

分散協調作業環境における WebDAV を用いた分散的バージョン管理手法の提案\*

2X-5

竹山 純平† 亀山 渉†  
早稲田大学大学院国際情報通信研究科 †

1.はじめに

ネットワークの発達、PCの普及やその性能の向上などにより、将来的に「分散環境上での協調作業」が普及し、複数の人がネットワークを通じて一つのコンテンツを、共同で制作するという作業形態が一般的になると考えられる。本稿では上記のような作業形態に対応するために、コンテンツを、その構成要素単位でバージョン管理することが可能な「分散的バージョン管理手法」を提案する。

2.分散協調作業環境におけるバージョン管理

2.1 想定される作業環境

「分散環境上での協調作業」では、次のような特徴が考えられる。(図1参照)

複数の制作者の参加

一つのコンテンツは、複数の制作者が、各人が特定の範囲を担当する、といった状況の下で作成される。

複数の構成要素からのコンテンツの生成

一つのコンテンツは、各制作者がそれぞれ担当した構成要素（テキストや画像など）によって構成される。また、それぞれの構成要素自体も、複数の制作者によって作成されたりする場合が考えられる。

コンテンツ間での構成要素の共有

コンテンツの構成要素は、必ずしも、ある特定のコンテンツの一部である必要はなく、異なるコンテンツが同一の構成要素を利用する、といったことも可能である。

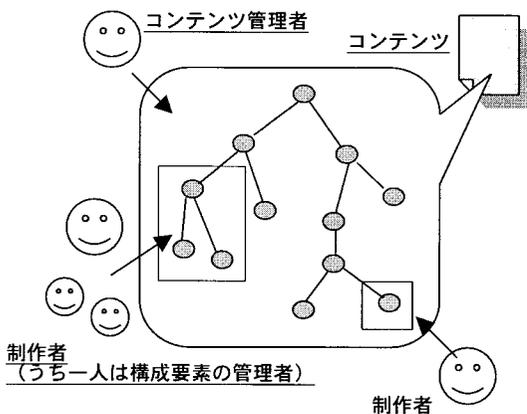


図1 分散的協調作業の例

協調作業における各制作者の権限

複数の制作者がコンテンツ、もしくは構成要素の制作にあたる場合、管理者とそれに従属する制作者によって制作がなされる場合と、制作にあたるメンバー全員が管理者権限を持つ場合、またそれらが複合した場合が考えられる。(図2参照)

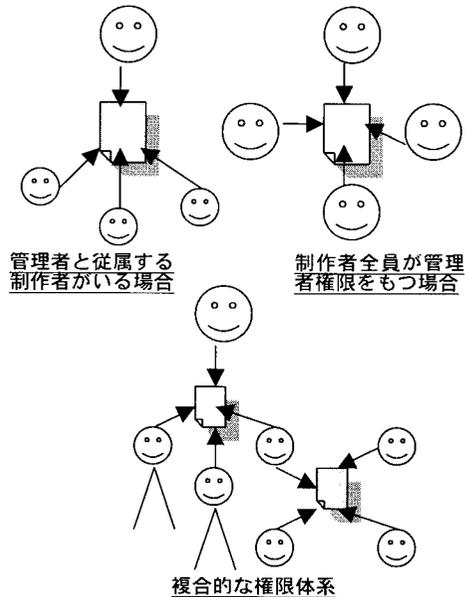


図2 分散協調作業環境における権限体系の図

2.2 提案する手法

このような作業環境では、コンテンツのバージョンをコンテンツ単位でなく、その構成要素単位で管理することによって、各構成要素の制作者達は、それぞれの担当する部分の制作に専念することができ、また、目的や用途に応じて構成要素ごとに最適なバージョンを選択し、その組み合わせによってコンテンツを生成することが出来るので、以前に作成された構成要素の再利用による有効活用が出来、またコンテンツのアップデートも効率的に行うことができるというメリットがあると考えられる。

しかし、従来のバージョン管理システム[1][2]には、前述したような、コンテンツを構成要素単位でバージョン管理するという機能を備えていない。そこで本稿では、IETFにおいて標準化が進められている WebDAV[2]及び、W3C で標準化が進められている XPointer[3]を用いて、コンテンツのバージョンをその構成要素ごとに管理し、必要に応じてコンテンツを構成要素単位でアップデートすることでコンテンツの生成を行うことが出来る「分散的バージョン管理手法」について検討した。

\*A dispersive versioning system for distributed cooperative working environment using WebDAV

† Jumpei Takeyama, Wataru Kameyama

† Graduate School of Global Information and Telecommunication Studies, Waseda University

### 2.3 ワークフローの概略

本システムでの具体的な作業の流れは、以下の通りである。(図3参照)

