

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 1/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## Table of Content

**GENERAL ..... 2**

**Release Information ..... 2**

**Introduction ..... 2**

**Installation ..... 3**

**Hardware and Software requirements ..... 4**

**Compatibility ..... 5**

**Updates in PC SDK 5.10 vs. 5.09 ..... 6**

**Updates in PC SDK 5.11 vs. 5.10 ..... 6**

**Updates in PC SDK 5.12 vs. 5.11 ..... 7**

**Updates in PC SDK 5.13 vs. 5.12 ..... 7**

**Information 5.11.01 ..... 8**

**Information 5.12 ..... 9**

**Information 5.12.01 ..... 10**

**Information 5.12.02 ..... 10**

**Information 5.13 ..... 11**

**Corrected “Product Defect Document” 5.12 PDD ..... 12**

**Corrected “Product Defect Document” 5.12.01 PDD ..... 12**

**Corrected “Product Defect Document” 5.12.02 PDD ..... 12**

**Corrected “Product Defect Document” 5.13 PDD ..... 12**

**Known Limitations ..... 13**

**Product Support ..... 13**

# ABB AB Robotics

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 2/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## General

### Release Information

The information should be considered as last minutes information and most up-to-date.

For more information please visit RRI Homepage:

<http://prodapp2.se.abb.com/rri/>

### Introduction

This file contains release notes for PC SDK 5.13.

Previously PC SDK and FlexPendant SDK were distributed as the product Robot Application Builder (RAB). Starting with RobotWare 5.13, PC SDK and FlexPendant SDK are distributed together with RobotStudio. The SDKs are not affected by this change.

PC SDK 5.13 is included in the RobotStudio installer, which is distributed on the RobotWare 5.13 DVD. It can be used for free by anyone who wants to develop a customized PC operator interface, for the IRC 5 controller. It can also be used to develop RobotStudio add-ins that interacts with the controller. With PC SDK it is possible to create an application that connects to one or several IRC5 controllers, real as well as virtual. No license is required to develop applications using PC SDK. The end user of a PC SDK SDK application, needs to have the option PC Interface on the targeted controller.

The installation includes software, documentation and tools as specified below:

#### Software

PC SDK (5.13)

#### Documentation

Application manual – PC SDK (5.13), Html Help and Pdf).

PC SDK Reference Documentation (5.13), documentation of class libraries with method signatures in C# and Visual Basic (Html Help).

#### Tools

ABBControllerAPI.msm - merge module including the PS SDK dlls to be used when a PC SDK application is deployed to a customer's PC

ABB Industrial Robot Communication Runtime.msi - to be used when a PC SDK application is deployed to a PC without RobotStudio

After installation the documentation can be launched from Windows Start Menu\Programs\ABB Industrial IT\Robotics IT\Robot Studio 5.13.

In no event shall ABB be liable for incidental or consequential damages arising from use of this product, of the software and hardware described in relating product documentation.

Visit our web site at <http://www.robotstudio.com/community> for information and updates. Click the Download symbol to the right in order to download RobotStudio 5.13, including PC SDK 5.13 for free.

Click Developer Tools to learn more about the PC SDK and visit the User Forum, where developers discuss software problems and solutions online.



# ABB AB Robotics

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 3/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## Installation

To install PC SDK click *RobotStudio* on the RobotWare & RobotStudio DVD. If you select the default installation option Full, PC SDK will be installed. If you want to install only PC SDK and not RobotStudio select the installation option **Custom**. PC SDK will be installed side by side with any previous version. If you want further details about the installation click *Browse DVD Contents* on the RobotWare & RobotStudio DVD.

Before you start the installation you are recommended to read chapter 2, *Installation and development environment*, in the PC SDK Application Manual which is available on the Documentation DVD in PDF format.

Microsoft Visual Studio development environment is used to develop PC SDK applications. Visual Studio 2005 Express or better, or Visual Studio 2008 Express or better is required.

PC SDK 5.13 is installed side by side with any previous PC SDK version.

The default installation path is C:\Program Files\ABB Industrial IT\Robotics IT\SDK\PC SDK 5.13.

Before you start the installation you are recommended to read chapter 2, *Installation and development environment*, in the PC SDK Application Manual which is available on the DVD in PDF format.

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 4/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## Hardware and Software requirements

### Software requirements

**Microsoft Windows Vista SP1**

**Microsoft Windows XP SP3**

**Microsoft Windows 7**

**Microsoft Windows 7 – 64bit edition**

***Note!** PC SDK application that shall be executed on Windows 7 – 64 bit edition, must be built with target platform set to “x86”. The default setting in VisualStudio 2008 is “AnyCPU”*

**Microsoft Visual Studio 2005** or

PC SDK requires Express edition or better.

