

組み込み Java™ の内部状態の監視機能の実装

井奥 章 中野 正樹 鈴木 貴之 森本 義章 里山 元章

(株)日立製作所 システム開発研究所

1. 背景：組み込み Java の高信頼化の必要性

インターネットの普及とともに PC の WEB ブラウザや WS への適用を通じて発展してきた Java™ 実行環境は、さらに従来の枠に収まらない新しい適用対象を求めて、進化し続けている。よく知られている携帯電話や PDA といった機器への適用に加え、非 PC 型の組み込み機器への適用拡大の流れは、Java™ のマルチプラットフォームの特性による利便性のため、一層広範な動きを示している。

たとえば、情報制御システム分野では、IP ネットワークを活用した新サービスの提供・運用・保守の効率化が求められている。そのため、システム構築の核となるコントローラなどでは、インターネット対応・プラットフォーム独立・管理コスト低減の狙いから、Java™ 実行環境のサポートが重要となってきた。

Java™ 実行環境の高信頼化は、このような制御系組み込み機器ならではの適用課題として重要になると考えられる。

2 高信頼化に必要な機能

高信頼化に必要と考えられる機能の一例を示す。いずれも既存の組み込み Java 実行環境で不十分であるか、または存在しないと見なせるものである。これらは、機器やサービスの RASIS (R:信頼性/A:可用性/S:保守性/I:保全性/S:機密性) を支援する機能に相当する。

2.1 実行時の監視

(1) 統計情報収集機能

運用中に JavaVM の各種統計情報を収集 (モニタ) できるようにする。

GC / メモリ使用状況 / スレッドの状態など。

(2) 警報機能

設定した条件で警告を発する。

2.2 障害時のデバッグ・復旧支援

(1) 障害情報収集 (障害発生時のダンプ機能)

障害解析を支援するための情報を出力する。

エラー発生点での変数値 / バイトコード逆アセンブルリストなど。

(2) オンライン Java デバッガとの接続機能

遠隔保守端末から組み込み機器内部の Java ソースデバッグを行う。

(3) Java アプリケーションの救済機能

障害発生前のデータを再開後に引継ぎ、プロセスを復元する。

2.3 保守運用機能

(1) 起動機能の拡張: config ファイルからの Java アプリ名、起動オプションの指定を可能にする。

(2) Java アプリケーションの強制終了
不具合を発生したアプリケーションのみを選択的に外部から強制終了させ、被害を局所化する。

本稿では、Java 実行環境の高信頼化の端緒として、2 章の一端を担う監視ツールを採り上げる。

3 組み込み機器向けの実装上の課題

組み込み向けの機器特性や開発作業の特徴から、次の 2 点が大切な課題になると考えた。

3.1 省メモリでの動作

組み込み機器は、メモリ制約が強い。メモリ搭載量が物理的に少ないという制約のみならず、仮想記憶機構をサポートしていないなどの制約が加わる。

3.2 低工数での機能実現

組み込み機器の開発では、専用ハードウェア上にシステムが構築されることが多くノウハウや開発成果の共有が難しい。

一方で、組み込み機器ソフトウェアに対して、開発期間の短縮に対する要求が厳しくなっており、工数抑制が、開発工程の成否に大きく関与する。

これら課題を意識して、高信頼化機能の実現に際しての開発工数やメモリ消費、さらには性能への影響を把握し、今後の機能拡張の指針を得ることを当面の目標と考えた。

4 プロトタイプ実装

このような背景の元、筆者らは、JavaVM の内部状態の監視、及びメソッドの実行監視を行う JVMPi 準拠のエージェントを、SH (日立製 RISC プロセッサ^[1]) / Linux 用組み込み Java™ (J2ME/CDC^[2]) に対応させ、実用性評価を行った。

課題に挙げた低工数、省メモリでかつ実行性能を劣化させずに Java 実行環境の高信頼化機能を実現させる方式として、Java™ のプロファイル用標準 API である JVMPi (Java Virtual Machine Profiler Interface^[3]) を

Implementation of Monitoring the Internal Status of Embedded Java

Akira Ioku, Masaki Nakano, Takayuki Suzuki,
Yoshiaki Morimoto, Motoaki Satoyama
Hitachi, Ltd., Systems Development Laboratory

採用した。

その結果、PC/WS の Java™ 用に開発した Java 実行環境の監視モニタを、組み込み Java™ 環境でも効果的に活用できる見通しを得た。

今回のデモとして紹介するのは、監視モニタを、Solution Engine (日立超 LSI システムズ製 SH 評価ボード^[4]) 上の組み込み Java™ 環境に対応させたプロトタイプである。

