

非同期 P2P 通信ミドルウェア —アプリケーション制御部の研究—

†伊東正起 †澤口宗和 †松原裕人 †大谷真

湘南工科大学 情報工学科

1 はじめに

Web サービスとは、インターネット内のシステム間でアプリケーションをクライアントサーバ型かつ同期型で接続する技術である。一方で、企業間、個人間での一連のメッセージ変換には、対等かつ非同期型の接続が必要である。特に、自律型システム間でのプロトコル自動整合では、このような接続が欠かせない。このため、非同期接続型 (P2P) 通信のミドルウェア (PMAW) を開発することにした。本論文では PMAW の AP (アプリケーション) 制御について開発方針とプロトタイプ開発結果について述べる。

2 PMAW の構成

図 1 に PMAW の構成を示す。

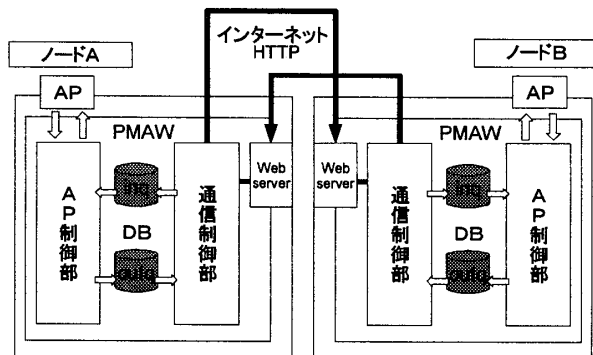


図 1 PMAW の構成

PMAW は AP に対し、非同期のメッセージング機能を提供する。PMAW は AP 制御部と通信制御部からなり各々は DB (データベース) を Stored and Forward 方式のキューを介してやり取りを行う。AP 制御部は、API を経由した AP とのやり取りや情報の管理を担当し、通信制御部はミドルウェア間のメッセージ通信を担当する。HTTP サーバを利用し、ミドルウェア間の通信プロトコルに HTTP (SOAP フレームワーク) を利用する。

PMAW では LongSession という概念を導入し、HTTP 上での非同期メッセージングを実現する。

LongSession とは PMAW 内で定義される通信路であり、lsid で一意に判別される。図 2 に示すように、AP1 と AP4 が一連のメッセージ交換を開始するとき LongSessionA が設定され終了時に解除される。

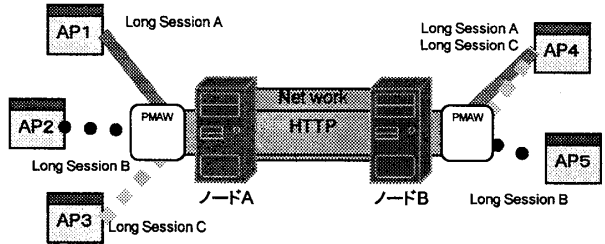


図 2 LongSession による AP 結合

LongSession を確立する際に ua (ユーザーエージェント) が必要となる。ua とはメッセージングの端点であり、各 ua には URI などによりユニークな名前が与えられている。

また、PMAW は Web サービスと同様に WSDL でシステムの外部インターフェース仕様を記述することを想定している。外部インターフェースはオペレーション (op) の集合である。op は送信時又は受信されたときに実行したい関数名に対応する。AP 内部で任意に定義できる。

3 キューの実現と API 仕様

3.1 キューの実現

図 1 内のキュー (inq と outq) は、特別のファイル構造を用いず DB で表現することとした。AP 制御と通信制御の同期は DB を介して行うことにより単純化させた。表 1 ~ 3 にキューを構成する主要なテーブルを示す。

表 1 inq と outq の共通カラム

カラム	内容	s_op	送信側 op
id(PK)	serialID	r_op	受信側 op
lsid	lsid	Message_data	送受信する情報
s_ua	送信側 ua	message_num	メッセージ通番
r_ua	受信側 ua		
s_url	送信側 url	status	情報状態
r_url	受信側 url		

表 2 outq だけのカラム

カラム	説明
send_id	送信情報 ID
time	DB に書き込まれた時間

表 3 inq だけのカラム

カラム	説明
Receive_id	受信情報 ID

Asynchronous P2P Communication Middleware
-Study on Application Manager-

† Masaki Itou, Hirokazu Sawaguchi, Hiroto Matsubara, Makoto Oya - Shonan Institute of Technology

