履歴情報管理システムのための多粒度アノテーション伝播

青戸　了† 清水　敏之†† 増田　耕一††† 吉川　正俊††

† 京都大学 工学部 情報学科
†† 京都大学 情報学研究科 社会情報学専攻
††† 海洋研究開発機構
E-mail: †aoto@db.soc.i.kyoto-u.ac.jp, ††{tshimizu,yoshikawa}@i.kyoto-u.ac.jp, †††masuda@jamstec.go.jp

あらまし 近年，データ処理に関する履歴情報管理の必要性が認知され始めている．履歴情報とは，データが生成される過程において実行された処理や，分析に使用されたデータに関する記録であり，それらはデータ品質の可視化や処理の再現性保証などの目的で利用され，データの系統的管理において不可欠な要素となっている．多様な観測データによって分析を行う地球観測データベースの運用においてもその必要性は高い．我々は履歴情報管理のアプリケーションとして，地球科学データベースのための履歴情報管理システムの構築を目的に議論を行う．その一環として，多粒度で関連付けられた注釈の伝播手法を提案するとともに，実験によって従来手法に対する優位性を示す．
キーワード 多粒度アノテーション，アノテーション伝播，履歴情報管理

On propagation of multi-granularity annotation for provenance management system

Ryo AOTO†, Toshiyuki SHIMIZU††, Kooiti MASUDA†††, and Masatoshi YOSHIKAWA††

† School of Informatics and Mathematical Science Faculty of Engineering, Kyoto University
†† Department of Social Informatics, Graduate School of Informatics, Kyoto University
††† Japan Agency for Marine-Earth Science and Technology
E-mail: †aoto@db.soc.i.kyoto-u.ac.jp, ††{tshimizu,yoshikawa}@i.kyoto-u.ac.jp, †††masuda@jamstec.go.jp

Abstract In recent years, the importance of provenance management is increasing. Provenance is the history of data and history of data processing. Provenance information is used to visualize the data quality and guarantee the reproducibility of data processing, and it is indispensable for managing scientific data systematically. Especially in fields such as earth observation databases which are used to analyze various observational data, the necessity of provenance management system is high. Our goal is to construct a provenance management system for earth observation databases and we propose a propagation technique of annotations related to data by multi-granularity. And we present the superiority of the proposed method by the experiment.
Key words multi-granularity annotation, annotation propagation, provenance management

1. はじめに

近年，データ処理に関する履歴情報管理の重要性が認知され始めている [1]〜[4]．履歴情報とは，出力データを構成する元となる起源データや，生成過程の詳細な記録である系譜情報などを指し，これらの情報は，データに対する理解を深める上で有用な情報であるとともに，データの品質，つまりデータの正確性や妥当性を証明する点においても不可欠な情報である．さらに，データの再現性保証や世代管理などの目的でも利用され，データを利用する側，管理する側のどちらの視点からも重要な情報であるといえる．現状として，このような履歴情報の系統的な管理が行われていることはまれであり，データ管理者の記憶にのみ留められている場合が多い．管理が行われている場合でも，人手によるものがほとんどである．しかし，膨大に存在するデータに対して，履歴情報の詳細な記録を行うことは非常に多くの労力を必要とするため，人手による完全な管理は困難である．さらに，それらは一般的には公開を前提として作られておらず，データ利用者が自由に履歴情報を得ることも不可能である．したがって現在，履歴情報の管理を支援するとともに，データ利用者に対して履歴情報の検索，閲覧を可能とする手段を提供する枠組みが必要となっている．

履歴情報管理に関する従来研究として，由来導出や注釈の付

加と伝播，系譜情報の記録など，様々な研究が行われている．由来導出に関する研究では，出力データに対して静的な分析を行うことで，起源データを追跡することが目的とされており，主にデータベース分野を中心として議論が行われている [5], [6]．注釈に関する研究では，データに関する付加情報を注釈として関連付け，またそれらを，関係演算を通して出力へ伝播させる手法について議論が行われている [7], [8]．また，系譜情報の記録に関する研究では，処理に使用されたデータやプログラム間の依存関係の保持を行う方法について研究が行われている [2], [9]．

地球観測データの蓄積，分析を行う地球観測データベースにおいても履歴情報の保持は重要度の高い問題である．観測データや分析データの品質は，観測値の信頼性や処理過程の妥当性からなり，それらは履歴情報の管理によって証明，確保される．よって，地球観測データベースの運用においても，それらを系統的に管理する機能が必要とされている．関連研究として，高橋ら [10] によって地球観測データを取り扱うためのデータモデルに関する議論が行われ，地球観測データのための注釈モデルが提案された．

本論文は履歴情報管理のアプリケーションとして，地球観測データベースのための履歴情報管理システムの構築を目的に議論を行う．システム構築のための議論として，多様な粒度のデータに対して付与された注釈を処理を通して伝播させる手法を提案する．さらに，実験により従来手法に対する空間コスト，時間コスト面での優位性を示す．本論文の注釈に関する議論は地球観測データモデル [10] にも適応可能である．

以下，構成として，2 節において履歴情報管理の関連研究を示す．3 節においてシステム構築のために要求される機能や構成について議論を行う．4 節において多粒度アノテーションに対する伝播手法を提案する．5 節において従来手法との比較実験と実験結果の考察を行い，最後に，6 節においてまとめと今後の課題に関して述べる．

