

HERMES: UN SISTEMA DE CORREO ELECTRONICO INTERUNIVERSITARIO

F. Aurtenechea, M. Hidalgo

Pontificia Universidad Católica de Chile
Santiago - Chile

1. INTRODUCCION

En la actualidad, el auge en el uso de la computación y la incorporación de redes de comunicación de datos, constituye una combinación muy útil para proveer, a distintas instituciones computacionales, mecanismos para compartir e intercambiar información.

Una aplicación fundamental para integrar usuarios de diversos computadores es el sistema de correo electrónico o correo basado en computadores [1,3,9,11,12,13]. Este trabajo describe el diseño de HERMES, un sistema de correo electrónico, cuya principal característica es su simplicidad y carencia de administración centralizada.

HERMES fue desarrollado en el Departamento de Computación de la Universidad Católica de Chile. Este primer diseño pretende dar los primeros pasos hacia una versión más completa y estándar para operar en el ámbito interuniversitario. Un servicio de intercambio de mensajes es fundamental para facilitar tanto el intercambio de información entre académicos e investigadores como el desarrollo de proyectos interuniversitarios.

En esta primera versión se han sacrificado ciertos aspectos de funcionalidad por simplicidad y transportabilidad del sistema. La simplicidad de su diseño hacen de HERMES un sistema que puede ser fácilmente implementado en diversas máquinas. La heterogeneidad de equipamiento computacional representa un escenario típico en las universidades. Esta primera versión de HERMES integra en funcionamiento a equipos con sistema operativo UNIX y VAX/VMS. Este tipo de equipos es muy utilizado en el ambiente universitario, lo que representa para HERMES una cobertura interesante de uso.

En un servicio de correo electrónico, la unidad de operación es la carta electrónica o mensaje de correo. HERMES provee un editor básico mediante el cual el usuario puede crear dichos mensajes para su posterior despacho. Además, integra la transferencia de archivos, siendo el recipiente de destino de éstos el mismo que él de los mensajes.

HERMES provee un servicio de despacho de mensajes certificados. Esto significa que el usuario puede solicitar la notificación del éxito o fracaso del despacho de ciertos mensajes. Si el mensaje llega correctamente a destino, el usuario recibe un reconocimiento positivo (mensaje tipo ACK) desde el nodo destino. En caso contrario, el usuario recibe un reconocimiento negativo (mensaje tipo NACK), ya sea desde el nodo destino (p.ej. si el usuario destino no existe) o desde algún nodo de enrutamiento (p.ej. si en la tabla de enrutamiento de éste, el nodo destino no existe).

Un nodo en HERMES es identificado por el par <Institución, Computador>, lo que representa el conjunto de instituciones de educación superior y sus recursos computacionales. Cada nodo puede mantener varias conexiones con nodos adyacentes, y además es responsable de mantener tablas para identificar los distintos nodos

destinatarios. Además, cualquier nodo puede operar en la modalidad de "store and forward" [9,11], almacenando mensajes en tránsito para su posterior despacho a otro destino. En este caso el nodo pasa a ser un nodo de enrutamiento. El enrutamiento es estático, manteniendo cada nodo una tabla de enrutamiento cuyas filas representan los destinos posibles y las columnas las diferentes líneas de salida, en orden de prioridad.

La transferencia de mensajes entre nodos se realiza en forma automática, a intervalos de tiempo establecidos para cada nodo en particular. Esta transferencia se efectúa en la modalidad maestro-esclavo. El nodo maestro es el que inicia la sesión de comunicación con el otro nodo (el nodo esclavo), activando en ambos nodos los procesos que realizan la transmisión y recepción de mensajes. Este esquema se repite por cada nodo adyacente al nodo maestro. Cualquier nodo HERMES puede ser un nodo maestro.

La comunicación es asíncrona a través interfaces RS232-C, ya sea mediante líneas locales ("null modems") o líneas telefónicas discadas o dedicadas. En este último caso, se proveen mecanismos para realizar discado y respuesta automática entre nodos remotos.

El modelo de comunicaciones usado contempla básicamente tres niveles. El nivel superior, denominado nivel de aplicación, provee los mecanismos básicos de interacción con el usuario para la creación, consulta y recepción de mensajes. El nivel medio o nivel de transporte ejecuta las solicitudes del nivel de aplicación organizando la correspondencia para su posterior despacho. Finalmente, el nivel bajo o nivel de comunicación es el responsable de interactuar con el medio y los protocolos de comunicación para permitir la transferencia de mensajes entre una máquina y otra. Este modelo representa una versión simplificada del modelo ISO [7] de redes de comunicación de datos.

