Applying Software Quality in Use Standards to Improve Scientific Software Selection

Yvette D. Hastings, Ann Marie Reinhold
Gianforte School of Computing
Montana State University
Bozeman, MT, USA
yvettehastings@msu.montana.edu, reinhold@montana.edu

***Abstract*—Across scientific domains, researchers are challenged by the process of selecting suitable modeling software. These challenges are particularly numerous in the earth sciences, arising from selecting software based on a few of the myriad complex earth system processes and wide availability of modeling software. Earth scientists lack a framework to guide scientific software selection. In this paper, we operationalize a framework based on the Quality in Use Model, as codified by the International Organization for Standardization (ISO) 25010 standard, to identify and create metrics to assess software that is used to simulate a subset of earth science processes known as soil processes. We applied this framework to assess software for three highly cited soil process models: Community Land Model (CLM), Decision Support System for Agrotechnology Transfer (DSSAT), and HYDRUS-1D. DSSAT scored the highest for the Quality in Use Model metrics, followed by HYDRUS-1D and CLM. This study is the first of its kind to apply the ISO 25010 Product Quality Model to a class of modeling software in the earth sciences, and its application shows promise for streamlining software selection.**

Keywords —ISO 25010: Quality in Use Model; software quality; soil process models; CLM; DSSAT; HYDRUS-1D

I. Introduction

End users in the earth sciences (here, earth-science end users (ESEUs)) are challenged by the process of selecting modeling software to meet their research needs. This is especially true for soil process models (SPMs), as they are foundational in many earth science research projects that investigate a wide array of processes in earth systems, including the effects of climate change and agriculture on the health and sustainability of soil and water systems [1]–[3]. While SPM software is widely available, SPMs have become computationally complex and mathematically complicated, as scientific software designers have attempted to keep step with advances in empirical research [2], [4]–[6].

These complexities present researchers with the challenge of determining which scientific modeling software best enables them to address their objectives efficiently and effectively. When selecting software, ESEUs must consider whether the SPM has the suitable spatio-temporal scales, whether the underlying mathematical algorithms are appropriate for their analysis, and whether the SPM is regularly maintained and updated with advancing scientific knowledge [1], [3], [4]. Further questions arise as to whether the ESEU has enough knowledge of how to use the modeling software (e.g., if the software is in a programming shell (e.g., Python or Java) or has a graphical user interface (GUI)) [7]. This knowledge impacts ESEU's experience with the software and can impact research results. For example, poor GUI design can result in ESEUs incorrectly setting up models or interpreting results incorrectly; in both cases, the validity and verification of simulations are seriously hampered.

ESEUs have an active forum on ResearchGate.net to assist one another with software selection [8], underscoring the multifaceted challenges associated with selecting and running SPM software. While literature review and peer commentary (e.g., forums) can provide useful information, ESEUs may still deliberate whether their chosen SPM software best addresses their research objectives or if another modeling software would have been a superior choice. Hence, we assert that ESEUs would benefit from considering a wide range of software-quality aspects when selecting modeling software.

The purpose of this paper is to operationalize a framework to evaluate the quality in use of SPM software to aid ESEUs in software selection. We identify aspects of the Quality in Use (QIU) Model, as codified and described, in the International Organization for Standardization (ISO) 25010 as a starting point for ESEUs to improve the efficiency of selecting, parameterizing, executing, and validating SPMs [9]. The Quality in Use Model identifies and defines five main components, with nine subcomponents, that software products aim to meet to allow users to meet their project goals [9]. Within each of these components and subcomponents, we developed metrics that can be applied to SPM software. We then used these criteria and metrics to evaluate three of the most utilized SPMs in Vereecken *et al.* [3]. Keeping the end user in mind, we also evaluate the software based on common considerations soil process modelers have.

