

---

---

# 「イベントリアル通信を可能とする

## サーバ通信手法」の開発

株式会社 アクトシステムズ

---

### ■ 執筆者Profile ■



高橋俊博

1979年 株式会社アクトシステムズ入社  
ホストシステム担当  
第一システム課課長



中居洋司

1986年 株式会社アクトシステムズ入社  
アパレル関係の各種システム開発を経験  
現在、第一システム課にて、各種開発のプロジェクトリーダーとして活躍中  
主査補



大見邦男

1991年 株式会社アクトシステムズ入社  
SE経験後技術開発室に所属  
現在、新技術の開発に活躍中  
主任

## ■ 論文要旨 ■

インターネットは急速に普及し、多くの場面でインターネットが使用されているが、基幹業務に使用できるシステムはあまり見受けられない。本論文では、「Webアプリケーションでありながら、オフィスコンピュータの使い易さ」を目的として開発した、「イベントリアル通信を可能とするサーバ通信手法」の開発経緯を述べている。今回の開発において、当初の目的を達成する事ができ、次の事が実現できた。

- 入力項目のリアルチェック
- マウスなしでも操作が可能
- ファンクションキーへの各種機能の振付
- 快適なレスポンス
- 既存ネットワークでの使用
- 高いメンテナンス性

## ■ 論文目次 ■

<b>1. はじめに</b> .....	《 5》
1. 1  当社の概要とアパレル業	
<b>2. Webアプリケーションの問題点</b> .....	《 5》
<b>3. 「イベントリアル通信を可能とするサーバ通信手法」とは</b> .....	《 6》
3. 1  「イベントリアル通信を可能とするサーバ通信手法」の機能目標	
3. 1. 1  入力項目のリアルチェック	
3. 1. 2  基幹業務に使用できる操作性	
3. 1. 3  基幹業務に使用できる快適なレスポンス	
3. 1. 4  既存のネットワーク環境で使用可能	
3. 1. 5  システムメンテナンス性の向上	
<b>4. 開発の実施状況</b> .....	《 7》
4. 1  基本構造	
4. 1. 1  通信モジュール	
4. 1. 2  サーバスクリプト	
4. 1. 3  HTML	
4. 1. 4  送受信メッセージ	
4. 2  開発状況 .....	《 8》
4. 2. 1  同期通信での問題点	
4. 2. 2  3KB超えのイベントメッセージが届かない問題発生	
4. 2. 3  非同期通信が正しく行われない問題	
4. 2. 4  非同期→同期通信切替時の非同期メッセージ再送処理の追加	
4. 2. 5  別の処理画面を開く際の問題	
4. 2. 6  同期通信非同期処理の追加	
4. 2. 7  レスポンスの改善	
4. 2. 8  その他の実現された機能	
<b>5. 開発を終わって</b> .....	《 14》
5. 1  今後について	

■ 図表一覧 ■

図 1	基本構造図	《 7》
図 2	同期通信のみ使用	《 8》
図 3	非同期通信を使用	《 8》
図 4	非同期通信の障害	《 9》
図 5	ソケットを使用した非同期通信	《 10》
図 6	レスポンスの到着順不整合	《 10》
図 7	未完メッセージ再送処理	《 11》
図 8	セッション領域	《 11》
図 9	セッション領域の分割	《 12》
図 10	同期通信処理	《 12》
図 11	同期通信非同期処理	《 12》
図 12	レスポンス改善	《 13》

## 1. はじめに

### 1. 1 当社の概要とアパレル業

当社は、昭和43年より広島県の東部福山市において、地域のお客様に各種情報システムサービスを提供している。

コンピュータ専用ビルを持ち、「富士通 GS8800/10Q」、「富士通 GS8800/8N」を設置し、「アウトソーシングサービス」から、「ソフトウェア開発」、「情報機器・サプライ販売」、「情報機器保守サービス」、「アパレルCAD」、「地積情報処理」など、コンピュータに関するあらゆるサービスを提供している。

この地域は古くより、アパレル業が盛んな地域であり、現在もワーキングウエア、ジーンズの日本を代表する産地である。しかし、アパレル業の現状は、「消費の低迷」「商品の低価格化」「多品種小ロット化」「量販店への対応」「海外生産」と大変厳しい状態にあり、「生産効率のアップ」「経費の節減」を行いながら、「詳細な製品管理」が必要となってきた。

当社でも、アパレル業界用にクライアント・サーバ版の「アパレル販売管理システム」を開発し、多くのお客様に利用して頂いているが、この厳しい状況に対応する為、より導入し易い「Web版アパレル販売管理システム」を開発する事となった。

今回述べるのは、「Web版アパレル販売管理システム」構築の為に開発した、「イベントリアル通信を可能とするサーバ通信手法」についてである。

## 2. Webアプリケーションの問題点

従来の一般的な通信手法で開発したWebアプリケーションの場合、販売管理等の基幹業務システムには下記のような点で不向きである。

- ・ 送信ボタンをクリックしないと入力内容のチェックをしない。
- ・ マウスを使用しないといけない。ファンクションキーの使用が出来ない。
- ・ クライアント側にビジネスロジックを持ってくる必要があるため入力レスポンスが悪い。
- ・ ユーザインターフェースとビジネスロジックが分離されてなく、メンテナンス性が悪い。

