

トヨタ生産方式の導入によるビジネスアプリケーション開発

Development of Business Applications by Introducing the Toyota Production System

あらまし

富士通アプリケーションズ（FAP）は、富士通グループにおけるビジネスアプリケーションの開発専門会社として2004年4月に設立された。基盤事業であるビジネスアプリケーション開発は、SDASの開発標準に基づくほか、開発期間短縮、高品質が求められている。そこで、2004年12月から、トヨタ生産方式（TPS）のビジネスアプリケーション開発への導入を試行・検討しながら、実際のプロジェクトの製造工程へ本格的な適用を始めた。

本稿では、SDASを基盤とする実際のビジネスアプリケーションの短期間大規模開発プロジェクトにおいて、TPSをどのように具体的に導入し、日々の改善活動に結びつけ成果を獲得したかを紹介する。また、事例をとおして判明した高生産性および高品質を獲得するための課題について述べる。

Abstract

Fujitsu Applications, Ltd. (FAP) was established in April 2004 as part of the Fujitsu Group to focus on developing business application software. For developing business application software (the basic task of FAP based on the SDAS development standard), reducing the development period and achieving high quality are essential. In December 2004, we began introducing TPS for developing a full-scale application software production phase, while studying and attempting to apply TPS for developing some business application software. This paper introduces how TPS was concretely introduced (for relatively short-term, large-scale projects involving actual business application software based on SDAS), and how daily *Kaizen* activities are linked to achieve expected results. Also described are the issues for achieving high productivity and high quality, as clarified through the case examples.



関村 勉（せきむら つとむ）
富士通アプリケーションズ株式会社
ソフトウェアエンジニアリングセン
ター 所属
現在、短期間大規模ビジネスアプリ
ケーション開発に従事。



丸山富子（まるやま とみこ）
富士通アプリケーションズ株式会社
ソフトウェアエンジニアリングセン
ター 所属
現在、開発技術マネジメントに
従事。

ま え が き

富士通アプリケーションズ株式会社（以下、FAP）は、ビジネスアプリケーション開発の専業会社であり、SDASによるJava開発の実践部門として製造プロセスを革新し、短期間・高品質を実現する役割の一端を担っている。

ビジネスアプリケーション開発期間短縮のためには、「設計仕様を早く決める」こと、「不要な開発を減らす」こと、「段取りの効率化」が必要である。しかし、製造工程の現場サイドでは、「なかなか決まらない設計仕様」に対し、「できる範囲で実開発の規模を削減し」、「製造プロセスの並列開発を実施し」開発期間の短縮を図っているのが現状である。

さらに、製造工程では、「コストをかけない」という目標も同時に課せられている。このため、SDASで提示される、開発標準やプロジェクト管理標準を、製造現場に確実に定着・運用し、かつ短時間で課題を解決するには、改善活動の仕組みが機能することも必須である。

そこでFAPでは、2004年11月にマネージャクラスを対象としたハードウェア部門の「トヨタ生産方式（以下、TPS）の現場指導会」参画を契機に、ビジネスアプリケーション開発でも、TPSの導入が可能ではないかと考え、短期間大規模開発プロジェクトでの本格適用を決定した。

TPSの導入において特に着目したのは次の2点である。第一は、「現場が人を育成し人を生かす」。第二は、「徹底したムダの排除」である。

本稿では、これら二つの考えを中核に、SDASとどう組み合わせ、TPS導入を図り、2.5箇月で1.1メガステップの製造工程を、オフショア先（海外の開発委託先）である富士通（西安）系统工程有限公司（以下、FXS）と協働して実現できたかを説明する。また、今回の取組みは、製造工程に閉じた取組みであり、プロジェクト全体の期間短縮とコスト削減にはまだ課題がある。製造工程の改善は無論であるが、製造工程側から見た設計工程への提案を、今回の事例を通じて得た貴重な教訓として説明する。

なお、本稿で使用する工程名は、本特集でも掲載している富士通のシステム開発標準に準拠している。

TPS実践（現場が人を育成し人を生かす）

FAPでは、まず製造工程の現場要員の自律した能力発揮と成長のために、つぎの四つの取組みを実践した。

(1) 標準作業は現場が制定し現場で改善

SDASの開発標準をベースに、現場のリーダーが中心となり、各作業項目を現場に合わせ詳細化し、各作業の順番を規定し、さらに各作業項目の予定時間を設定した。製造工程に関係する要員は、日々の作業項目の実績時間を、決められた形式で毎日提出し、リーダーが設定した予定時間と実績時間の差異を把握する。差異がある場合には、理由を明確にし、リーダーと検討の上、改善が必要な場合には即刻対策を講じた。作業項目の詳細化の例を図-1に示す。

