

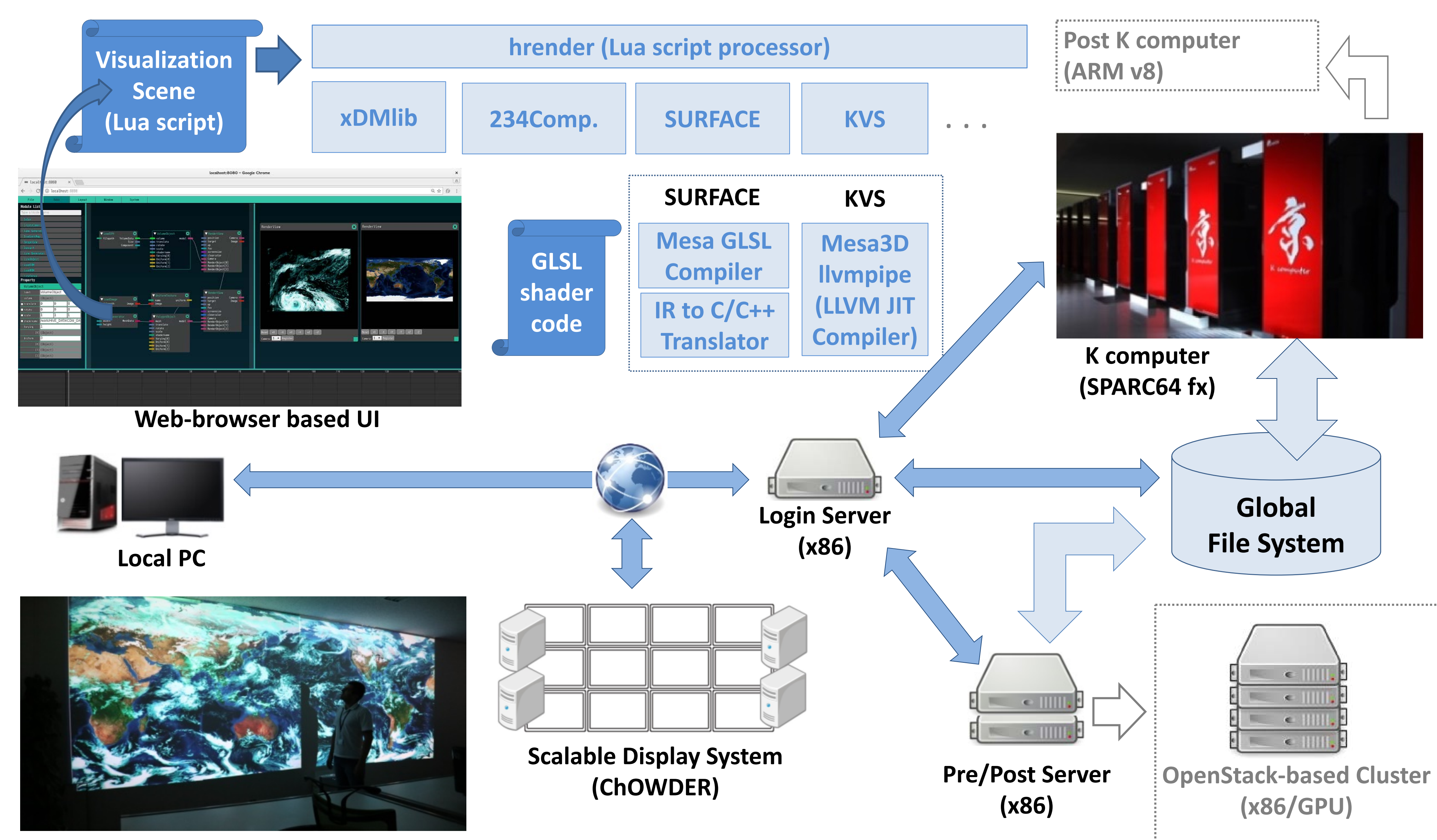
HIVE: A Cross-Platform, Modular Visualization Ecosystem for Heterogeneous Computational Environments

Jorji Nonaka, Tomohiro Kawanabe, Fumiyoshi Shoji (RIKEN R-CCS)
Kenji Ono (Kyushu University), Naohisa Sakamoto, Kengo Hayashi (Kobe University),
Masahiro Fujita (LTE Inc.), Kentaro Oku (KASHIKA), Kazuma Hatta (IMAGICA DIGITALSCAPE)

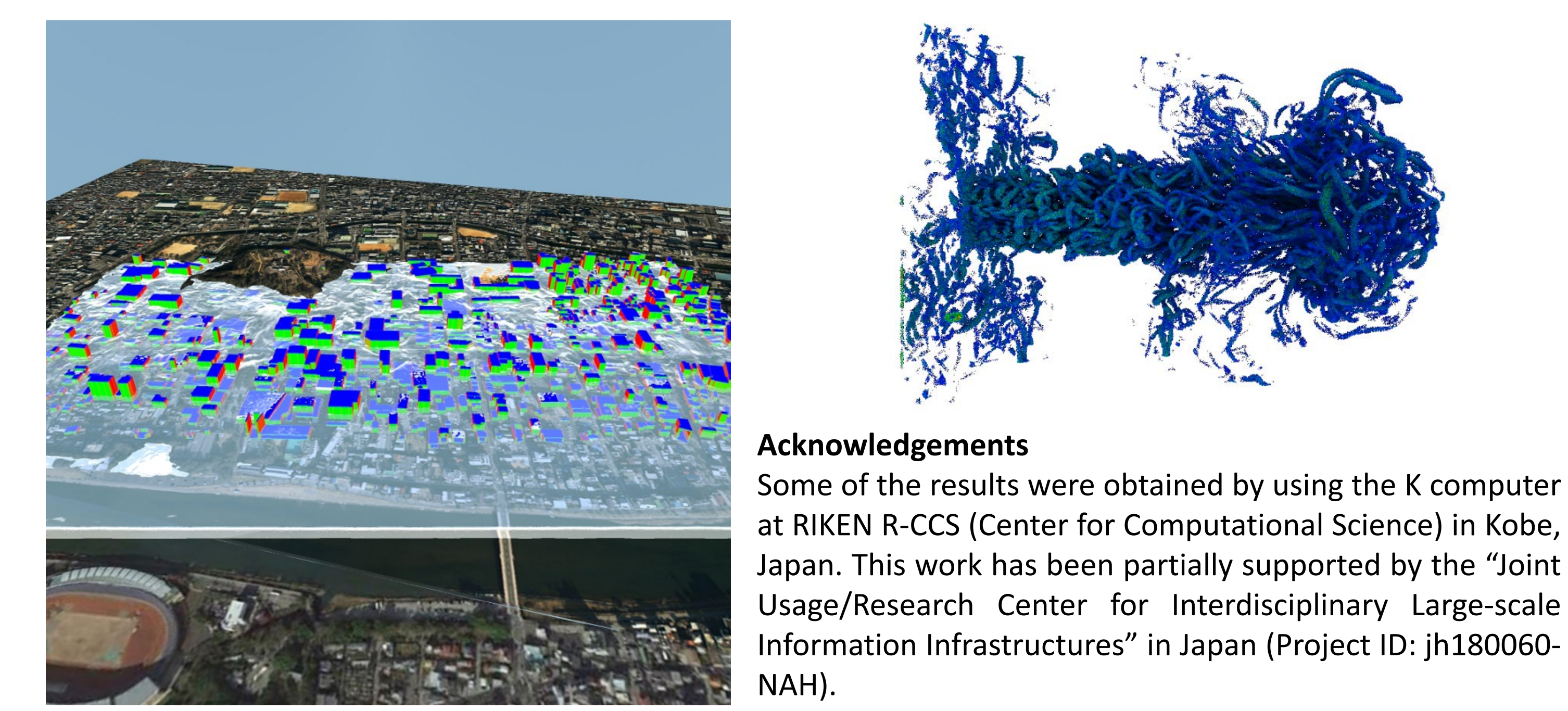
