

ホームネットワークにおける住宅ログを活用した 環境コンテキスト取得システムの実装

増田 哲也[†] 渡邊 雄一[†] 杉本 真佑[†] 佐伯 幸郎[†] 中村 匡秀[†]

[†] 神戸大学 〒657-8501 神戸市灘区六甲台町 1-1

E-mail: †{masuda,nabe}@ws.cs.kobe-u.ac.jp, ††{shinsuke,masa-n}@cs.kobe-u.ac.jp,
†††sachio@carp.kobe-u.ac.jp

あらまし 従来多くのコンテキストウェアサービスでは、表現可能なコンテキストが現在の状況に限定されていた。そこで我々は先行研究において、ホームネットワークシステム (HNS) に蓄積された住宅ログを活用することで、過去の状況も考慮したより高度なコンテキストを表現する手法を提案している。この先行研究では、住宅ログを用いたコンテキスト (ログコンテキスト) の定義と活用について提案を行っているが、具体的なシステムの実装にはいたっていない。そこで本研究では、実際の HNS 環境 (CS27-HNS) で蓄積している環境センサの値とそのログを対象を絞り、それらを利用した環境ログコンテキストを取得するシステムを設計・実装する。実装においては、ログコンテキストをできるだけ容易に作成・管理できるように、システムをアプリケーション、ログコンテキスト、ログクエリ、DB 接続の 4 つのレイヤで構成した。また実装したシステムを利用して、CS27-HNS において実用的な環境ログコンテキスト「昨日より 5℃以上寒い」「ここ数年で最も寒い」を定義し、システムを通して実際に取得できることを確認した。
キーワード コンテキスト, コンテキストウェアサービス, 住宅ログ, 環境センサ, クエリ

Implementing Environment Context Reasoning with House Log in Home Network

Tetsuya MASUDA[†], Yuichi WATANABE[†], Shinsuke MATSUMOTO[†], Sachio SAIKI[†], and
Masahide NAKAMURA[†]

[†] Kobe University, 1-1 Rokkodai, Nada, Kobe, Hyogo, 657-8501 Japan

E-mail: †{masuda,nabe}@ws.cs.kobe-u.ac.jp, ††{shinsuke,masa-n}@cs.kobe-u.ac.jp,
†††sachio@carp.kobe-u.ac.jp

Abstract In the conventional context-aware services, every context was defined with present situations only. Our previous work has extended the limitation so that a context can involve past situations, by using house log of the home network system (HNS). Although we presented the definition and application of the context with the house log (called log contexts), a concrete system has not been implemented yet. In this paper, we therefore design and implement a system that can obtain the log context using the environmental sensors and their log data stored in our actual HNS environment (CS27-HNS). In the implementation, we define four layers (application, log context, log query, and DB connection) of the system, so that the log context can be easily created and managed. Using the developed system, we define and evaluate log contexts of “it is 5 degree colder than yesterday” and “it is the coldest day today among these years” in CS27-HNS, in order to demonstrate practical feasibility.

Key words context, context-aware service, house log, environment sensor, query

1. はじめに

センサ技術や M2M 技術の発展にともない、実世界における様々な情報を取得・収集可能になってきている。コンテキスト

とは、センサやシステムから取得した情報に基づいて定義された「状況情報」を指す。例えば、温度センサから「現在の気温が 10℃未満」という情報が得られた場合、これに対して「寒い」という状況をひもづけることで、「寒い: 気温 < 10」という

コンテキストを定義できる。同様に、音量センサを用いた「騒がしい」や、電力会社が提供する電力使用率を用いた「電力が逼迫している」といったコンテキストも定義できる。

こうしたコンテキストを利用することで、その場その時の状況に合わせたアクションを自律的に実行するコンテキストウェアサービスを実現できる。例えば「寒い」というコンテキストに「暖房をつける」というアクションをひもづければ、「寒い時に自動的に暖房をつける」というコンテキストウェアサービスが実現できる。コンテキストウェアサービスはユビキタスコンピューティングの分野で盛んに研究されている。我々の研究グループでは、ホームネットワークシステム (HNS) [1] 内のコンテキストウェアサービスを作成・管理・運用するための様々な枠組み (例えば、センササービスフレームワーク SSF [2], センササービスバイнда SSB [3], RuCAS プラットフォーム [4]) を提案・開発している。

