



Sun™ Mainframe Transaction Processing ソフトウェア 構成 ガイド

Release 8.1.0

Sun Microsystems, Inc.
www.sun.com

Part No. 819-2515-10
2005 年 6 月, Revision A

コメントの送付: <http://www.sun.com/hwdocs/feedback>

Copyright 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします)は、本書に記述されている技術に関する知的所有権を有しています。これら知的所有権には、<http://www.sun.com/patents>に掲載されているひとつまたは複数の米国特許、および米国ならびにその他の国におけるひとつまたは複数の特許または出願中の特許が含まれています。

本書およびそれに付属する製品は著作権法により保護されており、その使用、複製、頒布および逆コンパイルを制限するライセンスのもとにおいて頒布されます。サン・マイクロシステムズ株式会社による事前の許可なく、本製品および本書のいかなる部分も、いかなる方法によっても複製することが禁じられます。

本製品のフォント技術を含む第三者のソフトウェアは、著作権法により保護されており、提供者からライセンスを受けているものです。

本製品の一部は、カリフォルニア大学からライセンスされている Berkeley BSD システムに基づいていることがあります。UNIX は、X/Open Company Limited が独占的にライセンスしている米国ならびに他の国における登録商標です。

本製品は、株式会社モリサワからライセンス供与されたリュウミン L-KL (Ryumin-Light) および中ゴシック BBB (GothicBBB-Medium) のフォント・データを含んでいます。

本製品に含まれる HG 明朝 L と HG ゴシック B は、株式会社リコーがリョービマジクス株式会社からライセンス供与されたタイプフェイスマスタをもとに作成されたものです。平成明朝体 W3 は、株式会社リコーが財団法人日本規格協会 文字フォント開発・普及センターからライセンス供与されたタイプフェイスマスタをもとに作成されたものです。また、HG 明朝 L と HG ゴシック B の補助漢字部分は、平成明朝体 W3 の補助漢字を使用しています。なお、フォントとして無断複製することは禁止されています。

Sun, Sun Microsystems, Java, AnswerBook2, docs.sun.com, JVM は、米国およびその他の国における米国 Sun Microsystems 社の商標もしくは登録商標です。サン・ロゴマークおよび Solaris は、米国 Sun Microsystems 社の登録商標です。

すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャーに基づくものです。

ORACLE は Oracle 社の登録商標です。PostScript は Adobe System 社の商標または登録商標で、特定の法的管轄下で登録されていることがあります。

OPENLOOK, OpenBoot, JLE は、サン・マイクロシステムズ株式会社の登録商標です。

ATOK は、株式会社ジャストシステムの登録商標です。ATOK8 は、株式会社ジャストシステムの著作物であり、ATOK8 にかかる著作権その他の権利は、すべて株式会社ジャストシステムに帰属します。ATOK Server/ATOK12 は、株式会社ジャストシステムの著作物であり、ATOK Server/ATOK12 にかかる著作権その他の権利は、株式会社ジャストシステムおよび各権利者に帰属します。

本書で参照されている製品やサービスに関しては、該当する会社または組織に直接お問い合わせください。

OPEN LOOK および Sun™ Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザーおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカル・ユーザー・インターフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

U.S. Government Rights—Commercial use. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われぬものとします。

本書には、技術的な誤りまたは誤植のある可能性があります。また、本書に記載された情報には、定期的に変更が行われ、かかる変更は本書の最新版に反映されます。さらに、米国サンまたは日本サンは、本書に記載された製品またはプログラムを、予告なく改良または変更することがあります。

本製品が、外国為替および外国貿易管理法 (外為法) に定められる戦略物資等 (貨物または役務) に該当する場合、本製品を輸出または日本国外へ持ち出す際には、サン・マイクロシステムズ株式会社の事前の書面による承諾を得ることのほか、外為法および関連法規に基づく輸出手続き、また場合によっては、米国商務省または米国所轄官庁の許可を得ることが必要です。

原典:	Sun™ Mainframe Transaction Processing Software Configuration Guide Part No: 817-7433-10 Revision A
-----	--



Adobe PostScript

目次

はじめに xvii

1. Sun MTP 構成の概要 1
 - 構成に関するタスク 1
 - Sun MTP 標準規格 2
 - 画面の形式 3
 - ファンクションキー 4
 - データ入力の検査 4
 - ファイル識別子 5
2. 領域環境変数の設定 7
 - 領域構成に関するタスク 7
 - ディレクトリ構造の作成 8
 - 環境変数 9
 - 標準環境変数 9
 - Sun MTP Secure 環境変数 17
 - Sun MSF 環境変数 19
 - 領域環境の定義 19
 - 設定ファイルの定義と実行 20

- 3. Sun MTP の起動と停止 23
 - Sun MTP サーバーの起動 23
 - ▼ サーバーを起動する 23
 - ローカル端末クライアントの起動 24
 - ▼ ローカル接続を確立する 24
 - トランザクションの実行 25
 - ▼ トランザクションを実行する 25
 - 端末セッションの停止 25
 - Sun MTP サーバーの停止 26
 - kixstop による領域の停止 27
 - クライアント接続 28
- 4. 構成ガイドライン 31
 - Sun MTP トランザクション処理プログラム 32
 - トランザクション処理プログラムの数の決定 33
 - バックグラウンドタスクの最大数の決定 34
 - バッチジョブの最大数の決定 34
 - 照会ジョブの最大数の決定 34
 - 回復による構成への影響 35
 - Sun MTP プロセス数の決定 35
 - ローカルプロセスメモリー 36
 - Solaris プラットフォーム 37
 - AIX プラットフォーム 38
 - AIX の大規模なアドレス空間モデル 40
 - 共有メモリー 40
 - 共有メモリーに必要な容量の計算 41
 - 動的共有メモリー 41
 - Solaris での動的共有メモリーのサイズ 42
 - AIX での動的共有メモリーのサイズ 42

共有メモリーの追加先	43
共有ライブラリの使用	43
メモリープロセスマップ	44
VSAM バッファープールおよび VSAM バッファ어의数の設定	46
スワップ空間の決定	48
カーネルのパラメータ	48
カーネルのパラメータの変更	49
コードページ変換テーブルの使用	52
▼ 領域のコードページ変換テーブルを選択する	52
▼ TN3270 エミュレータのコードページ変換テーブルを選択する	53
1 バイト文字のコードページ変換テーブルの例	53
5. アプリケーションリソースの定義	55
Table Manager の開始	58
▼ Table Manager を開始する	58
▼ 「Standard Tables」メニューを表示する	59
VCT のカスタマイズ	59
▼ VCT をカスタマイズする	59
PCT でのトランザクションの定義	62
▼ トランザクションを定義する	62
FCT での VSAM ファイルの定義	64
▼ ファイルを定義する	65
PPT でのリソースの定義	67
▼ オンラインプログラムとマップを定義する	67
SIT のカスタマイズ	69
▼ SIT をカスタマイズする	69

- 6. 回復の構成 75
 - VCT の回復の構成 75
 - ▼ 回復を構成する 75
 - VSAM データセットの回復の構成 76
 - ▼ KSDS および RRDS ファイルに回復を指定する 77
 - ESDS データセットの回復の構成 78
- 7. 代替リソース定義 79
 - リソースタイプとコマンド 79
 - サポートするコマンド 80
 - 代替リソース定義への領域の有効化 81
 - ▼ rdo グループを定義する 81
 - リソースの構成 82
 - 領域の起動時のリソースの有効化 82
 - ▼ 起動時にリソースを有効化する 82
 - リソースの管理 83
 - 代替リソース定義を使用する長所と短所 83
- 8. UNIX 端末の構成 85
 - キーファイル 85
 - kixkeydef を使用したキーファイルの作成または編集 86
 - ▼ kixkeydef を使用してキーファイルを作成または編集する 87
 - ▼ Return キー、改行キー、およびバックスペースキーを定義する 88
 - キーファイルの直接編集 89
 - Sun MTP X クライアントの定義 92
 - ASCII 端末の初期化 93

- 9. プリンタの構成 95
 - デフォルトのプリンタの構成 95
 - ▼ デフォルトのプリンタを構成する 96
 - 追加プリンタの構成 97
 - 一時データキューを使用するプリンタの構成 97
 - ▼ DCT に TDQ をプリンタとして定義する 97
 - ▼ プリンタと TDQ を関連付ける 99
 - LU タイプ 1 プリンタの構成 100
 - ▼ LU タイプ 1 プリンタを構成する 101
 - プリンタサイズの指定 102
 - kixprint によるプリンタの構成 102
 - kixprint シェルスクリプトの変更 103
- 10. システム間通信 (ISC) の構成 107
 - サポートされる ISC 機能 107
 - SNA の領域の構成 108
 - 通信マネージャーの設定 108
 - ▼ unikixdcl を開始するように unikixCommMgr を構成する 109
 - 遠隔システムの定義 110
 - ▼ 遠隔システムを定義する 110
 - TCP/IP の領域の構成 113
 - ▼ TCP サーバー (unikixtcp) を構成する 113
 - トランザクションセキュリティー 116
 - 遠隔領域の自動インストール 116

- 11. 遠隔クライアントへの Sun MTP の構成 117
 - TN3270 クライアントへの領域の構成 117
 - ▼ TN3270 サーバーを構成する 118
 - ▼ TN3270 クライアントを定義する 121
 - ▼ TN3270 クライアントから領域に接続する 122
 - SNA 3270 クライアントの領域の構成 123
 - TPS PU4/5 サーバーの構成 124
 - PU2 構成 125
 - PU4/5 構成 127
 - TPS PU4/5 SNA サーバーと動作する領域の構成 128
 - ▼ 領域を構成する 128
 - ECI/EPI クライアントの領域の構成 133
 - ▼ ECI/EPI クライアントを受け入れる領域を構成する 133
 - ECI/EPI クライアントの自動インストール 134
 - TCP/IP ソケットリスナーの起動 134
 - ▼ unikixsock プロセスを開始する 134
 - ソケットのタイムアウトの設定 135
 - SSL クライアントの領域の構成 135
 - ▼ SSL 要求を受け入れる領域を構成する 135
 - 認証データベースの設定 136
 - unikixssl サーバーへの通信マネージャーの設定 136
 - ▼ unikixssl を開始するように unikixCommMgr を構成する 136
 - ▼ 領域と SSL サーバーを起動する 140
 - WebSphere MQ クライアントの領域の構成 141
 - ▼ unikixqm サーバーを起動するために unikixCommMgr を構成する 141

- 12. Sun MBM と機能する Sun MTP の構成 143
 - VCT での Sun MBM の有効化 144
 - ▼ VCT で Sun MBM を有効にする 144
 - VSAM ファイルの再使用不可の定義 146
 - ▼ File Manager を使用して VSAM ファイルを再使用不可に定義する 146
 - ▼ ユーティリティーを使用してファイルを再使用不可に定義する 147

- 13. RDBMS と動作する領域の構成 149
 - RDBMS 環境変数の設定 149
 - kixinstall を使用した RDBMS の選択 150
 - ▼ RDBMS を選択する 150
 - SIT での RDBMS 値の指定 151
 - ▼ SIT で RDBMS 固有の値を設定する 151
 - Oracle に関する注意 152
 - Sybase に関する注意 153
 - 領域と RDBMS の起動 153
 - ▼ 領域と RDBMS を起動する 153
 - ▼ 起動時に RDBMS をバイパスする 153

- 14. DBCS のサポートの有効化 155
 - 前提条件 155
 - 領域での DBCS の有効化 156
 - ▼ DBCS のサポートを有効にする 156
 - DBCS と ISC の使用 157
 - データ変換テンプレートの定義の準備 157
 - ▼ データ変換テンプレートを定義する 158
 - トランザクション経路指定 162
 - DBCS コードページ変換テーブル 163
 - 使用上の考慮点 164

A. キーボードの割り当て	165
NCD 101 キーキーボード	166
NCD 97 キーキーボード	167
VT タイプのキーボード	168
Sun のキーボード	169
デフォルトの X キーボード	170
デスクトップ端末キーボード	171
用語集	173
索引	185

図目次

図 1-1	Sun MTP 画面形式の例	3
図 2-1	財務アプリケーションのディレクトリ構造の例	8
図 4-1	トランザクション処理プログラムの構造	32
図 4-2	Solaris のメモリーのイメージ	44
図 4-3	AIX のメモリーのイメージ	45
図 5-1	「Table Manager」 メインメニュー	58
図 5-2	「Standard Tables」 メニュー	59
図 5-3	VSAM 構成テーブル (VCT)	60
図 5-4	プログラム管理テーブル (PCT)	62
図 5-5	ファイル管理テーブル (FCT)	65
図 5-6	処理プログラムテーブル (PPT)	67
図 5-7	システム初期化テーブル (SIT)	70
図 6-1	VSAM 構成テーブル (VCT)	76
図 6-2	ファイル管理テーブル (FCT)	77
図 8-1	kixkeydef 画面	87
図 9-1	TCT - Printers 画面	96
図 9-2	DCT での一時データキュー (TDQ) の定義	98
図 9-3	TDQ とプリンタとの関連付け	99
図 9-4	TCT - Printers 画面 - 新規エントリの表示	100
図 9-5	TCT - 3270 Devices 挿入画面	101

図 10-1	TCT - System Entries 挿入画面	111
図 10-2	TCT - System Entries 挿入画面	114
図 11-1	TCT - 3270 Devices 挿入画面	121
図 11-2	単一ホストの PU5 構成	124
図 11-3	2 ホストの PU5 構成	125
図 11-4	TCT - 3270 Devices 挿入画面	129
図 11-5	TCT - 3270 Devices のホストとポートの画面	131
図 12-1	Sun MBM と接続した VSAM データセットへのアクセス	144
図 12-2	VSAM 構成テーブル (VCT)	145
図 12-3	File Manager - 再使用不可のファイルの定義	147
図 13-1	RDBMS Configuration 画面	150
図 13-2	システム初期化テーブル (SIT)	151
図 14-1	データ変換テンプレートテーブル (CVT)	158
図 14-2	CVT の挿入画面	159
図 14-3	CVT 詳細画面 - ページ 1	160
図 14-4	CVT 詳細画面 - ページ 2	161
図 14-5	CVT 詳細画面 - キーの定義	162

表目次

表 1-1	画面の形式	3
表 1-2	ファンクションキー	4
表 1-3	ファイル識別子	5
表 2-1	標準の Sun MTP 環境変数	9
表 2-2	Sun MTP Secure 環境変数	18
表 2-3	Sun MSF 環境変数	19
表 3-1	Sun MTP へのクライアント接続	28
表 4-1	トランザクション処理プログラムの数の決定	33
表 4-2	共有メモリー領域	41
表 4-3	VSAM バッファおよびバッファープール	47
表 4-4	プロセス間通信のパラメータ	49
表 4-5	カーネルのパラメータ	50
表 5-1	アプリケーション要件と関連テーブル	56
表 7-1	CEDA コマンド	80
表 7-2	dfhusdup ユーティリティーコマンド	80
表 7-3	リソースをロードする際の unikixmain オプション	83
表 7-4	従来環境と混合環境の相違点	84
表 8-1	キーボード定義のエスケープシーケンス	90
表 8-2	TERM=xterms のキーボード割り当て	90
表 8-3	3270 キーボード機能	91

表 10-1	Sun MTP でサポートされる ISC 機能	107
表 10-2	unikixrc.cfg ファイル - DCL サーバーのエントリ	110
表 11-1	unikixrc.cfg ファイル - TN3270 サーバーのエントリ	119
表 11-2	サポートする ECI/EPI クライアント	133
表 11-3	unikixrc.cfg ファイル - SSL サーバーエントリ	137
表 11-4	unikixrc.cfg ファイル - WebSphere MQ サーバーエントリ	142
表 13-1	SIT の RDBMS 値	152
表 14-1	DBCS 変換テーブル	163

コード例

コード例 4-1	ユーザーの制限	51
コード例 8-1	kixkeydef ユーティリティで作成されたキーファイル	89
コード例 9-1	変更していない kixprint シェルスクリプト	103
コード例 9-2	変更された kixprint シェルスクリプト	105
コード例 11-1	sna.cnf.pu2 構成ファイル	125
コード例 11-2	自動ログイン構成ファイル	126
コード例 11-3	リスナー構成ファイル	126
コード例 11-4	sna.cnf.pu45 構成ファイル	127
コード例 11-5	Ethernet 構成ファイル	127
コード例 14-1	データセットのリソースのコピーブック	157

はじめに

このマニュアルでは、Sun™ Mainframe Transaction Processing ソフトウェア (Sun MTP) の管理者、システムプログラマ、またはその他の個人ユーザーを対象としています。このソフトウェアを実行するオペレーティング環境および IBM CICS 環境の構成を熟知しているものとします。このマニュアルを使用する前に、Sun MTP および実行するその他のソフトウェア、たとえば COBOL および Sun Mainframe Batch Manager ソフトウェア (Sun MBM) をインストールし、サンプルアプリケーションを使用してインストールを確認してください。

マニュアルの構成

第 1 章では、構成タスクとユーザーインターフェースについて説明します。

第 2 章では、領域を定義する環境変数と、それらの設定方法について説明します。

第 3 章では、領域を起動する方法、ローカル接続を実現する方法、および領域を停止する方法について説明します。

第 4 章では、トランザクション処理プログラムの構成、回復、およびメモリーに関するガイドラインについて説明します。

第 5 章では、システムテーブルを使用してアプリケーション環境を設定する方法について説明します。

第 6 章では、領域全体の回復を構成する方法および VSAM ファイルごとに回復を構成する方法について説明します。

第 7 章では、システムを設定して、リソース定義ファイルを使用する方法について説明します。

第 8 章では、さまざまなタイプの UNIX® 端末を使用する方法について説明します。

第 9 章では、プリンタの設定方法について説明します。

第 10 章では、領域を設定し、TCP/IP と SNA を介して遠隔領域との通信を確立する方法について説明します。

第 11 章では、SNA 3270 デバイスや TN3270 クライアントなど、遠隔クライアント接続を受け入れるように領域を設定する方法について説明します。

第 12 章では、Sun MBM とともに機能するように領域を設定する方法について説明します。

第 13 章では、リレーショナルデータベース管理システム (RDBMS) を扱えるように領域を設定する方法について説明します。

第 14 章では、2 バイト文字セット (DBCS) をサポートして領域を実行する方法について説明します。

付録 A では、さまざまな UNIX システムとパーソナルコンピュータのキーボード機能が、どのように 3270 キーボードに割り当てられているかについて記述しています。

UNIX コマンド

このマニュアルには、システムの停止、システムの起動、およびデバイスの構成などに使用する基本的な UNIX[®] コマンドと操作手順に関する説明は含まれていない可能性があります。これらについては、以下を参照してください。

- 使用しているシステムに付属のソフトウェアマニュアル
- 下記にある Solaris[™] オペレーティングシステムのマニュアル

<http://docs.sun.com>

シェルプロンプトについて

シェル	プロンプト
UNIX の C シェル	<i>machine_name%</i>
UNIX の Bourne シェルと Korn シェル	\$
スーパーユーザー (シェルの種類を問わない)	#

書体と記号について

書体または記号*	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例。	.login ファイルを編集します。 ls -a を実行します。 % You have mail.
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して表します。	% su Password:
<i>AaBbCc123</i>	コマンド行の可変部分。実際の名前や値と置き換えてください。	rm <i>filename</i> と入力します。
『 』	参照する書名を示します。	『Solaris ユーザーマニュアル』
[]	参照する章、節、または、強調する語を示します。	第 6 章「データの管理」を参照。 この操作ができるのは「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅を超える場合に、継続を示します。	% grep `^#define \ XV_VERSION_STRING`
[]	省略可能な項目を示します。	unikixmain [-Q]
{ }	セパレータ () で区切られた代替オプション	kixfile [-r{Y N}]
	区切り文字 (セパレータ) です。この文字で分割されている引数のうち 1 つだけを指定します。	EXEC CICS READ DATASET FILE

* 使用しているブラウザにより、これらの設定と異なって表示される場合があります。

このマニュアルでは、次の書式を使用してコマンドを表記します。

```
$ command required-argument [optional-argument]
```

コマンドに省略可能な引数が記述されていない場合は、そのコマンドを入力して Return キーを押します。

関連マニュアル

製品	タイトル	Part No.
Sun Mainframe Transaction Processing ソフトウェア	『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』	819-2514-10
	『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』	819-2516-10
	『Sun Mainframe Transaction Processing ソフトウェア インストールガイド』	819-2517-10
	『Sun Mainframe Transaction Processing ソフトウェア メッセージガイド』	819-2518-10
	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』	819-2519-10
	『Sun Mainframe Transaction Processing ソフトウェア 障害追跡とチューニング』	819-2520-10
	『Sun Mainframe Transaction Processing ソフトウェア XA リソースマネージャの使用』	819-2358-10
Sun Mainframe Administration Tool (Sun MAT)	『Sun Mainframe Transaction Processing ソフトウェア ご使用にあたって (Solaris プラットフォーム用)』	819-2521-10
	『Sun Mainframe Administration Tool ユーザーズガイド』	819-2523-10

製品	タイトル	Part No.
Sun Mainframe Batch Manager ソフトウェア	『Sun Mainframe Batch Manager ソフトウェア 構成ガイド』	819-2505-10
	『Sun Mainframe Batch Manager ソフトウェア インストールガイド』	819-2506-10
	『Sun Mainframe Batch Manager ソフトウェア メッセージガイド』	819-2507-10
	『Sun Mainframe Batch Manager ソフトウェア 移行ガイド』	819-2508-10
	『Sun Mainframe Batch Manager ソフトウェア リファレンスマニュアル』	819-2360-10
	『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』	819-2509-10
	『Sun Mainframe Batch Manager ソフトウェア ご使用にあたって (Solaris プラットフォーム用)』	819-2510-10
Sun Cluster 用の 高可用性エージェ ント	『Sun Mainframe Transaction Processing ソフトウェア 高可用性 (HA) データサービス (Sun Cluster 用)』	819-2522-10
	『Sun Mainframe Batch Manager ソフトウェア 高可用性 (HA) データサービス (Sun Cluster 用)』	819-2511-10
	『Sun Mainframe Security Facility 高可用性 (HA) データサービス (Sun Cluster 用)』	819-2512-10
Sun Mainframe Security Facility	『Sun Mainframe Security Facility 管理者ガイド』	819-2359-10
	『Sun Mainframe Security Facility ご使用にあたって (Solaris プラットフォーム用)』	819-2513-10
IBM CICS	『CICS アプリケーション・プログラミング・リファレンス』	SC33-1170
	『CICS アプリケーション・プログラミング・ガイド』	SC33-0674
	『CICS Master Index』	SC33-1074
	『CICS Supplied Transactions』	SC33-1686
	『CICS System Programming Reference』	SC33-1689
Server Express	Server Express のマニュアル	*
ACUCOBOL-GT	ACUCOBOL-GT のマニュアル	*
Open PL/I	『Liant Open PL/I User's Guide』	*
	『Liant Open PL/I Language Reference Manual』	*
	『Liant CodeWatch Reference Manual』	*
C	C コンパイラのマニュアル	*
C-ISAM	『C-ISAM Programmer's Manual』	*
	『System Performance Tuning』、Mike Loukides 著、砂原秀樹監訳、株式会社アスキー発行、1995	

* これらのマニュアルは、使用するプラットフォームによって異なります。プラットフォームに該当するマニュアルについては、ご購入先にお問い合わせください。

Sun のマニュアルの注文方法

日本語版を含め、Sun のマニュアルは次のサイトで、表示や印刷、または購入ができます。

<http://www.sun.com/documentation>

Sun の技術サポート

この製品に関して、このマニュアルでも解決しない技術的な質問がある場合は、次のサイトからお問い合わせください。

<http://www.sun.com/service/contacting>

コメントをお寄せください。

マニュアルの品質改善のため、お客様からのご意見およびご要望をお待ちしております。コメントは下記よりお送りください。

<http://www.sun.com/hwdocs/feedback>

ご意見をお寄せいただく際には、下記のタイトルと Part No. を記載してください。

『Sun Mainframe Transaction Processing ソフトウェア 構成ガイド』, Part No. 819-2515-10

第1章

Sun MTP 構成の概要

アプリケーション環境はそれぞれ異なるので、Sun MTP の導入も環境ごとに独自のものになります。このマニュアルでは、基本的な構成を実行する手順について説明します。ご使用のアプリケーション環境に合わせてシステムをカスタマイズする方法については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』と『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。

この章の内容は、次のとおりです。

- 1 ページの「構成に関するタスク」
 - 2 ページの「Sun MTP 標準規格」
-

構成に関するタスク

Sun MTP を実行する前に、領域環境を定義する必要があります。定義したあと、環境に固有の一連の構成タスクを実行できるようになります。

領域環境の定義

- アプリケーションのディレクトリ構造を定義および作成します。
- プログラムとデータファイルを適切なディレクトリに移動します。
- 領域のすべての環境変数を設定する設定ファイルを作成します。

アプリケーションのリソースを定義します。

- Sun MTP のリソースを定義します。
- Sun MTP テーブルの制御情報を設定します。
- アプリケーションが VSAM ファイルにアクセスする場合、次の作業も必要です。
 - VSAM カタログの作成
 - VSAM ファイルの構築

UNIX 端末の構成

- 特定のタイプの端末とキーボードから、アプリケーションを実行できることを確認します。

プリンタの設定

- 領域と通信できるように、プリンタと一時データキューを設定します。

通信の設定

- プログラム間の通信が可能になるように、システム間通信 (ISC) を設定します。Sun MTP は SNA 上の ISC と TCP/IP をサポートします。
- 領域を設定し、遠隔クライアントとの通信を確立します。

オプション製品の構成

- リレーショナルデータベースなどのその他の製品を認識して連携できるように、領域を設定します。

注 – Sun 以外の XA リソースマネージャーを使用する場合は、『Sun Mainframe Transaction Processing ソフトウェア XA リソースマネージャーの使用』を参照してください。

Sun MTP 標準規格

この節では、Sun MTP の標準規格に関する次の事項について説明します。

- 3 ページの「画面の形式」
- 4 ページの「ファンクションキー」
- 4 ページの「データ入力の検査」
- 5 ページの「ファイル識別子」

ご使用のユーザーアプリケーションによって、標準規格が異なる場合があります。

画面の形式

次の表で説明するように、データ入力画面は、すべて4つの領域に分かれています。

表 1-1 画面の形式

画面の領域	説明
ヘッダー領域 (1 行目)	現在の画面名および現在の日付/時刻が示されます。
詳細領域(2 ~ 20 行目)	メニュー、データエントリ画面、およびファイル選択領域が含まれます。
応答領域 (21 行目)	通常は、行全体に下線 (_) が引かれています。データをディスクに書き込む処理などの実行でエラーが発生すると、エラー条件を示すメッセージが強調表示されます。また、操作の状態も表示されます。
ファンクションキー説明領域 (22 ~ 24 行目)	画面で使用可能なファンクションキーおよびそれを押したときに実行できる操作が表示されます。

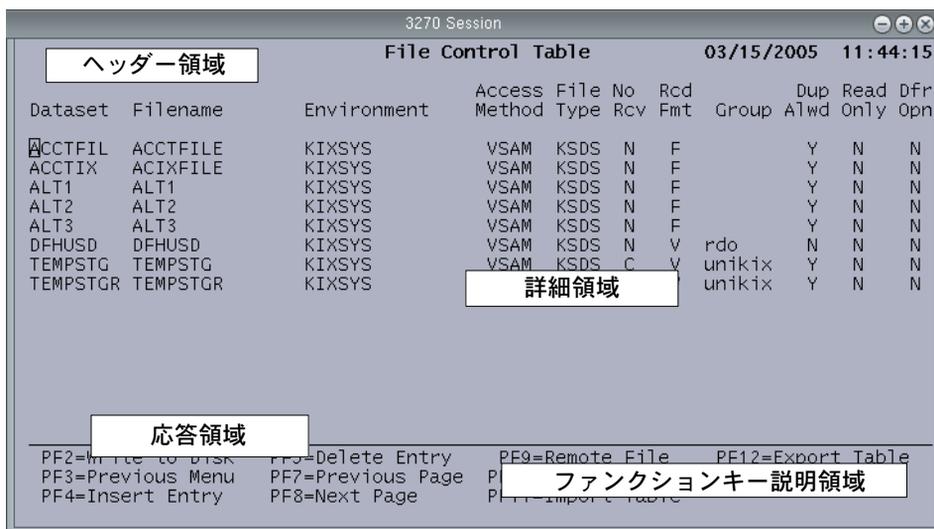


図 1-1 Sun MTP 画面形式の例

ファンクションキー

次の表で説明するファンクションキーは、Sun MTP 画面で定められた操作を実行します。

表 1-2 ファンクションキー

ファンクション キー	アクション
PF3	前の画面に戻ります。画面で変更したデータをディスクに保存する前にこのキーを押すと、次の警告が表示されます。 Table has been modified. Press PF3 if modification is only temporary. 変更を保存しない場合は PF3 キー、そうでない場合はほかのキーを押し、操作を実行します。
Enter	現在表示されているエントリのデータを変更します。保護されていない値を上書きしてこのキーを押すと、画面の内容が変更されます。すべてのフィールドが関連の検証テストに通った場合だけデータが変更されます。
Clear	新しいトランザクションを入力できるように画面を消去します。
Reset	数値フィールドへのテキストの入力など、操作エラーのあとでシステムをリセットします。

キーボードの割り当て情報については、付録 A を参照してください。

データ入力の検査

データフィールドには大文字および小文字の両方を入力できます。ほとんどの場合、小文字は大文字に変換されます。一部のデータフィールドでは、小文字が有効なデータとして受け入れられます。通常、CICS コマンド関連のフィールドでは大文字に変換され、その他のフィールドでは入力されたままの状態になります。たとえば、データセットは大文字に変換され、ファイル名は入力されたままの状態になります。

データ入力の検査により、無効なデータが含まれるフィールドが強調表示されます。エラーデータが入力されている最初のフィールドにカーソルが移動し、応答領域に次のようなメッセージが表示されます。

```
Data in field invalid/required
```

ファイル識別子

ファイル識別子は、次の 2 つの要素で構成されます。

- ディレクトリ、または 1 つ以上のディレクトリを指定できる環境変数
- ファイル識別子の最後の部分を構成するファイル名

ファイル識別子を入力する場合、次の表に記載されている指定事項に従ってください。

表 1-3 ファイル識別子

ファイル識別子	説明
ディレクトリ	Sun MTP で使用される絶対ディレクトリ名は、50 文字以内でなければなりません。パス名の任意の部分に代えて、先頭にドル符号 (\$) を付けた環境変数を使用できます。たとえば、次の 2 行はどちらも有効であり、同一のディレクトリ名を示します。 <ul style="list-style-type: none">• mtp/mtp8/finance/sys• \$KIXSYS \$ 符号は、KIXSYS 環境変数を完全な値に展開します。
環境変数	ディレクトリやファイルの名前、または値 (1 ~ 14 文字)。環境変数はすべて大文字です。
ファイル名	拡張子も含む Sun MTP ファイル名 (1 ~ 14 文字)。

第2章

領域環境変数の設定

この章では、アプリケーション領域の定義に必要なタスクについて説明します。「領域」とは、各アプリケーションを定義するプロセス、テーブル、ファイル、および環境変数のセットです。

次のトピックがあります。

- 7 ページの「領域構成に関するタスク」
- 8 ページの「ディレクトリ構造の作成」
- 9 ページの「環境変数」
- 19 ページの「領域環境の定義」

領域構成に関するタスク

Sun MTP アプリケーション環境を構成するには、まず領域を定義します。ただし、最初にディレクトリ構造を設計し、アプリケーションのデータとプログラムファイルを適切なディレクトリに移動させる必要があります。図 2-1 に、ディレクトリ構造の例を示します。

領域の定義には、いくつかの作業が必要です。

- 領域に関する環境変数の設定
- アプリケーションリソースの定義
 - 必要なテーブルにリソースを定義する方法については、第 5 章を参照してください。リソースを動的に定義する方法については、第 7 章を参照してください。
 - すべてのリソーステーブルについては、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。
- アプリケーションプログラムが VSAM ファイルを使用する場合の、VSAM カタログと VSAM ファイルの構築
 - VSAM ファイルの管理については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

- アプリケーションが RDBMS を使用する場合は、第 13 章を参照してください。

環境変数を設定してから、第 3 章に目を通してください。ここでは、領域を開始および停止する方法、ローカルクライアント接続を確立する方法について説明されています。

ディレクトリ構造の作成

オペレーティングシステムが提供する機能を使用して、次の図にあるようなディレクトリ構造を作成します。この例では、特定のディレクトリに対応する環境変数の一部を示しています。設定する環境変数は、構造内のディレクトリを指す必要があります。

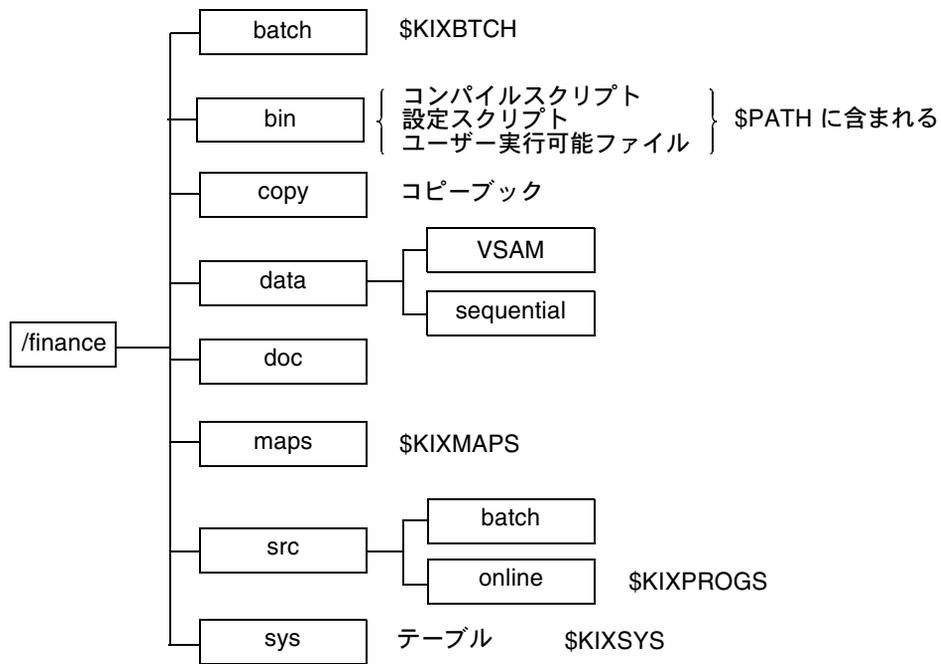


図 2-1 財務アプリケーションのディレクトリ構造の例

環境変数

この節では、領域を設定するために必要なほとんどの環境変数について説明します。アプリケーションによっては、その他の環境変数も必要です。

- RDBMS を使用する場合は、第 13 章を参照してください。
- Sun Mainframe Security Facility などの外部のセキュリティ管理ソフトウェアを使用する場合は、17 ページの「Sun MTP Secure 環境変数」と 19 ページの「Sun MSF 環境変数」を参照してください。

標準環境変数

次の表は、ファイル、アプリケーション、ライブラリの検索と、プロパティの構成を行うために領域が使用する環境変数とその説明をまとめたものです。これらの環境変数のほとんどは、領域の設定ファイルで設定します。このファイル以外で設定する変数についても、この表で説明します。

表 2-1 標準の Sun MTP 環境変数

環境変数	設定方法	任意/ 必須	説明
ACUCOBOL	設定ファイル	任意	ACUCOBOL-GT を使用する場合、ACUCOBOL-GT ソフトウェアがインストールされているディレクトリのフルパス名に設定します。設定したパスに \$ACUCOBOL/bin が含まれている必要があります。
ADMIN_JVM_OPTIONS	設定ファイル	任意	unikixadmin デーモンによって使用される Java™ 仮想マシン (JVM™) のプロパティを指定します。Sun Mainframe Administration Tool (Sun MAT) を使用している場合に、この変数の使用が必要になることがあります。詳細については、『Sun Mainframe Transaction Processing ソフトウェア 障害追跡とチューニング』を参照してください。
COBCPY	設定ファイル	任意	Server Express を使用する場合、使用する COBOL コピーライブラリと \$UNIKIX/copy ライブラリを含むディレクトリに設定します。これは、コロンで区切ったディレクトリのリストです。
COBDIR	設定ファイル	任意	Server Express を使用する場合、Server Express ソフトウェアがインストールされているディレクトリのフルパス名に設定します。設定したパスに \$COBDIR/bin が含まれている必要があります。

表 2-1 標準の Sun MTP 環境変数 (続き)

環境変数	設定方法	任意/ 必須	説明
COBMODE	設定ファイル	任意	Server Express を使用する場合、32 に設定して、実行環境が 32 ビットになるようにします。
COBPATH	設定ファイル	任意	Server Express を使用する場合、CALL 文で呼び出される Server Express プログラムが含まれるディレクトリに設定します。
CODE_PREFIX	設定ファイル	任意	ACUCOBOL-GT を使用する場合、CALL 文で呼び出される ACUCOBOL-GT プログラムが含まれるディレクトリに設定します。RDBMS を使用する場合は、kixinstall セッション中に指定された Sun MTP 実行可能ファイルのターゲットディレクトリを組み込んで実行可能ファイルを構築する必要もあります。
COPYPATH	設定ファイル	任意	ACUCOBOL-GT を使用する場合、使用する COBOL コピーライブラリと \$UNIKIX/copy ライブラリを含むディレクトリに設定します。これは、コロンで区切ったディレクトリのリストです。
DCLRTERM	設定ファイル	任意	<p>ECI/EPI クライアントと遠隔 Sun MTP または CICS 領域から、SNA 接続で同時に実行できる、インバウンド要求の最大数。0 より大きい値でなければいけません。利用できるセッションよりも要求数が多い場合、多すぎる要求はセッションが利用できるようになるまでキューに入れられるので、パフォーマンスに影響する可能性があります。</p> <p>unikixdc1 プロセスを構成している場合は、必須です。詳細は、108 ページの「SNA の領域の構成」を参照してください。</p> <p>インバウンドセッションとアウトバウンドセッションの合計数 (DCLRTERM + DCLSTERM) は、200 を超えてはいけません。</p>
DCLSTERM	設定ファイル	任意	<p>遠隔 Sun MTP または CICS 領域への SNA 接続の、並行アウトバウンド要求の最大数。0 より大きい値でなければいけません。多すぎる要求はキューに入れられるのでパフォーマンスに影響することがありますが、30 秒たってもセッションが利用できない場合はタイムアウトします。</p> <p>unikixdc1 プロセスを構成している場合は、必須です。詳細は、108 ページの「SNA の領域の構成」を参照してください。</p> <p>最小値の 2 に設定します。</p> <p>インバウンドセッションとアウトバウンドセッションの合計数 (DCLRTERM + DCLSTERM) は、200 を超えてはいけません。</p>

表 2-1 標準の Sun MTP 環境変数 (続き)

環境変数	設定方法	任意/ 必須	説明
KIX_ADJ_DST	設定ファイル	任意	夏時間の変更時に、スパンをまたいで非同時に起動されたトランザクションが、そのまま必ず夏時間を維持するように値を設定します。 この環境変数を EXEC CICS START INTERVAL API で使用するの詳細は、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。
KIX62PASSTHRU	設定ファイル	任意	設定すると、Sun MTP は、ECI クライアント要求から取得したユーザー名とパスワードを、ローカルの検証なしに遠隔領域に渡します。
KIXBIN	設定ファイル	任意	unikixtran の実行可能バージョンの代替ディレクトリ。\$KIXBIN が設定されている場合、領域を起動するとき、強制的な中止または CINI トランザクションのあとで、トランザクション処理プログラムが再起動されるときは、このディレクトリを最初に検索してトランザクション処理プログラムをロードします。 \$KIXBIN がないか NULL の場合、またはトランザクション処理プログラムが \$KIXBIN ディレクトリにない場合、現在あるすべての検索アルゴリズムが適用されます。検索規則は、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』の unikixmain -1 に記述されています。
KIXBTCH	設定ファイル	任意	バッチシェルスクリプトを含むディレクトリをコロンで区切ったリスト。ディレクトリ名には、開始時間と終了時間を角かっこで囲んで続けることができます (たとえば、[HHMM-HHMM])。これらの時間は、実行するシェルスクリプトをそのディレクトリから検索する時間を示しています。 \$KIXBTCH の詳細は、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。
KIXCONVTRACE	設定ファイル	任意	変換トレースを有効にします。トレースを有効にするには、この環境変数を YES に設定します。出力は、\$KIXSYS/unikixmain.dbg ファイルに書き込まれます。 この環境変数は、ご購入先による指示がある場合にのみ使用します。
KIX_ENABLE_JAVA	設定ファイル	任意	領域での Java プログラムの使用を有効にするには、この変数を YES に設定します。Java 仮想マシン (JVM) がトランザクション処理プログラムにロードされるようになります。

表 2-1 標準の Sun MTP 環境変数 (続き)

環境変数	設定方法	任意/ 必須	説明
KIXDUMPDIR	設定ファイル	任意	すべての内部トレースダンプファイルが書き込まれるディレクトリに設定します。設定しない場合、これらのファイルは \$KIXSYS/debugkix ディレクトリに書き込まれます。
KIXDYNRTE	設定ファイル	任意	設定した場合、Sun MTP は、アウトバウンド機能シップ、分散プログラムリンク (DPL)、非同期処理、またはトランザクション経路指定の要求時、セッションを割り当てる前に動的な経路指定の出口を呼び出します。
KIXGETMAINDMP	設定ファイル	任意	EXEC CICS GETMAIN 呼び出しによって取得した共有メモリーのダンプを有効にします。アプリケーションのダンプについては、『Sun Mainframe Transaction Processing ソフトウェア 障害追跡とチューニング』を参照してください。
KIXHOST	設定ファイル	任意	3270 SNA クライアントに接続する場合は、必須です。Sun MTP サーバーを実行する UNIX ホストの名前に設定します。Sun MTP を起動する前に設定する必要があります。
KIX_JVM_OPTIONS	設定ファイル	任意	この変数は、JVM の動作を変更しますが、通常はデバッグ用のみ設定します。KIX_JVM_OPTIONS を設定すると、通常スタンドアロンの JVM にコマンド行のパラメータとして与えられる任意の起動フラグを、java 実行可能ファイルに含めることができます。
KIXLIB	設定ファイル	任意	コロンで区切ったディレクトリのリスト。Sun MTP は C、C++、PL/I の共有オブジェクトまたは共有ライブラリをロードするとき、このリストを左から右に検索します。 \$KIXLIB が設定されていない場合、ローダーは \$KIXSYS を探します。C、C++、または PL/I の使用方法については、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。
KIXLICDIR	設定ファイル	必須	ライセンスファイルが置かれているディレクトリに設定します。インストール方法については、『Sun Mainframe Transaction Processing ソフトウェア インストールガイド』を参照してください。
KIXMAPS	設定ファイル	必須	コロンで区切ったディレクトリのリスト。Sun MTP がマップをロードするときに、左から右に検索します。マップは、\$KIXMAPS から、 \$KIXSYS/_kix_reserved_maps または _KIX_RESERVED_MAPS 環境変数で指定されるディレクトリにロードされます。

表 2-1 標準の Sun MTP 環境変数 (続き)