2. 関 連 研 究

2.1 履歴情報管理

実行された処理や入力パラメータ，また処理に対する入出力データなどに代表される処理過程に関する情報は，一般的に履歴情報（*provenance*）と呼ばれ，それらの管理に関して現在までにさまざまな研究が行われている [1]〜[4]．provenance に関する研究は大きく *workflow provenance* と *data provenance* に関する研究の 2 つに分類される．

workflow provenance に関する研究では，一連の処理を系譜として保存し，過程におけるデータの流れを記録することが目的となっている．系譜情報は一般にグラフ（DAG）を用いて表現され，系譜グラフは，実行された処理や，処理に対する入出力データからなる節点と，節点間の依存関係を表す有向枝から構成される．節点に一意的な識別子を与え，ノード間の依存関係を保持することで，ワークフローの表現が可能となる．一般的に，処理 (プログラムなど) はブラックボックス化され，動的に系譜の記録がおこなわれる．代表的な科学ワークフロー管理システムとして myGrid [11] や Kepler [12] などが挙げられる．

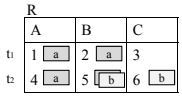

図 1 Buneman らの手法 [7]

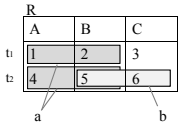

図 2 Gerrts らの手法 [8]

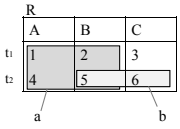

図 3 Srivastava らの手法 [14]

一方，data provenance に関する研究では，前者に比べてさらに粒度の細かい対応関係，つまりデータセルを最小とする単位でのフロー追跡が目的となっている．言い換えれば，あるデータの起源となったデータを求めることに主眼が置かれており，そのような起源データを一般的に由来と呼ぶ．また，data provenance に関する研究は主にデータベースのコミュニティ内で行われており，対象とする処理が関係演算に限定される点が異なる．由来の導出には主に二種類のアプローチが存在する [1]．ひとつは *annotation-based* な手法と呼ばれ，データの一つ一つに識別子を付与し，それらを処理を通して伝播させることで由来情報の保持を行う手法である．代表的な研究として Bhagwat ら [13] によるものがある．もうひとつは *non annotation-based* な手法と呼ばれ，関係演算の逆関数を用いて起源データを逆算する手法である．代表的な研究としては Cui ら [5] によるものがある．Cui らは基本的な関係代数演算の逆関数を定義し，それらを用いて出力を逆変換することで，由来の導出を行った．ここでの逆関数とは，演算に対する入力データや入力パラメータなどの情報から起源データを生成する関数を指す．

2.2 注釈の伝播

注釈の伝播とは，入力データに付与されている注釈を出力データへ継承させることを表す．通常，注釈の関連付け情報は，処理を通すことによって失われるが，付加位置を再計算することで，注釈の伝播を行うことができる．

Buneman らにより，初めてデータベース分野における注釈伝播の議論が行われた [7]．Buneman らのモデルでは，図 1 のように，注釈をセル単位で付加する．つまり，注釈の付加位置を表，組，属性の三つ組を用いて指定する．このモデルに従い，関係代数の各基本演算に関する注釈の伝播規則が定義された．Bhagwat ら [13] により，Buneman らの伝播規則 [7] を利用した由来導出手法が示された．Gerrts らの研究 [8] では Buneman らの注釈モデルを拡張し，セル単位だけでなく，図 2 のように特定の組の任意の属性集合に対して注釈を付加できるモデルが示された．また，注釈付きデータに対する関係代数である *color algebra* を定義することで注釈の伝播計算と問合せを可能とし，さらに，複数属性に付与された注釈を，集合として意味を持つ注釈とそうでないものに分類するなど，注釈の意味を考慮した議論も行われた．Srivastava ら [14] は注釈付加の議論をさらに拡張し，図 3 のように任意の組の任意の属性集合に対して注釈付けを行えるモデルを示した．注釈とデータの関連付けは問合せを利用し，目的の範囲を選択することで行われる．また，従来手法 [7], [8] との性能比較を行っており，範囲指定を用いた注釈管理には空間コストや更新コストにおいて優位性があること

表 1　従来研究との関連

	[7]	[8]	[14]	本研究
付加方法	特定の組の 特定の属性（セル単位）	特定の組の 任意の属性集合	任意の組の 任意の属性集合	任意の組の 任意の属性集合
アノテーション伝播の議論	○	○	×	○

を示した．

表 1 で示されるように，従来研究では明示的に関連付けられた注釈に対する伝播の議論は行われているが，範囲指定された注釈の伝播に関する議論は行われていない．優位性のある範囲指定型の手法を用いて，注釈の伝播計算が行えれば，従来の伝播手法に対して空間コストや時間コスト面での性能改善が可能であると考えられる．よって本論文では，第 4 章においてその実現方法について議論し，さらに，第 5 章において提案手法の評価を行う．

3.　履歴情報管理システム

本研究では，履歴情報管理のアプリケーションとして，地球観測データベースのための履歴情報管理システムの構築を目的としている．よって本章ではまず，地球観測データの利用に際して管理すべき履歴情報とそこから考えられる要求について議論し，その後，現在想定している履歴情報管理システムの構成について概説する．

3.1　システムへの要求

地球観測データを利用する上で，具体的には以下のような履歴情報が生じる．

（1）　データが有する情報

- データ取得に関する記録：

観測データは研究機関や個人から提供される．よってデータの取得元，提供元，取得日時，責任者などの情報が生じる．

- データの生成過程に関する記録：

観測データを統合，分析することで新たなデータが生成される．よって生成されたデータに対して，材料となったデータや使用された手続きに関する記録が生じる．また，データが生成された時刻，実行環境，作成者などの情報も記録すべき情報として挙げられる．処理によって発見された問題点など，データに対して付加される情報も存在する．

