平成 27 年度 技術本部成果発表会

プログラム

13:30~	開会あいさつ 漆原技術本部長(審査委員長)	
13:35~	スケジュール・表彰について 奥村技術本部長補佐	
13:40~14:10	OSS 活用の障害。ロボット用 OSS の活用 技術高度化委員会 OSS 活用 WG 竹岡 尚三 氏	1p∼19p
14:10~14:40	要求の仕様化に関する活動の最終報告 安全性向上委員会 中村 洋 氏	20p∼35p
14:50~15:00	IPA との 連携活動のご紹介(障害原因診断 WG) (独)情報処理推進機構 三原 幸博 様	36p∼41p
15:00~15:30	レガシーコードの 蘇生術 技術高度化委員会状態遷移設計研究会 小林 良 氏	42p∼55p
15:30~16:00	ロボット向け組込みソフトウェアライブラリ OpenEL 応用技術調査委員会プラットフォーム研究会 古俣 学氏	56p∼71p
16:10~16:30	JASA の対外アピールについて~クミコちゃんを中心に~ 応用技術調査委員会技術セミナーWG 冨岡 理 氏	72p∼75p
16:30~16:40	これからの「ものづくり技術者」の育成 ハードウェア委員会 矢部 哲美 氏	76p∼83p
16:40~16:50	新技術の調査(ロボットや制御機器の技術調査) ハードウェア委員会 笠木 大幹 氏	84p~90p
16:50~17:20	アジャイル研究会の挑戦〜組込みとアジャイルと、契約と〜 (中部支部)アジャイル研究会 戸澤 充 氏	91p~102p
17:40~	交流会・表彰式 @レストラン"Hitokadoru"	



OSSライセンスと ロボット用OSS

2015年5月20日 OSS活用WG/技術本部副本部長 竹岡尚三 (株)アックス



© Japan Embedded Systems Technology Association 2015



今、OSS (オープンソース・ソフトウェア) なのか?



© Japan Embedded Systems Technology Association 2015

-

大人のOSS使用は、すでに十数年の歴史あり



- ■1980年代中,後期~
 - Ciscoは、BSD UNIXベースのルータ
- ■1990年代末期~
 - ・Linuxが、家電各社(SONY,パナソニック,シャープなど)にて採用
 - ・半導体企業も独自CPUに、Linuxを移植

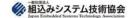
■ナウ

- Android, Linux, BSDが無ければ、IT機器を作るのが大変
 - ※JASA会員企業いわく: お客様の指定で、OSS採用を行わねばならない!
- ■これから
 - ■ロボット用 OSSが、続々と出現中

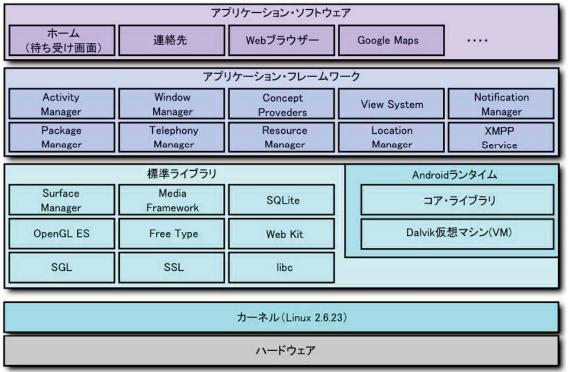
OSSでも、巨大ソフトウェア・スタック



- ■OS(Linux, BSD, OpenSolaris)
- ■インターネット接続
- ■Windowシステム(窓、描画管理)
- ■文字レンダリング
- ■3Dグラフィックス
- ■仮名漢字変換
- ■ブラウザ、メーラ
- ■マルチメディア(音楽、動画)の記録/再生
- ■プログラミング言語 Ruby,Perl,PHP…
- ■オフィススイート



Androidのアーキテクチャ



• 蓬田宏樹、他著「Androidの野望」 日経エレクトロニクス 2007 年12月17日号 p.47-69

OSSの本当の利点



- ■「無料だから嬉しい」とか言ってると、負ける
- ■特定企業のOSとは違い、ソースがあるので
- ■理解し、独自の改良が可能
- ■デファクト・スタンダードがOSSなら、特定ベンダに 囲い込まれない
- ■特定のソフトウェアのリリース(バージョンアップ)に 引っ張られない
 - 独自に品質を上げたソフトウェアを、自由に維持できる
 - ・※日本の一部の分野に、アレな発注主も居るようだが…



OSSって面倒くさい!?



- ■ライセンス問題
 - ・ソースコード公開義務(!?)
 - 一誤解が多い
- ■品質問題
 - ・適当に作られた無料のソフトウェアの品質は?
 - 一どこの誰だか、知らない人が作ったものでしょ…
 - ──タダのものがそんなに良ければ、プロは商売あがったりだよ ✓タダのものが、かなり良い、ということは、非常に頻繁にある
 - ・品質の押さえ方は、大事
- ■OSS共通の問題をみんなで考えよう
 - ·→JASA OSS活用委員会



OSSのせいで、仕事が減る?!



- **■**これまで、
 - □ 受託して,仕様を起こし、一品づつ作っていた
 - 『ライブラリ/モジュール(サブルーチン)を作って販売していた
- ■しかし、現在、
 - □ JPEG デコード・ルーチンの開発を委託する顧客がいるだろうか?
- ■顧客が、OSS前提で仕様を決めている
 - 『サービス主導の時代
- ■Windows+ブラウザで、できていることは、できて欲しい
- ■高度なロボットを、すぐに作りたい
- ■OSSを活用する世界をみんなで考えよう
 - ·→JASA OSS活用委員会

組込みシステム技術協会



代表的な OSSライセンス



代表的なライセンス



- **■**GPL
- ■BSDライセンス
 - MITライセンス
- ■Apacheライセンス
- ■Mozilla publicライセンス(MPL)



GPL



■ GPL

- ソースコードは、そのバイナリを持っている人には、公開されるべき。という思想
- ・ 改変したソースコードも、公開されなければならない
 - ─独自に改良したコードも公開しなければならない
 - ―(組み込みの世界では、厳しいと言えるだろう)
- GPLのソフトウェアにリンクしたソフトウェアのソースコードも、公開しなければならない - 独自に作ったソフトウェアも、GPLソフトウェアにリンクしたら、自動的に公開義務が生ず
 - ─(極めて厳しい、と言えるだろう)

■ LGPL

- ライブラリのために作られたライセンス
- LGPLライブラリは、ダイナミック・リンクであれば、リンクを行っても、他のソフトウェアに、GPL (LGPL)が伝播することは無い
- 小さな組み込みシステムでは、ダイナミック・リンクが現実的ではないかも
- → 結局、小規模 組み込みシステムでは厳しい、と言えるかも

■ GPL v3問題

- ・ ソースコードを寄付した人が持っている特許が、そのソースコードに含まれていた場合、当該 特許を無償で無制限に許諾しなけ`ればならない
- 大企業の多くが、GPL v3のソフトウェアの開発には関わっていない
 - ─Linuxは、GPLv2 維持宣言をしている(2014年秋 現在)



11

GPL:よくある誤謬



- ■ソースはインターネットで配布しなければならない
 - 嘘です
- ■ソースを誰にでも上げなければいけない
 - 嘘です
 - バイナリを持っている人にだけ、ソース入手の権利があります
- ■誰とライセンス契約をすればいいのか
 - する必要はありません
- ■GPL違反をするとストールマンに訴えられる
 - 嘘です
 - 日本では、プログラムの著作権者しか訴える権利がありません。
 - ・ストールマンが怒る可能性はありますが、それには法的な意味はありません

MPL

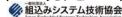


■Mozilla publicライセンス(MPL)

- MPLのソースコードは、MPLとして公開する(GPLへの移行は可能)
- ・改変しても構わない。ただし、特許を含むコードは改変 不可
- 他のモジュール,ファイルは、別なライセンスでも構わない
 - 一独自に作ったモジュールの公開義務は生じない

CDDL

- ・旧Sunが、NetBeans IDEやOpen Solarisで採用していた
- MPLに非常に近い。
- ・ソフトウェアの開発者の規定が明確



10

BSDに近い



■BSDライセンス

- MITライセンスは、BSDと同じ
- 何をしても、ソースコードの開示義務は無い
- 古いBSDライセンスは「宣伝条項」があったが…一宣伝条項: すべての広告にライセンス表示を入れる
- ・現在のBSDライセンスは、「宣伝条項」は無い

■Apacheライセンス

- Apache Licenseのコードが使われていることを知らせる文言を入れる。という義務がある程度
- 何をしても、ソースコードの開示義務は無い
- ライセンスを変更してもよい
- ライセンスされたファイルに元々ある著作権と特許権の記述は そのまま保持
- Apache Licenseのコード中に特許を含んでいる場合、特許を 無償で無限に許諾しなければならない

※ 組込みシステム技術協会

ライセンスの知識は必要



- ■正しい知識があれば、問題は出ない
 - GPLであっても、ソフトウェア作成者は、
 - みんなに使って欲しいと考えている
 - → ユーザが困るようなことはしていない(はず)
- ■ズルは止めよう
 - ・GPLのコードの断片を、独自開発のソフトウェアに 混ぜると、公開義務が生じる
 - ─BSDライセンスのものは、ライセンス表示をソースコード に入れれば、問題無い



15



ロボット用 OSS

ロボットもOSS時代



■英語版Wikipedia「Open-source robotics」 の項

http://en.wikipedia.org/wiki/Open-source robotics

- ■フル・ロボット・プロジェクト32個
- ■ソフトウェアのみのプロジェクト 28 個
- ■オープン・ハードウェア・プロジェクト:3個

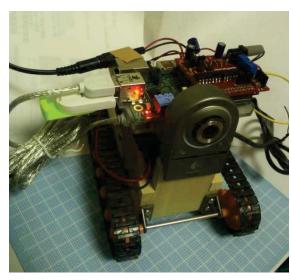


17

OSSロボット例 (たけおか謹製)



- ■人間の顔を見つけたら、追尾する
 - ・顔の位置を判断して、右へ行くか、左へ行くかを決める
- ■Linux+OpenCV+OpenEL
- ■OpenCVは顔認識などを含む
- ■総合的な画像処理ライブラリ
- ■OpenELはJASAなどが推進している
- ■ロボット用の下位の抽象化層の規格
 - ・DCモータの制御はOpenEL
- ■ハードウェア
 - RaspberryPi
 - —ARM11@700MHz
 - -512MBytesRAM
 - ・USBカメラ
 - DCモータ





OpenCV



- ■画像処理のミドルウェア
 - Intel, Willow Garage が開発
 - ・ いろんなものが入っている
 - ■極めて便利
 - 一画像処理 (Image Processing)
 - 一構造解析 (Structural Analysis)
 - ーモーション解析と物体追跡 (Motion Analysis and Object Tracking)
 - 一パターン認識 (Pattern Recognition)
 - カメラキャリブレーションと3次元再構成 (Camera Calibration and 3D Reconstruction)
 - 一機械学習
 - --ューザインタフェース



19

OpenCVによる顔認識

■OpenCV

- 一般的画像処理
- ・画像認識
 - 一学習器
 - 一認識器
- ・カメラ入力部
- ■USBカメラを接続
 - ・ロジクール QCAM-E2500, 30万画
 - Logicool 861200
 - カメラ入力

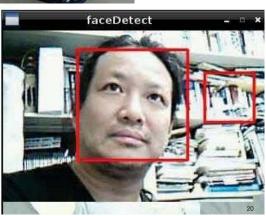
■顔認識

- ・学習済みデータが用意されている
- 簡単に使用できる
- ロボットの目玉にでも

「たけおか opencv raspberrypi」でググ?









