OSAKA NDS Embedded Linux Cross Online Forum #13

組込みLinux開発における課題と商用ディストリビュータの位置づけ

WNDRVR

ウインドリバー株式会社
営業技術部 シニアエンジニア
木内 志朗　(shiro.kiuchi@windriver.com)

2021年7月9日

2021 WIND RIVER, ALL RIGHTS RESERVED

アジェンダ

1. 組込みLinux開発の動向
2. Linuxを使用した製品開発における課題
3. 商用ディストリビューション としてのWind River Linux
4. Wind River ソリューションまとめ
5. まとめ

WNDRVR

2021 WIND RIVER, ALL RIGHTS RESERVED

Wind River 製品ポートフォリオ

CLOUD NATIVE

WIND RIVER STUDIO

DEVELOPER CAPABILITIES

- DIGITAL TWINS
- DEVSECOPS PIPELINE
- AUTOMATION
- DIGITAL FEEDBACK LOOP

OPERATOR CAPABILITIES

- BIG DATA ANALYTIC
- VM CLOUD
- AUTOMATION
- DISTRIBUTED CONTAINER CLOUD

CLOUD SCALE

RTOS

OPERATING SYSTEMS

VXWORKS

- VXWORKS 653
- VXWORKS CERT

WIND RIVER LINUX

- MANAGED SERVICES

WIND RIVER HELIX

- STATIC
- DYNAMIC

STARLAB

- TITANIUM + CRUCIBLE

GPOS

DEVELOPMENT

LIFECYCLE + DEVELOPMENT TOOLS

- DIAB COMPILER
- WIND RIVER WORKBENCH
- SIMICS

CUSTOMER SUCCESS

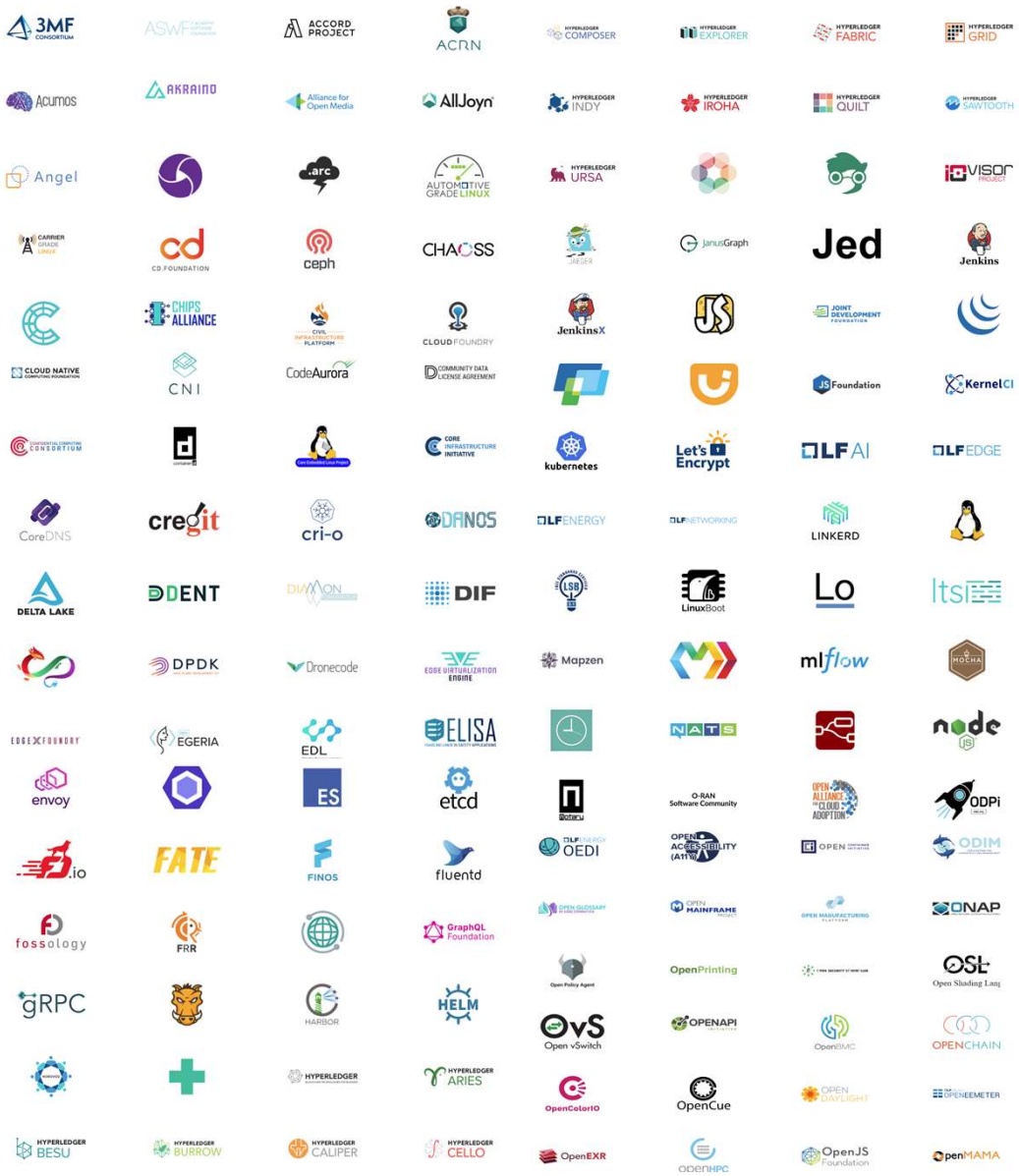
- PROFESSIONAL SERVICES
- IP SERVICES
- EDUCATION

WNDRVR

2021 WIND

1.

組込みLinux開発の動向



WNDRVR

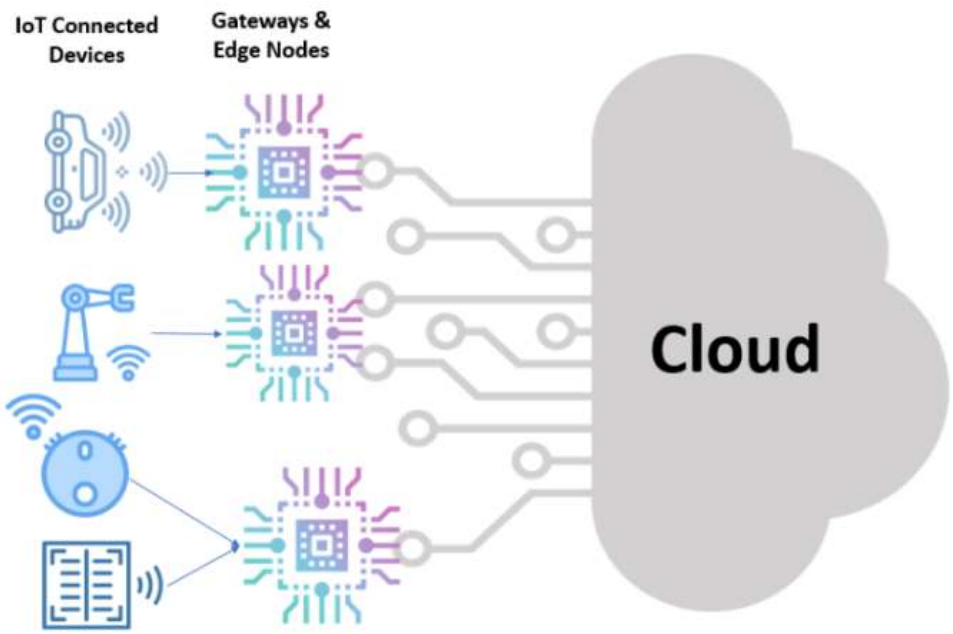
組込みシステムからインテリジェントエッジへ

from Embedded System to Intelligent Edge

- **従来のIoT ⇒ インテリジェントエッジへ**
  IoT → インテリジェントエッジ → クラウド

- **なぜインテリジェントエッジ？**
  - リアルタイム性の確保
  - セキュリティリスクの低減
  - 通信量の削減

- **インテリジェントエッジにLinuxは最適**

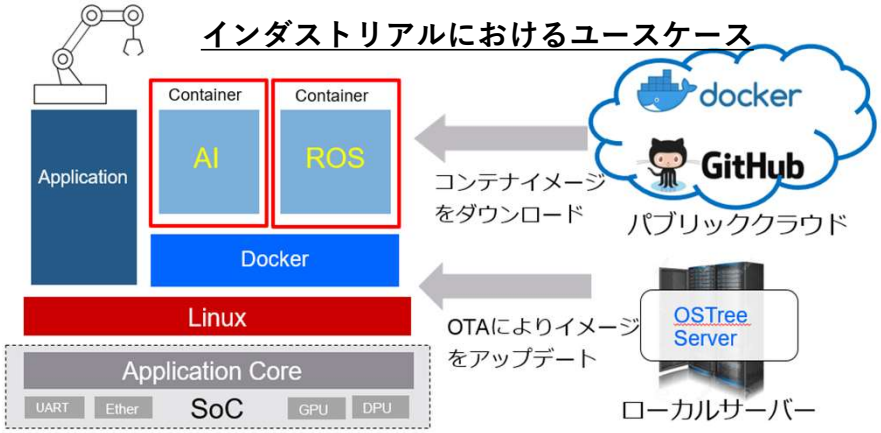


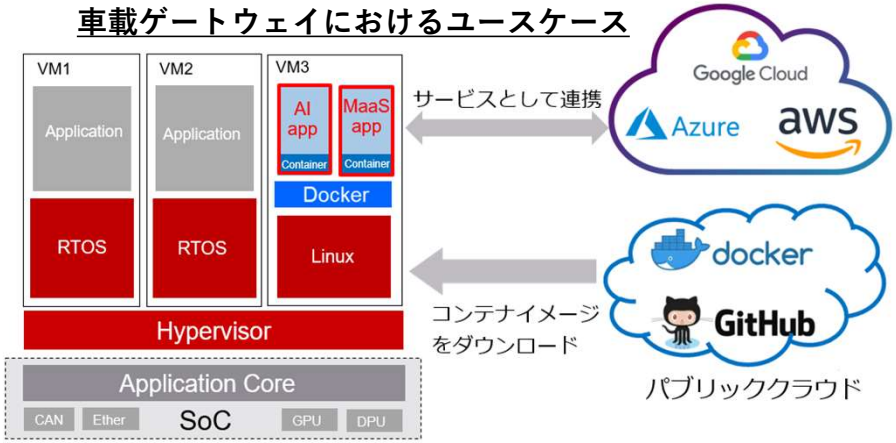
出典：IoT Magazine (https://theiotmagazine.com)