環境変数	設定方法	任意/ 必須	説明
KIXMAXIST	設定ファイル	任意	Sun MTP 領域に自動インストールできる、ECI/EPI クライアントと遠隔 Sun MTP または CICS 領域の最大数。その領域の端末テーブルに使用可能なエントリがない場合、遠隔領域からの要求は拒否されます。133 ページの「ECI/EPI クライアントの領域の構成」も参照してください。
KIX_PGM_MODE	設定ファイル	任意	PPT でタイプが K であると定義されている C プログラムにのみ該当します。TXSERIES に設定して、C プログラム (共有オブジェクト) が新しいタスクまたは同じタスクによって実行されるたびに、このプログラムの新しいコピーがロードされることを指定します。
KIX_PGMTXN_MODE	設定ファイル	任意	PPT でタイプが K であると定義されている C プログラムにのみ該当します。TXSERIES に設定して、各タスクのあとで C プログラムが解放されることを指定します。
KIXPROGS	設定ファイル	必須	コロンで区切ったディレクトリのリスト。Sun MTP が COBOL プログラムをロードするときに左から右に検索します。
KIXRDBMS_BYPASS	設定ファイル	任意	任意の値 (たとえば、KIXRDBMS_BYPASS=1) に設定すると、構成されている RDBMS に接続せずに Sun MTP を起動します。詳細は、153 ページの「領域と RDBMS の起動」を参照してください。
KIX_READLOCKON	バッチジョブ ステップ	任意	VSAM ファイルのリストにバッチ読み取りロックを指定します。この指定はバッチジョブのステップを実行している間または別の環境変数の設定が見つかるまでの間、VSAM カタログに定義されている各ファイルの「バッチ読み取りロック」の定義に優先します。 詳細は、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。
KIX_READLOCKOFF	バッチジョブ ステップ	任意	バッチ処理による更新時に、ロックしない VSAM ファイルのリストを指定します。この指定はバッチジョブのステップを実行している間または別の環境変数の設定が見つかるまでの間、VSAM カタログに定義されている各ファイルの「バッチ読み取りロック」の定義に優先します。 詳細は、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。
KIXREMANIMPORT	設定ファイル またはシェル プロンプト	任意	Solaris プラットフォーム Sun MTP が Micro Focus Remote Animation に使用するポート番号に設定します。 詳細は、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。

表 2-1 標準の Sun MTP 環境変数 (続き)

環境変数	設定方法	任意/ 必須	説明
KIXREMANIMTOUT	設定ファイル またはシェル プロンプト	任意	<p>Solaris プラットフォーム</p> <p>エラーを返す前に、Sun MTP が Cross Animation サーバーを待機するタイムアウト時間を秒単位で設定します。デフォルト値は 5 秒です。Micro Focus Remote Animation 専用です。</p> <p>詳細は、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。</p>
_KIX_RESERVED_MAPS	設定ファイル	任意	<p>Sun MTP がマップセットをロードするロード元のディレクトリに設定します。ディスクからマップセットをロードするときに、Sun MTP はまず \$_KIX_RESERVED_MAPS を検索し、そこにあればそのバージョンを使用します。対象のマップセットがない場合、Sun MTP は \$KIXMAPS を検索し、そのバージョンを \$_KIX_RESERVED_MAPS にコピーして使用します。この変数を設定しない場合、Sun MTP はデフォルトのディレクトリ \$KIXSYS/_kix_reserved_maps を作成します。</p> <p>この変数を \$KIXMAPS に設定しないでください。</p> <p>詳細については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。</p>
KIXSNAPDIR	設定ファイル	任意	<p>kixsnap ユーティリティーがその出力ファイルを書き込むディレクトリに設定します。この変数を設定しない場合、Sun MTP はファイルをディレクトリ \$KIXSYS/debugkix に書き込みます。</p> <p>kixsnap についての詳細は、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。</p>
KIXSNT_NOMEMUPDATE○	設定ファイル	任意	<p>動的なパスワード変更機能を無効にするには、領域を起動する前に、この環境変数 KIXSNT_NOMEMUPDATE を任意の値 (たとえば、KIXSNT_NOMEMUPDATE=1) に設定する必要があります。動的なパスワード変更機能を無効にした場合は、SNT に変更を反映させるために、領域を再起動する必要があります。詳細については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。</p>
KIXSOCKET_TIMEOUT	設定ファイル	任意	<p>unikixsock プロセスが、クライアントを切断するまでソケットデータを待機する時間を指定します。秒単位の値に設定してください。</p>

表 2-1 標準の Sun MTP 環境変数 (続き)

環境変数	設定方法	任意/ 必須	説明
KIXSYS	設定ファイル	必須	システムテーブルを含むディレクトリを設定します。この環境変数は、領域を一意に定義します。多くの領域を同じホスト上で同時に実行できますが、各領域には \$KIXSYS として一意の値が必要です。Sun MTP テーブルを更新する必要があるユーザーは、このディレクトリへの書き込み権限が必要です。
KIXVSAMPOOLS	設定ファイル	任意	VSAM データベースに使用するバッファープールの数に設定します。バッファープールの最小数 (デフォルト) は 16 です。したがって、この環境変数は 16 より大きな値に設定します。 この変数の設定が使用されるのは、領域がオペレーティングシステムの相互排除を使用しているときだけです。領域がセマフォールを使用している場合にこの変数を設定すると、エラーメッセージが表示され、この領域の起動は失敗します。詳細は、46 ページの「VSAM バッファープールおよび VSAM バッファの数の設定」を参照してください。
LD_LIBRARY_PATH	設定ファイル	必須	Solaris プラットフォーム 共有ライブラリのディレクトリ。\$UNIX/lib を含む必要があります。 Server Express の場合は、\$COBDIR/lib を含めます。 ACUCOBOL-GT の場合は、\$ACUCOBOL/lib を含めます。 Sun Mainframe Administration Tool (Sun MAT) を使用する場合は、JVM の場所を含む必要があります。
LIBPATH	設定ファイル	必須	AIX プラットフォーム 共有ライブラリのディレクトリ。\$UNIX/lib を含む必要があります。 Server Express の場合は、\$COBDIR/lib を含めます。 Sun Mainframe Administration Tool (Sun MAT) を使用する場合は、JVM の場所を含む必要があります。
MTPADMINTRACE	設定ファイル または シェルプロンプト	任意	Sun MAT を使用する場合、unikixadmin サーバーのデバッグトレースを有効にするには、数値を少なくとも 1 に設定する必要があります。

表 2-1 標準の Sun MTP 環境変数 (続き)

環境変数	設定方法	任意/ 必須	説明
PATH	設定ファイル	必須	<p>シェルに実行可能ファイルを検索する場所を指示します。パス名をコロンで区切ったリストで、左から右に検索されます。シェルで自動的に設定されます。ディレクトリ \$UNIX/local/bin と \$UNIX/bin をこの順番で追加する必要があります。</p> <p>Server Express の場合は、bin ディレクトリ (\$COBDIR/bin) を PATH 変数に追加します。</p> <p>ACUCOBOL-GT の場合は、bin ディレクトリ (\$ACUCOBOL/bin) を PATH 変数に追加します。</p> <p>RDBMS を使用する場合、RDBMS の実行可能ファイルのディレクトリを PATH 変数に追加します。たとえば、\$ORACLE_HOME/bin を追加します。</p>
TCPRTERM	設定ファイル	任意	<p>ECI/EPI クライアントと遠隔 Sun MTP または CICS 領域から、TCP/IP 接続できる並行インバウンド要求の最大数。利用できるセッションよりも要求数が多い場合、Sun MTP は超過した要求をキューに入れます。これはパフォーマンスに影響する可能性があります。</p> <p>TCP サーバープロセスを構成する際に必要とされます。詳細は、113 ページの「TCP/IP の領域の構成」を参照してください。</p> <p>インバウンドセッションとアウトバウンドセッションの合計数 (TCPRTERM + TCPSTERM) は、200 を超えてはいけません。</p> <p>設定されている各 \$TCPRTERM には、32K バイトの共有メモリーが必要です。</p>
TCPSTERM	設定ファイル	任意	<p>遠隔 Sun MTP または CICS 領域に、TCP/IP 接続できる並行インバウンド要求の最大数。利用できるセッションよりも要求数が多い場合、Sun MTP は超過した要求をキューに入れます。これはパフォーマンスに影響する可能性があります。</p> <p>TCP サーバープロセスを構成する際に必要とされます。詳細は、113 ページの「TCP/IP の領域の構成」を参照してください。</p> <p>最小値の 2 に設定します。</p> <p>インバウンドセッションとアウトバウンドセッションの合計数 (TCPRTERM + TCPSTERM) は、200 を超えてはいけません。</p> <p>設定されている各 \$TCPSTERM には、32K バイトの共有メモリーが必要です。</p>

表 2-1 標準の Sun MTP 環境変数 (続き)

環境変数	設定方法	任意/ 必須	説明
TMPDIR	設定ファイル	任意	Sun MTP が一時ファイル用に使用するディレクトリに設定します。設定しないと、Sun MTP によってデフォルトで /tmp に設定されます。その他のアプリケーションも一時ファイルに /tmp を使用する可能性があるため、このファイルシステムが一杯になることがあります。
UNIXIX	設定ファイル	必須	Sun MTP インストールディレクトリのフルパス名に設定します。

Sun MTP Secure 環境変数

Sun MTP テーブルで定義されたトランザクションセキュリティーキーを使用するか、Sun MTP Secure を使用することによって、トランザクションセキュリティーを実装することができます。Sun MTP Secure は、外部セキュリティーマネージャー (ESM) へのインタフェースになり、トランザクションとリソースセキュリティーのチェック機能を提供します。Sun Mainframe Security Facility (Sun MSF) は、役割ベースの包括的なアクセス制御システムを提供するために Sun MTP に統合されています。基本的な Sun MTP セキュリティーと Sun MTP Secure については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。Sun MSF については、『Sun Mainframe Security Facility 管理者ガイド』を参照してください。Sun MTP ソフトウェアには、Sun MSF を使用しない場合にサイト固有の ESM を開発するために使用可能なセキュリティーユーザー出口コードも用意されています。

表 2-2 は、Sun MTP Secure と ESM の使用を有効化および無効化したり、各リソースクラスのセキュリティーチェックのレベルを定義したりするオプションの環境変数とその説明をまとめたものです。これらの変数は、領域の設定ファイルで設定します。

デフォルトでは、Sun MTP Secure は無効とされています (KIXSEC=NO)。また、管理者が定義したトランザクションセキュリティーキーのメカニズムが使用され、リソースセキュリティーメカニズムは使用できません。Sun MTP Secure が有効であり (KIXSEC=YES)、ESM が構成されている場合は、外部のセキュリティーチェックは、すべてのトランザクションとリソースについて実行され、管理者が定義したトランザクションセキュリティーのキーは無視されます。対応する KIXxxxSEC 環境変数を NO に指定すれば (たとえば、KIXFCTSEC=NO)、リソースタイプごとに外部のセキュリティーチェックを無効にできます。

表 2-2 Sun MTP Secure 環境変数

環境変数	説明
KIXSEC	Sun MTP Secure を有効または無効にします。NO (デフォルト) に設定すると、その他の“SEC”変数はすべて無視されます。定義済みのトランザクションセキュリティレベル (TSL) キーを代わりに使用します。
KIXSECDFLTUSER	サインオンテーブル (SNT) に定義されているユーザー名に設定します。 UNIX ログインなしで、また Sun MTP に CESN や CSSN でサインオンせずに確立された接続に対して使用されます。あるいは CESF や CSSF でサインオフしたあとに使用されます。 ユーザー名とパスワードは、外部セキュリティマネージャーで認証します。ユーザー名が認証されない場合、Sun MTP は終了します。デフォルトでは、この環境変数は kixuser に設定されていますが、kixuser は SNT にあらかじめ定義されていません。このため、Sun MTP Secure を有効にする前に、SNT と外部セキュリティ管理データベースにデフォルトのユーザー名とパスワードを定義する必要があります。
KIXSEC_LOGGING	領域の unikixmain.log ファイルに記録する、ESM から得たメッセージのレベルを制御します。 <ul style="list-style-type: none"> • ALL では、すべての ESM 応答メッセージ (成功および失敗) が記録されます。 • DENIALS では、アクセスが拒否された ESM 応答メッセージのみが記録されます。これがデフォルトです。 • NONE では、ESM 応答メッセージは記録されません。
KIXSECPREFIX	各 Sun MTP アプリケーションに固有のリソースのアクセス権を、外部のセキュリティ管理者が定義できるようにします。YES に設定すると、領域を起動した UNIX ユーザー ID がすべてのリソース名の接頭辞として使用されます。デフォルトは NO です。
KIXAPPSEC	YES (デフォルト) および KIXSEC=YES に設定すると、\$KIXSYS 値として定義されている、領域の UNIX アプリケーション名に対してセキュリティチェックが実行されます。
KIXCMDSEC	YES (デフォルト) および KIXSEC=YES に設定すると、CEMT と EXEC CICS SET/INQUIRE コマンドに対してセキュリティチェックが実行されます。
KIXDCTSEC	YES (デフォルト) および KIXSEC=YES に設定すると、一時データキューに対してセキュリティチェックが実行されます。
KIXFCTSEC	YES (デフォルト) および KIXSEC=YES に設定すると、VSAM ファイルに対してセキュリティチェックが実行されます。
KIXJCTSEC	YES (デフォルト) および KIXSEC=YES に設定すると、ジャーナルに対してセキュリティチェックが実行されます。
KIXPCTSEC	YES (デフォルト) および KIXSEC=YES に設定すると、付加トランザクションに対してセキュリティチェックが実行されます。
KIXPPTSEC	YES (デフォルト) および KIXSEC=YES に設定すると、アプリケーションプログラムに対してセキュリティチェックが実行されます。

表 2-2 Sun MTP Secure 環境変数 (続き)

環境変数	説明
KIXSTTSEC	YES (デフォルト) および KIXSEC=YES に設定すると、非同期で起動されたトランザクションに対してセキュリティチェックが実行されます。
KIXTCTSEC	YES (デフォルト) および KIXSEC=YES に設定すると、端末に対してセキュリティチェックが実行されます。
KIXTSTSEC	YES (デフォルト) および KIXSEC=YES に設定すると、一時記憶域キューに対してセキュリティチェックが実行されます。

Sun MSF 環境変数

次の表では、Sun MSF ソフトウェアに関連する環境変数について説明します。これらの変数は、領域の設定ファイルで設定します。これらの変数の使用方法については、『Sun Mainframe Security Facility 管理者ガイド』を参照してください。

表 2-3 Sun MSF 環境変数

環境変数	説明
KIXSEC_SERVERHOST	サーバーのホスト名または IP アドレスに設定します。Sun MSF がセキュリティサーバーへのソケット接続を使用するため、必要です。
KIXSEC_SERVERPORT	サーバーのソケットポート番号に設定します。Sun MSF はセキュリティサーバーにソケット接続を使用するため、必要です。

領域環境の定義

領域の環境は、必要な環境変数をファイルに設定することで定義します。このファイルは、ユーザー固有のログインファイルとすることも、多くのユーザーが実行できる設定ファイルとすることもできます。

- ユーザーが複数の領域でアプリケーションを実行する必要がある場合は、領域ごとに異なる設定ファイルを作成します。ユーザーは、領域に対応する設定ファイルを実行できます。
- ユーザーが 1 つの領域のみでアプリケーションを実行する場合、それぞれのユーザーのログインファイルに環境変数を設定できます。ユーザーはその領域にのみ接続できます。

設定ファイルの定義と実行

次の例は、領域に環境変数を設定する Bourne または Korn シェル設定ファイルを示します。ファイルには、次の注意点があります。

- 複数のディレクトリを環境変数に割り当てるときは、パスをコロンで区切る必要があります。
- スクリプトの最初に定義されている SYSDIR 環境変数は、ユーザー定義の環境変数の例です。これによりアプリケーションの上位レベルのディレクトリを定義します。
- 示されている COBOL の環境変数は、Server Express 用です。

```
TMPDIR=/tmp/mtp;export TMPDIR
UNIKIX=/pkgs/mtp/MTP8.1.0;export UNIKIX
SYSDIR=/prod/appl/acctg;export SYSDIR
KIXSYS=$SYSDIR/sys;export KIXSYS
KIXPROGS=$SYSDIR/progs;export KIXPROGS
KIXMAPS=$SYSDIR/maps;export KIXMAPS
VSAMDATA=$SYSDIR/data/vsam;export KIXDATA
SEQFILES=$SYSDIR/data/sequential;export SEQFILES
KIXBTCH=$SYSDIR/batch;export KIXBTCH
KIXLICDIR=/pkgs/mtp/MTP8.1.0/lib;export KIXLICDIR
COBDIR=/opt/cobol/4.0;export COBDIR
COBCPY=$SYSDIR/copy:$UNIKIX/copy;export COBCPY
COBMODE=32;export COBMODE
LD_LIBRARY_PATH=$UNIKIX/lib:$COBDIR/lib;export LD_LIBRARY_PATH
PATH=$UNIKIX/local/bin:$UNIKIX/bin:$COBDIR/bin:$SYSDIR/bin:$PATH;export PATH
```

領域の環境変数を設定する `kixsetup` という名前の Bourne シェルまたは Korn シェルの設定ファイルを実行するには、“.” (ドット) コマンドを使用します。 . と `kixsetup` との間には空白が必要です。

```
$ . kixsetup
```

次の例は、C シェルユーザー用の設定ファイルを示しています。これらの変数の値は、一つの例です。

```
setenv UNIKIX /pkgs/mtp/MTP8.1.0
setenv KIXSYS /prod/appl/acctg/sys
setenv KIXPROGS /prod/appl/acctg/progs
setenv KIXMAPS /prod/appl/acctg/maps
setenv KIXBTCH /prod/appl/acctg/batch/runq
setenv VSAMDATA /prod/appl/acctg/data/vsam
setenv SEQFILES /prod/appl/acctg/data/sequential
setenv COBDIR /opt/lib/cobol/4.0
setenv COBCPY /prod/appl/acctg/copy:${UNIKIX}/copy
setenv LD_LIBRARY_PATH ${UNIKIX}/lib:${COBDIR}/lib
setenv KIXHOST `uname -n`
setenv PATH ${UNIKIX}/local/bin:${UNIKIX}/bin:${COBDIR}/bin:$PATH
setenv TMPDIR /tmp/mtp
tput init
```

`kixsetupc` という名前の C シェルの設定ファイルを実行するには、`source` コマンドを次のように使用します。

```
machine1% source kixsetupc
```


第3章

Sun MTP の起動と停止

この章では、Sun MTP の起動と停止方法およびローカルクライアントの起動方法について説明します。次のトピックについて説明します。

- 23 ページの「Sun MTP サーバーの起動」
- 24 ページの「ローカル端末クライアントの起動」
- 25 ページの「トランザクションの実行」
- 25 ページの「端末セッションの停止」
- 26 ページの「Sun MTP サーバーの停止」
- 28 ページの「クライアント接続」

アプリケーションを実行するには、まずサーバーを起動し、次にクライアントを起動します。Sun MTP サーバーは、端末のクライアントからトランザクションが入力されるまで待機します。

Sun MTP サーバーの起動

Sun MTP メインサーバープロセスを開始する `unikixmain` コマンドには、多数のオプションがあります。unikixmain オプションについては、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

▼ サーバーを起動する

1. 領域の設定ファイルを実行します。
詳細は、19 ページの「領域環境の定義」を参照してください。

2. コマンドプロンプトで、unikixmain コマンドとそのオプションを入力します。

または、unikixmain のオプションを使用して kixstart シェルスクリプトを使用します。kixstart スクリプトは、このオプションを unikixmain コマンドに渡します。

注 - ローカル端末クライアントを使用している場合は、-Q オプションを使用します。

unikixmain が起動されると、コマンドを起動した端末とは切り離されるので、その端末は別の処理に使用できます。

ローカル端末クライアントの起動

さまざまなクライアントからトランザクションを実行できます。この手順は、ローカル端末クライアントを使用して有効な領域に接続する方法です。

▼ ローカル接続を確立する

1. ユーザーとしてホストにログインします。
2. 適切な設定ファイルを読み込み、環境を設定します。
3. Sun MTP 端末クライアントを起動します。

たとえば、X 端末 (Xterm) からクライアントを起動するには、次のように入力します。

```
$ xtermunikix unikix
```

4. 新しいウィンドウに、Sun MTP の著作権画面が表示されたら、いずれかのキーを押して画面をクリアします。

著作権画面のソースは、\$UNIX/lib/logofile です。

起動シェルスクリプトについては、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。また、xtermunikix などの xxxunikix シェルスクリプトでクライアントを起動する方法については、第 8 章を参照してください。このスクリプトは、使用する端末を正しいキーボードタイプに設定します。

クライアントが起動に失敗した場合、テキストエディタを使用して \$KIXSYS ディレクトリにある unikixmain.log ファイルを表示します。エラーメッセージから失敗した原因を探ってください。また、『Sun Mainframe Transaction Processing ソフトウェア 障害追跡とチューニング』も参照してください。

トランザクションの実行

領域に正しく接続し、著作権画面を消去すると、トランザクションをサブミットできるようになります。

▼ トランザクションを実行する

1. 任意のキーを押して、著作権画面を消去します。
2. 空白の画面が表示されるので、上部左隅にトランザクション識別子を入力します。
3. Enter キー (Return キーではない) を押し、トランザクションを Sun MTP に送ります。

たとえば、Table Manager を表示するには、CTBL を入力して Enter キーを押します。

端末セッションの停止

端末でセッションを停止するために使用できるトランザクションは、複数あります。これらのトランザクションは Sun MTP サーバーを停止しません。これらのトランザクションを実行するには、トランザクション識別子とそのパラメータを入力できるだけの十分な空きスペースが画面に必要です。画面にスペースがない場合は、Clear キーを押して画面を消去し、トランザクションを入力します。

次のいずれかを入力します。

```
CSSF GOODNIGHT
```

```
CSSF LOGOFF
```

```
CSMT SHUT,NO
```

これらのトランザクションについては、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

Sun MTP サーバーの停止

Sun MTP サーバーを正しく停止する方法は、4 つあります。これらの方法へのアクセスを制限すれば、ユーザーが偶発的にサーバーを停止してしまうことが避けられます。

方法 1

CEMT PERFORM SHUTDOWN トランザクションを実行してシステムを SHUT 状態にします。

有効なトランザクションは、正常に完了することができます。新しいトランザクションはサブミットできません。新しいユーザーはログインできず、端末も接続できません。

方法 2

開発システムメインメニューで、PF3 キーを押し、停止プロセスを開始します。要求を確認するプロンプトが表示される場合は、PF3 キーをもう一度押します。

Sun MTP の組み込みセキュリティーを使用する場合、CMNU トランザクションへのアクセスを持つユーザーに、この方法で領域を停止させないようにするセキュリティーチェックは実行されないの、注意してください。この方法の結果は、方法 1 と同じです。

方法 3

CSMT SHUT, YES トランザクションを実行します。

このトランザクションは、ほかのユーザーがまだシステムに接続されている場合でも、ユーザーセッションと一っしょに領域を停止します。動作中のトランザクションはすべて終了します。

方法 4

CEMT PERFORM SHUTDOWN IMMEDIATE トランザクションを実行します。

このトランザクションは、Sun MTP を即座に停止します。このトランザクションを発行したユーザーを含むすべてのユーザーセッションが停止され、動作中のトランザクションがすべて終了されます。この方法の結果は、方法 3 と同じです。

注 – これらの方法はすべて、外部セキュリティーマネージャーでセキュリティーをチェックする必要があります。

kixstop による領域の停止

kixstop ユーティリティーを使用すると、コマンド行から領域を停止できます。kixstop を使用するには、少なくとも 1 つのクライアントが使用できる必要があります。コマンドプロンプトで kixstop を呼び出すと、端末ハンドラが起動され、CEMT PERFORM SHUTDOWN トランザクションを発行します。このトランザクションでは、すべてのトランザクションを終了し、領域を休止してから、領域を正しく停止できます。

kixstop に `-i` オプションを使用すると、プログラムは `CSMT SHUT,YES` トランザクションを発行します。このトランザクションは、システムを即座に停止します。動作中のトランザクションはすべて終了します。

注 – トランザクションセキュリティーを有効にし、CEMT または CSMT トランザクションを保護している場合、セキュリティーアクセスを持つユーザーだけが、kixstop を実行できます。

組み込みのセキュリティーによるトランザクションのセキュリティー保護については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。外部セキュリティーマネージャーの Sun Mainframe Security Facility によるトランザクションのセキュリティー保護については、『Sun Mainframe Security Facility 管理者ガイド』を参照してください。

クライアント接続

表 3-1 は、ローカルまたは遠隔システムからの Sun MTP への接続およびほかの遠隔システムに対してサポートする接続について簡単にまとめています。また、各タイプの接続に必要なサーバプロセスとクライアントプロセスについても示します。

Sun MTP クライアントとサーバプロセスの説明については、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

表 3-1 Sun MTP へのクライアント接続

接続タイプ	プロセス	説明
ASCII または PC 端末、あるいは X term からのローカルクライアント	unikixmain unikix クライアント	unikix プログラムは、すべてのオプションをローカルクライアント unikix1 に渡します。詳細は、24 ページの「ローカル端末クライアントの起動」を参照してください。
TCP/IP ソケットクライアント	unikixmain, unikixsock (ソケットリスナープロセス)、 ユーザー作成のクライアントアプリケーション	遠隔システムのクライアントアプリケーションは、Sun MTP ホストに要求を送信します。このホストは、あらかじめ定義された TCP/IP ポートを待機しています。リスナープロセスは、トランザクション処理プログラムにこの要求を送信します。詳細は、134 ページの「TCP/IP ソケットリスナーの起動」を参照してください。
ECI/EPI クライアント	unikixmain, unikixtcp (TCP/IP)、 unikixdcl (SNA)、 Sun MTP クライアント	ユーザーは Sun MTP ホストに接続し、外部呼び出しインタフェース (ECI) または外部表示インタフェース (EPI) を使用するユーザー作成のアプリケーションプログラムからデータを転送します。詳細は、133 ページの「ECI/EPI クライアントの領域の構成」を参照してください。
EPI アプリケーション	unikixmain, ユーザー作成のクライアントアプリケーション	ユーザーは、ATM やバーコードリーダーなどの GUI 用 EPI または通常端末以外の端末クライアントを使用するユーザー作成のアプリケーションから Sun MTP ホストに接続します。 EPI の説明については、『External Presentation Interface (EPI) Technical Note』を参照してください。

表 3-1 Sun MTP へのクライアント接続 (続き)

接続タイプ	プロセス	説明
TN3270 と TN3270E エミュレータ	unikixmain, unikixtnemux, Telnet クライアント	ユーザーは、PC、Macintosh、ローカルまたは遠隔の UNIX システム上の TN3270 または TN3270E エミュレータから Sun MTP に接続します。詳細は、117 ページの「TN3270 クライアントへの領域の構成」を参照してください。
Sun MTP J3270 エミュレータ	j3270 j3270.exe	ユーザーは、Solaris、UNIX、Microsoft Windows システムから Sun MTP に接続します。Sun MTP J3270 は、3278 端末のモデル 2、3、4、5 の拡張モードと非拡張モードの両方をサポートします。詳細は、『Sun MTP J3270 ユーザーズガイド』を参照してください。
3270 端末	unikixmain, unikixtrin, unikixi クライアント イニシエータ、 3270 デバイスクライ アント	ユーザーは、オプションの TPS PU4/PU5 ソフトウェアを使用して、3270 デバイスから Sun MTP に接続します。 \$KIXHOST が設定されている場合、unikixmain が unikixtrin を開始させ 3270 端末との通信が可能になります。ユーザーが端末を設定するために unikixi を呼び出すと、unikixb が開始されます。 3270 端末は、ローカル接続されるか (PU4)、別のホストに所有されます (PU5)。 詳細は、123 ページの「SNA 3270 クライアントの領域の構成」を参照してください。

第4章

構成ガイドライン

この章では、アプリケーション環境を構成するためのガイドラインについて説明します。この章の内容は、次のとおりです。

- 32 ページの「Sun MTP トランザクション処理プログラム」
- 35 ページの「回復による構成への影響」
- 35 ページの「Sun MTP プロセス数の決定」
- 36 ページの「ローカルプロセスメモリー」
- 40 ページの「共有メモリー」
- 44 ページの「メモリープロセスマップ」
- 46 ページの「VSAM バッファープールおよび VSAM バッファの数の設定」
- 48 ページの「スワップ空間の決定」
- 48 ページの「カーネルのパラメータ」
- 52 ページの「コードページ変換テーブルの使用」

『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』も参照してください。ここでは、VSAM ファイルサイズなど、Sun MTP リソースの制限値が一覧表示されています。

Sun MTP トランザクション処理プログラム

Sun MTP トランザクション処理プログラムの主要な機能は、トランザクションに関連するプログラムを実行することです。領域には、少なくとも 1 つのトランザクション処理プログラムが必要ですが、224 を超えることはできません。

トランザクション処理プログラムは、「テキスト」セグメントと「データ」セグメントから構成されています。

- テキストセグメントは、Sun MTP コード、COBOL、PL/I ランタイムコード、C コード、およびその他の Sun 以外のオブジェクトで構成されます。Sun 以外のオブジェクトとは、たとえば、リレーショナルデータベース管理システム (RDBMS) 実行時システムやソートランタイムシステムまたはその他のアプリケーションコードなどです。
- データセグメントは、作業記憶域、Sun MTP 環境変数、RDBMS、割り当てメモリー、およびアプリケーションプログラムモジュールで構成されます。

Sun MTP 領域では、すべてのトランザクション処理プログラムは同じテキストセグメントを持ちます。ただし、データセグメントは実行するトランザクションの種類やサイズによって変わります。デフォルトでは、各トランザクション処理プログラムのデータセグメントサイズはあらかじめ設定されています。次の図は、トランザクション処理プログラムの構造を示します。

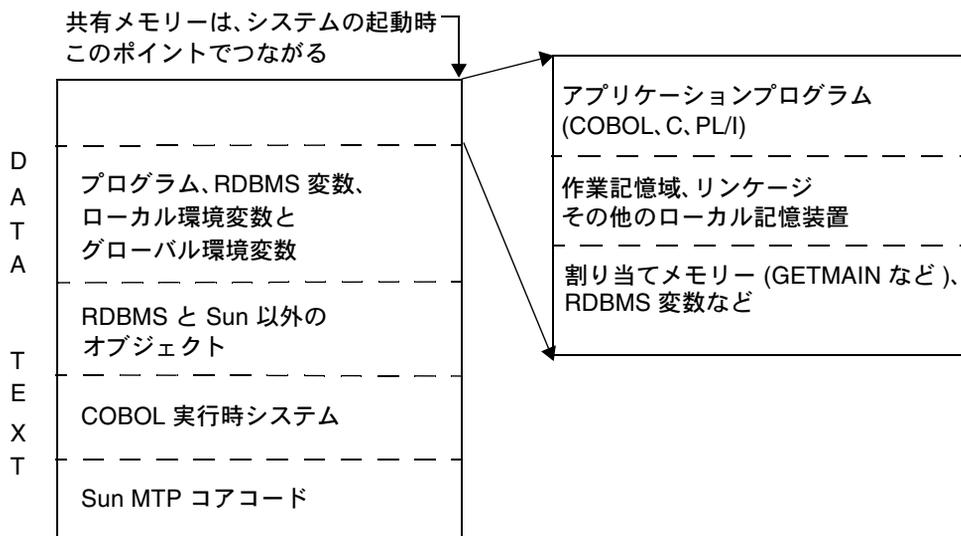


図 4-1 トランザクション処理プログラムの構造

トランザクション処理プログラムの数の決定

VCT の「Transaction Servers」フィールドで、領域に使用可能なトランザクション処理プログラムの数を指定します。トランザクション処理プログラムは、トランザクションやオンラインプログラム、またバッチプログラムの実行に必要なため、このフィールドは同時に実行できるトランザクションとプログラムの最大数を定義することになります。Sun MTP は、この値を使用して、トランザクション処理プログラムが使用可能かどうかを判断します。

このフィールドに設定する値は、使用しているアプリケーション環境によって決まります。たとえば、すべてのログインユーザーが、擬似会話型トランザクションではなく、会話型のトランザクションを実行する場合、この数字は、少なくともログインユーザーと同じ数が必要です。ただし、ログインユーザーが擬似会話型トランザクションだけを実行する場合、この数字は小さくしてもかまいません。

24 ~ 36 の値を使用します。トランザクションが小さなプログラムで、ユーザーがユーザー画面に大量のデータを入力する場合は、これらの値より小さくできます(ユーザーの入力は、トランザクション処理プログラムがトランザクションを処理する速度よりも遅いため)。これらのトランザクションがバックグラウンドのトランザクションを起動する場合、値は 36 にします。

kixdump -sq の出力を調査すると、プロセスキューで待機しているトランザクションがあるかどうかを特定できます。効率的なシステムには、待機中のトランザクションがほとんどないか、まったく存在しません。ご使用の環境に最適な数を判断するために、トランザクション処理プログラムの数の値を変えて実験してください。次の表に示す値は、有効なユーザー数に基づいて、トランザクション処理プログラムの数を設定する際の目安になります。非同期トランザクション、バッチジョブ、照会ジョブの数によって、必要なトランザクション処理プログラムの数が増大します。必要なパフォーマンスを得るためには、システムを監視し、トランザクションプロファイルを調査する必要があります。

表 4-1 トランザクション処理プログラムの数の決定

有効なユーザー数	トランザクション処理プログラムの数
250	30 ~ 40
500	60 ~ 70
500 より多い	80 ~ 224

注 - 本番稼働環境に最適なトランザクション処理プログラムの数を決定する場合、CMNU、CEMT、CEBR、および CEDF などの会話型トランザクションを考慮する必要はありません。代わりに、本番稼働領域ではこれらのトランザクションの使用を制限してください。

トランザクションクラスを使用してアプリケーションの作業負荷を管理する方法については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

バックグラウンドタスクの最大数の決定

VCT の「Maximum Background Tasks」フィールドは、同時に実行できるバックグラウンドトランザクションとプログラムの最大数を指定します。バックグラウンドタスクとは、端末に接続されずに実行されるプログラムまたはトランザクションです。このフィールドの値は、トランザクション処理プログラムに設定した値の 50% を超えてはいけません。ただし、アプリケーションが TRIGGER または START コマンドで頻繁に開始されるバックグラウンドトランザクションに依存する場合、トランザクション処理プログラムに設定した値の 75% の値に設定できます。

バッチジョブの最大数の決定

VCT の「Maximum Batch Jobs」フィールドは、バッチトランザクションを実行するために使用できるトランザクション処理プログラムの数を指定します。このフィールドの値は、トランザクション処理プログラムに設定した値の 50% を超えてはいけません。

注 - 「Maximum Background Tasks」と「Maximum Batch Jobs」の合計は、トランザクション処理プログラムの数と等しくするか、それより小さくする必要があります。

照会ジョブの最大数の決定

VCT の「Maximum Query Jobs」フィールドは、同時に実行できる会話型プロセスの合計数を設定します。このパラメータは、Sun 以外のタスク処理に割り当てるトランザクション処理プログラムの数を制限します。

回復による構成への影響

回復は、領域の構成に影響を及ぼすことがあります。

- VSAM は、ロックされた各リソース (レコード) に約 300 バイトの共有メモリーを使用します。バッチプログラムに VSAM RC (読み取り整合性) を使用していない場合は、これによる領域への影響はありません。これは、VSAM が、バッチプログラムによって更新されるすべてのレコードにロックを必要とするからです。使用するバッチプログラムが多数のレコードを更新する場合は、共有メモリーのサイズを増やすことを考慮する必要があります。詳細は、40 ページの「共有メモリー」を参照してください。別の方法として、頻繁に変更をコミットするようにバッチプログラムを変更できます。
- バッチプログラムを実行する場合には、各コミットとコミットの間で残さなければならない更新前イメージやブロック分割情報を保持できるだけの容量が、回復ファイルにあることを確認します。

Sun MTP プロセス数の決定

Sun MTP は、初期化時にトランザクション処理プログラムとシステムプロセスを生成します。Sun MTP が生成するプロセスの数とユーザーが生成するプロセスの数を計算することで、プロセスの合計数を決定できます。

プロセス数 = トランザクション処理プログラムの数

- + Sun MTP 回復ごとに 1 プロセス
- + 間隔制御ごとに 1 プロセス
- + 印刷サービスごとに 1 プロセス
- + ほかのプロセスごとに 1 コントローラプロセス
- + (オプションで) 遠隔端末通信ごとに 1 プロセス
- + (オプションで) TCP/IP ソケットを使用した通信ごとに 1 プロセス
- + (オプションで) SNA を使用した遠隔領域からの ISC ごとに 1 プロセス
- + (オプションで) TCP/IP を使用した遠隔クライアント/領域からの ISC ごとに 1 プロセス
- (オプションで) Sun MTP 管理フレームワークごとに 1 プロセス
- + (オプションで) WebSphere MQ 統合ごとに 1 プロセス
- + (オプションで) Sun MTP 通信マネージャーごとに 1 プロセス
- + (オプションで) TN3270 サーバーごとに 1 プロセス
- + (オプションで) トランザクションクラスごとに 1 プロセス

システムの実行時、Sun MTP は、たとえば CTBL、CFMS、および印刷要求のトランザクションを実行するスタンドアロンプロセスのような特定のタスクを実行するプロセスを追加することがあります。

領域への接続方法によっては、クライアントに追加のプロセスが必要になる場合があります。

- ユーザーがローカルにログインする場合、少なくとも2つのプロセスが必要です。最初のプロセスはシェルを実行し、2つ目のプロセスはunikix1などのSun MTP クライアントを実行します。X 端末のユーザーの場合は、さらにプロセスが必要になります。
- 遠隔非 SNA クライアントは、Sun MTP サーバーを実行するシステムに追加のプロセスを必要としません。
- SNA ユーザーは、SNA サーバーとして1つのプロセスが必要です。また、SNA サーバーがクライアントと同じプラットフォーム上で実行している場合は、各 SNA ユーザーに追加プロセスが1つ必要です。

このため、VCT 画面の「Maximum number of users」フィールドは、次のようにプロセスの数に影響します。

最大ユーザープロセス数 = ローカルユーザーの最大数 * 2
+ SNA ユーザーの最大数 + 1 (SNA を使用している場合)

ユーザーの最大数は、VCT で指定したユーザー数を超過できません。このため、領域に必要なプロセスの合計数は、次のように決定されます。

合計プロセス数 = Sun MTP サーバープロセス数
+ 最大ユーザープロセス数

ローカルプロセスメモリー

unikixtran プロセスが使用できるローカルプロセスメモリーまたはヒープ領域の量は、`-M t` および `-M c` 領域起動オプションによって指定します。しきい値のパラメータである `-M t` は、ローカルプロセスメモリー (ヒープ) がどのくらいまで増大できるかを指定します。最大コアのパラメータである `-M c` は、unikixtran プロセスが自動的に再起動する前に使用可能な、しきい値を超えるメモリー量を指定します。メモリーがしきい値を超えて割り当てられても、現在のトランザクションは続行され、最大コアに達していなければ正常に終了します。その時点で、unikixtran プロセスが再起動されます。現在のトランザクションのメモリー割り当てが最大コアに達した場合、トランザクションは即座に強制的に中止され、unikixtran プロセスが再起動されます。新しい unikixtran プロセスには、新規のヒープメモリーがあります。

しきい値の最小値、または `-M t` オプションを指定していない場合のデフォルト値は、32M バイトです。デフォルトの `-M c` の値は (指定していない場合)、4M バイトです。つまり、デフォルトでは、トランザクション処理プログラムのローカルプロセスメモリーは、現在実行しているトランザクションを強制的に中止して新しい unikixtran プロセスを起動する前に、36M バイトまで増大することが可能です。

注 – Sun MTP のトランザクション処理プログラムに結合されている、RDBMS などの Sun 以外の製品を使用している場合は、unikixtran のプロセスローカルメモリーが影響を受けることがあります。ローカルプロセスメモリーの割り当て要件に関するガイドラインについては、その製品のマニュアルを参照してください。

Solaris プラットフォーム

注 – 8.1.0 よりも前のリリースの Sun MTP から移行する場合、現在の `-M t` および `-M c` 起動オプションの指定内容を維持するときは、『Sun Mainframe Transaction Processing ソフトウェア インストールガイド』の移行に関する章を参照してください。

Solaris プラットフォームでは、`-M t` および `-M c` オプションは、仮想メモリアドレス `0xa00000` からの相対的な値を指定します。

`-M t` または `-M c` オプションを使用せずに領域を起動すると、ローカルプロセスメモリーの合計は仮想メモリアドレス `0xa00000` から 36M バイトまで増大することが可能です。

しきい値の仮想メモリアドレス: $a00000 + 32M = 2a00000$

最大コアの仮想メモリアドレス: $2a00000 + 4M = 2e00000$

`kixdump -M` を実行すると、次の内容が報告されます。

```
Values of Shared Memory, Max Core, Threshold Memory
Address of Shared Memory      (2e00000 )
Address of Max Core           (2e00000 )
Address of Threshold Memory   (2a00000 )
```

`-Mt 70M` および `-Mc 12M` と指定して領域を起動すると、ローカルプロセスメモリーの合計は仮想メモリアドレス `0xa00000` から 82M バイトまで増大することが可能です。

しきい値の仮想メモリアドレス: $a00000 + 70M = 5000000$

最大コアの仮想メモリアドレス: $5000000 + 12M = 5c00000$

kixdump -M を実行すると、次の内容が報告されます。

```
Values of Shared Memory, Max Core, Threshold Memory
Address of Shared Memory      (5c000000 )
Address of Max Core           (5c000000 )
Address of Threshold Memory   (50000000 )
```

kixdump コーティリティーの詳細については、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

AIX プラットフォーム

AIX プラットフォームでは、`-M t` および `-M c` オプションは、仮想メモリアドレス `0x20400000` からの相対的な値を指定します。

`-M t` または `-M c` オプションを使用せずに領域を起動すると、ローカルプロセスメモリーの合計は仮想メモリアドレス `0x20400000` から 36M バイトまで増大することが可能です。

しきい値の仮想メモリアドレス: $20400000 + 32\text{M バイト} = 22400000$

最大コアの仮想メモリアドレス: $22400000 + 4\text{M バイト} = 22800000$

kixdump -M を実行すると、次の内容が報告されます。

```
Values of Shared Memory, Max Core, Threshold Memory
Address of Shared Memory      (30000000 )
Address of Max Core           (22800000 )
Address of Threshold Memory   (22400000 )
```

`-Mt 70M` および `-Mc 12M` と指定して領域を起動すると、ローカルプロセスメモリーの合計は 82M バイトまで増大することが可能です。

しきい値の仮想メモリアドレス: $20400000 + 70\text{M バイト} = 24a00000$

最大コアの仮想メモリアドレス: $24a00000 + 12\text{M バイト} = 25600000$

kixdump -M を実行すると、次の内容が報告されます。

```
Values of Shared Memory, Max Core, Threshold Memory
Address of Shared Memory      (30000000 )
Address of Max Core           (25600000 )
Address of Threshold Memory   (24a00000 )
```

注 -M t および -M c オプションの組み合わせは 256M バイトを超えることはできません。

kixdump ユーティリティの詳細については、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

また、AIX プラットフォームでは、-M t および -M c オプションは ulimit データ値にも関連付けられています。コマンド ulimit -d をコマンド行で入力すると、使用しているユーザー ID のローカルプロセスメモリーに対するシステムの制限が表示されます。-M t および -M c オプションの合計がこの値を超えている場合は、領域は起動しません。合計が ulimit -d の値よりも小さくなるように -M t および -M c オプションの値を調整するか、プロセスメモリーのサイズを変更する次のコマンドを使用して、ローカルプロセスメモリーのサイズを増やす必要があります。

```
$ ulimit -d nnnnn
```

この nnnnn は、プロセスメモリーの新しいサイズです。

次の例には、ulimit のデータと、-M t および -M c の値との相互作用が示されています。

例 1:

ulimit -d の出力が 131,072K バイトまたは 128M バイトを示している。

-M t および -M c 起動オプションを指定しない場合は、デフォルト(-M t32M と -M c4M) が使用されます。領域が起動し、トランザクション処理プログラムのローカルプロセスメモリーは、強制的に中止して再起動する前に 36M バイトまで増大することが可能です。

例 2:

ulimit -d の出力が 131,072K バイトまたは 128M バイトを示している。

アプリケーションが大量のローカルメモリーを割り当てる場合は、-M t100M および -M c10M オプションを起動時に指定できます。領域が起動し、トランザクション処理プログラムのローカルプロセスメモリーは、強制的に中止して再起動する前に 110M バイトまで増大することが可能です。

例 3:

`ulimit -d` の出力が 131,072K バイトまたは 128M バイトを示している。

アプリケーションに約 200M バイトのローカルメモリーが必要な場合、`-M t200M` および `-M c4M` オプションを起動時に指定しても、204M バイトはシステム制限である 128M バイトを超えているため領域は起動しません。`ulimit` コマンドを使用して、プロセスメモリーを増加する必要があります。たとえば、メモリーを 210M バイト (215,040K バイト) に増やすには、次のコマンドを実行します。

```
$ ulimit -d 215040
```