ロボット用 OSS ミドルウェア



© Japan Embedded Systems Technology Association 2015

2

en.Wikipedia「Open-source robotics」のOSS上位など



- **■**NXJ
 - ・Lego NXTロボット・キット用のOSS Javaプログラミング環境
- ■CLARAty
 - ・JPL(ジェット推進研究所)で開発されている、Mars(火星)プログラムの一部として。
- ■ROS (Robot Operating System)
 - BSDライセンス。すでに50以上のロボットで動作
- **■**URBI
 - ・C++の分散/組み込み コンポーネント・フレームワーク
 - ・ + 並列/イベント駆動 制御スクリプト言語
- ■Player (robot framework)
- ■OpenRTM-aist (robotics technology middleware)
 - 日本の産総研が開発
- ※上記のいくつかは、ロボット・ミドルウェアである



ロボット用ミドルウェアとは



■フレームワーク

- コンポーネントをつなぐ
- ・ コンポーネントの独立性を高めさせる
- オーバーヘッド無し
- 存在を感じさせない

参考

http://en.wikipedia.org/wiki/Robotics_middleware



23

ROS



- ■Robot Operationg System
- http://wiki.ros.org/
- ■ロボットのミドルウェアと、たくさんのコンポーネント
- ■BSDライセンス
- ■すでに50以上のロボットで動作
- 2007年 the Stanford Artificial Intelligence Laboratory で開発
 - Stanford Al Robot STAIR project
- ■2008~2013年 Willow Garageが中心に開発中
- ■2013年~ the Open Source Robotics Foundation

※参考

http://en.wikipedia.org/wiki/Robot_Operating_System



ROS



- ■データの流れに応じて、コンポーネントをつなぐ
- ■自動車の自動運転でも採用
- ■OpenCVも含まれている
- ■雑に言ってしまえば…

■ロボットを作るためのソフトウェア部品の多くが含

まれている

E FUELER

画像は下記より引用

http://www.ros.org/news/resources/2010/poster2color_revis.jpg

✓ 組込みシステノ

25

ROSパッケージの適用分野



- 知覚 (Perception)
- 物体認知 (Object Identification)
- セグメンテーションと認識 (Segmentation and recognition)
- 顔認識 (Face recognition)
- ジェスチャ認識 (Gesture recognition)
- ■動作追跡 (Motion tracking)
- エゴモーション (Egomotion)
- 運動理解 (Motion understanding)
- 運動からの構造 (Structure from motion (SFM))
- ステレオビジョン:2台のカメラを通して、奥行き知覚
 - (Stereo vision: depth perception via two cameras)
- モーション (Motion)
- 移動ロボット工学 (Mobile robotics)
- ■制御 (Control)
- ■計画 (Planning)
- ■把握 (Grasping)

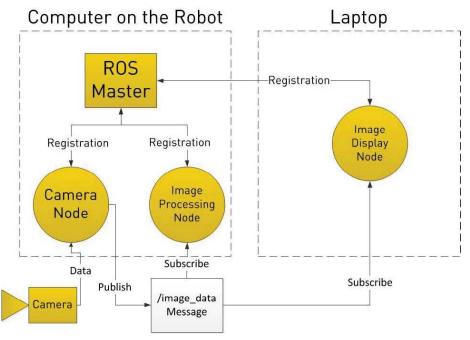
※http://en.wikipedia.org/wiki/Robot_Operating_Systemから引用、和訳



ROS



■データの流れに応じて、コンポーネントをつなぐ



http://www.clearpathrobotics.com/blog/how-to-guide-ros-101/ より引用

27

ROS



- ■ROSエコ・システムのソフトウェアは3つのグループ に分けられる
 - (1) 言語やプラットフォームから独立したツールROSベースのソフトウェアを作ったり分散させるために使用
 - (2) ROSクライアント・ライブラリ実装ーroscpp, rospy, roslispなど
 - (3)アプリケーションに関連するコードを含んだパッケージ — ひとつ以上のROSクライアント・ライブラリが使用
- ※http://en.wikipedia.org/wiki/Robot_Operating_Systemから引用、和訳



RTミドルウェア



- ■産総研などが開発しているロボット用ミドルウェア
 - RTコンポーネントは、OMGにて、国際標準化
- ■RT (Robot Technology/Robotic Technology)とは
 - ロボット機能要素
- ■RTミドルウエアとは
 - 様々な機能要素をモジュール化 (RTコンポーネント)
 - RTCを、ソフトウエア的に統合するためのプラットフォーム
- ■RTミドルウエアの目的
 - 仕様をオープンにする
 - 様々な実装同士が相互接続できるようにする
 - オープンアーキテクチャのプラットフォームを確立する
- 通信はCORBAベース
- ■「OpenRTM-aist」は、RTミドルウェアの産総研による実現
 - ライセンスは、LGPLおよび産総研と個別に契約するライセンスのデュアルライ センス方式

画像引用元,参考 http://www.openrtm.org/openrtm/ja/node/161#toc5 http://www.openrtm.org/openrtm/ja/content/openrtm-aist-officialwebsite **※組込みシステム技術協会**



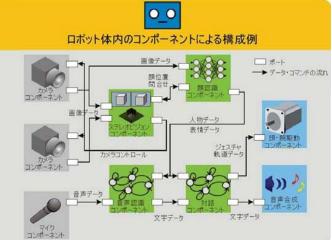
RTミドルウエア







※RTCはネットワーク上に分散可能

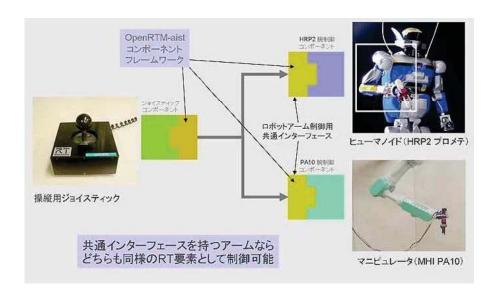


http://www.openrtm.org/openrtm/ja/content/rt%E3%83%9F %E3%83%89%E3%83%AB

%E3%82%A6%E3%82%A8%E3%82%A2%E3%81%A8%E3%81%AF%EF%BC 組込みシステム技術協会 %9F-0 よりコピー

RTミドルウェア





引用元

http://www.aist.go.jp/aist_j/press_release/pr2005/pr20050224/pr20050224.html

31

RTC(RTコンポーネント) OMG標準



- RTM / OpenRTM-aist は
 - ・コンポーネントモデル
 - ・ そのインターフェー ス
 - である「RTC」が OMG 国際標準
- OMG (Object Management Group)
 - 1989年に設立されたソフトウエア標準化団体
 - ・ CORBA (Common Object Request Broker Architecture)分散オブジェクトミドルウエア:
 - UML (Unified Modeling Language) ソフトウェアモデリング言語
 - ・ などを策定・管理している組織
- RTCのインターフェース仕様
 - ・ OMG において, 産総研と米国ミドルウエアベンダ RTI (Real Time Innovations) により標準化
 - ・RTC (Robotic Technology Component) Specification (http://www.omg.org/spec/RTC/1.0/) として2008年4月に公式リリース

http://www.openrtm.org/openrtm/ja/content/rt%E3%83%9F%E3%83%89%E3%83%AB

%E3%82%A6%E3%82%A8%E3%82%A2%E3%81%A8%E3%81%AF%EF%BC%9F-0#toc4 より引き写し (※) 組込みシステム技術協会

OpenEL



- ■JASA,産総研などが推進しているフレームワーク
- ■国際規格にするべく、OMGに提案中
- ■RTミドルウェアなどのコンポーネントの可搬性を 高める
- ■ハードウェアに近い層を、抽象化
- ■OpenEL準拠で書かれたソフトウェア(RTC)は、
- ■ハードウェア・ドライバに依存せずに動作する 参考

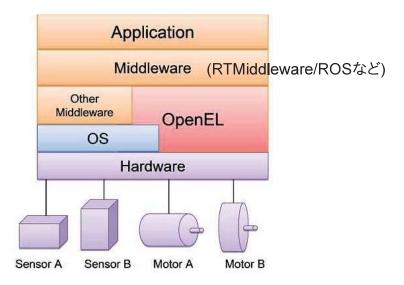
http://jasa.or.jp/openel/Main_Page/ja#OpenEL.E3.81.AE. E6.AD.B4.E5.8F.B2.E3.81.A8.E6.99.AE.E5.8F.8A.E3.83.BB. F5.95.93.F7.99.BA.F6.B4.BB.F5.8B.95



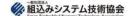
33

OpenEL



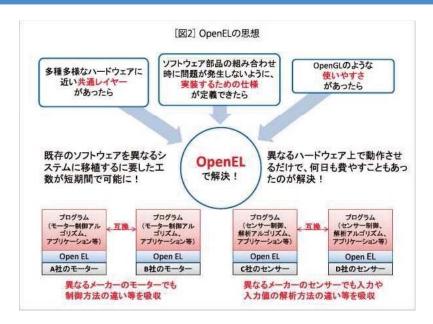


引用元 http://jasa.or.jp/openel/Main_Page/ja

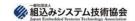


OpenEL





引用元 http://www.jasa.or.jp/top/activity_bulletin/bulletin041-05.html



35

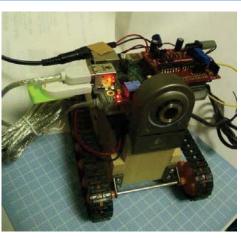
OpenEL使用 オレオレ実例





・ライントレーサ Multi Thread BASICインタープリタ OS無し OpenELでDCモータ駆動

PIC32MX250 (MIPSコア@50MHz、ROM128KB, RAM32KB) 秋月で360円



・顔追尾ロボット OpenCV under Linux OpenELで DCモータ駆動

Raspi (ARM11@700MHz, 512MBメモリ)





「OSSライセンスとロボット用OSS 2015」

2015/05/20 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。



© Japan Embedded Systems Technology Association 2015



技術本部成果発表会資料

要求の仕様化に関する活動の最終報告

~必ずしも"厳密さ"ではなく "意図"したものが実現できる要求定義を求めた活動の最終報告~

> 2015年5月20日 安全性向上委員会 株式会社レンタコーチ 中村 洋 hiroshi19.nakamura@nifty.com



Japan Embedded Systems Technology Association

Japan Embedded Systems Technology Association 2015

目次

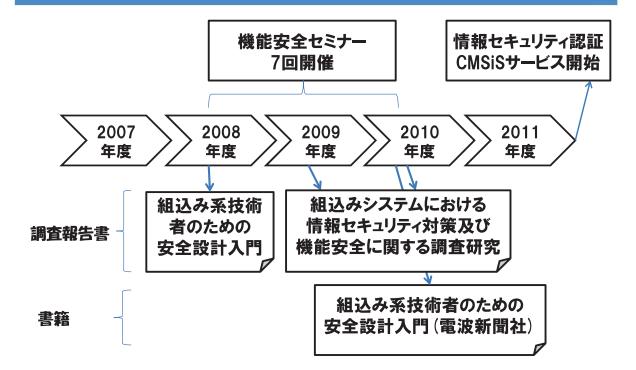


- 1. 活動計画と概況
- 2. 成果報告書の概要
- 3. 要求の仕様化を支援するプロセスと手法
 - システム要求設計技法による安全要求の記述
- 4. 今年度の活動体制



安全性向上委員会の活動実績





《 組込みシステム技術協会

• Japan Embedded Systems Technology Association 2015

3

2012年度からの活動テーマ





要求の仕様化に関する活動計画





目標とする成果物

- 要求の仕様化に関する要求事項や課題
- 要求の仕様化を支援するプロセスと手法やツール
- ・ 各層における発注・受託関係者に役立つ報告書。

■ 活動方法

- 機能安全や情報セキュリティ関連の規格を対象とし て、要求事項を洗い出す。
- REBOKや要求工学などの専門書を調べて、要求の仕 様化のプロセスと課題を整理する。
- ・ モデルベース開発で使われている手法やツールを調 べる。
- 実際に使われている形式手法とそのツールを調べる。
- JASA会員を対象としてアンケートを取り、プロセ ス、手法やツール、課題に関する実態を調査する。

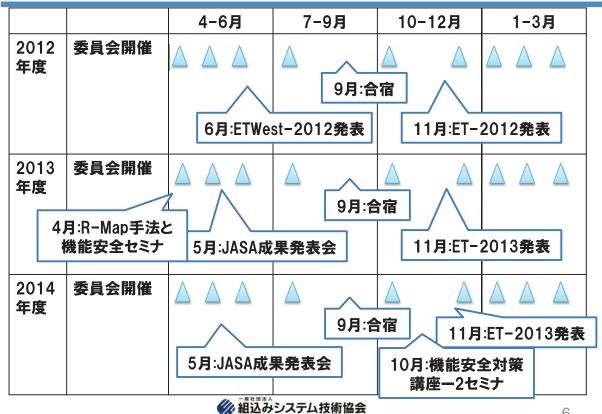
《 組込みシステム技術協会

Japan Embedded Systems Technology Association 2015

2012-2014年度の活動概況



6



Japan Embedded Systems Technology Association 2015



成果報告書の概要



Japan Embedded Systems Technology Association

• Japan Embedded Systems Technology Association 2015

7





要求の仕様化に関する課題、プロセス及び手法

~必ずしも"厳密さ"ではなく "意図"したものが実現できる要求定義を求めた活動の成果報告書~

2015年3月20日 安全性向上委員会



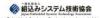
目次



はじめに

- 1.要求の仕様化に関する課題
- 2.目指す要求の仕様化プロセス像
- 3.要求の仕様化を支援するプロセスと手法
 - 3.1 システム要求設計技法
 - 3.2 物物関係分析法
 - 3.3 操作シナリオベース開発手法
 - 3.4 SPINを活用する仕様検証

おわりに





はじめに





- ▶ 安全性向上委員会は、2012年から3年間、要求の仕様化に 関して、その課題、プロセス及び手法を調査する活動を行いました。この報告書は、その成果をまとめたものです。
- ▶ 当委員会は、情報セキュリティと機能安全をテーマとして活動していますが、セキュリティも安全も、その要求定義が曖昧では土台から崩れると気づき、この活動に着手しました。
- ▶ 活動期間中に、組込み総合技術展などで都度、概略報告を 行ってきました。この報告書は、その発表資料に手を加え、要約 解説を追加して、次のように構成されています:

4組込みシステム技術協会

- 追加した要約解説
- 発表資料の完全版

おわりに



- 本活動は、開発者が要求仕様書の品質に悩まされていることから出発しています。しかし、 かといって、そのサブタイトルが示すように、要求仕様書は"厳密"でありさえすればよいという 訳ではなく、"意図"のようなものにも配慮すべきではないかというものです。
- ➤ 活動は「意図」を十分解明したとはいえませんが、活動としては仕様記述の方法を従来の方法も含めて比較検討しました。いくつかの重要な形式手法も調査の対象にしています。また抽象的な議論ばかりではなく、実際の事例をちって調査研究したところにも特色があります。要求開発は「要求獲得⇔分析⇔仕様化〜妥当性確認」と進む訳ですが、本活動で示された成果は、この要求開発のプロセスを見直すにあたり参考になると思います。
- ➤ 「意図」に関していえば、意図は要求には「理由や背景」を書くへきであるとするXDDPなどの 見解と重複します。意図を書くことにより、記述した内容がよく伝わったり、記述こみの指摘が あったり、またよりよい程案があったりします。仕様記述の量も減らすことができるのではないで しょうか。これは意図のどういう作用なのだろうというところに関心があります。
- ➤ もしかしたら知識には階層的な構造があり、擬人的にいえば、意図を投げかけると「それは自分に関係がある」と知識たちが手を挙げるのかもしれません。知識を用語化し、用語を階層的にうまく組み立てることができれば、限られたドメインの中ではありますが、意図によって仕様の領域を特定できるかもしれません。例えば、ここでドメイン知識は或る製品に関すること(要求仕様)であり、意図はその製品の型番に対応するかもしれません。そして、このようなドメイン知識の実現は「オントロジー」は核に依さるかれらいません。
- ➤ 委員会では組合員各位の参加を求めています。いろいろ議論しましょう。

