

カプセル化コンテンツ流通基盤(1) - 全体構成と利用状況適応機能 -

1 K - 7

細見 格 中江 政行 市山 俊治

NECヒューマンメディア研究所

1. はじめに

商品としてのデジタルコンテンツの流通を考えた場合、その主な問題としてデジタルコンテンツの複製の容易さに基づく商品の不正使用が挙げられる。この問題への対処法には大きく二種類あり、一方はコンテンツの無断複製を防ぐコピープロテクトやアクセス制限、もう一方は複製自体は自由だが特定の手続き無しにはコンテンツを利用できなくするカプセル化(暗号化)である。本稿では、カプセル化を採用した上で柔軟な利用と課金制御を可能にしたコンテンツ流通基盤、およびその特徴として利用者の要求や利用環境に応じた内容を提供する技術について述べる。

2. カプセル化コンテンツ流通基盤

本研究では、コンテンツのカプセル化アーキテクチャとして MediaShell [1] を採用したコンテンツ流通基盤を構築している。本研究のコンテンツ流通モデルは図1のような構成となっている。

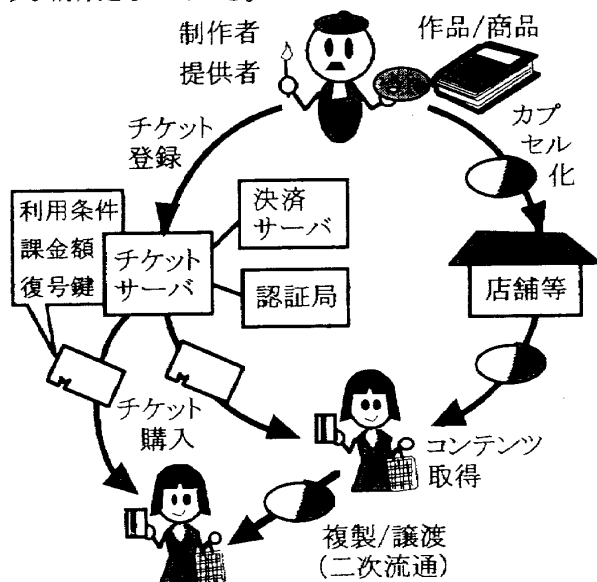


図1 カプセル化コンテンツ流通モデル

図1でチケットとは、ある特定のコンテンツの価格と利用条件、およびそのコンテンツを復号するための鍵を含む暗号データである。デジタルコンテンツは複製が容易な上、同じコンテンツでも利用時の端末装置や利用者の要求は様々である。そこで本研究では、利用者が適

*A Capsulated Content Distribution Architecture (1)
- Total structure & adaptive content services -
Itaru Hosomi, Masayuki Nakae, Shunji Ichiyama
Human Media Labs., NEC Corp.*

当なチケットを選択して購入することにより、同じコンテンツでも用途に適した品質や構成で内容を得られ、それらに応じた支払いを行なうことができるようにしている。また、本研究では得られたコンテンツを各状況下で快適に利用できるようにするために、利用状況適応機能を前述の MediaShell の機能として実装した。なお、利用と課金の制御については別稿 [2] で詳述している。

3. デジタル情報コンテンツ MediaShell

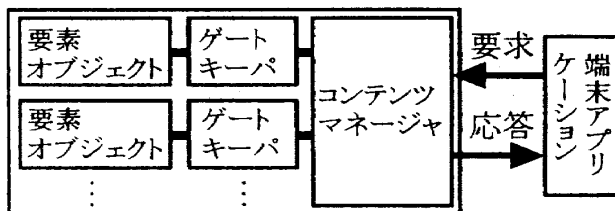
3.1 カプセル化コンテンツとコンテンツ

コンテンツをカプセル化したものをカプセル化コンテンツまたは単にカプセルと呼び、1つのカプセルに納められる要素の種類や構成を定義したものをコンテンツと呼ぶ。カプセルはそれ自体の安全性が高いため自由な経路で流通させることができるが、反面、コンテンツを利用できる環境や表現形態への制限が多い。本研究では、コンテンツの安全性と利用の柔軟性を両立させたカプセルを実現するデジタル情報コンテンツ MediaShell を提案し、その開発を進めている [1]。

3.2 MediaShell アーキテクチャ

本流通システムのベースとなる MediaShell の基本構造については先の論文[1]で述べた。MediaShell 形式のカプセルは、コンテンツマネージャ、ゲートキーパおよび各ゲートキーパが管理する要素オブジェクト群からなる(図2)。既存技術との主な違いは以下のような点である。

1. 要素オブジェクトの構成を管理するだけでなく、要求に応じて出力時の構成変更が可能。
2. 各要素オブジェクトの役割やコンテンツ制作者の意向を考慮した上記構成変更処理の制御が可能。



MediaShell カプセル

図2 MediaShell の基本構造

4. コンテンツの利用状況への適応

2章で述べたようにデジタルコンテンツの利用状況は多種多様であり、各状況において可能な限り効果的にコンテンツを提供する技術は、利用者/提供者双方にとつ

て重要である。Media-Shellは要素オブジェクトを個別に管理し、その構成を木構造で保持・制御することにより図3に示すような各種の利用状況適応機能を実現している。以降では論文[1]で述べていないレイアウト変更機能と部分要素抽出機能について説明する。

