

OpenSim Creator: An Interactive User Interface for Building OpenSim Musculoskeletal Models

Adam Kewley¹✉, Julia van Beesel^{1,2,3}, and Ajay Seth¹

¹ Biomechanical Engineering, TU Delft, Delft, Netherlands ² Department of Development and Regeneration, KU Leuven Campus Kulak, Kortrijk, Belgium ³ Dept. of Human Origins, Max Planck Institute for Evolutionary Anthropology, Leipzig, Germany ✉ Corresponding author

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

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

Editor: [Kevin M. Moerman](#) ✉

Reviewers:

- [@davidpagnon](#)
- [@cvhammond](#)
- [@modenaxe](#)

Submitted: 25 March 2025

Published: unpublished

License

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

Summary

OpenSim Creator is an open-source integrated graphical interface that lowers the barrier to building and evaluating musculoskeletal models in OpenSim, which often required editing XML files or custom code. This paper describes the architecture and features of OpenSim Creator and demonstrates its application in developing new musculoskeletal models.

Statement of Need

OpenSim ([Delp et al., 2007](#); [Seth et al., 2018](#)) is an open-source musculoskeletal modeling platform with a large and growing user base across diverse fields, demonstrated by its use in clinical ([Marconi et al., 2023](#)), biomechanical ([Porsa et al., 2016](#)), robotics ([Beck et al., 2023](#)), and paleobiological ([Beesel et al., 2021](#); [Demuth et al., 2023](#)) studies. However, creating a musculoskeletal model remains labor-intensive and technically demanding because it requires the careful integration of biological data, biomedical images, mesh editing tools, model/data file formats, and musculoskeletal anatomy.

For example, building a biomechanical model of a dinosaur involves 3D scanning fossils, segmenting/registering bones, identifying reference frames and muscle attachments, rigging/articulating skeletal structures, reconstructing musculature, and running biomechanical simulations ([Bishop et al., 2021](#)). Such workflows often need to be repeated for different subjects or specimens. While scaling existing models can reduce this effort, developing robust and unbiased scaling procedures that handle diverse shapes, injuries, or deformities remains challenging ([van der Kruk, 2025](#)); therefore, real-time visual feedback during model building and scaling is a powerful feature for model creation, verification, and validation.

The lack of graphical tools not only affects research but also limits accessibility for students and new researchers learning musculoskeletal modeling. As OpenSim's use expands beyond biomechanics (e.g., medicine, paleobiology), and open education becomes more widespread (e.g., online and open courses ([Weller et al., 2018](#))), there is a growing need for open-source, visual, and easy-to-install tools that educators can use to teach musculoskeletal model-building concepts clearly — tools that lower barriers to entry compared to traditional code- or biomechanics-focused workflows.

State of the Field

Similar model creation and editing challenges are addressed in fields such as 3D design (e.g., Blender ([Blender, 2018](#))), engineering (e.g., FreeCAD ([FreeCAD, 2002](#))), and game development (e.g., Godot ([Godot, 2014](#))). Engineers commonly use these software tools to

reduce labor by automating actions such as 3D object placement. They also enhance user expertise by enabling model editing in a safe, undoable environment with immediate visual feedback.

OpenSim includes a user interface (UI) that can edit the parameters of existing OpenSim models, but it is specialized around tools that accept existing model files as input. For example, it includes tools for scaling models, computing inverse kinematics (IK) from marker motion recordings, and computed muscle control (CMC), but it does not include tools for adding bodies or joints to an existing model. Therefore, to create new models, users typically use a combination of the following three options:

- **Manually write the model file (XML):** The majority of models used by the OpenSim community were created by the development team by manually editing OpenSim model files (.osim files, XML format). It is possible to build any model this way, but it requires knowledge of and expertise with OpenSim's data structures. For average end-users, it is highly error-prone because it does not provide immediate or visual feedback.
- **Use code/libraries to customize a model:** Examples include the MAP Client (Zhang et al., 2014), Neuromusculoskeletal Modeling Pipeline (Hammond et al., 2025), and Muscle Parameter Optimizer (Modenese et al., 2016). These are powerful tools to make existing models more representative of individuals, but they typically do not enable the creation of new models.
- **Use visual tools that can export OpenSim models:** Examples include NMSBuilder (Valente et al., 2017) and MuSkeMo (Bijlert, 2024). These visual tools manipulate a separate data structure exported to OpenSim and do not support direct editing, viewing, or simulation in-UI.

