FFmpeg // STF 2024
Project report

1a) How has this contract impacted the Open Source component you are working on?

We want to understand better the overall impact the investment had on the technology. This is less about the activities themselves, but more about the overall outcomes of the contract.

The investment has demonstrated, in a public and open fashion, that meaningful and concrete contributions to the maintenance of critical open source infrastructure is valued, and compensated. This is important since it dispels a common trope that maintenance of open source projects is thankless.

1b) What would you like to share with the public about the progress you made during the contracted period?

We would like to share and talk about the work you did and the progress you made, e.g. as an additional paragraph in the project description on our website. Please provide us with a short description, or let us know if we can use the response from 1a.

NOTE: The response from 1a can be used as well.

FFmpeg is software for decoding, encoding, transcoding, multiplexing, demultiplexing, streaming, filtering, and playback of a wide range of multimedia formats, from ancient, obscure standards to cutting-edge ones. It is a cornerstone in the multimedia processing landscape.

The project helped the community fix outstanding bugs identified through static analysis as well as improve existing components (FFV1 video codec and libswscale module) and refactor old code (libpostproc) to facilitate maintenance going forward.

2) What impact did the financial support have on your team, your organization, and the community?

We would like to understand better how to investment and contract has influenced the people behind the technology. Are there short-term and long-term changes that you recognize as a result of the investment?

It is challenging for individuals to find the time to support community projects pro bono when other projects are paying the bills. In the short-term, the investment therefore allowed

community members to dedicate more of their time to maintaining FFmpeg. In the long run, the investment will hopefully draw more individuals to FFmpeg and other community projects.

3) What challenges did you face while working on this project? Were you able to address them? If so, how? If not, why?

Feedback on this question is optional but incredibly helpful and allows us to understand better the challenges projects might face during the contract period and how we can be potentially helpful in alleviating or preventing risks. Any kind of challenges are helpful for us to know about - personnel, planning, management, technical, etc. and we appreciate your openness.

The FFmpeg community consists of individuals distributed worldwide, with varied personal and professional backgrounds, e.g., employed by large multinationals, self-employed, students, hobbyists, etc. While the community has extensive experience collaborating on software development on that scale, the STF investment was the first time that the community had to collaborate on a funding proposal. This required a new kind of (non-technical) collaboration and the development of new processes, which is ongoing and will make future investments even more productive.

4) What's next for the Open Source component, your team or your organization? What is your long-term vision?

Please help us to understand where your technology and/or team is heading. We are also interested in the larger picture - what emerging developments should we keep an eye out for in the technology's ecosystem?

The vision for FFmpeg remains unchanged: to be the leading open-source multimedia framework, to decode, encode, transcode, mux, demux, stream, filter and play pretty much anything that humans and machines have created – from the most obscure ancient formats up to the cutting edge, no matter if they were designed by some standards committee, the community or a corporation. In addition to supporting a wide range of formats, FFmpeg also strives to be highly portable: it compiles, runs, and passes our testing infrastructure FATE across Linux, Mac OS X, Microsoft Windows, the BSDs, Solaris, etc. under a wide variety of build environments, machine architectures, and configurations.

5) Lastly, we'd like to know if you have any feedback for us? What went well, where could we improve?

As discussed at (3), managing the non-technical aspects of the project could perhaps be improved by providing self-service online tools to create the proposal, track progress, create invoices, etc.