

HITAC L-330エンドユーザー機能 —ワークステーションと簡易オンラインシステム—

End User Facilities of HITACHI Computer System HITAC L-330 (Work Station and Easy-Simple Online System)

電子計算機が普及するに従い、その直接のユーザーであるエンドユーザー向けの機能が重視されつつある。HITAC L-330のオペレーティングシステムであるVOS 0では、エンドユーザー向け機能としてワークステーション機能と簡易オンライン機能を開発した。ワークステーション機能により、端末からのプログラムの作成と実行、端末画面を使用した問い合わせとデータ入力を可能とした。また簡易オンライン機能を使用することにより、伝票発行を含む事務処理を主体としたオンラインシステムの開発を、エンドユーザー部門が行なえるようにした。これらのエンドユーザー向け機能の効果を更に高めるため、汎用の簡易言語“NHELP”を提供し、システム全体を簡易言語だけで運用することを可能とした。

秋田英彦* Akita Hidehiko
 館上 章* Tachigami Akira
 斉木吉彦* Saiki Yoshihiko
 石塚忠嗣* Ishizuka Tadatsugu

1 緒 言

電子計算機システムが普及するに従って、エンドユーザーすなわち電子計算機を自らの業務に直接利用するユーザーが、プログラムの開発やシステムの建設を行なう機会が増えつつある。エンドユーザーは一般には電子計算機による情報処理の専門家ではないため、理解しやすく使いやすいシステムを提供することが要請されている。HITAC L-330のオペレーティングシステムであるVOS 0 (Virtual Storage Operating System 0)でも、エンドユーザー向けの機能の強化を行なった。この論文では、エンドユーザー向け機能の設計思想とそれを実現した製品の仕様を中心に述べる。

タ処理の即時化、エンドユーザーの好み(いわゆるローカル色)が強調されてきている。事務管理部門でのバッチ処理ではタイミングよく必要な情報が得られないとか、あるいは逆に大規模オンライン処理では中央主導形でエンドユーザーごとに融通が利く処理ができないといった指摘がある。そのため、エンドユーザーがそれぞれ独自にシステムを管理したり、運用したりすることが増えてきている。そこに着目して、VOS 0ではエンドユーザー向け機能として、次のように三つの機能を提供した。

2 VOS 0 エンドユーザー向け機能の特長

顧客での電子計算機の使用形態が多様になるにつれ、デー

タ処理の即時化、エンドユーザーの好み(いわゆるローカル色)が強調されてきている。事務管理部門でのバッチ処理ではタイミングよく必要な情報が得られないとか、あるいは逆に大規模オンライン処理では中央主導形でエンドユーザーごとに融通が利く処理ができないといった指摘がある。そのため、エンドユーザーがそれぞれ独自にシステムを管理したり、運用したりすることが増えてきている。そこに着目して、VOS 0ではエンドユーザー向け機能として、次のように三つの機能を提供した。

第一に、従来からあるビルディングマシン、オフィスコンピュータでのデータエントリ機能、伝票発行機能に大容量のディスクファイルを接続し、しかも簡便で高度なデータ処理機能を付加する形で考えられたワークステーション機能がある。

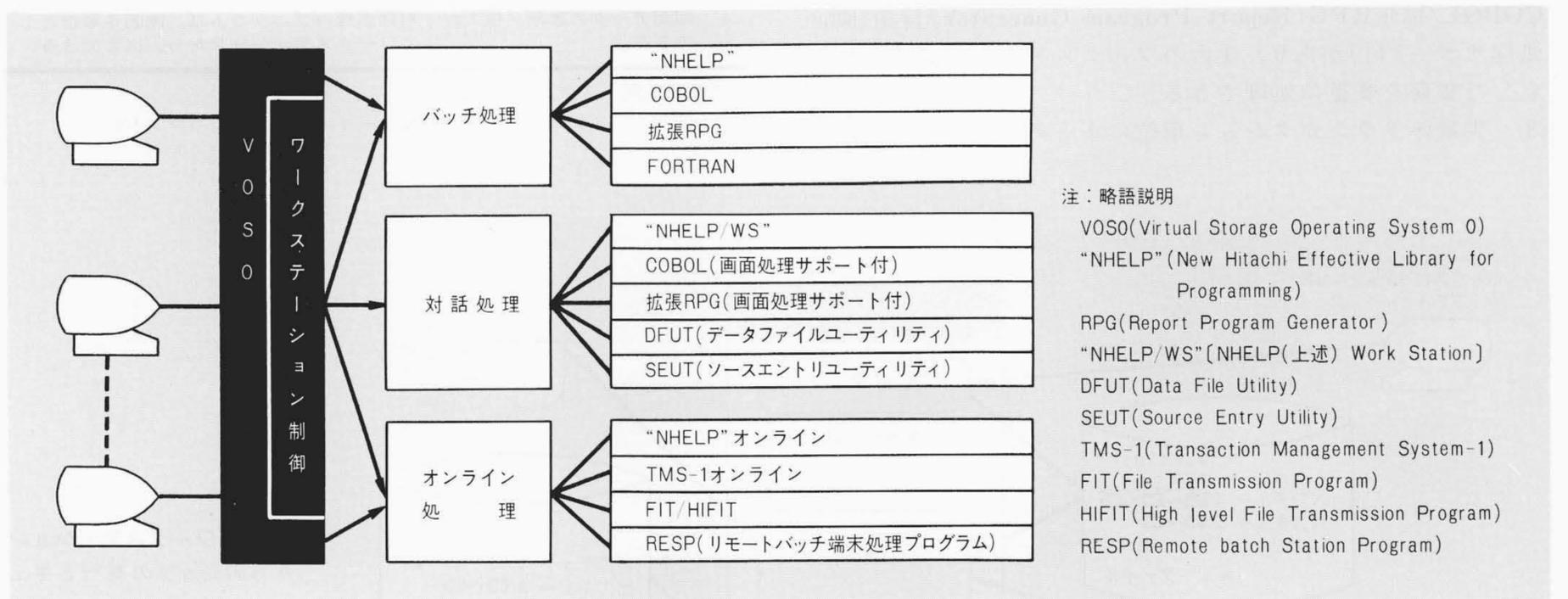


図1 VOS 0ワークステーションシステムの基本構成 VOS 0ワークステーションシステムでは、ワークステーションからバッチ処理、対話処理、オンライン処理のいずれかを選択して実行することができる。