これで、`-M t` および `-M c` オプションがシステム制限内に収まるため、領域が起動します。

AIX の大規模なアドレス空間モデル

Sun MTP は、ローカルプロセスメモリーにセグメント 2 を超えて追加のセグメントを割り当てる方法である、AIX での大規模なプロセスアドレス空間モデルをサポートしていません。



注意 – `EXTSHM` 環境変数による拡張共有メモリーの使用はサポートされていません。

共有メモリー

Sun MTP は、6 種類の共有メモリーセグメントを使用します。

- トランザクション処理プログラムのシステム領域
- 端末領域
- VSAM バッファ領域
- スピンロック領域 (Solaris プラットフォームではデフォルト、AIX プラットフォームではオプション)
- システム領域
- ユーザー領域

最初の 4 つの領域は、「静的」です。VCT の構成に基づいて、起動時に決定されません。システム領域とユーザー領域は、「動的」な共有メモリーセグメントです。領域の起動時にユーザーがサイズを定義します。システム領域は、ユーザー処理に基づいて割り当てられた Sun MTP 内部構造を格納するために使用します。ユーザー領域は、`COMMAREA` など、アプリケーションから見えるユーザーが要求した領域および Sun MTP が割り当てた領域を格納するために使用します。

共有メモリーに必要な容量の計算

次の表では、各共有メモリーについて説明し、共有メモリーに必要な容量を計算するために使用する式を示します。

表 4-2 共有メモリー領域

種類	共有メモリー領域	説明
静的	トランザクション領域	トランザクション処理プログラムが、実行トランザクションに関するコントロール構造に使用する共有メモリーセグメント。次の式を使用します。 ユーザー領域のサイズ = (トランザクション処理プログラムの数) * 8312 バイト
静的	端末領域	端末ユーザーに関連するコントロール構造に使用する共有メモリーセグメント。次の式を使用します。 グローバル領域サイズ = (ユーザー数) * 32K バイト
静的	VSAM バッファ領域	VSAM アクセス方法で、第 1 レベルのバッファキャッシュとして使用する VSAM バッファ領域共有メモリーセグメント。サイズは、VCT の「Number of Shared Buffers」フィールドに、ブロックあたりのバイト数をかけたものと同じです。 データベースバッファ領域のサイズ = (バッファの数) * VSAM ブロックサイズ。 unikixmain に -b オプションを付けて、「VSAM block size」を指定します。詳細は、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。
静的	スピロック領域	Solaris プラットフォーム: 起動時に自動的に作成されます。 AIX プラットフォーム: オプションの領域。unikixmain の -j オプションが指定されている場合、領域の起動時に作成されます。
動的	システム領域 ユーザー領域	Solaris プラットフォーム: unikixmain に -s オプションを使用して指定します。デフォルトの値は、16M バイトです。 AIX プラットフォーム: デフォルトの値は、256M バイトです。11 個のすべての共有メモリーセグメント (使用可能な場合) は、必要に応じて Sun MTP によって割り当てられます。 デフォルトよりも小さいサイズの共有メモリーセグメントで領域を起動した場合、そのあとの共有セグメントの割り当てはいずれもデフォルトのサイズ (256M バイト) になります。

動的共有メモリー

Sun MTP の実行に必要な動的メモリー容量は、多くのアプリケーションとシステム要因によって決まるので、動的共有メモリーの容量は実験的に決めていくしかありません。

ユーザー共有メモリーのセグメントサイズは、少なくとも実行するオンラインプログラムの中で最大の作業記憶域と同じ容量が必要です。

動的共有メモリーの使用に影響を及ぼす可能性のある要因には次のものがあります。

- アプリケーションが使用する共有メモリー
- 同時 VSAM ロックの数
- Sun MTP リソース定義
- Sun MTP 拡張追跡機能
- 間隔制御コマンド
- 起動されたトランザクション
- ISC
- COBOL 作業記憶域
- BMS マップ
- 一時記憶域構造
- 一時データ構造
- VSAM の同時ブラウズ数

動的共有メモリーのサイズは、共有メモリーのセグメントサイズを指定する、unikixmain の -s オプションで制御されます。

Solaris での動的共有メモリーのサイズ

デフォルトの値は、16M バイトです。処理要求が必要とした場合は、動的領域が新たに作成されます (10 まで)。5 つ目の領域が追加されたあと、可能であれば、前のサイズを 2 倍にして各追加領域が追加されます。オペレーティング環境が、この増大したサイズを拒否する場合、領域のサイズは半分にされて、再度追加が試みられます。

サイズは、数字または M (メガバイト) あるいは K (キロバイト) を付けた数字で指定できます。

AIX での動的共有メモリーのサイズ

unikixmain の -s オプションの有効な値の範囲は、16 ~ 256M バイトです。AIX では、このパラメータのデフォルト値は 256M バイトです。これは、ほとんどのアプリケーション環境に対して最適な値です。

AIX の標準のアドレス空間モデルでは、プロセスは、それぞれが最大 256M バイトである 11 の共有メモリーセグメントに制限されます。このため、256M バイト未満のセグメントサイズを使用して (たとえば、起動パラメータとして -s32M を使用して) 共有メモリーを割り当てるように Sun MTP に指定しても、Sun MTP は割り当てられたセグメントの最初の 32M バイトにしか対処できません。仮想アドレス空間の残りの 224M バイトは、セグメントに使用できません。この結果、Sun MTP やアプリケーションがさまざまな共有メモリーセグメントに格納できるデータの量が制限されます。

この制限に対処するため、Sun MTP はデフォルトの共有メモリーセグメントのサイズである 256M バイトを使用します。-s 起動パラメータを使用して、小さいセグメントサイズを指定することは可能ですが、16M バイト未満の値、または 256M バイトを超える値で領域を起動しようとする、Sun MTP によって、この値よりもデフォルトの 256M バイトが優先され、警告メッセージが出力されます。

AIX では、領域が起動され、トランザクション処理プログラムが COBOL 環境を呼び出すと、Micro Focus のライセンスマネージャーが 1 つの共有メモリーセグメントを割り当て、Sun MTP と Sun 以外のソフトウェア用に最大 10 個の共有メモリーセグメントが残されます。

注 – AIXでは、構成または調整が必要な共有メモリーのカーネルパラメータはありません。AIX では、SHMMAX、SHMSEG、SHMMNI は有効ではありません。

共有メモリーの追加先

Solaris プラットフォーム:

共有メモリーは、最大コアに追加されます。

共有メモリーの追加ポイントは、unikixmain -M t オプションおよび -M c オプションを使用して変更できます。最大しきい値および最大コアの値については、36 ページの「ローカルプロセスメモリー」を参照してください。

AIX プラットフォーム:

共有メモリーは、0x30000000 に追加されます。-M t および -M c オプションは、共有メモリーの追加先には影響しません。

共有ライブラリの使用

Solaris プラットフォーム

共有ライブラリは、libc.so など、Solaris オペレーティング環境が提供するライブラリ、ceci.so など Sun MTP が提供するライブラリ、Sun 以外の製品が提供するライブラリでもかまいません。共有ライブラリがロードされると、共有ライブラリはライブラリ領域のアドレス空間を占有し、Sun MTP 共有メモリーより上位のアドレス空間を使用します。共有ライブラリ、または新しい Sun MTP 共有メモリーセグメントのアドレス空間がなくなった場合、Solaris オペレーティング環境は errno 22 (shmat の場合) を発行します。Sun MTP は、errno 22 の状態を検出して、エラーメッセージ KIX0208E を出力します。

AIX プラットフォーム

AIX の共有ライブラリは、独立した 256M バイトのセグメントを利用するため、11 個の共有メモリーセグメントとの競合は発生しません。

メモリープロセスマップ

Solaris プラットフォーム

Solaris の `pmap` コマンドは、プロセスのメモリーマップを表示します。UNIX コマンド `ps -elf` は、メモリーのページにあるプロセスのサイズを一覧表示します。これらのコマンドの詳細については、該当のマニュアルページを参照してください。

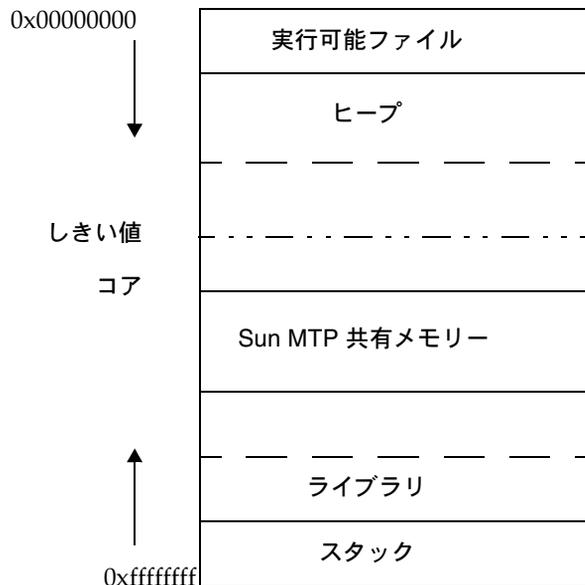


図 4-2 Solaris のメモリーのイメージ

AIX プラットフォーム

procmap と svmon という 2 つの AIX コマンドでは、システムで実行されているプロセスのメモリのレイアウト情報が得られます。svmon コマンドでは、共有メモリーセグメントの利用状況に関する情報が得られます。これらのコマンドの詳細については、AIX のマニュアルを参照してください。

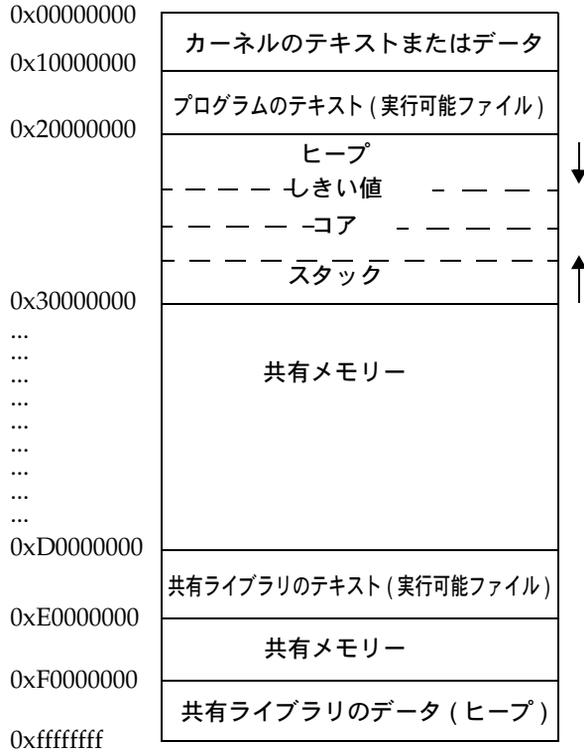


図 4-3 AIX のメモリーのイメージ

VSAM バッファープールおよび VSAM バッファの数の設定

Sun MTP は、メモリー内バッファープールを使用して、最近アクセスされた VSAM データを管理します。デフォルトでは、領域は 16 個のバッファープールを使用して設定されます。すべての VSAM 操作では、いずれかのバッファープール内の、いずれかのバッファからデータを取得する必要があります。バッファを検索している間、バッファープールはロックされます。デフォルトではバッファープールの数は 16 であり、16 の並行アクティビティーを行うことができます。ただし、領域に多数のトランザクション処理プログラム (100 など) が存在し、多数のトランザクションが VSAM データにアクセスする場合は、16 個のバッファープールでは十分ではないことがあります。このため、一部のトランザクション処理プログラムはバッファープールへのアクセスを待機しなければならなくなり、パフォーマンスが低下する場合があります。バッファープールの数を増やすと、トランザクション処理プログラムがバッファープールへのアクセスを待機する時間を減らすことができます。

注 ディスクに書き込まれる一次記憶域キューと一次データキューへのアクセスは、実質的には VSAM ファイルへのアクセスであるため、設定されているバッファープールの数が領域のパフォーマンスに影響します。このため、領域に必要なバッファープールの数を判断するときは、このような VSAM ファイルへのアクセスを考慮します。

この VSAM バッファーマモリーは、共有メモリー領域に割り当てられますが、主記憶の一部でもあります。

デフォルトの相互排他ゲート制御メカニズムを使用して領域を実行している場合のみ、バッファープールの数を増やすことができます。セマフォアを使用して実行されるように領域が構成されている場合、KIXVSAMPOOLS 環境変数が設定された状態で領域を起動しようとする、エラーメッセージが表示され、領域は起動しません。セマフォアを使用して実行されている領域は、バッファープールを 16 個しか使用できません。

バッファープールの数を増やすときは、VSAM バッファの数も増やして各バッファープールに少なくとも 1 つのバッファが含まれるようにする必要があります。バッファープールの数を増やすには、領域を起動する前に KIXVSAMPOOLS 環境変数を設定します。VSAM バッファの数を増やすには、VCT の「Number of Shared Buffers」フィールドを変更します。VSAM バッファの数を調べるには、バッファープールの数を 8 で乗算します。たとえば、VSAM バッファープールの数を 128 に増やす場合は、次のようにバッファの数を計算します。

$$\text{バッファの数} = 128 * 8 = 1024$$

バッファープールの数は、常に索引プールとデータプールに 2 等分されます。ただし、unikixmain 起動コマンドに -I オプションを使用して、索引ブロックに割り当てるバッファール数を指定できます。このオプションを使用しない場合、索引ブロックとデータブロックに割り当てられるバッファールの数は等しくなります。

例

100 個のバッファープール (KIXVSAMPOOLS=100) と 8,000 個のバッファールを使用して領域を構成しています。50 個のデータプールと 50 個の索引プールが存在し、バッファールはこれらのプールに均等に割り当てられているため、データバッファールが 4,000 個で、索引バッファールが 4,000 個になっています。そうではなく、索引バッファールがデータバッファールよりも多くなるように割り当てを変更したい場合、unikixmain -I オプションを使用して領域を起動します。たとえば、unikixmain -I 75 と指定すると、8,000 個のバッファールの 75% (6,000 個) が索引バッファールとして割り当てられ、2,000 個がデータバッファールとして割り当てられます。6,000 個の索引バッファールが 50 個の索引プールに均等に分配されるため、各索引プールの索引バッファールは 120 個になります。また、2,000 個のデータバッファールが 50 個のデータプールに均等に分配されるため、各データプールのデータバッファールは 40 個になります。

次の表は、領域に対して構成するバッファープールと VSAM バッファールの数を決定するための目安になります。

表 4-3 VSAM バッファールおよびバッファープール

バッファール プール	バッファール	アプリケーション環境
16	128	一時記憶域キューおよび一時データキューの使用は最小限。VSAM データセットへのアクセスは最小限。RDBMS へのアクセスのみ
32	512	一時記憶域キューおよび一時データキューの使用は中程度。VSAM データセットへのアクセスは最小限。RDBMS へのアクセス
256	2048 ~ 4096	一時記憶域キューおよび一時データキューの使用はやや多い。VSAM データセットへのアクセスは中程度。
512	4096 ~ 8192	一時記憶域キューおよび一時データキューの使用は非常に多い。VSAM データセットへのアクセスは非常に多い。

アプリケーションが大量のバッファープールに適しているかどうかを調べる方法については、『Sun Mainframe Transaction Processing ソフトウェア 障害追跡とチューニング』を参照してください。

スワップ空間の決定

スワップ空間は、ホストシステム上のその他の処理と Sun MTP の仮想記憶要件に合わせる必要があります。スワップ空間は、同時に実行するすべてのアプリケーションの記憶要件の合計と等しくするか、それより多くする必要があります。

ターゲットシステムの現在のファイルシステムが必要なスワップ空間を提供できない場合、ファイルシステムの構成を変更する必要があります。多くのシステムは、ディスクのパーティションの実際のブロックレイアウトを表示するコマンドをサポートしています。

注 – 有効なスワップパーティションと空のパーティションを間違えないようにしてください。スワップ空間の縮小は、最後の手段としてだけ実行してください。

空のパーティションは、使用可能なファイルシステムに、`newfs` コマンドやその派生コマンドで作成します。ほとんどのシステムには、`mkfs` コマンドがあり、そのプリプロセッサは `newfs` です。このコマンドは、ディスクパーティションにファイルシステムを作成します。

ファイルシステムは、ユーザーまたはアプリケーションがアクセスできるようにマウントする必要があります。新しいファイルシステムをマウントするには、`mount` コマンドを使用するか、ファイルシステムのマウントテーブル (通常は、`/etc/fstab`) にエントリを追加します。ファイルシステムの詳細については、Solaris のマニュアルを参照してください。

それでもターゲットシステムのリソースに必要な領域を割り当てられない場合、通常は内部または外部にハードディスクを追加できます。

カーネルのパラメータ

一部のオペレーティング環境 (カーネル) のパラメータは、領域の起動やパフォーマンスに影響を与えます。この節では、これらカーネルのパラメータに関する情報を提供します。

UNIX カーネルの構成を決めるパラメータは、1 つ以上のファイルに存在します。パラメータの既定値は、ハードウェアプラットフォームごとに異なる上に、事前に調整されている場合もあるので、現在の設定を見直すことが必要です。パラメータの現在の設定が Sun MTP 要件に一致している、または十分満たしている場合は、変更しないでください。

カーネルのパラメータの変更

カーネル値の変更方法は、システムによって異なります。これらのパラメータの正確な場所やパラメータの変更方法などの詳細については、使用しているオペレーティングシステムのマニュアルを参照してください。システム管理やカーネルのチューニングについて扱う、Sun 以外のマニュアルも数多くあります。

次の表には、プロセス間通信 (IPC) のパラメータが記載されています。

Solaris プラットフォーム: これらのパラメータは調整が必要なことがあります。Solaris 9 プラットフォームでは、この作業には、`/etc/system` ファイルの変更とシステムの再起動が必要です。Solaris 10 プラットフォームでは、調整にリソース管理コマンドを使用します。

AIX プラットフォーム: カーネルのパラメータは動的です。

表 4-4 プロセス間通信のパラメータ

名前	説明
MSGTQL	Solaris のみ。システムで未処理の最大メッセージ数。これは、少なくとも VCT で定義されたユーザーの最大数と同じでなければなりません。複数の領域を使用している場合は、定義されているすべてのユーザーの合計数を使用します。
MSGMNI	システム全体のメッセージキューの数。各 Sun MTP 領域は、(トランザクション処理プログラムの数 + トランザクションクラスの数 + 11) のキューを使用します。同時に実行するすべての領域に必要なメッセージキューの合計数に MSGMNI の Solaris のデフォルト値を加えた値に、MSGMNI を設定します。
SHMMAX	1 つの共有メモリーセグメントのサイズ。このパラメータは、Sun MTP の最大セグメントサイズに設定します。セグメントサイズについては、40 ページの「共有メモリー」で説明します。
SHMMNI	システム内のすべてのプロセスに割り当てられる共有メモリーセグメントの最大数。
SEMMNI	セマフォセットの最大数。Sun MTP では、領域ごとに 2 つのセマフォセットが必要です。
SEMMNS	Solaris 9 で使用します。セマフォの最大数です。Sun MTP では、領域ごとに 25 のセマフォが必要です。
SEMMSL	セマフォセットあたりのセマフォの最大数。Sun MTP では、領域あたりの最大数は 24 です。
SEMMNU	Solaris 9 で使用します。システム全体のセマフォの取り消し構造体の数です。SEMMNS と同じ値に設定します。

次の表には、その他のカーネルのパラメータが記載されています。

表 4-5 カーネルのパラメータ

Sun MTP での 名前	Solaris での名前	AIX での名前	説明
MAXUP	maxuprc	maxuproc	<p>ユーザーごとのプロセスの最大数。Sun MTP 領域は、35 ページの「Sun MTP プロセス数の決定」で述べた多数のプロセスを最初に作成します。次の追加プロセスが作成されます。</p> <p>印刷: 最大で 2 つの追加プロセスが作成されます。</p> <p>デバッグ: デバッグのために別に「Animator」ウィンドウを使用するユーザーごとに、1 つの追加プロセスを作成します。</p> <p>CMNU: 有効なトランザクションごとに、6 つまでのプロセスを開始できます。</p> <p>CTBL: 有効なトランザクションごとに 2 つの追加プロセスが開始されます。</p> <p>CEMT: 有効なトランザクションごとに 2 つの追加プロセスが開始されます。</p> <p>CFMS: 有効なトランザクションごとに 2 つの追加プロセスが開始されます。</p> <p>CRED: 有効なトランザクションごとに、6 つまでのプロセスを開始できます。</p> <p>CSGU: 有効なトランザクションごとに、6 つまでのプロセスを開始できます。</p> <p>バッチ: Sun MTP では、VCT で指定された有効なバッチジョブごとに、少なくとも 2 つの追加プロセスが必要です。ユーザーのシェルスクリプトが追加のプロセスを作成する場合があります。</p> <p>一般に、MAXUP はシステムで最も多くのプロセスを必要とするユーザーが必要なプロセスの最大数に設定します。X 端末のユーザーは、ウィンドウだけで 10 ~ 15 のプロセスを使用することに注意してください。</p> <p>Solaris プラットフォーム: /etc/system ファイルを更新し、システムを再起動する必要があります。</p> <p>AIX プラットフォーム: SMIT を使用して更新します。再起動は必要ありません。</p>
NPROCS	max-nprocs		<p>システムのプロセスの最大数。複数の領域が一度にアクティブになる場合、この値は同時に実行するすべての領域に適応できる大きさにする必要があります。詳細は、35 ページの「Sun MTP プロセス数の決定」を参照してください。この値に領域の最大数を乗じ、ユーザーに必要なプロセス数を加算します。</p> <p>Solaris プラットフォーム: /etc/system ファイルを更新し、システムを再起動する必要があります。</p>

表 4-5 カーネルのパラメータ (続き)

Sun MTP での名前			
名前	Solaris での名前	AIX での名前	説明
	rlim-fd-max		HFNOLIM パラメータの最大値。 Solaris プラットフォーム: /etc/system ファイルを更新し、システムを再起動する必要があります。
NOFILES	rlim-fd-cur		プロセスごとのオープンファイルの最大数である SFNOLIM パラメータのデフォルト値。この値は、Sun MTP にオープンされるすべてのソケット数も含む必要があります。Sun MTP 領域で使用されるファイル数を計算する方法については、『Sun Mainframe Transaction Processing ソフトウェア 障害追跡とチューニング』を参照してください。

次の表では、ユーザーの制限について説明します。

コード例 4-1 ユーザーの制限

名前	適用範囲	説明
SCPULIM	CPU 時間	秒で示すユーザーとシステム時間を組み合わせた最大値に対するソフトリミット。この制限値を超えるプロセスには、SIGXCPU シグナルが送信されず、これは無制限に設定します。
HCPULIM	CPU 時間	SCPULIM の最大値〇
SFSZLIM	最大ファイルサイズ	バイトで示す最大ファイルサイズに対するソフトリミット。この制限値を超えるファイルへの書き込みプロセスには、SIGXFSX シグナルが送信されず、また書き込みは、EFBIG エラーでも失敗します。これは無制限に設定します。
HFSZLIM	最大ファイルサイズ	SFSZLIM の最大値
SDATLIM	最大データサイズ	バイトで示すプロセスのヒープ領域の最大サイズに対するソフトリミット。この制限値を超えてヒープ領域を拡張しようとするプロセスは、ENOMEM が表示されて失敗します。これは無制限に設定します。
HDATLIM	最大データサイズ	SDATLIM の最大値
SFNOLIM	オープンファイル記述子の最大数	ファイル記述子の最大数に対するソフトリミット。この制限値に達した場合、次のファイルをオープンしようとする、EMFILE が表示されて失敗します。アプリケーションに合った適切な値に設定します。
HFNOLIM	オープンファイル記述子の最大数	SFNOLIM の最大値を定義します。

コードページ変換テーブルの使用

この節では、言語に関する要件を満たす、1 バイト文字セット (SBCS) のコードページ変換テーブルを使用して領域を構成する方法を説明します。2 バイト文字セット (DBCS) の変換テーブルの使用法については、第 14 章を参照してください。

注 – コードページ変換テーブルは、あるコードページのグリフに割り当てられている値を、それに対応する別のコードページのグリフの値に変換するために使用されます。

unikixmain コマンドを使用すると、-A オプション付きで使用すると、SBCS コードページ変換テーブルを指定できます。使用しているアプリケーション環境で英語 (U.S.) 以外の言語を使用している場合は、このオプションを使用して領域を起動し、Sun MTP に適切なテーブルを使用するように指定します。

コードページ変換テーブルは、\$UNIKIX/lib ディレクトリに格納されています。Sun MTP の『ご使用にあたって』にも、テーブルの一覧があります。

▼ 領域のコードページ変換テーブルを選択する

1. データを表すのに使用する ASCII コードページを決定します。
これは、VSAM ファイルに格納されたときに文字を表すのに使用するコードページです。また、COBOL ソースや BMS ソースを格納するときに使用されるコードページです。
2. Sun MTP J3270 などの TN3270 エミュレータを使用して Sun MTP 領域に接続している場合は、データの変換先の EBCDIC コードページを決定します。
3. 使用するコードページを変換する、\$UNIKIX/lib ディレクトリの変換テーブルを決定します。
4. unikixmain に -A オプションを付けて Sun MTP 領域を起動し、テーブルを指定します。

▼ TN3270 エミュレータのコードページ変換テーブルを選択する

- TN3270 エミュレータを起動し、前の手順で選択した EBCDIC コードページを指定します。

コードページ変換テーブルを指定せずに領域を起動した場合は、IBM-1047 コードページを使用します。

1 バイト文字のコードページ変換テーブルの例

この例では、言語固有のコードページ変換テーブルを選択して使用方法を説明します。

イタリアのサイトの管理者が、Sun MTP 領域の ASCII コードページとして ISO-8859-15 を使用すると決定しました。このコードページは、ユーロの通貨記号を含んでおり、ほとんどの北ヨーロッパ言語のグリフを表すことができます。

領域のすべてのユーザーは TN3270 クライアントを通じて接続するため、この管理者は EBCDIC コードページとして、「イタリア語用 EBCDIC + ユーロ」で知られている、IBM-1144 を使用することに決めました。

領域を起動するため、管理者は次の -A オプションを使用します。

```
$ unikixmain -A IBM-1144toISO8859-15.table
```

領域に接続するとき、TN3270 クライアントはホストコードページとして IBM-1144 を選択します。Sun MTP J3270 を使用している場合は、コマンド行でコードページを指定するか、GUI でコードページを選択できます。

アプリケーションリソースの定義

領域の環境変数を設定してから、アプリケーションが使用するリソースを定義する必要があります。Sun MTP には、アプリケーションリソースを定義する一連のテーブルがあります。アプリケーションの実行方法を制御するテーブルもあります。ただし、必須テーブルは 4 つだけです。

- VSAM 構成テーブル (VCT)
- プログラム管理テーブル (PCT)
- VSAM ファイルを使用する場合は、ファイル管理テーブル (FCT)
- 処理プログラムテーブル (PPT)

ほとんどのアプリケーション環境では、システム初期化テーブル (SIT) もカスタマイズします。

この章では、領域のアプリケーションリソースを定義する方法について説明します。次のトピックについて説明します。

- 58 ページの「Table Manager の開始」
- 59 ページの「VCT のカスタマイズ」
- 62 ページの「PCT でのトランザクションの定義」
- 64 ページの「FCT での VSAM ファイルの定義」
- 67 ページの「PPT でのリソースの定義」
- 69 ページの「SIT のカスタマイズ」

アプリケーションが VSAM ファイルにアクセスする場合、VSAM カタログも設定する必要があります。詳細は、『Sun Mainframe Transaction Processing ソフトウェア管理者ガイド』を参照してください。

次の表は、アプリケーション要件、関連するテーブル、およびそのテーブルに関する情報が記述されているマニュアルの場所を示します。

表 5-1 アプリケーション要件と関連テーブル

アプリケーション要件	テーブル	マニュアル
システム間通信 (ISC) SNA と TCP/IP	端末管理テーブル (TCT) - すべての ISC 構成で必要	『Sun Mainframe Transaction Processing ソフトウェア 構成ガイド』
ISC: アウトバウンドトランザクション経路指定	PCT SIT	『Sun Mainframe Transaction Processing ソフトウェア 構成ガイド』
ISC: インバウンドトランザクション経路指定	SIT	『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』
ISC: アウトバウンド機能シップ	FCT、宛先管理テーブル (DCT)、一時記憶域テーブル (TST)	『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』
ISC: 機能シップデータ変換	データ変換テンプレートテーブル (CVT)	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』
ISC: アウトバウンド非同期処理	PCT	『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』
ISC: アウトバウンド分散プログラムリンク	PPT	『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』
ISC: インバウンド分散プログラムリンク	PPT、PCT	『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』
3270 デバイス接続	TCT	『Sun Mainframe Transaction Processing ソフトウェア 構成ガイド』
ECI/EPI クライアント	TCT (ISC の設定)	『Sun Mainframe Transaction Processing ソフトウェア 構成ガイド』
TN3270 クライアント、事前定義のみ	TCT	『Sun Mainframe Transaction Processing ソフトウェア 構成ガイド』
アプリケーション作業負荷の管理	トランザクションクラステーブル (TXC)	『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』

表 5-1 アプリケーション要件と関連テーブル (続き)

アプリケーション要件	テーブル	マニュアル
印刷	TCT、DCT	『Sun Mainframe Transaction Processing ソフトウェア 構成ガイド』
ユーザーセキュリティ	サインオンテーブル (SNT)	『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』
グローバルアカウントिंग	SIT	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』
ユーザーアカウントिंगジャーナル	ジャーナル管理テーブル (JCT)	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』
トランザクションとユーザーログ	監視管理テーブル (MCT)	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』
Sun 以外のデータベースの統合	SIT	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』
自動開始プログラム	プログラムリストテーブル (PLT)	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』
一時データコマンドによって処理できるキュー	DCT	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』
テーブル情報のグループ化	グループ管理テーブル (GCT)	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』
一時記憶域コマンドによって処理できる回復可能なキュー	TST	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』
バッチジョブのオンラインでの発行	DCT	『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』

すべてのテーブルの詳細な説明については、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

Table Manager の開始

領域テーブルのカスタマイズや保守には、Table Manager を使用します。テーブルをカスタマイズしてから、領域をシャットダウンして再起動すると、変更が有効になります。

▼ Table Manager を開始する

- 次のいずれかの方法を使用します。
 - ローカルクライアントセッションから、空白のトランザクション画面に CMNU トランザクションを入力し、開発システムメインメニューを表示します。次に、PF6 キーを押して、「Table Manager」メインメニューを表示します。
 - クライアントから、空白のトランザクション画面に CTBL トランザクションを入力すると、開発システムメインメニューを表示しないで、直接「Table Manager」メニューを表示します。

次の図は、「Table Manager」メインメニューを示します。

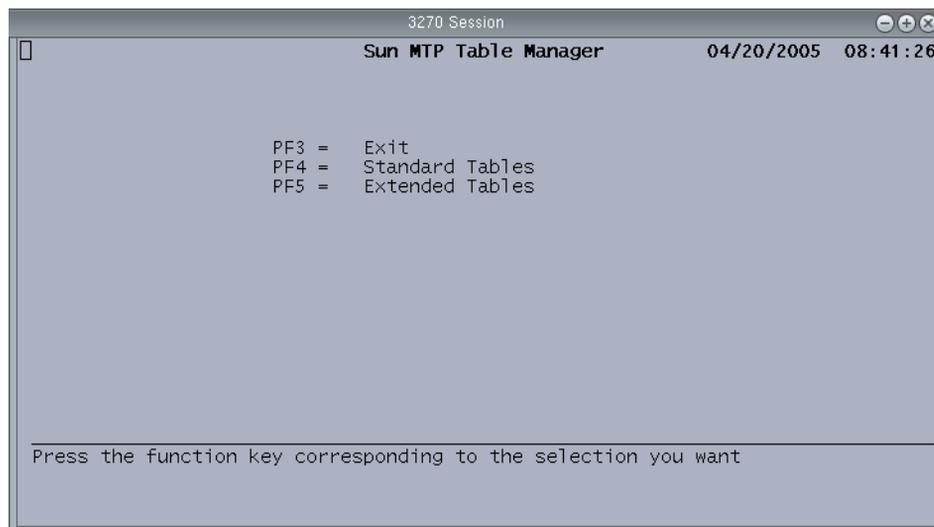


図 5-1 「Table Manager」メインメニュー

この章で説明するテーブルは、「Standard Tables」メニューに表示されます。

▼ 「Standard Tables」メニューを表示する

- 「Table Manager」メインメニューで PF4 キーを押します。

次の画面が表示されます。

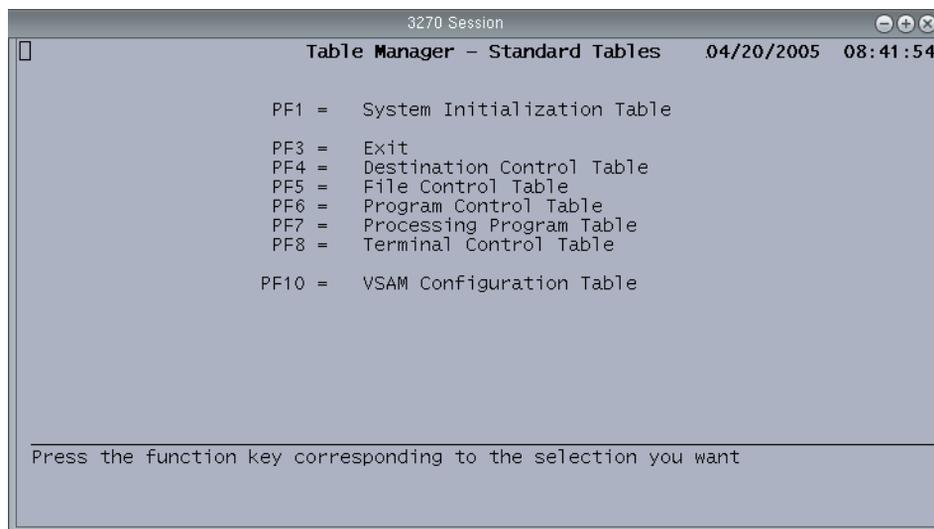


図 5-2 「Standard Tables」メニュー

VCT のカスタマイズ

VCT には、領域の動作を制御するデータが含まれています。VCT のフィールドの値の設定に関するガイドラインは、第 4 章を参照してください。

▼ VCT をカスタマイズする

1. Table Manager を開始し、「Standard Tables」メニューを表示します。
2. PF10 キーを押して、VCT を表示します。

図 5-3 に示す画面が表示されます。

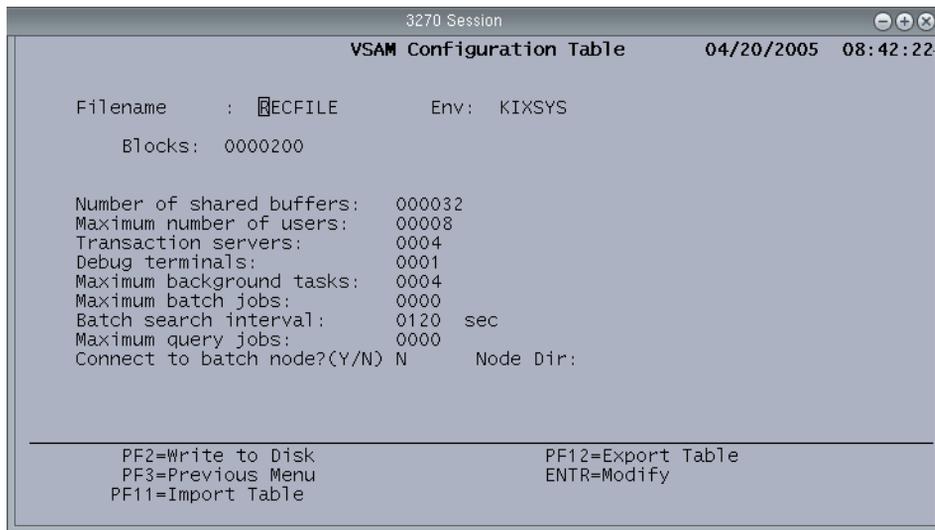


図 5-3 VSAM 構成テーブル (VCT)

3. 回復を設定している場合は、次の手順を実行します。そうでない場合は、手順 4 に進みます。
 - a. 「Filename」フィールドで、回復データファイルの名前を入力します。
 - b. 「Env」フィールドに、回復データファイルが置かれているディレクトリを示す環境変数を入力します。
 - c. 「Blocks」フィールドに、回復ファイルのブロック数を入力します。
この値は、100 以上でなければなりません。そうでないと、領域の起動時に回復ファイルが作成されません。
4. 「Number of Shared Buffers」フィールドに、VSAM データをコアに格納する際の初期化時に割り当てる必要がある VSAM ブロックサイズのバッファの数を入力します。
最小値は 128 です。バッファの最大数は、最大の共有メモリーセグメントサイズによって決まります。詳細は、46 ページの「VSAM バッファープールおよび VSAM バッファの数の設定」を参照してください。
5. 「Maximum Number of Users」フィールドに、同時に領域にログインできる、同時に接続される端末および専用のプリンタの最大数を入力します。
このフィールドの初期値は、Sun MTP ライセンスによって決まります。

6. 「Transaction Servers」フィールドに、同時にアクティブにできるトランザクション、セッション、バッチジョブ、および会話型の照会ジョブの合計数を入力します。最大値は 224 です。
トランザクションは、端末、ソケット、バッチジョブ、ISC セッションなどからサブミットできます。詳細は、33 ページの「トランザクション処理プログラムの数の決定」を参照してください。
7. 「Debug Terminals」フィールドに、有効にできるデバッグセッションの最大数を入力します。
このフィールドの初期値は、Sun MTP ライセンスによって決まります。
8. 「Maximum Background Tasks」フィールドに、アクティブにできるバックグラウンドタスクの最大数を入力します。
この値は、CICS 一時データトリガーと、端末 ID を指定しない START コマンドによって開始されるトランザクション数を制限します。
詳細は、34 ページの「バックグラウンドタスクの最大数の決定」を参照してください。
9. 「Maximum Batch Jobs」フィールドに、同時に実行できるバッチジョブの最大数を入力します。
適切な値を決定する方法については、34 ページの「バッチジョブの最大数の決定」を参照してください。また、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』の「バッチ処理」という章も参照してください。
10. 「Batch Search Interval」フィールドに、実行を待機しているバッチジョブを検索する間隔を秒数で入力します。
このフィールドを小さい値に設定すると、オンラインの応答時間に影響する場合があります。
11. 「Maximum Query Jobs」フィールドに、同時に実行できる会話型の照会プロセスと更新プロセスの最大数を入力します。
12. Sun MBM 製品を使用している場合、「bach node」フィールドに入力する値については、第 12 章を参照してください。
13. PF2 キーを押して、VCT を保存します。
14. PF3 キーを押して、「Standard Tables」メニューに戻ります。

PCT でのトランザクションの定義

PCT には、Sun MTP がトランザクションの ID や初期化に使用する情報が含まれません。これらのトランザクションは、ユーザーやプログラムによって初期化できます。

▼ トランザクションを定義する

1. Table Manager を開始し、「Standard Tables」メニューを表示します。
2. PF6 キーを押して、PCT を表示します。

Program	Task	Trans ID	TWA Size	Trans Sec	Scrn Size	File Acct ID	SysID	Remote TranCd	Group	Loc Que	AP PC
ACCT01		AC01		1	DEF	D				Y	
ACCT02		AC02		1	DEF	D				Y	
ACCT03		AC03		1	DEF	D				Y	
ACCT03		AC05		1	DEF	D				Y	
ACCT00		ACCT		1	DEF	D				Y	
ACCT03		ACLG		1	DEF	D				Y	
CEDA		CEDA		1	DEF	D			rdo	N	
CEDA		CEDB		1	DEF	D			rdo	N	
CEDA		CEDC		1	DEF	D			rdo	N	
BATCH000		CBCH		1	DEF	N			unikix	Y	
KXCCIN		CCIN		1	DEF	D			unikix	N	Y
CEBR		CEBR		1	DEF	N			unikix	Y	
CECI		CECI		1	DEF	D			unikix	N	
CEDF		CEDF		1	DEF	N			unikix	Y	
CEMT		CEMT		1	DEF	N			unikix	Y	

PF2=Write to Disk PF5=Delete Entry PF9=Txn Class PF12=Export Table
PF3=Previous Menu PF7=Previous Page PF10=Search ENTR=Modify
PF4=Insert Entry PF8=Next Page PF11=Import Table

図 5-4 プログラム管理テーブル (PCT)

3. PF4 キーを押して、「PCT Insert」画面を表示します。
4. アプリケーションで必要とされれば、次のフィールドに適切な値を入力します。
 - a. 「Program name」フィールドにプログラム名を入力します。

プログラム名は、CICS コマンドで使用されるので大文字である必要があります。拡張子は使用しません。

Server Express プログラムには、.int または .gnt の拡張子があります。
.gnt と .int の両方がある場合、.gnt が優先されます。

ACUCOBOL-GT[®] プログラムには、.acu 拡張子があります。

Liant Open PL/I プログラムと C プログラムには、.so 拡張子があります。
Java プログラムには、.class 拡張子があります。

- b. 「Task Request」フィールドに、トランザクションとしてこのプログラムを開始するキーを入力します。

有効な値は次のとおりです。

PF1 ~ PF24
PA1 ~ PA3

このフィールドが空白の場合、このプログラムはキーを押しても呼び出されません。

- c. 「Trans ID」フィールドに、トランザクションとしてプログラムを開始する、1 ~ 4 文字の識別子を入力します。

小文字で入力すると、大文字に変換されます。

- d. 「TWA Size」フィールドに、0 ~ 32767 の数値を入力します。

これは、バイトによる、トランザクション作業領域 (TWA) のサイズです。

- e. 「Trans Sec」フィールドに、トランザクションセキュリティレベルを指定する 1 ~ 64 の数字を入力します。

この値は、トランザクションを入力した端末のユーザーに関連付けられた SNT エントリのセキュリティキー領域に設定されているビット値に対応します。セキュリティの詳細は、『Sun Mainframe Transaction Processing ソフトウェア管理者ガイド』を参照してください。

- f. 「Scrn Size」フィールドに、デフォルトの画面サイズを受け入れるか、代替画面サイズを入力します。

- g. 「Acct」フィールドは、このトランザクションについて、アカウントिंगがオンかオフかを示します。次のいずれかの値を入力します。

Y: アカウントिंगは、このトランザクションについてオンです。SIT のアカウントिंगフィールドは、Y である必要があります。

N: このトランザクションに対するアカウントिंगはオフです。

D: アカウントिंगは、SIT の「Accounting」フィールドと MCT で制御されます (デフォルト)。

- h. 「File ID」フィールドに、このトランザクションのジャーナルファイル識別子を定義する 1 ~ 99 の数字を入力します。

アカウントINGが N に設定されている場合、このフィールドは無視されます。

- i. このトランザクションが遠隔システムで実行される場合、「SysID」フィールドに 4 文字のシステム ID を指定します。

この ID は、TCT のシステム項目テーブルの SysID エントリと一致する必要があります。

- j. 「Remote TranCd」フィールドに、このトランザクションを経路指定する際、または遠隔システムでトランザクションの非同期処理を使用する際に、遠隔システムで使用するトランザクションコードを入力します。
 - k. トランザクションがあるグループに属する場合、「Group」フィールドにグループ名を入力します。
グループは、グループ管理テーブル (GCT) に定義しておく必要があります。
 - l. 「Loc Que」フィールドには、Y または N を入力し、遠隔システムが使用できない場合は、遠隔システムにシップされるはずであったトランザクションをローカルキューに入れるかどうかを指定します。
 - m. 「APPC」フィールドを使用して、トランザクションがバックエンドの分散トランザクション処理 (DTP) トランザクションかどうかを示します。
Y: トランザクションは、バックエンドの DTP プログラムで、ISC 機能を実行します。
N: トランザクションは、バックエンドの DTP プログラムではありません。アウトバウンド ISC 機能は実行するが、遠隔 Sun MTP または CICS 領域によって開始された場合、想定されたようには実行されません。
5. 別のトランザクションを追加するには、Return キーを押して適切な値を入力します。
 6. 入力後、Enter キーを押してエントリを挿入し、PCT メイン画面に戻ります。
 7. PF2 キーを押して、PCT を保存します。
 8. PF3 キーを押して、「Standard Tables」メニューに戻ります。