2021 WIND RIVER, ALL RIGHTS RESERVED

コンテナ/Docker /OTA技術の利用拡大

- IT系で使われてきている技術が組込みシステムで求められている
  - **Linux コンテナ**
    - 共通のカーネル上で独立したアプリケーション実行環境
  - **Docker**
    - コンテナ型の仮想環境を作成、配布、実行するためのプラットフォーム
  - **OTA（Over The Air）**
    - 無線通信などによるカーネルやアプリケーションのアップデート



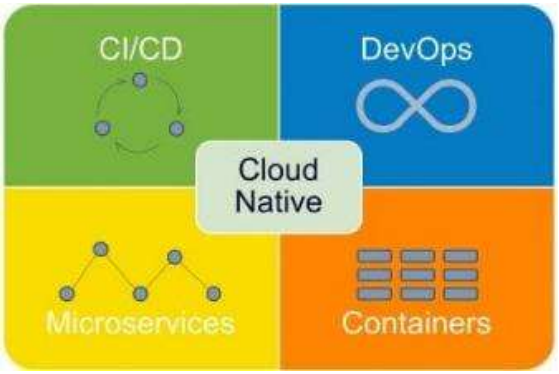


2021 WIND RIVER, ALL RIGHTS RESERVED

新しいソフトウエア開発の流れ

- 早くて安全な開発に対応するためIT系で培われた開発プロセス、開発スキームの導入が求められている
  - **CI/CD**
  - **DevOps、DevSecOps**
  - **Cloud Native**

CI/CD



Cloud Native

DevOps

DevSecOps

2021 WIND RIVER, ALL RIGHTS RESERVED

2.
Linuxを使用した製品開発における課題

WNDRVR

Linux/OSSを使用した組込み開発の課題

OSSを使用した組込みシステム開発は、非常に大きなメリットとともに、多くの問題や課題をクリアする必要がある。
予め想定されるリスクへの準備が重要となります。

製品企画

- パフォーマンス
- GENIVI Alliance
- OSSコミュニティ
- Yocto
- AI
- ブートローダ
- ユーザビリティ

製品設計

- OSSライセンス
- デバイス対応
- Linuxバージョン
- ツールチェーン
- ファイルシステム
- 省電力化
- 起動時間短縮

製品開発

- 技術者育成
- 信頼性確保
- 脆弱性対応
- 堅牢性
- GUI
- リアルタイム性
- 高可用性

製品テスト

- グローバル開発
- 改版管理
- メモリサイズ削減
- 自動テスト環境
- ネットワーク
- セキュリティホール

保守・メンテナンス

- CI/CD
- Linux Standard Base
- 知的財産権リスク
- 長期間メンテナンス
- フィールドでの障害
- サービサビリティ

2021 WIND RIVER, ALL RIGHTS RESERVED

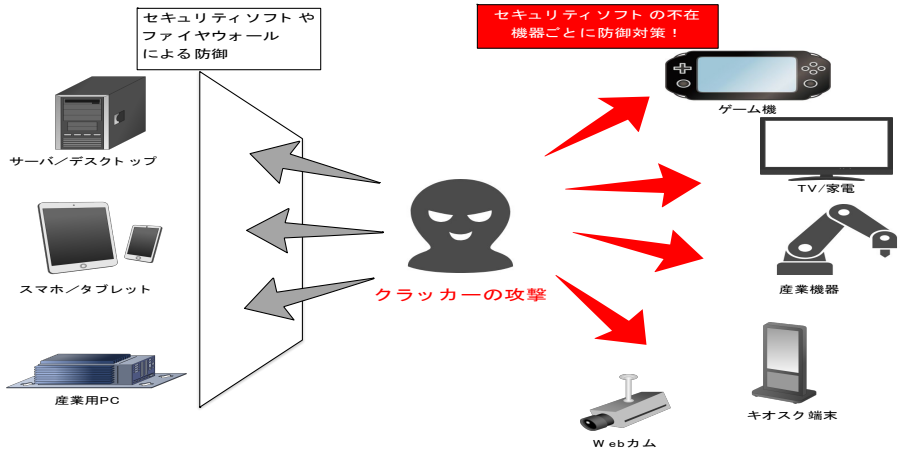
Linuxを使用した製品開発における主な課題

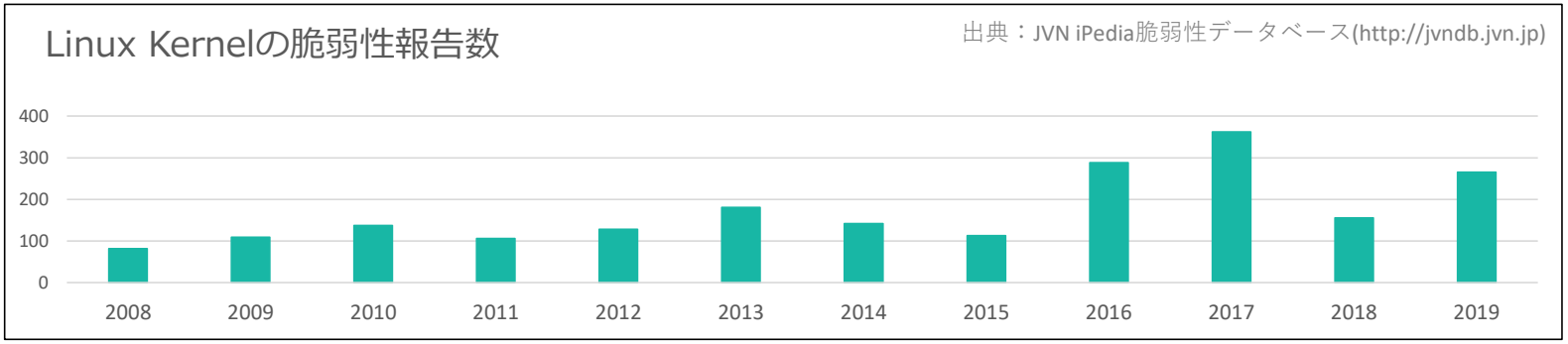
- システムデザイン
- 製品の信頼性確保
- オープンソースライセンス
- セキュリティ脆弱性
- 製品ライフサイクルにわたるメンテナンス
- CI/CD、DevOpsなどの導入
- 技術者の開発スキル

2021 WIND RIVER, ALL RIGHTS RESERVED

セキュリティ脆弱性への対応

- IoT機器などのネットワークに接続される組み込みシステムにおいてセキュリテイへの対策が非常に重要になりつつある
  - **インターネットへの接続**
    - 外部からの乗っ取り、情報漏洩、DDoS攻撃
  - **PCと同様のセキュリティの脅威**
    - PCと同様のセキュリティの脅威にさらされている
    - 組み込み製品向けのセキュリティ対策ソフトが不在





