「実用化を目指した組込みシステム用
ディペンダブル・オペレーティングシステム」
平成 18 年度採択研究代表者

H22 年度
実績報告

石川 裕

東京大学情報基盤センター・センター長

# 並列・分散型組込みシステムのためのディペンダブルシングルシステムイメージ OS

## §1. 研究実施の概要

石川グループ

ソフトウェアシステムのライフサイクルを通した、ディペンダビリティを維持していけるソフトウェア体系とディペンダビリティ評価指標を確立し、産業界での実用化を目指す。マルチコア環境を想定し、大きく5つの研究開発を行ない、要素技術の基本実装が終了した。今後、これらの基本実装を基に全体を統合し、ディペンダビリティ維持システムの枠組みを示していく。

1) Linux を基に、安全かつ容易に OS 機能を拡張できる機構 P-Bus の研究開発
2) CPU、メモリ、デバイスの論理分割機構の研究開発
3) クラスタリングされたコンピュータ群で単一システムイメージを提供する機構の研究開発
4) ディペンダブルシステムの開発、運用・保守、改良を支援する手法である D-Case を記述するツールおよびベンチマークフレームワークの研究開発
5) オープンシステムディペンダビリティを継続的に担保することを可能にするフレームワーク技術の開発（DEOS センターと共同で作業を進めた）

島田グループ

ディペンダブルシステムの開発、運用・保守、改良を支援する D-Case 手法に基づいた、開発支援環境の構築を目的として研究を進める。実例に対する D-Case 手法の適用実験を通して、手法の詳細化と効果・効用の明確化を行うと共に、その過程で課題と要件抽出を行いツールへ反映させる。具体的には以下の3項目の活動を進めた。

1) 手法の普及を促進するために、手法導入コストとそのメリットを示すことが可能な実事例の探索、
2) 既存の開発、運用・保守、改良プロセスと最小のコストで無理なく D-Case 手法を統合・適用するための考え方やガイドラインの作成、
3) 前記2点の活動を通して、抽出した要件に基づき、D-Case ダイアグラム記述が容易に行えると共に、既存の開発、運用・保守、改良プロセスと連携が行えるツール開発。

平成 22 年度は D-Case 手法における D-Case ダイアグラムを容易に記述・参照するためのツールについて研究・開発を進めて、基本機能の実装を完了した。今後、手法の有効性を端的に示す事例を示していく。

## §2. 研究実施体制

(1)「石川」グループ

①研究分担グループ長：石川裕 （東京大学情報基盤センター・センター長）

②研究項目

- 高信頼シングルシステムイメージ OS

(2)「島田」グループ

①研究分担グループ長：島田利郎 （富士ゼロックス株式会社・研究技術開発本部システム要素技術研究所・グループ長）

②研究項目

・ D-Case に基づくソフトウエア開発支援環境

## §3. 研究実施内容

(文中に番号がある場合は(4－1)に対応する)

(1)石川グループ

本年度は下記の項目に関する研究を行った。

1) D-Case[6]および DS-Bench

(1) 富士ゼロックスと D-Case エディタの共同開発

Eclipse をベースにしたエディタのプロトタイプを実装した。詳細は島田チームの項を参照のこと。

(2) D-Case データ構造の設計

D-Case のデータ構造を定め、パターン、変数、型を導入した。これにより、従来の Assurance Case では行えなかったエディタからのパターン利用、および変数の自動型チェックが可能となった。

(3) D-Case 記述実験

DEOS センターと共同で受付ロボットシナリオを設定し、デモシステムを記述した。また、記述プロセス自体も D-Case で表現する取り組みも新たに行ない、骨組みを提示することができた。D-Case と D-fops の連携実験を行い、ET2010、DEOS 国際シンポジウムにおいてデモを行った。

(4) D-Case 標準化活動

Object Management Group などにおける Assurance Case の標準化会議に参加し、D-Case による、Assurance Case プロセスの提案、Assurance case におけるパターン、変数や型の導入などの提案を行った。

(5) Metrics, DS-Benchmark の統合

Metrics のための基本的な評価機能を D-Case Editor に実装した。今後、DS-Bench との連携を行って行く。

(6) 平成 21 年度に開発した DS-Bench (Version 1)を改良し、DS-Bench Version 2 として以下の機能を実装し、ET2010、PRDC 2010、DEOS 国際シンポジウムのそれぞれでデモを行った。

