ロバスト構造化文書処理技術

武市正人　胡振江　松崎公紀

情報理工学系研究科数理情報学専攻

概 要

ロバスト構造化文書処理技術プロジェクトは，Programmable Structured Document（PSD）の手法を提案し，堅牢な文書処理技術の方法論を構築することを目標としている．PSD は，XML に代表される構造化文書のなかにプログラムの記述を許し，このプログラム記述に対して，これまでに得られている関数プログラミングおよびプログラム変換手法を適用することにより，ロバストかつ効率的な構造化文書処理を実現しようとするものである．

1　はじめに

データや文書を表現する方法として，XML（eXtensible Markup Language）と呼ばれる，構造を表すためのタグが付けられたテキストが広く普及するようになった．例えば学校の成績表であれば，学年の下にクラスが，クラスの下に学生が列挙され，その学生の下に科目毎の成績が示される，というような構造を持つことになる．このような電子的な構造化文書情報の蓄積と効果的な情報利用技術は，インターネットを含む広範な情報の交換・流通にとってきわめて重要な位置を占めている．XML に代表されるこれらの技術は，発展の著しい WEB による情報環境に向けて既存の技術の延長線上で実務的に開発されたもので，事実上の標準となってはいるがその言語的な概念が十分には整理されていない．このような体系的な処理技術の欠如が今後の情報交換の発展を阻害し，既存技術による個別的対処や人手による個別対応では一般性を欠く文書情報を蓄積することとなっており，この問題を解決することが重要な課題となっている．

構造化文書はプログラミング言語のデータ構造と類似しており，文書処理のアルゴリズムを記述するためには関数型言語が適している．本研究の中核となる Programmable Structured Document（PSD）プログラムの記述を含む文書を対象とし，これまで関数型言語での処理およびプログラム運算（変換）手法を利用して，ロバストでかつ効率的な構造化文書処理を実現しようとするものである．すなわち，構造化文書をプログラミングにおける構造化データであるとみなし，プログラミング言語に関する理論を適用することによって，安全かつ信頼性の高い処理を実現する．また，処理を行うコードを対象文書に埋め込むことで，文書の高い可搬性を実現する．

PSD を構成する要素として，PSD アプリケーションである文書そのもの，PSD を実行するためのプラットフォーム，およびその理論的裏付けを与え，その正当性や安全性を保証する理論の 3 つがあげられる．PSD フレームワーク（図 1 参照）では文書がその文書自身を扱うことができるようなコードを含んでいるが，そのコードを記述するプログラム言語を限定しないことが望まれるため，PSD を利用するためのプラットフォームは，XML 文書を処理する XML エディタと，PSD に埋め込まれたコードを実行する外部評価系を分離した構成となる．また，PSD の普及を促進するためには，PSD 評価システムや PSD 作成支援に加えて，既存の構造化文書を効率的に PSD に変換する機構の提供が不可欠である．PSD を実現するための重要な課題は，

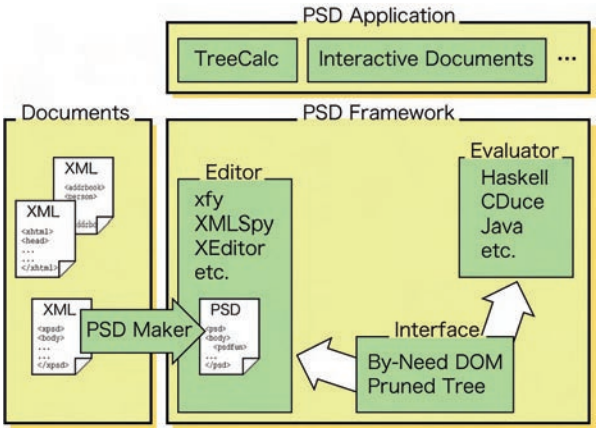

図 1: PSD プロジェクト概要

- 構造化文書に必要な準構造データの概念を型として捉える形式的枠組の定義および，データ型に基づく効率的変換手法，
- PSD のための計算機構を組み込んだ構造化文書の実現手法，
- 関数型言語におけるデータ型の理論を発展させた，準構造データに適した型の理論（これは，代数的プログラム変換，すなわち運算の成果を構造化文書に適用し，自己参照による文字変換や変換戦略を文書自体に付随させるという運算随伴機構に関する理論を含む）

を与えることである．

本プロジェクトでは，構造化文書に対する PSD 構造化手法と変換（運算）規則を体系化し，ソフトウェアの信頼性確保の基礎となる参照用文書（言語仕様・設計仕様）を作成する．

2　平成 17 年度の研究成果

2.1　双方向変換機構と双方向変換言語

双方向変換は，2 つの構造化データの間での同期を取ることを目的に考案された技術である．これらの双方向変換は，始点と終点の 2 つのデータ間について，順方向の変換を記述することが同時に逆方向への情報の更新方法も実現するように設

