



ATM Software

XFS4IoT SP-Dev Workgroup

4th April 2023

- Recap from previous meeting
- What's new in the latest release
- Sigma presentation
- Next meeting

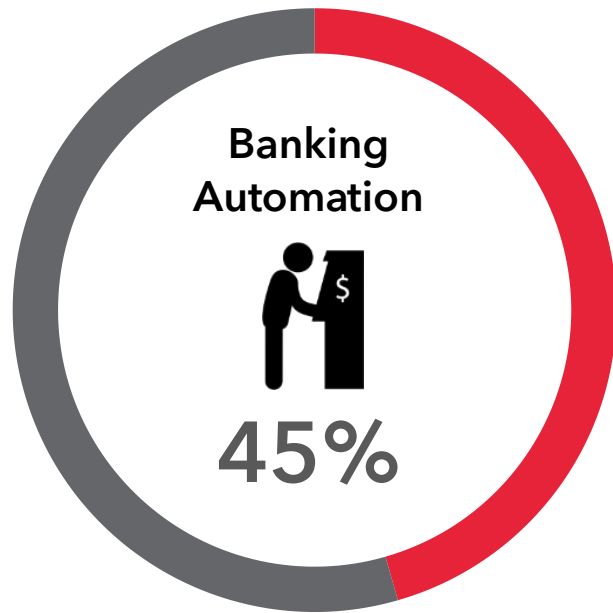
Recap from previous meeting

- Discussed XFS4 committee's progress
 - New mixed media class, moved "check scanning" out of printer class
 - Stricter definition of 'required' versus 'optional' parameters
 - Future releases will include more End-to-End security among other items
- Detailed discussion about End-to-End Security
 - Reviewed what is supported in the current spec and later this year
 - Brief discussion on 'post 2023' plans



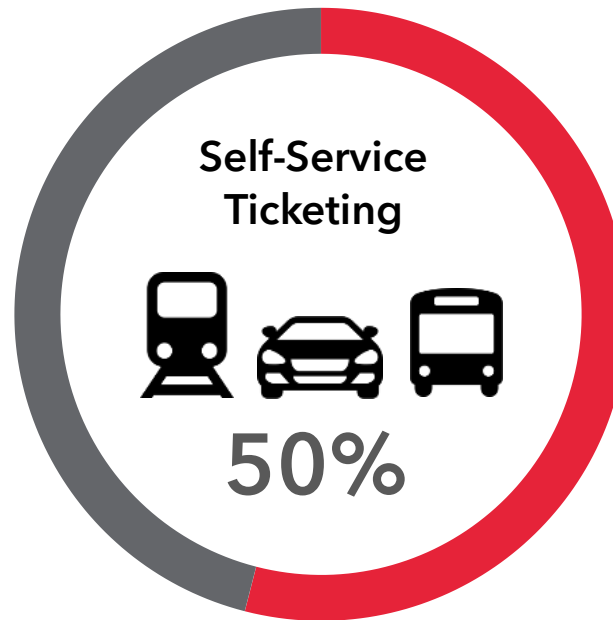
Our job, your solution

Business Area



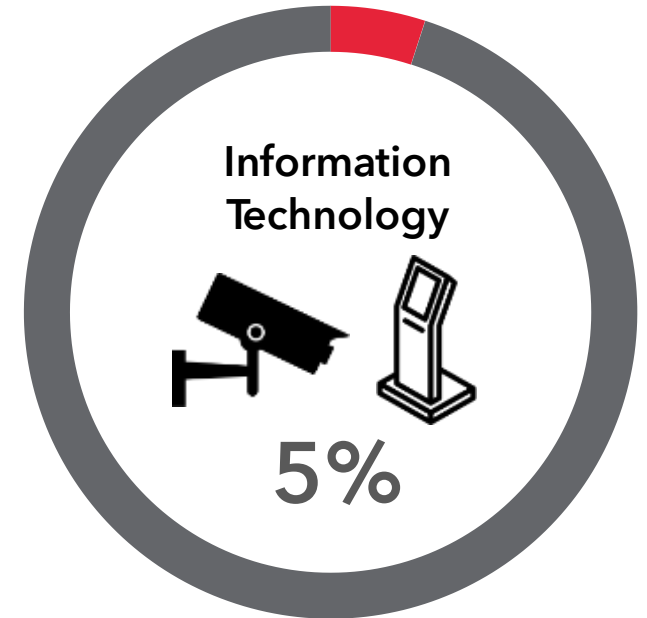
Bank Automation Machines

- Cash Dispensing
- Assisted Self Service
- Cash Recycling
- Teller Cash Recycler
- Banking Kiosk



Ticketing, Tolling and Parking

- Ticket Vending machines
- Self-service payment system
- Input-output parking devices
- OBU Dispensers

