NAME

caesar − compilation & verification of LOTOS specifications

SYNOPSIS

caesar [**-aldebaran**] [**-analysis**] [**-bcg**] [**-cc** *options*] [**-comments**] [**-depend**] [**-english**] [**-error**] [**-exec**] [**-exit**] [**-external**] [**-e7**] [**-e7old**] [**-force**] [**-french**] [**-functionality**] [**-gc**] [**-gradual**] [**-graph**] [**-indent**] [**-iso**] [**-map**] [**-monitor**] [**-more** *command*] [**-network**] [**-newstyle**] [**-nupn**] [**-oldstyle**] [**-open**] [**-root** *instantiation*] [**-safety**] [**-silent**] [**-simulator**] [**-trigger** *optimization*] [**-verbose**] [**-version**] [**-v3**] [**-v4**] [**-warning**] *filename*[**.lotos**]

DESCRIPTION

caesar [Gar89b,GS90] is a compiler that translates a LOTOS specification into executable code that can be used to explore, on-the-fly or exhaustively, the graph (also called labelled transition system, reachability graph, state space, etc.) corresponding to the behaviour of this specification. **caesar** itself does not embody verification capabilities, but it smootly interfaces with many tools that can perform explicit-state and on-the-fly verification on the generated graph, including model checking, equivalence checking, and visual checking.

Taking as input *filename***.lotos**, which contains a LOTOS specification, optionally accompanied by *filename***.h**, which provides C types and functions implementing the LOTOS sorts and operations defined in *filename***.lotos**, **caesar** performs successive transformation steps and produces a C program that allows to execute, simulate, and/or build the corresponding graph. In the latter case, the result of **caesar** is the graph itself, rather than the C program, which is removed after its execution.

When generated by **caesar.adt**(LOCAL), *filename***.h** may include two files *filename***.t** and *filename***.f** if there are external sorts and/or operations declared in the LOTOS specification.

Refer to the **lotos**(LOCAL) manual page for a detailed description of the conventions to be followed by *filename***.lotos**, *filename***.h**, *filename***.t** and *filename***.f**.

OPTIONS

-aldebaran

Generate output file *filename***.aut**, which contains a graph in the AUT textual format. See the **aut**(LOCAL) manual page for a description of this format. Not a default option.

-analysis

Analyze *filename***.lotos** in order to detect errors and stop. The following phases are performed: syntactic analysis, static semantic analysis, restriction, expansion and generation. This option does not generate any output, except error diagnostics. Not a default option.

-bcg

Generate output file *filename***.bcg**, which contains a graph in the BCG (Binary-Coded Graphs) format. See the **bcg**(LOCAL) manual page for a description of this format. Default option unless one of the following options is set: **-aldebaran**, **-graph**, **-exec**, or **-open**.

-bpn

As of April 2014, this option was renamed into **-nupn** (see item #1828 in file "**$CADP/HISTORY**" for details).

-cc *options*

Pass *options* to the C compiler when it is invoked. *options* is a list of compiler options (enclosed in quotes or double quotes). These options are appended to the compiler options, if any, contained in the **$CADP_CC** environment variable (see ENVIRONMENT VARIABLES below). Not a default option.

-comments

Issue a warning message for each LOTOS sort or operation that is not properly labelled by a special comment of the form `(*!...*)`. Not a default option.

-depend

Display the list of library files included (directly or transitively) in *filename*[**.lotos**] and stop. This list may be incomplete if the LOTOS specification is syntactically incorrect. Not a default option.

-english

Print messages in English. Opposite of **-french**. This option overrides the **$CADP_LANGUAGE** environment variable (see ENVIRONMENT VARIABLES below).

-error

A file, *filename***.err**, is generated by **caesar.** It contains detailed error diagnostics. When it terminates, **caesar** displays the content of this file on the screen, using the **$CADP/src/com/cadp_more** command, unless **-error** option is set.

-exec

Generate the kernel module *filename***.c** to be used in the EXEC/CAESAR environment. Not a default option.

