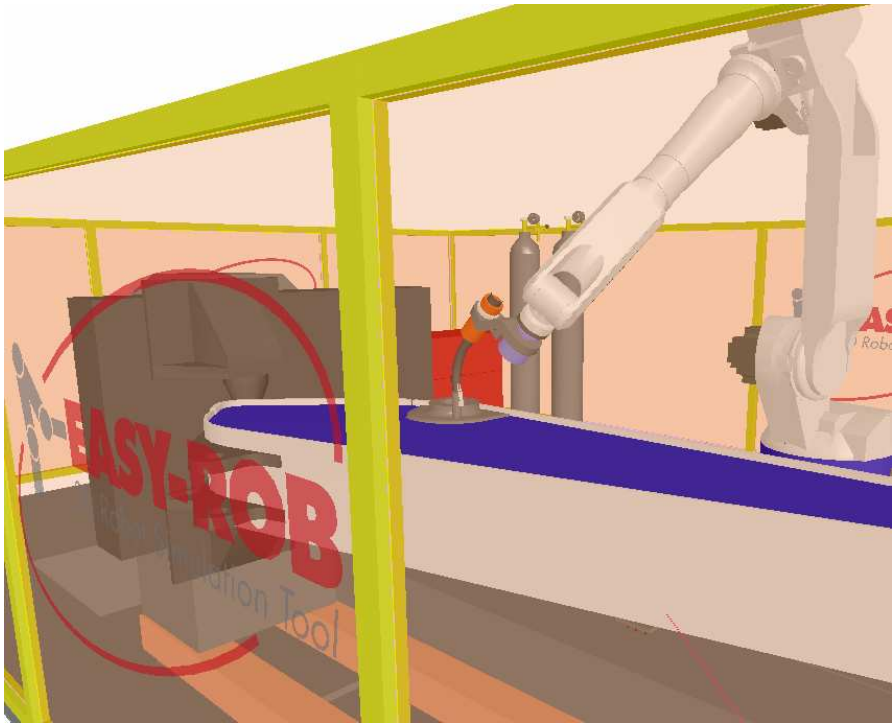
The new Version

EASY-ROB™ V5.3



January 2010

Version 1.03

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™

Table of contents

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™ V5.3

The new Version 5.3

We are glad to deliver the new EASY-ROB™ Version 5.3 with numerous new features and improvements. Below document will give you a first overview about most important changes. You will find a more detailed description in the Operation references

- **CAD Data-Import**
  Support the neutral format 3D-Studio 3DS File Format
- **Transparent Geometries**
  Geometries can be visualized transparent, using alpha blend
- **New Collision Detection Algorithm**
  A new collision detection algorithm was implemented, which allows to define tolerances
- **Replace Robots**
  Replacing robots and devices with one mouse click improves layout planning
- **SNAP-Function**
  Easy placement of robots, devices, geometries and tags on surfaces and vertices.
- **Simplified TOOL Definition**
  The definition of tool data is much easier now.
- **TAG Attributes**
  Additional tag attributes for better overview ERPL programs.
- **Motion Planning with Robot TURNS**
  The motion planner considers TURNS for PTP motions
- **Measurement Functions**
  Measuring of circles and distances between devices and circle center, as well as surfaces
- **Character Font Size**
  Adjustable font size for geometry names and tags for better readability
- **Extended API**
  The new exported class ER_CAPI structures API-Function and simplifies their usage
- **Windows® 7**
  EASY-ROB™ compatible on 32- and 64-Bit

The new version is available free of charge for all customers with a valid license key for EASY-ROB™ V5.3. For customers using older versions, it will be possible to purchase an update. We would like to thank you for your suggestions and ideas in advance.

Thank you

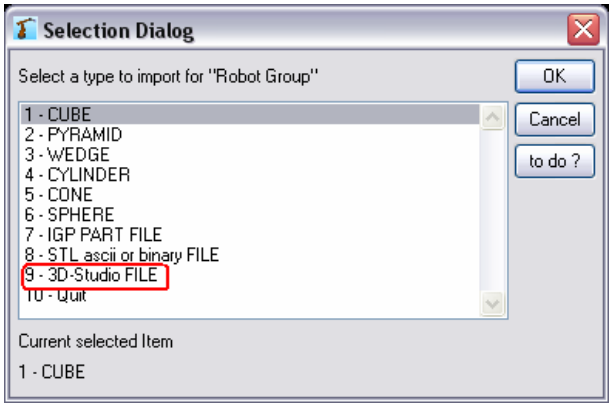
Stefan Anton
EASY-ROB
3D Robot Simulation Tool

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

3D-Studio CAD Data-Import Format

EASY-ROB™ supports the neutral 3D - Studio 3DS File Format, besides the CAD formats IGP Part File, STL (ASCII and binary). This format can be imported directly without conversion in advance. The 3DS format is exported by many CAD Systems and is characterized by its compactness. It can be imported very fast.



3D CAD Menu > Create/Import new 3D CAD Body

Other formats, such as the neutral formats STEP, IGES, VDA, VRML II and JT-Open, are converted using the CAD Data-Import Kernels (3D_Evolution® API). The current Import Kernel has increased performance, especially reducing the number of triangles.

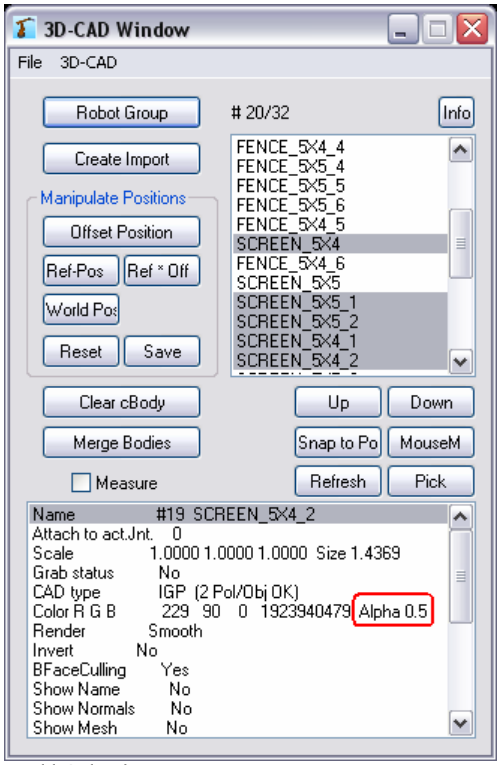
The CAD Data Import Kernel allows it also to read native formats, such as CATIA V4 und CATIA V5 R19, Pro/E, UG, SolidWorks, Robface and other formats.

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

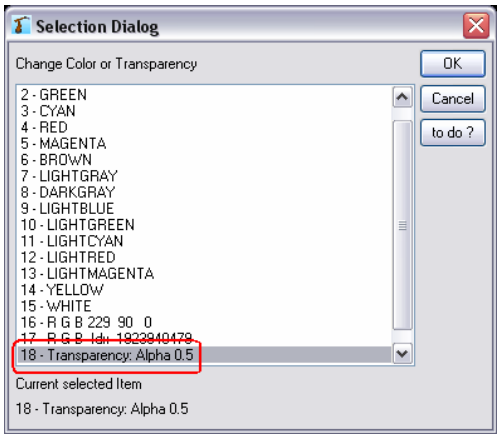
Transparent Geometries

All geometries can be shown in transparent mode using the alpha blend value

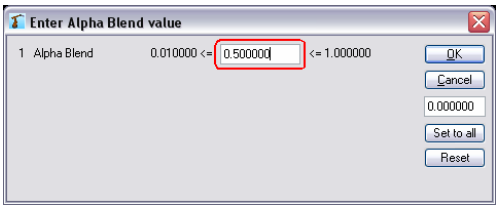
Open the 3D-CAD Window and select one or multiple bodies. Double click line „Color R G B“



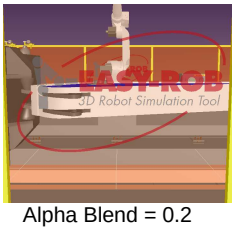
„Multi-Selection“

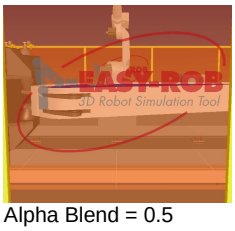


Select line „18 - Transparency: Alpha 0.5“ and enter a new alpha blend value in the range from 0 to 1.



