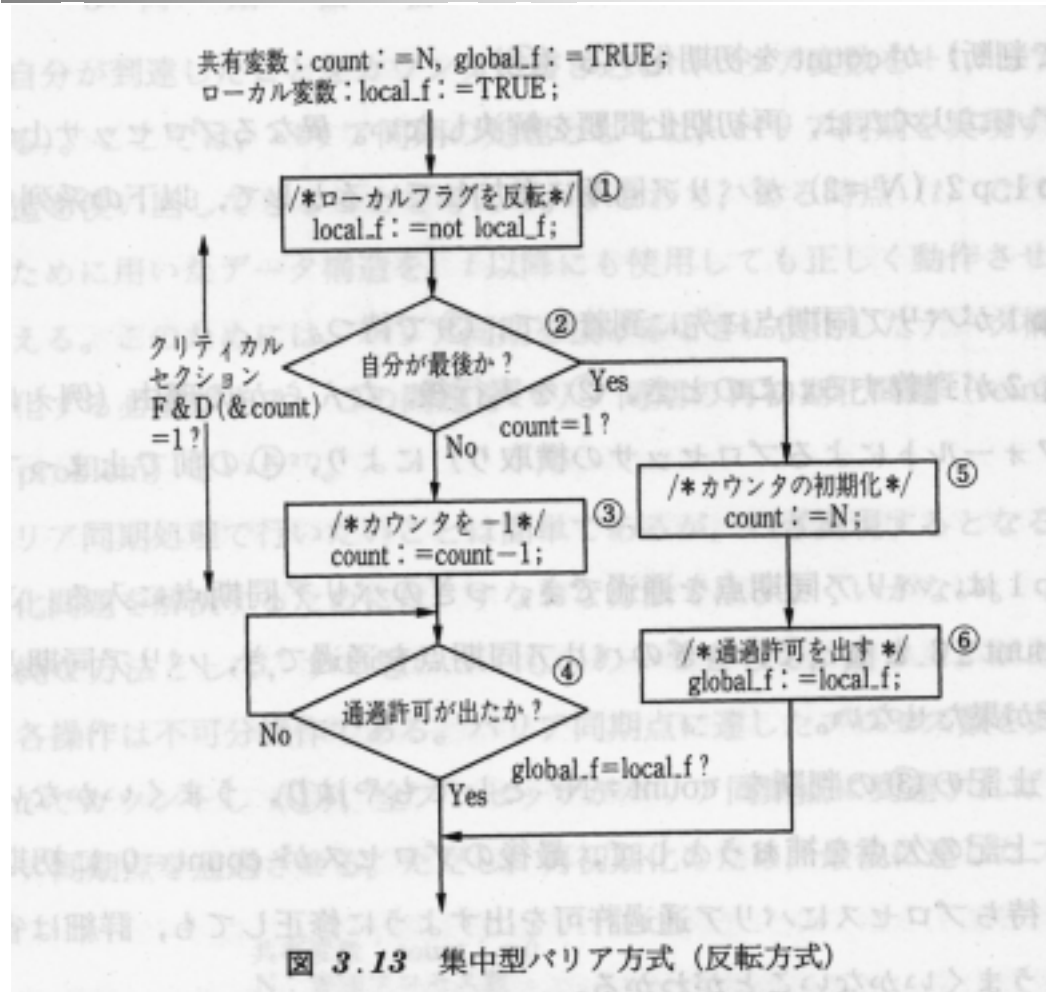
- Test and Test&Set.....同期待ちトラフィックの軽減



# 集中型バリア同期機構の例

カウンタベース

再利用可能



# 分散バリア同期機構の例

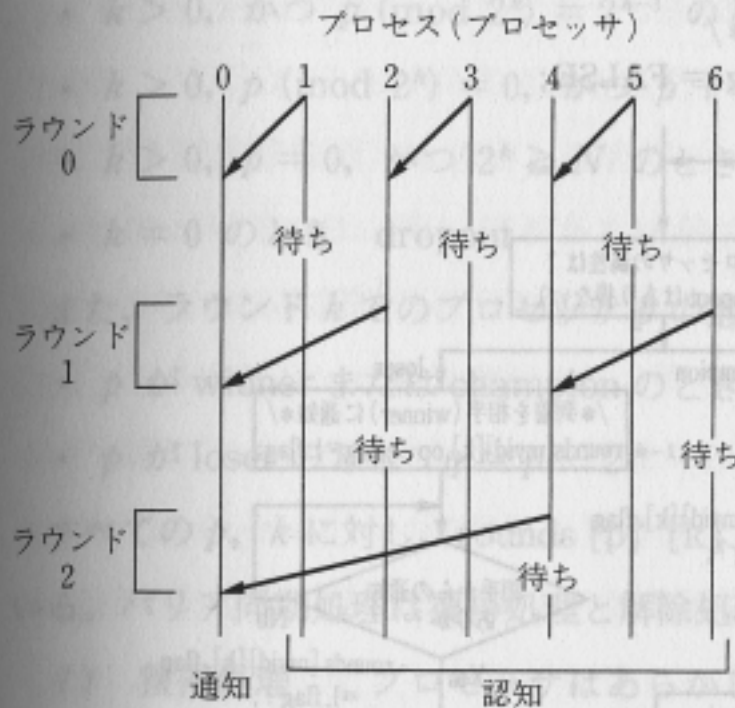


図 3.16 トーナメントバリア方式における通信パターン例 ( $N=7$ )

