



# Sun™ Mainframe Batch Manager ソフトウェア 構成ガイド

---

Release 10.1.0

Sun Microsystems, Inc.  
[www.sun.com](http://www.sun.com)

Part No. 819-2505-10  
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ー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ー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 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。サンのロゴマークおよび 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。

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ず、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 Batch Manager Software Configuration Guide Part No: 817-7440-10 Revision A
-----	---



# 目次

---

はじめに xxiii

1. Sun MBM について 1
  - ノード 1
  - サブシステム 2
  - アプリケーションのディレクトリ構造の作成 3
2. Batch Administration Manager 5
  - BAM のメニュー構造 5
  - BAM の起動 9
    - ▼ BAMを起動する 9
3. ノードの設定 11
  - ノードの起動と停止 12
    - ▼ ノードを起動する 12
    - ▼ ノードを停止する 13
  - システム状態および有効なサブシステムの一覧表示 14
    - ▼ システム状態メニューを表示する 14
    - ▼ 有効なサブシステムを一覧表示する 15
    - ▼ 有効なジョブを一覧表示する 16
    - ▼ 完了したジョブを一覧表示する 17

## ジョブのログ管理 18

- ▼ 完了したジョブの一覧表示画面からジョブのログファイルを保存する 18
- ▼ コマンド行からジョブのログファイルを保存する 18
- ▼ ジョブのログファイルを消去する 19
- 保存されているジョブログファイルの表示 20
- ▼ 保存されているジョブログファイルを表示する 20

## コンソールの割り当て 20

- ▼ コンソールを割り当てる 21
- ▼ 現在のコンソール設定を表示する 22
- コンソール端末の設定 22
- ▼ コンソール端末を ON に設定する 23
- ▼ コンソール端末を OFF に設定する 23
- コンソールファイルの ON の設定 23
- ▼ コンソールファイルを ON に設定する 24
- ▼ コンソールファイルを OFF に設定する 24

## 循環履歴ファイルの設定 24

- ▼ 循環履歴ファイルを ON に設定する 25
- ▼ 循環履歴ファイルを OFF に設定する 26

## コンソール識別子の変更 26

- ▼ コンソール識別子を変更する 26

## ジョブの履歴ファイルのリダイレクト 27

- ▼ デフォルトのジョブ履歴ディレクトリを変更する 27

## 初期ジョブ番号の割り当て 28

- ▼ 初期ジョブ番号を割り当てる 28

- 4. 日付および時刻の変更 31
  - Sun MBM の現在の日付の変更 32
    - 領域の同期化 32
      - ▼ 領域を同期化する 33
      - ▼ Sun MTP/Sun MBM の日付を設定する 35
      - ▼ Sun MBM の日付形式を変更する 38
  - Sun MBM の時刻形式の変更 39
    - ▼ 時刻形式を変更する 39
  - 言語環境の管理 40
    - ▼ 言語環境を変更する 40
- 5. ノード間の通信 43
  - 使用可能なノードの一覧表示 44
    - ▼ 使用可能なノードを一覧表示する 44
  - 複数のノードの設定 46
    - ノードのサーバー (ローカル) 指定 46
      - ▼ ノードをサーバーとして指定する 46
    - サーバー (ローカル) 指定の削除 48
      - ▼ ノードのサーバー指定を削除する 48
  - 遠隔ノードの定義 48
    - ▼ 遠隔ノードを定義する 48
  - 遠隔ノードの変更 50
    - ▼ 遠隔ノードのポート番号を変更する 50
  - 遠隔ノードの削除 51
    - ▼ 遠隔ノードを削除する 51

- 6. ジョブアカウンティングの設定 53
  - Job Accounting メニュー 54
    - ▼ Job Accounting メニューを表示する 55
  - 現在のジョブアカウンティング設定の表示 55
    - ▼ ジョブアカウンティング設定を表示する 55
  - ジョブアカウンティングの有効化および無効化 56
    - ▼ ジョブアカウンティングを有効化する 56
    - ▼ ジョブアカウンティングを無効化する 57
  - ジョブアカウンティング設定のデフォルト設定 57
    - ▼ デフォルト設定を設定する 57
  - 別のジョブアカウンティングファイルへの変更 58
    - ▼ ジョブアカウンティングファイルを変更する 58
  - ジョブアカウンティングのレコード形式の変更 58
    - ▼ ジョブアカウンティングのレコード形式を変更する 59
- 7. サブシステムの作成 63
  - サブシステムの作成準備 64
  - 使用可能なサブシステムの一覧表示 66
    - ▼ ノードで使用可能なすべてのサブシステムを一覧表示する 66
  - サブシステムの作成 67
    - ▼ サブシステムを作成する 67
    - ▼ アプリケーション言語を選択する 69
    - ▼ 1つまたは複数のデータベースを選択する 70
    - ▼ データファイルのタイプを選択する 71
    - ▼ Sun 以外のパッケージを選択する 72
    - ▼ 日付/時刻管理を選択する 73
    - ▼ COBOL 実行時システムにユーザー固有オブジェクトを取り込む 74

- メインフレーム互換性オプションの指定 75
  - ▼ Mainframe Compatibility メニューを表示する 75
  - ▼ MVS REPRO オプションを設定する 76
  - ▼ MVS OUTPUT オプションを設定する 78
  - ▼ MVS MAXCC 値を設定する 79
  - ▼ MVS RLS オプションを設定する 80
  - ▼ VSE JCL REPRO オプションを設定する 81

#### 設定内容の表示 82

- ▼ 設定内容を表示する 82

#### サブシステムの作成 83

- ▼ サブシステムを作成する 83

### 8. サブシステムのカスタマイズ 87

#### サブシステムの現在の構成の表示 89

- ▼ サブシステムの構成を表示する 89

#### サブシステムの更新 92

- ▼ サブシステムの Update メニューを表示する 92
- ▼ ジョブ履歴ディレクトリを変更する 93

#### Sun MTP 領域の変更 94

- ▼ 領域を変更する 94

#### サブシステムの File\_Map の更新 96

- ▼ File\_Map を更新する 96

#### サブシステムのユーザー設定ファイルのカスタマイズ 97

- ▼ 定義済み設定ファイルのエントリを \$USER\_SETUP に挿入する 97
- ▼ ユーザー設定ファイルを編集する 98

#### デフォルトのサブシステムの定義 98

- ▼ デフォルトのサブシステムを定義する 99

#### サブシステムの削除 100

- ▼ サブシステムを削除する 101

- 削除したサブシステムの再作成 102
  - ▼ 削除したサブシステムを再作成する 102
- サブシステムのインポート 104
  - ▼ サブシステムをインポートする 104
- 9. セキュリティーおよびユーザーの設定 107
  - サブシステムへのアクセス制御 108
    - ▼ サブシステムへアクセスできるユーザーまたはグループを指定する 108
  - Sun MBM コマンドへのアクセス制御 110
    - ▼ Command Controls メニューを表示する 110
    - ▼ ノードを起動できるユーザーを追加または変更する 111
    - ▼ ノードを停止できるユーザーを追加または変更する 113
    - ▼ ノードを管理できるユーザーを追加または変更する 114
    - ▼ サブシステムを管理できるユーザーを追加または変更する 115
    - ▼ クラスおよびアクティビティーを作成できるユーザーを追加または変更する 117
    - ▼ クラスおよびアクティビティーを削除できるユーザーを追加または変更する 118
  - ユーザーの作業用ディレクトリの管理 119
    - ▼ User's Work Directory メニューを表示する 120
    - ▼ すべてのユーザーの作業用ディレクトリを一覧表示する 121
    - ▼ ユーザーの作業用ディレクトリを作成する 122
    - ▼ ユーザーの作業用ディレクトリを変更する 123
    - ▼ ユーザーの作業用ディレクトリを削除する 124
  - Sun MBM コマンドを実行するためのユーザー環境の設定 125
    - ▼ メインメニューから環境を設定する 125
    - ▼ コマンド行から環境を設定する 126

- 10. ジョブクラスおよびアクティビティの設定 127
  - ジョブクラスの一覧表示 128
    - ▼ すべてのジョブクラスを一覧表示する 128
  - ジョブクラスおよびアクティビティの作成 130
    - ▼ クラスおよびアクティビティを作成する 131
  - アクティビティ数の変更 132
    - ▼ アクティビティ数を変更する 132
  - ジョブクラスの削除 133
    - ▼ ジョブクラスを削除する 133
  
- 11. Problem Determination 135
  - エラーログファイル 136
    - ▼ エラーログファイルを表示する 136
    - ▼ ログファイルのサイズを変更する 137
  - サブシステムの追跡 140
    - ▼ サブシステムを追跡する 140
  - プロセスの追跡 142
    - ▼ プロセスを追跡する 142
  - メッセージの追跡 143
    - ▼ メッセージログを有効化する 143
    - ▼ メッセージ追跡を表示する 144
    - ▼ メッセージログを無効化する 145
  - 共有メモリーの表示 145
    - ▼ Sun MBM の共有メモリーを表示する 145
  - ディスク容量の確認 146
    - ▼ BAM メニューによってディスク容量を確認する 147
    - ▼ bam コマンドによってディスク容量を確認する 147

障害追跡	148
▼ システムのスナップショットを作成する	148
▼ テストを実行する	149
BAM ログファイル	150
A. ジョブの実行環境の設定	151
設定ファイルの処理順序	151
\$PACK/btshrc ファイル	152
\$HOME/.btshrc ファイル	152
プリプロセスファイル	153
ポストプロセスファイル	153
環境変数	154
\$FMROOT のリセット	154
リセットできない環境変数	154
オンライントランザクションによるバッチジョブのサブミット	155
遠隔ジョブのサブミット	162
遠隔ジョブサブミット機能を有効化する	162
ftp コマンドによる遠隔ジョブのサブミット	164
mvstrans または dostrans のオプションの指定	165
遠隔ログの有効化および有効化解除	165
▼ ログを有効化する	165
▼ ログを有効化解除する	165
B. .install ファイル	167
.install ファイルの内容	167
.install ファイルの例	173

C.	batchenv ファイルのカスタマイズ	177
	batchenv ファイルの内容	177
	Sun MBM コマンド出力の制御	179
	EBM_DATE_FORMAT	179
	EBM_TIME_FORMAT	180
	LC_TIME	180
D.	サブシステム設定ファイル (\$SETUP)	183
	標準環境変数	184
	ACUCOBOL	184
	A_TERMCAP	184
	COBDIR	184
	EBMTMPDIR	184
	FILEMAP	184
	HANDLE_MFVS	184
	KIXSYS	184
	LD_LIBRARY_PATH	185
	LIBPATH	185
	LPI_PRODUCT_DIR	185
	RUNBDIR	185
	RUNBPATH	185
	SORT_MODE	185
	UNIKIX	186
	RDBMS 環境変数	186
	RDBMS_CONNECT	186
	RTS_RDBMS	186
	その他の環境変数	186
	FORMS	186
	SYSOUTDIR	186

E.	RDBMS 用アプリケーションのカスタマイズ	187
	RDBMS へのアクセス	188
	fsusrexit.c - Oracle 関数	189
	fsusrexit.c - DB2 UDB 関数	191
	fsusrexit.c - Sybase 関数	193
F.	Sun MTP との通信	197
	Sun MTP 環境変数	197
	Sun MTP VSAM データセットへのアクセス	198
	ノードに接続する領域の設定	199
G.	WebSphere MQ との統合	201
	前提条件	201
	MQ サブシステムの作成	202
	▼ MQ VSAM サブシステムを作成する	202
	▼ 領域と MQ サブシステムを関連付ける	204
	用語集	205
	索引	211

# 目次

---

図 1-1	Sun MBM 階層	2
図 1-2	給与計算アプリケーションのディレクトリ構造の例	4
図 2-1	BAM 最上位メニュー	5
図 2-2	ソフトウェアライセンス管理のメニュー構造	6
図 2-3	システム環境メニュー構造	6
図 2-4	アプリケーションとサブシステムのメニュー構造	7
図 2-5	セキュリティーとユーザーの設定メニュー構造	7
図 2-6	ジョブクラスとアクティビティーの設定メニュー構造	8
図 2-7	問題特定のメニュー構造	8
図 2-8	BAM メインメニュー	10
図 3-1	システム環境メニュー構造	11
図 3-2	BAM System Environments メニュー	12
図 3-3	BAM Report System Status メニュー	14
図 3-4	List Active Subsystems 画面	15
図 3-5	List Active Jobs 画面	16
図 3-6	Query Completed Jobs 画面	17
図 3-7	Erase Job Log メニュー	19
図 3-8	Assign the Console メニュー	21
図 3-9	Assign the Console Configuration 画面	22
図 3-10	循環履歴ファイルを ON に設定する画面	25

図 3-11	Redirect Job Output 画面	27
図 3-12	Initial Job Number 画面	28
図 4-1	日付/時刻の変更のメニュー構造	31
図 4-2	Change the Date 画面	33
図 4-3	Sun MTP 領域同期化の画面	34
図 4-4	「Sun MTP 日付/時刻の構成ユーティリティー」メニュー	35
図 4-5	Sun MTP/Sun MBM Current Date Configuration 画面	36
図 4-6	Sun MTP Startup Date Configuration	36
図 4-7	Sun MBM Startup Date Configuration 画面	37
図 4-8	Sun MBM 日付形式の画面	38
図 4-9	Sun MBM 時刻形式の変更画面	40
図 4-10	Sun MBM ロケール時間カテゴリの画面	41
図 5-1	ノード間通信のメニュー構造	43
図 5-2	Inter-Node Communications メニュー	45
図 5-3	すべてのノードの一覧表示の画面	45
図 5-4	ローカルノードの定義画面	47
図 5-5	遠隔ノードの定義画面	49
図 5-6	遠隔ノードの変更	50
図 5-7	遠隔ノードの削除画面	51
図 6-1	ジョブアカウンティングのメニュー構造	54
図 6-2	Job Accounting メニュー	55
図 6-3	Current Job Accounting Information 画面	56
図 6-4	ジョブアカウンティングのレコード形式ファイル	59
図 7-1	サブシステム作成のメニュー構造	63
図 7-2	Applications & Subsystems メニュー	66
図 7-3	サブシステムの一覧表示画面	67
図 7-4	Create メニュー	68
図 7-5	Application Languages 選択画面	69
図 7-6	DBMS 選択画面	70
図 7-7	Data Management 画面	71

図 7-8	Other Packages 画面	72
図 7-9	COBOL Date/Time 画面	73
図 7-10	User-Specific Objects ファイル	74
図 7-11	Mainframe Compatibility メニュー	75
図 7-12	MVS Environment メニュー	76
図 7-13	MVS REPRO 画面	77
図 7-14	MVS OUTPUT 画面	78
図 7-15	MVS MAXCC 画面	79
図 7-16	MVS RLS 画面	80
図 7-17	VSE REPRO 画面	81
図 7-18	サブシステムの現在の設定内容の表示	82
図 7-19	Sun MTP インストールディレクトリの指定	83
図 8-1	サブシステムの照会、更新、削除、デフォルトの変更/表示、インポートのメニュー構造	88
図 8-2	Query メニュー	89
図 8-3	サブシステムの特性画面	90
図 8-4	サブシステムの構成画面	90
図 8-5	サブシステムの設定画面	91
図 8-6	Query File_Map 画面	92
図 8-7	サブシステムの Update メニュー	93
図 8-8	MTP 領域の変更画面	95
図 8-9	MTP 領域の変更確認画面	95
図 8-10	File_Map ファイルの更新	96
図 8-11	定義済み設定ファイル	97
図 8-12	Default Subsystem 画面	99
図 8-13	デフォルトのサブシステムの選択	100
図 8-14	Delete Subsystem 画面	101
図 8-15	サブシステムの再作成画面	102
図 8-16	サブシステムの再作成の確認画面	103
図 8-17	Import Subsystem - 既存のサブシステムリスト	104

図 8-18	ソースノードサブシステム	105
図 9-1	セキュリティーとユーザーの設定メニュー構造	108
図 9-2	Security & Users メニュー	109
図 9-3	Application Controls 画面	109
図 9-4	Command Controls メニュー	111
図 9-5	ノード起動の許可ファイル	112
図 9-6	ノード停止許可ファイル	113
図 9-7	ノード管理の許可ファイル	114
図 9-8	サブシステム管理の許可ファイル	116
図 9-9	クラスとアクティビティーの作成の許可ファイル	117
図 9-10	クラスとアクティビティーの削除の許可ファイル	118
図 9-11	User's Work Directory メニュー	120
図 9-12	User's Work Directory 一覧	121
図 9-13	User's Work Directory 作成	122
図 9-14	User's Work Directory 変更	123
図 9-15	User's Work Directory 削除	124
図 9-16	Command Prompt 画面	125
図 10-1	ジョブクラスとアクティビティーの設定メニュー構造	128
図 10-2	Classes & Activities メニュー	129
図 10-3	List ALL Job Classes 画面	130
図 10-4	ジョブクラスの作成	131
図 10-5	アクティビティー最大数の設定画面	132
図 10-6	ジョブクラスの削除画面	134
図 11-1	問題特定のメニュー構造	135
図 11-2	lgdem デーモン	136
図 11-3	エラーログ (elgfile) 画面	137
図 11-4	ADMLOG Functionality メニュー	138
図 11-5	ADMLOG: Configure Log Environment メニュー	138
図 11-6	ADMLOG: 循環ファイル設定画面	139
図 11-7	サブシステムの追跡画面	141

図 11-8	Trace Processes 画面	142
図 11-9	Trace Messages メニュー	143
図 11-10	メッセージ追跡の画面	144
図 11-11	Dump Memory 画面	146
図 11-12	BAM ログファイルの例	150
図 F-1	Sun MBM による VSAM データセットへのアクセス	198
図 G-1	Other Packages 画面	203



# 表目次

---

表 1-1	ディレクトリ構造	3
表 4-1	日付/時刻の形式設定指示	38
表 6-1	ジョブアカウンティングの形式設定指示	60
表 7-1	サブシステムのタイプで必要な情報	64
表 A-1	Sun MBM ジョブ設定ファイル	151
表 B-1	<code>.install</code> ファイルの内容	167



# コード例

---

コード例 3-1	ノードの起動メッセージ	13
コード例 3-2	ノードの停止メッセージ	13
コード例 A-1	kixjob シェルスクリプト	157
コード例 A-2	unikixjob コマンドによるジョブのサブミット	161
コード例 B-1	.install ファイル	173
コード例 C-1	Solaris システムの batchenv ファイル — 例	177
コード例 E-1	Oracle 用の fsusrexite.c	189
コード例 E-2	DB2 UDB 用の fsusrexite.c	191
コード例 E-3	Sybase 用の fsusrexite.c	193



# はじめに

---

このマニュアルでは、Sun™ Mainframe Batch Manager (Sun MBM) システムの構成方法について説明します。このマニュアルは、Sun MBM 以外に Sun Mainframe Transaction Processing ソフトウェア (Sun MTP) や COBOL などの必要なソフトウェアがインストールされていることを前提としています。使用するアプリケーション環境に応じたシステムのカスタマイズ方法の詳細は、『Sun Mainframe Batch Manager ソフトウェア 移行ガイド』および『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。

このマニュアルは、Sun MBM 管理者、システムプログラマ、および Sun MBM を実行する運用環境に精通したユーザーを対象としています。

---

## マニュアルの構成

第 1 章では、Sun MBM ソフトウェア、ノード、およびサブシステムについて説明します。アプリケーションのディレクトリ構造の例も示します。

第 2 章では、Batch Administration Manager (BAM) の起動方法について説明します。

第 3 章では、バッチシステム環境の管理方法について説明します。

第 4 章では、日付の設定および同期化の方法について説明します。

第 5 章では、ノード間通信の構成および管理の方法について説明します。

第 6 章では、ジョブアカウンティングの有効化/無効化および管理の方法について説明します。

第 7 章では、サブシステムの作成方法について説明します。

第 8 章では、サブシステムの更新、デフォルトサブシステムの定義、およびサブシステムの削除と再作成の方法について説明します。

第 9 章では、ノードとサブシステムでのセキュリティーの確保の方法、およびユーザーの作業用ディレクトリの管理方法について説明します。

第 10 章では、ジョブクラスおよびアクティビティーの構成方法について説明します。

第 11 章では、問題診断のための BAM の使用法について説明します。

付録 A では、設定ファイルの実行のされ方、オンライントランザクションからのジョブのサブミットの方法、および遠隔ジョブのサブミットの方法について説明します。

付録 B では、`.install` ファイルについて説明します。

付録 C では、`batchenv` ファイルのカスタマイズの方法について説明します。

付録 D では、サブシステムの設定ファイルについて説明します。

付録 E では、リレーショナルデータベース管理システム (RDBMS) を使用するためのアプリケーションのカスタマイズの方法について説明します。

付録 F では、Sun MTP と通信するための環境の設定方法について説明します。

付録 G では、Sun MBM と WebSphere MQ を統合する方法について説明します。

構成パラメータのいくつかは設定後に Sun MBM ノードのシャットダウンが必要ですが、大部分のパラメータはノードを実行したままで動的に設定できます。

---

## 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<sup>®</sup> 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<sup>™</sup> 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.
<b>AaBbCc123</b>	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。	% <b>su</b> Password:
<i>AaBbCc123</i>	コマンド行の変数部分。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。	% <b>grep</b> '^#define \ XV_VERSION_STRING'
[ ]	省略可能な引数を示します。	dltjob <i>jon</i> [-n <i>name</i> ]
	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。	abtjob <i>jon</i> [-s <i>job</i>   <i>cmd</i> ]

\* 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。

---

## ファイル識別子

ファイル識別子は、次の 2 つの部分から構成されます。

- 1 つ以上のディレクトリを指定するディレクトリ、または環境変数
- ファイル識別子の最後の構成要素であるファイル名

---

ファイル識別子	説明
ディレクトリ	Sun MBM で使用される絶対ディレクトリ名は、60 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、同じディレクトリを指定します。 <ul style="list-style-type: none"><li>• /local/mbm/pack/bin</li><li>• \$PACK/bin</li></ul> 2 行目の PACK 環境変数は /local/mbm/pack に設定されています。\$ 指示子を使用することにより、PACK 環境変数がその完全な意味に展開されます。
環境変数	ディレクトリ名やファイル名を含む名前であり、通常 1 ~ 14 個の大文字です。
ファイル名	ファイル名は、ご使用のプラットフォームの制限事項に従う必要があります。

---

# 関連マニュアル

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 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 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 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 構成ガイド』	819-2515-10
	『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 ご使用に あたって (Solaris プラットフォーム用)』	819-2521-10
『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	
IBM MVS	『IBM MVS/ESA JCL Reference』	GC28-1479
IBM VSE	『IBM VSE/ESA System Control Statements』	SC33-6713
	『IBM VSE/POWER Administration and Operation』	SC33-6733
Server Express を 使用してくださ い。	Server Express のライセンス設定	*
ACUCOBOL-GT	ACUCOBOL-GT のマニュアル	*

---

製品	タイトル	Part No.
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	*

---

\* マニュアルの注文については、ベンダー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 Batch Manager ソフトウェア 構成ガイド』、Part No. 819-2505-10

# 第1章

---

## Sun MBM について

---

この章では、Sun MBM について説明します。この章の内容は、次のとおりです。

- 1 ページの「ノード」
- 2 ページの「サブシステム」
- 3 ページの「アプリケーションのディレクトリ構造の作成」

Sun MBM は、ジョブの開始時刻や優先順位などの構成パラメータに従って、バッチプログラムの実行を管理およびスケジューリングするためのバッチマネージャー製品です。実行するアプリケーションプログラムとそれらに必要なリソースは、複数のマクロを使って定義します。一連のマクロ文は 1 つのファイルにまとめ、単一のジョブとしてサブシステムにサブミットします。

ユーザーは、バッチ環境を制御するサブシステムにジョブをサブミットします。バッチジョブによって VSAM データセットがアクセスされる場合、Sun MBM が、VSAM ファイルを処理する Sun MTP 領域と接続するように設定する必要があります。バッチジョブとオンライントランザクションは、同じ VSAM データセットにアクセスできます。詳細は、71 ページの「データファイルのタイプを選択する」を参照してください。どのジョブを実行する場合も、Sun MBM ノードを設定し、少なくとも 1 つのサブシステムを定義する必要があります。

---

## ノード

ノードとは、特定のディレクトリにインストールされる Sun MBM パッケージです。同じプラットフォーム上に、複数の独立したノードを定義できます。たとえば、テスト用に 1 つのノード、本番稼動用に 1 つのノードを指定できます。または、給与計算アプリケーション用に 1 つ、在庫管理用に 1 つのノードを定義することもできます。

各ノードは、それぞれがインストールされたディレクトリによって区別されます。図 1-1 に、Sun MBM 階層を示します。

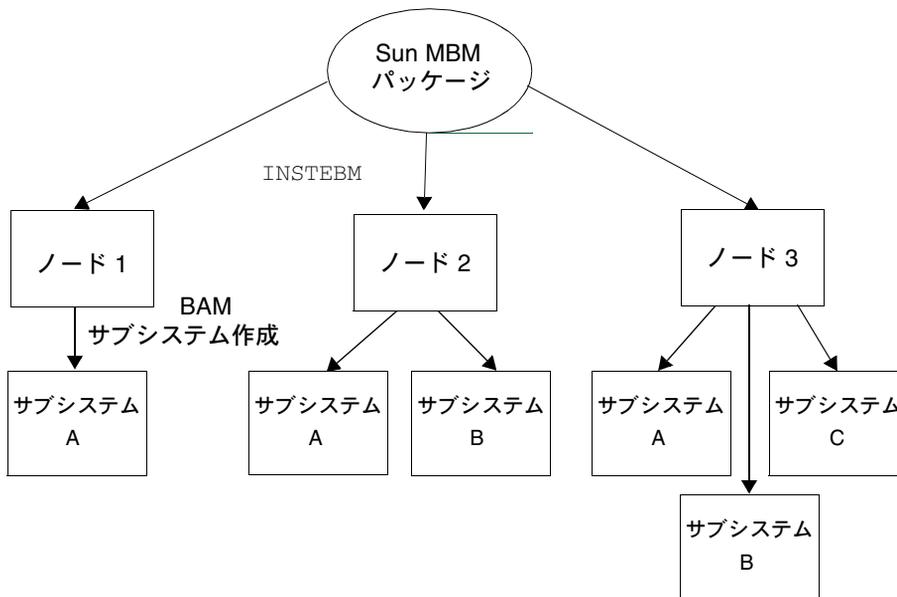


図 1-1 Sun MBM 階層

---

## サブシステム

「サブシステム」は、ジョブを実行するための環境を定義します。このサブシステムにより、ジョブと VSAM データ管理用の Sun MTP 領域または RDBMS とのインタフェースが実現します。各ノードには、それぞれ 8 つのサブシステムを作成できます。

各サブシステムには 1 つ以上の構成ファイルがあります。プライマリ構成ファイルは、サブシステム設定ファイルとも呼ばれ、Batch Administration Manager (BAM) を使用してサブシステムを作成するときに生成されます。このファイルには、サブシステムに関連する環境変数が含まれます。詳細は、付録 D を参照してください。

ジョブをサブミットするときには、そのジョブが実行されるサブシステムを指定する必要があります。ジョブの実行環境で、そのサブシステムの設定ファイルが使用されます。

# アプリケーションのディレクトリ構造の作成

サブシステムを作成する前に、アプリケーションのリソースを配置するディレクトリを特定する必要があります。図 1-2 に示すような独立型の構造を作成することもできます。

次の表は、図中に示されているディレクトリの説明です。環境変数は参考として示しています。

表 1-1 ディレクトリ構造

ディレクトリ	内容
bin	コンパイルスクリプト、コンパイラ命令ファイル、設定スクリプト、ユーザー実行可能ファイル
copy	COBOL コピーブック; Server Express を使用する場合は \$COBCPY、ACUCOBOL-GT を使用する場合は \$COPYPATH
data	VSAM データファイル
dclgen	RDBMS インクルードメンバー
history	バッチジョブログ
include	SQL および PL/I インクルードメンバー
jcl	JCL およびマクロスクリプトの最上位ディレクトリ
• ish	変換されたジョブ; \$JCLLIB
• ishp	変換されたプロシージャ; \$PROCLIB
• jmvms	ネイティブ MVS ジョブ
• mvsp	ネイティブ MVS プロシージャ
• include	ネイティブ MVS インクルードメンバー; \$MVSINCLDIR
• jdoss	ネイティブ VSE ジョブ
• dospp	ネイティブ VSE プロシージャ
• sli	ネイティブ VSE ソースライブラリインクルード (SLI) メンバー; \$SLIDIR
parmlib	プログラム、ソート、およびユーティリティーの制御文; \$PARMLIB
seqfiles	順編成ファイル; \$SEQFILES

表 1-1 ディレクトリ構造

ディレクトリ	内容
src/batch	バッチプログラム; \$JOB LIB
sys/licenses	ライセンスファイル; \$KIXLICDIR
sysoutdir	SYSOUT ファイル; \$SYSOUTDIR

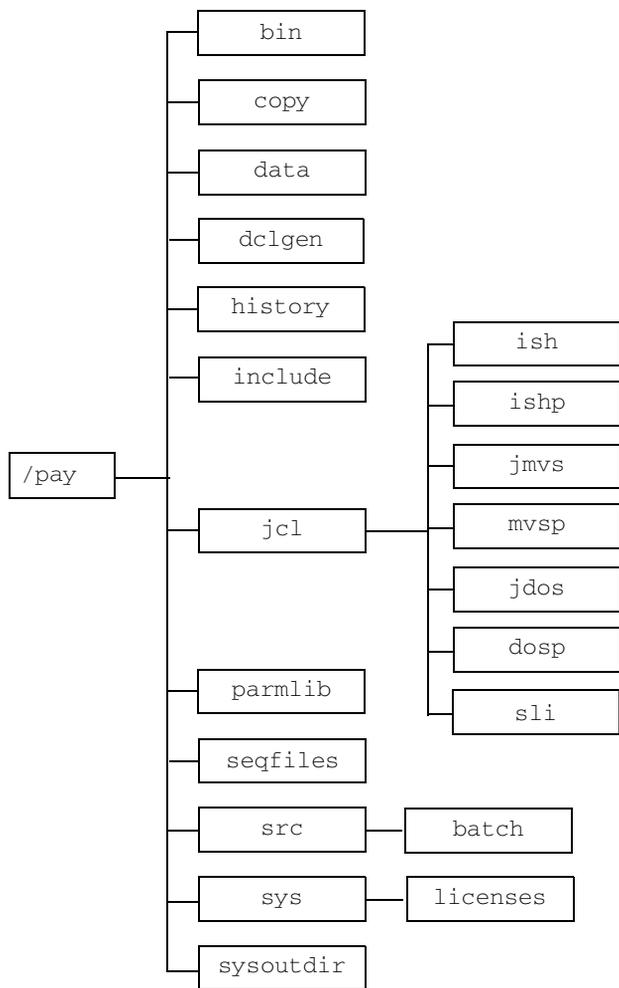


図 1-2 給与計算アプリケーションのディレクトリ構造の例

## 第2章

# Batch Administration Manager

Batch Administration Manager (BAM) は、構成作業を簡略化するための対話型インタフェースです。BAM の使用法についての情報は、「Help」メニューから参照できます。この章では、BAM のメニュー構造の概要を示し、BAM の起動方法について説明します。

## BAM のメニュー構造

次の図は、BAM 最上位メニュー項目です。

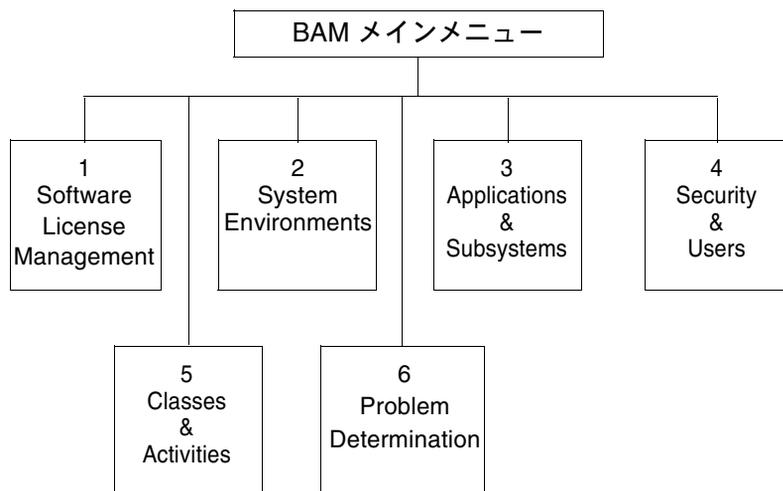


図 2-1 BAM 最上位メニュー

ノードのインストール後に Sun MBM を起動するには、有効なライセンスが必要です。次に、ソフトウェアライセンス管理のメニュー構造を示します。

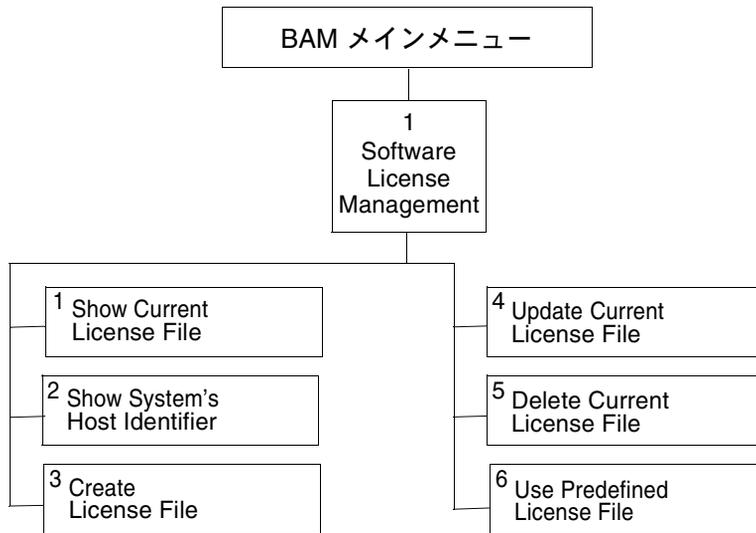


図 2-2 ソフトウェアライセンス管理のメニュー構造

ソフトウェアライセンスの設定については、『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。

次の図は、System Environments メニューです。

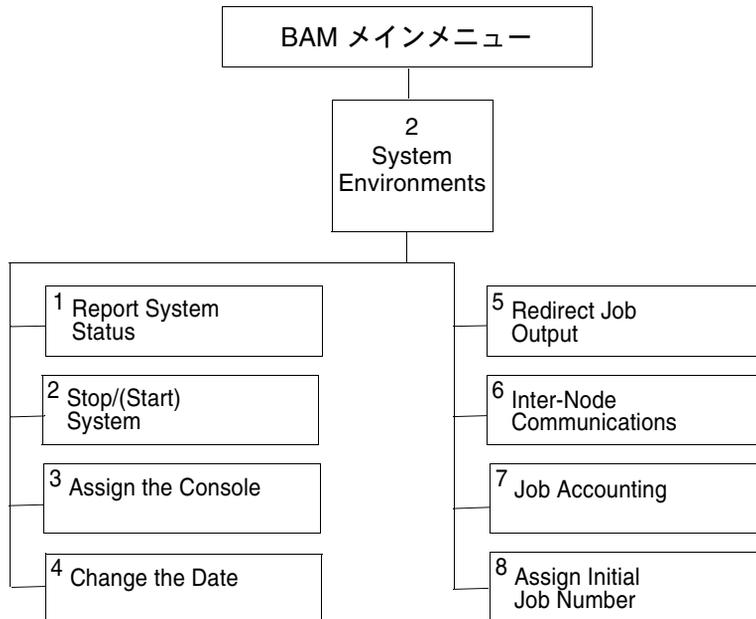


図 2-3 システム環境メニュー構造

次の図は、アプリケーションとサブシステムメニューです。

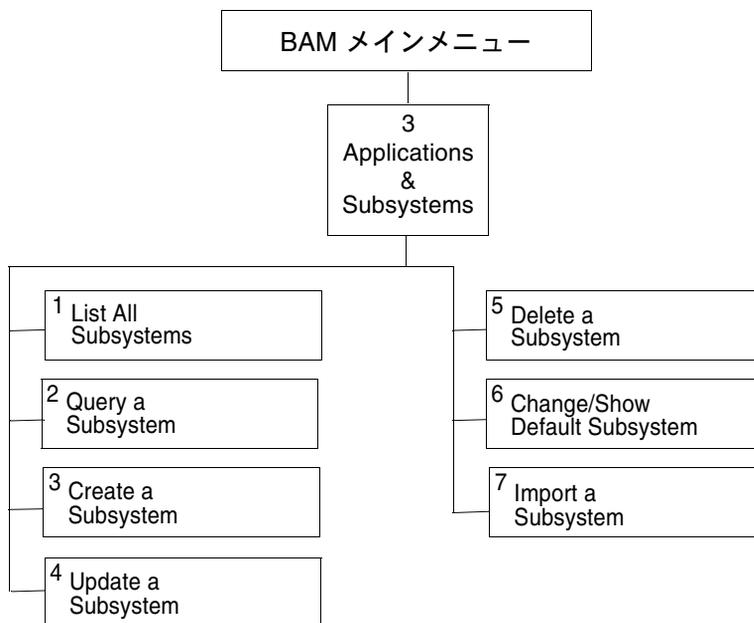


図 2-4 アプリケーションとサブシステムのメニュー構造

次の図は、セキュリティーとユーザーの設定メニューです。

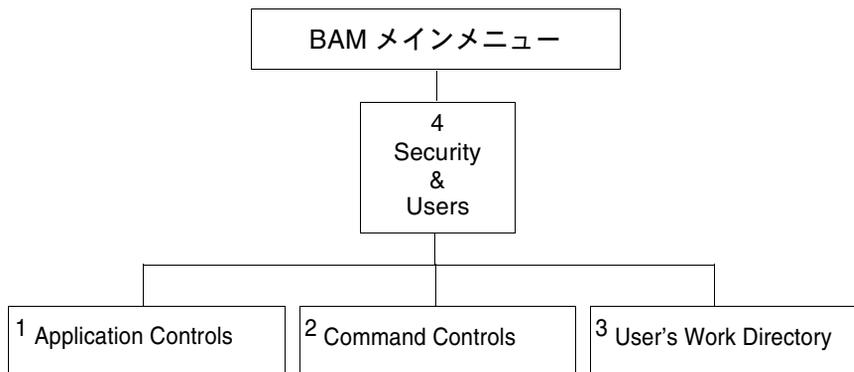


図 2-5 セキュリティーとユーザーの設定メニュー構造

次の図は、ジョブクラスとアクティビティの設定メニューです。

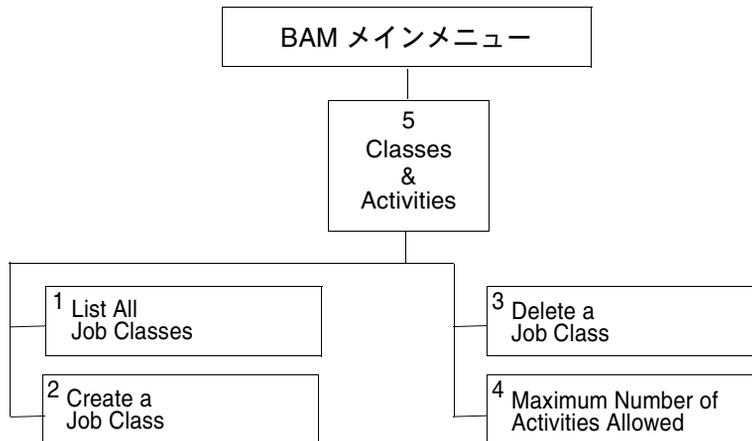


図 2-6 ジョブクラスとアクティビティの設定メニュー構造

次の図は、問題特定メニューです。

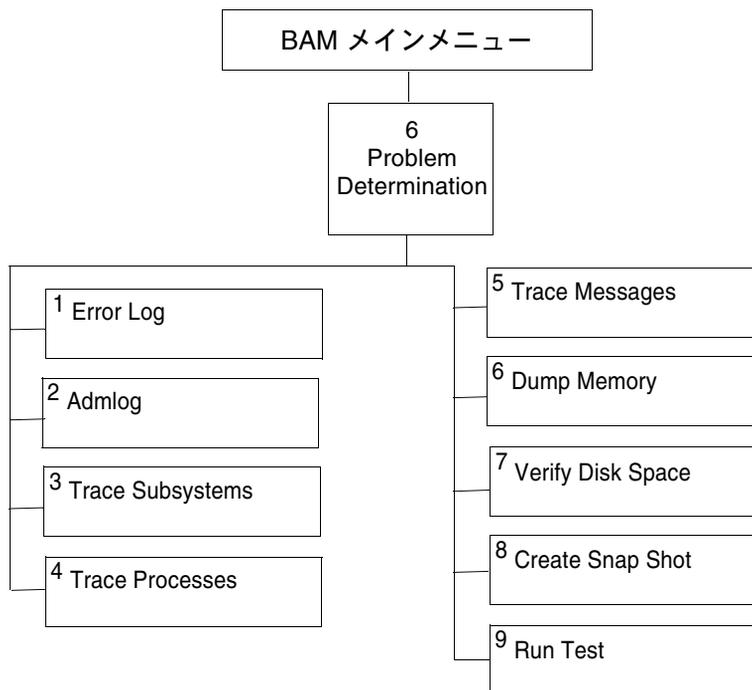


図 2-7 問題特定のメニュー構造

---

# BAM の起動

## ▼ BAMを起動する

1. ノードをインストールする必要があります。  
詳細は、『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。
2. ノードのインストールディレクトリに変更し、`batchenv` ファイルに基づいてノードの環境を設定します。

```
$ cd /pkgs/node1  
$ . ./batchenv
```

---

注 – 先行するドット文字と `./batchenv` との間には空白文字が必要です。

---

3. 次のいずれかの方法で BAM ユーティリティーを起動します。
  - コマンドプロンプトで、`bam` コマンドを実行します。次に例を示します。

```
[node1] # bam
```

- `ebmx` コマンドを入力し、「BAM」アイコンをクリックして、Sun MBM メインメニューを表示します。

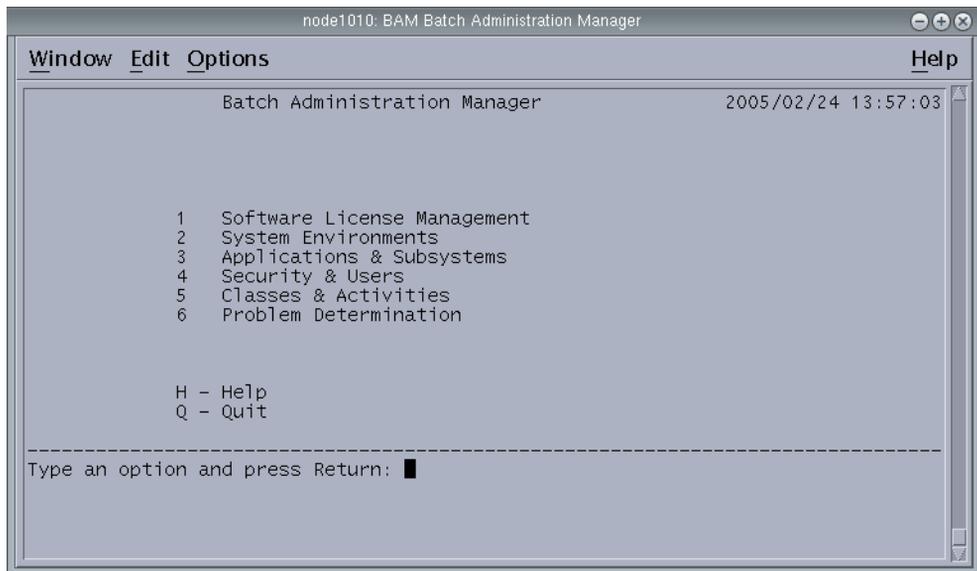


図 2-8 BAM メインメニュー

4. ヘルプ情報が必要な場合は、**H** と入力すると BAM のヘルプファイルが開き、各メニュー項目の簡潔な説明を参照できます。
  - Return キーを押すと、1 行ずつ後続の情報が表示されます。
  - スペースバーを 1 回押すと、1 画面ずつ後続の情報が表示されます。
  - ヘルプの表示を終了するには、**q** キーを押します。
  - **q** キーを押すか、すべての画面を表示したあと、Return キーを押すと、BAM メインメニューに戻ります。

## 第3章

# ノードの設定

ノードをインストールしたら、サイトの要件に基づいてジョブを実行できるようにノードを設定する必要があります。ノード全体のパラメータ設定では、実際に試しながら調整できます。たとえば、同時に実行するバッチジョブの最大数を決定する必要があります。通常、本番稼働に必要な値を特定できるまでは、デフォルトでも十分です。

ノードの設定に使用する BAM メニューオプションを図 3-1 に示します。この章の内容は、次のとおりです。

- 12 ページの「ノードの起動と停止」
- 14 ページの「システム状態および有効なサブシステムの一覧表示」
- 18 ページの「ジョブのログ管理」
- 20 ページの「コンソールの割り当て」
- 27 ページの「ジョブの履歴ファイルのリダイレクト」
- 28 ページの「初期ジョブ番号の割り当て」

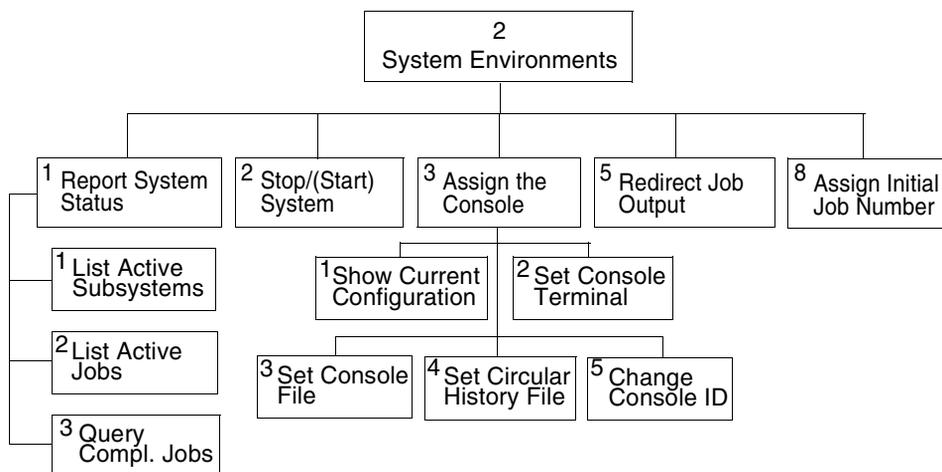


図 3-1 システム環境メニュー構造

## ノードの起動と停止

ノードの起動および停止は、Sun MBM メインメニューまたは BAM メインメニューから行えます。この節では、BAM メインメニューの使用法について説明します。Sun MBM メインメニューの使用法については、『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。

### ▼ ノードを起動する

1. BAM メインメニューを表示します。
2. オプション「2 System Environments」を選択します。
3. BAM の System Environments メニューでオプション「2」を選択します。

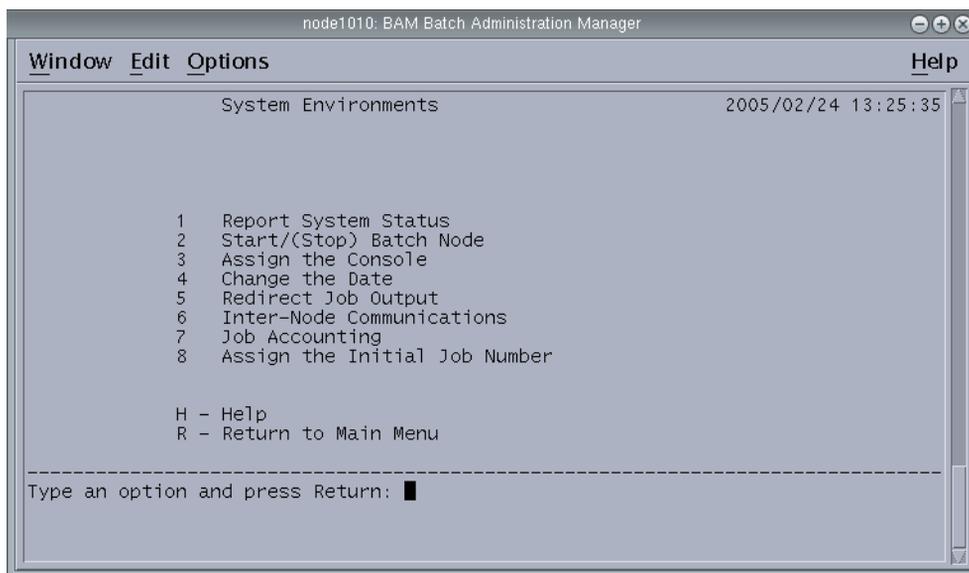


図 3-2 BAM System Environments メニュー

起動中に次のようなメッセージが表示されます。

コード例 3-1 ノードの起動メッセージ

```
OS1004(I) Starting Sun MBM Batch Node (Version 10.1-01/31/2005)
OS1005(I) Startup in Progress
OS1006(I) Startup of ebmmd daemon executed
OS1006(I) Startup of lgdem daemon executed
OS1006(I) Startup of vcf daemon executed
OS1006(I) Startup of psg daemon executed
OS1006(I) Startup of bqm daemon executed
OS1007(I) Startup Completed

Restoring Class Configuration ...
```

4. Return キーを押し、BAM メインメニューを表示します。

## ▼ ノードを停止する

1. BAM メインメニューを表示します。
2. オプション「2 System Environments」を選択します。
3. BAM の System Environments メニューでオプション「2」を選択します。  
次のような停止メッセージが表示されます。

コード例 3-2 ノードの停止メッセージ

```
Saving Job Class Configuration

OS1000(I) Shutdown Started
OS1001(I) Shutdown in Progress
OS1002(I) Shutdown of bqm daemon
OS1002(I) Shutdown of psg daemon
OS1002(I) Shutdown of vcf daemon
OS1002(I) Shutdown of lgdem daemon
OS1002(I) Shutdown of ebmmd daemon
OS1003(I) Shutdown Completed
```

4. Return キーを押し、BAM メインメニューを表示します。

# システム状態および有効なサブシステムの一覧表示

## ▼ システム状態メニューを表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Report System Status」を選択します。  
「Report System Status」メニューが表示されます。

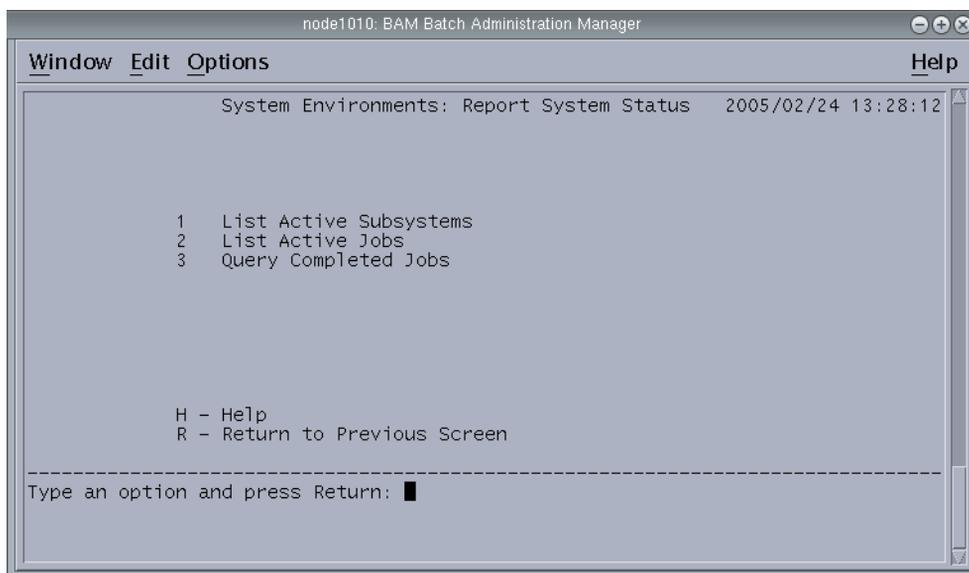


図 3-3 BAM Report System Status メニュー

## ▼ 有効なサブシステムを一覧表示する

1. 「Report System Status」メニューで、オプション「1 List Active Subsystems」を選択します。  
有効なサブシステムを示す次の画面が表示されます。

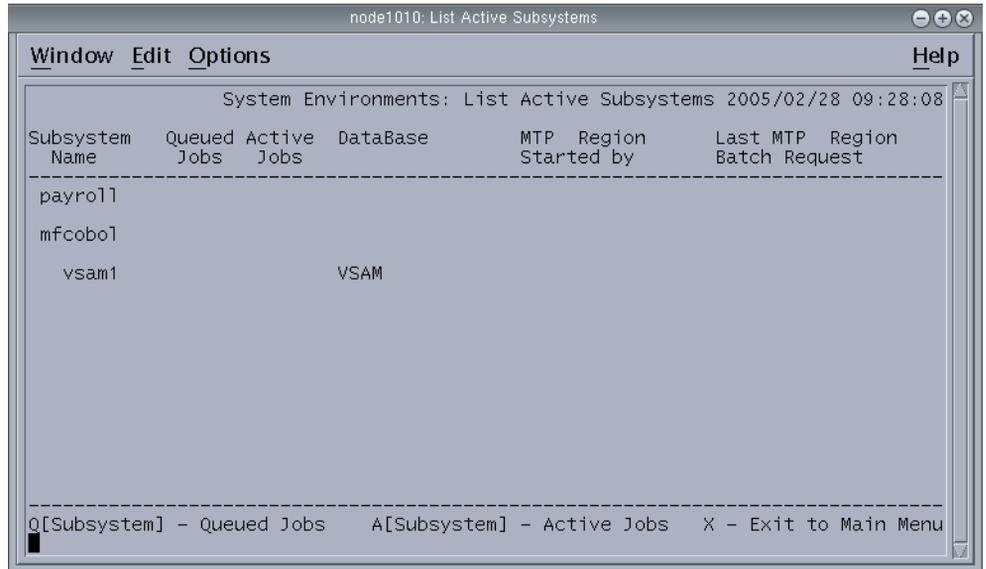


図 3-4 List Active Subsystems 画面

2. X と入力して Return キーを押し、「Report System Status」メニューを表示します。

## ▼ 有効なジョブを一覧表示する

1. 「Report System Status」メニューで、オプション「2 List Active Jobs」を選択します。

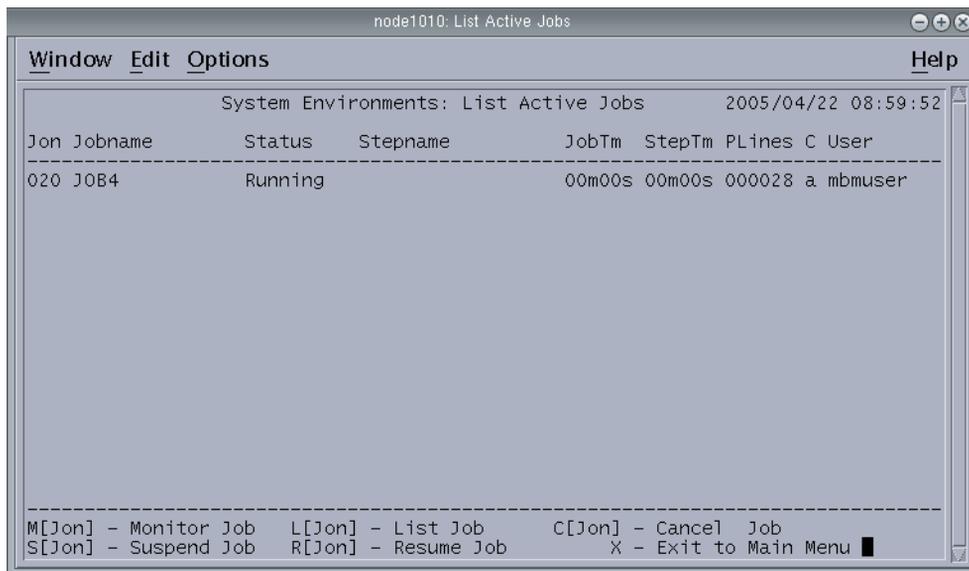


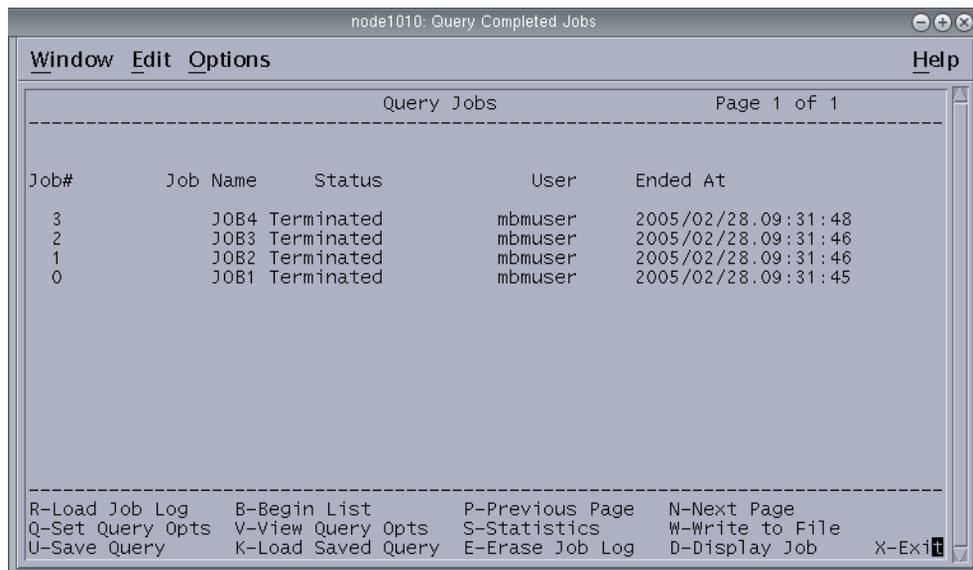
図 3-5 List Active Jobs 画面

2. X と入力して Return キーを押し、「Report System Status」メニューを表示します。

## ▼ 完了したジョブを一覧表示する

1. 「Report System Status」メニューで、オプション「3 Query Completed Jobs」を選択します。

次の画面に完了したジョブが表示されます。



The screenshot shows a terminal window titled 'node1010: Query Completed Jobs'. The window has a menu bar with 'Window', 'Edit', 'Options', and 'Help'. The main content area displays a table of query jobs. The table has five columns: 'Job#', 'Job Name', 'Status', 'User', and 'Ended At'. There are four rows of data, all with a status of 'Terminated'. Below the table, there is a footer area with various navigation and utility options like 'R-Load Job Log', 'B-Begin List', 'P-Previous Page', etc.

