Radiotherapy and Oncology 220 (2026) 111587

Contents lists available at ScienceDirect

Radiotherapy and Oncology

journal homepage: www.thegreenjournal.com

Review Article

Resource sharing and open-source software in radiation oncology: From challenges to opportunities for community-wide benefit☆

Samuel Ingram [a,b,*], Lorenzo Brualla [c,d], Alexia Delbaere [e], Masoud Zarepisheh [f], Hendrik Piersma [g], Aditya Apte [f], Roman Ludwig [h], Erik Roelofs [i], Marcel Nachbar [j], Ali Zaila [k], Krzysztof Domański [l], Josh Kirby [m], Giorgio Cartechini [n], Tobias Becher [o,p,q], Samuele Cavinato [r], Goran Stanic [o,p,q], Stine Korreman [s], Niklas Wahl [o,p], Josh Mason [t], Ana M. Barragán-Montero [u]

[a] *Christie NHS Foundation Trust, Manchester, United Kingdom*
[b] *University of Manchester, Manchester, United Kingdom*
[c] *Westdeutsches Protonentherapiezentrum Essen, Essen, Germany*
[d] *Instituto de Física Corpuscular (IFIC), CSIC-UV, Valencia, Spain*
[e] *University of Toulouse, Oncopole Claudius Regaud, IUCT-Oncopole, Inserm, CNRS, CRCT, Toulouse, France*
[f] *Department of Medical Physics, Memorial Sloan Kettering Cancer Center, New York, NY, USA*
[g] *Radiotherapeutisch instituut Friesland, Leeuwarden, the Netherlands*
[h] *Department of Radiation Oncology, University Hospital Zurich, Zurich, Switzerland*
[i] *Department of Radiation Oncology (Maastro), GROW School for Oncology and Reproduction, Maastricht University Medical Centre+, Maastricht, the Netherlands*
[j] *Section for Biomedical Physics, Department of Radiation Oncology, University Hospital and Medical Faculty, Eberhard Karls University Tübingen, Tübingen, Germany*
[k] *King Faisal Specialist Hospital and Research Center, Biomedical Physics Department, Riyadh, Saudi Arabia*
[l] *Maria Skłodowska-Curie National Research Institute of Oncology, Poland*
[m] *The Newcastle Hospitals NHS Foundation Trust, Newcastle upon Tyne, United Kingdom*
[n] *Maastricht University Medical Centre+, Department of Radiation Oncology (Maastro), GROW School for Oncology and Reproduction, Doctor Tanslaan 12, Maastricht 6229 ET, the Netherlands*
[o] *Department of Medical Physics in Radiation Oncology, German Cancer Research Center-DKFZ, Heidelberg, Germany*
[p] *Heidelberg Institute of Radiation Oncology (HIRO) and National Center for Radiation Research in Oncology (NCRO), Heidelberg, Germany*
[q] *Department of Physics and Astronomy, Heidelberg University, Heidelberg, Germany*
[r] *Medical Physics Department, Veneto Institute of Oncology IOV-IRCCS, 35128 Padua, Italy*
[s] *Centre for Particle Therapy, Department of Clinical Medicine, Aarhus University, Aarhus, Denmark*
[t] *Imperial College Healthcare NHS Trust, London, United Kingdom*
[u] *UCLouvain – Institut de Recherche Expérimentale et Clinique – Molecular Imaging Radiotherapy and Oncology (MIRO), Brussels, Belgium*

A B S T R A C T

Modern radiotherapy workflows depend heavily on software, yet proprietary solutions limit adaptability and transparency. Many institutions create in-house tools, resulting in duplicated efforts and sustainability challenges. Free and open-source software (FOSS) offers transparency, customization, and collaborative development opportunities, but faces barriers including fragmented efforts, regulatory complexity, and clinical adoption hurdles. This vision paper, inspired by the 2024 ESTRO Physics Workshop on "Resource sharing: open-source software & development in radiotherapy" explores these challenges and proposes strategies to promote sustainable and discoverable open-source ecosystems in radiation oncology. Key recommendations include improving interoperability, clear licensing, risk management, validation frameworks, and creating a centralised registry for resource visibility. Embracing open-source practices can accelerate innovation, reduce redundancy, and enhance patient care.

Introduction

Modern radiotherapy (RT) is increasingly shaped by software, both as a tool for research and as a fundamental component of clinical workflows. However, access to flexible, modifiable, and shareable software remains a major challenge within the RT community. The

☆ This article is part of a special issue entitled: 'Ben Mijnheer (RADION)' published in Radiotherapy and Oncology.

* Corresponding author at: The Christie 550 Wilmslow Road, Manchester M204BX, United Kingdom.
E-mail address: samuel.ingram@manchester.ac.uk (S. Ingram).

https://doi.org/10.1016/j.radonc.2026.111587
Received 12 January 2026; Received in revised form 22 April 2026; Accepted 6 May 2026
Available online 8 May 2026

dominance of proprietary medical software restricts opportunities for modification, interoperability, and transparent validation. As a result, to meet local clinical needs or explore new applications, many medical physicists and researchers independently develop in-house software. Developed largely in isolation, these efforts are often duplicated across institutions and struggle with visibility and long-term maintenance. This redundancy not only wastes valuable resources but also slows down innovation and ultimately restricts the potential for improving patient care.

Free and open-source software (FOSS) offers a compelling alternative, providing transparency, adaptability, and opportunities for collaborative development and validation [1–3]. While this approach has demonstrated success in areas such as medical image processing and analysis, with well-known libraries like MITK [4,5], ITK/SimpleITK [6], 3D Slicer [7], MONAI [8], or DIPY[9], the 2024 ESTRO Physics Workshop highlighted that the radiotherapy community is still actively navigating the unique barriers to achieving this same level of widespread FOSS integration. The current landscape of open RT software is fragmented, with valuable projects scattered across platforms and often lacking sustained community support. Smaller developments, such as AI models or planning scripts, sometimes shared alongside publications, lack the support needed for reuse, extension, or long-term maintenance. Solutions developed by clinical departments are typically constrained by regulation, institutional policies and vendor-specific environments. Together, these challenges reflect broader operational barriers: inconsistent standards and interfaces, limited documentation and discoverability, data privacy and regulatory constraints, and the absence of infrastructure and incentives for sharing and maintenance. These obstacles further complicate reuse and collaborative development for both researchers and clinical medical physicists which ultimately impacts the rate of innovation for patient benefit.

This lack of a cohesive ecosystem persists despite a growing consensus among healthcare organisations and professional societies on the necessity of transparent, rigorously validated software. Broader healthcare policies such as the UK Goldacre Review [10] and NHS Service Standards [11] increasingly mandate open-source coding practices. Concurrently, professional bodies provide comprehensive recommendations for software validation and the integration of IT specialists in radiation oncology (e.g., AAPM TG-53/TG-201 [12], IPEM guidelines [12,13], and recent joint position statements [12–14]). However, a significant gap remains between these high-level guidelines and the practical, day-to-day realities of open-source resource sharing within the clinical radiotherapy community.

In response to these challenges, a dedicated ESTRO Physics Workshop was convened in 2024, entitled “Resource Sharing: open-source and development in RT”, to discuss strategies for improving resource sharing and open-source software development in radiotherapy. The workshop brought together software developers, researchers, and clinicians to identify needs, best practices, and potential pathways towards more sustainable, visible, and collaborative software ecosystems. Discussions focused not only on open-source code, but also on related resources such as benchmark datasets, planning protocols, workflows, and community infrastructures.

