

シナリオによるサービス連携方式の実験評価

渡邊 拓二, 築 栄司, 内藤 岳, 山本 修一郎

NTT 情報流通プラットフォーム研究所

概要

電子商取引への SOHO や中小企業の参入を容易化するためには、電子商取引システムの構築を容易化する必要がある。このため筆者らは、課金、認証などの EC で共通的に必要となる要素サービスを連携できるプラットフォームを構築し、EC サービスを要素サービスの組み合わせで実現する効率的なサービス連携方式について研究を進めている。

本稿では、サービス連携方式として InfoSTAGE を提案するとともに、InfoSTAGE を用いた EC サイトを実験的に構築し、サービス構築工数を約 80%向上できることを示す。

The experimental evaluation of the service integration system using Scenario

Takuji WATANABE, Eiji YANA, Takashi NAITO, Shuichiro YAMAMOTO
NTT Information Sharing Platform Laboratories

Abstract

In order to make easy entry of SOHO and SMEs to e-commerce, e-commerce service development needs to be simplified. For this reason we are developing the service integration platform, which can integrate the common EC services such as billing and authentication.

In this paper, we propose a service integration platform InfoSTAGE. We also evaluate the effectiveness of InfoSTAGE by constructing EC site experimentally using infoSTAGE. The result shows the productivity of service development can improve about 80%.

1. はじめに

電子商取引市場では、SOHO の参入が期待されているが、コスト面や技術的知識不足から、SOHO が電子商取引になかなか参入できていないのが現状である。従って、安価で容易に電子商取引システムの構築を行う手法が求められている。

筆者らは、Web を用いたサービス(以下 WSV Web based SerVices)と人間の作業を連携するための言語として、XML ベースのシナリオ記述言語 CSML を提案した。[1][2] また、CSML に基づいて、サービス連携プラットフォーム(以下サービス連携 PF)InfoSTAGE の開発を行っている。本稿では、この InfoSTAGE によるビジネスモデルを、検証するために構築した実験システムと、その評価・考察について述べる。

2. CSML と InfoSTAGE の特徴

2.1. CSML の特徴

CSML の特徴は、次の 3 つである。

[特徴 1] XML 形式のシナリオ記述言語

XML のタグを用いて、連携処理、並行処理、分岐処理が構造的に記述できるので、高度なビジネスプロセスを容易に表現可能である。

[特徴 2] データ条件の記述を排除

CSML では、WSV の実行順序および実行条件のみを記述し、ビジネスプロセスを定義する。WSV の固有データを記述しないので、シナリオを書き換えることなく、シナリオの再利用が可能である。

[特徴 3] 例外処理が記述可能

シナリオ内の任意の場所で発生する可能性のある処理を例外処理として一か所に記述でき、高信頼なシナリオが作成できる。

このことからわかるように、CSML は、柔軟で容易に理解でき、WSV に関する知識のないユーザにでも、ビジネスプロセスを開発することが可能となるシナリオ言語である。

2.2. InfoSTAGE の特徴

InfoSTAGE の特徴は、次の 3 つである。

[特徴 1] HTTP による WSV 連携

WSV の呼び出しに HTTP を利用することで、ネットワーク上を介した様々な WSV の連携が可能となる。

[特徴 2] データ連携が容易

WSV 間で交換するデータを格納する共有データインスタンスを提供することにより、各 WSV は連携する他の WSV を意識せずに、データの入出力を行うことが可能となる。

[特徴 3] WSV のトランスペアレントな管理

WSV の物理位置をディレクトリで管理し、シナリオでは WSV の論理名を記述することにより、WSV の物理位置の変更に対するシナリオの不変性を保証できる。

このように、InfoSTAGE では、ビジネスプロセスのみを記述したシナリオによって制御されるため、ビジネスプロセスの開発と WSV 開発とを分離することができる。図 1 に従って、InfoSTAGE を用いた WSV のビジネスモデルを説明する。

「一次サービス提供者」は、ビジネスプロセスを意識せず WSV を開発する。ビジネスプロセスを開発する「二次サービス提供者」は、「一次サービス提供者」が提供する WSV を組み合わせてシナリオを記述、再利用して、ネットワークを介した複数の WSV による連携サービスを提供する。