安全性向上委員会委員長 漆原 憲博

参 組込みシステム技術協会

Japan Federalish System Technology Associate

24

9

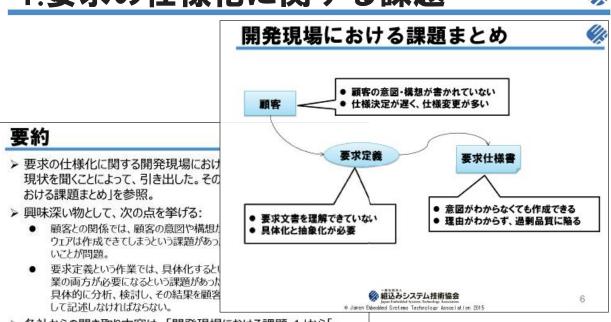
《組込みシステム技術協会

Japan Embedded Systems Technology Association 2015

1.要求の仕様化に関する課題

安全性向





各社からの聞き取り内容は、「開発現場における課題-1」から「-3」までを参照。

● 銀込みシステム技術協会

*** Just Minddel System Technology Responsed to 2015

*** は込みシステム技術協会

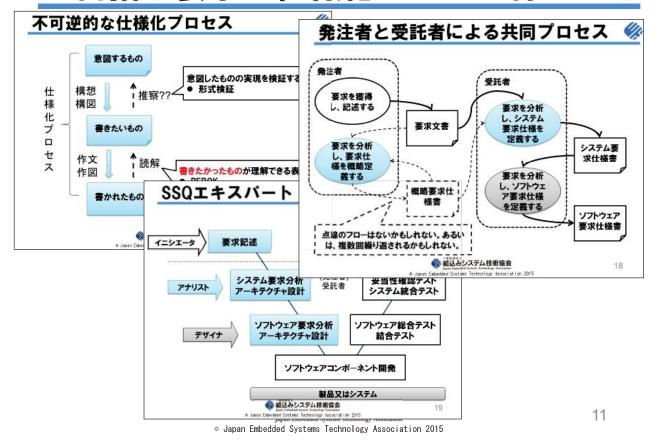
*** 組込みシステム技術協会

Japan Embedded Systems Technology Association

Japan Embedded Systems Technology Association 2015

2.目指す要求の仕様化プロセス像





3.要求の仕様化を支援するプロセスと手法



要約

- ▶ 書かれた仕様から意図するものを容易に再現できるためには、仕様化プロセスに工夫が必要である。その工夫を委員で分担して試みた。何を狙いとし、どのプロセスと手法を選択したかは、「仕様記述実験一覧」を参照。その中から次の4件を説明する:
 - システム要求設計技法による安全要求の
 - 物物関係分析法による安全制約の記述
 - 操作シナリオベース開発手法
 - SPINを活用する仕様検証

仕様記述実験一覧



狙い	題材	プロセス	手法	節番号
顧客合意		操作シナリオベース	VDM++	
意図の検証	車速制限		SPIN	
安全要求	電気ボット	システム要求設計技法	SysML	3.1
制約事項	イベント登録管理	物物関係分析法	VDM++	
安全制約	電気ポット	物物関係分析法	VDM++	3.2
仕様検証	モータのインバータ		SPIN	3.4
安全要求	電動支援自転車	操作シナリオベース開発	VDM++	3.3

● 組込みシステム技術協会 ション Failed Add System Technology Association as: Embedded Systems Technology Association

12

16

🥢 組込みシステム技術協会



仕様化を支援するプロセスと手法の一つ

システム要求設計技法による安 全要求の記述



13

システム要求設計技法の特徴



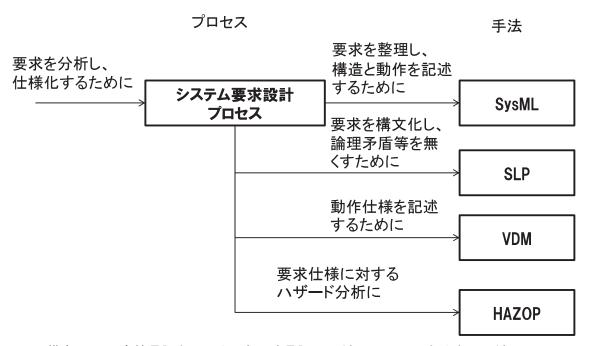
- システム又は製品に対する要求をもとに して、システムの構成要素に対する要求 を定義するプロセスを示す。
- 標準的な開発プロセスモデルにおけるシ ステム要求分析とシステムアーキテク チャ設計に適用できる。
- 表記法としてSvsMLを活用する。

RADT: システム要求設計技法



プロセスと手法





備考:SLPは自然言語を用いる要求記述言語の一種。VDMは形式手法の一種。



Japan Embedded Systems Technology Association 2015

15

安全要求を記述する手順



- 1. 要求分析
 - 1. 要求を獲得する
 - 2. システムとその境界を決める
 - 3. システムの機能を定める
 - 4. ユースケースの動作を表現する
- 2. アーキテクチャ設計
 - 1. システムを構成要素に分解する
 - 2. 部品の相互作用を定義する
 - 3. 部品の相互接続を定義する
- 3. ハザード分析
- 4. 安全に関する要求定義
- 5. 要求分析とアーキテクチャ設計を繰り返す
- 6. ハザード分析の確認



(1) 要求分析



- 1. 要求を獲得する
 - 1. 電気ポットに関する要求を製品企画部門から入手する。
 - 2. 与えられた要求を上位方向に抽象化し、下位方向に詳細化し、過不足ないように要求図に整理する。
- 2. システムとその境界を決める
 - 1. 電気ポットの境界をシステムコンテキストとして表現する。
- 3. システムの機能を定める
 - 1. 操作に関する要求からユースケースを洗い出す。
- 4. ユースケースの動作を表現する
 - 1. 電気ポットの全体動作を状態機械図で記述する。
 - 2. 各状態における動作(振舞い)をアクティビティ図で記述する。



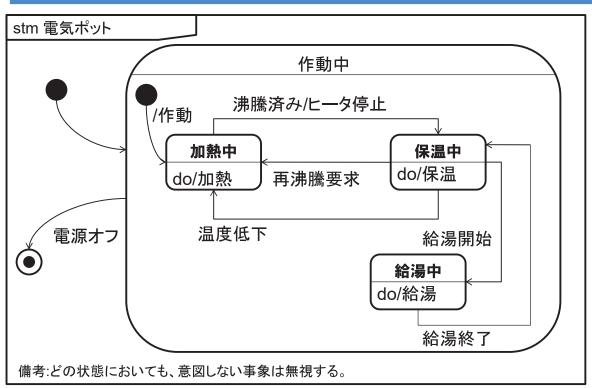
Japan Embedded Systems Technology Association

Japan Embedded Systems Technology Association 2015

17

状態機械図





※ 組込みシステム技術協会

Japan Embedded Systems Technology Association 2015

(2)アーキテクチャ設計



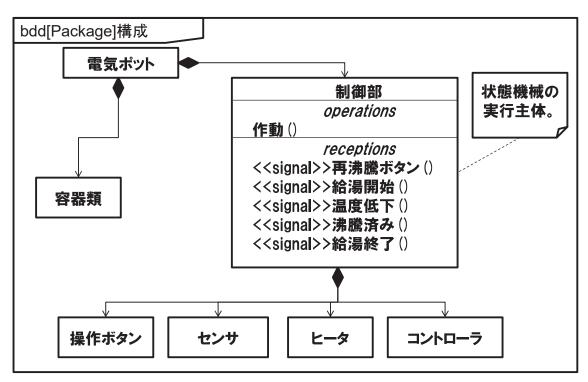
- 1. システムを構成要素に分解する
 - 1. 電気ポットを制御部と容器類に分け、制御 部を構成する部品を記述する。
- 2. 部品の相互作用を定義する
 - 1. アクティビティ図における各処理(アクショ ン)を、制御部を構成する部品に割当てる。
- 3. 部品の相互接続を定義する
 - 1. 制御部を構成する部品間をポートで接続し、 部品間のデータを記述する。



19

システム構成



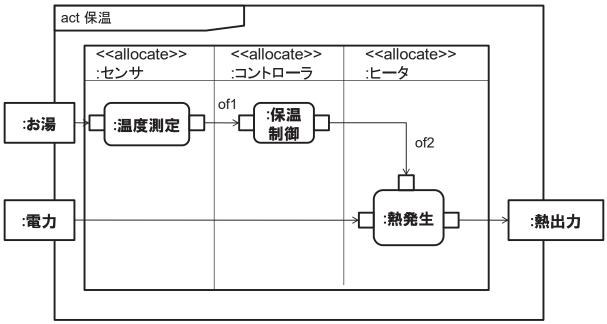


組込みシステム技術協会

Japan Embedded Systems Technology Association 2015

処理の割当て





備考:保温アクティビティを構成する処理をシステムを構成する部品に割当てる。



Japan Embedded Systems Technology Association

Japan Embedded Systems Technology Association 2015

21

(3) ハザード分析と安全要求定義



- 1. ハザード分析
 - 1. 電気ポットの全体動作を示す状態機械に対して、HAZOP手法を用いてハザード分析を行う。
- 2. 安全に関する要求定義
 - 1. すべての識別されたハザードに対して、その安全要求を定義する。



HAZOPによるハザード分析(一部)



状態遷移: 加熱中 → 保温中

事象: 沸騰済み

ガイドワード	原因と状況	結果と対策
No	沸騰に達したが、沸騰済み事 象を発生しない。	加熱中状態で加熱が続く。吹き出 すかもしれないし、水量が減って、 空だきになるかもしれない。
As well as	沸騰していないけど、沸騰済 みを誤検出。	保温中へ遷移し、沸騰に至らない まま、保温処理を行う。
Part of	ヒータを止めることができずに 保温中へ遷移する。	保温中状態で加熱が続くが、保温 処理がヒータを止めるはず。
Other than	AS well asに同じ	



23

(4) 要求分析と設計の再実施



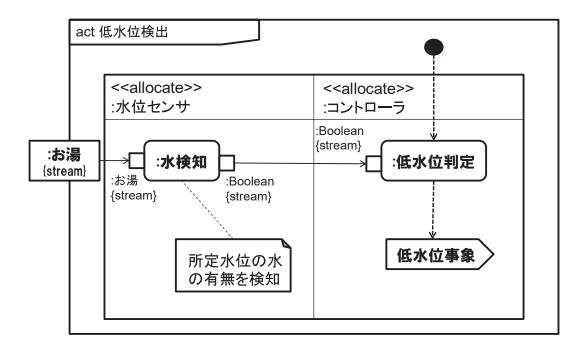
1. 要求分析

- 1. 電気ポットの動作に安全要求を反映し、それに 対応して状態機械を見直す。
- 2. 追加される低水位検出のための動作(振舞い)に関して、その処理とデータの流れを記述する。
- 2. アーキテクチャ設計
 - 1. システム構成に水位センサを追加する。
 - 2. 水位センサとコントローラに追加された処理 (ア クション) を割当てる。
- 3. ハザード分析の確認
 - 1. ハザード分析を再実施し、対策後にハザードがないことを確認する。



追加される処理の割当て







Japan Embedded Systems Technology Association 2015

25

考察



■ 要求定義手法

- 電気ポットの与えられた要求から、要求分析と アーキテクチャ設計を行って、その結果として 要求を記述した。
- 提供する機能をシステムの構成要素に割当てる には、この手法は有効である。

■ SysML

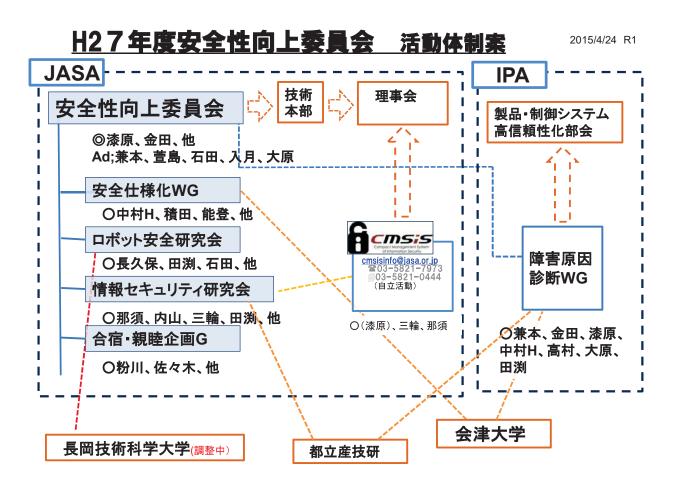
- システムの要求を体系的に整理できるので、漏れを防ぐには有効である。
- ハードやソフトにとらわれずに、システムを要素に分解でき、この構成要素によってシステムの機能を記述できることがわかった。
- ・ これで書かれた設計資料はハザード分析に役立つことを確認した。





