

ソフトウェア開発プロセスモデルの Quality gate による管理について

安田 淳 高橋 寿一

(株) SAP ジャパン 〒135-0064 東京都江東区青海 2-43 青海フロンティアビル

E-mail: jun.yasuda@sap.com, juichi.takahashi@sap.com

あらまし ソフトウェアが巨大化するとともに、多種多様なソフトウェアが毎年出荷されていく。単独ソフトウェア会社規模では指折りを誇る SAP ではその複雑化するソフトウェア出荷体制による品質障害を昨今抱えてきた。そこで Quality gate というシステムを導入することにより、巨大かつ複雑なソフトウェアの出荷をスムーズかつ高品質なシステムを導入した。

キーワード ソフトウェアプロセス、ビジネスソフトウェア、ISO9000

Quality gate control in software development process model

Jun Yasuda Juichi Takahashi

SAP Japan, Aomi Frontier Bldg. 2-43 Aomi Kotoku Tokyo, 135-006 Japan

E-mail: jun.yasuda@sap.com, juichi.takahashi@sap.com

Abstract Recently, a huge size and complicated software have released in a world. In this circumstance, SAP Co, which is one of the largest software company, have been to struggle to ship qualified products. Now, we employ novel methodology that is called "Quality Gate", besides we shall ship huge and complicated software by the Quality Gate system.

Keyword software process, business software, ISO 9001

1. ソフトウェア開発の現状

近年、ERP に代表される業務システムソフトウェアの開発は単一の製品として機能するだけでなくビジネスシナリオに基づいて複数の製品を連携させ、(ビジネスシナリオとは、企業内外の様々なプロセスを一連の業務の流れに沿って構成したものである。)さらに複数の外部ソフトウェアと連携するようになった。そのため、SAP では製品を単一の製品としての機能開発のみならず、複数の製品の連携を前提とした開発が必要とされるようになった。

実際には、このようなビジネスシナリオに合わせたシステムの構築では、多くのアドオンがなされた。例えば法改正、会計システムの変更等々によりそのアドオンが既存のソフトウェアに追加される。

さらにこのようなビジネスソフトウェアシステムは SAP で開発されたソフトウェアによってのみで動作するわけではなく、Microsoft, SUN などのオペレーティングシステム、SQL, Oracle に代表されるデータベース上で動作する。ビジネスソフトウェアシステムとしての品質を保持するには単一の製品の品質を保証するだけのアプローチでは問題の発生を防ぐことは難し

い。従来 SAP では単体の製品の品質を高めることに集中しており、ユニットテスト、コードレビューの徹底、ドキュメント管理など ISO9001, CMM にできるだけ従う開発手法をとってきた。しかし、ここ数年来製品の品質が同じ手法を適用しているにも関わらず、障害件数の増加もしくは品質の劣化が認められてきた。このため、SAP では個々の製品品質の向上にとりくむだけではなく、依存製品の品質を重要な品質保証活動の一部とすることによりシステム全体の品質を上昇させる独自のプロセスを実現させた。このような企業風土に適合したプロセスは ISO, CMM に準拠した形で遂行し、成果をあげた例を今回紹介する。[2]

2. ビジネスシナリオ及び製品の依存要素

2.1. ビジネスシナリオ



図 1

ビジネスシナリオとは、企業内外の様々なプロセスを一連の業務の流れに沿って構成したものである。製品間の技術的な連携だけではなく、顧客の日常業務に沿って製品の連携を考えてたものである。図 1 はビジネスシナリオとして提案されたプロセスをカバーするために必要とされる製品をマッピングしたものである。縦軸にビジネスシナリオ、横軸に製品が示され、一番上の列で出荷製品のバージョンが決定され、二行目以降にそのビジネスシナリオに対応するために必要な製品が×印で示されている。縦軸の製品はそれぞれ独立した製品であるため、すべての製品を一度の導入する必要がなく、顧客の状況に合わせた導入が可能となっている。

2.2. システム構成依存

SAPではオープンプラットフォーム戦略をとっているため、Microsoft や Sun に依存するような形ではソフトウェア開発を行っていない。しかし、それ故に OS では Windows, Linux, Solaris、データベースでは SQL, Oracle, DB2 などをサポートしているため、例えば Windows と SQL のプラットフォームでソフトウェアが問題ないと確認できても、その他のプラットフォームデータベースでの動作試験を省略するわけにはいかない。このような複雑なデータベース及び OS での動作試験を各製品品質マネージャが効率よく行うことは不可能である。往々にして製品品質マネージャの技術不足や時間の制約により省略した形で動作試験が行われ障害が顧客システム上で発生することが散見された。このため Quality gate ではシステム依存テストを独立した担当部署で行うことによりシステム依存の障害を減少させるとともに、製品担当品質マネージャの負荷を減らすことに成功した。