今回は、短期間大規模開発であるため、従来なら1プロセス1名で一貫開発{プログラム構造設計（以下、PS工程）～プログラムテスト（以下、PT工程）}することが多かったが、今回は、図-2のような作業分割と並行開発を実践した。

(2) 異常を容易に判断できる仕組みを整備

TPSでいう、「見える化」である。

関係者が早く異常に気付き、改善のための手を早く打つことができる仕組みである。よくある例では、白板に進捗状況、品質状況、課題などを書いて関係

| PG(プログラミング) | |
|------------------|---|
| 画面アプリケーション | |
| 対象者: 開発者(拠点:FXS) | |
| インプット | JSP(JavaServer Pages) DataBeanクラス(Java) 電文クラス(Java) オンライン定義書 画面要素制御表 画面項目定義書 業務ID_画面AP品質チェックシート |
| 作業内容 | 資産配置 PG作業手順 6 |
| 作業時間 | JSP作成(固有部)&修正 18 JSP修正手順書 |
| 詳細作業手順書 | 画面アプリケーション作成(固有部) 102 画面アプリケーション開発記述手順書 |
| | 画面アプリケーション自己レビュー 18 |
| | 画面アプリケーション対面レビュー 120 |
| | 264 |
| アウトプット | JSP(実装済み) 画面アプリケーションクラス(Java 実装済み) |

単位:分

図-1 作業項目の詳細化例
Fig.1-Example of details of work item.

トヨタ生産方式の導入によるビジネスアプリケーション開発

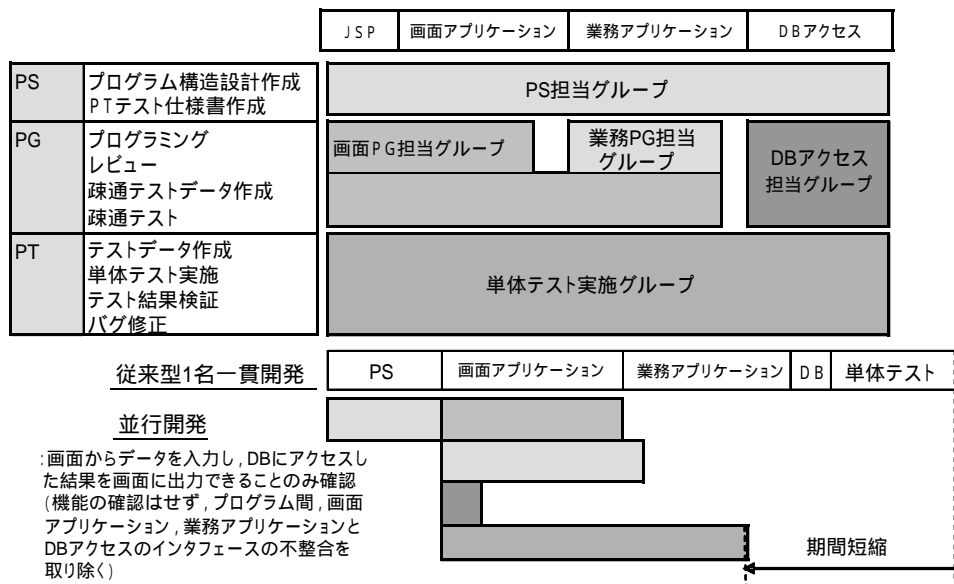


図-2 作業分割と並行開発イメージ
Fig.2-Sharing of work and parallel development image.

