

K7-86

**学生による超小型人工衛星開発における技術継承プロセスの提案**  
 ～ALOS-2 小型相乗り副衛星「SPROUT」開発プロジェクトにおけるシステムズエンジニアリング～  
**Proposal of Knowledge Transfer Process on nano-satellite development by student**  
 ～Systems Engineering on "SPROUT" development project of the piggy-back satellite of ALOS-2～

○角川雄基<sup>1</sup>, 山崎政彦<sup>2</sup>, 宮崎康行<sup>2</sup>\* Yuki Tsunokawa<sup>1</sup>, Masahiko Yamazaki<sup>2</sup>, Yasuyuki Miyazaki<sup>2</sup>

Abstract: Nihon University department of Aerospace Engineering Miyazaki-laboratory launched the "SEEDS- II" for the purpose of bus technology demonstration of nano-satellite in 2008, and "SPROUT" for the purpose of early demonstration of engineering technology in 2014. In order to maintain the development system in laboratory of turnover of people hard, we have been doing the approach to the Issue of Knowledge Transfer, by conducting the "freshmen training program" the aim of acquisition of basic skill in the development of nano-satellite, accumulating development information as the "Design Report" and others. In this presentation, we propose Knowledge Transfer Process of newly constructed to lead successful nano-satellite development with a basis of Systems Engineering, and we will explain how to use and its usefulness as based on the "SPROUT" satellite development

## 1. はじめに

現在, 世界中の多くの大学・研究機関等で超小型衛星の開発が盛んに行われている. 我々, 日本大学理工学部航空宇宙工学科宮崎研究室でも, 2008 年に超小型衛星のバス技術実証を目的として「SEEDS-II」衛星を打ち上げ, 2014 年には工学技術の早期実証を目的とした「SPROUT」衛星を打ち上げた.

超小型衛星の開発を行っている多くの大学研究室のような教育・研究機関が抱える問題点として, 人の入れ替わりが激しく, プロジェクトの重要な情報の暗黙知化や技術レベル低下など技術継承問題が挙げられる<sup>[1]</sup>. 一般に, 宇宙システムはたとえ規模が小さくても, 複雑度, 統合度が非常に高度になるために, その全体像を把握すること・継承することは非常に困難とされている<sup>[3]</sup>. これに対し, 本研究室では衛星開発経験者が未経験者へ開発の考え方の基礎を刷り込むことを目的とした「新入生教育プログラム」<sup>[2]</sup>の実施や, 開発情報を「設計報告書」として保存する等のアプローチを行ってきた. しかし, 現状では, 多数のコンポーネントを有し, かつ, 相互関係を伴う超小型衛星システムに対して十分な技術継承プロセスが確立されておらず, 暗黙知のまま継承されない知識・技術が多く存在するといった状況にある. 例えば, SPROUT 衛星システムに課せられた要求は「SPROUT 設計書」においてある程度, 形式知化されているが, サブシステムレベルにおいて各 SPROUT 衛星サブシステムに課せられた要求は, 当事者間の合意によってのみ決定されているため, ほとんどが暗黙知化している.

本発表では, 新たに構築したシステムズエンジニアリングを基とした超小型衛星開発を成功に導くための, 技術継承プロセスを提案し, SPROUT 衛星開発を例にとりその有用性と利用方法を説明する.

## 2. 現状の技術継承プロセスの問題点と提案する技術継承プロセス

暗黙知を次の世代に伝えるプロセスとして主に 2 通りある<sup>[4]</sup>.

- ① 暗黙知→暗黙知 (有識者の暗黙知を OJT (On the Job Training), ディスカッション, レクチャーなどによって他者へ伝達するプロセス)
- ② 暗黙知→形式知→暗黙知 (有識者の暗黙知を文書化, モデル化するなどして, 他者がそこから知識を得るプロセス)

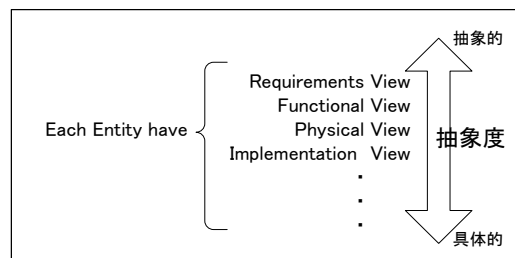


図 1. View と抽象度

SPROUT 開発チームでは, ①, ②を実現する努力を行ってはいるが, 十分に SPROUT の特徴を伝達できていないといえず, 特に, 粒度の定義, Entity(境界, 構成要素)の定義, 抽象度の高い View<sup>[5]</sup> (要求や機能アーキテクチャ) に関するモデル, それらのトレーサビリティが開発担当者の暗黙知となってしまうことが多い. トレーサビリティとは,