**Microsoft Visual Studio 2008**

PC SDK requires Express edition or better.

***Note!** PC SDK requires Robot Communications Runtime to be installed on your PC. It comes with the RobotStudio installation, but can also be installed separately from C:\Program Files\ABB Industrial IT\Robotics IT\Robot Application Builder 5.12\redistributable\RobotCommunicationRuntime after you have installed PC SDK.*

### Recommended hardware

10 MB free disk-space on the installation hard disk

### IRC5 requirements

- RobotStudio 5.13 for building a test system and for debugging and testing in the virtual environment
- RobotWare option "PC Interface" for communication with a real IRC5 controller.

***Note!** PC applications developed with PC SDK 5.13 require RobotWare version 5.07 or higher on the IRC5 controller.*

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 5/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## Compatibility

### Compatibility

PC SDK 5.13 and 5.12 are compatible, there are no breaking changes.

PC SDK communicates with the robot controller using the Robot Communication Runtime, which is designed to be backwards compatible with earlier versions of RobotWare.

Note! Functionality introduced in later versions of RobotWare will not be available for a PC SDK application that is connected to a controller with an older version of RobotWare.

For example the Messaging functionality is only supported on RobotWare 5.10 and above. This means that a PC SDK application cannot use the `Ipc` class when communicating with a controller with RobotWare 5.09.

The code will compile, but an exception will be thrown at runtime. Application developers are responsible for handling this scenario in their applications. Please refer to the PC SDK Reference Documentation for details.

Note! Compatibility between RobotWare revisions is guaranteed (PC SDK 5.13 will be compatible with PC SDK/RW 5.13.01 etc).

Compatibility with other products:

Use RobotStudio 5.13

### **EventHandlers and events**

For 5.09 the internal event architecture was completely redesigned.

1. All previous event handlers are [Obsolete] and existing events are changed to the `EventHandler<TEventArgs>` generic delegate.

**Ex:** This example shows how to change from the old to the new event handler type.

*// This old line will fail...*

```
myEventLog.MessageWritten += new MessageWrittenEventHandler( OnMessage );
```

*// and should be replaced like this...*

```
myEventLog.MessageWritten += new EventHandler<MessageWrittenEventArgs>( OnMessage );
```

*//or preferably...*

```
myEventLog.MessageWritten += OnMessage;
```

2. Previous versions of PC SDK used the Windows Thread Pool internally to raise events.

From 5.09 a single thread is used to dispatch all events. This reduces the risk of race conditions in client code. However, it makes it even more important to use the `Control.BeginInvoke( ... )/Control.EndInvoke( ... )` pattern to avoid event starvation.

Additionally, we now raise all events internally prior to any external subscribers, as this will reduce the risk of race conditions between inner and external subscribers.

If your application is based on `Control.Invoke( ... )` and not on `Control.BeginInvoke( ... )` all events will be serialized, both internally and externally. However in previous PC SDK versions you may encounter "out-of-order" events and thread pool starvation through the use of `Control.Invoke( ... )`.

Some public events raise an initial event immediately when the subscription is activated; however this is not consistent or by design and should therefore be avoided. Later versions of PC SDK will remove all initial events.

3. `Mastership.Request( ... )` throws an `InvalidOperationException` if the user is not authenticated against the controller, previous versions raised an `ArgumentException`.

# ABB AB Robotics

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 6/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## Updates in PC SDK 5.10 vs. 5.09

### **New features:**

#### New domains:

Messaging (Together with RAPID Message Queue this functionality represents a new, flexible way for a RAB application to interact with a RAPID task. It makes use of the IPC mechanism provided by Microsoft Windows OS for facilitating communications and data sharing between applications.)

#### Miscellaneous (new classes, events and methods):

High priority event subscriptions. (To speed up event notification you can set up subscription priorities. This applies to I/O signals and persistent RAPID data.)

MechUnit.DriveModule (Gets the drive module of the mechanical unit.)

Controller. ReleaseUnmanagedResources(Should be called periodically by the application developer to release unmanaged resources.)