これらの問題を解決する為、今回、「イベントリアル通信を可能とするサーバ通信手法」を開発し、その手法を使用した「Web版アパレル販売管理システム」を開発することとなった。

### **3. 「イベントリアル通信を可能とするサーバ通信手法」とは**

「イベントリアル通信を可能とするサーバ通信手法」の開発目的は、基幹業務に使用できる使いやすいWebシステムの開発を可能にすることである。

ブラウザの画面上で、発生したイベントを、通信モジュールがリアルタイムに、サーバに伝え、その内容に対応した答えをクライアントのブラウザに返し、その内容を画面に表示する。

#### **3. 1 「イベントリアル通信を可能とするサーバ通信手法」の機能目標**

##### **3. 1. 1 入力項目のリアルチェック**

入力項目にデータが入力された時点でリアルタイムにチェックを行い、エラーの表示、入力項目に対応する名称表示を行うこと。

##### **3. 1. 2 基幹業務に使用できる操作性**

マウス無しでも全ての操作ができること。

ファンクションキーに「更新」「メニュー」等の機能を振付けることができること。

##### **3. 1. 3 基幹業務に使用できる快適なレスポンス**

クライアント・サーバ間の伝送に独自のフォーマットを使用して、快適なレスポンスを実現すること。

初画面表示以外では画面全体の再表示を無くし（変更データのみ表示し）、快適なレスポンスを実現すること。

##### **3. 1. 4 既存のネットワーク環境で使用可能**

ファイアーウォール・プロキシサーバの変更無しで使用可能なこと。

SSLでの暗号化に対応できること。

##### **3. 1. 5 システムのメンテナンス性の向上**

ユーザインターフェースとビジネスロジックを分離して、メンテナンス性を向上させること。

## 4. 開発の実施状況

### 4.1 基本構造

イベントリアル通信の仕組みとしては、**図1**基本構造図の様に、Webサーバ内に、画面情報のHTML（Hyper Text Markup Language※1）とビジネスロジックのサーバスクリプトが、クライアント内には、通信モジュールとスクリプトが存在する。

#### 4.1.1 通信モジュール

通信モジュールはJavaApplet（※2）で作成され、WebサーバとHTTP通信（Hyper Text Transfer Protocol※3）を行い、クライアントで発生したイベントをリアルタイムに通知する。当初の計画では、サーバとは同期通信を行う予定であった。又、サーバスクリプトからのメッセージによってクライアントの画面制御も行う。ブラウザとのインタフェースにはDHTML(Dynamic Hyper Text Markup Language※4)を使用する。

#### 4.1.2 サーバスクリプト

サーバスクリプトはIIS+ASPで作成され、クライアント画面の制御メッセージを作成して返信する。この部分がビジネスロジックでとなる。

#### 4.1.3 HTML

クライアント画面は、HTMLで作成し、ビジネスロジックとは、完全に分離される。

#### 4.1.4 送受信メッセージ

送受信メッセージは、独自のフォーマットを採用しサイズを縮小することによりナローバンドへの対応を行う。

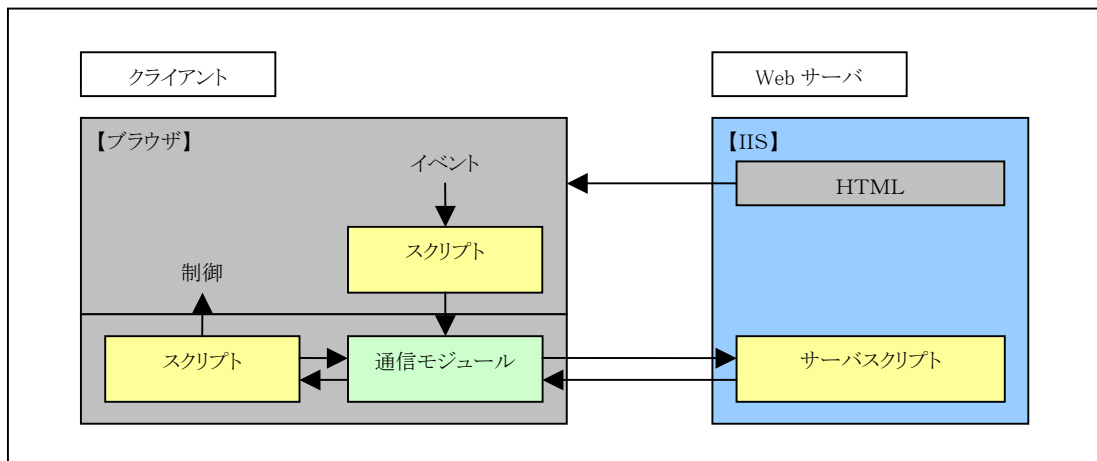


図1 基本構造図

- ※1 HTML : Hyper Text Markup Language. Web ページを記述するための言語
- ※2 JavaApplet : ブラウザ上でダウンロード・実行される Java アプリケーション.
- ※3 HTTP : Hyper Text Transfer Protocol. Web サーバとクライアントが通信するために使用されるアプリケーションプロトコル.
- ※4 DHTML : Dynamic Hyper Text Markup Language. Web ページに対話性を持たせるための拡張 HTML. VBScript・JavaScript で記述する.

