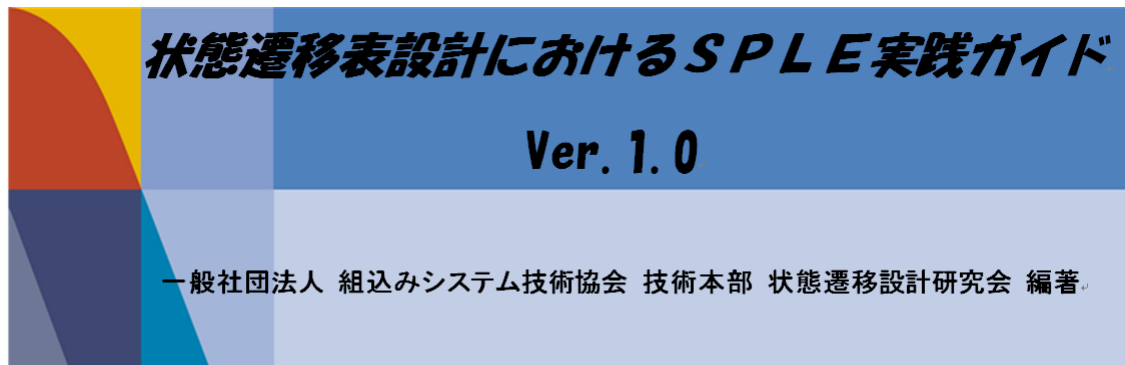


「状態遷移表設計におけるSPL E実践ガイド」紹介

J A S A 技術本部 状態遷移設計研究会、2012年度成果報告



■はじめに

状態遷移設計研究会は、状態遷移の検討漏れに気づきやすいという特性を持つ表形式の設計手法(状態遷移表)を普及させることを目的に活動しています。本研究会では2010年度より、「状態遷移表設計を適用し系列製品開発における状態遷移表の再利用性を向上させる手法の研究」をテーマに活動してきました。

系列製品開発の品質向上と効率改善は、組込みソフトウェア業界の共通の課題であり、多くの企業の関心を集めるテーマです。

研究のなかで、系列製品開発におけるフィーチャモデルⁱの作成や、状態遷移設計の状態の定義法、状態遷移表の階層化とフィーチャⁱⁱの対応法などのノウハウをまとめ「状態遷移表設計におけるSPL E実践ガイド」を策定しました。

本ガイドは、系列製品開発の再利用を前提としたプロセスであるソフトウェアプロダクトライン開発(SPL E)と状態遷移表設計を融合させたプロセスガイドで、多くの皆様に活用していただけるよう、JASA ホームページに掲載しています。

http://www.jasa.or.jp/top/activity/state_transition.html

ここに、その概要を紹介します。

「状態遷移表設計におけるS P L E実践ガイド」紹介

■目的・背景

状態遷移表設計の普及・啓蒙活動にあたり、系列製品開発の体系だった開発手法と状態遷移表設計を融合させたプロセスを定義し、再利用しやすい状態遷移モデルを構築し、品質・生産性の向上に貢献することを目的に、このガイドを策定しました。

■「状態遷移表設計におけるS P L E実践ガイド」解説

本実践ガイドは、1～3章の構成になっています。

1章は「S P L Eと状態遷移表設計」【概念編】として、S P L Eと状態遷移表設計の特徴、概念を簡単に説明しています。

2章は「S P L E開発プロセス」【解説編】で、今回制定した融合プロセスを要求仕様からのフィーチャモデルの構築、状態の導出・定義、フィーチャと状態遷移表の対応法を実践例で紹介しています。

