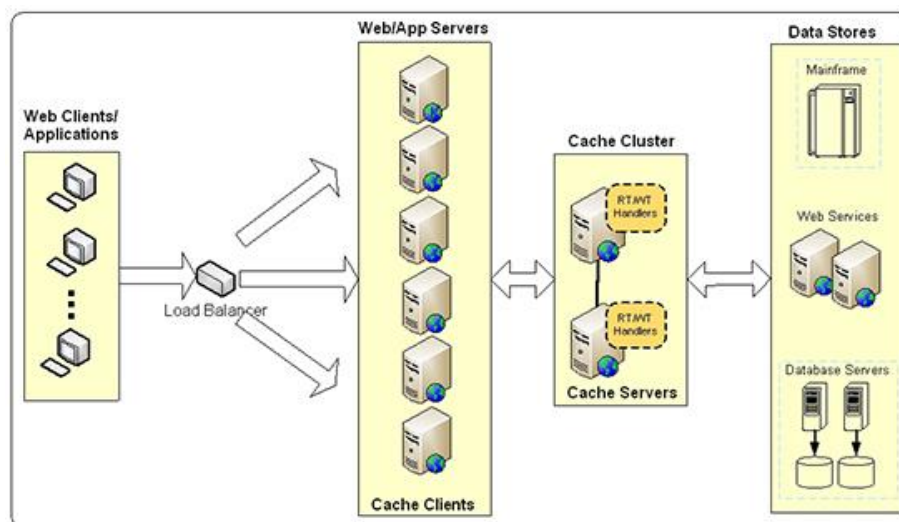


## Crie exemplos de aplicabilidades para os modelos

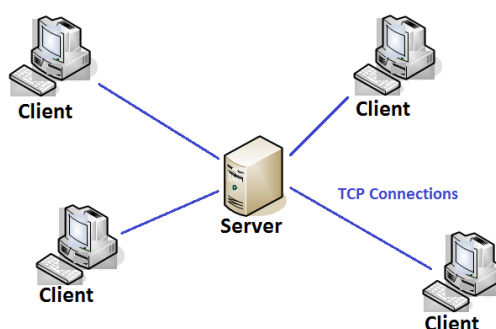
### 1. Servidor Proxy-Cache

Um proxy de cache HTTP ou, caching proxy, permite por exemplo que o cliente requisiute um documento na World Wide Web e o proxy procura pelo documento na sua caixa (cache). Se encontrado, a requisição é atendida e o documento é retornado imediatamente. Caso contrário, o proxy busca o documento no servidor remoto, entrega-o ao cliente e salva uma cópia em seu cache. Isto permite uma diminuição na latência, já que o servidor proxy, e não o servidor original, é requisitado, proporcionando ainda uma redução do uso da banda. Desta forma, qualquer usuário que acessa a internet pode se beneficiar de um servidor proxy-cache, especialmente em momento de maior tráfego na rede em que diversos clientes solicitam dados iguais, como imagens, de um mesmo servidor destino.



### 2. Cliente-servidor padrão

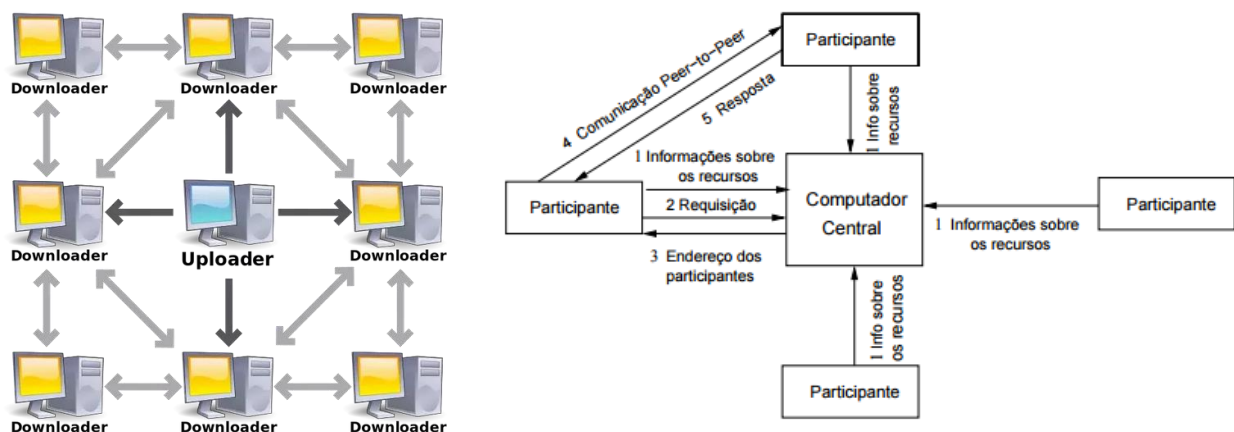
Uma rede de computadores de clientes comunicando com um servidor através da Internet. Tanto os clientes como o servidor são os pontos de comunicação (nós) na rede. A organização dos nós de uma rede designa-se por topologia de rede.



