CANape ASAM-MCD3 Interface
Version 3.2
2018-06-19
Application Note AN-AMC-1-103

Author Vector Informatik GmbH
Restrictions Public Document
Abstract This is document is a general introduction explaining the CANape ASAM-MCD3 Interface (CANapeAPI).

Table of Contents
1 Overview .................................................................................................................................2
  1.1 Detailed Programming Documentation.............................................................................2
2 Use Cases ..............................................................................................................................2
  2.1 Use Case 1: CANape Running on Local Host .................................................................3
  2.2 Use Case 2: TCP/IP Remote Control of CANape .........................................................4
  2.3 CANape in ASAP3 Mode ...............................................................................................4
3 Creating local host client application in MSVC++ 2010 ..................................................5
  3.1 Creating a MSVC++ Project .........................................................................................5
4 Contacts ...............................................................................................................................7
1 Overview

This application note offers the reader a basic introduction to the CANape API. In order to understand the CANape API specified in this documentation, the reader must first be familiar with:

- CANape Measurement, Calibration, and Diagnostic Tool
- C/C++ Programming Language
- ASAM MCD-3 Interface

Readers that are not already familiar with the CANape tool, or C/C++ based programming languages will find this document insufficient for developing an understanding of how to build applications which utilize the CANape API. For more information on the ASAM MCD-3 standard navigate to http://www.asam.net/. For detailed programmatic examples using the CANape API please refer to section 1.1 below.

1.1 Detailed Programming Documentation

Within every installation of CANape lies a more detailed programming document named the “CANapeAPI Help”. In order to access the CANapeAPI Help document navigate to your Windows Start menu and click on the CANapeAPI Help icon (Figure 1.0 below). This documentation contains a very comprehensive description of the CANapeAPI class hierarchy, along with example code for all class methods and properties.

Figure 1.0

2 Use Cases

There are two possible ways to implement the ASAP3 interface with CANape. The first use case involves running CANape on your local machine. This would involve CANape and the client application running on the same local host. The second use case implements a TCP/IP remote connection between CANape (Server) and a client application (Figure 2.0).
2.1 Use Case 1: CANape Running on Local Host

In order to develop a client application using the CANape API it is necessary to link your development environment to the CANapAPI.LIB in order to reference the CANapAPI.DLL. Local host ASAP3 functionality is realized with this library (Figure 3.0).

The CANapeAPI.DLL provides the following functionalities:

- Exchange of data between CANape and other external applications
- Creation of an mdf (Measurement Data Format) file to log Measurement Data
- Automation of calibration sequences
- Execution of script files and services
2.2 Use Case 2: TCP/IP Remote Control of CANape

In order to develop a remote client application using the CANape API it is necessary to link your development environment to the CANaptcp.LIB in order to reference the CANaptcp.DLL. Remote TCP/IP access to CANape is realized with this library (Figure 4.0).

Figure 4.0

The CANaptcp.DLL provides the following functionalities:

> Encompasses all CANapAPI.DLL functionality
> Remote TCP/IP access instead of local host access

2.3 CANape in ASAP3 Mode

Originally, the acronym ASAP referred to a European task force which focused on standardization of tools involved in the measurement, calibration, and diagnostics of electronic control units. The ASAP3 specification was a standard defined for the interfacing of a measurement and calibration system to an automation system in order to provide the capability of automatic computer controlled testing. Access to the ASAP3 interface is possible via two methods:

> CANape API
> CANape COM Interface

The CANape API is a C/C++ implementation of the ASAP3 interface. Access to ASAP3 functionality is realized by CANapeAPI.DLL routines. The CANape COM interface encapsulates the most commonly used subset of the CANape API. When developing a CANape API application, CANape is launched in ASAP3 mode. Further, there are two distinct ASAP3 modes when CANape is launched: non-modal mode and modal mode. Activating CANape in modal mode presents an “ASAP3 Active” dialog box to the user (Figure 5.0) preventing manual interaction with CANape. Launching CANape in non-modal mode provides connection to the automation system via CANape’s manual interface.

Figure 5.0
3  Creating local host client application in MSVC++ 2010
The following tools and components are used in this example:

- Microsoft Visual Studio C++ 2010 Professional Edition
- CANape v13.0
- CANapAPI.DLL
- CANapAPI.LIB
- CANapAPI.h
- test_console.cpp
- CANape's XCP Demo

Where are these files and demos located?

With every installation of CANape

3.1  Creating a MSVC++ Project
Using Microsoft Visual Studio create a C++ Win32 Console Application. Next, in order to use the CANape API functionality the following steps must be taken:

1. Include the CANapAPI.h header file in your project along with the test_console.cpp file
2. Navigate to your project’s Property Pages then Linker->General page and make sure the Additional Library Directories points to the correct folder path that contains the CANapAPI.LIB.

3. Next, navigate to Linker->Input page and enter CANapAPI.lib for Additional Dependencies.
4 Contacts

For a full list with all Vector locations and addresses worldwide, please visit http://vector.com/contact/.