WWW コンテンツ一貫性維持のためのリンク更新機構の提案

A Link Updating Mechanism for Integrity Maintenance of WWW Contents

中溝 昌佳[1] 森嶋 厚行[2] 有山 智洋[3]
杉本 重雄[4] 北川 博之[5]

Akiyoshi NAKAMIZO Atsuyuki MORISHIMA
Tomohiro ARIYAMA Shigeo SUGIMOTO
Hiroyuki KITAGAWA

近年，WWW は社会における重要なメディアのひとつとして大きな役割を果たしている．WWW の特徴としては，分散管理，動的な更新，リンクなどがある．これらの特徴は WWW を役立つメディアとする一方で，コンテンツの一貫性管理を困難にする要因になっている．本稿では，データベースにおける一貫性制約の考え方を WWW の文脈に導入し、WWW コンテンツにおける一貫性維持を行うための機構を提案する．特に WWW コンテンツにおけるリンクの一貫性維持を行うための機構について提案する．

The World Wide Web (WWW) has become one of the most important media in our society. Its characteristics include distributed management, dynamic updates, and links. The characteristics are not only making the WWW a useful tool, but also making it difficult to manage the integrity of its contents. This paper proposes a method that applies to the WWW 's context the concept of integrity constraints, which is common in database contexts. We focus on the problem of managing the integrity of WWW links.

1. はじめに

近年, WWW は社会における重要なメディアのひとつとして大きな役割を果たしている．WWW の特徴としては，分散管理, 動的な更新, リンクなどがある. これらの特徴は WWW を役立つメディアとする一方で，コンテンツの一貫性管理を困難にする要因になっている．コンテンツに一貫性が無い具体例としては，例えば次のようなものがある．(1)同じ組織の情報を保持する複数のページで電話番号が異なっている.(例えば，片方では 0298-59-abcd であり，もう片方では 029-859-abcd など)．これは，一方の情報だけが更新されたことによって生じる.(2)リンクをたどると，その参照先のページが存在しない(リンク切れの問題)．(3)特定のニュースを参照するために，ニュースページへのリンクを張ったとする．その後，しばらくたつとそのニュースの情報はバックナンバーページへ移動しまい，そのリンクの参照先が意図したものと異なってしまう．

ここでは, 以上のような問題をなくすための仕組みを Web コンテンツの一貫性管理と呼ぶ．データベースの分野では一般に，一貫性の管理を行うためにデータベースが満たすべき制約(一貫性制約[1])を記述するというアプローチがとられる．本稿ではこの考え方を WWW の文脈に導入する．すなわち，WWW コンテンツで成立すべき制約を記述し，それを用いて一貫性管理を行うための機構を提案する．特に，リンク切れやリンク先の内容の変更という，リンクの一貫性管理の問題に焦点を当てる．1999 年の調査[2]によると，Web サイトの平均的なリンク切れの割合は 5.7%である．また，科学教育のための Web ページを調査したところ，20 ヶ月後には全リンクの 18.8%が切れていたという報告もある[6]. ジョージア工科大の GVU センターにおける調査では，約 6 割のユーザが「リンク切れは WWW の利用における重大な問題の一つである」と答えており[3]，リンクの一貫性を維持することは重要な問題であると考えられる．分散管理という WWW の性質上, 一貫性を完全に保証することは困難であると考えられるが，本プロジェクトでは，可能な限り一貫性を満たすよう WWW が自律的にリンクの更新を行う世界の実現を目指す.

関連システム・研究としては次のようなものがある．まず，各種のリンク切れ発見ツール(リンクチェッカ)が存在する[7][8]．これらは，指定されたページ(群)に記述されたリンクについて，参照先がリンク切れ等でないかチェックし，レポートを作成するものである．リンクチェッカを用いてリンク切れを発見すると, Web サイト管理者にメールを送ることによってリンクの更新管理を行っているサイト[9]も存在する．また，ハイパーテキスト DB における動的リンクの研究[4]や，ハイパーメディアデータベース等のコンテキストで，参照経路の一貫性の研究[5]が行われている. しかし我々の知る限り, WWW コンテンツのリンク一貫性維持の自動化の問題に取り組んだ研究は存在しない.