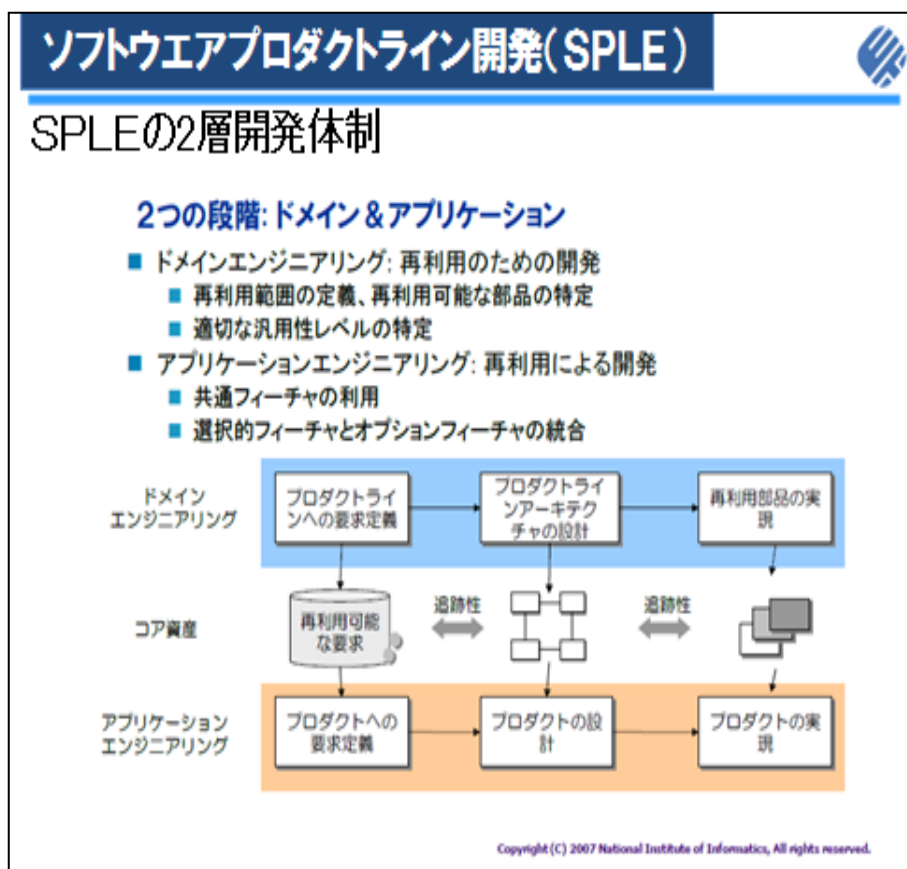
そして3章は、「S P L E開発プロセス」【プロセス定義編】で、プロセスの範囲とタスク構成を定義し説明しています。

「状態遷移表設計におけるSPLE実践ガイド」紹介

■ソフトウェアプロダクトライン開発（SPLE）

SPLEの特徴は、

- ①システムの特徴や機能を**共通性と可変性**に着目してフィーチャモデルで分離すること。
 - ②シリーズ開発のために、**アーキテクチャを資産化して、再利用**すること。
 - ③**コア資産の開発（ドメインエンジニアリング）とプロダクトの開発（アプリケーションエンジニアリング）を分離した、2層開発体制**であること。
- にあります。



「状態遷移表設計におけるSPL E実践ガイド」紹介

■状態遷移表設計手法

状態遷移表設計は、モデリング開発の一つです。


モデリングの対象には、機能、構造、振る舞いという観点の違った対象があり、それぞれをモデリングするためのダイアグラムがあります。

状態遷移モデルは振る舞いを対象としたモデル手法であり、リアクティブ（反応型）なシステムをモデリングするには、振る舞いを対象にしたモデリングが適しています。

そして状態遷移表は、要求や設計の漏れ・抜けを発見しやすいという特長があります。

状態遷移表設計手法

- **モデリングとは？**
設計品質向上のスタートポイント。
正確で曖昧性の少ない記述方法に基づいて設計すること。
- **モデリングのメリット**
 - ・正確な設計
 - ・レビュー効率向上
 - ・コミュニケーションの円滑化
 - ・標準化
 - ・設計、検証作業のツール化
- **モデリングの対象**




モデリング対象	意味	ダイアグラム
機能	入力と出力の関係で表すモデル	ユースケース図、データフロー図
構造	概念、データ、情報との関係で表すモデル	クラス図、コンポーネント図、パッケージ図、配置図など
振る舞い	入力イベントに対して、処理と出力イベントの関係で表すモデル	アクティビティ図、シーケンス図、タイミング図、コミュニケーション図、状態遷移図、状態遷移表など