En la actualidad, parte de este modelo está implementado usando protocolos y sistemas de comunicación existentes. Al respecto cabe señalar que el protocolo de comunicaciones corresponde a una versión modificada del protocolo KERMIT [4,5]. Por otra parte, el despacho final de correspondencia al usuario se realiza mediante un proceso (Cartero) que interactúa con el correo electrónico local (Mail) de cada computador en particular.

2. COMPONENTES BASICOS DE HERMES

A continuación se describe la arquitectura general y las componentes lógicas involucradas en el diseño de HERMES.

2.1 Arquitectura General

Cada nodo de HERMES es una entidad autónoma, que maneja un servicio simple de enrutamiento de mensajes entre los demás nodos. Las funciones de cada nodo son fundamentalmente, edición y almacenamiento de correspondencia, certificación de mensajes, despacho local y redirección remota de correspondencia, y transmisión y *scheduling* automático de mensajes entre nodos adyacentes.

En la figura 2.1 se muestra las principales componentes de la arquitectura HERMES. En base a esta figura, a continuación se describe la operación general de un nodo HERMES, en función del flujo de información que se maneja y de los procesos involucrados. Los procesos y componentes fundamentales de esta arquitectura se describen más adelante.

- (1) El usuario prepara un mensaje con ayuda del editor de HERMES.
- (1)' El usuario puede reenviar o cancelar un mensaje certificado que recibió un NACK.

