EdgeTech+ 2023
2023/11/15

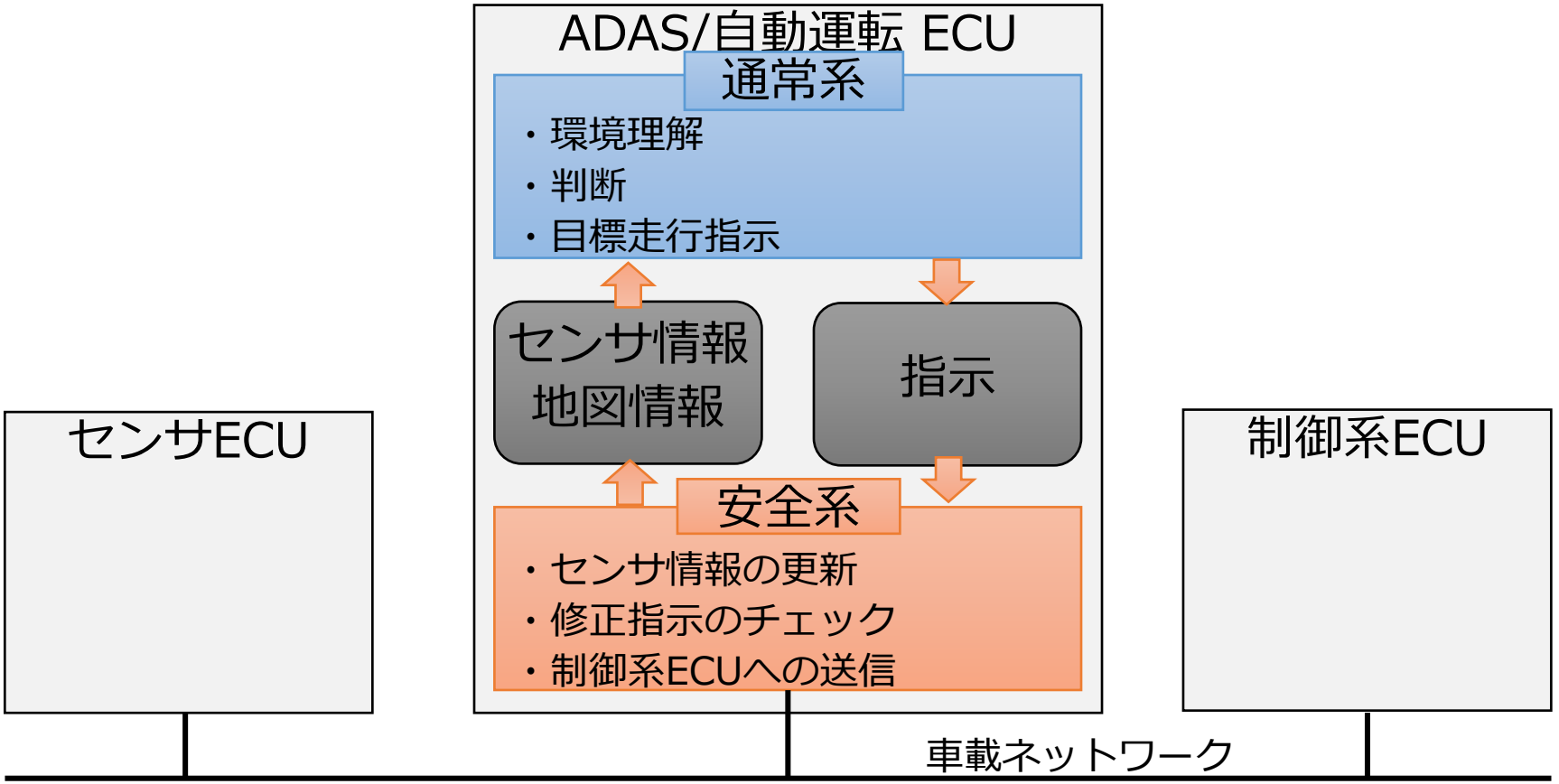
マルチOS向け環境「SafeG64」と TEE（Trusted Execution Environment）の共存技術

本田　晋也
南山大学 理工学部
TOPPERSプロジェクトシニアテクニカルエキスパート
shonda@nanzan-u.ac.jp

最終更新：2023/11/13

ADAS/自動運転 ECUの構成

- 高機能と信頼性(リアルタイム性)を実現するため2種類の系の組み合わせが有力
  - 通常系：認知・判断・操作を行う
  - 安全系：外部ECUとの通信，操作内容のチェック



