

研究テーマ(案)の概要(修正版)

2016/6/15

鈴木 森羅

1 はじめに

本資料は資料< 302-03 >の修正版である。鈴木が取り組む研究テーマ(案)とその概要について述べる。また、提案する具体的な手法とそれに伴う課題を述べる。

2 修正箇所

提案する具体的な手法について図を用いて述べる4章を追加した。また、5章の課題をより具体的なものに修正した。

3 研究テーマ(案)

(題目) 実行中のソフトウェアを外部から複製する手法の研究

(概要) チェックポイントの実装法の1つに Fault Tolerance Interface (以降 FTI)[1] がある。FTI はライブラリであるため、FTI を用いてソフトウェアにチェックポイントを実装するには、ソフトウェアのソースコードを改変する必要がある。つまり、ソフトウェアに依存する。これにより、既にチェックポイントを実装しているソフトウェア、またはソースコードの手に入るソフトウェアでしかチェックポイントを利用できなくなる。また、実行中のプロセスが自プロセスを複製する技術として、フォークがある。しかし、フォークを使って複製できるのは自プロセスであるため、フォークを実装する場合も、ソフトウェアのソースコードを改変する必要がある。また、フォークでは複製したプロセスを走行させてしまう。そこで、本研究では、フォークを改変して複製したプロセスを走行させず保存する。その手段として、実行中のソフトウェアのシステムコールをフックし、改変したフォークを実行させる。これにより、実行中のソフトウェアを複製する。結果として、実行中のソフトウェアを外部から複製することにより、ソフトウェアに依存することなく、実行中のソフトウェアの状態を保存する。本研究ではフォークを用いて、ソフトウェアに依存することなく、実行中のソフトウェアを外部から複製する手法を提案する。本研究は、ソフトウェアを停止させることなく他の計算機に移行させるライブマイグレーションや、障害発生リスクに備えるためのソフトウェアの冗長化などの場面への利用が考えられる。

4 具体的な手法

4.1 手法の概要

本章では、複製するソフトウェアに改変したフォークを外部から実行させるための具体的な手法について説明する。複製するソフトウェアに改変したフォークを外部から実行させるということは、一般化すると任意のソフトウェアに任意のシステムコールを外部から実行させるということである。そこで、任意のソフトウェアに任意のシステムコールを外部から実行させるためには、まずその契機を定める必要がある。本手法では、手法の対象となるソフトウェアにより特定のシステムコールが発行されたことを契機とする。契機となるシステムコール(以降、契機システムコール)が発行されたことの判断は、システムコールゲートで行う。これは、全てのプロセスのシステムコールがシステムコールゲートを通るからである。契機の判断を行うために、システムコールゲートに新たに契機を管理するためのテーブル(以降、契機管理テーブル)を持たせる。契機管理テーブルのエントリには以下のものを格納する。

- (1) システムコールを実行させるプロセスの pid
- (2) 契機システムコール番号
- (3) 契機の対象となる実行させたいシステムコール(以降、対象システムコール)番号

契機管理テーブルを用いて以下のように処理をする。

- (1) 格納された pid を持つプロセスが契機システムコールを発行
- (2) システムコールゲートにシステムコール番号とシステムコールに応じた引数の受け渡し
- (3) 契機管理テーブルを用いて契機を判断
- (4) 契機システムコールと対象システムコールを順に実行

他のプロセスからこの契機管理テーブルに契機を設定することで、外部から任意のソフトウェアに任意のシステムコールを実行させる。このため、契機管理テーブルに契機を設定するシステムコール(以降、契機設定システムコール)を新たに実装する。図 1 では、実装または改変の必要なものに色を付けて示す。

4.2 手法の流れ

続いて、図 1 を用いて、本手法のおおまかな処理の流れについて説明する。図 1 では、プロセス A とプロセス B の pid をそれぞれ 4 と 5 とする。また、改変したフォークを forkx とし、fork と forkx のシステムコール番号を 100 と 101 とする。処理の流れは以下の通りである。

- (1) 契機の設定
プロセス A は契機設定システムコールを用いて、契機管理テーブルに pid 5、契機システムコール番号 100、および対象システムコール番号 101 を格納する。
- (2) 契機システムコールの発行

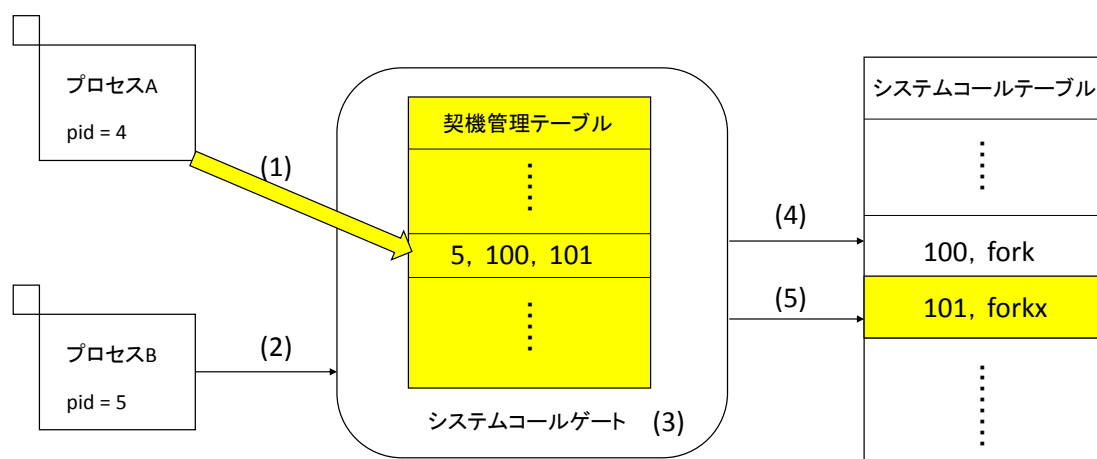


図1 処理流れ

プロセス B は契機システムコールを発行し，システムコール番号 100 と引数をシステムコールゲートに渡す．

(3) 契機の判断

システムコールゲートは，プロセス B の pid とシステムコール番号から契機であると判断する．

(4) 契機システムコールの実行

(5) 対象システムコールの実行

システムコールゲートは契機システムコール fork と対象システムコール forkx を順に実行する．

これにより，任意のソフトウェアに外部から任意のシステムコールを実行させる．

5 今年度の課題 (案)

本研究で，提案する手法の実現に向けての課題 (案) を以下に示す．

- (1) システムコールゲートの改変
- (2) 複製する契機を登録するシステムコールの実装
- (3) フォークの改変

6 おわりに

本資料では，鈴木が取り組む研究テーマ (案) とその概要について述べた．

参考文献

- [1] Bautista-Gomez, L., Tsuboi, S., Komatitsch, D., Cappello, F., Maruyama, N. and Matsuoka, S.: FTI: High Performance Fault Tolerance Interface for Hybrid Systems, *Proceeding of 2011*

International Conference for High Performance Computing, Networking, Storage and Analysis(SC'11), No. 32 (2011).