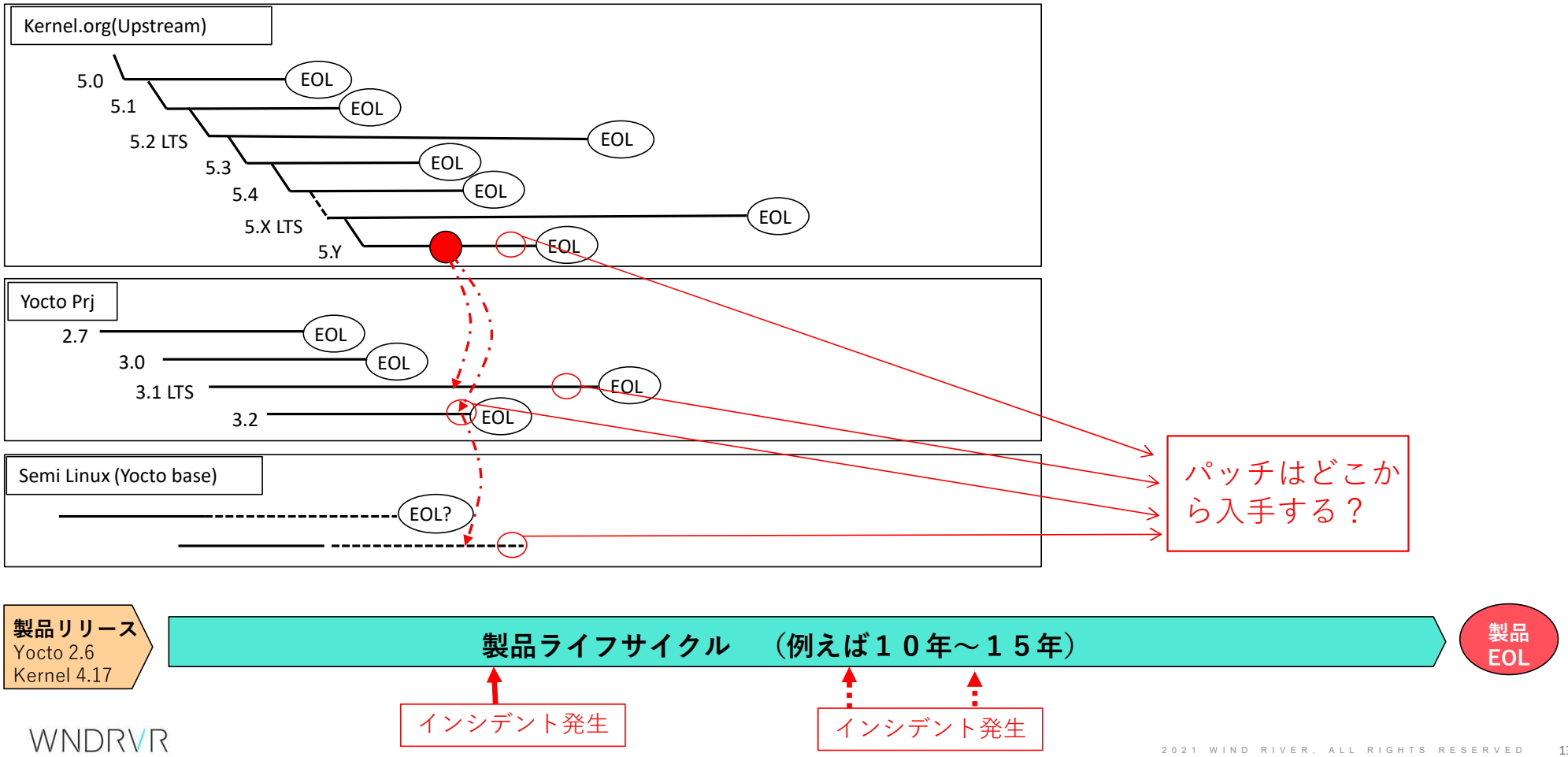
2021 WIND RIVER, ALL RIGHTS RESERVED

セキュリティ脆弱性へ対応するためには

- **セキュアなシステムの開発**
  - セキュリティ対策の作り込みが必要
  - Linux Security Module（SELinux、AppArmor、Tomoyo、SMACKなど)によるアクセス制御
  - セキュアーブート、ファイル暗号化
- **セキュリティ脆弱性情報の収集（セキュリティモニタリング）**
  - 製品開発中、リリース後に発見される脆弱性についての情報を把握する必要がある
  - 該当する脆弱性が製品に与える影響を分析
- **素早いセキュリティ脆弱性への対応**
  - ゼロデイ攻撃を防ぐ
  - OSSコミュニティでの修正パッチの適用
  - 自社製品に影響がある場合、脆弱性に対応した修正を開発、あるいはOSSアップデートパッチを適用
- **製品のアップデート機能**
  - リリース後の製品に対する修正プログラムの配布方法をあらかじめ組み込む必要がある
  - OTA（Over The Air）、S/Wアップデート配布、回収／アップデート／再配布

2021 WIND RIVER, ALL RIGHTS RESERVED

製品ライフサイクルとOSSのメンテナンス



2021 WIND RIVER, ALL RIGHTS RESERVED

製品の信頼性確保

- **OSSを使用した製品開発において品質を確保するには課題が多い**
  - 膨大なOSSの組み合わせ
  - OSSの場合、設計の上流工程まで遡ることが困難
  - OSSの仕様書はどこに？
- **使用しているOSSに対して何を保証するかを明確にする**
  - ２７００万行以上あるLinuxカーネルをすべてテストして信頼性を確保できますか？
  - 使用しているOSSの組み合わせで何を保証するか？
- **基本的に自分自身で保証しなくてはならない**
  - 製品リリースする前にはきちんとテストする、あるいは誰かにテストしてもらう必要がある
- **安全認証はまだ課題**
  - ELISAプロジェクト（https://elisa.tech)への期待

2021 WIND RIVER, ALL RIGHTS RESERVED

テストの自動化による信頼性の確保

- **OSSアップストリームとの連携**
  - OSSコミュニティでの対応状況の把握
  - 関連する修正パッチの適用
- **OSSを理解したテスト工程の制定**
- **繰り返しテスト（CI/CD パイプラインの導入）**
  - 自動化テストシステムの利用
  - テストツールなどの利用
- **品質の作り込み**

非常に多くのテスト、フレームワークが利用可能

Test Automation Tools

- 0-day
- Autotest
- Avocado (Autotest next generation)
- buildbot
- ci-rt - Linutronix Test Environment GitHub
- CKI Project - Continuous Kernel Integration @ Red Hat (code available at GitHub and GitLab)
- Fuego Test system (bitbucket repository)
- Gentoo Kernel CI
- hottest - inspired by LAVA and Fuego
- Jenkins - Automation server for building, deploying and automating any project
- kdevops (blog description)
- K2 - Test framework and tools for embedded testing
- KernelCI (wiki and docs)
- kerneltests
- Ktest Automated kernel testing
- KTF Kernel Test Framework
- labgrid - Embedded board control Python library Read The Docs
- LAVA Software Community Project - (Community Project GitLab repository)
- Opentest
- Phoronix Test Suite
- r4d - Remote For Device-under-test and (Part of ci-rt)
- Red Hat Test Project (Dead link)
- Samsung SLAV - Managing devices, tesframework, workers
- syzbot - Unsupervised coverage-guided kernel fuzzer
- tbot - Testing/automation tool that is focused on usage in embedded development GitHub
- TCF (Test Case Framework) - Simplifies the creation and execution of test cases
- Xilinx Test
- Yocto Project - Includes many test package and test elements (ptest)
- Znail - RPi network emulator, emulate packet delay, loss, duplication, corruption, rate, override DNS

See also https://bottest.wiki.kernel.org/ - which has a list of projects that test the kernel.

See also http://www.opensourcetesting.org/category/testing-tools-overview/functional/page/1/

Test Suites

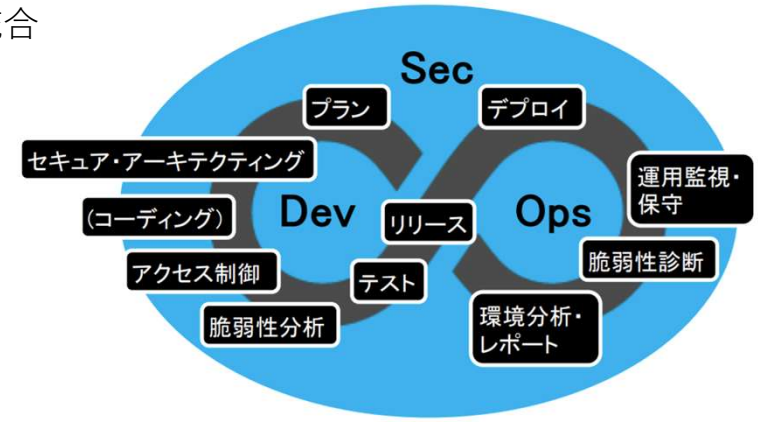
- blktests
- Linux Test Project
- LTP-DDT
- kselftest - See notes at kselftest Notes

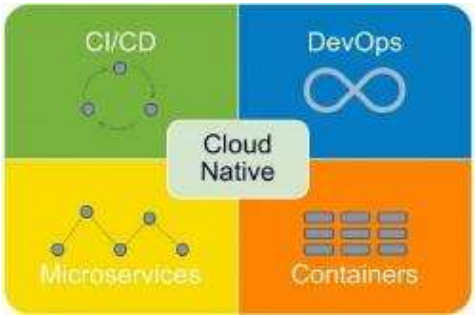
出典：https://elinux.org/Test_Systems

2021 WIND RIVER, ALL RIGHTS RESERVED

新しい開発の仕組みの導入

- **CI/CD**
  - バージョン管理システムにソースコードをコミットしたタイミングで、自動的にソースコードのビルドとテストを実行することで、バグを早期に検出する手法
- **DevOps**
  - 「開発チーム（**Dev**elopment）と運用チーム（**Op**erations）がお互いに協調し合うことで、開発・運用するソフトウェア／システムによってビジネスの価値をより高めるだけでなく、そのビジネスの価値をより確実かつ迅速にエンドユーザーに届け続ける」という概念
- **DevSecOps**
  - DevOpsにセキュリティの概念を統合
- **Cloud Native の世界へ**