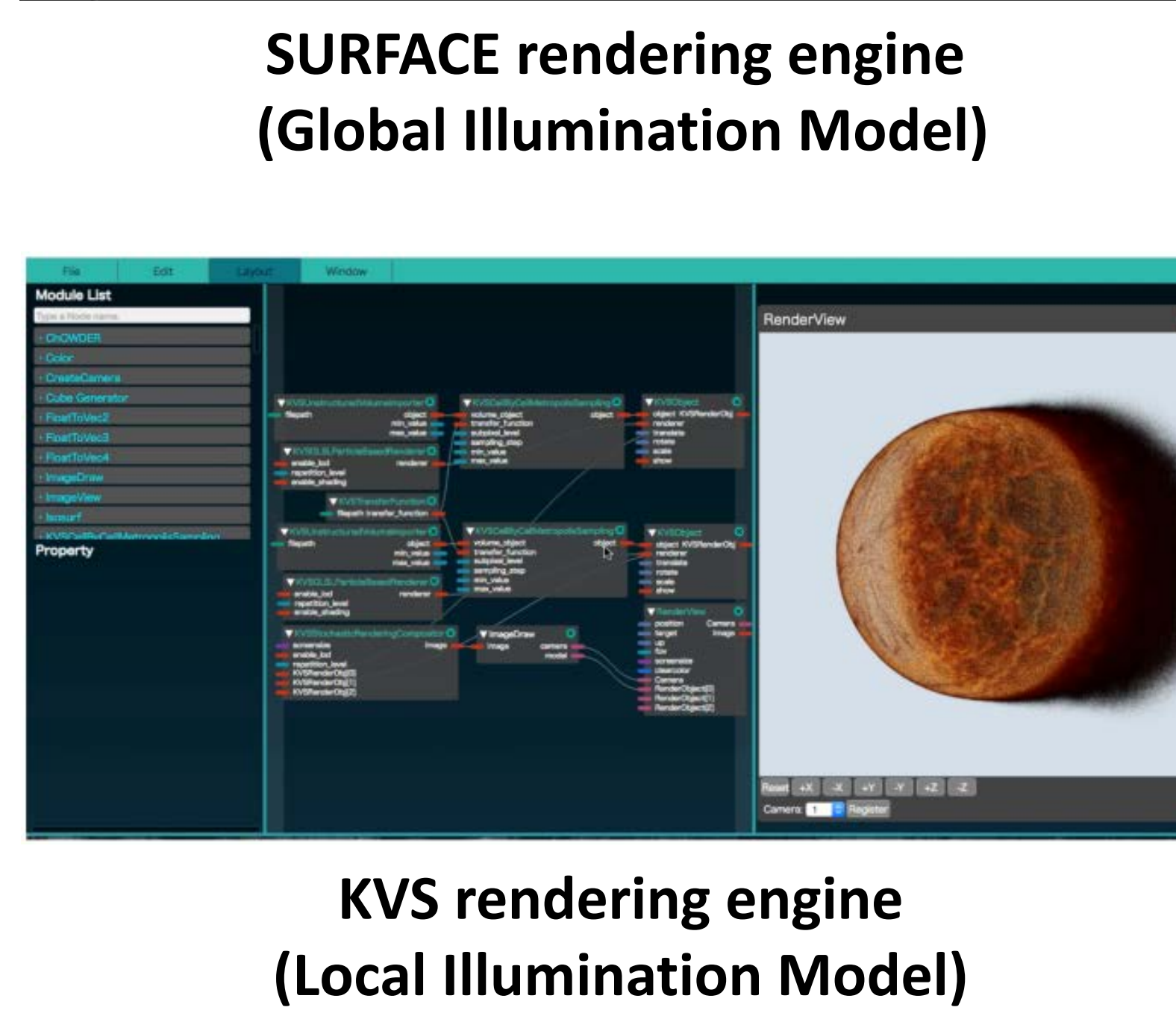
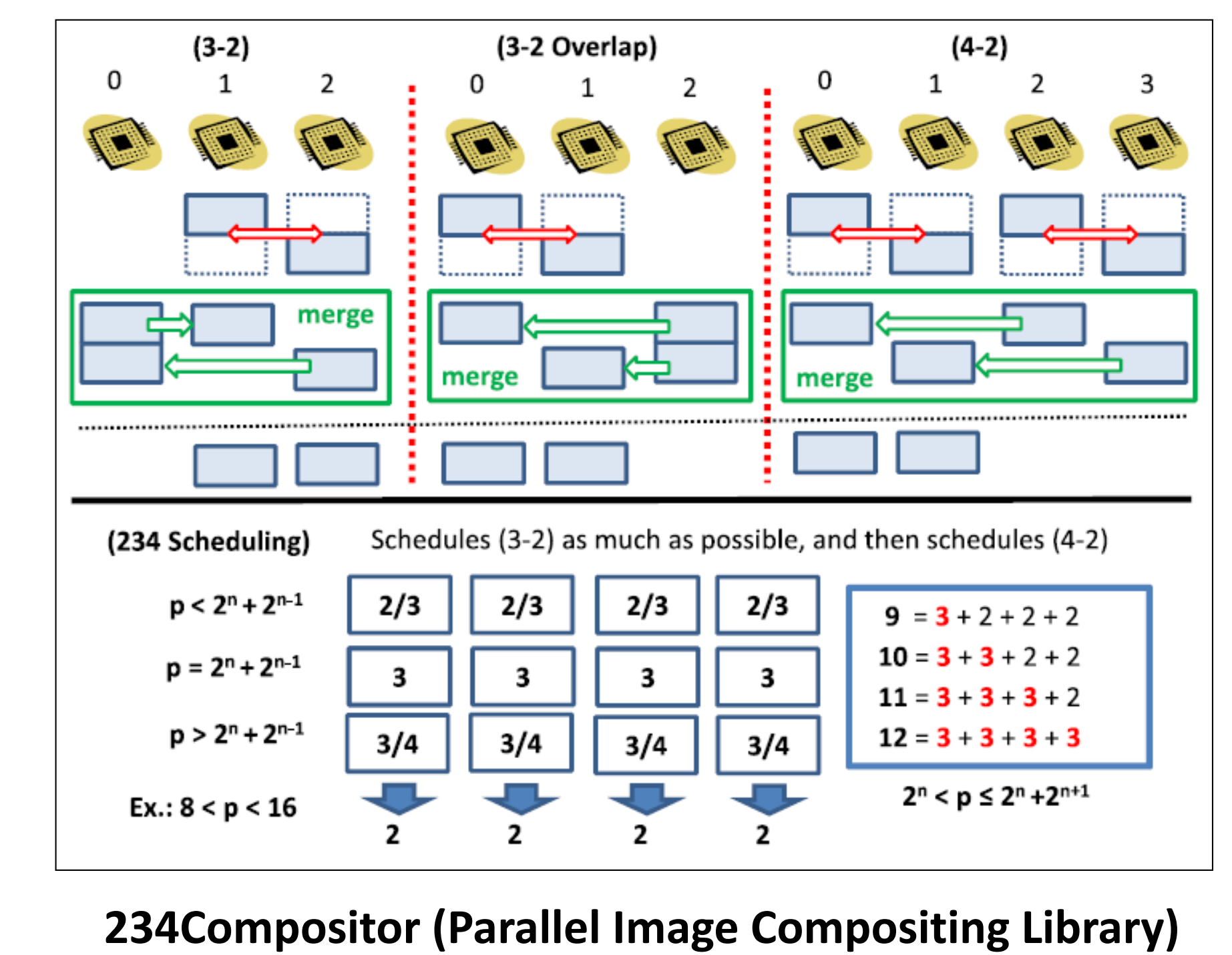
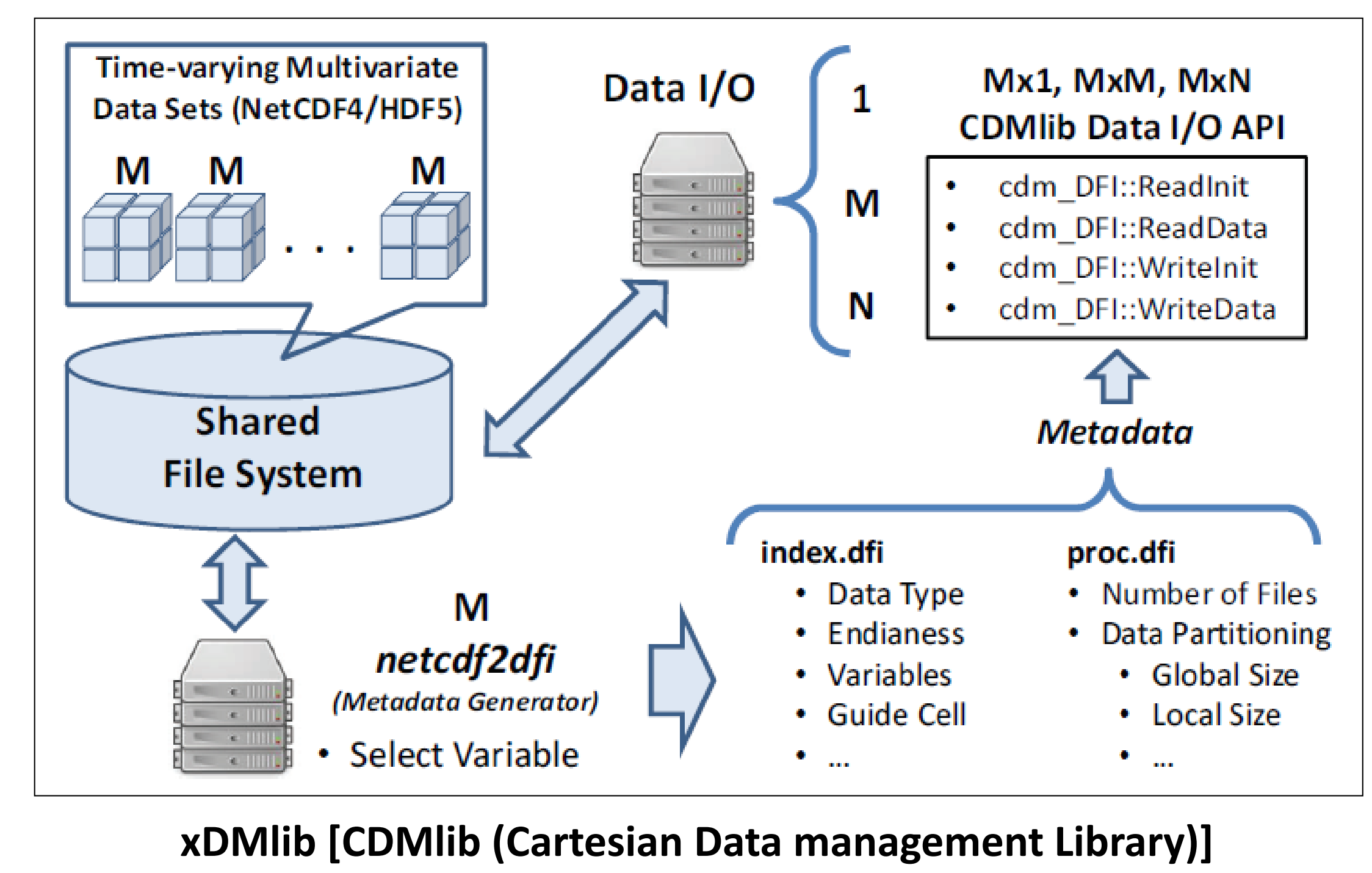
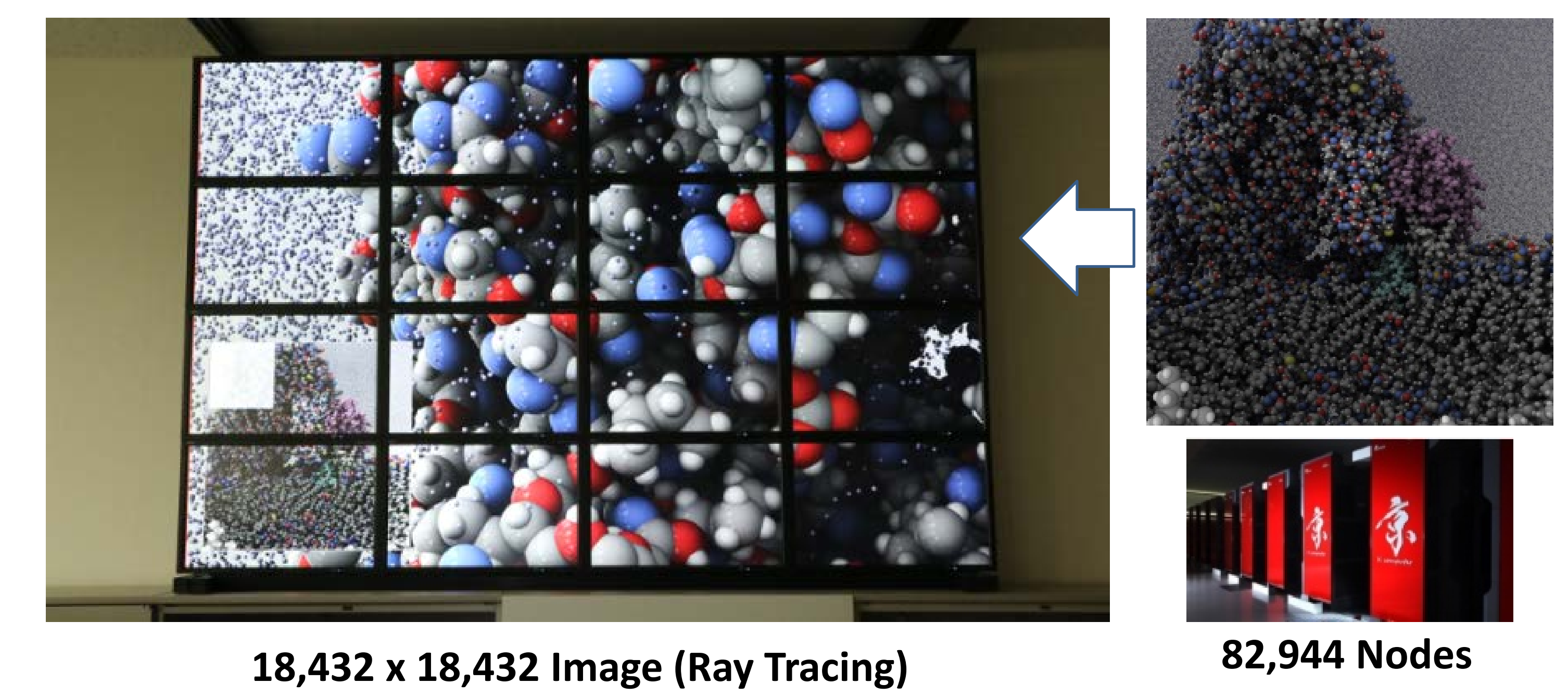
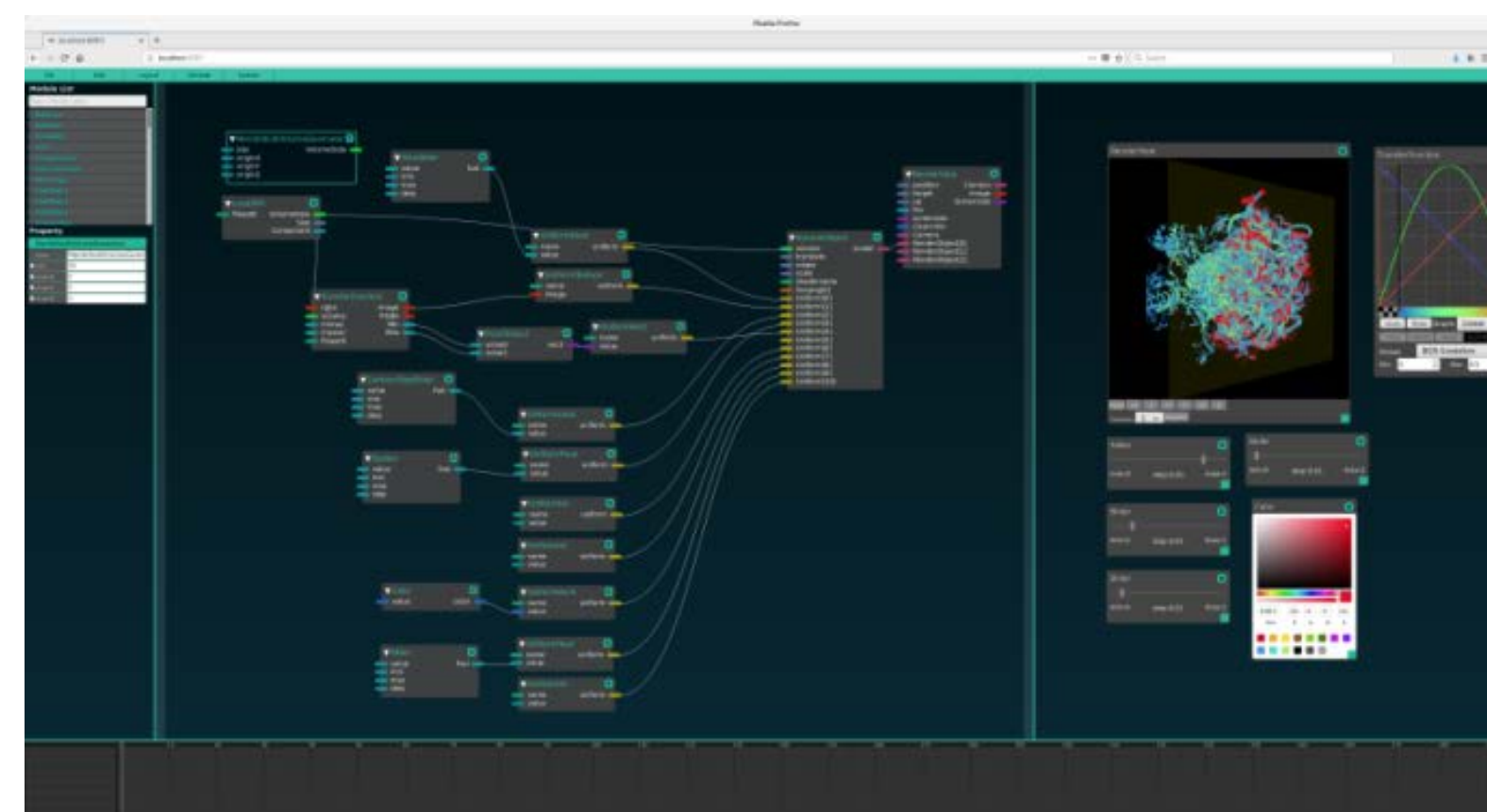
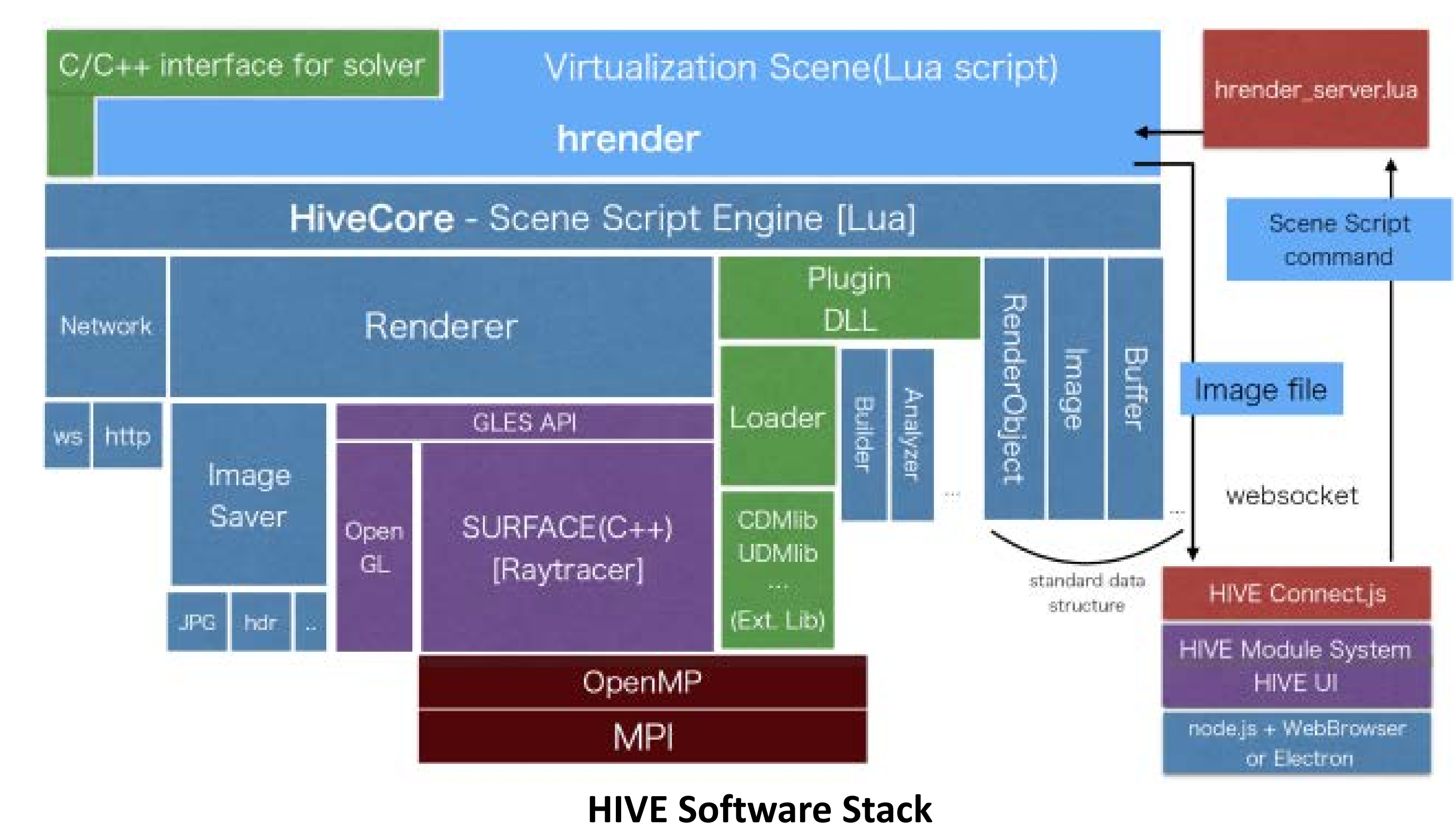


$$I_j = \int_{\Omega} f_j(\bar{x}) d\mu(\bar{x})$$

Abstract
HPC operational environments usually have supporting computational systems for assisting pre- and post-processing activities such