## Updates in PC SDK 5.11 vs. 5.10

### **New features:**

#### Development environment:

Changed installation (supports working with several PC SDK versions side by side on a PC) Robot Application Builder 5.11 is installed side by side with any previous RAB version. It includes PC SDK 5.11 and FlexPendant SDK 5.11

Support for Windows Vista

Support for VS 2008

*Note! RAB 5.11 has not yet been fully tested with Visual Studio 2008.*

#### Miscellaneous (new classes, events and methods):

IOSystem.GetIOBuses (gets all IO buses of the controller)

IOSystem.GetIOUnits (gets all IO units of the controller)

IOBus properties (Name, State, Type)

IOUnit properties (Name, Bus, Address, Type, Logical State, PhysicalState)

IOUnit events (SignalChanged event, StateChanged event)

Task.SetProgramPointer

Task.SearchRapidSymbol & Task.GetRapidData now support accessing RAPID data declared in SHARED module

Limitation: RAPID data declared in a SHARED HIDDEN module cannot be accessed.

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 7/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## Updates in PC SDK 5.12 vs. 5.11

### New features:

*Remote UI-Instruction* enables an operator to communicate with the executing RAPID program from a PC SDK application (instead of using the FlexPendant) when a UI- or TP-RAPID instruction is executed. The following instructions are supported:

UIAlphaEntry,  
UIListView,  
UIMessageBox,  
UIMsgBox,  
UINumEntry,  
UINumTune,  
TPErase,  
TPReadFK,  
TPReadNum,  
TPWrite

To make use of this new function the PC SDK application sets up a subscription to the *UIInstructionEvent*. When notified of an event it creates a dialog according to the event arguments. The method *SendAnswer* transfers the operator response back to the RAPID program.

### Miscellaneous (new classes, events, methods and properties):

- RapidData properties: IsTaskPers (whether data is declared as PERS or TASK PERS), IsLocal (declared in a module that is not globally visible), TypeUrl (path to the declaration of the RAPID type), RapidType (eg. "num")
- Signal.Type (DigitalOut, GroupIn etc)
- Task methods: GetRobTarget, GetJointTarget and overloaded DeleteProgram, LoadProgramFromFile ( allows a PC SDK application to delete, load and start a RAPID program without any unnecessary delays)

## Updates in PC SDK 5.13 vs. 5.12

### DNUM support

- Dnum type in the RapidDomain namespace
- Support for 32 bit group signals with PCSDK

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 8/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## Information 5.11.01

### **Documentation for RAB FlexPendant SDK has to be rebuilt (DSE8261)**

The reference documentation for the FP SDK 5.11 has been updated.

### **NumEditor causes NullReferenceException on FlexPendant (DSE8270)**

When using the NumEditor control in a TpsControl, the ClickEvent raises a NullReferenceException in Taf.exe that requires the FlexPendant to be restarted. When using the DataEditor control in a TpsControl, it does not display an alpha pad when it is clicked. This has now been corrected.

### **PictureBox control not available in toolbox (DSE8358)**

The FP SDK PictureBox control was missing among the FP SDK controls added to the Visual Studio toolbox. It has now been added.

### **FP SDK application - unwanted entry in ABB menu (DSE8450)**

In RAB 5.10 an FP SDK application using TpsViewStartupTypes.Automatic in the TpsView attribute did not have an entry in the ABB menu of the FlexPendant. In RAB 5.11 this error was corrected. But no solution was provided for users who relied on the previous behavior and *do not want* an entry in the ABB menu for applications that are launched automatically at start-up.

In this revision (5.11.01) a new TpsView argument, StartPanelLocation.None, can be used to prevent *Automatic* and *Invisible* applications from showing up in the ABB menu. Note, however, that the FlexPendant project wizard has not yet been changed, so StartPanelLocation.None has to be applied directly in the code (instead of StartPanelLocation.Left /Right).

**Note!** Be aware that FP SDK applications that make use of *StartPanelLocation.None* using RAB 5.11.01 CAN NEVER be run on RobotWare 5.11, but only on RobotWare 5.11.01 and later!

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 9/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## Information 5.12

### **FP SDK - Error in designer when creating new VB project (DSE8267)**

When creating an FPSDK project for VB using an empty form (the default) as the startup view, an error message may be displayed in the designer the first time the viewer is opened. The problem does not occur when the startup type is *Form*, or in C#. Work around: Close and re-open the designer.

### **FPRapidData creates data object that cannot be disposed (DSE8273)**

When creating new instances of FPRapidData, FPToolCalibration and FPWorkObjectCalibration, some objects were created internally that could not be disposed by the SDK user. These appeared in the log as possible memory leaks. This has been corrected. The user must dispose the returned RapidDataType object returned by RapidData.DataType (See FP SDK Ref Help.)

### **Data binding with FP SDK controls caused unexpected updates in the controller (DSE8274)**

Before 5.12, the use of data binding with FlexPendant SDK controls could cause the value of a connected RAPID variable or signal to be changed unexpectedly at instantiation of the GUI control. This could happen if "Data Source Update Mode" of the binding source was set to OnPropertyChanged and a non-default value, not matching the actual value of the connected data/signal, was assigned to the control. For example, if the text of a label was set to "1" in design time and the label bound to an I/O signal with the value "0", the signal would be set to "1" once the label was created.

