KISMET: A short Step by Step Guide

Version 1.2

24.06.1997

Hans-Georg Krumm
Dr.-Ing. Uwe Kühnapfel

Forschungszentrum Karlsruhe GmbH
Institut für Angewandte Informatik
Postfach 3640
76021 Karlsruhe, Germany
E-mail: jet@iai.fzk.de

List of Contents

List of figures

List of tables

1 Introduction

This document is a short step by step guide for KISMET. It shows the novice user how to generate a simple kinematic structure by creating the "kinematic skeleton" and adding simple geometric parts.

Note: The KISMET model *empty.sim* and all the files described in chapter 3.2 must exist. In case this model does not exist, please send an email to:

jet@iai.fzk.de
Subject: empty.sim

Descriptions of menus or buttons to be pressed or selected in KISMET are printed like ***this***.

2 Project Directory Structure and KISMET file types

kis_home						
envlib	**mpclib**	**mpolib**	**teachlib**	**scripts**	**textures**	**mpdlib**
.sim .mat .col .lay .lgt .dof	.mpc .cal	.mpo	.ird .wpf	.spt .vws	.rgb	.def .stf .ita .csf

Table 1: Project Directory Structure

The UNIX shell variable *kis_home* defines the search path for the project directory. For the following examples it must point to that directory whose subdirectory *envlib* contains the file *empty.sim.* The 7 different directories listed in Table 1 contain the different KISMET file types:

- **Simulation Control File**
  - .sim Simulation Control File
- **Files describing layout parameter**
  - .lay Window layout and viewing parameters
  - .lgt Description of light sources
  - .mat Description of surface properties (materials)
  - .col Definition of colour tables
- **Files useful for Animation and Simulation**
  - .spt Script Files
  - .vws Predefined views
- **Files used for Modelling**
  - .mpo Definition of geometric shapes
  - .mpc Definition of kinematic structures
  - .dof Specification of degrees of freedom
  - .cal Calibration file used for monitoring (online interface online)
- **Robot Programming**
  - .wpf Workframe Path File
  - .ird IRDATA robot programs
- **Dynamics**
  - .def Basic information describing some dynamic properties
  - .stf Sitffness matrix
  - .ita Inertia matrix
  - .csf Control system description

3 How to create a kinematic structure

3.1 Introduction

Figure 1 shows an example of a 2 link mechanism. This chapter is about the basic steps to create such a model.

A kinematic structure or mechanism is regarded as being composed of rigid links connected by joints. KISMET provides so called frames to create the kinematic skeleton. Geometric parts are then added to the frames to complete the model.

3.2 Example

Starting with empty.sim the following chapters will enable the reader to build a 2 link mechanism with 2 rotational joints.

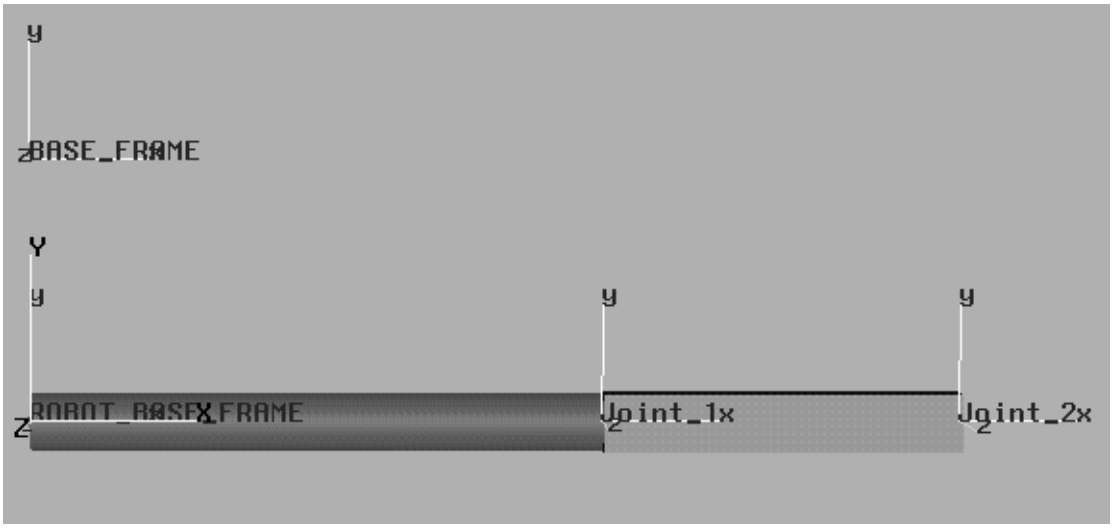

Figure 1: The 2 link mechanism example

Note: The following files must exist:

envlib	**mpclib**
empty.sim	empty_base.mpc
empty_color.col	empty_robot.mpc
empty_layout.lay	
empty_lights.lgt	
empty_material.mat	
empty.dof	

Table 2: The empty.sim files

3.3 Building a 2 link mechanism: An example

1. **Start KISMET and open the model empty.sim:**

You will see an empty screen with just 1 frame (Figure 2).

