## deRSE25 and SE25 Timetables



Contribution ID: 149

Type: Poster

## Agile and optimized methodology for VENQS - an automated, cross-platform, modules versioning management system for RSE

Wednesday 26 February 2025 19:40 (20 minutes)

Code development and maintenance in a team can be a daunting process especially when multiple modules are interconnected with variegated dependencies, dispersed over a few git repositories and/or developed in different versions of the software. Consequently, VENQS is established to set up an infrastructure and workflow for a semi-automated version and dependency management. This is achieved with a desktop application, with a simple GUI and a repository system in Gitlab, that allows the build and download of a package of modules selected by the user. Here each module is inter-compatible and has no dependency issues. This enables users to share their configuration and initialize a software project for new members in a drastically shorter time.

Erroneous pull requests from a wrong branch or the misplacement of modules in the directory are a few common problems in research software development. All resulting in unexpected behavior, like unreasonable results, code not working or behaving differently on other systems. VENQS is developed to solve such problems, which increase with the projects size, operating system changes and software updates. Additionally, users might become contributors by creating their own modules, which need to be incorporated. Therefore, the storage of information about module dependencies and compatibilities becomes significant, requiring a versioning methodology. Subsequently, making standardization for a modules'structure and documentation essential. Such issues can partially be solved with git repositories by using tags and maintaining a list of compatible module versions. However, this will not resolve the tedious and error prone task of setting up a project. In our use case, the modules are developed in Simulink, a MATLAB-based graphical programming environment, to provide a library for the simulation of the orbit propagation of satellites. VENQS is developed with other languages in mind like C++ and C which are also applicable in Simulink as S-functions.

Standardization of modules makes metadata easily accessible for VENQS, allowing for automation of tests and source code compilation. Furthermore, generating the advantage of saving legacy code in a well-documented and structured environment allowing for smoother development in the future.

Versioning is managed for modules and the packages of modules. While modules get stable releases after a development cycle, packages are versioned to handle compatibility inter-modules and between two packages from differing repositories. Each package contains metadata regarding software version and operating system, which is used in VENQS to display available packages for a given system. This enables selection of intercompatible modules for users, while VENQS handles all necessary dependency selections automatically using the metadata. Thereafter, VENQS loads and constructs the necessary folder structure and module setups on a given system and leaves a YAML file containing information on the module configuration, which is sharable, allowing set-up in another system by only providing the YAML file.

## I want to participate in the youngRSE prize

yes

Primary author: PFANNENSTIEL, Mike (DLR - Institute for Satellite Geodesy and Inertial Sensing)

**Co-authors:** Dr BREMER, Stefanie (DLR - Institute for Satellite Geodesy and Inertial Sensing); CHAND, Suditi (DLR - Institute for Satellite Geodesy and Inertial Sensing)

Presenter: PFANNENSTIEL, Mike (DLR - Institute for Satellite Geodesy and Inertial Sensing)

Session Classification: Poster and Demo Session together with Reception

Track Classification: Research Software: research software usability