はじめに

平成23年9月1日

トップエスイープロジェクト

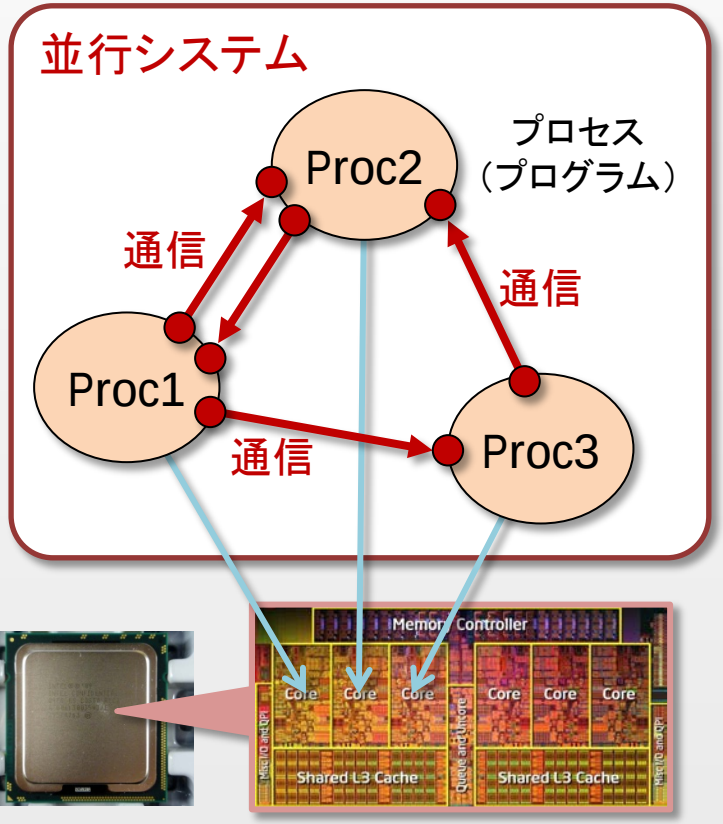
磯部祥尚 （産業技術総合研究所）

本講座の背景と目標

- 背景:
  マルチコアCPUやクラウドコンピューティング等、
  並列/分散処理環境が身近なものになっている。
  - ✓ 複数のプロセス(プログラム)を同時に実行可能。
  - ✓ 通信等により複数のプロセスが協調可能。
    ⇒ 並行システムの構築

- 並行システムの長所と短所:
  - ✓ 長所: 協調動作による処理の高速化・分散化
  - ✓ 短所: 協調動作による設計の複雑化
    ⇒ システムの信頼性の低下
    (デッドロック、リソース競合等)

- 本講座の目標:
  並行システムの検証と実装方法を習得する。
  (信頼性の高い並行システムを構築するため)



例: 6コアCPU Core i7-980X ([1]より写真を転載)

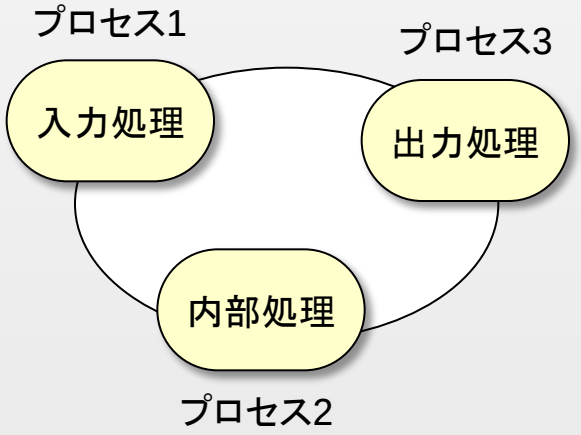
[1] http://journal.mycom.co.jp/special/2010/gulftown/index.html

