

3D-7

COBOLデバッガでのDB/DC単体テスト機能

今城 哲二*1

(日立製作所 ソフトウェア工場)

高見沢 正己*2, 森 陽子*2, 加藤 豊*2 (日本コンピュータ研究所)

1. はじめに

近年、高級言語によるオンラインプログラム作成が増加すると共に、DB/DC(Data Base/Data Communication)の需要も高まり、DB/DC環境を使用したテストも増加してきた。

このDB/DC環境がテストの段階で完成されていない場合は、テスト用データベース(以降、DBと記述する)や端末シミュレータの作成が必要となり、これに費やす労力は多大なものとなる。

ここで、DB/DC環境がなく、テスト用DBや端末シミュレータもない状態で、テストを可能とする機能が求められるようになった。

今回、日立のCOBOLデバッガ(COBOL言語のテスト支援ツール)で、これらのDB/DCの環境がなくてもDB入出力やメッセージ送受信が可能なDB/DC単体テスト機能を開発した。

本稿では、DB/DC単体テスト機能の特長と実現方法を中心に述べる。

2. 特長

COBOLプログラムで、DB/DCを使用したユーザアプリケーションプログラム(以降、UAPと記述する)を作成してテストする場合、図1に示すように、DBと端末を使用するDB/DCシステムが必要となる。これを、COBOLデバッガで行

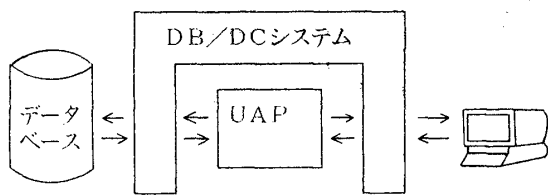


図1. DB/DCシステムとUAPの構成

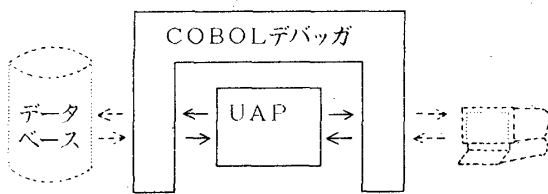


図2. COBOLデバッガとUAPの構成

うと、図2に示すようにDBも端末も不要となる。すなわち、DB/DC単体テスト機能では、COBOLデバッガとUAPだけでテストできるため、対話(TSS)環境やバッチ環境で容易にテストできる。

DB入出力やメッセージ送受信は、COBOLプログラム実行前にシミュレートサブコマンド(註1)で、入力や受信エリアの値の設定と、出力や送信エリアの値の表示を設定しておくことにより、簡単にシミュレートできる。

シミュレートサブコマンドは、予めライブラリに登録しておくことにより、プログラム実行前に毎回入力しなくても展開することができる。

(注1) : ユーザのデバッガに対する指示をサブコマンドという。

3. 機能

(1) DBシミュレートサブコマンド

DBシミュレートサブコマンドは、DBのレコードビュー対応に定義でき、開始サブコマンドと終了サブコマンドの間にシミュレーション情報を記述する。

図3に、GET命令とMODIFY命令に対するシミュレーションの使用例を示す。

GETとMODIFY命令を、レコードビュー対応にDBSIMとENDDBSIMサブコマンドで定義し、命令実行単位にRVIEWとENDRVIEWサブコマンドでシミュレート内容を設定する。

例では、1回目のGET命令でレコードビューエリアに100を設定し、2回目のGET命令で200を設定する。MODIFY命令を実行すると、レコードビューエリアBの値を表示する。

