NAME

lnt2lotos − LNT to LOTOS translator

SYNOPSIS

lnt2lotos [**−silent** | **−verbose**] [**−version**] [**−depend**] [**−force**] [**−more** *command*] [**−pidlist**] [**−root** *instantiation*] *filename*[**.lnt**]

DESCRIPTION

LNT (LOTOS New Technology, formerly noted LOTOS NT) is an imperatively styled specification language for concurrent processes. The **lnt2lotos** program translates a LNT specification to a LOTOS specification.

The input to **lnt2lotos** is a LNT file, whose name should contain only letters, digits, and underscores, and which must have the extension **.lnt**. If the user does not specify the extension **.lnt** on the command line, this extension will be appended automatically, so that **lnt2lotos** will read *filename***.lnt** as input. External C code can be provided by auxiliary files, namely *filename***.tnt** for data type definitions and *filename***.fnt** for function definitions.

Typically, the input should have been pre−processed by **lpp** before being passed to **lnt2lotos**. The pre−processor expands extended notation such as literal numbers and strings, which is not accepted by **lnt2lotos** directly.

The principal output of **lnt2lotos** is a LOTOS specification named *filename***.lotos** (unless the **-root null** option is used, in which case a LOTOS library *FILENAME***.lib** is generated instead; see below for further details). Two auxiliary files are also generated, namely *filename***.t**, which contains C code for external data types, and *filename***.f**, which contains C code for external functions. Note that *filename***.t** includes *filename***.tnt** (if present), and that *filename***.f** includes *filename***.fnt** (if present).

To avoid confusion between source code and generated code, all output files created by **lnt2lotos** will be placed in a special directory that **lnt2lotos** creates if it does not exist already. If the creation of the directory fails, **lnt2lotos** issues an error message and stops. The name of this directory is either given by the environment variable **$LNTGEN**, if this variable is set, or is **./LNTGEN** by default. Note that **./LNTGEN** (or **$LNTGEN**) is created relative to the directory from which the user calls **lnt2lotos**, not relative to the directory containing the input file.

The name of the input file is used to construct the names of the output files, with the particular rule that all letters are turned to upper case when constructing the names of **.lib** files. For an input file **example.lnt**, **lnt2lotos** creates the LOTOS library **EXAMPLE.lib** or the LOTOS specification **example.lotos**, and two auxiliary files **example.t** and **example.f**.

To avoid clashes between generated files and user-written files, **lnt2lotos** writes a special tag at the beginning of each generated file. This tag is a comment containing the name and the version of **lnt2lotos** that generated the file. **lnt2lotos** uses this tag for two purposes:

-       To prevent **lnt2lotos** from overwriting a file that was not generated by itself: if the output file already exists but has no special tag or has an invalid tag indicating that the file was not generated by the right tool, **lnt2lotos** issues an error message and stops.

-       To avoid unnecessary compilations: **lnt2lotos** recompiles a LNT file only if the source file was modified since the last translation, or if the output file was generated by an older version of **lnt2lotos**.

OPTIONS

−silent   Execute silently, reporting only errors. This is the opposite of **−verbose**. The default option is **−verbose**.

−verbose

Report activities and progress, including errors, to the user's screen. This is the opposite of **−silent**. The default option is **−verbose**.

−version

Display the tool version and exit.

−depend

List all the LNT files transitively included in *filename***.lnt** and exit. Not a default option.

−force Overwrite the output files, even if they were edited by the user or do not need to be updated.

-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.

−pidlist List the names of all processes without value parameters that occur in the input file and exit. In other words, list all processes that can be used as a main process having no value parameter (by instantiation with the **−root** option). This option is used by the EUCALYPTUS graphical user interface.