Alpha Blend Value
= 0 invisible
]0,1[transparent visualization
= 1 transparency disabled


Alpha Blend = 0.2


Alpha Blend = 0.5

ERCL - Command

```
ERC TRANSPARENCY
      BODY [ROBOT,TOOL]
      bodyname alpha
ERC TRANSPARENCY
      BODY_GRP [ROBOT_GRP,TOOL_GRP]
      alpha

mit alpha = [0,1]
```


EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

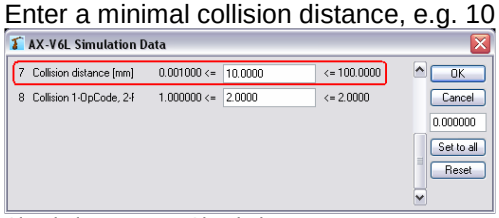
New Collision Algorithm

The new collision algorithm allows defining tolerances and minimal distances.

While simulating, the collision is shown if the distance between two geometries is less than the minimal allowed distance. This allows improving planning results considering tolerances in parts and other existing real deviations.

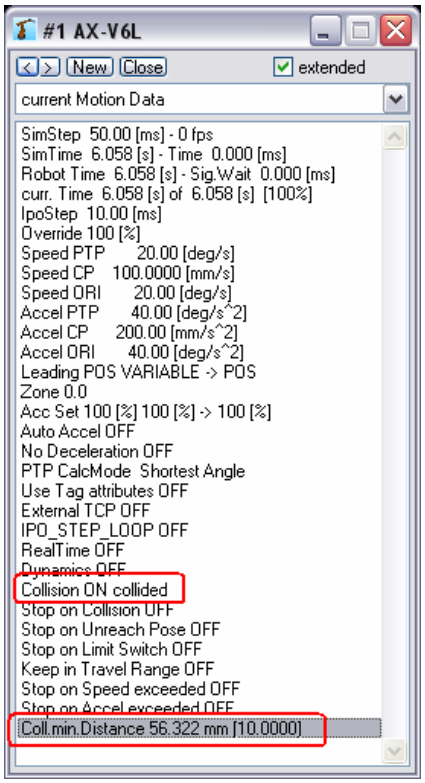
Enable collision 

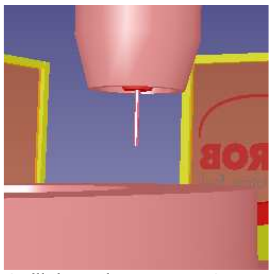
Enter a minimal collision distance, e.g. 10 mm



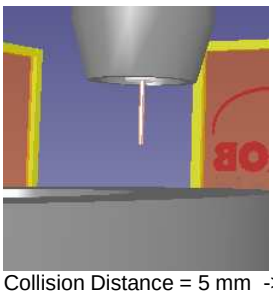
Simulation Menu > Simulation Data

Open Online Output Data 

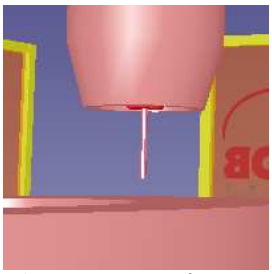




Collision Distance = 10 mm -> Collision



Collision Distance = 5 mm -> no Collision



TCP moves 5 mm down
Collision Distance = 5 mm -> Collision

ERCL - Command

```
ERC COLLISION ON/OFF
ERC COLLISION DISTANCE distance
mit distance in [mm]
```

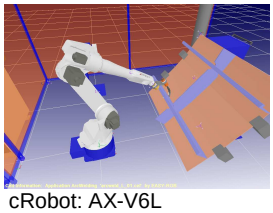

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

Replace Robots and Devices

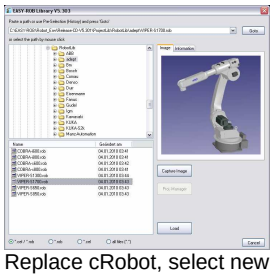
The replacements of robots and devices are significant improved. Robots in a work cell are normally connected in their environment and have a position w.r.t. a reference system. Furthermore they have tool data and an ERPL-Program.

When replacing robots, all these attributes are taken over to the new robot. Therefore it will be very easy now to replace a robot in case its reach ability is not sufficient. This new feature improves layout planning and saves time.

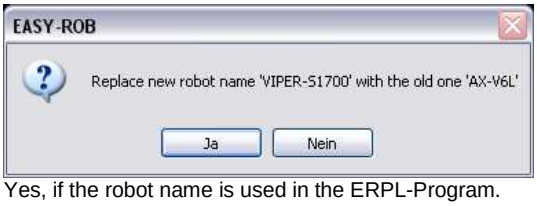
File Menu > Load > Replace cRobot



cRobot: AX-V6L

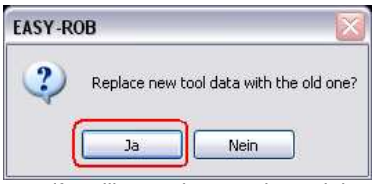


Replace cRobot, select new one



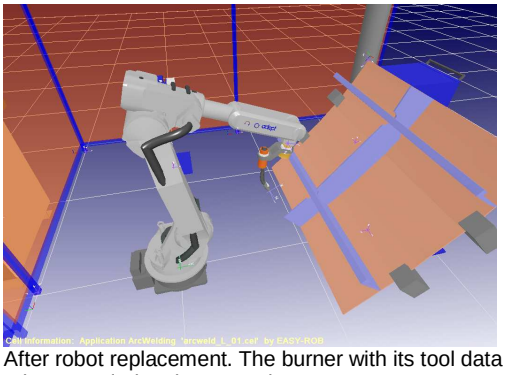
Yes, if the robot name is used in the ERPL-Program.
No, if there is no reference to the robots name.

In our example the command
„ERC CURRENT_DEVICE SET AX-V6L“
is used, thus we choose YES

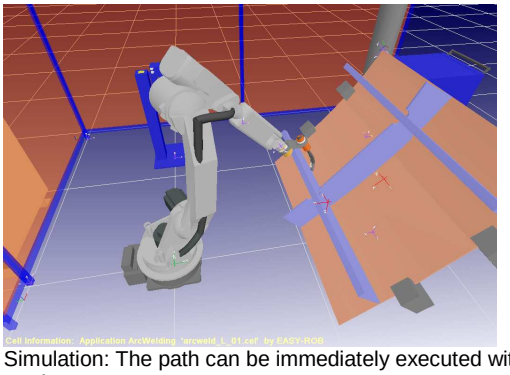


Yes, if we like to take over the tool data from the old robot. This makes sense when replacing robots.
No, make only sense in case we replace a tool, i.e. another burner.

In our example we replace a robot and we want take over the tool data from the old robot. We choose Yes.



After robot replacement. The burner with its tool data have been copied to the new robot



Simulation: The path can be immediately executed with the new robot.