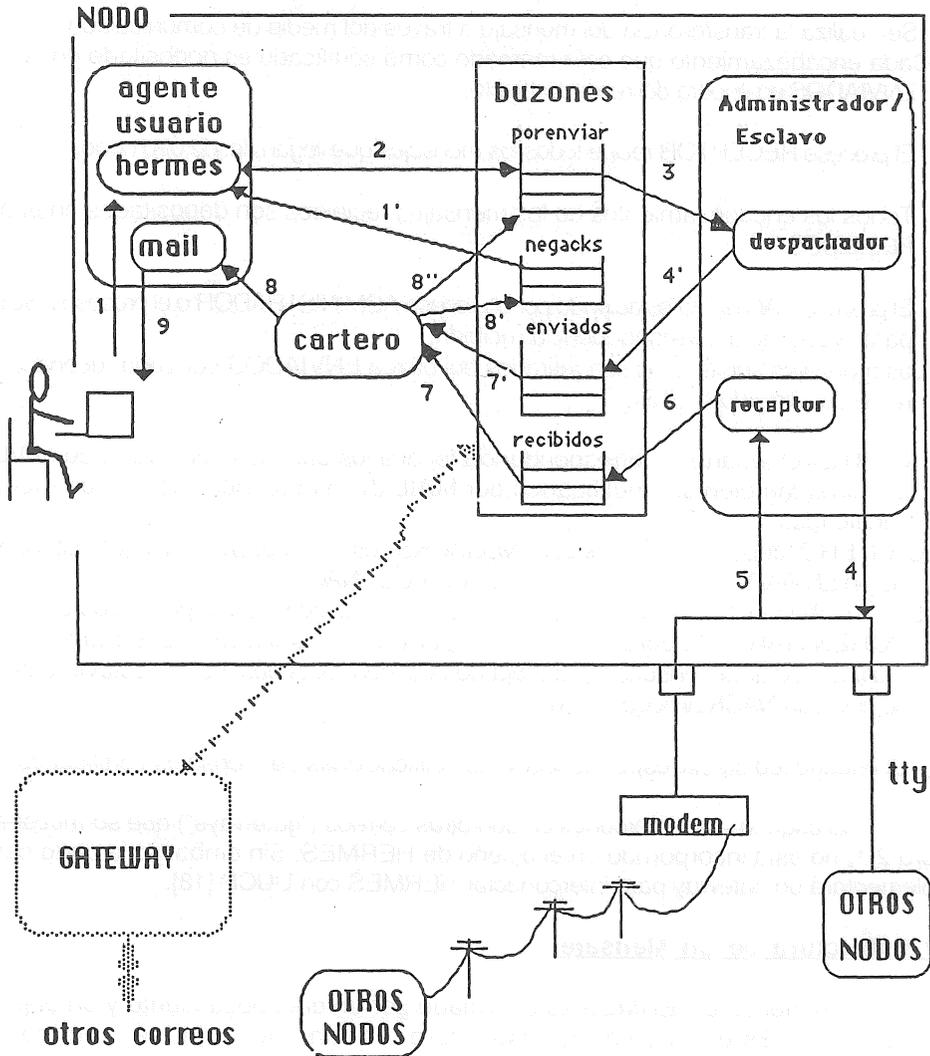


Figura 2.1 Operación General de un Nodo HERMES

- (2) El encabezamiento del mensaje es almacenado en el buzón PORENVIAR. Este queda allí pendiente hasta que el proceso DESPACHADOR lo transmita. El usuario tiene la posibilidad, mientras el mensaje esté en estado pendiente, de cancelarlo o modificarle parámetros. El texto del mensaje se maneja por separado (ver sección 2.3).
- (3) El proceso ADMINISTRADOR o proceso ESCLAVO activa al proceso DESPACHADOR para que comience el despacho de los mensajes pendientes del buzón PORENVIAR.
- (4) Se realiza la transferencia del mensaje a través del medio de comunicación.
- (4)' Cada encabezamiento que esté marcado como certificado es depositado en el buzón ENVIADOS en espera del reconocimiento.
- (5) El proceso RECEPTOR recibe todos los mensajes que llegan desde otro nodo.
- (6) Todos los encabezamientos de los mensajes recibidos son depositados en el buzón RECIBIDOS.
- (7) El proceso CARTERO es activado por el proceso ADMINISTRADOR o el proceso ESCLAVO para procesar la correspondencia recibida.
- (7) Los mensajes certificados son retirados del buzón ENVIADOS por cada reconocimiento recibido, dirigido a ellos.
- (8) CARTERO reparte la correspondencia local a los usuarios con ayuda de MAIL. Los usuarios también son notificados por MAIL de los reconocimientos de mensajes certificados.
- (8)' CARTERO deposita en el buzón NEGACKS los encabezamientos que estaban en el buzón ENVIADOS a los cuales se les ha enviado un NACK.
- (8)" Los encabezamientos de mensajes con destino no local son depositados en el buzón PORENVIAR para continuar su transporte (enrutamiento). El enrutamiento se realiza mediante consultas a la tabla de destinos. Si el destino no estuviera definido, se envía un NACK al nodo origen.
- (9) El usuario lee su correspondencia y las notificaciones de mensajes certificados.

El esquema de interconexión con otros correos ("gateways") que se muestra en la figura 2.1, no está incorporado en el diseño de HERMES. Sin embargo, a corto plazo se implementará un gateway para interconectar HERMES con UUCP [13].

2.2 Estructura de un Mensaje

Un mensaje HERMES está formado por un encabezamiento y un objeto (ver figura 2.2). Este modelo de estructura de un mensaje se conoce como texto-encabezamiento.

El encabezamiento contiene la información necesaria para permitir la recepción y despacho de los mensajes. El objeto es la información que el usuario desea transferir y no es interpretada para los efectos de administración y manejo de mensajes.

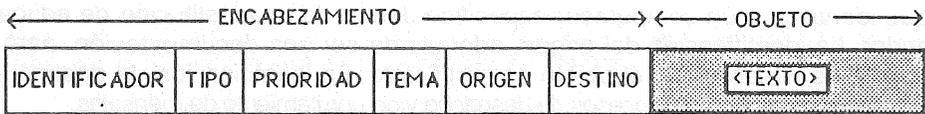


FIGURA 2.2 Estructura de un Mensaje

El encabezamiento de un mensaje HERMES está compuesto por los campos: Identificador, Tipo, Prioridad, Tema, Origen y Destino. A continuación se describe cada uno de éstos.

El campo **Identificador** asocia una identidad única a todos los mensajes residentes en un nodo. Su propósito es permitir hacer referencias del mensaje (p.ej. operaciones sobre éste por parte del usuario o envío de reconocimientos). Este identificador se usa, además, como una forma de documentación, ya que está formado por la fecha y hora en que el mensaje ha sido enviado por el usuario.

