

## コンテキストの再利用に関する考察

中村竜也、山田大輔、中尾太郎  
株式会社NTT データ 技術開発本部  
{nakamurattc, yamadad, nakaotr}@nttdata.co.jp

あらまし：アプリケーション主導に基づくコンテキストの構造化では、構造化された情報として扱うことができるコンテンツとしてコンテキストを扱い、コンテキストを再利用して新たなコンテキストを定義する方法を提案している[1][2]。コンテキストが再利用される状況においては、コンテンツとコンテキストの境界が曖昧であり、コンテキストの振る舞いやそれを管理するコンテナに求められる機能の具体化が、ともすると高機能化されたコンテンツ管理となんら異ならないという結果に陥ってしまう。本稿では、実際のケーススタディにおいて求められたコンテキストの再利用要件と、そのケーススタディから得られたフィードバックに基づき、再利用されるコンテキストに求められる構造化について考察する。

### 1 はじめに

電子タグによる個の特定とその状況をセンシングするセンサネットワーク技術の進展により、ユーザやモノの置かれている環境や状況を理解し、能動的な情報サービス提供を行おうとするコンテキストウェアコンピューティングに対する注目が高まっている。図-1にコンテキストウェアコンピューティングの概念図を示すように、コンテキストウェアコンピューティングは、受動的な計算機利用を前提とするのではなく、ユーザやモノを中心とし、コンテキストに基づくシステム側からの能動的な情報サービス提供を行うことを目的としており、ユビキタスコンピューティングにおける情報サービス提供概念として捉えることが可能である[3]。

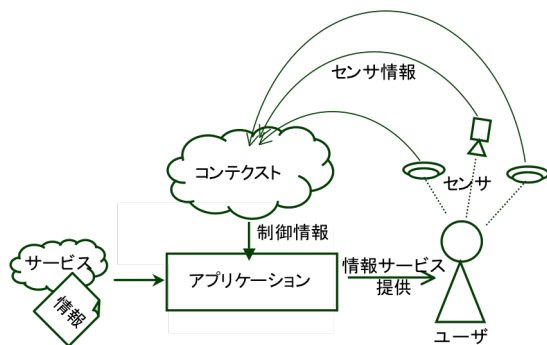


図-1 コンテキストウェアコンピューティングの概念図

コンテキストウェアコンピューティングのゴールは、実世界情報の監視と解釈を代行することで、ユーザやモノに提供される情報サービスの制御情報を提供することにある。このゴールを達成することにより、情報サービス提供者はユーザやモノの多様なコンテキストの解釈を意識することなく、制御情報に基づくサービス

アプリケーションの開発を行うことができる。つまり、CRM や SCM などユーザやモノに関する情報管理者とその情報に基づきサービスを提供する者が互いに独立しながらも、コンテキストに基づく情報サービスを構成することができるようになる。すなわち、制御情報の生成論理とその利用論理を分離しながら、状況や環境に応じた情報サービスを構成できるようになる。制御情報の生成論理とその情報の利用論理を分離することが多様化するコンテキストに基づく情報サービス提供には必須となるものと予想される。そのためにも実世界から取得される各種情報を抽象化し、コンテキストとしてモデリングすることが欠かせないと考えている。

これまでに、Schmidt や Dey らにより画一的な構成要素に基づくモデルが提案されている[4][5]。構成要素が画一的なことから、コンテキストのコンポーネント化を実現できるというメリットが主張されているが、その枠に収まらない多様なコンテキストは存在し、これに対処するため対象領域に閉じた構成要素の拡張が一般化しており、モデルのコンセプトと実世界への適用では乖離が生じていると考えざるをえない。

筆者らは、Schmidt や Dey らによるコンテキストの構造化方式（以降モデル主導に基づく構造化方式）とは異なり、情報サービス提供者がそのサービス設計時に、制御情報とするコンテキストを自由に構造化し、必要に応じてコンテキストの参照を繰り返しながらコンテキストを次々と構造化する方式（以降アプリケーション主導に基づく構造化方式）を提案している

[1][2]。アプリケーション主導に基づくコンテキストの構造化方式では、コンテキストのコンポーネント化や共有性をモデル化のコンセプトとして捉えるのではなく、コンポーネント化や共有化はこれまでに蓄積された概念情報共有フレームワークを援用し実現される機能要件として捉えている点異なる。

