DEOS-FY2009-PP-02

オープンシステムディペンダビリティ実現に向けての取り組み

コアチーム代表

東京大学

石川裕

コアチームメンバ(あいうえお順):

1期:追川 修一、加藤 真平、菅谷 みどり、中澤 仁、塙 敏博、藤田 肇、杵渕　雄樹、前田俊行、松田 元彦、三浦 信一

3期:石綿 陽一、倉光 君郎、高村 博紀、松野 裕、山田 浩史、吉田 哲也

発表概要

- 全体概要
  - 課題に対するDEOSの取り組みと中間成果の関係図
  - 貢献
  - 研究開発体制
- コアチームの概要
- D-Core: Open Systems Dependability Core技術

  D-Case/Metrics, Configuration, DS-Bench
  - 開発プロセス時支援
  - 運用時支援
  - 関連研究
- システム安全構築技術
  - **P-Components and P-Bus**
- 第一期採択チーム主要成果概要
- まとめ
  - 貢献
  - 将来課題
- このあとの発表

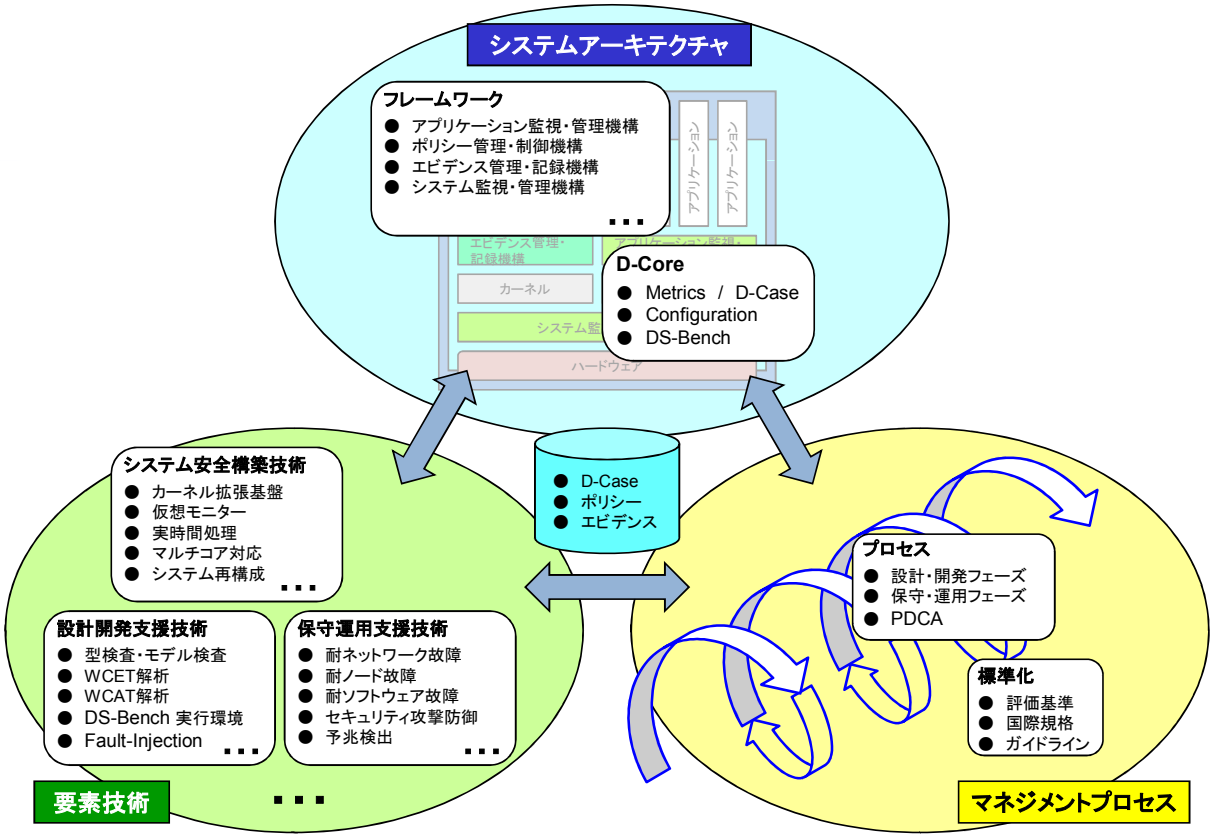
課題に対するDEOSの取り組み

障害要因の最小化（不完全性）

- 要求、仕様、設計、実装、テストの見える化
- 動作解析、振舞い検証
- 動作状況記録、説明責任マネージメント支援
- 国際標準・規格

障害影響の最小化（不確実性）

- 実環境・実時間での仮稼働
- 稼働中の予知
- 障害の最小化、迅速な復旧支
- 動作状況記録、説明責任マネージメント支援



課題に対するDEOSの取り組みの状況

障害要因の最小化（不完全性）

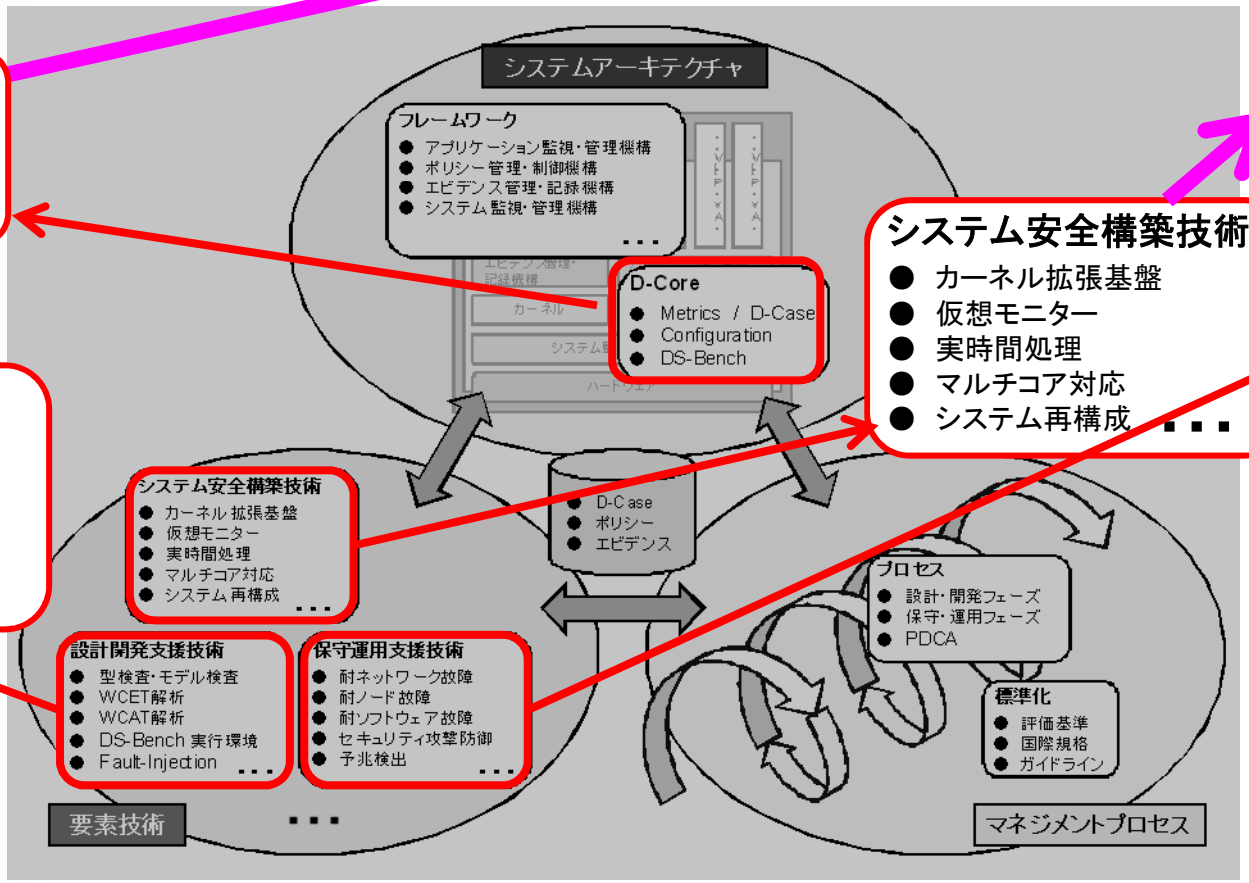
- 要求、仕様、設計、実装、テストの見える化
- 動作解析、振舞い検証
- 動作状況記録、説明責任マネージメント支援
- 国際標準・規格

障害影響の最小化（不確実性）

- 実環境・実時間での仮稼働
- 稼働中の予知
- 障害の最小化、迅速な復旧支
- 動作状況記録、説明責任マネージメント支援

D-Core 技術
- Metrics / D-Case
- Configuration
- DS-Bench

設計開発支援技術
- 型検査・モデル検査
- WCET解析
- WCAT解析
- DS-Bench 実行環境
- Fault-Injection (D-Cloud)
- ・・・



システム安全構築技術
- カーネル拡張基盤
- 仮想モニター
- 実時間処理
- マルチコア対応
- システム再構成
- ・・・

保守運用支援技術
- 耐ネットワーク故障
- 耐ノード故障
- 耐ソフトウェア故障
- セキュリティ攻撃防御
- 予兆検出 ・・・

貢献

対象分野

- 社会インフラ
  - 携帯・組み込み、FA/OA、ロボット、サーバ

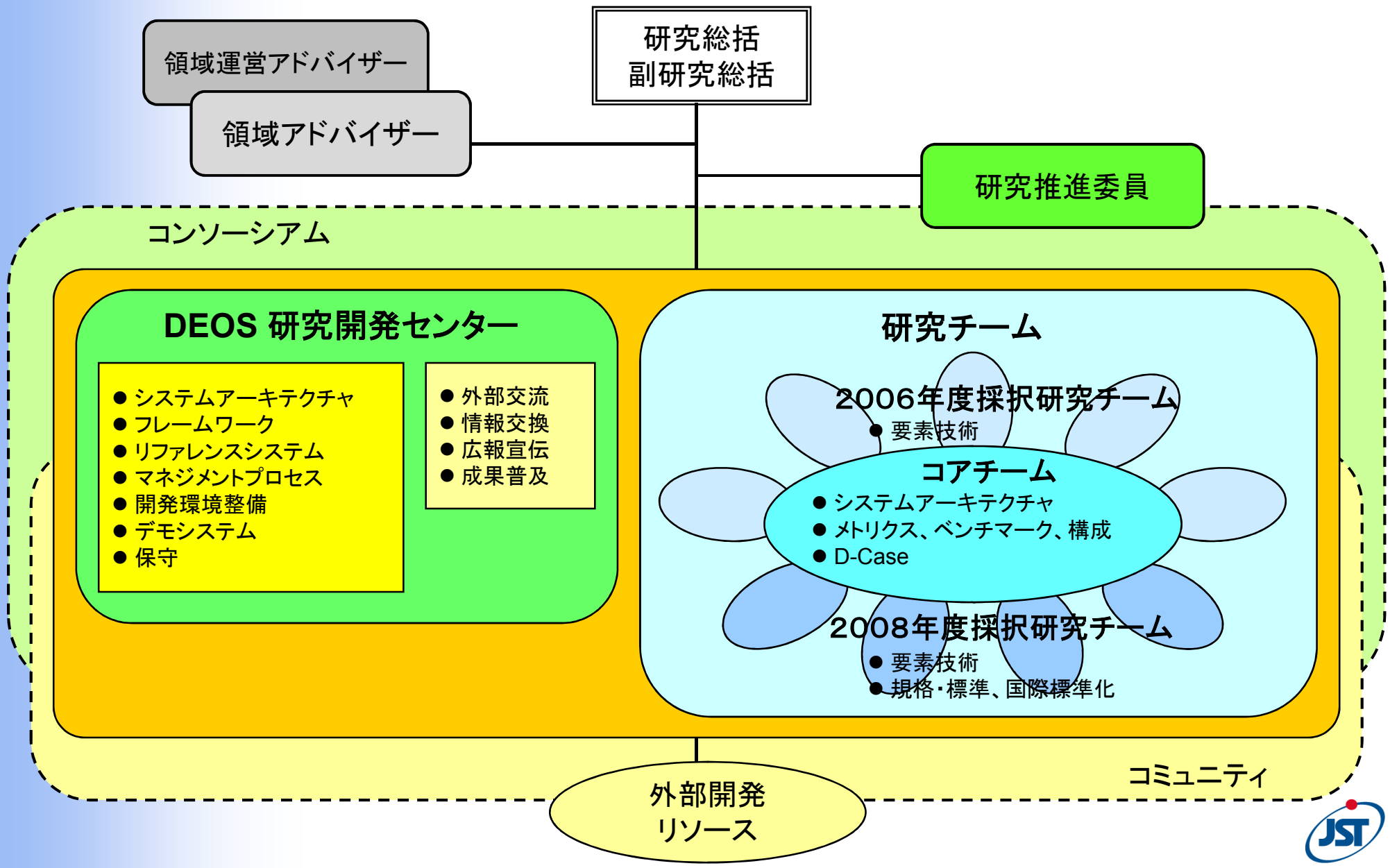
産業界への貢献

- LCC(Life Cycle Cost)低減
  - 開発コスト
  - 保守コスト
- PL(製造責任) 問題
- Value Added(製品付加価値)
- CSR (Cooperate Social Responsibility)
  - 社会的責任:　安心・安全・グリーンIT

