

4R-7

可視化ツールVickey V3-(2)

堀田勇次^{*}†, 市川 至†, 鈴木忠道‡, 清家健志‡, 灰塚凡樹‡
 †(株)富士通研究所, ‡富士通株式会社

1. はじめに

情報化社会の発展にともない、ソフトウェアの開発量および種類が増大し、信頼性、生産性の向上が必要となっている。我々は、抽象度の高い図形情報を活用しプログラム動作の理解を支援する可視化の技術を用いた、通信ソフトウェアの試験工程における異常診断を支援するツールとして、「可視化ツールVickey」[1][2]の研究を行っている。本発表では、主にVickey V3の時間制御における、ログ記録から表示データを復元する場合の問題点とその解決手法について述べる。

2. 可視化ツールVickey

可視化ツールVickeyは、対象システム上のソフトウェアのふるまいについてのログ記録を解析し、人間に理解し易い形で可視化表示するツールである。異常診断支援は、バグによる異常走行時にとられたログを本ツールで可視化することで行われる。すでに、プロトタイプ版としてVickey 1号, 2号を試作し、UNIXのプロセス管理動作についてのログについて可視化実験を行っている。

この結果、表示時間制御では、表示される対象の時間をVTR的な操作により推移させて、動的な状態変化を動的に表示(プロトタイプVickey 2号による例を図1に示す)したり、現在・過去・未来について時系列的に表示(図2)することが、対象の動作の理解支援において有効であることが判明した。図1. プロトタイプでの動的表示例

現在、実機よりのログを扱う、より実用的なプロトタイプとしてVickey V3を作成している。Vickey V3の基本的な構成は、Vickey 1号, 2号を引き継いでおり、すなわち、対象のログを解析し、可視化表示する。Vickey V3は、将来の拡張性を持たせるために、表示モデルに基づいて実現を行っている[2]。

Vickey V3には時系列表示の機能があり、ある中心時間の前後についても表示することになる。さらに、時間制御においては、表示単位の時間間隔を変更する機能があ

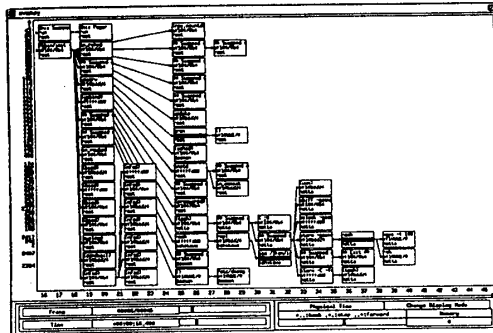


図1. プロトタイプでの動的表示例

A Visualization Tool 'Vickey' - (2)

Yuji HOTTA¹, Itaru ICHIKAWA¹, Tadamichi SUZUKI², Takeshi SEIKE²,
 Tsuneki HAIZUKA²

1:Fujitsu Laboratories LTD., 2:Fujitsu LIMITED.

り、異常箇所の絞り込みには有効と思われる、また、ある時刻の状態を動的に表示する場合にも、表示要素の最適配置には前後の時間での変化が加味される。このため、表示の中心となる時間の状態だけでなく、その前後のある一定範囲の時間それぞれの状態、さらには、指定する任意の時間について、対象の状態を得る必要がある。

Vickey V3では、未加工のログを直接的・逐次的に解析しながら各時間の状態を得るのではなく、前もって生のログを前処理的にある程度解析し、ログから得られた対象の状態履歴を内部表現化して記録し、この内部記録から各時間の状態を得るようにした。

3. ログ記録の内部表現化

Vickey V3が扱うログには、(1)ある時間におけるシステムの部分状態、(2)ある時間におけるシステムに発生したイベント、が記録されているものとする。

この対象についてのログ記録については、

「ログを取るにより対象の動作に影響が出ない」と仮定している。これはVickey V3が扱うログが、メモリダンプやトレーサの出力として、従来の異常診断支援において用いられてきたデータであるためである。

Vickey V3では、ログをとる影響で再現性が変化してしまうような微妙なタイミングが問題となる場合は、エミュレータやシミュレータにより、対象から得るログと同じフォーマットのログを生成させ、それを可視化することで対処する。

さらに、異常箇所をどのようにログの中に記録するかについては、ログをとる際の問題として、Vickeyの外で対処することにする。

Vickey 2号においては、ログを記録する時点で、ある程度の解析をして、(1)初期状態、(2)前に記録した時間の状態との間での双方向性の差分、という形式で記録を行った。これにより、連続的な、再生、逆方向再生をスムーズに行うことが可能であった。

しかし、早送り、早戻し、単位時間の変更、任意の指定時間への中心時間の推移、などの表示時間制御機能に対しては、方向性の差分による記録上で逐次的に時間を進めていく方式では処理時間がかかり、表示の連続性に難があった。

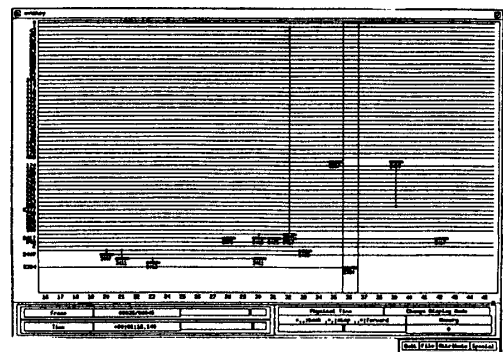


図2. プロトタイプでの時系列表示例

Vickey 2号は基本機能を検討するプロトタイプであったため、この問題は重要ではなかったが、Vickey V3では、扱うログに記録されたデータは、その対象の数、その時間単位の細かさ、その時間範囲の長さ、においてVickey 2号よりはるかに多く、復元を逐次的に行っていく方式では復元処理の速度が問題となる。

