高信頼UNIX「風雅」の概要*

6M－1

岸本　光弘†，中島　淳‡
（株）富士通研究所§
{kiss, nakajima}@flab.fujitsu.co.jp

1　はじめに

オープンシステムの普及に伴い，従来メインフレームとそのオペレーティング・システム(OS)上に構築されてきた，企業の基幹業務システムを，オープンシステム特にUNIX上に構築する事例が増加している．しかし，UNIXはメインフレームのOSと比較すると，一般に1桁程度信頼性が低いと言われており，基幹業務システムへのUNIX適用を躊躇する原因の一つとなっていた．

我々はフォールト・トレランス技術のOS自身への適用方式を研究し，高い信頼性を持つOSを実現することができた．本稿では，開発した高信頼UNIX「風雅」の概要を報告する．

2　UNIXの信頼性

UNIXは歴史的にエンジニアリング分野で成長してきたこともあり，メインフレームのOSと比較すると，信頼性が十分とは言えなかった．信頼性を計る尺度である，可用性（＝運用時間／（運用時間＋停止時間））で言えば，メインフレームのOSが99.9～99.99%であるのに対し，現在の商用UNIXは99～99.9%程度である．[1]

原因の一つは「lasy panic」と呼ばれるコードがOS内部に存在していることである．「lasy panic」とは，予期しない資源不足のような対処が難しい事態が発生した場合，すぐにOS全体の異常終了処理を実行するものである．

商用のUNIXにおいては，「lasy panic」を除去したり，設計やコーディングのバグを，レビューや試験により修正する努力が払われている．これらはフォールト・アボイダンスと総称される技術である．しかし，OSは複雑なシステムであり，誤りを根絶することは非常に難しい．特にUNIXは，次々に新機能が追加され，成長していくOSでありフォールト・アボイダンスだけによる高信頼化には限界がある．

また，CPUやI/O装置を多重化し，ハードウェアの故障をソフトウェアに見えなくするハードウェア・フォールト・トレラント・マシンが幾つか商品化されている．しかしハードウェアのフォールト・トレランスでは，実際に誤りの大半を占めるソフトウェアのバグに対応することができない．

我々は，フォールト・アボイダンスの技術に加え，ソフトウェアによるフォールト・トレランスの技術をOS自身に適用することにより，UNIXの信頼性をメインフレームOSと同等，もしくはそれ以上にする研究を行った．フォールト・トレランス技術とは，内部で誤りが発生しても，それを検出，隔離，リカバリすることで，誤りをマスクする技術である．

富士通のSURE SystemやタンデムのNon Stopなど，ノーダウン，ノンストップを実現したフォールト・トレラント・コンピュータが商品化されている．これらは99.999%以上の可用性や24時間連続運転，さらにハード，ソフトの活性増設，交換といった高度な機能を実現している．そのため専用の高価な二重化ハードウェアを用い，プロプライアタリなOSを採用しており，高信頼UNIXとは別の市場を狙ったシステムとなっている．

3　風雅システムの特徴

高信頼UNIX「風雅」の主要な特徴を示す．

「ユーザプログラムの修正が不要」

OS（システムコール）のレベルで故障をマスクするので，ユーザのプログラムを修正したり，リコンパイルしたりする必要がない．

「ソフトウェアのバグによる故障を救済する」

ハードウェアの故障だけでなく，実際の故障の大部分を占めるソフトウェアのバグによる故障も救済することができる．

「特別なハードウェアを使わない」

移植性を高め，オープンなハードウェア上での動作のために，高信頼性の実現に特別なハード

*Highly Available UNIX "FUGA" Overview
†Mitsuhiro Kishimoto, ‡Jun Nakajima
§Fujitsu Laboratories Ltd.

