SICE 2011 RT-Middleware Tutorial

Date: 2011/9/13 10:00~16:30 Place: SICE 2011



RT-Middleware tutorial



10.00	Part 1: Introducing RT-Middleware
10:00 -	Tetsuo Kotoku (AIST)
	An introduction to RT-Middleware, RT-Systems and RT-Components.
11.00	Part 2: Building RT-Systems using RT-Middleware
11:00 -	Geoffrey Biggs (AIST)
	Hands-on practice using small samples to construct complete RT-Systems.
	Part 3: Creating RT-Components
13:30 -	Geoffrey Biggs (AIST)
10.00	Hands-on practice creating RT-Components.
	Part 4: Human interaction with OpenHRI
15:15 -	Yosuke Matsusaka (AIST)
10.00	A demonstration of RT-Components for human-robot interaction.
	Part 5: Discussion
16:00 -	
16:30	



Part 3: Creating RT-Components

Geoffrey Biggs (AIST)

2011.7.25 青梅商工会議所主催 RTM講習会

RTCBuilder



RTCBuilder outline



What is RTCBuilder?

- A tool for specifying a component profile and generating template source code.
- Support for generating templates in new languages can be added via plugins.
 - ≻ C++
 - > Java
 - Python
- C ++ code generation is included by default.
 Other languages are offered as plugins.

€RTC Builder - Flip/RTC.xml -	Eclipse SDK		_O X
ファイル(E) 編集(E) ナビゲート(N) 検	漆(A) プロジェクト(P) 実行(P) ウィンドウ(W) ヘルブ(H)		
			RTC Builder 🔤 🎽
■ パッケージ・エクスプロ 🛛 🔍 🗖 🗖	> Flip 🛛		
E 🖏 🎽	基本		
RTC.xml		► E>F	
E E Fio	 RT-Component Basic Profile ○RT-Component Basic Profile ○RT-Component Basic Profile ○RT-Component Setting Settin	 セント モジュール名: モジュール須要: パージョン: ペンダ名: モジュールカテゴリ: コンポーネント型: アクライビティ型: コンポーネント経知: 東クテ型: コンポーネント経知: 環たうご: 東(市日期): 概要: RTC Type: コード生成: パッケージル: インボート: パッケージング: 	ドロンボーネントを転影するため前を推定します。 しなかれてユンボーネントのペーズインスタンならた使用なれます。 使用できるま大耳が別ったり、結束、パイス、レンダースコンのみです。 ドロンボーネンルが一切、検索に見ます。 やロンボーネンルが一切、検索に見ます。 やロンボーネンルが一切、検索に見ます。 やロンボーネンルを作用した作者を、ペンダをを指定します。 ふるしなアンドログロできます。 使用できるます。 ドロンボーネンルのたずから入りします。 使用できるます。 ドロンボーネンルのかざりを入りします。 使用できるます。 ドロンボーネンルのかざりを入りします。 使用できるます。 ドロンボーネンルのかざりを入りします。 使用できるます。 ドロンボーネンルのかざりを入りします。 になったいからずきます。 ドロンボーネンルのかざりを入りします。 ドロンボーネンルのかざりを入りします。 ドロンボーネンルのかざりを入りします。 ドロンボーネンルのがます。 ドロンボーネンルのかざりを入りします。 ドロンボーネンルのがます。 ドロンボーネントのかざりを入りします。 になったいかです。 ドロンボーネントのかざりを入りします。 ドロンボーネントのかざりを入ります。 ドロンボーネントのかざりを入ります。 ドロンボーネントのかが、加まれ、バイン、パンダースコルのみです。 ドロンボーストのからか、日本、バイン 「アレストーストののなり」のないから、 になったいからかられたいたいたいたいたいたいたいたいたいたいたいたいたいたいたいたいたいたいたい
	基本 アクティビティ データボート サービスボート コンフィギュレーション ドキュメント生成 音語・環境 RTG×ml		
	W BuildView 🛛		
	20 originallimage Fig	2	ticoedinage D
 ∫ ⊡ °			

Screen layout





2011.7.25 青梅商工会議所主催 RTM講習会



- Download USBCamera.zip from this URL http://www.openrtm.org/openrtm/en/content/ sice-2011-openrtm-aist-tutorial
- Extract the archive.
 ※ If extracted to a path with spaces in it, errors will occur when building with VC++.

Preparation



Change the perspective

(1) Click the "Change perspective" button in the top right, and select "Other"



②Select "RTCBuilder"

パースペクティブを聞く
 ○C/O++ ○C/Sリポジトリー・エクスプローラー ジ Java (デフォルト) ジ Java 会照 ○ Planning ● Pydev ○ RT System Editor ● RT C Builder ● チーム同期化 ジ チノルジリース
<u> </u>

Perspective
A tool in Eclipse.
Changes the menus, toolbars, editors, views, etc. to match the perspective's goals.