* 日立製作所ソフトウェア工場

表1 ワークステーションシステムに対するニーズとVOS0での実現状況 システムの簡易化へのニーズに対し、VOS0では右側に記した機能を提供した。

ニ ー ズ	VOS0での実現	
1. プログラム開発 の省カラム	プログラミングを最小限に抑えて業務を進めたい。	豊富なサービスプログラム データエントリ用DFUT
	エンドユーザーでもプログラミングを簡単に行ないたい。	"NHELP" "NHELP/WS"
	プログラム作成、変更を手軽に行ないたい。	SEUT
2. エンドユーザーへのサービス性向上	対話処理で業務を進めたい。	"NHELP/WS" COBOL(画面処理サポート付) 拡張RPG(画面処理サポート付) "NHELP"オンライン
	画面を使用する業務処理の性能を向上させたい。	簡易画面アクセス法
	エンドユーザーが独立に業務を進めたい。	ワークステーションからのジョブ入力
	オペレーションを簡単にしたい。	PFK, ジョブメニュー プロシジャコマンド
3. システム運用の極小化	機器構成をできるだけ小さくしたい。	コンソールとワークステーションの兼用機能 ワークステーションプリンタ機能 ワークステーション切離し/接続機能 DFUT
	マスタファイルの更新に伴う障害対策は、システムに任せたい。履歴情報を残したい。	"NHELP"オンライン 画面印刷出力機能

注：略語・用語説明

- PFK(Program Function Keyの略であり、特別な機能をもったキー)
- プロシジャコマンド(ジョブ制御文をまとめて一つの機能を実行するコマンドにしたもの)
- ジョブメニュー(あらかじめ登録したジョブの一覧表)

ワークステーション機能のねらいは次のとおりとした。

- (1) ワークステーション装置をエンドユーザーがコンソール装置のように扱える。
- (2) データエントリ、問い合わせ応答、伝票発行に適した"NHELP/WS"(New Hitachi Effective Library for Programming/Work Station), DFUT(Data File Utility), COBOL/拡張RPG(Report Program Generator)言語(画面処理サポート付)があり、1台のワークステーションごとに独立した業務を多重に処理できる。
- (3) 少量のトランザクション単位にまとめて、バッチ処理を

行なえる。したがって、障害発生時のリラン対策、ファイルのバックアップ対策については通常の事務用バッチ処理と同じ方式—例えば3世代ファイル更新方式—を適用することができる。

第二に、従来からある中・大形オンラインシステムのうちからエッセンスだけを抽出して標準化を進め、簡易にしかも効率的なシステムを組み立てられるようにした"NHELP"オンラインがある。これはリラン/ファイル回復機能を内蔵しており、1台から16台までのワークステーション装置/オンライン端末装置を利用して多端末処理ができる。更に、ユーザー独自のデータ処理を"NHELP"言語で記述することとあいまって、小形システムでコンパクトな、しかもシステム仕様の明確な信頼できるオンラインシステムを建設できるようにしている。

第三に、簡易言語"NHELP"がある。バッチ処理、オンライン処理及び対話処理のすべての処理形態を"NHELP"言語により実現することができるようにした。

これら三つのエンドユーザー向け機能をもったシステムを、VOS0ワークステーションシステムと呼ぶ。図1は、その基本構成を示したものである。VOS0ワークステーションシステムのもとでは、従来からあるバッチ処理プログラム、オンライン処理プログラムに加え、画面を使用した対話処理プログラムをワークステーションから起動して業務処理を実行で

表2 VOS0ワークステーション機能の特長 ワークステーションからできる仕事の範囲は広い。

項 目	内 容
1. ジョブの起動/終了の指示ができる。	各ワークステーションごとに独立して、ジョブの起動及び終了の指示ができる(ジョブの内容に制約はない)。
2. あらゆる種類のコマンドの入力ができる。	システム制御コマンド、スプール制御コマンド及びプロシジャコマンドを入力できる。
3. 行単位にデータの表示/投入ができる。	通常のプログラムでは、ワークステーションに対して行単位にメッセージを表示したり、応答を受け付けることができる。
4. 画面データの表示/投入ができる。	対話処理用プログラムは、画面を単位としてワークステーションから入出力できる。

注：用語説明 スプール制御コマンド

(ジョブの入出力のキューイングに関するコマンド)