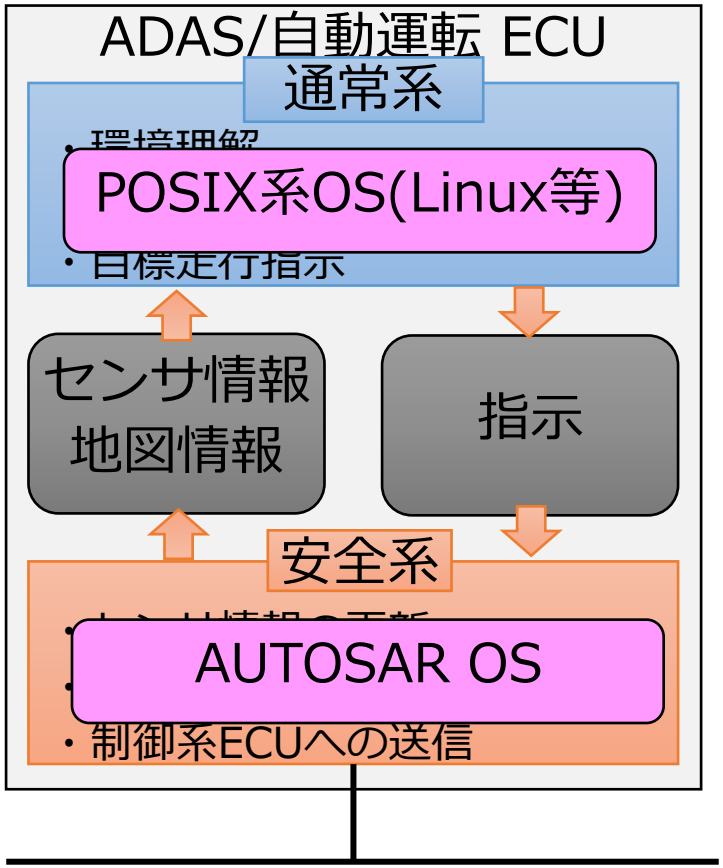
ソフトウェアアーキテクチャ

単一のOSによる実現は困難であるため系毎に独立したOSで実現

- POSIX系OS + AUTOSAR OS
- それぞれのOSが動作する系をドメインと呼ぶ

- POSIX系OSのみで実現
  - 信頼性の実現が課題（安全系と同じレベル）
    - 規模が大きく信頼性の確保が困難
    - バージョンを固定すると最新のライブラリが使用出来ない
    - AUTOSAR 関連のミドルウェアの作り直しが必要

- AUTOSAR OSのみで実現
  - POSIX系OS向けライブラリの作り直しが必要
    - 最新のライブラリを使用出来ない

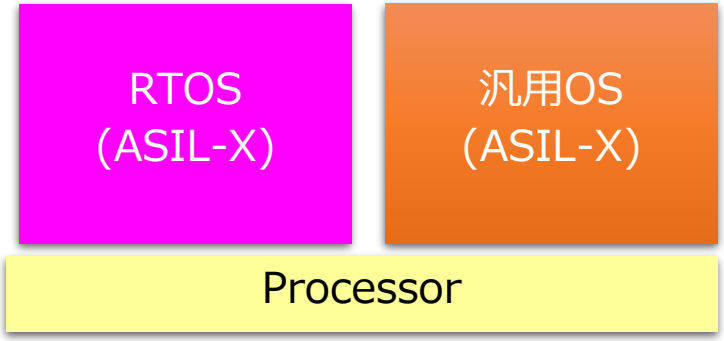


パーティショニング

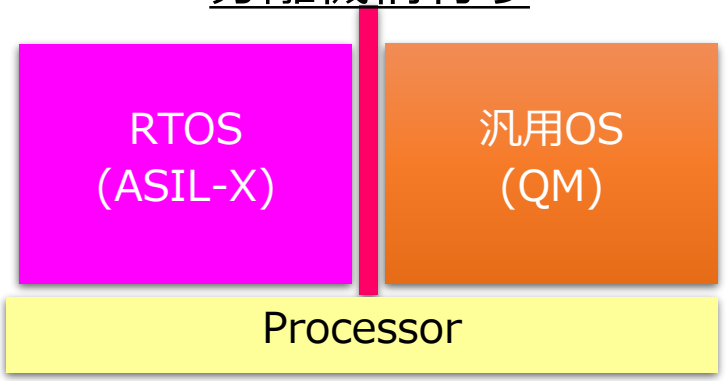
パーティショニング（分離機構）の必要性

- 分離機構がないと，最も高い安全度が要求されるモジュールの安全度水準(SIL/ASIL)でシステム全体を開発する必要がある.
  - 汎用OSをRTOSと同じ検証レベルで開発可能か?
- 分離機構があると，モジュール毎に独立した安全度水準で開発可能
  - ただし分離機構は最高の安全度水準で開発