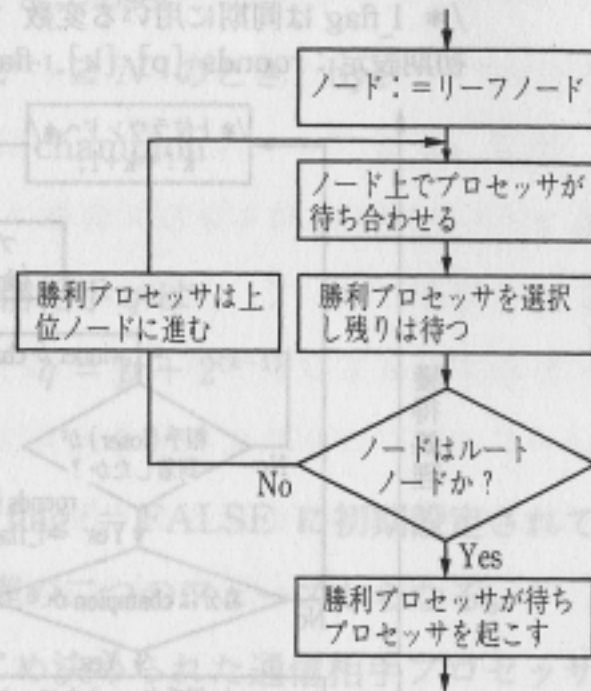


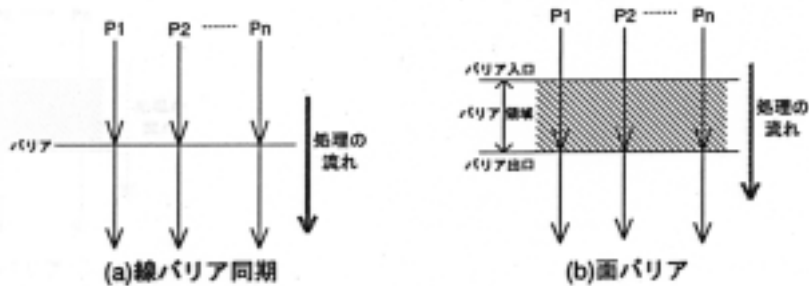
図 3.17 トーナメントバリア方式の概要

# バリア同期モデルの分類

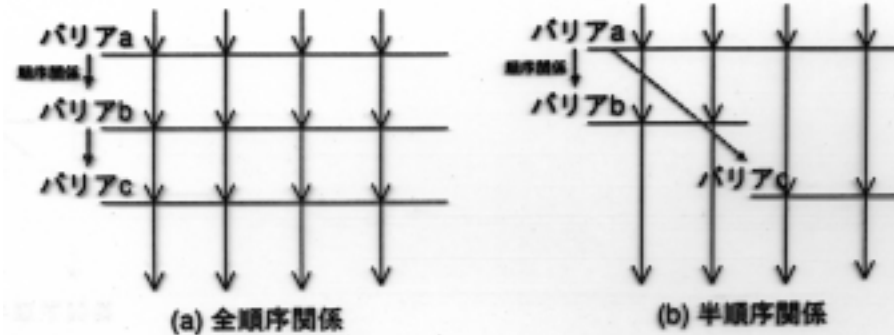
分類項目				バリア同期モデル
線 / 面	強制参加 / 任意参加	全順序関係 / 半順序関係	オーバラップ 不可能 / 可能	バリア同期モデル
線	強制参加	全順序関係	不可能	(普通の)バリア
	任意参加	全順序関係		SBM
		半順序関係		HBM, LDBM, FDBM
面	強制参加	全順序関係	不可能	Fuzzyバリア
			可能	重複可能バリア
	任意参加	全順序関係	不可能	文献[104]で提案
			可能	Elasticバリア
		半順序関係	不可能	文献[104]で提案
			可能	文献[104]で提案

# バリア同期モデルの分類

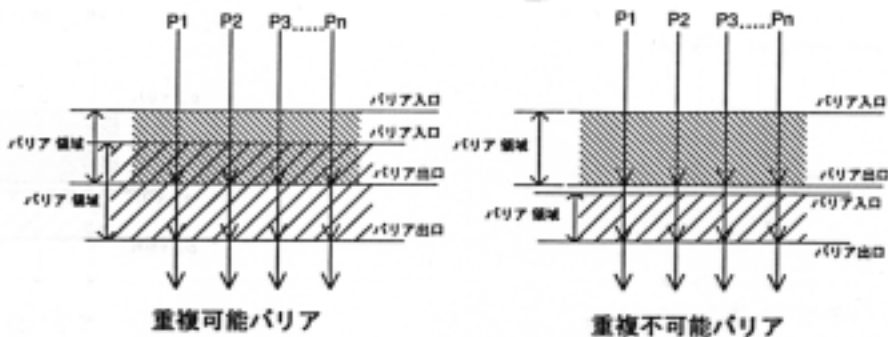
## 1) 線バリア vs. 面バリア



## 2) 全順序 vs. 半順序



## 3) 重複可能 vs. 重複不可能



## 4) 強制参加 vs. 任意参加





# 負荷の不均衡の解消

## ■ 動的負荷分散方式

### ■ Doall型の例

- Self-Scheduling, Chunk Scheduling, Guided Self-Scheduling

### ■ 反復計算向けの負荷分散アルゴリズム

- N回目の計算時の実行時情報からN+1回目の計算時の最適な処理の分割を予測する。

非均質な計算環境への適用可(heteroTINPAR by 富田研)

