

文書ドラフト形成支援システム Nelumbo (1)

4W-4

- ドラフト形成プロセス -

林浩一 関島章文

富士ゼロックス(株) システム・コミュニケーション研究所

1. はじめに

近年、文書の作成過程を一貫して支援する文書処理システムに関する研究が、数多く報告されている。中でも WE[1]は重要な先駆的研究である。WEは、文書の作成過程を表現するオーサリング・モデルを定義し、オーサリング・モデルに基づいて文書構造・文書エディタを設計することにより、実用的な文書作成支援環境を実現しようとした。WEのオーサリング・モデルでは、文書の作成過程を(1)アイデア蓄積、(2)論旨構成、(3)レイアウト生成の3つのフェイズに分類し、これらのフェイズが順に実行されることにより最終的な文書が作成されると定義している。このオーサリング・モデルは現在もなお有力であり、多くの文書作成支援システムの基礎となっている。

しかし、現実の文書作成作業は、必ずしも WEのオーサリング・モデルに従うとは限らない。我々は、より現実的な観点からオーサリング・モデルとそれに基づく文書モデル(文書構造の定義およびエディタとの相互干渉モデル)を再定義し、更に実験システム Nelumbo を試作した。本稿では、我々のオーサリング・モデルと文書モデルについて述べる。

2. オーサリング・モデル

WEのオーサリング・モデルに基づいたシステムでは、文書作成の各フェイズに固有のデータ構造によって文書が表現され、また、フェイズの目的に応じたエディタによって編集作業が行われる。

(1)アイデア蓄積

文書作成の最初の段階では、文書を構成するアイデア断片の蓄積が行われる。アイデア断片をノードとするネットワーク構造(ハイパーテキスト)が、カード型エディタなどによって構築される。

(2)論旨構成

次の段階では、蓄積されたアイデアから文書の論旨が抽出・構成される。文書の論旨は、文書の内容断片をノードとする木構造として表現され、アウトラインエディタなどによって編集される。

(3)レイアウト生成

文書作成の最終段階では、文書を印刷するためのレイアウト構造が生成される。レイアウト構造は、文書の内容断片を一定の順序で割り付けて得られる線形構造であり、論旨の木構造から割り付け処理によって自動的に生成され、WYSIWYG エディタなどで編集される。

WEのオーサリング・モデルでは、これらのフェイズを段階的に経て最終的な文書が作成されるとしている。しかし、これは必ずしも現実の文書作成過程を反映していない。現実の文書作成過程においては、しばしば文書を一旦レイアウトした後に、アイデアの追加・論旨の再構築が必要となる。

このような観察を踏まえて、新しいオーサリング・モデルを提案する。我々は、文書の作成過程を、上記の各フェイズが循環的に繰り返されることによって漸近的に文書の完成度が高められていく過程(ドラフト形成プロセス)と定義する。

このオーサリング・モデルを実現するためには、文書作成の各フェイズにおける編集結果を、別のフェイズにおいても取り出すことを可能にする文書モデルが必要である。WEの文書モデルでは、各フェイズの編集結果がフェイズに固有な文書構造内に留まってしまい、基準を満たさない。

我々は、このオーサリング・モデルを実現するために、新しい文書モデル「フィールド・アンカーモデル」を提案する。フィールド・アンカーモデルは、文書作成のフェイズ間で共有される文書構造を提供する。即ち、異なるフェイズで用いられる各種エディタ(カード型エディタ、アウトラインエディタ、WYSIWYG エディタ)が、それぞれの方法で編集対象を定めることが出来る共通の文書構造を与える。従って、各フェイズにおける編集結果を常に共通の

Nelumbo: a Drafting Support System (1)

- Authoring Model for Drafting Processes -

Koichi Hayashi and Akifumi Sekijima

Systems & Communications Lab., Fuji Xerox Co., Ltd.