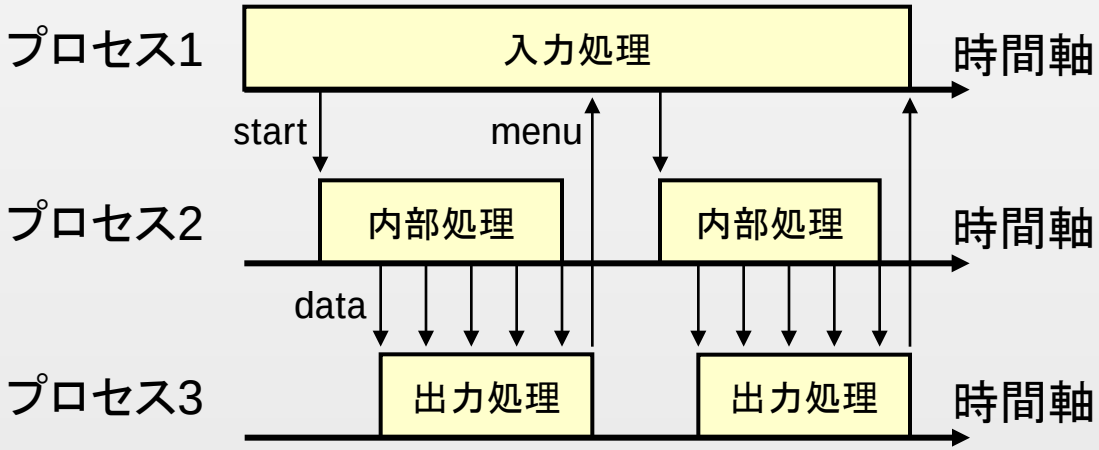
はじめに

- 並行システムの概論
- JCSP, FDR, CSPとは
- 本講座の特長

並行システムの特徴（並行処理）

- 入力処理プロセスで常に入力を受け付けつつ、
- 内部処理プロセスで入力操作に応じて内部処理を実行し、
- 出力処理プロセスで内部処理中にその進捗状況を表示する。