従来多くのコンテキストウェアサービスでは、センサやシステムの現在の値を利用してコンテキストを定義してきた。そのため表現可能なコンテキストが現在の状況に限定される。例えば上記の「寒い」というコンテキストは現在の気温のみで表現できた。しかし「昨日より 5℃以上寒い」というコンテキストは、過去の情報が必要となるため表現できない。

そこで我々の最新の研究 [5] では、ホームネットワークシステム (HNS) に蓄積された住宅ログを活用することで、過去の状況も考慮したより高度なコンテキスト (ログコンテキストと呼ぶ) を表現する手法を提案している。住宅ログとは、HNS 内の環境センサの計測値や機器の操作、消費電力等の住宅の情報を周期的に記録し、蓄積した履歴情報 (ログ) である。例えば「昨日より 5℃以上寒い」というログコンテキストは、住宅ログに対して「昨日同時刻の気温センサの値」を問い合わせ、現在の気温と比較することで、次のように表現できる: 「昨日よりずっと寒い: 現在の気温 \leq 昨日の気温 - 5℃」。なお、この先行研究 [5] では、住宅ログを用いたログコンテキストの定義方法と利用体系について提案を行っているが、具体的なシステムの実装にはいたっていない。

本研究の目的は、先行研究 [5] を実際のシステムとして実装し、その実用性を確認することである。本研究では、我々が以前より開発・運用している実際の HNS 環境 (CS27-HNS) [6] を使用する。また、CS27-HNS で蓄積している住宅ログの中から、環境センサの値とそのログを対象を絞り、それらを利用した環境ログコンテキストを取得するシステムを設計・実装する。

本研究では、住宅ログを利用したログコンテキストをできるだけ容易に作成・管理できるように、システムを 4 つの層 (レイヤ) で構成している。

- (1) アプリケーション・レイヤ: ログコンテキストを利用してコンテキストウェアサービスを実行・管理する層。
- (2) ログコンテキスト・レイヤ: ログコンテキストを定義し、定義式に基づいてコンテキストの値を評価・取得する層。
- (3) ログクエリ・レイヤ: ログコンテキストを構成するための住宅ログへの問い合わせ (ログクエリ) を管理する層。
- (4) DB 接続レイヤ: 住宅ログを蓄積する DB に接続し、

ログクエリに基づいて過去の状況情報を取得する層。

実装したシステムを利用して、CS27-HNS において下記の 2 つのログコンテキストを定義し、取得できることを確認した。

- 「昨日より 5℃以上寒い」
- 「ここ数年で最も寒い」

実装の結果、提案手法は CS27-HNS の環境コンテキストの表現の幅を大きく広げられることがわかった。また、実装したログコンテキストを用いることで、「より気の利いた」コンテキストウェアサービスの実現が期待できる。

2. 準備

2.1 ホームネットワークシステム (HNS)

ホームネットワークシステム (HNS) は様々な家電機器やセンサをネットワークに接続し、計算機による制御を可能とすることで新たな付加価値を提供するシステムである。我々の研究室では、サービス指向アーキテクチャ (SOA) を適用し、家電やセンサの機能を Web サービスとして利用できる HNS 環境 CS27-HNS を開発している [6]。各家電の操作やセンサからの値の取得は、SOAP または REST 形式の Web-API を介して実行でき、様々なプログラムに容易に組み込める。例えば、温度センサの値を取得するには `http://cs27-hns/TemperatureSensor/getValue` に、エアコンを暖房でつけるには `http://cs27-hns/AirConditioner/on?mode=heating` というように、URL にアクセスするだけでよい。Web-API の実行結果は XML 形式で返される。

2.2 住宅ログ

住宅ログとは、HNS 等の情報化住宅において取得可能な、機器操作や環境状態、消費電力量といった情報を記録・蓄積した履歴情報 (ログ) を指す。我々の研究グループでは、文献 [7] において住宅ログを以下の 3 つに分類している。

