

半永続データのモデル化及び自己組織化管理の諸問題の考察

成 凱[†] 岩井原 瑞穂[‡] 牛島 和夫[†]

[†]九州産業大学情報科学部 〒813-8503 福岡県福岡市東区松香台 2-3-1

[‡]京都大学情報学研究科 〒606-8501 京都市左京区吉田本町

E-mail: [†]{chengk,ushijima}@is.kyusan-u.ac.jp, [‡]iwaihara@i.kyoto-u.ac.jp

あらまし データ工学ではこれまで研究対象としたデータは永続データ（明示的に変更・削除しない限り保存し続けるデータ）、ストリームデータ（大量で高速に到来し続けるデータの流れ）に限られている。一方、キャッシュデータ、検索エンジンデータ、一部の機密データ、個人・組織の日常扱うデータの多くは、一定の期間内しか有効ではないので、寿命付きデータと扱う必要がある。本研究では、このように明示的或は暗黙的に寿命が付けられ、期限内のみ保存されるデータを「半永続データ（Semi-Persistent Data）」と捉え、その性質の解明とモデル化を図る。本論文では、半永続管理システム（SPDMS：Semi-Persistent Data Management System）を目指して、半永続データのモデル化に関わる問題、寿命のモデル化、構造化と永続性、自己組織化を考察し、それぞれの問題の解決方法を提案した。

キーワード 半永続データ、データモデリング、ストリームデータ、寿命付きデータ

Issues in Modeling and Management of Semi-Persistent Data

Kai CHENG[†] Mizuho IWAIHARA[‡] and Kazuo USHIJIMA[†]

[†] Faculty of Information Science, Kyusan University, 2-3-1 Matsukadai, Higashi-ku, Fukuoka, 813-8503
Japan

[‡] Graduate School of Informatics, Kyoto University, Yoshida, Sakyo-ku, Kyoto, 606-8501 Japan

E-mail: [†]{chengk,ushijima}@is.kyusan-u.ac.jp, [‡]iwaihara@i.kyoto-u.ac.jp

Abstract Research in data engineering has been dealt with two categories of data. One is persistent data, data that will be stored permanently until being explicitly deleted or modified. The other is stream data, time-stamped data that arrive at varying rates and will not be stored at all for further access. However, there is another important kind of data that cannot fall into either of these categories, for example, cache data, data collected by search engine crawlers, secret data, which will be stored but will not fix in disks for every. We call data with definite lifetime semi-persistent data. Aimed to develop a complete new Semi-Persistent Data Management System (SPDMS), we will investigate a number of issues concerned with semi-persistent data modeling, self-organization, and characteristics of structured, semi-structured and unstructured semi -persistent data. In conclusion, we outline a research agenda for the new promising field.

Keyword Semi-Persistent Data, Data Modeling, Stream Data, Data Lifecycle Management

1. はじめに

インターネットの帯域幅を節約するために、キャッシュが広く使われ大量のデータがキャッシュディスクに蓄積されるようになってきている。本来、キャッシュは限られたメモリしか持たず、蓄積されたデータが比較的少なく、しかも頻繁に置き換えられると想定し、データベースのように永続的に蓄積して利用することは考えられていない。しかし、近年、RAID や SAN を

代表とするストレージ技術の発展に伴い、キャッシュに使えるディスクスペースが数 TB まで拡大され、キャッシュにとって最も重要な制限が緩和されつつあり、大量のデータがより長期間にわたってディスクに蓄えることが可能になっている。これからも、大容量のストレージ設備が更に低価格で入手できるようになっていくと思われ、キャッシュデータのような寿命付きデータを如何に活用していくのかは重要な課題である。

寿命付きデータはキャッシュデータだけでなく、検索エンジンが集めてきたデータは時間が経つことにつれ古くなっていくため、寿命付きのデータと考えられる。また、個人や組織内部が日常に扱うデータのうち、一生保存すべきものが少なく、ほとんど年月が経つことにより価値がなくなるデータである。更に、会議期間中に出された機密データは一定の期限に限って見られるが、期限が過ぎると自動的に消えると望まれている。

