2007－SE－155 (11)
2007／3／22

クリティカルソフトウェアに対する IV&V：Independent Verification and Validation

加藤 淳† 星野 伸行† 片平 真史‡ 石濱 直樹‡ 宮本 祐子‡ 神武 直彦‡

† 有人宇宙システム株式会社 ‡ 宇宙航空研究開発機構

E-mail: {atsushi,nhoshino}@jamss.co.jp, katahira@computer.org,
{ishihama.naoki,miyamoto.yuko,kohtake.naohiko}@jaxa.jp

我々は，国際宇宙ステーションに代表される有人宇宙ミッション，社会と密接に係わる観測衛星，通信衛星などのクリティカルシステムに対し，ソフトウェア IV&V：Independent Verification and Validation（独立検証及び有効性確認）を実施している．ソフトウェア IV&V とは，発注元やメーカ等の開発組織に対し，組織的・予算的・技術的に独立した立場で，ソフトウェア開発におけるリスクを低減し，品質を向上させる活動である．本稿では，ソフトウェア IV&V とその評価手法について説明し，適用事例を紹介する．そして，我々が適用する評価手法に対して考察を行う．

IV&V：Independent Verification and Validation for Critical Software

Atsushi Kato †, Nobuyuki Hoshino †, Masafumi Katahira ‡,
Naoki Ishihama ‡, Yuko Miyamoto ‡, and Naohiko Kohtake ‡

†Japan Manned Space Systems Corporation.
‡Japan Aerospace Exploration Agency.

We perform the Software IV&V : Independent Verification and Validation for the Critical Software about the Manned Space Mission and Observation or Communication Satellites. The Software IV&V is activity that reduces risks and improves the quality in the software development and we take a stance of managerial, financial, and technical independence. In this paper, we show the Software IV&V and methods that we adopt and introduce the application experience. And we consider our methods.

1 はじめに

クリティカルソフトウェアとは，鉄道，航空などの交通システムや原子力発電所，医療機器など，社会生活との係わりが深く，そして何よりも信頼性，安全性が重視されるシステムのソフトウェアを指す．そして、国際宇宙ステーション（International Space Station：以降、ISS）に代表される有人宇宙ミッションや観測衛星，通信衛星など，宇宙開発における宇宙機ソフトウェアも，信頼性，安全性が要求されるクリティカルソフトウェアである．

宇宙機ソフトウェアについて，NASA[§]は，開発組織とは独立した機関による信頼性，安全性に関する評価，つまり，ソフトウェア IV&V：Independent Verification and Validation（独立検証及び有効性確認）の重要性を唱え，1993 年に宇宙機ソフトウェアの独立評価機関である NASA IV&V Facility[¶]を設立した．また，JAXA[‖] においても，1995 年より宇宙機ソフトウェアに対するソフトウェア IV&V が開始された．現在，有人宇宙システム株式会社（以降，JAMSS）[**]は，宇宙機ソフトウェアに対するソフトウェア IV&V の実施機関として活動している．

本稿では，我々が宇宙機ソフトウェアに対し実施するソフトウェア IV&V について，概念や評価手法を説明し，適用事例を紹介する．そして，我々が適用する評価手法に対して考察を行う．

2 宇宙機ソフトウェアとソフトウェア IV&V

本章では，クリティカルソフトウェアの 1 つである宇宙機ソフトウェア，及び，宇宙機ソフトウェアを対象としたソフトウェア IV&V について説明する．

[§]http://www.nasa.gov/
[¶]http://www.ivv.nasa.gov/
[‖]http://www.jaxa.jp/

[**]http://www.jamss.co.jp/

2.1 宇宙機ソフトウェア

2.1.1 分類

宇宙機ソフトウェアを分類すると，1. ISS 等の有人システムをはじめ，ロケット，人工衛星等の無人システムに搭載されるフライトソフトウェア（ファームウェアを含む），2. 宇宙機の地上管制システムで使用され，宇宙機との間のインタフェースを有する地上システムソフトウェア，3. フライトソフトウェアの開発・試験・維持等に使用される開発支援ソフトウェア，4. 地上システムに対し，宇宙機の軌道上運用を模擬する試験・検証ソフトウェアの 4 つに大別される（表 1）[1]．現在，我々は，1. 及び 2. を対象に，ソフトウェア IV&V を実施している．

