Vantage6 software management plan

Bart van Beusekom, Frank Martin, Leonard Wee & Gijs Geleijnse

Date	Authors	Description	version
05-06-2023	BB, FM, GG	Initial version	0.1
16-08-2023	All authors	First published version	1.0

Contents

Introduction

This document describes a software management plan for the vantage6 software for federated learning following the guide created by the eScience centre [1]. The goal of a software management plan is to ensure software usability and maintainability in the longer term.

The guide to software development plans [1] describes three levels of software management: low, medium, and high. Based on their criteria, **we have classified the vantage6 software into the high management level class**: the software is a vital component of at least a dozen research projects and cannot be rewritten during a project's lifetime. Therefore, it requires a higher management level than 'medium', which is usually aimed at software that is used as a component of one or a few research projects.

Below, we will describe all steps listed in the guide to software management plans [1]. First, the vantage6 software will be described. Then, we discuss how our way of working helps to ensure maintainability of vantage6, as well as how this may be further improved.

Software description

Vantage6 is an implementation of the Personal Health Train concept [2] to enhance individual-level privacy during the process of federated learning [3]. Its purpose is to provide researchers with an infrastructure framework that allows them to analyse combined datasets without transferring the datasets either between institutions or from institutions to a researcher. Federated learning was designed to address some specific legal and governance concerns in collaborations conducting research on sensitive data from multiple sources.

In a typical vantage6 research project, each organization provides a vantage6 'data station' (also known as a 'node'). Then, vantage6 algorithms are executed on the data station to obtain research results. These algorithms only share aggregated, non-identifiable data. By using vantage6, the organizations can exchange insights and compile global research results without the need to centralize all data sources in a single location.

Vantage6 thus allows researchers to run federated learning analyses as purely local computations on the site of the data owner, and then to share only the summarized results of the learning with others. Vantage6 also supports other protocols such as multi-party computation (MPC) [4]. What these scenarios have in common is that several parties with data agree to learn from combining their data, but they are not allowed or not willing to do so at individual subject record level.

The vantage6 infrastructure consists of several components (see Figure 1). These components are all software packages that may be distributed separately. Briefly, the vantage6 infrastructure at present contains the following components:

- The central server. This application is responsible for managing users, organizations, collaborations, etc. It handles authentication and stores results of analyses.
- The data stations (typically 2 or more in a collaboration). They check if new tasks are available for it from the central server, and if so, it executes them and returns the results.
- The command line interface (CLI). This software package allows the administrators of servers and data stations to easily start or stop the server or data station, or to create new instances of them.

- The user interface. Via the graphical user interface, users can easily communicate with the central server. For instance, they can easily create new tasks and download the results when the task is done.
- The Python client. This application provides users with a Python interface that facilitates communication with the central server API.

The vantage6 algorithms are **not** part of the infrastructure and therefore not discussed in this document which is only about the infrastructure; vantage6 algorithms should ideally have their own, separate software management plan. When a new research project is started that uses vantage6, the research project should allocate resources to create the required algorithms. Note that some of these algorithms may already be available from other research projects.

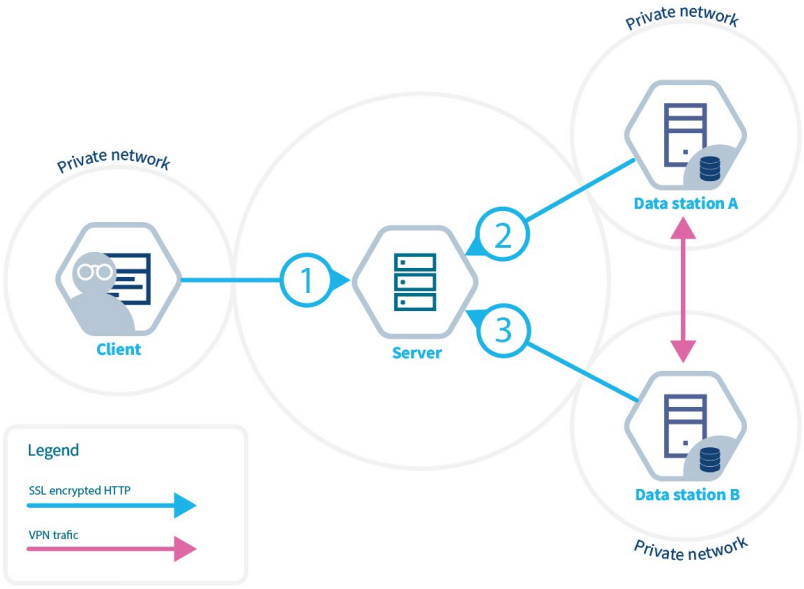

Figure 1 – High-level overview of the vantage6 infrastructure. The VPN connection is an optional feature of the infrastructure and is only required for certain algorithms. The client and data stations connect with the server; the node collects information from the server.