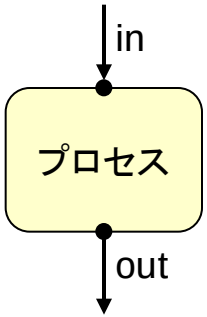


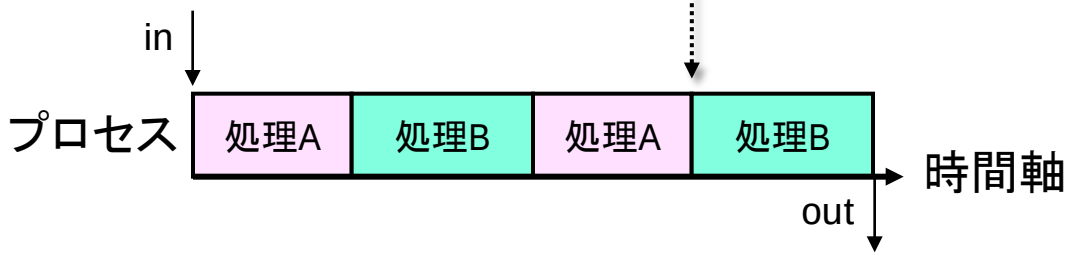


並行システムの特徴（高速処理）

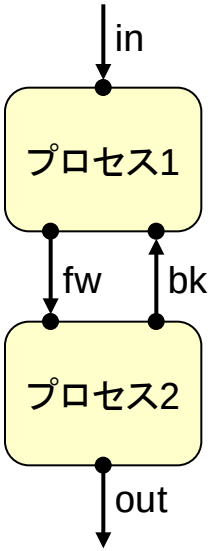
A, B: 互いに依存関係があり、同時に実行する必要のない2つの処理

逐次処理

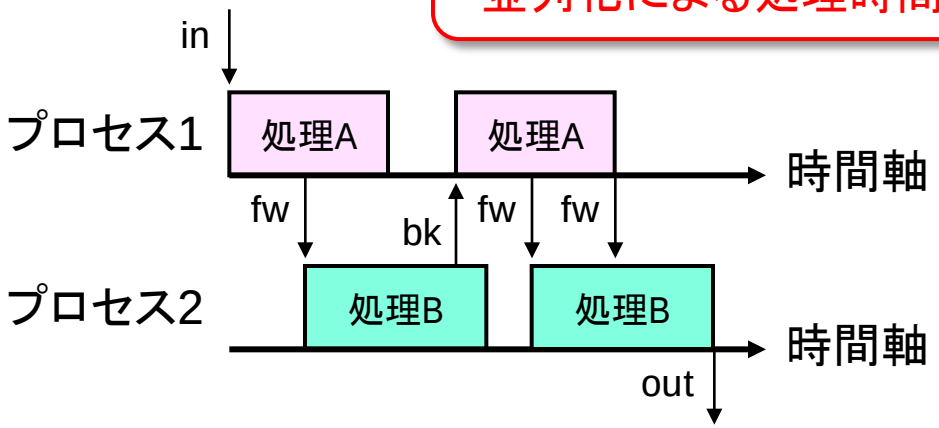




並列処理

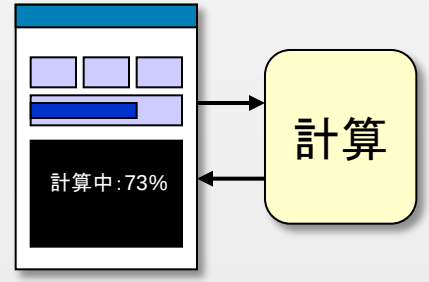


並列化による処理時間の短縮

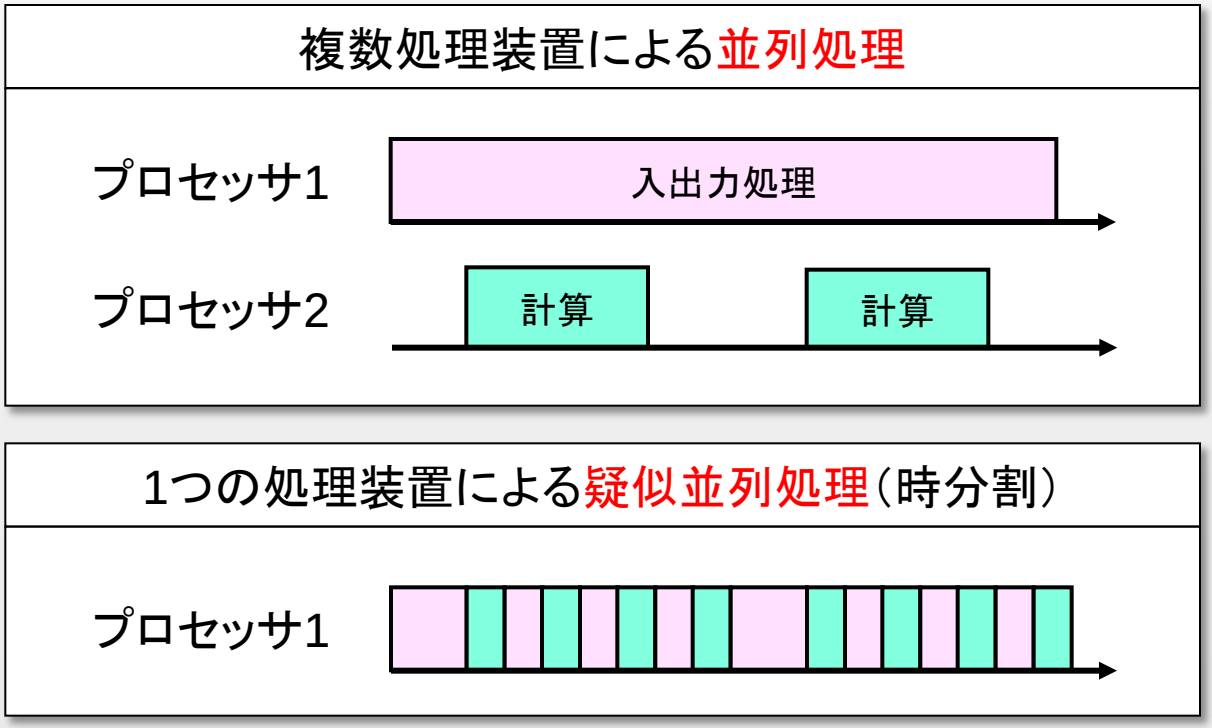


“並列処理”と“並行処理”の違い

- 並列処理：複数の処理を同時に実行すること。
- 疑似並列処理：複数の処理を時分割で実行すること。
- 並行処理：複数の処理を時間的重なりをもって実行すること。