## 4. 2 開発状況

### 4. 2. 1 同期通信での問題点

当初、サーバとの通信には同期通信を採用したが、実際にテストを行ってみると、スムーズな入力作業が行なえなかった。これは、同期通信では、イベントに対するレスポンス処理を完了してから次のイベントを受け付ける為、ネットワークおよびサーバの負荷により、イベント発生からイベント受付状態になるまでの間隔が一定でない場合が発生し、待ち状態が発生するからであった。（**図 2** 同期通信のみ使用を参照）

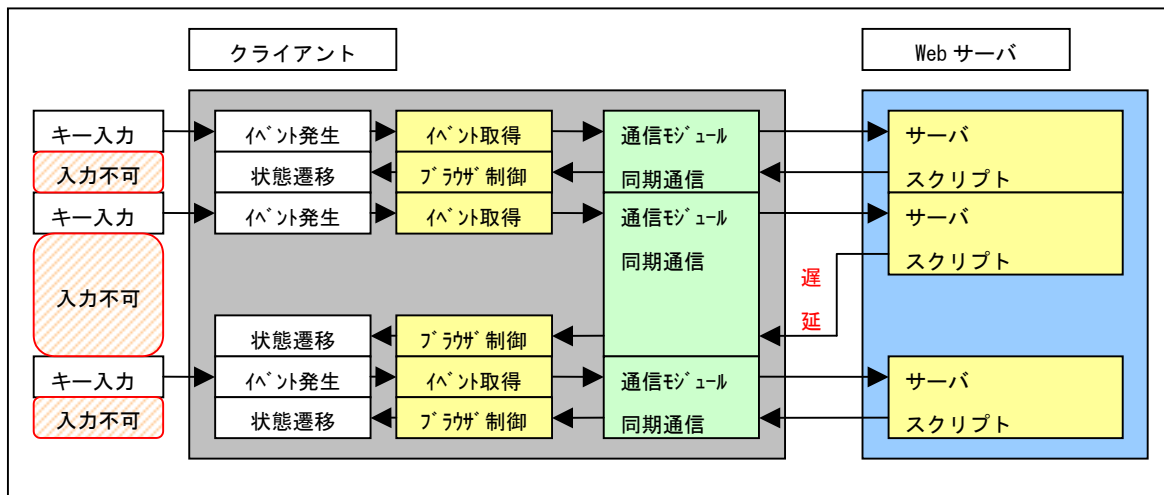


図 2 同期通信のみ使用

また、通信モジュールのメンテナンスの問題とロード時間の短縮のため、ActiveX コントロールを作成することとなった。

非同期通信を採用することにより、サーバと通信中もブラウザのイベントを受け付けられ、ネットワークやサーバの負荷の影響を受けず、スムーズな処理が可能になった。（**図 3** 非同期通信を使用を参照）

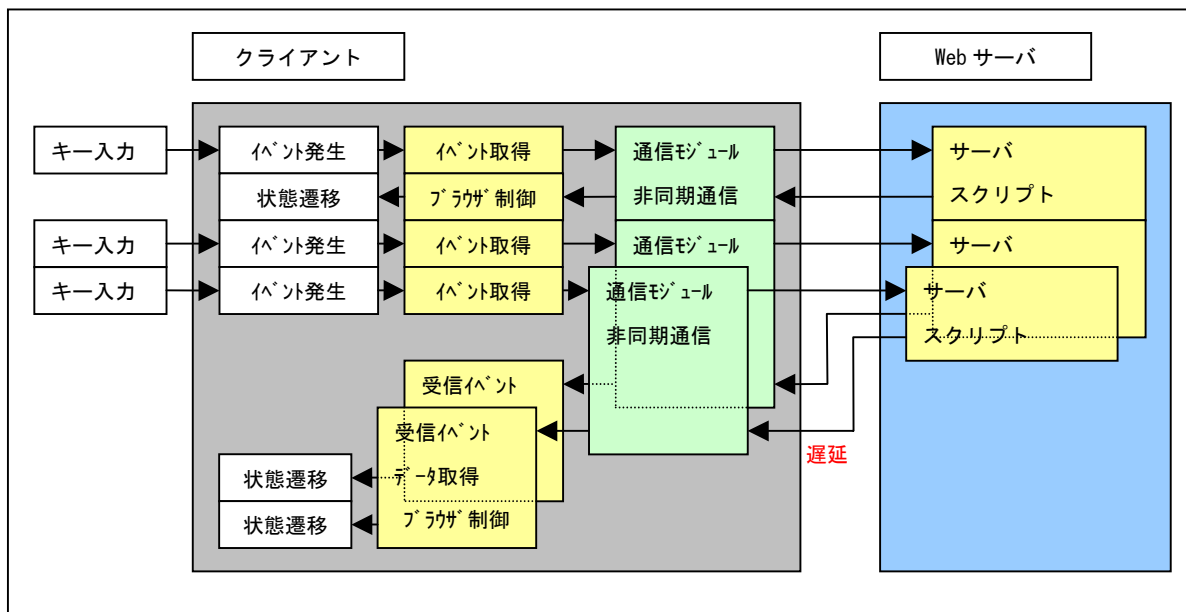


図 3 非同期通信を使用



ただし、イベント発生から画面の状態遷移までのタイムラグが大きくなる可能性があるため、入力制限を目的とする項目プロテクトの切り替えや項目表示・非表示の切り替えには非同期通信は、使用できなかった。

