

情報の抽象化に基づいた広く分散した動的な情報の共有手法

八 木 哲[†]

本稿では、広く分散した多数の情報源において頻繁に変化する情報を観測し、その最新の情報を広く分散した多数の利用者に提供するための、情報共有手法を提案する。課題は次のとおり。(a) 広く分散した多数の情報源に起因する多量の更新処理。(b) 広く分散した多数の利用者に起因する多量の参照処理。(c) 頻繁に変化する情報に起因する高い頻度の更新処理。(d) 最新情報の共有。これらの課題に対処するために、提案手法は次の特徴を持つ。「利用者は情報の詳細を段階的に参照する」という参照パターンを前提条件として、(1) 自律型モジュールを分散配置し、配置した地域で観測された情報を管理させる。(2) 各自律型モジュールは、観測された情報の要約を動的に生成し、これを相互に複製して、各々が保有する観測された情報への索引として用いる。これらにより、参照処理と更新処理を各自律型モジュールに分散・局所化する。複数の自律型モジュールにまたがる更新処理の頻度を削減する。この結果、規模の拡大に対しても自律型モジュールを追加することで対処できる。本稿では、提案手法の枠組と実現法を示す。また例題に照らして提案手法を評価する。

A Method of Sharing Dynamic and Globally Distributed Information Based on Information Abstraction

SATORU YAGI[†]

This paper proposes a method of sharing dynamic information observed on globally distributed information sources among globally distributed users in real time. There are four issues concerning this method: (a) the volume of operations for updating by a large number of globally distributed information sources; (b) the volume of operations for retrieval by a large number of globally distributed users; (c) the frequency of operations for updating by dynamic information and (d) information sharing in real time. To cope with these issues, this method has two features for stepwise retrieval of detailed information by users: (1) distributed autonomous elements manage information observed in their regions and (2) they abstract observed information dynamically and replicate it for each other for use as an index. These features distribute operations for updating and retrieval and decrease the frequency of operations for updating. As a result, it is possible to cope with an increase in operations by increasing the number of autonomous elements. This paper describes the concept and implementation of our method and discusses an evaluation of it.

1. はじめに

広く分散した多数の情報源において頻繁に変化する情報を観測し、その最新の情報を広く分散した多数の利用者に提供できる仕組みがあれば、より利便性の高い広域情報案内サービスを構築できる。たとえば、道路交通情報¹⁾・地域情報・気象環境情報^{2)~5)}を提供できる仕組みがあれば、デジタルシティ⁶⁾・カーテレマティクス・歩行者ITS⁷⁾を融合させたような、計画の立案からドア・ツー・ドアの行動までを支援するサービスを構築できる。ネットワーク情報⁸⁾・計算機資源

情報を提供できる仕組みがあれば、広域ネットワーク監視サービス⁹⁾やグローバルコンピューティング向けの計算機資源管理サービス¹⁰⁾を構築できる。このような仕組みを情報共有基盤と呼ぶことにする。情報共有基盤の実現には、次の課題に対処できる情報共有手法が求められる。

- (a) 広く分散した多数の情報源に起因する多量の更新処理。
- (b) 広く分散した多数の利用者に起因する多量の参照処理。
- (c) 頻繁に変化する情報に起因する高い頻度の更新処理。
- (d) 最新情報の共有。

従来から、情報共有基盤として DNS (Domain

[†] NTT 未来ねっと研究所
NTT Network Innovation Laboratories

Name System)¹¹⁾・ディレクトリ^{12),13)}がある。DNSのラウンドロビン機能・広域負荷分散装置¹⁴⁾・ディレクトリ・ポータルサイトを用いて、サーバ群を束ねたシステム^{15),16)}がある。これらに用いられている情報共有手法を課題(a),(b),(c),(d)に照らせば次のようである。課題(a)のために、自律型モジュールを分散配置し、配置した地域で観測された情報を管理させる。これにより、観測された情報に対する更新処理を分散・局所化する。課題(b)のために、各自律型モジュールは、観測された情報を相互に複製あるいはキャッシュする。これにより、観測された情報に対する参照処理を分散・局所化する。このときの課題(c)は「複製やキャッシュに対する高い頻度の更新処理」である。そこで、課題(c)のために、複製やキャッシュに対する更新処理の頻度を制限する。しかし、複製やキャッシュが最新情報を保持し難くなるために、課題(d)には対処し難い。この問題に関して、グローバルコンピューティング向けの計算機資源管理用ディレクトリ¹⁰⁾では、情報に時刻印や有効期限を添付することで、無効な情報を利用者が棄却できるようにしている。しかし課題(d)に対処できるわけではない。

一方、我々は、広域マルチデータベースシステムMRC¹⁷⁾のリソース管理系¹⁸⁾を構築するために、WISE(Wide-area Information Sharing Engine)^{9),20)}と呼ぶ情報共有基盤の研究を進めている。WISEに用いた情報共有手法は次の特徴を持つ。「利用者は情報の詳細を段階的に参照する」という参照パターンを前提条件として、(1)自律型モジュールを分散配置し、配置した地域で観測された情報を管理させる。(2)各自律型モジュールは、観測された情報の要約を動的に生成し、これを相互に複製して、各々が保有する観測された情報への索引として用いる。これらにより、参照処理と更新処理を各自律型モジュールに分散・局所化する。複数の自律型モジュールにまたがる更新処理の頻度を削減する。この結果、規模の拡大に対しても自律型モジュールを追加することで対処できる。