今年度の活動体制





安全仕様化WG 活動概要



- 目的
 - ・ 安全に関する要求や制約等を仕様化するプロセスを研究し、 その仕様化を支援するプロセス又は手法を提案すること。
- 方針
 - ・ 課題選定は自主的に、活動は計画的に。
- 取組む課題として
 - ・ 正確に仕様を記述する手法
 - 一 自然言語(SLP、、)
 - 一 半形式手法(モデルベース、、)
 - 一 形式手法(VDM、、)
 - ・ 仕様を検証する手法(SPIN、STAMP/STPA、、)
 - ・ 意図を記述するプロセスと手法(SvsML、GSN、、)
 - 合意や理解等を得るためのプロセスと手法 (SysML、GSN、、)
- 活動として
 - ・ 仕様記述実験に適する題材の選定又は作成
 - ・ プロセス又は手法に関する委員からの報告や提案等
 - ・ 有識者を招いての勉強会



Japan Embedded Systems Technology Association 2015

29

ロボット安全研究会 活動概要



- 目的
 - ・ 生活支援ロボットの機能安全の講師育成・テキスト 作成
 - ロボットの安全開発に関して、ソフトウェア面から 貢献
- 方針
 - ・ まずは、ISO13482を手始めとして関連規格の把握
- 取組む課題として
 - ・ ソフトウェア面から見た、ロボットの安全開発
- 活動として
 - ISO13482の輪講会(2015年度1Q完了予定)
 - · 関連規格の勉強会(2015年度上期予定)
 - ・ 有識者を招いての勉強会



情報セキュリティ研究会 活動概要



背景

情報セキュリティ事故はとどまるところを知らない。近年はネットワークシステムばかりではなく、自動車や社会インフラの組込みシステムに関する情報セキュリティ対策も必要性が叫ばれている。

しかし情報セキュリティは技術的にも、その講ずべき対策も、難しい、よく分からないと言われている。

活動内容

そこで、情報セキュリティ事故を取り上げ、そこでなすべきであった対策を具体的に解明することで、情報セキュリティの技術やその対策に理解を深める。事故から学ぶ、情報セキュリティのいわば「事後V&V」(検証と妥当性の検討)である。

アプローチの視点

情報セキュリティには、経営組織が守るべきものと、製品やシステムが守るべきものとがある。前者は**経営組織的な対** 策による防御であり、後者は**技術的な対策**による防御である。

情報セキュリティにはすでに国際標準規格があり、前者が<mark>ISO/IEC 27000</mark> 系(ISMS)であり、後者が<mark>ISO/IEC15408</mark>で ある

当WGではこれら両視点を学びながら、事故へのなすべきであった(事後)対策に理解を深めて行く。

活動の方法

メンバーによる定期的な調査研究活動に加え、有識者からの指導を得、また有識者による講演会などから知見を広める。

期待される成果

調査研究成果をもとに組合員各位、ならびに広く国内中小企業等に情報セキュリティ対策を提供できるようにする。

直近のこととして、すでにある経営組織の情報セキュリティ対策規格としてのJASA規格のCMSiSをより簡便にしてほしいという要望があり、その規格の作成を行う。



31

© Japan Embedded Systems Technology Association 2015

ご清聴ありがとうございました



要求の仕様化に関する活動の最終報告

2015/5/20 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。





レガシーコードの蘇生術 ~リバースモデリングによる状態遷移表の作成~

2015年5月20日 状態遷移設計研究会



状態遷移設計研究会とは



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



© Japan Embedded Systems Technology Association 2015

現在の組込みソフト開発の状況



■ソフト開発の傾向

- ▶ 短納期・高品質の要望
- ▶レガシーコードの肥大化・複雑化
- > メンテナンス性の低下
 - ⇒設計ドキュメントがない!
 - ⇒担当者も、もういない!
 - ⇒改修したら、違うところでBUGがでた!!





ソースコードのブラックボックス化が進行中!

 3

新たな取り組みに移行できない背景



効率化を謳った手法は、様々ある。

モデルベース開発、ソフトウエアプロダクトライン開発など しかしながら、開発プロセス、設計手法の移行は進まない。

レガシーコードの存在が足かせになっている!

派生開発が主流の開発で、既存コードの設計を踏襲せざるを得ない。

そこで、

■リバースエンジニアリング

今、注目を集めているのが「リバース・エンジニアリング」 リバース・エンジニアリングとは、現行のシステム(ソースコード やデータベース)を解析し、その仕様を明らかにする技術

研究テーマ



■【仮説】

「フラグがある所に、状態はある!!」

■ 研究テーマ

「状態遷移表のリバースモデリングへの適用」

⇒レガシーコードから状態を抽出し、

状態遷移表モデルを導出する手法の研究。

⇒状態遷移表が導出できれば、それをベースにして レビュー・リファクタリング・修正の検討ができる!









5

本研究の対象範囲



■【リファクタリングの流れと研究対象範囲】

ソースコード

リバース エンジニアリング

再作成 (修正計画含む)



本研究の対象

ソースコード前処理

フローチャート作成

条件処理表の作成

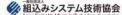
状態変数の抽出

状態遷移表の作成

本研究の対象外

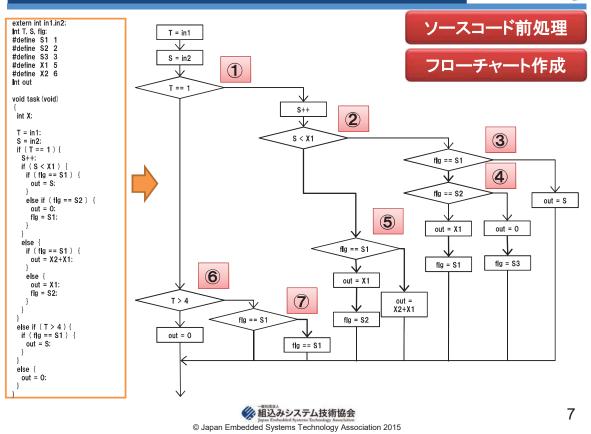
状態遷移表の洗練

レビュー(検討)



© Japan Embedded Systems Technology Association 2015





リバースモデリングの作業手順



条件処理表の作成

条件				処理
無条件実行				T=in1,S=in2
T==1 1				S++
	S <x12< th=""><th>flg==S1</th><th>3</th><th>(Out=S)</th></x12<>	flg==S1	3	(Out=S)
		flg==S2	4	(Out=O),flg=S3
		else		S++,flg=S1
	else	5 flg==\$1		(Out=X2+X1)
		else		(Out=X1),flg=S2
T>4 6	7 flg==S1			(Out=S)
	else			(Out=0)

ソースの 分岐条件から 条件を左、 処理を右、 に置いた仮の 状態遷移表を 作成

フローチャートの
①~⑦の構成と、
条件処理表の
①~⑦の構成は
同じになる



状態変数の抽出

- 「条件」で見ている変数から、状態変数を抽出 「状態変数の定義」に合致するか、で特定
 - 1. 状態変数は有限個である。
 - 特定の状態を示すもの、無限はあり得ない
 - 2. 状態変数は、内部で更新される。
 - イベントは、外部から渡される
 - 状態は、イベントを受けて内部で更新される



9

リバースモデリングの作業手順

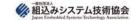


「条件」で参照する変数は「T」「S」「flg」

状態変数の抽出

	条件		処理	
	無条件実行		T=in1,S=in2	
T==1			S++	「T」「S」は、外部の値
	S <x1< th=""><th>flg==S1</th><th>(Out=S)</th><th>を代入しているだけ</th></x1<>	flg==S1	(Out=S)	を代入しているだけ
		flg==S2	(Out=0),flg=S3	
		else	S++,flg=S1	「flg」は、内部処理で S1/S2/S3 のいずれ
	else	flg==S1	(Out=X2+X1)	かが設定される
		else	(Out=X1),flg=S2	2
T>4	flg==S1		(Out=S)	
	else		(Out=0)	

→「flg」が状態変数だ!





状態変数「flg」の判定式で、「else」となっている個所を有限個の取りうる値に展開

状態遷移表の作成

T T T	11 G 13 24 E	074X 7 7 8 1E 1C	124 1213
	条件		処理
	無条件実行	÷	T=in1,S=in2
T==1			S++
		flg==S1	(Out=S)
	S <x1< td=""><td>flg==S2</td><td>(Out=0),flg=S3</td></x1<>	flg==S2	(Out=0),flg=S3
		flg==S3	S++,flg=S1
		flg==S1	(Out=X2+X1)
	else	flg==S2	(Out=X1),flg=S2
		flg==S3	(Out=X1),flg=S2
T>4	flg	==\$1	(Out=S)
	flg==S2		/
	flg	==\$3	/
	else		(Out=0)



11

リバースモデリングの作業手順



状態変数「flg」を列の要素に移動

状態遷移表の作成

	条件		flg	
		S1	S2	\$3
無条	件実行	T=in1,S=in2	=in1,S=in2	T=in1,S=in2
T==1		S++	S++	S++
	S <x1< th=""><th>(Out=9</th><th>/</th><th>/</th></x1<>	(Out=9	/	/
			(Out=0),State=S3	/
		/	/	S++,State=S1
	else	(Out=X2+X1)	/	/
		/	(Out=X1),State=S2	/
		/	/	(Out=X1),State=S2
T>4		(Out=S)	/	/
		/	/	/
		/	/	/
Н	else	(Out=0)	(Out=0)	(Out=0)

組込みシステム技術協会

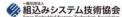


状態遷移表の作成

条件		flg				
		S1	S1 S2			
無条件	牛実行	T=in1,S=in2	T=in1,S=in2	T=in1,S=in2		
T==1		S++	S++	S++		
	S <x1< th=""><th>(Out=S)</th><th>(Out=0),flg=S3</th><th>S++,flg=S1</th></x1<>	(Out=S)	(Out=0),flg=S3	S++,flg=S1		
	else	(Out=X2+X1) (Out=X1),flg=S2 (Out=X1)		(Out=X1),flg=S2		
T>4		(Out=S)	/	/		
el	se	(Out=0)	(Out=0)	(Out=0)		

うん、 状態**遷移表** になった。

これで、やっとレビューができる!!



© Japan Embedded Systems Technology Association 2015

13

サンプル検証



■ 新人研修での実習(スロットマシンの作成)で、
新人の作成したプログラムを検証



- スロットは3つ、ボタンは1つ
- 起動直後は3つのスロットがすべて回転
- ・ すべて回転しているときに、掛け金設定
- 1回目のボタン押下で、掛け金を設定し、 一番左のスロットを止める
- 2回目のボタン押下で、真ん中のスロット を止める
- 3回目のボタン押下で、右のスロットを止めると同時に、止まった3つの絵柄を比較し、その結果でもち金を増減
 - 3つとも同じ→掛け金5倍
 - 2つ同じ →掛け金そのまま
 - ・ すべて異なる→掛け金没収
- 4回目のボタン押下で、3つのスロットを すべて回転している状態に戻す

《組込みシステム技術協会

サンプル検証1(意味不明→仕様明確化)



	関数	条件	flag_b1_c	状態	状態を0~3の直値で指定		
わ	状態変数名が意味不明		0	1	2	3	
	Entry4	L,M,Rがすべ て一致	Flag_b1_c = 1	Flag_b1_c = 2	Flag_b1_c = 3 Total += Bet*5	Flag_b1_c = 0	
		L,M,Rの2つ が一致	Flag_b1_c = 1	Flag_b1_c = 2	Flag_b1_c = 3 Total += Bet	Flag_b1_c = 0	
		else	Flag_b1_c = 1	Flag_b1_c = 2	Flag_b1_c = 3	Flag_b1_c = O	

条件は 状態2 のときのみ有効



15

サンプル検証2(バグが検出された)



・状態を++で加算している、条件追加など修正が困難

PressCour	PressCounter						
0	1		2	3		4	
PressCounter ++	PressCounter++ PressCo		PressCounter ++	PressCounter + +		PressCounter++	
	無条件	bAmount = (int) p3		L,M,Rがす べて一致	Amount += (bAmount*5)	Amount > 0	PressCounter = 0
	bAmount > Amount	bAmount = Amount		L,M,Rの2 つが一致	Amount += bAmount		
	無条件	Amount = bAmount		else	/	else	/

Amount <= 0 だと PressCounter が 0 に戻らない!