また、グループ移動の際には同期通信によりグループ全体のチェックを行うように変更した。

同期・非同期の切り替えはクライアントスクリプト(共通)内で自動的に行う。

#### 4. 2. 2 3KB 超えのイベントメッセージが届かない問題発生

3KB を超えるメッセージを送信するプログラムを作成したところ、サーバにメッセージが届かない障害が発生した。原因は、通信モジュールで送信メソッドに GET を使用していたためである。この問題は、POST で送信するよう変更したことで解決できた。これにより、メッセージ長の制限はなくなった。

#### 4. 2. 3 非同期通信が正しく行われない問題

通信モジュールを使用して非同期通信のテストを行ったところ、VisualBasic からの実行では正常に非同期通信を行えたが、ブラウザ上で実行するとイベントを受け付けられない現象が発生した。

非同期通信の際、通信モジュールからブラウザのスクリプトに受信イベントを発生させ、スクリプトで受信データの取得を行っていたが、ブラウザのスクリプトでは多重処理が行われないことが原因であった。(図4 非同期通信の障害を参照)

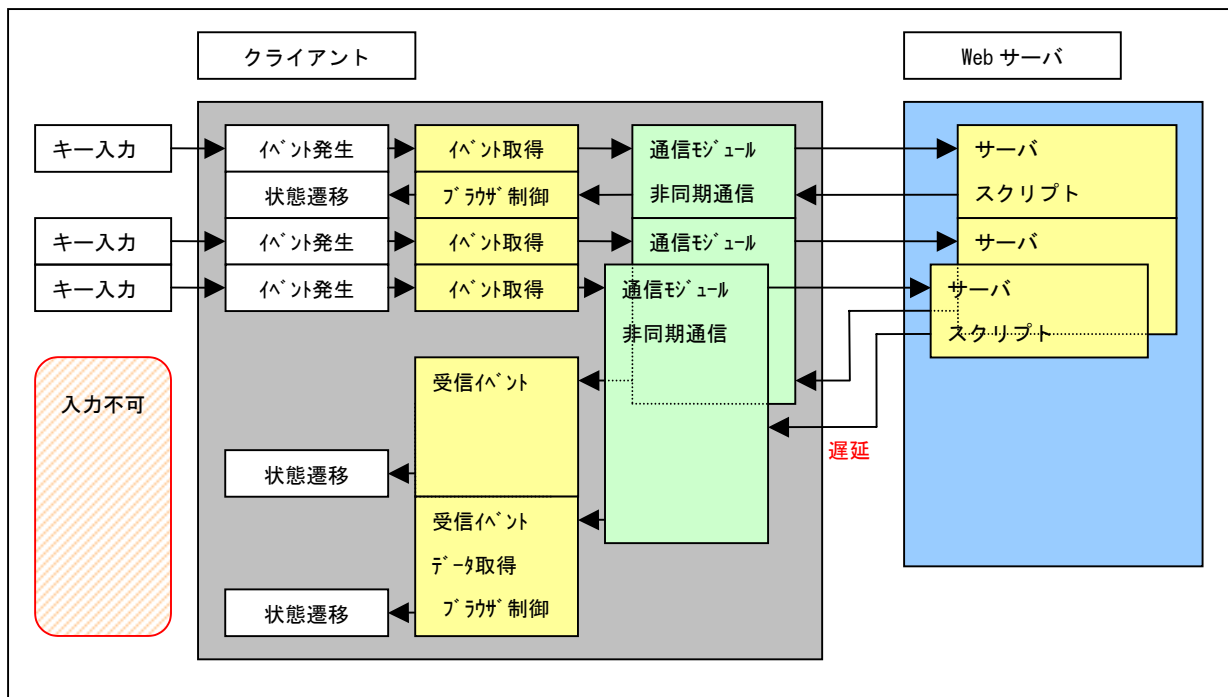


図4 非同期通信の障害

そこで、多重処理の行える ActiveX(※5)内でデータ取得をおこない解決した。

ActiveX内の非同期通信処理については、非同期通信を行う WindowsAPI でテストしたが、動作不安定の為、使用できなかった。Inet コントロールでは POST 送信時のメッセージ長に 100KB の制限があり使用不可であった。最終的には、ソケット (WinSock) を使用することにより可能となった。(図5 ソケットを使用した非同期通信を参照)

