

問題解決手法としてのXDDP 適用事例のご紹介 (2013-2023年)

2024/05/24 (金)

10:10-10:50 (発表30分、質疑10分)

派生開発推進協議会/株式会社 構造計画研究所
本田英稔

目次

1. はじめに
2. 構造計画研究所と自己紹介
3. テストプロセス策定と駆動（PFD）
4. 実証実験プロジェクトの救済（変更設計書）
5. スマートロックシステムの派生開発の実践
（USDMM/TM）
6. 今後について
7. おわりに

はじめに

清水吉男氏の提唱したXDDP (eXtreme Derivative Development Process) は派生開発の QCD (Quality, Cost and Delivery) を向上させる有効なプロセスであることが認識されるようになってきた。しかしながらXDDPの根底には問題解決プロセスがある。ステークホルダー間でプロセス適用範囲、対象をspecifyしつつ、手戻りを最小限にし最短で解決を導出可能である。

- テストプロセス策定と駆動 (PFD)
- 実証実験プロジェクトの救済 (変更設計書)
- スマートロックシステム(RemoteLOCK)派生開発の実践 (USDMM/TM)

いずれも プロジェクト途中から参画し、XDDPの部分適用でも有効性はあるの言葉(仮説)から、三点セットとPFDを用いて問題解決プロセスを提案して実践したものである。記載内容の抽象度を上げることによる 問題解決手法としてのXDDP適用事例となる。いずれもビジネス目標を達成するために発生したもので、資料も実際のもものを例示した。

はじめに

参考としてXDDPの成果物と技法について定義と読み替え内容を示す。
(『[困ってませんか？ 派生開発 ～XDDP はじめの一步～](#)』から引用・変更)

#	項目	内容
1	変更要求仕様書 (USDM)	要求仕様の表記法。要求と仕様を階層的に表現する。 また、仕様の抜けや漏れを防ぐことを目的として、要求の背景としての「理由」(why/reason)を記述する。
2	トレーサビリティマトリクス(TM)	USDMとシステム(変更設計書)とのトレーサビリティを確保するためのマトリクス形式文書。
3	変更設計書	派生開発においてシステムの 変更箇所 を記述した文書。原則として、TMのマトリクス交点にマークした箇所に対応して作成する。
4	プロセスフローダイアグラム(PFD)	プロセスと成果物の連鎖を記述する表記法。「プロセス」と「成果物」の連鎖を「フロー」でつないで表現する。

構造計画研究所(略称：KKE)について

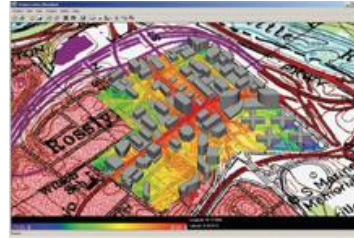
大学、研究機関と実業界をブリッジするデザイン&エンジニアリング企業として、社会のあらゆる問題を解決し「次世代の社会構築・制度設計」の促進に貢献する。創業1956年で「工学知」を重視している。2020年6月時点で日本国籍以外の所員が約8%を占める。

建設・防災分野



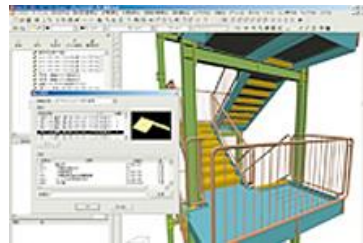
- 熊本城の修復(1960年代)
- 超高層建築・大規模特殊構造物
- 防災コンサルティング

情報通信分野



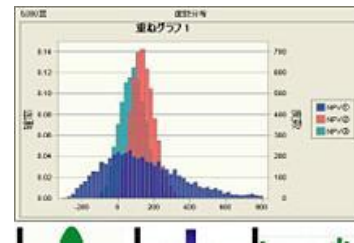
- ソフトウェアの受託開発
- 電波伝搬シミュレーション

製造分野

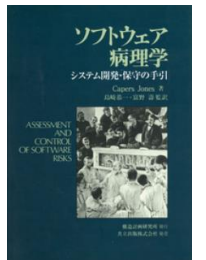


- 生産・物流シミュレーション
- 仕様設計支援

意思決定支援分野



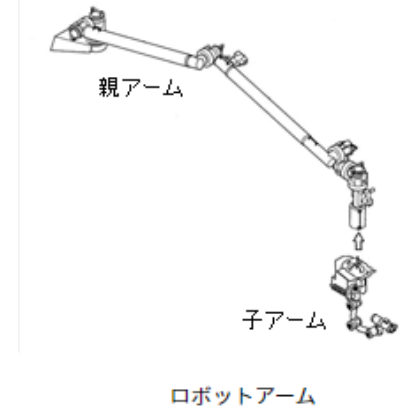
- リスク分析シミュレーション
- マルチエージェントシミュレーション



自己紹介：本田 英稔（ヒデトシ）

経歴：実験・運用システム開発から運用、品質保証・カスタマーサクセスの越境人生...

- 1990年代
 - 武蔵野通信研究所におけるISDN活用のための音楽配信実証実験
 - ISS「きぼう」ロボットアームOS部分の製造・試験
 - 第2世代携帯電話の監視制御システムの開発～運用
- 2000年代
 - 第3世代携帯電話の監視制御システムの開発～運用
 - センサーネットワークの実証実験(Bluetooth/WIFI/Zigbee/Z-wave)
 - 衛星携帯電話の監視制御システムの開発～運用
- 2010年代
 - 清水吉男さんに出会う(2012/11/27,28於日科技連セミナー)
 - **2013:海運大手の自動車専用船(PCTC)積付けシステムの品質保証(PFD)**
 - **2014:公的研究機関の次世代WiFiシステム試作の救済(変更設計書)**
 - 準天頂衛星システムの品質保証体制構築、Autowareを使った自動運転実証実験
- 2020年代
 - RemoteLOCKチームのCS担当兼2023:**ハードウェア派生開発企画(USDMM/TM)**



テストプロセス策定と駆動(PFD)

対象

- 日本大手海運会社の自動車専用船(PCTC)積付け支援システム
- 船型データ図面上に自動で輸送物を最適配置し運航データを作成する。多岐に渡る業務知識と実装技術を必要とする複雑なシステム

