

Visualising Event-B models with B-Motion Studio*

Lukas Ladenberger, Jens Bendisposto and Michael Leuschel

Institut für Informatik, Heinrich-Heine Universität Düsseldorf
Universitätsstr. 1, D-40225 Düsseldorf
`{bendisposto, leuschel}@cs.uni-duesseldorf.de`

1 Motivation

The communication between a developer and a domain expert (or manager) is very important for successful deployment of formal methods. On the one hand it is crucial for the developer to get feedback from the domain expert for further development. On the other hand the domain expert needs to check whether his expectations are met. An animation tool allows to check the presence of desired functionality and to inspect the behaviour of a specification, but requires knowledge about the mathematical notation. To avoid this problem, it is useful to create domain specific visualisations. One tool which performs this task is Brama. This tool is very important for ClearSy, and is being used for several industrial projects and has helped to obtain several contracts. However, the tool cannot be applied in conjunction with ProB. Also, creating the code that defines the mapping between a state and its graphical representation is a rather time consuming task. It can take several weeks to develop a custom visualisation.

In [1], we introduced a tool that like Brama allowed to create sophisticated visualisations using Macromedia Flash. The tool, however, still required the user to write some gluing code in Java to link the model and the visualisation. The visualisation built into PROB as described in [2] did not require to write code, as it uses a function written in B to link the model and its visualisation. These visualisations are rather simple and restricted. Also writing the required animation function can still be a considerable challenge.

We now introduce B-Motion Studio, a tool that allows to create visualisations as easy as using animation functions in PROB while being almost as sophisticated as our previous Flash based tool (e.g., see Figure 1). B-Motion Studio comes with a graphical editor that allows to create a visualisation within the modeling environment. Also, it does not require to use a different notation for gluing the state and its visualisation.

B-Motion Studio uses two important concepts: Controls and Observers. A control is a graphical representation of some aspects of the model. Typically we use labels, images or buttons to represent informations. For instance, if we model

* This research is being carried out as part of the DFG funded research project GEPAVAS and the EU funded FP7 research project 214158: DEPLOY (Industrial deployment of advanced system engineering methods for high productivity and dependability).

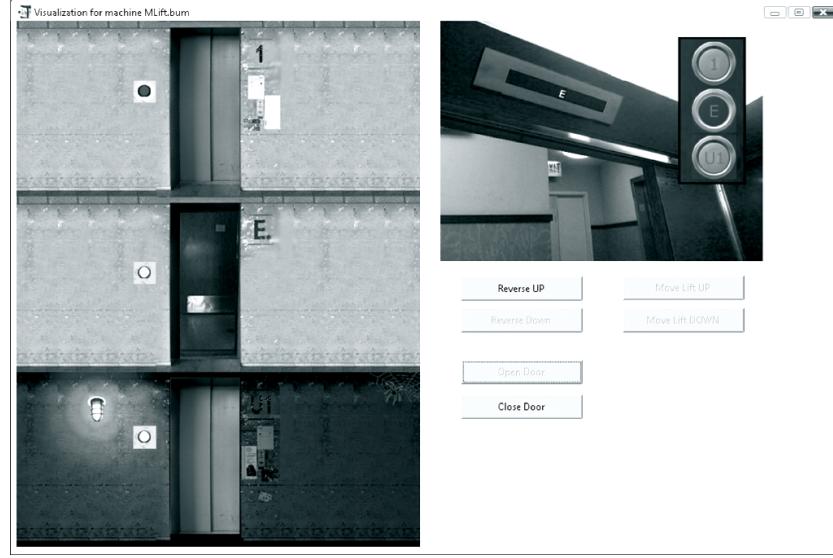


Fig. 1. The Visualisation in Action

a system that has a temperature and a threshold temperature that triggers a cool down, we might simply use two labels displaying both values, or maybe we can incorporate both information into a gauge display. It is also possible to define new controls for domain specific visualisations. Observers are used to link controls to the model's state, i.e., they do the same as the animation function in PROB. The main difference is, that we allow to decompose the animation function into different aspects, i.e., if our model contains information about the speed of a motor, we can separate all information regarding the speed from the information regarding the temprature. This allows us to write small functions and combine them rather than writing a single funtion covering all aspects of the model.

2 Conclusion

The main advantages of B-Motion Studio are:

- The modeler stays within a single notation. B-Motion Studio uses Event-B predicates and expressions as gluing code.
- An easy to use graphical editor, that allows to create visualisations with a few mouse clicks (see Figure 2).
- B-Motion Studio comes with a number of default observers and controls that are sufficient for most visualisations.
- It can be extended for specific domains.

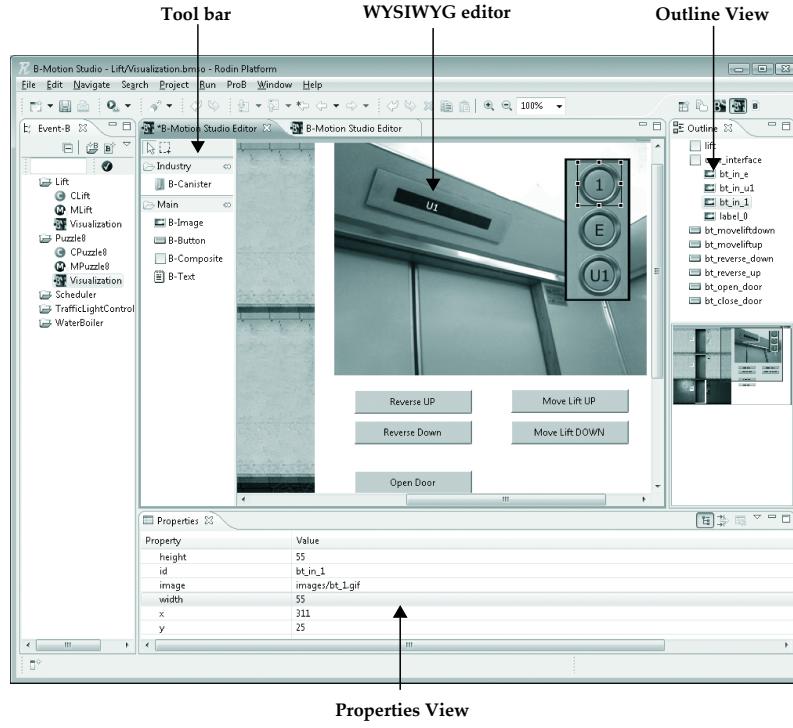


Fig. 2. The B-Motion Studio Perspective

In summary, we hope that B-Motion Studio provides a way to quickly generate domain specific visualisations for a formal model, enabling domain experts and managers to understand and validate the model. We also believe that our tool will be of use when teaching formal methods, both during lectures as a way to motivate students to write their own formal models.

3 Further information

More technical details, a tutorial and installation instructions can be found at the project's website <http://www.stups.uni-duesseldorf.de/BMotionStudio>.

References

1. J. Bendisposto and M. Leuschel. A generic flash-based animation engine for ProB. In *Proceedings of the 7th International B Conference (B2007)*, LNCS 4355, pages 266–269, Besancon, France, 2007. Springer-Verlag.
2. M. Leuschel, M. Samia, J. Bendisposto, and L. Luo. Easy Graphical Animation and Formula Viewing for Teaching B. *The B Method: from Research to Teaching*, pages 17–32, 2008.