■ 状態遷移設計

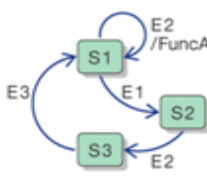

⇒組み込みシステムの特徴は、ユーザによる操作やセンサ信号などの外部からの事象(イベント)に対して、ランプを点けたり、モータを回転させたりなどの処理(アクション/振る舞い)を行う**反応型(リアクティブ)システム**であることが多い。

このようなリアクティブなシステムにおいては、内部の状態と外部の事象によって制御をおこなうよう、**振る舞いをベースにしたモデリングが適している。**

■ 状態遷移表のメリット



状態遷移図 状態遷移表

=同じ情報

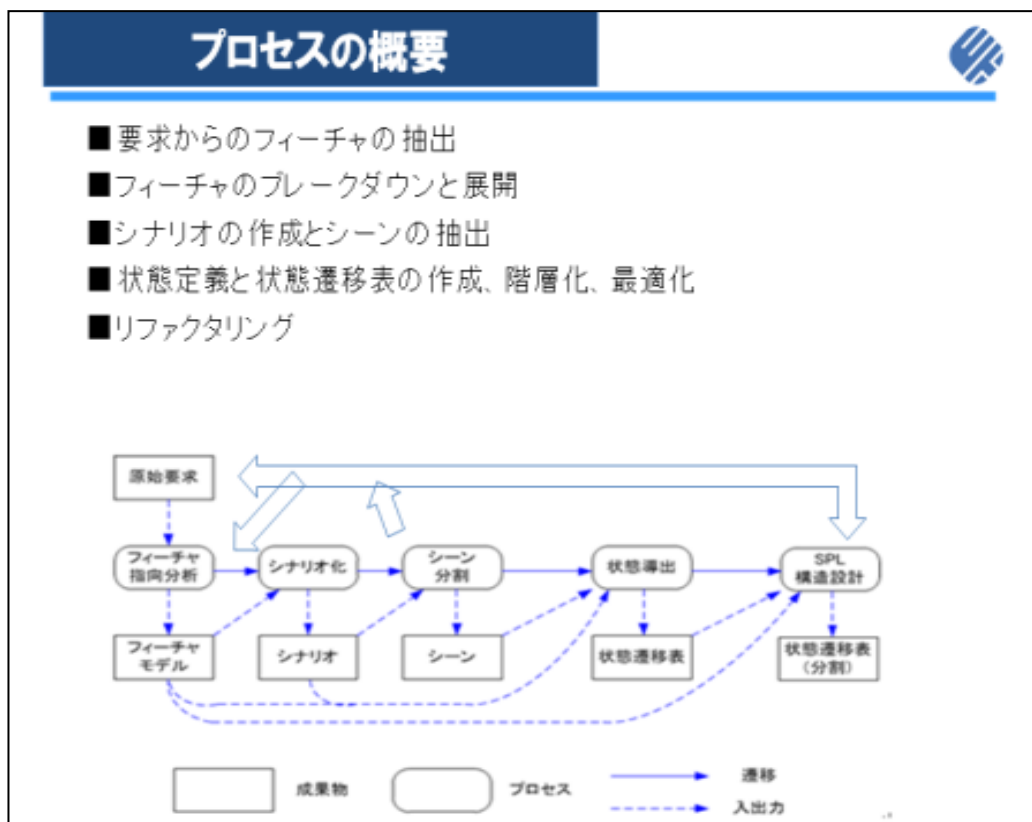
モデルズケ発見=バグ発見!

