

段階的なシステムテストによる IoT システム開発効率化

西村 治

パナソニック株式会社

nishimura.osm@jp.panasonic.com

要旨

様々な商品が連携し複雑化している当社の IoT システム開発において、従来のシステムテストでは、テスト終盤で不具合が収束せず、不具合対応に計画以上の時間を要し、開発納期に影響するという課題が発生している。

これを解決するために、段階的なシステムテストを導入し、テスト終盤の不具合を大幅に削減できた。この取り組みを本稿で紹介する。

1. はじめに

ネット家電やホームオートメーションなどから始まった家電のシステム化は、昨今 IoT システムと呼ばれるようになった[1][2]。この IoT システムは、構成する商品が多様化し、社内の異なる組織で開発された商品や他社商品を含む構成が一般的となっている。これら多様な組織で開発された商品は、品質が一定でないことに加え、不具合への対応も一律の手順では円滑に進まないケースが多くなってきた。

このため、従来のシステムテストでは非効率な面が顕在化し、新たなシステムテストの仕組みが必要となってきた。この新たな仕組みとして、段階的なシステムテストを導入することとした。この導入時の課題とその解決策を事例を踏まえて紹介する。

2. IoT システム開発の変化と課題

2.1. 背景

当社では、独立した複数の商品の組み合わせで価値を提供する商品を「システム商品」と呼んでいるが、従来のシステム商品開発は、システム商品を構成する全ての商品を単一の組織で開発し、その後別の組織が第三者としてシステムテストを実施するという体制をとるケースが多かった。

これに対して、IoT システムのように多様な複数組織が連携して行うシステム商品開発では、従来のブラックボックステスト[3]を行うシステムテストでは非効率な面が顕在化し、新たなシステムテストの仕組みが求められていた。

2.2. IoT システムの例

IoT システムの例として、当社のシステム商品である「スマート HEMS」を紹介する。スマート HEMS は、図 1 に示す通り、コントローラである AiSEG2 を中心に、社内の別組織で開発されたエアコンやレンジフード、他社商品の電動窓シャッターといった多様な商品が無線あるいは有線の通信で接続されており、センサー情報に基づいたエネルギー消費の見える化や、省エネのための家電の最適制御を実現している。また、AiSEG2 はインターネット経由でサーバと接続されており、スマートフォン向けの HEMS サービスを提供している[4]。

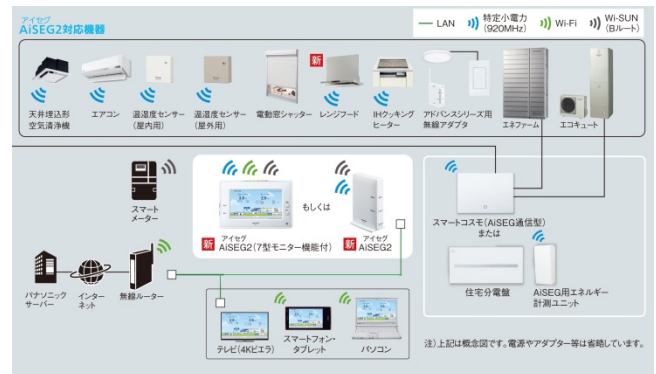


図 1 スマート HEMS システム構成図⁽¹⁾

2.3. システムテストにおける課題

IoT システムにおける従来のシステムテストでは、不具

¹ 商品カタログより一部抜粋。一部表示されていない対応商品があります。

合の収束に時間がかかるという課題が発生していた。

これは、テスト終盤になっても不具合が断続的に発生している状況である。テストの前半では、一見すればわかる画面表示に関する不具合が多いが、後半になると、これら画面表示に関する不具合は減少し、システム内部の振る舞いを通信ログやデバッグログで確認しないと原因がわからない不具合が見つかるようになる。

さらに、これらの不具合が商品間の通信に関するもの(以下、通信不具合と呼ぶ)である場合、複数の組織間で整合を図りながら、不具合対応を進める必要があるため、時間がかかることになる。