- ベンチマークシナリオ作成・編集・実行におけるユーザインターフェースの改良(図 1)
- ベンチマーク結果のデータベース化機能
- プロビジョニング機能(ベンチマーク実行時にベンチマークプログラムをターゲットハードウェアにインストールする機能)

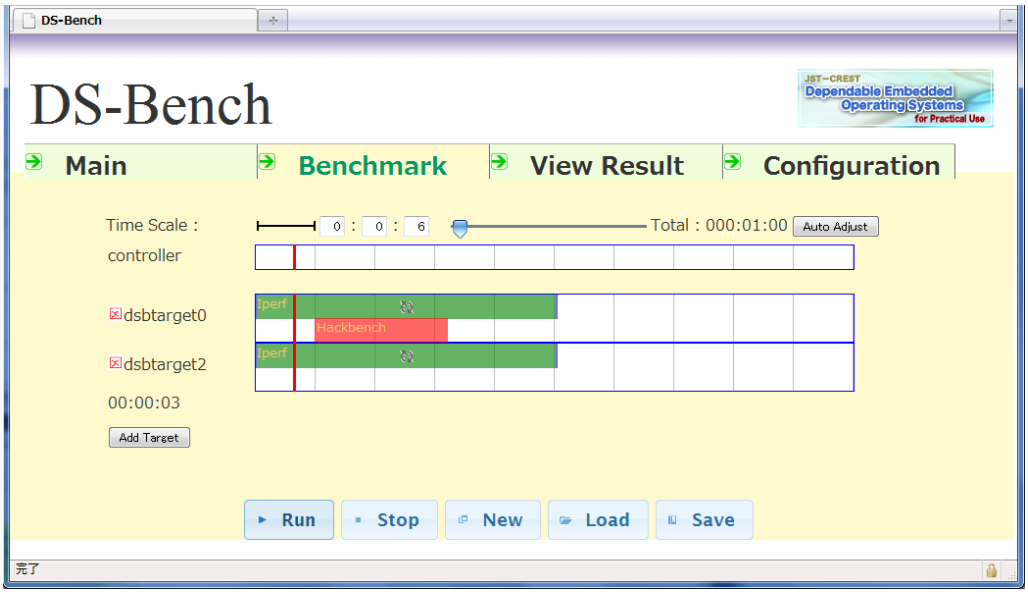


図 1 DS-Bench2 のベンチマーク実行画面

2) P-Bus およびシステムソフトウェア検証

検証サブコアチームという形で、システムソフトウェア検証に関する研究開発を前田チームと一体となって進めた。今年度は東大前田チームで開発中のモデル検査器について、特に Linux カーネルのソースコードを念頭に、実際のシステムソフトウェア検証を行う上での不具合の指摘や機能強化に関する提言を行った。

3) シングルシステムイメージ

平成 21 年度に開発したシングル IP アドレス機構について、性能評価を行い論文として発表した[4][7]。また、前述のシングル IP アドレス機構において、サーバノードの障害時に複数のノードが応答してしまう問題に対処する機構の研究を行った[5]。さらに、平成 21 年度までに開発してきたチェックポイント機構を応用し、マルチコア CPU を用いたサーバの耐故障性を向上させる研究[1]や、サーバクラスタの負荷分散を行う研究[3]、リアルタイムプロセスや対話的プロセス向けのチェックポイント機構[2]について、成果を発表した。

4) 論理分割機構

平成 21 年度から行っている論理分割機構の再実装をほぼ完了し、x86 アーキテクチャの 32bit, 64bit 両方で動作することを確認し、さらに複数の Linux が動作することを確認した。また、実装の改善により、異なるバージョンの Linux を同時に実行できるようになった。この実装をもとにして、Linux の耐故障性を高める機能を追加する研究を行った。

5) 静的 WCET(Worst Case Execution Time)検証ツール

平成 21 年度までの研究開発で得られた成果物をツールとして以下のウェブページで公開した。http://www.il.is.s.u-tokyo.ac.jp/retas/

6) D-Framework for Open Systems Dependability (D-fops)