### ■ OSレベルでの対応

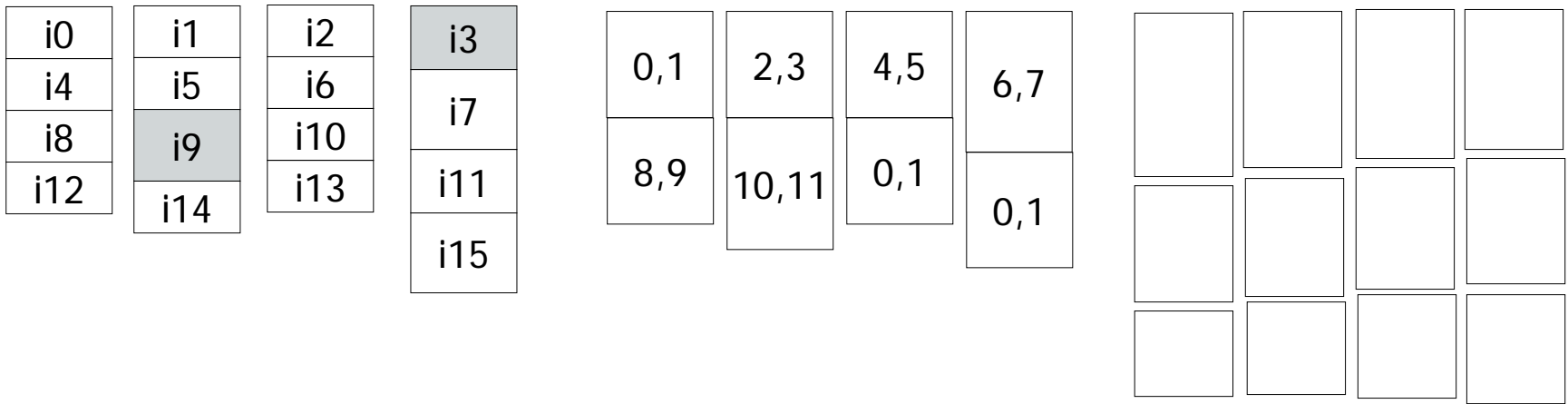
- Cache Affinity Scheduling (ラストプロセッサ方式、最小介入数方式, 等)
- Memory Affinity Scheduling
  - 1)ホーム, 2)コピー, 3)Minimum Load Cluster, 4)Other
- ギャングスケジューリング/Co-scheduling

### ■ その他.....HTGを利用した実行時粒度制御機構

# 負荷の不均衡の解消

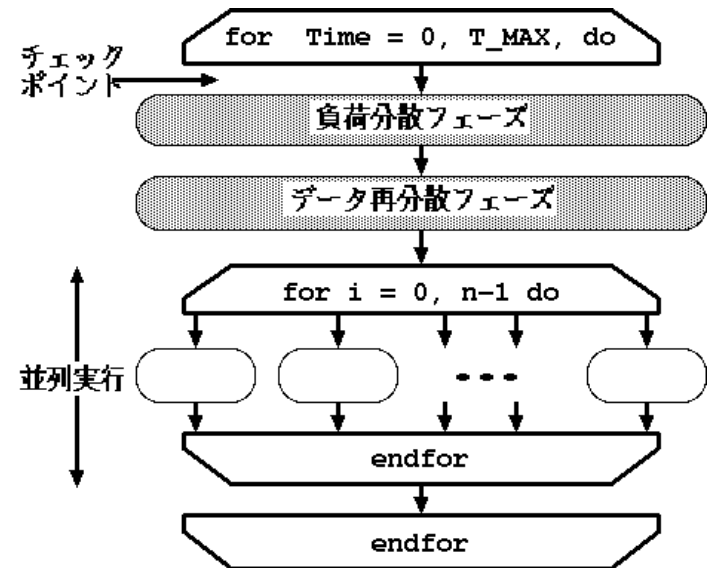
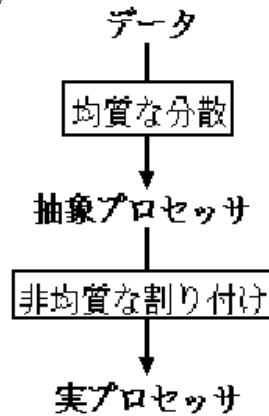
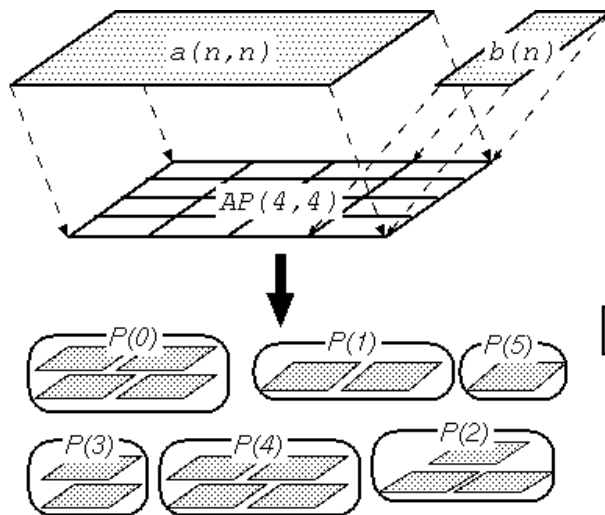
## (広義の)Self-scheduling方式