このため、テスト後半に通信不具合が多発すると、不具合対応に予想以上の時間を要するという問題が発生し、納期遅延のリスクが高まる。

3. 課題解決に向けて

3.1. 原因

2.3 節で述べた課題である不具合の収束に時間がかかることを解決するために原因を考える。

従来のシステムテストは、ユーザーインターフェースに対する機能ベースのブラックボックステストが中心であった。このブラックボックステストでシステム商品の内部の振る舞いである通信不具合を検出するには、限界があり、効率的に不具合を検出できないテストであった。

このため、商品間の通信において、詳細レベルを確認するテストが足りていないことが原因と考える。

3.2. 対策案

対策案として、前節で述べた足りていないテストを追加する。このテストは、システム商品の内部の振る舞いを確認するテストであるので「システム商品のホワイトボックステスト」と位置付け「システム商品結合テスト」と呼ぶ。一方、従来のシステムテストで行っていたブラックボックステストは「システム商品適格性確認テスト」と呼ぶ。

システム商品結合テストは、システム商品の内部の振る舞いを確認するテストであるので、方式設計書に基づき、商品間の通信の網羅的なテストを行う。

次に、このテストの実施フェーズは、

- 商品開発後に行うシステムテストに位置付ける。
- ブラックボックステストであるシステム商品適格性確認テストの前に実施する。

とする。この理由は以下のとおりである。

一般に、通信の詳細レベルのテストは、商品開発におけるプロトタイプ開発や通信の方式評価で行い、当社でも同様のフェーズで行っている。このフェーズでは、プロトタイプとしてエミュレータやシミュレータで確認しているが、これらの完成度が低い場合が多く、商品開発時では、接続に問題がなかったが、実機どうして接続すると接続できない、あるいは、1台は接続できるが、複数台の接続はできないといった問題が従来のシステム商品適格性確認テストの開始時で発生していたためである。

以上より、システムテストは、図2のような段階的なシステムテストとなる。

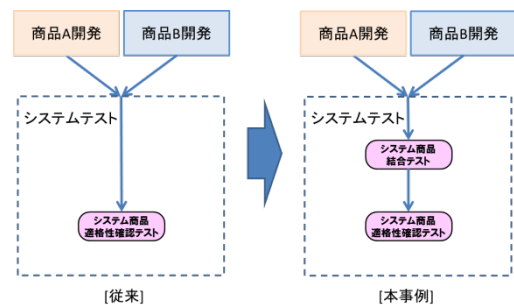


図2 システムテストの流れ

3.3. 対策案の導入に向けた課題

システム商品結合テストを導入するにあたり、主な課題を3つ説明する。

3.3.1. テスト担当者のスキルの問題

システム商品結合テスト(以下、本テストと呼ぶ)の実施にあたり、以下の観点で問題があった。

[通信の内容の確認スキル]

本テストは、ホワイトボックステストとして、通信の内容を確認するテストなので、従来のブラックボックステストで、ユーザーインターフェースの動作を確認してきたシステム商品適格性確認テストの担当者が行えるかという問題がある。具体的には、通信モニタのログから通信シーケンスや通信データなどを確認し、テスト結果の合否判断を行うことになる。このため、システム商品適格性確認テストの担当者は、テストの実施はできるが、テスト結果の確認ができない可能性があり、商品開発の担当者が本テストを実

施したほうが良いと考える。

[異なる組織と調整するスキル]

本テストの担当者は、仕様整合時や不具合対応時などは、商品毎に異なる組織の担当者との意思の疎通を行い、整合する必要がある。しかし、同一の組織内での商品開発のみを行ってきた担当者では、このような経験が少ないので、うまくいかないことが考えられる。さらに、自組織の利害関係を優先したりすると、不具合の修正方針を公平に判断できず、担当する商品の変更をできるだけ少なくする方針で対応したりすることが予想され、システム商品の品質が低下することが考えられる。このため組織間で公平な対応ができる第三者が対応することがより良いと考え、システム商品適格性確認テストの担当者が本テストを担当したほうが良いと考える。