本稿では、以下、このプロトタイプ実装について、機能仕様、性能評価を示し、利便性への評価・考察を述べる。

5 監視モニタ

5.1 機能概要

監視モニタとは、JavaVM の実行状況を監視し、異常発生時にはその原因究明とデバッグを支援するツールである。次の機能を持つ。

- (1) GC、スレッドの状態変化、オブジェクトの生成、メソッド呼び出しなど、イベントの通知
- (2) メモリ使用量の監視
- (3) ヒープ、スレッド、競合しているモニタのダンプ

5.2 構成

監視モニタの構成を図 1 に示す。

JVMプロセス

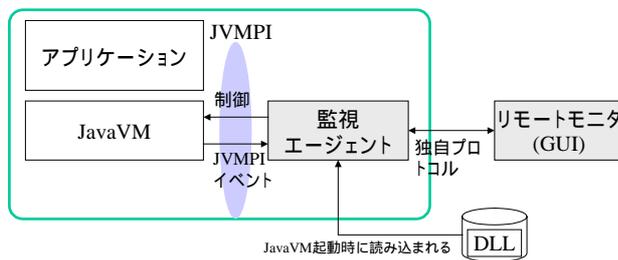


図 1 監視モニタの構成

(1) 監視エージェント

監視対象の JavaVM と同じプロセスで動作し、JVMPI を通じて JavaVM の監視 / 制御を行う。

(2) リモートモニタ(GUI)

監視エージェントとは別のプロセスで動作し、監視エージェントと通信して監視 / 制御をリモートから GUI で行う。リモートモニタは Java アプリケーションであり、JDK 1.3 以上の環境で動作する。

6 測定：監視機能の追加によるオーバーヘッド

6.1 ベンチマーク

監視モニタに対して監視情報をリアルタイムに出力しながら測定した Embedded Caffeine Mark(ECM) 値を表 1 に示す。ECM は、組み込み Java™ における標準的なベンチマークプログラムである。

メモリ、GC、Thread および Monitor を監視する分には、性能への影響が比較的小さい結果が得られた。

表 1 監視モニタ ECM 結果

監視対象	GC	GC, Thread	GC, Thread Monitor	GC, Thread Monitor Object	Method 含む 全て	監視モニタなし
Sieve	80	80	80	75	77	108
Loop	74	74	74	72	65	112
Logic	76	78	77	73	79	109
String	305	309	309	50	3	457
Float	87	91	91	88	23	82
Method	80	81	80	81	0	103
Overall	99	100	100	71	0	131

6.2 バイナリサイズ

Java 実行環境 (J2ME/CDC) のバイナリサイズの比較を表 2 に示す。

表 2 JVMPI 有無によるバイナリサイズの相違

JVMPI サポート有無	JVMPI 無し	JVMPI 有り
サイズ	2,798k バイト	2,850k バイト

7 考察・結言

JavaVM の内部状態の監視、及びメソッドの実行監視を行う JVMPI 準拠のエージェントを、組み込み Java™ に対応させた。そして、プロトタイプ実装を通じて、実行時の監視機能について、速度面、メモリ消費、に関わるデータを得ることができた。

監視機能の追加にともない若干のオーバーヘッドが生じる結果となったが、必要機能とメモリ消費等のトレードオフを考慮し取捨選択が容易なので、柔軟にカスタマイズ可能であり、利便性は高い、と考える。

また、標準仕様 (JVMPI) の採用により、PC/WS の Java™ 用に開発した監視モニタを、組み込み Java™ 環境で活用できた。その結果、高い移植性を達成でき、システム全体の設計を低工数でまとめることができた。標準仕様をベースに設計することにより、実用性の高いものができたと考える。

今後、さらなる機能の追加ならびに高速化を検討し、実用化に努める。また、2章でとりあげた他の高信頼化機能に対しても、あわせて検討をすすめる。

8 参考文献

- [1] <http://www.super-h.com/>
- [2] <http://java.sun.com/products/cdc/>
- [3] Sun, "Java Virtual Machine Profile Interface", <http://java.sun.com/j2se/1.3/ja/docs/ja/guide/jvmpi/>
- [4] <http://www.hitachi-ul.co.jp/SH-SE/>

Java 及びすべての Java 関連の商標およびロゴは、米国及びその他の国における米国 Sun Microsystems, Inc. の商標または登録商標です。その他記載の会社名、製品名は、それぞれの会社の商標もしくは登録商標です。