Arquitectura de gestión de redes para entornos heterogéneos y su aplicación a las redes EHAS

Documento
Trabajo de Grado
Maestría en Ingeniería, Área Telemática

Ingeniera Eva Juliana Maya Ortiz

Director
Magíster Andrés Lara Silva

Asesores
Ingeniero Arnau Sánchez Sala, Fundación EHAS
Doctor Joaquín Soane Pascual, Universidad Politécnica de Madrid

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telemática
Grupo de Ingeniería Telemática
Popayán, noviembre de 2007

TABLA DE CONTENIDO

INTRODUCCIÓN

En las redes de comunicaciones actuales se utilizan diferentes tecnologías de comunicaciones, tanto cableadas como inalámbricas, que tienen una gran cantidad de características y difieren entre sí en varios aspectos, como por ejemplo, velocidad, ancho de banda y alcance máximo. Además, en estas redes de comunicaciones, los equipos tienen hardware y software con múltiples y variadas características que hacen que los entornos de comunicaciones actuales sean realmente heterogéneos. Estas redes de comunicaciones interoperan para proporcionar los servicios requeridos por los usuarios, es decir, que entre estos múltiples entornos heterogéneos se crea una verdadera simbiosis para ofrecer servicios de datos, voz y video a los usuarios actuales que cada vez exigen más y mejores servicios.

Así como los requerimientos de los usuarios en cuanto a servicios crecen, las exigencias en cuanto a las características de las redes también aumentan rápidamente. Por este motivo, las grandes empresas y grupos de investigación y desarrollo líderes en el sector de la computación y la comunicación dedican gran parte de sus recursos a crear o a participar en la creación de estándares, arquitecturas, plataformas y herramientas de gestión de redes que permitan monitorear y controlar las redes de comunicaciones de una forma adecuada, más aún si se tienen en cuenta los entornos heterogéneos que conforman las redes de comunicaciones actuales. Pero este interés ha hecho que exista un gran abanico de posibilidades que puede llegar a ser confuso y complejo de manejar, incluso abrumador para los encargados de la gestión de redes.

Por otro lado, la Fundación EHAS (Enlace Hispanoamericano de Salud) es una organización sin ánimo de lucro que busca contribuir a la mejora de los servicios de los sistemas públicos de asistencia sanitaria y de salud en las zonas rurales de los países de América Latina mediante el uso de tecnologías adecuadas de información y comunicaciones, es decir, de bajo costo, de bajo consumo e inalámbricas. EHAS utiliza tecnologías como: Wi-Fi, VHF y HF. La tecnología que se escoge para cada zona a comunicar depende de sus características de propagación. Teniendo en cuenta esto, se puede decir que las redes EHAS contienen una mezcla de tecnologías, es decir, son redes híbridas que permiten la mejor comunicación de las zonas objetivo.

Para que la Fundación cumpla con su objetivo es fundamental que las redes EHAS funcionen adecuadamente la mayor parte del tiempo, pero una vez que se instalaban no se conocía su configuración, desempeño ni sus fallas, no se podía detectar sus problemas ni sus causas oportunamente y por tanto no se podían resolver eficiente y eficazmente, incluso, no se sabía cuando se caía un equipo y podían pasar días antes de saberlo, es decir, las redes podían permanecer fuera de servicio por mucho tiempo sin que se tomara alguna acción al respecto, lo que cobra mayor importancia debido a que la Fundación EHAS trabaja con comunidades aisladas que generalmente no tienen mucho contacto con sistemas informáticos y pueden perder fácilmente la credibilidad y confianza en el proyecto.

En consecuencia, se detectó la necesidad imperiosa de conocer la disponibilidad de las redes EHAS así como también su uso, ya que esto da una medida del impacto de las tecnologías de información y comunicaciones en las comunidades. Además, también se

encontró muy importante determinar si las redes funcionan como se espera ya que la Fundación EHAS realiza investigación en tecnologías de información y comunicaciones de bajo costo y es necesario saber si el camino de investigación que la Fundación ha elegido es adecuado y por tanto si está cumpliendo su objetivo. Por otro lado, se pensó en que sería importante poder realizar tareas de control en forma remota ya que las redes se encuentran en zonas apartadas, y en algunas ocasiones, aunque el personal técnico de los puestos y centros de salud se capacita para atender los sistemas instalados se presentan situaciones que requieren la atención por parte de personal con mayor experiencia que de no ser por este mecanismo conllevarían a un alto consumo de tiempo y dinero.

Esto hizo evidente la necesidad de un sistema de gestión de redes EHAS que permitiera obtener información relevante de los equipos de las redes, como por ejemplo, datos de las interfaces de red inalámbricas, que desplegara los datos en tablas y gráficos, así como también, que generara alarmas cuando se cayera un equipo o se sobrepasaran ciertos umbrales, también, que permitiera la ejecución de comandos, y por otro lado, que realizara la adición de los equipos automáticamente, y que además se accediera vía Web y fuera fácil de utilizar para cualquier usuario. Teniendo en cuenta esto, se planteó la obtención de un sistema de gestión que permitiera realizar tareas de monitoreo y control sobre las redes EHAS de forma adecuada, por tanto, se tuvo en cuenta que las redes EHAS son muy diferentes a las redes tradicionales, pero precisamente esto hizo que este proyecto de investigación fuera diferente a los demás proyectos de gestión de redes.

En este proyecto se tuvo en cuenta que las redes EHAS combinan tecnologías como Wi-Fi, VHF y HF, y además tienen dos características muy importantes. La primera, es que tienen bajo ancho de banda, incluso los enlaces Wi-Fi, debido a que son de larga distancia, y la segunda, es que algunos equipos pueden no estar conectados todo el tiempo a la red, sobre todo las estaciones VHF y HF debido al gran consumo de energía que esto implicaría.

En el caso de las redes EHAS, el gestor generalmente no tiene conectividad permanente con los equipos gestionados, un escenario totalmente diferente a los escenarios comunes de gestión de redes, por tanto, el objetivo de este proyecto fue obtener un sistema de gestión adecuado a las redes EHAS, una solución muy diferente a las convencionales ya que las redes EHAS no son como la mayoría de las redes, pero precisamente en esto radica la importancia de este proyecto, ya que una solución adecuada es aquella que tiene en cuenta las necesidades reales, los condicionantes del entorno y en este caso, las características particulares y un tanto complejas de las redes EHAS que son las que hacen a este proyecto de investigación bastante interesante.

En este proyecto se planteó todo un reto de investigación y desarrollo, la obtención de una arquitectura de gestión de redes adecuada para los entornos heterogéneos actuales y además, su aplicación a la redes EHAS.

En este proyecto se plantearon los siguientes objetivos:

OBJETIVO GENERAL

- Obtener una arquitectura de gestión de redes para entornos heterogéneos y realizar su aplicación a las redes EHAS.

OBJETIVOS ESPECÍFICOS

- Determinar las capacidades y los alcances de las tecnologías de gestión de redes actuales. CAPÍTULO 1.
- Establecer los escenarios y campos de aplicación de las tecnologías de gestión de redes actuales, y sus relaciones y posibilidades de integración. CAPÍTULO 2.
- Establecer las principales características de las redes EHAS que se deben tener en cuenta para obtener un sistema de gestión de redes adecuado. CAPÍTULO 3.
- Seleccionar la opción más adecuada para gestionar las redes EHAS teniendo en cuenta sus características. CAPÍTULO 4.
- Obtener un sistema de gestión de redes que permita realizar tareas de monitoreo y control de forma adecuada sobre las redes EHAS. CAPÍTULO 5.

En el capítulo 1 se describen los estándares de gestión de redes y servicios más importantes actualmente, además, una nueva tecnología de red caracterizada por entornos heterogéneos conocida como Grid, así como también su gestión, y finalmente se describen las iniciativas que grupos de investigación y desarrollo, y empresas están trabajando para lograr la gestión de entornos heterogéneos.

En el capítulo 2 se describen las posibilidades de integración de los estándares de gestión de redes más importantes actualmente como SNMP, IPMI, HPI, AIS, WBEM, entre otros, y finalmente se describen alternativas para lograr la interoperabilidad de WBEM, uno de los estándares más utilizado hoy en día, y que se integra muy bien con muchos otros estándares de gestión.

En el capítulo 3 se describen las características de las redes EHAS y se resalta el trabajo que la Fundación EHAS ha realizado para lograr mejores comunicaciones a través HF, VHF e IEEE 802.11, además, se describe la red EHAS Colombia y EHAS Perú Cusco y finalmente, dos alternativas para las redes EHAS que son WIMAX y DTNs, así como también su gestión utilizando SNMP.

En el capítulo 4 se presentan los requerimientos de Linux como proveedor de servicios y una arquitectura basada en este sistema operativo para ofrecer servicios de alta disponibilidad como base para establecer una arquitectura de un servicio de gestión de redes, y se centra la atención en la definición de una arquitectura de gestión para entornos heterogéneos, se realizan adaptaciones de esa arquitectura utilizando los estándares de gestión más utilizados actualmente, y finalmente se plantea una arquitectura de gestión para las redes EHAS.

En el capítulo 5 se describen diferentes alternativas que se estudiaron para gestionar estaciones HF, VHF y enrutadores inalámbricos, así como también, la herramienta de gestión utilizada, Zabbix, además, el diseño y la implementación del sistema de gestión de redes EHAS, tanto en los equipos gestionados como en el gestor, y finalmente, se muestran las funcionalidades del sistema de gestión, y se describe su instalación y uso en las redes EHAS.

1. TECNOLOGÍAS DE GESTIÓN Y ENTORNOS HETOROGÉNEOS

En este capítulo se describen los estándares de gestión de redes y servicios más importantes actualmente, además, una nueva tecnología de red caracterizada por entornos heterogéneos conocida como Grid, así como también su gestión, y finalmente se describen las iniciativas que grupos de investigación y desarrollo, y empresas están trabajando para lograr la gestión de entornos heterogéneos.

1.1. TECNOLOGÍAS DE GESTIÓN

A continuación se describe brevemente IPMI, HPI, AIS, CIM, WBEM, SMASH, DASH, WS-Management y WSDM.

1.1.1. IPMI. La Interfaz de Gestión de Plataforma Inteligente, IPMI (Intelligent Platform Management Interface) es un estándar de Intel. IPMI [1] define interfaces estandarizadas, basadas en mensajes para el subsistema de gestión de plataforma inteligente.

El término gestión de plataforma inteligente hace referencia a las funciones de monitoreo y control que se incorporan al hardware de la plataforma. La gestión de plataforma inteligente generalmente permite la monitorización de temperaturas del sistema, voltajes, ventiladores, fuentes de potencia, buses, seguridad física del sistema, etc., además el restablecimiento automático y manual del sistema, es decir, permite realizar reinciaciones locales o remotos, encender y apagar el sistema, registrar condiciones anormales o fuera de rango para las que es necesario el envío de alertas, y además, permite almacenar información de inventario que puede ayudar a identificar una unidad de hardware con fallas. La característica clave de la gestión de plataforma inteligente es que estas funciones están disponibles independientemente de los procesadores principales, BIOS y sistema operativo, cuando el software de gestión del sistema no está disponible e incluso cuando el sistema está apagado.

Además de la especificación principal para IPMI existe un conjunto de especificaciones de soporte:

- Bus de Gestión de Plataforma Inteligente, IPMB (Intelligent Platform Management Bus) es un bus basado en I2C[1] que proporciona una interconexión estandarizada entre diferentes boards dentro de un chasis. En otras palabras, es un bus de gestión interno que permite extender la gestión de plataforma dentro de un chasis.
- Asignación de Direcciones IPMB documenta los diferentes rangos y asignaciones de direcciones en el IPMB.
- Bus de Gestión de Chasis Inteligente, ICMB (Intelligent Chassis Management Bus) proporciona una interfaz estandarizada para el monitoreo y el control de gestión de plataforma entre chasis. El ICMB está diseñado de tal forma que puede ser implementado con un dispositivo que se conecta al IPMB. Esto permite al ICMB ser implementado como una adición a sistemas que tienen un IPMB existente. En otras palabras, es un bus de gestión externo entre sistemas con IPMI.
- La especificación de Formato de Traps de Eventos de la Plataforma define el formato de traps SNMP utilizados para las alertas.

[1] I2C. Bus Inter-Circuitos Integrados (Inter-Integrated Circuit bus). Es un bus serial, de dos hilos, multi-maestro.

- Definición de Almacenamiento de Información FRU de Gestión de Plataforma define el formato de la información de Unidades Reemplazable en Campo, FRUs (Field Replaceable Units), tal como números seriales y números de partes para varias boards y otros componentes reemplazables, accesibles en un sistema basado en IPMI.

En la figura 1 se muestra el diagrama en bloques de IPMI.

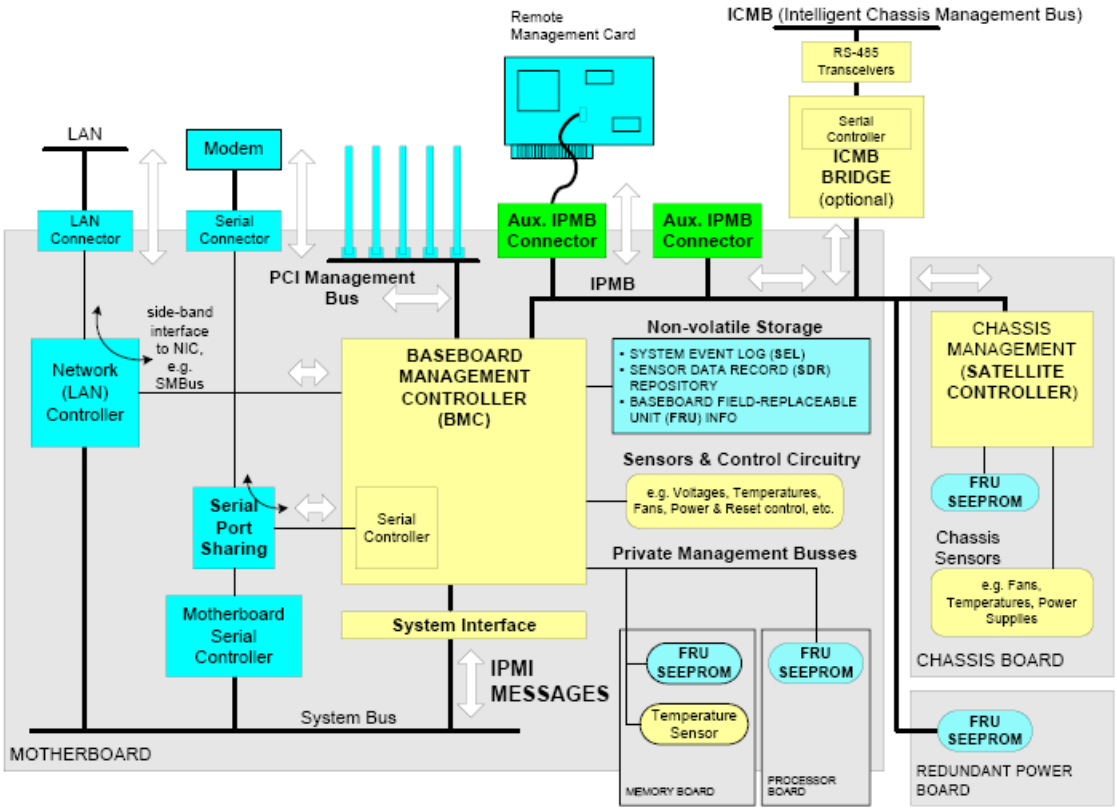


Figura 1. Diagrama en bloques de IPMI [1]

En la figura1 se observan los principales componentes de una implementación IPMI. En el corazón de la arquitectura IPMI está un microcontrolador llamado BMC, Controlador de Gestión Baseboard (Baseboard Management Controller). El BMC proporciona la inteligencia detrás la gestión de plataforma inteligente, ya que gestiona la interfaz entre el software de gestión del sistema y el hardware de gestión de plataforma, proporciona monitoreo autónomo, registro de eventos y control de recuperación, y sirve como el gateway entre el software de gestión del sistema y el IPMB y el ICMB.

IPMI soporta la extensión de gestión de plataforma al conectar controladores de gestión adicionales al sistema utilizando el IPMB. El IPMB se enruta entre los principales módulos del sistema y se utiliza para la comunicación a y entre controladores de gestión. Debido a que los controladores de gestión adicionales generalmente están distribuidos en otras boards dentro del sistema, lejos del BMC central, algunas veces se llaman controladores satélite.

IPMI utiliza además de IPMB, diferentes interfaces también basadas en mensajes para el subsistema de gestión de plataforma, tales como serial/módem, LAN, ICMB, Bus de

Gestión PCI y las Interfaces del Sistema del lado del software del sistema al BMC, que se representan con las flechas dobles en la figura 1. La mensajería IPMI utiliza un protocolo de requerimiento/respuesta donde los mensajes de requerimientos IPMI comúnmente se llaman comandos.

IPMI define Interfaces del Sistema estandarizadas que el software del sistema utiliza para transferir y recibir mensajes IPMI al y del BMC. Para soportar una variedad de microcontroladores, IPMI ofrece un conjunto de Interfaces del Sistema, por tanto, utilizar estas interfaces es clave para habilitar el software cross-plataforma. Las Interfaces del Sistema IPMI son SMIC (System Management Interface Chip), KCS (Keyboard Controller Style), BT (Block Transfer) y SSIF (SMBus System Interface). El software local ejecutándose en el sistema gestionado y que utiliza las Interfaces del Sistema al BMC generalmente se llama Software de Gestión de Sistema, SMS (System Management Software). Las interfaces ICMB, LAN y serial/módem generalmente se utilizan para la comunicación con el software de gestión en otro sistema. El software remoto que se utiliza para comunicarse con el BMC se llama consola remota.

El IPMB permite que el BMC acepte mensajes de requerimientos IPMI de otros controladores de gestión en el sistema y también proporciona una conexión que permite que otras tarjetas accedan al subsistema de gestión de plataforma. Un BMC que incluye soporte de la interfaz IPMB también proporciona la capacidad para que el software del sistema envíe y reciba mensajes a y desde el IPMB utilizando el BMC como una clase de controlador de comunicación.

La especificación ICMB describe las interfaces para permitir el acceso a través de un Controlador Puente ICMB. El ICMB está especificado de tal forma que el Controlador Puente ICMB se puede adicionar a una implementación IPMI existente que contenga un IPMB.

IPMI también permite la ejecución de comandos de acceso I2C de bajo nivel que se pueden utilizar para acceder dispositivos I2C no inteligentes, es decir, dispositivos que no manejan comandos IPMI, que se encuentran en el IPMB o buses privados, a través de un controlador de gestión. El IPMB también puede soportar dispositivos esclavo SMBus[2], con algunas restricciones.

IPMI permite utilizar buses privados como un mecanismo para acceder SEEPROMs (Serial Electrically Erasable Programmable ROMs) que mantienen información FRU (Field Replaceable Unit). Los buses privados también se pueden utilizar para proporcionar interfaces de acceso de bajo nivel a otros dispositivos I2C o SMBUS, aunque la especificación IPMI no cubre el modo en que se utilizan tales dispositivos.

Un sistema de clase empresarial generalmente tendrá información FRU para cada board del sistema principal, por ejemplo, para la board del procesador, la board de memoria, la board de I/O, etc. Los datos FRU incluyen información tal como número serial, número de parte, modelo y etiqueta.
IPMI proporciona información FRU en dos modos, a través de un controlador de gestión o a través de SEEPROMs FRU. La información FRU que es manejada por un controlador

[2] SMBUS. Bus de Gestión del Sistema (System Management BUS). Es un bus que se basa en I2C al que se conectan la mayoría de dispositivos de monitorización que permiten obtener información del hardware.

de gestión se accede utilizando comandos IPMI. Por otro lado, las SEEPROMs FRU proporcionan un mecanismo para almacenar información FRU sin requerir un controlador de gestión en la unidad reemplazable. Las SEEPROMs FRU se pueden acceder a través de un bus de gestión privado conectado al controlador de gestión, o si es necesario, se pueden conectar directamente al IPMB o Bus de Gestión PCI.

El acceso a información monitoreada, tal como temperatura y voltajes, estado de ventiladores, etc., se proporciona a través del Modelo de Sensores IPMI, que en lugar de dar acceso directo al hardware de monitoreo, permite utilizar comandos de sensor abstractos, implementados a través del controlador, que aíslan el software de cambios en la implementación del hardware de gestión de plataforma.

La información que describe las capacidades de gestión de plataforma se proporciona a través de dos mecanismos, comandos de capacidades y Registros de Datos de Sensores, SDRs (Sensor Data Records). Los comandos de capacidades son comandos dentro de los conjuntos de comandos IPMI que proporcionan información sobre otros comandos y funciones que el controlador puede manejar. Los SDRs son registros de datos que contienen información acerca del tipo y número de sensores en la plataforma, umbrales de sensores, capacidades de generación de eventos e información sobre qué tipos de lecturas proporcionan los sensores. Los SDRs se mantienen en una sola área de almacenamiento no volátil centralizada que es gestionada por el BMC llamada Repositorio SDR.

En los SDRs está incluida la información que indica qué entidad del sistema está monitoreando el sensor, por ejemplo, una board de memoria, y también proporciona un enlace a la información FRU para la entidad.

El BMC proporciona un Log de Eventos del Sistema, SEL (System Event Log) centralizado, no volátil. Una entrada SEL puede indicar el controlador, sensor, tipo de sensor y tipo de evento asociado con el evento. Esta información es útil en sí misma, pero cuando se combina con información SDR, el evento se puede correlacionar a la entidad o FRU asociada con el evento. Correlacionar un evento con la FRU puede ayudar a guiar al área del problema, o incluso a identificar las partes de reemplazo que se deben cambiar.

Los controladores de gestión, sensores, información SEL, información SDR, etc. son de valor limitado sin el Software de Gestión de Sistema, SMS (System Management Software) que interprete, maneje y presente la información. La gestión de plataforma es solamente un subconjunto de la gestión de sistemas.

Con respecto a la arquitectura de gestión de plataforma, el Software de Gestión de Sistema se encarga de:

- Obtener información del SEL para conseguir información de nuevos eventos y actuar sobre ellos como sea apropiado.
- Gestionar el SEL.
- Leer e interpretar la información del Repositorio SDR.
- Obtener información de sensores.
- Ser la fuente de mensajes de eventos potenciales.

Los promotores de IPMI además de Intel son Hewlett Packard, NEC, Dell, y existen más de 192 adaptadores, un número que sigue creciendo.

Existe una implementación IPMI de fuente abierta, OpenIPMI [2], además, herramientas como ipmitool [3] e ipmiutil [4].

1.1.2. HPI. La Interfaz de Plataforma Hardware, HPI (Hardware Platform Interface) es una especificación del SAF (Service Availability Forum). HPI [5] separa el hardware del middleware de gestión y los hace independientes uno del otro. HPI especifica un mecanismo genérico para monitorear y controlar sistemas altamente disponibles a través de un conjunto de interfaces de programación consistente, independiente de la plataforma. La especificación HPI proporciona estructuras de datos y definiciones funcionales que se pueden utilizar para interactuar con subconjuntos gestionables de una plataforma o sistema. HPI permite que aplicaciones y middleware, en otras palabras, el Usuario HPI, accedan y gestionen componentes hardware a través de una interfaz estandarizada ya que su meta principal es permitir la portabilidad de código de Usuario HPI a través de una variedad de plataformas hardware.

Con la evolución de las redes de comunicaciones hacia paquetes de datos conmutados multi-servicio, el concepto de disponibilidad de sistema se ha ampliado a disponibilidad de servicio, que se centra en el usuario y busca cumplir las demandas de cinco nueves, o 99,999%, pero en un entorno computacional distribuido que cumple con estándares, neutrales a la plataforma. En este entorno, el sistema que proporciona un servicio crítico que realmente puede ser altamente distribuido y compuesto de varias plataformas individuales, heterogéneas, cooperantes.

La nueva industria consiste de proveedores de hardware y software que entregan varios componentes, es decir, bloques de construcción, que se deben integran fácilmente en una plataforma, sistema, elemento de red o solución completa. El compuesto resultante debe ser compatible con una arquitectura de sistema distribuida que proporcione cinco nueves de disponibilidad para servicios críticos.

Sin importar la fuente, el fabricante o el integrador de estos bloques de construcción, cada componente del sistema debe ser compatible por sí mismo con todos los otros componentes. Adicionalmente, cada bloque de construcción debe presentar al entorno externo una colección mínima de capacidades así como también medios no específicos a la plataforma para interactuar con estas capacidades.

En resumen, cada bloque de construcción debe:

- Proporcionar capacidades de gestión de fallas, tales como facilidades para el monitoreo de estados, detección de fallas y diagnósticos, aislamiento de fallas, recuperación de fallas y reemplazo de componentes.
- Comunicar la configuración, estado de disponibilidad, condiciones de falla, resultados de pruebas de diagnósticos, acciones de recuperación de fallas y datos de estados internos relacionados a Usuarios HPI.

Además de la gestión de fallas y las capacidades de comunicación locales o atómicas, los grupos integrados por bloques de construcción o plataformas deben:

- Ser capaces de gestionar y recuperarse de ciertas clases de fallas sin impactar al servicio ejecutado por la plataforma.
- Monitorear y controlar el hardware del sistema y comunicar los datos del estado de la plataforma a los Usuarios HPI.
- Probar y validar bloques de construcción individuales integrados en la plataforma para asegurar que las acciones de gestión de fallas y las comunicaciones funcionan

apropiadamente y dentro de las restricciones de tiempo prescritos por el servicio o aplicación.

La especificación HPI maneja las interfaces entre bloques de construcción, subsistemas y plataformas agregadas para simplificar la integración de bloques de construcción hardware y software dentro de plataformas funcionales, altamente disponibles para servicios críticos. En la figura 2 se muestra HPI.

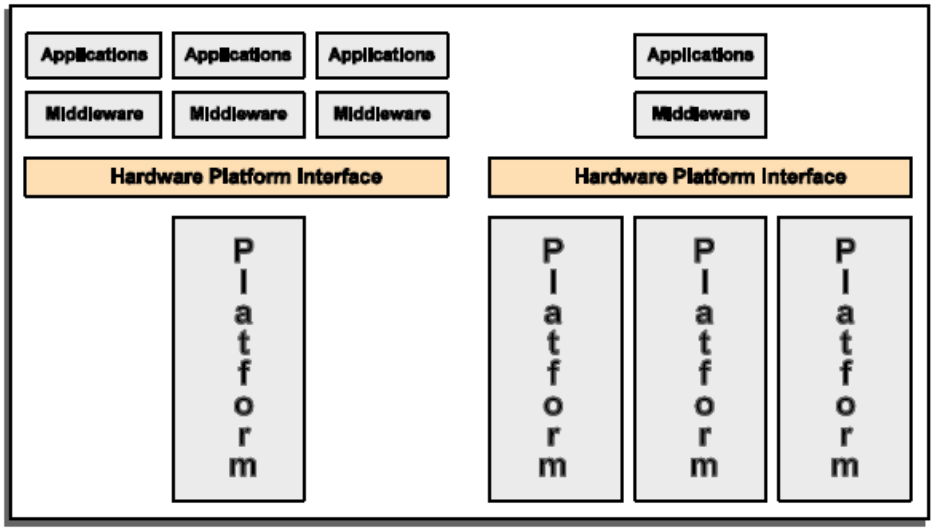


Figura 2. HPI [5]

HPI del SAF se basa fuertemente en los conceptos establecidos por la especificación IPMI para definir capacidades y formatos de datos independientes de la plataforma. Así, una implementación de la interfaz HPI en una plataforma que utiliza IPMI como infraestructura de gestión de plataforma puede ser muy directa. Sin embargo, ya que HPI es una especificación de interfaz genérica, se puede implementar en cualquier plataforma con la suficiente tecnología de gestión de plataforma.

El modelo HPI está compuesto de cuatro conceptos básicos: sesiones, dominios, recursos y entidades, que se muestran en la figura 3, y se describen brevemente a continuación.

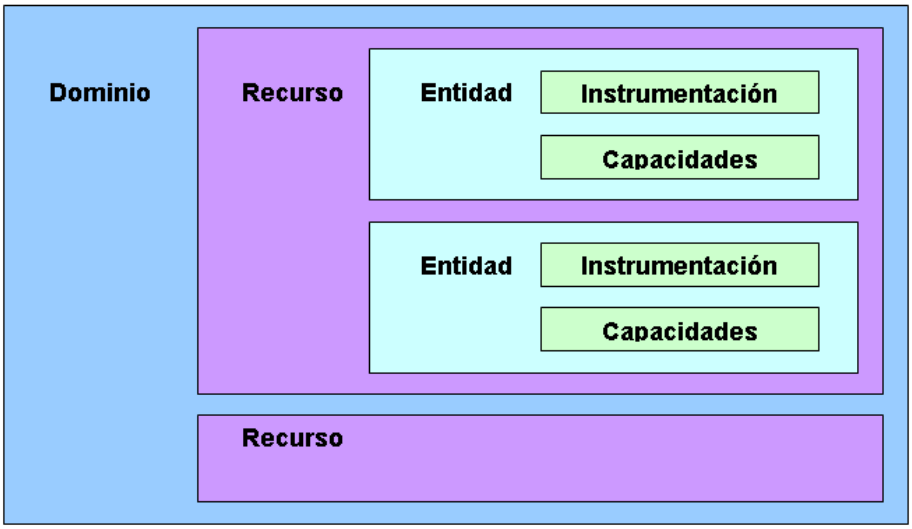


Figura 3. Modelo de HPI

Entidad. Las entidades representan los componentes físicos del sistema. Cada entidad tiene un identificador único llamado trayecto de entidad que está definido por la localización del componente en la jerarquía de contención física del sistema. La capacidad de gestión de una entidad está modelada en HPI por instrumentos de gestión y capacidades de gestión que son los mecanismos por los cuales los Usuarios HPI pueden controlar y recibir información acerca del estado del sistema. La gestión de entidad a través de HPI puede incluir cualquier combinación de las siguientes funciones:

- Leer valores relacionados con la operación o el estado de una entidad. Esta habilidad para leer datos operacionales o de estado se modela a través de instrumentos de gestión Sensores asociados con la entidad.
- Controlar aspectos de la operación de una entidad. Esta habilidad para controlar una entidad se modela a través de instrumentos de gestión Control asociados con la entidad.
- Reportar o actualizar datos de inventario y de configuración estática para una entidad. Estos datos se acceden a través de instrumentos de gestión Repositorio de Datos de Inventario asociado con la entidad.
- Operar timers watchdog asociados con una entidad. Cuando expiran los timers watchdog pueden ocurrir acciones definidas en la implementación. La habilidad para operar timers watchdog se modela a través de instrumentos de gestión Timers Watchdog asociados con la entidad.
- Anunciar la información del estado y condiciones de falla en subsistemas de despliegues o alerta específicos a la plataforma. Esto se lleva a cabo al utilizar instrumentos de gestión Anunciantes asociados con la entidad.
- Encender o apagar una entidad. Esto se lleva a cabo al utilizar la capacidad de gestión Control de Potencia de un recurso asociado con la entidad y las llamadas a las APIs asociadas.
- Ejecutar un reinicación en frío o en caliente sobre una entidad o mantener una entidad en reset. Esto se lleva a cabo al utilizar la capacidad de gestión Control de Reinicación de un recurso asociado con la entidad y las llamadas a las APIs asociadas.

- Designar el software a ser cargado cuando una entidad es encendida o reiniciada. Esto se lleva a cabo al utilizar la capacidad de gestión Cargar ID de un recurso asociado con la entidad y las llamadas a las APIs asociadas.
- Gestionar la inserción o extracción hot swap[3] de una entidad. Esto se lleva a cabo al utilizar la capacidad de gestión Hot Swap Gestionada de un recurso asociado con la entidad y las llamadas a las APIs asociadas.
- Actualizar el firmware contenido en un componente. Esto se lleva a cabo al utilizar Instrumentos de Gestión de Actualización de Firmware, FUMI (Firmware Update Management Instruments) asociados con la entidad.
- Iniciar y monitorear programas de diagnóstico en un componente. Esto se lleva a cabo al utilizar Instrumentos de Gestión Iniciadores de Diagnóstico, DIMI (Diagnostic Initiator Management Instruments) asociados con la entidad.

Recurso. Los recursos proporcionan acceso de gestión a las entidades dentro del sistema. Frecuentemente, los recursos representan funciones ejecutadas por un procesador de control local utilizado para la gestión del hardware de la entidad. Cada recurso es responsable de presentar un conjunto de instrumentos de gestión y capacidades de gestión al Usuario HPI. Los recursos se pueden adicionar o remover dinámicamente en un sistema como los componentes del sistema hot swap que incluyen capacidades de gestión se adicionan y remueven. Adicionalmente, los recursos pueden proporcionar las siguientes funciones:

- Describir el conjunto de instrumentos de gestión contenidos en el recurso. Cada instrumento de gestión se describe con un Registro de Datos de Recurso.
- Almacenar un log histórico de eventos de ese recurso para posterior recuperación. Este mecanismo de almacenamiento y recuperación se modela como un Log de Eventos de un Recurso de Recurso contenido en el recurso.
- Actualizar parámetros de gestión, almacenar nuevos parámetros en almacenamiento no volátil, o cargar parámetros por defecto o almacenados previamente.

Dominio. Un dominio proporciona acceso a conjuntos de recursos. Cada dominio también proporciona información acerca de los recursos que son accesibles a través de ese dominio. Muchos sistemas pueden tener solamente un solo dominio, mientras que los sistemas que tienen áreas dedicadas a tareas separadas, por ejemplo, pueden gestionarlas a través de dominios separados. Los dominios proporcionan las siguientes funciones:

- Mantener y gestionar sesiones abiertas por Usuarios HPI que requieren acceso de gestión a las entidades y recursos del sistema.
- Proporcionar información acerca del conjunto de recursos que son accesibles a través del dominio, además, monitorear la adición o eliminación de recursos por medio de eventos Hot Swap y una Tabla de Presencia de Recursos, RPT (Resource Presence Table).
- Reenviar eventos generados por recursos accesibles a través del domino a Usuarios HPI que se hayan suscrito para recibir eventos del domino.
- Almacenar un log histórico de eventos de los recursos en el domino para recuperación posterior. Este mecanismo de almacenamiento y recuperación se modela como un Log de Eventos de Domino contenido en el domino.

[3] Hot-swapping. Habilidad para remover o reemplazar componentes de una máquina, usualmente un computador, mientras está operando.

- Mantener una tabla de las condiciones de alarma actuales en el dominio y en los recursos que son accesibles a través del domino.
- Mantener una tabla de referencias a otros dominios que un Usuario HPI también puede acceder.

Sesión. Las sesiones proporcionan a Usuarios HPI todo el acceso a una implementación HPI. Un Usuario HPI puede tener múltiples sesiones abiertas y puede haber múltiples sesiones abiertas en cualquier dominio dado. Se prevé que, en futuros lanzamientos, el control de acceso HPI será realizado a nivel de sesión, así diferentes sesiones pueden tener diferente control de acceso. Las sesiones también proporcionan acceso a eventos creados o reenviados por el dominio accedido por la sesión.

Existe una implementación IPMI de fuente abierta, OpenHPI [6].

1.1.3. AIS. La Especificación de Interfaz de Aplicación, AIS (Application Interface Specification) es una especificación del SAF. AIS [7] estandariza la interfaz entre el middleware de alta disponibilidad y las aplicaciones de servicios, por tanto, AIS permite a desarrolladores de aplicaciones escribir software que es portable a través de múltiples plataformas siempre que el middleware subyacente cumpla con AIS.

AIS incluye el Framework de Gestión de Disponibilidad, AMF (Availability Management Framework) y los Servicios AIS.

AMF es la entidad software que proporciona disponibilidad de servicio al coordinar recursos redundantes dentro de un cluster para entregar un sistema sin un solo punto de falla. AMF proporciona una vista de un cluster lógico que consiste de un número de nodos cluster. Estos nodos contienen varios recursos en un entorno computacional distribuido. AMF proporciona un conjunto de APIs para lograr aplicaciones altamente disponibles, que permiten determinar los estados de un componente y monitorear su funcionamiento, así como también, permiten a un componente solicitar al AMF información acerca del estado del componente.

Los Servicios AIS proporcionan la funcionalidad básica del clúster en el que el AMF y la aplicación altamente disponible se implementan. Los Servicios AIS son:
- Servicio de Membresía de Clúster
- Servicio de Punto de Chequeo
- Servicio de Eventos
- Servicio de Mensajes
- Servicio de Bloqueo
- Servicio de Gestión del Modelo de Información
- Servicio de Notificación
- Servicio de Log
- Servicio de Nombrado
- Servicio de Timer

Cada uno de estos servicios se describe brevemente a continuación.

Servicio de Membresía de Clúster. El Servicio de Membresía de Cluster proporciona a las aplicaciones información de membresía acerca de los nodos que han sido configurados administrativamente en la configuración del clúster y es la base de cualquier

sistema con clústers. Un clúster consiste de este conjunto de nodos configurados, cada uno con un nombre de nodo único.

Servicio de Puntos de Chequeo. El Servicio de Puntos de Chequeo permite que los procesos registren datos de puntos de chequeo incrementalmente que se puede utilizar para proteger una aplicación contra fallas. Cuando los procesos se recuperan de una falla con un reinicio o un procedimiento sobre una falla, el Servicio de Puntos de Chequeo se puede utilizar para recuperar los datos de puntos de chequeo previos y para reiniciar la ejecución desde el estado registrado antes de la falla lo que minimiza el impacto de la falla.

Servicio de Eventos. El Servicio de Eventos es un mecanismo de comunicación multipunto a multipunto de publicación/suscripción que está basado en el concepto de canales de evento y que consiste en que uno o más publicadores se comunican asincrónicamente con uno o más suscriptores al utilizar eventos sobre una entidad por todo el clúster llamada canal de eventos. Los publicadores también pueden ser suscriptores en el mismo canal de eventos.

Servicio de Mensajes. El Servicio de Mensajes especifica un sistema de paso de mensajes almacenados basado en el concepto de cola de mensajes para procesos en el mismo nodo o en nodos diferentes. Los mensajes se escriben en colas de mensajes y se leen de ellas. Una sola cola de mensajes permite una comunicación multipunto a punto. Las colas de mensajes son persistentes o no persistentes, y además se pueden agrupar para formar grupos de colas de mensajes, que permiten una comunicación multipunto a multipunto y se pueden utilizar para distribuir mensajes entre colas de mensajes que pertenecen al grupo de colas de mensajes.

Servicio de bloqueo. El Servicio de Bloqueo es un servicio de bloqueo distribuido, previsto para ser utilizado en un clúster, donde los procesos en diferentes nodos pueden competir con otros para acceder a un recurso compartido.

El Servicio de Bloqueo proporciona entidades llamadas recursos de bloqueo que los procesos de una aplicación utilizan para coordinar el acceso a recursos compartidos, además, proporciona un modelo de bloqueo simple que soporta un modo de bloqueo para acceso exclusivo y otro para acceso compartido.

Servicio de Gestión del Modelo de Información. Las diferentes entidades de un clúster, tales como componentes proporcionados por el AMF, puntos de chequeo proporcionados por el Servicio de Puntos de Chequeo, o colas de mensajes proporcionadas por el Servicio de Mensajes son representadas por varios objetos del Modelo de Información.

El Modelo de Información, IM (Information Model) está especificado en UML y es gestionado por el Servicio de Gestión del Modelo de Información IMM (Information Model Management).

Los objetos en el Modelo de Información tienen atributos y operaciones administrativas, es decir, operaciones que se pueden ejecutar en las entidades representadas al utilizar interfaces de gestión del sistema. Para aplicaciones de gestión o gestores de objetos, el Servicio IMM proporciona las APIs para crear, acceder y gestionar estos objetos, y por consiguiente entrega las operaciones requeridas a los servicios o aplicaciones AIS

apropiados, llamados implementadores de objetos, que implementan estos objetos para ejecución.

Los objetos y atributos del Modelo de Información se pueden clasificar en dos categorías:
- Objetos y atributos de configuración.
- Objetos y atributos en tiempo de ejecución.

El Servicio IMM expone dos conjuntos de APIs:
1. Una API de Gestión de Objetos, OM-API (Object Management API) expuesta generalmente a aplicaciones de gestión del sistema, por ejemplo, agentes SNMP.
2. Una API que Implementa Objetos, OI-API (Object Implementer API) restringida a Implementadores de Objetos.

Servicio de Notificación. El Servicio de Notificación está basado en alto grado en el Modelo de Gestión de Fallas de la ITU-T.

El Servicio de Notificación se centra alrededor del concepto de notificación, el cual explica un incidente o cambio de estado. Este servicio define cinco tipos de notificaciones con distintos parámetros y son alarma, cambio de estado, creación/eliminación de objetos, cambio de valores de atributos, alarma de seguridad, y se basa en el paradigma publicar/suscribir.

Servicio de Log. El SAF distingue entre un Servicio de Log y un Servicio de Seguimiento, el primero es para información significante para el cluster, basada en funciones adecuadas para administradores de sistemas o herramientas automatizadas, mientras el segundo es para información específica a la implementación de bajo nivel adecuada para desarrolladores o ingenieros de campo.

El Servicio de Log permite a las aplicaciones reenviar registros de logs a través de flujos de logs conocidos que se dirigen a destinos de salida particulares tales como un archivo. Una vez en el destino de salida, un registro de log se somete a reglas de formateo de salida configurables y públicas, y ya que el formato de salida es público, herramientas de tercera parte pueden leer estos archivos de log.

El Servicio de Log define cuatro tipos de flujos de log:
- Alarma
- Notificación
- Sistema
- Aplicación

Servicio de Nombrado. El Servicio de Nombrado proporciona un mecanismo por el cual nombres amigables para el humano se asocian con objetos, de tal forma que estos objetos se pueden buscar dados sus nombres. Los objetos generalmente representan puntos de acceso al servicio, puntos finales de comunicación y otros recursos que proporcionan alguna clase de servicio.

El Servicio de Nombrado no impone una disposición específica o una convención ni en los nombres o los objetos a los que se asocian, y permite a los usuarios del servicio seleccionar y utilizar su propio esquema de nombrado sin asumir ningún hardware específico o configuración de software lógico.

Servicio Timer. El Servicio de Timer proporciona un mecanismo por el que se notifican los procesos cliente cuando un timer expira. Un timer es un objeto lógico que se crea dinámicamente y representa ya sea un tiempo absoluto o una duración.

El Servicio de Timer proporciona dos tipos de timers, timers de eventos simples y timers periódicos. Los timers de eventos simples expiran y se eliminan después de la notificación. Los timers periódicos expiran cada vez que una duración especificada se alcanza y el proceso es notificado acerca de las expiraciones. Los timers periódicos tienen que se eliminados explícitamente al invocar la función de eliminación del timer.

Servicios AIS de Modelamiento. El SAF no especifica ninguna implementación particular de los diferentes servicios AIS. Sin embargo, el SAF recomienda fuertemente utilizar las abstracciones de modelamiento del sistema y las entidades lógicas disponibles por la especificación AMF cuando se implementan tales servicios. Esto promueve un solo esquema de modelamiento AIS unificado, basado en entidades lógicas del AMF, para servicios AIS y aplicaciones SAF, es decir, los servicios AIS son modelados, gestionados y actualizados en el mismo modo en que cualquier otra aplicación SAF sería modelada, gestionada y actualizada.

Dependencias. El AMF y los servicios AIS que generan alarmas o notificaciones tienen una dependencia con el Servicio de Notificación que a su vez tiene una dependencia con el Servicio de Logs.

El Servicio de Gestión del Modelo de Información expone los objetos de configuración y las operaciones administrativas en nombre de los servicios AIS, por lo tanto, cualquier servicio que exponga tal interfaz de gestión depende del Servicio IMM.

Otras interacciones entre el AMF y los servicios AIS, y entre servicios AIS, pueden depender de funciones que no están definidas por esta versión de la especificación y pueden ser definidas por versiones futuras de la especificación.

Existe una implementación IPMI de fuente abierta, OpenAIS [8].

1.1.4. CIM. El Modelo de Información Común, CIM (Common Model Information) es un estándar del DMTF (Distributed Management Task Force). CIM [9] es un modelo de información que permite describir entidades computacionales y de negocios en entornos de Internet, empresa y proveedores de servicios. CIM proporciona una definición y estructura de la información de gestión consistentes utilizando técnicas orientadas a objetos.

CIM es un modelo de información jerárquico que facilita la definición de las variadas interdependencias entre diferentes objetos gestionados como las que existen entre conexiones de redes lógicas y dispositivos físicos subyacentes, o entre una transacción de e-commerce y los servidores Web y de bases de datos de los que depende.

CIM es una vista conceptual del entorno gestionado que unifica y extiende los estándares de instrumentación y gestión existentes tales como SNMP, CMIP, etc., y no requiere ninguna instrumentación particular o formato de repositorio de información persistente.

CIM está compuesto de una Especificación y un Esquema. La Especificación CIM describe el Meta Esquema y sus elementos, además el Formato de Objetos Gestionados,

MOF (Manager Object Format) y establece cómo utilizar el Lenguaje de Modelado Unificado, UML (Unified Modeling Language) para realizar los diagramas de los modelos CIM. Por otro lado, el Esquema CIM describe el Modelo Núcleo y los Modelos Comunes. A continuación se realiza una breve descripción de la Especificación y el Esquema CIM.

1.1.4.1. Especificación CIM. La Especificación CIM describe el Meta Esquema, un meta modelo orientado a objetos basado en UML, que contiene elementos que se deben presentar a las aplicaciones de gestión, tales como clases, propiedades, métodos, entre otros. Además, define un leguaje de sintaxis CIM basado en el Lenguaje de Definición de Interfaces (IDL) llamado MOF y el mecanismo de Nombrado CIM. La especificación CIM no describe implementaciones CIM específicas, APIs o protocolos de comunicaciones, esto está fuera del alcance de la Especificación.

Meta Esquema. Los elementos del Meta Esquema son Esquemas, Clases, Propiedades y Métodos, Indicaciones y Asociaciones que son tipos especiales de Clases, y Referencias que son un tipo especial de propiedad, además Calificadores y Triggers. Los elementos del Meta se muestran en la figura 4.

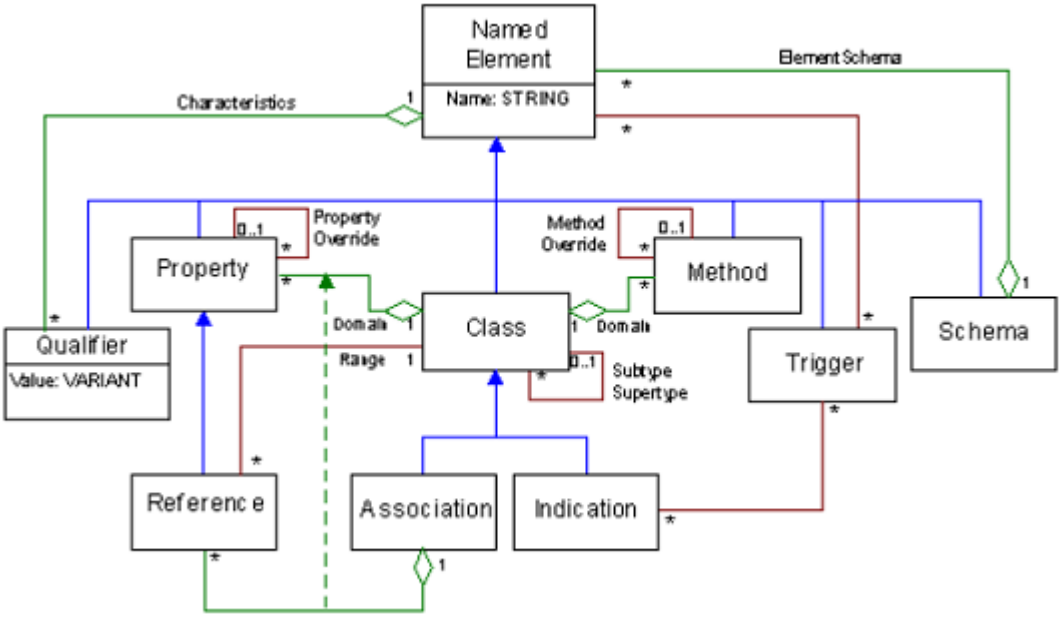


Figura 4. Meta Esquema CIM [9]

Esquema. Es un conjunto de Clases con un solo propietario. Los Esquemas se utilizan para la administración y el nombrado de Clases. Los nombres de las Clases deben ser únicos dentro de su propio Esquema e incluyen el nombre del Esquema.

Clase. Define las Propiedades y los Métodos comunes a un tipo particular de objetos, en otras palabras, es una plantilla para un tipo de elemento gestionado. Las Clases contienen Propiedades, las cuales describen los datos de la clase, y Métodos, los cuales describen el comportamiento de la Clase. El alcance de una Clase es el Esquema al que pertenece y una Clase debe pertenecer solamente a un Esquema, además, el nombre de la Clase debe ser único dentro de ese Esquema y debe incluir el nombre del Esquema.

Propiedad. Es una característica de una Clase. El alcance de una Propiedad es la Clase en la que está definida y debe ser única dentro de la Clase. Una Propiedad tiene un nombre, un tipo de dato, un valor y opcionalmente un valor por defecto. Una Propiedad que no tiene un valor por defecto se inicializa a nulo.

Método. Es una operación de una Clase. El alcance de un Método es la clase en la que está definido y debe ser único dentro de la clase. Una Clase puede tener cero o más Métodos. La declaración de un Método incluye un nombre, tipo de retorno, parámetros de entrada opcionales y parámetros de salida opcionales.

Referencia. Es un tipo de Propiedad especial que identifica las relaciones que las Asociaciones definen entre Clases. Las Asociaciones definen dos o más Propiedades Referencia.

Asociación. Es un tipo de Clase que contiene dos o más Referencias y el Calificador Asociación. Una Asociación establece relaciones entre dos o más Clases, y ya que es una Clase, no afecta a ninguna de las Clases relacionadas. Una Asociación no puede ser una subclase de una Clase no Asociación.

Indicación. Es la representación de la ocurrencia de un evento. Una Indicación es una Clase que tiene el Calificador Indicación, y ya que es un tipo de Clase puede tener propiedades y métodos, y se puede definir jerárquicamente. Las instancias de una Indicación son transitorias y no se pueden recuperar, además, las Indicaciones solamente se pueden recibir si se ha suscrito para ellas antes de que ocurran.

Trigger. Es un reconocimiento del cambio del estado de una Clase.