学術的貢献

- D-Core: Open Systems Dependability Core技術　(以上コアチーム中心)
- 各研究チームの研究内容

研究開発体制


領域運営アドバイザー
領域アドバイザー
研究総括
副研究総括
研究推進委員
コンソーシアム
DEOS 研究開発センター
● システムアーキテクチャ
● フレームワーク
● リファレンスシステム
● マネジメントプロセス
● 開発環境整備
● デモシステム
● 保守
● 外部交流
● 情報交換
● 広報宣伝
● 成果普及
研究チーム
2006年度採択研究チーム
● 要素技術
コアチーム
● システムアーキテクチャ
● メトリクス、ベンチマーク、構成
● D-Case
2008年度採択研究チーム
● 要素技術
● 規格・標準、国際標準化
コミュニティ
外部開発
リソース

発表概要

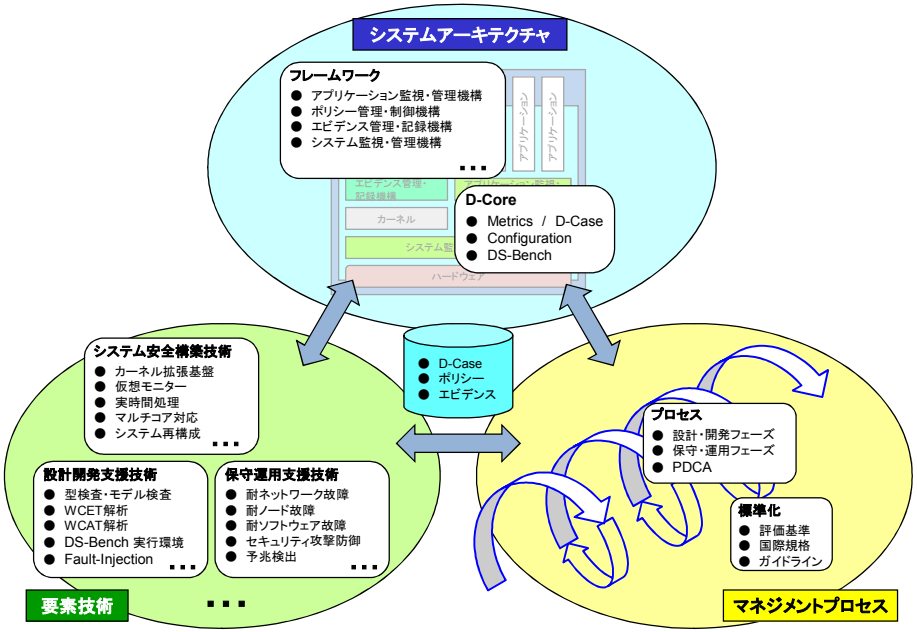
- 全体概要
  - 課題に対するDEOSの取り組みと中間成果の関係図
  - 貢献
  - 研究開発体制
- コアチームの概要
- D-Core: Open Systems Dependability Core技術

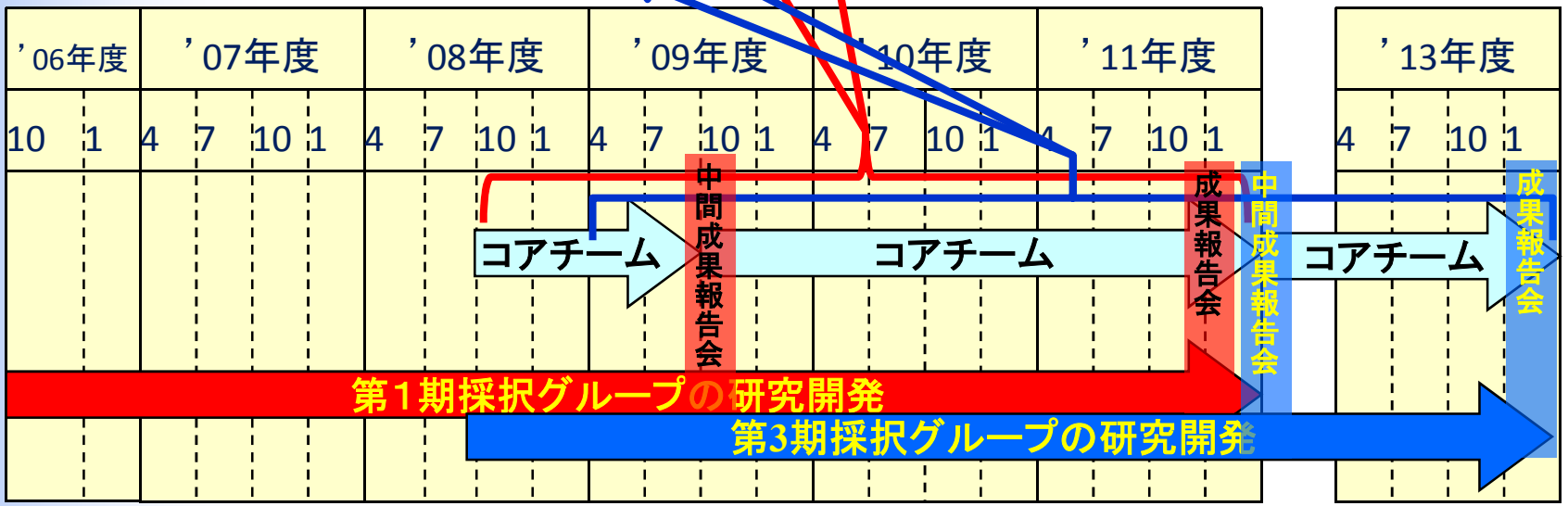
  D-Case/Metrics, Configuration, DS-Bench
  - 開発プロセス時支援
  - 運用時支援
  - 関連研究
- システム安全構築技術
  - P-Components and P-Bus
- 第一期採択チーム主要成果概要
- まとめ
  - 貢献
  - 将来課題
- このあとの発表

コアチームの概要

ミッション

- D-Core: Open Systems Dependability Core
  - 開発・運用支援環境の統合
  - D-Case, Metrics, Configuration, Benchmarks (DS-Bench)
- システム安全構築技術
  - OS要素技術の統合
    - ✓ P-Bus, P-Components
- システムアーキテクチャ
- ディペンダビリティ評価規格





発表概要

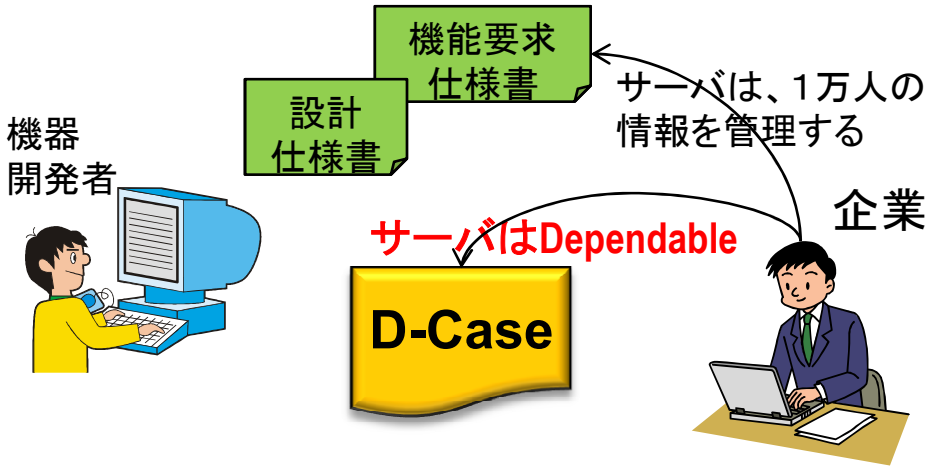
- 全体概要
  - 課題に対するDEOSの取り組みと中間成果の関係図
  - 貢献
  - 研究開発体制
- コアチームの概要
- D-Core: Open Systems Dependability Core技術

  D-Case/Metrics, Configuration, DS-Bench
  - 開発プロセス時支援
  - 運用時支援
  - 関連研究
- システム安全構築技術
  - P-Components and P-Bus
- 第一期採択チーム主要成果概要
- まとめ
  - 貢献
  - 将来課題
- このあとの発表

Open Systems Dependability Coreに基づく機器開発例

- ある製品のディペンダビリティ(Goal)に対して、何を達成していなければならないか以下のメトリクスで、議論（Argument)していく

支援目的	フェーズ	対象機能	阻害要因
■可用性 ■信頼性 ■安全性 ■整合性 ■保全性	■仕様 ■設計 ■実装・単体試験 ■総合試験 ■流通 ■運用 ■保守・更新 ■破棄・再利用	■ファイルシステム ■通信 ■実時間処理 ■電力・性能 ■システム全体	■自然現象 ■人的ミス ■悪意ある攻撃 ■Hardware故障 ■Software故障

