Webstrates, Codestrates v2, and Varv: A Software Stack for Computational Media

Marcel Borowski
marcel.borowski@cs.au.dk
Aarhus University
Aarhus, Denmark

Clemens N. Klokmose
clemens@cs.au.dk
Aarhus University
Aarhus, Denmark

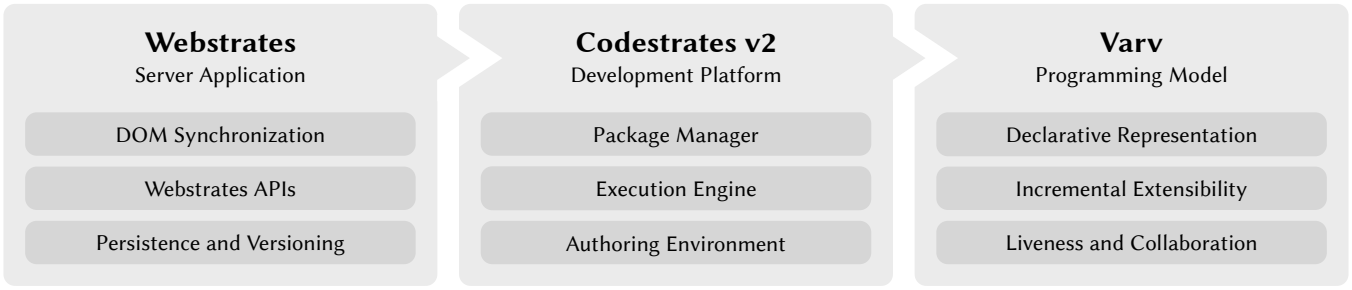

Figure 1: Overview of the three components of the software stack.

ABSTRACT

Today's software often comes in the form of applications: rigid turn-key products, which usually cannot be modified by their users. Computational media is an alternative vision of software that is inherently extensible and collaborative. With the Webstrates platform, we started to explore computational media almost a decade ago. Recently, we added the Codestrates v2 development platform and the Varv programming model to Webstrates. Together the three components enable the creation of inherently extensible and collaborative software on the web. In this demonstration, we introduce these three components and showcase their potential in a series of examples.

CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**; *Web-based interaction*; *Collaborative interaction.*

KEYWORDS

Computational Media, Webstrates, Codestrates, Malleability, Declarative Programming

ACM Reference Format:
Marcel Borowski and Clemens N. Klokmose. 2022. Webstrates, Codestrates v2, and Varv: A Software Stack for Computational Media. In *Adjunct Proceedings of the 2022 Nordic Human-Computer Interaction Conference (NordiCHI Adjunct '22), October 8–12, 2022, Aarhus, Denmark.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3547522.3547714

NordiCHI Adjunct '22, October 8–12, 2022, Aarhus, Denmark

ACM ISBN 978-1-4503-9448-2/22/10.
https://doi.org/10.1145/3547522.3547714

1 INTRODUCTION

Software is often synonymous with applications and designed as turn-key and one-size-fits-all products. In this application-centric model, users are mere consumers of software and have little to no control over the functionality of their software. Even as a trained programmer it requires large efforts to modify one's applications — often, modifications are even entirely prohibited by software vendors. Computational media is an alternative model of software that emphasizes modification and combination of software tools. With computational media, users are not mere consumers but can actively modify their software and reclaim control over it.

Over the last ten years, we have explored the potentials and challenges of the computational media vision in a variety of projects and gradually developed a software stack to support creating prototypes that adhere to this vision. The foundation is the web-based Webstrates [9] platform. Later, we developed authoring tools with the Codestrates v1 and v2 platforms [2, 14] and recently the Varv programming model [4]. Together these three platforms act as a technological foundation and software stack for creating computational media prototypes. In this demonstration, we summarize the three platforms Webstrates, Codestrates v2, and Varv and showcase their potentials in a series of examples.

