
edaLink

Transfer of Schematics and Layouts

1	edaLink.....	7
1.1	Overview.....	7
1.2	System Architecture.....	8
2	Rules.....	10
2.1	Overview.....	10
2.2	Examples.....	10
2.3	Governing Script.....	11
3	Examples.....	12
4	Translation Options	14
4.1	Translation.....	14
4.2	Take-as-is.....	14
4.3	Full Translation Guarantee	15
5	Questions and Answers	16
6	Conventions.....	17

Version: 2.8	05.03.2007	edaLink
Language: English	Rau	Product Description
Content		P: 3/18

Figures

Figure 1: Overview	7
Figure 2: edaLink Translation	8
Figure 3: Sample of Rules File for SCH-Import	10
Figure 4: Sample of Rules File for PCB-Import	11
Figure 5: Example of Source Schematic	12
Figure 6: Schematic in Target System after edaLink Translation	13

1 edaLink

1.1 Overview

edaLink is a program system to translate electronic design data from a source to a target EDA workstation.

Translation includes the options

- logic libraries and physical description
- logic schematics
- layout libraries
- board layout.

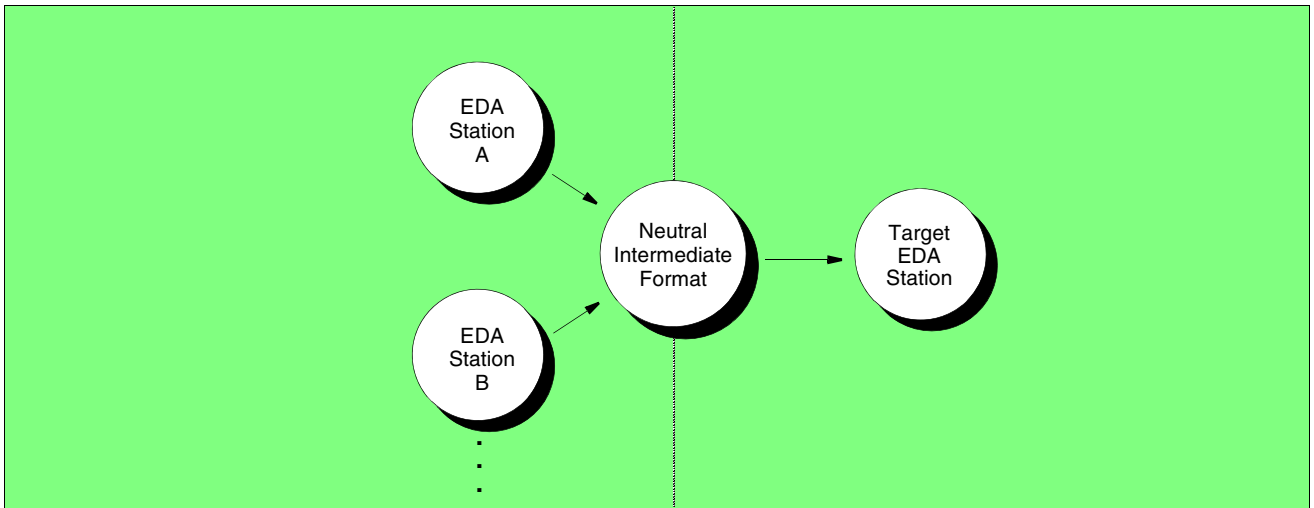


Figure 1: Overview

edaLink products are based on the EDA Vendor's access routines to the design databases which are included in the standard suite of the application. **edaLink** products maintain complete compatibility with documented interfaces of the EDA workstation Vendor.

Translated schematics, layouts, and libraries are visually the same and ready for use in the target environment.

Applications:

- transfer of libraries between EDA stations
- transfer of schematics between EDA stations
- transfer of PCB layouts between EDA stations
- schematic entry on PC, further processing of the design on the workstation
- archiving of designs in condensed Edif 200 format.