アプリケーション主導に基づくコンテキストの構造化では、コンテキストを構成する要素を固定しないことから、情報サービス提供者の思惑に沿った複雑なコンテキストの構造化が可能となる。また、実世界の状況や環境を制御情報としてどのように構造化して捉えるかは、コンテキストの構造としてアプリケーション設計者から事前に提供されるため、計算機による自律的な環境理解に頼る必要がない。

情報サービスが次々に作成されれば、これに伴ってさまざまなコンテキストが情報サービス提供者により構成されることとなる。こうして構成されたコンテキストは、それぞれが集約した情報（以降コンテンツ）に基づき情報サービス提供者の視点から状況や環境を理解し表現したものであるといえる。このコンテキストを、抽象度の高まった状況や環境情報として再利用し、新たなコンテキストを構成できれば、既存の状況や環境理解を活用したより抽象度の高いコンテキストの構造化が可能となる。

このように、コンテキストがコンテンツとして再利用される状況では、コンテキストの何が再利用されコンテキストを再定義するのか検討する必要がある。なぜならば、コンテキストはユーザやモノを取り巻く状況や環境の解釈の結果生成されたコンテンツとしてだけではなく、情報サービスの制御情報としての側面も持っており、制御情報の伝播する経路を指示する構造とも捉えることができるからである。

筆者等は、アプリケーション主導に基づくコンテキストの構造化を採用したコンテキストウェアフレームワーク YACAN (Yet Another Context-AwareNess) を開発しているが、コンテキストの実体化や記述に際して、コンテキストのコンテンツ的側面と制御情報伝播記述を混在させていた。

本稿では、コンテキストの再利用が求められたケーススタディを実施することにより得られたフィードバックを考察し、コンテキストの構

造を再定義するとともに、その構成方法を示す。また、再利用を前提としたコンテキストを流通させるための機能要件を示す。

以下、2章でアプリケーション主導に基づくコンテキストの構造化方式について触れ、コンテキストウェアフレームワーク YACAN の概要を示す。3章では、コンテキストの再利用が求められたケーススタディの概要を示し、実際に YACAN による具体化で得られたフィードバック情報を考察する。4章では、前章で得られた考察をもとに、再利用を前提としたコンテキストの再定義を行う。5章では、コンテキストを再利用するために、求められるコンテキストの検索サービスなどいくつかの機能要件を示す。最後に今後の課題に触れるとともに本稿をまとめる。

## 2 アプリケーション主導に基づくコンテキストの構造化

センシング技術の高度化やそれらを有機的に結合するセンサネットワークの実用化に伴い、ユーザやモノを取り巻く状況や環境を表現する要素情報が多様化している。これら情報に基づき、状況や環境情報を一般的な方法で計算機可読なコンテキストとして構造化する試みがなされてきた。

ここではアプリケーション主導に基づくコンテキストの構造化について概要を示す。詳細については[1][2]を参照願いたい。

### 2.1 先行研究経緯

Schmidt らは、コンテキストを computing context と user context、physical context の3つのカテゴリに分類することでコンテキストの記述を一般化しようとした[4]。しかし、Chen らによって time context や context history という概念の不足が指摘されるなど、Schmidt らのアプローチは不完全に終わった[5]。Dey らは primary context として location と entity、activity、time を採用し、これらをインデックスとしたより複雑な contextual information を構成するアプローチを提案した[6]。しかし、Schmidt らの構造化方式と同様、定義した4つのインデックスの範疇に収まらない多様化したコンテキストが依然多数存在する。これを例証するように、Dey らのモデル主導に基づくコンテキストの構造化を採用する多くの研究開発では、primary context の拡張を適用領域に合わせて行っている。Dey らは、primary

context を規定することで、コンテキストのコンポーネント化、共有化を達成できるとしているが、primary context を適用領域に合わせて拡張する段階でコンセプトは破綻しているものと考えられる[7]。primary context とそれに索引される contextual information の関係をオントロジで表現するアプローチも見られるが[8]、コンテキストの推論に議論が集中しており、primary context に収まらないコンテキストの扱いについては詳述されていない。

