

アクティブルールを用いた Web ページの動的再構成

清光 英成[†] 竹内 淳記^{††} 田中 克己^{††}

Web ページやページ間のリンクは閲覧者がどのようにアクセスしても変化しない静的なデータである。そのため、閲覧者個別のページ構成やリンク構造をページ作者が実現するためには Web サーバアプリケーションや CGI プログラミングの知識を必要である。本論文では、アクティブルールを用いた Web ページの動的再構成機構により、ページ作者が宣言的にルールを記述することで閲覧者個別のページ内容やリンク構造を動的に生成するアイデアを提案する。さらに、アクティブデータベースの ECA ルールと比較して、アクティブ Web システムのルール記述と処理の特徴を議論する。

Dynamic Web Page Reconfiguration Based on Active Rules

HIDENARI KIYOMITSU,[†] ATSUNORI TAKEUCHI^{††}
and KATSUMI TANAKA^{††}

In this paper, we propose an idea for reconfiguring web pages based on an accession to the pages and user's browsing history. The major objective of the proposed mechanism is the reflection of a web page author's intention about his web data. A reconfiguration rule for providing an appropriate web content including hyperlinks is described in a declarative manner, which the author can define easily. The ECA rule and our reconfiguration rule look apparently alike, but they have some differences. And more, we propose an implementation method for the web page reconfiguration using a XML expression.

1. はじめに

インターネット技術の長足の発達により、提供されるデータは爆発的な増加を続け、即時的な情報収集が可能となっている。また、Web ページを作成するためのソフトウェアが普及し、ページ作成者が JavaScript や DynamicHTML の自動生成機能を用いて変化のあるページを特別な知識なしで作成できる環境が整いつつある。

一方、ページ作成者が各閲覧者に個別のページ構成やリンク構造を提供したいという要求も高まっている。例えば、ページ作者のサイトを頻繁に訪れる閲覧者に対して、他の閲覧者と異なる情報を提供することで、閲覧者の差別化を図るアプリケーションを構築できれば有用である。しかしながら、Web ページやページ間のリンクは閲覧者がどのようにアクセスしても変化しない静的なデータである。そのため、閲覧者が過去にアクセスしたページや巡行したリンクなどの情報に

基づいてページコンテンツを変化させようとする、内容のほとんど異なる複数の HTML ファイルをあらかじめ用意しておくか、CGI などのサーバアプリケーションを構築しなければならない。そこで、ページ内コンテンツの加工や表示体裁を変化させる機能を宣言的に定義すれば実現できるような機構があれば有用だと考えられる。

著者らはこのような問題に対して、閲覧者のリンク巡行履歴に基づいて巡行可能になるリンクを動的に制御するリンク活性化式を導入し、ページ作成者の意図をリンク機構に反映する枠組を提案した^{1),2)}。また、閲覧者のページ閲覧履歴やアクセスしている位置に基づいてページコンテンツを動的に変化させるためにコンテンツの動的再構成ルールを宣言的に記述して CGI プログラムを自動生成するアイデアを提案してきた^{3),4)}。

本論文は、アクティブルールを参考にした動的再構成ルールを用いてページコンテンツを動的に変化させるアクティブ Web システムの構築を目的とする。そのため、

- 動的再構成ルールを機能させるための手法
- XML 環境での実装法
- アクティブ Web システムに固有の性質

[†] 神戸大学経済学研究科

Graduate School of Economics, Kobe University

^{††} 神戸大学大学院自然科学研究科

Graduate School of Science and Technology, Kobe University

を議論する。

2. Web システムの特徴とアクセス権

ページ作成者の意図どおりにリンクを巡回しなければ、情報が十分に伝わらないだけでなく誤解を生じる可能性が大きいという問題がある。例えば、工作や揮発性物質の取扱いを表現する一連のページ集合では、第一回及び第二回ページを閲覧せずに、第三回ページのみを閲覧すると閲覧者を危険にさらす恐れがある。このため、ページ作成者の意図したリンク巡回を閲覧者に対してある程度強制する機構が必要であると考えられる。また逆に、作成者の意図した閲覧経路に従った閲覧者にはある種の特典を与えるという機構も Web の商業利用では重要である。

