

## 遠隔監視機能を持つホームサーバの設計とプロトタイプの開発

新谷 和司 河野 英太郎† 前田 香織† 天野 橋太郎‡

E-mail : kazu@nets.ce.hiroshima-cu.ac.jp

広島市立大学大学院 情報科学研究科  
〒731-3194 広島市安佐南区大塚東 3-4-1  
†広島市立大学 情報処理センター  
‡広島市立大学 情報科学部

## 概要

家電のネットワーク対応化が進み、USB、Bluetooth、IEEE1394 など新しい通信規格が提案され、ホームサーバを含むホームネットワーク構築の環境が整いつつある。ホームサーバに遠隔監視機能等を導入し、遠隔から家庭内やビル等の監視や制御を手軽に行うことが検討されている。本研究では遠隔監視を遠隔会議などのマルチメディア通信の一つの形態と考え、その観点から特徴をまとめる。それにより、遠隔地の映像だけでなく、様々な状態変化の取得を行う遠隔監視システムを設計できることを示す。遠隔監視システムの一例として、制御可能な USB カメラと Linux サーバを用いたプロトタイプを開発し、性能評価を行った。また、本システムで実現した遠隔 Plug & Play 機能を UPnP、Jini と比較する。

## A Design of a Home-server with Remote Monitoring and Development of its Prototype

Kazushi SHINTANI Eitaro KOHNO Kaori MAEDA Kitsutarō AMANO

Graduate School of Information Sciences, Hiroshima City University  
3-4-1 Ozuka-Higashi, Asa-minami, Hiroshima, 731-3194, Japan

## Abstract

By the spreading of home appliance with networks and technical standards such as USB, Bluetooth and IEEE1394, it is becoming easy to develop home networks having home servers. Therefore, we can observe and control remote devices. In our study, we consider remote monitoring as one of multimedia communication like distance lectures, and discuss its feature. We show a design of remote monitoring system with not only remote video but also various status (temperature, connectivity of appliances and so on). Also, we develop and evaluate its prototype system. That consists of a observational client and a server having a controllable USB camera on Linux. Finally, we compare plug-and-play feature for remote devices by our system with UPnP and Jini.

## 1. はじめに

近年、一般家庭内にある家電機器を相互にネットワーク接続し活用する「ホームネットワーク」により、機器相互の遠隔操作や遠隔監視機能を実現する方法が提案されている。将来は我々の家庭で使われる多くの家電機器がネッ

トワーク対応することにより家電以外のネットワーク機器との相互接続が可能になると考えられる。

現在ホームネットワークを形成するための一方法として、家庭内と外界との間にゲートウェイを設け、家電機器が対外的に通信を行う際にはすべてこのゲートウェイを通過させる方

式がある。このゲートウェイを「ホームサーバ」と呼ぶ。このような構成をとる利点としては、ホームネットワークを構成する機器の混在する通信規格の差異を吸収できることなどが挙げられる。ホームネットワークを形成する家電機器の通信規格として Bluetooth[1], IEEE1394[2], USB (Universal Serial Bus) [3]などがあるが、これらの特徴の一つとして、ユーザが機器同士の接続の際、詳細な設定を行わなくても使用可能になる「Plug & Play」機能がある。

このような背景より、今後家電機器を遠隔から利用する場面が増えると予想される。既に遠隔からの機器状態の監視例として、電子ポット[4]の状態監視による家庭内構成員の生存確認などがある。しかし、これらの実装では機器固有の通信規格が使われていることが多く、使用範囲が制限される。

そこで、本稿では機器に依存しない監視システムを提案する。また、遠隔にある機器の Plug & Play の機能を容易に実現できるような仕組みを提案し、それを遠隔 Plug & Play 機能と呼ぶ。この遠隔 Plug & Play 機能を積極的に活用し、家電機器などの遠隔監視を行うためのホームサーバの設計について述べる。また、通信規格として USB 機器を対象とするプロトタイプを開発した。このプロトタイプの実装と簡単な評価結果についても述べる。

