A SYSTEM AND METHOD FOR SUPPORTING MULTIPLE VENDOR TELEPHONY HARDWARE ON A COMPUTING PLATFORM

Technical field of the invention

The present invention is directed, in general, to computer networks and telephone systems, more specifically, to a system and method for managing multiple vendor resources within a computer platform. This invention includes connectivity to PBX, PSTN and Packet-Based voice networks. This invention is directed to more than the hardware interface to the network elements, specifically the present invention is directed to a system and method for integration hardware capable of accessing multiple network types.

Background of the invention

Historically, entirely separate computer platforms have been employed to connect PSTN, PBX and Packet-Based voice networks together. Each platform would have specific knowledge of a single type of voice network and would be connected to each other using analog telephone lines.

Over a century ago, analog telephone networks were developed to carry analog audio signals. As businesses grew, the need for Private Telephone Networks developed requiring the Telephone companies to build the PBX, which is a small version of the phone company's larger central office switch.

The nature of traditional Key Systems, PBXs and Hybrid telephone systems is to make telephone circuits a shared resource. However, traditional switch manufacturers build proprietary platforms that require third parties to integrate new technologies (like packet-based voice) through analog telephone ports off of the PBX. This leads to lack of compatibility between platforms, and this can lead to delays or problems in systems.

Accordingly, there is a need for a system and method which obviates the need for the switch integration which breeds this systematic inefficiency. In particular, there is a need for all switching to be is done inside of the computer platform using multiple vendors and networks, such that there is no need to integrate with external switch platforms.

Summary of the invention

To address the above-discussed deficiencies of the prior work, the present invention provides an effective way to develop computer telephony application on an open computing architecture. It also defines a client-server model in which applications use a collection of objects to monitor and control telephony hardware resources and abstracts implementation details of call processing hardware and switch fabrics to enables telephony applications to be built without vendor dependencies.

In the attainment of the above primary objective, the present invention provides a system and method for supporting multiple vendor telephony hardware on a standard computer platform. This is accomplished by implementing a software module where all switching is performed inside the computer platform which is using multiple vendors and networks in order to remove the need for integration with an external switch platform. In general, the system is comprised of a two major elements, the client and server modules.

The server module comprises server architecture which has a Hardware Abstraction Layer that is allocated and controlled by the Resource Manager. The Resource Manager utilizes the Device Factories to determine the available Hardware Abstraction resources available.

Clients connect to the Resource Manager for the purpose of monitoring and initiating calls. The client module has an architecture which may comprise various elements, but preferably simply consists of an Application Session, which connects to the Resource Manager.

An inbound call will be manipulated according to an executable application which is installed and registered as a component which establishes a session and assumes control over a call through a protocol between the application and a server.

Accordingly, the present system and method abstracts the hardware and switch fabric from the client applications. The system comprises:

(1) Hardware Abstraction Layer, which defines a common interface in which the resource manager can address the hardware without knowledge of the vendor. The hardware abstraction layer also implements vendor specific functionality based on the equipment being utilized within the computing platform. The hardware abstraction layer includes but is not limited to:

- Telephone lines
- Telephone trunks
- Telephone extensions
- Multi-media resources capable of recording and playing to an audio stream
- Fax resources
- Voice Over IP stack

(2) A Device Factory exists for each Hardware Abstraction Layer, which is utilized by the Resource Manager to determine what hardware abstraction components are available on the system and must be initialized.

(3) A Resource Manager describes and manages the relationship of the various hardware abstraction components of the server. The resource manager also offers an external call control interface to applications.

(4) The Telephony Network Module is an abstraction layer that allows the Resource Manager to perform call control functions on resources without having knowledge of the type of the Telephony Network the system is connected to.

(5) An Application Session creates an interaction with the Resource Manager as a mechanism by which an application can control an active operation on resource within the system.

Brief description of the drawings

For a more complete understanding of the present invention and the advantages thereof, reference is now made to the following descriptions taken in conjunction with the accompanying drawings, in which:

Figure 1 Illustrates a computer network and telephone system that forms an environment within which the present invention can operate.

Figure 2 Illustrates a how the client and server components inter-operate to achieve the primary goal.

Figure 3 Illustrates an example computing platform with the specialized telephony hardware installed.

Detailed description of the Preferred Embodiments

Referring initially to Figure 1, illustrates is a computer network that forms an environment within which the present invention can operate. The

network is illustrated as including a telephone instrument such as the Venture™ telephone set made by Nortel Networks Corporation of Brampton, Ontario (Canada) along with a client PC or client laptop PC such as that manufactured by Dell Computers of Austin, Texas (USA) connected to a computer network. Other telephone instruments may be coupled to the PC for the transmission of voice over the computer network. The definition of transmission of voice over the computer network may comprise any of the formats as known in the art of voice transmission. However, the present invention is intended to seamless integrate calls from the PSTN or Packet-Based voice.

The PCs and Laptop computers are connected to the computer network and to the Internet for the purpose of providing a graphical user interface with the capabilities of screening conferencing, transferring, holding and parking telephone calls. Except for the condition where the telephone is connected directly to the PC, the audio traffic is occurring through the PSTN or through the internal telephone system.

The telephony server such as that manufactured by Dell Computers of Austin, Texas (USA) is connected to the telephone instruments and is further connected to the computer network. The telephones are used for the transmission of the audio, and the computer network is used for the transmission of control commands between the PCs and the server.