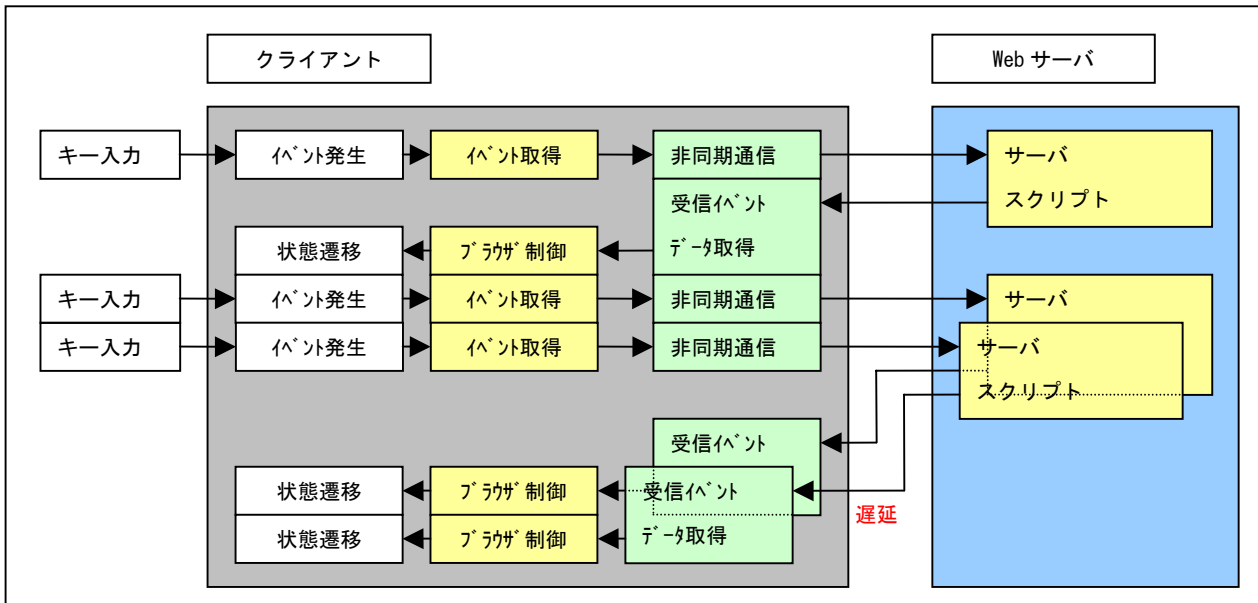


図5 ソケットを使用した非同期通信

※5 ActiveX: I E上でダウンロード・実行されるアプリケーション部品。

#### 4. 2. 4 非同期→同期通信切替時の非同期メッセージ再送処理の追加

同期通信が行われる際に未完の非同期メッセージが残っていた場合、同期処理終了後に非同期メッセージのレスポンスが到達する障害が発生した。(図6 レスポンスの到達順不整合を参照)

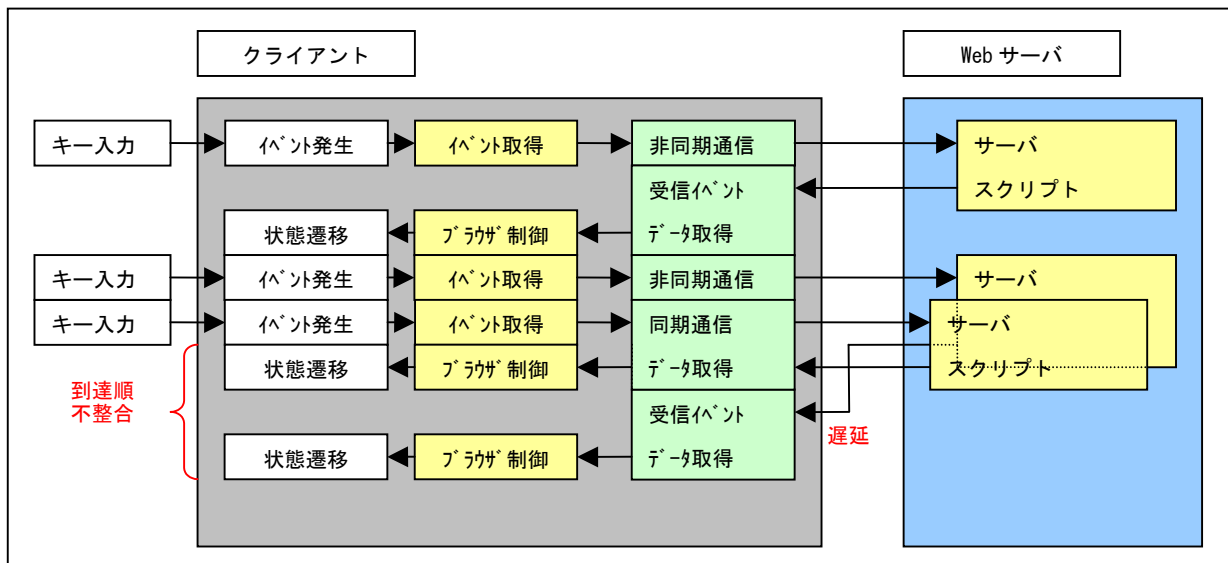


図6 レスポンスの到達順不整合

同期通信要求時に未完の非同期通信をキャンセルし、同期通信メッセージに含めることによりレスポンスメッセージの到達順不整合を無くした。(図7 未完メッセージ再送処理を参照)