平成 22 年度は東大石川チームと DEOS センターとの協業の元に、以下の項目の研究開発を進めた。

1) DEOS コンセプト

プロジェクト総括および領域コアチームと共同して DEOS アーキテクチャをまとめた。図 2 における右下部分が D-Framework が担当する機能群である。D-Case との連携をオンラインとオフラインの両面から実現しているのが特徴であり、ステークホルダーの合意を確実に実行できるとともに、システムが合意から何らかの理由で乖離した場合にも適切にステークホルダーが関与・対処できる。

2) フレームワークアーキテクチャ

オープンシステムディペンダビリティを継続的に向上させるために必要な機能を 4 つの層(Foundation Service, Sustainable Dependability Service, Operating System Service, & Application Management Service)に分解して定義し、各層の機能責任を明確にした(図 3)。

3) 実装

DEOS センターにおいて、フレームワークプロトタイプの実装を行った。成果は ET2010 始め、国際シンポジウムで発表した。11 月に 0.5 版の実装をリリースした。

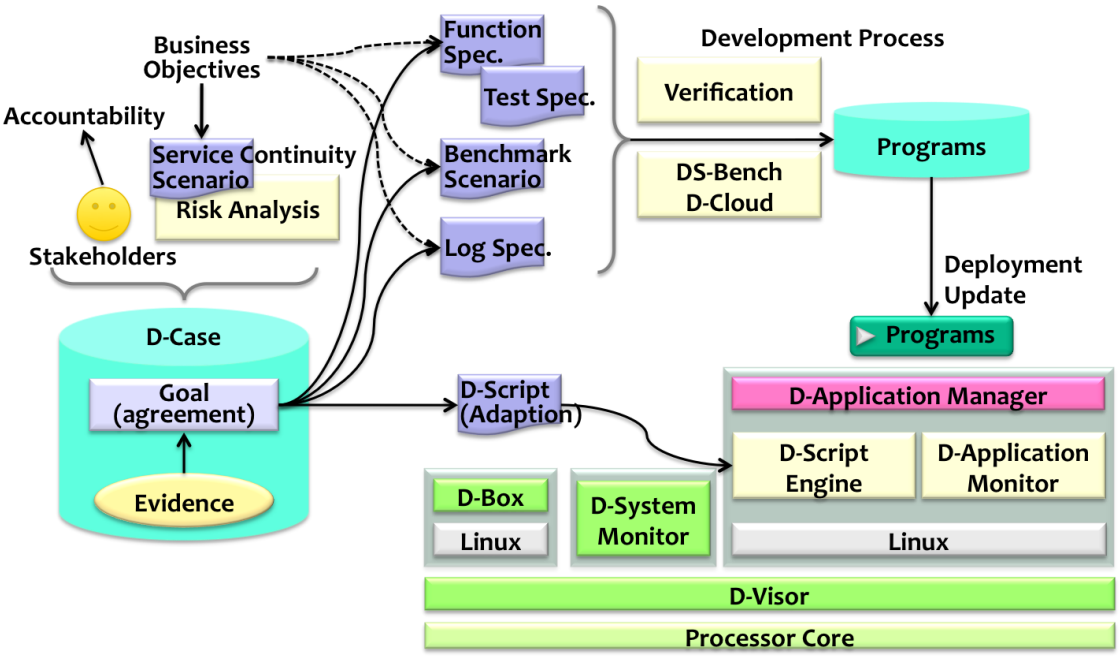


図 2 DEOS アーキテクチャ

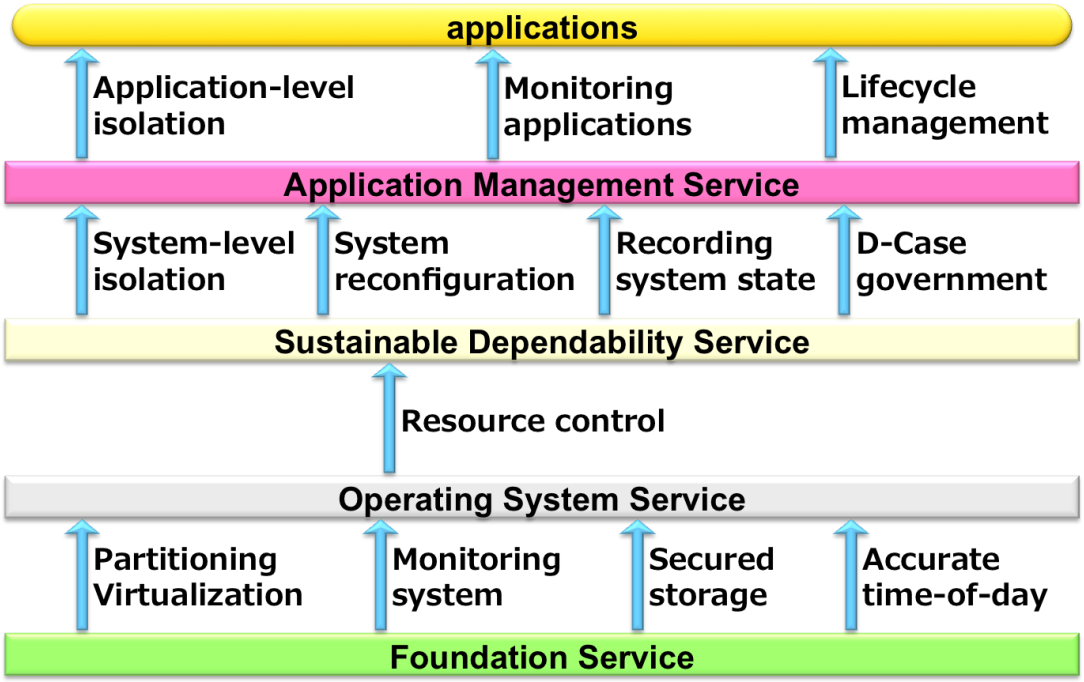