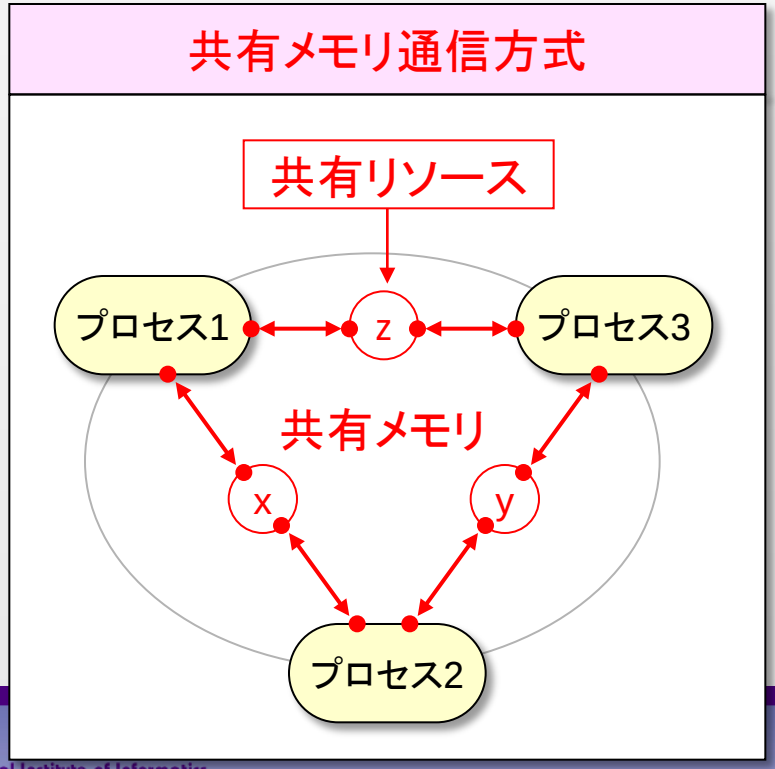
ユーザインターフェース

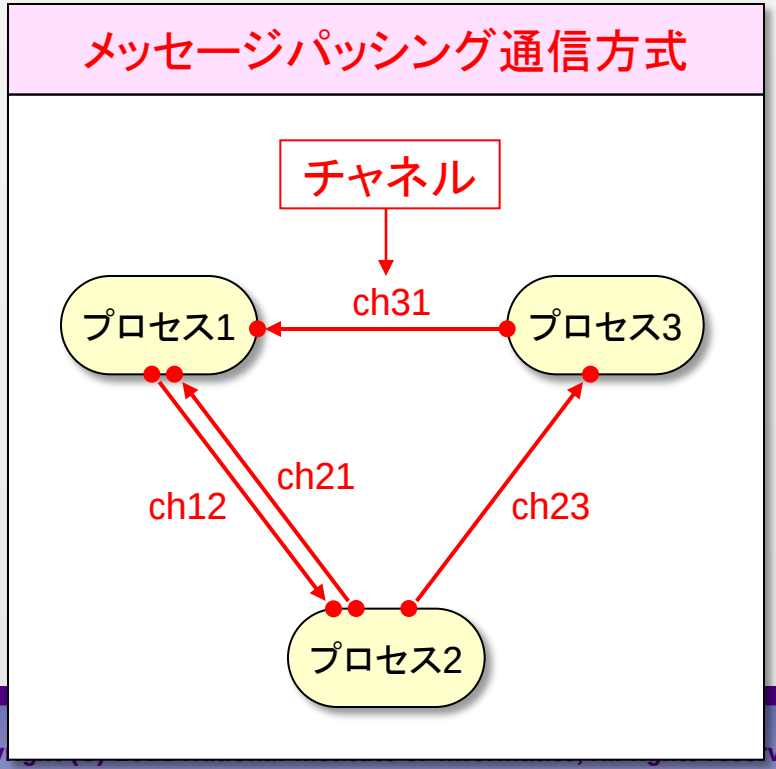


並行処理の意味は並列処理と疑似並列処理の両方を含む

2つの通信方式