Project creation and starting the editor

1 Click the editor's button in the toolbar

ファイル(F) 編集(E) ソース(S) リファクタリング(T) ナビゲート(

🚝 RTC Builder – Eclipse SDK

📑 🗖 🔡

||温 パッケージ・エクスフロ|

From the File menu create a new project. In the New Project screen, select "Other" – "RTCBuilder" and click "Next"
 From the File menu, select "Open New Builder Editor"

🎄 • 🜔 • 💁 • 🛛 🗀 🔗 •

Open New RtcBuilder Editor

*To specify the project location, in step 2, uncheck the "Default location" checkbox and enter a path. This may be outside your workspace, but it will be treated as if it is inside the workspace.

Project name: USBCamera

2 Enter a project name



(3) The project is generated.

擋 パッケージ・エクスプロ 🔀



公



Import the profile



1Click "Import" in the "Basic" tab



②Select the XML file.



- Reusing existing RTCProfiles.
 - Existing RTCProfiles can be exported and imported.
 - Imported RTCProfiles can be used to generate code.
 - Files can be imported and exported in XML and YAML formats.

Code generation



- -

 ∇

Generate the code template





🖃 🥽 USBCamera 😑 🧀 cmake modules 📄 cmake_uninstall.cmake.in CPackWIXcmake FindOpenRTM.cmake 😑 🧀 cpack resources Description.txt License.rtf License.txt 🗎 wix.xsl.in CMakeLists.txt Doxyfile.in rtc.conf RTC.xml RTC.xml20110715204615 USBCamera.conf 1 🕶 USBCamera.cpp b USBCamera.h 🕶 USBCameraComp.cpp

🏥 パッケージ・エクスプローラ 🔀

The perspective changes after generation.



%If the code is not displayed, click "Refresh."

$C++ RTC \rightarrow CDT$
Java RTC \rightarrow JDT
Python RTC \rightarrow PyDev



Tab	Explanation
Basic profile	Enter the RTC's basic profile and information. Generate code, import/export profiles and manage packaging.
Activity profile	Select the activities the RTC will support.
Data port profile	Manage the RTC's data ports.
Service port profile	Manage the RTC's service ports and the service interfaces they support.
Configuration	Manage the configuration parameters and sets that can be edited by the RTC's users, and system configuration parameters.
Documentation	Edit the documentation to add to the generated code.
Language, environment	Select the language to generate, set the environment.
RTC.xml	Displays the current RTCProfile in XML format.

Basic profile



■ RTコンポーネントの名称など,基本的な情報を設定

⊁*USBCamera XX	5			- <i>a</i>	
基本					
➡ RT-Compone	nt Basic Profile	* ピント			
このヤクションではR	TTンポーネントの基本情報を指定します。	モジュール名:	RTコンポー2	ネントを識別する名前を指定します。	
*モジュール名:	USBCamera		この名称は、	ンボーネントのベースインスタンス名にも使用されます。 女字はアルファベット、数字、ハイアン、アンダースフアのみです。	
モジュール概要:	USB Camera component.	モジュール概要:	RTコンポース	シントが提供する機能の概要を入力します。	
*バージョン:	1.0.0	バージョン:	ASULL文子7 RTコンボース	い実用できます。 ネントのバージョンを指定します。	
*/心发名:	AIST	17 .B.A.	x.y.z(x,y,z(t	数字)の形式で入力してください。	
*モジュールカテゴリ	: ImageProcessing	×	ASCI文字/	いていていていていていていていていていていた。 が使用できます。	
コンポーネント型:	STATIC	モジュールカテゴリ:	RTコンポー 選択肢にた		
アクティビティ型:	PERIODIC	▲ コンボーネント型:	使用できる RTコンポー	Module name: US	BCamera
コンポーネント種類	: DataFlow FSM MultiMode	フカティビティ用す	·UNIQUE ·COMMU	Description: Optic	nal (USB Camera component)
最大インスタンス数	tā <mark>1</mark>		PERIOD		
実行型:	PeriodicExecutionContext	*	·EVENTE	Version 100	
実行周期:	1000.0	コンボーネント種類	・DataFlox		
概要:	USB camera component using OpenCV	最大インスタンス教	・FiniteSta ・MultiMod ・生成可能	Vendor: Optional	(AIST)
DTO T		■ 実行型: 実行型:	実行型を	Catagory: Ontion	(ImageDrassoning)
RIC Type:		Set Industri	この設定値		al (Illiager I ocessing)
▼ コード生成とパー	ッケージ化	概要: BTC Type:	RTコンボー 特定機能	Assessment type	
コードの生成および	パッケージ化を行います。 - ケージ化		值が省略	Component type.	21A110
▼ プロファイル情報		コード生成: パッケージ化:	設定した情 RTCのソー	Activity type: PF	RIODIC
プロファイル情報の	インボートおよびエクスボートを行います。	(s. 16 - 1			
インポート エク	スポート	インホート: パッケージング:	RtcProfile 設定した情	Component categ	orv' DataFlow
		1000000			
				Maximum instanc	
基本 アクティビティーテ	「-タボ-ト サ-ビスポート コンフィギュレーション ドキュメント生成 言語・環境	RTC.xml	_	Execution type: D	ariadiaEvacutianContaxt
				Execution rate: 1	000 0

Xalues in red must be provided.Explanations are given on the right.