## 2.2 アプリケーション主導に基づくコンテキストの構造化

筆者らは、コンテキストはコンテンツのメタな情報であり、コンテンツとコンテンツを取り巻く情報（コンテンツ）の間の概念として捉え、ユーザやモノに提供される情報サービスを構成しているアプリケーションの設計者がコンテキストの構造や処理方法を理解しているという考察に基づき、情報サービス提供者が情報サービス提供の制御情報となるコンテキストを構造化するアプリケーション主導に基づくコンテキストの構造化方式を提案してきた[1][2]。

状況や環境を表現するコンテンツを組み合わせ、計算機が自律的にコンテキストを準備することは認識や経験的手続きの援用など困難を伴う作業となるが、サービス提供者はどのようなコンテンツが得られればコンテキストとして十分であり、またそのコンテキストに従い何を処理すべきか知っているはずであるから作業は明示的である。つまり、計算機による自律的なコンテキストの構成は困難であるが、特定サービス向けの例示は容易である。

コンテキストは構造化されたあるスキーマに基づく情報であるからコンテンツとして扱うことが可能である。つまり、コンテキストをコンテンツとして参照するコンテキストの定義が可能であり、情報サービス提供者の視点から抽象化されたコンテキストを更に抽象化して情報サービスを提供することが可能となる。

コンテキストは、あるコンテンツとそれを取り巻く環境が決定するコンテンツの間の概念構造であるから、対象となるコンテンツが異なれば実体化されたインスタンスとしてのコンテキストも異なる。また、環境が決定するコンテンツが変化すれば、それに伴い実体化されるコンテキストも異なる。

コンテキストウェアコンピューティングでは、コンテキストの生成や更新、消滅を契機として、コンテキストを制御情報とするアプリケーションを呼び出し、ユーザやモノにコンテキストに応じた情報サービスを提供する。

## 2.3 YACANフレームワークの概要

YACAN は、アプリケーション主導に基づくコンテキストの構造化を採用したコンテキストウェアコンピューティングのためのフレームワークである。

図- 2に YACAN の情報参照モデルを示す。

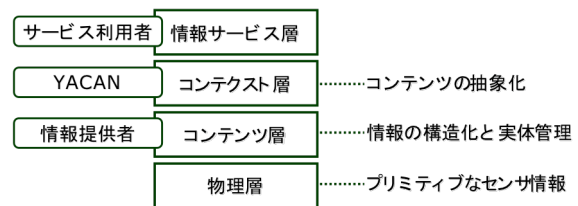


図- 2 YACAN の情報参照モデル

物理層は、センサネットワークにおいてセンサから取得できる温度や電子タグ情報などプリミティブなデータを扱う層である。この層の情報は、数字であったり文字列であったりするが、それ自体では意味をなさない。

コンテンツ層は、物理層から得られるプリミティブな情報を構造化し意味を持たせ、計算機可読な情報を構成する。YACAN が扱うコンテンツについては、情報構造が公開されていることを前提とし、構造に従ったコンテンツの参照が行えることを前提としている。

物理層とコンテンツ層は YACAN 外部のアクタにより提供されることを想定している。

コンテキスト層では、コンテンツの変化とそれによって活性化されるサービスの関連付けを行う。YACAN はコンテンツからコンテキストを構成し、情報サービス提供者に代わって実世界あるいは仮想世界について記述されたコンテキストの変化を監視することでコンテキスト層を担い、コンテキストに応じたトリガの発行を管理する。コンテキストは OWL を用いてオントロジとして記述される。

コンテキストは、コンテキストに集約されるコンテンツの参照先が決定した時点で実体化される。実体化されたコンテキストは、コンテンツの要素を参照するための URI (xsd:anyURI) を値として持つが、コンテンツの要素値それ自体を要素として持たない。これは、コンテキストの概念を対象

となるコンテンツから独立させるためである。

異なる適用領域で定義されたコンテキストの共有性や拡張性、相互運用性については、オントロジーを基板としたフレームワークにより実現される。

サービス層では、コンテキスト層からのトリガに応じてサービスアプリケーションを実行し、ユーザに情報サービスを提供する。情報サービス提供者は YACAN 外部のアクタとして捉えている。