II. Methods

We conducted a systematic search for studies that use software to generate SPMs; the suite of potential software was constrained to those that generate one of the SPMs listed in Table 2 in [3] within the categories of water cycling, nutrient cycling, biological activity, salinization, buffering and filtering, recycling of waters, and biomass production for food, fiber and energy. We searched Google Scholar to determine which of these SPM software had the greatest number of citations [19]–[21]. We evaluated the top three: Community Land Model (CLM; 25,000 citations), Decision Support System for Agrotechnology Transfer (DSSAT; 17,900 citations), and HYDRUS-1D (11,400 citations).

Below, we use the term "basic simulation" to describe the simple models that we executed in CLM, DSSAT, and HYDRUS-1D. Although each software creates SPMs, we could not identify a use case where we could simulate the same environmental system with the same parameters across all software. The reason for this is that differences among the intended use, scope, and parameters of the scientific modeling software prevented a direct comparison of resulting models. All simulations were run to completion without warning or error.

- CLM is a component of the Community Earth System Model (CESM) v2.1.3. Required software that is listed in the CESM2.1.z Quickstart Guide was installed on an Ubuntu virtual machine and the CESM v2.1.3 GitHub repo was cloned [10], [11]. Data stored on the GLADE drive of the ARC NCAR/UCAR Cheyenne server[1] was used for a basic simulation. This simulation was performed following the instructions in the CESM Quickstart Guide and from an online CESM tutorial [10], [12]. The only modification from the tutorial was to change the project identifier between building and submitting the model. This was not a step in the tutorial but was required to run the model.
- DSSAT v4.8.0.0 and associated GUI were downloaded and installed in accordance with the developer's instructions [13]–[16]. The 'Introductory Simulation', found in the 'Accessories' tab of the user interface, was followed to perform a basic simulation. We provided values for the required input fields for the test simulation.
- HYDRUS-1D v4.17.0140 and associated GUI were downloaded and installed from the developer's website [17]. Tutorials 1 and 3, available at [18], were used to create a basic simulation. Instructions and data for the tutorials, where applicable, were included on their respective webpages.

A. *ISO 25010: Quality in Use*

CLM, DSSAT, and HYDRUS-1D were evaluated using the five components, nine subcomponents, and provided definitions as codified in the Quality in Use Model of [9]. In accordance with implementing the QIU Model, our research group—composed of software engineers and ESEUs—held brainstorming sessions over multiple days to critically evaluate and identify measurable metrics to quantify each component and subcomponent based on an ESEU perspective. Our scoring scale was set from 1-10, with 1 being the lowest score and 10 being the highest score possible. We used this scale to keep score ranges universal for all components/subcomponents.

Components/subcomponents that assess a range of values (e.g., costs) were broken up into equal intervals bounded between 1 and 10 using one of the following equations:

$$y = \frac{x}{n} * 10, \quad (1)$$

$$y = 1 + \left(\frac{n-x}{n}\right) * 10, \quad (2)$$

where y is the metric score, n is the highest possible boundary value, and x is a value between $\frac{n}{10}$ and n. Components/subcomponents measured with a Boolean received a score of 10 for "true" and 1 for "false"; where appropriate, components/subcomponents not meeting a true or false answer received intermediate scores. Higher scores indicate higher QIU.

We created metrics for each component/subcomponent of the ISO 25010 QIU model (Fig. 1) as follows.