サンプル検証3(実装のブラッシュアップ)



関数	条件	g_pushStatus						
		Rolling	Stop1	Stop2	Stop3	Hit2	Hit3	Miss
Entry_Calc ulate			/	g_pushStat us = Hit3 Total += Bet*5		/ 態の粒度: p3状態で		
	L,M,Rの2 つが一致	/	/	/	g_pushStat us = Hit2 Total += Bet		8条件と	
	else	/	/	/	g_pushStat us = Miss	/	/	/
ENTRY_Ch argeStatus		g_pushStat us = Stop1	g_pushStat us = Stop2	g_pushStat us = Stop3	/	g_pushStat us = Rolling	g_pushStat us = Rolling	g_pushStat us = Rolling

● 編込みシステム技術協会 場立のシステム技術協会 © Japan Embedded Systems Technology Association 2015 17

適用効果の考察



- 1. 機械的にソースコードから状態遷移表を作成するプロセスが見えてきた。
- 2. 振る舞いの可視化、モデル化によりレビュー がしやすくなる。

振る舞いの漏れ・抜けが発見できる。

- 3. 状態変数名の変更など、リファクタリングの要素を抽出できる。
- 4. 言語に依存せず、すべてのコードに適用できる。



状態遷移表への変換パターン洗い出し



- 先の検証により、仮説の裏付け、手順が適用できることは証明できたが、様々なパターンに適用できるかの検証が不足
- 実際に変換に迷うパターンを抽出して検討
 - 1. 状態の階層化パターン
 - 2. 状態変数の演算パターン
 - 3. 状態変数の置き換えパターン
 - 4. 状態変数の関数渡しパターン
 - 5. 状態の並列化パターン

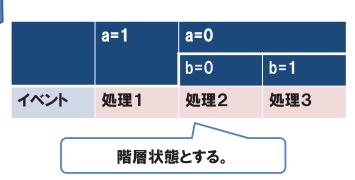


1. 状態の階層化パターン



■ if-else で条件が階層化されている場合 →状態遷移表も階層構造とする

条件	処理	
If (a==1)	処理1	
else If (b==0)	処理2	
else	処理3	





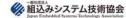
2. 状態変数の演算パターン



- and や or で演算してから判定している場合
 - → 演算式を状態変数として判定

条件	処理
If (a==1&&b==1)	処理1
If $(a==1\&\&b==0)$	処理2
If (a==0&&b==1)	処理3
If $(a==0\&\&b==0)$	処理4
• • •	

	a=1&&b=1	a=1&&b=0	a=0&&b=1	a=0&&b=0
イベント	処理1	処理2	処理3	処理4



© Japan Embedded Systems Technology Association 2015

3. 状態変数の置き換えパターン



A&BをCに代入して、Cで判定。 でも、変数の変更はA,Bで行っている場合。

C=A&B				
	C=0?	A=2 B=1		
	C=1?	A=2 B=0		
	C=2?	A=0 B=1		
	C=3?	A=0 B=0		



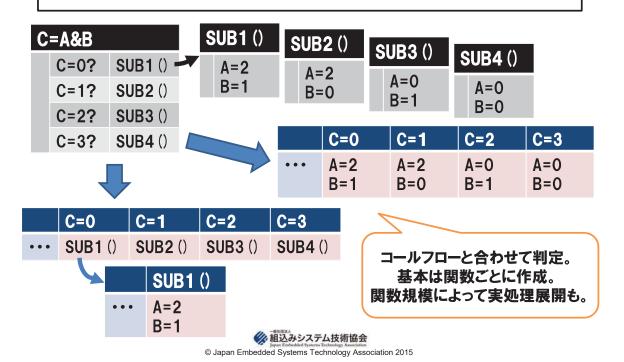
C代入時の式も条件に記載

	C=A&B				
	C=0	C=1	C=2	C=3	
• • •	A=2 B=1	A=2 B=0	A=0 B=1	A=0 B=0	

4. 状態変数の関数渡しのパターン



A&BをCに代入して、Cで判定。 変数の変更は関数内で行っている場合。



5. 状態の並列化パターン



条件	処理			a=1	a=0
If (a==1)	処理1			ŭ .	u o
else	処理2		イベント	処理1	処理2
• • •					
If (b==1)	処理3			b=1	b=0
If (b==1) else	処理3 処理4	7	イベント	b=1 処理3	b=0 処理4

基本は異なるSTMとして処理する。

並列状態とするかは、イベント処理の位置で判断する。

		a=1		b=1	b=0	
	イベント	処理1	処理2	処理3	処理4	

2015年度 活動予定



① ツール化活動

• enPIT/Emb(分野・地域を超えた実践的情報教育協働ネットワーク組込みシステム分野)を活用した、OJLによるツール開発。

② 普及活動

出張セミナー、出張コンサルサービスの展開。リバース手順の無償レクチャー。

③ ガイドライン

キャプチャー動画 or スライドショー形式による 操作手順(リバース手順)のガイド公開。



①ツール化活動



- リバースモデリングツールの検討・作成
 - 自動でできる作業はツールで実行
 - 一 ソースから条件・処理対応表(条件処理表)を生成
 - 一 状態変数候補を抽出、選択→状態遷移表に展開
 - 状態変数の特定など人力が必要な作業に集中
 - 一 現場で導入がしやすくなる、検証してもらえる
- 2014年度はツール要件固めを進めた
- 2015年度は、enPIT を利用して実際に作成



平成27年度 技術本部成果発表会 「レガシーコードの蘇生術~リバースモデリングによる状態遷移表の作成~」

2015/05/20 発行

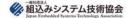
発行者 一般社団法人 組込みシステム技術協会

東京都中央区日本橋浜町1丁目8-12

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。



© Japan Embedded Systems Technology Association 2015



JASA技術本部成果発表会

ロボット向け組込みソフトウェアライブラリ



2015年5月20日 プラットフォーム研究会 古俣 学



目次



- 1. プラットフォーム研究会の紹介
- 2. 日本の成長戦略と研究会の取り組み
- 3. OpenEL®
- 4. 全体計画
- 5. 平成26年度の活動結果
- 6. 平成27年度の活動計画
- 7. まとめ





1. プラットフォーム研究会の紹介



3

概要



- 設立: 2000年
- 目的: プラットフォームの事例、特性等の評価、利活用の指針策定
- リーダー
 - □ 中村憲一
- アドバイザー
 - □ 神徳徹雄

独立行政法人産業技術総合研究所 イノベーション推進本部 イノベーション推進企画部 総括企画主幹

□ 武居直行

首都大学東京 准教授 システムデザイン学部 工学博士

□ 佐野明人

名古屋工業大学 教授 機能工学専攻 日本機械学会フェロー 工学博士

□ 古荘純次

福井工業大学 教授 機械工学科 博士(工学)

□ 金田光範

(地独) 東京都立産業技術研究センター 産学官連携コーディネーター

■ メンバー/参加企業・団体

アップウィンドテクノロジー・インコーポレイテッド

図研エルミック株式会社

株式会社コア

株式会社エヌデーデー

株式会社セントラル情報センター

(地独)東京都立産業技術研究センター

東芝システムテクノロジー株式会社

株式会社ビッツ

株式会社イーシーエス

日本システム開発株式会社

イーソル株式会社

株式会社パトリオット

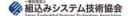
大宮技研合同会社

オリエンタルモーター株式会社

株式会社KSK

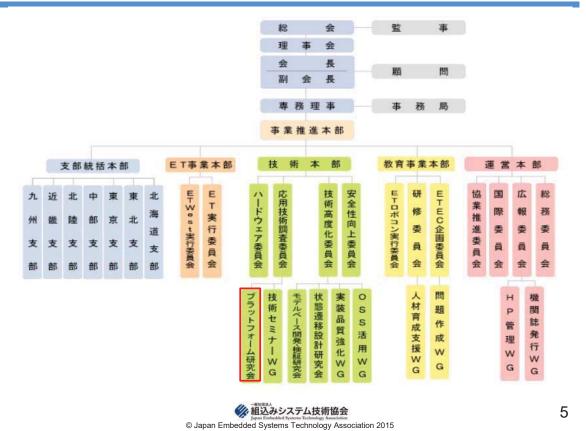
第一精工株式会社

株式会社アックス



JASA内での位置付け





研究会のテーマ



我が国の成長戦略である「ロボット新戦略」に基づき、ロボット技術の課題であるデバイス制御の標準化を推進する。

アプリケーションソフトウェアからAPIでデバイス制御を可能にする;

- 技術開発
- この技術を基盤にした国際標準化

を推進する。

ロボット新戦略: (第6回ロボット革命実現会議(2015年1月23日経済産業省)) 分野横断的テーマ(ロボット国際標準化)への対応





2. 日本の成長戦略と研究会の取り組み



7

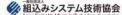
ロボット新戦略 (第6回ロボット革命実現会議(2015年1月23日経済産業省))



ロボット新戦略では3つの柱を推進

- 1. 日本を世界のロボットイノベーション拠点とする「ロボット創 出力の抜本強化」
- 世界一のロボット利活用社会を目指し、日本の津々浦々に おいてロボットがある日常を実現する「ロボットの活用・普及 (ロボットショーケース化) |
- ロボットが相互に接続しデータを自律的に蓄積・活用するこ とを前提としたビジネスを推進するためのルールや国際標 準の獲得等に加え、さらに広範囲な分野への発展を目指す 「世界を見据えたロボット革命の展開・発展」

「ロボット新戦略」(ロボット革命実現会議とりまとめ)http://www.meti.go.jp/press/2014/01/20150123004/20150123004.html



ロボット国際標準化と研究会の技術的取組



分野横断的テーマ(ロボット国際標準化)への対応

国の考え方

- 既存のロボットは特定の用途に特化しているため再利用ができず、高コスト
- ハードウェア、ソフトウェアをモジュール化・共通化することで多種多様なロボットに部品を使用することができ、これにより低価格なロボットを構成
- このような仕組みを実現するには、ハードウェア・ソフトウェアともに多くのメーカーが作る部品同士をつなぐインターフェースをいかに共通化、標準化するかが重要



JASAの活動

研究会ではOpenEL®を開発し、OMG(Object Management Group)へ HAL4RT(Hardware Abstraction Layer for Robotic Technology)として国際標準化を提案、さらにISOへ提案予定



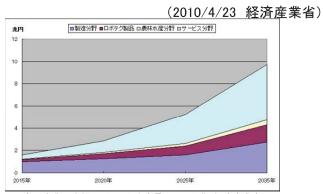
9

市場規模



■「2035年に向けたロボット産業の将来市場予測」

2015年 1.6兆円 2020年 2.9兆円 2025年 5.3兆円 2035年 9.7兆円



ロボット産業の市場予測(国内生産量) (※出典:経済産業省/NEDO)

■「2012年ロボット産業の市場動向」

(2013/7/18 経済産業省)

産業用ロボットの2011年の市場における日本企業のシェアは50.2% これに基づき、本標準規格を世界の企業の50%が採用すると仮定すると、国際市場獲得効果は、2015年時点で少なくとも8,000億円を目指したいところである。



3. OpenEL®

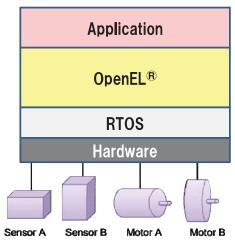


11

OpenEL®とは

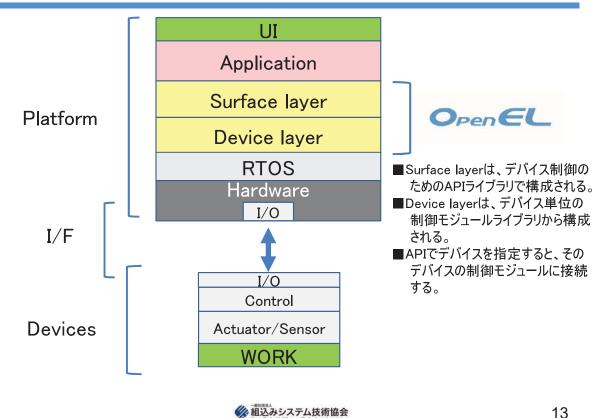


- OpenEL® (Open Embedded Library)
 - ロボットで用いるセンサやアクチュエータなどのデバイスを制御する仕組みで、ライブラリ構造になっている。







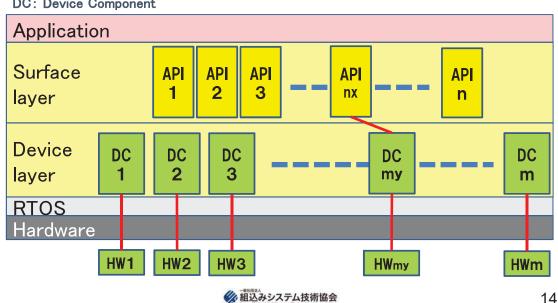




あるAPInxを用いデバイスID(my)を指定すると、APInxはDCmy に接続し、デバイスであるHWmyに、APInxで定義されている仕事 をさせる。

© Japan Embedded Systems Technology Association 2015

DC: Device Component



© Japan Embedded Systems Technology Association 2015



4. 全体計画



15

標準化戦略



- JASAは3段階のステップで国際標準を推進
 - ステップ1 (国内の業界標準)
 - JASA標準としてOpenEL® 2.0を策定し、アーキテクチャーを確立
 - ステップ2 (北米を中心とした業界標準)
 - ー OpenEL®をベースに、OMG/Robotics-DTF(Domain Task Force)にてHAL4RT標準化を推進
 - ステップ3 (世界標準)
 - 一 OMGとISOのリエゾン推進し、HAL4RTをISO TC184/SC2に提案するとともに、ISO TC184/SC2/WGXX国内委員会を立ち上げ、日本発の提案としてISO国際標準化を推進