情報サービス提供者は、提供するサービスアプリケーションの制御情報となるコンテキストを定義し YACAN に登録する。

図- 3 に YACAN の概念構成図を示す。YACAN はコンテンツプロバイダとコンテンツブローカ、コンテキストマネージャの3つのコンポーネントから構成される。

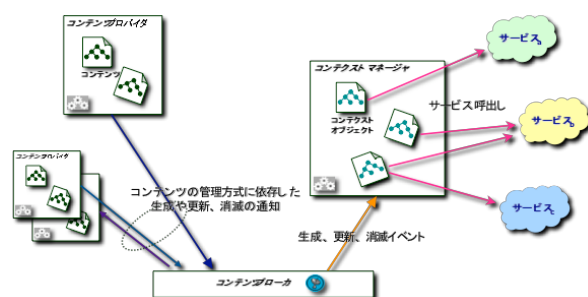


図- 3 YACAN の概念構成

コンテンツプロバイダは、コンテンツを保持するシステムにおいて、YACAN に対して自律的にコンテンツの変化イベント（生成と更新、消滅）を通知する機能を持ち、状況や環境を構成するコンテンツを提供するシステムである。イベント通知機能については、コンテンツプロバイダが提供する場合と、YACAN がアダプタ実装として提供する場合を想定している。後者の場合、YACAN が提供するアダプタ実装が監視を代行し、変化イベントをコンテンツブローカに通知する。

コンテンツプロバイダは、YACAN 外部アクタでありデータベースや web ページなど任意の形式で情報を管理するが、それら情報は公開されたスキーマに従う URI により一意に参照できるものとする。

コンテンツブローカは、コンテンツプロバイダが通知するイベントの多様性を吸収し、正規化されたイベント（生成と更新、消滅）をコンテキストマネージャに通知する。また、URI を単位として複数のコンテキストがイベントを同時に受信するためのイベント通信モデル(Publish/Subscribe

通信モデル) を利用している。

コンテキストマネージャは、コンテンツブローカから配信されるコンテンツ更新イベントを受信し、コンテキストに従ったアプリケーションを呼び出す。コンテキストマネージャは、情報サービス提供者が記述したコンテキストのスキーマを格納するコンテキストデータベースを持つ。コンテキストマネージャは、コンテンツブローカからイベントを受信すると、コンテキストデータベースを検索し実体化可能なコンテキストを選択しコンテキストを実体化する。実体化されたコンテキストは、コンテキストマネージャのライフサイクルに基づき管理される（リソース管理、永続化/活性化管理等）。

コンテキストマネージャがイベントに従い呼び出す情報サービスは、基本的にステートレスであることを想定している。

図- 3 に示した各コンポーネントは分散配置が可能であり、コンテンツプロバイダが通知するイベントの規模により、コンポーネントの配置を自由に行うことが可能である。

### 3 ケーススタディに関する考察

具体化した YACAN フレームワークを用いて、いくつかのケーススタディを実施している。それら事例の中で、定義したコンテキストを再利用し、サービスの構成を階層化するケースがあった。

以下では、その事例の概要を紹介するとともに、得られたフィードバックについて考察する。

#### 3.1 降雨量に基づく災害情報通報システムの概要

地域の降雨量に基づき災害対策部の管理者に通報を行う災害情報通報システムの概要を図- 4 に示す。

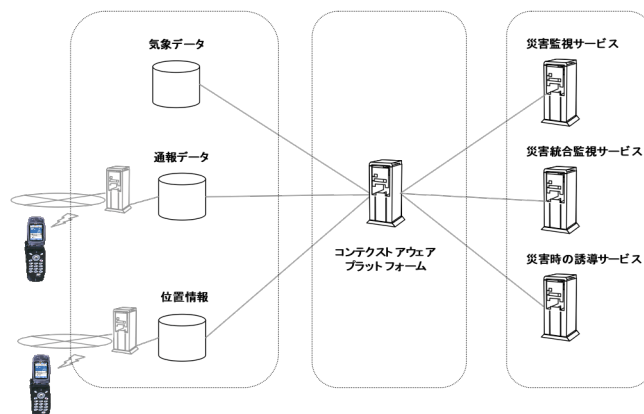


図- 4 災害情報通報システムの概要

このシステムは、地域の降雨量を蓄積する気象データベースと地域の住民が携帯電話などから実際に発生した災害情報を受け付け、地域ごとに蓄積する通報データベース、地域の住民の位置情報を管理する位置情報データベースをコンテンツとし、地域の危険度を管理するコンテキストを記述、その危険度に従い、所轄地域の災害を管理する管理者に指示（電子メールやメッセージングによる）を与えるサービスと呼び出す。