Calificador. Proporciona información adicional acerca de Clases, Asociaciones, Indicaciones, Referencias, Propiedades, Métodos o parámetros de Métodos. Un Calificador tiene un nombre, tipo, valor, “flavor”, alcance y un valor por defecto opcional. El “flavor” define un comportamiento adicional para un Calificador y el alcance define los Meta Elementos a los que el Calificador se puede aplicar, y puede ser un Meta Elemento, una combinación de Meta Elementos o todos los Meta Elementos.

MOF. La Especificación CIM define un lenguaje basado en el Lenguaje de Definición de Interfaces, IDL (Interface Definition Language) llamado Formato de Objetos Gestionados, MOF (Manager Object Format).

MOF permite escribir definiciones de objetos gestionados en una forma textual. Sus principales componentes son descripciones textuales de declaraciones de Clases, Propiedades, Métodos, Asociaciones, Referencias, Instancias y sus Calificadores asociados.

UML. El DMTF utiliza el Lenguaje de Modelado Unificado, UML (Unified Modeling Language) que es el lenguaje estándar de la industria para construir, especificar, visualizar, y documentar modelos.

En UML, una Clase es representada por un rectángulo que contiene el nombre de la Clase. Una Clase con Propiedades es representada por un rectángulo dividido en dos regiones, una contiene el nombre de la Clase y la otra una lista de Propiedades. Los Métodos son representados por una tercera región que contiene la lista de Métodos. La

herencia, o relación subclase/superclase, se representa por una línea dibujada entre la subclase y la superclase con una flecha indicando la superclase. Las Asociaciones son representadas por líneas con el nombre de la Asociación usualmente colocado cerca al centro de la línea.

Además, generalmente se utilizan líneas azules para herencia, líneas rojas para asociaciones y líneas verdes para agregaciones. El código de colores hace que los diagramas grandes sean mucho más fáciles de leer pero no es parte del estándar UML.

1.1.4.2. Esquema CIM. CIM está estructurado de tal modo que el entorno gestionado se puede ver como un conjunto de sistemas interrelacionados que están compuestos por elementos discretos. El Esquema CIM define Clases, Propiedades, Métodos, etc. que proporcionan un framework conceptual dentro del que es posible organizar la información acerca del entorno gestionado. El Esquema CIM está compuesto por el Modelo Núcleo y los Modelos Comunes.

Modelo Núcleo. Captura nociones que son aplicables a todas las áreas de gestión. El Modelo Núcleo define Clases, Asociaciones, Propiedades y Métodos, etc. que proporcionan un vocabulario básico para describir sistemas gestionados. El Modelo Núcleo representa un punto de partida para determinar cómo extender un Modelo Común.

Modelos Comunes. Capturan nociones que son comunes a áreas de gestión particulares, pero son independientes de cualquier tecnología o implementación. Los Modelos Comunes definen Clases, Asociaciones, Propiedades y Métodos, etc. que proporcionan una vista del área de gestión suficientemente detallada y se puede utilizar como una base para el diseño de programas y, en algunos casos, para su implementación. Los Modelos Comunes son: Aplicaciones, Bases de datos, Dispositivos, Eventos, Interoperación, Métricas, Red, Físico, Políticas, Soporte, Sistemas, Usuario.

El Esquema CIM, es decir, tanto el Modelo Núcleo como los Modelos Comunes se muestran en la figura 5.

Figura 5. Esquema CIM [9]

Esquemas de extensión. Representan extensiones de los Modelos Comunes específicas a la tecnología. Estos esquemas son específicos a entornos, tales como sistemas operativos. Se espera que los Modelos Comunes evolucionen como un resultado de la promoción de los Esquemas de Extensión.

Existen varias herramientas relacionadas con CIM [10] y algunas de ellas son: SMI-S CIM Miner de SNIA, mof2html converter de Nortel Networks, Pretty Printer for MOF de Oracle, MOF Editor de Microsoft, CIM Compatabilty Checker de Intel.

1.1.5. WBEM. La Gestión de Empresa Basada en Web, WBEM (Web Based Enterprise Management) es una especificación del DMTF. WBEM [9] es un conjunto de tecnologías estándar de gestión e Internet desarrolladas para unificar la gestión de entornos computacionales distribuidos. WBEM permite intercambiar información CIM, por lo que incluye mapeos, protocolos, lenguajes de consulta, mecanismos de descubrimiento, etc. para hacerlo de una manera interoperable y eficiente. Tanto WBEM como CIM son tecnologías muy interesantes, pero solamente cuando se combinan proporcionan una solución de gestión de empresa extremo a extremo bastante potente.

1.1.5.1. Componentes de WBEM. Los principales componentes de WBEM son el CIMOM, los Proveedores WBEM y los Clientes WBEM, y se muestran en la figura 6.

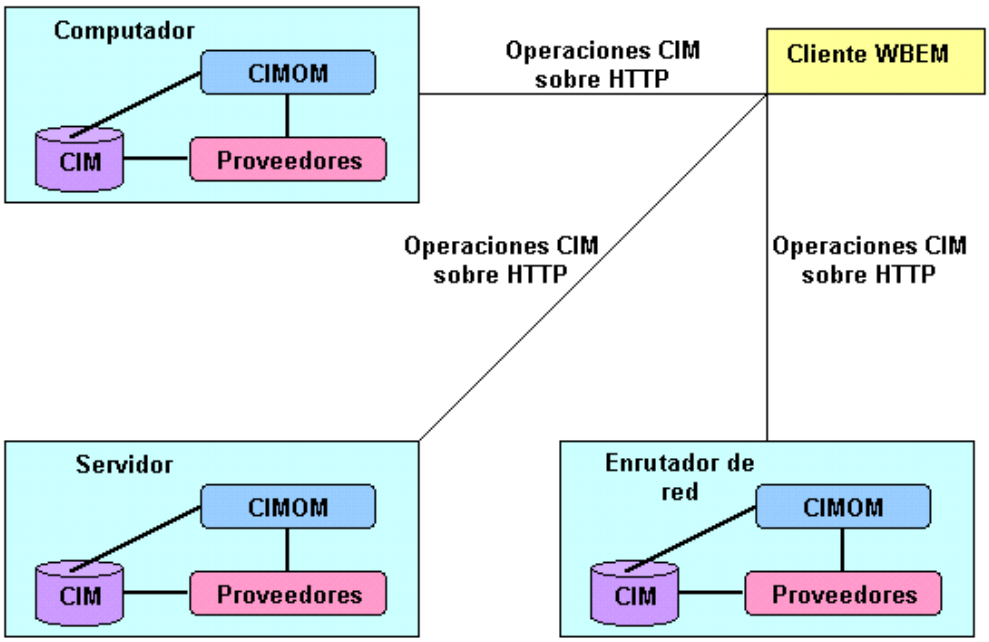


Figura 6. Componentes de WBEM

CIMOM. Es el componente principal de WBEM. El CIMOM es responsable de utilizar los Proveedores necesarios para satisfacer los requerimientos del Cliente WBEM y también de informar a los Clientes adecuados cuando ocurre un evento. El CIMOM tiene un repositorio en el que almacena todos los Esquemas CIM, por tanto, puede verificar que los datos enviados por los Clientes o Proveedores son correctos, además también puede almacenar datos de instancias CIM creadas por los Clientes o Proveedores. Estos datos se llaman datos estáticos, mientras que los datos dinámicos son datos que los

Proveedores obtienen de los recursos gestionados directamente. Si un Cliente WBEM quiere modificar o acceder datos estáticos, el CIMOM solo modifica o accede su repositorio, pero si los datos son dinámicos, el CIMOM llama al Proveedor adecuado, el cual a su vez modifica o accede el recurso gestionado que corresponda.

Proveedor WBEM. Se puede considerar como la interfaz entre el recurso gestionado y el CIMOM. Los datos que son proporcionados por un Proveedor se llaman datos dinámicos. Cuando el CIMOM requiere datos dinámicos, el Proveedor consigue los datos del recurso gestionado y los retorna al CIMOM.

Normalmente, los Proveedores residen en el mismo recurso que el CIMOM, y a diferencia de la comunicación entre el Cliente WBEM y el CIMOM, no existe una interfaz estándar entre los Proveedores y el CIMOM, pero se están haciendo esfuerzos por conseguirla. Generalmente, al CIMOM junto con los Proveedores se lo conoce con el nombre de Servidor CIM.

Clientes WBEM. Se puede considerar como la interfaz entre el gestor WBEM y el CIMOM. El DMTF define Operaciones CIM sobre HTTP para la comunicación entre el Cliente WBEM y el CIMOM. El Cliente genera los requerimientos de operaciones CIM para el CIMOM, y recibe y procesa las respuestas de esos requerimientos. Sin embargo, la mayoría de implementaciones también soportan otros mecanismos de comunicación, como por ejemplo, RMI (Remote Method Invocation) para implementaciones Java, DCOM para la implementación de Microsoft e IPC (Inter Process Communication) para implementaciones UNIX. Sin embargo, el uso de HTTP garantiza la compatibilidad entre cualquier Cliente WBEM y cualquier CIMOM. Un Cliente WBEM también se puede suscribir con el CIMOM para un evento especial, entonces el CIMOM notifica al Cliente cuando un evento ocurre.

1.1.5.2. Especificaciones de WBEM. WBEM contiene las especificaciones que se muestran en la tabla 1.

Tabla 1. Especificaciones de WBEM

Especificaciones de WBEM	
Mapeos	URI XML (eXtensible Markup Language) Esquema XML (bajo desarrollo)
Protocolos	CIM-XML WSDM (bajo desarrollo) WS-Management (bajo desarrollo)
Descubrimiento	SLP
Lenguaje de consulta	CQL

WBEM URI. La especificación de Mapeo URI WBEM define el formato URI (Universal Resource Identifier) para WBEM. Una URI WBEM es una cadena de caracteres compacta que identifica un Elemento CIM. En otras palabras, esta especificación define el mapeo de Nombrado CIM a la sintaxis URI. La URI WBEM es utilizada por WBEM como el método para identificar elementos CIM.

xmlCIM. El mapeo de CIM a XML, también conocido como xmlCIM, está definido en las siguientes especificaciones:

- Representación de CIM utilizando XML
- DTD (Document Type Definition) CIM

xmlCIM está diseñado como un mapeo de Meta Esquema, lo que significa que cada elemento CIM se puede representar en XML. xmlCIM se puede utilizar tanto para representar declaraciones CIM como Clases, Instancias, etc., así como también Mensajes CIM utilizados por los protocolos WBEM. El DMTF también proporciona el DTD CIM.

Además, el DMTF actualmente está definiendo el mapeo de CIM a un Esquema XML. Un documento XML contiene datos en XML mientras un Esquema XML, al igual que un DTD, describe la estructura de un documento XML pero utiliza XML.

CIM-XML. CIM-XML es un protocolo para intercambiar información CIM. CIM-XML incluye los siguientes componentes:
1. CIM
2. xmlCIM
3. Conjunto de operaciones para obtener y manipular datos CIM
4. Encapsulación HTTP

En CIM-XML, xmlCIM es la carga útil, y HTTP es el transporte. CIM-XML define las interacciones entre clientes de gestión e infraestructura de gestión como mensajes CIM, los cuales son paquetes de datos de requerimiento o respuesta utilizados para intercambiar información. Además, el DMTF proporciona el DTD CIM-XML.

Descubrimiento WBEM. El descubrimiento WBEM es cualquier mecanismo que permite a los Clientes el descubrimiento de la localización de CIMOMs WBEM. Aunque WBEM ya ha definido una especificación de descubrimiento que utiliza el Protocolo de Localización de Servicio, SLP (Service Location Protocol), no excluye mecanismos de descubrimiento adicionales en un futuro.

CQL. El Lenguaje de Consulta CIM, CQL (CIM Query Language) proporciona la capacidad para seleccionar propiedades de conjuntos de instancias CIM.

Perfiles de gestión. Un perfil de gestión del DMTF define el Modelo CIM y el comportamiento asociado para un dominio de gestión. El Modelo CIM incluye las Clases, Asociaciones, Propiedades, Métodos, etc., mientras el dominio de gestión es un conjunto de tareas de gestión relacionadas.

Existen implementaciones WBEM de fuente abierta como OpenPegasus [11], OpenWBEM [12], SBLIM [13], WBEM Services [14].

1.1.6. SMASH. La Arquitectura de Gestión de Sistemas para el Hardware de Servidor, SMASH (Systems Management Architecture for Server Hardware) es una iniciativa del DMTF.

El Grupo de Trabajo en Gestión de Servidor, SMWG (Server Management Working Group) del DMTF se encarga de SMASH [15] y desde que se anunció, ha atraído más de 400 miembros de más de 50 compañías. Esto demuestra un fuerte compromiso de vendedores y usuarios de la industria por lograr una gestión de servidor independiente del vendedor y neutral a la plataforma que ha producido como resultado SMASH, un conjunto

de especificaciones que permiten la gestión de servidor a través de diversos entornos en el centro de datos.

Como parte de la Iniciativa SMASH [16], el DMTF ha publicado la Especificación del Protocolo de Línea de Comandos, CLP (Command Line Protocol) para la Gestión de Servidor, SM (Server Management), la Especificación de Direccionamiento de Elemento Gestionado SM, la Especificación de Mapeo CLP a CIM SM, los Requerimientos de Implementación SMASH, y más de 30 Perfiles de Gestión del DMTF, así como también un documento de la Arquitectura SMASH.

SMASH garantiza interoperabilidad ya que utiliza protocolos estándar de la industria, y debido a que está basado en CIM, CLP aprovecha la riqueza de este modelo de información, además, como crea Perfiles estándar de la industria, permite que CIM se pueda aplicar de una manera consistente de tal forma que los sistemas ofrecidos por diferentes vendedores se representan en modos similares.

El DMTF ha realizado un esfuerzo extra en el desarrollo de las Especificaciones SMASH para lograr implementaciones livianas sin sacrificar la riqueza de CIM, y que sean interoperables, independientemente de la arquitectura de la CPU, el hardware del sistema, el vendedor o el sistema operativo.

1.1.6.1. Modelo de servicio. Este modelo describe los términos En Banda, Fuera de Banda, En Servicio y Fuera de Servicio y cómo se utilizan dentro del contexto de la gestión de servidor actualmente.

La gestión En Banda opera con el soporte de componentes hardware que son controlados por el sistema operativo, por ejemplo, una Tarjeta Interfaz de Red, NIC (Network Interface Card).

La gestión Fuera de Banda opera con el soporte de componentes hardware que no son controlados por el sistema operativo. Estos componentes están dedicados a la gestión y permiten el monitoreo y control del hardware del sistema independiente de su estado. Generalmente, estos componentes también pueden interactuar con el sistema operativo, por ejemplo un BMC.

La gestión En Servicio opera con el soporte de componentes software que dependen del sistema operativo, por ejemplo, a través de un servicio o proceso dentro del sistema operativo.

La gestión Fuera de Servicio opera con el soporte de componentes software que requieren que el sistema operativo esté fuera de servicio o no disponible y se coloque en un entorno de gestión alterno.

1.1.6.2. Modelo de arquitectura. Es un modelo que describe la gestión de servidor en términos abstractos independiente de la implementación que abarca desde pequeños a grandes servidores, blades, racks así como también servidores estándar de la industria, servidores de telecomunicaciones y de misión crítica. Este modelo se muestra en la figura 7.

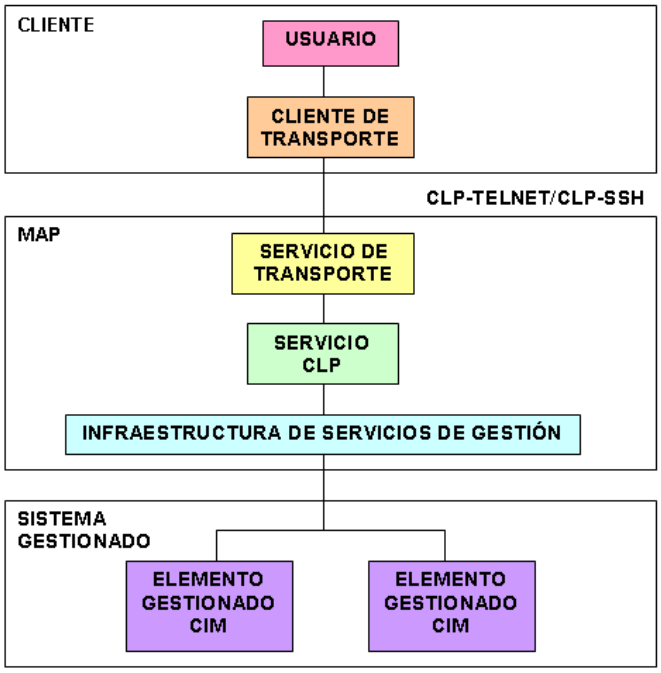


Figura 7. Modelo de arquitectura SMASH

En la figura 7, las líneas entre el cliente y el servicio de transporte indican los protocolos que son visibles externamente, es decir, representan las interfaces de comunicación entre el Punto de Acceso de Gestión, MAP (Manageability Access Point) y el Cliente, y los datos que fluyen, por ejemplo a través de la red. Las otras líneas indican las interfaces visibles semánticamente, es decir, los protocolos, las interfaces y los paquetes no son visibles externamente pero el hecho de que son componentes separados con sus propias semánticas sí lo es.

SMASH define además el Modelo de Operaciones, Perfiles, Direccionamiento de Objetivos, Seguridad y Descubrimiento.

Como se mencionó anteriormente, SMASH define CLP que permite la gestión simple e intuitiva de servidores heterogéneos en el centro de datos independiente del tipo de servidor, estado de la máquina y del estado del sistema operativo, incluso en entornos de gestión Fuera de Banda y Fuera de Servicio.

CLP es un protocolo comando/respuesta. La sintaxis de entrada de CLP define un conjunto de comandos basados en texto, los cuales incluyen verbos y opciones que operan sobre los objetivos. La sintaxis de salida CLP también está definida explícitamente, con formatos seleccionables y opciones que incluyen texto en forma libre, palabra clave=valor y CLPXML.

Los requerimientos de gestión son iniciados por un Usuario CLP, ya sea un humano o un script, y son transmitidos a través de un protocolo, por ejemplo Telnet o SSHv2, al MAP. El MAP tiene un Servicio CLP el cual procesa los comandos recibidos y retorna una respuesta formateada apropiadamente al usuario que la solicitó, ya que el MAP también cuenta con una Infraestructura de Servicios de Gestión que le proporciona acceso a la instrumentación de los sistemas gestionados. Aunque el CLP actualmente define mapeos

a Telent y SSHv2, se puede utilizar cualquier protocolo que pueda transportar mensajes comando/respuesta.

A través de CLP, los usuarios pueden ejecutar operaciones en una estación tales como encender y apagar el sistema, desplegar logs del sistema, configurar el orden de arranque, etc. utilizando los mismos comandos a través de plataformas de diferentes vendedores.

Por otro lado, los Perfiles están diseñados para simplificar la gestión del arranque, la potencia, almacenamiento, actualización de firmware, configuración del sistema, hardware, etc.

Existen dos implementaciones de fuente abierta: omcSMASH [17] y SMASHProxy [18].

1.1.7. DASH. La Arquitectura de Escritorio y móvil para el Hardware de Sistemas, DASH (Desktop and mobile Architecture for System Hardware) es una iniciativa del DMTF.

El Grupo de Trabajo de Escritorio y Móviles, DMWG (Desktop and Mobile Working Group) es responsable de DASH [19] y desde que se anunció ha atraído más de 180 miembros de más de 35 compañías. Esto demuestra un fuerte compromiso de vendedores y usuarios de la industria por lograr una gestión de sistemas de escritorio y móviles independiente del vendedor y neutral a la plataforma que ha producido como resultado DASH, un conjunto de especificaciones que permiten la gestión de estos sistemas independientemente del estado de la máquina o el sistema operativo.

La Iniciativa de Gestión DASH [20] facilita la interoperabilidad ya que utiliza protocolos estándar de la industria, aprovecha la riqueza de CIM debido a que está basada en este modelo de información y como crea Perfiles estándar de la industria, permite que CIM pueda ser aplicado de una manera consistente por todos los vendedores. Además de CIM, DASH aprovecha todas las ventajas del estándar WS-Management del DMTF.

El DMTF ha realizado un esfuerzo extra en el desarrollo de las Especificaciones DASH para lograr implementaciones livianas sin sacrificar la riqueza de CIM, y que sean interoperables, independientemente de la arquitectura de la CPU, el hardware del sistema, el vendedor o el sistema operativo. Las implementaciones incluyen soluciones solamente de software y pequeñas soluciones de firmware.

1.1.7.1. Modelo de servicio. DASH ha adoptado el Modelo de Servicio de SMASH, es decir, las definiciones, términos y modelo para gestión En Banda, Fuera de Banda, En Servicio y Fuera de Servicio de SMASH se aplican a DASH.

1.1.7.2. Modelo de arquitectura. Es un modelo que describe la gestión de sistemas de escritorio y móviles en términos abstractos independiente de la implementación que abarca todas las plataformas. Este modelo se muestra en la figura 8.

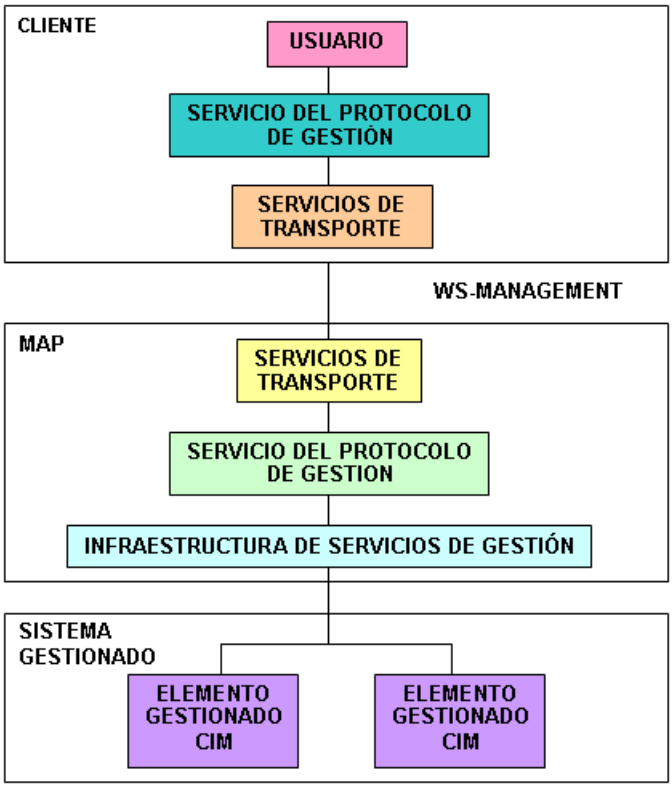


Figura 8. Modelo de arquitectura DASH

En la figura 8, las líneas entre el cliente y el servicio de transporte indican los protocolos que son visibles externamente, es decir, representan las interfaces de comunicación entre el Punto de Acceso de Gestión, MAP (Manageability Access Point) y el Cliente, y los datos que fluyen, por ejemplo a través de la red. Las otras líneas indican las interfaces visibles semánticamente, es decir, los protocolos, las interfaces y los paquetes no son visibles externamente pero el hecho de que son componentes separados con sus propias semánticas sí lo es.

Este modelo es similar al de SMASH, pero en este caso se utiliza WS-Management en lugar de CLP/Telnet o CLP/SSHv2.

1.1.7.3. Protocolos de soporte. DASH utiliza WS-Management con el fin de transportar mensajes para ejecutar operaciones CIM en los objetos CIM representados en Perfiles CIM DASH. La pila de protocolos WS-Management para DASH se muestra en la figura 9.

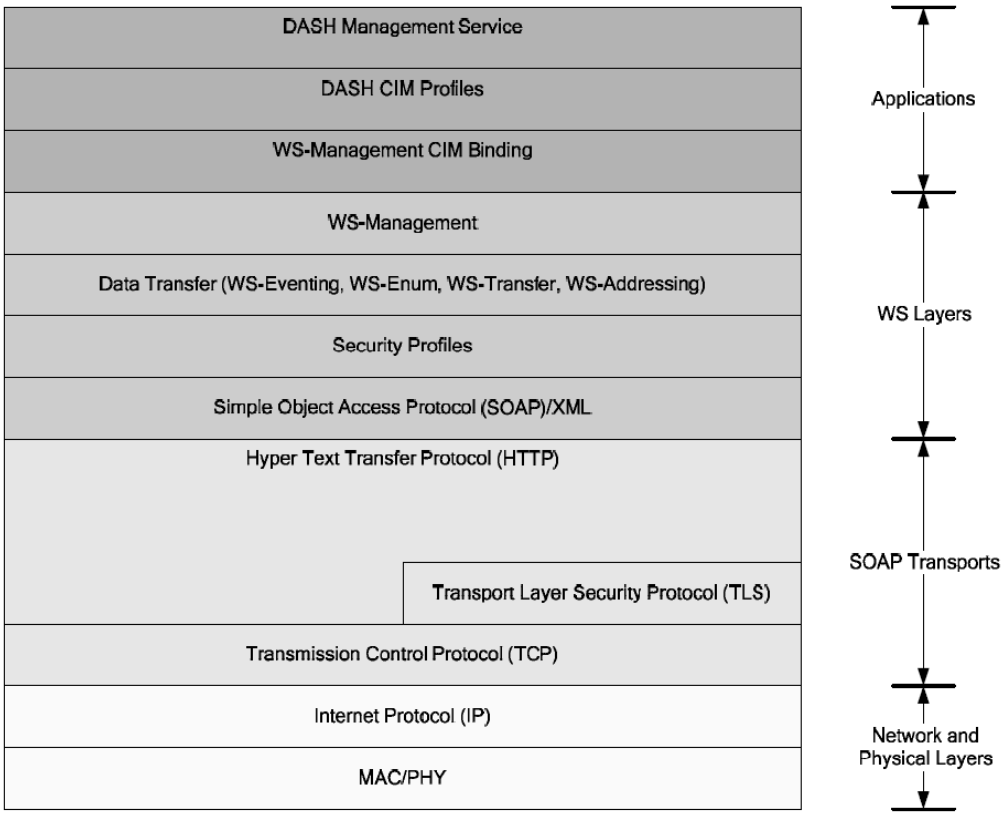


Figura 9. Pila de protocolos WS-Management para DASH [19]

Como se observa en la figura 9, la pila WS-Management se basa en Servicios Web. Las capas más bajas en la pila son la capa física, de enlace y de red. Las siguientes capas son las de transporte de mensajes SOAP que son TCP, TLS y HTTP 1.1. TCP y HTTP 1.1 son obligatorios mientras que TLS no lo es, este protocolo es requerido por Perfiles de seguridad. La siguiente capa maneja mensajes SOAP/XML y los Perfiles de seguridad definen los mecanismos de seguridad requeridos. Sobre esta capa se encuentra la capa de transferencia de datos, la cual se basa en múltiples especificaciones de Servicios Web tales como WS-Transfer, WS-Enumeration y WS-Eventing, para transferir información de gestión. Las tres capas más altas representan las aplicaciones WS-Management. Los Perfiles DAHS se mapean sobre la pila de protocolos WS-Management utilizando WS-Management-CIM Binding.

DASH define además Modelos de Gestión, Eventos, Perfiles, Descubrimiento y Seguridad.

Un beneficio clave de la iniciativa DASH es que entrega una gestión de sistemas de escritorio y móviles más segura ya que incluye el cifrado de capa de red y de transporte, mecanismos de autenticación y autorización estándar de la industria, y establece Perfiles estándar para roles, autorización y gestión de cuentas.

Finalmente, DASH permite controlar potencia, el arranque, obtener información de la versión del firmware, información del hardware como CPU, memoria, ventiladores, fuentes de potencia y sensores, etc. de sistemas de escritorio y móviles.

Este estándar se lanzó a comienzos de este año por tanto aún no existen implementaciones.

1.1.8. WS-Managment. La Gestión con Servicios Web, WS-Management, es una especificación del DMTF.

WS-Management [21] describe un método basado en SOAP para gestionar sistemas tales como computadores, servidores, dispositivos, Servicios Web y otras aplicaciones, y entidades gestionables.

La arquitectura de Servicios Web está basada en un conjunto de especificaciones que definen funciones que además se pueden integrar para cumplir diferentes requerimientos de servicios.

Una importante aplicación de estos servicios se encuentra en el área de gestión de sistemas. Para promover interoperabilidad entre aplicaciones de gestión y recursos gestionados, esta especificación identifica un conjunto núcleo de especificaciones y requerimientos de uso de Servicios Web que exponen un conjunto de operaciones para la gestión de sistemas que permiten:

- Descubrir recursos de gestión.
- Manipular recursos de gestión es decir, crear, eliminar, renombrar, obtener, establecer.
- Enumerar el contenido de logs o tablas.
- Subscribirse/desubscribirse a eventos de recursos.
- Ejecutar métodos de gestión específicos.

Para cada uno de estos aspectos, esta especificación define los requerimientos mínimos, pero una implementación se puede extender más allá de este conjunto de operaciones y también puede decidir no soportar uno o más de ellas si la funcionalidad no es apropiada para el dispositivo o sistema objetivo.

Esta especificación busca cumplir los siguientes requerimientos:

- Restringir los protocolos y formatos de Servicios Web de tal forma que se puedan implementar fácilmente en servicios de gestión tanto de hardware como de software.
- Definir requerimientos mínimos sin limitar las implementaciones.
- Asegurar la integración con otras especificaciones de Servicios Web.
- Minimizar mecanismos adicionales aparte de la arquitectura de Servicios Web actual.

En WS-Management se emplea la siguiente terminología:

Cliente. Una aplicación que utiliza Servicios Web para acceder el servicio de gestión.

Servicio. Una aplicación que proporciona servicios de gestión a clientes al exponer Servicios Web. Generalmente, un servicio es equivalente al oyente de la red, está asociado con una dirección de transporte física y es esencialmente un tipo de punto de acceso de gestión.

Recurso gestionado. Es una entidad que puede ser de interés para un administrador. Puede ser un objeto físico tal como un computador, una impresora o una entidad abstracta tal como un servicio.

Clase de recurso. Es una representación abstracta, es decir, un tipo de recurso gestionado. Define la representación de operaciones y propiedades de gestión. Un ejemplo de una clase de recurso es la descripción de operaciones y propiedades para un conjunto de computadores.

Instancia de un recurso. Es una instancia de una clase de recurso. Un ejemplo es el conjunto de operaciones y valores de propiedades de gestión para un computador específico.

Selector. Un par nombre valor relativo a un recurso que actúa como un discriminante a nivel de instancia. Es esencialmente un filtro o clave que identifica la instancia deseada de un recurso.

La relación de servicios a clases e instancias de recursos es la siguiente:
- Un servicio consiste de una o más clases de recursos.
- Una clase de recurso puede contener cero o más instancias.
- Si existe más de una instancia para una clase de recurso, las instancias se identifican a través de partes de la dirección SOAP del recurso.

WS-Management utiliza un modelo de direccionamiento, WS-Addressing, y además tres modelos de transferencia de datos que son WS-Transfer, WS-Enumeration y WS-Eventing.

WS-Addressing. Permite identificar y acceder recursos. WS-Adressing define un formato de referencia que utiliza una Referencia de Punto Final, EPR (EndPointReference).

WS-Transfer. Permite manejar representaciones de recursos. WS-Transfer se basa en SOAP y establece las siguientes operaciones:
a. Get. Obtiene una representación de un recurso.
b. Put. Actualiza un recurso al proporcionar una representación de reemplazo.
c. Create. Crea un recurso y proporcionar su representación inicial.
d. Delete. Elimina un recurso.
e. Además, WS-Management define la operación de renombrado.

WS-Enumeration. Permite realizar una enumeración de recursos. WS-Enumeration se basa en SOAP y define las siguientes operaciones:
a. Enumerate. Inicia una enumeración.
b. Pull. Obtiene ramificaciones de una enumeración.
c. Release. Cancela una enumeración.

WS-Eventing. Permite que un Servicio Web reciba mensajes acerca de eventos de otro Servicio Web que le interesan. WS-Eventing define las siguientes operaciones: Subscribe, Renew, GetStatus, Unsubscribe y SubscriptionEnd.

WS-Management-CIM Binding. Define la relación entre la representación de Servicios Web de CIM y WS-Management. WS-Management-CIM Binding define:
- Utilizar direccionamiento basado en WS-Addressing para identificar y acceder objetos CIM.
- Obtener y actualizar instancias de una clase utilizando WS-Transfer.
- Enumerar instancias de clases utilizando WS-Enumeration.
- Ejecutar operaciones utilizando operaciones WS-Management equivalentes.

Como se mencionó anteriormente, se planea utilizar este estándar con WBEM, lo que potenciaría aún más ambas tecnologías.

1.1.9. WSDM. La Gestión Distribuida con Servicios Web, WSDM (Web Services Distributed Management) es un estándar de OASIS (Organization for the Advancement of Structured Information Standards).

WSDM [22] busca unificar infraestructuras de gestión al proporcionar una infraestructura neutral al vendedor, plataforma, red y protocolos para permitir que tecnologías de gestión accedan y reciban notificaciones de recursos gestionables. Aunque se basa en un conjunto estandarizado de especificaciones XML, proporciona características para gestionar recursos que otras tecnologías de gestión propietarias no, y se puede utilizar para estandarizar la gestión de muchos dispositivos, desde dispositivos de gestión de red hasta dispositivos electrónicos, tales como televisores, reproductores de video digital y PDAs.

La tecnología de Servicios Web fue diseñada para resolver el problema de integración de aplicaciones, especialmente de aplicaciones desarrolladas con un conjunto heterogéneo de tecnologías y plataformas de implementación. La adopción de estándares de Servicios Web abiertos creó una oportunidad para que la comunidad de gestión de sistemas promoviera estas tecnologías para la integración de aplicaciones de gestión utilizadas para gestionar recursos heterogéneos. Al utilizar Servicios Web, la infraestructura de gestión de sistemas es neutral al vendedor, independiente de la plataforma y permite el uso de un protocolo de mensajería común entre un recurso gestionable y un gestor, y entre gestores, y justamente esto es lo que especifica WSDM.

WSDM fue desarrollado con base en Servicios Web y una Arquitectura Orientada al Servicio, SOA (Service Oriented Architecture). WSDM es una especificación y un conjunto de especificaciones, MUWS y MOWS, para gestionar dispositivos así como también Servicios Web utilizando Servicios Web, los cuales son inherentemente dependientes de SOA. El estándar WSDM especifica cómo la capacidad de gestión de un recurso se hace disponible a gestores a través de Servicios Web.

El foco de la arquitectura WSDM es un recurso gestionable que se debe representar como un Servicio Web. En otras palabras, la información de gestión referente a un recurso debe ser accesible a través de un Punto Final de Servicio Web que para proporcionar acceso a un recurso debe ser referenciado por una Referencia de Punto Final, o EPR, como está definido en el estándar WS-Addressing. Los Puntos Finales que soportan acceso a recursos gestionables se llaman Puntos Finales de Gestión cuya implementación debe obtener y manipular la información relacionada con un recurso gestionable.

El EPR proporciona el recurso gestionable al que el gestor envía mensajes mientras que el recurso gestionable puede enviar notificaciones de eventos a un gestor siempre que el gestor se haya subscrito para recibir notificaciones. Así, WSDM cubre tres modos de interacción entre un recurso gestionable y un gestor. Estos modos de interacción son los siguientes:

- Un gestor puede obtener información de gestión acerca del recurso gestionable. Por ejemplo, el gestor puede obtener el estado operativo actual del recurso gestionable o el estado actual de un proceso ejecutándose en el recurso gestionable.

- Un gestor puede cambiar el estado de algún recurso gestionable al cambiar su información de gestión.
- Un recurso gestionable puede informar o notificar a un gestor sobre un evento. Este modo de interacción requiere que el gestor se suscriba par recibir eventos sobre un aspecto deseado.

El lenguaje XML es fundamental para la pila WSDM. Todos los mensajes WSDM se serializan y transportan como documentos XML cuyo formato se define rigurosamente a través de un Esquema XML y los respectivos estándares. WSDM también utiliza Esquemas XML para definir partes críticas del intercambio de mensajes entre un recurso gestionable y un gestor.

La especificación WSDM se basa en dos tecnologías de Servicios Web básicas, SOAP y WSDL, y como se mencionó anteriormente, ya que un componente clave de la arquitectura WSDM es la Referencia de Punto Final, EPR (Endpoint Referente), la especificación WSDM también se basa en el estándar WS-Addressing.

WSDM utiliza WSDL para describir la interfaz proporcionada por un Punto Final de Gestión. El siguiente nivel de la pila de la tecnología WSDM consiste de un conjunto de estándares OASIS que especifican cómo representar y acceder una representación XML de un recurso que puede ser un recurso gestionable. Este conjunto de estándares generalmente se llama WS-ResourceFramework.

El conjunto de estándares WS-Framework consiste de los siguientes estándares:
- La especificación WS-Resource. Especifica la noción básica de un recurso.
- La especificación WS-ResourceProperties. Define el documento de propiedades de un recurso.
- La especificación de WS-ResourceLifetime. Especifica la interfaz para destruir un recurso.
- La especificación WS-ServiceGroup. Define cómo se pueden agrupar recursos.
- La especificación WS-BaseFaults. Define un formato estándar para mensajes de fallas.

Un conjunto final de las especificaciones OASIS definen los Patrones de Intercambio de Mensajes, MEPs (Message Exchange Patterns) que se utilizan en las suscripciones y notificaciones. Estas especificaciones son:
- La especificación WS-Topics. Define cómo los aspectos en los que se basan las suscripciones y notificaciones se representan en XML.
- La especificación WS-BaseNotification. Define la estructura básica de los mensajes de suscripción y notificación.
- La especificación WS-BrokeredNotification. Define una arquitectura avanzada para subscripciones y notificaciones sobre Servicios Web que involucra un receptor/distribuidor de mensajes de tercera parte.

WSDM consiste de dos estándares conocidos como Gestión Utilizando Servicios Web, MUWS (Management Using Web Services) y Gestión de Servicios Web, MOWS (Management of Web Services).

El estándar MUWS trata los mecanismos básicos y MEPs para gestionar cualquier recurso gestionable utilizando Servicios Web como la plataforma para el intercambio de mensajes. MUWS está especificado en dos partes:

- MUWS parte 1, contiene las capacidades fundamentales de un recurso gestionable que son identidad de un recurso, características de gestión y propiedades correlacionables. Además, proporciona una discusión del Formato de Eventos WSDM, WEF (WSDM Event Format).
- MUWS parte 2 contiene el conjunto de capacidades de gestión WSDM adicionales a las cubiertas en MUWS parte 1.

Tanto MUWS parte 1 como MUWS parte 2 tienen un Esquema XML y una definición WSDL. Estos documentos se deberían incorporar en la definición de un recurso gestionable para hacer uso de las facilidades MUWS.

El estándar MOWS define la gestión de un Servicio Web en sí mismo. El estándar MOWS puede ser visto tanto como una aplicación y como una extensión del estándar MUWS. En MOWS, un Servicio Web es el recurso gestionable. Un Servicio Web así como también cada proceso que comprende el servicio tienen un estado expresado por un Modelo de Estado como está definido por el Ciclo de vida del Servicio Web, Service Lifecycle (WSLC). Así, un Servicio Web puede ser gestionado por Servicios Web. Además de utilizar las capacidades y características de gestión del estándar MUWS, el estándar MOWS introduce sus propias capacidades de gestión y extiende varias capacidades MUWS para acomodar requerimientos especiales para gestionar un Servicio Web.

Como se mencionó anteriormente, se planea utilizar este estándar con WBEM, lo que potenciaría aún más ambas tecnologías.

No se puede terminar esta sección sin mencionar que además de estos estándares se estudiaron otros como SNMP [23] y JMX [24], pero no se incluyeron en este documento por ser bastante conocidos en el mundo de la gestión y para lograr que este documento no fuera muy extenso y denso.

1.2. ENTORNOS HETEROGÉNEOS

En las redes de comunicaciones actuales se utilizan diferentes tecnologías de comunicaciones, tanto cableadas como inalámbricas, que tienen una gran cantidad de características y difieren entre sí en varios aspectos, como por ejemplo, velocidad, ancho de banda y alcance máximo. Además, en estas redes de comunicaciones, los equipos tienen hardware y software con múltiples y variadas características que hacen que los entornos de comunicaciones actuales sean realmente heterogéneos. Estas redes de comunicaciones interoperan para proporcionar los servicios requeridos por los usuarios, es decir, que entre estos múltiples entornos heterogéneos se crea una verdadera simbiosis para ofrecer servicios de datos, voz y video a los usuarios actuales que cada vez exigen más y mejores servicios.

Estas redes de comunicaciones constituyen una gran red de redes que se conoce como Internet, la cual permite la comunicación en cualquier momento y lugar, y día tras día se vuelve más grande y compleja, y cobra mayor importancia en los ámbitos de negocios, educación, medicina, y en general, en la vida diaria de todos y cada uno de los habitantes del planeta que desean estar comunicados. Incluso, un poco más allá de Internet, se encuentra una nueva iniciativa que está ganando el interés de grupos de investigación y desarrollo, así como también de las empresas, que se conoce como Computación Grid.

1.2.1. Grid. Esta nueva tecnología se caracteriza por entornos heterogéneos, el eje central de este trabajo, por lo que resulta importante conocer un poco más acerca de ella, pero ya que es reciente, en estos momentos no existe una única definición, cada grupo la define desde su perspectiva de trabajo, por tanto, continuación se encuentran las definiciones de algunos de los grupos más importantes que están interesados en esta nueva iniciativa.

CERN. Para CERN (European Organization for Nuclear Research) [25], la Web permite compartir información sobre Internet mientras que Grid permite compartir potencia computacional y capacidad de almacenamiento de datos sobre Internet. Grid va más allá de una simple comunicación entre computadores y busca finalmente convertir la red de computadores global en un vasto recurso computacional.

Grid es un trabajo en progreso ya que la tecnología subyacente todavía está en una fase de prototipo y está siendo desarrollada por cientos de investigadores e ingenieros de software alrededor del mundo. Grid está atrayendo mucho interés debido a su futuro que aunque todavía es incierto es potencialmente revolucionario.

Existe un interés en curso por definir estándares Grid, y como resultado, el Foro Grid Global, GGF (Global Grid Forum), actualmente el Foro Global Abierto, OGF (Open Grid Forum) ha creado la Arquitectura de Estándares Grid Abierta, OGSA [26] (Open Grid Standards Architecture) que es soportada tanto por una gran parte de la comunidad científica como crecientemente por la industria.

Lo que OGSA está tratando de hacer, básicamente, es armonizar el trabajo en curso para desarrollar el Globus Toolkit, una iniciativa principalmente académica que utiliza servicios Web, los cuales están siendo apoyados por la industria para proporcionar un estándar común para servicios ofrecidos sobre www. En la práctica, OGSA está siendo respaldada por el grupo académico detrás del Globus Toolkit en colaboración con IBM. La visión actual es que los servicios Grid, en la práctica, solo serán una subclase de servicios Web pero que darán acceso a la clase de potencia computacional que los Grids permiten.

Mucha de la actividad alrededor de Grid en estos días está en estándares abiertos, los cuales se necesitan para asegurar que el mundo de la investigación y desarrollo pueden contribuir en un modo constructivo a Grid, y que la industria está preparada para invertir en el desarrollo de servicios e infraestructuras Grid comerciales.

Muchos expertos imaginan una realidad con Grids dedicadas, cada una para una comunidad o un conjunto de comunidades diferentes, con sus propios requerimientos y objetivos. Tales comunidades se llaman Organizaciones Virtuales, VO (Virtual Organizations), y parecen ser la mejor alternativa para asegurar la explotación exitosa de Grid.

GridComputing. Para GRID Infoware o Grid Computing Information Centre [27], Grid es un tipo de sistema paralelo y distribuido que permite la compartición, selección y agregación de servicios de recursos heterogéneos, autónomos, distribuidos geográficamente a través de múltiples dominios administrativos, en tiempo de ejecución, con base en su disponibilidad, capacidad, desempeño, costo y requerimientos de calidad de servicio de los usuarios.

Los Grids presentan los diferentes recursos como un solo recurso unificado para resolver aplicaciones computacionales a gran escala e intensivas en datos. La idea es análoga a la red de potencia eléctrica, Grid, donde los generadores de potencia se distribuyen, pero los usuarios son capaces de acceder potencia eléctrica sin preocuparse por la fuente de energía y su localización.

Los Grids buscan explotar las sinergias que resultan de la cooperación, en otras palabras, la habilidad para compartir y agregar capacidades computacionales distribuidas y entregarlas como un servicio.

Como cualquier sistema distribuido, los Grids necesitan resolver varios problemas y desafíos incluyendo: seguridad, autonomía, heterogeneidad de interfaces de acceso a recursos, políticas, capacidad, precio, localización de datos, variación dinámica en la disponibilidad de recursos y complejidad en la creación de aplicaciones. Por lo tanto, Grid sigue una combinación de arquitectura jerárquica y descentralizada para la gestión de recursos, y una arquitectura en capas para la implementación de varios servicios.

IBM. Para IBM [28], la computación Grid habilita la virtualización de recursos computacionales y de datos distribuidos tales como procesamiento, ancho de banda de la red y capacidad de almacenamiento para crear una sola imagen de sistema, concediendo a usuarios y aplicaciones acceso a vastas capacidades. Como un usuario de Internet ve una instancia unificada del contenido a través de la Web, un usuario Grid esencialmente ve un solo computador virtual grande.

En su núcleo, la computación Grid se basa en un conjunto de protocolos y estándares abiertos, por ejemplo, OGSA, que habilita la comunicación a través de entornos heterogéneos, dispersos geográficamente. Con la computación Grid, las organizaciones pueden optimizar los recursos computacionales y de datos, utilizarlos para cargas de trabajo de gran capacidad, compartirlos a través de redes y permitir la colaboración.

Finalmente, Ian Foster, un investigador reconocido internacionalmente y un líder en el área de la computación Grid define a Grid como un sistema que cumple con la siguiente lista de requerimientos:
1. Coordina recursos que no están sujetos a control centralizado.
2. Utiliza interfaces y protocolos estándar, abiertos y de propósito general.
3. Entrega calidades de servicio no triviales.

Teniendo en cuenta que un Grid se puede considerar como un entorno computacional heterogéneo, y que el tema principal de este trabajo, es la gestión de entornos heterogéneos, es muy interesante conocer la forma en la que se realiza la gestión Grid.

1.2.2. Gestión Grid. Los Grids, como cualquier entorno computacional, requieren algún grado de gestión de sistemas. La gestión en Grids [29] es una tarea potencialmente compleja dado que los recursos son a menudo heterogéneos, distribuidos y a través de múltiples dominios de gestión. El modelo computacional Grid enfrenta todos los problemas de gestión tradicionales y también tiene nuevos desafíos, no solamente por la gestión de sus recursos, sino también por la gestión del Grid en sí mismo.

La gestión de sistemas efectiva es posible solamente si los recursos son gestionables y si existen las herramientas para gestionarlos. Actualmente, los administradores de sistemas pueden escoger entre una amplia variedad de herramientas de gestión de diferentes

vendedores de sistemas, proveedores de tercera parte y la comunidad de fuente abierta. Sin embargo, estas herramientas tienden a operan independientemente y a usar interfaces y protocolos propietarios para gestionar un conjunto de recursos limitado, lo que hace más difícil para una organización construir un sistema de gestión eficiente y bien integrado. Este problema se está resolviendo a través del desarrollo de estándares de gestión que permiten a las herramientas de gestión monitorea y controlar recursos de una manera uniforme e interoperar con las otras. A su vez, esto permite que los administradores de sistemas escojan sus herramientas y proveedores de gestión quienes conocen que sin importar el origen de las herramientas, ellas pueden trabajar cooperativamente en un entorno de gestión integrado.

Gestión en OGSA. La base de gestión en un Grid OGSA es la especificación WSDM MUWS. Esto significa que para que un recurso sea gestionable, debe proporcionar el conjunto mínimo de capacidades de gestión especificadas por MUWS.

A continuación se enumera los principales requerimientos para gestión en OGSA. Estos requerimientos son especialmente importantes en un entorno de gran escala, distribuido sin noción centralizada de control, tal como un Grid: escalabilidad, interoperabilidad, seguridad, confiabilidad, políticas, monitoreo de desempeño, requerimientos de gestión peer to peer.

En un Grid OGSA existen tres tipos de gestión que involucran los recursos:

- Gestión de los recursos en sí mismos (por ejemplo, reiniciar un equipo, o establecer particiones en un switch).
- Gestión de los recursos Grid (por ejemplo, reservar recursos, monitorear y controlar).
- Gestión de la infraestructura OGSA, la cual en sí misma está compuesta de recursos (por ejemplo, monitorear un servicio de registro).

Diferentes tipos de interfaces realizan estos tipos de gestión. Estas interfaces se pueden categorizar en tres niveles mostrados en la columna del medio de la tabla 2 y también al lado derecho de la figura 10.

Tabla 2. Tipos de gestión en OGSA

Tipo de gestión	Nivel de interfaz	Interfaz
Gestión de los recursos en sí mismos	Nivel de recurso	WBEM, SNMP, etc.
	Nivel de infraestructura	WSRF, WSDM, etc.
Gestión de recursos en el Grid	Nivel de funciones OGSA	Interfaces funcionales
Gestión de la infraestructura OGSA		Interfaces de gestión específicas

En la figura 10 se muestran los niveles de gestión en OGSA. Las interfaces se muestran en pequeños círculos.

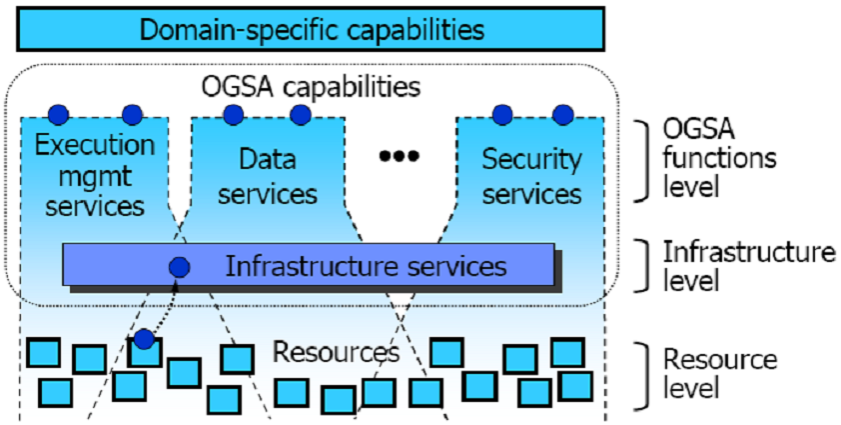


Figura 10. Niveles de gestión en OGSA [29]

Como se observa en la figura 10, a nivel de recurso, éstos se gestionan directamente a través de sus interfaces de gestión nativas. Para recursos discretos, éstas son usualmente SNMP, CIM/WBEM, JMX, o interfaces propietarias. La gestión a este nivel involucra monitoreo (por ejemplo, obtener el estado del recurso, utilizar técnicas tales como notificación de eventos), configuración y control (por ejemplo, establecer el estado del recurso) y descubrimiento.