- エネルギーログ: 家庭内で利用された、エネルギーの消費履歴を指す。水道、ガス、電力などの履歴が含まれる。
- 機器ログ: 家庭内に設置された家電機器全般に関する操作履歴と状態履歴を指す。いつ、誰が、どの機器に対して、どのような操作を行ったのか、といった情報が含まれる。家電機器としては、テレビや冷蔵庫、エアコンなどがある。
- 環境ログ: 家庭内外の環境状態に関するログを指す。住宅内や住宅周辺の気温、湿度、照度といった環境情報や、在宅人数などが含まれる。

CS27-HNS では、エネルギーログとして 32 種類の機器の消費電力を 3 秒毎に記録している。また、機器ログとして、HNS で操作された家電機器の操作ログを蓄積している。さらに、環境ログとして、室内外の気温、湿度、照度の値や天気、部屋の在室人数などを 1 分毎に記録している。

蓄積した住宅ログは MySQL 関係データベースで管理されており、SQL を用いて過去の任意の時刻の必要な属性のログを自由に検索可能である。CS27-HNS では、数年に渡り住宅ログを蓄積しており、その件数は膨大に及ぶ。例えば、環境ログは約 4 年、5000 万件のセンサデータを蓄積している。

2.3 コンテキストウェアサービス

コンテキストは、様々なセンサやシステムから収集した情報

に基づいて定義される「状況情報」である。コンテキストウェアサービスとは、あるコンテキストが成立した時に、適切なアクションを自律的に起こすサービスである。

我々の研究グループでは、HNSにおけるコンテキストウェアサービスを手軽に作成し、統合的に管理するフレームワークについて研究を行っている [2] [3] [4]。これらのフレームワークでは、各コンテキストを [コンテキスト名: 成立条件式] の形で定義する。ここで、コンテキスト名はそのコンテキストを識別するラベルで、通常そのコンテキストの意味づけを表す文字列で表現する。一方、成立条件式は、値をセンサやシステムから取得可能な変数を用いた論理式で定義される。成立条件式の変数に現在の値を代入し評価することで、コンテキストは真 (true) または偽 (false) の値をとる。

例えば、「寒い」というコンテキストを「現在気温が 10℃未満である」という成立条件式で表現すると、次のようになる。

```
[ 寒い : 現在の気温 < 10 ]
```

ここで、「現在の気温」という変数は、HNS内の温度センサから値をとると仮定している。このセンサの値が 10℃未満の場合、上記のコンテキストは true、すなわち「寒い」が成立する。上記の従来フレームワークでは、成立条件式で利用可能な変数として現在の値を取得できるものに限定している。そのため、現在の状況から評価できるコンテキストしか表現できなかった。例えば、「今寒い」というコンテキストは表現できるが、「昨日より寒い」というコンテキストは表現できない。

2.4 先行研究：住宅ログを活用したコンテキストの拡張

渡邊ら [5] は、住宅ログを活用することで過去の状況情報を考慮したより高度なコンテキスト (ログコンテキスト) を定義する方法を提案した。具体的には、コンテキストの成立条件式に住宅ログを用いて計算可能な変数を許すような拡張を行った。

例えば「昨日の同時刻の気温」という変数は、住宅ログを検索して、現在と同時刻の昨日の気温を取得する事で求められる。実際には、住宅ログを管理するデータベースに対してクエリをかける事で、取得する事ができる。そのため「昨日の同時刻の気温」のような変数をログクエリと呼ぶ。文献 [5] ではログクエリを SQL 文で定義しており、本論文では各ログクエリをログクエリ名 {SQL 文} の形で表現する。例えば、「昨日の同時刻の気温」は次のように表す。

```
昨日の同時刻の気温 {  
  SELECT: AVG(Temperature)  
  FROM:   EnvironmentLog  
  WHERE:  Date >= 昨日の同時刻-5 分&&  
         Date <= 昨日の同時刻+5 分 }
```

この SQL 文は、住宅ログの中の環境ログ (EnvironmentLog) のうち、日付が昨日の同時刻± 5 分のものを検索し、条件に合致した行の気温の平均値を求めている。つまりこの SQL を実行することで、ログクエリ「昨日の同時刻の気温」の値が求まる。

