



OPEN-SOURCE INSPIRED WORKFLOW FOR JEM MAINTENANCE

JVET-C0034 | Emmanuel Thomas

TNO innovation
for life

DISCLAIMER

This not about the JEM software per se but merely about the tooling to maintain the software

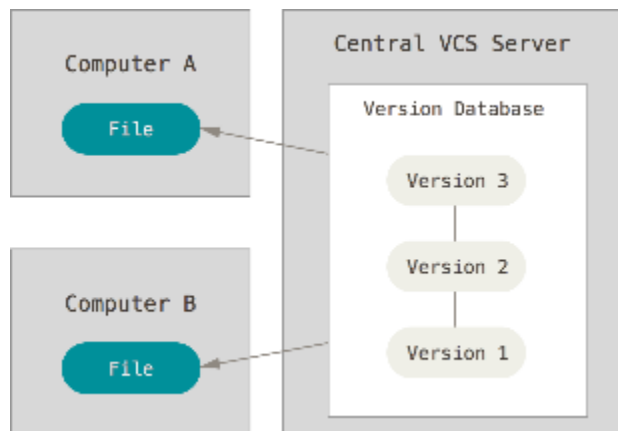
MOTIVATION

- › At JVET #2, the maintenance of the reference software was discussed.
- › The increase of exploratory tracks makes the maintenance of the whole software challenging.
- › However, we believe this is mainly due to the workflow and tooling used at the moment, i.e. a centralised code versioning system.
- › This contribution is proposal to make future development of JEM easier.

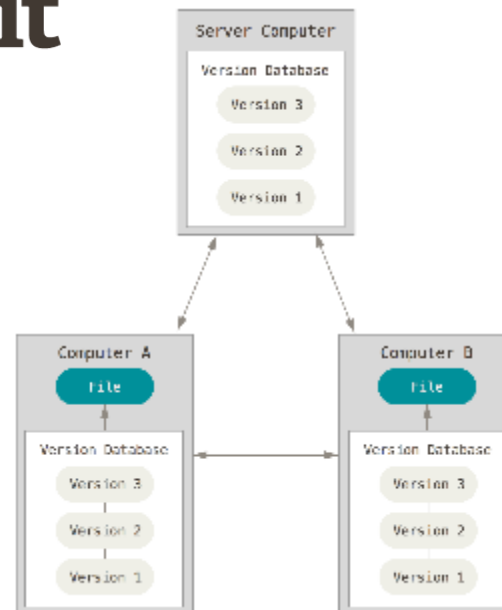
CENTRALISED VS DISTRIBUTED



Centralised



Distributed



THE PROPOSED CHANGES FOR JEM



THE PROPOSED CHANGES FOR JEM

- › This is not changing just for changing.
- › This bring the following benefits:
 - › Every participant may create as many branches necessary in his/her own repository for exploration
 - › When desired (e.g. EE), participants may share their code with others on Gitlab
 - › The main repository stay clean and may not need to contain all the branches of ongoing studies
 - › Less work for the SW managers. Accepting new source code means clicking on a button in the web UI
 - › Automated test (does it compile?) when integrating contribution can be configured
 - › And also the usual bug tracker, milestone tracking (bugs fixed in a new version), wiki, etc...

POSSIBLE WORKFLOW