## 2. 遠隔監視の特徴

遠隔監視では監視対象の状態が変化した場合に監視者に通知をすることが重要な役割である。監視者への状態通知は監視機器からの能動的な通知である。状態の通知方法としては文字情報や動画像、静止画像、音声などがある。

文字情報による通知は機器状態をあらかじめ定義した文字列に変換して通知を行う。この方法は例えば SNMP (Simple Network Management Protocol)[5]によるネットワークに接続された機器の監視において用いられている。動画像や音声は監視機器が周辺情報を取得する際に用いることができる。例えば、ルータやスイッチなどネットワーク機器を監視する場合、SNMP などによる機器内部の温度異常が検出された場合、画像を見て監視者が火災などの周辺状況を判断できる。

このように、遠隔監視にマルチメディア情報を用いることで、より多角的な判断ができる。

そこで本研究では遠隔監視もマルチメディア通信の一つと考えることとする。マルチメディア通信の代表的なアプリケーションとして遠隔会議があるが、テレビ会議用のシステムが遠隔監視のシステムとして使用可能かを検討するため表 1 のように遠隔会議と遠隔監視を比較した。

遠隔監視と遠隔会議で取り扱うメディアは共に同じだが、遠隔会議は双方向でメディアを転送するので、転送データ量は対称になる。遠隔監視は主に監視対象からメディアを転送し、監視者からは監視の設定変更など比較的少ないデータ量で済む。よって、遠隔監視のデータ転送量は非対称になるケースが多い。また、データの流れ方についても違いが見られる。遠隔会議では会議開催時には絶え間なくデータ転送が行われているが、遠隔監視は監視を行っていてもデータ転送が起こっていない場合がある。したがって、一般的には不定期の通信であると言える。

遠隔監視と遠隔会議で最も違う点は転送されるデータに優先度が異なるデータがあるかないかである。遠隔監視では異常を検出すると監視者に異常の通知を行う。異常を通知するときには通常の機器状態通知よりも優先度を上げて通知を行う必要がある。しかし、遠隔会議にはネットワーク上で転送されるデータには種類による優先度はあまり無い。

表 1：遠隔監視と遠隔会議の比較

	遠隔監視	遠隔会議
メディアの種類	文字データ 動(静止)画像 音声	資料データ 動(静止)画像 音声
転送データ量	非対称	対称
データの流れ方	不定期	定期的
異常を示すデータ	あり	なし

異常を示すデータを転送する遠隔監視システムでは異常通知を実現するために、監視対象や周辺の情報を取得する。取得した情報の解析を行い、正常か異常かを判断する。正常か異常かを判断する閾値は監視対象によって異なるため、監視者は閾値を設定する機能が必要となる。また、遠隔監視ではホームサーバに多くの監視機器が接続される可能性があるが、遠隔 Plug & Play 機能を実装することで、監視機器が増加、減少した場合に監視者が容易に対応できるようにする。

このように、遠隔監視に用いるマルチメディア伝送システムとして一般的なテレビ会議システムでは対応できない点がある。本研究ではここで指摘したテレビ会議システムにはない「状態変化の通知」の機能に焦点をあて、3章以降でこの機能を実現するプロトタイプシステムについて述べる。