「状態遷移表設計におけるSPL E実践ガイド」紹介

■プロセスの概要

本プロセスは、

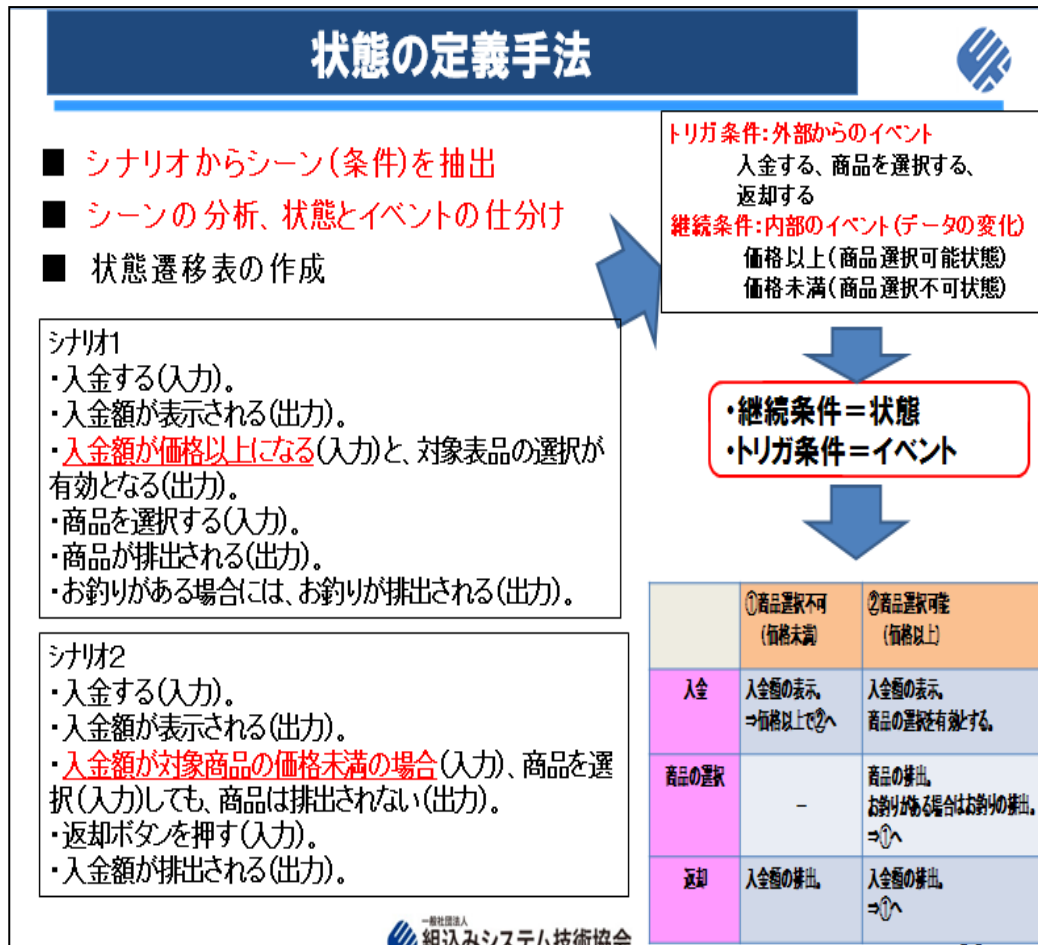
- ①要求からのフィーチャ抽出（フィーチャ指向分析）、
 - ②シナリオの作成とシーンの抽出、
 - ③状態定義と状態遷移表の作成、
 - ④フィーチャと状態遷移表の対応関係の維持
- からなります。



「状態遷移表設計におけるSPL E実践ガイド」紹介

■状態の定義手法

シナリオから入力条件を抽出して、シーンを定義します。
 シーンを分析し、システムの外部からの条件と内部の条件に分離します。
 システムの外部からの条件は、トリガ的条件としてイベントに、システムの内部の条件は、継続的条件として状態と定義します。



「状態遷移表設計におけるSPL E実践ガイド」紹介

■フィーチャモデルの階層化法

フィーチャモデルとアーキテクチャの遊離の問題を、フィーチャモデルの4階層定義とハード構成を含むアーキテクチャ構造とのマッピングで解決します。

フィーチャモデルを、


- ①特性層 (サービスレイヤ)
- ②機能特性層 (実現レイヤ)
- ③ハードコンポーネント層 (環境レイヤ)
- ④アスペクト層 (ドメインレイヤ)