−root *instantiation*

where *instantiation* can take three different forms: **module**, **null**, or a character string of the form "*P [G1, ..., Gm] (V1, ..., Vn)*" according to the syntax for a process instantiation given in the reference manual [Champelovier-Clerc-Garavel-et-al-10]. By default, if the "**−root**" option is absent, it is assumed to be of the third form and identical to "**−root MAIN**".
If the option "**−root module**" is specified, **lnt2lotos** will generate a LOTOS library (i.e., a "**.lib**" file without main behaviour). If the LNT specification contains a process called **MAIN**, it will be treated like an ordinary process.
If the option "**−root null**" is specified, **lnt2lotos** will generate a LOTOS specification whose main behaviour is "**stop**". If the LNT specification contains a process called **MAIN**, it will be treated like an ordinary process.
In the third case, **lnt2lotos** will generate a LOTOS specification whose main behaviour is the instantiation of process *P* with actual gate identifiers *[G1, ..., Gm]* and actual value parameters *(V1, ..., Vn)*.
As processes cannot be overloaded in LNT, there must be at most one process called *P* in the LNT specification, either directly defined in *filename***.lnt** or defined in a included module included transitively.
The list of actual gate parameters is optional; if this list is missing and if *filename***.lnt** does not contain a process named *P*, an empty list of gate parameters is assumed; if this list is missing and if *filename***.lnt** contains a process named *P*, **lnt2lotos** will replace a missing list of actual gate parameters by the list of formal gate parameters of the process *P*. If process *P* is defined in *filename***.lnt**, the list of actual gate parameters can also be given using named-style parameters ("**=>**") and ellipses ("**...**").
The list of actual value parameters is also optional; if this list is missing, an empty list of value parameters is assumed. It should only contain algebraically-closed terms (i.e., contain no variables) and be compatible, in number and types, with the list of formal variable parameters of process *P*. Process *P* should have only **in** parameters (i.e., no **out** or **inout** parameter). If the actual value parameters use rich-term syntax notations [Champelovier-Clerc-Garavel-et-al-10, chapter 3], these notations must be expanded before invoking **lnt2lotos**; this is automatically performed by **lnt.open**(LOCAL) which calls **lpp**(LOCAL) to expand rich-term syntax before invoking **lnt2lotos**.

OPERANDS

*filename***.lnt**	LNT specification (input)
*filename***.tnt**	C code for data types (input)
*filename***.fnt**	C code for functions (input)

$LNTGEN/∗.sig	imported modules signatures (input)
$LNTGEN/*filename***.lotos**	LOTOS code (optional output)
$LNTGEN/*FILENAME***.lib**	LOTOS code (optional output)
$LNTGEN/*filename***.t**	C code (output)
$LNTGEN/*filename***.f**	C code (output)
$LNTGEN/*filename***.err**	detailed error messages (output)
$LNTGEN/*filename***.sig**	module signature (output)

FILES

> **$CADP/lib/LNT_V1.lib**
>
> > LNT predefined library (LOTOS code)
>
> **$CADP/incl/LNT_V1.h**
>
> > LNT predefined library (C code)

ENVIRONMENT VARIABLES

> **$LNTGEN**
>
> > 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.

AUTHORS

> David Champelovier, Xavier Clerc, Hubert Garavel, Gideon Smeding, Frederic Lang, Wendelin Serwe (INRIA Rhone−Alpes)

SEE ALSO

> **caesar.adt**(LOCAL), **caesar**(LOCAL), **lotos**(LOCAL), **lnt.open**(LOCAL), **lpp**(LOCAL), and the "Reference Manual of the LNT to LOTOS Translator" available from http://cadp.inria.fr/publications/Champelovier-Clerc-Garavel-et-al-10.html
>
> 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 type system of **lnt2lotos** is not implemented in full detail, hence, some incorrect LNT programs will be accepted by **lnt2lotos** and translated into LOTOS. However, these errors will be detected by the LOTOS compilers **caesar** and **caesar.adt**.
>
> Please report any mistranslations or other problems with **lnt2lotos** to cadp@inria.fr