

複数 PC 間データ自動同期ソフトウェアの開発 —進化する同期ソフトウェア『Progressync』—

1. 背景

ここ数年間、年間の PC の出荷台数は毎年 1,000 万台強となっています。このことからわかるように、1 家や 1 人に 1 台の PC ではなく、デスクトップ PC やノート PC、自宅と会社といったように、1 人が 2 台以上の PC を持つ時代になってきたと言えます。

このような便利な時代において、新たな問題になるのが PC 間のデータの不整合です。データファイルに限らず、ブラウザのブックマークやメーラのメールデータなど、データの不整合は複数 PC を使う人にとって、非常に面倒な問題になりつつあります。

2. 目的

背景で述べた PC 間の不整合を解決するために、新たに同期ソフトウェアを開発することを目的としました。

既存の同期ソフトウェアを使用することによっても、PC 間の不整合を解決することができます。しかし、既存のソフトウェアを調査した結果、以下のような問題点があることがわかりました。

- ・ 3 台以上の PC に対応していない

既存のソフトウェアの多くは 2 台の PC の同期には対応しているものの、3 台以上の PC の同期には対応していません。そのため、3 台以上の PC 間で同期をとりたい場合は、他の方法を考える必要が生じます。

- ・ Windows 専用

現在の Windows の圧倒的なシェアを考えると商業的には致し方ないと思われますが、現実問題として PC の OS は Windows だけとは限りません。

- ・ 対応ソフトウェアが限定されている

例えば、ブラウザのブックマークを同期したいと思っても、既存のソフトウェアでは「Internet Explorer」にしか対応していないことがほとんどです。そのため、同期ソフトウェアのせいで使用するソフトウェアが制限されたり、ユーザーにとっては使えない機能になってしまったりすることが考えられます。

このような既存の同期ソフトウェアの問題点を克服した新しいソフトウェアを開発することを、本プロジェクトの目的としました。

3. 開発の内容

既存のソフトウェアの問題点を解決する新しい同期ソフトウェア『Progressync』を開発しました。

プログラミング言語には JAVA を使用しました。そのため、JAVA 1.4 以上の実行環境

が必要となりますが、OS は特に問いません。

動作方法ですが、まず、動作対象のコンピュータやアイテムを指定するために「パートナーシップ」というものを定義します。これは XML ファイルで記述します。

次に初期化处理を行います。同期処理では過去の状態との比較が必要となるため、過去の状態が取得できない状態では同期が行えません。そのため、最初に初期化处理を行い、最初の状態を保存する必要があります。