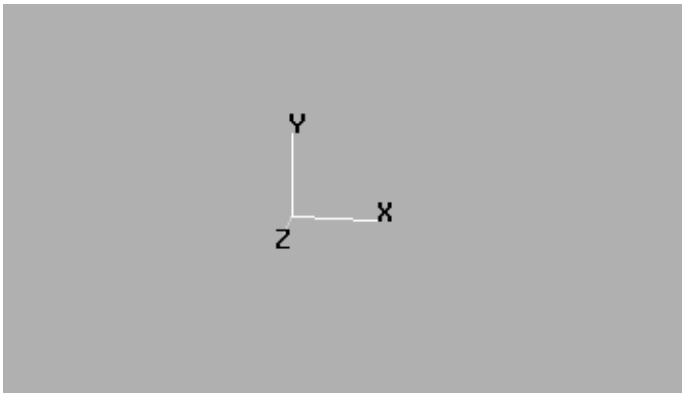

Figure 2: empty.sim

2. Switch to ***MODEL*** main menu and select the ***Edit FRAME*** menu. Press ***Show FRAMEs***. 2 new frames will appear: BASE_FRAME and ROBOT_BASE_FRAME (Figure 3).

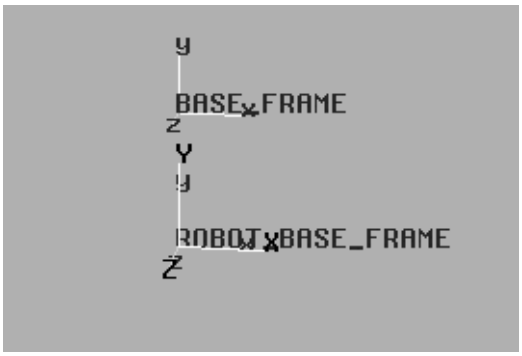

Figure 3: The frames of empty.sim

3. **Create a new frame:**

During the process of creating a new frame you have to determine 2 predecessor frames:

- The first predecessor frame determines the mpc file where the new frame should be added two.
- The second frame determines the kinematic predecessor.

In order to determine these 2 frames, you have to pick the frames by moving the cursor to the frame origin and pressing the left mouse button.

Use ROBOT_BASE_FRAME as predecessor frame in both cases:

- Use ***CREATE*** in the Edit Frame Panel
- The message *Pick tree predecessor* will appear. Pick the ROBOT_BASE_FRAME.
- The message *Pick kinematic predecessor* will appear: Use ROBOT_BASE_FRAME as kinematic predecessor.
- Enter the following values:

  Enter JOINT_NAME: Joint_1
  Enter JOINT_TYPE: 0 (rotational joint)

Enter JOINT_MINIMUM_LIMIT: -90.0
Enter JOINT_MAXIMUM_LIMIT: 120.0
Enter MAX_JOINT_VELOCITY: 10.0
Enter MAX_JOINT_ACCELERATION: 10.0

4. **Move Joint_1 to the position shown in Figure 1:**

– Use ***Pick FRAME*** in order to pick Joint_1

– Use ***Translate*** and move Joint_1 1200mm in x-direction (see Figure 4)

Figure 4: Joint_1 moved 1200mm

5. **Create a geometric part for link 1:**

– Switch to ***Edit GEO***

– Select ***CREATE***

– Select ***Cyl.*** from the Primitive Submenu. Enter the following values:

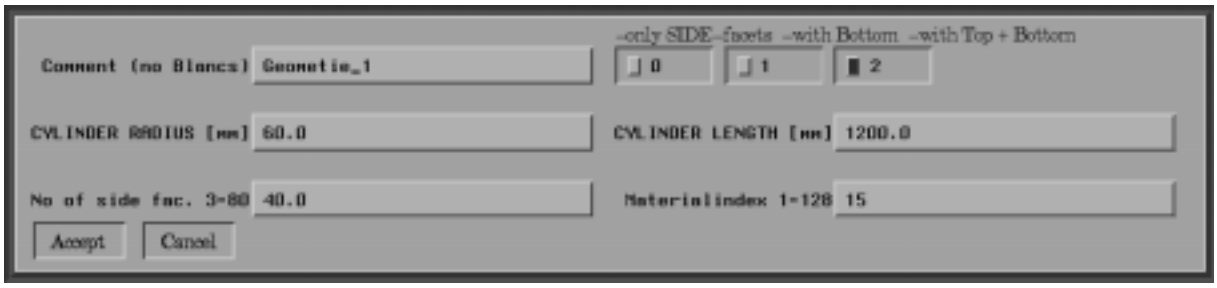

Figure 5: The Cylinder Primitive submenu

– Enter Geo_1.mpo as name.

– Connect Geo_1.mpo with frame Joint_1 frame.:

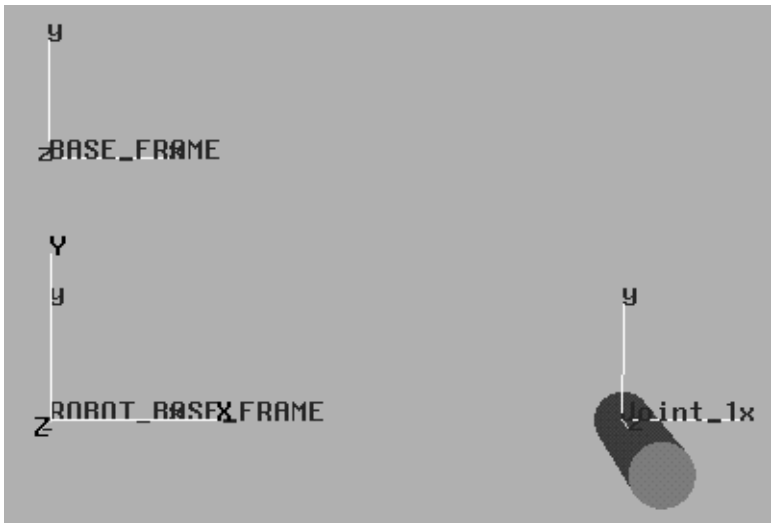

Figure 6: Connecting Geo_1.mpo with Joint_1

6. **Move Geo_1.mpo** to its final position (Figure 7)
   - Use ***PICK*** Geo_1.mpo and
   - use ***Rotate*** to rotate the GEO to it's final position
   - Use ***CONFIRM*** to confirm the final position of Geo_1.mpo (Figure 7)

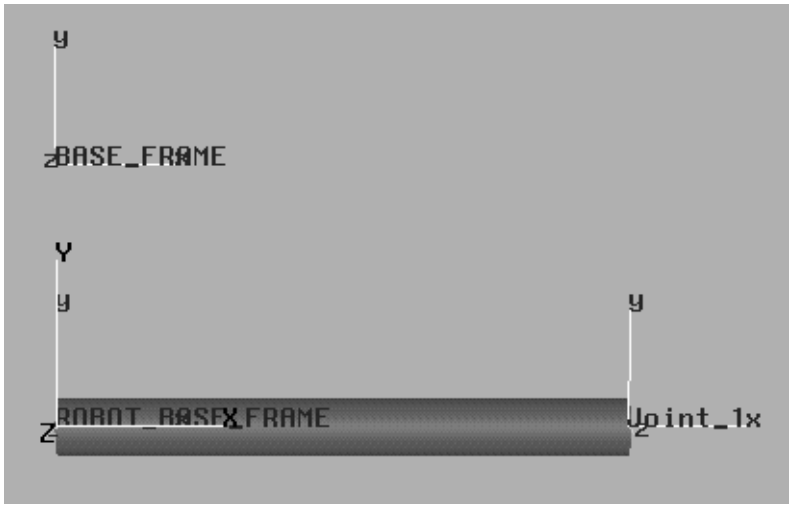

Figure 7: empty.sim with Geo_1.mpo

7. **Save the model created so far:**

- Switch to the ***DETAIL*** menu
- Select ***Create robot***:

  Create a new model by entering robot_1.sim as sim-file name. Now you have to change the name of the mpc-base file. The following window will appear:

Figure 8: The BASE mpc file menu

Change the name of the mpc file and update the dependent mpc file:

Figure 9: The altered BASE mpc file menu

A new window will appear. Change the values to:

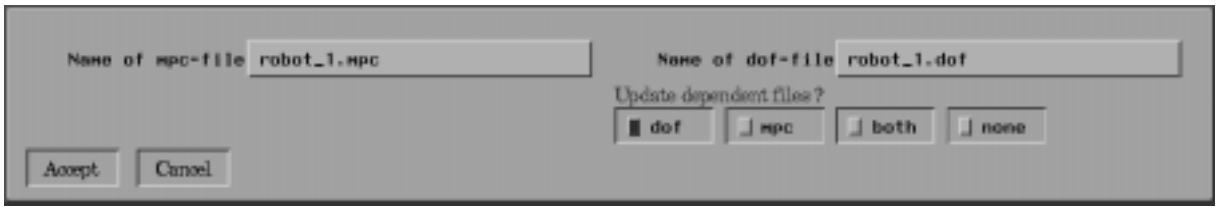

Figure 10: The mpc-file menu

Now you have created a new KISMET model called *robot_1.sim* with all its mpc and dof files. The next steps are:

8. **Load** your new KISMET model:

– Use ***New MODEL*** and select your KISMET model robot_1.sim

9. **Create a new frame** according to step 3 and use Joint_1 as kinematic predecessor. Enter Joint_2 as name-

10. **Move the new frame** 750 mm in x direction according to step 4.

11. **Create a geometric part** for link 2 according to step 4 and call it Geo_2.mpo. Use a different geometric primitive (e.g. the box primitive) this time and take account of the different link length (750mm instead of 1200mm). Enter the following values for the box primitive:

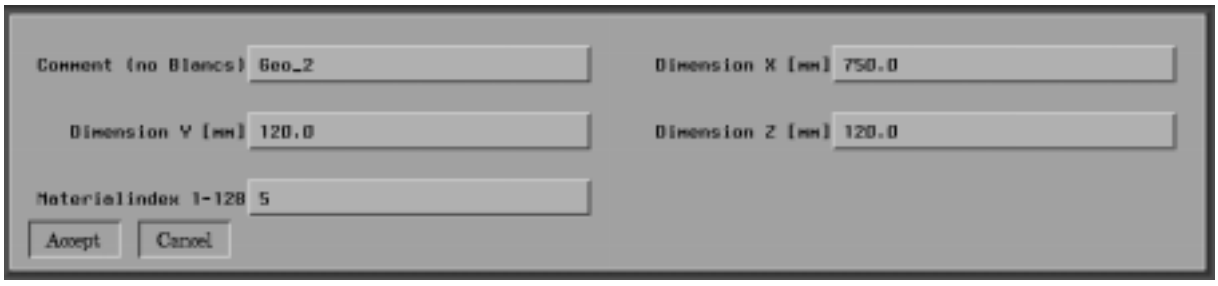

Figure 11: The Box primitive

12. **Move Geo_2.mpo** to it's final position.

Pick Geo_2.mpo and use ***Translate***. Try to make the result look like this:

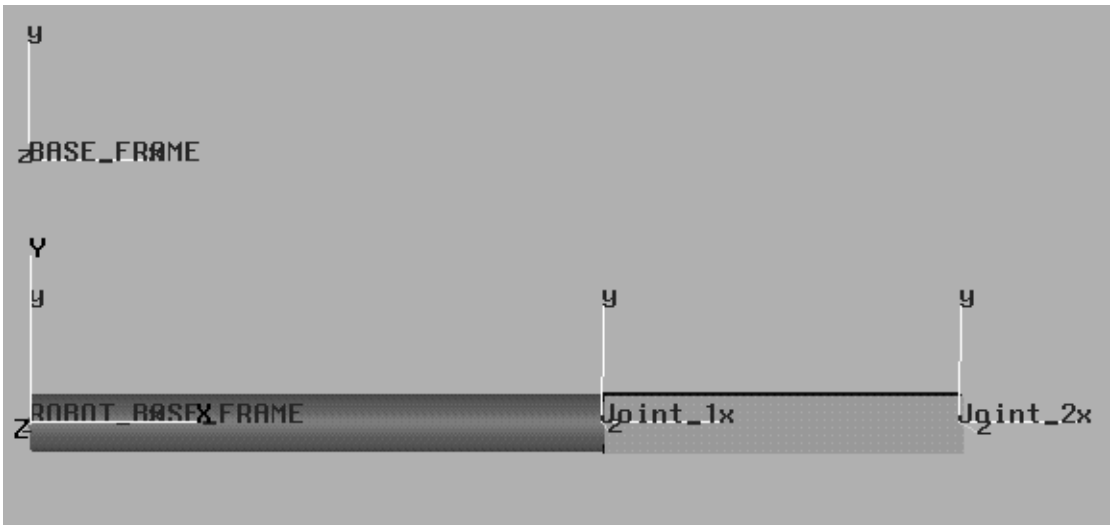

Figure 12: The complete 2 link mechanism

13. **Save the model:**
    Since you have only added a new frame and a new GEO to your robot_1.mpc file, you don't have to save the whole model but only the robot_1.mpc file:

    - Switch to ***DETAIL*** main menu
    - Use ***Pick GEO*** and select one of the 2 GEOs
    - Use ***Save as*** to save the mpc file. **Do not change any values.**
    - Overwrite the existing robot_1.mpc file.