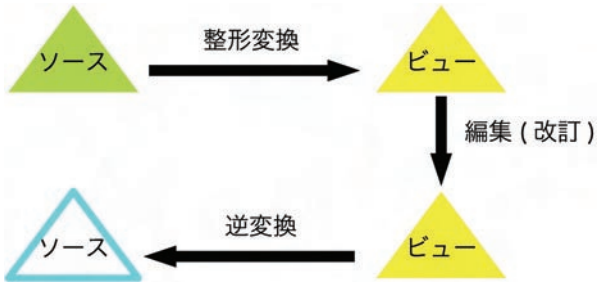

図 2: 双方向編集機構

計されている．また，一般の逆変換と違い，変換元と変換先の 2 つの情報を使って逆方向の反映を行うため，データの削除や挿入など幅広い変換を記述することができるのが特徴である．

文書の編集作業，特に XML 文書の編集作業は，直接 XML 文書に手を加えるのではなく，もともとの XML 文書から生成される View と呼ばれるエディタ上の表示情報に対して編集が行なわれる．このとき，変更された結果をもとの文書に反映させなければならない (図 2)．

このような表示に基づく編集機能を設計する際に困難なことは，どのような変更もその結果がもとの文書に反映させることを保証することができ，かつ妥当な変換規則の制約を定めることである．このような仕様記述に適した双方向に変換可能な言語に関する研究を行い，プログラミング言語 X を設計した．言語 X は文書から表示への変換を記述するための言語である．言語 X の双方向性が証明され，言語 X で書かれた全ての変換に対して，その逆変換である表示から文書への変換の自動導出が可能であることが確認された．さらに，X に基づいて，双方向変換を記述するための Java ライブラリ BiXJ を設計し実現した．

2.2　双方向変換に基づく構造化文書の管理 : Bi-Link ファイルマネージャ

ファイルの操作において，ショートカット，シンボリックリンクといったファイル別名の存在は，しばしばユーザを混乱させる．初心者にとってファイル実体とそれに対する別名の区別は難しいため，メールへの添付やメディアへのバックアップの際

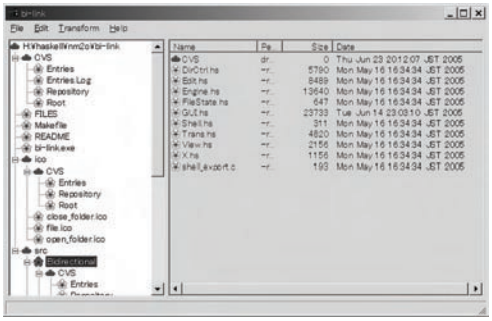

図 3: 「梅林」のスクリーンショット

にファイル実体でなく別名を用いてしまうことがある．また，別名と実体の関係は非対称であり，ファイル別名を消してもファイル実体は削除されず，逆にファイル実体を消すと無効なファイル別名が残る．アプリケーションのアンインストールを行う際に，ファイル別名のみを削除し，ファイル実体を削除しないという失敗は少なくない．実際に Microsoft Windows のファイルマネージャであるエクスプローラでは，このような失敗を防ぐため，デスクトップなどに存在するファイル別名を消去する際に，ファイル実体が削除されない旨の確認ダイアログが出る．また，一般にユーザにとって，あるファイル実体を参照している全てのファイル別名を列挙するのは容易ではないため，ファイル実体を消去した後に，存在しない実体を指し示している無効なファイル別名が残ったままとなりやすい．

そこで本研究では，ファイルへの参照すべてを対称かつ統一的に抽象化して扱うことのできるファイルマネージャ「梅林 (Bi-Link)」を，木上の双方向変換の技術を用いて実現した．図 3 が本研究において開発された双方向変換ファイルマネージャ「梅林 (bi-link)」のスクリーンショットである．複数の場所から 1 つのファイルを参照したい場合，既存のファイルマネージャはファイル実体と別名，つまりファイル名であるファイル参照とファイル参照への参照を用いる．これに対し，ファイルマネージャ「梅林」では，あるファイル参照を双方向変換により複数のファイル参照に変換する方法を用いる．この複数のファイル参照は互いに同期され，1 つに加えられた変更は別の同期されたファイル参照に反映される．たとえば，あるファイル参照の名前を変更すると同期されたファイル参照の名前も変更され，あるファイル参照の指す実体を削除すると同期されたファイル参照は全て削除される．ファイル実体を削除せずに同期されたファイル参照の 1 つを削除したい場合には，双方向変換によって同期されたファイル参照を “見えなく” することにより，ファイル別名の削除に対応する操作ができる．このように同期されたファイル参照は互いに対称に扱われ，無効なファイル別名を生じない．