Job#	Job Name	Status	User	Ended At
3	JOB4	Terminated	mbmuser	2005/02/28.09:31:48
2	JOB3	Terminated	mbmuser	2005/02/28.09:31:46
1	JOB2	Terminated	mbmuser	2005/02/28.09:31:46
0	JOB1	Terminated	mbmuser	2005/02/28.09:31:45

Footer options:  
R-Load Job Log    B-Begin List    P-Previous Page    N-Next Page  
Q-Set Query Opts    V-View Query Opts    S-Statistics    W-Write to File  
U-Save Query    K-Load Saved Query    E-Erase Job Log    D-Display Job    X-Exit

図 3-6 Query Completed Jobs 画面

2. X と入力して Return キーを押し、「Report System Status」メニューを表示します。

---

## ジョブのログ管理

Sun MBM ジョブのログファイルには、処理されたバッチジョブに関する状態および統計情報が書き込まれます。ジョブのログファイルは追加モードで開かれるので、ファイルサイズが非常に大きくなることがあります。したがって、ファイルのサイズを定期的に小さくする必要があります。ディスク容量の確認方法については、146ページの「ディスク容量の確認」を参照してください。

この節では、ジョブのログ内容の保存方法、ジョブのログファイルの消去方法、および保存されているジョブのログファイルの表示方法について説明します。

### ▼ 完了したジョブの一覧表示画面からジョブのログファイルを保存する

1. 「Query Completed Jobs」画面 (図 3-6) で「U」を選択し、ログ内容を保存します。
2. プロンプトにバックアップファイルの名前を入力します。  
次に例を示します。

```
Enter Backup File Name:pathname/filename
```

3. X と入力して Return キーを押し、「Report System Status」メニューを表示します。

### ▼ コマンド行からジョブのログファイルを保存する

1. Sun MBM メインメニューで、「Command Prompt」アイコンをクリックします。
2. コマンドおよびバックアップファイル名を入力します。  
たとえば、/bkup/log.71500 というバックアップファイルを作成するには、次のように入力します。

```
$ bam BackupJobLog=/bkup/log.71500
```

3. このファイルを表示するには、ebmx と入力して「Completed Jobs」アイコンをクリックします。

## ▼ ジョブのログファイルを消去する

1. 「Query Completed Jobs」画面 (図 3-6) で「E」を選択します。
2. 図 3-7 のメニューが表示されたら、次のオプションのいずれかを入力します。
  - 1 は、次回のバッチノード起動時にジョブのログファイルを消去します。
  - 2 は、バッチノードを起動するたびにジョブのログファイルを消去します。
  - 3 は、ジョブのログファイルを消去しません。

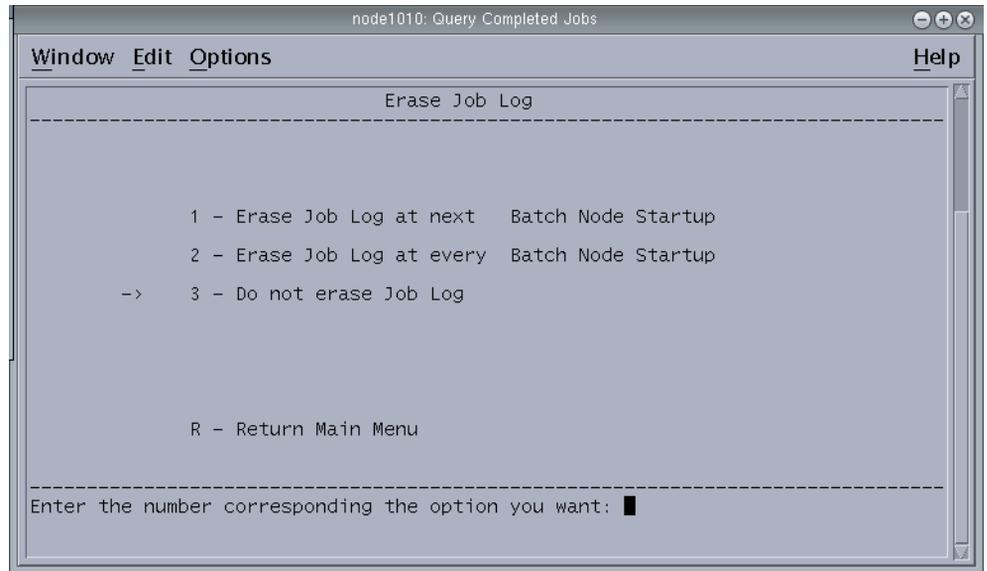


図 3-7 Erase Job Log メニュー

3. R と入力して Return キーを押し、「Query Completed Jobs」画面を表示します。
4. X と入力して Return キーを押し、「Report System Status」メニューを表示します。
5. ノードを再起動して、選択したオプションを適用します。

## 保存されているジョブログファイルの表示

バックアップファイルに保存したジョブは、「Load Saved Query」オプションを使用して表示できます。

### ▼ 保存されているジョブログファイルを表示する

1. 「Query Completed Jobs」画面 (図 3-6) で「K」を選択します。
2. プロンプトに、保存されているバックアップファイルの名前を入力します。  
たとえば、`$HOME/batch/log.52798` という名前のファイルが保存されている場合、次のように入力します。

```
Enter Saved Backup File Name:$HOME/batch/log.52798
```

3. オプション「R Load Job Log」を選択します。  
このオプションでは、現在のバックアップファイルが閉じられ、保存されているジョブログの内容が表示されます。
4. X と入力して Return キーを押し、「Report System Status」メニューを表示します。

---

## コンソールの割り当て

Sun MBM コンソール端末は、ノードに対してオペレータコンソールとして定義された端末装置です。コンソール端末は、その設定に基づいて、Sun MBM システムメッセージまたはジョブの起動/終了メッセージ、あるいはその両方を表示します。

Sun MBM コンソールファイル `$SYSINDIR/ebm_console` は、Sun MBM コンソール端末と同じメッセージタイプのメッセージを格納するシリアルファイルです。cat や tail などのオペレーティングシステムコマンドを使用してメッセージを表示できます。

Sun MBM 履歴ファイル `$SYSINDIR/HistoryFile` は、ジョブの起動および終了メッセージを格納する循環ファイルです。このファイルを表示するには、`lststs` または `histprt` コマンドを使用します。

---

**注** – この節の手順に従ってコンソールを割り当てたあとは、変更した設定を適用するためにノードを再起動する必要があります。

---

## ▼ コンソールを割り当てる

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」→「Assign the Console」を選択します。

図 3-8 に示す画面が表示されます。

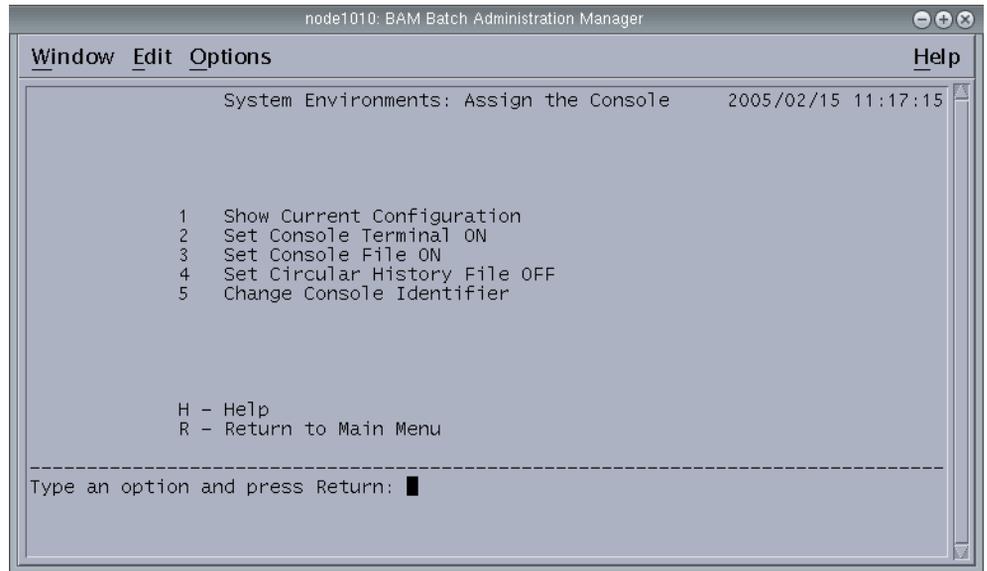


図 3-8 Assign the Console メニュー

「Assign the Console」メニューには、次の操作ができる 5 つのオプションがあります。

- 現在の設定を表示する
- コンソール端末を ON または OFF に設定する
- コンソールファイルを ON または OFF に設定する
- 循環履歴ファイルを ON または OFF に設定する
- コンソール識別子を変更する

このメニューで変更した設定内容は、次にノードを再起動したときに有効になります。

## ▼ 現在のコンソール設定を表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Assign the Console」 → 「Show Current Configuration」を選択します。

コンソールの設定状態が図 3-9 の画面に表示されます。

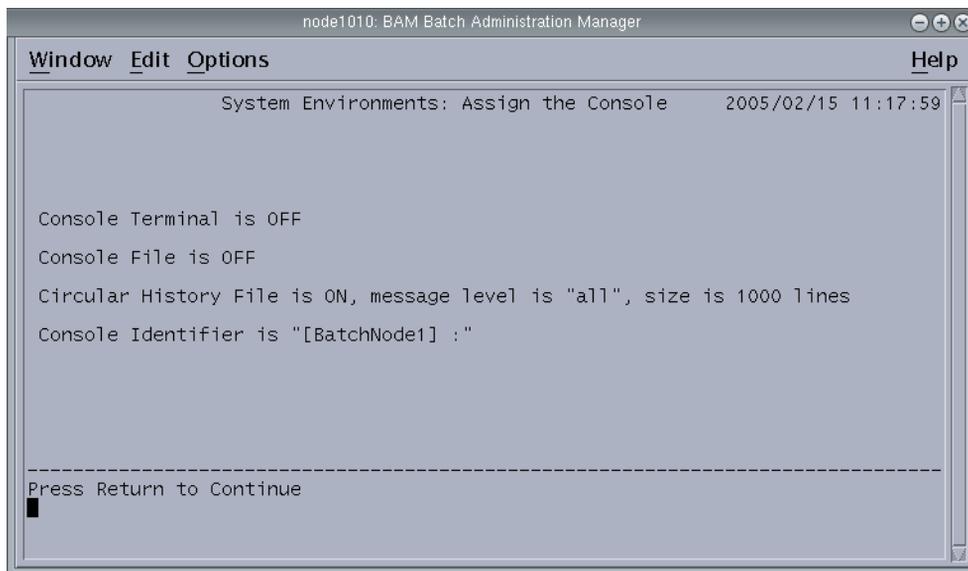


図 3-9 Assign the Console Configuration 画面

3. Return キーを押し、「Assign the Console」メニューを表示します。

## コンソール端末の設定

コンソール端末を有効化するには、Sun MBM コンソールメッセージを表示させる端末で `tty` コマンドを入力します。このコマンドにより、その端末のフルパス名、たとえば `/dev/pts/ttyve` が戻されます。

---

注 – 同じコンソール端末を複数のノードで共有できます。

---

## ▼ コンソール端末を ON に設定する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Assign the Console」 → 「Set Console Terminal ON」を選択します。  
表示される 2 つのプロンプトに対し、次のように実行します。
3. コンソール端末装置のフルパス名を入力し、Return キーを押します。
4. 次のメッセージレベルのオプションを 1 つ入力し、Return キーを押します。
  - S - Sun MBM システムメッセージのみ (sysout)
  - J - ジョブの起動/終了メッセージのみ
  - A - システムメッセージおよびジョブの起動/終了メッセージの両方
5. 確認画面が表示されたら、Return キーを 2 回押して「System Environments」メニューを表示します。
6. ノードを再起動します。

## ▼ コンソール端末を OFF に設定する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Assign the Console」を選択します。
3. オプション「2 Set Console Terminal OFF」を選択します。
4. 確認画面が表示されたら、Return キーを 2 回押して「System Environments」メニューを表示します。
5. ノードを再起動します。

## コンソールファイルの ON の設定

`$$SYSINDIR/ebm_console` はシリアルファイルです。このファイルはノード固有のディレクトリにあるので、複数のノードで共有できません。コンソールファイルを有効化すると、追加モードで開かれます。したがって、ファイルのサイズが非常に大きくなることがあるので、定期的にサイズを小さくする必要があります。ディスク容量の確認方法については、146 ページの「ディスク容量の確認」を参照してください。

## ▼ コンソールファイルを ON に設定する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」 → 「Assign the Console」 → 「Set Console File ON」を選択します。  
メッセージレベルのプロンプトが表示されます。
3. 次のいずれかのメッセージレベルを入力します。
  - S - Sun MBM システムメッセージのみ (sysout)
  - J - ジョブの起動/終了メッセージのみ
  - A - システムメッセージおよびジョブの起動/終了メッセージの両方
4. Return キーを押します。
5. 確認画面が表示されたら、Return キーを 2 回押して「System Environments」メニューを表示します。
6. ノードを再起動します。

## ▼ コンソールファイルを OFF に設定する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」 → 「Assign the Console」 → 「Set Console File OFF」を選択します。
3. Return キーを 2 回押し、「System Environments」メニューを表示します。
4. ノードを再起動します。

## 循環履歴ファイルの設定

循環履歴ファイルが有効な場合、Sun MBM ユーティリティ `lststs` または `histprt` を使用してこのファイルにアクセスできます。これらのコマンドの詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。

## ▼ 循環履歴ファイルを ON に設定する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Assign the Console」 → 「Set Circular History File ON」を選択します。
3. メッセージレベルのプロンプトが表示されたら、次のとおり実行します。
  - a. 次のメッセージレベルのオプションを 1 つ入力し、Return キーを押します。
    - J - ジョブの起動/終了メッセージのみ
    - A - Sun MBM システムメッセージおよびジョブの起動/終了メッセージ
  - b. 既存のエントリの上書きが開始されるまでの、循環履歴ファイルに格納できる行数を指定し、Return キーを押します。

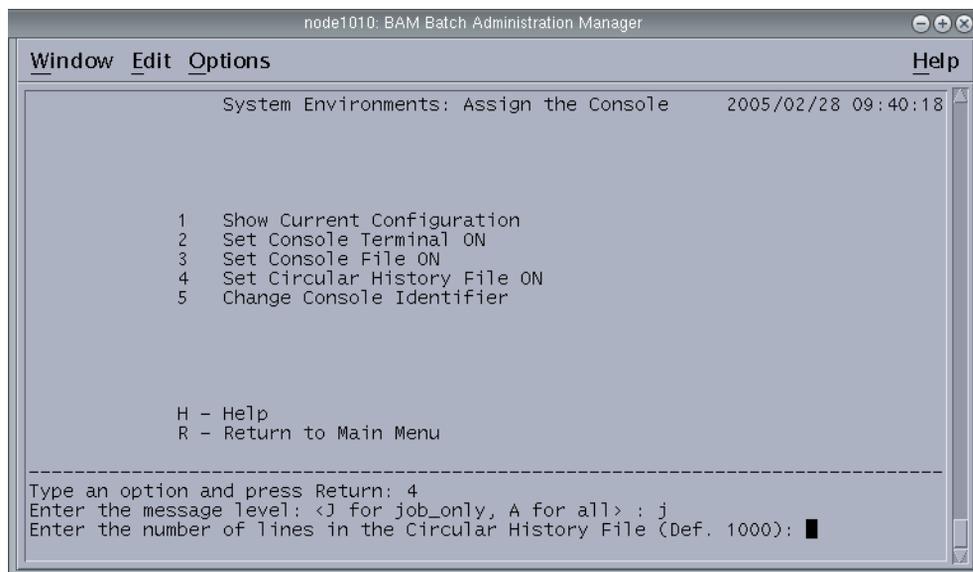


図 3-10 循環履歴ファイルを ON に設定する画面

4. 確認画面が表示されたら、Return キーを 2 回押して「System Environments」メニューを表示します。
5. ノードを再起動します。

## ▼ 循環履歴ファイルを OFF に設定する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」 → 「Assign the Console」 → 「Set Circular History File OFF」を選択します。
3. オプション「4」を選択すると、確認画面が表示されます。
4. Return キーを 2 回押し、「System Environments」メニューを表示します。
5. ノードを再起動します。

## コンソール識別子の変更

複数の Sun MBM ノードで 1 つのコンソール端末およびコンソールファイルを共有する場合、一意のコンソール識別子を割り当てることにより、どのメッセージが特定ノードから発行されているかを特定できます。

## ▼ コンソール識別子を変更する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」 → 「Assign the Console」 → 「Change Console Identifier」を選択します。
3. プロンプトにノードの新しいコンソール識別子を入力し、Return キーを押します。
4. 確認画面が表示されたら、Return キーを 2 回押し「System Environments」メニューを表示します。
5. ノードを再起動します。

## ジョブの履歴ファイルのリダイレクト

この節では、ジョブの履歴ファイルが作成される、ノードのデフォルトディレクトリの変更方法について説明します。

注 – ファイルシステムのオーバーフローを防ぐには、デフォルトの履歴ディレクトリを監視する必要があります。詳細は、146 ページの「ディスク容量の確認」を参照してください。

### ▼ デフォルトのジョブ履歴ディレクトリを変更する

1. 9 ページの「BAM の起動」の説明に従って、BAM を起動します。
2. 「System Environments」 → 「Redirect Job Output」を選択します。
3. 図 3-11 の画面が表示されたら、`y` と入力してディレクトリを変更します。

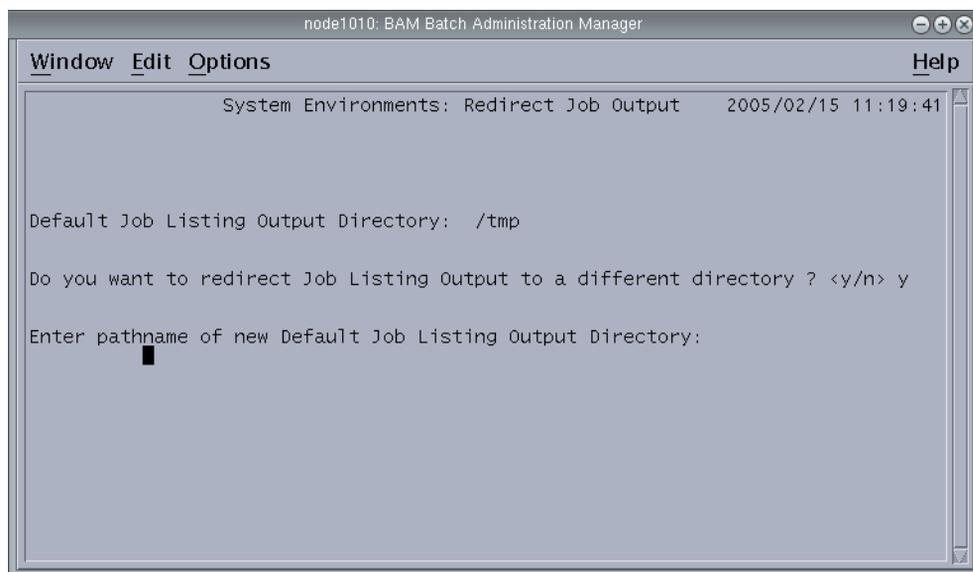


図 3-11 Redirect Job Output 画面

4. 新しいディレクトリのパスを入力して Return キーを押すと、確認画面が表示されます。
5. Return キーを押して、「System Environments」メニューを表示します。

## 初期ジョブ番号の割り当て

ユーザーがジョブをサブミットすると、使用可能な最も小さな値のジョブ番号が Sun MBM によって割り当てられます。SYSIN ファイルが空であれば、ノードを再起動するたびに初期ジョブ番号は 0 になります。初期ジョブ番号をユーザーが割り当てると、ノードを再起動するたびに初期ジョブ番号が更新されるという動作を変更できます。これにより、再起動してもジョブ番号は連続した番号になります。

---

注 - 初期ジョブ番号を設定する前に、ノードを停止する必要があります。

---

### ▼ 初期ジョブ番号を割り当てる

1. 9 ページの「BAM の起動」の説明に従って、BAM を起動します。
2. 「System Environments」→「Assign the Initial Job Number」を選択します。
3. 次の画面が表示されたら、次にノードを起動したときに割り当てるジョブ番号を入力し、Return キーを押します。  
D と入力すると、デフォルトの動作になります。

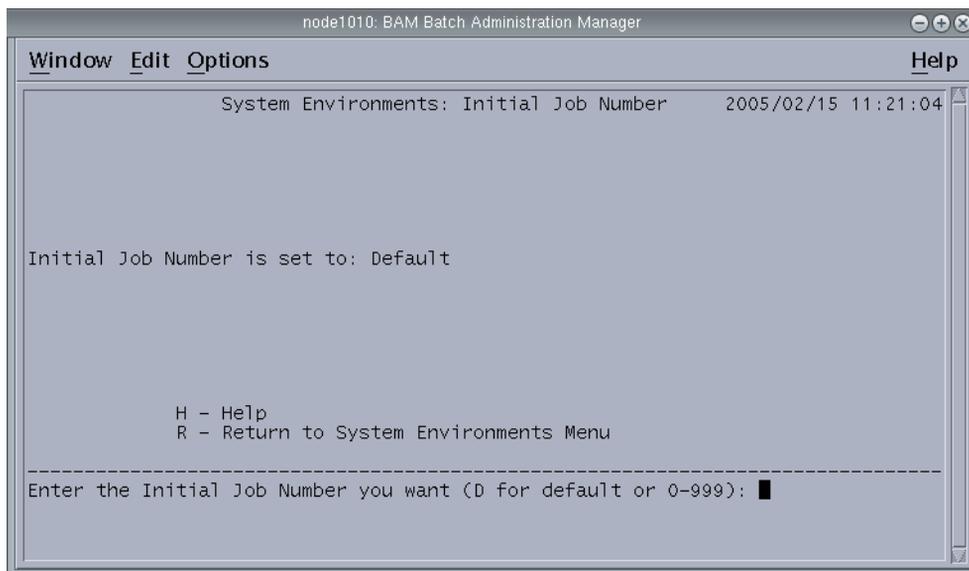


図 3-12 Initial Job Number 画面

4. Return キーを押して、「System Environments」メニューを表示します。



## 第4章

# 日付および時刻の変更

この章では、Sun MBM または Sun MTP 領域の日付と時刻の設定方法、領域の同期化の方法、および言語環境の管理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。

- 32 ページの「Sun MBM の現在の日付の変更」
- 39 ページの「Sun MBM の時刻形式の変更」
- 40 ページの「言語環境の管理」

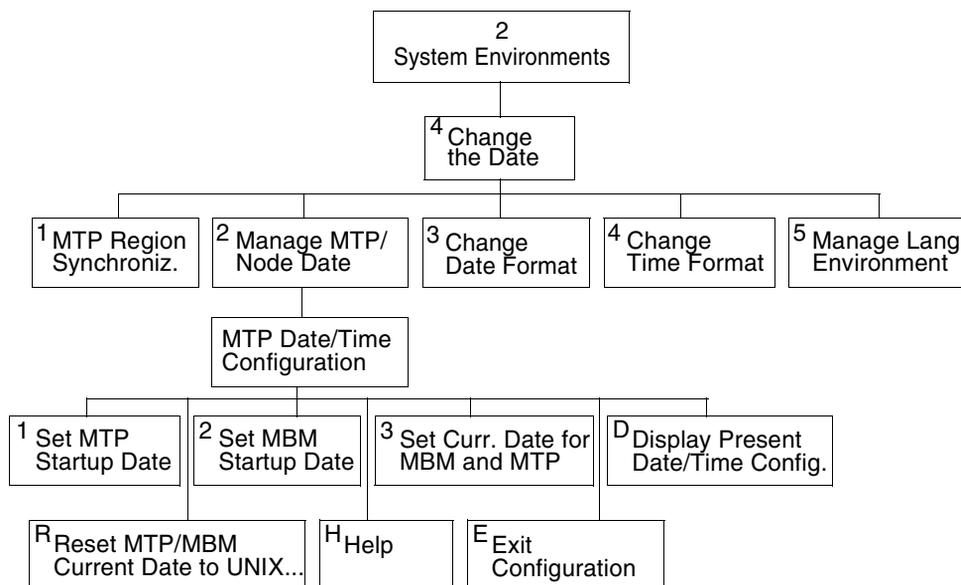


図 4-1 日付/時刻の変更のメニュー構造

---

## Sun MBM の現在の日付の変更

この節では、Sun MBM または Sun MTP 領域の日付の設定方法について説明します。



---

**注意** – この機能を使用すると、標準の COBOL または Open PL/I の日付/時刻ルーチンによって返される日付および時刻に影響を与えるおそれがあります。したがって、現在の日付/時刻は、テスト以外の目的では変更しないでください。

---

### 領域の同期化

デフォルトでは、BAM を使用して Sun MBM ノードと同期をとることができる Sun MTP 領域は 1 つだけです。複数の領域を同一のノードと同期をとるには、次の作業が必要です。

1. この節の手順を実行して、Sun MBM と Sun MTP のシステムの日付の同期をとります。
2. 同期をとった領域の \$KIXSYS/datefile ファイルを、その他のすべての領域の \$KIXSYS ディレクトリにコピーします。
3. すべての領域を再起動します。

## ▼ 領域を同期化する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」→「Change the Date」を選択します。
3. オプション「4」を選択し、図 4-2 のメニューを表示します。

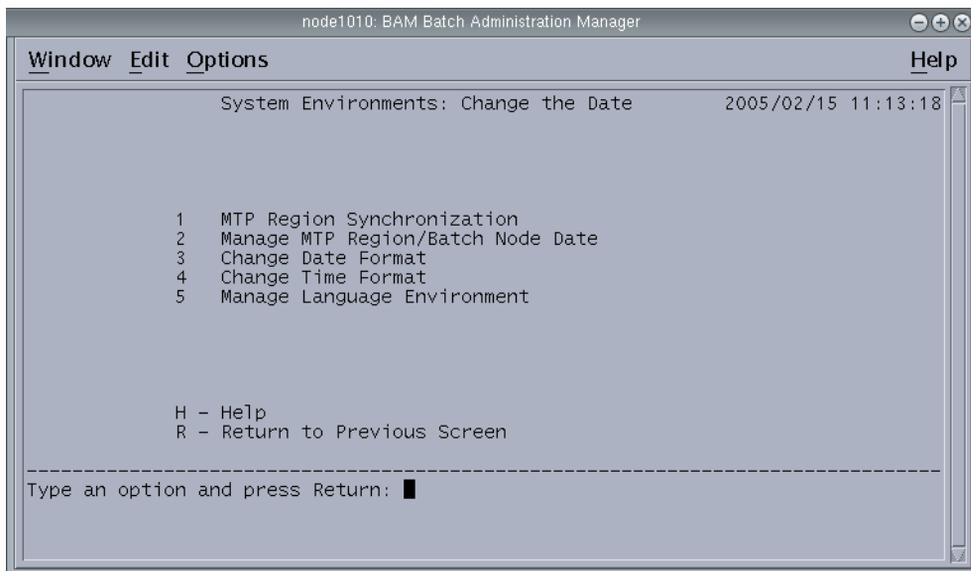


図 4-2 Change the Date 画面

4. オプション「1」を選択します。
5. 図 4-3 の画面が表示されたら、次のように実行します。
  - a. 同期をとる領域を定義するため、`y` と入力します。
  - b. 領域のフルパス名を入力します。それは領域の `KIXSYS` 環境変数の値です。

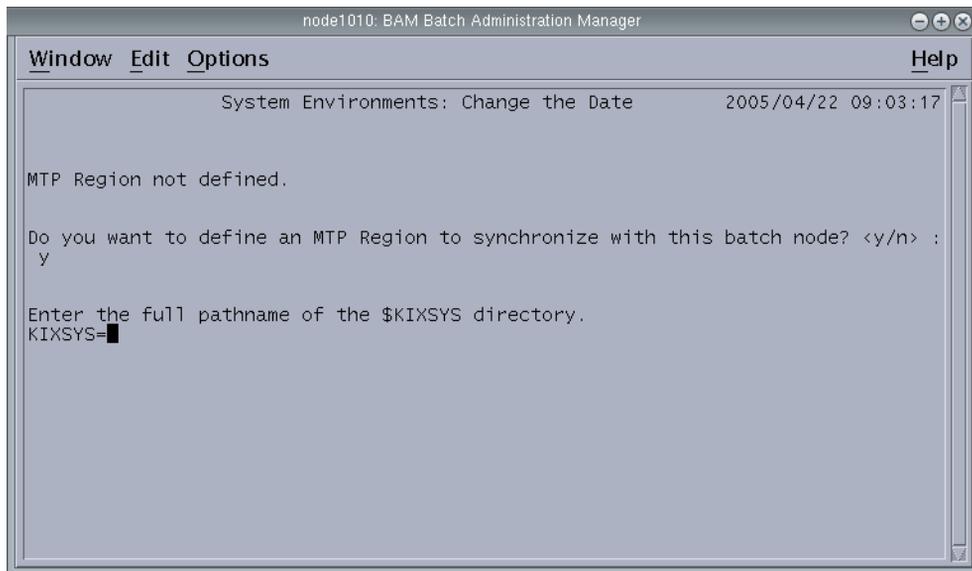


図 4-3 Sun MTP 領域同期化の画面

6. Return キーを押し、「Change the Date」メニューを表示します。

## ▼ Sun MTP/Sun MBM の日付を設定する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Change the Date」 → 「Manage MTP Region/Batch Node Date」を選択します。
3. 「日付/時刻の構成ユーティリティー」メニューで、オプション「3」を選択します。

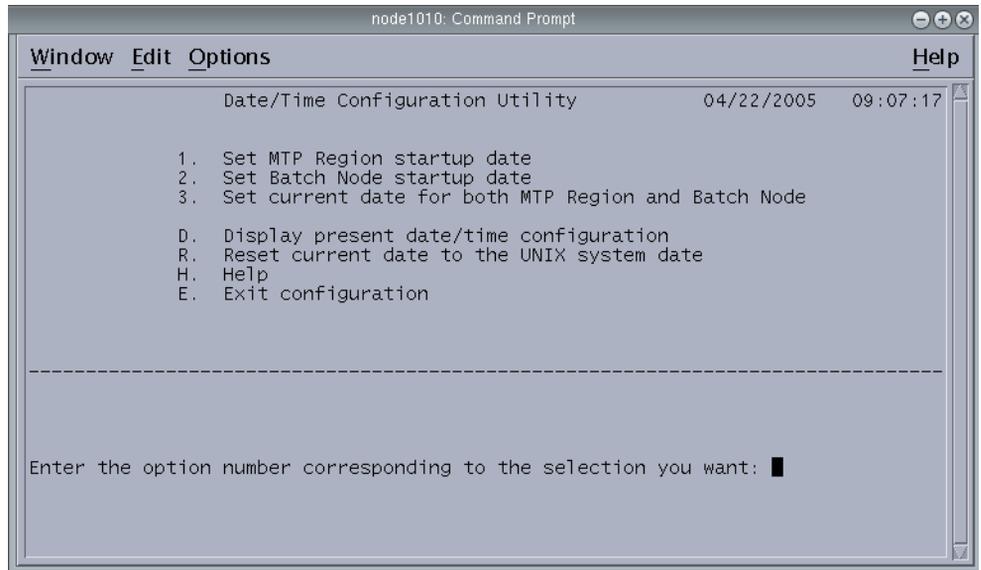


図 4-4 「Sun MTP 日付/時刻の構成ユーティリティー」メニュー

4. 図 4-5 の画面が表示されたら、そこに示されている形式で日付および時刻を入力し、Return キーを押します。

この日付は、Sun MBM と Sun MTP の両方に使用されます。

入力内容が適切であれば、確認メッセージが表示されます。

```
node1010: Command Prompt
Window Edit Options Help
MTP Region and batch Node Current Date Configuration
-----
The date entered on this panel will be treated as the current date
for BOTH MTP Region and Batch Node. By "current", we mean that the date/time
shall be transparently incrementing with each passing second even if
the Batch Node and/or the MTP Region are not running. This will ensure that
the Batch Node and the MTP Region date/time are in sync.

For example, if you specify 12/31/1999-23:50:00 now and bring
the MTP Region and the Batch Node up after 20 minutes, their date/time will be
01/01/2000-00:10:00

Enter current MTP Region and Batch Node date in the format MM/DD/YYYY-HH:MM:SS
04/22/2005-09:09:40

Successfully set Batch Node current date to 04/22/2005-09:09:40
You will need to restart Batch Node for the new date setting to take effect
Successfully set MTP Region current date to 04/22/2005-09:09:40
You will need to restart MTP Region for the new date setting to take effect
Press <Enter> to continue
█
```

図 4-5 Sun MTP/Sun MBM Current Date Configuration 画面

5. Return キーを押します。
  - Sun MBM と Sun MTP に別個の日付を設定するには、手順 6 から手順 11 を実行します。
  - 同じ日付を設定する場合は、手順 12 に進みます。
6. 「日付/時刻の構成ユーティリティ」メニューで、オプション「1」を選択します。
7. 図 4-6 の画面が表示されたら、そこに示されている形式で日付および時刻を入力し、Return キーを押します。

入力内容が適切であれば、確認メッセージが表示されます。

```
MTP Region Startup Date Configuration
-----
Enter MTP Region Startup Date in the format MM/DD/YYYY-HH:MM:SS
04/22/2005-09:20:30
Successfully set MTP Region startup date to 04/22/2005-09:20:30
You will need to restart MTP Region for the new date setting to take effect
Press <Enter> to continue
```

図 4-6 Sun MTP Startup Date Configuration

8. Return キーを押し、「日付/時刻の構成ユーティリティ」メニューを表示します。
9. オプション「2」を選択します。
10. 次の画面が表示されたら、そこに示されている形式で日付および時刻を入力し、Return キーを押しします。  
入力内容が適切であれば、確認メッセージが表示されます。

```
Batch Node Startup Date Configuration
-----

Enter Batch Node Startup Date in the format MM/DD/YYYY-HH:MM:SS

04/22/2005-09:22:30
Successfully set Batch Node startup date to 04/22/2005-09:22:30
You will need to restart the Batch Node for the new date setting to take effect
Press <Enter> to continue
```

図 4-7 Sun MBM Startup Date Configuration 画面

11. Return キーを押し、「日付/時刻の構成ユーティリティ」メニューを表示します。
12. オプション「R」を選択して、日付を変更します。
13. 画面が表示されたら、次のように実行します。
  - a. オプション「D Display present date/time configuration」を選択し、Sun MBM および Sun MTP の日付を表示します。  
これらの日付は、絶対日付または相対日付です。
  - b. オプション「E Exit configuration」を選択し、「System Environments」の「Change the Date」メニューを表示します。
14. Sun MBM ノードおよび Sun MTP 領域を再起動して、変更を適用します。

## ▼ Sun MBM の日付形式を変更する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」→「Change the Date」→「Change Date Format」を選択します。
3. 「Change the Date」画面が表示されたら、現在の日付形式を変更するために `y` と入力します。

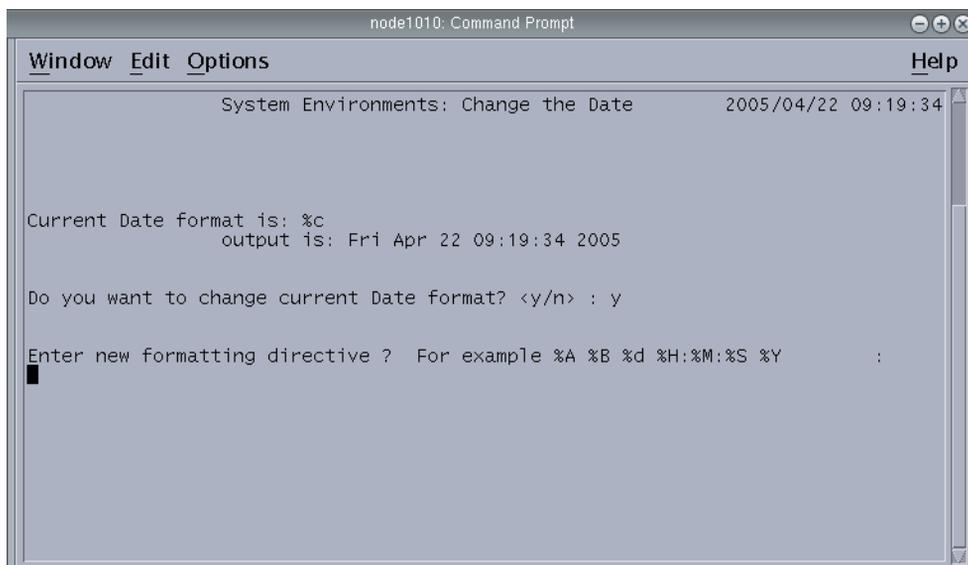


図 4-8 Sun MBM 日付形式の画面

4. 新しい形式設定指示を入力します。  
次の表に、形式設定指示を示します。

表 4-1 日付/時刻の形式設定指示

指示	説明	例
<code>%a</code>	曜日の省略名	Wed
<code>%A</code>	曜日の完全名	Wednesday
<code>%b</code>	月の省略名	Jan
<code>%B</code>	月の完全名	January
<code>%c</code>	現在日時の表示	Wed Jan 12 19:45:58 2000
<code>%d</code>	2桁10進数の日付: 01 ~ 31	04

表 4-1 日付/時刻の形式設定指示 (続き)

指示	説明	例
%e	先頭に 0 を付けない 2 桁 10 進数の日付: "1" ~ "31"	"1", "12"
%H	2 桁 10 進数の時間 (24 時間表示): 00 ~ 23	19
%I	2 桁 10 進数の時間 (12 時間表示): 01 ~ 12	07
%m	2 桁 10 進数の月: 01 ~ 12	01
%M	2 桁 10 進数の分: 00 ~ 59	45
%p	午前/午後: AM または PM	PM
%S	2 桁 10 進数の秒 (うるう秒に対応): 00 ~ 61	58
%T	米国形式の 24 時間表示。%H:%M:%S と同じ	19:45:58
%x	現在日付の表示	01/12/00
%X	現在時刻の表示	19:45:58
%y	下 2 桁 10 進数の年: 00 ~ 99	98
%Y	4 桁 10 進数の年: 1970 ~ 2030	1999

5. Return キーを押し、「Change the Date」メニューを表示します。

## Sun MBM の時刻形式の変更

デフォルトでは、Sun MBM での出力時刻の表示には、MVS ジョブの時刻の形式設定指示が使用されます。Sun MBM 管理者は必要に応じて、表 4-1 の指示を使用して、別の出力時刻形式を定義できます。EBM\_TIME\_FORMAT については、167 ページの「.install ファイルの内容」を参照してください。

### ▼ 時刻形式を変更する

- 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
- 「System Environments」→「Change the Date」→「Change Time Format」を選択します。
- 図 4-9 の画面が表示されたら、現在の時刻形式を変更するために `y` と入力します。

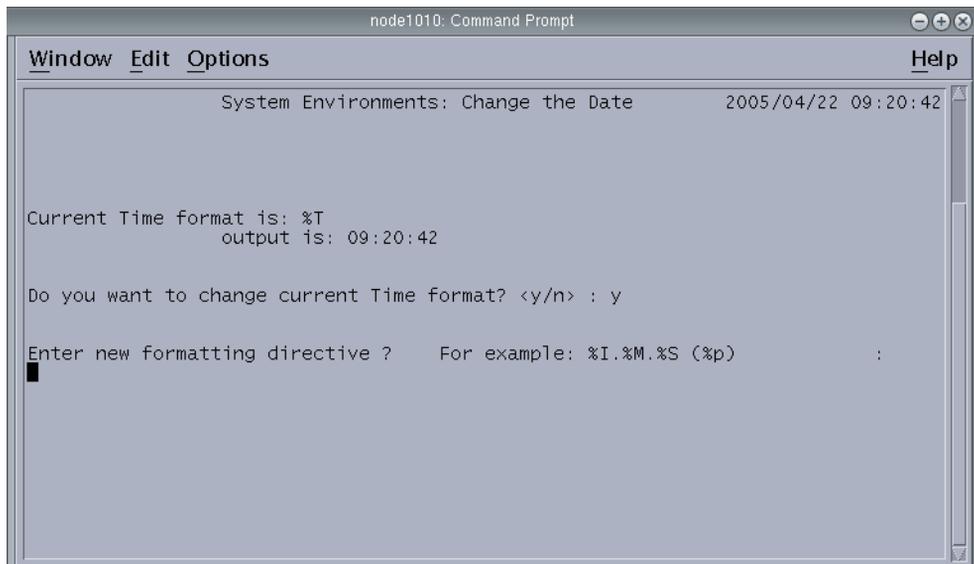


図 4-9 Sun MBM 時刻形式の変更画面

4. 新しい形式設定指示を入力します。
5. Return キーを押し、「Change the Date」メニューを表示します。

---

## 言語環境の管理

Sun MBM システムでは、ロケール時間カテゴリの指定に基づいてジョブと Sun MBM コマンド出力の日付および時刻を表示します。言語環境のオプションを使用すると、システムでサポートされる各国の言語をいずれか 1 つ選択できます。詳細は、167 ページの「.install ファイルの内容」も参照してください。

### ▼ 言語環境を変更する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Change the Date」 → 「Manage Language Environment」を選択します。
3. 図 4-10 の画面が表示されたら、*y* と入力します。

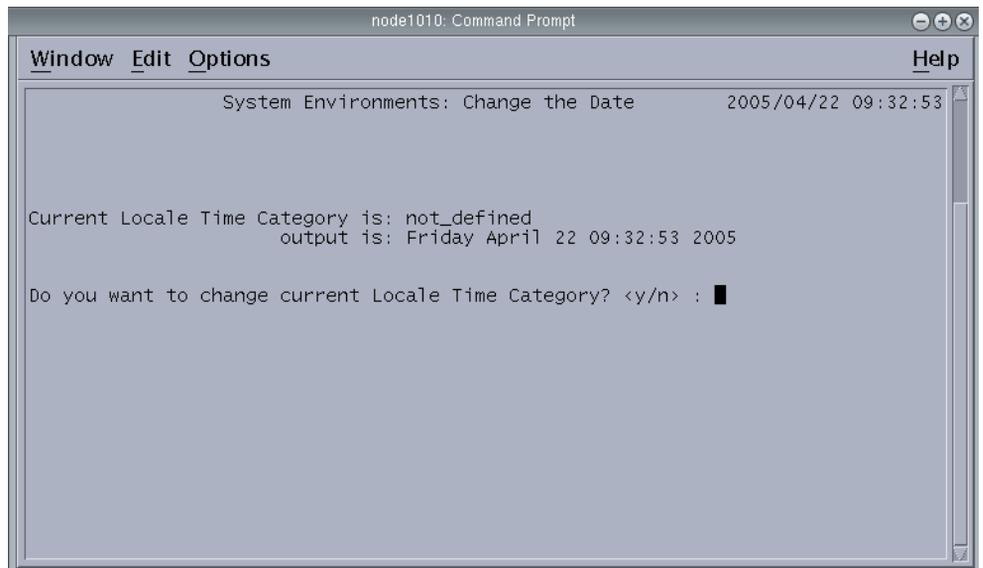


図 4-10 Sun MBM ロケール時間カテゴリの画面

4. 新しいロケール時間カテゴリを入力します。
5. Return キーを押し、「Change the Date」メニューを表示します。



## 第5章

# ノード間の通信

この章では、ノード間通信のための Sun MBM ノードの設定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。

- 44 ページの「使用可能なノードの一覧表示」
- 46 ページの「複数のノードの設定」
- 46 ページの「ノードのサーバー (ローカル) 指定」
- 48 ページの「サーバー (ローカル) 指定の削除」
- 48 ページの「遠隔ノードの定義」
- 50 ページの「遠隔ノードの変更」
- 51 ページの「遠隔ノードの削除」

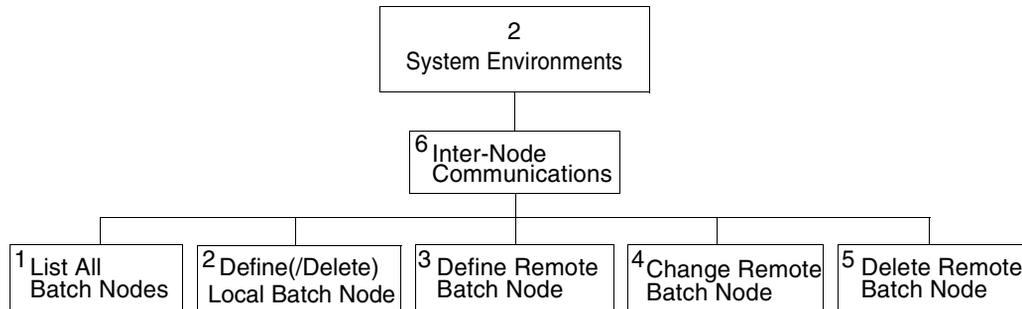


図 5-1 ノード間通信のメニュー構造

---

## 使用可能なノードの一覧表示

すべてのローカルノードおよび遠隔ノードを一覧表示するには、BAM の「System Environments」の「Inter-Node Communications」オプションを使用します。

画面に「local」と表示されるノードは、そこに示されるポート番号のサーバーです。ノード名が表示されるノードは、サーバーとして実行中であれば、そのノードに対して遠隔要求をサブミットできます。