There remains a need for an integrated UI that supports real-time model development, live simulation, and displays key quantities (e.g., muscle moment arms, force vectors) directly from the OpenSim API during editing without export or translation.

Design Principles and Architecture

We surveyed open-source tools (e.g., Blender (2018)) to identify common design and architectural patterns. We found that, in most cases, they focus on providing immediate visual feedback whenever a user performs an action and are continuously improved over many releases. Immediate feedback is typically achieved by integrating the UI directly with the underlying model/state, rather than through indirect mechanisms such as remote procedure calls (RPCs) or export steps. Based on these observations, and our prior experience with OpenSim, we decided that OpenSim Creator's architecture should focus on the following four key requirements:

- **Enable responsive editing performance:** supports interactive model editing and fast visual feedback.
- **Provide low-level UI control:** supports a highly customized user experience. Specialized tasks in biomechanical model building (e.g., landmark placement, mesh reorientation, mesh warping, real-time data visualization) may require low-level control over the rendering process.
- **Directly integrate with the OpenSim API:** provides on-demand access to the OpenSim component API. This enables a wide variety of features, such as the ability to add and edit components in the UI, or show biomechanical quantities (e.g., muscle lengths, moment arms) during editing/simulation.
- **Support iterative software and model development:** facilitates rapid prototyping and feedback from the research community.

To meet these requirements, OpenSim Creator draws inspiration from the games industry, which prioritizes rapid prototyping and real-time rendering. Architecturally, we combine C++ for direct and high-performance OpenSim API integration, a custom OpenGL renderer for

real-time 3D editing, and Dear ImGui (Cornut, 2014) for a customizable low-level 2D UI toolkit. Additionally, real model-building challenges from our own research heavily informed OpenSim Creator's design and architecture, ensuring that practical workflows were supported.

Organizationally, we adopted an open iterative development methodology. All releases of OpenSim Creator have been available on GitHub under an Apache 2.0 license from the first month of development, with new releases every 1–2 months. We have continuously received and incorporated feedback from the research community over 36 release cycles. To further support adoption and educational use, OpenSim Creator includes fully documented tutorials written using Sphinx that guide users with no prior OpenSim experience through model construction and interactive visualization. These resources allow students and new researchers to quickly learn musculoskeletal modeling concepts, explore workflows hands-on, and leverage OpenSim Creator's real-time features.

Features and Scope

Figure 1 and Figure 2 illustrate some of the interactive workflows in OpenSim Creator. While it is challenging to quantify every feature in this article, users benefit from the following capabilities:

- **Live-reload of the underlying .osim file:** OpenSim Creator automatically reloads the model file from disk whenever it detects changes. This enables seamless integration with existing workflows, including manual XML edits, script-based modifications, or edits performed in the OpenSim UI.
- **Live muscle plots:** OpenSim Creator can plot muscle quantities (e.g., moment arm, fiber length) that automatically update whenever the model is edited—either via OpenSim Creator or via external tools. Users can overlay curves from CSV files on these plots, making it easier to create models that match experimental measurements.
- **Output watches:** The OpenSim API provides an “outputs” abstraction, allowing each model component to declare quantities of interest (e.g., a body's center of mass, a joint's reaction force, etc.). OpenSim Creator displays these outputs in real-time during editing and simulation, providing essential insight into model behavior.

