KEPCO MPS

モデル検査のデバッグへの適用とモデル化作業支援

JaSST '06 in OSAKA　2006年5月11日

関西電力株式会社　電力技術研究所
篠崎孝一　太田 弘

メルコ・パワー・システムズ株式会社　技術統括部
星野 光勇　早水 公二

KEPCO MPS

はじめに

ソフトウェアシステムへの信頼性、安全性の要求

IEC61508(JIS C 0508)
機能安全規格
SIL=4にて要求

情報システムの信頼性向上に関するガイドライン(案)
2006.4.4　経済産業省

数理論理学を使ってソフトウェアを厳密に検証

形式手法 (形式的仕様記述 ・ 形式検証)

モデル検査
(Modelchecking)

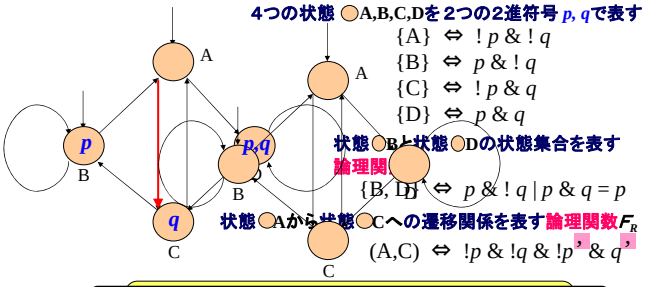
KEPCO MPS

システムのモデル化

システムの動き → 状態集合と遷移関係 → 論理関数

4つの状態 ○A,B,C,Dを2つの2進符号 p, q で表す

{A} ⇔ $!p \,\&\, !q$
{B} ⇔ $p \,\&\, !q$
{C} ⇔ $!p \,\&\, q$
{D} ⇔ $p \,\&\, q$



状態○Bと状態○Dの状態集合を表す論理関数
{B, D} ⇔ $p \,\&\, !\,q \mid p \,\&\, q = p$

状態○Aから状態○Cへの遷移関係を表す論理関数F_R
(A,C) ⇔ $!p \,\&\, !q \,\&\, !p' \,\&\, q'$

ソフトウェアシステムの動き(処理の流れ)を状態遷移図で捉える

KEPCO MPS

モデル検査器

対象システム
(仕様書)
(ソースコード)

検査用モデル
(専用言語でプログラム作成)
(検査対象の模擬システム)

```
MODULE main
VAR
Home:boolean;
Bus:{on, off}
ASSIGN
init(PC) : = ON
next(PC):= case
```

検査項目(時相論理式)

```
SPEC
EF(vend=COIN & pow = ON & can = OFF…
SPEC
AG(vend=COIN & pow = ON & can =OFF…
```

モデル検査器
(大学、研究グループのオープンソース)