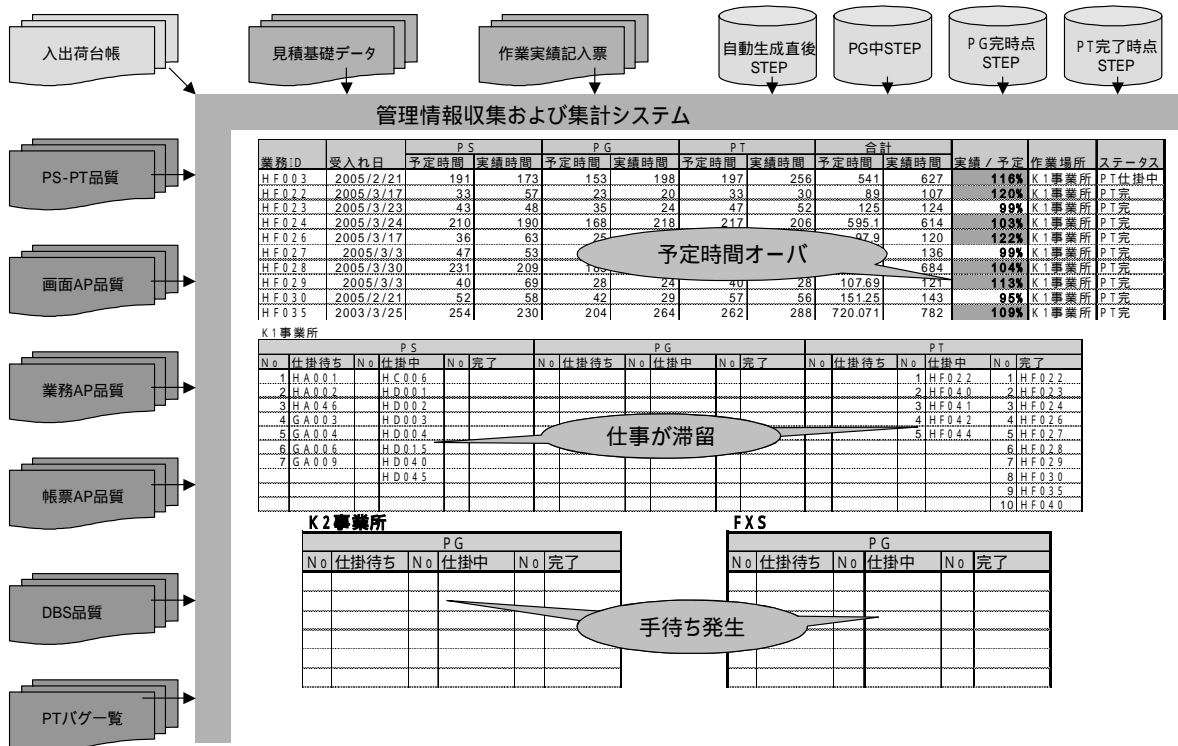


図-3 プロジェクト管理情報の収集
Fig.3-Collection of project management information.

者と共有化する。この方法は作業場所が1箇所の場合には有効であるが、複数拠点(オフショアを含む)で並行開発が実施される場合には、ITを活用した全拠点間での異常を判断できる仕組みの整備が必須である。ただし、仕組みの簡素化を図るために、

関係者に三つの約束を徹底させた。一つ目は作業時間を毎日提出する。二つ目は決められた共有サーバのフォルダに情報を格納する。そして三つ目が決められたワークシートに正しく情報を記述する。簡単な約束ごとの遵守により、図-3のように、情

報の収集は容易になり、各開発拠点にまたがった異常の検出ができるようになった。

(3) 人がツールを使いこなす

短期開発においては、開発者の作業量を削減するために、プログラムソースの自動生成は必須技術である。本プロジェクトでも例外でなく、SDASで提供するフレームワーク（Interstage Application Framework Suite）上に自社開発した自動生成ツールをプロジェクト用に構築した。ただし、ツール開発では、TPSの考えにのっとり、三つの前提を置いた。一つ目は、自動化を目的とせず、人でしかできないことに人の作業を集中させる。二つ目は、単純ミスを未然に防止する仕組み（簡単転送や繰り返し作業など）を組み込む。三つ目は、標準化規約にのっとり固定的なプログラム記述を生成する。本プロジェクトでは80%強の自動生成を実現した。

(4) 日々の改善サイクル

国内での改善活動は無論であるが、オフショア先であるFXSでも同時に改善活動を推進した。今回の範囲がPG工程に限定されたため、仕様を伝達するブリッジSEを確保するのではなく、TPS導入による製造プロセス定着のために、現地に国内のリーダークラスの要員を常駐させた。その結果、表-1のような愚直な改善が日々実施され、本プロジェクトにとどまらず、他プロジェクトのオフショアでも改善活動が横展開できるようになった。とくに、PG工程を依頼する場合の仕様書の書き方の規定方法に関しては大きな収穫であった。