### 3. 遠隔監視機能を持つホームサーバシステム

#### 3.1 プロトタイプシステムの設計と概要

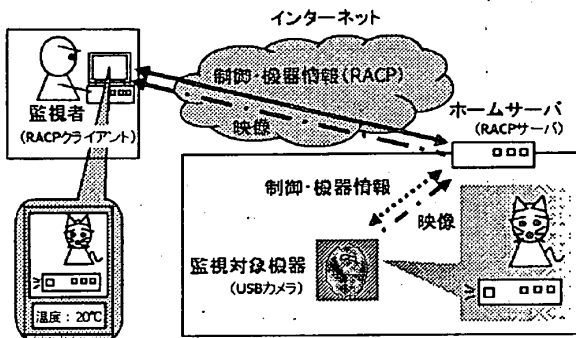


図1：システムの概要

プロトタイプシステムで状態変化の通知は

- ・ 異常通知機能
- ・ 遠隔 Plug & Play 機能

として実現する。プロトタイプシステムの概要を図1に示す。ホームサーバはUSBカメラの接続状況や状態変化を通して、監視対象の正常、異常の判断を行う。異常と判断すると監視者（クライアント）に通知する。このとき、ホームサーバはUSBカメラを管理し、USBカメラ[6]とインターネットの接続を仲介する。ホームサーバとUSBカメラはUSBプロトコルにより通信を行う。インターネット上の通信はRACP (Remote Appliance Control Protocol) [7]を用いて行う。本システムではUSBカメラの状態変化を通知するだけでなく、ホームサーバ自身の状態変化としてUSBカメラの接続や切断をクライアントへ通知する。これを遠隔 Plug & Play 機能とした。異常の通知にはRACPのTRAP機能を再定義し利用する。

状態通知以外の機能として、本システムでは

- ・ 遠隔機器制御機能
- ・ 監視操作ユーザインタフェースの統一を実装した。遠隔機器制御はRACPを用いて

開発した遠隔カメラ制御[8]の手法を用いた。また、監視者に統一したユーザインタフェースを提供し、監視機種に依存することなく監視操作ができるようにした。

#### 3.2 RACPの概要

RACPは機器を制御するためのプロトコルとして設計されており、TRAPという機器の状態変化をクライアントに通知する仕組みを持つ。RACPクライアントはRACPサーバから送信されるTRAPにより遠隔機器の状態変化を取得する。

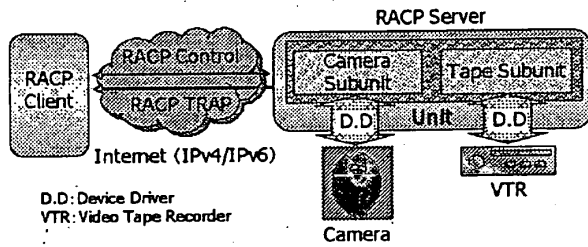


図2：RACPの概要

RACPのモデルを図2に示す。RACPはサーバ内部にユニットを持ち、ユニット内部に制御対象機器を抽象化した複数のサブユニットを持っている。RACPの制御はTCP (Transmission Control Protocol)とASCII文字列を用いて行い、サブユニットに対してRACPで独自に定義したコマンド（表2）を発行する。このコマンドに機器固有の情報は含まない。機器依存部分には専用のデバイスドライバを別途用意し、これを用いて機器制御を行う。そのためRACPは機器に依存しない制御が可能となる。RACPで機器の状態変化、すなわちユニットの状態変化が起こった場合にはTRAPを用いてRACPクライアントに通知する。制御対象機器から通知されるTRAPメッセージはUDP (User Datagram Protocol)で図3のパケットフォーマットに従って送られる。図3のコードのうち本システムに関連のあるもののみを表3に示す。TRAPをクライアントに発行するための条件設定はRACPコマンドの一種であるTRAPコマンド（表2）で行う。