ノードに対して「local」の表示があれば、サーバーとして遠隔要求を受け入れることを意味します。それ以外のシステム表示の場合、そのノードがサーバーとして指定されて実行中であれば、Sun MBM コマンドの `-N` オプションでそのノード名を指定することにより要求をサブミットできることを意味します。`-N` オプションについては、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。

たとえば、同じシステムに2つのノード Node1 および Node2 があるとします。Node1 の設定では、それ自身をポート 3000 のローカルノード (サーバー) として定義します。Node2 の設定では、Node1 をポート 3000 の遠隔ノードとして定義します。これにより、Node2 のユーザーは Node1 に対してジョブをサブミットできます。Node2 の設定はそれ自身をローカルノードとして定義していないので、Node1 のユーザーは Node2 に対してジョブをサブミットできません。

### ▼ 使用可能なノードを一覧表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」→「Inter-Node Communications」を選択します。
3. 図 5-2 の画面が表示されたら、オプション「1」を選択します。

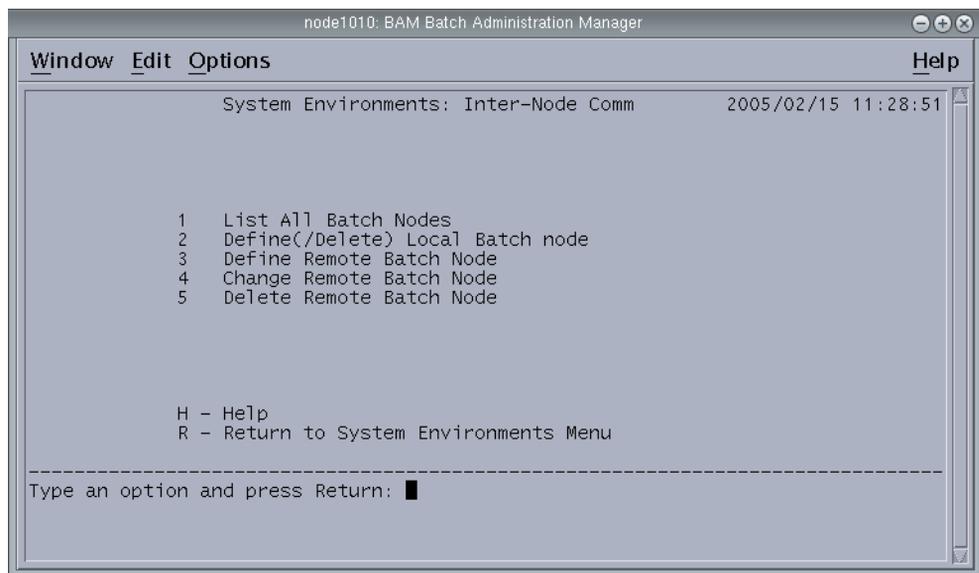


図 5-2 Inter-Node Communications メニュー

4. 次の画面は、使用可能なノードとそのポート番号を示しています。

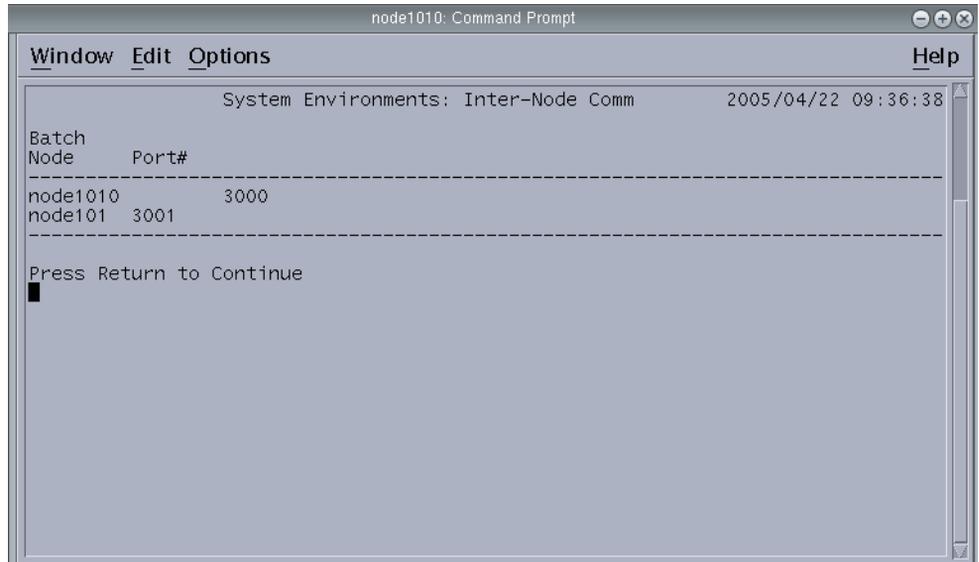


図 5-3 すべてのノードの一覧表示の画面

5. Return キーを押して、「Inter-Node Communications」メニューを表示します。

---

## 複数のノードの設定

同じホスト上に、複数の独立したノードをインストールできます。2つ以上のノードを相互に通信させるには、それらのノードを起動する前に、BAM を使用してローカルおよび遠隔のエントリを定義する必要があります。それにより、ローカルノードから遠隔ノードに対してジョブをサブミットできます。ただし、ユーザーが複数のノードにアクセスし、権限指定のコマンドを遠隔ノードで実行するには、ローカルと遠隔の両ノードに対するアクセス権限を持つ必要があります。詳細は、108 ページの「サブシステムへのアクセス制御」を参照してください。

---

## ノードのサーバー (ローカル) 指定

ローカルノードを定義するには、BAM の「System Environments」の「Inter-Node Communications」オプションを使用します。サーバーとして指定されたノードは、遠隔要求を受け入れます。

### ▼ ノードをサーバーとして指定する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」 → 「Inter-Node Communications」 → 「Define(/Delete) Local Batch Node」を選択します。  
図 5-4 の画面が表示され、ローカルノードを定義するように求められます。
3.  $y$  と入力します。  
ポート番号の入力行が表示されます。
4. そのローカルノード用に割り当てられたポート番号を入力し、Return キーを押します。

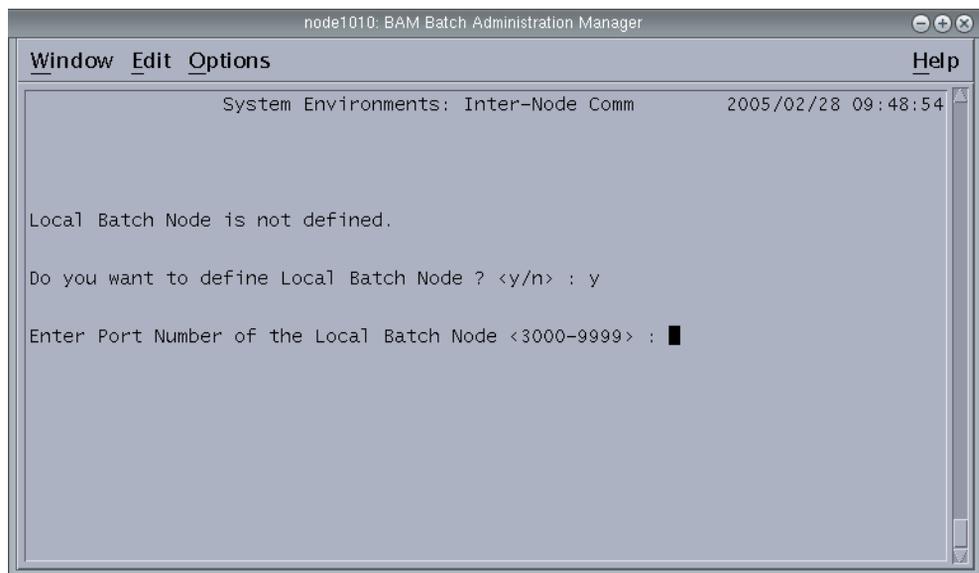


図 5-4 ローカルノードの定義画面

5. 確認画面が表示されたら、Return キーを押して「Inter-Node Communications」メニューを表示します。
6. ノードを再起動して、変更を適用します。

---

## サーバー (ローカル) 指定の削除

### ▼ ノードのサーバー指定を削除する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Inter-Node Communications」 → 「Define/(Delete) Local Batch Node」を選択します。
3. 確認画面が表示されたら、`y` と入力します。  
`y` と入力してそのローカルノードを再起動すると、そのノードはサーバー指定が削除されて遠隔要求を受け付けなくなります。  
また、`-N` オプションをサポートする Sun MBM コマンドでこのオプションを使っても、そのノードを参照できなくなります。
4. 確認画面が表示されたら、Return キーを押して「Inter-Node Communications」メニューを表示します。
5. ノードを再起動して、変更を適用します。

---

## 遠隔ノードの定義

### ▼ 遠隔ノードを定義する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Inter-Node Communications」 → 「Define Remote Batch Node」を選択します。  
次の画面が表示されます。

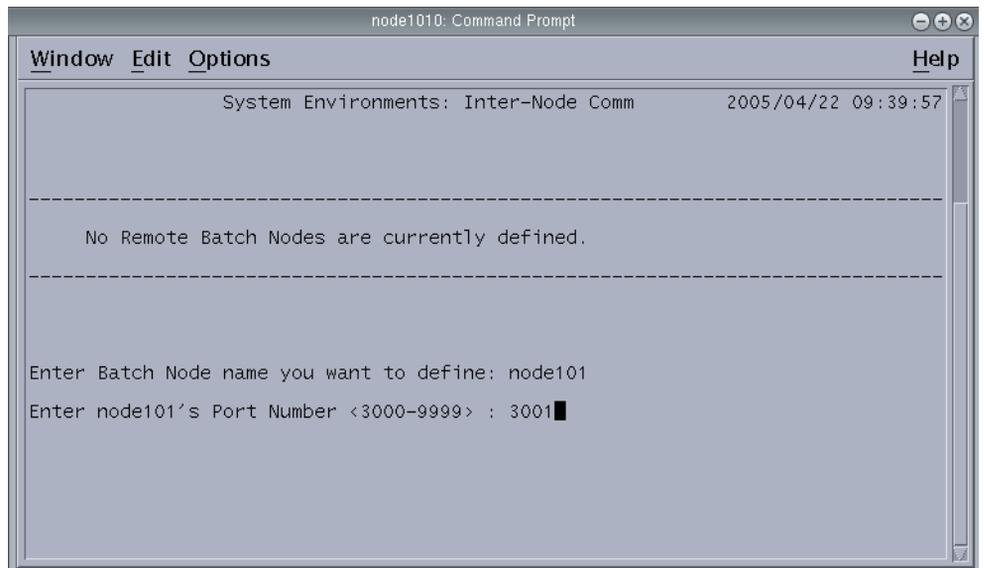


図 5-5 遠隔ノードの定義画面

3. Return キーを押します。
4. 遠隔ノード名を入力します。
5. 遠隔ノードのポート番号を入力します。
6. Return キーを押します。  
ノード名およびポート番号を入力しないで Return キーを押すと、次のメッセージが表示されます。

```
#  
# Invalid Node name. Configuration aborted.  
#  
Press Return to Continue
```

7. 確認画面で Return キーを押して「Inter-Node Communications」メニューを表示します。

# 遠隔ノードの変更

## ▼ 遠隔ノードのポート番号を変更する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」 → 「Inter-Node Communications」 → 「Change Remote Batch Node」を選択します。
3. 図 5-6 の画面が表示されたら、次のように実行します。
  - a. 遠隔ノード名を入力し、Return キーを押します。
  - b. その遠隔ノードの新しいポート番号を入力し、Return キーを押します。

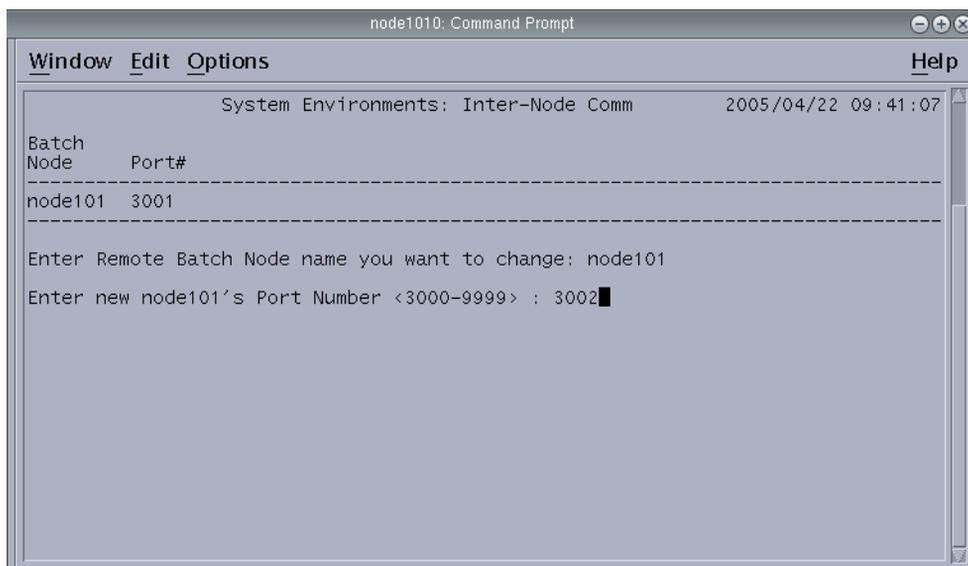


図 5-6 遠隔ノードの変更

4. 確認画面で Return キーを押して「Inter-Node Communications」メニューを表示します。

## 遠隔ノードの削除

遠隔ノードを削除すると、Sun MBM コマンドの `-N` オプションを使用してそのノードを参照することができなくなります。

### ▼ 遠隔ノードを削除する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Inter-Node Communications」 → 「Delete Remote Batch Node」を選択します。
3. 図 5-7 の画面が表示されたら、遠隔ノード名を入力し、Return キーを押します。

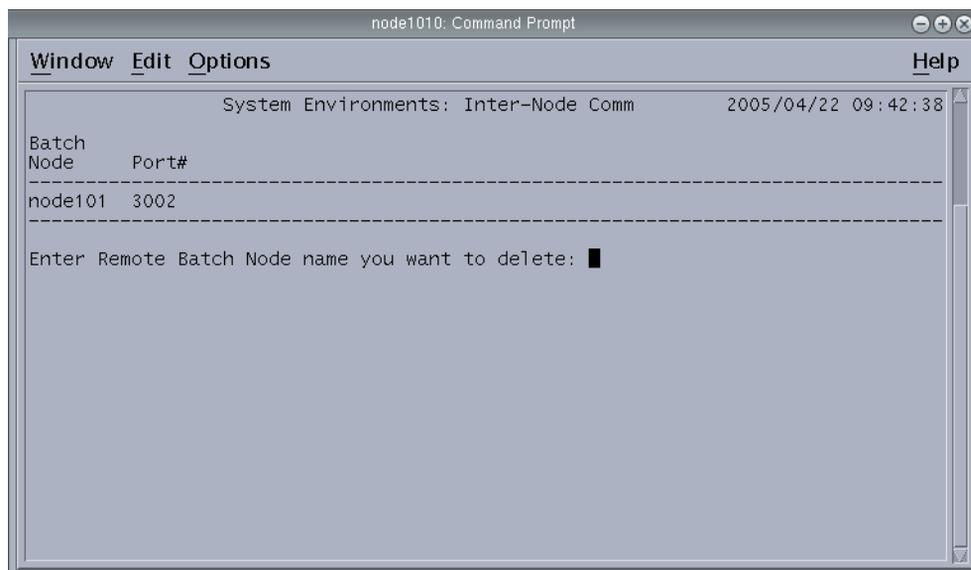


図 5-7 遠隔ノードの削除画面

4. 確認画面で Return キーを押して「Inter-Node Communications」メニューを表示します。



# ジョブアカウンティングの設定

---

ジョブアカウンティング機能は、複数のツールで構成されていて、ユーザー作成のアプリケーションプログラムとシェルスクリプトを含むアカウンティングシステムを構築するために使用できます。ノードのジョブアカウンティングを有効にすると、ジョブが終了したときに、ジョブアカウンティングファイルに1つ以上のユーザー定義レコードが書き込まれます。このファイルのデータは、アカウンティングの目的でユーザープログラムによって集計できます。

この章では、ノードを設定するためのBAMの「Job Accounting」メニューオプションの使用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。

- 54 ページの「Job Accounting メニュー」
- 55 ページの「現在のジョブアカウンティング設定の表示」
- 56 ページの「ジョブアカウンティングの有効化および無効化」
- 57 ページの「ジョブアカウンティング設定のデフォルト設定」
- 58 ページの「別のジョブアカウンティングファイルへの変更」
- 58 ページの「ジョブアカウンティングのレコード形式の変更」

ジョブアカウンティングのメニュー構造は、図 6-1 を参照してください。

ジョブアカウンティング機能の設定を変更できるのは、実行中のジョブがない場合のみです。

---

**注** – ジョブアカウンティングファイルは、自動的に消去されません。したがって、ジョブアカウンティングファイルを監視し、ファイルサイズが過大にならないよう定期的に消去する必要があります。詳細は、146 ページの「ディスク容量の確認」を参照してください。

---

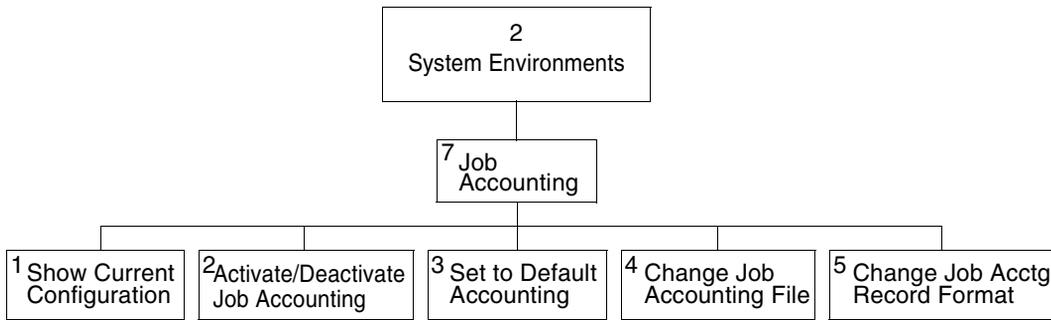


図 6-1 ジョブアカウンティングのメニュー構造

## Job Accounting メニュー

「Job Accounting」メニューには、次の 5 つのオプションがあります。

- Show the Current Configuration

ジョブアカウンティングの状態、現在のジョブアカウンティングファイル、およびジョブアカウンティングのレコード形式が表示されます。

- Activate/(Deactivate) Job Accounting

ジョブアカウンティング機能が有効でない場合にこのオプションを選択すると、アカウンティング処理が開始され、前回のジョブアカウンティング設定が復元されます。

ジョブアカウンティングが有効な場合にこのオプションを選択すると、アカウンティング処理が停止し、現在のジョブアカウンティング設定が保存されます。

- Set to Default Job Accounting Configuration

現在のジョブアカウンティング設定がデフォルト値にリセットされます。

- Change Job Accounting File

別のジョブアカウンティングファイルを定義できます。

- Change Job Accounting Record Format

1 行または複数行で形式設定指示を使用して、ジョブアカウンティングレコードの形式をカスタマイズします。

## ▼ Job Accounting メニューを表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」→「Job Accounting」を選択します。

図 6-2 のメニューが表示されます。

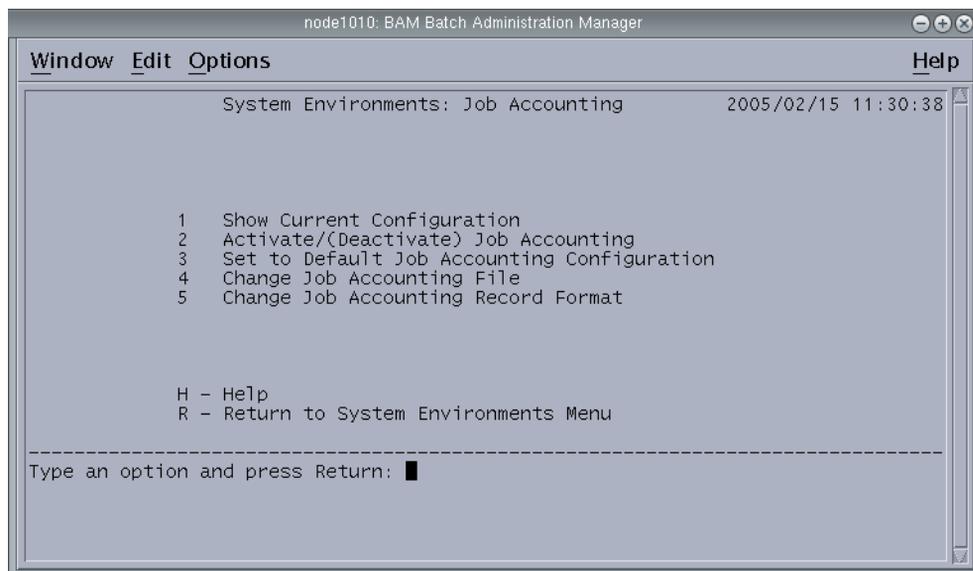


図 6-2 Job Accounting メニュー

---

## 現在のジョブアカウンティング設定の表示

### ▼ ジョブアカウンティング設定を表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」→「Job Accounting」→「Show Current Configuration」を選択します。

図 6-3 の画面に、ジョブアカウンティングの現在の状態、ファイル名、およびレコード形式が表示されます。

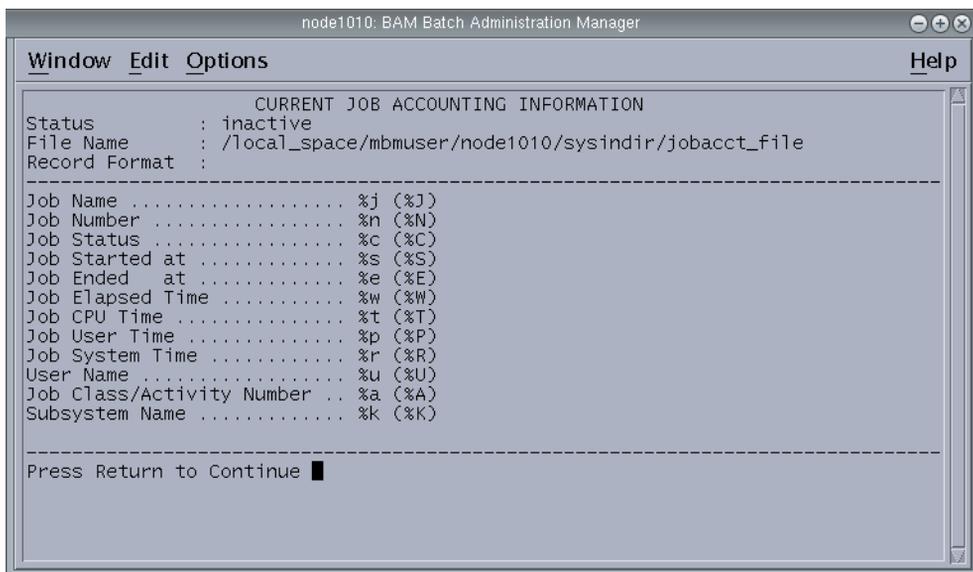


図 6-3 Current Job Accounting Information 画面

3. Return キーを押して、「Job Accounting」メニューを表示します。

---

## ジョブアカウントिंगの有効化および無効化

### ▼ ジョブアカウントिंगを有効化する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」→「Job Accounting」→「Activate/(Deactivate) Job Accounting」を選択します。
3. 次の確認メッセージが表示されたら、Return キーを押して「Job Accounting」メニューを表示します。

```
Job Accounting activated
```

## ▼ ジョブアカウンティングを無効化する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Job Accounting」 → 「Deactivate/(Activate) Job Accounting」を選択します。
3. 次の確認メッセージが表示されたら、Return キーを押して「Job Accounting」メニューを表示します。

Job Accounting deactivated

---

## ジョブアカウンティング設定のデフォルト設定

### ▼ デフォルト設定を設定する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Job Accounting」 → 「Set to Default Job Accounting Configuration」を選択します。
3. 確認メッセージが表示されたら、Return キーを押して「Job Accounting」メニューを表示します。

---

## 別のジョブアカウンティングファイルへの変更

### ▼ ジョブアカウンティングファイルを変更する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「System Environments」 → 「Job Accounting」 → 「Change Job Accounting File」を選択します。  
現在のジョブアカウンティングファイルのパス名が表示されます。
3. 「Enter full pathname of the new Job Accounting File」のプロンプトに新しいジョブアカウンティングファイルのフルパス名を入力し、Return キーを押します。
4. ジョブアカウンティングファイルのパス名が変更されたことを示す確認画面が表示されたら、Return キーを押して「Job Accounting」メニューに戻ります。

---

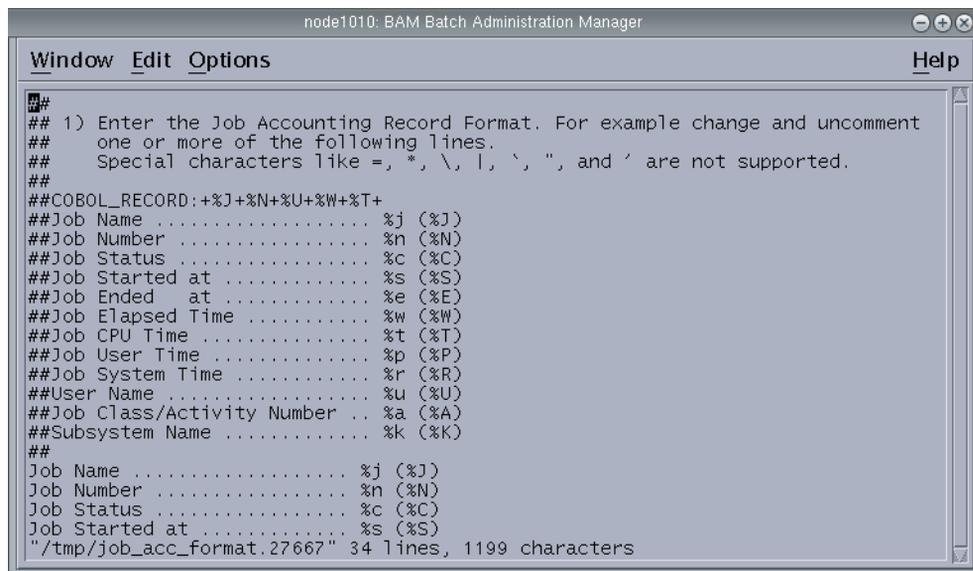
## ジョブアカウンティングのレコード形式の変更

ジョブアカウンティング機能では、ジョブアカウンティングレコードファイルにユーザーが指定した形式設定指示に従って、アカウンティング情報が表示されます。図 6-4 に、このファイルの例を示します。各行は、文字列および 1 つ以上の形式設定指示で構成されています (形式設定指示がない場合もある)。すべての文字列は、ジョブアカウンティングのレコード出力にそのままコピーされます。ジョブアカウンティングのレコード形式はカスタマイズ可能ですが、レコードサイズは 1024 文字を超えることはできません。形式設定指示の詳細は、表 6-1 を参照してください。

## ▼ ジョブアカウンティングのレコード形式を変更する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. 「System Environments」 → 「Job Accounting」 → 「Change Job Accounting Record Format」を選択します。

図 6-4 のようなファイルが表示されます。



```
node1010: BAM Batch Administration Manager
Window Edit Options Help
##
## 1) Enter the Job Accounting Record Format. For example change and uncomment
## one or more of the following lines.
## Special characters like =, *, \, |, \, ", and ' are not supported.
##
##COBOL_RECORD:+%J+%N+%U+%W+%T+
##Job Name ..... %j (%J)
##Job Number ..... %n (%N)
##Job Status ..... %c (%C)
##Job Started at ..... %s (%S)
##Job Ended at ..... %e (%E)
##Job Elapsed Time ..... %w (%W)
##Job CPU Time ..... %t (%T)
##Job User Time ..... %p (%P)
##Job System Time ..... %r (%R)
##User Name ..... %u (%U)
##Job Class/Activity Number .. %a (%A)
##Subsystem Name ..... %k (%K)
##
Job Name ..... %j (%J)
Job Number ..... %n (%N)
Job Status ..... %c (%C)
Job Started at ..... %s (%S)
"/tmp/job_acc_format.27667" 34 lines, 1199 characters
```

図 6-4 ジョブアカウンティングのレコード形式ファイル

3. テキストエディタを使用して、各行を編集します。
  - 行頭のコメントインジケータ(##)を削除し、表 6-1 の説明に従って形式設定指示を変更します。
  - また、スクリプトの末尾に指示を追加することも可能です。
4. ファイルを保存し、エディタを終了します。
5. ジョブアカウンティングのレコード形式が変更されたことを示す確認画面が表示されたら、Return キーを押して「Job Accounting」メニューに戻ります。

ジョブアカウンティングのレコード形式ファイルの形式設定指示は、次の形式をとります。

`%char`

`char` は指示コードです。

表 6-1 に、形式設定指示の一覧をその説明および出力例とともに示します。出力されるレコードはすべて改行文字で終わります。

注 – 無効な形式設定指示は変換されません。たとえば、%G と入力すると、レコード出力では、指定したフィールドに %G と表示されます。

表 6-1 ジョブアカウンティングの形式設定指示 ( 1 / 3 )

指示	説明	例
%j	可変長フィールドのジョブ名。	"JOB001"
%J	左詰め 14 文字固定長フィールドのジョブ名。	"JOB001           "
%n	可変長フィールドのジョブ番号。	"1", "32", "299"
%N	3 桁 10 進数固定長フィールドのジョブ番号。	"001", "032", "299"
%c	3 桁 10 進数固定長フィールドのジョブ終了状態コード。 コード 000 は、ジョブが正常に完了したことを示します。 000 以外のコードは、ジョブが強制的に中止されたか、取り消されたことを示します。	"001"
%C	左詰め 10 文字固定長フィールドのジョブ終了状態。	"Terminated" "Cancelled " "Aborted   "
%s	ジョブの開始日付/時刻。*	"Fri May 06 00:39:13 2005"
%S	固定長フィールドのジョブ開始日付/時刻： YYYY/MM/DD.hh:mm:ss 説明 YYYY: 4 桁 10 進数の年 : 1970 ~ 2030 MM: 2 桁 10 進数の月 : 01 ~ 12 DD: 2 桁 10 進数の日付 : 01 ~ 31 hh: 2 桁 10 進数の時刻 (24 時間表示) : 00 ~ 23 mm: 2 桁 10 進数の分 : 00 ~ 59 ss: 2 桁 10 進数の秒 : 00 ~ 59	2005/05/06.09:46:10
%e	ジョブの終了日付/時刻。*	"Fri May 06 00:41:12 2005"

表 6-1 ジョブアカウンティングの形式設定指示 ( 2 / 3 )

指示	説明	例
%E	固定長フィールドのジョブ終了日付/時刻 : YYYY/MM/DD.hh:mm:ss 説明 YYYY: 4 桁 10 進数の年 : 1970 ~ 2030 MM: 2 桁 10 進数の月 : 01 ~ 12 DD: 2 桁 10 進数の日付 : 01 ~ 31 hh: 2 桁 10 進数の時刻 (24 時間表示) : 00 ~ 23 mm: 2 桁 10 進数の分 : 00 ~ 59 ss: 2 桁 10 進数の秒 : 00 ~ 59	2005/05/06.09:46:10
%W	可変長フィールドのジョブ経過クロック時間 : seconds.hundredths-of-seconds	710.25
%W	固定長フィールドのジョブ経過クロック時間 : HH:MM:SS.hh 説明 HH: 2 桁 10 進数の時刻 (24 時間表示) : 00 ~ 23 MM: 2 桁 10 進数の分 : 00 ~ 59 SS: 2 桁 10 進数の秒 : 00 ~ 59 hh: 2 桁 10 進数の 100 分の 1 秒 : 00 ~ 59	"01:03:20.32"
%t	可変長フィールドのジョブの CPU 時間。ジョブの累積実行時間 : seconds.hundredths-of-seconds	132.12
%T	固定長フィールドのジョブの CPU 時間。ジョブの累積実行時間 : HH:MM:SS.hh 説明 HH: 2 桁 10 進数の時刻 (24 時間表示) : 00 ~ 23 MM: 2 桁 10 進数の分 : 00 ~ 59 SS: 2 桁 10 進数の秒 : 00 ~ 59 hh: 2 桁 10 進数の 100 分の 1 秒 : 00 ~ 59	00:03:20.32
%p	可変長フィールドのジョブユーザー時間。ジョブのユーザー実行時間累計 : seconds.hundredths-of-seconds	"100.06"

表 6-1 ジョブアカウンティングの形式設定指示 (3 / 3)

指示	説明	例
%P	固定長フィールドのジョブユーザー時間。ジョブのユーザー実行時間累計： <i>HH:MM:SS.hh</i> 説明 <i>HH</i> : 2 桁 10 進数の時刻 (24 時間表示): 00 ~ 23 <i>MM</i> : 2 桁 10 進数の分: 00 ~ 59 <i>SS</i> : 2 桁 10 進数の秒: 00 ~ 59 <i>hh</i> : 2 桁 10 進数の 100 分の 1 秒: 00 ~ 59	"00:02:10.22"
%r	可変長フィールドのジョブシステム時間。ジョブのシステム実行時間累計： <i>seconds.hundredths-of-seconds</i>	"32.06"
%R	固定長フィールドのジョブシステム時間。ジョブのシステム実行時間累計： <i>HH:MM:SS.hh</i> 説明 <i>HH</i> : 2 桁 10 進数の時刻 (24 時間表示): 00 ~ 23 <i>MM</i> : 2 桁 10 進数の分: 00 ~ 59 <i>SS</i> : 2 桁 10 進数の秒: 00 ~ 59 <i>hh</i> : 2 桁 10 進数の 100 分の 1 秒: 00 ~ 59	"00:01:10.10"
%u	可変長フィールドのユーザー名。	"carole"
%U	左詰め 14 文字固定長フィールドのユーザー名。	"carole"
%a	可変長フィールドのジョブクラス/アクティビティ番号。通常は、先頭の文字がジョブクラス、後続の文字がアクティビティ番号を示します。	ジョブがクラス a、アクティビティ番号 3 で実行された場合、出力は a_03。
%A	左詰め 4 文字固定長フィールドのジョブクラス/アクティビティ番号。通常は、先頭の文字がジョブクラス、後続の文字がアクティビティ番号を示します。	ジョブがクラス a、アクティビティ番号 3 で実行された場合、出力は a_03。
%k	可変長フィールドの Sun MBM サブシステム名。	"kix1"
%K	左詰め 8 文字固定長フィールドの Sun MBM サブシステム名。	"kix1"

\* この指示で出力される月と曜日を特定の言語にするには、オプション「System Environments」→「Change the date」→「Manage Language Environment」を選択して言語を指定します。  
この指示で出力される日付/時刻形式をユーザー固有の形式に変更するには、オプション「System Environments」→「Change the date」→「Change Sun MBM Date Format」を選択して指定します。

## 第7章

# サブシステムの作成

この章の内容は、次のとおりです。

- 64 ページの「サブシステムの作成準備」
- 66 ページの「使用可能なサブシステムの一覧表示」
- 67 ページの「サブシステムの作成」

「サブシステム」は、バッチジョブの実行環境です。バッチジョブを実行するには、事前にサブシステムを作成しておく必要があります。ノードごとに 8 つまでのサブシステムを作成できますが、ジョブをサブミットするには最低 1 つのサブシステムが必要です。単一のサブシステムを使用するサイトでは、そのサブシステムをデフォルトとして作成します。作成したすべてのサブシステムは、サブシステムテーブルに保存されます。

次に、サブシステムの一覧表示および作成用の Batch Administration Manager (BAM) メニュー構造を示します。

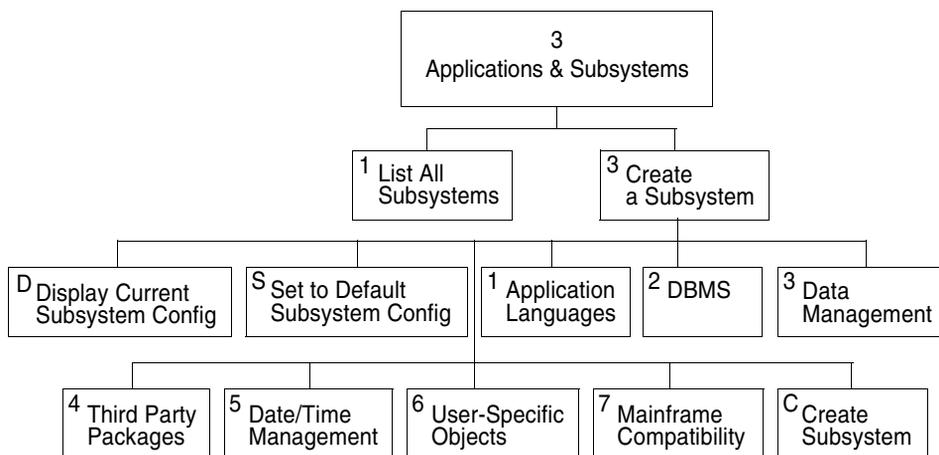


図 7-1 サブシステム作成のメニュー構造

# サブシステムの作成準備

サブシステムを作成する前に、次の点を考慮する必要があります。

- バッチジョブを作成する言語 (複数可)
- プログラムがアクセスするデータファイルのタイプ (VSAM、VSAM 以外、データベース、または以上の組み合わせ)
- ソートパッケージなど、Sun 以外のソフトウェア
- COBOL の日付/時刻管理の必要性
- COBOL ランタイムでのユーザールーチンの統合の必要性
- IDCAMS REPRO 文の動作など、メインフレーム互換性の要件

アプリケーションの実行環境のためのディレクトリ構造を作成する必要があります。BAM では、変換されたジョブやプロシージャのマクロスクリプトが含まれるディレクトリのパス名などの指定を求められます。ディレクトリが存在しない場合、エラーメッセージが表示されて、そのサブシステムは作成されません。ディレクトリ構造の例については、3 ページの「アプリケーションのディレクトリ構造の作成」を参照してください。また、JCL ジョブや JCL プロシージャのディレクトリ構造、および変換されたマクロスクリプトのディレクトリ構造については、『Sun Mainframe Batch Manager ソフトウェア 移行ガイド』を参照してください。

次の表に、作成するサブシステムのタイプを基に BAM で要求される情報を示します。複合サブシステムを作成する場合、たとえば、COBOL プログラムを使用して VSAM データセットにアクセスするサブシステムを作成する場合は、COBOL で必要な情報と Sun MTP で必要な情報を指定する必要があります。

表 7-1 サブシステムのタイプで必要な情報

サブシステムのタイプ	情報
COBOL	COBOL インストールディレクトリ
Open PL/I	PL/I インストールディレクトリ
C/C++	C/C++ インストールディレクトリ
Java™	Java インストールディレクトリ
Sun MTP	Sun MTP インストールディレクトリ (\$UNIKIX) 領域のシステムディレクトリ (\$KIXSYS)
Oracle	Oracle インストールディレクトリ (\$ORACLE_HOME) \$ORACLE_SID の値 データベースログイン ID

表 7-1 サブシステムのタイプで必要な情報 (続き)

サブシステムのタイプ	情報
IBM DB2 UDB	DB2 インスタンスのホームディレクトリ (\$INSTHOME) デフォルトデータベース (\$DB2DBDFT) インスタンス名 (\$DB2INSTANCE)
Sybase	Sybase インストールディレクトリ \$DSQUERY の値 (ユーザーの環境で設定されていない場合)
EXCI	Sun MTP Client インストールディレクトリ
WebSphere MQ	WebSphere MQ インストールディレクトリ (\$MQSERIES) MQ サーバーの情報 (\$MQSERVER)
すべて	変換されたジョブのマクロスクリプトを含むディレクトリ (ish) 変換されたプロシーチャーのマクロスクリプトを含むディレクトリ (ishp) バッチアプリケーションプログラムを含むディレクトリ システム出力ファイル (sysout ファイル) 用のディレクトリ 履歴ファイル (ジョブ出力一覧) 用のディレクトリ

## 使用可能なサブシステムの一覧表示

「Applications & Subsystems」メニューからすべてのサブシステムを一覧表示した場合に示されるのが、使用可能なすべてのサブシステムです。「System Environments」→「Report System Status」→「List Active Subsystems」を選択して表示されるリストは、これとは異なります。その場合に表示されるのは、各サブシステムに関する情報です。詳細は、図 3-4 を参照してください。

### ▼ ノードで使用可能なすべてのサブシステムを一覧表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. BAM メインメニューでオプション「3」を選択し、「Applications & Subsystems」メニューを表示します。

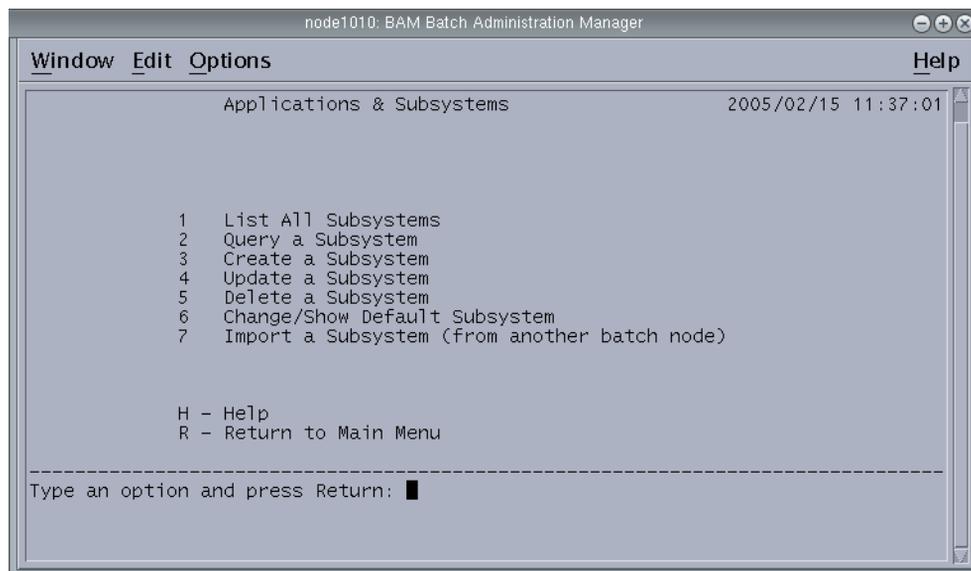


図 7-2 Applications & Subsystems メニュー

3. オプション「1 List All Subsystems」を選択します。

図 7-3 に示す画面が表示されます。

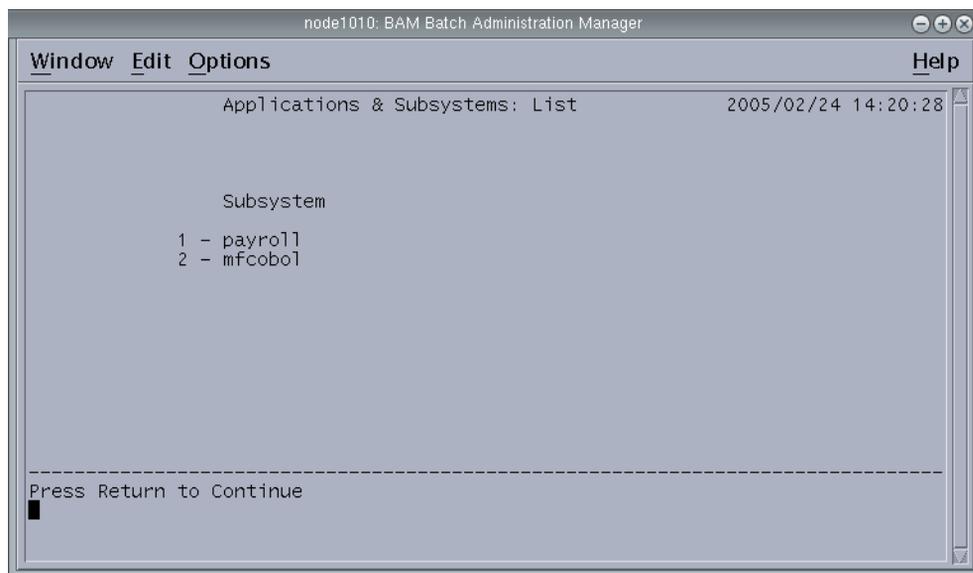


図 7-3 サブシステムの一覧表示画面

4. Return キーを押して、「Applications & Subsystems」メニューを表示します。

---

## サブシステムの作成

この節では、サブシステムを作成するための BAM ユーティリティの使用法について説明します。それぞれの小節で、データ管理、メインフレーム互換性などのオプションについて説明します。

### ▼ サブシステムを作成する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
2. 「Applications & Subsystems」→「Create a Subsystem」を選択します。  
既存のサブシステムを一覧表示する画面が表示されます。

3. 作成するサブシステムの一意の名前を入力し、Return キーを押します。

図 7-4 の「Create」メニューが表示されます。

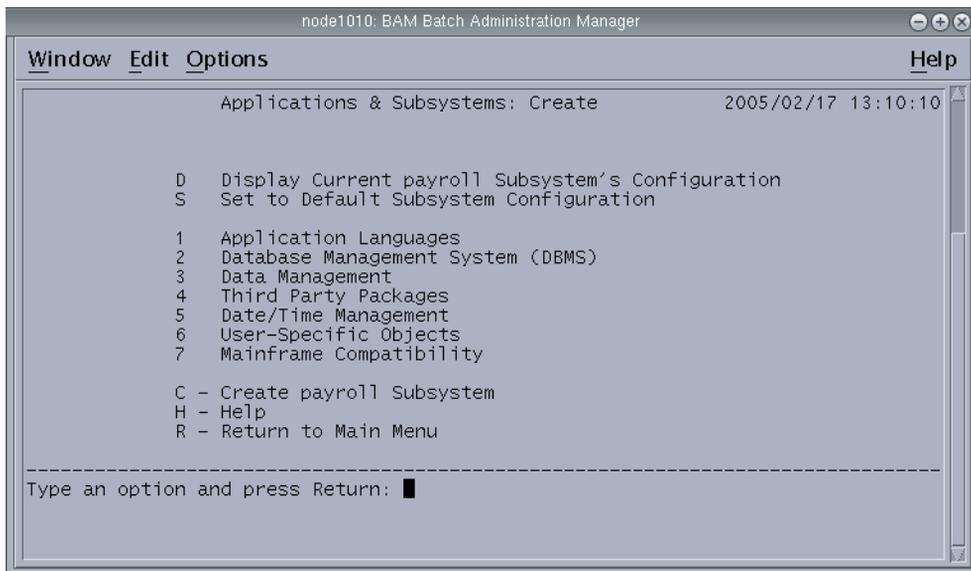


図 7-4 Create メニュー

設定オプションについては、次のそれぞれの節を参照してください。

- 69 ページの「アプリケーション言語を選択する」
- 70 ページの「1 つまたは複数のデータベースを選択する」
- 71 ページの「データファイルのタイプを選択する」
- 72 ページの「Sun 以外のパッケージを選択する」
- 73 ページの「日付/時刻管理を選択する」
- 74 ページの「COBOL 実行時システムにユーザー固有オブジェクトを取り込む」
- 75 ページの「メインフレーム互換性オプションの指定」

## ▼ アプリケーション言語を選択する

サブシステムの作成時に、複数のプログラミング言語をサポートするように設定できます。

1. 「Create」メニューでオプション「1 Application Languages」を選択します。

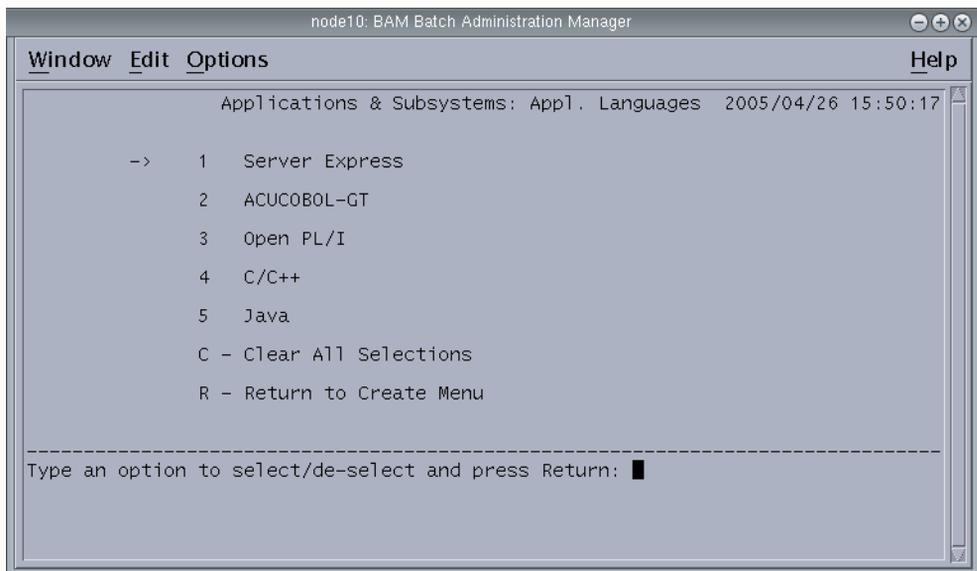


図 7-5 Application Languages 選択画面

2. サブシステムでサポートする言語の次の番号 (複数可) を入力します。
  - 1 は、Server Express (デフォルト言語) を選択/選択解除します。
  - 2 は、ACUCOBOL-GT<sup>®</sup> を選択/選択解除します。
  - 3 は、Open PL/I を選択/選択解除します。
  - 4 は、C/C++ を選択/選択解除します。
  - 5 は、Java を選択/選択解除します。
3. R キーを押して、「Create」メニューに戻ります。

## ▼ 1 つまたは複数のデータベースを選択する

サブシステムでは、1 つまたは複数のデータベースをサポートできます。VSAM ファイルへのアクセスもサポートする必要がある場合は、サブシステムが Sun MTP に接続するように設定します (付録 F を参照)。

1. 「Create」メニューでオプション「2 Database Management Systems (DBMS)」を選択し、データベース選択画面を表示します。

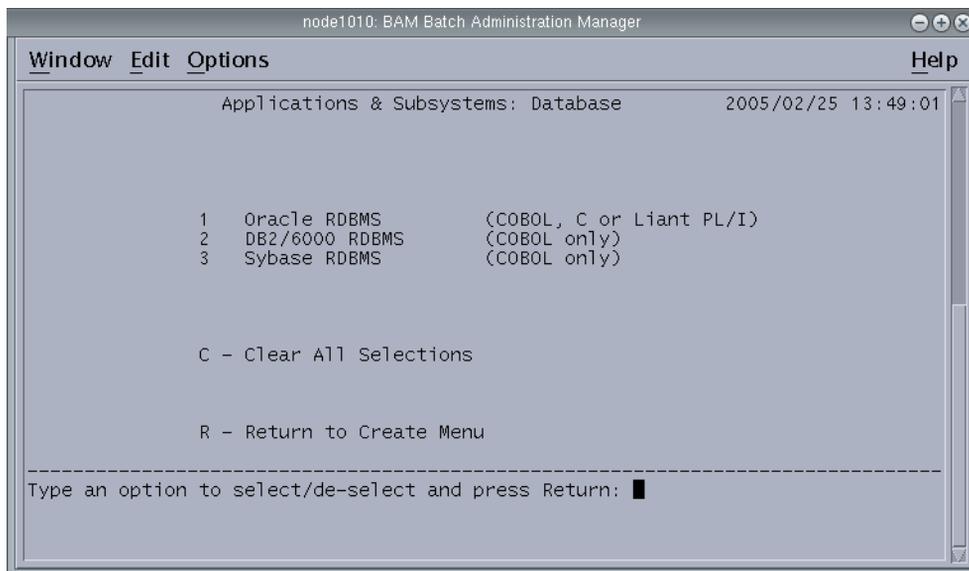


図 7-6 DBMS 選択画面

2. 使用するデータベースに対応する番号を選択します。  
複数のデータベースを選択できます。  
データベースの選択を解除するには c キーを押し、データベースを選択し直します。
3. Return キーを押して「Create」メニューを表示します。