- **Effectiveness**: SPMs need to incorporate an array of processes interacting in the environment. We have listed three critical processes ESEUs must simulate. Using Eqn.1, the highest possible boundary value (n) is 3, where an x of 3 received a score of 10 and an x of 1 received a score of 3.33.
- **Efficiency**: **Costs** – Many government agencies and academic institutions use SPMs to study the environment. Often, funds to support software licensing are limited. SPM software we evaluated are free, but many others may require licensing fees. Using Eqn. 2, where n is \$2,000 USD, licenses costing ≥ \$2,000 USD received a score of 1 on our scoring scale, and licenses costing ≤ \$200 USD received a score of 10. **Time** – High memory and CPU usage hinder the productivity of ESEUs. Using Eqn. 2, where n is 100%, CPU usage > 90% received a score of 1 and CPU usage ≤ 10% received a score of 10. CPU and memory usage are measured on the same scale. **Materials** – Software that requires multiple programs and dependencies to run reduces user experience, as these are often cumbersome and tedious to install. Using Eqn. 2, where n is 10, software requiring ESEUs to manually install ≥ 10 dependencies to run an SPM received a score of 1, and software requiring ≤ 1 dependency to be manually installed received a score of 10.
- **Satisfaction**: **Usefulness** – Not all ESEUs are familiar with a terminal window. Having a GUI can often improve user experience. **Trust** – Software that has been validated and verified improves user trust. **Pleasure** – Having available tutorials that follow model flow improves user experience and learnability of the software. **Comfort** – We identified two metrics to test this subcomponent. Metric 1 assesses if the software requires constant user input during a long model run, thus potentially reducing user experience. Metric 2 looks at how many commands/steps a user is required to go through from start to finish (i.e., parameterization to analysis). Metrics for usefulness, trust, pleasure, and the first comfort subcomponents were measured with a Boolean. For the second comfort metric, using Eqn. 1, where n is 5, software requiring ≥ 5 steps received a score of 2 and software requiring ≤ 1 step received a score of 10.
- **Freedom From Risk**: **Economic Risk Mitigation** – Cyber-attacks are costly. Having secure servers to run models and store data reduces cyber threats. **Health and Safety Risk Mitigation** – No obvious test metric exists to assess this subcomponent. Therefore, this is not included in our evaluation and is denoted as N/A in Fig. 1. **Environmental Risk Mitigation** – High energy use in the home/office is a risk to the environment. Here, we did not assess economic and environmental risk mitigation subcomponents, but they merit follow-up in future work.

[1]https://arc.ucar.edu/knowledge_base/70549542

This material is based upon work supported by the National Science Foundation under the following Grant Numbers: SitS CBET-2034430 and EPSCoR Cooperative Agreement OIA-1757351. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

