

**イノベーションとは、新しいものを生産する、あるいは
既存のものを新しい方法で生産すること
オーストリアの経済学者 シュンペーター**



スマート・ソフトウェアテスト・ファクトリ (SSTF) ～ ソフトウェアテストのIoT化 ～

■ スマート・ソフトウェアテスト・ファクトリ(SSTF)とは？

スマート・ファクトリをソフトウェアテストに応用した概念。
ソフトウェアテスト内外の環境やツール、製品、人間が相互につながり、
「考えるソフトウェアテスト工場」を実現すること

SHIFT 高木 陽平

SHIFTは、ソフトウェアの「品質保証」を支援する会社

社名	株式会社SHIFT
業務内容	ソフトウェアの品質保証支援、ソフトウェアテスト事業
設立	2005年9月7日
上場	東証マザーズ 3697
従業員数	822名 ※契約社員含む(2015年10月時点)
代表者	代表取締役社長 丹下 大 Masaru Tange
所在地	【本社&東京TC】東京都港区麻布台2-4-5メソニック39MTビル 【札幌TC】北海道札幌市中央区北1条西3丁目3 札幌プラザビル 【福岡TC】福岡県福岡市中央区天神1-15-6 綾杉ビル ※TC…テストセンター
子会社	SHIFT INDIA PRIVATE LIMITED(インド国) SHIFT GLOBAL PTE LTD(シンガポール国) 株式会社SHIFT PLUS(日本)

製造業向けのコンサルからソフトウェアテスト業界に参入

製造業向けコンサルティング

2005年独立、株式会社SHIFT設立

業務改善コンサルティング

“ソフトウェアテスト”に出会う

テスト内容、生産性、品質などが
可視化されておらず、かつて製造業で
経験した**熟練者が多い**世界と同じ。

製造業向けコンサルティング実績

カメラ
大手C社



釣具
大手S社



家電
大手S社



自動車
大手N社



※他、全50社以上に業務改善コンサルを展開。

すべてのソフトウェアに
Made in Japanの品質を

従来のソフトウェアテストに製造業の品質保証ノウハウを
転用し、コンサルティング～ソフトウェア検証サービスを提供しております。

ソフトウェアテスト業界で成長率No.1

設立から10年、
ますます
大きく成長中。



ITの現場、コンサル、経営、学問の全てを実際に経験

■ 学歴

日本工業大学 情報工学科 2003年卒

東京理科大学大学院 技術経営学科(MOT) 2012年卒

■ 職歴

- 大手Sler (2003 - 2009) SE

組込み機器(通信関係)の開発

CMMI、ISO9001の内部監査員を兼務

- ベンチャー (2012 - 2015) 技術責任者/ITコンサルタント

フィリピン事務所の立ち上げ

組込み会社(車載関係)へのコンサルタント

- **SHIFT 2015～ IoT・組込みセクションの責任者**

IoT・組込みセクションの立ち上げ

IoT・組込みセクション長に就任



高木 陽平

Agenda

1. なぜ、SSTFを始めたのか？
2. どのような、SSTFを構築したのか？
3. SSTFの今後について

Agenda

1. なぜ、SSTFを始めたのか？

2. どのような、SSTFを構築したのか？

3. SSTFの今後について

なぜ、SSTFを始めたのか？

某メーカー企業様のコンサルをして思いました

テストマネージャの激務をなんとかしたい

なぜ、SSTFを始めたのか？

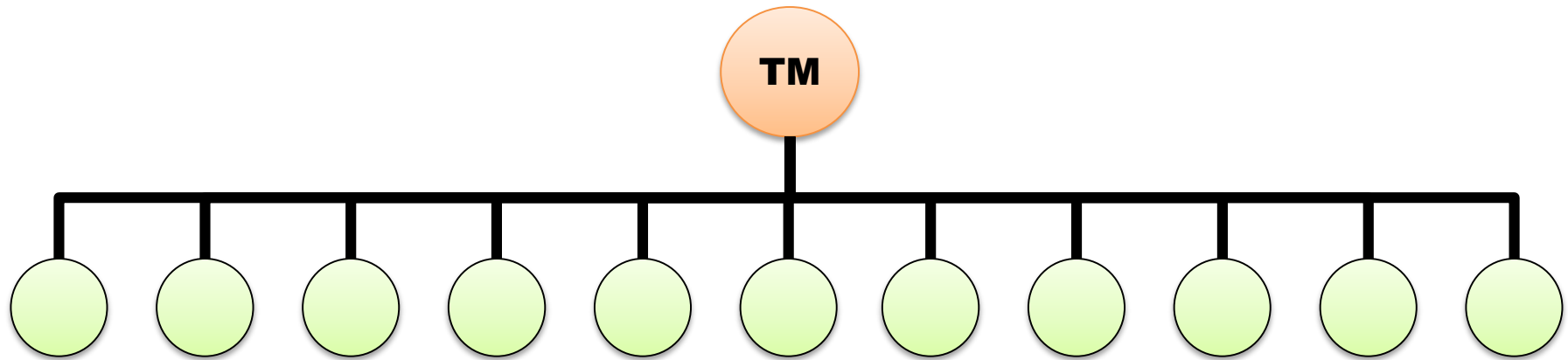
某メーカー企業様のプロフィール

- 組み込み製品のFW検証
- システムテスト
- 人手によるテストが物理的に不可能(テストプログラムが必要)
- 評価機器700台を並列で実行
- 規格等の高度な知識が必要

なぜ、SSTFを始めたのか？

某メーカー企業様の組織体制

■ フラット組織



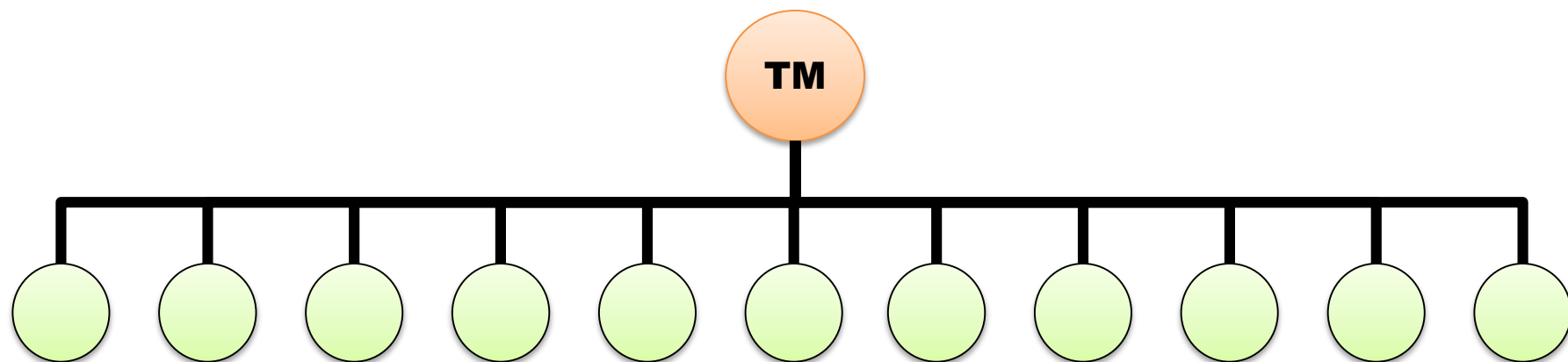
【良い所】

意思決定のスピードが上がる
風通しが良い などなど

なぜ、SSTFを始めたのか？

某メーカー企業様の組織体制

■ フラット組織



【良い所】

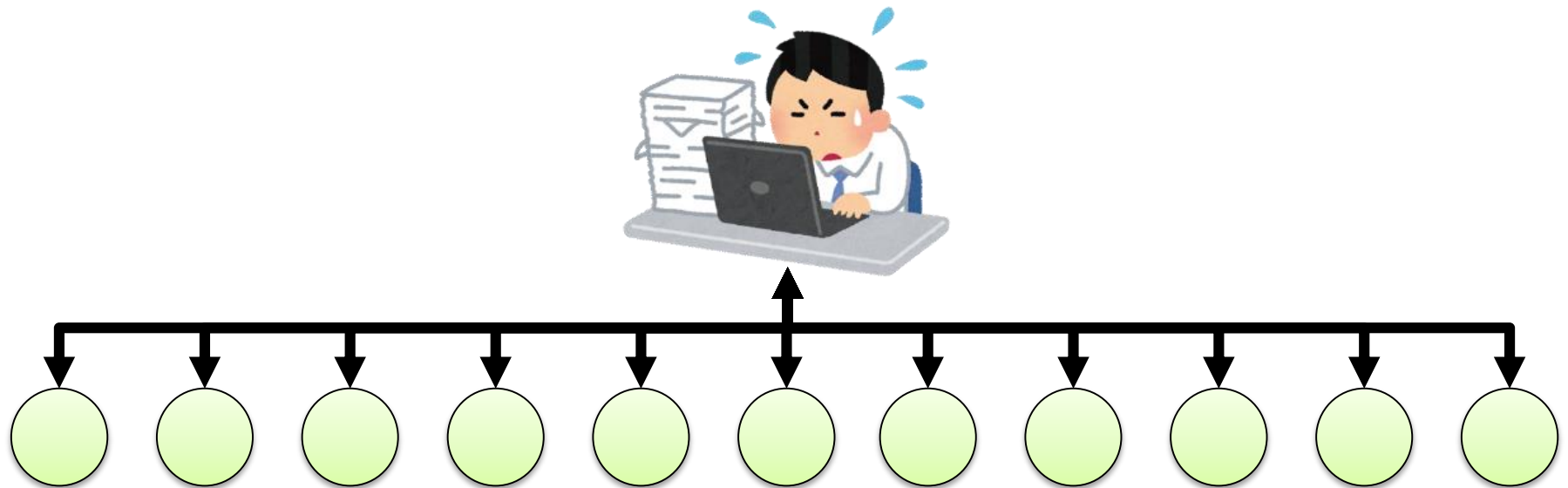
意思決定のスピードが上がる
風通しが良い などなど

上手く機能していた。ただ1点を除いて、、、

なぜ、SSTFを始めたのか？

フラット組織はテストマネージャに負荷がかかる

- 大量の情報を処理しなければならない
- 事務作業が大量に発生する
- メンバーのフォローをしなければいけない



なぜ、SSTFを始めたのか？

テストマネージャがパンク寸前の状態だった

- 夜は誰よりも遅くまで残っている
- 朝は誰よりも早く出社している
- 土曜日はいつもいる
- 日曜日もいることが多い
- **平均150 時間/月以上の残業**



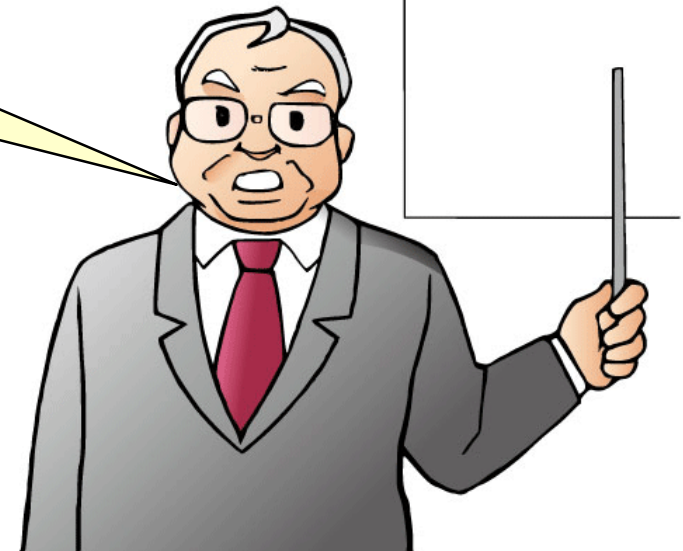
なぜ、SSTFを始めたのか？

そんな時、上層部から新たな指令がきた

なぜ、SSTFを始めたのか？

そんな時、上層部から新たな指令がきた

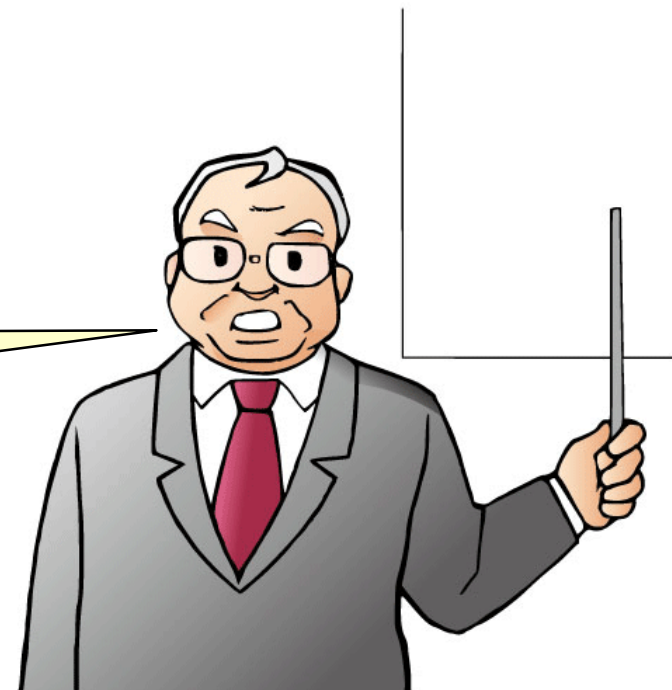
我が社は厳しい競争に勝ち抜かなければならない！
その為に、競合他社より**1日でも早く新製品を市場に投入**する必要がある



