SICK AppSpace SDK 1.0.0 release note

Revision history

Status	RELEASED
Version	1.0
Last update	08 Nov 2022
Revision comments	

Summary

SICK AppSpace - SDK (free demo) is the Software Development Kit for the creation of application software for programmable sensors, released as an extension in the Visual Studio Code marketplace. This document pertains to the version 1.0.0, released on 08 Nov 2022 . It contains an overview of the important features and remaining known limitations and issues.

Table of contents

Highlights

Free usage

SICK AppSpace - SDK (free demo) is published in the VSCode Marketplace.

It is publicly available and usable for free. The user receives and agrees to a free demo license according to the general terms and condition. The user further agrees to send telemetry data according to https://www.sick.com/de/en/privacy-policy-data-processing-information-sick-appspace-tools/w/dataprotection-dataprocessing-information/.

Included in this release is a **getting started guide** which provides installation instructions and a **how-to build your first sensor app** (https://github.com/SICKAG/SICK-AppSpace-SDK-Docs). The user may download the SICK AppEngine(x64) for Windows or for Linux and use it for non-productive use in a demo mode, **SICK ID registration required**.

Features

Develop & Execute

LUA language support is added to the editor. This includes support for SensorApp specific language features and device and algorithm APIs. With the UI-Builder, user interfaces for SensorApps are created in a structured view, just by drag and drop of UI Elements.

Execute the SensorApp directly to a connected programmable device.

Building the source code as a SensorApp package, completes the development journey.

Device file system access

The device file system is integrated with VSCode to provide easy access to files on the device.

Device Console

The device console shows the output of deployed SensorApps in VSCode and thus provides immediate feedback.

Roadmap

Limitation	Description
Debugging	Debugging of SensorApps is planned to be released in a future version.
CLI	A Command Line Interface is planned to be released in a future version.
FlowEditor	The FlowEditor to define data flows is planned to be released in a future version.
Usability	Settings and detailed configurations are based on text file. Depending on user feedback, we may offer user interface for certain configurations.

Limitations

Limitation	Description
Protect SensorApp	Protection settings for your SensorApp to restrict reading the source code by others or duplicating the app from one sensor to another are currently not available. The source code of all sensor apps build with this version are readable by others.
Import app from device	Importing an installed SensorApp from device into your workspace is not available.

Known issues

General

Issue	Description	Workaround
Failed to rename folder on device	Renaming folders does not work.	Create a new folder and copy content.
Build watcher does not register changes	The watcher does NOT work for symlinked **files** it only works for directories.	If possible, include the parent directory as symlink.
Multiple build triggered	After removing a symlinked directory, the VSCode watcher still triggers the change event as if the symlink was never removed. This shouldn't have any bad side effects, except for performance (unnecessary rebuilds)	Restart VSCode

Language server

Issue	Description	Workaround
Custom path structure not loaded	Directories defined by *Lua.workspace.library* are not loaded.	A further *scripts* folder must be placed in which the Lua code must be placed.
Incorrect "missing parameter" warning	*table.insert* and *table.unpack* showing missing-parameter warning.	
Missing code action	SensorApp specific code actions are not shown if the last non empty line is a comment.	Use the action in another line.

UI-Builder

Issue	Description	Workaround
Missing parameter bindings	"Parameter Bindings" work only with parameters in the workspace. Parameters from the device are not loaded.	
Unexpected behavior on shortcut	Changed keyboard shortcuts are not respected by the UI-Builder.	If changed one should add *activeCustomEditorId != 'sickag.ui-builder-plugin.HtmlEditor'* to the *When* condition in Keyboard Shortcuts.
Unexpected behavior with edit menu	It is not recommended to use VSCode menu Edit > operations with UIBuilder as • Cut-copy-paste operates on VSCode's history and not the UIBuilder's. • Other operations should have no effect.	Use the built-in functionality of the UI-Builder.
No log output for broken MSDD builds	There is no error log in OUTPUT > SICK AppSpace Build, if your .msdd build breaks.	Open Help > Toggle Developer Tool to display UI logs.

Device connectivity

Issue	Description	Workaround
No reachable device selected	After removing and adding the same device again, the device model configuration shows "No reachable device selected"	Restarting VSCode fixes this (Wait 5 seconds to make sure the device connectivity process is stopped)
"Failed to activate extension: sickag.device-connectivity"	Possible reasons: 1. The device connectivity services uses fixed ports. When this error occurs it is most likely due to a blocked port blocked. 2. VSCode has been closed during the installation of the device connectivity services. The installation might be corrupt.	Possible solution: 1. Check the log of the device connectivity (see output channel) The blocked ports should be listed in the log. 2. Delete the following folder: • Windows: '%appdata%/SICK /AppSpace/MaxwellConnect' • Linux: '~/.config/SICK /AppSpace/MaxwellConnect'

Device / App Model

⚠️ Loading the manifest from a selected device is an experimental feature! We list the known issues here, but there may be more that have been undetected. We aim to completely rewrite this feature to improve performance and stability.

Issue	Description	Workaround
Missing app CROWN	Manifest of newly created apps is not updated.	Unselect and select the affected apps again to manually trigger the update.
Device manifest not loaded	Device must be in the same network and accessible from the current device.	• Ensure device is in the same network • Add device IP to NO_PROXY environment variable
Too many CROWNs displayed	If you provide Apps with CROWNs that don't start with the AppName, even though they are not selected.	Use a offline manifest.