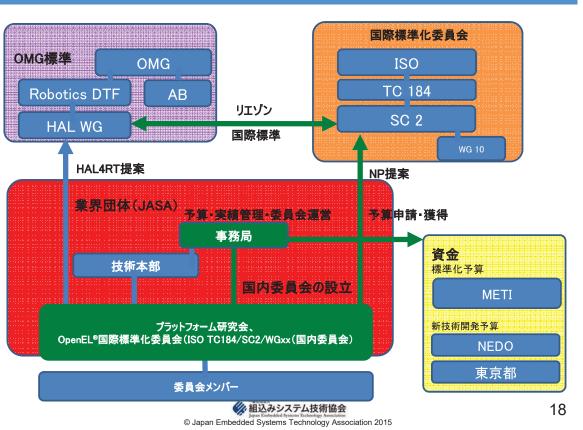
国際標準化シナリオ 2013 2014 2015 2016 2017 2018 OpenEL® アーキテクチャを確立 ステップ V2.0 ▼ V1.1 ▼ V1.0 メンテナンス ロボットWG OpenEL®をベースに、HAL4RTの標準化 NP RFI RS 採択 **RFP** 公開 メンテナンス HAL WG FTF手順(公文書作成) HAL4RTをベースに、日本主導によるISO国際標準化 メンテ NP提案 投票 WD CD DIS ナンス FDIS IS 承認 ISO/TC184/SC2/

組込みシステム技術協会

© Japan Embedded Systems Technology Association 2015

国内•国際組織間関係







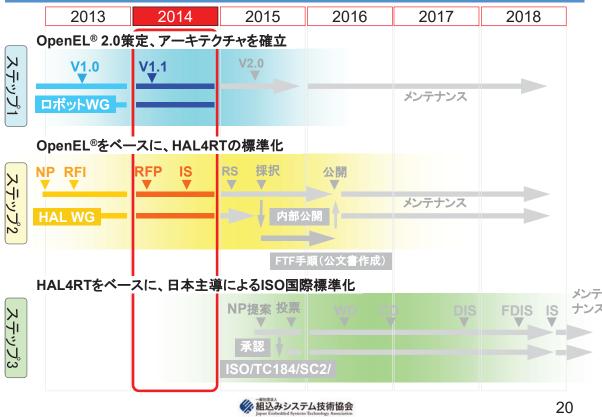
5. 平成26年度の活動結果



19

平成26年度の活動報告 2013 2014 2015





© Japan Embedded Systems Technology Association 2015

平成26年度主たる活動記録



- A) JASA/研究会・ロボットWG開催(毎月1回 年間12回)
- ⇒ OpenEL® 2.0の仕様策定作業
- ⇒ HAL4RT提案書作成作業
- B) OMG国際会議出席(平成26年6月、9月、12月、平成27年3月)
- C) JASAからOpenEL®の仕様をOMGに提出し、HAL4RTのOMG標準 化を推進
- ⇒ 2014 (H26) 年 6月: OMGでHAL4RTの標準提案公募(RFP)発行
- ⇒ 2014 (H26) 年 9月: OMG/CHAL4RT RFPの提案登録(LOI)を提出
- □ 2014 (H26) 年12月: OMGにHAL4RT RFPへの一次提案を提出
- □ 2015 (H27) 年 3月: OMGにHAL4RT RFPへの二次提案を提出
- E) ISO/TC184/SC2とリエゾン
- F) 平成27年度工業標準化推進事業委託費を申請し、予算獲得
- G) 普及·啓発活動
- H) 他委員会との連携
- ⇒ 安全性向上委員会、ハードウェア委員会



21

予算獲得



■ 予算元: 経済産業省

平成27年度工業標準化推進事業委託費(戦略的国際標準化加速事業)

テーマ

ロボット向け組込みソフトウェアに関する国際標準化

事業期間

3年

委託予算額

10,000千円(上限)/初年度

「平成27年度戦略的国際標準化加速事業」に係る政府戦略分野、社会ニーズ等分野の国際標準開発を実施する事業者の公募のご案内について http://www.mri.co.jp/news/press/public offering/recruit/017935.html



普及•啓発活動



- 国際ロボットカンファレンス2014
 - 日経エレクトロニクスと共催
 - 参加人数:155名
- ET2014
 - ロボットセッション
 - テーマ: ロボット技術における国際標準化の動向
 - ・ 参加者: 91名 技術本部セミナー
 - テーマ: 日本発の国際標準を狙う次世代のロボット開発プラットフォームOpenEL®
 - 参加者: 54名
 - プレゼンシアター
 - テーマ: 日本発の国際標準/次世代のロボット開発OpenEL®
 - 参加者: 28名/30名
 - JASAブース
 - OpenEL®を使ったジャイロセンサのデモ







© Japan Embedded Systems Technology Association 2015



6. 平成27年度の活動計画

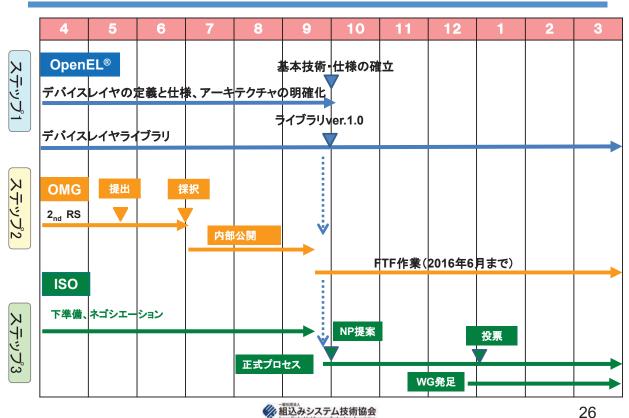
平成27年度の活動計画 2013 2014 2016 2017 2018 OpenEL® 2.0策定、アーキテクチャを確立 ステップ V1.0 メンテナンス ロボットWG OpenEL®をベースに、HAL4RTの標準化 RS NP RFI **RFP** IS 公開 メンテナンス HAL WG FTF手順(公文書作成) HAL4RTをベースに、日本主導によるISO国際標準化 NP提案 投票 **FDIS** 承認 ISO/TC184/SC2/

《組込みシステム技術協会

© Japan Embedded Systems Technology Association 2015

平成27年度活動計画スケジュール





平成27年度の活動計画



- A) JASA/研究会・ロボットWG開催(毎月1回、年間12回)
- ⇒ OpenEL®の仕様策定作業
- B) OpenEL®の実証実験(国内外)
- C) OMG国際会議出席(平成27年6月、9月、12月、平成28年3月)
- D) JASAからOpenEL®の仕様をOMGに提出し、OMGでHAL4RTの国際標準化を 進める。
- ⇒ 2015 (H27) 年 5月:OMG HAL4RT RFP二次提案締切
- ⇒ 2015 (H27) 年 6月:標準提案の採択(OMG内部公開)
- ⇒ 2015 (H27) 年 9月: FTFによる公文書作成
- ⇒ 2015 (H27) 年12月: FTFによる公文書作成
- ⇒ 2016 (H28) 年 3月: FTFによる公文書作成
- ⇒ 2016 (H28) 年 6月: OMG標準仕様の発行(一般公開)
- E) ISO国際会議出席(平成27年6月、10月、平成28年2月)
- ⇒ OMGで進めるHAL4RTのドラフト承認が得られた時点で、これをISO TC184/SC2 にNPとして提出し、ISO国際標準化を開始する。

FTF: Finalization Task Force



27

課題



- 活動予算の確保
 - 標準化予算獲得: METI 標準化支援予算継続
 - 技術開発予算獲得: NEDO、TIRI、その他
- 体制の強化
 - JASA/委員会事務局の強化
 - 一 METI等に対する予算申請、予算管理
 - 国際委員会に対応するJASAの役割
 - 一 OMG/HAL4RTの役割: 国内委員会の担当団体(事務局)
 - ISO/TC184/SC2の役割: ISO国内委員会の担当団体(事務局)
- OpenEL®の普及・啓発
 - ドキュメントの整備





7. まとめ



29

まとめ



- 昨年度は計画に従った所定の成果を達成
 - JASA/OpenEL® 2.0の策定の推進
 - OMG/HAL4RT提案書策定の推進
 - ISO/TC184/SC2への参加準備
 - 経済産業省から国際標準化予算の獲得
- 今年度も、ロードマップに従って計画を遂行



ご清聴ありがとうございました。

2015/5/20 発行

ロボット向け組込みソフトウェアライブラリ OpenEL® 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8-12 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。





JASAの対外アピールについて ~クミコちゃんを中心に~

> 2015年5月20日 技術セミナーWG (広報委員会) 冨岡 理



編込みシステム技術協会 Japan Embedded Systems Technology Association

© Japan Embedded Systems Technology Association 20YY

目次



- 盛り上がるロボットセミナー (技術セミナーWG)
- ミライ・グミコ クミコ・ミライ をどうぞよろしく (広報委員会)
- 日経テクノロジーオンラインへの連載コラム (広報委員会)



国際ロボットカンファレンス報告



~ロボットと人の共生へ道筋を探る~

2015年8月29日 JA共催ビル

「人とロボットの未来」

大阪大学 石黒氏

「浸透する人工知能ロボット技術:第三次ロボットブームは本物か?」

ソニーコンピュータサイエンス研究所 北野氏

「人型ロボットのための制御ソフトウェア『V-Sido』」

水道橋重工 吉崎氏

「Autonomous Data Machines」

米Knightscope社 William Santana Li氏

「スーパー飛行ロボットを目指したミニサーベイヤーの現在と未来」 千葉大学 野波氏

参加人数:128名(一般109名、JASA会員9名、招待客10名)



クミコ・ミライ



応用技術調査委員会 技術セミナーワーキンググループ | 冨岡 理氏

JASA の対外アピールについて~クミコちゃんを中心に~

技術セミナーWG と広報委員会は、合同で活動し、JASA の対外アピールを担って 昨年から今年にかけての3つの大きな活動を報告する。

概要

(1)成り上がるロボットセミナー

(2)ミライ・クミコをどうぞよろしく

(3)日経アクノロジーオンラインへの連載コラム

間違い



JASAの紹介(動画)

https://www.youtube.com/watch?v=F2dzgMOzjPw

組込み業界の紹介(動画)

https://www.youtube.com/watch?v=pzpGFZ4k5HI





日経テクノロジーオンラインへの連載コラム



- JASAのプレゼンス向上
- 月に2回(第一・第三金曜日)、6ヶ月
- すでに連載スタート
 - 4月3日『ラズパイでOpenELを動かしてみた』
 - 4月17日『モデリングは芸術、"アートな状態遷 移表設計"のすすめ』
 - 5月8日『システムの故障原因、第三者の立場で どう突き止めるか?』



組み込み業界では今、何が起きているのか





http://techon.nikkeibp.co.jp/article/COLUMN/20150401/412221/



掲載予定(仮題)



レガシーコードを生き返らせる(リバースエンジニアリングによるモデル化)

キャッツ 竹田 彰彦 ⇒あさって22日(金)掲載!

ETロボコン奮戦記(1) ジェイテック 星 光行

ETロボコン奮戦記(2) 富士ゼロックス 土樋 祐希

本当の品質は要求仕様から~半形式手法のすすめ~

ジェーエフピー 漆原 憲博

リアルタイムクラウド概論 日本マイクロソフト 太田 寛

リアルタイムクラウド詳細 ユビキタス 間中信一

ハードウェア設計と標準化~規制とライセンスの壁~

フラットーク 樫平 扶

組込みの人材育成~学生へのアプローチ~

グレープシステム 宮下 光明

こんなに使えるSysML レンタコーチ 中村 洋

ロボットの安全~26262の次に来るのは~ DTS/アートシステム長久保隆一

OpenELでらくらくロボット制御 アップウィンドテクノロジー 古俣 学

ハードウェアモデリング マルツエレック 掘込氏





拡散希望





中間報告 「もの作り技術者」の育成

2015年5月20日 ハードウェア委員会 WG1 矢部哲美



「もの作り技術者」の育成

ハードウェア委員会 WG1



1. テーマ選択の理由。

2013年11月のET2013組込み総合技術展において 「組込みシステム技術者のハードウェアスキルとは! **〜新しく要求されるスキル**」を当委員会の木下氏が発表した。 この発表の締め括りの一文より。

「組込みシステム分野は変革期を迎えています。新しい技術は 高度化していき、要求スキルも高まります。 興味を持って探究できる様にする、組織的に教育環境を 高めることなどに工夫していく必要がある。」



「もの作り技術者」の育成をテーマとしWG1において 育成方法を探っていくこととした。



2. これまでの活動。

П

П

П

2014年3月25日 K-Skill説明会。

2014年7月 8日 ルネサスエレクトロニクス「SmartAnalog」デモ。

2015年1月28日「組込みシステム業界の発展と技術者の育成

シンポジウム」。

2015年2月20日 企業訪問

HOTMOCK

O Min

株式会社ホロンクリエイト。(商品デザイン会社)

- ■ラピットプロトタイピングツール。 「HOTMOCK」のデモンストレーション。
- •エクスペリエンスビジョン概要解説。 「ユーザが望む新たな体験・経験を提供する手法」。
- ディスカッション。

テーマ:「プロダクトデザイナー」と「組込み技術者」の融合。 ~技術者に求められるものとは・・・感性価値の意識。



「もの作り技術者」の育成

ハードウェア委員会 WG1



- 3. 企業における人材育成の問題。
 - •ハードをゼロから組むことが非常に少ない。
 - ■学習用CPU基板を使うOFF-JOBトレーニングが多い。



- ツールが発達し、ユーザー目線、物作り目線で設計しない。
- 中間職のレベルアップをどうするか。
- 新技術に対し貪欲に取り組まない。受け身である。



4. 大学や高専等での取り組み。

学生版組込みスキル標準や、もの作りカリキュラムを充実。



組込みシステム業界の発展と 技術者の育成シンポジウム



かつ、どの様な教育が必要か、企業側の意向を探っている。



「もの作り技術者」の育成

ハードウェア委員会 WG1



5

5. これまでの技術者養成イメージ

これまでの、多くの技術者教育は、技術要素を広げ、技術レベルを上げる教育が主流です。



エンジニアリング領域

※組込みシステム技術協会

П

ここまでのポイント

- ☆ システムの高度化により要求される高度スキル。
- ☆ 技術者教育は企業内でも危惧している。
- ☆ ハードウェア的思考が弱い。
- ☆ これまでの教育は、技術要素の拡張と レベルアップが主流。

ハードウェアへの意識の低下が否めない。

いっぽう、他の業界や国の取り組みは?