本稿では、課題(a),(b),(c),(d)に対処できる情報共有手法として、WISEに用いた情報共有手法を提案する。まず、情報共有手法の枠組み(2章)と、その実現法としてWISEの設計概要を示す(3章)。次に、広域ネットワーク監視サービスを例題として、従来の情報共有手法を用いた場合と提案する情報共有手法を用いた場合とを比較する(4章)。両手法の方針は、複数の自律型モジュールの使用と、自律型モジュール間の依存関係の削減である。換言すれば、自律型モ

ジュール間の通信頻度の削減である。そこで「自律型モジュール間の通信頻度」を尺度として両手法を比較する。最後に、本稿の内容をまとめ、今後の課題を示す(5章)。

2. 枠組み

前提条件とする「利用者は情報の詳細を段階的に参照する」という参照パターンの詳細を次に示す。

- 各利用者は、第1段階では大多数の情報の概観を参照する。第2段階では興味を持った情報の詳細を参照する。
- 各情報は、第1段階では大多数の利用者から概観を参照される。第2段階では興味をいだいた利用者から詳細を参照される。

これに基づいて情報共有手法の枠組みを示す。参照パターンの第1段階では、各情報の概観が参照の対象である。利用者が網羅的に参照するために、各情報の参照頻度は比較的高い。そこで、各情報源において観測された頻繁に変化する情報から次の特徴を持つ情報を動的に生成し、これを利用者へ提供する。

- (1) 観測した情報の特徴だけを反映する。
- (2) 観測した情報と比べて時間的な変化が少ない。

この情報を「抽象化した情報」と呼ぶことにする。「抽象化した情報」の複製を利用者のいる地域に配置することにより、「抽象化した情報」に対する参照処理を分散・局所化する。「抽象化した情報」の前記(1),(2)の特徴により、「抽象化した情報」の複製に対する更新処理の頻度を削減する。参照パターンの第2段階では、各情報の詳細が参照の対象である。利用者が選択的に参照するために、各情報の参照頻度は比較的低い。そこで、各情報源において観測された頻繁に変化する情報を、そのまま利用者へ提供する。この情報を「素な情報」と呼ぶことにする。情報源のある地域ごとに「素な情報」を個別管理することにより、「素な情報」に対する更新処理を分散・局所化する。「抽象化した情報」を索引として、参照する「素な情報」を利用者に選択させることにより、「素な情報」に対する参照処理を分散させる。

このような枠組みを持つ情報共有手法を課題(a),(b),(c),(d)に照らせば次のようである。課題(a)のために、自律型モジュールを分散配置し、配置した地域の「素な情報」を管理させる。これにより、「素な情報」に対する更新処理を分散・局所化する。課題(b)のために、各自律型モジュールは、「素な情報」から「抽象化した情報」を生成し、これを相互に複製して、各々が保有する「素な情報」への索引として用いる。これ

により、‘抽象化した情報’に対する参照処理を分散・局所化する。‘素な情報’に対する参照処理を分散させる。課題(c),(d)のために、前記(1),(2)の特徴を持つ‘抽象化した情報’を複製の対象とする。これにより、複製に対する更新処理の頻度を削減したうえで、複製が最新情報を保持できるようにする。

3. 実現法

3.1 概要

2章で示した情報共有手法を実現するには、‘素な情報’から‘抽象化した情報’を生成し、これを自律型モジュールの間で複製するための、体系的な仕組みが必要である。このような仕組みの実現法として、ここでは WISE の設計概要を示す。

WISE の構成を図 1 に示す。地域ごとにセンサと RM (Resource Manager) がある。センサは、情報源において情報を観測し、観測した情報に時刻印を押して、RM に渡す。RM は、2章で示した自律型モジュールに相当する。利用者は、近傍の RM を介して情報を参照する。RM の構成を図 2 に示す。データベースは、‘素な情報’と‘抽象化した情報’を格納する。‘素な情報’は、センサから受け取った情報である。抽象化機能は、‘素な情報’から‘抽象化した情報’を生成する。生成した‘抽象化した情報’を、情報交換機能を介して他の RM に複製する。このとき、RM が相互に複製する‘抽象化した情報’が持つ情報量と、前提条件とする参照パターンの第 1 段階に必要な情報量とが、一致しない場合がある。前者の方が少ない場合、利用者は近傍の RM が保有する‘抽象化した情報’に加えて他の RM が保有する‘素な情報’を参照する。この結

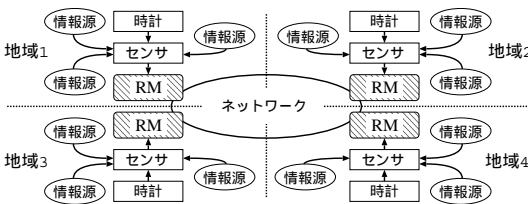


図 1 WISE の構成
Fig. 1 Configuration of WISE.

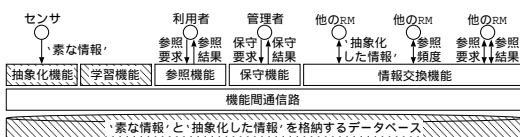


図 2 RM の構成
Fig. 2 Configuration of RM.