本研究の特徴は次の通りである．(1)通常の Web アーキテクチャの自然な拡張であり，既存の Web コンテンツとも容易に組合せ可能である. (2)単純かつ強力な制約記述言語を提供する．本論文の構成は次の通りである．2 章では，WWW コンテンツの一貫性を管理するためのフレームワークを提案する．3 章では，WWW におけるリンクに関して成立する制約を記述するための言語である，WIDL(Web Integrity Description Language)/Link の説明を行う．また，制約の一種を表現するために，特別な WWW ページである Link Authority の概念を導入する．4 章では，WIDL によって記述された制約を可能な限りみたすようリンク更新を動的に行う機構の提案を行う. 5 章では, Link Authority Discovery Engine (LADE) を用いた Link Authority の探索について議論する.

2. WIM Server を用いた WWW コンテンツ一貫性管理

我々が提案するフレームワークの重要な構成要素は，WIDL(Web Integrity Description Language)/Link および WIM(Web Integrity Management) Server である．これらは，Web Server とファイルシステムに格納された Web ペー

[1] 学生会員 芝浦工業大学大学院工学研究科
m103198@sic.shibaura-it.ac.jp
[2] 正会員 筑波大学知的コミュニティ基盤研究センター
mori@sils.tsukuba.ac.jp
[3] 図書館情報大学大学院情報メディア研究科
ariyama@ulis.ac.jp
[4] 筑波大学知的コミュニティ基盤研究センター
sugimoto@slis.tsukuba.ac.jp
[5] 正会員 筑波大学電子･情報工学系
kitagawa@is.tsukuba.ac.jp

ジ群，という通常の Web アーキテクチャに追加する形式で利用する(図 1)．これは従来のアーキテクチャの自然な拡張であるため，既存の Web コンテンツとも容易に組合せ可能である．WIDL/Link は Web コンテンツにおけるリンクに関する制約を記述するための言語である．詳細は 3 章で説明するが，WIDL/Link では HTML ページや XML ページの各リンク要素の属性として制約を記述する．WIM Server は，管理下の Web コンテンツを監視し，WIDL/Link によって記述された制約のうち Web コンテンツ変更などの理由により満たされなくなったものを発見すると，その制約を満たすように Web コンテンツを更新する(実際に変更先の探索を行うのは，WIM Server に組み込まれた WIM Engine である)．例えば，あるリンクに対して「リンク切れがあってはならない」という制約が WIDL/Link で記述されていたとする．その場合，WIM Server がそのリンクに対してリンク切れを検出すると，代わりとなるリンク(変更先リンク)の候補群を求める．その際，WIM Server は各変更先リンク候補に対してスコアを計算する．後述するが，WIDL/Link では，スコアに関する閾値を指定することによって，WIM Server に対して変更先リンクの候補群を求めるだけでなくリンクの自動更新を指示することができる．

3. 制約記述言語 WIDL/Link

WIDL(Web Integrity Description Language)/Link は，HTML ページもしくは XML ページ(以下 Web ページ)で記述されるリンクに関する制約を記述するための言語である．リンクの制約の記述は，Web ページのリンクに WIDL/Link で規定された属性を追加することによって行われる．次に WIDL/Link を用いた制約記述の例を示す．

```
<A wi:matches="*XLearner*"
href="http://www.mlab.org/news">ニュース</>
```

ここで，wi は WIDL/Link の名前空間であり，matches=*e* はリンク先のページの内容がパターン *e* にマッチすることを表す制約である．したがって，この例の制約はリンクが指す先のページに「XLearner」という文が含まれていることを表している．

WIDL/Link では 4 つの属性を定義している．図 2 にそれらの一覧を示す．属性は 2 種類に分類される．第一のグループ(isAlive, matches, follows)は制約を表すための属性である．残りの 1 つ(threshold)はリンクが制約を満たさなくなった場合，システムが適切なページを探した後，どの基準で自動的にリンクを更新するかを指定する閾値である．