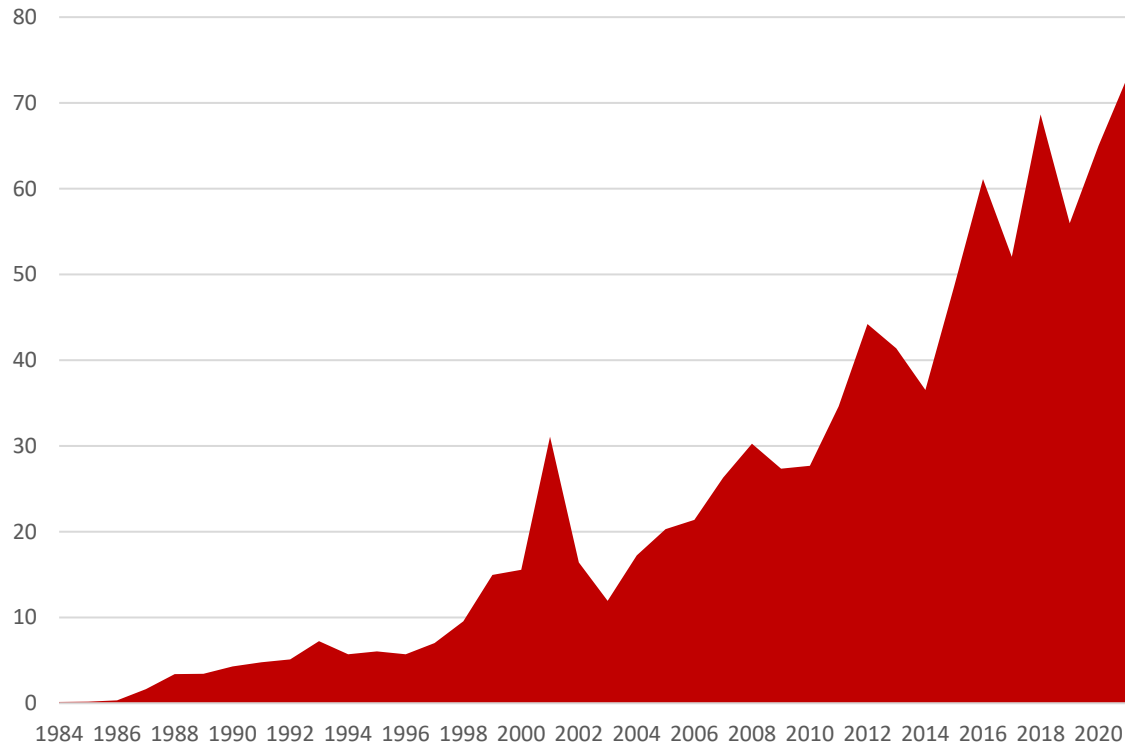


Security

- CCTV
- Access control systems
- Intelligent safe
- Fire detection
- Queue management system