FCT での VSAM ファイルの定義

FCT には、アプリケーションの VSAM ファイルに関する情報が含まれます。各ファイルには、アプリケーションプログラムが発行したコマンドを、Sun MTP が検査して実行する際に使用する一連の特性が関連付けられています。FCT のエントリごとに、VSAM カタログにエントリを作成し、領域がこのファイルにアクセスできるようにする必要があります。

注 – 使用しているアプリケーションが、VSAM ファイルを使用しないで RDBMS のみを使用する場合、FCT には何も追加する必要はありません。

▼ ファイルを定義する

1. Table Manager を開始し、「Standard Tables」メニューを表示します。
2. PF5 キーを押して、FCT を表示します。

Dataset	Filename	Environment	Access Method	File Type	No Rcv	Rcd Fmt	Group	Dup Alwd	Read Only	Dfr Opn
ACCTFIL	ACCTFILE	KIXSYS	VSAM	KSDS	N	F		Y	N	N
ACCTIX	ACIXFILE	KIXSYS	VSAM	KSDS	N	F		Y	N	N
ALT1	ALT1	KIXSYS	VSAM	KSDS	N	F		Y	N	N
ALT2	ALT2	KIXSYS	VSAM	KSDS	N	F		Y	N	N
ALT3	ALT3	KIXSYS	VSAM	KSDS	N	F		Y	N	N
DFHUSD	DFHUSD	KIXSYS	VSAM	KSDS	N	V rdo		N	N	N
TEMPSTG	TEMPSTG	KIXSYS	VSAM	KSDS	C	V unikix		Y	N	N
TEMPSTGR	TEMPSTGR	KIXSYS	VSAM	KSDS	N	V unikix		Y	N	N

PF2=Write to Disk PF5=Delete Entry PF9=Remote File PF12=Export Table
PF3=Previous Menu PF7=Previous Page PF10=Search ENTR=Modify
PF4=Insert Entry PF8=Next Page PF11=Import Table

図 5-5 ファイル管理テーブル (FCT)

3. PF4 キーを押して、挿入画面を表示します。
4. 次のフィールドに値を入力します。
 - a. 「Dataset」フィールドに、データセットの名前を入力します。

これは、プログラムがファイルを参照するために使用する名前です。単一の物理的な VSAM ファイルに、複数の論理的なデータセットを関連付けられます (ファイルは環境によって制限される)。これらのデータセット名は、「別名」と呼ばれます。
 - b. 「Filename」フィールドに、データセットまたは代替索引に関連付けられた物理ファイル名を入力します。

ファイル名は最大で 10 文字です。環境で制限される単一のファイルには、8 つまでの論理データセット名を関連付けられます。
- c. ファイルが置かれているディレクトリを指す環境変数の名前を入力します。

注 - 「Filename」フィールドが空白である場合、データセットは別名としてカウントされません。

- d. 「Access Method」フィールドに VSAM と入力します。
その他のアクセス方式は、現在サポートされていません。
- e. 「File Type」フィールドに、次の VSAM ファイルタイプのいずれかを入力します。
KSDS: キーシーケンスデータセットまたは代替索引データセット
ESDS: 入力順データセット
RRDS: 相対レコードデータセット
- f. 「No Rcv」フィールドに、次のいずれかの値を入力します。
Y: 回復を無効化します。
N: 回復を有効化します。
C: 回復を無効にし、VSAM ファイルのシステムキャッシュの使用を有効にします。このオプションは、回復可能でない一時記憶域ファイルに対するデフォルトです。
このフィールドは、特定のファイルについて、回復がアクティブかどうかを示します。回復が有効な場合、このフィールドは、VCT の回復オプションの設定に優先されます。VCT で回復が無効の場合、このフィールドは無視されます。

注 - 複数のデータセットエントリが同じ物理ファイルを使用する場合、Table Manager は、このファイルのすべてのデータセットが、「No Rcv」フィールドで同じ値が指定されていることを確認し、一貫性がない場合はエラーをレポートします。

- g. 「Rcd Fmt」フィールドには、ファイルに可変長レコードが含まれる場合は Y、固定長レコードが含まれる場合は F を入力します。
この値は、一部の CICS コマンドの妥当性を検査する際に使用します。
- h. ファイルがあるグループに属する場合、「Group」フィールドにグループ名を入力します。
このグループは、GCT で定義されている必要があります。
- i. 「Dup Alwd」フィールドには、代替索引に重複キーが認められている場合は Y、そうでない場合は N を入力します。
- j. 「Read Only」フィールドには、データセットが読み取り専用の場合は Y、読み取りおよび書込みが可能な場合は N を入力します。
- k. 「Dfr Open」フィールドには、起動時にデータセットを開かない場合に Y を入力します。
ファイルは、何かほかのイベントやプログラムが開始されるまで、開かれません。N を入力すると、データセットは領域の起動時に開かれます。
5. 別のファイルを追加するには、Return キーを押して適切な値を入力します。

6. 入力後、Enter キーを押してエントリを挿入し、FCT のメイン画面に戻ります。
7. 定義したファイルの一部が、遠隔システム上に置かれている場合、FCT の「Remote File Characteristics」画面から、追加のファイル特性を入力する必要があります。
FCT のメイン画面でファイルを選択して PF9 キーを押し、「Remote File Characteristics」画面を表示します。遠隔ファイルの定義については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。
8. データをすべて入力し終えてから、PF2 キーを押して FCT を保存します。
9. PF3 キーを押して、「Standard Tables」メニューに戻ります。

PPT でのリソースの定義

PPT は、ご使用のアプリケーション環境で使用できるオンラインプログラム、テーブル、マップセットを一覧します。

▼ オンラインプログラムとマップを定義する

1. Table Manager を開始し、「Standard Tables」メニューを表示します。
2. PF7 キーを押して、PPT を開きます。

Program	Typ	SysID	RmtProg	RmtTrn	Group	Shared Library	P/L API
ACCT00	K					acct00	F
ACCT01	K					acct01	F
ACCT02	K					acct02	F
ACCT03	K					acct03	F
ACCTSET	M						F
ACCTSETM	M						F
CEDA	S				rdo		
CEDADIS	M				rdo		
BATCH000	B				unikix		F
CEBR	S				unikix		F
CEBRMAP	M				unikix		F
CECI	K				unikix	lib/ceci	F
CEDF	S				unikix		F
CEMT	S				unikix		F
CEMTMAP	M				unikix		F
CESF	S				unikix		F

PF2=Write to Disk PF5=Delete Entry PF9=JAVA Details PF12=Export Table
 PF3=Previous Menu PF7=Previous Page PF10=Search ENTR=Modify
 PF4=Insert Entry PF8=Next Page PF11=Import Table

図 5-6 処理プログラムテーブル (PPT)

3. PF4 キーを押して、PPT 挿入画面を表示します。
4. 次のフィールドに適切な値を入力します。
 - a. 「Program」フィールドに、プログラム、テーブル、またはマップセットの名前を入力します。

名前は一意である必要があります。マップセットの名前は 7 文字まで、プログラムとテーブルの名前は 8 文字までです。
 - b. 「Typ」フィールドは、リソースタイプを指定します。有効な値は次のとおりです。
 - A: CICS の LOAD と RELEASE コマンドで参照するテーブル
 - B: バッチプロセッサを指定する Sun MTP だけのタイプ
 - C: COBOL プログラム
 - J: Java プログラム
 - K: C プログラム
 - L: PL/I プログラム
 - M: マップセット
 - S: スタンドアロンユーティリティーを指定する Sun MTP だけのタイプ
 - c. このプログラム、マップセット、またはテーブルが遠隔システムで実行される場合、「SysID」フィールドに 4 文字のシステム ID を指定します。

この ID は、TCT のシステム項目テーブルの SysID エントリと一致する必要があります。遠隔プログラムには、遠隔システムが動作している場合だけアクセスできます。
 - d. ローカルプログラム名と遠隔プログラム名が異なる場合は、「RmtProg」フィールドに遠隔システムのプログラム名を入力します。

デフォルトでは、ローカル名および遠隔名は同じです。
 - e. 「RmtTran」フィールドに、遠隔システムで使用するトランザクションコードを入力します。

このプログラムが遠隔である場合、このフィールドで指定されたトランザクションコードは、分散プログラムリンク (DPL) 機能に使用されます。セキュリティまたはアカウントの目的で、名前を変更できます。
 - f. プログラムまたはマップセットがあるグループに属する場合、「Group」フィールドにグループ名を入力します。

このグループは、GCT で定義されている必要があります。
 - g. プログラムが共有ライブラリ (共有オブジェクト) の場合、その共有ライブラリの名前を入力します。

名前には、16 文字まで使用できます。環境変数 KIXLIB が設定されていない限り、すべてのエントリは \$KIXSYS からの相対パスです。

- h. トランザクション処理プログラムの起動時に共有オブジェクトをロードさせる場合は、「P/L」フィールドを Y に設定します。そうでない場合は、N に設定するか、空白のままにします。

Sun MTP で PL/I と C/C++ の共有オブジェクトをロードする方法については、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。
- i. 「API」フィールドに次の値のいずれかを入力します。このフィールドは、DPL 機能のローカルでのテストを定義します。

F: すべての CICS コマンドが、サーバープログラムで使用可能です (デフォルト)。

D: サーバープログラムは、DPL に制限された API を実行できません。
5. 別のリソースを追加するには、Return キーを押して手順 4 を再度実行します。
6. 入力後、Enter キーを押してエントリを挿入し、PPT メイン画面に戻ります。
7. Java プログラムを定義する場合は、Java プログラムを選択し、PF9 キーを押して Java クラスの詳細画面を表示します。そうでない場合は、手順 8 に進みます。

Java プログラムの使用については、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。
8. PF2 キーを押して、PPT を保存します。
9. PF3 キーを 2 回押して、「Table Manager」メインメニューに戻ります。

SIT のカスタマイズ

SIT には、領域の初期化に使用する情報が含まれます。この情報には、データベース接続データ、パスワード管理、日付形式、および BMS キーマッピングがあります。SIT に設定した値は、同じ領域で別のアプリケーションを実行中の場合でも、領域全体に適用されます。

▼ SIT をカスタマイズする

1. Table Manager を開始し、「Standard Tables」メニューを表示します。
2. PF1 キーを押して、SIT を表示します。

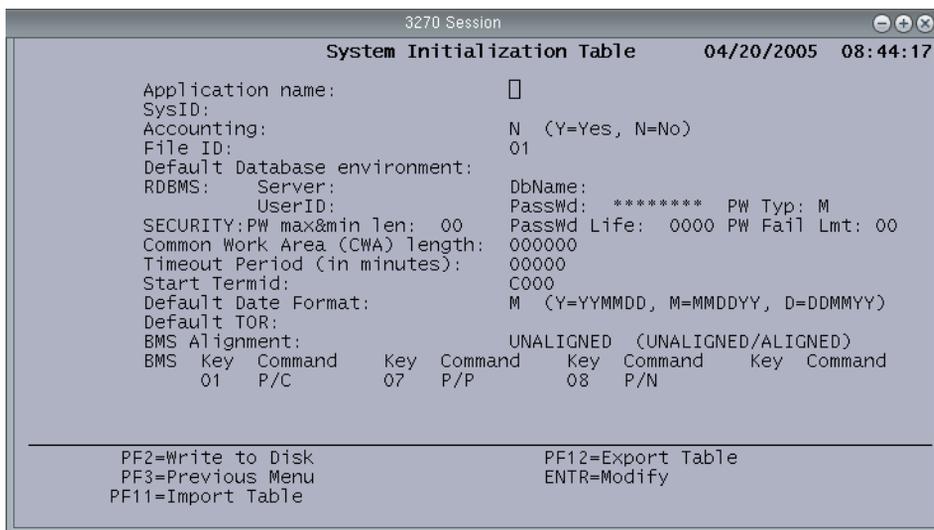


図 5-7 システム初期化テーブル (SIT)

3. 「Application name」フィールドに、8 文字までのアプリケーション名を入力します。

これは、ASSIGN と INQUIRE SYSTEM コマンドの APPLID オプションで返される値です。ファイル名の妥当性の検査は、このフィールドについて実行されます。

このフィールドは、トランザクション経路指定に必要です。トランザクション経路指定の詳細は、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

4. 「SysID」フィールドに、システム識別子を入力します。

このオプションの 4 文字フィールドは、ASSIGN コマンドの SYSID オプションで返される値です。

5. 「Accounting」フィールドは、領域全体のアカウントिंगを制御します。次のいずれかを入力します。

Y: アカウントिंगが有効になり、ユーザー ID とトランザクション ID に基づく指定が有効になります。「File ID」フィールドも指定する必要があります。アカウントिंगについての詳細は、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

N: このほかの設定に関係なく、アカウントングレコードは作成されません。これがデフォルトです。

6. 「File ID」フィールドに、ジャーナルファイルの識別子を入力します。

JCT に「File ID」が指定されていない場合は、JRNL と File ID を結合して作成されます。たとえば、JRNL05 という名前のジャーナルファイルを作成します。

7. 「Default Database environment」フィールドに、VSAM カタログを含むディレクトリを指す環境変数を入力します。

これは、通常 \$KIXSYS です。

8. 「RDBMS」フィールドの、「Server」、「UserID」、「DbName」、および「PassWd」は、リレーショナルデータベースの Sun MTP ユーザー出口プログラムで返されます。

ユーザー出口プログラムは、データベースへの接続の初期設定時に呼び出すことができます。

ご使用のデータベースに適切なフィールド値については、151 ページの「SIT での RDBMS 値の指定」を参照してください。

注 - パスワードは、テーブルをディスクに保存するときに暗号化されます。

9. 領域全体のセキュリティーに関連するフィールドは、次のとおりです。

注 - 外部セキュリティーマネージャー (ESM) が有効になっている場合、これらのフィールドは無視されます。

- a. 「PW Typ」フィールドに、現在のパスワードを、EXEC CICS CHANGE PASSWORD コマンド、または CESN トランザクションを使用して変更する場合、有効になるパスワードのタイプを指定する 1 文字を入力します。

有効な値は次のとおりです。

N: 新しいパスワードは、数値でなければなりません (0 ~ 9)

A: 新しいパスワードは、アルファベットでなければなりません (A ~ Z, a ~ z)

M: 新しいパスワードは、数字、文字、またその他の文字を組み合わせることが可能です。これがデフォルトです。

- b. 「PW max&min len」フィールドに、パスワードの長さの最大と最小を指定する 2 桁の値を入力します。

形式は、MN です。この M と N はそれぞれ、1 ~ 8 文字で作成されるパスワードの最大長と最小長を示す 1 桁の数値です。

ゼロに設定すると、パスワード長の確認は行われません。EXEC CICS CHANGE PASSWORD コマンドまたは CESN トランザクションのどちらかでパスワードが変更される場合、1 ~ 8 文字のパスワードであれば、すべて認められます。

- c. 新しく変更されたパスワードが有効な日数を定義する値を、「PassWd Life」フィールドに入力します。

このフィールドは、ある領域のすべてのユーザー ID に適用されます。有効なエントリーは 0000 ~ 9999 です。

ゼロに設定すると、パスワードの有効期限はチェックされません。

- d. 「PW Fail Lmt」フィールドに、パスワードの認証に続けて何回失敗すると、システムがそのユーザーのパスワードを自動的に使用停止にするかを指定します。有効なエントリーは 00 ～ 99 です。

ゼロに設定すると、パスワード認証エラーの回数の確認は行われません。これがデフォルトです。ユーザー ID ごとの失敗回数は、共有メモリーでだけ保持され、領域を再起動すると消去されます。

あるユーザー ID がパスワードに正しくアクセスすると、そのユーザー ID のパスワードの失敗回数は即座にゼロにリセットされます。

10. 「Common Work Area (CWA) length」フィールドに、システムの初期設定時に CWA に割り当てられる記憶領域の容量を指定します。

これは、0 ～ 32767 の整数でなければなりません。

11. 「Timeout Period」フィールドに、アクティブではない端末のタイムアウト値を入力します。

領域は、接続方法に関係なく、この時間 (2 分間隔) を過ぎてもアクティブではない端末を自動的にログアウトします。デフォルトはゼロで、アクティブでない端末を切り離しません。

12. 「Start Termid」フィールドに、その領域の起動端末識別子を入力します。

これは、1 つの文字と 3 桁の数字でなければなりません。たとえば、C000 です。領域は、この識別子を使用してメインフレーム端末識別子と区別します。

領域は、ユーザーがクライアントを起動するときの所定のユーザーセッションに端末識別子を割り当てます。実際の端末識別子は、クライアントが領域に接続された順番に基づいて割り当てられます。

13. 「Default Date Format」フィールドに、FORMATTIME コマンドの、DATE と DATEFORM オプションで返される日付の形式を指定します。

有効な値は次のとおりです。

Y: YYMMDD または YYYYMMDD という形式の、年、月、日付

M: MMDDYY または MMDDYYYY という形式の、月、日付、年

D: DDMMYY または DDMMYYYY という形式の、日付、月、年

14. 「Default TOR」フィールドに、ローカルシステムの端末所有領域 (TOR) を入力します。

このフィールドは、ユーザー出口が見つからない端末に関連します。ユーザー出口の詳細は、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

15. 「BMS Alignment」フィールドに、マップの位置合わせを行うかどうかを指定します。

この領域のすべてのマップは、同じオプションである必要があります。SIT をエクスポートすると、このフィールドは、位置合わせなしであれば U、位置合わせする場合は A がエクスポートされます。

UNALIGNED: 記号マップと物理マップは、ハーフワードによる位置合わせを行いません。これがデフォルトです。BMS コピーメンバーを Sun MTP でリコンパイルする場合、このオプションを指定する必要があります。

ALIGNED: 記号マップと物理マップは、ハーフワード境界で位置合わせを行います。BMS コピーメンバーが、元の IBM ハーフワード位置合わせコピーメンバーである場合、このオプションを指定して BMS マップが正しく表示されるようにする必要があります。

16. 「BMS (paging keys)」フィールドには、次のように CSPG トランザクションを制御するキーを 8 個まで定義できます。

ファンクション キー	有効なキー値
PF1 ~ PF24	01 ~ 24
PA1 ~ PA3	P1 ~ P3

各キーは BMS ページコマンドに対応します。BMS コマンドを呼び出すには、CSPG トランザクションの実行中に定義されたキーを押します。

17. すべて入力し終えてから、PF2 キーを押して SIT を保存します。
18. PF3 キーを 2 回押して、「Table Manager」メインメニューに戻ります。

サポートされる BMS ページングコマンドのその他の情報については、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

第6章

回復の構成

Sun MTP では、領域全体で回復を実行できるほか、ファイル単位での回復を実行できます。

この章の内容は、次のとおりです。

- 75 ページの「VCT の回復の構成」
- 76 ページの「VSAM データセットの回復の構成」

VCT の回復の構成

この節では、VCT で回復を構成する方法について説明します。

注 – XA リソースマネージャーを使用して領域を構成している場合は、VCT で回復を構成する必要があります。2 フェーズコミットの使用の詳細については、『Sun Mainframe Transaction Processing ソフトウェア XA リソースマネージャーの使用』を参照してください。

▼ 回復を構成する

1. Table Manager を開始し、「Standard Tables」メニューを表示します。
2. PF10 キーを押して、VCT を表示します。
最初の3つのフィールドは、回復に関するものです。

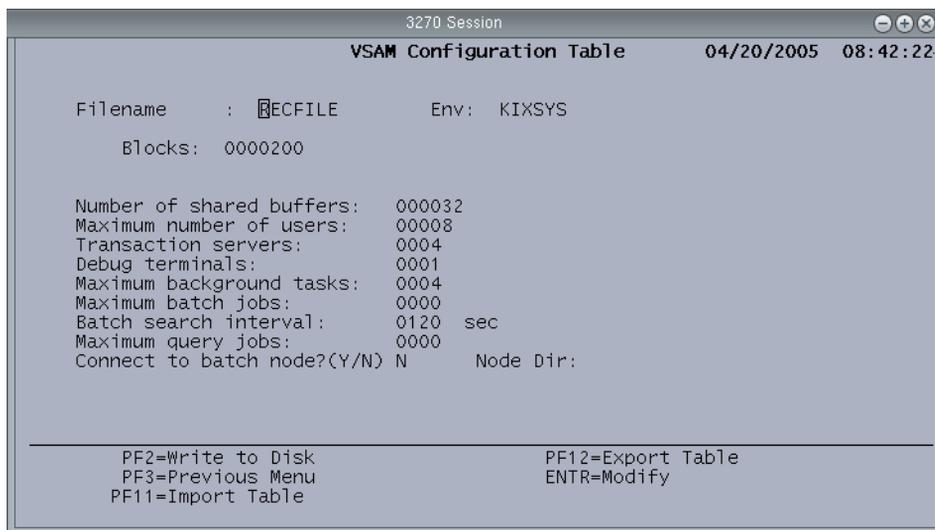


図 6-1 VSAM 構成テーブル (VCT)

3. 次の回復情報を入力します。

Filename: 回復データファイルの名前を入力します。

Env: 回復データファイルが置かれているディレクトリを指す環境変数を入力します。

Blocks: 回復ファイルのブロックの数を入力します。この値は、100 以上でなければなりません。そうでないと、領域の起動時に回復ファイルが作成されません。

4. PF2 キーを押して、VCT を保存します。

5. PF3 キーを押して、Table Manager の「Standard Tables」メニューに戻ります。

6. 変更を有効にするため、領域を再起動する必要があります。

VSAM データセットの回復の構成

FCT のファイルのエントリに属性を設定すれば、アプリケーションの KSDS および RRDS VSAM ファイルに回復を選択的に構成できます。ESDS VSAM ファイルに回復を構成するには、78 ページの「ESDS データセットの回復の構成」を参照してください。

VSAM ファイルの管理については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

▼ KSDS および RRDS ファイルに回復を指定する

1. Table Manager を開始し、「Standard Tables」メニューを表示します。
2. PF5 キーを押して、FCT を表示します。

Dataset	Filename	Environment	Access Method	File Type	No Rcv	Rcd Fmt	Dup Group	Read Alwd	Dfr Only	Opn
ACCTFIL	ACCTFILE	KIXSYS	VSAM	KSDS	N	F	Y	N	N	N
ACCTIX	ACIXFILE	KIXSYS	VSAM	KSDS	N	F	Y	N	N	N
ALT1	ALT1	KIXSYS	VSAM	KSDS	N	F	Y	N	N	N
ALT2	ALT2	KIXSYS	VSAM	KSDS	N	F	Y	N	N	N
ALT3	ALT3	KIXSYS	VSAM	KSDS	N	F	Y	N	N	N
DFHUSD	DFHUSD	KIXSYS	VSAM	KSDS	N	V rdo	N	N	N	N
TEMPSTG	TEMPSTG	KIXSYS	VSAM	KSDS	C	V unikix	Y	N	N	N
TEMPSTGR	TEMPSTGR	KIXSYS	VSAM	KSDS	N	V unikix	Y	N	N	N

PF2=Write to Disk PF5=Delete Entry PF9=Remote File PF12=Export Table
PF3=Previous Menu PF7=Previous Page PF10=Search ENTR=Modify
PF4=Insert Entry PF8=Next Page PF11=Import Table

図 6-2 ファイル管理テーブル (FCT)

3. 新しいエントリを作成する場合、64 ページの「FCT での VSAM ファイルの定義」の手順に従います。既存のエントリを修正する場合、カーソルをそのデータセット名に合わせます。
4. Tab キーを使用して、カーソルを「No Rcv」フィールドに移動し、N を入力します。
VCT で回復が有効な場合、そのファイルに対して回復が有効です。
5. Enter キーを押して、変更を適用します。
6. 別のファイルを修正するには、手順 3 ~ 5 を繰り返します。
7. すべて入力し終えてから、PF2 キーを押して FCT を保存します。
8. PF3 キーを押して、「Standard Tables」メニューに戻ります。

ESDS データセットの回復の構成

デフォルトの VSAM 回復機能は、KSDS および RRDS データセットにのみ適用できます。ESDS データセットは、入力順です。動的なトランザクションバックアウト段階ではバックアウトできません。

ESDS 回復機能を使用すれば、ESDS データセットのレコードを特別なタグでマークしておいて、あとの段階でさらに処理できます。これらのタグは、レコードが論理的に削除されているものとして扱えることを示します。ただし、レコードにタグを付けても、そのレコードは物理的または論理的に削除されていることを正確に示すわけではありません。Sun MTP は、VSAM 操作時およびエクスポートまたはインポート (unikixbld) 操作時に、それらを有効なレコードとして処理します。

ESDS データセットに回復を指定する必要がある場合は、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

代替リソース定義

代替リソース定義は、IBM のリソース定義オンライン (RDO) と同じです。これにより、リソースを定義して、領域の実行中にそれらを利用できるようになります。

Sun MTP では、通常 Table Manager を使用して領域にリソースを追加します。このとき、グループ名で各リソースを識別します。グループ管理テーブル (GCT) を使用して、領域を開始するときに含めるグループ (アプリケーションに関連する複数のリソース) を指定します。

リソースを定義する別の方法として、CEDA トランザクション、および dfhusdup ユーティリティーを使用する方法もあります。CEDA と dfhusdup ユーティリティーを使用して定義されたリソースは、リソース定義ファイル (DFHUSD) に保存されません。

この章では、代替リソース定義を使用する方法について説明します。この章の内容は、次のとおりです。

- 79 ページの「リソースタイプとコマンド」
- 81 ページの「代替リソース定義への領域の有効化」
- 82 ページの「リソースの構成」
- 82 ページの「領域の起動時のリソースの有効化」
- 83 ページの「リソースの管理」
- 83 ページの「代替リソース定義を使用する長所と短所」

リソースタイプとコマンド

次のリソースタイプがサポートされています。

- マップセット
- プログラム
- トランザクション

サポートするコマンド

サポートするコマンドは、IBM の RDO にあるコマンドのサブセットです。CEDA コマンドは、オンラインのリソース定義に使用します。また、dfhusdup は同様の機能を提供するバッチユーティリティです。

表 7-1 CEDA コマンド

コマンド	説明
DEFINE	リソースを定義します。
DELETE	リソースを削除します。
DISPLAY	グループと各グループ内のリソースを表示します。
INSTALL	領域を実行中に、その領域のリソース定義を追加または置換します。

表 7-2 dfhusdup ユーティリティコマンド

コマンド	説明
INITIALIZE	リソース定義ファイルを初期化します。CEDA トランザクションまたは dfhusdup コマンドのスクリプトを実行する前に、INITIALIZE コマンドを実行して DFHUSD ファイルを初期化する必要があります。
DEFINE	リソースを定義します。
DELETE	リソースを削除します。
LIST	グループ、そのグループ内のリソース、およびそのリソースの属性をレポートします。
COPY	あるグループから別のグループにリソースをコピーします。

注 – Sun MTP は、LIST をサポートしていません。これは、IBM の RDO でグループ名を順番に並べたセットです。

CEDA コマンドとオプションの説明については『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』、dfhusdup ユーティリティオプションの説明については『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

代替リソース定義への領域の有効化

DFHUSD ファイルが VSAM カタログと FCT で定義されており、rdo グループが GCT で定義されている場合、領域で代替リソース定義が有効になります。DFHUSD ファイルは Sun MTP をインストールしたときに定義されるため、必要なのは rdo グループを構成することだけです。

領域の起動時に、\$KIXSYS ディレクトリに rdo.dir サブディレクトリが作成されます。このディレクトリのテーブルには、リソース定義ファイルと CEDA トランザクションを有効にするのに必要なすべてのエントリが含まれています。これらのリソースが領域で利用できるようにするには、\$KIXSYS/rdo.dir ディレクトリを指す rdo グループを GCT で定義する必要があります。

▼ rdo グループを定義する

1. Table Manager を開始し、PF5 「Extended Tables」 を押します。
2. PF7 キーを押して、GCT を表示します。
3. PF4 キーを押して、新しいエントリを挿入します。
4. 「Group Name」 フィールドに、rdo を入力します。
5. 「R/O」 フィールドに、N を入力します。
6. 「Directory」 フィールドに、\$KIXSYS/rdo.dir を入力します。
7. Enter キーを押してエントリを挿入し、GCT 画面に戻ります。
8. PF2 キーを押して、テーブルをディスクに保存します。
9. PF3 キーを 2 回押して、「Table Manager」メインメニューに戻ります。
10. Table Manager を終了します。
11. 変更を有効にするため、領域を再起動します。

リソースの構成

領域の代替リソース定義を有効にすると、オンラインの CEDA トランザクションまたは dfhusdup バッチユーティリティーを使用して、マップセット、プログラム、およびトランザクションにリソースを定義できます。

リソースを定義してから、それらをインストールする必要があります。

領域の起動時のリソースの有効化

DFHUSD リソース定義ファイルに定義されているリソースは、その領域で使用できます。ただし、これらのリソースは、領域の起動時に自動的に有効にならないので、そのままでは使用できません。CEDA INSTALL コマンドを使用して、リソースを動的にインストールする必要があります。Sun MTP には、あらかじめ定義されているリソースを領域の起動時に有効化してインストールする方法が用意されています。

▼ 起動時にリソースを有効化する

1. 自動的にインストールするグループをリストしたテキストファイルを作成します。グループ名は、スペース、または改行で区切ります。
2. `-G filename` オプションで領域を起動します。ここで、*filename* はテキストファイルの名前です。次に例を示します。

```
$ unikixmain -G resfile . . .
```

リソースの管理

次の unikixmain チューニングオプションは、動的にロードできるリソースの数に制限を設定する場合や、マップセットを使用するときの共有メモリーの動作を設定する場合に便利です。

表 7-3 リソースをロードする際の unikixmain オプション

オプション	説明
-a <i>pnmn</i>	領域に動的に追加できるエントリの数を指定します。
-a <i>tnnn</i>	<ul style="list-style-type: none">• <i>p</i> 引数は、プログラムとマップセットのリソースを指定します。<i>nmn</i> は、領域の実行中に追加できるエントリの数を指定します。デフォルト値は 200 です。• <i>t</i> 引数は、トランザクションのリソースを指定します。<i>nmn</i> は、領域の実行中に追加できるエントリの数を指定します。デフォルト値は 100 です。
-Z	使用期間中だけマップセットを共有メモリーに保持します。マップセットの使用後、メモリーは解放されます。

代替リソース定義を使用する長所と短所

Sun MTP ソフトウェアはこれまで、Table Manager から定義したリソースだけをサポートしていました。現在では、マップセット、プログラムおよびトランザクションについて代替リソース定義を使用できます。代替リソース定義を使用する長所は、これらのリソースで変更を有効にする場合、領域を停止してから再起動する必要がなく、動的に変更できることです。短所は、一時、リソースが DFHUSD ファイルと Sun MTP テーブルに分かれて保存されている混合環境で作業をする点です。

次の表は、従来のテーブルベースの環境とテーブルと代替リソース定義の両方を使用する環境との間に見られる、リソースを定義する方法、およびロードする方法の相違点を示します。

表 7-4 従来の環境と混合環境の相違点

機能/環境	従来の Sun MTP	混合環境
リソースの定義	Table Manager を使用して、すべてのリソースをテーブルに定義します。	プログラム、マップセット、およびトランザクションを CEDA と dfhusdup ユーティリティを使用して定義するか、テーブルに定義します。これらのリソースを CEDA または dfhusdup を使用して定義する場合、その定義は DFHUSD ファイルに保存されます。Table Manager を使用する場合、この定義は \$KIXSYS または GCT で指定したディレクトリのテーブルに保存されます。Table Manager を使用して、その他すべてのリソースをテーブルに定義します。
リソースのロード	<ol style="list-style-type: none"> 1. \$KIXSYS ディレクトリのテーブルからのエントリ 2. \$KIXSYS/unikix.dir ディレクトリのテーブルからのシステムエントリ 3. GCT で指定したディレクトリのテーブルからのエントリ 	<ol style="list-style-type: none"> 1. \$KIXSYS ディレクトリのテーブルからのエントリ 2. \$KIXSYS/unikix.dir ディレクトリのテーブルからのシステムエントリ 3. \$KIXSYS/rdo.dir ディレクトリを含む、GCT で指定したディレクトリのテーブルからのエントリ 4. unikixmain に -G オプションを付けて指定した、DFHUSD ファイルのグループ
重複リソースの処理	重複リソース定義が見つかったと、最初の定義が有効になり、2 番目の定義は破棄されます。	Table Manager を使用して定義したすべてのリソースは、標準の順番でロードされます。重複エントリが見つかったと、最初の定義が有効になり、2 番目の定義は破棄されます。DFHUSD ファイルからロードされた定義がテーブルで定義されているリソースと重複する場合、そのテーブル定義は無効になります。たとえば、テーブルと起動時にロードされる DFHUSD ファイルにマップセットが定義されている場合、DFHUSD ファイルの定義が、テーブルの定義よりも優先されます。

第8章

UNIX 端末の構成

この章では、Sun MTP に UNIX 端末を構成して使用方法について説明します。次のトピックについて説明します。

- 85 ページの「キーファイル」
- 92 ページの「Sun MTP X クライアントの定義」
- 93 ページの「ASCII 端末の初期化」

IBM 3270 キーボードには、端末を制御するキーや、トランザクションプログラムの実行を制御する CICS アプリケーションに、アテンション識別子 (AID) を生成するキーがあります。ほとんどの端末には、一連の複数の文字を送信するファンクションキーがあります。

UNIX 端末を 3270 デバイスのように動作させるいくつかの方法があります。

- Sun MTP キーファイルを使用して、端末の一連の文字を 3270 制御機能と CICS AID に変換します。これにより、3270 データストリームを使用する背後のアプリケーションは正しく動作できます。
- 端末やキーボードタイプごとの X リソース定義を含むシェルスクリプトを使用します。

キーファイル

Sun MTP は、異なる端末タイプごとにキーファイルを用意して、標準端末をサポートしています。

パーソナルコンピュータで端末エミュレータソフトウェアを使用する場合、設定メニューから VT エミュレーションの 1 つを選択すると、Sun MTP は正しく実行できます。

注 - 端末エミュレーションプログラムを使用する場合は、エミュレータが生成するキーシーケンスが、そのエミュレータの Sun MTP キーファイルのキーシーケンスと同じであることを確認します。キーを再定義する必要がある場合、ほとんどのエミュレーションパッケージでは、設定メニューから実行できます。詳細については、エミュレーションパッケージのマニュアルを参照してください。

端末タイプごとに、デフォルトのキーボードファイルがあります。このファイル名は、TERM 環境変数で使用する名前に接尾辞 `.key` を付けたものです。一部の端末は、キー数やレイアウトが異なる、オプションのキーボードをサポートしています。これらのタイプの端末には、`$UNIKIX/lib` ディレクトリ内の接尾辞 `.key` を持つファイルに、追加のキーボードレイアウトが定義されています。

Sun MTP は、自動的にデフォルトのキーファイルを探してロードします。端末がデフォルトのキーボードを使用しない場合は、ローカルクライアントを起動する際、`unikix` コマンドに `-k` オプションを付けて明示的にキーボードタイプを定義する必要があります。X 端末クライアントは、通常 `xterms` に設定された TERM 環境変数で Sun MTP を実行します。

次のコマンドは、NCD 97 キーキーボードに合わせたローカルクライアントを起動します。

```
$ unikix -k ncd97.key
```

注 - キーファイルは、UNIX 端末だけに使用されます。3270 SNA デバイスのサポートには使用されません。

ユーザーの利便性を考えて、Sun MTP では、数値入力キーの一部のキーをコントロールキーとして使用します。可能であれば、数値入力キーの数値、加算、減算、および小数点キーを、キーボード定義に残しておいてください。端末のファンクションキーが足りない場合は、手で入力した複数キーのシーケンスを定義できます。たとえば、バックタブ機能を 2 文字のシーケンス、Escape と Tab で定義できます。キーファイルの定義を変更し、名前を変えてサイト固有のオプションを含むディレクトリ構造である `$UNIKIX/local` に保存すれば、特定の要件に合わせて、供給された定義の変更や新しい端末タイプの追加が可能です。

kixkeydef を使用したキーファイルの作成または編集

使用しているキーボードの定義を含むキーファイルを作成または変更するには、`kixkeydef` ユーティリティを使用します。キーごとに、5 つまでの定義を用意できます。ただし、定義や編集を実行できるのは、`kixkeydef` 画面にある最初の 2 つ

だけです。1つのキーに3つ以上の定義がある場合、そのキー定義の横にアスタリスクが表示されます。これらの定義の変更や追加は、キーファイル自体を編集する必要があります。方法については、89ページの「キーファイルの直接編集」を参照してください。

▼ kixkeydef を使用してキーファイルを作成または編集する

1. 領域が実行中でないことを確認します。
2. シェルプロンプトから kixkeydef ユーティリティを実行し、kixkeydef 画面を表示します。

現在のディレクトリに myterm.key という名前でキーファイルを作成するには、次のように入力します。

```
$ kixkeydef myterm.key
```

既存のファイルを編集する場合は、定義が表示されます。新しいファイルを作成する場合は、図 8-1 の画面と同様の空白の画面が表示されます。キーファイルの名前を入力した場合は、ファイル名が画面の右下隅に表示されます。

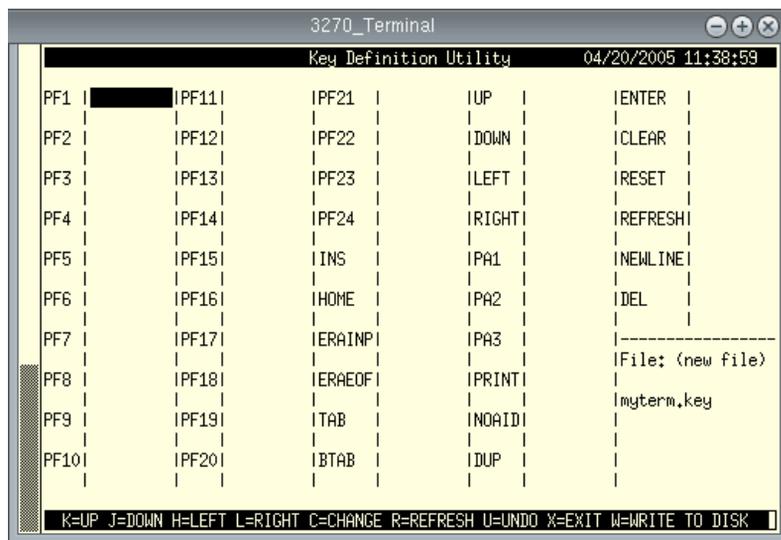


図 8-1 kixkeydef 画面

3. 定義するキーにカーソルを移動します。
次のキーを使用して、kixkeydef 画面のカーソルを移動します。
 - h: 1 フィールド左に移動
 - j: 1 行下に移動
 - k: 1 行上に移動
 - l: 1 フィールド右に移動
4. c を押して、変更を指定します。
5. 定義するキーを押します。
6. Return キーを押して、変更を終了します。
7. 定義するキーごとに、手順 3 ~ 6 を繰り返します。
8. すべてを終えてから、w を押して新しいキーファイルをディスクに書き込みます。
9. x を押して、kixkeydef ユーティリティーを終了します。
10. 新しいキーファイルを、\$UNIKIX/lib または \$UNIKIX/local/lib: に移動します。

```
$ mv myterm.key $UNIKIX/lib
```

11. -k オプションを使用してキーファイルを指定し、ローカルクライアントを再起動します。たとえば次のようになります。

```
$ unikix -k myterm.key
```

▼ Return キー、改行キー、およびバックスペースキーを定義する

1. kixkeydef 画面で、定義するキーの場所にカーソルを移動します。
2. Return キーを定義するには、c を押し、\r を入力してから Return キーを押します。
3. 次に定義するキーの場所にカーソルを移動します。
4. 改行キーを定義するには、c を押し、\n を入力してから Return キーを押します。
5. 次に定義するキーの場所にカーソルを移動します。
6. バックスペースキーを定義するには、c を押し、\b を入力してから Return キーを押します。
7. 変更を保存して、kixkeydef を終了するには、w を押してから x を押します。キーファイルを変更しないで終了する場合は、x を押します。

編集中に定義を変更する場合は、そのフィールドにカーソルを移動し、新しい定義を入力します。元に戻す (u) または再表示 (x) 機能を使用して定義の編集もできます。

kixkeydef ユーティリティを使用してコメントを含むキーファイルを修正する場合、そのコメントはファイルを保存すると失われます。

コード例 8-1 は、kixkeydef ユーティリティを使用して作成されたキーファイルを示します。

コード例 8-1 kixkeydef ユーティリティで作成されたキーファイル

```
.
#
# Value:  ESC OPHex:1b 4f 50Octal: 33 117 120
PF1=\EOP
#
# Value:  ESC OQHex:1b 4f 51Octal: 33 117 121
PF2=\EOQ
#
# Value:  ESC ORHex:1b 4f 52Octal: 33 117 122
PF3=\EOR
#
# Value:  ESC OSHex:1b 4f 53Octal: 33 117 123
PA1=\EOS
#
# Value:  ESC [MHex:1b 5b 4dOctal: 33 133 115
PA2=\EOQ
#
.
```

キーファイルの直接編集

あるキーに 3 つ以上の定義を作成する場合または以前に作成済みの 3 つ以上の定義を変更する必要がある場合は、kixkeydef ユーティリティを使用する代わりに、キーファイルを直接編集する必要があります。たとえば、COBOL デバッガをサポートする新しいキーファイルを定義する場合、4 つのカーソルまたは矢印キーが定義されていることを確認する必要があります。

キーファイルの各エントリの形式は、次の例のとおりです。

```
UP=\E[A
```

定義は、列 1 から始まります。このエントリは、「1 つ上にカーソルを移動する」というキーシーケンスを定義します。\`\E` は、ESC 文字 (16 進数では 0x1b) を表します。この例で UP 機能を定義する文字のシーケンスは、ESC[A です。次の表に示す特殊文字のシーケンスを定義に使用できます。

表 8-1 キーボード定義のエスケープシーケンス

文字	定義
<code>\E</code>	エスケープ (0x1b)
<code>\b</code>	バックスペース (0x8)
<code>\t</code>	タブ (0x9)
<code>\n</code>	復帰改行 (0xa)
<code>\r</code>	キャリッジリターン (0xd)
<code>\nnn</code>	8 進数値 (000 ~ 377)

1 つのキーに複数の値を割り当てられます。たとえば、LEFT キーに `\b` と `\E[D` の両方を定義できます。

行の最初の列にポンド記号 (#) を付ければ、コメント行をファイルに追加できます。

端末のキー変換を含むデフォルトのファイル名は、TERM 環境変数の端末の名前に、.key 接尾辞を付けたものになります。キーファイルは、\$UNIKIX/lib または \$UNIKIX/local/lib のディレクトリに置く必要があります。

\$UNIKIX/local/lib ディレクトリが最初に検索され、次に \$UNIKIX/lib が検索されます。別の方法として、unikix コマンドの -k パラメータを使用して、そのキーファイルの相対パス名または絶対パス名を指定する方法もあります。

次の表は、NCD 101 キーキーボードを使用する TERM=xterms のキーボード割り当てを示します。付録 A では、その他にサポートされているキーボードの割り当てを示します。

表 8-2 TERM=xterms のキーボード割り当て

3270 機能	文字	キーボードの説明
UP	<code>\E[A</code>	上矢印
DOWN	<code>\E[B</code>	下矢印
LEFT	<code>\E[D</code>	左矢印
LEFT	<code>\b</code>	バックスペース
RIGHT	<code>\E[C</code>	右矢印
TAB	<code>\t</code>	タブ
BTAB	<code>\E\t</code>	シフト - タブ

表 8-2 TERM=xterms のキーボード割り当て (続き)

3270 機能	文字	キーボードの説明
ERAEOF	\E[6~	Page Down
ERAINP	\E[3~	Page Up
INSMODE	\E[1~	挿入
DEL	\E[4~	削除
ENTER	\EOM	Enter
CLEAR	\EOk	キーパッドの加算 (+)
RESET	\Eom	キーパッドの減算 (-)
HOME	\E[2~	Home
REFRESH	\O14	Ctrl-l (小文字 L)
PF1-PF12	\E[11~ to \E[24~	PF1 ~ PF12
PA1	\E[OP	キーパッドの除算 (/)
PA2	\E[OQ	キーパッドの乗算 (*)
PA3	\E[OT	改行
PRINT	\E[5~	終了

