

D-Script: 分散環境における障害管理を行うスクリプト言語基盤

養安元気^{††} 岡本悠希[†]
菅谷みどり^{††} 倉光君郎^{††,†††}

1. はじめに

従来のディペンダブルシステムは、想定されるフォルトに対して、その対策を事前に作り込む手法で開発されてきた。しかし、実際に稼働しているシステムは絶えず更新されるため、事前に想定することの難しい障害が発生する可能性がある。例えば、新しく導入したシステムからフォルトが発生し、そのフォルトが伝播し障害になった場合である。このような環境では、従来のような事前の障害対策を行うだけでは、新たに発生した障害に対応することが難しい。

したがって、我々は上記のような環境ではフォルトやモニタリング結果からフィードバックをもらい、管理システムの監視対象を動的に変更を行えるようなモデルが望ましいと考えた。

JST/DEOS プロジェクトでは、「ディペンダブル組み込みオペレーティングシステム」の研究開発に端を発し、オープンシステムにおけるディペンダビリティをマネジメントする新しい DEOS プロセス/DEOS アーキテクチャの構築を進めている。D-Script は、DEOS プロセス/DEOS アーキテクチャにおけるスクリプト言語基盤であり、システム運用時に発生するシステム障害に対するマネジメントを外部記述し、安全に実行させる手段を提供することを目指している。¹⁾ D-Script の目的は D-Script で書かれたアプリケーションの信頼性を高めることではなく、あくまでも、既存のシステム、ミドルウェア、アプリケーションを外部から制御し、信頼性を高めることである。

2. Script of Script

従来、スクリプト言語は、システムの再構成を補助

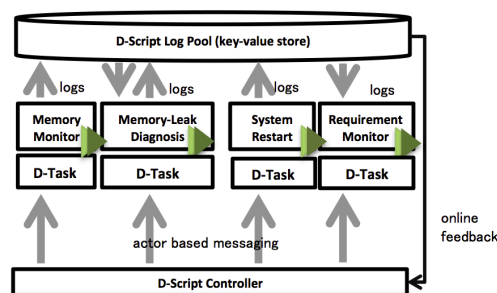


図 1 D-Script の設計図

する言語として活用されてきた。特徴は、グルー言語としての役割であり、さまざまな外部機能（コマンド、サービス）を呼び出すことで、言語の拡張性を高めていることにある。D-Script のアイデアも、外部機能として障害管理に必要となる、例えばモニター機能やライフサイクル管理（再起動、更新）などの個別の外部呼び出しの機能を、スクリプトで連携させ、統合的な管理を行うことである。スクリプトで用意された外部機能をスクリプトにより記述することから、我々はこれを Script of Script と呼んでいる。

D-Script では、従来の外部機能を利用するためのライブラリ群である Low-Level Script を用意する。ユーザは障害管理に必要な、複数の Low-Level Script を組合わせて、管理の対象となるノードに対して、処理 (D-Task) を記述する。D-Task は、Low-Level Script を組み合わせたスクリプトが記述されるが、その働きにより、監視、解析、回復、対応など異なる役割を持つ。この時、監視から回復までの一連の D-Task 群は、Scenario に従って動作する。そして、それぞれの障害に対し、Scenario の D-Task を追加/変更を加えて対応する。

3. 設計/実装

D-Script は図 1 のような構成になっており、以下のようなライフサイクルを繰り返し、システムの変更に対応する。

[†] 横浜国立大学

Yokohama National University

^{††} 横浜国立大学大学院

graduated Yokohama National University

^{†††} 日本科学技術振興機構

JST/CREST

- (1) D-Script Controller は各ノードに対して、D-Task を送る。
- (2) D-Script LogPool は各ノードから出たログを集約し、一つのストレージに記憶する。
- (3) ログの解析を行う。
- (4) 解析結果を元に、ユーザが新たにモニターや回復処理をノードに行わせる。

なお、D-Script Controller はアクターベースで D-Task をノードに送る。また、D-Task は D-Script Controller に対して、成功、失敗の真偽値を返し、取得した情報は D-Script LogPool に書き込む。D-Script LogPool は Key Value Store でログを保存する。key にはログのメタ情報とユニークな値を記述し、解析処理の時に利用する。実装例として vmstat を用いたモニタリングの例を list.1 に示す。DVMStat(vmstat の出力結果を LogPool に書き込む処理) をサブプロセスとして実行する。引数はそれぞれ、ログに対応させるキーの値、ログの書き込み先、インターバル時間 (sec) となっている。

Listing 1 vmstat

```
1 Subprocess p = System.startDVMStat("KEY-NAME", "MEMCACHED", 1);
2 if (isError(p)) {
3   p.kill();
4 }
```

4. 今後の展望

現在の D-Script はサーバマシンなどの OS 上で動作する環境をターゲットにしている。今後の発展としては、D-Script による SMNP などネットワークプロトコル通信の実装を行い、スイッチングハブなどのハードウェアに対してもモニタリングを行えることを目指す。

5. まとめ

本研究では、障害の検知から障害解析、回復までの一連の作業を支援するシステム D-Script の基本的な構成について述べた。また、事前に知ることのできない障害に対しても、D-Task を追加/更新させることで、対応方法の概念を示した。

謝辞 本研究は、JST/CREST 「実用化を目指したディペンダブル組み込みオペレーティングシステム」の研究として実施されている。

参考文献

- 1) Tokoro, M: Dependable Embedded Operating System Project White Paper version 3.0, October 2011.

- 2) 養安元気, 若森拓馬, 中田晋平, 菅谷みどり, 倉光君郎: D-Script フレームワーク: 分散システムを統一的に管理, 運用するためのフレームワーク, PPL2012(2012).