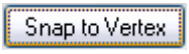
EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

SNAP-Function

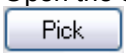
The new SNAP-Function allows locating robots, devices, tools, geometries and tags on surfaces and vertices.

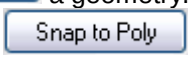
1. SNAP Devices
   - Open the Kinematics Window (Robotics Menu > Open Kinematics Window or Ctrl+K) and select with 

a device.
   - Select 

or 

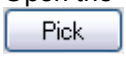
and click on a surface or vertex in your work cell.
   - The device base „jumps“ to the selected surface or to the selected vertex. Using Snap to Vertex, the orientation will be kept constant.

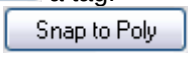
2. SNAP Bodies/Geometries
   - Open the 3D CAD Window (3d CAD Menu > Open 3D CAD Window) and select with 

a geometry.
   - Select 

or 

and click on a surface or vertex in your work cell.
   - The geometries reference system „jumps“ to the selected surface or to the selected vertex. Using Snap to Vertex, the orientation will be kept constant.

3. SNAP Tags
   - Open the Tag Window (Tags Menu > Open Tag Window) and select with 

a tag.
   - Select 

or 

and click on a surface or vertex in your work cell.
   - The tag „jumps“ to the selected surface or to the selected vertex. Using Snap to Vertex, the orientation will be kept constant.

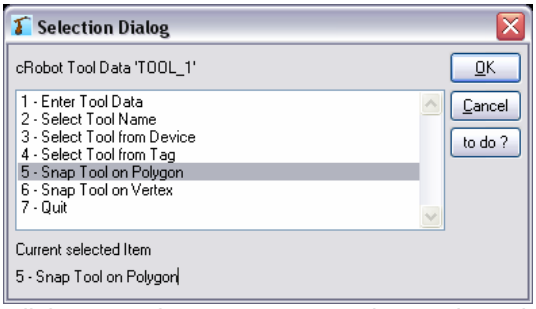
Hint: First place your object on a surface to get the surface normal and then snap on vertex afterwards.

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

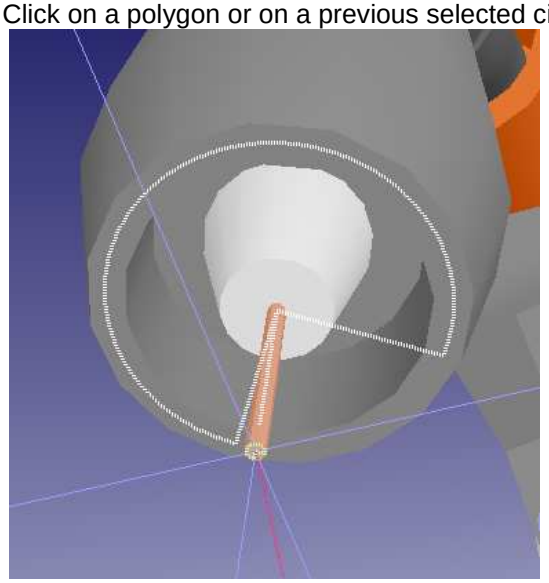
Simplified Tool Definition

When creating a tool, usually the tool geometry is loaded to the “Tool-Device” first and the TCP data are entered afterwards. Finally the tool is saved as “.rob” file with an appropriate image.

1. Tool definition at geometry
   - Zoom the geometry in a suitable range on your screen.
   - Choose from the Robotics Menu > cRobot Tool Data > 5 - Snap Tool on Polygon



   - Click on a polygon or on a previous selected circle segment (here it is white or green).



   The new TCP „jumps“ to the selected surface.
   Hint: First place your object on a surface to get the surface normal and then snap on vertex afterwards.

2. Robot TCP definition from another tool device
   - If a robot is loaded and a tool device is mounted at the robots flange, the tool data can be taken over. Choose from the Robotics Menu > cRobot Tool Data > “3 - Select Tool from Device“ and select the according tool device. Finally select „1 - Use cTool“.
   - The robot has taken over the Tool data from the tool device.

In a same way it is possible to take the tool data from a tag.

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

Additional TAG Attributes

Additional tag attributes allow storing information such as motion type, speeds, accelerations, motion planner behavior, turns, wait time and other relevant parameter on the path or at the target location.

The usage of tag attributes simplifies the ERPL program for the robot and results in more clearness. Furthermore tag attributes can be changed during the simulation to test and verify different ideas. The program keeps unchanged.

TAG Attributes:

- Motype	Motion Type: PTP, LIN, VIA, CIRC
- Speed PTP Ov [%]	Percentage Speed for PTP
- Speed CP [mm/s]	CP Speed for LIN, CIRC
- Speed ORI [deg/s]	Orientation Speed for LIN, CIRC
- Speed PTP Ax Ov [%]	Percentage Speed for PTP for each axis
- Accel PTP Ov [%]	Percentage Acceleration for PTP
- Accel CP [mm/s]	CP Acceleration for LIN, CIRC
- Accel ORI [deg/s]	Orientation Acceleration for LIN, CIRC
- Accel PTP Ax Ov [%]	Percentage Acceleration for PTP for each axis
- AutoAccel ON/OFF	Automatic calculation of acceleration depending on programmed speed
- AccSet Acc [%]	Reduced Acceleration & Deceleration [20% - 100%]
- AccSet Ramp [%]	Change of Acceleration and Deceleration [20% - 100%]
-	
- Configuration	Robot configuration, used only for PTP
- No_Decel ON/OFF	ON – no deceleration, speed at target equals programmed speed OFF – deceleration, speed at target equals 0 (default)
- Zone [mm,%]	Fly by parameter 0-fine point (default), >0 zone value
- Leading Position	0 - leading orientation, 1 - leading position, 2-variable (default)
- Lead Time [s]	Wait time before motion starts
- Lag Time [s]	Wait time when target is reached
- LIN Ori Mode	0-Var, 1-Fix, 2-Tang, 3-Aux, 4-Quaternion (default)
- CIRC Ori Mode	0-Var, 1-Fix, 2-Tang, 3-Aux, 4-Var2, 5-Quaternion (default)
- Turn Use ON/OFF	Use Turns if defined
- Turn_Ax	Turn-Values for each axis of the robot according defined Turn interval
- User String	User defined string for API

To consider TAG attributes in the program, “Use Tag Attributes“ must be enabled.

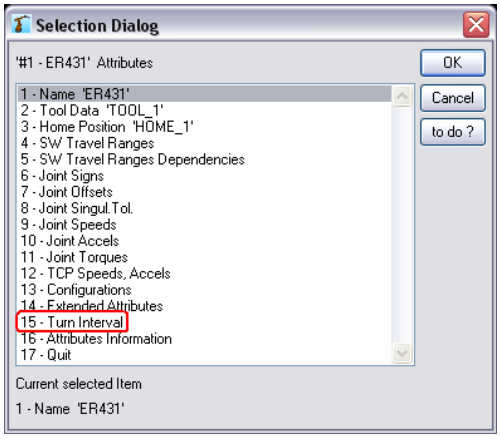
ERC USE_TAG_ATTRIBUTES ON/OFF

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

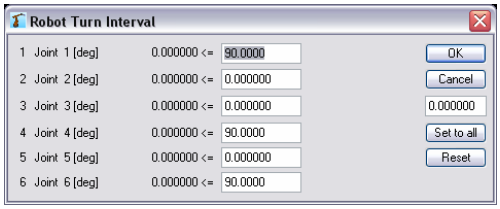
Motion Planner with Robot TURNS

The motion planner considers programmed TURN values for PTP motions. For rotational axis with a travel range greater than 360° or without limits, it is po ssible to specify the quadrant for the solution. This allows a preturning of joints. Important are the defined TURN intervals for the robot, i.e. for ABB robots 90° or for KUKA and Fanuc robots 180°.