This error has now been corrected. The erratic behavior was first discovered in the new FP SDK controls 'Led' and 'Switch'. For security reasons, their data binding property was disabled in the designer in 5.11. The designer support for data binding for these controls has now been enabled.

### **DataEditor displays unwanted quotation marks with data (DSE8276)**

Text data, read through the FlexPendant SDK data binding components, does not contain any surrounding quotation marks.

### **No Start Menu link to Compliance tool in RAB Installation (DSE8325)**

From RAB 5.11 the compliance tool is found in the installation folder. SDK users who frequently use it can manually add the link to the start menu, quick launch, or desktop.

### **FPRapidData returns Cancel when OK button is pressed (DSE8326)**

The dialogs for RAPID data, tool calibration and work object calibration now return the correct result from user interaction (Cancel or OK).

### **Undocumented output from memShow command (DSE8327)**

Documentation changed.

### **Console command fpcmd -a output does not match documentation (DSE8328)**

Documentation changed.

### **Closable property in FP SDK doesn't work correctly (DSE8528)**

FlexPendant SDK applications that were launched automatically and had the property Closable set to false, were still launched with the close button activated. Applications that were launched on operating mode change were always opened with the close button enabled even if the property Closable was set to false. These problems have now been fixed.

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 10/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## FontName warning with RAB 5.12 (DSE8749)

The property FontName of type ABB.Robotics.Tps.Windows.Forms.TpsControl has been marked as obsolete in the FP SDK 5.12 release.

If the FontName property is used, a compiler warning will be generated, that informs the user that the property Font should be used instead (see below). VB: Me.Font = ABB.Robotics.Tps.Windows.Forms.TpsFont.Font10b C#: this.stBackupFolder.Font = ABB.Robotics.Tps.Windows.Forms.TpsFont.Font10b;

## Data binding source causes errors in Visual Studio (DSE8780)

In intermediate builds of FP SDK 5.12, version numbers were not updated correctly for its data binding components.

This caused Visual Studio to fail to generate code for any SignalBindingSource or RapiDataBindingSource components added to a view.

This is now corrected.

## RS does not support older FP app (DSE8844)

There is a remaining problem on the Virtual FlexPendant resulting in a CustomAttributeFormatException exception. The problem occurs when the SDK application is built towards a different revision compared to the one running. The problem is related to embedded images.

There are a number of work-arounds:

1. Recompile the SDK application towards a Robot Application Builder with the same revision number as the running system.
2. Remove the StartupType assembly attribute assignment in the SDK TpsView header, i.e., for the TpsView remove StartupType = TpsViewStartupTypes.....)
3. Use one of the new TpsView constructors that contain the TpsViewStartupTypes setting.

The original problem cannot be solved by the ABB development team because it is in software developed by Microsoft. A workaround is now implemented by adding constructors containing the start-up type (see work-around 3). The wizards creating new FlexPendant applications are updated and now use one of the new constructors.

## Information 5.12.01

No new info.

## Information 5.12.02

### From 5.12.02; it is possible to do remote axis calibration and update of revolution counters via PC SDK.

Four new methods in name space MotionDomain:

FineCalibrate - fine calibrate a specified TCP mechanical unit and axis. The current position of the axis will be the new calibration value.

SetRevolutionCounter - set revolution counter on a specified TCP mechanical unit and axis.

GetMechanicalUnitStatus - get status of mechanical unit.

GetSingleAxisStatus - get status of single axis

FineCalibrate and SetRevolutionCounter requires that the system is in Auto mode and Motors off state. These methods also require UAS grants for fine calibration and update rev counters.

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 11/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## Information 5.13

**It is now possible to access RAPID data that are declared in hidden modules and in hidden tasks.**

E.g

```
RapidData data;
Controller controller;
```

```
// Access data declared in a hidden task.
```

```
// A hidden task has the -hidden attribute defined in the Controller Configuration.
```

```
data = controller.Rapid.GetRapidData("hidden_task", "module1", "num1");
```

```
// Access data declared in a hidden module
```

```
// A hidden module has the -hidden attribute defined in the Controller Configuration.
```

```
data = controller.Rapid.GetRapidData("T_ROB1", "hidden_module1", "num2");
```

```
// Access data declared in a shared hidden module
```

```
// A shared hidden module has the -hidden and the -shared attributes defined in the Controller Configuration.
```

```
data = controller.Rapid.GetRapidData("num3");
```

```
// Access data declared in an installed module
```

```
// An installed module is loaded from a install script using the -install option.
```

```
data = controller.Rapid.GetRapidData("T_ROB1", "#SYS", "num4");
```

```
// Access data declared in an installed hidden module
```

```
// An installed module is loaded from an install script using the -install and -hidden options.
```

```
data = controller.Rapid.GetRapidData("T_ROB1", "#SYS", "num5")
```

### Simulate physical setting of input signals on a Virtual Controller

It is now possible to simulate the physical setting of the value of an input signal.