1.2 System Architecture

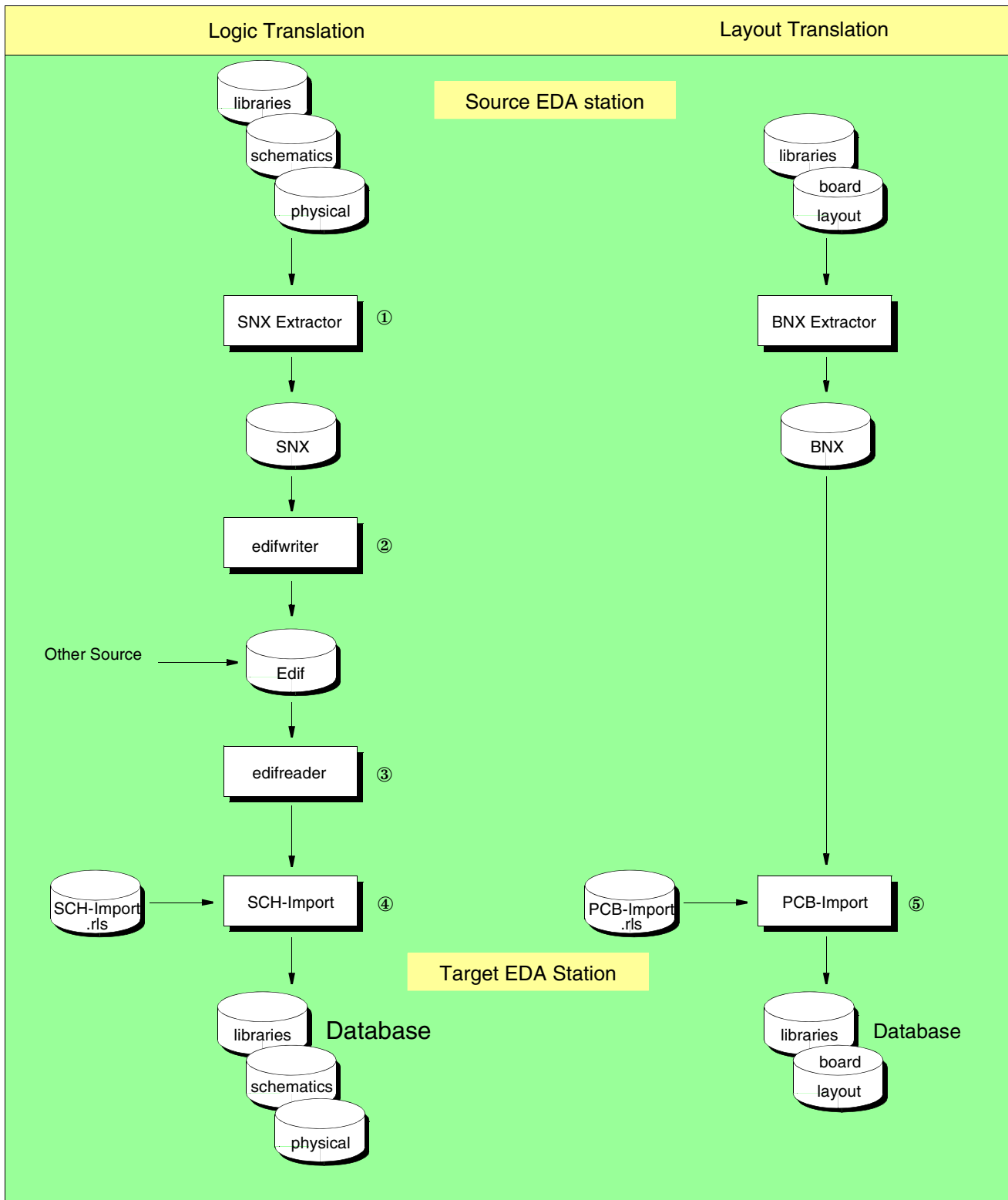


Figure 2: **edaLink** Translation

1. and 5. SNX and BNX Extractor

This software program is based on the source workstation's design database extraction utilities. The Neutral Exchange Format (NX) is basically an ASCII dump of the source database

which includes all logic design data of the source. Wherever possible, SNX and BNX extraction routines directly read the source EDA station's design database.