このログクエリを用いれば「昨日より 5℃以上寒い」という

ログコンテキストを次のように定義できる。

```
[ 昨日より 5℃以上寒い :  
  現在の気温 + 5 <= 昨日の同時刻の気温 {  
    SELECT: AVG(Temperature)  
    FROM:   EnvironmentLog  
    WHERE:  Date >= 昨日の同時刻-5 分&&  
         Date <= 昨日の同時刻+5 分 } ]
```

上記で、「現在の気温」は、センサより取得した気温であり、それに 5℃を加えた値が、右辺のログクエリ「昨日の同時刻の気温」以下である事を表している。つまり、「昨日より 5℃以上寒い」というコンテキストを表現している事になる。

なお、文献 [5] では上記の枠組みの提案に留まっており、具体的なシステムの実装を行っていない。そこで本研究では、CS27-HNS 環境においてログコンテキストの取得システムを実装し、提案手法の実用性を確認する。今回の実装においては、CS27-HNS で蓄積している環境ログと環境センサを対象を限定し、環境ログコンテキストを取得するシステムを目指す。

3. 住宅ログを活用した環境コンテキスト取得システムの実装

3.1 アーキテクチャ

本節では、2.4 で述べたログコンテキストを実際の HNS から取得・評価するためのシステムの実装を行う。実装においては、様々なアプリケーションからログコンテキストをできるだけ容易に作成・管理可能なアーキテクチャを決めるべきである。そのために、ログコンテキストを構成する多数の要素 (コンテキスト名、成立条件式、ログクエリ、ログクエリ名、SQL 文、住宅ログ DB など) を段階的に詳細化する階層化アーキテクチャを採用する。具体的には、実装するログコンテキスト取得システムを以下の 4 つの層 (レイヤ) で構成する。

(1) アプリケーション・レイヤ: ログコンテキストを利用してコンテキストウェアサービスを実行・管理する層。

(2) ログコンテキスト・レイヤ: ログコンテキストを定義し、成立条件式に基づいてコンテキストの値を評価する層。

(3) ログクエリ・レイヤ: ログコンテキストを構成するための住宅ログへの問い合わせ (ログクエリ) を管理する層。

(4) DB 接続レイヤ: 住宅ログを蓄積する DB に接続し、ログクエリに基づいて過去の状況情報を取得する層。

上記アーキテクチャに基づいて設計したシステムのクラス図を図 1 に示す。以降の節で、各レイヤについて詳細に説明する。

3.2 アプリケーション・レイヤ

アプリケーション・レイヤは、ログコンテキストを利用してコンテキストウェアサービスを実行・管理するレイヤである。図 1 のクラス図は [4] で提案されている RuCAS プラットフォームにおけるコンテキスト管理を踏襲したクラス設計になっている。RuCAS プラットフォームでは、コンテキストウェアサービスをイベント、コンディション、アクションの組を用いた ECA ルールで定義する。このうち、イベントとコン

ンサ (またはシステム) から値を取得できるようになる。

アプリケーション・レイヤにおいては、ログコンテキストも通常の (住宅ログを用いない) コンテキストも同じ方法で管理するほうが合理的である。したがって、ログコンテキストを外部センサの一種とみなし、Adaptor を介して値を取得することで、既存の RuCAS プラットフォームを改造することなくコンテキストアウトウェアサービスを作成・管理できる。各クラスの説明は以下のとおりである。

- Context : 現在の状況によるコンテキストを表す抽象クラス。属性として、コンテキストの識別子である contextid, 型を表す type, 成立条件式 expression, コンテキストの説明 description などを持つ。メソッドとして、センサ等から取得した値を expression に代入する resolve(), 成立条件式を評価する eval() などを持つ。

- Adaptor : コンテキストの成立条件式の変数にひもづく外部センサ (またはシステム) からの情報を取得するためのアダプタ。本実装では、ログコンテキストを外部センサとして呼び出し、その成否を getValue() の戻り値として返す。

