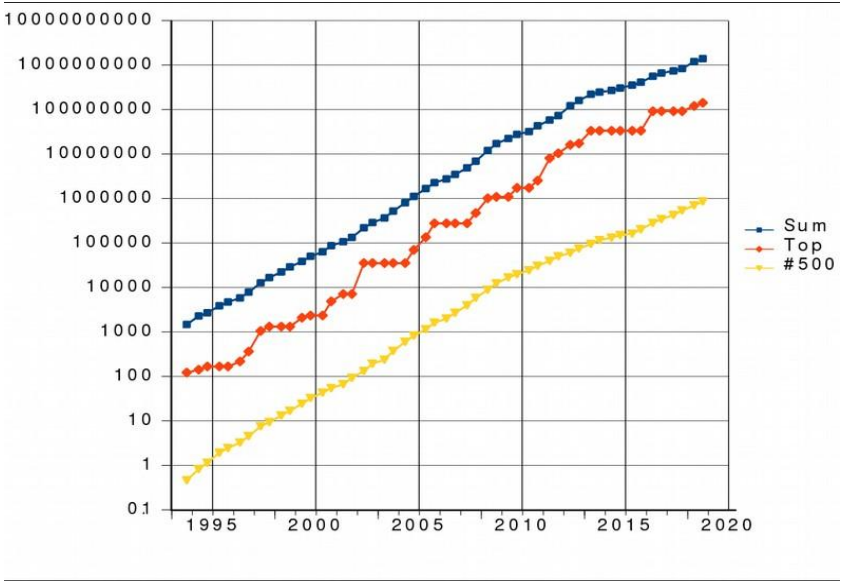
# Cloud visualization service using ParaViewWeb

Dmitry Puzyrkov, Sergey Polyakov, Sergey Markizov, Victoria Podryga, Nikita Tarasov

Keldysh Institute of Applied Mathematics (Russian Academy of Sciences)

# Introduction

- In the last decade there was a major breakthrough of the HPC technologies.
- It results to an very rapid increasing of the computational experiments results.
- Modern personal workstations cannot download such a big data volume.
- The first solutions, like ParaView, require to install specific software versions and do not have an interface for integration for remote control calculations systems.



# Introduction

- Modern web browsers support very useful technologies, such as HTML5, WebGL[1], WebSockets[2], and the last versions of JavaScript(ECMAScript)[3].
- There are a lot of open source packages for data visualization .
- ParaView holds a leading position in data visualization, works with different data formats, and has algorithms for data analysis.
- The ParaView project team has recently continued with a tool for integration with web services - ParaViewWeb.
- We had a system for a remote control of computational experiments - Kiam MoISDAG[8] and wanted to integrate it with ParaViewWeb for data visualization.

# ParaViewWeb

ParaView[4] is a open source framework for data analysis and data visualization developed by the Kitware company. And ParaViewWeb[5] is a subsystem of it, but now it is included into a basic ParaView version.

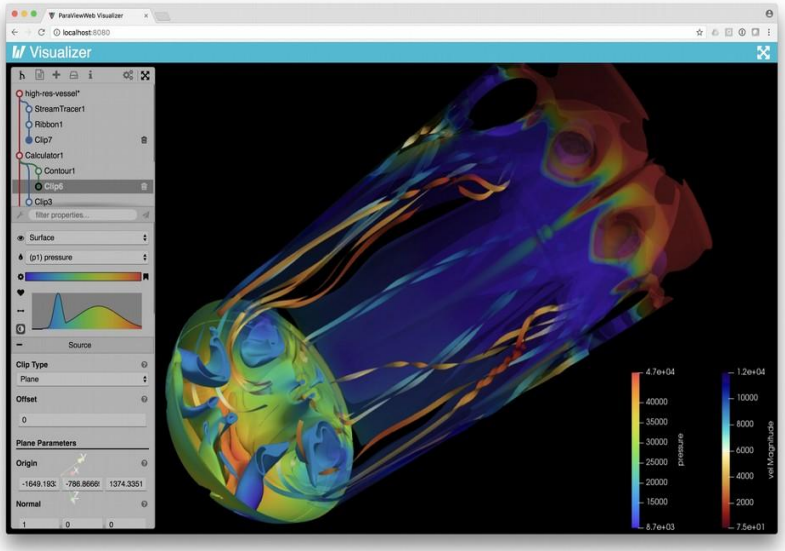
There are four parts of ParaViewWeb:

1. ParaViewServer is a server side tool for data preprocessing.
2. Necessary means for rapid data transport between data storage and client side application on the Internet.
3. JavaScript library for data visualization with the WebGL technology.
4. ParaView open interface on Python programming language and its own interpreter pvpython.