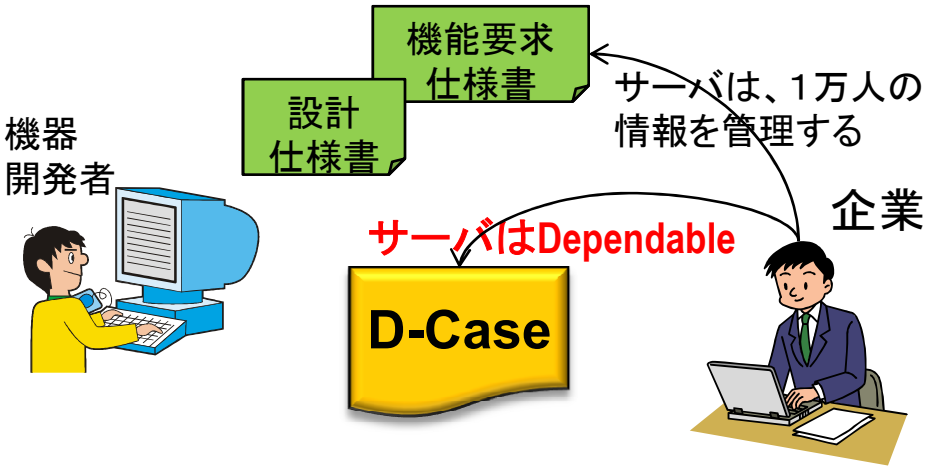


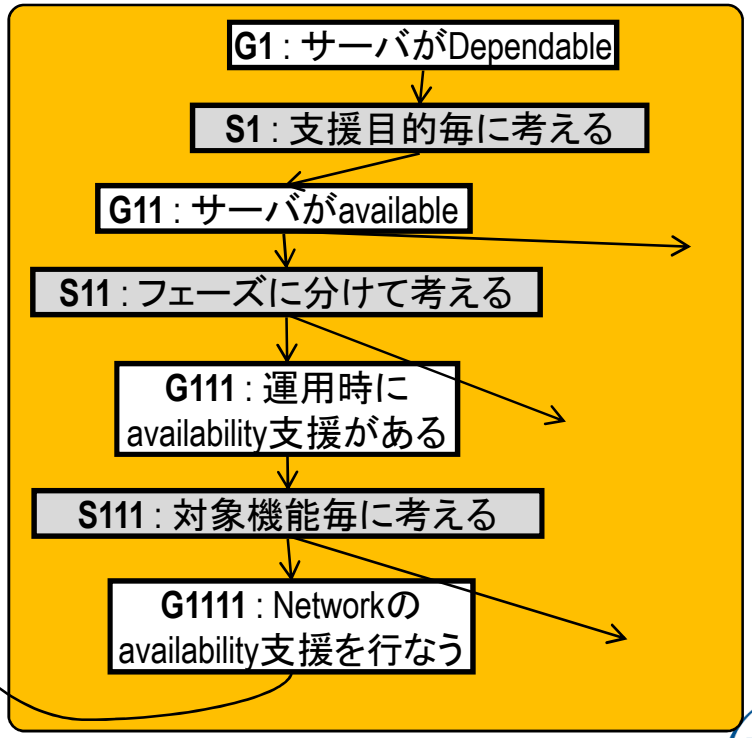
機能要求段階で、ディペンダビリティに関する要求が明確にできない場合、設計と同時に議論を進めていく

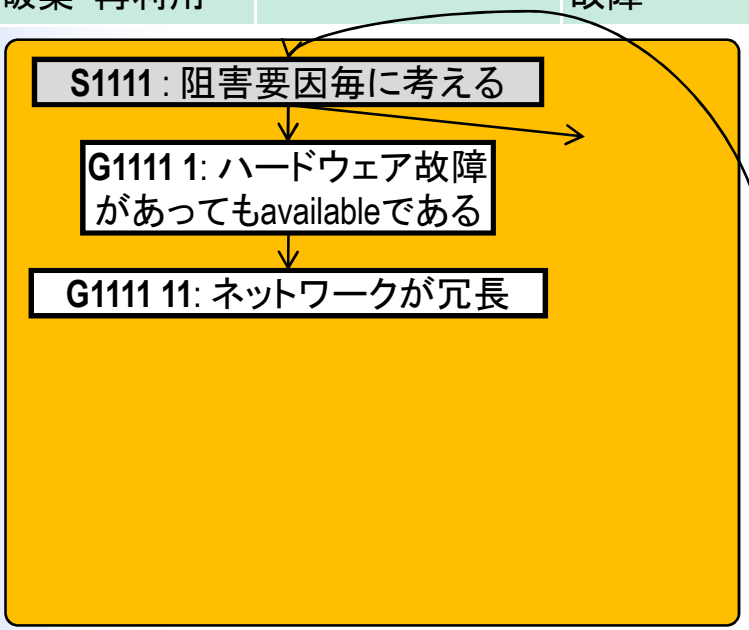
Open Systems Dependability Coreに基づく機器開発例

- ある製品のディペンダビリティ(Goal)に対して、何を達成していなければならないか以下のメトリクスで、議論（Argument)していく

支援目的	フェーズ	対象機能	阻害要因
■可用性 ■信頼性 ■安全性 ■整合性 ■保全性	■仕様 ■設計 ■実装・単体試験 ■総合試験 ■流通 ■運用 ■保守・更新 ■破棄・再利用	■ファイルシステム ■通信 ■実時間処理 ■電力・性能 ■システム全体	■自然現象 ■人的ミス ■悪意ある攻撃 ■Hardware故障 ■Software故障

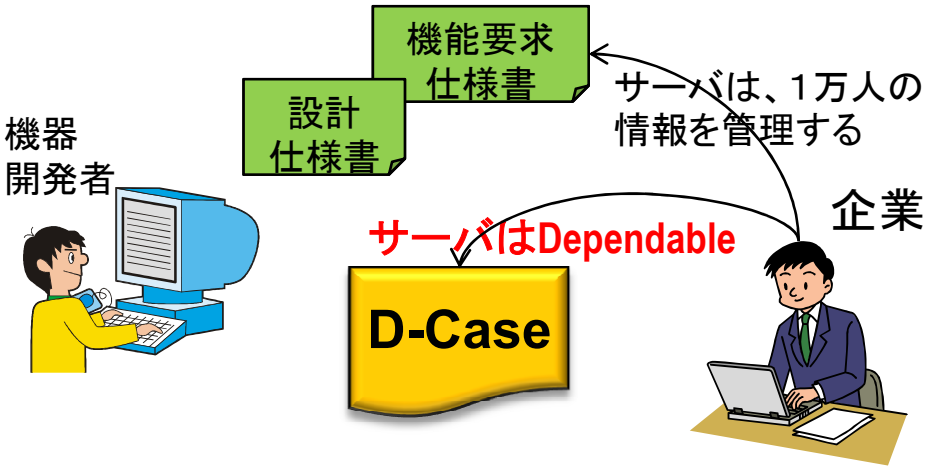


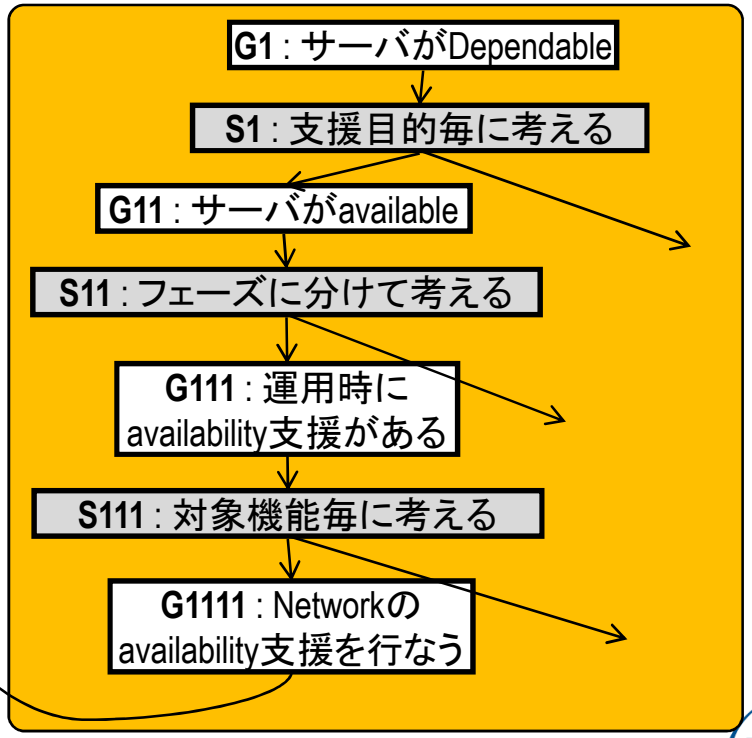


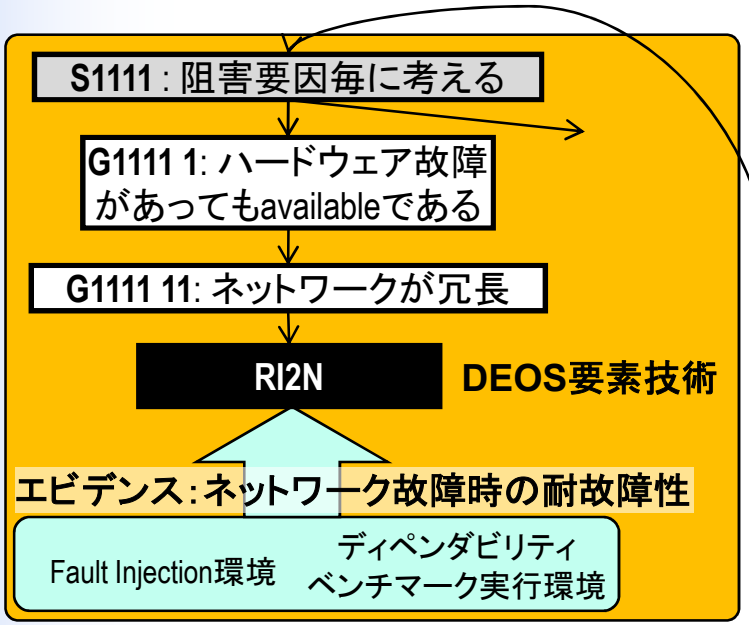


Open Systems Dependability Coreに基づく機器開発例

- ある製品のディペンダビリティ(Goal)に対して、何を達成していなければならないかメトリクス(Metrics)に基づいて議論（Argument)していく
- 必要とされるディペンダビリティ要素技術を選択 (Configuration)
- ディペンダビリティベンチマーク(DS-Bench)によってディペンダビリティ要件が達成されているかどうかの根拠(Evidence)を提示する