El nivel de infraestructura proporciona el comportamiento de gestión básico de los recursos, lo que forma la básica de gestión en un entorno OGSA. La estandarización de este comportamiento de gestión base se requiere para integrar el vasto número y tipos de recursos, y el conjunto más limitado de gestores de recursos, que son introducidos por múltiples proveedores. El nivel de infraestructura proporciona:

- El modelo de gestión básico, que representa recursos como servicios y permite que los recursos en OGSA sean manipulados a través de servicios Web estándar para descubrimiento, acceso, etc. Este modelo permite a los recursos llegar a ser gestionables al menos en un grado mínimo, al habilitar descubrimiento, terminación, introspección, monitoreo, etc.
- Funcionalidad básica que es común a las capacidades OGSA, por ejemplo, interfaces para capacidades que son comunes a muchos recursos (por ejemplo, start, stop, etc.), representación del estado de un recurso, modos para describir y descubrir relaciones entre recursos, incluyendo los tipos de relaciones (contiene, usa, etc.), notificaciones.
- Una interfaz de gestión genérica que es común a todos los servicios que implementan capacidades OGSA. Esta interfaz de gestión tiene funcionalidades tales como introspección, monitoreo, y creación y destrucción de instancias de servicios.

A nivel de funciones OGSA existen dos tipos de interfaces de gestión, denotadas por los dos círculos sobre cada una de las capacidades mostradas en la figura 10.

- **Interfaz funcional:** algunas capacidades OGSA comunes (tales como gestión de trabajos) son una forma de gestión de recursos. Los servicios que proporcionan estas capacidades las exponen a través de interfaces funcionales.
- **Interfaz de gestión:** cada capacidad tiene una interfaz de gestión específica a través de la cual la capacidad es gestionada (por ejemplo, monitoreo de registros, monitoreo de un gestor de trabajos, etc.). Esta interfaz podría extender la interfaz de gestión genérica al adicionar cualquier interfaz de gestión que sea específica para la gestión de esta capacidad.

Las interfaces tanto en los niveles de infraestructura como de funciones OGSA son proporcionadas por servicios, pero tienen una naturaleza diferente. En el nivel de infraestructura, los servicios son un wrapper alrededor de la gestión de un recurso (principalmente las semánticas del modelo del recurso) que proporciona un medio para acceder esta gestión. En el nivel de funciones OGSA, los servicios proporcionan funcionalidad a un nivel más alto que las características de los recursos, o proporcionan interfaces que no acceden un modelo de recurso.

Finalmente, la división en niveles permite definir su interoperabilidad al definir interfaces claras entre ellos. Mientras es posible construir servicios (implementando capacidades OGSA) que puenteen estos niveles (por ejemplo, utilizando un adaptador propietario en un recurso que proporcione datos directamente al servicio), eso no es deseable desde el punto de vista de interoperabilidad, debido, a que por ejemplo, limita las clases de recursos con las que el servicio y el adaptador son compatibles.

En conclusión, la gestión Grid se basa en niveles y en estándares abiertos, por tanto, estos dos aspectos se tuvieron en cuenta para definir la arquitectura de gestión para entornos heterogéneos. La gestión Grid plantea la utilización de estándares como SNMP, WBEM, WSDM, etc, los cuales se tuvieron en cuenta para el desarrollo de este trabajo lo que significa que se escogió un camino de investigación adecuado.

1.2.3. Gestión de entornos heterogéneos. Existe un gran interés de grupos de investigación y desarrollo así como también por empresas por conseguir la gestión de entornos heterogéneos entonces a continuación de describen brevemente los esfuerzos que han realizado hasta ahora por lograrlo.

Consorcio de Gestión Abierta. El Consorcio de Gestión Abierta, OMC (Open Management Consortium) [30] se conformó en el 2006 y promueve los beneficios ofrecidos por las tecnologías de fuente abierta y de estándares abiertos, además proporciona un foro para la colaboración en el desarrollo de un producto entre proyectos de gestión de fuente abierta.

El Consorcio está compuesto por proyectos de fuente abierta líderes que están desarrollando y conformando las tecnologías utilizadas para gestionar infraestructura, particularmente infraestructura de fuente abierta tal como la que se encuentran en Linux.

La gestión de sistemas de fuente abierta reemplaza la tendencia limitada a un solo vendedor con una aproximación modular. Se selecciona lo que se necesita, se lo personaliza a las especificaciones exactas y se lo complementa conforme las necesidades cambian. Debido a que los productos de fuente abierta reciben contribuciones de usuarios, socios y otras terceras partes, se deben basar en estándares. Esta aproximación basada en estándares facilita la interoperabilidad entre soluciones de fuente abierta a través del ciclo de vida de gestión de sistemas.

El Consorcio busca utilizar estándares abiertos para la gestión de sistemas dentro de la industria, asistiendo a gestores y administradores de sistemas para personalizar o crear soluciones que satisfagan mejor sus necesidades de negocio. Uno de los primeros proyectos en marcha en estos momentos, ya que el consorcio es muy joven, involucra el desarrollo de protocolos para gestionar componentes de infraestructura, incluyendo información acerca de servidores, dispositivos de almacenamiento, configuraciones, modelos de red, middleware, aplicaciones y otros datos relevantes, para crear una

aproximación unificada de gestión de sistemas para vendedores y proyectos de fuente abierta. Además, el Consorcio también se encarga de diseñar varios caminos de integración para intercambiar datos con sistemas propietarios.

Actualmente, los miembros del OMC son: Alterpoint, Altinity, Artica Soluciones Tecnologicas, Ayamon, LLC, Babel Enterprise, BigSister, Elegant Software, Emu Software, Enomalism, Enomaly, Hyperic, Informed Control, INGSOFT, jbmc Software, Nagios, NetDirector, NodeDirector, NX Engineering, Open Country, Opengear, openSIMS, openQRM, Optena, Pandora, PING, Qlusters, Inc., Reductive Labs, Spread SUse.net, Symbiot, Inc., synetics, uXcomm.com, Webmin, X-tend, Zenoss. Cada uno de estos grupos de investigación ha hecho esfuerzos increíbles en el campo de la gestión que trabajando en conjunto obtendrán lo mejor de todos ellos para lograr una solución adecuada.

Gestión Abierta con CIM. Gestión Abierta con CIM, OMC (Open Management with CIM) [31] es un proyecto paraguas de fuente abierta que incluye implementaciones de perfiles de gestión y especificaciones de estándares abiertos definidos por el DMTF, herramientas y utilidades asociadas, y otros proyectos relacionados con CIM/DMTF. Además, cualquier implementación con estándares abiertos que utiliza CIM, encaja en el proyecto OMC. Es una iniciativa bastante joven, tanto que aún no se ha completado ninguna implementación.

El OMC es un proyecto con varios subproyectos. Como tal, utiliza varias licencias de fuente abierta. Incluso dentro de un proyecto, algunos archivos pueden tener una licencia más restrictiva que la licencia principal para el proyecto por varias razones. Además, cada subproyecto determina su propia línea de tiempo, cronograma, etc.

Todas las comunidades de fuente abierta exitosas comienzan con una visión fuerte, el soporte y liderazgo de un grupo pequeño. El proyecto OMC comenzó con ingenieros Novell que trabajaban para implementar los perfiles SMASH del DMTF, y hasta ahora, el OMC es patrocinado y mantenido por Novell y se espera que llegue a ser un verdadero esfuerzo de comunidad, con IHVs e ISVs contribuyendo con su experiencia para completar los perfiles del DMTF y para proporcionar herramientas y aplicaciones que los utilizan.

Los proyectos del OMC son:
- omcSmash
- omcBase
- omcCLP
- omcTools
- y más proyectos que están por ser lanzados

Además, los proyectos de fuente abierta relacionados con el OMC son:
- SBLIM
- CimomAbstract
- OpenWBEM

Actualmente el proyecto más desarrollado es omcSmash, que es una implementación de fuente abierta de SMASH del DMTF. En estos momentos, omcSmash incluye aproximadamente 20 perfiles que están en varios estados de desarrollo.

Entre los planes del OMC se encuentran:
- Completar omcSmash.
- Continuar desarrollando perfiles para omcSmash
- Mejorar omcCLP para trabajar out-of-band

Además, considera nuevos proyectos:
- Perfiles adicionales
- Consolas
- Utilidades de prueba
- APIs
- Clientes

Como se puede observar, el tema de gestión de entornos heterogéneos es muy reciente y es del interés de grupos de investigación y desarrollo que tienen iniciativas bastante jóvenes en las que aún falta trabajo por realizar, así como también de empresas como por ejemplo Avocent, lo que demuestra aún más la relevancia de este tema.

Avocent. Avocent [32] es un proveedor líder global de soluciones de gestión de infraestructura para centros de datos de empresas, negocios pequeños/medianos y oficinas sucursales. Con más de dos décadas de experiencia, innovaciones de productos y adquisiciones estratégicas, Avocent está posicionado como un líder internacional en tecnologías de avanzada digitales, embebidas y móviles.

Avocent participa activamente en el desarrollo de muchos estándares y especificaciones de la industria que integra en sus productos, es un miembro de la iniciativa ATCA, el DMTF y SAF.

Como el sucesor de la primera compañía de hardware en el mundo en soportar comercialmente un producto en plataformas de fuente abierta (Cyclades Corporation), Avocent entiende los beneficios técnicos y de negocios de impulsar el Software Libre y de Fuente Abierta, FOSS (Free and Open Source Software) y mantener una relación positiva con la comunidad FOSS. Avocent considera que bajo circunstancias específicas, ciertos tipos de software de propósito general (tales como kernels de sistemas operativos) se desarrollan mejor bajo un modelo de comunidad, bajando los costos de desarrollo y generando innovación que beneficia a los usuarios.

Avocent ofrece firmware y software IPMI, es decir herramientas de gestión de sistemas modulares que proporcionan acceso centralizado a dispositivos con IPMI. Avocent utiliza estándares de la industria tales como CLI de Cisco, CIM, SNMP, HPI, WMI, SOAP y XML lo que permite desarrollar utilidades de gestión específicas que se pueden expandir para acomodar tareas adicionales u opciones de integración del usuario. Para aquellas compañías que tienen poca o ninguna experiencia en gestión de firmware o software, Avocent también espera ofrecer paquetes de desarrollo que permitan a cualquier compañía ofrecer soluciones IPMI a una tasa acelerada.

Avocent además integra SMASH en sus productos embebidos para gestionar entornos servidor heterogéneos. La adición del nuevo estándar SMASH SM CLP responde a la demanda de los administradores por una Interfaz de Línea de Comandos Estándar para gestionar servidores desde cualquier parte en cualquier momento.

TOMAS. Ándago Ingeniería es una empresa dedicada a servicios de consultoría que trabaja en proyectos de gran envergadura y que utiliza tecnologías de Software Libre. Uno de esos proyectos es la creación de un estándar y una solución completa y abierta para la gestión efectiva de grandes redes de computadores con sistemas operativos *nix, principalmente Linux. Se trata del proyecto TOMAS³ [33]: Towards An Open Management Arquitecture for Systems, Software and Services, que se inició en el 2006.

El proyecto TOMAS³ pretende fomentar el uso del software libre de dos formas: primero, integrando en su arquitectura soluciones de código abierto, aunque no limitándose a éstas, y segundo basando la gestión del proyecto en herramientas de desarrollo colaborativo que permiten una libre y efectiva diseminación del conocimiento adquirido durante el desarrollo del proyecto a toda la comunidad.

Asimismo pretende avanzar en el desarrollo de estándares utilizándolos para el diseño de la arquitectura de TOMAS³, como el modelamiento por medio de CIM del DMTF y la aplicación de servicios Web vía WSDM de OASIS entre varios otros estándares.

Además de ello, la intención de Ándago con respecto a TOMAS³ es la de liberar el resultado del proyecto en el dominio público a fin de implicar en su desarrollo a una comunidad de desarrolladores y usuarios interesados en el mismo para su permanente soporte, extensión y mejora, haciendo el resultado del proyecto útil y duradero en el tiempo.

TOMAS³ es un proyecto PROFIT del Ministerio de industria, turismo y comercio y del Ministerio de educación y ciencia de España. Junto con Ándago están participando en el desarrollo de TOMAS³ las siguientes entidades: el Centro de Referencia Linux de la Universidad Autónoma de Madrid y el Laboratorio de Sistemas Distribuidos de la Universidad Politécnica de Madrid que proporcionan recursos de investigación además de los aportados por Ándago. Adicionalmente, algunas empresas privadas también han mostrado su interés en el proyecto.

La motivación del proyecto fue que una red de computadores tipo cluster puede contener una gran variedad de equipos, con distintas características hardware y con funciones muy dispares. Gestionar una red de computadores de esta complejidad es una tarea ardua y difícil. El mantenimiento de la red no sólo implica la gestión de los componentes individuales (actualizaciones, parches, etc.), sino también la gestión de las interdependencias entre los distintos componentes, lo que exige un orden preciso de las tareas a realizar. También es importante que la ejecución de las tareas de administración tenga en cuenta la integridad de los trabajos.

Distribuir la configuración entre las máquinas que componen una red también es un trabajo laborioso y delicado ya que puede ser diferente de una máquina a otra en un gran número de máquinas, y un mínimo error en una de ellas puede suponer una gran degradación o incluso la caída de todo el sistema. Por ello se hace necesario un sistema de gestión de la configuración que permita llevar un control preciso del estado de cada máquina.

Otra función compleja de administración es la monitorización de los equipos de la red, que debe ofrecer la posibilidad de escoger las métricas necesarias para poder obtener conocimiento del estado de la red y a su vez agrupar la información de varios ordenadores para tener una visión global del conjunto.

Una vez establecido un sistema fiable de monitorización también es deseable un sistema de alarmas que permitan la localización rápida de problemas, y más aún un sistema de tolerancia a fallos que permita ejecutar acciones que solucionen los problemas localizados de forma automática.

Normalmente, estas complejas operaciones de administración se realizan de manera manual, aunque se trata de procedimientos difíciles, caros, y muy propensos a errores, sobre todo en redes de gran tamaño. TOMAS³ viene a subsanar esta carencia.

Existen numerosas herramientas de ayuda para la instalación, configuración y mantenimiento de una red de computadores bajo Linux. Algunas de ellas son muy potentes pero no ofrecen soluciones completas sino parciales para las múltiples tareas de administración descritas. Otras son suficientemente generales, pero son productos propietarios y cerrados que no permiten auditorías, ampliación ni mejoras de su código. Por todo ello es deseable una solución de código abierto.

Entre las herramientas de software libre que proporcionan soluciones integradas de administración se encuentra Quattor [34]. Quattor proporciona un sistema de configuración de las máquinas centralizado, un repositorio de software común y un sistema de despliegue automático, cumpliendo varios de los requisitos especificados para TOMAS³. Por ello el desarrollo de TOMAS³ partirá de los elementos proporcionados por Quattor adaptándolos a las nuevas necesidades y añadiéndole todo el resto de funcionalidades especificadas usando otras soluciones.

TOMAS³ es tanto una aplicación middleware para la administración de medianos y grandes sistemas informáticos como la definición del conjunto de requisitos que ese sistema de administración centralizada debe cumplir. Además proporciona un framework para integrar las soluciones que ofrezcan las funcionalidades que cumplan dichos requisitos. Por tanto TOMAS³ no es una aplicación monolítica sino una infraestructura donde se integran diversas soluciones. Además de dicho framework, el proyecto TOMAS³ proporcionará un entorno completo que integrará un conjunto de soluciones seleccionadas y será perfectamente funcional.

La arquitectura global de TOMAS³ ofrece las siguientes funcionalidades:

- **Configuración.** Gestión y almacenamiento centralizado de la información de configuración de los equipos de la red, incluye información sobre el hardware, configuración del sistema y configuración de las aplicaciones,
- **Instalación.** Instalación inicial de los equipos, distribución e instalación de paquetes de software, y configuración y mantenimiento de los equipos de acuerdo a la información proporcionada por el subsistema de configuración,
- **Monitorización.** Recopilación, almacenamiento y consulta de la información sobre el estado actual de los equipos
- **Tolerancia a fallos.** Correlación de la información proporcionada por el subsistema de monitorización con una información patrón, y ejecución de acciones correctoras en caso de ser necesario,

- **Gestión de recursos.** Gestión de la distribución de la carga entre los elementos de computación de la red, proporcionando una capa de abstracción sobre los gestores locales.
- **Administración de servicios.** Posibilidad de desplegar, arrancar, reiniciar y parar servicios de forma controlada y remota en una o múltiples máquinas simultáneamente.
- **Administración de software.** Control del software desplegado en una o varias máquinas y la posibilidad de instalar o desinstalar software de forma remota llevando un inventario preciso del software contenido en cada máquina.
- **Administración de hardware.** Inventario preciso y actualizado de los componentes hardware de una red o un cluster.
- **Administración de usuarios.** Gestión de usuarios del sistema tanto para los sistemas administrados como para los propios recursos de administración de TOMAS³.

Las funcionalidades y las herramientas que las implementan se muestran en la figura 11.

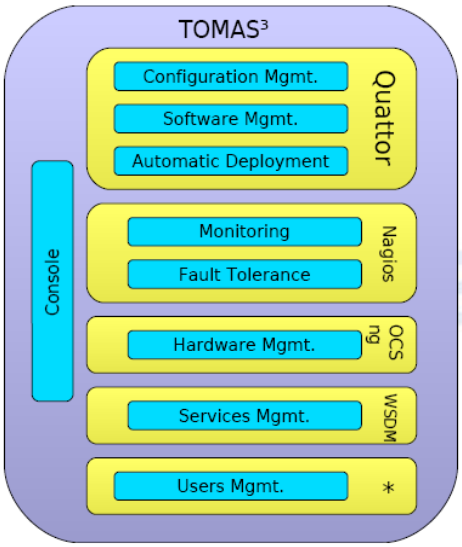


Figura 11. Funcionalidades y herramientas en TOMAS [33]

En la figura 11 se observa la herramienta Quattor, responsable de la gestión de configuración, de la gestión del software y del despliegue automático, así como también, otras herramientas y estándares a cargo de otras áreas de gestión. Todas las herramientas se acceden a través de una única consola.

Además, una vista más detallada de la forma en la que TOMAS³ integra aplicaciones de gestión mediante servicios Web se muestra en la figura 12.

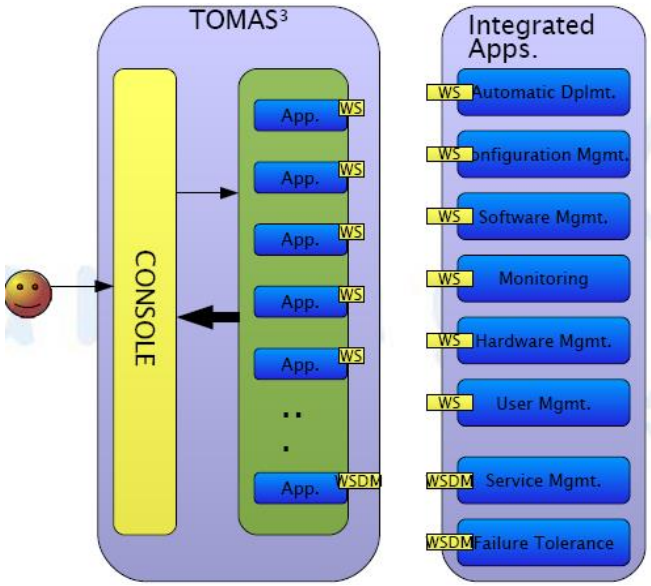


Figura 12. Integración de aplicaciones en TOMAS³ [33]

En la figura 12 se observa que la integración de las aplicaciones responsables del despliegue automático, de la gestión de configuración y de software, del monitoreo, de la gestión de hardware, de usuarios y servicios, y de la tolerancia a fallas se integran mediante servicios Web a TOMAS³.

En conclusión, TOMAS busca proporcionar una arquitectura abierta para la gestión autónoma de grandes sistemas distribuidos. Muchas compañías y organizaciones cuentan con redes de miles de computadores con software de negocios, con un gran grado de heterogeneidad, tanto en hardware como en software. La actualización y mantenimiento del software es un problema difícil en estas redes. Hacerlo manualmente es muy costoso. Sin embargo, no hay infraestructura middleware para ejecutar estas tareas automáticamente. Adicionalmente, los sitios están localizados en diferentes localidades geográficas y tienen restricciones de seguridad. La meta de TOMAS es precisamente crear un middleware de fuente abierta para resolver este problema. TOMAS se basa en los esfuerzos de estandarización reciente en el área de servicios Web para gestión distribuida, tal como WSDM de OASIS y está desarrollando una implementación de WSDM que integra con los proyectos de fuente abierta existentes, tales como Quattor, para crear una infraestructura middleware integrada que permita la automatización de actualización y mantenimiento de software en grandes redes, incluso se puede decir que este proyecto busca desarrollar una plataforma de fuente abierta para la gestión autónoma de grandes Grids de empresa.

2. INTEGRACIÓN DE TECNOLOGÍAS DE GESTIÓN

En este capítulo se describen las posibilidades de integración de los estándares de gestión de redes más importantes actualmente como SNMP, IPMI, HPI, AIS, WBEM, entre otros, y finalmente se describen alternativas para lograr la interoperabilidad de WBEM, uno de los estándares más utilizado hoy en día, y que se integra muy bien con muchos otros estándares de gestión.

2.1. IPMI Y CIM

La Especificación IPMI a CIM [35] es de Intel y define mapeos CIM para áreas específicas de la gestión de servidores. El propósito de esta especificación es asegurar interoperabilidad al lograr un mapeo cross-vendedor de IPMI a CIM. Adicionalmente, esta especificación define un modo consistente para conectar BMCs (Baseboards Management Controllers) que implementan interfaces de control IPMI a interfaces de control basadas en CIM.

A continuación se presenta el proceso general que el software del proveedor de instrumentación utilizaría para acceder un subsistema de gestión de plataforma basado en IPMI, descubrir sus capacidades de gestión disponibles y mapear esas capacidades a objetos CIM. Los pasos que se presentan a continuación son una guía para el diseño de software y para aplicar las especificaciones IPMI.

1. Descubrir y conectarse a la interfaz IPMI del sistema gestionado particular. La instrumentación que mapea de IPMI a CIM primero debe descubrir y conectarse al subsistema de gestión de plataforma IPMI a través de una de las interfaces IPMI. Las interfaces IPMI se pueden clasificar en tres tipos, de Sistema, Remotas e Internas. Las interfaces de Sistema son KCS (Keyboard Controller Style), BT (Block Transfer), SMIC (Server Management Interface Chip) y SSIF (SMBus System Interface). Las interfaces remotas son serial/módem y LAN (Ethernet). Las interfaces internas son IPMB (Intelligent Platform Management Bus), SMBus PCI e ICMB (Intelligent Chasis Management Bus), que se comentaron en el capítulo 1.

2. Descubrir y enumerar la información y asociaciones de Entidades, Sensores y FRUs IPMI. Una vez el software es capaz de acceder a IPMI, el siguiente paso principal del proceso de mapeo es que el software descubra y enumere las capacidades IPMI de la plataforma particular de tal forma que se puedan mapear a CIM. Esto incluye descubrir información de sensores y FRUs (Field Replaceable Unit), y las asociaciones entre ellos para las entidades que son gestionadas a través de IPMI.

Esto proceso de descubrimiento y enumeración se puede pensar como la creación de una lista lógica de capacidades IPMI para una plataforma dada. Una vez esta lista lógica se crea, se puede utilizar para dirigir la instanciación de objetos CIM para acceder esas capacidades.
Los SDRs (Sensor Data Records) son un elemento clave para la enumeración y descubrimiento de información de sensores y FRUs IPMI, y las entidades con las que ellos se asocian.

3. Instanciar objetos y asociaciones CIM. Una vez que se tiene la lista lógica de los sensores, entidades y FRUs IPMI disponibles, el siguiente paso es instanciar objetos CIM. Esto se realiza de acuerdo a la información de tipo de sensor dado en el SDR, al ID de la Entidad, y la información FRU IPMI, además se crean las asociaciones CIM que relacionan unos objetos con otros.

La Especificación define diferentes mapeos que permiten gestionar elementos IPMI a través de CIM.

- **Mapeo de Sensor IPMI.** Permite gestionar Sensores IPMI.
- **Mapeo de Logs IPMI.** Permite gestionar SELs (System Event Log) IPMI.
- **Mapeo de Comandos IPMI.** Permite ejecutar un subconjunto de comandos IPMI.
- **Mapeo de Control de Potencia IPMI.** Permite gestionar el estado de potencia de un sistema gestionado y el BMC.
- **Mapeo de Gestión de IDs de Usuario IPMI.** Permite gestionar IDs de Usuario, passwords y privilegios del BMC.
- **Mapeo de BMC IPMI.** Permite modelar y gestionar un BMC, así como también el sistema gestionado en el que está embebido el BMC.
- **Mapeo de Versión IPMI.** Permite modelar la versión del mapeo que se está implementado.
- **Mapeo de Entidad IPMI.** Permite modelar entidades físicas y dispositivos lógicos correspondientes descubribles a través de la interfaz IPMI.
- **Mapeo de Watchdog IPMI.** Permite gestionar el watchdog IPMI.
- **Mapeo de Identidad Software IPMI.** Permite representar software y firmware (incluido el firmware del BMC).

Y para cada uno de estos mapeos proporciona: diagrama de clases, ejemplos de instancias, diagramas de instancias, requerimientos de mapeo y describe detalladamente las clases CIM utilizadas.

A manera de ejemplo, en la figura 13 se muestra el diagrama de clases en el que se observan las diferentes clases CIM que se pueden utilizar para mapear Sensores IPMI a CIM, y en la figura 14 se muestra el diagrama de instancias de un sensor de presencia de voltaje discreto (el voltaje es bueno o malo) y un sensor de temperatura analógico (retorna un valor en un rango numérico).

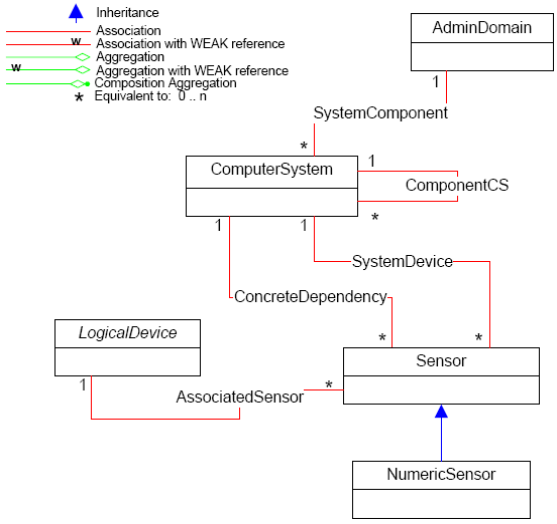


Figura 13. Diagrama de clases Sensores IPMI [35]

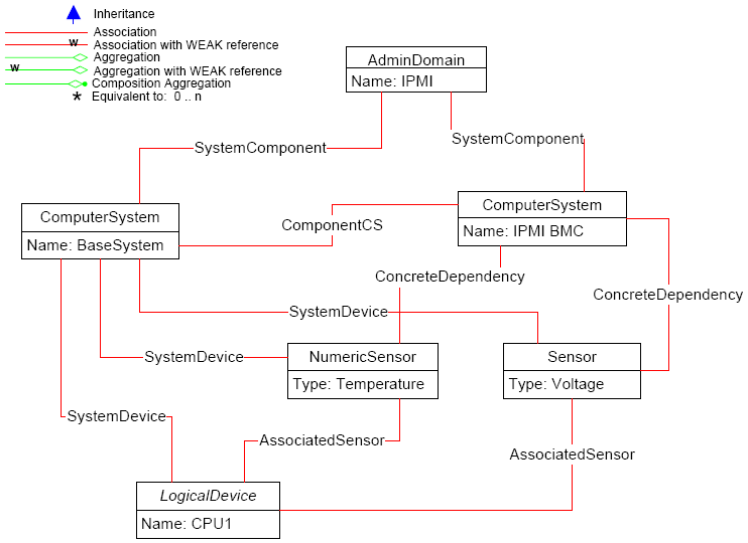


Figura 14. Diagrama de instancias de un sensor de voltaje y de temperatura [35]

2.2. IPMI Y OTROS ESTÁNDARES DE GESTIÓN

IPMI es una especificación de interfaz a nivel de hardware que es neutral al software de gestión y proporciona funciones de monitoreo y control que se pueden exponer a través de interfaces software de gestión estándar tales como DMI, WMI, CIM, SNMP, etc. Como una interfaz a nivel de hardware se encuentra en la parte baja de una pila de software de gestión como se muestra en la figura 15.

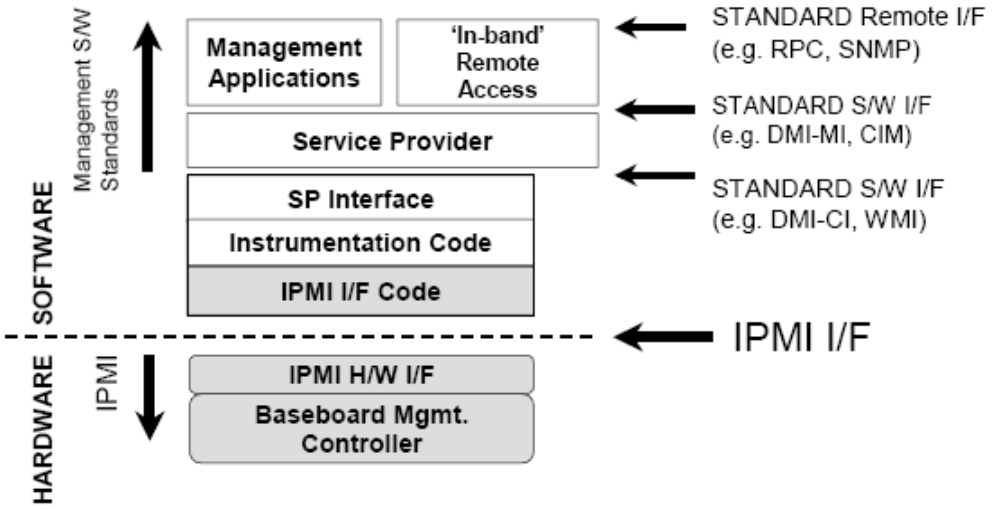


Figura 15. IPMI y otros estándares de gestión [1]

Por otro lado, el Formato Estándar de Alertas, ASF [36] (Alert Standar Format) es un estándar del DMTF que permite la gestión pre-inicio o pre-sistema operativo y sistema operativo ausente. ASF utiliza Traps de Eventos de Plataforma, PET (Platform Event Trap), que permite generar Traps SNMP, para la generación y envío de alertas a través LAN, además, el Protocolo de Control de Gestión Remota, RMCP (Remote Management Control Protocol), un protocolo de requerimiento/respuesta simple que utiliza UDP y que permite la ejecución de funciones de control tales como reinicio, encendido, apagado, del sistema gestionado en forma remota a través de la LAN.

Pre-inicio o pre-sistema operativo y sistema operativo ausente son los estados de un sistema computacional en los que el sistema operativo no está disponible Estos estados se pueden presentar cuando el sistema no ha terminado de cargase o debido a problemas en el arranque o errores en el sistema operativo, o debido a que el sistema está en un estado de bajo consumo de potencia.

Tanto, IPMI [1] como ASF son especificaciones complementarias que pueden proporcionar gestión de plataforma en un entorno de pre-inicio o pre-sistema operativo y de sistema operativo ausente. IPMI utiliza como el principal elemento del sistema de gestión un microcontrolador de gestión, mientras ASF utiliza un dispositivo de envío de alertas que realiza el polling[4] de los dispositivos de la motherboard, tales como sensores, y genera y envía alertas autónomamente. Además, tanto IPMI como ASF utilizan PET y RMCP para la gestión a través de la LAN.

Se puede considerar que el alcance de ASF son los sistemas de escritorio y móviles, y de PMI son los servidores, en los que las capacidades IPMI adicionales tales como registro de eventos, múltiples usuarios, autenticación remota, múltiples transportes, buses de extensión de gestión, acceso a sensores, etc., son muy importantes. Sin embargo no existen restricciones en ninguna especificación sobre la clase de sistemas en que se pueden utilizar. Por ejemplo, se puede utilizar IPMI para sistemas de escritorio y móviles, y ASF para servidores si el nivel de gestión se ajusta a los requerimientos.

[4] Polling. Proceso de solicitar y obtener información en forma secuencial.

2.3. HPI Y SNMP

Esta especificación define la MIB SNMP para la Especificación HPI [37]. Esta MIB define la instrumentación HPI, en otras palabras, define un conjunto de objetos que permiten la gestión de entidades en una variedad de plataformas hardware. Cuando se utiliza esta MIB se recomienda utilizar SNMPv3 por razones de seguridad.

La MIB ve a una plataforma hardware como un conjunto de entidades físicas que se pueden gestionar individualmente. Una agrupación lógica de estas entidades comprende un dominio de gestión y cada entidad tiene un conjunto de atributos reflejados en tablas.

El identificador de empresa privado asignado a SAF es 18568, por tanto, SAF publica MIBs SNMP que están bajo el OID: 1.3.6.1.4.1.18568, iso.org.dod.internet.private.enterprise.saforum. Por ejemplo, para openhpi, la implementación abierta de hpi que se mencionó en el capítulo 1, el OID es: 1.3.6.1.4.1.18568.1.1.1, iso.org.dod.internet.private.enterprise.saforum.experimental.hpi.openhpi.

En la figura 16 se muestra la ubicación y conformación de la MIB HPI. Esta imagen se obtuvo al compilar las MIB HPI de SAF [38].

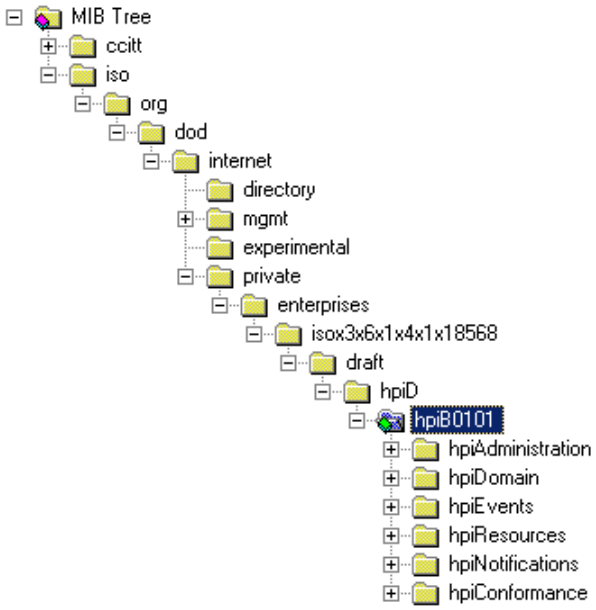


Figura 16. MIB SNMP HPI

2.4. AIS Y SNMP Y WBEM

Esta especificación busca describir la integración entre AIS y SMMP y CIM/WBEM [39], los estándares de gestión de red más importantes actualmente. Esta especificación define cómo exponer objetos de monitoreo y control para la funcionalidad de servicios de gestión AIS y AMF, pero en estos momentos, la especificación solamente define MIBs SNMP.

En la figura 17 se muestra la relación entre AIS, SNMP y WBEM.

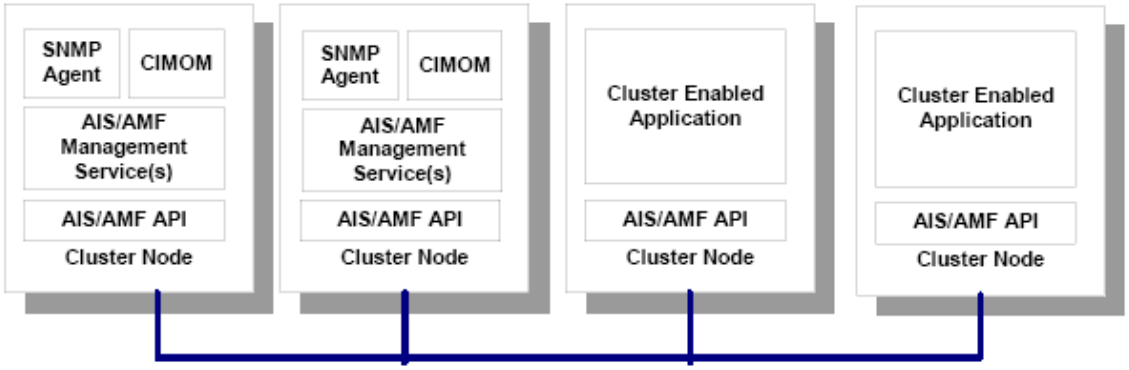


Figura 17. AIS y SNMP y WBEM [7]

Los servicios de gestión AIS/AMF se puede ejecutar en uno o más nodos y proporcionan monitoreo y control a la capa de aplicación AIS/AMF, mientras los agentes de gestión SNMP proporcionan acceso distribuido a estos servicios de gestión. Los servicios de gestión AIS/AMF pueden utilizar las APIs AIS/AMF pero también puede utilizar otras APIs.

Como se mencionó anteriormente, el identificador de empresa privado asignado a SAF es 18568, por tanto, SAF publica MIBs SNMP que están bajo el OID: 1.3.6.1.4.1.18568, iso.org.dod.internet.private.enterprise.saforum.

La Especificación define una MIB SNMP de convenciones textuales globales y otra de convenciones textuales AIS, además, MIBs SNMP tanto para AMF como para todos los servicios AIS: AMF, Puntos de Chequeo, Membresía de Clúster, Eventos, Bloqueo, Log, Mensajes, Nombrado, Notificación.

A manera de ejemplo, en la figura 18 se muestra la MIB SNMP de convenciones textuales globales, en la figura 19 la MIB SNMP de convenciones textuales AIS, en la figura 20 la MIB SNMP AMF y en la figura 21 la MIB SNMP de Puntos de Chequeo. Estas imágenes se obtuvieron al compilar las MIBs AIS de SAF [40].

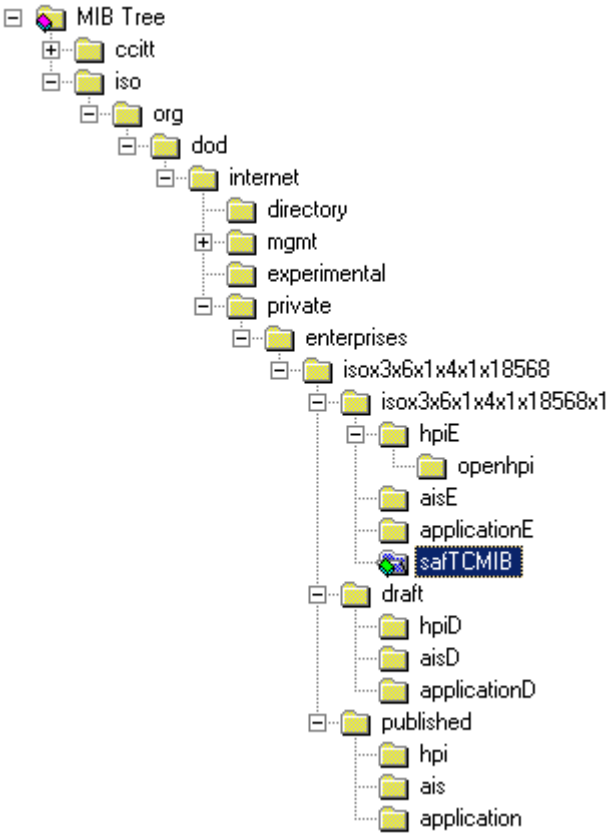


Figura 18. MIB SNMP de convenciones textuales globales

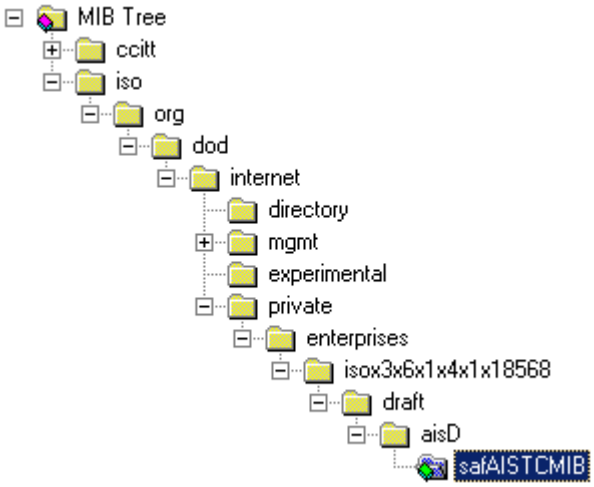


Figura 19. MIB SNMP de convenciones textuales AIS

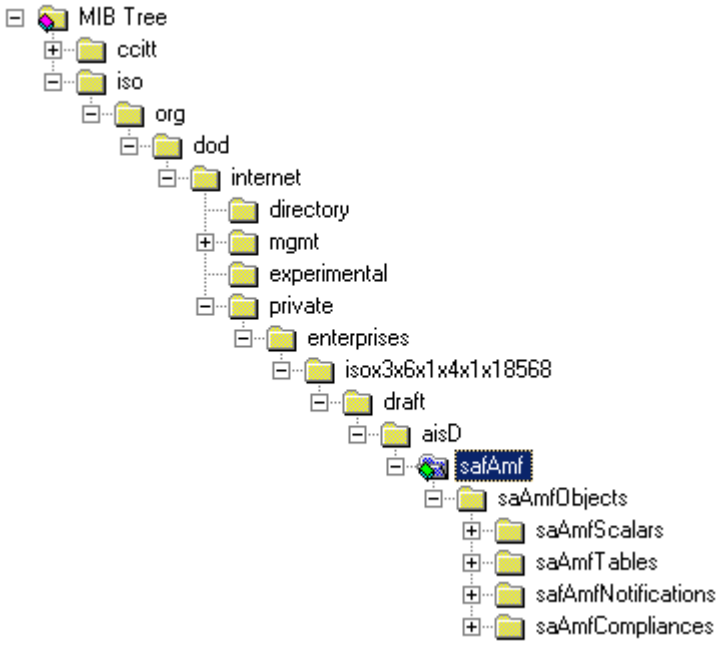


Figura 20. MIB SNMP AMF

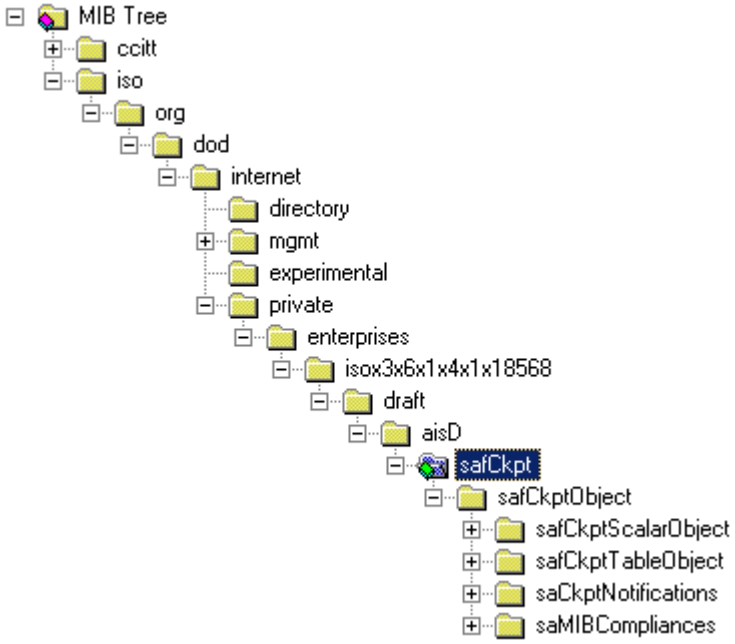


Figura 21. MIB SNMP de Puntos de Chequeo

Además del protocolo SNMP se podría utilizar el Protocolo AgentX [7] en una arquitectura maestro-subagente, es decir, con un agente SNMP y subagentes SNMP, en este caso HPI y AIS. En la figura 22 se muestra un ejemplo.

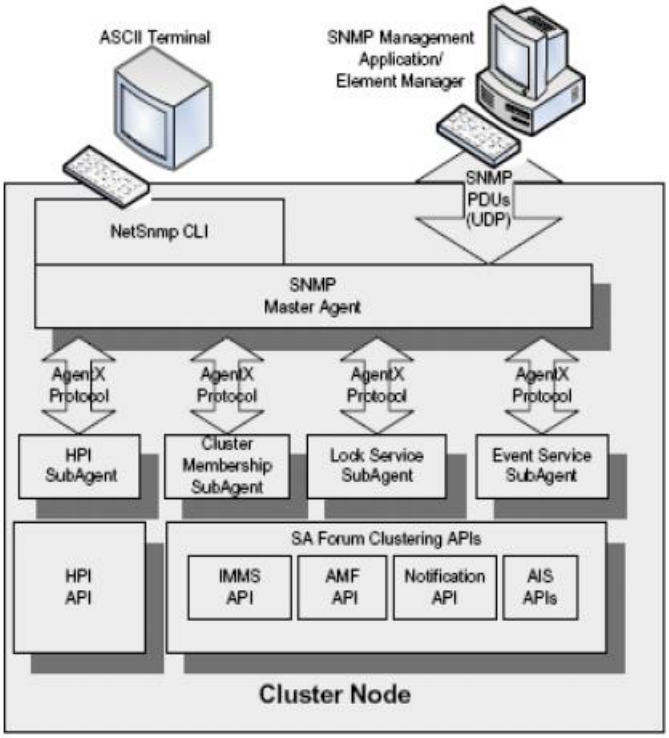


Figura 22. SNMP y AgentX con HPI y AIS [7]

En la arquitectura maestro-subagente SNMP todos los requerimientos SNMP se envían a un solo proceso dentro del nodo gestionado conocido como el agente maestro. El agente maestro puede ejecutar el requerimiento por sí mismo o enviarlo a un subagente. El subagente retorna una respuesta de regreso al agente maestro quien a su vez responde de regreso al gestor. Esta arquitectura evita al usuario conocer los múltiples agentes SNMP en el sistema ya que el usuario solamente necesita conocer al agente maestro.

El agente es responsable de recibir, validar, autenticar, autorizar y ejecutar requerimientos de usuario. El agente llama las MIBs apropiadas para ejecutar los requerimientos y retorna el resultado. Si ninguna MIB está disponible para ejecutar los requerimientos se retorna un error. El agente también es responsable de generar eventos asíncronos (Traps) para el usuario.

El agente maestro es un agente SNMP que soporta el protocolo AgentX que permite al agente SNMP ser extendido dinámicamente de tal forma que se pueda adicionar y remover fácilmente el soporte para varias MIBs. El protocolo AgentX sirve como el mecanismo de comunicaciones entre el maestro y los subagentes.

El subagente AgentX es responsable de implementar una o más MIBs y solamente utiliza protocolo AgentX y no el protocolo SNMP. Un subagente debe ejecutarse junto con un agente maestro AgentX. Un subagente se puede adicionar y remover del sistema en cualquier momento.

En primer lugar un subagente establece una sesión con un agente maestro y registra las MIBs por las que va a ser responsable y el agente maestro utiliza esta información de

registro cuando envía requerimientos a los subagentes. Finalmente, el subagente remueve su registro con el agente maestro y cierra la sesión.

2.5. ESTÁNDARES DEL DMTF

En la figura 23 se muestran las tecnologías del DMTF [41] como CIM, WBEM e iniciativas como SMASH, entre otras.

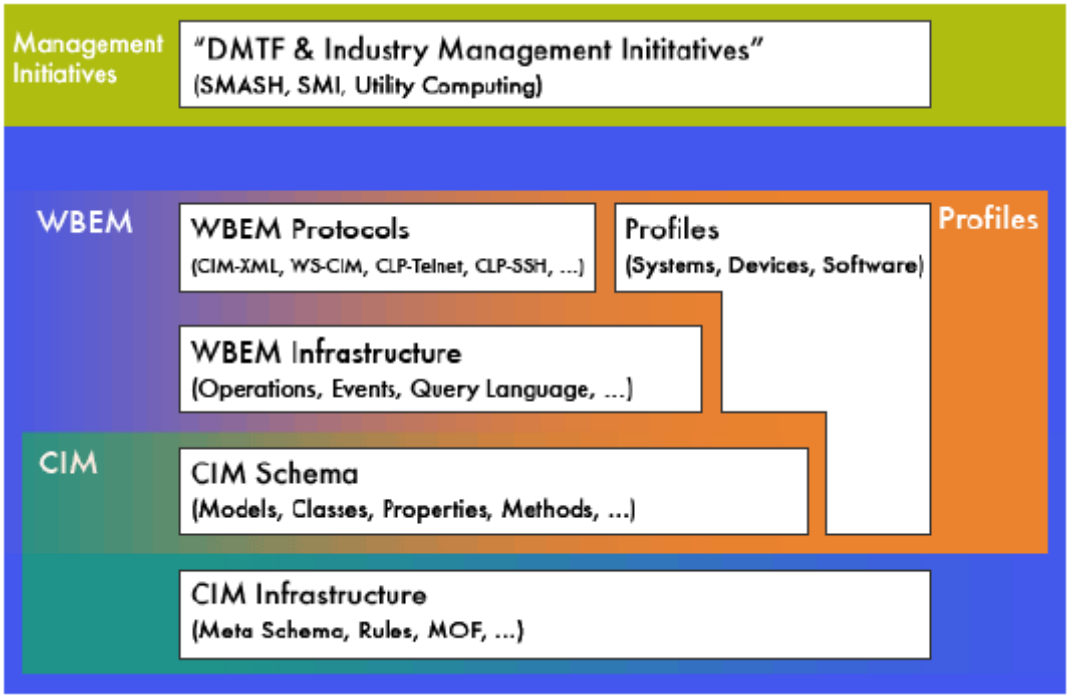


Figura 23. Tecnologías del DMTF [41]

Las tecnologías del DMTF están diseñadas para trabajar en conjunto con el fin de satisfacer las necesidades y requerimientos de la industria por una gestión distribuida e interoperable. Estos estándares proporcionan interfaces bien definidas que se basan en los otros estándares del DMTF y entregan capacidades de gestión e interoperabilidad extremo a extremo. Las relaciones entre las tecnologías del DMTF entregan un valor incremental a través de la pila lo que proporciona valor agregado con cada capa adicional que se implementa.