2.1 Web システムの性質

閲覧者がページにアクセスしたり、リンクを巡回することを考えると、Web システムには以下のような性質がある。

- ページの URL を知っていればページにアクセス可能
- 設定されているリンクを巡回すれば無条件にリンク先ページへアクセス可能
- 過去にどのようなページ閲覧やリンク巡回をしたかにかかわらずページ内容とリンク巡回権限は不変

例えば、広告ページ 1 を閲覧して top ページに戻ってきた閲覧者には特典ページ 1 へのリンクを表示し、広告ページ 2 を閲覧して top ページに戻ってきた閲覧者には特典ページ 2 へのリンクを表示したい場合、図 1 のように、top ページを三つ用意しなければならない。

そこで、元のページ内容からの変更を記述できるようなしくみをページ作成者が容易に設定できると有用であると考えられる。

例えば、図 2 のように、広告ページ 1 を閲覧して top ページに戻ってきた閲覧者には特典ページ 1 へのリンクを表示し、広告ページ 2 を閲覧して top ページに戻ってきた閲覧者には特典ページ 2 へのリンクを表示するような機構を実現できれば有用である。

2.2 アクセス権管理

Web のようなハイパーテキストシステムは読み出し専用メディアデータを扱うため、すべてのアクセスはデータベースの読み出しトランザクションと考えることができる。Web アプリケーションで扱うべきア

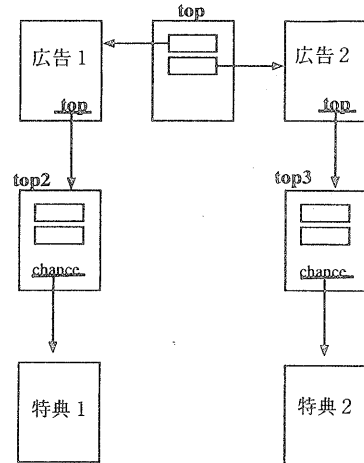


図 1 リンク先の変更

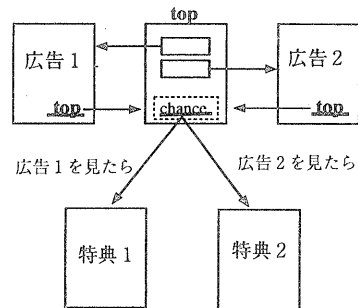


図 2 ページの再構成によるリンク先の変更

クセス権管理は以下のものがある。

- ページのアクセス権
個々のページ全体に対する閲覧権限を扱うアクセス権管理であり、アクセス権限があればすべて見ることができ、なければ見ることができないような All-or-Nothing アプローチである。システムはユーザやグループを対象にパスワード認証などからページのアクセス権限を発行する
- リンクのアクセス権
ページ作成者はページ間の関係を表現するためにリンクを設定する。リンク巡回権限は閲覧者やグループに対して与えられるが、システムはリンクの巡回が可能かどうかを判別するだけである。
- コンテンツのアクセス権
ページ中の特定のコンテンツへのアクセス権限を扱うことで、ページ作者が主張する意図の違いを以下のような視覚的表現が可能である。
 - コンテンツの配置が違う

- 表示されるコンテンツ自体が違う
- 文字列中のある単語がすべてリンクアンカーになる
- アンカー文字列が画像バナーになる
- 同じアンカーをクリックしてもリンク先ページが違う

以上のような機能を実現するために本論文では、ページ作成者自身がルールを設定することでページコンテンツを動的に変化させる機構を提案する。また、リンクのアクセス権限を管理するためには、リンクアンカーの表示・非表示を制御しなければならない。つまり、コンテンツのアクセス権管理と同時に以下を考慮する必要がある。