問題

- **品質保証を踏まえたテストプロセスを立上げ**できるメンバがいなかった
補足：ネットワーク化含めた最新の開発環境での全面作り直しというある程度の規模の受託開発であり、品質保証の哲学から語る事が難しかった

解決

- テスト計画策定から、該当システムのテストプロセスを**PFD用いてテラリング**しテスト完了まで伴走した。
品質保証の考え方をチームメンバに教育するための簡潔かつ分かりやすい表現技法としてPFDを用いた。

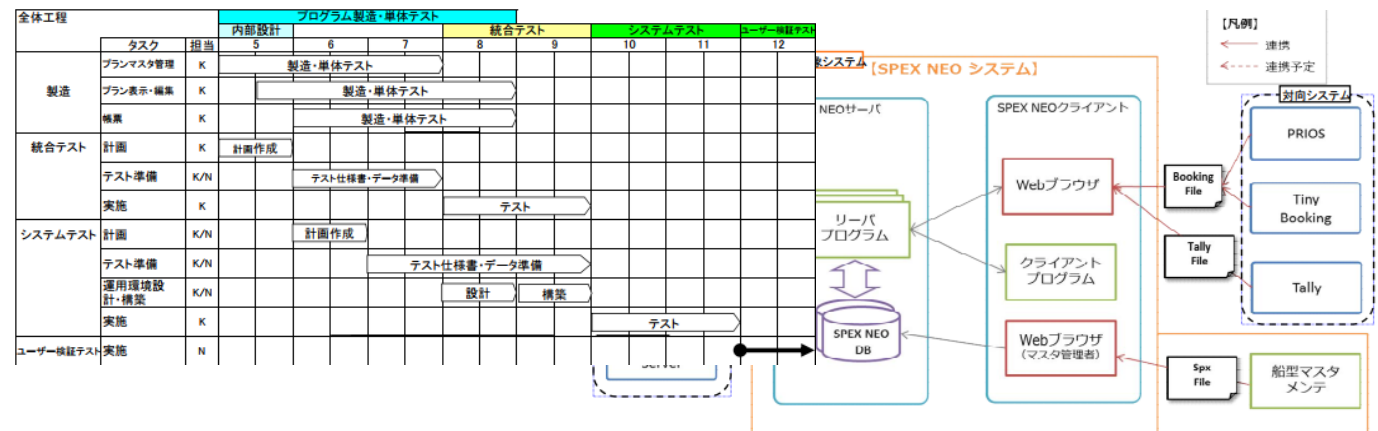
テストプロセス策定と駆動(PFD)

経緯

- 開発：2012年12月25日-2014年2月15日
- 支援：
 - 2013年4-6月 テストプロセス策定
→システムテスト、統合、単体と工程を定義し、
重視すべきは統合テスト工程であり、テストに関わるレビュープロセスも顧客と合意、**テスト計画書にPFDの考え方を適用した。**
 - 2013年8-12月統合テスト実施、工程完了報告を完了

機能テスト	テストの観点	単体テスト	統合テスト	システムテスト
サブシステム内テスト	サブシステム内の機能を単機能、単機能結合といった順序で確認する	○	—	—
サブシステム間テスト	データの流れに沿っての機能をテストケースに従って確認	—	○	—
システム連携テスト	対向システムとの連携を、テストケースに従って確認する	—	—	—
業務適合テスト				
プラットフォームテスト				
OS/ブラウザテスト	OS、ブラウザの違いによる問題が発生していないことを確認する	—	○	—
プリンタテスト	指定された機種によって、目的の出力結果が得られていることを確認する	—	○	—
その他のテスト				
性能テスト	画面の応答性、業務の迅速性、バッチの処理時間を測定・検証する	—	レスポンス測定	—
負荷テスト	想定された負荷が発生した場合でも、問題が無いことをテストする。	—	ラッシュテスト※1	—
脆弱性テスト	システム運用において、脆弱が存在しないかを検証する	—	サーバ異常時の復旧テスト	—