La base de las tecnologías del DMTF es el Modelo de Información Común, CIM (Common Information Model). La Especificación CIM define las reglas de CIM y proporciona los detalles para la integración con otros modelos de gestión. La siguiente capa es el Esquema CIM, el cual entrega descripciones del modelo orientado a objetos, rico semánticamente para todos los elementos gestionados. El Esquema CIM facilita la integración y la reducción de costos al permitir el intercambio de información de gestión en un modo independiente de la plataforma y neutral a la tecnología.

Sobre CIM se encuentra la Gestión de Empresa Basada en Web, WBEM (Web-Based Enterprise Management), un conjunto de tecnologías estándar de gestión e Internet desarrolladas para unificar la gestión de los entornos computacionales distribuidos. WBEM permite que la industria entregue un conjunto de herramientas de gestión basadas

en estándares, que facilitan el intercambio de datos a través de tecnologías y plataformas diferentes. Además de CIM y WBEM, el DMTF tiene otros estándares que son: SMBIOS (System Management BIOS), ASF (Alert Standar Format) y DMI (Desktop Management Interface).

SMBIOS [42] determina cómo los vendedores de motherboards y de sistemas presentan información de gestión acerca de sus productos en un formato estándar al extender la interfaz BIOS en sistemas con arquitectura x86.

ASF [43] define interfaces de alertas y control remotas para gestionar principalmente los entornos de sistema operativo ausente.

DMI [44] genera un framework estándar para gestionar componentes en un computador de escritorio o servidor. Debido al rápido advenimiento de las tecnologías del DMTF, tales como CIM, el DMTF definió un proceso de "Fin de Vida" para DMI, el cual terminó el 31 de marzo de 2005.

En La figura 23 también se observan los Perfiles, los cuales proporcionan una plantilla para tratar dominios de gestión específicos. Al desarrollar un modo unificado para describir un dominio de gestión dado en CIM, los Perfiles ofrecen un medio simplificado para lograr una gestión distribuida interoperable fácilmente.

En la parte superior están las iniciativas de gestión del DMTF, así como también de otras organizaciones de la industria que se basan en las tecnologías del DMTF. Estas iniciativas, las cuales entregan funcionalidad para aplicaciones e industrias verticales específicas, incluyen iniciativas importantes tales como SMASH (Systems Management Architecture for Server Hardware), CDM (Common Diagnostic Model), así como también SMI-S de SNIA (Storage Networking Industry Association's Storage Management Initiative Specification).

CDM [45] es una extensión de CIM (Common Model Information y se utiliza ampliamente dentro de la industria para evaluar el estado de sistemas computacionales en entornos multi-vendedor. La iniciativa CDM crea instrumentación de diagnóstico que puede ser utilizada por aplicaciones de gestión, y su alta sinergia con otros dominios de gestión en CIM permite la integración futura de diagnósticos en funciones de gestión críticas.

SMI-S [46] es una especificación desarrollada por SNIA (Storage Networking Industry Association) para estandarizar las tecnologías de gestión de almacenamiento interoperables, con base en la rica fundamentación proporcionada por CIM y WBEM.

Todas estas tecnologías del DMTF entregan soluciones potentes que ayudan a sobrellevar los desafíos asociados con los entornos de tecnologías complejos y heterogéneos de hoy.

2.6. SNMP Y WBEM

Teniendo en cuenta que tanto WBEM como SNMP son dos de los estándares más importantes actualmente, se han buscado formas de lograr su integración.

Adaptador SNMP. En primer lugar, un adaptador SNMP para WBEM [47] permite a las aplicaciones de gestión SNMP acceder información de gestión del sistema proporcionada por WBEM. Este adaptador mapea requerimientos SNMP a requerimientos WBEM

equivalentes. Además, remapea las respuestas del CIMOM a respuestas SNMP, la cual se retorna a la aplicación de gestión. Un archivo de mapeo contiene el nombre de la clase, nombre de la propiedad OID y el tipo ASN.1 para cada objeto

Proveedor SNMP. Por otro lado, un proveedor SNMP [48] permite a las aplicaciones de gestión WBEM acceder información SNMP. El proveedor SNMP es un componente software que proporciona información acerca de los elementos gestionados al CIMOM. Solamente el gestor necesita soportar WBEM ya que gracias al proveedor SNMP se puede comunicar con cualquier sistema gestionado que soporte SNMP.

Las variables de las MIBs SNMP se puede leer y escribir, y los traps se pueden mapear automáticamente a eventos WMI. Para acceder información SNMP a través del proveedor SNMP, se utiliza el archivo de una MIB para generar un archivo MOF. El proveedor SNMP mapea operaciones CIM que se ejecutan en clases CIM a operaciones SNMP. El proveedor SNMP soporta traps. Los traps corresponden a eventos. Cuando un cliente se subscribe con un proveedor para un evento, el proveedor escucha los traps y la información del trap se copia a la indicación, y en seguida la indicación se entrega al cliente.

Debido a que SNMP es ampliamente utilizado, los administradores buscan tener un medio para gestionar sus dispositivos habilitados con SNMP a través del CIMOM, lo que hace que el proveedor SNMP una alternativa muy interesante y adecuada para lograr este objetivo.

2.7. HPI Y AIS Y CIM

El SAF y el DMTF realizaron una alianza [49] en la que las dos organizaciones trabajan para lograr la interoperabilidad de sus estándares. El SAF se beneficia por la integración de funciones de gestión estándar de alta disponibilidad del SAF en clientes de gestión de empresa que soportan los estándares CIM/WBEM del DMTF, mientras el DMTF se beneficia por la adición de soluciones CIM/WBEM del DMTF que incluyen aplicaciones de alta disponibilidad, y por la actualización y verificación de CIM en el área de alta disponibilidad.

La alianza trabaja para definir la gestión para las funciones en las especificaciones del SAF, HPI y AIS, y para extender CIM para soportar tanto la gestión de funciones/servicios SAF (posiblemente al definir una nueva subclase de servicio, nuevos eventos y asociaciones necesarias) así como también definir nuevas entidades y asociaciones que traten aspectos específicos de gestión.

El proceso de enlazar HPI, AIS y CIM es el siguiente:

- Realizar el mapeo de objetos HPI y AIS a objetos CIM existentes.
- Donde el CIM existente no permita un mapeo, proponer extensiones de CIM a través de grupos de trabajo del DMTF.
- Crear perfiles de interoperabilidad WBEM tanto para HPI como para AIS.

El SAF tiene un subgrupo de Gestión de Sistemas, SM (System Management) que es responsable de crear los modelos y las interfaces de gestión estándar para HPI y AIS.

2.8. INTEROPERABILIDAD DE WBEM

Finalmente, como se puede observar, WBEM es uno de los estándares más importantes en el mundo de la gestión que se integra muy bien con otras tecnologías de gestión, razón

por la cual ha despertado el interés de grupos de investigación y desarrollo que han realizado diferentes implementaciones, han desarrollado varios CIMOM y proveedores. Pero como se mencionó en el capítulo 1, no existe una interfaz estandarizada entre CIMOMes y proveedores, y además, las aplicaciones de gestión son diferentes para cada implementación, que hacen que las implementaciones de este estándar aún no se puedan integrar entre ellas, pero afortunadamente, muchos grupos de investigación y desarrollo tienen como principal objetivo de trabajo solucionar esta situación y lograr que WBEM sea una tecnología que se pueda integrar fácilmente en soluciones de gestión y además llegue a ser interoperable con otras tecnologías de gestión. Estas iniciativas son CIMPLE/CMPI y Capa de Abstracción las cuales se describen a continuación.

2.8.1. CIMPLE y CMPI. CIMPLE [50] es una máquina de proveedores. Ofrece un entorno completo para desarrollar y una interfaz para invocarlos para invocarlos desde otras aplicaciones. CIMPLE se utiliza para:

a. Construir proveedores que trabajan con una variedad de Servidores CIM.
b. Proporcionar una base para implementar estándares basados en CIM tales como WBEM, SMASH, WS-Management, WSDM.

CIMPLE tiene tres principales ventajas sobre las tecnologías de proveedores convencionales:

- Reduce el tiempo de desarrollo de un proveedor.
- Permite desarrollar proveedores para un rango amplio de Servidor CIM.
- Permite obtener proveedores muy eficientes y pequeños.

Por otro lado, la Interfaz de Programación de Gestión Común, CMPI [51] (Common Manageability Programming Interface) es una interfaz estándar para proveedores CIM desarrollada por el Open Group. Esta interfaz es soportada por muchos Servidores CIM populares tales como OpenPegasus, OpenWBEM y SFCB (SBLIM Small Footprint CIM Broker). CMPI es una interfaz proveedor que hace el desarrollo de proveedores CIM independiente de una implementación de Servidor CIM particular.

Los proveedores desarrollados con CIMPLE trabajan con una variedad de implementaciones de Servidor CIM, incluyendo OpenPegasus así como también servidores que cumplen con CMPI. Esto hace posible escribir un solo proveedor que interopere con un rango amplio de servidores CIM.

Por supuesto, el mismo rango de interoperabilidad se puede lograr con CMPI, pero CIMPLE ofrece muchos beneficios adicionales (por ejemplo: clases concretas, generación de un proveedor automática, reducción de operación, validación y registro automatizados y patrones de proveedores).

Sería un error pensar que CIMPLE compite con CMPI. Por el contrario, CIMPLE complementa a CMPI al proporcionar una interfaz más segura y más conveniente para desarrollar proveedores CMPI. Para una aplicación CMPI, los proveedores CIMPLE y los proveedores CMPI son indistinguibles. Cualquier proveedor CIMPLE se puede enmascarar como un proveedor CMPI al utilizar un adaptador como se muestra en la figura 24.

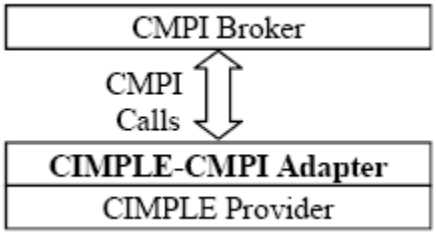


Figura 24. CMPI y CIMPLE [50]

Además de la interfaz CMPI, CIMPLE tiene un gestor de proveedores que permite que los proveedores CIMPLE trabajen con el servidor OpenPegasus en instalaciones que carecen de soporte CMPI.

El CIMOM implementa dos estándares: CIM y WBEM. CIM define la infraestructura CIM subyacente mientras WBEM define el protocolo para acceder esa infraestructura. Visto de este modo, WBEM es el front end de la implementación del CIMOM y CIM es el back end, como se muestra en la figura 25.

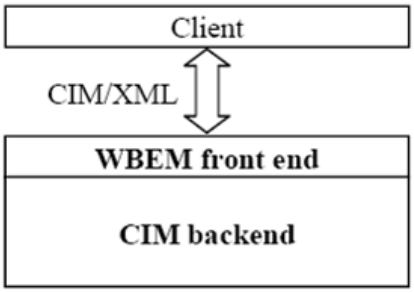


Figura 25. CIM y WBEM [50]

A partir del CIMOM, han surgido otros estándares basados en CIM tales SMASH, WS Management y WSDM. Cada una de estas arquitecturas emergentes necesitan un back end CIM pero ninguna necesita un front end WBEM, ya que cada una define su propio protocolo alternativo.

Los implementadores de SMASH, WS Management y WSDM deben reutilizar o reinventar la mayor parte del back end CIM. Desafortunadamente, no es posible reutilizar el backend CIM de cualquier servidor CIM existente, ya que no existe una interfaz formal para hacerlo y no existe un modo de utilizarlo aparte del front end WBEM.

Afortunadamente, CIMPLE fue diseñado para ser embebido directamente en otras aplicaciones. Es ideal para implementaciones SMASH, WS Management y WSDM, así como también como CIMOMs. CIMPLE proporciona al desarrollador una máquina de proveedores para cargar y gestionar proveedores y una interfaz para invocarlos.

CIMPLE es parte del esfuerzo en curso para simplificar la adopción de CIM y se están buscando nuevos modos de lograr esto. Se está tratando de determinar los patrones clave para desarrollar proveedores y cómo se pueden automatizar, y si se pueden generar frameworks de proveedores automáticamente a partir de los perfiles, entre otros.

Teniendo en cuenta lo anterior, CIMPLE [52] se puede utilizar en los siguientes escenarios.

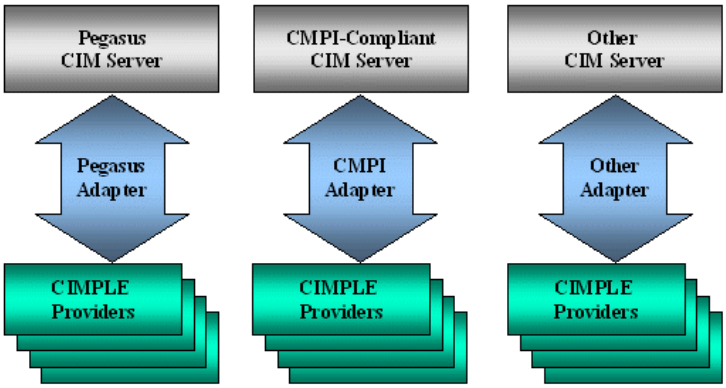


Figura 26. Servidores CIM y proveedores CIM [52]

En la figura 26 se observa que los proveedores CIMPLE se pueden integrar con cualquier Servidor CIM, como el de Pegasus, un Servidor CIM que utilice CMPI o cualquier otro Servidor CIM mediante los adaptadores adecuados.

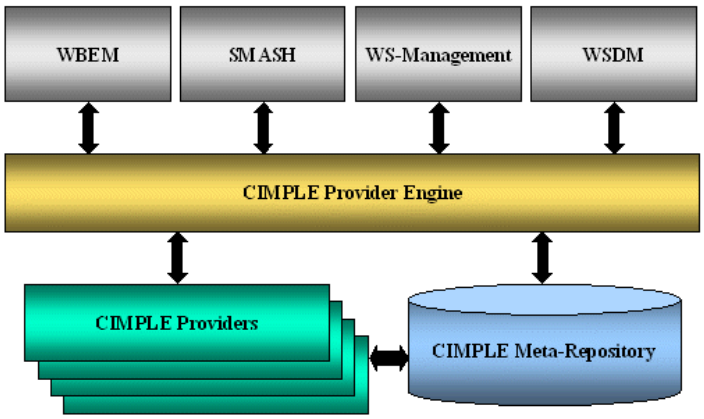


Figura 27. Implementaciones basadas en CIM y proveedores CIMPLE [52]

En la figura 27 se observa que los proveedores CIMPLE se pueden integrar con cualquier implementación WBEM, SMASH, WS-Management y WSDM.

2.8.2. Capa de abstracción. La interoperabilidad entre cliente y servidor en aplicaciones WBEM está limitada severamente por los múltiples protocolos de comunicación que se utilizan en las implementaciones WBEM. Las implementaciones más importantes son WMI para Windows y OpenWBEM para Linux, las cuales utilizan dos protocolos de comunicación diferentes, DCOM y CIM-XML sobre HTTP, respectivamente. Adicionalmente, existe la iniciativa de realizar futuras implementaciones con nuevos protocolos, WS Management de Microsoft y WSDM de OASIS.

Teniendo en cuenta que las implementaciones WBEM utilizan varios protocolos, un método para cargar y seleccionar dinámicamente el protocolo en el cliente soportado por un servidor en tiempo de ejecución consiste en escribir una aplicación cliente diferente

para cada protocolo, pero afortunadamente existe la Capa de Abstracción de CIMOM [53], cuya arquitectura se muestra en la figura 28.

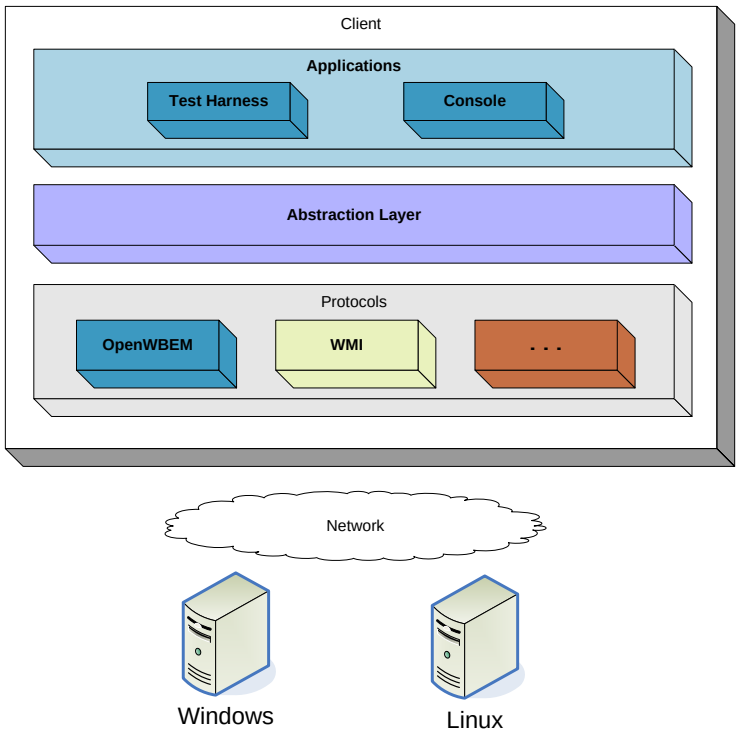


Figura 28. Capa de abstracción de CIMOM [53]

La solución de la Capa de Abstracción consiste de los siguientes componentes:

- Las interfaces a la capa de abstracción.
- Adaptador de protocolo para el protocolo CIM-XML sobre HTTP de WBEM.
- Adaptador de protocolo para DCOM de WMI a la capa de abstracción.

La Capa de Abstracción permite la carga dinámica de adaptadores de protocolo, descubrimiento en línea del protocolo requerido para comunicarse con el servidor y proporciona un framework de adaptadores de protocolo extensible.

La aplicación cliente carga dinámicamente los adaptadores de protocolo al inicio (pero se puede adaptar para que los cargue en tiempo de ejecución si se requiere), y además se puede utilizar una interfaz de protocolo programable para proporcionar protocolos adicionales. El cargador de protocolo busca las bibliotecas del protocolo en un directorio predefinido, las cuales se cargan y se buscan clases que implementan la interfaz del protocolo. De este modo los adaptadores de protocolo se pueden adicionar y remover al adicionar o remover los archivos de la biblioteca del protocolo del directorio. El componente de descubrimiento detecta el protocolo correcto para un servidor ya que el algoritmo de descubrimiento de protocolo itera los adaptadores de protocolo cargados hasta que detecta el protocolo apropiado y entonces el adaptador de protocolo adecuado puede ser utilizado por la aplicación. El componente del adaptador de protocolo es utilizado por la aplicación para comunicarse con el servidor una vez se determina el protocolo correcto.

3. REDES EHAS

En este capítulo se describen las características de las redes EHAS y se resalta el trabajo que la Fundación EHAS ha realizado para lograr mejores comunicaciones a través HF, VHF e IEEE 802.11, además, se describe la red EHAS Colombia y EHAS Perú Cusco y finalmente, dos alternativas para las redes EHAS que son WIMAX y DTNs, así como también su gestión utilizando SNMP.

3.1. CARÁCTERÍSTICAS DE LAS REDES EHAS

La Fundación EHAS (Enlace Hispanoamericano de Salud) es una organización sin ánimo de lucro que busca contribuir a la mejora de los servicios de los sistemas públicos de asistencia sanitaria y de salud en las zonas rurales de los países de América Latina mediante el uso de tecnologías adecuadas de información y comunicaciones.

La Fundación EHAS ha desarrollado hasta el momento tres programas: EHAS-Perú, EHAS-Colombia y EHAS-Cuba, y varios proyectos con el apoyo de diferentes entidades financiadoras. En los tres países, Perú, Colombia y Cuba, la Fundación EHAS trabaja en zonas rurales aisladas, generalmente de muy difícil acceso, que no cuentan con sistemas de comunicaciones, incluso algunas de ellas no tienen energía eléctrica y lastimosamente parecen estar olvidadas por los gobiernos. La Fundación EHAS busca conectar los puestos de salud con su centro de salud de referencia u hospital e Internet para que el personal de salud se pueda comunicar con profesionales con mayor formación o experiencia, pueda remitir pacientes a centros de salud u hospitales y alertar sobre epidemias, además, para facilitar el intercambio de información sanitaria y de salud entre puestos y centros de salud u hospitales que pueden encontrarse separados por varios kilómetros, y en los que por lo tanto, este proceso llevaría horas e incluso días y podría llegar a ser bastante costoso teniendo en cuenta los pocos medios de transporte disponibles en algunas de las zonas. Gracias a los sistemas de comunicaciones que instala, la Fundación EHAS capacita al personal de salud en temas que interesan a la comunidad por medio de cursos que elaboran médicos de los hospitales y se envían tanto a los centros como puestos de salud, y además, capacita al personal técnico para que se encargue del mantenimiento de los sistemas que la Fundación instala.

EHAS trabaja con tecnologías de información y comunicaciones adecuadas a las zonas objetivo, ya que estas tecnologías se caracterizan por:

- **Bajo costo.** Permite que los sistemas que se instalan puedan ser sostenidos por los puestos y centros salud que generalmente tienen muy pocos recursos.
- **Bajo consumo.** Permite alimentar los sistemas de una forma relativamente fácil y económica ya que en muchas ocasiones los puestos y centros de salud no tienen energía eléctrica.
- **Inalámbricas.** Permite comunicaciones de buena calidad entre puestos y centros de salud bastante aislados, ya que estas tecnologías se seleccionan teniendo en cuenta las características de propagación de las zonas, en algunos casos bastante difíciles de comunicar.

EHAS utiliza tecnologías como: Wi-Fi, VHF y HF. Wi-Fi, también conocido como el estándar IEEE 802.11b, es una tecnología full duplex que requiere línea de vista entre los

puntos que se desea comunicar pero no necesita el pago de licencias por la utilización de frecuencias y ofrece un buen ancho de banda para las comunicaciones. Además, gracias a la configuración de ciertos parámetros del protocolo, se han logrado enlaces de longitudes superiores a las establecidas para esta tecnología. Además, se está utilizando el estándar IEEE 802.11 g para alcanzar mayores distancias y ofrecer mayor velocidad, y en estos momentos se están estudiando otros estándares para mejorar las características de las comunicaciones y conseguir redes que se puedan administrar y mantener más fácilmente, como son: IEEE 802.11 e, redes Mesh, entre otros. HF y VHF son tecnologías semi duplex que permiten la comunicación de voz a velocidades menores que las que ofrece Wi-Fi, pero a diferencia de esta tecnología no requieren línea de vista y por tanto son adecuadas para lograr la comunicación de zonas más aisladas y con características de propagación más complejas. Además, se han realizado adaptaciones que permiten la comunicación no solo de voz sino también de datos de buena calidad y a velocidades mayores de las que ofrecen VHF y HF normalmente.

En las redes EHAS se utilizan estas tres tecnologías. La tecnología que se escoge para cada zona a comunicar depende de sus características de propagación. Teniendo en cuenta esto, se puede decir que las redes EHAS contienen una mezcla de tecnologías, es decir, son redes híbridas que permiten la mejor comunicación de las zonas objetivo. Sobre estas redes de comunicaciones se ofrecen servicios de información que se basan en el servicio de correo electrónico, ya que este servicio se proporciona tanto a través de las redes Wi-Fi, como VHF y HF. En estas últimas gracias a que como se mencionó anteriormente es posible la transmisión de datos en VHF y HF.

En las redes EHAS, en forma general, los establecimientos de atención primaria en los países en desarrollo se pueden agrupar en dos categorías: Centros de Salud (también llamados en otros países policlínicos) y Puestos de Salud (también llamados consultorios).

Hospitales o Centros de Salud. Son establecimientos de mayor jerarquía dentro del sistema público de atención primaria, situados en capitales de provincia o distrito donde suele llegar línea telefónica. Un Centro de Salud es centro de referencia de varios Puestos de Salud y es dirigido por médicos, posee infraestructura y equipamiento para realizar algunas pruebas diagnósticas y hospitalizaciones, y además, coordina las actividades de los Puestos de Salud asociados.

Puestos de salud. Son establecimiento de menor jerarquía dentro del sistema público de atención primaria y constituyen la puerta de acceso al sistema para la población rural. Suelen estar situados en poblaciones de no más de mil habitantes, generalmente sin líneas telefónicas y mal dotadas de infraestructura de carreteras.

Varios Puestos de Salud dependen de un único Centro de Salud conformando lo que se conoce como microred de salud, la cual se convierte en la unidad básica de atención primaria. Las microredes están dirigidas por un médico que es el responsable del Centro de Salud y que coordina las acciones de los Puestos de Salud que dependen del Centro de Salud. La mayoría de estos Puestos de Salud están dirigidos por técnicos de enfermería, enfermeras o a lo sumo un médico recién graduado, personal con escasa formación y que necesita comunicación con su médico de referencia.

En la figura 29 se muestran diferentes posibilidades para lograr la comunicación entre puestos y centros de salud utilizando las tecnologías Wi-Fi, VHF y HF que se mencionaron anteriormente.

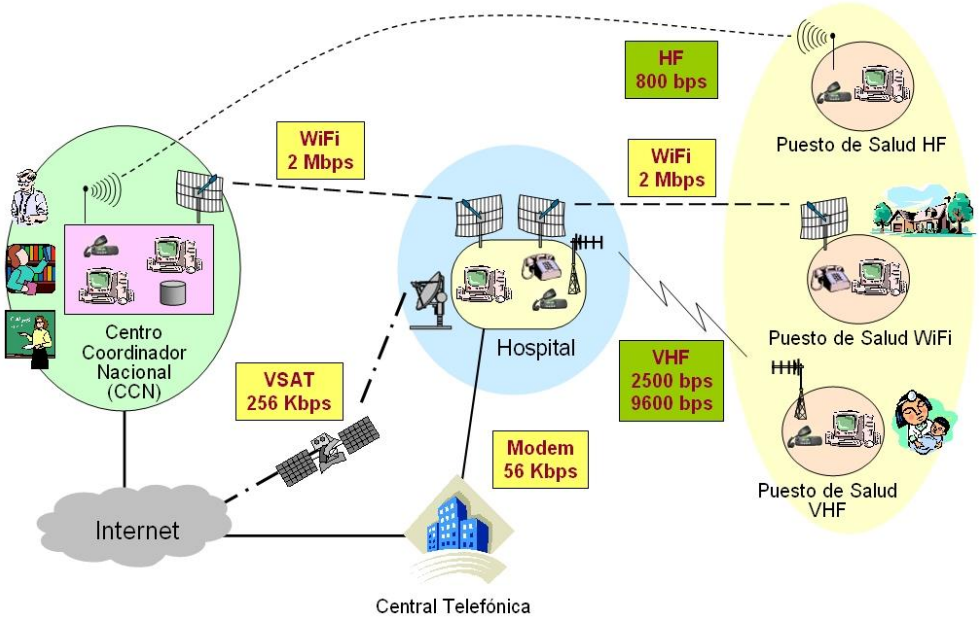


Figura 29. Redes EHAS [57]

La red conformada por un hospital y los puestos de salud VHF que dependen de él se llama microred. Por otro lado, los puestos de salud HF se comunican directamente con el CCN ya que HF es una tecnología de largo alcance. Teniendo en cuenta las tecnologías utilizadas en EHAS, la red puede estar constituida por enrutadores inalámbricos, estaciones VHF o HF, y servidores de correo electrónico ubicados en cada hospital, que se comunica con Internet a través de módem, Wi-Fi o satélite. Las redes Wi-Fi permiten además, comunicaciones de VoIP, incluso con la red telefónica básica, gracias a centralitas software que se encuentran en los enrutadores, y para aprovechar los teléfonos analógicos se utilizan ATAs (Adaptador de Teléfono Analógico). Los hospitales por tanto son estación Wi-Fi como estación VHF. Por supuesto, además son necesarias antenas, y si no hay energía eléctrica, un sistema de energía solar.

El sistema operativo instalado en las estaciones es Linux, lo que evita que los establecimientos de salud tengan que pagar por licencia. Además, las aplicaciones y programas son de fuente abierta y libres, lo que permite mayor control de las comunicaciones y del sistema.

En seguida se realizará una breve descripción del trabajo que la Fundación EHAS ha realizado tanto en VHF y HF así como también en Wi-Fi, el cual se constituye en uno de los principales aportes del proyecto para buscar el bienestar de la comunidad.

3.1.1. HF y VHF. En primer lugar, para las comunidades rurales, las comunicaciones de voz son muy importantes y lo son aún más si se complementan con comunicaciones de datos. Teniendo en cuenta la filosofía del bajo costo, estas comunicaciones se logran utilizando los mismos equipos que se utilizan para las comunicaciones de voz, además con un bajo consumo de potencia. Teniendo en cuenta el bajo ancho de banda tanto de VHF como HF [54], el servicio de datos más adecuado sobre este tipo de redes es el correo electrónico.

Por una parte, VHF/UHF (30-3000 MHz), permite comunicaciones de corta y media distancia, entre puntos sin visibilidad directa, en un radio máximo de 50 Km. Esta banda presenta gran estabilidad y no depende en gran medida de las condiciones medioambientales.

HF (3-30 MHz), también llamada de onda corta, permite comunicaciones de larga y muy larga distancia gracias a la propagación ionosférica (también por ondas superficiales en cortas distancias). El principal inconveniente es la baja calidad de los canales y por tanto la calidad de las comunicaciones, ya que éstas dependen de muchos factores medioambientales.

Las estaciones VHF o HF cuentan con un computador que tiene instalado el sistema operativo Debian, y una tarjeta de sonido, además, un radio VHF o HF, además de una impresora. El radio se conecta al computador mediante una tarjeta o placa de interfaz, denominada placa PTT, debido al control Push to Talk, que adapta las señales de audio para la tarjeta de sonido que hace las veces de radio-módem, y que contiene circuitos que le permiten al computador controlar algunas funciones del radio. A esta tarjeta se le han agregado otras funciones de monitoreo, como el nivel de agua de las baterías. En este caso, las tareas de audio quedan encomendadas a la tarjeta de sonido, y la de activación del PTT queda reservada a la placa de interfaz diseñada para un puerto USB.

Todo el trabajo relacionado con la tarjeta de sonido ha estado a cargo de EHAS-España y ha consistido básicamente en el estudio de los diferentes protocolos de comunicación, los servicios que se podría ofrecer y la realización de una serie de mejoras que han permitido obtener lo mejor de esta tecnología, es decir lograr mayor calidad y velocidad no solo en la comunicación de voz sino también de datos.

Esto se ha logrado gracias a la utilización de software libre de fuente abierta existente que trabajando en conjunto ofrece un sistema de transmisión digital funcional para correo electrónico y acceso a Internet. Algunas de estas aplicaciones tenían serias carencias que repercutían en la velocidad y calidad de transmisión del usuario, pero el trabajo realizado se detectaron y se mejoraron, gracias por supuesto a que el código es accesible y modificable.

3.1.2. IEEE 802.11. Por otro lado, en cuanto a las comunicaciones utilizando los estándares IEEE 802.11 [55], EHAS ha decidió utilizar una placa Soekris, un computador empotrado con arquitectura x86, que cuenta con tres interfaces inalámbricas y tiene una alta robustez ante condiciones climáticas adversas, dos características superiores de esta palca con respecto a otras como WRAP. Además, esta placa tiene un bajo costo y soporta Linux.

Por otro lado, en esta placa se decidió utilizar el sistema operativo Pebble, una versión reducida de Linux a la que se le hicieron los ajustes de paquetes necesarios para EHAS y se denominó pebble-EHAS.

En 802.11 todas las transmisiones de datos unicast son confirmadas por el receptor mediante ACKs, por tanto, para largas distancias, el valor de ACKTimeout, es decir el tiempo que se espera de un ACK, debe ser diferente al de cortas distancias y debe tener el valor adecuado.

Se han realizado trabajos de investigación para modificar el nivel MAC de 802.11 para el caso particular de enlaces Wi-Fi de larga distancia pero muchos solo se han implementado de manera comercial. Muchos de estos trabajos se basan en la idea de que determinados controladores pueden modificar parámetros como el ACKTimeout que no está especificado por el estándar 802.11, deshabilitar ACKs o ajustar o deshabilitar las retransmisiones.

En EHAS se trabaja con dos controladores, uno de ellos es madwifi que permite modificar el valor de ACKTimerout, además de CTSTimeout y SLOT Time. El otro controlador que se utiliza en EHAS es Hostap que no permite modificar ninguno de estos valores, sin embargo las tarjetas manejadas por este controlador tienen un ACKTimeout por defecto mayor. Además, este controlador permite un modo llamado Pseudos-IBSS en el que se eliminan los ACKs. El trabajo en esta área es cada vez mayor pero aún se requiere que estos drivers sigan madurando y adicionando nuevas características.

Como se mencionó anteriormente, en los enrutadores se utilizó Pebble-EHAS, ya que es muy adecuada para las capacidades de almacenamiento de las placas, pero además, se realizaron adaptaciones a la distribución Linux que se utiliza para las estaciones y se comprobó su buen funcionamiento en los enrutadores. Aún se continúa trabajando en el sistema operativo de los enrutadores tanto para la placa soekris como para otras placas que se han decido probar también por sus buenas características, adecuadas a los requerimientos del proyecto.

Algo muy importante que se ha agregado a los enrutadores es Asterisk, una centralita en software que ha permitido la comunicación de VoIP utilizando ATAs y teléfonos convencionales.

EHAS-España, además, ha estado trabajando para lograr un enrutador inalámbrico [56] con alimentación solar, de bajo coste, con capacidad para crear redes inalámbricas Wi-Fi de larga distancia y alta velocidad, que además, sea autoconfigurable, soporte QoS y ofrezca seguridad.

Como se mencionó, hasta ahora en las redes EHAS se han utilizado los estándares IEEE 802.11 b e IEEE 802.11 g y existe un gran interés en las redes Mesh Wi-Fi ya que por sus características podrían ser una solución muy adecuada para los escenarios en los que trabaja EHAS.

Las redes Mesh Wi-Fi se pueden definir como redes que se forman espontáneamente sin ninguna infraestructura, conformando mallas que conectan nodos entre sí, en las que cada nodo puede ser al mismo tiempo estación de trabajo de un usuario, servidor o enrutador. Este tipo de redes tiene las siguientes características.

- Utiliza la tecnología Wi-Fi ampliamente conocida.

- Bajo costo, debido a la popularidad de Wi-Fi.
- Tiene una arquitectura descentralizada y autoconfigurable. Evitaría la caída de grandes porciones de la red, y facilitaría la configuración, administración y mantenimiento de la red. Todo esto disminuiría las grandes inversiones en tiempo y dinero que requieren los desplazamientos del personal técnico a los sitios de instalación.
- Baja potencia de alimentación. Los nodos Mesh Wi-Fi necesitan mínima energía con respecto a otras tecnologías. Esto permitiría alimentar los nodos con energía natural como energía solar.
- Flexibilidad. Un nodo puede adherirse a la red si ve alguno de sus nodos vecinos. Esto facilitaría la instalación de las redes.

El creciente interés por las redes Mesh y el gran avance de las comunicaciones móviles llevaron al IETF a la creación del grupo MANET (Mobile Ad-hoc NETwork) lo que demuestra la potencialidad de este tipo de redes. El trabajo en EHAS en estos momentos se centra en redes redes Mesh estáticas.

En redes Mesh los nodos pueden descubrirse y enlazarse adecuadamente de forma automática en cualquier situación. Esto es difícilmente conseguible con BSS (modo infraestructura), pero es teóricamente factible con IBSS (modo Ad-Hoc). No obstante, el cada vez más implementado modo WSD (Wireless Distribution System), que permite constituir por vía inalámbrica ESS (Extended Service Sets) está siendo utilizado por investigaciones muy reciente para poder establecer enlaces AP-AP en redes Mesh.

Uno de los aspectos que se deben tener en cuenta en las redes Mesh es la autoconfiguración a nivel IP para lo que se han analizado algunas alternativas.

Otro aspecto es el encaminamiento dinámico multisalto. Para las redes Mesh se han desarrollado muchos protocolos específicos que son capaces de encaminar paquetes entre nodos contiguos o no contiguos de la red, o entre cualquier nodo de la red y el exterior, sin importar que cada nodo tenga una dirección IP completamente independiente de los otros.

Además, en las redes Wif-Fi y por supuesto también en las redes Mesh Wi-Fi, existe otro aspecto que solo recientemente ha sido abordado por algunos grupos de investigación y por muy escasos proyectos de desarrollo, la QoS. Este aspecto, en el caso de EHAS es muy importante, ya que las comunicaciones telefónicas son uno de los servicios más demandados y prioritarios y por tanto deben ser de buena calidad.

Como se comentó anteriormente, el enrutador se instala en zonas rurales aisladas montañosas y selváticas, generalmente sin instalaciones eléctricas, lo que hace necesario un sistema de alimentación basado en energía natural, como por ejemplo un sistema de energía solar diseñado para proporcionar suficiente energía de forma continuada.

El costo de este sistema es proporcional a la potencia consumida por el enrutador por tanto es fundamental que el enrutador consuma baja potencia y se están estudiando diferente posibilidades para lograrlo y en general es un campo inmaduro pero cada vez se le dedican más esfuerzos, ya que en el caso de EHAS se ha visto que la utilización de la red no se realiza todo el tiempo, es decir que no es necesario que todos los nodos de la red estén funcionando a la vez.

Finalmente, otro aspecto que se debe tener en cuenta es la seguridad de la red, y más aún en redes Mesh ya que todos los mecanismos a excepción de WEP se han diseñado para arquitecturas centralizadas y por tanto, no se podría utilizar por ejemplo WPA, y entonces sería necesario añadir seguridad a niveles superiores.

A continuación, se describen las redes EHAS Colombia, para mostrar todo el trabajo que el GIT (Grupo de Ingeniería Telemática) ha realizado para cumplir con el objetivo de la Fundación.

3.2. RED EHAS COLOMBIA

En Colombia existen muchos proyectos de telemedicina pero muy pocos han trabajado en las montañas y en la selva del pacífico del Departamento del Cauca muy complicadas por las condiciones socio-económicas, políticas, geográficas y de violencia, y además utilizando tecnologías similares a las presentadas anteriormente.

La red de telecomunicaciones del Programa EHAS en Colombia [57] que se muestra en la figura 30 consta de cinco subredes, basadas en sendos hospitales del Departamento del Cauca. En orden de instalación son: Silvia (Hospital San Carlos), Guambía (Hospital Mamá Dominga) y Jambaló (Hospital de Jambaló), en las montañas de la Cordillera Central, y Timbiquí (Hospital Santa Bárbara) y Guapi (Hospital San Francisco de Asís), en la selvática costa pacífica del Departamento.

La subred de Silvia consta del Hospital San Carlos, en la cabecera municipal, y nueve puestos de salud: Usenda, Pitayó, Quizgó, Miraflores, La Estrella, Tumburao, Quichaya, Vallenuevo y Santa Lucía. La conexión a Internet del Hospital San Carlos es provista por la Universidad del Cauca a través de un enlace WiFi de 42 Km en dos saltos, línea roja continua en la figura 30. El primer salto va de la Facultad de Ingeniería Electrónica y Telecomunicaciones (FIET), en Popayán, a la torre del repetidor de la emisora Namuy Wuam en el cerro Nueva Guambía, en el Resguardo Indígena de Guambía, con una distancia de 36 Km y una velocidad de 2 Mbps. El segundo salto va del repetidor al hospital, con una distancia de 6 Km.

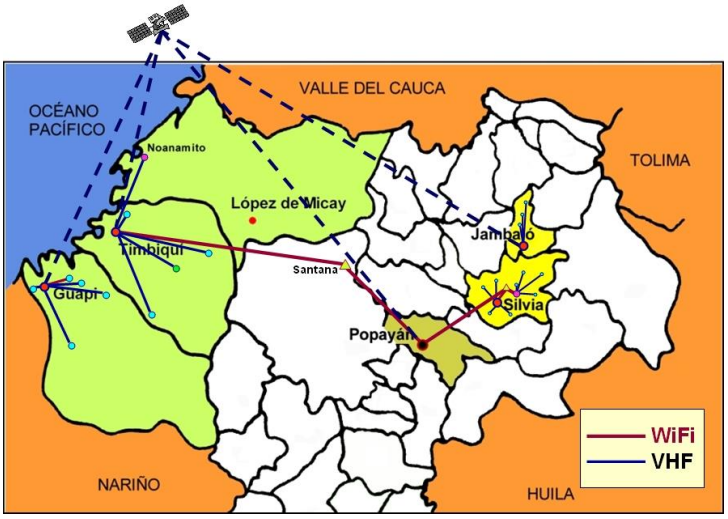


Figura 30. Red EHAS Colombia [57]

Siete puestos de salud están conectados con el Hospital San Carlos a través de enlaces VHF: Usenda, Pitayó, Quizgó, Miraflores, La Estrella, Quichaya y Vallenuevo. La topografía montañosa de la zona ha obligado a utilizar también un repetidor de VHF, que fue localizado en el mismo cerro Nueva Guambía, para posibilitar la conexión de algunos de estos puestos de salud. Esta solución no bastó, sin embargo, para los puestos de salud Tumburao y Santa Lucía, donde se tuvo que instalar radios HF que los comunican directamente con la sede de la Universidad del Cauca en Popayán.

La subred de Guambía consta del Hospital Mamá Dominga y siete puestos de salud: El Cacique, La Campana, Sierra Morena, Santa Clara, El Trébol, Agua Bonita y El Cofre, todos en el Resguardo Indígena de Guambía, localizado en el Municipio de Silvia. La conexión a Internet del Hospital Mamá Dominga es provista también por la Universidad del Cauca, a través del enlace Wi-Fi que va al cerro Nueva Guambía, con un salto de poco más de 1 Km.

Todos los puestos de salud, con la excepción de Agua Bonita, están conectados al Hospital Mamá Dominga mediante enlaces VHF, utilizando así mismo algunos de ellos el repetidor VHF del cerro Nueva Guambía. El puesto de salud Agua Bonita está conectado directamente a Popayán a través de un enlace HF.

En Jambaló, la subred consta del Hospital, localizado en la cabecera municipal, el Centro de Salud de Loma Redonda, y los puestos de salud de La Mina y Loma Gruesa. En este caso las montañas hicieron inviable una conexión Wi-Fi desde Silvia o Popayán, por lo que fue necesario utilizar una estación satelital. La estación se instaló en la alcaldía municipal, y desde allí se estableció un enlace Wi-Fi de 200 m aprox. hasta el hospital.

A pesar de la irregularidad del terreno, el centro de salud y los dos puestos de salud se conectaron directamente al hospital mediante enlaces VHF.

La subred de Timbiquí incluye el Hospital Santa Bárbara, en la cabecera municipal, cuatro puestos de salud (Puerto Saija, Santa Rosa de Saija, Aguaclarita y Santa María), y el Centro de Salud de Noanamito en el vecino municipio de López de Micay. La conexión a Internet en el Hospital Santa Bárbara tiene dos opciones. La primera de ellas consiste en un enlace Wi-Fi de 130 Km desde Popayán, que utiliza un repetidor en el cerro Santana, en la Cordillera Central. La distancia del salto desde la FIET, en Popayán, hasta el repetidor es de 41 Km, y desde el repetidor hasta el Hospital Santa Bárbara es de 88,5 Km. Este enlace se encuentra en período de prueba, y ofrece una velocidad efectiva de 1,3 Mbps, siendo uno de los enlaces Wi-Fi más largos de que tengamos noticia.

La otra opción consiste en un acceso satelital del Programa Compartel, que fue adjudicado al hospital e instalado después de iniciado el proyecto EHAS. El enlace Wi-Fi será utilizado como acceso primario por proveer mayor velocidad, y el enlace satelital será usado como enlace de respaldo.

Los cuatro puestos de salud de Timbiquí y el Centro de Salud de Noanamito están conectados al Hospital Santa Bárbara mediante enlaces VHF. Esta ha sido la subred más difícil de instalar, por las distancias y las condiciones del transporte. Se trata de localidades muy alejadas, a donde sólo se llega por transporte fluvial. Por ejemplo, el viaje de ida a Aguaclarita, que es una comunidad indígena de la familia Embera, requiere todo el día cuando los ríos están llenos.

La subred de Guapi, la última en ser instalada, consta del Hospital San Francisco de Asís, en la cabecera municipal, y cinco puestos de salud: Limones, San José de Guare, San Antonio de Guajuí, Chanzará y El Naranjo. El acceso a Internet del hospital es provisto por una estación satelital instalada por el Programa Compartel. Se tiene prevista la instalación de un enlace Wi-Fi hasta el Hospital Santa Bárbara, pero está pendiente de la gestión de los recursos.

La subred había sido diseñada de modo que los puestos de salud de Limones y El Naranjo se conectarían al hospital mediante enlaces Wi-Fi. Sin embargo, el enlace con El Naranjo, distante 20 Km del hospital, no se pudo establecer. Así pues, cuatro puestos de salud (San José de Guare, San Antonio de Guajuí, Chanzará, y El Naranjo) están conectados por enlaces VHF, y uno (Limones) por un enlace Wi-Fi.

3.3. RED EHAS PERÚ CUSCO

A continuación se describe una de las redes EHAS Perú, la red EHAS Perú Cusco, ya que en esta red se realizaron pruebas de funcionamiento de la herramienta desarrollada y fue una de las primeras redes en la que se instaló el sistema de gestión de redes EHAS como se va a comentar en el capítulo 5.

La red EHAS Perú Cusco [58] comunica 12 puestos y centros de salud de las provincias de Quispicanchi y Acomayo del Departamento de Cusco, los cuales se muestran en la figura 31.

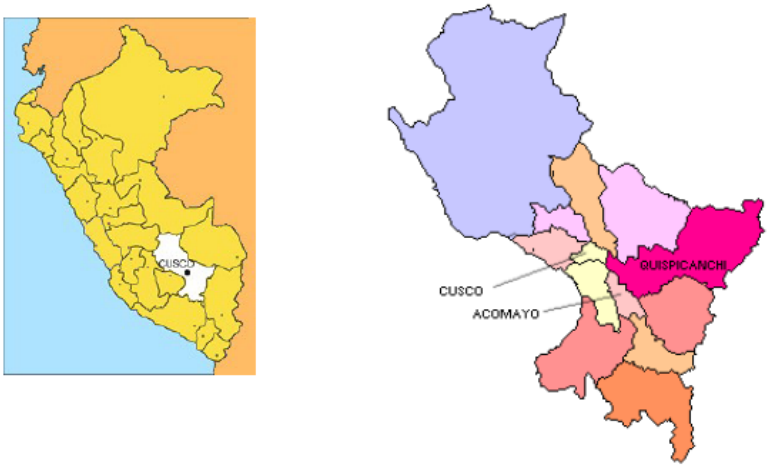


Figura 31. Mapa de Cusco [http://images.google.com.co/]

Los establecimientos de salud son: DIRESA de Cusco, Ccatcca, Kcaury, Urpay, Urcos, Acopia, Pomacanchi, Marcaconga, Sangarará, Acomayo, Acos y Pillpinto, y los repetidores que conectan estos establecimientos de salud son: Hospital Cusco, Josjojahuarina1, Josjojahuarina2, Huiracochán, Don Juan, Laykatuyoc y Huascar. En la figura 32 se muestra un diagrama de la red EHAS Perú Cusco.

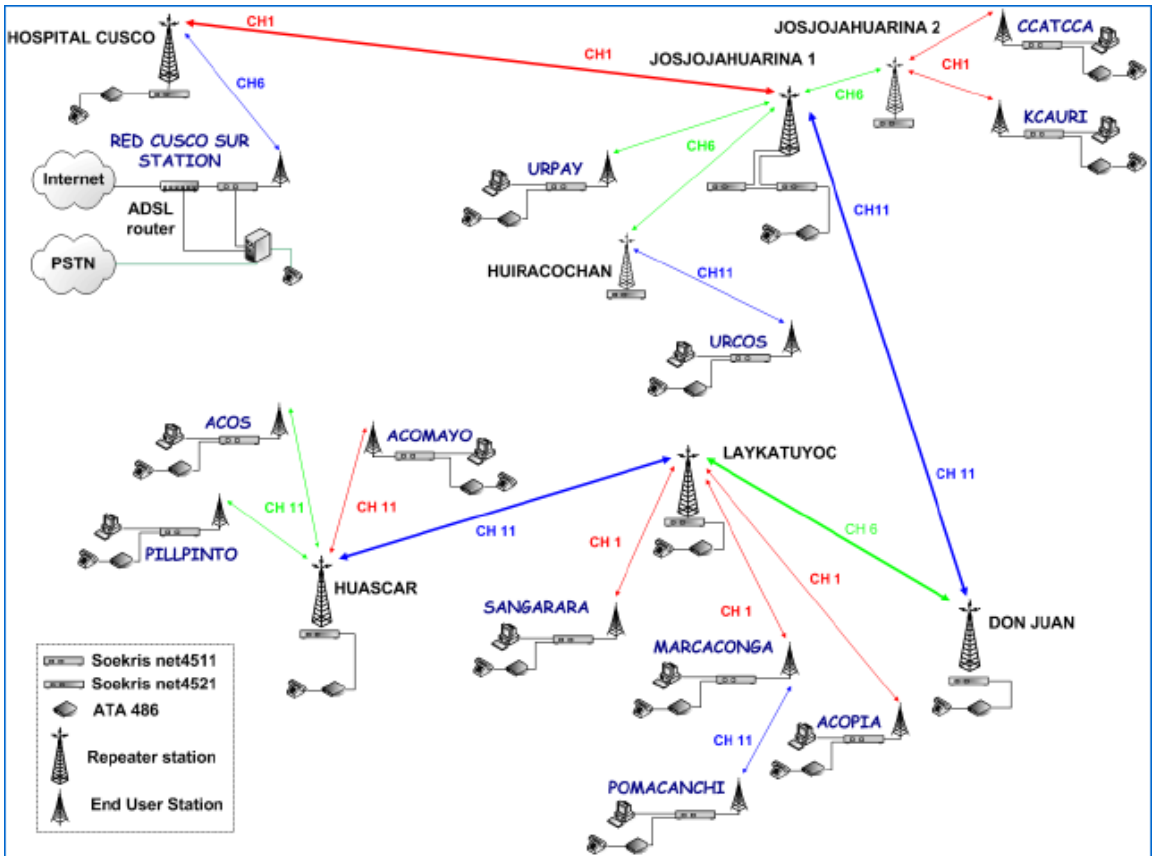


Figura 32. Red EHAS Perú Cusco [58]

En la tabla3 se muestra las distancias entre los siete enrutadores de la red EHAS Perú Cusco.