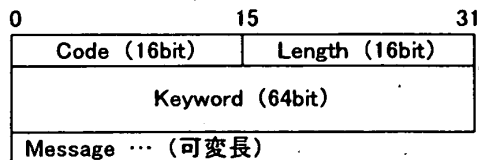


図3：TRAPパケットフォーマット

表 2 : RACP コマンドの一例

Subunit	Description	Command Syntax
<i>Unit Commands:</i>		
	USER NAME	USER <i>user-name</i>
	PASSWORD	PASS <i>password</i>
	LOGOUT	QUIT
	SET TRAP	TRAP <i>keyword</i> [[ <i>h1</i> ,..., <i>h4</i> , <i>p1</i> , <i>p2</i>   <i>h1</i> ,..., <i>h16</i> , <i>p1</i> , <i>p2</i>   <i>host-name port</i> ]]
	SHOW STATUS	STAT [ <i>subunit-name</i> [ <i>subunit-num</i> ]]
	SHOW HELP	HELP [ <i>subunit-name</i> [ <i>subunit-num</i> ]]
<i>Subunit Commands:</i>		
CAMERA	SET HORIZONTAL	CAMERA <i>n</i> HORI <i>horizontal-position</i>
	SET VERTICAL	CAMERA <i>n</i> VERT <i>vertical-position</i>
	SET TRAP	CAMERA <i>n</i> TRAP {ACTIVATE   STOP}
	SHOW CAMERA STATUS	CAMERA <i>n</i> STAT
THERMO	SET TRAP	THERMO <i>n</i> TRAP RANGE <i>threshold threshold</i>
	SHOW THERMOMETER STATUS	THERMO <i>n</i> STAT

表 3 : TRAP のコード (一部)

Code	Keyword	Message
21	ACTIVATE	< <i>Subunit_name</i> > Subunit was activated
22	STOP	< <i>Subunit_name</i> > Subunit was stopped
23	RANGE	Temperature is out of range. Temperature is <i>temperature</i> degree Celsius.

表 4 : プロトタイプシステムの仕様

ホームサーバ(サーバ)	
CPU	Celeron 300 MHz
メモリ	192 MB
OS	Red Hat Linux 7.3 (kernel-2.4.18)
監視者端末(クライアント)	
CPU	Celeron 300 MHz
メモリ	192 MB
OS	Red Hat Linux 7.1J (kernel-2.4.2)
監視機器(USBカメラ)	
キーエンス社 ROUND VIEW (USB 1.1)	

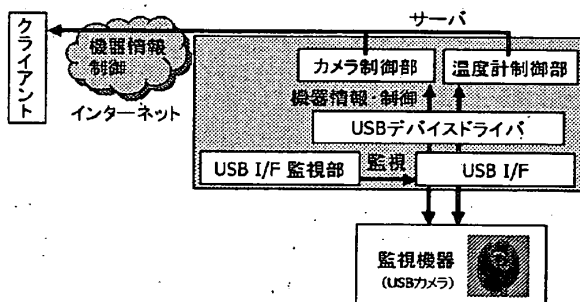


図 4 : ホームサーバ内部の構成

### 3. 3 実装

プロトタイプシステムの仕様を表 4 に示す。開発は C 言語を用いた。プロトタイプシステムに用いた USB カメラはマイクが内蔵されているため、周辺情報として映像と共に音声の取得が可能である。また、温度センサーが装備され、周辺温度が監視できる。USB 機器の接続状態の監視も可能である。ホームサーバ内部の構成を図 4 に示し、それぞれの機能について述べる。

#### 1) 遠隔 Plug & Play 機能

ホームサーバは内部に USB インタフェースを監視する部分を持っている。ホームサーバに USB 機器が接続されると、まず OS がその USB

機器に対応したデバイスドライバをロードする。次に USB インタフェース監視部は USB 機器の接続を感知し、インターネットと USB デバイスを橋渡しとなる USB 機器制御部を生成し、クライアントに機器が接続されたことを通知する。USB 機器がいくつかの機能を持っている場合には機能ごとに制御部を生成する。本システムで用いた USB カメラはカメラの他に温度計の機能を持ち合わせているため、カメラ制御部と温度計制御部を生成し、クライアントにはカメラと温度計が接続されたとして通知を行う。従って、クライアントからホームサーバの機器の接続情報を取得すると二つの機器が接続しているように見える。

逆に USB 機器が切断されると、同様に USB インタフェース監視部が感知し、カメラ制御部と温度計制御部の削除の指令を行う。カメラ制御部と温度計制御部はそれぞれクライアントに対して切断されることを通知し、消滅する。接続と切断の通知はそれぞれ図 3 のパケットフォーマットで表 3 の Code (21, 22) を用

いて行う。

## 2) 「異常の条件」設定機能

本システムにおいて異常の条件を設定するにはクライアントからホームサーバ内部の温度計制御部に対し条件の設定を行う。設定はRACPで定義した制御コマンド(表2)のうちのTRAPコマンドを用いて正常と判断する範囲を入力する。