なぜ、SSTFを始めたのか？

そんな時、上層部から新たな指令がきた

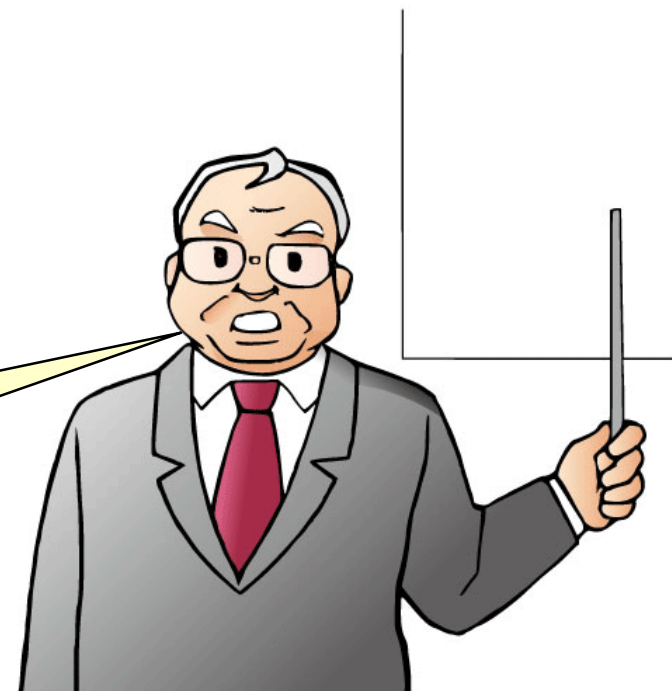
今までの**半分の時間で商品をリリース**するぞ！



なぜ、SSTFを始めたのか？

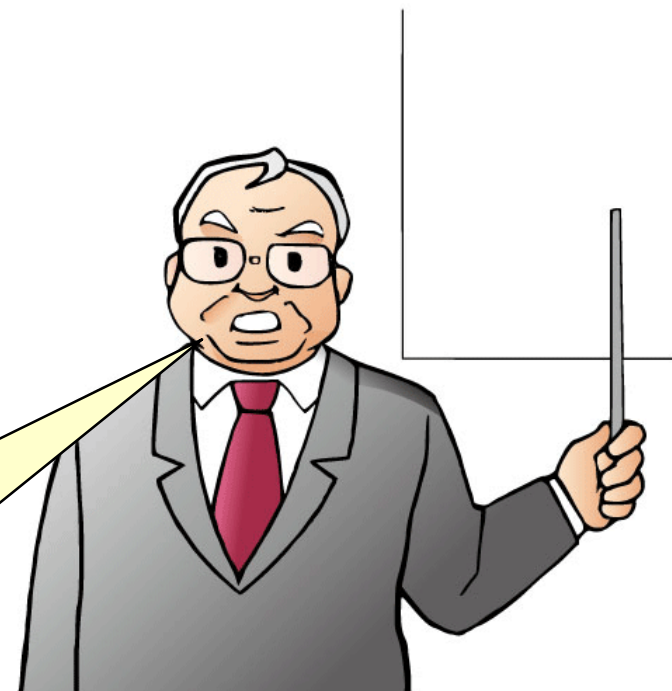
そんな時、上層部から新たな指令がきた

ただし、機能は落とすな！
品質も絶対に落とすな！



なぜ、SSTFを始めたのか？

そんな時、上層部から新たな指令がきた



その代わり、**新しい人員を4人追加する。**
評価機器も300台追加しよう！

なぜ、SSTFを始めたのか？

それを聞いて、私は思った

なぜ、SSTFを始めたのか？

それを聞いて、私は思った

テストマネージャ死ぬな！

ついでに、それを支援している私たちも死ぬ。。



なぜ、SSTFを始めたのか？

ただし、同時に、こうも思った

やらなきゃ、生き残れないのも事実！

なぜ、SSTFを始めたのか？

超難題に呆然。ただし、これだけは分かっていた

やり方を抜本的に変える必要がある！

なぜ、SSTFを始めたのか？

そんな時、一冊の本のことを思い出す



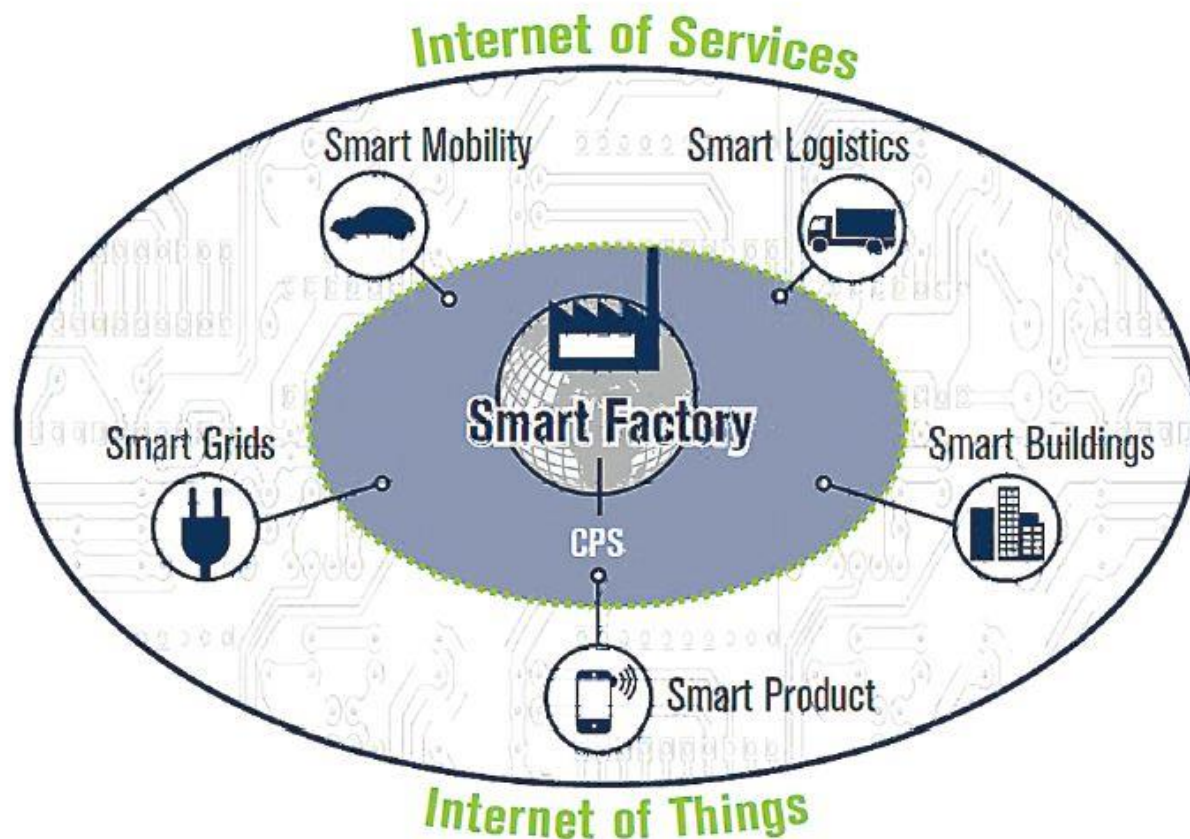
なぜ、SSTFを始めたのか？

インダストリー4.0は、第4次 IoT産業革命

			
第1次 蒸気機関 18世紀	第2次 電気エネルギー 20世紀初頭	第3次 コンピュータ による自動化 20世紀後半	第4次 IoT産業革命 2015年～
蒸気機関を利用した、英国の繊維工業が発展した	電力を利用し、ベルトコンベアを使った大量生産が実現した	コンピュータを利用し、生産の自動化が進み大量生産が進化した	IoTを利用し、工場内外の生産設備や製品、人間が相互につながり、「考える工場」が実現する

なぜ、SSTFを始めたのか？

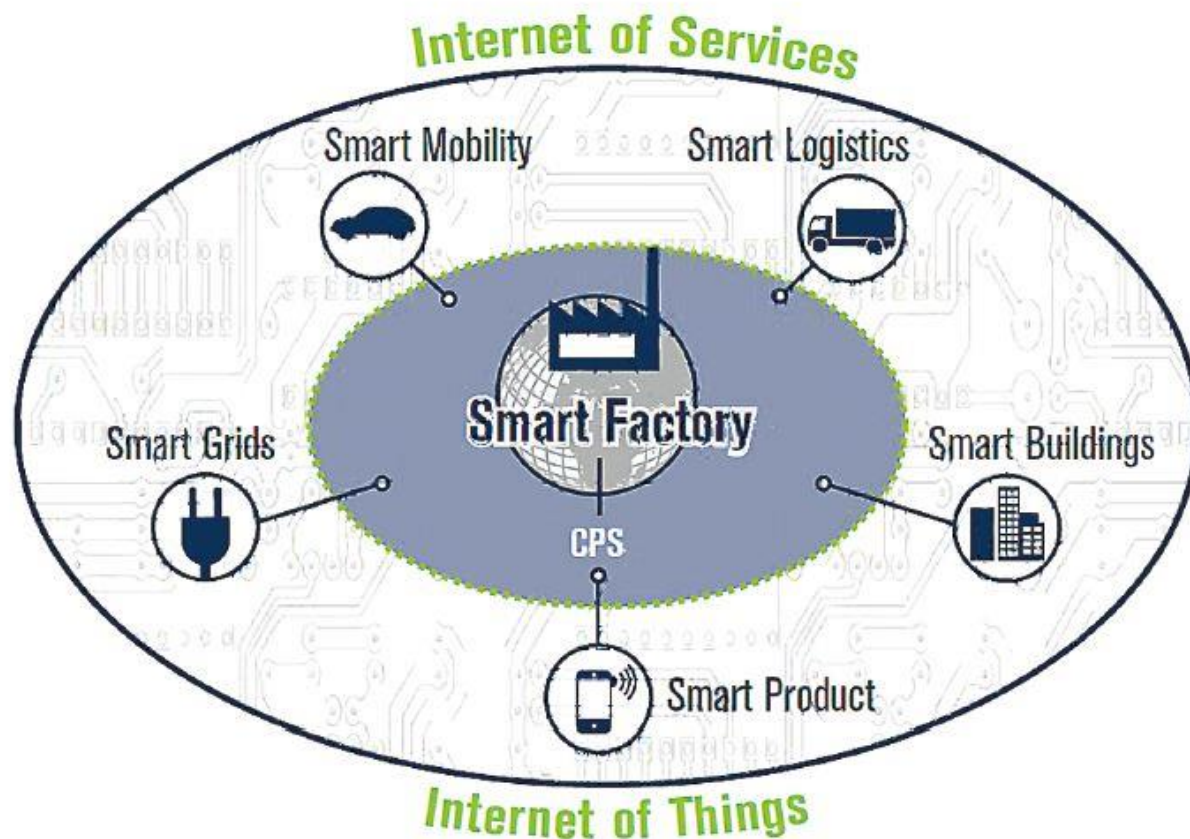
工場をあらゆるものに繋げ、飛躍的に生産性を向上させる



出典：日経テクノロジー「<http://techon.nikkeibp.co.jp/article/COLUMN/20140502/349823/?rt=nocnt>」