Tabla 3. Distancia entre puestos de salud Cusco

Desde	Hasta	Distancia (Km)
Hospital Cusco	Josjojahuarina1	39.2
Josjojahuarina1	Josjojahuarina2	0.4
Josjojahuarina1	Huiracochan	6.1
Josjojahuarina1	Don Juan	41.5
Don Juan	Laykatuyoc	17.4
Laykatuyoc	Huascar	10.2

3.4. ALTERNATIVAS PARA LAS REDES EHAS

Por otro lado, teniendo en cuenta que lo que busca la Fundación EHAS es lograr comunicaciones de larga distancia de buena calidad, una alternativa para las redes EHAS puede ser WiMAX, cuando las implementaciones de esta tecnología sean más estables y de bajo costo. Por otro lado, debido a la naturaleza desconectada de las redes EHAS, otra alternativa para las redes EHAS pueden ser las DTNs, ya que este tipo de redes soportan conectividad intermitente y largos retardos en los enlaces, claro que es una tecnología reciente que aún está en proceso de desarrollo. A continuación se describe

WiMAX y DTN y su gestión a través de SNMP, el estándar de gestión de red más simple y utilizado.

3.4.1. WiMAX. WiMAX [59] está definido como Interoperabilidad Mundial para Acceso por Microondas (Worldwide Interoperability for Microwave Access) por el WiMAX Forum, formado para promover la conformidad e interoperabilidad del estándar IEEE 802.16, oficialmente conocido como WirelessMAN. WiMAX busca proporcionar datos inalámbricos sobre largas distancias, en una variedad de modos diferentes, desde enlaces punto a punto a acceso tipo celular móvil total. El Foro describe WiMAX como una tecnología basada en estándares que permite la entrega de acceso de banda ancha inalámbrico de última milla como una alternativa al cable y DSL. Actualmente, se habla de WiMAX fijo y WiMAX móvil.

WiMAX fijo. Se utiliza frecuentemente para hacer referencia a sistemas que se construyen utilizando 802.16-2004 (frecuentemente llamado 802.16d) como la tecnología de la interfaz aérea.

WiMAX móvil. Se utiliza para hacer referencia a sistemas que se construyen utilizando 802.16e-2005 (frecuentemente llamando 802.16e) como la tecnología de la interfaz aérea. Algunas compañías celulares están evaluando WiMAX como un medio para incrementar el ancho de banda de una variedad de aplicaciones intensivas de datos. En línea con estas posibles aplicaciones está la habilidad de la tecnología para servir como un backhaul de alto ancho de banda para Internet o tráfico telefónico celular desde áreas remotas detrás de un backbone Internet.

En áreas sin cable físico o redes telefónicas pre-existentes, WiMAX puede ser una alternativa viable para acceso de banda ancha. Dada la infraestructura cableada limitada en algunos países en desarrollo, los costos de instalar una estación WiMAX o incluso como un hub solitario son probablemente más pequeños en comparación a desarrollar una solución cableada. Áreas de baja densidad de población y terreno plano son particularmente adecuadas para WiMAX. Para países que no tienen mucha infraestructura cableada como resultado de los costos prohibitivos y difícil geografía, WiMAX puede mejorar la infraestructura inalámbrica de una forma efectiva, descentralizada, de fácil despliegue, y en un futuro no lejano a bajos costos.

Las comparaciones entre WiMAX y Wi-Fi son frecuentes, pero ambos estándares están dirigidos a aplicaciones diferentes. WiMAX es un sistema de largo rango (muchos kilómetros) que utiliza espectro licenciado y no licenciado para entregar una conexión punto a punto a Internet desde un ISP al usuario final.

Una idea falsa comúnmente mantenida es que WiMAX entrega 70 Mbits/seg sobre 48 kilómetros. Cada uno de estos datos es verdad individualmente, dadas circunstancias ideales, pero no son verdaderos simultáneamente

Existen diferentes asociaciones como WiMAX Forum y WiMAX Spectrum Owners Alliance, WiSOA, WiBro.

Gestión WiMAX. Ya que el tema central de este trabajo es la gestión de redes, se buscó posibilidades de gestionar WiMAX y se encontró que IEEE 802.16i [60] especifica un Módulo MIB que define objetos gestionados para Estaciones Suscriptoras y Estaciones Base con base en IEEE 802.16-2004 e IEEE 802.16e-2005. La MIB contiene objetos

gestionados que son comunes tanto para redes inalámbricas de banda ancha fijas como móviles.

3.4.2. DTN. Teniendo en cuenta la naturaleza desconectada de las redes EHAS, una opción que podría ser interesante son las Redes Tolerantes al Retardo, DTN [61] (Delay Tolerant Network). A continuación se encuentra una descripción de este tipo de redes que demuestra sus potencialidades para las redes EHAS.

Una Red Tolerante al Retardo, DTN (Delay Tolerant Network) es una red de redes regionales que permite la interoperabilidad de estas redes ya que adapta largos retardos dentro y entre ellas, y convierte características de comunicación entre estas redes, lo cual les permite soportar la movilidad y potencia limitada de los dispositivos de comunicación inalámbricos actuales.

Las tecnologías de DTNs inalámbricas pueden ser diversas, incluyendo no solamente Radio Frecuencia, RF (Radio Frequency) sino también Banda Ultra Ancha, UWB (Ultra Wide Band), tecnologías acústicas (sonar o ultrasónicas) y ópticas de espacio libre.

Muchas redes no cumplen con los requerimientos de Internet sino que por el contrario se caracterizan por:

- Conectividad intermitente: si no existe un trayecto extremo a extremo entre fuente y destino, es decir, si existe un particionamiento de la red, la comunicación que utiliza TCP/IP no funciona y se requieren otros protocolos.
- Retardo largo y variable: además de la conectividad intermitente, los largos retardos de propagación entre nodos y los retardos variables en los nodos contribuyen a los retardos de trayecto extremo a extremo que pueden afectar los protocolos y aplicaciones de Internet que dependen del retorno rápido de reconocimientos o datos.
- Tasas de datos asimétricas: Internet soporta asimetrías moderadas de la tasa de datos bidireccional ya que las asimetrías grandes afectan los protocolos conversacionales.
- Altas tasas de error: los errores de bits en enlaces requieren corrección, que implica más bits y más procesamiento, o retransmisión del paquete entero, que genera más tráfico en la red. Para una tasa de error de enlace dada, se necesitan más pocas retransmisiones salto por salto que extremo a extremo.

Las DTNs resuelven los problemas asociados con conectividad intermitente, retardos largos o variables, tasas de datos asimétricas y altas tasas de error al utilizar conmutación de mensajes con almacenamiento y reenvío, un método bastante antiguo. Todos los mensajes, es decir, bloques enteros de datos de usuario de un programa de aplicación, o piezas, fragmentos de tales mensajes se mueven o reenvían de un lugar de almacenamiento en un nodo, switch, a un lugar de almacenamiento en otro nodo, a lo largo de un trayecto que eventualmente alcanza el destino.

Capa bundle. La arquitectura DTN implementa conmutación de mensajes con almacenamiento y reenvío al sobreponer una nueva capa de protocolo llamada capa bundle sobre las capas más bajas específicas a la región heterogéneas. La capa bundle une estas capas de tal forma que los programas de aplicación se pueden comunicar a través de múltiples regiones.

La capa bundle almacena y reenvía bundles, es decir, mensajes, enteros o fragmentos de bundles entre nodos y se utiliza a través de todas las redes o regiones que conforman una

DTN, mientras que las capas por debajo de la capa bundle, capa de transporte e inferiores, se escogen por ser apropiadas al entorno de comunicación de cada región.

En la figura 33 se muestra la capa bundle y se compara las capas de protocolo de Internet con las capas de protocolo de una DTN.

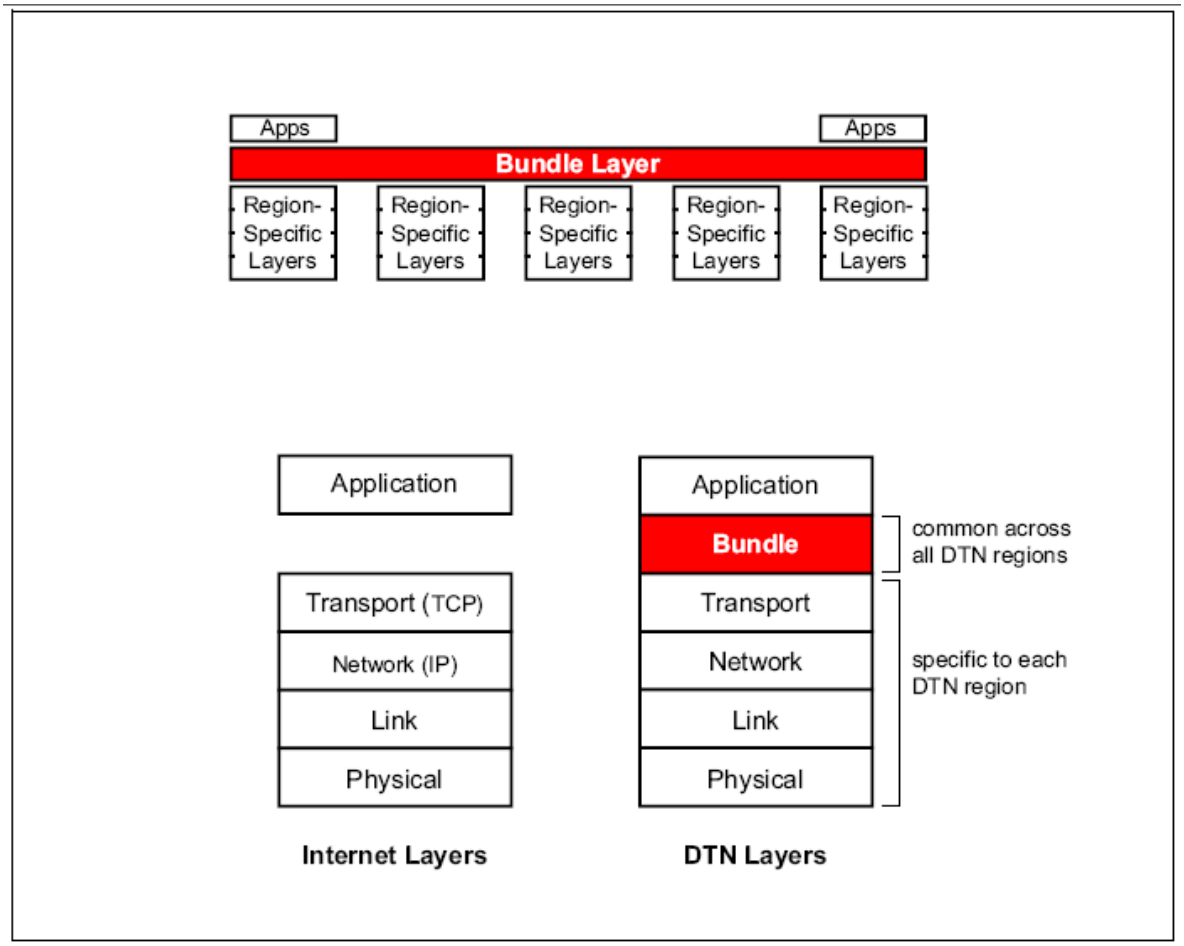


Figura 33. Capa bundle DTN [61]

Bundles. Los bundles consisten de tres partes: la primera, los datos de usuario de una aplicación fuente, la segunda, información de control, proporcionada por la aplicación fuente para la aplicación destino, que describe cómo procesar, almacenar, disponer, y manejar los datos del usuario, y la tercera, un encabezamiento bundle, insertado por la capa bundle. Como los datos de usuario de un programa de aplicación, los bundles pueden ser arbitrariamente largos.

Protocolos no conversacionales. En enlaces conectados intermitentemente con largos retardos, los protocolos conversacionales tales como TCP que involucran muchos round trips extremo a extremo pueden tomar cantidades imprácticas de tiempo o fallar completamente. Por esta razón, las capas bundle se comunican entre ellas utilizando sesiones simples sin o con un mínimo de round trips. Cualquier reconocimiento del nodo receptor es opcional y depende de la clase de servicio seleccionado.

Los protocolos de las capas más bajas que soportan intercambios de capa bundle pueden, por supuesto, ser conversacionales como TCP, pero en enlaces conectados intermitentemente con largos retardos se pueden implementar protocolos de capas más bajas no conversacionales o mínimamente conversaciones.

Nodos DTN. En una DTN, un nodo es una entidad con una capa bundle que puede ser un host, un enrutador o gateway, o alguna combinación, como se muestra en la figura 34.

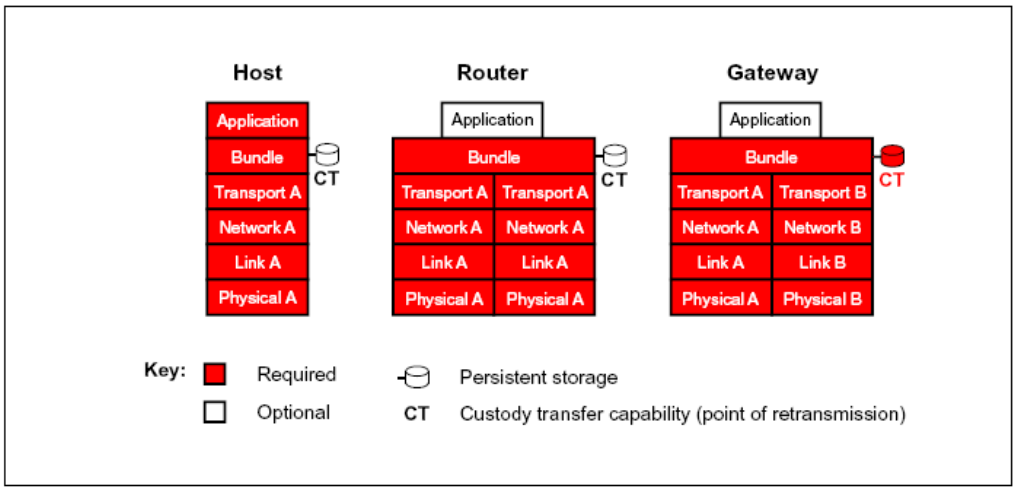


Figura 34. Nodos DTN [61]

- Host: envía y/o recibe bundles, pero no los reenvía. Un host puede ser una fuente o un destino de una transferencia de bundles. La capa bundle de los hosts que operan sobre enlaces de largo retardo requieren almancenamiento persistente para colocar los bundles en colas hasta que los enlaces outbound estén disponibles. Opcionalmente, los hosts pueden soportar transferencias de custodias, que se explicarán más adelante.
- Router: reenvia bundles dentro de una sola región DTN y puede ser un host. La capa bundle de los routers que operan sobre enlaces de largo retardo requieren almacenamiento persistente para colocar los bundles hasta que los enlaces outbound estén disponibles. Opcionalmente, los routers pueden soportar transferencias de custodias.
- Gateway: reenvía bundles entre dos o más regiones DTN y puede ser un host. La capa bundle de los gateways deben tener almacenamiento persistente y soportar transferencias de custodias. Los gateways proporcionan conversiones entre los protocolos de capas más bajas de las regiones que abarcan.

Se necesita almacenamiento persistente, es decir, almacenamiento de mensajes indefinidamente, por una o más de las siguientes razones:

- Un enlace de comunicación al siguiente salto puede no estar disponible por un largo tiempo.
- Un nodo puede enviar o recibir datos mucho más rápido o más confiablemente que el otro nodo.
- Un mensaje, una vez transmitido, puede necesitar ser retransmitido si ocurre un error en un nodo o enlace upstream, hacia el destino, o si uno de estos nodos declina la aceptación de un mensaje reenviado.

Aislamiento del retardo. En Internet, el protocolo TCP proporciona confiabilidad extremo a extremo, es decir, de fuente a destino, al retransmitir cualquier segmento que no sea reconocido por el destino. Las capas física, de enlace y de red proporcionan otros tipos de servicios de integridad de datos. En una DTN, la capa bundle se basa en estos protocolos de capas más bajas para asegurar la confiabilidad de la comunicación.

Sin embargo, en los routers y gateways DTN, que son los nodos que pueden reenviar bundles dentro y entre regiones DTN, respectivamente, las capas bundle actúan como fuentes y destinos extremo a extremo lo que produce que los protocolos de capas más bajas conversacionales de regiones de bajo retardo se aíslen en la capa bundle de largos retardos en otras regiones del trayecto extremo a extremo.

Transferencias de custodias. Las DTNs soportan la retransmisión nodo a nodo de datos perdidos o corruptos tanto en la capa de transporte como en la capa bundle. Sin embargo, debido a que ningún protocolo de capa de transporte, el principal medio para lograr una transferencia confiable, opera extremo a extremo a través de una DTN, la confiabilidad extremo a extremo solamente se puede implementar en la capa bundle.

La capa bundle soporta la retransmisión nodo a nodo por medio de las transferencias de custodias, las cuales se realizan entre las capas bundle de nodos sucesivos por el requerimiento inicial de la aplicación fuente. La capa bundle envía un bundle al siguiente nodo e inicia un timer de retransmisión para el tiempo de reconocimiento. Si la capa bundle del siguiente salto acepta la custodia, retorna un reconocimiento al emisor pero si no retorna ningún reconocimiento antes de que el tiempo para reconocimiento del emisor expire, el emisor retransmite el bundle. El valor asignado al timer de retransmisión para el tiempo de reconocimiento puede ser distribuido a los nodos con información de enrutamiento o computado localmente, con base en la experiencia de transmisión a un nodo particular.

Una capa bundle debe almacenar un bundle hasta que el otro nodo acepta la custodia, o hasta que expira el tiempo de vida del bundle. Se pretende que el tiempo de vida del bundle sea mucho más largo que el tiempo de reconocimiento, si embargo, el tiempo de reconocimiento debería ser lo suficientemente largo para dar a los protocolos de transporte subyacentes la oportunidad de completar una transmisión confiable.

Las transferencias de custodias no proporcionan confiabilidad extremo a extremo garantizada ya que esto solamente se puede lograr si una fuente requiere tanto transferencias de custodias como recepción de retorno. En ese caso, la fuente debe retener una copia del bundle hasta recibir un recibo de retorno y si no lo recibe debe retransmitir.

Movimiento de puntos de retransmisión. La capa bundle utiliza protocolos de capa de transporte confiables junto con transferencias de custodias para mover los puntos de retransmisión progresivamente hacia el destino. El avance de los puntos de retransmisión minimiza el número de saltos de retransmisión potenciales, la consecuente carga adicional de la red causada por retransmisiones y el tiempo total para llevar un bundle confiablemente a su destino.

Esto beneficia las redes con enlaces con largos retardos o con muchas pérdidas. Para trayectos que contienen muchos enlaces con pérdidas, los requerimientos de retransmisiones son mucho más bajos para retransmisiones salto por salto que para

retransmisiones extremo a extremo, incluso, se puede hablar de un incremento lineal versus un incremento exponencial, con respecto a la cuenta de saltos.

Internet versus DTN. En Internet, los protocolos TCP e IP se utilizan a través de la red. TCP opera en los puntos finales de un trayecto y gestiona la entrega extremo a extremo confiable de segmentos de mensajes. IP opera en todos los nodos de un trayecto y enruta datagramas de mensajes. Los enrutadores de Internet no requieren una capa de transporte para enrutar, pero implementan capas de transporte y de aplicación para el mantenimiento de las tablas de enrutamiento y otros propósitos de gestión.

En una DTN, las pilas de protocolo de todos los nodos incluyen tanto la capa bundle como de transporte. Los gateways DTN tienen las mismas capas doble-pila que los enrutadores DTN, pero los gateways pueden ejecutar diferentes protocolos de capas más bajas, es decir, por debajo de la capa bundle, en cada lado de su pila doble. Esto permite a los gateways abarcar dos regiones que utilizan protocolos de capas más bajas diferentes.

Gestión de DTNs. Teniendo en cuenta que este tipo de redes es una muy buena alternativa para las redes EHAS que necesitan ser monitoreadas y controladas, y que SNMP [62] es uno de los estándares más conocido y utilizado en el mundo de la gestión, se decidió analizar la posibilidad de utilizarlo en este tipo de redes.

Por una parte, la latencia excesiva afecta a TCP directamente al limitar severamente su throughput o interferir con el establecimiento de la conexión, por tanto, cualquier protocolo de capa de aplicación que utilice TCP como su transporte subyacente es afectado. La latencia excesiva también afecta adversamente la operación apropiada de protocolos de enrutamiento convencionales. UDP no es sensible a latencia excesiva debido a que no contiene temporizadores que afecten su operación. Sin embargo, protocolos de aplicación que lo requieran necesitarán demasiado tiempo cuando se encuentren con retardo excesivo, disparando fallas de aplicación.

Los largos retardos de propagación y las altas tasas de error hacen el funcionamiento apropiado de la mayoría de protocolos de red comunes, tales como TCP, SCTP, UDP, IP, SNMP, RIP y BGP imposible, ya que retransmisiones frecuentes, expiración de TTLs, enlaces marcados como no operables, y así sucesivamente, los hace abortar todos los canales de comunicación iniciados. El modelo de almacenamiento y envío de las DNTs es capaz de superar todos estos problemas, esto quiere decir que en una red DTN se podría utilizar por ejemplo SNMP para realizar su gestión sin mayores problemas.

4. ARQUITECTURA DE GESTIÓN PARA ENTORNOS HETEROGÉNEOS Y LAS REDES EHAS

En este capítulo se presentan los requerimientos de Linux como proveedor de servicios y una arquitectura basada en este sistema operativo para ofrecer servicios de alta disponibilidad como base para establecer una arquitectura de un servicio de gestión de redes, y se centra la atención en la definición de una arquitectura de gestión para entornos heterogéneos, se realizan adaptaciones de esa arquitectura utilizando los estándares de gestión más utilizados actualmente, y finalmente se plantea una arquitectura de gestión para las redes EHAS.

4.1. CGL

Como se mencionó en el capítulo 3, el sistema operativo más adecuado para enrutadores, servidores e incluso estaciones de usuario EHAS es Linux. Este sistema operativo se está convirtiendo rápidamente en el sistema operativo más utilizado en la industria ya que es capaz de competir con sistemas operativos propietarios a nivel de costo y porque permite proporcionar un valor agregado a la empresa y además ofrece un muy buen desempeño, a tal punto, que actualmente se habla de Linux de Grado de Carrier, CGL (Carrier Grade Linux).

Linux de Grado de Carrier, CGL (Carrier Grade Linux) se encuentra en el centro del movimiento de las arquitecturas abiertas. Alrededor de hace 3 años, un grupo de representantes de la industria de vendedores de plataformas, proveedores de distribuciones Linux y proveedores de equipos de red establecieron definir cómo CGL podría permitir entornos con requerimientos más altos de disponibilidad, nivel de servicio y escalabilidad, entonces se formó el grupo de trabajo en CGL del Laboratorio de Desarrollo de Fuente Abierta, OSDL (Open Source Development Lab). Desde su formación, el grupo de trabajo ha producido dos versiones de una especificación que define estas capacidades requeridas. En respuesta, los proveedores de distribuciones Linux ahora están demostrando que pueden suplir las necesidades emergentes de telecomunicaciones al registrar cómo sus productos satisfacen los requerimientos definidos en la Definición de Requerimientos de CGL [63]. Hoy en día, el grupo de trabajo CGL ha crecido para incluir más de tres docenas de representantes de vendedores de plataformas, proveedores de distribuciones Linux, proveedores de equipos de red y carriers, y miembros de comunidades de desarrollo en todo el mundo

Los componentes del middleware de alta disponibilidad y el middleware de disponibilidad de servicio que se ejecuta en sistemas CGL son manejados por organizaciones tales como el DMTF, OMG y SAF. Las plataformas hardware de alta disponibilidad subyacentes al CGL son manejadas por organizaciones tales como PICMG e IPMI, como se muestra en la figura 35.

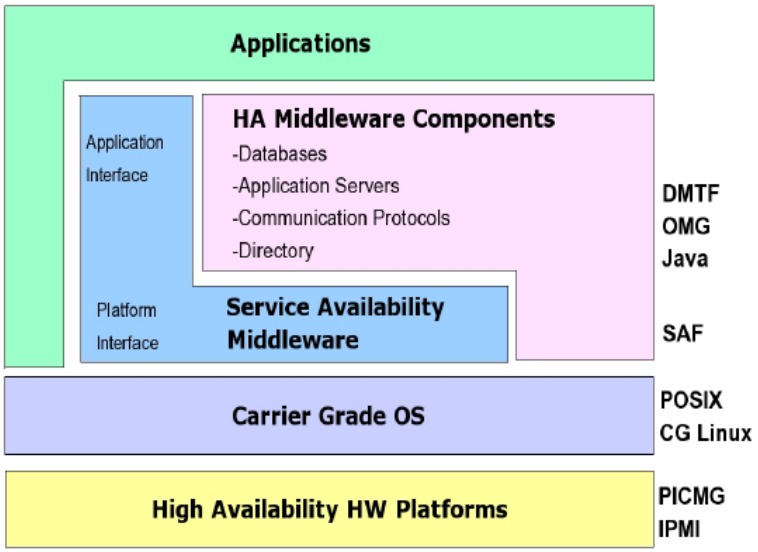


Figura 35. Arquitectura para de servicios alta disponibilidad [63]

Para claridad y facilidad de uso, la especificación ha sido dividida por tema en los siguientes documentos:

- **Resumen de la definición de requerimientos.** Proporciona un resumen de CGL y las especificaciones.
- **Definición de Requerimientos de Disponibilidad.** Describe la funcionalidad útil y necesaria para la disponibilidad y la recuperación de nodos simples.
- **Definición de requerimientos de clústers.** Describe componentes útiles y necesarios para construir un conjunto en clúster de sistemas individuales. El objetivo clave es agrupar para lograr alta disponibilidad, aunque el balance de carga y el desempeño son objetivos secundarios.
- **Definición de requerimientos de capacidad de servicio.** Describe características útiles y necesarias para servir y mantener un sistema y cubre herramientas que soportan capacidad de servicio.
- **Definición de requerimientos de desempeño.** Describe características útiles y necesarias que contribuyen al desempeño adecuado de un sistema, tales como requerimientos de tiempo real. También describe componentes del sistema operativo base para soportar herramientas de desempeño (requerimientos para las herramientas en sí mismas no se direccionan).
- **Definición de requerimientos estándar.** Proporciona referencias a APIs, especificaciones y estándares útiles y necesarios, tales como los estándares del SAF.
- **Definición de requerimientos hardware.** Describe soporte específico al hardware útil y necesario que se relaciona con un entorno operativo carrier.
- **Definición de requerimientos de seguridad versión.** Describe características útiles y necesarias para construir sistemas seguros.

En la figura 35 se presenta una arquitectura muy interesante, adecuada para cumplir los requerimientos de gestión de las redes EHAS. Por otro lado, esta arquitectura también utiliza niveles y los estándares abiertos mencionados en el capítulo 1.

La distribución que se utiliza en EHAS es Debian y desde hace poco Debian GNU/Linux Sarge 3.1 es una distribución registrada CGL, lo que demuestra que la selección de este

sistema operativo es muy adecuada para lograr redes de comunicaciones y servicios de información de calidad que satisfagan completamente las necesidades de los usuarios.

4.2. ARQUITECTURA DE GESTIÓN PARA ENTORNOS HETEROGÉNEOS

Es importante tener en cuenta que así como los requerimientos de los usuarios en cuanto a servicios crecen, las exigencias en cuanto a las características de las redes también aumentan rápidamente. Por este motivo, las grandes empresas, y grupos de investigación y desarrollo, líderes en el sector de la computación y la comunicación tienen una gran preocupación y dedican gran parte de sus recursos a crear o a participar en la creación de estándares, arquitecturas, plataformas y herramientas de gestión de redes que permitan monitorear y controlar las redes de comunicaciones de una forma adecuada, más aún si se tienen en cuenta los entornos heterogéneos que conforman las redes de comunicaciones actuales. Pero este interés ha hecho que exista un gran abanico de posibilidades que puede llegar a ser confuso y complejo de manejar, incluso abrumador para los encargados de la gestión de redes.

Tal es la situación, que existen muchos estándares que aunque pueden ser buenos no llegan a ser totalmente conocidos, no se realiza una implementación de ellos y finalmente no cuentan con el respaldo suficiente y se abandonan. Algunos grupos prefieren crear un nuevo estándar o incluso desarrollar una nueva herramienta que aunque no cumple con ningún estándar es adecuada para un escenario concreto y finalmente se posiciona como una de las más utilizadas para gestionar redes. Todo esto incrementa el número de posibilidades para realizar la gestión de redes hasta el punto que muchos prefieren regresar al tradicional SNMP y por tanto la evolución de las tecnologías de gestión de redes se vuelve una tarea muy compleja.

Como se describió en el capítulo 1, actualmente existen muchos estándares de gestión, y se podría considerar que una tecnología tiene ventajas sobre otra, pero esto puede depender del entorno que se desee gestionar, o se podría pensar que una tecnología es el reemplazo de otra, pero si realiza el mapeo adecuado entre ellas, como se mostró en el capítulo 2, se podrían complementar e integrar para conseguir una arquitectura de gestión adecuada para un entorno determinado y lograr la satisfacción del usuario. Este panorama, un tanto complejo, es al que se enfrentan los encargados de la gestión de redes, pero además, como se mencionó anteriormente, muchos otros grupos interesados en el tema de gestión de redes no crean estándares sino herramientas, y por otro lado, los desarrolladores de Linux también proporcionan muchas herramientas para lograr la gestión adecuada de este sistema operativo que cada día gana más y más usuarios, lo que hace que el panorama para la elección de una tecnología de gestión de redes se vuelva aún más complejo.

Tal es la preocupación, que muchos grupos de investigación y desarrollo, y cada vez más empresas interesadas en lograr una alta utilización de sus productos, participan en la creación y fortalecimiento de estándares de gestión de redes, en el estudio de posibilidades para lograr la integración de estos estándares y en el desarrollo de herramientas de gestión de redes que permitan obtener un sistema de gestión de redes abierto. Por otro lado, grupos como OSDL (Open Source Development Labs), que buscan lograr el mayor uso del sistema operativo Linux, están muy interesados en el tema de gestión de redes para lograr su meta al demostrar que este sistema es fácilmente gestionable, además, otros grupos también interesados en este sistema operativo, como por ejemplo, los desarrolladores de drivers de tarjetas de red, ofrecen herramientas que son muy utilizadas por los usuarios para configurar las comunicaciones. Todo esto

sucede al mismo tiempo que los grupos de investigación y desarrollo en tecnologías de redes continúan su trabajo para lograr la evolución y mejora estas tecnologías, lo que incrementa el interés por las alternativas de gestión de redes.

Actualmente existen grupos de investigación y desarrollo así como también empresas que tienen un gran interés por el tema de la gestión abierta, es decir, la que utiliza estándares abiertos y en muchos casos se dirige a sistemas operativos abiertos como Linux, a pesar de lo complejo que esto puede llegar a ser. A continuación se presenta una breve descripción del trabajo realizado por algunos de los grupos más importantes en esta área.

Este trabajo buscó determinar las posibilidades de integración de los estándares, arquitecturas, plataformas y herramientas de gestión para obtener lo mejor de ellas, lograr su simbiosis y sinergia, y de esta forma conseguir una arquitectura de gestión de redes adecuada a los entornos heterogéneos actuales que permita estar un paso más cerca de un verdadero ecosistema de tecnologías de gestión de redes.

En este proyecto se planteó todo un reto de investigación y desarrollo, la obtención de una arquitectura de gestión de redes adecuada para los entornos heterogéneos actuales y además, su aplicación a la redes EHAS.

Como se puede concluir por lo visto en el capítulo 3, para que la Fundación EHAS cumpla con su objetivo es fundamental que las redes EHAS funcionen adecuadamente la mayor parte del tiempo, pero una vez que se instalaban no se conocía su configuración, desempeño ni sus fallas, no se podía detectar sus problemas ni sus causas oportunamente y por tanto no se podían resolver eficiente y eficazmente, incluso, no se sabía cuando se caía un equipo y podían pasar días antes de saberlo, es decir, las redes podían permanecer fuera de servicio por mucho tiempo sin que se tomara alguna acción al respecto, lo que cobra mayor importancia debido a que la Fundación EHAS trabaja con comunidades aisladas que generalmente no tienen mucho contacto con sistemas informáticos y pueden perder fácilmente la credibilidad y confianza en el proyecto.

En consecuencia, se observó la necesidad imperiosa de conocer la disponibilidad de las redes EHAS así como también su uso, ya que esto da una medida del impacto de las tecnologías de información y comunicaciones en las comunidades. Además, también se encontró muy importante determinar si las redes funcionan como se espera ya que la Fundación EHAS realiza investigación en tecnologías de información y comunicaciones de bajo costo y es necesario saber si el camino de investigación que la Fundación ha elegido es adecuado y por tanto si está cumpliendo su objetivo. Por otro lado, se pensó en que sería importante poder realizar tareas de control en forma remota ya que las redes se encuentran en zonas apartadas, y en algunas ocasiones, aunque el personal técnico de los puestos y centros de salud se capacita para atender los sistemas instalados se presentan situaciones que requieren la atención por parte de personal con mayor experiencia que de no ser por este mecanismo conllevarían a un alto consumo de tiempo y dinero.

Esto hizo evidente la necesidad de un sistema de gestión de redes EHAS que permitiera obtener información relevante de los equipos de las redes, como por ejemplo, datos de las interfaces de red inalámbricas, que desplegara los datos en tablas y gráficos, así como también, que generara alarmas cuando se cayera un equipo o se sobrepasaran ciertos umbrales, también, que permitiera la ejecución de comandos, y por otro lado, que

realizara la adición de los equipos automáticamente, y que además se accediera vía Web y fuera fácil de utilizar para cualquier usuario. Teniendo en cuenta esto, se planteó la obtención de un sistema de gestión que permitiera realizar tareas de monitoreo y control sobre las redes EHAS de forma adecuada, por tanto, se tuvo en cuenta que las redes EHAS son muy diferentes a las redes tradicionales, pero precisamente esto hizo que este proyecto de investigación fuera diferente a los demás proyectos de gestión de redes.

En este proyecto se tuvo en cuenta que las redes EHAS, como se mencionó en el capítulo 3, combinan tecnologías como Wi-Fi, VHF y HF, y además tienen dos características muy importantes. La primera, es que tienen bajo ancho de banda, incluso los enlaces Wi-Fi debido a que son de larga distancia, y la segunda, es que algunos equipos pueden no estar conectados todo el tiempo a la red, sobre todo las estaciones VHF y HF debido al gran consumo de energía que esto implicaría. Por otro lado, los organismos de estándares de gestión ofrecen diferentes estándares e iniciativas, pero además, existen herramientas y otras opciones que pueden ser una buena alternativa para gestionar las redes EHAS teniendo en cuenta sus características.

En el caso de las redes EHAS, el gestor generalmente no tiene conectividad permanente con los equipos gestionados, un escenario totalmente diferente a los escenarios comunes de gestión de redes, por tanto, el objetivo de este proyecto fue obtener un sistema de gestión adecuado a las redes EHAS, una solución muy diferente a las convencionales ya que las redes EHAS no son como la mayoría de las redes, pero precisamente en esto radica la importancia de este proyecto, ya que una solución adecuada es aquella que tiene en cuenta las necesidades reales, los condicionantes del entorno y en este caso, las características particulares y un tanto complejas de las redes EHAS que son las que hacen a este proyecto de investigación bastante interesante.

Al tener en cuenta las iniciativas de los diferentes grupos de investigación y desarrollo en gestión como el Consorcio de Gestión Abierta, el proyecto Gestión Abierta con CIM y la empresa AVOCENT, además el proyecto TOMAS³ de Ándago, una arquitectura de gestión adecuada para entornos heterogéneos es aquella que utiliza tanto estándares abiertos como herramientas de software libre y de fuente abierta. El trabajo realizado por PICMG, SAF, DMTF, OASIS muestra que es posible utilizar otros estándares aparte de SNMP como IPMI, HPI, AIS, WSDM, WS-Management, WBEM, SAMSH, DASH, etc. y que además existen diferentes posibilidades de integración entre ellos. Esto demuestra que existe una evolución constante en el campo de la gestión y que es necesario estar a la vanguardia en tecnologías de gestión para lograr soluciones de gestión realmente adecuadas. La utilización de estos estándares está respaldada por el trabajo de investigación llevado a cabo por el OGF para la gestión de entornos computacionales Grid y también por el trabajo de OSDL en CGL. Pero además, existen numerosas herramientas de gestión de fuente abierta que se ejecutan sobre Linux, y muchas utilidades que este sistema operativo ofrece para administrarlo y gestionarlo, que también pueden ser alternativas para lograr un sistema de gestión de redes adecuado para entornos heterogéneos y en este caso en particular para las redes EHAS. En conclusión, una arquitectura de gestión adecuada para entornos heterogéneos es aquella que utiliza estándares abiertos, herramientas de fuente abierta y en el caso particular de EHAS para sistemas abiertos.

Teniendo en cuenta los estándares de gestión que se estudiaron en el capítulo 1, y sus posibilidades de integración que se analizaron en el capítulo 2, y además, que en las redes EHAS se utiliza el sistema operativo Linux Debian que cumple con los

requerimientos de CGL de OSDL, se estableció una arquitectura de gestión con varios niveles como se muestra en la figura 36.

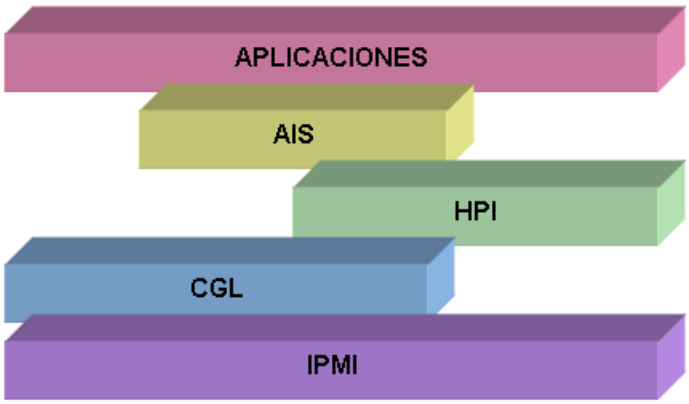


Figura 36. Arquitectura de gestión para entornos heterogéneos

Como se observa en la figura 36, en el nivel más bajo se encuentra IPMI que define un subsistema hardware independiente del sistema gestionado y que permite gestionar su hardware de una forma estándar. Sobre este nivel se encuentra el sistema operativo, que si es Linux debe cumplir con los requerimientos de CGL de OSDL. Sobre este nivel se encuentra HPI, que permite gestionar el hardware de una forma estándar, independientemente del hardware y de las aplicaciones. HPI se encuentra sobre el sistema operativo o también sobre IPMI ya que existe un mapeo de IPMI a HPI. Sobre este nivel se encuentra AIS, que permite lograr la independencia de la aplicación, y se encuentra sobre el sistema operativo o también sobre HPI. En el último nivel se encuentran las aplicaciones, que por supuesto se encuentran sobre el sistema operativo, o también sobre AIS e incluso sobre HPI. En este caso, las aplicaciones de gestión pueden ser aplicaciones SNMP, WBEM, SMASH, DASH, etc, entonces a continuación se presenta esta arquitectura adaptada a los estándares de gestión más importantes actualmente y se analiza para cada una de ellas los modelos de información, organizacional, de comunicación y funcional que son los modelos que definen cualquier arquitectura de gestión.

4.2.1. SNMP. SNMP es uno de los estándares de gestión más conocidos y utilizados en el mundo de la gestión gracias a su simplicidad, por lo que generalmente es la primera alternativa de una solución de gestión y muchas veces es la más adecuada, pero si este no es el caso, para aprovechar la gran difusión de este estándar, es importante buscar la forma en que la solución planteada se puede integrar con él, y precisamente por esta razón, muchos de los estándares de gestión actuales analizan esta posibilidad como lo hacen HPI y ASI. Como se presentó en el capítulo 2, HPI y AIS se mapean a SNMP, es decir, se definen MIBs SNMP para estos dos estándares y por tanto se puede crear agentes SNMP que utilicen estas MIBs y que se comuniquen mediante SNMP con el gestor. En la figura 37 se muestra la arquitectura con SNMP.

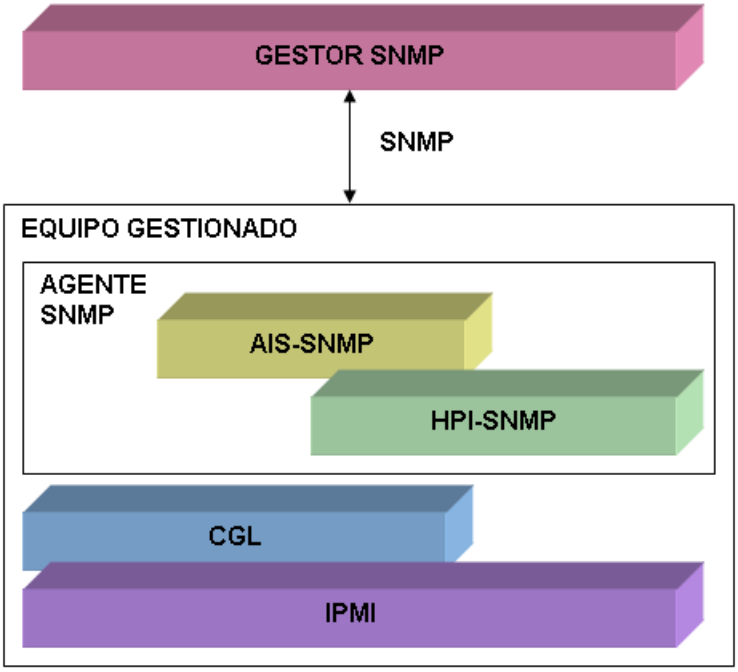


Figura 37. Arquitectura de gestión con SNMP

Como se observa en la figura 37, el agente SNMP aprovecha el mapeo de HPI y AIS a SNMP, pero por supuesto, el agente también podría trabajar directamente con el sistema operativo para gestionar el sistema. Debido a los mapeos de IPMI a HPI, y de HPI y AIS a SNMP, los modelos de esta arquitectura corresponden a los de SNMP y se describen a continuación.

Modelo de información. El modelo de información en SNMP no es orientado a objetos porque no tiene el concepto de clases, solo de variables, y consta de MIBs que describen los recursos gestionados. Por otro lado, la Estructura de Información de Gestión, SMI (Structure of Management Information) especifica que todos los objetos gestionados deben tener asociado a ellos un nombre, OID, una sintaxis, ASN.1, y una codificación, BER (Basic Encoding Rules). Además, para garantizar la identificación única de cada variable de gestión, SMI introduce el concepto de Árbol de Nombrado.

Modelo organizacional. SNMPv1 utiliza el modelo gestor-agente centralizado. RMON (Remote Monitoring) utiliza gestión jerárquica. SNMPv2 y SNMPv3 pueden seguir el paradigma de control multicentro.

Modelo de comunicación. En SNMP, la comunicación entre gestores y agentes se hace a través de diferentes tipos de mensajes enviados asincrónicamente dentro de diferentes PDUs SNMP. SNMP soporta un estilo de comunicación no orientado a la conexión, y aunque no existen requerimientos con respecto al protocolo sobre el que se envía, está diseñado para correr sobre UDP, y por lo tanto éste es el más utilizado.

Modelo funcional. En SNMP no se define un modelo funcional, la funcionalidad de gestión está implícitamente contenida en las MIBs.

4.2.2. SNMP-AgentX. SNMP también se puede utilizar en una arquitectura con un agente y subagentes SNMP. En este caso se tiene un subagente HPI y un subagente AIS que se comunican mediante AgentX con el agente maestro, el cual se comunica mediante SNMP con el gestor. En la figura 38 se muestra la arquitectura con SNMP y AgentX.

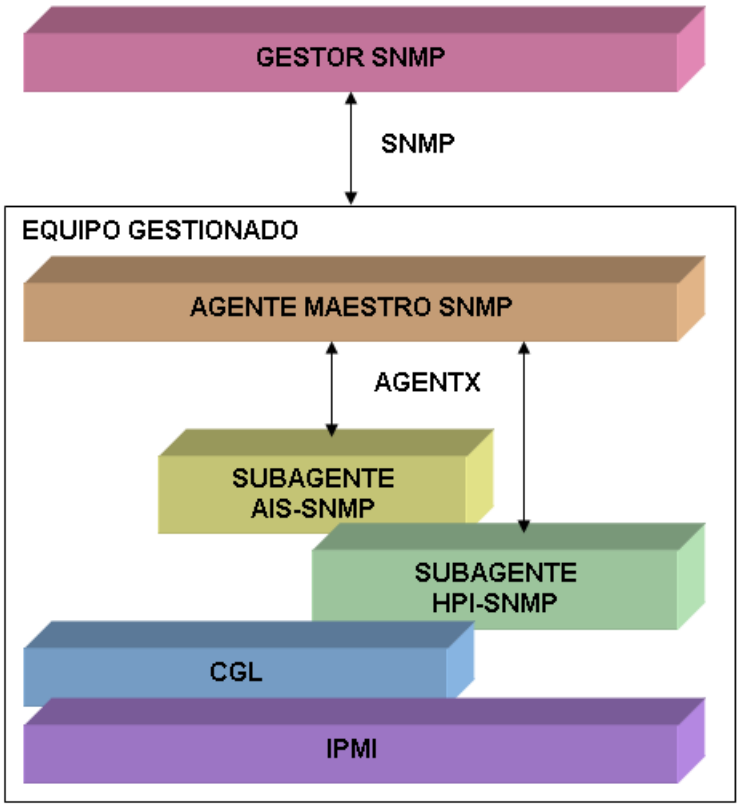


Figura 38. Arquitectura de gestión con SNMP y AgentX

Como se observa en la figura 38, el agente maestro se puede comunicar con los subagentes SNMP HPI y AIS, los cuales utilizan el mapeo de HPI y AIS a SNMP, pero también podría trabajar con el sistema operativo para gestionar el sistema. En este caso, el modelo de información, organizacional y funcional de esta arquitectura corresponden con los mencionados anteriormente para SNMP, pero el modelo de comunicación incluye además el protocolo AgentX para la comunicación entre el agente maestro y los subagentes.

4.2.3. WBEM. WBEM es uno de los estándares más importantes en el mundo de la gestión actualmente. Los grupos de investigación y desarrollo continúan trabajando fuertemente en este estándar y cada día, más y más empresas le apuestas a la gestión vía Web utilizando esta tecnología. Una de las principales fortalezas de WBEM es CIM, que es uno de los modelos de información más utilizados por sus características orientadas a objetos, y es tal su importancia, que muchos estándares definen su integración con este estándar como es el caso de HPI y AIS. Como se presentó en el capítulo 2, HPI y AIS se mapean a CIM, lo que permite tener funcionalidad AIS y HPI en Esquemas CIM y por tanto construir proveedores que utilicen estos esquemas CIM y se comuniquen con el CIMOM, el cual a su vez se comunica con el gestor a través de HTTP. En la figura 39 se muestra la arquitectura con WBEM.

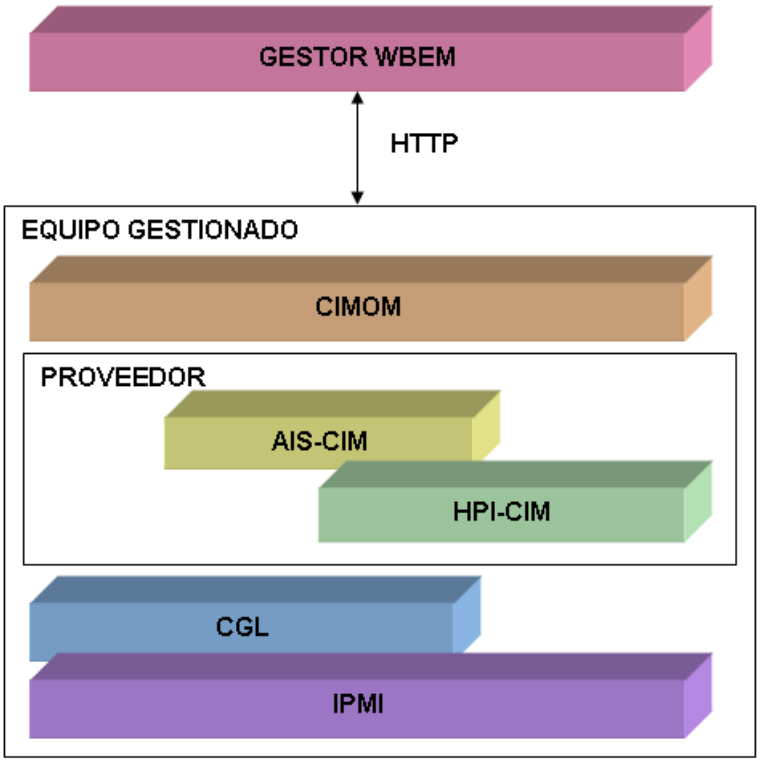


Figura 39. Arquitectura de gestión con WBEM

Como se observa en la figura 39, el proveedor aprovecha el mapeo de HPI y AIS a CIM, pero además puede trabajar con el sistema operativo y manejar muchos más esquemas CIM. Por otro lado, se podría tener un proveedor HPI y otro AIS independientes o juntos sin ningún problema. Gracias al mapeo de HPI a IPMI, y de HPI y AIS a CIM, los modelos de esta arquitectura corresponden a los de WBEM y se describen a continuación.

Modelo de Información. El estándar CIM es el Modelo de Información de WBEM. CIM utiliza técnicas orientada a objetos y está conformado por una Especificación y un Esquema como se mencionó en el capítulo 1. El lenguaje estándar utilizando para definir los elementos de CIM es el Formato de Objetos Gestionados, MOF (Managed Object Format) y además los modelos CIM se representan en UML (Uniform Modelling Language).

Modelo Organizacional. WBEM sigue el modelo gestor-agente y se puede implementar ya sea gestión central o control multipunto que a su vez puede ser control multicentro o gestión jerárquica, todo depende de las capacidades de las aplicaciones de gestión.