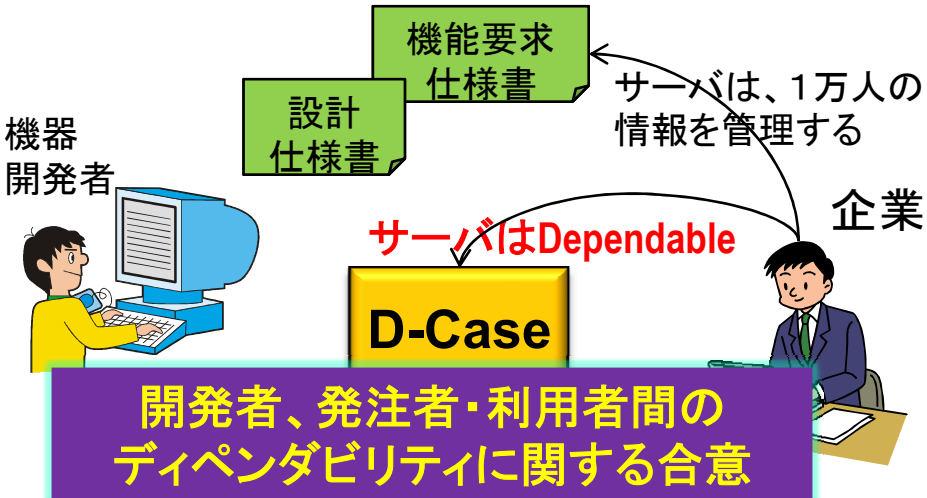


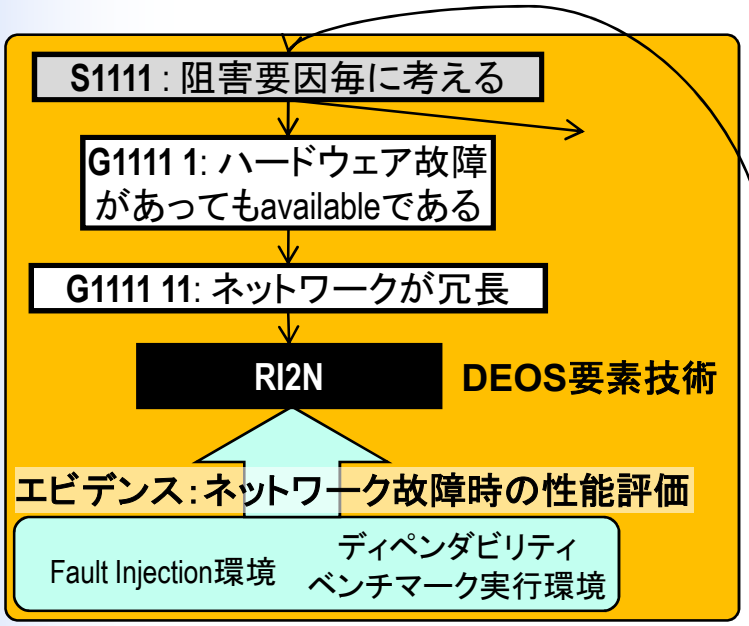




Open Systems Dependability Coreに基づく機器開発例

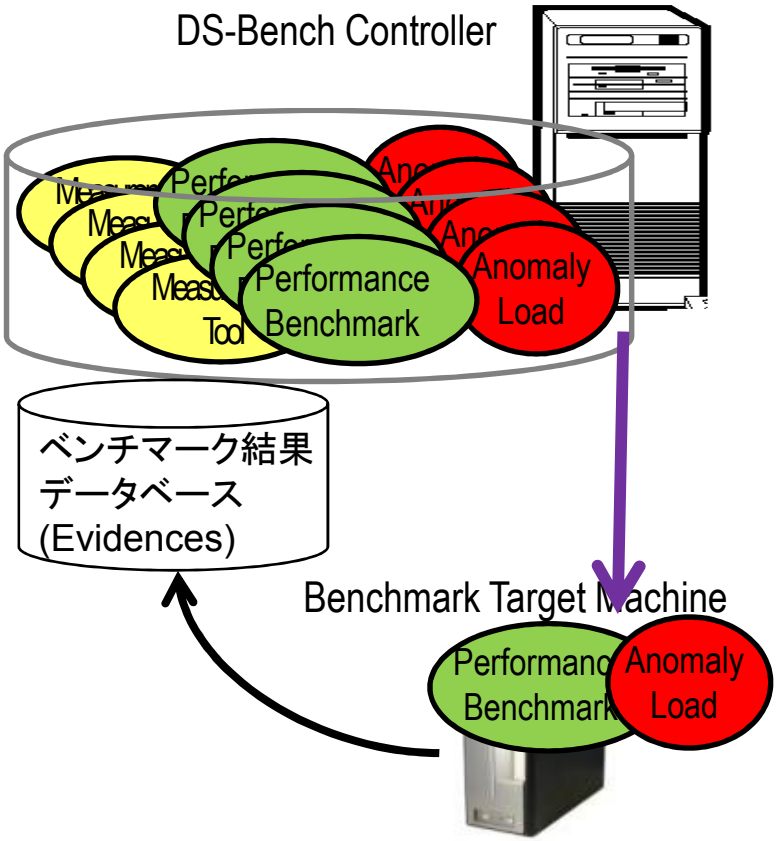
- ある製品のディペンダビリティ(Goal)に対して、何を達成していなければならないかメトリクス(Metrics)に基づいて議論（Argument)していく
- 必要とされるディペンダビリティ要素技術を選択 (Configuration)
- ディペンダビリティベンチマーク(DS-Bench)によってディペンダビリティ要件が達成されているかどうかの根拠(Evidence)を提示する





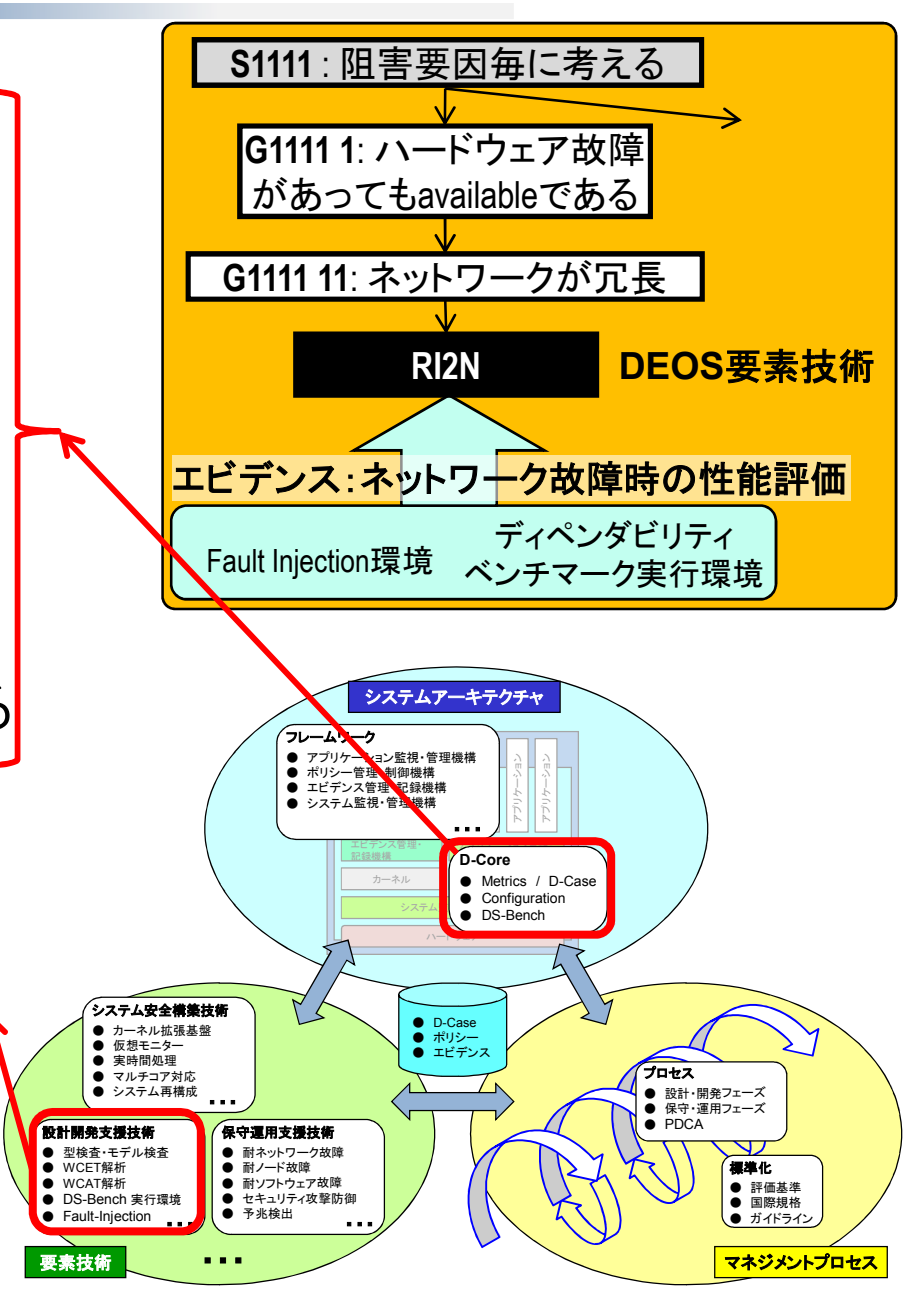
DS-Bench

- 人工的に変則的状態(Anomaly) を発生させ、モジュールあるいはシステム全体の挙動を調べ、D-Caseの根拠 (Evidence) とする
- Anomaly 付加
  - 人工的故障発生(Fault Injections)
    - Power, Memory, Board, Disk, Network
  - 過負荷発生(Overloaded Injections)
  - 人的エラー(Human Error Injections)
- 正常時およびAnomaly付加時の計測項目
  - 故障耐性を持っているか
  - 故障検知時間
  - 回復時間
  - 基本性能の変化
    - 応答時間、スループット、通信性能、実時間性等
  - アプリケーション性能
    - SPEC WEB, TPC-C等
  - 電力消費の推移



D-CaseとEvidence

- D-Case
  - ある製品のディペンダビリティ(Goal)に対して、何を達成していなければならないかメトリクス(Metrics)に基づいて議論（Argument)していく
  - 必要とされるディペンダビリティ要素技術を選択 (Configuration)
  - ディペンダビリティベンチマーク(DS-Bench)によってディペンダビリティ要件が達成されているかどうかの根拠(Evidence)を提示する
- DEOSが提供するEvidence支援要素技術
  - DS-Bench実行環境 (石川チーム)
  - Fault Injection環境 (佐藤チーム)
  - 型検査・モデル検査 (前田チーム)
  - 最悪実行時間予測 (石川チーム)
  - 最悪バッテリ駆動予測 (徳田チーム)



貢献

□ 対象分野

- 社会インフラ
  - 携帯・組み込み、FA/OA、ロボット、サーバ

□ 産業界への貢献

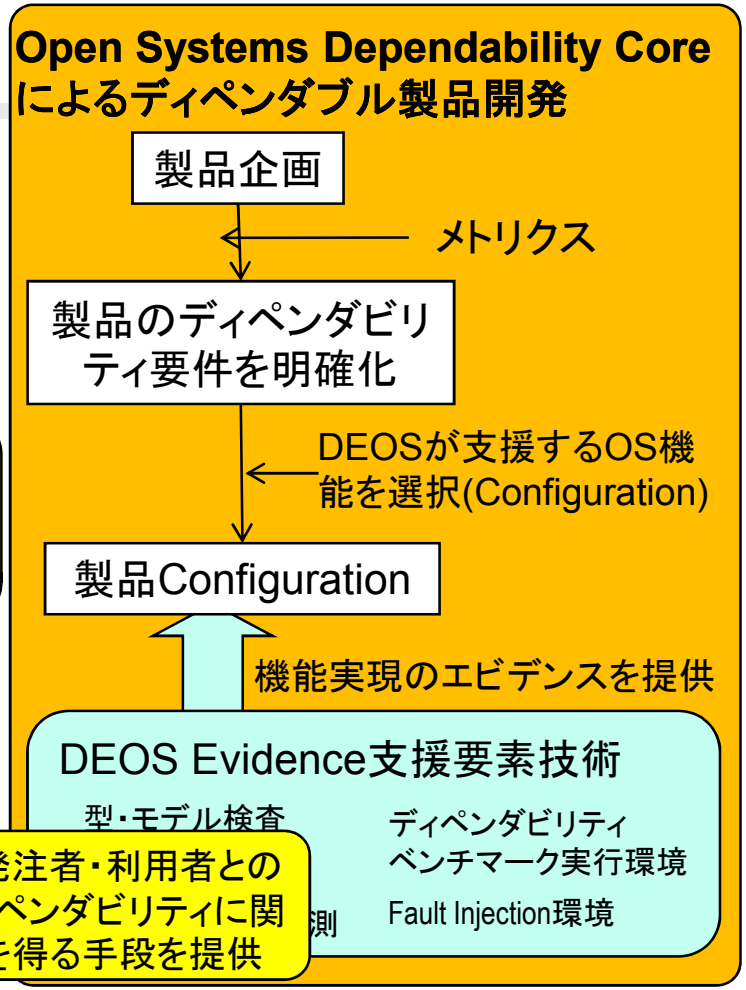
- LCC(Life Cycle Cost)低減
  - 開発コスト
  - 保守コスト
- PL(製造責任) 問題
- Value Added(製品付加価値)
- CSR (Cooperate Social Responsibility)
  - 社会的責任:　安心・安全・グリーンIT