サービスとしては、降雨量に基づく地域の危険度を通知する災害監視サービスと、降雨量に基づく危険度だけではなく、実際に発生した災害に基づく危険度を通知する災害統合監視サービス、更に地域住民の位置情報を加味して、誘導情報を提供する誘導サービスから構成されている。

サービスの構成から分かるように、一連のサービスは、地域の降雨量に基づき算出された危険度を基底として、これに通報情報を加えた統合危険度、更に地域住民の位置情報を加えた位置危険度に基づき階層化されている（図- 5 参照）。それぞれのサービスは独立しており、災害監視を行う管理者の設定により、すべてのサービスを同時に提供する場合もあれば、特定のサービスのみを提供する場合もある。

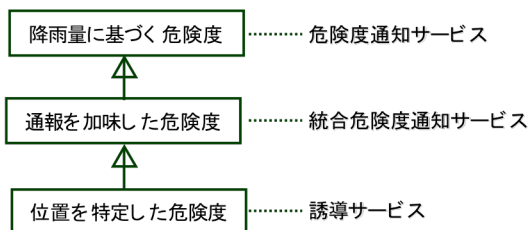


図- 5 災害情報サービスの階層化

この災害情報通報システムは、降雨量から算出される危険度をコンテキストとし、コンテキストを再利用してより抽象度の高い危険度を定義することが要件とされるコンテキストアウェアコンピューティングであると捉え、実際に YACAN を適用してケーススタディを実施した。

### 3.2 コンテキストの構成

災害情報通報システムでは降雨量に基づく危険度を基底とし、それを拡張することでコンテキストを再利用している。図- 5 に示した危険度は、基底となるコンテキストを拡張して定義されるが、それぞれの危険度を示す値は同様であり、黄（降雨量を観測しているが、過去に遡り災害は発生していない）、橙（過去に災害が発生した降雨量を観

測している）、赤（過去に災害が多発した降雨量を観測している）が割り当てられる。

図- 6 が実際にケーススタディに使用したコンテキストの構成図である。

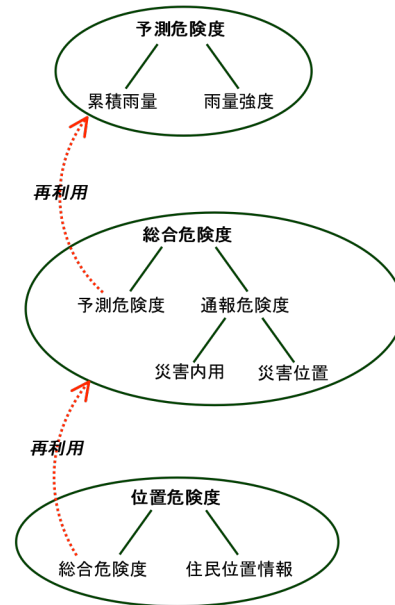


図- 6 コンテキストの構成

図- 6 に示したそれぞれの危険度コンテキストは OWL により個別のオントロジとして記述される。図- 6 中、矢印で示される再利用の関係はオントロジに反映されるものではなく、それぞれの危険度コンテキストが実体化される際に、どの危険度コンテキストの実体に基づき実体化されているのかを示している。

### 3.3 フィードバックと考察

降雨量に基づく災害情報通報システムでは、降雨量に基づく予測危険度を基底として、総合危険度と位置危険度を算出する。つまり、再利用される危険度コンテキストは永続化される値を持ち、その危険度をコンテンツとして公開していることが求められる。

また、図- 6 のコンテキストスキーマに従い実際の危険度コンテキストが実体化されるが、総合危険度コンテキストや位置危険度コンテキストについては、どの予測危険度コンテキストあるいは総合危険度コンテキストに基づいて実体化されたのか管理する必要が生じる。つまり、危険度コンテキストとして永続化される値が、どのコンテンツ（危険度コンテキスト）に基づき決定されたのかを管理しておかなくては、再利用した危険度コンテキストが更新された場合、イベントの伝播を