For more information on vantage6, see the project website (https://vantage6.ai/) and the documentation pages (https://docs.vantage6.ai/).

Audience

The vantage6 infrastructure was originally developed for epidemiological and machine learning studies in the field of oncology, but it may be generalized to many other data-driven research areas as well. In principle, the infrastructure may be of interest to any group of organizations that wishes to do analyses on federated data but do not wish to move data at the individual subject-level outside of their own organization. Vantage6 supports both horizontally-partitioned data (where different parties have the same data attributes on different individuals/entities/…) and vertically-partitioned data (where different parties have different data attributes on the same individuals/entities/…).

It is important to note that some use cases are presently more “mature” than others in vantage6. For instance, projects requiring high-performance computing (such as deep learning neural networks) are generally not yet as well supported as epidemiological and simple machine learning projects (which usually require descriptive statistics and/or regression models) due to the legacy of vantage6. However, vantage6 is an open-source project that encourages contributions to mitigate any shortcomings, including those outside of its original area of focus.

A person that interacts with vantage6 usually has one or more of four main roles: (1) they are simply users that log in with a vantage6 user account ; (2) they are responsible for administration of a data station; (3) they maintain a central server; and/or (4) they develop vantage6 algorithms. We will refer back to these roles where appropriate.

Versioning

Vantage6 uses semantic versioning format [5]. This means each version is formatted as `Major.Minor.Patch.Pre[N].Post<n>`. Major version increments introduce breaking changes, i.e. changes that are not compatible with previous versions, and are not compatible with one another. We aim that algorithms may be run on all compatible minor versions, e.g. if an algorithm runs on v2.1.0, it should also run on v2.7.3. Also, the central server API should return the same fields in responses to users (new fields may be added in new minor versions). However, servers and nodes of different minor versions are *not* always compatible: a node on v2.2 is not guaranteed to function properly with a v2.3 server. This choice was made because it is difficult to achieve this at present with available resources and test setup. Nodes and servers may be expected to function properly together between patch versions, e.g. a v1.2.3 node should work with a v1.2.7 server.

Minor versions are released for new features, enhancements and other changes, and patches are released for bugfixes. ‘Pre’ tags only apply to release candidates and ‘post’ tags are releases where the code has not changed.

For more details, see https://docs.vantage6.ai/en/main/devops/release.html#version-format.

Finally, it should be noted that all components of the vantage6 infrastructure are released at the same time, with the same version. This facilitates working with compatible versions.

Software availability

All components of the vantage6 infrastructure are open source and available on GitHub in the organization ‘vantage6’: https://github.com/vantage6. The repository that contains most of the infrastructure code is in https://github.com/vantage6/vantage6 and the user interface is available at https://github.com/vantage6/vantage6-UI. Future extensions of the infrastructure will also be made available in the vantage6 organization on GitHub.

Documentation

The documentation is made available at https://docs.vantage6.ai/. The documentation is part of the main infrastructure repository, as is the code to generate it. The documentation contains separate sections for each of the main roles that were identified in the Audience section.

For more technical details on the documentation, see https://docs.vantage6.ai/en/main/devops/documentation.html#how-this-documentation-is-created.

Contribution guidelines

Guidelines on contributing to vantage6 are available in the 'Contribute' section of the documentation (https://docs.vantage6.ai/en/main/devops/contribute.html). This section lists how to report issues or security vulnerabilities, guidelines for pull requests to resolve issues or add new features, and how one may participate in community meetings.

