A Fully Integrated Solution to Update Your IoT Devices
Why FullMetalUpdate?
A couple of hours later - Containers

Container
- Application
  - QT 5.8
  - Network
  - Touch Screen

Container
- MQTT
  - Network

yocto PROJECT
A couple of hours later - CGroups

Fine-grained control over allocating, prioritizing, denying and managing system resources
A couple of hours later - Security

- Secomp limit access to system calls
- Rootless containers
- Mandatory Access Control Policy
Early morning

**Power Failure:** Delta updates are based on Overlayfs, AuFS, ... which are not power failure resistant.

**Complexity:** Docker comes with multiple layers to manage containers life cycles, updates, ... which are not needed for embedded devices.
Late morning…

Does it exist???

resin.io → balena

Librsync  Git

docker
What do we need?

1. Runtime to run containers
2. Build system to create containers
3. Delta updates
4. Backend to manage the different updates
5. Client to download update
6. Tool to manage containers life cycle
Step 1 - Open Container Initiative

A Runtime Specification which outlines how to run a “filesystem bundle” that is unpacked on disk.

Multiple implementations:

RunC
Clear Containers
RailCar
Step 2 - Create containers

- Need a replacement for dockerfiles
- Should work for ARM, X86, ... architectures
- A tool to pull only needed dependencies
- Build system itself in a Docker container

Your Application

Minimum set of dependencies to run your application

Configuration of your container

Container App – V1

Optimize for your processor!
Step 3 – OsTree for delta updates

Used by:

- Git for binaries
- Fully atomic updates
- Only update delta
- Used to update containers and the OS
- Everything happens in the background
Step 3 - Delta updates and containers

- **OSTree server**
  - Delta Updates
  - Remote server

- **Local Server**
  - Build farm
  - Container
    - OS – V1
    - App – V1

- **OSTree client**
  - Delta Updates
  - Remote client

- **Embedded system**
  - OS – V1
  - Container
    - App – V1
Step 4 - Update server

- Hawkbit as a backend and frontend
- Fully integrated with the Yocto build system
Step 4 - Cloud agnostic

- Easy to setup with Docker Compose

Hawkbit
OsTree Server
Maria DB
Rabbit MQ
Step 5 & 6 - Client

- Based on Rauc Hawbit client
- Integration with OsTree
- SystemD to manage containers life cycles
Overview

Cloud

OSTree Server
- Hawkbit
- Maria DB
- Rabbit MQ

- FMU Client
- System D
- OS – V1
- Container
- App – V1

OSTree local Delta Updates

Build farm

Local Server

Google Cloud

Azure

Embedded systems
What’s available?
fullmetalupdate-cloud-demo

Docker containers started with docker-compose:
- Start an Hawkbit server instance (with MariaDB)
- Start an OsTree instance
fullmetalupdate-yocto-demo

Docker container to automate the build process:
- Select the Machine and the Yocto version to Sync
- Build the OS image: StartBuild.sh  fullmetalupdate-os
- Build the Containers: StartBuild.sh  build-container container-evcs
- Start a bash shell: StartBuild.sh bash
meta-fullmetalupdate

Includes everything which is not specific to a machine:
- Container framework recipe
- FullMetalUpdate Python client and dependencies
- BB classes to push containers and OS images to OsTree and Hawkbit
Includes everything which is specific to a machine
- GPU stuff
- Dynamic layers per machine
- Containers recipes
- Image creation scripts
Tensorflow lite in a container

- Add a new container to an IoT device
- Run this container and realize an image classification task
- Measure the performance of the classification model
QT in a container

Update your QT application seamlessly

First version

Add multiple languages

Add credit card support
ROADMAP
Roadmap

- Port to STM32MP1
- Port to i.MX8
- Final Integration with Skopeo
- Integration with Container Network Interface
- Port to Warrior and Sumo
FullMetalUpdate

- Available on GitHub:
  https://github.com/FullMetalUpdate

- Documentation:
  https://www.fullmetalupdate.io/docs/documentation/

- Do you want to contribute?
  https://www.fullmetalupdate.io/docs/contribute/
Questions
Accountability & Commitment  
Cutting Edge expertise  
Agility & Scalability  
Reliability  
Local Teams  
Worldwide presence  
Global Actions  
Compliance with customers teams

Thank you

Witekio France  
14, rue Rhin et Danube  
69009, Lyon France  
Phone : + 33 (0) 4 27 01 87 06  
sales.emea@witekio.com

Witekio USA  
3150 Richards Roads Suite 210  
Bellevue, WA, 98005, USA  
Phone : + 1 425-749-4335  
sales.amer@witekio.com

Witekio Germany  
Am Wartfeld- 61169  
Friedberg, Germany  
Phone : + 49 6031 693 7070  
sales.dach@witekio.com

Witekio UK  
Hollywood Mansion, Hollywood Lane, Bristol BS10 7TW, UK  
Phone : + 44 (0) 117 379 0220  
sales.uk@witekio.com

©2018 Witekio & Subsidiaries. All Rights Reserved.  
This document and the information it contains is confidential and remains the property of our company. It may not be copied or communicated to a third party or used for any purpose other than that for which it is supplied without the prior written consent of our company.