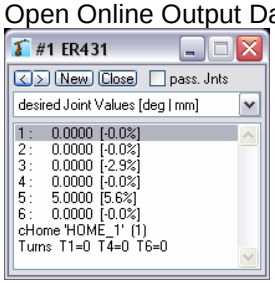
Load robot ER431 from the standard robot library. Open the Kinematics Window (Ctrl+K) and click on button „Attributes“, select „15 - Turn Interval“.



Define for axis 1,4 and 6 a Turn-Interval of 90° each.

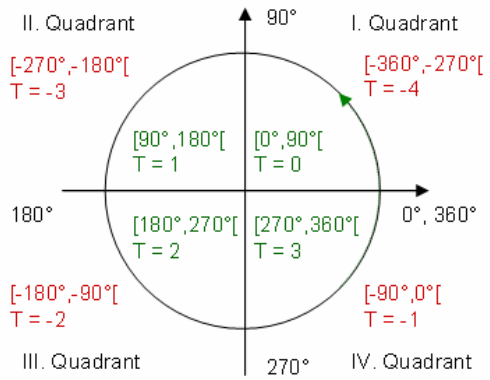


Open Online Output Data



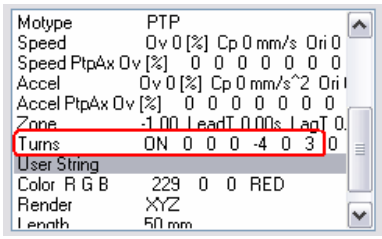
The axis values and the Turn-Values for axis 1,4 and 6 are displayed: T1=0 T4=0 T6=0

Moving axis 4 to -60° and axis 6 to +120° results in Turns T4=-1 T6 = 1



Example, Turn-Interval 90°

Turns can be saved with Tags or programmed, see example: „`Tutorial\Proj_example_erpl\functions_turns.cel`“



Tag „T6“ uses Turns (ON) T4=-4, T6=3 (axis 4 = -360°, axis 6 = 360°)

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

ERCL - Command

```
ERC TURN_INTERVAL Ax1 ...Axn [deg]

PTP_CALC_MODE calc_mode

mit calc_mode =
       SHORTEST_ANGLE,
       MATH,
       IN_TRAVEL_RANGE,TRAVEL_RANGE
       TURN

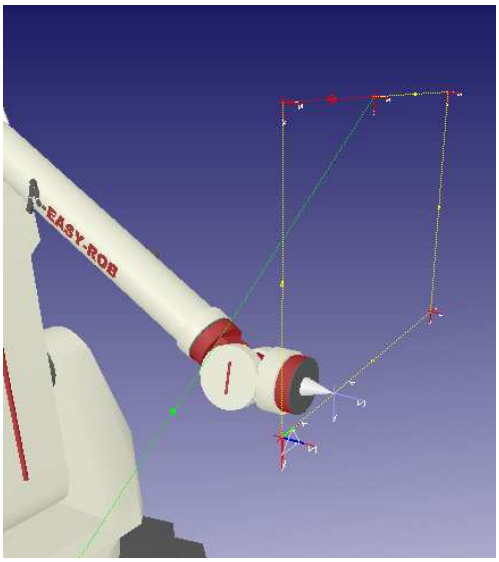
ERC TURN Turn_Ax1 ... Turn_Axn
```

Per default PTP_CALC_MODE is SHORTEST_ANGLE. Rotational joints in the target are calculated depending on current joints.

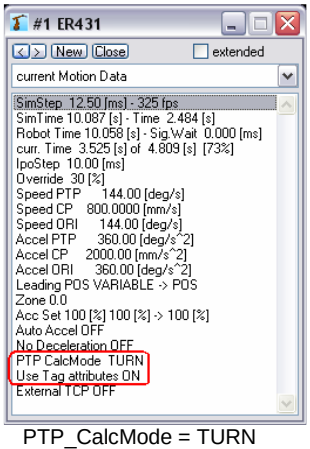
Setting to MATH, results in a calculation for rotational joints in the range [-180,180°].

Setting to IN_TRAVEL_RANGE or TRAVEL_RANGE results in rotational joints within the robot travel ranges, if possible.

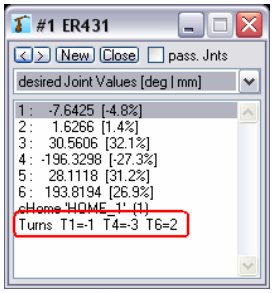
Setting to TURN, the programmed Turns are used to calculate the programmed target joint values. In case the Turn values are invalid, the PTP target is unreachable.
Error: UNREACH



Example: "functions_turns.cel"



PTP_CalcMode = TURN
Use Tag Attributes = ON



Ax1 = -7.64° -> T1 = -1
Ax4 = -196.3° -> T4 = -3
Ax6 = 193.8° -> T6 = 2

ERCL-Example:

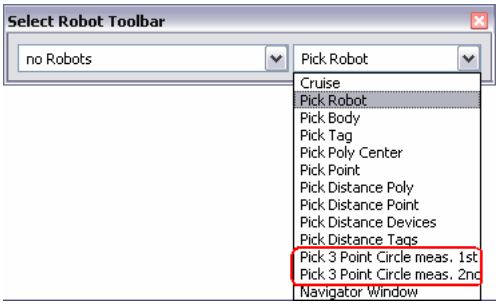
```
ERC TURN_INTERVAL 90 0 0 90 0 90
PTP_CALC_MODE TURN
TURN 0 @ @ -2 0 1
PTP T_1
PTP_CALC_MODE SHORTEST_ANGLE
PTP T_2
```


EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

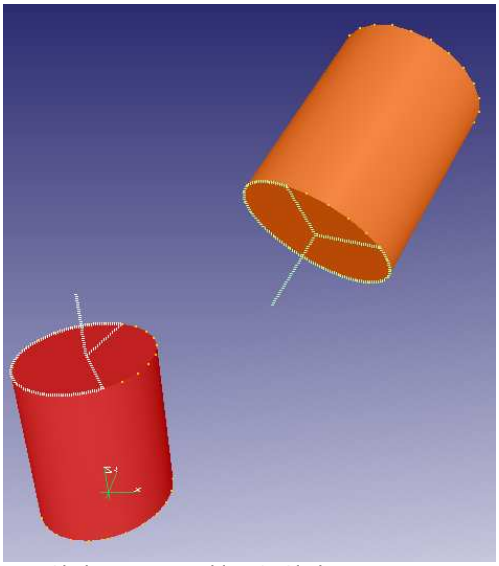
Measurement Functions

The measurement of circles and distances between devices, center of circles, surfaces and vertices is improved. Especially measuring circles when creating new kinematics is helpful, when the technical data are not available, just the geometries.

Enable measuring from „Select Robot Toolbar“ or from the Navigator Window.

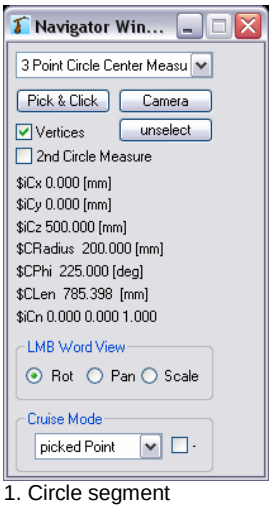


To measure the distance between circles, select for the
1st circle:„Pick 3 Point Circle meas. 1st“ and then for the
2nd circle „Pick 3 Point Circle meas. 2nd“.