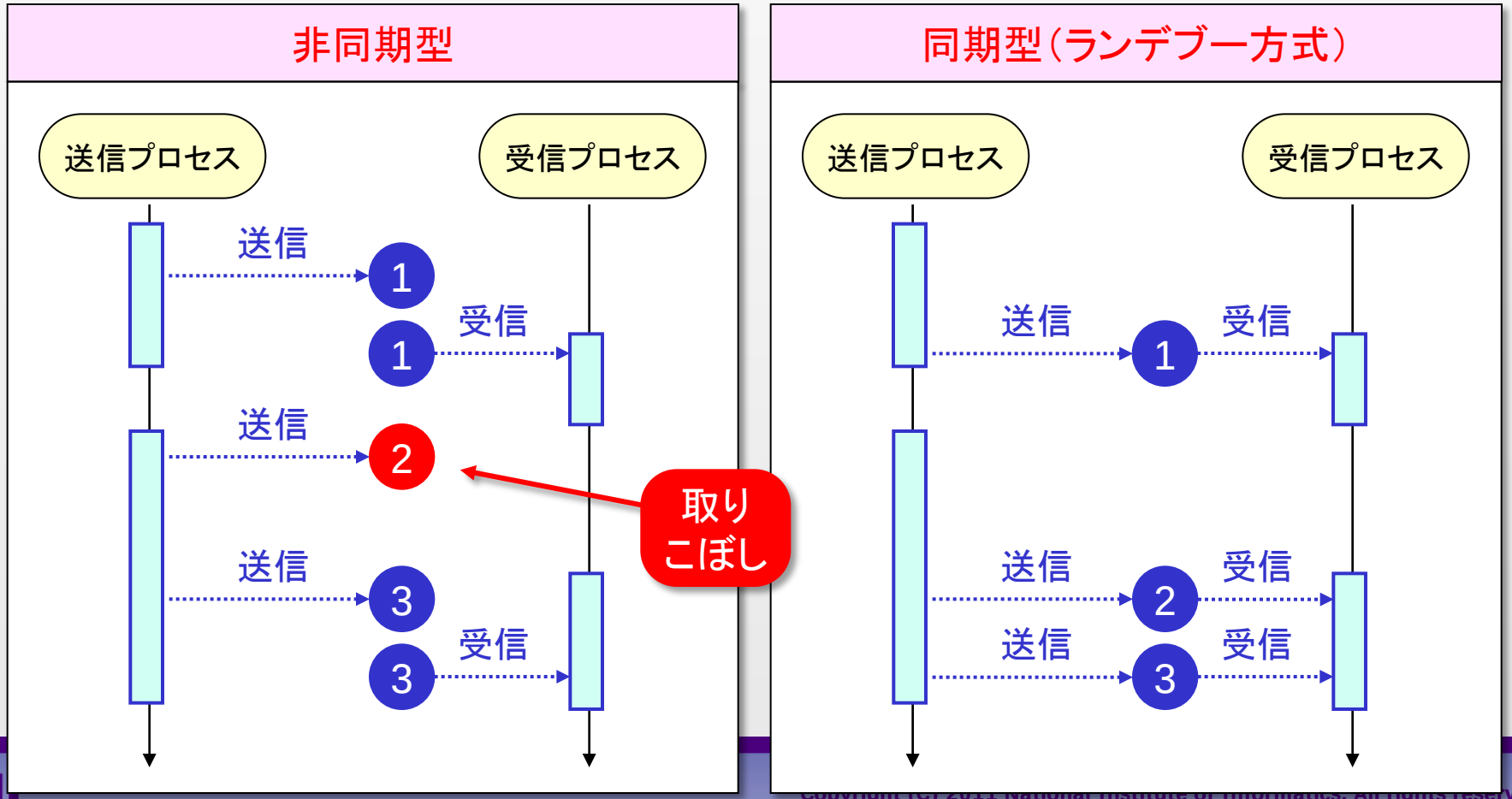
- 共有メモリ通信方式： 共有メモリ上でメッセージを交換
  ⇒ 処理が軽い（早い）
- メッセージパッシング通信方式： チャネルを通してメッセージを送受信
  ⇒ 動作が分かりやすい