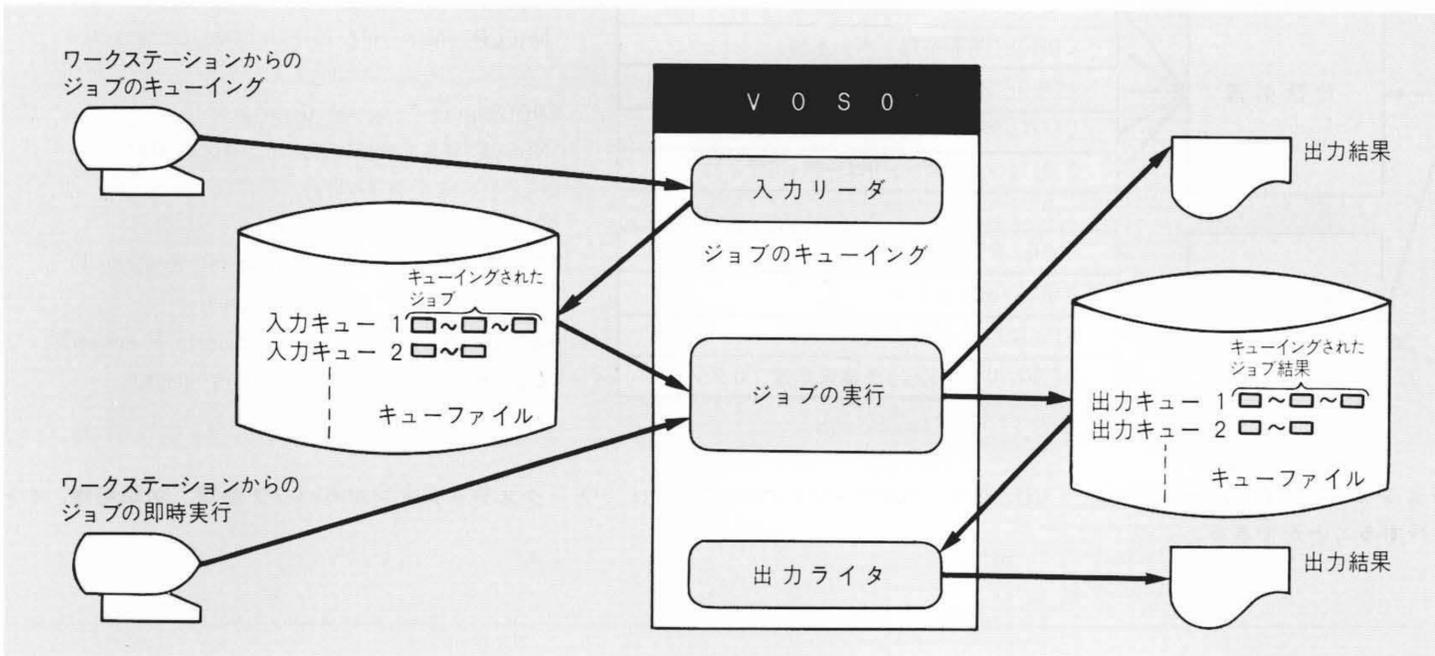


図2 ワークステーションからのジョブの実行とキューイング ワークステーションからのジョブの起動は、キューイングすることも即時に実行することもできる。ジョブの出力についても、即時出力とキューイング後の出力のいずれかができる。

きる。このため、長い歴史と経験から生まれたバッチ処理、オンライン処理のもつ長所をそのまま生かし、かつワークステーション装置を使用する便利な運用が可能となった。

表1は顧客の一般的ニーズとVOS0での実現状況をまとめたものである。

3 ワークステーション機能

ワークステーション機能の特長を表2に示す。

3.1 ワークステーションからのジョブの即時実行とキューイング

各ワークステーションから独立にジョブを入力し、即時に実行したり、キューイングした後、ジョブのスケジューリングに従って実行できる。この結果、同一のワークステーションから入力した複数のジョブを同時に実行できることになる。ジョブの実行結果は直接出力するか、又は一度キューイングした後出力ライタによって出力する。ワークステーションを使用したジョブの入出力の流れを図2に、またワークステーションからのジョブの入力方法を表3に示す。

3.2 コンソールとワークステーションの共用

VOS0のもとで複数のプログラムの実行を監視するために、コンソール装置が必要である。そこで、HITAC L-330に接続されている複数(又は1台)のワークステーションの中から、1台をコンソールとしても兼用できるようにした。このため、コンソール専用のディスプレイ装置は不要である。この場合、兼用するワークステーション装置では、ワークステーションから起動したプログラムとの対話用の画面と、コンソール装置としての監視用の画面を適宜切り換えながら使用することになる。

3.3 ワークステーションの切離し/接続機能

1プログラムで複数台のワークステーション装置をサービスできる“NHELP”オンラインでは、必要な時点でワークステーションディスプレイ及びプリンタ装置を切り離したり、再接続したりできる。このことにより、突発的に発生する業

表3 ワークステーションからのジョブの入力方法 ワークステーションからのジョブの入力方法は豊富である。ユーザーは自分にとって最も使いやすい方法を選択できる。

分類	入力方法	内容
1. ジョブの即時実行	あらかじめ登録済み(プロシジャ)の実行	PFK入力 プロシジャコマンドに対応付けられたPFKを押下することにより、あらかじめ登録されているプロシジャを呼び出す。
		プロシジャコマンド入力 ジョブの形であらかじめ登録されているプロシジャをコマンドにより呼び出す。
	任意のジョブを実行	MENUコマンド入力 あらかじめ作成されているジョブのメニューを画面に表示し、メニュー番号を指定してジョブを起動する。
		STARTコマンド+ジョブ制御文入力 STARTコマンドに続き、ジョブを実行するために必要なジョブ制御文を1枚ずつ入力する。その場で任意のジョブを作成できる。
2. ジョブのキューイング	@SRTコマンド+ジョブ制御文入力 @SRTコマンドで入力リーダを起動し、ジョブ制御文を1枚ずつ入力する。その場で任意のジョブを作成でき、ジョブの入力終了時にワークステーションは解放される。	
3. 入力キューからのジョブ実行	STARTコマンドによる入力キュー指定 入力キューを指定したSTARTコマンドによって、入力キューからジョブを取り出して実行する。	

注：略語説明 SRT(Start Reader or Writer)