We also enforce these guidelines by executing automated checks on each pull request. These checks include unit tests, unit test coverage, and a static code analysis whether the code conforms to the widely used PEP8 style guide for Python code. Pull requests that do not conform to the quality standards will not be merged; instead, developers can view what they have to improve and the pull request will be merged after they have made these improvements.

License

Vantage6 is released under the Apache License 2.0. The SPDX license identifier is Apache-2.0. We do not foresee any changes to the license in the near future.

Installation requirements

The installation requirements of vantage6 depend on which components of the infrastructure you are installing. The documentation contains separate sections for each of the main roles that were identified in the Audience section, and these sections include installation requirements for that role. For algorithm developers, there are no hard requirements to install anything, but for those developing algorithms in Python, installing the Python client is recommended so they may use the specific Python algorithm client in their algorithms. Note that in version 4.0, which is planned for September 2023, the tools for developing Python algorithms will no longer be in the vantage6-client package, but rather in a separate package called vantage6-algorithm-tools.

The links to the installation requirements for each component are provided in Table 1. Note that the user interface is one of the optional central server components, and that the CLI is itself a requirement for running a data station or central server.

Software component	**Link to installation instructions**
Python client	https://docs.vantage6.ai/en/main/user/pyclient.html#requirements.
Data station	https://docs.vantage6.ai/en/main/node/requirements.html
Central server	https://docs.vantage6.ai/en/main/server/requirements.html.
Optional central server components	https://docs.vantage6.ai/en/main/server/optional.html

Table 1: Installation requirements per vantage6 infrastructure component

Citing the software

Two academic papers [6, 7] have been written about the vantage6 infrastructure:

1. Moncada-Torres, A., Martin, F., Sieswerda, M., Van Soest, J., & Geleijnse, G. (2020). VANTAGE6: an open source priVAcy preserviNg federaTed leArninG infrastructurE for Secure Insight eXchange. In *AMIA Annual Symposium Proceedings* (Vol. 2020, p. 870). American Medical Informatics Association.
2. Smits, D., van Beusekom, B., Martin, F., Veen, L., Geleijnse, G. & Moncada-Torres, A. (2022). An Improved Infrastructure for Privacy-Preserving Analysis of Patient Data. *Advances in Informatics, Management and Technology in Healthcare*, *295*, 144.

The first paper is a description of the main components of the infrastructure. The second paper concerns the extension where algorithms on different data stations can communicate over a VPN network, which was introduced in version 3.0.0.

Users of the software can cite any or both of these works. In the view of the maintainers, the first paper should always be cited as it contains the most generic overview of the infrastructure. When using the VPN communication feature, one should also cite the second paper.

Additionally, researchers may also cite the code itself, and which version they used. This is available on https://doi.org/10.5281/zenodo.7382602.

An up-to-date overview of citations is also maintained on our website at https://vantage6.ai/vantage6/references/.

Development team

The vantage6 infrastructure has so far been developed most actively at IKNL (Comprehensive Cancer Organization Netherlands), with contributions from several others, most notably the eScience centre and Maastricht University. Contributions from other institutions are at present mostly limited to code; administration, such as planning, testing and releasing the software is executed by IKNL employees at present. In the future, we foresee that this responsibility may be shared more with other institutions.

When referring to 'development team' in the following, it depends on the context whether this includes IKNL personnel or also personnel from other institutions. For example, at present IKNL personnel is responsible for new releases, so when mentioning the development team in the release process, this refers to IKNL personnel (though in the future, this may be subject to change). For other processes, non-IKNL personnel may be also be counted as part of the development team.

When new developers join the vantage6 development team, we will ensure that they become familiar with the procedures relevant to them, so that maintenance is kept at the same level.

Documentation for future developers

All functions in the infrastructure contain docstrings according to the Numpydoc format [8]. Newly added code should also adhere to these standards (see Contribution guidelines). From these docstrings, documentation of function and classes is generated in the technical section of our documentation (https://docs.vantage6.ai/en/main/technical-documentation/index.html). This provides developers with an overview of the infrastructure code.

The user interface, which in contrast to the other infrastructure components, is not written in Python but with the Angular framework, at present lacks this form of documentation. The following GitHub issue was opened to track mitigation of this shortcoming: https://github.com/vantage6/vantage6-UI/issues/116. Mitigating this issue is simple – it only requires developers to add comments in a certain format – but requires time investment. The development team aims to address this issue for existing code in the time allotted to quality improvement (see Software quality improvement). Also, the development team will start including this form of documentation in new code changes from now on.

