

木構造データの双方向変換システムにおける 安全なビュー編集操作に関する研究

数理情報第7研究室 森田 直幸
指導教員: 胡 振江 助教授

2007年2月6日

1 背景と目的

双方向変換を Web アプリケーションで用いるにあたって、安全性の問題は避けては通れない。元のデータベースの構造や一貫性を破壊してしまうようなデータは拒否する必要がある。

Foster らや Hu らが提案した双方向変換言語は状態指向であり [1][2], 逆方向変換による更新の際にソース全体を再生成する為、その度にソースの構造を検査するのは効率が悪い。

本研究の貢献は以下の3点である。

編集操作の安全性を保証 タグを用いてソースである木構造データ上に編集が可能かどうかの表現を行い、その情報を双方向変換によってビューへ伝播することで、ビュー上での安全な編集を検証できる枠組みを提案する。これにより、双方向変換システムは、ユーザが送信してきた編集済ビューを検査することで編集が安全であるかを確認できる。

ユーザの利便性を改善 本手法では、ビュー上の各部分が編集可能かどうかという情報を付加するので、ユーザはその情報を頼りに適切に編集操作を行うことが可能になる。

システムの構築 図1に示すような、双方向変換における編集操作の安全性を検証するシステムを実装し、動作を確認した。

2 双方向変換

全ての変換には、ソースである木の列 s からビューである木の列 t を生成する順方向変換と、ビュー t とソース s から新しいソース s' を生成する逆方向変換が定義される。ここでは、 x の順方向変換によってソ

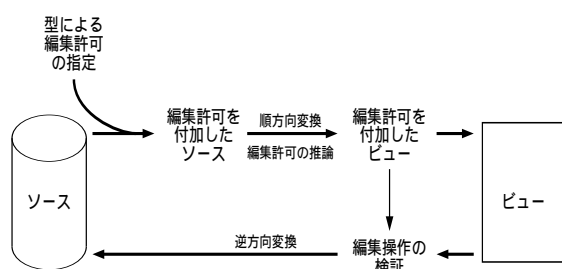


図1: 編集の安全性を保証する双方向変換システム

ス s からビュー t が生成されることを

$$x \vdash_F s \Rightarrow t$$

と書き、逆方向変換によってビュー t とソース s から新しいソース s' が生成されることを

$$x; s \vdash_B s' \Leftarrow t$$

と書く。

3 編集許可

以下のような編集許可タグを定義する。

- $u\{\text{mode} = \text{"mod"}\}[d]$: 文字列 d は修正することが許されている。
- $u\{\text{mode} = \text{"ins"} \text{ type} = \text{"T"}\}[d]$: データ列 d は T 型の要素からなり、要素の挿入が許されている。
- $u\{\text{mode} = \text{"del"} \text{ type} = \text{"T"}\}[d]$: データ列 d は T 型の要素からなり、要素の削除が許されている。

- 挿入・削除が共に可能な場合は、
 $u\{\text{mode} = \text{"ins|del"} \text{ type} = \text{"T"}\}[d]$ と表現する。

例 (タグ付けの例).

「1個以上である限りにおいて挿入・削除が可能」であるような email 要素の編集許可を表現する例を以下に示す。

要素が2個以上である場合は挿入・削除が可能である。

```
<email>abc@example.com</email>
<email>efg@example.com</email>
```

タグ付けの結果は以下の通りである。

```
<u mode="ins|del" type="email[String]">
  <email>abc@example.com</email>
  <email>efg@example.com</email>
</u>
```

要素が1個の場合は挿入のみ可能であるから、

```
<email>abc@example.com</email>
```

以下のようにタグを付ける。

```
<u mode="ins" type="email[String]">
  <email>abc@example.com</email>
</u>
```

4 ビュー上の編集許可

順方向変換の結果、ビューの型が $()$ になる場合は編集許可タグを取り除く。

$$\frac{x :: S \leftrightarrow () \quad m \in \{\text{"ins"}, \text{"del"}, \text{"ins|del"}\}}{x \vdash_F u\{\text{mode} = m \text{ type} = \text{"S"}\}[s] \Rightarrow ()} \quad (\text{F-U-EMPTY})$$

それ以外の場合については変換後の型を持つタグを付加したビューを生成する。ただし、データ列に対する編集操作の制約から、変換後に要素数が1より大きくなるような変換の場合はタグを取り除いてデータのみをビューに出力する。

$$\frac{x :: S \leftrightarrow T \neq () \quad \forall t \in [T], \#t \leq 1 \quad x \vdash_F s \Rightarrow v \quad m \in \{\text{"ins"}, \text{"del"}, \text{"ins|del"}\}}{x \vdash_F u\{\text{mode} = m \text{ type} = \text{"S"}\}[s] \Rightarrow u\{\text{mode} = m \text{ type} = \text{"T"}\}[v]} \quad (\text{F-U-INSDEL-1})$$

$$\frac{x :: S \leftrightarrow T \neq () \quad \exists t \in [T], \#t > 1 \quad x \vdash_F s \Rightarrow v \quad m \in \{\text{"ins"}, \text{"del"}, \text{"ins|del"}\}}{x \vdash_F u\{\text{mode} = m \text{ type} = \text{"S"}\}[s] \Rightarrow v} \quad (\text{F-U-INSDEL-2})$$

文字列の修正を許可するタグについては、以下の2つの規則を追加する。

$$\frac{s \text{ が文字列}}{\text{isText} \vdash_F u\{\text{mode} = \text{"mod"}\}[s] \Rightarrow u\{\text{mode} = \text{"mod"}\}[s]} \quad (\text{F-ISTEXT-MOD})$$

$$\frac{s \text{ が文字列}}{\text{isElem} \vdash_F u\{\text{mode} = \text{"mod"}\}[s] \Rightarrow ()} \quad (\text{F-ISELEM-MOD})$$

以上の5つのルールによって、ビュー上の編集許可を推論することができる。

5 今後の課題

本研究が対象とした双方向変換言語では、いくつかの重要と考えられる変換を取り扱っていない。特に、データの値による条件分岐は実用的なアプリケーションでは重要となる。また編集操作について、今回扱っている文字列の変更、要素の挿入・削除以外に、構造の変更が起きるような操作が考えられる。このような点を含めて、より豊富な変換や編集操作を取り扱うことができ、同時に性質の保証が行いやすいような双方向変換の言語設計について議論していく必要がある。

参考文献

- [1] J. N. Foster, M. B. Greenwald, J. T. Moore, B. C. Pierce, and A. Schmitt. Combinators for bidirectional tree transformations: a linguistic approach to the view update problem. In *ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL), Long Beach, California*, pages 233–246, 2005.
- [2] Z. Hu, S.-C. Mu, and M. Takeichi. A programmable editor for developing structured documents based on bidirectional transformations. In *PEPM '04: Proceedings of the 2004 ACM SIGPLAN symposium on Partial evaluation and semantics-based program manipulation*, pages 178–189, Verona, Italy, 2004. ACM Press.