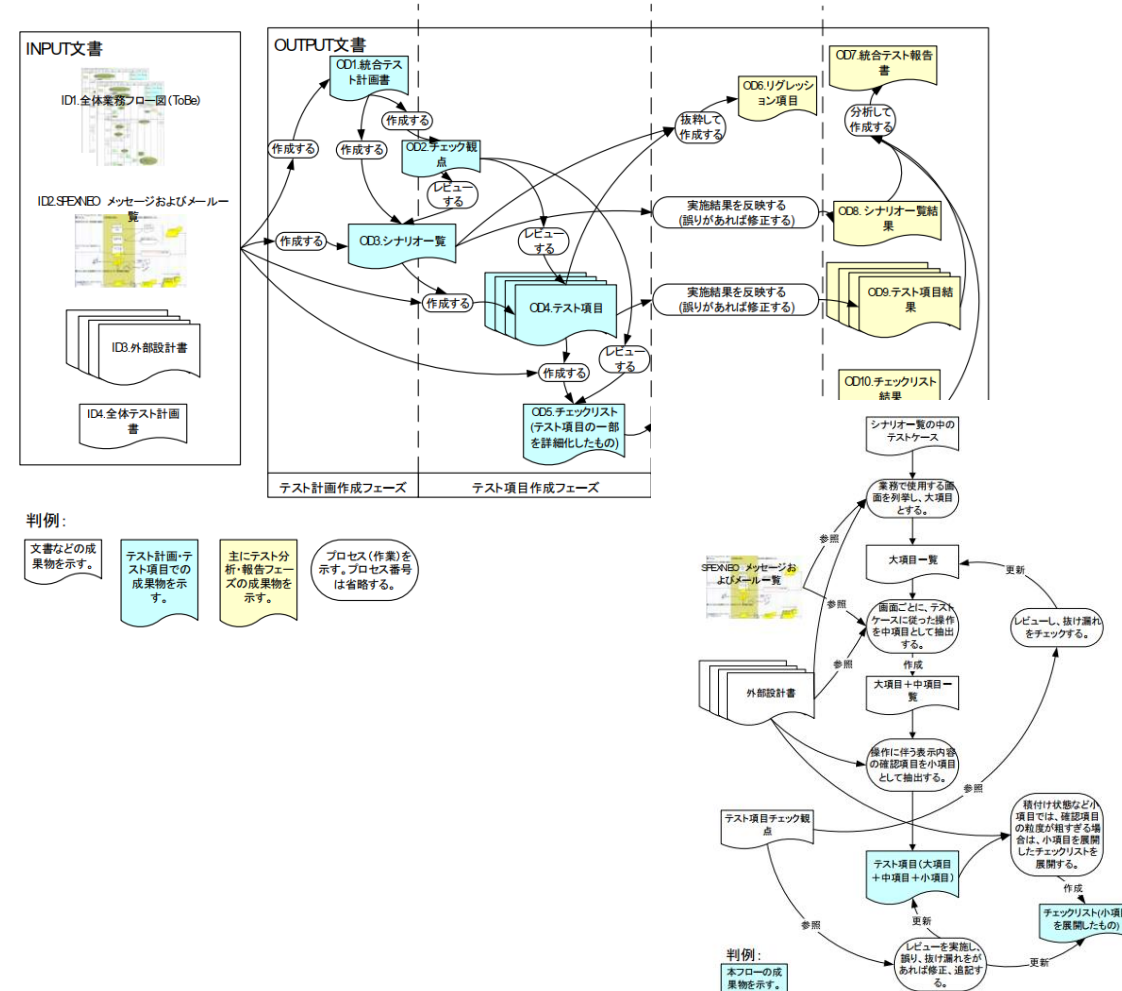
※1 複数のクライアントからの同時アクセスにより不適合が発生しないかの確認。



テストプロセス策定と駆動(PFD)

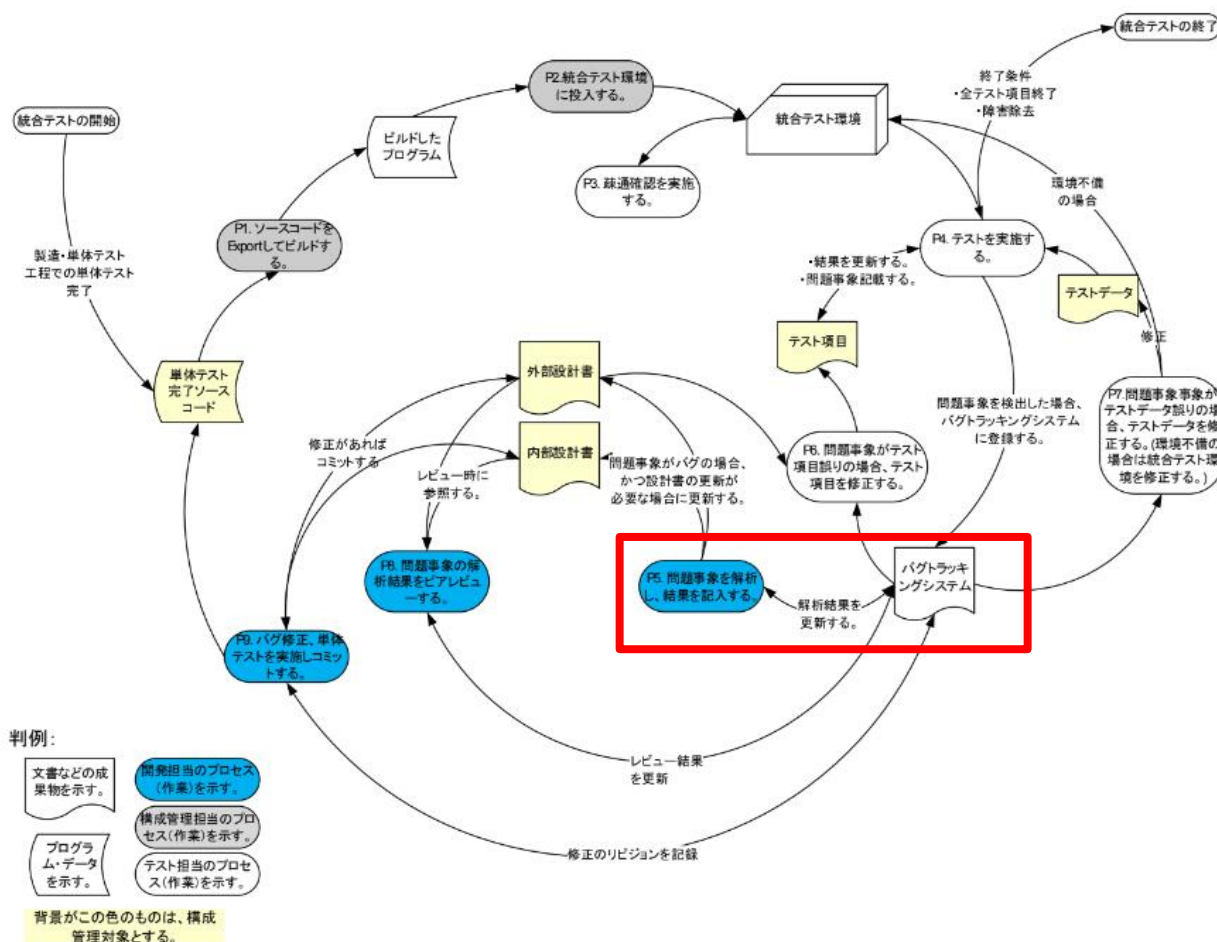
統合テスト計画の目次と文書構成図、INPUT文書、OUTPUT文書を定義し、誰が、どのレビュー観点でレビューを実施するかを明示した。
補足：右図はPFDではないとのこと(清水氏)

1	はじめに.....	3
1.1	統合テストの文書構成.....	3
1.1.1	INPUT 文書.....	3
1.1.2	OUTPUT 文書.....	4
1.2	用語.....	5
2	概要.....	7
3	シナリオ一覧・テスト項目作成詳細.....	9
3.1	シナリオ一覧・テスト項目の考え方.....	9
3.2	シナリオ一覧抽出手順.....	10
3.3	テスト項目抽出手順.....	11
3.3.1	機能要件テスト項目抽出手順.....	11
3.3.2	非機能要件テスト項目抽出手順.....	12
3.4	レビュー観点.....	13
4	テスト実施詳細.....	14
4.1	テストの流れ.....	14
4.1.1	テスト開始条件.....	16
4.1.2	テスト終了条件.....	16
4.1.3	テスト管理方法.....	16
4.1.4	問題事象解析について.....	16
4.1.5	問題事象解析結果の記載について.....	17
4.2	テスト環境.....	18
5	評価方針.....	19



テストプロセス策定と駆動(PFD)