ディペンダビリティ要求と阻害要因を明確にすることにより、適正な部品調達、ソフトウェア仕様によるコスト削減が可能

阻害要因対応技術の検証方法が明確となり、テスト期間の短縮かつ完成度の向上に貢献

製造者、発注者・利用者との間で、ディペンダビリティに関する合意を得る手段を提供

製品のディペンダビリティ検証結果をオープンにすることにより、製品品質のアピール



ユーザ視点でのメリット

□ 複数製品の比較

- 製品間のディペンダビリティを定性的に比較可能（現状）
  - 現状、全体として一つの定量的評価軸を提供しているわけではない
- 製品調達要求仕様、技術審査
  - 必要とされるディペンダビリティの質を検証できる

A製品

阻害要因	支援目的	フェーズ	対象機能
■自然現象 ■人的ミス ■悪意ある攻撃 ■Hardware故障 ■Software故障	■可用性 ■信頼性 ■安全性 ■整合性 ■保全性	■仕様 ■設計 ■実装・単体試験 ■総合試験 ■流通 ■運用 ■保守・更新 ■破棄・再利用	■ファイルシステム ■通信 ■実時間処理 ■電力・性能 ■システム全体

B製品

阻害要因	支援目的	フェーズ	対象機能
■自然現象 ■人的ミス ■悪意ある攻撃 ■Hardware故障 ■Software故障	■可用性 ■信頼性 ■安全性 ■整合性 ■保全性	■仕様 ■設計 ■実装・単体試験 ■総合試験 ■流通 ■運用 ■保守・更新 ■破棄・再利用	■ファイルシステム ■通信 ■実時間処理 ■電力・性能 ■システム全体

発表概要

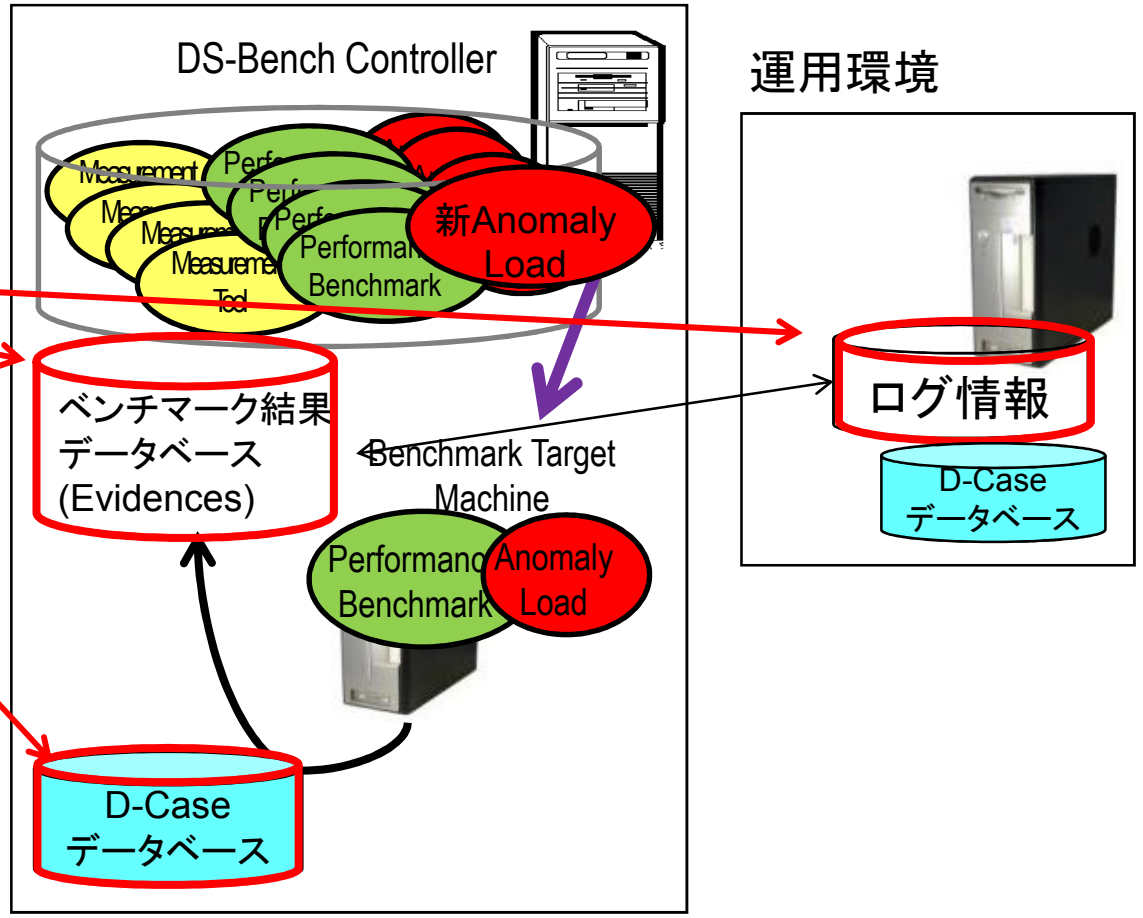
- 全体概要
  - 課題に対するDEOSの取り組みと中間成果の関係図
  - 貢献
  - 研究開発体制
- コアチームの概要
- D-Core: Open Systems Dependability Core技術

  D-Case/Metrics, Configuration, DS-Bench
  - 開発プロセス時支援
  - 運用時支援
  - 関連研究
- システム安全構築技術
  - P-Components and P-Bus
- 第一期採択チーム主要成果概要
- まとめ
  - 貢献
  - 将来課題
- このあとの発表

Open Systems Dependability Coreに基づく保守運用例

□ 未知の障害発生時

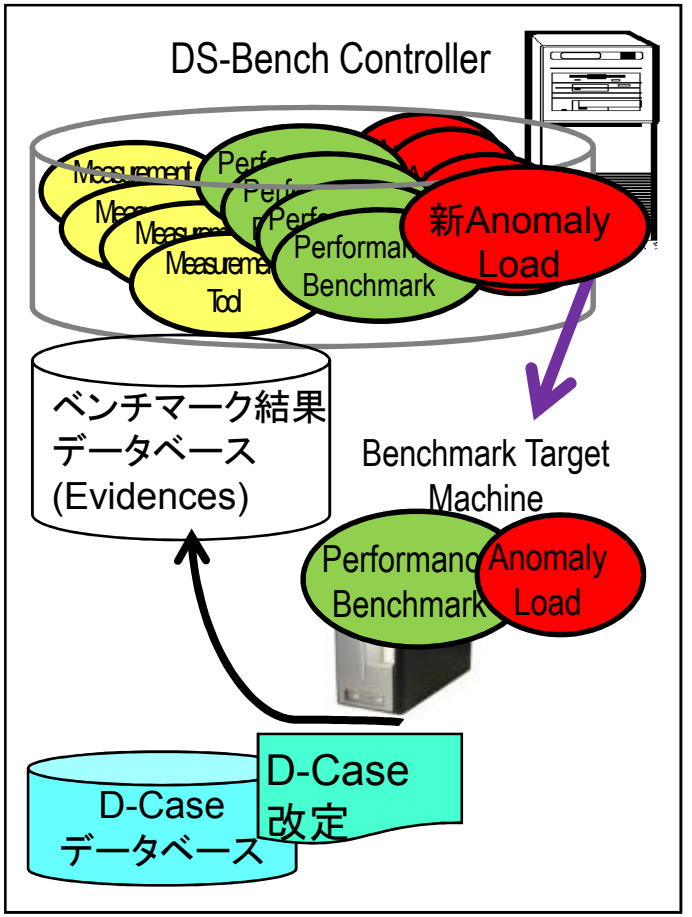
1. ログ情報から、D-Caseをトレースすることにより　どの対応に問題があるかを確認
2. 新規OSモジュールが必要な場合は、P-Componentとして実装
   - P-Componentは、OSモジュールの安全性を保証するモジュールの総称。DEOSで開発されている（後述）
3. D-Caseを改定
4. 発見された障害に基づくAnomaly Loadを作成しDS-Benchに登録
5. 改良システムのディペンダビリティ確認
   - DS-Benchを使用



Open Systems Dependability Coreに基づく保守運用例

ロ バージョンアップ時

1. 機能追加に対する想定障害をリストアップし、対応機構を検討する
2. 新規OSモジュールが必要な場合は、P-Componentとして実装
   - P-Componentは、OSモジュールの安全性を保証するモジュールの総称。DEOSで開発されている（後述）
3. D-Caseを改定
4. 追加された想定障害に対応したAnomaly Loadを作成しDS-Benchに登録
5. 改良システムの安全確認
   - DS-Benchを使用



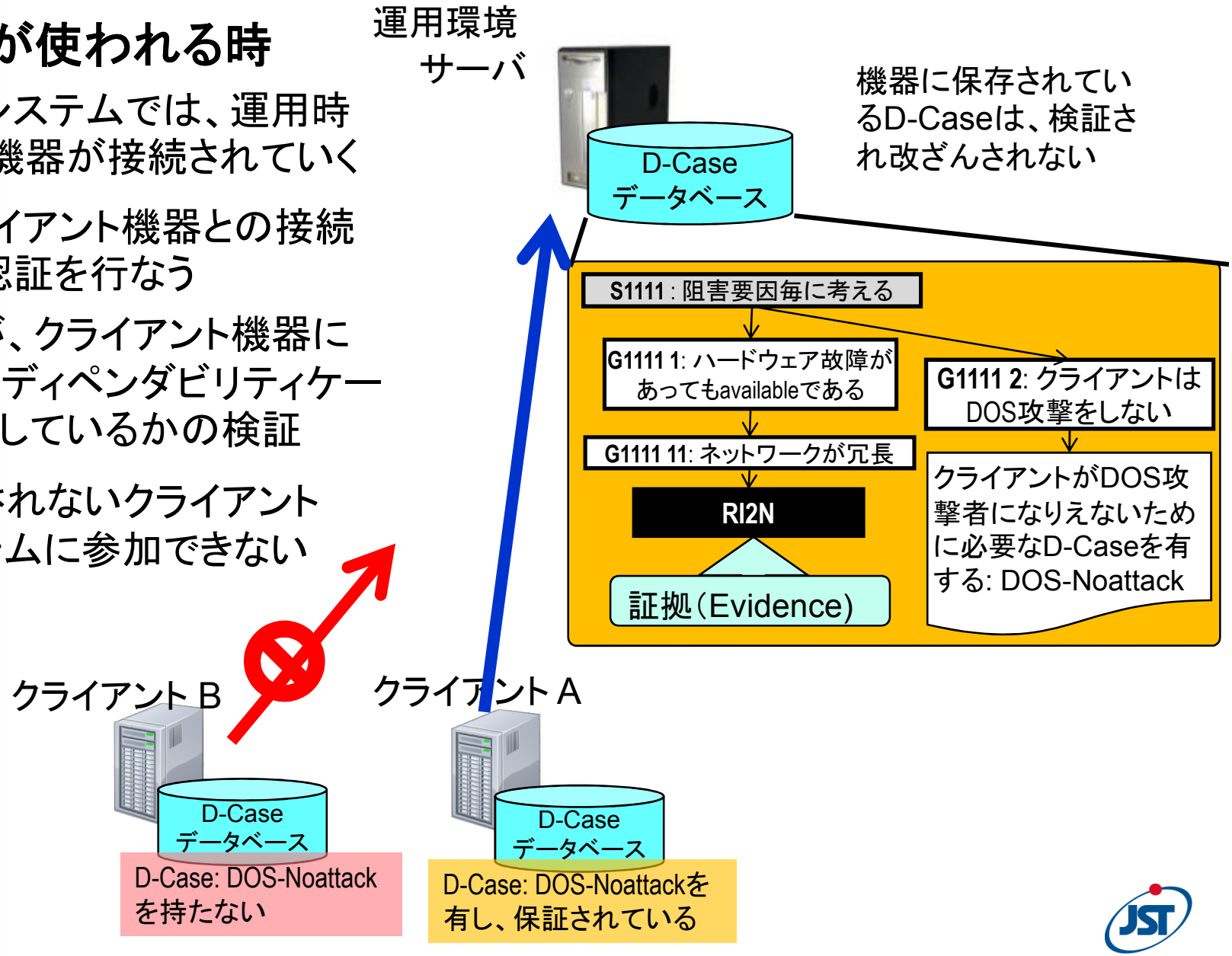
運用環境

Open Systems Dependability Coreに基づく保守運用例

□ 新しい機器が使われる時

- オープンシステムでは、運用時に様々な機器が接続されていく

1. サーバとクライアント機器との接続時にD-Case認証を行なう
   - サーバが、クライアント機器に要求するディペンダビリティケースを保持しているかの検証
2. D-Case認証されないクライアント機器はシステムに参加できない