果、参照処理を各 RM に分散・局所化できない。前者の方が多い場合、RM は参照されない‘抽象化した情報’を他の RM に複製する。この結果、複製に対する更新処理の頻度を削減できない。この問題に対処するために、学習機能は、他の RM に複製すべき‘抽象化した情報’を参照頻度に基づいて選定する。選定に必要な参照頻度を、情報交換機能を介して他の RM と交換する。参照機能は、利用者からの参照要求に応じて‘素な情報’と‘抽象化した情報’を提供する。他の RM が保有する‘素な情報’や‘抽象化した情報’が参照要求の対象である場合には、情報交換機能を介して参照要求を転送する。保守機能は、管理者からの保守要求に応じて‘素な情報’と‘抽象化した情報’を操作する。次節以降では、RM の中核であるデータベース・抽象化機能・学習機能の詳細を示す。

3.2 ‘素な情報’と‘抽象化した情報’を格納するデータベース

まず、‘素な情報’と‘抽象化した情報’を、次に示す単位データ d の集合として表す。

- $d = (g, t, a, s, v)$
 - g : 地域。 d が生成された地域を示す。識別名¹²⁾を用いる。
 - t : 時刻。 d の生成時刻を示す。単位時間ごとに単調増加するシーケンス番号を用いる。これにより、ある時点に生成された一連の単位データの指定を容易にする。単位データが時系列情報である場合に有効である。
 - a : 抽象度。値が大きいくほど抽象化されており、よりいっそう、観測した情報の特徴だけを反映し、観測した情報と比べて時間的な変化が少ないことを示す。‘素な情報’は 0。‘抽象化した情報’は正の整数。同じ‘素な情報’から生成された‘抽象化した情報’に対しては、更新頻度が低いほど大きい抽象度を与える。異なる‘素な情報’から生成された‘抽象化した情報’に対しては、‘一瞥する’ための情報・‘概要を知る’ための情報・‘詳細を知る’ための情報など、同じ目的のための情報であれば同じ抽象度を与える。
 - s : 情報源。 d を生成した情報源を示す。 g に対する相対識別名¹²⁾を用いる。
 - v : 値。

次に、前提条件とする参照パターンに則して利用者が単位データを参照するように、次に示す WISE 情報モデル (IMW: Information Model for WISE) と呼ぶグラフ構造を d の集合に与える。

- $IMW = (N, AA, AG)$
 - N はノードの集合 . $N = \{n_f | n_f = \{d_f\} d_f$ は、地域が g_f 、抽象度が a_f である単位データ $\}$.
 - AA は、ノードの抽象度の高低関係を表すアークの集合 . $AA = \{ (n_{ah}, n_{al}) | n_{ah}$ は、抽象度 ah の単位データを要素に持つノード . n_{al} は、抽象度 al の単位データを要素に持つノード . $ah > al$. $\}$.
 - AG は、ノードの地域の隣接関係を表すアークの集合 . $AG = \{ (n_{a1}, n_{a2}) | n_{a1}$ は、抽象度 $a1$ の単位データを要素に持つノード . n_{a2} は、抽象度 $a2$ の単位データを要素に持つノード . $a1 = a2$. $\}$.

グラフ構造を与えられた d の集合を、IMW のインスタンスと呼ぶことにする . 図 3 に例示する . この例では、抽象度 0・抽象度 1・抽象度 2 の単位データを要素に持つノードがある . 抽象度の大きい単位データを要素に持つノードほど、大まかな地域情報を持つ . 抽象化機能が、抽象度 0 の単位データから抽象度 1 以上の単位データを生成する (3.3 節参照) . 利用者は、次のように単位データを参照する .

- (1) 全地域を ‘一瞥する’ . すなわち、ノードの地域の隣接関係を示すアークをたどりながら、抽象度 2 の単位データを参照する .
- (2) 興味のある地域を選択する . すなわち、抽象度 2 の単位データを要素に持つノードのなかから、興味のある地域のノードを選択する .
- (3) 選択した地域の ‘概要を知る’ . すなわち、選択したノードから、ノードの抽象度の高低関係を示すアークをたどり、抽象度 1 の単位データを参照する .
- (4) 選択した地域の ‘詳細を知る’ . すなわち、さらにノードの抽象度の高低関係を示すアークをたどり、抽象度 0 の単位データを参照する .

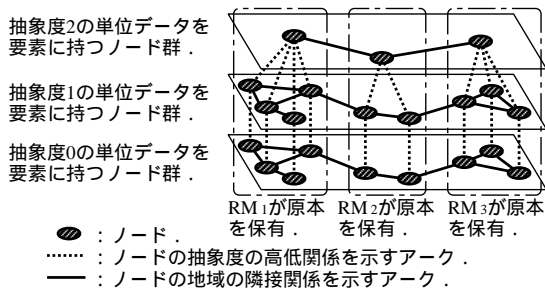


図 3 IMW のインスタンス
 Fig. 3 An instance of IMW.

この例では、 $RM_1 \cdot RM_2 \cdot RM_3$ が単位データを共有している . 各 RM は、IMW のインスタンスのトポロジ情報を保有する . さらに、配置された地域のセンサから受け取った抽象度 0 の単位データと、その抽象度 0 の単位データから生成した抽象度 1 以上の単位データを、単位データの原本として保有する . 他の RM が保有する単位データの原本に関しては、その RM への参照リンクと部分的に複製を保有する . たとえば RM_1 は、図 3 左の 9 つのノードの要素である単位データの原本を保有する . 図 3 中央の 5 つのノードの要素である単位データに関しては、 RM_2 への参照リンクと部分的に複製を保有する . 図 3 右の 7 つのノードの要素である単位データに関しては、 RM_3 への参照リンクと部分的に複製を保有する . $RM_2 \cdot RM_3$ の学習機能が、 $RM_2 \cdot RM_3$ が保有する単位データの原本のなかから、 RM_1 へ複製すべき単位データを選定する (3.4 節参照) .

3.3 ‘抽象化した情報’を生成する抽象化機能

抽象化機能の動作を説明する . 基本となる動作は、ノード n_{al} の要素である抽象度 al の単位データから、ノード n_{ah} の要素である抽象度 $ah (> al)$ の単位データを生成する動作である . この動作は次の特徴を持つ .

- (1) n_{al} の要素である単位データの値が表す情報の特徴だけを、 n_{ah} の要素である単位データの値が表す情報に反映させる .
- (2) n_{al} の要素である単位データの値の変化が、 n_{ah} の要素である単位データの値に現れ難いようにする .

