

Step to Core Translation User Guide

The program is used for Step to Core translation. It takes program in Step and creates 3 output files: XML version of this program, XML translation of this program in Core language and XLINK file that contains information about all non-trivial conversions between the languages.

Run the program:

From command line from Windows

```
> StepToCore <Step input file> <Core output xml file> <Xlink file> <Step output xml file>
```

As jar file from UNIX:

```
> java -jar StepToCore.jar <Step input file> <Core output xml file> <Xlink file> <Step output xml file>
```

Step input file – name of file with Step program you want to translate

Core output xml file – the output of the translation will be written there

Xlink file – the “add info” (information about non-trivial translations) will be written there

Step output xml file – Xml version of Step program will be written there

Xlink file:

Used to log the information about non-trivial conversions with following titles:

1. Module Variable to global – variable declaration moved to global space
2. Module Type to global – type declaration moved to global space
3. Augmentation – translation of augmentation
4. Restriction – translation of restriction
5. Hiding – translation of hiding
6. Transition Hiding – transition was removed from export list
7. Variable Hiding – variable translated to local
8. Module Long Composition – Long composition (more than 2 modules) was translated to usual composition
9. Module Composition – internal composition (e.g. Hide(allb)) was moving to stand-alone module (d=allc)

The information about the structure of XLINK file with additional information you can find at \info\add_info.ps

Comments:

- The Rational and real types can't be simulated in Core, so we don't translate them
- Step SPL program will be translated first by Step parser to Module Transition System program and our translator will translate it and not the original program