（2）　処理が有する情報：

実行された処理の仕様や，入力パラメータ，実行時刻，実行者，計算機環境などの情報が生じる．

以上の履歴情報を管理するために，具体的に以下に挙げるような要求が生じる．

（1）　系譜情報の保存：

あるデータの生成にはどのような処理が実行され，どのようなデータが使用されたのかという情報を保存したい．

（2）　メタデータの付加：

データや処理に関する情報をメタデータを付加することで保存したい．さらに，データが持つ情報の一部を出力にも継承し，出力からも情報が得られるようにしたい．

また，データの修正や問題箇所の確認などを目的として，起源データの特定も要求のひとつである．

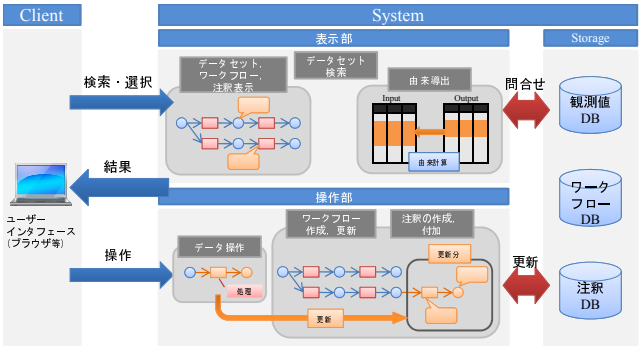

図 4　システムの概要図

3.2　想定されるシステムの概要

想定されるシステムの概要図を図 4 に示す．システムの利用はブラウザ等のユーザーインターフェースを通して行うことを想定している．システムは大きく分けて三つの部分から構成され，データセット表示部，データセット操作部，記憶装置からなる．記憶装置は観測値データベース，ワークフローデータベース，注釈データベースからなる．観測値データベースには地球観測データが格納され，ワークフローデータベースには系譜情報，注釈データベースには注釈情報が格納される．

データセット表示部は主にデータ利用者の問い合わせに基づき地球観測データの検索，表示を行う．データセットの検索時には，同時に系譜データベースと注釈データベースへ問い合わせを行い，該当データセットの系譜情報と注釈情報を収集する．それらの情報を統合し，データセットと同時にその生成過程や注釈に関する情報を利用者に提示する．また，由来を求めたいデータを選択することにより，系譜情報から由来計算を行い，起源データの提示を行う．

データセット操作部ではデータに対する処理の実行と処理過程の記録を行う．処理の例としては，観測データの誤りを訂正する処理や，独自のプログラムを用いた観測データの分析などが挙げられる．データ処理が行われると出力データが観測値データベースに格納され，同時に系譜データベースが更新されることで処理過程の記録が行われる．実行時刻や実行者などの情報は自動的に記録されるが，処理の仕様などの情報は実行者に記入してもらう必要があるため，入力を促す．またこのとき，注釈の伝播計算も行われ，入力データに付加されている注釈は処理後のデータに伝播される．伝播された注釈情報は注釈データベースに格納される．データ処理は基本的にはユーザーインターフェースを通してシステム上で行われることとなる．その場合は実行内容は自動的に記録される．また，データ処理がローカル環境で行われた場合は，利用者が出力データをアップロードし，処理内容に従って系譜情報と注釈情報の更新を行う．

以下ではシステム構築の実現に向け，注釈の伝播に焦点を絞り議論を進める．

4.　多粒度アノテーション伝播

注釈とは，一般にデータに付加されるコメントのようなものを表し，説明文章の付与やデータ自身に対する情報の保持など

に利用される．特に我々が想定する地球観測データベースにおいては，注釈の利用により例えば以下に述べるような利点が生じる．注釈として頻繁に使われる用途として，欠損値の報告が挙げられる．観測データには機器の故障，観測方針の変更，人的な誤りなどの様々な理由で欠損が生じる．欠損情報は原データには欠損値として記録されている．しかし，これらの情報は一度処理を通すと失われてしまい，通常は出力データからその存在を知ることができない．そこで，欠損情報を注釈としてデータに付加し，それらを処理を通して出力データへ伝播させることで，本来失われてしまう情報を最終的な出力にまで継承させることが可能となる．これにより，出力データ単体から起源データの欠損情報を得ることができる．注釈を使わずとも由来導出を利用し，対応する原データまでさかのぼれば欠損を確認できるが，通常そのような確認を逐一行うことは現実的ではない．また，コストの面でも由来計算を行うよりも注釈の伝播を利用する方が有利である．他にもデータの作成日や作成者などの情報，生成過程で行われた処理に関する情報なども注釈を利用することで保持が可能となる．

4.1 注釈モデル

多様な粒度を持つデータに対する注釈付与モデルは以下の通りである．注釈はコメントと付加条件の二つ組を用いて表現される．付加条件とは注釈が関連付けられる範囲を指定するもので，表，選択条件式，属性集合の三つ組からなる．ここで，選択条件式とは注釈付けを行う組を指定する論理式である．つまり付加条件は，表を指定し，選択条件式によって組を選択，さらに属性集合を指定することで決定される．これらを設定することで，任意の範囲に注釈を関連付ける．注釈は以下の式で表現される．

$$A = (a, l) = (a, (R, T_c, \mathcal{A}))$$

A, a, l はそれぞれ，注釈，コメント内容, 付加条件を表す．また，l は $R, T_c, \mathcal{A}$ を使って表現され，それぞれ表，選択条件式，属性集合を表す．この注釈モデルは，表現は異なるが Srivastava らのモデル [14] と等価である．

例として図 5 の表 R に対する注釈付けを考える．表 R は従業員の名簿であり，属性として ID，名前，年齢，都市，電話番号を持つ．以下，それぞれの属性を省略して I, N, O, C, T と表すこととする．例えば京都市に住んでいる 27 歳以上の従業員の名前と電話番号に対して何らかのコメント a を付加するには，以下のように注釈 A を作成する．

$$A = (a, (R, (C = \text{"}kyoto\text{"}) \wedge (O \geq 27), NT))$$

図 5 における影付きの部分に注釈 A が関連付けられる．

多粒度注釈モデルの利点は，条件を設定することで範囲指定による柔軟な注釈の付与が行える点である．それによって複数のデータに対して機械的に注釈の関連付けを行うことができ，生成を自動化できる．また特徴として，注釈の動的な関連付けが可能な点が挙げられる．条件を満たすデータが挿入された場合，それらは自動的に参照対象に追加されるため，注釈の関連付けを再度行う必要がないという利点が生じる．しかし，注釈