なぜ、SSTFを始めたのか？

工場をあらゆるものに繋げ、飛躍的に生産性を向上させる



これだ！

Agenda

1. なぜ、SSTFを始めたのか？

2. どのような、SSTFを構築したのか？

3. SSTFの今後について

どのような、SSTFを構築したのか？

実現したことは3つ。これにより生産性2倍を達成！

- ① あらゆるものをつなげた
- ② 組織体制を変えた
- ③ 進捗管理の方法を変えた

どのような、SSTFを構築したのか？

実現したことは3つ。これにより生産性2倍を達成！

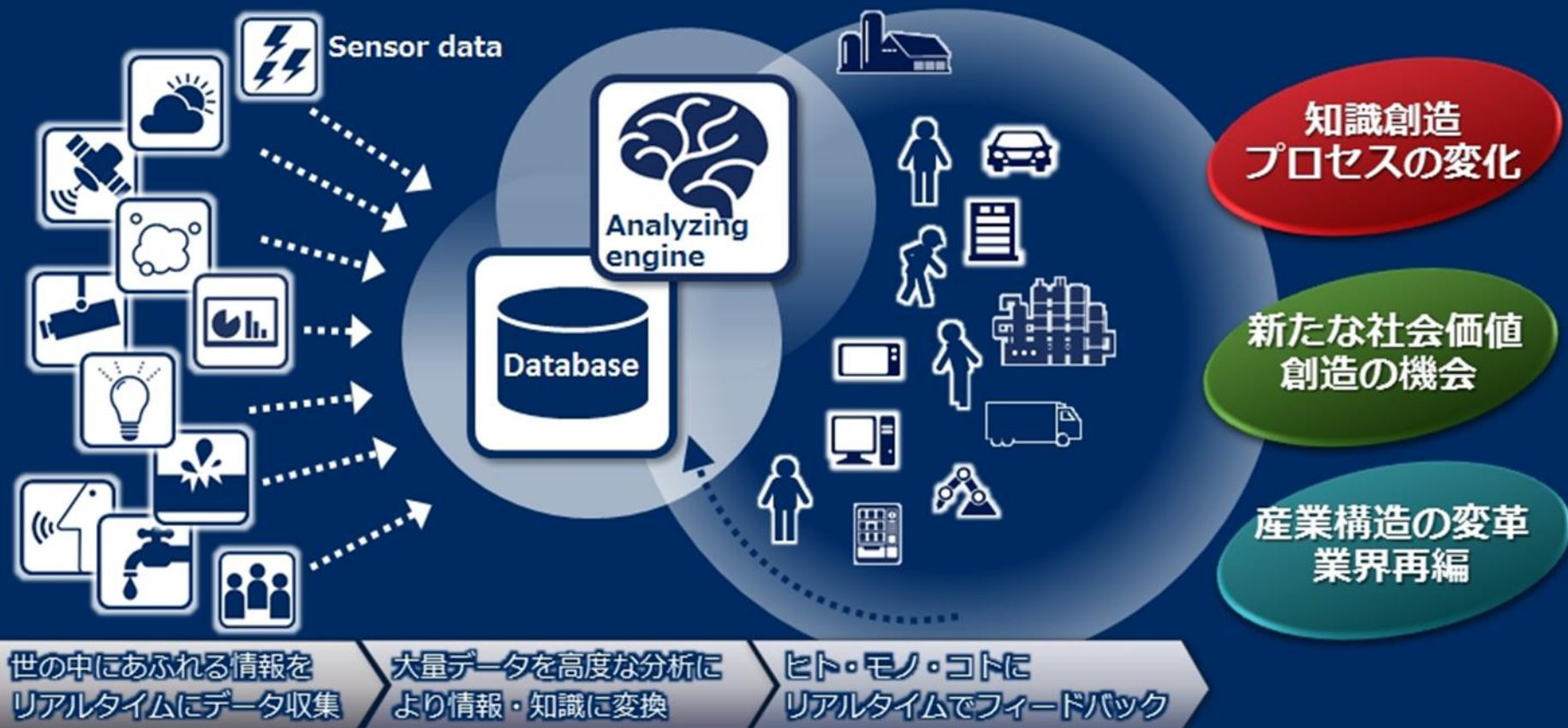
① あらゆるものをつなげた

② 組織体制を変えた

③ 進捗管理の方法を変えた

① あらゆるものをつなげた

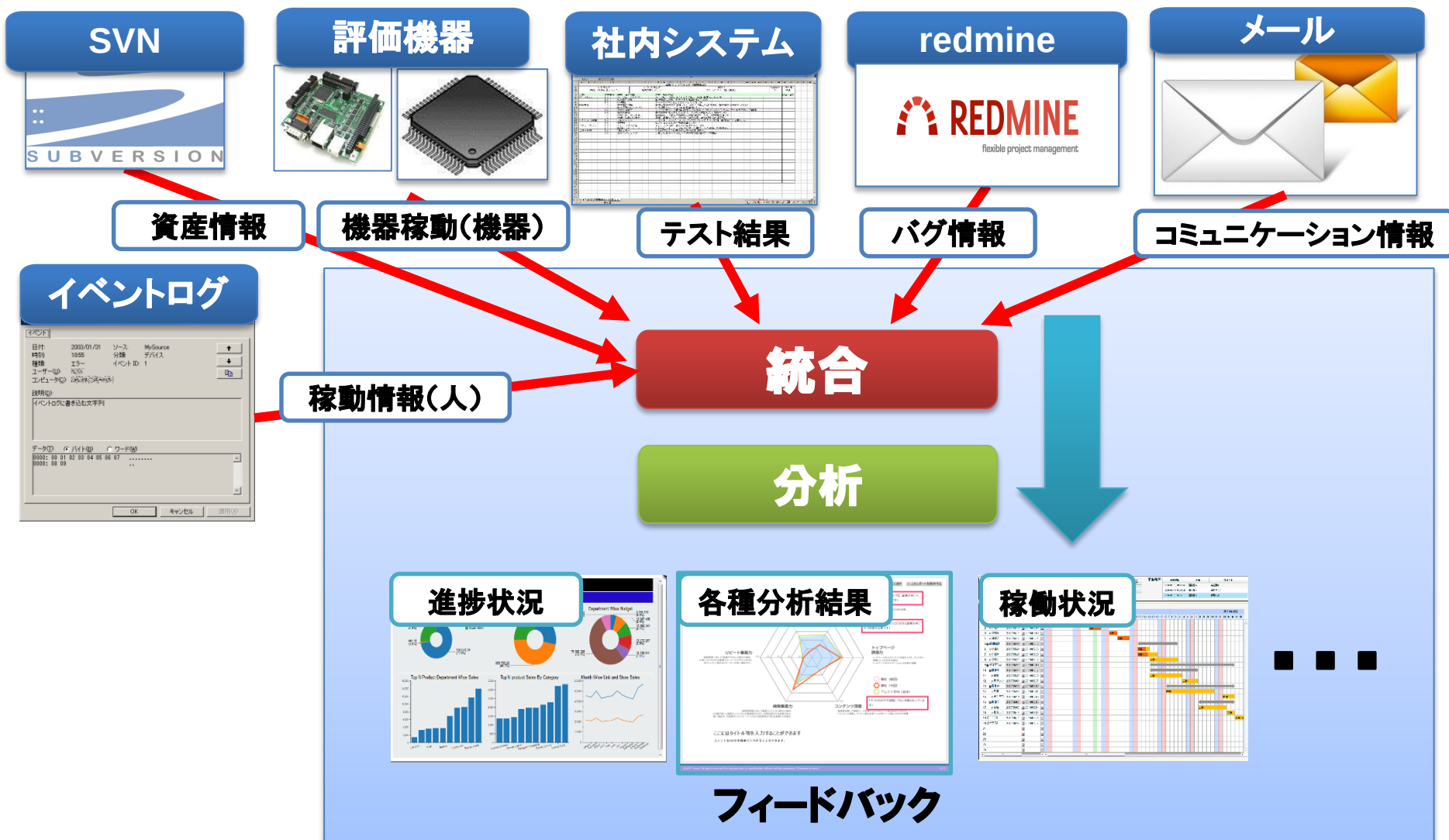
IoTの基本は、「センサ」→「解析」→「フィードバック」のエコシステム



出典: NEC HP 「<http://jpn.nec.com/iot/>」

① あらゆるものをつなげた

これがSSTFのエコシステムだ！



① あらゆるものをつなげた

何をフィードバックするか？それが問題だ

何をフィードバックするか？それが問題だ

テストマネージャを支援するフィードバックにしたい！！

① あらゆるものをつなげた

悩んだ結果、ダッシュボードを提供することにした



① あらゆるものをつなげた

ダッシュボードを提供することにした

車のダッシュボードよりも好きな例えは、野球のスコアボード



① あらゆるものをつなげた

ダッシュボードを提供することにした

車のダッシュボードよりも好きな例えは、野球のスコアボード



選手、監督、観客などの関係者全員が同じ情報を共有

みんなが情報を共有できるダッシュボードを構築

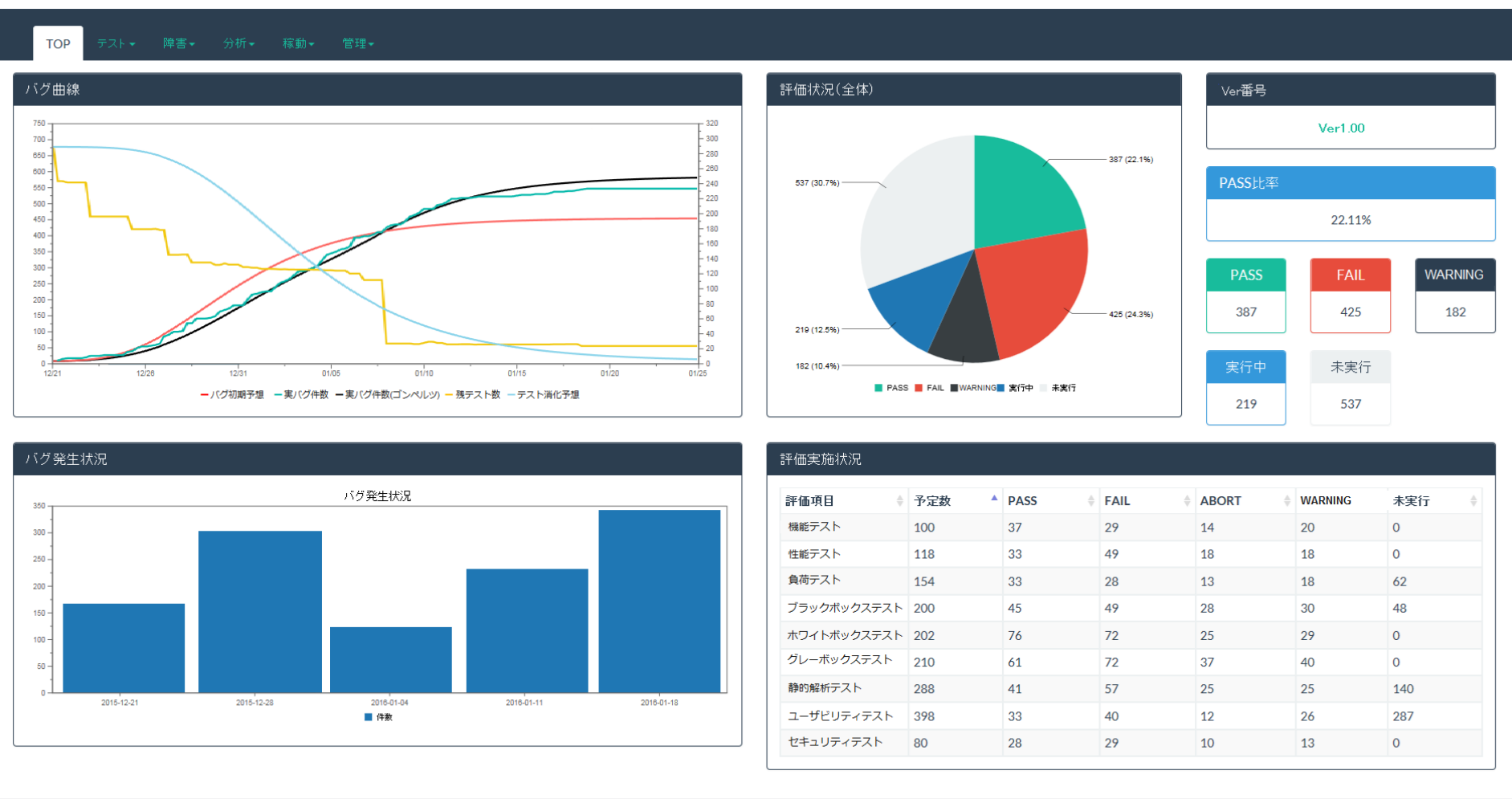
■ 主要5機能

- (1) トップ画面（テスト結果サマリー機能）
- (2) テストタイプ別テスト結果サマリー機能
- (3) フォロー対象チケット表示機能
- (4) 評価機器稼働状況表示機能
- (5) スクリプト実行履歴追跡機能