Software testing

Unit tests are part of the vantage6 infrastructure repository, but they do not cover the complete codebase. The CLI and central server have rather good coverage, and these tests are run automatically upon every pull request using a GitHub Action pipeline. The results of automated unit tests are public for each pull request.

In contrast, unit tests have not yet been developed for the data station and user interface code. The Python client at this moment contains unit tests under development that are not executed automatically when code changes are made.

Integration tests are executed manually before a new release of the software. One of these tests is a so-called 'feature tester' algorithm that checks if features are behaving as expected in a release candidate. By running this algorithm, we cover most basic functionality of the vantage6 infrastructure. Additionally, there is an algorithm 'v6-node-to-node-diagnostics' that is usually run to check if the VPN communication is working properly. Finally, the development team tests each of the new features, changes and bug fixes that are changed in the new release. This procedure is documented at https://docs.vantage6.ai/en/main/devops/release.html#testing-a-release.

Finally, it is important to note that input from the community is much appreciated. We acknowledge that the testing of the infrastructure is not as complete as we would like it to be. The development team aims to fully cover the codebase with unit tests, and automate and expand integration testing, but is limited by time constraints. We welcome community contributions both to extend automated testing and to help with manual testing.

Licenses of dependencies

The development team has checked that the licenses of the used dependencies are respected at this time (for more details, see https://github.com/vantage6/vantage6/issues/699). Most dependencies that are used in the infrastructure are well-known open-source libraries for Python or Angular, and their policies are unlikely to change. The risk of violating the licenses of dependencies is therefore small and hence, there are no further automated policies to prevent this at present.

The development team is aware that licenses may impose restrictions and are in the habit of checking licenses before adding a new dependency. In June 2023, a manual check was done to ensure no licenses are violated, and no new dependencies have been added since that time. At present there are no automated checks yet to enforce this practice in the future. A Github issue was created to mitigate this and updates may be found there: https://github.com/vantage6/vantage6/issues/699.

Software packaging and distribution

The Python packages from the vantage6 infrastructure are available from the Python Package Index (PyPI):

- The Python client (https://pypi.org/project/vantage6-client/) – for users to interact with the central server
- The command line interface (https://pypi.org/project/vantage6/) – for data station administrators and server administrators to interact with the data station and server, respectively.

- The central server (https://pypi.org/project/vantage6-server/) – this package is installed within the server docker image (see below).
- The data station (https://pypi.org/project/vantage6-node/) – this package is installed within the data station docker image (see below).
- Library of common functions (https://pypi.org/project/vantage6-common/) – this package is installed automatically as dependency of the other packages.

The User Interface is an Angular application that is not available as a package. It is simply distributed via its Github repository (https://github.com/vantage6/vantage6-UI).

Apart from the distribution of packages and code, several vantage6 components are usually run in docker containers. The docker images for each of these components are available a from public docker registry:

- The central server (https://harbor2.vantage6.ai/infrastructure/server)
- The data station (https://harbor2.vantage6.ai/infrastructure/node)
- The user interface (https://harbor2.vantage6.ai/infrastructure/ui)

The central server and data station images can be run using the CLI; instructions on how the user interface may be run can be found in the documentation and in its GitHub repository. Note that the docker registry at https://harbor2.vantage6.ai does not *just* host the server, data station and user interface images. It also hosts other Docker images that help to run optional features (e.g. there is a VPN client image), as well as many algorithm images.

User support

Support is available by reaching out on a Discord channel (https://discord.gg/yAyFf6Y) for questions. Of course, users with issues can also submit these to the relevant GitHub issue tracker. The development team is committed to helping the users as best they can.

That said, we cannot guarantee users that they will be supported to their satisfaction. There are time constraints that would not enable developers to provide user support indefinitely, especially if the community grows rapidly. The development team therefore tries to encourage (experienced) users of the infrastructure to help other users.

There are commercial efforts to host vantage6 and provide support for specific research projects and users. One of these is Medical Data Works (www.medicaldataworks.nl) and there may be more in the future. Note that IKNL does and will not support or prefer any commercial partner.

