A tale of high-level features in the Petri Box Calculus

Hanna Klaudel[1], Franck Pommereau[1], and Elisabeth Pelz[2]

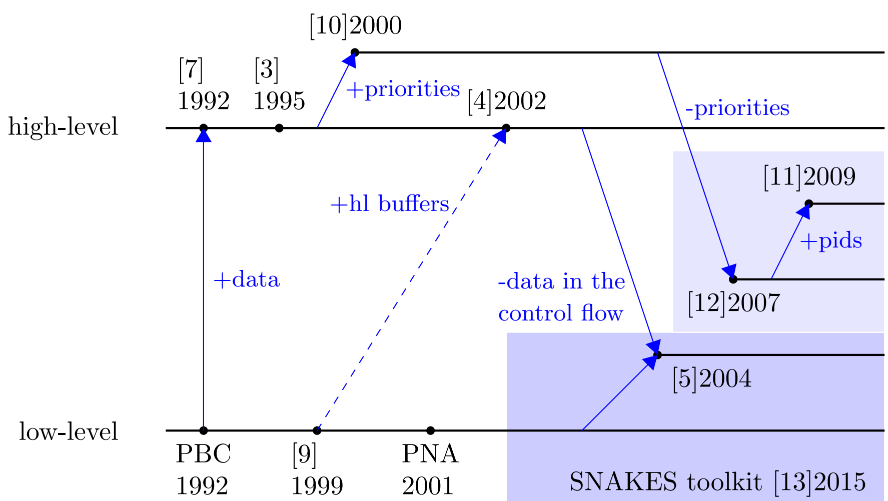
[1] University of Paris-Saclay, University of Évry
[2] University of Paris-East-Créteil

Abstract. Once upon a time the Petri Box Calculus was invented. Let us tell you our tale of the introduction of high-level features in this remarkable formalism that gave us a research topic for our whole research career.

The Petri Box Calculus [1] is a formalism that combines Petri nets and process algebras, which have been an idea that can be traced back to the early 70's. A team started to build on this topic with two EU-funded research projects DEMON and CALIBAN. It resulted in a formal framework (see also [2]) in which Petri nets can be composed like terms in a process algebras. This approach yields standard Petri nets that can be analysed automatically through the numerous methods and tools.

Our tale begins with the publication of the Petri Box Calculus (PBC) in 1992, it tells our story of introducing high-level features in the PBC and the steps it took to progressively improve these features until we were fully pleased with them. This is a personal story, and we apology for the many works we have left apart. Consequently, this tale has no end yet and much remains to be written...

The whole story takes place on this timeline:



The first step was a giant's step, it was made by somebody that's about 5ft tall, and it reached probably too far away as we will see in the sequel. In 1992, closely after the publication of PBC, A-nets [7] were introduced and defined a variant of PBC in which places could contain arbitrary values from arbitrary algebraic data types. This was later ported to the more practical domain of coloured Petri nets by introducing M-nets in 1995 [3]. M-nets were further extended in 2000 by introducing preemption, allowing to suspend/resume/abort arbitrary sub-nets [10]. This was obtained through the introduction of priorities between transitions, which was conceptually nice but led very far away from any hope of a reasonable or efficient implementation of the framework...

In the mean time, asynchronous communication through buffer places were added, first to PBC in 1999 [9], then to M-nets in 2002[4]. While this is not a feature that is directly related to high or low-level aspects, it paved the way to a compromise between the hairy generality of M-nets and the data-bareness of PBC. Indeed, it draws a clear distinction between buffer places that are resources and control-flow places that can remain low-level. This idea led in 2004 [5] to a formalism that can be seen as PBC extended with high-level buffer places, and preserving all the original PBC operations.

At this point, we had a practical formalism that was at the same time expressive enough to model complex systems, and disciplined enough to allow its implementation. This coincides with the starting of the development of the SNAKES toolkit [13]. Still, the preemption features were missing. They could be introduced again, without transitions priorities, in 2007 [12] by forbidding nested parallelism and by considering two-colours control-flow where black tokens $\bullet$ implemented the standard PBC control-flow and white tokens $\circ$ implemented the preemption control-flow.

The idea of pushing parallelism to the top-level exists in CSP [6] and is similar to the multi-threaded programming paradigm. But, we still lacked the ability to create new processes dynamically, which threads have and CSP does not. This feature was introduced in 2009 [11] by porting into the PBC family the ideas from [8]. Each data or control-flow token is now tagged with a process identifier. This allows the concurrent execution of many instances of a single Petri net with no undesired interference (communication between processes is possible). This solution is remarkably versatile as it proposes unique features: i) process identifiers are rich data structures that allow to trace every process' history; ii) state-space is reduced against process identifiers (more accurate that just a symmetry reduction); and iii) usual notions of threads and processes are unified. For the last feature, that allows the abortion of a working thread, there is no programming language able to implement it at the level of threads.

Research is still active in the domain, for instance, we are presently working on a new PBC variant with operations that are very different from the original control-flow ones, with the aim of developing directly with Petri nets the acting system of autonomous robots.

Looking forward, we feel very lucky to have chosen to embark the PBC ship and get inspiration and research topics for the rest of our life.

References

1. Best, E., Devillers, R., Hall, J.G.: The box calculus: a new causal algebra with multi-label communication. In: Advances in Petri Nets 1992, The DEMON Project. Springer (1992)
2. Best, E., Devillers, R., Koutny, M.: Petri net algebra. Springer (2001)
3. Best, E., Fraczak, W., Hopkins, R.P., Klaudel, H., Pelz, E.: M-nets: An algebra of high-level Petri nets, with an application to the semantics of concurrent programming languages. Acta Informatica **35**(10) (1998)
4. Bui Thanh, C., Klaudel, H., Pommereau, F.: Box calculus with coloured buffers. Tech. Rep. 16, LACL, University Paris East (2002)
5. Bui Thanh, C., Klaudel, H., Pommereau, F.: Box calculus with high-level buffers. In: DADS'04. SCS (2004)
6. Hoare, C.: Communicating Sequential Processes. Commun. ACM **21**(8) (1978)
7. Klaudel, H.: Traitement des données dans l'Algèbre des Petri Boxes. Master thesis, University Paris XI, Orsay, France (1992)
8. Klaudel, H., Koutny, M., Pelz, E., Pommereau, F.: Towards efficient verification of systems with dynamic process creation. In: ICTAC'08. LNCS, vol. 5160. Springer (2008)
9. Klaudel, H., Pommereau, F.: Asynchronous links in the PBC and M-nets. In: ASIAN'99. LNCS, vol. 1742. Springer (1999)
10. Klaudel, H., Pommereau, F.: A concurrent and compositional Petri net semantics of preemption. In: IFM'00. LNCS, vol. 1945. Springer (2000)
11. Pommereau, F.: Algebras of coloured Petri nets. Habilitation thesis, UPEC (2009)
12. Pommereau, F.: Versatile boxes: a multi-purpose algebra of high-level Petri nets. In: DADS'07. SCS/ACM (2004)
13. Pommereau, F.: SNAKES: a flexible high-level Petri nets library. In: Proc. of PETRI NETS'15. LNCS, vol. 9115. Springer (2015)