as the visualization and analysis of simulation results. A wide variety of hardware systems can be found at different HPC sites, and in our case, we have a CPU-only (x86) large memory server, and there is a plan to replace this with a modern OpenStack-based CPU/GPU Cluster. HPC systems themselves can also be used for executing visualization related processing when applying the in-situ approach, and in our case this will be a SPARC64 fx CPU based HPC system (K computer). It is publicly announced that the current system will be replaced with an ARM based HPC system in a near future. Therefore heterogeneity and scalability are needed to be tackled in order to efficiently use these heterogeneous computational resources for large-scale data visualization on both post-hoc and in-situ contexts. In this poster we present HIVE (Heterogeneously Integrated Visual-analytics Environment), a cross-platform and modular ecosystem for providing visualization service building blocks in such heterogeneous computational environments. Lightweight Lua scripting language is used to glue necessary visualization pipeline related modules, and this loosely coupled modular approach facilitates the long-term development and maintenance.



HIVE Visualization Ecosystem
HIVE adopted modular design approach for integrating some own developed as well as third party tools and libraries to facilitate functionality enhancements and maintainability. The figure on the right side shows an overview of the software stack of the HIVE with some of the currently integrated tools and libraries. Most of the libraries and tools have been written using C and C++ language, and the visualization pipeline related functionalities are provided to the users as a Lua-based API. JSON has been used to provide a Web browser-based visualization workspace. Websocket was used for communication between the HIVE rendering module and Web-browser based UI, for enabling interactive visual exploration of remotely stored data sets. Visualization scenes prepared in the GUI workspace can be exported as a Lua script to be used in batch-based visualization. This offline rendering capability can be used to render medium and large datasets on clusters as well as on the supercomputers.



Acknowledgements
Some of the results were obtained by using the K computer at RIKEN R-CCS (Center for Computational Science) in Kobe, Japan. This work has been partially supported by the "Joint Usage/Research Center for Interdisciplinary Large-scale Information Infrastructures" in Japan (Project ID: jh180060-NAH).