Activity profile



The activities to be implemented in the RTC.

アクティ	ィビティ				1 (1)Select the activity to
▼ ₱₱₱₸₮₭₸₮	0				
このセクションで	は使用するアクションコールバックを指定します。		onInitialize	初期化処理です。コンポーネントライフサイクル開始時に一度だけ呼ばれます。常に有効。	edit.
	コンポーネントの初期化と終了処理に関する	もアクション	on Finalize	終了処理です。コンボーネントライフサイクルの終了時に1度だけ呼ばれます。 EverytionContextがまたを問かするとき1度だけ呼ばれます。	
onInitialize	onFinalize		onShutdown	ExecutionContextが実行を停止するとき1度だけ呼ばれます。	
	実行コンテキストの起動と停止に関するア	79ション	onActivated	非アクティブ状態からアクティブ化されるとき1度だけ呼ばれます。	A ST A ST
onStartup	onShutdown		onDeactivater	」 アクティフ状態から非アクティフ化されるとき1度たけ呼ばれます。 FBBOR状態に入る前に1度だけ呼ばれます。	onActivated
	alive状態でのコンポーネントアクショ	ン	onError	ERROR状態にいる間周期的に呼ばれます。	onError りん
onActivated onError	onDeactivated	onAborting	onReset	ERROR状態からリセットされ非アクティブ状態に移行するときに1度だけ呼ばれます。	
onenor	Dataflow型コンポーネントのアクショ	N/	onExecute onStateLIndat	アクテイフ状態時に周期的に判決す。 e onExecuteの後毎回呼ばれます。	
onExecute	onStateUpdate	onRateChanged	onRateChang	ed ExecutionContextのrateが変更されたとき呼ばれます。	Durn it on or off
	FSM型コンポーネントのアクション		onAction	対応する状態に応じた動作を実行するために呼ばれます。 	
onAction			UnmodeOnang	eu t - 1 m e e convententententen ver 20	
	Mode型コンポーネントのアクション		動作概要:	アクティビティの概要説明を記述します。	
onModeChang	ed		 事前条件: 事後条件・ 	アクティビティを実行する前に成立すべき事前条件を記述します。 アクティビティを実行した後に成立すべき事後条件を記述します。	
➡ Documenta	ation		фласжент	7 7 7 4 C 7 4 C 8 1 0 7 BUCKALL 7 1 C GFBCK1 1 CBL/C 00 7 6	
このセクションで 上崎のアクションで	は各アクションの概要を説明するドキュメントを記述します。 ^を選択すると、それぞれのドキュメントを記述できます。	•			
			1		
アクティビティ名	oninitialize	O ON O OFF			
	コンボーネント自身の各種初期化処理	A	1		
動作概要:					
			1		I on Activated
事前条件:			-		
			1		I onDeactivated
末 後を供	ロンボーネントの初期化処理が正常に完了している	<u> </u>	1		
争技采注		-	1		
	L		3		
基本 アカティドティ	- デーカポート サードフポート コンフィギュ レーション・ ドキ	ansh本式 言語・環境 PTC vari	Manning ID LISP (PrP RTC-CANopen	⊥
100 C C C C C C C C C C C C C C C C C C	היוןעבע שבהייעעבןיו מאס ען י מאי ען		mapping to 000 1		

The currently-selected activity is displayed in red.

Enabled activities are highlighted in blue.
All activities may have execution, pre- and post-condition documentation attached.

Data port profile



Data ports to add to the RTC.

- I BL-	/ Clari コンホーネント0	JDataPort(7 -9	ホートの情報を設定します。		テーダホート:	データを出力す
*ボート名	(InPort)	Add	*ボート名 (OutPort) image	Add	InPort:	InPortとOutPor BTコンポーネン
		Delete	111060	Delete	20.000	他のRTコンポー
		_			OutPort:	RTコンポーネン 他のRTコンポー
					ポート名:	データポートを讀
)etail						ホート名は、同一ポート名に対して
セクション	ではデータポート毎の根	腰を説明するド	キュメントを記述します。		デニカ刑・	ASCII文字が使 データポート関7
カデータボ	ートを選択すると、それ	それのドキュメント	が記述できます。) ~9至.	InPort OutPor
-1:32 : []]	mage (OutPort)					テーダ空はOper
データ型	RTC::CameraImage			~	変数名:	データポートに関 変数の名称は言
· 数名	image				ポートの場所:	RTSystemEdit
	RIGHT			~	ドキュメント:	データポートに関
示位置						全てを記述する
長示位置						
表示位置 locument	tation	raa data fuam ti		1000		
長示位置 locument 東朝日	tation Capture imag	ges data from th	ie camera	~		
表示位置 Pocument 既要説明	Capture image	ges data from tł	ie camera	<		
e示位置 Pocument 要説明 Fog型:	Capture image:	ges data from th Image	ie camera			

①Select "Add" next to the type of port to add.