貢献

□ 対象分野

- 社会インフラ
  - 携帯・組み込み、FA/OA、ロボット、サーバ

□ 産業界への貢献

- LCC(Life Cycle Cost)低減
  - 開発コスト
  - 保守コスト
- PL(製造責任) 問題
- Value Added(製品付加価値)
- CSR (Cooperate Social Responsibility)
  - 社会的責任:　安心・安全なグリーンIT

ディペンダビリティ要求と阻害要因を明確にすることにより、適正な部品調達、ソフトウェア仕様によるコスト削減が可能

阻害要因対応技術の検証方法が明確となり、テスト期間の短縮かつ完成度の向上に貢献

故障の早期発見、D-Caseの成長によるディペンダビリティ向上、DS-Bench充実によるテスト環境充実による

製造者、発注者・利用者との間で、ディペンダビリティに関する合意を得る手段を提供

製品のディペンダビリティ検証結果をオープンにすることにより、製品品質のアピール

ネットワーク上の機器が変更、追加されたとき、それら機器に対するディペンダビリティ要件が満たされているか自動的に検査可能。

発表概要

- 全体概要
  - 課題に対するDEOSの取り組みと中間成果の関係図
  - 貢献
  - 研究開発体制
- コアチームの概要
- D-Core: Open Systems Dependability Core技術

  D-Case/Metrics, Configuration, DS-Bench
  - 開発プロセス時支援
  - 運用時支援
  - 関連研究
- システム安全構築技術
  - P-Components and P-Bus
- 第一期採択チーム主要成果概要
- まとめ
  - 貢献
  - 将来課題
- このあとの発表

Open Systems Dependability Core関連技術 (1/2)

- Dependability Case
  - 開発者が、システムのディペンダビリティ(Goal)に対して、何を達成したかを、議論（Argument）しながら実験や根拠（Evidence）によって示し、stakeholderと合意を得るための手法
  - D-CaseはDependability Caseが基になっているが、以下の新規性がある
    - 要素技術ノードを導入し、要素技術の組み合わせ（Configuration）を議論できる
    - メトリクスの定性評価項目（ディペンダビリティ属性、ライフサイクル、対象機能、阻害要因）と対応させ、開発者とstakeholderが共通の議論＆合意形成可能
    - 根拠(Evidence)として系統的評価環境DS-Benchがある
    - データベース管理され更新し続ける。実行時にも各機器にD-Caseデータベースを保持することにより自動的にディペンダビリティ要件が満たされているか検査可能
- リスク評価手法: FMEA (Failure mode and effects analysis)、 FTA (Fault tree analysis)、ETA (Event tree analysis)、Hazard Matrix、R-Map (Risk Map)など
  - Evidenceとして用いることなどが出来る。
    - 例：要素技術Xが障害Yに対応できることを、トップ事象をYとしたFTA解析をEvidenceとして使って示す。

Open Systems Dependability Core関連技術 (2/2)

- ゴール指向要求分析（ゴールを設定し、詳細化するトップダウン）手法
  - NFR(Non Functional Requirements) Framework(トロント大学、カナダ)
    - 明確でない非機能要件を具体的な達成手段まで詳細化する手法
    - 要件を分析するのであり、evidenceによる説得を目的としてはいないが、分析手法は応用できると考えられる
    - 日本では、IPA/SEC 非機能要件とアーキテクチャWGで検討されている
  - ATAM(Architecture Tradeoff Analysis Method) (SEI, カーネギーメロン大学、アメリカ)
    - システムのサービスの質を高める品質特性（ 可用性、パフォーマンスなど）から、アーキテクチャの設計を決める手法
    - Utility Treeと呼ばれる、システムの品質特性を詳細化していき、最後に詳細化された特性に対するシナリオを記述する（例：可用性に対して、「ネットワーク障害発生後１．５分以内に復旧する」、パフォーマンスに対して「リアルタイムにビデオデータを配信する」など）手法を用いる
    - NFR Frameworkと同様、evidenceの提示を目的としていないが、シナリオ記述によるstakeholderへのわかりやすい説明など、参考になる点があると考えられる
- ISO 9126 Software Engineering Product Qualityなどでの品質特性分類
  - 現在D-Caseは古典的なディペンダビリティ属性（可用性、信頼性、...）のみを考えているが、ISO 9126の品質特性分類（使用性、効率性、移植性、...）などとの組み合わせは今後の検討課題である
  - さらに、開放性ディペンダビリティ属性を考える必要がある
  - 日本では、非機能要件グレード検討会などで、非機能要件の分類に関する検討が行われている

課題に対するDEOSの取り組みの状況

障害要因の最小化（不完全性）

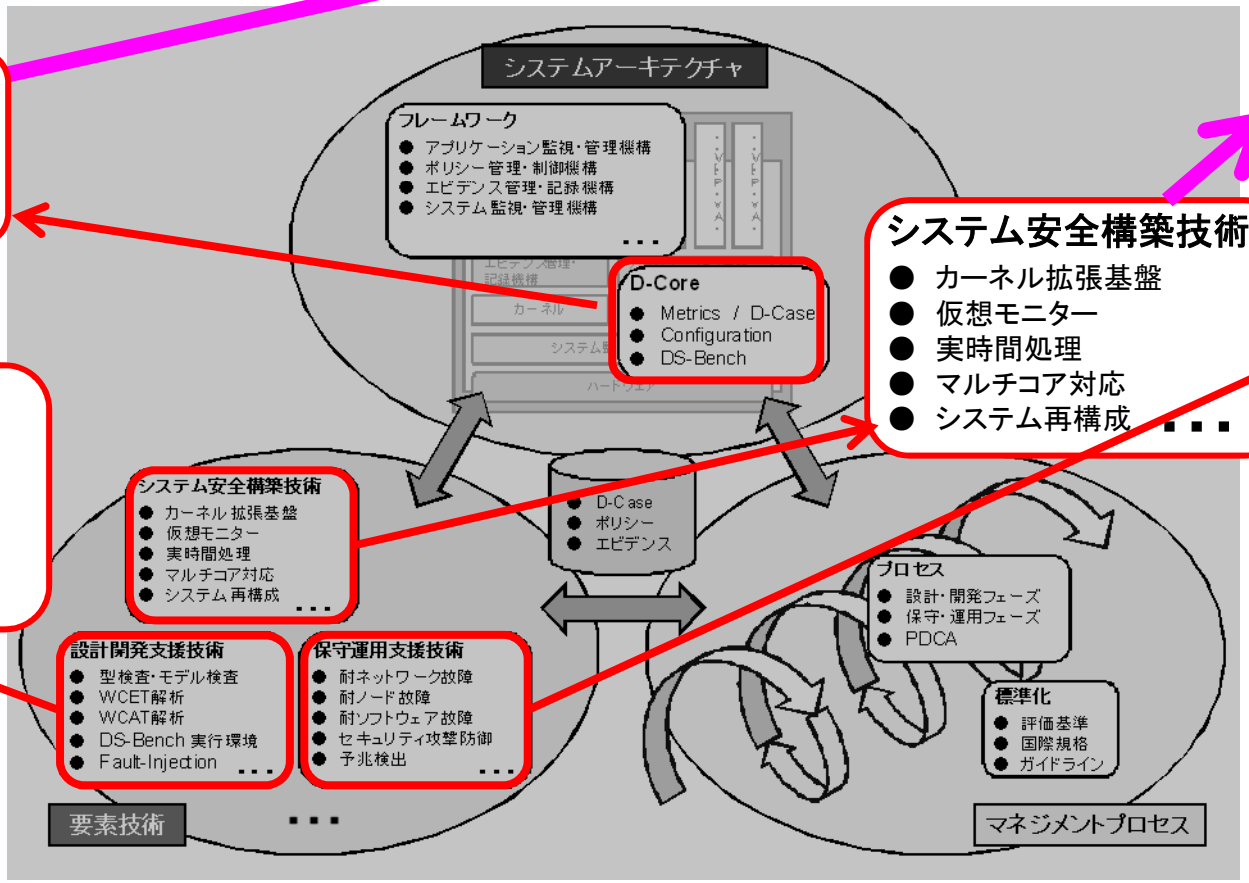
- 要求、仕様、設計、実装、テストの見える化
- 動作解析、振舞い検証
- 動作状況記録、説明責任マネージメント支援
- 国際標準・規格

障害影響の最小化（不確実性）

- 実環境・実時間での仮稼働
- 稼働中の予知
- 障害の最小化、迅速な復旧支
- 動作状況記録、説明責任マネージメント支援

D-Core 技術
- Metrics / D-Case
- Configuration
- DS-Bench

設計開発支援技術
- 型検査・モデル検査
- WCET解析
- WCAT解析
- DS-Bench 実行環境
- Fault-Injection (D-Cloud)
- ・・・



システム安全構築技術
- カーネル拡張基盤
- 仮想モニター
- 実時間処理
- マルチコア対応
- システム再構成
- ・・・

保守運用支援技術
- 耐ネットワーク故障
- 耐ノード故障
- 耐ソフトウェア故障
- セキュリティ攻撃防御
- 予兆検出 ・・・

発表概要

- 全体概要
  - 課題に対するDEOSの取り組みと中間成果の関係図
  - 貢献
  - 研究開発体制
- コアチームの概要
- D-Core: Open Systems Dependability Core技術

  D-Case/Metrics, Configuration, DS-Bench
  - 開発プロセス時支援
  - 運用時支援
  - 関連研究
- システム安全構築技術
  - P-Components and P-Bus
- 第一期採択チーム主要成果概要
- まとめ
  - 貢献
  - 将来課題
- このあとの発表

システム安全構築技術

OSによるディペンダビリティ支援機構課題

- 既知の障害要因対応
  - 耐故障（Fault Tolerance）とQOS
    - アプリケーションに対して故障を気付かせず、故障時QOS制御
  - 耐脅威
- 未知の障害要因対応
  - 障害要因の特定
  - 新発見された障害要因に対応するOS支援機構を開発
  - OS自体の故障（バグ）回避

