nRF Connect SDK

Next generation SDK for Nordic wireless solutions

Version 0.7

May 2020

Today's host

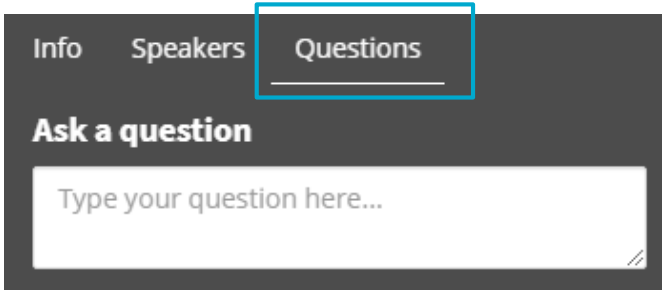
Bjørn Kvaale

Product Marketing Engineer

- Master of Science degree in Engineering Cybernetics & Robotics from the Norwegian University of Science & Technology
- Two years of experience working in the tech support group at Nordic
  - Focusing on SW related support cases, with a major focus on Bluetooth mesh & the nRF91 Series.

This webinar

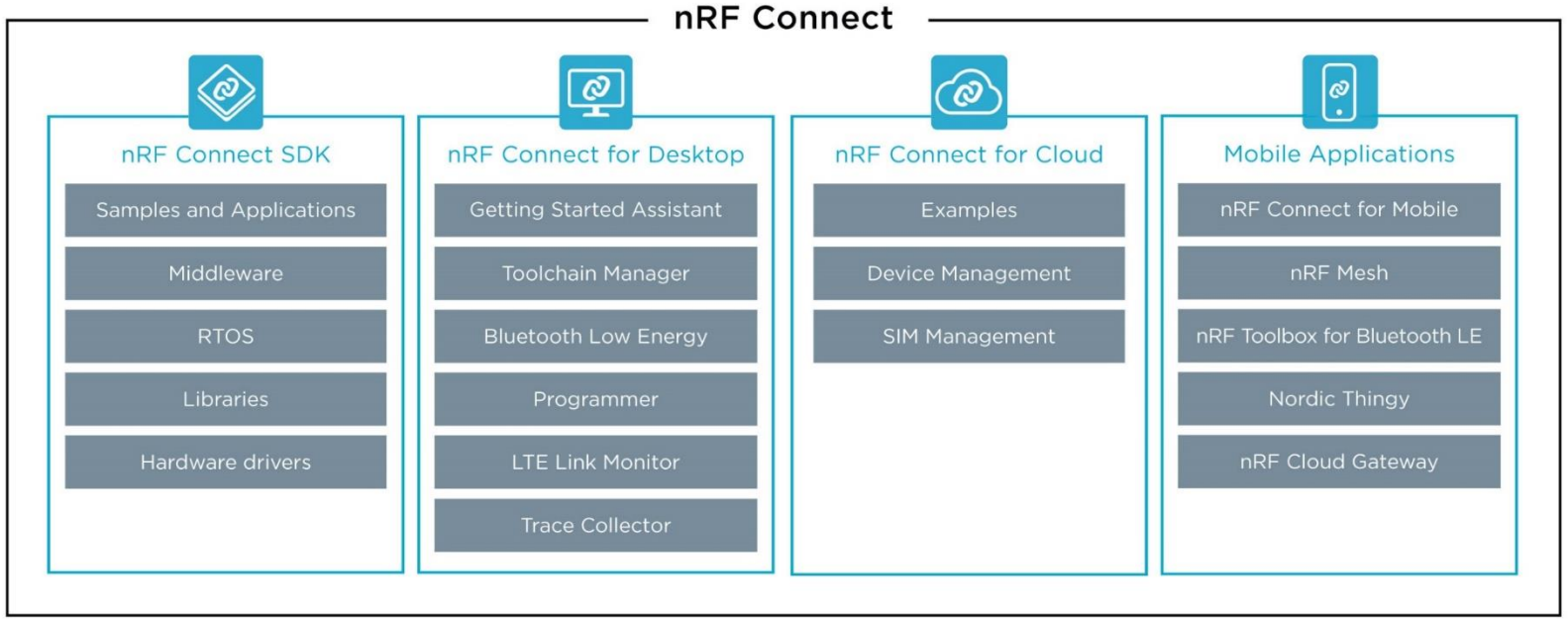
- Duration: 1 hour
- Questions are encouraged!
  Please type your question in the Questions tab on the right sidebar.
- All questions are anonymously posted.
- We will do our best to answer all questions at the end of the webinar



Content

- Overview
- Repositories and source code
- What is an RTOS?
- Enterprise Technologies and Tools
- How to manage and configure source code
- Evolution of nRF Connect SDK
- The nRF Connect platform

nRF Connect



nRF Connect SDK

Introduction

nRF Connect SDK

- All in one place
- One code base and toolchain for nRF91, nRF53 and nRF52 Series
- Optional for nRF52 Series
- Includes Bluetooth LE and LTE-M/NB-IoT
- Bluetooth mesh, Thread/Zigbee, Bluetooth 5.1 and Bluetooth 5.2 support arriving soon

LTE-M

NB-IoT

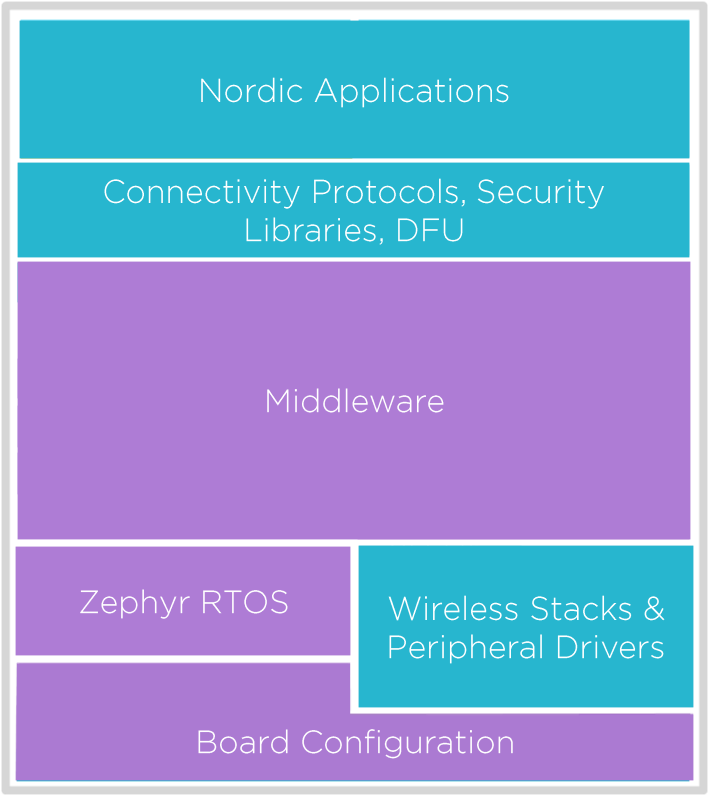
zigbee

Bluetooth®

THREAD CERTIFIED COMPONENT

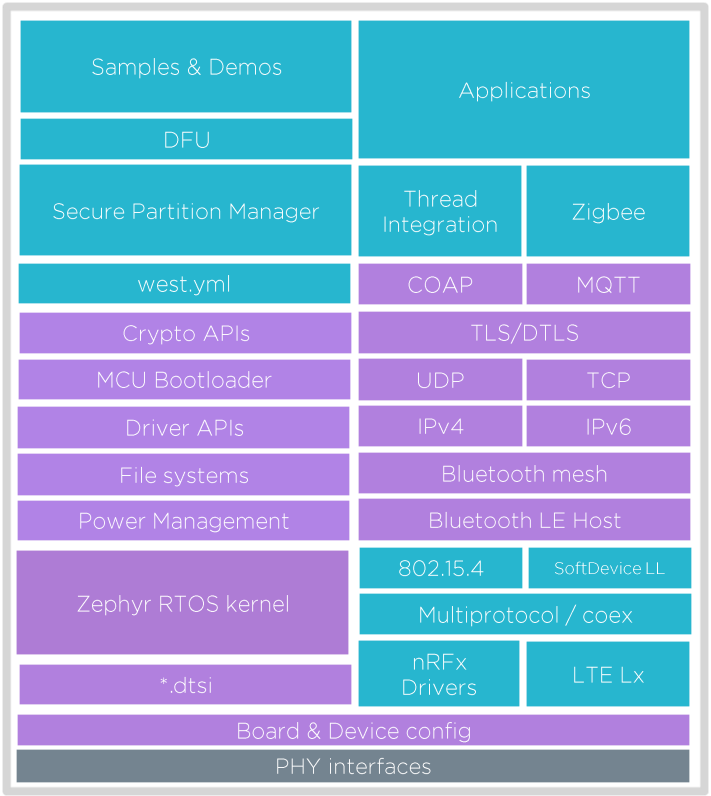
Code Base

- Contains app code, connectivity protocols, wireless stacks and peripheral drivers
- Code is organized in 4 main repositories:
  - nRF – Application & Connectivity Protocols
  - nrfxlib – Peripheral Drivers and Stacks
  - Zephyr – RTOS & Board configuration
  - MCUBoot – Bootloader