### 3. Peer-to-peer

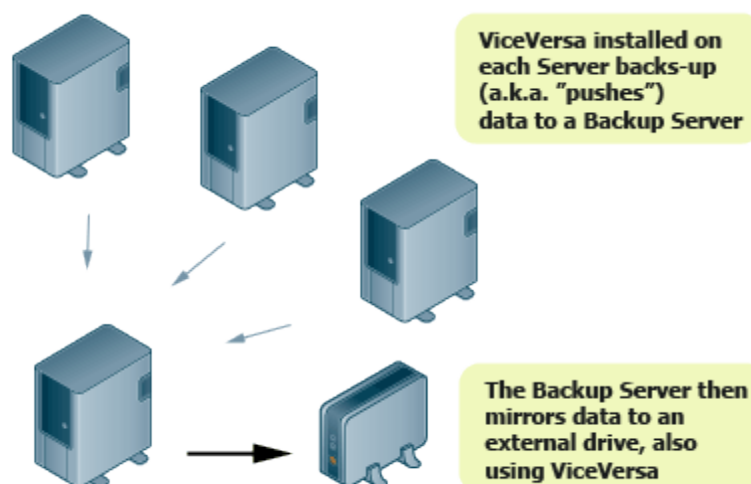
Em um modelo peer-to-peer não existe o conceito cliente servidor, pois em todas as máquinas possuem a informação. Imagine um sistema de catálogo de música onde ele só sabe o endereço de onde estão as músicas, e elas estão distribuídas em diversas máquinas (dos próprios usuários do sistema).

Um exemplo é o Napster onde ele é uma máquina de busca dedicada a localizar arquivos de música MP3. Um servidor central é utilizado para armazenar uma lista com as músicas disponibilizadas pelos usuários e onde elas estão localizadas. O programa cliente Napster, instalado no computador dos usuários, faz uma consulta ao servidor Napster para obter informações sobre o arquivo desejado. O servidor Napster responde se existe o arquivo desejado e onde ele está localizado. Caso exista, uma conexão direta é estabelecida com o computador onde o arquivo está armazenado para que seja efetuado o download.



### 4. Servidor N Servidores

Uma arquitetura servidor para n servidores pode ser utilizada numa situação de backup em Data Center, na qual diversos servidores se comunicam com um servidor central, um *storage* server, com objetivo de armazenar todos os dados.

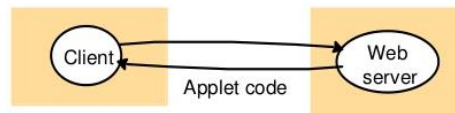


## 5. Código Móvel

Será um código remotamente executado, ou seja, executa no cliente, podendo ser um exemplo um JAVA Applet. Este nada mais é do que uma aplicação em Java (similar ao Java Desktop) que pode ser executado no browser juntamente com sua aplicação web, reduzindo assim a carga do servidor.

### Mobile Code - Web applets

a) client request results in the downloading of applet code



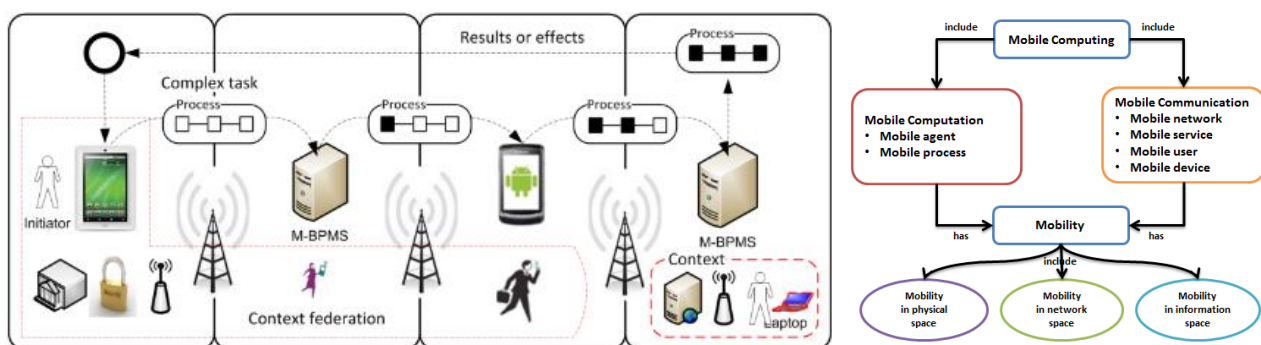
b) client interacts with the applet



- Good interactive response

## 6. Agentes Móveis

Agentes móveis são programas tipicamente escritos em uma linguagem script, que podem ser disparadas de um computador cliente e transportados para um computador remoto para execução, Exemplo disto é o Java que monitora a máquina para avisar que tem uma atualização ou o waze que coleta informações de localização e tráfego através dos smartphones.



É possível criar um sistema P2P se baseado em um modelo cliente servidor?

- Se sim, como eu poderia fazer?
- Se não, por quê? O que impediria de fazer?

Somente de forma híbrida, pois a base do sistema cliente servidor é a centralização da informação, indo de encontro a arquitetura do sistema peer-to-peer. Assim, determinados servidores poderiam ser utilizados como “proxy-cache” para otimizar o desempenho de uma rede peer-to-peer que esteja solicitando demasiadamente determinados arquivos, o que iria ampliar a escalabilidade caso estes servidores tenham robusta capacidade de retransmitir estes dados. Estes servidores agiriam como catalizadores nesta rede.