RedmineとSVNを用い全変更プロセスをすべて記録。朝ミーティング、夕ミーティングを実施。プロセス・プロダクティブの妥当性を、テストプロセス策定と駆動を通して示すことができた。本統合試験内**バグ分析をトリガに大幅な機能改善や強化試験を行える仕組み**を盛り込んだ。**メンバにも本資料を元に説明会を複数回実施して意識付けを行った。**



#	プロセス	内容	備考
P1	ビルドする。	D]をビルドサーバにExportしてビルドする。	
P2	統合テスト環境に投入する。	ビルドしたプログラムを統合テスト環境に投入する。	
P3	疎通確認を実施する。	テスト担当により、疎通確認を実施する。 ・障害の場合は修正内容。 ・シナリオをベースにした、リグレッション項目	リグレッション項目はT.B.D(期限 2013/9/30)とする。
P4	テストを実施する。	・テスト項目を元にテストを実施する。テスト項目に合格した場合は、OKを記載する。エビデンスとして画面キャプチャを残す。 ・問題が発生した場合は、バグトラッキングシステムに記載し、開発担当に「問題事象解析依頼」を行う。テスト項目にもNGを記載する。4.1.4を参照のこと。 ・障害が改修されたテスト項目の確認(P9->P1->P2)の場合は、改修確認にOKを記載する。バグトラッキングシステムの該当チケットをクローズする。 ・テスト項目誤り(P6)から選択した場合は、再テストを実施し、問題なければテスト項目をOKにする。 ・テストデータ誤り(P7)から選択した場合は、テスト項目誤りと同様とする。「環境不備」などの場合も同様とする。	
P5	問題事象を解析し、結果を記入する。	「問題事象解析依頼」を受け取った開発担当は、障害を解析し、バグトラッキングシステムに記載する。記載については、4.1.5を参照のこと。 ・テスト項目誤りの場合はP6に進む。 ・テストデータ誤りの場合はP7に進む。 ・障害の場合は、その旨をバグトラッキングシステムに記載する。P8に進む。	
P6	問題事象がテスト項目誤りの場合、テスト項目を修正する。	バグトラッキングシステムに記載された内容を元に、テスト項目を修正する。その後P4に進む。	テスト項目の修正の妥当性は開発担当が行う。
P7	問題事象がテストデータ誤りの場合、テストデータを修正する。	バグトラッキングシステムに記載された内容を元に、テストデータを修正する。その後P4に進む。	
P8	問題事象の解析結果をビジュアル化する。	障害の場合に改修前に問題事象の解析結果をビジュアル化する。	
P9	障害修正、単体テストを実施しコミットする。	・障害を修正する。 ・単体テストを実施する。単体テストOKでSVNにコミットする。 ・修正バージョン(SVN)および単体テスト結果をバグトラッキングシステムに記載する。	

テストプロセス策定と駆動(PFD)

問題事象解析後のバグトラッキングシステムへの記載内容。
テスト消化完了後に分析。分析内容に応じて強化テストを実施した。

- 原因：問題事象の原因を記載する。
- 対処：ソースの修正内容を記載、**分析の結果として設計書修正の場合はその内容を記載。**
- 分析：混入工程：どの工程で混入したのか記載する。
- 分析：混入原因：混入工程での原因を記載する。
- 分析：未検出原因：単体テストで検出できなかった理由を記載する。
- **水平展開：混入原因と同じ原因が混入していないかの観点でソースコードを調査した結果を記載する。調査結果で問題なければ、問題なかった旨を記載する。問題があった場合は、同時対処内容について記載する。**
- 確認内容：対処内容の確認方法を記載する。水平展開で対処した場合は、その方法も記載する

テストプロセス策定と駆動(PFD)

その後のエピソード

- この時のお客様側リーダーは2017年4月創業132年目で初の女性船長に就任
- この時の実績もあり、ONE（NYK、商船三井、川崎汽船のコンテナ事業統合）によるAOCTEL（次世代海上コンテナ輸送研究所）を熊本構造計画研究所内に設立 <https://www.kke.co.jp/release/9816>

実証実験プロジェクトの救済（変更設計書）

対象

- 公的研究機関の次世代WiFiシステム試作。既に存在するネットワークプロトタイプを、広域化・マルチユーザ化し新世代ネットワークにおける種々の研究課題の評価のために用いるため大幅な拡張を行った

問題

- ハードウェア含めたシステム全体の基本設計・機能設計を行い開発を行ったが「発注仕様書」の要求が実現できるか分からないというところから混乱。即ち顧客による受入が出来ない状況に陥った。

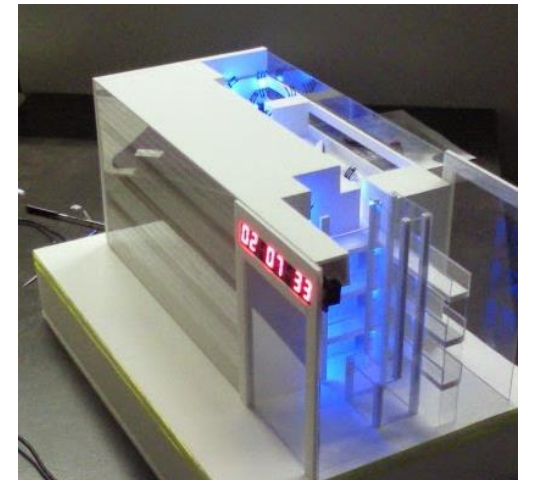
解決

- 「発注仕様書」を「動作原理ドキュメント」と「変更設計書」に分解
- 「変更設計書」で何をどのように実現すれば良いか顧客とレビューし（設計書・ソース・テスト項目も明記）実行結果のエビデンス、顧客動作確認まで実施。発注者側視点の確認内容をレビューできるためである。

実証実験プロジェクトの救済（変更設計書）

経緯

- 開発期間：2013年8月-2014年6月末
- 支援期間：2014年1月-7月4日に顧客最終確認まで対応
- 上記をベースに納品までのプロセスとスケジュールを明示、確認ポイントの可視化、信頼を回復しつつ対応
- その他、支援要員として参画したが越境してサブPM相当を遂行
公的研究機関の研究者として現場常駐、Interop2014展示で説明