図 1 に示したように、InfoSTAGE を利用した新たなビジネスモデルにより、電子商取引システムの構築を容易化できる。

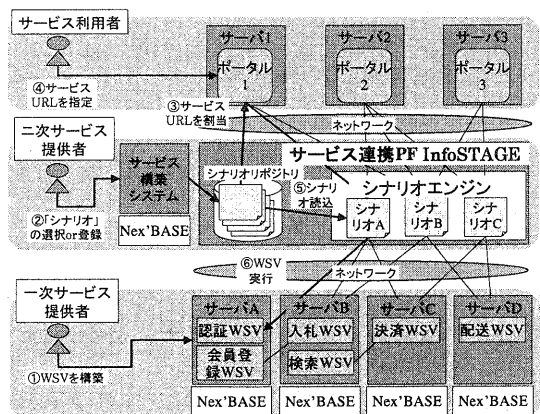


図 1 InfoSTAGE によるビジネスモデル

3. 実験による評価

3.1. 実験の概要

InfoSTAGE の有効性を評価するため、次の手順で実験を実施した。

【評価手順 1】オークション構築システム(以下 **ASDS Auction Service Development System**)を構築

【評価手順 2】ASDS を用いて、サービス1で必要となるシナリオ・WSV を作成

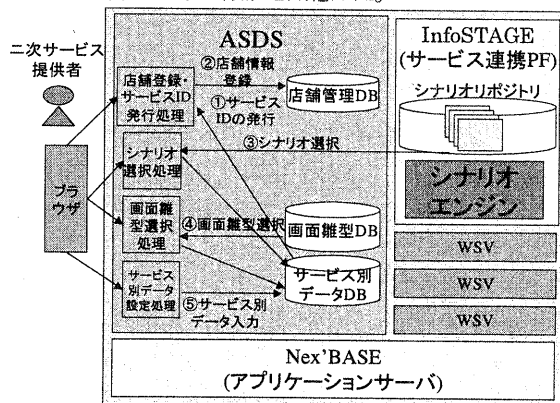
【評価手順 3】サービス 1 を再利用して、サービス 2 で必要となるシナリオ・WSV を作成

【評価手順 4】サービス 1 とサービス 2 の開発工数を比較評価

オークションサイトを評価対象とした理由は、会員登録やメール送信などの基本的な WSV を含み、かつ多様な連携サービスが想定可能ということからである。ASDS では、「二次サービス提供者」がオークションサイトを構築するためのシステム構築機能を持ち、「一次サービス提供者」によって作成された複数の WSV を、「二次サービス提供者」が、シナリオを選択することによって、連携することができる。

3.2. 【評価手順 1】 ASDS の構築

ASDS では、シナリオ・WSV の再利用を可能にするため、次のような機能を用意した。



【機能 1】 店舗登録・サービス ID 発行

サービス別データ DB が、サービス ID を払い出し、

店舗登録情報を店舗管理 DB に登録する。

【機能 2】 シナリオ選択

InfoSTAGE のシナリオリポジトリに格納されているシナリオを選択し、サービス別データ DB に選択したシナリオの ID を登録する。

【機能 3】 画面レイアウト雛型選択

画面雛型 DB より、画面レイアウト雛型を選択し、雛型 ID をサービス別データ DB に登録する。

【機能 4】 サービス別データ選択・入力

選択した画面レイアウト雛型に埋め込むために、店舗タイトルやロゴなどを選択・入力する。

各データが登録されると、サービスが利用するシナリオを起動するためのリンクが、モールドトップ WSV に張られる。そして、そのリンクを選択すると、シナリオが起動し、各 WSV は引き継がれたサービス ID をキーとして、画面雛型 DB より画面レイアウト雛型、サービス別データ DB より、店舗データを参照し、動的に画面を生成表示する。

以上の機能によって、シナリオの選択と WSV のカスタマイズのみで、「二次サービス提供者」の迅速なサービス構築を可能とした。

また、ASDS では、「一次サービス提供者」が作成する WSV と、「二次サービス提供者」が記述するシナリオを、あらかじめ用意することとした。このため既存オークションサイトを調査し、最小限必要な機能を抽出した。次いで、シナリオ・WSV の再利用を可能とするため、以下のような作成ルールに基づいてシナリオ・WSV を設計した。

【シナリオ粒度ルール】

- ・ 繰り返し利用されるプロセスを 1 シナリオとする
例えば、会員登録プロセスは、店舗トップ WSV からも、認証 WSV からも起動されるため、1 つのシナリオとする。

【WSV 粒度ルール】

- ・ シナリオにより入替を行う単位を 1 WSV とする
例えば、ID/PW 発行はメールで送る、Web に表示するという 2 パターンがあるので、ID/PW 発行を 1 つの WSV とする。

この粒度に従って、シナリオ・WSV を作成したところ、付録 2,3 のような構成となった。

次に、シナリオ・WSV の例外処理について次のようにルールを定義した。

[例外処理ルール]

- ① WSV 内の再実行処理は、WSV で処理を行い、シナリオの強制終了、再実行はシナリオの例外処理で行う
- ② 認証エラー時に、シナリオの例外処理で会員登録シナリオに遷移する

InfoSTAGE では、機能としてシナリオインスタンスの親子関係を管理し、インスタンス間でのデータのやり取りなどを可能としている。ASDS では以下のルールで親子シナリオを利用することとした。