## ▼ データファイルのタイプを選択する

1. 「Create」メニューでオプション「3」を選択し、「Data Management」画面を表示します。

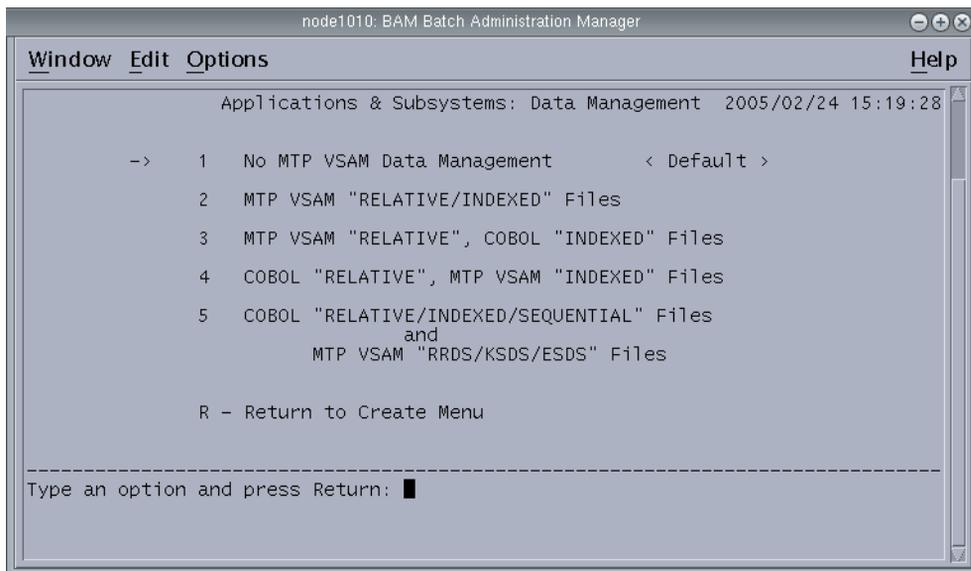


図 7-7 Data Management 画面

2. バッチジョブがアクセスするデータファイルのタイプを示すオプションを選択します。
  - 1: 標準の順編成ファイル、相対編成ファイル、および索引編成ファイルを使用します。VSAM ファイルは使用しません。つまり、サブシステムは Sun MTP 領域と対話しません。
  - 2: Sun MTP 相対編成ファイルおよび索引編成ファイルを使用します (COBOL または Open PL/I プログラムの場合)。または、C/C++ プログラムからの Sun MTP C-ISAM インタフェースを使用します。
  - 3: Sun MTP 相対編成ファイルおよび COBOL 索引編成ファイルを使用します。
  - 4: COBOL 相対編成ファイルおよび Sun MTP 索引編成ファイルを使用します。
  - 5: COBOL プログラムからすべてのアクセス方式を使用します。詳細は、『Sun Mainframe Batch Manager ソフトウェア 移行ガイド』を参照してください。
3. Return キーを押して「Create」メニューを表示します。

## ▼ Sun 以外のパッケージを選択する

1. 「Create」メニューでオプション「4 Third Party Packages」を選択します。

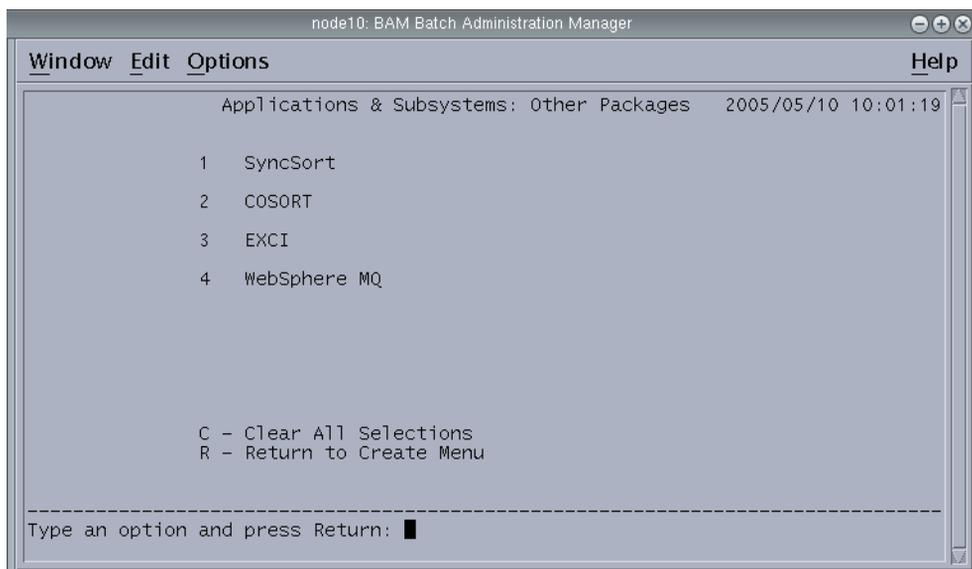


図 7-8 Other Packages 画面

2. 使用するパッケージと一致する番号を選択します。  
パッケージの選択を解除する場合は、D キーを押します。
3. Return キーを押して「Create」メニューを表示します。

## ▼ 日付/時刻管理を選択する

サブシステムをテスト環境として使用し、COBOL 日付/時刻管理が必要な場合は、次の手順を実行してください。

1. 「Create」メニューでオプション「5 Date/Time Management」を選択します。

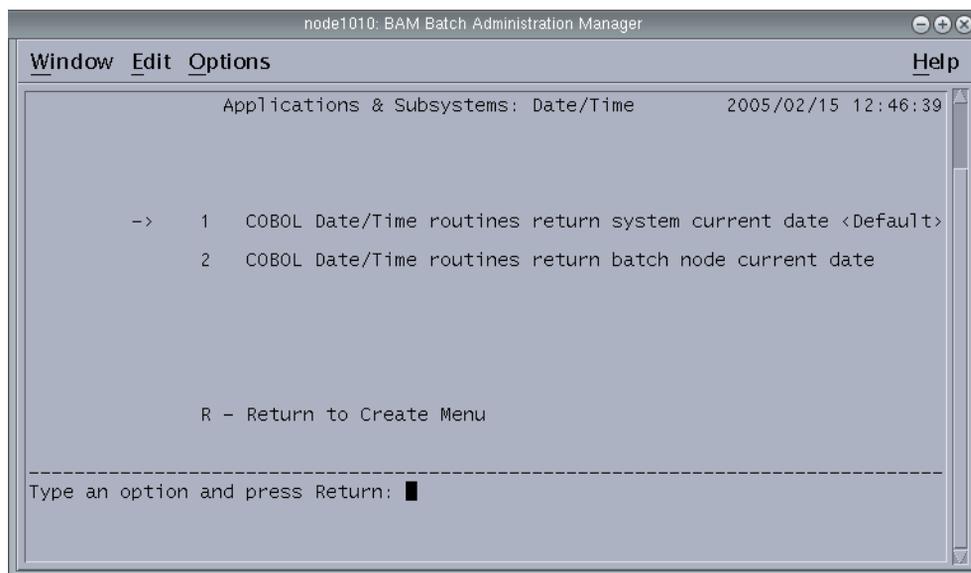


図 7-9 COBOL Date/Time 画面

2. オプションを選択して、Return キーを押します。

COBOL プログラムを使用するサブシステムを設定し、システムで標準 COBOL 日付/時刻ルーチンによって Sun MBM の現在日付が返されるようにする場合は、オプション「2」を選択します。

---

**注** - ご使用のプラットフォームおよび COBOL バージョンがこの機能をサポートしているかを確認するには、『Sun Mainframe Batch Manager ソフトウェア ご使用にあたって』を参照してください。

---

ノードがテスト環境として設定されている場合は、標準 Open PL/I 日付/時刻ルーチンにより常に Sun MBM の現在日付が返されます。



## メインフレーム互換性オプションの指定

ジョブが MVS 環境から移行されたか VSE 環境から移行されたかに応じて、それぞれの互換性オプションが Sun MBM によってサポートされます。

### ▼ Mainframe Compatibility メニューを表示する

1. 「Create」メニューでオプション「7」を選択します。
2. メニューが表示されたら、「MVS」または「VSE」の環境のオプションを選択します。

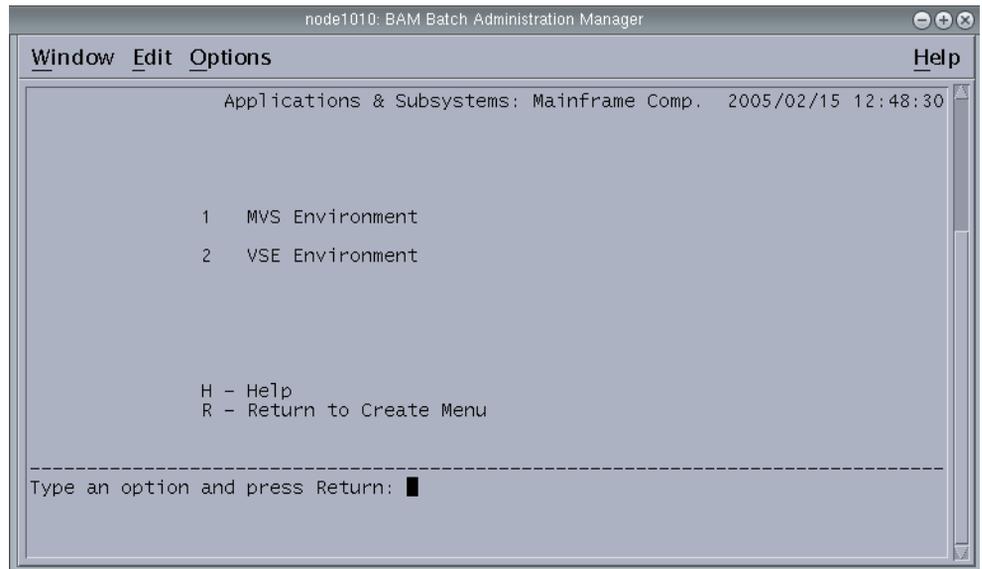


図 7-11 Mainframe Compatibility メニュー

メインフレーム環境から移行する場合の互換性オプションの設定方法については、次の節で説明します。

- 76 ページの「MVS REPRO オプションを設定する」
- 78 ページの「MVS OUTPUT オプションを設定する」
- 79 ページの「MVS MAXCC 値を設定する」
- 80 ページの「MVS RLS オプションを設定する」
- 81 ページの「VSE JCL REPRO オプションを設定する」

## ▼ MVS REPRO オプションを設定する

1. 「Mainframe Compatibility」メニューで、オプション「1 MVS Environment」を選択します。

次の画面が表示されます。

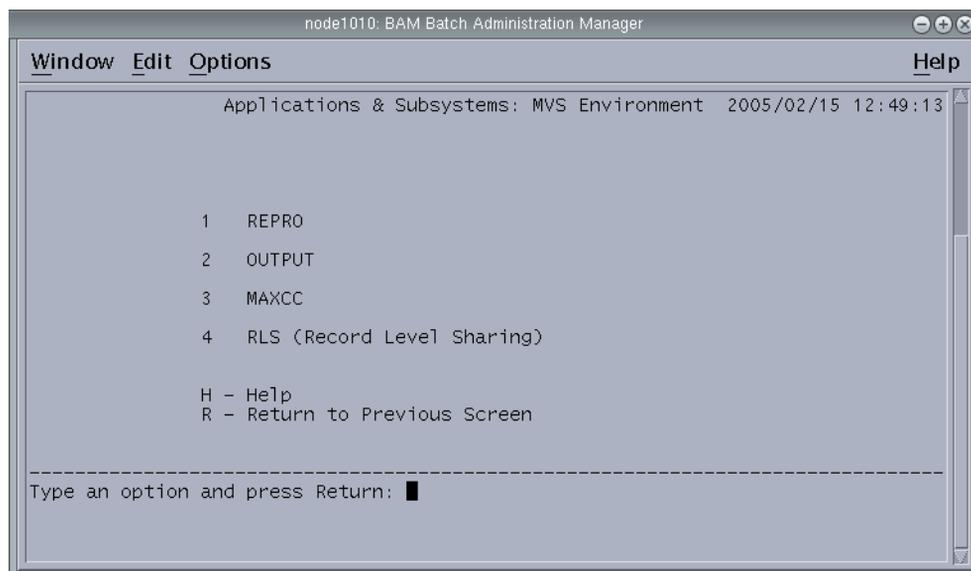


図 7-12 MVS Environment メニュー

2. オプション「1」を選択して、「REPRO」画面を表示します。

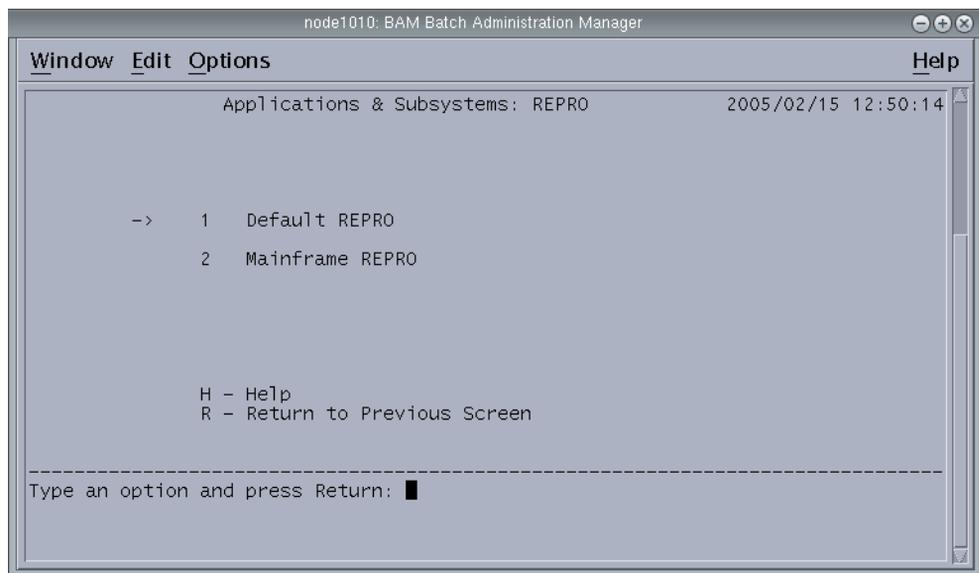


図 7-13 MVS REPRO 画面

3. サブシステムで使用する「REPRO」オプションを次のように選択します。
  - オプション「1」は、現在のリリースで実行されている既存のアプリケーションとの下位互換性を確保します。

下位互換性のため、IDCAMS REPRO には、以前の REUSE の動作がデフォルトアクションとして保持されています。パラメータ NOREUSE、REPLACE、および NOREPLACE は、IDCAMS SYSIN ファイルに指定されても無視されます。
  - オプション「2」では、メインフレーム REPRO デフォルトの NOREUSE および NOREPLACE が使用されます。
4. R と入力して **Return** キーを押し、「**MVS Environment**」メニューを表示します。

## ▼ MVS OUTPUT オプションを設定する

OUTPUT 文を使用している MVS JCL アプリケーションを実行するには、MVS JCL トランスレータ mvstrans によるジョブとプロシーチャーの変換時に適用される JCL OUTPUT 変換規則を指定する必要があります。

1. 「Mainframe Compatibility」メニューで、オプション「1 MVS Environment」を選択します。
2. 「MVS Environment」メニューでオプション「2 OUTPUT」を選択します。

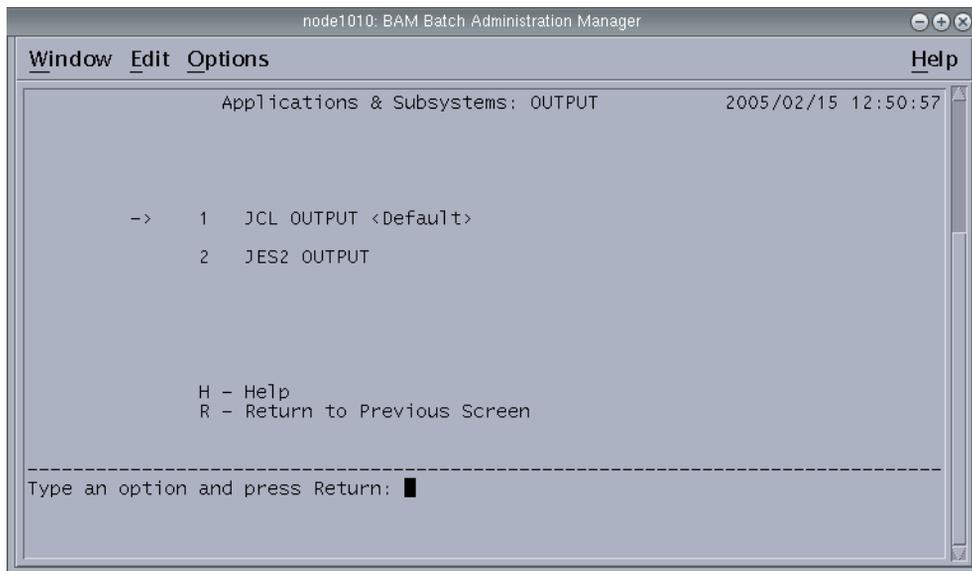


図 7-14 MVS OUTPUT 画面

3. サブシステムで使用する「OUTPUT」オプションを次のように選択します。
  - オプション「1」は、MVS JCL 文を含むジョブおよびプロシーチャーを mvstrans で変換するときに MVS JCL OUTPUT 変換規則を適用します。
  - オプション「2」は、JES2 文を含むジョブおよびプロシーチャーを mvstrans で変換するときに MVS JES2 OUTPUT 変換規則を適用します。
4. R と入力して Return キーを押し、「MVS Environment」メニューを表示します。

## ▼ MVS MAXCC 値を設定する

IDCAMS ステップから生成される MAXCC 値をメインフレームと同じように Sun MBM で処理させるには、次の手順を実行します。詳細については、『Sun Mainframe Batch Manager ソフトウェア 移行ガイド』を参照してください。

1. 「Mainframe Compatibility」メニューで、オプション「1 MVS Environment」を選択します。
2. 「MVS Environment」メニューでオプション「3」を選択し、Return キーを押します。

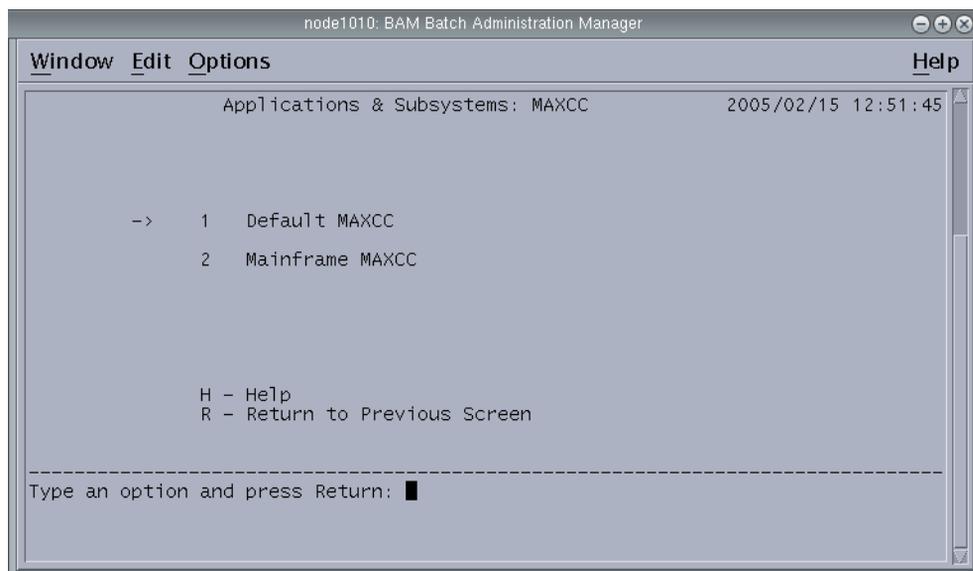


図 7-15 MVS MAXCC 画面

3. サブシステムで使用する「MAXCC」オプションを次のように選択します。
4. R と入力して Return キーを押し、「MVS Environment」メニューを表示します。

## ▼ MVS RLS オプションを設定する

メインフレームスタイルのレコードレベル共有 (RLS) をサポートするようにサブシステムを設定したり、あるいは Sun MBM の以前のバージョンとの下位互換性を確保したりできます。MVS JCL DD 文の RLS パラメータの使用法については、『Sun Mainframe Batch Manager ソフトウェア 移行ガイド』を参照してください。

1. 「Mainframe Compatibility」メニューで、オプション「1 MVS Environment」を選択します。
2. 「MVS Environment」メニューでオプション「4」を選択し、Return キーを押します。

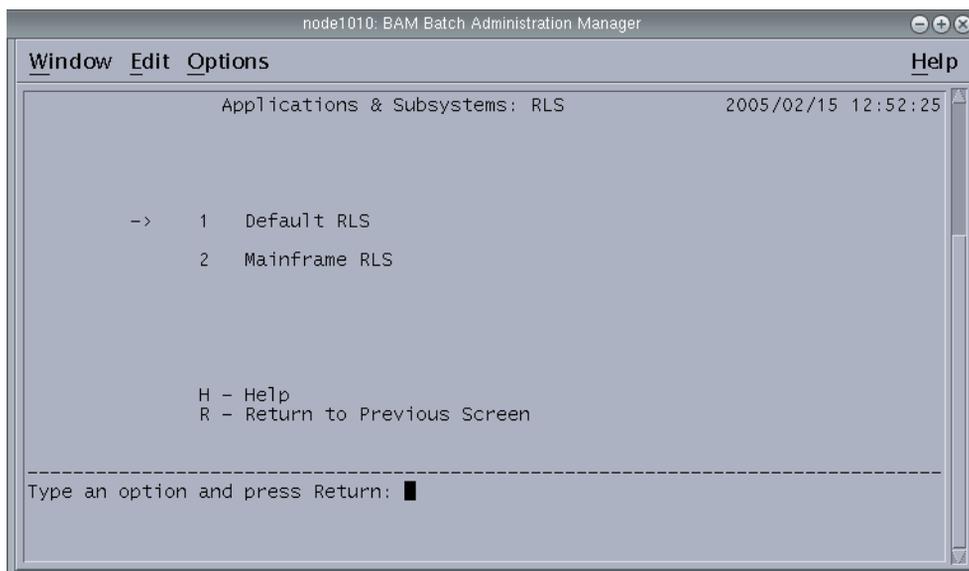


図 7-16 MVS RLS 画面

3. 「RLS」画面で、サブシステムでサポートする動作を次のように選択します。
  - オプション「1」は、下位互換性を確保します。このオプションがデフォルトです。
  - オプション「2」は、RLS パラメータをメインフレームと同じようにサブシステムで処理します。
4. R と入力して Return キーを押し、「MVS Environment」メニューを表示します。

## ▼ VSE JCL REPRO オプションを設定する

1. 「Mainframe Compatibility」メニュー (図 7-11) でオプション「2 VSE Environment」を選択します。
2. 「VSE Environment」画面でオプション「1 REPRO」を選択します。

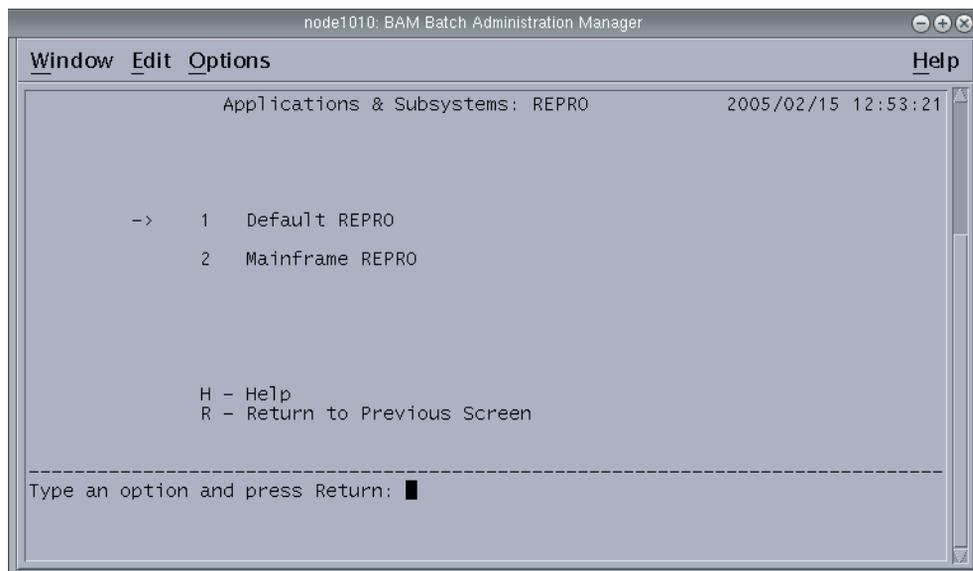


図 7-17 VSE REPRO 画面

3. サブシステムで使用する「REPRO」オプションを次のように選択します。
  - オプション「1」は、現在のリリースで実行されている既存のアプリケーションとの下位互換性を確保します。

下位互換性のため、IDCAMS REPRO には、以前の REUSE の動作がデフォルトアクションとして保持されています。パラメータ NOREUSE、REPLACE、および NOREPLACE は、IDCAMS SYSIN ファイルに指定されても無視されます。
  - オプション「2」では、メインフレームの REPRO デフォルトの NOREUSE および NOREPLACE が使用されます。
4. R と入力して Return キーを押す操作を 3 回繰り返して、「Create」メニューを表示します。

## 設定内容の表示

サブシステムを作成する前に、行なった変更を確認するために設定内容を表示します。

### ▼ 設定内容を表示する

1. 「Create」メニューでオプション「D Display Current *subsystem-name* Subsystem's Configuration」を選択します。

次の画面が表示されます。

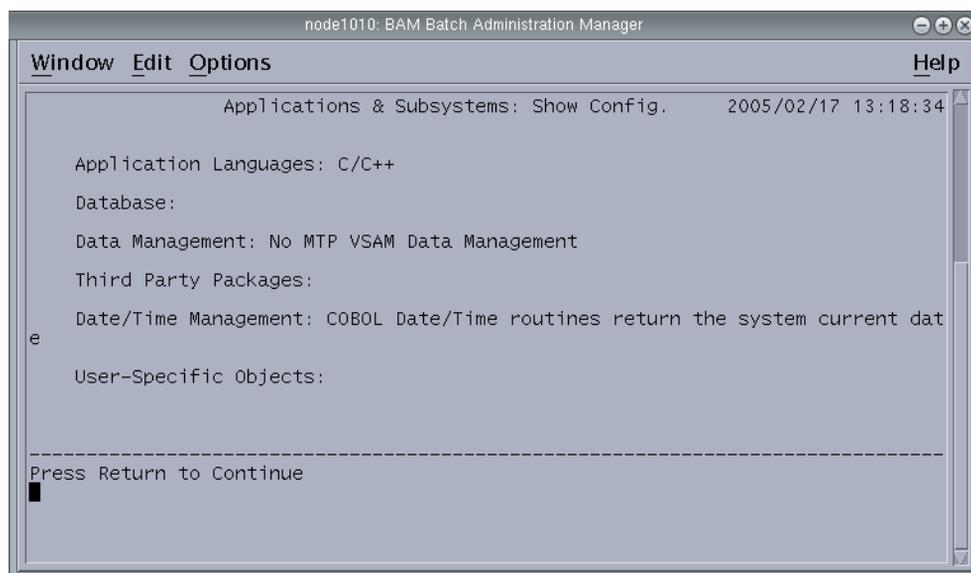


図 7-18 サブシステムの現在の設定内容の表示

2. 選択内容を確認します。

- 設定内容に問題がなければ、Return キーを押して「Create」メニューを表示して 83 ページの「サブシステムの作成」に進みます。
- 設定内容が適切でない場合は、Return キーを押して「Create」メニューを表示してオプション「S」を選択すると、サブシステムの設定がデフォルト値にリセットされ、設定作業をやり直すことができます。

# サブシステムの作成

## ▼ サブシステムを作成する

1. 「Create」メニューで、サブシステムの構成ファイルを作成するためのオプション「C Create *subsystem-name* Subsystem」を選択します。  
選択内容に基づいて、ディレクトリパス名などの情報の指定を求める BAM の画面が表示されます。  
このパス名は、サブシステム設定ファイルの環境変数を作成するために使用されます。存在しないディレクトリを入力すると、エラーメッセージが表示されます。
2. C/C++ 以外のアプリケーション言語を選択した場合は、言語のインストールディレクトリをプロンプトに入力します。
3. Sun MTP Data Management を選択した場合は、2 つのディレクトリが要求されます。
  - a. 図 7-19 の画面が表示されたら、Sun MTP インストールディレクトリのパスを入力します。



図 7-19 Sun MTP インストールディレクトリの指定

---

注 – VSAM Data Management を選択した場合は、サブシステムに Sun MTP 領域が必要です。詳細は、付録 F を参照してください。

---

- b. プロンプトで要求されたら、KIXSYS 環境変数のディレクトリも指定する必要があります。

サブシステムの作成が完了したら、指定した領域 (\$KIXSYS) が Sun MBM に接続できるように設定するために、その領域の VCT を更新する必要があります。詳細は、199 ページの「ノードに接続する領域の設定」を参照してください。

4. データベースを選択すると、必要な値の指定を求める BAM の画面が表示されます。適切な値を入力します。
5. システム出力ディレクトリの親ディレクトリの名前を入力します。これは、SYSOUT DD 文によってジョブのシステム出力データセットが作成されるディレクトリです。たとえば、SYSOUT データセットにディレクトリ /pkgs/mbm/node1/sysoutdir を使用する場合、この画面ではディレクトリ /pkgs/mbm/node1 を指定します。
6. 変換された JCL ジョブが書き込まれるディレクトリの名前を指定します。たとえば、\$HOME/batch/ish を指定します。
7. 変換された JCL プロシージャが書き込まれるディレクトリの名前を指定します。たとえば、\$HOME/batch/ishp を指定します。これは、\$PROCLIB ディレクトリとも呼ばれます。
8. システム実行時に実行されるアプリケーションプログラムがあるデフォルトディレクトリの名前を指定します。たとえば、\$HOME/batch/cbl を指定します。
9. ジョブ履歴ファイルの位置となるジョブ出カ一覧ディレクトリを指定します。次のオプションをいずれか 1 つ入力し、Return キーを押します。
  - D: デフォルトのジョブ出カ一覧ディレクトリを使用します。詳細は、27 ページの「ジョブの履歴ファイルのリダイレクト」を参照してください。
  - U: 新しいディレクトリを入力します。プロンプトで要求されたら、新しい値を入力します。
  - C: 現在のディレクトリ設定を使用します。

10. 入力の完了後、「Environment Setting Completed」画面で **y** と入力します。

BAM によって構成ファイルが作成されます。作成が完了すると、システム構成が更新されてサブシステムが作成されたことを示すメッセージが表示されます。

作成中にエラーが発生した場合は BAM によってエラーメッセージが表示され、作成プロセスは強制的に中止されて「Applications & Subsystems」の「Create」メニューに戻ります。必要に応じて、画面に表示されている場所にある `bam.log` ファイルを参照し、このサブシステムの作成に関するすべてのメッセージを確認できます。

手順 1 に戻ってエラーを修正します。BAM には前回の入力内容が保存されているので、すべての項目を入力し直す必要はありません。

11. Return キーを押して、「Applications & Subsystems」メニューを表示します。

12. オプション「1 List All Subsystems」を選択し、サブシステムが作成されていることを確認します。

サブシステムの構成の変更については、92 ページの「サブシステムの更新」を参照してください。



# サブシステムのカスタマイズ

---

バッチアプリケーションのライフサイクルの間、アプリケーションを実行するサブシステムに対して構成変更が必要になることがあります。必要な変更を確定したら、使用に最も適した方法を選択します。多くの場合、この章で説明される BAM オプションを1つ以上使用できます。次のトピックがあります。

- 89 ページの「サブシステムの現在の構成の表示」
- 92 ページの「サブシステムの更新」
- 98 ページの「デフォルトのサブシステムの定義」
- 100 ページの「サブシステムの削除」
- 102 ページの「削除したサブシステムの再作成」
- 104 ページの「サブシステムのインポート」

現在のサブシステムを分析するには、BAM の照会機能を使用してサブシステムの特  
性、構成、ジョブ履歴ディレクトリ、ジョブ、設定ファイル (\$SETUP)、  
File\_Map、およびユーザー設定ファイル (\$USER\_SETUP) を表示できます。

何を変更するかを確定したら、その変更方法を決定します。ジョブ履歴ディレクトリ  
の変更、サブシステムが関連付けられている Sun MTP 領域の変更、File\_Map の編  
集、デフォルトサブシステムの変更、またはサブシステムの \$USER\_SETUP の変更が  
必要なときは、適切な BAM オプションを選択して変更できます。ただし、サブシ  
ステムに対する変更が Sun MBM や COBOL などのその他のパッケージの新しいリ  
リースへのアップグレードに起因する場合は、重要な決定が必要になります。サブシ  
ステムの削除、再作成、またはインポートを行います。このような決定をするときは、  
次の点を考慮してください。

- サブシステムを削除すると、\$SETUP ファイルが削除されます。そのサブシステム  
を再作成しても、カスタム設定は失われています。Release 10 以降で作成したサ  
ブシステムを削除した場合、\$SETUP にカスタム設定はなくなります。すべてのカ  
スタム設定は、\$USER\_SETUP にあります。
- サブシステムを削除したら、同じ名前で作成できます。アプリケーションでサ  
ブシステム名をハードコーディングした場合は、この点を考慮に入れる必要があ  
ります。

- Release 9.2、10.0.0、または 10.0.1 からサブシステムをインポートすると (Release 9.2 以前からのインポートはサポートされていない)、\$SETUP ファイルが再作成され、そのカスタム設定は失われます。Release 10.0.0 以降で作成したサブシステムをインポートすると、\$USER\_SETUP のカスタム設定が保持されます。その場合、設定内容が有効かどうかを確認する必要があります。
- Release 10.1.0 で作成したサブシステムをインポートする場合は、別の名前を付ける必要があります。アプリケーションでサブシステム名をハードコーディングした場合、これに従わないと、ジョブの実行に影響する可能性があります。
- サブシステムのインポートには、新しいサブシステム構成が期待どおりに機能しない場合に以前のサブシステムで実行できるという、サブシステムの削除/再作成にはない利点があります。

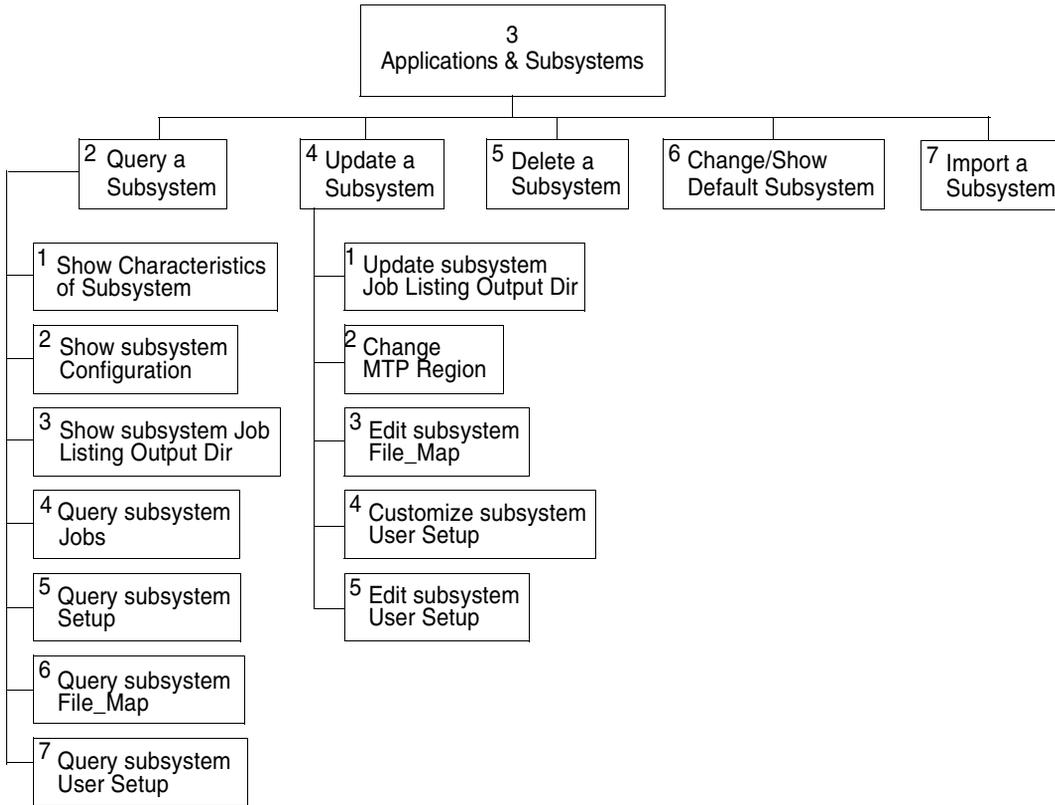


図 8-1 サブシステムの照会、更新、削除、デフォルトの変更/表示、インポートのメニュー構造

# サブシステムの現在の構成の表示

## ▼ サブシステムの構成を表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. 「Applications & Subsystems」 → 「Query a Subsystem」を選択します。
3. 「Query」画面で、表示するサブシステムの番号を入力します。
4. 「Query」メニューが表示されたら、オプション「1 Show Characteristics of *subsystem-name* Subsystem」を選択します。

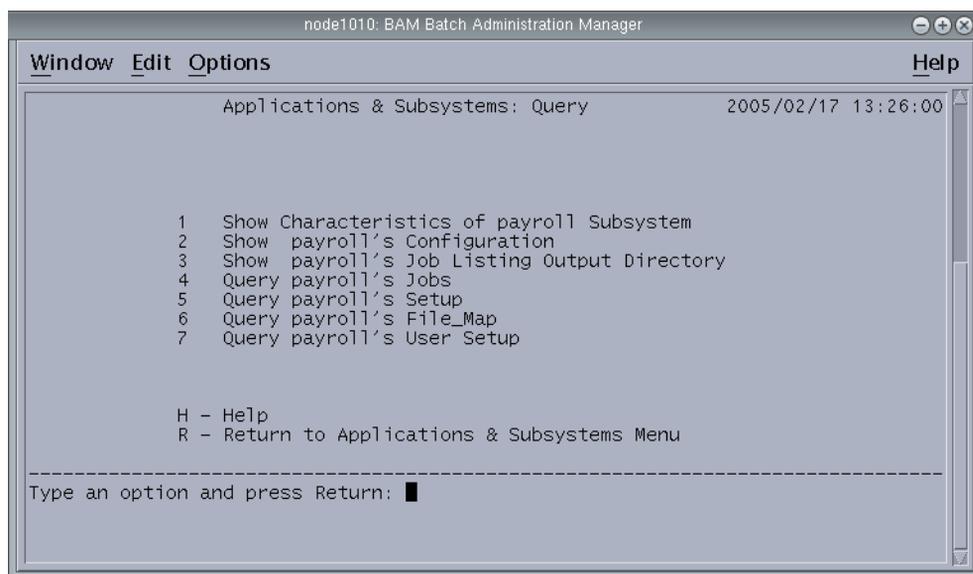


図 8-2 Query メニュー

5. 図 8-3 のサブシステムに関する情報を確認し、Return キーを押して「Query」メニューを表示します。

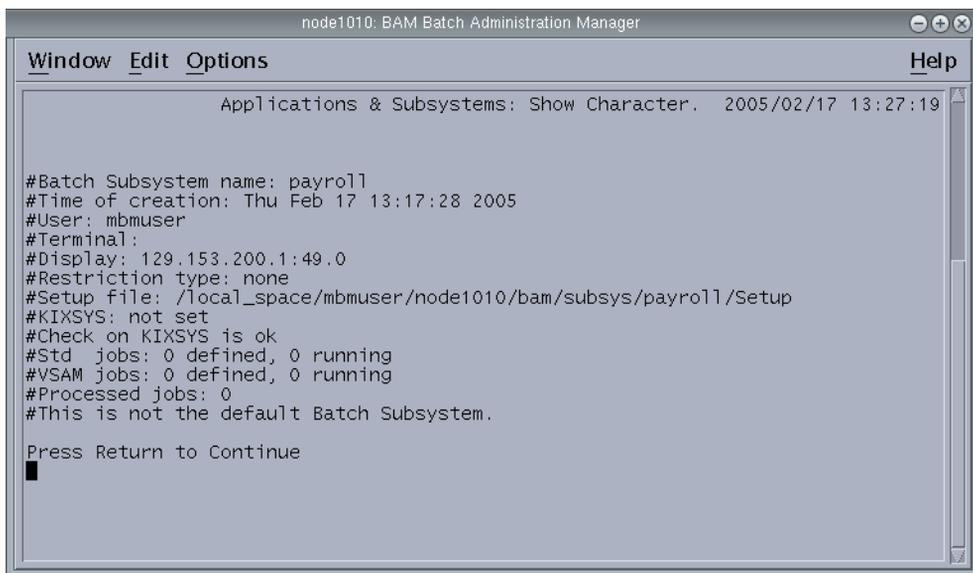


図 8-3 サブシステムの特性画面

6. オプション「2 Show *subsystem-name's* Configuration」を選択し、サブシステムの構成を表示します (図 8-4)。

この画面には、サブシステムを作成したときの選択内容が表示されます。

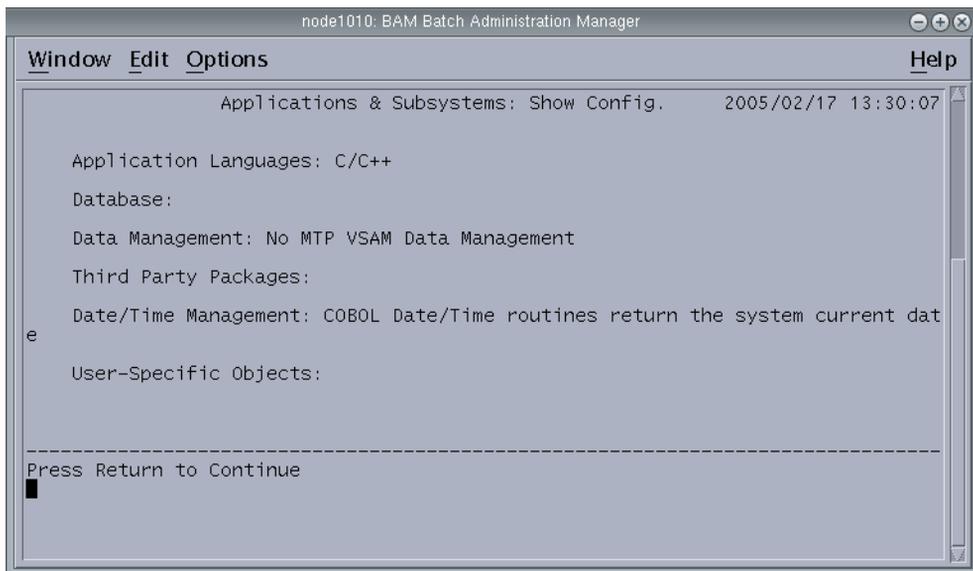


図 8-4 サブシステムの構成画面

7. **Return** キーを押し、「**Query**」メニューを表示します。
8. オプション「3 Show *subsystem-name*'s Job Listing Output Directory」を選択し、ディレクトリパスを表示します。  
画面には、ノードのデフォルトディレクトリおよびサブシステムのジョブ履歴ディレクトリ (定義済みの場合) が表示されます。サブシステム固有のディレクトリが定義されている場合は、そのディレクトリにジョブ履歴ファイルが書き込まれます。定義されていない場合は、デフォルトディレクトリが使用されます。このディレクトリを変更する手順は、93 ページの「ジョブ履歴ディレクトリを変更する」を参照してください。
9. **Return** キーを押し、「**Query**」メニューを表示します。
10. オプション「4 Query *subsystem-name*'s Jobs」を選択し、サブシステムに現在サブミットされているジョブを表示します。
11. **Return** キーを押し、「**Query**」メニューを表示します。
12. オプション「5 Query *subsystem-name*'s Setup」を選択し、サブシステムに関連付けられている設定ファイル (\$SETUP) を表示します。

```

node1010: BAM Batch Administration Manager
-----
Window Edit Options Help
-----
#
#The following are payroll Subsystem's specific settings
#
# Note:
#
# Do not edit this file.
# Subsystem environment modifications must be made in the User Setup file
#
setenv FILEMAP /local_space/mbmuser/node1010/bam/subsys/payroll/File_Map
setenv SETUP /local_space/mbmuser/node1010/bam/subsys/payroll/Setup
setenv USER_SETUP /local_space/mbmuser/node1010/bam/subsys/payroll/Setup_usr
setenv SYSOUTS /local_space/mbmuser/node1010
setenv SYSOUTDIR $SYSOUTS/sysoutdir
setenv FORMS $SYSOUTS/formdir
setenv PROCLIB /local_space/mbmuser/node1010/samples/test1/ishp
setenv JOBLIB /local_space/mbmuser/node1010/samples/test1/pgm
setenv JCLLIB /local_space/mbmuser/node1010/samples/test1/ish
setenv LD_LIBRARY_PATH /local_space/mbmuser/node1010/pack/lib/lang:
setenv LIBPATH /local_space/mbmuser/node1010/pack/lib/lang:
setenv SHLIB_PATH /local_space/mbmuser/node1010/pack/lib/lang:
setenv SUBSYS_OUTDIR /local_space/mbmuser/node1010/samples/test1/temp
setenv EBMTMPDIR /tmp
--More-- (77%)

```

図 8-5 サブシステムの設定画面

13. エントリの確認後、**Return** キーを押して「**Query**」メニューを表示します。  
スペースバーまたは **Return** キーを押してエントリ内をスクロールします。
14. オプション「6 Query *subsystem-name*'s File\_Map」を選択し、サブシステムに関連付けられている `File_Map` を表示します。

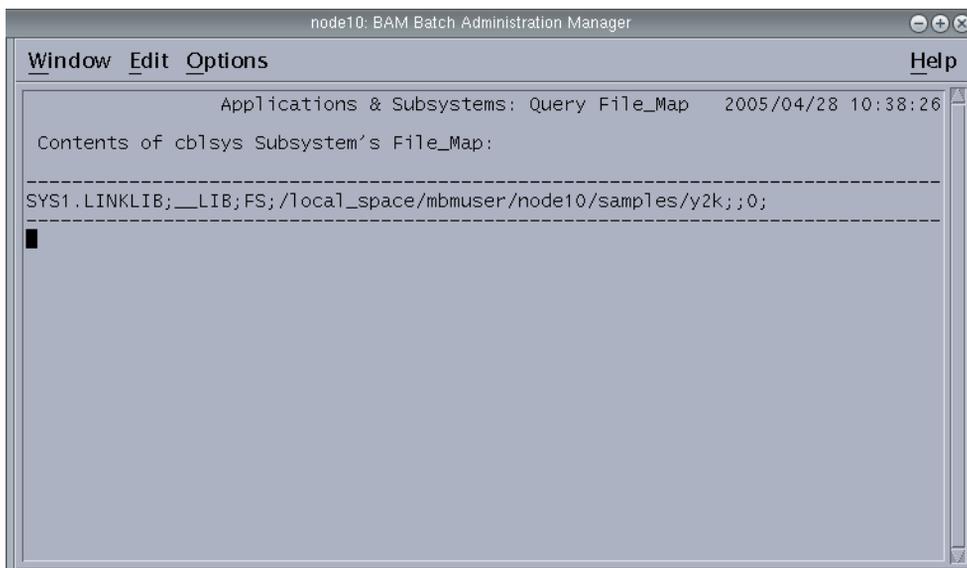


図 8-6 Query File\_Map 画面

15. エントリの確認後、**Return** キーを押して「**Query**」メニューを表示します。
16. オプション「7 Query *subsystem-name*'s User Setup」を選択し、サブシステムに関連付けられているユーザー設定ファイル (\$USER\_SETUP) を表示します。
17. Return キーを 2 回押して、「Applications & Subsystems」メニューを表示します。

---

## サブシステムの更新

サブシステムの「Update」メニューでは、ジョブ履歴ディレクトリの変更、Sun MTP 領域の変更、File\_Map の変更、およびユーザー設定ファイルのカスタマイズができます。

### ▼ サブシステムの Update メニューを表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。

2. 「Applications & Subsystems」 → 「Update a Subsystem」 を選択します。
3. 選択画面が表示されたら、更新するサブシステムの番号を入力して Return キーを押します。

「Update」メニューが表示されます。

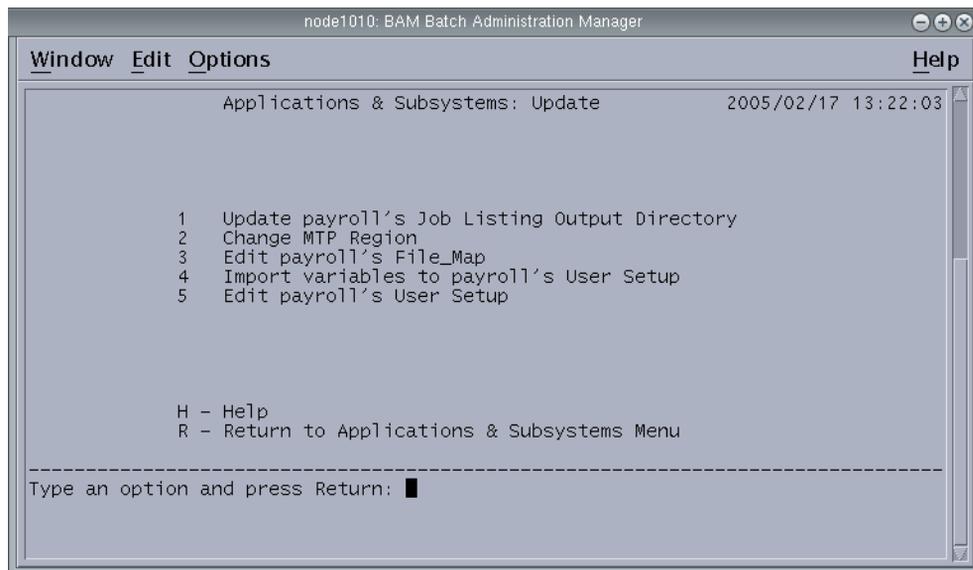


図 8-7 サブシステムの Update メニュー

## ▼ ジョブ履歴ディレクトリを変更する

1. サブシステムの「Update」メニューでオプション「1 Update *subsystem-name*'s Job Listing Output Directory」を選択します。

「Update Job Listing」画面に、デフォルトディレクトリおよびそのサブシステムのディレクトリ (定義済みの場合) に関する現在の設定が表示されます。
2. ディレクトリを変更するために *y* と入力します。
3. ディレクトリのパス名を入力します。あるいは *D* と入力し、ディレクトリ定義を削除してシステムデフォルトを使用します。
4. Return キーを押し、サブシステムの「Update」メニューを表示します。

---

注 - 変更したディレクトリは、すべてのユーザーが使用できるようにアクセス権が設定されていることを確認してください。また、ジョブ履歴ディレクトリは、ファイルシステムのオーバーフローを防ぐために監視する必要があります。ディスク容量の確認方法については、146 ページの「ディスク容量の確認」を参照してください。

---

## Sun MTP 領域の変更

サブシステムの接続先となる Sun MTP 領域 (\$KIXSYS) が変更された場合、サブシステムを領域に再同期する必要があります。この変更は、領域の再配置によって領域のパス名が (たとえば、/user/payroll11/sys から /user/payroll12/sys に) 変更された場合などに必要となります。\$KIXSYS 値は、\$KIXSYS ディレクトリを削除してから、パス名はそのままバックアップを基に復元した場合に変更されることもあります。これは、ディレクトリを削除するときに、Sun MBM が領域の識別に使用するノードも削除するために起こります。

---

注 - 新しいバージョンの Sun MTP をインストールした場合は、\$UNIKIX 環境変数の更新も必要になります。詳細は、『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。

