

# 開発アセットの構築と計画的な利用を促進する規律

細野 繁<sup>†</sup>

サービス開発アセットの構築と計画的な利用スキームを開発現場に定着させるため、開発成果物からのアセット設計、適用計画、構築、利用までプラクティスと、その実践方法を述べる。

## Disciplines for Asset Development and Reuse in Service Production

Shigeru Hosono NEC Corporation

To establish asset development and reuse in service production firmly, this paper discusses disciplined manners of planning, building assets and adapting them to a number of service development projects.

### 1. はじめに

IaaS や PaaS 等、クラウドサービスの普及につれ、そのサービス・インタフェースを利用した Web サービスが構築し易くなった。このような環境には、必要な機能の段階的な実装と確認を繰り返し行うアジャイル開発方法が適しており、無駄な機能開発を排した開発が期待されるようになった。更なる開発の効率化には、過去の開発知識の活用が有効である。しかし、アジャイル開発方法に移行しても、アーキテクチャの根拠となったビジネスルールや制約などは、依然としてプログラムや開発文書などの成果物にデータとして残され難く、過去の開発知識は必ずしも効果的に活用されていない。また、活用できなかった理由もデータとして残され難く、開発の効率化は頭打ちとなっている。そこで、SEMAT (Software Engineering Method and Theory) [1]に着目し、開発アセットの構築から、次の開発で活用するまでに、規律(ディシプリン)を与える方法について述べる。

### 2. アセット構築・活用の課題

著者らは、開発過程の成果物や実績値を利用するために、サービス生産プロセスを資産とする、アセット化方法を提案している[2]。本手法は、開発工程毎にアセット化の対象を設計モデルやソフトウェアコンポーネントを独立してアセットとして扱うのではなく、最小構成の開発～運用(DevOps)プロセスを単位とする。また、このアセットには、各工程での、設計モデルの作成や利用に関わる人手の協調作業を定量データとして含める。このアセットをパタン化したことで、このアセットを類似したサービス開発の成果物や進捗管理のベースとして利用し、個々の顧客の要件の違いに合わせた、カスタマイズ開

発を支援する。また、モノの生産プロセス管理と同様に、人手の関与が多く含まれるサービス生産プロセスの定量的な生産管理・分析を支援する。しかし、このアセットの効用は、適用先のサービス開発の要求仕様との差異が、ベースのカスタマイズ・インタフェースに収まる場合に限られている。

この差異は、要件や仕様の詳細が確定されないまま、繰返し開発を行いながらターゲットに近づけるアジャイル開発では大きくなりがちで、アセットの適合性が低下する。要件や仕様に不確定性(値の範囲)が生じ、過去の開発情報の適用選択肢が多くなるため、最も適切なアセットを選び難くなるからである。

そこで、アセットの設計、適用計画、構築、利用を確実に行うために、達成すべき状態群を定義する。この状態の達成度を開発者に常に意識させることで、開発プロセスにアセット構築・活用のディシプリンを与える。

### 3. アセット活用ディシプリン

#### 3.1. SEMAT

SEMAT は、理論と原理とベストプラクティスに基づくソフトウェア開発方式である。SEMAT は、開発チームが必ず考慮する必要がある関心領域の Customer, Solution, Endeavor から成るカーネルを定義している。各領域では、アルファ(Alpha; Abstract-Level Progress Health Attribute)によって、活動の目標を与えている。基本アルファとして、Customer 領域には Opportunity と Stakeholders を、Solution 領域には Requirements と Software System を、Endeavor 領域には Team, Work と Way of Working を定義している。これらのアルファには、それぞれ到達ゴールを示す状態とその構成項目が定義される。アジャイル開発での繰返される試行過程のように、開発工程全体での位置を把握しにくい場合に、

<sup>†</sup>NEC 情報・ナレッジ研究所

有効である。開発チームは、アルファに基づくアルファカードを並べて、各達成項目から開発進捗を位置取りし、次に何を達成すべきかを把握できる(図1)。

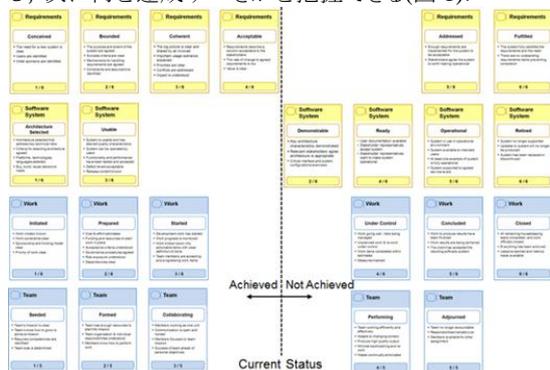


図1 アルファカード使用例(一部)

