# Model Driven Approach for Virtual Lab Authoring - Chemical Sciences Labs

by

Shovam Swain, LALIT Mohan Mohan, Venkatesh Choppella, Y.Raghu Babu Reddy

in

*The 18th IEEE International Conference on Advanced Learning Technologies (ICALT 2018)*

Report No: IIIT/TR/2018/-1

Centre for Software Engineering Research Lab
International Institute of Information Technology
Hyderabad - 500 032, INDIA
July 2018

# Model Driven Approach for Virtual Lab Authoring - Chemical Sciences Labs

*Abstract*—**Virtual Labs allow performance of experiments without real world instrumentation needs. Most of these virtual experiments are simulation based and developed using proprietary or opensource simulation, Adobe Flash and other client side software. The procedural steps involved in conducting these experiments are hardwired and require software modification for enhancements or alignment to course structure. We propose a model based approach that eases development of virtual experiments without dependency on software programmers for any changes. We demonstrate our model driven based approach on Chemical Sciences labs of Virtual Labs, a Government of India initiative. With our model driven based approach, the effort for new experiment development or conversion of existing experiments using outdated technologies is less than a person day as compared to more than a person month.**

*Keywords*—**Virtual Lab, Lab Authoring, Model Driven Development, Chemical Sciences.**

## I. Introduction

Improving digital literacy and cheaper internet is providing opportunities for supplementing and complementing class room education. Through these internet channels, Massive Open Online Courses (MOOC) from top institutes of the world has made best education affordable and available to remote locations. Most of the MOOCs have teachers delivering lectures using applicable pedagogical models. It is very common to find students have theoretical (through textbooks or MOOCs) knowledge of the subject, but lack practical experience in the field. Along with the theory and lectures, ability to perform experiments would further enhance the learning [1]. To enhance experiential learning, academics [2] and industry (including EdTech[1] startups) are researching and implementing various modes of Virtual Labs/Experiments. Virtual Labs can massively scale and reach to various locations of the world enabling teachers and students to perform experiments without spending on lab infrastructure. More specifically, Virtual Labs can aid institutions having scarce resources. Virtual Labs provide flexibility to access from anywhere and anytime without depending on lab technician or room availability, provides instant feedback and ability to redo experiment that can enhance critical thinking, access to latest equipment and all this at zero or very low cost. This also reduces the skillgap between students from top institutes to lower tier institutes, thus leading to better employability. To encourage spread of Virtual Labs, academic institutes, industry and government organizations are funding these initiatives.

[1]https://goo.gl/GdH3NQ

Virtual Labs can be broadly categorized into Simulations, Remote Triggered and Virtual/Augmented Reality based labs. To build these simulation based labs, various opensource/proprietary animation software including Adobe Flash are being used. These software provide rich UI and contain libraries that aid in rapid development of experiments. However, these software are not domain specific (leading to pedagogical issues) and generate a runtime that is hardwired and has dependency on software programmers for future modifications [3]. Some of this software also require a browser plugin leading to compatibility issues. Experiments built using Flash is reduced to a series of predefined frames being put in motion by a predefined set of mouse clicks. Flash also cannot handle user input to dynamically create and update objects. Source code maintenance of Flash applications is also an issue. There might be two instances of the same lab, but with very minute differences between them. However the code for these two labs might differ a lot. With the advances in Model Driven Development [4] [5], domain specific modeling software are also being used for building simulations and have made experiment development faster and consistent. Some of the proprietary and opensource software such as Simulink, OpenModelica[2], Dymola and others are being widely used in academics and industry. These software are developed with large repositories of resuable assets and require less code for changes. However, some of their programming languages are proprietary, have server side dependencies and do not provide user specific Application Programming Interface (APIs) for making it experiential while using virtual labs. Adding an authentication or analytics API to the simulations would require changes to the base code of modeling software.

Based on the best practices and shortcomings in Flash and Modeling software based Virtual Labs [6] [7] [8] [9] [10] and [11], we propose a Lab Authoring Kit. The kit provides flexibility to rapidly develop experiments in a model driven approach, is available on a web browser with no server/browser side plugins, contains linkages/services to consume custom APIs using simple HTML and Javascript technologies. We demonstrate our approach on Chemical Sciences discipline in Virtual Labs[3]. Virtual Labs is a HRD initiative from Government of India with  2 Million usages across 1000+ experiments of 10+ engineering disciplines.