また，既存のファイルマネージャは実際のディレクトリ木の見せ方に関する自由度が低い．ファイルに対する注釈，表示するファイルの順序，特定のファイルの隠蔽などいくつかの基本的な機能は提供されているが，ユーザはそれらを細かにカスタマイズすることはできない.「梅林」では，このような “見せ方” を双方向変換として記述する．そのため，これらの機能を統一的に表現でき，なおかつ見せ方の自由度を高めている．

2.3 対話的学習教材の作成支援環境の構築

iDocument は対話的に利用者が与えた入力に基づいて内容が動的に変化する文書であり，その応用先として教科書等の教育用アプリケーションがある．iDocument と併せて，これらを容易に作成できるツール，iDocument Builder も開発しており，従来のチュートリアル，教科書にはない新しい有用性を備えた対話的文書の開発に用いることが可能である．iDocument の応用事例として，(1) iTutorial: 利用者が与えた入力に基づいて説明文が動的に変わるチュートリアル; (2) iTextBook: 演習問題を本文の説明の中に埋め込み，学習者が対話的に理解できる教科書; (3) iExam: 採点機能や，解答に応じたコメントの表示の機能を備えた電子的試験用紙等を開発した．

iDocument は ジャストシステムが開発した複数の XML 文書をシームレスに閲覧・編集できる「xfy」上で作成された XML 文書に対して，ユーザの入力を受け取りそれに応じた内容を動的に作成

するコードを埋め込むことで対話的文書の作成を実現している．コードの評価にはこれまでに我々が開発した PSD 処理システムを用いた．

3　研究活動

本年度の研究活動としては，上記の研究成果を国際会議等で発表したほか，2005 年 12 月 7 日〜9 日には，東京大学山上会館で The Fourth Workshop on Programmable Structured Documents（第 4 回 PSD ワークショップ）を開催した．国外から 4 名の関連分野の研究者を招聘し，30 名近い参加者を得て，研究発表・討論による密度の高い研究交流を行った．なお，このワークショップは，本 21 世紀 COE プログラムと，文部科学省リーディングプロジェクト e-Society「高信頼性構造化文書変換技術」，および科学研究費補助金基盤研究 (A)(2)「運算随伴方式による文書情報処理言語の設計とその効果的利用に関する研究」により実施したものである．

4　今後の課題

現在の双方向変換機構に基づく構造化文書の作成システムは，一般の構造化文書のみに対応しているため，コードが文書内に埋め込まれているような PSD を想定していない．コードを扱うためには，実装と理論の両方の立場で，現在の枠組を拡張する必要がある．また，これまで開発した言語 X に限らず，一般の構造化文書変換言語 XSLT などに対して，X への埋め込みが可能であるかを追究することも重要である

本年度発表した関連論文

- Shin-Cheng Mu, Zhenjiang Hu and Masato Takeichi, Bidirectionalizing Tree Transformation Languages: A Case Study, **コンピュータソフトウエア**, Vol. 23, No. 2, 2006.
- 松田一孝，大川徳之，野村芳明，森田直幸，筧一彦，胡振江，武市正人, 木上の双方向変換を利用したファイルマネージャの実現, **情報処理学会論文誌**, Vol. 47, プログラミング (PRO28), 2006.
- Dongxi Liu, Zhenjiang Hu, Masato Takeichi, An Environment for Maintaining Computation Dependency in XML Documents, *ACM Symposium on Document Engineering (DocEng 2005)*, Bristol, United Kingdom, 2–4 November 2005, pp.42–51.
- Yasushi Hayashi, Zhenjiang Hu, Masato Takeichi, iDocument Builder: An Environment for Building XML-Based Interactive Teaching Materials, *The 3nd International Conference on Education and Information Systems, Technologies and Applications (EISTA 2005)*, Orlando, Florida, 14–17 July, 2005.
- 劉東喜，筧一彦，胡振江，武市正人，王浩，A Java Library for Bidirectional XML Transformation , **日本ソフトウェア科学会第 *22* 回大会**, 東北大学 青葉山キャンパス，2005 年 9 月 13 日 (火)〜15 日 (木).
- 穆信成， 胡振江， 武市正人，Bidirectional scripting for Structured Documents, **日本ソフトウェア科学会第 *22* 回大会**, 東北大学 青葉山キャンパス， 2005 年 9 月 13 日 (火)〜15 日 (木).
- 林 康史, 胡 振江, 武市 正人, 対話的学習教材の作成支援環境について, **情報処理学会・コンピュータと教育研究会 情報教育シンポジウム *2005***, ヤマハリゾートキロロ (北海道後志支庁 余市郡 赤井川村)， 2005 年 8 月 21 日〜23 日.