MS 2000 Series Basics

MS 2000 Series platform

The Nortel Media Server 2000 Series (MS 2000 Series) is built on the AudioCodes IPmedia 2000 chassis. The IPmedia 2000 cPCI, rack mount chassis is 1U high and 19 inches wide. The chassis contains one board, the IPM-1610 and its rear transition module, which contains the Ethernet interface for the unit. Up to six MS 2000 Series nodes can be configured in a SAMF frame.

Although the MS 2000 Series IPM-1610 board occupies only one slot in the IPmedia 2000 chassis, it consists of two separate, logical media gateway modules from an OAM&P management perspective. Each module has its own MAC address and IP address. Both modules share a redundant LAN connection through an internal Ethernet switch.

The MS 2000 Series is available in four configurations. These configurations are described in the table below.

Services Enabled	Number of Logical IPM-1610 modules	Number of Ports
- No Conferencing - IVR (announcements and digit collection) - Test Trunks - Legal Intercept (CALEA)	1	120
- Conferencing - IVR (announcements and digit collection) - Test Trunks - Legal Intercept (CALEA)	1	120

Services Enabled	Number of Logical IPM-1610 modules	Number of Ports
- No Conferencing - IVR (announcements and digit collection) - Test Trunks - Legal Intercept (CALEA)	2	240
- Conferencing - IVR (announcements and digit collection) - Test Trunks - Legal Intercept (CALEA)	2	240

MS 2000 Series applications

As an application server supporting audio services, the MS 2000 Series provides an interface for caller interactive features that require the collection of user input and prompt playback. In this capacity, the MS 2000 Series supports the following functions:

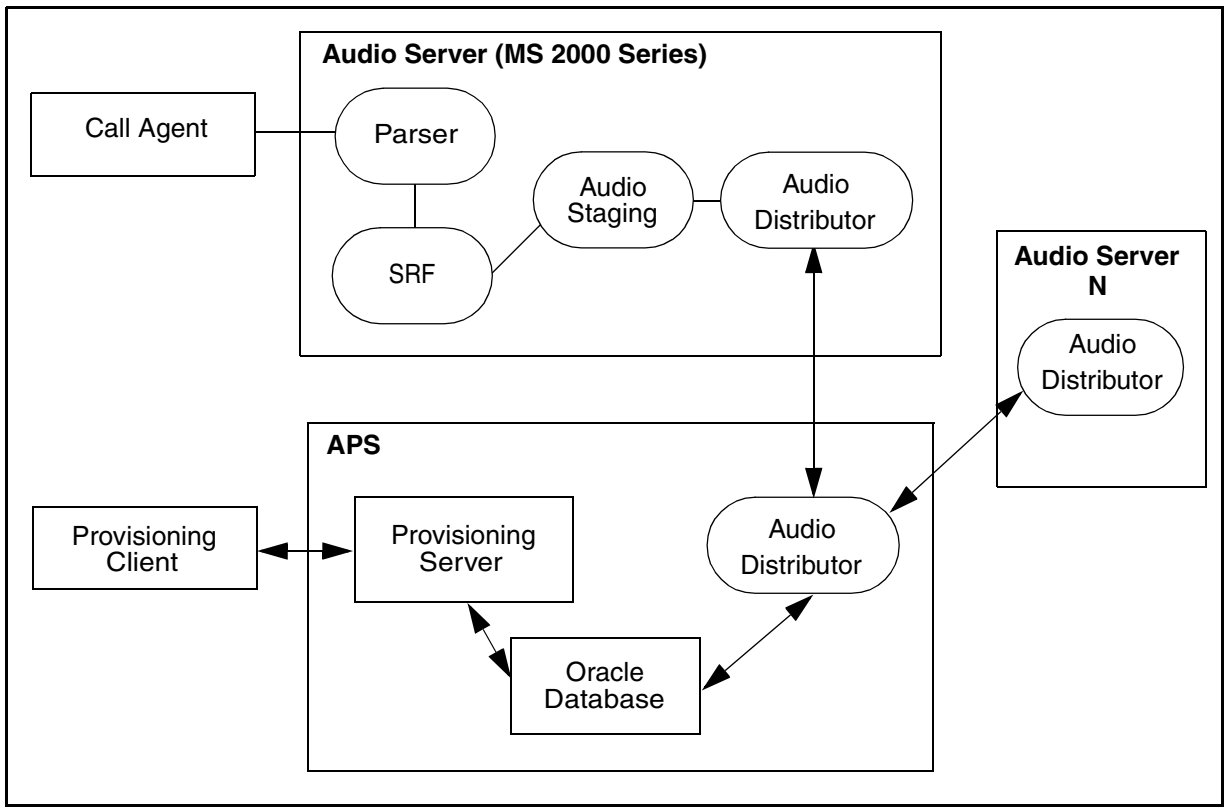
- plays announcements stored as G.711 encoded mulaw and alaw
- plays a set of announcements to the caller which can be interruptible by Dual Tone MultiFrequency (DTMF) digit entry
- plays an announcement and collects DTMF digits
- plays a particular announcement and collects DTMF digits, potentially looking for a specific DTMF digit response using a specified DTMF digit pattern (specific digits, maximum number of digits, or specific digits that can interrupt the announcement)
- plays an announcement that is stored in the runtime database

Audio access

One of the primary roles of the MS 2000 Series is to play audio as directed by a Call Agent. The Call Agent can specify that a single piece of audio be played, or it can specify a more complex audio assemblage be played, such as the current date in Spanish.