---

### ▼ 領域を変更する

1. サブシステムの「Update」メニューでオプション「2 Change MTP Region」を選択します。
2. 図 8-8 の画面が表示されたら、次のいずれかを実行します。
  - \$KIXSYS ディレクトリを再配置した場合は、新しい領域のパス名を入力して Return キーを押します。
  - バックアップした \$KIXSYS ディレクトリを復元し、パス名が変わらない場合は、Return キーを押します。



図 8-8 MTP 領域の変更画面

3. 次の画面に、\$KIXSYS に対して行なった変更内容およびその変更を有効にするための手順が表示されます。

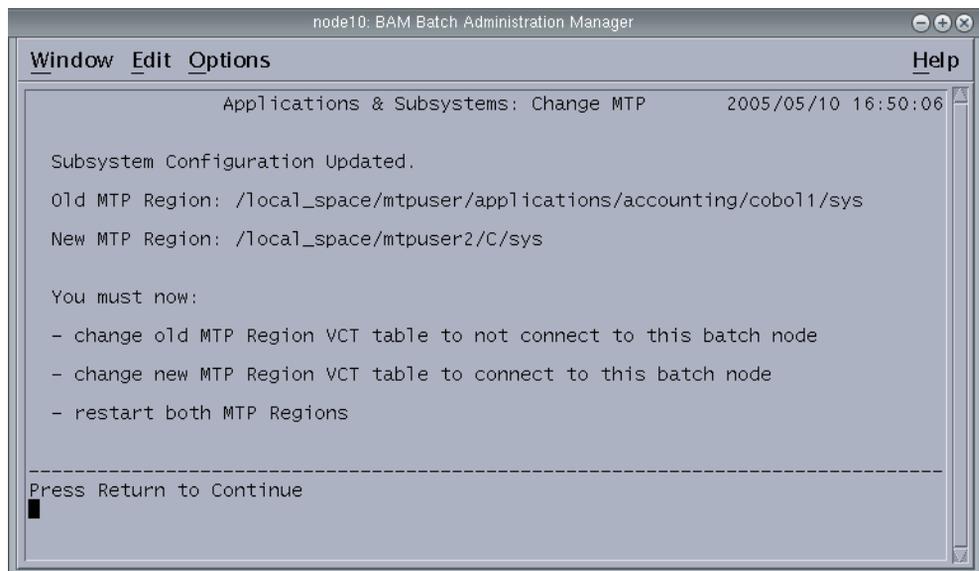


図 8-9 MTP 領域の変更確認画面

4. Return キーを押し、サブシステムの「Update」メニューを表示します。

## サブシステムの File\_Map の更新

File\_Map は、IBM データセット、ライブラリ、および世代データグループ (GDG) をそれぞれの UNIX パス名と関連付けるためのエントリがまとめられている特殊なファイルです。このファイルは、Sun MBM JCL トランスレータおよびサブシステムによって、メインフレーム JCL ストリームの変換時やマクロジョブスクリプトの実行時に使用されます。サブシステムはそれぞれ 1 つの File\_Map に関連付けられています。

ツールキットおよび cfm コマンドによる更新方法などの File\_Map の詳細は、『Sun Mainframe Batch Manager ソフトウェア 移行ガイド』を参照してください。また、一度に複数の File\_Map エントリを更新できる cfm コマンドの構文については、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。

### ▼ File\_Map を更新する

1. サブシステムの「Update」メニュー (図 8-7) でオプション「3 Edit subsystem-name's File\_Map」を選択します。これにより、テキストエディタで File\_Map ファイルが開かれます。

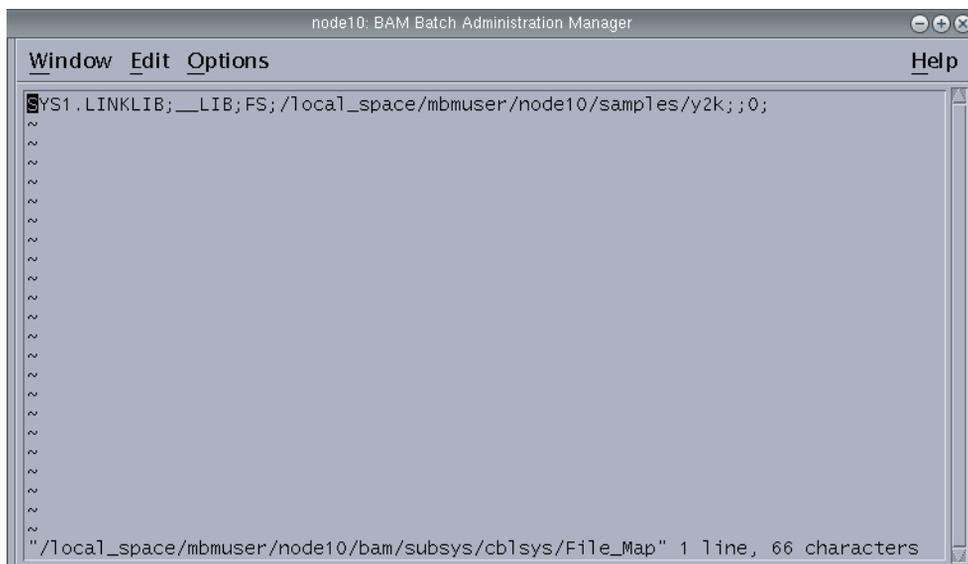


図 8-10 File\_Map ファイルの更新

2. 必要に応じて情報を変更してファイルを保存し、エディタを終了します。
3. Return キーを 2 回押して、「Applications & Subsystems」メニューを表示します。

## サブシステムのユーザー設定ファイルのカスタマイズ

BAM でサブシステムを作成すると、2つの設定ファイルが作成されます。1つは読み取り専用の編集できないファイル (\$SETUP) で、もう1つはユーザー定義の設定ファイルの値を取り込んだり直接編集したりしてカスタマイズできるユーザー設定ファイル (\$USER\_SETUP) です。取り込める設定ファイルは C シェル形式です。

別のノードからのインポートまたは削除/再作成操作によってサブシステムを再作成するたびに、BAM は新しいサブシステムの設定に基づいて \$SETUP ファイルを再作成しますが、\$USER\_SETUP の内容は変更しません。

BAM によって作成されるデフォルトの \$USER\_SETUP ファイルは空です。この節では、アプリケーション固有の環境変数を定義済み設定ファイルから組み込む方法について説明します。詳細は、98 ページの「ユーザー設定ファイルを編集する」も参照してください。

### ▼ 定義済み設定ファイルのエントリを \$USER\_SETUP に挿入する

1. 定義済み設定ファイルを C シェル形式で作成して保存します。

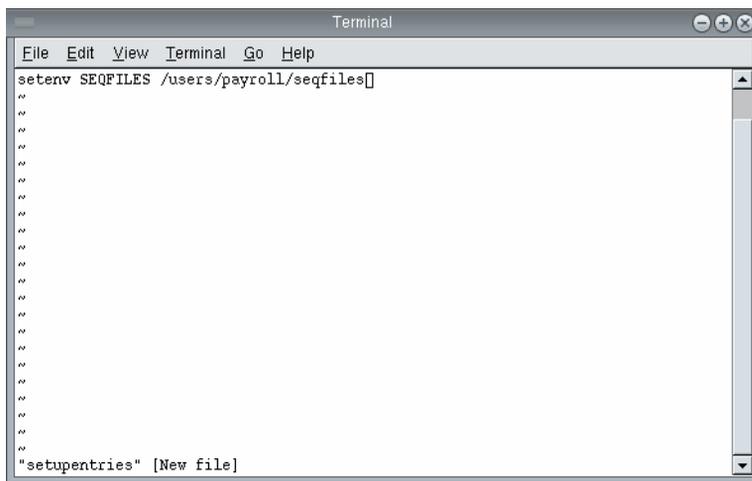


図 8-11 定義済み設定ファイル

2. サブシステムの「Update」メニューでオプション「4 Customize *subsystem-name*'s User Setup」を選択します。

3. 作成した設定ファイルのパス名を入力し、Return キーを押します。  
確認メッセージが表示されます。
4. Return キーを押し、サブシステムの「Update」メニューを表示します。

## ▼ ユーザー設定ファイルを編集する

1. 「Update」メニューで、オプション「5 Edit *subsystem-name*'s User Setup」を選択します。  
設定ファイルがテキストエディタで開かれます。
2. 必要に応じて情報を変更します。
3. ファイルを保存し、エディタを終了します。

---

## デフォルトのサブシステムの定義

デフォルトのサブシステムとは、ジョブをサブミットしたときにサブシステムが指定されていなかった場合にジョブが実行されるサブシステムです。デフォルトのサブシステムを定義すると、`subjob` または `unikixjob` コマンドを `-k` オプションなしでジョブをサブミットできます。

---

**注** – サブシステムがノードで1つしか定義されていない場合でも、それをデフォルトのサブシステムとして明示的に定義する必要があります。

---

## ▼ デフォルトのサブシステムを定義する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. 「Applications & Subsystems」 → 「Change/Show Default Subsystem」を選択して、「Default Subsystem」画面を表示します。

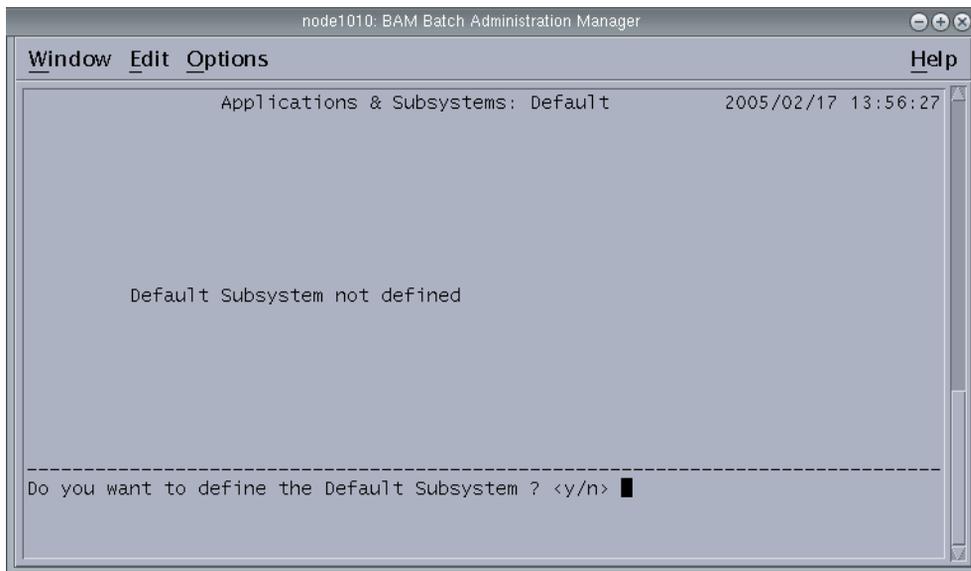


図 8-12 Default Subsystem 画面

3. プロンプトで、デフォルトのサブシステムを定義するかどうかを次のように指定します。
  - 次の画面 (定義済みサブシステムの一覧) を表示するには、`y` を入力し、手順 4 に進みます。
  - デフォルトのサブシステムの定義の操作を取り消すには、`n` を入力します。

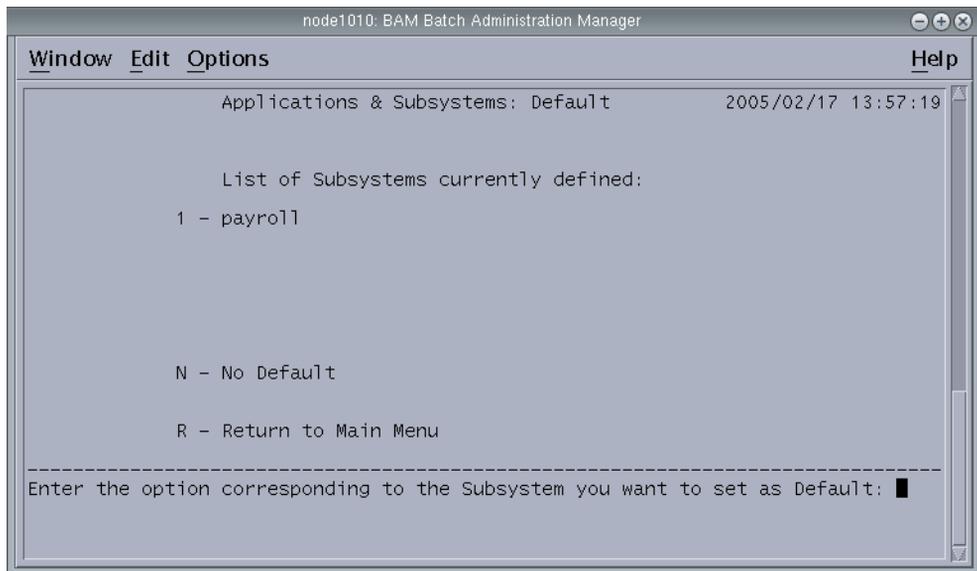


図 8-13 デフォルトのサブシステムの選択

4. デフォルトとして指定するサブシステムの番号を入力します。  
確認画面が表示されます。
5. Return キーを押して、「Applications & Subsystems」メニューを表示します。

---

## サブシステムの削除

サブシステムを削除できるのは、そのサブシステムの所有者または Sun MBM 管理者だけです。

サブシステムを削除する前に、次の点を確認してください。

- 該当のサブシステムに対してサブミットされているすべてのジョブを取り消す必要があります。ジョブの取り消しについては、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。
- 削除するサブシステムに接続されていた Sun MTP 領域は、そのノードに接続しないように領域の VSAM 構成テーブル (VCT) を変更し、領域を再起動する必要があります。

---

注 - 削除したサブシステムに対してジョブのサブミットはできません。

---

## ▼ サブシステムを削除する

- 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
- 「Applications & Subsystems」 → 「Delete a Subsystem」を選択します。
- 選択画面が表示されたら、削除するサブシステムの番号を入力し、Return キーを押します。

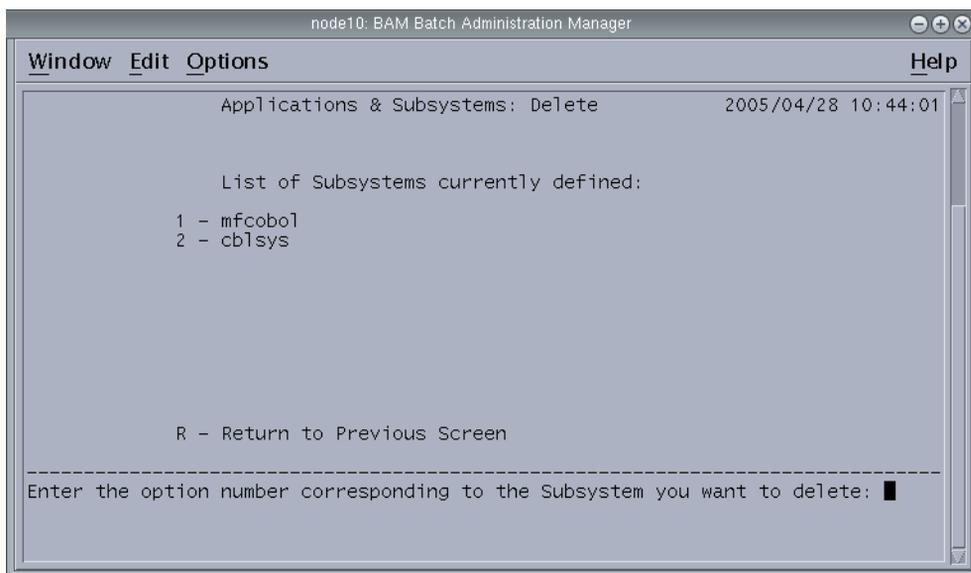


図 8-14 Delete Subsystem 画面

- サブシステムの特長画面が表示されたら、次のように実行します。
  - y と入力すると、サブシステムが削除されます。確認メッセージが表示されます。
  - n と入力すると、削除操作が取り消されます。
- Return キーを押して、「Applications & Subsystems」メニューを表示します。

---

注 - 削除したサブシステムを再び有効化することもできます。その場合は、「Application & Subsystems」メニューで「Create a Subsystem」を選択し、削除したサブシステムの名前を入力します。これは、構成からサブシステムを削除しても、そのサブシステムに関する設定値が BAM に保存されているためです。詳細は、102 ページの「削除したサブシステムの再作成」を参照してください。

---

# 削除したサブシステムの再作成

## ▼ 削除したサブシステムを再作成する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. 「Applications & Subsystems」 → 「Create a Subsystem」を選択します。
3. 再作成する削除したサブシステムの名前を入力し、Return キーを押します。
4. 図 8-15 の画面が表示されたら、サブシステムを再作成するかどうかを指定します。
  - y と入力すると、削除前の構成を使用してサブシステムが再作成されます。  
確認画面が表示されます。手順 6 に進みます。
  - n と入力すると、削除前の構成を使用しないで、次の手順に進みます。

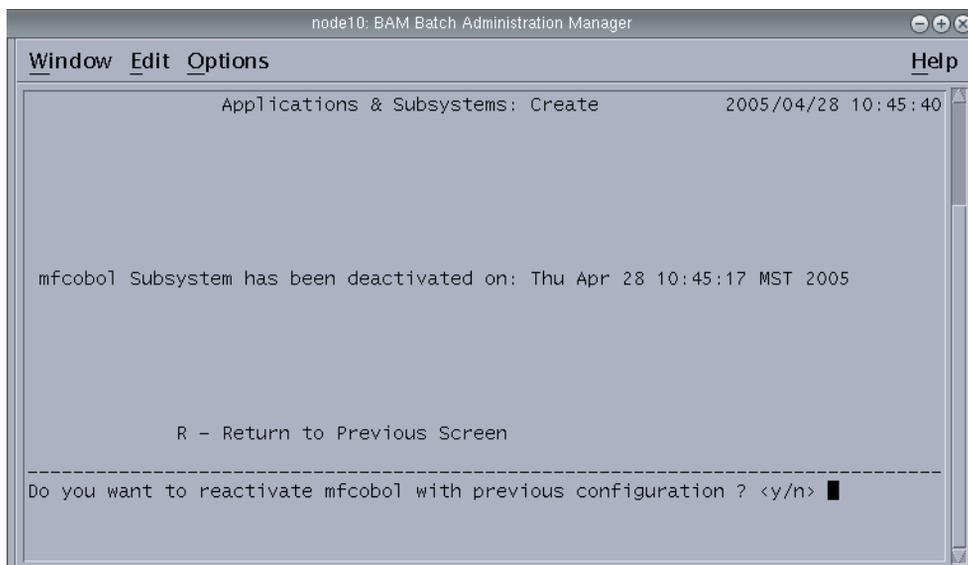


図 8-15 サブシステムの再作成画面

5. 図 8-16 の画面に、前の構成を削除する必要があることを示すメッセージが表示されたら、`y` と入力し、既存の構成を削除します。

---

注 - 構成を削除する前に、必要なときにバックアップとして使用できるように、その `File_Map` を別のディレクトリに保存することをお勧めします。

---

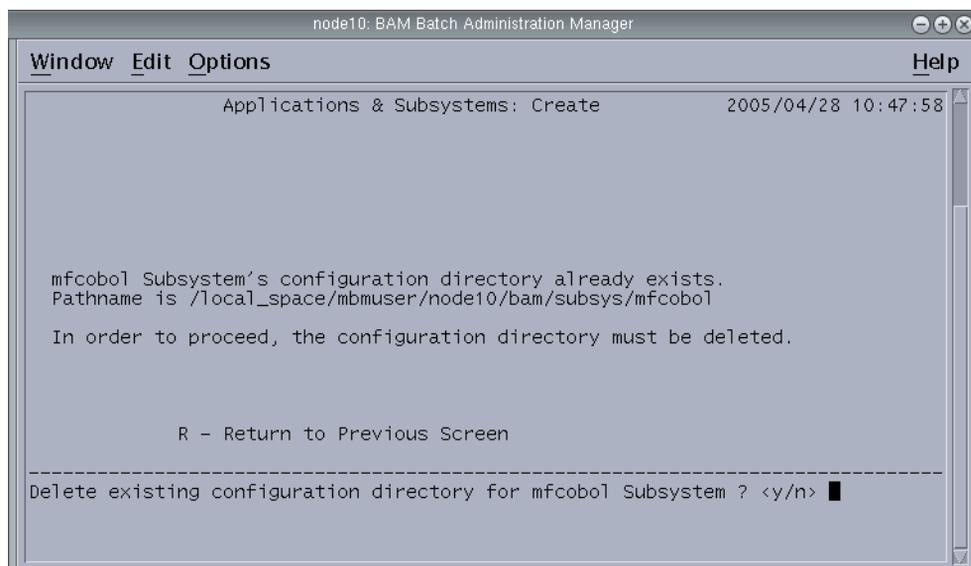


図 8-16 サブシステムの再作成の確認画面

6. サブシステムの構成値を設定します。  
詳細は、67 ページの「サブシステムの作成」を参照してください。

# サブシステムのインポート

Release 9.2 以降で作成されたサブシステムはインポートできます。「ソース」ノード (インポート元のノード) は、「ターゲット」ノード (インポート先のノード) と同じホスト上にあるか、ターゲットノードのホストマシンからソースノードが見えるようにマウントされているファイルシステムにある必要があります。

## ▼ サブシステムをインポートする

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メニューから「Applications & Subsystems」→「Import a Subsystem (from another node)」を選択します。

現在のノードに定義されているサブシステムを一覧表示する図 8-17 の画面が表示されます。現在のノードが新しいノードの場合、サブシステムは表示されません。

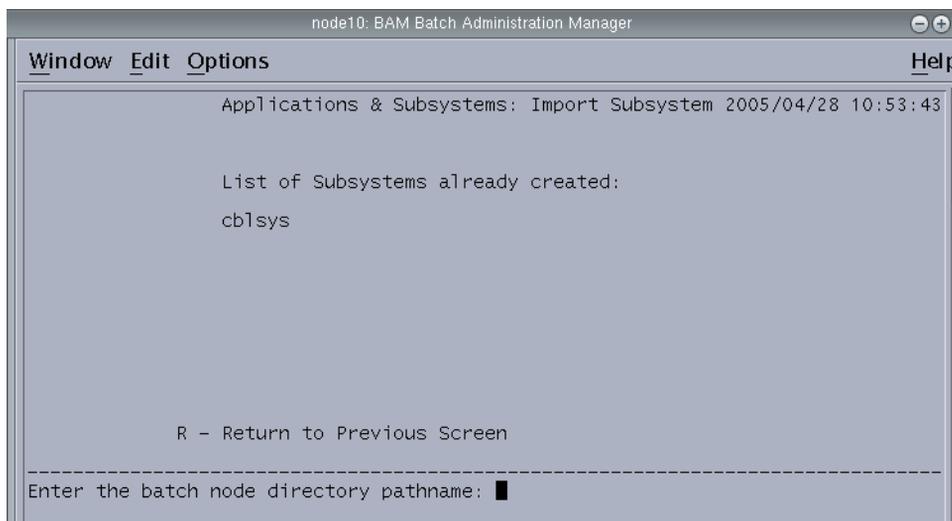


図 8-17 Import Subsystem - 既存のサブシステムリスト

3. プロンプトにソースノードのパス名を入力し、Return キーを押します。

ここに指定するのは、サブシステムのインポート元のノードです。次に例を示します。

```
Enter the batch node directory pathname:/local_space/mbmuser/node1010
```

ソースノードに定義されているサブシステムのリストが表示されます。

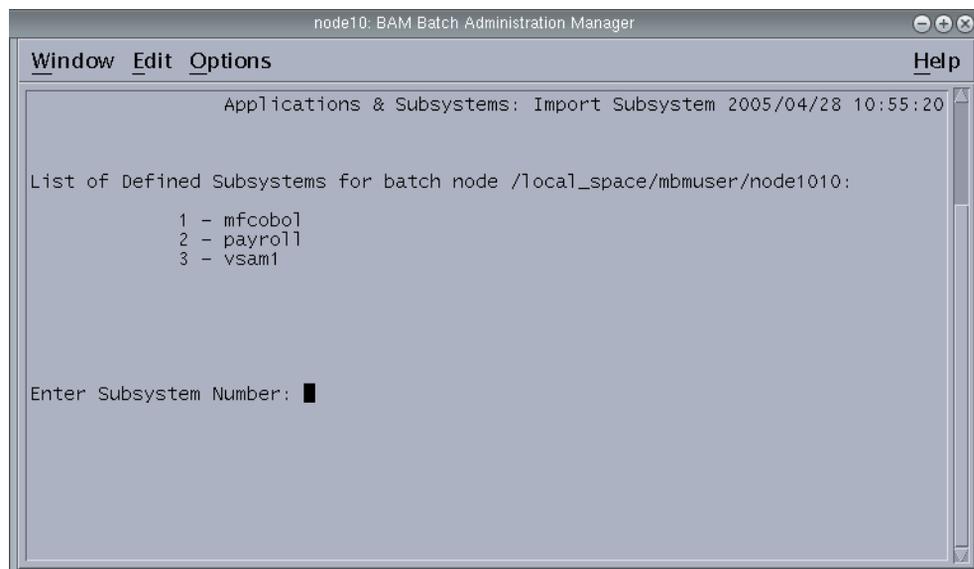


図 8-18 ソースノードサブシステム

4. インポートするサブシステムの番号を入力し、Return キーを押します。

同じノードからサブシステムをインポートする場合、一意の名前を入力するように求められます。

5. 「Applications & Subsystems」画面で次の手順を実行します。「Create」メニューで、サブシステムを作成する前に構成を変更します。

構成の選択については、第7章を参照してください。

6. 「Applications & Subsystems」画面で次の手順を実行します。「Create」メニューでオプション「c」を選択し、Return キーを押して現在のノードにサブシステムを作成します。

作成時に変更できる設定もあります。たとえば、新しいリリースの Sun MTP をインストールした場合は、新しい Sun MTP ディレクトリ (\$UNIXIX) を指定できます。

7. サブシステムが作成されて確認画面が表示されたら、Return キーを押します。

Sun MTP 領域に接続するサブシステムをインポートした場合、その領域の VSAM 構成テーブル (VCT) を更新するようにメッセージが表示されます。詳細は、付録 F を参照してください。

# セキュリティおよびユーザーの設定

---

ノードをインストールしてサブシステムを作成したあとで、ノードに対するセキュリティを設定する必要があります。また、Sun MBM コマンドを実行する権限およびサブシステムにジョブをサブミットする権限をどのユーザーに許可するかを定義する必要があります。デフォルトでは、すべてのコマンドの実行権限およびすべてのサブシステムに対するジョブのサブミット権限が全ユーザーに与えられます。

この章では、次の作業を行うための BAM ユーティリティの使用法について説明します。

- 特定のサブシステムへのアクセスをユーザー別またはグループ別に制御します。詳細は、108 ページの「サブシステムへのアクセス制御」を参照してください。
- ノードの起動/停止、ノードまたはサブシステムの管理、およびクラスとアクティビティの作成/削除に関する Sun MBM コマンドの実行権限を制御します。詳細は、110 ページの「Sun MBM コマンドへのアクセス制御」を参照してください。
- ユーザー固有の作業用ディレクトリを作成します。詳細は、119 ページの「ユーザーの作業用ディレクトリの管理」を参照してください。

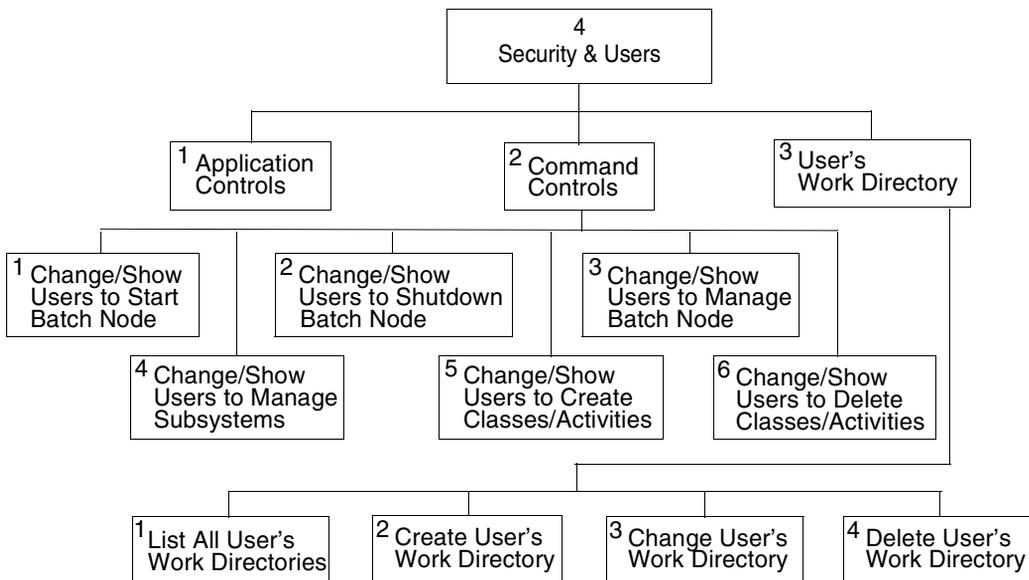


図 9-1 セキュリティーとユーザーの設定メニュー構造

## サブシステムへのアクセス制御

特定のサブシステムへのアクセスは、「Security & Users」メニューの「Application Controls」オプションでユーザー別またはグループ別に制御できます。デフォルト動作では、サブシステムにすべてのユーザーのアクセスが可能です。

### ▼ サブシステムへアクセスできるユーザーまたはグループを指定する

- 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
- メインメニューでオプション「4」を選択し、「Security & Users」メニューを表示します。

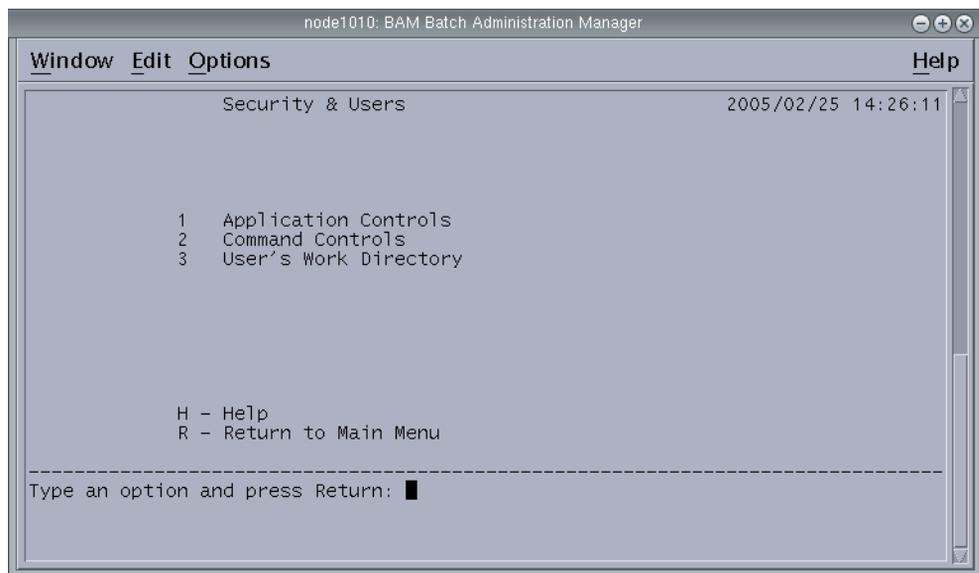


図 9-2 Security & Users メニュー

3. オプション「1 Application Controls」を選択し、特定のサブシステムへのアクセスを許可するユーザーまたはグループを指定します。

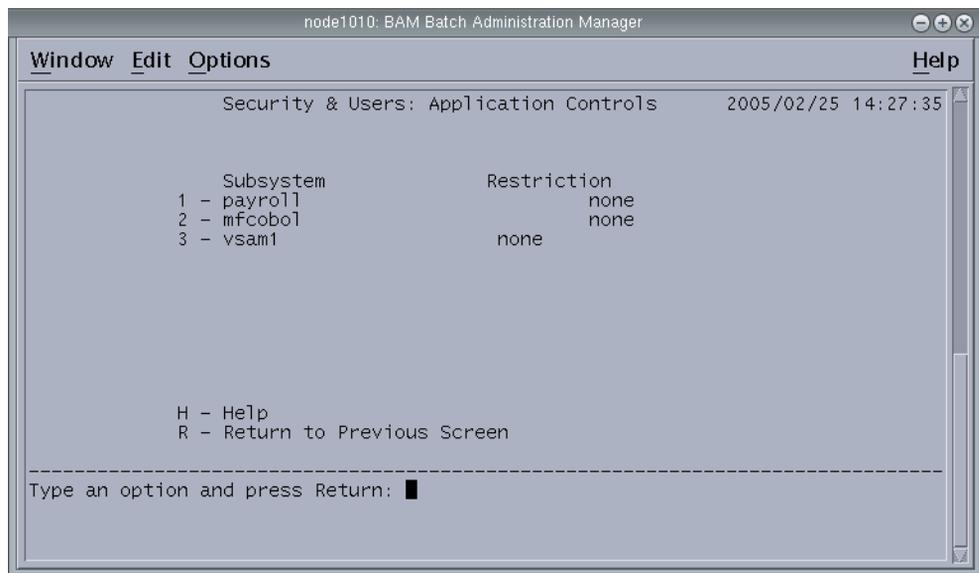


図 9-3 Application Controls 画面

4. 「Application Controls」画面のプロンプトに従って、次のように実行します。
  - a. サブシステムの番号を入力します。
  - b. サブシステムに設定するアクセス制限のタイプ (user、group、none) を指定します。
  - c. 有効なユーザー名またはグループ名を入力します。
  - d. Return キーを押します。確認メッセージが表示されます。
5. Return キーを 2 回押し、「Security & Users」メニューを表示します。

---

## Sun MBM コマンドへのアクセス制御

この節および小節では、次の権限を持つユーザーを追加、変更、または表示するための「Command Controls」メニューオプションの使用方法について説明します。

- ノード (システム) の起動または停止
- ノードまたはサブシステムの管理
- クラスまたはアクティビティーの作成と削除

アクセスの各タイプは、編集可能な許可ファイルによって制御されます。

### ▼ Command Controls メニューを表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. メインメニューで「Security & Users」を選択します。
3. オプション「2」を選択し、「Command Controls」メニューを表示します。  
図 9-4 に示す画面が表示されます。

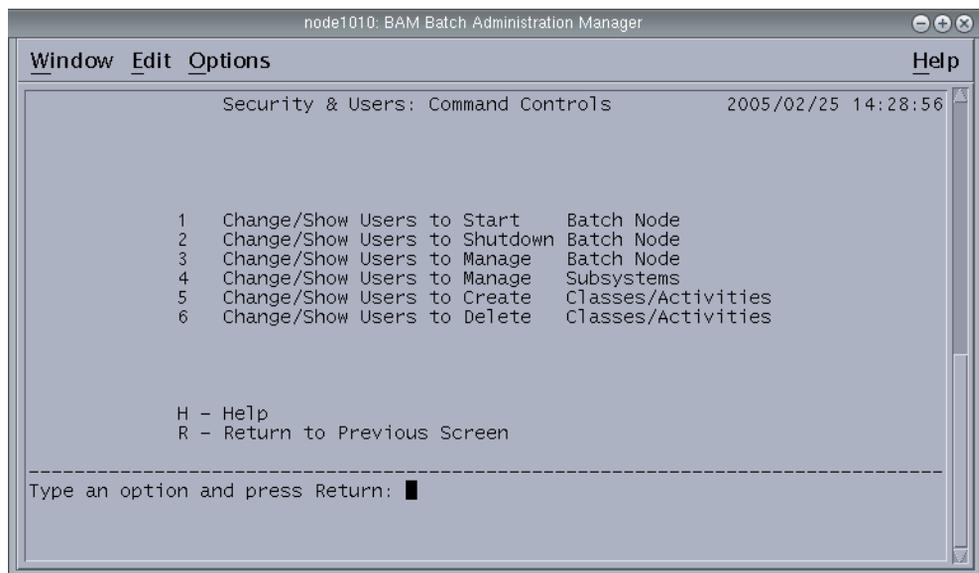
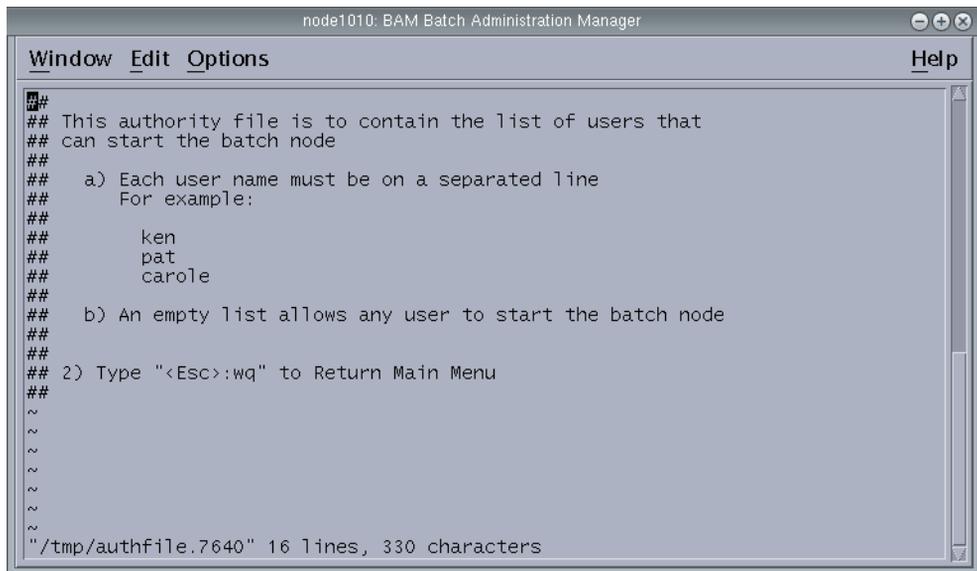


図 9-4 Command Controls メニュー

## ▼ ノードを起動できるユーザーを追加または変更する

1. 「Command Controls」メニューで、オプション「1」を選択し、ノードを起動できるユーザーのリストを表示します。  
ユーザーが表示されない場合は、どのユーザーでもノードを起動できます。
2. リストにユーザーを追加したり変更したりするには、`y` と入力します。
3. 許可ファイルが開かれたら、ノードの起動を許可されているユーザーのリストを確認します。



```
node1010: BAM Batch Administration Manager
Window Edit Options Help
##
## This authority file is to contain the list of users that
## can start the batch node
##
## a) Each user name must be on a separated line
## For example:
##
##     ken
##     pat
##     carole
##
## b) An empty list allows any user to start the batch node
##
##
## 2) Type "<Esc>:wq" to Return Main Menu
##
~
~
~
~
~
~
~
~/tmp/authfile.7640" 16 lines, 330 characters
```

図 9-5 ノード起動の許可ファイル

4. 次のように変更します。

a. ユーザー ID を追加または削除します。

行頭に ## は入力しないでください。

b. ファイルを保存します。

c. テキストエディタを終了します。

- 正常に更新されると、次のメッセージが表示されます。

```
# Configuration updated
```

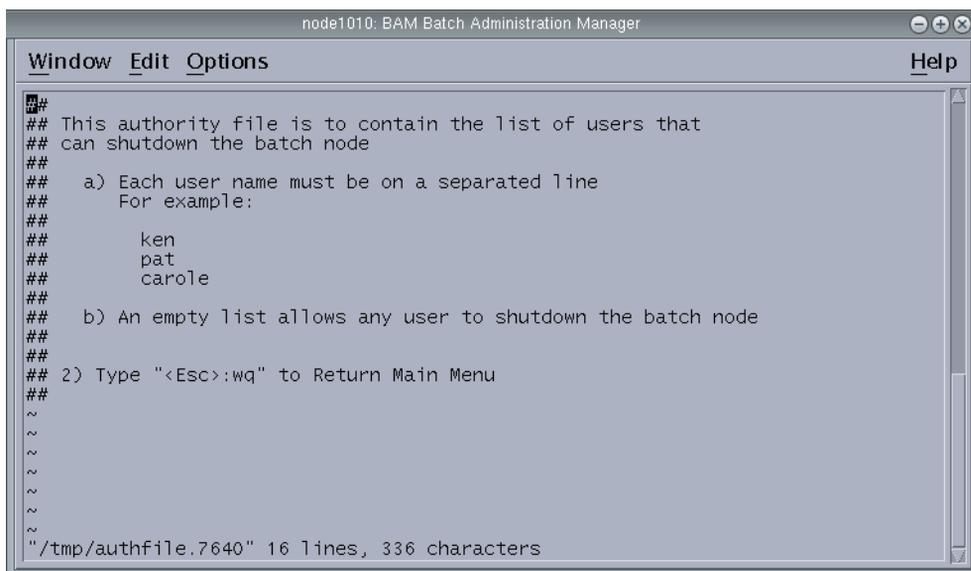
- 更新できなかった場合は、次のメッセージが表示されます。

```
# Current configuration not updated
Press Return to Continue
```

5. Return キーを押し、「Command Controls」メニューを表示します。

## ▼ ノードを停止できるユーザーを追加または変更する

1. 「Command Controls」メニューでオプション「2」を選択し、ノードの停止および現在の設定を表示できるユーザーを指定します。  
ユーザーが表示されない場合は、どのユーザーでもノードを停止できます。
2. リストにユーザーを追加したり変更したりするには、`y` と入力します。
3. 許可ファイルが開かれたら、ノードの停止を許可されているユーザーのリストを確認します。



```
node1010: BAM Batch Administration Manager
Window Edit Options Help
##
## This authority file is to contain the list of users that
## can shutdown the batch node
##
## a) Each user name must be on a separated line
## For example:
##
##     ken
##     pat
##     carole
##
## b) An empty list allows any user to shutdown the batch node
##
## 2) Type "<Esc>;wq" to Return Main Menu
##
~
~
~
~
~
~
~/tmp/authfile.7640" 16 lines, 336 characters
```

図 9-6 ノード停止許可ファイル

4. 次のように変更します。
  - a. ユーザーの名前を追加または変更します。  
行頭に ## は入力しないでください。
  - b. ファイルを保存します。
  - c. テキストエディタを終了します。
    - 正常に更新されると、次のメッセージが表示されます。  
# Configuration updated

- 更新できなかった場合は、次のメッセージが表示されます。

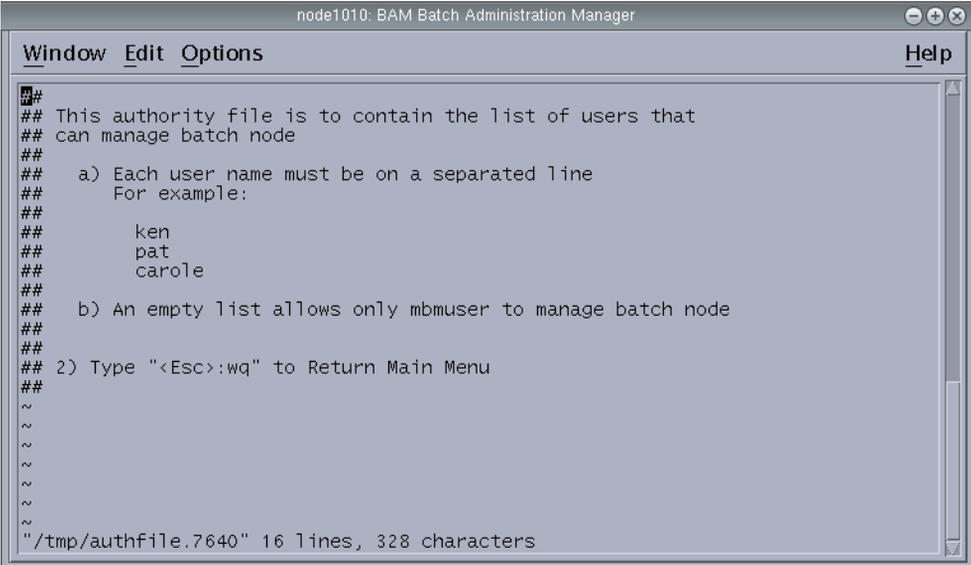
```
# Current configuration not updated
Press Return to Continue
```

5. Return キーを押し、「Command Controls」メニューを表示します。

## ▼ ノードを管理できるユーザーを追加または変更する

「Command Controls」メニューのオプション「3」を選択すると、BAM の実行中に Sun MBM 管理者として作業可能なユーザーを定義できます。デフォルト設定では、BAM セッション中に管理者ユーザーとして作業できるのは MBM ノードの所有者だけです。

1. 「Command Controls」メニューでオプション「3」を選択し、ノードの管理および現在の設定の表示ができるユーザーのリストを表示します。  
ユーザーが表示されない場合は、ノードの所有者だけがそのノードを管理できます。
2. リストにユーザーを追加したり変更したりするには、`y` と入力します。
3. 許可ファイルが開かれたら、ノードの管理を許可されているユーザーのリストを確認します。



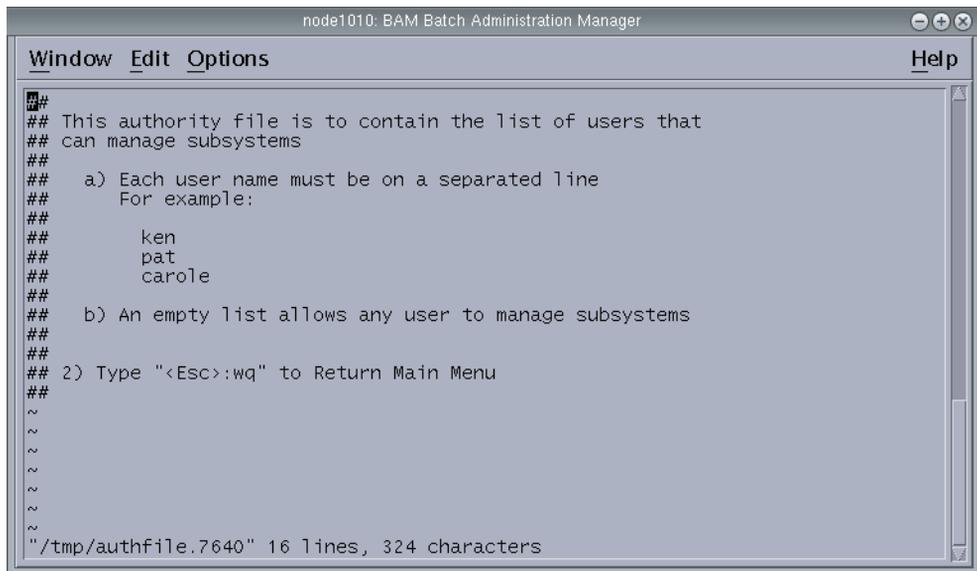
```
node1010: BAM Batch Administration Manager
Window Edit Options Help
##
## This authority file is to contain the list of users that
## can manage batch node
##
## a) Each user name must be on a separated line
## For example:
##
##     ken
##     pat
##     carole
##
## b) An empty list allows only mbmuser to manage batch node
##
## 2) Type "<Esc>:wq" to Return Main Menu
##
~
~
~
~
~/tmp/authfile.7640" 16 lines, 328 characters
```

図 9-7 ノード管理の許可ファイル

4. 次のように変更します。
  - a. ユーザーの名前を追加または変更します。  
行頭に ## は入力しないでください。
  - b. ファイルを保存します。
  - c. テキストエディタを終了します。
    - 正常に更新されると、次のメッセージが表示されます。  
`# Configuration updated`
    - 更新できなかった場合は、次のメッセージが表示されます。  
`# Current configuration not updated`  
`Press Return to Continue`
5. Return キーを押し、「Command Controls」メニューを表示します。

## ▼ サブシステムを管理できるユーザーを追加または変更する

1. 「Command Controls」メニューでオプション「4」を選択し、サブシステムの管理および現在の設定の表示ができるユーザーのリストを表示します。  
ユーザーが表示されない場合は、ノードの所有者だけがサブシステムを管理できません。
2. リストにユーザーを追加したり変更したりするには、`y` と入力します。
3. 許可ファイルが開かれたら、サブシステムの管理を許可されているユーザーのリストを確認します。



```
node1010: BAM Batch Administration Manager
Window Edit Options Help
##
## This authority file is to contain the list of users that
## can manage subsystems
##
## a) Each user name must be on a separated line
## For example:
##
##     ken
##     pat
##     carole
##
## b) An empty list allows any user to manage subsystems
##
## 2) Type "<Esc>:wq" to Return Main Menu
##
~
~
~
~
~
~
~/tmp/authfile.7640" 16 lines, 324 characters
```

図 9-8 サブシステム管理の許可ファイル

4. 次のように変更します。

a. ユーザーの名前を追加または変更します。

行頭に ## は入力しないでください。

b. ファイルを保存します。

c. テキストエディタを終了します。

- 正常に更新されると、次のメッセージが表示されます。

```
# Configuration updated
```

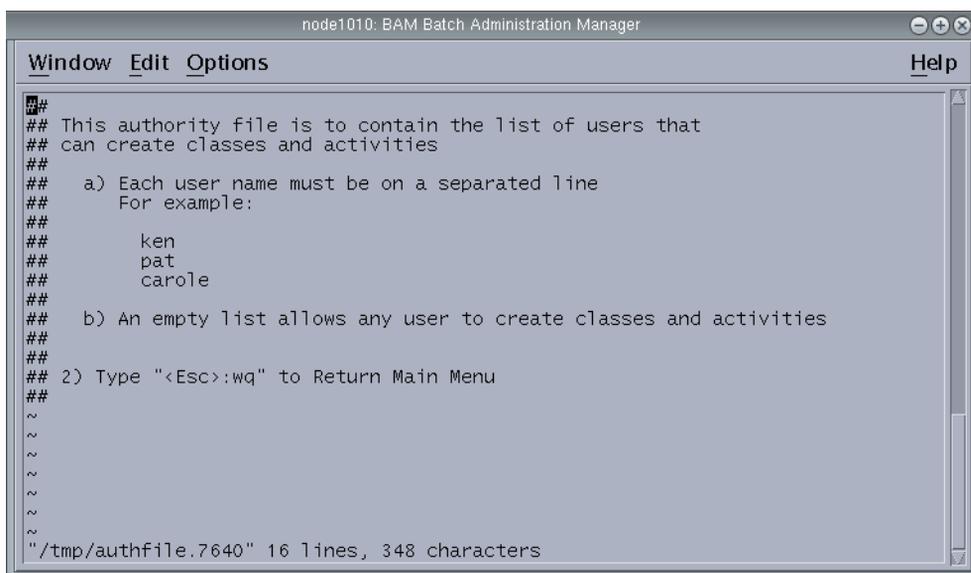
- 更新できなかった場合は、次のメッセージが表示されます。

```
# Current configuration not updated
Press Return to Continue
```

5. Return キーを押し、「Command Controls」メニューを表示します。

## ▼ クラスおよびアクティビティーを作成できるユーザーを追加または変更する

1. 「Command Controls」メニューでオプション「5」を選択し、アクティビティーの作成/変更および現在の設定の表示ができるユーザーを指定します。  
ユーザーが表示されない場合は、どのユーザーでもアクティビティーを作成および変更できます。
2. リストにユーザーを追加したり変更したりするには、y と入力します。
3. 許可ファイルが開かれたら、クラスおよびアクティビティーの作成または変更を許可されているユーザーのリストを確認します。



```
node1010: BAM Batch Administration Manager
Window Edit Options Help
##
## This authority file is to contain the list of users that
## can create classes and activities
##
## a) Each user name must be on a separated line
## For example:
##
##     ken
##     pat
##     carole
##
## b) An empty list allows any user to create classes and activities
##
## 2) Type "<Esc>;wq" to Return Main Menu
##
~
~
~
~
~
~
~/tmp/authfile.7640" 16 lines, 348 characters
```

図 9-9 クラスとアクティビティーの作成の許可ファイル

4. 次のように変更します。
  - a. ユーザーの名前を追加または変更します。  
行頭に ## は入力しないでください。
  - b. ファイルを保存します。
  - c. テキストエディタを終了します。
    - 正常に更新されると、次のメッセージが表示されます。  
# Configuration updated