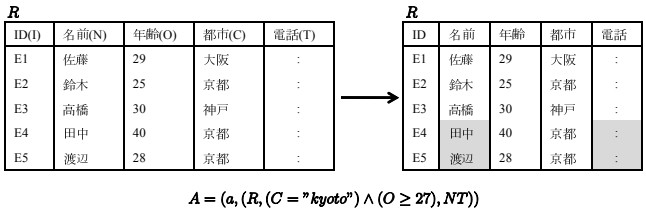

図 5 注釈の例

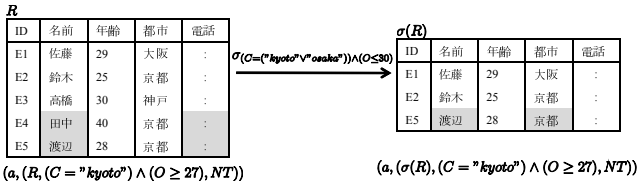

図 6 選択に関する注釈伝播

の生成時に設定された付与対象を固定したいという要求も存在する．よって，我々は注釈を静的に関連付ける方法についても議論する．

4.2 伝播モデル

本節では，前節で述べた多粒度注釈モデルに対する伝播規則を定義する．伝播規則とは注釈の付加範囲の変化を表現するもので，規則に従って注釈の選択範囲を更新することで伝播計算を行う．対象となるのは基本的な各関係演算 (選択，射影，直積，和，差，属性名変更) である．各演算に対する伝播規則は以下のようになる．左辺が入力の注釈を表し，右辺が伝播後の注釈を表す．

- **選択**：$\sigma_c(R)$

$$(a, (R, T_c, \mathcal{A})) \rightarrow (a, (\sigma_c(R), T_c, \mathcal{A}))$$

入力の選択条件式 T_c がそのまま出力へ適応される．選択演算の出力にはパラメータの選択条件を満たさない組は存在せず，新たな組が追加されることはないため，選択条件式を変化させずに注釈の伝播が可能である．

例として，表 R から大阪または京都に住んでおり，年齢が 30 歳以下のデータを選択した場合を考える (図 6)．結果は $E5$ のデータが注釈の参照先として残り，図 6 のように注釈が付加される．

- **射影**：$\pi_B(R)$

$$(a, (R, T_c, \mathcal{A})) \rightarrow (a, (\pi_B(R), T_c, \mathcal{A} \cap B))$$

射影演算による属性集合の変化に合わせて，付加範囲の属性集合を変更する．これは注釈の属性集合 $\mathcal{A}$ とパラメータ B の共通集合を取ることで行う．ここで，選択条件式を構成する属性が射影によって消去された場合，注釈の付加範囲を決定する情報が欠落してしまう．よって，射影によって切り出す属性のほかに，選択条件式を構成する属性も出力へ保持することで，付加条件の保存を行う．保存された属性集合は利用者からは不可視な属性として扱われる．

例として，表 R から ID と名前を射影する操作を考える (図 7)．通常ならば ID と名前のみが残るが，条件式を構成す

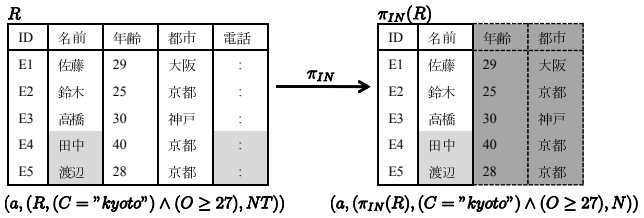

図 7　射影に関する注釈伝播

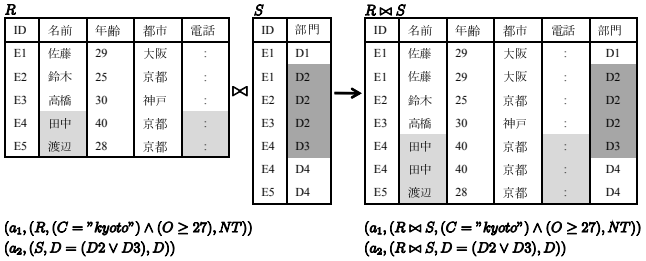

図 8　結合に関する注釈伝播

る年齢と都市も出力へ保存しておく．そうすることで，注釈の付加条件を保持しつつ射影演算を行うことができ，図 7 のように注釈が伝播される．

- **結合**：$R_1 \bowtie R_2$

$$(a_1,(R_1,T_{c_1},\mathcal{A}_1)) \rightarrow (a_1,(R_1\bowtie R_2,T_{c_1},\mathcal{A}_1))$$
$$(a_2,(R_2,T_{c_2},\mathcal{A}_2)) \rightarrow (a_2,(R_1\bowtie R_2,T_{c_2},\mathcal{A}_2))$$

結合条件を満たさない組は結果に表れないため，選択条件式をそのまま適応することで注釈の伝播が可能である．

表 R と従業員の所属する部門を表す表 S との結合を例として考える (図 8)．表 S にも異なる注釈が付加されている．表を結合し，選択条件を出力へ継承することでそれぞれの注釈が出力へ伝播される．

- **和**：$R_1 \cup R_2$

$$(a_1,(R_1,T_{c_1},\mathcal{A})) \rightarrow (a_1,(R_1\cup R_2,T_{c_1},\mathcal{A}))$$
$$(a_2,(R_2,T_{c_2},\mathcal{A})) \rightarrow (a_2,(R_1\cup R_2,T_{c_2},\mathcal{A}))$$