- Self-Scheduling, Chunk Scheduling, Guided Self-Scheduling



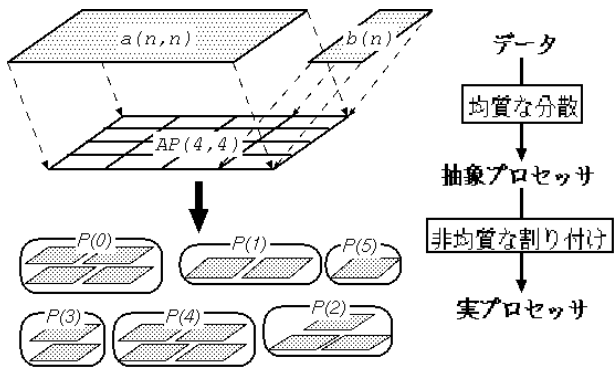
# 非均質環境における動的負荷分散の例 (heteroTINPAR)

```
While(収束条件成立){  
  for l = 0, n-1 do{  
    .....  
  }  
}
```

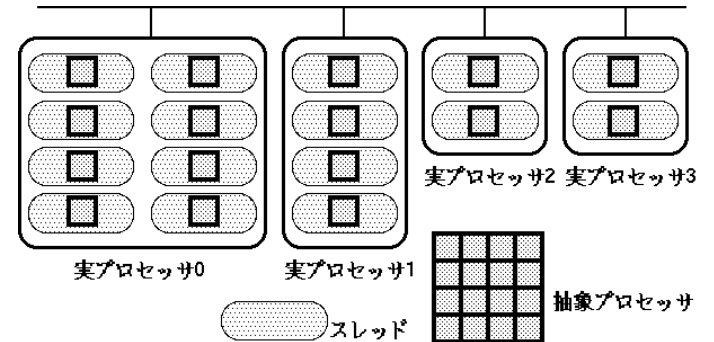


# 非均質環境における動的負荷分散の例 (heteroTINPAR)

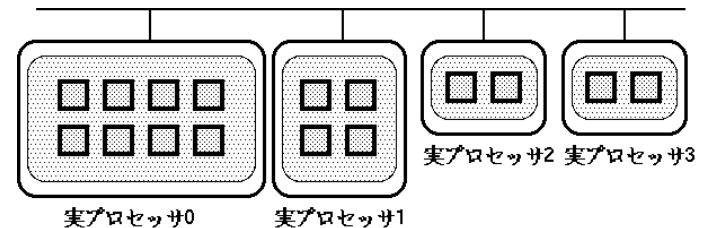
実プロセッサにおける  
複数抽象プロセッサの処理方式



## マルチスレッド方式



## シングルスレッド方式



# 非均質環境における動的負荷分散の例 (heteroTINPAR)

## 動的負荷分散による実行時間の変化

評価環境: SS20x2台 + Ultra1 x 2台

評価プログラム:

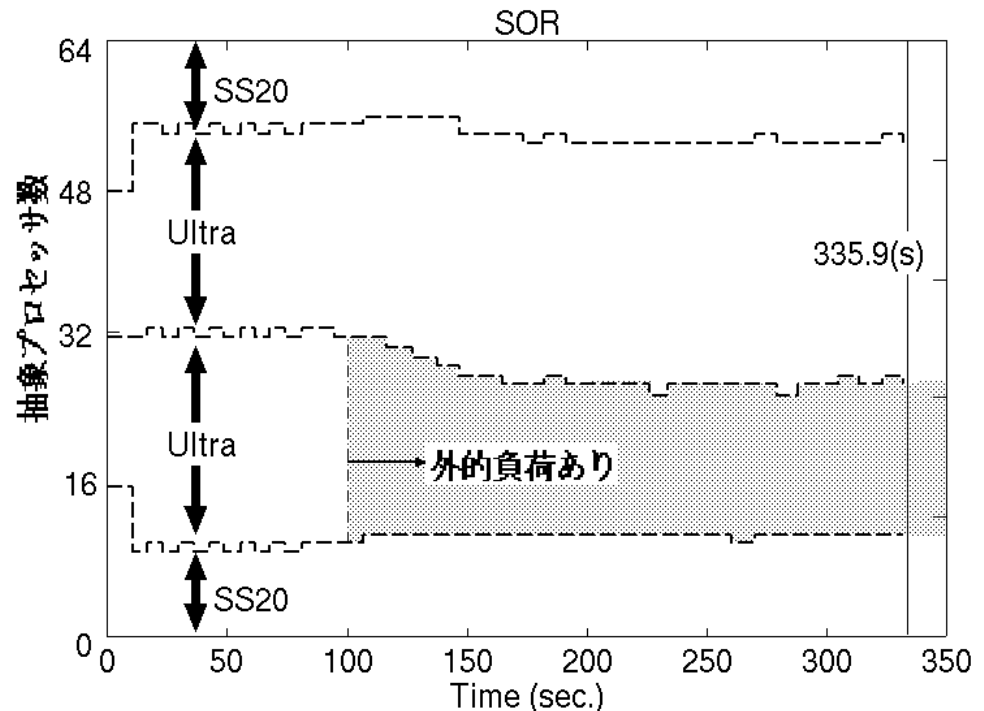
SOR(1024x1024, 2000反

復)

抽象プロセッサ数:64