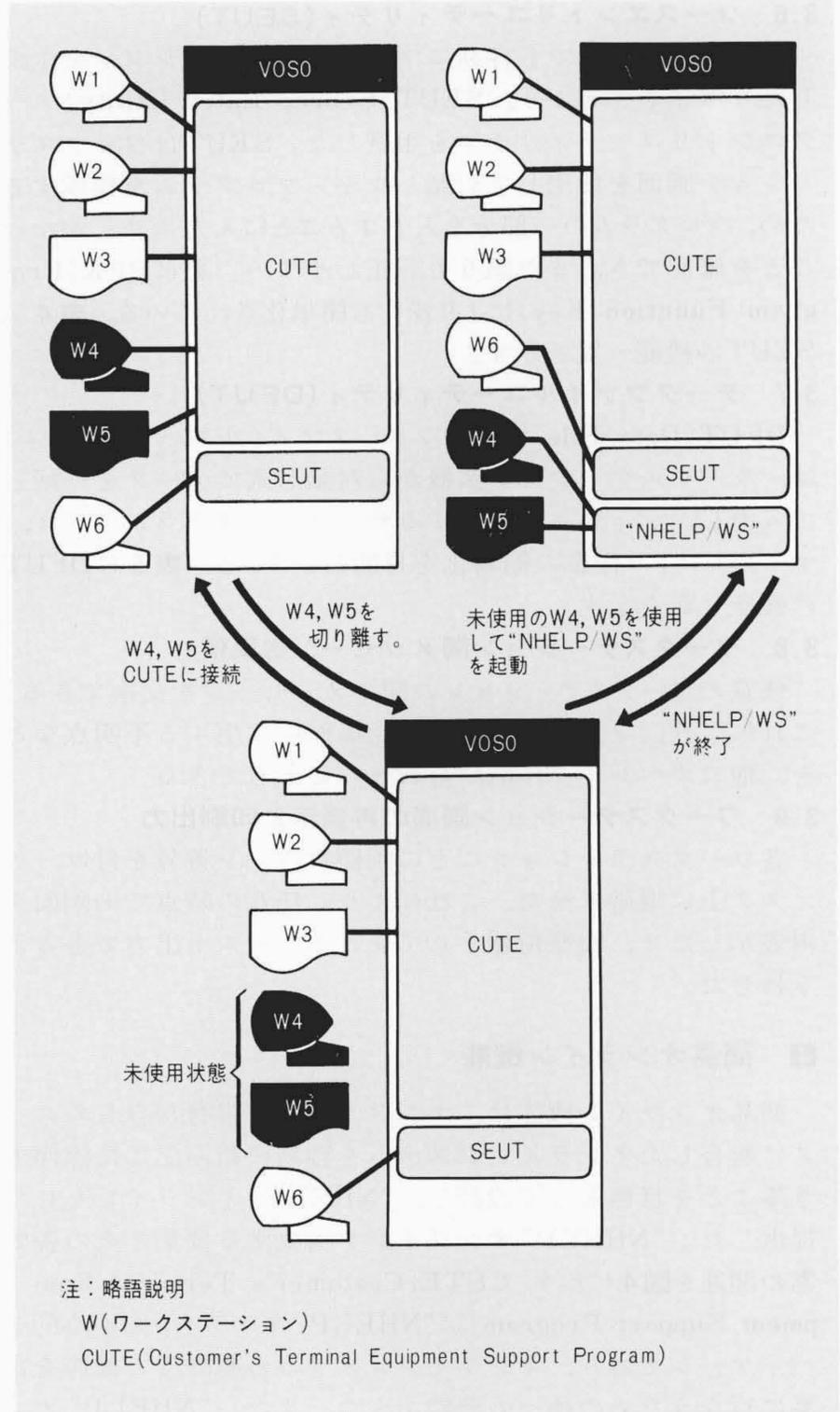


図3 ワークステーションの切離しと接続 オンラインシステムに属している一部の端末を切り離して対話処理を行ったり、それを再度もとの状態に戻したりできる。

務、処理件数が少ない業務専用ワークステーション装置を設置する必要をなくした。図3にワークステーションを切り離したり、接続したりする場合の例を示す。

3.4 ワークステーションプリンタ機能

LTC(Local Terminal Controller)を介してHITAC L-330に接続されたプリンタ装置を、ワークステーションプリンタ装置として使用できる。HITAC L-330から離れた場所に置いたワークステーションディスプレイ装置からジョブを入力して実行し、結果を近くにあるワークステーションプリンタ装置に出力できる。プリント出力方式には、出力ライタ経由のプリント方式とオンラインプログラムによる直接プリント方式の二通りあるが、この機能を使用すれば1台のプリンタ装置で両方式のプリント出力が可能である。

3.5 対話処理プログラムの作成

“NHELP/WS”, COBOL, 拡張RPGによって、ワークステーション画面を使用して対話形式でデータエントリ、問い合わせなどを行なう業務プログラムを作成できる。これらのプログラムは、ワークステーションから起動できる。

3.6 ソースエントリユーティリティ(SEUT)

エンドユーザーでも容易にプログラムやプロシジャを作成したり保守できるよう、SEUT(Source Entry Utility:ソースエントリユーティリティ)を用意した。SEUTはワークステーション画面を使用して対話しながらプログラムを作成するため、プログラムの一部分を入力するごとに入力エラーがないことを確認でき、また誤りも訂正しやすい。更にPFK(Program Function Key)により操作も簡単化されている。表4にSEUTの機能一覧を示す。

3.7 データファイルユーティリティ(DFUT)