Modelo de Comunicación. xmlCIM y el estándar Operaciones CIM sobre HTTP conforman el modelo de comunicación. La razón para representar CIM en XML fue el hecho de que XML fue y es el principal formato para representar datos estructurados en Internet, y la meta de WBEM fue y es utilizar los estándares existentes basados en Web tanto cómo sea posible. Operaciones CIM sobre HTTP es una especificación sobre cómo intercambiar información CIM sobre el protocolo HTTP y ofrece la ventaja de que todas

las implementaciones cliente WBEM que lo utilizan se pueden comunicar con cualquier CIMOM que cumpla con este estándar.

Modelo Funcional. WBEM divide la gestión en varias áreas definidas en el Modelo Común del Esquema CIM. Hasta ahora se han establecido las siguientes: Aplicaciones, Bases de datos, Dispositivos, Eventos, Interoperación, Métricas, Red, Físico, Políticas, Soporte, Sistemas, Usuario. Desde el punto de vista OSI, se puede decir que WBEM cubre las FCAPS.

4.2.4. WBEM-SNMP. Hasta ahora se ha hablado de arquitecturas que utilizan SNMP y WBEM en forma independiente, pero es posible lograr una arquitectura en la que se integren estos dos estándares ya que se puede construir un proveedor SNMP que aproveche el mapeo de HPI y AIS a SNMP para obtener información del sistema o los sistemas gestionados a través de SNMP. El proveedor SNMP al igual que cualquier otro proveedor se comunica con el CIMOM, el cual a su vez se comunica con el gestor mediante HTTP como se muestra en la figura 40.

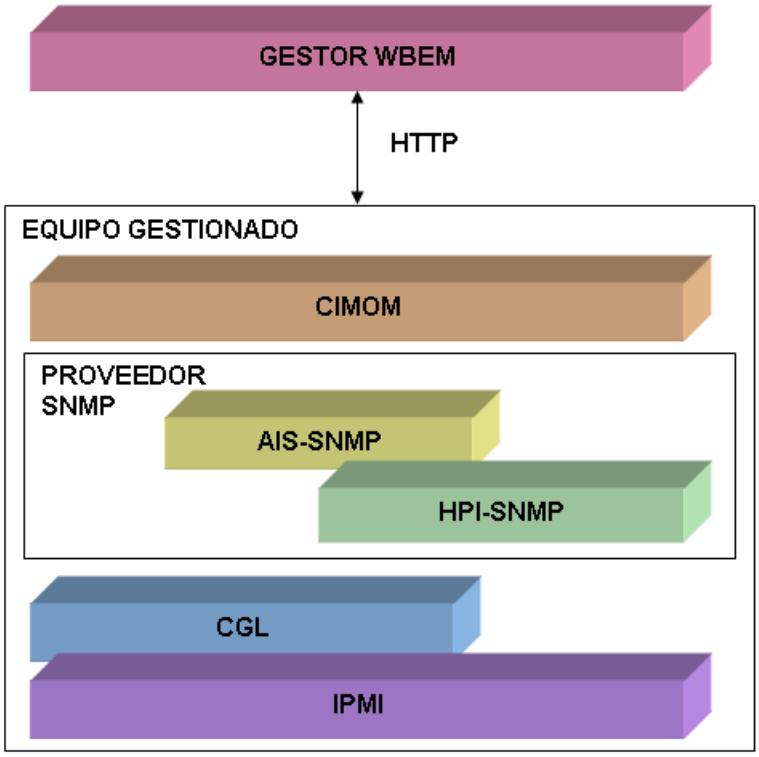


Figura 40. Arquitectura de gestión con WBEM y SNMP

En este caso, ya que WBEM se extiende mediante SNMP, el modelo organizacional corresponde con el de WBEM, pero los modelos de información, de comunicación y funcional integran los modelos de las tecnologías.

En las arquitecturas que utilizan WBEM, el CIMOM se puede encontrar en el sistema gestionado junto con los proveedores pero también puede estar en otro equipo, y se comunicaría con el o los proveedores mediante algún protocolo establecido.

4.2.5. WBEM-SMASH. Como se mencionó en el capítulo 2, SMASH es un conjunto de especificaciones que entregan semánticas arquitecturales, protocolos estándar de la industria y perfiles para unificar la gestión del centro de datos. SMASH permite la gestión simple e intuitiva de servidores heterogéneos en el centro de datos independiente del estado de la máquina, el estado del sistema operativo, etc., es decir, la gestión en o fuera de banda, y en o fuera de servicio. SMASH utiliza CIM y CLP, un protocolo comando/respuesta transmitido y recibido sobre un protocolo de transporte basado en mensajes de texto. Adicionalmente, el DMTF ha colocado especial interés en el desarrollo de este estándar para lograr implementaciones livianas. En la figura 41 se muestra la arquitectura que utiliza SMASH en un contexto WBEM.

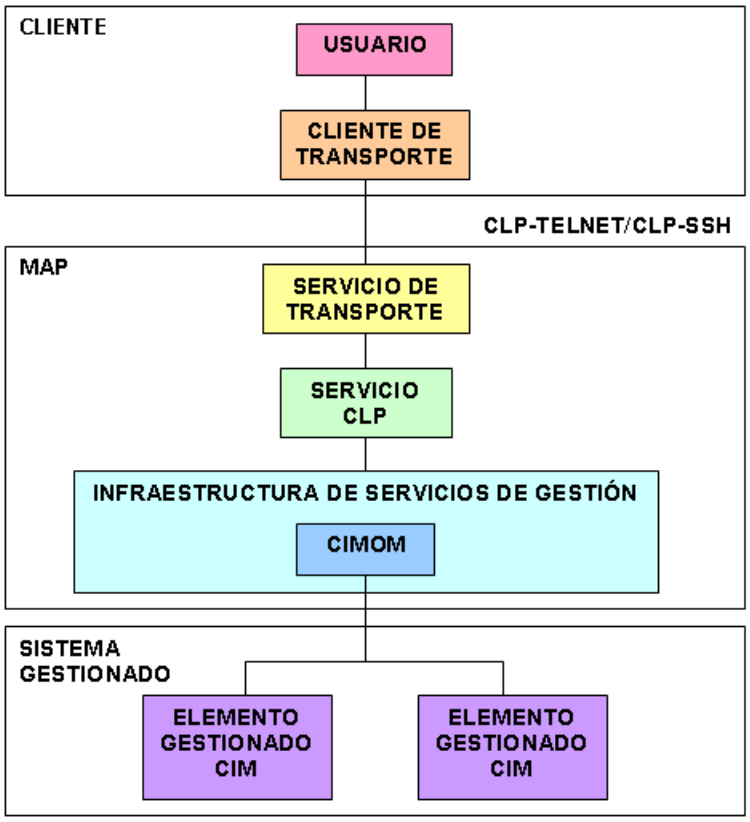


Figura 41. Arquitectura de gestión con WBEM y SMASH

Al igual que el CIMOM WBEM, el MAP puede estar en el sistema gestionado o en otro equipo. Por otro lado, teniendo en cuenta que en este caso SMASH extiende a WBEM, los modelos de información, organizacional y funcional de esta arquitectura corresponden a los de WBEM pero el modelo de comunicación incluye CLP sobre Telnet, SSH o incluso otros protocolos basados en mensajes de texto.

4.2.6. WBEM-DASH. DASH es un conjunto de especificaciones que establecen semánticas arquitecturales, protocolos estándar de la industria, y un conjunto de perfiles para estandarizar la gestión de sistemas de escritorio y móviles independiente del estado de la máquina, plataforma operativa o vendedor, y permite la gestión en o fuera de banda, y en o fuera de servicio. DASH utiliza WS-Management y CIM, y al utilizar protocolos estándar facilita la interoperabilidad, y además incrementa la simplicidad y funcionalidad

en soluciones de gestión de empresa heterogéneas. Adicionalmente, el DMTF ha colocado especial interés en el desarrollo de este estándar para lograr implementaciones livianas. En la figura 42 se muestra la arquitectura que utiliza DASH en un contexto WBEM.

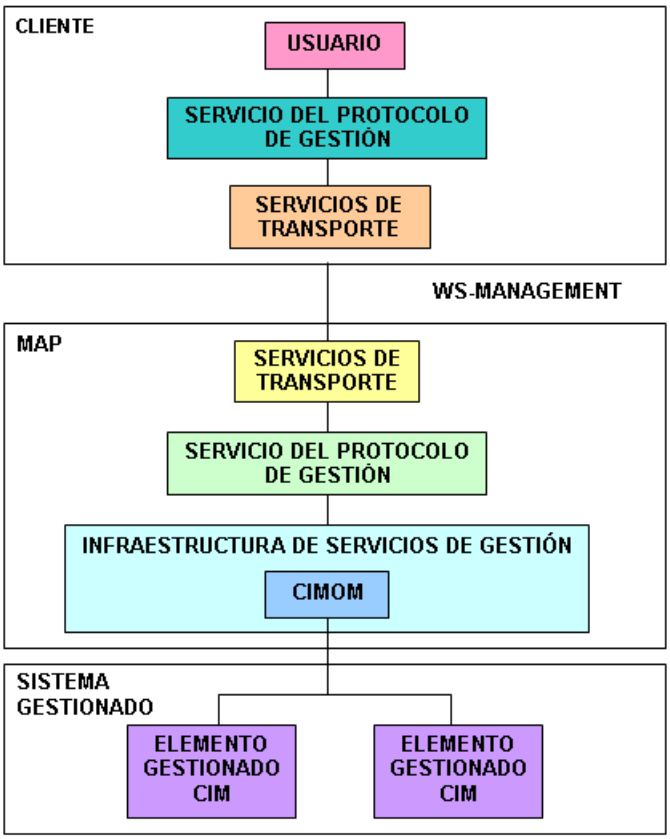


Figura 42. Arquitectura de gestión con WBEM y DASH

Al igual que el CIMOM WBEM, el MAP puede estar en el sistema gestionado o en otro equipo. Por otro lado, teniendo en cuenta que en este caso DASH extiende a WBEM, los modelos de información, organizacional y funcional de esta arquitectura corresponden a los de WBEM pero el modelo de comunicación se basa en la utilización de WS-Management.

4.3. ARQUITECTURA DE GESTIÓN PARA LAS REDES EHAS

Hasta ahora se han establecido posibles arquitecturas de gestión de redes adecuadas para entornos heterogéneos, entonces a continuación se realiza un análisis de estas arquitecturas teniendo en cuenta las características de las redes EHAS y sus posibilidades de implementación, y se define una arquitectura de gestión para las redes EHAS.

En primer lugar, al analizar las arquitecturas que utilizan SNMP y que se mostraron en la sección anterior, en las figuras 37 y 38, de arriba hacia abajo, para el gestor, se puede utilizar cualquier lenguaje de alto nivel y una API SNMP para desarrollar la aplicación, y para el equipo gestionado, se puede utilizar el agente SNMP más evolucionado para

Linux, NetSNMP, el cual soporta una gran cantidad de MIBs y aunque no las MIBs SNMP HPI ni SNMP AIS directamente, soporta el protocolo AgentX.

Por otro lado, existe una implementación del subagente SNMP HPI que expone HPI a través de SNMP, el cual se encuentra en openhpi. Este subagente incluye un soporte completo para recursos, eventos, sensores y controles de hardware, y todos los otros elementos de la especificación HPI, y además, se basa en NetSNMP. A través de este subagente se puede gestionar múltiples tipos de hardware tal como IPMI. En cuanto al subagente SNMP AIS, lastimosamente openais aún no lo soporta, por tanto, en este caso sería necesario extender el agente NetSNMP para AIS.

Finalmente, como se mencionó en el capítulo 3, en EHAS se realizó una placa de interfaz para lograr la comunicación de datos a través de VHF y HF, además de la monitorización de diferentes parámetros, por lo que se decidió desarrollar tanto el hardware como el firmware de la placa con el objetivo de tener un control total de ella.

Teniendo en cuenta esto, al utilizar una solución propia para el hardware, ya no se estaría utilizando IPMI, y si se quisiera utilizar HPI sería necesario realizar todo un desarrollo para lograr la gestión del hardware utilizando este estándar, y una vez se tuviera se podría utilizar el subagente SNMP HPI de openhpi, lo que haría a esta solución sería bastante compleja para lograr la gestión de los sistemas EHAS.

Por otro lado, al analizar las arquitecturas que utilizan WBEM, que se mostraron en la sección anterior, en las figuras 39, 40, 41 y 42, se puede utilizar alguna de sus implementaciones como OpenPegasus, OpenWBEM, SBLIM, WBEM Services and WBEMsource, y sería necesario determinar si la implementación tiene proveedores HPI y AIS, y si no es así, desarrollarlos o utilizar proveedores de este tipo de alguna otra implementación si es posible, una solución que es aún más compleja que la solución basada en SNMP. Además, desde el punto de vista de comunicaciones, WBEM consume mucho más ancho de banda que SNMP, y el ancho de banda en las redes EHAS es un recurso muy limitado.

Como se comentó en el capítulo 2, el sistema operativo de las estaciones VHF y HF, y de los enrutadores es Linux, que como es bien conocido, tiene una característica muy especial que lo hace muy atractivo para los administradores de redes, y es su capacidad de ser manejado mediante línea de comandos incluso en forma remota, por lo que SMASH al utilizar CLP resulta ser una iniciativa bastante interesante para EHAS. Lastimosamente, aún no existen muchas implementaciones de esta iniciativa y la que existe aún está en desarrollo.

Los enrutadores inalámbricos no tienen toda la capacidad de una estación de trabajo, por tanto, se requiere que el sistema de gestión sea liviano y probablemente las implementaciones de DASH lo serán, ya que esta es una iniciativa que está pensada para entornos móviles y de escritorio, pero aún no existen implementaciones debido a que es una iniciativa que se lanzó a comienzos de este año.

Por otro lado, tanto SMASH como DASH permiten la gestión fuera de banda y fuera de servicio, que como se describió en el capítulo 1, se refieren a la gestión que se puede seguir realizando independiente del estado de la máquina y del sistema operativo, y aunque no tienen ninguna relación con la gestión en ausencia de conectividad entre el gestor y el equipo gestionado, son características bastante interesantes para cualquier

solución de gestión. Tanto SMASH como DASH parecen ser dos iniciativas bastantes interesantes para las redes EHAS, por tanto, sería importante en un futuro, analizar las implementaciones que se realicen de estos dos estándares para determinar si se ajustan a las necesidades del proyecto. Tanto SMASH como DASH complementan muy bien a WBEM, por tanto, al igual que este estándar, tampoco son la solución más adecuada para las redes EHAS por las razones que se mencionaron anteriormente.

Finalmente, los estándares basados en servicios Web tampoco son la mejor alternativa para la gestión de las redes EHAS ya que su uso también conlleva a un elevado consumo de ancho de banda.

Después de este análisis se llegó a una importante conclusión que consistió en que en el sistema de gestión de redes EHAS no se pueden utilizar estándares de gestión pero sí la arquitectura de gestión para entornos heterogéneos de la figura 36, que como se observó, es muy adecuada para diferentes escenarios de gestión y se caracteriza por tener varios niveles que permiten independizar funcionalidades, pero debido a que ninguno de los estándares estudiados se puede utilizar y a que tienen características muy interesantes que hacen que puedan coexistir, integrarse y potenciarse unos con otros para lograr un sistema de gestión integral, se realizaron las siguientes abstracciones:

- IPMI define una interfaz de gestionabilidad a un subsistema hardware inteligente que permite monitorear y controlar el hardware del sistema principal. Generalmente se utiliza para monitorear temperatura, niveles de voltajes, etc.

- HPI representa las características del hardware en un modelo abstracto, además permite realizar llamadas a funciones estándar para monitorear y controlar el hardware. Una implementación de HPI representa el hardware en términos del modelo y convierte las llamadas a funciones en acciones apropiadas para el hardware. HPI proporciona servicios a aplicaciones independientemente del hardware.

- AIS define una interfaz a aplicaciones de alta disponibilidad y permite escribir software de aplicación. AIS representa las características de alta disponibilidad del sistema en un modelo abstracto y define APIs para funciones que soportan ese modelo. Una implementación de AIS utiliza ese modelo y traslada funciones de las APIs a acciones apropiadas para el sistema.

Tanto HPI como AIS definen un modelo y una API, y permiten lograr que las aplicaciones sean independientes del hardware y del sistema operativo.

Adicionalmente, la arquitectura de gestión para entornos heterogéneos muestra que la división por niveles o, en otras palabras, la modularidad, debe ser una de las características más importantes de un sistema, en este caso, del sistema de gestión de redes EHAS. Esta característica permite lograr claridad en el diseño, facilitar las adiciones o mejoras, y además, realizar la división de trabajo y por tanto conseguir un menor tiempo de desarrollo, un aspecto bastante importante ya que la Fundación EHAS tiene socios en diferentes países. Entonces, la idea es definir los componentes necesarios y determinar la forma en la que ellos se van a integrar tratando de que mantengan la independencia suficiente uno del otro para no afectar el sistema completo en caso de que alguno de ellos se modifique.

En la Fundación EHAS se decidió desarrollar los siguientes componentes:

- El hardware y firmware de la placa de interfaz que permite la comunicación de datos a través de VHF y HF, y además monitorear parámetros como temperatura, voltaje, etc. de los sistemas gestionados. Este último aspecto es el que más nos interesó para este trabajo.
- Un paquete que realiza una abstracción del hardware, la placa de interfaz, y permite realizar llamadas a funciones para manejarla y obtener información de gestión. En las estaciones este paquete se llama ehas-board y en los enrutadores se llama erouterboard.
- Un paquete que permite realizar el monitoreo y control de estaciones y enrutadores, y que se llama ehas-netman. Este paquete utiliza ehas-board o erouterboard en las estaciones y enrutadores respectivamente.
- Un paquete que permite la configuración y adecuado funcionamiento de los equipos EHAS, tanto estaciones como enrutadores, y que utiliza los paquetes mencionados anteriormente y muchos otros necesarios para este fin. En las estaciones este paquete se llama ehas-station y en los enrutadores se llama ehas-router.

Como se comentó en el capítulo 2, el sistema operativo tanto de estaciones HF y VHF como de enrutadores inalámbricos EHAS es Linux, y por supuesto, lo más deseable es que cumpla con los requerimientos de CGL de OSDL para proporcionar servicios de alta disponibilidad, y en particular, la distribución que se utiliza en EHAS los cumple.

Por otro lado, para el gestor, siguiendo una de las tendencias de la gestión de entornos heterogéneos y la filosofía colaborativa de EHAS, se buscaron herramientas de software libre y de fuente abierta que se caracterizaran por su buen diseño e implementación, especialmente por su modularidad y claridad de código, y que permitieran realizar las adiciones y modificaciones necesarias, e introducir el subsistema basado en correo electrónico para la comunicación entre el gestor y los equipos gestionados, y después de un largo proceso se encontró que la herramienta Zabbix cumple muy bien con estos requisitos y cuenta con el apoyo de un gran número de desarrolladores y usuarios por sus más y mejores características con respecto a otras herramientas de gestión.

Teniendo en cuenta las abstracciones de IPMI, HPI y AIS, mencionados anteriormente, y los componentes del sistema de gestión de redes EHAS se puede realizar la correspondencia que se muestra en la tabla 4.

Tabla 4. Correspondencia entre estándares de gestión y desarrollos EHAS

Estándar de gestión	Componente EHAS
IPMI	Hardware y firmware de la placa de interfaz
HPI	ehas-board/erouterboard
AIS	ehas-netman
Aplicaciones	**Equipo gestionado:** ehas-station/ehas-router **Equipo gestor:** Zabbix

Finalmente, en la figura 43 se muestran los niveles de la arquitectura de gestión para las redes EHAS.

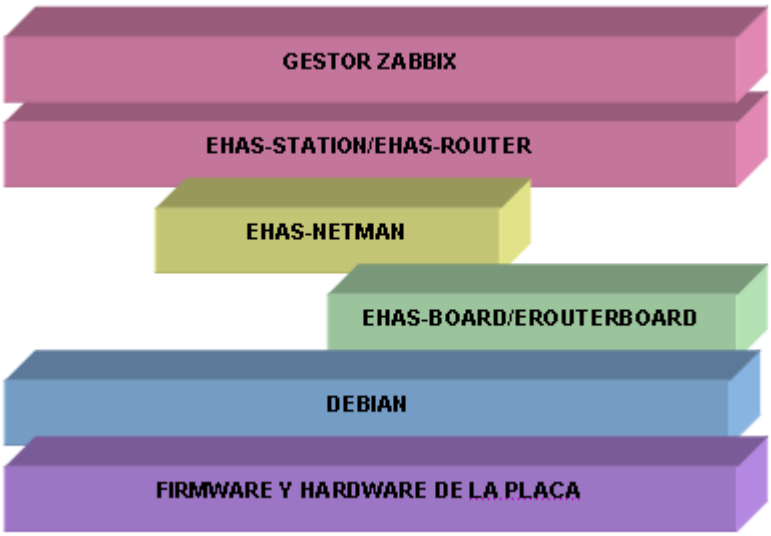


Figura 43. Niveles de la arquitectura de gestión para las redes EHAS

En esta figura 43 se observa que ehas-board/erouterboard, a diferencia de HPI, no se comunica directamente con la placa de interfaz sino solamente con el sistema operativo para manejarla, además, ehas-netman utiliza ehas-board/erouterboard y se comunica con el sistema operativo para realizar una gestión completa de los equipos gestionados, y finalmente, ehas-station/ehas-router utiliza ehas-netman, ehas-board y otros paquetes del sistema para gestionarlo, manejar la placa de interfaz, y además, configurarlo y así lograr su adecuado funcionamiento.

Una vez establecidos los niveles de la arquitectura de gestión para las redes EHAS, fue necesario determinar la forma en la que el gestor y el equipo gestionado se comunican, ya que por las razones mencionadas anteriormente, los estándares de gestión no son la mejor opción para las redes EHAS, y para esto se tuvo en cuenta que las redes EHAS además de ser entornos heterogéneos tienen una característica muy particular, y es que por naturaleza son desconectadas. Las estaciones VHF y HF solamente tienen conectividad durante ciertos instantes de tiempo y los enrutadores inalámbricos también pueden tener períodos de desconexión ya que, como se mencionó en el capítulo 1, se utiliza la tecnología IEEE 802.11 en largas distancias. Entonces, ni SNMP, ni WBEM, ni los otros estándares de gestión estudiados como SMASH y DASH son adecuados para la gestión de este tipo de redes, ya que ellos se basan en la conectividad permanente entre el gestor y el equipo gestionado. Por tanto, fue necesario buscar una solución adecuada para las redes EHAS, por supuesto, diferente a las convencionales debido a las características de las redes EHAS.

En primer lugar, se estableció que en el sistema de gestión de redes EHAS, el gestor no debe depender de la conectividad permanente con los equipos gestionados para poder realizar sus tareas de monitoreo y control, y además, que los equipos gestionados deben recolectar su propia información de gestión y enviarla en el momento en el que tengan conexión. Estos dos aspectos determinaron que la mejor alternativa para lograr la comunicación entre el gestor y los equipos gestionados de las redes EHAS es el correo electrónico, como se muestra en la figura 44.

Figura 44. Componentes de la arquitectura de gestión EHAS

Teniendo en cuenta los niveles de arquitectura de gestión para las redes EHAS que se mostraron en la figura 43 y que la comunicación entre el gestor y el equipo gestionado se realiza a través de correo electrónico como se mostró en la figura 44, se obtuvo la arquitectura para la gestión de las redes EHAS que se muestra en la figura 45.

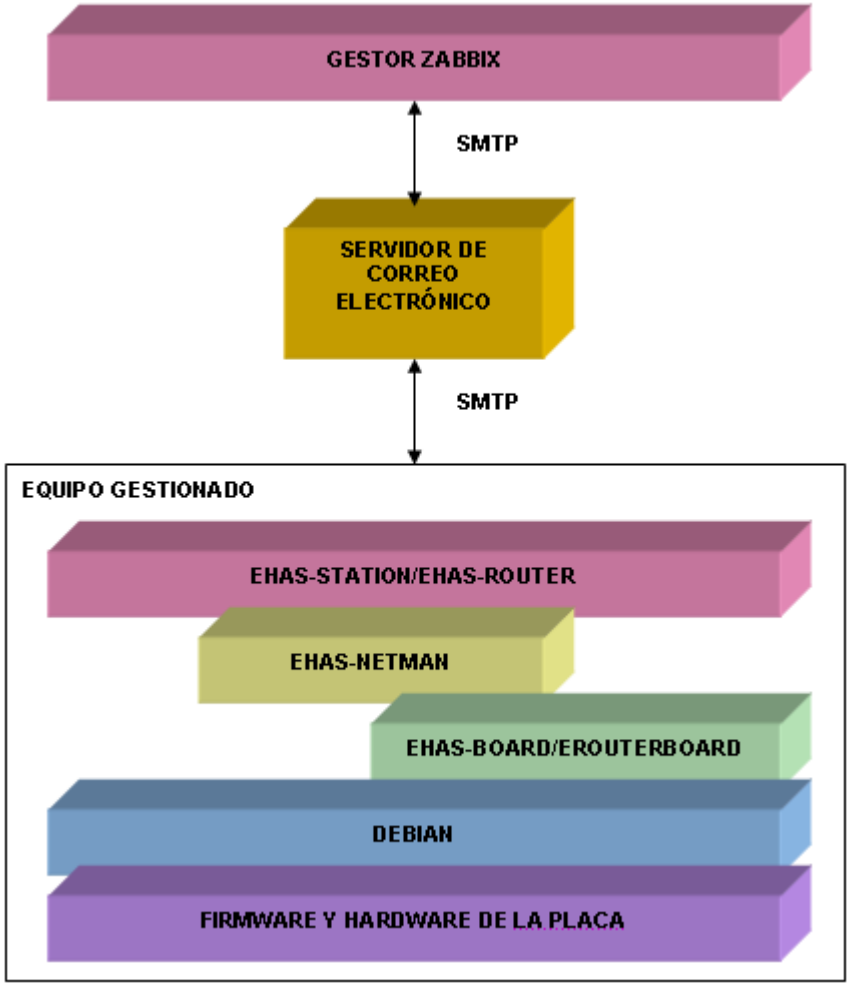


Figura 45. Arquitectura de gestión para las redes EHAS

Adicionalmente, ya que en la arquitectura de gestión para las redes EHAS, al igual que en las arquitecturas de gestión que utilizan SNMP, WBEM, DASH, SMASH que se mostraron en las figuras 37, 38, 39, 40, 41 y 42, se deseó observar un protocolo de comunicación entre el gestor y el equipo gestionado, se realizó un diagrama adicional para la arquitectura de gestión para las redes EHAS que se muestra en la figura 46.

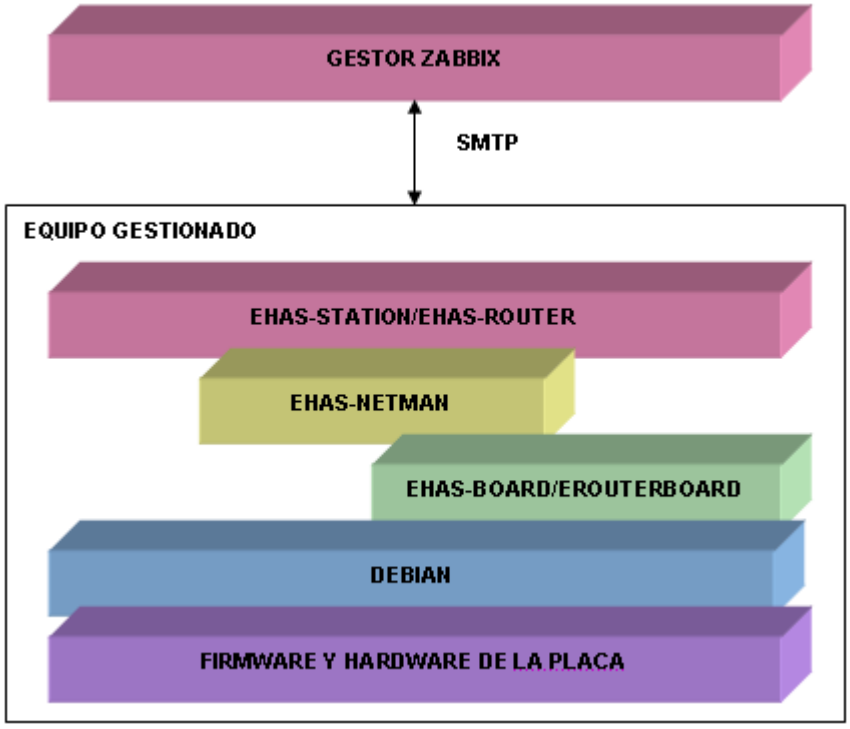


Figura 46. Protocolo de gestión para las redes EHAS

En conclusión, la arquitectura de gestión para las redes EHAS se basó en la arquitectura planteada para entornos heterogéneos que se mostró en la figura 36, pero no se utilizaron estándares de gestión sino desarrollos propios, y herramientas de software libre y de fuente abierta, y además, para la comunicación entre el gestor y los equipos gestionados, es decir, enrutadores inalámbricos, y estaciones VHF y HF, que tienen el sistema operativo Linux, se utilizó el correo electrónico. Realmente se planteó una solución muy diferente a las convencionales ya que se tuvieron en cuenta las características muy particulares de la redes EHAS, pero justamente por esta razón, la solución propuesta es adecuada y se convirtió en todo un reto que implicó un gran trabajo no solo de investigación, para obtener la arquitectura de gestión para entornos heterogéneos y para las redes EHAS, sino también de desarrollo, ya que fue necesario realizar su aplicación a las redes EHAS, es decir, implementar el sistema de gestión de redes EHAS y de esta forma cumplir con todos los objetivos que se plantearon en este proyecto de grado. En el capítulo 5 se describe un poco más en detalle el trabajo de desarrollo que se realizó en este proyecto de grado para implementar el sistema de gestión de redes EHAS.

Como se mencionó en el capítulo 2, en los enrutadores se decidió utilizar Pebble-EHAS, pero posteriormente se comprobó que el sistema operativo de las estaciones también funciona bien en ellos y que los componentes del sistema de gestión para las estaciones también permiten realizar el monitoreo y control de los enrutadores, lo que demuestra aún más que el desarrollo que se realizó es adecuado.

EHAS-Colombia, a través de este trabajo de grado, se hizo responsable del sistema de gestión de redes EHAS, y debido al gran trabajo que implicaba su desarrollo, y a que ehas-station/ehas-router y la placa no solamente están relacionados con el sistema de gestión de redes sino que su principal objetivo es permitir la configuración y lograr el

adecuado funcionamiento de los equipos, y comunicar datos a través de HF y VHF, respectivamente, se vió la necesidad y la gran oportunidad de trabajar en conjunto con EHAS-España y EHAS-Perú. EHAS-España trabajó en ehas-station/ehas-router, y EHAS-Perú trabajó en la placa, mientras que EHAS-Colombia se encargó de ehas-netman, es decir, del componente que realiza las tareas de gestión en los equipos gestionados, en otras palabras, del agente de gestión que por supuesto se debe integrar muy bien con los otros paquetes del sistema y con el sistema, y además, EHAS-Colombia también se encargó de la aplicación de gestión, es decir, del gestor, en este caso, de las adaptaciones y mejoras de la herramienta Zabbix. EHAS-Colombia estuvo a cargo de los equipos gestionados y del gestor ya que tuvo la responsabilidad de determinar y obtener la información de gestión necesaria como la relacionada con las interfaces de red, realizar su almacenamiento y procesamiento, así como también, su despliegue y visualización, además, de realizar todas las pruebas y ajustes necesarios en estaciones VHF y HF, y en enrutadores y, finalmente instalar el sistema de gestión en las redes EHAS. Este trabajo, por tanto, se convirtió en una experiencia de investigación y desarrollo bastante interesante no solo porque se logró encontrar una solución diferente y adecuada para gestionar las redes EHAS sino también porque permitió el intercambio entre los tres países, España, Perú y Colombia. Como se mencionó anteriormente, una descripción un poco más detallada del trabajo de desarrollo que se realizó en EHAS-Colombia a través de este proyecto de grado para implementar el sistema de gestión de redes EHAS se encuentra en el capítulo 5, además, en el anexo A se encuentra el manual de usuario del sistema de gestión de redes EHAS.

A continuación se va a realizar una pequeña descripción de la placa de interfaz y de ehas-station.

4.3.1. Placa de interfaz. Como se mencionó anteriormente, estuvo a cargo de EHAS-Perú. Placa de interfaz entre estaciones y radios VHF o HF [64] que permite la comunicación de datos HF y VHF, además del monitoreo de ciertos parámetros. Para la placa de interfaz se realizó el hardware, el firmware y el software necesarios, y como se mencionó anteriormente, en las estaciones el paquete de la placa se llama ehas-board, y en los enrutadores se llama erouterboard, aunque para estos equipos, aún es un trabajo pendiente. La placa utiliza un puerto USB que es el puerto más utilizado actualmente.

Hardware. Se realizó el circuito impreso que se muestra en la figura 47.

Figura 47. Placa de interfaz EHAS [64]

Firmware. Se utilizó un microcontrolador, exactamente el PIC16C745/JW y principalmente se trabajó con su conversor analógico digital.

Software. Al trabajar con hardware, en primer lugar es necesaria una aplicación que se puede desarrollar en C, C++, Python, etc., y en segundo lugar un control, que es el traductor entre el código de nivel de aplicación y el código de nivel de hardware. En el caso de la placa de interfaz EHAS, la aplicación es ehas-station y el control es ehas-board, el cual contiene eboardc, una aplicación que maneja el puerto USB mediante una librería Linux, y eboardd, un servicio que envía información de la placa cuando la aplicación lo requiere. Realmente este control no es como los controles Windows y Linux los cuales tienen una ubicación específica en el sistema y por este motivo se lo llama pseudocontrol. El paquete ehas-board permite la ejecución de comandos para controlar y obtener información de la placa de interfaz, la cual se probó tanto en Windows como en Linux.

A continuación se encuentran las funciones de monitoreo que realiza la placa de interfaz tanto para radios HF como VHF.
Funciones de monitoreo para radios HF

- Lectura de temperatura
- Lectura de voltaje del sistema

Funciones para la radio VHF

- Lectura de temperatura
- Lectura de voltaje del sistema
- Lectura del ROE

Existen opciones comerciales que se pudieron adquirir en lugar de la placa de interfaz pero se decidió diseñarla y desarrollara para tener el control de todo el sistema, y satisfacer las necesidades particulares de EHAS así como también para soportar futuras adiciones y mejoras.

4.3.2. ehas-station. Como se mencionó anteriormente, estuvo a cargo de EHAS-España. ehas-station [65] funciona como meta-paquete para la instalación conjunta de todo el software necesario para una estación EHAS, es decir, radio y módem telefónico. Pero además de las dependencias, ehas-station incorpora una serie de programas propios con aplicaciones diversas. Una de ellas permite controlar y ajustar la tarjeta de sonido, controlar los radios, y además, config-ehas permite configurar las conexiones telefónicas y por radio, y en general toda la estación.

Además de ehas-station se tiene ehas-router que tiene funciones similares de configuración pero para los enrutadores, aunque como se mencionó anteriormente, también se comprobó que es posible instalar en los enrutadores ehas-station, lo que hace posible mantener una sola aplicación. Las dos aplicaciones difieren en el agente de transferencia de correo, el de ehas-router es más liviano, pero se encontró que es posible instalar el mismo tanto en estaciones como en enrutadores.

Como se mencionó anteriormente, ehas-station utiliza muchos paquetes que son: asterisk-phonepatch, ax25-apps-ehas, ax25proxy, ax25-tools-ehas, ehas-asterisk, ehas-board, ehas-metadistro, ehas-netman, ehas-office-es, ehas-router, ehas-station, ehas-ubuntu, erouterboard, gdmtheme-ehas, grunt-ehas, gserversel, mpartclone, netconf, psk31-module, rrconn_xsrej, rtcpowerup, soundmodem-ehas, soundmodem-log,

soundmodem-newfsk, sshlink, tcptunnel, uuax25, xchat25, xchat31, que son los que permiten configurar una estación y además lograr su adecuado funcionamiento a todo nivel.

5. SISTEMA DE GESTIÓN DE REDES EHAS

En este capítulo se describen diferentes alternativas que se estudiaron para gestionar estaciones HF, VHF y enrutadores inalámbricos, así como también, la herramienta de gestión utilizada, Zabbix, además, el diseño y la implementación del sistema de gestión de redes EHAS, tanto en los equipos gestionados como en el gestor, y finalmente, se muestran las funcionalidades del sistema de gestión, y se describe su instalación y uso en las redes EHAS.

5.1. ALTERNATIVAS PARA EL SISTEMA DE GESTIÓN DE REDES EHAS

En principio se decidió buscar la forma de gestionar los enrutadores inalámbricos y las estaciones HF y VHF [66] en forma independiente, ya que los enrutadores inalámbricos idealmente tienen conectividad permanente mientras que las estaciones no. Para la gestión de enrutadores inalámbricos se pensó en utilizar SNMP porque es uno de los protocolos de gestión de red más implementado, además, porque es muy adecuado para este caso por su baja complejidad y porque sigue la filosofía del mejor esfuerzo. Entonces, se buscó un agente SNMP que fuera libre, que permitiera la gestión de tarjetas inalámbricas y que además funcionara en Pebble, que como se mencionó en el capítulo 3, es el sistema operativo de los enrutadores inalámbricos, y se encontró que una empresa llamada Avantcom [67] proporciona las MIBs 802.11 y 802.11 de Avantcom para el agente Net-SNMP [68], y además proporciona el agente SNMP para Pebble con soporte para estas dos MIBs, muy importantes para la gestión de las tarjetas inalámbricas Prism II y Atheros que son las que utiliza la Fundación EHAS. En seguida, se realizaron pruebas con el agente SNMP para Pebble y se utilizó para conocer, por medio de correos electrónicos, el estado y la disponibilidad diaria de los enrutadores inalámbricos de uno de los enlaces que EHAS-Colombia instaló en Silvia.

Una vez que se comprobó el funcionamiento del agente SNMP para los enrutadores inalámbricos se estudiaron diferentes herramientas de software libre y de fuente abierta, entre ellas MRTG [69], una herramienta bastante conocida en el mundo de la gestión SNMP, Cricket [70], una herramienta que viene en Pebble, y Cacti [71], una herramienta bastante potente y muy utilizada por sus más y mejores características con respecto a otras herramientas de gestión, y en los tres casos, se encontró que era necesario realizar una adaptación de la herramienta para que incorporara el correo electrónico como el mecanismo de comunicación entre el gestor y los equipos gestionados de las redes EHAS, y para que la herramienta permitiera gestionar los enrutadores inalámbricos de la forma más completa posible, lo que incluye la gestión de los clientes de las tarjetas inalámbricas, cuya información se maneja de una forma especial en las MIBs 802.11 y 802.11 de Avantcom, o en otras palabras en las MIBs inalámbricas.

Teniendo en cuenta esto, se estudió la posibilidad de desarrollar una aplicación propia y adecuada a las necesidades de gestión de las redes EHAS, y se decidió utilizar la herramienta en la que se basan las herramientas mencionadas anteriormente y muchas herramientas de gestión, RRDTool [72]. Esta herramienta permite trabajar con bases de datos Round Robin, es decir, bases de datos que no crecen con el tiempo porque tienen un comportamiento circular, es decir, la base de datos tiene un tamaño determinado y cuando se escribe el último dato, el siguiente dato sobrescribe el primero y así sucesivamente. RRDTool es una herramienta que no solo se utiliza en el campo de la

gestión. RRTool permite crear, actualizar, extraer datos, graficar y desplegar datos, etc. de bases de datos Round Robin.

En vista de la potencia de esta herramienta se realizó un prototipo que permitía conocer vía Web, los enrutadores inalámbricos gestionados y sus clientes, además, escoger alguno de ellos, así como también, seleccionar las variables y el período de tiempo, y desplegar el gráfico y los datos de las variables seleccionadas del período de tiempo especificado. En este prototipo se utilizó el correo electrónico para la comunicación entre el gestor y los equipos gestionados, y se desarrollaron diferentes módulos, para lo que se utilizó Perl y PHP, además, Postgres y Apache. En el equipo gestionado se definieron las interfaces a monitorear, las variables de cada interfaz y el tipo de dato de cada una de ellas, además, se realizaron operaciones SNMP como get, walk y translate para obtener los datos deseados tanto de las interfaces como de sus clientes y se tuvo en cuenta el manejo que la información de los clientes tiene en las MIBs inalámbricas. Finalmente, se utilizaron diferentes utilidades de RRDTool para crear, actualizar y extraer datos de las bases de datos Round Robin de cada interfaz y de cada uno de sus clientes, se comprimió la información extraída de las bases de datos y se envió en un correo electrónico a una cuenta especificada.

En el gestor se seleccionaban los correos electrónicos que contenían información de gestión, se obtenía su adjunto, y al igual que en el equipo gestionado, se utilizaron diferentes utilidades de RRDTool para crear bases de datos Round Robin para cada interfaz y cada uno de sus clientes, además, para actualizar cada una de esas bases de datos, graficar las variables y desplegar los datos de las variables en el período de tiempo especificado para un enrutador o un cliente seleccionado por el usuario a través de la interfaz Web. En la figura 48 se muestra una imagen del prototipo realizado:

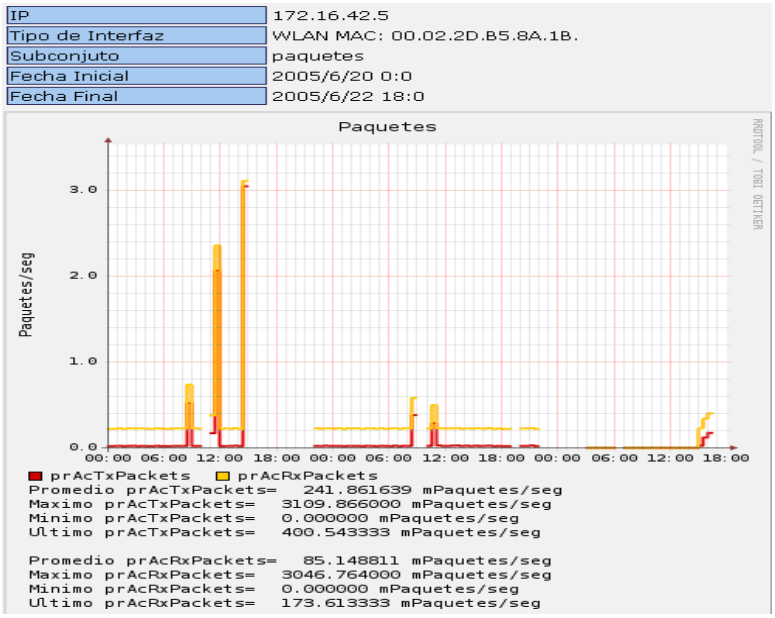


Figura 48. Prototipo con RRDTool y SNMP

Para la gestión de estaciones HF y VHF se decidió utilizar comandos del sistema y logs para obtener la información de gestión necesaria, y además, se compararon herramientas

de gestión como Nagios [73] y Zabbix [74], que son herramientas de monitoreo de redes libres y de fuente abierta, muy potentes y utilizadas.

Al comparar Nagios y Zabbix, se encontró que Nagios es una herramienta más desarrollada que Zabbix, y esto se debe a que Nagios se creó un poco antes que Zabbix, pero Zabbix es una herramienta que está creciendo rápidamente gracias al trabajo constante de su grupo desarrollador, y a sugerencias y contribuciones realizadas por sus usuarios, incluso por usuarios de Nagios que ven en Zabbix una posibilidad muy interesante para monitorear sus redes. Una muestra del rápido avance de Zabbix son las notables diferencias y mejoras entre versiones. Nagios tiene una característica particular, y es que las funcionalidades adicionales se encuentran en módulos independientes, que por un lado, hacen a Nagios una herramienta extensible, pero por otro, implican el manejo de diferentes aplicaciones para conseguir la funcionalidad deseada de esta herramienta. Por el contrario, Zabbix contiene toda su funcionalidad, no requiere de módulos adicionales, y además, tiene una arquitectura bastante modular y flexible que, en el caso en el que fuera necesario, haría posible adicionar casi cualquier funcionalidad adicional requerida.

Teniendo en cuenta esto, se realizó un primer prototipo [75] a partir de la versión 1.0 de Zabbix a la que se le adicionó el soporte para la comunicación a través de correo electrónico entre gestor y equipos gestionados, y además funcionalidades que se consideraron necesarias. En la figura 49 se observa una imagen de Zabbix 1.0 modificado que muestra la carga del procesador de un equipo en una hora.

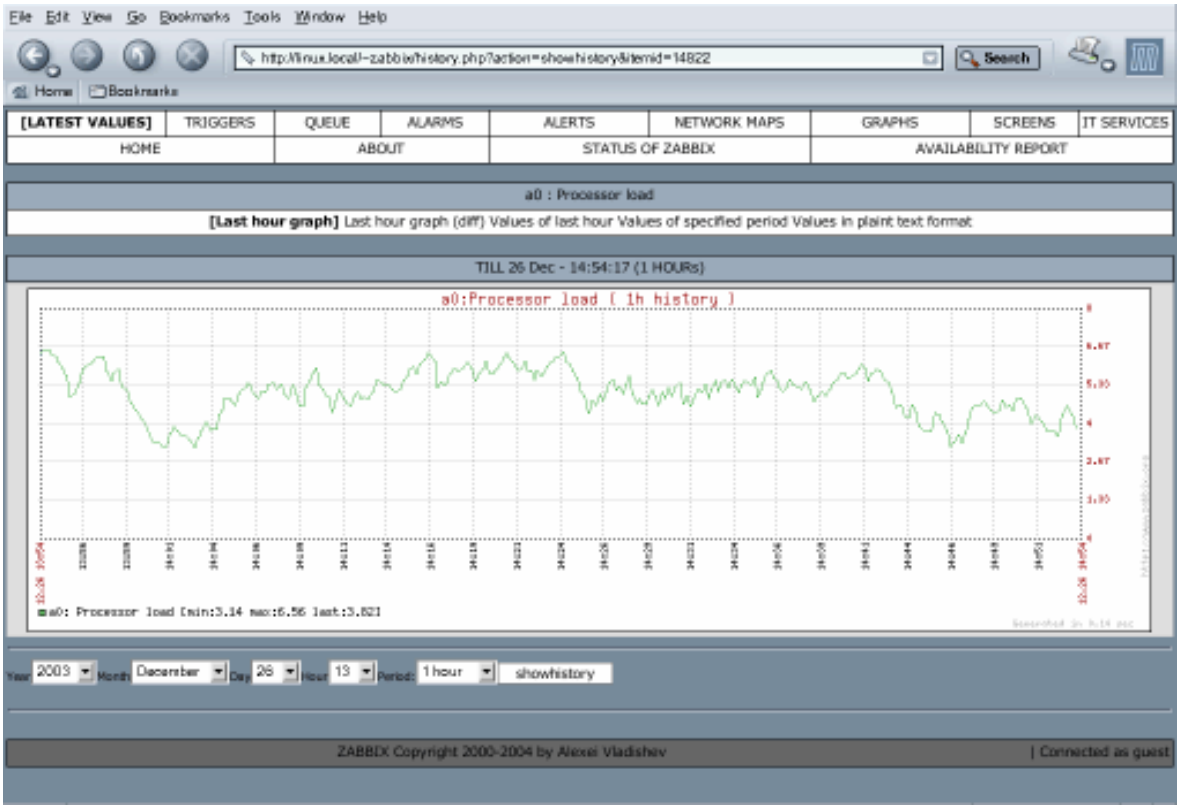


Figura 49. Zabbix 1.0 modificado

Una vez analizadas las posibilidades para la gestión de enrutadores inalámbricos y de estaciones HF y VHF se tomaron varias decisiones. La primera, integrar los componentes de gestión desarrollados para los enrutadores con los componentes de gestión desarrollados para las estaciones HF y VHF, y además mejorarlos, para realizar una gestión más completa tanto de enrutadores inalámbricos como de estaciones HF y VHF, y la segunda, utilizar Zabbix, pero su siguiente versión, es decir, Zabbix 1.1, que tiene grandes cambios con respecto a la versión 1.0 que la hacen mucho mejor y con mayores ventajas con respecto a la versión anterior.

Zabbix 1.1 tiene diferencias conceptuales con respecto a la versión 1.0, pero esas diferencias son las que hacen que esta nueva versión sea mucho más organizada y completa. Debido a esto fue necesario estudiar la nueva versión de Zabbix para poder realizar las adiciones y modificaciones necesarias, y conseguir el sistema de gestión de redes EHAS deseado. A continuación se encuentra una breve descripción de esta herramienta.

5.2. ZABBIX

Zabbix es una herramienta de monitoreo de redes que permite el polling y trapping de datos de los equipos gestionados, es decir, que los datos puedan ser solicitados por el gestor, o puedan ser proporcionados por el equipo gestionado sin petición del gestor. Esta herramienta permite desplegar datos, gráficos, mapas, además, realizar la configuración de la herramienta vía Web y notificar la ocurrencia de eventos predefinidos. Zabbix cuenta con una documentación, con un foro y listas de correo que hacen posible intercambiar experiencias con otros usuarios, solucionar problemas de una manera más rápida, realizar sugerencias, reportar fallos y publicar parches para algún caso en particular.

Zabbix tiene diferentes utilidades pero en este caso solo se utilizó zabbix-server y zabbix-sender. zabbix-server es el que se encarga de toda la funcionalidad de la herramienta, es el que lleva a cabo todas las tareas del gestor, y zabbix-sender es una utilidad de línea de comandos que permite insertar el valor de una variable en la base de datos de una forma más fácil. Esta utilidad es muy importante en este caso porque permite insertar los valores de los datos obtenidos del procesamiento de los logs de gestión que llegan como adjuntos en correos electrónicos al gestor.

Zabbix permite el manejo de usuarios y grupos de usuarios de la herramienta, equipos y grupos de equipos gestionados, variables a monitorear, disparadores de eventos, alertas, alarmas, acciones cuando se producen eventos, datos históricos de las variables monitoreadas, mapas, gráficos, entre otros aspectos. En la figura 50 se observa una imagen de Zabbix 1.1 que muestra los bytes recibidos por la interfaz eth0 del equipo ehase.ca.co.ehas.org en un día.

