

Structure of the ESP-r Source Code Archive Version of May 1, 2006

A version control system will be used to facilitate the management of the ESP-r source code archive. A schematic representation of the version control system's proposed configuration for archiving the source code is given in Figure 1.

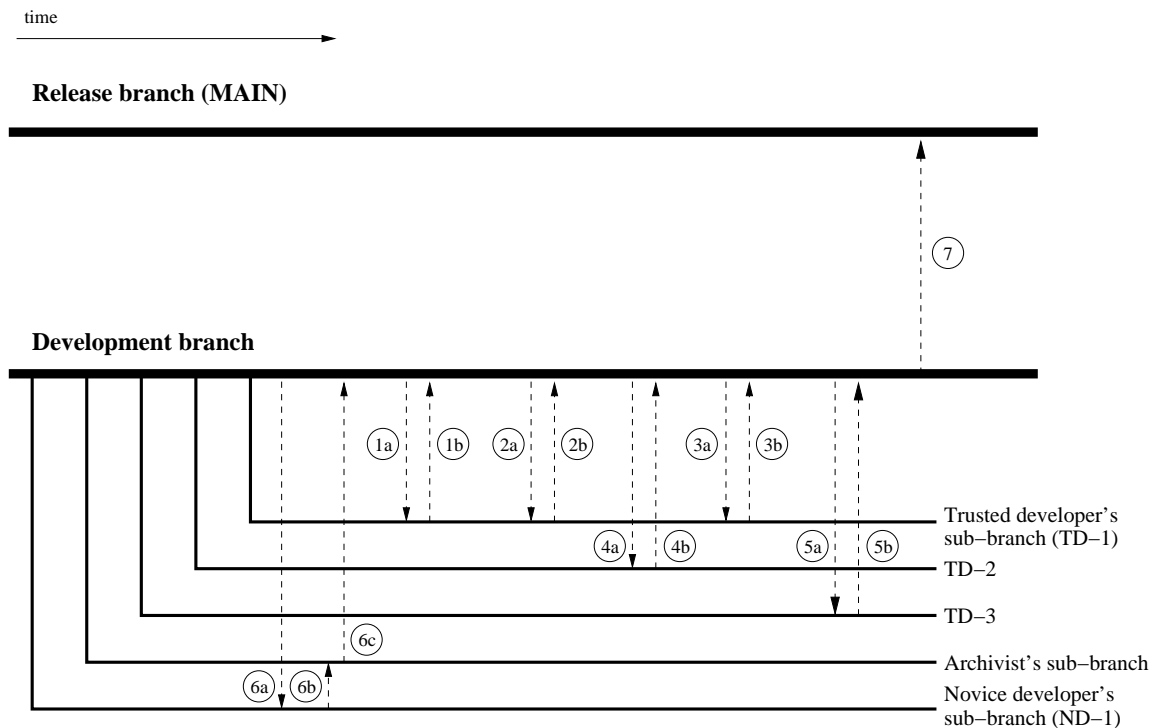


Figure 1: Schematic representation of the ESP-r archive.

The release branch contains the general distribution of ESP-r. Any interested party (both users and developers) can, at any time, download the source code representing the latest, fully quality assured version of ESP-r. The release branch is updated periodically (about every three months) from the development branch following the completion of a quality assurance procedure (represented by '7' in Figure 1). This procedure comprises the simulation of a number of pre-constructed models and the comparison of predictions against archived results corresponding to previous releases. The procedure also entails testing on all supported operating systems. The details of the quality assurance procedure are given in a separate document. The released version is accompanied by a brief summary of the inherent changes.

All ESP-r users will have access to the release branch through anonymous login, while developers who wish to contribute source code will be provided with an account that gains them access to the development branch. Furthermore, developer-specific sub-branches will be created for each developer (e.g. ND-1 and TD-1 in Figure 1). These branches essentially represent a copy of the development branch, which is managed by the version control system. An automatic log of all downloads (and uploads) will be maintained.

The operation of the system is illustrated with the example of the novice developer who 'owns' sub-branch ND-1. Developer ND-1 will perform their development work on their local machine, periodically

committing their new and/or modified source code to sub-branch ND-1 in the version control system. These periodic code commitments have no impact upon other users/developers as they are strictly available only on the ND-1 sub-branch. Once developer ND-1 feels confident that their code is ready for sharing with the entire ESP-r community, they will perform a number of steps prior to contacting the Archivist.

Firstly, developer ND-1 will synchronise sub-branch ND-1 with the development branch. This is represented by '6a' in Figure 1. The version control system will facilitate this operation although some human intervention on the part of developer ND-1 will be required. (The prudent developer will synchronise frequently throughout their development work to avoid coding conflicts with colleagues.) Developer ND-1 will then ensure that their coding adheres to the ESP-r coding guide (see separate document) and that the necessary testing has been performed (as outlined in the separate quality assurance guide). Finally, a brief description outlining the system enhancements or alterations is drafted and sent to the Archivist.

Following a dialogue between the Archivist (assisted by a technical support team) and developer ND-1, the Archivist will merge developer ND-1's code into the Archivist's sub-branch. This is represented by '6b' in Figure 1. Following an appropriate degree of scrutiny, verification and testing, the Archivist will transfer developer ND-1's code to the development branch and thus make the code available to all ESP-r developers. This is represented by '6c' in Figure 1. A brief summary of the enhancements or alterations will be documented within the version control system.

A more streamlined procedure will be applied for developers who have a demonstrated history of safe code contributions and adherence to the coding guide. This is illustrated with the example of the trusted developer who owns sub-branch TD-1. The process outlined above for developer ND-1 is followed up to the point at which the Archivist merges the developer's code into the Archivist's sub-branch. Rather than performing this operation, the Archivist will engage in a dialogue with the developer and will scrutinise and verify (but not test) the code that resides on sub-branch TD-1.

Once the Archivist is satisfied with the coding contribution, the developer will be authorised to transfer the code to the development branch, accompanied by a brief summary of the alterations. Figure 1. illustrates two instances, '1a' and '1b' in which developer TD-1 first synchronises sub-branch TD-1 and then transfers their code contribution to the development branch.

An identical procedure is applied for other trusted developers, as represented by TD-2 and TD-3 in Figure 1. It is expected that the number of trusted developers will be small. For example, a given developer organisation may wish to structure its activities such that all code is filtered through one trusted individual who scrutinises the code before contacting the Archivist.