また、Vickey V3が扱うログには、初期状態を得るために必要な、観測以前についての記録がなされていない。そこでVickey V3では；

- まず、ログを解析し、記録開始時点のいわば初期状態を復元する処理を行う。
- そして、各時間における完全状態を解析により得、その結果を用いて、次章に述べる方法で差分化して内部記憶を行う。

というアプローチをとることにする。

4. 差分記録

Vickey V3のために任意の時点の完全状態を復元する方法について、以下の2つのアルゴリズムを検討した。

4.1 部分的に完全情報を記録する方法

この方法では、差分情報だけでなくデータ中に一定間隔で、その時点の完全な情報を記録しておく(図3)。連続的に状態を復元する場合は、現在保持している状態と読み込んだ差分情報とを用いて次の時点の状態を復元する。

任意時間の状態を復元する場合は、復元する時間に最も近い位置にある完全情報を読み込み、そこから目標の位置に向かって状態の復元を進める。

この方法は簡単に実現できる割に速度的な効果は大きい。一定間隔で完全な情報を記録する必要があるため、保持するデータが大きくなる欠点がある。

特に、完全情報を用意する間隔を広いものにする、Vickey 2号で発生した同じ問題を抱えることになる。

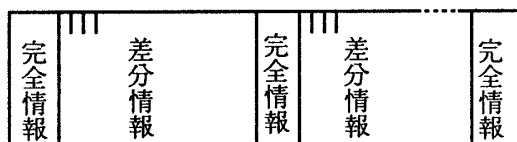


図3. 部分的に完全情報を記録する方法

4.2 階層的に差分を記録する方法

この方法では、連続して差分情報を記録するのではなく、図4に示したように、幾つかの階層に分けてその階層内での差分情報を記録する。このとき上下の階層間には、「下位の階層の代表のデータが直接の上位の階層のデータそのものである」という関係がある。連続的に状態を復元する場合は、内部的には階層の数の状態を記憶しておくことと効率よく復元できる(図4の場合3個)。

記録した全ての状態を連続的に復元する時は、例えば図4の場合、最初の完全な情報1を読み込み(この時点で三つの階層分の状態S1~S3を記憶する)、最下位の階層の1に連なる2, 3, 4を順に読み込んで1から4の状態を復元する。ここで第三階層の最初の集団を読み終わったので、記憶している三つの状態のうち状態S2(第二階層の状態)の状態に対して、5を読み込んで5の状態を復元し、状態S2および状態S3とする。こうして5, 6, ...と読み進み、状態S3の復元を進める。

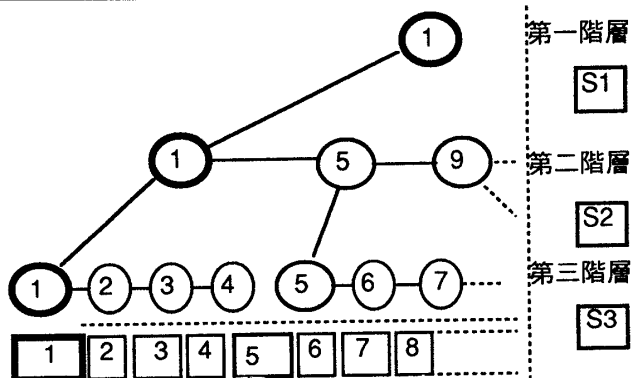


図4. 階層的に差分を記録する方法

任意時間の状態を復元する時は、例えば図4の6の状態を復元する場合、状態S3が第三階層の5の集団のいずれかの情報を記憶していれば、その集団中の差分情報から復元することで容易に6の状態を得られる。また状態S3が第三階層の5の集団中の状態を記憶していない場合は、状態S2を用いて第二階層の状態5を復元してその状態を状態S3とし、状態6を復元する。

この方法は、階層を構成する木を何進木にするかにもよるが、任意時間の状態を復元する場合は一般的に4.1の方法よりも高速になることが期待できる。しかし、上の階層になるにつれ、対象についての離れた時間の差分となり、必ずしも連続した差分情報を記録する訳ではない。このため、対象のデータによっては、全く異なる情報となり、差分をとり情報を圧縮する意味をなさないこともありうる。

Vickeyにおいては、通常、完全に状態についての情報を復元するのは、ある一定範囲でよい。情報の復元において、連続する最大範囲は、固定範囲内となる。4.1の方法は、この範囲内およびその近傍に常に完全情報が含まれるような間隔にすれば、連続する範囲内の復元が高速になる。4.2の方法も、この範囲内のある階層以下の復元作業になれば、同様に高速に処理することができる。

しかし、[2]で述べた、単位時間を変更する場合を考えると、任意時間を得るのに高速な4.2の方法に利点がある。さらに、4.2の方法で、最下層ではなく中間の層のレベルで復元すると、単位時間を拡大した場合の状態の復元が簡単に実現できる。

そこで、4.2の方法を基本にした差分化により、ログを内部表現化することにする。

5. まとめ

可視化ツールVickey V3の時間制御における、ログ記録から表示データを復元する場合の問題点とその解決手法について述べた。

Vickey V3の表示時間制御機能を実現するために、対象のログを差分技術を用いて内部表現化する。この結果、任意時間の状態へのアクセスは、必要に応じて復元することで実現される。

今後はさらに実現を進め、実用化を図るとともに、設計工程との関係を進める予定である。

[参考文献]

- [1] 市川, 堀田, 毛利, 灰塚: 「システムソフトウェアの可視化に関する一考察」, 情報処理学会大40回全国大会, 1R-7, p.1012, 1990.
- [2] 市川, 堀田, 清家, 鈴木, 灰塚: 「可視化ツールVickey V3 - (1)」, 本全国大会予稿中, 1991.