

NAME

fsp2lotos – FSP to LOTOS translator

SYNOPSIS

fsp2lotos [-version] [-pidlist] [-root *root-process*] *filename* [.its]

DESCRIPTION

FSP (Finite State Processes) is a specification language for concurrent processes defined by Magee & Kramer (Imperial College, London). The **fsp2lotos** program translates an FSP specification to files that can be verified using CADP.

The input to **fsp2lotos** is an FSP file, which must have the extension **.its**. If the user does not specify the extension **.its** on the command line, it will be appended automatically and **fsp2lotos** will read *filename.its* as input.

When **fsp2lotos** is called with the **-root** option and the name of a process *root-process*, then **fsp2lotos** translates *root-process*. Otherwise, **fsp2lotos** translates the first sequential process defined in *filename.its*. In both cases, we call main process the process that **fsp2lotos** translates. The outputs are the following files:

- a LOTOS library file *FILENAME.lib* (where *FILENAME* corresponds to the character string defined in the note below), which contains LOTOS type definitions,
- a LOTOS file *filename.lotos*, which contains LOTOS processes;
- two C files *filename.t* and *filename.f*, which contain C types and functions needed by the LOTOS types and operations defined as external in *FILENAME.lib*,
- an EXP.OPEN file *filename.exp*, which contains a network of communicating automata describing the behaviour of the main process, and
- an SVL file *filename.svl*, which contains an SVL script that generates the BCG graphs contained in *filename.exp*.

The output files are placed in the directory whose path is given by the environment variable **\$FSPGEN**, or in **./FSPGEN** if this variable is not defined. Note that **./FSPGEN** is created relative to the directory from which the user calls **fsp2lotos**, not relative to the directory containing the input file.

The SVL script can be run using the following Bourne shell commands:

```
cd ${FSPGEN:-FSPGEN}
svl svl-options
```

See **svl(LOCAL)** for more information about the SVL tool and its options.

Once the SVL script has been executed, the generated file *filename.exp* can be manipulated as usual, using the EXP.OPEN tool. For instance, the BCG graph corresponding to the main process can be generated by running the following shell command:

```
exp.open filename.exp generator filename.bcg
```

See **exp.open(LOCAL)** for more information about the EXP.OPEN tool and its options.

Note: The name of the input file is used to construct the names of the output files, with particular rules when constructing the names of **.lib** files. For an input file *filename.its*, **fsp2lotos** creates the LOTOS library *FILENAME.lib*, where *FILENAME* is constructed as follows:

- *FILENAME* corresponds to *filename* with all letters turned to upper case and all non-alphanumeric characters replaced by '_'. For instance, for an FSP specification **fsp-example.its**, **fsp2lotos** creates the LOTOS library **FSP_EXAMPLE.lib**.
- In the particular case where *filename* is not a valid LOTOS identifier, the prefix **LIB_** is appended to *FILENAME*. For instance, for an FSP specification **forall.its** (**forall** being a LOTOS keyword),

fsp2lotos creates the LOTOS library **LIB_FORALL.lib**. For an FSP specification **0407.lts**, **fsp2lotos** creates the LOTOS library **LIB_0407.lts**.

OPTIONS

-version

Display the tool version and exit.

-pidlist List the names of all processes that occur in the input file and exit. In other words, list all processes that can be used as main process (by instantiation with the **-root** option).

-root *root-process*

Use the process *root-process* as the main process, meaning that the generated files have the behaviour of the main process. Not a default option.

NOTES ABOUT THE TRANSLATION FROM FSP TO LOTOS

LANGUAGE ACCEPTED AS INPUT

The **fsp2lotos** translator supports the FSP language recognized by version 2.3 of LTSA.

In rare cases, **fsp2lotos** is unable to bind a local process call to its definition. If so, it issues an error message and then exits.

The following FSP constructs are not translated by **fsp2lotos**:

- the "@" (set cardinality) operator
- fluents, menus, progress and properties

All correct FSP descriptions are accepted by **fsp2lotos** provided the root process is independent of those constructs. If the root process depends on any of those constructs, then **fsp2lotos** issues an error message and then exits.

SEMANTIC CHECKS

The **fsp2lotos** translator does not fully check the correctness of the input FSP specification (parsing, type checking, binding, etc.). The "Build/Parse" menu of the LTSA tool should be used for such verifications. As a consequence, incorrect FSP input files may be accepted by **fsp2lotos**, in which case the result of the translation is irrelevant.