# ParaViewWeb Visualizer

ParaViewWeb Visualizer is a application from Kitware. But it is a bad choice if you need to modify its functional or to integrate it into your system. In the last case you need to implement a lot of wrappers and your infrastructure grows very fast.

In the development we started from ParaViewWeb Visualizer for the test.



# Technological stack

- Nginx - the most popular fast proxy server and load balancer
- Django - MVC(Model-View-Controller) web framework for fast server side development. It allows to use with ParaView source code[6]
- Vue.js - lightweight frontend framework for a fast development of SPA(Single Page Application)[7]



# Architecture(connection)

- The are five main components in the system: web browser with client side application, Django server application, nginx proxy server, Launcher for new pvpython processes and ParaViewWeb processes.
- Nginx proxy all the WebSocket connections between users and ParaViewWeb instances
- On new user the server side application sends a request for a new ParaViewWeb process and returns a access data(UID)
- After that the client side application interacts with ParaViewWeb process by WebSocket

# Architecture

# Integration

Django application provides access to static files and API for new user registration with launching of a ParaViewWeb process. And it works with user sessions. It is possible to run the ParaViewWeb Launcher or ParaViewWeb visualization processes on an another server.

The main part of a remaining source code of the integration relates to frontend  development. ParaViewWebClient is really easy to use and there is a way to integrate it into any modern web application. Also technologies, such as Electron.js allow to wrap frontend application and receive a desktop application.

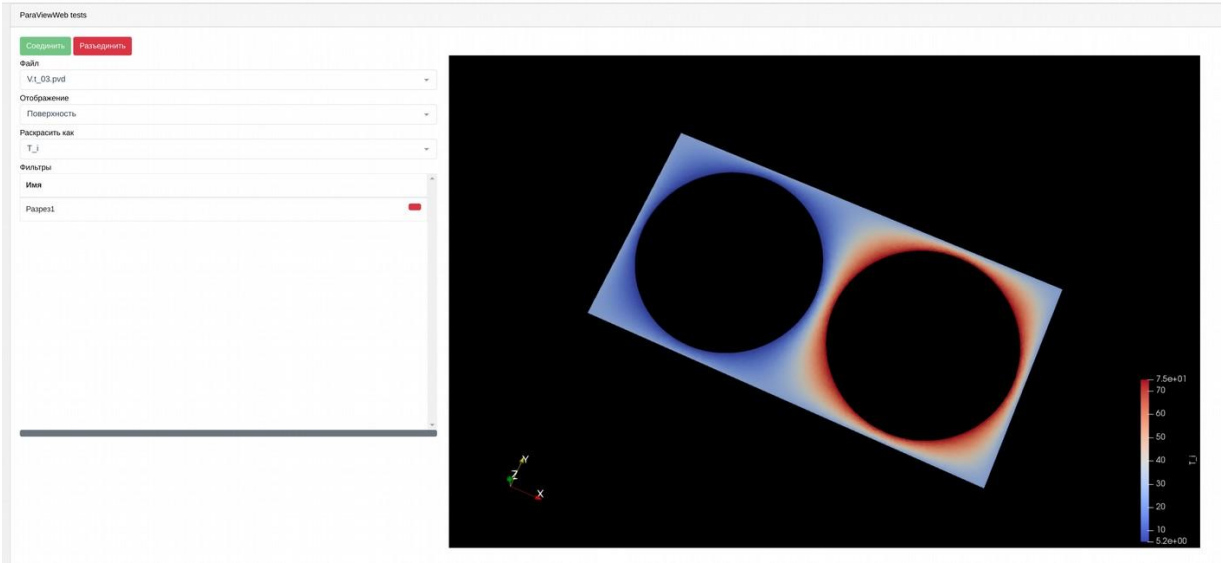
# Conclusion

ParaViewWeb is the best choice for the integration with custom web laboratories. We successfully integrated it into our remote control system - Kiam MolsDAG and get the first results of data visualization. The same integration do not require to make a difficult infrastructure or write a lot of code for a integration. You can integrate it into any modern web application.