アドホックにOSを改造していくのはOS自体の信頼性低下につながる

DEOSアプローチ

安全にOSを拡張できるシステム安全構築技術を確立

システム安全構築技術
- カーネル拡張基盤
- 仮想モニター
- 実時間処理
- マルチコア対応
- システム再構成
- ・・・

その上でディペンダビリティを向上させる保守運用支援技術を提供する

保守運用支援技術
- 耐ネットワーク故障
- 耐ノード故障
- 耐ソフトウェア故障
- セキュリティ攻撃防御
- 予兆検出 ・・・

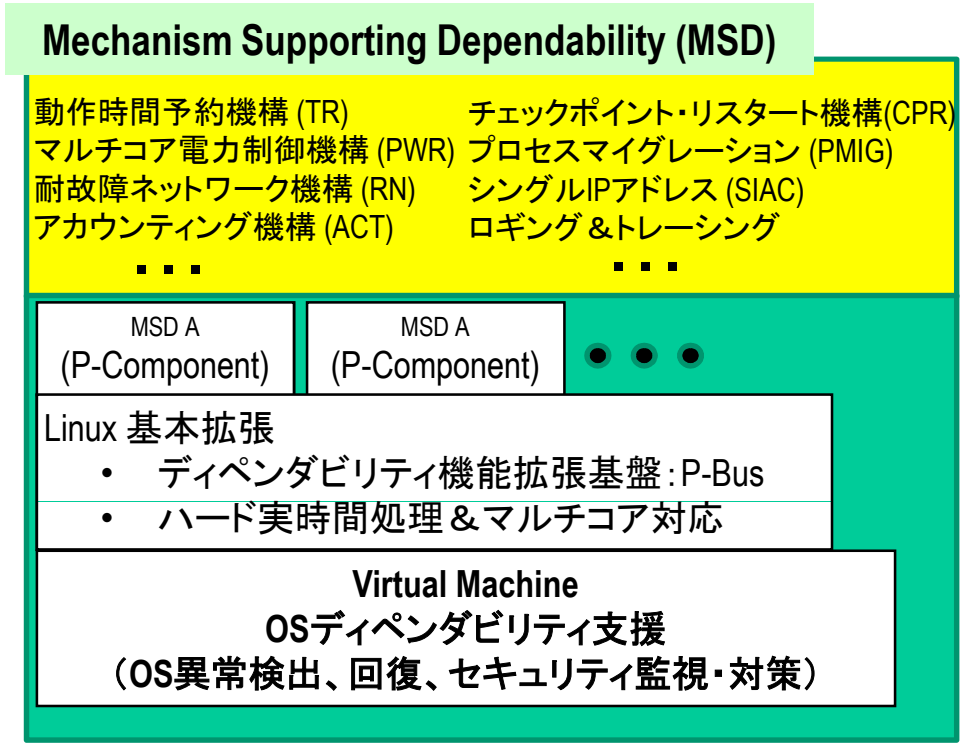
システム安全構築技術：P-Bus, P-Components, and VM

- ✓ Linuxを改良
  - ➢ ディペンダビリティ機能拡張基盤

    P-Bus, P-Component

    1. 安全にOSを拡張できる基盤提供
    2. 拡張基盤上に保守運用支援機構を実現
       - 実行時耐故障機構
       - モニタリング＆ロギング＆トレース機構
  - ➢ P-Busで対応できないOSカーネルの根本を変更する部分はLinuxを直接変更
    - ➢ 実時間処理＆マルチコア対応
- ✓ Virtual Machine
  - ➢ OSディペンダビリティ支援

    OS異常検出、回復



システム安全構築技術
- カーネル拡張基盤
- 仮想モニター
- 実時間処理
- マルチコア対応
- システム再構成

保守運用支援技術
- 耐ネットワーク故障
- 耐ノード故障
- 耐ソフトウェア故障
- セキュリティ攻撃防御
- 予兆検出

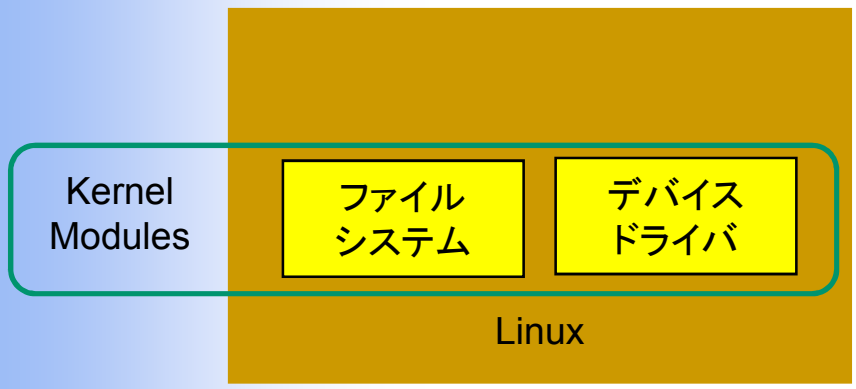
ディペンダビリティ機能拡張基盤：P-Bus 設計思想

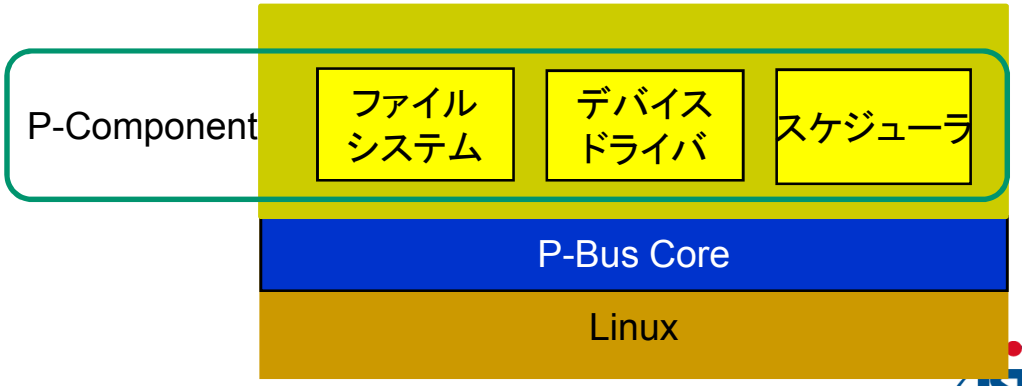
□ 現在のLinuxの機能拡張

- カーネルと機能拡張
  - 密なやりとり
- インタフェース
  - 流動的
  - 曖昧
  - 限定的

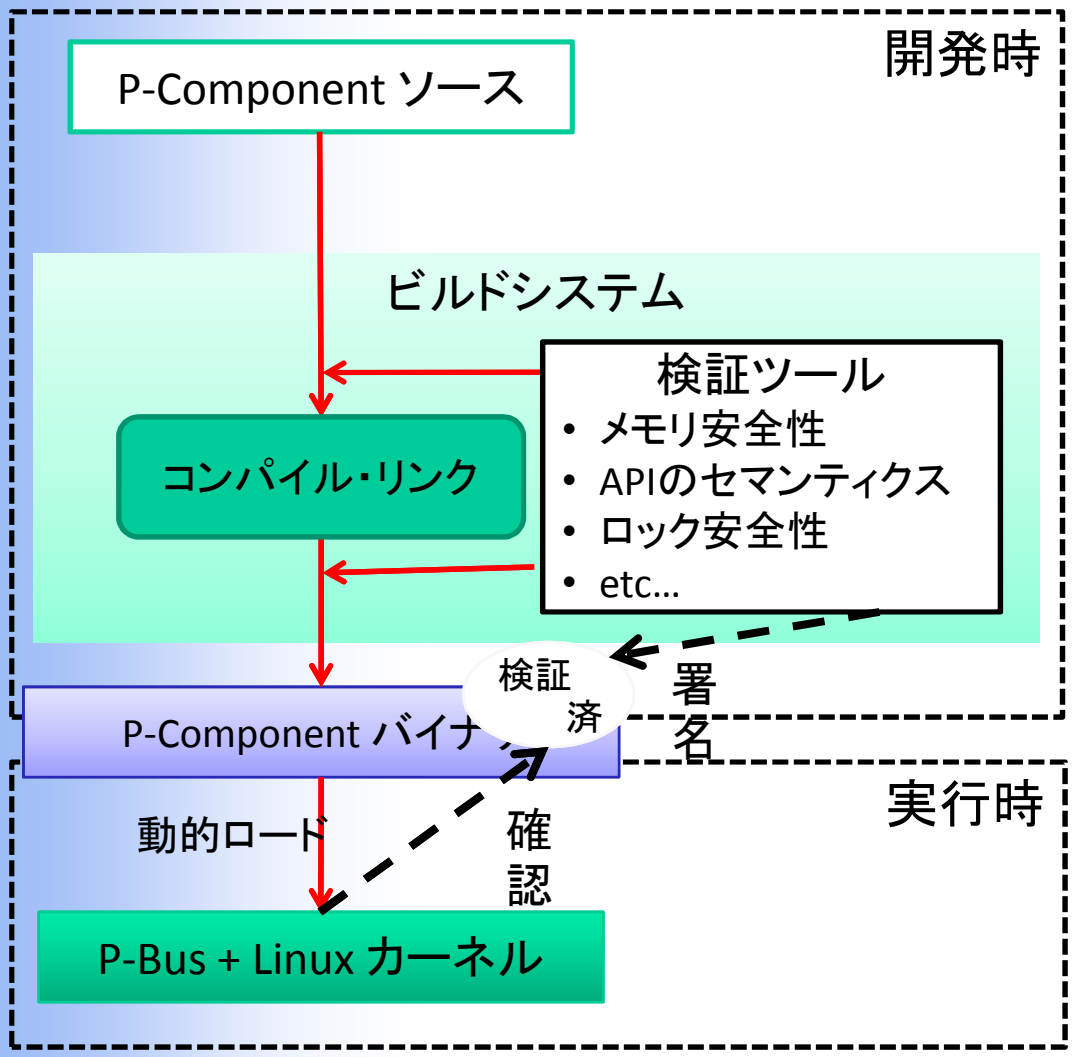
□ P-BusとP-Components

- カーネルと機能拡張コンポーネント(P-Components)の切り分け
- カーネルインタフェースの抽象化と明確化、拡張
- C言語機能の制限
  - e.g., アンバウンドなメモリアクセスを制限
- これらにより、プログラム検証を可能とする





ディペンダビリティ機能拡張基盤: P-Bus仕様記述例



int pbus_bmtx_extrylock(pbu_bmtx_t *mtx)
ブロッキングmutexの排他的確保を試みる

- 実行コンテキスト:プロセスコンテキストのみ
- 休眠: しない
- 事前条件: mtxは、pbus_bmtx_initによって初期化済みであること
- 返り値: 0(成功)もしくはEBUSY(失敗)
- 事後条件: 返り値が0ならmtxはロックされている。EBUSYならmtxは不変

```
/*@ requires context == PBUSV_CTX_PROCESS;
    requires ¥valid(mtx);
    requires  *mtx != PBUSV_UNINITIALIZED;
    assigns   *mtxt;
    ensures  ¥result == 0 || ¥result == EBUSY;
    ensures  ¥result == 0 ➔ *mtx == EX_LOCKED;
    ensures  ¥result == EBUSY ➔ *mtx == ¥old(*mtx);
    */
```

発表概要

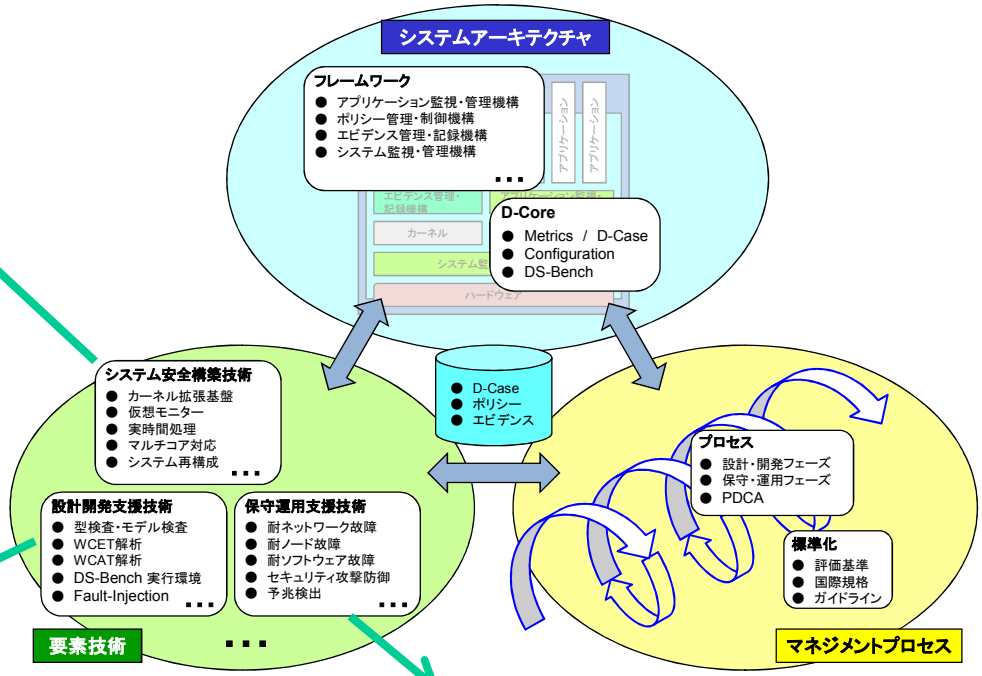
- 全体概要
  - 課題に対するDEOSの取り組みと中間成果の関係図
  - 貢献
  - 研究開発体制
- コアチームの概要
- D-Core: Open Systems Dependability Core技術