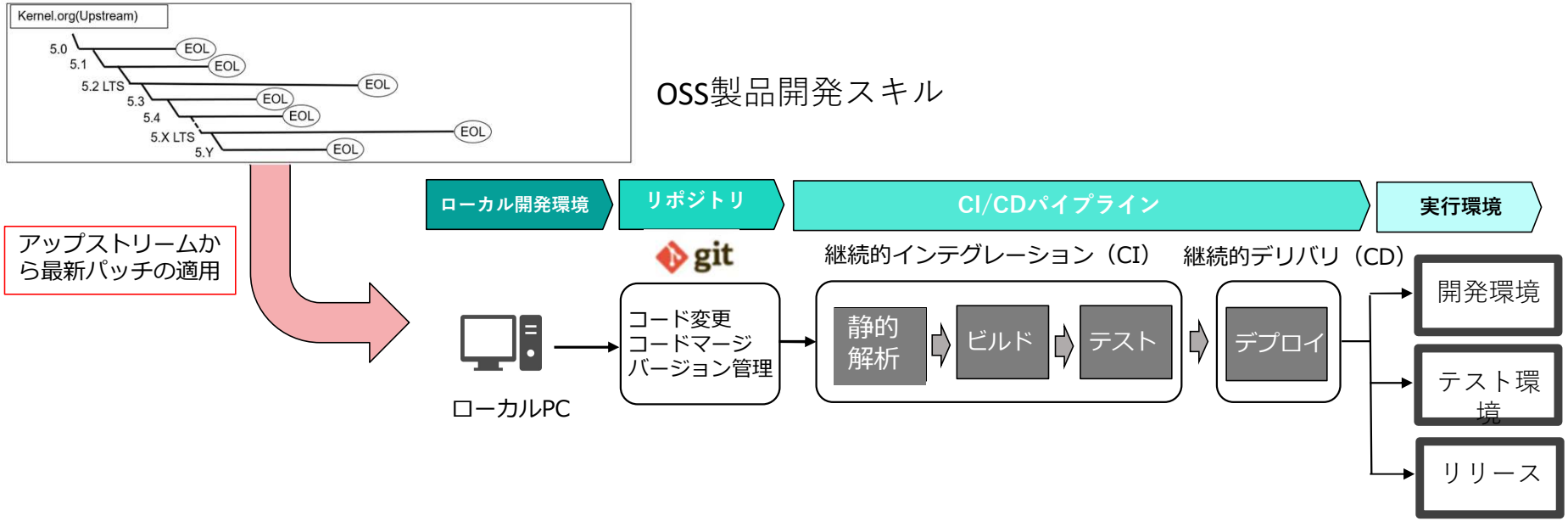




2021 WIND RIVER, ALL RIGHTS RESERVED

CI/CD パイプライン

- **不具合、セキュリティ脆弱性へ効率的に対応するためにCI/CDパイプラインといった自動化の仕組みは必要**
  - 製品開発中にも多くのバグ修正やセキュリティ対応が追加されている
  - CI/CDパイプラインの導入は簡単ではない



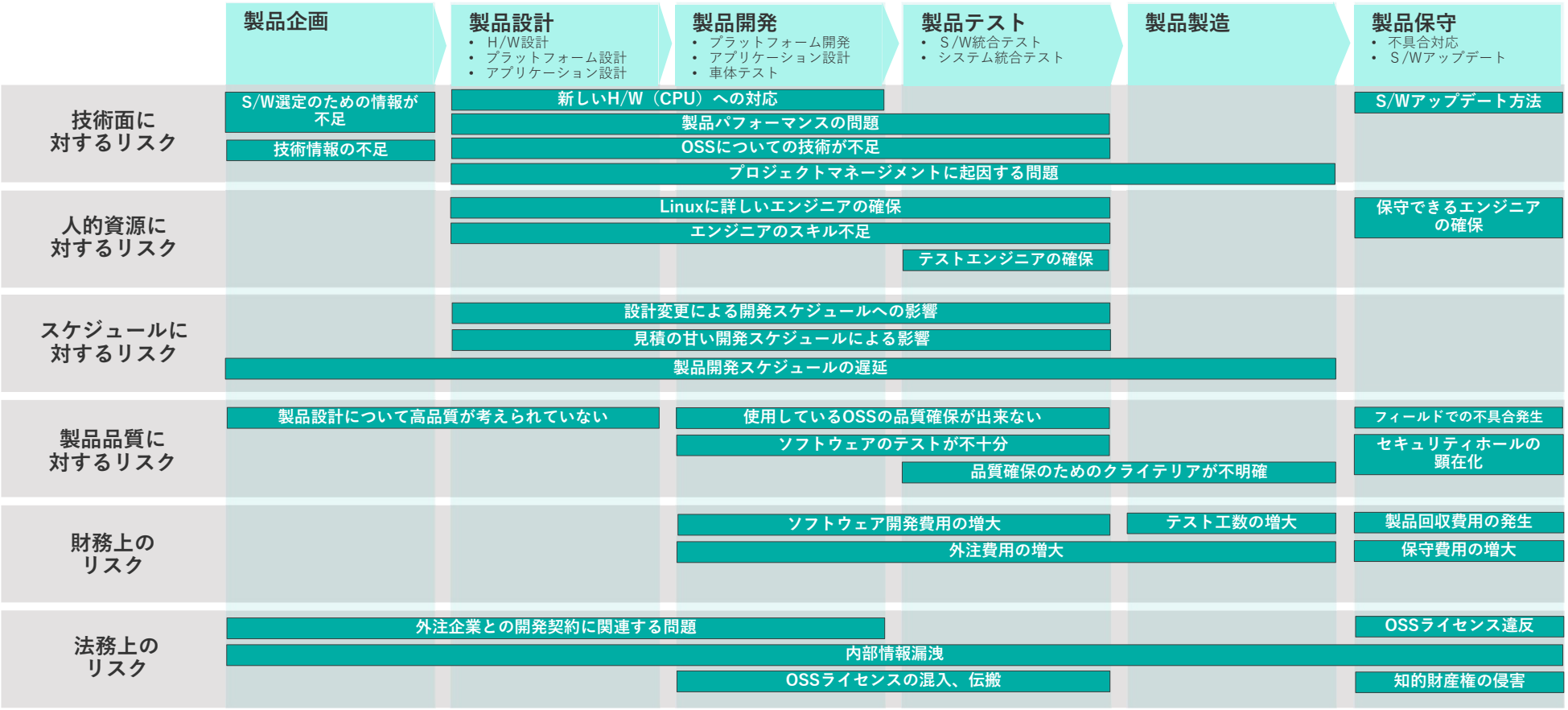
2021 WIND RIVER, ALL RIGHTS RESERVED

3.
商用ディストリビューションとしてのWind River Linux

WNDRVR

LINUXを使用した組込みシステム開発におけるリスクマップ

OSSを使用した組込みシステム開発では数多くの課題が発生する可能性があります。
予めリスクに対する構えをもつことにより、最大限OSSのメリットを享受することができます。



WNDRVR

2021 WIND RIVER, ALL RIGHTS RESERVED

OSS開発の潜在的リスクに対するウインドリバーソリューション

ウインドリバーが提供する包括的なソリューションにより、Linux/OSS開発にともなうリスクを軽減します。
トータルな製品開発費用の削減が可能です。

	製品企画	製品設計 • H/W設計 • プラットフォーム設計 • アプリケーション設計	製品開発 • プラットフォーム開発 • アプリケーション設計 • 車体テスト	製品テスト • S/W統合テスト • システム統合テスト	製品製造	製品保守 • 不具合対応 • S/Wアップデート
技術面に対するリスク	S/W選定のための情報が不足 技術情報の不足	新しいH/W（CPU）への対応	プロジェクトマネージメントに起因する問題			S/Wアップデート方法
人的資源に対するリスク		Linuxに詳しいエンジニアの確保				保守できるエンジニアの確保
スケジュールに対するリスク		見積の甘い開発スケジュールによる影響	製品開発スケジュールの遅延			
製品品質に対するリスク	製品設計について高品質が考えられていない		使用しているOSSの品質確保が出来ない			フィールドでの不具合発生 セキュリティホールの顕在化
財務上のリスク			ソフトウェア開発費用の増大	テスト工数の増大		製品回収費用の発生 費用の増大
法務上のリスク			外注企業との開発契約に関連する問題 OSSライセンスの混入、伝搬			OSSライセンス違反 知的財産権の侵害

Wind River Linux サブスクリプション

Wind River Linux プロフェッショナルサービス

- カスタムBSP、デバイスドライバ開発
- BSPバリデーション/サポート/メンテナンス
- Long Term Maintenance
- Custom Contents Management
- Frozen Branch Maintenance