付加条件を変化させることなく，注釈の伝播が可能であるが，和演算によって注釈の選択条件式を満たす組が新たに加わった場合，それらは自動的に注釈の付加対象となる．もともと注釈付けされていたデータ以外に注釈の関連付けを許すかどうかは，ユーザーが選択できることが望ましい．そこで，以下では注釈の静的な関連付けについて議論する．

静的な関連付けを実現するには，注釈の生成時に存在したデータを記憶する必要がある．それは，次の方法で実現可能である．ユーザーから不可視な属性（ここでは E とする）を一つデータ表へ追加し，データ表の名称を注釈の生成時に格納する．さらに，注釈の選択条件式に属性 E が注釈生成時のテーブル名と等しい値を持つという条件を追加する．つまり，注釈 $(a,(R,T_c,\mathcal{A}))$ が表 R に対して生成された場合，それは $(a,(R,T_c\wedge(E=R),\mathcal{A}))$ と書き換えられる．以上の操作を注釈の生成時に行い，追加した条件を満足するかどうかを注釈の参照時に検査することで，生成時に存在したデータのみを選択

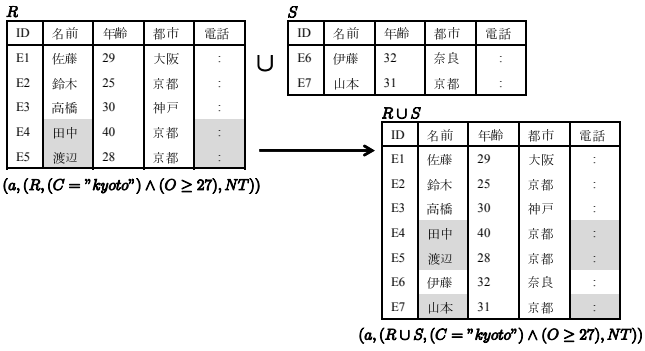

図 9　和に関する注釈伝播

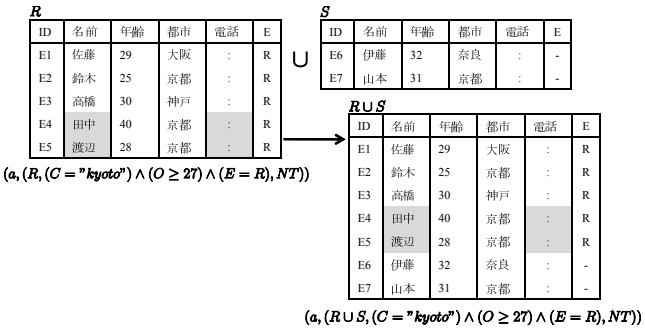

図 10　静的な関連付け

できるため，注釈の静的な付与が可能となる．

和演算の例として図 9 のような操作を考える．入力の注釈が伝播されるとともに，付加条件を満たしている $E7$ のデータも注釈の参照対象に追加される．もともとの付加対象となっていた $E4$ と $E5$ のデータのみを参照対象とするには，図 10 のように注釈生成時のテーブル名を記録し，選択条件式に条件を追加する．条件が加わったことにより，参照対象が表 R に存在したデータであるかが検査され，静的な関連付けが可能となる．

注釈付与対象の属性集合が等しく，さらに入力の注釈が同一とみなせる場合は，それらを統合でき，以下のように表される．

$$(a,(R_1,T_{c_1},\mathcal{A}))$$
$$(a,(R_2,T_{c_2},\mathcal{A})) \rightarrow (a,(R_1\cup R_2,T_{c_1}\vee T_{c_2},\mathcal{A}))$$

- **差**：$R_1 - R_2$

$$(a,(R_1,T_{c_1},\mathcal{A})) \rightarrow (a,(R_1-R_2,T_{c_1},\mathcal{A}))$$

差によって除かれる組は出力に出現しないため，付加範囲を変化させることなく，出力へ適応することで伝播が行える．

例として表 R と表 S の差を取る操作を考える (図 11)．結果として $E4$ のデータが注釈の参照先として残り，入力の付加範囲を出力へ適応させることで注釈が伝播されている．

- **属性名変更**：$\delta_\theta(R)$

$$(a,(R,T_c,\mathcal{A})) \rightarrow (a,(\delta_\theta(R),\theta(T_c),\theta(\mathcal{A})))$$

属性名変更に伴って，注釈の属性集合と選択条件式に現れる属性名を変更する．

例として名前，都市，電話の属性名をそれぞれ苗字，住所，内線に変更する操作を考える (図 12)．属性名の変更に伴い，付

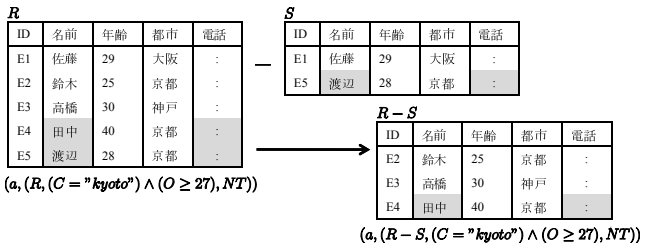

図 11　差に関する注釈伝播

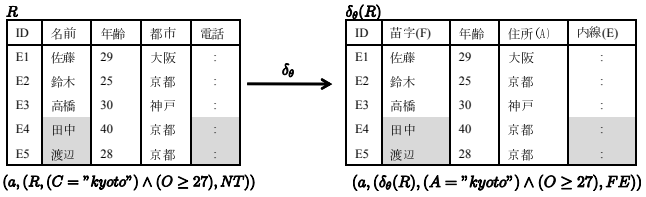

図 12　属性名変更に関する注釈伝播

加条件の選択条件式と属性集合の属性名を変更することで伝播が行われる．

5. 実　　験

本章では，我々の提案する注釈伝播モデルの性能評価を行う．性能評価は，注釈の保持に要する記憶量と，注釈伝播時の計算時間を従来手法と比較することで行う．

評価実験を行うため，提案手法に基づき，注釈管理システムのプロトタイプを実装した．プロトタイプ実装は MySQL を用いて行い，次のような計算機環境上で実験を行った: OS:Windows Vista Business 32bit，CPU:Intel Core 2Duo CPU 1.2GHz, RAM:4GB.