  D-Case/Metrics, Configuration, DS-Bench
  - 開発プロセス時支援
  - 運用時支援
  - 関連研究
- システム安全構築技術
  - **P-Components and P-Bus**
- 第一期採択チーム主要成果概要
- まとめ
  - 貢献
  - 将来課題
- このあとの発表

1期採択チームの個別主要中間成果概要

システム安全構築技術

開発モジュール	チーム名
仮想モニタ モニタリング(VMO) マルコア制御(VMC)	中島
P-Bus Core 論理分割(LPAR)	石川



設計開発支援技術

開発ツール	チーム名
型検査・モデル検査(TCHK/MCHK)	前田
最悪実行時間予測(RETAS)	石川
電力使用量予測(GREEN)	徳田
Fault Injection (D-Cloud)	佐藤
ディペンダブルシステムベンチマーク実行環境(DS-Bench)	石川

保守運用支援技術

開発モジュール	チーム名
動作時間予約機構(TR) 耐故障ネットワーク機構(SCTP+FHO)	徳田
耐故障ネットワーク機構(RI2N/PEACH)	佐藤
アカウンティング機構(ACT)	中島/センター
シングルIPアドレス機構(SIAC)	石川

発表概要

- 全体概要
  - 課題に対するDEOSの取り組みと中間成果の関係図
  - 貢献
  - 研究開発体制
- コアチームの概要
- D-Core: Open Systems Dependability Core技術

  D-Case/Metrics, Configuration, DS-Bench
  - 開発プロセス時支援
  - 運用時支援
  - 関連研究
- システム安全構築技術
  - **P-Components and P-Bus**
- 第一期採択チーム主要成果概要
- まとめ
  - 貢献
  - 将来課題
- このあとの発表

まとめ：課題に対するDEOSの取り組みと中間成果の関係図

障害要因の最小化（不完全性）

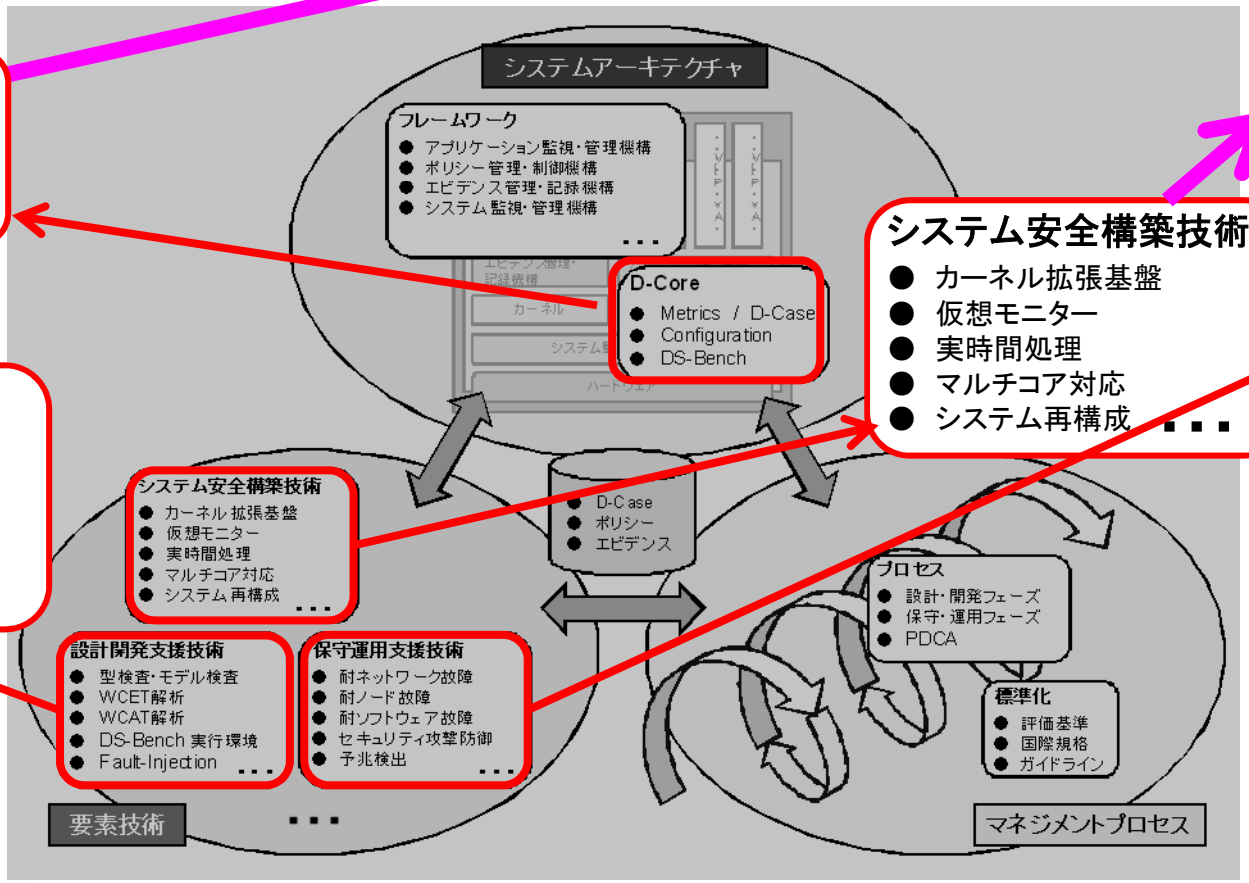
- 要求、仕様、設計、実装、テストの見える化
- 動作解析、振舞い検証
- 動作状況記録、説明責任マネージメント支援
- 国際標準・規格

障害影響の最小化（不確実性）

- 実環境・実時間での仮稼働
- 稼働中の予知
- 障害の最小化、迅速な復旧支
- 動作状況記録、説明責任マネージメント支援

D-Core 技術
- Metrics / D-Case
- Configuration
- DS-Bench

設計開発支援技術
- 型検査・モデル検査
- WCET解析
- WCAT解析
- DS-Bench 実行環境
- Fault-Injection (D-Cloud)
- ・・・

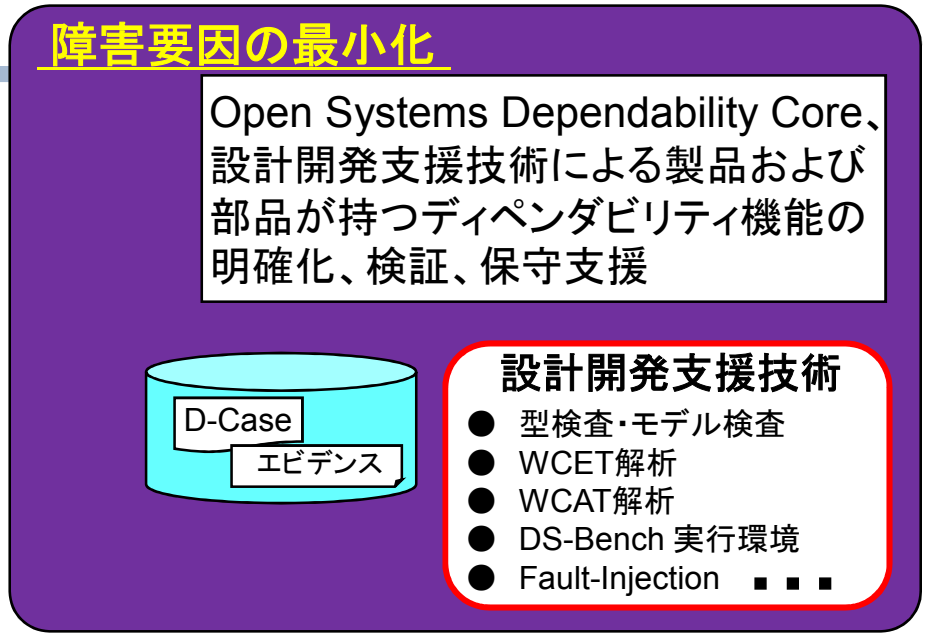


システム安全構築技術
- カーネル拡張基盤
- 仮想モニター
- 実時間処理
- マルチコア対応
- システム再構成
- ・・・

保守運用支援技術
- 耐ネットワーク故障
- 耐ノード故障
- 耐ソフトウェア故障
- セキュリティ攻撃防御
- 予兆検出 ・・・

まとめ：貢献

- 対象分野
  - 社会インフラ
    - 携帯・組み込み、FA/OA、ロボット、サーバ
- 産業界への貢献
  - LCC(Life Cycle Cost)低減
    - 開発コスト
    - 保守コスト
  - PL(製造責任) 問題
  - Value Added(製品付加価値)
  - CSR (Cooperate Social Responsibility)
    - 社会的責任:　安心・安全・グリーンIT
- 学術的貢献
  - D-Core: Open Systems Dependability Core技術
    - D-Case/Metrics, Configuration of P-Components, DS-Bench



まとめ：将来課題

- D-Core: Open Systems Dependability Core技術
  - D-Case表現力の確認とツール&実行時環境開発
  - 現在のメトリクスでは、ディペンダビリティは定性的にしか扱えてない。今後定量的に扱える枠組みが必要
    - ディペンダビリティベンチマークによる定量値は、ディペンダビリティの一側面しか扱えていない
  - DS-Benchツールの開発&データベースの充実
- 未知の障害に対する支援機構、PDCA支援機構
  - 3期採択チームと共に今後検討
- ディペンダビリティ評価規格の策定
  - 3期採択チームと共に今後検討
- システムアーキテクチャ＆フレームワーク
  - コアチーム＆DEOSセンターで検討中

発表概要

- 全体概要
  - 課題に対するDEOSの取り組みと中間成果の関係図
  - 貢献
  - 研究開発体制
- コアチームの概要
- D-Core: Open Systems Dependability Core技術

  D-Case/Metrics, Configuration, DS-Bench
  - 開発プロセス時支援
  - 運用時支援
  - 関連研究
- システム安全構築技術
  - **P-Components and P-Bus**
- 第一期採択チーム主要成果概要
- まとめ
  - 貢献
  - 将来課題
- このあとの発表

このあとの発表

- 全体デモの説明

- ディペンダブルサーバ構築要素技術～DEOSで実現する簡単・安心・省エネサーバー～
  - 石川　裕（東京大学　情報基盤センター　教授）
- DEOSのためのディペンダブル通信機構と消費電力管理機構
  - 徳田英幸（慶應義塾大学　環境情報学部　教授）
- 仮想化を利用したディペンダブルOSの構築
  - 中島達夫（早稲田大学　理工学術院　教授）
- プログラム検証によるディペンダビリティ支援手法
  - 前田俊行（東京大学　大学院情報理工学系研究科　助教）
- 高性能・省電力でディペンダブルな通信リンクPEARL
  - 佐藤三久（筑波大学　計算科学研究センター　センター長）