DFUT(Data File Utility:データファイルユーティリティ)は、ワークステーション装置から対話形式でデータを一括して入力し、ディスクに格納するサービスプログラムであり、データエントリ作業の簡易化を目的としている。表5にDFUTの機能一覧を示す。

3.8 ワークステーション間メッセージ送受信

任意のワークステーションの間でメッセージを交換できる。これにより、ワークステーション操作中に生ずる不明点などを、他のオペレータに問い合わせることもできる。

3.9 ワークステーション画面の再表示と印刷出力

各ワークステーションごとに画面にページ番号を付け、ディスク上に退避させた。これにより、任意の時点で旧画面を再表示したり、履歴情報としてまとめてリスト出力できるようにした。

4 簡易オンライン機能

簡易オンライン機能は、エンドユーザー自身が自らのニーズに適合したオンラインシステムを容易に組み立てて操作できることを目標として設計し、“NHELP”オンラインとして提供した。“NHELP”オンラインを構成する要素とその各要素の関連を図4に示す。CUTE(Customer's Terminal Equipment Support Program)は“NHELP”オンラインの中心的なパッケージであり、オンラインシステムの組立てと操作を容易に行なうための種々の機能をもつ。また、“NHELP”オンラインの業務プログラムは簡易言語“NHELP”で作成される。

4.1 “NHELP”オンラインの組立て

“NHELP”オンラインの組立ては、CUTEの開始時にCUTEが自動的に行なう。ユーザーはシステムを構成するハードウェア機器のアドレスや使用する業務プログラムの名称などを、オンライン開始パラメータとして指定する。CUTEはパラメータを読み込み、その内容に適するシステムを組み立て、オンライン処理を開始する。

表4 SEUTの機能一覧 ディスプレイ端末を使用して原始プログラムやプロシジャを作成したり、保守したりするための豊富な機能が用意されている。

機能項目	内 容
1. 入 力	(1)原始文の集まりであるブックを新規に作成して、原始文ライブラリに登録する。 (2)ジョブ制御文の集まりであるプロシジャを新規に作成してプロシジャライブラリに登録する。
2. 更 新	既に作成済みのブック/プロシジャの文を変更/挿入する。
3. 削 除	ブック/プロシジャの文を削除する。
4. 移 動	ブック/プロシジャの文のある場所から削除して別の場所に移動する。
5. 複 写	ライブラリ中のブック/プロシジャの文を複写する。
6. ロールダウン	現在表示中の文から一定の数だけ後にある文を表示する。
7. 印 刷	(1)現在表示されている画面データをリスト出力する。 (2)SEUTの終了時にブック/プロシジャをリスト出力する。
8. 通し番号付け	SEUTの終了時にブック/プロシジャに通し番号付けを行なう。

注: 略語・用語説明

ブック(原始文ライブラリに登録された一つのプログラムをブックと呼ぶ)。
プロシジャ(ジョブ制御文をまとめたもの)
SEUT(Source Entry Utility)

表5 DFUTの機能一覧 ディスプレイ端末を使用して、ファイルの作成や保守を行なうための豊富な機能が用意されている。

機能項目	内 容
1. レコードの入力機能	(1)画面を通してフィールド単位にレコードを入力し、ディスク上に索引順編成ファイルを作成する。 (2)特定のフィールドを以前のレコードから自動的に複写する。 (3)入力フィールドの属性ごとにデータを編集する。 (4)入力データに対しモジュラス10/11の自己検査を行なう。
2. レコードのベリファイ機能	入力するレコードの比較、検証を行なう。
3. レコードの更新機能	作成済みのレコードをフィールド単位に更新する。
4. レコードの削除機能	作成済みのレコードを削除する。
5. レコードの挿入機能	作成済みの索引順編成ファイルのレコードの間に新しいレコードを挿入する。

注: 略語説明 DFUT(Data File Utility)

4.2 業務プログラムの端末入出力の簡易化

業務プログラムは、CUTEを介して図5に示すように端末との入出力を行なうことができる。業務プログラムがCUTEによって起動されたとき、端末から入力されたデータはB1と名付けられたエリアに置かれている。業務プログラムはこのことを前提として、面倒な入出力のプログラミングを行な