これより更に拡大していくと、すべてのデータはライフサイクルがある。データの量が膨大化になっていくにつれ、そのライフサイクルの管理は重要性が増す。

しかし、データ工学でこれまで扱われてきた主なデータは**永続データ**であり、明示的に削除・変更しない限りそのまま永続的に保存し続ける。永続性を維持するために、バックアップ/リカバリーをはじめ、様々な技術の開発を積み重ねてきた。そして、つい最近、ネットワーク利用関連のデータ、センサーデータ、サーバアクセスログ、電話接続記録など、大量で高速に到来し続ける**ストリームデータ**に注目が急速に高まっている。ストリームデータは永続データと違い、一般に全て保存する必要がなく、一度処理が済むと二度とアクセスすることができない。

ストリームデータの寿命は0であれば、永続データの寿命は無限大(∞)と考えられる。寿命 ∞ の永続データに対して、データベースを中心とした技術は盛んに研究をおこなってきた。寿命0のストリームデータに対する研究もつい最近ホットなトピックとなっており、様々な研究開発を展開している [2]。しかしながら、寿命(0, ∞)のデータを対象とした研究は著者らの知る限りまだ体系的に行っていない。

データ	定義	寿命
永続データ	明示的に変更・削除しない限りそのまま保存し続けるデータ	∞
ストリームデータ	高速に到達し一度しか見られないデータの流れ	0
半永続データ	明示的或は暗黙的に寿命が付けられ、期限内のみ保存されるデータ	(0, ∞)

表 1. 各種データとその寿命

本研究では、明示的・暗黙的に(0, ∞)の寿命が付けられ、寿命期限内に限って保存するデータを**半永続データ**と呼び、それぞれのデータの関連と違いを表1にまとめた。

本論文では、半永続データ管理システム SPDMS (Semi-Persistent Data Management System) の開発を目指して、半永続データの性質の解明とモデル化にかかわる諸問題と可能な解決方法を検討する。2 節は寿命

のモデル化と寿命を管理する手法について述べる。続いて3 節では、構造化程度から見た半永続データのモデル化に解決する必要な問題を検討し、半永続データ管理システム SPDMS の実現方法を提案する。

2. データの寿命とそのモデル化

半永続データをモデル化するために最も重要なのは「寿命」のモデル化と考えられる。

2.1. データの寿命

寿命とは「1. 生命の存続する期間。特に、あらかじめ決められたものとして考えられる命の長さ。2. 物の使用に耐える期間。また、その限界」である。データの寿命は以下のように定義できる。

1. **存続する期間 (L 寿命)**: データのあらかじめ決められた存続する期間。期間を超えると、データが**自動的に消える**。例えば、秘密データの寿命。Lifetime の意味から、このような寿命を **L 寿命** という。
2. **使える期間 (V 寿命)**: 有効に使える期間。その期間を超えると、データが無効になり使えなくなる。データ自体は消えるかどうか、他の条件と合わせて決定する。例えば、検索エンジンデータの寿命。Validation の意味から、このような寿命を **V 寿命** という。

一般に、各データに L と V の二種類の寿命が同時に持つ。(L, V) のように表わせる。また、寿命は期間で表現することもできるし、例えば、「1 ヶ月」、と期限で表現することもできる、例えば、「2040 年 10 月 1 日まで」

2.2. 寿命の種類

寿命は以下の種類ある。それぞれの決め方も異なる。まず、寿命とデータの構造との関連の有無によって絶対寿命と相対寿命に分類できる。

- **絶対寿命(mandatory)**: ほかのデータに依存せず、自らの決められる寿命。例えば、秘密データは受け取った後3時間以内に消すと寿命を設定すると、システムが自動的に消してくれる。
 - **相対寿命(reciprocal)**: 他のデータの存続により、自らの存続が決める。例えば、退社、退学した人のデータはその人のアカウントデータの存続によって、消すか消さないかを判断する。
- そして、寿命の長さを変更できるかどうかによって、固定型寿命と適応型寿命と分けられる。
- **固定型寿命(fixed)**: あらかじめ決められ自動的に変更しない寿命。例えば、10年間、1週間。
 - **適応型寿命(adaptable)**: 利用パターン等によって、寿命が適応していく。例えば、利用状況によって寿命が長くなったり、短くなったり。あるいは逆

に利用されないと寿命が長くなる。例えば、未読みのメールは消さないほうがよいので、寿命が長くすべきである。

上記の分類を組み合わせ、固定型絶対寿命、適応型相対寿命のように、寿命の種類を記述することもできる。前者の意味は「他のデータと独立して決められた適応しない寿命」であり、後者は「ほかのデータに依存しながらも状況によって適応的に変更も可能な寿命」である。