- 更新できなかった場合は、次のメッセージが表示されます。

```
# Current configuration not updated
Press Return to Continue
```

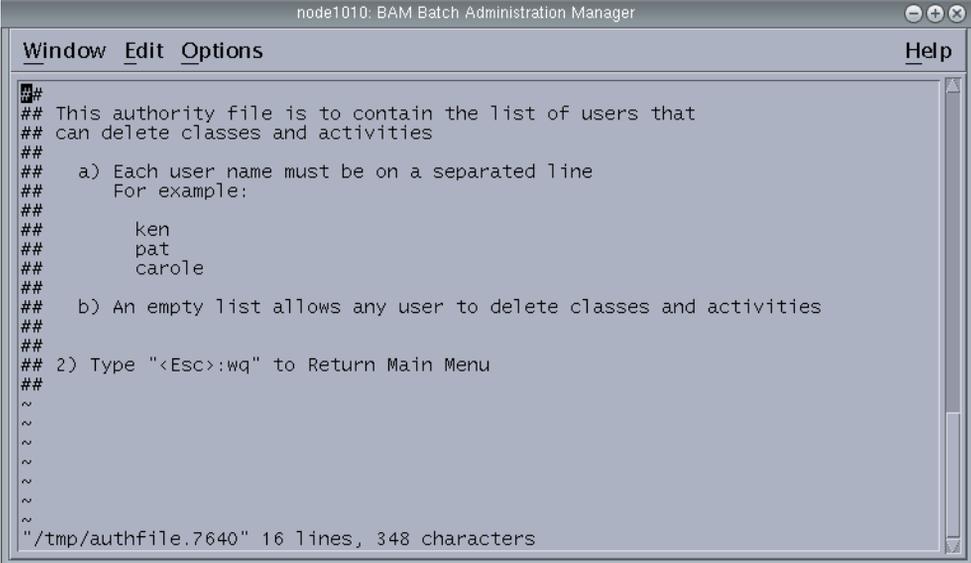
5. Return キーを押し、「Command Controls」メニューを表示します。

## ▼ クラスおよびアクティビティを削除できるユーザーを追加または変更する

1. 「Command Controls」メニューでオプション「6」を選択し、クラスおよびアクティビティを削除できるユーザーを指定します。

ユーザーが表示されない場合は、どのユーザーでもクラスとアクティビティを削除できます。

2. リストにユーザーを追加したり変更したりするには、`y` と入力します。
3. 許可ファイルが開かれたら、クラスおよびアクティビティの削除を許可されているユーザーのリストを確認します。



```
node1010: BAM Batch Administration Manager
Window Edit Options Help
##
## This authority file is to contain the list of users that
## can delete classes and activities
##
## a) Each user name must be on a separated line
## For example:
##     ken
##     pat
##     carole
##
## b) An empty list allows any user to delete classes and activities
##
## 2) Type "<Esc>;wq" to Return Main Menu
##
~
~
~
~
~
~/tmp/authfile.7640" 16 lines, 348 characters
```

図 9-10 クラスとアクティビティの削除の許可ファイル

4. 次のように変更します。
  - a. ユーザーの名前を追加または変更します。  
行頭に ## は入力しないでください。
  - b. ファイルを保存します。
  - c. テキストエディタを終了します。
    - 正常に更新されると、次のメッセージが表示されます。  
`# Configuration updated`
    - 更新できなかった場合は、次のメッセージが表示されます。  
`# Current configuration not updated`  
`Press Return to Continue`
5. Return キーを 2 回押し、「Security & Users」メニューを表示します。

---

## ユーザーの作業用ディレクトリの管理

この節では、次の方法について説明します。

- すべてのユーザーに定義されている作業用ディレクトリを一覧表示します。  
作業用ディレクトリが定義されていないユーザーがジョブをサブミットした場合、そのジョブはユーザーのホームディレクトリで実行されます。
- ユーザーの作業用ディレクトリを作成します。  
ユーザーはディレクトリの所有者であり、実行、読み取り、書き込みが許可されている必要があります。
- 指定したユーザーの作業用ディレクトリを変更します。  
ユーザーはディレクトリの所有者であり、実行、読み取り、書き込みが許可されている必要があります。
- ユーザーの作業用ディレクトリを削除します。



---

**注意** – 複数のノードを実行する場合、任意のノードで複数のユーザー作業用ディレクトリが同一のディレクトリを指さないようにしてください。ノードを識別できないため、同じジョブ番号のジョブでファイル名が重複することがあります。

---

## ▼ User's Work Directory メニューを表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メニューで「Security & Users」を選択します。
3. 「Security & Users」メニューでオプション「3 User's Work Directory」を選択します。

詳細は、図 9-11 を参照してください。

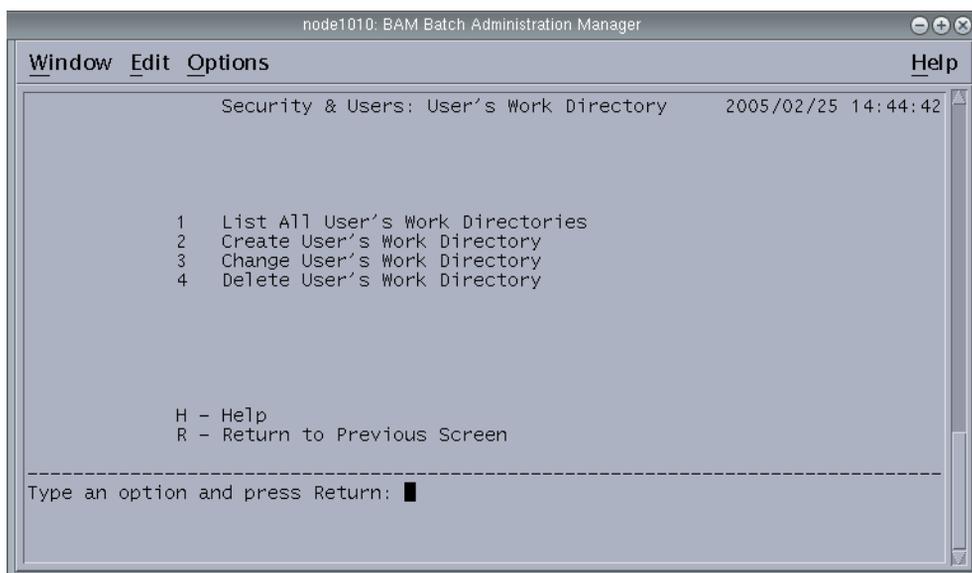


図 9-11 User's Work Directory メニュー

## ▼ すべてのユーザーの作業用ディレクトリを一覧表示する

1. 「User's Work Directory」メニューで、オプション「1 List All User's Work Directories」を選択します。

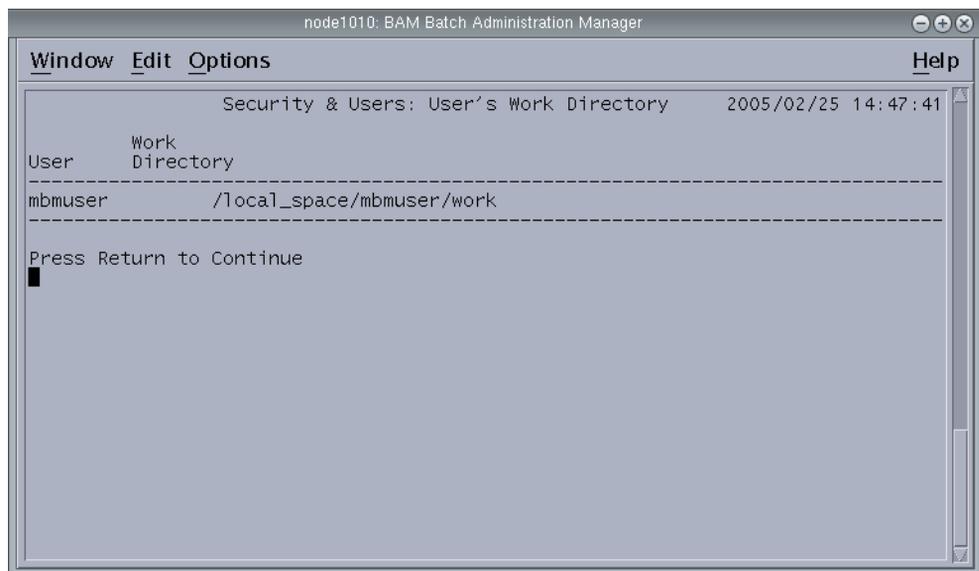


図 9-12 User's Work Directory 一覧

2. Return キーを押し、「User's Work Directory」メニューを表示します。

## ▼ ユーザーの作業用ディレクトリを作成する

1. 「User's Work Directory」メニューでオプション「2 Create User's Work Directory」を選択します。



図 9-13 User's Work Directory 作成

2. ユーザーの名前を入力し、Return キーを押します。
3. ユーザーの作業用ディレクトリのパスを入力し、Return キーを押します。

---

注 - ユーザーがディレクトリの読み取り権、書き込み権、および実行権を持っていることを確認してください。

---

作業用ディレクトリが作成されると、確認画面が表示されます。

4. Return キーを押し、「User's Work Directory」メニューを表示します。

## ▼ ユーザーの作業用ディレクトリを変更する

1. 「User's Work Directory」メニューでオプション「3 Change User's Work Directory」を選択します。

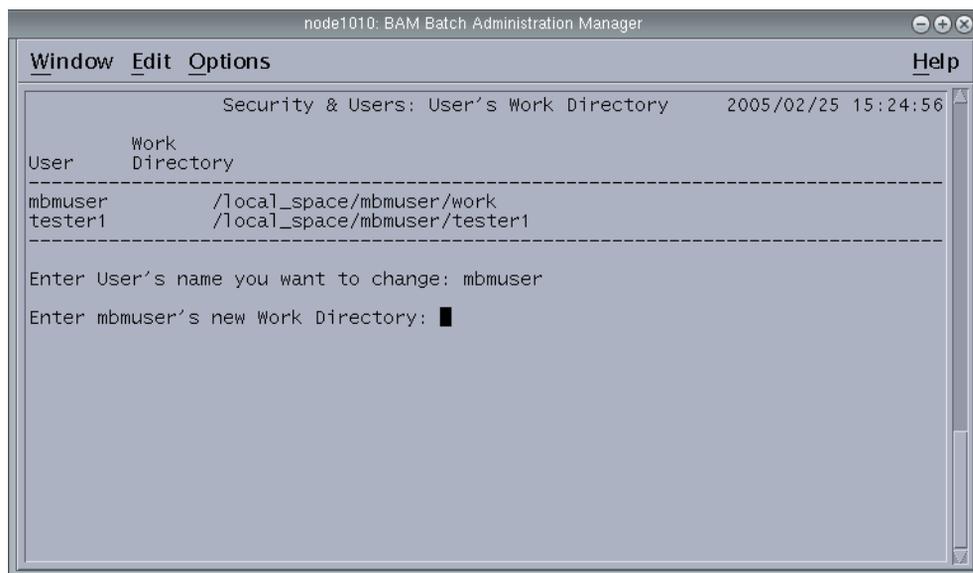


図 9-14 User's Work Directory 変更

2. ユーザーの名前を入力し、Return キーを押します。
3. ユーザーの新しい作業用ディレクトリのパスを入力し、Return キーを押します。  
作業用ディレクトリが変更されると、確認画面が表示されます。
4. Return キーを押し、「User's Work Directory」メニューを表示します。

## ▼ ユーザーの作業用ディレクトリを削除する

1. 「User's Work Directory」メニューでオプション「4 Delete User's Work Directory」を選択します。

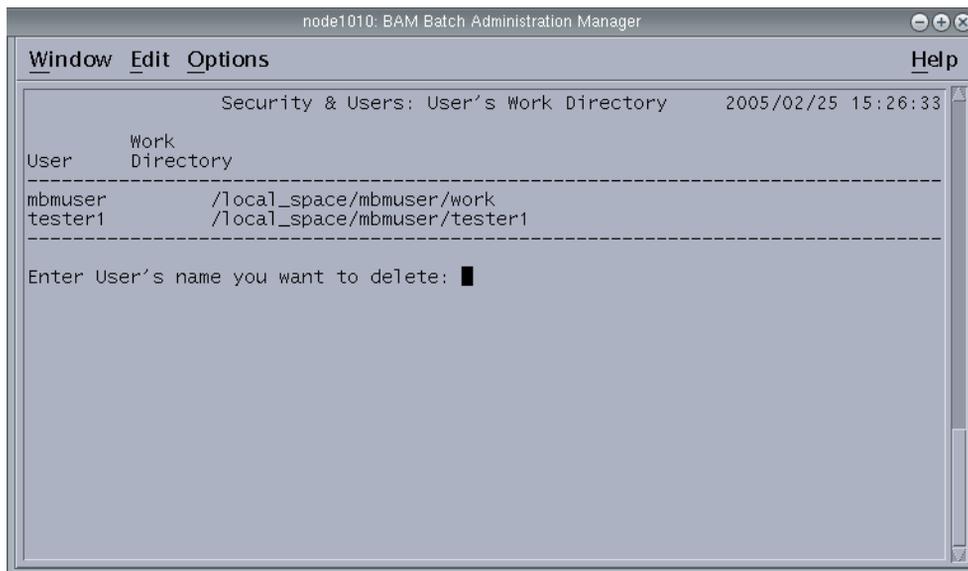


図 9-15 User's Work Directory 削除

2. 作業用ディレクトリを削除するユーザーの名前を入力し、Return キーを押します。
3. 作業用ディレクトリが削除されると、確認画面が表示されます。  
ユーザーの作業用ディレクトリを削除したあと、そのユーザーによってサブミットされるジョブは、そのユーザーのホームディレクトリで実行されます。
4. Return キーを押し、「User's Work Directory」メニューを表示します。

# Sun MBM コマンドを実行するためのユーザー環境の設定

この節では、特定のサブシステムに対して Sun MBM コマンドを実行できるようにするためのユーザー環境の設定方法について説明します。

## ▼ メインメニューから環境を設定する

1. Sun MBM メインメニューで、「Command Prompt」アイコンをクリックします。使用可能なサブシステムを一覧表示する次の画面が表示されます。



図 9-16 Command Prompt 画面

2. 設定するサブシステムの名前を入力し、Return キーを押します。これにより、そのサブシステムの設定ファイルがソースとなります。

## ▼ コマンド行から環境を設定する

1. システムにログインし、Sun MBM ノードがインストールされているディレクトリに移動します。

たとえば、ノードのインストールディレクトリが `/mybatch/node1` である場合は、次のコマンドを入力します。

```
$ cd /mybatch/node1
```

2. このノードにグローバル環境変数を設定し、特定のサブシステムの環境変数を継承するには、`batchenv` にそのサブシステム名を指定します。

たとえば、使用可能なサブシステムが `prod1` の場合は次のように入力します。

```
$ . ./batchenv prod1
```

実行中のセッションに `prod1` の環境変数が継承され、プロンプトは次のように変わります。

```
[node1/prod1] #
```

---

**参考** – 使用可能なサブシステムを一覧表示するには、66 ページの「使用可能なサブシステムの一覧表示」を参照してください。

---

## 第10章

# ジョブクラスおよびアクティビティーの設定

---

この章では、特定のノードのジョブクラスとアクティビティーの設定、作成、および変更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。

- 128 ページの「ジョブクラスの一覧表示」
- 130 ページの「ジョブクラスおよびアクティビティーの作成」
- 132 ページの「アクティビティー数の変更」
- 133 ページの「ジョブクラスの削除」

Sun MBM では、ジョブクラスを使用してバッチジョブの実行を制御できます。デフォルトクラスで実行しない場合は、ジョブをサブミットするたびにジョブクラスを指定する必要があります。「アクティビティー」(スレッド)と呼ばれる1つ以上の「パーティション」を持ったクラスを設定できます。パーティションとは、共有メモリーのセグメントのことです。

各ジョブは1つのアクティビティー内で実行され、そのアクティビティーはジョブが完了するまで使用されます。1つのクラスで同時に実行できるジョブの数は、アクティビティーの数によって決まります。たとえば、5つのアクティビティーを持つクラス e を設定し、30件のジョブをサブミットすると、クラス e では30件のジョブのうち5件だけが同時に実行されます。ジョブのいずれかが終了すると、使用可能になったアクティビティーで実行待ちのジョブを実行するように Sun MBM がスケジューリングします。定義できるクラスは a ~ z までの最大 26 クラスです。デフォルトでは、ジョブはクラス a に分類されます。クラス内のジョブの実行スケジュールは、それぞれのジョブの優先順位によって決定されます。最高優先順位のジョブが最初にスケジューリングされます。

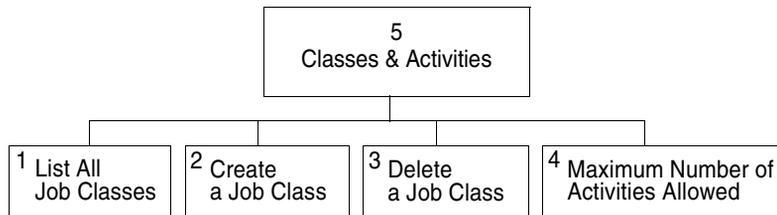


図 10-1 ジョブクラスとアクティビティの設定メニュー構造

---

## ジョブクラスの一覧表示

ジョブクラスを一覧表示すると、そのノードのすべてのクラスが表示されます。

### ▼ すべてのジョブクラスを一覧表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。

2. BAM メニューでオプション「5 Classes & Activities」を選択します。

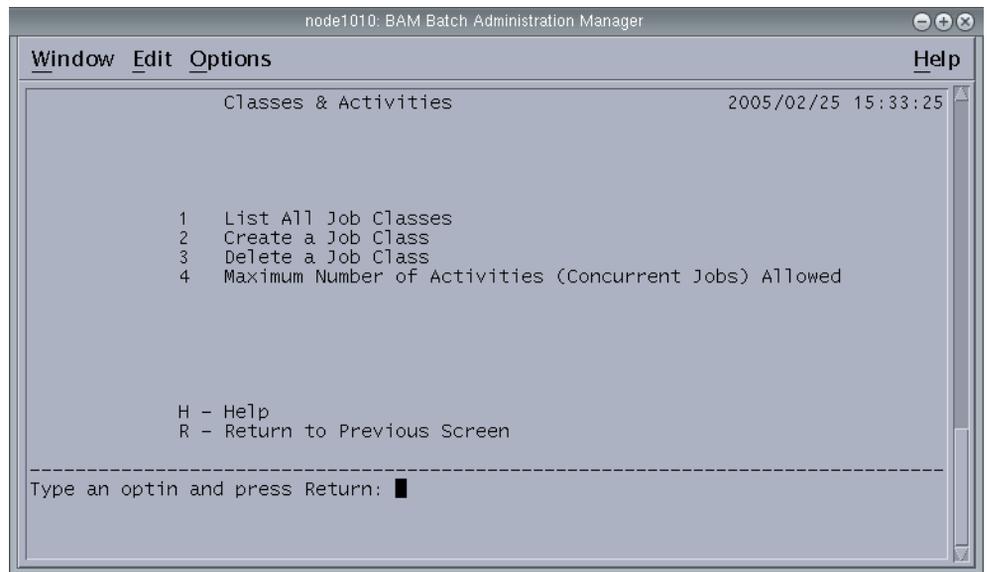


図 10-2 Classes & Activities メニュー

3. オプション「1 List All Job Classes」を選択し、図 10-3 の画面を表示します。

クラスごとに次の情報が表示されます。

- Active Threads : 有効なジョブによって使用されているスレッドの数。
- Available Threads : 使用可能なスレッドの数。
- Total Threads : スレッドの合計数 (並行して実行できるジョブの最大数)。
- Pending Jobs : 待機中のジョブの数。通常は、使用可能なスレッドを待機するジョブです。または、ジョブが VSAM ファイルを使用している場合は、Sun MTP トランザクション処理プログラムを待機するジョブです。

node1010: List Job Classes

Window Edit Options Help

System Environments: List Job Classes 2005/02/25 15:37:15

Class	Active Threads	Available Threads	Total Threads	Pending Jobs
a	00	08	08	00
b	00	02	02	00
-----				
Total	00	10	10	00

A[class] - Active Jobs      P[class] - Pending Jobs      X - Exit to Main Menu

図 10-3 List ALL Job Classes 画面

4. x と入力して Return キーを押し、「Classes & Activities」メニューを表示します。

## ジョブクラスおよびアクティビティの作成

ジョブが実行されるには、ジョブクラスが存在していて、アクティビティ (スレッド) が使用可能になっている必要があります。ジョブクラスおよびアクティビティが用意されていないと、ジョブは保持状態のままになります。初めてノードを起動すると、デフォルトのジョブクラスが 2 つ作成されます。1 つはクラス a で 8 つの有効のスレッドがあり、もう 1 つはクラス b で 2 つの有効のスレッドがあります。独自に作成したクラスまたはアクティビティは、削除するまで有効です。

## ▼ クラスおよびアクティビティを作成する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メニューで「Classes & Activities」 → 「Create a Job Class」を選択します。
3. ジョブクラスの作成画面が表示されたら、次のように実行します。
  - a. 作成するクラスの名前を入力し、Return キーを押します。
  - b. 新しいクラスに割り当てるアクティビティの数 (1 ~ 99) を入力し、Return キーを押します。

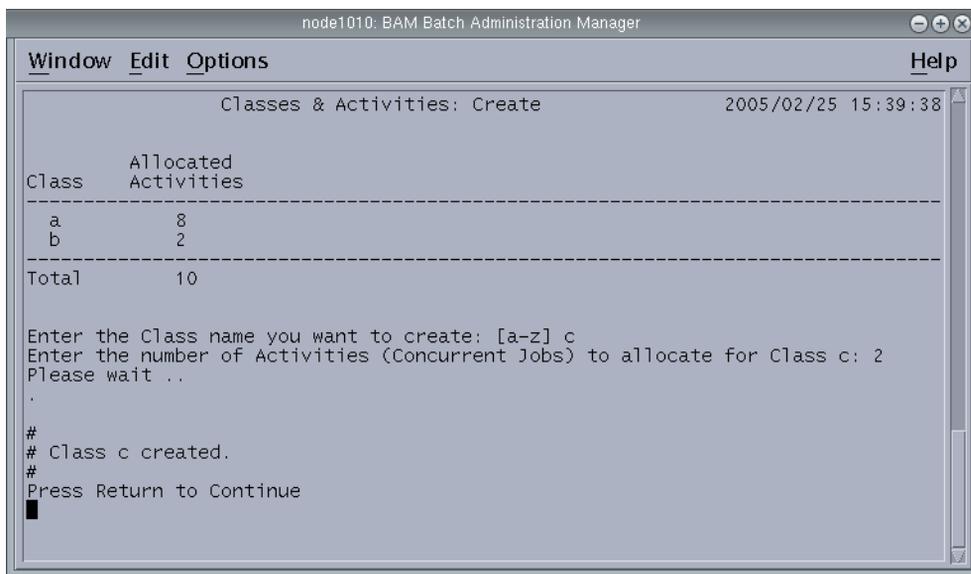


図 10-4 ジョブクラスの作成

4. Return キーを押し、「Classes & Activities」メニューを表示します。

## アクティビティー数の変更

この節では、ノードに許可するアクティビティーの最大数を変更する方法について説明します。この値は、すべてのジョブクラスに定義されているアクティビティーの総数以上である必要があります。この設定値によって、ノードのすべてのサブシステムで並行して実行できるジョブの数が決まります。

### ▼ アクティビティー数を変更する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メニューでオプション「Classes & Activities」→「Maximum Number of Activities (Concurrent Jobs) Allowed」を選択します。

図 10-5 の画面に、アクティビティーの最大数の現在設定されている値が表示されます。

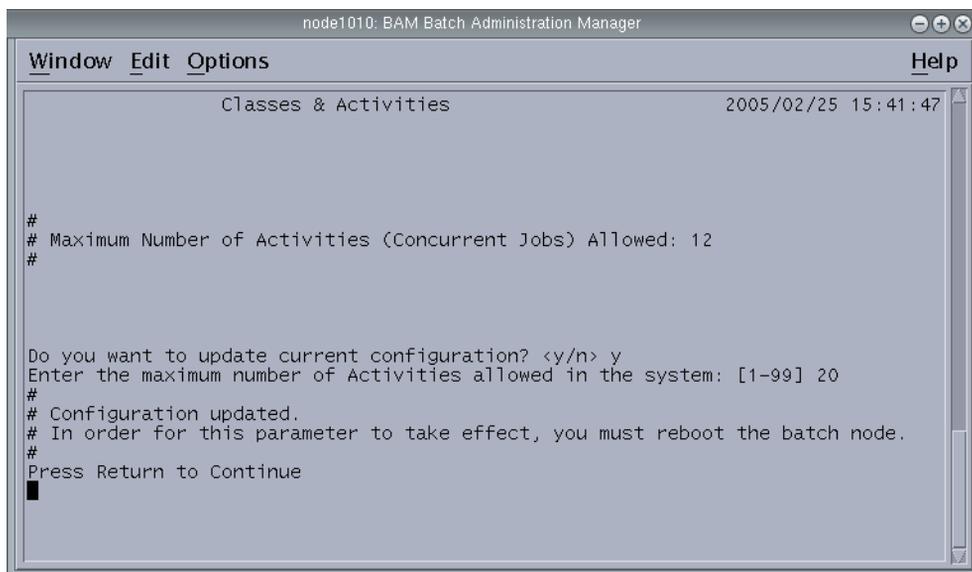


図 10-5 アクティビティー最大数の設定画面

3. 値を変更するには、次のように実行します。
  - a. プロンプトに `y` と入力します。
  - b. 新しいアクティビティー数 (1 ~ 99) を入力します。
  - c. Return キーを押します。設定の更新を確認する画面が表示されます。
4. Return キーを押し、「Classes & Activities」メニューを表示します。

---

注 - 変更を適用するには、ノードを再起動する必要があります。

---

## ジョブクラスの削除

### ▼ ジョブクラスを削除する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。
2. BAM メニューで「Classes & Activities」 → 「Delete a Job Class」を選択します。  
定義されているクラスおよび各クラスに割り当てられているアクティビティーを一覧表示するジョブクラスの削除画面が表示されます。

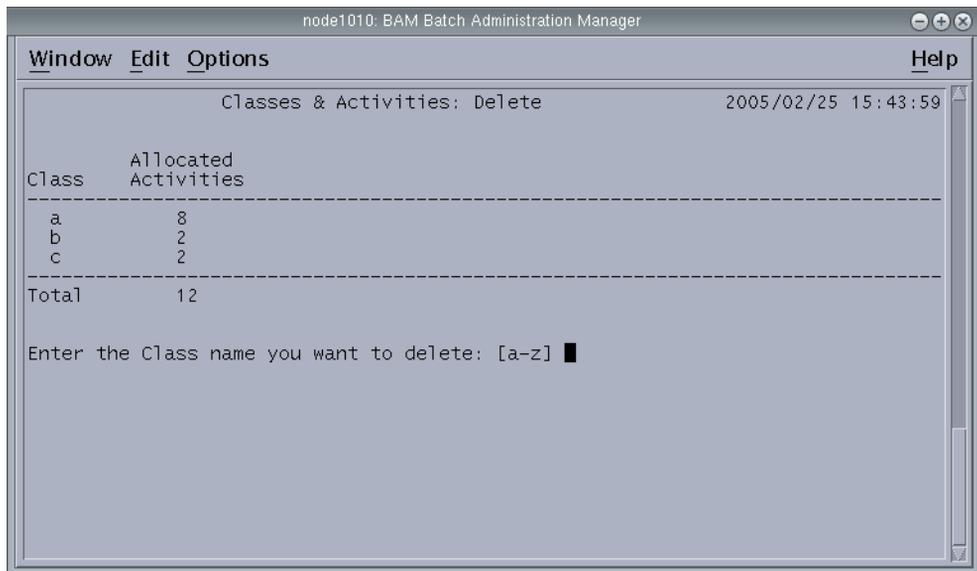


図 10-6 ジョブクラスの削除画面

3. 削除するジョブクラスの文字を入力し、Return キーを押します。

これにより、そのクラスに対してジョブをサブミットできなくなります。削除したクラスにジョブをサブミットするには、そのクラスを再作成する必要があります。

4. Return キーを押し、「Classes & Activities」メニューを表示します。

# 第11章

## Problem Determination

この章では、ログの使用法および Sun MBM システムに関する問題の原因を特定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。

- 136 ページの「エラーログファイル」
- 140 ページの「サブシステムの追跡」
- 142 ページの「プロセスの追跡」
- 143 ページの「メッセージの追跡」
- 145 ページの「共有メモリーの表示」
- 146 ページの「ディスク容量の確認」
- 148 ページの「障害追跡」

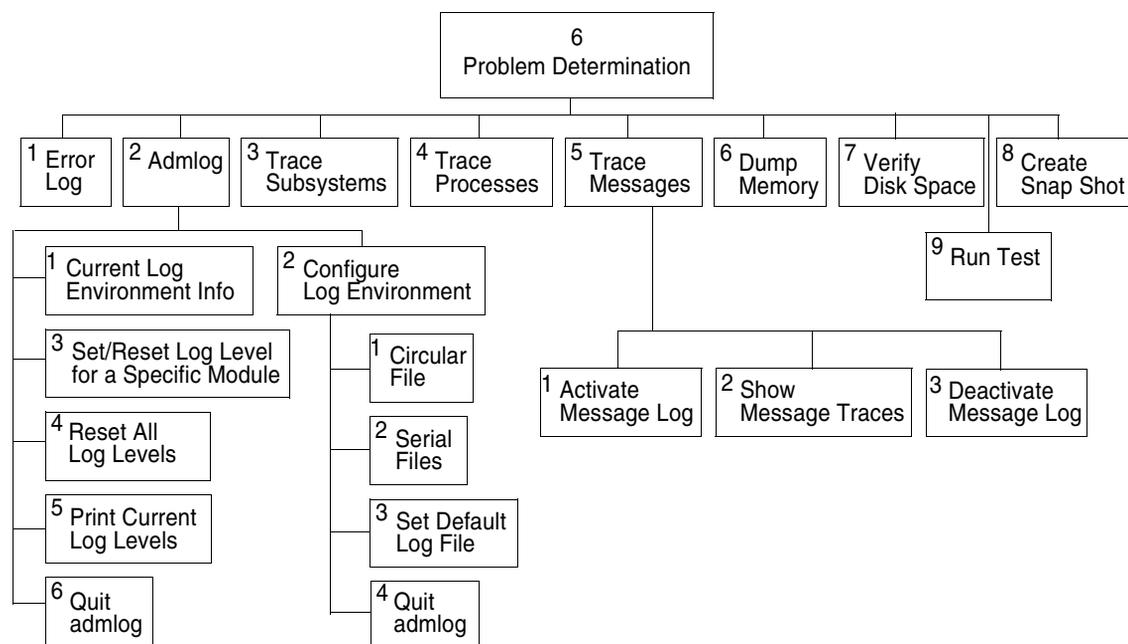


図 11-1 問題特定のメニュー構造

# エラーログファイル

ログファイルには、Sun MBM デーモンおよびバッチジョブを実行する特別なバッチプロセスからのデバッグメッセージが収集されます。デフォルトのログファイルは、図 11-2 に示すように lgdem デーモンによって管理される 32K バイトの循環ファイルです。このログファイルは主としてデバッグに使用されます。

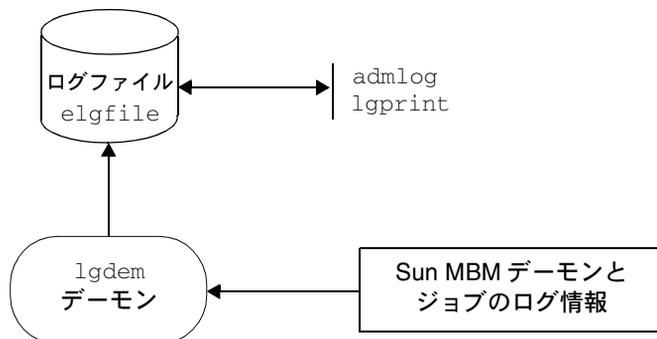


図 11-2 lgdem デーモン

## ▼ エラーログファイルを表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メニューで「Problem Determination」→「Error Log」を選択し、図 11-3 のエラーログ (elgfile) を表示します。

```
node1010: BAM Batch Administration Manager
Window Edit Options Help
Problem Determination: Error Log 2005/03/02 09:21:35
*****
* LOG FILE : elgfile *
*****
common logg
Wed Mar 02 09:18:24 2005 PID=21918 CPU=0
ID EBMD Pid 21918 ( lev2) .setting i=8 fd=8 pid=21929 mbname=psg_daemon.
Wed Mar 02 09:18:24 2005 PID=21918 CPU=0
ID EBMD Pid 21918 ( lev2) .setting i=9 fd=9 pid=21932 mbname=bqm.
Wed Mar 02 09:18:24 2005 PID=21918 CPU=0
ID EBMD Pid 21918 ( lev2) .@ set 10 fds.
--More--(1%)
```

図 11-3 エラーログ (elgfile) 画面

3. ログファイルをナビゲートするには、次のように実行します。
  - Return キーを押すと、1 行ずつ後続の情報が表示されます。
  - スペースバーを 1 回押すと、1 画面ずつ後続の情報が表示されます。
  - ヘルプの表示を終了するには、q キーを押します。
4. すべての画面を表示したあと、あるいは q キーを押したあと、Return キーを押して「Problem Determination」メニューに戻ります。

## ▼ ログファイルのサイズを変更する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メニューでオプション「6」を選択し、「Problem Determination」メニューを表示します。
3. オプション「2 Admlog」を選択し、図 11-4 の「ADMLOG Functionality」メニューを表示します。

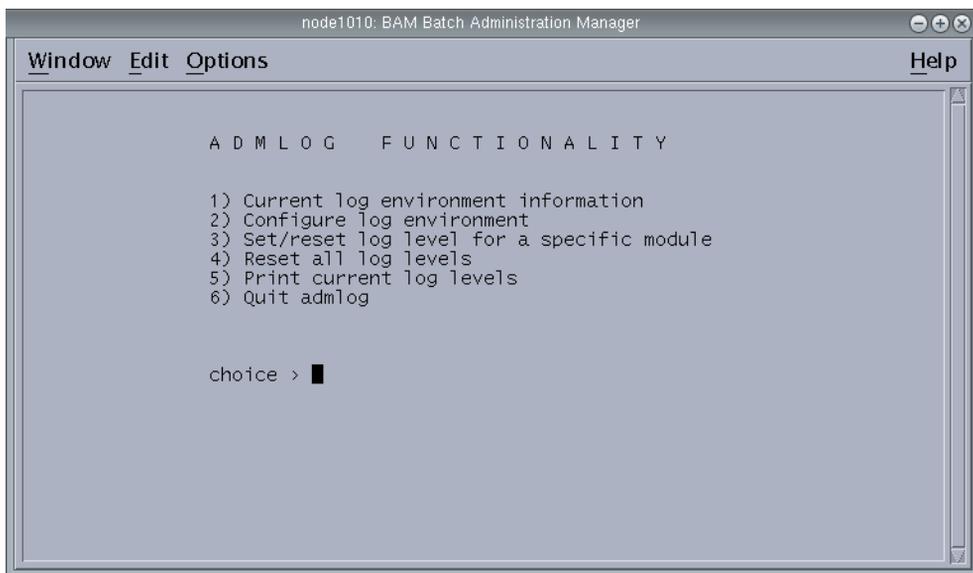


図 11-4 ADMLOG Functionality メニュー

4. オプション「2 Configure Log Environment」を選択し、「Configure Log Environment」メニューを表示します。

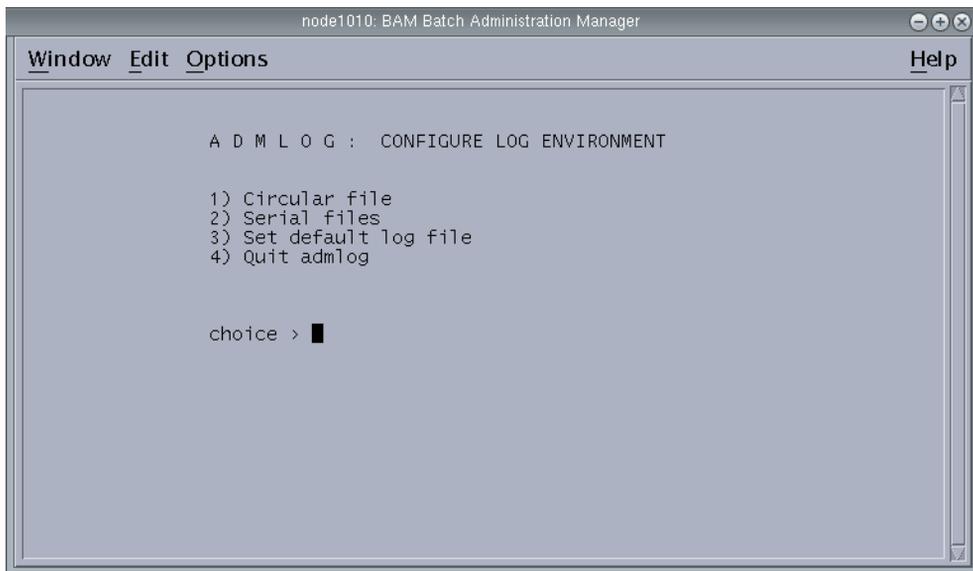


図 11-5 ADMLOG: Configure Log Environment メニュー

5. シリアルファイルを定義する場合は、オプション「2 Serial files」を選択し、次のとおり実行します。

- a. 作成するログファイルの数 (1 ~ 100) を入力し、Return キーを押します。
- b. ファイルの最大サイズを入力し、Return キーを押します。
- c. ログファイルのデフォルトのパス名が表示されたら、Return キーを押してそのパス名を受け入れるか、別のパス名を入力して Return キーを押します。

確認メッセージが表示されます。

複数のシリアルファイルを定義すると、Sun MBM によって最初のファイル名に接尾辞 \_00 が追加され、次のファイル名に \_01 が追加されていきます。すべてのシリアルファイルにログが書き込まれると、Sun MBM は最初のファイル (接尾辞 \_00) から順に上書きし、再びすべてのファイルが上書きされると最初のファイルに戻ります。admlog コマンドの詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。

6. 循環ファイルを定義する場合は、「Configure Log Environment」メニューでオプション「1 Circular file」を選択し、次のとおり実行します。

- a. ファイルの最大サイズを K バイト単位で入力し (デフォルト値は 32K バイト)、Return キーを押します。
- b. デフォルトのファイルパス名が表示されます。別のパスを指定する場合は、該当するファイルパス名を入力してから、Return キーを押します。

定義できる循環ファイルは 1 つだけです。

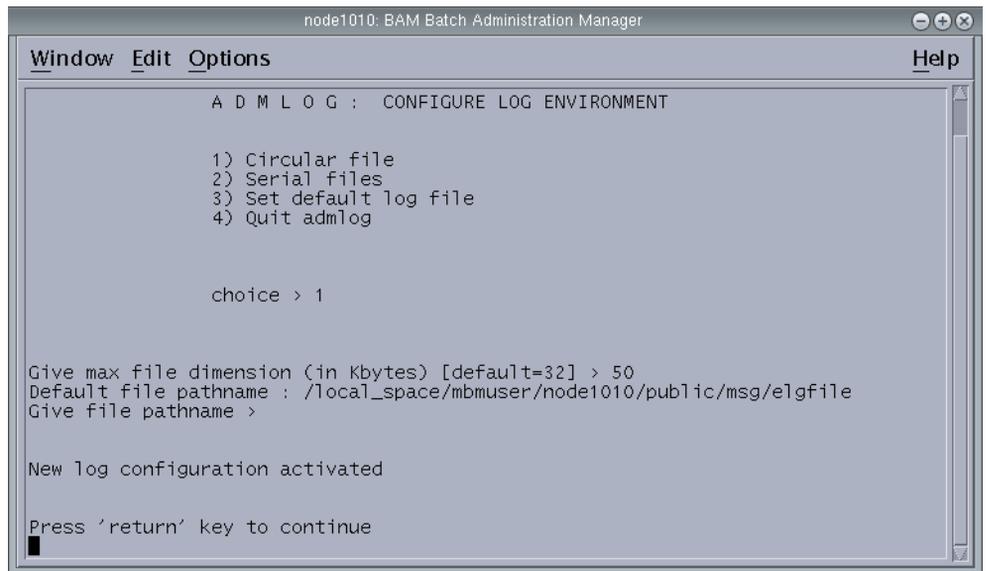


図 11-6 ADMLOG: 循環ファイル設定画面

7. Return キーを押し、「ADMLOG Functionality」メニューを表示します。
8. オプション「6 Quit admlog」を選択し、「Problem Determination」メニューに戻ります。

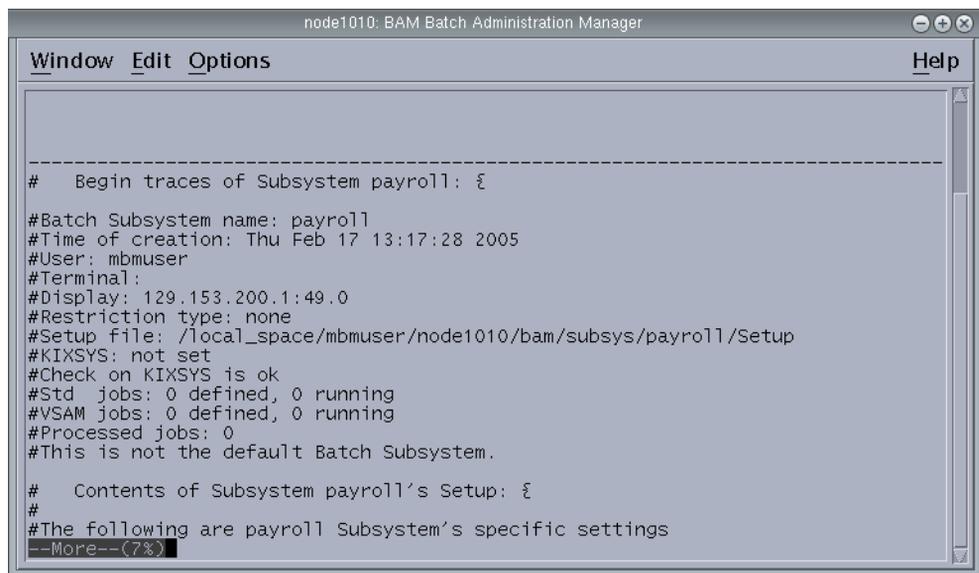
---

## サブシステムの追跡

サブシステムの追跡機能を使用すると、定義されている各サブシステムの設定値と現在の状態を表示できます。これにより、サブシステムが正しく設定されているかを任意に確認できます。さらに、現在実行中のジョブ数や完了したジョブ数も表示できます。

### ▼ サブシステムを追跡する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メニューで「Problem Determination」→「Trace Subsystems」を選択し、サブシステムの追跡画面を表示します。



```
node1010: BAM Batch Administration Manager
Window Edit Options Help
-----
#   Begin traces of Subsystem payroll: {
#Batch Subsystem name: payroll
#Time of creation: Thu Feb 17 13:17:28 2005
#User: mbmuser
#Terminal:
#Display: 129.153.200.1:49.0
#Restriction type: none
#Setup file: /local_space/mbmuser/node1010/bam/subsys/payroll/Setup
#KIXSYS: not set
#Check on KIXSYS is ok
#Std jobs: 0 defined, 0 running
#VSAM jobs: 0 defined, 0 running
#Processed jobs: 0
#This is not the default Batch Subsystem.
#   Contents of Subsystem payroll's Setup: {
#
#The following are payroll Subsystem's specific settings
--More--(7%)
```

図 11-7 サブシステムの追跡画面

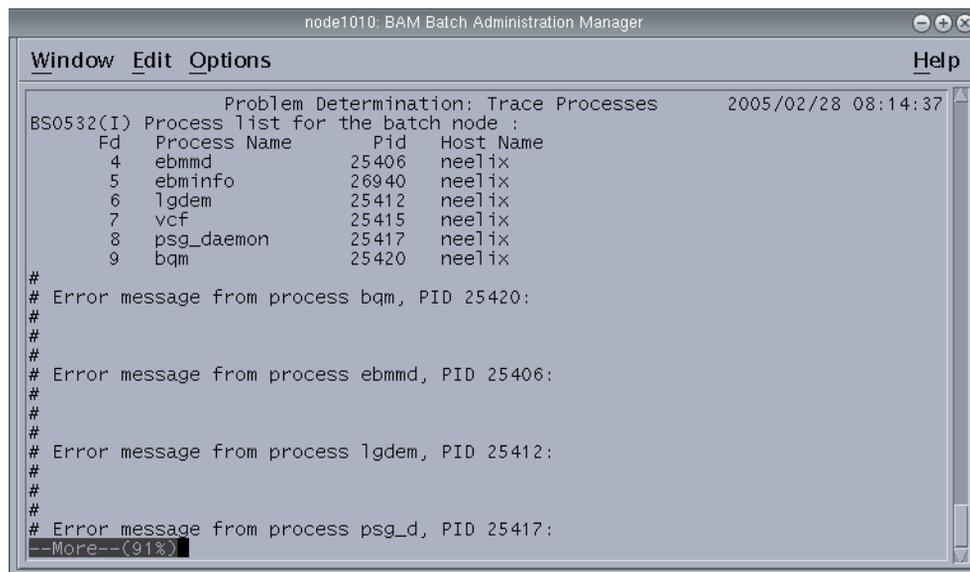
3. 画面をナビゲートするには、次を実行します。
  - Return キーを 1 回押すと、1 行ずつ後続の情報が表示されます。
  - スペースバーを 1 回押すと、1 画面ずつ後続の情報が表示されます。
  - ヘルプの表示を終了するには、q キーを押します。
4. すべての画面を表示したあと、あるいは q キーを押したあと、Return キーを押して「Problem Determination」メニューに戻ります。

# プロセスの追跡

プロセス追跡機能を使用すると、ノードに関するプロセスのリストを表示できます。このリストには、プロセス名、プロセス ID (PID)、ホスト名が含まれ、それらのプロセスにエラーがあった場合はエラーメッセージも書き込まれます。

## ▼ プロセスを追跡する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。ノードが実行されていることを確認します。
2. BAM メニューで「Problem Determination」→「Trace Process」を選択し、次の画面を表示します。



```
node1010: BAM Batch Administration Manager
Window Edit Options Help
Problem Determination: Trace Processes 2005/02/28 08:14:37
BS0532(I) Process list for the batch node :
  Fd  Process Name      Pid  Host Name
  4   ebmmd             25406 neelix
  5   ebminfo          26940 neelix
  6   lgdem            25412 neelix
  7   vcf              25415 neelix
  8   psg_daemon       25417 neelix
  9   bqm              25420 neelix
#
# Error message from process bqm, PID 25420:
#
#
# Error message from process ebmmd, PID 25406:
#
#
# Error message from process lgdem, PID 25412:
#
#
# Error message from process psg_d, PID 25417:
--More--(91%)
```

図 11-8 Trace Processes 画面

3. 画面をナビゲートするには、次を実行します。
  - Return キーを押すと、1 行ずつ後続の情報が表示されます。
  - スペースバーを 1 回押すと、1 画面ずつ後続の情報が表示されます。
  - ヘルプの表示を終了するには、q キーを押します。
4. すべての画面を表示したあと、あるいは q キーを押したあと、Return キーを押して「Problem Determination」メニューに戻ります。

# メッセージの追跡

メッセージ追跡機能を使用すると、次の作業を実行できます。

- メッセージログの有効化
- メッセージ追跡の表示
- メッセージログの無効化

## ▼ メッセージログを有効化する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メニューで「Problem Determination」→「Trace Messages」を選択し、「Trace Messages」メニューを表示します。



図 11-9 Trace Messages メニュー

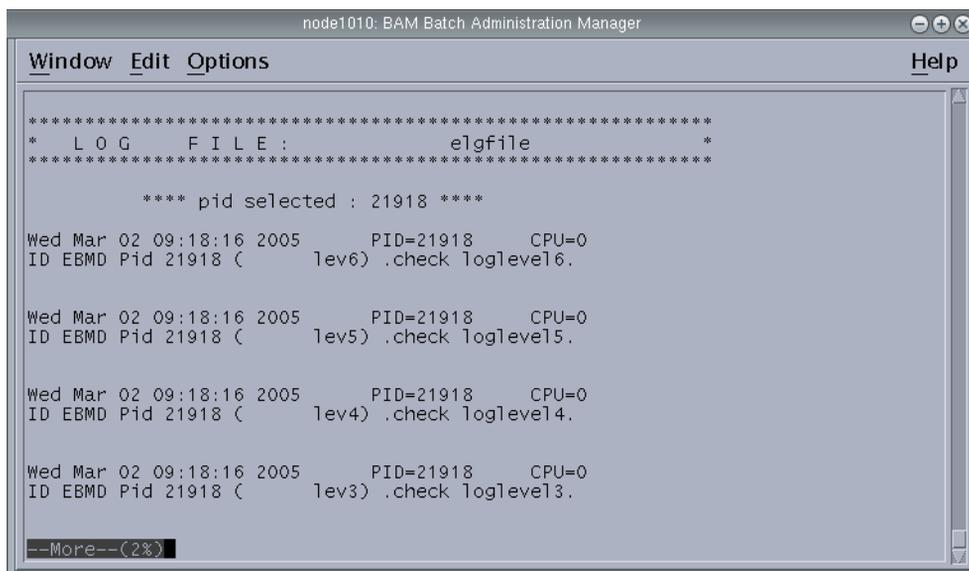
3. オプション「1 Activate Message Log」を選択します。  
画面下部に、「Message Log activated」というメッセージが表示されます。

- Return キーを押すとこのメッセージが消え、「Trace Messages」メニューが再び表示されます。

## ▼ メッセージ追跡を表示する

- 「Trace Messages」メニューでオプション「2 Show Message Traces」を選択します。

次の画面 (図 11-10) は、循環ファイルからの出力例です。



```
node1010: BAM Batch Administration Manager
Window Edit Options Help
*****
* LOG FILE: elgfile *
*****

**** pid selected : 21918 ****

Wed Mar 02 09:18:16 2005 PID=21918 CPU=0
ID EBMD Pid 21918 ( lev6) .check loglevel6.

Wed Mar 02 09:18:16 2005 PID=21918 CPU=0
ID EBMD Pid 21918 ( lev5) .check loglevel5.

Wed Mar 02 09:18:16 2005 PID=21918 CPU=0
ID EBMD Pid 21918 ( lev4) .check loglevel4.

Wed Mar 02 09:18:16 2005 PID=21918 CPU=0
ID EBMD Pid 21918 ( lev3) .check loglevel3.

--More--(2%)
```

図 11-10 メッセージ追跡の画面

- 画面をナビゲートするには、次を実行します。
  - Return キーを押すと、1 行ずつ後続の情報が表示されます。
  - スペースバーを 1 回押すと、1 画面ずつ後続の情報が表示されます。
  - ヘルプの表示を終了するには、q キーを押します。
- すべての画面を表示したあと、あるいは q キーを押したあと、Return キーを押して「Trace Messages」メニューに戻ります。

## ▼ メッセージログを無効化する

1. 「Trace Messages」メニューでオプション「3 Deactivate Message Log」を選択します。  
画面下部に、「Message Log deactivated」というメッセージが表示されます。
2. Return キーを 2 回押すとこのメッセージが消え、「Problem Determination」メインメニューが表示されます。

---

## 共有メモリーの表示

### ▼ Sun MBM の共有メモリーを表示する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メニューで「Problem Determination」→「Dump Memory」を選択し、メモリーダンプを表示します。

