The way it's currently implemented, the files are downloaded separately instead of as a single .mix. Downloaded files are stored in the TT filesystem (TTFS) with some metadata to determine which files belong to which packages (the .tpi files in the TTFS). Therefore, having a .mix with the same files is not identical to having the files and metadata in the TTFS. There are various reasons why it was implemented this way:

- .mix archives often contain the same files. For example, almost every map has the same version of scripts.dll (or one of the few versions of custom scripts.dll). I found that for the ubermappack, iirc, about 85% of the space was wasted by duplication. In the TTFS, there are no duplicate files, which makes it smaller and reduces the amount of data to be downloaded for new packages.
- Multiple variants of the same map (with the same name) could not be stored and referenced properly using .mix files. This could for example let one server install a cheat-modified map on your client, and then your client would use it on other servers.
- There were several problems loading and unloading .mix archives dynamically, so we either had to fix .mix packages or come up with something new (TTFS).

Note that, initially, we wanted to store downloaded files in a virtual filesystem for efficient access, and we wanted to make stand-alone packages that would be easy to download and install. In fact, we wanted to provide some kind of "store" in the launcher, where you would be able to list and download maps.

As for lag caused by downloading: it will not really decrease the server to client data stream, but it can make it less reliable (packets not arriving, causing latency) if your connection is exhausted. The download should theoretically automatically throttle back in that case, although I have never carefully studied this.