行うことができなくなってしまう。

ケーススタディを行った時点での YACAN フレームワークには、以上のような管理機能は具体化されておらず、独自データベースを用いた管理システムにより実現した。これらの機能をコンテキストの記述とコンテキストマネージャの機能として具体化することができれば、任意のコンテキストを再利用することが可能となる。

以上より、コンテキストを再利用する場合には下記要件を具体化する必要があると考察できる。

- 1) コンテキスト値をコンテンツとして公開できる
- 2) コンテキスト値が決定されるために必要となるコンテンツの関連を管理する
- 3) トリガイベントは2で管理される関連情報に基づき伝播される

次章では、この考察に基づき、再利用されることを前提としたコンテキストの再定義を行い、それを具体化するフレームワークの概要を示す。

#### 4 コンテキストの再定義

この章では、3章のケーススタディから得られたコンテキスト再利用のための要件を反映し、コンテキストを再定義する。

また、コンテキストの再利用を前提とした YACAN フレームワークの再検討を行う。

##### 4.1 再利用されるコンテキストの値とは？

再利用されるコンテキストの値は、コンテキストに集約されたコンテンツの要素値を情報サービス提供者が処理した結果を示す。3章で示した災害情報通報システムの場合、予測危険度コンテキストは、累積雨量と雨量強度に基づき、情報サービスである危険度通知サービスが処理した値（危険度）を永続化する。この予測危険度が生成あるいは更新、消滅すると、予測危険度を参照して決定されるコンテキストである総合危険度にイベントが通知される。

つまり、再利用されるコンテキストの値は、情報サービス提供者のなんらかのロジックが、コンテキストスキーマに指定されたコンテンツに基づいて算出し永続化したコンテンツであると捉えることができる。

情報サービス提供者は、どのコンテンツを集約して情報サービスを提供するか事前に把握しているだけではなく、永続化したいコンテキストの値について構造化を行うことが可能である。この構

造を公開することにより、永続化したコンテキストの値をコンテンツとして公開することが可能となる。

コンテキストが永続化したい値の構造をコンテキストの構造の一部として捉え、コンテキストがコンテンツとして再利用される場合に必要となるスキーマを提供する。

##### 4.2 要素展開と実体化されたコンテキストの関連管理

3章の考察で示したように、どのコンテンツに基づいてコンテキストの値が決定されたのかを管理することにより、コンテキストがコンテンツとして再利用される際のイベント伝播経路を決定する。

コンテキストのスキーマを記述する段階では、どの要素値に基づいてコンテキストが生成されるかは確定されないため、イベントの伝播経路はコンテキストが実体化される段階で決定され管理されなくてはならない。

ところで、どのようなコンテンツに基づきコンテキスト値を決定し永続化するかは情報サービスを構成するサービスアプリケーションに依存するが、アプリケーション側でこれを管理させることは、情報サービス提供者を外部アクタとして分離していることに反することとなる。

そこで、どのような単位のコンテンツに基づいてコンテキストの値が決定されるのかを、コンテキストのスキーマとして記述させ、コンテキストマネージャがこれを単位として、コンテキストの永続化値とコンテンツの関連を管理するようにしたい。

コンテキストを実体化する際の要素としてのコンテンツの単位には、

- 1) 同一スキーマを持ち一意の URI で参照される個体
- 2) 1の中で生成あるいは更新、消滅した個体
- 3) 同一スキーマに基づき実体化されたコンテンツの集合

が考えられる。このコンテンツ単位を、コンテキストを決定するコンテンツの記述に利用することができれば、コンテキスト値の実体化に利用されたコンテンツを特定することができ、コンテンツの変更イベントを伝播させることが可能となる。