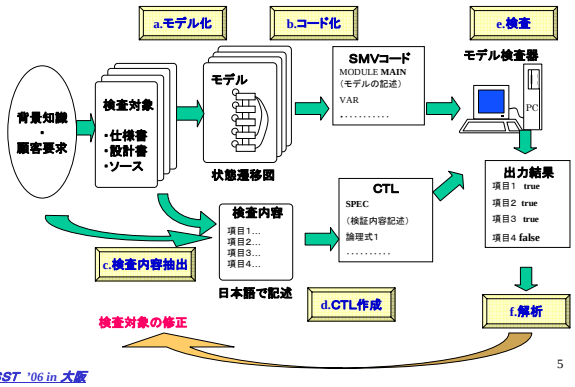
論理関数に変換して、発生し得る全状態を展開する

検査項目(時相論理式)が、満たされているかを自動検査する

検査項目が満たされない場合、反例を出力する

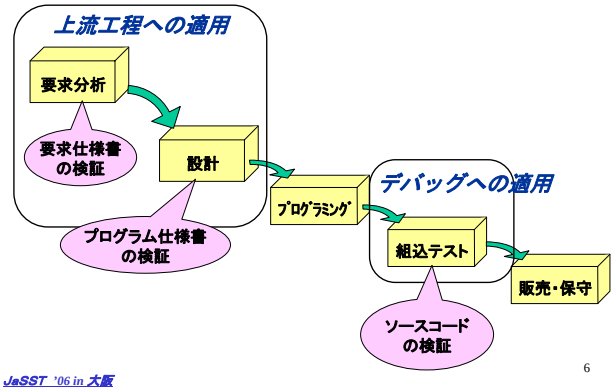
KEPCO MPS

モデル検査の流れ



KEPCO MPS

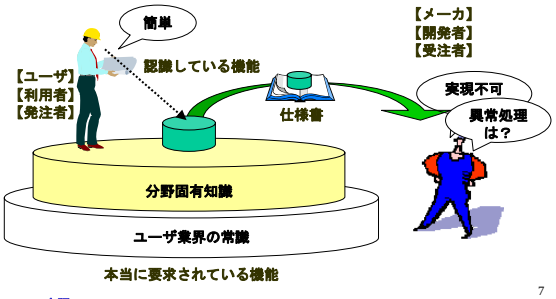
ソフトウェア開発工程への適用



KEPCO MPS

仕様書作成の問題点

経験則による要求提示 ⇔ 論理的な意思疎通 ⇔ モデル検査で判明する要求のあいまさ、矛盾

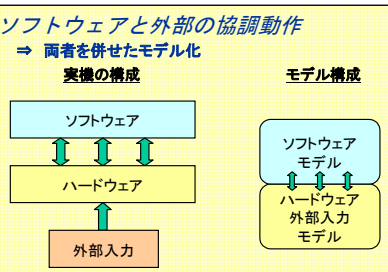


KEPCO MPS

設計工程でのモデル検査

仕様の詳細化によりシステムが複雑化
⇒ サブシステムへの分割
⇒ サブシステムの組み合わせ

ソフトウェアと外部の協調動作
⇒ 両者を併せたモデル化



KEPCO MPS

デバッグへのモデル検査適用

デバッグ
1.不具合を再現する
2.原因を発見する
3.原因を修正する
4.修正を確認する

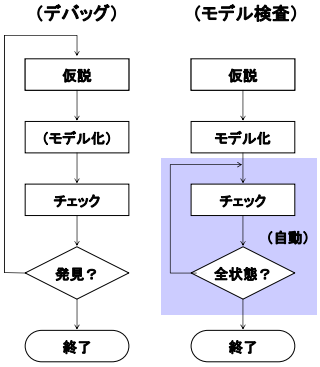
S.McConnel: Code Complete.
完全なﾌﾟﾛｸﾞﾗﾐﾝｸﾞを目指して
日経BPソフトプレス

再現困難な場合には、原因予測（仮説）して確認する
- 発生した不具合現象、ソフトウェア構造、経験から原因予想
- コード見直し、テストデータ入力、デバッグツール活用により確認

従来の方法では、デバッグの効率化が難しい

KEPCO MPS

デバッグとモデル検査の流れ



KEPCO MPS

モデル検査によるデバッグのメリット

1. 網羅的な検査ができる
2. 不具合時のトレースが見れる
3. 不具合内容が判明しておりモデル化が容易
4. モデルが小さいと作業時間が短く、状態爆発も起きにくい

KEPCO MPS

モデル検査とテストの比較

Q.モデル化の時に不具合をどう推論するの？

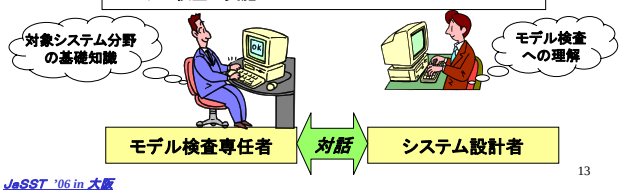
A.関係しそうな要素を漏れなくモデルに取り込むだけ。

モデル検査	*テスト*
モデル（状態遷移系）を検査する	コードを検査する
モデル作成が必要	モデル作成不要
テストデータ不要（検査項目は必要）	テストデータが必要
全状態を網羅的にチェック	特定の状態だけをチェック
状態爆発すると答えが返らない	答えは返るが、何度も検査が必要