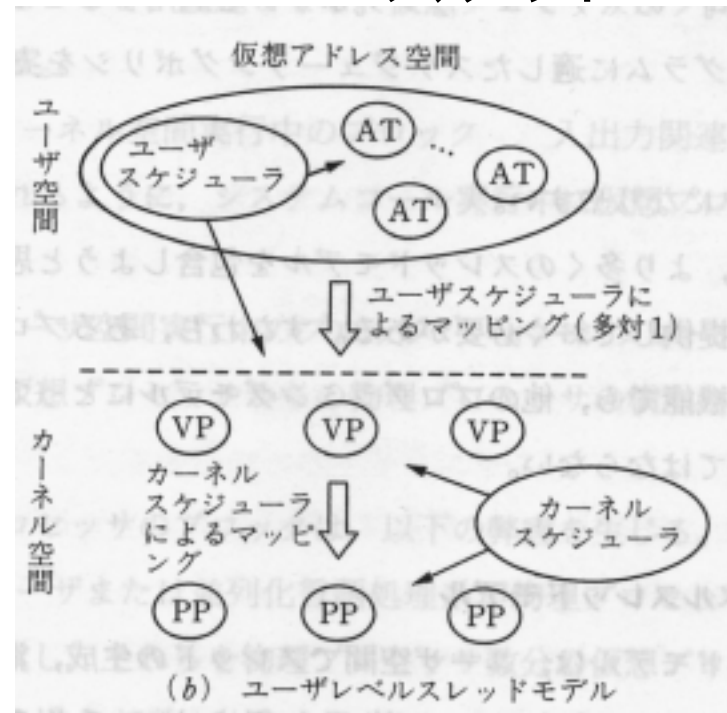
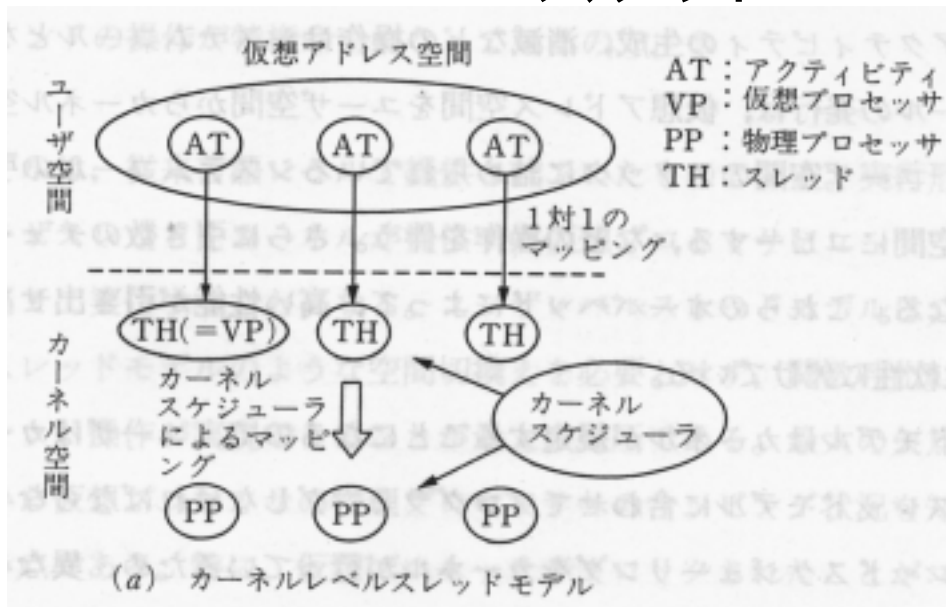
実行開始100秒後に一方のUltra1に負荷をかける