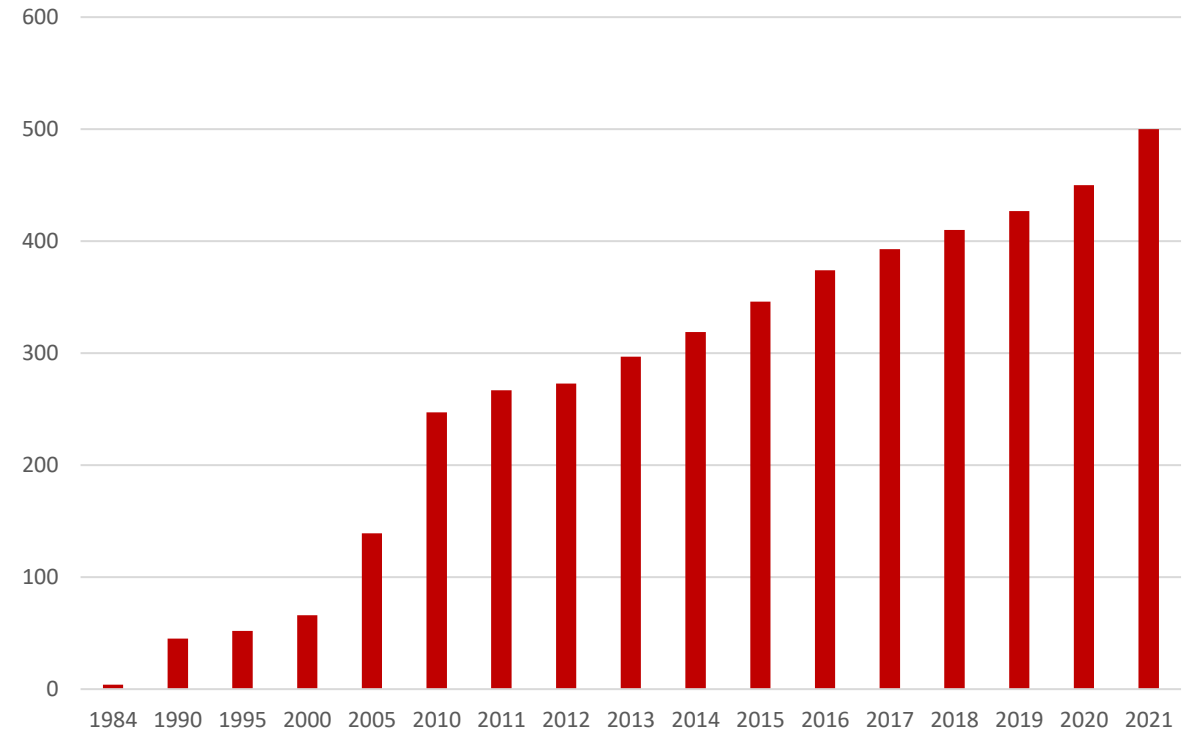
Corporate Growth

SALES €/Mnl



ANNUAL TURNOVER IN 2021: € 74.184.171

EMPLOYEES



EMPLOYEES 2021: 487

EMPLOYEES 2022: 517

Where We Are

Headquarters: Via dell'Industria 19, Monterubbiano (FM) - Italy



MONTERUBBIANO SITE

Building 1
Production Traffic

Building 2
Logistic

Building 3
R&D Laboratory

Building 4
Production Ticketing
Global Service Center

Building 5
Global Service Logistic

Building 6
Executive Offices
Administrative and Commercial Offices



OTHER SITE

COMUNANZA (AP)
Production Banking

MORESCO (FM)
Logistics Center

CONCORDIA S. Secchia (MO)
R&D Laboratory

DORZANO (BI)
Global Service Point

MILANO
Global Service Point

ROMA
Global Service Point



FOREIGN SUBSIDIARIES

SIGMA UK

Kemp House Suite 7000 152 - 160
City Road, LONDON (United Kingdom)



SIGMA Automatique

7 Avenue des Droits de l'Homme
45000, ORLEANS (France)



SIGMA Belgio

Rue Anatole France 115 - 121 - B 1030
BRUXELLES (Belgium)



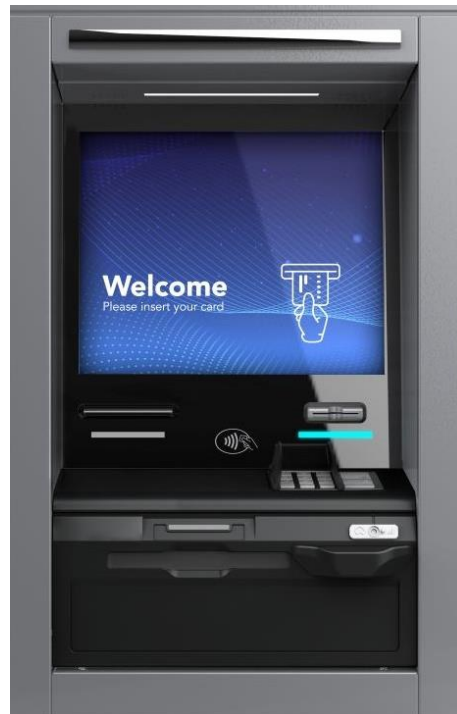
Banking

TCR



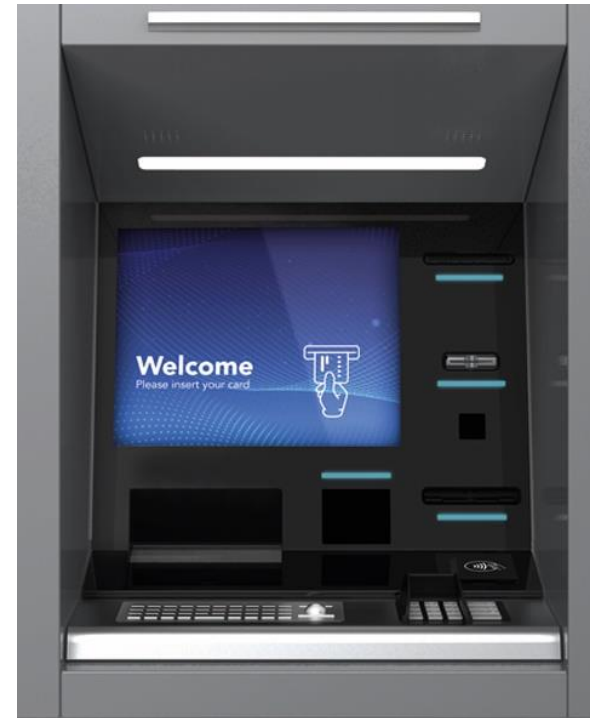
Z1HR

CASH DISPENSING



ST 400EVO

CASH RECYCLING



SD 960EVO

ASSISTED SELF-SERVICE



SR 2000

Banking

MULTI-FUNCTION KIOSKS AND QMS



CRONO



RTS 2020



KIS

Fare Collection Solutions

TICKETING AUTOMATION



ETS70



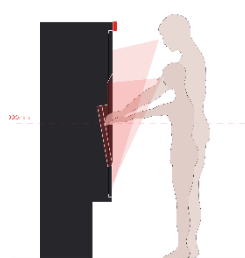
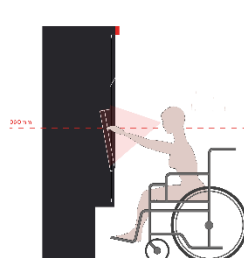
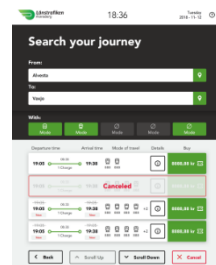
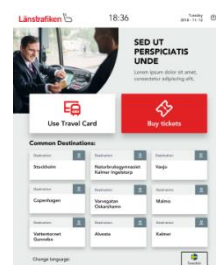
ETS90