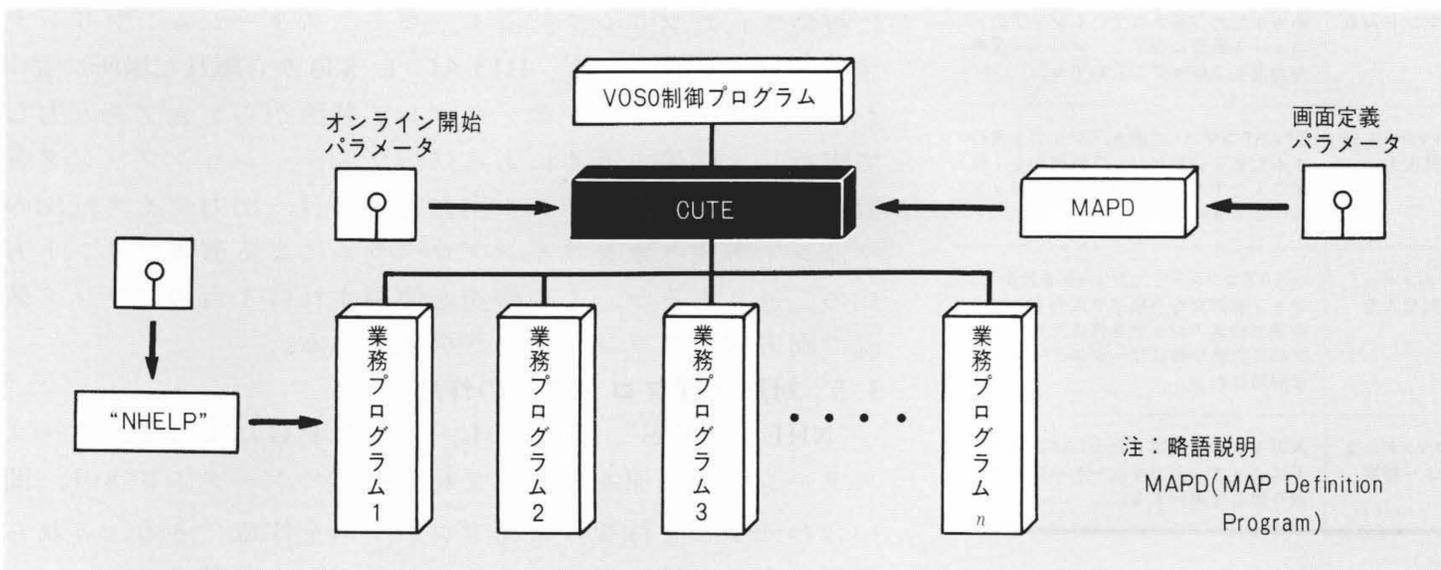


図4 “NHELP”オンラインの構成 オンライン処理 VOSO 制御プログラム, CUTE, 業務プログラムの組み合わせで行なう。“NHELP”及びMAPDはオンライン処理の実行を開始する前に、それぞれ画面情報の編集、業務プログラムの翻訳を行なう。

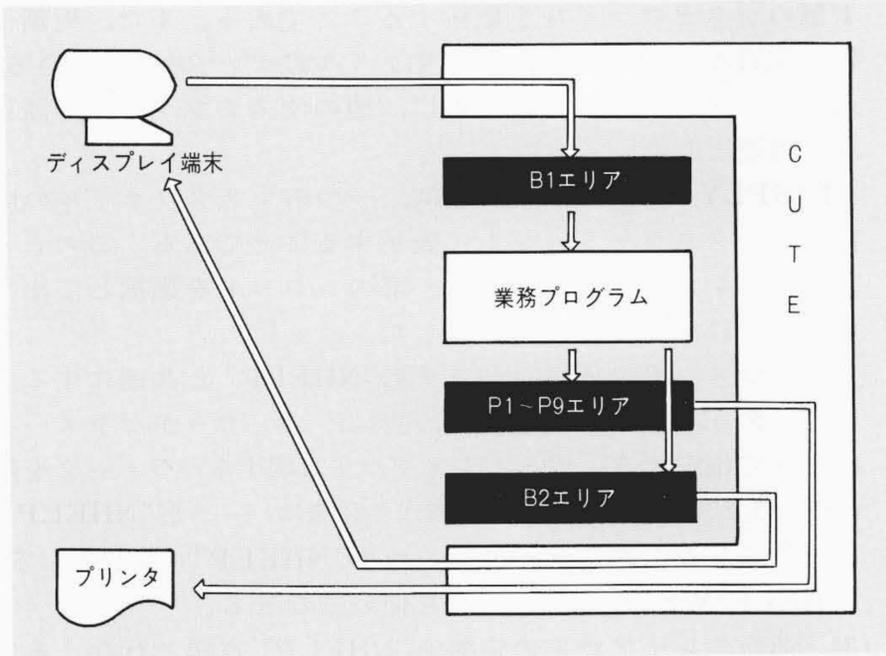


図5 業務プログラムの端末入出力 端末から入力されたデータは、B1エリアに置かれて業務プログラムに渡される。端末へ出力するデータはP1～P9エリア、B2エリアに作成してCUTEへ渡す。

わなくても、入力データの処理を行なえる。端末へデータを出力する場合は、プリンタ装置の場合はP1～P9、ディスプレイ装置の場合はB2と名付けられたエリアにデータを設定しておけば、CUTEが自動的に端末に出力する。このように、オンライン処理の詳しい知識がなくても、業務処理プログラムを作成できるのが特長である。

4.3 障害処理

CUTEはオンライン実行中、システムジャーナルと呼ぶ運転履歴情報を自動的に採取する。障害の発生によってシステムが停止したとき、CUTEはジャーナルを使用して、システムの状態を障害発生の直前の状態に戻し、各端末に処理の再開始点を連絡する機能(リラン機能と呼ぶ。)をもっている。また、ジャーナルにはファイル更新情報を含み、CUTEはこの情報を使用して処理の再開始点までファイルの状態を回復する。ユーザーはリラン機能を利用することにより、複雑な障害対策の準備をプログラムすることから解放される。

5 簡易言語による運用

VOS0のジョブ処理形態には、バッチ、オンライン及び対話の3種類がある。VOS0の下でこれらの処理形態を混在させることが可能なので、ユーザーは各業務に最適な処理形態を選択できる。これら三つの処理形態のすべてに対して簡易言語“NHELP”を提供しており、ユーザーは簡易言語だけで計算機システムを運用することができる。“NHELP”はHITAC 8150の“HELPII”(Hitachi Effective Library for Programming II), HITAC 8250の“HELP”を統合し、更に拡張した簡易言語である。

5.1 バッチ形“NHELP”

事務処理を行なう計算機のプログラムを分析すると、データのチェック、ソート、報告書の作成など基本的な機能の組み合わせから構成されていることが多い。個々のプログラムが異なっているのは、それらの機能の組み合わせ方法やデータの形式が異なっているためである。そこで、バッチ形“NHELP”では事務処理を18個の基本機能に分類し、各機能に対応して18個のルーチン(各機能に対応するプログラムをルーチンと呼ぶ。)を提供した。更に、基本機能以外の事務処理を行なうために1ルーチンを設けたので、バッチ形“NHELP”は全部で