Wind River Linux製品

包括的なソリューションにより全体コストを最適化

Wind River コンサルティングサービス

WNDRVR

2021 WIND RIVER, ALL RIGHTS RESERVED

Wind River Linux 全体図

OSSコンプライアンス用書類に対応

脆弱性対応

最大15年の長期サポート
（5年間の標準サポート＋オプションのプレミアムサポート・メンテナンス）

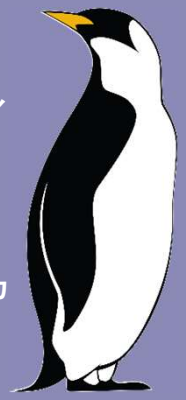
Yocto Project互換のオープンソース開発環境

yocto™ PROJECT

最適化されたクロスアーキテクチャランタイムによる優れたエクスペリエンス

追加テストと検証による高い信頼性

必要な技術資料が用意されたシームレスなカスタマーエクスペリエンス



長期サポートとメンテナンス

Star Lab セキュリティ製品群

Workbench 統合開発環境とツールキット

ディストリビューション管理サービス

2021 WIND RIVER, ALL RIGHTS RESERVED

Yocto Projectのトップレベルのコントリビュータとして貢献

オープンソースにおけるリーダーシップ

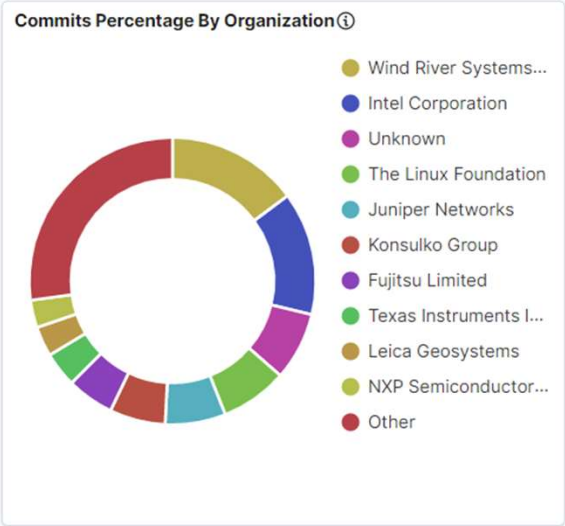
yocto PROJECT

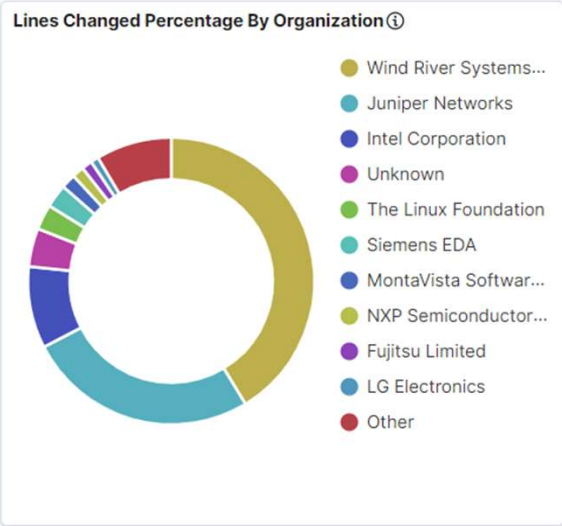
PLATINUM MEMBERS

intel COMCAST arm

cisco facebook XILINX

Microsoft WNDRVR aws





出典:https://insights.lfx.linuxfoundation.org/projects
Screenshot:May27/2021 (Select time range last 6 months)

2021 WIND RIVER, ALL RIGHTS RESERVED

定期セキュリティ監視

CVE数は年々増加

	2014	2015	2016	2017	2018	2019
モニタリング：CVE数	8486	6588	6527	13191	16536	17191
評価：Wind River Linux に対する修正数	1844	2330	5161	9920	6656	6407
修正：サポートしたリリース数	4	4	4	5	5	5
重要なセキュリティ脆弱性案件	CVE-2014-0160 aka Heartbleed	CVE-2015-0235 aka Ghost	CVE - CVE-2016-0800 aka DROWN	Stack-heap Overflow CVE-2017-1000364-66	Spectre / Meltdown CVE-2018-3639-40	Zombieload FallOut RIDL

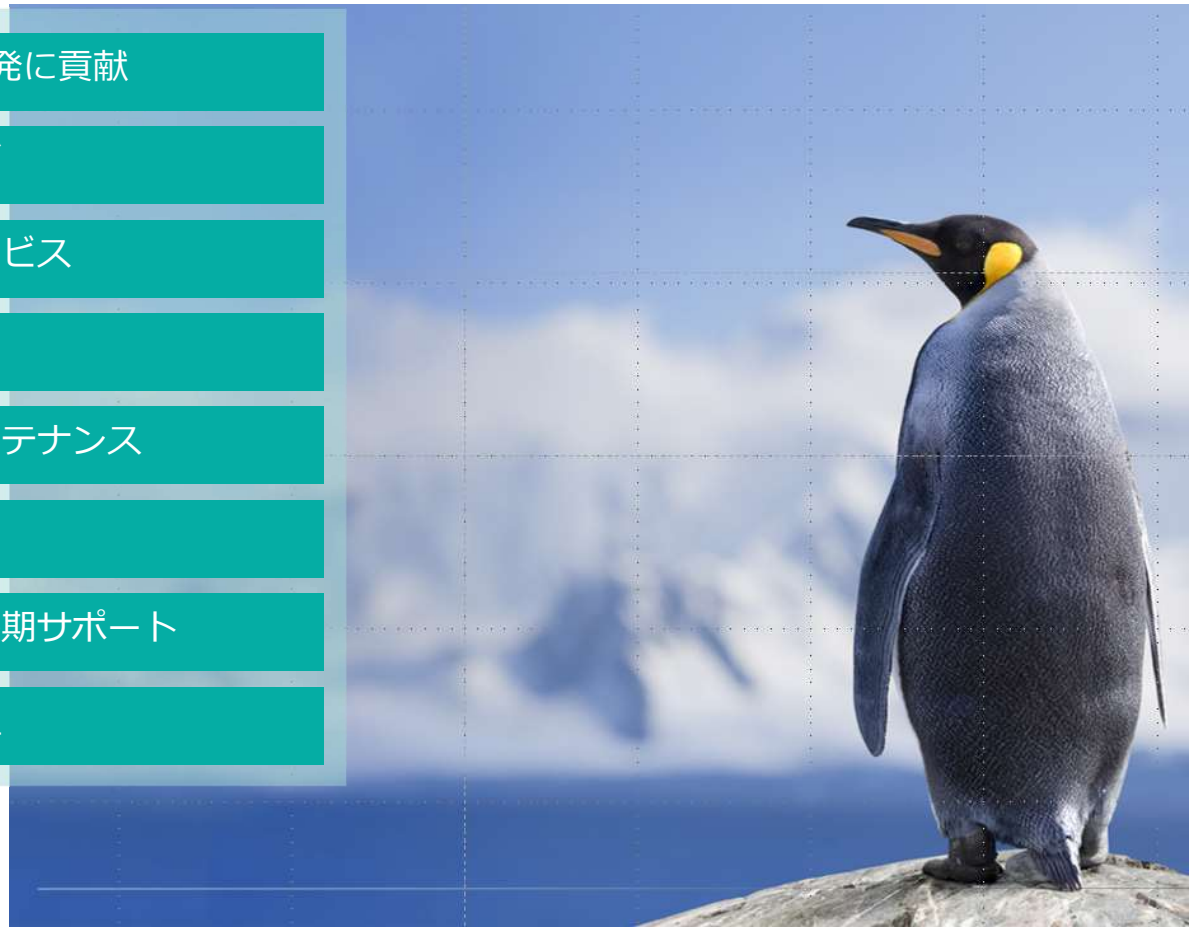
2019年の1日の平均CVEの数 = 47

WNDRVR

2021 WIND RIVER, ALL RIGHTS RESERVED

Wind River Linuxの優位性

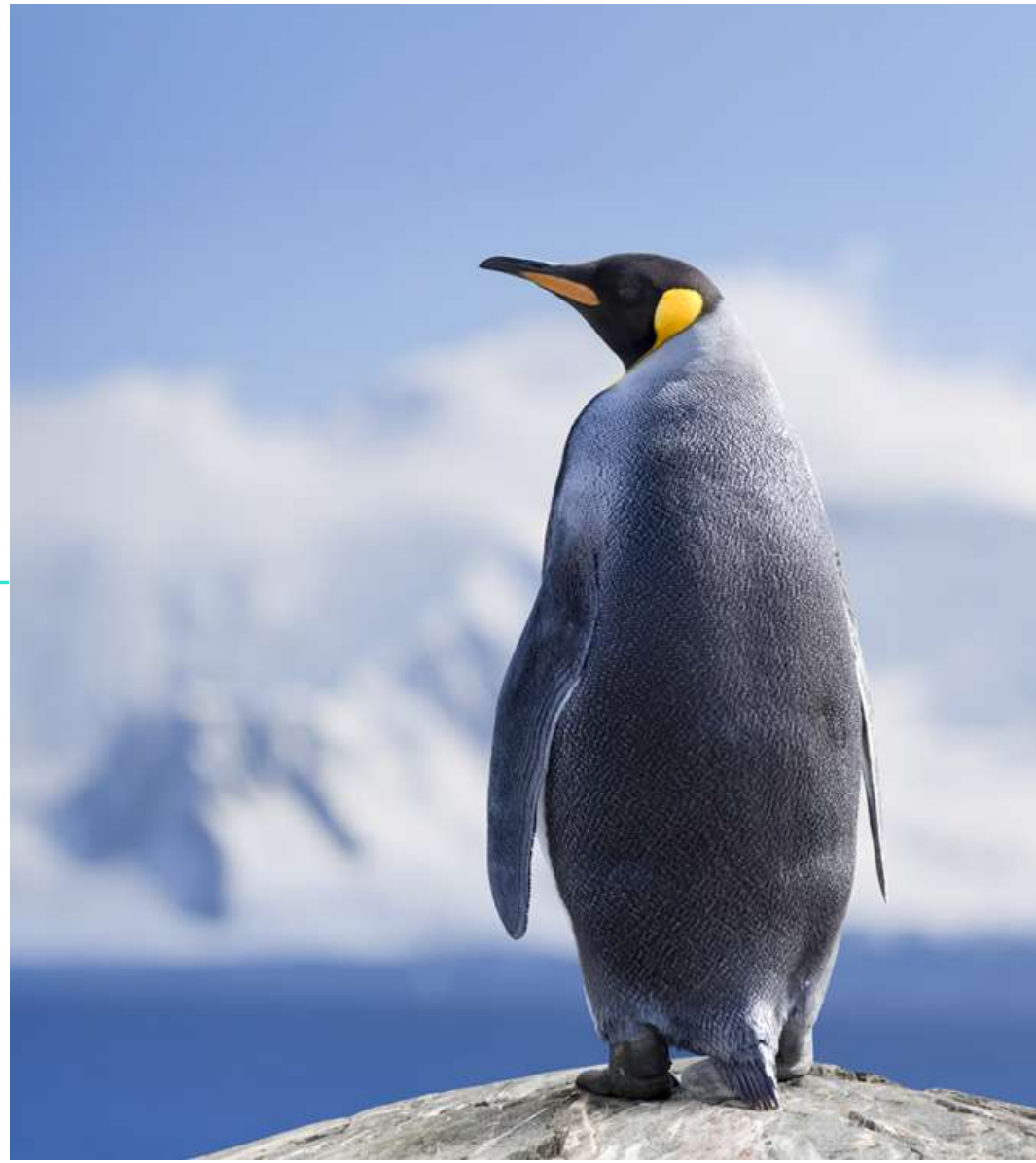
- ソリューションパートナとしてLinux開発に貢献
- 適切なレベルに応じたアウトソーシング
- 検証された製品＋豊富なサポート＆サービス
- セキュリティ脆弱性対応
- グローバルチームによるサポート＆メンテナンス
- 敵的なパッチ提供による品質維持
- お客様の製品ライフサイクルに応じた長期サポート
- ライセンスコンプライアンス・輸出管理