- リンク巡回権限を与えてアンカーに適用
- 巡回権限が与えられたリンクの先のページにアクセスできる権限を発行

これらのアクセス権管理を実際的な効果を伴って実現するためには、ページデータを変化させなければならない。そこで、ページの動的再構成のためのルールをアクティブデータベースシステムのECAルールにならって記述する。また、動的再構成ルールをXML形式で表現し、表示体裁をスタイルファイルを用いて動的に変化させる手法を提案する。

3. ページの動的再構成

3.1 動的再構成の意義

ページ作成者は伝達したい意図をページの内容とリンク構造で表現し、Web上で公開している。ページ内容やページ間のリンクは静的なデータであるため、ほんの僅かな違いを表現するために異なるHTMLファイルを用意しなければならない。

ページ内容の動的再構成は閲覧者のページ閲覧履歴や過去のリンク巡回からページの表示内容を変化させる。そのため、元のコンテンツを利用してルールを記述すれば、ページ作成者の意図するコンテンツ体裁となって閲覧者に表示される。そのため、コンテンツを再利用できるので、内容のほとんどかわらないHTMLファイルを用意する必要がなくなり、Webデータの生産性を向上させることができる。

図3の左側は従来の静的なページ内容とリンク構造により作成されたページ集合とリンク構造を示している。ページ p_{41} , p_{42} , p_{43} は僅かな違いがあるだけのページであり、ページ p_{51} , p_{52} , p_{53} , p_{54} も同様である。それらにルールを適用して動的に内容を変化させることができれば、図3の右側のように、五つのページ

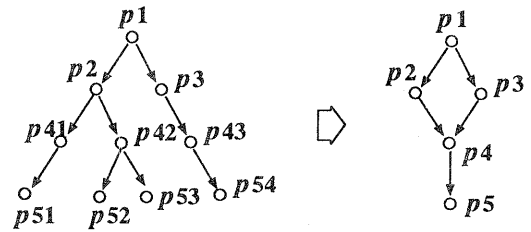


図3 ページデータの再利用

データを用意し、適用するルールを設定すればよい。CGIやAPSなどを用いて実現することも可能だが、手続き的な手法であるため、構築に専門的な知識を必要とし、システムの更新も大がかりとなる。ページ内容やリンク構造の動的再構成をルールベースで宣言的に定義することで実現できれば、更新やデバッグもルールを再構成するだけでよいと容易になると考えられる。

3.2 動的再構成ルール

動的再構成ルールは以下の記法で表現する。

`page_name:incidental,histories ⇒ function`

ルール中の各要素は、

- `page_name`
動的再構成ルールを適用するページ名
- `incidental`
アクセス時刻、位置、リンク元などのページアクセス情報
- `histories`
アクセス回数、最後にアクセスした時刻などの閲覧者履歴とサーバに蓄積されている履歴情報に対する条件
- `function`: `add`, `hide`, `replace`
ルールを満たしたときに発火するアクション

を意味する。

例えば、閲覧者がページ `default.asp` にアクセス要求をしたときに、リンク元ページが `adv1.asp` であれば、`special1.asp` へのリンクを追加し、リンク元ページが `adv2.asp` であれば、`special2.asp` へのリンクを追加するルールは以下のように記述する。

```
default.asp:(referere,adv1.asp),(),() ⇒
  add(anchor, './special1.asp', 'SpecialOffer')
default.asp:(referere,adv2.asp),(),() ⇒
  add(anchor, './special2.asp', 'SpecialOffer')
```

これは、図 2 で示したページ作者の意図を表現するルールである。

4. XML を用いた動的再構成手法

Web ページの個別化は、ASP や CGI だけで実装できるが、ページ毎にプログラムを用意しなければならないことが多く、ページの動的再構成にはこれらの専門的知識も不可欠であるような手続き的手法である。本節では、Web ページの表現を閲覧者個別に変化させる宣言的手法を提案する。