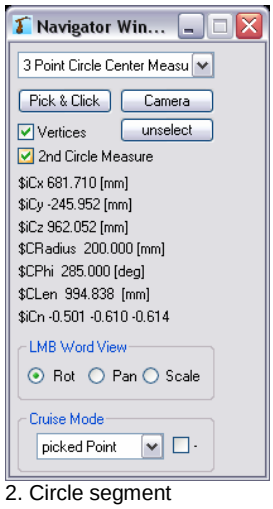


1. Circle segment white, 2. Circle segment green

The results are shown in the Navigator Window

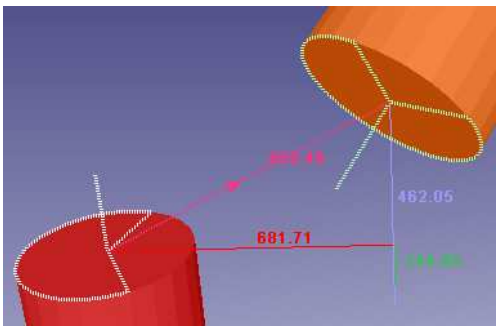


1. Circle segment



2. Circle segment

To measure the distance select for example „Pick Distance Poly“ and pick the circle segments.



Distance of two circle segments.

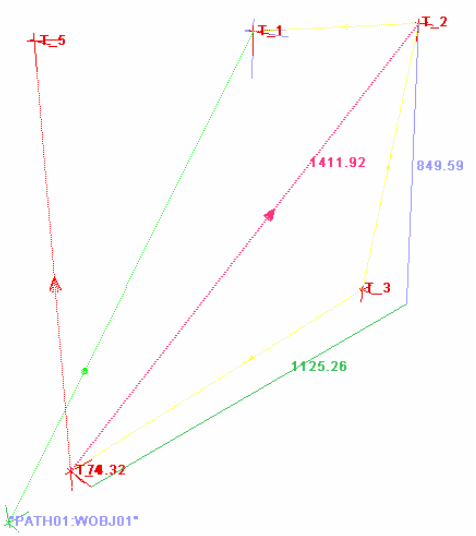
Circle segments can be used together with the SNAP-Function, to locate devices, geometries or Tags. Here the surface normal is used.

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

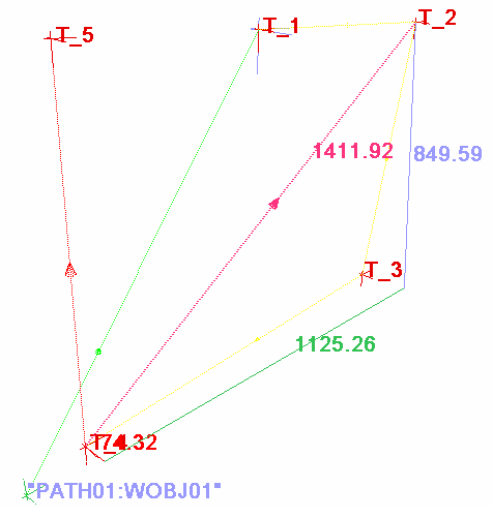
Character Font Size

Monitors and more significant notebooks have a high resolution, so the characters for geometries and tags inside the 3D scene are very small. With the new version, the font size can be increased and saved into the environment file “easy-rob.env“ afterwards.

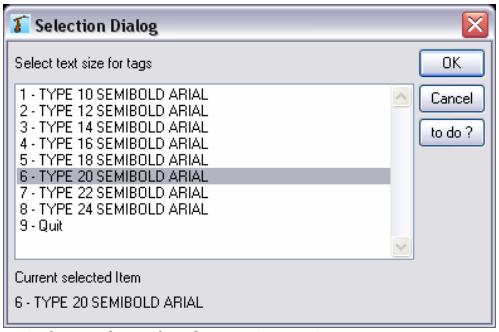
Select from the View Menu:
Render > Text Size Standard
or
Render > Text Size for Tags & Paths



Example: Standard-font size, TYPE 12 SEMIBOLD ARIAL



Example: font size, TYPE 20 SEMIBOLD ARIAL



Select a font size from 10pt to 24pt

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

ERPL-Commands

PTP_CALC_MODE SHORTEST_ANGLE [TURN, MATH, IN_TRAVEL_RANGE]

Calculation specification for PTP motion

SHORTEST_ANGLE	– shortest angle
TURN	– use TURN parameter
MATH	– mathematical within [-180°,180°]
IN_TRAVEL_RANGE	– within valid travel ranges if possible

TURN Turn_Ax1 ... Turn_Axn

TURN-Values for each axis Ax1…Axn, for the following PTP Target

ERCL- Commands

ERC TURN_INTERVAL Ax1 ...Axn [deg]

TURN -Interval for each axis Ax1…Axn, within [0°,360°[

ERC TRANSPARENCY BODY [ROBOT,TOOL] bodyname alpha

Set transparency for each single part or body. alpha = [0,1]

ERC TRANSPARENCY BODY_GRP [ROBOT_GRP,TOOL_GRP] alpha

Set transparency for each for all parts in a group. alpha = [0,1]

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

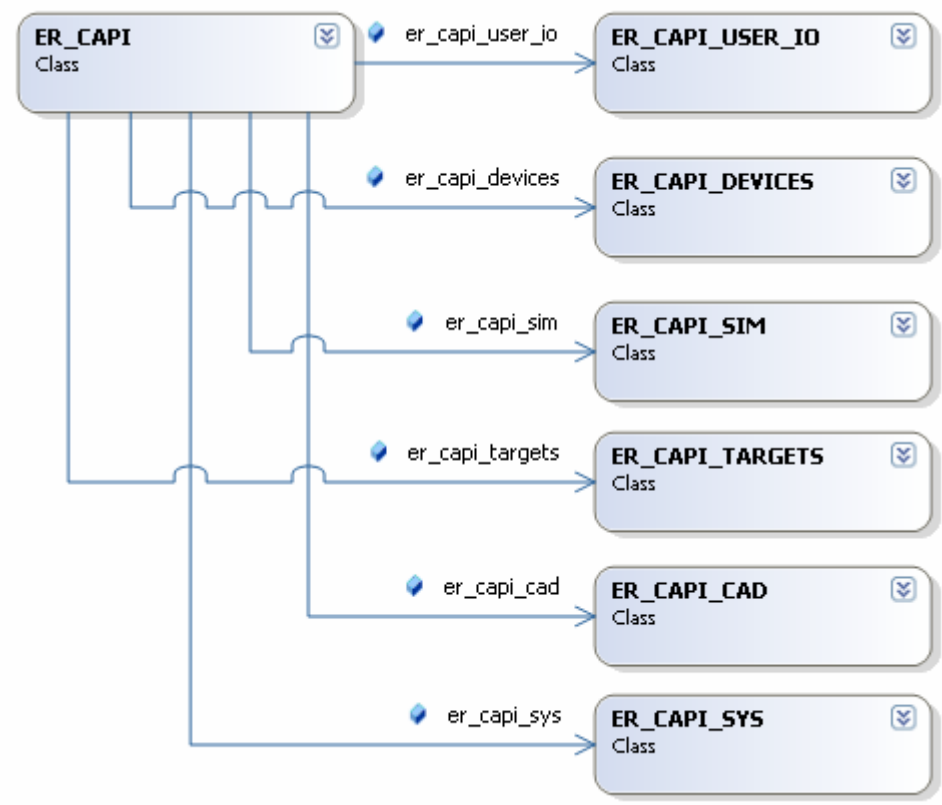
New Method-Class ER_CAPI

ER_CAPI

The new exported class ER_CAPI structures all EASY-ROB™ API-Functions and simplifies the usage. Class ER_CAPI is a pure methods-class, defined in header „./er_dvlp/**er_capi.h**“. All methods are Standard ANSI C.