	メホート/の) 情報を設定します。 	Add
	dp_name	Delete
2)Select	t the port's properties.	
2)Selec	t the port's properties.	
2)Select	t the port's properties.	
2) Select - Detail このセクショ 上のデータ ポート名:	t the port's properties. aンではデータボート毎の概要を説明するドキュメントを記述します。 ボートを選択すると、それぞれのドキュメントが記述できます。 image (OutPort)	
2)Select このセクショ 上のデータ ポート名: *デー短	t the port's properties.	

The available data types are specified in IDL files, which must be set in the settings screen.

The data types provided with OpenRTM-aist (in RTM_Root]rtm/idl) can be used by default.

*****Documentation can be added to the ports.



*Port information is displayed in the build view.

BuildView 🛛		- 8
	image D	
	1000	
	USBCamera	
	15	

OutPort
 Port name: image
 Data type: RTC::Cameralmage
 Variable name: image
 Display position: right

Service port profile



Service ports to add to the RTC.



- Service port interface settings
 - Specifying an IDL file displays the interfaces in that file.

This sample does not have any service ports.

Configuration profile



Manage the configuration parameters of the RTC.

ባ በተታታዩ አካምር	キロエコンポーセントのコンフィギュレーション・パラマークを指う	21.47	Config Param.	RT-11/dt-
がビジンヨンCC *名称 teviceNumbe	x	Add	Connig. Paranit.	コンフィギュ 再利用性 パラメーダ:
	a	Delete	パラメータ名:	コンフィギュ パラメータ4 名前にはア
			データ型:	コンフィギュ 基本型のf
			デフォルト値:	コンフィギュ RTコンポー 解釈不能
			変動を	ゴンフィギュ
Detail			S. 27.40.	実際の名称
Detail Dセクションで(ま各コンフィギュレーション・パラメータの詳細情報を指定しま	ŧす。	单位:	実際の名称 コンフィギュ
Detail のセクションで(ラメータ名: c	は各コンフィギュレーション・パラメータの詳細情報を指定しま Jevice Number	ŧす。	单位: 制約条件:	実際の名称 コンフィギュ コンフィギュ
Detail のセクションで(ラメータ名: C	は各コンフィギュレーション・パラメータの詳細情報を指定しま leviceNumber	ŧŧ.	单位: 制約条件:	実際の名称 コンフィギュ コンフィギュ ・指定なし ・100(即催
Detail のセクションで(ラメータ名: c データ型	は各コンフィギュレーション・パラメータの詳細情報を指定しま leviceNumber int	토코。 	单位: 制約条件:	実際の名称 コンフィギュ コンフィギュ ・指定なし ・100(即催 ・範囲指定 ・列挙型指
Detail のセクションで(ラメータ名: で データ型 デフォルト値	は各コンフィギュレーション・パラメータの詳細情報を指定しま leviceNumber int 10	tj.	单位: 制約条件:	実際の名名 コンフィギュ コンフィギュ ・100(即指定 100(加加 第一型型型 ・ 小 のの の の の の の 名 コンフィギュ
Detail のセクションで(ラメータ名: c データ型 デフォルト値 変数名:	は各コンフィギュレーション・パラメータの詳細情報を指定しま leviceNumber int 0	t. J .	单位: 制約条件: Widget:	実際の名 コンフィギュ コンフィギュ ・指の(囲掌) ・100(囲掌) 「 シンフィギュし ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(囲掌) 「 シンフィギュ ・100(コンフィギュ ・100(コン マン マン コン コン コン コン コン コン コン コン コン コン コン コン コン
Detail のセクションで(ラメータ名: c データ型 デフォルト値 変数名: 単位:	は各コンフィギュレーション・パラメータの詳細情報を指定し。 leviceNumber nt 0	t.j.	单位: 制約条件: Widget: Step:	実際の名 コンフィイなし の イギュ い 100(加 新 ア リンフィイな い 間 で リ の (指 定 リ ア フィイな た の (指 定 り の (イギュ い し の (指 定 り の (イギュ い) の (イギュ い) の (イギ つ 、)) の (イギ に)) の (イギ))) の (イギ))) の ())))) ()))) ())))
Detail のセクションで(ラメータ名: [c データ型 ドデフォルト値 変数名: 単位: 目160条(4)	は各コンフィギュレーション・パラメータの詳細情報を指定し。 leviceNumber nt 0	t. J	单位: 制約条件: Widget: Step:	実際の名 コンフィギュ コンフィギュ ・100(即走 利利 列シシィギュ ・100(加 加 学 和 ジンフィギュ ・100 (加 ティギュ ・ 100 (加 ティ ギュ ・ コンフィ ギュ ・ 100 (加 ティ ギュ ・ 100 (加 ティ ギュ ・ 100 (加 ティ ギュ ・ コンフィ ギュ ・ 100 (間 一 初 フィ キュ ・ 日 ンフィ ・ ギョ ・ の の ・ の の の の の の の の の の の の の う の の の の
Detail のセクションで(ラメータ名: データ型 デフォルト値 変数名: 単位: 制約条件:	は各コンフィギュレーション・パラメータの詳細情報を指定し。 leviceNumber int 0	t, J.	单位: 制約条件: Widget: Step:	実際の名 コンフィギュ コンフィギュ ・100(1) 1
Detail のセクションで(ラメータ名: で データ型 ドデフォルト値 変数名: 単位: 制約条件: Midget:	は各コンフィギュレーション・パラメータの詳細情報を指定し。 leviceNumber int 0 し し text	t	单位: 制約条件: Widget: Step:	実際の名 コンフィギュ コンフィギなし 100(指生 100(指生 100(指生 100) 100(指生 100) 100(指生 100) 100(100(