3. 1 API仕様

PMAWのAPIはLongSession Classのクラスメソッドとして各種機能を実装した。表4にその一部を示し、図3に使用例を示す。

表4 API関数一覧

関数	説明
コンストラクタ	LongSession内の固定値の指定・決定
send	非同期送信
check_send	送信結果の確認(非ブロック型)
wait_send	送信結果の確認(ブロック型)
receive	受信対象のop指定
check_receive	非ブロック型の受信
wait_receive	ブロック型の受信
lsdestroy	LongSessionの終了

```

ls = new LongSession(ua1, ua2, url1, lurl2); LSの生成
sid = send(s_op, "TV h8型");           データ送信
res = wait_send(sid);                   送信の完了待ち
rid = receive(r_op);                     データの受信
res = wait_receive(rid);                 受信待ち
destroy();                               LSの終了
    
```

注 ua1, url1 は自システムの ua2, url2 は相手システムの ua, url s_op, r_op は自分のシステムの送信 op, 受信 op

図3 API使用例

4 実装

PMAWのアプリケーション制御部はAPとはLongSessionクラスを経由し、DBとは確立されたLongSessionに対応するAccessClassを経由しやり取りを行う。AccessClassはlsid, ua, urlといったLongSessionの固定的情報の保持とDBに対する読み書きなどの役割を持つ。

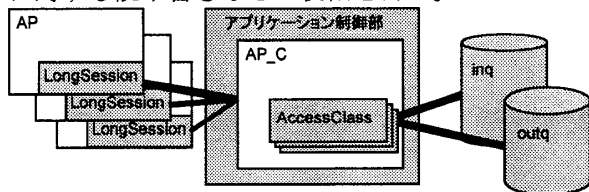


図3 アプリケーション制御部の接続

LongSessionを確立するまでの流れを図4に示す。LongSessionを確立するにはlsidを確定する必要がある。lsidは、通信相手と生成用データの送受信を行いそのデータをもとに生成される。この処理は、LongSessionクラスを作成する際に、再帰的にLongSessionを確立することで実現した。実際の流れとしてLongSessionクラスの作成を行うと、そのコンストラクタ内部でlsid生成するためLongSessionクラスが作成され、lsidが確定するまでAPにLongSessionクラスのインスタンスは戻らず待たされることとなる。

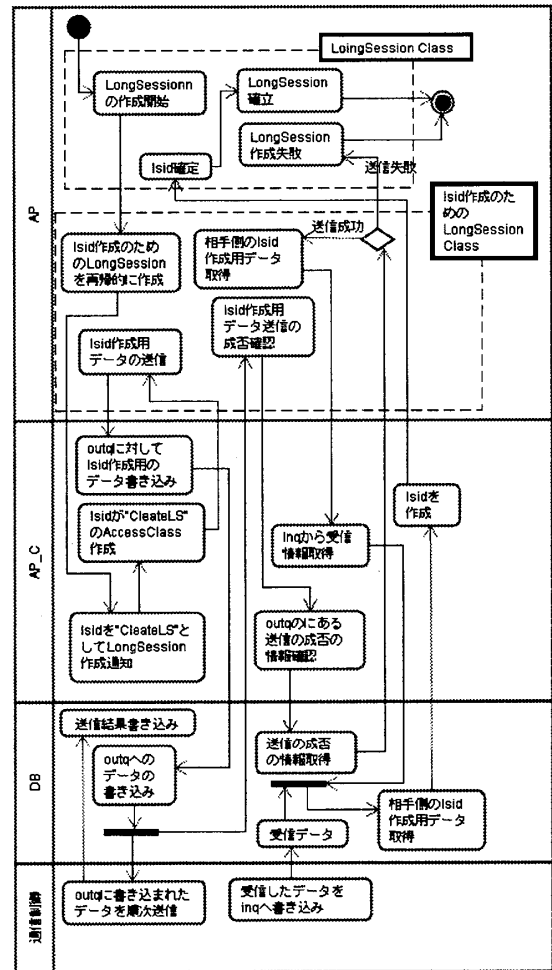


図4 LongSession 確立の流れ

5 考察とまとめ

本論文に述べた方式およびAPI仕様によって、AP制御のプロトタイプを開発した。言語はPHP、データベースはPostgreSQL、Web ServerはApacheを使用した。同時に開発した通信制御部と組み合わせ、異なるシステム上のAP間でのメッセージ送信テストを、シングルプロセス・マルチプロセスに関して行い、目標仕様通りに動作することを確認した。これによりAP制御およびPMAW全体の方式設計の妥当性が検証できた。

一方、PHPを使用したことが主要因で、性能・スレッド制御に不十分な点が残った。今後、本研究結果を元に、Javaなどを用いて本格的なPMAWの研究を開始する予定である。

6. 参考文献

1. 澤口他「非同期P2P通信ミドルウェア-通信制御部の研究-」情報処理学会第70回全国大会, 2008
2. Simple Object Access Protocol (SOAP) 1.1 <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>
3. Web Services Description Language (WSDL) 1.1 <http://www.w3.org/TR/wsdl>