3H-1

マルチテナント型テナントアクセス制御方式に関する研究

○小杉　優†　　楓　仁志†　　佐藤　雅之†　　山足　光義†

三菱電機株式会社　情報技術総合研究所†

1. はじめに

データセンタ上のインフラにインターネットを介して情報システムを利用する、クラウドコンピューティング及び SaaS(Software as a Service)へ移行する企業（テナント）が増えてきている。クラウドコンピューティングでは、サービス提供者にとっては、ハードウェアリソースやソフトウェアを集中して管理できるため保守性が向上され、サービス利用者にとっても従来、パッケージを購入する必要があったソフトウェアをサービスという形態で利用可能なため、導入コストを抑えられるというメリットがある。例えば、図１のようにデータセンタ上に各テナントが共用可能な WEB アプリケーション及びデータベースを構築することで、インターネット経由でテナントユーザが利用する形態のサービスを提供可能である。このサービス運用において、多くのテナントに対して同種のサービスを提供し、同一のアプリケーションを共用させつつ、ソフトウェア・ハードウェアリソースを削減するアプリケーションの集約度向上のための技術が重要となる。

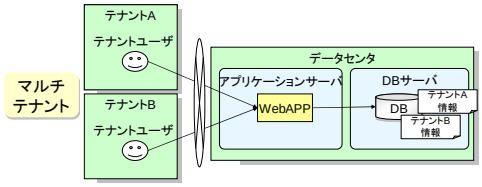

図 1：マルチテナント型サービス例

マルチテナント型のサービスを提供する際に、従来のシングルテナント用の既存ソフトウェアをマルチテナント化する要求が高まっている。

2. 課題

既存ソフトウェアをマルチテナント型アプリケーションに改修に当たって以下の点を考慮して開発を行う必要がある。

(1)セキュリティの担保

マルチテナント型のアプリケーションでは複数の企業によってアプリケーションが共用されるため、セキュリティを担保しながらアクセス制御を実施する必要がある。そのために、既存アプリケーションに対し、テナントを識別した上でアクセス可否を実現する仕組みが必要となる。

(2)改修量の削減

既存ソフトウェアをマルチテナント化する際には、既存ロジックにテナントを識別するためのロジックを追加する必要があるが、全てのロジックに対してテナント識別ロジックを追加する場合、改修量が増大になってしまう。そこで可能な限り既存部分に手を加えないで改修可能かどうかが重要である。

上記技術として、[1]ではレガシーアプリケーションで用いていたパラメータにテナント情報を埋め込む「マルチテナントデータ切替制御方式」を示した。この方式では、既存アプリケーションのユーザインタフェース部分及びデータベース接続部分について切り出し、既存アプリケーションで利用しているパラメータ文字列にテナント情報を付加することで改修量を抑えつつテナントの切替を行うことが可能となった。

しかし、[1]を実際のアプリケーションに適用する場合は、必ずしも工数削減につながらないケースが明らかになった。既存アプリケーションへの入力として渡すパラメータは、操作する対象のデータ(個人情報、組織情報等)によって異なるため、データの種別ごとにどのパラメータに対してテナント情報（テナントコード、テナント名、分離種別などテナントに関する情報）を付加するのかを個別に選定（パラメータの選定）する必要がある。データ種別が多ければ多いほど、既存パラメータに対し、テナント情報を埋め込む作業及びそれらの確認作業のみ実施というわけにはいかず、使用しているデータクラスについてそれぞれパラメータの選定、改修を個別にする必要が生じてしまうことが明らかになった。

Tenant Access Control Method for Multi-tenancy Application
†Yu Kosugi, Satoshi Kaede, Masayuki Sato,
Mitsuyoshi Yamatari,
Information Technology R&D Center, Mitsubishi Electric Corporation

3. 解決方法

課題の解決策として[1]の技術を改良した「マルチテナント型テナントアクセス制御方式」を示す。

「テナントアクセス制御方式」では、既存アプリケーションに対しテナント情報等マルチテナント化により新たに追加されたパラメータをハッシュマップ形式で保持するために基底クラスを追加し、この基底クラスを他のデータクラスで継承を行う方式をさす。また、キーを指定することで任意のパラメータを取得・更新することを可能にするメソッドや、共通パラメータ自体を取得、設定可能にするメソッドを提供する。

図 2 では、本手法で利用する基底クラスの例を示している。既存基底クラスではハッシュマップ形式のパラメータを用意し、テナント情報等のマルチテナント化の際に追加になったパラメータを保持する。また、テナント情報を取得するためのメソッドを提供する。

この基底クラスを個人情報や組織情報等の既存のデータクラスから継承するようにすることで、共通パラメータをそれぞれのデータクラス間で利用可能となる。

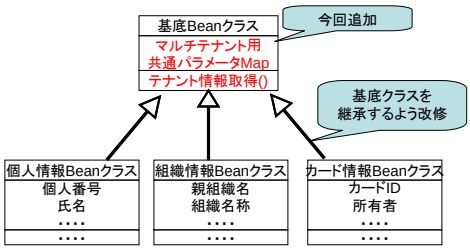

図 2：基底クラスの例

解決手法の構成図を図 3 に示す。従来アプリケーションをラップする形でマルチテナントアプリケーションが提供される。その際、ユーザと既存アプリケーション間及び、既存アプリケーションとデータベース間のやりとりは UI 接続部、DB 接続部が実現する。

実際のアプリケーション上では、以下のように動作させることでテナント識別子のようなテナント情報を受け渡しが可能となる。

①テナントAユーザが UI 接続部にアクセスする。

②UI 接続部でテナント A のテナント情報を個人情報データクラスの共通パラメータに設定する。

③既存アプリケーションは、テナント情報を意識せずに従来どおりの処理を実施する。

④DB 接続部は個人情報データクラスの共通パラメータからテナント情報を取得し、データベース検索時の SQL に埋め込みを行う。

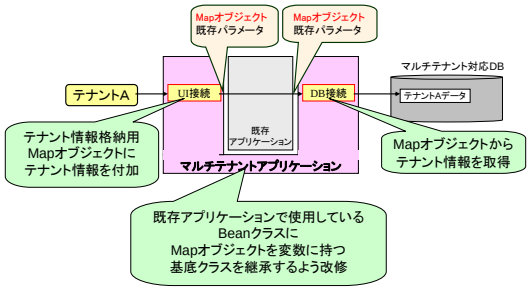

図 3：マルチテナント型テナントアクセス制御方式

4. 評価

今回提案した手法について、あるソフトウェアをマルチテナント化する際にかかる工数を机上で評価を行った結果を表 1 に示す。

表 1：工数の比較

既存手法	提案手法
100	79.18

※既存手法を 100 に換算

上記のように提案手法によって約 2 割程度の工数削減が達成可能となることが分かった。

5. おわりに

本報告書では、既存アプリケーションをマルチテナント化する際の方式について、「マルチテナント型データ切替制御方式」を改良した「マルチテナント型テナントアクセス制御方式」を検討した。今回の評価では、本提案手法が有効であることが確認できた。今後は、他のアプリケーションについても同様の検証を実施し評価を実施するとともに、更なる改良を検討する。

参考文献

[1] 小杉,入不二,小川,山足,「マルチテナント対応データ切替制御に関する研究」, 2011, 第 74 回情報処理学会全国大会論文集,4H-5,4-525～4-526