実証実験プロジェクトの救済（変更設計書）

変更設計書(詳細票と呼称)の例。下記は設計変更を伴わないタイプの資料で最小限の記載。最大のものでは**30ページを超えるもの**もあった。

詳細票

プロジェクト	仮想化対応無線NWプロジェクト	作成日	2014年5月26日
発注仕様書#	カテゴリ: パラメータ	作成者	本田
	YNOS ないしはそれに準じる機能からの制限、命令に基づき、稼働中の veBS の数及びこれらに対する veBS の割り当て状況、帯域割り当て状況、実データ転送状況、送受信可能バンドにおける周波数チャネル利用状況を計測し、応答パケットにより通知できること。また OFS 受信回路を格納し、基地局の設置位置に関する情報を計測し、応答パケットにより通知できること。	確認日(予定)	5月27日
	YNOS ないしはそれに準じる機能からの制限、要求に基づき、稼働中の veBS の数、これらに対する割り当て帯域の状況及び実データ転送状況に関するパラメータについて、計測し通知できること。	確認者	
	ネットワーク上で、veBS 及び veBS-SW が保有する各種パラメータの状況計測を、定期的ないしは任意でログを生成し、応答として得たパラメータをタイムにグラフ等で表示できること。		

■概要
本詳細票では、パラメータに関する仕様を、図1 本詳細票の確認内容の仕様(概要)、処理概要等を示す。最後に確認内容を示す。

表-1 各機能に関する処理概要

項目#	内容	確認内容
1	veBS から以下の結果を MainCPU で実行 ・ WiFi CPU で実行 ・ OFS (veBS) のフローエントリ	viewconfig, vivivlist, vivpriv, wlanconfig, を実行し veBS の情報が取得できることを確認する
2	veBS-SW から以下の結果を MainCPU で実行 ・ フローエントリ	「画面」でご確認ください。 「画面」でご確認ください。 「画面」でご確認ください。
3	veBS および veBS-SW から以下の結果を MainCPU で実行 ・ MySQL に保存する。	「画面」でご確認ください。 「画面」でご確認ください。
4	■詳細機能設計書に関する変更(既存記載箇所) コマンド viewconfig, vivivlist, wlanconfig, athstats, 80211stats を実行して結果の取得ができる。	「画面」でご確認ください。 「画面」でご確認ください。
5	(追記・修正すべき内容) なし。	
6	■データ構造の変更 なし。	
7	■環境に関する変更 なし。	

「発注仕様書」の内容を明記

この記載は変更が発生しなかったケース。関数レベルではなく一段上の機能詳細レベルで記載

発注仕様書#を確認するためのシナリオと内容を記載→この項目をパスすると合格!

テストスクリプトやエビデンスを一式で提供→必要に応じて再現試験も可能とした

実証実験プロジェクトの救済（変更設計書）

参考：変更設計書テンプレート

『「派生開発」を成功させるプロセス改善の技術と極意』から引用

変更設計書

プロジェクト名	XDDPプロジェクト	作成日	2007年 7月 10日	
ソース名/ タスク名	RectCalPart1.c	作成者		
		修正者		<input type="checkbox"/> 修正
変更要求仕様	計算の基準が設定値だけの状態から、一時的に基準値を変更できるようにする	確認者		<input type="checkbox"/> 確認
# CAL01.2		見積り行数	35	見積り時間
		変更行数		作業時間

修正方針
特になし

データ構造の変更
なし

関数呼び出し構造の変更
iConvRectAllのモジュールの凝集度が悪化するのを回避するために、このなかの処理を一部分離し、新しく「iConvRectSub」の関数を設けて、そこに関連する処理を移して、基準値の変更処理をそこで実現する。その他の下位モジュールには変更はない。

関数外の変更

項目 #	変更内容	予想行数	
1	関数名変更 (iConvRect → iConvRectAll) にともなって定義を変更する	1	<input type="checkbox"/>
2	分離独立する関数 (iConvRectSub) の定義を追加する	1	<input type="checkbox"/>

関数の変更

関数名	iConvRectAll (Vert, Rect, Base)		<input checked="" type="checkbox"/> 変更、 <input type="checkbox"/> 追加、 <input type="checkbox"/> 削除
-----	---------------------------------	--	--

変更内容：

項目 #	変更内容	予想行数	
1	関数のパラメータに、前回の基準値を表す情報 (Baseint) を追加する。	1	<input type="checkbox"/>
2	計算前に、測定関係設定情報 (TblSetInf) の「Kjun」から基準値を取り出ししている処理を切り出して、新しい関数 (iConvRectSub) に分離し、Baseの値をその関数に引き継ぐ。関数からの戻り値を新しい「基準値」として計算する。その後の計算式に変更はない。	7	<input type="checkbox"/>
3	iConvRectSubの関数仕様書は、この変更設計書に添付する。	20	<input type="checkbox"/>

確認項目：

項目 #	確認内容	チェック
	iConvRectSubの関数を繋いだ状態でテストを行う	
1	設定情報のKjun=35, Vert=120, Rect=25, Base=0としたとき、基準値としてKjunの値が採用されて関数の戻り値が「1230」となる	
2	設定情報のKjun=35, Vert=120, Rect=25, Base=45としたとき、基準値としてBaseの値が採用されて関数の戻り値が「1375」となる	
3	設定情報のKjun=35, Vert=181, Rect=25, Base=45としたときは、エラーとして「-2」が返される	

関数名

		<input type="checkbox"/> 変更、 <input type="checkbox"/> 追加、 <input type="checkbox"/> 削除
--	--	---

