

UCF を使用する基本通信オブジェクトの開発 Development of basic communication objects using UCF

渡邊 正裕[†] 田名部 創大[‡] 武田 利浩[†] 平中 幸雄[†]
Masahiro Watanabe Souta Tanabe Toshihiro Taketa Yukio Hiranaka

1. はじめに

著者らはこれまで、様々なオブジェクト間の通信を一般化して表現し、通信のシミュレーションや一般化通信プログラムを作成する方法として、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[1-4]、イーサネットフレーム上に直接 UCF データを載せ通信することにより、より現実的な通信を目指してきた[1]。

これまでの研究から、実際の通信において、通信オブジェクトが共通に保持していることが期待される機能がいくつかあることがわかってきた。

そこで、本稿では UCF に関する規定を検討し、通信の基本となる機能を実装した基本通信オブジェクトの規定作成と実証実験を行い、その結果を報告する。

2. UCF の目的と概要

UCF の最終的な目標は、様々な通信オブジェクトが相互通信可能な状況を作ることである。そのために、様々な通信を表現することができる、統一的なデータ形式として UCF を定義する。UCF 自体は合目的な機能を持つものではなく、実際の通信プロトコルなどで問題があるとする、その問題も表すことができる表現と考える。UCF で表現されたデータをやり取りするものを UCF オブジェクトと呼ぶ。

UCF であらゆる通信を表現し、その UCF 表現されたデータが流通する通信網を実際に構成することもできる。その通信網で、双方向性の応用や、クロスレイヤー通信など従来の通信網にない機能を実現することができる。

通信の最小限の表現は、宛先指定とデータの組で、この組の表現を単一化すれば、通信の統一的表現が実現する。具体的には、次のように全て ASCII 文字で表し、宛先を表すオブジェクト名をタグにし、そのタグでデータを挟む形で通信全体を表す。

`<OBJ>MSG</OBJ>`

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。

`<OBJ1>MSG1<OBJ2>MSG2</OBJ2></OBJ1>`

エラー制御、フロー制御などのプロトコル機能も、それを実現するオブジェクトへの通信の形で実現することができる。

UCF では汎用性を高めるため、規定を最小限に止める。オブジェクト名の最大長はオブジェクトが決めるものとするが、必要最小限が望ましい。また、1つの UCF オブジェクトは、複数の UCF オブジェクト名を所持できる。

[†] 山形大学 Yamagata University

[‡] 株式会社エイチ・アイ・ディ HID Co.,Ltd.

これは、既存の通信プロトコルを表現するためである。

3. UCFSO

UCF を用いた通信において一般的に必要とされる機能を備えた UCF オブジェクトを Universal Communication Format Standard Object (略称 UCFSO) として定義する。具体的には以下の6つが代表的なものである。

(1) 中継処理

タグで囲まれた部分はそのタグが表すオブジェクトへ中継する。また、中継する場合は、次の s を用いて自分が中継したことをデータ中に埋め込む。

(2) s

返信先のデータを処理すべきオブジェクトである。s は送信元や返信時の中継オブジェクトを表すときに使うが、その値を処理するのが s という名前のオブジェクトであるとして形式化することができる。

(3) ucf

全ての UCF オブジェクトは <ucf> も自分宛として処理する。この UCFSO は伝送路上のブロードキャストと同じ意味になる。

(4) def

送信されてきた Java ソースなどをオブジェクトの定義として受け入れるためのオブジェクトである。

(5) v

UCF タグを含むデータを単なるデータをして処理するためのダミーオブジェクトである。

(6) rep

一般応答用オブジェクトで、オブジェクト機能などの問い合わせを受け付け、返事をするオブジェクトである。

4. 基本通信オブジェクトの開発・実験

本稿で定義した規定に則って、通信の基本となる応答オブジェクトを開発し、通信実験を行うことによって、規定が実現可能であるかどうか検証する。

FreeBSD6.1R 上に BPF(Berkeley Packet Filter)を利用して図1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。con オブジェクトは UCF の入出力を行うオブジェクトであり、UCF データを送信し、返信されてきたデータを処理・表示するオブジェクトである。また、reth オブジェクトは中継オブジェクトであり、イーサフレームの送受信を行うオブジェクトである。

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2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1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。UCFSO である rep オブジェクトは UCF データの内容