次の表は、3270 キーボードの機能を定義します。

表 8-3 3270 キーボード機能

3270 キー	目的	AID
UP	カーソルを 1 つ上の位置に移動	
DOWN	カーソルを 1 つ下の位置に移動	
LEFT	カーソルを 1 つ左の位置に移動	
RIGHT	カーソルを 1 つ右の位置に移動	
TAB	次の保護されていないフィールドにタブ移動	
BTAB	前の保護されていないフィールドにタブ移動	
NEWLINE	カーソルを次の行の列 1 に移動	
HOME	カーソルを行 1 の列 1 に移動	
ERAEOF	フィールドの終わりまで消去	
ERAINP	保護されていない入力を消去 (すべてのフィールド)	

表 8-3 3270 キーボード機能 (続き)

3270 キー	目的	AID
INSMODE	文字挿入モードを切り替え	
NOAID	操作なし	あり
DEL	文字を削除	
ENTER	メッセージを送信	あり
CLEAR	画面を消去	あり
RESET	キーボードのロック状態をリセット	
PRINT	画面を印刷	
REFRESH	画面を再表示	
PF1-PF24	1 ~ 24 のプログラムファンクションキー	あり
PA1-PA3	1 ~ 3 のプログラムアテンションキー	あり

Sun MTP X クライアントの定義

xterm 端末エミュレータの起動時に、keySYM 変換のための X リソース定義は、クライアントとして動作する xterm VT100 端末エミュレータに、アプリケーションキーパッドに置かれたキーを生成するための正しい文字シーケンスを伝えます。これは、xterm -xrm パラメータ文字列を使用して実行します。

次に NCD 101 キーボードに使用される X リソースのサンプル値を示します。

```
'XTerm*vt100*translations:#override\
<Key>Insert:      string(0x1b)      string("[1~")\n\
<Key>Home:        string(0x1b)      string("[2~")\n\
<Key>Prior:       string(0x1b)      string("[3~")\n\
<Key>Delete:      string(0x1b)      string("[4~")\n\
<Key>End:         string(0x1b)      string("[5~")\n\
<Key>Next:        string(0x1b)      string("[6~")\n\
<Key>KP_Divide:   string(0x1b)      string("[OP")\n\
<Key>KP_Multiply: string(0x1b)      string("[OQ")\n\
<Key>KP_Subtract: string(0x1b)      string("[Om")\n\
<Key>KP_Add:      string(0x1b)      string("[Ok")\n\
<Key>KP_Enter:   string(0x1b)      string("[OM")\n\
<Key>Linefeed:   string(0x1b)      string("[OT")\n\
Shift<Key>Tab:    string(0x1b)      string(0x09) '
```

<Key> のあとに続く名前は、X keysym 値です。これは、キーの説明を直接示していないことがあります。xev と呼ばれる X ユーティリティがあり、これはあるキーによって生成される keysym を決定できます。

サポートする X サーバーとキーボードの、各組み合わせごとのリソース定義を含むシェルスクリプトは、\$UNIX/bin ディレクトリにあります。これらのシェルスクリプトを使用して、新しいウィンドウで各ユーザークライアントを開始します。次に例を示します。

```
$ xtermunikix unikix
```

または

```
$ ncd101unikix unikix
```

接尾辞の start (たとえば、ncd101start や xtermstart) で終わる名前を持つシェルスクリプトは、別ウィンドウのプログラム開発環境を提供します。start シェルスクリプトは、unikix で終わる名前を持つシェルスクリプトを呼び出します。詳細は、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

別のレベルの構成も、X サーバーで使用できます。これは、キーボードによって生成されたスキャンコードを X keysym に割り当てる機能です。通常、このレベルの調整は必須ではありません。詳細は、xmodmap(1) のマニュアルページを参照してください。

ASCII 端末の初期化

ASCII 端末は、/etc/inittab ファイルに定義します。このファイルは、init プロセスで読み込まれて処理されます。次に、/etc/inittab のサンプルエントリを示します。

```
t0:2:respawn:/etc/getty ttyml du_1200 none LDISCO
```

フィールド none は、端末タイプを含むことができますが、getty で使用可能な端末タイプのリストには制限があります。端末を初期化するには、通常は .profile または .login ファイルにある、tput init コマンドを使用します。init プロセスの詳細については、inittab(4) のマニュアルページを参照してください。

第9章

プリンタの構成

この章では、プリンタの構成方法について説明します。この章の内容は、次のとおりです。

- 95 ページの「デフォルトのプリンタの構成」
- 97 ページの「追加プリンタの構成」
- 97 ページの「一時データキューを使用するプリンタの構成」
- 100 ページの「LU タイプ 1 プリンタの構成」
- 102 ページの「プリンタサイズの指定」
- 102 ページの「kixprint によるプリンタの構成」
- 103 ページの「kixprint シェルスクリプトの変更」

プリンタはさまざまな機能に関連付けられます。たとえば、次の EXEC CICS コマンドを使用して印刷を実行できます。

- `ISSUE PRINT`
- `DUMP`
- `SEND MAP PRINT`
- `SEND MAP`: トランザクションがプリンタで開始されたとき
- `SEND`: トランザクションがプリンタで開始されたとき
- `WRITEQ TD: PRINT` として定義された一時データキューに書き込むとき

デフォルトのプリンタの構成

デフォルトのプリンタは、`CPRT` です。ただし、端末管理テーブル (TCT) に定義された IBM 3270 ではないプリンタの任意の 4 文字の ID も指定できます。

▼ デフォルトのプリンタを構成する

1. 空白のトランザクション画面に CTBL を入力し、Table Manager に直接アクセスします。
2. PF4 キーを押して、「Table Manager」メニューで「Standard Tables」を選択します。
3. PF8 キーを押して、TCT を選択します。
4. PF8 キーを押します。

TCT のプリンタ画面に、CICS コマンドで使用できる定義済みの端末識別子が表示されます。

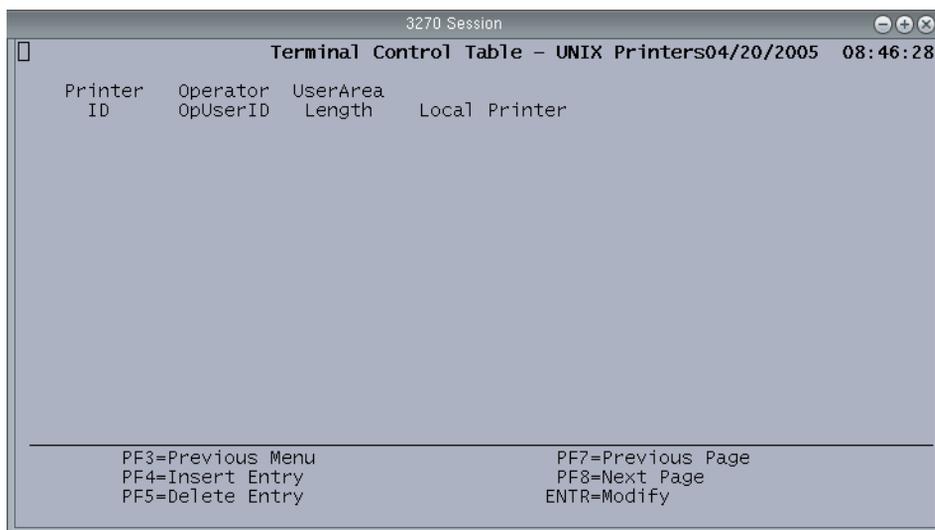


図 9-1 TCT - Printers 画面

5. 新しいプリンタの記述を入力するには、PF4 キーを押します。
6. 挿入画面が表示されるので、次の必須フィールドに値を入力します。
 - a. 「Printer ID」フィールドに、デフォルトのプリンタを特定するために CPRT を入力します。
 - b. 「Local Printer」フィールドに、14 文字までの名前を入力します。これは、-p パラメータで kixprint に渡されます。ほかのフィールドはオプションです。
7. Enter キーを押し、エントリを挿入して確認します。
8. PF3 キーを押して、TCT メインメニューに戻ります。

9. PF2 キーを押して、変更をディスクに書き込みます。
10. PF3 キーを押して、TCT を終了します。

注 – デフォルトのプリンタを使用できるようにするには、領域を再起動する必要があります。

追加プリンタの構成

追加プリンタを構成するには、95 ページの「デフォルトのプリンタの構成」の手順に従って、識別子 CPRT の代わりに一意の 4 文字のプリンタ ID を使用します。

一時データキューを使用するプリンタの構成

プリンタを一時データキュー (TDQ) に関連付けるには、宛先管理テーブル (DCT) と TCT の両方に定義する必要があります。

注 – WRITEQ TD コマンドを使用して 1 つまたは複数のレコードをプリンタに出力するときは、印刷されるレコードの幅が DCT の「Length」フィールドによって定義されます。

▼ DCT に TDQ をプリンタとして定義する

1. 空白のトランザクション画面に CTBL を入力し、Table Manager を開きます。
2. PF4 キーを押して、「Table Manager」メニューで「Standard Tables」を選択します。
3. PF4 キーをもう一度押して DCT を表示します。
4. 「Destination Control Table」メニューで、PF9 キーを押して「Extrapartition Destinations」画面を表示します。
5. PF4 キーを押して、挿入画面を表示します。

6. プリンタごとに、次のフィールドに値を入力します。
- 「Dest-ID」フィールドに、一時データキューの 4 文字の名前を入力します。
これは、WRITEQ TD コマンドで使用するキューの名前です。
 - 「Format」フィールドに、PRINT を入力します。
 - 「Length」フィールドに、キューの最大レコードサイズとする数値を入力します。
 - 「Open」フィールドに Y を入力します。
 - 「IOSt」フィールドに、0 を入力します。
- ほかのフィールドはオプションです。

3270 Session							
Destination Control Table						05/09/2005 09:33:14	
Extrapartition Destinations							
Dest-ID	Filename	Environment	Group	Format	Length	Open	IOSt
TDQB				PRINT	256	Y	0

PF3=Previous Menu
ENTR=Insert

図 9-2 DCT での一時データキュー (TDQ) の定義

- Enter キーを押し、変更を検査します。
- PF3 キーを押し、DCT メインメニューに戻ります。
- PF2 キーを押し、変更をディスクに書き込みます。

▼ プリンタと TDQ を関連付ける

1. DCT メインメニューで PF3 キーを押し、Table Manager の「Standard Tables」メニューに戻ります。○
2. PF8 キーを押し、「Terminal Control Table」を選択します。
3. PF8 キーを押し、「Terminal Control Table - Printers」画面を表示します。
4. PF4 キーを押して、「TCT - Printers Insert」画面を表示します。詳細は、図 9-3 を参照してください。
5. 次のフィールドに入力します。
 - a. 「Printer ID」フィールドに、DCT の「Dest-ID」フィールドと同じ 4 文字の値を入力します。
 - b. 「Local Printer」フィールドに、名前 (14 文字まで) を入力します。これは、-p パラメータで kixprint に渡されます。ほかのフィールドはオプションです。

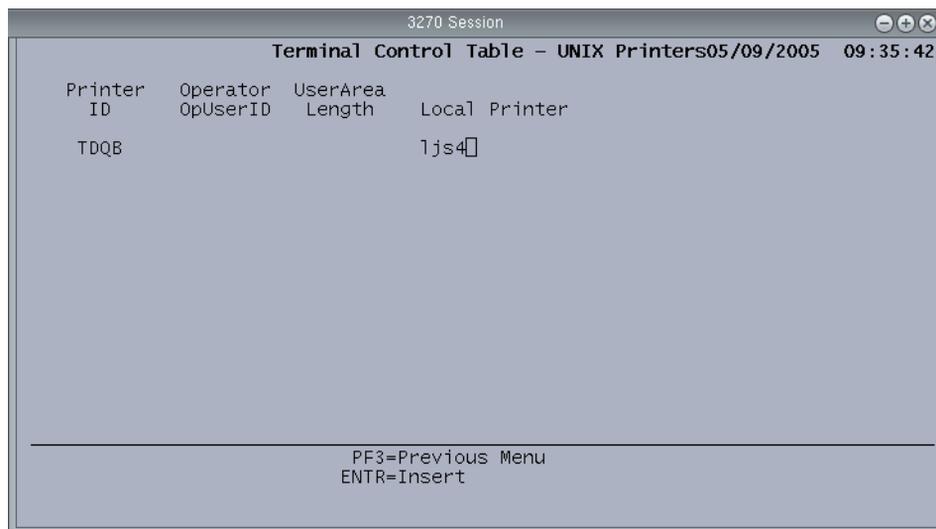


図 9-3 TDQ とプリンタとの関連付け

6. Enter キーを押し、エントリを挿入して確認します。
7. 「TCT - Printers」画面に、新しいエントリが表示されます。○

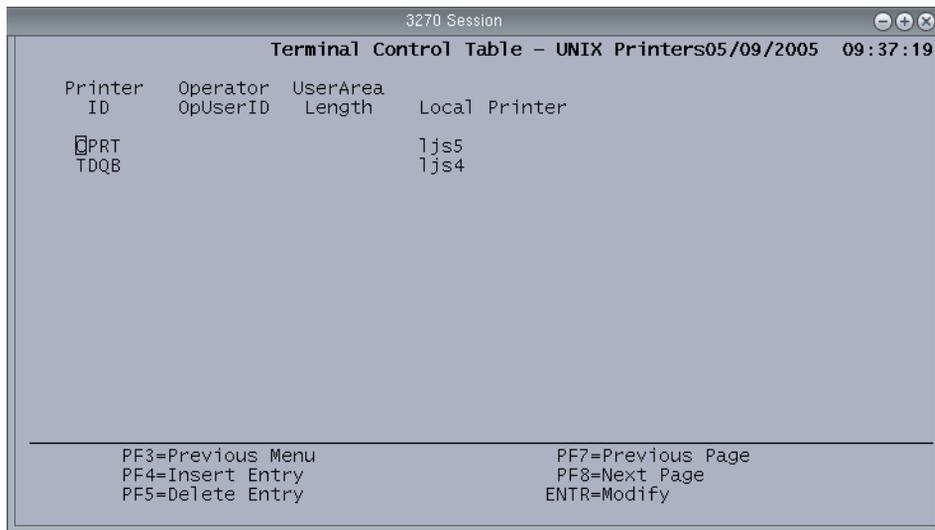


図 9-4 TCT - Printers 画面 - 新規エントリの表示

8. PF3 キーを押して、TCT メインメニューに戻ります。
9. PF2 キーを押して、変更を保存します。
10. 開発システムメインメニューまたは CTBL の終了画面が表示されるまで、PF3 キーを押します。
11. 構成を有効にするには、領域を停止して再起動する必要があります。
Sun MTP の起動と停止については、第 3 章を参照してください。

LU タイプ 1 プリンタの構成

この節では、LU タイプ 1 プリンタの構成方法について説明します。Sun MTP 環境では、領域と LU タイプ 1 プリンタとの通信は TN3270 サーバーによって管理されます。

LU タイプ 1 プリンタは、EXEC CICS SEND コマンドで送信されるデータをサポートしています。EXEC CICS SEND MAP や EXEC CICS SEND TEXT などのほかの SEND コマンドを使用すると、出力が判読不能になることがあります。

▼ LU タイプ 1 プリントを構成する

1. TN3270 サーバーを構成します。
詳細は、113 ページの「TCP サーバー (unikixtcp) を構成する」を参照してください。
2. ローカルクライアントまたは TN3270 クライアントから空白のトランザクション画面で CTBL を入力し、「Standard Tables」メニューを表示して PF4 キーを押します。
3. PF8 キーを押して TCT を表示し、PF9 キーを押して「3270 Devices」画面を表示します。○
4. 「3270 Devices」画面で PF4 キーを押し、図 9-5 に示す挿入画面を表示します。

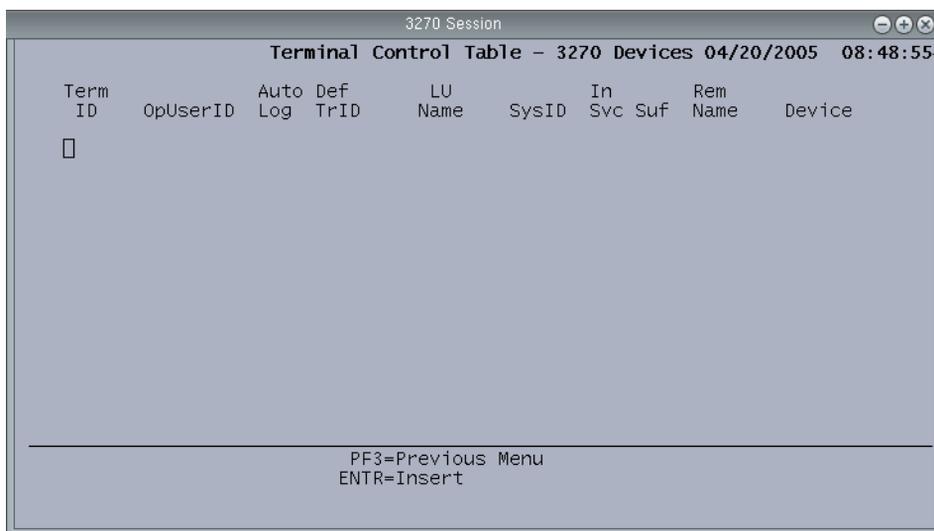


図 9-5 TCT - 3270 Devices 挿入画面

5. 定義するプリンタごとに、次のフィールドに値を入力します。
 - a. 「Term ID」フィールドで、プリンタの 4 文字の識別子を入力します。
これは、アプリケーションで認識されているプリンタ名です。これは一意の名前でなければなりません。
 - b. 「LU Name」フィールドに、論理ユニット名を入力します。
 - c. 「Device」フィールドに、値 SCSPRINT を入力します。

6. Enter キーを押し、エントリを挿入して確認します。
「Auto Log」フィールドと「In Svc」フィールドには、自動的に Y(es) が入力されます。
7. PF3 キーを押して、TCT メニューに戻ります。
8. PF2 キーを入力してディスクへの変更内容を保存します。
9. PF3 キーを押して、TCT を終了します。
10. Table Manager を終了し、領域を再起動して、変更を有効にします。

プリンタサイズの指定

デフォルトのプリンタサイズは、132 列 × 66 行ですが、環境変数を設定すれば、別の数の行と列を指定できます。この環境変数名は、TCT の 4 文字の端末定義に、4 文字の ROWS または COLS を続けたものです。

たとえば、CPRT が、80 列で 24 行のプリンタの場合、次のように、Bourne または Korn シェル設定ファイルに環境変数を定義してエクスポートします。

```
CPRTCOLS=80
CPRTROWS=24
export CPRTCOLS CPRTROWS
```

kixprint によるプリンタの構成

kixprint シェルスクリプトは、Sun MTP の主要印刷機能を実行します。トランザクションの実行時、ディスクに印刷出力が収集されます。SYNCPOINT コマンドによって明示的にまたはトランザクションの終了によって暗黙的に同期点が発行されると、すべてのスプールされている出力が印刷されます。

通常、kixprint は、Sun MTP プリントサーバーから呼び出されます。Sun MTP で実行されるすべての実行可能ファイルのように、kixprint は、\$UNIX/local/bin または \$UNIX/bin あるいは unikixmain の -l オプションで指定されたディレクトリから実行されます。kixprint スクリプトは、\$KIXSYS に設定されている現在のディレクトリで実行されます。

kixprint シェルスクリプトで、デフォルトの印刷コマンドは、`lp -s` です。これは、デフォルトの Sun MTP プリンタに関連付けられています。SPOOLER 環境変数を設定すれば、このデフォルトを変更できます。プリンタ ID が与えられている場合、AT&T のスプーラであれば `-d printerid` をコマンドに追加し、BSD のスプーラであれば `-P printerid` をコマンドに追加します。

kixprint の出力を破棄する場合は、Sun MTP を起動する前に、SPOOLER 環境変数を有効な印刷コマンドではない値に設定します。

kixprint シェルスクリプトを変更して、プリンタ ID に基づいて特定の処理を実行できます。103 ページの「kixprint シェルスクリプトの変更」を参照してください。

kixprint シェルスクリプトの変更

DCT と TCT でプリンタを構成してから kixprint を変更すれば、サイト固有の手順を実行できます。変更は、シェルスクリプトの `special handling if printer id given` のセクションに入力します。

コード例 9-1 は、このセクションを変更していない状態を示します。

コード例 9-1 変更していない kixprint シェルスクリプト (2 / 1)

```
# special handling if printer id given
if test          -n "$PRINTID"
then
    # if UCB line printer spooler, add "-P" switch;
    # if ATT line printer spooler, add "-d" switch;
    # if NT print.exe command add a /d:PrinterName switch;
    # all other spoolers should accept -p
    case "$SPOOLER" in
        lpr*)
            SPOOLCMD="$SPOOLER -P $PRINTID"
            ;;
        lp*)
            : 'check for existence'
            if test          "`lpstat -v|fgrep "$PRINTID"``"
            then
                : 'printid OK'
                SPOOLCMD="$SPOOLER -d$PRINTID"
```

コード例 9-1 変更していない kixprint シェルスクリプト (2 / 2)

```
        else
            : 'printid not OK'
            SPOOLCMD="$SPOOLER"
        fi
        ;;
    *print.exe*)
        SPOOLCMD="$SPOOLER \\/d:$PRINTID"
        ;;
    *)
        SPOOLCMD="$SPOOLER -p $PRINTID"
        ;;
    esac
fi
```

このソフトウェアに含まれるバージョンの kixprint には、プリンタ ID のデフォルト処理だけがあります。プリンタ ID がこのスクリプトに渡される場合、\$SPOOLER の値がチェックされます。\$SPOOLCMD には、\$SPOOLER の値に基づいて、プリンタ ID を処理する値が割り当てられます。使用するプリンタ ID の処理をカスタマイズするには、\$SPOOLER case 文の前に新しい case 文を追加します。

例

ご使用のネットワークに、3 台の物理的なプリンタと 1 台の論理的なプリンタがある場合、それぞれフォーマット要件は異なります。各プリンタは次のとおりです。

- **Printer1:** PostScript データのみを受け入れる PostScript™ レーザープリンタです。
- **Printer2:** 標準の 132 列のラインプリンタです。
- **Printer3:** 標準の 80 列のラインプリンタです。
- **Printer4:** プリンタに印刷するのではなく、印刷ファイルを作成します。

すべてのプリンタを適応させるには、kixprint シェルスクリプトに次の case 文を挿入します。

コード例 9-2 変更された kixprint シェルスクリプト

```
case "$PRINTID" in
Printer1)      # Postscript printer
               # Set FORMATTER to the ASCII to Postscript
               # conversion routine.
               FORMATTER="lwf"
               # Set PRINTID to the name of the dedicated
               # Postscript printer.
               PRINTID=ps
               ;;
Printer2)      # 132 line printer
               # Set FORMATTER to 132 columns wide and 60 lines per page
               FORMATTER="pr -t -w132 -l60"
               # Set PRINTID to the name of the wide printer
               PRINTID=wp
               ;;
Printer3 )     # 80 column printer
               # Set FORMATTER to 72 columns wide and 66 lines per
               # page (This is the default setting for pr.)
               FORMATTER="pr -t"
               # Set PRINTID to the name of the narrow printer
               PRINTID=np
               ;;
Printer4)      # Output to print file
               # Set SPOOLER to cat. This can be used to send the
               # output to a file.
               SPOOLER=cat
               # Clear PRINTID
               PRINTID=
               ;;
esac
```

Printer4 では、SPOOLER が cat に設定されています。この場合は、SPOOLER case 文で処理されないので、別の case を追加する必要があります。次を最後の case の前の SPOOLER case 文、つまり *) に追加します。

```
cat)
SPOOLCMD="$SPOOLER > $$$.prt"
;;
```


第10章

システム間通信 (ISC) の構成

Sun MTP は、SNA と TCP/IP によるシステム間通信 (ISC) をサポートしています。メインサーバープロセスである unikixmain は、ご使用の ISC 構成に適切な数の通信サーバープロセスを起動します。

- SNA の場合、unikixdcl プロセスは、Solaris オペレーティングシステムの SNAP-IX をサポートします。
- TCP/IP の場合、Sun MTP は unikixtcp プロセスを使用します。

この章の内容は、次のとおりです。

- 107 ページの「サポートされる ISC 機能」
- 108 ページの「SNA の領域の構成」
- 113 ページの「TCP/IP の領域の構成」
- 116 ページの「トランザクションセキュリティー」
- 116 ページの「遠隔領域の自動インストール」

サポートされる ISC 機能

次の表は、サポートされる ISC 機能を示します。

表 10-1 Sun MTP でサポートされる ISC 機能

SNA	TCP/IP
トランザクション経路指定	トランザクション経路指定
機能シップ	機能シップ
分散プログラムリンク (DPL)	分散プログラムリンク (DPL)
非同期処理	非同期処理
分散トランザクション処理 (DTP)	n/a

トランザクション経路指定、機能シップ、非同期処理、および DPL は、アプリケーションプログラマに意識させないようにすることができます。アプリケーションは、構成定義の制御に基づいて、遠隔トランザクション、遠隔プログラム、および遠隔リソースのすべてにアクセスできます。

ISC 機能は、双方向モデルをサポートします。このモデルでは、追加や再配置された CICS アプリケーションとリソースを、既存の CICS や Sun MTP 環境に提供するか、その逆を実行するホストシステムとして Sun MTP が機能します。

これらの機能の設定方法については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

SNA の領域の構成

注 - Sun MTP は、同期レベル 2 のセッションをサポートしていません。このため、機能シップは同期レベル 1 のミラー CVMI と CPMI を使用します。これらのミラートランザクションは、遠隔 CICS 領域で定義する必要があります。

SNA によるシステム間通信をサポートする領域を構成するには、次を実行する必要があります。

1. SNA サーバーソフトウェアを構成します。
SNA のマニュアルを参照してください。
2. 通信マネージャーを設定します。
詳細は、108 ページの「通信マネージャーの設定」を参照してください。
3. 領域の設定ファイルに必須環境変数を設定します。
詳細は、第 2 章を参照してください。
4. Terminal Control Table - System Entries テーブルにシステム情報を入力します。
詳細は、110 ページの「遠隔システムの定義」を参照してください。

通信マネージャーの設定

通信マネージャー (unikixCommMgr プロセス) は、unikixmain によって自動的に起動されます。起動時、unikixCommMgr は \$KIXSYS/unikixrc.cfg という名前の構成ファイルを読み込みます。このファイルには、unikixCommMgr が起動する必要がある通信サーバーのタイプに関する情報と、これらサーバーを実行する上での制約が含まれています。各領域に、それぞれの unikixrc.cfg ファイルが必要です。unikixdcl プロセスは、unikixCommMgr によって起動されます。

▼ unikixdcl を開始するように unikixCommMgr を構成する

1. 領域設定ファイルを探して、UNIKIX と KIXSYS 環境変数が設定されていることを確認します。
2. unikixrc ファイルを、\$UNIKIX/lib ディレクトリから構成対象の領域の \$KIXSYS ディレクトリにコピーし、unikixrc.cfg という名前に変更します。

注 - unikixrc.cfg ファイルがすでに \$KIXSYS ディレクトリにある場合は、その既存のファイルを開いて適切なエントリを編集できます。

3. unikixrc.cfg ファイルへの書き込み権があることを確認します。
4. ファイルを開き、DclServer* エントリを編集します。

表 10-2 では、各エントリについて説明します。

unikixrc.cfg ファイルを変更する際、次の構文ルールが適用されます。

- 感嘆符 (!) はコメントを示します。感嘆符のあとの行にあるテキストは無視されません。
- キーワードと引用符に囲まれない値には、大文字と小文字の組み合わせを使用できます。
- パス名やファイル名などの引用符に囲まれた値は、大文字と小文字が区別されません。

次の例は、DclServer* エントリを示します。

!Keywords	Values	Possible Values
DclServer*Active:	True	!True/False
DclServer*Processes:	1	! 1 - 255
DclServer*LocalLu:	"MTP02"	!Any Legal LU Name
DclServer*Debug:	False	!True/False
DclServer*MultipleLUs:	True	!True/False

5. 作業が終了したら、ファイルを保存します。
6. 110 ページの「遠隔システムの定義」の手順に従って、構成プロセスを続けます。

注 - \$KIXSYS/unikixrc.cfg ファイルを変更するたびに、領域を再起動して変更を有効にする必要があります。

次の表は、unikixrc.cfg ファイルで指定できる項目を示します。値を指定しない場合は、デフォルト値が使用されます。

表 10-2 unikixrc.cfg ファイル - DCL サーバーのエントリ

キーワード	説明
DclServer*Active	True の場合、unikixdcl サーバーを起動します。False の場合、そのあとのすべての DclServer キーワードが無視されます。
DclServer*Processes	この Sun MTP 領域に呼び出す、1 ~ 255 の unikixdcl サーバードプロセスの数。デフォルトは 1 です。
DclServer*LocalLu	この Sun MTP 領域の SNA LU 名。この値は、DCL ローカル LU の別名に一致します。
DclServer*Debug	Sun MTP トレース情報をファイルに保存します。値は、True または False (デフォルト) です。ご購入先からこの機能を有効にするように指示されない限り、False に設定してください。
DclServer*MultipleLUs	このエントリのデフォルト値は、False です。このエントリをデフォルトのままにするか、False に設定すると、アウトバウンドを割り当てるローカル LU は、unikixrc.cfg ファイルから取得されます。このエントリを True に設定すると、アウトバウンドを割り当てるローカル LU は、「SysID」フィールドを見て特定されたシステムのテーブルから取得されます。

遠隔システムの定義

領域を構成する前に、SNA サーバードソフトウェアを設定する必要があります。領域の構成パラメータのいくつかは、SNA サーバードの構成値と一致する必要があります。

複数の Sun MTP 領域が、SNA サーバードを介して複数の遠隔システムと同時に通信できます。ただし、1 つの Sun MTP 領域が通信できるのは、1 つの SNA サーバードだけです。

▼ 遠隔システムを定義する

1. DCLRTERM と DCLSTERM 環境変数をまだ設定していない場合、これらを設定します。この環境変数は、インバウンドとアウトバウンドの並行要求の最大数を定義します。

インバウンド要求とアウトバウンド要求の合計数は、200 を超えることはできません。これらの変数については、第 2 章を参照してください。

2. 領域を開始して、クライアント接続を確立します。

3. 空白のトランザクション画面に CTBL を入力し、「Table Manager」メニューを表示します。
4. PF4 キーを押して、「Standard Tables」を選択します。
5. PF8 キーを押して、「Terminal Control Table」を選択します。
6. 「Terminal Control Table」メニューが表示されるので、PF10 – 「System Entries」を押します。
7. 「System Entries」画面で、PF4 キーを押して挿入画面を表示します。

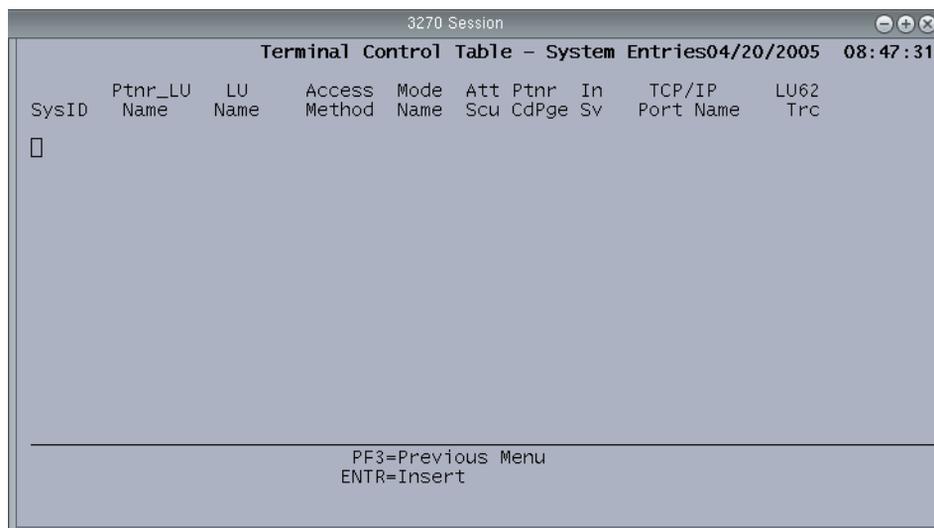


図 10-1 TCT - System Entries 挿入画面

8. 挿入画面で、ご使用の環境に必要な情報を入力します。
 - a. 「SysID」フィールドに、EXEC CICS コマンドの SYSID オプションで使用する、またはそのリソースが遠隔であることを定義するために Sun MTP テーブルで使用する遠隔システムのローカル識別子を入力します。
 - b. 「Ptnr_LU Name」フィールドは空白のままにします。
 - c. 「LU Name」フィールドに、この領域に関連付けられたローカル LU を特定するために使用する名前を入力します。
 - d. 「Access Method」フィールドに DCL を入力します。
 - e. 「Mode Name」フィールドに、遠隔システムに接続する際に使用するモードグループの名前を入力します。
この値は、接続に使用する SNA モードと一致します。

- f. 「Att Scu」フィールドに次の値のいずれかを入力し、遠隔システムのセキュリティレベルを指定します。
- Local の場合は L、N、または空白
 - Identify の場合は、I
- 詳細は、116 ページの「トランザクションセキュリティ」を参照してください。
- g. 「Ptnr CdPge」フィールドに、遠隔システムで使用するコードページを入力します。
- 遠隔システムが、EBCDIC システムの場合、このフィールドに値を指定します (00037 など)。この値は、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』の説明にある、トランザクション経路指定ユーザー出口ルーチンに渡されます。
 - 遠隔システムが ASCII システムの場合、このフィールドは空白のままにします。
 - DBCS 環境で TN3270 端末を使用して遠隔システムにアクセスしている場合は、このフィールドに値を指定する必要があります。
- h. 「In Svc」フィールドは、この接続で、最初からサービスを提供するかないかを定義します。Y または N を入力します。
- サービスを提供しない場合、インバウンドとアウトバウンドの両方の ISC 要求は、この接続からは送信できません。CEMT トランザクションを使用して、領域の実行中に接続の状態を変更します。
- i. 「TCP/IP Port Name」フィールドは空白のままにします。
- j. 「LU62 Trc」フィールドは、その接続に LU6.2 トレースを実行するかどうかを指定します。Y または N を入力します。
- トレースがオンの場合、その接続を使用するトランザクション処理プログラム (unikixtran) についてトレースが開始されます。トレースファイルは、isclu62_pid という名前で \$KIXSYS ディレクトリに書き込まれます。この pid は、トレースファイルに書き込むプロセスのプロセス ID です。
9. Enter キーを押して、新しいエントリを挿入します。
- 「System Entries」画面に新しいエントリが表示されます。
10. PF9 キーを押します。
11. 次の画面が表示されます。
- a. 「Nq_Ptnr_Name」フィールドに、遠隔 Sun MTP または CICS 領域のネットワーク修飾パートナー LU 名を入力します。
- この値は、Sun MTP が通信する領域の完全修飾パートナー LU 名と一致します。
- b. 「Host Name」フィールドは空白のままにします。
- c. データ入力を終了してから、Enter キーを押します。

12. PF3 キーを押して、「System Entries」画面に戻ります。
13. PF3 キーを押して、TCT メニュー画面に戻ります。
14. PF2 キーを押して、変更をディスクに書き込みます。
15. PF3 キーを 2 回押して、「Table Manager」メインメニューに戻ります。
16. 領域を停止して再起動します。
詳細は、第 3 章を参照してください。

TCP/IP の領域の構成

Sun MTP TCP サーバー (unikixtcp) には、TCP/IP プロトコルを使用して別の領域と通信する機能があります。TCP サーバーに対して、各領域を構成します。

▼ TCP サーバー (unikixtcp) を構成する

1. TCP サーバーの unikixtcp が待機するポート番号を決めます。
ポートは、番号または名前で指定できます。
TCP/IP 接続を必要とする同じマシン上で複数の Sun MTP 領域を実行する場合、そのホスト上の各 unikixtcp サーバーに一意のポートを指定する必要があります。
2. TCPRTERM と TCPSTERM 環境変数をまだ設定していない場合、これらを設定します。この環境変数は、インバウンドとアウトバウンドの並行要求の最大数を定義します。
インバウンド要求とアウトバウンド要求の合計数は、200 を超えることはできません。これらの変数については、第 2 章を参照してください。
3. 領域を開始して、クライアント接続を確立します。
4. 空白のトランザクション画面に CTBL を入力し、「Table Manager」メニューを表示します。
5. PF4 キーを押して、「Standard Tables」を選択します。
6. PF8 キーを押して、「Terminal Control Table」を選択します。
7. 「Terminal Control Table」メニューが表示されるので、PF10 – 「System Entries」を押します。
8. 「System Entries」画面で、PF4 キーを押して挿入画面を表示します。

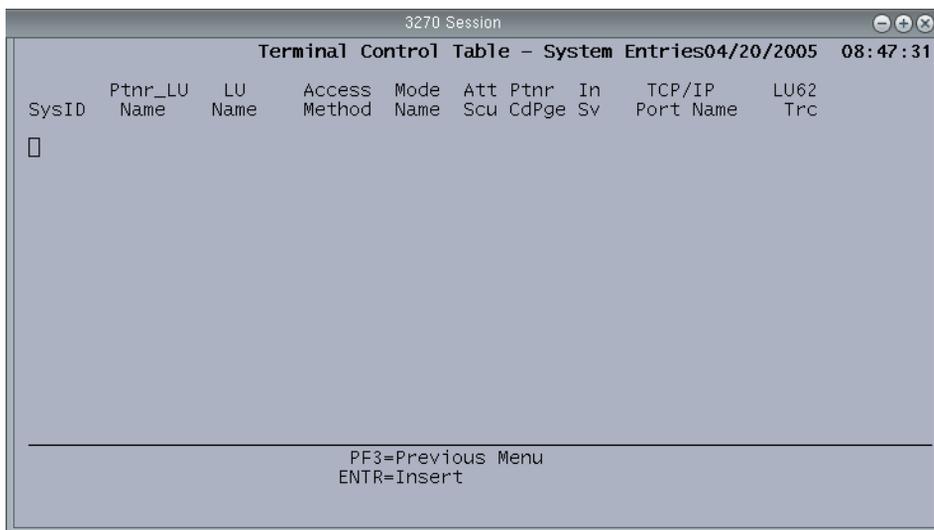


図 10-2 TCT - System Entries 挿入画面

9. 挿入画面で、ご使用の環境に必要であれば、次のフィールドに情報を入力します。
 - a. 「SysID」フィールドに、EXEC CICS コマンドの SYSID オプションで使用する、またはそのリソースが遠隔であることを定義するために Sun MTP テーブルで使用する遠隔システムのローカル識別子を入力します。
 - b. 「LU Name」フィールドに値を入力します。
これは、表示目的にだけ使用します。
 - c. 次のいずれかの値を「Access Method」フィールドに入力します。
 - 別の Sun MTP 領域への、保護されていないセッションの場合は TCP。
 - 保護セッションの場合は、TCP+。保護セッションは、ISC 通信リソースマネージャーによって管理されます。保護セッションの詳細については、『Sun Mainframe Transaction Processing ソフトウェア XA リソースマネージャーの使用』を参照してください。
 - d. 「Att Scu」フィールドに、次の値のいずれかを入力し遠隔システムのセキュリティレベルを指定します。
 - Local の場合は L、N、または空白
 - Identify の場合は、I
 詳細は、116 ページの「トランザクションセキュリティ」を参照してください。

- e. 「Ptnr CdPge」フィールドに、遠隔システムで使用するコードページを入力します。
 - 遠隔システムが、EBCDIC システムの場合、このフィールドに値を定義します (00037 など)。この値は、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』の説明にあるように、トランザクション経路指定出口ルーチンに渡されます。
 - 遠隔システムが ASCII システムの場合、このフィールドは空白のままにします。
 - f. 「In Svc」フィールドは、この接続で、最初からサービスを提供するかしらないかを定義します。Y または N を入力します。

サービスを提供しない場合、インバウンドとアウトバウンドの両方の ISC 要求は、この接続からは送信できません。CEMT トランザクションを使用して、領域の実行中に接続の状態を変更します。
 - g. 「TCP/IP Port Name」フィールドに、アウトバウンド ISC 要求を実行するために接続する遠隔システムのポート名または番号を入力します。
 - h. 「LU62 Trc」フィールドは、その接続に LU6.2 トレースを実行するかどうかを指定します。Y または N を入力します。

トレースがオンの場合、その接続を使用するトランザクション処理プログラム (unikixtran) についてトレースが開始されます。トレースファイルは、tcplu62_pid という名前です。\$KIXSYS ディレクトリに書き込まれます。この pid は、トレースファイルに書き込むプロセスのプロセス ID です。
10. Enter キーを押して、新しいエントリを挿入します。

「System Entries」画面に新しいエントリが表示されます。
 11. PF9 キーを押します。
 12. 次の画面が表示されます。
 - a. 「Host Name」フィールドに、Sun MTP または CICS 領域が実行される遠隔システムの名前を入力します。
 - b. 「Nq_Ptnr_Name」フィールドは空白のままにします。
 - c. データ入力を終了してから、Enter キーを押します。
 13. PF3 キーを押して、「System Entries」画面に戻ります。
 14. PF3 キーを押して、TCT メニュー画面に戻ります。
 15. PF2 キーを押して、変更をディスクに書き込みます。
 16. PF3 キーを 2 回押して、「Table Manager」メインメニューに戻ります。
 17. この領域を停止します。

18. テーブルの変更を有効にし、unikixtcp サーバーを起動するには、領域を再起動します。
- P オプションを付けて unikixmain (または、kixstart) を入力します。また、-L オプションやほかのオプションも必要であれば付けます。
 - -P オプションは、unikixtcp サーバーが TCP/IP 接続を待機するポートを指定します。これは、手順 1 で定義したポートです。
 - -L オプションは、サーバーがサポートできるクライアントの数を制限するファイル記述子の数を指定します。
19. -P オプションを使用して遠隔領域を起動して、両方の領域が TCP/IP で通信できるようにします。

トランザクションセキュリティー

Sun MTP は、ISC でサポートされる 2 つのレベルのセキュリティーを提供します。

- **Local:** セキュリティー情報は、システム間で渡されません。
- **Identify:** 遠隔システムを信頼して、ユーザーを確認します。発信側のシステムはユーザー名を送信し、遠隔システムがユーザーをサインオンします。

ご使用の SNA 製品の構成で使用される通信セキュリティーレベルは、TCT - System Entries テーブルで指定したセキュリティーレベルと一致している必要があります。

遠隔領域の自動インストール

TCP/IP を介して接続されている遠隔領域は、Sun MTP 領域に自動的にインストールできます。TCT - System Entries テーブルのエントリで、遠隔パートナー接続を定義する必要はありません。

KIXMAXIST 環境変数を使用して、領域に自動インストール可能な TCT システムエントリの最大数を指定します。デフォルトでは、この値はゼロに設定されています。\$KIXMAXIST に関する追加情報は、表 2-1 を参照してください。

注 - この方法は、インバウンド ISC が必要な場合だけ使用します。これは、Sun MTP 領域がこのエントリにシステム ID を作成するからです。このシステム ID を特定し、アウトバウンド ISC を領域に戻すために使用することは容易ではありません。

第11章

遠隔クライアントへの Sun MTP の構成

この章では、領域をさまざまなクライアントと機能できるように構成する方法について説明します。次のトピックについて説明します。

- 117 ページの「TN3270 クライアントへの領域の構成」
- 123 ページの「SNA 3270 クライアントの領域の構成」
- 133 ページの「ECI/EPI クライアントの領域の構成」
- 134 ページの「TCP/IP ソケットリスナーの起動」
- 135 ページの「SSL クライアントの領域の構成」
- 141 ページの「WebSphere MQ クライアントの領域の構成」

ほかのクライアントを使用する方法については、それぞれのマニュアルを参照してください。

- 『Sun MTP J3270 ユーザーズガイド』
 - 『Sun MTP Client ユーザーズガイド』
-

TN3270 クライアントへの領域の構成

TN3270 クライアントをサポートする領域を構成する前に、次のことを決定する必要があります。

- 次の TN3270 サーバーの構成要素
 - サポートする TN3270 サーバードプロセスの最大数
 - 終端の最大数
 - TN3270 サーバーの待機ポート
 - ユーザーがログインを要求されるかどうか
- 次の TN3270 クライアントの構成要素
 - 端末が事前定義されているかどうか
 - 事前定義されているプリンタ