以上の動作を抽象化動作と呼ぶことにする . 具体的には次のような動作である .

- 精度削減型の動作 : 単位データの値の有効桁数を削減し、その結果を値として単位データを生成する . あるいは、単位データの値をあらかじめ定めた種別に分類し、その種別名を値として単位データを生成する .
- 要素選択型の動作 : 注目度の高い情報・一次判断に必要な情報・情報の概説・情報の定義情報など、特徴的な情報を表す単位データを選択することにより、単位データを生成する .
- 要素集約型の動作 : 単位データの値の集合に対して統計関数・恣意的な関数を適用し、その結果を値として単位データを生成する .

抽象化機能は、IMW のインスタンスに対して抽象化動作を行う . すなわち、抽象度 0 の単位データの更新を契機として、更新された抽象度 0 の単位データから、ノードの抽象度の高低関係を示すアークにより

表 1 抽象化ルール
Table 1 The abstraction rule.

抽象化ルール	= * (識別子 事象 条件 抽象化動作 複製動作 連鎖動作)
事象	= 地域 抽象度 情報源
条件	= 閾値
抽象化動作	= 地域 時刻 抽象度 情報源 値生成
複製動作	= * (RM (“複製する” / “複製しない”))
連鎖動作	= 識別子

対応付けられた、抽象度 1 の単位データを生成する。生成した抽象度 1 の単位データから、ノードの抽象度の高低関係を示すアークにより対応付けられた、抽象度 2 の単位データを生成する。これを繰り返し、連鎖反動的に単位データを生成する。また、生成した単位データを他の RM に複製する。このような IMW のインスタンスに対する抽象化動作は、ECA ルールを用いたアクティブデータベース²¹⁾の動作として解釈できる。そこで、IMW のインスタンスに対する抽象化動作を、抽象化ルールと呼ぶ ECA 型のルールとして規定する。抽象化ルールを拡張 BNF²²⁾を用いて表 1 に示し、各項目を説明する。各抽象化ルールは、‘事象’の発生回数を表すカウンタ ec を持つ。初期値は 0。

- 抽象化ルール：‘識別子’は抽象化ルールの識別子。‘事象’が発生したときに‘条件’を満たせば‘抽象化動作’・‘複製動作’・‘連鎖動作’を行う。
- 事象：‘地域’・‘抽象度’・‘情報源’を要素に持つ単位データが追加されたときに‘事象’が発生したと判断する。同時に ec を 1 増やす。
- 条件： ec が‘閾値’を超えたときに‘条件’を満たしたと判断する。同時に ec を 0 にする。
- 抽象化動作：‘値生成’を実行し、その実行結果を‘値’とする。‘地域’・‘時刻’・‘抽象度’・‘情報源’・‘値’を要素に持つ単位データを生成する。
- 複製動作：‘抽象化動作’により今回生成した単位データの‘値’が前回生成した単位データの‘値’と異なる場合に、今回生成した単位データを‘複製する’と組を作る‘RM’に複製する。一方、‘複製する’と組を作る‘RM’は、所定の時間間隔で複製が更新されない場合には、単位データの‘値’に変化はないと判断し、自律的に複製を更新する。
- 連鎖動作：‘識別子’が示す抽象化ルールの ec を 1 増やす。このとき、その抽象化ルールの‘条件’を満たせば、その抽象化ルールの‘抽象化動作’・‘複製動作’・‘連鎖動作’を行う。連鎖反動的に単位データを生成するための動作である。

3.4 複製すべき‘抽象化した情報’を選定する学習機能

学習機能の動作を説明する。学習機能は「自律型モジュール間の通信頻度」を尺度として、RM が保有する単位データの原本のなかから、他の RM へ複製すべき単位データを選定する。たとえば RM_1 の学習機能は、 RM_1 から RM_2 に単位データ d を複製するか否かを次のように判断する。

- r は、 RM_2 の近傍の利用者による、 RM_1 が保有する d の原本と RM_2 が保有する d の複製に対する単位時間あたりの参照回数。 RM_2 からの定期報告を利用して RM_1 が計数する。
- u は、 RM_1 による、 d に対する単位時間あたりの更新回数。 RM_1 が計数する。
- r と u を用いて次のように判断する。厳密には通信プロトコルに依存した重み付けが r と u に必要であるが、ここでは単純化のために省略した。
 - (1) $r > u$ の場合。 d は、複数回参照される間に 1 回更新される。同じ d が繰り返し転送されることを防ぐために、 RM_1 から RM_2 に d を複製する。 RM_2 の近傍の利用者は、 RM_2 が保有する d の複製を参照する。
 - (2) $r < u$ の場合。 d は、1 回参照される間に複数回更新される。複製しても参照されない d の転送を防ぐために、 RM_1 から RM_2 に d を複製しない。 RM_2 の近傍の利用者は、 RM_1 が保有する d の原本を、 RM_2 を介して参照する。

これにより、冗長・無意な転送がないという意味において「自律型モジュール間の通信頻度」を最低限に削減する。複製するか否かの判断は、 r と u を計数する単位時間が短いほど正確に行える。その反面、 r にパース的な変化がある場合には、参照先となる RM が頻繁に代わるために、参照に支障をきたす。そこで、安定した参照が行えるように、 r と u の大小関係が指定回数継続するまで判断を保留する。以上の動作を学習動作と呼ぶことにする。このような学習動作は、ECA ルールを用いたアクティブデータベースの動作として解釈できる。そこで、学習動作を学習ルールと呼ぶ ECA 型のルールとして規定する。拡張 BNF を用いて学習ルールを表 2 に示し、次のパラメータを用いて各項目を説明する。

- d は、ある抽象化ルールが生成した単位データ。
- rc_i ($i = 0, 1, 2, \dots, n$) は、 RM_i ($i = 0, 1, 2, \dots, n$) の近傍の利用者による、 d の参照回数を表すカウンタ。初期値は 0。

表 2 学習ルール
Table 2 The learning rule.