次項では，宇宙機ソフトウェアとして，フライトソフトウェアの特徴を説明する．

2.1.2 特徴

宇宙機ソフトウェアが搭載される人工衛星などの宇宙機は，宇宙という特殊な環境で動作することを前提としている．ここで，宇宙機に搭載される電子機器の半導体素子が高エネルギーの放射線に晒された場合，メモリ等に記憶されるディジタル情報（0 ／ 1）が単発的に反転するなどのシングルイベント効果や，放射線の吸収量に比例して半導体素子が劣化し誤動作を引き起こすトータルドーズ効果が知られている．これらは，1 ビットの誤りを訂正し 2 ビットの誤りを検出する専用のハードウェアや，耐放射線性能の高い電子機器を搭載することで対処されているが，搭載される宇宙機ソフトウェアについても，宇宙環境においてそれらの現象が発生することを前提とした設計，実装を行う必要がある．

次に，ISS 等の有人システムは言うにおよばず，人工衛星などの無人システムにおいても，宇宙機は，高い信頼性を有する必要がある．従って，宇宙機には，宇宙環境下で使用される電子機器等の偶発的なハードウェア故障を想定した冗長化設計や，自動でハードウェアの故障を検出し，自律的に故障機能の回復を図る FDIR（Failure Detection，Isolation，and Recovery）機能を有するものもある．この場合，宇宙機ソフトウェアは，それらをサポートする設計，実装を行う場合がある．

また，打上げ後，搭載した宇宙機ソフトウェアに不具合が発見される場合や，宇宙機の軌道上運用において不測の事態が生じ，搭載するソフトウェアでは対処できない場合が想定される．そのような場合においてもミッションが達成できる様，無線通信により地上からソフトウェアそのものを更新する機能を有する宇宙機ソフトウェアも多い．

更に，宇宙機ソフトウェアは，リソースに厳しい制約がある場合も多く，限られた処理性能，メモリ容量の中でこれらの機能を実現する必要がある．

表 1: 宇宙機ソフトウェア分類

No.	名称
1	フライトソフトウェア
2	地上システムソフトウェア
3	フライトソフトウェア開発支援ソフトウェア
4	地上システム試験・検証ソフトウェア

2.2 ソフトウェア IV&V

2.2.1 概念

ソフトウェア IV&V とは，発注元やメーカ等の開発組織に対し，組織的・予算的・技術的に独立した組織がソフトウェアに起因するミッションの不達成リスクを低減し，品質を向上させるための活動である．開発組織から組織的・予算的に独立することにより，客観的な視点で，また，技術的に独立することにより，様々な評価手法や観点でソフトウェアを評価することが可能となる．図 1 に，ソフトウェア IV&V について示す．NASA IV&V Facility では，Verificaion を，"Are we building the product right?" つまり，『ソフトウェア開発の各開発フェーズで，定められた技術要求を満たしているか，また，開発プロセス及び管理要求に沿った正しい開発を行っているか評価すること』，また，Validation を，"Are we building the right product." つまり，『開発するソフトウェアがミッション及びユーザの要求を満足しているか評価すること』と定義している．

2.2.2 実施プロセス

宇宙機ソフトウェアを対象としたソフトウェア IV&V を実施するにあたり，網羅性という意味では，宇宙機に搭載されるすべてのソフトウェアを評価対象とすることが望ましい．しかし，限られた人員・時間・予算を有効に活用するために，JAMSS では，対象とするソフトウェア及び機能を選定し，選定した機能を中心にソフトウェア IV&V を実施している．図 2 に，ソフトウェア IV&V を実施する際の一例を挙げる．まず，仕様書のレビュー等により，ミッション及び宇宙機の喪失防止に係わる重要なコンポーネントを識別する．そして，識別したコンポーネントに対し，システム全体の FTA（Fault Tree Analysis：フォールト・ツリー解析）や FMEA（Failure Modes and Effects Analysis：故障モード影響解析）等を基に，評価対象とする重要な機能を識別する．