WNDRVR

2021 WIND RIVER, ALL RIGHTS RESERVED

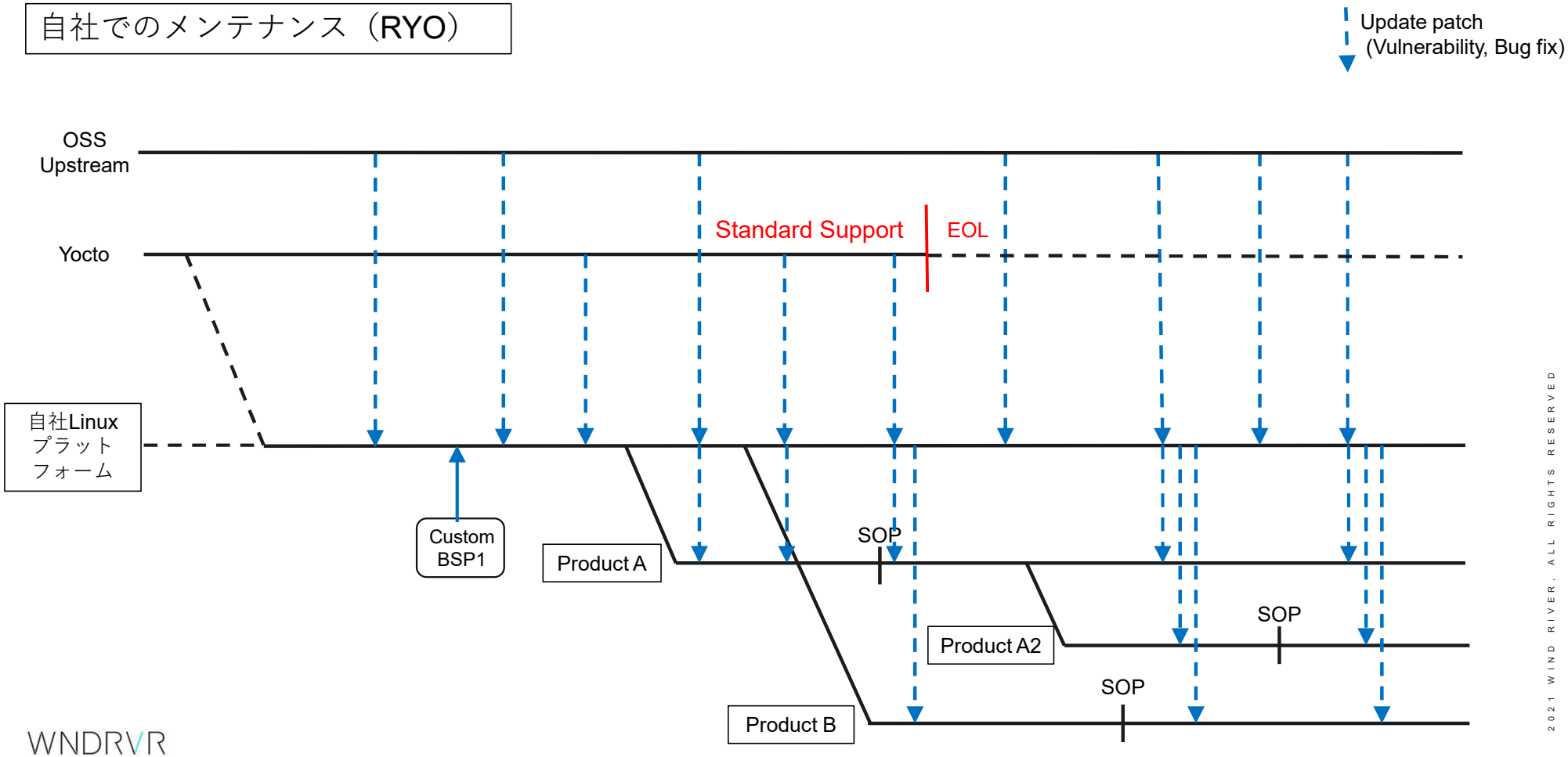
4.
Wind River ソリューション



WNDRVR

Linuxを使用した製品開発におけるメンテナンススキーム

自社でのメンテナンス（RYO）

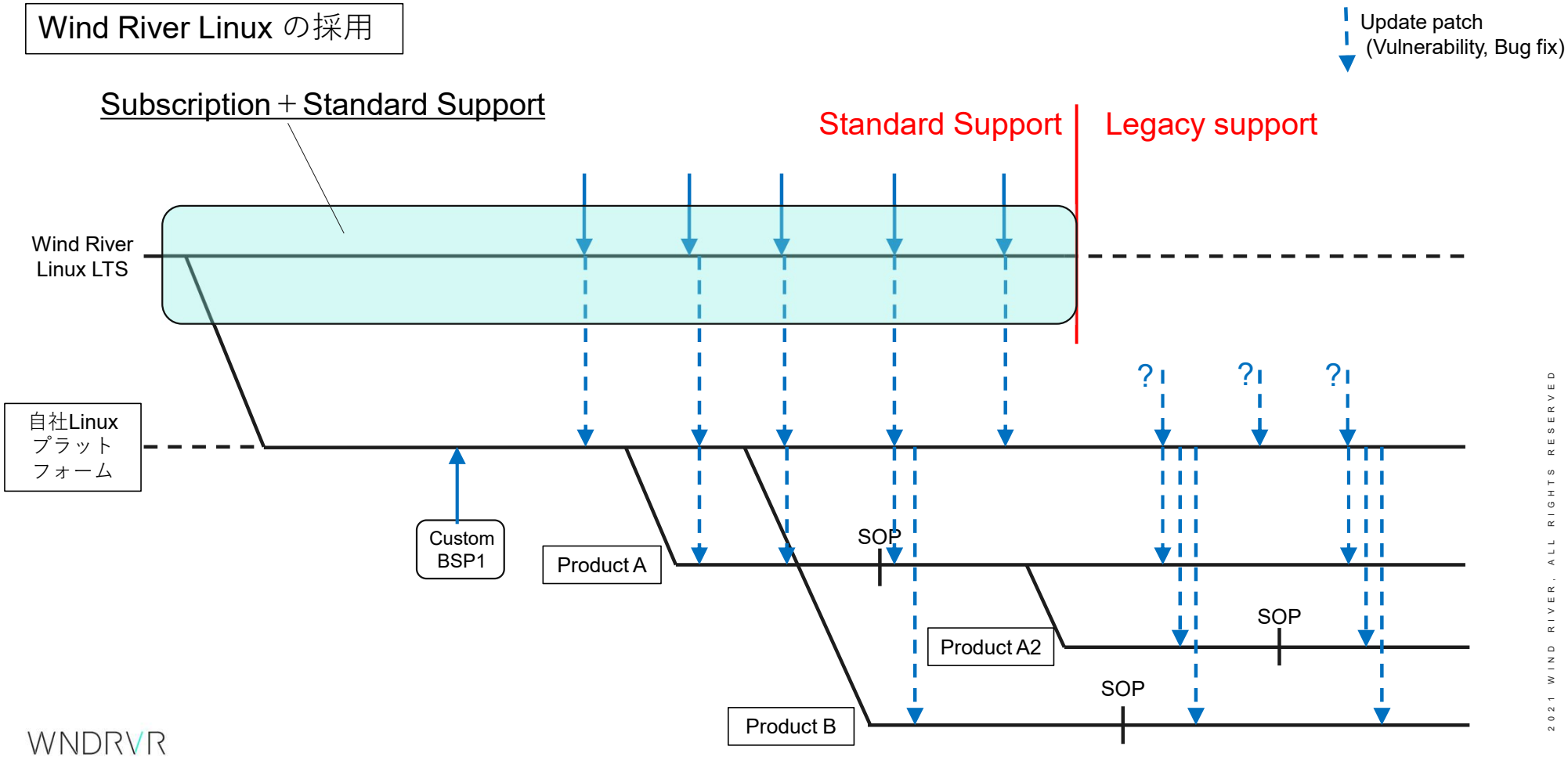


WNDRVR

2021 WIND RIVER, ALL RIGHTS RESERVED

Linux/OSS Branch control (2)