[2]https://openmodelica.org/

[3]https://vlabs.ac.in

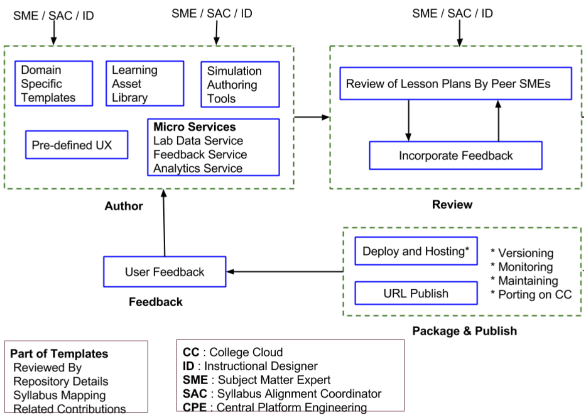


Fig. 1. Lab Authoring Kit

## II. Approach

Our model driven approach for Lab Authoring is depicted in the figure 1. Our architecture and design abstraction provides consistent and scalable user experience. In our approach, there are primarily four stages,

- Authoring - This stage contains asset library, templates, simulation software and services of available APIs (Feedback and Analytics). The assets could be images of chemistry apparatus, electrical and mechanical devices. These assets are used to create some templates such as an experiment on Internal Combustion engines or chemistry lab on spectroscopy. These stage can be used by Faculty that have pedagogical understanding of the course and students.
- Review - Using the existing templates or based on asset library, an experiment is created and this experiment is sent for peer/SME review. This stage is iterative with the authoring stage to ensure quality output is created.
- Package & Publish - The reviewed experiment is pushed to the next stage for packaging and deploying. To address Virtual Labs scalability, the experiments are hosted on a cloud environment.
- Feedback - The existing modeling software are mostly standalone and do not have feedback system. Our proposed approach with services for Analytics would lead us to enhancing experiments, templates and asset library.

### A. Chemical Sciences Lab

As the initial phase, we created the Stage 1 software of the Lab Authoring Kit. We demonstrate it on Virtual Labs Chemical Science Labs. There are approximately 100 experiments across 12 labs. Most of these labs were developed using Adobe Flash software and their maintenance has become a concern as most of the browsers are stopping to support flash. Converting these labs to current supported technologies still leads to future maintenance issues and also require few weeks to develop as each component needs to be programmed separately. To address this in Lab Authoring Kit, our design treats every lab apparatus like an object and each objects have properties and actions/methods that can be modified. Each instance of an 'apparatus' on the 'table' is a separate object with its own methods and properties. Also, the interactions between objects can be defined as actions. Each interaction may change one or more of the other object's properties. Thus, through the course of the experiment, objects interact with each other, changing their properties accordingly. Using this logic, we can build a wide range of experiments. The figure 2 shows the setup of a

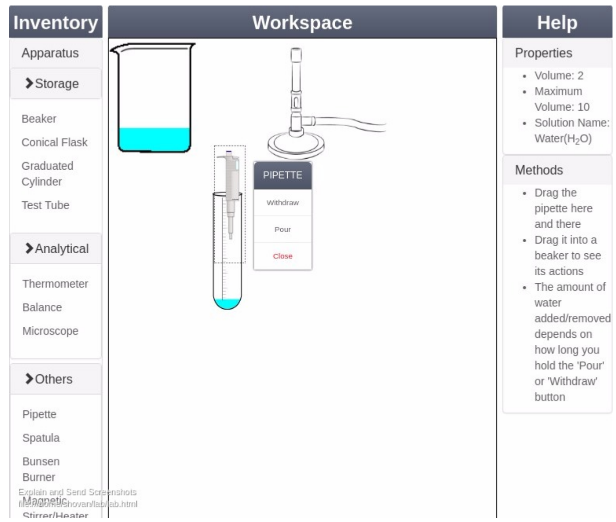


Fig. 2. Screenshot of Chemical Sciences Lab

Chemical Sciences lab. On the left, we have the 'Inventory' containing apparatus or asset library. In the center, we have a 'Workspace' that can be considered as an experiment workbench, the place for user to perform the experiment. On the right, there is a 'Help' section that acts like a tutorial, or a tooltip and has two subsections - 'Properties' and 'Methods'. The properties bar is a section that displays all the specified properties of the element which has been last clicked upon. It is dynamically updated by the object itself. The methods bar is more like a tutorial section which explains to the user how to use that particular piece of equipment.

In order to structurally develop the lab keeping in mind that the code should be maintainable, we used Software Engineering 'Builder Design Pattern'. The provides a template for creating/updating new objects, a certain pattern is followed irrespective of the object. This ensures that creating/updating new labs is considerably easier now, and thus maintainability is ensured. To make sure that each object is not just a rigid entity with fixed properties, we attributed each object with suitable state variables. These variables may change within themselves with the passage of time, or due to interaction with some other object. This usage of a pattern ensured that the user has freedom when it comes to using apparatus and exploring their possible

| Software Name / Categrory | Lab Authoring Kit | ChemCollective | Phet |
|---|---|---|---|
| Add Assets and Templates | Yes | Partial | Partial |
| API Link | Yes | No | No |
| Properties and Actions | Yes | Partial | No |
| Staged Deployment | Yes | No | No |
| Open source | Yes | Yes | No |
| Cloud Ready | Yes | No | No |