3.3.2. テスト設計に方式設計書が必要

システム商品結合テストでは、通信の内容の詳細を確認する十分な方式設計書をもとにテスト設計を行う必要がある。この通信に関する方式設計書は、商品開発側で作成するものであるが、これまでは、システム商品適格性確認テストでは必要ないものであったため、その記載レベルが不明である。このため、本テストで十分な内容であることを確認する必要がある。

もしも、この設計書の記載レベルが低い場合、開発担当者に設計書に通信内容を追記してもらう必要がある。

この設計書の確認時期については、商品開発が始まり、方式設計フェーズで確認するのが効率的と考える。

3.3.3. ホワイトボックステストのテスト工数増大への対応

一般にホワイトボックステストを実施する場合、テスト項目が増大するという問題がある。しかし、今回は、不具合の早期検出を目的として、不足している通信内容の詳細な確認テストを追加することになっている。

このため、納期遅延リスクが増えない範囲で、テスト項目の増加によるテスト工数の増加を許容し、テスト実施を効率化する必要がある。

すなわち、テスト工数の増加によるテストカバレッジの向上とテスト実施時間の短縮による効率化を実現することを考える。

次に、システム商品結合テストで行う通信内容の詳細なレベルでの確認方法についてその考え方を説明する。

例えば、通信に異常が発生した場合のシステム商品の動作を確認するテストで説明する。

この場合、通信断を発生させてシステム商品の挙動を確認する。この挙動として、通信失敗がユーザに表示されるか、通信断が復旧すると、システム商品が正しく動作しているかを確認することになる。図 3 に示すような、商品 P が商品 A の 3 つの状態を取得する場合を考える。この場合、3 往復の通信シーケンスとなり、商品 P で 1 往復毎にタイムアウトが設定されているとする[5]。

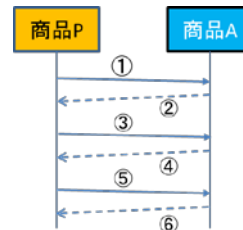


図 3 通信シーケンス

従来のシステム商品適格性確認テストでは、このようなテストは、テスト担当者が通信タイミングをみて、通信線を抜くといった手順で行うことになる。このため、確実に発生させることができず、複数回の実施が必要であり、時間のかかるテストとなっていた。また、①～⑥の全てのシーケンスで通信失敗を発生させることは非現実的で、①～⑥のいずれかのシーケンスで通信失敗となるテストとなっていた。このため、10 回実施しても通信失敗を 2, 3 回しか発生させることができないといった状況であった。

一方、システム商品結合テストでは、①～⑥の全てのシーケンスで確実に通信失敗させることを目指し、任意のシーケンスを破棄させるようなテストツールを作成し、確実に 1 回で指定したシーケンスを通信断できるようにする。この場合、6 回のテスト実施で全てのシーケンスで通信断を発生させることが可能となる。

以上のことから、システム商品結合テストのほうがカバレッジの向上と実施時間の短縮が可能となり、効率的なテスト実施を行い、不具合を確実に検出できるようになると考える。

4. 事例紹介

3.3 節の課題を解決し、システム商品結合テストを導入した事例について、当社のシステム商品であるスマート HEMS の開発プロジェクトを紹介する。

4.1. テスト担当者の決定

スマート HEMS の開発プロジェクト発足にあたり、システム商品結合テストのチーム体制を以下のように決定した。

- 商品開発のメンバに、通信に関するスキルがあり、かつシステムテストを経験したメンバがいたので、システム商品結合テストの担当者とした。
- システム商品適格性確認テストを担当する組織には、通信に関するスキルを持つメンバもいたが、他の開発プロジェクトを担当していたので、本プロジェクトにアサインできなかった。