また、寿命の付け方によって、各々のデータにそれぞれ寿命を付けるか、それとも多くのデータをまとめて寿命を付けるか、という選択もある。一般にまとめて付けは基本として、個別（例外）を指定できるようにしたほうがよい。例えば、すべてのメールは5年有効とし、「重要メール」フォルダのメールだけ、20年有効とする。

2.3. データのライフサイクル

寿命と関連するデータのライフサイクルを考える。あるデータ管理システムにおいて、データのライフサイクルは次のようになる。

- 発生：データが始めてシステムに入る
- 変更：新しいバージョンが発生する
- 移動：サブシステム間に移動する
- 合併：他のデータと合併する。
- 消滅・削除：システムから消える・消す

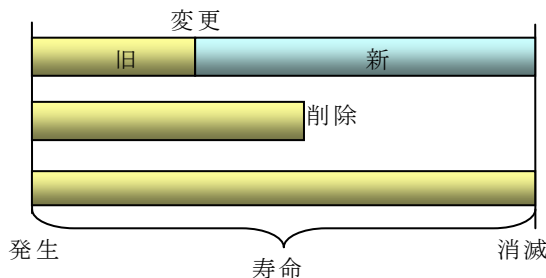


図4. 寿命とデータのライフサイクル

3. データの構造化と半永続データ管理

データの構造化程度によって、半永続データ管理の方法も異なる。表2では、データの構造化程度と永続性種類により、異なるデータ管理技術をまとめた。

3.1. 永続データ管理

3.1.1. 構造化永続データ

構造化永続データは従来のデータベースシステムで扱う対象であり、DBMS (Database Management Systems) を代表とするソフトウェアにより、データ管理を実現している。データモデルはリレーショナルデータモデル (RDB)、オブジェクトリレーショナルモデ

ル (OORDB) がしられている。SQL は標準的問合せ言語である。

3.1.2. 半構造化永続データ

半構造化データでは、構造自体は持っているものの、RDB モデルのようにあらかじめ定めたレイアウトに拘束されない可変的なデータのことをいう。XML で記述した半構造化データに関する研究は最も代表的である半構造化永続データを管理するためのシステムは DBMS のような既存のデータベースシステムを利用するのは一般的である。問合せ言語に、XQL、XQuery 等が知られている。

表2. データの種類とデータ管理技術

		構造化程度		
		構造化	半構造化	非構造化
永続性	永続	DBMS	XML DB	IR
	ストリーム	DSMS	XML-DSMS	時系列
	半永続	SPDMS	XML-SPDMS	SPDFS

3.1.3. 非構造化永続データ

データにはっきりした構造のない文書型情報に対して、情報検索技術 IR (Information Retrieval) を利用するのは一般的である。中でも、ハイパーテキストのリンク関係や、文の構造を利用した検索精度を高める技術も知られているが、構造化データの意味上の「構造」は持っていないため、より高度な問合せが

3.2. ストリームデータ管理

3.2.1. データ・ストリーム

ストリームデータはタイムスタンプ付きのタプルの集合である。個々のデータは構造をもっている。複数のストリームの間や、ストリームと RDB の表の間をジョインして問合せを行うこともある。ただし、データにタイムスタンプが付いているため、問合せを実行するときに、Sliding Window で参加するタプルを最近の部分だけに限定する。データ・ストリーム管理システム (Data Stream Management Systems) というようなソフトを中心に行う。

3.2.2. 半構造化データ・ストリーム

半構造化文書の時系列或は更新し続ける XML 文書に対して、連続問合せやオンライン予測[1,4]について、Sliding Window における XQuery の実現を目指している。

3.3. 半永続データ管理

半永続データも、構造化、半構造化、非構造化データに分けられる。それぞれのデータの特徴に応じたデータ管理仕組みを開発する必要がある。

3.3.1. 構造化半永続データ管理システム

構造化データに対して、それぞれのデータモデルに寿命を組み込む拡張をする必要がある。

リレーショナルデータモデルでデータを構造化する場合、データの寿命付け方は次のように考えている。

- **スキーマレベルの寿命**
スキーマレベルの寿命とは、リレーションスキーマの寿命のことである。例えば、一時表は寿命 1 時間とする。
- **タプルレベルの寿命**
タプルレベルの寿命とは、ある条件を満たすタプルのみ適用する寿命である。例えば、学生の学習出席状況は卒業から 5 年間まで保存する。そのような条件を満たすタプルに 5 年間の寿命を付けることになる。