Code Base - In detail

- A wide range of wireless technologies and applications is supported by one integrated code base
- Code management, build and configuration tools allow developers to focus on the components required for their specific designs while having a powerful solution toolbox available



Supporting Product Development

Sample

Shows how it works

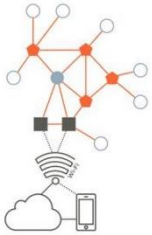
Simple examples

Demo

See what it does

Proof of concept

Application



Build a product

Fully integrated code for an application

Connectivity

LTE-M NB-IoT

Bluetooth®

Complete, verified and optimized wireless stacks, drivers and protocols

Technologies and Tools

IDE

Configure, Code & Debug

CMake and kconfig integration

OSS Platforms

Enterprise, open-source solutions

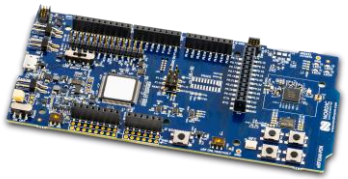
Zephyr RTOS
MCU Bootloader

Enterprise toolchain

Configure, Build & Continuously integrate

Leading standard tools integrated

Development boards

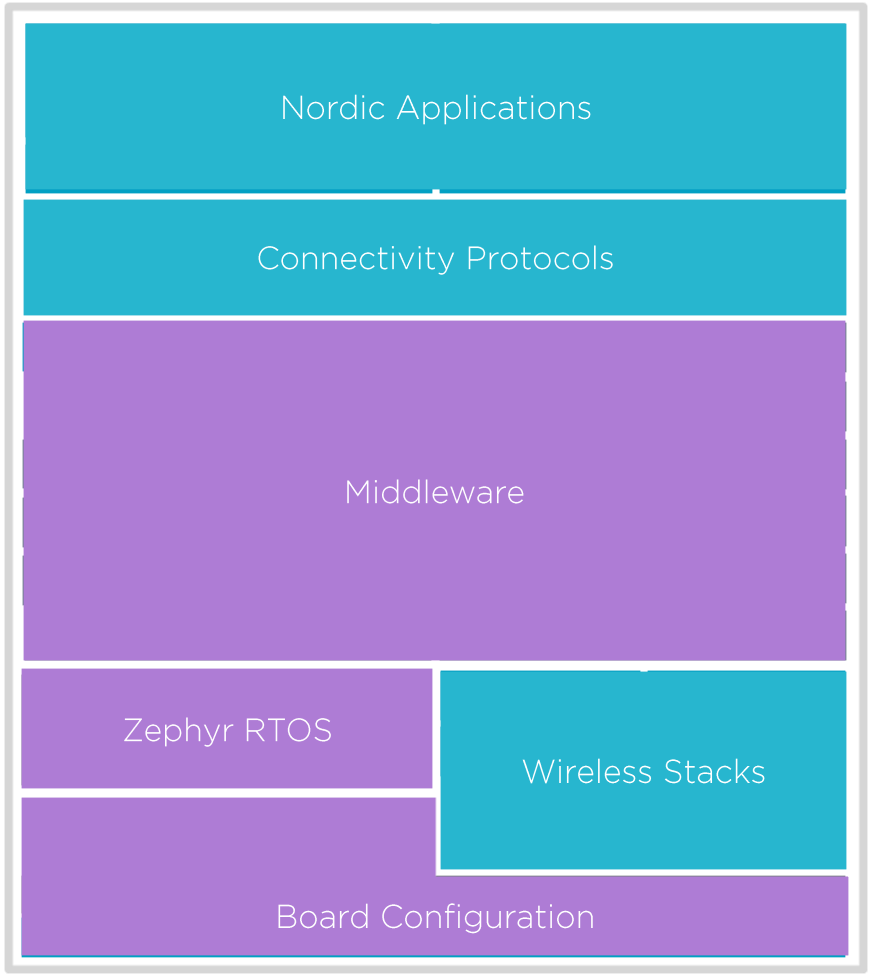


Nordic & 3rd-party board support

Prototype, develop, test, succeed

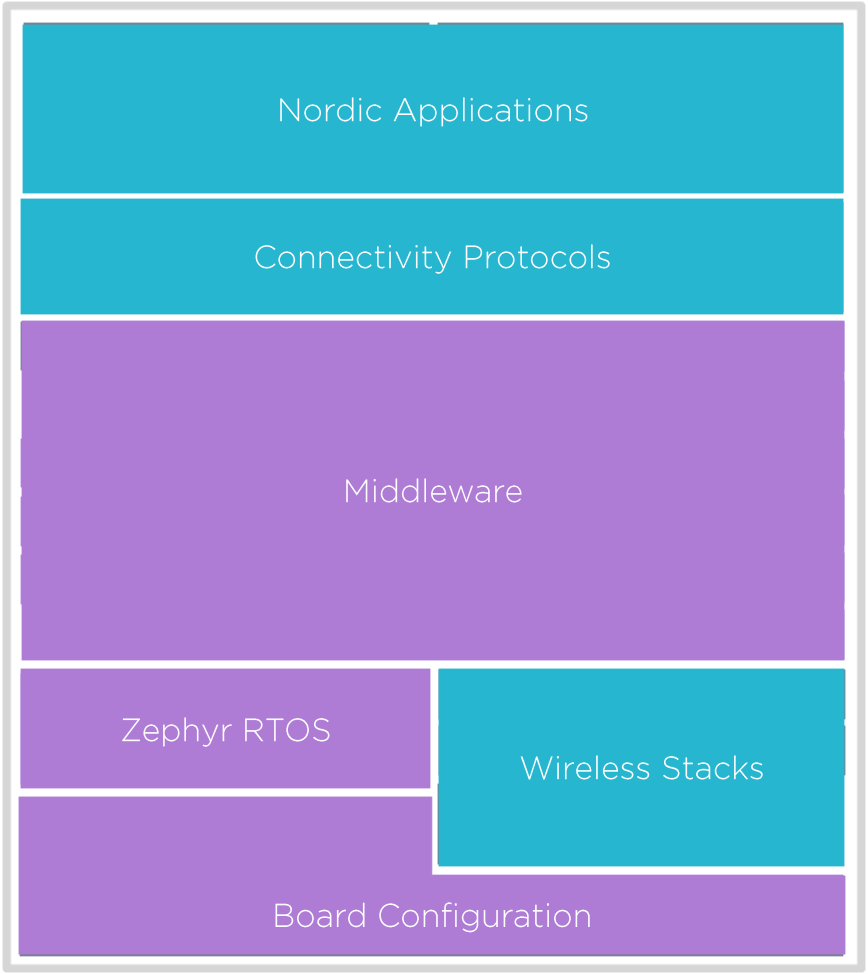
Nordic SW

- The main SDK repositories with samples, applications and connectivity stacks
- Source code exclusively written by Nordic Semiconductor
- Licensed with permissive Nordic 3-clause or 5-clause BSD license