3.1 isAlive

属性 isAlive は,「リンク切れでない」という制約を表す．次に例を示す．

```
<A wi:isAlive
href="http://ho-expo.org/">東大宮博覧会</>
```

isAlive が成立しない状況とは，ページが移動した場合や，ページが無くなった場合などがある．したがって，isAlive が条件として指定されているとき，システムはリンク切れを検知するとそのリンクが制約を満たさなくなったと判断する．

3.2 matches

先に説明したとおり，matches は「リンク先のページが

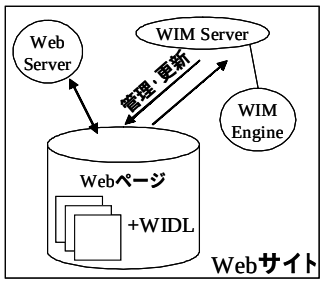

図 1 アーキテクチャ

属性	意味
isAlive	リンク切れでない
matches	リンク先が指定の内容とマッチする
follows	リンクが指定の Link Authority に従う．
threshold	自動的にリンクを更新するための閾値

図 2 WIDL/Link で用意される属性

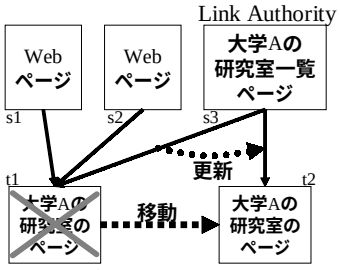

図 3 Link Authority

特定の内容にマッチする」という制約を表す．これは次のように，関数と組合せて利用することもできる．

```
<A wi:matches=snapshot()
href="http://www.mlab.org/">素敵な日々</>
```

ここで，snapshot()は，WIM Server がこの制約記述を初めて発見した時点での，リンク先のページの内容を文字列としてそのまま返す関数である．したがって，この例のように制約が指定されているとき，システムはリンク先の内容に少しでも変更が加えられると，リンクが制約が満たされなくなったと判断する．

3.3 follows

WIDL/Link では, Link Authority という概念を導入する．ある Web ページ *p* は次の条件を満たすとき，別の Web ページ *q* に関する Link Authority であると言う．(1)*p* が *q* へのリンクを持っており，かつ(2)*q* が *q0* に移動すると，必ず *p* 中の *q* へのリンクは *q0* へのリンクに変更される．例を図 3 に示す．まず，ある大学 A の研究室の Web ページ t1 が存在したとする．t1 は，複数のページ s1, s2, s3 からリンクされている．これらのうち，s3 は大学 A においてその研究室が所属する学科の研究室一覧ページである．このとき，一般には s3 は t1 に関する Link Authority である．したがって，例えば次のような状況が生じる. t1 が t2 に移動したとする．すると, t1 を参照しているページではリンク切れが存在するが，通常 s3 だけは t1 へのリンクを t2 に張り替えるはずである．このとき，例えば s3 のアドレスが http://l.s.ac.jp/lab である場合は s1 や s2 のリンクに次のように制約を記述することができる．

```
<A wi:follows="http://l.s.ac.jp/lab"
href="http://m.l.s.ac.jp/">某研究室</>
```

この例のように制約が指定された場合，follows による制約が満たされなくなるのは，指定された Link Authority において t1 へのリンクが変更されるか，もしくは Link Authority そのものが存在しなくなった場合である．

一般に，あるページに関する Link Authority は複数存在し，それぞれ異なる(暗黙の)制約を表す．例えば，上の例における t1 に関する Link Authority として，t1 の研究室の教員 B が学生時代に所属していた研究室の OB リンク集ページ s4 があるとする．そのとき，s4 は(きちんとメンテナンスされていれば)「教員 B が運営する研究室」に関する Link Authority である．一方，s3 は「大学 A に所属する研究室」に関する Link Authority である．このように，Link Authority を適切に選択する事により，多様な(暗黙の)制約を記述することができる．