2 THE SOFTWARE STACK

Our software stack for computational media consists of three components: Webstrates, Codestrates v2, and Varv. Each of these platforms builds on top of each other and enables the creation of malleable and collaborative software on the web. All three platforms are open source and available on GitHub.

Webstrates[1] [9] is a data synchronization and persistence layer. It is a web server application that serves websites — so-called *webstrates* — where the document object model (DOM) is synchronized and persisted on the Webstrates server. This means that all changes

[1] Webstrates: https://webstrates.net/ (Retrieved July 14, 2022)

to the DOM, e.g., using the developer tools of a web browser, are synchronized in real-time across clients. This includes changes to JavaScript and CSS code.

Codestrates v2[2] [2] acts as a development platform for Webstrates and consists of an execution engine, the Cauldron authoring environment, and a package manager. The execution engine of Codestrates v2 makes it possible to edit and execute code fragments from within a webstrate. A code fragment is piece of code that can be executed and edited individually. Cauldron is an authoring environment that provides a tree-browser for code fragments and a tabbed editor to edit them. The package manager, lastly, enables sharing and distribution of code between webstrates and bundling of functionality. With the Codestrates v2 platform it is, e.g., trivial to implement a computational notebook on top of Webstrates.[3]

Varv[4] [4], finally, adds a programming model for computational media, which allows for inherently and incrementally extensible software. Varv represents reprogrammable interactive software as a declarative data structure. Interactive behavior is defined as a set of concepts consisting of a schema and actions. These make it possible to incrementally add, modify, or suppress interactive behavior through the addition of new code. Varv concepts can be authored in Cauldron and additional tooling for Cauldron makes it possible to inspect the view and state of an application created with Varv.

3 A STEP TOWARDS SOFTWARE AS COMPUTATIONAL MEDIA

Throughout the years, we and others have used Webstrates, Codestrates v1, and Codestrates v2 to explore computational media in a variety of projects including collaborative data visualization [1, 8, 12], public libraries [6, 15], collaborative video editing [10], collaborative programming assignments [5], computational laboratory notebooks [13], collaborative writing tools [3], affinity diagramming [11], and hybrid meetings [7].

Our software stack enables to create inherently extensible and collaborative software. In future work, we plan to also explore using Varv as a foundation for interactive software. More work, however, is still needed towards lowering the threshold for modifying software for users without programming knowledge. We hope that with this demonstration we motivate others to employ the presented software stack or parts of it as a technological foundation for prototypes and to join us in exploring software based on computational media.

ACKNOWLEDGMENTS

We want to thank all co-authors and collaborators on the presented projects. This work has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 740548) and from Carlsbergfondet (grant agreement No CF17-0643).

[2]Codestrates v2: https://codestrates.projects.cavi.au.dk/ (Retrieved July 14, 2022)
[3]Codestrates v2 Examples: https://codestrates.projects.cavi.au.dk/docs/examples/ (Retrieved July 14, 2022)
[4]Varv: https://varv.projects.cavi.au.dk/ (Retrieved July 14, 2022)

REFERENCES

[1] Sriram Karthik Badam, Andreas Mathisen, Roman Rädle, Clemens N. Klokmose, and Niklas Elmqvist. 2018. Vistrates: A Component Model for Ubiquitous Analytics. *IEEE Transactions on Visualization and Computer Graphics* (2018), 586–596. https://doi.org/10.1109/TVCG.2018.2865144

[2] Marcel Borowski, Janus Bager Kristensen, Rolf Bagge, and Clemens N. Klokmose. 2021. *Codestrates v2: A Development Platform for Webstrates.* Technical Report. Aarhus University. https://pure.au.dk/portal/en/publications/codestrates-v2-a-development-platform-for-webstrates(66e1d4d9-27da-4f6b-85b3-19b0993caf22).html

[3] Marcel Borowski and Ida Larsen-Ledet. 2021. Lessons Learned From Using Reprogrammable Prototypes With End-User Developers. In *End-User Development (IS-EUD '21)*. Springer. https://doi.org/10.1007/978-3-030-79840-6_9