[親子シナリオの利用ルール]

- ① 認証エラー時に、子シナリオとしてシナリオ会員登録シナリオに遷移し、ID/PW 発行後、親シナリオへ復帰する
- ② 親シナリオ起動用のパラメータを共有データインスタンスへ格納し、親へ復帰する WSV がそのパラメータを参照し、復帰する

ASDS における WSV の入出力データは、以下の 3 種類に分類し設計した。

[WSV 入出力データ分類ルール]

- ① 共有データインスタンス入出力
シナリオ内の WSV 間で交換するデータを規定。
- ② データベース入出力
シナリオが終了しても保存されるデータ、シナリオ間で受渡しをするデータを規定。
- ③ パラメータ入出力
InfoSTAGE が WSV に受け渡すパラメータを規定。

[WSV 画面設計ルール]

ASDS の WSV 画面仕様は、ヘッダー部、本体部、フッター部に分かれる。ヘッダー部、フッター部は各 WSV 共通で、店舗構築の際入力したカスタマイズデータにより実装される。本体部は WSV 単位で設計を行う。

[共有データ利用ルール]

モールトップページシナリオを起動する時に作成される共有データインスタンスを、すべてのシナリオ

間で引き継ぐことにより、データの引き渡しを行う。よって一度認証 WSV で入力したユーザ ID は、他の店舗に行っても保持される。また作成された共有データインスタンスは、モールエンド WSV に遷移することにより消去される。

[サービス ID 引継ぎルール]

サービス ID は、モールトップ WSV で店舗を選択する際に、共有データインスタンスに格納する。そして、サービス ID が格納された共有データインスタンスをシナリオ間で引き継ぐことにより、サービス ID を引き継ぐこととする。

3.3. [評価手順 2] サービス 1 の開発

シナリオ 5 個、WSV 20 個を作成。会員登録シナリオ内の会員登録 ID/PW 発行 WSV では、ID/PW を Web のページに表示する。また入札シナリオ・商品登録シナリオ内の認証 WSV において、認証に失敗すると、子シナリオとして会員登録シナリオを起動し、登録後入札シナリオ・商品登録シナリオへ戻る。

3.4. [評価手順 3] サービス 2 の開発

サービス 1 で作成したシナリオと WSV のうち、シナリオ 3 個、WSV 17 個をそのまま流用し、シナリオ 2 個と WSV 3 個を改造することによって、サービス 2 を作成した。

改造したシナリオは、オークション会員登録シナリオと提供者会員登録シナリオである。また改造したサービスである会員登録 ID/PW 発行 WSV では、ID/PW 発行をメールで行うように改造した。また入札シナリオ・商品登録シナリオで用いる認証 WSV を改造した。改造は、認証に失敗すると、シナリオを消滅し会員登録シナリオを起動し、登録後は店舗トップページ WSV へ戻るよう変更したことである。

3.5. [評価手順 4] 比較評価

サービス 2 の作成では、図 3 のようにサービス 1 で作成したシナリオ・WSV の内、シナリオを 60%、WSV を 85% 流用し、残りのシナリオ・WSV は改造した。その結果、サービス 1 の開発時間 23.5H に対して、サービス 2 の開発時間が 5H になった。

よって、サービス 1 を新規開発した工数に対して、サービス 2 の再利用の工数は、サービス 1 の約 20%

となることが明らかとなった。

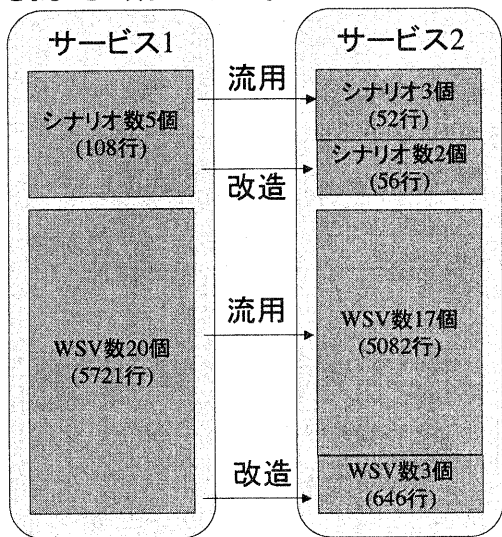


図 3 サービス 1 とサービス 2 の関係

4. 考察

4.1. 生産性の向上

同様のサービスを新規開発するときの工数を再利用による開発と比較すると、再利用方式の工数が新規開発方式の約 20%であることを明らかにした。よって InfoSTAGE を用いることにより、EC サイトの構築工数を約 80 向上させることが可能であることがわかった。

4.2. 共有データインスタンスの有効性

ASDS では、共有データインスタンスにより、WSV 間、シナリオ間のデータの持ち回りを行った。この機構を用いることによって、一度認証入力をすれば、ブラウザのセッションが切れたり、他のシナリオに遷移したりした後も、認証情報の引き継ぎが可能となり、ユーザのキー入力の煩雑さを少なくすることが可能となった。