3.4 threshold

属性 threshold は自動的にリンクを更新するための閾値を指定する．閾値は 0 以上 1 以下もしくは”infinity”で記述する．例を次に示す．

```
<A wi:isAlive wi:threshold="1"
href="http://toyosu-it.ac.jp/">豊洲工大</>
```

指定された閾値を x とする．この場合，システムが発見した変更先リンクのスコアが x 以上の時のみ，WIM Server はリンクをその変更先リンクに自動的に書き換える．スコアが 1 になる場合の例としては，ページのリダイレクトが存在した場合がある．閾値以上の変更先候補が存在しない場合には，変更先のリンク候補のランキングページを作成し，そこへのリンクに変更する．

3.5 一般的な規則

WIDL/Link では以上の属性を組み合わせて制約を記述する．ただし，次の規則がある．

- デフォルトで isAlive の指定が存在
- デフォルトで threshold=" infinity "の指定が存在

したがって，次の二つの例は同じである．

```
<A href="http://toyosu-it.ac.jp/">豊洲工大</>

<A wi:isAlive wi:threshold="infinity"
href="http://toyosu-it.ac.jp/">豊洲工大</>
```

このようなデフォルトの制約の存在によって，既存の WWW コンテンツであっても，WIM Server の機能を利用可能である．

4. 制約を満たすリンクの探索

4.1 最上位アルゴリズム

WIDL/Link が記述可能な制約集合 C = { isAlive, follows, matches}の要素の組合せ間の関係を，順序集合 ($P(C)$, ⊂)のハッセ図(図 4(a))として示す(ただし $P(C)$は Cのべき集合)．上位のものほど厳しい制約である．WIM Server はあるリンク lに関する制約の組み合わせ C_l（⊂C）が満たされなくなっていることを発見すると，次のような手順でリンクの探索を行う．(1) 順序集合 ($P(C_l)$, ⊂)のハッセ図 G を作成する．これは図 4(a)の部分グラフとなる．例えば，isAlive と follows が指定されている場合は図 4(b)となる．また必ず最大元と最小元が存在する．(2) G の最大元から順に，その条件を満たす変更先リンクの探索を行う．したがって発見された変更先リンクには，そのリンクが満たす制約に対応する(G 中の)ノードが存在する．各ページのスコアは G のトポロジカルソートに矛盾しない順に割り当てる．特にスコア 1 の候補は，必ず G の最大元に対応する制約を満たすページに割り当てる．(3) 変更先リンクの候補をスコア順に並べ，1 列のリンク列を作成する．

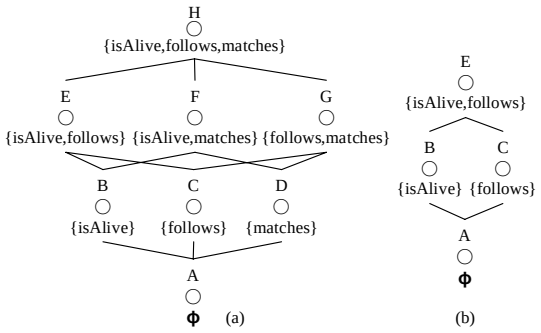

図 4 制約の組合せ間の関係(a)とその部分グラフ

実際に制約を満たさなくなったリンクが与えられたとき，本アルゴリズムはその制約のタイプ(A,B 等)に応じてそれぞれ処理を行う．

4.2 各場合における処理

以下では図 4 における E,B,A に対応する制約の組合せが満たされなかった場合の処理の概要を説明する．重要なポイントは，どの場合においても，制約を満たす範囲で「以前指していたページの変更先を探す」という暗黙の仮定があることである．したがって，各処理では制約を満たすページというだけでなく，この点を考慮して変更リンク候補の選択が行われる．各処理は下位のノードに対応する処理を呼び出すことがある．その際，上位ノードで出力したページのスコアの最小値を超えないようスコアの上限に制限が加えられていく．