Wind River Linux の採用

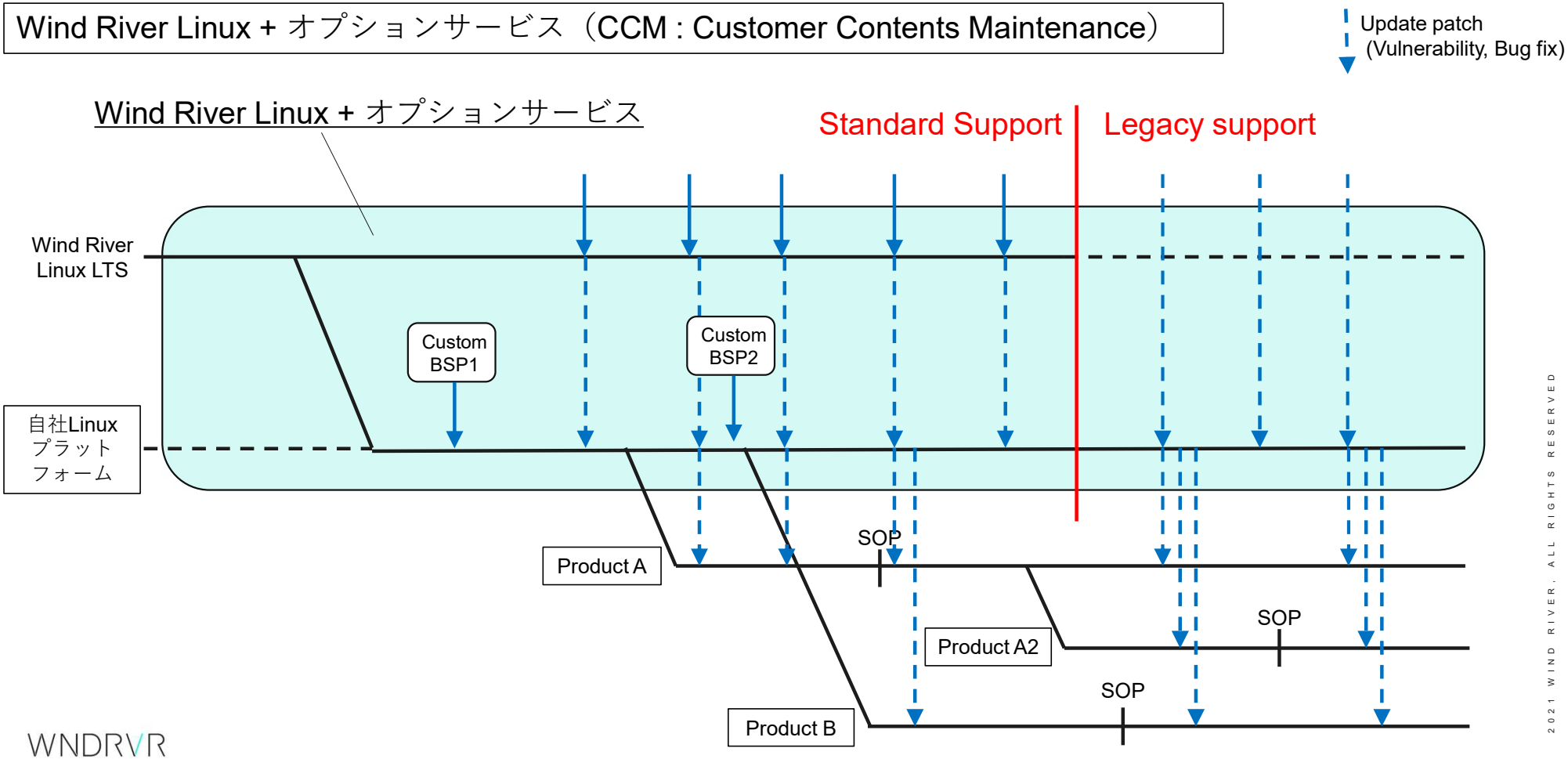


WNDRVR

2021 WIND RIVER, ALL RIGHTS RESERVED

Linux/OSS Branch control (3)

Wind River Linux + オプションサービス（CCM : Customer Contents Maintenance）

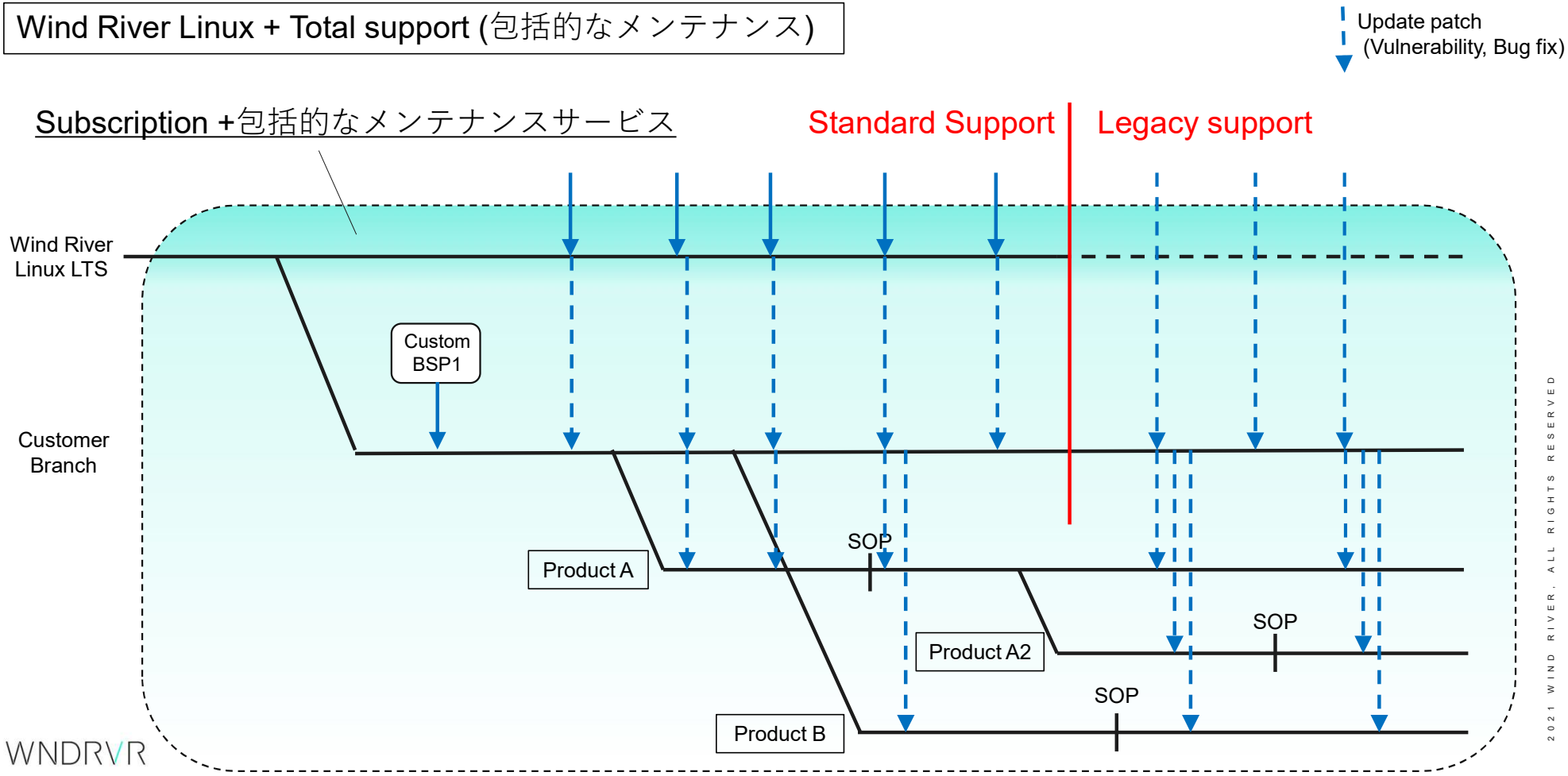


WNDRVR

2021 WIND RIVER, ALL RIGHTS RESERVED

Linux/OSS Branch control (4)

Wind River Linux + Total support (包括的なメンテナンス)