KOS



KIS



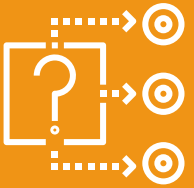


Demo XFS4IoT Printer Service

Created using KAL SP-Dev framework

Main Targets

- Implement an XFS4IoT Printer Service
- Run the Printer Service on Single Board Computer



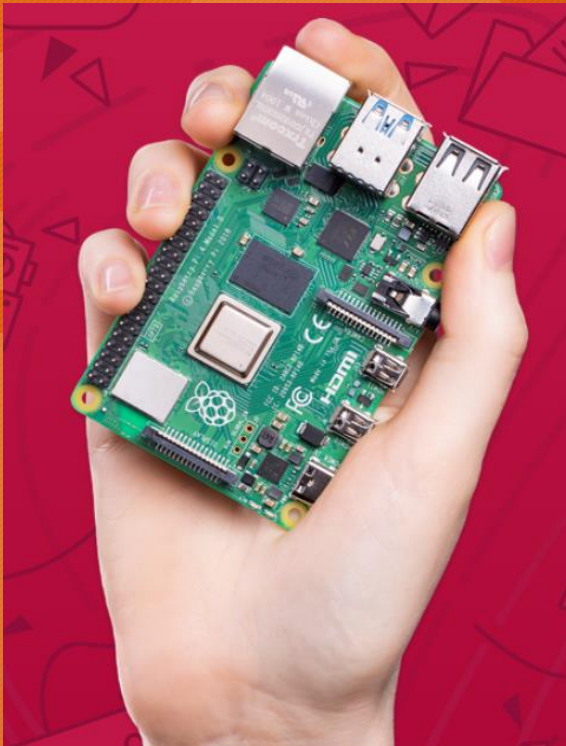
Single Board Computers (SBC)

- **Single board design:** SBCs are self-contained computer systems that are built on a single circuit board. This makes them highly compact and portable.
- **Low power consumption:** SBCs are designed to be highly energy-efficient, consuming much less power than traditional desktop or laptop computers.
- **Low cost:** SBCs are typically much less expensive than traditional computers, making them a cost-effective solution for many applications.
- **Variety of form factors:** SBCs come in a variety of form factors, including the popular Raspberry Pi form factor, as well as other form factors such as Nano-ITX and Pico-ITX.
- **Customizable:** SBCs are highly customizable and can be configured to suit a wide range of applications and use cases.
- **Versatile:** SBCs can be used for a wide range of applications, including embedded systems, robotics, IoT devices, media centers, and more.
- **Accessible:** SBCs are accessible to a wide range of users, from hobbyists and makers to professional developers and engineers.
- **Expandable:** Many SBCs have expansion options, such as GPIO pins, add-on boards, and other interfaces, allowing users to expand their capabilities as needed.

These are just some of the main characteristics of single board computers.

Raspberry Pi 4

(<https://www.raspberrypi.com/>)



- **Processor:** Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) at 1.5GHz
- **RAM:** 2GB, 4GB, or 8GB LPDDR4-3200 SDRAM (depending on the model)
- **Connectivity:** Gigabit Ethernet, dual-band 802.11ac wireless, Bluetooth 5.0, BLE
- **USB Ports:** 2 USB 3.0 ports and 2 USB 2.0 ports
- **Video Ports:** 2 micro-HDMI ports (up to 4Kp60 supported)
- **Audio Ports:** 3.5 mm audio port, stereo audio and digital surround 7.1 support
- **Mass storage:** microSD card slot
- **GPIO:** 40-pin GPIO header, compatible
- **Power:** USB-C power supply (5V, 3A), Power-over-Ethernet (PoE) support enabled via a PoE HAT (sold separately)
- **Dimensions:** 88 x 58 x 19.5 mm

These are just some of the main specifications of the Raspberry Pi 4. The board also has many other features and capabilities.