次に、次の作業を行なって領域を構成する必要があります。

- unikixtnemux プロセスと呼ばれる、TN3270 サーバーを構成します。TN3270 サーバーは、unikixmain が自動的に起動する Sun MTP 通信マネージャー (unikixCommMgr) によって起動されます。領域が起動すると、TN3270 サーバーに関する情報が格納されている \$KIXSYS/unikixrc.cfg 構成ファイルが読み取られます。TN3270 クライアントをサポートする各領域には、一意のポート番号が含まれている unikixrc.cfg ファイルが必要です。詳細は、118 ページの「TN3270 サーバーを構成する」を参照してください。
- 事前定義された TN3270 端末とプリンタを構成します。これらのエントリは「事前定義の TCT」または「プリセット TCT」と呼ばれることもあります。詳細は、121 ページの「TN3270 クライアントを定義する」を参照してください。

▼ TN3270 サーバーを構成する

1. 領域設定ファイルを探して、UNIKIX と KIXSYS 環境変数が設定されていることを確認します。
2. unikixrc ファイルを、\$UNIKIX/lib ディレクトリから構成対象の領域の \$KIXSYS ディレクトリにコピーし、unikixrc.cfg という名前に変更します。

注 - unikixrc.cfg ファイルがすでに \$KIXSYS ディレクトリにある場合は、その既存のファイルを開いて適切なエントリを編集できます。

3. unikixrc.cfg ファイルへの書き込み権があることを確認します。
4. ファイルを開き、TNServer* エントリを編集します。

表 11-1 では、各エントリについて説明します。

unikixrc.cfg ファイルを変更する際、次の構文ルールが適用されます。

- 感嘆符 (!) はコメントを示します。感嘆符のあとの行にあるテキストは、無視されます。
- キーワードと引用符に囲まれない値には、大文字と小文字の組み合わせを使用できます。
- パス名やファイル名などの引用符に囲まれた値は、大文字と小文字が区別されません。

次の例は、TNServer* エントリを示します。

!Keywords	Values	Possible Values
TNServer*Active:	True	!True/False
TNServer*Processes:	1	! 1 - 255
TNServer*EndPoints:	200	! 16 - 1000/2000
TNServer*ListenPort:	5001	! 1025 - 32768
TNServer*UserLogin:	True	!True/False
TNServer*ASIS:	True	!True/False
TNServer*Debug:	False	!True/False
TNServer*DebugFileName:	"TNdebug.log"	!Any Legal Filename
TNServer*MessageLog:	False	!True/False
TNServer*MessageLogFileName:	"TNmsg.log"	!Any Legal Filename
TNServer*DefaultTransaction:	CPLT	!Any Legal Transaction
TNServer*ForceKeyboardReset:	False	!True/False

5. 作業が終了したら、ファイルを保存します。

6. この領域を起動します。

構成した数の TN3270 プロセスが起動されます。

注 - \$KIXSYS/unikixrc.cfg ファイルを変更するたびに、領域を再起動して変更を有効にする必要があります。

次の表では、unikixrc.cfg ファイルの TN3270 サーバーの項目を説明します。値を指定しない場合は、デフォルト値が使用されます。

表 11-1 unikixrc.cfg ファイル - TN3270 サーバーのエントリ

キーワード	説明
TNServer*Active	True の場合、unikixtnemux サーバーを起動します。False の場合、そのあとのすべての TNServer キーワードが無視されます。
TNServer*Processes	この領域に呼び出す unikixtnemux サーバープロセスの数。1 (デフォルト) ~ 255。1 つの unikixtnemux プロセスから、多数の TN3270 クライアントを多重化できます。
TNServer*EndPoints	1 つの unikixtnemux プロセスで管理できる同時実行ユーザーの最大数 (20 ~ 1000)。この値を TNServer*Processes で乗算した値は、一度に想定される TN3270 または TN3270E の同時ユーザーの最大数と等しくなければなりません。 unikixtnemux プロセス間で、過度の「リスナースワッピング」の発生を避けるために、10 ~ 20 の接続を追加する必要があります。デフォルトは 60 です。

表 11-1 unikixrc.cf ファイル - TN3270 サーバーのエントリ (続き)

キーワード	説明
TNServer*ListenPort	TN3270/TN3270E クライアントが接続する、1025 ~ 32,768 のポート番号。この値は一意でなければなりません。デフォルトは 2001 です。
TNServer*UserLogin	TN3270 ユーザーに領域へのアクセスを許可する前に、それぞれの UNIX ユーザー名とパスワードが使用されていることを検査します。値は、True (デフォルト) または False です。
TNServer*ASIS	True の場合、入力データを大文字に変換しません。値は、True または False (デフォルト) です。
TNServer*Debug	Sun MTP デバッグ情報をファイルに保存します。値は、True または False (デフォルト) です。 この機能を有効にするようにご購入先から指示されない限り、False に設定します。
TNServer*DebugFileName	TNServer*Debug 値が True に設定されている場合、このファイルを使用して、Sun MTP デバッグ情報を保存します。任意のファイル名を使用できます。デフォルトは、\$KIXSYS/TNdebug.log です。ファイルを作成する際、unikixtnemux のプロセス ID (pid) がファイル名に付加されます。たとえば、TNdebug.log.234 となります。
TNServer*MessageLog	TN3270 画面メッセージをファイルに保存します。値は、True または False (デフォルト) です。 この機能を有効にするようにご購入先から指示されない限り、False に設定します。
TNServer*MessageLogFilename	TNServer*MessageLog 値が True に設定されている場合、TN3270 メッセージが保存されるファイル。任意のファイル名を使用できます。デフォルトは、\$KIXSYS/TNmsg.log です。ファイルを作成する際、unikixtnemux のプロセス ID (pid) がファイル名に付加されます。たとえば、TNmsg.log.234 となります。
TNServer*DefaultTransaction	ユーザーが Sun MTP 領域にログインするときに実行される初期トランザクション。デフォルトは、CPLT です。
TNServer*ForceKeyboardReset	キーボードを強制的にリセットします。値は、True または False (デフォルト) です。True は、ユーザーが CICS コマンドでキーボードのリセットを指定しているかどうかに関係なく、トランザクションの終了時にユーザーのキーボードを強制的にリセットします。

TNServer*UserLogin キーワードが False に設定されている場合、クライアントの接続時に Sun MTP の著作権画面を表示するには、TNServer*DefaultTransaction キーワードをコメントアウトする必要があります。たとえば次のようにします。

```
!TNServer*DefaultTransaction: .....
```

▼ TN3270 クライアントを定義する

1. 領域が実行されていない場合は、起動します。
2. ローカルクライアントから、空白のトランザクション画面で CTBL を入力して「Standard Tables」メニューを表示し、PF4 キーを押します。
3. PF8 キーを押して、「Terminal Control Table」を表示します。
4. PF9 キーを押して、「3270 Devices」を表示します。
5. 「3270 Devices」画面で PF4 キーを押し、図 11-1 に示す挿入画面を表示します。

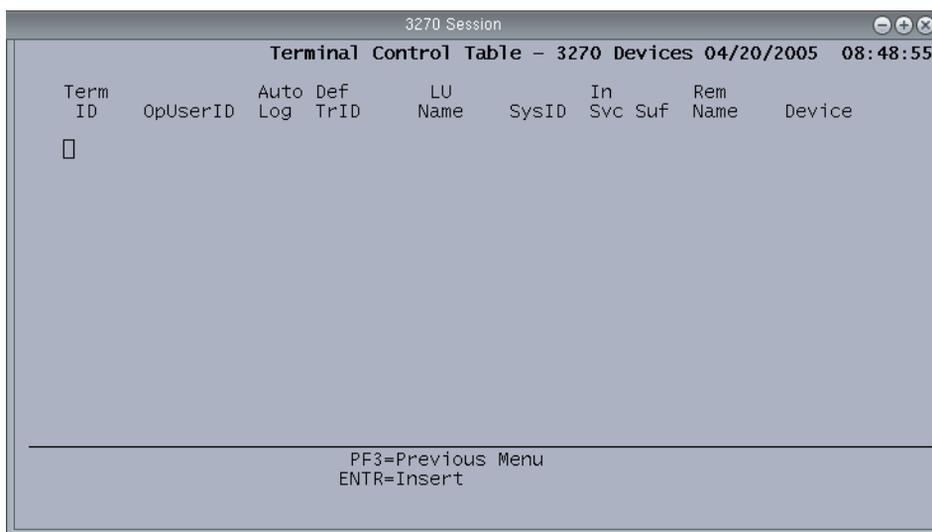


図 11-1 TCT - 3270 Devices 挿入画面

6. 「Term ID」フィールドに、アプリケーションに知らせる 4 文字の端末識別子を入力します。
この値は一意である必要があります。
7. 「LU Name」フィールドに、8 文字までの端末の名前を入力します。
プリンタなど、特定の Term ID を持つ複数の端末は、このフィールドで見分ける必要があります。
8. その他のフィールドはオプションです。
 - a. 「OpUser ID」フィールドに、サインオンテーブル (SNT) エントリと一致する 8 文字の値を入力します。
 - b. TN3270 デバイスには「Auto Log」フィールドは使用しません。

- c. 「Default TranID」フィールドにトランザクション識別子を入力すると、端末に入力されているトランザクション識別子は無効になります。
 - d. 「SysID」フィールドが空白、または SIT に設定されている SysID と一致する場合、Sun MTP はローカルシステムを使用します。別の値が入力されている場合、端末は遠隔とみなされます。遠隔システムは、TCT のシステムエントリとしても定義される必要があります。
 - e. 「In Svc」フィールドは使用しません。
 - f. 「Suf」フィールドに、BMS マッピング要求に使用する、7 文字のすべてのマップセットに付加される接尾辞を入力します。
このフィールドは、代替画面サイズや各国語サポートに使用できます。
 - g. 「RmtNam」フィールドに、遠隔システムに知らせる端末識別子を入力します。
このフィールドは、同じ端末 ID を持つ異なる遠隔システム上の 2 台の端末が、両方とも Sun MTP ヘトランザクションを経路指定する場合に使用します。これらの端末を、遠隔システムで定義されている遠隔名と一意の端末 ID を持つ端末として定義します。
 - h. 「Device」フィールドは、LU タイプ 1 プリンタを構成するときのみ使用します。
詳細は、100 ページの「LU タイプ 1 プリンタの構成」を参照してください。
- 9. Enter キーを押してエントリを挿入し、「3270 Devices」画面に戻ります。
 - 10. PF3 キーを押して、TCT メニューに戻ります。
 - 11. PF2 キーを押して、変更をディスクに保存します。
 - 12. Table Manager を終了し、領域を停止します。

▼ TN3270 クライアントから領域に接続する

- TN3270 クライアントから、接続先のホストとポート番号を指定します。

ポート番号は、unikixrc.cfg ファイルにあるポート番号に一致させる必要があります。

TN3270E では、クライアントは、端末とプリンタの LU 名を指定してプリンタと端末セッションを事前定義の端末 ID に関連付けることができます。

- クライアントが LU 名を指定しない場合、LU 名は自動的に作成されます。この名前の形式は *txxyyyyy* です。この *xx* は TN3270 サーバープロセスの番号で、*yyyyyy* は接続のソケットファイル記述子です (t0123456 など)。

- LU 名を指定する場合、TN3270 サーバーは TCT から対応するエントリを検索し、まだ使用されていない場合はそのエントリを使用します。プリンタの場合、この機能によって既知の端末 ID が提供されるので、アプリケーションはこの端末名にデータを出力できます。事前定義のエントリが見つからない場合、クライアントは自動的にインストールされ、LU 名が TCT に保存されます。

SNA 3270 クライアントの領域の構成

Sun MTP は、次のコンポーネントを使用して 3270 SNA デバイスをサポートします。

- **TPS PU4/5 サーバー**。PU4/5 サーバーは、3270 端末ユーザーに初期ログイン画面を表示します。このログイン画面は、ユーザーがアプリケーションを選択するように構成できます (TPS のマニュアルを参照)。メインフレームに接続する場合、ログイン画面は通常 VTAM (Virtual Telecommunications Access Method) が表示します。Sun MTP へのログインでは、ログインの `applid()` メソッドが必要です。
- **トランザクションイニシエータサーバー (unikixtrin)**。複数の 3270 クライアントプロセスが別々のシステムにある場合、遠隔クライアントは、Internet User Datagram ソケットを使用して Sun MTP ホストと通信します。トランザクションイニシエータサーバー、つまり `unikixtrin` は、適切な Sun MTP のメッセージキューにメッセージを置き、Sun MTP サーバースystemで遠隔クライアントのエージェントとして動作します。トランザクション処理プログラムと開始サーバーは、出力メッセージに同じデータグラムメカニズムを使用して、直接遠隔クライアントと相互に機能できます。

1 つの `unikixtrin` プロセスで、すべての 3270 クライアントをサポートします。

- **3270 クライアントイニシエータ (unikixi)**。3270 デバイスでアクセスされる各領域では、`unikixi` プロセスを実行する必要があります。3270 クライアントイニシエータは、次の 4 つの機能を実行します。
 - 3270 デバイスが使用可能になると領域に知らせます。
 - 3270 デバイスからの接続要求を受け取り、要求された 3270 デバイスに 3270 クライアントを起動します。
 - 領域の利用度を監視し、領域が使用できない場合 3270 クライアントに知らせます。
 - 自動ログインデバイスが構成されている場合、端末ログインを開始します。

3270 クライアントイニシエータは、Sun MTP と同じホストで実行できます。この方法が推奨されます。または、TPS SNA サーバーを実行するホストで実行できます。3270 クライアントイニシエータは、1 つの SNA サーバーだけをサポートします。各 Sun MTP 領域に接続される 3270 クライアントイニシエータは、1 つだけです。

- 3270 クライアント (unikixb)。領域に接続されている 3270 デバイスごとに、1 つのクライアントが起動されます。

TPS PU4/5 サーバーの構成

この節では、2 種類の構成シナリオについて示します。また、Sun MTP 領域と TPS PU4/5 サーバーを実行するために必要な構成ファイルについても説明します。

図 11-2 は、Sun MTP と SNA サーバーが同じホストに共存する構成を示します。

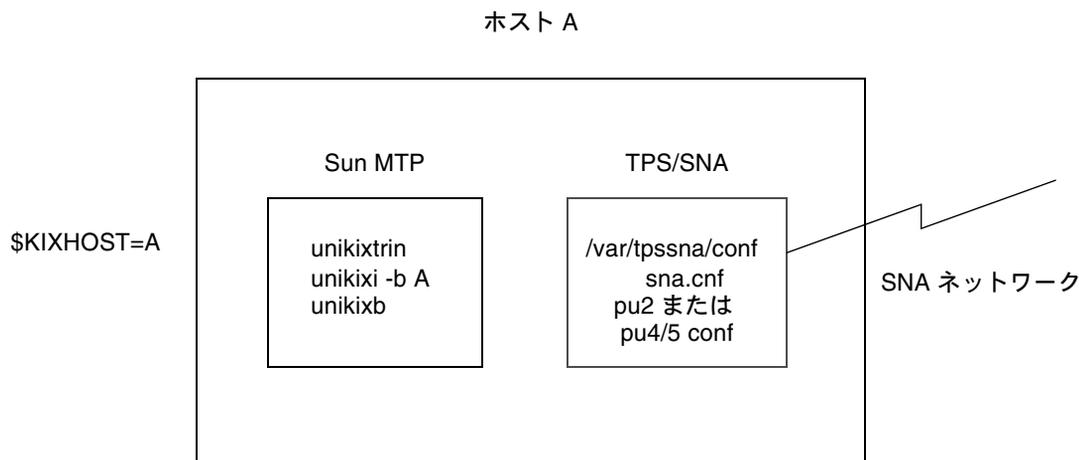


図 11-2 単一ホストの PU5 構成

次の図は、Sun MTP 領域がホスト A に、TPS SNA サーバーがホスト B にある 2 ホストの構成を示します。

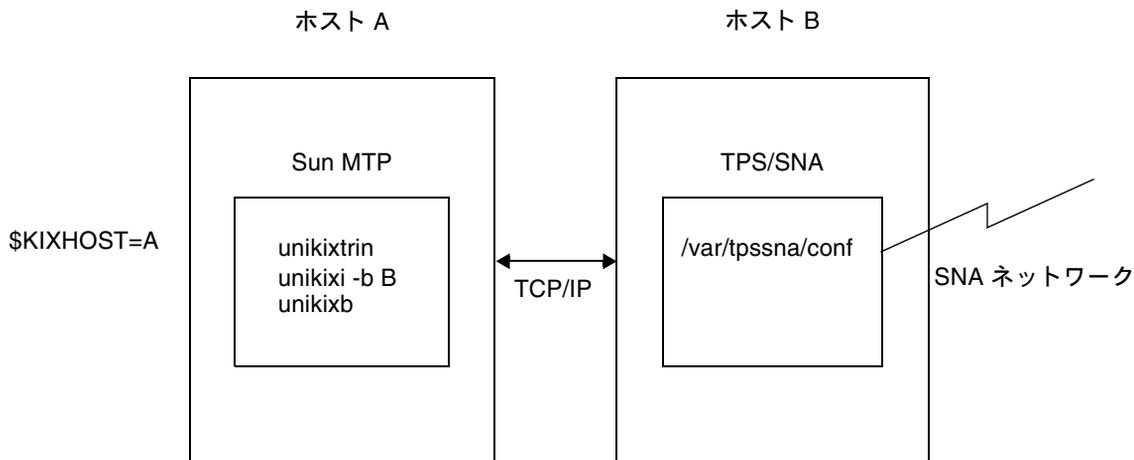


図 11-3 2 ホストの PU5 構成

次の例に示す構成ファイルは、SNA サーバーの /var/tpssna/conf ディレクトリに置く必要があります。常に、sna.cnf ファイルを必要とします。サイトの構成によっては、その他の構成ファイルが必要になることもあります。

PU2 構成

ご使用の TPS サーバーが PU2 デバイスに接続されている場合、次の例に示す sna.cnf.pu2 ファイルをカスタマイズし、名前を sna.cnf に変更して、/var/tpssna/conf ディレクトリに保存します。SSCPPORT エントリが 8100 であることを確認します。

コード例 11-1 sna.cnf.pu2 構成ファイル

```
# Sample sna.cnf for PU2
# NOTE:SSCPPORT must be set to 8100
#
cpname=IBMLAN.ORBITCP
SSCPPORT=8100
```

サイトの要件によって、自動ログオン構成ファイルとリスナー構成ファイルのどちらかを /var/tpssna/conf ディレクトリに置きます。

端末が領域に自動的にログインされる場合、次の例に示す構成ファイルが必要です。

コード例 11-2 自動ログイン構成ファイル

```
#
# sample PU2 configuration file:
# used to call (i.e. autologon) terminals between MTP and
# the PU2
#
DLCTYPE=8023
PUTYPE=PRI
DEVICE=ent0
CALLTYPE=CALL
RMTADDR=02608C2D001F
LCLSAP=64
RMTSAP=04
LINERETRY=10
LUPORT=8101
#
LU=1-5, NOTIFY=YES, LOG=NO, PSERVIC=02800000000185000007E00, PACING=00010100,
RUSIZES=87C7
#
```

コード例 11-3 に示す構成ファイルは、PU2 デバイスが 3270 端末からの接続を待機する場合に必要です。

コード例 11-3 リスナー構成ファイル

```
#
# Sample PU2 configuration:
# Used to 'listen' for PU2 connections
# where RUSIZES 87 -> 1024 and C7 -> 1536
#
DLCTYPE=8023
PUTYPE=PRI
DEVICE=ent0
CALLTYPE=LISTEN
RMTADDR=02608C2D001F
LCLSAP=64
RMTSAP=04
LINERETRY=10
LUPORT=8101
# LU=1-10
#
LU=1-5, NOTIFY=YES, LOG=NO, PSERVIC=02800000000185000007E00, PACING=00010100,
RUSIZES=87C7
#
```

PU4/5 構成

次のコード例に示す SNA 構成ファイルは、メインフレームへの PU4/5 接続をサポートします。TPS サーバーがメインフレームに接続されている場合、使用しているサイトのこのファイルをカスタマイズし、名前を `sna.cnf` に変更して、`/var/tpssna/conf` ディレクトリに保存します。コード例 11-4 に示すように、`SSCPPORT` エントリが `8100` であることを確認します。

コード例 11-4 `sna.cnf.pu45` 構成ファイル

```
#
# Sample sna.cnf for PU4/5 connection
# NOTE:SSCPPORT must be 8100
#
SSCPID=050000000111
# SSCPIO should be unique in network
XID=FFF00001
# XID number for XID exchange
CPNAME=USUTPHX1.CPORBIT
# local system Control Point name
SSCPPORT=8100
# SSCP server port number
```

サイトの要件によって、SDLC 構成ファイルと Ethernet 構成ファイルのどちらかを `/var/tpssna/conf` ディレクトリに置きます。

コード例 11-5 は、TPS サーバーが Ethernet によってメインフレームに接続されている環境の PU4/5 構成ファイルを示します。

コード例 11-5 Ethernet 構成ファイル

```
# Sample PU4/5 configuration for ethernet connection
# to mainframe/NCP
#
PUTYPE=PRI4CD           # PU 4/5 using Cross Domain
DLCTYPE=8023           # IEEE 802.3
DEVICE=ent0            # Ethernet device
IFRAME=1476            # I-frame minus 20 for FID4 headers 1476
LUPORT=8102            # LU API server port number
LOG=LONG                # change to NO when done testing
CALLTYPE=CALL          # call remote system
RMTADDR=080074700001   # remote MAC address
SUBAREA=4               # local subarea address
RMTSAP=08
LCLSAP=08
DESTSA=6, VR=0, ER=0   # remote subarea
BUFCNT=600             # sun restriction on shared memory size
LU=1-255                # local dependent LU table
```

端末やプリンタに自動ログインを使用する場合、コード例 11-5 に示す構成ファイルに次の行を追加する必要があります。

```
RMTSSCP=fully-qualified-name-of-SSCP, SUBAREA=n  
RMTRESC=LU-name
```

次に例を示します。

```
RMTSSCP=USUTPHX1.P390SSCP, SUBAREA=6  
RMTRESC=CPXREEN3
```

RMTSSCP 文の完全修飾名は、接続する LU を制御する SSCP の完全修飾名でなければなりません。SUBAREA は、その SSCP のサブ領域のアドレスを指定します。

RMTRESC エントリは、前にある RMTSSCP 文によって制御する LU 名を指定します。ワイルドカード名にすることもできます。RMTRESC 文は、複数でもかまいません。

注 - RMTRESC は、通常 RMTRESC=* に設定されます。これは、すべての自動ログイン要求が、RMTSSCP 文によって指定された SSCP に送信されることを示します。

TPS PU4/5 SNA サーバーと動作する領域の構成

この節では、遠隔の SNA 3270 デバイスをサポートするように領域を構成する方法について説明します。

▼ 領域を構成する

1. TPS SNA サーバソフトウェアをインストールします。

パッケージに付属しているマニュアルを参照してください。

124 ページの「TPS PU4/5 サーバーの構成」は、/var/tpssna/conf ディレクトリに配置する必要がある構成ファイルのサンプルを示します。

2. Sun MTP ホストとクライアントイニシエータシステムに KIXHOST 環境変数を設定します。

クライアントイニシエータシステムで、Sun MTP 領域を実行するホストを指定します。Sun MTP 領域を実行するホストで、自分自身を指定し、3270 クライアントからの接続が予想されることを領域に知らせます。

3. unikixi が接続するポート番号を決めます。指定するポート番号が、ほかのアプリケーションと重複していないことを確認します。5000 より大きなポート番号を割り当てることをお勧めします。
すべての端末が自動インストールされる場合は、手順 16 に進みます。
4. ローカルクライアントまたは TN3270 クライアントから空白のトランザクション画面で CTBL を入力し、「Standard Tables」メニューを表示して PF4 キーを押します。
5. PF8 キーを押して TCT を表示し、PF9 キーを押して「3270 Devices」画面を表示します。
6. 「3270 Devices」画面で PF4 キーを押し、図 11-4 に示す挿入画面を表示します。

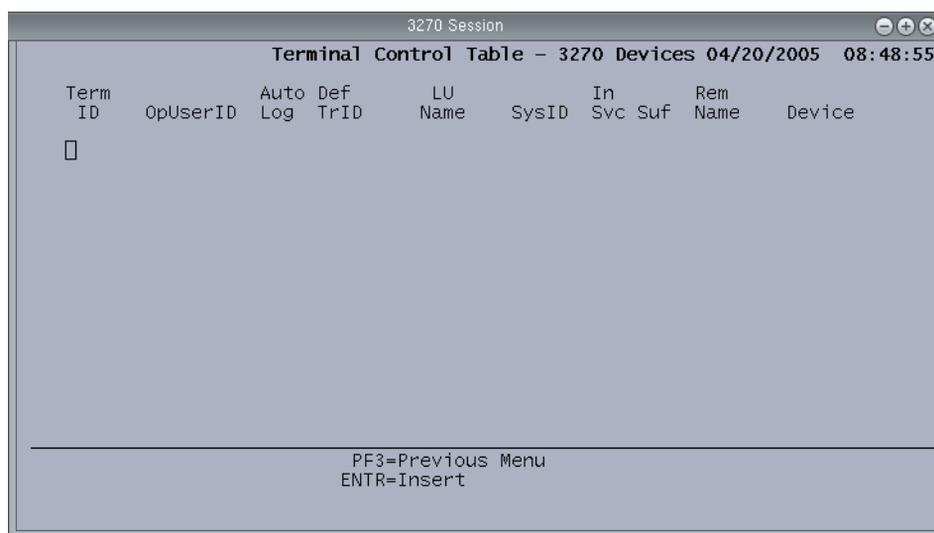


図 11-4 TCT - 3270 Devices 挿入画面

7. 自動ログインには、次のフィールドへの入力が必要です。
 - a. 「Term ID」フィールドに、アプリケーションに知らせる 4 文字の端末識別子を入力します。
この値は一意である必要があります。
 - b. 3270 クライアントイニシエータが有効なときにこの端末 (またはプリンタ) に自動ログインするには、「Auto Log」フィールドに、Y を入力するか空白のままにします。

- c. 「LU Name」フィールドに、SNA ネットワークに知らせる 8 文字の端末の名前を入力します。

LU 名は、SNA サーバーを PU4/5 パッケージとインストールするときに構成されます。LU 名の割り当てについては、PU4/5 のマニュアルを参照してください。

プリンタなど、特定の Term ID を持つ複数の端末は、このフィールドで見分ける必要があります。

8. その他のフィールドはオプションです。

- a. 「OpUser ID」フィールドに、サインオンテーブル (SNT) エントリと一致する 8 文字の値を入力します。
- b. 「Default TranID」フィールドにトランザクション識別子を入力すると、端末に入力されているトランザクション識別子は無効になります。
- c. 「SysID」フィールドが空白、または SIT に構成されている SysID と一致する場合、Sun MTP はローカルシステムを使用します。別の値が入力されている場合、端末は遠隔とみなされます。遠隔システムは、TCT のシステムエントリとしても定義される必要があります。
- d. 「In Svc」フィールドは、3270 接続の場合は無視されます。

- e. 「Suf」フィールドに、BMS マッピング要求に使用する、7 文字のすべてのマップセットに付加される接尾辞を入力します。
このフィールドは、代替画面サイズや各国語サポートに使用できます。

- f. 「RmtNam」フィールドに、遠隔システムに知らせる端末識別子を入力します。
このフィールドは、同じ端末 ID を持つ異なる遠隔システム上の 2 台の端末が、両方とも Sun MTP へトランザクションを経路指定する場合に使用します。これらの端末を、遠隔システムで定義されている遠隔名と一意の端末 ID を持つ端末として定義します。

- g. 「Device」フィールドは、LU タイプ 1 プリンタを構成するときのみ使用します。

LU タイプ 1 プリンタは、TN3270 デバイスとしてのみサポートされます。詳細は、100 ページの「LU タイプ 1 プリンタの構成」を参照してください。

9. Enter キーを押してエントリを挿入し、「3270 Devices」画面に戻ります。
10. 遠隔システムから端末やプリンタに自動ログインする場合、端末のポート番号も指定する必要があります。
 - a. メイン画面で端末識別子を選択します。
 - b. 「3270 Devices」画面の PF9 キーを押して、ホストとポートの画面を表示します。



図 11-5 TCT - 3270 Devices のホストとポートの画面

11. 次のフィールドに適切な情報を入力します。

注 - 「Term ID」フィールドには、メイン画面の Term ID 値が含まれます。

- a. 「Host」フィールドに、3270 クライアントイニシエータが実行中のホストの名前を入力します。
このフィールドが空白の場合、Sun MTP は領域が実行中のホスト名を代用します。
 - b. 「Port Number」フィールドに、16 文字までのポート番号または名前を入力します。
このポート番号は、unikixi に -o オプションを付けて指定する必要があります。これは、3270 クライアントイニシエータが、自動ログインを行う unikixtstrt からのメッセージを待機するポート番号です。
12. Enter キーを押して値を挿入し、「3270 Devices」画面に戻ります。
 13. PF3 キーを押して、TCT メニューに戻ります。
 14. PF2 キーを押して、変更をディスクに保存します。
 15. Table Manager を終了し、領域を停止します。
 16. unikixmain (kixstart) に -s オプションを付けて、領域を起動します。これは、unikixtrin、つまりトランザクションイニシエータサーバーを起動し、3270 デバイスが待機するポートを指定します。

17. 3270 クライアントイニシエータプロセスである unikixi を、-s オプションと、必要であればその他のオプションも付けて起動します。この -s オプションは、unikixmain の -s オプションと同じポート番号を指定する必要があります。

手順 11 で、TCT にポート番号を入力した場合は、-o オプションも unikixi に指定する必要があります。

必要なオプションを付けて領域を起動後、3270 クライアントは次のように機能します。

1. 3270 クライアントイニシエータは、バックグラウンドで実行され、Sun MTP から 3270 デバイスへの接続要求や、3270 デバイスから Sun MTP への接続要求を待機します。3270 端末には、3270 クライアントイニシエータの起動時に自動的にログオンすることも、ユーザーが手動でログインすることもできます。
2. ユーザーが、3270 端末で、手動ログオンシーケンスを開始します。正しい構成の Sun MTP と SNA サーバーを実行すると、サーバーに接続されている 3274 コントローラに取り付けられている端末にログイン画面が表示されます。通常このログイン画面には、使用できるアプリケーションの名前が表示されます。
3. ログインするには、必要なアプリケーションの名前を入力します。アプリケーションか領域のどちらかのクライアントイニシエータが実行されていない場合、ユーザーがログインしようとするすると端末はエラーメッセージを表示します。
4. ユーザーがログインを要求したあと、3270 クライアントである unikixb が起動し、Sun MTP の著作権画面を表示します。
5. この時点では、端末から何を伝送しても、初期のデフォルトトランザクション、つまり CPLT を領域にサブミットすることになります。ユーザー入力は破棄され、CPLT は画面を消去してカーソルをホームポジションに置きます。
6. これでユーザーはトランザクションを入力できるようになります。
7. 端末のログアウト時には、VTAM または TPS ログイン画面が表示されます。

unikixi の -u と -s オプションと、KIXHOST 環境変数は、SNA サーバーアプリケーション名とそのホストと、実行する Sun MTP アプリケーションの間のリンクを提供します。

unikixi オプションについては、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

ECI/EPI クライアントの領域の構成

次の表は、Sun MTP がサポートする ECI/EPI クライアントを示します。

表 11-2 サポートする ECI/EPI クライアント

ECI/EPI クライアント	サポートするプロトコル
Sun MTP クライアント	TCP/IP
IBM CICS クライアント	TCP/IP SNA
IBM CICS ユニバーサルクライアント	TCP/IP SNA

ECI/EPI クライアントアプリケーションの構成や記述については、使用する ECI/EPI クライアントのマニュアルを参照してください。

▼ ECI/EPI クライアントを受け入れる領域を構成する

1. クライアントシステムで、必要な構成タスクを実行します。
2. 領域が実行中であれば、停止します。
3. 領域の設定ファイルに、`KIXMAXIST` 環境変数を設定します。これは、領域に自動インストールできる ECI/EPI クライアントの数を指定します。
詳細は、134 ページの「ECI/EPI クライアントの自動インストール」を参照してください。
4. ISC 接続を構成します。
 - TCP/IP 上で実行中の場合は、113 ページの「TCP/IP の領域の構成」の説明にあるように TCP サーバーを構成します。
 - SNA 上で実行中の場合は、108 ページの「SNA の領域の構成」の説明にあるように `unikixdcl` サーバーを構成します。
5. 領域を再起動して、使用している通信サーバーも起動します。
これで、ECI/EPI クライアントを受け入れる準備が整いました。

ECI/EPI クライアントの自動インストール

Sun MTP ECI/EPI クライアントまたは CICS クライアントは、Sun MTP に自動インストールできます。TCT - System Entries テーブルで、遠隔パートナー接続を定義する必要はありません。

KIXMAXIST 環境変数を使用して、領域に自動インストール可能なシステム項目の最大数を指定します。\$KIXMAXIST に関する追加情報は、表 2-1 を参照してください。

TCP/IP ソケットリスナーの起動

領域でソケットクライアントからの通信を受け入れるには、ソケットリスナープロセス unikixsock を開始する必要があります。この節では、ソケットリスナーを開始する方法と、タイムアウト時間を設定する方法について説明します。

『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』に、ソケットの設定とトランザクションの送受信を行うユーザーアプリケーションプログラムについて説明されています。ソケットユーザー出口プログラムのカスタマイズ方法については、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』も参照してください。

▼ unikixsock プロセスを開始する

1. unikixmain コマンドの -p オプションを使用して、ポート番号、またはよく使用される名前を指定します。

次に例を示します。

```
$ unikixmain . . . -p 5090
```

または

```
$ unikixmain . . . -p cicstp
```

ポート番号と名前は、一意のものを使用する必要があります。

注 - 5000 以上のポート番号を割り当てれば、通常は重複しません。

ソケットのタイムアウトの設定

ソケットプロトコルは、ソケットクライアントが接続してから、初期ソケットメッセージを送信することを指定しています。この初期の送信が行われないと、ソケットクライアントが切断しない限り、このソケットは unikixsock プロセスによって管理されたままになります。この結果、その領域の動作が異常になります。

KIXSOCKET_TIMEOUT 環境変数を使用すると、クライアントを切断する前に unikixsock プロセスがソケットデータを待機する時間を指定できます。領域を起動する前に、この環境変数を設定する必要があります。たとえば、タイムアウト時間を 10 秒に設定するには、次のように変数の値を 10 に設定します。

```
KIXSOCKET_TIMEOUT=10;export KIXSOCKET_TIMEOUT
```

SSL クライアントの領域の構成

クライアントアプリケーションは、SSL を使用して Sun MTP 領域と通信できます。リスニングサーバプロセスの unikixssl は、事前定義されたポートで着信要求を待機します。unikixssl プロセスは、unikixsock リスナープロセスへ要求を経路指定します。このプロセスは、領域で実行する必要があります。

▼ SSL 要求を受け入れる領域を構成する

1. TCP/IP ソケットリスナープロセスの unikixsock に、待機ポートを決定します。
unikixsock リスナープロセスを開始するために領域を起動する際、unikixmain に `-p port-number` オプションを使用する必要があります。
このポート番号は、手順 6 で unikixrc.cfg ファイルに追加する必要があります。
2. SSL リスナープロセスの unikixssl に、待機ポートを決定します。
このポート番号は、手順 6 で unikixrc.cfg ファイルに追加する必要があります。

注 - これは、unikixsock のものとは異なる待機ポートである必要があります。

3. Network Security Services (NSS)¹ ライブラリがシステムにインストールされていることを確認します。
通常は、これらのライブラリは /usr/lib/mps/secv1 にあります。

1. Network Security Services (NSS) は SSL クライアント/サーバアプリケーションをサポートしています。

4. NSS ツールがシステムにインストールされていることを確認します。
通常は、これらのツールは /usr/sfw/bin にあります。
5. 認証データベースを設定します。
詳細は、136 ページの「認証データベースの設定」を参照してください。
6. unikixrc.cfg ファイルを SSL エントリで作成、または更新します。
詳細は、136 ページの「unikixssl サーバーへの通信マネージャーの設定」を参照してください。
7. インストールされている NSS ライブラリのディレクトリが、領域の設定ファイルの LD_LIBRARY_PATH 環境変数で指定されていることを確認します。
8. unikixssl が SSL ユーザー出口ライブラリ (libkxsslxit.so) にアクセスできるように、\$UNIKIX/lib ディレクトリが LD_LIBRARY_PATH 環境変数で設定されていることを確認します。

認証データベースの設定

Sun MTP の設定時、SSL サーバー認証および信頼される認証局の一覧を格納する認証データベースのディレクトリを指定する必要があります。サンプルの設定スクリプト \$UNIKIX/src/socket/sslsetup を参照してください。このサンプルでは、SSL で使用する認証データベースの設定例が示されています。

unikixssl サーバーへの通信マネージャーの設定

通信マネージャー (unikixCommMgr) は、unikixmain によって自動的に起動されます。起動時、\$KIXSYS/unikixrc.cfg という名前の構成ファイルを読み込みます。unikixrc.cfg ファイルには、unikixCommMgr が起動する SSL サーバーである unikixssl に関する情報が含まれます。SSL サーバーをサポートする各領域には、それぞれの unikixrc.cfg ファイルが必要です。

▼ unikixssl を開始するように unikixCommMgr を構成する

1. 領域設定ファイルを探して、UNIKIX と KIXSYS 環境変数が設定されていることを確認します。
2. unikixrc ファイルを、\$UNIKIX/lib ディレクトリから構成対象の領域の \$KIXSYS ディレクトリにコピーし、unikixrc.cfg という名前に変更します。

注 - unikixrc.cfg ファイルがすでに \$KIXSYS ディレクトリにある場合は、その既存のファイルを開いて適切なエントリを編集できます。

3. unikixrc.cfg ファイルへの書き込み権があることを確認します。
4. ファイルを開き、SslServer* エントリを編集します。

表 11-3 では、各エントリについて説明します。

unikixrc.cfg ファイルを変更する際、次の構文ルールが適用されます。

- 感嘆符 (!) はコメントを示します。! のあとに表示されるテキストは、無視されません。
- キーワードと引用符に囲まれない値には、大文字と小文字の組み合わせを使用できます。
- パス名やファイル名などの引用符に囲まれた値は、大文字と小文字が区別されません。

注 – すべての暗号スイートのデフォルトは、False です (無効)。少なくとも 1 つの暗号スイート (True に設定) を有効にしなければ、すべてのクライアント要求が拒否されます。

表 11-3 unikixrc.cfg ファイル - SSL サーバーエントリ (1 / 4)

キーワード	説明
SslServer*Active	True の場合、unikixssl サーバーを起動します。 False の場合 (デフォルト)、unikixssl サーバーは起動されず、そのあとのすべての SslServer キーワードは無視されます。
SslServer*Debug	unikixssl トレース情報をファイルに保存します。値は、True または False (デフォルト) です。ご購入先からこの機能を有効にするように指示がない限り、False に設定します。
SslServer*Host	クライアント接続を待機するホスト。ホスト名、IP アドレス、any (すべての IP アドレス - INADDR_ANY) または loopback (ローカルホスト - INADDR_LOOPBACK) を指定できます。IP アドレスを指定する場合は、引用符で囲む必要があります。 このエントリを組み込まない場合、デフォルトは any です。ほとんどの場合、これが適切です。
SslServer*Port	クライアント接続を待機する SSL ポート。

表 11-3 unikixrc.cfg ファイル - SSL サーバーエントリ (2 / 4)

キーワード	説明
SslServer*Sockhost	unikixsock ホスト。これは、unikixmain コマンドに <code>-h</code> オプションを付けて指定したホストです。ホスト名、IP アドレス、any (すべての IP アドレス - <code>INADDR_ANY</code>) または loopback (ローカルホスト - <code>INADDR_LOOPBACK</code>) を指定できます。IP アドレスを指定する場合は、引用符で囲む必要があります。 このエントリを組み込まない場合、デフォルトは any です。ほとんどの場合、これが適切です。
SslServer*Sockport	unikixsock 待機ポート。これは、unikixmain コマンド行の <code>-p</code> オプションで指定されたポート番号と一致する必要があります。
SslServer*Certdir	unikixssl サーバー認証を含む認証データベースのパス名。○
SslServer*Nickname	unikixssl サーバー認証のニックネーム。○
SslServer>Password	unikixssl サーバー認証データベースのパスワード。○
SslServer*Clientcertrequested	True の場合、unikixssl サーバーはクライアントに認証を要求します。これがデフォルトです。 False の場合、クライアントは認証するように求められません。
SslServer*Clientcertrequired	SslServer*Clientcertrequested エントリが True に設定されている場合に有効です。 True の場合、クライアントは有効な認証を提供し、接続を確立する必要があります。これがデフォルトです。 False の場合、クライアントは認証をオプションで提供できます。クライアントが有効な認証を提供するかどうかに関係なく、接続は確立されます。
SslServer*SSL_EN_RC4_128_WITH_MD5	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_EN_RC4_128_EXPORT40_WITH_MD5	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。

表 11-3 unikixrc.cfg ファイル - SSL サーバーエントリ (3 / 4)

キーワード	説明
SslServer*SSL_EN_RC2_128_CBC_WITH_MD5	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_EN_RC2_128_CBC_EXPORT40_WITH_MD5	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_EN_DES_64_CBC_WITH_MD5	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_EN_DES_192_EDE3_CBC_WITH_MD5	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_RSA_WITH_NULL_MD5	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_RSA_EXPORT_WITH_RC4_40_MD5	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_RSA_WITH_RC4_128_MD5	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_RSA_WITH_DES_CBC_SHA	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_RSA_WITH_3DES_EDE_CBC_SHA	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_FORTEZZA_DMS_WITH_NULL_SHA	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_FORTEZZA_DMS_WITH_FORTEZZA_CBC_SHA	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。

表 11-3 unikixrc.cfg ファイル - SSL サーバーエントリ (4 / 4)

キーワード	説明
SslServer*SSL_FORTEZZA_DMS_WITH_RC4_128_SHA	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_RSA_FIPS_WITH_3DES_EDE_CBC_SHA	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*SSL_RSA_FIPS_WITH_DES_CBC_SHA	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*TLS_RSA_EXPORT1024_WITH_DES_CBC_SHA	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。
SslServer*TLS_RSA_EXPORT1024_WITH_RC4_56_SHA	暗号スイート。 True の場合、この暗号スイートは有効です。 False の場合、この暗号スイートは無効です。

5. 作業が終了したら、ファイルを保存します。

注 - \$KIXSYS/unikixrc.cfg ファイルを変更するたびに、領域を再起動して変更を有効にする必要があります。

▼ 領域と SSL サーバーを起動する

1. unikixrc.cfg ファイルに適切な値が設定されていることを確認します。

2. 領域の環境設定を確認します。

共有ライブラリの環境変数に、SSL ライブラリのディレクトリの場所と \$UNIKIX/lib が含まれていることを確認します。

3. この領域を起動します。

unikixmain (kixstart) コマンドで -p オプションを指定し、ソケットリスナーを開始する必要があります。ポート番号は、unikixrc.cfg ファイルにある SslServer*Sockport 値に一致させる必要があります。

オプションで、-h オプションを指定できます (ソケットリスナーをバインドするホストを指定)。-h オプションを使用する場合、その値は、unikixrc.cfg ファイルの SslServer*Sockhost 値と一致している必要があります。-h オプションの詳細は、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

SSL の使用については、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』を参照してください。

WebSphere MQ クライアントの領域の構成

WebSphere MQ は、Sun MTP とともに使用できる Sun 以外の製品です。WebSphere MQ の使用方法については、『Sun Mainframe Transaction Processing ソフトウェア 開発者ガイド』および IBM WebSphere MQ のマニュアルを参照してください。

通信マネージャー (unikixCommMgr) は、unikixmain によって自動的に起動されます。これは、\$KIXSYS/unikixrc.cfg と呼ばれる構成ファイルを起動時に読み込みます。unikixrc.cfg ファイルには、unikixCommMgr が起動する MQ サーバーである unikixqm に関する情報が含まれます。WebSphere MQ をサポートする各領域に、それぞれの unikixrc.cfg ファイルが必要です。