El campo **Tipo** se usa para distinguir grupos de mensajes. Existen cuatro tipos de encabezamientos: Los de tipo objeto (objetos certificados o no certificados), que tienen asociado un objeto físico y los de tipo reconocimiento (ACK o NACK), que no tienen asociado un objeto físico.

Si un objeto es certificado, el usuario del mensaje recibirá una notificación del éxito o fracaso de su llegada a destino.

Si el tipo es ACK, se trata de un reconocimiento de un mensaje certificado que llegó a destino exitosamente. Si el tipo es NACK, se trata de un reconocimiento de un mensaje certificado que no llegó a destino por algún motivo (p.ej. el usuario de destino no existe).

El campo **Prioridad** sirve para jerarquizar el despacho de correspondencia. Actualmente, el sistema asigna prioridades por defecto a los mensajes; por ejemplo, un mensaje certificado tiene una prioridad mayor que un mensaje sin certificar. Estas pueden ser ajustadas por el usuario al momento de invocar el comando. El sistema de prioridades, potencialmente, puede sofisticarse asignando prioridades según grupos de usuarios y tamaños de mensajes.

El campo **Tema** es un string de caracteres, asociado al mensaje, provisto por el usuario a modo de documentación. Este es el único campo del encabezamiento no interpretado por HERMES.

Los campos **Origen** y **Destino** identifican al usuario originador y usuario destinatario del mensaje respectivamente. Cada campo está constituido por la tupla <Nodo, Id-usuario >. El campo *Id-usuario* representa al identificador del usuario en el computador (p.ej. JPEREZ). El campo *Nodo* corresponde al par <institución, computador >

lo que identifica a un computador específico dentro de una institución de educación superior. La identificación del origen, además de ser una documentación para los usuarios, es necesaria si el objeto es certificado. La identificación de destino es la información clave para los procesos de despacho y de enrutamiento de mensajes.

2.3 Sistema de Buzones y Almacenamiento de Mensajes

Dado que HERMES es un sistema de correo electrónico de almacenaje y despacho, es necesario dar una administración eficiente a los mensajes. Si los encabezamientos se manipulan por separado, se logra un manejo ágil de éstos permitiendo diferentes clasificaciones y ordenamientos. Para ello, existe un sistema común de buzones dedicados a contener exclusivamente los encabezamientos; por otra parte, existe un área particular (área HERMES) para la residencia de los objetos.

Cada encabezamiento de tipo objeto, almacenado en los buzones, apunta al lugar de residencia de su objeto asociado. Los objetos creados por el usuario, con el editor del correo, residen en área HERMES. Algunos objetos residen en los directorios de los usuarios; esto ocurre cuando el usuario escoge la opción de hacer un traspaso de un archivo que previamente existía en su directorio. En este caso, a diferencia de los demás mensajes, el usuario es dueño del mensaje y no HERMES.

Los encabezamientos de tipo reconocimiento no apuntan a objetos; éstos son sólo encabezamientos de control.

Como se mostró en la figura 2.1, HERMES maneja cuatro tipos de buzones. El buzón PORENVIAR contiene los encabezamientos de los objetos que deben salir del nodo. Estos encabezamientos pueden ser de origen local o externo. Los primeros corresponden a encabezamientos de objetos generados por usuarios locales, y los segundos a encabezamientos de objetos no locales que están en tránsito por el nodo.

El buzón ENVIADOS contiene los encabezamientos de objetos locales que efectivamente fueron despachados y que esperan ser reconocidos (objetos certificados). La existencia de este buzón permite al usuario conocer el estado de su correspondencia certificada y también efectuar un posible reenvío en caso de fracasar el traspaso a destino.

El buzón RECIBIDOS contiene todos los encabezamientos de los objetos que han llegado al nodo y que no han sido procesados.

Finalmente, el buzón NEGACKS contiene los encabezamientos de objetos locales para los cuales su traspaso a destino fracasó (mensajes que han recibido un nack). Este buzón permite al usuario efectuar un reenvío corrigiendo el tipo de error mediante algún cambio en los parámetros del comando.

3 MODELO DE COMUNICACION

HERMES ha sido modelado particionando el sistema en componentes funcionales,

cada una dedicada a un conjunto de tareas específicas. Estas componentes han sido particionadas en tres niveles, los cuales corresponden a una simplificación del modelo de referencia de comunicaciones ISO. A continuación se describe cada uno de estos niveles.