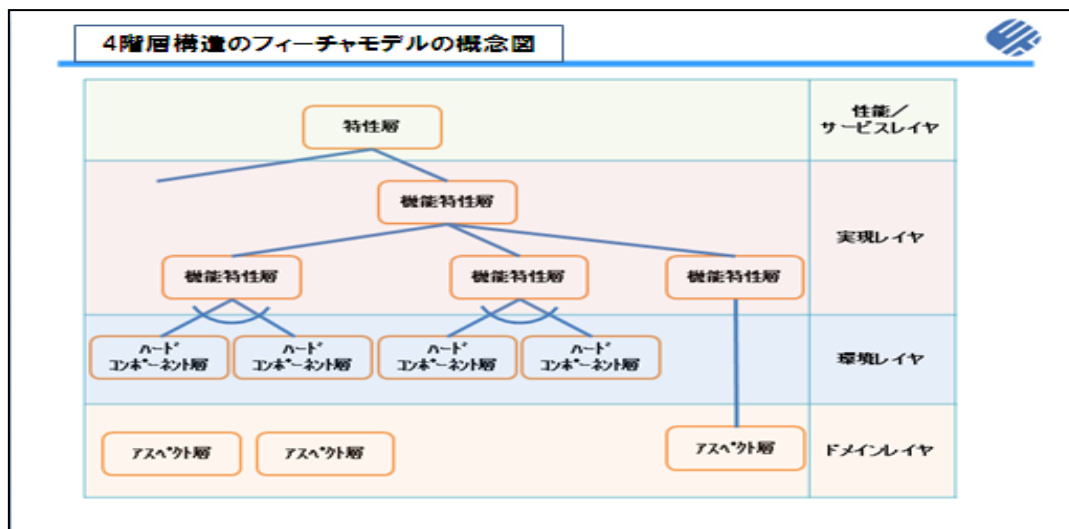
の4階層で定義します。

フィーチャモデルの階層化法

■フィーチャモデルを、以下の4階層で定義する。

- ① **特性層** ... サービスレイヤ
 - ・ユースケースから抽出した実現したい状態や状況
- ② **機能特性層** ... 実現レイヤ
 - ・特性層で抽出したフィーチャを実現するために必要なフィーチャとして抽出
 - ・非機能特性に関してもこのレイヤでフィーチャ化を考慮
- ③ **ハードコンポーネント層** ... 環境レイヤ
 - ・フィーチャの最下位層と1:1に対応させた、ハードウェアコンポーネント
- ④ **アスペクト層** ... ドメインレイヤ
 - ・共通の機能要件、ドメインの標準技術などをフィーチャとして抽出