This vision paper emerges from that initiative, outlining the motivation, challenges, and proposed recommendations and actions to improve resource sharing and encourage best practices when developing and implementing open-source software both in research and in clinical RT environments.

In this manuscript, software refers to computational tools used in radiotherapy workflows, including standalone applications, libraries/toolkits, and scripts or plugins that extend existing platforms. These tools may be developed for research or clinical purposes and implemented using either open-source or proprietary programming environments. On the other hand, free refers to software that is available at no cost, while open-source (or libre) refers to software whose source code is accessible, modifiable, and redistributable under an appropriate license. These concepts are related but distinct, as software may be free of charge without being open-source.

The landscape of open-source software and resource sharing practices in RT

Current status and challenges: Results of the survey

A pre-workshop survey (supplementary material) was distributed to the ESTRO physics mailing list. Of 103 responses received 80% worked in a clinical setting, 40% in research and 5% in industry (multiple responses were allowed to be selected for all questions). 71% of respondents were users of open-source code and 42% developed open-source code, 20% did neither. Asked if they had ever used open-source code, 88% had used it for research and 55% had used it clinically, while 73% of respondents had either developed, contributed to or shared open-source code or resources. Github was by far the most commonly used platform for code sharing. The main methods identified for finding resources were internet search, Github, networking, conference abstracts and journal publications. Asked to identify which radiotherapy research or clinical applications would especially benefit from resource sharing, the answers included all aspects of the radiotherapy pathway encompassing patient databases, data tools, scripts, templates and protocols for planning and checking, as well as tools and models for AI, Patient-Specific Quality Assurance (PSQA) and online adaptive workflows. Respondents were asked what they thought the main barriers to resource sharing were, and the answers are summarised in Table 1. The main barrier for research profiles was the need to improve (e.g. tidy up, document) or further test the code before sharing, followed by lack of time; while the main barrier for clinical profiles was the fear to be responsible for the code if something goes wrong, followed by intellectual property or licensing issues.

Although the sample of respondents is limited, the results suggest that the use and development of open-source code is rather high, even in clinical environments, and that the medical physics community is affected by the same common barriers for open-source adoption (i.e. documentation, lack of time for sustainable maintenance, licensing, etc).

During the workshop, the pros and cons of both open-source software and commercial solutions were discussed and a summary is presented in Table 2.

Motivation and benefits

Resource sharing through open-source software and scripts in healthcare and radiation oncology provides transformative benefits for both research and clinical practice [15,16]. Open-source solutions facilitate collaboration among institutions, researchers, and clinicians by pooling expertise and resources for robust studies and accelerated discovery [17]. These tools, in some cases, generate significant cost savings by eliminating upfront licensing fees. However, institutions must carefully weigh these savings against the internal 'total cost of

Table 1
Results of survey question “What are the main barriers to resource sharing in radiotherapy?”, showing the percentage of respondents who selected each option for both research and clinical usage.

Barrier	Research	Clinical
Lack of an obvious platform	44%	47%
Lack of time	48%	42%
Intellectual property or licensing issues	45%	50%
Institution or employer policy	38%	42%
The need to improve (i.e. tidy up, document) or further test code before sharing	55%	42%
Not wanting to be responsible for the code	37%	64%
Insufficient time to support the code or answer questions	26%	23%
Lack of academic credit compared to journal publication	40%	16%

Table 2
Commercial vs open-source comparison.

Feature	Commercial Software	Open-Source Software
Cost	High initial purchase costs and ongoing licensing fees	Free or minimal cost; may require greater investment for local support or customisation
Transparency	"Black box" implementation with limited visibility into algorithms	Full transparency with accessible source code for review and modification
Regulatory Approval	Typically comes with regulatory clearance (FDA, UKCA, EMA CE marking)	Often lacks formal regulatory approval, requiring local validation and quality management for clinical use – requiring local institutes to take on all responsibility and liability.
Support	Professional support with service level agreements	Community-based support; may offer paid support options
Documentation	Comprehensive for users, professionally developed	Variable quality; depends on community contributions
Updates & Maintenance	Regular updates provided by vendor	Community-driven updates; may be frequent but less structured
Customisation	Limited to vendor-approved options	Unlimited customisation potential
Integration	Optimised for specific workflows; may have limited interoperability	Variable integration capabilities: newer platforms like PortPy offer commercial system compatibility
Validation	Validated by manufacturer	Requires in-house validation before clinical implementation
Innovation Speed	Slower development cycles driven by commercial priorities	Rapid innovation through collaborative development
Examples	Eclipse, RayStation, Monaco	matRad, OpenTPS, PortPy, RDS, CERR
Training & Education	Vendor-provided training programs	Self-directed learning; community resources
Legal Liability	Vendor assumes some liability	Institution assumes full liability
Data Ownership	May have restrictions on data use and export	Complete data ownership and control
Research Flexibility	Limited ability to modify for research purposes	Ideal for research with complete modification freedom

ownership'; without dedicated commercial support, running and maintaining open-source tools requires organisations to actively allocate funds toward specialised in-house staffing, IT infrastructure, and local validation efforts (as summarised in Table 2) [18]. Transparency inherent to open-source software increases flexibility for customisation to specific clinical workflows and can provide enhanced stability through community-driven identification and correction of bugs across diverse use cases [19]. Public issue tracking clarifies tool capabilities and limitations, reducing redundant efforts [20].

Interoperability is a key advantage, as open-source tools break down silos between different healthcare systems and devices. They achieve this by natively supporting vendor-neutral standards (such as DICOM and FHIR), providing transparent access to underlying data structures, and offering open Application Programming Interfaces (APIs) that allow institutions to build custom bridges between otherwise incompatible commercial systems. This supports seamless data sharing essential for coordinated care and multi-institutional research collaborations. Additionally, open-source can foster innovation at pace, with an active community of users and developers who can work together to create solutions incredibly quickly. A few examples relevant to the medical physics community are MONAI [8], MatRad [21], OpenTPS [22], CERR [23], 3DSlicer [7], PortPy [24,25], Pylinac [26] or pymedphys [27], among many others. This collaborative approach has proven especially valuable during crises like COVID-19, where "plug and play" open-source technologies enabled rapid deployment of solutions across geographical and intellectual boundaries [28].

Open-source in the era of Artificial Intelligence (AI)

Artificial Intelligence (AI), broadly defined here as machine learning and deep learning algorithms designed to automate or augment clinical tasks, has demonstrated substantial benefits in radiotherapy, particularly for contouring and planning, by saving time, reducing interobserver variability (IOV) and improving quality and guideline adherence. There is a vast amount of AI models for segmentation built upon the open-source nnU-Net framework [29], establishing a standardised baseline for development. For instance, high-quality open-source models for pelvic MRI segmentation have been released to the community, demonstrating how open collaboration can produce robust clinical tools [30]. Furthermore, AI models for head and neck organ [31] and nodal segmentation [32] have shown reduced contour variability, and national-level models for breast segmentation have further enhanced consistency and accuracy [33]. More recently, expert-trained AI models have been proposed as a paradigm shift for disseminating radiotherapy contouring guidelines, such as in the EPTN consensus atlas for CT- and MR-based neuro-oncology contouring, where models are directly linked to expert consensus definitions [34,35]. AI-driven planning models similarly yield more consistent and improved treatment plans [36,37]. While most of these studies are done within a single institution or country at best, sharing these models publicly and releasing their code open-source would translate into a community-wide benefit for a more consistent and improved clinical practice world-wide. In addition, open-source AI offers greater control and transparency compared to many commercial tools, which is particularly important when tailoring models to small patient cohorts or rare diseases, where commercial interest may be limited. Open-source AI also allows dynamic adaptation and fosters trust through community scrutiny.