我々の作成したシステムでは，図 13 のように各データ表に対して注釈表と属性リスト表を作成することで注釈の保持を行う．注釈表は注釈の付加範囲とコメントを保持する表であり，属性リスト表はデータ表の属性名と属性識別子の対応を保存する表である．注釈の付加範囲は注釈表の属性識別子列と選択条件式属性から決定される．属性識別子列はブール型であり，対応する属性に注釈が付加されるなら 1，そうでないならば 0 が格納される．また，選択条件式属性は注釈の付与対象となる組を指定するための属性であり，図 13 のように論理式を値として持つ．以上のような表をデータ表に付随して作成することで，注釈を任意の範囲に対応付けることが可能となる．注釈の伝播は前章で定義した伝播規則に従って行う．実際の伝播計算はデータ表への問合せを実行するとともに，問合せの書き換えを行い，それらを注釈表と属性リスト表へ適応することで行う．例えば，射影や属性名変更が実行される場合は属性リスト表を更新し，和の場合は注釈表の和をとることで伝播計算を行う．注釈の静的な関連付けに関しては，テーブル名の保持を行う表を別に作成し，注釈を参照する際にデータ表と結合させることで実現する．また，選択や結合，差によって無効な (どのレコードにも関連付けされない) 注釈が生じる場合があるが，検査を行うことで除去できる．この検査は問合せの実行時ではなく，必要に応じて品質管理の一環として行える．

性能比較の対象として，従来研究で提案された注釈管理システムである DBNotes [13] と MONDRIAN [8] を選択し，これらについても独自に実装を行った．DBNotes では，図 14 のようにデータ表の各属性に対して追加列を作成し，そこに注釈を格納する．一つのレコードに複数の注釈が関連付けられる場合は，注釈毎にレコードを複製し，その追加列に注釈を格納する．そのため，同一のデータに複数の注釈が付加される場合はデータレコードの重複が発生する．図 14 では $(A, B) = \{(1, 2), (3, 4)\}$ のそれぞれに注釈が付加されており，$(1, 2)$ には注釈 a が，$(3, 4)$ には注釈 b, c が格納されている．ここで，同一のセルに複数の注釈が存在するため，$(3, 4)$ が重複して格納されいてる．注釈はデータに明示的に関連付けられているため，データ表に対する問合せにより，注釈伝播も自動的に行われる．

MONDRIAN では，注釈はデータレコードごとに対応付けられ，図 15 のように別の表に分離して保持される．図 15 のデータ表は図 14 と同様の注釈が付加されている．注釈表はコメントを格納する属性に加えて，データ表の各属性に対応するブール型の属性を持ち，同名の属性に注釈が付加されるならば 1 が，付加されないならば 0 が格納される．DBNotes との相違点は，注釈が任意の属性集合を被覆できる点である．もともとの MONDRIAN では，データレコードと注釈の対応は，注釈表にデータレコードそのものを格納することで保持するが，他手法との公平な比較を行うため，我々の実装では MONDRIAN を独自に拡張した構造を採用しており，データレコードと注釈との対応を識別子を用いて保存している．MONDRIAN も DBNotes と同様に，注釈の付加が明示的に行われている．よって，注釈の伝播はデータ表と注釈表を結合後，データ表に対する問合せを実行することで自働的に行われる．

実験用のデータセットとして，タイ気象局 3 時間間隔値地上気象データ [15] を使用した．データ表はデータセット名，年，月，日，時刻，時間座標，観測値の七属性からなり，185917 レコードを有する．また，付加範囲とコメントを持つ注釈を，ランダムに 500 個から 3000 個まで 500 個間隔で生成し，各手法の格納方法に従って付加することで，合計 18 個のテストデータを生成した．今回の実験では，ランダムな 50 個の英数字からなる文字列をコメント内容として使用した．性能比較は各テストデータに対し，以下の三つの問合せを実行することで行った．

問合せ 1　選択演算：データセットから，時刻が 7 時から 19 時までかつ月が 1 月から 6 月までのデータを選択．

問合せ 2　射影演算：データセットから，月，日，時刻，観測値の四属性を射影．

問合せ 3　選択+射影演算：上記の問合せを組み合わせたもの．

5.1 記　憶　量

注釈保持に使用される記憶量を各注釈管理システム間で比較した．公平な比較を行うため，DBNotes の空間コスト測定には注釈が付与されたデータ表の容量を用い，MONDRIAN と我々のシステムの測定には注釈表とデータ表の容量を加算した数値を用いた．