3 評価手法

宇宙機ソフトウェアに対するソフトウェア IV&V を実施するにあたり，評価対象となるソフトウェア及び機能の特徴や評価時点の開発フェーズ，評価可能な成果物，評価作業に関するコスト等の観点から，最適な評価手法を選定する．我々が実施するソフトウェア IV&V の活動は，要求仕様，設計仕様の評価のみならず，ソースコード評価や運用フロー

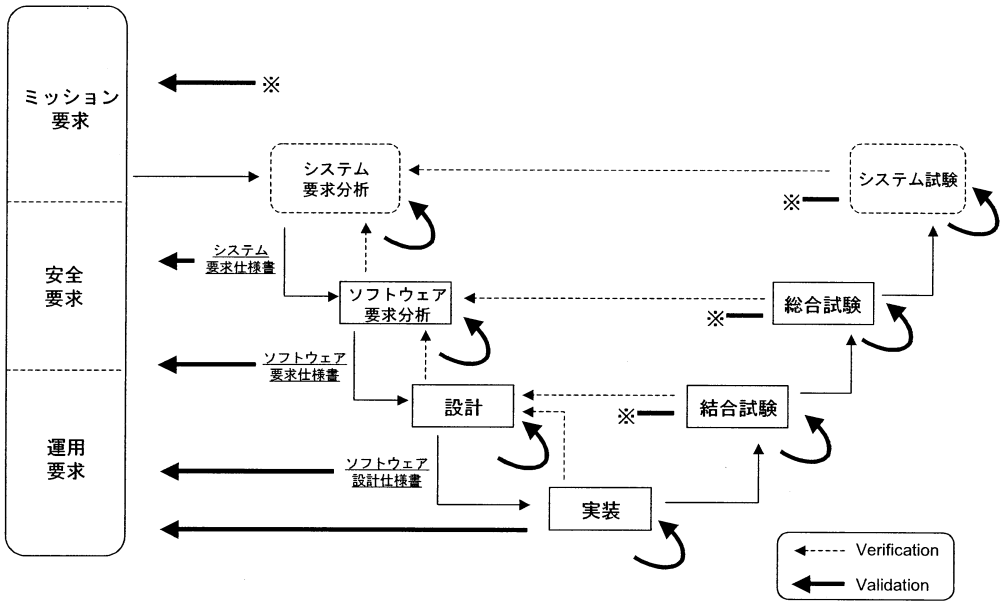

図 1: ソフトウェア IV&V

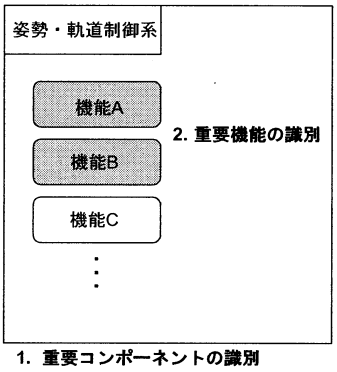

図 2: 評価対象機能の識別

評価など，ソフトウェアライフサイクルの全フェーズに及ぶ．宇宙機ソフトウェアについても，他分野のソフトウェアと同様に，上流フェーズにおける品質の作り込みが重要である．本章では，要求分析フェーズ，設計フェーズという上流フェーズを中心に，我々が適用する評価手法について説明する．要求分析フェーズ，設計フェーズにおける成果物としては，ソフトウェア要求仕様書，設計仕様書などがある．自然言語や処理フローといった形式で仕様が記述される仕様書に対し適用可能な評価手法として，我々は，文書レビュー及びモデル検査を実施する．

3.1 文書レビュー

文書レビューとは，システム全体の整合性等を踏まえながら，ソフトウェア要求仕様書，ソフトウェア設計仕様書等の成果物をレビューする作業である．本節では，我々が文書レビュー時に適用する評価手法として，チェックリスト評価を説明する．