## 3) 機器状態変化通知機能

温度計制御部に異常の条件を設定した後、温度に異常が起こったと判断された場合には、温度計制御部からクライアントに対して図3のパケットフォーマットで表3のCode(23)を用いて通知を行う。

## 4) 遠隔機器制御機能

本システムではカメラの遠隔制御を可能にすることで広範囲の監視が行えるようにする。カメラの制御はクライアントがカメラ制御部に対しRACPで定義した制御コマンド(表2)を用いて行う。カメラ制御部ではUSBデバイスドライバを用いてUSBカメラを制御する。

## 3. 4 測定と評価

図5に示す構成でシステムのオーバーヘッドを測定した。

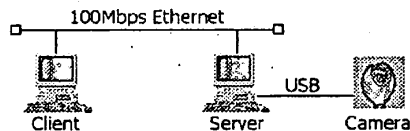


図5: 実験のネットワーク構成

### 1) 遠隔 Plug & Play 処理時間の測定

測定方法は、USBカメラとホームサーバ、ホームサーバのモニタ、クライアントのモニタを並べてデジタルビデオ(DV)で撮影しながら、ホームサーバに接続されているUSBカメラを抜き差しし、それを50回行った。録画された映像をコマ送りし、USBカメラがホームサーバに接続されてからそれぞれのモニタに文字列が表示されるまでのフレーム差(30frame/sec)からオーバーヘッドを測定した(図6の①, ②)。

表4に示した結果から、ホームサーバにUSBカメラのデバイスドライバがロードされてからクライアントに通知するまでの時間は10msecとなる。その結果、遠隔 Plug & Play 処理全体のうちRACP TRAPが占めるオーバヘ

ッドの割合は5.37%(9.84msec)となる。

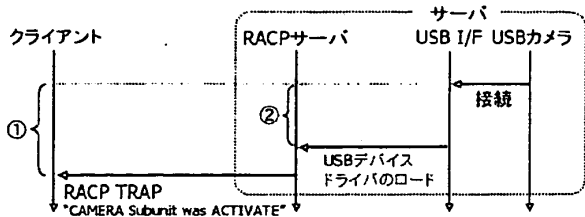


図6: 遠隔 Plug & Play 処理時間の測定

表4: 遠隔 Plug & Play 処理時間の測定結果

	時間(msec)	差分(msec)
①	183.3	10.0
②	173.3	

### 2) 異常通知時間の測定

ホームサーバがUSBカメラから温度情報を取得し、温度が異常であると判断し、クライアントに異常を通知するまでにかかった時間を測定した。結果を表5に示す。表5の差分はホームサーバが異常を検知してからクライアントに異常を通知するまでの時間となり、RACPによるオーバーヘッドは処理全体の約0.2%であった。

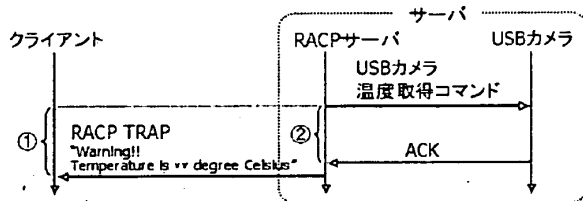


図7: 異常通知時間の測定

表5: 異常通知時間の測定結果

	時間(msec)	差分(msec)
①	72.03	0.16
②	71.87	

### 3) 遠隔カメラ制御時間の測定

クライアントでカメラの水平位置を"0"に設定するRACP制御コマンド"HORI 0"を送信し、クライアントにACKが返ってくるまでの時間を測定した。表6の差分は0.55msecとなり、これはネットワーク遅延とRACP制御のオーバーヘッドとなる。このときのネットワーク遅延は0.16msecであり、遠隔カメラ制御時間のうちRACPが占めるオーバーヘッドの割合は0.11%(0.39msec)となる。