E の場合：これは，isAlive および follows の指定が行われている場合である．この制約が満たされない時，満たされない条件によって 5 つの場合に分ける事ができる．以下で，C1 は「isAlive が満たされている」C2 は「follows の指定先である Link Authority が存在する」C3 は「指定された Link Authority とリンクが矛盾しない」ということとする．

(1) $C1 \wedge \neg C2$ ：処理 1
(2) $C1 \wedge C2 \wedge \neg C3$ ：処理 2
(3) $\neg C1 \wedge C2 \wedge C3$ ：処理 3
(4) $\neg C1 \wedge \neg C2$ ：処理 1
(5) $\neg C1 \wedge C2 \wedge \neg C3$ ：処理 2

[処理1] Link Authority が移動したと仮定し移動先の探索を行う．移動先の探索については 5 章で説明する LADE を利用する．移動先の候補が見つかった場合，それを Link Authority とみなして制約の再評価を行う．見つからなかった場合には，isAlive の真偽に応じて，現在のリンクを候補とするかもしくは場合 B の処理に移る．

[処理2] 指定された Link Authority で，旧リンクが新たなリンクに変更されていれば，それを変更先リンク候補とする．Link Authority は存在するが, 旧リンクの代わりとなる新たなリンクが見つからない場合は処理 1 を行う．

[処理3] 現在指している(制約を満たさない)リンクを変更先候補とすると同時に，場合 B の処理に移る．

B の場合：リンク切れを起こすと制約を満たさなくなったと判断され，システムは現在アクセス可能なページであり，かつ変更先であると考えられるページを探索する．具体的には，

次のヒューリスティクスに基づいて変更先のリンクを探索する．(1)同じ Web ページは，時間が近いものほど内容が似ている傾向にある．(2)Web ページが移動するとき，同じサイト内で移動する可能性が高い．(3)Web ページの移動先がリダイレクトされている場合，移動先のページの URL がわかる．(4)Web 検索エンジンなどで逆引きすると，Link Authority を発見できる可能性がある．

具体的には次の処理を行う．まず，リンクが切れた場合に備えて，システムはリンク先のページを定期的にキャッシュする．リンクが切れた場合，次の処理を行い変更先リンクを探索する．(a)リダイレクト先が保存されている場合，そのページにリンクを書き換える．(b)リダイレクト先が保存されていない場合，次の二つの処理を行う．(b1)検索エンジンなどを用いてリンクの逆引き検索を行う．結果のページの中から，変更先のリンクが存在しないか調べる．(b2)キャッシュしたページに類似した内容を持つページの探索を行う．まず同じ Web サイト内のページを探索し，それでも見つからなければ，検索エンジンを用いて Web 全体からの検索を行う．

A の場合：この場合，現在アクセス可能なページでなくても良いので，WIM Server がキャッシュしているページ，もしくは Internet Archive や各種検索エンジンなどに格納されているページを変更先リンク候補とする．

5. Link Authority Discovery Engine

3 章で説明した Link Authority は単純かつ強力な概念であるが，属性 `follows` の指定を行うためには，あらかじめどのページが Link Authority であるかを知っている必要がある．そこで我々は Link Authority Discovery Engine (LADE)を提案する。これは，多数の Web ページの中から，指定されたページの Link Authority を発見するためのソフトウェアである．通常の Web ページ検索エンジンと同じように，他のソフトウェアや利用者からネットワークを通じて利用可能となるように設置され，提案フレームワークを間接的に支援するために利用される．例えば，(1)Web サイト管理者が WIDL/Link による制約を記述する際の参考にする．(2)Link Authority に指定していたページが移動してしまった際に新しい Link Authority を探す，などの目的に利用する．また，将来的には `follows` 属性の自動作成などに利用することも想定している．