2.3. 言語依存

SAP では多くの製品で英語版出荷と同時期に 30 ヶ国語以上に対応したローカライズ版が出荷される。ローカライズに関しては多くの時間とコストが発生するが未だ効率的な方法を見出すのが困難であると同時に多言語に対応した高い製品を出すことは困難である[1]。システム構成依存の章で述べたようにこのような困難を製品品質マネージャにすべてを任すのは難しい。確かに SAP, Microsoft のような国際的なソフトウェア会社は世界中にローカライズの拠点があるが、製品品質マネージャがこれらの拠点に対し適切な担当者を見つけローカライズのタスクをアサインするのは膨大な時間を要するであろう。このような場合、製品品質マネージャとは別にソリューション品質マネージャをアプリケーションローカライズの品質に従事させることがより効率的な作業そしてローカライズ製品品質を高めることに寄与した。

3. 開発プロセスにおける要点

ビジネスシナリオを考慮に入れた開発において大きなポイントとなるのは、以下詳細に説明するように「各製品開発の状況の把握」と「複数の製品の開発のプロセス管理」である。

3.1 各製品開発の状況の把握

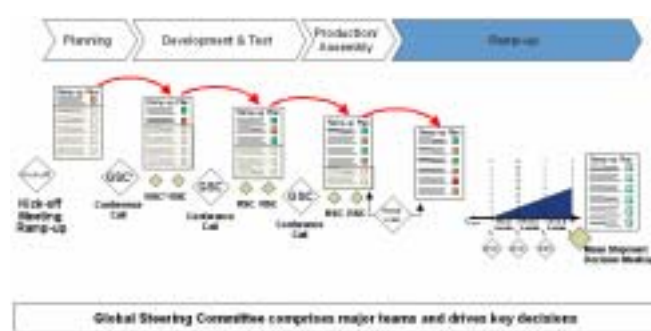


図 2

これまでの製品開発と異なり、仕様/設計による変更が製品単体としては問題がない場合であっても、ビジネスシナリオもしくはシステム依存、言語依存の観点から問題となる場合がある。仕様/設計による変更等がない開発を想定することは現実的ではないため、多くの開発では開発終了時点におけるテストに重点が置かれるが、ビジネスシナリオに基づいた製品の開発においては開発終了時点のテストでは発生した問題の原因を特定し修正するには時間がかかりすぎる。その対策として、Quality gate を設け、これまで曖昧であったフェーズ毎の区切りを明確にし、責任の所在を確定することにより、これまでの経過を把握しやすくした。

Quality gate においてチェックされる項目は、ビジネスシナリオと依存性に基づいて作成された仕様書の該当部分および、16 項目の Product standard 呼ばれる製品管理基準である。下記に 16 項目の Product standard を示す。

- 1.Accessibility
- 2.Application integration & Interfaces
- 3.Customizing & Configuration
- 4.Data Archiving
- 5.Development Environment
- 6.Documentation
- 7.Functional correctness
- 8.Globalization
- 9.Multiple clients
- 10.Opensource
- 11.Performance
- 12.Security
- 13.Solution management
- 14.Technical implementation
- 15.Third party
- 16.Usability

Product standard は単一の製品の品質を標準化する。ビジネスシナリオの部分で複数製品の連携の管理をソリューション品質マネージャが担当する。Quality gate にて使用されるチェックリストはソリューション品質マネージャにより管理され、製品担当品質マネージャによって遂行される。仕組みとしては、大きく計画、開発、製品化、出荷と 4 つのフェーズに分かれており、それぞれのフェーズの移行時に Quality gate が設定されソリューション品質マネージャによりチェックが行われる。チェックされる項目は計画の段階で決定されている。計画段階におけるチェック項目の決定ミーティングには関係するマネージャの参加が要求されており、この時点で決定された項目に合格しない限り、次フェーズへの移行は次フェーズのマネージャによって却下される。これにより、引き渡し時の不備の確認、変更箇所の報告が確実に行われる。これまでの開発においては後フェーズになればなるほど前フェーズの遅延、資料不足に悩まれ、結果として工期内に十分なテストが出来ない状況にあった。フェーズ間の引き渡し時の責任が不明確であったため、原因が後半のフェーズに発覚し、計画、開発時に起因する問題が追求しにくい状況があった。Quality gate ではフェーズ毎の責任が明確化となったため、次フェーズに於いて必要な作業環境を提供するための努力がなされるようになった。これにより、例えば、出荷フェーズの担当者は

出荷時に何が用意され、どの段階で準備が出来ているかが確認でき、現状で受け入れ可能かを決定できるようになった。また、引き渡す側としても不十分な状態であるにもかかわらず、完了部分のみを次フェーズへ引き渡すその場しのぎの行動が是正され、現状の報告と今後の対応を次フェーズと協議する場ができた。