図 3 オープンシステムを実現するための4層

(2) 島田グループ

平成 22 年度は東大石川チームのサブテーマとして D-Case 研究チームの松野リーダーの下で、以下の項目の研究開発を進めた。

１） D-Case エディタ開発

平成 22 年 3 月にチーム内部にリリースした D-Case エディタのプロトタイプをベースに、要件抽出と仕様策定、実装と試用による更なる要件抽出を繰り返すインクリメンタル開発を行い、4半期毎にツールのリリースを行った。オープンソースの統合開発環境である Eclipse を対象とし、Eclipse 上のグラフィックアプリケーションフレームワークである GMF(Graphical Modeling Framework)を利用する構成を踏襲し、基本図形描画機能の実装と共に、エディタと外部ソフトウェア或は外部機能との連携に必要な拡張機能の実装を行った。

拡張機能としては、D-Case 手法の普及・展開を図る際に既存のソフトウェア開発プロセスへの導入を容易にすることを目的に、OSS プロジェクト管理ツールである redmine との連携機能を実装した。また、D-fops と連携してリアルタイムにディペンダビリティの不具合箇所を提示する機能を実装した。これら 2 つの拡張機能について学会・ワークショップ・展示会で、デモンストレーションを行い、手法やコンセプトを実業務にどう適用するのかを紹介するための材料として活用した。D-Case エディタの外観を図4に示す。