このように、GETとMODIFY命令が実行されるたびに対応するRVIEWサブコマンドを実行することで、DBと入出力を行っているかのような働きをさせる。

この方法で、DBのシミュレーションを実現している。

(2) 端末シミュレートサブコマンド

端末シミュレートサブコマンドは端末対応に定義でき、開始サブコマンドと、終了サブコマンドの間にシミュレーション情報を記述する。

図4に、RECEIVE命令とSEND命令に対するシミュレーションの使用例を示す。

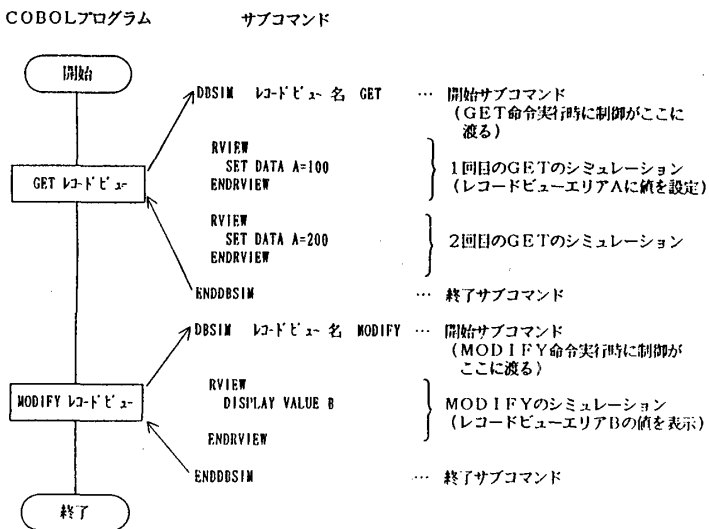


図3. 使用例

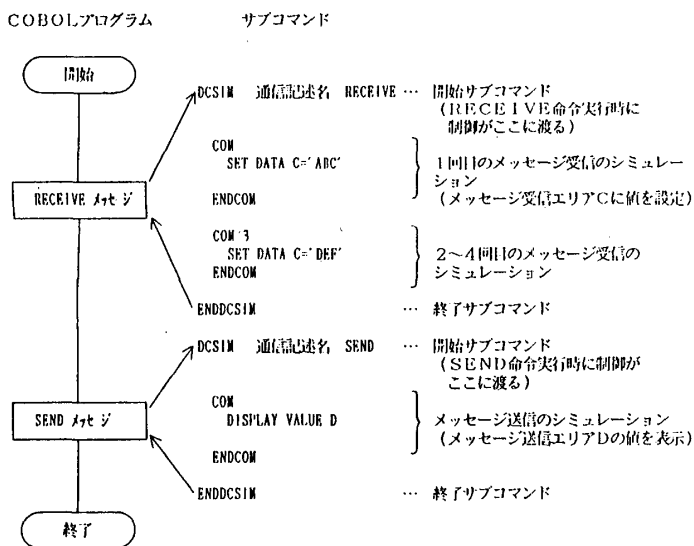


図4. 使用例

RECEIVEとSEND命令を、通信記述名対応にDCSIMとENDDCSIMサブコマンドで定義し、命令実行単位にCOMとENDCOMサブコマンドでシミュレート内容を設定する。

例では、1回目のRECEIVE命令でメッセージ受信エリアCに'ABC'を設定し、2~4回目で'DEF'を設定する。SEND命令を実行すると、メッセージ受信エリアDの値を表示する。

このように、RECEIVEとSEND命令が実行されるたびに対応するCOMサブコマンドを実行することで、端末と送受信を行っているかのような働きをさせる。

この方法で、端末のシミュレーションを実現している。

4. 実現方法

データベース入出力やメッセージ送受信命令の、シミュレート実行時の制御の流れを図5に示す。

(1) シミュレートサブコマンドでシミュレート内容を設定後、COBOLプログラムを開始する。

(2) COBOLプログラムで、DB入出力やメッセージ送受信命令が実行されると、COBOL実行時ルーチン(注2)に制御を渡す。

(3) COBOL実行時ルーチンは、COBOLデバッグに制御を渡す。

(4) COBOLデバッグはシミュレート対象命令かどうか判定する。シミュレート対象であれば、対応するシミュレートサブコマンドを実行する。

(5) COBOL実行時ルーチンに制御が戻ると、シミュレート実行済みか判定する。シミュレート未実行のときは、DB/DCシステムに制御を渡し、DBの入出力やメッセージの送受信を行う。この方法で、DB/DC単体テスト機能を実現している。

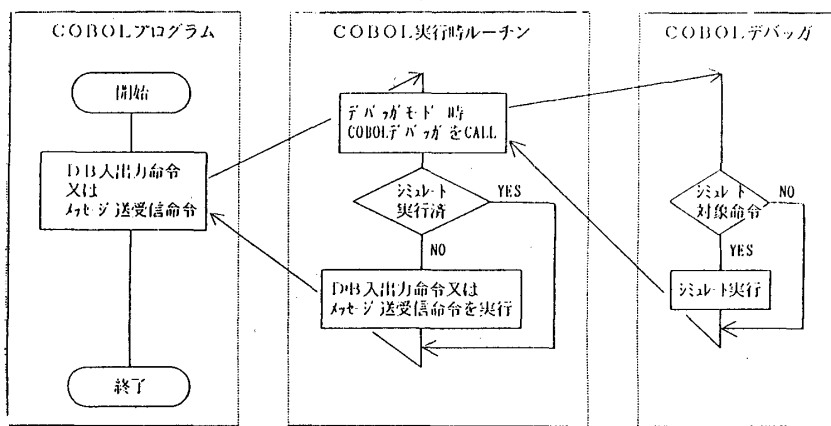


図5. DB/DC単体テスト機能の制御の流れ

(注2) : COBOL実行時ルーチンとは、COBOLプログラムと、COBOLデバッグとのインタフェースをとるプログラムである。

5. おわりに

本稿で述べたDB/DC単体テスト機能でシミュレートする送受信メッセージは、COBOL言語で定義する01レベルの論理レコードに対応した形式で実現した。今後は、オンライン端末自体を対話(TSS)端末でシミュレートすることにより、オンライン端末上での操作性テストを可能とするなどのエンハンスが必要と考えている。