-exit

Perform network reduction in order to replace all tau-transitions derived from a LOTOS enabling operator ">>" by epsilon-transitions, often leading to a smaller graph. This transformation preserves safety and trace equivalence, but neither strong, branching, nor observational equivalence. Not a default option.

-external

Generate a file *filename***.c.proto** containing skeletons for EXEC/CAESAR's gate functions, i.e., functions associated to all visible gates of the LOTOS specification. These functions are incomplete and have to be completed manually (at the places marked "...") with input/output operations so as to interface the LOTOS specification with its real environment. This option must be used together with the **-exec** option. Not a default option.

Note: if *filename***.c.proto** already exists in the current directory, **caesar** will not overwrite it, because it might have been modified manually.

-e7

Do not perform the new (BDD based) optimization E7, which removes dead transitions, still preserving strong equivalence. Alternatively, one may set the environment variable **$CAESAR_BDD_TIMEOUT** to impose a time limit on optimization E7, which will stop whenever this limit is exceeded, also preserving strong equivalence; see the **caesar.bdd**(LOCAL) manual page for details. Not a default option.

-e7old

Perform the old (explicit-state based) optimization E7 instead of the new (BDD based) optimization E7. Not a default option.

-force

Force **caesar** to regenerate *filename***.c** even if not necessary. This option is only meaningful if **caesar** is used with options **-exec** (EXEC/CAESAR mode), **-open** (OPEN/CAESAR mode), or **-simulator**. Not a default option. By default **caesar** will attempt not to regenerate *filename***.c** if this file already exists in the current directory, and if it has been modified more recently than
(1) the corresponding LOTOS file (*filename***.lotos**, *filename***.lot**, or *filename***.l**),
(2) than any LOTOS library transitively included (using the "library" clause) in this LOTOS file, and
(3) than any C file transitively included (using the "`#include`" clause) in *filename***.c** itself.

-french

Print messages in French. Opposite of **-english**. This option overrides the **$CADP_LANGUAGE** environment variable (see ENVIRONMENT VARIABLES below). Even when this option is set, some warning and error messages related to lexical and syntactic analysis may still be displayed in English.

-functionality
Do not check functionality constraints (“`exit`” and “`noexit`”). Not a default option.

-gc Invoke the Boehm-Demers garbage collector to reuse the (potentially large) amounts of memory that have been allocated for abstract data types and that are no longer used. This option is especially suitable for LOTOS descriptions involving dynamic data structures, such as: lists, queues, stacks, etc. Not a default option.

-gradual
Apply the optimizations gradually when generating the network. By default, the optimizations are applied only after the network is fully generated. Using this option, the optimizations are applied to each sub-network generated from each operand of a parallel composition operator. This option is slower, but it can be useful for dealing with larger LOTOS descriptions and/or for generating smaller networks. Not a default option.

-graph
Generate output file *filename***.gph**, which contains a graph in a textual, undocumented format intended for debugging purpose; this format contains a lot of information but is expensive in disk space. Not a default option.

-indent Do not format using the shell-script located in **$CADP/src/com/cadp_indent** the file *filename***.c.proto** generated by option **-external**. Not a default option.

-iso Use the standard LOTOS semantics as defined in ISO/IEC International Standard 8807, disabling the various language enhancements mentioned in the section EXTENSIONS TO LOTOS of the **lotos**(LOCAL) manual page and implemented in **caesar**. Not a default option. Not to be used when processing LOTOS specifications generated by **lnt2lotos**(LOCAL)

-map
Generate *filename***.map**, which gives correspondence between sort and operation names occuring in *filename***.lotos** and C type and function names occuring in *filename***.h**. Not a default option.

-monitor
Open a window for monitoring in real-time the generation of a BCG graph (see option "**-bcg**" above). Not a default option.

-more *command*
Use *command* to display the error messages, instead of "**$CADP/src/com/cadp_more**", which is the default. *command* is a shell command (preferably enclosed in quotes or double quotes) containing the pathname of the chosen pager, possibly followed by a list of options. Not a default option.

