



Transmisión punto a punto de grandes volúmenes de datos

Transfer of Very Large Files between Two End Users

◆ J. A. Corrales, F. García Pandavenes y R. Muñiz

Resumen

A pesar del gran uso que tiene Internet en la actualidad, un usuario final se va a encontrar con muchos problemas al transmitir grandes cantidades de datos. Tanto los protocolos como las aplicaciones disponibles no fueron diseñados para estos casos donde la latencia de la red, su inestabilidad y la gran cantidad de tiempo requerido para la transmisión juegan en contra de una entrega adecuada. En este trabajo presentamos una herramienta para enviar archivos muy grandes entre dos usuarios finales.

Palabras clave: Transferencia de archivos muy grandes, envío de grandes cantidades de datos.

Summary

Despite the current full use of Internet today, an end user may experience problems when transmitting large amounts of data. The available protocols and applications have not been designed for these cases where the latency of the network, their instability and the high amount of time required for the transmission play against the successful delivery. In this work we present a tool to send very large files between two end users.

Keywords: Transfer of very large files. Sending large amounts of data.

1.- Introducción

Es bien conocida la evolución de Internet, desde la primitiva empleada básicamente para el envío de información textual, una segunda fase con predominio de información gráfica estática hasta la actual donde ha desembarcado la información dinámica tanto en su versión de audio como de vídeo. Este crecimiento ha ido de la mano del aumento de la potencia de los sistemas informáticos tanto en velocidad de proceso como en capacidad de almacenamiento y velocidad de acceso a la red.

No obstante, un usuario final que hoy en día intente enviar a otro un archivo de algunos centenares de megabytes se va a encontrar con muchos problemas para ello, pese a que tal cantidad de información es relativamente pequeña si la comparamos con la capacidad que seguramente tiene el disco duro de su equipo.

Probablemente trate de usar en un principio el correo electrónico para llevarse la desagradable sorpresa de que su estafeta no le permite el envío de archivos tan grandes. Quizás intente usar alguna otra cosa más, tal como configurar un servidor web o ftp o bien usar alguno ya existente en su organización, pero lo más probable es que acabe por abandonar la idea de la transmisión electrónica y que tras grabarlo en un CDROM lo envíe por correo postal ordinario a su destinatario final.

2.- Alternativas

Realmente nuestro usuario no tiene muchas alternativas. Estas son:

- 1) Correo electrónico: No le va a permitir el envío de un archivo muy grande dado que la mayor parte de las estafetas han sido configuradas para aceptar mensajes de un tamaño máximo que

◆
Un usuario final que hoy en día intente enviar a otro un archivo de algunos centenares de megabytes se va a encontrar con muchos problemas

◆
El correo electrónico no le va a permitir el envío de un archivo muy grande pues la mayor parte de las estafetas han sido configuradas para aceptar mensajes no superiores a las decenas de megabytes



generalmente ronda las decenas de megabytes. Incluso suponiendo que su mensaje se acepte es difícil que el buzón del destinatario tenga capacidad para almacenarlo. Incluso proveedores de correo como Gmail [1] que proporcionan más de 2 GB rechazarían el envío de una imagen ISO de un DVD (4.7 GB). A todo ello habría que añadir además las dificultades para recibir el mensaje desde el servidor de correo hasta el equipo del usuario.

- 2) Transferencia de archivos: El remitente podría poner el archivo a enviar en algún servidor de archivos, bien WWW o bien FTP proporcionando al destinatario una clave para acceder al documento. Esto requiere bien configurar un servidor en el propio equipo del remitente o bien usar algún servidor ya establecido de su organización en cuyo caso se duplican los problemas, primero debe llevar su archivo grande al servidor de su organización y segundo el destinatario debe encontrar la manera de transferirlo a su propio equipo.

Es difícil que vía WWW o vía FTP se mantenga la conexión durante varios días o semanas. Cortes de los operadores de telefonía, reinicios de equipos de comunicaciones y otras operaciones de mantenimiento hacen que Internet sea inherentemente de poca confianza en este aspecto. Lo más probable es que la transmisión quede interrumpida y deba reiniciarse de nuevo desde el principio, perdiendo desgraciadamente todo lo hecho.