Despite its benefits, open-source AI in healthcare faces distinct challenges, especially related to data sharing and privacy. Sensitive patient data used to train AI models require stringent protections, complicating sharing across institutions due to regulatory, ethical, and privacy constraints [38]. This problem is amplified in low- and middle-income countries (LMICs) where infrastructural barriers further restrict effective data sharing for AI development [39]. The granularity and sensitivity of AI-related datasets, often including biometric and imaging data, raise privacy concerns beyond typical healthcare record exchanges. Open sharing of trained model weights versus training data itself remains a debated topic, impacting reproducibility and community validation while requiring careful governance to protect patient confidentiality [40].

Efforts in the development of "foundation models" by national or international academic consortia are starting to emerge. Projects such as the Danish AIM@CANCER initiative [41] utilise supercomputing power to generate robust foundation models on secure, national-level data. These generalised models will be able to support a wide range of clinical tasks. Foundational models can also be built upon or fine-tuned using local data, thereby ensuring high performance on specific patient cohorts without exposing the underlying data [42]. Crucially, this local optimisation also allows institutions to adapt models to their specific clinical practices (e.g., institutional contouring guidelines and planning techniques) and unique hardware setups (e.g., specific imaging equipment and protocols).

Finally, the translation of these open-source innovations into routine practice requires a framework for seamless integration into the clinical workflow. Open-source tools such as RadDeploy [43] have been developed specifically to bridge this gap, allowing in-house developed software and AI models to be deployed directly and efficiently into radiotherapy workflows.

Towards broader and sustainable resource sharing and use of FOSS

Regulatory overview

Regulations governing medical devices, including software, continuously evolve to address technological advances and emerging challenges. The summary below reflects the current regulatory landscape as of now, but institutions and developers should regularly consult official regulatory bodies and guidance documents to ensure compliance with the latest requirements.

Open-source software is not exempt from regulatory compliance, except if used for non-clinical research purposes only. When used in clinical settings, open-source software must meet the similar regulatory requirements as proprietary solutions but may not require formal approval (e.g. CE marking). In the European Union, developers should review the Medical Device Regulation (MDR 2017/745), which classifies certain health-related software as medical devices and imposes strict conformity assessment (e.g. documentation, quality management, monitoring) procedures. Additionally, the recently implemented EU Artificial Intelligence Act introduces risk-based obligations for AI systems, including open-source models, particularly if they are used in healthcare or safety–critical domains. In the United States, the Food and Drug Administration (FDA) regulates software as a medical device (SaMD), and open-source tools must adhere to FDA guidance on software validation. In the United Kingdom, post-Brexit regulations align closely with the MDR through the UK Medical Devices Regulations 2002 (as amended), with oversight from the MHRA. Developers and institutions using open-source software should assess whether their tools fall within the scope of these regulations and implement appropriate quality management systems, documentation, and risk mitigation strategies to ensure compliance.

Guidance documents are available to support the development and use of in-house Software as a Medical Device (SaMD) within healthcare institutions. For example, the Institute of Physics and Engineering in Medicine (IPEM) in the UK provides detailed best-practice advice on software production and sharing in medical contexts, emphasising risk management, validation, and regulatory considerations tailored to in-house development [44]. Similarly, others have presented frameworks for quality management, clinical evaluation, and continuous post-market surveillance of in-house SaMD, aligning institutional practices with evolving regulatory expectations [45]. These guidelines facilitate compliance while enabling innovation by clarifying how local developers can safely integrate SaMD into clinical workflows, provided that rigorous internal procedures are maintained. Institutions should closely follow these and other regional regulatory guidance documents to ensure their in-house SaMD meets patient safety requirements and adapts proactively to regulatory changes.

Improving clinical adoption of Open-Source software

Successfully translating FOSS into a clinical environment requires a well-structured ecosystem. Fig. 1 illustrates a hierarchy, governance, and workflow necessary to support this, highlighting how responsibilities for innovation, quality assurance, and ongoing risk management are distributed among key stakeholders. The adoption of open-source software in clinical settings presents unique challenges that require thoughtful strategies to overcome. As previously mentioned, despite the potential benefits of open-source solutions, healthcare institutions often hesitate to implement them due to concerns about reliability, support, and regulatory compliance. However, it is important to dispel a common misconception; acquiring a commercial software system does not absolve healthcare organisations of operational risk associated with that software. Responsibility for safe and compliant use remains with the institution, which must perform its own risk assessment, ongoing monitoring, and integration checks regardless of whether