3.1.1 チェックリスト評価

チェックリストとは，特定の仕様に対し，過去の経験則から確認すべき要点をまとめたリストである．文書レビューを実施するにあたり，ある特定の仕様については，チェックリストを用い重点的に評価することができる．宇宙機ソフトウェアについて，多くの場合，複数のソフトウェアがデータの送受信を行い，それぞれが協調して動作する．また，ハードウェアを制御するファームウェアも存在する．そのような場合，ソフトウェアが扱うデータの定義や処理タイミングなどの評価が重要となる．ソフトウェアのインタフェース等に関する評価として，我々は，ボイジャー・ガリレオチェックリスト [3] を適用する．また，衛星システム等のフライトソフトウェアの場合，一般に，ミッション及び宇宙機の喪失防止に係わるソフトウェアとして，姿勢・軌道制御系コンポーネントが識別される．そのような場合，姿勢・軌道制御の演算処理に関する評価として，我々は， 演算処理要求チェックリストを適用する．

表 2: ボイジャ・ガリレオチェックリスト（抜粋）

No.	評価内容
3	予期せぬ入力に対しても，必ず応答を返す設計となっているか.
4	規定外のデータに対する処理が規定されているか.
7	入力データに上限・下限が規定されているか.
9	応答を受信しない場合のソフトウェアの挙動は規定されているか.
11	ソフトウェアの起動前に受信したデータ，あるいは，ソフトウェアの終了後に受信したデータが残っていた場合の挙動は規定されているか.
12	コマンド受信後にレスポンスを返す場合，レスポンス受信機器が実際に動作しうる十分な Delay 時間をとっているか.

ボイジャー・ガリレオチェックリスト [3] Robyn R. Lutz は，宇宙機開発を経て得られた宇宙機ソフトウェアの Lesson & Learned をまとめ，ボイジャー：Voyager（1977 年打上げ）及びガリレオ：Galileo（1986 年打上げ）という NASA の惑星探査機の開発ケースを用いて，その有効性を検証した．我々は，この Lesson & Learned をボイジャ・ガリレオチェックリストと呼んでいる．表 2 に，ボイジャ・ガリレオチェックリストについて，16 項目のうち 6 項目を示す．

演算処理要求チェックリスト 宇宙機の姿勢制御など，数値計算のアルゴリズムを評価する場合，演算処理に関する Lesson & Learned をまとめた演算処理要求チェックリストを適用する．表 3 に，演算処理要求チェックリストについて，35 項目のうち 3 項目を示す．

3.2 モデル検査

我々は，宇宙機ソフトウェアに対するソフトウェア IV&V の評価手法として，形式的手法の 1 つであるモデル検査を採用している．要求分析フェーズの成果物であるソフトウェア要求仕様書などは，仕様に対する記述が不足している場合もある．そのような場合，我々は，曖昧な仕様でも比較的，モデル化が容易な SpecTRM[4] を適用する．また，設計フェーズの成果物であるソフトウェア設計仕様書などには，詳細な処理シーケンスが記述されている場合が多い．そのような場合，我々は，ソフトウェアの振る舞いに関するモデル化が容易な SPIN[5] を適用する．

表 3: 演算処理要求チェックリスト（抜粋）

評価対象	評価内容
変数	単位は明記されているか.
処理フロー	初期値設定が正しく行われているか.
アルゴリズム	アルゴリズム上，ゼロ割り算を回避する対策はとられているか.

3.2.1 SpecTRM[4]

SpecTRM は，米 Safeware Engineering Corporation.[††]が開発した解析対象ソフトウェアのモデリングやシミュレーション，安全解析を行うツールである．SpecTRM は，マサチューセッツ工科大学航空・宇宙工学部のソフトウェア安全工学に基づくもので，解析対象となるソフトウェアの要求仕様を状態遷移マトリクスでモデリングし，システムの状態遷移を評価することができる．図 3 に SpecTRM のサンプルを示す．

3.2.2 SPIN[5]

SPIN は 1980 年，米ベル研究所にて開発された．SPIN は状態変数と遷移条件の組み合わせでシステムをモデル化するのではなく，Promela という C 言語に似たプログラム処理記述に近い形でモデル化する．そして，Promela によるモデル記述を入力として，システムに求める性質を線形時相論理（LTL：Linear Temporal Logic）として与え，モデルが取り得る状態遷移を網羅的に生成して，自動的に検証する。また，SPIN は，プロセスを定義することができるため，複数の処理が並行して動作する分散・並列システムの検証に用いることができる．図 4 に SPIN のサンプルを示す．