The previously existing method to set the value of a signal is similar to setting a signal from RAPID. Depending on the Access Level the controller safety system may or may not allow the signal value to be changed. When a physical signal is set by connecting it to +24V the controller cannot prevent it from happening, no matter what Access Level the signal has. By setting the property **-InputAsPhysical** on the **Signal** class, a subsequent write operation will simulate a physical write. This is important in a scenario where a PC SDK application shall set signal values in order to verify a RAPID program in a Virtual Controller

### Dnum variable not accessible from PC SDK (DSE9020)

It is now possible to access RAPID dnum variables via PC SDK.

A new class Dnum has been added.

### 32 bit group signals is now supported by the PC SDK. (DSE9386)

32 bit group signals is now supported by the PC SDK.

### Memory leak (DSE9493)

There was a memory leak in PCSDK when retrieving ipc messages. It leaked about 400 bytes per message it retrieves, this has now been corrected.

# ABB AB Robotics

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 12/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

<b><i>Corrected "Product Defect Document" 5.12</i></b>	<b><i>PDD</i></b>
--	-------------------

FPRapidData creates a copy of data object that cannot be disposed	DSE8273
DataEditor displays unwanted quotation marks with data	DSE8276
RAB Installation no longer creates Start Menu link to Compliance	DSE8325
FPRapidData returns a Cancel result when OK button is pressed	DSE8326
Undocumented output from memShow command	DSE8327
Console command fpcmd -a output does not match documentation	DSE8328
FontName warning with RAB 5.12	DSE8749
Data binding source causes errors in Visual Studio	DSE8780

<b><i>Corrected "Product Defect Document" 5.12.01</i></b>	<b><i>PDD</i></b>
---	-------------------

<b><i>Corrected "Product Defect Document" 5.12.02</i></b>	<b><i>PDD</i></b>
---	-------------------

RW5.12 RC2 - RAB PC SDK: loss of performance when transferring	DSE8903
--	---------

<b><i>Corrected "Product Defect Document" 5.13</i></b>	<b><i>PDD</i></b>
--	-------------------

Dnum variable not accessible from PC SDK	DSE9020
32 bit group signals is now supported by the PC SDK.	DSE9386
Memory leak	DSE9493

# ABB AB Robotics

Copies to	Title <b>Release Notes PC SDK 5.13</b>	Ref RP5.13 - xxx	
Department DMRO/ST	Date 2010-04-07	Filename Release Notes PC SDK 5.13.doc	Page 13/13
	Dealt with by, telephone Anders Trillkott, +46 21 344863		

## *Known Limitations*

- In manual mode, when a PC SDK application releases master and immediately requests master again, the FlexPendant is locked up when the user presses the prompt to grant write access to the PC SDK application. In automatic mode there is no problem.
- PC SDK IPC Messaging - the PC SDK erratically ALWAYS sends 444 byte. Data must therefore be null terminated. Also, there is a problem reading the data if the PC SDK thread is running as STA. Changing it to MTA will solve that problem.
- ArrayData.FillFromString is not correctly implemented in the PC SDK. Users are recommended to use RapidData.StringValue to read and write small arrays (e.g. 100 num variables) in one call.
- Due to a Microsoft bug the ImageList.Dispose method does not work, causing a memory leak. Microsoft's advice is to set the ImageList to null instead of calling Dispose. This way the memory will correctly be reclaimed by the garbage collector.

**avoid:**

```
imageList.Dispose();
```

**use:**

```
imageList = null;
```

- In PC SDK the property IsLocal on the class RapidData, returns true for shared data, even though such data is visible from all modules.

The declarations of default zonedata and speeddata has been moved from module BASE.SYS in each task, and are now installed as shared data.

If an application used to create a RapidData object representing for example "v1000", the IsLocal property returned false in RobotWare 5.12, but will return true in RobotWare 5.13."

The new GUI controls introduced in FP SDK 5.12 are not documented in User's Guide.

## *Product Support*

If you need help or advice while using the product please visit the User Forum referred to in the introduction. For product support please turn to your local ABB office.