4 How to move the kinematic structure

- Switch to ***MOTION*** Main menu
- Select ***Single DOF***
- Press ***ATTACH Dof*** and select Joint_1_DOF with the left mouse button
- Press ***ATTACH DOF*** and select Joint_2_DOF with the middle mouse button
- By moving the mouse with pressed button (left or middle) you will move one of the two joints.

5 A closer look at the new KISMET model

After creating a new KISMET model it's time to have a closer look to the file-, frame- and geometric structure:

- Switch to ***DISPLAY***
- Select the ***Logical*** Display mode:

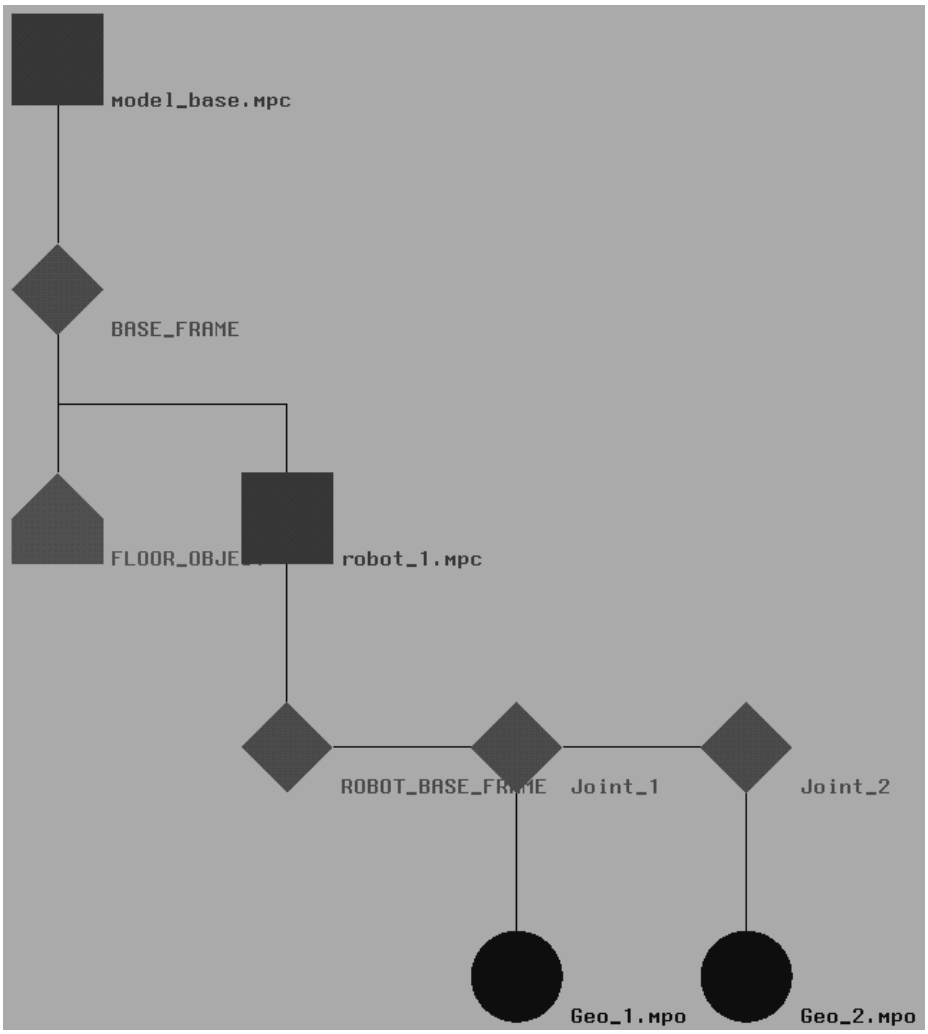

Figure 13: The Logical View

In the logical view you see 2 mpc file (model_base.mpc and robot_1.mpc) with it's frames and geometric parts.