-network
Generate output file *filename***.net**, which contains an Interpreted Petri Net in a textual, undocumented format. Not a default option.

-newstyle
When generating skeletons for EXEC/CAESAR’s gate functions (see **-external** option above), use the new-style function declarations (i.e., with prototypes) introduced in ANSI/ISO Standard C. This option must be used together with the **-external** option. Default option when option **-external** is selected.

-nupn
Generate output file *filename***.nupn**, which contains a Nested-Unit Petri Net in the NUPN format documented in the **caesar.bdd**(LOCAL) manual page. Not a default option.

-oldstyle

When generating skeletons for EXEC/CAESAR's gate functions (see **-external** option above), use the old-style function declarations (i.e., without prototypes) available in Kerninghan and Ritchie C. This option is only applicable when gate functions are not overloaded, i.e., when each gate is always used with the same number of offers, and with offers of the same sorts and same directions (input or output). This option must be used together with the **-external** option. Not a default option.

-open

Generate the graph module *filename***.c** to be used in the OPEN/CAESAR environement. Not a default option.

-root *instantiation*

Ignore the behaviour expression following the **`behaviour`** (or **`behavior`**) keyword in the LOTOS specification contained in *filename***.lotos**, and replace this behaviour by the one given by *instantiation*, which is a character string (preferably enclosed between single quotes) denoting a LOTOS process instantiation. This string may have the form '**`P`**', or '**`P [G1, ..., Gm]`**', or '**`P (V1, ..., Vn)`**', or '**`P [G1, ..., Gm] (V1, ..., Vn)`**', where **`P`** is either the identifier of the LOTOS specification or the identifier of a LOTOS process declared at the top-level of the specification (processes nested within another process are not accepted, and the specification identifier has precedence over a process identifier having the same name); where **`[G1, ..., Gm]`** is a list of gate identifiers that must either have the same number of elements as the list of formal gate parameters of **`P`** or be the empty list (in such case, it is replaced by the list of formal gate parameters of **`P`**); and where **`(V1, ..., Vn)`** is a list of LOTOS value expressions that must be algebraically-closed (i.e., contain no variables) and be compatible, in number and types, with the list of formal variable parameters of **`P`**. Not a default option. The particular case where *instantiation* is equal to the string '**`-`**' is accepted, but left undocumented.

-safety

Perform network reduction in order to replace all tau-transitions by epsilon-transitions, often leading to a smaller graph. This transformation preserves safety and trace equivalence, but neither strong, branching, nor observational equivalence. Not a default option.

-silent

Execute silently. Opposite of **-verbose**. Default option is **-verbose**.

-simulator

Generate the simulator program *filename***.c** and stop, neither compiling, executing, nor removing this file. This option can be useful to port *filename***.c** to another (e.g., more powerful) machine, on which the simulator program can be compiled and executed. Not a default option.

-trigger *optimization*

Print statistics about which optimizations cause a given *optimization* to become effective. This option is only useful for CAESAR's development. Not a default option.

-verbose

Print one line for each successive phase performed by **caesar** to inform the user about the progress of activities.

-version

Display the current version number of the software and stop. Not a default option.

-v3 Do not perform optimization V3, which discovers variables whose values remain constant during the simulation phase, and replaces them by constants. Not a default option.

-v4 Do not perform optimization V4, which evaluates statically guards whose value is constant and removes transitions whose guard is false. Not a default option.

-warning

Suppress all warning messages, keeping (more severe) error messages, at the risk of leaving undetected issues in the LOTOS specification. Not a default option.