学習ルール	= *	(識別子 事象 複製開始条件 複製開始動作 複製中止条件 複製中止動作)
事象	=	周期
複製開始条件	=	閾値
複製開始動作	=	“複製する”
複製中止条件	=	閾値
複製中止動作	=	“複製しない”

- uc は、 d の更新回数を表すカウンタ。初期値は 0。各学習ルールは、 rc_i ($i = 0, 1, 2, \dots, n$) と uc の大小関係の継続回数を表すカウンタとして、 hc_i ($i = 0, 1, 2, \dots, n$) と lc_i ($i = 0, 1, 2, \dots, n$) を持つ。初期値は 0。
- 学習ルール: ‘識別子’ は学習ルールの識別子。‘事象’ が発生したときに、‘複製開始条件’ を満たせば ‘複製開始動作’ を行い、‘複製中止条件’ を満たせば ‘複製中止動作’ を行う。
- 事象: ‘周期’ ごとに ‘事象’ が発生したと判断する。同時に次の動作を行う。
 - $rc_i > uc$ の場合、 hc_i を 1 増やし、 lc_i を 0 にする。 rc_i と uc を 0 にする。
 - $rc_i < uc$ の場合、 hc_i を 0 にし、 lc_i を 1 増やす。 rc_i と uc を 0 にする。
- 複製開始条件: hc_i が ‘閾値’ を超えたときに ‘複製開始条件’ を満たしたと判断する。同時に hc_i と lc_i を 0 にする。
- 複製開始動作: d を生成した抽象化ルールの ‘複製動作’ について、 RM_i と組を作る値を ‘複製する’ にする。 d を RM_i に複製する。
- 複製中止条件: lc_i が ‘閾値’ を超えたときに ‘複製中止条件’ を満たしたと判断する。同時に hc_i と lc_i を 0 にする。
- 複製中止動作: d を生成した抽象化ルールの ‘複製動作’ について、 RM_i と組を作る値を ‘複製しない’ にする。 d の複製の削除命令を RM_i に送る。

4. 評価

4.1 例題

課題 (a), (b), (c), (d) を含む分かりやすい例題として、次のような広域ネットワーク監視サービス^{9),23)}を取り上げる。まず、監視対象となるネットワークを図 4 に示す。単純化のために Group の数は 2 である。Client1・Client2 は、Server1・Server2 が提供するサービスを両方とも利用する。このとき、広域ネットワーク監視サービスは、次に示す負荷情報を観測し、これ

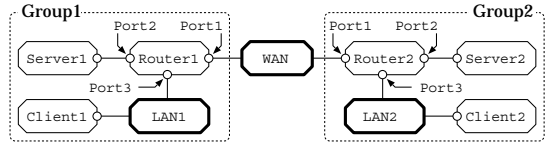


図 4 監視対象となるネットワーク
Fig. 4 A target network for observation.

を Group1・Group2 に所属する利用者に提供する。

• 負荷情報

- (1) Router1・Router2 の Port1・Port2・Port3 のトラフィック (bit/sec) .
- (2) Router1・Router2 の Port1・Port2・Port3 のトラフィック (packet/sec) .
- (3) Server1・Server2 の CPU 使用率 (%) .

利用者は、広域ネットワーク監視サービスから提供される負荷情報を次のように参照する。

- ‘サービスのユーザビリティを示す情報’ として、つねに負荷情報 (1), (2), (3) を参照する。このときに必要な負荷情報の分解能は、一般的なネットワーク観測ツールである MRTG²³⁾ の標準的な観測間隔に準じて、5 分とする。
- ‘性能管理や障害管理のための情報’ として、‘サービスのユーザビリティを示す情報’ が高い負荷を示したときに、負荷情報 (1), (2), (3) の対応する項目を参照する。各項目の参照時間は運用時間の 10% とする。このときに必要な負荷情報の分解能は、DoS (Denial of Service) 攻撃などの不正アクセスの検出・追跡²⁴⁾ が可能なように、1 分とする。

次節以降では、このような広域ネットワーク監視サービスを構築するために従来の情報共有手法を用いた場合と提案する情報共有手法を用いた場合とを「自律型モジュール間の通信頻度」を尺度として比較する。

4.2 従来の情報共有手法を用いた場合

一般的なネットワーク監視システムでは、SNMP エージェントが管理する MIB を SNMP マネージャがポーリングすることにより、ネットワークを監視する⁸⁾。SNMP エージェントを情報源、SNMP マネージャを自律型モジュールとして見れば、従来の情報共有手法を用いたネットワーク監視システムとしてとらえることができる。このようなネットワーク監視システムを 4.1 節の例題に適用する場合には、次のようにする。SNMP マネージャを図 4 の Group1・Group2 に配置する。SNMP マネージャは、‘サービスのユーザビリティを示す情報’ と ‘性能管理や障害管理のための情報’ を利用者に提供するために、4.1 節に示し

た負荷情報を SNMP エージェントから収集し保持する．負荷情報の更新間隔は，負荷情報の分解能の指定に従って 1 分である．

4.3 提案する情報共有手法を用いた場合

提案する情報共有手法を用いたネットワーク監視システムを 4.1 節の例題に適用する場合には，次のようにする．RM を図 4 の Group1・Group2 に配置する．それぞれ RM₁・RM₂ と呼ぶことにする．RM₁・RM₂ は，‘性能管理や障害管理のための情報’を利用者に提供するために，配置された Group に属する Router・Server の負荷情報を SNMP エージェントから収集し，次に示す‘素な情報’として保持する．‘素な情報’の更新間隔は，負荷情報の分解能の指定に従って 1 分である．