①Click the **"**Add" button and enter the parameter information.

	ບັບຫາເຊຍາລະເຫຼົາສາສາແອເອົາມະຄານ ການສະ_ວ່າ, ພາການການສະຫຼຸມ	このな地空します
JUE993JUAP		-3411E04.9%
<u>*名称</u>		6A]
cont_nameU	N	
	NE	Dele

2)Enter the variable name, etc.

```
Name: deviceNumber
Data type: int
Default value: O
Variable: deviceNumber
Constraints:
Widget: text
```

※Available data types include short, int, long, float, double, and string. Direct entry is also possible.

*Constraints and widget information are used by RTSystemEditor.

Setting constraints and widgets



- About constraints
 - Can be set on data ports and configuration parameters
 - Checking must be performed by the developer.
 - \succ Constraints do not mean the middleware enforces them.
- Entering constraints
 - No constraints: Blank
 - Direct: Use the value as-is
 - ≻ e.g. 100
 - Range:<, >, <=, >=
 e.g. 0<=x<=100
 - Enumeration: (値1, 値2, ・・・)
 e.g. (val0, val1, val2)
 - Array: Value 1, Value 2, •••
 - ≻ e.g. val0, val1, val2
 - Key-value: { Key 0:Value 0, Key 1:Value 1,
 ... }
 - ▷ e.g. { key0:val0, key1:val1}

- Widget
 - text (Default)
 - slider
 - For numerical values with a range
 - > Set the step size with "step"
 - spinbox
 - For numerical values with a range
 - Set the step size with "step"
 - radio button
 - For enumerations
 When the widget and constraints do not match, text is used.

Language and environment



Set the language to generate and environment settings

而 · 垛児		
		▼ ビント
のセクションでは使用する言語を指定します ● C++ ● Python ● Java ● Ruby	Use old build environment.	言語: RTコンボーネントを作成する言語を選択します。リスト中の言語から選択可能です。 環境: 言語ごとのライブラリの依存関係や、使用するOSなどの環境を選択します。 詳細情報で設定した内容(OS情報、ライブラリ情報など)は、プロファイル内にのみ保存されます。
環境		
のセクションでは依存するライブラリや使用するOSなどを指定します		
Version OS	Add	
		このチェックボックスをONにすると、 旧バージョンと同様なコード(Cmake を利用しない形式)を生成
i羊糸田竹春幸候		
OS Version Add CPU Delete	Add	



Generating the RT-Component



Implementing an RT Component



- RTC states
 - Created
 - Alive
 - Inactive
 - Active
 - Error
 - Finalised



RTC lifecycle (UML state chart)



Available call-back functions

関数名	概要
onInitialize	Called once when the lifecycle starts.
onActivated	Called once when the component is activated.
onDeactivated	Called once when the component is deactivated.
onExecute	Called regularly while the component is active.
onStateUpdate	Called after onExecute.
onAborting	Called once when changing to the Error state.
onError	Called regularly while in the Error state.
onReset	Called once when leaving the Error state.
onShutdown	Called once when the EC shuts down.
onStartup	Called once when the EC starts.
onFinalize	Called once when the lifecycle ends.