負荷分散なしの実行時間 = 357.6(s)



# マルチスレッド環境

## Kernel Level スレッド vs. User Level スレッド



[HW実装(余談)] Simultaneous Multi-threading(SMT) ...Intel用語ではHT

# SSI クラスタ

- Complete Transparency.... Shared Address Space
- Scalable Performance..... Small SSI Overhead
- Enhanced Availability..... Fault Tolerant

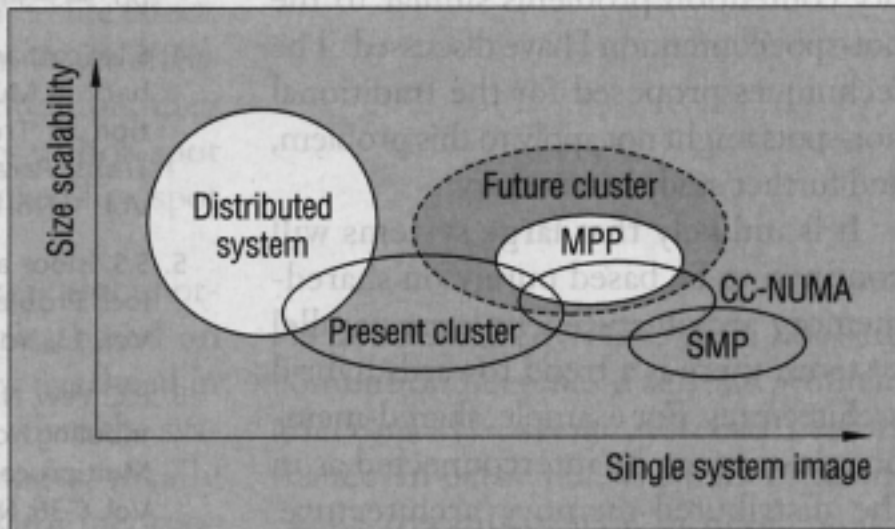


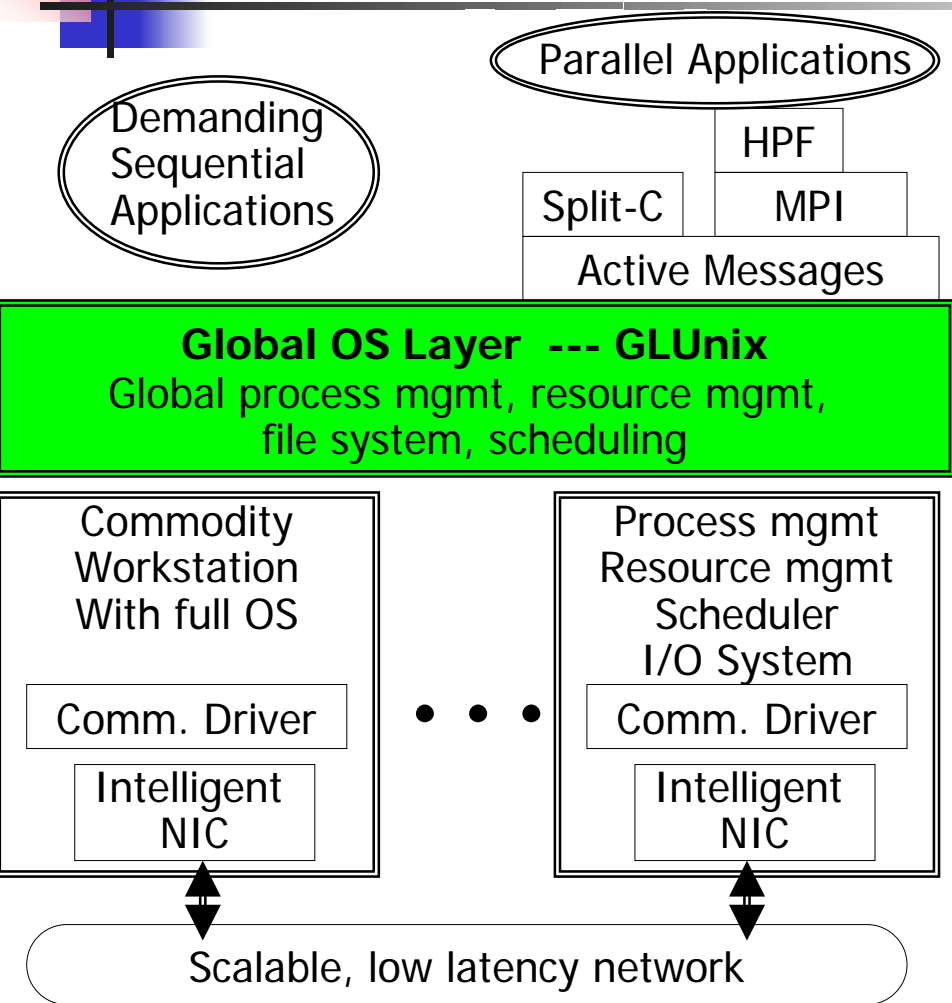
Figure 1. Design space of competing computer architectures.

Designing SSI Clusters with Hierarchical Checkpointing and Single I/O Space

Kai Hwang, Hai Jin, Edward Chow,  
Cho-Li Wang, and Zhiwei Xu

IEEE Concurrency, Vol.7, No.7, pp.60-69, January-March 1999

# UC Berkley “NOW”



## GLUnix

UNIX のシステムコールを横取り  
SSI環境を実現