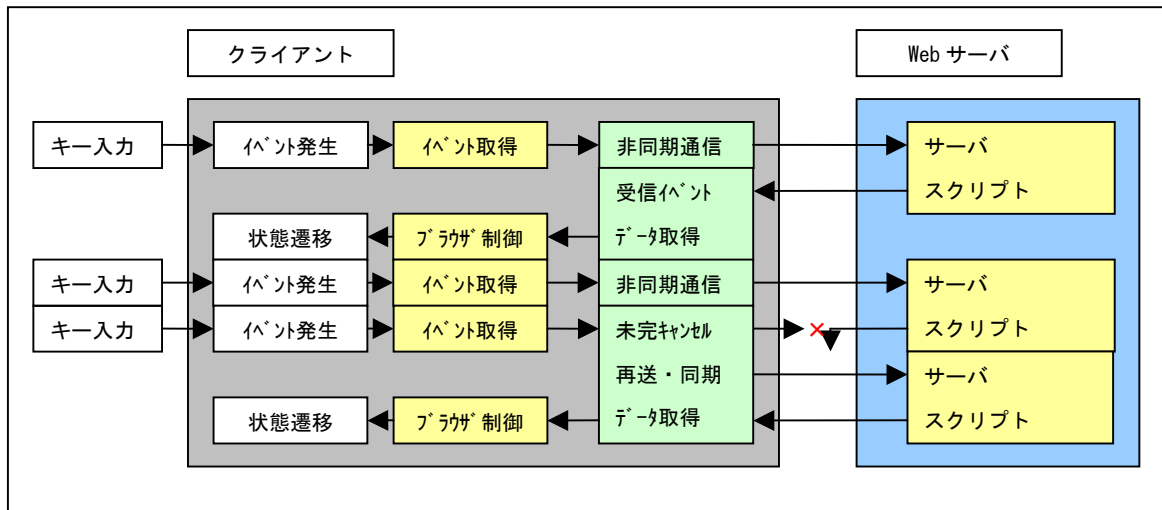


図7 未完メッセージ再送処理

#### 4. 2. 5 別の処理画面を開く際の問題

アパレル販売管理パッケージを開発していると、処理画面から別の処理画面を開く機能が必要となった。

各画面の入力値はサーバ側のセッション領域に保存して管理しているが、ブラウザから新規ブラウザを開くと同一セッションを参照するため、同じ項目名を持つ別の画面を同時に開くとセッションの保存領域が干渉する問題が発生した。(図8 セッション領域を参照)

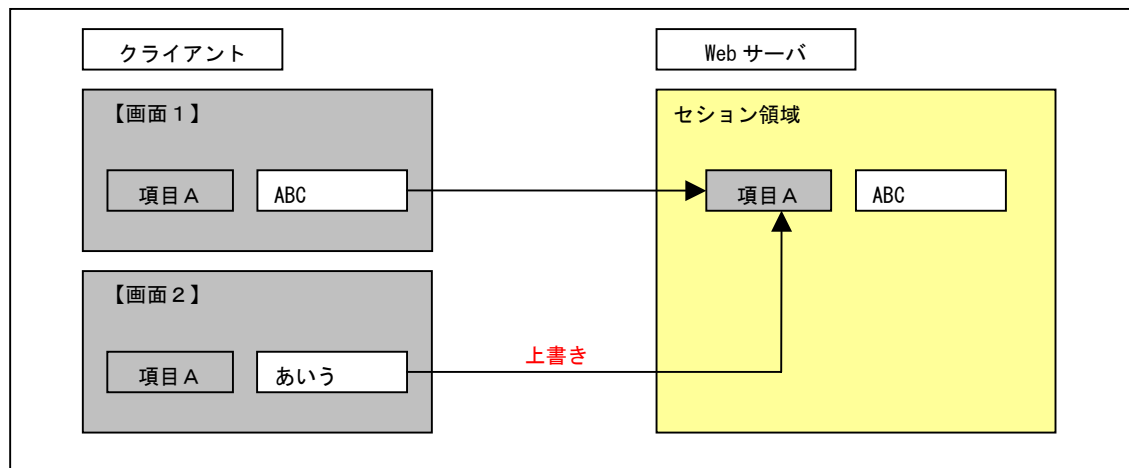


図8 セッション領域

この問題を解決するため、通信コントロール起動時にセッションNo.を生成し、セッション領域内を画面毎に分割する機能を追加した。(図9 セッション領域の分割)