Implementing the component





Implementing the component



To make this an RTC:

```
RTC::ReturnCode_t USBCamera::onInitialize() {
    // Set OutPort buffer
    addOutPort("image", m_imageOut);
    // Bind variables and configuration variable
    bindParameter("deviceNumber", m_deviceNumber, "0");
    //Create an instance of the camera class
    cam = new ds_Camera();
    return RTC::RTC_OK;
}
```

```
RTC::ReturnCode_t USBCamera::onFinalize() {
    // Delete the camera object
    delete cam;
    return RTC::RTC_OK;
}
```



To make this an RTC:

```
RTC::ReturnCode_t
USBCamera::onActivated(RTC::UniqueId ec_id) {
    // Initialise the camera object
    if(cam->initialize())
        return RTC::RTC_OK;
    return RTC::RTC_ERROR;
}
```

```
RTC::ReturnCode_t
USBCamera::onDeactivated(RTC::UniqueId ec_id) {
    // Shut down the camera object
    cam->finalize();
    return RTC::RTC_OK;
}
```

Implementing the component



```
RTC::ReturnCode_t USBCamera::onExecute(RTC::UniqueId ec_id) {
  // Capture an image from the camera
  if (cam->capture() < 0)
    return RTC::RTC_OK;
 // Get the image size
  int len = cam->getImageSize();
  CvSize size = cam->getSize();
  // Set up the output data
  m image.pixels.length(len);
  m_image.width = size.width;
  m_image.height = size.height;
  // Add the image data to the output
  memcpy((void *)&(m_image.pixels[0]), cam->getImageData(), len);
  // Write the output
  m_imageOut.write();
  return RTC::RTC_OK;
```

Compiling (Windows, CMake)



1Start the CMake GUI and specify the source and binary directories.

🗛 C Make 2.8.5 – C:/w	ork/workspace200/USBCamera	a/build 📃	
File Tools Options He	elp		
Where is the source code:	C:/work/workspace200/USBCamera	Browse S	ource
Where to build the binaries:	C:/work/workspace200/USBCamera/b	build 🛛 🖌 Browse	<u>B</u> uild
Search:	Grouped Advanced	Add Entry	Entry
Name			
<			>
Press Configure to update	and display new values in red, then pre: files.	ss Generate to generate selecte	d build
Configure Generate	Current Generator: None		
			1

②Press "Configure" and select the platform to build for.

🔺 C Make 2.8.5 - C:/w	ork/workspace200/USBCamera/build 📃 🗖 🔀
File Tools Options H	alp
Where is the source code:	C:/work/workspace200/USBCamera Browse Source
Where to build the binaries:	C:/work/workspace200/USBCamera/build Browse Build
Search:	c make-sui
Name	
	isual Studio 9 2008
Press Configure to up	Use default native compilers
Configure Gen O	Specify native compilers
0	Specify toolchain file for cross-compiling
0	Specify options for cross-compiling
	< Back Einish Cancel
L	

Specify separate directories for the binary and source directories.

Compiling (Windows, CMake)



3Click "Generate" once configuration has completed.





🦉 USBCamera - Visual C++ 2008 Express Edition

(4)Open the solution file in the binary directory.





Compiling (Windows, CMake)



5 Build the solution



6 In the "Debug" directory of the binary directory, start the component.

	USBCaneratep E-ports Lorary File 1.88 USBCaneraik Inconsectal Linker File	USBCameradil	m. (almost Kabab
	1,161 KB	Utbcaneraidb VC++ Minimum Rebuild Depende, 347 KB	
	Witcamerapdb Program Debug Database 2,789 KB	USBCameralb Object Frie Library 2.00	
	ubcaneraconp.kb UC++ Marinum Rebuild Depende. 347 KB wibcaneraconp.pb Program Debue Database 2264 Hz	USBCameraCompaxe	 build マイドキュシト 井和ドキュシト マイエビューラ マイネットワーク
			IT is (*)
		52	Debug ファイル フォルダ 夏新日時: 2011年7月16日、2129
h	ANN Detroit, b		東新田中 2013年7月16日。2129

C	4.16	0-51-52	Q+C- ##	8 19		10 K	r RT System E.
Name S II Peposit - D	a +Syste	m Distram			÷ 0	TD/5-1	4+ 400 E
1 20201 • D querte sat-u dust at • USBCommilte: USBCommilte:	Conte					#DUSY- # ## # ## ## ######	fit 1270.011/operation- USDC-average USDC-average USDC-average USDC-average Not Not Not Not Not Not Not Not
	Compone	othine USDC	ane ContinuationSet deta	Special of the suscent of			
	active	confie	name	Value			
	e	detault	deviceNumber	0			

Check operation



- 1. Start the CameraViewer
 - [Start menu]→[All programs]→[OpenRTM-aist
 - $1.1] \rightarrow [C++] \rightarrow [components]$
 - \rightarrow [opencv_rtcs] \rightarrow [CameraViewerComp.exe]
- 2. Connect the components.

Drag and drop the components into the system editor.