表-1 オフショア先における日々の改善事例

| 日付 | No. | 問題提起とアクション |
|------------|-----|--|
| | ∴ | |
| | 4 | 各開発者にOJTで発生した問題点・課題を最低五つ以上抽出させ、1名ずつ面談実施 |
| 2005/01/17 | 5 | 仕様書に記述されている日本語の意味が不明「下2桁」・「4未満」・「AおよびB」など 「分かりやすい日本語.XLS」を作成。 今後更新継続 |
| 2005/01/21 | 6 | 待ち時間の発生 立上げ教育で準備した演習問題を繰り返し実施 |
| | 7 | 対面レビューにより共通のミスを発見 抽出項目を自己レビューチェックシートに反映 |
| | ∴ | |

TPS実践（徹底したムダの排除）

製造工程におけるムダの排除は、TPS導入では重要である。TPSには次の七つのムダ⁽¹⁾の概念があり、それぞれに対応するビジネスアプリケーション開発の製造工程でムダの排除を現場で実践した。

(1) つくりすぎのムダ

システム構造設計（以下、SS工程）にない機能は作らない。

(2) 手待ちのムダ

仕様書の提示遅れによる製造工程の開始の遅れは、要員を確保し待機している場合には、待ち時間が発生する。また、プログラムテスト時の障害発生によるテスト待ちなども手待ちのムダとなる。

(3) 運搬のムダ

各工程間の資産の移動に伴う作業のムダが該当する。仕様書の入荷チェックや納品時の出荷チェックは一見ムダがないが、入出荷チェックを提供側と受領側の両者で行う場合が結構あり、重複チェックによるムダがある。

(4) 加工そのもののムダ

開発現場で機能していない作業項目の実施や、作業分担の誤りによる過剰な作業時間の投入が該当する。

(5) 在庫のムダ

仕様書記述内容の確認や、誤り指摘によるQA待ち状態により、出荷ができない仕掛中プログラムの状況などが該当する。

(6) 動作のムダ

関係者が同一拠点にいるにもかかわらず、離れた場所で作業をしている場合のムダな移動に伴う場合などが該当する。

(7) 不良をつくるムダ

製造工程におけるバグの作込みなどが該当する。

TPS第一号導入効果

TPS実践による導入効果を図-4に示す。1次開発（以下、1次）と2次開発（以下、2次）では、製造工程の生産性は46%向上した。

なお、生産性の単位は、プロジェクトや担当者により異なるため、誤解を招かないよう、図-4では1次の全体を100とする2次の比と、1次と2次の製造工程の各工程の作業時間比を比較し、導入効果を算

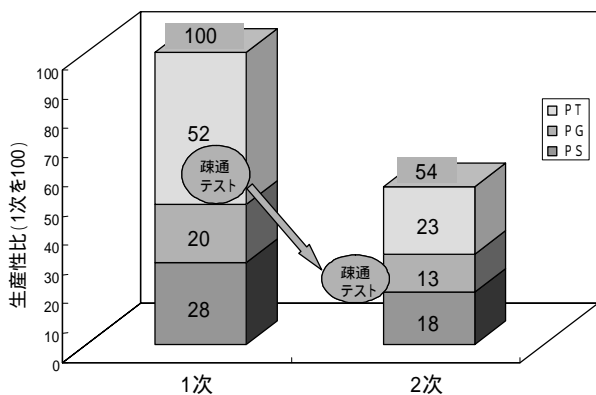


図-4 TPS第一号導入効果
Fig.4-First TPS introduction effect.

出した。

(1) PS工程は生産性36%アップ

従来の製造プロセスは、PS工程からPT工程まで一貫して1名の要員が実施することが多く、PS工程でプログラム内の処理手順や処理条件を明確にしないまま問題を先送りし、PG工程やPT工程で問題を解決することが多くあった。

しかし、本プロジェクトでは製造プロセスを見直し、PS工程をきちんと実施しないと実装に進めない仕組みとした。それは、一つのプログラムの製造工程を複数要員で分割して作成することと、PS工程で作成する仕様書からプログラム行の80%以上を自動生成する仕組みの導入であった。

今回の製造プロセスは初めての試みであったが、1次の生産性は、FAPの過去の他プロジェクトと比較するとトップであった。また2次では以下の2点の改善が功を奏し、36%の効率向上を実現できた。第一は、ペアプログラミングを応用したペアプログラミング構造設計による経験の浅い要員の生産性の向上である。第二は、QA回答時間の短縮によるPS工程での在庫のムダの排除であり、数値的には1次の翌日回答率57.5%を2次では77.5%まで高めることができた。