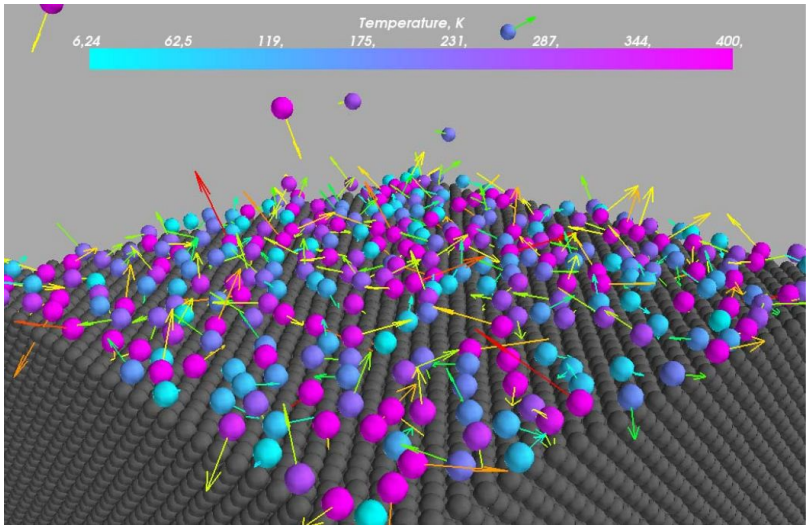
Also in this work we got a simple supporting module - ParaViewWebHelper. It can create a new WebSocket connection to a server side ParaViewWeb application and get an interface to work with ParaView methods, such as working with filters.

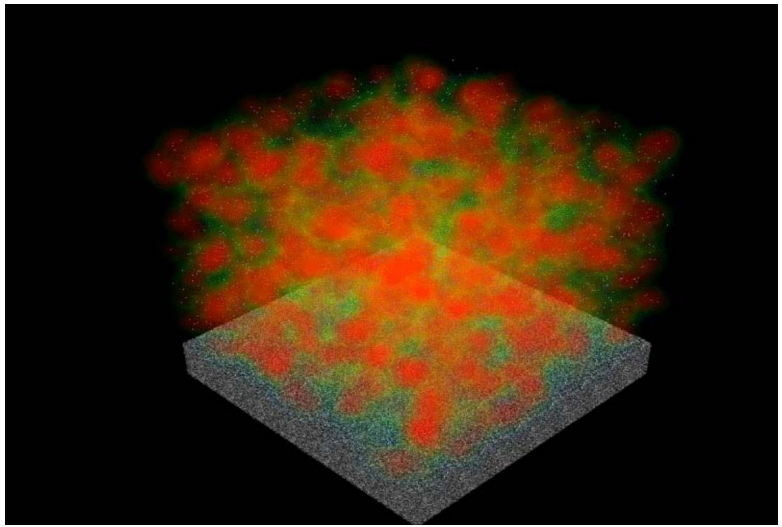
# Conclusion

Interface implemented with ParaViewWeb, Vue.js and Django for a numerical solution of a heat equation



# Conclusion



Molecular dynamics data visualization

# References


1. WebGL Overview - The Khronos Group Inc (online). https://www.khronos.org/webgl/
2. RFC 6455 - The WebSocket Protocol - IETF Tools (online). https://tools.ietf.org/html/rfc6455
3. Standard ECMA-262 - Ecma International (online). https://www.ecma-international.org/publications/standards/Ecma-262.htm
4. Ahrens, James, Geveci, Berk, Law, Charles, ParaView: An End-User Tool for Large Data Visualization, Visualization Handbook, Elsevier, 2005, ISBN-13: 978-0123875822.
5. Jourdain, Sebastien & Ayachit, Utkarsh & Geveci, Berk. (2010). ParaViewWeb: A Web Framework for 3D Visualization and Data Processing. IADIS international conference on web virtual reality and three-dimensional worlds.
6. Django [Computer Software] (online). https://djangoproject.com
7. Vue.js - The Progressive JavaScript Framework (online). https://vuejs.org/
8. Д.В. Пузырьков, В.О. Подрыга, С.В. Поляков, “Облачный сервис для масштабных молекулярно-динамических расчетов: от идеи до реализации”, pp. 406-416 in the volume “Научный сервис в сети Интернет: труды XIX Всероссийской научной конференции” (18-23 сентября 2017 г., г. Новороссийск). — М.: ИПМ им. М.В.Келдыша, 2017. — 480 с. doi:10.20948/abrau-2017

# Thank you