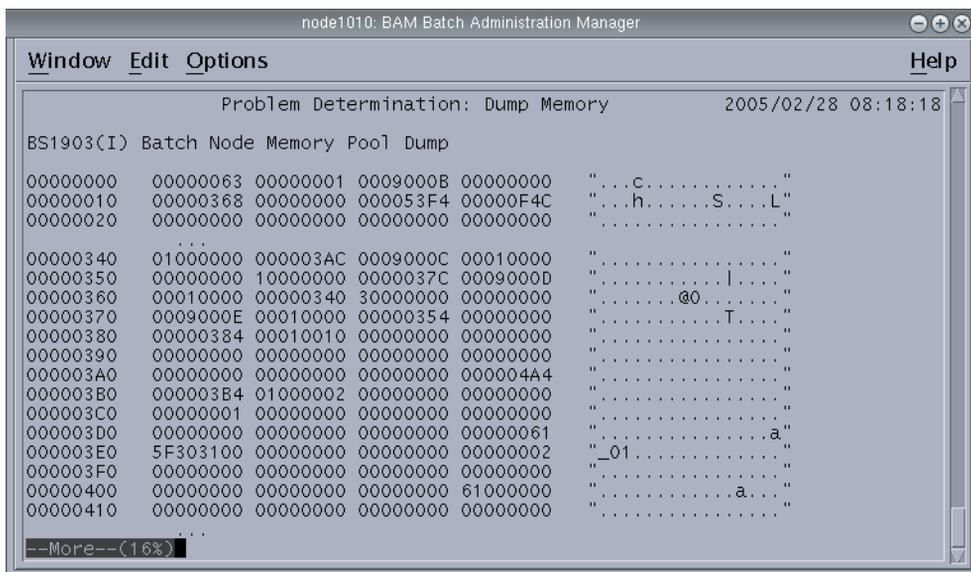


図 11-11 Dump Memory 画面

3. 画面をナビゲートするには、次を実行します。
  - Return キーを押すと、1 行ずつ後続の情報が表示されます。
  - スペースバーを 1 回押すと、1 画面ずつ後続の情報が表示されます。
  - ヘルプの表示を終了するには、q キーを押します。
4. すべての画面を表示したあと、あるいは q キーを押したあと、Return キーを押して「Problem Determination」メニューに戻ります。

## ディスク容量の確認

「Problem Determination」メニューの「Verify Disk Space」オプションを使用すると、次のディレクトリおよびファイルのために使用可能なディスク容量を確認できます。

- ジョブの出力ディレクトリ
- ユーザーの作業用ディレクトリ
- システムの一時ディレクトリ
- Sun MBM システムログファイル

## ▼ BAM メニューによってディスク容量を確認する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メニューで「Problem Determination」→「Verify Disk Space」を選択します。
  - 十分なディスク容量があれば、「Verification Completed Successfully」というメッセージが表示されます。
  - ディレクトリのための容量が不足している場合は、対処方法を含むメッセージが表示されます。
3. 画面をナビゲートするには、次を実行します。
  - Return キーを押すと、1 行ずつ後続の情報が表示されます。
  - スペースバーを 1 回押すと、1 画面ずつ後続の情報が表示されます。
  - ヘルプの表示を終了するには、q キーを押します。
4. すべての画面を表示したあと、あるいは q キーを押したあと、Return キーを押して「Problem Determination」メニューに戻ります。

## ▼ bam コマンドによってディスク容量を確認する

- 次のコマンドを入力します。

```
$ bam diskspace
```

このコマンドによって、Sun MBM ソフトウェアが内部ファイルおよび一時ファイルの領域のために要求したファイルシステムのディスク利用率が 90% 未満であることが確認されます。必要なディレクトリのいずれかの利用率が 90% を超えている場合だけ、警告メッセージが表示されます。それ以外の場合は、次のメッセージが表示されます。

```
Verification Complete
```

このコマンドをシステム管理ルーチンに組み込めば、ディスク容量を定期的に監視できます。

## 障害追跡

この節では、発生した問題を障害追跡するときのシステムスナップショットの作成方法、テストの実行方法、ログファイルの表示方法について説明します。

### ▼ システムのスナップショットを作成する

1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メインメニューで「Problem Determination」→「Create a Snap Shot」を選択します。  
次のようなメッセージが表示されます。

```
Starting Batch Node SnapShot at:Wed Mar 2 09:03:37 MST 2005

Temporary Directory [option -t]:/local_space/mbmuser/temp/batchsnapdir.mbmuser
Output Directory [option -d]:/local_space/mbmuser/temp/batchsnapdir.mbmuser

Creating Batch Node SnapShot File, Please Wait ...
EBMSNAP_GETISH is NOT set, ebmsnap WILL NOT COLLECT ish and ishpc
EBMSNAP_GETHIST is NOT set, ebmsnap WILL NOT COLLECT history files for 2005-03-02
EBMSNAP_TRACEUSER is NOT set, using mbmuser for btshrc.trail

Command successfully completed.
SnapShot File
Name:/local_space/mbmuser/temp/batchsnapdir.mbmuser/batchsnapfile.0302_090341.
tar.Z

Press Return to Continue
```

圧縮 tar ファイルが一時ディレクトリに作成されます。このファイルをご購入先に送信して、障害追跡を依頼できます。

このスナップショットを作成するときに実行される ebmsnap ユーティリティーについては、『Sun Mainframe Batch Manager 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。

3. Return キーを押して、「Problem Determination」メニューを表示します。

## ▼ テストを実行する

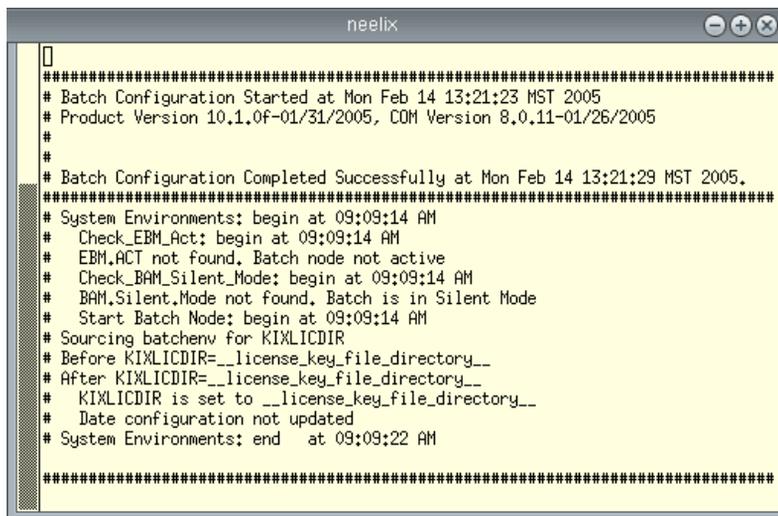
1. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティーを起動します。  
ノードが実行中であることを確認します (14 ページの「システム状態および有効なサブシステムの一覧表示」を参照)。実行中でない場合はノードを起動します (12 ページの「ノードの起動と停止」を参照)。
2. BAM メインメニューで「Problem Determination」→「Run Test」を選択します。  
このテストでは、サブシステムの作成、ジョブの実行、サブシステムの削除が行われます。テストが正常に実行されると、次のメッセージが画面に表示されます。

```
Starting Test ....
Submitting Job JOB1, number 016 ....
BS1079(I) User user21:job 16 ---> Running
BS1079(I) User user21:job 16 ---> Finished
Submitting Job JOB2, number 017 ....
Submitting Job JOB3, number 018 ....
    17    18
    Run  Run
    Abrt Run
    Abrt Abrt
Submitting Job JOB4, number 019 ....
BS1079(I) User user21:job 19 ---> Running
BS1079(I) User user21:job 19 ---> Finished
Test terminated
```

3. Return キーを押して、「Problem Determination」メニューに戻ります。  
テストの実行に失敗した場合、ノードが正しくインストールされているかどうかを確認します。確認しても問題が発生する場合は、ご購入先に連絡してください。

## BAM ログファイル

BAM が表示するすべての情報は、BAM ログファイルに格納されます。BAM を終了すると、bam.log ファイルのパスを示すメッセージが表示されます。BAM を終了しても解消しない問題がある場合は、問題解析のためにこの BAM ログファイルのコピーをご購入先へ送信してください。

A screenshot of a terminal window titled 'neelix'. The window contains a log file with the following text:

```
#####  
# Batch Configuration Started at Mon Feb 14 13:21:23 MST 2005  
# Product Version 10.1.0f-01/31/2005, COM Version 8.0.11-01/26/2005  
#  
#  
# Batch Configuration Completed Successfully at Mon Feb 14 13:21:29 MST 2005.  
#####  
# System Environments: begin at 09:09:14 AM  
#   Check_EBM_Act: begin at 09:09:14 AM  
#   EBM,ACT not found. Batch node not active  
#   Check_BAM_Silent_Mode: begin at 09:09:14 AM  
#   BAM,Silent,Mode not found. Batch is in Silent Mode  
#   Start Batch Node: begin at 09:09:14 AM  
#   Sourcing batchenv for KIXLICDIR  
#   Before KIXLICDIR=__license_key_file_directory__  
#   After KIXLICDIR=__license_key_file_directory__  
#   KIXLICDIR is set to __license_key_file_directory__  
#   Date configuration not updated  
# System Environments: end   at 09:09:22 AM  
#####
```

図 11-12 BAM ログファイルの例

# 付録 A

## ジョブの実行環境の設定

この付録では、Sun MBM 設定ファイルによってジョブの実行環境を制御する仕組みを説明します。

### 設定ファイルの処理順序

バッチジョブを実行すると、ジョブは次の表に示す順序で環境変数を継承します。

これらの設定ファイルはいずれも C シェル形式で記述する必要があります。



**注意** – このとき、ジョブ実行設定ファイルにジョブの実行フローを変更するシェルコマンド (cd /tmp/myfiles など) を含めないでください。ジョブ実行設定ファイルには、\$HOME/.btshrc、\$PACK/btshrc、\$SETUP、および \$USER\_SETUP があります。ジョブの実行フローを変更するシェルコマンドをこれらのファイルに含めると、予期しない動作が起きる可能性があります。

表 A-1 Sun MBM ジョブ設定ファイル

処理順序	設定ファイル	使用法	番号	状態
1	\$PACK/btshrc	ノード	各ノードに 1 つ	任意
2	サブシステム設定ファイル (\$SETUP)	サブシステム	各サブシステムに 1 つ	必須
3	ユーザーサブシステム設定ファイル (\$USER_SETUP)	サブシステム	各サブシステムに 1 つ	任意
4	\$HOME/.btshrc	ユーザー	各ユーザーに 1 つ	任意

表 A-1 Sun MBM ジョブ設定ファイル (続き)

処理順序	設定ファイル	使用法	番号	状態
5	\$PUBLIC/bin/pre_process.btsh	ノード	各ノードに 1 つ	任意
	または \$HOME/.pre_process.btsh	ユーザー	各ユーザーに 1 つ	任意
6	\$PUBLIC/bin/post_process.btsh	ノード	各ノードに 1 つ	任意
	または \$HOME/.post_process.btsh	ユーザー	各ユーザーに 1 つ	任意

## \$PACK/btshrc ファイル

\$PACK/btshrc 設定ファイルには、実行前にバッチジョブが継承するノード全体の環境変数を定義できます。

このファイルの環境変数は、特定のサブシステムまたはユーザーに関連付けないようにします。たとえば、\$PACK/btshrc ファイルの PATH 環境変数は、次のように設定します。

```
setenv PATH $PATH:/usr/local/bin
```

---

**注** - 当該ノードに関連付けられたサブシステムにユーザーがサブミットするすべてのジョブは、このファイルに定義された環境を継承します。

---

## \$HOME/.btshrc ファイル

\$HOME/.btshrc ファイルには、ユーザー固有の環境変数を含めることができます。

異なる Sun MBM ノードまたはサブシステムに対して同一ユーザーがサブミットするすべてのジョブは、そのユーザー固有の \$HOME/.btshrc ファイルに定義された環境を継承します。したがって、このファイルに設定する環境変数が特定の Sun MBM ノードまたはサブシステムに関連付けられないようにします。

`$HOME/.btshrc` ファイルに設定できる環境変数は、次のとおりです。

---

<code>SYSDIR</code>	アプリケーションプログラム出力レポート (SYSOUT DD 文) が含まれるディレクトリ。デフォルトの <code>\$SYSDIR</code> が各サブシステムの設定ファイルに定義されます。 <code>\$HOME/.btshrc</code> ファイルに定義した場合は、その値によってサブシステムの設定が上書きされます。
<code>FORMS</code>	アプリケーションプログラム出力レポートが書き込まれる代替ディレクトリを指定する <code>SYSDIR JCL</code> 文の <code>F</code> オプション。 <code>\$SYSDIR</code> と同様に、デフォルトの <code>\$FORMS</code> が各 Sun MBM サブシステムの設定ファイルに定義されます。 <code>\$HOME/.btshrc</code> ファイルに定義した場合は、その値によってサブシステムの設定が上書きされます。
<code>HISTORY</code>	コマンド <code>subjob</code> または <code>unikixjob</code> のオプション <code>-o</code> によって使用されるディレクトリ。これらのコマンドの詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。

---

## プリプロセスファイル

プリプロセスファイルは、ジョブの実行前に処理されます。このファイルを指定するには、コマンド `subjob` または `unikixjob` の `-x` のオプションを使用します。

ノード全体のプリプロセスファイル `pre_process.btsh` は、`$PUBLIC/bin` ディレクトリに常駐させることができます。

ユーザーのプリプロセスファイル `$HOME/.pre_process.btsh` も作成できます。シェルスクリプト `$HOME/.pre_process.btsh` が存在する場合は、それが実行されます。存在しない場合は、シェルスクリプト `$PUBLIC/bin/pre_process.btsh` があればそれが実行されます。詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』で、Sun MBM `subjob` および `unikixjob` コマンドに関する説明を参照してください。

## ポストプロセスファイル

ポストプロセスファイルは、ジョブの実行後に処理されます。このファイルを指定するには、コマンド `subjob` または `unikixjob` の `-x` のオプションを使用します。

ノード全体のポストプロセスファイル `post_process.btsh` は、`$PUBLIC/bin` ディレクトリに常駐させることができます。

ユーザーのポストプロセスファイル `$HOME/.post_process.btsh` も作成できます。シェルスクリプト `$HOME/.post_process.btsh` が存在する場合は、それが実行されます。これが存在しない場合は、シェルスクリプト `$PUBLIC/bin/post_process.btsh` があればそれが実行されます。詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』で、`subjob` および `unikixjob` コマンドに関する説明を参照してください。

---

## 環境変数

この節では、FMROOT 環境変数のリセット方法、およびジョブの実行時にリセットできない環境変数について説明します。

### \$FMROOT のリセット

FMROOT 環境変数を別の環境変数値にリセットするには、\$USER\_SETUP ファイルで次の構文を使用します。

```
setenv FMROOT "\\\${ENVVAR}"
```

\$FMROOT は妥当性検査時にだけ使用されます。

### リセットできない環境変数

次に示す環境変数は、ジョブの実行時にリセットできません。

環境変数	説明	アクション
RUNBPATH	サブシステム設定ファイルで設定します。	サブシステムを削除および再作成してリセットすることのみが可能です。
FILEMAP	サブシステム File_Map のディレクトリ。サブシステム設定ファイルで設定します。	サブシステムを削除および再作成してリセットすることのみが可能です。
KIXSYS	Sun MTP 領域のディレクトリ。サブシステム設定ファイルで設定します。	BAM を使用して \$KIXSYS をリセットできます。

# オンライントランザクションによる バッチジョブのサブミット

Sun MTP では、オンライントランザクションからバッチジョブをサブミットする場合に `kixjob` スクリプトを使用します。このスクリプトは、`$UNIKIX/bin` ディレクトリにあります。

コード例 A-1 に示す `kixjob` スクリプトを確認し、使用する環境に応じて修正します。

`EBMHOME` 環境変数を設定した場合、`kixjob` は Sun MBM に対してバッチファイルのサブミットを試みます。デフォルトでは、`kixjob` による処理にはバッチシェルスクリプト形式のバッチファイルが想定されています。Sun MBM バッチシェルスクリプトの作成および実行方法については、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。このスクリプトにより、バッチファイルが `$EBMHOME/public/ish` にコピーされ、`unikixjob` コマンドを使用してデフォルトのサブシステムに対してそのファイルがサブミットされます。

トランザクションによって実行されるアプリケーションプログラムは、JOB というタイプの Sun MTP 一時データ (TD) キューにエントリを書き込むことによってバッチジョブをサブミットします。TD キューは、領域の宛先管理テーブル (DCT) に定義されます。トランザクションによって同期点が発行された場合、またはトランザクションが完了した場合は、Sun MTP によって `$KIXSYS` ディレクトリ内のファイルに TD キューが書き込まれます。

ファイル名の形式	パラメータ	説明
<code>xxxxnnnn.job</code>	<code>xxxx</code>	TD キュー名
	<code>nnnn</code>	一意の連続整数

バッチスクリプトまたは MVS JCL ジョブカードが書き込まれている TD キューから、バッチファイルを作成できます。JCL ストリームバッチファイルは、Sun MBM で実行する前にバッチスクリプトに変換する必要があります。そのためには、ジョブをサブミットするための Sun MBM コマンド (`subjob` または `unikixjob`) に、オプション `-t j` を指定します。

`kixjob` は、Sun MBM に JCL ジョブのサブミットをするとき、最初に、バッチファイルの移動をデフォルトのバッチジョブスクリプト用の `$EBMHOME/public/ish` ではなく `$EBMHOME/public/jmvs` に行います。ユーザーは、バッチファイルが `$EBMHOME/public` の正しいディレクトリに移動されるように `kixjob` を修正するとともに、ジョブのサブミットに使用するこの Sun MBM コマンドに適切なオプションが指定されていることを確認する必要があります。

Sun MTP でのバッチファイルの実行時に、kixjob はファイルのモードを変更し、\$KIXSYS に置かれているファイルを KIXBTCH 環境変数に指定されている最初のディレクトリに移動します。領域を起動したユーザーの環境変数に \$KIXBTCH が設定されていない場合、ジョブファイルは \$KIXSYS に置かれたままとなり実行されません。

JCL 変換エラーを解消し、新しい File\_Map エントリをカスタマイズするために、kixjob による実行の前に、Sun MTP システムの外部で JCL ストリームを変換しておくことをお勧めします。これにより kixjob では、\$PUBLIC/ish 下で JCL ストリームをバッチシェルスクリプトに変換でき、実行時の構文エラーもなくなります。そのあと、Sun MBM に対するバッチジョブのサブミットに進むことができます。さらにユーザーは、\$KIXSYS/unikixmain.err ファイルを調べて、ジョブのサブミットに使用した Sun MBM コマンドによってレポートされたエラーがないか確認してください。

---

**注** – kixjob を使って Sun MBM に対してバッチファイルをサブミットする場合は、batchenv ファイルに指定されている Sun MBM 環境変数および FILEMAP 環境変数が設定されていることを、Sun MTP 領域の起動前に確認してください。プロシージャを実行する JCL ストリームを変換する場合は、PROCLIB 環境変数の設定も必要です。

---

コード例 A-1 kixjob シェルスクリプト ( 1 / 4)

```
*****
# usage:kixjob [file ...]
# formats file and sends it to KIXBTCH; removes file
# file(s) to move (optional); default stdin
FILE=
# formatter (default cat)
FORMATTER=cat

# collect arguments
set -- `getopt p: $*`
# while options to process
while test "$1" != "--"
do
    case $1 in
        -p)
            PRINTID=$2
            shift
            ;;
        *)
            echo "Invalid Parm $2"
            exit 1
            # illegal option encountered
            ;;
    esac
    shift
done
# skip --
shift

# all arguments are files to go to batch
FILE="$*"

# if no file given, assume stdin
if test -z "$FILE"
then
    FILE="-"
    RMCMD=
else
    # must remove file after printing
    RMCMD="/bin/rm -f $FILE"
fi
```

コード例 A-1 kixjob シェルスクリプト ( 2 / 4 )

```
if [ "A$EBMHOME" = "A" ]
then
##
## JOB described by $FILE will be submitted to Sun MTP Standard Batch
##
  if test -z "$KIXBTCH"
  then
    echo "KIXBTCH is not set. Job $FILE left in KIXSYS" >&2
    exit 0
  fi
#
# The shell will not be executed unless its mode is executable
#
  chmod 755 $FILE
#
# If KIXBTCH has multiple directories, strip off all but the first.
# If KIXBTCH contains a time, strip it off also.
#
  BTCHDIR=`echo "$KIXBTCH" | cut -f1 -d":" | cut -f1 -d"["`
#
# Move the file to the batch directory for execution
  mv $FILE $BTCHDIR
  retstatus=$?
  export retstatus
else
##
## Because $EBMHOME is set, JOB described by $FILE will be
## translated from JCL to btsh script, if necessary, and
## submitted to Sun MBM.
##
  chmod 755 $FILE

## Extract the first eight characters of $FILE as the jobname
## and assign them to $JOB.

  JOB=`echo $FILE | cut -f1 -d'.'`
  export JOB

##
## Move file $FILE to $EBMHOME/public/jmvs/$JOB
## or to $EBMHOME/public/ish/$JOB depending on
## whether the job has to be translated or not.
## Use one or the other move command.
##
```

コード例 A-1      kixjob シェルスクリプト ( 3 / 4 )

```
#
# For translation and execution
#
# mv $FILE $EBMHOME/public/jmvs/$JOB
#
# For execution only
#
mv $FILE $EBMHOME/public/ish/$JOB
#
##
## Change directory to $EBMHOME/public so that the Sun MBM command
## used to submit the job may locate the job.
##
cd $EBMHOME/public
##
##
## Execute either subjob or unikixjob using one of the following formats
## or the unikixjob command coded below :
##
## subjob $JOB --> Submit Sun MBM btsh script $JOB using the
## execution environment described by the
## default Sun MBM subsystem.
##
## subjob $JOB -k subsysn --> Submit Sun MBM btsh job script using the JOB
## execution environment described by Sun MBM
## subsystem subsysn.
##
## subjob $JOB -t j --> Translate MVS JCL job stream using the
## Sun MBM JCL converter mvstrans, creating the
## translated btsh job script under
## $EBMHOME/public/ish. Following successful
## translation, the job is submitted to Sun MBM
## for execution in the job execution
## environment described by the default Sun MBM
## subsystem.
##
## subjob $JOB -t j -k subsysn
##
## unikixjob $JOB --> Submit Sun MBM btsh job script, that
## accesses VSAM datasets, using the
## JOB execution environment described
## by the default Sun MBM subsystem.
##
```

コード例 A-1 kixjob シェルスクリプト ( 4 / 4 )

```
## unikixjob $JOB -k subsysn --> Submit Sun MBM btsh job script, that
##     accesses VSAM datasets, using the
##     JOB execution environment described
##     by Sun MBM subsystem subsysn.
##
## unikixjob $JOB -t j --> Translate MVS JCL job stream that
##     accesses VSAM datasets, using the Sun MBM
##     JCL converter mvstrans to create the
##     translated btsh job script under
##     $EBMHOME/public/ish. Following
##     successful translation, the job is
##     submitted to Sun MBM for execution in the
##     job execution environment described by
##     the default Sun MBM subsystem.
##
## unikixjob $JOB -t j -k subsysn
##
## Note:  If you use the -t j option with subjob or unikixjob you must
##         ensure that $FILEMAP and $PROCLIB from the Sun MTP startup
##         environment is the same as that required for the Sun MBM execution
##         environment.
##
##
unikixjob $JOB
retstatus=$?
export retstatus
fi
exit $retstatus
```

kixjob スクリプトを使用する代わりに、JCL をファイル (次の例では testish) に変換し、CICS アプリケーションから変換済みの JCL に対して unikixjob コマンドを実行できます。この例は、unikixjob コマンドを使用してジョブ testish をサブミットする COBOL プログラムです。

#### コード例 A-2 unikixjob コマンドによるジョブのサブミット

```
IDENTIFICATION DIVISION
PROGRAM-ID.          SUBONL.
AUTHOR.              ABCDE

ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
DATA DIVISION.
WORKING-STORAGE SECTION.
COPY DFHAID.
01    UNIX-COMMANDA          PIC X.
01    UNIX-COMMAND.
      05    UNIKIXJOB        PIC X(51)
      VALUE "unikixjob testish -k ascom -W /porting10/ascom/data".
      05    TERMINATOR      PIC X          VALUE LOW-VALUES.
01    JOB-DONE-LIT          PIC X(13)      VALUE "JOB SUBMITTED".
LINKAGE SECTION.
PROCEDURE DIVISION.
      CALL "SYSTEM" USING UNIX-COMMAND.
      EXEC CICS SEND TEXT    FROM(JOB-DONE-LIT)
                                LENGTH(13)
                                FREEKB

      END-EXEC
      EXEC CICS RETURN
      END-EXEC.
```

---

# 遠隔ジョブのサブミット

Sun MBM は、メインフレームまたはパーソナルコンピュータから ftp コマンドを使用してジョブをサブミットできる機能を備えています。

## 遠隔ジョブサブミット機能を有効化する

1. Sun MBM 管理者としてログインします。
2. ノードが実行中でないことを確認します。
3. `$EBMHOME` 下に `autosub` というディレクトリを作成し、適切なアクセス権を設定します。次に例を示します。

```
$ mkdir $EBMHOME/autosub
$ chmod 775 $EBMHOME/autosub
```

4. 遠隔ジョブの実行が可能なすべてのサブシステム用のサブディレクトリを作成します。次に例を示します。  
現在、ノードに 4 つのサブシステム (`prod1`、`prod2`、`test1`、`test2`) が定義されており、そのうち `prod2` および `test2` だけが遠隔ジョブを実行するとします。
  - a. 次のようにサブディレクトリを作成します。

```
$ mkdir $EBMHOME/autosub/prod2
$ mkdir $EBMHOME/autosub/test2
```

- b. 次のようにアクセス権を設定します。

```
$ chmod 775 $EBMHOME/autosub/prod2
$ chmod 775 $EBMHOME/autosub/test2
```

5. サブシステムにそれぞれの宛先ディレクトリを作成します。次に例を示します。

a. JCL ジョブ用の宛先ディレクトリを次のように作成します。

■ MVS JCL を使用する場合

```
$ mkdir $EBMHOME/autosub/prod2/jmvs  
$ mkdir $EBMHOME/autosub/test2/jmvs
```

■ VSE JCL を使用する場合

```
$ mkdir $EBMHOME/autosub/prod2/jdos  
$ mkdir $EBMHOME/autosub/test2/jdos
```

b. Sun MBM マクロジョブ用、および変換済みの MVS または VSE JCL ジョブ用の宛先ディレクトリを次のように作成します。

```
$ mkdir $EBMHOME/autosub/prod2/ish  
$ mkdir $EBMHOME/autosub/test2/ish
```

c. エラーの JCL、マクロ、およびログが書き込まれる宛先ディレクトリを次のように作成します。

```
$ mkdir $EBMHOME/autosub/prod2/error  
$ mkdir $EBMHOME/autosub/test2/error
```

d. すべての宛先ディレクトリに適切なアクセス権を次のように設定します。

```
$ chmod 775 $EBMHOME/autosub/*/*
```

6. ノードを起動します。

12 ページの「ノードの起動と停止」を参照してください。

## ftp コマンドによる遠隔ジョブのサブミット

ノードが遠隔ジョブのサブミットを受信できるように設定すると、ノードの起動時に autosub デーモンが自動的に起動します。このデーモンは、各サブシステムのディレクトリ `$EBMHOME/autosub/subsystem-name` に作成された宛先ディレクトリ `jmvs`、`jdos`、および `ish` を 60 秒ごとに検索します。

- `jmvs` 宛先ディレクトリでジョブが検出されると、autosub デーモンは MVS JCL トランスレータ `mvstrans` を使用してそのジョブを変換し、該当のサブシステムにサブミットします。
- `jdos` 宛先ディレクトリでジョブが検出されると、autosub デーモンは VSE JCL トランスレータ `dostrans` を使用してそのジョブを変換し、該当のサブシステムにサブミットします。
- `ish` 宛先ディレクトリでジョブが検出されると、autosub デーモンはそのジョブを Sun MBM マクロジョブとみなし、JCL 変換を行わないで該当のサブシステムにサブミットします。

ジョブの実行はそのジョブの所有者として行うため、ジョブファイルの所有者には有効なユーザー ID が必要です。

---

**注** – スーパーユーザー所有のジョブは、セキュリティ上の理由で実行されません。

---

宛先ディレクトリ `jmvs`、`jdos`、および `ish` に対してサブミットするジョブの名前には、次の文字は使用できません。

. ) (

したがって、メインフレームから `ftp put` コマンドを使用する場合に、メインフレームでの元のジョブ名が完全修飾名であれば、もう 1 つのパラメータを指定する必要があります。

`ftp` コマンドによってノードに転送されるファイルはすべて、標準の ASCII 形式である必要があります。プラットフォーム間でのファイル転送時に検出されることのある `Control-Z` と `Control-M` が削除される以外に、文字変換は行われません。

ジョブの変換および実行に必要なファイルは、ノードがあるシステムで使用可能である必要があります。このようなファイルには、アプリケーションプログラム、プロシージャ、カタログ `SYSIN` ファイル、データファイル、`File_Map` などがあります。

サブミットされたジョブは、宛先ディレクトリからリリースされます。ジョブの変換またはサブミット時にエラーが発生すると、autosub デーモンによって対応するエラーディレクトリにエラーファイルが作成されます。

---

注 – ファイルシステムのオーバーフローを防ぐには、このエラーディレクトリに保存されるファイルの数とサイズを監視する必要があります。詳細は、146 ページの「ディスク容量の確認」を参照してください。

---

## mvstrans または dostrans のオプションの指定

遠隔サブミット機能を使用するときに JCL トランスレータのオプションの設定が必要な場合は、ノードを起動する前に、batchenv ファイルで次の環境変数を設定しエクスポートする必要があります。

- MVS JCL トランスレータ mvstrans の場合は、MVSTRANS\_OPT
- VSE JCL トランスレータ dostrans の場合は、DOSTRANS\_OPT

## 遠隔ログの有効化および有効化解除

### ▼ ログを有効化する

1. Sun MBM 管理者としてログインします。
2. ノードが実行中でないことを確認します。
3. autosub ディレクトリにファイルを作成し、適切なアクセス権を設定します。次に例を示します。

```
$ touch $EBMHOME/autosub/ebmautosub.log
$ chmod 666 $EBMHOME/autosub/ebmautosub.log
```

4. ノードを再起動します。

### ▼ ログを有効化解除する

1. Sun MBM 管理者としてログインします。
2. ノードが実行中でないことを確認します。
3. ログファイルを次のように削除します。

```
$ rm $EBMHOME/autosub/ebmautosub.log
```

4. ノードを再起動します。



# 付録 B

## .install ファイル

.install ファイルは、Sun MBM のグローバル構成ファイルで、インストールされたノードおよびそのノード内のサブシステムに関する情報が含まれます。

.install ファイルは、ノードをインストールディレクトリにインストールするとき  
に作成されます。ノードを起動するたびに .install ファイルが読み取られます。  
コード例 B-1 に、このファイルの各項目の形式が示されています。



**注意** - .install ファイルは、ここでは参照のためにだけ含まれます。このファイル  
を壊れないようにするには、直接編集する代わりに BAM を使用して更新します。

## .install ファイルの内容

.install ファイルには、次の定義が保存されます。

表 B-1 .install ファイルの内容 (1 / 7)

項目	説明
pack	Sun MBM コマンドを含むディレクトリ。次のように定義されます。 <i>node-installed-dir</i> /pack 詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照し てください。
public	Sun MBM ユーティリティを含むディレクトリ。次のように定義されます。 <i>node-installed-dir</i> /public 詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照し てください。

表 B-1 .install ファイルの内容 (2 / 7)

項目	説明
sysindir	<p>ノードに対してサブミットされるジョブを含むディレクトリ。次のように定義されます。</p> <p><code>node-installed-dir/sysindir</code></p> <p>詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。</p>
Sun MBM サブシステム	<p>1 つ以上のサブシステムの名前および関連付けられている設定ファイルのリスト。詳細は、66 ページの「使用可能なサブシステムの一覧表示」を参照してください。</p>
デフォルトの履歴ディレクトリ	<p>実行されたジョブの履歴ファイル (ジョブ出カ一覧ファイル) が Sun MBM によって作成されるディレクトリ。このデフォルトの履歴ディレクトリは、ジョブの実行時にコマンド <code>subjob</code> または <code>unikixjob</code> にオプション <code>-h</code> を指定することにより上書きできます。これらのコマンドの詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。また、次の項目「代替履歴ディレクトリのリスト」も参照してください。</p> <p>デフォルトの履歴ファイルが設定されていない場合、システムのデフォルトは <code>/tmp</code> です。</p> <p>ファイルシステムのオーバーフローを防ぐには、デフォルトの履歴ディレクトリに保存されるファイルの数およびサイズを監視する必要があります。</p>
代替履歴ディレクトリのリスト	<p>Sun MBM がジョブの履歴ファイル (ジョブ出カ一覧ファイル) をリダイレクトできる代替ディレクトリ。各代替ディレクトリは、コマンド <code>subjob</code> または <code>unikixjob</code> のオプション <code>-h</code> で使用される名前で識別されます。これにより、ジョブ履歴ファイルはデフォルトの履歴ディレクトリではなく、その名前に関連付けられたディレクトリにリダイレクトされます。これらのコマンドの詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。</p> <p>ファイルシステムのオーバーフローを防ぐには、この代替履歴ディレクトリに保存されるファイルの数およびサイズを監視する必要があります。</p>

表 B-1 .install ファイルの内容 (3 / 7)

項目	説明
許可ファイル	<p>Sun MBM コマンドへのアクセスを制御するセキュリティーファイル。許可ファイルには、すべてのユーザーの読み取り専用アクセス権が必要です。</p> <p><code>bam_authority_file</code></p> <p>BAM を実行できる Sun MBM 管理者を定義します。デフォルトでは、Sun MBM ノードの所有者だけが管理操作を実行できます。</p> <p><code>batch_start_authority_file</code></p> <p>当該ノードを起動できるユーザーを定義します。</p> <p><code>batch_shut_authority_file</code></p> <p>当該ノードをシャットダウンできるユーザーを定義します。</p> <p><code>crtact_authority_file</code></p> <p>当該ノードに関するアクティビティーを作成できるユーザーを定義します。</p> <p><code>dltact_authority_file</code></p> <p>当該ノードに関するアクティビティーを削除できるユーザーを定義します。</p> <p><code>ebmsys_authority_file</code></p> <p>当該ノード内のサブシステムを作成または削除できるユーザーを定義します。</p> <p>許可ファイルの内容の形式は、次のとおりです。</p> <pre> user_name user_name2 user_name3 . . </pre> <p>ファイルにユーザー名が指定されているユーザーは、その許可ファイルが関連付けられているコマンドを使用できます。</p> <p>あるコマンドの許可ファイルが .install ファイル内に指定されていない場合、そのコマンドはすべてのユーザーが実行できます。</p>
作業用ディレクトリ	<p>ジョブによって一時ファイルが作成されるユーザー固有のディレクトリ。詳細は、119 ページの「ユーザーの作業用ディレクトリの管理」を参照してください。</p> <p>形式:</p> <pre> wk_user-name=directory-path </pre> <p>説明</p> <p><code>user-name</code> : ユーザー名。</p> <p><code>directory-path</code> : <code>user-name</code> の一時ファイルが作成されるディレクトリの絶対パス名。</p> <p>Sun MBM 作業用ディレクトリが設定されていない場合、一時ファイルはそのユーザーのホームディレクトリ下に作成されます。</p>
ローカルおよび遠隔ノード	<p>Sun MBM ノード間の通信に必要です。詳細は、46 ページの「複数のノードの設定」を参照してください。</p>
アクティビティーの最大数	<p>当該ノードに作成できるアクティビティーの最大数。デフォルトは 12 です。</p>

表 B-1 .install ファイルの内容 (4 / 7)

項目	説明
HistoryFile の最大サイズ	<p>Sun MBM の HistoryFile 循環ファイルのレコード最大数。デフォルトは 1000 です。</p> <p>このレコード数を削減する場合は、Sun MBM を再起動する前に、既存の \$SYSINDIR/HistoryFile を削除するか別のファイル名で保存します。</p>
Sun MBM コンソールファイル	<p>Sun MBM システムメッセージおよびジョブの開始と終了メッセージを書き込むシリアルファイルを作成するように Sun MBM に指示するフラグ。作成されたファイルは、\$SYSINDIR ディレクトリに ebm_console という名前で作成されます。デフォルトでは作成されません。</p> <p>コンソールファイルについては、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。</p> <p>Sun MBM は、コンソールファイルを常に追加モードで開きます。ファイルのサイズが大きくなる可能性があるため、定期的に削除する必要があります。</p>
Sun MBM コンソール端末	<p>ジョブおよびシステムメッセージを表示する端末装置のパス名。端末 ID を取得するには、メッセージを表示させる端末で tty コマンドを実行します。</p>
Sun MBM コンソールメッセージ接頭辞	<p>Sun MBM コンソールファイルおよび Sun MBM コンソール端末に書き込まれるすべてのメッセージのメッセージ識別子。デフォルト値は、次のとおりです。</p> <p>BatchNode1:</p>
Sun MBM コンソールメッセージレベル	<p>Sun MBM コンソールファイル \$SYSINDIR/ebm_console および Sun MBM コンソール端末に経路指定されるメッセージのタイプ。</p> <p>次のメッセージタイプをいずれか 1 つ選択してください。</p> <p>sysout_only: システムメッセージのみ。</p> <p>job_only: ジョブの開始および終了メッセージのみ。</p> <p>all: システムメッセージとジョブメッセージの両方。デフォルト値です。</p>
HistoryFile メッセージレベル	<p>Sun MBM HistoryFile (\$SYSINDIR/HistoryFile) に経路指定されるメッセージのタイプ。</p> <p>次のメッセージタイプのいずれか 1 つを選択してください。</p> <p>job_only: ジョブの開始および終了メッセージのみ。デフォルト値です。</p> <p>all: ジョブメッセージと Sun MBM システムメッセージの両方。</p> <p>none: すべてのメッセージを抑制します。</p> <p>Sun MBM コンソール端末または Sun MBM コンソールファイル機能を使用する場合は、システムのパフォーマンス向上のために、循環ファイル Sun MBM HistoryFile に書き込まれるメッセージを削減または抑制できます。オプション none を指定すると、Sun MBM コマンド lststs および histprt を実行してもジョブ出力メッセージは表示されません。</p> <p>Sun MBM 管理者は、履歴ファイルに書き込まれたすべてのメッセージを参照できません。その他のユーザーが参照できるのは、各自がサブミットしたジョブに関するジョブメッセージのみです。</p>

表 B-1 .install ファイルの内容 (5 / 7)

項目	説明
Sun MBM 日付形式	<p>実行中のジョブによって出力される日付形式を定義します。</p> <pre>EBM_DATE_FORMAT=<i>format</i></pre> <p>デフォルトでは、日付形式設定指示 %c に基づいて出力されます。次に例を示します。</p> <pre>Start of job job1 at Wed May 04 01:35:10 2005 Start COB1 at Wed May 04 01:35:32 2005 End of COB1 at Wed May 04 01:35:32 2005 Job job1 aborting at Wed May 04 01:35:33 2005</pre> <p>Sun MBM 管理者は別の日付形式を指定することもできます。有効な日付形式設定指示のリストについては、表 4-1 を参照してください。また、『Sun Mainframe Batch Manager ソフトウェア 移行ガイド』および『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』も参照してください。</p> <p>このエントリを変更するたびに、Sun MBM を再起動する必要があります。</p> <p>次の日付形式設定指示では、完全な曜日名、完全な月の名前、西暦年の下 2 桁の順序で表示されます。</p> <pre>EBM_DATE_FORMAT="Phoenix, %A %B %d %H:%M:%S %y"</pre> <p>ジョブ出力</p> <pre>Start of job job1 at "Phoenix, Wednesday May 04 01:45:06 05" Start COB1 at "Phoenix, Wednesday May 04 01:45:28 05" End of COB1 at "Phoenix, Wednesday May 04 01:45:28 05" Job job1 terminating at "Phoenix, Wednesday May 04 01:45:29 05"</pre>

表 B-1 .install ファイルの内容 (6 / 7)

項目	説明
Sun MBM 時刻形式	<p>Sun MBM のジョブの時刻に関する出力形式を定義します。デフォルトでは、Sun MBM は MVS ジョブの時刻形式設定指示に従って時刻を表示します。</p> <pre>EBM_TIME_FORMAT=<i>format</i></pre> <p>Sun MBM の管理者は、必要に応じて別の時刻出力形式を定義できます。詳細は、39 ページの「Sun MBM の時刻形式の変更」と表 4-1 を参照してください。また、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』も参照してください。</p> <p>ジョブ出力</p> <pre>STEP01 step was executed cond code 255 step STEP01 start at TIME=13:45:27 step STEP01 stop at TIME=13:45:28 JOB job1 start at TIME=13:45:27 JOB job1 stop at TIME=13:45:29</pre> <p>このエントリを変更するたびに、Sun MBM を再起動する必要があります。</p> <p>次の時刻形式設定指示では、12 時間表示の時刻および AM/PM の文字列が出力されます。</p> <pre>EBM_TIME_FORMAT=\$I:\$M:%S (%p)</pre> <p>ジョブ出力</p> <pre>STEP01 step was executed cond code 255 step STEP01 start at TIME=14:04:10 (PM) step STEP01 stop at TIME=14:04:11 (PM) JOB job1 start at TIME=14:04:10 (PM) JOB job1 stop at TIME=14:04:12 (PM)</pre>
現在のロケール時間カテゴリ	<p>特定のロケール時間カテゴリを定義します。このエントリを変更するたびに、Sun MBM を再起動する必要があります。Sun MBM ノードがインストールされているシステムで使用可能な任意のロケールを入力パラメータとして指定できます。詳細は、40 ページの「言語環境の管理」を参照してください。</p> <p>形式:</p> <pre>nls_locale_time_category=<i>locale-name</i></pre> <p>次の例は、スペインのロケール時間カテゴリ <code>es_ES.iso88591</code> が使用できるシステムのロケール定義です。</p> <pre>nls_locale_time_category=es_ES.iso88591</pre> <p>ジョブ出力</p> <pre>Start of job job1 at "Martes Mayo 04 01.17.38 2005" Start COB1 at "Martes Mayo 04 01.17.40 2005" End of COB1 at "Martes Mayo 04 01.17.40 2005" Job job1 aborting at "Martes Mayo 04 01.17.42 2005"</pre>
ジョブ アカウントिंग ファイル	<p>形式:</p> <pre>job_acc_file=<i>pathname</i></pre> <p>現在のジョブアカウントिंगファイルを定義します。詳細は、57 ページの「ジョブアカウントING設定のデフォルト設定」を参照してください。</p>

表 B-1 .install ファイルの内容 (7 / 7)

項目	説明
ジョブ アカウントिंग レコード形式	形式: job_acc_format=record-format ジョブアカウントングレコード形式を定義します。詳細は、58 ページの「別のジョブ アカウントングファイルへの変更」を参照してください。

## .install ファイルの例

コード例 B-1 に、.install ファイルの例を示します。

注 – 設定の変更を簡単に行うには、この構成ファイルを管理するための BAM ユーティリティーを使用します。

コード例 B-1 .install ファイル (1 / 4)

```
#Configuration file generated by installation Thu Mar 10 14:03:06 MST 2005
pack=/pkgs/mybatch/nodel/pack
public=/pkgs/mybatch/nodel/public
sysindir=/pkgs/mybatch/nodel/sysindir
#
#The format of each Batch Subsystem entry is:
#<Batch Subsystem>=<setup file path name>
subsys1=/pkgs/mybatch/nodel/subsys1.setup
#
#The default history directory is the directory where the job output is
#redirected if the -h option is not specified in the subjob/unikixjob command.
#The default history directory is "/tmp".
#Add the following entry to change the default history directory:
#history_default=<directory path name>
#
#The job output can be redirected to a different history directory
#by specifying in the -h option of the subjob/unikixjob command one of
#the defined history directories.
#The format of each history directory is:
#<history name>=<directory path name>
history1=/tmp/batchpat/hist1
history2=/tmp/batchpat/hist2
history3=/tmp/batchpat/hist3
#
```

## コード例 B-1 .install ファイル (2 / 4)

```
#The format of each authority file entry is:
#<command>_authority_file=<file path name>
batch_start_authority_file=/pkgs/mybatch/node1/batch_start_auth
#
#Add the following entry to change the default user working directory:
#wk_<user>=<directory path name>
wk_ken=/tmp/ken
#
#Add the following entry to allow remote batch node connections:
#local_ebmnode=<ebmnode>:<port#>

#The format of each remote batch node entry is:
#remote_ebmnode_<ebmnode>=<machine>:<port#>
#
#The default number of activities is 12.
#This is changed by modifying the following line.
#
max_num_acts=20
#
#The default number of records in the circular Sun MBM History File is 1000.
#This is changed by modifying the following line.Note:If you decrease the
#number of records, delete the History File before starting up the batch node.
#
max_history_file_size=1000
#
#The batch console file can be created under the sysindir directory.
#If you choose to create it, it will be called "ebm_console".
#Note:This file is a serial file and batch node opens it in append mode.
#Because the file can grow very large, you must remove it periodically.
#Uncomment the following line to generate the batch console file.
#
ebm_console_file=yes
#
#The batch console terminal must be the full pathname of a terminal device.
#Enter the Unix tty command at the terminal on which you want to display the
#batch console messages.
#It will return the full pathname of that terminal, for example /dev/console.
#Uncomment and modify the following line to activate the batch console terminal.
#
#ebm_console_terminal=/dev/console
#
#The default prefix for messages written to the batch console terminal and/or
#the batch console file is "BatchNode:".
#This is changed by modifying and uncommenting the following line, for example:
#
#ebm_console_ID=[BatchNode1] :
#
```

コード例 B-1      .install ファイル ( 3 / 4 )

```
#The default batch console terminal and/or the batch console file msg level
#is set to "all".This includes all batch node system messages (sysout) and
#job startup and termination messages.
#This is changed by uncommenting one of the following lines:
#
#ebm_console_msg_level=sysout_only
#ebm_console_msg_level=job_only
ebm_console_msg_level=all
#
#If you are using the batch console terminal and/or the batch console file
#you may wish, for performance reasons, to suppress or reduce the messages
#written to the circular batch history file.
#Note:If the "none" option is selected, the lststs and histprt commands
#will not display job output messages.
#Uncomment one of the following lines.Note that the default is to display
#job startup and termination messages (job_only).
#
#ebm_history_file_msg=all
ebm_history_file_msg=job_only
#ebm_history_file_msg=none
#
#
#The default job output date is the same as for the %c formatting
#directive of the standard UNIX command date.
#Enter the UNIX command "man date".It will return the list of available
#formatting directives that you can choose for displaying the job output date.
#Uncomment and update the following date format line to activate your selected
#job output date.
#
#ebm_date_format=%c
#
#Example:ebm_date_format=%A %B %d %H:%M:%S %Y
#
#
ebm_date_format=%A %B %d %H:%M:%S %Y
#
#The default job output time is the same as for the %T formatting
#directive of the standard UNIX command date.
#Enter the UNIX command "man date".It will return the list of available
#formatting directives that you can choose for displaying the job output time.
#Uncomment and update the following time format line to activate your selected
#job output time.
#
#ebm_time_format=%T
#
```

## コード例 B-1 .install ファイル (4 / 4)

```
#Example:ebm_time_format=%I.%M.%S (%p)
#
#
ebm_time_format=%I.%M.%S (%p)
#
#A different locale time category can be set for the job execution environment.
#If available on your system, enter the UNIX command "locale -a".
#It will return the list of available locales.
#Uncomment and update the locale name in the following line.
#
#nls_locale_time_category=locale name
#
#Example:nls_locale_time_category=es_ES.iso88591
#
```

## 付録 C

# batchenv ファイルのカスタマイズ

batchenv ファイルには、ノードの実行を制御する環境変数が含まれています。一部の環境変数には、Sun MBM ソフトウェアがあるディレクトリの名前が含まれています。

## batchenv ファイルの内容

batchenv ファイルは、ノードのインストール時にインストールディレクトリに作成されます。たとえば、Sun MBM が /mybatch/node1 ディレクトリにインストールされると、batchenv ファイルは /mybatch/node1/batchenv となります。

**注** - batchenv ファイルは Bourne シェルスクリプトです。C シェルを使用する場合は、C シェル形式でコピーを作成する必要があります。

batchenv ファイルには、コード例 C-1 に示す環境変数が含まれています。各変数の設定は、実際のシステム設定およびオペレーティングシステムによって異なります。BAM ユーティリティーを使用して Sun MBM システムを構成すると、このファイルは自動更新されて設定が反映されます。

### コード例 C-1      Solaris システムの batchenv ファイル — 例 ( 1 / 2 )

```
#
# Batch Environment Setting
#

EBMHOME=/usr/local/pkgs/MBM10.1.0;export EBMHOME
DEVNULL=/dev/null; export DEVNULL
EBMSOCKDIR=$EBMHOME/sockets;export EBMSOCKDIR
KIXLICDIR=/usr/local/pkgs/licenses; export KIXLICDIR
```

## コード例 C-1 Solaris システムの batchenv ファイル — 例 (2 / 2)

```
PACK=$EBMHOME/pack; export PACK
PUBLIC=$EBMHOME/public; export PUBLIC
SYSINDIR=$EBMHOME/sysindir; export SYSINDIR
TPDATE=Y2MD/; export TPDATE
DUMPSHMEM=1; export DUMPSHMEM
DUMPCORE=1; export DUMPCORE
DUMPDIR=/tmp; export DUMPDIR
PS1="[\`basename $EBMHOME`] $PS1"; export PS1
PS1="[\`basename $EBMHOME`] # 1"; export PS1
MBMNLSPATH=$EBMHOME/pack/locale/%L/%N:$EBMHOME/pack/%N:$NLSPATH; export
MBMNLSPATH
NLSPATH=$MBMNLSPATH; export NLSPATH
...
...
...
EBM_OS_NAME=SUN2; export EBM_OS_NAME
PATH=$PACK/bin:$PATH:/usr/openwin/bin; export PATH
XTERM=dtterm; export TERM
```

### 説明

---

EBMHOME	Sun MBM インストールディレクトリを示します。
DEVNULL	システム固有のダミー入出力デバイスを示します。
DUMPSHMEM, DUMPCORE, DUMPDIR	内部 Sun MBM 処理に使用されます。DUMPDIR ディレクトリは、Sun MBM が内部ダンプを書き込む位置を示します。
KIXLICDIR	ライセンスファイルが含まれるディレクトリを示します。
EBMSOCKDIR	内部ソケットディレクトリを示します。各 Sun MBM ノードが 3 つのドメインソケットファイルを \$EBMSOCKDIR ディレクトリに作成します。このディレクトリは変更しないでください。この 3 つのファイルは、メッセージサーバーデーモン、仮想コンソール機能デーモン、およびログデーモンにより、通信チャネルとして使用されます。
PACK, PUBLIC, SYSINDIR	pack、public、および sysindir ディレクトリのパスを示します。
PATH	\$PATH ディレクトリを示します。\$PACK/bin ディレクトリが \$PATH のパス名リストに追加されます。Sun MBM コマンドを実行するために必要です。

---

---

注 - ジョブエディタを使用する場合、\$JDKROOT 環境変数を batchenv ファイルに追加する必要があります。\$JDKROOT は、Java ソフトウェアのインストールディレクトリを示します。

---

## Sun MBM コマンド出力の制御

batchenv ファイルに追加の環境変数を設定することにより、Sun MBM コマンド出力を制御できます。この設定変更は、環境変数を追加した端末だけに適用されるもので、ジョブの実行日時には影響しません。

### EBM\_DATE\_FORMAT

次のコマンドの出力日付に影響します。

```
ebmdate
histprt1
ebminfo (-D オプション)
lgprint
ebmsys
```

### 形式

```
EBM_DATE_FORMAT=format; export EBM_DATE_FORMAT
```

適切な出力日付形式を選択できます。有効な形式設定指示は、表 4-1 を参照してください。EBM\_DATE\_FORMAT のデフォルト値は %c です。

### 例

完全な曜日名、完全な月の名前、および 12 時間表示形式による時刻を表示します。

```
export EBM_DATE_FORMAT=" %A %B %d %I:%M:%S %p %Y "
```