19ルーチンから成る。表6にバッチ形“NHELP”のルーチンの名称と機能の一覧を示す。ユーザーは必要なルーチンを選択し、組み合わせることによって業務処理を行なうことができる。

各ルーチンの主要機能に対し、詳細処理はパラメータで指定する。パラメータの指定は、ユーザーごとのデータ形式の相違に備えると同時に、ユーザー固有の処理を可能にするためである。表7は“NHELP”の1ルーチンであるMATCH1ルーチンに対する代表的なパラメータを示したものである。“NHELP”のパラメータでは、項目の指定はその項目が属するレコードやエリア上での項目の開始位置(バイト位置)、長さ(バイト数)及びエリア名称によって行なう。これは“NHELP”の特長であり、直観的に理解しやすいものである。

5.2 “NHELP/WS”

バッチ形“NHELP”の考え方を対話処理に拡張したのが

表6 バッチ形“NHELP”のルーチン一覧 バッチ形“NHELP”は、事務処理を18個の基本機能に分類し、それぞれの機能に対してルーチンを提供している。また、基本機能から外れる機能に対して特別の1ルーチンを提供している。

ルーチン名	機能
INPUT1	入力データのチェック及び記録の変換を行なう。
PTCON1	紙テープデータを入力して他の媒体に変換する。
PTOUT1	紙テープデータの出力を行なう。
BRAIN1	入力データに四則演算を行なって出力データを作成する。
MATRIX	2次元配置表を作成するため準備を行なう。
SUMUP1	項目の合計を計算し、合計レコードを出力する。
MATCH1	マッチング処理を行ない、マッチング結果を出力する。
UPDATE	マッチング処理を行ない、マスタファイルを更新する。
MERGE1	二つのファイルを一つに統合する。
REPORT	一定の形式の報告書を作成する。
GRAPH1	棒グラフ又は点グラフを作成する。
INDEX1	レコードの索引となる項目の一覧表を作成する。
SEARCH	外部媒体上のテーブルを参照して入力データを加工する。
CHANGE	ファイルの編成を変更する。
DIVIDE	一つのレコードを二つ以上のレコードに分割する。
PRINT1	ユーザーの指定した形式の報告書を作成する。
GATHER	二つ以上のレコードを一つのレコードに統合する。
SORT	レコードをソートする。
BUILD1	基本機能に含まれない機能を実行する(ユーザーがプログラムのロジックを組み立てる)。

注：用語説明

マスタファイル(電子計算機により事務処理を行なうため、各種の基礎データを定常的に保管するファイル)

表7 MATCH1ルーチンのパラメータの例 ルーチンの処理詳細はパラメータで指定する。レコードやエリアの内の項目は、その項目の先頭からの位置(バイト位置)、長さ、エリア名称によって指定する。

パラメータ名	概要
@MKEY	機能 レコード中のキーの位置を指定する。
	使用例 @MKEY△△11△0002014
@ITEM	意味 マスタファイルのレコードのキーは、レコードの2バイト目から14バイトである。8桁目の1がマスタファイルかトランザクションファイルかの区別を指定している。
	機能 エリアからエリアへ項目を移動する。
@ITEM	使用例 @ITEM△△△△△000205001Z000205011Z
	意味 I1エリアの2バイト目から長さ50バイトをO1エリアの2バイト目から長さ50バイトへ移動する。データ形式は文字形式である。

注：I1エリアは、マスタファイルのレコードを読み込むエリアであり、O1エリアは、出力レコードを準備するエリアである。

“NHELP/WS”である。対話処理の利点はデータが発生した現場で即時に処理ができ、しかも画面を利用して投入したデータのチェックを行ないながら処理を進められる点である。この利点を最大限に活用できるように“NHELP/WS”は次の方針で設計した。

(1) バッチ形“NHELP”と組み合わせて使用することを前提として、画面を利用した対話処理向きの機能だけをサポートする。

バッチ形“NHELP”では18種類の基本機能を提供しているが、“NHELP/WS”は対話処理向きの機能だけをサポートする。また、バッチ形“NHELP”では各基本機能をルーチンに対応させているのに対し、“NHELP/WS”では対話処理向き機能を組み合わせた形で三つのパターンに分けて提供した。

図6に三つのパターンの入出力概要を示す。

ENTRYパターンの主機能は、データの入力及びチェックを行ない出力ファイルにデータを蓄積することである。入力データのチェックのために、マスタファイルの読み込みを可能とした。

MODIFYパターンの主機能は端末からの入力データにより

1個のマスタファイルを更新することである。また、更新情報の記録などのために、出力ファイルにデータを出力できる。入力データのチェックのために、他のマスタファイルの読み込みを可能とした。

DISPLYパターンの主機能は、一つのマスタファイルの内容をワークステーション上に表示することである。このとき表示したレコードの中から、一部のレコードを選択して出力ファイルに出力できるようにした。

(2) パラメータの仕様はバッチ形“NHELP”と共通にする。

各パターンの処理の具体的な内容は、ユーザーがパラメータによって指定する。画面のレイアウトに関するパラメータを除いて、ほとんどのパラメータの形式と意味はバッチ形“NHELP”と同じにした。ユーザーはバッチ形“NHELP”の延長として、“NHELP/WS”のパラメータを使いこなせる。

(3) 画面のレイアウトの定義を“NHELP”言語で行なえるようにする。

ディスプレイ端末の画面はフィールドに分割して使用する。従来、画面上のフィールドのレイアウトの定義は、画面定義プログラムに対するパラメータの形で行なっていたが、“NHELP/WS”では@SCR(Screen), @FLD(Field), @DATAという3種類のパラメータを設け、“NHELP”言語により画面定義を簡易に行なえるようにした。@SCRパラメータは画面全体の性質を定義し、@FLDパラメータは個々のフィールドを定義する。@DATAパラメータは@FLDパラメータと組み合わせてフィールドに初期値を与える。