● ‘素な情報’

- (1) 配置された Group に属する Router の Port1・Port2・Port3 のトラヒック (bit/sec) ．
- (2) 配置された Group に属する Router の Port1・Port2・Port3 のトラヒック (packet/sec) ．
- (3) 配置された Group に属する Server の CPU 使用率 (%) ．

また，‘サービスのユーザビリティを示す情報’を利用者に提供するために，‘素な情報’ (1)，(2)，(3) から次に示す‘抽象化した情報’ (1)，(2)，(3) を生成する．‘抽象化した情報’の更新間隔は，負荷情報の分解能の指定に従って 5 分である．

● ‘抽象化した情報’

- (1) 配置された Group に属する Router の WAN 接続ポート (Port1) のトラヒック (kbit/sec) ．
- (2) 配置された Group に属する Router のスイッチング負荷 (100 packet/sec) ．
- (3) 配置された Group に属する Server の CPU 使用率 (4 段階表示) ．

‘素な情報’ (1)，(2)，(3) の抽象度を 0，‘抽象化した情報’ (1)，(2)，(3) の抽象度を 1 として，RM₁・RM₂ が保有する‘素な情報’ (1)，(2)，(3) と‘抽象化した情報’ (1)，(2)，(3) から作成した IMW のインスタンスを図 5 に示す．図 5 に基づいて，‘素な情報’ (1)，(2)，(3) から‘抽象化した情報’ (1)，(2)，(3) を生成するために RM₁ で行われる抽象化動作 (1)，(2)，(3) の概要を次に示す．

● 抽象化動作

- (1) ‘Router1 の Port1 のトラヒック (bit/sec)’

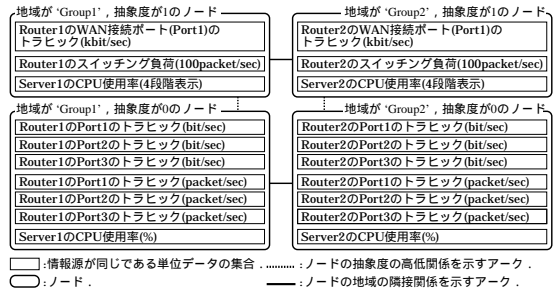


図 5 例題における IMW のインスタンス
Fig. 5 An instance of IMW for the exercise.

が 5 回更新されるごとに，5 回分の単位データの値の平均値を求め，その単位を kbit/sec にすることにより，‘Router1 の WAN 接続ポート (Port1) のトラヒック (kbit/sec)’ を生成する．

- (2) ‘Router1 の Port1 のトラヒック (packet/sec)’・‘Router1 の Port2 のトラヒック (packet/sec)’・‘Router1 の Port3 のトラヒック (packet/sec)’ が 5 回更新されるごとに，5 回分の単位データの値の平均値を求め，その単位を 100 packet/sec にすることにより，‘Router1 のスイッチング負荷 (100 packet/sec)’ を生成する．
- (3) ‘Server1 の CPU 使用率 (%)’ が 5 回更新されるごとに，5 回分の単位データの値の平均値を求め，求めた平均値を 4 段階で表すことにより，‘Server の CPU 使用率 (4 段階表示)’ を生成する．

抽象化動作 (1) に対応する抽象化ルールを，表 1 の各項の設定値として表 3 に示す．値生成の項は，地域・時刻・抽象度・情報源・値を属性に持つリレーションとして単位データが関係データベースに格納されていると仮定し，SQL を用いて記述した．抽象化動作が単位データの集合に対する動作であるために，SQL を用いた値生成の項の記述は比較的容易である．抽象化動作が精度削減型の動作の場合には算術演算子，要素選択型の動作の場合には SELECT 句，要素集約型の動作の場合には統計関数やユーザ定義関数を用いて記述できる．この抽象化ルールに対する学習動作は，5 分ごとに参照頻度と更新頻度を比較し，同じ大小関係が 2 回継続したときに複製するかどうかを切り替えることにする．この学習動作に対応する学習ルールを，表 2 に示した各項の設定値として表 4 に示す．

4.4 比較

従来の情報共有手法を用いた場合 (4.2 節) の「自

表3 抽象化ルールの記述例
Table 3 An example of abstraction rule.

事象	地域 = 'o=Group1' * 1 抽象度 = 0 情報源 = 'cn=Router1のPort1の トラフィック(bit/sec)' * 1
条件	閾値 = 5
抽象化動作	地域 = 'o=Group1' * 1 時刻 = \$TIME * 2 抽象度 = 1 情報源 = 'cn=Router1のWAN 接続ポート (Port1)のトラフィック(kbit/sec)' * 1 値生成 = select AVG(値)/1000 from 単位データ where 地域 = 'o=Group1' * 1 and 時刻 > \$TIME * 2 - 5 and 抽象度 = 0 and 情報源 = 'cn=Router1のPort1 のトラフィック(bit/sec)' * 1
複製動作	RM ₂ 複製しない
連鎖動作	識別子 = null

* 1: ' ' で括られた文字列は識別名あるいは相対識別名である。
* 2: \$TIME は現在時刻を示す予約語。実行時に展開される。

表4 学習ルールの記述例
Table 4 An example of learning rule.