1: 日大理工・院 (前)・航宇 2: 日大理工・教員・航宇

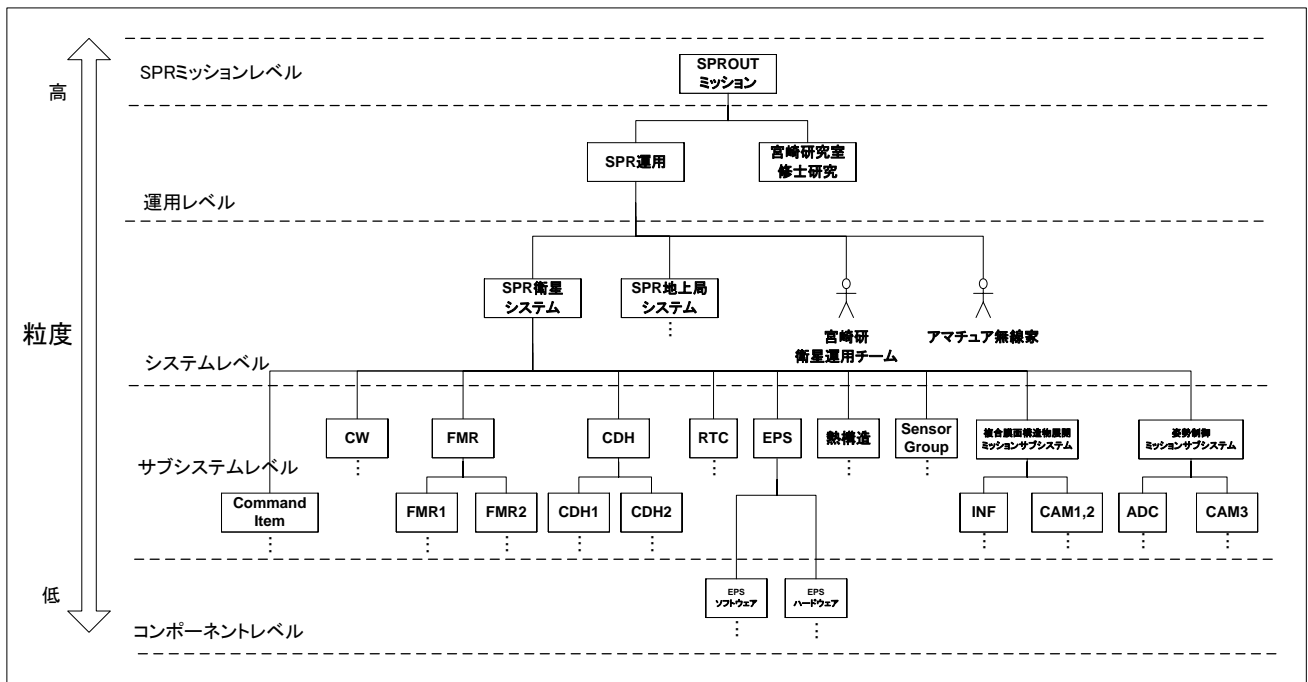


図 2. SPROUT ミッションの粒度の定義と Entity(境界、構成要素)

相互関係をもつ Entity やモデル同士がお互いを参照できることである。トレーサビリティが適切に継承されないと、思わぬところで不整合が発生してしまいかねない。このことから開発担当者の暗黙知となっている SPROUT 衛星開発のアーキテクチャを適切な粒度と抽象度に切り分けて後世に伝えるための新たなプロセスを構築する必要がある。

そのため、現在、SPROUT 開発チームでは、SPROUT のそれぞれの粒度における Entity の定義、それぞれの抽象度に応じたアーキテクチャとそれらのトレーサビリティを網羅した SPROUT ミッションのアーキテクチャ記述<sup>[5]</sup> (設計書) の開発を行っている。これによって②のプロセスを十分に満たすことだけでなく、アーキテクチャ記述の実現が①のプロセスを指南することが期待できる。

図 2 に「SPROUT ミッションの粒度の定義と Entity(境界、構成要素)」を示す。この図では一つ一つのブロックが SPROUT ミッションアーキテクチャの Entity(境界、構成要素)を示している。ここでは EPS サブシステムのみコンポーネントレベルにまで分解したものを示したがアーキテクチャ記述では他のエンティティについても同様に分解をおこなっている。アーキテクチャ記述では技術継承がうまく行われるように抜けもれなく適切な粒度でエンティティを分解し、それぞれのエンティティに関して、図 1 で例を挙げた View と View に関するモデルを抜けもれなく記述していく。

### 3. 今後の展望

実際の SPROUT 開発の事例を考慮し、SPROUT のアーキテクチャ記述 (設計書) を基として、実際の衛星開発と並行して実現できる新たな技術継承プロセスを構築していく。そして、実際に行ってきた SPROUT 衛星開発のプロセスと提案する技術継承プロセスを比較することで、その有用性と利用方法の検討を行っていく予定である。

### 4. 参考文献

- [1] 石井亮介, 佐原宏典:「超小型衛星のミッション正否に基づく現状把握と将来予測」, 2012
- [2] 井上祥子, 藤井大輔, 他:「超小型衛星開発の技術継承における新入生教育の試み」, UNISEC Space Takumi J, Vol, No.3, pp.40-62, 2012
- [3] UNISEC:「UNISEC Space Takumi Journal for Practical Study of Problem Finding and Solving in Space Systems」, Vol.1, 31 March 2011
- [4] Hideki Uchikawa, Toshiyuki Yasui, Seiko Shirasaka:「Hierarchical Knowledge Transfer and Creation Model」, APCOSE2013
- [5] Software Engineering Standards Committee of the IEEE Computer Society:「IEEE Recommended Practice for Architectural Description of Software-Intensive Systems」, IEEE Std 1471-2000