

**NAME**

pbg, PBG – Partitioned BCG File format

**DESCRIPTION**

The PBG (Partitioned BCG File) format is a software implementation of the theoretical concept of "Partitioned Labelled Transition System" introduced in [GMS01].

A file in the PBG format has a **.pbg** extension and gathers a collection of BCG files, called "fragments". Taken altogether, these fragments form a partition of a Labelled Transition System, the states and transitions of which have been distributed among the various fragments, as specified in [GMS01].

Note that, taken individually, each fragment is usually meaningless; in particular, it may be a disconnected graph, which is never the case with a state space generated using forward reachability exploration.

The PBG format is currently undocumented, and may evolve in the future. However, PBG files are text files easily readable by humans. Concretely, a PBG file is a text file that contains references to a GCF file and to BCG fragments. See **gcf(LOCAL)** and **bcg(LOCAL)** for further information about the GCF and BCG formats.

There is one BCG fragment per instance listed in the GCF file. The BCG fragments are either stored on the local machine in case of a shared filesystem (e.g., NFS, etc.) or stored on remote machines in the working directories specified in the GCF file.

**HOW TO CREATE, READ, AND HANDLE PBG FILES**

A PBG file can be created using the **distributor(LOCAL)** tool.

A PBG file and its fragments can be recombined into a single BCG file using the **bcg\_merge(LOCAL)** tool, which performs additional actions, such as state renumbering.

PBG files and their BCG associated fragments can be handled using the **pbg\_cp(LOCAL)**, **pbg\_info(LOCAL)**, **pbg\_mv(LOCAL)**, and **pbg\_rm(LOCAL)** tools.

**BIBLIOGRAPHY**

[GMS01] Hubert Garavel, Radu Mateescu, and Irina Smarandache. Parallel State Space Construction for Model-Checking. In Matthew B. Dwyer, ed, Proceedings of the 8th International SPIN Workshop on Model Checking of Software (SPIN'01), Toronto, Canada, LNCS 2057, pp. 217-234, May 2001. Revised version available as INRIA Research Report RR-4341, December 2001. Available from <http://cadp.inria.fr/publications/Garavel-Mateescu-Smarandache-01.html>

[GMB+06] Hubert Garavel, Radu Mateescu, Damien Bergamini, Adrian Curic, Nicolas Descoubes, Christophe Joubert, Irina Smarandache-Sturm, and Gilles Stragier. DISTRIBUTOR and BCG\_MERGE: Tools for Distributed Explicit State Space Generation. In Holger Hermanns and Jens Palberg, eds., Proceedings of the 12th International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS'06), Vienna, Austria, LNCS 3920, pp. 445-449, March-April 2006. Available from <http://cadp.inria.fr/publications/Garavel-Mateescu-Bergamini-et-al-06.html>

[GMS12] Hubert Garavel, Radu Mateescu, and Wendelin Serwe. Large-scale Distributed Verification using CADP: Beyond Clusters to Grids. Electronic Notes in Theoretical Computer Science, vol. 296, pp. 145-161, August 2012. Available from <http://cadp.inria.fr/publications/Garavel-Mateescu-Serwe-12.html>

**SEE ALSO**

**bcg(LOCAL), gcf(LOCAL), bcg\_merge(LOCAL) distributor(LOCAL), pbg\_cp(LOCAL),  
pbg\_info(LOCAL), pbg\_mv(LOCAL), pbg\_rm(LOCAL)**

Additional information is available from the CADP Web page located at <http://cadp.inria.fr>

Directives for installation are given in files **\$CADP/INSTALLATION\_\***.

Recent changes and improvements to this software are reported and commented in file **\$CADP/HISTORY**.

**BUGS**

Please report bugs to [cadp@inria.fr](mailto:cadp@inria.fr)