3.2 複数の製品の開発のプロセス管理

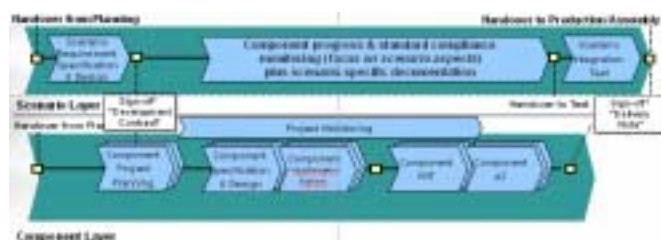


図 3

開発においては、シナリオベースのプロセス管理とコンポーネントベースのプロセス管理という 2 つの観点から開発を管理する。(図 3)

コンポーネントベースではこれまで行われてきた単一製品の開発プロセスが継承されている。開発時に要件として定義されるものの中に新たにビジネスシナリオの仕様で要求される項目が含まれ、ビジネスシナリオの担当者と各製品の開発マネージャの間で約束が交わされる。この仕様に基づいて各製品における仕様書の作成、開発、テストが行われる。この一連の流れの中における品質の基準となるのが前述の Product standard である。これにより各コンポーネントの品質の統一を図り、開発終了後のビジネスシナリオに基づいた統合テストに備える。

4. 品質マネジメント組織

多くの場合、製品の品質マネージャがその製品の品質の責任を負う立場にある。しかし上記に説明したように多くのビジネスシナリオ、OS、言語に依存した製品はそれらの全てに対し担当製品が動作するかということ进行测试する作業は著しく困難である。そのため SAP ではこの困難を乗り越えるため、そして Quality gate を実現させるために 2 種類の品質マネージャを置いている。

1. 製品担当品質マネージャ
2. ソリューション品質マネージャ

製品品質マネージャは従来から存在するある担当製品品質のマネージャである、ソリューション品質マネージャはある Quality gate を通過した製品が、全ての顧客の環境に適応できるかを担当する。そのためそ

の担当者は言語、OS の適合試験等々を担当する。製品品質マネージャとソリューション品質マネージャは全く同等の地位にあり、どちらかが製品の品質を保証しなければその製品が出荷されることはない。

4. まとめ

2002 年度における開発プロジェクトの中で Quality gate にてプロジェクト全体の再評価が行われた件について報告する。このケースの場合、製品化から出荷のフェーズに於いて発生。開発終了後、製品化における稼働テストでバグが規定以上発生。不合格の項目が 3、要検討項目が 3、合格 1 という検査結果であり、不合格製品は出荷が差し止められた。その最大の原因は

- ビジネスシナリオ間の矛盾による製品のエラー
- テスト量に対するリソースの不足
- テスターのスキルとテスト内容の不一致による遅延

が挙げられた。チェック項目があらかじめ決定されていたこと、前フェーズより引き渡される項目が明確に定義されていたため、従来、このような問題が発生した場合、2 ヶ月弱かかっていた再評価が3 週間弱に短縮された。

2002 年 12 月 4 日時点における報告。

Validation Criteria		Result	Value	Yellow	Red	Rel.
TCM - Installation/Upgrade	Number of CSN problems reported (prio 1+2)	(OK)	7	5	10	>
	Number of open CSN problems after validation (prio 1)	Not OK	2	0	0	>
	Number of open CSN problems after validation (prio 2)	Not OK	5	0	2	>
	Number of CSN problems completed, but not confirmed	OK	0	1	3	>
	Number of notes required (total)	(OK)	14	7	15	>
	Number of notes required (non Note Assistant, informational)	Not OK	14	6	12	>
	Run time (all component in days)	(OK)	2	1	2	>
		N/A				

図 4

2002 年 12 月 18 日時点における報告。

Validation Criteria		Result	Value	Yellow	Red	Rel.
Customizing & Configuration	Number of CSN problems reported (prio 1+2)	OK	0	5	10	>
	Number of open CSN problems after validation (prio 1)	OK	0	0	0	>
	Number of open CSN problems after validation (prio 2)	OK	0	0	2	>
	Number of CSN problems completed, but not confirmed	OK	0	1	3	>
	Number of notes required (total)	OK	0	4	9	>
	Number of notes required (non Note Assistant, informational)	OK	0	3	6	>
	Configuration time (standard config. in hrs)	OK	0	2	4	>

図 5

さらに、このようなプロセスが見える形そしてアウトプットを従う形で実現したことは開発者にとって有益であった。従来型のプロセスは残念ながら開発者にとって何が有益で何が不利益なのかを伝える手法に欠けていた。このようなシンプルで見える形のプロセスは多数の開発者及び開発者以外の部門が密接の関わる製品には適切と思われる。

文 献

[1] Takahashi, J. “Is Special Software Testing Necessary Before Releasing Products to an International Market?” *Quality Week 2000*, 1999

[2] Katy Dickinson, “Software Process Framework at Sun” *StandardView*, Sep. 1996, vol. 4 Issue 3