Supported hardware & software

This release supports the following SICK AppSpace-enabled devices:

Product family	Part number	Device type
SICK AppEngine	1613796	SICK AppEngine (x64)
InspectorP61x	1116350	V2D611P-MMSCE4
	1114809	V2D611P-MMSBE4
InspectorP62x	1110847	V2D621P-2MSFBB5
	1110848	V2D621P-2MSFFB5
	1110849	V2D621P-2MDFGB5
InspectorP63x	1082298	V2D631P-2MXCXB0
	1082299	V2D631P-2MXSXB0
	1082300	V2D632P-2MXCXB0
	1082301	V2D632P-2MXSXB0
InspectorP64x	1082302	V2D642P-2MCXXA6
InspectorP65x	1082303	V2D652P-2MCXXA6
	1082304	V2D654P-2MCXXA6
	1082305	V2D652P-2MEWHA6
	1082306	V2D654P-2MEWHA6
MRS1000P	1104278	MRS1104P-111011
RFU61x	1091102	RFU610-10600
	1099890	RFU610-10601
	1101394	RFU610-10605
	1104441	RFU610-10614
	1104443	RFU610-10603
	1104444	RFU610-10604
	1104445	RFU610-10613
	1104446	RFU610-10610
	1104447	RFU610-10607
	1104448	RFU610-10618
	1104449	RFU610-10609
RFU62x	1062599	RFU620-10100
	1062600	RFU620-10400
	1062601	RFU620-10500
	1062602	RFU620-10101
	1062603	RFU620-10401
	1062604	RFU620-10501
	1068727	RFU620-10107
	1068728	RFU620-10105
	1069453	RFU620-10503

	1069677	RFU620-10104
	1070407	RFU620-10504
	1077860	RFU620-10505
	1077863	RFU620-10514
	1083557	RFU620-10510
	1083976	RFU620-10507
	1084997	RFU620-10111
	1086439	RFU620-10110
	1088871	RFU620-10508
	1091355	RFU620-10103
	1092037	RFU620-10112
	1094605	RFU620-10108
	1096414	RFU620-10114
	1101686	RFU620-10118
	1101700	RFU620-10102
RFU63x	1054396	RFU630-13100
	1054397	RFU630-13101
	1057943	RFU630-13105
	1058117	RFU630-04100
	1058775	RFU630-13102
	1059999	RFU630-04101
	1061498	RFU630-13107
	1067133	RFU630-13106
	1067473	RFU630-13103
	1068569	RFU630-04106
	1068726	RFU630-13104
	1070903	RFU630-13108
	1070904	RFU630-04108
	1073196	RFU630-04105
	1073376	RFU630-04102
	1073377	RFU630-04109
	1073442	RFU630-13110
	1074302	RFU630-13112
	1077861	RFU630-13113
	1077862	RFU630-13111
	1083558	RFU630-13115
	1087776	RFU630-04117
	1093152	RFU630-04104
	1095224	RFU630-13114
	1104670	RFU630-04103
RFU65x	1073556	RFU650-10100

	1076522	RFU650-10101
	1083559	RFU650-10105
	1083560	RFU650-10106
	1087587	RFU650-10102
	1092036	RFU650-10104
	1096413	RFU650-10103
SID	1098321	SID120
	1101360	SID70
SIM10xx	1097816	SIM1000-0P0B100
	1098146	SIM1012-0P0G200
	1111314	SIM1012-0P0G200S01
	1098148	SIM1004-0P0G311
SIM2x00	1080579	SIM2000-0A10A00
	1081902	SIM2000-2P04G10
	1092673	SIM2500-2P03G10
SIM10xx|Flexi Soft	1097817	SIM1000-0P0B110
SIM4x00	1078787	SIM4000-0P03G10
TIM8xxP	1090292	TIM881P-2100101
TriSpectorP1000	1091318	V3T11P-MR12A8
	1091319	V3T12P-MR32A8
	1091320	V3T13P-MR62A8
	1091321	V3T12P-MR32A7
	1091322	V3T11P-MR12A7
	1091323	V3T13P-MR62A7
Visionary-T AP	1102953	V3S140-2AAAAAA
	1102954	V3S140-2AABAAB
Visionary-S AP	1114319	V3S142-1AAAAAA
	1114320	V3S142-1AABAAB

Installation

Components

SICK AppSpace development tool	1.0.0
Code samples	See menu "Help > Samples" or https://gitlab.com/sick-appspace/samples
API reference	See API reference after installation of the emulator based on SICK AppEngine (x64) (note: <version> references the version of the installed emulator): %localappdata%\SICK\AppStudio\.SAEinstallations\<version>\docs
UI Builder	3.4.0, WYSIWYM and web programming tool for creating app UIs, based on DaVinci basic elements 5.6.0

System requirements

OS	Windows 10 x86_64 Windows 11 x86_64 Ubuntu 22.04 x86_64
HDD	min 1 GB
RAM	min 4 GB, 8 GB recommended
Processor	1 GHz CPU
Browser	Chrome for support of WebGL and WebSockets
IDE	Microsoft Visual Studio Code v1.71

Installation instructions & first steps

Please find installation instructions and first steps tutorials here: https://github.com/SICKAG/SICK-AppSpace-SDK-Docs