Enviroments

Development
Environment



Microsoft Windows 11
Microsoft Visual Studio 2022
KAL SP-Dev

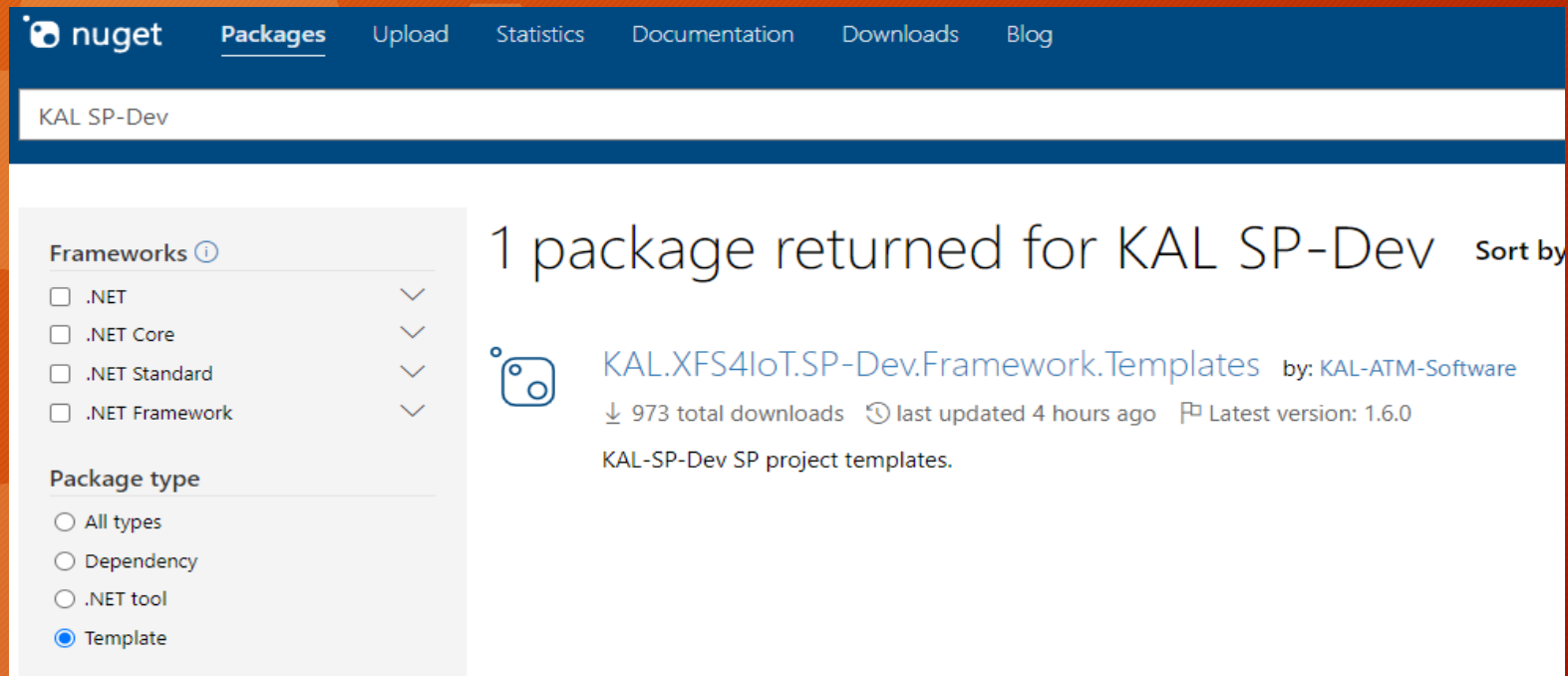
Test Environment



Raspberry Pi 4
Raspberry Pi OS
Kernel version: 5.15
Debian version: 11

SP-Dev Framework

SP-Dev Framework 1.6.0 available on nuget.org



The screenshot shows the NuGet.org website interface. At the top is a navigation bar with links: nuget, Packages, Upload, Statistics, Documentation, Downloads, and Blog. Below this is a search bar containing the text 'KAL SP-Dev'. On the left side, there are two filter sections. The 'Frameworks' section has four checkboxes: .NET, .NET Core, .NET Standard, and .NET Framework, each with a dropdown arrow. The 'Package type' section has four radio buttons: All types, Dependency, .NET tool, and Template (which is selected). The main content area displays the search results: '1 package returned for KAL SP-Dev'. Below this, there is a single package entry: 'KAL.XFS4IoT.SP-Dev.Framework.Templates' by 'KAL-ATM-Software'. The entry includes a download icon, the text '973 total downloads', a clock icon with 'last updated 4 hours ago', a document icon with 'Latest version: 1.6.0', and a description: 'KAL-SP-Dev SP project templates.'

nuget Packages Upload Statistics Documentation Downloads Blog

KAL SP-Dev


Frameworks ⓘ

- ☐ .NET
- ☐ .NET Core
- ☐ .NET Standard
- ☐ .NET Framework

Package type

- ☐ All types
- ☐ Dependency
- ☐ .NET tool
- ☒ Template

1 package returned for KAL SP-Dev Sort by

 [KAL.XFS4IoT.SP-Dev.Framework.Templates](#) by: KAL-ATM-Software

↓ 973 total downloads ⌚ last updated 4 hours ago 📄 Latest version: 1.6.0

KAL-SP-Dev SP project templates.