5. まとめ

本稿では、InfoSTAGE を利用した、シナリオによるサービス連携方式の提案と、実験システムの構築・評価を行い、以下のことを明らかにした。

- ・ EC サイト構築工数を約 80%向上できる
- ・ 共有データインスタンスにより、他の WSV を意識せず、WSV 間でデータを持ち回ることが可能となる

また、InfoSTAGE では、WSV における入出力データの整合性を人手で設計時に確認する必要がある。今後は作成したシナリオ内のデータの整合性を自動的に検証する機構を、検討する必要がある。

<参考文献>

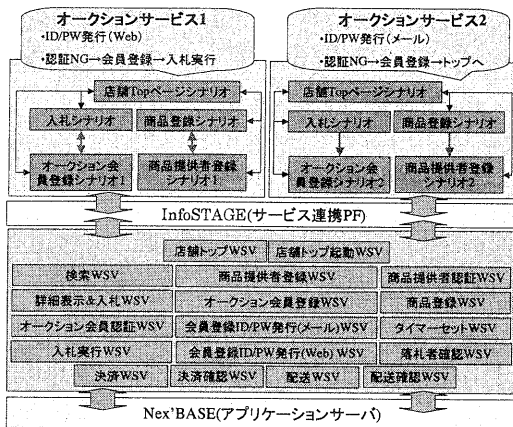
- [1]畑,内藤,山本:「InfoSTAGE イベントドリブン型言語使用(CSML)の検討」,電子情報通信学会知能ソフトウェア工学研究会,1999年5月
- [2]HATA, etc., Collaboration oriented Scenario Markup Language:CSML, WSVSITT'99, Aug.1999
- [3]畑,内藤:「異種サービス連携基盤 InfoSTAGE の提案」,情報処理学会第 59 回全国大会,1999年9月
- [4]山本:「相互の理解 2.組織のバーチャル化と相互理解」,岩波書店,1999
- [5]高田,鈴木,長岡:「情報流通プラットフォーム・コンポーネント結合基盤 Nex'BASE の開発状況と今後の課題」,NTT 技術ジャーナル Vol.11,No.5, pp.13-19,1999
- [6]富田,横関,高田:「情報流通アプリケーション開発を効率化する Nex'BASE スクリプト」,NTT 技術ジャーナル Vol.11,No.5,pp.20-23,1999
- [7]高田,鈴木,山本:「インターネットサービス構築用スクリプト言語の提案」,電子情報通信学会 KBSE99-37,pp.15-21,1999年11月

```

<?xml version="1.0" encoding="Shift_jis"?>
<Scenario Name="入札シナリオ" Ver="1.0">
  <DeclarationPart>
    宣言部
    <Description>
      オークションを実施する
    </Description>
  </DeclarationPart>
  <BodyPart>
    本体部
    <Activity Name="検索WSV"/>
    <Activity Name="入札WSV"/>
  </BodyPart>
  <ExceptionPart>
    例外処理部
    <Exception Event="NG">
      <Activity Name="店舗トップ起動WSV"/>
    </Exception>
  </ExceptionPart>
</Scenario>

```

付録1 CSMLの記述例



付録2 シナリオ-WSV構成図

	サービス1のシナリオ	行数	サービス2のシナリオ	行数
流用部分	店舗トップシナリオ	9	同左	9
	商品登録シナリオ	23		23
	入札シナリオ	20		20
改造部分	オークション会員登録シナリオ1	28	オークション会員登録シナリオ2	28
	商品提供者会員登録シナリオ1	28	商品提供者会員登録シナリオ2	28
合計		108		108

	サービス1のWSV	行数	サービス2のWSV	行数
流用部分	店舗トップWSV	190	同左	190
	オークション会員登録WSV	546		546
	会員登録ID/PW発行(WEB)WSV	117		117
	店舗トップ起動WSV	22		22
	オークション会員登録WSV	546		546
	商品提供者登録WSV	556		556
	商品登録WSV	551		551
	タイマーセットWSV	355		355
	落札者確認WSV	384		384
	決済WSV	159		159
	決済確認WSV	195		195
	配送WSV	369		369
	配送確認WSV	208		208
	店舗トップ起動WSV	22		22
	検索WSV	263		263
詳細表示&入札WSV	357	357		
入札実行WSV	242	242		
改造部分	会員登録ID/PW発行(WEB)WSV	117	会員登録ID/PW発行(メール)WSV	111
	商品提供者認証1WSV	261	商品提供者認証2WSV	263
	オークション会員登録1WSV	261	オークション会員登録2WSV	272
合計		5721		5728

付録3 シナリオ-WSV一覧