

# McCAD v1.0

## User Manual

Moataz Harb  
moataz.harb@kit.edu  
Karlsruhe Institute of Technology (KIT), Hermann-von-Helmholtz-Platz 1,  
76344 Eggenstein-Leopoldshafen, Germany

May 18, 2022

# Contents

0.1	Introduction . . . . .	2
0.2	Installation from Source . . . . .	2
	0.2.1 Linux . . . . .	2
	0.2.2 Windows . . . . .	3
0.3	I/O . . . . .	4
	0.3.1 Decomposition . . . . .	4
	0.3.2 Conversion . . . . .	4
0.4	Notes on Usage . . . . .	4
	0.4.1 Decomposition . . . . .	5
	0.4.2 Conversion . . . . .	5
0.5	Known Issues . . . . .	5
	0.5.1 Decomposition . . . . .	5
	0.5.2 Conversion . . . . .	5
0.6	Theory of McCAD Conversion . . . . .	5
	0.6.1 Decomposition . . . . .	5
	0.6.2 Conversion . . . . .	5

## 0.1 Introduction

McCAD is an interface library for the conversion of CAD solid models to MCNP input syntax, from Boundary Representation "BREP" to Constructive Solid Geometry "CSG".

## 0.2 Installation from Source

McCAD library is supported on both Linux and Windows operating systems. The library has three 3rd-party dependencies: CMake, Boost C++ libraries, and Open CASCADE Technology. CMake is the standard build system for McCAD and it comes by default with most Linux distributions as well as in Windows OS. Boost C++ libraries consists of header files that are used for parallel processing in McCAD. Open CASCADE Technology (OCT) is used as a geometry engine for geometry solids manipulation and decomposition. Below are guiding steps for installation on both Linux and Win systems.

### 0.2.1 Linux

Listed below are the currently supported Linux distributions:

- Ubuntu 20.04 LTS
- Ubuntu 18.04 LTS

Testing of installation on other distributions is still underway! thus Ubuntu is recommended as a distribution to build McCAD library. Below are general steps to build McCAD library and its dependencies.

- **CMake**
  - Download cmake-3.23.0.tar.gz from <https://cmake.org/download/> then execute the commands below in a terminal.
  - `$ tar -xzf cmake-3.23.0.tar.gz`
  - `$ cd cmake-3.23.0`
  - `$ mkdir build`
  - `$ cd build`
  - `$ cmake .. -DCMAKE_USE_OPENSSL=OFF -DCMAKE_INSTALL_PREFIX=.`
  - `$ make`
  - `$ make install`
- **Boost C++ libraries**
  - Download boost\_1\_78\_0.tar.gz from <https://www.boost.org/users/download/> then execute the commands below in a terminal.
  - `$ tar -xvf boost_1_78_0.tar.gz`
  - `$ cd boost_1_78_0`
  - `$ mkdir build`
  - `$ cd tools/build`
  - `$ ./bootstrap.sh`

- \$ ./b2 install --prefix=.././build/

- **Open CASCADE Technology (OCCT)**

- *NOTE*: the instructions on the installation of dependencies can be found in the side menu by navigating to "Build, Debug and Upgrade > Build 3rd-parties" then following the instructions under "Installation from Official Repositories" in <https://dev.opencascade.org/doc/occt-7.5.0/overview/html/index.html>
- Download opencascade-7.5.0.tgz from <https://dev.opencascade.org/release/previous> then execute the commands below in a terminal.
- \$ tar -xzf opencascade-7.5.0.tgz
- \$ cd opencascade-7.5.0
- \$ mkdir build
- \$ cd build
- \$ cmake .. -DCMAKE\_BUILD\_TYPE=Release -DBUILD\_LIBRARY\_TYPE=Shared -DCMAKE\_INSTALL\_PREFIX=. -DINSTALL\_TEST\_CASES=TRUE -DINSTALL\_DOC\_Overview=TRUE
- \$ make
- \$ make install

- **McCAD**

- *NOTE*: building a shared library is recommended! Should a static library be needed, the user has to insure a compliant build of Open CASCADE Technology by changing the build type; -DBUILD\_LIBRARY\_TYPE=STATIC.
- \$ git clone <https://github.com/inr-kit/McCAD>
- \$ cd McCAD
- \$ mkdir build
- \$ cd build
- \$ CMake .. -DCMAKE\_INSTALL\_PREFIX=. -DBUILD\_STATIC=OFF -DBOOST\_CUSTOM\_ROOT=<PATH to boost.1.78.0> -DOCC\_CUSTOM\_ROOT=<PATH to opencascade-7.5.0/build> -DBUILD\_RPATH=ON
- \$ make
- \$ make install

## 0.2.2 Windows

Listed below are the currently supported Windows versions:

- Windows 10

Testing of installation on other versions is still underway! Below are general steps to build McCAD library and its dependencies.

- **CMake** (optional)

- *NOTE*: If usage of IDE - such Microsoft Visual Studio (VS) - is intended, then installing CMake can be skipped since most IDE builds CMake by default.
- Download and run the installer cmake-3.23.1-windows-x86\_64.msi from <https://cmake.org/download/>.
- **Microsoft Visual Studio** (optional)
  - Download and run the "community" installer from <https://visualstudio.microsoft.com/downloads/>.
- **Boost C++ libraries**
  - Download boost\_1\_78\_0.zip from <https://www.boost.org/users/download/>.
  - Unzip boost\_1\_78\_0.zip.
  - Documentation can be found in index.html in the unzipped folder.
- **Open CASCADE Technology (OCCT)**
  - Download and run the installer opencascade-7.5.0-vc14-64.exe from <https://dev.opencascade.org/release/previous>.
- **McCAD**
  - Download source code from <https://github.com/inr-kit/McCAD> by selecting Code > Download ZIP.
  - Unzip McCAD.
  - Open MSVC and select the folder containing McCAD source code.
  - From the "Solution Explorer - Folder Review" double click CMakeSettings.json file. This will open the file in IDE.
  - Set a "Configuration name".
  - Ensure that "Configuration type" is set to "Release" and "Toolset" is set to msvc\_x64\_x64.
  - On "CMake command arguments" set -DBOOST\_CUSTOM\_ROOT="<PATH to boost\_1\_78\_0>"
  - On "CMake command arguments" set -DOCC\_CUSTOM\_ROOT="<PATH to OpenCASCADE-7.5.0-vc14-64\opencascade-7.5.0>"
  - From the top menu select "Build > Build All".
  - From the top menu select "Build > Install McCAD".

## 0.3 I/O

Input and output file formats. Coming soon ...

### 0.3.1 Decomposition

### 0.3.2 Conversion

## 0.4 Notes on Usage

General notes on how to use McCAD. Coming soon ...

#### **0.4.1 Decomposition**

#### **0.4.2 Conversion**

### **0.5 Known Issues**

A list of known issues and how to manually fix it in solid models. Coming soon ...

#### **0.5.1 Decomposition**

#### **0.5.2 Conversion**

### **0.6 Theory of McCAD Conversion**

Comperhensive details on the inner workings of McCAD classes for developers. Coming soon ...

#### **0.6.1 Decomposition**

#### **0.6.2 Conversion**