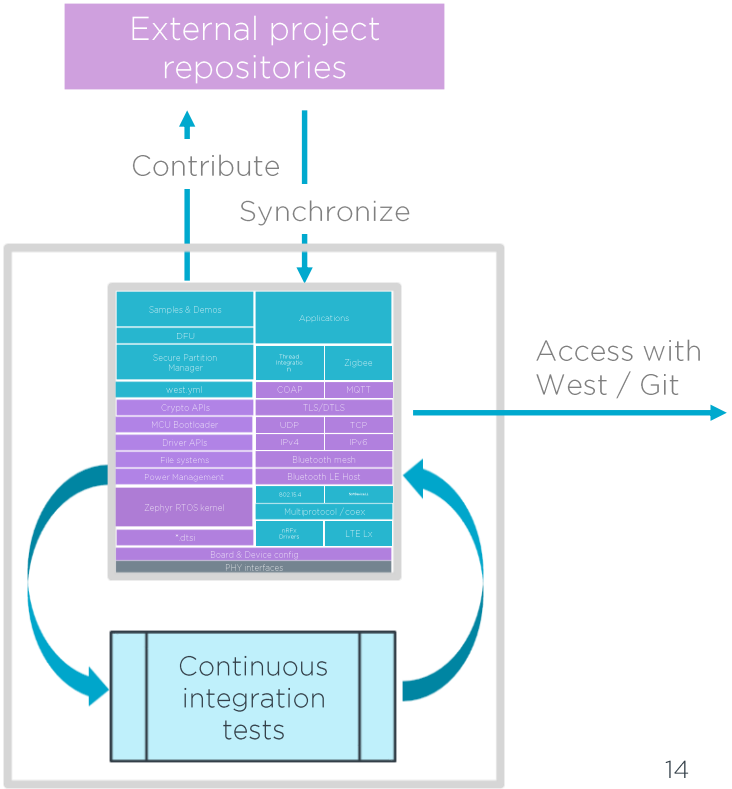
Open Source SW (OSS)

- nRF Connect SDK re-distributes OS for standard platform components
- Nordic collaborates with communities of industry experts to deliver these components as part of the nRF Connect SDK
- OSS repositories are licensed with permissive license (e.g. Apache 2.0)



Nordic Synchronizes with External Repos

- nRF Connect SDK is a single platform
- All source code is distributed by Nordic
- Includes open source code from external projects
- Nordic contributes to, and synchronizes with external projects
- Nordic runs integration test on all source code and manages configuration
- Customers clone a tag using git and west

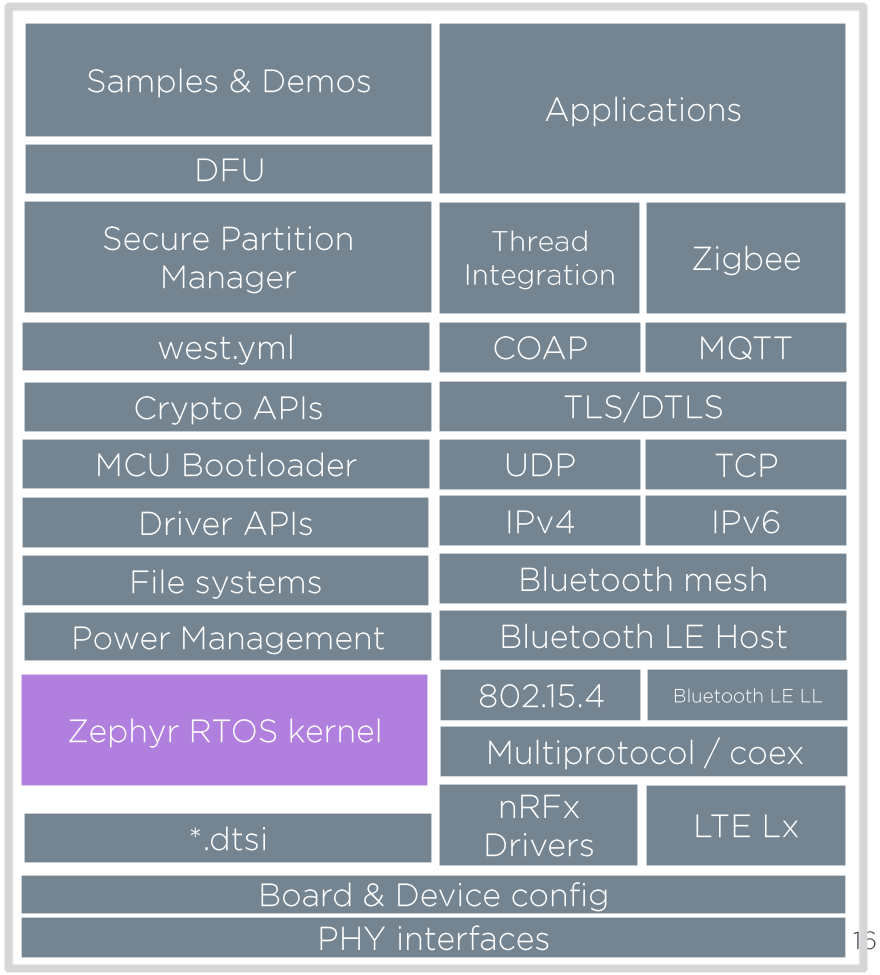


RTOS

Introduction

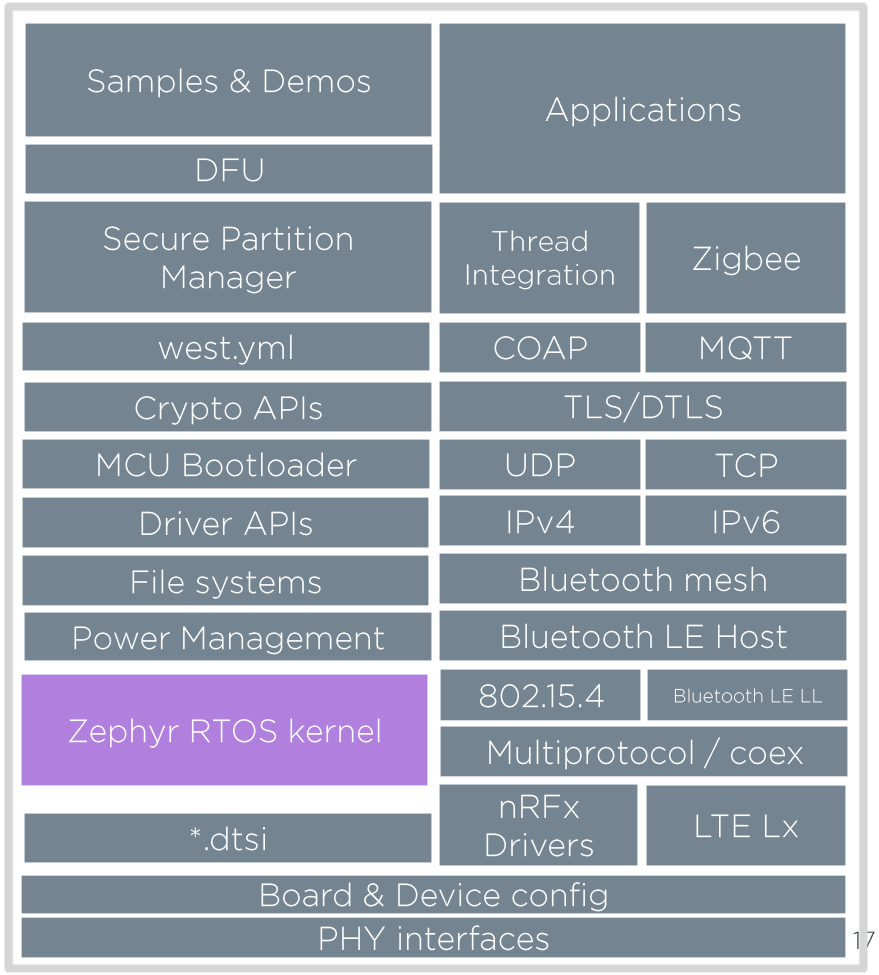
What is an RTOS?

- Real Time Operating System
  - Goal is to ensure predictable/deterministic execution pattern
  - Embedded systems often have strict timing requirements
  - Scheduler decides which task to execute at which time
  - Achievable by setting a priority for each execution thread



Why use an RTOS?

- How does an RTOS help a product developer?
  - Separation of Concern
  - More portable & re-usable applications
  - Controls application complexity in large memory devices
- Higher-level programming model
  - -> faster time to market



Why the Zephyr RTOS?

