

イーサネットフレームによるオブジェクト間通信 Inter-Object Communication through Ethernet Frames

平中 幸雄* 伊藤大視*† 市川直樹*‡ 武田 利浩*
Yukio Hiranaka Taishi Ito Naoki Ichikawa Toshihiro Taketa

1. まえがき

さまざまなオブジェクト間の通信を一般化して表現するため、著者らは汎用通信フォーマット UCF(Universal Communication Format)と呼ぶデータ形式を提案している[1]。共通の通信プロトコルを単純な形式と最小限の機能条件で規定することを目指して、「何でもリモコン」などへの応用を試みてきたが[2]、これまでは TCP/IP などを利用してきたために、UCF のデータ形式としての実用的意義が薄れていた。今回、イーサネットフレーム上に直接 UCF データを載せることで、より実際の通信形態とすることを目指した。また、NS 2 ネットワークシミュレータに直接 UCF データを送受させることにより、通信オブジェクトをエミュレーションできるようにした。実際に音データを通信するオブジェクトを作成し、確認実験を行ったので報告する。

2. UCF の目的と概要

UCF を機器間通信の標準形式として使うことで、通信およびアプリケーション間の接続性向上を狙っている。通信の最小限の表現は、宛先指定とデータの組である。この組の表現を単一化すれば、通信の互換性が実現する。しかし、実際的には効率向上のため、その組情報がさまざまな圧縮形式で表現されることになる。そこで、プログラム上の一般形表現形式として定義し、未知の相手との通信に使用すべき初期通信形式をUCFの機能としている。相互の通信が可能になれば、交渉に基づき任意のデータ表現形式に移行し、効率化・暗号化などを行えばよい。

具体的には、次のようにすべてASCII文字で表し、宛先を現すオブジェクト名をタグ<A>にし、そのタグでデータを挟む形で通信全体を表す。

```
<A> data </A>
```

入れ子を許すことで、宛先のオブジェクト構造を考慮する。

```
<A> <C> data </C> </A>
```

送信元情報は、以下のように送信者タグ<s>で表し、相手の<s>タグ処理内部オブジェクトへのデータとする。

```
<A> <s> B </s> data </A>
```

オブジェクト自体の伝送は、オブジェクト定義のための内部オブジェクトへのデータ送信で行う。

```
<A> <def> <name> C </name> <java> Java source  
</java> </def> </A>
```

フロー制御、エラー制御などのプロトコル機能も、それを実現する内部オブジェクトへの通信の形で実現することができる。最も簡単なフロー制御としては、1個のデータを送信後待機し、<ack>へ次のようなデータが返ってくれば、1個分先のデータへ進むという方法がある。

```
<B> <s> A </s> <ack> 1 </ack> </B>
```

3. イーサネットフレームでの形式

上記の UCF データをイーサネットフレームのデータ部分に直接入れるようにした。イーサタイプは仮の値を設定した(0x1234)。宛先 MAC アドレスは、ブロードキャストアドレスを用いた。UCF 通信の一般化の観点からは、イーサネットアドレスを意味のないものとして扱う方がよいが、実用的観点からは、UCF の先頭オブジェクト名と MAC アドレスの変換を行い、データ量の節約とハードウェア処理の利用をはかることが考えられる。

データ部分も ASCII 文字表現とするため、実行プログラムや音データなどのバイナリ値は、16進表示での伝送を基本とする。Base64 やその他の表現方法、圧縮方法はオブジェクト相互の交渉によるオプション機能として実現すればよい。

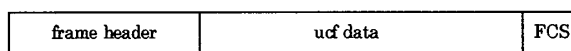


図1 イーサネットフレームでの UCF データ

4. UCF 通信オブジェクトの実現方法

FreeBSD 6.0R 上に、BPF (Berkeley Packet Filter) を利用して図2のオブジェクト機能として動作するプログラム(ucfinit)を作成した。

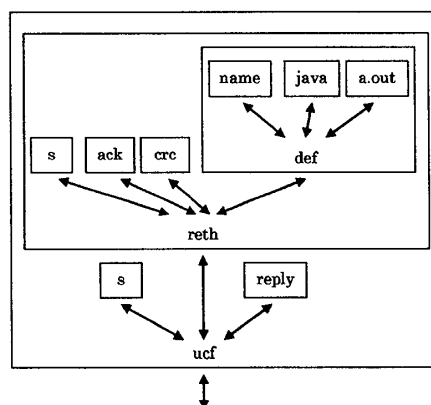


図2 UCF 通信 PC のオブジェクト構造

* 山形大学, Yamagata University

† 現在, 東北大学大学院, Tohoku University

‡ 現在, 株式会社アルフ, Alf Inc.