- 3) Troceamiento del archivo: Es la primera alternativa seria que se puede emplear. Hay multitud de aplicaciones que permiten romper un archivo en varias partes de menor tamaño y reconstruirlo posteriormente. El remitente podría fragmentar su archivo en trozos lo suficientemente pequeños como para poder ser enviados por correo electrónico o a través de un servidor WWW o FTP y el destinatario podría recomponer el archivo original tras recibir correctamente cada una de las partes.
- 4) Programas P2P: Es posible configurar Emule [2] y otras aplicaciones "peer to peer" para enviar archivos grandes a una persona en particular. No obstante no es sencillo dado que estas aplicaciones no han sido diseñadas para esto sino para compartir archivos con otros muchos usuarios simultáneamente. Aparte de los potenciales problemas de seguridad, en algunas organizaciones este tráfico P2P está filtrado.
- 5) Otras posibilidades: No existen de forma estándar y/o uso difundido. Pueden clasificarse como aquellas que permiten la comunicación directa entre remitente y destinatario y las que requieren usar un servidor interpuesto. De entre las primeras podríamos citar el "File Tunnel" de Gatherbird [3] y de entre las segundas Yousendit [4] aunque hay soluciones intermedias que en algunos casos es difícil distinguirlos de un sistema de backup remoto [5] o de un almacenamiento global [6]. Desde luego las que requieren un servidor interpuesto son rechazables: consideremos el coste que implica enviar todos los datos del equipo del remitente a un servidor que puede estar en el otro lado del planeta para volver a descargarlos al equipo del destinatario que podría estar mucho más cercano geográficamente al remitente.

La aplicación que proponemos se parece bastante a las del primer tipo aunque con algunas mejoras como veremos más adelante. De cualquier manera cumple una característica básica común con todas las de estos dos últimos tipos y que consiste en la tolerancia a fallos.

3.- Análisis de requisitos

Para determinar los requisitos de la aplicación se ha pensado en un usuario medio típico de la comunidad universitaria que por cualquier motivo se encuentra con el problema de transferir uno o



Es difícil que vía WWW o vía FTP se mantenga la conexión durante varios días o semanas



El troceamiento del archivo es la primera alternativa seria que se puede emplear



El requisito más fundamental es el sistema tolerante a fallos, que en caso de caída de la red o interrupción de la transmisión, debe reanudarse en el mismo punto donde se había quedado detenida sin perder el trabajo previo

Teniendo en cuenta que el sistema operativo más difundido por los usuarios medios es con mucho Windows, se desarrolló la aplicación en C#

varios archivos grandes a otro usuario –un colega quizás– que se encuentra en parecida situación. Con ello en mente se han encontrado lo que serían las características que debería cumplir y que son:

- 1) De uso sencillo para usuarios normales sin conocimientos especiales de informática o comunicaciones.
- 2) Ejecutable único que no requiera ni siquiera instalación. Esto se traduce en que no van a ser necesarios privilegios de administración bajo una plataforma Windows.
- 3) Aplicación única. Que opere al mismo tiempo como cliente y como servidor.
- 4) Que ocupe poco tamaño (del orden de muy pocos cientos de KB) para que se pueda redistribuir fácilmente por correo electrónico.
- 5) Con transmisión punto a punto desde el equipo original hasta el destinatario, sin que haga falta ningún tipo de servidor intermedio. De esta forma se evita la sobrecarga de transmisiones innecesarias y la dependencia de terceros.
- 6) Sistema tolerante a fallos. En caso de caída de la red o de interrupción de la transmisión, debe reanudarse en el mismo punto donde se había quedado detenida sin perder el trabajo previo. Este requisito es desde luego el más fundamental de todos.

Adicionalmente se han considerado también **otras cualidades** que pueden catalogarse como **convenientes aunque no imprescindibles**. A saber:

- 7) Aplicación multihebra: se transmite en cada hebra un fragmento distinto de cada fichero para aprovechar al máximo todo el ancho de banda disponible.
- 8) Números de puertos configurables por si hubiera cortafuegos y no se pudieran emplear los predeterminados.
- 9) Incorporar un canal privado de conversación tipo “chat” para facilitar la comunicación verbal entre el usuario emisor y el receptor.
- 10) Ayuda integrada HTML.
- 11) Soporte multilingüe.

Con estas condiciones y teniendo en cuenta que el sistema operativo más difundido por los usuarios medios es con mucho Windows, se desarrolló la aplicación en C# de manera que corre bajo las plataformas Windows 2000, Windows Server 2003 y Windows XP con el entorno .NET (runtime) [7].