- Zephyr is designed & built for low power wireless
- Zephyr is an independent technology project
  - Governed under the Linux Foundation
  - Contributions made by over 600 industry embedded exper
- Zephyr is scalable
  - Very small configurations for memory constrained devices (< 8 kB)
  - Powerful, feature-rich, configurations for large memory, high-processing power devices (multiple MBs)
- Zephyr includes a lot of fundamental pieces of code
  - Nordic can focus on other important features

Zephyr Project Members

- Nordic a member & contributor since 2016
- We will continue to work with Zephyr
  - -> ensure best in class features & performance

Platinum Members

Silver Members

Back to nRF Connect SDK

Tools

- Use the same tools to manage your code
- Your code can be in separate repositories from nRF Connect SDK and open source
- Simplify incorporating new versions of SDK or patches
- Facilitate scalable development for multiple products and multiple teams in your organization

nRF Connect SDK Basic Terminology

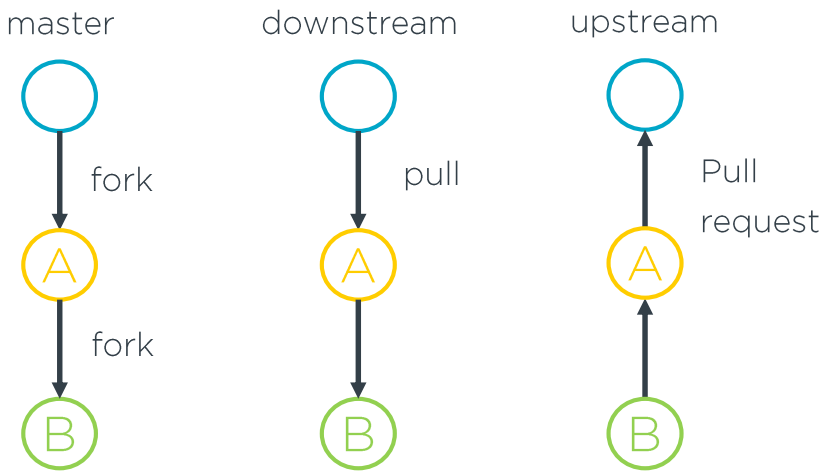
- Git
  - An open source distributed version-control system for managing source code changes
- West
  - Tool for multi-repository management and flashing devices
  - More on West later...

nRF Connect SDK Basic Terminology

- Repository
  - A version-controlled project folder e.g. nrf, zephyr
  - Every code commit creates a repository “version” with unique SHA identifier
- Tag
  - Points to a specific commit SHA, immutable
  - A human readable version reference
- Master branch
  - Points to the most recent commit SHA, mutable

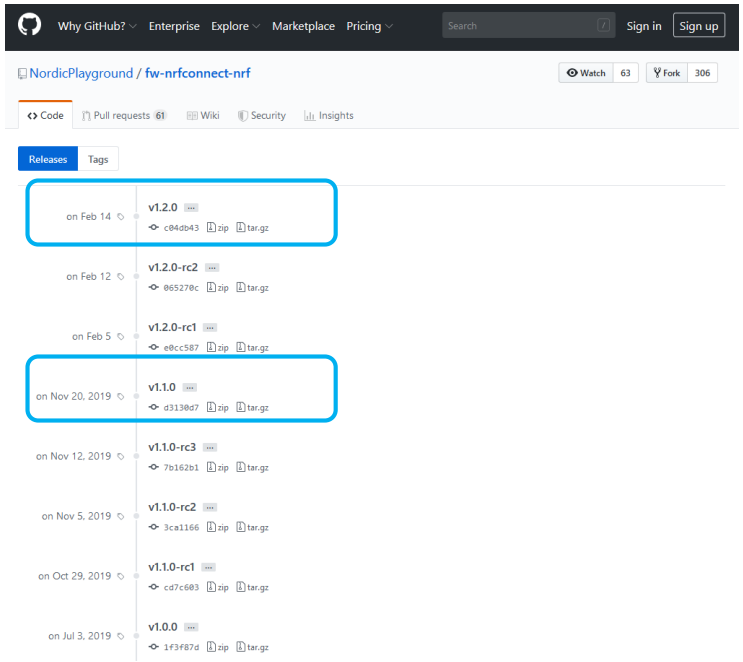
Upstream vs Downstream

- Developer A makes a fork of nRF Connect SDK master branch
- Developer B makes a fork of A's fork
- downstream
  - Flow of data is downstream from the source to the different forks (e.g. new features in master branch)
- upstream
  - Flow of data is upstream towards the source (e.g. bug fix from dev B)



Tagged vs Master Branch of nRF Repo

- Start with tagged revisions to ensure Nordic product development support is available
  - https://github.com/NordicPlayground/fw-nrfconnect-nrf/tags
  - V1.2.0 is latest tagged revision
- Use master branch to prototype with the latest features earlier
  - Newest version of nrf repository
  - Support for nRF52 Series (v1.3.0 will have support for nRF52 Series)



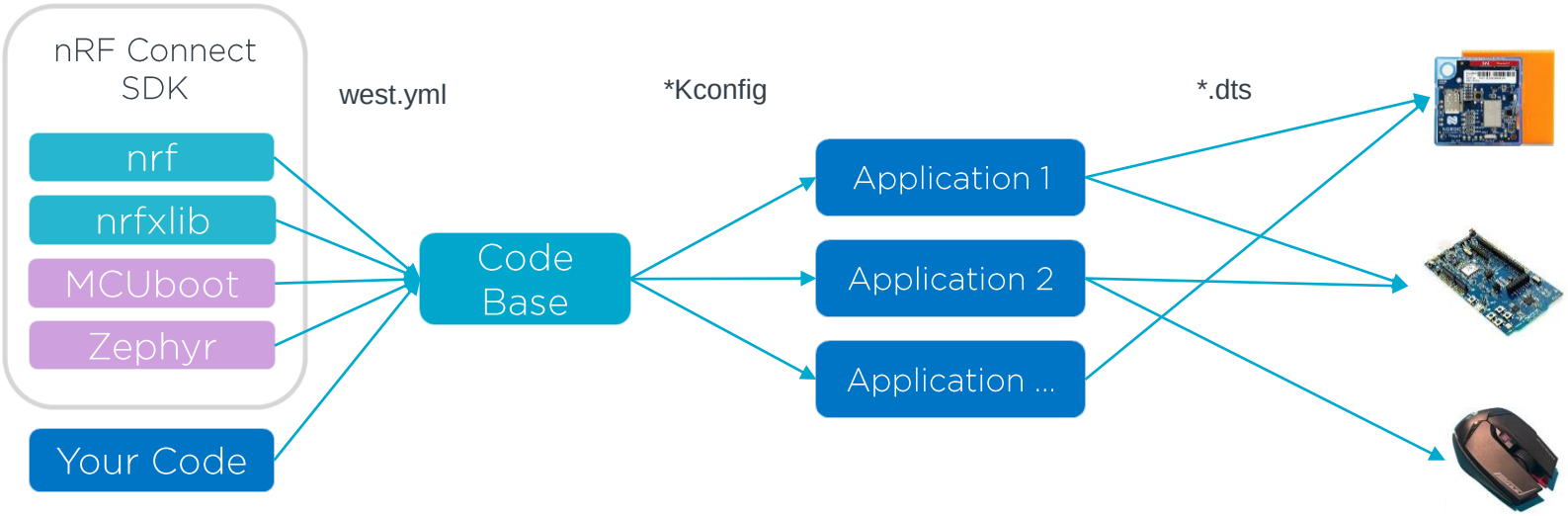
Manage Source Code and Configurations

West
Multi-repository management tool

Kconfig
Source module / feature configuration for compile

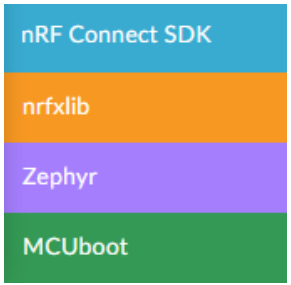
Device Tree
Target Board / Device description

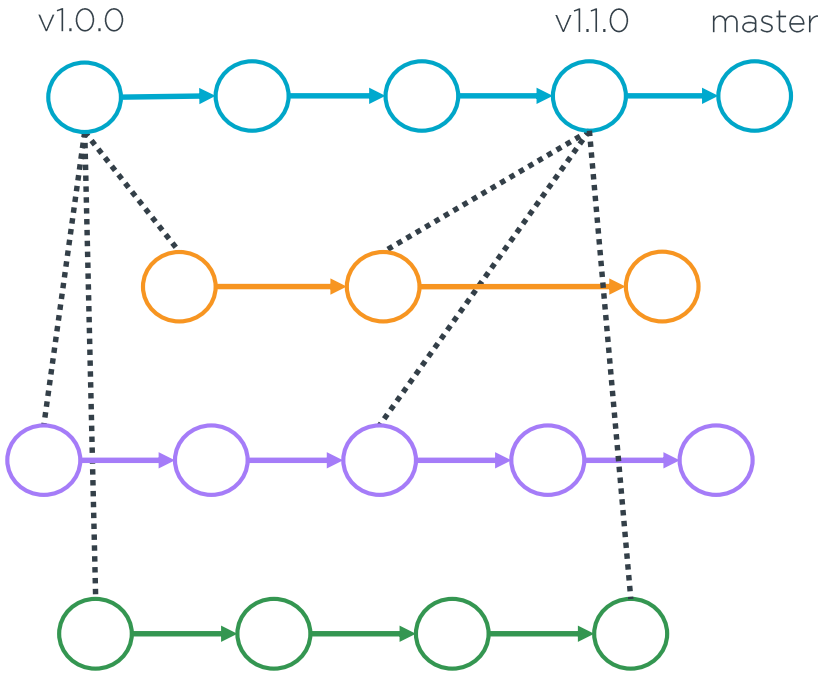
Clone / update → Configure features → Configure target