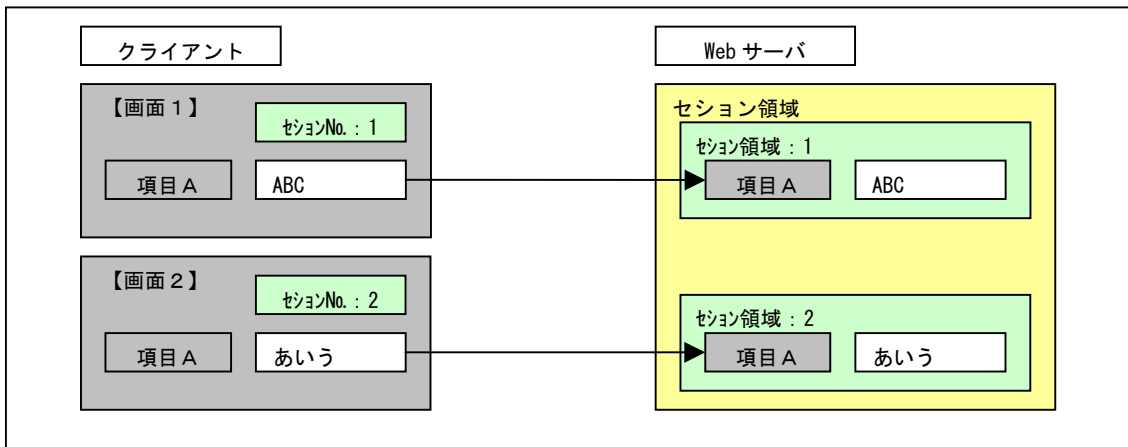


図9 セッション領域の分割

#### 4. 2. 6 同期通信非同期処理の追加

同期通信処理中は画面をプロテクトし処理終了後に入力を受け付けるため、イベントメッセージの送信後、サーバから制御メッセージを全て受信した後に制御処理を行っていた。

(図10 同期通信処理を参照)

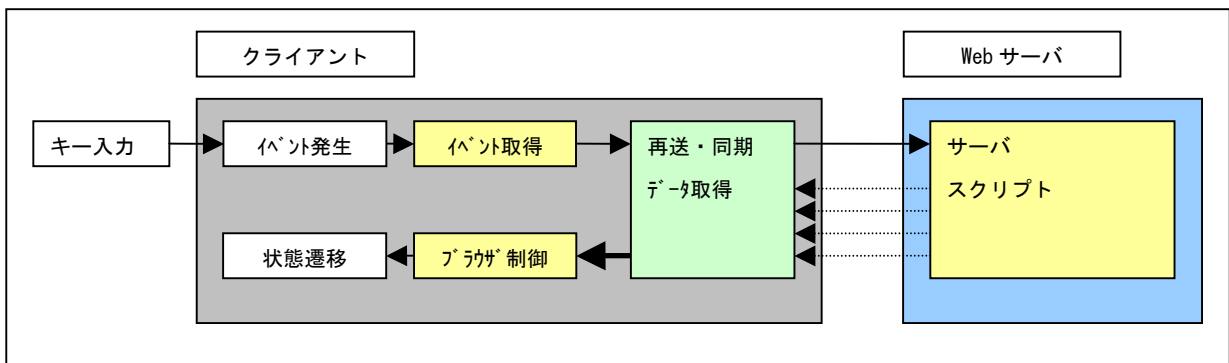


図10 同期通信処理

画面プロテクト中に画面の表示を更新したいとの要望があり、制御メッセージを受信したもののから即時実行する機能を追加した。(図11 同期通信非同期処理を参照)

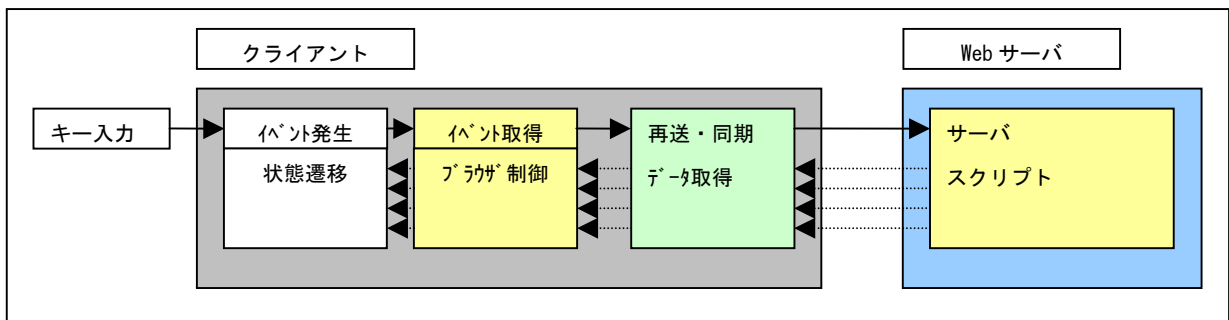


図11 同期通信非同期処理

#### 4. 2. 7 レスポンスの改善

高性能クライアントでは、問題無かったが、多少性能の落ちるクライアントでは、レスポンスが悪い現象が発生した。

調査の結果、以下のことが原因であることが確認できた。

- ・ ブラウザの表示更新処理が非常に遅い。
- ・ プロパティ変更を行った場合、プロパティの変更毎に表示更新処理が発生する。
- ・ グループ処理ではグループ内の項目をまとめて処理するため処理時間が長い。

これに対し、グループ処理メッセージ取得時にブラウザの表示状態の HTML を取得し、取得した HTML に対して更新を行い、ブラウザに戻すことにより、編集処理の高速化および表示更新処理回数を減らすことができた。(図 1 2 レスポンス改善を参照)