に応じた処理を行い、結果を送信元に返信するオブジェクトである。機能としては時刻の返答、オブジェクト名の返答機能を実装した。

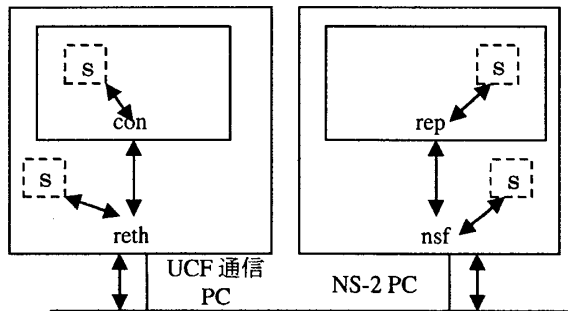


図1 アプリケーション内部の構造

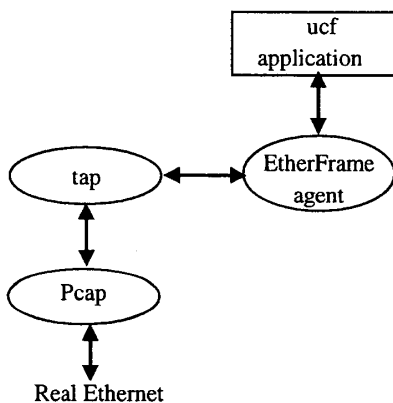


図2 NS-2 エミュレータ上の構成

5. 通信実験

作成した通信オブジェクト間の UCF 通信を図 3 の構成で確認した。図 4 は通信データを示し、(1)左の UCF 通信 PC 上の con オブジェクトから UCF データが入力送信され、reth へ渡され、(2)受け取った reth がイーサフレームを作成し、NS-2 PC 上の nsf へ送信し、(3)nsf はイーサフレームを受信し、rep 宛のデータを作成し送信する、(4)rep は適切な処理をし、返信 UCF データを作成し、reth へ送信し、(5)reth はイーサフレームを受け取り、con 宛の UCF データを作成し送信する。(6)con は受信すると、処理をし、結果を出力するという過程が実現されている。通信の基本となる応答の機能の実現できた。

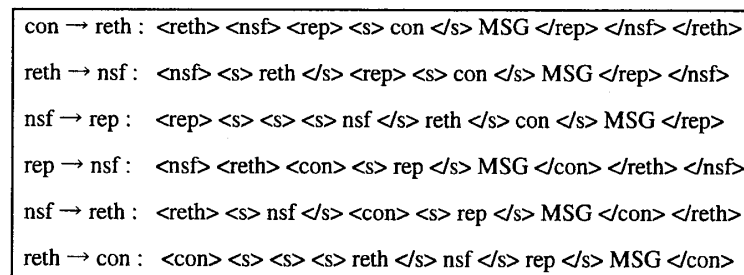


図4 オブジェクト間通信データ

各オブジェクトも想定通りの処理を行い、想定通りの UCF データをやり取りしていたことを確認した。

また、それぞれのオブジェクト名の代わりに UCFSO である ucf 宛のデータを用いて通信実験を行っても応答機能が実現できた。

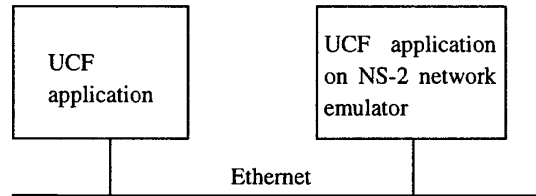


図3 オブジェクト間通信実現例

6. まとめ・課題

UCF の規定を検討し UCF オブジェクトの基本となる UCFSO の機能を検討し実装実験を行った。加えて、NS-2 でも UCF 通信を実現したことにより、機器に依存することなく、UCF 通信が実現できることが実証された。

実ネットワークにおける利用のため、UCF の規定、UCF オブジェクトの規定、並びに UCFSO の規定については今後も検討していく必要がある。また、様々な通信を表現するために、適時規定に変更を加えていく他、通信の基本要素から徐々に複雑な機能へと拡張し、実現していくためのオブジェクトの開発を進めていくことが望まれる。

参考文献

- [1] 平中 幸雄, 伊藤 大視, 市川 直樹, 武田 利浩, “イーサネットフレームによるダイレクト UCF 通信”, FIT2007(第6回情報科学技術フォーラム), 第4分冊 23-24, L-011(2007).
- [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).
- [3] 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).
- [4] Yukio Hiranaka, Hitoshi Sakakibara, Toshihiro Taketa, “Universal Communication Format for Multimedia Data”, Proc. Sixth International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2005), 338-339(2005).