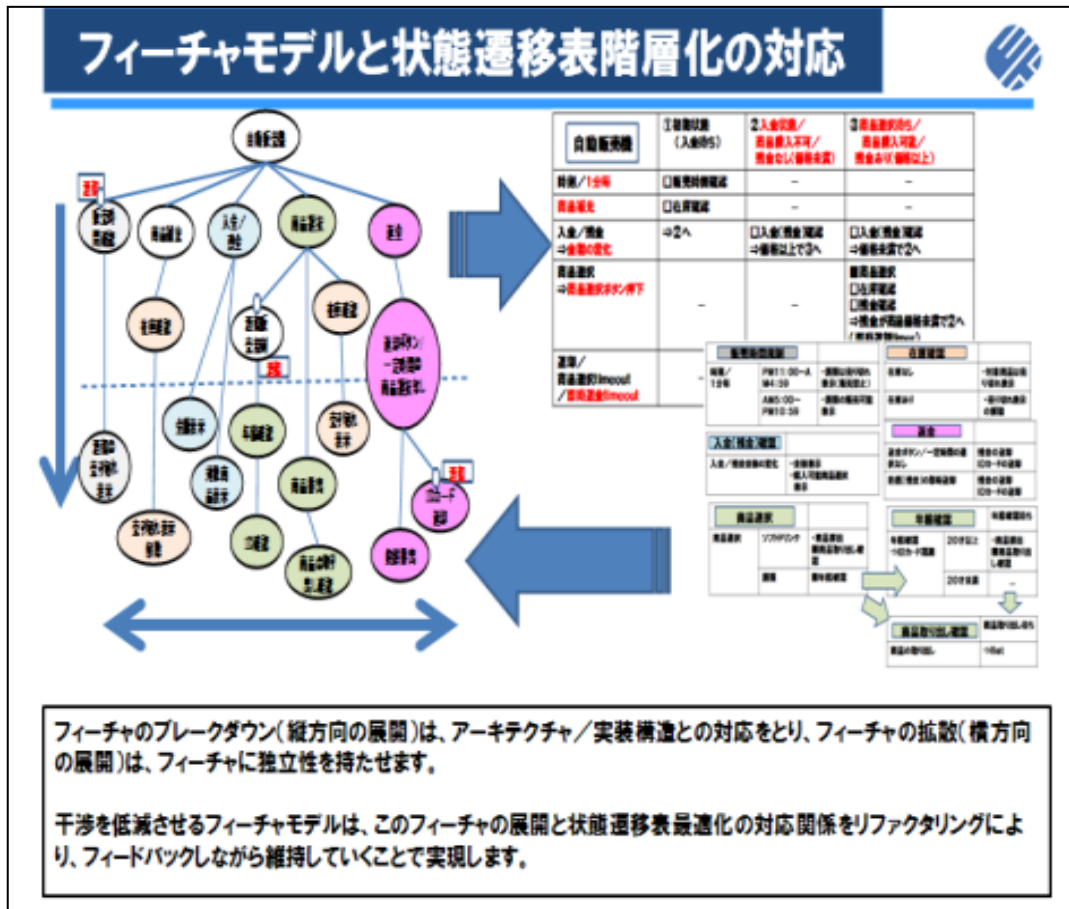
機能特性層で、サービスの実現手段をシステムの視点でブレイクダウンします。そして、ハードウェアコンポーネントが1対1で対応するところまで落とし込むことがポイントです。結果、実現手段との乖離が起きにくくなります。

「状態遷移表設計におけるSPL E実践ガイド」紹介

■フィーチャモデルと状態遷移表階層化の対応

フィーチャのブレークダウンは、アーキテクチャとの対応をとり、横展開はフィーチャに独立性を持たせます。

状態遷移表の階層構造とフィーチャの対応をとり、フィーチャの抜き差しによるプロダクトの構成を、対応した状態遷移表の抜き差しにより実現することで、状態遷移モデルの再利用とメンテナンス性を向上させることが可能となります。