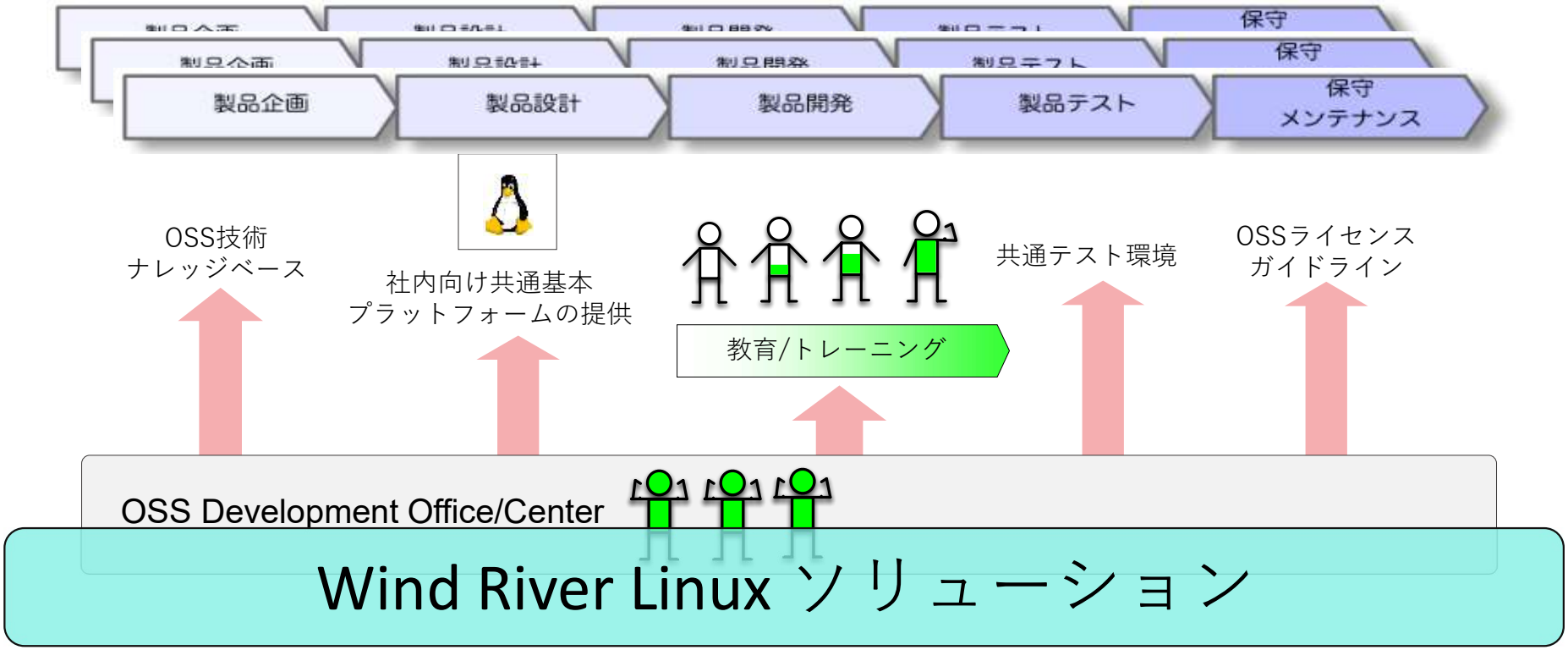


WNDRVR

2021 WIND RIVER, ALL RIGHTS RESERVED

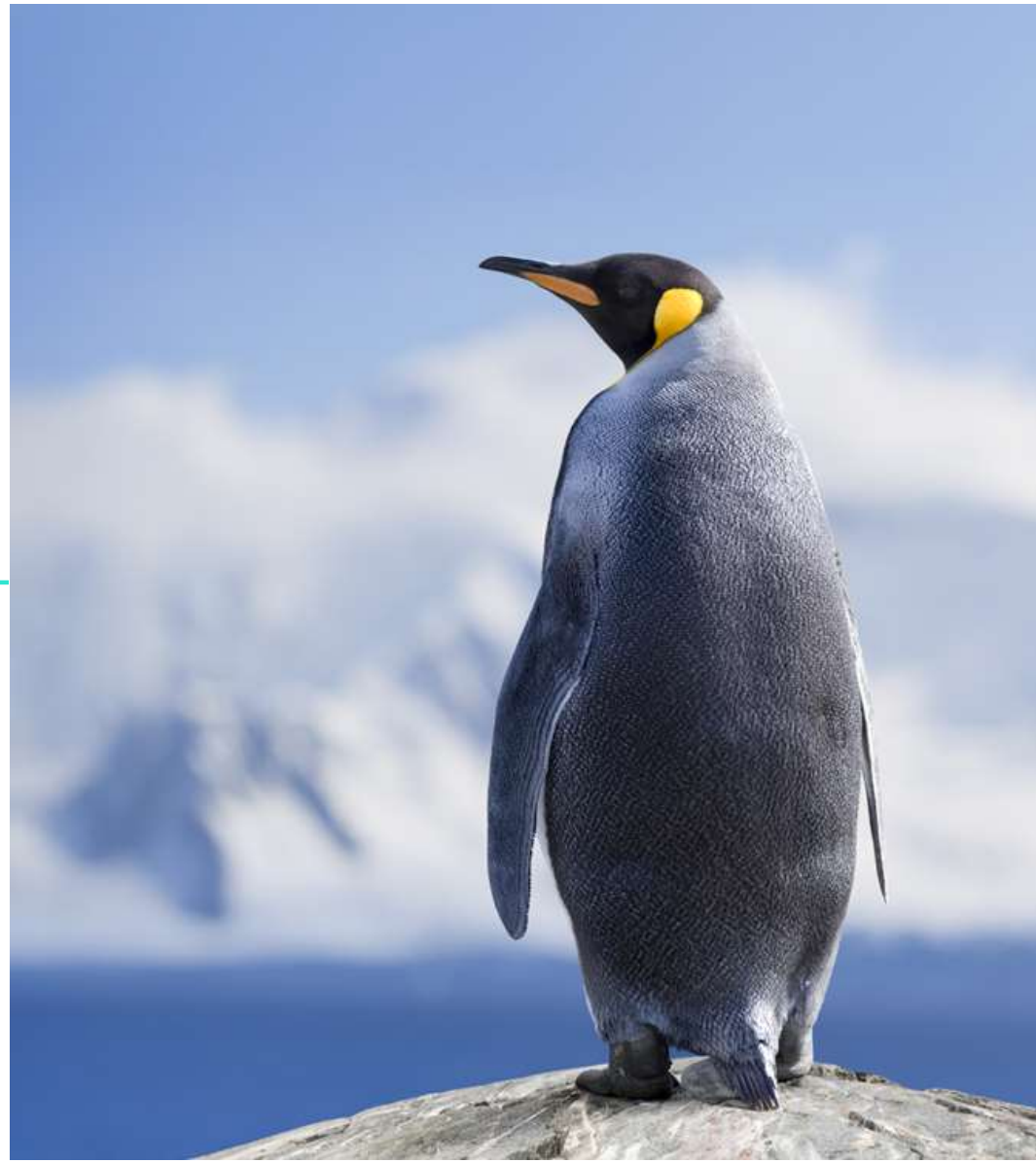
OSSプラットフォーム開発の習熟性

OSベンダーとの協業により競争領域の開発に専念することが可能になります。



2021 WIND RIVER, ALL RIGHTS RESERVED

5.
まとめ



WNDRVR

まとめ

- **組込みシステム／IoTでOSSの使用は不可欠になりつつある**
  - デバイス対応、サポート機能など最新トレンドを追従するにはOSSの利用なしには考えにくい

- **OSSを利用した製品開発での大きなリスクは次の５点**
  - **技術的な達成可能性**
    - まずは評価/検証。手を動かす。　→　OSSのメリットの１つ
  - **品質評価**
    - 自社でやると腹をくくるか？信頼できるパートナーと組むか？
    - CI/CD、DevOps
  - **OSSライセンス**
    - OSSを利用するための組織としてのポリシーが必要
  - **セキュリティ対策**
    - 製品ライフサイクルを考慮した脆弱性対策の制定
    - 製品に搭載している脆弱性の把握と迅速な対応
  - **技術スキル**
    - どのような技術スキルが求められるかを明確化
    - その上で足りないスキルをどのように補うか？ 社内教育　or　アウトソース

- **OSSを利用するメリット　→　全体最適によるコスト削減**

WNDRVR

2021 WIND RIVER, ALL RIGHTS RESERVED

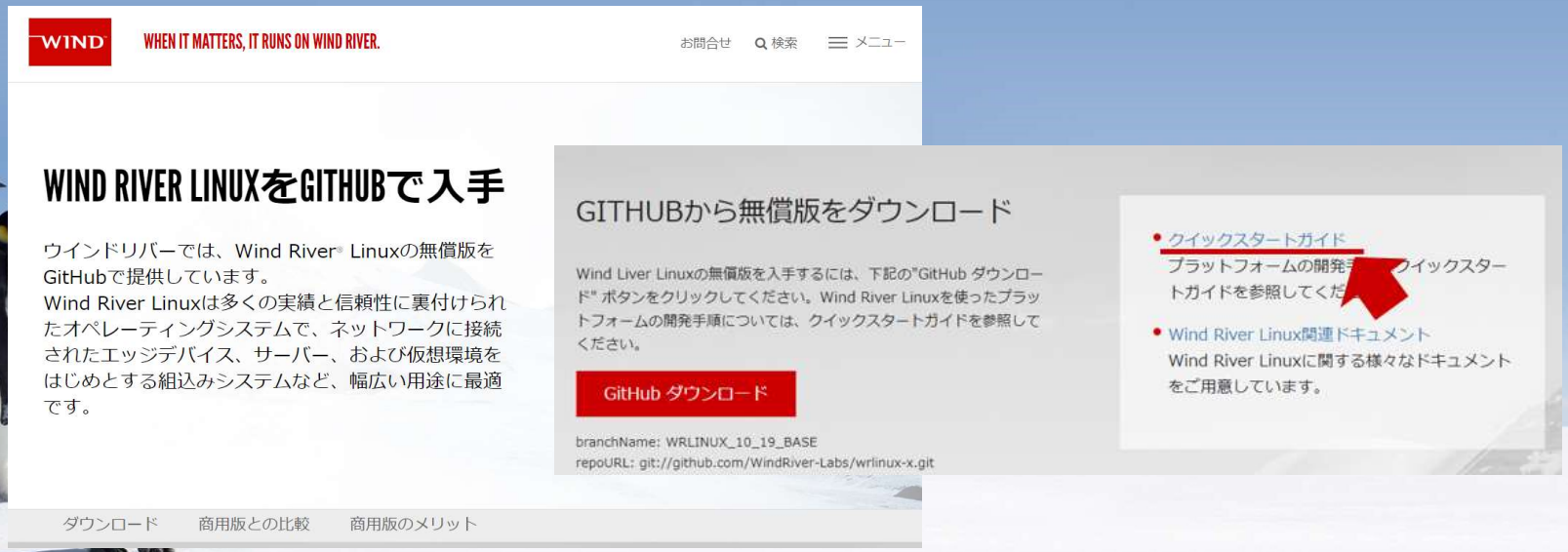
コミュニティ版 WIND RIVER LINUXへのアクセス

GitHubからWind River Linux無償版をダウンロード

https://www.windriver.com/japan/products/linux/download/

お問い合わせはこちらから

www.windriver.com/japan/company/contact



組込みLinux開発における課題と商用ディストリビュータの位置づけ

ウインドリバー株式会社
お問い合わせ
https://www.windriver.com/japan/contact

WNDRVR