KEPCO MPS

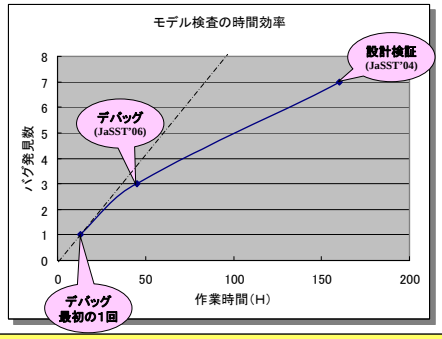
モデル検査によるデバッグ手順

- ・検査対象の選定　（効果の大きいものを）
- ・検査メンバーの選定　（モデル検査が理解できる人）
- ・モデル化の準備　（必要な要素を全て抽出）
- ・モデルの作成　（ソースを絞り込みモデル化）
- ・モデル検査の実施



KEPCO MPS

デバッグの評価



デバッグでは、作業時間が飛躍的に短く、効率が良い

KEPCO MPS

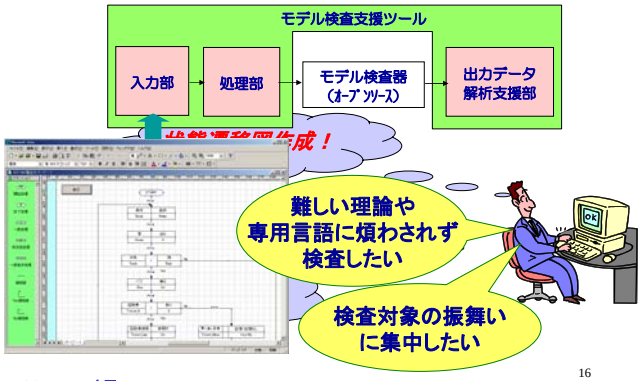
モデル検査で見つかるバグ例

	組み込みソフトウェアの例		監視制御ソフトウェアの例
	仕様書の検証	プログラム仕様書の検証（～160H、7件）	コードのデバッグ（～50H、3件）
検査項目	要求仕様どおりにデータ書替	外部入力の変動に追従して切替できるか	稀にデータ取得せず
	デッドロックしない		テスト時に一度だけ不正出力
発見したバグ	想定外の書替手順	・想定外の変数アクセス ・想定外の割り込み ・想定外の外部入力 ・割り込みのｵｰﾊﾞｰﾍｯﾄﾞ	ループ文の誤り
	タイムアウト記述忘れ		コメントアウトの誤り

仕様や動作の協調、場合分けや手順の正しさに関する不具合

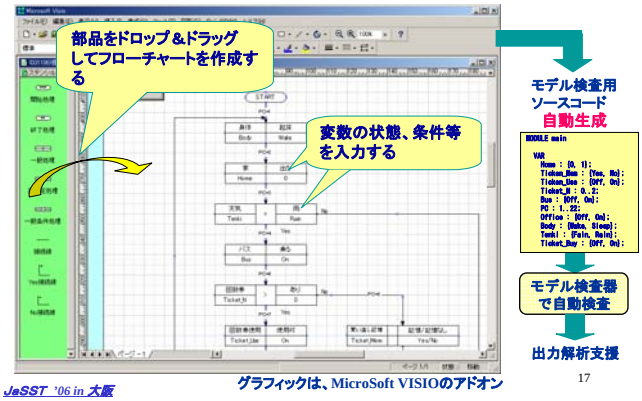
KEPCO MPS ENEGATE

モデル作成作業の支援



KEPCO MPS ENEGATE

入力支援機能（フロー図）



グラフィックは、MicroSoft VISIOのアドオン

KEPCO MPS ENEGATE

今後の取り組み

【モデル検査によるソフトウェアテストの実践研究】についてホームページを立ち上げました

http://www.modelcheck.jp

これまでのモデル検査によるソフトウェアテストの経験から、事例や実務適用に不可欠なノウハウの一部を公開しています


HOME
実践！ソフトウェアモデル検査
品質確認とバグ解決に向けて
ソフトウェア開発でお困りではありませんか？
取り組み内容
トピックス
モデル検査とは
適用事例
メンバー専用
【更新履歴】
2005/01/23 ホームページ公開