分離機構無し

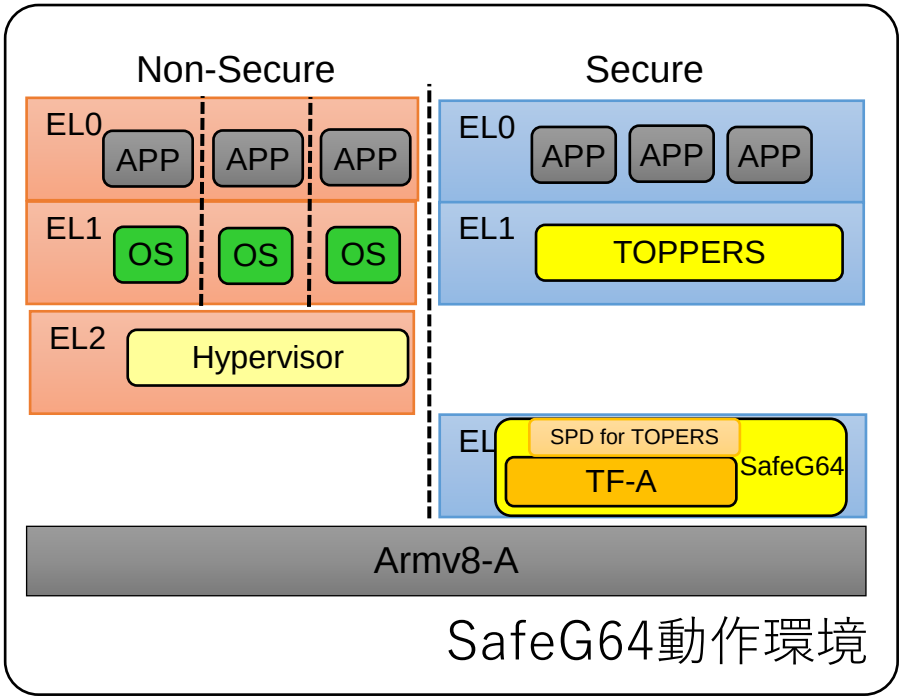


分離機構有り



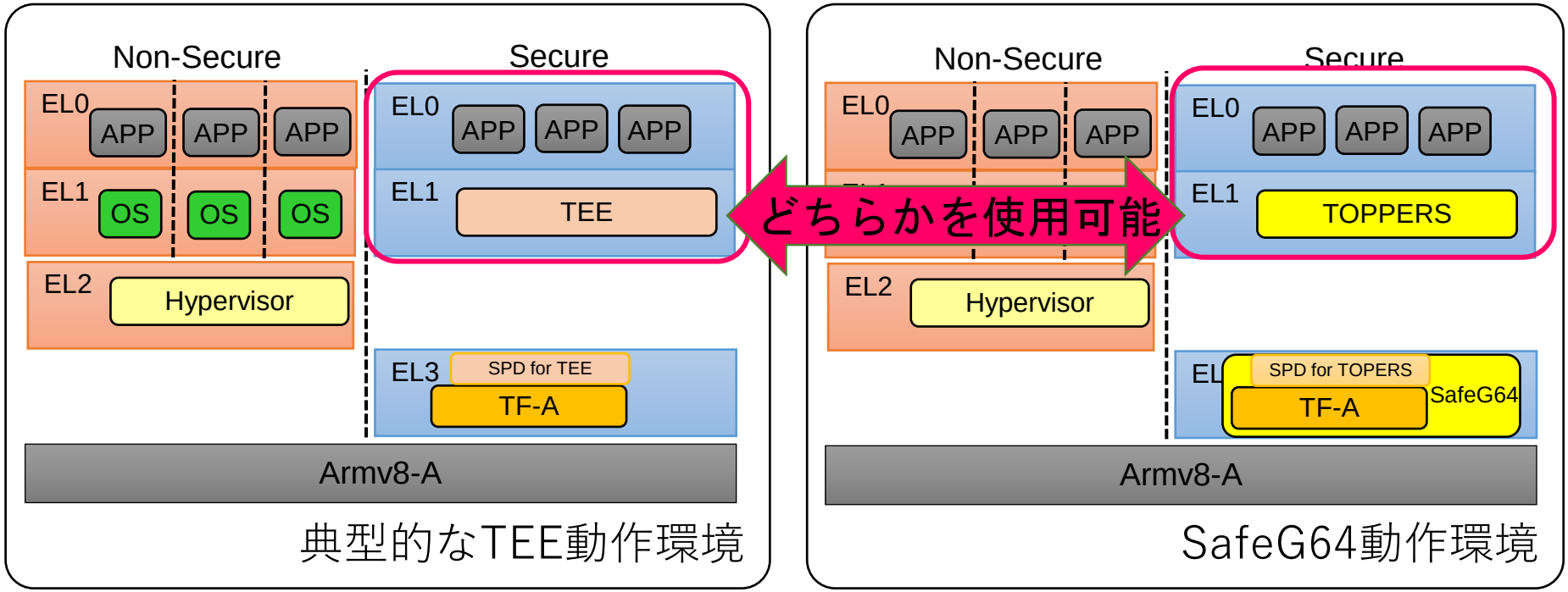
SafeG64

- ATF（Arm Trusted Firmware）をベースにARMv8-A上で汎用OSとTOPPERS OS を動作させる環境
  - Secure領域でTOPPERS OS を動作させることによりNon-Secure領域で動作する Linux から空間及び時間的に保護する
  - TOPPERS OS として FMPカーネルをサポート
  - OS間通信のサポート