3.1 Nivel De Aplicación

Este nivel, también llamado agente del usuario (AU), es el nivel más alto. Interactúa con el usuario o, de otro modo, es la interfaz entre el usuario y los servicios de despacho y recepción. Las componentes de este nivel asumen todas las funciones que están directamente bajo control de una persona, actuando en favor de ella, asistiéndola en el envío y lectura de mensajes. El AU permite una comunicación persona a persona muy parecida a la que se lograría con un sistema de correo convencional.

Este nivel provee un conjunto de comandos para atender los requerimientos del usuario. Estos comandos son los siguientes:

- Enviar: El usuario puede crear (editar) y enviar un mensaje o puede enviar un archivo ya preparado en su directorio.
- Reenviar: El usuario puede cambiar los parámetros de un mensaje que no ha sido despachado aún o cuyo traspaso a destino fracasó.
- Buzón: El usuario puede listar los encabezamientos de los mensajes que están por despacharse, los que ya han sido despachados y esperan ser reconocidos o aquellos que han recibido un NACK.
- Cancelar: Es posible cancelar un mensaje que aún no ha sido despachado o que ha recibido un NACK.

Una vez efectuado un comando, este nivel procede a ajustar los parámetros para los requerimientos del nivel de transporte, utilizando el sistema de buzones. Esto incluye la definición del encabezamiento del mensaje, el depósito de éste en el buzón correspondiente y el almacenamiento del objeto si es necesario. Además, este nivel se encarga de entregar la correspondencia al usuario para que éste pueda leerla.

Con respecto al reparto de mensajes que llegan, el sistema realiza una interacción directa con el correo local incorporado en cada máquina (MAIL), el cual finalmente entrega la correspondencia a los usuarios. De ese modo, el usuario puede mirar la correspondencia HERMES de la misma manera que normalmente lo hace mediante su correo local. Este esquema se usa para el reparto, tanto de mensajes como de archivos. Luego, desde el punto de vista de la entrega de correspondencia, no existe diferencia entre ambos tipos de objetos.

3.2 Nivel De Transporte

La función exclusiva de las componentes del nivel de transporte es permitir, con la ayuda del nivel de comunicación, el despacho y la recepción de mensajes entre dos nodos, estableciendo la sincronización necesaria de los procesos de despacho y recepción

entre dos nodos. Además, procesa los mensajes recibidos, incluyendo enrutamiento y reparto de mensajes a usuarios.

Las componentes de este nivel son básicamente, el sistema común de buzones, el área de almacenamiento de los mensajes y las tablas de destinos y líneas que facilitan el enrutamiento de mensajes y el establecimiento de sesiones de comunicación entre nodos adyacentes. Una sesión consiste en la conexión remota entre dos nodos (login remoto), la transferencia de correspondencia entre ambos y, finalmente, su desconexión.

El despacho y recepción de mensajes, se realiza, en forma automática, mediante sesiones periódicas establecidas entre dos nodos. En una sesión hay un nodo que contiene el proceso administrador, que actúa como nodo maestro y el otro es el nodo esclavo. El nodo esclavo es activado por el nodo maestro cada vez que este último decide establecer una sesión. Esto significa que el traspaso de mensajes en el nodo esclavo depende directamente del nodo maestro. Es posible descentralizar la responsabilidad haciendo coexistir en cada nodo el maestro y el esclavo. Con la existencia simultánea de ambos, la iniciativa de los traspasos de mensajes se reparte desde el momento en que ambos nodos pueden decidir establecer una sesión, pero manteniendo la relación maestro-esclavo en una sesión cualquiera.

El administrador tiene a su cargo todas las líneas de los nodos vecinos, las que recorre permanentemente para establecer sesiones. En cada sesión, el administrador sincroniza su proceso DESPACHADOR con el proceso RECEPTOR del esclavo para vaciar su buzón PORENVIAR en el buzón RECIBIDOS del nodo esclavo. A continuación se establece el sincronismo para los procesos RECEPTOR del nodo maestro y DESPACHADOR del nodo esclavo. Así, el esclavo vacía su buzón PORENVIAR traspasando la información al buzón RECIBIDOS del maestro. Una vez que el administrador ha recorrido todas las líneas, procede a repartir o enrutar la correspondencia recibida mediante el proceso CARTERO. El esclavo, realiza el reparto y enrutamiento en su nodo, inmediatamente finalizada su sesión particular.