※ 業務フロー図を描き、ヒアリング、プロセスの見直しを実施して必要な機能を洗い出した

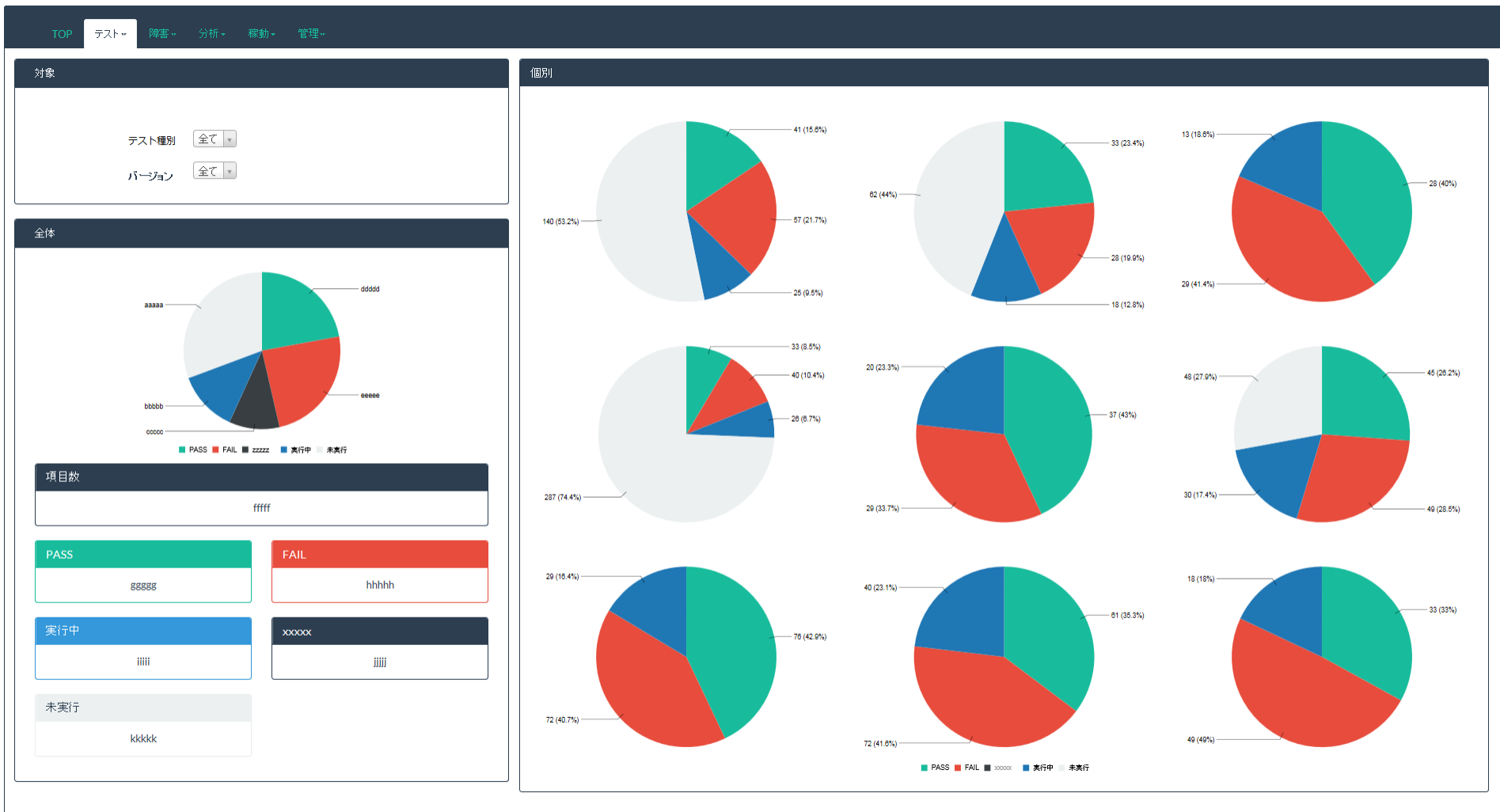
(1) トップ画面 (テスト結果サマリー機能)

評価機器・Redmineから、テスト結果を自動で収集・集計

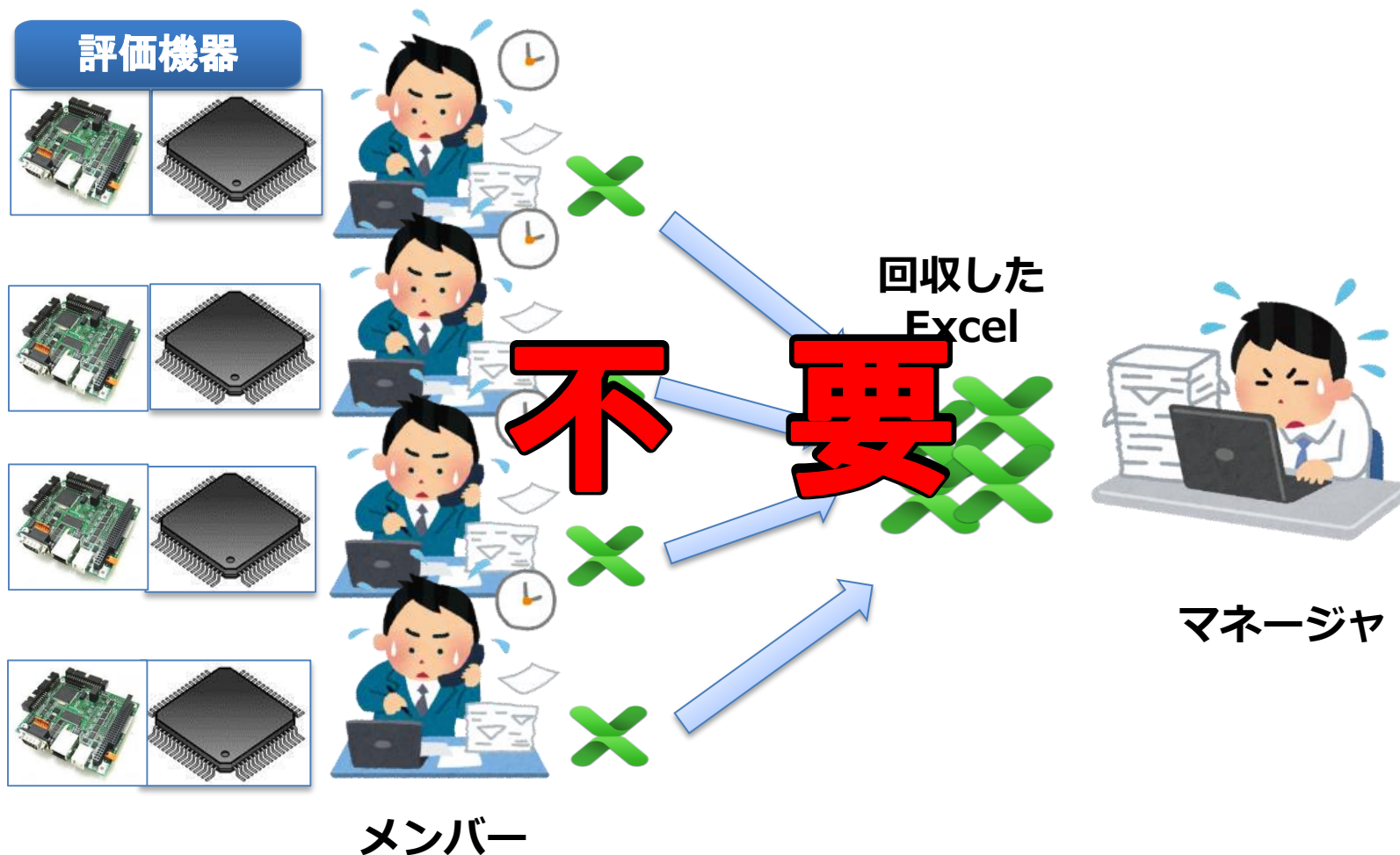


(2) テストタイプ別テスト結果サマリー機能

テストタイプ毎の結果もリアルタイムで表示



今までのエクセルでの回収と集計が不要になった

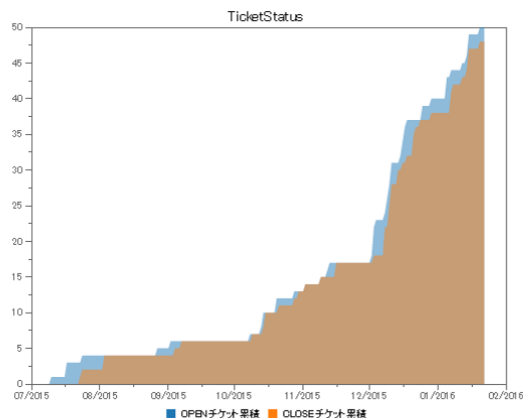


(3) フォロー対象チケット表示機能

フォローが必要なバグ (チケット) を自動で表示

TOP テスト 障害 分析 稼働 管理

発行/Closeチケット累積



Openチケット数

337

ステータス別チケット数

ステータス	件数
フィードバック	2
却下	2
新規	335
終了	3

3日以上更新されていないチケット一覧

Show 10 entries

Search:

更新日	放置日数	#No	ステータス	担当	タイトル
-----	------	-----	-------	----	------

No data available in table

Showing 0 to 0 of 0 entries

Previous Next

起票から7日以上経過しても終了していないチケット一覧

Show 10 entries

Search:

作成日	経過日数	#No	ステータス	担当	タイトル
-----	------	-----	-------	----	------

No data available in table

Showing 0 to 0 of 0 entries

Previous Next

障害状態で3日以上経過しているチケット一覧

Show 10 entries

Search:

更新日	放置日数	#No	ステータス	担当	タイトル
-----	------	-----	-------	----	------

No data available in table

Showing 0 to 0 of 0 entries

Previous Next

優先度が低いものの確認が不要になった



Accueil Projects Aide Connexion S'enregistrer

Redmine Recherche:

Aperçu Download Activité Roadmap **Demandes** Annonces Wiki Forums Dépôt

Demandes

Filtres
☒ Statut: ouvert Ajouter le filtre:
☒ Appliquer ☐ Effacer

#	Tracker	Statut	Priorité	Sujet	Date	Catégorie
☐ 3480	Defect	New	Normal	Bot filter plugin crashes when missing useragent	2009-06-11 08:50	Plugins
☐ 3479	Defect	New	High	SEGFALT in Mysql	2009-06-11 07:15	Tickets
☐ 3477	Patch	New	Normal	Fix access handler to remove the need for a separate svn-private/git-private	2009-06-10 21:54	UI
☐ 3476	Defect	New	Normal	Right-floating TOC funky in the roadmap	2009-06-10 19:52	Emails
☐ 3473	Feature	New	Normal	Can Redmine support notification when a issue will be overdue?	2009-06-10 07:17	Projects
☐ 3472	Defect	New	Normal	Role given to a non-admin user who creates a project	2009-06-10 06:37	Administration
☐ 3471	Defect	Resolved	Normal	Project managers should be able to assign "sub-project of"	2009-06-10 05:49	SCM
☐ 3470	Feature	New	Normal	Projects should inherit documents and files from sub-projects	2009-06-10 23:08	Tickets
☐ 3469	Defect	New	Normal	senAS	2009-06-10 04:48	Projects
☐ 3468	Defect	Resolved	Normal	Subversion : View Differences 500 error	2009-06-09 21:21	Administration
☐ 3467	Feature	New	Normal	Due date sort order should sort issues with no due date to the end of the list	2009-06-09 08:12	Wiki
☐ 3466	Patch	New	Normal	ja label for text_status_changed_by_changeset	2009-06-09 05:06	SCM
☐ 3465	Feature	New	Urgent	Default project	2009-06-09 04:46	Permissions
☐ 3464	Feature	New	Normal	columns "user_id" & "created_on" in tables like projects, documents & custom_fields...	2009-06-08 21:29	Custom fields
☐ 3463	Feature	New	Low	Export (all) Wiki-Pages to PDF/DOC	2009-06-09 09:02	SCM
☐ 3462	Defect	New	High	CVS path encoding problems	2009-06-08 16:24	Emails
☐ 3461	Patch	New	Normal	Manage permission on issue assignment	2009-06-10 20:41	Projects
☐ 3457	Defect	New	Normal	Default value on log text fields	2009-06-08 11:01	SCM
☐ 3454	Defect	New	Normal	Mercurial Repository Browsing Disappearing	2009-06-06 13:20	Emails
☐ 3453	Feature	New	Normal	Issue creation via email by anonymous	2009-06-07 05:51	Projects
☐ 3452	Feature	New	Normal	Per project email notifications settings panel	2009-06-07 16:38	Emails
☐ 3451	Defect	Resolved	Normal	Issue Creation Via Email not Working	2009-06-10 12:34	SCM
☐ 3450	Feature	New	Normal	Project Workflow	2009-06-07 00:31	Projects
☐ 3449	Defect	New	High	Redmine Takes Too Long On Large Mercurial Repository	2009-06-10 12:35	SCM
☐ 3448	Feature	New	Normal	Add issue watcher that isn't a maintainer	2009-06-10 11:31	Projects

