Protocolos de Comunicación

Cuando hay una comunicación entre 2 procesos a través de internet aparecen los protocolos para indicar como se realizan estas comunicaciones. Hay 2 principales tipos:

- basados en texto: se envían cadenas de caracteres en una representación ASCII.
- De Solicitud-Respuesta: Un lado invoca una petición y espera la respuesta de la otra parte.

Protocolo HTTP

- Hyper Text Transfer Protocol
- Basado en la definición de URL, Universal Resource Locator. Que apunta a un recurso.
- Un cliente realiza una petición al servidor y este le envía el resultado de realizar la acción.
- HTTP 0.9 tiene texto sin formato, mientras que el HTTP 1.0 incluye formatos, el HTTP 1.1 incluye nuevos servicios y reinicio de comunicaciones.

Características

- Utilizado por el 100% de usuarios y servidores
- URL: Objeto al que se le suele pedir la ejecución de un método y obtener unos resultados en función de los parámetros
- Protocolo tan genérico que permite definir tus propios comportamientos
- No recuerda peticiones de clientes, utiliza cookies para saber peticiones anteriores (Fichero con información que se guarda en el navegador.) Las cookies se envían al servidor junto a la petición.

Mensajes

- 3 cabeceras: General Header, Response Header y Entity Headar pormadas por pares:
    - field-name:field-value<CRLF>
- Dos tipos de mensajes
    - Petición
    - Respuesta

Estructura del mensaje:

Primera Línea	Header	CRLF	Entity Body

Mensaje de petición

- Primera Línea
    - Método(Get, Post, Head y otros) a aplicar sobre:
        - Recurso elegido(URL)
        - Versión de HTTP
- Cabeceras con los pares campo-valor
- Cuerpo de la petición

    - Información extra para el servidor, puede contener parámetros para el método.

Mensajes de Respuesta

- Primera línea (de estado)
    - Versión de HTTP
    - Código de estado de 3 dígitos
    - Frase descriptiva
- Cabeceras con los pares campo-valor
- Cuerpo de la Respuesta
    - Información extra para el cliente

Métodos HTTP 1.0

- GET: Recupera URL especificada
- POST: Envía datos al URL especificado
- HEAD: igual que GET pero solo retorna carreteras

Métodos HTTP 1.1

- PUT: Guarda los datos en la URL
- PATCH: Como PUT pero con una lista de diferencias
- COPY: Copia el contenido del recurso a las direcciones especificadas
- MOVE: mueve el contenido del recurso a la dirección especificada
- DELETE: Borra el recurso especificado

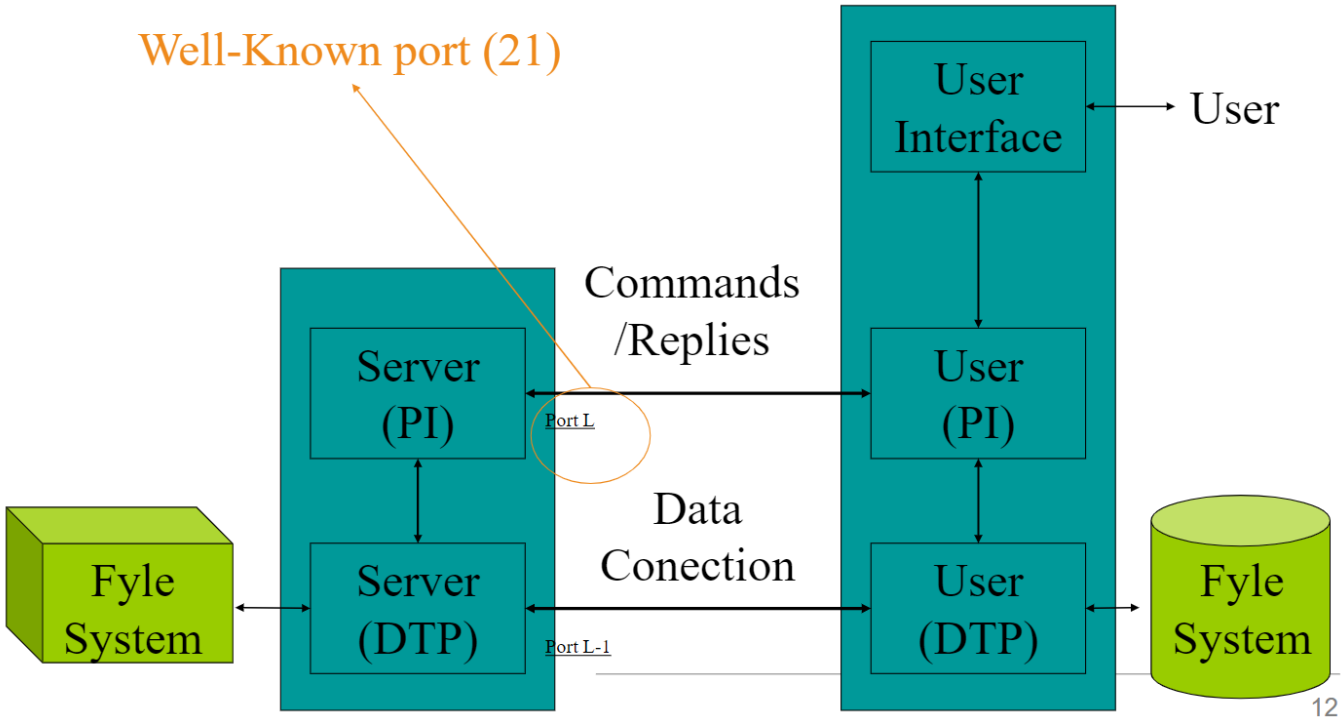
FTP: File Transfer Protocol

FTP tiene la siguiente serie de objetivos:

- Permite compartir ficheros binarios o de datos
- Facilita el uso de ordenadores remotos
- Protege al usuario de las variaciones en el sistema de ficheros de los diferentes hosts
- Transfiere ficheros real y eficientemente

Esta diseñado para ser usado por programas

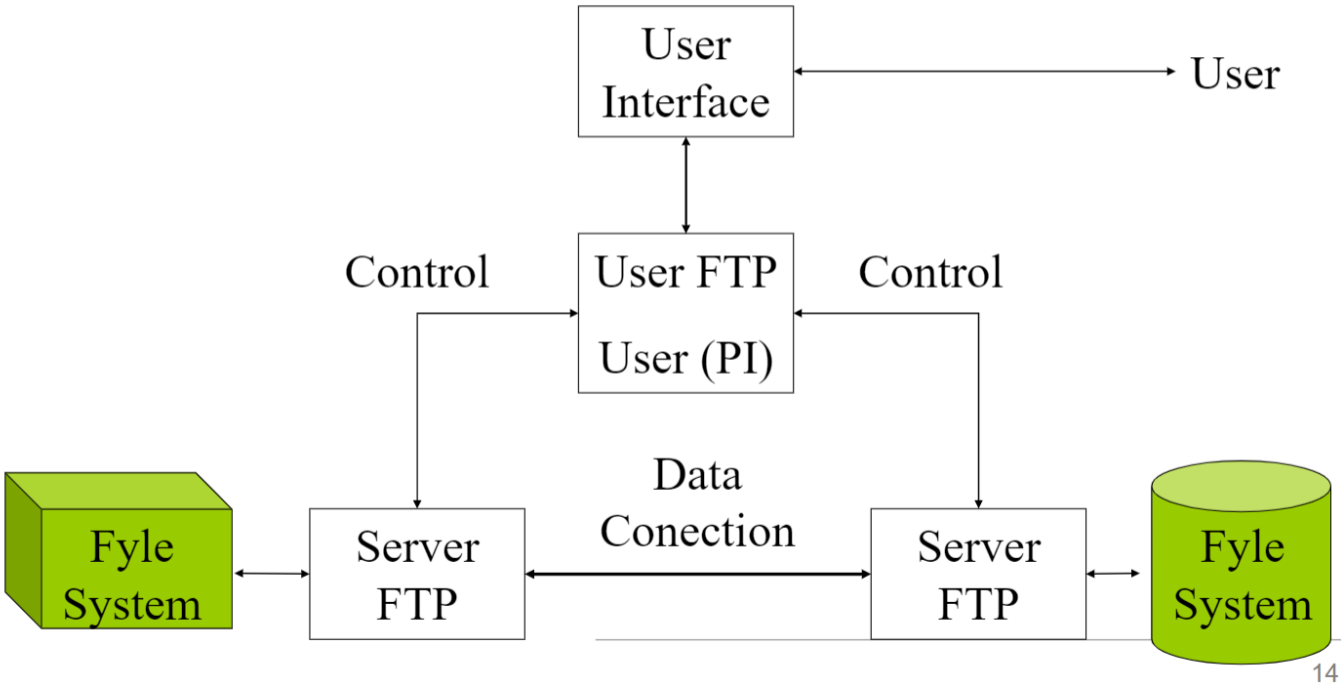
Esquema 1 de FTP



Enviamos al servidor con 2 canales.

- PI: Protocol Interpreter
- DTP: Data Transfer Protocol
- UserInterface: Programa o interfaz gráfica usada, por ejemplo, filezilla

Esquema 2 de FTP



Nos conectamos con un servidor, luego con otro, se mandan comandos y la transferencia de datos se hace entre los 2 servidores. Se establece un canal de datos entre 2 entidades y se realiza la transferencia. Transferencia de usuarios entre 2 servidores.

Ejemplo de funcionamiento Esquema 2

- El Cliente C se conecta con los servidores A y B.
- C le dice a A que se ponga en modo Pasivo.
- A responde diciendo que entra en modo pasivo indicando la IP y el Puerto
- C le dice a B que se conecte a las IP y puertos de B
- B le envía confirmación a C
- Comienza la transferencia de datos

Representación y almacenamiento

From:
https://www.knoppia.net/ - **Knoppia**

Permanent link:
https://www.knoppia.net/doku.php?id=dad:protocolos_comunicacion&rev=1697711072

Last update: **2023/10/19 10:24**