Fig. 3. Comparison of Chemistry Labs Authoring Tools

actions. The code is written in pure Javascript, Node.js[4], interactjs[5] library for the animations and the inter object interactions providing modularity. Adding new objects and improving upon existing objects is made easy. Every new object type has its own actions. This is stored in its own JS file in the 'actions' folder and defines the experiment. Any kind of lab a developer may want to make, user can just write down the right order and nature of interactions in these JS files. This approach is not only available of the current Virtual Chemical Sciences lab but also in other labs (Electronics, Data structure, etc.)
With our approach, the conversion of labs from Flash to Javascript has reduced from weeks[3] to a day. Our source code is available on Github[6] with GNU Affero GPL V3 licensing for easing future enhancements and changes. We demonstrated our approach with a faculty responsible for 'Spectroscopy Labs', faculty's first reaction was 'Wow' and mentioned that if he had this thought earlier it would have reduced the development time. The faculty also suggested some domain specific changes that related to calibration of the assets. We also compared some of the popular Virtual Chemistry Labs environments from CMU's ChemCollective[7] and University of Colarado's PhET[8] as against our approach, the comparison results are available in figure 3.

## III. Conclusions and Future Work

The research reviewed several existing virtual labs and suggested a Lab Authoring Kit that provides templatizing, adding new libraries and provides consistency in user experience with less code. The new Virtual Labs would be interactive, provide opportunities to perform 'What-If' analysis during experiment while reducing the focus on menial lab intricacies. This approach also reduced the conversion of Flash Conversion Labs from weeks to a day. With this approach, institutes can develop/use experiments that will align to their course adoption, thereby, increasing the adoption of virtual labs. Some of our future works include modifying and implementing this architecture to accommodate other disciplines, crowdsourcing development of experiments and refining APIs, working towards an exhaustive asset and template library.

[4]https://nodejs.org/
[5]http://interactjs.io/
[6]https://github.com/Virtual-Labs/lab-authoring-tool-iiith
[7]http://chemcollective.org/activities/autograded/124
[8]https://phet.colorado.edu/en/simulations/category/chemistry

## References


[1] D. A. Schön, *Educating the Reflective Practitioner: Toward a New Design for Teaching and Learning in the Professions.* Jossey-Bass, 1987.

[2] "Virtual Laboratories for Education in Science, Technology, and Engineering: A Review, author=Potkonjak, Veljko and Gardner, Michael and Callaghan, Victor and Mattila, Pasi and Guetl, Christian and Petrović, Vladimir M and Jovanović, Kosta, journal=Computers & Education, volume=95, pages=309–327, year=2016, publisher=Elsevier."

[3] Maheshwari, Yogesh and Reddy, Y. Raghu, "A Study on Migrating Flash Files to HTML5/JavaScript," in *Proceedings of the 10th Innovations in Software Engineering Conference*, ser. ISEC '17. New York, NY, USA: ACM, 2017, pp. 112–116. [Online]. Available: http://doi.acm.org/10.1145/3021460.3021472

[4] Sendall, Shane and Kozaczynski, Wojtek, "Model Transformation: The Heart and Soul of Model-driven Software Development," *IEEE software*, vol. 20, no. 5, pp. 42–45, 2003.

[5] France, Robert and Rumpe, Bernhard, "Model-driven development of complex software: A research roadmap," in *2007 Future of Software Engineering*. IEEE Computer Society, 2007, pp. 37–54.

[6] H. Matsuda and Y. Shindo, "Virtual Chemical Experiment using Cyber Assistant Professor: CAP," in *Eighth IEEE International Conference on Advanced Learning Technologies*. IEEE, 2008, pp. 977–981.

[7] Morozov, Mikhail and Tanakov, Andrey and Gerasimov, Alexey and Bystrov, Dmitriy and Cvirco, Evangelia, "Virtual Chemistry Laboratory for School Education," in *IEEE International Conference on Advanced Learning Technologies*. IEEE, 2004, pp. 605–608.

[8] Zhong, Ying and Liu, Chang, "User Evaluation of a Domain-oriented End-user Design Environment for Building 3D Virtual Chemistry Experiments," in *Proceedings of the First International Workshop on User Evaluation for Software Engineering Researchers*. IEEE Press, 2012, pp. 5–8.

[9] Song, Kyohyun and Kim, Gunhee and Han, Inkyu and Lee, Jeongyoung and Park, Ji-Hyung and Ha, Sungdo, "CheMO: Mixed Object Instruments and Interactions for Tangible Chemistry Experiments," in *CHI'11 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2011, pp. 2305–2310.

[10] Gomes, Alessandra and Hermano, Rafael Erthal and Santanchè, André, "Web Metalaboratory: Composition of Laboratories on the Web," in *IEEE 10th International Conference on e-Science*, vol. 2. IEEE, 2014, pp. 43–48.

[11] de Jong, Ton and Sotiriou, Sofoklis and Gillet, Denis, "Innovations in STEM Education: the Go-Lab Federation of Online Labs," *Smart Learning Environments*, vol. 1, no. 1, p. 3, 2014.