Aunque las nuevas versiones de Windows lo incluirán integrado, aquellos usuarios con versiones antiguas deberán tener actualizado su sistema descargando de Microsoft e instalando el entorno .NET (runtime) antes de ejecutar la aplicación propuesta.

4.- Resultados y conclusiones

El funcionamiento es muy sencillo. Ambos usuarios deben ejecutar la aplicación. El transmisor debe indicar primero qué ficheros desea enviar arrastrándolos a la ventana de envío. Luego debe escribir la

dirección IP del destinatario e iniciar la transmisión. Al receptor le llega una solicitud de envío y debe aceptarla explícitamente (o rechazarla). Caso de que la acepte, se inicia la transmisión. Tanto los directorios de envío como los de recepción se crean por defecto en el mismo lugar donde se encuentra la aplicación si los usuarios no especifican otros distintos, lo cual supone una medida de seguridad razonable.

Los archivos que se envían se trocean internamente y se transmite cada parte de forma independiente intentando así aprovechar el máximo ancho de banda. Si por cualquier circunstancia se cae la red o se apaga alguno de los equipos se reanudará la transferencia en el mismo punto en que se había quedado en cuanto exista conectividad de nuevo.

El desarrollo de la aplicación se concibió como un proyecto fin de carrera en la Escuela Politécnica Superior de Ingeniería de Gijón realizándose sin ningún problema digno de mención. Las pruebas hechas con una versión beta de la aplicación permitieron enviar ficheros del orden del gigabyte en tiempos de alrededor de la decena de minutos, notándose una dependencia fuerte de la hora del día, es decir del tráfico de la red.

Un segundo grupo de pruebas se realizó entre un equipo de RedIRIS y otro con ADSL doméstico situado tras un cortafuegos con NAT. En tal caso y al igual que con otras aplicaciones en idénticas circunstancias es necesario configurar el router/cortafuegos para redireccionar los puertos empleados por la aplicación hacia equipo deseado.

Debe destacarse que aunque la aplicación no está pensada para trabajar tras un cortafuegos funciona satisfactoriamente en tales circunstancias y debería funcionar también incluso si los dos equipos terminales (remitente y destinatario) se encontrasen en estas condiciones. El único problema que se ha detectado es la determinación de la dirección IP del interface WAN del router/cortafuegos metropolitano ya que cada usuario debe conocer la IP del otro. Hay suficientes sitios de Internet donde se nos indica cual es la dirección IP bajo la cual estamos conectados pero esto supone ya una dependencia con terceras partes.

Como conclusiones del presente trabajo, quizás lo más reseñable sea el haber detectado nichos de aplicaciones en Internet que todavía no están debidamente cubiertos, y que en el caso que nos ocupa aunque el número potencial de usuarios es escaso probablemente vaya en aumento en los próximos años.

En cualquier caso, enviar un gran archivo con los resultados de un experimento, con un vídeo didáctico, con una imagen ISO de un CD o DVD, con un backup de un disco o con un kit de instalación de un sistema operativo será cada vez menos problemático.



Los archivos que se envían se trocean internamente y se transmite cada parte de forma independiente intentando así aprovechar el máximo ancho de banda



Los archivos que se envían se trocean internamente y se transmite cada parte de forma independiente intentando así aprovechar el máximo ancho de banda



Referencias

- [1] A Google Approach to Email.
Consultado en <http://www.gmail.com/> el 1-9-2005
- [2] Peer to Peer File Sharing.
Consultado en <http://www.emule.org/> el 1-9-2005
- [3] Gatherbird File Tunnel.
Consultado en <http://www.gatherbird.com/> el 1-9-2005
- [4] Yousendit.
Consultado en <http://www.yousendit.com/> el 1-9-2005
- [5] Viceversa file Synchronization.
Consultado en <http://www.tgrmn.com/> el 1-9-2005
- [6] Streamload.
Consultado en <http://www.streamload.com/> el 1-9-2005
- [7] Microsoft .NET Framework Development Center.
Consultado en <http://msdn.microsoft.com/netframework/> el 1-9-2005

José A. Corrales
(ja@lsi.uniovi.es)
Félix García Pandavenes
(fgpanda@hotmail.com)
Rubén Muñiz Sánchez
(rubenms@lsi.uniovi.es)
Universidad de Oviedo
Departamento de Informática