2. edifwriter

The program writes an Edif file with library data, with schematic data and with physical component data. The file meets the standard Edif 2.0.0, level 0, keyword level 0.

3. edifreader

The program reads any Edif file with graphic and connectivity view. The file may be generated by edifwriter or by another approved Edif source.

4. SCH-Import

The import utility for logic designs builds the complete design environment for the translated libraries and schematics in the target system.

SNX extractor and edifwriter maintain design data as they were extracted. SCH-Import performs the mapping of design objects of the source EDA station into design objects in the target schematics. This requires conscious decisions which may vary with the customer's design methodology. For this purpose, the rules file technology has been provided which controls the import process. The SCH-import utility is available for all renown EDA systems.

5. PCB-Import

The import utility for layout libraries and board layout data builds the complete padstack and package libraries, optionally synchronized with logic component descriptions and a pcb description. PCB-Import is rules controlled to map source design objects into target design objects corresponding to the customer's design methodology. pcb-import is available for all renown EDA systems.

2 Rules

2.1 Overview

Workstation constructs differ largely. For instance in logic designs, busses, connectors, and bus taps are all handled differently between workstations. Properties have different names and are placed in a different context.

2.2 Examples

The rules file *SCH-Import.rls* is hand-edited and includes conversion rules which control the translation process and design object mapping.

```
/* sch-import.rls V1.0 for import V1.0 */
{SCALE          /* Scaling of all geometrical objects */
.
.
.
{CENTER_PAGE On|Off} /* Center page coordinates; default off */

{CHAR_MAP      /* Mapping of special characters in all names */
}
{NET           /* Applies to nets, operates on net names */
  {'*$', '^-' } /* substitutes * at end with - at start of word */
  {'~', '*$' }  /* substitutes ~ anywhere with * at end */
  {'~$', '*$' } /* substitutes ~ at end with * at end */
  {'\'', '\_'}  /* substitutes ' with _ */
}
{PIN           /* Applies to pins, operates on pin names */
  {'\'', '\_'}  /* substitutions for NET also apply to PIN */
}
{BODY         /* Attached to symbol, operates on symbol names */
  {'\'', '\_'}  /* substitutions for NET; also apply to BODY */
}
}

{CONCATENATE  /* Concatenation of single nets into a bus */
  {NET_CONC   /* Operates on net names */
    {','}     /* Concatenation character is comma */
  }
  /* If no rule -> concatenation of nets not allowed */
  /* in source system */
  {PIN_CONC   /* Concatenation of pin names */
    {'@'}     /* Concatenation character is ampersand */
  }
.
.
.
```

Figure 3: Sample of Rules File for **SCH-Import**

The file is structured similarly to the sch-import rules file but includes rules for the mapping of layout objects.

```
/* PCB-IMPORT */

/* #include "other.rls" */

/*****
{ PCB
#include "scale.rls"
  { GRID 1.0 1 }          /* grid, accuracy */
}
*****/
/*****
/*****
{ MODIFY
  { LAYER_COPY
    { PADST "PAD_all_electrical" "POWER" }
    { PADST "PAD_comp"           "SOLDER_MASK_1" }
    { PADST "PAD_Solder"         "SOLDER_MASK_2" }
    { LAYSYM 20000 "Component"   "SILKSCREEN" }
    { LAYSYM 20000 "Placement"   "SILKSCREEN" }
  }
  { INST_ADD
#include "inst_add.rls"
  }
}
*****/
{ LAYER
  { NAME          /* Mapping of the layer name */
    /* DRAWING */
    { "^Assem_(DIP)$"      "DRAWING" }
    { "^Assem_(SIP)$"     "DRAWING" }
    { "^Assem_(SMD)$"     "DRAWING" }
    .
    .
    .
  }
}
```

Figure 4: Sample of Rules File for **PCB-Import**

2.3 Governing Script

Translation is governed by a procedure which executes the translation programs. The procedure is adapted to requirements of the customer's workstation system.