(2) PG工程は生産性35%アップ

PG工程はオフショアとしてFXSが担当し、積極的かつ日々の愚直な改善運動により効果がでた。とくに2次開発前に、1次のQA発行原因内容の分析を実施しPS工程の仕様書の書き方の型決めをした。その結果、同一規模でのQA発行件数を約1/3に削減した。さらに、PS工程と同様にQA回答期間の短縮

の目標を立て、即日回答率100%を実現した。

(3) PT工程は生産性55%アップ

1次では、画面アプリケーションと業務アプリケーションの疎通確認にPT工程の35%の時間を要した。七つのムダの一つである加工そのもののムダを、疎通確認という作業で役割分担の誤りにより発生させていた。改善として、国内の疎通確認の作業項目を、FXSのPG工程の作業項目に移管し解決した。

また、不良をつくるムダを回避するために、PT工程で発見されたバグを1次で分析し、組織的に横展開する対応を実施した。

上記3工程の効率化は、特別な技法やツールなどを適用しなくても、プロジェクトに適合した製造プロセスの定義、ムダの発見とムダの排除という日々の改善活動により大きな効果を生むことを証明できた。

今後の課題

TPSは、ともすると方式に特有の用語や手段が、ことさら強制的に取り上げられ、例えば「ムダを徹底的に追求し、在庫をもたない」に着目されがちである。本来は、生産に関する全体組織の形態および運用の仕方、さらにそれに底流する基本的な思想⁽²⁾が従来と異なり、生産への変革姿勢を問われていると言っても過言でない。

FAPのTPS導入は、スタートラインに立ったばかりであり今後一層の適用を図っていく。TPSの第一号導入経験から、とくに以下の二つの課題に取り組む。

第一は、FAPへ期待されている、大規模ビジネスアプリケーションの短期間開発と高品質実現のために、現場要員を早期に立上げる技術を確立する。とくに、新たな標準化、技法やツールを適用する場合には、要員の力量を把握した事前教育の場を設定し、プロセスの遂行ができるレベルまで引き上げる。そして、要員自らが改善課題を見つけて解決できる方法を提示するとともに、改善意欲も同時に植付けていく。ソフトウェア開発は個人の生産性に差があり、本プロジェクトでも、PG工程に着目すると生産性格差は5.5倍であった。これから分かるように、平均的な取組みでなく、一人一人を見つめた対応を実施する必要がある。

なお、本プロジェクトにおける製造プロセスの立上げ教育は、全体の4.5%の工数を投入した。今後は、生産性は無論であるが、妥当な教育工数も統計的に蓄積して、データの有効な活用方法も模索する。

第二は、TPSの取組みを製造工程にとどめることなく、設計工程やテスト工程にも拡大する。とくに、短期間開発では、仕様書記述に関するQA件数の削減が必須であり、設計側と協働し、設計の型決めを現場レベルで具体化する。SDASでも開発標準としてドキュメント標準を提示している。しかし、それは汎用的であり、現場の実態に合わせ、さらに次の4段階でQA件数を確実に削減し、仕様書の曖昧さも減少させていく。

第1段階は、仕様書の記述項目の設計側と製造側の意識を合わせる。FAPからは、実装上不可欠な仕様書情報を提示し、設計者側は、顧客承認などで必要な仕様書情報を提示し、両方で徹底的に検討した上で記述項目を決める。

第2段階は、記述項目のレベル（深さ）の合意を取る。

第3段階は、実プロジェクトでの仕様書記述例を

先行業務開発過程で評価しながら確定する。

第4段階は、設計側および製造側に、仕様書の記述方法を教育し徹底する。

む す び

FAPにおけるTPSの導入とSDASとの融合はまだまだ経験が浅く、これからが本格的導入となる。改善活動の定着は一朝一夕でできるものでなく、愚直な改善活動とその徹底により成立するものである。改善活動を個々のプロジェクト活動から組織的活動に高め、改善の推進状況を「見える化」により改善意欲を喚起できるデータと仕組みを考え、FAPの活動にとどめることなく、情報を積極的に公開し、関係者の改善活動の議論ができる場を提供していきたい。

参考文献

- (1) 大野耐一：トヨタ生産方式 脱規模の経営をめざして．ダイヤモンド社，1978．
- (2) 岩城宏一：実践トヨタ生産方式 人と組織を活かすコスト革命．初版，日本経済新聞社，2005．