▼ unikixqm サーバーを起動するために unikixCommMgr を構成する

1. unikixrc ファイルを、\$UNIKIX/lib ディレクトリから構成する領域の \$KIXSYS ディレクトリにコピーします。
2. MQServer* エントリを編集し、ファイルを unikixrc.cfg として保存します。
表 11-4 では、各エントリについて説明します。

unikixrc.cfg ファイルを変更する際、次の構文ルールが適用されます。

- 感嘆符 (!) はコメントを示します。! のあとの行にあるテキストは、無視されます。
- キーワードと引用符に囲まれない値には、大文字と小文字の組み合わせを使用できます。
- パス名やファイル名などの引用符に囲まれた値は、大文字と小文字が区別されます。
- ピリオドなどの特殊文字を含む値は、引用符で囲みます。

- すべて大文字の値は、引用符で囲みます。
次の例は、MQServer* エントリを示します。

!Keywords	Values	Possible Values
MQServer*Active:	True	!True/False
MQServer*QueueManagerName:	"TESTQM"	!Any legal name
MQServer*QueueName:	"UNIKIX.TRIGGER.QUEUE"	!Any legal name
MQServer*Debug:	False	!True/False

3. このファイルを unikixrc.cfg として保存します。

WebSphere MQ 接続の unikixrc.cfg ファイルに、次の項目を定義できます。値を指定しない場合は、デフォルト値が使用されます。

表 11-4 unikixrc.cfg ファイル - WebSphere MQ サーバーエントリ

キーワード	説明
MQServer*Active	True の場合、unikixqm サーバーを起動します。 False の場合、そのあとにあるすべての MQServer キーワードが無視されます。
MQServer*QueueManagerName	48 文字までの、キューマネージャーの名前。
MQServer*QueueName	48 文字までの、キューの名前。
MQServer*Debug	Sun MTP デバッグ情報をファイルに保存します。値は、True または False (デフォルト) です。実装されていません。

第12章

Sun MBM と機能する Sun MTP の構成

この章では、Sun MBM と機能できるように領域を構成する方法について説明します。この章の内容は、次のとおりです。

- 144 ページの「VCT での Sun MBM の有効化」
- 146 ページの「VSAM ファイルの再使用不可の定義」

注 – Sun MTP をアップグレードし、それと既存のバージョンの Sun MBM を使用する場合、ご使用のサブシステムを更新して新しいバージョンの Sun MTP を使用方法については、『Sun Mainframe Batch Manager ソフトウェア インストールガイド』を参照してください。また、リリースに固有の情報については、『ご使用にあたって』を参照してください。

Sun MBM バッチジョブが VSAM ファイルにアクセスするとき、Sun MTP 領域がジョブの実行を制御します。図 12-1 は、Sun MBM が VSAM データセットにアクセスする方法を示します。Sun MBM ジョブが VSAM ファイルとリレーショナルデータベース (Oracle® など) の両方にアクセスする場合、領域は VSAM と RDBMS アクセスの両方を制御します。

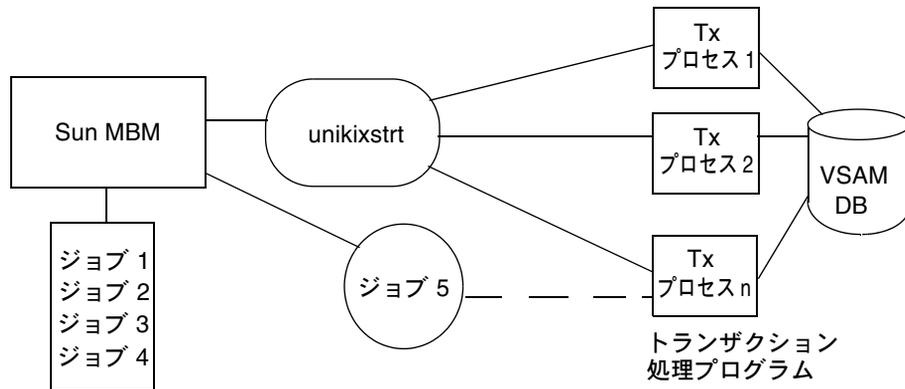


図 12-1 Sun MBM と接続した VSAM データセットへのアクセス

VCT での Sun MBM の有効化

▼ VCT で Sun MBM を有効にする

1. Table Manager を開始し、PF4 キーを押して「Standard Tables」メニューを表示します。
2. PF10 キーを押して、図 12-2 に示すように VCT を表示します。

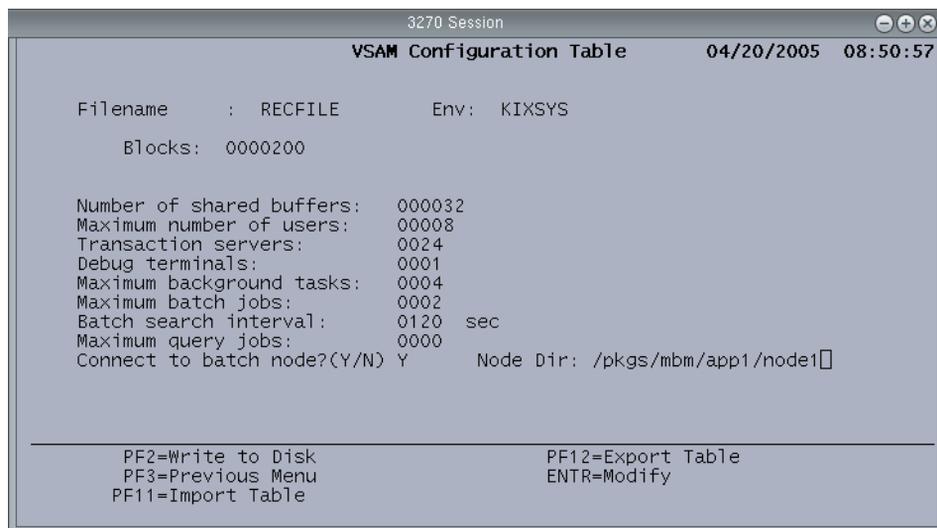


図 12-2 VSAM 構成テーブル (VCT)

3. ご使用の環境に合わせて VCT をカスタマイズしていない場合は、59 ページの「VCT のカスタマイズ」の手順を実行します。
4. Sun MBM を使用するためには、次のフィールドに値を入力する必要があります。
 - a. 「Maximum Batch Jobs」フィールドには、すべての Sun MBM ジョブクラスに割り当てられたスレッドの最大数以上の値を入力する必要があります。
ジョブクラスとスレッドについては、Sun MBM 『Sun Mainframe Batch Manager ソフトウェア 構成ガイド』を参照してください。
 - b. 「Batch Search Interval」フィールドに、実行を待機しているバッチジョブがあるかどうかを領域がチェックする頻度を秒数で指定します。
 - c. 「Connect to Batch Node」フィールドに、Y を入力します。
 - d. 「Node Dir」フィールドに、ノードのインストールディレクトリのパス名を入力します。
5. PF2 キーを押して、VCT を保存します。
6. PF3 キーを押して、VCT を終了します。
7. Table Manager を終了します。
8. 領域を再起動し、変更を有効にします。

VSAM ファイルの再使用不可の定義

Sun MBM を Sun MTP と使用する場合、IDCAMS REPRO と DELETE 機能は、Sun MTP unikixbld ユーティリティを呼び出します。これは、VSAM カタログのクラスタ定義を使用して、VSAM ファイルが再使用可能かどうかを判断します。unikixbld ユーティリティの詳細については、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

デフォルトでは、VSAM ファイルは再使用可能としてカタログに定義されますが、これはデフォルトが再使用不可に設定されているメインフレームと一致しません。バッチジョブが確実に期待通りに実行されるようにするには、VSAM カタログのクラスタ定義を調査し、ファイルが正しく定義されていることを確認する必要があります。

注 – Sun MBM 環境で、メインフレームのデフォルト、NOREUSE を使用する場合は、システムをそのように構成する必要があります。メインフレームの互換性オプションの詳細は、『Sun Mainframe Batch Manager ソフトウェア 構成ガイド』を参照してください。

VSAM ファイルを再使用不可に定義するには、定義するファイルが少ない場合は File Manager を使用します。ファイル数が多い場合は、kiximpcat と kixexpcat ユーティリティを使用すると簡単に再定義できます。

▼ File Manager を使用して VSAM ファイルを再使用不可に定義する

1. 領域を開始して、接続を確立します。
2. 空白のトランザクション画面で、CFMS を入力し、File Manager を表示します。
3. 新しいクラスタを追加する場合は、PF6 キーを押します。
または、変更するクラスタを選択して PF2 キーを押します。

4. クラスタ画面が表示されるので、「Reuse Allowed」フィールドに N を入力します。

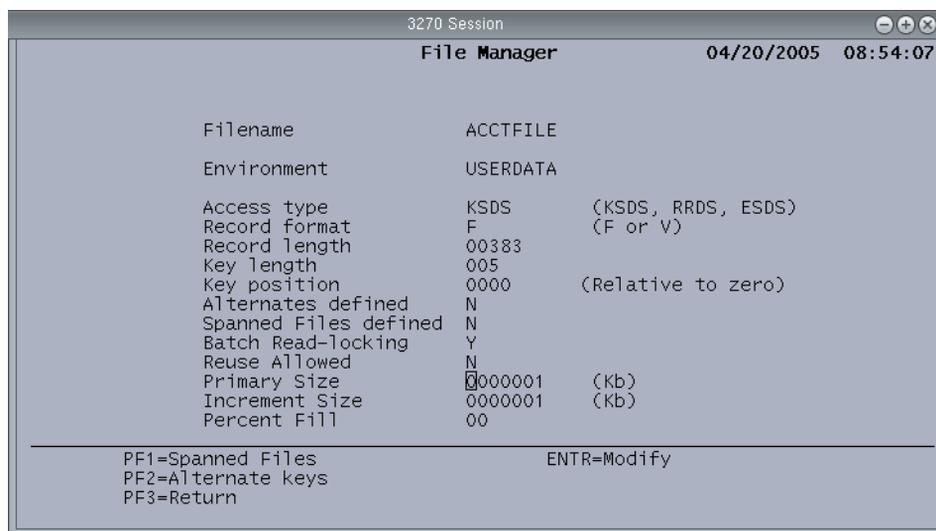


図 12-3 File Manager - 再使用不可のファイルの定義

5. Enter キーを押します。
6. PF3 キーを押して、「File Manager」画面に戻ります。

注 - 「Reuse Allowed」フィールドは、「File Manager」メイン画面には表示されません。

7. PF3 キーを押して、File Manager を終了します。

▼ ユーティリティを使用してファイルを再使用不可に定義する

1. この領域を停止します。
2. \$KIXSYS ディレクトリに変更して VSAM カタログをエクスポートします。

```
$ cd $KIXSYS
$ kixexpcat -c CATALOG -l CATALOG.lst
```

3. テキストエディタを使用して、再使用不可に定義するファイルごとに、「reuse allowed」指示子を N に変更します。

「reuse allowed」指示子は、各プライマリクラスタ定義の最後のアルファベットのエントリです。

```
P,KIXDATA0      ,JFILE001  ,F,KSDS,00080,00008,00000,Y,N,0000001,0000001,00
```

4. CATALOG.1st ファイルを保存します。
5. \$KIXSYS ディレクトリで、変更済みの CATALOG.1st ファイルをインポートします。

```
$ kiximpcat -l CATALOG.1st -c CATALOG
```

VSAM カタログの管理についての詳細は、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

第13章

RDBMS と動作する領域の構成

注 - XA リソースマネージャーを使用して領域を構成している場合は、この章の情報は当てはまりません。代わりに、『Sun Mainframe Transaction Processing ソフトウェア XA リソースマネージャーの使用』を参照してください。

この章では、リレーショナルデータベース管理システム (RDBMS) を扱えるように領域を構成する方法について説明します。次の節で各タスクについて説明します。

- 149 ページの「RDBMS 環境変数の設定」
- 150 ページの「kixinstall を使用した RDBMS の選択」
- 151 ページの「SIT での RDBMS 値の指定」
- 153 ページの「領域と RDBMS の起動」

すべて構成し終えたあと、必要であれば、データベースのユーザー出口ルーチンをカスタマイズできます。詳細は、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。

RDBMS とともに Sun MBM を使用する場合、『Sun Mainframe Batch Manager ソフトウェア 構成ガイド』を参照してください。

RDBMS 環境変数の設定

各データベースには、それぞれ設定する必要がある環境変数があります。これは、Sun MTP の実行可能ファイルを構築してデータベースを使用する前に設定する必要があります。環境変数を設定する手順については、第 2 章を参照してください。また、RDBMS のマニュアルも参照してください。

サンプルアプリケーションの Readme ファイルには、使用しているデータベースとともにサンプルアプリケーションを実行するために必要な環境変数についての説明が記載されています。このサンプルアプリケーションは、\$UNIKIX/examples ディレクトリにあります。

kixinstall を使用した RDBMS の選択

データベースに環境変数を設定したあと、データベースを選択し、構成ユーティリティで Sun MTP 実行可能ファイルを作成するには、次の手順に従います。

▼ RDBMS を選択する

1. \$UNIXIX/src ディレクトリに移動します。
2. kixinstall ユーティリティを実行します。
3. 「Configuration Utility」メインメニューで、オプション 2 (RDBMS) を選択し、Return キーを押します。
4. 「RDBMS Configuration」画面 (図 13-1) が表示されるため、使用する RDBMS の数を入力します。

複数の RDBMS を選択できます。

すべての RDBMS の選択を解除するには、R を押します。

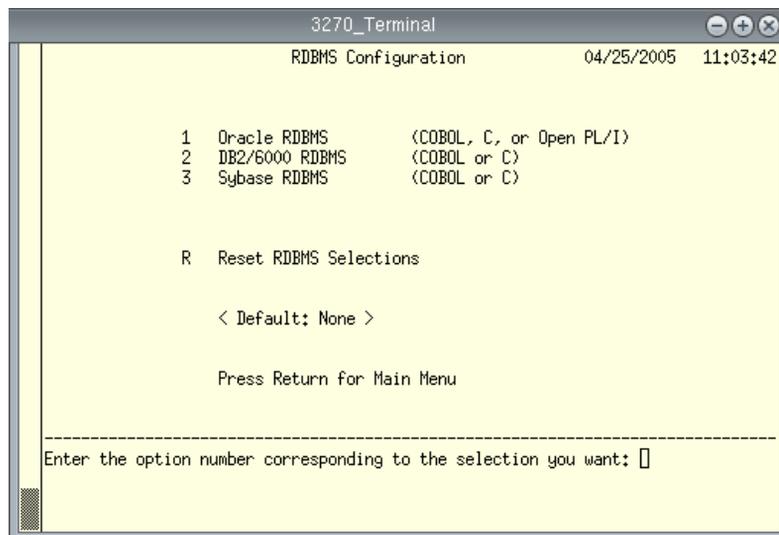


図 13-1 RDBMS Configuration 画面

注 - RDBMS の環境変数が設定されていない場合、それらを設定しなければ処理を続行できないことを示すエラーが表示されます。

5. Return キーを押して、「Configuration Utility」メインメニューを表示します。
6. 『Sun Mainframe Transaction Processing ソフトウェア インストールガイド』の説明にあるように、構成プロセスを完了します。

SIT での RDBMS 値の指定

Sun MTP は、領域の初期化時にデータベース固有の情報を SIT 内で検索する機能をユーザー出口モジュールで提供します。このデフォルトの方法を使用しない場合は、ユーザー出口をカスタマイズする必要があります。その場合、この節は省略できます。

▼ SIT で RDBMS 固有の値を設定する

1. Table Manager を開始し、「Standard Tables」メニューを表示します。
2. PF1 キーを押して、SIT を表示します。

```
3270 Session
System Initialization Table      04/20/2005 08:44:17

Application name:                [ ]
SysID:                          [ ]
Accounting:                      N (Y=Yes, N=No)
File ID:                         01
Default Database environment:
RDBMS:   Server:                DbName:
        UserID:                PassWd:  ***** PW Typ: M
SECURITY;PW max&min len: 00     PassWd Life: 0000 PW Fail Lmt: 00
Common Work Area (CWA) length: 000000
Timeout Period (in minutes):    000000
Start Termid:                   C000
Default Date Format:             M (Y=YYMMDD, M=MMDDYY, D=DDMMYY)
Default TOR:                    [ ]
BMS Alignment:                  UNALIGNED (UNALIGNED/ALIGNED)
BMS  Key Command   Key  Command   Key  Command   Key  Command
    01  P/C        07  P/P        08  P/N

-----
PF2=Write to Disk      PF12=Export Table
PF3=Previous Menu     ENTR=Modify
PF11=Import Table
```

図 13-2 システム初期化テーブル (SIT)

3. ご使用の環境に合わせて SIT をカスタマイズしていない場合は、69 ページの「SIT のカスタマイズ」の手順を実行します。
4. 表 13-1 の情報を使用して、適切な RDBMS のフィールドに値を入力します。
RDBMS について次の 4 つのフィールドが、ユーザー出口プログラムで返されます。
ユーザー出口プログラムは、指定したデータベースへの接続の初期設定時に呼び出すことができます。
Server: サーバーの名前。
DbName: リレーショナルデータベースの名前。
UserID: リレーショナルデータベースのユーザー名。
PassWd: リレーショナルデータベースのパスワード。このフィールドは SIT をディスクに保存するときに暗号化されます。
5. PF2 キーを押して、SIT を保存します。
6. PF3 キーを押して、SIT を終了します。

次の表は、各 RDBMS と SIT で必要なフィールドを示します。

表 13-1 SIT の RDBMS 値

テーブルの値/ データベース	Server	UserID	DbName	PassWd
DB2 UDB	不要	不要	必要	不要
Oracle	Remote*	必要	不要	必要
Sybase	必要†	必要	不要	必要

* 遠隔データベース接続にだけ値を指定します。詳細は、152 ページの「Oracle に関する注意」を参照してください。

† マシン名を指定します。

Oracle に関する注意

遠隔の Oracle データベースを使用する場合、SIT にサーバー名が必要です。このサーバーは、TNSNAMES.ORA ファイルに定義されている必要があります。このファイルでのエントリの定義については、Oracle のマニュアルを参照してください。

ローカルの Oracle データベースを使用し、SIT のサーバー名が誤っている場合、領域を起動しようとする、Oracle SQL コード -12154 を受け取り、領域は起動に失敗します。

Sybase に関する注意

Sybase データベースに接続している Sun MTP 領域は、COBOL または C を対象とした Sybase インタフェースのみをサポートできます。C プログラムおよび COBOL プログラムの両方から Sybase データベースにアクセスする必要がある場合は、C 用と COBOL 用の 2 つの領域を作成する必要があります。

領域と RDBMS の起動

▼ 領域と RDBMS を起動する

1. SIT で必要な値をすべて設定します。

RDBMS と連動するトランザクション処理プログラムをすでに構築しているが、SIT を更新していなかった場合、RDBMS に接続しないで領域を起動します。これで、正しい RDBMS 値で SIT を更新できます。詳細は、153 ページの「起動時に RDBMS をバイパスする」を参照してください。

2. この領域を起動します。

▼ 起動時に RDBMS をバイパスする

1. KIXRDBMS_BYPASS 環境変数を任意の値に設定します。

次に例を示します。

```
$ export KIXRDBMS_BYPASS=1
```

2. この領域を起動します。

このバイパスが有効なことを示す情報メッセージが、unikimain.log に書き込まれます。

3. Table Manager を開き、SIT を表示します。
4. 「Server」フィールドの値を空白に置き換えます。
5. SIT を保存します。
6. この領域を停止します。

7. KIXRDBMS_BYPASS 環境変数を無効にします。
次に例を示します。

```
$ unset KIXRDBMS_BYPASS
```

8. この領域を起動します。

第14章

DBCS のサポートの有効化

Sun MTP は、2 バイト文字セット (DBCS) を対象とした次の機能をサポートしています。

- BMS アセンブラによる PS と SOSI 属性の処理
- トランザクション処理プログラムによる PS と SOSI 属性の 3270 データストリームの作成
- フィールドに PS または SOSI 属性が含まれる場合、Sun MTP ローカルクライアントによる DBCS 文字の受け入れ
- TN 3270 サーバーによる DBCS 3270 データストリームの処理
- システム間通信 (ISC)
- PU4/PU5 接続のサポート

この章は次のトピックで構成されており、DBCS のサポートについて説明します。

- 155 ページの「前提条件」
- 156 ページの「領域での DBCS の有効化」
- 157 ページの「DBCS と ISC の使用」
- 163 ページの「DBCS コードページ変換テーブル」
- 164 ページの「使用上の考慮点」

前提条件

DBCS を使用する前に、次のことを実施する必要があります。

- 使用しているオペレーティングシステムに各国語サポート (NLS) があることを確認します。
- データベースやソートプログラムなど、使用している Sun 以外の製品が DBCS 環境に適合していることを確認します。
- 使用しているプログラミング言語のコンパイラの DBCS オプションを理解します。

領域での DBCS の有効化

次の手順では、領域の 2 バイト文字セット (DBCS) のサポートを有効にする方法について説明します。ISC を使用している場合は、157 ページの「DBCS と ISC の使用」を参照してください。

▼ DBCS のサポートを有効にする

1. Sun MTP ライセンスが DBCS の使用を認めていることを確認します。
ライセンスファイルを作成する手順については、『Sun Mainframe Transaction Processing ソフトウェア インストールガイド』を参照してください。
2. 日本語環境を使用している場合は、領域の設定ファイルで LANG 環境変数を ja_JP.PCK に設定します。

```
LANG=ja_JA.PCK;export LANG
```

3. 領域設定ファイルを探します。
4. 適切なオプションを付けて、領域を起動します。

DBCS 機能を有効にし、使用する変換テーブルの名前を指定するには、unikixmain (kixstart) に -B オプションを使用する必要があります。たとえば、日本語のカタカナ - 漢字の DBCS 変換 (EBCDIC から ASCII、ASCII から EBCDIC) を有効にするには、次のオプションを指定して領域を起動します。

```
$ kixstart -B IBM-930toIBM-932.table other-options
```

注 - 起動時、\$UNIKIX/lib ディレクトリで変換テーブルが検索されます。

すべての unikixmain オプションについては、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』、および 163 ページの「DBCS コードページ変換テーブル」を参照してください。

DBCS と ISC の使用

EBCDIC データが必要な遠隔の IBM CICS 領域に、ISC を介してデータを送信する場合、この節の手順に従ってリソースのデータ変換テンプレートを定義する必要があります。

注 - トランザクション経路指定を使用している場合は、データ変換テンプレートを指定する必要はありません。

データ変換テンプレートの定義の準備

データ変換テンプレートテーブル (CVT) で変換テンプレートを定義する前に、必要なコピーブックを参照に使用できるようにします。

次の例は、図 14-3 と図 14-4 で定義されている DBACFIL リソースのマッピングが含まれているコピーブックを示しています。

コード例 14-1 データセットのリソースのコピーブック

```
VSAM files:DBACFIL - Copy book DBACREC -  
  
* DBACREC - ACCOUNT FILE RECORD  
02 DBACDO          PIC X(5) .          # SBCS - KEY  
02 SNAMEDO        PIC X(18) .         # DBCS  
02 FNAMEO         PIC X(12) .         # DBCS  
02 MIDO           PIC X .             # SBCS  
02 TTLDO         PIC X(4) .          # SBCS  
02 TELDO         PIC X(10) .         # SBCS  
02 ADDR1DO       PIC X(24) .         # Mixed SBCS & DBCS  
02 ADDR2DO       PIC X(24) .         # Mixed SBCS & DBCS  
02 ADDR3DO       PIC X(24) .         # Mixed SBCS & DBCS  
02 AUTH1DO       PIC X(32) .         # SBCS  
02 AUTH2DO       PIC X(32) .         # SBCS  
02 AUTH3DO       PIC X(32) .         # SBCS  
02 AUTH4DO       PIC X(32) .         # SBCS  
02 CARSDO        PIC X .             # SBCS  
02 IMODO         PIC X(2) .          # SBCS  
02 IDAYDO        PIC X(2) .          # SBCS  
02 IYRDO         PIC X(4) .          # SBCS  
02 RSNDO         PIC X .             # SBCS  
02 CCODEO        PIC X .             # SBCS  
02 APPRDO        PIC X(4) .          # DBCS
```

コード例 14-1 データセットのリソースのコピーブック (続き)

02	SCODE1DO	PIC X.	# SBCS
02	SCODE2DO	PIC X.	# SBCS
02	SCODE3DO	PIC X.	# SBCS
02	STATDO	PIC X(2).	# SBCS
02	LIMITDO	PIC X(8).	# SBCS
02	PAY-HIST OCCURS 3.		
04	BAL	PIC X(8).	# SBCS
04	BMO	PIC 9(2).	# SBCS
04	BDAY	PIC 9(2).	# SBCS
04	BYR	PIC 9(4).	# SBCS
04	BAMT	PIC X(8).	# SBCS
04	PMO	PIC 9(2).	# SBCS
04	PDAY	PIC 9(2).	# SBCS
04	PYR	PIC 9(4).	# SBCS
04	PAMT	PIC X(8).	# SBCS

▼ データ変換テンプレートを定義する

1. Table Manager を開いて、「Extended Tables」メニューを表示します。
2. PF9 キーを押して、CVT を開きます。

次の図は、データセットの CVT のエン트리と、キーのエントリを示しています。

Type	Name	Seq No	Group
FC	DBACIX	001	
FC	DBACFIL	K	
FC	DBACFIL	001	
FC	DBACIX	K	

PF2=Write to Disk PF5=Delete Entry PF9=Details ENTR=Modify
 PF3=Previous Menu PF7=Previous Page PF11=Import Table
 PF4=Insert Entry PF8=Next Page PF12=Export Table

図 14-1 データ変換テンプレートテーブル (CVT)

3. PF4 キーを押して、挿入画面を表示します。

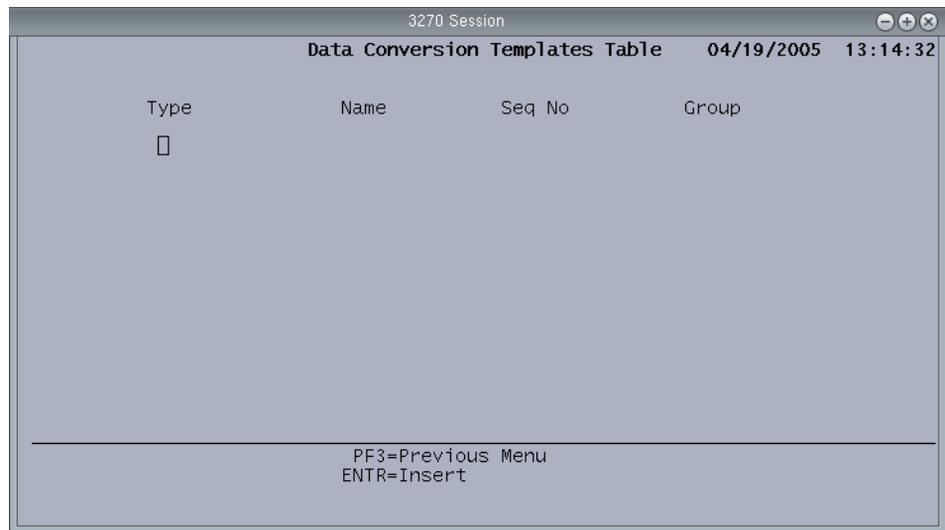


図 14-2 CVT の挿入画面

4. エントリを挿入します。

a. 「Type」フィールドに値を入力します。

- 2 バイト文字のみを変換する場合、CVT 詳細画面の「Type」フィールドで値 D を指定します。または、50 ~ 80 の値を指定できます。
- 1 バイト文字と 2 バイト文字が混在する文字列を変換する場合、CVT 詳細画面の「Type」フィールドで 50 ~ 80 の値を指定する必要があります。

b. リソースの名前を入力します。

c. シーケンス値を入力します。

有効なシーケンス値は、001 ~ 254、K、および D です。

d. 必要に応じて、グループ名を入力します。

e. Return キーを押して、別のエントリを追加します。

5. エントリの作成が終了したら、Enter キーを押してエントリをテーブルに挿入します。

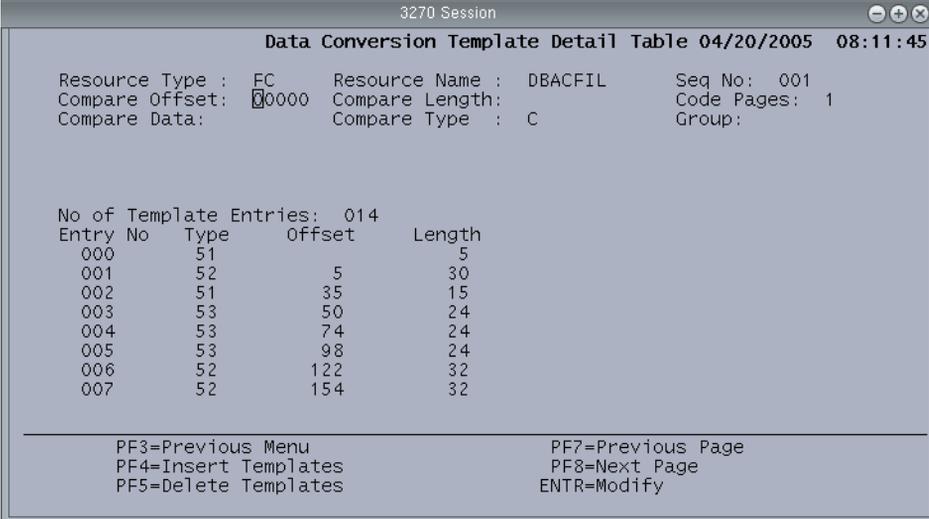
6. CVT のメイン画面で、リソース名を選択して PF9 キーを押します。

7. 選択したエントリの詳細情報を入力し、Enter キーを押して、その情報を挿入します。

8 個以上のエントリを追加する場合は、Enter キーを押して最初の画面を保存し、PF4 キーを押します。2 番目の画面が表示され、さらにエントリを追加できます。

図 14-3 は、CVT 詳細画面の最初のページを示しています。この画面には、DBACFIL リソースの完全なエントリのセットがあります。この図と図 14-4 のエントリでは、「Type」フィールドの数値が使用されます。51 は 1 バイト文字を表し、52 は 2 バイト文字を表しているほか、53 は SOSI 文字のある、1 バイト文字と 2 バイト文字が混在する文字を表しています。これらのエントリは、コード例 14-1 に示されているコピーブックの情報に対応します。

「Type」フィールドに 50 ~ 80 の値を指定する場合は、\$UNIXIX/src/convert/kxcvtxlt.c ユーザー出口で変換をコード化する必要があります。ユーザー出口の詳細は、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』を参照してください。



3270 Session

Data Conversion Template Detail Table 04/20/2005 08:11:45

Resource Type : FC Resource Name : DBACFIL Seq No: 001
Compare Offset: 0000 Compare Length: Code Pages: 1
Compare Data: Compare Type : C Group:

No of Template Entries: 014

Entry No	Type	Offset	Length
000	51		5
001	52	5	30
002	51	35	15
003	53	50	24
004	53	74	24
005	53	98	24
006	52	122	32
007	52	154	32

PF3=Previous Menu PF7=Previous Page
PF4=Insert Templates PF8=Next Page
PF5=Delete Templates ENTR=Modify

図 14-3 CVT 詳細画面 - ページ 1

図 14-4 は、DBACFIL リソースの CVT 詳細画面の 2 番目のページを示しています。

3270 Session

Data Conversion Template Detail Table 04/20/2005 08:12:12

Resource Type : FC Resource Name : DBACFIL Seq No: 001
Compare Offset: 00000 Compare Length: Code Pages: 1
Compare Data: Compare Type : C Group:

No of Template Entries: 014

Entry No	Type	Offset	Length
008	52	186	32
009	52	218	32
010	51	250	11
011	52	261	4
012	51	265	13
013	51	278	120
014			
015			

PF3=Previous Menu PF7=Previous Page
PF4=Insert Templates PF8=Next Page
PF5=Delete Templates ENTR=Modify

図 14-4 CVT 詳細画面 - ページ 2

8. 作業が終了したら、PF3 キーを押して CVT のメイン画面に戻ります。
9. 定義したリソースに関連付けられているキーがある場合は、シーケンス番号が K であるリソースを選択し、PF9 キーを押します。
10. キーのテンプレートを定義します。
前に定義したリソースと同じように、値を入力します。

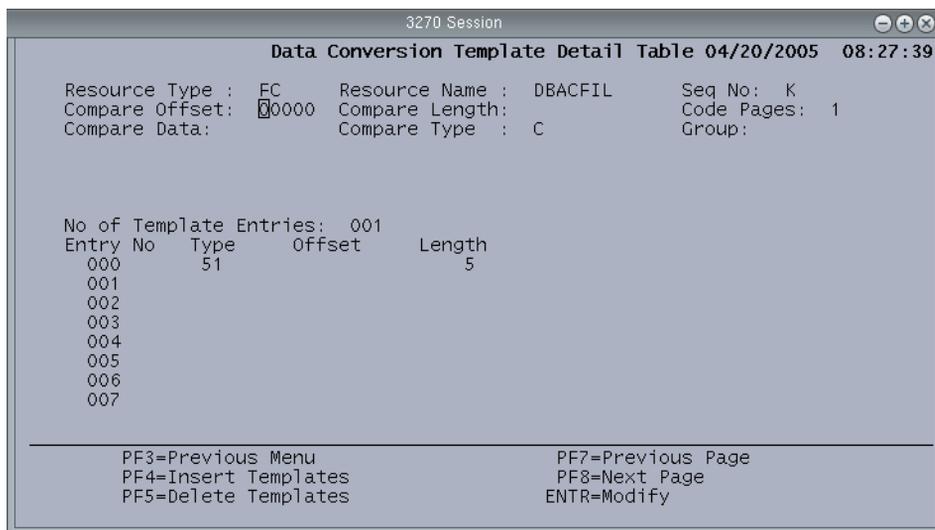


図 14-5 CVT 詳細画面 - キーの定義

11. 作業が終了したら、Enter キーを押してエントリを完成します。
12. PF3 キーを押して、CVT のメイン画面に戻ります。
13. CVT へのデータ入力の終了時、PF2 キーを押してテーブルを保存します。
14. Table Manager を終了します。
15. 領域を停止して再起動し、変更を有効にします。

CVT の詳細については、『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

トランザクション経路指定

トランザクション経路指定を DBCS 環境で使用するときには、使用している端末の種類によって、Terminal Control Table - System Entries 画面での遠隔システムの構成方法が決まります。

- EBCDIC 端末 (TN3270) を使用している場合は、端末所有領域と、アプリケーション所有領域またはファイル所有領域との、両方の領域の TCT の「Ptrn CdPge」フィールドに値を指定する必要があります。こうすると、領域間で EBCDIC が渡されます。
- ASCII 端末 (ローカルクライアント) を使用している場合は、両方の領域の「Ptrn CdPge」フィールドは空白にする必要があります。こうすると、領域間で ASCII が渡されます。

注 - TN3270 端末とローカルクライアントの両方を使用して領域間でのトランザクション経路指定を行うことはできません。どちらかの端末タイプを使用してください。

DBCS コードページ変換テーブル

DBCS をサポートするには、次の変換テーブルのいずれかを使用します。このテーブルは、\$UNIX/lib ディレクトリに格納されています。

表 14-1 DBCS 変換テーブル

変換テーブル	言語	文字セット
IBM-930toIBM-932.table IBM-930toIBM-943.table IBM-930toIBM-eucJP.table	日本語カタカナ漢字	DBCS (16 ビット)
IBM-939toIBM-932.table IBM-939toIBM-943.table IBM-939toIBM-eucJP.table	日本語英数小文字漢字	DBCS (16 ビット)
IBM-290toJIS-Roman.table IBM-1027toJIS-Roman.table	日本語	SBCS (8 ビット)
IBM-933toIBM-934.table IBM-933toIBM-eucKR.table	韓国語	DBCS (16 ビット)
IBM-935toIBM-1381.table IBM-935toIBM-eucCN.table	簡体字中国語	DBCS (16 ビット)
IBM-937toIBM-938.table IBM-937toIBM-eucTW.table IBM-937tobig5.table	繁体字中国語	DBCS (16 ビット)

使用上の考慮点

アプリケーションで DBCS データを使用する場合、次の問題に注意してください。

- DBCS データは、SEND MAP や RECEIVE MAP などのマッピング操作でだけサポートされています。データストリームの raw sends では (SEND コマンドの使用)、2 バイト文字を SO と SI バイトの間に埋め込む必要があります。
- 独自のコードページ変換テーブルを作成する場合、それらは \$UNIKIX/lib ディレクトリに置く必要があります。

DBCS コンパイラのオプションとデータ型については、使用しているプログラミング言語のマニュアルを参照してください。

付録 A

キーボードの割り当て

この付録では、3270 機能と同等の機能を実行するキーボードに関する情報を提供します。

- 166 ページの「NCD 101 キーキーボード」
- 167 ページの「NCD 97 キーキーボード」
- 168 ページの「VT タイプのキーボード」
- 169 ページの「Sun のキーボード」
- 170 ページの「デフォルトの X キーボード」
- 171 ページの「デスクトップ端末キーボード」

Sun MTP に付属する次のキーファイルは、\$UNIX/lib にあります。

dtterm.key	ncd101.key	sun.key
xterms.key	ncd97.key	Åúsun-cmd.keyÅö
xterm.key	ncdvt.key	vt200.key
vt320.key	vt100.key	wy50.key

これら .key ファイルのほとんどは、TERM 環境変数の値に基づいて、自動的に使用されます。ncdxx.key ファイルは、

TERM=xterms と一緒に使用して、X 端末で使用するキーボードを指定します。これらのキーファイルは、xxxstart および xxunikix ユーティリティーで使用されます。これらのユーティリティーと kixkeydef ユーティリティーについては、

『Sun Mainframe Transaction Processing ソフトウェア リファレンスマニュアル』を参照してください。

これらのユーティリティーを使わず、TERM=xterms または TERM=xterm の場合、選択されているデフォルトのキーボードは、NCD 101 キーキーボードです。キーファイルについては、85 ページの「キーファイル」で説明しています。

NCD 101 キーキーボード

NCD 101 キーキーボードを使用する TERM=xterms のキーボード割り当てです。

3270 機能	キーボードの説明
UP	上矢印
DOWN	下矢印
LEFT	左矢印
LEFT	バックスペース
RIGHT	右矢印
TAB	タブ
BTAB	シフト - タブ
ERAEOF	Page Down
ERAINP	Page Up
INSMODE	挿入
DEL	削除
ENTER	キーパッドの Enter
CLEAR	キーパッドの加算 (+)
RESET	キーパッドの減算 (-)
HOME	Home
REFRESH	Control-l (小文字 L)
PF1 ~ PF12	F1 ~ F12
PA1	キーパッドの除算 (/)
PA2	キーパッドの乗算 (*)
PA3	改行
PRINT	End

NCD 97 キーキーボード

NCD 97 キーキーボードを使用する TERM=xterms のキーボード割り当てです。

3270 機能	キーボードの説明
UP	上矢印またはキーパッドの 8
DOWN	下矢印またはキーパッドの 2
LEFT	左矢印、バックスペース、またはキーパッドの 4
RIGHT	右矢印またはキーパッドの 6
TAB	タブ
BTAB	シフト - タブ
ERAEOF	キーパッドの 3
ERAINP	キーパッドの 9
INSMODE	キーパッドの 0
DEL	キーパッドの小数点 (.) または削除
ENTER	キーパッドの Enter
CLEAR	キーパッドのセパレータ (,)
RESET	キーパッドの減算 (-)
HOME	キーパッドの 7
REFRESH	PF4 または Control-I (小文字 L)
PF1 ~ PF12	F1 ~ F12
PA1 ~ PA3	PF1 ~ PF3
PRINT	キーパッドの 1

注 - 数値入力キーは、数値を生成しません。

VT タイプのキーボード

TERM=vt100、vt200、または vt320 のキーボード割り当てです。

3270 機能	キーボードの説明
UP	上矢印
DOWN	下矢印
LEFT	左矢印
LEFT	バックスペース
RIGHT	右矢印
TAB	タブ
BTAB	シフト - タブ
ERAEOF	次の画面
ERAINP	削除
INSMODE	検索
DEL	選択
ENTER	キーパッドの Enter
CLEAR	キーパッドのセパレータ (,)
RESET	キーパッドの減算 (-)
HOME	ここに挿入
REFRESH	PF4 または Control-I (小文字 L)
PF1 ~ PF12	F1 ~ F12
PA1 ~ PA3	PF1 ~ PF3
PRINT	前の画面

Sun のキーボード

TERM=sun または TERM=sun-cmd のキーボード割り当てです。

3270 機能	キーボードの説明
UP	キーパッドの上矢印
DOWN	キーパッドの下矢印
LEFT	バックスペースまたは削除
RIGHT	キーパッドの右矢印
TAB	タブ
BTAB	シフト - タブ
RETURN	リターンまたは改行
HOME	Home
INSERT	挿入
DEL	削除
ENTER	Enter
CLEAR	キーパッドの加算 (+)
RESET	キーパッドの減算 (-)
REFRESH	Control-I (小文字 L)
PF1 ~ PF12	F1 ~ F12
PF1 ¹	一時停止
PF13 ~ PF24	エスケープ - F1 ~ エスケープ - F12 ²
PF13	エスケープ - 一時停止
PA1	キーパッドの等号 (=)
PA2	キーパッドの除算 (/)
PA3	キーパッドの乗算 (*)

1. Sunview では、一時停止には PF1 キーを使用する必要があります。

2. エスケープキーに続けて、目的のファンクションキーを押します。

デフォルトの X キーボード

TERM=xterms または TERM=xterm のキーボード割り当て (指定キーボードなし) です。

注 - このキーボード定義は、ncd101unikix または ncd101start を実行するとき
に使用する NCD 101 キーボード定義と同じです。

3270 機能	キーボードの説明
UP	上矢印
DOWN	下矢印
LEFT	左矢印
LEFT	バックスペース
RIGHT	右矢印
TAB	タブ
BTAB	シフト - タブ
ERAEOF	Page Down
ERAINP	Page Up
INSMODE	挿入
DEL	削除
ENTER	キーパッドの Enter
CLEAR	キーパッドの加算 (+)
RESET	キーパッドの減算 (-)
HOME	Home
REFRESH	Control-l (小文字 L)
PF1 ~ PF12	F1 ~ F12
PA1	キーパッドの除算 (/)
PA2	キーパッドの乗算 (*)
PA3	改行
PRINT	End

デスクトップ端末キーボード

TERM=dtterm のキーボード割り当て (指定キーボードなし) です。

3270 機能	キーボードの説明
UP	上矢印
DOWN	下矢印
LEFT	左矢印
LEFT	バックスペース
RIGHT	右矢印
TAB	タブ
BTAB	シフト - タブ
ERAEOF	Page Down
ERAINP	Page Up
INSMODE	挿入
DEL	削除
ENTER	キーパッドの Enter
CLEAR	キーパッドの加算 (+)
RESET	キーパッドの減算 (-)
REFRESH	Control-l (小文字 L)
PF1 ~ PF10	F1 ~ F10
PA1	キーパッドの除算 (/)
PA2	キーパッドの乗算 (*)
PRINT	End

用語集

数字

- 1 バイト文字セット (SBCS) (名詞) 文字ごとに1バイトを必要とする言語スクリプトは1バイト文字セット (SBCS) と呼ばれます。たとえば、英語、スペイン語、およびフランス語が1バイト文字セット (SBCS) です。
- 2 バイト文字セット (DBCS) (名詞) 一部の言語スクリプトでは、文字を表すのに2バイトが必要です。そのスクリプトは2バイト文字セット (DBCS) と呼ばれます。たとえば、日本語、中国語、および韓国語が2バイト文字セット (DBCS) です。
- 2 フェーズコミット (名詞) トランザクションとその完了の回復モデル。フェーズ1で、TM はすべての RM にトランザクションブランチをコミットする準備 (あるいはトランザクションブランチの準備) を行うように求めます。フェーズ2で、TM はすべての RM に対し、場合に応じてトランザクションブランチのコミットまたはロールバックを行うように実際の要求を出します。すべての RM は、共有リソースに対する変更のコミットまたはロールバックを行い、続いて TM に状態を返します。これによって、TM はそのグローバルトランザクションに関するその情報を破棄できます。¹
- 3270 SNA デバイス (名詞) IBM SNA 3270 データストリームを表示するターミナルデバイス。