West Tool

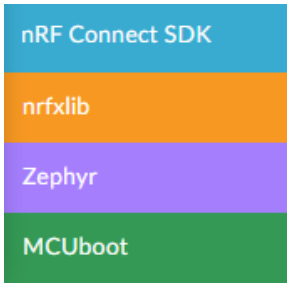
- West init
  - Initialize a new west installation
- git checkout <tag>
  - e.g. git checkout v1.1.0
- West update
- Used to update the four repositories

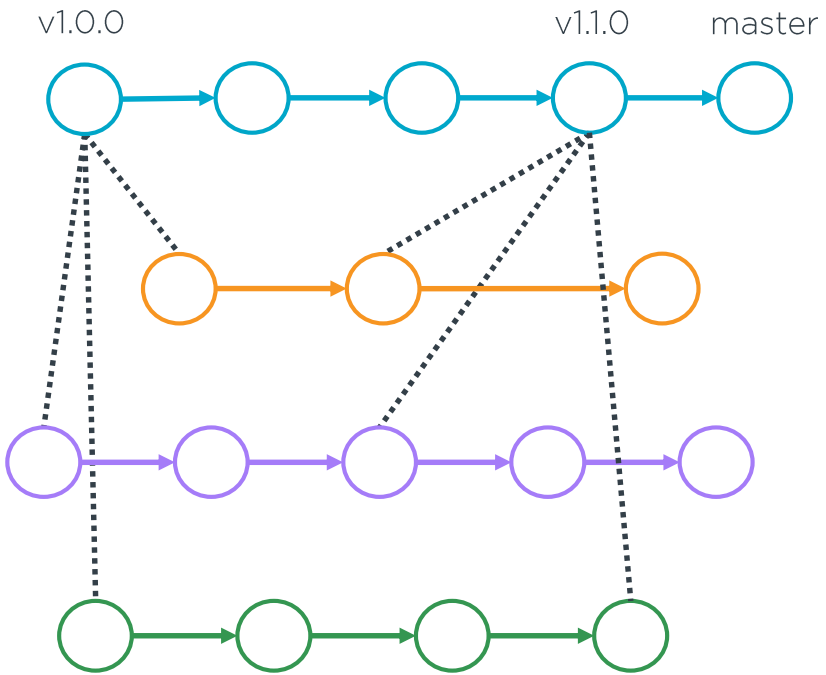




West Tool

- Using git and west simplifies multi repository management by making it seem you are only updating one repo
- Simple to migrate between different tags

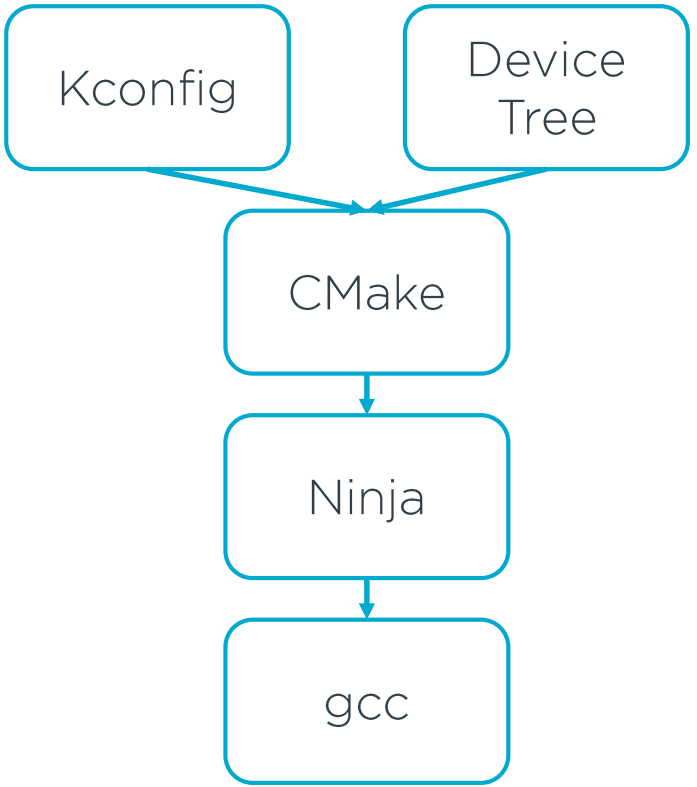




nRF Connect SDK

Toolchain

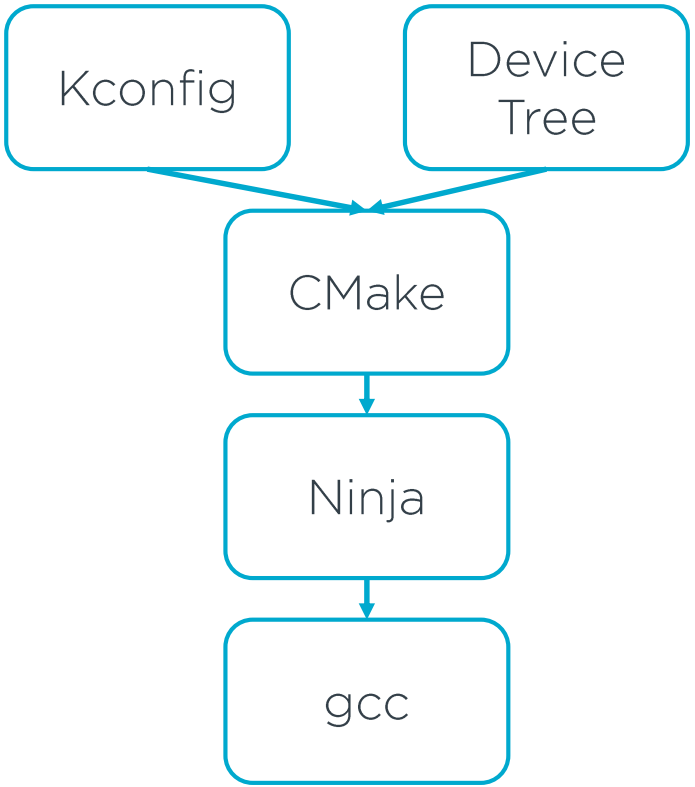
- Kconfig
  - Generates definitions to configure the system (e.g. GPS, MQTT settings)
  - Generally located in an example folder
  - → Configure features without changing source code
- Device Tree (dts,dtsi)
  - Describes HW, pin layout
  - Allows for flexible HW modification via an overlay file
  - → Build for different PCB designs and SoCs without changing source code



nRF Connect SDK

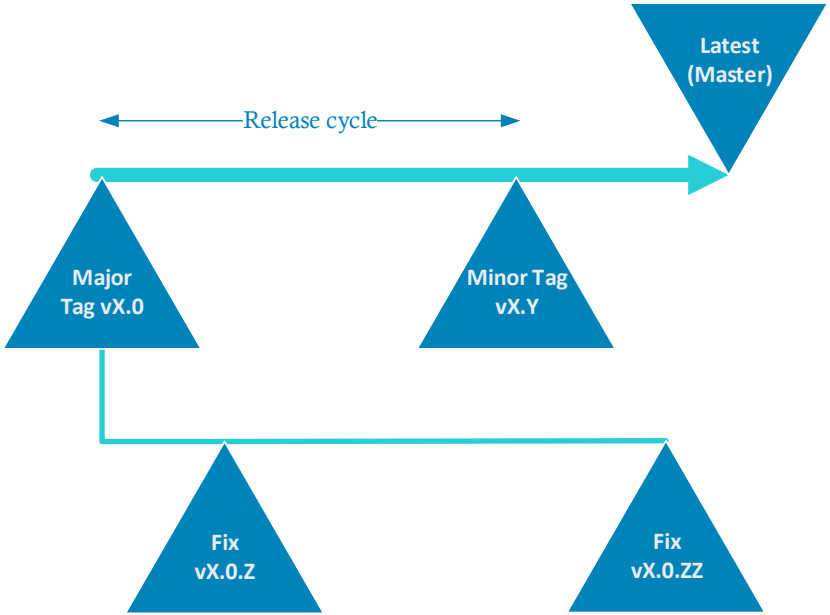
Toolchain

- CMake
  - Generates build files
- Ninja
  - Similar to make
  - Faster than make when performing incremental builds
  - Requires CMake in order to generate build files
- gcc
  - Creates executables (hex file)