- ContextManager : コンテキストの管理を行うクラス。Context と Adaptor は、このクラスに集約されている。

3.3 ログコンテキスト・レイヤ

ログコンテキストを定義し、成立条件式に基づいてコンテキストの値を評価するレイヤである。LogContext クラスは、Context と同様にコンテキストの識別子 contextid, 成立条件式 expression, 説明 description を属性として持っている。また、2.4 で見たように、ログコンテキストは成立条件式の中にログクエリを変数として持つことが出来るため、使用するログクエリの集合を保持する queries を持つ。varMap は、成立条件式内の変数名と値を対で記録するマップであり、成立条件式 expression の評価に利用される。getContext() は、ログコンテキストの値を取得するメソッドである。内部で varMap の値が更新され (update()), expression の評価が行われる。

2.4 の「昨日より 5℃以上寒い」コンテキストを例にすると、LogContext のインスタンスは次のようになる。

```

contextId: 昨日より 5℃以上寒い
description: 現在の気温が昨日の同時刻の気温より 5℃以上低い時 True を返す。
expression : currentTemp + 5 <= yesterdayTemp
queries: [currentTemp, yesterdayTemp]
varMap : {currentTemp:6.3, yesterdayTemp:13.6}
    
```

上記のインスタンスにおいて、現在の気温 currentTemp は住宅ログを用いなくても温度センサから直接取得可能である。しかしこの例では、currentTemp も yesterdayTemp と同様にログクエリとみなして、最新の住宅ログから値を取得することで処理を統一化している。