SP-Dev Framework

```
dotnet new install KAL.XFS4IoT.SP-Dev.Framework.Templates::1.6.0
```

```
dotnet new list
```

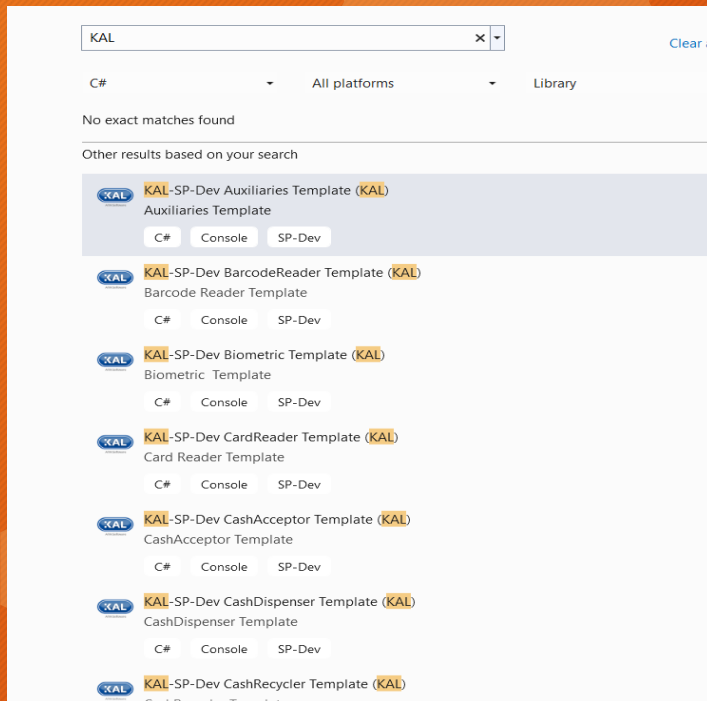
```
dotnet new list --tag SP-Dev
```

These templates matched your input: --tag='SP-Dev'

Template Name	Short Name	Language	Tags
KAL-SP-Dev Auxiliaries Template	spdev-Ax	[C#]	SP-Dev/Console
KAL-SP-Dev BarcodeReader Template	spdev-Bcr	[C#]	SP-Dev/Console
KAL-SP-Dev Biometric Template	spdev-Bio	[C#]	SP-Dev/Console
KAL-SP-Dev CardReader Template	spdev-Crd	[C#]	SP-Dev/Console
KAL-SP-Dev CashAcceptor Template	spdev-Caa	[C#]	SP-Dev/Console
KAL-SP-Dev CashDispenser Template	spdev-Cad	[C#]	SP-Dev/Console
KAL-SP-Dev CashRecycler Template	spdev-Car	[C#]	SP-Dev/Console
KAL-SP-Dev Lights Template	spdev-Lig	[C#]	SP-Dev/Console
KAL-SP-Dev PinPad Template	spdev-Pin	[C#]	SP-Dev/Console
KAL-SP-Dev Printer Template	spdev-Prn	[C#]	SP-Dev/Console
KAL-SP-Dev TextTerminal Template	spdev-Txt	[C#]	SP-Dev/Console
KAL-SP-Dev VendorApplication Template	spdev-Vda	[C#]	SP-Dev/Console
KAL-SP-Dev VendorMode Template	spdev-Vdm	[C#]	SP-Dev/Console

Create the SP Project

Choose the KAL-SP Dev Printer template



Configure your new project

KAL-SP-Dev Printer Template (KAL)

C#

Console

SP-Dev

Project name

DemoPrinterXfs4IoT

Location

C:\ProjectsGit\KAL_SpDev\Sigma\Sigma_SP_Dev

Solution name ⓘ

DemoPrinterXfs4IoT

☐ Place solution and project in the same directory

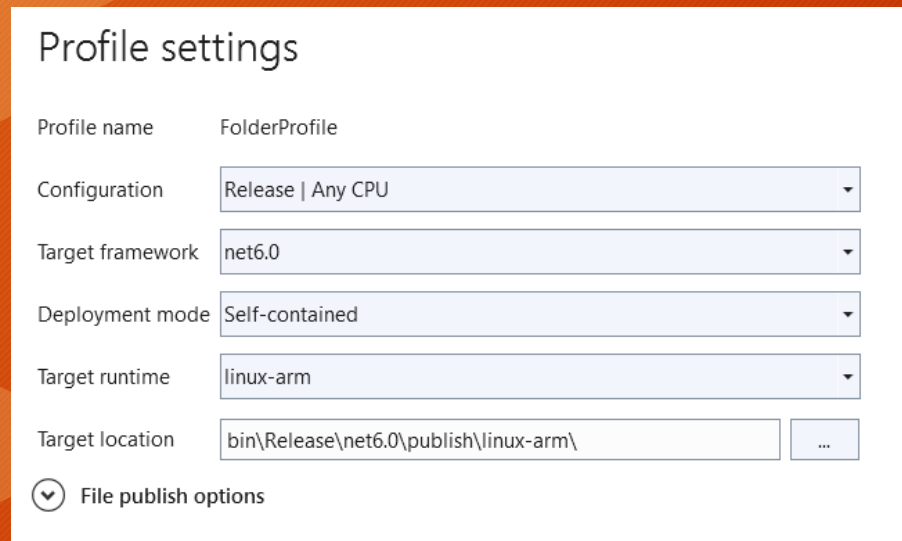
```
dotnet new spdev-Prn -n PrinterXfs4IoT
```