4.1 動的再構成ルールの XML 表現

動的再構成ルールを XML データとして記述する。ページ作成者は、任意のデータを ASP(Active Server Page) として記述し、ASP から XML ルールファイルとスタイルシートを呼び出す。

まず、動的再構成ルールを実現する XML の DTD 部分を以下に示す。

```
<!ELEMENT RULES (RULE*)>
<!ELEMENT RULE (EVENT,CONDITION?,ACTION)>
<!ELEMENT EVENT
  (READ,REFERER?,DATE?,LOCATION?)>
<!ELEMENT CONDITION (USER?,AGGREGATE?) >
<!ELEMENT ACTION (ADD|HIDE|REPLACE)>
<!ELEMENT READ (URL)>
<!ELEMENT REFERER (#PCDATA)>
<!ELEMENT DATE (#PCDATA)>
<!ELEMENT LOCATION (#PCDATA)>
<!ELEMENT USER (TIMES?,DATE?)>
<!ELEMENT AGGREGATE
  (CONTENT, TIMES?,REFERER?)>
<!ELEMENT ADD (LINK|IMAGE|...)>
<!ELEMENT HIDE (LINK|IMAGE|...)>
<!ELEMENT REPLACE
  ((LINK|IMAGE|...),(LINK|IMAGE|...))>
<!ELEMENT LINK ((URL,LABEL)|(URL,IMG))>
<!ELEMENT IMAGE (URL,ALT?)>
<!ATTLIST RULE
  ID (#PCDATA)
  NAME (#PCDATA)?
>
```

Web システムが扱うのは読み出し専用データであるため、閲覧者がサーバに発行できるのは読み出し要求だけである。そのためイベントには、閲覧者が読み出しを要求するページの URL、参照元、アクセス時刻・

場所を READ の子要素として型定義している。

条件部分は、アクティブデータベースシステムと同じく、定義してもしなくてもよい Option となっている。参照元も型定義しているが、イベント部分の参照元はアクセス時の参照元であり、条件部分の参照元は履歴中に存在する過去のアクセス時の参照元を示している。

アクション部分は、追加・隠蔽・置換を記述し、子要素としてリンク・画像などの Web ページに表現可能なコンテンツを記述できるようになっている。特に、置換 (replace) は、アンカー文字列を画像バナーに置き換えたりするような、メディアの変化に追従できる型定義としている。

本研究では、動的再構成ルールを XML で表現する DTD を用意するだけでなく、ページ作者がルールを宣言的に記述することで、XML 記述が生成されることも想定している。

3.2 節の例は以下の XML で表現することができる。

```
<RULES>
  <RULE ID="rule1">
    <EVENT>
      <READ>
        <URL>default.asp</URL>
      </READ>
      <REFERER>adv1.asp</REFERER>
    </EVENT>
    <ACTION><ADD>
      <LINK>
        <URL>special1.asp</URL>
        <LABEL>SpecialOffer</LABEL>
      </LINK>
    </ADD> </ACTION >
  </RULE>
  <RULE ID="rule2">
    <EVENT>
      <READ>
        <URL>default.asp</URL>
      </READ>
      <REFERER>adv2.asp</REFERER>
    </EVENT>
    <ACTION><ADD>
      <LINK>
        <URL>special2.asp</URL>
        <LABEL>SpecialOffer</LABEL>
      </LINK>
    </ADD>
  </RULE>
```

```

</LINK>
</ADD> < /ACTION >
</RULE>
</RULES>

```

二つの動的再構成ルールを記述しているが、ルールのIDはサイト中で唯一の値でなければならない。ルールのNAMEはoptionであるが、サイト中で唯一の値でなくても構わない。システムはID属性によって、ルール記述を特定し、メタデータ管理を行うが、NAME属性はページ作成者が付与して利用する。