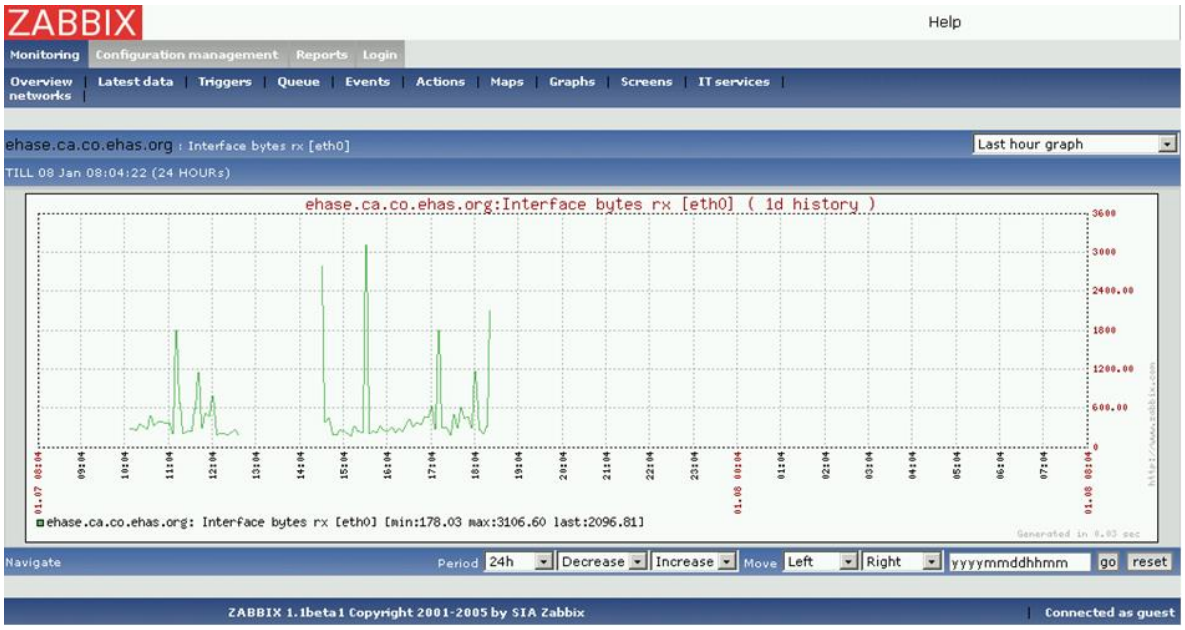


Figura 50. Zabbix 1.1

Además de todas estas características, una de las principales ventajas de Zabbix es que su arquitectura se basa en el paradigma Modelo-Vista-Control, MVC (Model-View-Controller), que permite separar la lógica de control de la visualización y los datos, como se muestra en la figura 51.

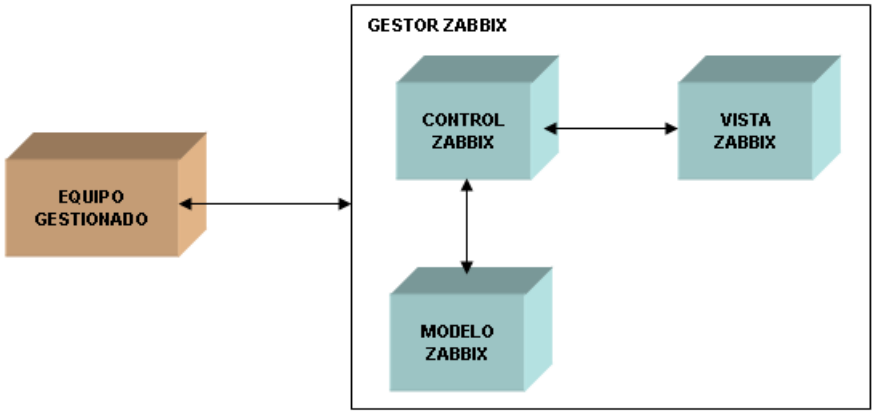


Figura 51. Arquitectura de Zabbix

5.3. DISEÑO DEL SISTEMA DE GESTIÓN DE REDES EHAS

Teniendo en cuenta la arquitectura de Zabbix que se muestra en la figura 51 y la arquitectura de gestión para las redes EHAS que se muestra en la figura 45, los componentes del sistema de gestión de redes EHAS se muestran en la figura 52.

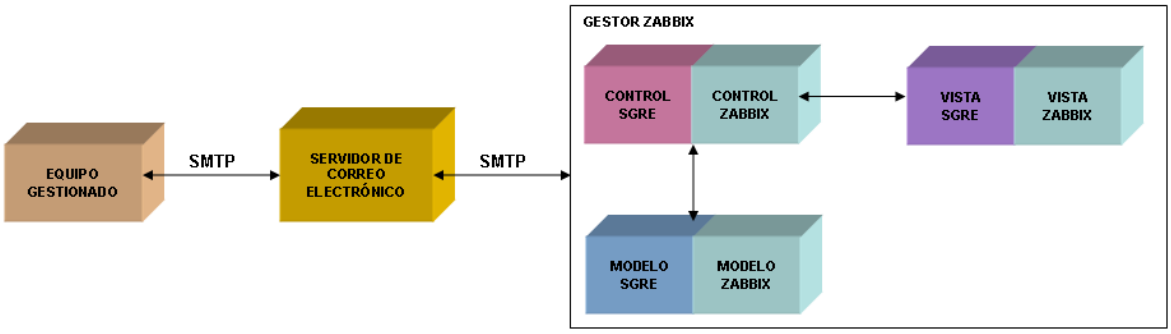


Figura 52. Componentes del sistema de gestión de redes EHAS

En la figura 52 se observan los componentes de la arquitectura de gestión de las redes EHAS: el equipo gestionado, el servidor de correo electrónico y el gestor Zabbix. El equipo gestionado tiene todos los niveles establecidos para este componente en la arquitectura de gestión para las redes EHAS, es decir, el hardware y el firmware de la placa de interfaz, el sistema operativo Debian, ehas-board/erouterboard, ehas-netman y ehas-station/ehas-router. El servidor de correo electrónico permite la comunicación a través de SMTP entre el gestor y el equipo gestionado, y finalmente, en el gestor, la aplicación que se utiliza es Zabbix, que como se observa en la figura 52 no solamente tiene sus componentes modelo, vista y control sino que junto a cada uno de ellos se encuentra un componente modelo, vista y control del sistema de gestión de redes EHAS, SGRE (Sistema de Gestión de Redes EHAS), ya que como se mencionó anteriormente, fue necesario realizar adiciones y mejoras a Zabbix para conseguir el sistema de gestión que permita realizar las tareas de monitoreo y control sobre las redes EHAS.

Como se mencionó en el capítulo 3, en el equipo gestionado, ehas-netman es el que se encarga de realizar las tareas de monitoreo y control, en otras palabras, es el agente de gestión, y contiene los módulos que se muestran en la figura 53.

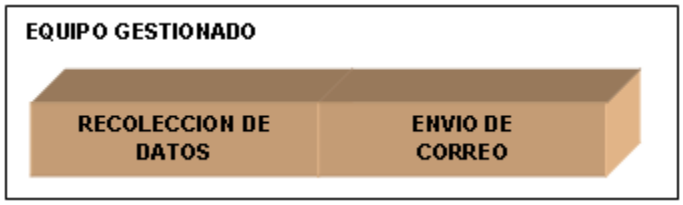


Figura 53. Módulos del equipo gestionado

El módulo de recolección de datos se encarga de obtener toda la información de gestión necesaria en los momentos adecuados, mientras que el módulo de envío de correo se encarga de comprimir la información de gestión y enviarla como adjunto en un correo electrónico cuando corresponda al gestor.

En el gestor, el componente control SGRE se complementa con el componente control Zabbix para mantener toda la lógica de control necesaria del sistema de gestión de redes EHAS. El componente control SGRE contiene los módulos que se muestran en la figura 54.

Figura 54. Módulos del componente control SGRE del gestor

El módulo de recepción de correo se encarga de recibir los correos electrónicos que contienen información de gestión y descomprimir los adjuntos, mientras que el módulo de procesamiento de correo se encarga de determinar si el equipo ya existe en el sistema de gestión y si no es así realiza diversas acciones para adicionar el equipo en forma automática a través del módulo de adición de equipo, y finalmente, el módulo de procesamiento de log se encarga de obtener todos los datos de gestión y de ejecutar las acciones necesarias para almacenarlos en la base de datos en forma adecuada. El componente control SGRE junto con el componente control Zabbix son el punto de unión entre el componente vista SGRE y Zabbix, y el componente modelo SGRE y Zabbix.

En el gestor, el componente modelo SGRE se complementa con el componente modelo Zabbix para manipular los datos de la base de datos del sistema de gestión de redes EHAS. El componente modelo SGRE contiene los módulos que se muestran en la figura 55.

Figura 55. Módulos del componente modelo SGRE del gestor

El módulo de almacenamiento de datos permite adicionar un equipo al sistema de gestión de redes, además, para cada equipo, permite adicionar variables, disparadores de eventos, acción por eventos y gráficos, y además, permite insertar los datos que llegan en los logs de gestión a la base de datos.

En el gestor, el componente vista SGRE se complementa con el componente vista Zabbix para desplegar la información de gestión de una forma agradable para los usuario del sistema de gestión de redes EHAS. El componente vista Zabbix contiene los módulos que se muestran en la figura 56.

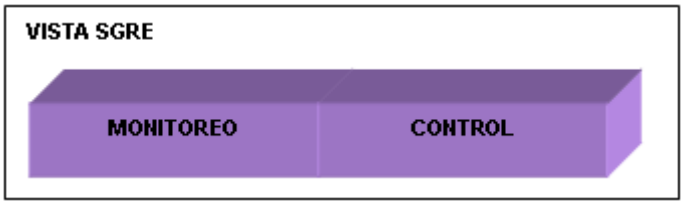


Figura 56. Módulos del componente vista SGRE del gestor

El módulo monitoreo permite desplegar información de gestión adicional a la que despliega Zabbix, mientras que el módulo control permite ejecutar comandos sobre uno o varios equipos gestionados en forma segura.

5.4. IMPLEMENTACIÓN DEL SISTEMA DE GESTIÓN DE REDES EHAS

A continuación se realiza una descripción de la implementación de los módulos responsables de la gestión tanto en el equipo gestionado como en el gestor.

5.4.1. Equipo gestionado. Como se mencionó anteriormente, un equipo gestionado puede ser una estación o un enrutador inalámbrico, y contiene un módulo de recolección de datos y de envío de correo que se encuentran en ehas-netman, pero ya que ehas-netman se integra con ehas-station/ehas-router que son los paquetes que permiten configurar todos los aspectos de un equipo y lograr su adecuado funcionamiento, a continuación se va a explicar en forma global las funcionalidades de gestión implementadas en el equipo gestionado.

Una estación debe tener instalado ehas-station, ehas-netman y grunt-ehas, que permite la ejecución remota segura de comandos, además, Procmail, un agente de entrega de correo, y Postfix, un agente de transferencia de correo. Un enrutador debe tener instalado ehas-router, ehas-netman, grunt-ehas, además erouterboard, Procmail y Masqmail, que al igual que Postfix es un agente de transferencia de correo.

ehas-station contiene a config-ehas, una aplicación que permite configurar una estación a través de una interfaz gráfica. El equivalente de ehas-station para los enrutadores inalámbricos se llama ehas-router y contiene a config-enrutador que permite configurar un enrutador a través de una interfaz gráfica. config-ehas y config enrutador permiten configurar si el envío de logs está activo, el correo electrónico al que se envían los logs, el modelo de la estación o del enrutador, el tiempo en el que se envía el correo que indica que la estación está “alive”, que puede ser cada 2 horas, 8 horas, 2 días, 6 días o 15 días, además, config-ehas y config-enrutador permiten establecer las interfaces de red que se desea monitorear: loopback, ethernet, wi-fi. config-ehas también permite determinar si el monitoreo diario de correo electrónico, placa de interfaz, conexiones de radio, conexiones por módem y telefónicas está activo, mientras que config-enrutador permite determinar si el monitoreo diario de correo electrónico, placa de interfaz, y conexiones telefónicas está activo. En la figura 57 se muestra una imagen de config-ehas y en la figura 58 se muestra una imagen de config-router, en la que se observan las opciones de configuración que brindan.

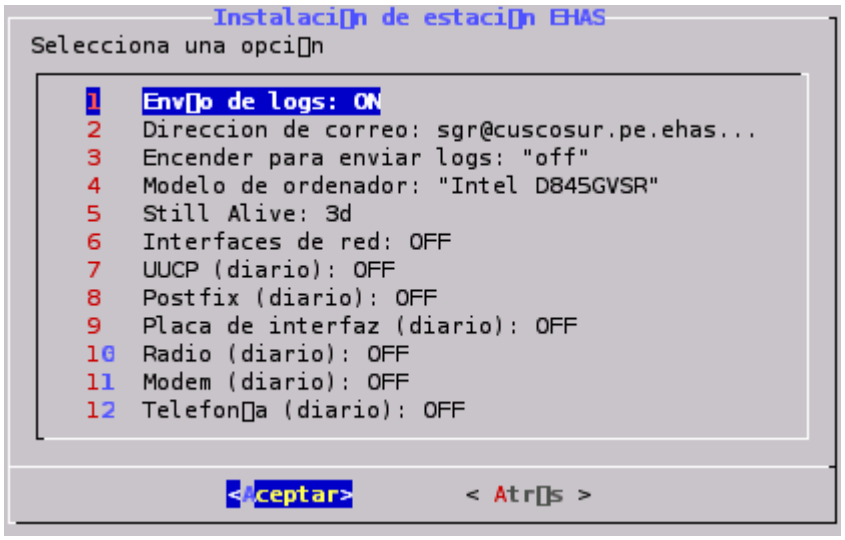


Figura 57. config-ehas

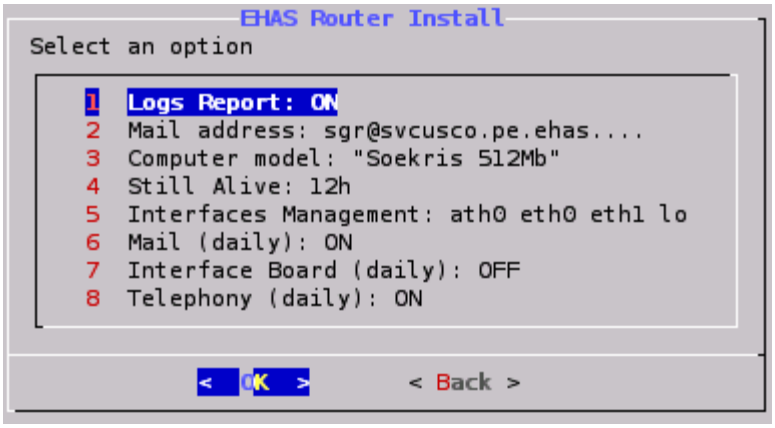


Figura 58. config-router

El equipo gestionado maneja los siguiente tipos de logs: log con información del sistema, log con datos de cada 5 minutos, log de “alive” y log diario. ehas-netman se encarga de obtener la información de cada uno de esos logs cuando corresponde gracias a la utilización del cron que permite la programación de tareas en el sistema. La información de gestión se obtiene por la ejecución de comandos o de logs del sistema, y además, ehas-netman realiza diferentes operaciones sobre los logs, y cuando es necesario, los envía comprimidos como adjunto en un correo electrónico al gestor. A continuación se describe la información de gestión que contiene cada uno de los logs del equipo gestionado.

Log con información del sistema contiene

- Nombre del equipo
- Paquetes instalados
- Tabla de particiones
- Memoria
- Sistema de archivos

- Buses
- Impresoras conectadas al equipo
- Información de configuración del equipo.

Este log se envía cada 15 días.

Log con información de cada 5 minutos

- Temperatura y estado de la CPU. Se utiliza ACPI. Se registra la temperatura y el estado de la CPU
- Tiempo de encendido. Se registran el intervalo de tiempo en el que la estación estuvo encendida.
- Interfaces. Se hará referencia a ellas un poco más adelante en esta sección.

Log de "alive"

- Si el período de "alive" es inferior a 1 día la información de gestión de las interfaces monitoreadas se envía como adjunto en este correo, pero si el período de "alive" es superior a 1 día esa información se envía junto con la información de gestión diaria.

Log diario

- Información de configuración de gestión.
- Estado de discos duros. Se utiliza smartools que proporciona información de los discos duros, permite saber si hubo algún problema o si va a haber alguno en las próximas horas, etc.
- "Crashes". Se determina las reinciaciones no adecuadas del sistema.
- Espacio total, usado y disponible en bytes e inodos.
- Estadísiticas de la CPU. Se analizan los datos registrados en el log que contiene información de cada 5 minutos y se determina la temperatura mínima y su timestamp, la temperatura máxima y su timestamp, y se obtiene el promedio de las temperaturas, los timestamps inicial y final de los datos registrados, y a partir de la temperatura promedio se determina el estado de la CPU para lo que se tiene en cuenta los estados especificados por ACPI según la temperatura.
- Estadísticas de encendido. Se leen los datos almacenados en el log que contiene información de cada 5 minutos y se determinan los intervalos de tiempo en los que el computador estuvo encendido.
- Correo electrónico, placa de interfaz y conexiones por módem. Se obtiene la información registrada en el sistema, y para correo electrónico se determina el número de correos entrantes y salientes, así como también, las cuentas de correo con el número de correos que han enviado o recibido, para la placa de interfaz se obtiene el nivel de batería, y para cada una de las conexiones de radio, el SWR y temperatura de cada una de ellas.
- Telefonía. Se utiliza un log de Asterisk que es la herramienta que permite integrar la PSTN e Internet, y se determina los tipos de llamadas, extensiones desde las que se han realizado llamadas y de las cuales se ha recibido llamadas con el número de llamadas, entre otros datos.
- Cola de correo. Se determina el número de mensajes, el tamaño en bytes de esos mensajes y el tiempo más largo que un mensaje ha estado en la cola de correo.
- Cola de correo UUCP. Se determina el número mensajes, el tamaño en bytes de esos mensajes y el tiempo más largo que un mensaje ha estado en la cola de correo de las estaciones HF y VHF que utilizan UUCP (Unix to Unix CoPy) para el envío y la recepción de correo electrónico.

- Conexiones por radio. Se utiliza la información registrada en el sistema para cada conexión tanto HF como VHF, y se calcula el tiempo de uso del proxy, además, se determina la BER HF y VHF, y el timestamp inicial y final de los datos medidos, así como también, la velocidad VHF de transmisión y de recepción teniendo en cuenta que la velocidad VHF solo puede tener ciertos valores, la velocidad HF de transmisión y de recepción así como también la SNR, tanto máximas como mínimas, y los timestamps en los que se producen esos valores, además, se calculan los promedios de cada uno de esos datos y se determina los timestamps inicial y final en los que se realizaron las medidas.
- Impresoras. Se obtiene el número de trabajos, el tamaño en bytes de esos trabajos y el tiempo más largo que un trabajo ha estado en la cola de impresión.
- Estadísticas de UUCP. Se determina los la velocidad en bits/seg, los bytes y los segundos que han tomado las comunicaciones por UUCP.
- Estadísticas de uucall. Se determina los segundos que han demorado las autenticaciones, así como también, la velocidad en bits/seg, los bytes y los segundos que han tomado las transferencias que utilizan uucall, una utilidad desarrollada por la Fundación EHAS para las comunicaciones HF y VHF.
- Usuarios. Se determina el número de cuentas de usuario que existen en el sistema.

Para la monitorización de las interfaces, en un principio se pensó en utilizar SNMP como se comentó anteriormente, pero se encontró que el agente SNMP de Avantcom depende de la versión del driver de la tarjeta para obtener toda la información de gestión de las MIBs inalámbricas, entonces se estudiaron algunas herramientas que permiten el monitoreo de interfaces, e incluso, se analizó la forma en la que trabaja el agente SNMP, y finalmente se determinó los comandos y archivos del sistema que permiten obtener la información de gestión deseada para el sistema de gestión de redes EHAS.

Teniendo en cuenta esto, se realizó una implementación para monitorear las interfaces que contiene las variables a monitorear de las interfaces de una estación o enrutador inalámbrico, el tipo de dato de cada variable, que fue necesario debido a que se identificaron dos tipos de datos, el primero, para los datos que se utilizan tal como son leídos y el segundo, para los datos que se obtienen como la diferencia entre el valor actual y el valor inmediatamente anterior dividido por la diferencia entre los tiempos en que se realizaron las lecturas, y además, esta implementación contiene información para poder realizar el filtrado de la información de gestión y obtener los datos de gestión que se necesitan. En este caso, se utilizaron comandos y archivos del sistema para obtener información de las interfaces, como por ejemplo, para todas las interfaces: dirección IP, dirección MAC, paquetes, bytes y errores de transmisión y recepción, etc., para las interfaces inalámbricas: punto de acceso, frecuencia, tasa de bits, señal, ruido, y potencia de transmisión, etc., para las tarjetas Prism: descartes, errores de FCS en recepción, y tramas y octetos, unicast y multicast de transmisión y recepción, etc., para las tarjetas Atheros: errores a nivel físico, tramas de transmisión descartadas, tramas transmitidas sin ACK, fallas de CRC en recepción, reintentos fallidos en transmisión, etc., para los clientes de las tarjetas Prism: bytes transmisión y recepción, señal, ruido, tasa de bits, paquetes a 1, 2, 5 y 11 Mbps de transmisión y recepción, etc., y para los clientes de las tarjetas Atheros: señal, ruido, entre otros.

En cuanto al control de estaciones HF y VHF, así como también de enrutadores inalámbricos, se utilizó grunt, una herramienta que permite la ejecución remota de comandos y en forma segura, gracias a que trabaja junto con GPG que permite firmar digitalmente la información. grunt contiene gruntrun y gruntreceive que permiten,

respectivamente, enviar un correo electrónico con un comando, y ejecutar el comando una vez el correo electrónico llega a su destino. El sistema de gestión de redes EHAS ejecuta un comando para solicitar el envío de la información de sistema y poder adicionar el equipo en el sistema de gestión de redes EHAS y también para poder actualizar esta información, y además, permite ejecutar cualquier comando solo si se conoce una contraseña, y en ambos casos, la ejecución del comando se puede realizar sobre uno o un grupo equipos. Debido a esto se crearon dos alias en el equipo gestionado, uno que ejecuta el comando que permite el envío de información del sistema y otro que permite la ejecución de cualquier comando, y además, se generaron los correspondientes pares de claves públicas y privadas.

En el enrutador no se utiliza Postifx sino Masqmail debido a que éste último es más simple, más liviano y por lo tanto más adecuado para los enrutadores inalámbricos, aunque, como se mencionó en el capítulo 3, también se comprobó que Postfix funciona correctamente en los enrutadores. Masqmail al igual que Postfix maneja aliases y trabaja con Procmail.

5.4.2. Gestor. El gestor es una estación que tiene el sistema operativo Linux y Zabbix, la herramienta de gestión seleccionada. El gestor contiene un componente control SGRE que junto con el componente control Zabbix mantienen la lógica del sistema de gestión de redes EHAS y son el punto de unión entre los módulos del componente vista y modelo tanto SGRE como Zabbix para lograr el funcionamiento adecuado del sistema de gestión de redes, entonces, teniendo en cuenta esto, a continuación se va a explicar la implementación de los módulos del componente de control SGRE.

El gestor, además de tener Zabbix, debe tener instalado Procmail y Postfix, y todos los paquetes mencionados para un equipo gestionado, si se desea gestionar el gestor. En el gestor existe un usuario que es a quien se envían los correos electrónicos de gestión.

Recepción de correo. La implementación del módulo de recepción de correo utiliza Procmail, que permite la entrega de correos a sus destinatarios y además analizarlos. En este caso, Procmail determina el nombre y el dominio del equipo, y si el correo contiene un log con información del sistema, un log de “alive” o un log diario que son los logs que envía el equipo gestionado y que se mencionaron anteriormente, además, obtiene el adjunto con munpack, una utilidad que permite desempaquetar mensajes.

Procesamiento de correo. Gracias a la implementación del módulo de procesamiento de correo, cuando llega un log de “alive” o un log diario se verifica si el equipo ya existe en el sistema de gestión, si no existe se le pide al equipo gestionado que envíe el log con información del sistema para lo que se utiliza grunt. Cuando llega un log con información del sistema se determina nuevamente si el equipo ya existe en el sistema de gestión, si no existe se adiciona, esto se explica en detalle más adelante, y además, se obtiene la información del sistema como: el nombre del equipo, paquetes instalados, información de configuración, de particiones, sistema de archivos, memoria, buses e impresoras conectadas al equipo para poder desplegarla a través de la interfaz Web. Cuando el equipo existe en el sistema de gestión se realiza el procesamiento del log de “alive” o del log diario, esto se explica en detalle más adelante. Cuando llega un log con información del sistema y el equipo ya existe en el sistema de gestión, se actualiza su información de sistema.

Adición de equipo. En primer lugar, Zabbix utiliza plantillas. Una plantilla tiene variables, disparadores de eventos, acciones por eventos y gráficos. Zabbix puede tener una plantilla que está enlazada con otra u otras plantillas, esta es una de las características de las nuevas versiones de Zabbix, que significa que esa plantilla contiene todo lo de las otras plantillas y es la plantilla que se puede utilizar para realizar la adición de un equipo. El enlace entre plantillas permite que se pueda adicionar, modificar o eliminar variables, disparadores de eventos, acciones por eventos y gráficos en una plantilla, y que eso se refleje en la plantilla que la contiene y por tanto en los equipos que se han adicionado con base en esa plantilla. En este caso se tiene una sola plantilla tanto para estaciones como para enrutadores, la cual se encuentra enlazada con otra plantilla que contiene todas las variables, disparadores de eventos, acciones por eventos y gráficos para los equipos gestionados.

Para realizar la adición de un equipo sin duplicar la funcionalidad existente en Zabbix se estudió su funcionamiento y se implementó el módulo de adición de equipo del componente control SGRE que determina el país del equipo a gestionar y sus subdominios, que son los grupos a los que adiciona el equipo para lograr un mejor despliegue en la interfaz Web. Este módulo además utiliza las funcionalidades del componente control Zabbix que se tuvo que modificar sin afectar su funcionamiento normal para poder realizar la adición automática de un equipo una vez llega un correo electrónico con un log de gestión. Una vez se determinan los subdominios, se determina si el equipo ya existe en el sistema, y si no existe se inserta en la base de datos, se enlaza con la plantilla adecuada, y se agrega a cada uno de los grupos, es decir, a los subdominios encontrados. En el momento en el que se adiciona un equipo se envía un correo electrónico a los responsables de la gestión de las redes EHAS.

El módulo de adición además de utilizar las funcionalidades del componente control Zabbix modificadas, utiliza las funcionalidades implementadas en el módulo de almacenamiento de datos del componente modelo SGRE, y además las que se encuentran en el componente modelo Zabbix.

Procesamiento de log. La implementación del módulo de procesamiento de log, en primer lugar, si el log es diario, actualiza los disparadores de eventos de “alive” y de nivel de batería de acuerdo a las configuraciones realizadas en el equipo gestionado y que se encuentran en la sección de información de configuración de gestión, y cada vez que llega un log de “alive” o diario actualiza la variable “alive”. Además, tanto para un log diario como para para un log de “alive” cuando contiene información de gestión de las interfaces, analiza cada una de las líneas de los logs para obtener los datos de gestión y poder insertarlos en la base de datos.

Antes de insertar un dato, el módulo de procesamiento de log determina si ese dato ya existe en la base de datos, si no es así, inserta el dato con zabbix-sender. Si zabbix-sender produce un resultado negativo significa que no se insertó el dato y que la variable del equipo no existe en la base de datos, entonces se adiciona la variable, los disparadores de eventos, acciones por eventos y gráficos relacionados con la variable, como se explica en detalle más adelante, y a continuación se ejecuta nuevamente zabbix-sender, y si el resultado es positivo se realizan las actualizaciones necesarias en la base de datos.

Para lograr la adición de variables, disparadores de eventos, acciones por eventos y gráficos de una variable, el módulo de procesamiento de log utiliza funcionalidades del

componente control Zabbix, que también se tuvo que modificar sin afectar su funcionamiento normal, y además utiliza las funcionalidades del módulo de almacenamiento de datos del componente modelo SGRE y además del componente modelo Zabbix.

La adición de una variable se basa en la identificación de un patrón en la variable. Los disparadores de eventos tienen expresiones que contienen la variable, una función que se ejecuta sobre la variable y un valor que se compara con el resultado de ejecución de la función sobre la variable, y dependiendo de la variable, fue necesario lograr el cambio de la función o del valor del disparador del evento. Por otro lado, las acciones a realizar en caso de que se presente un evento consisten en el envío de un correo electrónico a los responsables de la gestión de las redes EHAS, y finalmente, los gráficos muestran la variable junto con aquellas que se consideró importante contrastar.

Monitorización y control. La implementación del módulo de monitorización y control permite la visualización de cuatro pestañas más en la sección de Monitorización de Zabbix. La primera pestaña despliega un árbol con los subdominios y sus equipos que permite la fácil localización de un equipo en la red. La segunda pestaña permite el despliegue de la información del sistema de un equipo, es decir, de los paquetes instalados, el hardware y la configuración, además, permite guardar localmente esa información en formato xls o txt, la tercera pestaña permite el despliegue de un log diario de una fecha seleccionada y también permite guardar localmente esa información en formato xls o txt, y finalmente, la cuarta pestaña, permite ejecutar comandos en forma remota y segura sobre un equipo o un grupo de equipos. A través de esta última pestaña se puede pedir el envío del log que contiene la información del sistema o ejecutar cualquier comando si se conoce la contraseña y además se puede proporcionar una dirección de correo electrónico a la se desea que llegue el resultado de la ejecución del comando. En la figura 59 se muestra una imagen de Zabbix 1.1 modificado que muestra información del hardware como tabla de particiones, memoria y sistema de archivos del equipo svcusco.pe.ehas.org.

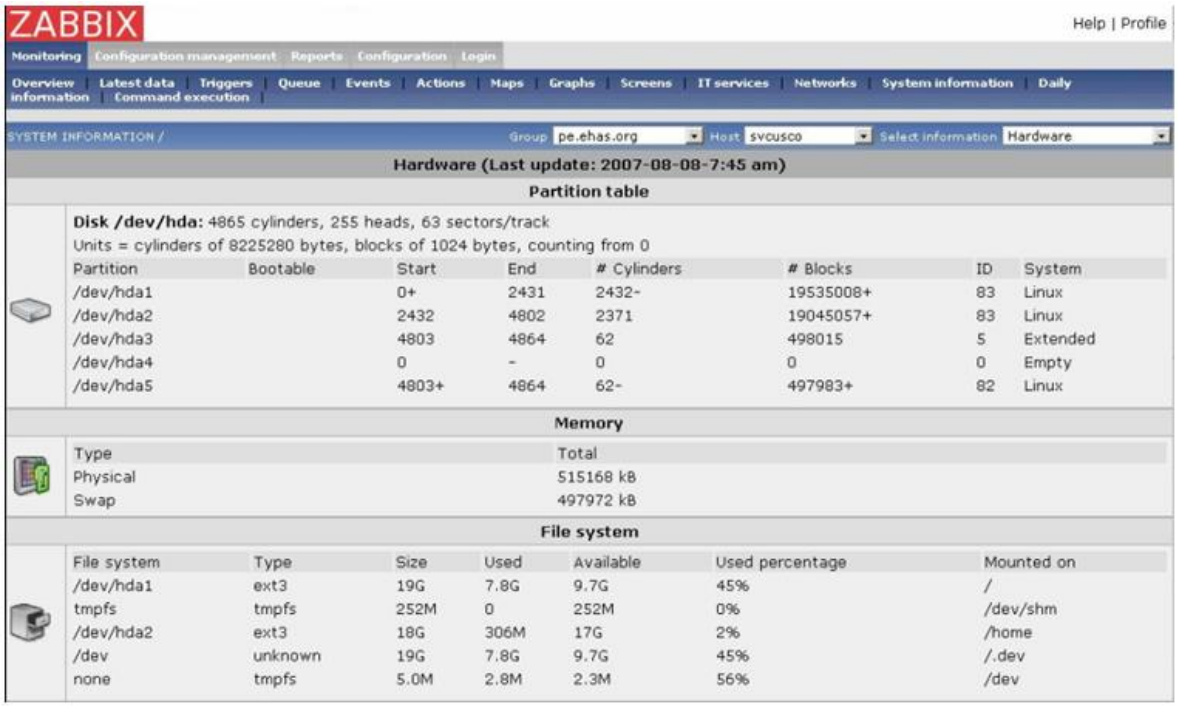


Figura 59. Zabbix 1.1 modificado

Finalmente, una vez se desarrolló y comprobó las funcionalidades de los módulos del equipo gestionado, es decir de ehas-netman, este paquete se incluyó en ehas-station/ehas-router, así mismo, una vez se desarrolló y comprobó las funcionalidades de los componentes modelo, vista y control Zabbix junto con los componentes modelo, vista y control SGRE se decidió realizar dos paquetes Debian llamados zabbix y ehas-zabbix para facilitar el proceso de instalación del gestor. Estos dos paquetes contienen, respectivamente, Zabbix, y las adiciones y mejoras que se realizaron a esta herramienta para realizar las tareas de monitoreo y control sobre las redes EHAS. En el Anexo A se encuentra una descripción detallada del proceso de instalación del sistema de gestión de redes EHAS.

5.5. FUNCIONALIDADES DEL SISTEMA DE GESTIÓN DE REDES EHAS

A continuación se describe en forma general las funcionalidades que ofrece el sistema de gestión de redes EHAS desarrollado. Una explicación más detallada se realiza en el manual de usuario que se encuentra en el Anexo A.

En el sistema de gestión de redes EHAS puede tener varios usuarios con diferentes permisos. Si el usuario no es administrador tiene acceso a la sección de Monitorización, Gestión de Configuración y Reportes, mientras que si es administrador, además, tiene acceso a la sección de Configuración.

La sección de Configuración permite configurar el servidor de correo electrónico que se va a utilizar para enviar alertas, además, adicionar o eliminar las imágenes que se van a utilizar como íconos y como fondos en los mapas. También, permite adicionar, modificar o eliminar usuarios del sistema, y configurar sus parámetros tales como permisos, contraseña, idioma, cuentas de correo a las que se pueden enviar las alertas y períodos de tiempo en los que se pueden enviar, y además, permite adicionar, modificar o eliminar

grupos de usuarios. Por otro lado, esta sección permite adicionar, modificar o eliminar equipos, y configurar sus variables, disparadores de eventos, acciones por eventos y gráficos, pero como se mencionó anteriormente, ya que el sistema de gestión de redes EHAS desarrollado realiza la adición automática de equipos, esta sección se puede utilizar principalmente para cambiar la configuración de los equipos adicionados. La sección de Configuración además de permitir adicionar, modificar o eliminar variables, disparadores de eventos y acciones por eventos, también permite realizar gráficos con variables de diferentes equipos, y configurar colores y tipos de gráficos, además, realizar mapas en los que se pueden ubicar equipos y entre ellos establecer enlaces, los cuales se pueden asociar con un disparador de eventos y según el estado del disparador pueden tener diferentes colores los cuales sirven para saber si hay o no alarmas en una forma visual, y finalmente, esta sección permite adicionar, modificar o eliminar pantallas en las que se puede tener gráficos de una o varias variables, además, tablas de datos y mapas.

Por otro lado, la sección de Monitorización permite observar los últimos datos y disparadores de eventos de un grupo de equipos, así como también, los datos de una variable en cualquier período de tiempo y además, si de tipo numérico, su gráfico. Esta sección permite observar los eventos, es decir, los cambios de estado de un disparador de eventos, y además, las acciones, es decir los correos electrónicos enviados por la activación de un disparador de eventos. Adicionalmente, la sección de Monitorización permite observar los mapas, gráficos y pantallas configurados, así como también, las redes en forma jerárquica, la información de sistema, logs diarios y finalmente, esta sección también permite acceder a la sección de control que permite la ejecución de comandos sobre uno o varios equipos con y sin contraseña.

La sección de Gestión de Configuración permite tener un inventario de los equipos ya que permite observar el tipo, nombre, sistema operativo, número serial, etiqueta, dirección MAC, hardware, software, contacto, localidad y notas específicas del equipo.

Finalmente, la sección de Reportes muestra, entre otras cosas, los estados de los disparadores de eventos tanto en tablas como en gráficos.

5.5.1. Plantillas EHAS

Al instalar los paquetes zabbix y ehas-zabbix se adiciona:
Default group: grupo al que se envían los mensajes de adición de equipos al sistema de gestión de redes EHAS y además los mensajes debido a la ocurrencia de eventos.
Host.EHAS: plantilla que se utiliza para realizar la adición automática de un equipo EHAS.
Host.EHAS.Template: plantilla de Host.EHAS, por tanto, plantilla de todos los equipos que se adicionan al sistema de gestión de redes EHAS con base en Host.EHAS.

Además se adiciona un enlace entre Host.EHAS y Host.EHAS.Template.

Host.EHAS.Template contiene todas las variables, disparadores de eventos, acciones por eventos y gráficos necesarios para gestionar estaciones Linux, estaciones Linux HF y VHF y enrutadores inalámbricos.

Las variables que se encuentran en la plantilla Host.EHAS.Template se muestran en la tabla 5.

Tabla 5. Variables de la plantilla EHAS

Variable	Descripción	Significado	Muestreo	Log de envío
Alive	Alive	Se inserta un 1 cada vez que se procesa un log.	Se actualiza con la llegada de cada log	El disparador de eventos depende del log stilalive
board_battery	Borrad battery (volts)	Nivel de batería. Este dato está en voltios.	Cada 24 horas	Log diario
board_swr[radio]	Board SWR [radio]	SWR (Stationary Wave Rate), relación de onda estacionaria.	Cada 24 horas	Log diario
board_temp[radio]	Board temp [radio] (C degrees)	Temperatura de la placa. Este dato está en grados centígrados.	Cada 24 horas	Log diario
Cpu_temp_avg	CPU temp avg (C degrees)	Temperatura promedio de la CPU. Este dato está en grados centígrados.	Cada 24 horas	Log diario
Cpu_temp_max	CPU temp max (C degrees)	Temperatura máxima de la CPU. Este dato está en grados centígrados.	Cada 24 horas	Log diario
Cpu_temp_min	CPU temp min (C degrees)	Temperatura mínima de la CPU. Este dato está en grados centígrados.	Cada 24 horas	Log diario
Cpu_temp_status	CPU temp status	Estado de la CPU. Puede ser OK o No OK.	Cada 24 horas	Log diario
Crashes	Crashes	Reiniciaciones no adecuadas del sistema.	Cada 24 horas	Log diario
Disk_free[df]	Disk free [df] (1024 blocks)	Espacio de disco libre. Este dato está en bloques de 1024 bytes.	Cada 24 horas	Log diario
Disk_free_inodes[df_inodes]	Disk free [df_inodes] (inodes)	Espacio de disco libre. Este dato está en inodos.	Cada 24 horas	Log diario
Disk_total[df]	Disk total [df] (1024 blocks)	Espacio de disco total. Este dato está en bloques de 1024 bytes.	Cada 24 horas	Log diario
Disk_total_inodes[df_inodes]	Disk total [df_inodes] (inodes)	Espacio de disco total. Este dato está en inodos	Cada 24 horas	Log diario
Disk_used[df]	Disk used [df] (percentage)	Espacio de disco usado. Este dato se calcula con base en el espacio de disco usado en bloques de 1024 bytes. Este dato está es un porcentaje.	Cada 24 horas	Log diario
Disk_used_inodes[df_inodes]	Disk used [df_inodes] (percentage)	Espacio de disco usado. Este dato se calcula con base en el espacio de disco usado en inodes. Espacio de disco usado. Este dato es un porcentaje.	Cada 24 horas	Log diario
email_incoming	E-mail incoming	Número de correos	Cada 24	Log diario

		que han llegado al host.	horas	
email_outcoming	E-mail outcoming	Número de correos que han salido del host.	Cada 24 horas	Log diario
email_queue_bytes	E-mail queue bytes	Bytes en la cola de correo de Postfix.	Cada 24 horas	Log diario
email_queue_messages	E-mail queue messages	Mensajes en la cola de correo de Postfix.	Cada 24 horas	Log diario
email_queue_secs_old	E-mail queue secs old	Segundos que el mensaje más antiguo ha estado en la cola de correo de Postfix.	Cada 24 horas	Log diario
email_user_incoming	E-mail user incoming	Cuentas de correo y número de correos que ha recibido cada cuenta, así: cuenta_correo1[número de correos recibos], cuenta_correo2[número de correos recibos].	Cada 24 horas	Log diario
email_user_outcoming	E-mail user outcoming	Cuentas de correo y número de correos que ha enviado cada cuenta, así: dominio de la red[número de correos recibos], otros[número de correos recibos]	Cada 24 horas	Log diario
Hard_disk_status[hd_errors]	Hard disk status [hd_errors]	Estado del disco duro. Puede ser OK y no OK.	Cada 24 horas	Log diario
Iface_address_ip[interface]	Interface address ip [interface]	Dirección IP de una interfaz.	Cada 5 minutos	Log stillalive o diario
Iface_address_mac[interface]	Interface address mac [interface]	Dirección MAC de una interfaz.	Cada 5 minutos	Log stillalive o diario
Iface_bytes_rx[interface]	Interface bytes RX [interface]	Bytes recibidos/seg de una interfaz.	Cada 5 minutos	Log stillalive o diario
Iface_bytes_tx[interface]	Interface bytes TX [interface]	Bytes transmitidos/seg de una interfaz.	Cada 5 minutos	Log stillalive o diario
Iface_errors_rx[interface]	Interface errors RX [interface]	Errores recibidos/seg de una interfaz.	Cada 5 minutos	Log stillalive o diario
Iface_errors_tx[interface]	Interface errors TX [interface]	Errores transmitidos/seg de una interfaz.	Cada 5 minutos	Log stillalive o diario
Iface_packets_rx[interface]	Interface packets RX [interface]	Paquetes recibidos/seg de una interfaz.	Cada 5 minutos	Log stillalive o diario
iface_packets_tx[interface]	Interface packets TX [interface]	Paquetes transmitidos/seg de una interfaz.	Cada 5 minutos	Log stillalive o diario
iface_wlathclient_noise[interface][client]	Interface wireless_ath_client noise [interface][client]	Nivel de ruido de un cliente de una interfaz inalámbrica ath. Este dato está en dBs.	Cada 5 minutos	Log stillalive o diario
iface_wlathclient_quality[interface][client]	Interface wireless_ath_client quality [interface][client]	Calidad de un cliente de una interfaz inalámbrica ath.	Cada 5 minutos	Log stillalive o diario
iface_wlathclient_signal[interface][client]	Interface wireless_ath_client signal [interface][client]	Nivel de señal de un cliente ath. Este dato está en dBs.	Cada 5 minutos	Log stillalive o diario

iface_wlath_errors_physical[interface]	Interface wireless_ath errors physical [interface]	Errores a nivel físico/seg de una interfaz inalámbrica ath.	Cada 5 minutos	Log stillalive o diario
iface_wlath_frames_tx_alternate_rate[interface]	Interface wireless_ath frames TX alternate rate [interface]	Tasa alterna de transmisión de tramas de una interfaz inalámbrica ath.	Cada 5 minutos	Log stillalive o diario
iface_wlath_frames_tx_discarded[interface]	Interface wireless_ath frames TX discarded [interface]	Tramas de transmisión descartadas antes de la asociación/seg de una interfaz inalámbrica ath.	Cada 5 minutos	Log stillalive o diario
iface_wlath_frames_tx_no_ack[interface]	Interface wireless_ath frames TX no ACK [interface]	Tramas de transmisión sin ACK/seg de una interfaz inalámbrica ath.	Cada 5 minutos	Log stillalive o diario
iface_wlath_rssi_last_reception[interface]	Interface wireless_ath RSSI last reception [interface]	Último RSSI (Received Signal Strength Indication) en recepción de una interfaz inalámbrica ath.	Cada 5 minutos	Log stillalive o diario
iface_wlath_rx_discarded_large[interface]	Interface wireless_ath RX discarded large [interface]	Recepciones descartadas por ser demasiado largas/seg.	Cada 5 minutos	Log stillalive o diario
iface_wlath_rx_failed_bad_crc[interface]	Interface wireless_ath RX failed bad CRC [interface]	Recepciones fallidas debido a un CRC (Cyclic Redundancy Code) erróneo/seg.	Cada 5 minutos	Log stillalive o diario
iface_wlath_tx_failed_retries[interface]	Interface wireless_ath TX retries failed [interface]	Reintentos de transmisión fallidos /seg.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_bytes_rx[interface][client]	Interface wireless_wlan_client bytes RX [interface][client]	Bytes recibidos/seg de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_bytes_tx[interface][client]	Interface wireless_wlan_client bytes TX [interface][client]	Bytes transmitidos/seg de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_noise[interface][client]	Interface wireless_wlan_client noise [interface][client]	Nivel de ruido del cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_packets_rx[interface][client]	Interface wireless_wlan_client packets RX [interface][client]	Paquetes recibos/seg de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_packets_rx_11M[interface][client]	Interface wireless_wlan_client packets RX 11M [interface][client]	Paquetes de 11M recibidos/seg de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_packets_rx_1M[interface][client]	Interface wireless_wlan_client packets RX 1M [interface][client]	Paquetes de 1M recibidos/seg de un dliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_packets_rx_2M[interface][client]	Interface wireless_wlan_client packets RX 2M [interface][client]	Paquetes de 2M recibidos/seg de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_packets_rx_5.5M[interface][client]	Interface wireless_wlan_client packets RX	Paquetes de 5.5M recibidos/seg de un cliente de una interfaz	Cada 5 minutos	Log stillalive o diario