Este esquema maestro-esclavo es necesario por razones de eficiencia y de disponibilidad restringida de líneas. Para el acceso y liberación de líneas se manejan tres parámetros por línea: El horario disponible de la línea, el número de intentos fallidos en el acceso dentro del horario y el horario de la próxima visita a la línea (ver figura 3.2).

Las líneas están normalmente disponibles sólo a ciertas horas del día, de modo que los traspasos por una línea deben regirse por su horario de disponibilidad. Si el número de intentos fallidos en usar una línea excede un cierto límite, se notifica en un archivo bitácora y la línea es ignorada en los próximos accesos. En cuanto al horario de la próxima visita, después de terminar una sesión, se espera un cierto lapso de tiempo (p.ej. 1 hora) para intentar la próxima sesión, liberando el recurso durante ese lapso.

El administrador también se rige por un horario; no es adecuado que éste opere en horas de un flujo nulo de mensajes, aún en horas disponibles de alguna línea. Los horarios del administrador y de las líneas son particulares de cada institución.

Actualmente, el programa administrador está implementado en una máquina

UNIX y el programa esclavo en una máquina VMS. Con ello se provee una amplia cobertura de comunicación a nivel universitario dada la gran disponibilidad de estos equipos en las universidades.



Figura 3.2 Esquema Administrador Esclavo

3.3 Nivel de Comunicación

Este nivel es el encargado de establecer la transferencia de mensajes, libre de errores, a través del medio de comunicación. Este nivel es básicamente una adaptación y automatización del protocolo Kermit. La gran disponibilidad de este protocolo en diversas máquinas motivó su adaptación, lo que facilita la portabilidad del nivel de comunicación de HERMES a otras máquinas.

El traspaso de mensajes se realiza mediante los procesos DESPACHADOR y RECEPTOR. Estos procesos automatizan la transferencia de grupos de mensajes dentro del protocolo Kermit, y son activados por el proceso administrador una vez que el nodo maestro ha establecido la conexión con el nodo esclavo. Esta conexión equivale a un "login remoto" desde el nodo maestro. Cada nodo administrador hace uso de tablas que contienen los procedimientos para establecer conexiones con sus nodos vecinos.

4. CONCLUSIONES

Hermes es un sistema de correo electrónico distribuido que provee las operaciones básicas de despacho y recepción de mensajes usuario a usuario. Su objetivo principal es proporcionar un servicio básico de intercambio de información entre equipos heterogéneos tal como se presenta en el ámbito interuniversitario.

En lo que respecta al nivel de comunicación entre nodos, el protocolo de transferencia de mensajes está basado en el protocolo Kermit. Esto, además de asegurar la confiabilidad de las transmisiones, facilita la portabilidad del correo hacia diversos computadores.

Hermes provee un editor básico de texto, mediante el cual los usuarios pueden crear mensajes (archivos ASCII) los cuales conforman la correspondencia básica de

Hermes. Además, permite el envío de archivos previamente creados en el directorio del usuario (p.ej. un archivo de datos). Sin embargo la distinción entre mensajes y archivos no es transparente para el usuario, ya que el recipiente de destino de ambos es el mismo.

El despacho de correspondencia puede ser certificado. Esto significa que el usuario puede pedir la notificación del éxito o fracaso de la llegada a destino de ciertos mensajes.

El transporte de mensajes entre un nodo y otro es completamente transparente para el usuario. El sistema establece sesiones de comunicación entre dos nodos adyacentes, en forma automática, y provee mecanismos para realizar discado y respuesta automática entre nodos remotos.

El despacho de correspondencia se realiza mediante un esquema de intervalos de tiempo, que es paramétrico por cada línea de comunicación. La frecuencia de verificación del acceso a cada línea depende del horario del administrador del nodo. Estos valores pueden ser ajustados externamente. El programa administrador verifica el comportamiento de cada línea, y puede desactivar automáticamente una línea si ésta presenta fallas reiteradas en la transferencia de mensajes.

Cada nodo opera en la modalidad de "store and forward". Esto es, parte de la correspondencia recibida por un nodo puede estar constituida por mensajes "en tránsito" hacia otro nodo. Los mensajes recibidos cuyo destino no es el nodo local, son depositados en el buzón de salida, previa verificación de la existencia del nodo destino.