### 3.2. アセット適応アルファの導入

カーネルアルファを、アセットの構築と利用を定着させるのに、応用する。カーネルアルファの Software System に、以下にその状態群とその達成項目から成る「アセット適応(Asset Adaptation)」アルファ(表1)の定義を示す。これを基本アルファに加えて開発プロセスに導入しアセット活用の現状と次の達成項目を意識させる。

表1 アセット適応アルファ(Asset Adaptation Alpha)

	Alpha State	Goal
1	Asset identified	- Existence of related production patterns confirmed - A production pattern for reuse selected from asset repositories - Work products for reuse identified to meet afterwards identified requirements
2	Difference identified	- Work products in a production pattern for reuse determined - Differences between requirements and the work products for reuse identified
3	Asset adapted	- Work products for reuse customized - New work products for reuse created when no production pattern is applicable
4	Reuse planned	- Work products specified for reuse - Reuse of work products planned
5	Asset stored	- A set of work products stored as a production pattern into repositories - Reuse plan stored into repositories
6	Validity assessed	- Validity of work product assessed - Production patterns in repositories refreshed

### 4. 考察

カーネルアルファに基づくアセット活用ディシプリンは、開発進捗を、成果物起点ではなく、達成すべき状態で把握し管理する。このディシプリンは、ある時点の

開発の状態をきめ細かく構造化するため、現場で実践されている開発プロセスに織り込み易く、開発のある工程で期待される達成項目・タスクに組み込める。そのため、アセットの構築時に、業務ルールや制約を構造に転化させることや、開発の途中に、新たに表出した要求に対して、別のアセットの適用が容易になる。これにより、開発ライフサイクルを通じてアセット構築と利用性を最大化し、Web サービス開発者の競争力を発揮し得る。

開発現場では、現行の開発プロセスのマイルストーン毎に、アセット適応アルファの状態カードを対応付け、更に、開発プロセスで規定した WBS (Work Breakdown Structure)の各成果物と状態カードの各達成項目とを対応付けることで、実践を図れる。現行のプロジェクト管理プロセスの上位に重ね合わせることで、現場に適合させる。具体的には、例えば、アルファカードの状態管理をプロジェクト管理データ交換仕様 PROMCODE (PROject Management for COnttracted DELivery) [3]モデルで実装しアセット構築と流通を監視する Software Service Supply Network [4]の構築が考えられる。

また、アセット活用ディシプリンは、他の開発ディシプリンとも共存し、開発方法を強化し得る。例えば、ITソリューション・デリバリーのための人を重視した学習思考のハイブリッド型アジャイルアプローチに Disciplined Agile Delivery (DAD) [5]プロセス・フレームワークがある。これは、ゴール駆動型でスケーラブルであり、リスクと価値とを秤にかけ、費用と期間の制約の中で、高品質で有用なソリューションを定期的に提供する。このDAD ディシプリンの実践に、アセット適応アルファを組込むことで、アセットベースのDADを実現できる。

### 5. おわりに

今後、アセット活用ディシプリンをプロジェクト管理と共に実践し現場のアセット構築と利用を強化していく。

### 参考文献

- [1] Jacobson, I., Ng, P.W., McMahon, P., Spence, I., Lidman, S.: The Essence of Software Engineering: Applying the SEMAT Kernel, Addison-Wesley; 2013
- [2] Hosono, S., Shimomura, Y.: Towards Establishing Production Patterns to Manage Service Co-creation, CIRP IPS<sup>2</sup> 2013, p. 95-106
- [3] Nanzan University, IBM Japan, Ltd., Fujitsu Ltd., NEC Corp., NTT DATA Corp., Nomura Research Institute, Ltd.: PROMCODE Specifications v1.0, <http://lab.nise.org/PROMCODE/>
- [4] Aoyama, M.: A Model and its Management Architecture of Software Service Supply Chains, ICSServ2013, p. 130-135
- [5] Amber, S.W., Lines, M.: Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise, IBM Press; 2012