初期化处理として、3つの方法を用意しました。

1. マスター上書き

ある1つのアイテムを指定し、そのアイテムで強制的に他のアイテムを上書き保存した上で、その状態を保存します。

2. 和

各アイテムの論理和を保存します。名前が同じアイテムが存在した場合は、競合解決処理(ユーザーが選択)を行います。

3. そのまま

各アイテムの状態をそのまま保存します。そのため、各アイテムは一致せず、同じ状態にはなりません。今後の更新処理に対してのみ同期処理が行われます。

初期化处理終了後に、同期処理が行えるようになります。以降、ボタンをクリックするだけで、複数の PC の環境が同じになります。

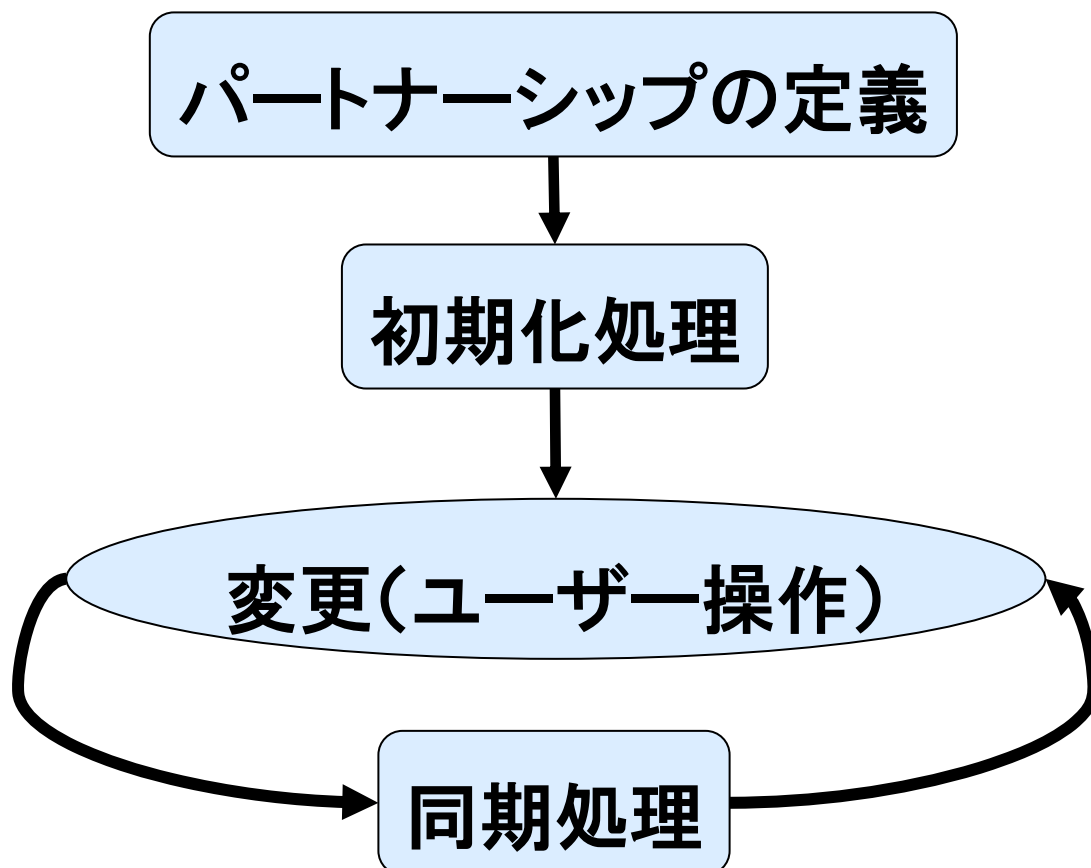


図1 同期処理

4. 従来の技術(または機能)との相違

既存のソフトウェアと差別化できる要素として、以下の特徴があります。

- ・マルチプラットフォーム

開発時点で、Windows だけでなく Macintosh や Linux 上でも動作することを目標としていました。マルチプラットフォームを実現する技術としては、いろいろな手段が考えられると思いますが、プログラミングのしやすさなどを考慮し、JAVA を使用しました。

- ・3 台以上の PC に対応

設計の段階から、n 台での同期を考え、開発を進めてきました。パフォーマンスに関しましては、本プロジェクトでは妥協し、あくまで開発目標を実現することにフォーカスしました。

- ・プラグイン形式

プラグインによって対応ソフトウェアを増やすことができるようにしました。開発言語に JAVA を使用することを決定したため、プラグインに関しても JAVA で作成することとしました。プラグインは、基本となるクラスを用意し、それを継承したクラスを作成することで、簡単に作れるようにしました。プラグイン作成者は、ソフトウェアに対応したデータの読み書き部分を記述すればよく、同期のアルゴリズムなどに関しては特に考慮する必要はありません。

また、プラグインを利用する部分に関しましては、JAVA のリフレクションの機能を使用しました。リフレクションを使用することによってパフォーマンスの懸念がありましたが、動作テストを見ている限り、まったく気にならないレベルでした。

5. 期待される効果

ユーザーは、このソフトウェアを使うことによって、いつでもどこでも同じ環境で PC を操作することができるようになります。特にモバイルユーザーにとって、このソフトは必須のものになると期待しております。

6. 普及(または活用)の見通し

一般的なユーザー、特にモバイルユーザーに関しては便利なソフトウェアとして使ってもらいたいと考えております。また、開発者の方には、本ソフトウェアの特徴でもあるプラグインを積極的に開発して欲しいと考えています。そのために、本ソフトウェアをオープンソースにすることや、ドキュメントを充実させることを考えています。

7. 開発者名(所属)

安達 宜隆(東京大学大学院 工学系研究科)