4.2 動的再構成ルールの評価と実行

動的再構成ルールのイベントと条件はスタイルシート中で評価され、起動するアクションのテンプレートが選択される。テンプレートにしたがってサーバ側でHTMLに変換し、閲覧者に提示する。システムは、以下の工程で処理を行う。

(1) イベント検知

閲覧者がアクセスしようとしているページのURLとREADの子要素のURL値を比較し、選択される候補ルールを決める

(2) ルールの選択

(1)の各候補ルール中でREAD要素以外の要素値を満足するルールをトリガ(trigger)する

(3) 条件の評価

トリガされたルールの条件と履歴情報を比較する。アクセス情報を履歴情報として蓄積領域に格納する。条件を満たすルールに(4)を行う

(4) アクションの実行順序を決定

(1), (2), (3)で同時に複数のルールが適合した場合、各ルールのアクションを実行する順序を決定する。

(5) アクションの実行

(4)で決められた順序にしたがって、ルールのアクションを実行する

本手法の概要を図4に示す。上記の例では、ルールのIDがrule1, rule2である動的再構成ルールのどちらを適用するかで、表示すべきアンカー文字列とリンク先が決定する。どちらのイベントでもない場合は二つとも適用されないで、動的再構成処理は行われない。アクションに記述された動的再考性処理は、スタイルファイルで実行される。アンカー文字列やバナー画像とリンク先URLとを一般的なリンクとして閲覧者に提供するために、以下のテンプレート記述を適用する。

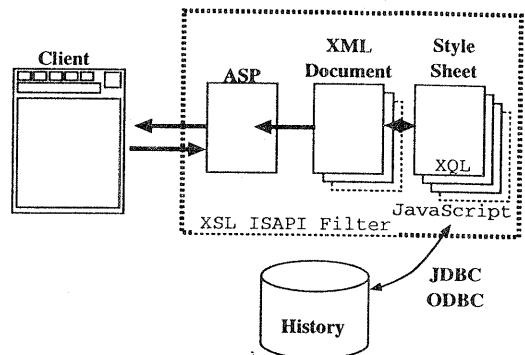


図4 システムの概要

```

<xsl:for-each
  select="RULES/RULE/ACTION/ADD">
  <xsl:if test="LINK/LABEL">
    <A><xsl:attribute="HREF">
      <xsl:value-of select="LINK/URL">
        </xsl:attribute>
      <xsl:value-of select="LINK/LABEL">
        </A>
    </xsl:if>
    <xsl:if test="LINK/IMAGE">
      <A><xsl:attribute="HREF">
        <xsl:value-of "select=LINK/URL">
          </xsl:attribute>
        </A>
      </xsl:if>
    </xsl:for-each>

```

テンプレートはXML表現をHTMLに変換するだけでなく、XQLによるXML内の問合せが可能である。また、閲覧者履歴を動的再構成ルールの条件とする場合は、XSLスタイルシート中のJavaScriptからJAVA Apletを呼び出すことでデータベースにアクセスすることが可能であるため、アクセス情報蓄積領域(History)からJDBCやODBCを介した問合せ結果をJavaScriptの該当変数への返り値として利用することができる。アクセスされるページから呼び出すXMLのスタイルシートはサーバサイドのフィルターで選択される。

一方、任意のデータをXMLファイルに記述し、プログラムやスクリプトで再構成を実現する手法も考えられる。スタイルシート中でリンク先を判定するスクリプトファイルを

```
<SCRIPT LANGUAGE="javascript" SRC="links.js">
```

のように指定して、links.js 中に

```
prev==history.previous;
if (prev=="http://.../adv1.asp")
  {url = "./special1.asp";}
else if (prev=="http://.../adv2.asp")
  {url = "./special2.asp";}
else url = "";
```