コンテキストは単一のコンテンツから定義されるものではないから、コンテキストに集約されるコンテンツの単位を展開した数だけコンテキスト

値が存在することとなる (図- 7 参照)。

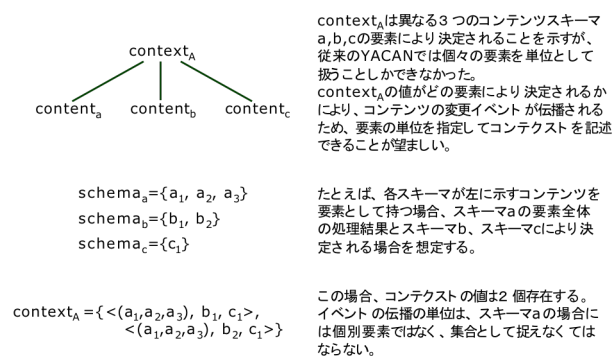


図- 7 要素展開とその値に基づくコンテキスト値管理

### 4.3 コンテキストの再定義

以上の検討に基づきコンテキストの構造を再定義する。以降定義する要素は、[1][2]で定義した基底コンテキストのコンテキスト属性として再定義する。

コンテキストの永続化値を以降アスペクトと呼ぶ。アスペクトは情報サービス提供者により構造化される。アスペクトの構造に従いコンテキストの永続化値がコンテンツとして公開される。

アスペクトを決定するためのコンテンツの記述をパラメータと呼ぶ。パラメータは、コンテンツスキーマに従い実体化されているコンテンツの参照単位を示す記述である。アスペクトを決定するコンテンツの単位に従いパラメータが展開され、コンテキストが実体化される段階でパラメータの組み合わせに従うアスペクトがコンテキストマネージャにより対応付けられる。

コンテキストの実体化条件記述をコンディションと呼ぶ。コンディションはアスペクトを決定するコンテンツのスキーマを単位とした論理式として記述される。図- 7 の例では、スキーマ b とスキーマ c の要素が変更されなければ、コンテキストを実体化しないことを指示する論理式 B and C を記述する。この論理式により、パラメータを展開する制約を与え、不用意なアスペクト (イベントの伝播を必要としない不要なアスペクト) の生成を抑制する。

基底コンテキストは、従来の基底コンテキストのコンテキスト属性に加え、アスペクトとパラメータ、コンディションをコンテキスト属性として持つ。それぞれのコンテキスト属性は、従来の方式と同様に OWL のオントロジとして記述し、コンテキストデータベースに登録される。

### 4.4 フレームワークの構成

図- 8 にフレームワークの概念構成図を示す。

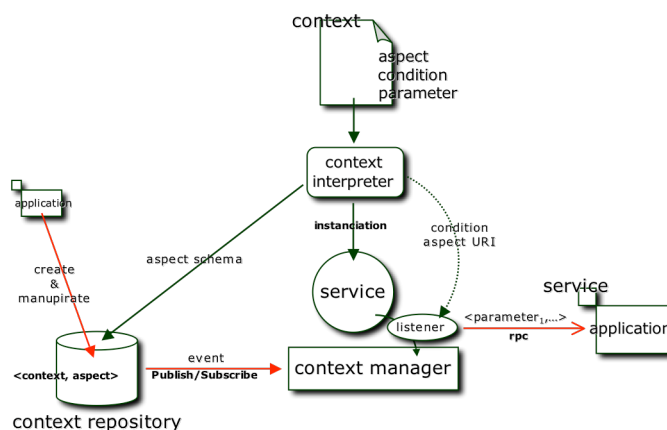


図- 8 フレームワークの概念構成

コンテキストに記述されたアスペクトとコンディション、パラメータは、コンテキストインタプリタにより解釈され、コンテキストマネージャが管理するサービスオブジェクトを生成すると同時に、パラメータで指定されたコンテンツの単位に従いイベントを受信するためのリスナを登録する。

アスペクトのスキーマは、アスペクトをコンテンツとして公開するため、コンテンツプロバイダを拡張して実装されるコンテキストレポジトリに登録される。コンディションで指定される論理式を満足するイベントが受信されると、情報提供者が準備したアプリケーションが呼び出される。その結果アスペクトが実体化される。

コンテキストが再利用される場合には、コンテキストレポジトリに格納されたアスペクトをコンテキスト定義のパラメータとして指示する。コンテキストレポジトリはコンテンツプロバイダを拡張して実装されるため、アスペクトの参照は従来通り xsd:anyURI により指示することが可能である。