メッセージパッシング通信方式

- 非同期型：送信と受信は別々に起きる ⇒ 処理が軽い（早い）
- 同期型：送信と受信は同時に起きる ⇒ メッセージの取りこぼしがない。



はじめに

- 並行システムの概論
- JCSP, FDR, CSPとは
- 本講座の特長

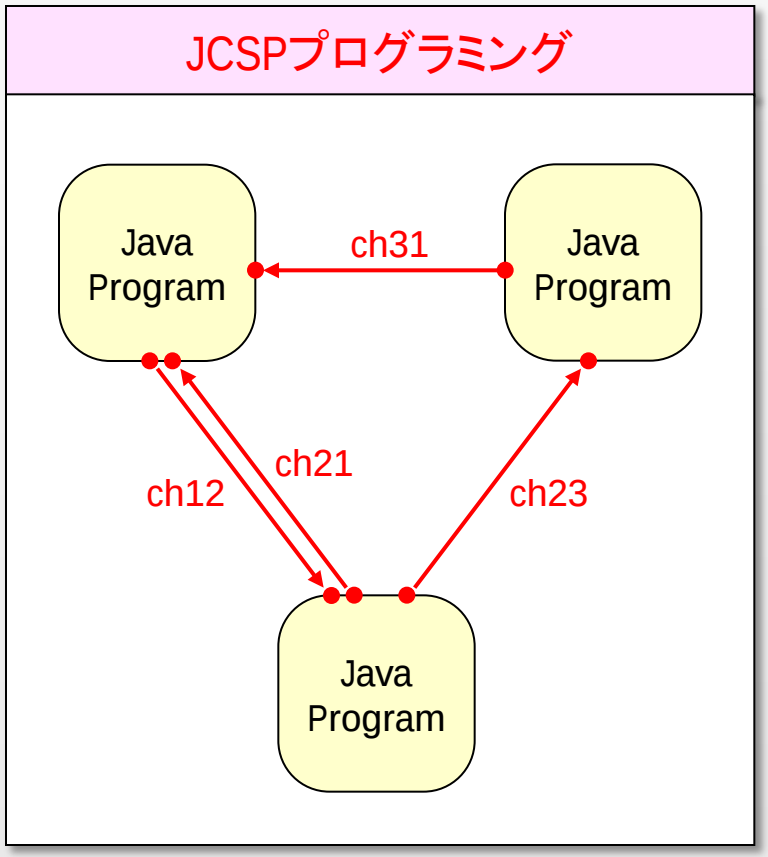
Javaライブラリ: JCSP

- JCSP: Java で 同期型メッセージパッシング通信を実装するためのライブラリ

新しい言語を覚えなくてよい
（敷居が低い）

比較的その動作を予測しやすい
（信頼性を高められる）

信頼性の高い並行システムを構築できる！



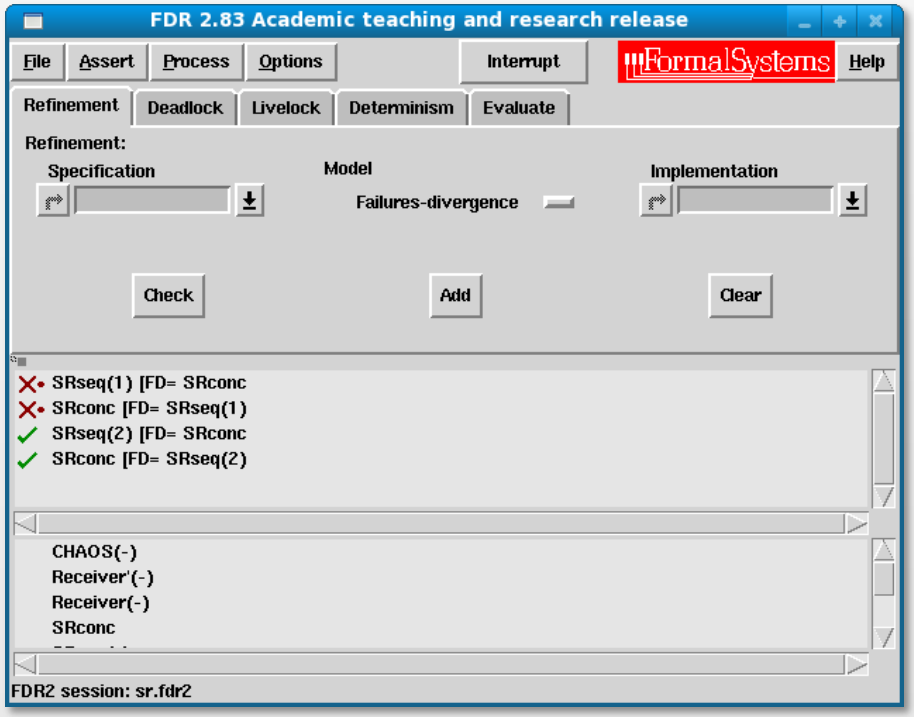
モデル検査器：FDR

- FDR：JCSPの 同期型メッセージパッシング通信を検証できるツール

並行動作の正しさを判定できる
（期待通りに動作するか）

さらに信頼性の高い並行システム
を構築できる !!