文書構造に反映させることが可能となり、フェイズの枠組を越えた編集作業の共有が実現する。

3. フィールド・アンカーモデル

(1)基本構造

フィールド・アンカーモデルは、以下の要素から構成されるネットワーク構造である(図1)。

プリミティブ: 操作、表示の対象となる最小の単位である。例えば文字はプリミティブである。

フィールド: プリミティブの集合を保持するストレージである。フィールドは、表現している内容体系に応じて3つのタイプに分類され、各タイプに応じた固有の空間が定義される(表1)。また、フィールドに含まれるプリミティブは、フィールドに固有な空間内の位置を保持する。

表1 フィールドのタイプ

タイプ	固有空間	表現する内容体系
ストリーム	順序集合	テキスト
マップ	二次元平面	グラフィック
グリッド	格子平面	表

アンカー: プリミティブの1つで、フィールドを参照する指標である。アンカーは常に唯一のフィールドを参照するが、フィールドは複数のアンカーからの参照を許す。アンカーには、*On* または *Below* の「状態」が付与される。アンカーの状態が *On* であることは、そのアンカーが論旨構造中の特定の位置に現れることを意味する。

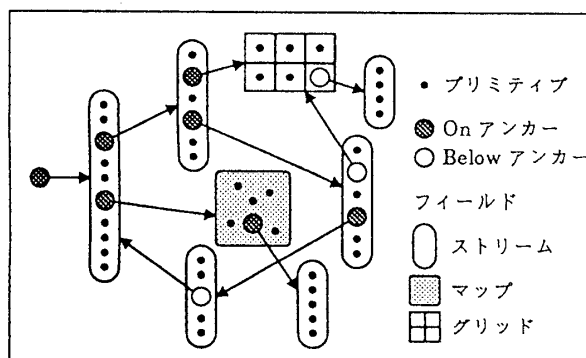


図1 フィールド・アンカーモデル

(2)論旨構造

On アンカーとそのアンカーが参照するフィールド内の *On* アンカーの間に親子関係を定めることによって、*On* アンカーから成る木構造(論旨構造)を抽出することが出来る。

一般に、順序付木構造では兄弟ノードの間に単純

な順序が定義されているが、ここで抽出される論旨構造では、フィールドの固有空間内での位置が、兄弟アンカーの関係を定める。従って、論旨構造は自然に *tnt* 文書構造[2]とみなすことができ、文書構造と多様な内容体系の混在を許す。

(3)レイアウト構造

論旨構造から、割り付け処理によってレイアウト構造が生成される。このとき、フィールドの内容は、それぞれの内容体系に固有のフォーマットによって割り付けられる。論旨構造中には異なる内容体系を持つフィールドが任意に混在するので、異種フォーマット間のインタフェイス技術[3]が必要となる。

(4)エディタと文書構造の相互干渉

基本構造は文書作成の第1フェイズ、論旨構造は第2フェイズ、レイアウト構造は第3フェイズでそれぞれ対応するエディタの編集対象となる。個々のエディタ設計には、①フィールドを参照しているアンカーが取るべき状態、②フィールド中で編集対象となるアンカーの状態、③ユーザによる編集作業が引き起こすアンカーの状態変化が定義される。編集作業に伴うアンカーの状態変化は、ユーザが意図して行う場合と、フィールドを参照・編集する際に暗黙的に行われる場合がある。これはユーザによる参照・編集の履歴をアンカーの状態に反映させることにより、論旨構造の形成に利用するためである。

4. まとめ

アイデア蓄積、論旨構成、レイアウト生成の各フェイズを循環的に実行し、漸近的に文書を完成に導くオーサリング・モデルとそれに基づく新しい文書モデルを定義した。我々は、このオーサリング・モデルに基づく実験システム *Nelumbo* を試作した。*Nelumbo* の概要については別稿で述べる。

参考文献

- [1] J. B. Smith, S. F. Weiss and G. J. Ferguson, 'A Hypertext Writing Environment and its Cognitive Basis', *Proceedings of Hypertext '87*, 195-214, 1987.
- [2] R. Furuta, 'Concepts and models for structure documents', *Structured Documents*, Cambridge University Press, 7-38, 1989
- [3] M. Murata and K. Hayashi, 'Formatter Hierarchy for Structured Documents', *EP92*, Cambridge University Press, 77-94, 1992.