

Codecs for audio, or video, sharing have a limited size, compared to the size of the shared file.

There are many standard, which vary over time, and each reader (hardware or software) must be crafted for the current standard, and they easily become obsolete.

The same problem is present in the dissemination of scientific data, where there is no standard for files containing scientific data: it is difficult to extract data from old scientific files, unless you have the necessary documentation for extraction, such as files or a protocol.

This common problem could be solved by inserting a header in the audio, video or scientific files. A source file (for example C language), could be included in the initial part of the file, then the reader (hardware or software), could read the header, compile the source (the C language work on almost all the devices), and it could run the decoding program on each device, in each time, getting the decoding process.

In the scientific literature it would no longer be necessary to share the encoding of the file, in addition to obtaining data preservation for future memory, at the cost of a small increase in memory used.

Some tricks could be used to reduce memory usage, such as to use all the possible include directive (so that the include directive can be neglected), to use main function without arguments (so that the main function can be neglected), to use compact source (reducing the number of characters reducing the legibility of the code, for example using a single letter variables, or deleting the comments).

I understand that there may be security problems, in the sense that an uncertified executable could be used to infect a computer, but if the environment of use was limited to some areas of safe memory, there should be no problems: I explain, if an area sure it was accessible to every computer, associating it at the beginning of the execution, where the executable can write the data, and where the operating system can access only for reading, then there should be no infection problem.

Open science data should use methods like this, or alternatives, to simplify future use for different operating systems, to make data accessible in the next centuries in the simplest way possible