コンテキストレポジトリとコンテキストマネージャ間はイベントの形式を規定して構わないため、従来コンテンツブローカとコンテキストマネージャ間で採用していた Publish/Subscribe 通信モデルによるイベント伝播を行う。

## 5 コンテキストの再利用を前提とした機能要件

### 5.1 アスペクトのインデクシングと検索サービス

コンテキストを再利用する際には、コンテキストレポジトリに格納されたアスペクトのスキーマ公開とそのスキーマに基づく実体としてのアスペクトを検索できなくてはならない。

コンテキストレポジトリに存在するアスペクトについては、アスペクトを決定するためのコンテンツを管理している。そのため、アスペクトのスキーマとアスペクトの実体を指示できれば、コンテキストの再利用が可能である。

アスペクトの検索条件式として、何を与えるかについては、コンテキストの流通とそのメリットについて検討しなくてはならない。

現段階では、既知のアスペクトスキーマのインスタンスを列挙する機能のみ具体化しているが、コンテキストの流通がもたらす効果についての検討を進めた段階で、検索サービスの具体化やオントロジに基づく分散インデクシングや検索などの技術を援用したいと考えている。

### 5.2 コンディションの論理式評価と同時性の扱い

コンディションはコンテキストを実体化する条件を論理式で与えるための記述である。この論理式の評価は、コンテンツが変化する時間軸のあるスナップショットにおいて評価されなくてはならない。ところで、コンテンツはYACANの外部アクタであるため、時間軸やその精度が均一であることが期待できない。そのため、コンディションを評価するコンテキストマネージャが単位時間を与え、その時間内で変化イベントを受け付け、スナップショットを生成し評価する方式を検討している。

論理式のオペレータとして許しているnotについては、その単位時間内で変化イベントが受信されていないことを示すものとして解釈している。

コンディションの論理式記述と評価、同時性の扱いについては検討を進めている段階であり、性能や品質を確保するために、いくつかの戦略を立案している。

## 6 まとめ

本稿では、コンテキストの再利用に着目したアプリケーション主導に基づくコンテキストの構造化方式とそれを導入したフレームワークについて検討した。

コンテキストを、プリミティブな情報から抽象度の高いコンテンツとして扱い、流通させ再利用することで、コンテキストの語彙の幅を広げるこ

とが可能となる。サービス提供者による状況や環境の解釈を蓄積することで、多様化する状況や環境をコンテキストとして定義できるのではないかと考えている。

現段階では突き詰めて検討していないが、コンテンツの生成イベントを契機とした適用可能なコンテキストの検索や情報サービスのコンテキストに依存した決定、異なる適用領域間における相互運用性など、ケーススタディを行いながら検証する予定である。

## 参考文献

- [1] 中村竜也、山田大輔、中尾太郎、“コンテキストウェアネスとコンテキストの定式化”、人工知能学会第7回セマンティックウェブとオントロジ研究会、SIG-SWO-402-03、July 2004
- [2] 山田大輔、中村竜也、中山一美、“コンテキストの構造化に関する考察とコンテキストウェアフレームワーク”、情報処理学会研究報告、2004-UBI-5、pp63-70、June 2004
- [3] M. Weiser、“The computer of the 21<sup>st</sup> century”、Scientific American、pp94-100、September 1991
- [4] Albrecht Schmidt, Michael Beigl, and Hans-W. Gellersen, “There is more to Context than Location”, Proceedings of the International Workshop on Interactive Applications of Mobile Computing (IMC98), Nov 1998.
- [5] G. Chen and D. Kotz、“A Survey of Context-Awareness Mobile Computing Research”、Technical Report TR2000-381、Dept of Computer Science、Dartmouth College、November 2000
- [6] A. K. Dey、“Understanding and Using Context”、Personal and Ubiquitous Computing、Vol. 5、No. 1、pp4-7、2001
- [7] Huadong Wu、“Sensor Data Fusions for Context-Aware Computing Using Dempster-Shafer Theory”、Ph. D. Dissertation Abstract、Carnegie Mellon University
- [8] Tao Gu, Xiao Hang Wang, Hung Keng Pung, and Da Qing Zhang, “An Ontology-Based Context Model in Intelligent Environments”、Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS'04), Jan. 2004.