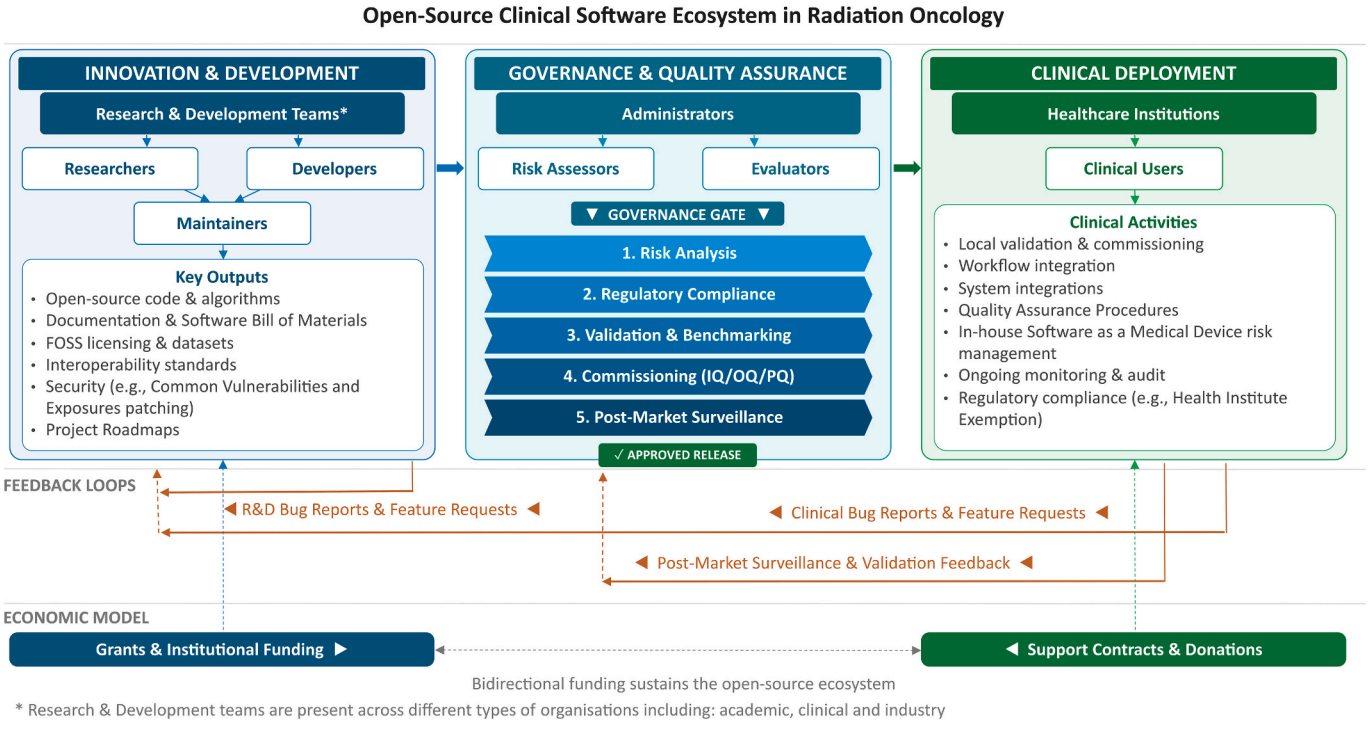

Fig. 1. The Open-Source Clinical Software Ecosystem. This schematic illustrates the workflow, governance structure, and distributed risk management required to translate FOSS into clinical practice. The ecosystem operates across three domains: **Innovation & Development** generates the core algorithms and FOSS outputs; **Governance & Quality Assurance** acts as the critical regulatory gateway where administrators and risk assessors enforce quality management systems (QMS); and **Clinical Deployment** represents the local validation and end-use by healthcare institutions. The economic model is sustained via bidirectional funding (grants and support contracts), while continuous risk management is maintained through direct feedback loops from the clinic back to developers and validators.

the system is purchased commercially or developed in-house. Addressing these barriers requires a multifaceted approach that considers both technical and organisational factors. An illustrative example is the structured, team-based development and implementation cycle for in-house clinical software described by Moran et al.[2] In the following, we provide a set of recommendations to improve the clinical adoption of FOSS.

- **Integration and interoperability** are essential for clinical adoption of open-source software. Solutions that can interface with commercial systems (e.g. running the software through scripting within the commercial system, or automatic DICOM import/export) have significant advantages, as they allow healthcare providers to leverage open-source innovation while maintaining compatibility with existing infrastructure. This bridging approach enables gradual adoption without disrupting established clinical workflows. A modular software design further enhances this capability by separating computational algorithms for different processes, eliminating dependencies that make model modifications and extensions cumbersome in complex systems. This component-based architecture allows users to select specific tools rather than adopting an entire ecosystem, making implementation more flexible and targeted to specific clinical needs.
- **Clear licensing** is crucial for healthcare adoption, as many open-source licenses that exclude commercial use inadvertently exclude hospital use as well. Web tools can be used to help navigate premade licences (e.g. choosealicense.com) and specific care should be taken when writing custom licences. Hospitals from a legal context are often viewed as a business that generates revenue through patient care, even when they are non-profit institutions, making their use of software typically classified as commercial. Intellectual property and licensing concerns were identified as a major barrier in the survey (Table 1). While web tools (e.g., choosealicense.com) help navigate premade licenses, developers must engage early with their institutional technology transfer or legal offices to align FOSS goals with local IP policies. To assist developers and institutions in navigating these complexities, Table 3 summarizes the fundamental components of FOSS licensing and practical mitigation strategies.
- **Roadmap design**. Alongside licensing, well-defined development roadmaps provide essential clarity about a project's vision and direction. These roadmaps help contributors and users understand the project's objectives, aid in prioritising tasks, and foster trust by offering transparency into development plans. A clear description of what features will be supported and what remains outside the scope helps clinical users make informed decisions about adoption.
- **Risk analysis.** Documented failure mode and effect analysis (FMEA), hazard testing and risk analysis evidence (e.g. TG100, ISO14971 and DCB0129) can be useful for clinical teams to adopt solutions [46] and provide a parallel to commercial solutions, but it should be stressed that implementation risk management is required by adopting entities.
- **Intended use and users.** It is key to establish a formal intended purpose statement along with establishing mechanisms to showcase who is using the code, such as collaborator lists, and how they are implementing it builds credibility and confidence among potential adopters. Not wanting to be responsible for the code was the largest barrier reported by clinical users (Table 1). While absolute liability cannot always be removed, this fear can *sometimes* be mitigated by clearly defining intended use boundaries (e.g., sharing for research purposes only) and explicitly transferring clinical validation responsibilities to the adopting institution, as further detailed in Section 3.3.
- **Versioning**. Clear versioning that distinguishes between major and minor changes helps clinical users understand the implications of updates and plan accordingly.
- **Structure and documentation**. In general, open-source software benefits substantially from adherence to development best practices. A structured and well-documented project increases ease of adoption and credibility. Resources like the CLEAR checklist [47] provide actionable steps to ensure published software meets a baseline standard. In addition, cookie-cutter templates, such as this example created by the ESTRO working group (GitHub repository), can aid project leaders to scaffold new tools and support them in creating high-quality software. Additionally there are an increasing number of standardisations for developed AI models that can help improve adoption [48,49]. Together, these two approaches lower barriers to sharing and adopting code by promoting higher quality and easier implementation. The survey (Table 1) highlighted the 'need to tidy code' as the primary barrier for researchers. Utilising standardised resources like the CLEAR checklist or community-provided cookie-cutter templates from the start of a project can significantly improve the readiness of code when it is time to share.
- **AI-assisted Development.** The rapid emergence of AI-assisted coding tools may offer new opportunities to accelerate the development of open-source software by lowering technical barriers, improving documentation, and facilitating code generation and maintenance. However, the reliance on AI-generated code introduces new barriers that the community must navigate carefully. Primary among these are the legal ambiguities surrounding the copyright and licensing of AI-generated snippets, as well as the risk of introducing subtle vulnerabilities or 'hallucinations' into safety–critical clinical software. Therefore, while AI can accelerate development, it does not replace the need for rigorous human oversight, peer review, and automated testing.
- **Security.** Security and compliance integration which includes data encryption, access controls, and appropriate isolation are all relevant in healthcare settings. Pilot projects and phased implementation strategies allow for controlled adoption with minimal disruption. If available, it can be useful to compare open-source and proprietary tools directly. There should be a documented approach for dealing with Common Vulnerabilities and Exposures (CVEs).
- **Regulatory compliance.** Ultimately compliance is the duty of the teams implementing FOSS. However, several features can aid in FOSS being adopted:
  - Defined software scope, requirements, and traceability for changes.
  - Software bill of materials (SBOM, a comprehensive inventory of all underlying software components and dependencies)
  - Procedure documented for release management and Common Vulnerabilities and Exposures (CVEs, publicly disclosed cybersecurity flaws)
  - Mechanisms for software audit logging (tracking system events and user actions for security, debugging, and traceability)
- **Commissioning.** Commissioning confirms that open-source software functions correctly and is safe for both research and clinical use.

Table 3
Key considerations and practical strategies for open-source software licensing in clinical settings.

Licensing Consideration	Practical Details for RT FOSS
Key License Components	Dictates permissions (commercial use, modification, distribution), conditions (attribution, disclosing source code/copyleft), and limitations (liability, warranty disclaimers).
Factors for Selection	Influenced by institutional IP policies, compatibility with underlying software dependencies, and whether the primary goal is broad clinical adoption vs. eventual commercialisation.
Mitigating Restrictive Licenses	If a license precludes hospital use (e.g., strict non-commercial clauses), institutions can contact the authors to negotiate a dual-license or a specific institutional exemption.

Key steps include defining intended use, testing core features against requirements, and documenting all results. For clinical settings, teams may opt to use standardised protocols such as Installation Qualification (IQ), Operational Qualification (OQ), and Performance Qualification (PQ) to ensure reliability in real-world environments. Keeping validation records current supports audits and ongoing quality control, helping adopters maintain regulatory compliance and trust across diverse healthcare contexts.