変更内容：

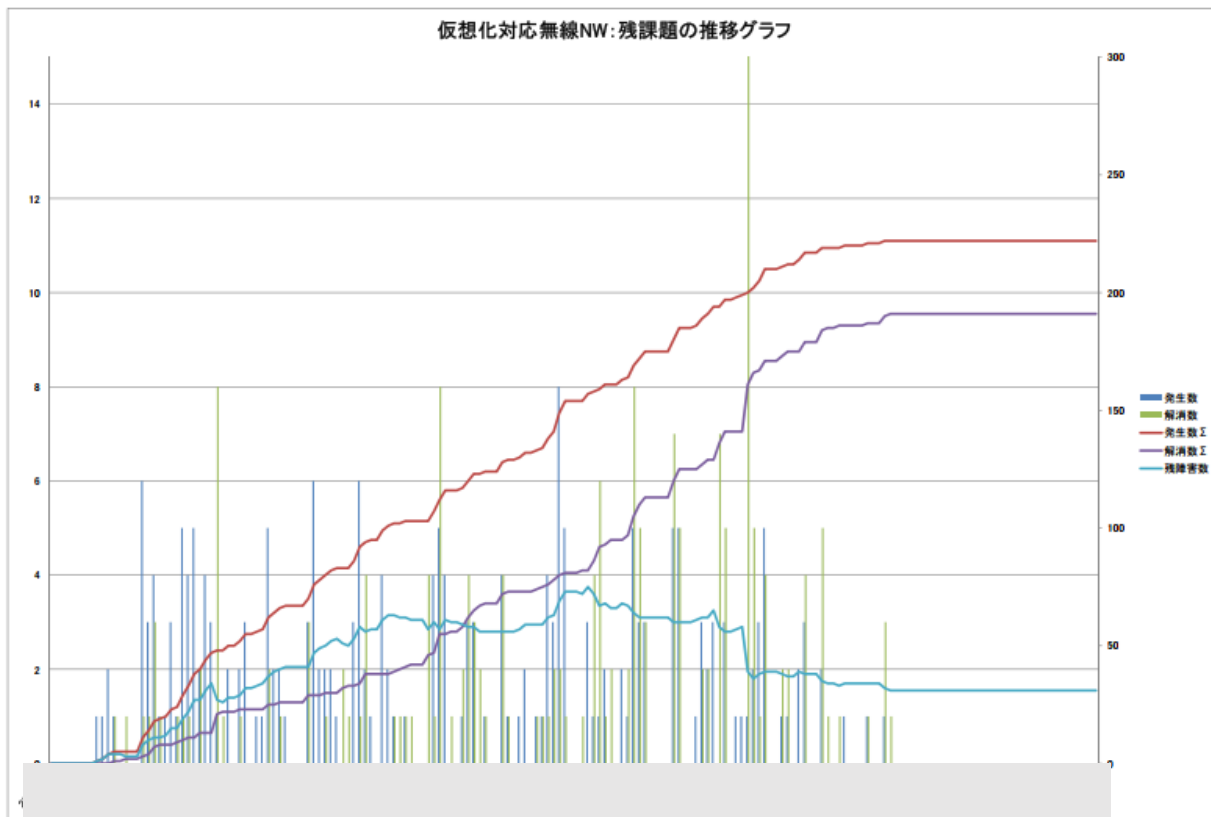
項目 #	変更内容	予想行数	
			<input type="checkbox"/>

確認項目：

項目 #	確認内容	

実証実験プロジェクトの救済（変更設計書）

SVN/Redmineでの残課題管理。残課題は「発注仕様書」を元にUSDM表現した項目数で算出。朝or夕のミーティングで進捗状況・残課題を日次で管理



[リビジョン]: 2487

[日付]:

[作者]:

[コミットログ]:

[AI#121]テスト項目★3および★4の結果ドキュメント追加

■チケット

refs #3252

■詳細

テスト項目

★3：フロー検出によるHOおよび端末数によるスライス拡張

★4：フロー検出によるHOおよびトラフィック量によるスライス拡張

のスクリプトを作成した。

その際のテスト結果ドキュメントを追加した。

+++++

A:追加 U:変更 D:削除

+++++

A flow_detection_ho/
A flow_detection_ho/TeminalSetJoinableSlice_ho_ver.sh
A flow_detection_ho/create_vBS_ho_ver.sh
A flow_detection_ho/flow_detection_ho.py
A
A

見.docx

見.docx



スマートロックシステムの派生開発の実践(USDMM/TM)

対象

- 対象ハードウェアはパートナーで製造。これまでスマートロックではじめての**QRコードリーダーを搭載した製品派生開発**。
- **RemoteLOCKクラウドをプラットフォームとして、複数のパートナーのサービスを統合してホテルでのQRコードのユーザエクスペリエンスを実装**

問題

- 営業チームの提案で導入ホテルが決定。2022年8月にホテルの営業開始、6月に現場施工、逆線を引くと2022年3月には製品開発完了が必須！
- 2021年後半にRemoteLOCK 8jのQRコード派生開発は計画していたが15ヶ月かかるとハードウェアパートナーが計画
- **トップ同士で1年前倒し(15ヶ月→3ヶ月)で派生開発を合意**
 - 2022年1月に正式開発スタート
 - **2022年6月設置工事に合うのだろうか？**

スマートロックシステムの派生開発の実践(USDMM/TM)

RemoteLOCK（リモートロック）とは

- RemoteLOCKはWi-Fiを通じてインターネットに接続し、クラウド上で多数の利用者の入室を管理します。遠隔で利用者毎に暗証番号やQRコードなどのカギ発行を行い、ゲストはキーレスで施設の利用ができるようになります。

複数の拠点・ドア・入室者を一元管理



細やかな入室制御でセキュリティを向上



運用をより便利にする高度な機能



多彩な機能でビジネスを強力にバックアップ！

クラウドだからメンテナンスフリーで常時最新バージョンの機能をお使いいただけます。

スマートロックシステムの派生開発の実践(USDMM/TM)



派生開発推進協議会

AFFORDD

経緯：お客様の課題

Before

非効率でムダの多いフロント業務



- 宿泊者カードへの記入
- 物理鍵（カード）の受け渡し、預かり
- フロント行列の整理や誘導
- チェックアウト時の精算対応

After

事務作業を削減しおもてなし強化へ



- 周辺観光の案内
 - ウェルカムドリンクの提供
 - オプションサービスの紹介
- ▶アップセル・クロスセルやファンづくりへ

省人化で人手不足に対応。人は付加価値の高い業務に注力

スマートロックシステムの派生開発の実践(USD M/TM)

経緯：お客様の要求

暗証番号やQRコードが鍵代わりに

- ✓ 対面での鍵のやり取りが不要
- ✓ キーレス化で自由な滞在へ

非対面
非接触

キーレ
ス化

オンラ
イン化



店舗業務の無人化と
体験価値の向上へ

QRコード解錠イメージ動画

施設全体を制御！

①入口自動ドア②エレベータ③客室



スマートロックシステムの派生開発の実践(USDMM/TM)