- (1) コンテンツ本体を XML 形式で作成する。
- (2) コンテンツを構成する各部品は、それぞれの制作者がサーバーにアップロードし、アップロードされると同時に、サーバー側でコンテンツ自体を「最新」バージョンに更新する。
- (3) バージョンの操作をする場合、クライアント側の制作者は、コンテンツのある特定のノード(もしくはコンテンツ全体)にアクセスし、そのコンテンツのノードに最適な、構成要素のバージョンを選び、コンテンツの更新を行う。
- (4) コンテンツ及びの各構成要素ごとのバージョンと各ノードにおける更新履歴をデータベースで管理する。

この手法により、コンテンツの各構成要素のバージョンを、コンテンツそのものから独立して管理することが可能となる。

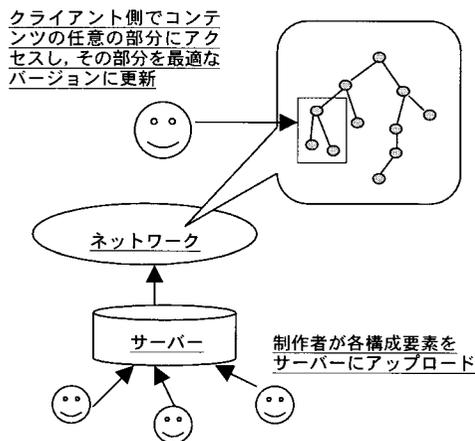


図3 ワークフローの概略図

### 3. システム構成

このシステムは、「クライアント側の制作者が、扱うコンテンツの文書構造について既知である」という前提で実現される。

- (1) クライアントは、バージョンの更新を行いたい部分を指定し、それがサーバーに XPointer によるコンテンツのノードの指定としてリクエストされる。
- (2) サーバーはリクエストを受けると、対象となるコンテンツを構文解析し、クライアントからのリクエストによって指定されたノードに WebDAV の LOCK メソッドを適用し(図4参照)、排他制御を行った上で、そのノードにおける利用可能なバージョンの一覧をクライアントに通知する。
- (3) クライアントは、通知されたバージョン一覧の中から最適なバージョンを選択し、サーバーに更新をリクエストする。

- (4) サーバーは WebDAV の UNLOCK メソッドで、先程適用された LOCK を解除し、COPY メソッドを用いてコンテンツのバックアップをとった上で、クライアントから指定されたノードの更新を行う。
- (5) サーバーはコンテンツの各バージョンおよび一連の更新履歴について、それぞれを対応するデータベースに格納する。

本システムの概略は上記の通りであるが、このうちバージョンの選択と更新機能の実装方法については、各ノードにおける利用可能なバージョンを XLink [4] の拡張リンク機能を用いて表現し、その中から選択された最適なバージョンをコンテンツと結びつけて更新を行うという手法及び、現在 IETF において策定中の DeltaV [5] (WebDAV へのバージョン管理機能の拡張) を実装し、その機能をコンテンツの任意のノードごとに適用するという手法の、二通りの実装方法を検討している。

```
LOCK /~foo/foo.xml#xpointer(/foo/chapter[1]) HTTP/1.1
Host: localhost
Timeout: Infinite, Second=4100000000
Content-Type: text/xml; charset="utf-8"
Content-Length: 272

<?xml version="1.0" encoding="utf-8" ?>
<D:lockinfo xmlns:D="DAV:">
<D:lockscope><D:exclusive/></D:lockscope>
<D:locktype><D:write/></D:locktype>
<D:owner>
<D:href>http://www.foo.or.jp/~foo/foo.html</D:href>
</D:owner>
</D:lockinfo>
```

図4 XPointer によって特定のノードを LOCK する例

### 4. まとめと今後の課題

本稿にて、分散協調作業環境において作成されたコンテンツのバージョン管理の手法として、コンテンツをその構成要素ごとにバージョン管理する「分散的バージョン管理手法」の提案を行った。今後は、今回提案した手法を実装し、コンテンツのアップデートを行う際のアクセス管理および、本システムに用いるユーザーインターフェースや、コンテンツの更新履歴の検索手法などについても検討を行い、分散協調作業環境におけるコンテンツ制作システムの核となる部分を構築していく予定である。

### 参考文献

- [1] Concurrent Versions System:  
<http://www.cvshome.org/>
- [2] Y. Goland, E. Whitehead, A. Faizi, S. Carter, D. Jensen: "HTTP Extensions for Distributed Authoring -- WEBDAV", RFC2518, February 1999
- [3] XML Pointer Language (XPointer) Version 1.0:  
<http://www.w3.org/TR/xptr/>
- [4] XML Linking Language (XLink) Version 1.0:  
<http://www.w3.org/TR/xlink/>
- [5] G. Clemm, J. Amsden, T. Ellison, C. Kaler, J. Whitehead: "Versioning Extensions to WEBDAV", Internet-Draft draft-ietf-deltav-versioning-16, July 2001