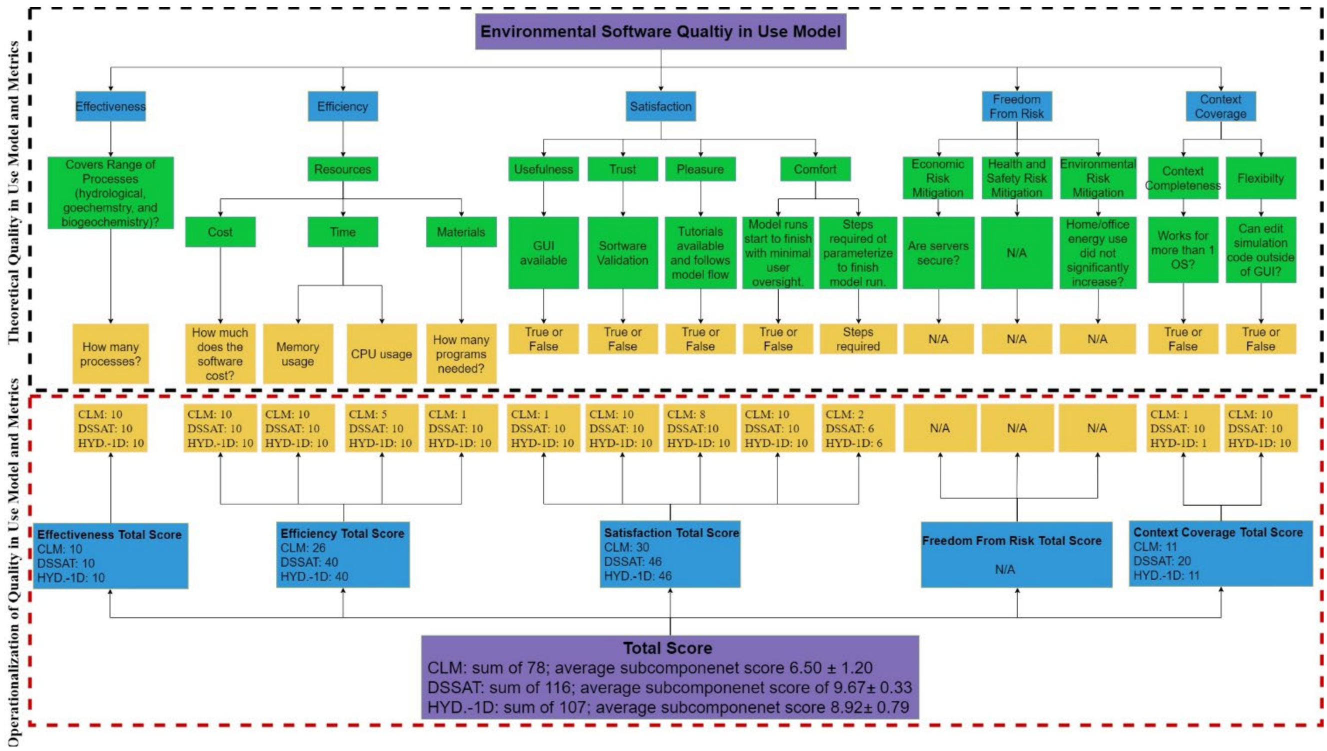

Figure 1. Quality in Use (QIU) Framework developed from the ISO 25010 Software Quality in Use Model. QIU Framework was used to evaluate the three SPM software: CLM, DSSAT, and HYDRUS-1D (abbreviated as HYD.-1D). Black-dashed box: the theoretical QIU Framework and metrics; dashed red box: operationalization of the theoretical QIU model and metrics; purple boxes: environmental software QIU model and operationalization scores; blue boxes: QIU components; green boxes: QIU subcomponents; yellow boxes: QIU metrics. Arrows indicate the flow of information.

- **Context Coverage**: **Context Completeness** – Software that runs on multiple operating systems (OSs) allows users the freedom to use computer environments (i.e., "contexts") that are familiar to them. **Flexibility** – ESEUs perform research covering a wide array of scientific theories. Code that is modifiable allows ESEUs to add/update algorithms that meet their research goals. Context completeness and flexibility are measured with a Boolean.

B. *ESEU Considerations*

CLM, DSSAT, and HYDRUS-1D were assessed to determine which ESEU primary considerations were met by each scientific modeling software. These common considerations overlap with certain components of the QIU Model; e.g., W*hich operating system is required to run the software? Is the software easy to learn and use, with ample tutorials? To what extent is run time a consideration? Can end users modify the simulation parameters using code (rather than a GUI) if needed?* In addition, ESEUs also must consider how models created by the software represent the earth system of interest; e.g., *How are hydrological processes simulated? Which chemical processes can be simulated and how so (kinetics, thermodynamics, etc.)? Is this software appropriately verified and validated for an environmental system of interest?*

III. Results & Discussion

Software popularity was not directly related to QIU scores. Although CLM had the most citations, DSSAT had the highest QIU score, with a total score of 116 points and an average subcomponent score of 9.67 ± 0.33 points (standard error). HYDRUS-1D received the second highest score in our QIU scoring index, with a total score of 107 points and an average subcomponent score of 8.92 ± 0.79 points. CLM received the lowest score of the three software, with a total score of 78 points and an average subcomponent score of 6.50 ± 1.20 points. These findings indicate DSSAT provides ESEUs with greater user experience related to ease of use and flexibility of the modeling software.

