A three-tier, tTCP/IP gateway architecture to link home networks with the internet.

Title	A three-tier, tTCP/IP gateway architecture to link home networks with the internet.
Author(s)	Corcoran, Peter M.
Publication Date	1998-06-04
Publisher	IEEE

WPM 14.2

A Three-Tier, TCP/IP Gateway Architecture to link Home Networks with the Internet.

Peter M. Corcoran.

Dept. of Electronic Engineering, University College, Galway

Abstract - ***A three-tier gateway architecture for internetworking between home automation networks and a TCP/IP based wide area network, such as the internet, is described. The architecture abstracts the functionality of any home network into a driver layer (tier one), and provides a common access layer (tier two) from any TCP/IP network application (tier three) to a local home automation network. Clients and application programs may transparently access services and resources on the home network and appliances connected to the home network may access resources and services on the TCP/IP network.***

1. Introduction

A unified gateway architecture for home automation networks is described. The purpose of this architecture is to provide, a common access layer from any TCP/IP network to a local home automation network. Thus remote application clients may access services and resources on the home network, regardless of the underlying home network. In the same way, devices and appliances connected to the home network may access resources and services on the TCP/IP network.

The work described in this paper is partly inspired by the recent Home Plug 'n Play (HPnP) initiative [1]. HPnP is intended to enhance the existing US industry standard for the Consumer Electronics Bus (CEBus) as described in IS-60 [2]. HPnP is an initiative which seeks to integrate CEBus with other existing and emerging Home Bus standards, including IEEE 1394, LonWorks, European Home Systems (EHS) and IrDA.

Essentially HPnP adopts the Common Application Language (CAL) as the unifying application layer protocol. The CAL product model and the standard industry contexts of CAL thus form much of the core of our work to integrate generic home automation networks with TCP/IP networks.

The gateway architecture described in this paper employs a three-tier model, rather than a simpler client-server implementation. At the physical interface between the gateway and the home automation network a "driver-agent" - *SIOd* - monitors and interprets the network traffic, performs real-time event handling, packet acknowledgements and filtering tasks. The raw traffic from the home network is then passed on to a "broker-agent" - *CALNetd* - who is responsible for translating and interpreting the network messages and maintaining a dynamic "state-model" of the real home network. *CALNetd* is also responsible for interacting with TCP/IP based client applications and for generating events and placing messages on to the home network via *SIOd*.

2. The Gateway Architecture

In Fig 1 the relationships between the system software components of the home gateway, the local home network and the wide area network is illustrated. In particular we note that the software components, *SIOd*, *CALNetd* and any system *UIDevs/HiPlets* [3,4] all communicate with one another via network sockets. In this configuration, which is employed in our prototype gateway, all of the main components run on the same hardware unit - in our case a standard PC. Applications may run on other *Application Clients*, but all of the key system software components run, as independent *daemon* processes on a single PC.

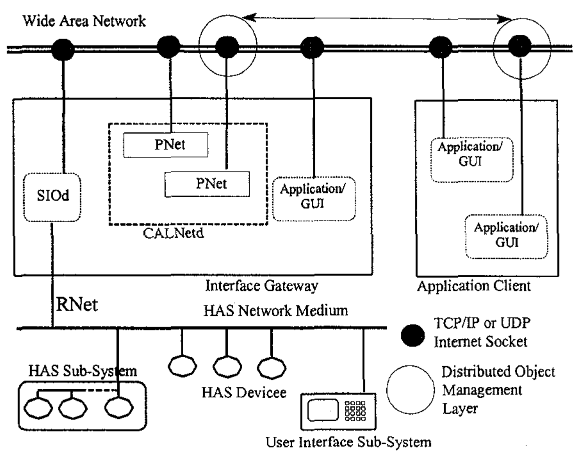

Fig 1: *Diagramatic representation of the relationships between system software components, the Wide Area Network and the Home Network.*

In fact, this is not a requirement for the architecture we have described and it is interesting to consider the software architecture in more general terms, particularly the potential for separating the system components physically and allowing them to run on separate hardware units, but all connected to the same WAN infrastructure. This is illustrated in Fig 2 where the *SIOd* runs on a separate hardware unit from the

0-7803-4357-3/98/ $10.00 © 1998 IEEE

CALNetd, which runs on an independent *Routing Host*. Application programs, UIDevs and HiPlets run on a third, *User Interface Unit*. Such an arrangement is practical because all of these software components communicate with one another over network sockets.