事象	周期 = 5(分)
複製開始条件	閾値 = 2
複製開始動作	複製する
複製中止条件	閾値 = 2
複製中止動作	複製しない

律型モジュール間の通信頻度」を基準値 (=1) として、提案する情報共有手法を用いた場合 (4.3 節) の「自律型モジュール間の通信頻度」を求めることで、両手法を比較する。基準値は、従来手法を用いた場合に SNMP マネージャが負荷情報を収集することで生じる通信頻度である。これは、提案手法を用いた場合に RM が「素な情報」を相互に複製することで生じる通信頻度に等しい。すなわち、「素な情報」の更新頻度が 1 である。以降では、「素な情報」の更新頻度を 1 として、提案手法を用いた場合の「自律型モジュール間の通信頻度」を求める。

RM₁ と RM₂ の間の通信頻度 t の平均値 \bar{t} は、抽象度が a ($a = 0, 1$) である単位データに関する通信頻度を t_a ($a = 0, 1$)、その平均値を \bar{t}_a ($a = 0, 1$) とすれば、次のようである。

$$\bullet \bar{t} = \bar{t}_0 + \bar{t}_1$$

t_a ($a = 0, 1$) の変域を求める。学習機能は、単位データの参照頻度と更新頻度が等しいときを境に、単位データを他の RM に複製するか否かを切り替える。参照頻度の方が小さい場合は、複製しないために通

信頻度は参照頻度に等しい。更新頻度の方が小さい場合は、複製するために通信頻度は更新頻度に等しい。通信頻度の最大値は、更新頻度が抽象化ルールによって決まる固定値であるために、更新頻度である。更新頻度の最大値は、抽象度 0 の単位データの更新頻度 (=1) である。したがって、 t_a ($a = 0, 1$) の変域は次のようである。

- $0 \leq t_a = \min(r_a, u_a) \leq u_a \leq 1$ ($a = 0, 1$)
 - r_a : 参照頻度。抽象度 a の単位データの参照頻度。
 - u_a : 更新頻度。実際に参照された抽象度 a の単位データの更新頻度。
- $u_a = 1 \cdot ur_a \cdot rr_a$ ($a = 0, 1$)
 - ur_a : 更新比率。抽象度 0 の単位データの更新頻度に対する、抽象度 a の単位データの更新頻度の比率。 $0 \leq ur_a \leq 1$ 。
 - rr_a : 参照比率。抽象度 a の単位データ数に対する、実際に参照された抽象度 a の単位データ数の比率。 $0 \leq rr_a \leq 1$ 。

ここで、 t_a ($a = 0, 1$) が最悪(大)値をとるように、参照頻度と更新頻度に対して次の仮定を置く。

- 単位データが参照されるときは、つねに更新頻度の方が参照頻度よりも小さく、単位データは RM₁ と RM₂ の間で複製される。

利用者の「サービスのユーザビリティを示す情報」の参照の仕方 (4.1 節参照) と前記の仮定から、抽象度 1 の単位データに関する通信頻度 t_1 の平均値 \bar{t}_1 は次のようである。すべての単位データがつねに複製されるために、 $t_1 = 1 \cdot ur_1 \cdot rr_1$, $rr_1 = 1$ 。抽象度 0 の単位データに対して、単位データ数が 3/7、更新頻度が 1/5 であるために、 $ur_1 = 3/7 \cdot 1/5 \approx 0.09$ 。したがって \bar{t}_1 は次のようである。

$$\bullet \bar{t}_1 \approx 1 \cdot 0.09 \cdot 1 = 0.09$$

利用者の「性能管理や障害管理のための情報」の参照の仕方 (4.1 節参照) と前記の仮定から、抽象度 0 の単位データに関する通信頻度 t_0 の平均値 \bar{t}_0 は次のようである。参照されるときはつねに複製されるために $t_0 = 1 \cdot ur_0 \cdot rr_0$ 。各単位データは 1/10 の確率で参照されるために $rr_0 = 0.1$ 。抽象度 0 の単位データであるために $ur_0 = 1$ 。したがって \bar{t}_0 は次のようである。

$$\bullet \bar{t}_0 = 1 \cdot 1 \cdot 0.1 = 0.1$$

\bar{t}_0 と \bar{t}_1 から、 \bar{t} は次のようである。すなわち、提案手法を用いた場合の「自律型モジュール間の通信頻度」は、従来手法を用いた場合の「自律型モジュール間の通信頻度」の 19% 程度である。

$$\bullet \bar{t} = \bar{t}_0 + \bar{t}_1 \approx 0.1 + 0.09 = 0.19$$

4.5 考 察

前提条件とする「利用者は情報の詳細を段階的に参照する」という参照パターンを利用して「自律型モジュール間の通信頻度」を削減するには、利用者が必要とする詳しさに応じて、自律型モジュールの間で通信する情報の詳しさを調節すればよい。従来の情報共有手法を用いた場合(4.2節)には、SNMP マネージャのポーリング対象とポーリング間隔が固定されているために、前記のような通信はできない。提案する情報共有手法を用いた場合(4.3節)には、RM が抽象度の異なる単位データを生成し参照頻度の高い単位データだけを相互に複製するために、前記のような通信ができる。

また、複数の組織が参加するネットワークを監視対象とする場合には、セキュリティの確保やプライバシーの保護の観点から、パケットの内容などの詳細な情報を共有できないことが多い。提案する情報共有手法では、加工済の情報を共有する次のような方法が利用できる。

- 抽象度の高い単位データだけを共有し、抽象度の低い単位データは各 RM に閉じて使用する。
- 抽象度の低い単位データに対する参照手段として、特定のストアドプロシージャのみを許可する。