Methods and Properties Implemented

Features	Methods\Properties
Capabilities	PrinterCapabilities
Status	PrinterStatus
Reset	ResetDeviceAsync
Send Raw Data	RawPrintAsync
Print using forms	FormRules <ul style="list-style-type: none">• DotsPerInchTopX / DotsPerInchBottomX• DotsPerInchTopY / DotsPerInchBottomY• DotsPerMMTopX / DotsPerMMBottomX• DotsPerMMTopY / DotsPerMMBottomY• DotsPerRowTop / DotsPerRowBottom• DotsPerColumnTop / DotsPerColumnBottom PageSize MediaSpecs ExecutePrintTasksAsync
Present Ticket	ControlMediaAsync
Send Events	RunAsync

Publish the project as portable for linux-arm

the self-contained package for Linux ARM contain all the libraries necessary to run the application on the specific platform, without the need to separately install .NET

A screenshot of the 'Profile settings' dialog box in Visual Studio. The dialog has a white background and a title bar. It contains several settings for publishing a project. The 'Profile name' is 'FolderProfile'. The 'Configuration' is set to 'Release | Any CPU'. The 'Target framework' is 'net6.0'. The 'Deployment mode' is 'Self-contained'. The 'Target runtime' is 'linux-arm'. The 'Target location' is 'bin\Release\net6.0\publish\linux-arm\'. At the bottom, there is a section for 'File publish options' which is currently collapsed.

Profile name	FolderProfile
Configuration	Release Any CPU
Target framework	net6.0
Deployment mode	Self-contained
Target runtime	linux-arm
Target location	bin\Release\net6.0\publish\linux-arm\
File publish options	

```
dotnet publish -c Release -r linux-arm --self-contained true -p:PublishReadyToRun=true  
-p:PublishSingleFile=true -p:TargetFramework=net6.0
```

Publish the project as portable for linux-arm

The output folder contains:

NET Core runtime includes a set of libraries that implement the core functionalities of the framework

The SP-Dev Framework core assemblies

The custom assemblies and/or libraries implemented for driving the device

- libclrjit.so
- libcoreclr.so
- libcoreclrtraceptprovider.so
- libdbgshim.so
- libhostfxr.so
- libhostpolicy.so
- libmscordacore.so
- libmscordbi.so
- libSystem.Globalization.Native.

- System.Linq.dll
- System.Linq.Expressions.dll
- System.Linq.Parallel.dll
- System.Linq.Queryable.dll
- System.Memory.dll
- System.Net.dll
- System.Net.Http.dll
- System.Net.Http.Json.dll

XFS4IoT.SP.Framework.Common.dll

XFS4IoT.SP.Framework.Core.dll

XFS4IoT.SP.Framework.F

XFS4IoT.SP.Framework.S

XFS4IoT.SP.Framework.S

- Sigma.XFSIoT.Common.Communication.Usb.dll
- Sigma.XFSIoT.Common.Utility.Logger.dll
- Sigma.XFSIoT.Common.Utility.Thread.dll
- Sigma.XFSIoT.Device.Custom.Printer.dll

Sigma.XFSIoT.Device.Custom.Printer.dll

Running on Raspberry Pi

Copy the Publish folder in a Linux directory

Do runnable the published file (`chmod +x DemoPrinterXfs4IoT`)

Configure the ip address:

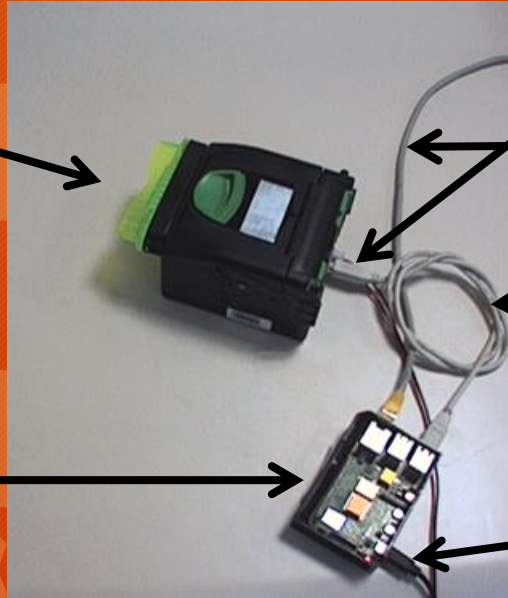
- “DemoPrinterXfs4IoT.dll.config” file
- linux firewall