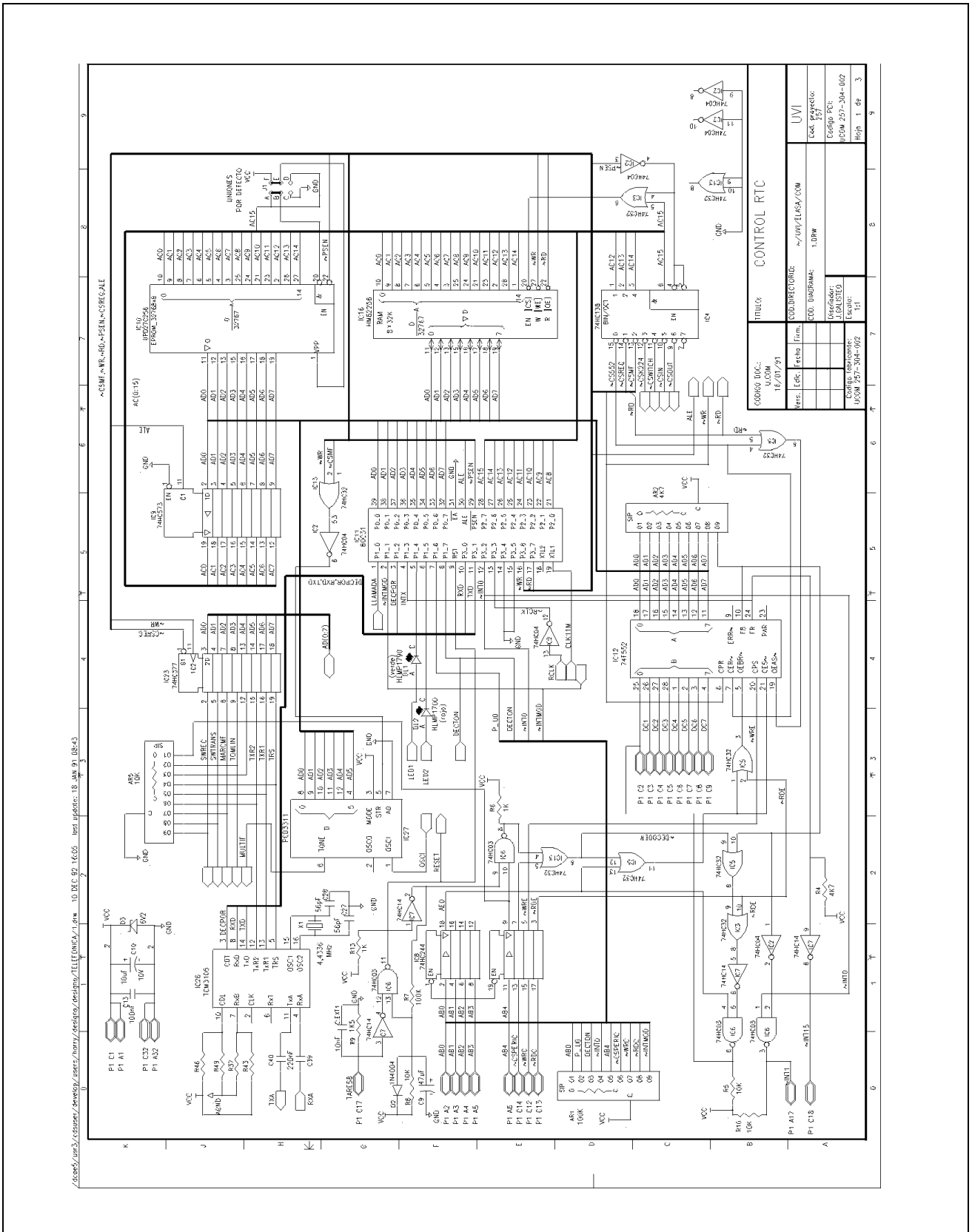


Figure 6: Schematic in Target System after edaLink Translation

4 Translation Options

4.1 Translation

If you prefer to have designs transferred once, you may want to use our transfer service:

- Transfer is calculated based on complexity and details of design.
- The assessment of the transfer effort is based on source workstation design data on magnetic media, preferably with plots.
- Quality control including netlist comparison source - target is part of the translation service.
- Our shipment includes the complete design directory with all control files and plots of the design.