データ表に付与された注釈の空間コストは図 16 のようになった．また，各問合せ実行後のデータ表に対する注釈の空間コスト

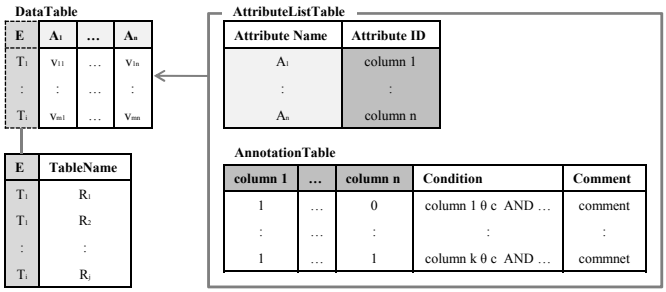

図 13　注釈の保存方法

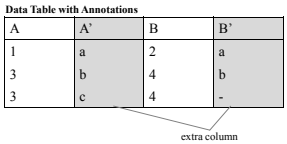

図 14　DBNotes

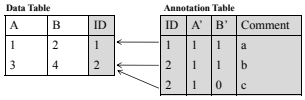

図 15　MONDRIAN

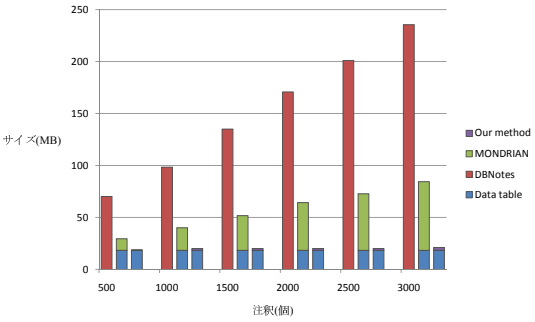

図 16　元テーブルに付加された注釈のサイズ

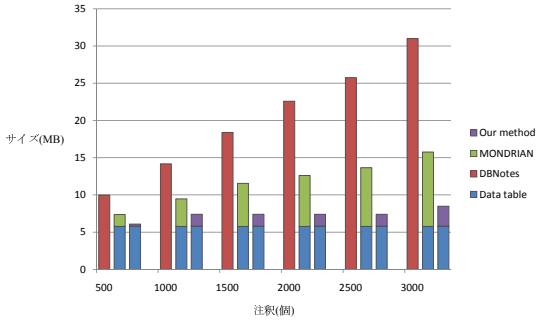

図 17　問合せ 1 実行後の注釈サイズ

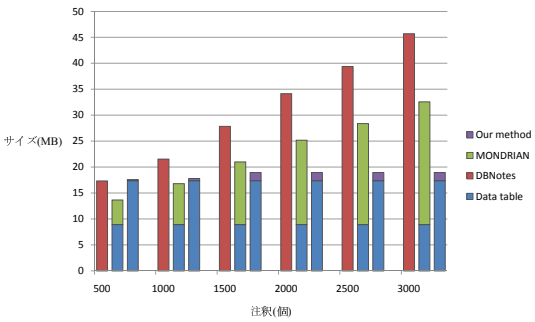

図 18　問合せ 2 実行後の注釈サイズ

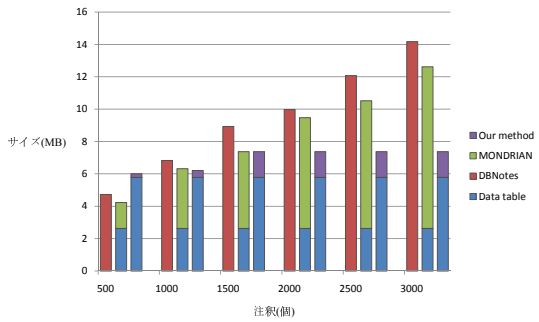

図 19　問合せ 3 実行後の注釈サイズ

はそれぞれ図 17，図 18，図 19 のようになった．図 16 から図 19 の Data table はデータ表の容量を表しており，MONDRIAN と我々の手法はデータ表と注釈表の容量を加算した値を図示している．なお，我々の手法は，注釈の付加方法が文献 [14] と等価であるため，図 16 において見られた優位性は文献 [14] において既知の結果である．

実験結果から，我々の手法は従来手法に比べ，ほぼすべての場合において空間コストが抑えられていることがわかる．この結果は，注釈の保存方法の違いが原因であると考えられる．DBNotes では注釈がセルごとに付加されるため，注釈内容が組ごと，属性ごとに繰り返し保存される．また，MONDRIAN では注釈内容が，参照している組ごとに繰り返し保持される．つまり，注釈が関連付けられるレコード数が増えれば増えるほど同一の内容のコメントが複数回保持されることとなり，空間コストの増加につながっている．対して我々の手法では，各注釈を別の表に分離し，それぞれの内容を一度づつ保持するため，従来手法のような注釈内容の重複が発生しない．よって実験結果のような差が表れたと考えられる．

問合せ 2（射影演算）と問合せ 3（選択+射影演算）について，注釈数が 1000 以下の場合では，我々の手法の空間コストが従来手法よりも大きくなっている（図 18, 図 19）．これは，我々の射影伝播規則の特性によるものである．我々の手法では，選択条件に使用されている属性は射影によって削除されず，出力へ保存される．対して従来手法では，射影は通常通り行われるため，我々の手法では，射影を含む問合せの出力サイズは従来手法と比較して大きくなる傾向にある．実験に用いた問合せは，七属性のうち四属性を射影するものであるが，我々の手法ではそれらに加えてデータセット名，年の二属性が出力データへ保持されている．よって，出力データ表の容量は増大し，注釈数が少ない場合には従来手法よりも空間コストが大きくなるという結果となった．

5.2　問合せ実行時間

注釈の記憶量に加え，各注釈管理システムの問合せ実行時間を比較した．DBNotes の計測には，注釈が付与されたデータ表への問合せ時間を用い，MONDRIAN と我々の手法の計測にはデータ表への問合せ実行時間と注釈表への処理時間を加算した値を用いた．

