

DSU：分散ソフトウェア自動更新システムの開発

6 J-7

多田 宏一, 岡本 隆司, 齋藤 正史

三菱電機 (株) 情報電子研究所

1 はじめに

ワークステーション (WS) の普及により、1人1台のコンピュータ資源を利用することが可能となった。さらに、WSの高性能化、機種多様化により、大規模で複雑な相互依存関係を持つ異機種分散システムが構築されるようになった。

このような中で、ソフトウェアの更新は、人間がWS1台ごとに行っていたが、近年、WSを相互に結ぶネットワークの発達に伴い、この更新作業をネットワークを介して自動的にかつ定期的に行なうソフトウェア自動更新システムが開発されてきた。

われわれは、そのようなシステムの1つで、分散システムの多数のWSに対するソフトウェアの配布及び更新を自動的に実行する、分散ソフトウェア自動更新システム (DSU) を提案した [1]。

本論文では、DSUの詳細設計、実運用やシステムの大规模化に対応するために盛り込んだ機能、その実現方法とそれによってもたらされた効果について述べる。

2 DSU システムの概要

ソフトウェアの更新は、1度だけではなく、定期的な何度も生じるため、この作業には、多量のマンパワーと時間を必要とする。したがって、ソフトウェア自動更新システムの果たす役割は大きい。

そこで私達は、次のような特徴を具備したDSUの提案を行なった。

- ファイルサーバの分散
- 異機種環境のサポート
- 複数のソフトウェアバージョンの配布と、インストール位置の柔軟性の実現

次に、DSUのシステム構成を簡単に説明する。

DSUは、ソフトウェアの転送を行なう Distributed Software Update Server (DSUS)、サーバ機能を持つWSのホスト名と保有しているソフトウェア名を一元管理する Server Manager (SM)、ソフトウェア名を指定して更新要求を行なう Distributed Software Update Client (DSUC) から構成されている。

その動作を図1をもとに説明する。

1. DSUCは、SMに対してソフトウェア名を指定してサーバであるDSUSのホスト名を問い合わせる

2. SMからソフトウェアのサーバであるDSUSのホスト名を入手する
3. 入手したホスト名のDSUSへ接続する
4. ソフトウェアの転送を受ける

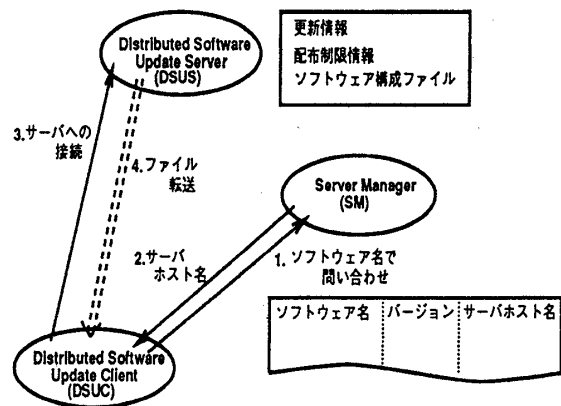


図 1: DSU の動作

3 DSU の実装

今回は、BSD UNIX¹ と System V UNIX 上で動作し、サーバ・クライアント形式でソフトウェアの配布及び更新を行なう PDS の Coda²[2] をベースに開発を行なうことにした。

Coda採用の理由としては、Codaが、サーバ・クライアント形式を採用していることと、現在Codaを利用しているユーザがスムーズにDSUに移行できるというDSUの普及性を考慮した。

実際には、サーバ機能を持つWSのホスト名と保有しているソフトウェア名を一元管理するSMを新規に作成するとともに、SMとCodaのサーバ・クライアントの3者間のインタフェースを修正することによって、DSUを実現した。

また、Codaを調査した結果、ライセンス機構、複数のソフトウェアバージョンの配布、及びソフトウェア転送後のインストールに関して、機能的に不十分であると判断し、以下に示すような機能を付加した。

DSU: An Implementation of Distributed Software Update system
Kouichi TADA, Takashi OKAMOTO, Masashi SAITO
Mitsubishi Electric Corp.