以上三つの測定結果より、本システムにおいてUSBカメラを遠隔地から利用する場合とロ

一カルに利用する場合でほとんど差が無いことを確認できた。そのため、遅延に関しては実用上問題ないと考えることができる。

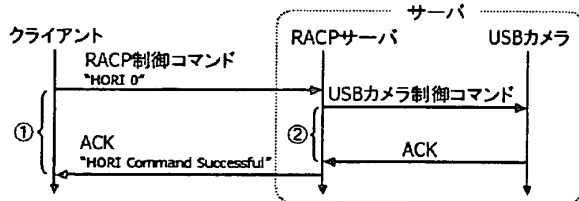


図 8：遠隔カメラ制御時間の測定

表 6：遠隔カメラ制御時間の測定結果

	時間(msec)	差分(msec)
①	338.24	0.55
②	337.70	

#### 4) 周辺温度の測定による異常状態検出

本システムを寒暖の差がある場所に設置し、運用した。気温の上昇や下降により異常を検出し、監視者へ通知することを確認した。

### 3. 5 他の提案方式を用いた状態変化の通知方法に関する考察

情報家電機器のための通信ミドルウェアの代表的なものとして UPnP (Universal Plug and Play) [9]や Jini[10]があるが、この二つは遠隔監視に必要な状態変化の通知機能を持つだろうか。

Jiniは本稿での提案システムと同様にホームサーバに相当する LUS (Look Up Service) を導入し、それを經由して、監視者と監視対象が通信する。現在の仕様では監視対象の状態変化の通知を実現するためには監視者は能動的に LUS へ監視対象の情報を問い合わせることになる。

UPnPには HTTP(HyperText Transfer Protocol) や SSDP (Simple Service Discovery Protocol) [11]を用いることにより、本稿で定義した遠隔 Plug & Play 機能は実現できる。また、機器で発見したイベントの処理も可能で、UPnPで状態変化の通知はできる。ただし、本稿で想定しているホームサーバを經由するタイプではなく、監視側と監視対象側に TCP/IP と HTTP を実装して、相互に直接通信する。

## 4. まとめ

本稿では遠隔監視をマルチメディア通信の一種として捉え、遠隔監視の特徴を述べた。そ

して、遠隔監視機能を持ったホームサーバを設計し、そのプロトタイプシステムの実装と評価を行った。

現在のプロトタイプシステムは温度による監視のみが可能である。今後は、映像による監視が可能になるように Linux 上で動作する USB カメラのデバイスドライバを開発する。本システムを実際の監視に用いて監視の正確性などの評価を行う。さらに、状態変化通知方法の似ている UPnP との詳細な比較も行う。

## 謝辞

本研究の一部は広島市立大学特定研究費(平成 14 年度 2206) の支援を受けて実施されている。ここに記して謝辞を記す。

## 参考文献

- [1] Bluetooth Web site, <http://www.bluetooth.com/>
- [2] IEEE1394 Web site, <http://www.1394ta.org/>
- [3] USB Web site, <http://www.usb.org/>
- [4] みまもりほっとライン, <http://www.mimamori.net/>
- [5] J.D. Case, M. Fedor, M.L. Schoffstall, C. Davin. "Simple Network Management Protocol (SNMP)", Request for Comments 1157, May. 1990.
- [6] 株式会社キーエンス, MOVING CAMERA ROUND VIEW, <http://www.keyence.co.jp/hobby/roundview/mc-1000/index.html>
- [7] 西村 浩二, 前田 香織, 相原 玲二, "遠隔機器制御プロトコル RACP のフレームワークとその応用", 情報処理学会論文誌, Vol.42, No.12, pp.2869-2877, Dec. 2001.
- [8] 西村 浩二, 前田 香織, 河野 英太郎, 相原 玲二, "機器制御プロトコル CRCP を利用した遠隔会議装置制御システム"情報処理学会研究報告, 99-DSM-15, pp. 61-66 (1999)
- [9] Universal Plug and Play Web site, <http://www.upnp.org/>
- [10] Jini Web site, <http://www.jini.org/>
- [11] T. Cai, P. Leach, Y. Gu, S. Albright, "Simple Service Discovery Protocol/1.0", draft-cai-ssdp-v1-03.txt, April. 2000.