Configure the runnable file on linux boot

Demo Video

Printer Device

Raspberry Pi 4



Device power cable

Network cable

USB cable

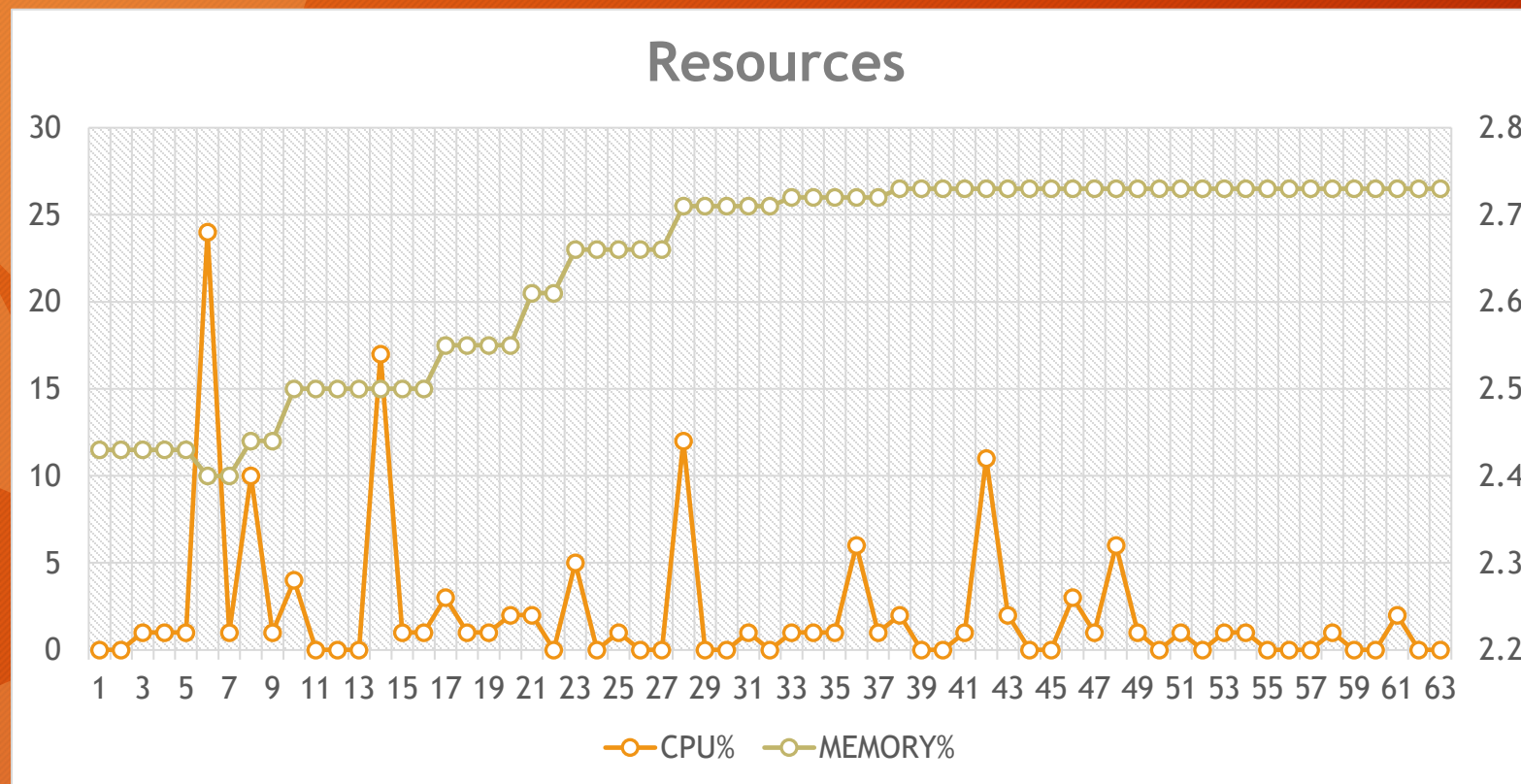
Raspberry power cable

Demo Video



External Video

Resources



Conclusion

Fast software and easy implementation of XFS4lot service provider

Platform portability (The self-contained .NET C# solution can be easily ported to different platforms)

Easy deployment

Scalability

Low hardware resource usage

Cost-effective

What's new in the latest release v1.6.2

- #9 - Fix valid fonts check and SendAsync()
— @RotoGiacomo
 - #10 - Some bugs in PrintForm handle
— @LinuxTvT
 - #11 - Fix bug: Command not dequeue to handle
— @LinuxTvT
- Multiple ways to contribute!
 - Raise pull requests with changes
 - Raise GitHub Issues with questions, bugs reports or suggestions

- Updated to ensure only a single message can be sent to the client at a time
- Reduced memory allocations when receiving messages from the client
- Updated Command Queue to always handle any queued commands

- Fix check for valid fonts in Print Form handler
- Save printer Form and Media information using persistent storage implementation
- Fix various issues with printer form field assignment
- Correct override for VendorMode CommandPostProcessing implementation

Zoom

- First Tuesday of each month at 1300 UK time

Next call: 2 May 2023 at 1300 UK

Calls are 30 mins long

We will continue to use Zoom

(Interpretation in Japanese, Chinese and Spanish is available using Zoom's interpretation feature)