Long term maintenance

Up to now, the bulk of the maintenance has been performed by employees of IKNL, the organization from where vantage6 originated. Since 2018, IKNL has been supporting various projects in Europe and internationally depending on vantage6, with consistent software support of 1 to 2 FTE. IKNL as an organization cannot guarantee a long-term commitment to maintaining vantage6.

The main strategy to fund vantage6 maintenance at present is by participating in funded research projects that make use of vantage6. In such research projects, part of the funding may be allocated to maintain and extend the vantage6 infrastructure. While vantage6 is currently deployed in research projects with fixed scope, budget and time period, the ambition is to deploy vantage6 as

the infrastructure for federated registries as well. The establishment of permanent registries for rare cancers (EURACAN Registry, as piloted in IDEA4RC, Blueberry and other EURACAN projects) as well as a registry for childhood or adolescent and young adults (AYA in strong AYA) with cancer is under current discussion. Longer term service and software maintenance for such permanent federated registries should be provided by a specialized entity.

The development team stimulates other organizations that use vantage6 to participate in the development, deployment and servicing of vantage6. We believe open-source software is most sustainable with an active community, and are very willing to help developers from new organizations – both public and private – get started with vantage6. The higher the number of research projects using vantage6 is, the smaller the percentage of funding needed to allocate to its development will be.

Other considerations

Software quality improvement

The development team is aware that there are several points of improvement, such as better testing coverage and updates to the documentation. We allocate several hours a week specifically to work towards these goals. In doing so, we aim to further improve the quality of future versions of vantage6.

Security and privacy

Vantage6 is an infrastructure that facilitates running algorithms on multiple sites and combining their results. The infrastructure offers complete freedom in the implementation of algorithms, so it is important to properly configure the data stations so that users can only run an approved set of algorithms. More information on this may be found in the node configuration section of the documentation (https://docs.vantage6.ai/en/main/node/configure.html).

We are also working on risk assessments of the security and privacy within the vantage6 infrastructure, and for specific algorithms. This information is planned to be made available before the end of 2023. For now, please contact the authors of this document directly if you are interested in it.

Additional documentation

We encourage users of the vantage6 infrastructure to go through the documentation and the main project website vantage6.ai. In these places, they will also find additional relevant resources such as an example Data Privacy and Impact Assessment (DPIA), and other documents that are often important to vantage6 research projects.

In case of questions, you are encouraged to reach out to us via Discord (https://discord.gg/yAyFf6Y).

Acknowledgements

We would like to acknowledge André Dekker (Maastricht University), Carlos Martinez Ortiz (Netherlands eScience Center), and Luis Sanchez Gomez (Medical Data Works) for their thorough reviews and helpful contributions. We are also grateful to the European Commission-funded STRONG-AYA project for their financial support to IKNL and Maastricht University.

References

[1] C. Martinez-Ortiz, P. Martinez Lavanchy, L. Sesink, O. G. Brett, J. Meakin, M. de Jong, Z. Ancion, J. de Bruin, A. Culina, C. Erdmann, M. Grootveld, F. E. Psomopoulos, V. Sarkol, C. M. van Leeuwen and J. J. Vinju, “Practical guide to Software Management Plans,” 2023.

[2] [Online]. Available: https://www.health-ri.nl/initiatives/personal-health-train.

[3] Open Data Institute, “Federated learning: an introduction,” 2023.

[4] C. Zhao, S. Zhao, M. Zhao, Z. Chen, C. Z. Gao, H. Li and Y. Tan, “Secure multi-party computation: theory, practice and applications,” *Information Sciences,* vol. 476, pp. 357-372, 2019.

[5] “https://semver.org/,” [Online].

[6] A. Moncada-Torres, “VANTAGE6: an open source priVAcy preserviNg federaTed leArninG infrastructurE for Secure Insight eXchange,” in *AMIA Annual Symposium Proceedings*, 2020.

[7] D. Smits, B. van Beusekom, F. Martin, L. Veen, G. Geleijnse and A. Moncada Torres, “An Improved Infrastructure for Privacy-Preserving Analysis of Patient Data,” *Advances in Informatics, Management and Technology in Healthcare,* pp. 144-147, 2022.

[8] “https://numpydoc.readthedocs.io/en/latest/format.html,” [Online].