Release Cycles

- Regular releases (e.g. quarterly)
- Publicly hosted on GitHub
- Fixes released as needed
  - Long term supported releases can have fixes applied and delivered after new releases
- Latest development version available
- Version control management with Git:
  - manage new version and fix adoption
  - tool supported merging



nRF52 Series SDK Support

- nRF5 SDK
  - Good if you do not require an RTOS
  - If you are used to this SDK and do not require newer features after Bluetooth 5.0
- nRF Connect SDK support optional
  - Will support newer features after Bluetooth 5.0
  - Optional for the nRF52 Series

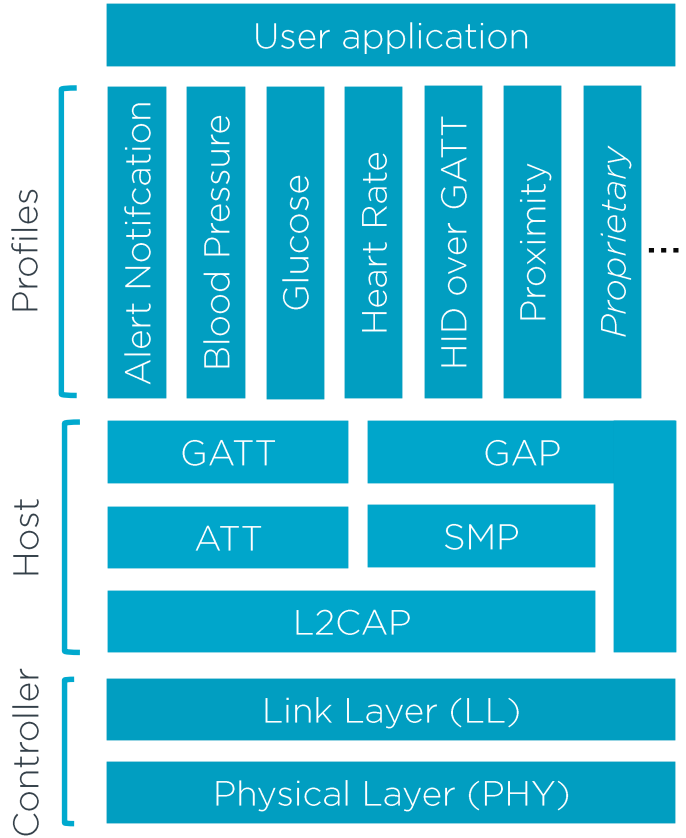
nRF52 Series Supported SDKs
nRF5 SDK
nRF5 SDK for Bluetooth mesh
nRF5 SDK for Thread/Zigbee
nRF Connect SDK

nRF Connect SDK

Evolution

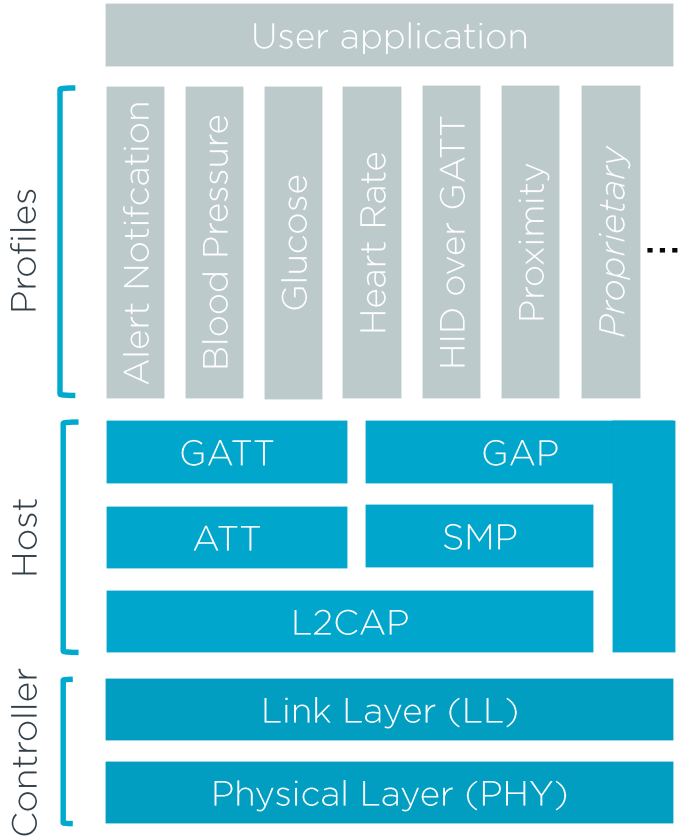
nRF5 SDK + SoftDevice

- Bluetooth LE SDK for nRF52 and nRF51 Series
- Contains multiple examples to speed up development
- Contains many Bluetooth LE profiles
- Application interfaces via SoftDevice API to SoftDevice
- nRF Connect SDK is optional for the nRF52 Series