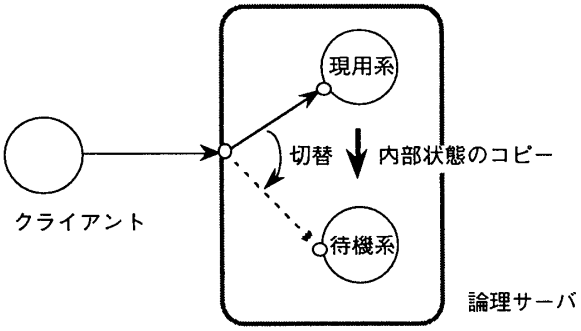

図1　プロセスペア方式

ウェアは使わない．必要な機能は，定常性能の低下を小さく抑えて，ソフトウェアで実現する．

「保守性と拡張性の維持」

既存のUNIXのソースプログラムはOS部分だけで500Kステップ以上ある．高信頼化のためには既存ソースの修正が必要であるが，障害管理機構と高機能なライブラリを用意することで，既存の処理論理を変更せず修正量も僅かですむ．プログラムの保守性や将来の拡張性を維持することができる．

4　風雅のアーキテクチャ

風雅は，Chorusマイクロカーネルとその上で動作するマルチサーバUNIXサブシステムをベースとして採用している．[2]　UNIXサブシステムは，複数の内部モジュールに分割されており，クライアント／サーバモデルで動作している．

サーバを現用系と待機系のペア（プロセスペア）で二重化し，故障のリカバリを可能にしている（図1）．通常の運用時には，現用系がクライアントからの要求を処理する．現用系は，内部状態を待機系にコピーして引き継ぎに備える．SURE SYSTEMで採用した，オーバヘッドが小さく故障伝搬の可能性の低い，遅延引き継ぎおよびエッセンス引き継ぎの技術を用いている．[3]　現用系に故障が発生すると，待機系が処理を引き継ぎ，現用系に切り替わる．そして新たな待機系を起動し，将来の故障に備える．

サーバのプロセスペア化を支援するために，図2に示すように，1つの新規サーバと2つのライブラリを作成した．[4]

「Fault Manager　(FTM)」

故障の検出からリカバリまでの一連の処理の指示を出す故障管理サーバで，各ノードに配置され，

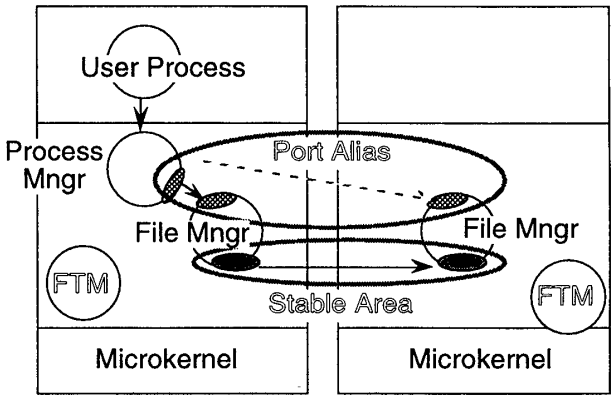

図2　風雅のアーキテクチャ

同一の内部状態を持つことで，自分自身も耐故障性を備える．

「Port Alias　(PA)」

耐故障性を備えたプロセス間通信機構で，クライアントからのリクエストの宛先の自動切り替え処理や，故障発生時の再送処理，冗長なリクエストを検出し処理する機能を提供する．

「Stable Area　(SA)」

現用系の内部データを格納する機構で，現用系に故障が発生した時は，待機系が内容を読み出す．

5　風雅のプロトタイプ

実際にファイル管理サブシステムをプロセスペア化し，イーサネットで結合したPCクラスタ上で動作させ，ソフト・ハードの故障を救済できることを実証した．また，実行時のオーバヘッドや故障救済時間，コードの変更量を測定，分析し改善手法の検討を行った．[4]

謝辞

本研究の推進について御指導頂いた，システム研究部の村松洋部長，吉田浩部長付に感謝します．

参考文献

[1] Unix International High Availability Working Group, "Requirement for High Availability Technology", Apr. 5, 1993

[2] Batlivala et al., "Experience with SVR4 Over CHORUS", USENIX Workshop on Micro-Kernels and Other Kernel Architectures, Apr. 27, 1992

[3] 村松他，"システムを止めずに保守・運用が可能なOSを開発"，日経エレクトロニクス，No.520, pp. 209-223, 1991

[4] 岸本他，"高信頼Unix「風雅」"，第52回情報処理全国大会論文集 6M2-6, 1996