Web Based one-to-many Data Distribution for Retrieving Continuous Stream or Bulk Data using Rate-Controlled Multicast Transfer

Logical Configuration

In the configuration of figure 1, a single server transmit data to a number of simultaneous receivers with different connectivity. In figure 1 we have placed a receiver in the same LAN as the sender, another receiver connected to the server through an Internet path and a third receiver connected by means of a low-bandwidth point-to-point link.

The users of this system request their data using a web browser, that perform http transactions which end up in scheduling the actual data transmission at a given time. The web browser is also used to receive the data via a multicast-capable Proxy server.

The actual data is transmitted using multicast in order to save bandwidth and server processing power, exploiting receivers aggregation.

The difference in the available bandwidth affect receivers in two ways, depending on the data type: if a continuous stream is being transmitted, a lower bandwidth results in a worse data quality (we can not have reliable transfer); if bulk data is being transmitted, reliability is possible —often it is a requirement— and lower bandwidth results in a larger transfer time.

Physical Configuration

The logical configuration shown above can be mapped in the following physical setup, see figure 2.

In this configuration all the browser are run on the same machine but they still have a different connectivity because they use different proxies. Proxies are run on distinct machines on which a multicast router is run as well. Each multicast router, receiving multicast data via a separate paths, has the function of feeding the local proxy. The first router is directly connected to the server by means of a LAN and feeds the other routers via multicast tunnels. The Internet path is obtained using a remote host as a reflector. This is obtained by running either a multicast router or an IP tunnel on the remote host.

To set up the configuration of figure 2 we need at least 5 machines, 4 of which should be multicast-capable. All the machine, exuding the one where the browsers are run, should have:

- **OS** *FreeBSD* or *Solaris* (if Solaris they should have Multicast-3.5+ installed, if FreeBSD version 2.0 or later is preferred).
- **Multicast-capable network interface**.

One of the machine above should have two network interfaces.

The machine where the browser is run can have any OS, but it should be quite fast and have good graphical capability.

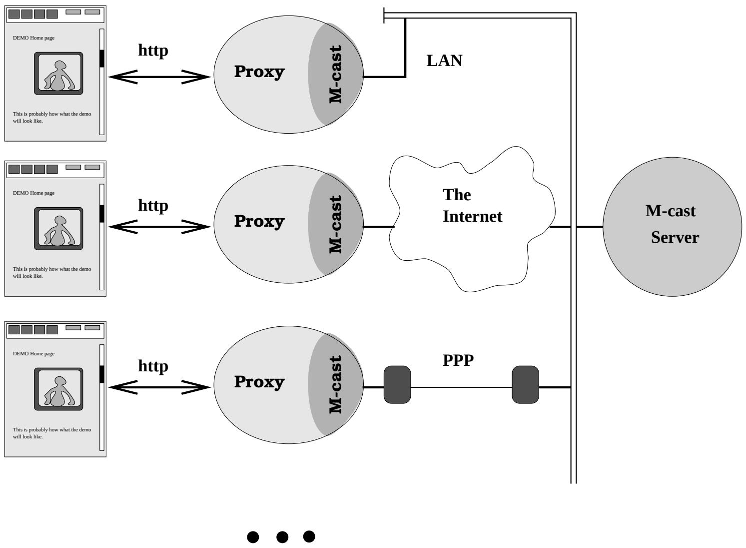

Figure 1:

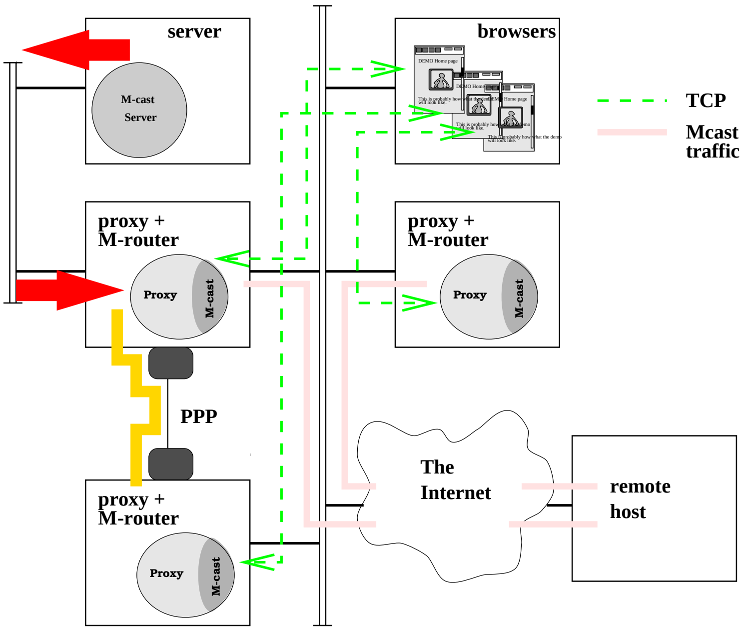

Figure 2:

Moreover we need *Internet connection*, the local hosts should have *valid Internet addresses* and we need the devices to set up at least one point to point connection. They either can be, *analogical modems*, *ISDN modems*, *GSM modems.* If none of them can be provided we can use a *null-modem cable* to implement the point-to-point connection.

Other low-bandwidth connections can also be used, such as an ATM VC.