のようなプログラミングを行えば、DHTML や ASP だけでも実現は可能である。しかしながら、この方式では、ルールに対する記述やスタイルをページ毎に用意しなければならないので効率的とはいえない。そのため、動的再構成ルールを XML で記述し、XSL スタイルシートで XQL や JDBC, ODBC を用いて条件を評価した後に、表示体裁をテンプレートとして表現するアプローチを選択している。

5. アクティブデータベースシステムとの差異

データベースシステムは、更新トランザクションと読み出しトランザクションの両方を扱う。一般的に Web システムは読み出し専用のアプリケーションであるため、閲覧者の読み出し要求に対する操作のみを行う。ページ内容の動的再構成は、閲覧者がアクセスするページの見え方を変化させるため、データベースのビューインスタンスを生成する操作とよく似ている。

本節では、アクティブデータベースシステムのルールとアクティブ Web システムのルールを、記述内容やルール評価とアクション起動のタイミングについて比較する。さらに、クライアントが蓄積しているページのキャッシュとデータベースの実体化ビューの類似点と差異について議論する。

5.1 知識モデル

アクティブデータベースのアクティブルールはイベント・条件・アクションの組である。我々のアクティブ Web システムもアクティブルールに準じた分類で表 1 に整理した。

アクティブデータベースシステムではイベントに構造操作・トランザクション・例外等の組合せを記述できる。例えば、テーブルにタプルが挿入されるトランザクションが中断した事象を条件部分の評価を開始する引金 (trigger) として記述することができる。一方、我々のアクティブ Web システムは、閲覧者の位置や時刻などの閲覧者の状態とアクセスされるページの組を記述し、指定されたページにアクセスされた時に条件部分の評価してアクションを適用する。条件部

	Active DB	Active Web
Event	構造操作 トランザクション 例外	ページアクセス 閲覧者の位置 時刻
Conditon	データベースの 状態	閲覧者個別履歴 履歴情報の集約
Action	データベースの 操作	コンテンツの 再構成 ページ活性化

表 1 ルールの記述内容

分は、アクティブルールでは指定したトランザクションが開始されたかや、他のルールのアクションが実行中か等のデータベースの状態を記述する。アクション部分はともに操作を記述している。特に、*replace* はアクティブデータベースの *do instead* とよく似た機能である⁶⁾。

5.2 実行モデル

アクティブルールの条件は、イベントが発生したら即座に評価されるものと、ある期間だけ遅延して評価されるものがある。アクティブ Web システムでは、閲覧者の read 要求に対してページコンテンツを動的に変化させるためにルールを適用するため、イベントによるルール選択と条件は即座に評価されサーバがアクセス情報を記録する。データベースシステムでは、条件の評価対象がトランザクションの実行状況やその時点でのデータベースの状態であるのに対して、アクティブ Web システムでは閲覧者個別のアクセス履歴や全閲覧者のアクセス履歴の集約などが評価の対象となる点で異なる。アクションにも、即時起動型 (immediate) と遅延起動型 (deferred) があるが、アクティブデータベースではルールでデータベースの更新処理を記述できるため、一度しか起動しないアクションや一度条件を見たとせばその後はグループ内のすべてのトランザクションに対して適用できるアクションを記述できるようになっている。アクティブ Web も、即時起動型と、「来月から画像を置き換える」などの遅延起動型のアクションが必要である。これに加えて、「明日から三ヶ月、夏用のスタイルにする」などの条件はルールで指定することも考えられるため、ある一定期

	Active DB	Active Web
Conditon	Deferred Immediate Detached	Immediate
Action	Immediate Deferred Detached	Immediate Deferred

表 2 実行モデルの比較

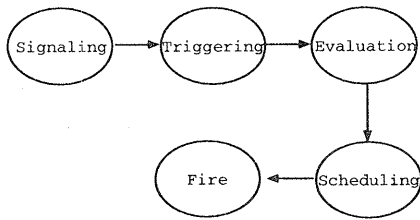


図5 ルールの選択とアクションの実行

間だけアクションを適用する機能も必要である。

アクティブデータベースシステムではアクションの起動タイミングをイベントとなったトランザクションのコミットに依存させることができる。アクティブ Web システムにはこの機能は必要ない。