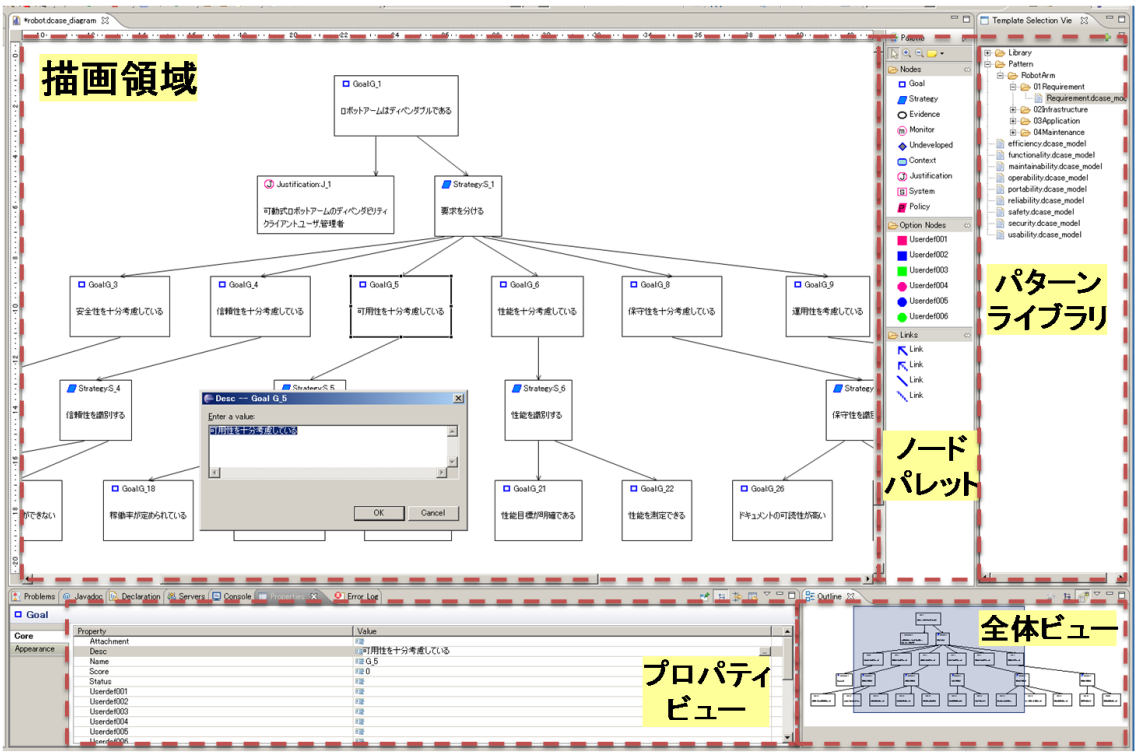


図 4　D-Case エディタ

２）D-Case Viewer 開発

ディペンダビリティを合意形成する際には、ステークホルダーは必ずしも開発者とは限らない。開発者でないステークホルダーが D-Case ダイアグラムの拡大・縮小を簡単に操作して、議論を行うためのツールとして、iPad 上で動作する D-Case Viewer の提案・試作を行い、D-Case エディタと合わせてデモンストレーションを行った。D-Case Viewer の外観を図5に示す。