3.4 ログクエリ・レイヤ

ログコンテキストの成立条件式で利用されるログクエリを定義・管理するレイヤである。あるログコンテキストで定義した

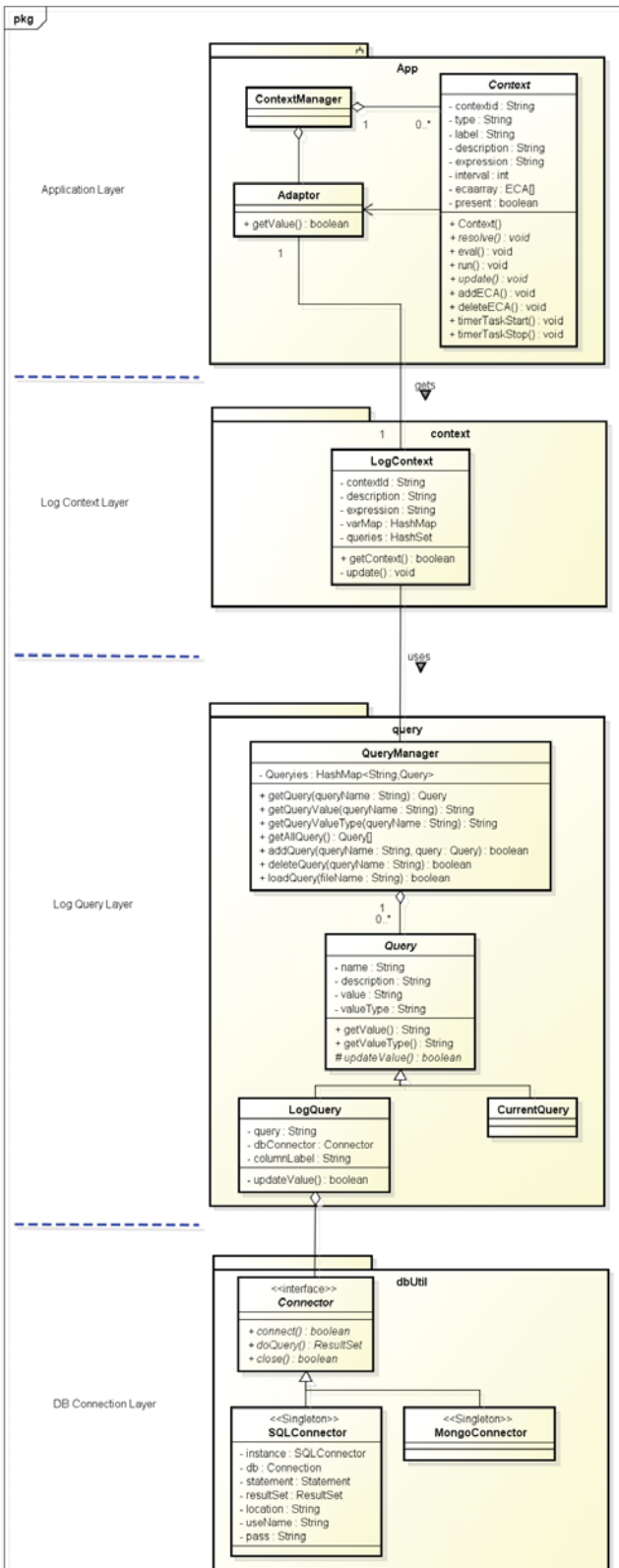


図 1 実装したシステムのクラス図 powered by Astal

ディクションは、コンテキストを用いて定義される。各コンテキストは、センサやシステムから取得された値を利用して定義される。この時、値を取得するためのインタフェース (API) はセンサやシステムによって異なるため、Adaptor クラスをはさむことで、プラットフォームは統一的方法 getValue() で当該セ

ログクエリを別のログコンテキストから再利用しやすくするため、ログクエリの管理を QueryManager クラスで一括で行う設計にした。

QueryManager は、作成した全てのクエリ queries を管理するクラスであり、クエリの追加 addQuery(), 取得 getQuery(), 削除 deleteQuery() を行うメソッドを持つ。各クエリにはクエリの ID を指定してアクセスする。

Query は、ログクエリ LogQuery と住宅ログを用いない通常のクエリ CurrentQuery を抽象化するクラスである。クエリの名前 name, 説明 description, 値 value, 型 valueType を属性として持つ。

LogQuery は、具体的なログクエリを表すクラスである。属性として SQL 等で書かれたクエリの実行文 query, 住宅ログの DB への接続 dbConnector 等を持つ。メソッド updateValue() が実行されると、当該 DB に接続し、クエリの実行文を実行し値を取得、属性の value を更新する。

2.4 のログクエリ「昨日の同時刻の気温」を例にすると、LogQuery のインスタンスは次のようになる。

```

name: yesterdayTemp
description: 環境ログの昨日同時刻の気温センサの値を検索し、値を求める。
value: 13.6
valueType: double
query:
SELECT AVG( Temperature ) FROM 'EnvironmentLog'
WHERE ( 'Date' = DATE(NOW()) - INTERVAL 24 HOUR
AND ( 'Time' >= TIME(NOW()) - INTERVAL '24:05'
HOUR_MINUTE ) AND ( 'Time' <= TIME(NOW()) -
INTERVAL '23:55' HOUR_MINUTE )
dbConnector: MySQLConnector

```

上記のインスタンスでは、住宅ログを問い合わせる DB として MySQL を想定している。query の SQL 文における WHERE 節は、「昨日と同時刻 ± 5 分」を MySQL の文法で表現している。

3.5 DB 接続レイヤ

住宅ログを蓄積する DB に接続し、指定されたクエリを実行して過去の状況情報を取得するレイヤである。本研究では、Connector インタフェースを介して、様々な種類のデータベースの接続に対応できるように設計した。Connector インタフェースでは、DB に接続する connect(), クエリを実行する doQuery(), DB の接続を閉じる close() の 3 つのメソッドが規定されている。具体的な DB コネクタは、Connector インタフェースを実装する形となる。現在、MySQL への接続を行う SQLConnector と MongoDB への接続を行う MongoConnector が存在する。

3.6 ログコンテキスト取得の流れ

ログコンテキストは、図 1 で定義した 4 つのレイヤのクラスの相互作用によって計算される。このログコンテキストの取得までの流れを図 2 のシーケンス図に示す。この例では 2.4 の「昨日より 5℃以上寒い」というログコンテキストを取得する