アクティブ Web システムの動的再構成ルールの処理は以下のように行われる。(図5)

- イベント発生の検知 (Signaling)
- 検知されたイベントが記述されているルールを選択 (Triggering)
- 選択されたルールの条件を評価 (Evaluation)
- 条件を満たすルールのアクションの実行方法を決定 (Scheduling)
- アクションの実行 (Fire)

実行モデルではルール処理に違いがないように見えるが、アクティブ Web システムは、ページの元データに変更を加えることがないので、スケジューリング工程でルールの衝突だけを監視すればよい。例えば、同一のイベントにより選択された複数のルール中で条件部分をもたないルールと条件を満たしてアクションを実行しようとしているルール間の適用順序の決定がこれにあたる。

Web システムでは、応答時間の短縮と通信コストの節約を目的としてクライアントが一時的にページデータをキャッシュしている。そのため、短期間に同じ URL のページを表示する時にはデータが更新されず、ページを動的に再構成するアクティブ Web システムではデータベースシステムの実体化ビュー更新問題と同様なことが起こる。つまり、ページデータのキャッシュ期間によって、閲覧者が実際に得られる情報量は違ってくるといえる問題がある。

データベースシステムの実体化ビューに対する更新は、実データベースへの更新の反映が困難なため利用されることは少ない。しかし、移動体上にデータベースのスナップショットを記憶させて、通信状況に応じて更新処理を行うような場合はデータベースの一貫性を乱す恐れがある。

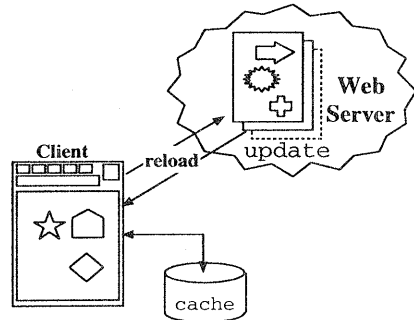


図6 キャッシュによるサーバデータとクライアントデータの差

Web システムでも、長期間に及ぶキャッシュ利用により、クライアントとサーバ間でデータの整合性がなくなる問題は発生するが、重要な更新が行われた時に更新データのプッシュや能動的更新も可能であることに加えて、サーバ側で再構成処理をすべて行うため、実体化ビューの更新問題とは本質的に異なる。

6. まとめ

Web システムにおけるアクセス権管理を具体的に実現するために、ページコンテンツの動的再構成について議論した。動的再構成ルールをアクティブデータベースシステムのECAルールにならって記述してルール処理を行い、閲覧者個別のページビューを表現する実装法を示している。また、アクティブデータベースの知識モデル記述と動的再構成ルールによる表現の違いを議論するとともに、実行モデルにおける差異にも言及している。

関連研究として、Ceri らと Atzeni らの研究がある^{7)~9)}。Ceri らは、電子商取引におけるビジネスルールを XML 表現して利用者プロフィールとともに格納し、個別販売のビジネスモデルを提案している。Atzeni らは、Web ページをデータベース技術を用いて再構成する Araneus データモデルを提案している。我々のアクティブ Web システムは、ページ作成者が閲覧者に伝達したい情報の意図をルールとして宣言的に定義できることを目的としている。さらに、閲覧者がコンテンツを利用するための制約という観点からでなく、ページ作者の意図どおりの閲覧やリンク巡行に対して、閲覧者にもメリットがあるハイパーテキストシステムの構築を目指している点でこれらの研究と異なる。

移動体通信や携帯電話のインターネット接続サービスを利用する場合、利用者の実際の位置とともに目的地等の仮想位置の利用が有効である。また、実位置と仮想位置によるアクセス方法を区別することで提供