	5.5M [interface][client]	wlan.		
iface_wlwlanclient_packets_tx[interface][client]	Interface wireless_wlan_client packets TX [interface][client]	Paquetes transmitidos/seg de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_packets_tx_11M[interface][client]	Interface wireless_wlan_client packets TX 11M [interface][client]	Paquetes de 11M transmitidos/seg de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_packets_tx_1M[interface][client]	Interface wireless_wlan_client packets TX 1M [interface][client]	Paquetes de 1M transmitidos/seg de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_packets_tx_2M[interface][client]	Interface wireless_wlan_client packets TX 2M [interface][client]	Paquetes de 2M transmitidos/seg de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_packets_tx_5.5M[interface][client]	Interface wireless_wlan_client packets TX 5.5M [interface][client]	Paquetes de 5.5M transmitidos/seg de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_quality[interface][client]	Interface wireless_wlan_client quality [interface][client]	Calidad de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_rate_tx[interface][client]	Interface wireless_wlan_client rate TX [interface][client]	Tasa de transmission de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlanclient_signal[interface][client]	Interface wireless_wlan_client signal [interface][client]	Nivel de señal de un cliente de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlan_discards[interface]	Interface wireless_wlan discards [interface]	Descartes/seg de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlan_errors_fcs_rx[interface]	Interface wireless_wlan errors FCS RX [interface]	Errores de FCS (Frame Check Sequence)/seg en recepción.	Cada 5 minutos	Log stillalive o diario
iface_wlwlan_frames_multicast_rx[interface]	Interface wireless_wlan frames multicast RX [interface]	Tramas multicast recibidas/seg de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlan_frames_multicast_tx[interface]	Interface wireless_wlan frames multicast TX [interface]	Tramas multicast transmitidas/seg de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlan_frames_unicast_rx[interface]	Interface wireless_wlan frames unicast RX [interface]	Tramas unicast recibidas/seg de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlan_frames_unicast_tx[interface]	Interface wireless_wlan frames unicast TX [interface]	Tramas unicast transmitidas/seg de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlan_octets_multicast_rx[interface]	Interface wireless_wlan octets multicast RX [interface]	Octetos multicast recibidos/seg de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlan_octets_multicast_tx[interface]	Interface wireless_wlan octets multicast TX [interface]	octetos_multicas transmitidost/seg de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wlwlan_octets_unicast_rx[interface]	Interface wireless_wlan octets unicast RX [interface]	Octetos unicast recibidos/seg de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario

iface_wlwlan_octets_unicast_tx[interface]	Interface wireless_wlan octets unicast TX [interface]	Octetos unicast transmitidos/seg de una interfaz wlan.	Cada 5 minutos	Log stillalive o diario
iface_wl_access_point[interface]	Interface wireless access point [interface]	Punto de acceso de una interfaz inalámbrica ath o wlan.	Cada 5 minutos	Log stillalive o diario
iface_wl_essid[interface]	Interface wireless essid [interface]	ESSID (Extended Service Set ID) de una interfaz inalámbrica ath o wlan.	Cada 5 minutos	Log stillalive o diario
iface_wl_frequency[interface]	Interface wireless frecuency [interface]	Frecuencia de una interfaz inalámbrica ath o wlan.	Cada 5 minutos	Log stillalive o diario
iface_wl_mode[interface]	Interface wireless mode [interface]	Modo trabajo de una interfaz inalámbrica ath o wlan.	Cada 5 minutos	Log stillalive o diario
iface_wl_noise[interface]	Interface wireless noise [interface]	Nivel de ruido de una interfaz inalámbrica ath o wlan.	Cada 5 minutos	Log stillalive o diario
iface_wl_power_tx[interface]	Interface wireless power TX [interface]	Potencia de transmission de una interfaz inalámbrica ath o wlan.	Cada 5 minutos	Log stillalive o diario
iface_wl_quality[interface]	Interface wireless quality [interface]	Calidad de una interfaz inalámbrica ath o wlan.	Cada 5 minutos	Log stillalive o diario
iface_wl_rate_bit[interface]	Interface wireless rate bit [interface]	Tasa de bits de una interfaz inalámbrica ath o wlan.	Cada 5 minutos	Log stillalive o diario
iface_wl_sensitivity[interface]	Interface wireless sensitivity [interface]	Sensibilidad de una interfaz inalámabrica ath o wlan.	Cada 5 minutos	Log stillalive o diario
iface_wl_signal[interface]	Interface wireless signal [interface]	Nivel de señal de una interfaz inalámbrica ath o wlan.	Cada 5 minutos	Log stillalive o diario
power_on	Power on (segs)	Tiempo que el host ha estado encendido.	Cada 24 horas	Log diario
printer_bytes[printer]	Printer bytes [printer]	Bytes en la cola de impresión de una impresora	Cada 24 horas	Log diario
printer_jobs[printer]	Printer jobs [printer]	Trabajos en la cola de impresión de una impresora	Cada 24 horas	Log o diario
printer_secs_old[printer]	Printer secs old [printer]	Segundos que el trabajo más antiguo ha estado en la cola de impresión de una impresora.	Cada 24 horas	Log diario
proxy_time[radio]	Radio Proxy time [radio] (segs)	Tiempo de uso del proxy de una conexión de radio.	Cada 24 horas	Log diario
radio_hf_ber[hf_radio]	Radio HF BER [hf_radio]	BER (Bit Error Rate) de una conexión de radio HF.	Cada 24 horas	Log diario
radio_hf_rate_rx_avg[hf_fradio]	Radio HF rate RX avg [hf_radio] (bps)	Tasa de recepción promedio de una conexión de radio HF. Este dato está en bits/seg.	Cada 24 horas	Log diario
radio_hf_rate_rx_max[hf_radio]	Radio HF rate RX max [hf_radio] (bps)	Tasa de recepción máxima de una conexión de radio HF. Este dato está en bits/seg.	Cada 24 horas	Log diario
radio_hf_rate_rx_min[hf_radio]	Radio HF rate RX min [hf_radio] (bps)	Tasa de recepción mínima de una	Cada 24 horas	Log diario

		conexión de radio HF. Este dato está en bits/seg.		
radio_hf_rate_tx_avg[hf_radio]	Radio HF rate TX avg [hf_radio] (bps)	Tasa de transmisión promedio de una conexión de radio HF. Este dato está en bits/seg.	Cada 24 horas	Log diario
radio_hf_rate_tx_max[hf_radio]	Radio HF rate TX max [hf_radio] (bps)	Tasa de transmisión máxima de una conexión de radio HF. Este dato está en bits/seg.	Cada 24 horas	Log diario
radio_hf_rate_tx_min[hf_radio]	Radio HF rate TX min [hf_radio] (bps)	Tasa de transmisión promedio de una conexión de radio HF. Este dato está en bits/seg.	Cada 24 horas	Log diario
radio_hf_snr_avg[hf_radio]	Radio HF SNR avg [hf_radio]	SNR (Signal Noise Rate), es decir, relación señal a ruido promedio de una conexión de radio HF.	Cada 24 horas	Log diario
radio_hf_snr_max[hf_radio]	Radio HF SNR max [hf_radio]	SNR (Signal Noise Rate), es decir, relación señal a ruido máxima de una conexión de radio HF.	Cada 24 horas	Log diario
radio_hf_snr_min[hf_radio]	Radio HF SNR min [hf_radio]	SNR (Signal Noise Rate), es decir, relación señal a ruido mínima de una conexión de radio HF.	Cada 24 horas	Log diario
radio_uucall_auth_secs[radio]	Radio uucall auth secs [radio]	Segundos de autenticación de uucall de una conexión de radio.	Cada 24 horas	Log diario
radio_uucall_transfer_bps[radio]	Radio uucall transfer bps [radio]	bits/seg transferidos por uucall en una conexión de radio.	Cada 24 horas	Log diario
radio_uucall_transfer_bytes[radio]	Radio uucall transfer bytes [radio]	bytes transferidos por uucall en una conexión de radio.	Cada 24 horas	Log diario
radio_uucall_transfer_secs[radio]	Radio uucall transfer secs [radio]	Segundos de transferencia por ucall de una conexión de radio.	Cada 24 horas	Log diario
radio_vhf_ber[vhf_radio]	Radio VHF BER [vhf_radio]	BER (Bit Error Rate) de una conexión de radio VHF.	Cada 24 horas	Log diario
radio_vhf_rate_rx[vhf_radio]	Radio VHF rate RX [vhf_radio] (bps)	Tasa de recepción de una conexión de radio VHF. Este dato está en bits/seg.	Cada 24 horas	Log diario
radio_vhf_rate_tx[vhf_radio]	Radio VHF rate TX [vhf_radio] (bps)	Tasa de transmisión de una conexión de radio VHF. Este dato está en bits/seg.	Cada 24 horas	Log diario
telephony_call_applications	Telephony call applications	Número de llamadas telefónicas de tipo aplicación.	Cada 24 horas	Log diario
telephony_call_internal	Telephony call internal	Número de llamadas telefónicas de tipo interno.	Cada 24 horas	Log diario
telephony_call_pstn_external	Telephony call pstn external	Número de llamadas telefónicas de tipo externo PSTN.	Cada 24 horas	Log diario
telephony_call_pstn_in	Telephony call pstn in	Número de llamadas telefónicas de tipo	Cada 24 horas	Log diario

		entrante PSTN.		
telephony_call_voip_external	Telephony call voip external	Número de llamadas telefónicas de tipo externo VOIP.	Cada 24 horas	Log diario
telephony_call_voip_in	Telephony call voip in	Número de llamadas telefónicas de tipo entrante VOIP.	Cada 24 horas	Log diario
telephony_command_answer	Telephony command answer	Número de llamadas telefónicas con comando respuesta.	Cada 24 horas	Log diario
telephony_command_background	Telephony command background	Número de llamadas telefónicas con comando fondo.	Cada 24 horas	Log diario
telephony_command_dial	Telephony command dial	Número de llamadas telefónicas con comando dial.	Cada 24 horas	Log diario
telephony_command_hangup	Telephony command hangup	Número de llamadas telefónicas con comando colgar.	Cada 24 horas	Log diario
telephony_command_meetme	Telephony command meetme	Número de llamadas telefónicas de tipo encuentrame.	Cada 24 horas	Log diario
telephony_command_playback	Telephony command playback	Número de llamadas telefónicas con comando grabación.	Cada 24 horas	Log diario
telephony_command_response_timeout	Telephony command response timeout	Número de llamadas telefónicas con comando tiempo de respuesta agotado.	Cada 24 horas	Log diario
telephony_command_ringing	Telephony command ringing	Número de llamadas telefónicas con comando timbrando.	Cada 24 horas	Log diario
telephony_command_voicemail	Telephony command voicemail	Número de llamadas telefónicas con comando correo de voz.	Cada 24 horas	Log diario
telephony_command_waitmusiconhold	Telephony command waitmusiconhold	Número de llamadas telefónicas con comando música de espera.	Cada 24 horas	Log diario
telephony_extensions_called	Telephony extensions called	Extensiones que se han realizado llamadas y las llamadas de cada extensión. Así: extensión[número de llamadas], extensión[número de llamadas].	Cada 24 horas	Log diario
telephony_extensions_calling	Telephony extensions calling	Extensiones que han recibido llamadas y las llamadas de cada extensión. Así: extensión[número de llamadas], extensión2[número de llamadas]	Cada 24 horas	Log diario
telephony_response_answered	Telephony response answered	Número de llamadas telefónicas con respuesta respondido.	Cada 24 horas	Log diario
telephony_response_busy	Telephony response busy	Número de llamadas telefónicas con respuesta ocupado.	Cada 24 horas	Log diario
telephony_response_no_answer	Telephony response no answer	Número de llamadas telefónicas con respuesta no respondido	Cada 24 horas	Log diario
telephony_response_timeout	Telephony response timeout	Número de llamadas telefónicas con	Cada 24 horas	Log diario

		respuesta tiempo agotado.		
users_station	Users station	Usuarios del host. Así: usurio1, usuario2	Cada 24 horas	Log diario
uucp_queue_bytes[uucp_server]	UUCP queue bytes [uucp_server]	Bytes de la cola de correo de UUCP de un servidor UUCP.	Cada 24 horas	Log diario
uucp_queue_emails[uucp_server]	UUCP queue emails[uucp_server]	Correos electrónicos de la cola de correo de UUCP de un servidor UUCP.	Cada 24 horas	Log diario
uucp_queue_secs_old[uucp_server]	UUCP queue secs old [uucp_server]	Segundos que el mensaje más antiguo ha permanecido en la cola de correo de UUCP de un servidor UUCP.	Cada 24 horas	Log diario
uucp_stats_bps[uucp_server]	UUCP stats bps [uucp_server]	bits/seg de de un servidor UUCP.	Cada 24 horas	Log diario
uucp_stats_bytes[uucp_server]	UUCP stats bytes [uucp_server]	Bytes de un servidor UUCP.	Cada 24 horas	Log diario
uucp_stats_secs[uucp_server]	UUCP stats secs [uucp_server]	Segundos de transferencia de un servidor UUCP.	Cada 24 horas	Log diario

Los disparadores de eventos que se encuentran en la plantilla Host.EHAS.Template se muestran en la tabla 6.

Tabla 6. Disparadores de eventos de la plantilla EHAS

Disparador de eventos	Expresión	Severidad
Alive	{Host.EHAS.Template:alive.**nodata(**9999999)}=1	High
board battery (volts)	{Host.EHAS.Template:board_battery.**last(**0)}<0	Warning
board temp [radio] (C degrees)	{Host.EHAS.Template:board_temp[radio].**last(**0)}>80	Average
CPU temp max (C degrees)	{Host.EHAS.Template:cpu_temp_max.**last(**0)}>80	Average
Crashes	{Host.EHAS.Template:crashes.**last(**0)}>3	Warning
Disk free [df] (1024 blocks)	{Host.EHAS.Template:disk_free[df].**last(**0)}<10000	Average
interface wireless signal [interface]	{Host.EHAS.Template:iface_wl_signal[interface].**last(**0)}>95	Average
interface wireless_ath_client signal [interface][client]	{Host.EHAS.Template:iface_wlathclient_signal[interface][client].**last(**0)}>95	Average
interface wireless_wlan_client signal [interface][client]	{Host.EHAS.Template:iface_wlwlanclient_signal[interface][client].**last(**0)}>95	Warning

En la plantilla Host.EHAS.Template, para cada disparador de eventos se tienen dos acciones, una para cuando el disparador de eventos pasa a activo y otra para cuado pasa a inactivo. Por ejemplo, para el disparador de eventos “alive” se tienen las acciones que se muestran en la tabla 7.

Tabla 7. Acciones de la plantilla EHAS para el disparador de eventos “alive”

Alcance	Enviar mensaje a	Cuando el disparador de eventos	Asunto	Repeticiones
Disparador de eventos	Default group	ON	Host is down	No repeats
Disparador de eventos	Default group	OFF	Host is up	No repeats

Los gráficos que se encuentran en la plantilla Host.EHAS.Template se muestran en la tabla 8.

Tabla 8. Gráficos de la plantilla EHAS

Nombre

board temp
interface bytes TX RX
interface packets TX RX
interface wireless signal noise
interface wireless_ath_client signal noise
interface wireless_wlan_client packets RX 1 2 5.5 11 M
interface wireless_wlan_client packets TX 1 2 5.5 11 M
interface wireless_wlan_client signal noise
radio hf rate TX RX avg
radio vhf rate TX RX

5.5.2. Funcionalidades de acuerdo a las FCAPS

En la tabla 9 que muestra las variables de la tabla 5 así como también las funcionalidades del sistema de gestión de redes EHAS organizadas de acuerdo al área de gestión a la que pertenecen.

Tabla 9. Funcionalidades de acuerdo a las FCAPS

Capa	Variables y funcionalidades
Fallas	Alive board_battery board_temp[radio] cpu_temp_max crashes disk_free[df] hard_disk_status[hd_errors] iface_errors_rx[interface] iface_errors_tx[interface] iface_wlathclient_signal[interface][client] face_wlath_errors_physical[interface] face_wlath_frames_tx_discarded[interface] iface_wlath_frames_tx_no_ack[interface] iface_wlath_rx_failed_bad_crc[interface] iface_wlath_tx_failed_retries[interface] iface_wlwlanclient_signal[interface][client] iface_wlwlan_discards[interface] iface_wlwlan_errors_fcs_rx[interface] iface_wl_signal[interface] radio_hf_ber[hf_radio] radio_hf_snr_max[hf_radio] radio_vhf_ber[vhf_radio]
Configuración	Sección de Gestión de Configuración que permite obtener: tipo, nombre, sistema operativo, número serial, etiqueta, dirección MAC, hardware, software, contacto, localidad y notas específicas del equipo.
Contabilidad	No se tiene ninguna variable.
Desempeño	Todas las variables de la tabla 5.
Seguridad	Variables de los clientes inalámbricos ya que pueden indicar la presencia de intrusos. iface_wlathclient_noise[interface][client] iface_wlathclient_quality[interface][client] iface_wlathclient_signal[interface][client] iface_wlwlanclient_bytes_rx[interface][client] iface_wlwlanclient_bytes_tx[interface][client] iface_wlwlanclient_noise[interface][client] iface_wlwlanclient_packets_rx[interface][client] iface_wlwlanclient_packets_rx_11M[interface][client] iface_wlwlanclient_packets_rx_1M[interface][client] iface_wlwlanclient_packets_rx_2M[interface][client] iface_wlwlanclient_packets_rx_5.5M[interface][client] iface_wlwlanclient_packets_tx[interface][client] iface_wlwlanclient_packets_tx_11M[interface][client] iface_wlwlanclient_packets_tx_1M[interface][client] iface_wlwlanclient_packets_tx_2M[interface][client] iface_wlwlanclient_packets_tx_5.5M[interface][client] iface_wlwlanclient_quality[interface][client] iface_wlwlanclient_rate_tx[interface][client]

	iface_wlwlanclient_signal[interface][client]

5.6. PRUEBAS E INSTALACIÓN DEL SISTEMA DE GESTIÓN DE REDES EHAS

Una vez se terminó la implementación del sistema de gestión de redes EHAS se realizaron pruebas con diferentes equipos como, computadores en los que se configuraron conexiones HF y VHF, enrutadores inalámbricos, etc. También, gracias a que en el laboratorio de EHAS Colombia se encontraban los equipos que se iban a instalar en la red EHAS del Pacífico, se hicieron pruebas con Mini-ITXs y Soekris, y en todos los casos se comprobó que el sistema de gestión de redes EHAS desarrollado funcionaba correctamente, ya que los equipos obtenían su propia información de gestión, la enviaban por correo electrónico y el gestor la procesaba, la almacenaba y desplegaba vía Web sin ningún problema.

Después de realizar estas pruebas en el laboratorio se instaló el sistema de gestión de redes en una de las redes EHAS Perú, en la red EHAS Perú Cusco, para comprobar su correcto funcionamiento en un escenario real. Después de instalar el sistema de gestión de redes EHAS y realizar diferentes pruebas se hicieron los ajustes, adiciones y modificaciones necesarias, y finalmente se creó la versión definitiva de los paquetes ehas-netman y ehas-zabbix que se instalaron en la red EHAS Perú Cusco. Adicionalmente, una vez terminó la instalación de la red EHAS Colombia Pacífico se instaló el sistema de gestión de redes EHAS y se dejó listo para su funcionamiento en los enrutadores que se encuentran en la FIET, Guapi, Santa Ana y Timbiquí.

5.6.1. Usos del sistema de gestión de redes en la red EHAS Perú Cusco. En la red EHAS Perú Cusco, el sistema de gestión de redes EHAS desarrollado ha contribuido a detectar y a diagnosticar problemas que afectan el buen funcionamiento de la red, además, ha permitido conocer el uso y las necesidades de mejora de los servicios ofrecidos por EHAS, entonces a continuación se describen los casos más importantes en los que el sistema de gestión de redes ha sido de gran utilidad [56].

1.
En la figura 60 se muestra en color azul el tráfico transmitido y en color rojo el tráfico recibido a lo largo de dos días en la interfaz inalámbrica de Don Juan que se conecta con Josjojahuarina1. De la gráfica se puede concluir que el throughput medio es 50KBytes/seg que es suficiente para satisfacer las necesidades de los usuarios EHAS, además que el tráfico tiene lugar de 8 a.m a 12 p.m aproximadamente y que se presentan picos en ciertas horas.

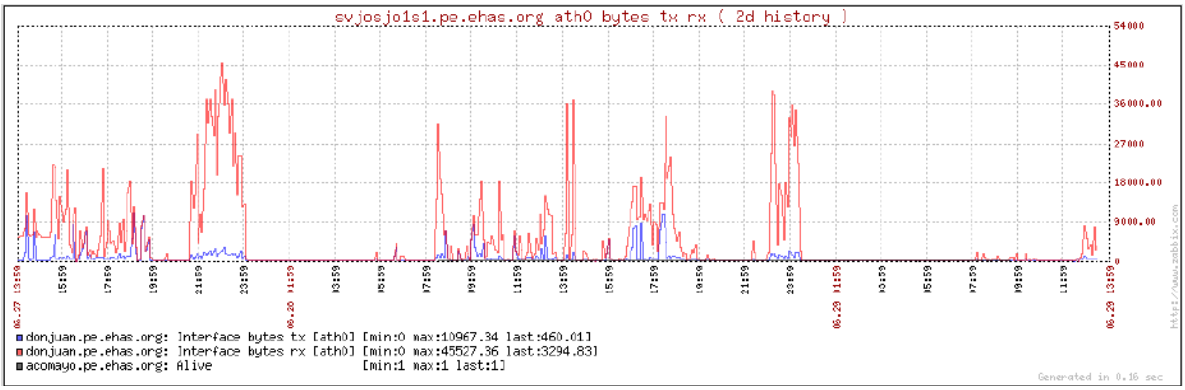


Figura 60. Bytes transmitidos y recibidos

2.
El sistema de gestión envió un correo electrónico informando que Don Juan y por tanto toda la red sur estaba caída, entonces debido a que no se encontró ninguna causa aparente en el sistema de gestión se realizó un viaje a la zona y se encontró que el sistema de alimentación de la soekris se había quemado por un rayo debido a que en la sierra peruana se presentan lluvias bastantes fuertes justamente en esa época del año. Una vez se conoció el problema se inició el proceso de mejora del sistema de seguridad eléctrica y el reemplazo de los equipos dañados para restablecer la red sur, y finalmente, cuando se superó esta situación el sistema de gestión informó que la red del Sur estaba funcionando nuevamente.

3.
El sistema de gestión informaba mediante un correo electrónico que Josjojahuarina1 y por tanto todos los enlaces que dependen de ese enrutador se caían entre las 9 p.m y 10 a.m de todos los días. Normalmente en la sierra peruana el sol sale alrededor de 12 horas, entre las 6 a.m y 6 p.m entonces parecía que el problema estaba relacionado con el sistema de energía solar y efectivamente al visitar el sitio así se comprobó ya que se encontró las baterías descargas lo que hizo evidente un problema de mal dimensionamiento del panel solar. Una vez se detectó el problema se decidió reemplazar las baterías y tomar las medidas necesarias, y desde ese momento los enlaces siguieron funcionando de forma continuada. En la figura 61 se muestra el nivel de señal del enlace en la semana en que se cambiaron las baterías.

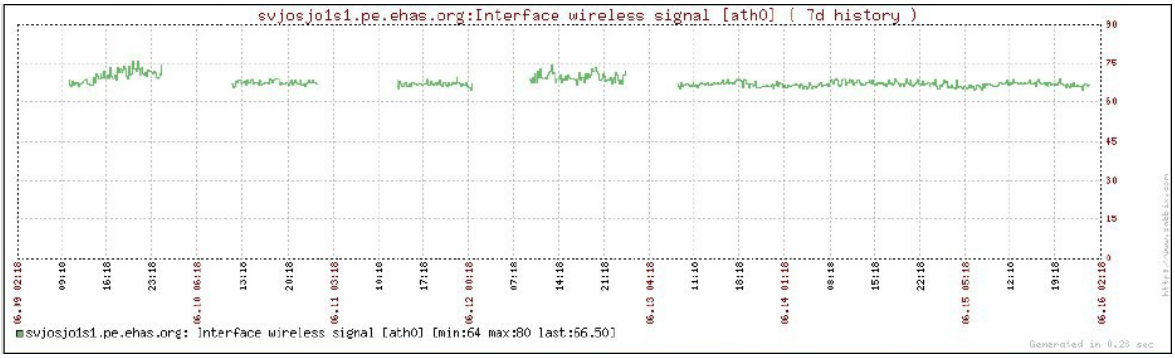
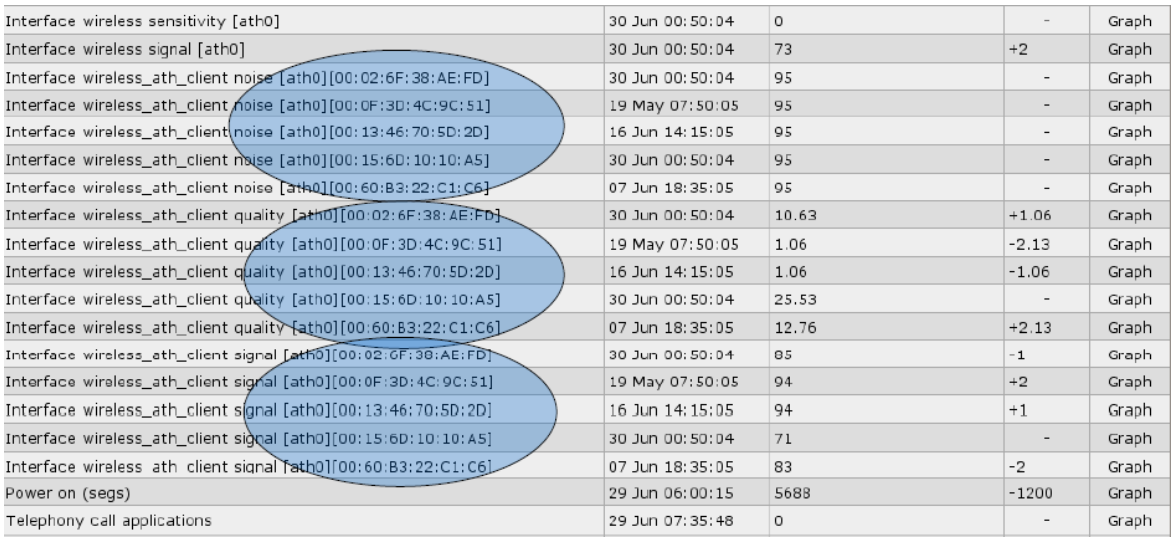


Figura 61. Nivel de señal de una interfaz de red inalámbrica

4.
Algunos usuarios reportaron una mala calidad en las comunicaciones, especialmente de voz, ya que se entrecortaban y tenían muchos retardos, entonces se analizaron los parámetros que proporciona el sistema de gestión y se encontró valores bastante altos en errores en el medio físico y un mal CRC que indican, sobre todo el último parámetro, interferencias en el canal. Una vez detectada una posible causa del problema se realizaron más pruebas y finalmente nuevamente gracias al sistema de gestión de red se comprobó que efectivamente el problema era de interferencias como se muestra en la figura 62.



Interface wireless sensitivity [ath0]	30 Jun 00:50:04	0	-	Graph
Interface wireless signal [ath0]	30 Jun 00:50:04	73	+2	Graph
Interface wireless_ath_client noise [ath0][00:02:6F:38:AE:FD]	30 Jun 00:50:04	95	-	Graph
Interface wireless_ath_client noise [ath0][00:0F:3D:4C:9C:51]	19 May 07:50:05	95	-	Graph
Interface wireless_ath_client noise [ath0][00:13:46:70:5D:2D]	16 Jun 14:15:05	95	-	Graph
Interface wireless_ath_client noise [ath0][00:15:6D:10:10:A5]	30 Jun 00:50:04	95	-	Graph
Interface wireless_ath_client noise [ath0][00:60:B3:22:C1:C6]	07 Jun 18:35:05	95	-	Graph
Interface wireless_ath_client quality [ath0][00:02:6F:38:AE:FD]	30 Jun 00:50:04	10.63	+1.06	Graph
Interface wireless_ath_client quality [ath0][00:0F:3D:4C:9C:51]	19 May 07:50:05	1.06	-2.13	Graph
Interface wireless_ath_client quality [ath0][00:13:46:70:5D:2D]	16 Jun 14:15:05	1.06	-1.06	Graph
Interface wireless_ath_client quality [ath0][00:15:6D:10:10:A5]	30 Jun 00:50:04	25.53	-	Graph
Interface wireless_ath_client quality [ath0][00:60:B3:22:C1:C6]	07 Jun 18:35:05	12.76	+2.13	Graph
Interface wireless_ath_client signal [ath0][00:02:6F:38:AE:FD]	30 Jun 00:50:04	85	-1	Graph
Interface wireless_ath_client signal [ath0][00:0F:3D:4C:9C:51]	19 May 07:50:05	94	+2	Graph
Interface wireless_ath_client signal [ath0][00:13:46:70:5D:2D]	16 Jun 14:15:05	94	+1	Graph
Interface wireless_ath_client signal [ath0][00:15:6D:10:10:A5]	30 Jun 00:50:04	71	-	Graph
Interface wireless_ath_client signal [ath0][00:60:B3:22:C1:C6]	07 Jun 18:35:05	83	-2	Graph
Power on (segs)	29 Jun 06:00:15	5688	-1200	Graph
Telephony call applications	29 Jun 07:35:48	0	-	Graph

Figura 62. Clientes de una interfaz de red inalámbrica

En la figura 62 se observa los parámetros de la interfaz de Josjojahuarina1 que se comunica con Cusco, y sus clientes, de los cuales se observa la dirección MAC. El único cliente que debería tener esta interfaz debería ser Cusco pero se ven otros cuatro, algunos de los cuales aparecen de forma intermitente mientras que otros lo hacen regularmente como también se puede observar en la figura.

Uno de esos clientes corresponde a Jojsojahuarina2 lo que demostró la necesidad de realizar ajustes al diseño de la red EHAS Perú Cusco y además, para evitar otros clientes externos a EHAS la necesidad urgente de implementar medidas de seguridad.

5.
El sistema de gestión de redes informó nuevamente que Don Juan y toda la red del sur estaba caída entonces se observaron todos los parámetros de este enrutador en el sistema y no se encontró ninguna causa aparente del problema, por tanto se decidió realizar un viaje al sitio y se encontró que se habían robado el panel solar, lastimosamente, en ese momento, no se contaba con los recursos para la compra de equipos y la red estuvo caída por bastante tiempo hasta que finalmente se pudo comprar un nuevo panel y se tomaron las medidas de seguridad física necesarias. Una vez se realizó la instalación del nuevo panel, el sistema de gestión de redes informó que la red estaba funcionando nuevamente.

5.6.2. El sistema de gestión de redes en otras redes EHAS. Como se puede observar, el sistema de gestión de redes permitió conocer aspectos de las redes, así como también fallas relacionadas con su diseño, además de problemas originados por factores externos como las condiciones climáticas y el mismo hombre. A las situaciones planteadas se suma que el sistema ayudó a detectar caídas esporádicas de los enutadores, niveles de señal bajos, reinicios frecuentes del servidor de Cusco, entre otros. Finalmente, debido a la importancia del sistema de gestión de redes desarrollado se decidió instalarlo en todas las redes EHAS Perú, lo que confirma que el trabajo desarrollado cumplió todas las expectativas. Las redes de EHAS Perú en las que se instaló el sistema de gestión de redes EHAS desarrollado se muestran en la tabla 10.

Tabla 10. Redes EHAS Perú gestionadas

Red	Departamento	Tecnología
Red Alto Amazonas	Amazonas	Red HF y VHF
Red Pastaza	Amazonas	Red HF y VHF
Red Morona	Amazonas	Red HF y VHF
Red Napo	Loreto	WI-Fi
Red de Iquitos	Loreto	HF y VHF

En estos últimos meses se han adicionado al sistema de gestión de redes EHAS los equipos que se muestran en la tabla 11.

Tabla 11. Equipos de la redes EHAS Perú gestionados

Red	Equipos
Cusco	donjuan.pe.ehas.org, sredcuscosur.pe.ehas.org, svcusco.pe.ehas.org, svhuascar.pe.ehas.org, svjosjo1s1.pe.ehas.org, svjosjo1s2.pe.ehas.org, svlayka.pe.ehas.org
Alto Amazonas	yuri.pe.ehas.org
Pastanza	mkarusha.pe.ehas.org, nprogreso.pe.ehas.org, nyarina.pe.ehas.org, slor.pe.ehas.org, ullpayacu.pe.ehas.org
Morona	caballito.pe.ehas.org, palegria.pe.ehas.org, pamerica.pe.ehas.org, pijuayal.pe.ehas.org, shinguito.pe.ehas.org, sjmorona.pe.ehas.org
Napo	angoterosw1.pe.ehas.org, angoterosw2.pe.ehas.org, camposeriow1.pe.ehas.org, camposeriow2.pe.ehas.org, copalurcow1.pe.ehas.org, copalurcow2.pe.ehas.org, cpantojaw2.pe.ehas.org, rumitumiw1.pe.ehas.org, rumitumiw2.pe.ehas.org, sanrafaelw1.pe.ehas.org, sanrafaelw2.pe.ehas.org, sclotilde.pe.ehas.org, sclotildew1.pe.ehas.org, sclotildew2.pe.ehas.org, tcausanaw1.pe.ehas.org, tcausanaw2.pe.ehas.org, tcurarayw2.pe.ehas.org, tempestadw1.pe.ehas.org, tempestadw2.pe.ehas.org, tupacnapow1.pe.ehas.org, tupcanapow2.pe.ehas.org
Iquitos	buenavista.pe.ehas.org, Iquitos.pe.ehas.org

Esto muestra que la solución de gestión de redes planteada para EHAS realmente es adecuada y como se observa a lo largo de todo este documento es el resultado de un largo camino de investigación y desarrollo que finalmente permitió cumplir con todos los objetivos de este trabajo de grado.

6. CONCLUSIONES

GESTIÓN DE REDES

- SNMP, así como también, los estándares del DMTF y los estándares basados en servicios Web están ganando un gran espacio en el mundo de la gestión por tener más y mejores características con respecto a otros estándares.
- El campo de la gestión está en constante evolución y actualmente existen varios estándares como IPMI, HPI, AIS, WBEM, DASH, SMASH, WS-Management y WSDM que tienen características muy interesantes como la capacidad de integrarse con otros estándares.
- La tendencia de las redes a ser entornos heterogéneos es cada vez mayor, lo que incrementa la importancia, pero al mismo tiempo, la complejidad de la gestión de este tipo de entornos computacionales.
- Los estándares de gestión abiertos y las herramientas de fuente abierta son alternativas muy importantes para la gestión de entornos heterogéneos, ya que se puede lograr su integración, y de esta forma constituir una solución de gestión adecuada que permita gestionar estos entornos computacionales.
- Linux es un sistema operativo que está creciendo días tras día y que ofrece muchas facilidades para realizar su gestión gracias a la capacidad de ejecución de comandos y mantenimiento de logs.

PROCESO DE DESARROLLO

- En una red en la que los equipos no tengan conectividad permanente, que sea heterogénea y con reducido ancho de banda, el correo electrónico es una alternativa bastante interesante para implementar cualquier servicio.
- El software libre es muy importante porque permite ahorrar esfuerzos y tiempo en el desarrollo de aplicaciones, y además cuenta con el respaldo permanente de una gran comunidad.
- La modularidad es uno de los aspectos más importantes en el diseño de una solución, y más aún, para las comunidades investigativas de hoy en día, que gracias a las comunicaciones pueden estar constituidas por varios grupos, incluso interdisciplinarios, ubicados en diferentes partes del mundo.
- La solución adecuada no es la más simple ni la más compleja, no es la que utiliza la última tecnología, ni la que solamente reutiliza las tecnologías existentes, sino la que satisface las necesidades de los usuarios, pero debido a que puede ser muy diferente a las tradicionales por el contexto en el que tiene que funcionar, generalmente exige un gran trabajo de investigación y desarrollo.

PROCESO DE INVESTIGACIÓN

- Los proyectos de cooperación para el desarrollo son una gran oportunidad no solamente para hacer investigación sino también para girar nuestra mirada a todas esas comunidades abandonadas por los gobiernos e incluso por nosotros mismos.
- Las TIC son un instrumento para lograr facilitar el desarrollo de nuestras comunidades, y más aún de las rurales, que generalmente ni siquiera cuentan con las mínimas capacidades de comunicación.
- El sector de la salud es uno de los que más necesita inversión y colaboración por parte de todos para lograr salir adelante, y desde este punto de vista, la telemedicina es una

alternativa bastante interesante para lograr llevar salud a las zonas apartadas de los países latinoamericanos.

7. TRABAJOS FUTUROS

- Incrementar las posibilidades de despliegue de la herramienta para mostrar estadísticas de variables en un período de tiempo determinado, en forma de gráficos de barras y tortas, incluso en los mapas.
- Mejorar el despliegue de mapas que realiza la herramienta, que aunque permite encadenar mapas visualmente, no lo hace lógicamente, lo que impide observar en un mapa de más alta jerarquía el estado general de un mapa de menor jerarquía.
- Establecer un mecanismo para evitar cuellos de botella en el módulo de procesamiento de correo del gestor en caso de que el número de equipos gestionados sea muy elevado.
- Permitir la manipulación de la información que maneja ehas-netman a través de operaciones SNMP, es decir, convertir el agente de gestión ehas-netman en un agente SNMP mediante la implementación de las operaciones de este estándar.
- Estudiar nuevas herramientas de gestión de fuente abierta que permitan realizar la siguiente versión del sistema de gestión de redes con más y mejores características relacionadas con el despliegue de información de gestión.
- Evaluar sistemas de gestión como los de los fabricantes de enrutadores que permitan que la carga de trabajo relacionada con gestión de los equipos sea menor, ya que, especialmente los enrutadores no tienen muchas capacidades de procesamiento.
- Continuar explorando las implementaciones de fuente abierta como: openipmi, openhpi, openais, implementaciones WBEM, omcSMASH y futuras implementaciones de DASH, así como también las implementaciones con servicios Web que prometen ser alternativas bastante interesantes.
- Continuar explorando las iniciativas de integración de los estándares mencionados, sobre todo las que favorecen la utilización de WBEM, uno de los estándares con mayor proyección en el mundo de la gestión.

8. RECOMENDACIONES

- Fomentar el interés por resolver problemas de nuestras comunidades ya que a partir de sus necesidades se puede llegar a tener todo un proyecto de investigación que beneficiaría a todo un grupo y por tanto sería muy gratificante.
- Promover la utilización de software libre que permite aprovechar los esfuerzos de una gran comunidad y además permite proporcionar valor agregado a las soluciones.

9. REFERENCIAS

[1] IPMI-Intelligent Platform Management Interface Specification Second Generation v2.0, Estándar de Intel, 2006, febrero 15 [En línea]. Disponible en: http://www.intel.com/design/servers/ipmi/pdf/IPMIv2_0_rev1_0_E3_markup.pdf.

[2] SourceForge.Net. OpenIPMI [En línea]. Disponible en: http://openipmi.sourceforge.net/.

[3] SourceForge.Net. (2007, abril 26). IPMITool [En línea]. Disponible en: http://ipmitool.sourceforge.net/.

[4] SourceForge.Net. IPMI Management Utilities [En línea]. Disponible en : http://ipmiutil.sourceforge.net/.

[5] Service Availability Forum Hardware Platform Interface, SAI-HPI-B.02.01, Estándar del SAF, 2006, diciembre 13 [En línea]. Disponible en: http://www.saforum.org/specification/getspec_content/SAF-HPI_B.02.01_2006-12-13e.pdf.

[6] SourceForge.Net. (2007, septiembre 28). OpenHPI [En línea]. Disponible en: http://www.openhpi.org/.

[7] Service AvailabilityTM Forum Service Availability Interface, SAI-Overview-B.03.01, Estándar del SAF, 2007 [En línea]. Disponible en: http://www.saforum.org/specification/getspec_content/saiOverview.B0301.pdf.

[8] MontaVista Software, Inc. OpenAIS, Standars-Based Clúster Frameworks [En línea]. Disponible en: http://developer.osdl.org/dev/openais/.

[9] DMTF Tutorial, DMTF, 2006 [En línea]. Disponible en: http://www.wbemsolutions.com/tutorials/DMTF/dmtftutorial.pdf.

[10] DMTF. (2007). CIM and WBEM Tools, and Open Source [En línea]. Disponible en: http://www.dmtf.org/members/tools.

[11] The Open Group. (2007). OpenPegasus, "C++ CIM/WBEM Manageability Services Broker" [En línea]. Disponible en: http://www.openpegasus.org/.

[12] SourceForge.Net. (2007). OpenWBEM [En línea]. Disponible en: http://openwbem.sourceforge.net/.

[13] SourceForge.Net. (2007). Standards Based Linux Instrumentation [En línea]. Disponible en : http://sourceforge.net/projects/sblim/.

[14] SourceForge.Net. (2005, octubre 10). WBEM Services, Java Web Based Enterprise Management [En línea]. Disponible en: http://wbemservices.sourceforge.net/.

[15] DMTF. (2007, agosto 25). System Management Architecture for Server Hardware, White Paper, Version 2.0.0 [En línea]. Disponible en: http://www.dmtf.org/standards/published_documents/DSP2001.pdf.

[16] DMTF. (2006, diciembre). SMASH Simplifies Cross-Platform Server Management, Technical Note [En línea]. Disponible en: http://www.dmtf.org/education/SMASH_Tech_Note-_final_pdf.pdf.

[17] Novell. (2007, febrero 17). omc, open management with CIM [En línea]. Disponible en: http://developer.novell.com/wiki/index.php/OMC-smash.

[18] IBM. SMASH Proxy Installation and User's Guide, Release 1.0 [En línea]. Disponible en: http://publib.boulder.ibm.com/infocenter/toolsctr/v1r0/index.jsp?topic=/com.ibm.smash.doc/smash_t_usingsmash.html.

[19] DMTF. (2007, febrero 2007). Systems Management Architecture for Mobile and Desktop Hardware, White Paper, Version 1.0.0a [En línea]. Disponible en: http://www.dmtf.org/standards/published_documents/DSP2014.pdf.

[20] DMTF. (2007, diciembre). DASH Delivers Multi-Vendor Management for Desktop and Mobile Systems, Technical Note [En línea]. Disponible en: http://www.dmtf.org/education/DASHTechNote.pdf.

[21] DMTF. (2006, abril 5). Web Services for Management (WS-Management), Version: 1.0.0a [En línea]. Disponible en: http://www.dmtf.org/standards/published_documents/DSP0226.pdf.

[22] OASIS. (2006, febrero 24). An Introduction to WSDM. OASIS [En línea]. Disponible en: http://www.oasis-open.org/committees/download.php/16998/wsdm-1.0-intro-primer-cd-01.doc.

[23] Douglas R. Mauro, Kevin J. Schmidt. (2001, julio). Essential SNMP. ISBN: 0-596-00020-0 [En línea]. Disponible en: http://www.unix.org.ua/orelly/networking_2ndEd/snmp/.

[24] Sun Microsystems. (2007). Trail: Java Management Extensions (JMX) [En línea]. Disponible en: http://java.sun.com/docs/books/tutorial/jmx/index.html.

[25] CERN. GridCafé, The place for everybody to learn about Grid [En línea]. Disponible en: http://Gridcafe.Web.cern.ch/Gridcafe/.

[26] I. Foster-Argonne & U.Chicago (Editor), H. Kishimoto-Fujitsu (Editor), A. Savva-Fujitsu (Editor), D. Berry-NeSC, A. Djaoui-CCLRC-RAL, A. Grimshaw-UVa, B. Horn-IBM, F. Maciel-Hitachi, F. Siebenlist-ANL, R. Subramania-Intel, J. Treadwell-HP, J. Von Reich-HP, Open Grid Forum. (2006, julio 24). The Open Grid Services Architecture, Version 1.5, GFD-I.080 [En línea]. Disponible en: http://www.ogf.org/documents/GFD.80.pdf.

[27] gridbus project. Grid Computing Info Centre (GRID Infoware) [En línea]. Disponible en: thttp://www.Gridcomputing.com/.

[28] IBM. IBM Grid computing [En línea]. Disponible en: http://www-03.ibm.com/Grid/.

[29] Frederico Buchholz Maciel-Hitachi, Ltd (Editor), Global Grid Forum. (2005, marzo 1). Resource Management in OGSA, GFD-I.045 [En línea]. Disponible en: http://www.ogf.org/documents/GFD.45.pdf.

[30] openmanagement consortium. (2007, mayo 21). openmanagement consortium [En línea]. Disponible: http://www.openmanagement.org/.

[31] Novell. (2006, noviembre 15). omc, open management with CIM [En línea]. Disponible en: http://developer.novell.com/wiki/index.php/OMC.

[32] Avocent. (2007). What is IPMI? [En línea]. Disponible en: http://www.avocent.com/web/en.nsf/Content/learnmoreIPMI.

[33] Ismael Herrero Rodríguez-Ándago Ingeniería. (2006, junio 2). TOMAS³: Towards and Open Management Architecture for Systems, Software and Services [En línea]. Disponible en: http://www.csi.map.es/csi/tecnimap/tecnimap_2006/04T_PDF/tomas%203.pdf.

[34] CERN. (2007, mayo 21). quattor, system administration toolsuite [En línea]. Disponible en: http://quattor.web.cern.ch/quattor/.

[35] IPMI CIM Mapping Guideline, Versión 0.60, Estándar de Intel, 2006, junio 2 [En línea]. Disponible en: http://download.intel.com/design/servers/ipmi/mapguide.pdf.

[36] DMTF. (2003, enero). The Alert Standard Format, Technical Note [En línea]. Disponible en: http://www.dmtf.org/education/technote_ASF.pdf.

[37] Distributed Systems Management for HPI-SNMP, SAI-HPI-SNMP-B.01.01, Estándar del SAF, 2007 [En línea]. Disponible en: http://www.saforum.org/specification/getspec_content/SAI-HPI-SNMP-B.01.01.pdf.

[38] SAF. (2005, noviembre 10). HPI-B0101-MIB [En línea]. Disponible en: http://www.saforum.org/specification/getspec_content/SAI-HPI-SNMP-MIB-B.01.01.mib.

[39] Distributed Systems Management for AIS-SNMP, SAI-AIS-SNMP-A.01.01, Estándar del SAF, 2007 [En línea]. Disponible en: http://www.saforum.org/specification/getspec_content/SAI-AIS-SNMP-A.01.01.pdf.

[40] SAF. (2005, noviembre 11). SAI-AIS-SNMP-MIB-A.01.01 [En línea]. Disponible en : http://www.saforum.org/specification/getspec_content/SAI-AIS-SNMP-MIB-A.01.01.zip.

[41] DMTF. (2007). DMTF Technologies Diagram [En línea]. Disponible en: http://www.dmtf.org/standards/stackmap/.

[42] DMTF. (2007). System Management BIOS (SMBIOS) Specification [En línea]. Disponible en: http://www.dmtf.org/standards/smbios/.

[43] DMTF. (2007). Alert Standard Format (ASF) Specification [En línea]. Disponible en: http://www.dmtf.org/standards/asf/.

[44] DMTF. (2007). Desktop Management Interface (DMI) Standards [En línea]. Disponible en: http://www.dmtf.org/standards/dmi/.

[45] DMTF. (2007). Common Diagnostic Model (CDM) Initiative [En línea]. Disponible en: http://www.dmtf.org/standards/mgmt/cdm/.

[46] SNIA, Storage Networking Industry Association. (2007). The SNIA Storage Management Initiative, Simplifying storage management with a global technology standard [En línea]. Disponible en: http://www.snia.org/smi/home/.

[47] Sun Microsystems. (2007). SNMP Adapter for WBEM [En línea]. Disponible en: http://docs.sun.com/app/docs/doc/806-6827/6jfoa8m7d?a=view.

[48] Sun Microsystems. (2007). SNMP Provider [En línea]. Disponible en: http://docs.sun.com/app/docs/doc/806-6827/6jfoa8m7j?a=view.

[49] SAF, DMTF. (2004, agosto 7). SAF / DMTF Work Register [En línea]. Disponible en: http://www.dmtf.org/about/register/SAF-DMTFWorkRegister.pdf.

[50] Michael E. Brasher, Karl Schopmeyer. (2006, marzo 20). CIMPLE: An Embeddable CIM Provider Engine [En línea]. http://cimple.org/whitepaper.pdf.

[51] The Open Group. (2003, septiembre 17). Systems Management: Common Manageability Programming Interface (CMPI), Version 1.3 [En línea]. Disponible en: http://www.openpegasus.org/uploads/40/4031/CMPI_Specification_13.pdf.

[52] Karl Schopmeyer, Mike Brasher. Architectural Overview [En línea]. Disponible en: http://cimple.org/overview.html.

[53] SourceForge. (2006, abril 21). Abstraction Layer [En línea]. Disponible en: http://www.mirrorservice.org/sites/download.sourceforge.net/pub/sourceforge/c/ci/cimom-abstract/AbstractionLayerHLD.doc.

[54] Arnau Sánchez-Fundación EHAS, Joaquín Soane-Universidad Politécnica de Madrid. (2006, febrero). Software libre para transmisión digital en enlaces radio. ISSN 1818-728X 9771818728000 [En línea]. Disponible en: http://interno.ehas.org/intranet/comunicacion-1/comunicacion/articulos/forotelemed2005/memorias_i_forotelmed.pdf.

[55] Javier Simó-Fundación EHAS, Pablo Osuna-Fundación Rafael Escolá, Joaquín Soane-Universidad Politécnica de Madrid, Andrés Martínez-Universidad Rey Juan Carlos. (2006, febrero). Router solar autoconfigurable para redes Mesh IEEE 802.11 de telemedicina rural en América Latina. ISSN 1818-728X 9771818728000 [En línea]. Disponible en: http://interno.ehas.org/intranet/comunicacion-1/comunicacion/articulos/forotelemed2005/memorias_i_forotelmed.pdf.

[56] Pablo Osuna-Universidad Politécnica de Madrid. (2006, julio 17). Redes “MESH” Wi-Fi de bajo coste, desarrollo de un router inalámbrico solar autónomo, proyecto de fin de carrera [En línea]. Disponible en: http://interno.ehas.org/prp/routersolar/descargas/pfc_pabloosuna.pdf.

[57] Álvaro Rendón Gallón-Grupo de Ingeniería Telemática, María Fernanda Dulcey Morán-Grupo de Ingeniería Telemática, Eva Juliana Maya Ortiz-Grupo de Ingeniería Telemática. (2006, octubre 25). EHAS: Una alternativa de conectividad para el Sistema de Salud Pública del Cauca. ISBN 978-958-9451-16-8 [En línea]. Disponible en: http://www.unicauca.edu.co/grupomantis/ebookweb.swf.

[58] David Espinoza Aguilar-Universidad Católica del Perú, River Quispe Tacas-Universidad Católica del Perú. (2006, febrero). Diseño de una red WiFi y telefonía IP para centros y puestos de salud rurales en Quispicanchis y Acomayo. ISSN 1818-728X 9771818728000 [En línea]. Disponible en: http://interno.ehas.org/intranet/comunicacion-1/comunicacion/articulos/forotelemed2005/memorias_i_forotelmed.pdf.

[59] Carl Eklund-Nokia Research Center, Roger B. Marks-National Institute of Standards and Technology, Kenneth L. Stanwood- Ensemble Communications Inc., Stanley Wang-Ensemble Communications Inc. (2002, junio 4). IEEE Standard 802.16: A Technical Overview of the WirelessMAN Air Interfac e for Broadband Wireless Access. IEEE C802.16-02/05 [En línea]. Disponible en: http://www.ieee802.org/16/docs/02/C80216-02_05.pdf.

[60] IEEE. (2006, octubre 16). Management Information Base Extensions for Local and Metropolitan Area Networks [En línea]. Disponible en: http://www.ieee802.org/16/netman/docs/80216i-06_001r4.zip.

[61] Forrest Warthman-Warthman Associates. (2003, mayo 3). Delay Tolerant Networks (DTNs), Version 1.1 [En línea]. Disponible en: http://www.dtnrg.org/docs/tutorials/warthman-1.1.pdf.

[62] V. Cerf-Worldcom/Jet Propulsion Laboratory, S. Burleigh-NASA/Jet Propulsion Laboratory, A. Hooke-NASA/Jet Propulsion Laboratory, L. Torgerson-NASA/Jet Propulsion Laboratory, R. Durst-The MITRE Corporation, K. Scott-The MITRE Corporation, K. Fall-Intel Corporation, H. Weiss-SPARTA, Inc. (2002, agosto). Delay-Tolerant Network Architecture: The Evolving Interplanetary Internet, Internet-Draft [En línea]. Disponible en: http://www.ipnsig.org/reports/draft-irtf-ipnrg-arch-01.txt.

[63] Carrier Grade Linux Working Group, Open Source Development Labs. (2007). Carrier Grade Linux Requirements Definition Overview, Version 4.0 [En línea]. Disponible en: http://developer.osdl.org/dev/cgl/cgl40/cgl40-overview.pdf.

[64] Him Cuper Cansaya Herrera, Arnau Sánchez Sala, César Cordova Bernuy. (2006, febrero). Interfaz USB para el control de estaciones de radio. ISSN 1818-728X 9771818728000 [En línea]. Disponible en: http://interno.ehas.org/intranet/comunicacion-1/comunicacion/articulos/forotelemed2005/memorias_i_forotelmed.pdf.

[65] EHAS. (2006). cvs: ax25/ehas-station [En línea]. Disponible en : http://www.ehas.org/cgi-bin/viewcvs.cgi/ax25/ehas-station/

[66] Arnau Sánchez Sala-Fundación EHAS, Oscar Ramos Moreno-Universidad Politécnica de Madrid, Eva Juliana Maya Ortiz-Universidad del Cauca. (2006, febrero). Sistema de gestión de redes EHAS. ISSN 1818-728X 9771818728000 [En línea]. Disponible en: http://interno.ehas.org/intranet/comunicacion-1/comunicacion/articulos/forotelemed2005/memorias_i_forotelmed.pdf.

[67] Avantcom Corporation. (2007, junio 2). Avantcom Corporation [En línea]. Disponible en: http://www.avantcom.com.

[68] SourceForge.Net. (2007, marzo 1). Net-SNMP [En línea]. Disponible en: http://net-snmp.sourceforge.net/.

[69] Tobi Oetiker. (2007, abril 6). MRTG, Multi Router Traffic Grapher [En línea]. Disponible en : http://oss.oetiker.ch/mrtg/.

[70] SourceForge.Net. (2003, abril 21). Cricket Home [En línea]. Disponible en: http://cricket.sourceforge.net/.

[71] The Cacti Group. (2007). Cacti, The complete rrdtool-based graphing solution [En línea]. Disponible en: http://cacti.net/.

[72] Tobias Oetiker. (2007, agosto 25). RRDtool, loggin & graphing [En línea]. Disponible en: http://oss.oetiker.ch/rrdtool/.

[73] NagiosCommunity. (2007, octubre 8). Nagios [En línea]. Disponible en: http://www.nagios.org/.

[74] Zabbix SIA. (2007). Zabbix. [En línea]. Disponible en: http://www.zabbix.com.

[75] Oscar Ramos-Universidad Politécnica de Madrid . (2005, junio 11). Sistema de gestión de red para el programa EHAS, Sistema de monitorización en diferido basado en correo electrónico, proyecto de fin de carrera [En línea]. Disponible en: http://interno.ehas.org/intranet/tecnologia/GestionRed/monitorizacion.pdf/view.