1 2 3 ... 55 Suivant > (1-25/1365) | Par page: 25, 50

Formats disponibles: Atom | CSV | PDF

不

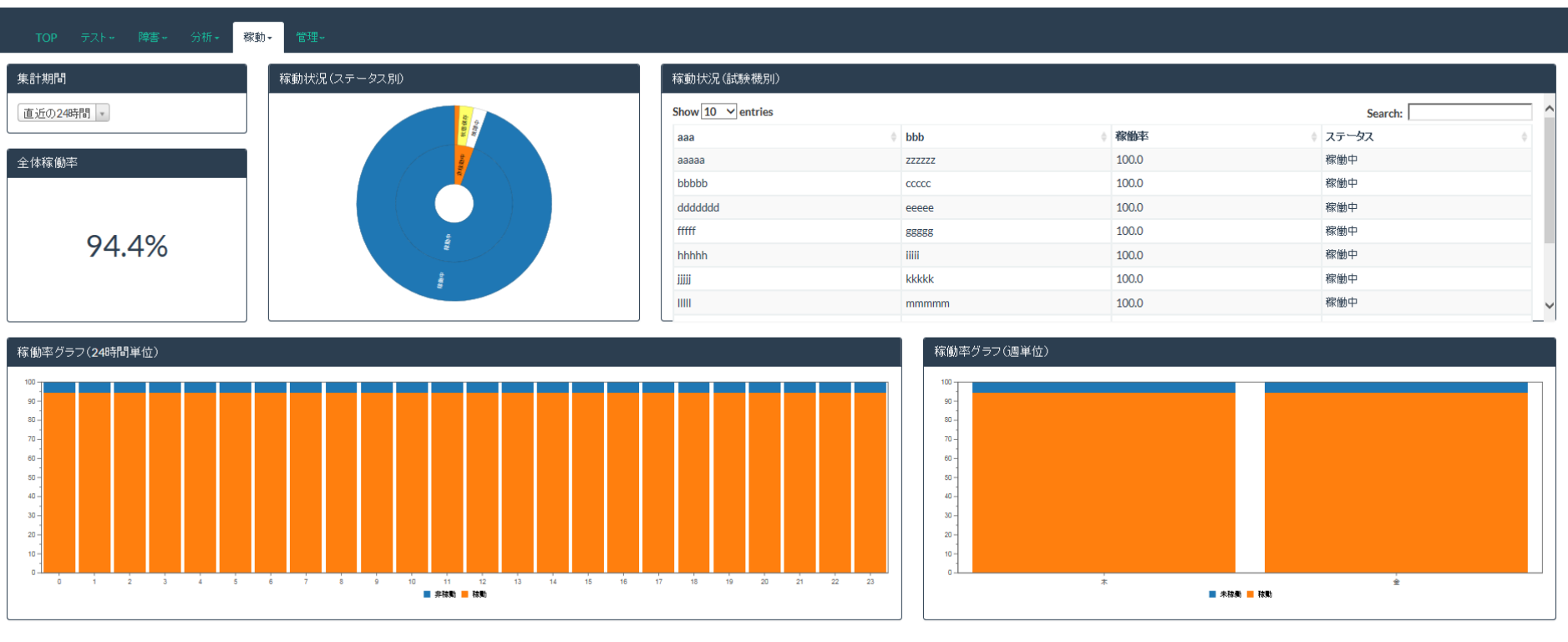
要

1つ1つ
フォロー



(4) 評価機器稼働状況表示機能

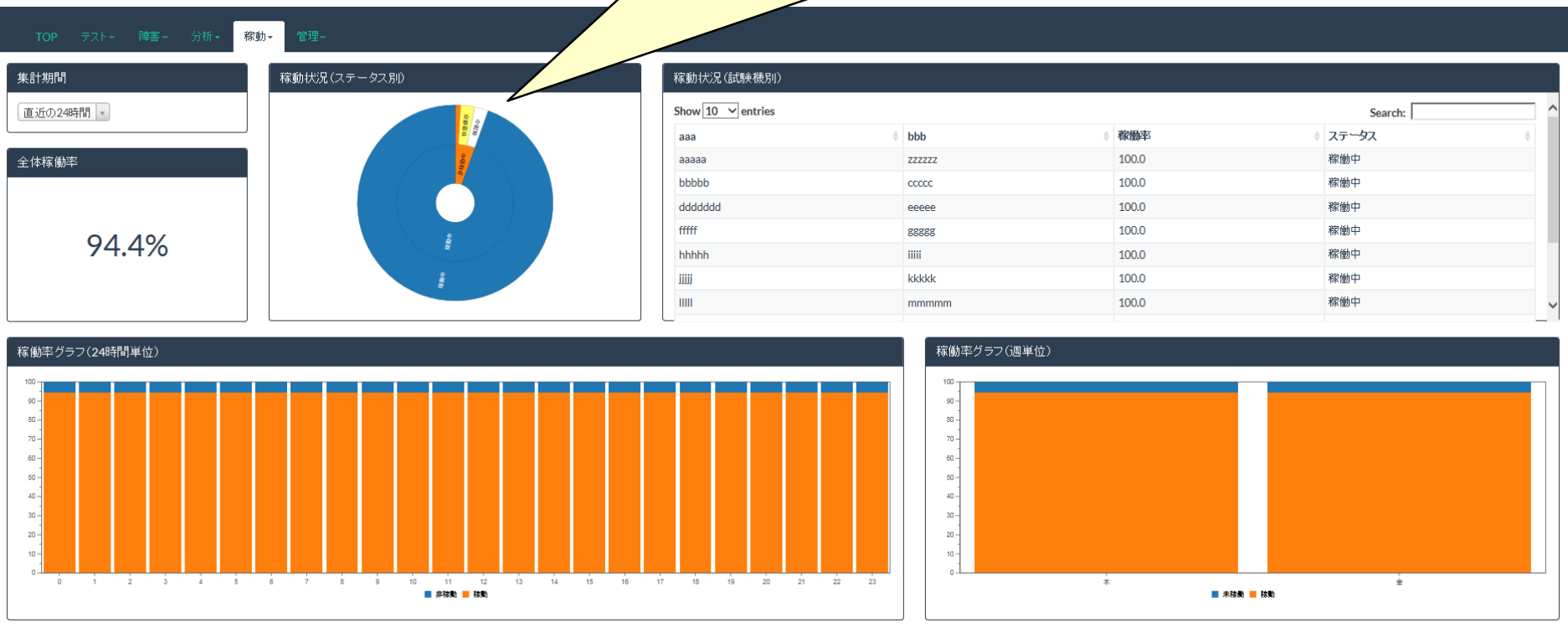
非効率な評価機器をすぐに特定できる



(4) 評価機器稼働状況表示機能

非効率な評価機器をすぐに特定できる

1000台の評価機器の状況が一目で分かる



(4) 評価機器稼働状況表示機能

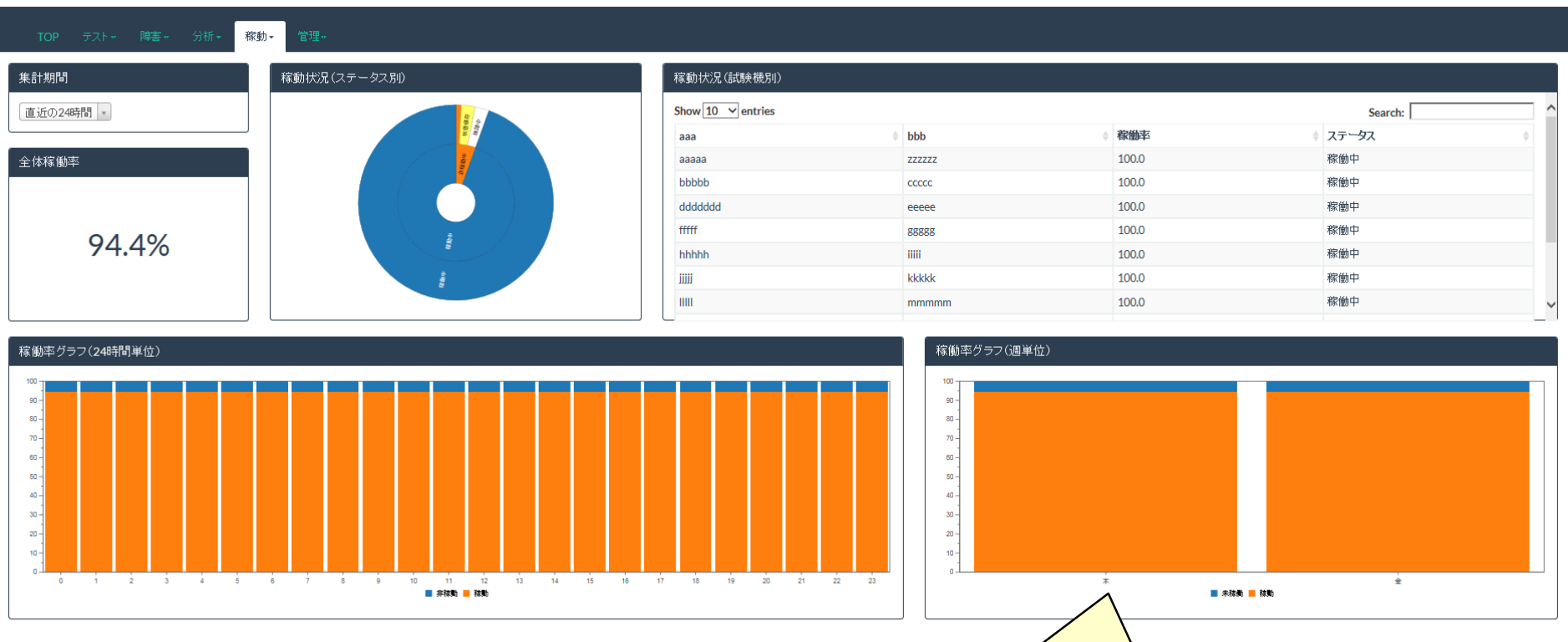
非効率な評価機器をすぐに特定できる



どの時間帯の稼働率が低いかチェック

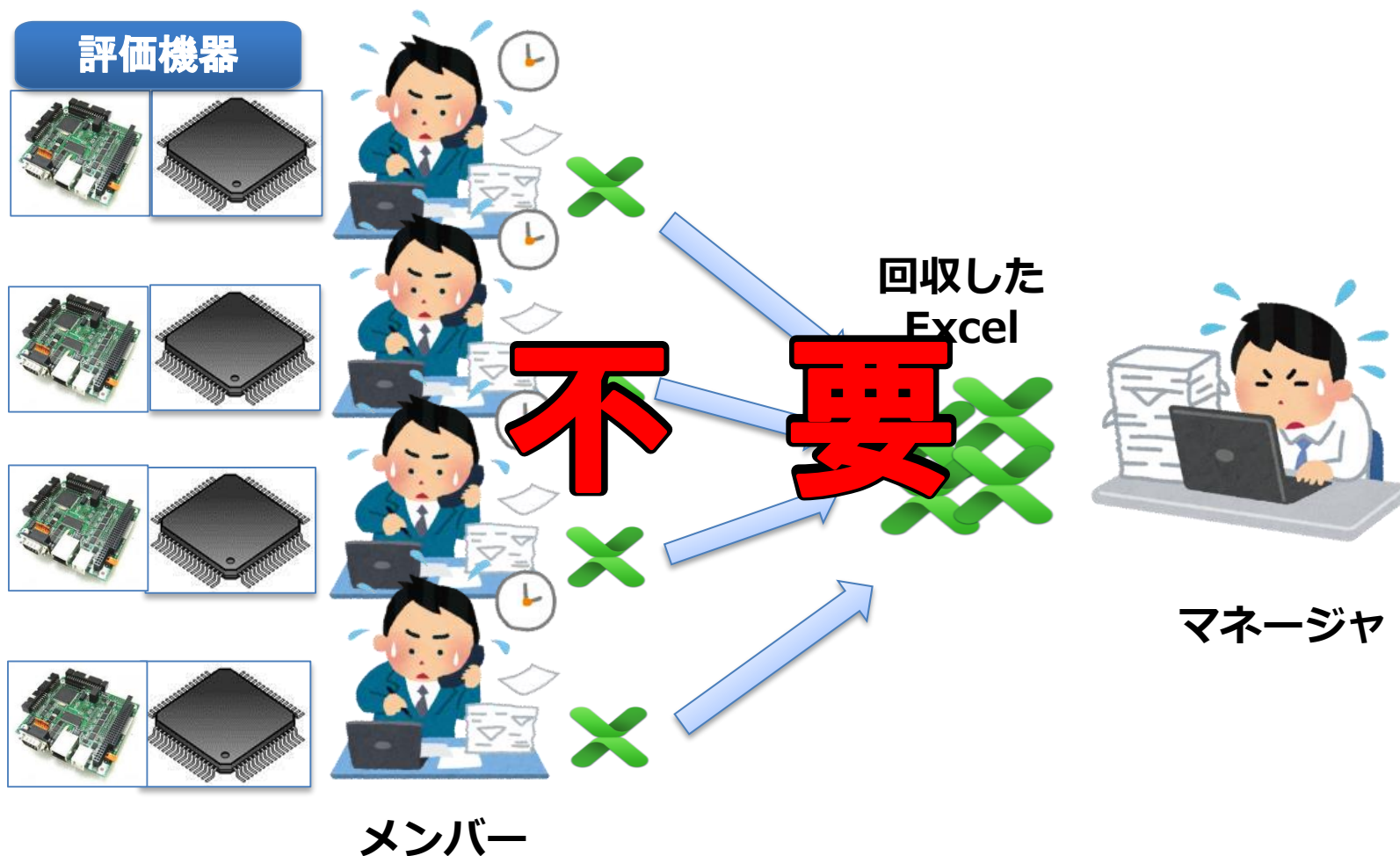
(4) 評価機器稼働状況表示機能

非効率な評価機器をすぐに特定できる



何曜日が稼働が低いかチェック

評価機器の使用状況のエクセル管理が不要になった



(5) スクリプト実行履歴追跡機能

自動スクリプト毎の履歴も楽に追える

TOP

テスト

障害

分析

稼働

管理

大項目

全て

中項目

全て

小項目

全て

テスト名

Android CTS

テスト内容

Package CTS: out/host/linux-x86/cts/android-cts.zip
cts make file: mydroid/build/core/tasks/cts.mk
run cts program: mydroid/out/host/linux-x86/bin/cts
test plans: mydroid/out/host/linux-x86/cts/android-cts/repository/plans
test packages: mydroid/out/host/linux-x86/cts/android-cts/repository/testcases
test results: mydroid/out/host/linux-x86/cts/android-cts/repository/results
CTS program settings value: mydroid/cts/tools/utis/host_config.xml