要素縮小表示	代替要素提供
要素表示オンデマンド化	レイアウト変更
出力不可要素削除	部分要素抽出

図3 MediaShell の利用状況適応機能一覧

4.1 オブジェクト構造の木構造化

MediaShell は、要素オブジェクトの表示画面上の構成を木構造で表現し、汎用的なルールを用いて構成を変更することにより、以下のような要求にも対応できる。

- ・ 未知の解像度／縦横比の端末画面への自動適応
 - ・ 要素オブジェクト間の隣接関係や包含関係の維持
- 木構造は画面上の領域分割をノードの分岐に対応させ、以下のように表現している。

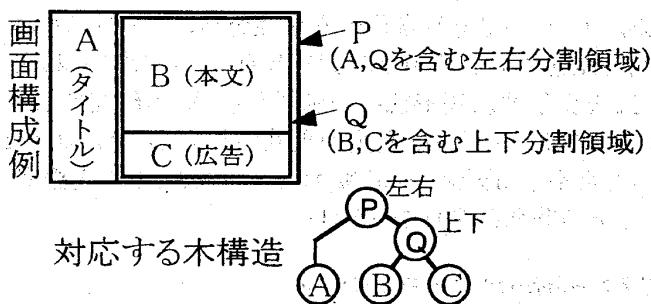


図4 コンテンツの出力画面構成例と対応する木構造

4.2 レイアウト変更機能

端末の制約や利用者の要求は以下のようなスクリプトで表現し、カプセルにメッセージとして送る。

```
例: play adaptive
    condition client.disp.width = "640"
    and client.disp.height = "480";
```

これに対して、カプセル内には適応表示用のルールを例えば次のように記述する。

```
1: cond="client.disp.height < profile.disp.height"
   // 条件: 端末画面の高さが必要解像度未満なら、
2: ope="horizontal gk03.ad" // 広告を90度配置変更
3: ope="reduce gk01.title" // タイトルを簡略表示
```

カプセル外部から受け取ったメッセージ (1) とカプセル内部のルール (2) から、コンテンツマネージャが要素オブジェクトの構成を表す木構造を変更する。

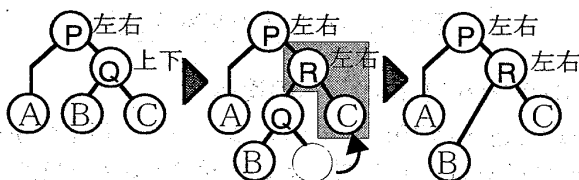


図5 ルール (2) と図4の例に基づく変更処理

ルール(2)で cond="..." を条件、ope="..." をメソッドと呼ぶ。図5はルール(2)の 2: で示した horizontal メソッドを実行した場合の処理である。このメソッドは、変更対象の表示要素と隣接する要素との位置関係を上下から左右に変更する。結果として、図6右側のように表示領域の高さの不足に対処したレイアウトとなる。ここで領域AやCの高さは許容範囲内で自動調整され、許容最小サイズを超える場合はルール中の次のメソッドを適用する。条件を満たすかメソッドが尽きれば、そのルールの参照を終える。

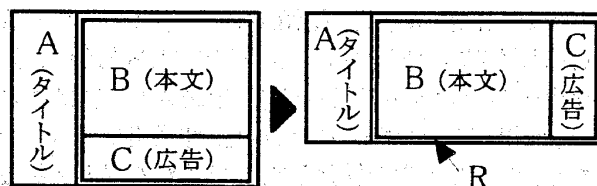


図6 レイアウト変更結果

このような木構造への部分木挿入やノード付替えにより、配置の変更を行なっても要素オブジェクト間の隣接関係や包含関係(木構造のノードの親子関係)が自動的に維持されるため、例えばメニューとその選択結果の表示領域との対応が崩れるのを防ぐことができる。

4.3 部分要素抽出機能

MediaShellでは下記の例のようなメッセージによりコンテンツ内の希望する部分のみを抽出することができる。

```
例: select *title // タイトルを検索
    condition client.disp.imageout = "no" // 画像不可
    where date >="19980716" // 制作日条件指定
    and format = "html"; // ファイル形式は HTML
```

上記の例では検索条件(where)の他に前記 (1) と同様の出力条件(condition)がある。検索条件を満たしても出力条件を満たさない要素は前節と同様に可能ならば適応化処理を施し、不可能ならばその由を示す。

5. おわりに

以上のような機能により、MediaShell ではコンテンツをカプセル化した上で同じコンテンツでも表示される内容を端末や要求内容に応じて再構成できるため、コンテンツをより快適に利用できる。また、再構成した結果として表示される各要素オブジェクトを個別に MediaShell のカプセル内で管理しているため、同じコンテンツに対して表示内容に応じた課金も可能である[2]。

参考文献

[1] 細見, 他: 多様な再生環境に適応する流通コンテンツ・アーキテクチャの提案, 情報処理学会第55回全国大会論文集, 6Q-05, 1997.
 [2] 中江, 他: カプセル化コンテンツ流通基盤(2) -チケットによる利用制御方式-, 情報処理学会第57回全国大会論文集, 1K-08, 1998.