KT System Editor RCP				RT System Editor RCP			
<u>File Window H</u> elp				<u>File Window H</u> elp			
: 👯 📴 💓 👹 🍪 🎲				i 👧 📴 i 📂 🗰 🍪 🔐			
🖋 Name Servic_ 🕥 RepositoryVi 😑 🗖	🔐 *System Diagram 🔀 🗖 🗖	□ プロパティー	~ - 8	🗯 Name Servic 🕥 RepositoryVi 🖓 🗆	📆 *System Diagram 🔀 🗖 🗖	🔲 วือパティー	~ - 0
🏠 🗘 🗢 🛸 🗸 🗸		プロパティー	値 🛆	🏠 🗘 🤤 🗸 🗳 🗯 🎽		プロパティー	値
		🖃 📴 CameraViewer0		G T localhost		🖃 边 USBCamera0	
ia _ [] opaist-sk host_cxt	· · · · · · · · · · · · · · · · · · ·	Path URI	localhos	ia 🚺 oper 🛛 ist-sk host_cxt		Path URI	localhost/o
		Instance Name	Camera			Instance Name	USBCamer-
		Type Name	Camera	USBCameraU[rtc		Type Name	USBCamer-
		Description	USB Ca		USBCafferat	Description	USB Came
		Version	1.0.0			Version	1.0.0
		Vendor	AIST			Vendor	AIST
	Comerci/iemerl	Category	example		Cameral/iewerfl	Category	ImageProce
	Calificational	State	INACTI		Contractionard	State	INACTIVE
							_
		owned				i owned	
		🖃 🦓 ExecutionCo	or 📃			🖃 💞 ExecutionCo	r
		ID	0			ID	0
		State	RUNNIN			State	RUNNING
		Kind	PERIOD			Kind	PERIODIC
	Conf. 27 Mars. 27 Con. 27 Fund. 27 DT I. C. D	Rate	1000.0		Caufi 27 Mana 27 Cam 27 Funa 27 DT I -	Rate	1000.0
		🖃 🐏 InPort				🖃 🍢 OutPort	
	ComponentNa ConfigurationSet: default	Name	.in		ComponentNa ConfigurationSet: default	Name	USBCamer-
		Data Type	IDL:RTC			Data Type	IDL:RTC/C.
	active c name \` 適用	Interface Ty	p corba_c(active c name ' 通用	Interface Ty	p corba_cdr
	C d. image_height 2.	Dataflow Ty	p pullpusł		C d. deviceNumber U	Dataflow Ty	p pull,push
	Thindge_widen S Fryten	Subscription	i Any		キャンセル	Subscription	1 flush,new,pe
		UutPort					
		Name	.Key out				
		Data Type	IDE:BTC		「複製」「追加」「削除」	<	>
Pos: (206,56) Size: (93,99)				Pos: (35,57) Size: (93,39)			

Check operation



3. Connect the ports.



5. Activate the components.



4. Use the default profile.



6. The camera image will be displayed.



Document generation (Windows, CMake)



The doc/html/ directory in the binary directory contains Doxygen-generated documentation.



Add some more components



- 1. Start the Flip component
 - [Start menu]→[All programs]→[OpenRTM-aist
 - $1.1] \rightarrow [C++] \rightarrow [components]$
 - \rightarrow [opencv_rtcs] \rightarrow [FlipComp.exe]
- 2. Add the Flip component to the system editor.
- 3. Disconnect the USBCamera and CameraViewer components.
 - (1) Select the connection line.
 - (2) Right-click on it.
 - (3) Click "Delete."



Add some more components



- 1. Connect USBCamera's OutPort to Flip's InPort.
- 2. Connect Flip's OutPort to CameraViewer's InPort.
- 3. Activate the Flip component.
 - (1) Right-click on Flip.
 - (2) Select "Activate" from the context menu.

RT System Editor RGP					RT System Editor RGP					RT System Editor RCP				
Elle Window Help 62 de : 19 🕊 88 39					Elle Window Help 6권 8년 : 1월 🕊 6명 5명					Elle Window Help		9.45g		
Kone Gener. () Resultavity. ()	USDCaneval	CaneraViewer0	■ 70/54- 70/74- System Disgram System Disgram System Disgram System Disgram Kind Create Date Update Date Composite	ONLINE None	Nene Servic. (1) Repository (1) (1) (1) (1) (1) (1) (1) (1) (1) (1)	General Careers Career	CameraViewed	TOUTS TOUTS R RS System Disgram System D Kind Create Date Update Date Update Date Update Date Composite	I CONLINE ONLINE None	A Name Service () Reconstryvit. () A Constraint () A Constraint () A Constraint () Constraint ()	System Diagram 33	イロロック (1) イロロック (1) 「日本市内 (2) Carlo Carlo Stor (2) Stor (2) Stor (2) Nature - 2-> (1) ドロック (2) ドロック (2) ドロック (2) 「オークントの(7) (2) 「オークントの(7) (2) (2) (2) (2) (2) (2) (2) (2	20/57 70/57	In the second se
	Confi. ZT Mana. ZT Com.	Exec. RT RT L. 90				Confi. ZT Mana. ZT Com	ET Exec. ZT RT L. 90				Confi. IT Mana. IT C	X Delete	Rate	1000.0
	ComponentNi ConfigurationSet	1 1 1 1 1 1 1 1 1 1				ComponentN/ ConfigurationSet					ComponentN: Configuration ective c name (c d, flp_mode	Al Activate (2) Al Activate (2) Al Deactivate (2) Al Start (8) Al Start (8)	Name Data Type Interface Ty Datation Ty Subscriptor	priginal IDLRTC prostba.cr prostba.cr prostba.cr
	atai (ista)	FTR	¢			1810	NTR .	¢		Pos: (106:15) Sam: (93:39)	77.20	Open Save As Open and Restore Open and Quick Restore	properties OutPort Name Data Type	fiped in TDI RTC M 3