次に、テスト担当者は、他の組織と調整を行う必要があるため、商品の開発には関わらず、システム商品結合テスト専任とした。テスト担当者を専任とすることで、開発プロジェクト発足時から参加可能となり、他の組織との関係を早くから築くことができた。さらに、商品開発を担当していないので、個別の商品に肩入れせずに、システム商品全体を見渡し、各商品の開発を行っている様々な組織と公平に調整ができた。

4.2. テスト設計に必要な方式設計書の充実

4.1 節のテスト担当者は、システム商品結合テストの設計に必要な方式設計書を商品開発担当者に開発開始時に作成依頼と提供依頼を行うようにした。さらに方式設計フェーズで方式設計書のデザインレビューにもレビューアとして参加し、内容の確認を行い、不足していると思われる内容については追加の依頼を行った。

実際に、追加した内容としては、通信の異常系に関するものが多かった。正常系の通信に関する方式設計は、問題が少なかった。この結果、方式設計書の内容が充実し、品質が向上した。

想定外の問題としては、システムテストが開始されると商品開発の担当者の一部が他の開発プロジェクトに異動した。この影響で、方式設計書のメンテナンスができなくなった。不具合発生時などで仕様変更となる場合、方式設計書を変更してもらう必要があるが、担当者がいない状況となっていた。このため、緊急対応として、システム商品結合テストの担当者が修正対応を行った。

4.3. テスト工数増大への対応

3.3.3 節で述べたように、テスト工数の増加によるテストカバレッジの向上のとテスト実施時間の短縮による効率化を実現する方法として、テストツールの活用を提案した。

今回、スマート HEMS で開発したテストツールを紹介する。このテストツールは、意図した通信異常を確実に発生させることができるツールである。

従来のシステムテストでは、通信異常に関するテストに多くの時間を要していたので、テストツールを開発し、効率化を実現した。

説明を簡単にするため、スマート HEMS における AiSEG2 とエアコンとの通信動作を考える。この AiSEG2 とエアコンは、特定小電力無線を使って通信を行う。

この場合、図 4 の(a)に示すように、AiSEG2 はエアコンに対して要求パケットを送信するとエアコンが応答を返す通信シーケンスを考える。この通信を図 4 の(b)に示すようにテストツールがパケットを破棄して切断することで AiSEG2 がタイムアウトし、通信異常となる。この時の AiSEG2 の動作を確認するテストが可能となる。

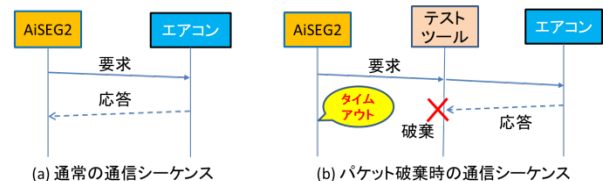


図 4 テストツールによる効率化

システム商品適格性確認テストとテストツールを活用した場合を順に表 1 と表 2 に示す。

表 1 システム商品適格性確認テストの場合

手法	実機 2 台の物理的な距離を離して、パケットの破棄または、遅延を実現する
発生回数 (カバレッジ)	要求か応答のいずれかのパケットを対象とし、破棄か遅延のいずれかを 1 回だけ発生
試行回数	発生するまで何回でも
所要時間	平均 1 時間

表 2 システム商品結合テストでテストツールを活用した場合

手法	実機2台とテストツールを使って机上で容易にパケットの破棄と遅延を実現する
発生回数 (カバレッジ)	要求と応答で各1回と破棄と遅延の各1回で2回×2回の計4回
試行回数	4回
所要時間	平均15分