- **Challenges**. Algorithm challenges are central for transparent comparison and reproducible software development. In particular, deep learning challenges have become powerful benchmarking tools through shared datasets, predefined evaluation metrics, and public leaderboards. Within radiation oncology, such competitions have advanced tasks including segmentation [50–52], synthetic CT generation[53,54], treatment planning [55], tumour tracking [56], and deformable image registration [57]. Hosted on platforms such as the Grand Challenge or CodaBench (formerly CodaLab), these initiatives often produce curated open datasets and evaluation pipelines that serve as valuable benchmarks for the community. While challenges have so far been predominantly associated with deep learning, similar frameworks could be extended to other types of algorithms, providing structured mechanisms to evaluate and validate software before broader clinical adoption.

A safe approach to sharing

As previously mentioned, software developed for healthcare institutions can often meet the definition of a medical device in the UK, EU, and USA, requiring compliance with relevant regulations (see Section "Regulatory Overview"). In all three regions, software that controls clinical equipment or influences clinical decisions must meet medical device regulations. In the UK and EU, local in-house use can be exempt from full conformity assessment [58,59], but any external distribution requires regulatory approval and marking. Health Institution Exemption (HIE) is consistent across these jurisdictions, with region-specific standards and procedures guiding compliance. Sharing software that is intended for research purposes but is adopted by clinical teams clearly places the liability on the clinical institute. However, sharing software that has been created and used by healthcare institutions for clinical purposes is complicated due to limitations imposed by regulations such as the EU & UK MDR.

One approach to sharing software from the clinic is to only share components of it that do not have the purpose of being clinical software but are proposed as research or educational tools. This distinction is crucial, as it changes the regulatory requirements significantly. However, this requires clear communication about the intended use of the software and explicit disclaimers that it is not intended for clinical decision-making without appropriate validation and documentation consistent with the requirements of the HIE by the receiving institution. Even with this it may not be enough to remove all liability of open-source software from the institution [60] and it is advisable to discuss directly with the locally appointed Regulatory Agency.

Developing license addendums specifically designed for non-medical device software sharing could provide the community with greater confidence in this approach. These addendums should include explicit language stating that the software is being shared for research, educational, or developmental purposes only, and not as a certified medical device. They would clarify that any clinical implementation requires the receiving institution to perform their own validation and assume responsibility for its use, wherever possible this should be guided by a locally appointed Regulatory Agency. An alternative approach which is more thorough is to entirely separate out clinically implemented and open-source versions; there are multiple architectural approaches that can be used to achieve this, and it offers clearer distinction between the software being used clinically and that which is for non-clinical use, these are detailed in Table 4.

A platform for increased visibility

Besides the above-described practical barriers regarding operability (e.g. documentation, licensing, regulations, etc), one of the keys for improved resource sharing is increasing the visibility of the existing initiatives and developed software. Currently, users seeking FOSS often rely on excellent but informally maintained community lists, such as the 'awesome-medphys' repository on GitHub [61], to discover available resources. To build upon these community efforts and standardise discoverability, the working group formed after the ESTRO workshop proposes the development of a centralised, user-friendly registry to improve visibility and accessibility of software tools and datasets in RT. Rather than hosting these tools directly, the platform will serve as a **reference hub** linking to established repositories such as GitHub, vendor sites (e.g., Varian or RaySearch), repositories for commercial solutions for specific topics (e.g. DLinRT.eu), or data platforms like TCIA and cancerdata.org, as well as broader scientific registries like Papers

Table 4
Sharing approaches for software that may be clinically viable.

Sharing Approach	Clinical Version Looks Like	Open-Source Version Looks Like	Pros	Cons
Building Blocks Only	Clinical software implements validated, institution-specific modules using shared documentation or abstract interfaces (no direct open-source code)	Open-source community maintains specification documents, interface definitions, or workflow blueprints	Enables knowledge transfer; reduces institutional liability by avoiding code sharing; facilitates interoperability	Lack of executable code for immediate adoption; variable implementation quality; higher effort required to develop compliant systems
License-Only	Closed-source copy of open-source code base that has undergone local governance for clinical use.	Open-source code base, licensed for research/education only	Simplicity; minimal copies of code; ensures everyone works from the same code base; can have separate validation processes	License restrictions may limit adoption; less flexibility for bespoke clinical customisation; may not be sufficient to remove institutional liability
Fork-Only	Separate clinical software fork with specific clinical features	Public open-source repository with community-driven features	Maintains open development; clear versioning; clinical fork can evolve independently	Code divergence; higher maintenance burden; possible duplication of effort
Separate Open-Source Core + Commercial Extension	Clinical software built on a stable open-source core with proprietary clinical code on top	Open-source core with minimal clinical features	Balance of openness and proprietary control; clinical code can be tightly governed	Complexity in managing integration; version compatibility challenges
Git Submodules	Clinical software includes open-source core as controlled Git submodule	Open-source core maintained independently.	Clear dependency management; enables reuse of stable code	Complexity in repository management; potential integration issues
Plugin Architecture	Clinical features implemented as plugins	Core software remains fully open-source	Modular, flexible customisation; easy to develop clinical features separately	Dependency management complexity; plugin compatibility issues

with Code or the institutionally supported Research Software Directory. The registry would encourage collaboration, validation, and reuse across the community, offering rich metadata, persistent identifiers (e.g., DOIs), and user-friendly submission templates. Automation, lightweight administration, and quality indicators (e.g., privacy statements, license compliance, and code comments) will promote best practices without erecting publication barriers. The ideal platform would also integrate community-driven feedback tools, such as a rating mechanism for resources, enabling users to assess relevance and quality over time. The development of this platform is currently an ongoing effort by this working group.

This initiative places strong emphasis on open and libre software principles, supporting FOSS licensing models that allow reuse, adaptation, and even commercial deployment, provided regulatory requirements are met. In this context, developers will be encouraged to contribute to existing FOSS projects (e.g., via pull requests), rather than duplicating efforts, fostering a sustainable innovation ecosystem. The platform will enforce license disclosure and promote transparent tracking of a tool's Technology Readiness Level (TRL), following ISO 16290:2013 standards. Software metadata tags could help the community to distinguish between resources from early-stage research tools (TRL 4–6) to clinically validated prototypes (TRL 7–8), but clinical deployment (TRL 9) will require institutions to take responsibility for regulatory compliance, such as the previously discussed EU MDR, US FDA, or UK MHRA regulations. Additional functionalities such as optional benchmarking with open datasets and a dedicated community forum would further support reproducibility and collaborative advancement in RT software development.

Eventually, sharing information is as important as sharing software or datasets. Cataloguing ongoing research efforts, commercial tools, and evaluation challenges in a centralised, accessible way adds tremendous value to the community. For example, the recent initiative DLinRT.eu offers a comprehensive overview of deep learning solutions in radiotherapy, focusing mainly on commercial solutions but also including academic initiatives like the Grand Challenges (e.g. SynthRAD [54], StructSeg [51], etc). This allows the community to easily compare products and track progress in the field. Integrating and referencing such initiatives within the registry will help maintain a dynamic, up-to-date landscape of resources and developments relevant to the RT community.

Discussion

The ESTRO Workshop on open-source radiotherapy software discussions and subsequent analysis highlight a central tension for software in radiation oncology. On one hand, there is a clear and growing need for adaptable and transparent digital tools to support research, innovation, and clinical workflows. On the other, the current reliance on proprietary systems and isolated institutional developments creates silos, slows innovation, and limits reproducibility. The open-source model provides a compelling path forward, but its widespread adoption requires addressing specific cultural, regulatory, and operational barriers.