「もの作り技術者」の育成

ハードウェア委員会 WG1

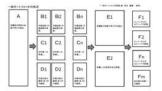


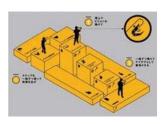
6. デザイン業界の流れ

様々な業界・業種、職能を交えての交流と ものづくりだけにこだわらないサービス開発 までも見据えた自由な発想の体験を行っている。



- 単純に教育と評価ではなく、感動体験発想と ポートフォリオで可視化している。
- ・エクスペリエンスビジョン これまでの問題解決型の手法から 「ユーザーが本質的に望む新たな 体験・経験を創出し、提供する手法」等 新しい手法を創りだし活用している。





7. 経産省などの動き

• 「感性価値創造イニシアティブ」創設し、活動への支援と事業と 環境の整備を行ってもの作りを応援。

日本の商品技術の繊細さは、世界的にも認知されている。
感性と感覚は、電気製品や工業製品にも不可欠。









デザイン思考なエンジニアリングとは 誰のためのどんな魅力的なものを作るか。 「知識力」+「知恵力」が大事。

技術はツールであり、何を作るかの発想力が大事。 すなわち、ユーザーの5W1Hの価値を生むためには?

では・・・

П

П

技術者の感性価値を高めるにはどうするか!



「もの作り技術者」の育成

ハードウェア委員会 WG1



8. これからの技術者に必要なもの

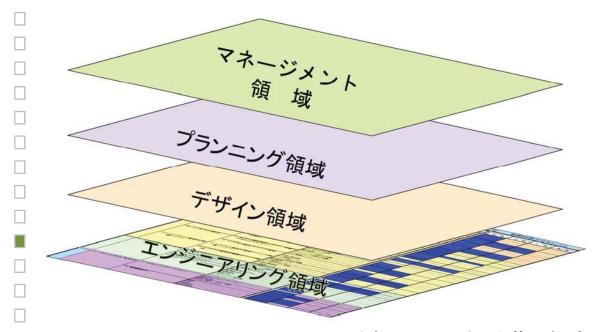
「ものづくり」・・・ 「技術者が感動的な物を **創**り、作り、造る」事と考え。



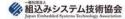
- ・上位プロセスへの意識拡張が必要。
- •単分野だけではなく、複数の分野知識が必要。
- ・これまでの問題解決型の手法だけでなく、ビジョン提案型の手法で 「新たな体験・経験」を通してユーザーに本質的価値を提供する。
- ・様々な業種・業界の人が集まり体験・経験を創出し、情報交換できる場が必要。
- ユーザー思考、社会環境思考、マネージメント思考が 出来る<u>開発技術視野が必要</u>。



9. 商品化までの領域ステップ



〈商品開発 領域階層概念図〉



© Japan Embedded Systems Technology Association 2015

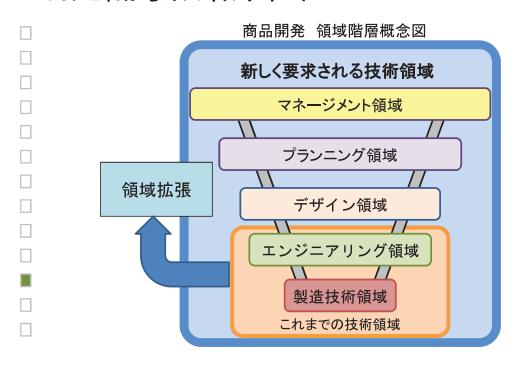
11

「もの作り技術者」の育成

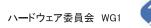
ハードウェア委員会 WG1



10. これからのスキルチャート



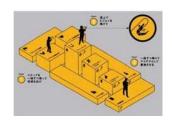
※ 組込みシステム技術協会

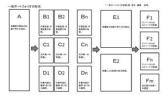


11. 今後の活動計画。

П

- ①、教育関係者やデザインハウスと意見交換を行い 「これからのスキルチャート」整える。
- ②、エクスペリエンスビジョンの講演を開催。
- ③、感性を高めるワークショップの開催形態を計画。 技能領域に縛られない形態等。
- ④、効果的なシラバスと評価方法を調査。 点数方式→ポートホリオ法等。
- ⑤、教材ツール・ギア等の調査。直感的に扱え、自由な発想ができる教材。
- ⑥、業種や会社を超えたエンジニアフォーラムの準備。 外部性を高める「エンジニアCafe」の設置へ。
- (7)、当協会の対象企業も拡張し、参加を促進する。







■ MAIN A System Technology Association 2015

© Japan Embedded Systems Technology Association 2015

13

「もの作り技術者」の育成

ハードウェア委員会 WG1



結びに

未だ道半ばのテーマでご座居ますが 皆様のご理解を賜り、今後の本活動への ご参加や、ご意見をお寄せ頂きますよう お願い申し上げます。

終わり。



参考ならびに引用文献。

/ なりひにつけれている。 ・ ハードウェア委員会木下氏より ET2013発表「組込み技術者に求められる新しいハードウェアスキルとは」。 ・ 株式会社ホロンクリエイト提供資料より フィジカル・ラピット・プロトタイピングHOTMOCK。

「ビジョン提案型デザイン手法」の概要。 又、高橋社長様には多大なご協力とご意見を頂戴いたしました、心よりお礼申し上げます。

・ 堀哲夫氏 編著より 「一枚ポートフォリオ評価」。

経済産業省資料より

が注条を目見付みり 監性価値適適とイニシアティブ kansei-honbun.pdf。 ものづくり白書(2004年)/第1部第2章第3節 今後のものづくりに求められる能力と人材育成の方向 g40601b123j.pdf。 ものづくり人材育成における大学(工学系)、高等専門学校、専門高校、専修学校の役割 honbun04_01_00.pdf。

- 富士ゼロックス株式会社HPより

「四次元ポケットPROJECT」 http://www.fujixerox.co.jp/company/ad/4d-project/。 富士通株式会社HPより

ヒューマンセントリック**エンジン**と サービス展開。

その他

K-Skill 学生向け組込み技術教育システム説明資料。

組込みシステム業界の発展と技術者の育成シンポジウム資料。

「もの作り技術者の育成」

2015/5/20 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8-12

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。

組込みシステム技術協会

© Japan Embedded Systems Technology Association 2015



中間報告ロボットや制御機器の技術調査

2015年 5月20日 ハードウェア委員会 WG2 笠木大幹



© Japan Embedded Systems Technology Association 2015

1

ロボットや制御機器の技術調査



目次

- 1. 目的
- 2. これまでの活動
- 3. 現状の考察
- 4. 提案機器
 - 4-1. セパレートタイプによるデバイスドライバ以下の交換
 - 4-2. FPGAを使用した全ハードロジックによる実現
 - 4-3. 新プロセッサ採用による差別化
 - 4-4. OpenEL仕様の汎用ボードの開発
- 5. 今後の展開





1. 目的

ハードウェア標準プラットフォームのプロトタイプ設計するために、ロボットやあらゆる制御機器に求められる技術要素を調査する

自ら設計することでみえる新たな課題をOpenEL (プラットフォーム研究会)にフィードバックする

既存の機器との差別化、どのような特徴を出せるかを考察する



© Japan Embedded Systems Technology Association 2015

3

ロボットや制御機器の技術調査



- 2. これまでの活動
 - 2014年 7月 8日 ルネサスエレクトロニクス"SmartAnalog"デモと意見交換会
 - 2015年 1月27日 プラットホーム研究会"OpenEL"デモと意見交換会
 - 2015年 2月19日 プラットホーム研究会会議参加
 - 2015年 3月10日 ルネサスエレクトロニクス"RZ/T1"デモと意見交換会





3. 現状の考察

既に、様々な機器に沢山の種類のアクチュエータやセンサーが組み込まれています



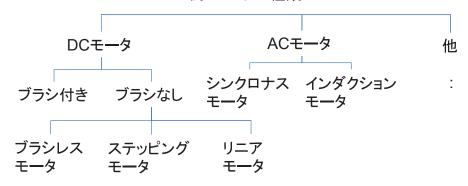
© Japan Embedded Systems Technology Association 2015

ロボットや制御機器の技術調査



3. 現状の考察

例:モータの種類



- ・比較的高頻度に起動や停止、正転・逆転が必要か
- ・低速での制御やトルクが必要か、高速運転が必要か
- ・位置決めの精度はどこまで必要か
- ・直線運動か回転運動か

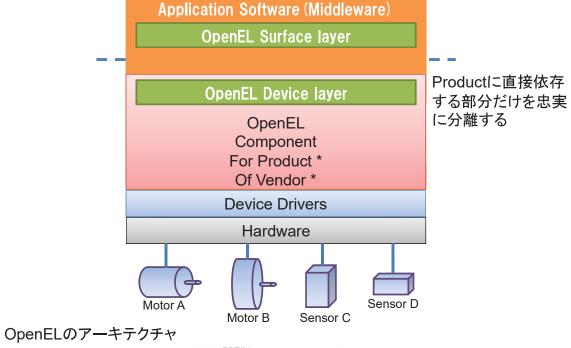
どの種類のモータから どのように特徴を出していくか



© Japan Embedded Systems Technology Association 2015



4-1. セパレートタイプによるデバイスドライバ以下の交換





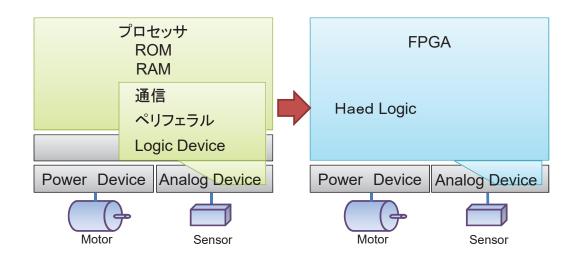
© Japan Embedded Systems Technology Association 2015

7

ロボットや制御機器の技術調査



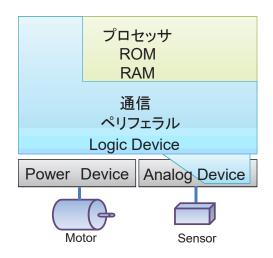
4-2. FPGAを使用した全ハードロジックによる実現



消費電力 1/10~1/100 動作クロック 1/10~1/100



4-3. 新プロセッサ採用による差別化(1)



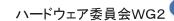
OpenEL仕様の独自プロセッサも可能



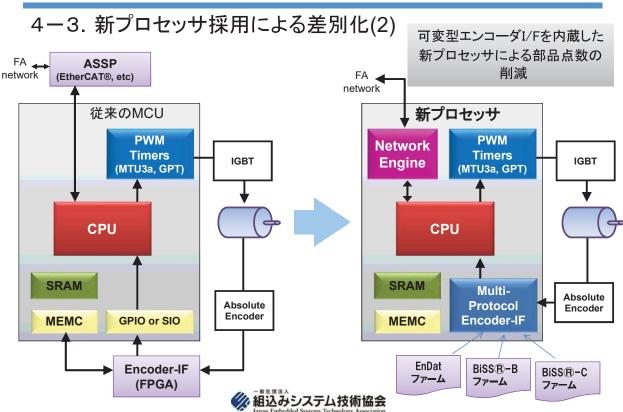
© Japan Embedded Systems Technology Association 2015

9

ロボットや制御機器の技術調査



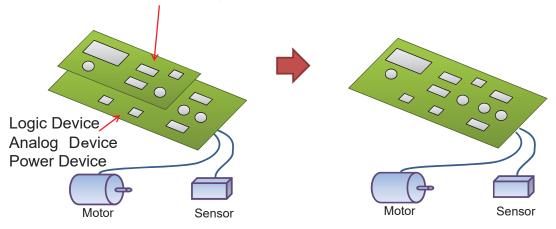






4-4. OpenEL仕様の汎用ボードの開発

Raspberry Pi、BeagleBoard、Arduino、mbed、他汎用基板



OpenEL仕様の独自汎用基板



© Japan Embedded Systems Technology Association 2015

11

ロボットや制御機器の技術調査



5. 今後の展開



アームロボットへの適応検証



教育用機材の企画・開発



ドローンへの適応検証



デバイスのパラメータ化・ データベース作成





「ロボットや制御機器の技術調査」

2015/ 5/20 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8-12 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。



13



アジャイル研究会の挑戦 ~組み込みとアジャイルと、契約と~

2015年5月20日 JASA中部 アジャイル研究会 (株)ヴィッツ 戸澤充



agenda.