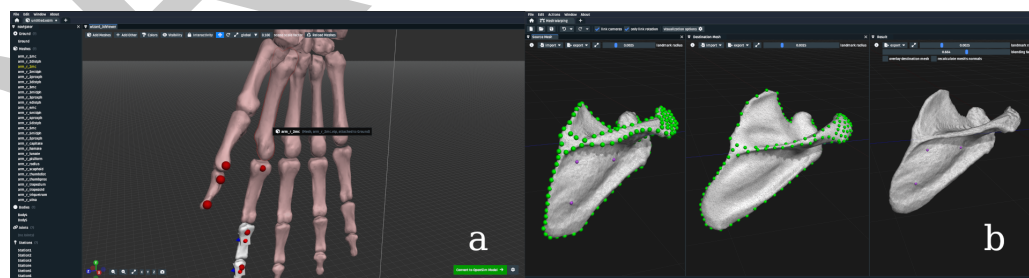


Figure 1: Screenshots of OpenSim Creator's specialized workflows. a) The mesh importer workflow, which focuses on reorienting and rigging meshes in a free-form editor, b) the mesh warper workflow, which focuses on placing corresponding landmarks on meshes (left and middle) with real-time updates to a Thin-Plate Spline (Bookstein, 1989) warped mesh (right).

These features accelerate the model-building process and illustrate the value of OpenSim Creator's software architecture: the direct connection between the OpenSim API and the UI ensures that user interactions immediately update visualizations and model outputs.

These interactive features rely on a highly responsive rendering engine. OpenSim Creator maintains real-time performance, typically running smoothly - even when manipulating complex models on modest hardware (e.g., Rajagopal et al. (2016) on an Intel Core i5 MacBook Air 2020). This responsiveness ensures that live muscle plots, output watches, and mesh

manipulations remain smooth and immediate, allowing users to focus on model construction rather than waiting for visual updates. The tight coupling between the UI and the OpenSim API is key to achieving this performance, avoiding indirect refresh mechanisms such as export/reload cycles or RPCs.

OpenSim Creator is primarily intended for interactive model construction, visualization, and direct inspection of biomechanical quantities. Its UI enables instant feedback during editing, supporting exploration and rapid iteration. Once a model is constructed, users can use the .osim file with other tools or scripts in the OpenSim ecosystem for simulation, analysis, or integration with experimental data. OpenSim Creator facilitates hands-on model construction, while downstream analyses leverage the broader OpenSim ecosystem.

Illustrative Use Cases

Subject-specific models are necessary because all individuals are unique and can have a wide variety of injuries and deformities. In one project, our goal was to create a subject-specific model of a human shoulder. Using OpenSim Creator's mesh importer UI (Figure 1a), a previously published (Seth et al., 2019) generic shoulder model was used as a template by superimposing it over subject-specific CT scans. The UI enabled real-time, free-form placement and orientation of the subject-specific joints and bodies. Muscle volumes from subject-specific MRI data were then combined with moment arm data from the generic model to create a line of action for each muscle in the model. OpenSim Creator's model editor UI provided real-time visual feedback about how each muscle's moment arm was affected by the placement of attachment points and wrapping surfaces on the muscle's line of action within the model (Figure 2).