TRANSLATION PHASES

The architecture of **caesar**(LOCAL) follows the principles exposed in [Gar89b] and Section 3 of [GS90]. The translation proceeds in several successive phases:

- syntax analysis phase

  The LOTOS specification is lexically and syntactically analyzed using a scanner and a parser that have been generated by the SYNTAX tool of INRIA, which produces analyzers that emit pertinent error messages and perform, as much as possible, automatic error recovery. Incorrect LOTOS specifications are rejected; otherwise, an abstract syntax tree is built. This phase is shared with **caesar.adt**(LOCAL)

- semantic analysis phase

  The static semantics constraints of the standard LOTOS definition are checked on the abstract syntax tree. This is done in several steps: binding of processes, binding of gates, binding of types, analysis of type signatures, binding of sorts, binding of variables, binding of operations, and analysis of process functionality. The LOTOS specifications not matching these constraints are rejected. This phase is also shared with **caesar.adt**(LOCAL)

- restriction phase

  The additional static semantics constraints listed in the section "STATIC-CONTROL CONSTRAINTS" of the **lotos**(LOCAL) manual page are checked, and the LOTOS specifications not satisfying these constraints are rejected. Also, warnings are emitted for the LOTOS processes that cannot be called from the entry point of the specification.

- expansion phase

  The LOTOS syntax tree is translated into another syntax tree implementing SUBLOTOS, a simplified language derived from LOTOS. Translation is done by replacing certain LOTOS operators by semantically-equivalent forms, and finitely unfolding certain process calls to obtain a statically-fixed hierarchy of concurrent processes in which each gate, variable, and process identifier plays a unique role.

- type survey phase

  If a file named *filename*.**h** exists, an auxiliary C program that includes this file is generated, compiled, and executed so as to obtain information on how LOTOS sorts are implemented in *filename*.**h** (which may itself include other files, such as *filename*.**t** or *filename*.**f**). This phase may fail if the contents of *filename*.**h** are incorrect or incomplete.

- generation phase

  The SUBLOTOS syntax tree is translated into an interpreted Petri net extended with epsilon-transitions and typed variables that can be read or written on transitions, and hierarchically structured into nested units. This network model provides a compact representation of the control and data flows. During this phase, warnings are emitted when certain actions cannot be synchronized with a compatible action (i.e., same gate and same offer types), as this situation is likely to cause local or global deadlocks.

- optimization phase

  The network model produced by the generation phase is simplified by applying a collection of transformations, each focusing on a particular aspect of the control or data flow. These optimizations preserve strong equivalence unless the **-safety** option is set. Certain transformations generate an auxiliary C program that is compiled and executed. The generation and optimization phases can be intertwined by setting the **-gradual** option.

- simulation phase

  The optimized network model is translated into a C program called *simulator*. With certain options (e.g., **-exec**, **-open**, or **-simulator**), the translation terminates there. Otherwise, the simulator program is compiled and executed to explore all the reachable states of the network and store the corresponding graph in a file, using the format requested by the user. If the graph is large, the simulation phase may take a long time. Because each new state is kept in memory, the simulation phase should always terminate, either because the available memory is sufficient to contain all reachable states, or because memory gets exhausted before the entire graph has been explored; in such case, **caesar** tries to properly close the file under generation to leave it in a coherent state. However, the simulation may loop forever if one of the C functions defined in *filename***.h** to implement LOTOS operations does not terminate.

ENVIRONMENT VARIABLES

$CADP_LANGUAGE

If this variable is set, its value determines the language in which diagnostic messages will be reported. Possible values are '**english**' and '**french**'. Incorrect values will be ignored silently. If this variable is unset, it is given the default value '**english**'.

$CADP_CC

If this variable is set, its value determines the name of the C compiler that will be invoked by **caesar**. See file **$CADP/INSTALLATION_2** for detailed information about this variable. If this variable is unset, the script-shell **$CADP/src/com/cadp_cc** will automatically determine the C compiler to be used by default.

$CADP_TMP

If this variable is set, its value determines the directory in which temporary files are created. If this variable is unset, it is given the default value '**/tmp**'.

$PAGER

If this variable is set, its value will be used by the script-shell **$CADP/src/com/cadp_more** to display error and warning messages.

EXIT STATUS

When the source is erroneous, error messages are issued. Exit status is 0 if everything is alright, 1 otherwise.