A key insight is that successful resource sharing must extend beyond software code to include documentation, clinical protocols, benchmark datasets, and workflows. Case studies presented at the ESTRO workshop demonstrate that even modest levels of sharing such as scripting libraries or AI segmentation models can generate community benefits by reducing duplication and providing baseline standards. Moving forward, the community would benefit from structured approaches to promote trust, ensure clinical safety, and manage liability risks. This involves clarity in licensing, careful risk analysis, and transparent validation strategies that mirror those already established for commercial solutions. There is a continued need for the publication of clear success stories to help exemplify to the wider community how to navigate the implementation of FOSS in the clinics. This should be supported by new article types, such as the Green Technical Vision "How to" articles within the Green Journal. Specifically, there is currently a lack of comprehensive data regarding how different healthcare administrations formally oversee, support, and assume liability for physicists using and developing software, and how these processes vary across jurisdictions. Future work must systematically collate institutional case studies to map these diverse governance strategies. Furthermore, scientific journals can drive cultural change by increasingly encouraging or mandating code sharing for publications that present in-house software with, for example special issues and dedicated article types. To directly address the lack of academic credit, a key barrier identified in our survey developers, should also leverage dedicated peer-reviewed software journals, such as the Journal of Open-Source Software (JOSS) or SoftwareX, which provide formal, citable recognition for codebases on top of existing, yet not peer-reviewed citable software services like Zendo.

Several themes emerged as particularly important for sustaining progress:

- **Community governance and visibility:** Without a concerted effort to organise and maintain collective projects, work conducted will remain fragmented. A central registry or metadata hub, as proposed by the working group, could provide a unifying framework.
- **Guidelines and Templates:** Identification of project features helps create high quality FOSS that can be adopted by the community. Tools such as CLEAR [47], PROBAST [62], TRIPOD [63] promote best practice.
- **Clinical integration:** Adoption depends on interoperability with existing systems, modular architectures, and clear distinctions between research and clinical versions. These strategies allow gradual integration without disrupting clinical workflows.
- **Regulation and safety:** Open-source, when employed clinically, is not exempt from regulatory requirements. Practical frameworks, such as in-house SaMD guidance, can help institutions navigate compliance while still benefiting from FOSS flexibility.
- **Keeping momentum:** Beyond formal publications, recurring workshops, webinars, and dedicated sessions at conferences such as ESTRO and ICCR are needed to sustain dialogue, share best practices, and foster new collaborations.

The ESTRO Workshop on open-source radiotherapy software demonstrates both the appetite and feasibility for coordinated action. Professional societies play a vital role in maintaining momentum by convening communities, endorsing standards, and embedding resource sharing within broader professional development frameworks. Furthermore, to alleviate the prohibitively high burden of regulatory compliance on individual institutions, these societies should consider establishing jurisdictional advisory committees. Such groups could provide templated documentation, standardised risk assessment forms, and peer-to-peer guidance for FOSS implementation. This effort must be echoed by universities and training programs integrating open-source practices directly into medical physics curricula. Furthermore, public health systems and funding bodies must evolve to explicitly incentivise and financially support not just the initial development, but the long-term maintenance of these shared resources. Over time, aligning incentives through recognition, citation, or funding mechanisms will also be important to encourage sustainable contributions.

Conclusion

In conclusion, resource sharing and open-source software represent not only a technical opportunity but also a cultural shift for radiation oncology. By embedding open, collaborative practices into the community's infrastructure, the field can accelerate innovation, promote equity of access, and ultimately improve care delivery for patients worldwide.

Funding sources

LB acknowledges financial support from EURATOM PIANOFORTE grant agreement 101,061,037 (DOSELIA), PID2022-140503OB-I00 by the Spanish MICIU/AEI/https://doi.org/10.13039/501100011033, ERDF/EUgrant, and GVA grant CIPROM/2021/64. MZ and AA acknowledge partial financial support by MSK Cancer Center Support Grant/Core Grant from the NIH (P30CA008748). AB is a Research Associate at FNRS – UCLouvain.

CRediT authorship contribution statement

Samuel Ingram: Writing – review & editing, Writing – original draft, Resources, Data curation, Conceptualization. **Lorenzo Brualla:** Writing – review & editing, Writing – original draft, Resources, Conceptualization. **Alexia Delbaere:** Writing – review & editing, Writing – original draft, Resources, Conceptualization. **Masoud Zarepisheh:** Writing – review & editing, Resources. **Hendrik Piersma:** Writing – review & editing, Writing – original draft, Resources. **Aditya Apte:** Writing – review & editing. **Roman Ludwig:** Writing – review & editing. **Erik Roelofs:** Writing – review & editing, Conceptualization. **Marcel Nachbar:** Writing – review & editing. **Ali Zaila:** Writing – review & editing. **Krzysztof Domański:** Writing – review & editing. **Josh Kirby:** Writing – review & editing. **Giorgio Cartechini:** Writing – review & editing. **Tobias Becher:** Writing – review & editing. **Samuele Cavinato:** Writing – review & editing. **Goran Stanic:** Writing – review & editing. **Stine Korreman:** Writing – review & editing, Resources. **Niklas Wahl:** Writing – review & editing, Project administration, Conceptualization. **Josh Mason:** Writing – review & editing, Writing – original draft, Resources, Project administration, Data curation, Conceptualization. **Ana M. Barragán-Montero:** Writing – review & editing, Writing – original draft, Supervision, Project administration, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

2024 ESTRO Physics Workshop on “Resource sharing: open-source software & development in radiotherapy” who have indirectly contributed to this work through lively and fruitful discussions in the various workshop sessions. Further, the authors acknowledge the support of the ESTRO Physics Committee for hosting the workshop to enable the framework for exchange and discussion leading to this work.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.radonc.2026.111587.