¹UNIXオペレーティングシステムは、UNIXシステムラボラトリー社が開発し、ライセンスしています。

²Coda: code distribution aid

1. ソフトウェア配布の制限

ライセンス契約に従って、ソフトウェアの配布や更新を、WS や CPU 単位で制限することが必要である。

しかし、Coda は、そのための明示的な制限パラメータを持っていないため、これを付加し、ソフトウェアライセンス及び配布制限機構を強化した。

この実現のためには、ソフトウェアの配布制限情報をソフトウェア管理情報の一部として、DSUS に持たせた。配布制限の情報は、以下の2つである。

- 再配布許可情報

パラメータを `redist` とし、指定の方法は、`yes` または `no` を指定する。`yes` ならば、再配布が許可され、`no` ならば、再配布は行なわれない。

指定例

```
redist      = yes ;
```

- 再配布許可ホスト名

パラメータを `ok` とし、指定の方法は、更新が許可されるホスト名を列挙するか、全てのホストを意味する `_ALL` を指定する。ホスト名が指定されると、更新はそのホスト名の集団にのみ許可され、`_ALL` ならば、全てのホスト名が、更新対象となる。

指定例

```
ok          = host-A, host-B ;
```

```
ok          = _ALL ;
```

なお、この機能は、テストバージョンの試使用にも応用できる。

2. バージョンを指定した更新

Coda では、クライアントの保有できるソフトウェアのバージョンは、次の2通りだけであった。

- 現在のバージョンに固定
- 常に最新のバージョンに更新

この場合、最新のバージョンに不具合があり、ファイルに上書きをしてしまった時の復帰要求や、研究者によっては、自分の望むバージョンを使用したいという要求に答えられなかった。

これに対し DSU では、以下のようにして、複数のソフトウェアバージョンの配布を実現し、この問題を解決した。

サーバホスト名とソフトウェア名を一元管理する SM に、バージョン番号も管理させる。

クライアントである DSUC には、オプション (-V) で所望のバージョンを指定させる。オプションには、バージョン番号による指定と常に最新を指定する `y(yes)` の指定を許可した。

コマンド入力例

```
dsuc -S ソフトウェア名 -V バージョン番号
```

```
dsuc -S ソフトウェア名 -V y
```

3. 更新後のコマンド実行

ソフトウェアのインストールの場合には、アーカイブファイル (tar ファイル) の展開や、インストールスクリプトの実行が必要である。

また、各クライアントの作業環境を尊重する場合には、単なるファイルやディレクトリの転送だけでは不十分であり、ファイルの移動や、名前の変更が必要である。

そこで今回の開発では、これまでファイルの転送や更新時に人間が入力していたコマンドを自動的に実行するための機構を付加した。

具体的には次のように実現した。サーバ WS にコマンドスクリプトファイル (DSUscript と呼ぶ) を持たせ、その中に、転送後に実行されるコマンドをシェルスクリプトの形で記述する。このファイルは、ソフトウェアの更新とともに、クライアント WS に転送される。クライアント WS 側では、オプション (-C) の指定により、そのコマンドスクリプトファイルの自動実行と、転送のみの別を指定可能にした。

実行指定

```
dsuc -C -S ソフトウェア名 -V バージョン番号
```

4 おわりに

異なる機種から構成される、大規模な分散システムにおいて、ソフトウェアの配布や更新を自動的にこなす DSU について述べた。

現在は試使用を行っており、実運用に向けて改良を重ねている。

さらに、その利用範囲をソフトウェアの配布や更新のみではなく、修正や拡張などの管理維持作業を必要とする、データベースの更新にも適用することを考えており、今後はまず、弊所内の電子メールサーバ WS 上のアドレスデータ更新への適用を予定している。

参考文献

- [1] 落合 真一, 齋藤 正史
「DSU: 分散ソフトウェアの自動更新システム」
第 41 回 (平成 2 年後期) 全国大会講演論文集 (分冊 5) 情報処理学会
- [2] “coda - Code distribution aid” and
“codaserver - Code distribution aid daemon”,
UNIX User's Manual Reference Guide