FDRによる検証画面



プロセス代数：CSP

- なぜJCSPの 同期型メッセージパッシング通信をFDRで検証できるのか？

  ⇒ JCSP：プロセス代数CSP のモデルを実装するためのJavaライブラリ
  FDR：プロセス代数CSP のモデルを検査するためのモデル検査器

CSP：同期型メッセージパッシング通信方式を採用しているプロセス代数

プロセス代数：並行システムを形式的に記述し、解析するための理論

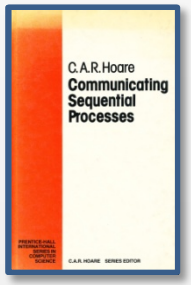
プロセス代数CSPでモデル化し、モデル検査器FDRで検証し、Javaライブラリで実装することによって、理論的に検証された高信頼な並行システムを実装できる !!!

はじめに

- 並行システムの概論
- JCSP, FDR, CSPとは
- 本講座の特長

本講座の3つのキーワード

理論 CSP → 検証 FDR → 実装 JCSP



Communicating
Sequential
Processes,
Hoare, 1985

プロセス代数 CSP:
並行システムを記述・解析するための理論

モデル検査器 FDR:
CSPモデルの詳細化関係を検証するツール

Oxford大学、Formal Systems
（アカデミックライセンスは無料）

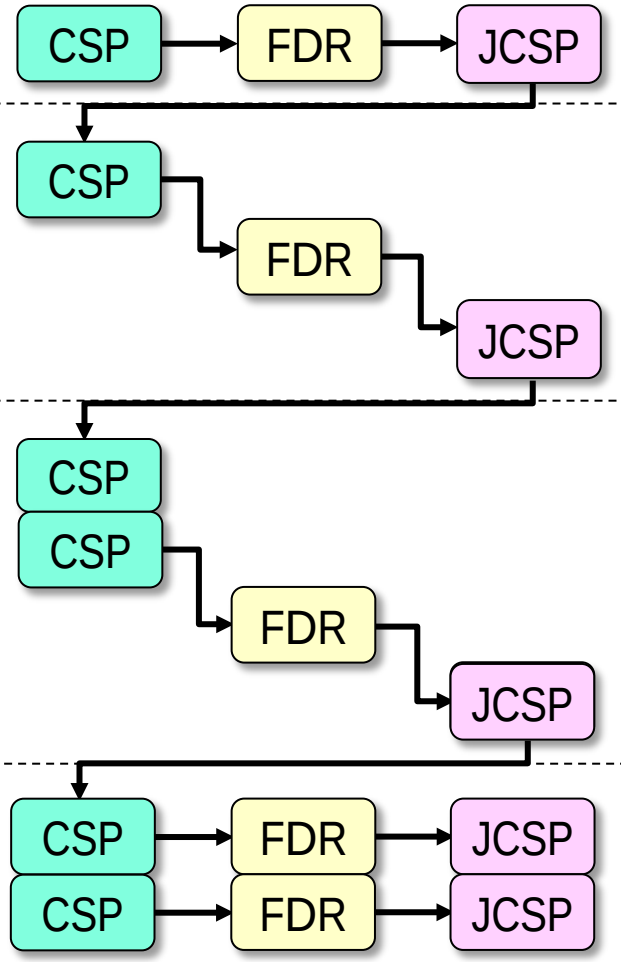
Javaライブラリ JCSP:
CSPモデルをJavaで実装するためのライブラリ

Kent大学
（2006年秋からオープンソース）

CSPモデル	システムのCSP記述（形式的な記述）
FDRスクリプト	FDRで読込み可能な記述（FDR入力言語）
JCSPプログラム	JCSPを利用しているJavaプログラム

本講座の構成

		講義予定
概要 レベル1	第1章　CSP, FDR, JCSP概論	1,2
入門 レベル2	第2章　CSP入門 第3章　FDR入門 第4章　JCSP入門	3,4 5 6,7
基礎 レベル3	第5章　CSP理論（動作表現） 第6章　CSP理論（動作解析） 第7章　FDR検証 第8章　JCSP実装	8 9 10 11
応用 レベル4	第9章　CSP, FDR, JCSP応用 第10章　CSP, FDR, JCSP実践	12 13-15 グループ実習