[4] Marcel Borowski, Luke Murray, Rolf Bagge, Janus Bager Kristensen, Arvind Satyanarayan, and Clemens N. Klokmose. 2022. Varv: Reprogrammable Interactive Software as a Declarative Data Structure. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. ACM. https://doi.org/10.1145/3491102.3502064

[5] Marcel Borowski, Johannes Zagermann, Clemens N. Klokmose, Harald Reiterer, and Roman Rädle. 2020. Exploring the Benefits and Barriers of Using Computational Notebooks for Collaborative Programming Assignments. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education (SIGCSE '20)*. ACM. https://doi.org/10.1145/3328778.3366887

[6] Carla Gröschel, Peter Dalsgaard, Clemens N. Klokmose, Henrik Korsgaard, Eva Eriksson, Raphaëlle Bats, Aurélien Tabard, Alix Ducros, and Sofia E. Serholt. 2018. PARTICIPATE: Capturing Knowledge in Public Library Activities. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems (CHI EA '18)*. ACM. https://doi.org/10.1145/3170427.3188605

[7] Jens Emil Grønbæk, Banu Saatçi, Carla F. Griggio, and Clemens Nylandsted Klokmose. 2021. MirrorBlender: Supporting Hybrid Meetings with a Malleable Video-Conferencing System. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. ACM. https://doi.org/10.1145/3411764.3445698

[8] Tom Horak, Andreas Mathisen, Clemens N. Klokmose, Raimund Dachselt, and Niklas Elmqvist. 2019. Vistribute: Distributing Interactive Visualizations in Dynamic Multi-Device Setups. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. ACM. https://doi.org/10.1145/3290605.3300846

[9] Clemens N. Klokmose, James R. Eagan, Siemen Baader, Wendy Mackay, and Michel Beaudouin-Lafon. 2015. Webstrates: Shareable Dynamic Media. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, 280–290. https://doi.org/10.1145/2807442.2807446

[10] Clemens Nylandsted Klokmose, Christian Remy, Janus Bager Kristensen, Rolf Bagge, Michel Beaudouin-Lafon, and Wendy Mackay. 2019. Videostrates: Collaborative, Distributed and Programmable Video Manipulation. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '20)*. ACM, 233–247. https://doi.org/10.1145/3332165.3347912

[11] Jiali Liu and James Eagan. 2021. ADQDA: A Cross-Device Affinity Diagramming Tool for Fluid and Holistic Qualitative Data Analysis. *Proceedings of the ACM on Human-Computer Interaction* 5, ISS (2021). https://doi.org/10.1145/3488534

[12] Andreas Mathisen, Tom Horak, Clemens N. Klokmose, Kaj Grønbæk, and Niklas Elmqvist. 2019. InsideInsights: Integrating Data-Driven Reporting in Collaborative Visual Analytics. *Computer Graphics Forum* 38, 3 (2019), 649–661. https://doi.org/10.1111/cgf.13717

[13] Midas Nouwens, Marcel Borowski, Bjarke Fog, and Clemens Nylandsted Klokmose. 2020. Between Scripts and Applications: Computational Media for the Frontier of Nanoscience. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM. https://doi.org/10.1145/3313831.3376287

[14] Roman Rädle, Midas Nouwens, Kristian Antonsen, James R. Eagan, and Clemens Nylandsted Klokmose. 2017. Codestrates: Literate Computing with Webstrates. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. ACM, 715–725. https://doi.org/10.1145/3126594.3126642

[15] Daisy Yoo, Aurélien Tabard, Alix Ducros, Peter Dalsgaard, Clemens Nylandsted Klokmose, Eva Eriksson, and Sofia Serholt. 2020. Computational Alternatives Vignettes for Place- and Activity-Centered Digital Services in Public Libraries. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. ACM. https://doi.org/10.1145/3313831.3376597