References

[1] Chauhan RS, Munshi A, Pradhan A. The role of python in enhancing radiotherapy department workflow efficiency and promoting open-source software utilisation. Clin Oncol (R Coll Radiol) 2025 Sep;45:103897.
[2] Moran JM, Paradis KC, Hadley SW, Matuszak MM, Mayo CS, Naheedy KW, et al. A safe and practical cycle for team-based development and implementation of in-house clinical software. Adv Radiat Oncol 2022;7(1):100768.
[3] Zhang L, Hub M, Mang S, Thieke C, Nix O, Karger CP, et al. Software for quantitative analysis of radiotherapy: overview, requirement analysis and design solutions. Comput Methods Programs Biomed 2013 Jun;110:528–37.
[4] The Medical Imaging Interaction Toolkit (MITK) [Internet]. [cited 2025 Sep 15]. Available from: https://www.mitk.org/.
[5] Wolf I, Vetter M, Wegner I, Böttger T, Nolden M, Schöbinger M, et al. The medical imaging interaction toolkit. Med Image Anal 2005 Dec;9:594–604.
[6] Lowekamp BC, Chen DT, Ibáñez L, Blezek D. The Design of SimpleITK. Frontiers in neuroinformatics [Internet]. 2013 Dec 30 [cited 2026 Mar 18];7. Available from: https://pubmed.ncbi.nlm.nih.gov/24416015/.
[7] Fedorov A, Beichel R, Kalpathy-Cramer J, Finet J, Fillion-Robin JC, Pujol S, et al. 3D Slicer as an image computing platform for the Quantitative Imaging Network. Magn Reson Imaging 2012 Nov;30:1323–41.
[8] Cardoso MJ, Li W, Brown R, Ma N, Kerfoot E, Wang Y, et al. MONAI: An open-source framework for deep learning in healthcare [Internet]. arXiv [cs.LG]. 2022. Available from: http://arxiv.org/abs/2211.02701.
[9] Garyfallidis E, Brett M, Amirbekian B, Rokem A, van der Walt S, Descoteaux M, et al. Dipy, a library for the analysis of diffusion MRI data. Front Neuroinf 2014 Feb;21:8.
[10] Goldacre B. Better, broader, safer: using health data for research and analysis [Internet]. Department of Health and Social Care, UK Government. 2022 [cited 2026 Mar 31]. Available from: https://assets.publishing.service.gov.uk/media/624ea0ade90e072a014d508a/goldacre-review-using-health-data-for-research-and-analysis.pdf.
[11] Make new source code open – NHS digital service manual [Internet]. nhs.uk. [cited 2026 Mar 31]. Available from: https://service-manual.nhs.uk/standards-and-technology/service-standard-points/12-make-new-source-code-open.
[12] AAPM Reports - Report of Task Group 201 of the American Association of Physicists in Medicine: Quality Management of External Beam Therapy Data Transfer [Internet]. [cited 2026 Mar 31]. Available from: https://www.aapm.org/pubs/reports/detail.asp?docid=213.
[13] Best-practice guidance for the in-house manufacture of medical devices and non-medical devices, including software in both cases, for use within the same health institution. [Internet]. [cited 2026 Mar 31]. Available from: https://www.ipem.ac.uk/media/0cjfijnf/ipem-best-practiceguidance-on-ihmu-v2-2_final.pdf.
[14] Peters S, Meinschad M, Dennstädt F, Bovet M, Fahrner H, Fröhlich M, et al. The need for IT specialists in radiation oncology - A position paper by the International Society for radiation oncology Informatics, endorsed by DGMP, SASRO, ÖGRO, ÖGMP, SRO and DEGRO. Radiotherapy and oncology : journal of the European Society for Therapeutic Radiology and Oncology [Internet]. 2025 Dec [cited 2026 Mar 31];213. Available from: https://pubmed.ncbi.nlm.nih.gov/41052607/.
[15] Erickson BJ, Langer S, Nagy P. The role of open-source software in innovation and standardization in radiology. J Am Coll Radiol 2005 Nov;2:927–31.
[16] Pinter C, Lasso A, Wang A, Jaffray D, Fichtinger G. SlicerRT: radiation therapy research toolkit for 3D Slicer. Med Phys 2012 Oct;39:6332–8.
[17] Toward Distributed Conduction of Large-Scale Studies in Radiation Therapy and Oncology: Open-Source System Integration Approach [Internet]. [cited 2025 Dec 19]. Available from: https://doi.org/10.1109/JBHI.2015.2450833.
[18] Peplinski JE, Pearce JM. Economic Efficiency of an Open-Source National Medical Lab Software in Canada. J Med Syst 2023 Apr 21;47:50.
[19] Gropper A. Open-source health care software. Virtual Mentor 2011 Sep 1;13:632–6.
[20] Akünal Ü, Bujotzek M, Denner S, Hamm B, Kades K, Schader P, et al. Kaapana: A Comprehensive Open-Source Platform for Integrating AI in Medical Imaging Research Environments [Internet]. 2025 [cited 2025 Dec 19]. Available from: http://arxiv.org/abs/2512.09644.
[21] Wieser HP, Cisternas E, Wahl N, Ulrich S, Stadler A, Mescher H, et al. Development of the open-source dose calculation and optimization toolkit matRad. Med Phys 2017 Jun;44:2556–68.
[22] Wuyckens S, Dasnoy D, Janssens G, Hamaide V, Huet M, Loÿen E, et al. OpenTPS – Open-source treatment planning system for research in proton therapy [Internet]. arXiv [physics.med-ph]. 2023. Available from: http://arxiv.org/abs/2303.00365.
[23] Deasy JO, Blanco AI, Clark VH. CERR: a computational environment for radiotherapy research. Med Phys 2003 May;30:979–85.
[24] PortPy: An Open-Source Initiative for Radiotherapy Treatment Planning Optimization Including Benchmark Data and Algorithms and Integration with Commercial TPS. [Internet]. [cited 2025 Oct 8]. Available from: https://www.iccr2024.org/papers/523895.pdf.
[25] Jhanwar G, Tefagh M, Taasti VT, Alam S, Tuomaala S, Nadeem S, et al. Portpy: An Open-Source Python Package for Planning and Optimization in Radiation Therapy Including Benchmark Data and Algorithms. In: AAPM 65th Annual Meeting & Exhibition [Internet]. AAPM; 2023 [cited 2026 Mar 31]. Available from: https://aapm.confex.com/aapm/2023am/meetingapp.cgi/Paper/4208.
[26] Kerns JR. Pylinac: image analysis for routine quality assurance in radiotherapy. J Open Sour Softw 2023 Dec 3;8:6001.
[27] Biggs S, Jennings M, Swerdloff S, Chlap P, Lane D, Rembish J, et al. PyMedPhys: a community effort to develop an open, Python-based standard library for medical physics applications. J Open Source Softw 2022 Oct 16;7:4555.
[28] Besançon L, Peiffer-Smadja N, Segalas C, Jiang H, Masuzzo P, Smout C, et al. Open science saves lives: lessons from the COVID-19 pandemic. BMC Med Res Method 2021 Jun 5;21:117.
[29] Isensee F, Jaeger PF, Kohl SAA, Petersen J, Maier-Hein KH. nnU-net: a self-configuring method for deep learning-based biomedical image segmentation. Nat Methods 2021 Feb;18:203–11.
[30] Lorenzen EL, Celik B, Sarup N, Dysager L, Christiansen RL, Bertelsen AS, et al. An open-source nnU-net algorithm for automatic segmentation of MRI scans in the male pelvis for adaptive radiotherapy. Front Oncol 2023 Nov;3:1285725.
[31] van der Veen J, Willems S, Deschuymer S, Robben D, Crijns W, Maes F, et al. Benefits of deep learning for delineation of organs at risk in head and neck cancer. Radiother Oncol 2019 Sep;138:68–74.
[32] van der Veen J, Willems S, Bollen H, Maes F, Nuyts S. Deep learning for elective neck delineation: more consistent and time efficient. Radiother Oncol 2020 Dec;153:180–8.

