Introducing compound components in JavaBIP

Simon Bliudze (simon.bliudze@inria.fr)

The goal of the project is to introduce compound components in JavaBIP. Indeed, in the original BIP framework, components can be assembled hierarchically to simplify re-use. This feature is missing in JavaBIP.

Context

This project focuses on JavaBIP [3], which is an open-source Java implementation of the BIP (Behaviour-Interaction-Priority) framework [1, 2] for the coordination of concurrent components.

A BIP model consists of a set of components and connectors that define all possible synchronisations among the components (see the figure below). Atomic components model the application activities, from the control point of view, as Finite State Machines (FSMs). Connectors define possible synchronisations among the transitions of these FSMs and the associated data transfer rules.

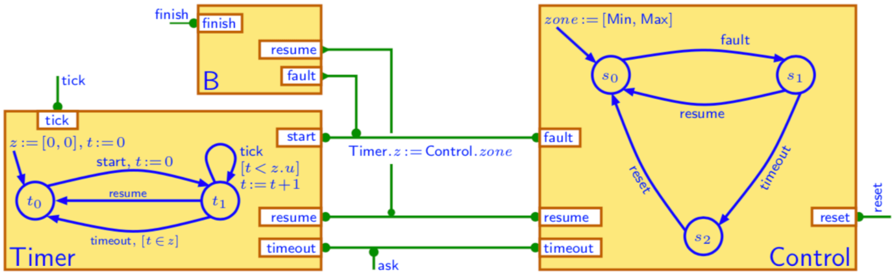

Figure 1: Example of a BIP model

As opposed to the original BIP implementation, where C++ code is generated for the complete system, JavaBIP realises the coordination of existing concurrent software components in an exogenous manner, relying exclusively on annotations, component APIs and external specification files. Component specifications in JavaBIP are given by additional annotated Java classes. They provide an abstract view of the coordinated software entities.

The work to be carried out

The student will have to

1. Understand the theory and implementation principles underlying JavaBIP

2. Define the component interface, i.e. the information that has to be exposed to allow compounding
3. Propose and implement the corresponding modifications to the JavaBIP software architecture
4. Illustrate the results through a meaningful case study

Location

The internship will be carried out in the Spirals project team at Inria Lille – Nord Europe.

Contact and application

For additional information and to apply please send an e-mail to Simon Bliudze (in English or French) with the subject “JavaBIP compounding internship”.

References

1. Simon Bliudze and Joseph Sifakis. The Algebra of Connectors — Structuring interaction in BIP. In Proceedings of the 7th ACM & IEEE International Conference on Embedded Software, EMSOFT 2007, page 11–20, Salzburg, Austria, October 2007. ACM SigBED. PDF
2. Ananda Basu, Saddek Bensalem, Marius Bozga, Jacques Combaz, Mohamad Jaber, Thanh-Hung Nguyen and Joseph Sifakis. Rigorous Component-Based System Design Using the BIP Framework. IEEE Software 2011. DOI
3. Simon Bliudze, Anastasia Mavridou, Radoslaw Szymanek, and Alina Zolotukhina. Exogenous coordination of concurrent software components with JavaBIP. *Software: Practice and Experience*, **47**(11):1801–1836, November 2017. PDF