経緯：ステークホルダー

- WEST inx社
 - RemoteLOCK 8j リーダー側スキャナー搭載
- RL社PCB担当：QRコード処理追加
 - RemoteLOCK 8j インナーユニット内のPCB
- RL社クラウド開発担当：QRコード処理追加
 - RemoteLOCKクラウド管理画面
- PMS/チェックインシステム担当社
 - QRコード処理追加


※施工については以下と調整が発生

- 施主：霞ヶ関キャピタル社（FAV HOTELシリーズを全国15拠点で展開、全てRemoteLOCK）
- 施工：ゼネコン、サブコン、電設会社、建具設備会社、鍵屋



スマートロックシステムの派生開発の実践(USDМ/TM)

解決：USDМ(パートナー連携)

- RemoteLOCK8j-Q派生開発についてUSDМを記載、ステークホルダ(パートナー)を列に並べる。段階的に詳細化▶詳細は各社で検討。USDМからパートナーの詳細設計にリンク
- USDМの記載内容の変遷
 - システム間IF変更も含め列挙。パートナー列は影響のあり、なしを記載
 - 該当システム/サブシステムの修正内容について記載
 - パートナーと意識合わせ  修正を繰り返す

#	種別	内容	対応時期	Where							ALMEX社	施工影響	その他
				RemoteLOCKクラウド	メイン基板	WEST Inxデンキーリーダー(基板群)	センサー基板	モーター基板	金型・部品				
6.1	要求	メイン基板・クラウド側でデンキーリーダーが、FelCaがQRコード対応か識別可能にする。	QR-A1										
6.2	理由	外部API経由でのPIN発行時に、QRコードがIdmにするか決定できるようにするため。											
6.3.1	仕様	デバイス設定画面に「リーダー種別」メニューを追加し、FelCa、QRコードの2種類とする。 デフォルトはFelCaとする。		あり	なし	なし	なし	なし	なし	なし	なし	なし	
		上位4bitは0xF、下位4bitは送信バイトを示す。 上位4bitは0xE、下位4bitは送信バイトを示す。 は、PINコードのフィールドではなく、ICカード(FelCa)と同様に、TOBIRAと共通で管理・認証するという仕様とあわせるため。		あり	あり	あり	なし	なし	なし	なし	なし	なし	
		8bytesのデータとする。4 - 最大16桁 (0 - f) 0xFFFFFFFFFFFFFFFF のように0xを付与したQRコードを作る。 1234567890 でエンコードされたQRコードを読み取り可能にする。 QRコードは、数字で桁数チェックなしで進めたい。		あり	あり	あり	なし	なし	なし	なし	なし	なし	
		すき、想定利用環境での適正な読み取りレベルの担保する。	QR-A1										
		【(パネルデザイン)】QRコードロックはブラックパネルとする。		なし	なし	なし	なし	なし	あり	なし	なし	なし	
6.3.2	仕様	【(パネルデザイン)】下記は一案。利用者へのわかりやすさを優先したい。ブラックパネルがNGの場合はシルク網糊なども想定したい。カメラへの距離をわかるようにしてほしい。 < QR提示位置のわかりやすさ1> カメラ画面にリンク状の発光部分を設けてQRコードをかざす場所を示す。 < QR提示位置のわかりやすさ2> さらに(パネル印字で、カメラ周囲にQRコードを模した白のアイコンを入れる。(リンクアイコンの外周または横)		なし	なし	なし	なし	なし	あり	なし	なし	なし	
		<ロゴ> ロゴは、デバイス側、WESTとRemoteLOCKの両方で使用する。											

ビジョン
制約条件
IF変更
etc...



How
は各社内部の活動

スマートロックシステムの派生開発の実践(USDM/TM)

解決：パートナーとのコミュニケーション

- マルチパートナーでのプロジェクト管理・特にWESTinx社とは週1で検討

会を開催

- 影響範囲の識別・明確化

- システム制約事項
- 修正箇所の最小化
- 機能と期間のトレードオフ

- ロードマップをパートナー間で合意

- 開発と製造スケジュール短縮

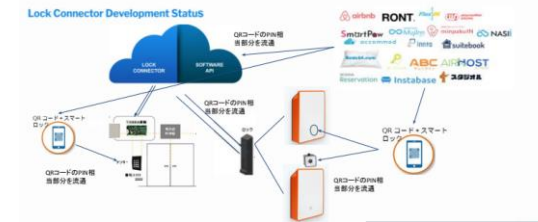
- IoTデバイスの製造と設置で3ヶ月

- ゼネコン・サブコンと調整

4. HWの変更イメージ

E06のFelIcA対応をベースにしてQRコード対応について検討した。HWの変更イメージは下記2つ。

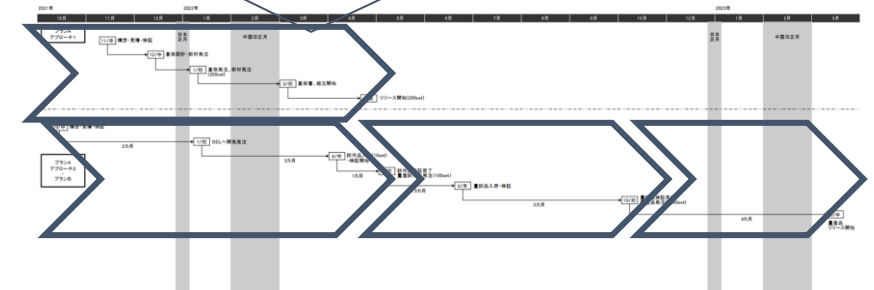
- 案1. 既存テンキーリーダーのNFCリーダ部分に取り付ける
-FelIcAは読み取りできなくなる。耐候性は担保しやすい。設置性は担保しやすい。
- 案2. テンキーリーダー上部(もしくは別の場所に)追加で取り付ける
-FelIcAと並列で読める=ACS(TOBIRA)リーダと同様。耐候性は劣る可能性がある。