以上のようにテストツールを活用することで、通信異常のテストは、テストカバレッジが4倍になり、かつ、所要時間を約1/4に削減できた。

このことから、通信異常のテストでは、約16倍の効率化を実現できた。

5. 成果と検証

システム商品結合テストを導入した成果と検証を行う。

5.1. 不具合の早期発見を実現

システム商品結合テストを実施することで、多くの通信不具合を検出することができた。通信不具合は、システム商品結合テスト全体の不具合の約3/4を占めていた。通信シーケンス、通信性能、通信競合に関するものが検出された。

また、システムテスト全体の通信不具合の約95%は、システム商品結合テストで検出できた。

このことから、通信不具合の多くをシステム商品結合テストで早期に検出でき、システム商品適格性確認テストの前に検出することができた。

5.2. システムテストの効率化

通信不具合の対応に関しても、方式設計書に基づく指摘であるため、公平な立場で商品開発側に対応をお願いできるようになり、組織の力関係や個別の商品の都合で対応方針が決まることが少なくなり、異なる組織の開発担当者との整合がうまくでき、不具合の収束も想定範囲内で推移し、計画通りにシステムテストを完了することができた。

今回、通信の内容確認ができるスキルを持つ開発経

験者を専任の担当者とすることで、開発の初期段階からテスト設計、テストツールを含むテスト環境の準備ができ、システム商品結合テストをスムーズに導入することができた。

さらに、システム商品適格性確認テストの開始をスムーズに始めることができるようになり、システム商品適格性確認テストの担当者は、システム商品結合テストの重要性を理解した。

5.3. 今後の課題

5.3.1. システム商品結合テストの定着化

今回の事例では、開発側にテスト経験のあるメンバをテスト担当者に決めたが、このようなスキルを持ったメンバは少ない。今後、システム商品結合テストを開発側が行った方式設計書に対する検証フェーズと考えた場合、定着化させるためには、開発側メンバに対するテストスキルの育成が必要となる。

また、方式設計書の品質が維持できるように、システムテストフェーズにおいても、開発側の担当者が必要であることを周知し、開発側で修正できる体制の維持が必要と考える。

5.3.2. システム商品結合テストの効率化

IoTシステムは、今後も規模が大きくなり、様々な商品が接続されていくと予想される。これに伴い、システム商品結合テストの工数も増加すると考えられる。このため、テスト自動化による効率的なテスト実施を実現し、テスト工数の増加を最小限に抑えたい。

6. おわりに

他社商品や異なる組織で開発が行われている様々な商品で構成されるIoTシステム開発において、テスト担当者の決定、通信に関する方式設計書の充実、テスト工数増大の対応を行い、システム商品結合テストを導入した。この結果、通信不具合の早期検出が可能となり、不具合の収束を早く行えるようになり、システムテストを効率良く進めることができた。

今後の課題としては、システム商品結合テストの定着化と、システム規模拡大に対応した効率化が必要と考える。

また、今回、システム商品結合テストを実施することで、通信品質が向上したと考えるので、今後の市場トラブルを未然に防ぐことができたことを確認したい。これは、商品発売後の初期トラブルの発生数を他の商品と比較することで確認したい。

謝辞

今回の経験論文の作成において、忙しい中、様々なアドバイスをいただいた当社の山本氏、藤村氏、植月氏に感謝いたします。

参考文献

- [1] wikipedia, “モノのインターネット”,
<https://ja.wikipedia.org/wiki/モノのインターネット>.
- [2] モバイルコンピューティング推進コンソーシアム, “IoT 技術テキスト”, リックテレコム (2016/10/28).
- [3] ISTQB テスト技術者資格制度 用語集, JSTQB,
<http://www.jstqb.jp/dl/JSTQB-glossary.V2.3.J02.pdf>.
- [4] スマート HEMS, Panasonic,
<http://www2.panasonic.biz/es/densetsu/aiseg/aiseg2/system.html>.
- [5] 竹政昭利 他, “かんたん UML 入門”, 技術評論社 (2013/6/7).
- [6] 独立行政法人 IPA, “【改訂版】組込みソフトウェア向け開発プロセスガイド”, 翔泳社(2007/11/19).