

帳票間データ処理のための記述モデルの提案と実装¹

3R-8

工藤 道治

日本アイ・ビー・エム(株) 東京基礎研究所

1. はじめに

オフィスでの伝票処理は、単一の伝票処理機能を複数組み合わせることで自動化できる作業が多い[1]。その中でも、伝票データを他の形式の伝票・帳票に転記する処理が全体の約30%を占めることがわかった[2]。本稿では、オフィスにおける転記処理を汎用化して、オブジェクトの内部状態およびユーザーの処理記述によりオブジェクト間のデータ転記仕様を指定するモデルを提案し、帳票オブジェクトの例を用いた具体的な実装例について説明する。

2. データ転記モデル

オブジェクト間でのデータ転送操作をユーザーが簡単に指示するメカニズムとして、カットパッファを使ったコピーアンドペースト操作やダイレクトマニピュレーションによるオブジェクトの移動・複写などがある。しかし、これらの操作は汎用処理としての操作モデルであり、内部データに依存した処理は定義できない。本稿のデータ転記モデル(Data Transcription Model)は、次に示す処理をユーザーが簡単に行うために転記処理を汎用化したものである。

1. 複数オブジェクト間のデータ転記処理
2. 内部データに依存した条件分岐が発生する処理
3. ユーザーインタラクションによって実行する場合と、マクロとして実行する場合の両方が透過的に実行される処理

例えば、一つのディレクトリにある複数ファイルをファイルエクステンションによって分類し、ファイルエクステンション別のディレクトリに複写する場合、copy コマンドやフォルダー間のダイレクトマニピュレーションでは複数回の操作を行う必要がある。その原因は、対象オブジェクトの内部データ(ここではファイルエクステンションの値)によ

って処理内容が変化することにある。このような処理を行うためにはプログラミング作業を必要とするが、本稿ではプログラムすることなしにユーザーインターフェースを通してデータ転記処理を記述できるモデルを提案する。

データ転記モデルは Selection, Transcription, Destination の三つの要素から構成される。Selection はデータ転記の元となるオブジェクトを指定する。Transcription は、データ転記における元オブジェクトと宛先オブジェクトとの関係を指定する。Destination は、宛先となるオブジェクトを指定する。ユーザーは、三つの要素をユーザーインターフェース等により指示することでデータ転記の仕様を指定する。(図1)

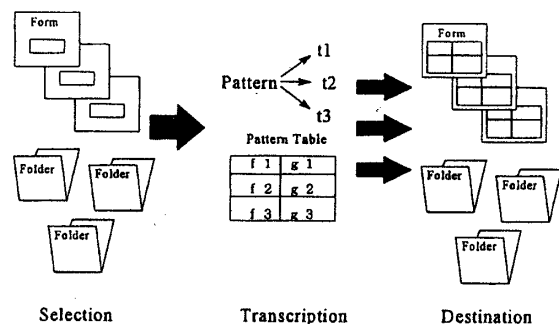


図1 データ転記モデル

2.1 Selection での記述

Selection では対象となるオブジェクトをユーザーが選択するか、オブジェクトの選択式を記述しておく。ここで選択された複数個のオブジェクトは、Transcription 記述に従い全て処理される。

2.2 Transcription での記述

Transcription では、次の項目を指定する。

¹ A Proposal of Generic Data Transcription Model For Office Objects
Michiharu KUDO, IBM Japan Ltd., Tokyo Research Laboratory

- Pattern...選択されたオブジェクトと宛先オブジェクトとの対応関係を指定する(表1)
- Pattern Table...選択されたオブジェクトと宛先オブジェクトとのPropertyの対応関係をテーブルで指定する。(表2)
- Data Transcription Table...Patternで定められた宛先オブジェクトにProperty値を転送する場合に指定する。(表2)

pattern	説明
Self	元オブジェクトそのものを宛先オブジェクトとして転記
CreateOne	宛先オブジェクトを一つだけ生成し転記
CreateAll	元オブジェクト一つに対して宛先オブジェクトを一つ生成して転記
Pattern Table Select	宛先オブジェクトをPattern Tableで検索、対応する宛先オブジェクトがあれば、それに転記、なければ新しく生成し転記

表1 Pattern

	属性	説明
前提条件	Condition	Transcriptionを行う前提条件
元オブジェクトに関する記述	SType	元オブジェクトのプロパティ名。
	SProperty	元オブジェクトのプロパティ
宛先オブジェクトに関する記述	DType	元オブジェクトのプロパティ名。
	DProperty	宛先オブジェクトのプロパティ
	DOperation(Transcription Tableで有効)	処理前の値との演算

表2 Pattern Table, Data Transcription Table

3. データ転記の実例

3.1 ファイルエクステンションによるファイル複写

あるディレクトリにあるファイルをファイルエクステンションの文字列に対応したディレクトリに複写する場合、ユーザーは次の記述を行う。

- Selection: c:\Documents*. *
- Transcription:
 - Pattern:self
 - Pattern table:Condition:1, SType: File Property, SProperty: File Extention, DType: Directory, DProperty: d:\MyDocuments\src(FileExtention)

- Data Transcription Table: None
- Destination: None

上記の指定により、一つのディレクトリにある複数ファイルをファイルエクステンションによって分類し、別のディレクトリに複写する。

3.2 伝票オブジェクト間の転記

Aフォルダに入っている全ての伝票をBankフィールドの値によって分類し、Bフォルダに入っている「銀行別集計表」フォームにTotalフィールドの値を転記する場合、ユーザーは次の記述を行う。

- Selection: SELECT @All in Folder A
- Transcription:
 - Pattern: Pattern Table Select
 - Pattern table: Condition:1, SType: Field, SProperty: bank, DType :Field, DProperty: bankCategory
 - Data Transcription Table: Condition:1,SType: Field, SProperty: Total, DType :Field, DProperty: bankTotal, DOperation: Append
- Destination:SELECT Form="銀行別集計表" in Folder B

4. 実装例

現在、3.2で説明した伝票オブジェクトの処理についてLotus Notes上で実装されている。伝票オブジェクトはNotesのForm、フォルダオブジェクトはNotesのDatabaseに相当する。ユーザーは、本稿で説明した各項目をパネル誘導型のユーザーインターフェースによって入力していくと、最後にデータ転記オブジェクトがNotesのDocumentとして生成される。ユーザーはDocument上に貼られた"Run"ボタンを押すことにより、データ転送を起動することができる。

5. おわりに

オフィスにおける伝票処理を分析し、伝票間のデータ転記処理を本稿のモデルで記述することができた。また、そのモデルをLotus Notes上に実装し、動作の確認を行った。本モデルはデータ転記における汎用的な処理となることを目指しており、現在伝票処理以外の応用分野について検討している。

参考文献

- [1]津田他、帳票処理システムとそのアプリケーションモデル、電子情報通信学会、SIG Office System, Vol. 93, No. 357, pp.19-24, 1993
- [2]工藤他、伝票を用いたオフィス業務の機能要素の分析と構成法、情報処理学会、第48回全国大会、vol.6, pp.223-224, 1994