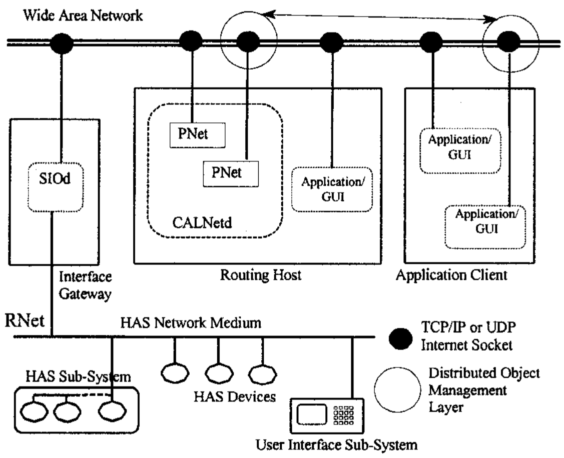

Fig 2: *As for Fig 1, but the SIOd and CALNetd agents now run on separate, independent hardware entities..*

3. The "Driver-Agent" - *SIOd*

This consists of two main threads for Tx and Rx of packets onto the Home Network. However these threads are not fully independent and may interact via a *Proxy Services* filter. This handles redirection and real-time response generation using a table of low-level device services, for example the IACK or immediate-acknowledge response of CEBus.

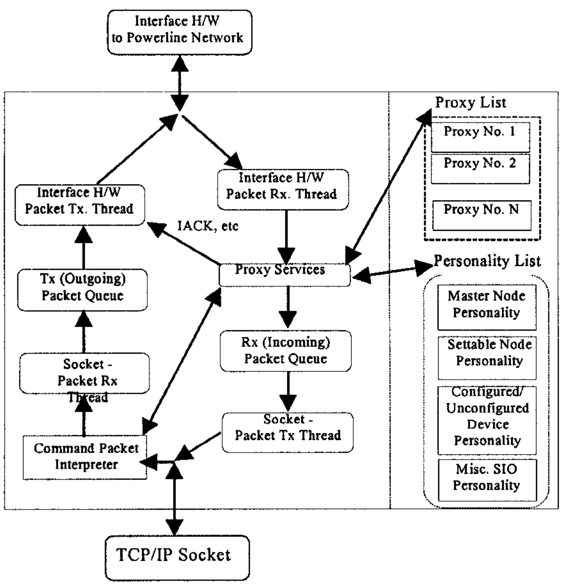

Fig 3: *Structure & Organization of SIOd "Driver Agent".*

4. The "Broker-Agent" - *CALNetd*

This system agent maintains a dynamic model of the present state of the home network. User applications and other tier-three services interact with the virtual model of the HAS contained in *CALNetd*. Amongst the principle services provided by the *CALNetd* broker-agent are those of packet-routing, device-registration, address binding and API/event-translation.

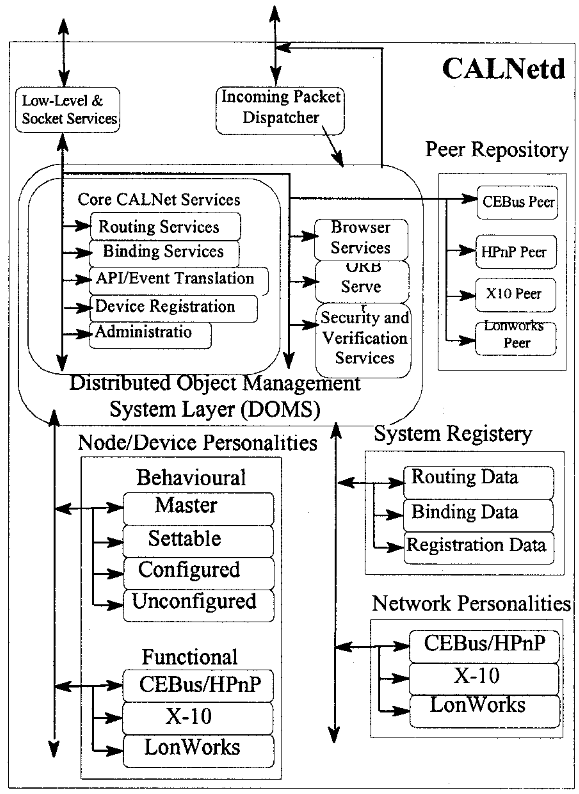

Fig 4: *Internal Structure and Organization of the CALNetd, or "Broker Agent".*

REFERENCES

[1] http://www.cebus.org

[2] EIA Home Automation System (CEBus) Standard IS-60, EIA, June 29th 1992.

[3] Desbonnet, J and Corcoran P. M.., "System Architecture and Implementation of an Internet/CEBus Gateway", *ICCE '97 - IEEE Conference on Consumer Electronics, Chicago Il*, Jun. 1997.

[4] Corcoran P. M. and Desbonnet, " Browser Style Interfaces to a Home Automation Network", *ICCE '97 - IEEE Conference on Consumer Electronics, Chicago Il*, Jun. 1997.