[33] Buhl ES, Lorenzen EL, Refsgaard L, Nielsen AWM, Brixen ATL, Maae E, et al. Development and comprehensive evaluation of a national DBCG consensus-based auto-segmentation model for lymph node levels in breast cancer radiotherapy. Radiother Oncol 2024 Dec;201:110567.
[34] Montero, A. Barragán and Huet-Dastarac, Margerie and Di Perri, Dario and Hofstede, David and Birimac, Nikolina Elsa and Geerts, Matijs and Roberfroid, Benjamin and Quéré, Emilien and Lee, John Aldo and Roelofs, Erik and van Elmpt, Wouter and Eekers, Danielle and Zegers, Catharina M.L, Expert trained AI-models as a paradigm shift for disseminating radiotherapy contouring guidelines: The EPTN consensus atlas for CT- and MR-based neuro-oncology contouring. Available at SSRN: https://ssrn.com/abstract=6309138 or https://doi.org/10.2139/ssrn.6309138.
[35] Mashayekhi M, McBeth R, Nguyen D, Yen A, Trivedi Z, Moon D, et al. Artificial intelligence guided physician directive improves head and neck planning quality and practice Uniformity: a prospective study. Clin Transl Radiat Oncol 2023 May;40:100616.
[36] Panteli D, Adib K, Buttigieg S, Goiana-da-Silva F, Ladewig K, Azzopardi-Muscat N, et al. Artificial intelligence in public health: promises, challenges, and an agenda for policy makers and public health institutions. Lancet Public Health 2025 May;10:e428–32.
[37] Kaushik A, Barcellona C, Mandyam NK, Tan SY, Tromp J. Challenges and opportunities for data sharing related to artificial intelligence tools in health care in low- and middle-income countries: systematic review and case study from Thailand. J Med Internet Res 2025 Feb;4:e58338.
[38] Harish KB, Nicholson Price W, Aphinyanaphongs Y. Open-source clinical machine learning models: critical appraisal of feasibility, advantages, and challenges. JMIR Formative Res 2022 Apr 11;6:e33970.
[39] Supercomputer and AI to boost Danish cancer treatment – new project awarded €7 million. 2025 Dec 18 [cited 2025 Dec 19]; Available from: https://health.medarbejdere.au.dk/en/display/artikel/supercomputer-and-ai-to-strengthen-danish-cancer-treatment-new-project-secures-dkk-53-million.
[40] Kawula M, Hadi I, Nierer L, Vagni M, Cusumano D, Boldrini L, et al. Patient-specific transfer learning for auto-segmentation in adaptive 0.35 T MRgRT of prostate cancer: a bi-centric evaluation. Med Phys 2023 Mar;50:1573–85.
[41] Ersted Rasmussen M, Dueholm Vestergaard C, Folsted Kallehauge J, Ren J, Haislund Guldberg M, Nørrevang O, et al. RadDeploy: a framework for integrating in-house developed software and artificial intelligence models seamlessly into radiotherapy workflows. Phys Imaging Radiat Oncol 2024 Jul;31:100607.
[42] Best-practice guidance for the in-house manufacture of medical devices and non-medical devices, including software in both cases, for use within the same health institution [Internet]. [cited 2025 Oct 8]. Available from: https://www.ipem.ac.uk/media/vp0ewy01/ipembe-1.pdf.
[43] Lagerburg V, van den Boorn M, Crane RF, Welvaars K, Groen JM. Applying and validating a quality management system for in-house developed medical software. Front Digit Health 2025 Apr;1:1461107.
[44] De Kerf G, Barragán-Montero A, Brouwer CL, Pisciotta P, Biston MC, Fusella M, et al. Multicentre prospective risk analysis of a fully automated radiotherapy workflow. Phys Imaging Radiat Oncol 2025 Apr;34:100765.
[45] Kocak B, Baessler B, Bakas S, Cuocolo R, Fedorov A, Maier-Hein L, et al. CheckList for EvaluAtion of Radiomics research (CLEAR): a step-by-step reporting guideline for authors and reviewers endorsed by ESR and EuSoMII. Insights Imaging 2023 May 4;14:75.
[46] Li H, Chen M. Automatic Structure Segmentation for Radiotherapy Planning Challenge 2020. [cited 2026 Mar 18]; Available from: https://zenodo.org/records/3718885.
[47] Gatidis S, Früh M, Fabritius MP, Gu S, Nikolaou K, Fougère CL, et al. Results from the autoPET challenge on fully automated lesion segmentation in oncologic PET/CT imaging. Nat Mach Intellig 2024 Oct 30;6:1396–405.
[48] Andrearczyk V, Oreiller V, Abobakr M, Akhavanallaf A, Balermpas P, Boughdad S, et al. Overview of the HECKTOR Challenge at MICCAI 2022: Automatic Head and Neck Tumor Segmentation and Outcome Prediction in PET/CT. Head and Neck Tumor Segmentation and Outcome Prediction. 2023;1–30.
[49] Huijben EMC, Terpstra ML, Galapon Jr A, Pai S, Thummerer A, Koopmans P, et al. Generating synthetic computed tomography for radiotherapy: SynthRAD2023 challenge report. Med Image Anal 2024 Oct;97:103276.
[50] Thummerer A, van der Bijl E, Galapon Jr A, Kamp F, Savenije M, Muijs C, et al. SynthRAD2025 Grand Challenge dataset: Generating synthetic CTs for radiotherapy from head to abdomen. Medical Physics 2025 Jul 1;52:e17981.
[51] Fully automated radiotherapy treatment planning: A scan to plan challenge. Radiotherapy and Oncology. 2024 Nov 1;200:110513.
[52] Wang Y, Lombardo E, Thummerer A, Blöcker T, Fan Y, Zhao Y, et al. TrackRAD2025 challenge dataset: real-time tumor tracking for MRI-guided radiotherapy. Med Phys 2025 Jul 1;52:e17964.
[53] Hering A, Hansen L, Mok TCW, Chung ACS, Siebert H, Hager S, et al. Learn2Reg: comprehensive multi-task medical image registration challenge, dataset and evaluation in the era of deep learning. IEEE Trans Med Imaging 2023 Mar;42:697–712.
[54] In-house manufacture of medical devices in Great Britain [Internet]. GOV.UK. [cited 2025 Oct 8]. Available from: https://www.gov.uk/government/publications/in-house-manufacture-of-medical-devices/in-house-manufacture-of-medical-devices.
[55] Guidance on the health institution exemption under Article 5(5) of Regulation (EU) 2017/745 and Regulation (EU) 2017/746 [Internet]. [cited 2025 Oct 8]. Available from: https://health.ec.europa.eu/system/files/2023-01/mdcg_2023-1_en.pdf.
[56] Advice on the production and sharing of Software in a Medical Context, including spreadsheets, scripting and functional documents. [Internet]. [cited 2025 Oct 8]. Available from: https://www.ipem.ac.uk/media/hvaav3fn/advice-note-on-software-in-a-medical-context-feb-2024-final.pdf.
[57] GitHub - jrkerns/awesome-medphys: A list of open-source software tailored for the Medical Physics domain [Internet]. GitHub. [cited 2026 Mar 18]. Available from: https://github.com/jrkerns/awesome-medphys.
[58] Wolff RF, Moons KGM, Riley RD, Whiting PF, Westwood M, Collins GS, et al. PROBAST: a tool to assess the risk of bias and applicability of prediction model studies. Ann Intern Med 2019 Jan 1;170:51–8.
[59] Collins GS, Reitsma JB, Altman DG, Moons KGM. Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis (TRIPOD): the TRIPOD Statement. BMC Med 2015 Jan;6:1.
[60] Website [Internet]. Available from: https://rt-modelcard.streamlit.app/.
[61] Deep learning (nnUNet) model for the EPTN contouring guidelines for OARs in neuro-oncology [Internet]. [cited 2026 Mar 31]. Available from: https://cancerdata.org/publication/deep-learning-nnunet-model-eptn-contouring-guidelines-oars-neuro-oncology.
[62] Sher DJ, Godley A, Park Y, Carpenter C, Nash M, Hesami H, et al. Prospective study of artificial intelligence-based decision support to improve head and neck radiotherapy plan quality. Clin Transl Radiat Oncol 2021 Jul;29:65–70.
[63] Montero B, María A. Standardising AI Model Reporting for Radiotherapy: A Domain-Specific Model Card. [cited 2026 Jan 12]; Available from: https://zenodo.org/records/15336016.