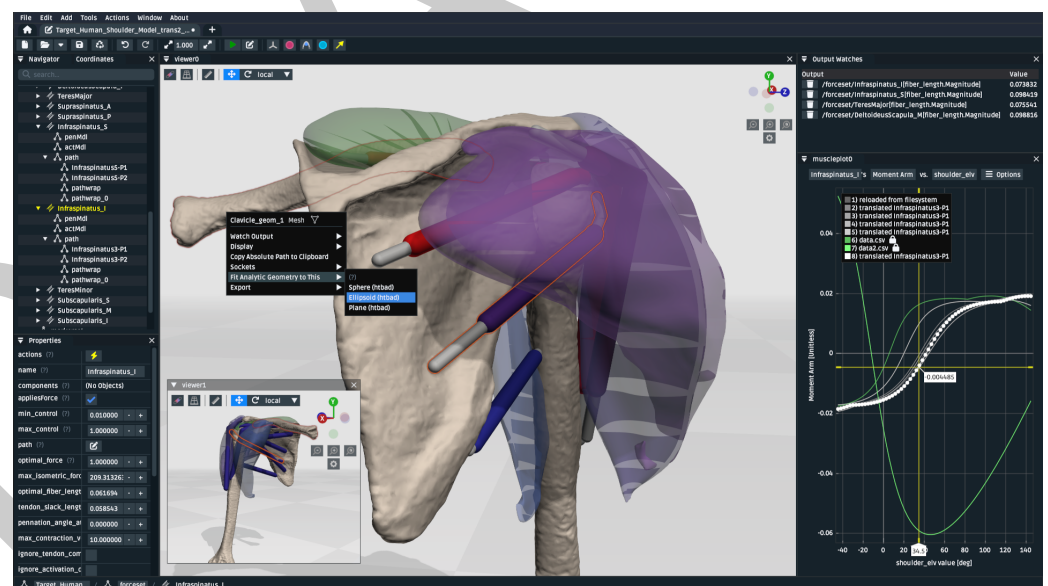


Figure 2: A screenshot of OpenSim Creator's model editor UI, showing a subject-specific human shoulder model created within the editor. The model editor can host multiple panels, each of which is updated whenever the user edits the model. For example, the muscle plotter (right) can be used to show how a muscle's moment arm changes as a user edits the muscle in the 3D viewport (middle) or in the properties panel (left).

Additional use cases that involve OpenSim Creator are available in the literature, such as creating mouse forelimb models (Gilmer et al., 2025), creating female-based lower-extremity models (Gouka, 2025), modeling the effect of foot deformities in children (Heuvel et al., 2023),

148 modeling BMXes (d'Aumerie, 2024), modeling speed skates (Devaraja, 2024), and modeling
149 sit-to-walk movements (Heijerman, 2023).

150 Conclusions

151 Model creation is a complex, multistep, and iterative process that is best served by realistic,
152 fast, and interactive visual feedback. By combining a rapid-prototyping architecture inspired
153 by the games industry with a development approach that continuously incorporates feedback
154 from the biomechanical research community, we designed and developed a graphical UI that
155 enables the creation and evaluation of OpenSim models more quickly and easily.

156 OpenSim Creator's architecture supports rendering multiple high-vertex-count bone models
157 in real time on a MacBook Air (2020, Intel Core i5). Its customizable UI allows researchers
158 to build models visually, and its tight integration with the OpenSim API provides immediate
159 feedback on biomechanical quantities such as muscle moment arms. Anecdotally, master's
160 students using OpenSim Creator in a musculoskeletal modeling course, as well as researchers
161 from the broader community, have reported reduced time and effort to construct, edit, and
162 evaluate models compared to traditional workflows, including XML editing or script-based
163 approaches. These models included complex musculature, multiple joints, and subject-specific
164 anatomical variation, demonstrating that OpenSim Creator can handle realistic research-level
165 scenarios efficiently.

166 Beyond research, OpenSim Creator has proven valuable in educational settings. It has been
167 used in student courses, supported by fully documented tutorials that guide users with no prior
168 OpenSim experience through model construction and visualization. This hands-on approach
169 teaches musculoskeletal modeling concepts while lowering barriers to entry, providing immediate
170 visual feedback, and removing the need for prior coding or engineering expertise.

171 Looking forward, OpenSim Creator will continue to evolve, with plans to support specialized
172 workflows such as visually-assisted model warping, real-time sensor visualization, and the import
173 of diverse biomedical data. These developments, combined with ongoing improvements to the
174 core user experience, will further streamline model creation from medical imaging and emerging
175 anatomical data, making OpenSim Creator a versatile tool for both research and education.

176 Acknowledgements

177 This project was sponsored by grant numbers 2020-218896 and 2022-252796 from the Chan
178 Zuckerberg Initiative DAF, an advised fund of Silicon Valley Community Foundation. It was
179 also sponsored by the Dutch Research Council, NWO XS: OCENW.XS21.4.161. The sponsors'
180 involvement was exclusively financial.

181 We thank DirkJan Veeger and Bart Kaptein for shoulder CT and MRI data for the human
182 model.

183 References

- 184 Beck, I., Belli, I., Peternel, L., Seth, A., & Prendergast, J. M. (2023). Real-time tendon
185 strain estimation of rotator-cuff muscles during robotic-assisted rehabilitation. *2023*
186 *IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*, 1–8. <https://doi.org/10.1109/Humanoids57100.2023.10375158>
187
- 188 Beesel, J. van, Hutchinson, J. R., Hublin, J.-J., & Melillo, S. M. (2021). Exploring the
189 functional morphology of the gorilla shoulder through musculoskeletal modelling. *Journal*
190 *of Anatomy*, 239(1), 207–227. <https://doi.org/10.1111/joa.13412>