トータル

9回

PASS

88.89%

8回

FAIL

0%

0回

WARNING

11.11%

1回

実行履歴

Show 10 entries

Search:

SW ver.	HW ver.	OS ver.	開始日	終了日	結果
1.00	2.00	Gingerbread	2014-03-20 21:23:25.0	2014-03-20 21:23:25.0	PASS
1.01	2.00	Gingerbread	2016-01-14 01:48:25.0	2016-01-14 01:48:25.0	PASS
1.02	2.00	Gingerbread	2014-08-31 23:02:25.0	2014-08-31 23:02:25.0	PASS
1.02_custom	2.00	Jelly Bean	2015-11-12 09:13:25.0	2015-11-12 09:13:25.0	PASS
1.03	2.00	Jelly Bean	2015-11-18 08:28:25.0	2015-11-18 08:28:25.0	PASS
2.01	3.00	Jelly Bean	2015-11-11 07:56:25.0	2015-11-11 07:56:25.0	PASS
2.02	3.01	KitKat	2014-10-07 01:57:25.0	2014-10-07 01:57:25.0	PASS
3.00	4.00	KitKat	2014-06-05 16:29:25.0	2014-06-05 16:29:25.0	WARNING
3.01	4.01	Lollipop	2014-11-13 15:58:25.0	2014-11-13 15:58:25.0	PASS

Showing 1 to 9 of 9 entries

Previous 1 Next

(5) スクリプト実行履歴追跡機能

自動スクリプト毎の履歴も楽に追える

TOP

テスト

障害

分析

稼働

管理

大項目

全て

中項目

全て

小項目

全て

テスト名

Android CTS

テスト内容

Package CTS: out/host/linux-x86/cts/android-cts.zip
cts make file: mydroid/build/core/tasks/cts.mk
run cts program: mydroid/out/host/linux-x86/bin/cts
test plans: mydroid/out/host/linux-x86/cts/android-cts/repository/plans
test packages: mydroid/out/host/linux-x86/cts/android-cts/repository/testcases
test results: mydroid/out/host/linux-x86/cts/android-cts/repository/results
CTS program settings value: mydroid/cts/tools/utis/host_config.xml

トータル

9回

PASS

88.89%

8回

FAIL

0%

0回

WARNING

11.11%

1回

実行履歴

Show 10 entries

Search:

SW ver.	HW ver.	OS ver.	開始日	終了日	結果
1.00	2.00	Gingerbread	2014-03-20 21:23:25.0	2014-03-20 21:23:25.0	PASS
1.01	2.00	Gingerbread	2016-01-14 01:48:25.0	2016-01-14 01:48:25.0	PASS
1.02	2.00	Gingerbread	2014-08-31 23:02:25.0	2014-08-31 23:02:25.0	PASS
1.02_custom	2.00	Jelly Bean	2015-11-12 09:13:25.0	2015-11-12 09:13:25.0	PASS
1.03	2.00	Jelly Bean	2015-11-18 08:28:25.0	2015-11-18 08:28:25.0	PASS
2.01	3.00	Jelly Bean	2015-11-11 07:56:25.0	2015-11-11 07:56:25.0	PASS
2.02	3.01	KitKat	2014-10-07 01:57:25.0	2014-10-07 01:57:25.0	PASS
3.00	4.00	KitKat	2014-06-05 16:29:25.0	2014-06-05 16:29:25.0	WARNING
3.01	4.01	Lollipop	2014-11-13 15:58:25.0	2014-11-13 15:58:25.0	PASS

Showing 1 to 9 of 9 entries

Previous 1 Next

■ 実行履歴が表示される

- ・ソフトウェアバージョン
- ・ハードウェアバージョン
- ・開始時間
- ・終了時間
- ・結果

(5) スクリプト実行履歴追跡機能

自動スクリプト毎の履歴も楽に追える

TOP

テスト

障害

分析

稼働

管理

大項目

全て

中項目

全て

小項目

全て

テスト名

Android CTS

テスト内容

Package CTS: out/host/linux-x86/cts/android-cts.zip
cts make file: mydroid/build/core/tasks/cts.mk
run cts program: mydroid/out/host/linux-x86/bin/cts
test plans: mydroid/out/host/linux-x86/cts/android-cts/repository/plans
test packages: mydroid/out/host/linux-x86/cts/android-cts/repository/testcases
test results: mydroid/out/host/linux-x86/cts/android-cts/repository/results
CTS program settings value: mydroid/cts/tools/utis/host_config.xml

トータル

9回

PASS

88.89%

8回

FAIL

0%

0回

WARNING

11.11%

1回

実行履歴

Search:

Show 10 entries

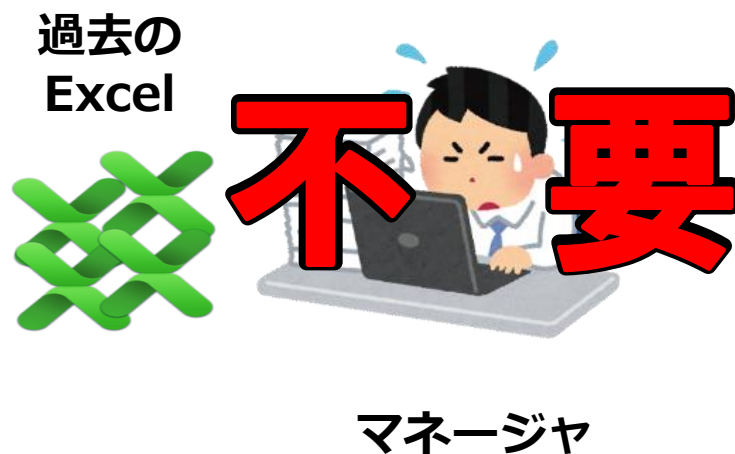
SW ver.	HW ver.	OS ver.	開始日	終了日	結果
1.00	2.00	Gingerbread	2014-03-20 21:23:25.0	2014-03-20 21:23:25.0	PASS
1.01	2.00	Gingerbread	2016-01-14 01:48:25.0	2016-01-14 01:48:25.0	PASS
1.02	2.00	Gingerbread	2014-08-31 23:02:25.0	2014-08-31 23:02:25.0	PASS
1.02_custom	2.00	Jelly Bean	2015-11-12 09:13:25.0	2015-11-12 09:13:25.0	PASS
1.03	2.00	Jelly Bean	2015-11-18 08:28:25.0	2015-11-18 08:28:25.0	PASS
2.01	3.00	Jelly Bean	2015-11-11 07:56:25.0	2015-11-11 07:56:25.0	PASS
2.02	3.01	KitKat	2014-10-07 01:57:25.0	2014-10-07 01:57:25.0	PASS
3.00	4.00	KitKat	2014-06-05 16:29:25.0	2014-06-05 16:29:25.0	WARNING
3.01	4.01	Lollipop	2014-11-13 15:58:25.0	2014-11-13 15:58:25.0	PASS

Showing 1 to 9 of 9 entries

Previous 1 Next

■ テスト実行のサマリ
トータルの実行回数
PASS率
FAIL率
WARNING率

過去のエクセルから、テスト結果を探す手間が不要になった



① あらゆるものをつなげた

主要5機能によって、事務作業が大幅に減少した

No	機能名	機能説明	効果
1	トップ画面 (テスト結果サマリー機能)	・ テスト結果のサマリー ・ 最新バージョンのテスト結果 ・ テストタイプ毎の結果一覧	エクセルによる回収や集計が 不要
2	テストタイプ別テスト結果 サマリー機能	・ テストタイプ別のテスト結果サマリー	エクセルによる回収や集計が 不要
3	フォロー対象チケット表示 機能	・ フォロー対象のチケットを表示 (更新頻度、クローズ期間、一次切り 分け期間)	Redmineによる一つ一つの 確認が不要
4	評価機器稼働状況表示 機能	評価機器の使用状況の表示	評価機器のエクセル管理が 不要
5	スクリプト実行履歴追跡 機能	スクリプトの実行履歴の表示	過去のエクセルからスクリプト の実行履歴の追跡が不要