Distributable package generation



Build the "PACKAGE" target in the solution.



An MSI installer is generated in the binary directory.

 The component is installed into: C:/Program Files¥OpenRTM-aist/1.1/components/<Language>/<Package name>

Supplement



Compiling on Windows without CMake



(1) Execute "copyprobs.bat" in the generated code directory to copy the properties files.



③In the USBCameraComp/Debug directory, execute USBCameraComp.exe

3 展5 • 6 • 윩 💭 秋田	20113		
U.R.(D) CHwork Hwork space 200 HUS	BCamera¥USBCameraComp¥Debug		✓ ● 194
ファイルとフォルダのタスク 🙁	BuildLoghtm	mt.dep DEP ファイル	
 ■ このファイルの名前を変更する このファイルを移動する ■ このファイルを移動する ■ このファイルを出ビーする ■ このファイルを聞きる ■ このファイルを電子メールで送信 	USBCameracbi Object File 913 KB	USBCameroCompress	
× このファイルを発展する その他 冬	USBCameraComp.exe.inter mediat.	1 KB USBCameraComp.exp Exports Library File 1 KB	
 USBDameraComp マイドキュメント 共有ドキュメント マイゴンピュータ 	USBCameraComp.ik Incremental Linker File 1224 KB USBCameraComp.obj Object File	USBCameraComp.lib Object File Library 2 KB USBCameraComp.pb Program Debug Database	
Eta (*)	VC914 KB VC914 Minimum Rebuild Depende	2,731 KB Program Debug Database 1,724 KB	
USBCameraComp.exe アブリケーション 更新日時 2011年7月15日、2203 サイズ: 210 KB			

2Build with Visual Studio





2011.7.25 青梅商工会議所主催 RTM講習会

Document generation without CMake



①Start the Doxygen GUI tool. Set the directories and project name.



(3)In the "Output" section, select HTML.

ep 1 opecity the working directory to	om which doxypen will run
:/work./doxygen	Select.
ep 2 Configure doxygen using the Wa	zard and/or Expert tab, then switch to the Pain tab to generate the documentation
Weard Expert Run	
Project Mode Output Diagrams	Select the output formatic/ to generate Primer HTML Option HTML Option HTML Option primer
	prepare for compressed HTML (chm) [7] With search function [Change color.]
	LaTeX as intermediate formal for hyperinked PDF as intermediate formal for PDF as intermediate formal for PDF as intermediate formal for PDFScript
	(2) [Mon pages] C Roh Text Format (RTF) 2064.
	Previous

②Set the output and language settings in the "Mode" section.

S Doxygen GUI frontend •	
File Settings Help	
Step 1: Specify the working directory from which	doxypen will run
C:/work/doxygen	Select
Step 2: Configure doxygen using the Wizard and/ Wizard Expert Run	or Expert tab, then switch to the Run tab to generate the documentation
Topics Hode	Select the desired extraction mode O Documented extracts only
Digerama	All Entities Include cross-referenced source code in the output
	Select programming language to optimize the results for © [Optimize for C++ output]
	Optimize for Constant Solution Optimize for Constant Constant Optimize for Constant Constant Optimize for Constant Constant
	Optimize for Fortran output Optimize for VHDL output
	Previous Next

3Click "Run Doxygen" in the Run tab.

S Doxygen Gui frontend f	
File Settines Help	
Step 1: Specify the working directory from which doxygen will run	
C./work/doxygen	Selec
e Wizard and/or Expert tab. then smi	tch to the Pain tab to generate the documentation
Run dosyseen Status: not run ne	Show contiguration Sove los
Store HTML calmed	

Various settings



Specify data types available for DataPorts by specifying the directories containing the IDL files.



Only necessary when using your own IDL files. The OpenRTM-aist default data types are set automatically.

- Default data types are available at [RTM_Root]rtm/idl
- \rightarrow BasicDataType.idl, ExtendedDataTypes.idl, etc.
- →By default, [RTM_Root]=C:/Program Files/OpenRTM-aist/1.1/

SICE 2011 RT-Middleware Tutorial