各問合せに対する実行時間はそれぞれ図 20，図 21，図 22 の

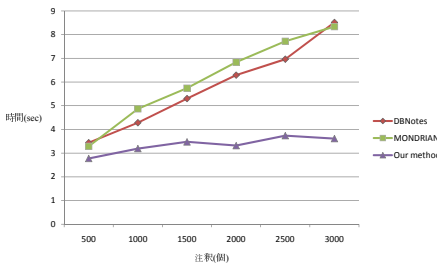

図 20　問合せ 1 の実行時間

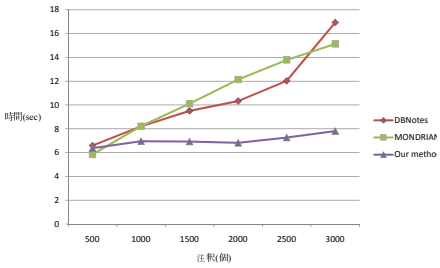

図 21　問合せ 2 の実行時間

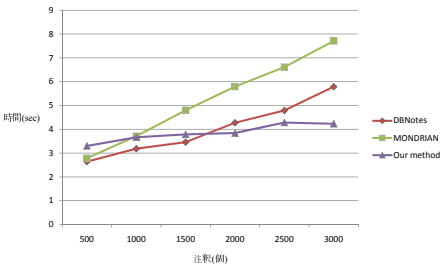

図 22　問合せ 3 の実行時間

ような結果となり，我々の手法は多くの場合で従来手法に比べて高速であった．

実験結果の傾向として，従来手法では注釈数に比例して処理時間が増加しているのに対し，我々の手法では大きな変化が見られない．これは，注釈の格納方法と伝播方法が関係していると考えられる．従来手法では，注釈数が増えていくにつれて複数回保存される注釈が増加し，DBNotes ではデータ表の容量が，MONDRIAN では注釈表の容量が増加していく．よって，データサイズの増加に伴い，実行時間に結果のような比例関係が表れたと考えられる．対して我々の手法では，注釈表のサイズが注釈の参照レコード数に左右されない格納方法をとっている．また，伝播計算も基本的には注釈表を複製する操作からなるため，実行時間に大幅な増加傾向が現れなかったと考えられる．

問合せ 3(選択+射影演算) の実行時間において，注釈数が 1500 以下の場合では従来手法の方が高速であった．DBNotes がこのような条件下で高速であったのは，単純に注釈が少ないため，データ表に対する問合せ時間がデータ表への問合せ時間と注釈表に対する操作時間の和を下回ったことが原因であると考えられる．また，MONDRIAN が我々の手法より高速であったのは，注釈数が少ない場合は，注釈表への問合せが注釈表の複製時間を下回ったことが原因であった．

6. おわりに

本研究では，履歴情報管理システムの構築に向け，必要となる要素や機能，その構成について議論した．また，その一環として注釈管理手法の提案を行った．提案した手法は，多様な粒度を持つデータに対してアノテーションを付与し，それらを処理を通して出力へ伝播させるものである．さらに，提案手法の優位性を比較実験によって確認した．本研究によって得られたおもな結論は以下のとおりである．

（1） 伝播規則を定義することで，多様な粒度によって付加された注釈の伝播を可能とした．

（2） 多粒度の注釈を用いることで，注釈を格納するための空間コストと，注釈を伝播させるための時間コストが改善された．

本論文では多様な粒度で付与された注釈の伝播を行うために基本的な関係演算に対して伝播規則を定義したが，発展的な課題として集約演算に関しても同様の議論が必要であると考えられる．さらに，多粒度注釈への問合せも課題として挙げられる．今後はこれらの議論と並行して，本論文では取り扱わなかった由来導出や系譜保存についても議論を行い，履歴情報管理システムの構築を目指す．

謝辞　本研究は，文部科学省委託業務研究費国家基幹技術「データ統合・解析システム」の支援を受けており，ここに記して謝意を表します．

文　献

[1] P. Buneman and W. C. Tan: “Provenance in databases”, In SIGMOD, pp. 1171–1173 (2007).

[2] S. B. Davidson, S. C. Boulakia, A. Eyal, B. Ludäscher, T. M. McPhillips, S. Bowers, M. K. Anand and J. Freire: “Provenance in scientific workflow systems”, IEEE Data Eng. Bull., **30**, 4, pp. 44–50 (2007).

[3] W. C. Tan: “Provenance in databases: Past, current, and future”, IEEE Data Eng. Bull., **30**, 4, pp. 3–12 (2007).

[4] Y. Simmhan, B. Plale and D. Gannon: “A survey of data provenance techniques”, Computer Science Department, Indiana University, Bloomington IN, **47405**, .

[5] Y. Cui, J. Widom and J. L. Wiener: “Tracing the lineage of view data in a warehousing environment”, ACM TODS, **25**, 2, pp. 179–227 (2000).

[6] A. Woodruff and M. Stonebraker: “Supporting fine-grained data lineage in a database visualization environment”, In ICDE, pp. 91–102 (1997).

[7] P. Buneman, S. Khanna and W. C. Tan: “On propagation of deletions and annotations through views”, In PODS, pp. 150–158 (2002).

[8] F. Geerts, A. Kementsietsidis and D. Milano: “Mondrian: Annotating and querying databases through colors and blocks”, In ICDE, p. 82 (2006).

[9] D. Holland, U. Braun, D. Maclean, K. Muniswamy-Reddy and M. Seltzer: “Choosing a data model and query language for provenance”, In IPAW (2008).

[10] A. Takahashi, M. Tatedoko, T. Shimizu, H. Kinutani and M. Yoshikawa: “Metadata management for integration and analysis of earth ovservation data”, Journal of Software, **5**, pp. 168–178 (2010).

[11] T. M. Oinn, M. Addis, J. Ferris, D. Marvin, M. Senger, R. M. Greenwood, T. Carver, K. Glover, M. R. Pocock, A. Wipat and P. Li: “Taverna: a tool for the composition and enactment of bioinformatics workflows”, Bioinformatics, **20**, 17, pp. 3045–3054 (2004).

[12] S. Bowers and B. Ludäscher: “Actor-oriented design of scientific workflows”, In ER, Vol. 3716 of Lecture Notes in Computer Science, pp. 369–384 (2005).

[13] D. Bhagwat, L. Chiticariu, W. C. Tan and G. Vijayvargiya: “An annotation management system for relational databases”, In VLDB, pp. 900–911 (2004).

[14] D. Srivastava and Y. Velegrakis: “Intensional associations between data and metadata”, In SIGMOD, pp. 401–412 (2007).

[15] “Thai meteorological department”. http://www.tmd.go.th/.