Or you prefer to perform the translation yourself. **edaLink** licenses are issued per site and come in two variants:

- take-as-is
- full translation guarantee.

4.2 Take-as-is

Mature **edaLink** products have been used in at least three different customer environments. It is likely that most of your requirements will be covered by the existing product.

Pre-requisites for the transfer of logic libraries and logic schematics are:

- schematic database of a sophisticated design
- plots of schematics
- library database with physical description.

Pre-requisites for the transfer of layout libraries and pcb layout are:

- layout library database
- pcb layout database
- layer function document
- plots of layers.

The product is shipped within four weeks and includes your translated libraries and design in the *tutorial* directory of the release.

The release includes: translation program on tape with script for “push-button” operation, user’s guide with tutorial, installation instruction, and release note.

4.3 Full Translation Guarantee

The design methodology varies with every customer site. cae consulting offers a special adaptation to customer requirements and a half year's guarantee to meet the complete scope of translation requirements as specified.

Pre-requisites for the transfer of logic libraries and schematics:

- complete libraries with physical description
- at least five schematic databases covering the scope of the customer's usage of design objects
- corresponding plots

Pre-requisites for the transfer of layout libraries and pcb layout data are:

- complete layout libraries
- at least five board layout databases [which fit to the schematic databases]
- layer mapping document
- preferred: plots of all layers of the layout databases
- films and assembly plots for all layouts.

Steps of the translation process:

- Analysis and specification of translation. Approval of customer.
- Shipment and installation of program prototype in customer site.
- Quality assurance covering complete functionality and physical netlist comparison Source-Expert.
- [Specification and synchronization of logic translation with layout translation if the source was synchronized.]
- Option: Advance shipment of translated libraries.
- Shipment of **edaLink** logic translator within eight weeks, shipment of **edaLink** layout translator within ten weeks. Shipment includes all translated libraries and designs, translator programs on tape with tutorial, user's guide and installation instruction, installation in customer site and training of customer's expert.

5 Questions and Answers

1 - Can I use the design without rework in the target system?

You can use the design immediately. Rework is generally not required.

2 - How long does the translation of a well-filled A3 schematic page take?

Translation takes between 1 and 5 min depending on database access routines. The process requires 5 Mbytes of swap space per A3 page.

3 - How long does the translation of a multi-layer Double-Eurocard PCB take?

Typically 15 minutes.

4 - Can I use the same sequence of edaLink modules to translate from various EDA environments into one target system?

Yes - the only difference is the extraction routine.

5 - Can I translate libraries of schematics and layouts separately?

Yes - note that translated libraries need to fit to translated schematics and board layouts and therefore should not be modified.

6 - Do logic library descriptions include physical assignments like part number, package name, pin swap, pin sets, power pins?

Yes - all objects of the source are represented.

6 Conventions

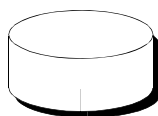
- {A|B} - Alternative A. B default.
<place_holder> - Replaced by actual value during execution.
[option] - Optional argument.



- Software process



- Directory



- File

- program** - Bold: names of executable programs.
/directory - Directory
design.rls - Italics: names of files and paths.
List - Courier: List as screen display.
% - System prompt in Unix.
- System manager prompt in Unix.
* - Wildcard function (any characters).
→ - Select with mouse click.
/1/ - Reference
./<directory> - The directory resides in the location of the actual execution.