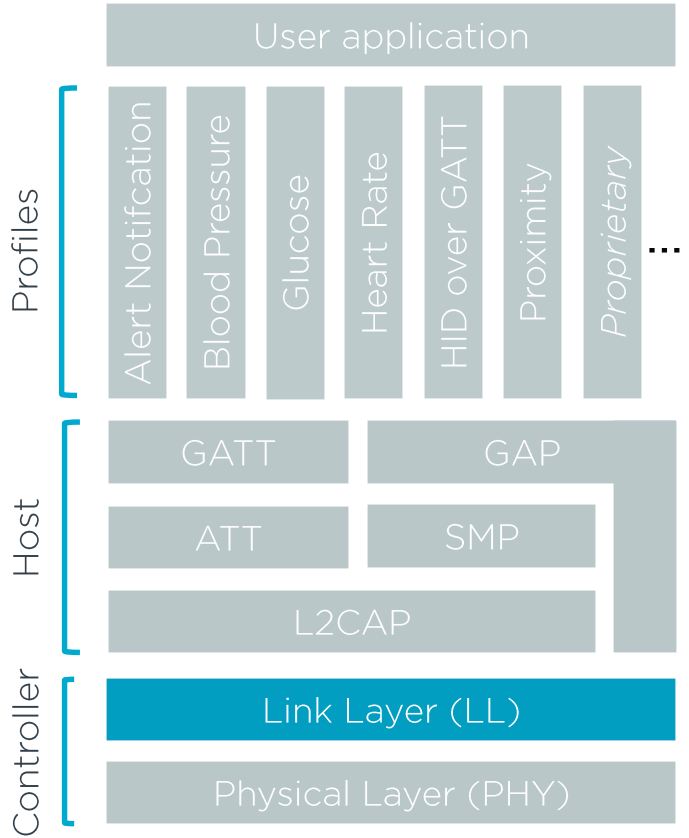
SoftDevice

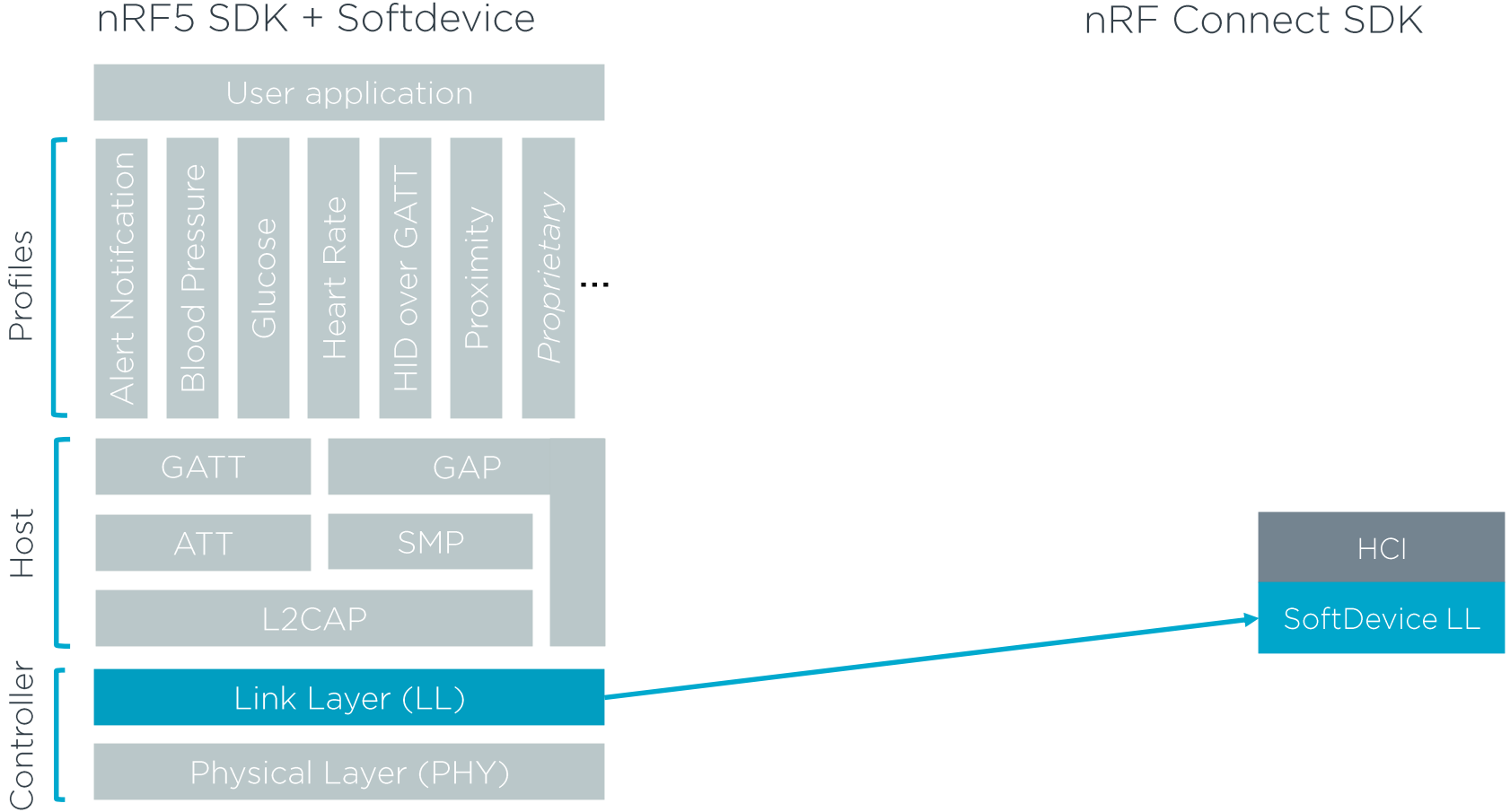
- Wireless protocol stack for Bluetooth LE 5.0 developed in house
  - Full control over features
  - We support all Bluetooth 5.0 features
- In development since 2011
- Enables best in class interoperability between SoC and smartphones
- Market leaders in Bluetooth LE in large part to our SoftDevice



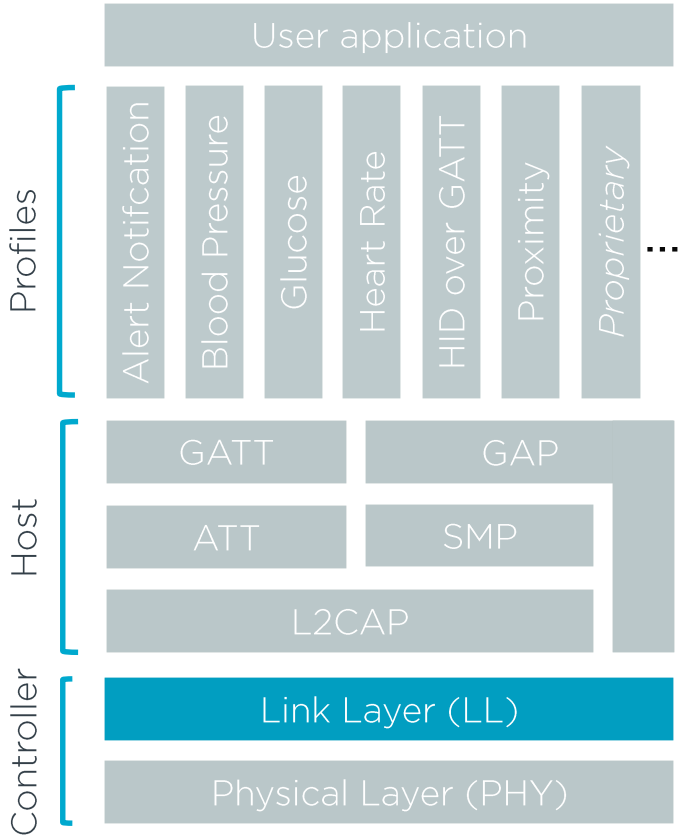
SoftDevice Link Layer

- SoftDevice Link Layer (LL) controls radio timing and enables **smartphone interoperability**
- Helps increase throughput and stability