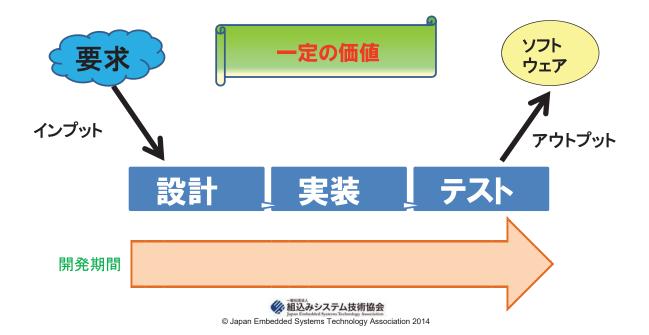
- 1. アジャイル開発について
- 2. アジャイル研究会とは
- 3. アジャイル研究会の昨年度までの活動
 - ①:「アジャイル」を知る
 - ②: E社の事例(Scrumの導入)
- 4. アジャイル研究会の挑戦 Part1
 - ①:ガイドラインの策定
 - ②:アジャイルな契約
- 5. アジャイル研究会の挑戦 Part2
 - ①:内面の壁に対する挑戦
 - ②:外面の壁に対する挑戦
- 6. まとめ



アジャイル開発について



■ ウォータフォール開発

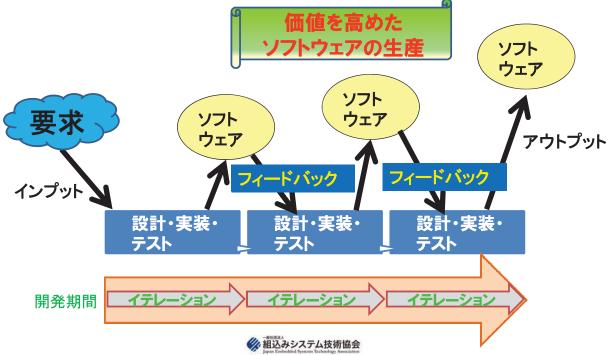


アジャイル開発について



3

■ アジャイル開発



© Japan Embedded Systems Technology Association 2014

92

.

アジャイル研究会とは



■ 研究会発足背景

• アジャイル開発を組み込みソフト開発に適用すること業界が抱える課題を解決することはできないか?

■ 研究会目的

アジャイルソフトウェア開発手法の導入ガイドラインの策定、 または、開発プロセスデザインのガイドラインの策定

■ 活動実績概要

- 2012年4月アジャイル研究会発足
- アジャイルソフトウェア開発手法の調査
- 実践と評価
- 2013年度JASA成果発表会「アジャイル適用の課題と、解決に むけての考察・実践」
- ET2013「組込アジャイル導入への序章 ~明確化と合意~」



アジャイル研究会の昨年度までの活動



① 「アジャイル」を知る

■ 目的

アジャイルの素人集団が、とにもかくにもアジャイル開発の実践にこぎ着けるために、最低限の知識を身につける。

- プラクティス(=手法)の調査と実践 アジャイル開発にある代表的なプラクティスをピックアップし、 3グループにわかれて、それぞれのテーマに沿って調査を実施。
 - テスト駆動開発実践グループ
 - ストーリー作成・タスク分析グループ
 - プロセス実践グループ

その後、代表した会社に実際プラクティスを実践してみた。

■ 実績

業務での実践事例

• E社:Scrumの実践



アジャイル研究会の昨年度までの活動



③ E社の事例 (Scrumの導入)

■ 目的

Scrumの実践を通じて、下記の改善を図る。

- 手戻りコストの削減
- 生産性の向上
- 属人性の排除
- 実践したこと

下記のプロジェクトに対して、Scrumを適用。

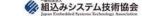
開発対象: センサーデータを用いたクラウドシステム

➤ Webアプリケーション開発

▶ DB開発

▶ 組み込み機器(通信機器)開発

開発規模: 7名開発期間: 3ヶ月



© Japan Embedded Systems Technology Association 2014

アジャイル研究会の昨年度までの活動

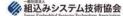


③ E社の事例 (Scrumの導入)

手法	課題	対策
定期リリース	要件や機能によって <mark>顧客の</mark> 希望納期が異なり、リリー ス間隔の固定が難しい	スプリントは可変とし、対象スプ リントで対応する要件・機能のボ リュームによって設定する
メンバーの固定化	組織・業務の特性上、メンバー入れ替えが発生し、固定化するのが難しい	1ストーリーに対して二人の担当 者を割り当て、ペアワークという 形で情報共有の円滑化を行う
自己組織化	開発担当者全員が自発的に 行動する必要があり、全員 の意識を変えるには時間が かかる	作業割り当てはスクラムマスター が行う

■ 実践結果

- 手戻りは大幅に減少。
- 生産性は低下(テストを自動化できなかったことによる)
- 属人性の排除を実現(ペアワークの導入効果)



© Japan Embedded Systems Technology Association 2014

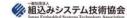
アジャイル研究会の昨年度までの活動



まとめ

- ✓ 机上でシミュレーションするだけでは、課題解決につながらない (つながるイメージがもてない)。
- ✓ 実践してみると、効果はすくなからずある。
 - Scrum導入による出戻りコストの削減
 - ペアリングによる属人性の排除
- ✓ とはいえ、「実践」のためには、いくつも壁が存在する。
 - ステークホルダー(顧客、上司、開発スタッフ)を納得させるだけの効果を事前に提示できない。
 - すぐにわかりやすい効果がでるとは限らない。逆効果の場合も。
- ✓ 「組み込み」+「アジャイル」の実践には、至っていない。納期の 問題、ハードとの絡みの問題、環境の問題…

「とにかく一回やってみる(そして成果をだす)」のは、思いのほか難しい…



© Japan Embedded Systems Technology Association 2014

アジャイル研究会の挑戦 Part1



①ガイドラインの策定

■ 目的

「とにかく(アジャイルを)一回やってみる」をやってみたノウ ハウを共有するために、いったんガイドラインとしてまとめる。

■ 実践したこと

IPAの「アジャイル型開発におけるプラクティス活用事例調査 調査報告書 ガイド編

(http://www.ipa.go.jp/files/000026849.pdf)」をベースとして、「組み込みソフトウェア開発向けアジャイルプロセスガイドライン」の作成に着手した。

- 上述のガイドラインの理解
- 組み込み開発ではそのまま適用できないプラクティスを抜粋
 - ✓ 「できる(やっている)」「できるが、やっていない」 「できない」に分類
 - ✓ どうすればすべてを「できる」にもってこられるかを、議 論。





①ガイドラインの策定

■ 実績

• 各社のプラクティス実施是非に関する見解まとめ。

カテゴリ	No	プラクティス	①できる・やっている		②できるけど、やっていない			③できない			
			Witz戸澤	• • •	合計	Witz戸澤		合計	Witz戸澤		合計
プロセ スト・	1	リリース計画ミーティング			4			0	0		3
	2	イテレーション計画ミーティング	0	0	4			1			2
	3	イテレーション		0	3			1	0		3
	4	プランニングポーカー			0	0	0	5			2
	14	:	0	0	3		0	3			1
	15	インセプションデッキ			0	0	0	4			3
	16	プロダクトバックログ			1	0	0	4			2
	17	迅速なフィードバック	0		4		0	1			2
技 術 ・ツ ル	18	ペアプログラミング			0	0	0	5			2
	19	自動化された回帰テスト			1			2	0		4
	20	テスト駆動開発			1		0	3	0		3
	21	ユニットテストの自動化			2	0		3			2
	22	受入テスト		0	6			0	0		1
	23	:	0		3			2			2
	26	シンプルデザイン			2			3	0		2
	27	逐次の統合			1			2	0		4
	28	継続的インテグレーション			3			0	0		4
	29	集団によるオーナーシップ			4	0		3			0

組込みシステム技術協会

© Japan Embedded Systems Technology Association 2014

アジャイル研究会の挑戦 Part1



11

①ガイドラインの策定

各社の見解を集計「できない」数→全体の30%



「できない」数→全体の**1%弱**

「できる」になった理由

- ・実施していないから「できない」を選択していた
- ・適用「できる」方法が分かった
- ・開発体制の問題

etc...

組込みシステム技術協会

© Japan Embedded Systems Technology Association 2014

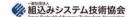


①ガイドラインの策定

■ 問題のある「技術・ツール」

プラクティス名	内容	問題の原因
テスト駆動開発	テストコードとプログラムを平 行開発。最低限の機能から始め て、少しずつ機能拡張する。	ソフトウェアは少しずつ作れて も、ハードウェアは完成しない とでてこない。ので、その時点 ではテストができない。
ユニットテストの 自動化	コスト (リソース) 削減のため、 既存のテストを自動実行できる ようにする。	環境(特にハードウェア)が揃わないとテストできない領域がある。コスト面で自動化は見合わない。
逐次の統合	複数の修正を一度に結合しない。 一結合一機能で動作確認する。	「ユニットテストの自動化」が 前提である。
継続的インテグレ ー ション	変更のある/なしに関わらず、定期的に結合・確認を実施する。	「ユニットテストの自動化」が 前提である。

「自動化」と「環境」の相性が課題。



© Japan Embedded Systems Technology Association 2014

13

アジャイル研究会の挑戦 Part1



②アジャイルな契約

- 「できる」のに「やらない」の要因?
 - ✓ 「アジャイル」に対する認識違い(誤解)
 - ✓ 既存プロセスを変更することに対する抵抗感
 - ✓ 組み込み特有の事情(ソフト/ハード双方の環境がそろうタイミングが遅れがち)により、テスト駆動開発が困難
 - ✓ 教育が浸透しない
- 顧客(発注元)の意向は無視できない。
 - ✓ 顧客にとってのメリットは?
 - ✓ 今までのやり方をかえるリスクはないのか 対策は?
 - ✓ 成果物はどう変わる?いままでどおりに納入できるの?
 - ✓ 何に対して、どれだけ報酬を払えばよいの?
- ⇒「ステークホルダー間の意識ズレ」に関わるもの。 最後まで残った「ズレ」の責任は、誰が取るの?







こっちは100% 確実にやりますよ。



箱におさまる量なら、要件が確定次第やりますよ。 準備もしておきます。



© Japan Embedded Systems Technology Association 2014

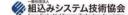
15

アジャイル研究会の挑戦 Part1



- まとめ
 - ✓ アジャイル導入への「壁」は、二つある。
 - ① 「できるのにやらない」内面の壁。
 - ② アジャイルに費用を払ってもらう(価値の共有)ために、 顧客の説得が必要。外面の壁。
 - ✓ 「内面の壁」克服のためには、以下が必要。
 - 「できること」の実績を増やす。
 - 「自動化と環境」の課題を克服する。
 - ✓ 「外面の壁」克服のためには、事前に責任の所在(「誰が」 「何に」「どれだけ」)が明確になるような、新しい仕組み が必要。

対策は、2本立てで。 『できるようになる』と『理解してもらう』。





①内面の壁に対する挑戦

■ 「できること」を増やす

参考本「アジャイルサムライ -達人開発者への道」(※)に対する輪講

- ⇒研究会前までに研究会メンバーで同じ章を読解
- ⇒研究会の冒頭15分間で、メンバー全員の意見や知識交換を 実施

※オーム社 出版 Jonathan Rasmusson 著 西村直人・角谷信太郎 監訳/近藤修平・角掛 拓未 訳



17

アジャイル研究会の挑戦 Part2



①内面の壁に対する挑戦

- 「自動化と環境」の実践 組み込みソフトウェア開発にテスト駆動開発を実践している方を 研究会に招いて、下記を講演
 - ▶ テスト駆動開発のワークを実施
 - ・追加する機能が、追加前では失敗することを 確認することが重要!!
 - ・追加する機能を実現する為、必要最低限のソースコードで 作成することが重要!!
 - ユニットテストの自動化に使用するツールの紹介 ⇒今後、研究会内でも実践していきたい



②外面の壁に対する挑戦

- ■W社のアジャイル開発導入の挑戦
- > STEP 1

取引先であるO社へアジャイル開発導入の

- 取引先であるO社にアジャイル開発の紹介
- W社が抱える現状の問題を説明
- 問題を改善する可能性のあるプラクティスを紹介



- 期間を区切ることを目的にすると失敗する可能性があるのでは?
- 事例から当てはめるのは難しい。もっと具体的なやり方をしりたいetc...



19

アジャイル研究会の挑戦 Part2



②外面の壁に対する挑戦

- ■W社のアジャイル開発導入の挑戦
- ➤ STEP 2 (現在) STEP 1 でO社から指摘されたことを研究会にて展開し、情報を 共有。
- STEP 3 O社の懸念を取り除くべく、更なる説明したい

懸念していること解決できるよう、さらなる議論。

➤ STEP 4 O社との合意の下、W社でのアジャイル開発を実践へつなげたい



- まとめ
 - ✓ 組み込みソフトウェアでもテスト駆動開発、ユニットテストの自動化は「できる」。 テスト駆動開発、自動テストツールの実践により、 「内面の壁」克服のための足かけを築く
 - ✓ W社-O社間でのアジャイル開発合意までの道のりをモデルに、 「外面の壁」に対する克服の一歩としていく



プラクティスを「できるようにする」ための実践方法や「理解してもらう」ための事例をガイドラインへ反映していき、 組み込みソフトウェア開発業界でも、すぐにアジャイル開発 を適用できる仕組みを作っていく。



© Japan Embedded Systems Technology Association 2014

21

まとめ



- ✓ アジャイル開発にて使用されるプラクティスは、組み込みソフトウェア開発でも活用『できる』ことが分かった。
- ✓ 但し、顧客に対して、『理解をしてもらう』 には至っていない。

もっともっと実践や事例を作り、組み込みソフトウェア開発者向けのガイドラインを策定し、組み込みソフトウェア開発業界全体にアジャイル開発を普及させていきたい



最後に





組込みシステム技術協会

© Japan Embedded Systems Technology Association 2014

23



「アジャイル研究会の挑戦〜組み込みとアジャイルと、契約と〜」

2015/5/20 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8-12

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。