習得する知識＆技術

- プロセス代数CSPの基礎知識：

  CSPによるモデル化、FDRによる検証、JCSPによる実装の方法を正しく理解するために必要な基礎知識の学習。

- モデル検査器FDRによる検証技術：

  オートマトン（クリプキ構造）と時相論理に基づくモデル検査器（SPIN, SMV等）とは一味違った、プロセス代数に基づくモデル検査器FDRによる検証方法の習得。

- ライブラリJCSPによる並行プログラミング技術：

  CSPモデルに基づく並行プログラミング（分散プログラミングを含む）技術の習得。

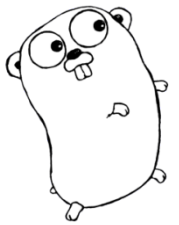
（実装を意識したモデル化が重要、検証には理論の基礎知識が必要）

Java以外にも有効

- JCSP (Java) 以外にもCSPモデルを実装するためのライブラリがある。基本的な考え方は同じで、本講座で習得する実装技術は他のプログラミング言語にも有効である。

ライブラリ	言語	研究開発元
JCSP	Java	ケント大学（QuickStone）
C++CSP	C++	ケント大学
CHP	Haskell	ケント大学
PyCSP	Python	トロムソ大学＆コペンハーゲン大学
Python-CSP	Python	ウォルバーハンプトン大学

- Go言語: 2009年秋にGoogleが発表したプログラミング言語
  - ✓ コンパイル言語のように速く、スクリプト言語のように使い易い。
  - ✓ CSPに基づく並列処理記述が言語レベルで可能。



Go言語の公式マスコットGordon
http://golang.org/

ソフトウェア以外にも…

- Transputer （1981年～1996年、INMOS社)
  - ✓ 並列コンピューティングに特化した汎用マイクロプロセッサ
  - ✓ CSPに基づく実装言語 Occamによるハードウェア実装

Transputer[1](Inmos社)

- XMOS （2008年～、XMOS社)
  - ✓ イベント駆動型マルチスレッドプロセッサ（XS1-G4, 4コア, 32スレッド）
  - ✓ CSPに基づくC言語風の実装言語XCによるハードウェア実装

XMOS[2](XMOS社)

[1] ウィキペディアの「トランスピュータ」（http://ja.wikipedia.org/wiki/トランスピュータ）の写真から引用
[2] XMOS社のウェブサイト(https://www.xmos.com/products)の写真から引用

まとめ

- CSPの基本アイデアはC. A. R. Hoareによって1978年に発表された。
- しかし過去の話ではなくCSPの理論研究は今も続けられている。
- そして最近はCSPの実装関係の研究もより活発になっている。

  ⇒ マルチコアCPUの登場により並行システムの重要性が高まっている。
  （誰でも簡単に並列プログラミング）

  ⇒ ネットワークの普及によって分散プログラミングのニーズも高まっている。

CSPの理論、検証、実装の基本的な流れを学習する意義は高い。

補足

- 目標：プロセス代数による検証と実装の“考え方”を習得すること。

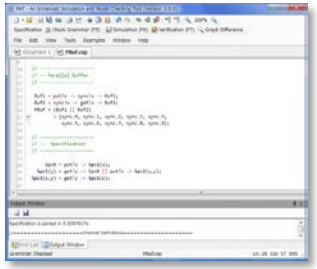
（FDRやJCSPの“使い方”を習得することは目標ではない）

⇒ “考え方”を習得すれば他のツールも使いこなせる。

プロセス代数CSP用モデル検査器の例

ツール名：FDR（Failures-Divergence Refinement）
開発元：オックスフォード大学、Formal Systems
特徴：CSPの代表的なツール。データ表現力が高い
URL：http://www.fsel.com/
備考：アカデミック＆研究目的でのみ無料

講義で使用



ツール名：PAT（Process Analysis Toolkit）
開発元：シンガポール大学
特徴：検証能力が高い。CSPの変更・拡張あり。
URL：http://www.comp.nus.edu.sg/~pat/
備考：アカデミック＆研究目的でのみ利用可能

FDRを使えれば…