AUTHORS

Hubert Garavel and Wendelin Serwe (INRIA Rhone-Alpes)

OPERANDS

*filename***.lotos**	LOTOS specification (input)
*filename***.h**	implementation in C of data types (input)
*filename***.c**	C code of the simulator (output of **-simulator**)
*filename***.c**	C code of the graph module (output of **-open**)
*filename***.c**	C code of the kernel module (output of **-exec**)
*filename.***c.proto**	skeleton for gate functions (output of **-external**)
*filename***.aut**	graph in AUT format (output)
*filename***.bcg**	graph in BCG format (output)
*filename***.net**	Interpreted Petri Net (output)
*filename***.nupn**	Nested-Unit Petri Net (output)
*filename***.gph**	graph in debugging format (output)
*filename***.err**	detailed error messages (output)
*filename***.map**	ADT correspondence table (output)
*libname***.lib**	user ADT library (input)

For simplicity, the standard error stream is not used; all messages are written to the standard output stream, which is made unbuffered. The file *filename.***err** is created at the beginning of execution and removed, if empty, at the end of execution.

FILES

$CADP/lib/*libname***.lib**	predefined ADT library (input)
$CADP/src/com/cadp_cc	C compiler shell
$CADP/src/com/cadp_more	pager shell
$CADP/gc	Boehm-Demers garbage collector (input)
$CADP/LICENSE	license file
$CADP_TMP/*.c	C code generated during type survey (temporary)
$CADP_TMP/*.c	C code generated during optimization (temporary)
$CADP_TMP/*.x	binary code for type survey (temporary)
$CADP_TMP/*.x	binary code for optimization (temporary)
$CADP_TMP/*.x	binary code of simulator program (temporary)
$CADP_TMP/*.inf	results of information phase (temporary)
$CADP_TMP/*.trt	contents of -root option (temporary)
$CADP_TMP/*.tsv	results of type survey (temporary)
$CADP_TMP/*.nupn	Nested-Unit Petri Net for optimization (temporary)
$CADP_TMP/*.oe7	results of old optimization E7 (temporary)
$CADP_TMP/*.ov3	results of optimization V3 (temporary)
$CADP_TMP/*.ov4	results of optimization V4 (temporary)
$CADP_TMP/*.ov7	results of optimization V7 (temporary)

BIBLIOGRAPHY

[Gar89b] Hubert Garavel. Compilation et verification de programmes LOTOS. These de doctorat, Universite Joseph Fourier, Grenoble, November 1989. Available from http://cadp.inria.fr/publications/Garavel-89-b.html

[GS90] Hubert Garavel and Joseph Sifakis. Compilation and Verification of LOTOS Specifications. In L. Logrippo, R. L. Probert, and H. Ural, editors, Proceedings of the 10th IFIP International Symposium on Protocol Specification, Testing and Verification (PSTV'90), Ottawa, Canada. North Holland, pages 379-394, June 1990. Available from http://cadp.inria.fr/publications/Garavel-Sifakis-90.html

[GS04] Hubert Garavel and Wendelin Serwe. State Space Reduction for Process Algebra Specifications. In Charles Rattray, Savitri Maharaj, and Carron Shankland, editors, Proceedings of the 10th International Conference on Algebraic Methodology and Software Technology (AMAST'04), Stirling, Scotland, UK. Lecture Notes in Computer Science, vol. 3116, pages 164-180, Springer, July 2004. Available from http://cadp.inria.fr/publications/Garavel-Serwe-04.html

[GS06] Hubert Garavel and Wendelin Serwe. State Space Reduction for Process Algebra Specifications. Theoretical Computer Science, vol. 351, num. 2, pages 131-145, February 2006. Available from http://cadp.inria.fr/publications/Garavel-Serwe-06.html

SEE ALSO

OPEN/CAESAR Reference Manual, **aut**(LOCAL), **bcg**(LOCAL), **caesar.adt**(LOCAL), **caesar.bdd**(LOCAL), **caesar.indent**(LOCAL), **lotos**(LOCAL), **lotos.open**(LOCAL), **lnt2lotos**(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 new bugs to cadp@inria.fr