

Conjuror: A DICOM RT Plan generator for Linac QA

João Silveira¹, James R. Kerns¹, and Hasan Ammar¹

¹ Radformation, New York, NY, United States of America

DOI: [DOIunavailable](#)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Pending Editor](#) ↗

Reviewers:

- [@Pending Reviewers](#)

Submitted: N/A

Published: N/A

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

Quality assurance (QA) of linear accelerators (linacs) is a critical component of safe and effective external beam radiation therapy (RT). QA procedures often require the delivery of standardized treatment plans to verify machine performance, including geometric accuracy, dosimetric consistency, and imaging system alignment. While the Digital Imaging and Communications in Medicine (DICOM) standard defines how radiotherapy treatment plans (RT Plans) should be represented (National Electrical Manufacturers Association, n.d.), constructing valid RT Plan files for QA purposes is challenging due to the complexity of the standard and the need for machine-specific configurations.

Conjuror is a Python toolkit that enables the creation of DICOM RT Plans tailored for Varian linacs QA. The toolkit provides an accessible and reproducible way to generate test plans for routine and specialized QA procedures, eliminating the need for manual authoring or reliance on proprietary planning systems.

Statement of need

Routine and end-to-end QA of linacs requires standardized test plans that probe key performance characteristics, such as multileaf collimator (MLC) positioning, imaging isocenter coincidence, and mechanical accuracy. Guidelines such as AAPM Task Group 142 (Klein et al., 2009) prescribe many of these tests.

In current practice, generating these QA plans typically involves either (1) manual construction within proprietary treatment planning systems (TPS), or (2) relying on vendor-provided test plans which may not be adaptable to institution-specific protocols or research needs. Additionally, QA plans are often distributed across multiple files, making it cumbersome to assemble all required beams in a single cohesive plan and hindering streamlined, automated workflows.

Conjuror addresses this gap by offering an open-source, Python-based framework for generating DICOM RT Plans specifically for Varian linacs. The library enables:

- Customization and reproducibility of commonly used QA plans based on pre-prepared procedures.
- Aggregation of multiple procedures into a single RT plan, enabling a streamlined QA workflow.
- Visualization of the expected fluence and the sequence of control points of each beam.
- Integration with Python QA ecosystems (e.g. Pylinac (Kerns, 2023)), supporting automated analysis and research into novel QA strategies.
- Built on established Python packages such as pydicom (Mason, 2011), ensuring compatibility with the broader medical imaging software ecosystem.

State of the field

Several open-source projects address aspects of linac QA, but none focus on generating valid DICOM RT Plan object for test delivery. For example, pylinac (Kerns, 2023) and pymedphys

(Biggs et al., 2022) provide robust tools for analyzing machine QA data but assume that test plans have already been generated and delivered. As a foundational dependency, pydicom (Mason, 2011) enables reading and writing of DICOM files, but it does not provide domain-specific functionality for radiotherapy QA plan creation.

In contrast, Conjuro bridges this gap by directly supporting the programmatic generation of DICOM RT Plans for Varian linac QA. This focus on reproducible, automated test plan creation complements existing QA analysis software and contributes to a more complete open-source ecosystem for medical physics research and clinical practice.

Software design

- **RT beams:** A `Beam` represents a DICOM RT Beam and defines its Control Point Sequence (monitor units, MLC/jaw positions, gantry/collimator/couch axes).
- **QA procedures:** A `QAProcedure` groups one or more `Beam` instances into domain-specific tests (e.g., a Winston–Lutz procedure with multiple beams at different gantry/collimator/couch angles).
- **Plan synthesis:** A `PlanGenerator` builds an RT Plan dataset containing all beams across procedures. Since a QA procedure is not part of the DICOM RT Plan hierarchy, beams are stored as a sequential list.
- **Fluence simulation:** An imager simulator generates fluence from beam control points with configurable pixel sampling and source-to-imager distance.
- **Visualization:** Plotting helpers (`Plotly`) can animate MLC motion and chart beam control-point dynamics for inspection.

Research impact statement

Conjuro enables QA procedures to be defined and customized programmatically, allowing researchers and clinical physicists to generate reproducible RT Plan variants for systematic testing. This supports QA studies that evaluate machine behavior across controlled parameter changes and accelerates the development and assessment of novel QA strategies. Conjuro has been applied in a collaborative workflow to generate customized plans for testing Volumetric Modulated Arc Therapy (VMAT) deliveries across multiple settings beyond the fixed configurations available in vendor-provided QA plans.

AI usage disclosure

No generative AI tools were used in the conceptual design or QA procedure domain logic. Generative AI was used to assist with minor code suggestions (e.g., unit-test scaffolding) and language polishing of this manuscript; all changes were reviewed by the authors.

Acknowledgements

We thank Radformation for continuing to support the open-source work of this project.

References

- Biggs, S., Jennings, M., Swerdloff, S., Chlap, P., Lane, D., Rembish, J., McAloney, J., King, P., Ayala, R., Guan, F., Lambri, N., Crewson, C., & Sobolewski, M. (2022). PyMedPhys: A community effort to develop an open, python-based standard library for medical physics applications. *Journal of Open Source Software*, 7(78), 4555. <https://doi.org/10.21105/joss.04555>

- Kerns, J. R. (2023). Pylinac: Image analysis for routine quality assurance in radiotherapy. *Journal of Open Source Software*, 8(92), 6001. <https://doi.org/10.21105/joss.06001>
- Klein, E. E., Hanley, J., Bayouth, J., Yin, F.-F., Simon, W., Dresser, S., Serago, C., Aguirre, F., Ma, L., Arjomandy, B., Liu, C., Sandin, C., & Holmes, T. (2009). Task group 142 report: Quality assurance of medical accelerators. *Medical Physics*, 36(9Part1), 4197–4212. <https://doi.org/https://doi.org/10.1118/1.3190392>
- Mason, D. (2011). SU-e-t-33: Pydicom: An open source DICOM library. *Medical Physics*, 38(6Part10), 3493–3493.
- National Electrical Manufacturers Association. (n.d.). *NEMA PS3 / ISO 12052, digital imaging and communications in medicine (DICOM) standard*. NEMA; <https://www.dicomstandard.org/current/>.