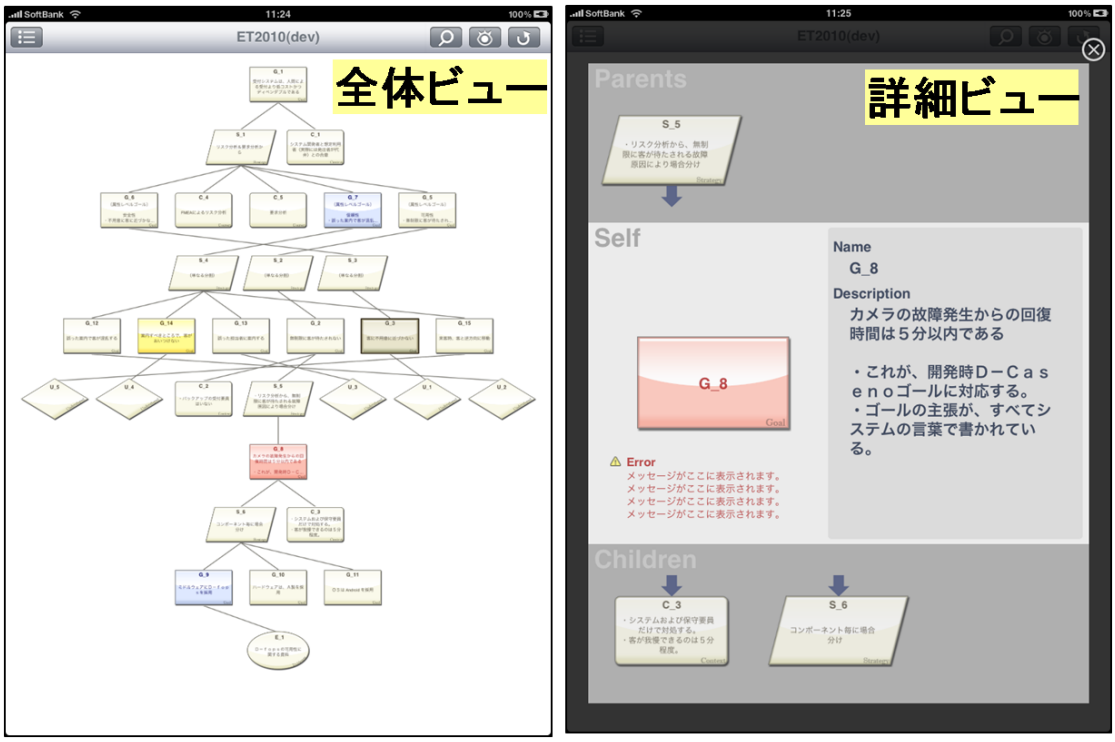


図 5　D-Case Viewer

3)既存開発プロセスへの適用及び実事例検討

D-Case 手法の有効性と普及促進を図るためには、組込システム商品の企画・開発プロセスに対して、D-Case 手法をどのような形で適用し統合するのかを明らかにすると共に、D-Case 手法導入コストやそのメリットが感じられる事例獲得を試みた。現時点では、まだ企画・開発プロセスへの統合の道筋を示すことができておらず、また、D-Case 手法導入のメリットを感じられる充分な事例を示すことはできていないが、Object Management Group への提案の中や、展示会で来場者に配布する冊子の中の参考事例で紹介することができた。

平成 22 年度は D-Case 手法における D-Case ダイアグラムを容易に記述・参照するためのツールについて研究・開発を進めて、基本機能の実装を完了した。今後、手法の有効性を端的に示していく。

## §4. 成果発表等

### （4－1） 原著論文発表

●論文詳細情報

[1] Balazs Gerofi, Yutaka Ishikawa, A Multi-core Approach to Providing Fault Tolerance for Non-deterministic Services, 2010 Ninth IEEE International Symposium on Network Computing and Applications, pp. 233-238, Jul 2010 (DOI: 10.1109/NCA.2010.42)

[2] Jianwei Liao, Yutaka Ishikawa, A New Concurrent Checkpoint Mechanism for Real-Time and Interactive Processes, 2010 IEEE 34th Annual Computer Software and Applications Conference 2010, pp. 47-52, Jul 2010 (DOI: 10.1109/COMPSAC.2010.12) (Short Paper)

[3] Balazs Gerofi, Hajime Fujita, Yutaka Ishikawa, An Efficient Process Live Migration Mechanism for Load Balanced Distributed Virtual Environments, 2010 IEEE International Conference on Cluster Computing (Cluster 2010), pp. 197-206, Sep 2010 (DOI:10.1109/CLUSTER.2010.25)

[4] Hajime Fujita, Yutaka Ishikawa, DTS: Broadcast-based Content-aware TCP Connection Handover, 第 22 回 コンピュータシステム・シンポジウム (ComSys 2010), pp. 21--30, Nov 2010

[5] Jun Kato, Hajime Fujita, Yutaka Ishikawa, Design and Implementation of a Fault Tolerant Single IP Address Cluster, The 16th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC'10), pp. 175--183, Dec 2010 (DOI: 10.1109/PRDC.2010.39 )

[6] Yutaka Matsuno, Jin Nakazawa, Makoto Takeyama, Midori Sugaya, Yutaka Ishikawa, Toward a Language for Communication among Stakeholders, Proc. of the 16th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC'10), pp. 93-100, Dec 2010 (DOI: 10.1109/PRDC.2010.47 )

[7] Hajime Fujita, Yutaka Ishikawa, DTS: Broadcast-based Content-aware TCP Connection Handover, IPSJ Transactions on Advanced Computing Systems (ACS 33) , Vol. 4 No. 2 pp.59-69

[8] Yutaka Matsuno, Kenji Taguchi, Parameterised Argument Structure for GSN patterns, Proc. IEEE 11th International Conference on Quality Software (QSIC 2011), 6 pages, July 2011 (DOI:TBA), to appear (Short Paper)

[9] Jin Nakazawa, Yutaka Matsuno, Hideyuki Tokuda, Evaluating Degree of Systems' with Semi-Structured Assurance Case, Proc. 13th European Workshop on Dependable Computing (EWDC 2011), 2 pages, May 2011 (DOI: TBA), to appear (Short Paper)

### （4－2） 知財出願

① 平成22年度特許出願件数(国内 2 件)

② CREST 研究期間累積件数(国内 2 件)