Figure 2 illustrates the primary software components required in the invention. The Application Session and the Event Session describe the primary components of the client within the client-server architecture. The Application Session uses a distributed component architecture that connects the server using Remote Procedure Calls over a TCP/IP connection. The Application Session sends asynchronous requests to server components in order to control the behavior of telephone calls. The Event Session which opens a TCP/IP port to the server and listens to TCP/IP broadcasts notifying

the client in an asynchronous manor of state changes within the switch. The Resource Manager which is a server component that reads a list of registered Device Factories and creates instances of the devices. The Resource Manager is the primary module in which the client Event Session connects to in order be notified of state changes within resources on the switch.) The Telephony Network is a server component constructed to retain the knowledge of the type of PSTN switch the server is connected to. The Telephony Network is also responsible for performing all the steps necessary to place calls, transfer, conference and end calls required for each type of PSTN network type. For example, a server can be connected to a Centrex switch which requires a flash hook to be performed in order to transfer a call. The Device Factory is a server side component that creates Hardware Abstraction components specific to each unique vendor supported by the server. The Device Factory retains all the intelligence for initializing the board and constructing the supported resources (i.e. fax, record and playback), and Hardware Abstraction Layer which is a server side component that retains all the specialized software for controlling each vendors boards. Each board manufacture uses a specialized protocol for initializing and controlling their hardware. The Hardware Abstraction layer has a common interface which is used by Application Sessions in order to control the hardware. The Hardware Abstraction layer takes commands from the Application Sessions and turns them into vendor specific protocol calls in order to complete the task.

The Application Session connects to the Resource Manager for the purpose of operating and controlling calls. However, the Event Session simply connects to the Resource Manager for the purpose of receiving events related to state changes within the switching platform.

The Resource Manager and Telephony Network combine to provide a software abstraction layer to the Telephony Application clients. The

Device Factory and Hardware Abstraction layer combine to provide a common interface for the Resource Manager, by which they isolate the Resource Manager from the complexity of each different type of telephony hardware.

Figure 3 illustrates a standard Computing Platform assumed within this invention. This illustration has a PRI Board (such as that manufactured by Dialogic Corporation of Parsippany, New Jersey (USA), Extension Board (such as that manufactured by Dialogic Corporation of Parsippany, New Jersey (USA), and Fax Board (such as that manufactured by Dialogic Corporation of Parsippany, New Jersey (USA), provided by different vendors installed into a single computer. The voice bus (such as that manufactured by Dialogic Corporation of Parsippany, New Jersey (USA), connects multiple boards together and is used to pass voice traffic between telephony boards.

A voice bus is a bus independent of the computer controlling the card that allows audio and signaling information to be passed between different voice processing components. As features such as fax integration, voice recognition and text-to-speech (which often are implemented by add-in boards connected via a voice bus) become more common, it is increasingly important that the voice bus be structured this way. Voice bus options may be implemented in accordance with these requirements, but in the preferred embodiment will include Dialogic's PCM Expansion Bus (PEB), Multi-Vendor Integration Protocol (MVIP) and SCBus.

Thus, by way of illustration, is a description of an Inbound call manipulated on the above described hardware. Each application will go through the same basic structure of setup, call handling and shutdown. First, an executable application is installed and registered as a component. The application first establishes a session by requesting an instance of the Resource Manager from the system. It further more registers its interest in events by creating an event sync, which is used as a call back by the

Resource Manager. At this point the application is ready to receive calls. Second, an application assumes control over a call through a command/response protocol between the client application and the server. API functions invoke a server operation on the server and the server responds back with a result asynchronously.

This invention assumes a specialized computing hardware platform. This platform is comprised of "off the shelf" computer telephony switching boards such as those manufactured by Dialogic Corporation of Parsippany, New Jersey (USA), that are connected together over a separate voice bus. A standard computer platform such as manufactured by Dell Computer of Austin, Texas (USA) can be used.

One typical standard for computer telephony switching between multiple boards within a single platform is called MVIP. The MVIP bus may be implemented using various configurations as known in the art, but preferably is manufactured by Dialogic Corporation of Parsippany, New Jersey (USA), and consists of eight 2 megabyte serial highways and clock signals that are routed from one card to another over a ribbon cable. Each of these highways is accordingly partitioned into 32 channels for a total capacity of 256 full-duplex voice channels on the MVIP bus.

Thusly configured, it is the telephony hardware and the MVIP bus which eliminates the external switching by providing a hardware platform capable of digital switching calls in a non-blocking switch matrix within the telephony hardware bus.

The foregoing has outlined rather broadly the features and technical advantages of the present invention in order that the detailed description of the invention that follows may be better understood. Additional features and advantages of the invention will be described hereinafter which for the subject of the claims of the invention. It should be appreciated by those

skilled in the art that the conception and the specific embodiment disclosed may be readily utilized as a basis for modifying or designing other structures for carrying out the same purposes of the present invention. It should also be realized by those skilled in the art that such equivalent constructions do not depart from the spirit and scope of the invention as set forth in the appended claims.

IN THE CLAIMS

We claim:



1.) A computer based apparatus for managing multiple vendor telephony resources within a computer platform wherein the apparatus comprises:

a telephony server interconnected with a telephony network for local and remote telephony client access;

a software abstraction layer residing on said server for connecting with telephony clients;

a computing platform having at least a voice bus and multiple telephony boards; and

a software abstraction layer for abstracting said computing platform.

2.) A computer based method for managing multiple vendor telephony resources:

determining the type of hardware abstraction layers that exist on a computer platform;

initiating a device factory for each type of hardware that currently exists in said competing platform;

creating requests from application sessions;

controlling computing platform based on requests from said application sessions;

establishing client event session with a telephony manager in order to establish a notification to telephony client reflecting a state change with said computing platform;

controlling said computing platform according to requests from said telephony client to said telephony manager, via a hardware abstraction layer