No existe en el nivel de aplicación (agente usuario), una integración entre la generación y lectura de la correspondencia. Ambas funciones se realizan en forma independiente. Actualmente, se utiliza el correo local de cada nodo para llevar finalmente la correspondencia al usuario. Esto limita la funcionalidad del correo electrónico, en cuanto al redireccionamiento de mensajes. Sería conveniente una interacción más directa con el correo local para permitir anotaciones adicionales en el mensaje, por parte del usuario, para ser posteriormente reenviado, al usuario de origen (respuesta) o a otro usuario ("forwarding").

HERMES corresponde a una primera etapa en el desarrollo de un sistema de correo electrónico interuniversitario. En el futuro se pretende ampliar su funcionalidad, seguridad y robustez (p.ej. incorporar un nivel de presentación que permita facilidades de codificación de la correspondencia), para incorporarlo como una aplicación de una futura red de computadores para las instituciones de educación superior. Una preocupación a corto plazo es la de implementar sistemas de interconexión ("gateways") de correos electrónicos para permitir, tanto la comunicación de correos incompatibles (p.ej. HERMES-UUCP) como el acceso a servicios de correo electrónicos públicos (p.ej. TELEMAIL).

BIBLIOGRAFIA

- [1] R. A. A. BUHR, D. A. MACKINNON; "Mailroom: A Computer-Based Message System Model for Person-to-Person and Process-to-Process Communication". The 3rd International Conference on Distributed Computing Systems. Miami/Ft. Lauderdale, Florida. October 18-22, pp. 818-823, 1982.
- [2] DOUGLAS COMER, JOHN T. KORB; "CSNET Protocol Software: The IP-to-X.25 Interface". SIGCOMM'83 Symposium, Communications Architectures & Protocols, University of Texas at Austin, pp. 154-159, March 8 & 9, 1983.
- [3] IAN CUNNINGHAM; "Electronic Mail Standards to Get Rubber-Stamped and Go Worldwide". Data Communications, May 1984, p. 159.
- [4] FRANK DA CRUZ, BILL CATCHINGS; "Kermit: A File Transfer Protocol for Universities. Part 1: Design Considerations and Specifications". Byte, p. 255, June 1984.
- [5] FRANK DA CRUZ, BILL CATCHINGS; "Kermit: A File Transfer Protocol for Universities. Part 2: States and Transitions, Heuristic Rules and Examples", Byte, p. 143, July 1984.
- [6] PETER J. DENNING, ANTHONY HEARN, C. WILLIAM KERN; "History and Overview of CSNET". SIGCOMM'83 Symposium, Communications Architectures & Protocols, University of Texas at Austin, pp. 138-145, March 8 & 9, 1983.
- [7] ISO TC97/SC16 "Open System Interconnection - Basic Reference Model, (ISO Draft proposal 7498)". Computer Communication Review, pp 15-65, April 1981.
- [8] WARWICK S. FORD; "Portable Implementation of Network Architecture Layers". SIGCOMM'83 Symposium, Communications Architectures & Protocols, University of Texas at Austin, pp. 240-245, March 8 & 9, 1983.
- [9] JOSE GARCIA-LUNA-ACEVES, FRANKLIN F. KUO; "Design Issues of Protocols for Computer Mail". Seventh Data Communications Symposium -1981. Maria Isabel Sheraton Hotel Mexico City, Mexico. October 26-29, pp. 28-36, 1981.
- [10] ROBERT GRAFF; "Modeling an Electronic Mail Network: A Primer". Data Communications, p. 163, August, 1983.
- [11] L. LANDWEBER, M. LITZKOW, D. NEUHENGEN, M. SOLOMON; "Architecture of the CSNET Name Server". SIGCOMM'83 Symposium, Communications Architectures & Protocols, University of Texas at Austin, pp. 146-153, March 8 & 9, 1983.
- [12] LOH-PING YU; "The Anatomy of a Distributed Electronic Mail Network". Data Communications, p. 153, March 1985.

- [13] D. A. NOWITZ; "UUCP Implementation Description". Ultrix-32 Supplementary Documents, Vol. 3.
- [14] PETER SCHICKER; "The Computer Based Mail Environment - An Overview". Computer Networks 5, pp. 435-443, 1981.
- [15] GERALD TOMANEK; "Implementing Electronic Mail in a Telephone System: More Than Just Talk". Proceedings, National Computer Conference, pp. 527-531, 1980.