どのような、SSTFを構築したのか？

実現したことは3つ。これにより生産性2倍を達成！

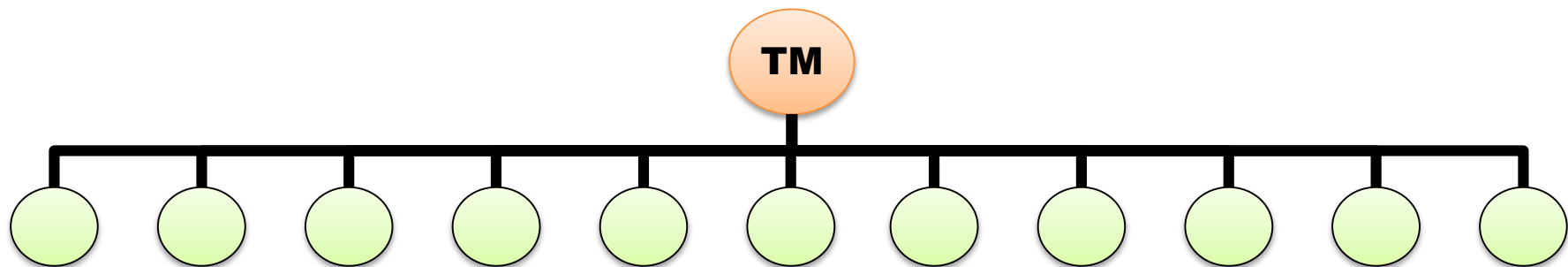
① あらゆるものをつなげた

② 組織体制を変えた

③ 進捗管理の方法を変えた

② 組織体制を変えた

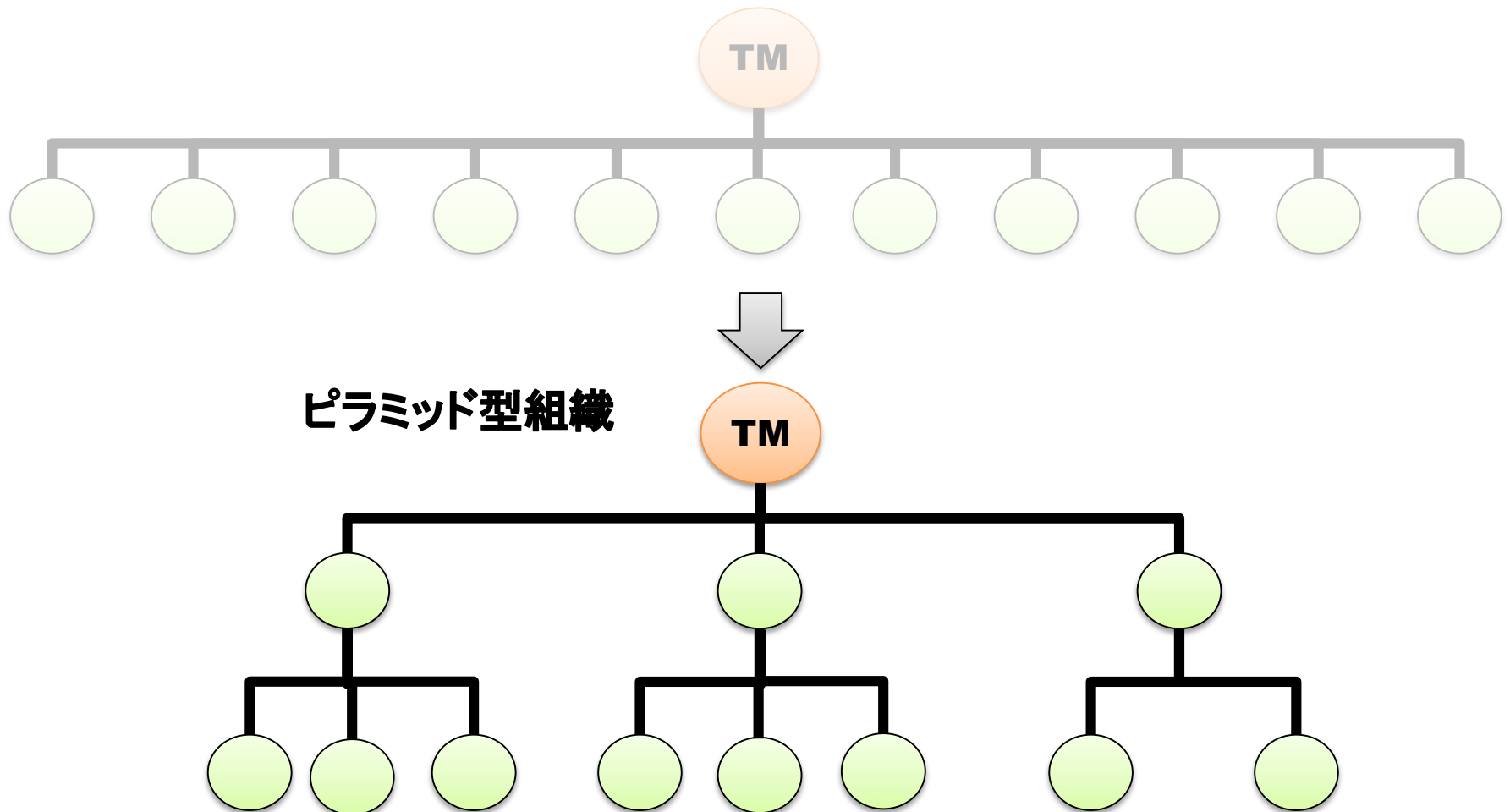
今までは、フラット組織でテストマネージャに負荷が集中



② 組織体制を変えた

今までは、フラット組織でテストマネージャに負荷が集中

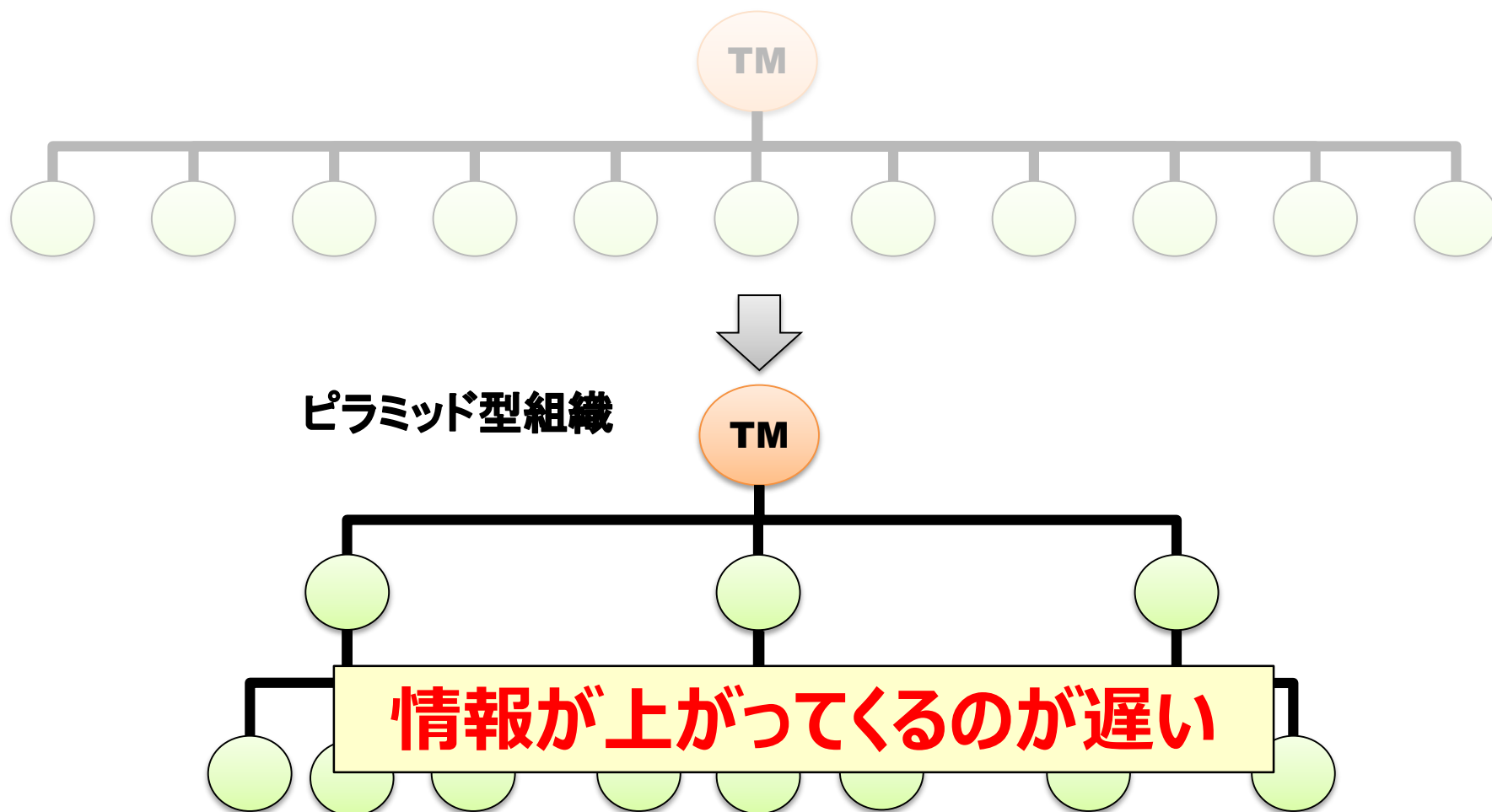
■ ダッシュボードがないと



② 組織体制を変えた

今までは、フラット組織でテストマネージャに負荷が集中

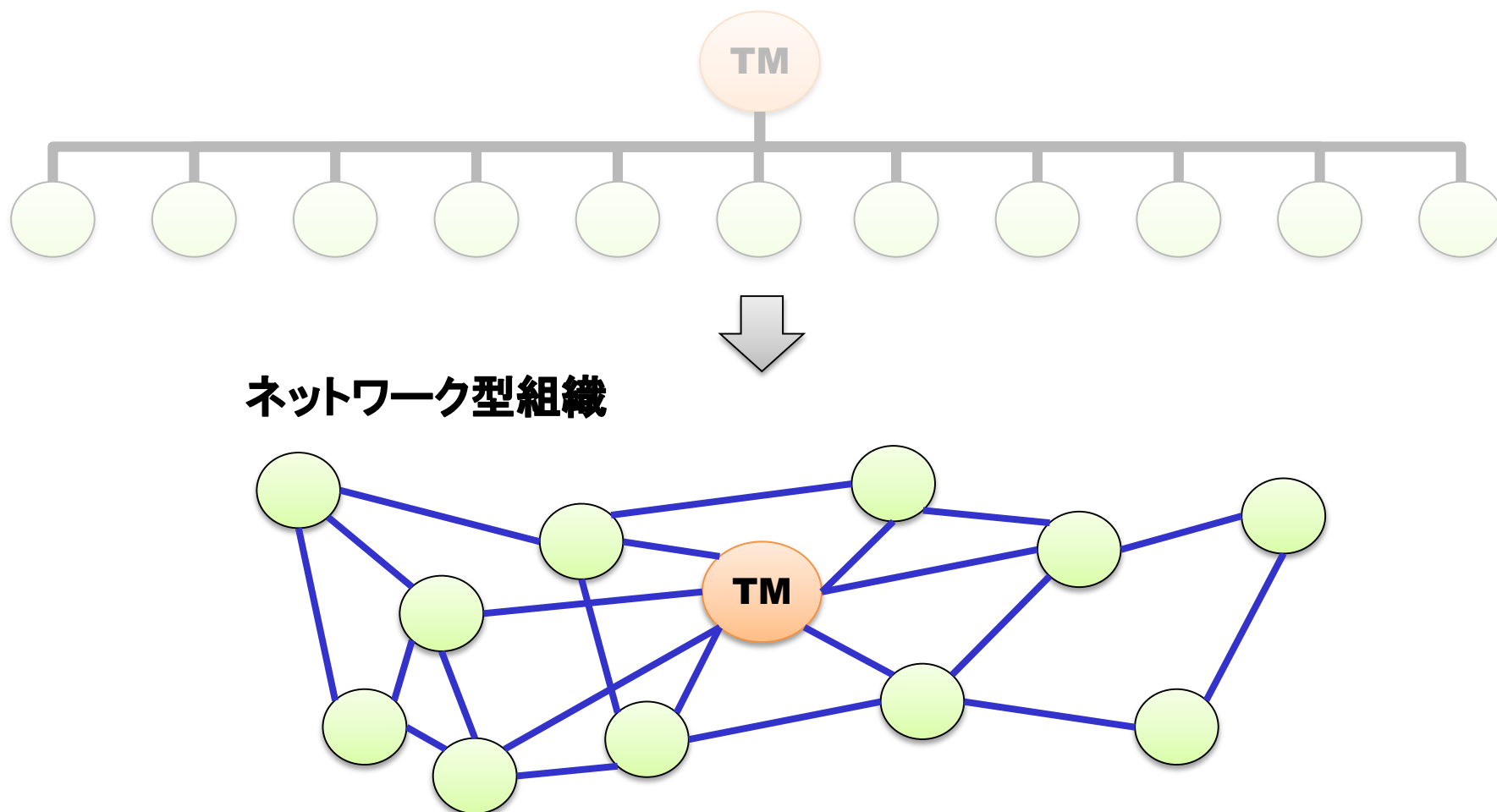
■ ダッシュボードがないと



② 組織体制を変えた

今までは、フラット組織でテストマネージャに負荷が集中

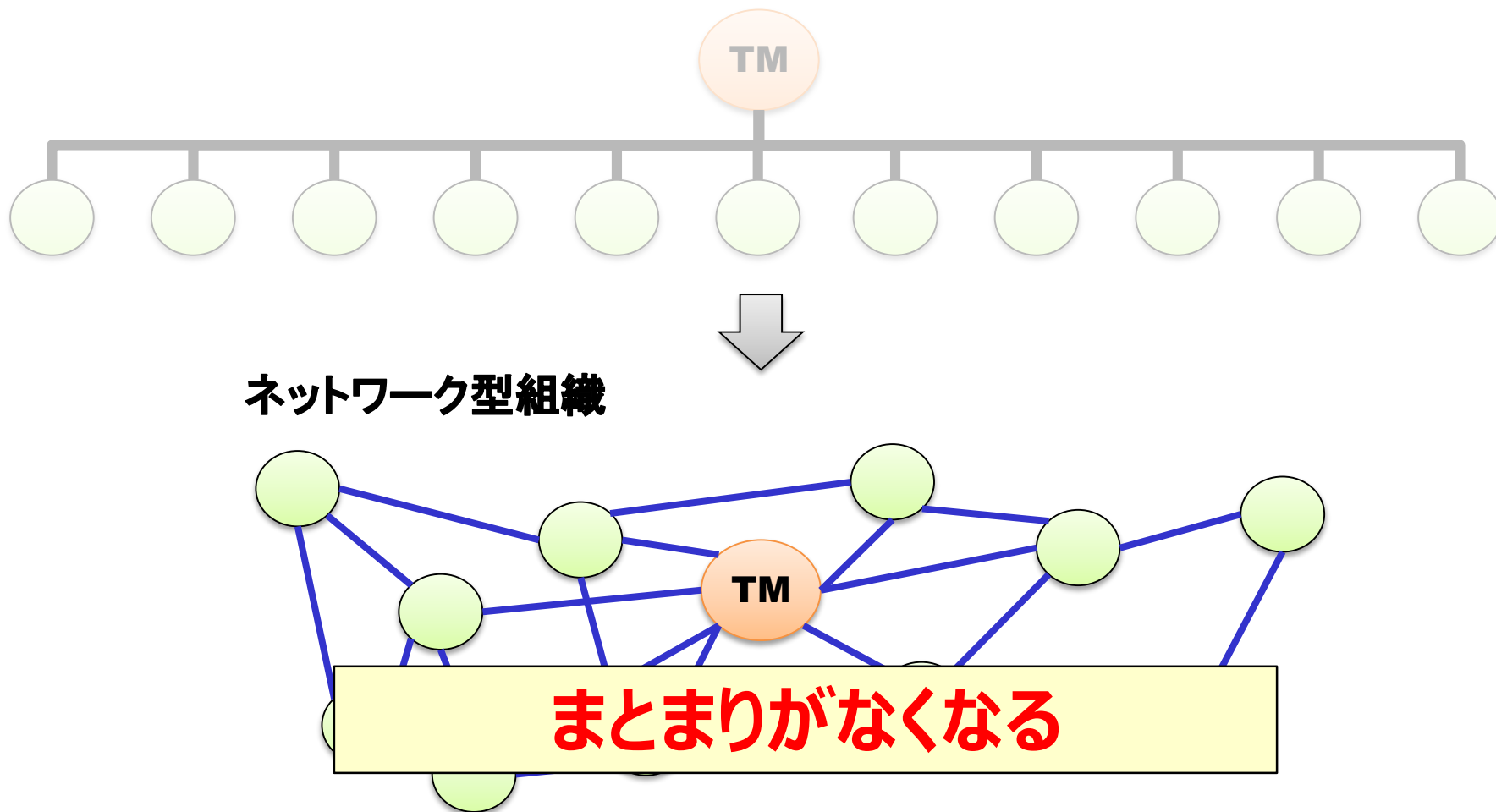
■ ダッシュボードがあると



② 組織体制を変えた

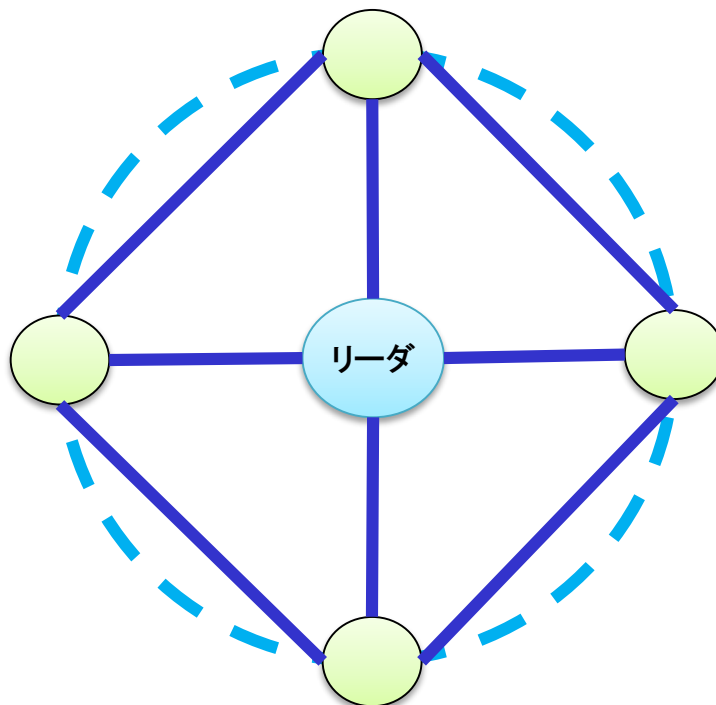
今までは、フラット組織でテストマネージャに負荷が集中

■ ダッシュボードがあると



② 組織体制を変えた

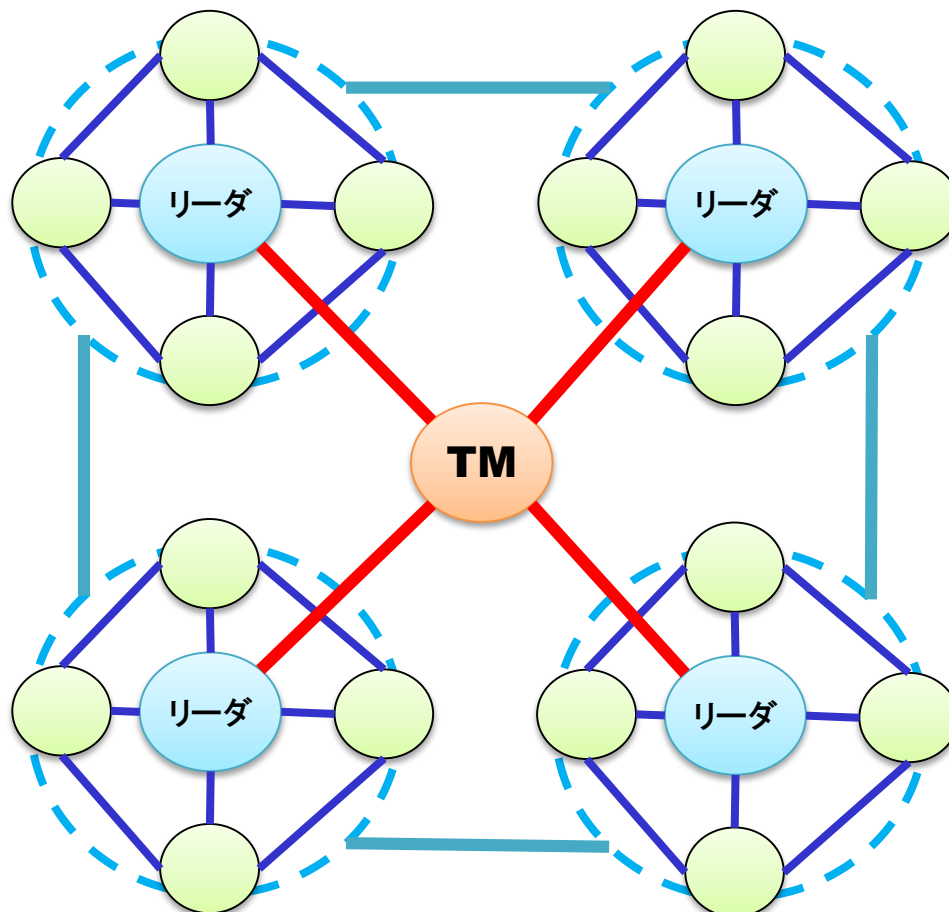
3～5人のユニットを作って、まとまりを作る



ユニットイメージ

② 組織体制を変えた

フラット組織から、ユニット・ネットワーク型組織に移行した



ユニット・ネットワーク型組織

どのような、SSTFを構築したのか？

実現したことは3つ。これにより生産性2倍を達成！

① あらゆるものをつなげた

② 組織体制を変えた

③ 進捗管理の方法を変えた

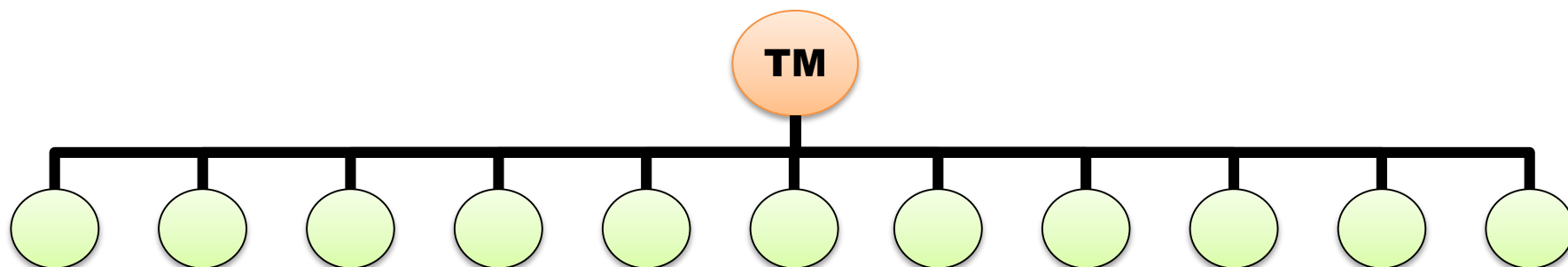
今までは、テストマネージャが全てを確認

■ 進捗会議（週1回）

- ◆ 全員参加（関係者が約20名参加）
- ◆ Openチケットを全部確認
- ◆ 会議時間は、2時間半

■ 日々の進捗確認

- ◆ 問題発生時に都度確認
- ◆ メールでの指示
- ◆ 問題は、テストマネージャがフォロー



テストマネージャは、上位判断が必要な問題に注力

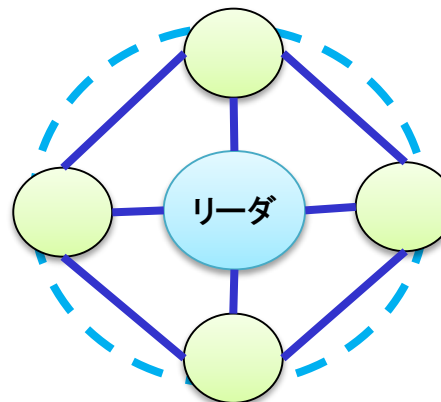
■ 進捗会議（週1～2回）

- ◆ リーダのみ参加（約5～6名参加）
- ◆ ダッシュボードで、フォロー対象になったもののみ確認
- ◆ 会議時間は、30分



■ 日々の進捗確認

- ◆ ユニット毎に「朝会」を実施
- ◆ ダッシュボードで都度進捗を確認
- ◆ ユニット毎に活発に議論
- ◆ 問題はお互いにフォロー



テストマネージャは、上位判断が必要な問題に注力

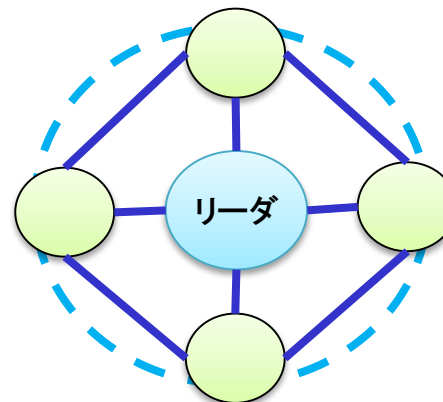
■ 進捗会議（週1～2回）

- ◆ リーダのみ参加（約5～6名参加）
- ◆ ダッシュボードで、フォロー対象になったもののみ確認
- ◆ 会議時間は、30分