これらの方法は「自律型モジュール間の通信頻度」を削減するうえでも有効である。

5. おわりに

本稿では、広く分散した多数の情報源において頻繁に変化する情報を観測し、その最新の情報を広く分散した多数の利用者に提供するための、情報共有手法を提案した。情報共有手法の実現法として、WISE の設計概要を示した。WISE の基本動作は、データ型を拡張した関係データベースと JAVA プログラムからなる簡易な実験システムを用いて確認している。また、例題として広域ネットワーク監視サービスを取り上げ、従来の情報共有手法を用いた場合と提案する情報共有手法を用いた場合とを「自律型モジュール間の通信頻度」を尺度として比較した。

今後の課題として、学習機能の拡張がある。IMW のインスタンスの抽象度の階層が深い場合には、同じ抽象度 0 の単位データから生成された抽象度の異なる複数の単位データが、同一の RM に複製されたり同一の RM を介して参照されたりすることがある。このとき、抽象度の高い単位データは抽象度の低い単位データから生成可能であるという意味において、単位データは冗長に転送されている。この結果、従来の情

報共有手法よりも「自律型モジュール間の通信頻度」を削減できないことがある。この問題に対して、特定の抽象度の単位データだけを複製し、より抽象度の高い単位データを複製先の RM において生成することにより、単位データの冗長な転送を排除できる見込みがある²⁵⁾。また情報交換機能の検討がある。RM の間で単位データを交換するために用いる、即時性・信頼性・転送効率などの要件を満たす通信プロトコルの確立が課題である。

謝辞 有益なご意見をいただいた名古屋工業大学高橋直久教授に感謝いたします。

参 考 文 献

- 1) (財)道路交通情報通信システムセンタ：VICIS の挑戦 (財)道路交通情報通信システムセンタ (1996).
- 2) (財)日本気象協会：tenki.jp. <http://tenki.jp/>
- 3) 国土交通省：川の防災情報。
<http://www.river.go.jp/>
- 4) 国土交通省：地震計ネットワーク情報。
<http://www.nilim.go.jp/japanese/database/nwdb/>
- 5) 環境省：大気汚染物質広域監視システム。
<http://w-soramame.nies.go.jp/>
- 6) 石田 亨：デジタルシティの現状，情報処理学会誌，Vol.41，No.2，pp.163-168 (2000).
- 7) 国土交通省道路局：地域の ITS・歩行者の ITS。
<http://www.its.go.jp/ITS/j-html/index.html>
- 8) Miller, M.A.: *Managing Internetworks with SNMP*, Second Edition, M & T Books (1997). トップスタジオ(訳): SNMP インターネットワーク管理, 翔泳社 (1998).
- 9) 荒野, 西郷: 大規模商用ネットワークの運用事例, 情報処理学会誌, Vol.39, No.10, pp.969-975 (1998).
- 10) Czajkowski, K., Fitzgerald, S., Foster, I. and Kesselman, C.: Grid Information Services for Distributed Resource Sharing, *Proc. 10th IEEE International Symposium on High-Performance Distributed Computing* (2001).
- 11) Mockapetris, P.V. and Dunlap, K.J.: Development of the Domain Name System, *Proc. ACM SIGCOMM'88* (1988).
- 12) 大山, 千田, 戸部, 窪田, 田中, 空: X.500 ディレクトリ入門, 東京電気大学出版局 (1997).
- 13) Wahl, M., Howes, T. and Kille, S.: Lightweight Directory Access Protocol (v3), RFC2251 (1997).
- 14) 下川, 吉田, 牛島: 多様な選択ポリシーを利用可能なサーバ選択機構, 信学会論文誌 D-I, Vol.J84-D-I, No.9, pp.1396-1403 (2001).
- 15) 馬場, 山口: DNS を用いた広域負荷分散の実

- 装, 情報処理学会研究会報告, DSM-9-7, pp.37-42 (1998).
- 16) 服部, 呉, 安田, 横井: ディレクトリサービスを利用した都市情報の分散型データベース構築に関する検討, 情報処理学会論文誌, Vol.41, No.12, pp.3307-3313 (2000).
- 17) 八木, 高橋: オープンな協調型データベースのためのアーキテクチャ, 情報処理学会論文誌: データベース, Vol.42, No.SIG8(TOD10), pp.108-123 (2001).
- 18) 八木, 高橋: メタ・コンピューティングのためのリソース管理フレームワーク, 第 61 回情報処理学会全国大会, 第 3 分冊, pp.519-520 (2000).
- 19) 八木, 高橋: 広域分散した動的な情報のための共有システムの構想, 2001-DPS-103-3, pp.13-18 (2001).
- 20) 八木, 高橋: 広域分散した動的な情報の共有システム WISE のデータベース, 2002-DBS-126-3, pp.17-24 (2002).
- 21) 石川 博: アクティブデータベース, 情報処理学会誌, Vol.35, No.2, pp.120-129 (1994).
- 22) Crocker, D. and Overell, P.: Augmented BNF for Syntax Specifications: ABNF, RFC2234 (1997).
- 23) 林, 中山, 箱崎: 総論: 安定したネットワークの構築, 運用をめざして, 情報処理学会誌, Vol.39, No.10, pp.959-963 (1998).
- 24) 武井, 太田, 加藤, 根元: トラヒックパターンを用いた不正アクセス検出及び追跡方式, 信学会論文誌 B, Vol.J84-B, No.8, pp.1464-1473 (2001).
- 25) 八木 哲: 広く分散した動的な情報のための情報共有手法の最適化, DEWS2003 (2003).
(平成 14 年 12 月 27 日受付)
(平成 15 年 4 月 8 日採録)

(担当編集委員 有川 正俊)



八木 哲 (正会員)

昭和 41 年生。平成 4 年大阪大学大学院工学研究科電子工学専攻博士課程前期修了。同年日本電信電話株式会社入社。ソフトウェア工学, 交換機シミュレータ, ネットワーク・システム, 広域データベース等の研究開発に従事。現在, NTT 未来ねっと研究所所属。