3.3 その他の評価手法

宇宙機ソフトウェアに対するソフトウェア IV&V を実施するにあたり，我々は，実装工程及び試験工程という下流フェーズにおいても，ソースコード中のすべての分岐条件について，変数をランダムに変更し網羅的に解析を行うソースコード解析や，共有メモリに対しメモリアクセスに競合が無いかを網羅的に評価する共有メモリ競合解析などを実施している．また，システムの運用についても，運用フローにモデル検査を適用することで，フローの一貫性（複数の状態が同時に発生することがないこと）や完全性（未定義状態が存在しないこと）を評価している．

4 適用事例

宇宙機ソフトウェアに対し，我々が実施するソフトウェア IV&V は、開発組織が実施する各開発フェーズの成果物評価とは重複しない評価が多く，また，独立的な立場で実施することにより，開発者の思い込み等を排除することができる．

宇宙機ソフトウェアに対するソフトウェア IV&V の適用事例の一部として，表 4 に宇宙機ソフトウェアに対しソフトウェア IV&V を適用した開発フェーズを示す．我々は，これまでに多くの宇宙機ソフトウェアに対し，要求分析フェーズ及び設計フェーズ

[††]http://www.safeware-eng.com/

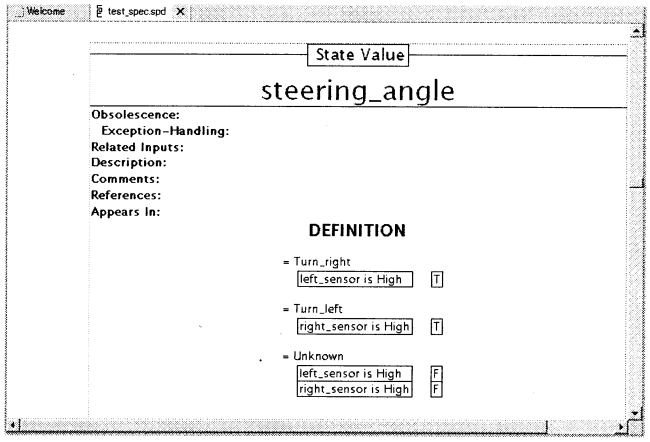

図 3: SpecTRM モデル（サンプル）

```
int RETURN_VALUE=0;
init {
        /* 変数値の範囲設定 */
        if
        :: skip -> message_type=STOP_COMMAND;
        :: skip -> message_type=99999;
        :: skip -> message_type=CLEAR_COMMAND;
        fi;
        run main();
}
proctype main() {
        /* エラー処理A */
        if
        :: (message_type==STOP_COMMAND || message_type==CLEAR_COMMAND) ->
            /* YES 処理 */
            ret=INVALID;
        :: else ->
            /* NO 処理 */
        fi;
        RETURN_VALUE=0;
        /* ** プログラムここまで ** */
}
```

図 4: SPIN モデル（サンプル）

表 4: 適用事例

宇宙機ソフトウェア	ソフトウェア IV&V 評価フェーズ			
	要求分析	設計	実装	試験
A	○	○		
B	○	○	○	○
C	○	○		
D	○	○	○	○
E				○
F	○	○	○	○