■ 日々の進捗確認

- ◆ ユニット毎に「朝会」を実施
- ◆ ダッシュボードで都度進捗を確認
- ◆ ユニット毎に活発に議論
- ◆ 問題はお互いにフォロー



きめ細かなフォローと時間の短縮を達成

どのような、SSTFを構築したのか？

SSTFの構築によって、**生産性2倍を達成！**

- ① あらゆるものをつなげた
- ② 組織体制を変えた
- ③ 進捗管理の方法を変えた

【効果】

- ✓ 事務処理が大幅に減った
- ✓ 重要問題に注力できるようになった
- ✓ 問題が小さいうちに解決するようになった

【進捗】

予定よりも少し早い進捗で進行中
(1/3の工程終了時点)



Agenda

1. なぜ、SSTFを始めたのか？
2. どのような、SSTFを構築したのか？
- 3. SSTFの今後について**

3つのパラダイム・シフトが起きる

1. 生産能力向上による、ビジネスモデルのパラダイム・シフト
2. 情報の偏在解消による、構造のパラダイム・シフト
3. 自動化の促進による、人材のパラダイム・シフト

3つのパラダイム・シフトが起きる

1. 生産能力向上による、ビジネスモデルのパラダイム・シフト
2. 情報の偏在解消による、構造のパラダイム・シフト
3. 自動化の促進による、人材のパラダイム・シフト



**大量生産
(逐次実行)**



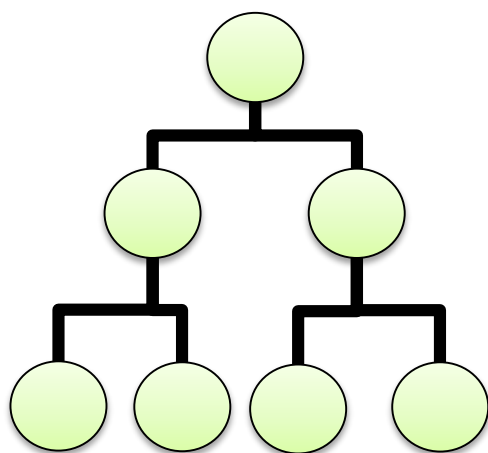
**少量多品種生産
(回帰テスト)**



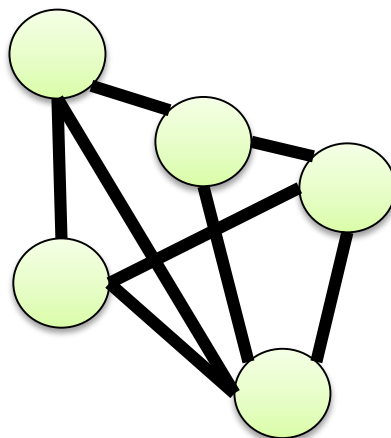
**マス・カスタマイゼーション
(フレキシブルテスト)**

3つのパラダイム・シフトが起きる

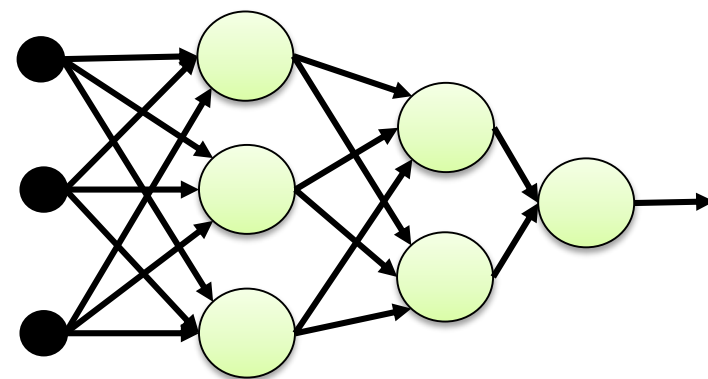
1. 生産能力向上による、ビジネスモデルのパラダイム・シフト
2. 情報の偏在解消による、組織構造のパラダイム・シフト
3. 自動化の促進による、人材のパラダイム・シフト



ピラミッド型



ネットワーク型



ニューラルネットワーク型

③ SSTFの今後について

3つのパラダイム・シフトが起きる

1. 生産能力向上による、ビジネスモデルのパラダイム・シフト
2. 情報の偏在解消による、構造のパラダイム・シフト
3. 自動化の促進による、人材のパラダイム・シフト



ブルーカラー
(テスト実行)



ホワイトカラー
(テスト実行管理)



ナレッジワーカー
(戦略、仕組みの創造)

最後に・・・

SSTFは顧客の課題と一緒に解決したいという思いから生まれた

SSTFは顧客の課題を一緒に解決したいという思いから生まれた

➤ 顧客のテストマネージャの激務をなんとかしたい

SSTFは顧客の課題と一緒に解決したいという思いから生まれた

- 顧客のテストマネージャの激務をなんとかしたい
- 経営上の難題にもチャレンジして解決したい

SSTFは顧客の課題と一緒に解決したいという思いから生まれた

- 顧客のテストマネージャの激務をなんとかしたい
- 経営上の難題にもチャレンジして解決したい

SHIFTは、ソフトウェアの「品質保証」を支援する会社

END

ソフトウェアテストといえば

SHIFT