UCF データを受け取る ucf オブジェクトが外部に対するオブジェクト名で、その内部にはたいのオブジェクトを持つべき、<s>タグや<reply>タグを処理する内部オブジェクトがある。reth は目的とする機能のオブジェクトであるが、その内部にも<s>タグや<ack>タグを処理するプログラムと、外部からのオブジェクト作成を受け付ける def オブジェクトがある。

一方、別の PC 上に NS-2 エミュレータ (nse) をインストールし、そのエミュレータ上で図3のシミュレーションを C++と OTcl スクリプトを用いて作成した。NS-2 上のアプリケーションとして、図4のように UCF で通信するオブジェクト (nsf) を実現し、イーサネット上の実データを NS-2 上のシミュレーションモジュールで操作できるようにした。図4中の sendobj オブジェクトでは、簡単のため、実行する OS に合わせて、a.out 形式の実行ファイルデータを転送することとした。

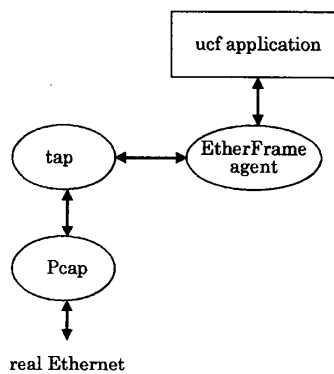


図3 NS-2エミュレータ上の構成

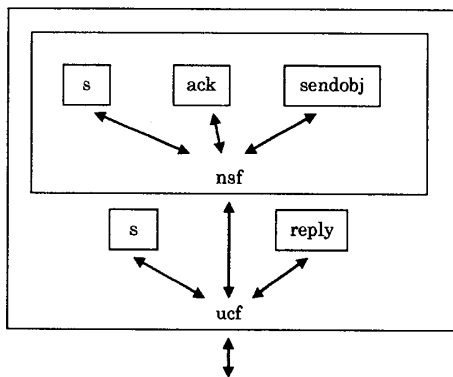


図4 NS-2アプリケーション内部の構造

5. UCF 通信実験

作成した通信オブジェクト間のUCF通信を図5の構成で

```
reth: <ucf> <s> reth </s> <reply> <s> found </s> nsf </reply> </ucf>
nsf: <ucf> <reth> <s> nsf </s> <found> nsf </found> </reth> </ucf>
reth: <ucf> <nsf> <s> reth </s> <sendobj> audioexe </sendobj> </nsf> </ucf>
nsf: <ucf> <reth> <def> <name> audioexe </name> <a.out> ... </a.out> </def> </reth> </ucf>
nsf: <ucf> <reth> <audioexe> sound data </audioexe> </reth> </ucf>
```

図6 オブジェクト間通信データ

確認した。図6は通信データを示し、左のPC上のrethオブジェクトと右のPCのnsfオブジェクトが通信し、(1)rethがnsfオブジェクトを探し、(2)nsfがrethに返事をする、(3)rethがnsfにaudioexeオブジェクトを要求する、(4)nsfがaudioexeを送信する、(5)nsfがaudioexeにデータを送信するという過程が実現されている。

実際には、1回の送信データは256バイトに制限し、前述のフロー制御を用いた。また、送信ごとにcrcによるエラーチェックを行っている。タイミング関連部分は省略したため、ネットワークの遅延時間が短か過ぎるとオブジェクト起動前にデータが到着し、フリーズするということがあるが、基本動作は確認され、受け取った音声データをスピーカで再生するところまで実行できた。

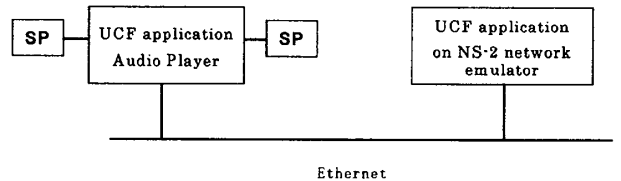


図5 UCFによるオブジェクト通信実現例

6. まとめ

イーサネットフレームを直接使い、UCFの通信を実現した。オブジェクトの移動、動的配置も行った。基本動作の確認だけ行ったが、オブジェクトを追加していくことで、処理の充実や効率化、高度化をはかれる仕組みができた。また、NS-2でもUCFの通信を実現したことにより、シミュレータコードとしてのプロトタイプからスタートでき、システム開発が段階的に行うことができる。

参考文献

- [1] 平中幸雄, 菅井栄治, 渡部修平, 武田利浩, 汎用通信フォーマットによる移動型オブジェクトの実現, FIT2005(第4回情報科学技術フォーラム), 第4分冊 293-294 M-038, 2005.9.9
- [2] 渡部良平, 渡邊高成, 武田利浩, 平中幸雄, 通信路を問わない汎用通信フォーマットの提案と何でもリモコンの作成, FIT2006(第5回情報科学技術フォーラム), 第4分冊 81-82, L-033, 2006.9.7
- [3] Yukio Hiranaka, Hitoshi Sakakibara and Toshihiro Taketa, Universal Communication Format for Multimedia Data, Proc. Sixth International Conference on Computational Intelligence and Multimedia Applications (ICCI 2005), 338-339, 2005.8.16-18.
- [4] The Network Simulator - NS-2, www.isi.edu/nsnam/ms/