A

ABEND (名詞) タスクを異常終了するのに使用される EXEC CICS コマンド。

1. 『The XA Specification』、p.8

B

Berkeley Software Distribution (BSD)

(名詞) バークレイのカルフォルニア大学によって配布された、UNIX の仕様または実装状態を指します。BSD という略語には、通常、配布された特定の UNIX バージョンを示す番号があとに付きます (たとえば、BSD 4.3)。多くのベンダーが、UNIX 製品の標準として BSD 仕様を使用します。

C

COMMAREA

(名詞) 通信領域。指定した端末と通信するタスク間で、データの受け渡しに使用される領域です。領域を使用して、タスク内のプログラム間でデータを受け渡すこともできます。

CPRT

(名詞) Sun MTP 領域のデフォルトのプリンタの名前。

E

EBCDIC

(名詞) 拡張 2 進化 10 進コード。多くのデータ処理システム、データ通信システム、および関連装置で情報交換に使用される、8 ビット符号化文字から構成された符号化文字セットです。

L

LU6.2

(名詞) 分散処理環境のプログラム間での一般的な通信をサポートする論理ユニットタイプ。

M

make 機能

(名詞) Sun MTP システムの再構築に使用されるユーティリティ。

MVS (名詞) 複数の仮想記憶域。CICS/MVS が実行される IBM メインフレームのオペレーティングシステム。

S

SOSI フィールド (名詞) SOSI (Shift-Out、Shift-In) フィールドは、1 バイト文字と 2 バイト文字の両方を含んだ 3270 フィールドです。SOSI フィールドのすべての 2 バイト文字は、SOSI 文字で挟まれている必要があります。

SQL (名詞) 構造化照会言語。一連の情報へのアクセスと更新に使用されるリレーショナルデータベース言語です。

Sun Mainframe Batch Manager ソフトウェア (Sun MBM)

(名詞) 制御された環境でバッチジョブを実行するための機能を提供するバッチマネージャー製品。Sun MBM は、バッチ生産負荷を処理し、開始時刻やバッチプロセスの最大数、およびジョブの優先順位といった割り当てられたパラメータによってジョブをスケジューリングします。

Sun Mainframe Transaction Processing ソフトウェア (Sun MTP)

(名詞) プロセス間通信サービス、ソケット、COBOL、C、PL/I などの機能を使用してアプリケーションを実行するユーザーアプリケーション。クライアント以外の Sun MTP のすべてのコンポーネントは、メインサーバープロセスである unikixmain によって起動します。

Sun MTP グループ

(名詞) 特定のアプリケーションのテーブルファイルセット。ファイルはファイルシステムの単一のディレクトリに配置されています。ディレクトリは GCT にリストされます。

Sun MTP シェルスクリプト

(名詞) \$UNIKIX/bin に配置されたシェルスクリプトのユーティリティプログラム。

Sun MTP 領域

(名詞) システム上の異なるアプリケーションを定義するプロセス変数、リソース変数、および環境変数のセット。

T

Table Manager (名詞) Sun MTP テーブルで領域のリソースの定義に使用する Sun MTP 機能。

TCP/IP (名詞) インターネットの基礎となるネットワークプロトコル群。伝送制御プロトコル (TCP) は、信頼性の高い全二重データストリームを提供するプロトコルです。インターネットプロトコル (IP) は、TCP のパケット配信サービスを提供するプロトコルです。TCP プロトコルは、ユーザープロセスではなく、IP と連携します。

TCTUA (名詞) 端末ユーザー領域。端末に関連するトランザクション間で、データの受け渡しに使用されます。

TN3270 サーバー
(unikixtnemux) (名詞) Sun MTP で、TCP/IP - TN3270 プロトコルを使用して、PC、Macintosh、および UNIX システムで実行する 3270 エミュレータのサポートを有効にします。TN3270E もサポートします。

TN3270 プロトコル (名詞) 従来の TCP/IP Telnet プロトコルの拡張で、ASCII 以外の文字、IBM-3270 などのブロックモードデバイス、Sun MTP などのアプリケーションで、TCP/IP を介した通信を可能にします。TN3270E も含まれます。

U

unikixdcl サーバー (名詞) DCL プロトコルスタックを使用する SNA サーバーに対する遠隔接続の数および状態を監視する Sun MTP サーバー。

unikixmain サーバー (名詞) Sun MTP メインサーバープロセス。

unikixqm サーバー (名詞) WebSphere MQ 接続の数と状態を監視する Sun MTP サーバー。

unikixrc.cfg
ファイル (名詞) unikixdcl、unikixqm、および unikixtnemux サーバーについての情報を収めたりソースファイル。起動時に、Sun MTP 通信マネージャーは、unikixrc.cfg ファイルを読み取り、該当するサーバーを起動します。

unikixtcp サーバー (名詞) TCP/IP 接続の数と状態を監視する Sun MTP サーバー。

unikixtnemux
サーバー (名詞) TN3270 サーバープロセス。「TN3270 サーバー」を参照。

V

VSAM クラスタ (名詞) VSAM 構造での最上位の命名。クラスタの名前は主アクセス名です。クラスタには、データ定義、索引定義、および任意の二次索引 VSAM 構成テーブル (VCT)

(名詞) 基本の Sun MTP 構成パラメータを定義する制御テーブル。

VSAM データセット (名詞) VSAM 規則に従って編成、格納、およびアクセスされる関連データの名前付き集合。

VSAM ファイル 「VSAM データセット」を参照。

あ

アカウントティング (名詞) ユーザーのアカウント情報を体系的に収集、記録、解釈、および表示する方法。

アカウントティング
ジャーナル (名詞) 関連付けられたジャーナルのアカウントティングレコードを Sun MTP が書き込むファイル。ジャーナルファイル名は物理ファイル名に対応します。

オブジェクトの所有者が任意の ACL をコントロールします。

アクティビティー
カウント (名詞) Sun MTP が保全性の管理に使用する方法。アクティビティーカウントは、各 VSAM ファイルヘッダーに置かれ、領域がファイルを開閉するたびに増加します。

宛先管理テーブル (DCT) (名詞) 一時データコマンドで処理される、宛先の名前またはキューを含んだ Sun MTP テーブル。テーブルは、これらのキューに割り当てられた特性の定義や管理に使用されます。

アプリケーションプログラ
ミングインタフェース
(API) (名詞) アプリケーションプログラムで使用される事前定義のインタフェース。API はルーチン名とルーチンの引数から構成され、関連付けられたアプリケーションプログラム言語の構文に従います。

い

一時記憶域テーブル
(TST) (名詞) ローカルおよび遠隔の一時記憶域キューの記憶域と回復を定義する Sun MTP テーブル。

インターネット
プロトコル (IP) 「TCP/IP」を参照。

か

外部表示インタフェース (EPI)

(名詞) CICS 以外のアプリケーションプログラムが、1 つ以上の標準 3270 端末として Sun MTP を表示することを可能にするプログラムを作成するための API。EPI アプリケーションは、実際の 3270 端末のように Sun MTP と通信します。

外部呼び出し インタフェース (ECI)

(名詞) DPL の規則に従いサーバーで実行中の CICS プログラムを、CICS 以外のアプリケーションプログラムが呼び出すことを可能にするプログラムを作成するための API。

会話型トランザクション

(名詞) トランザクションが有効な間、ユーザーとの会話 (通常、SEND/RECEIVE のシーケンス) が行われるトランザクション。

仮想記憶アクセス方式 (VSAM)

(名詞) さまざまなアクセス方式によってレコードにアクセスする方式。

ESDS (入力順データセット)。レコードは順次に記録され、アクセスされません。

RRDS (相対レコードデータセット)。レコードは、データセット内で占める位置番号によって検索されます。

KSDS (キーシーケンスデータセット)。レコードは索引またはキーによって検索されます。

仮想通信アクセス方式 (VTAM)

(名詞) 通信を制御し、SNA ネットワーク内のデータの流れを制御するプログラム。

カタログファイル

(名詞) VSAM データセットの名前と情報を含むファイル。

環境変数

(名詞) プログラムファイルおよびアプリケーションの位置を定義する変数。クライアントとサーバーは、どちらも環境変数を使用します。

監視管理テーブル (MCT)

(名詞) 領域で有効状態にあるアカウントिंगのデフォルトタイプ (トランザクションとユーザーロギング) を制御する Sun MTP テーブル。このテーブルには、アカウントINGを制御するフラグが付いた単一のエントリが含まれません。

き

キーシーケンスデータ
セット (KSDS)

(名詞) キーによって参照される可変長レコードの索引編成 VSAM ファイル。

機能シップ

(名詞) アプリケーションプログラムに透過的なプロセス。リソースがほかの CICS システムに実際に配置されている際に、このプロセスを使って CICS はそのリソースにアクセスします。

基本マッピングサポート
(BMS)

(名詞) データストリームを端末とやり取りする機能。入出力表示データをフォーマットします。BMS マクロ命令は Sun MTP BMS アセンブラで使用され、物理および記号定義のマッピングファイルを作成します。

共通作業領域 (CWA)

(名詞) システム内のタスクからのアクセスが必要なユーザーデータに対して、アプリケーションプログラムによって使用可能なタスク共有プール内の領域。

く

クラスタ

「VSAM クラスタ」を参照。

グループ

(名詞) 特定のアプリケーションのテーブルファイルセット。ファイルはファイルシステムの単一のディレクトリに配置されています。ディレクトリは GCT に定義されます。

グループ管理テーブル
(GCT)

(名詞) グループを定義する Sun MTP テーブル。各グループは、特定のアプリケーションの情報を含んだファイルシステムのディレクトリを定義します。

グローバル
トランザクション

(名詞) 同じ作業単位を支援するために複数のリソースマネージャーが動作するトランザクション。

こ

コードページ

(名詞) 16 進数の値とグリフとのマッピングテーブル。たとえば、ASCII のコードページ ISO8859-1 は 0x41 を使用して文字「A」を表し、EBCDIC のコードページ IBM-1047 は 0xC1 を使用して文字「A」を表します。

顧客情報管理システム
(CICS)

(名詞) ユーザー作成のアプリケーションプログラムによって、遠隔端末で入力されたトランザクションの並行処理を可能にする IBM の使用許諾を受けたプログラム。データベースの構築、使用、維持の各機能が含まれます。

さ

サインオンテーブル
(SNT)

(名詞) Sun MTP トランザクションを使用する認証されたユーザーのリストを含んだ Sun MTP テーブル。

し

システム間通信 (ISC)

(名詞) TCP/IP や SNA ネットワーキング機能、または SNA アクセス方式のアプリケーション間機能を使った別個のシステム間の通信。

システムサービス
制御ポイント (SSCP)

(名詞) 構成管理、ネットワークオペレータ機能および問題判定要求の調整、およびネットワークのエンドユーザーに対するディレクトリサポートやその他のセッションサービスの提供のための、SNA ネットワーク内の制御点。

システム初期化テーブル
(SIT)

(名詞) システムの初期化情報を含み Sun MTP システム名を識別する Sun MTP テーブル。

システムネットワーク
体系 (SNA)

(名詞) 情報単位を伝達し、ネットワークの構成と動作を制御するための論理構造、形式、プロトコル、および操作順序。

実行インターフェース
ブロック

(名詞) CICS プログラム内の各タスクに関連する制御ブロック。EIB には、アプリケーションプログラムの実行中に役立つ情報 (トランザクション識別子など) と、プログラムのデバッグのためにダンプを使用する際に役立つ情報が収められます。

ジャーナル管理テーブル
(JCT)

(名詞) トランザクションが、ほかのテーブルで参照されている 1 つ以上のジャーナルファイルを書き込むことができるかを指定する Sun MTP テーブル。ジャーナルはアカウンティングデータの書き込みにも使用されます。

処理プログラムテーブル
(PPT)

(名詞) Sun MTP トランザクションが参照するアプリケーションプログラムと BMS マップセットをリストする Sun MTP テーブル。

せ

セグメント (名詞) スパンファイルの一部分。「スパンファイル」を参照。

そ

**相対レコードデータ
セット (RRDS)**

(名詞) VSAM データセットで、そのレコードはデータセット内で占める位置番号で検索されます。

ソケット

(名詞) 異なるネットワークプロトコルを使用可能にするプロセス間通信の仕組み。

た

ダンプファイル機能

(名詞) Sun MTP からディスクファイルやシステムプリンタに、VSAM データセットのレコードの一部またはすべてを書き込む機能。

端末管理テーブル (TCT)

(名詞) 端末、プリンタ、および遠隔システム接続の識別情報を収めた Sun MTP テーブル。

て

データセグメント

(名詞) クラスタのデータ部分を収めたファイル。

データセット

「VSAM データセット」を参照。

データファイル

(名詞) レコードを収めた 1 つまたは複数のデータブロックから構成されます。

データファイルエディタ

(名詞) VSAM データセットの構築、変更、またはダンプが可能な Sun MTP メニュー。

**データ変換テンプレート
テーブル (CVT)**

(名詞) あるコードセットからほかのコードセットに (たとえば、EBCDIC から ASCII に) データ変換を行うテンプレートを定義する Sun MTP テーブル。ある環境から別の異機種システム混在環境に、データを転送する際に必要になります。

伝送制御プロトコル
(TCP)

「TCP/IP」を参照。

と

同期データリンク制御
(SDLC)

(名詞) 米国規格協会 (ANSI) の拡張データ通信制御手順 (ADCCP) および国際標準化機構 (ISO) のハイレベルデータリンク制御手順 (HDLC) のサブセットに準拠する規定。リンク接続上で、同期をとり透過的なコードで情報をビットごとにシリアル転送する管理に使用します。

同期点

(名詞) アプリケーションプログラムの実行での論理点。この論理点で、プログラムによるデータベースの変更は、整合性があり完全で、データベースにコミットが可能です。この点まで持続した出力は宛先に送信され、入力メッセージキューから削除され、ほかのアプリケーションでのデータベースの更新が可能になります。プログラムが異常終了した場合、回復と再起動の機能は最新の同期点以前の更新をバックアウトしません。

トランザクション
イニシエータサーバー
(unikixtrin)

(名詞) Sun MTP サーバーシステムのエージェントとして動作し、必要な Sun MTP メッセージキューにメッセージを配置します。トランザクション処理プログラムと開始サーバーが、出力メッセージ用に同じデータグラム機構を使用することにより、遠隔クライアントを通して直接対話できるようになります。一つの unikixtrin プロセスは、遠隔の 3270 デバイスクライアントすべてをサポートします。

トランザクションクラス
テーブル (TXC)

(名詞) 領域に定義されたトランザクションクラスに関する情報を含んだ Sun MTP テーブル。

トランザクション
経路指定

(名詞) Sun MTP または CICS 領域と接続した端末による、同じあるいは異なるシステム上のほかの Sun MTP または CICS 領域でトランザクションの実行を可能にします。

に

入力順データセット
(ESDS)

(名詞) データレコードを入力順に格納する可変長 VSAM ファイル。

は

パーティション外キュー

(名詞) 「DCT-Extrapartition Destinations」画面で識別されるキューに書き込まれるすべてのデータを収めた順編成ファイル。ファイルは、エントリで指定したレコード形式と長さで開かれます。

ひ

非同期処理

(名詞) プロセスの継続中に、メインフレームによる Sun MTP 領域でのトランザクション開始を可能にする、あるいは Sun MTP 領域によるメインフレームでのトランザクション開始を可能にする双方向のプロセス。

ふ

ファイル管理テーブル (FCT)

(名詞) Sun MTP アプリケーションプログラムがアクセスする VSAM データファイルについての情報を収めた Sun MTP テーブル。各ファイルには一連の特性が関連付けられます。この特性を Sun MTP のコマンドルーチンで使用して、アプリケーションプログラムが指定したコマンドを検査し実行します。

ファイルシステム

(名詞) 物理ディスクドライブをパーティションと呼ぶ小単位の領域に分割する機能。パーティションには、ファイルシステム、スワップ空間、ブートセクタその他の情報を含めることができます。

ファイルマネージャー

(名詞) VSAM カタログでの VSAM ファイルの定義に使用されます。ファイルタイプ、キーの長さ、サイズなどのファイルの属性を定義できます。

不正終了

(名詞) タスクの異常な終了。アプリケーションは、EXEC CICS ABEND コマンドを実行してタスクを異常終了させることができます。異常終了と同じ意味です。

プログラム管理テーブル (PCT)

(名詞) Sun MTP でトランザクションの識別と初期化に使用する制御情報を収めた Sun MTP テーブル。

プログラムリストテーブル (PLT)

(名詞) システムの起動時、ユーザーの起動時、またはシステムの停止時に Sun MTP によって自動的に開始されるプログラム名を収めた Sun MTP テーブル。

分散トランザクション
処理 (DTP) (名詞) システム間または領域間リンク上で互いに同期通信し合うトランザク
ション間での処理の配布。

分散プログラムリンク
(DPL) (名詞) 領域のプログラムがほかの領域のプログラムに同期リンクするシステム
間通信の方法。

ま

マニュアルページ (名詞) man コマンドを使用して、コマンドの使用方法を表示できます。たとえ
ば、grep コマンドについて表示するときは、プロンプトで man grep と入力
します。

り

領域 「Sun MTP 領域」を参照。

る

ルートファイルシステム (名詞) オペレーティングシステムと関連のファイルが入っています。ルート
ファイルシステムは、完全なファイル名の最初の文字としてスラッシュ (/) を
付けて、参照されます。

ろ

論理ユニット (LU) (名詞) SNA で、エンドユーザーが SNA ネットワークにアクセスして別のエン
ドユーザーと通信するためのポートまたはエンドユーザーがシステムサービス
制御点 (SSCP) によって提供される機能にアクセスするためのポート。

索引

記号

\$KIXSYS/_kix_reserved_maps, 14
\$UNIX/bin
 \$PATH への追加, 16
 リソース定義, 93
\$UNIX/local/bin, 16
/etc/inittab, 93

数字

1 バイト文字のコードページ変換テーブル, 52
2 バイト文字セット。「DBCS」を参照
3270
 Sun MTP の制御機能, 85
 キーボード機能, 91
 「3270 Devices」画面
 フィールド, 121 ~ 122, 129 ~ 130
 ホストとポートの画面, 131
3270 クライアントイニシエータ
 機能, 123
 自動ログイン, 129
 ポート番号, 131
3270 端末の接続, 29
3270 デバイスクライアント
 KIXHOST の設定, 12
 開始方法, 124
 設定, 123
 ログイン, 132
3270 トランザクションイニシエータサーバー, 123

A

ACUCOBOL 環境変数, 9
AIX のメモリーのコマンド, 45
Animator、Remote, 13
ASCII 端末、初期化, 93
ASSIGN コマンド, 70

B

BMS
 SIT の位置合わせフィールド, 73
 TCT のマップの接頭辞のフィールド, 122, 130
 ページングキー値, 73
BMS マップの各国語サポート, 122, 130
BMS マップの代替画面サイズ, 122, 130
Bourne シェル, 21

C

C/C++
 共有ライブラリの環境変数, 12
 プログラム, 63
CEDA トランザクション, 82
CEMT PERFORM SHUTDOWN IMMEDIATE トランザクション, 26
CEMT PERFORM SHUTDOWN トランザクション, 26
CEMT SHUT, YES トランザクション, 26

Clear キー, 4
CMNU トランザクション, 58
COBCPY 環境変数, 9
COBDIR 環境変数, 9
COBMODE 環境変数, 10
COBOL
 ACUCOBOL-GT のファイル名の拡張子, 62
 Micro Focus の作業モード, 10
 Remote Animator, 13
 Server Express のファイル名の拡張子, 62
COBPATH 環境変数, 10
CODE_PREFIX 環境変数, 10
COPYPATH 環境変数, 10
CPRT (デフォルトのプリンタ), 95
CSMT SHUT,NO トランザクション, 25
CSPG トランザクション, 73
CSSF GOODNIGHT トランザクション, 25
CSSF LOGOFF トランザクション, 25
CTBL トランザクション, 58
CVT。 「DBCS」を参照
C シェル, 21

D

DBCS
 ISC 上での EBCDIC データの変換, 157
 KIXCONVTRACE 環境変数, 11
 unikixmain オプション, 156
 実装されている機能, 155
 使用上の考慮点, 164
 前提条件, 155
 変換テーブル, 163
 有効化, 156
DBCS の EBCDIC データの変換, 157
DCLRTERM 環境変数, 10, 110
DCLSTERM 環境変数, 10, 110
dfhusdup ユーティリティー, 82
DFHUSD ファイル, 79, 81
DPL。 「分散プログラムリンク (DPL)」を参照

DTP。 「分散トランザクション処理 (DTP)」を参照
dtterm.key ファイル, 165, 171

E

ECI/EPI クライアント
 サポートされる, 133
 自動インストール, 134
 自動インストールされる最大数, 13
ECI。 「外部呼出しインタフェース (ECI)」を参照
Enter キー, 4

F

FCT。 「ファイル管理テーブル (FCT)」を参照

I

IBM CICS クライアント, 133
IBM CICS ユニバーサルクライアント, 133
IDCAMS DELETE, 146
IDCAMS REPRO, 146
Internet User Datagram ソケット, 123
ISC。 「システム間通信 (ISC)」を参照

J

J3270 端末エミュレータ, 29, 52
Java プログラム, 63

K

keysym 変換, 92
KIX62PASSTHRU 環境変数, 11
KIX_ADJ_DST 環境変数, 11
KIXAPPSEC 環境変数, 18
KIXBIN 環境変数, 11
KIXBTCH 環境変数, 11

KIXCONVTRACE 環境変数, 11
KIXDCTSEC 環境変数, 18
KIXDUMPDIR 環境変数, 12
KIXDYNRTE 環境変数, 12
KIX_ENABLE_JAVA 環境変数, 11
KIXFCTSEC 環境変数, 18
KIXGETMAINDMP 環境変数, 12
KIXHOST 環境変数, 12, 128
KIXJCTSEC 環境変数, 18
KIX_JVM_OPTIONS 環境変数, 12
kixkeydef, 86 ~ 89
KIXLIB 環境変数, 12
KIXLICDIR 環境変数, 12
KIXMAPS 環境変数, 12
KIXMAXIST 環境変数, 13, 116, 133, 134
KIXPCTSEC 環境変数, 18
KIX_PGM_MODE 環境変数, 13
KIX_PGMTXN_MODE 環境変数, 13
KIXPPTSEC 環境変数, 18
kixprint, 102 ~ 105
kixprint の出力の破棄, 103
KIXPROGS 環境変数, 13
KIXRDBMS_BYPASS 環境変数, 13
KIX_READLOCKOFF 環境変数, 13
KIX_READLOCKON 環境変数, 13
KIXREMANIMPORT 環境変数, 13
KIXREMANIMTOUT 環境変数, 14
_KIX_RESERVED_MAPS 環境変数, 14
KIXSECDFLTUSER 環境変数, 18
KIXSEC_LOGGING 環境変数, 18
KIXSECPREFIX 環境変数, 18
KIXSEC_SERVERHOST 環境変数, 19
KIXSEC_SERVERPORT 環境変数, 19
KIXSEC 環境変数, 18
KIXSNAPDIR 環境変数, 14
KIXSNT_NOMEMUPDATE 環境変数, 14
KIXSOCKET_TIMEOUT 環境変数, 14, 135
kixstart
 Sun MTP の起動, 24

 リソースの自動インストール, 82
kixstop, 27
KIXSTTSEC 環境変数, 19
KIXTCTSEC 環境変数, 19
KIXTSTSEC 環境変数, 19
KIXVSAMPOOLS 環境変数, 15, 46
Korn シェル, 21

L

LD_LIBRARY_PATH 環境変数, 15
LIBPATH 環境変数, 15
lp コマンド, 103
LU タイプ 1 プリンタ、構成, 100

M

MTPADMINTRACE 環境変数, 15

N

ncd101.key ファイル, 165
ncd97.key ファイル, 165
ncdvt.key ファイル, 165

P

PATH 環境変数, 16
PCT。「プログラム管理テーブル (PCT)」を参照
PL/I
 共有ライブラリの環境変数, 12
 プログラム, 63
PPT。「処理プログラムテーブル (PPT)」を参照
PU4/5。「TPS PU4/5 サーバー」を参照

R

RDBMS
 KIXRDBMS_BYPASS 環境変数, 153

- RDBMS と機能する Sun MTP の構成, 149 ~ 154
- SIT の値の設定, 151 ~ 152
- 環境変数の設定, 149
- 起動時のバイパス, 153
- システム初期設定時のユーザー出口ルーチンの呼び出し, 71, 152
- 選択, 150
- 領域の起動, 153
- RDBMS のユーザー出口ルーチン, 71, 152
- Remote Animator, 13
- Reset キー, 4

S

- Secure Socket Layer。 「SSL」 を参照
- SIT。 「システム初期化テーブル (SIT)」 を参照
- SNA
 - ISC がサポートされる製品, 107
 - unikixdcl の構成, 108 ~ 113
 - 環境変数, 10
 - サーバーの構成, 123
 - 通信マネージャー (unikixCommMgr) の設定, 108 ~ 110
 - プロセス、数, 36
- Solaris のメモリーのコマンド, 44
- source コマンド, 21
- SPOOLCMD 環境変数, 104
- SPOOLER 環境変数, 103
- SSL
 - Sun MTP の構成, 135
 - unikixrc.cfg ファイルのエントリ, 137
 - サーバーの起動, 140
- Sun Mainframe Batch Manager。 「Sun MBM」 を参照
- Sun MBM、 Sun MBM と機能する Sun MTP の構成, 143 ~ 148
- Sun MTP Secure, 18
- Sun MTP クライアント (ECI/EPI), 133
- Sun MTP サーバー
 - 起動, 23, 24
 - 終了, 26 ~ 27
- Sun MTP ユーティリティ

- dfhusdup, 82
- kixkeydef, 86 ~ 89
- kixprint, 102
- kixstart, 24
- kixstop, 27
- sun.key ファイル, 165
- sun-cmd.key ファイル, 165
- Sybase の構成, 153
- SysID
 - SIT フィールド, 70
 - 「TCT-3270 Devices」 フィールド, 122, 130
 - 「TCT-System Entries」 フィールド, 111, 114

T

- Table Manager, 58
- TCP/IP
 - unikixtcp サーバー, 113
 - unikixtcp の構成, 113 ~ 116
 - 環境変数, 16
 - ソケット接続, 28
- TCP/IP ソケットのサポート, 134
- TCPRTERM 環境変数, 16, 113
- TCPSTERM 環境変数, 16, 113
- TCT。 「端末管理テーブル (TCT)」 を参照
- TCT の 「LU Name」 フィールド, 121, 130
- TDQ。 「一時データキュー」 を参照
- TERM 環境変数, 86, 90
- TMPDIR 環境変数, 17
- TN3270
 - エミュレータの接続, 29
 - サーバーの設定, 118
 - ポート番号, 120
- TNmsg.log, 120
- TPS PU4/5 サーバー
 - 3270 デバイスのサポート, 123
 - 構成, 124
 - 自動ログインの構成, 128

U

unikixadmin サーバー, 15
unikixb、3270 クライアント
 開始方法, 29
 説明, 124
unikixbld および IDCAMS, 146
unikixCommMgr
 ISC 接続, 108
 SSL 接続, 136
 TN3270 の接続, 118
 unikixrc.cfg ファイル, 108, 118, 136, 141
 WebSphere MQ 接続, 141
 説明, 108, 118, 136, 141
unikixdcl、構成, 108 ~ 113
unikixi、3270 クライアントイニシエータ
 「TPS PU4/5 サーバー」も参照
 起動, 132
 説明, 123
 プロセス, 29
unikixl, 28
unikixmain
 1 バイト文字のコードページ変換テーブル, 52
 2 バイト文字のコードページ変換テーブル, 163
 DBCS の有効化, 156
 Sun MTP の起動, 24
 共有メモリーおよびプロセスローカルメモリーの起動パラメータ, 42
 ポート番号
 TCP/IP クライアント, 116
 遠隔の 3270 クライアント, 131
 ソケットクライアント, 134
 リソースの自動インストール, 82
unikixmain コマンド, 23
unikixqm、構成, 141 ~ 142
unikixrc.cfg
 DCL サーバー, 110
 SSL サーバー, 137
 TN3270 サーバー, 118 ~ 120
 WebSphere MQ サーバー, 142
unikixsock
 起動, 134
 説明, 28
unikixssl、構成, 135 ~ 140

unikixtcp、構成, 113 ~ 116
unikixtnemux, 118 ~ 120
unikixtran, 11
unikixtrin、3270 トランザクションイニシエータサーバー
 起動, 131
 説明, 123
UNIXIX 環境変数, 17
UNIX
 カーネルのパラメータ, 48 ~ 51
 端末の構成, 85 ~ 93

V

VCT。「VSAM 構成テーブル (VCT)」を参照
VSAM 構成テーブル (VCT)
 Sun MBM の有効化, 144
 共有メモリー, 40 ~ 42
 トランザクション処理プログラム, 33
 バッチパラメータ, 34
 表示, 59, 75, 144, 151
 ユーザー数, 36
VSAM バッファープールおよびバッファ, 46
VSAM ファイル
 Sun MBM からのアクセス, 143
 再使用できるかどうかの定義, 146
 タイプ, 66
vt100.key ファイル, 165
vt200.key ファイル, 165
vt320.key ファイル, 165

W

WebSphere MQ, 141
wy50.key ファイル, 165

X

xev ユーティリティ, 93
xterm.key ファイル, 165
xterms.key ファイル, 165

X クライアントの定義, 92

X 端末

Sun MTP X クライアントの定義, 92

キーファイル, 86

キーボード割り当て, 90

キーマップ, 170, 171

X リソース定義, 92

あ

アカウントティング

トランザクション, 63

領域全体, 70

アクティブでない端末、タイムアウト期間, 72

宛先管理テーブル (DCT), 97

アテンション識別子 (AID) キー, 85

アプリケーションリソース

代替リソース定義による構成, 79 ~ 84

テーブルでの構成, 55 ~ 73

い

一時データキュー (TDQ)

プリンタの関連付け, 97 ~ 100

インストーラディレクトリ, 17

え

遠隔クライアント

ECI/EPI の自動インストーラ, 134

設定, 117 ~ 132

遠隔システム

端末識別子, 122, 130

プログラム名, 68

遠隔領域

自動インストーラ, 116

自動インストーラされる最大数の設定, 13

お

オペレーティングシステムのパラメータ。「カーネルのパラメータ」を参照

か

カーネルのパラメータ

共有メモリーのパラメータ, 49

構成, 49

セマフォ, 49

プロセス, 50

プロセス間通信, 49

ガイドライン、構成

Sun MTP プロセスの数, 35

VSAM バッファープールおよびバッファ, 46

カーネルのパラメータ, 48

回復, 75

共有メモリー, 40

コードページ変換テーブル, 52

照会ジョブの最大数, 34

スワップ空間, 48

トランザクション処理プログラムの数, 33

バックグラウンドタスクの最大数, 34

バッチジョブの最大数, 34

ローカルプロセスメモリー, 36

回復

構成, 75 ~ 77

構成に関するガイドライン, 35

設定

FCT での, 66

VCT での, 60, 75

ブロックサイズ, 60, 76

外部セキュリティー管理

「Sun MTP Secure」も参照。

環境変数, 18

リソースのアクセス権, 18

外部表示インタフェース (EPI), 28

外部呼び出しインタフェース (ECI), 28

会話型プロセス、最大数, 61

画面の形式、Sun MTP, 3

環境変数

ACUCOBOL, 9

COBCPY, 9

COBDIR, 9
COBMODE, 10
COBPATH, 10
CODE_PREFIX, 10
COPYPATH, 10
DCLRTERM, 10, 110
DCLSTERM, 10, 110
KIX62PASSTHRU, 11
KIX_ADJ_DST, 11
KIXAPPSEC, 18
KIXBIN, 11
KIXBTCH, 11
KIXCMDSEC, 18
KIXCONVTRACE, 11
KIXDCTSEC, 18
KIXDUMPDIR, 12
KIXDYNRTE, 12
KIX_ENABLE_JAVA, 11
KIXFCTSEC, 18
KIXGETMAINDMP, 12
KIXHOST, 12, 128
KIXJCTSEC, 18
KIX_JVM_OPTIONS, 12
KIXLIB, 12
KIXLICDIR, 12
KIXMAPS, 12
KIXMAXIST, 13, 116, 133, 134
KIXPCTSEC, 18
KIX_PGM_MODE, 13
KIX_PGMTXN_MODE, 13
KIXPPTSEC, 18
KIXPROGS, 13
KIXRDBMS_BYPASS, 13
KIX_READLOCKOFF, 13
KIX_READLOCKON, 13
KIXREMANIMPORT, 13
KIXREMANIMTOUT, 14
_KIX_RESERVED_MAPS, 14
KIXSEC, 18
KIXSECDFLTUSER, 18
KIXSEC_LOGGING, 18
KIXSECPREFIX, 18
KIXSEC_SERVERHOST, 19
KIXSEC_SERVERPORT, 19
KIXSNAPDIR, 14
KIXSNT_NOMEMUPDATE, 14
KIXSOCKET_TIMEOUT, 14, 135
KIXSTTSEC, 19
KIXTCTSEC, 19

KIXTSTSEC, 19
KIXVSAMPOOLS, 15, 46
LD_LIBRARY_PATH, 15
LIBPATH, 15
MTPADMINTRACE, 15
PATH, 16
RDBMS, 149
SPOOLER, 103
SPOOLER 環境変数, 103
Sun MTP Secure, 17 ~ 19
TCPRTERM, 16, 113
TCPSTERM, 16, 113
TERM, 86, 90
TMPDIR, 17
UNIKIX, 17
外部セキュリティー管理。「Sun MTP Secure」
を参照
設定, 19 ~ 21
設定スクリプトの例, 20
設定ファイルの実行, 21
複数のパスの割り当て, 20
領域の指定, 20

き

キーシーケンスデータセット (KSDS), 66
キーファイル
kixkeydef での作成, 86 ~ 89
X 端末, 86
説明, 85
ディレクトリ, 90
付属, 165
編集, 89 ~ 90
例, 88
キーボード
定義, 90
デフォルト, 86
割り当て, 165
キーボード割り当て, 165 ~ 171
3270 端末, 91
X 端末, 90
例, 166 ~ 171
キーマップ, 165
起動

RDBMS をバイパスした領域, 153

SSL サーバー, 140

Sun MTP サーバー, 23 ~ 24

領域と RDBMS, 153

ローカルクライアント, 24

起動シェルスクリプト, 93

起動端末識別子, 72

機能シップ, 107

基本マッピングサポート。「BMS」を参照

共通作業領域 (CWA), 72

共有オブジェクト。「共有ライブラリ」を参照

共有バッファ, 60

共有メモリー

AIX プラットフォーム, 42

カーネルのパラメータ, 49

起動パラメータ, 42

共有ライブラリの使用, 43

種類, 40

使用法, 40

ダンプ, 12

追加ポイント, 43

要件の計算, 41

共有ライブラリ

PPT での設定, 68

環境変数, 15

く

クライアント

3270 デバイス, 29

ECI/EPI, 28, 133

EPI アプリケーション, 28

J3270 端末エミュレータ, 29

SNA 3270, 123 ~ 132

TCP/IP ソケット, 28, 134

TN3270 または TN3270E, 29, 117 ~ 123

WebSphere MQ, 141

自動インストール, 134

プロセス数の決定, 36

ローカル端末, 28

け

検索間隔, バッチ, 61

こ

構成

ECI/EPI クライアント, 133

RDBMS と機能する Sun MTP, 149 ~ 154

SNA 3270 クライアント, 123 ~ 132

SSL サーバー, 135

Sun MBM と機能する Sun MTP, 143 ~ 148

TN3270 サーバー, 118

TPS PU4/5 サーバー, 124

UNIX 端末, 85 ~ 93

WebSphere MQ クライアント, 141

アプリケーションリソース

FCT, 64

PCT, 62

PPT, 67

VCT, 59

遠隔クライアント, 117 ~ 132

カーネル, 48 ~ 51

ガイドライン, 31 ~ 51

システム間通信 (ISC)

SNA, 110 ~ 113

TCP/IP, 113 ~ 116

代替リソース定義, 81

タスクリスト, 1

データベース, 150

プリンタ, 102

領域, 7 ~ 21

構成ガイドライン。「ガイドライン、構成」を参照

コードページ, 52

コードページ変換テーブル

1 バイト文字, 52

2 バイト文字, 163

コピーライブラリ

ACUCOBOL-GT, 10

Micro Focus Server Express, 9

コマンド

ASSIGN, 70

lp, 103

source, 21

リソース定義, 80

し

シェル, 21

シェル スクリプト

起動, 93

シェルスクリプト

kixprint, 102

システム間通信 (ISC)

DBCS のデータの変換, 157

SNA のサポート, 107

機能シップ, 107

構成

SNA (unikixdcl), 108 ~ 113

TCP/IP (unikixtcp), 113 ~ 116

トランザクション経路指定, 107

トランザクションセキュリティ, 116

非同期処理, 107

分散トランザクション処理 (DTP), 107

分散プログラムリンク (DPL), 107

保護セッション, 114

システム識別子, 70

システム初期化テーブル (SIT)

RDBMS の値の設定, 151

表示, 69

フィールド, 70 ~ 73

システムネットワークアーキテクチャー。

「SNA」を参照

事前定義の TCT, 118

自動インストール

ECI/EPI クライアント, 134

遠隔領域, 116

自動ログイン

PU2 構成, 126

PU4/5 構成, 128

TCT のエントリ, 129

ジャーナルファイル識別子, 63, 70

従来の Sun MTP と代替リソース定義との

相違点, 84

終了

Sun MTP サーバー, 26 ~ 27

ユーザーセッション, 25

照会ジョブ、最大数, 61

照会ジョブの最大数, 34

処理プログラムテーブル (PPT), 67 ~ 69

す

スワップ空間、決定, 48

せ

セキュリティ、トランザクション, 63

セキュリティマネージャー、外部。「Sun MTP Secure」を参照

設定ファイル, 20

セマフォのパラメータ, 49

そ

相対レコードデータセット (RRDS), 66

ソケット

SSL。「SSL」を参照

タイムアウト値の設定, 135

リスナープロセスの開始, 134

た

代替リソース定義

DFHUSD ファイル, 79

GCT テーブルのエントリ, 81

Sun MTP の実装, 79

サポートされるリソースタイプ, 79

サポートするコマンド, 80

長所と短所, 83

タイムアウト期間、アクティブでない端末, 72

端末

ASCII, 93

アクティブでない、タイムアウト期間, 72

構成, 85

識別子

3270 Devices のホストとポートの画面, 131

- TCT の, 121, 129
- 遠隔システム, 122, 130
- 端末所有領域, 72
- 端末管理テーブル (TCT)
 - 3270 デバイスのエントリ, 129 ~ 132
 - SNA のシステムエントリ, 111 ~ 112
 - TCP/IP のシステムエントリ, 114 ~ 115
 - プリンタのエントリ, 99
- 端末識別子、起動, 72
- 端末所有領域 (TOR), 72
- 端末の構成, 85 ~ 86

ち

調整パラメータ。「カーネルのパラメータ」を参照

つ

通信

- ISC の構成, 107 ~ 116
 - 「unikixCommMgr」も参照
- 遠隔クライアントの構成, 117 ~ 142
 - 「システム間通信 (ISC)」も参照
- プロセス
 - unikixb, 124
 - unikixCommMgr, 108, 118
 - unikixdcl, 107, 108
 - unikixi, 123
 - unikixqm, 142
 - unikixsock, 134
 - unikixtcp, 107, 113
 - unikixtnemux, 118, 119
 - unikixtrin, 123

通信マネージャー。「unikixCommMgr」を参照

て

停止

- CEMT SHUT, YES トランザクション, 26
- Sun MTP サーバー, 26 ~ 27
- ユーザーセッション, 25

ディレクトリ

- \$ACUCOBOL/bin, 16
- \$COBDIR/bin, 16
- \$KIXSYS/_kix_reserved_maps, 14
- \$UNIKIX/bin
 - \$PATH への追加, 16
 - リソース定義, 93
- \$UNIKIX/lib, 90
- \$UNIKIX/local/bin
 - \$PATH への追加, 16
 - kixprint, 102
- \$UNIKIX/local/lib, 90
- アプリケーションの構造の例, 8
- マップセットのステージング, 14

データセット、定義, 65 ~ 67

データ入力の検査, 4

データ変換。「DBCS」を参照

データ変換テンプレートテーブル (CVT)。「DBCS」を参照

テーブル

- VSAM 構成テーブル (VCT), 59
- システム初期化テーブル (SIT), 69
- 処理プログラムテーブル (PPT), 67
- ファイル管理テーブル (FCT), 64
- プログラム管理テーブル (PCT), 62

デバッグセッション、アクティブ, 61

デフォルトユーザー, 18

と

動的共有メモリー, 41

トランザクション

- CSPG, 73
- PCT での定義, 62

実行, 25

セキュリティレベル, 63

バックエンドの DTP, 64

トランザクションアカウンティング, 63

トランザクションイニシエータサーバー。「unikixtrin」を参照

トランザクション作業領域 (TWA), 63

トランザクション処理プログラム

- VCT での数の指定, 61

概念, 32

構成に関するガイドライン, 33 ~ 34
構造, 32
バッチジョブの数の定義, 34
トランザクションセキュリティ
ISC の, 116
Sun MTP Secure の環境変数, 17
トランザクションの実行, 25
トレース
unikixadmin サーバー, 15
内部ダンプファイル, 12
変換, 11

に

入力順データセット (ESDS), 66
認証データベース, 136

の

ノード、Sun MBM, 145

は

パスワード
SIT フィールド, 71
動的な変更機能の無効化, 14
バックエンドの DTP トランザクション, 64
バックグラウンドタスク、最大数, 61
バックグラウンドタスクの最大数, 61
バッチ検索の間隔, 61
バッチシェルスクリプト, 11
バッチジョブ、最大数の設定, 34, 61
バッチジョブの最大数, 61
バッファ、VSAM, 46
バッファプール、VSAM, 46

ひ

日付、デフォルトの形式, 72
非同期処理, 64

ふ

ファイル
/etc/inittab, 93
DFHUSD, 79
TNdebug.log, 120
TNmsg.log, 120
unikixrc.cfg, 108, 110, 118, 136, 141, 142
回復, 76
キー, 85, 165
ファイル管理テーブル (FCT), 64, 77
ファイル識別子, 5
ファイルタイプ、VSAM, 66
ファイルの回復方法, 75
ファンクションキー, 4
プリンタ
CPRT, 95
kixprint の変更, 103 ~ 105
SPOOLCMD 環境変数, 104
SPOOLER 環境変数, 103, 104
行と列, 102
構成
kixprint の使用, 102 ~ 105
LU タイプ 1, 100
TDQ を使用するプリンタ, 97 ~ 100
追加プリンタ, 97
デフォルトのプリンタ, 96 ~ 97
サイズ, 102
デフォルト, 95
デフォルトの印刷コマンド, 103
デフォルトのサイズ, 102
プログラム管理テーブル (PCT), 62 ~ 64
プログラム、リソースとして定義
PCT, 62
PPT, 68
遠隔システムの, 68
プロセス
カーネルのパラメータ, 50
数の決定, 35
「通信」も参照
プロセス間通信 (IPC) のパラメータ, 49
プロセスのパラメータ, 50
プロセスローカルメモリー、起動パラメータ, 42
分散トランザクション処理 (DTP), 64, 107

分散プログラムリンク (DPL), 107

へ

別名, 65

変数、環境。「環境変数」を参照

ほ

ポート番号

3270 クライアント, 129, 131

TCP/IP サーバー, 116

ソケットクライアント, 134

保護セッション、構成, 114

ま

マップ、BMS 位置合わせ, 73

マップセット

PPT での定義, 68

マップセット

ステージングディレクトリ, 14

め

メモリー

共有, 40

プロセスのマップ, 44

ローカルプロセス (ヒープ), 36

ゆ

ユーザーセッション、停止, 25

ユーザーの最大数, 60

ら

ライセンスファイル, 12

り

リソース定義ファイル, 79

リソースのアクセス権, 18

領域

アカウントの有効化, 70

遠隔領域の自動インストール, 116

環境変数, 9

構成, 7~21

設定ファイル, 20

定義, 7

領域の起動時に RDBMS をバイパス, 153

領域のリソースの自動インストール, 82

ろ

ローカルキュー, 64

ローカルクライアント

起動, 24

説明, 28

ローカルプロセスメモリー, 36

ログイン画面、3270 デバイスクライアント, 123