„Old“ ANSI C Functions, defined in header „./er_dvlp/**er_dvlp.h**“ and „./er_dvlp/**er_dvlp_ext.h**“, are still available for compatible reasons..

The method class ER_CAPI was brake down in the following classes.



ER_CAPI_USER_IO: method class for interaction with EASY-ROB™

ER_CAPI_USER_IO_FILE	Load, save work cells, robots, etc.
ER_CAPI_USER_IO_DIALOG	Supplied dialogs, i.e. to enter and show values
ER_CAPI_USER_IO_PICK	Methods to „pick“ objects etc.
ER_CAPI_USER_IO_CRUISE	Methods to manipulate the 3D Scene

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

ER_CAPI_DEVICES: method class to create, attach, update devices, for kinematics calculations, motion planning and -execution

ER_CAPI_ROB	kinematics and transformations
ER_CAPI_MOP	motion planning and -execution

ER_CAPI_ROB: method class for kinematics and transformations

ER_CAPI_ROB_KIN	forward-, Inverse kinematics, desired robot joints, tools, position w.r.t. world and reference system
ER_CAPI_ROB_ATRIBUTES	robot/device attributes, travel ranges, max. speeds, home positions, etc.
ER_CAPI_ROB_DYN	dynamics, controller parameter, sampling rates

ER_CAPI_MOP: method class for motion planning and -execution

ER_CAPI_MOP_DATA	start-, target data, motion time, etc.
ER_CAPI_MOP_PATH	path specifications, motion type (PTP, LIN, CIRC), speeds, acceleration, waiting time, etc.
ER_CAPI_MOP_PREP	motion planning
ER_CAPI_MOP_EXEC	motion execution

ER_CAPI_SIM: method class for simulation settings

ER_CAPI_SIM_MONITORING	monitoring of travel ranges, speeds, etc.
ER_CAPI_SIM_COLLISION	collision, tolerances, etc.
ER_CAPI_SIM_CAMERA	camera settings
ER_CAPI_SIM_ERPL	executing ERPL- and ERCL commands
ER_CAPI_SIM_TRACK	TCP trace, reference and visualization

ER_CAPI_TARGETS: method class for paths and tags

ER_CAPI_TARGETS_TAG	manipulation of tags, access to tag attributes
ER_CAPI_TARGETS_PATH	creating, deleting and manipulating paths

ER_CAPI_CAD: method class for 3D CAD Data import and -export

ER_CAPI_CAD_IO	access to geometries, their location, tessellation, etc.
ER_CAPI_CAD_CREATE	creating primitives such as cubes, cylinder, spheres, etc.
ER_CAPI_CAD_IMPORT	importing existing 3D geometries (IGP, STL, 3DS)
ER_CAPI_CAD_CONVERT	converting external neutral and native CAD formats (STEP, IGES, JT-Open, CATIA, Pro/E, and so on) into the internal IGP Part File Format.

ER_CAPI_SYS: method class for mathematical calculations, simulation status, units

ER_CAPI_SYS_UTILITIES	helping functions
ER_CAPI_SYS_MATHEMATICS	mathematical calculations, multiplications of homogenous matrices, conversion Euler angle, triangle calculations, formula parser, etc.
ER_CAPI_SYS_VIEW	graphical update of the 3D scene, refresh dialogs
ER_CAPI_SYS_PREVIEW	CAD-Preview
ER_CAPI_SYS_STATUS	unload objects (work cells, robots, tools, programs, etc.) simulation status
ER_CAPI_SYS_UNITS	setting and calculating units

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

Usage

To use the class ER_CAPI, three lines must be included.

```
#define ER_DllExport
#include "../er_dvlp/er_capi_types.h"
#include "../er_dvlp/er_capi.h"
```

For simple and easy access the methods in each class, the predefined static variables in header „er_capi.h“ can be used..

```
// main class
static ER_CAPI                  er_capi;              // 0        CAPI

// class ER_CAPI                                      // 0
static ER_CAPI_USER_IO          er_user_io;           // 1.
static ER_CAPI_DEVICES          er_devices;           // 2.
static ER_CAPI_SIM              er_sim;               // 3.
static ER_CAPI_TARGETS          er_targets;           // 4.
static ER_CAPI_CAD              er_cad;               // 5.
static ER_CAPI_SYS              er_sys;               // 6.

// class ER_CAPI_USER_IO                              // 1.       USER_IO
static ER_CAPI_USER_IO_FILE     er_user_io_file;      // 1.1.
static ER_CAPI_USER_IO_DIALOG   er_user_io_dialog;    // 1.2.
static ER_CAPI_USER_IO_PICK     er_user_io_pick;      // 1.3.
static ER_CAPI_USER_IO_CRUISE   er_user_io_cruise;    // 1.4.

// class ER_CAPI_DEVICES                              // 2.       DEVICES
static ER_CAPI_ROB              er_rob;               // 2.1.
static ER_CAPI_MOP              er_mop;               // 2.2.

// class ER_CAPI_ROB                                  // 2.1  ROB
static ER_CAPI_ROB_KIN          er_rob_kin;           // 2.1.1
static ER_CAPI_ROB_ATRIBUTES    er_rob_attributes;    // 2.1.2
static ER_CAPI_ROB_DYN          er_rob_dyn;           // 2.1.3

// class ER_CAPI_MOP                                  // 2.2  MOP
static ER_CAPI_MOP_DATA         er_mop_data;          // 2.2.1.
static ER_CAPI_MOP_PATH         er_mop_path;          // 2.2.2.
static ER_CAPI_MOP_PREP         er_mop_prep;          // 2.2.3.
static ER_CAPI_MOP_EXEC         er_mop_exec;          // 2.2.4.

// class ER_CAPI_SIM                                  // 3.       SIM
static ER_CAPI_SIM_MONITORING   er_sim_monitoring;    // 3.1.
static ER_CAPI_SIM_COLLISION    er_sim_collision;     // 3.2.
static ER_CAPI_SIM_CAMERA       er_sim_camera;        // 3.3.
static ER_CAPI_SIM_ERPL         er_sim_erpl;          // 3.4.
static ER_CAPI_SIM_TRACK        er_sim_track;         // 3.5.

// class ER_CAPI_TARGETS                              // 4.       TARGETS
static ER_CAPI_TARGETS_TAG      er_targets_tag;       // 4.1.
static ER_CAPI_TARGETS_PATH     er_targets_path;      // 4.2.
```


EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

```
// class ER_CAPI_CAD                                    // 5.       CAD
static ER_CAPI_CAD_IO              er_cad_io;           // 5.1.
static ER_CAPI_CAD_CREATE          er_cad_create;       // 5.2.
static ER_CAPI_CAD_IMPORT          er_cad_import;       // 5.3.
static ER_CAPI_CAD_CONVERT         er_cad_convert;      // 5.4.

// class ER_CAPI_SYS                                    // 6.       SYS
static ER_CAPI_SYS_UTILITIES       er_sys_utilities;    // 6.1.
static ER_CAPI_SYS_MATHEMATICS     er_sys_mathematics;  // 6.2.
static ER_CAPI_SYS_VIEW            er_sys_view;         // 6.3.
static ER_CAPI_SYS_PREVIEW         er_sys_preview;      // 6.4.
static ER_CAPI_SYS_STATUS          er_sys_status;       // 6.5.
static ER_CAPI_SYS_UNITS           er_sys_units;        // 6.6.
```

Simple Examples

1. Reading and displaying current robot axis data

```
float *q = er_rob_kin.inq_q_solut(); // pointer to desired joint data
int n_dof = er_rob_kin.inq_num_active_jnts(); // number of robot joints
// show cJoint in Message Window
er_user_io_dialog._info_line_msg_q(0,"Joints",q);
```

