NAME

caesar.adt – translation of LOTOS abstract data types into C

SYNOPSIS

caesar.adt [**-cc** *options*] [**-comments**] [**-debug**] [**-depend**] [**-english**] [**-error**] [**-external**] [**-force**] [**-french**] [**-functionality**] [**-indent**] [**-infix**] [**-iso**] [**-macro**] [**-map**] [**-more** *command*] [**-numeral** *integer*] [**-prefix**] [**-silent**] [**-trace**] [**-verbose**] [**-version**] [**-warning**] *filename*[**.lotos**]

DESCRIPTION

caesar.adt [Gar89c,GT93] is a compiler that translates LOTOS abstract data types into executable code. Quite often, the code generated is given as input to the **caesar**(LOCAL) compiler, but it can also be used for other purpose.

Taking as input *filename***.lotos**, which contains a LOTOS specification, optionally accompanied by *filename***.t**, which contains hand-written C code, **caesar.adt** produces an output file *filename***.h** that contains C types and functions implementing the LOTOS sorts and operations defined in *filename***.lotos**.

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

OPTIONS

-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 which is not properly labelled by a special comment of the form `(*!...*)`. Not a default option.

-debug

Generate extra C code that helps to debug partial non-constructor definitions. When a C function aborts because the corresponding LOTOS operation is not completely defined by its equations, the name of the function and the actual values of its arguments are displayed. 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.adt.** It contains detailed error diagnostics. When it terminates, **caesar.adt** displays the content of this file on the screen, using the **$CADP/src/com/cadp_more** command, unless **-error** option is set.

-external

Generate a skeleton file *filename***.t.proto** and/or a skeleton file *filename***.f.proto** if the LOTOS specification contains sorts and/or operations declared `(*! external *)`. These skeleton files are incomplete, but form a basis for producing *filename***.t** and *filename***.f**. They have to be completed manually (at the places marked "...") with an implementation in C for external sorts and constructors, and/or operations. Also, it may be necessary to modify manually the order of C type

declarations in order to avoid forward references. Not a default option.

Note: if *filename***.t.proto** and/or *filename***.f.proto** already exist in the current directory, **caesar.adt** will not overwrite them, because they might have been modified manually.

-force

Force **caesar.adt** to regenerate *filename***.h** even if not necessary. Not a default option. By default **caesar.adt** will attempt not to regenerate *filename***.h** 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,
(3) than any C file included (using the "**`#include`**" clause) in *filename***.h** itself,
(4) than the *filename***.t** file if this file exists in the current directory, and
(5) than the *filename***.f** file if this file exists in the current directory.

-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.

-indent Do not format using the shell-script located in **$CADP/src/com/cadp_indent** the generated file *filename***.h**, nor the files *filename***.f.proto** and *filename***.t.proto** generated by option **-external**. This option can be useful when the **indent**(1) program invoked by **cadp_indent** crashes with a core dump, although **cadp_indent** is designed to recover properly in such event. Not a default option.

-infix Generate C code to print LOTOS binary operations in infix form when appropriate. 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.adt**. Not a default option. Not to be used when processing LOTOS specifications generated by **lnt2lotos**(LOCAL)

-macro

Prevent LOTOS non-constructor operations to be implemented by C macro-definitions (**`#define`**): all LOTOS non-constructor operations will be implemented as C functions instead. Not a default option.

-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.

-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.

-numeral *integer*

Specify the range of values to be used for implementing numeral sorts, i.e., all sorts *S* that have two constructor operations *F1* : -> *S* and *F2* : *S* -> *S*, and are thus isomorphic to natural numbers. If *integer* is greater than zero, the range of values will be 0...(*integer*-1). If *integer* is less than zero, the range of values will be 0...((2^(-*integer*))-1). With 64-bit versions of **caesar.adt**, the highest positive values for *integer* are interpreted as the negative numbers -64, -63, ..., -2, -1. By default, numeral sorts are represented using a single byte (default value of *integer* is 256). This option does

not apply to those numeral sorts *S* for which *filename.***t** defines a corresponding macro **CAESAR_ADT_HASH_...**.

-prefix Generate C code that prints LOTOS binary operations always in prefix form. Default option.

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

-trace
Generate extra C code that traces all calls and returns for a selected set of C functions. This option also sets the **-macro** option. Not a default option.

-verbose
Print one line for each successive phase performed by **caesar.adt** to inform the user about the progress of activities. Opposite of **-silent**. Default option is **-verbose**.

-version
Display the current version number of the software and stop. 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.adt**(LOCAL) follows the principles exposed in Section 1 of [GT93]. The translation from LOTOS to C 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**(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**(LOCAL)

- interface phase
The abstract syntax tree is traversed and its fragment that represents the abstract data types defined in the source LOTOS specification is extracted and copied into the *input tree*, a simpler syntax tree, which is itself specified using LOTOS abstract data types. From this point, all the forthcoming translation steps are written in LOTOS, meaning that the **caesar.adt** translator compiles itself (i.e., is bootstrapped).

- verification phase
The additional static semantics constraints listed in the section "RESTRICTIONS ON THE DATA PART" of the **lotos**(LOCAL) manual page are checked, and the LOTOS specifications not satisfying these constraints are rejected.

- type survey phase
If a file named *filename***.t** exists, an auxiliary C program that includes this file is generated, compiled, and executed so as to obtain information on how external LOTOS sorts are implemented in

filename.**t**. This phase may fail if the contents of *filename*.**t** are incorrect or incomplete.

- compilation phase

The abstract sorts and operations represented in the input tree are translated into concrete types and functions, which are stored in the *output tree*, another syntax tree closer to imperative languages, such as C.

- optimization phase

Various transformations are applied to the output tree, so as to reduce the space taken by types and the time spent in functions.

- C translation phase

The output tree is traversed and decompiled to produce C code stored in *filename*.**h**.

- indentation phase

The shell-script **cadp_indent** is invoked to format the generated file *filename*.**h** unless option **-indent** is set.

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.adt**. 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.

OPERANDS

filename.**lotos**	LOTOS specification (input)
filename.**t**	external C implementation for types (input)
filename.**t.proto**	skeleton for *filename*.**t** (output)
filename.**f**	external C implementation for functions (input)
filename.**f.proto**	skeleton for *filename*.**f** (output)
filename.**h**	C implementation (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/LICENSE	license file
$CADP_TMP/*.c	C code generated during type survey (temporary)
$CADP_TMP/*.x	binary code for type survey (temporary)
$CADP_TMP/*.tsv	results of type survey (temporary)

BIBLIOGRAPHY

[Gar89c] Hubert Garavel. Compilation of LOTOS Abstract Data Types. In Son T. Vuong, editor, Proceedings of the 2nd International Conference on Formal Description Techniques (FORTE'89), Vancouver, Canada. North Holland, pages 147-162, December 1989. Available from http://cadp.inria.fr/publications/Garavel-89-c.html

[GT93] Hubert Garavel and Philippe Turlier. CAESAR.ADT : un compilateur pour les types abstraits algebriques du langage LOTOS. In Rachida Dssouli and Gregor v. Bochmann, editors, Actes du Colloque Francophone pour l'Ingenierie des Protocoles (CFIP'93), Montreal, Canada, 1993. Available from http://cadp.inria.fr/publications/Garavel-Turlier-93.html

SEE ALSO

caesar(LOCAL), **caesar.indent**(LOCAL), **lotos**(LOCAL) **lotos.open**(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