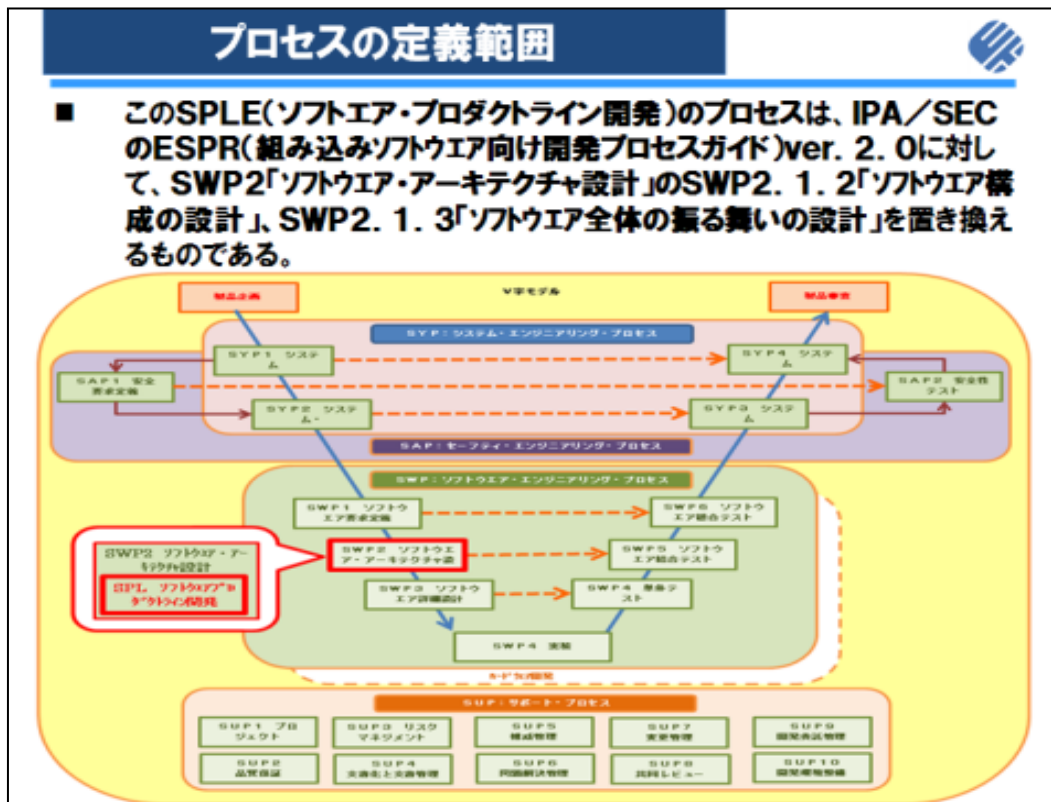


「状態遷移表設計におけるSPL E実践ガイド」紹介

■プロセスの定義範囲

本プロセスガイドは、IPA/SECⁱⁱⁱのESPR（組み込みソフトウェア向け開発プロセスガイド）ver. 2. 0で定義されているV字モデルのうち、SWP2「ソフトウェアアーキテクチャ設計」内の作業として、SPL「ソフトウェアプロダクトライン開発」を新たに定義します。

そして、SWP2. 1. 2「ソフトウェア構成の設計」、SWP2. 1. 3「ソフトウェア全体の振る舞いの設計」を置き換え、プロセス適用することを前提としています。



以上、「状態遷移表設計におけるSPL E実践ガイド」を紹介させていただきました。

JASA ホームページからダウンロードして、ハンドブックとして活用ください。

■今後の活動

状態遷移設計研究会では、本年度より新たにリバースモデリングをテーマに活動をしています。新たな手法やプロセスに取り組むにあたり、弊害となるのはレガシーコードの存在です。近年のソフトウェア開発では、新規にゼロベースで開発することは皆無です。

しかしながら、度重なるプログラム変更や機能追加でシステムのブラックボックス化が進み、システム障害やトラブル対応に戸惑う、システム改修時に膨大な調査期間がかかる。

このようなことが起きています。

これは、**現行のシステムがブラックボックスとなっており、資産として活かされていない**ことに起因しています。

そこで、レガシーコードから状態を抽出し状態遷移表モデルをリバース生成する手法の研究に取り組むこととしました。

テーマは、「**状態遷移表のリバースモデリングへの適用**」です。

「**フラグがある所に、状態はある！！**」と、仮説を設定しました。

この仮説を実証する研究活動を行っています。

このような活動に興味がありましたら、是非、状態遷移設計研究会にご参加ください。

2012年度 状態遷移設計研究会報告
「状態遷移表設計におけるS P L E実践ガイド」の紹介

2013/11/20 発行

発行者 一般社団法人 組込みシステム技術協会
東京都中央区日本橋浜町1丁目8-1
TEL: 03 (5821) 7973 FAX: 03 (5821) 0444
URL: <http://www.jasa.or.jp>

本書の著作権は一般社団法人組込みシステム技術協会(以下、JASA)が有します。
JASAの許可無く、本書の複製、再配布、譲渡、展示はできません。
また本書の改変、翻案、翻訳の権利はJASAが占有します。
その他、JASAが定めた著作権規程に準じます。

i フィーチャモデル=フィーチャを長期的／短期的視点で分析し、「共通 (mandatory)」・「選択 (Optional)」・「択一 (Alternative)」に分類しフィーチャのツリー構造図で階層的に表現したモデル。

ii フィーチャ=ユーザが識別できるシステムの目立った特性。

iii I P A / S E C = 独立行政法人 情報処理推進機構 ソフトウェア・エンジニアリング・センター