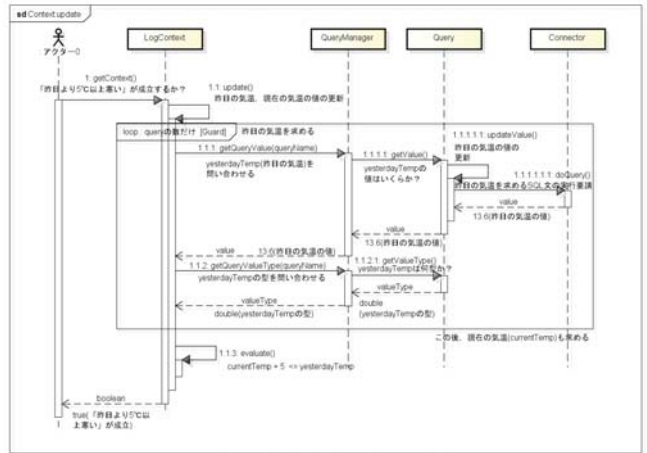


図 2 ログコンテキスト取得の流れ

流れを表している。

まずアプリケーションが、「昨日より 5℃以上寒い」ログコンテキストの getContext() を実行し、コンテキストが成立するか否かを問い合わせる。ログコンテキストは問い合わせに対して、update() を実行し、全ログクエリ (yesterdayTemp, currentTemp) の値の更新を行う。具体的には、各ログクエリについて、クエリマネージャにその値を問い合わせる。例えば、yesterdayTemp を問い合わせると、クエリマネージャは、yesterdayTemp のログクエリに対して値を問い合わせる。クエリは MySQL コネクタに昨日の同時刻の気温を問い合わせる SQL 文を実行するように要請し、求めた値をクエリに返す。その後、値はクエリマネージャを介して、ログコンテキストに返される。この作業を同様に currentTemp にも行う。ログコンテキストは最後に evaluate() を行い、成立条件式 currentTemp + 5 <= yesterdayTemp の評価を行う。最後に評価された値は真偽値としてアプリケーションに戻される。

4. ケーススタディ

実装したシステムおよび CS27-HNS で取得している環境ログを用いて、以下の 2 種類のログコンテキストを定義し、取得するケーススタディを行った。

- (C1) 昨日よりずっと寒い
- (C2) ここ数年で最も寒い

ログコンテキスト C1 は、現在の温度センサの値と、環境ログから呼び出した昨日の同時刻の気温とを比較し、コンテキストの成立を判定する。昨日の同時刻の気温は、昨日の同時刻の前後 5 分の温度のログの平均の値とする。現在の気温が昨日の同時刻の気温よりも低く、かつその差がある閾値 x より大きかった場合に、「昨日よりずっと寒い」コンテキストが成立する。判定の基準となる閾値 x は簡単に変更でき、季節やユーザの好みに合わせた設定ができるようにしている。ログコンテキスト C1 は以下のように定義される。

```
[ (C1) 昨日よりずっと寒い:
現在の気温 + x < {
SELECT: AVG(Temperature)
FROM: EnvironmentLog
WHERE: Date >= 昨日の同時刻-5分 &&
Date <= 昨日の同時刻+5分 } ]
```

次に、ログコンテキスト C2「ここ数年で最も寒い」は、現在の温度センサの値と、環境ログに記録された過去の同日の同時刻の気温のうち最も低い値を比較する。ログを遡る年数は変数 y で表し、現在を含めてその値の数と同じ年数分の気温の値を参照する。ログコンテキスト C2 は以下のように定義される。

```
[ (C2) ここ数年で最も寒い:
現在の気温 < {
SELECT: MIN(Temperature)
FROM: EnvironmentLog
WHERE: MONTH(Date) = 今日と同じ月 &&
DAY(Date) = 今日と同じ日 &&
Time > 現在の同時刻-1分 &&
Time < 現在の同時刻+1分 &&
YEAR(Date) < 今年の年 &&
YEAR(Date) > 今年の年 - y } ]
```

実装したシステムにおいて、上記 2 つのコンテキストを取得するのに要する時間を計測した。その結果、ログコンテキスト C1 は約 1 秒、ログコンテキスト C2 は約 1 秒 50 ミリの時間がかかった。また、ログコンテキスト内の閾値 x , y を変更しても、取得時間に変化は認められなかった。住宅ログを利用したコンテキストの評価が予想していたよりも短い時間でできることを確認した。その結果、開発したシステムはハードなリアルタイム性を要する場合以外には、実用に耐える見通しを得た。