するサービスを変えたり、クライアントの種類によっても転送する情報を選択するアプリケーションが考えられる。スケジュールを先取りするようなアプリケーションでは、自動的に位置情報を獲得する仕掛けとともに、時間による位置の変化や、経由地・目的地への到着時刻の予測機構が必要である。携帯電話のような通常移動することが想定される接続方法の場合は、移動速度や方向を実位置データとして取得できる仕組みも必要である。携帯電話ではアクセスポイントから位置情報を得ることができるが、GPS サービスを利用して携帯情報端末の位置情報を得たり、ナビゲーションシステムから目的地や経由地などの仮想位置を得ることも可能になると考えられる。さらに、位置と時間によるアクセス管理により、特定の場所や時刻により伝達する情報を変化させることができれば有用である。

本研究のゴールは、ページ作成者が容易にページ内容と設定したリンク構造の意図を表現できるシステムの実現である。そのため、特別な知識を必要としない実現手法として宣言的なルール記述による、ページ表現の個別化を提案した。そこではXML, XSL, XQLならびにデータベース接続を視覚的方法で設定するシステムを必要とする。また、キャッシュ利用によるクライアントとサーバ間のデータ不整合の解消は、本質的にデータベースの実体化ビュー更新問題と異なるため、注意を要する。設定されたルールの解釈とルールの衝突回避ならびに、プッシュ技術を用いた整合性保持の実装は今後の課題である。

<謝辞>

本研究をすすめるにあたり、多くの御意見と助言を頂いた本学工学部ならびに自然科学研究科田中研究室の皆様へ深く感謝致します。また、本研究の一部は、日本学術振興会未来開拓学術推進事業における研究プロジェクト「マルチメディア・コンテンツの高次処理の研究」(プロジェクト番号 JSPS-RFTF97P00501)ならびに、文部省科研費「分散型ハイパーメディアからの構造発見とアクセス管理」(課題番号 12680416)による。ここに記して謝意を表します。

参 考 文 献

- 1) 清光英成, 田中克己, "Web ページの巡行に基づく動的なリンク活性化とアクセス管理", 情報処理学会アドバンスデータベースシステムシンポジウム論文集, Vol. 99, No. 19, pp. 115-122, 1999.
- 2) Hidenari Kiyomitsu and Katsumi Tanaka, "Navigation-Dependent Web-Views: Defining

and Controlling Semantic Units of Web Pages", *Internet Applications*, IEEE LNCS-1749, pp. 421-426, Springer, 1999

- 3) 清光英成, 竹内淳記, 田中克己, 閲覧履歴に依存したWebコンテンツの動的生成方式, 第11回データ工学ワークショップ論文集, 2000年3月2日.
- 4) 清光英成, 竹内淳記, 田中克己, "アクセス履歴と利用者の位置に基づくコンテンツの再構成," 情報処理学会研究報告, Vol.2000, No.44, pp. 17-24, 2000.
- 5) Norman W. Patom, "Active Database System", *ACM Computing Surveys*, Vol. 31, No. 1, pp. 63-103, 1999.
- 6) Klaus R. Dittrich, Stella Gatzju, Andreas Geppert, "The Active Database Management System Manifesto: A Rulebase of ADBMS Feature", *Rules in Database System*, IEEE LNCS-985, pp. 1-17, Springer, 1995.
- 7) S. Ceri, P. Fraternali, A. Maurino, S. Paraboschi, "One-to-One Personalization of Data-Intensive Web Sites" *WebDB'99*, pp. 1-6, 1999.
- 8) S. Ceri, P. Fraternali, S. Paraboschi, "Data-Driven, One-To-One Web Site Generation for Data-Intensive Applications" *Proc. of VLDB'99*, pp. 615-626, 1999.
- 9) P. Atzeni, G. Mocca, and P. Merialdo, "To Weave the Web", *Proc. of VLDB'97*, pp. 206-215, 1997.