Comparison of common considerations of ESEUs across CLM, DSSAT, and HYDRUS-1D indicated that DSSAT had the greatest coverage (Table 1). DSSAT had 15 of the 18 assessed considerations listed, while HYDRUS-1D and CLM each had 13 of the 18 assessed considerations. What set DSSAT apart was that DSSAT can run on multiple OSs and platforms. Further, DSSAT is not limited to its GUI. ESEUs can run DSSAT in R and Python platforms [22], both of which can be operated in Windows and UNIX-like operating systems; neither CLM nor HYDRUS-1D had this option. Although DSSAT had the highest QIU scores (Fig. 1) and addresses the greatest number of ESEU considerations (Table 1), it is intended primarily for simulating agricultural systems. Consequently, it requires crop systems to be identified and associated parameter values to be inputted. Conversely, CLM and HYDRUS-1D are not limited to crop systems. This highlights the importance of identifying the intended purpose and scope of modeling software. Although intended purpose and scope are not explicitly part of the QIU framework presented here, they are related to the component of Effectiveness. While this could be viewed as a limitation of our operationalization of the QIU

Table 1. Common considerations for ESEUs. Software evaluated were CLM, DSSAT, and HYDRUS-1D (HYD-1D). Information is sourced from user manuals and software websites.

ESEU Consideration	CLM	DSSAT	HYD-1D
Operating Systems			
UNIX-like	x	x	-
Windows	-	x	x
Hypothesis Testing	x	x	x
Hydrological Mathematics			
Richard's Equation	x	-	x
Fickian Advection/Dispersion	-	-	x
Tipping Bucket	-	x	-
Reactive Soil Processes			
Biogeochemistry	x	x	x
Geochemistry	x	x	x
Modifiable Simulation Code	x	x	-
Time to Run Model	*	*	*
Spatial Scale			
Fine (Soil Column)	-	x	x
Moderate (Plot to Farm)	x	x	x
Coarse (Watershed to Global)	x	-	-
Parameterization Data			
Manual Entry	-	x	x
Data Upload	x	x	-
Learnability			
Tutorials	x	x	x
User Forums	x	x	x
Developer Contact	x	x	x
Validation	x	x	x
Correct Equations/Theories	*	*	*

*Not assessed in current study.

framework, this qualitative information was not readily quantifiable but was available on software websites, user manuals, and with minimal literature review.

The strength of our approach is in bridging between the computer sciences and earth sciences to operationalize a QIU framework to improve the software selection process for ESEUs. Our future work will incorporate the Product Quality Model codified by [9] into a similar framework, as well as comparing SPMs with common intended use and purposes through the lens of both QIU and Software Product Quality. Additionally, while the current investigation pertains to scientific modeling software in the earth sciences, the framework holds promise for assessing QIU in scientific software in other scientific disciplines and domains.

Acknowledgments

We thank Drs. S.A. Ewing, C. Izurieta, R.A. Payn, D. Reimanis, and S. Warnat for their collaboration, and the NCAR/UCAR support team for account support and server access.

References

[1] L. Li *et al.*, "Expanding the role of reactive transport models in critical zone processes," *Earth-Science Reviews*, vol. 165. Elsevier B.V., pp. 280–301, Feb. 01, 2017. doi: 10.1016/j.earscirev.2016.09.001.

[2] C. I. Steefel, D. J. DePaolo, and P. C. Lichtner, "Reactive transport modeling: An essential tool and a new research approach for the Earth sciences," *Earth Planet Sci Lett*, vol. 240, no. 3–4, pp. 539–558, Dec. 2005, doi: 10.1016/j.epsl.2005.09.017.

[3] H. Vereecken *et al.*, "Modeling Soil Processes: Review, Key Challenges, and New Perspectives," *Vadose Zone Journal*, vol. 15, no. 5, p. vzj2015.09.0131, May 2016, doi: 10.2136/vzj2015.09.0131.

[4] J. Carrera, M. W. Saaltink, J. Soler-Sagarra, W. Jingjing, and C. Valhondo, "Reactive Transport: A Review of Basic Concepts with Emphasis on Biochemical Processes," *Energies (Basel)*, vol. 15, no. 3, Feb. 2022, doi: 10.3390/en15030925.

[5] C. I. Steefel, S. B. Yabusaki, and K. U. Mayer, "Reactive transport benchmarks for subsurface environmental simulation," *Computational Geosciences*, vol. 19, no. 3. Kluwer Academic Publishers, pp. 439–443, Jun. 27, 2015. doi: 10.1007/s10596-015-9499-2.