## xFS ....Serverless FS

Network RAID(Striping)

Cooperating Caching

- 1) Clientの主記憶をFile Cacheとして使用
- 2) MP用のCC機構を利用してFileを管理



# RAID:

## Redundant Array of Inexpensive Disks

	RAID1	RAID2	RAID3	RAID4	RAID5	RAID0
ストライピングの有無	無	有	有	有	有	有
ストライピングの単位	---	bit	bit	block	block	block
故障検出単位	Disk	数bit	Disk	Disk	Disk	Disk
データ復元方法	2重化	ハミング符号	パリティ	パリティ	パリティ	---
冗長ディスク構成	固定	固定	固定	固定	分散	---

(striping)

DISK0	DISK1	DISK2	DISK3	DISK4
P0(B0-B3)	B0	B1	B2	B3
B4	P1(B4-B7)	B5	B6	B7
B8	B9	P2(B8-B12)	B11	B12

**RAID5の構成例**

# 近くのハードディスク vs 遠くのメモリ

## ■ HDDの転送速度

- SCSI系 ~ 320MB/s (ULTRA SCSI320)
- Fibre Channel 1.06 ~ 4.24Gbps
- IEEE1394系 ~ 400Mbps

## ■ HDDのレイテンシ

- 数ミリ秒のオーダー (DISKの回転速度が支配)



- ネットワークの転送速度 ..... 1 ~ 10Gbps
- ネットワークのレイテンシ ..... 数十マイクロ秒