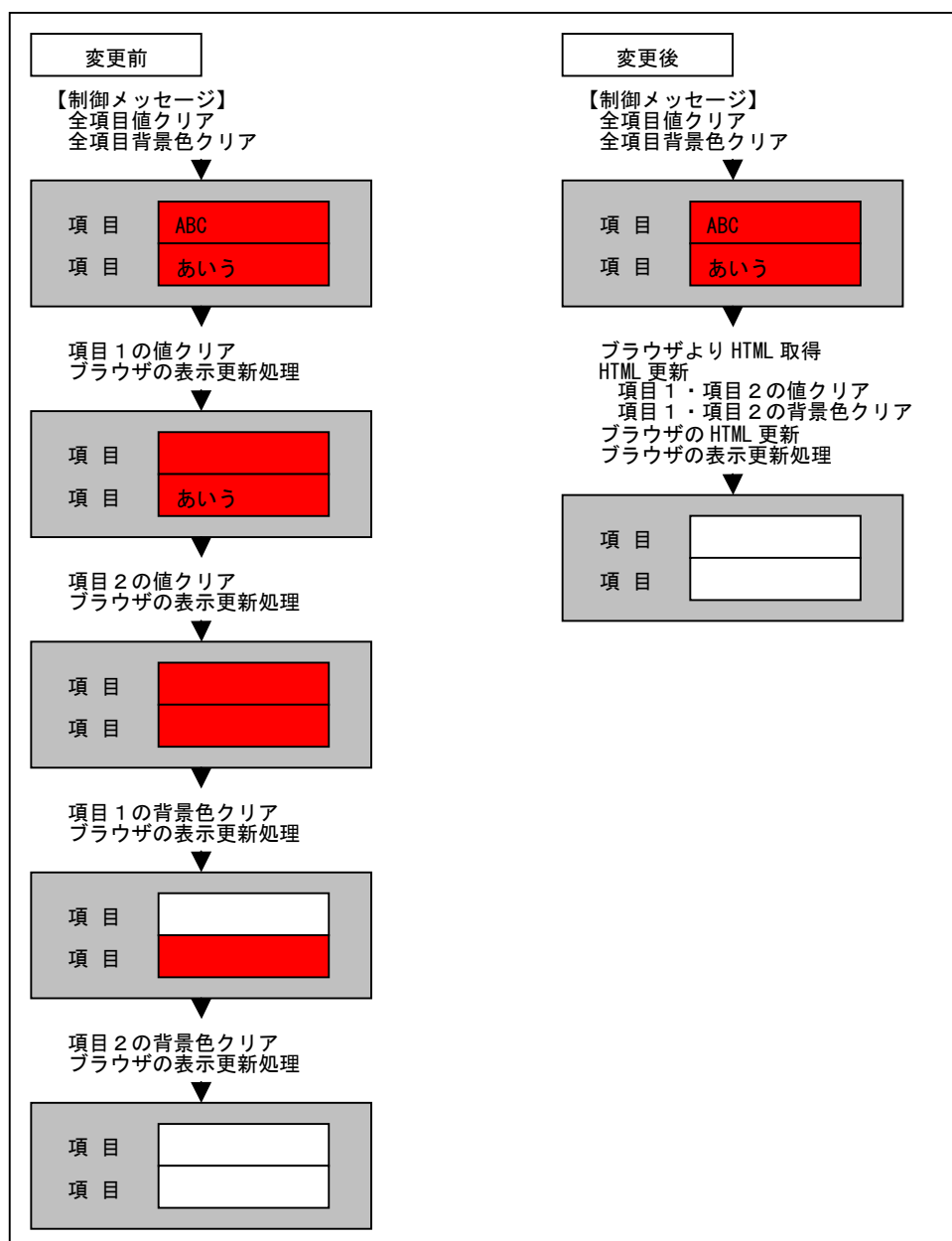


図 1 2 レスポンス改善

#### **4. 2. 8 その他の実現された機能**

a. IME の自動切り替え.

入力項目毎に日本語入力モードの ON・OFF を自動的に切り替えられるようにした.

b. 全ファンクションキーのアプリケーションでの使用.

ファンクションキー (1~12) の標準機能 (OS・IE の機能) を無効にし, アプリケーションから機能キーとして使用できるようにした.

c. Enter キーによる項目移動.

入力業務を円滑にするため, Enter キーによるカーソル移動ができるようにした. これにより, 標準機能である入力フォームの送信機能を無効にし, サーバへの誤送信がなくなった. また, ファンクションキーとの組み合わせにより, マウスを使用せずキーボードだけの操作が可能となった.

d. ダイアログウィンドウでのデータ検索.

通常では, 子画面を開いた際, 親画面にフォーカスを移すと親画面に子画面が隠れて操作できない状態が発生するが, 子画面を開いた際に親画面を操作しても子画面が隠れないようにした.

### **5. 開発を終わって**

今回のシステム開発では, 思ったようにレスポンスが出ないとか, 動きがスムーズでないとか, 併行的に開発を行っていたパッケージ開発チームより, 各種の機能要求がでたりして, 当初の目標に達成するまでには, かなり苦勞をしたが, 最終的には当初の目標を達成する事ができた.

#### **5. 1 今後について**

今回開発した, 「イベントリアル通信を可能とするサーバ通信手法」を使用した「Web 版アパレル販売管理システム」については, 当社の ASP (アプリケーションサービスプロバイダ) センターにて, ASP サービスをおこない, 多くのアパレル業のお客様に提供していく計画である.

また, 今回開発した, 「イベントリアル通信を可能とするサーバ通信手法」を使用して, 食品業界向けの Web 版販売管理システムも開発し, アパレル業と同様に, ASP サービスで提供していきたいと思っている.