[6] D. Wang *et al.*, "A scientific function test framework for modular environmental model development: Application to the community land model," in *Proceedings - 2015 International Workshop on Software Engineering for High Performance Computing in Science, SE4HPCS 2015*, Institute of Electrical and Electronics Engineers Inc., Jul. 2015, pp. 16–23. doi: 10.1109/SE4HPCS.2015.10.

[7] F. J. R. Meysman, J. J. Middelburg, P. M. J. Herman, and C. H. R. Heip, "Reactive transport in surface sediments. I. Model complexity and software quality," *Comput Geosci*, vol. 29, no. 3, pp. 291–300, 2003, doi: 10.1016/S0098-3004(03)00006-2.

[8] "How to choose reactive transport modeling software." ResearchGate.net. https://www.researchgate.net/post/How_to_choose_Reactive_Transport_Modeling_software (accessed May 05, 2023).

[9] *Systems and software engineering-Systems and software Quality Requirements and Evaluation (SQuaRE)-System and software quality models*, ISO/IEC FDIS 25010:2010(E), International Organization for Standardization, Geneva, CH, 2010.

[10] National Center for Atmospheric Research. "CESM Quickstart Guide (CESM2.1)." CESM2. https://escomp.github.io/CESM/versions/cesm2.1/html/ (accessed Jun. 04, 2023).

[11] *CESM*. (v2.1.3), National Center for Atmospheric Research. Accessed: June 8, 2023. [Online]. Available: https://www.cesm.ucar.edu/models

[12] National Center for Atmospheric Research. "Welcome to the CESM Tutorial." https://ncar.github.io/CESM-Tutorial/README.html (accessed Jun. 25, 2023).

[13] J. W. Jones *et al.*, "DSSAT Cropping System Model," *European Journal of Agronomy*, vol. 18, pp. 235–265, 2003, doi: 10.1016/S1161-0301(02)00107-7.

[14] G. Hoogenboom *et al.*, "Advances in crop modeling for a sustainable agriculture," *The DSSAT crop modeling ecosystem*, pp. 173–216, 2019, doi: 10.19103/AS.2019.0061.10.

[15] G. Hoogenboom *et al.*, "Decision Support System for Agrotechnology Transfer (DSSAT) Version 4.8," 2021. DSSAT.net

[16] DSSAT. (v4.8.0.0), DSSAT Foundation, Inc. Accessed: June 8, 2023. [Online]. Available: dssat.net

[17] *HYDRUS-1D*. (v4.17.0140), PC-Progress s.r.o. Accessed: May 31, 2023. [Online]. Available: https://www.pc-progress.com/en/Default.aspx?H1d-downloads

[18] "Hydrus-1D Tutorial Book." PC-Progress. https://www.pc-progress.com/en/Default.aspx?h1d-tutorials (accessed May 31, 2023).

[19] "CLM Google Scholar Results." Internet Archive WaybackMachine. https://web.archive.org/web/20230523152409/https://scholar.google.com/scholar?hl=en&as_sdt=0%2C27&q=Community+Land+Model+%28CLM%29&btnG= (accessed May 23, 2023).

[20] "DSSAT Google Scholar Results." Internet Archive WaybackMachine. https://web.archive.org/web/20230523151936/https://scholar.google.com/scholar?hl=en&as_sdt=0%2C27&q=%22DSSAT%22&btnG= (accessed May 23, 2023).

[21] "HYDRUS 1D Google Scholar Results." Internet Archive WaybackMachine. . https://web.archive.org/web/20230523153823/https://scholar.google.com/scholar?hl=en&as_sdt=0%2C27&q=%22HYDRUS+1D%22&btnG= (accessed May 23, 2023).

[22] P. D. Alderman, "A comprehensive R interface for the DSSAT Cropping Systems Model," *Comput Electron Agric*, vol. 172, May 2020, doi: 10.1016/j.compag.2020.105325.