2. Running a single command “Home 1”

```
// Check if a Cell or Robot is loaded
if (er_sys_status.ChkRobotCellLoaded(1)==1) return;
// Check if a program is currently running
if (er_sim_monitoring.ChkPrgRunning(0)) return;
er_sim_erpl.RunProgram("Home 1");
```

3. Running a complete program

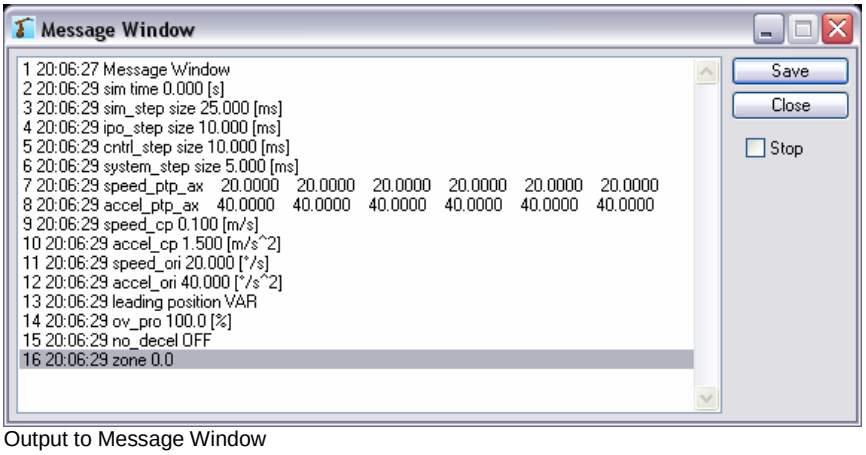
```
// Check if a Cell or Robot is loaded
if (er_sys_status.ChkRobotCellLoaded(1)==1) return;
// Check if a program is currently running
if (er_sim_monitoring.ChkPrgRunning(0)) return;
er_sim_erpl. RunProgram();
```


EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

4. Reading the current path specification fort he motion planner

```
er_user_io_dialog._info_line_msg(0,"sim time %.3f [s]",
       *er_sim.inq_sim_time());
er_user_io_dialog._info_line_msg(0,"sim_step size %.3f [ms]",
       *er_sim.inq_sim_step()*m2mm);
er_user_io_dialog._info_line_msg(0,"ipo_step size %.3f [ms]",
       *er_mop_data.inq_ipo_data_dt_ipo()*m2mm);
er_user_io_dialog._info_line_msg(0,"cntrl_step size %.3f [ms]",
       *er_rob_dyn.inq_cntrl_step()*m2mm);
er_user_io_dialog._info_line_msg(0,"system_step size %.3f [ms]",
       *er_rob_dyn.inq_system_step()*m2mm);

int n_dof = er_rob_kin.inq_num_active_jnts();
er_user_io_dialog._info_line_msg_v(0,"speed_ptp_ax",
       er_mop_path.inq_ipo_path_vq_axis(),n_dof,1);
er_user_io_dialog._info_line_msg_v(0,"accel_ptp_ax",
       er_mop_path.inq_ipo_path_aq_axis(),n_dof,1);
er_user_io_dialog._info_line_msg(0,"speed_cp %.3f [m/s]",
       *er_mop_path.inq_ipo_path_vx());
er_user_io_dialog._info_line_msg(0,"accel_cp %.3f [m/s^2]",
       *er_mop_path.inq_ipo_path_ax());
er_user_io_dialog._info_line_msg(0,"speed_ori %.3f [°/s]",
       *er_mop_path.inq_ipo_path_vx_ori());
er_user_io_dialog._info_line_msg(0,"accel_ori %.3f [°/s^2]",
       *er_mop_path.inq_ipo_path_ax_ori());
er_user_io_dialog._info_line_msg(0,"leading position %s",
       *er_mop_path.inq_ipo_path_leading_position()==IPO_LEADING_POS?"POS" :
       *er_mop_path.inq_ipo_path_leading_position()==IPO_LEADING_ORI?"ORI" :
       "VAR");
er_user_io_dialog._info_line_msg(0,"ov_pro %.1f [%%]",
       *er_mop_path.inq_ipo_path_ov_pro());
er_user_io_dialog._info_line_msg(0,"no_decel %s",
       *er_mop_path.inq_ipo_path_no_decel()?"ON":"OFF");
er_user_io_dialog._info_line_msg(0,"zone %.1f",
       *er_mop_path.inq_ipo_path_zone());
```



Output to Message Window

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

API-Methods

USER_IO_PICK

- **int ER_CAPI_USER_IO_PICK::PickSnapDeviceTo(int SnapToMode=0)**

  // Enables the user to snap a device to a polygon or vertex
  // SnapToMode 0-polygon, 1-vertices

- **int ER_CAPI_USER_IO_PICK::PickSnapToolTo(int SnapToMode=0)**

  // Enables the user to snap a tool to a polygon or vertex
  // SnapToMode 0-polygon, 1-vertices

- **int ER_CAPI_USER_IO_PICK::PickSnapTagTo(int SnapToMode=0)**

  // Enables the user to snap a tag to a polygon or vertex
  // SnapToMode 0-polygon, 1-vertices

- **int ER_CAPI_USER_IO_PICK::PickSnapBodyTo(int SnapToMode=0)**

  // Enables the user to snap a body to a polygon or vertex
  // SnapToMode 0-polygon, 1-vertices

- **int ER_CAPI_USER_IO_PICK::PickDistanceDeviceMeasure(int hold_first_point=0)**

  // Enables the user to measure the distance between two devices
  // hold_first_point 1-holds first selected point

- **int ER_CAPI_USER_IO_PICK::PickDistanceTagMeasure(int hold_first_point=0)**

  // Enables the user to measure the distance between two tags
  // hold_first_point 1-holds first selected point

MOP_PATH_DATA

- **int *ER_CAPI_MOP_PATH::inq_ipo_path_path_motion_idx(void)**

  // current Path-Index
  // 0 not a tag motion,
  // >0 Index of cPath containing the TargetTag
  Return Pointer to value

- **int *ER_CAPI_MOP_PATH::inq_ipo_path_tag_motion_idx(void)**

  // current Tag-Index
  // 0 not a tag motion,
  // >0 Index of TargetTag
  Return Pointer to value

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

- **int ER_CAPI_USER_IO_PICK::Pick3PointCircleMeasure (int vertices=1, int second_circle_measure=0)**

  // Enables the user to measure the center of a circle by picking three points
  // vertices 0-pcik polygon, 1-pick vertex
  // second_circle_measure 0-measure 1st-, 1-measure 2nd circle

TARGETS_TAG

Methods to read and set tag attributes, such as name, speed and accels.