TRANSLATION OF THE ERROR STATE

The error state, represented by a state numbered -1 in LTSA, is represented by a state containing a self arc labeled by ERROR in the BCG graph corresponding to the root process.

TRANSLATION OF THE NON-OBSERVABLE LABEL

The non-observable label noted "tau" in FSP is represented as the LOTOS label "i" in the BCG graph corresponding to the root process.

TRANSLATION OF OBSERVABLE LABELS

An observable FSP label is a sequence of elementary labels separated by the '.' symbol. Formally, any observable FSP label has the form $A_1.A_2...A_n$, where n is greater or equal to 1, and where each elementary label A_i is either a number (e.g., "-4", "0", "1") or an FSP identifier written in lower-case letters (e.g., "x", "y1", "read").

Each observable FSP label $A_1.A_2...A_n$ is translated into a LOTOS label of the form "EVENT ! $L_1.L_2...L_n$.NIL", where each L_i is derived from the corresponding A_i according to the following rules: if A_i is a number, then L_i is the same number; if A_i is an FSP identifier, then L_i is the same identifier converted to

upper-case letters. Such identifiers are declared in the *Symbol* LOTOS type defined in *FILENAME.lib*.

OPERANDS

<i>filename.lts</i>	FSP specification (input)
<i>\$FSPGEN/filename.exp</i>	Network of LTS (output)
<i>\$FSPGEN/filename.lotos</i>	LOTOS code (output)
<i>\$FSPGEN/FILENAME.lib</i>	LOTOS code (output)
<i>\$FSPGEN/filename.svl</i>	SVL script (output)
<i>\$FSPGEN/filename.f</i>	C code (output)
<i>\$FSPGEN/filename.t</i>	C code (output)

FILES

<i>\$CADP/lib/FSP_V1.lib</i>	FSP predefined library (LOTOS code)
<i>\$CADP/incl/FSP_V1.h</i>	FSP predefined library (C code)

ENVIRONMENT VARIABLES

<i>\$FSPGEN</i>	The target directory of the output files.
------------------------	---

EXIT STATUS

If the translation was successful the exit status is 0, even if warnings were issued during the execution. If any error occurred during translation, the exit status is 1.

BIBLIOGRAPHY

- A description of the FSP language is available in: Jeff Magee and Jeff Kramer. *Concurrency: State Models & Java Programming (2nd Edition)*. John Wiley & Sons, 2006.
- The grammar of the FSP language is available from http://www.doc.ic.ac.uk/ltsa/eclipse/help/appendix_b___fsp_language_spec.htm
- The website of the LTSA (*Labelled Transition System Analyser*) software tool is available from <http://www.doc.ic.ac.uk/ltsa>
- An early description of the FSP to LOTOS translation is given in: Gwen Salaun, Jeff Kramer, Frederic Lang, and Jeff Magee. *Translating FSP into LOTOS and Networks of Automata*. In Proceedings of IFM'07, Lecture Notes in Computer Science vol. 4591. Springer Verlag, 2007. Available from <http://cadp.inria.fr/publications/Salaun-Kramer-Lang-Magee-07.html>
- A more complete description of the FSP to LOTOS translation is given in: Frederic Lang, Gwen Salaun, Remi Herilier, Jeff Kramer, and Jeff Magee. *Translating FSP into LOTOS and Networks of Automata*. In Journal on Formal Aspects of Computing, Springer Verlag, 2010. Available from <http://cadp.inria.fr/publications/Lang-Salaun-Herilier-et-al-10.html>

AUTHORS

Remi Herilier, Frederic Lang, and Gwen Salaun, with contributions from Hubert Garavel (all at INRIA Rhone-Alpes).

SEE ALSO

See the following reference manuals: **bcg**(LOCAL), **caesar**(LOCAL), **caesar.adt**(LOCAL), **exp**(LOCAL), **exp.open**(LOCAL), **fsp.open**(LOCAL), **lotos**(LOCAL), **lotos.open**(LOCAL), **svl**(LOCAL), **svl-lang**(LOCAL), **xeuca**(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

The tool **fsp2lotos** may accept FSP specifications that are rejected by LTSA. In this case, incorrect code may be generated.

Please report any mistranslations or other problems with **fsp2lotos** to cadp@inria.fr