5. 考察

環境ログを利用したログコンテキストを定義・利用できることになったことで、よりリッチなコンテキストウェアサービスが可能となる。例えば「昨日よりずっと寒い」「ここ数年で最も寒い」といったログコンテキストを用いれば、より強力な暖房対策や外出時の衣服を厚着にするアドバイスなど、よりきめ細やかで気の利いたコンテキストウェアサービスを実現できる。また、昨日よりどれだけ寒いか、過去何年間で最も寒いのか、といったコンテキスト成立の閾値も柔軟に変更できるため、ユーザの好みや状況に合わせたコンテキストを定義できる。

一方で「昨日よりずっと寒い」といったログコンテキストが成立していても、暖房を利用する事が必ずしもユーザにとって望ましいとは限らない。たとえ寒かったとしても、過去に同様の状況で暖房を利用していなければ、今回も暖房を利用しないだろう。また光熱費の都合から、暖房の利用を控えたいと思っているケースも考えられる。ユーザの満足するサービスを実現

するためには、環境ログだけでは限界がある。サービスをより高度で、ユーザの需要に正確に対応できるようにするためには、機器ログやエネルギーログによるログコンテキストも併せて活用する必要がある。

6. おわりに

本稿では、ホームネットワークシステムで蓄積している住宅を活用したログコンテキストを取得するシステムを設計・実装した。「昨日より寒い」「ここ数年で最も寒い」といった過去の状況も考慮したコンテキストを定義できるようになった事で、より表現の幅の広がったコンテキストウェアサービスの実現が期待できる。

今後の課題としては、より多様なコンテキストを定義できるように、環境ログ以外のエネルギーログや機器ログにも対応することを考えている。また、実際にユーザに様々なログコンテキストを定義してもらい、コンテキストの精度や満足度を評価する実験も行っていきたい。

謝辞 この研究の一部は、科学技術研究費（基盤研究 C 24500079, 基盤研究 C12877795, 基盤研究 B 23300009）、および、積水ハウスの研究助成を受けて行われている。

文献

- [1] M. Nakamura, A. Tanaka, H. Igaki, H. Tamada, and K. Matsumoto, "Constructing home network systems and integrated services using legacy home appliances and web services," International Journal of Web Services Research, vol.5, no.1, pp.82-98, 2008.
- [2] M. Nakamura, S. Matsuo, S. Matsumoto, H. Sakamoto, and H. Igaki, "Application framework for efficient development of sensor as a service for home network system," the 8th IEEE 2011 International Conference on Services Computing (SCC), pp.576-583, 2011.
- [3] M. Nakamura, S. Matsuo, and S. Matsumoto, "Supporting end-user development of context-aware services in home network system," Studies in Computational Intelligence, vol.443, pp.159-170, Nov. 2012.
- [4] 高塚広貴, 佐伯幸郎, まつ本真佑, 中村匡秀, "異種分散 web サービスに基づくコンテキストウェアサービスの管理プラットフォームの実装," 信学技報, 第 113 巻, pp.71-76, LOIS2013-36, 11月 2013. 2013年 11月 28日(木)- 11月 29日(金) 東北大学 (ISEC, LOIS).
- [5] 渡邊雄一, 増田哲也, まつ本真佑, 佐伯幸郎, 中村匡秀, "ホームネットワークにおける住宅ログを活用したコンテキストウェアサービスの検討," 信学技報, 第 113 巻, pp.77-82, LOIS2013-37, 11月 2013. 2013年 11月 28日(木)- 11月 29日(金) 東北大学 (ISEC, LOIS).
- [6] 中村匡秀, "スマートハウスにおける家庭内ネットワークシステム," KEC 情報, vol.●●, no.216, pp.32-40, Jan. 2011.
- [7] S. Yamamoto, S. Matsumoto, and M. Nakamura, "Using cloud technologies for large-scale house data in smart city," International Conference on Cloud Computing Technology and Science (CloudCom2012), pp.141-148, Dec. 2012. Taipei, Taiwan.