# Description

Generic High Energy Physics Parts Library (GHEPlib) is a CAD library to manage data across multiple electronic CAD tools.

# Dependencies

* Symbol library
  + None
* Management tools
  + SOCI library
  + daniele77/cli

# Used On

* Caribou v2.0 project
* Internal (confidential) projects at contributor sites

# Primary Developers

* Brookhaven National Laboratory
* Carleton University Particle Physics Instrumentation Group

# IP Control

* See License File
* Data can be imported from manufacturers into this library only if that data is accompanied by a waiver of ownership (stated as “free to use for any purpose including sale and open source”)
* Imported data without modification is kept in separate import libraries. If data is modified or customized, it should first be copied to the custom libraries. The indicated author should also change.

# Graphical Conventions

* Imperial dimensions – no metric (yet)
* 0.100" pin-pin and primary grid spacing
* 2 pin component dimensions
  + Siemens < 200mil wide, 400mil pin to pin (per conventions of existing parts)
  + Altium <200mil wide, 300mil pin to pin (per conventions of existing parts)

# Device Property Conventions

* Unique Resistance, Capacitance, Inductance, … properties
  + Siemens tools typically prefer a single VALUE property that is displayed. This is also more convenient for BOM export. However, this lacks clarity since the exact property being referred to may need to be assumed (e.g. diode forward vs. reverse voltages)
* Siemens
  + Convention for DEVICE = “Symbol Name”
    - Formerly, DEVICE = “Symbol Name”\_PN. As of 2023-03-08, the PART\_NUMBER property was added to enable the use of different symbols for the same part within the same design. This property name is also more explicit and clear and CAD tool independent
    - DEVICE must be unique since the tool uses this property for layout-schematic linkage
  + PART\_NUMBER= “PART\_NUMBER” before completion
  + Property names target the Netlist PADS/DxDesigner flow. The PADS documentation indicates that the cases for the properties vary between the Netlist and Intgrated design flows. E.g. [Part Number (PART\_NUMBER)](../../MentorGraphics/PADSVX.2.10/docs/htmldocs/attr/topics/General_PartNumberPartNumber_idee3e0fd4.html" \l "idee3e0fd4-f115-4b2d-a24a-fb65ff02efa8__General_PartNumberPartNumber_idee3e0fd4.xml%23idee3e0fd4-f115-4b2d-a24a-fb65ff02efa8)
  + HETERO property is fully completed if applicable
  + PKG\_TYPE=”PKG\_TYPE” (unlinked – determined at PCB layout start)
    - PCB only parts do not have this property
  + Only properties necessary for display in the symbol and basic packaging are added to the symbol
    - This is faster to enter and allows direct addition of other parameters at the schematic level as preferred by designers, assuming that mapping to database fields are managed correctly
    - This is typically, DEVICE, PKG\_TYPE, PART\_NUMBER, REFDES
    - Note: For some reason, properties can only be moved if they don’t contain the default value

# Verification Flows

* PADS example project holds all newly developed symbols
  + Visual review
  + Tools→ PCB Interface… (Packaging)
  + Tools → Diagnostics
    - Shows if schematic library is out of date
  + PADS Databook → New hierarchical Verification Window
    - Shows if library items can’t be found
* Tested on the following CAD tools
  + Altium 23
  + PADS VX.2.10

# Debugging Tips

## Siemens

* Symbols can “break” and exhibit strange behavior when instantiated. This may occur if the text of the symbols is directly edited.
  + One observed symptom of this is that the tool may automatically (per an ambiguous / unknown rule set) change the case of characters in symbol names in schematics however, references to symbol filenames with different case characters still work
  + Another observed symptom of this was the inability to delete or add specific properties, especially ones that are not present in the properties list.
  + The solution to this is to copy and paste the symbol graphics to a new symbol – PADS will generate a new file and therefore clean errors
* All symbols and symbol instances in a HETERO part must have exactly the same properties, otherwise packaging fails
  + The visibility of the property can vary between symbols
* After Tools→ Update Libraries, pins may become corrupted and fail to package. To resolve this, the part must be full replaced: right click→ Replace Symbol→ Search for same symbol name→ Replace

# Table Descriptions

Primary keys in each table are indicated with “PK”

* CAD\_table
  + Relates parts to the **preferred** CAD files. A single part can have many CAD models or data files but only a few are preferred and/or tested.
  + MFG = Manufacturer
  + PN = Part Number. This is the primary manufacturer part number (not distributor, not internal)
  + All symbols, footprints, and simulation models are references to CAD files in the Data\_table
* Data\_table
  + path: This can be a path to a file or a section of a file.
    - If this refers to sections of a file, identifiers relevant to each CAD tool should be used.
    - Symbols recognized by the CAD tool can be used. This is useful for default libraries.
  + author: The most recent editor of the data. As soon as the data or a file is modified, the author is considered to have changed. This is for liability tracing
  + release\_version:
    - If a default library is used, the exact release version should be indicated.
    - If a file was directly downloaded from a manufacturer, the date of the download should be indicated.
  + sym\_group:
    - Some parts are best represented by multiple separate symbols. This indicates that they should be grouped together.
  + deployment\_history: A short description of the harshest environmental deployment that the design within the file has survived. Examples include “not deployed”, “functional”, “functional after shock and vibration testing…”, “functional after high temperature testing…”

# TODO

* Table generators
  + database to Altium database with dblink file
  + database to DxDatabook database
* Cleanup functions
  + CAD entry deduplicator with reference updates
* Import functions
  + Digikey BOM
  + Mouser BOM
  + General BOM