- **int ER_CAPI_TARGETS_TAG::cTagClone(int create_new=0)**

  // clones current Tag in current path at Tcp of cRobot
  // if cPath is empty a new tag will be created on TCP if create_new is set 1

- **int ER_CAPI_TARGETS_TAG::chk_rob_tag_idx(int idx)**

  // check for valid tag_idx, 0-OK, 1-idx does not exist

- **char *ER_CAPI_TARGETS_TAG::inq_rob_tag_prefix_idx(int idx)**

  // tag prefix name

- **char *ER_CAPI_TARGETS_TAG::inq_rob_tag_name_idx(int idx)**

  // tag name

- **char *ER_CAPI_TARGETS_TAG::inq_rob_tag_prefix_name_idx(int idx)**

  // complete tag name

- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_motype_idx(int idx)**

  // tag motion type: TAG_MOTYPE_DEF, TAG_MOTYPE_PTP, TAG_MOTYPE_LIN
  // TAG_MOTYPE_CIRC, TAG_MOTYPE_VIA

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_ptp_idx(int idx)**

  // ptp speed, 0-def, [rad/s] for tag_idx

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_cp_idx(int idx)**

  // cp speed, 0-def, [m/s] for tag_idx

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_ori_idx(int idx)**

  // ori speed, 0-def, [rad/s] for tag_idx

- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_confdata_idx(int idx)**

  // robot configuration, 0-def

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_zone_idx(int idx)**

  // programmed zone value [mm,%]

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_ptp_idx(int idx)**
  // ori accel, 0-def, [rad/s^2] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_cp_idx(int idx)**
  // cp accel, 0-def, [rad/s^2] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_ori_idx(int idx)**
  // ori accel, 0-def, [rad/s^2] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_ov_pro_idx(int idx)**
  // programmed override, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_ptp_ov_idx(int idx)**
  // programmed override PTP speed, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_cp_ov_idx(int idx)**
  // programmed override CP speed, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_ori_ov_idx(int idx)**
  // programmed override ORI speed, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_ptp_ov_idx(int idx)**
  // programmed override PTP accel, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_cp_ov_idx(int idx)**
  // programmed override CP accel, 0-def, [0-200%] for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_ori_ov_idx(int idx)**
  // programmed override ORI accel, 0-def, [0-200%] for tag_idx
- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_auto_accel_idx(int idx)**
  // auto acceleration, 0-def, 1-ON for tag_idx
- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_acc_set_acc_idx(int idx)**
  // AccSet Acc, 100-def [20-100%] for tag_idx

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_acc_set_ramp_idx(int idx)**

  // AccSet Ramp, 100-def [20-100%] for tag_idx

- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_no_decel_idx(int idx)**

  // No decelaration at target, 0-def, 1-enabled for tag_idx

- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_leading_pos_idx(int idx)**

  // 2-def, 1-Pos, 2-VAR, 0-Ori for tag_idx

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_lead_time_idx(int idx)**

  // wait time before move begins, 0-def [s] for tag_idx

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_lag_time_idx(int idx)**

  // wait time when target reached, 0-def [s] for tag_idx

- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_lin_ori_mode_idx(int idx)**

  // 4-def, 0-LIN_VARIABLE, 1-LIN_FIX, 2-LIN_TANGENTIAL, 3-LIN_AUX,
  // 4-LIN_QUATERNION for tag_idx

- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_circ_ori_mode_idx(int idx)**

  // 5-def, 0-CIRC_VARIABLE, 1-CIRC_FIX, 2-CIRC_TANGENTIAL, 3-CIRC_AUX,
  // 4-CIRC_VARIABLE2, 5-CIRC_QUATERNION for tag_idx

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_speed_ptp_ax_ov_idx(int idx)**

  // ptp_ax override speed 0-def [0-200%] for tag_idx

- **float *ER_CAPI_TARGETS_TAG::inq_rob_tag_accel_ptp_ax_ov_idx(int idx)**

  // ptp_ax override accel 0-def [0-200%] for tag_idx

- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_turn_use_idx(int idx)**

  // 0-do not use turn values (def), 1-use turns for tag_idx

- **int *ER_CAPI_TARGETS_TAG::inq_rob_tag_turn_ax_idx(int idx)**

  // turn values depending on current robots turn interval for tag_idx

- **char *ER_CAPI_TARGETS_TAG::inq_rob_tag_user_string_idx(int idx)**

  // user defined Tag string for tag_idx

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

TARGETS_PATH

Methods to reattach a path with index 'path_idx' by device idx, name or unique robot ID ‘uid’

path_idx	idx of path to reattach to
new_reference_type	new referecne type = REF_BASE=2, REF_TOOL=3, REF_WORLD=4, REF_CAD=7, REF_TIP=8, 9-REF_JNT
new_reference_device_idx	get reference device by idx
new_reference_cad_grp_type	reference grp_type = (UNDEF_GRP,ROBOT_GRP, TOOL_GRP,BODY_GRP) if new_reference_type = REF_CAD
new_reference_cad_name	reference cad name if new_reference_type = REF_CAD
new_reference_jnt_idx	reference jnt idx if new_reference_type = REF_JNT
keep_world_position	True will keep the world position of path
Return:	0-OK, 1-Error

- ```
  int Path_ReAttach_by_name(int path_idx, int new_reference_type,
      char *new_reference_device_name,
      int new_reference_cad_grp_type=ROBOT_GRP,
      char *new_reference_cad_name=NULL, int new_reference_jnt_idx=0,
      bool keep_world_position=false)
  ```

  // Reattaches the path with index 'path_idx' by device name

- ```
  int Path_ReAttach_by_idx (int path_idx, int new_reference_type,
      int new_reference_device_idx,
      int new_reference_cad_grp_type=ROBOT_GRP,
      char *new_reference_cad_name=NULL, int new_reference_jnt_idx=0,
      bool keep_world_position=false)
  ```

  // Reattaches the path with index 'path_idx' by device idx

- ```
  int Path_ReAttach_by_uid (int path_idx, int new_reference_type,
      long new_reference_device_uid,
      int new_reference_cad_grp_type=ROBOT_GRP,
      char *new_reference_cad_name=NULL, int new_reference_jnt_idx=0,
      bool keep_world_position=false)
  ```

  // Reattaches the path with index 'path_idx' by device uid

CAD_IO

- ```
  int *ER_CAPI_CAD_IO::inq_body_show_transparent(void *body_handle)
  ```

  // return pointer to show_transparent flag
  // body_handle geometry handle

- ```
  float *ER_CAPI_CAD_IO::inq_body_transparency(void *body_handle)
  ```

  // return pointer to transparency value
  // body_handle geometry handle

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com
  ```

Windows® 7 32- and 64-Bit

EASY-ROB™ is compatible with Windows® 7 Professional Ultimate & Enterprise (32-Bit and 64-Bit). Important for a high performance is graphics acceleration card. We recommend nVIDIA GeForce or ATI graphics CPU with the OpenGL 2.0 driver installed and 256 MB VRAM at least.

Further more we recommend 4GB RAM, so Windows® as 32-Bit Version is able to supply the maximal possible RAM for the application. Using Windows® 7, up to 3.25 GB are available.

At the moment we notice that the CPU together with the graphics performance is powerful enough. The bottle neck is in most cases the minor memory in 32-Bit applications, so in EASY-ROB™.
The problem occurs especially when using geometries with a high level of detail. So it is more and more important using the possibilities to influence and reduce the tessellation, supplied with the CAD data import in EASY-ROB™.

To solve the memory resource problem, we work on a 64-Bit solution in the future.

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com

Contact

EASY-ROB 3D Robot Simulation Tool

Stefan Anton
Hans - Thoma - Str. 26a, 60596 Frankfurt/Main, Germany
Tel. +49 (0) 69 677 24 287
Fax. +49 (0) 69 677 24 320

Sales Office - North
Gregor Hölting
West I 17, 48324 Sendenhorst, Germany
Tel. +49 (0) 2506 81 65 73
Fax. +49 (0) 2506 81 65 74

Email: contact@easy-rob.com
sales@easy-rob.com

Web: www.easy-rob.com

EASY-ROB Customer area

Online available: Program Updates and Robot libraries

Web: www.easy-rob.com/en/special/customer-area

Access data:

User: customer
Password: **********

Notes

EASY-ROB™ Stefan Anton Hans-Thoma-Str.26a 60596 Frankfurt/Main Germany Tel. +49 69 677 24-287 Fax. +49 69 677 24-320 info.er@easy-rob.com www.easy-rob.com