QRコードに設定する内容としては下記が想定される。それぞれメリット/デメリット(Pros/Cons)を一覧にまとめる

#	方式	メリット(Pros)	デメリット(Cons)
1	4-10桁の数値として扱う(問題があれば変更する)	<ul style="list-style-type: none"> ACS(TOBIRA)のFelIcA対応と仕様が共通化可能 クラウド変更不要 メイン基板変更不要 	<ul style="list-style-type: none"> 文字列は扱えない RLクラウドもしくはラッパーAPIでQRコード作成とメール添付処理が発生
2	外部システムの利用を前提に、外から与えられたASCII文字列からPINコードをテンキーリーダーで導出して認証	<ul style="list-style-type: none"> 人間が認識しやすくなる 	<ul style="list-style-type: none"> テンキーリーダーの処理があるため、認証までのタイムラグが発生する可能性がある。 TOBIRAのQRコードルールと別になってしまう。
<p>★前提：QRコードのPIN部分のみ読み取って認証する方式。 例えば、xxxxxxxxxxxx(文字列、URLなど)-12345678の12345678がPIN部分とする考え方</p>			
3	外部システムの利用を前提に、外から与えられたASCII文字列を流して認証	<ul style="list-style-type: none"> 人間が認識しやすくなる 	<ul style="list-style-type: none"> RLクラウド、メイン基板、テンキーリーダーすべて変更が発生する。 メイン基板のメモリを圧迫し、記憶できるコード数が減少する。

USDMをアンカーとした
パートナーとのコミュニケーション、
合意形成の短縮化

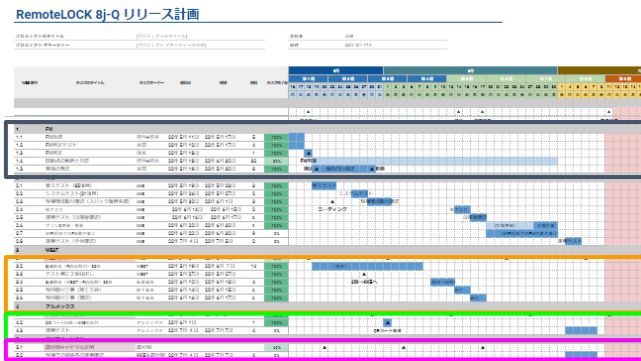


スマートロックシステムの派生開発の実践(USDМ/TM)

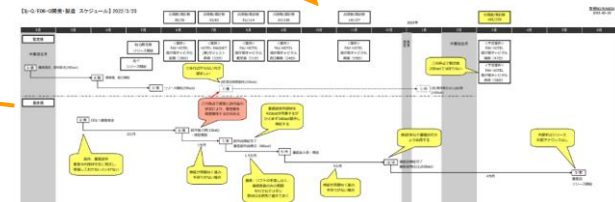
解決：TMによるトレーサビリティとスケジュール管理

- USDМがアンカーになる
- パートナーのスケジュールをベースにテスト計画をKKEで作成・実施
- 施工のための資料も作成
- 2022年6月までに施工完了

#	種別	内容	対応時期	RemoteLOCK		VEST Inx		センサー基板	モーター基板	金型・部品	ALMEX社	施工影響	その他
				クラウド	メイン基板	デンキリーダ(基板群)							
6.1	要求	メイン基板・クラウド側でデンキリーダが、FelCaがQRコード対応が個別可能にする。	QR-A1										
6.2	理由	外部API経由でのPIN実行時に、QRコードがIdmにするか決定できるようにするため。											
6.3.1	仕様	デバイス設定画面に「リーダ種別」メニューを追加し、FelCa、QRコードの2種類とする。 ・デフォルトはFelCaとする。		あり	なし	なし	なし	なし	なし	なし	なし	なし	
6.3.2	仕様	[2111012追加] QRコードの場合、上位4bitは0xF、下位4bitは送信バイトを示す。 #FelCaの場合、上位4bitは0x0、下位4bitは送信バイトを示す。 理由：QRコードは、PINコードのフィールドではなく、ICカード(FelCa)と同じ領域でクラウド・TOBIRAと共通で管理・認証するという仕様とあわせるため。		あり	あり	あり	なし	なし	なし	なし	なし	なし	
6.3.3	仕様	[2111130追加] FelCa Idmと同じ8bytesのデータとする。4 - 最大16桁 (0 - f) ・16進の場合は、0xFFFFFFFFFFFFFFFFのように0xを付与したQRコードを読み取り可能とする。 ・10進の場合は、1234567890 でエンコードされたQRコードを読み取り可能とする。 ・デンキリーダは、数字で桁数チェックなしで進めたい。		あり	あり	あり	なし	なし	なし	なし	なし	なし	
7.1	要求	【デザイン、カメラ】	QR-A1										
7.2	理由	QRコードのわかりやすさ、想定利用環境での適正な読み取りレベルの担保するため											
7.3.1	仕様	【(ネラデザイン)】QRコードロックはブラック(ネラ)とする。		なし	なし	なし	なし	なし	なし	あり	なし	なし	
7.3.2	仕様	【(ネラデザイン)】下記は一画、利用者へのわかりやすさを優先したい。 ブラック(ネラ)の場合にはツール用領域とも確定したい。 カメラへの邪魔をわかるようにしてほしい。 < QR提示位置のわかりやすさ1 > カメラ周囲にリンク状の発光部分を設けてQRコードをかざす場所を示す。 < QR提示位置のわかりやすさ2 > さらに(ネラ)印字で、カメラ周囲にQRコードを模した白のアイコンを入れる。 (リンクアイコンの外周または横) <ロゴ> ロゴ1のみを採り、WISTとRemoteLOCKロゴは使用しない。		なし	なし	なし	なし	なし	なし	あり	なし	なし	



Product update



今後について(プロセス・技術・手法)

- XDDPの部分適用でも有効性はあるの言葉（仮説）から、XDDPの三点セットとPFDを用いて問題解決プロセスを実践した
- 三点セットの要素の抽象度を上げることで、現代の開発においても有効であると実感した
- 今後はSaaS企業でもハードウェアとソフトウェアとの連携が発生
 - プロセス自体のテーラリング技術
 - API連携は疎結合であり重い開発プロセスは不要
 - IoTデバイス自体の共同開発はそれなりに重い
 - STM32/AWS IoT Core/クラウドサービス上の開発・運用技術/広範な知識が要求されるなかで、部分理解から必要な技術を取得しつつ、問題解決する技術が必要。

おわりに

お世話になりました皆様に御礼申し上げます。

- これまで仕事に関わって頂いた皆様
- 株式会社 構造計画研究所の皆様
- 派生開発推進協議会（AFFORDD） 斎藤代表およびT6/T4の皆様
- 故・清水吉男さん

ご清聴ありがとうございました！！！！



派生開発推進協議会

AFFORDD