LADE はページを発見するという意味では通常の Web ページ検索エンジンに似ているが，ページ選択の基準が全く異なるため，内部的には全く異なるアルゴリズムを実装する必要がある．例えば，次のような評価法が考えられる．(1)あるページへのリンクを持つページを長期的に観察し，リンク先ページの移動に従ってすぐに更新された場合，候補として高く評価する．(2)あるページへのリンクが `follows` 属性を持つことを発見すれば，指定されているページを候補として高く評価する．(3)リンクのメンテナンスがよく行われている(常にリンク切れが一定の割合以下の)Web サイトを発見し，そのサイトのページは全て候補として高く評価する．

6. おわりに

本稿では，一貫性制約の考え方を WWW の文脈に導入し，WWW コンテンツのリンク構造に関する一貫性管理を行うための手法を提案した．提案手法は既存の Web アーキテクチャの自然な拡張であり，また，単純かつ強力な制約記述言語を提供する．今後の課題としては，更新リンク候補のランキング手法のより詳細な検討，提案手法の実装と評価，それに基づくアルゴリズムの改良などがあげられる．

[謝辞]

ゼミなどでご議論いただきました筑波大学図書館情報学系の田畑孝一教授と阪口哲男助教授に感謝いたします．本研究の一部は文部科学省科学研究費補助金若手研究(B)(課題番号 15700108)による．

[文献]

[1] S. Abiteboul, R. Hull, V. Vianu: Foundations of Databases. Addison-Wesley 1995.

[2] All Things Web. State of the Web Survey. http://www.pantos.org/atw/35654.html

[3] Georgia Institute of Technology GVU Center. GVU's 8th WWW User Survey. http://www.gvu.gatech.edu/user surveys/survey-1997-10/.

[4] K. Tanaka, N. Nishikawa, S. Hirayama, and K. Nanba : Query Pairs As Hypertext Links. In Proceedings of 7th IEEE Data Engineering Conference, p 456-463, 1991.

[5] Eitetsu Oomoto, Youichi Shima: Integrity Constraints for Reference Links in Hypermedia Database Systems. CODAS 1996: 182-185.

[6] Science Education Broken Links: http:// www-class.unl.edu/biochem/url/broken links.html

[7] Xenu's Link Sleuth (TM): http://home.snafu.de/ tilman/xenulink.html

[8] Link Check with LinkAlarm : http://linkalarm.com/

[9] Planet SOSIG - A spring-clean for SOSIG: a systematic approach to collection management : http://www.ariadne.ac.uk/issue33/planet-sosig/

中溝 昌佳 Akiyoshi NAKAMIZO

芝浦工業大学大学院工学研究科修士課程在学中. WWW を利用した情報システム, XML, データベースなどに興味を持つ. 日本データベース学会学生会員.

森嶋 厚行 Atsuyuki MORISHIMA

筑波大学知的コミュニティ基盤研究センター(図書館情報学系)助教授. 1998 年 筑波大学大学院工学研究科修了. 博士(工学). 情報統合，XML とデータベース，Web の一貫性管理，Web アーカイブなどに興味を持つ. ACM, IEEE-CS, 情報処理学会, 電子情報通信学会, 日本データベース学会各正会員.

有山 智洋 Tomohiro ARIYAMA

図書館情報大学大学院情報メディア研究科在学中. WWW を利用した情報探索，XML，メタデータなどに興味を持つ.

杉本 重雄 Shigeo SUGIMOTO

筑波大学知的コミュニティ基盤研究センター(図書館情報学系)教授. 京都大学工学部情報工学科, 同大学院工学研究科情報工学専攻博士後期課程修了．工学博士．Digital Library, 特にメタデータに関心を持つ．ACM，IEEE CS，情報処理学会他会員.

北川 博之 Hiroyuki KITAGAWA

筑波大学電子・情報工学系教授．1980 年東京大学大学院理学系研究科修了．理学博士．異種情報源統合，文書データベース，WWW の高度利用等の研究に従事．著書「データベースシステム」(昭晃堂) 等. ACM, IEEE-CS, 情報処理学会，電子情報通信学会，日本ソフトウェア科学会，日本データベース学会各正会員.