The basic relationship between the parts of the audio access process is shown in the following illustration. In its basic operation, the Call Agent sends messages to the Audio Server with instruction to play an audio segment or to provide conferencing resources. The Parser in the Audio Server interprets the messages sent by the Call Agent and, through small, dedicated programs known as "SRFs (Service Resource Functions)," calls on other programs within the audio server to perform the work requested by the Call Agent. If audio is to be played, the SRFs coordinate retrieval of audio from audio staging.

Audio access



Audio is added to the system through the Provisioning Client. The new audio is sent from the Provisioning Client to the Provisioning Server, which, in turn, forwards the audio to the Oracle database in the APS. After the audio has been added to the database, the Audio Distributor can forward it on to other audio server nodes in the system. Audio is normally automatically distributed by the Audio Distributor to other audio server nodes in the system on a scheduled periodic basis.

Audio distribution

Audio is distributed to the MS 2000 Series either upon start-up of a node or on a periodic basis.

Upon start-up, the MS 2000 Series node sends a "heartbeat" to the APS, which causes the audio distribution process to begin. The MS 2000 Series audio distributor first checks to determine whether the node is defined in the APS database. If the node is defined, the distribution process continues; if the node is not defined, the distribution does not continue.

Periodic updates can be made either manually or through an automatic provisioning session that occurs hourly for all nodes. In either event, the

APS, rather than the MS 2000 Series node, initiates the audio distribution process. The audio distributor first determines whether there is sufficient memory in the node to store the audio being distributed to it. This is calculated by subtracting the amount of memory required for storing audio currently active in the node from the total memory allocated to the node. If there is not enough memory, an error log is generated and the distribution process stops.

The MS 2000 Series can store audio for a maximum of two languages. Before distributing audio to an MS 2000 Series node, the audio distributor determines whether primary and secondary languages have been defined for the node. If either a primary or secondary language has not been defined, audio variables in the missing language cannot be distributed to the node. If neither primary nor secondary languages have been defined, no audio can be distributed to the node.

At distribution time, the audio distributor checks to determine whether language audio files for the node have been imported into the APS database. If the language audio files have not been imported into the APS database, a log is issued. Without these language audio files, any requests sent to the MS 2000 Series node to play variables in the languages will fail.

Conferencing

The conferencing capacity of the MS 2000 Series is based on the total number of conferences and the total number of participants in all of the conferences that are occurring simultaneously. An MS 2000 Series node supporting 120 channels can support 120 conference participants. Each conference can accommodate from 3 through 64 participants. Therefore, the maximum number of simultaneous conferences that can be supported is 40.

Audio Provisioning Server (APS)

The G.711 encoded mulaw and alaw announcements that the MS 2000 Series plays are provisioned using the Audio Provisioning Server (APS). The APS provides a centralized location and Web-based Generic User Interface (GUI) (through Internet Explorer 5.0 or higher, or through Netscape Navigator 4.5 or higher) for uploading announcement files from a client machine to the APS, where the files can be prepared for use by a gateway.

APS hardware

The APS server is configured on a server that also hosts the CS 2000 Management Tools. For information about the server on which the APS resides, see your solution's Basics document.

APS software

The following list shows the layered software architecture elements of the APS:

- Java Applet on client workstation
- Remote database proxy on client workstation
- HTTP/ftp communication over Inter/Intra-net between APS and Client(s)
- Web server (Database Servlets)
- IPS base (Audits, Permissions)
- SSPFS (Installs, Job Scheduler, System Admin)
- DMP (MBrow, Apache, JDBC)
- Database (Oracle 9i)
- Operating System (Solaris 2.8. JRE 1.4.1)

The *applets* provide the client interface to the database. The applets gather information from the APS GUI that is sent to the servlet in the form of a request. The applets communicate with the database servlets through the network using the hypertext transport protocol (HTTP).

For the audio management GUI, the *servlets* act as a request and response handler between the applets and the database server application. The servlets use the remote method invocation (RMI) capabilities provided by the Java class libraries to communicate with the database server application. For the administrative GUI, the servlets communicate directly with the IPS database using the Java Database Connectivity (JDBC) interface.

The *database server application* communicates with the IPS database using the JDBC interface. The database server application translates servlet requests into database operations. It provides a response back to the servlet, which includes a success or failure of the database's ability to process the request.

The *Interactive Provisioning System (IPS)* relational database stores the actual physical audio segments and the properties or the relationships that have been defined for physical segments, packages, sequences, sets and variables. It interacts with the MS 2000 Series runtime database hourly to provide update audio management data.