で，文書レビューやモデル検査を実施している．また，実装フェーズ及び試験フェーズにおいても，ソフトウェア IV&V を実施している．

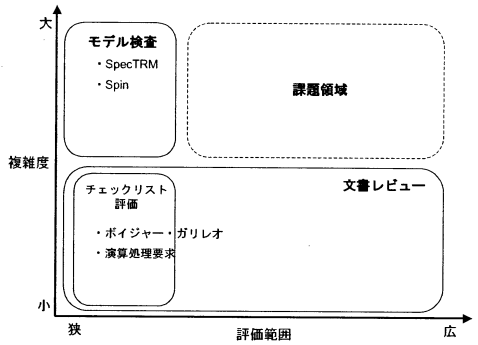

図 5: 評価手法における評価範囲と複雑度

5 考察

本章では，文書レビュー及びモデル検査を実施することで得られた知見を基に，ソフトウェア IV&V の評価手法について考察する．

5.1 文書レビューとモデル検査の併用

表 5 に，文書レビュー及びモデル検査の特徴をまとめる．まず，文書レビューの長所として，比較的短い期間で対象を評価することができる．一方，複雑な状態遷移や処理タイミングについては，評価が困難である．モデル検査の長所として，人手では検出が困難である複雑な処理を識別することができ，また，網羅的に検査を実施することができる．また，モデル検査を実施する場合に仕様書等を参照してモデル化を行うが，その過程の副産物として仕様の抜けなども確認できる．一方，短所として，モデルの作成及び検査結果の解析に多くの工数を要する．表 5 から，文書レビュー及びモデル検査は，それぞれ対象的な特徴を有していることがわかる．ソフトウェア IV&V を実施するにあたり，我々は，それぞれの特徴を活かした評価手法の組み合わせで，評価を実施することが重要である．

5.2 今後の課題

図 5 に，文書レビュー及びモデル検査について，それぞれの評価手法で評価可能な範囲／複雑度を示す．文書レビューを実施することにより，複雑な処理等については評価が困難であるものの，システム全体を通した整合性等を評価することができ，各チェックリストを適用することにより，特定の仕様に対し重点的に評価を実施することができる．また，モデル検査を実施することにより，評価範囲は限定されるものの，人手では困難である複雑な処理等を評価することができる．ソフトウェア IV&V を実施するにあたり，できる限り多くのソフトウェアを対象とし，それらを詳細に且つ網羅的に評価することが望ましい．しかし，モデル評価を考えた場合，現状，モデル化や検査結果の解析に要するコスト的な要因から，評価範囲を限定し，モデル検査を実施せざるを得えない．ここで 1 つの解決策として，モデル評価を実施するにあたり，モデルの作成を効率化することで評価範囲を広げる方法が考えられる．我々がモデル検査で用いる SPIN であるが，C 言語に似たプログラム処理記述に近い Promela を用いてモデル化する．今後，宇宙機ソフトウェアのソースコードから Promela への自動変換が可能なツールを整備し，それを実装フェーズや試験フェーズの初期段階においてソースコードに適用することで，モデル評価を実施する際のモデル化作業を効率化し，モデル評価における評価範囲を拡大することが期待される．

表 5: 評価手法の特徴

評価手法	長所	短所
文書評価	比較的短い期間で対象の評価が可能	複雑な状態遷移や処理タイミングに関する評価が困難
モデル検査	複雑な処理の評価，網羅的な検査が可能	モデルの作成／解析に多くの工数が必要

6 おわりに

本稿では，クリティカルソフトウェアの 1 つである宇宙機ソフトウェアに対するソフトウェア IV&V について，その概念や評価手法について説明し，適用事例を紹介した．そして，我々が適用する評価手法に対して考察を行った．

参考文献

[1] 狼，冨田，堀川，白木：”宇宙ステーションと支援技術”（東京，コロナ社，2004 年），p.218.

[2] The Aeronautics and Space Engineering Board National Research Council : "An Assessment of Space Shuttle Flight Software Development Processes"(USA: The NATIONAL ACADEMY PRESS, 1993), p.31-p.37.

[3] Robyn R. Lutz: "Targeting Safety-Related Errors During Software Requirement Analysis", Journal of Systems and Software, Vol.34 Issue.3(1996), p.223-230.

[4] Nancy G. Leveson, Jon D. Reese, Mats P.E.Heimdahl: "SpecTRM: A CAD System for Digital Automation", Digital Avionics Systems Conference Proceedings. 17th DASC. The AIAA/IEEE/SAE, vol.1(1998), B52/1-B52/8.

[5] Addison Wesly: "The Spin Model Checker: Primer and Reference Manual"(Canada: Pearson Education, 2003), p.608.