なお、“NHELP/WS”を使用して作成した出力ファイルを入力としてバッチ処理を行ないたい場合は、ワークステーションからバッチ形“NHELP”を起動して処理を行なうことができる。

5.3 “NHELP”のオンライン機能

VOS 0のオンラインは4章で述べたとおり、CUTEを使用して簡易に建設することができる。このオンラインシステムの業務プログラムは、“NHELP”言語で作成する。バッチ形“NHELP”は19個のルーチンから成るが、その他にオンライン業務プログラム作成用としてMPPGEN(Message Processing Program Generator)ルーチンを設けた。MPPGENルーチンに対しパラメータを与えることにより、オンライン用業務プログラムが生成される。MPPGENルーチンのパラメータは、他のルーチンと同じパラメータにオンライン処理特有のパラメータを追加したものであるため、ユーザーはバッチ処理の拡張としてオンライン用業務プログラムを作成することができる。

6 結 言

以上、HITAC L-330 VOS 0 エンドユーザー向け機能の設計思想と、それを実現した製品の仕様について述べた。この論文で述べた機能は現在ほぼ開発を完了し、顧客への提供が開始されている。エンドユーザー向け機能を拡充してゆくことは今後のソフトウェアの大きな課題の一つであり、実用システムでの経験を反映して更に簡易化を図ってゆく考えである。

参考文献

- (1) 岡田, 外: HITAC L-330システム, 日立評論, 60, 12, 881~886 (昭53~12)

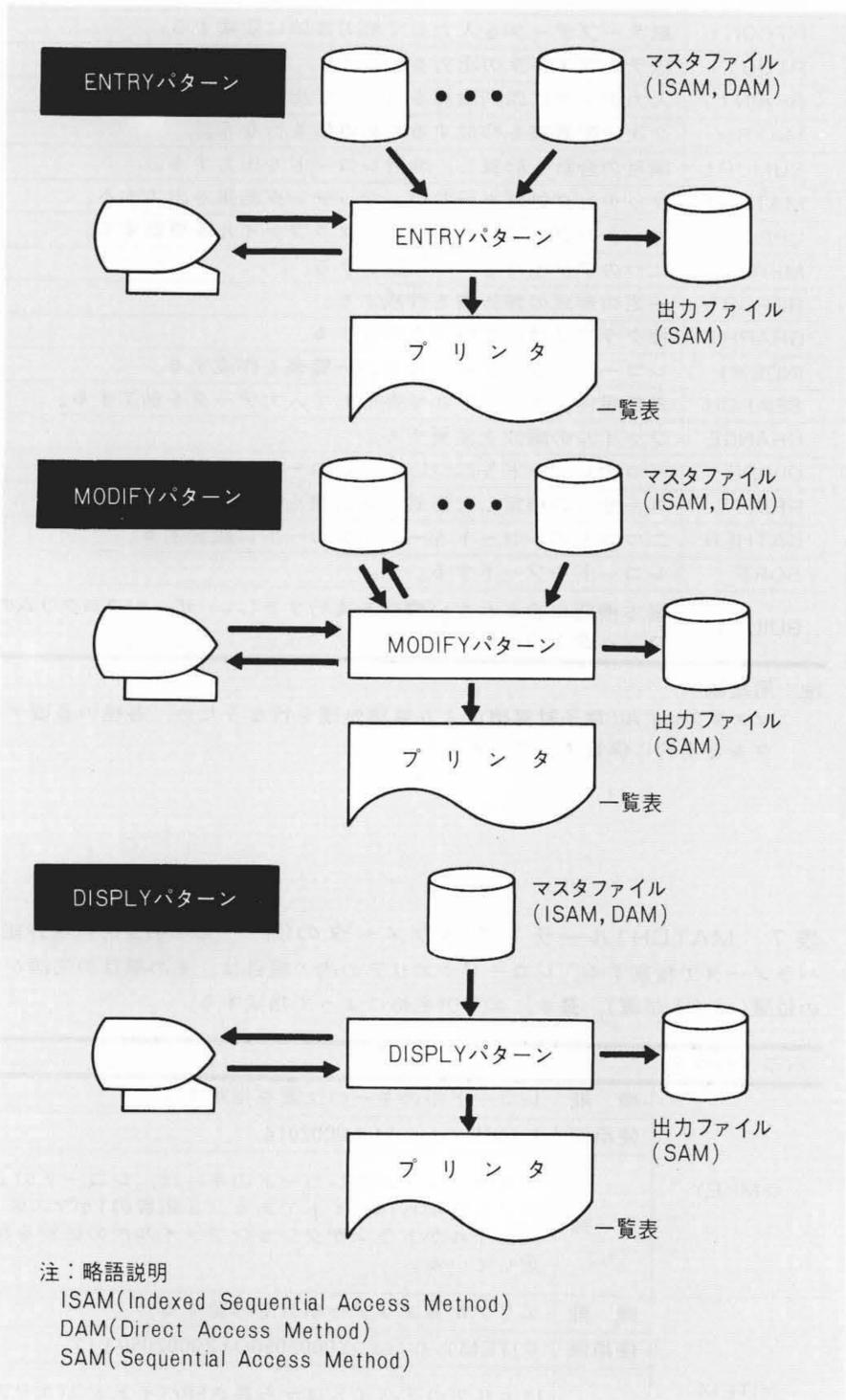


図6 “NHELP/WS”の三つのパターン “NHELP/WS”には三つの処理パターンが用意されており、これらを利用して容易に対話処理プログラムを作成することができる。