- 191 Bijlert, P. A. van. (2024). MuSkeMo: Open-source software to construct, analyze, and
192 visualize human and animal musculoskeletal models and movements in blender. *bioRxiv*.
193 <https://doi.org/10.1101/2024.12.10.627828>
- 194 Bishop, P. J., Cuff, A. R., & Hutchinson, J. R. (2021). How to build a dinosaur: Musculoskeletal
195 modeling and simulation of locomotor biomechanics in extinct animals. *Paleobiology*, 47(1),
196 1–38. <https://doi.org/10.1017/pab.2020.46>
- 197 Blender, O. C. (2018). *Blender - a 3D modelling and rendering package*. Blender Foundation.
198 <http://www.blender.org>
- 199 Bookstein, F. L. (1989). Principal warps: Thin-plate splines and the decomposition of
200 deformations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(6),
201 567–585. <https://doi.org/10.1109/34.24792>
- 202 Cornut, O. (2014). *Dear ImGui: Bloat-free graphical user interface for c++ with minimal*
203 *dependencies*. <https://github.com/ocornut/imgui>
- 204 d'Aumerie, J. C. (2024). *Optimal movement and timing in BMX starts*. <https://resolver.tudelft.nl/uuid:e5114387-2f1f-4163-a5ca-b149100fad05>
- 206 Delp, S. L., Anderson, F. C., Arnold, A. S., Loan, P., Habib, A., John, C. T., Guendelman,
207 E., & Thelen, D. G. (2007). OpenSim: Open-source software to create and analyze
208 dynamic simulations of movement. *IEEE Transactions on Biomedical Engineering*, 54(11),
209 1940–1950. <https://doi.org/10.1109/TBME.2007.901024>
- 210 Demuth, O. E., Wiseman, A. L. A., & Hutchinson, J. R. (2023). Quantitative biomechanical
211 assessment of locomotor capabilities of the stem archosaur euparkeria capensis. *Royal*
212 *Society Open Science*, 10(1), 221195. <https://doi.org/10.1098/rsos.221195>
- 213 Devaraja, P. (2024). *ODAH-SpeedSkater*. <https://resolver.tudelft.nl/uuid:fbe7a517-5fb8-4d86-b8d8-9661>
- 214 FreeCAD, O. C. (2002). *FreeCAD: Your own 3D parametric modeler*. FreeCAD Project
215 Association. <https://www.freecad.org/>
- 216 Gilmer, J. I., Coltman, S. K., Cuenu, G., Hutchinson, J. R., Huber, D., Person, A. L., &
217 Al Borno, M. (2025). A novel biomechanical model of the proximal mouse forelimb
218 predicts muscle activity in optimal control simulations of reaching movements. *Journal of*
219 *Neurophysiology*, 133(4), 1266–1278. <https://doi.org/10.1152/jn.00499.2024>
- 220 Godot, O. C. (2014). *Godot: A free, open-source game engine*. Godot Foundation. <https://godotengine.org/>
- 222 Gouka, Y. (2025). *Development of a female-based musculoskeletal model of the lower extremity*.
223 <https://resolver.tudelft.nl/uuid:cdeecb96-a27d-4493-8c10-f669e6cca2d1>
- 224 Hammond, C. V., Williams, S. T., Vega, M. M., Ao, D., Li, G., Salati, R. M., Pariser, K. M.,
225 Shourijeh, M. S., Habib, A. W., Patten, C., & others. (2025). The neuromusculoskeletal
226 modeling pipeline: MATLAB-based model personalization and treatment optimization
227 functionality for OpenSim. *Journal of NeuroEngineering and Rehabilitation*, 22(1), 112.
228 <https://doi.org/10.1186/s12984-025-01629-5>
- 229 Heijerman, S. (2023). *Addition of torso musculature to a simplified musculoskeletal model*
230 *for 2D predictive simulation of the sit-to-walk movement*. <http://resolver.tudelft.nl/uuid:91925ea8-199e-4304-8b8f-8f52519a7931>
- 232 Heuvel, G. V. D., Schallig, W., van der Krogt, M., Wellenberg, R., Maas, M., Buizer, A., &
233 Seth, A. (2023). The effect of varus foot deformities on muscle moment arms in children
234 with cerebral palsy. *Gait & Posture*, 106, S212–S213. <https://doi.org/10.1016/j.gaitpost.2023.07.256>
- 236 Marconi, G., Gopalai, A. A., & Chauhan, S. (2023). Effects of powered ankle-foot orthoses
237 mass distribution on lower limb muscle forces—a simulation study. *Medical & Biological En-*