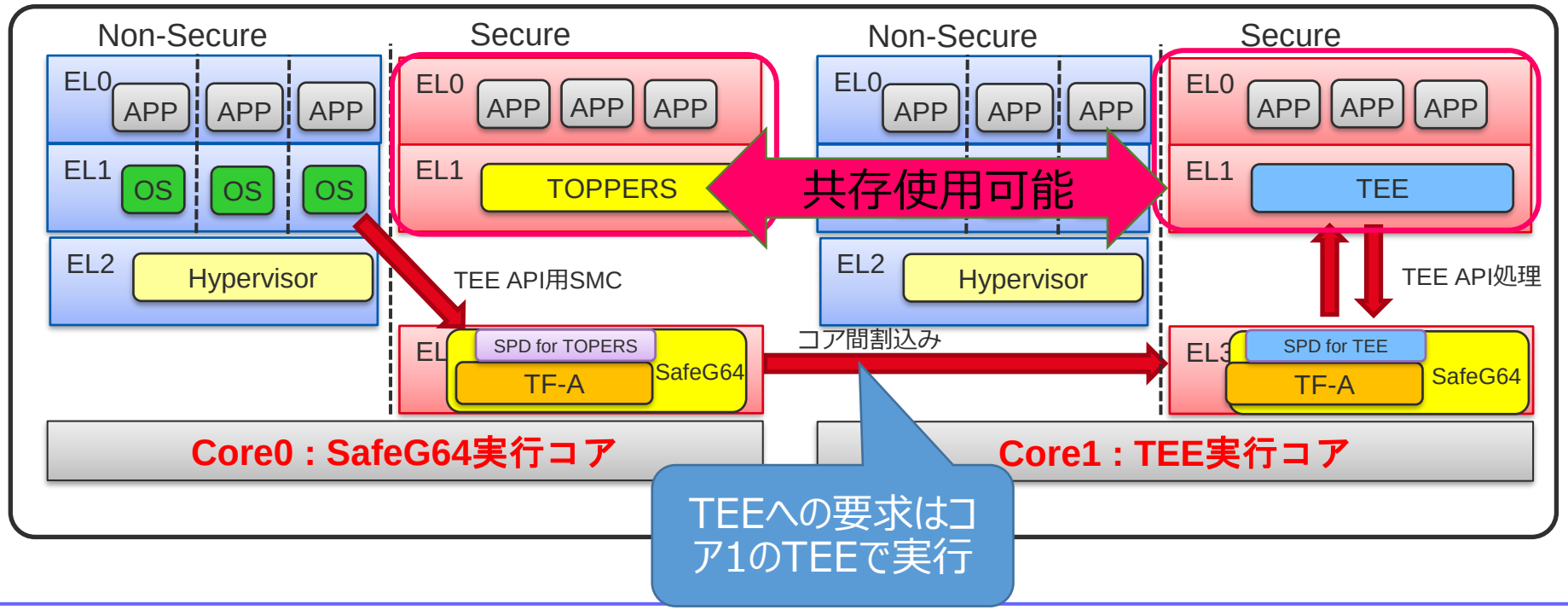
TEE(Trusted Execution Environment)

- 暗号化キーやセキュリティ証明書を安全に取り扱う環境
  - Linuxとは分離されたSecure領域で実行
  - ARM等から環境が提供されている.
- SafeG64との関係
  - どちらかを排他的にのみ使用可能
  - SafeG64と同様にSecure領域で動作するためこれまでは共存できなかった.



LinuxおよびリアルタイムOSとTEEの共存

- マルチコアにてTEE動作コアとRTOS動作コアを分ける
  - 例)TOPPERS@core1，TEE@core2
- Linux@core1からのTEE API呼び出しは、コア間割込みでcore2のTEEを呼び出して対応



まとめ

マルチOS向け環境「SafeG64」と
TEE（Trusted Execution Environment）
の共存技術について紹介

- SafeG64
  - LinuxとRTOSを同時に実行することで高機能とリアルタイム性を実現
- TEE
  - 暗号化キーやセキュリティ証明書を安全に取り扱う環境
- SafeG64とTEEの共存機能
  - 高機能とリアルタイム性に加えてセキュリティを実現