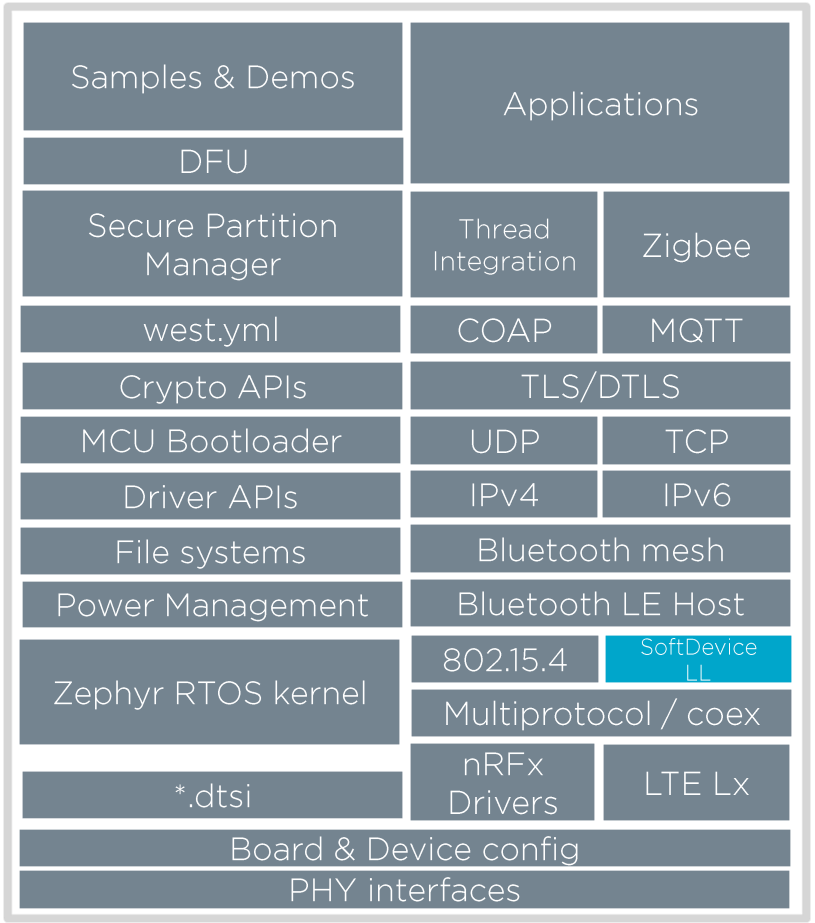




nRF5 SDK + Softdevice

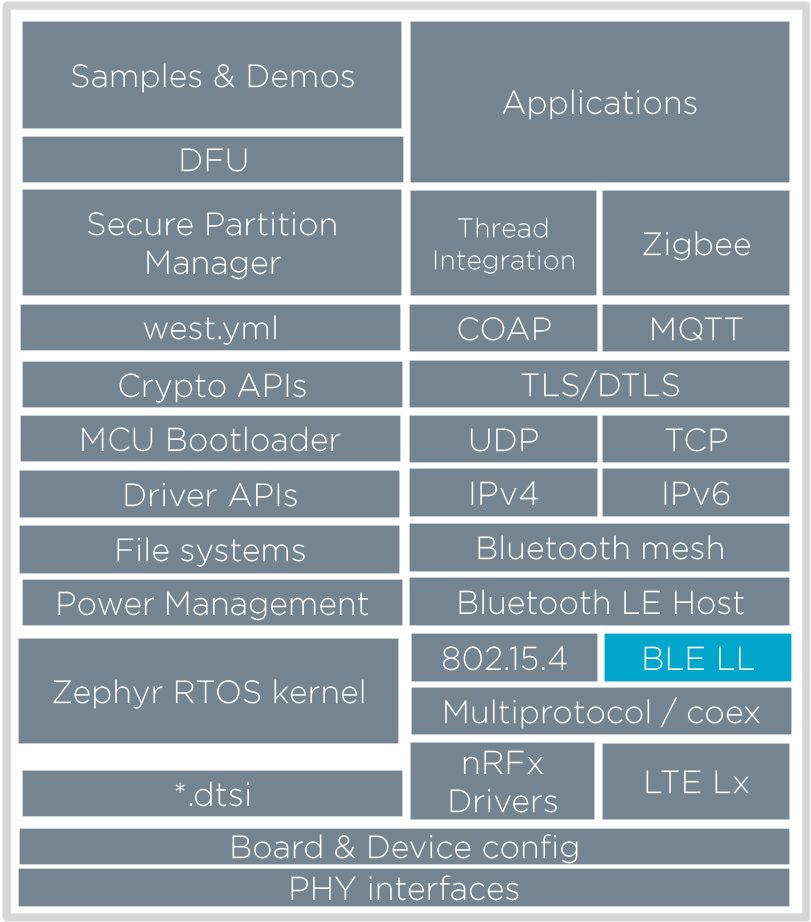


nRF Connect SDK



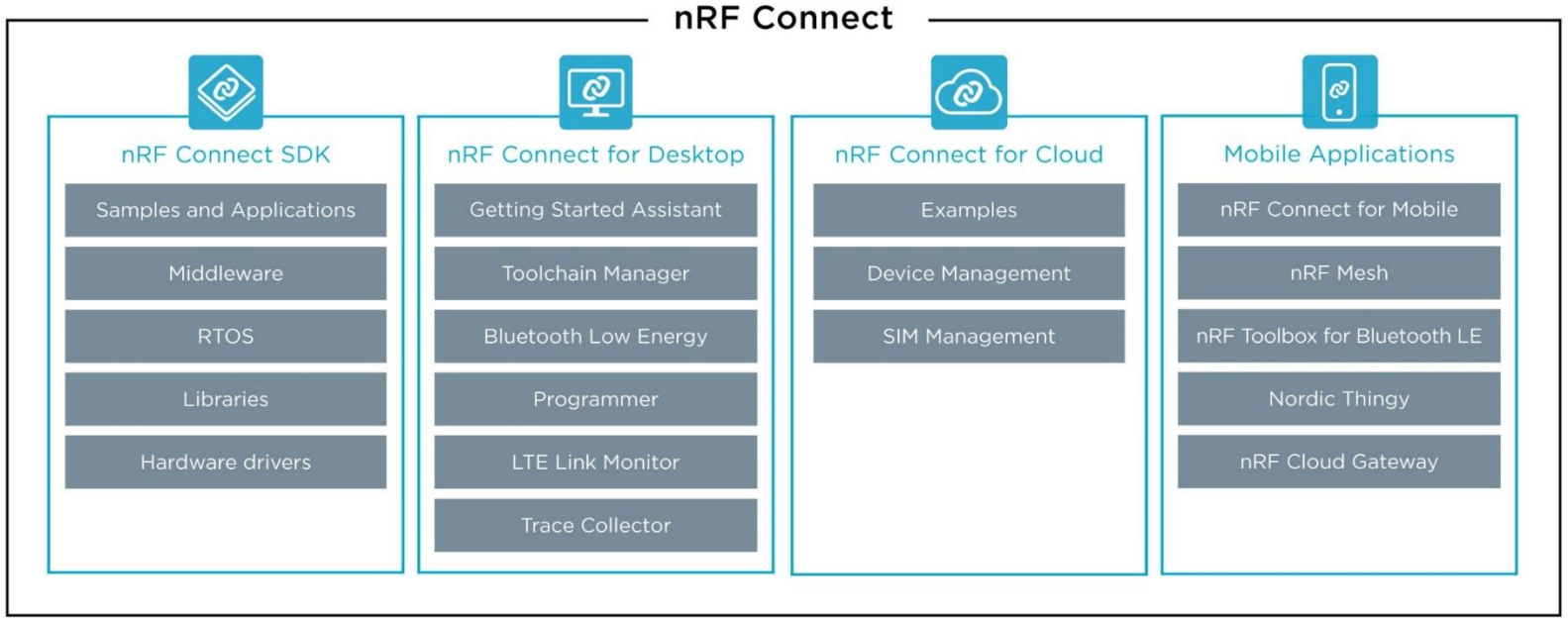
SoftDevice Link Layer

- Ported the SoftDevice LL to nRF Connect SDK
- Ensures same best in class interoperability in nRF Connect SDK as we have in nRF5 SDK
- Optimized for Nordic devices



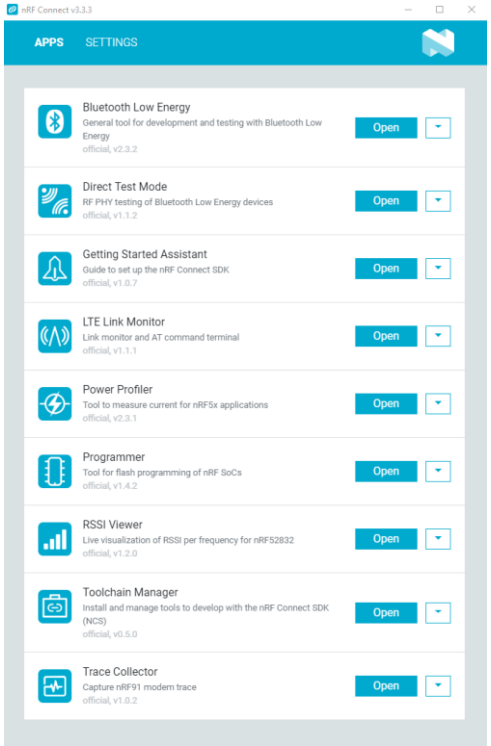
nRF Connect

nRF Connect



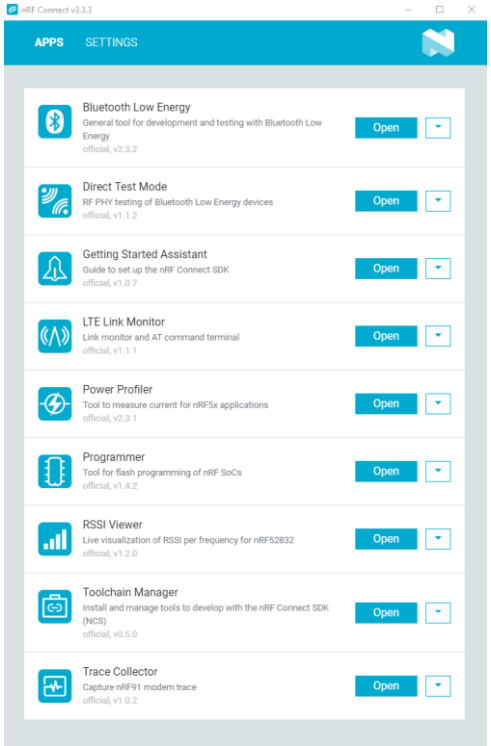
nRF Connect for Desktop

- Contains desktop apps for Windows, Linux and macOS
- Speeds up development
- LTE Link Monitor monitors LTE settings on nRF91 Series, sends AT commands, updates certificates
- Power profiler used together with Power Profiler Kit to optimize power consumption



nRF Connect for Desktop

- Programmer is a great visual tool for flashing Nordic devices, updating modem FW on nRF91 Series
- Trace Collector helps collect modem traces on nRF91 Series for further debugging via Nordic Tech Support



Toolchain Manager

- Toolchain manager provides single click download of entire code base and toolchain
  - Available for Windows as of now
  - Does not require admin rights to install
- Getting Started Assistant has same functionality, more manual process
  - Available for Windows, Linux and macOS

nRF Connect for Cloud

- Examples
  - Asset tracker – displays GPS coordinates on map and switch state
  - LTE sensor gateway – displays sensor data
- nRF9160 DK management
- SIM card management

Mobile Apps

- For Android & iOS
- Demonstrates Bluetooth SIG and proprietary services and profiles
- Plenty of apps available on Google Play and App Store:
  - nRF Connect for Mobile
  - nRF Toolbox
  - nRF Mesh
  - And more...
- Source code available on GitHub

HID Reference Design: nRF52 Desktop

- Mouse and keyboard ref HW/FW for nRF52840, nRF52832 and nRF52810 SoCs
- Gaming grade mouse report rate (1000 packets/sec) using Low Latency Packet Mode (LLPM) proprietary mode
- Uses nRF Connect SDK
- Bluetooth LE, HID, LLPM mode
- Supports OTA DFU, USB-DFU

nRF Connect SDK

Next generation SDK for Nordic wireless solutions

Version 0.7

May 2020