- 238 *gineering & Computing*, 61(5), 1167–1182. <https://doi.org/10.1007/s11517-023-02778-2>
- 239 Modenese, L., Ceseracciu, E., Reggiani, M., & Lloyd, D. G. (2016). Estimation of musculoten-
240 don parameters for scaled and subject specific musculoskeletal models using an optimization
241 technique. *Journal of Biomechanics*, 49(2), 141–148. [https://doi.org/10.1016/j.jbiomech.](https://doi.org/10.1016/j.jbiomech.2015.11.006)
242 [2015.11.006](https://doi.org/10.1016/j.jbiomech.2015.11.006)
- 243 Porsa, S., Lin, Y.-C., & Pandy, M. G. (2016). Direct methods for predicting movement
244 biomechanics based upon optimal control theory with implementation in OpenSim. *Annals*
245 *of Biomedical Engineering*, 44(8), 2542–2557. <https://doi.org/10.1007/s10439-015-1538-6>
- 246 Rajagopal, A., Dembia, C. L., DeMers, M. S., Delp, D. D., Hicks, J. L., & Delp, S. L.
247 (2016). Full-body musculoskeletal model for muscle-driven simulation of human gait. *IEEE*
248 *Transactions on Biomedical Engineering*, 63(10), 2068–2079. [https://doi.org/10.1109/](https://doi.org/10.1109/TBME.2016.2586891)
249 [TBME.2016.2586891](https://doi.org/10.1109/TBME.2016.2586891)
- 250 Seth, A., Dong, M., Matias, R., & Delp, S. (2019). Muscle contributions to upper-extremity
251 movement and work from a musculoskeletal model of the human shoulder. *Frontiers in*
252 *Neurorobotics*, 13. <https://doi.org/10.3389/fnbot.2019.00090>
- 253 Seth, A., Hicks, J. L., Uchida, T. K., Habib, A., Dembia, C. L., Dunne, J. J., Ong, C. F.,
254 DeMers, M. S., Rajagopal, A., Millard, M., Hamner, S. R., Arnold, E. M., Yong, J. R.,
255 Lakshmikanth, S. K., Sherman, M. A., Ku, J. P., & Delp, S. L. (2018). OpenSim: Simulating
256 musculoskeletal dynamics and neuromuscular control to study human and animal movement.
257 *PLOS Computational Biology*, 14(7), 1–20. <https://doi.org/10.1371/journal.pcbi.1006223>
- 258 Valente, G., Crimi, G., Vanella, N., Schileo, E., & Taddei, F. (2017). nmsBuilder: Freeware
259 to create subject-specific musculoskeletal models for OpenSim. *Computer Methods and*
260 *Programs in Biomedicine*, 152, 85–92. <https://doi.org/10.1016/j.cmpb.2017.09.012>
- 261 van der Kruk, E. (2025). BIASMECHANICS: Does an unconscious bias still persist in
262 biomechanics, positioning males as the default in human research? A meta-analysis on
263 the journal of biomechanics 2024 publications. *Journal of Biomechanics*, 181, 112560.
264 <https://doi.org/10.1016/j.jbiomech.2025.112560>
- 265 Weller, M., Jordan, K., DeVries, I., & Rolfe, V. (2018). Mapping the open education landscape:
266 Citation network analysis of historical open and distance education research. *Open Praxis*.
267 <https://doi.org/10.5944/openpraxis.10.2.822>
- 268 Zhang, J., Sorby, H., Clement, J., Thomas, C. D. L., Hunter, P., Nielsen, P., Lloyd, D.,
269 Taylor, M., & Besier, T. (2014). *The MAP client: User-friendly musculoskeletal modelling*
270 *workflows*. 182–192. https://doi.org/10.1007/978-3-319-12057-7_21