構造化半永続データ管理システム SPDM (Semi-Persistent Data Management Systems) において、普段の DBMS 機能のほか、寿命に巡る以下のような新しい機能を実現する必要がある。

- 寿命の定義
- 寿命の変更
- 寿命に関する問合せ
- 適応型寿命の維持
- 寿命に関わるデータ整合性

3.3.2. 半構造半永続データ

半永続 XML データを代表とする半構造半永続データを中心に SPDM を構築する。XML データの寿命には DTD レベルの寿命と XML 文書レベルの寿命がある。

DTD レベルの寿命は DTD に関連するすべての XML 文書の寿命とする。XML 文書レベルの寿命は、個々の XML 文書、或は文書集合の寿命を指定する。ある特定の文書に対して、両方の寿命に一致しないときには、短い方が優先とする。

3.3.3. 非構造半永続データ

非構造データはファイルとして扱うので、ファイルシステムレベルの半永続化を求められる。半永続データのためのファイルシステムは SPDFS (File Systems for Semi-Persistent Data) とする。キャッシュファイルシステム[5]にファイルの利用パターンや有効期限を加えて、SPDFS の基本的機能に備えることになる。

4. 半永続データの自己組織化管理

適応型寿命を維持するために、データの利用パターン(アクセス頻度、最近のアクセス時間)に関する情報を管理する必要がある。これにより、利用状況を考慮

したデータの自己組織化管理が可能になる。さらに利用パターンを反映した情報提供システムとして、実用化も期待できる。

4.1. 利用パターンの維持

チャレンジ的な問題は利用パターンの情報を孤立的に維持、提供するためのアルゴリズムの開発である。これを実現するために、Bloom Filters のような非常にアクセスの時間上でも、維持に必要なメモリ空間上でも、スケーラブルで、有力の候補である。[10]では我々は図 2 のようなデータ構造を提案した。複数のハッシュ関数で、データを一定の数のカウンターに射影する。当たったカウンターだけ増やす。カウンターの値は増える一方ではなく、ある一定の時間 T が経過すると、カウンターの値は Decaying 関数に従い、一定の量を減らす。

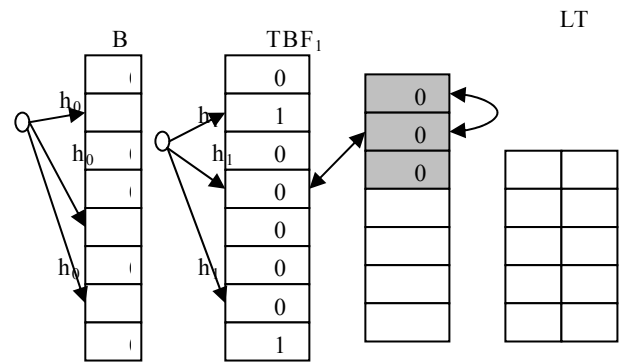


図 2. TBF (Time-Decaying Bloom Filters)

4.2. 半永続データの問合せ

半永続データへの問合せは一般の構造化データや XML データへと同じ問合せをサポートすると共に、下記のような利用パターンを反映した問合せもできる。

Q1: Data Stream に関する研究論文のうち、よく読まれるベスト 10 の一覧

```
SELECT Top 10 *
FROM ResearchPapers
WHERE title LIKE '%Data Stream%'
ORDER BY #References DESC
```

Q2: 著者は Michael Franklin であるレポートのうち、一番最近よばれた 5 つの一覧

```
SELECT Top 5 *
FROM Report
WHERE author='Michael Franklin'
ORDER BY #LastReferredTime ASC
```

5. ストレージ管理

半永続データ管理システムが相応しい物理的データ管理サブシステムはデータのライフサイクル管理ができるシステムである。M. Stonebraker ら[3]は、永続データのための多階層オブジェクトストアシステムを

提案している。その特徴は

1. ストレージ・デバイスの階層化
2. ポリシー設定による階層間マイグレーション
3. 論理データによるポリシー設定

例えば、職員のデータに対して、次のようなポリシーを適用すると、パフォーマンスとストレージ・デバイスの利用に適切である。

main memory representation:

EMP where age \geq 30 and age $<$ 60

disk representation: age $<$ 30

archive representation: age \geq 60

つまり、メイン・メモリに置くべきデータは最もよく使われる 30 才~60 才の職員データである。60 才以上は退職するので、アカイブに入れてもいい。

このようなポリシーに寿命付きデータの処理を加えることができる。

main memory representation:

EMP where age \geq 30 and age $<$ 60

disk representation: age $<$ 30

archive representation: age \geq 60 and age $<$ 100

drop away: age \geq 100

つまり、退職から 40 年を経つと、そのデータを完全に廃棄する。

6. 討 論

半永続化データはアクセスパターンやデータの寿命を考慮したディスク・キャッシュの一般化であり、ハードウェア技術の更なる進展に伴い、ディスク容量の制限が大幅に緩和されると想定しながら提案したものである。従来のメモリキャッシュでは、データの量が比較的少なく、データがメモリに滞在する時間もわずかで、データベースのような永続的データとして扱うことができなかつた。キャッシュに対する評価もヒット率とバイトヒット率などに限られている。

我々はウェブキャッシュに関する研究の中で、キャッシュデータの利用状況を分析した結果、ほとんどのキャッシュデータ(60%以上)がキャッシュされた期間にあまり利用されていない(20%前後のヒットしか得られていない)ことがわかった。この事実にあわせて、ヒット率と別にキャッシュコンテンツの利用率の観点から、キャッシュの仕組みを考え直す必要があると考え始めた。利用率を向上するためには、キャッシュコンテンツを直接利用者に提供できるようにモデル化する必要がある [8]。

このように一度アクセスしたらすぐ消えるデータに対して、キャッシュコンテンツは一定の期間データをディスクに保存するので、永続データに対して、半永続データとして扱うべきである。ディスク容量が十

分大きくなっていくにつれ、半永続データの滞在時間(寿命)が十分長くなり、データの明示的な利用も可能になる。

7. 終わりに

半永続データには寿命や、利用状況(利用頻度など)に時間的属性がもたれているため、モデル化には時間のモデリングが重要である。ストリームデータの寿命を 0 とし、永続データの寿命を ∞ とすれば、半永続データは寿命(0, ∞)となる。このように、ストリームデータ、半永続データ、永続データを統一した見方で扱うことによって、関連分野の研究への貢献も期待できる。

半永続データの研究は構造化データ、半構造化データにおいて、展開できるが、非構造化データとして、ファイルシステムレベルの研究課題が残っている。今後の研究として、オブジェクトリレーショナルデータと半構造化 XML データに絞って研究を深めて活きたい。とりわけ、以下の問題を優先的に解明していきたい。

- 寿命のモデル化と寿命管理機能
- SPDMS システムにおける時間管理機能
- 半永続化データのインデックス
- 半永続データの問合せ処理

文 献

- [1] Sujoe Bose, Leonidas Fegaras, Data stream management for historical XML data. SIGMOD Conference 2004, pp. 239-250
- [2] Lukasz Golab and M. Tamer Ozsu. Issues in Data Stream Management. In SIGMOD Record, vol. 32, no. 2, pp. 5-14, June 2003
- [3] Michael Stonebraker: Managing Persistent Objects in a Multi-Level Store. SIGMOD Conference 1991 pp.2-11
- [4] 河野正太郎 有村博紀 有川節夫, 半構造化データ系列のオンライン予測と XML データ圧縮への応用, DEWS 2003
- [5] Jonathan Ledlie and Matthew McCormick. A Fast File System for Caching Web Objects, University of Wisconsin - Madison, Research Report, May 2000
- [6] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom. Models and issues in data stream systems. PODS 2002, pp. 1-12, June 2002
- [7] Y. Kambayashi and K. Cheng. Capacity bound-free web warehouse. CIDR 2003, pp. 47-57, 2003..
- [8] E. A. M. Shriver, E. Gabber, L. Huang, and C. A. Stein. Storage management for Web proxies. In USENIX Annual Technical Conference, General Track, pp.203-216, 2001.
- [9] M. Rabinovich and O. Spatscheck. Web Caching and Replication. Addison-Wesley, 2002.
- [10] K. Cheng, et al. Efficient Web Profiling by Time-Decaying Bloom Filters, DBSJ Letters vol.4 no. 1.