### ジョブ出力

```
Wednesday May 04 02:53:05 AM 2005
```

## EBM\_TIME\_FORMAT

次のコマンドの出力に影響します。

```
ebmtime  
ebminfo (-T オプション)
```

### 形式

```
EBM_TIME_FORMAT=format; export EBM_TIME_FORMAT
```

デフォルトでは、MVS ジョブ時刻形式で時刻が出力されます。別の時刻形式も選択できます。有効な形式設定指示は、表 4-1 を参照してください。EBM\_TIME\_FORMAT のデフォルト値は %T です。

### 例

12 時間表示形式の時刻を出力します。

```
export EBM_TIME_FORMAT="%I:%M:%S %p"
```

ebmtime の出力 (または ebminfo -T)

```
[node1] $ ebmtime
```

```
03:09:20 AM
```

## LC\_TIME

ロケール時間カテゴリは、次のコマンドの動作に影響します。

```
ebmdate  
ebmtime  
ebmsys  
histprt  
lgprint  
subjob  
unikixjob
```

これらのコマンドの詳細は、『Sun Mainframe Batch Manager ソフトウェア ユーザーズガイド』を参照してください。

### 形式

```
LC_TIME=locale-name; export LC_TIME
```

Sun MBM ノードがインストールされているシステムで使用可能な任意のロケールを、画面に表示されるカテゴリの中から 1 つ選択して、入力パラメータとして指定できます。

## 例

LC\_TIME 環境変数に設定したスペインのロケール時間カテゴリ `es_ES.iso88591` に基づく出力です。

```
export LC_TIME=es_ES.iso88591
```

ebmtime (または ebminfo -D) 出力:

```
[node1] ebmtime  
** Sábado Mayo 07 03:21:43 2005 **
```



## 付録D

# サブシステム設定ファイル (\$SETUP)

---

この付録の内容は、次のとおりです。

- 184 ページの「標準環境変数」
- 186 ページの「RDBMS 環境変数」
- 186 ページの「その他の環境変数」

サブシステムを作成すると、BAM によって 2 つの設定ファイルが作成されます。

- Setup (\$SETUP) は、サブシステムの作成時に設定した環境変数を格納している読み取り専用ファイルです。このファイルは編集しないでください。
- Setup\_usr (\$SETUP\_USER) は、ユーザーによる編集が可能なファイルで、ほかの環境変数を追加できます。

---

**注** – 取り込める設定ファイルは C シェル形式です。

---

すべてのバッチジョブは、サブシステムにサブミットされます。そのサブシステムの設定ファイルが、実行環境を規定するためにジョブの開始時に実行されます。

この付録で説明する環境変数は、ユーザーがサブシステムを作成するときに選択したオプションに基づいて、読み取り専用のサブシステム設定ファイル (\$SETUP) に設定されます。ファイルには、選択内容に基づいて設定されたほかの環境変数が含まれていることがあります。ユーザー設定ファイルについては、97 ページの「サブシステムのユーザー設定ファイルのカスタマイズ」および 98 ページの「ユーザー設定ファイルを編集する」を参照してください。

---

# 標準環境変数

## ACUCOBOL

ACUCOBOL-GT を使用する場合、COBOL がインストールされているディレクトリを定義します。

## A\_TERMCPAP

ACUCOBOL-GT を使用する場合、ACUCOBOL-GT 端末定義ファイルのフルパスを表す `$ACUCOBOL/etc/a_termcap` に設定されます。

## COBDIR

Server Express を使用する場合、COBOL がインストールされているディレクトリを定義します。

## EBMTMPDIR

一時ファイル用のディレクトリ。

## FILEMAP

ジョブの実行時にどの `File_Map` を使用するかを定義します。必須エントリです。これは、JCL の変換時に使用される `File_Map` と同じです。次に例を示します。

```
/test/MBM10.1.0/bam/subsys/cobsubsys/File_Map
```

## HANDLE\_MFVS

COBOL アプリケーションが、Sun MTP 索引編成ファイルと相対編成ファイル、および COBOL 索引編成ファイルと相対編成ファイルにアクセスする場合、1 に設定されます。

## KIXSYS

Sun MTP 領域のシステムファイルが含まれるディレクトリを定義します。VSAM サブシステムを作成する場合に設定されます。`$KIXSYS` には絶対パス名だけを指定します (\$ および ~ 文字は使用不可)。詳細は、『Sun Mainframe Transaction Processing ソフトウェア 構成ガイド』を参照してください。

## LD\_LIBRARY\_PATH

Solaris プラットフォームのシステム共有ライブラリの場所を定義します。

## LIBPATH

AIX プラットフォームのシステム共有ライブラリの場所を定義します。

## LPI\_PRODUCT\_DIR

Open PL/I を使用する場合は、Open PL/I がインストールされているディレクトリを定義します。

## RUNBDIR

COBOL だけ、または COBOL および PL/I を実行する場合の環境変数です。

- RUNB COBOL 実行時システムがインストールされているディレクトリに設定されます。
- \$RUNBDIR は、RUNBPATH 環境変数のディレクトリのいずれかに設定されます。設定が正しくない場合は、実行時にエラーが発生します。

PL/I だけを実行する場合、\$RUNBDIR は次のように設定されます。

```
setenv RUNBDIR $PACK/bin
```

## RUNBPATH

COBOL だけ、または COBOL および PL/I を実行する場合の環境変数です。

複数の COBOL 実行時システムをインストールできるディレクトリのリストです。RUNBPATH を設定しなかった場合は、RUNBDIR を正しく設定してあっても、ジョブの実行時にエラーが発生します。

## SORT\_MODE

ジョブの実行時に使用されるソートユーティリティーを定義します。Sun 以外のソートパッケージのうち、次のいずれかを BAM サブシステム作成時に選択した場合、この環境変数が設定されます。

- SyncSort
- CoSORT

たとえば、ソートユーティリティーとして CoSORT を選択すると、設定ファイルでは次のようになります。

```
setenv SORT_MODE cosort
```

## UNIKIX

Sun MTP インストールディレクトリを定義します。Sun MTP 領域に接続するために VSAM サブシステムを作成する場合に設定します。

---

## RDBMS 環境変数

サブシステムが RDBMS にアクセスする場合は、BAM を使用してその RDBMS に関する環境変数を定義します。必要な環境変数については、ご使用の RDBMS のマニュアルを参照してください。

### RDBMS\_CONNECT

SQL CONNECT/ DATABASE 文に必要なデータベースまたはログイン ID 名を定義します。この環境変数は、Sun MTP VSAM データセットにアクセスしない Sun MBM COBOL 実行時システム用に設定されます。サブシステムの作成方法については、67 ページの「サブシステムの作成」を参照してください。

### RTS\_RDBMS

アクセスするデータベースを定義します。RDBMS の選択方法については、70 ページの「1 つまたは複数のデータベースを選択する」を参照してください。

---

## その他の環境変数

### FORMS

SYSOUT クラスが F の場合、アプリケーション出力レポートが書き込まれるディレクトリを定義します。

### SYSOUTDIR

SYSOUT クラスが F でない場合、アプリケーション出力レポートが書き込まれるディレクトリを定義します。

## 付録E

# RDBMS 用アプリケーションの カスタマイズ

---

この付録の情報が適用されるのは、リレーショナルデータベース管理システム (RDBMS) へのアクセスを必要とするが、Sun MTP 領域には関連付けられていないサブシステムのみです。Sun MBM の RDBMS サブシステムが領域に関連付けられている環境では、RDBMS へのアクセスは領域のトランザクション処理プログラムによって処理されます。詳細は、『Sun Mainframe Transaction Processing ソフトウェア 管理者ガイド』の「アプリケーション環境のカスタマイズ」の章を参照してください。

この付録の内容は、次のとおりです。

- 188 ページの「RDBMS へのアクセス」
- 189 ページの「fsusrexist.c - Oracle 関数」
- 191 ページの「fsusrexist.c - DB2 UDB 関数」
- 193 ページの「fsusrexist.c - Sybase 関数」

---

## RDBMS へのアクセス

\$PACK/RTSFS ディレクトリに、RDBMS へのアクセスをサポートするモジュールが含まれています。ユーザーは、使用するアプリケーション環境に応じて次のモジュールを変更できます。

---

runfs.c	データベースを割り当てたり、データベースをコミットまたはロールバックしたりするための RDBMS への呼び出しが含まれます。 このモジュールは、COBOL バッチプログラムの実行前に、runfs_main エントリポイントを実行します。 また、COBOL バッチプログラムの実行終了時には runfs_stop エントリポイントを呼び出します。
fsusrexite.c	指定した RDBMS 関数を呼び出す COBOL プログラムへの呼び出しが含まれます。 複数のデータベースまたは新しいデータベースにアクセスする必要がある場合は、BAM でサブシステムを作成する前に、fsusrexite.c モジュールを変更する必要があります。

---

RDBMS データベースは実行時システムによって自動的に割り当てられるので、バッチプログラムでデータベースを明示的に割り当てる必要はありません。COBOL プログラムに明示的な allocate 文が含まれていると、エラーメッセージが出力されます。その場合は allocate 文を削除するか、エラーメッセージを無視します。

回復プロシージャは通常、RDBMS によって処理されます。たとえば、COBOL プログラムが強制的に中止した場合、そのアプリケーションプログラムによって行われたデータベースに対する変更を RDBMS が自動的にロールバックします。

fsusrexite.c モジュールに定義されていないデータベースにアクセスする必要がある場合は、トランザクションまたはシステムの障害から回復するための適切な方法を決めておきます。

---

## fsusrexit.c - Oracle 関数

関数 KXORALGN、KXORALGF、KXORASAVE、および KXORAUNDO を、\$PACK/RTSFS ディレクトリにある KXORACLE.pco プログラムに定義します。

Oracle<sup>®</sup> コードが必要な場合は、makefile 内の RDBMSFLAG を -DORACLE に設定します。

### コード例 E-1 Oracle 用の fsusrexit.c (1 / 2)

```
/*
 *
 * The following code is specific to the support of Oracle databases.
 * The routines called by the functions below can be found in the
 * FSORACLE.pco source.
 *
 */

#ifdef ORACLE
/*
 ORACLE allocate/open function
 */

static int oracle_allocate() {
    if (strcmp(rts_rdbms, "ORA_APPL") == 0) {
#ifdef ACUCOBOL
        return(kxcobload("KXORACLE", KXDEBUG_NONE, 0, NULL, NULL, LOGERR, NULL, 0));
#else
        return(KXORALGN());
#endif
    }
    return(0);
}
```

コード例 E-1 Oracle 用の fsusrexit.c (2 / 2)

```
/******  
/* ORACLE deallocate/close function */  
/******  
  
static int oracle_deallocate() {  
    if (strcmp(rts_rdbms,"ORA_APPL") == 0) {  
#ifdef ACUCOBOL  
        return(kxcobload("KXORALGF",KXDEBUG_NONE,0,NULL,NULL,LOGERR,NULL,0));  
#else  
        return(KXORALGF());  
#endif  
    }  
    return(0);  
}  
  
/******  
/* ORACLE commit function */  
/******  
  
static int oracle_commit() {  
    if (strcmp(rts_rdbms,"ORA_APPL") == 0) {  
#ifdef ACUCOBOL  
        return(kxcobload("KXORASAVE",KXDEBUG_NONE,0,NULL,NULL,LOGERR,NULL,0));  
#else  
        return(KXORASAVE());  
#endif  
    }  
    return(0);  
}  
  
/******  
/* ORACLE rollback function */  
/******  
  
static int oracle_rollback() {  
    if (strcmp(rts_rdbms,"ORA_APPL") == 0) {  
#ifdef ACUCOBOL  
        return(kxcobload("KXORAUNDO",KXDEBUG_NONE,0,NULL,NULL,LOGERR,NULL,0));  
#else  
        return(KXORAUNDO());  
#endif  
    }  
    return(0);  
}  
  
#endif /* End Oracle code */
```

---

## fsusrexit.c - DB2 UDB 関数

関数 KXDB2LGN、KXDB2LGF、KXDB2SAVE、および KXDB2UNDO を、\$PACK/RTSFS ディレクトリにある KXDB2.cbl プログラムに定義します。

DB2 UDB コードが必要な場合は、makefile 内の RDBMSFLAG を -DDBTWO に設定します。

### コード例 E-2 DB2 UDB 用の fsusrexit.c (1 / 2)

```
/*
*****
/* The following code is specific to the support of DB2 databases */
/* The routines called by the functions below can be found in */
/* KXDB2.cbl. */
*****

#ifdef DBTWO

/*
*****
/* DB26000 allocate/open function */
*****

static int dbtwo_allocate() {
    if (strcmp(rts_rdbms,"DB2_APPL") == 0) {
#ifdef ACUCOBOL
        return(kxcobload("KXDB2LGN",KXDEBUG_NONE,0,NULL,NULL,LOGERR,NULL,0));
#else
        return(KXDB2LGN());
#endif
    }
    return(0);
}
}
}
```

コード例 E-2 DB2 UDB 用の fsusrexit.c (2 / 2)

```
/* **** */
/* DB26000 deallocate/close function */
/* **** */

static int dbtwo_deallocate() {
    if (strcmp(rts_rdbms,"DB2_APPL") == 0) {
#ifdef ACUCOBOL
        return(kxcobload("KXDB2LGF",KXDEBUG_NONE,0,NULL,NULL,LOGERR,NULL,0));
#else
        return(KXDB2LGF());
#endif
    }
    return(0);
}

/* **** */
/* DB26000 commit function */
/* **** */

static int dbtwo_commit() {
    if (strcmp(rts_rdbms,"DB2_APPL") == 0) {
#ifdef ACUCOBOL
        return(kxcobload("KXDB2SAVE",KXDEBUG_NONE,0,NULL,NULL,LOGERR,NULL,0));
#else
        return(KXDB2SAVE());
#endif
    }
    return(0);
}

/* **** */
/* DB26000 rollback function */
/* **** */

static int dbtwo_rollback() {
    if (strcmp(rts_rdbms,"DB2_APPL") == 0) {
#ifdef ACUCOBOL
        return(kxcobload("KXDB2UNDO",KXDEBUG_NONE,0,NULL,NULL,LOGERR,NULL,0));
#else
        return(KXDB2UNDO());
#endif
    }
    return(0);
}

#endif /* End db26000 code */
```

## fsusrexite.c - Sybase 関数

関数 KXSYBLGN、KXSYBLGF、KXSYBSAVE、および KXSYBUNDO を、\$PACK/RTSFS ディレクトリにある KXSYBASE.cop プログラムに定義します。

Sybase コードが必要な場合は、makefile 内の RDBMSFLAG を -DSYBASE に設定します。

### コード例 E-3 Sybase 用の fsusrexite.c (1 / 3)

```
/*
*****
/* The following code is specific to the support of Sybase databases */
/* The routines called by the functions below can be found in */
/* KXSYBASE.cop. */
*****

#ifdef SYBASE

/*
*****
/* SYBASE allocate/open function */
*****

static int sybase_allocate() {

/*-----*/
/* RS6000 and SYBASE10 specific for changing process group id */
/*-----*/

#ifdef SYBASE10
#ifdef AIX
int lrc;
pid_t l_pgid;

l_pgid = getpid();
lrc = setpgid(l_pgid, l_pgid);
if (lrc != 0)
{
perror("rtsfs:setpgid()");
return(-1);
}
#endif
#endif /* AIX condition */
#endif /* SYBASE10 condition */

```

コード例 E-3 Sybase 用の fsusrexit.c ( 2 / 3 )

```
/*-----*/
    if (strcmp(rts_rdbms,"SYB_APPL") == 0) {
#ifdef ACUCOBOL
        return(kxcobload("KXSYBLGN",KXDEBUG_NONE,0,NULL,NULL,LOGERR,NULL,0));
#else
        return(KXSYBLGN());
#endif
    }
return(0);
}

/*****
/*      SYBASE      deallocate/close function      */
/*****

static int sybase_deallocate() {
    if (strcmp(rts_rdbms,"SYB_APPL") == 0) {
#ifdef ACUCOBOL
        return(kxcobload("KXSYBLGF",KXDEBUG_NONE,0,NULL,NULL,LOGERR,NULL,0));
#else
        return(KXSYBLGF());
#endif
    }
return(0);
}

/*****
/*      SYBASE      commit function      */
/*****

static int sybase_commit() {
    if (strcmp(rts_rdbms,"SYB_APPL") == 0) {
#ifdef ACUCOBOL
        return(kxcobload("KXSYBSAVE",KXDEBUG_NONE,0,NULL,NULL,LOGERR,NULL,0));
#else
        return(KXSYBSAVE());
#endif
    }
return(0);
}
}
```

コード例 E-3      Sybase 用の fsusrexit.c (3 / 3)

```
/*
 * SYBASE rollback function
 */
static int sybase_rollback() {
    if (strcmp(rts_rdbms, "SYB_APPL") == 0) {
#ifdef ACUCOBOL
        return(kxcobload("KXSYBUNDO", KXDEBUG_NONE, 0, NULL, NULL, LOGERR, NULL, 0));
#else
        return(KXSYBUNDO());
#endif
    }
    return(0);
}

#endif /* End Sybase code */
```



# Sun MTP との通信

---

この付録では、Sun MBM が Sun MTP と通信する仕組みについて説明します。この章の内容は、次のとおりです。

- 197 ページの「Sun MTP 環境変数」
- 198 ページの「Sun MTP VSAM データセットへのアクセス」
- 199 ページの「ノードに接続する領域の設定」

Sun MBM は、Sun MTP 開始キュー処理プログラム unikixstrt と通信します。このプロセッサはバッチジョブをスケジューリングし、そのジョブにトランザクション処理プログラム unikixtran を割り当てます。ただし、ジョブは、割り当てられたトランザクション処理プログラムで実行されるのではなく、Sun MBM が作成したプロセス内で実行されます。トランザクション処理プログラムは、ジョブから VSAM データセットへのアクセスを可能にするために、そのジョブに対して割り当てられます。

---

注 - 外部 CICS インタフェース (EXCI) を使用して領域と通信する必要がある場合は、『Sun Mainframe Batch Manager ソフトウェア 移行ガイド』を参照してください。

---

---

## Sun MTP 環境変数

Sun MTP 領域に接続するサブシステムを作成する場合、サブシステム設定ファイルには領域の環境変数 UNIKIX および KIXSYS が含まれている必要があります。BAM によって、これらの変数の値が要求されます。領域の設定ファイルで設定され、Sun MTP 領域が開始されるときに継承される環境変数 (KIXDATA または VSAMDATA など) の値を基に、VSAM データセットの場所が決められます。

BAM を使用して環境変数を更新する方法については、92 ページの「サブシステムの更新」を参照してください。

# Sun MTP VSAM データセットへのアクセス

Sun MTP の VSAM データセットにアクセスするバッチジョブをサブミットするには、unikixjob コマンドを使用します。領域が有効であり、Sun MBM ノードに接続するように設定されている必要があります。ジョブのサブミット先は、デフォルトのサブシステムにも、unikixjob の -k オプションで指定したサブシステムにもできます。

Sun MBM は、有効なアクティビティーに対して最も優先順位の高いジョブを選択することで、どのジョブを実行するか決定します。その決定されたジョブが領域にサブミットされます。サブミットされたバッチジョブは、オンライントランザクションで使用されるのと同じ VSAM アクセス方式によって、領域側で実行されます。Sun MBM では、領域で実行可能なジョブ数に達するまで、優先順位の条件で選択したジョブのサブミットが続行されます。

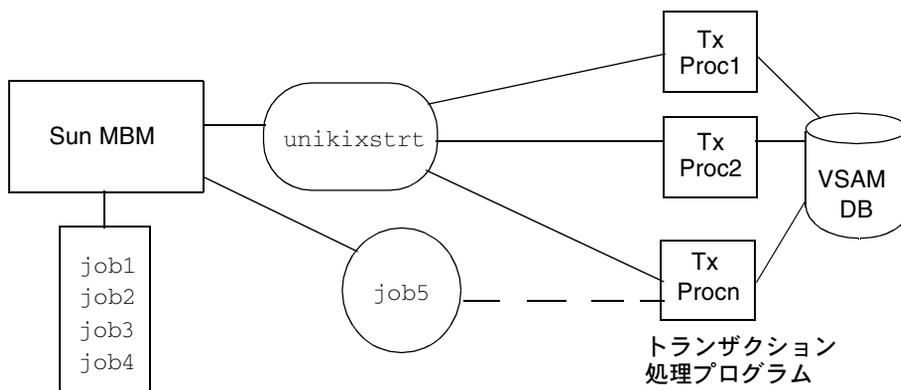


図 F-1 Sun MBM による VSAM データセットへのアクセス

---

## ノードに接続する領域の設定

Sun MTP の VSAM ファイルにアクセスするバッチジョブの場合、領域が、その領域の VSAM 構成テーブル (VCT) の値に基づいてバッチジョブの実行を制御します。

次の VCT パラメータを設定する必要があります。

---

Connect to Batch Node:(Y/N)	Y と入力します。
Node Dir	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。
Batch Search Interval	ジョブが実行可能であるかを確認するために領域が Sun MBM と通信する間隔を秒単位で指定します。
Maximum Batch Jobs	1 以上の値を設定します。Sun MBM で作成されるアクティビティの数は、このパラメータ値以上であることが前提となります。

---

Sun MTP VSAM データ管理を指定したサブシステムを定義する必要があります。

---

**注** - ノードが稼働中で、定義されている VSAM サブシステムがない場合、Sun MBM への接続が設定されている領域を起動しようとする、その領域がノードに接続されていないことを示すメッセージが、バッチ間隔 (VCT に設定されている時間間隔) ごとに領域のログに書き込まれます。

---

すべての VCT パラメータについては、『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。



# WebSphere MQ との統合

---

この付録では、IBM WebSphere MQ 製品との統合を実現するための Sun MBM 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。

- 201 ページの「前提条件」
- 202 ページの「MQ サブシステムの作成」

Sun MBM は、VSAM サブシステム環境で IBM WebSphere MQ をサポートします。2 フェーズコミットはサポートしないので、Sun MBM では、MQ はクライアントモードでのみ実行されます。

---

## 前提条件

MQ をサポートする Sun MTP 領域の作成。アプリケーション言語として Micro Focus Server Express を使用する必要があります。MQ と領域の統合については、『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。

Sun MTP 領域で使用される MQSERVER および MQSERIES 環境変数の値を識別します。MQSERVER 変数は MQ で必要とされます。この変数には、MQ に接続するためのチャンネル、プロトコル、ホスト、およびポート番号が、「CHANNEL1/TCP/myhost (10200)」などの形式で格納されています。MQSERIES 変数は、MQ がインストールされた場所を示します。

---

# MQ サブシステムの作成

Sun MBM サブシステムを作成し、MQ サポート付きで作成された領域に関連付けます。

## ▼ MQ VSAM サブシステムを作成する

1. 次の情報を確認してください。
  - MQSERVER 環境変数の値
  - MQSERIES 環境変数の値
  - UNIKIX 環境変数の値 (Sun MTP インストールディレクトリ)
  - KIXSYS 環境変数の値 (領域のシステムディレクトリ)
2. ノードが実行されていることを確認します。
3. 9 ページの「BAM の起動」の説明に従って、BAM ユーティリティを起動します。
4. BAM メニューで「Applications & Subsystems」 → 「Create a Subsystem」を選択します。
5. MQ サブシステムの名前を入力し、Return キーを押します。
6. 「Create」メニューでオプション「1 Application Languages」を選択し、Server Express が選択されていることを確認します。  
Server Express はデフォルトです。
7. Return キーを押して、「Create」メニューに戻ります。
8. 「Create」メニューでオプション「3 Data Management」を選択します。
9. VSAM データ管理オプションをいずれか 1 つ選択して、Return キーを押します。

10. 「Create」メニューでオプション「4 Third Party Packages」を選択します。

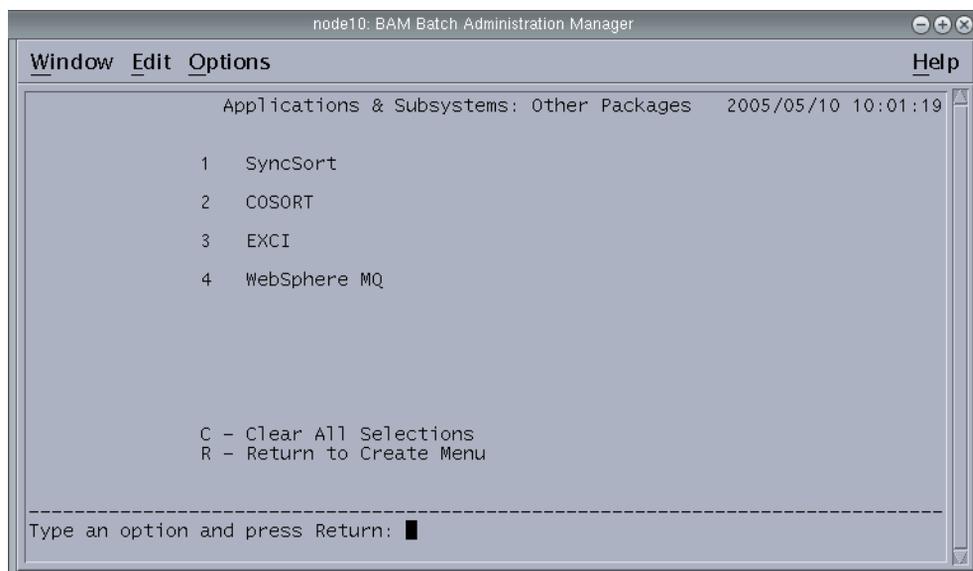


図 G-1 Other Packages 画面

11. WebSphere MQ を選択して、Return キーを押します。
12. 「Create」メニューで、サブシステム環境で必要なその他の選択を行います。
13. 「Create」メニューでオプション「C」を選択し、サブシステムを作成します。

MQSERVER 変数および MQSERIES 変数の値の指定を要求されたら、Sun MTP 領域に使用するものと同じ値を入力します。また、領域の \$UNIKIX および \$KIXSYS の値も指定する必要があります。必ず MQ サポート付きで作成した領域の値を指定してください。

14. サブシステムが正常に作成されたら、BAM メニューに戻ります。

## ▼ 領域と MQ サブシステムを関連付ける

1. Sun MTP の MQ 領域を開始して、クライアント接続を確立します。
2. テーブルマネージャーを開きます。
3. VSAM 構成テーブル (VCT) を開きます。
4. 「Maximum Batch Jobs」フィールドに 1 以上の値が入力されていることを確認します。  
この値は、バッチノード用に指定されているアクティビティーの数以下である必要があります。
5. 「Batch Search Interval」フィールドの値がサイトの要件に一致していることを確認します。
6. 「Connect to Batch Node」フィールドを Y に設定します。
7. 「Node Dir」フィールドに、MQ VSAM サブシステムを作成したノードのディレクトリパスを入力します。
8. 再度、領域を開始し、変更を有効にします。

# 用語集

---

---

## B

**Batch Administration  
Manager (BAM)**

(名詞) Sun MBM ノードやサブシステムを設定したり管理したりするために使用されるツール。

**batchenv ファイル**

(名詞) ノードの実行方法を制御する環境変数を含む設定ファイル。各ノードは固有の batchenv ファイルを持ち、それらを実行しないとノードが開始されません。

---

## F

**File\_Map**

(名詞) IBM データセット、ライブラリ、および世代別データグループ (GDG) と、対応する UNIX パス名とを関連付けるエントリを含む特別ファイル。このファイルは、Sun MBM JCL トランスレータおよびサブシステムによって、メインフレーム JCL ストリームの変換時やマクロジョブスクリプトの実行時に使用されます。サブシステムはそれぞれ 1 つの File\_Map に関連付けられています。

---

## I

**.install ファイル** (名詞) インストールされたノードおよびそれに関連付けられたサブシステムすべての情報を含んだグローバル Sun MBM 構成ファイル。 .install ファイルは、ノードをインストールディレクトリにインストールするときに作成されます。 .install ファイルは、ノードを開始するたびに読み込まれます。

---

## S

**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、バッチ作業負荷を処理し、割り当てられたパラメータ (開始時刻、バッチプロセスの最大数、ジョブの優先順位など) を基に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、および 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。 クライアントを除く 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 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。

---

## V

**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。

---

## あ

**アクティビティー** (名詞) ジョブを実行する共有メモリーのセグメント。アクティビティーはジョブクラスに割り当てられます。1つのクラスに99までのアクティビティーを割り当てることができます。

---

## え

**エラーログ** (名詞) Sun MBM によって、ジョブ実行に関わる Sun MBM デーモンおよびバッチプロセスからデバッグメッセージが収集されたファイル。

---

## か

**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。

---

## き

**許可ファイル** (名詞) ユーザーがノードを開始、管理、停止する権限、サブシステムを管理する権限、およびクラスやアクティビティーを作成、変更、削除する権限を制御するファイル。

---

## く

**クラス** (名詞) 1つまたは複数のアクティビティーを含む概念エンティティー。ジョブはクラスにサブミットされ、アクティビティーが利用可能な場合に実行されます。1つのノードで26クラスをサポートできます。

---

## こ

**コンソール端末** (名詞) ノードにオペレータコンソールとして定義される端末デバイス。さまざまなタイプのエラーメッセージを設定して、コンソール端末に表示できます。

**コンソールファイル** (名詞) Sun MBM コンソール端末に表示されるものと同じメッセージタイプを格納する連続したファイル。

---

## さ

**サブシステム** (名詞) 特定のノードに従属する環境であり、ここで特定のタイプのジョブが実行されます。たとえば、Sun MTP 領域の VSAM データセットにアクセスするジョブを実行するために使用するサブシステムを作成できます。

**サブシステム設定  
ファイル** (名詞) サブシステムの作成時、BAM は 2 つの設定ファイルを作成します。1 つは読み取り専用ファイルで、サブシステム作成時に設定される環境変数を含んでいます。もう 1 つはユーザー編集可能ファイルで、ほかの環境変数を追加できます。これらのファイルはそれぞれ、`$SETUP` および `$USER_SETUP` として参照されます。

---

## し

**ジョブクラス** 「クラス」を参照。

**ジョブの履歴ファイル** (名詞) バッチジョブの実行中に収集されたシステムメッセージとアプリケーションメッセージを含むファイル。「ジョブ出力一覧ファイル」とも呼ばれます。

---

## す

**スレッド** 「アクティビティ」を参照。

---

## て

デフォルトの  
サブシステム

(名詞) ジョブのサブミット時にサブシステムが指定されていない場合に、ジョブが実行されるサブシステム。

---

## の

ノード

(名詞) Sun MBM ソフトウェアの一意のインストール。

---

## は

パーティション

(名詞) 共有メモリーの 1 セグメント。

---

## ふ

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。

ファイルのアクセス権  
(またはモード)

(名詞) オペレーティングシステムの定義に従って、ファイルへのアクセスを制御します。

---

## 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

**履歴ファイル** (名詞) ジョブの開始およびジョブの終了メッセージを含む循環ファイル。

---

## 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その関連ファイルが格納されています。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 (/) で参照されます。

# 索引

---

## A

### ACUCOBOL-GT

- ACUCOBOL 変数, 184
- A\_TERMCAP 変数, 184

### Admlog 機能

- Configure Log Environment, 138
- 循環ファイル, 139
- メニュー, 137

### allocate 文, 188

### A\_TERMCAP 環境変数, 184

### autosub ディレクトリ, 162

### autosub デーモン, 164

## B

### BAM

- bam\_authority\_file ファイル, 169
- Sun 以外のパッケージ、選択, 72
- 画面
  - Activate Job Accounting, 56
  - Admlog メニュー, 137
  - Application Languages, 69
  - Applications & Subsystems メニュー, 66
  - Assign the Console メニュー, 21
  - Change Job Accounting File, 58
  - Change Remote Batch Node, 50
  - Change the Date, 32
  - Change User's Work Directory, 123
  - Classes & Activities メニュー, 129

### COBOL Date/Time, 73

### Command Controls メニュー, 110

### Create a Subsystem, 67

### Create User's Work Directory, 122

### Data Management, 71

### Deactivate Message Log, 145

### Default Subsystem, 99

### Define Remote Batch node, 48

### Delete a Job Class, 133

### Delete Remote Batch Node, 51

### Delete User's Work Directory, 124

### Delete/(Define) Local Batch Node, 48

### Dump Memory, 145

### Inter-Node Communications メニュー, 44

### Job Accounting メニュー, 55

### List Active Jobs, 16

### List Active Subsystems, 15

### List All Job Classes, 129

### List All User's Work Directories, 121

### Mainframe Compatibility メニュー, 75

### MTP 環境設定, 83

### MTP 領域の同期化, 34

### MTP 領域の変更, 95

### MVS Compatibility, 76

### Other Packages, 72

### Query Completed Jobs, 17

### Query File\_Map, 91, 92

### Query Setup, 91

### Query メニュー, 89

### Report System Status, 14

### Security & Users メニュー, 108

- Set Circular History File ON, 25
- Show Characteristics, 89
- Show Message Traces, 144
- Show Subsystem's Configuration, 90
- Sun MBM の起動日, 37
- System Environments メニュー, 12
- Trace Messages, 143
- Trace Processes, 142
- Trace Subsystems, 140
- User-Specific Objects, 74
- User's Work Directory メニュー, 120
- VSE 互換性, 81
- アクティビティーの最大数, 132
- 現在のジョブアカウンティング設定, 55
- サブシステムの一覧表示, 67
- サブシステムの現在の設定内容の表示, 82
- サブシステムの再作成, 102
- ジョブアカウンティングの無効化, 57
- ジョブアカウンティングのレコード形式, 59
- ジョブクラスの作成, 131
- ジョブの出力のリダイレクト, 27, 28
- ヘルプ, 10
- メインメニュー, 10
- ロケール時間カテゴリ, 40
- 起動, 9
- すべてのノードの一覧表示, 44
- データベース、選択, 70
- ノードコンソール
  - 現在の設定, 22
  - 端末, 22
  - ファイル, 23
- ノードの起動, 12
- ノードの停止, 13
- 表示
  - サブシステム設定ファイル, 91
  - サブシステムの File\_Map, 91, 92
  - サブシステムの構成, 90
  - サブシステムのジョブ, 91
  - ジョブ履歴ディレクトリ, 91
- ヘルプファイル, 10
- メインメニュー, 10
- ユーザー固有オブジェクト, 74
- ログファイル, 85, 150
- ロケール時間カテゴリの変更, 40

- bam\_authority\_file ファイル, 169
- bam.log ファイル, 85, 150
- bam\_shut\_authority\_file ファイル, 169
- bam コマンド, 9
- BAM の起動, 9
- BAM のヘルプ, 10
- batchenv ファイル
  - BAM の起動, 9
  - カスタマイズ, 177 ~ 181
  - コマンド出力の制御, 179
  - サブシステム環境の設定, 126
  - 内容, 177
- batch\_start\_authority\_file ファイル, 169
- .btshrc ファイル, 151, 152

## C

- C/C++、選択, 69
- COBDIR 環境変数, 184
- COBOL
  - allocate 文, 188
  - 索引編成ファイル, 71
  - 選択, 69
  - 相対編成ファイル, 71
  - 日付/時刻管理, 73
  - ユーザー固有オブジェクト, 74
- CoSORT, 72, 185
- crtact\_authority\_file ファイル, 169

## D

- DB2 UDB, 191
- DCT。「宛先管理テーブル」を参照
- DEVNULL 環境変数, 178
- dltact\_authority\_file ファイル, 169
- DOSTRANS\_OPT 環境変数, 165
- dostrans コマンド, 164, 165
- DUMPCORE 環境変数, 178
- DUMPDIR 環境変数, 178
- DUMPSHMEM 環境変数, 178

## E

ebm\_console ファイル, 170  
EBM\_DATE\_FORMAT 環境変数, 171  
EBMHOM 環境変数, 155, 178  
EBMSOCKDIR 環境変数, 178  
ebmsys\_authority\_file ファイル, 169  
EBM\_TIME\_FORMAT 環境変数, 172  
EBMTMPDIR 環境変数, 184  
elgfile ファイル, 136  
EXCI, 72

## F

File\_Map  
FILEMAP 環境変数, 184  
GDG, 96  
更新, 96  
定義, 96  
FILEMAP 環境変数, 154, 156, 184  
FORMS 環境変数, 153, 186  
fsusrexit.c モジュール, 188  
ftp コマンド, 162, 164

## H

HANDLE\_MFVS 環境変数, 184  
HISTORY 環境変数, 153  
histprt コマンド, 24

## I

IDCAMS REPRO メインフレームオプション, 76  
.install ファイル  
内容, 167  
例, 173  
ishp ディレクトリ, 84  
ish ディレクトリ, 84

## J

Java、選択, 69  
JCL  
MVS メインフレームオプション, 75  
OUTPUT 文, 78  
RLS オプション, 80  
SYSOUT 文, 153  
VSE メインフレームオプション, 75, 81  
変換されたジョブディレクトリ, 84  
変換されたプロシージャディレクトリ, 84

## K

kixjob コマンド, 155  
KIXLICDIR 環境変数, 178  
KIXSYS 環境変数, 84, 154, 184

## L

LC\_TIME 環境変数, 180  
LD\_LIBRARY\_PATH 環境変数, 185  
lgdem デーモン, 136  
LPI\_PRODUCT\_DIR 環境変数, 185  
lststs コマンド, 24

## M

MAXCC メインフレームオプション, 79  
MQSERIES 環境変数, 201  
MQSERVER 環境変数, 201  
MVS JCL  
遠隔ジョブ, 164  
メインフレームオプション, 75  
MVSTRANS\_OPT 環境変数, 165  
mvstrans コマンド, 164, 165

## O

Oracle, 189  
OUTPUT 文のメインフレームオプション, 78

## P

PACK 環境変数, 178  
pack ディレクトリ, 167  
PATH 環境変数, 178  
PL/I  
    LPI\_PRODUCT\_DIR 環境変数, 185  
    選択, 69  
    日付/時刻, 73  
PROCLIB 環境変数, 156  
PUBLIC 環境変数, 178  
public ディレクトリ, 167

## R

RDBMS  
    回復プロシージャ, 188  
    環境設定, 84  
    環境変数, 186  
    サポートモジュール  
        fsusrexite.c, 188  
        runfs.c, 188  
    モジュールのカスタマイズ, 187  
RDBMS\_CONNECT 環境変数, 186  
REPRO  
    パラメータ, 77, 81  
    メインフレームオプション, 76  
RLS メインフレームオプション, 80  
RTS\_RDBMS 環境変数, 186  
RUNBDIR 環境変数, 185  
RUNBPATH 環境変数, 154, 185  
runfs.c モジュール, 188

## S

Server Express の COBDIR 変数, 184  
SORT\_MODE 環境変数, 185  
subjob コマンド  
    デフォルトのサブシステム, 98  
    デフォルトのジョブ履歴ディレクトリ, 168  
Sun Mainframe Transaction Processing。 「Sun MTP」 を参照

Sun MBM コンソール機能, 20  
Sun MBM サブシステム。「サブシステム」 を参照  
Sun MBM ノード。「ノード」 を参照  
Sun MBM の概念, 1  
Sun MTP  
    kixjob コマンド, 155  
    VSAM 環境設定, 84  
    VSAM ファイル, 199  
    一時データキュー, 155  
    開始キュー処理プログラム, 197  
    環境設定, 83  
    環境変数, 197  
    起動日, 36  
    索引編成ファイル, 71  
    サブシステムとの通信, 197  
    トランザクション処理プログラム, 197  
    バッチジョブのサブミット, 155  
    日付/時刻の構成ユーティリティ, 35  
    日付/時刻の同期化, 32  
    領域とサブシステムの再同期, 94  
    領域の変更, 94  
Sun MTP との通信, 197  
Sun 以外のパッケージ、選択, 72  
Sybase, 65, 193  
SyncSort, 72, 185  
SYSINDIR 環境変数, 178  
sysindir ディレクトリ, 168  
SYSOUTDIR 環境変数, 153, 186  
SYSOUT 文, 153

## T

tty コマンド, 22, 170

## U

unikixjob コマンド  
    「subjob コマンド」 も参照  
VSAM データセット, 198  
デフォルトのサブシステム, 98

デフォルトのジョブ履歴ディレクトリ, 168  
例, 161  
UNIXIX 環境変数, 186

## V

VCT バッチパラメータ, 199  
VSAM  
Sun MTP データセットへのアクセス, 198  
環境設定, 84  
VSAM 構成テーブル (VCT), 199  
VSAM データセットへのアクセス, 198  
VSAM ファイル, 71, 199  
VSE JCL  
遠隔ジョブ, 164  
メインフレームオプション, 75, 81

## W

WebSphere MQ  
環境変数, 201  
サブシステムの作成, 202  
前提条件, 201

## あ

アカウントティング、ジョブ。「ジョブアカウン  
ティング」を参照  
アクセス制御。「ユーザーの権限」または「コマ  
ンド制御」を参照  
アクティビティー  
一覧表示, 128  
数の変更, 132  
最大数, 169  
最大数の指定, 132  
削除, 117  
作成, 117, 130  
作成/削除の権限, 117, 118  
アプリケーション  
言語、選択, 69  
プログラム、環境の設定, 84

## い

一時データキュー, 155

## え

エラーログファイル, 136  
遠隔ジョブ、ftp によるサブミット, 162  
遠隔ノード  
.install ファイル, 169  
一覧表示, 44  
削除, 51  
設定, 48  
変更, 50

## か

環境ファイル  
\$HOME/.btshrc, 151  
\$HOME/.post\_process.btsh, 152  
\$PACK/btshrc, 151  
\$PUBLIC/bin/post\_process.btsh, 152  
プリプロセス, 152  
ポストプロセス, 152  
環境変数  
ACUCOBOL, 184  
A\_TERMCAP, 184  
COBDIR, 184  
DEVNULL, 178  
DOSTRANS\_OPT, 165  
DUMPCORE, 178  
DUMPDIR, 178  
DUMPSHMEM, 178  
EBM\_DATE\_FORMAT, 171  
EBMHOME, 155, 178  
EBMSOCKDIR, 178  
EBM\_TIME\_FORMAT, 172  
EBMTMPDIR, 184  
FILEMAP, 154, 156, 184  
FORMS, 153, 186  
HANDLE\_MFVS, 184  
HISTORY, 153  
KIXLICDIR, 178  
KIXSYS, 84, 154, 184  
LC\_TIME, 180  
LD\_LIBRARY\_PATH, 185

- LPI\_PRODUCT\_DIR, 185
- MQSERIES, 201
- MQSERVER, 201
- MVSTRANS\_OPT, 165
- PACK, 178
- PATH, 178
- PROCLIB, 156
- PUBLIC, 178
- RDBMS, 186
- RDBMS\_CONNECT, 186
- RTS\_RDBMS, 186
- RUNBDIR, 185
- RUNBPATH, 154, 185
- SORT\_MODE, 185
- Sun MTP, 197
- SYSINDIR, 178
- SYSOUTDIR, 153, 186
- UNIKIX, 186
- WebSphere MQ, 201
- 実行順序, 151
- ジョブの実行, 154
- 完了したジョブ、一覧表示, 17

## き

- 起動日
  - Sun MBM, 37
  - Sun MTP, 36
- 共有メモリー、表示, 145
- 許可ファイル, 169
  - クラスとアクティビティーの削除, 118
  - クラスとアクティビティーの作成, 117
  - 形式, 169
  - サブシステムの管理, 115
  - タイプ, 169
  - ノードの管理, 114
  - ノードの起動, 111
  - ノードの停止, 113

## く

- クラス
  - アクティビティー数, 132
  - 一覧表示, 128

- 削除, 117, 133
- 作成, 117, 130
- 作成/削除の権限, 117, 118

## け

- 形式設定指示, 58, 59
- 言語、選択, 69
- 現在の日付, 32
- 現在のロケール時間カテゴリ, 172

## こ

### 構成

- BAM の使用, 5
- .install ファイル, 167
- タスク, xxiii
- ログ環境, 138

### 構成ファイル

- batchenv, 177
- .install, 167
- サブシステム設定, 183
- プリプロセス, 153
- ポストプロセス, 153

### コマンド

- bam, 9
- dostrans, 164, 165
- ftp, 162, 164
- histprt, 24
- kixjob, 155
- lststs, 24
- mvstrans, 164, 165
- subjob, 98, 168
- tty, 22, 170
- unikixjob, 98, 161, 168, 198
- アクセス制御, 110
- 出力の制御, 179
- ロケール時間によって影響を受ける  
コマンド, 180

### コマンド制御

- クラスとアクティビティーの削除, 118
- クラスとアクティビティーの作成, 117
- サブシステムを管理可能なユーザーの定義, 115

- ノードの起動, 111
- ノードの停止, 113
- ノードを管理可能なユーザーの定義, 114
- メニューの表示, 110
- コマンドへのアクセス制御, 110
- コンソール
  - 現在の設定, 22
  - 識別子、変更, 26
  - 端末
    - 無効化, 23
    - 有効化, 22
  - ファイル, 23, 170
    - \$SYSINDIR/ebm\_console, 20
    - 無効化, 24
    - 有効化, 23
  - メッセージタイプ, 170
  - 割り当て, 20, 21

## さ

- 作業用ディレクトリ, 119, 169
- サブシステム
  - 「環境ファイル」も参照
  - File\_Map, 96
  - Sun MTP への接続, 84, 94, 197
  - Sun MTP 領域との再同期, 94
  - アクセス制御, 108
  - アプリケーション言語, 69
  - インポート, 104
  - 更新
    - File\_Map, 96
    - ジョブ履歴ディレクトリ, 93
    - ユーザー設定ファイル, 98
  - 構成の表示, 90
  - 再作成, 102
  - 削除, 100
  - 作成, 67~85
  - 使用可能なサブシステムの一覧表示, 66
  - 状態, 14
  - ジョブ設定ファイル, 151
  - セキュリティー。「セキュリティー」を参照
  - 設定ファイル(\$SETUP), 183
  - 説明, 2

- 追跡, 140
- データベース, 70
- デフォルト, 98
- デフォルト設定へのリセット, 82
- 特性の表示, 89
- ノードごとの最大数, 63
- 表示
  - File\_Map, 91, 92
  - ジョブ, 91
  - ジョブ履歴ディレクトリ, 91
  - 設定内容, 82
  - 設定ファイル, 91
  - プリプロセスファイル, 153
  - ポストプロセスファイル, 153
  - 有効なサブシステムの一覧表示, 15
  - ユーザー設定ファイル, 151
- サブシステムのインポート, 104
- サブシステムの再作成, 102
- サブシステムの作成, 67

## し

- 時刻
  - 形式設定指示, 38
  - 形式の変更, 39
- 指示
  - 時刻, 38
  - ジョブアカウンティング, 60
  - 日付, 38
- システム出力ジョブディレクトリ, 84
- システムのスナップショット, 148
- システム日付、設定, 37
- システムメッセージ, 20
- 実行環境、ジョブ, 151
- 循環ファイル
  - 「Admlog 機能」も参照
  - 「Sun MBM コンソール」も参照
  - Admlog 機能, 139
  - 履歴ファイル, 24
  - ログファイル, 139

## 障害追跡

- BAM ログファイル, 150
- システムのスナップショット, 148
- テストの実行, 149

## 状態

- サブシステム, 14
- ジョブアカウンティング, 55

## ジョブ

- アクティビティー
- 一覧表示, 128
- 作成, 130

- 一覧表示
- 完了した, 17
- 有効な, 16

- 遠隔、ftp によるサブミット, 162

## クラス

- アクティビティー数, 132
- 一覧表示, 128
- 削除, 133
- 作成, 130, 131

- 設定ファイル, 151

- 番号、初期ジョブ番号の割り当て, 28

- ログファイルの保存, 18

## ジョブアカウンティング

- 形式設定指示, 58, 59
- 現在の設定の表示, 55
- 設定, 53
- 定義, 53
- デフォルト設定, 57
- ファイル, 53, 172
- ファイル、変更, 58
- 無効化, 57
- メニューへのアクセス, 55
- 有効化, 56
- レコード形式, 173
- レコード形式、変更, 58

- ジョブの実行環境、設定, 151

- ジョブの履歴ファイル, 168

- ジョブの履歴ファイル、デフォルトディレクトリの変更, 27

## ジョブのログ

- 消去, 19
- 保存, 18

- 保存されたログの表示, 20

- ジョブ番号、初期ジョブ番号の割り当て, 28

- ジョブ履歴ディレクトリ, 168

- ジョブ履歴ディレクトリ、更新, 93

## す

- スナップショット, 148

## せ

### セキュリティ

- アクセス制御, 109
- 許可ファイル, 169
- コマンドへのアクセス, 110
- サブシステムの管理, 115
- 設定の要件, 107
- ノードの管理, 114
- ノードの起動, 111
- ノードの停止, 113

- 世代別データグループ (GDG) ファイルタイプ, 96

### 設定内容

- サブシステムの表示, 82

### 設定ファイル

- 「batchenv ファイル」も参照
- \$USER\_SETUP のカスタマイズ, 97
- サブシステム, 151
- 定義済み, 97
- ユーザー, 151

## そ

- ソケットファイル, 178

## た

- 代替ジョブ履歴ディレクトリ, 168

- ダンプファイルディレクトリ, 178

### 端末コンソール

- 無効化, 23
- 有効化, 22

## つ

### 追跡

- サブシステム, 140
- プロセス, 142
- メッセージ, 143

## て

定義済み設定ファイル, 97

ディスク容量、確認, 146

### ディレクトリ

- autosub, 162
  - \$EBMHOME/public/ish, 155
  - \$EBMHOME/public/jmvs, 155
  - ish, 84
  - ishp, 84
  - pack, 167
  - \$PACK/bin, 178
  - \$PACK/RTSFS, 188
  - public, 167
  - sysindir, 168
  - \$UNIKIX/bin, 155
  - アプリケーション構造の例, 4
  - 遠隔ジョブのサブディレクトリ, 162
  - 作業用, 119, 120, 169
  - システム出力, 84
  - ジョブの履歴, 168
  - 代替ジョブ履歴, 168
  - ダンプファイル, 178
  - ユーザーの作業用, 119, 120, 169
- データファイルのタイプ、選択, 71
- データベース、選択, 70
- ### デーモン
- autosub, 164
  - lgdem, 136
- デフォルトのジョブ履歴ディレクトリ, 168

## と

ドメインソケットファイル, 178

トランザクション処理プログラム, 197

## な

内部ダンプ, 178

## の

### ノード

- .install ファイル, 167
  - 一覧表示, 44
  - 遠隔
    - 一覧表示, 44
    - 削除, 51
    - 設定, 48
    - 変更, 50
    - ポート番号, 49, 50
  - 起動, 12
  - 構成ファイル, 167
  - 設定, 11 ~ 27
  - 定義, 1
  - 停止, 13
  - 日付/時刻と Sun MTP 領域の同期化, 32
  - 複数の, 46
  - プロセスの追跡, 142
  - ポート番号
    - 遠隔, 49
    - ローカル, 46
  - ユーザーの権限
    - クラスとアクティビティの削除, 118
    - クラスとアクティビティの作成, 117
    - 制御, 110
    - ノードの管理, 114
    - ノードの起動, 111
    - ノードの停止, 113
  - ローカル
    - サーバーの削除, 48
    - サーバーの設定, 46
    - 説明, 44
    - ポート番号, 46
- ### ノード間通信
- 遠隔ノードの削除, 51
  - 遠隔ノードの設定, 48
  - 遠隔ノードの変更, 50
  - 使用可能なノードの一覧表示, 44
  - ノードサーバー指定の削除, 48

ノードのサーバー指定, 46  
複数のノードの設定, 46  
ノードの停止, 13

## は

バッチジョブ、オンライントランザクションからのサブミット, 155  
バッチノード、設定, 11 ~ 29

## ひ

日付  
Sun MBM の起動, 37  
Sun MTP の起動, 36  
形式の変更, 38  
現在の日付/時刻の表示, 37  
指示, 38  
システム, 37  
システムのリセット, 37  
設定  
Sun MBM, 35  
Sun MBM と Sun MTP に別個の日付, 36  
Sun MTP, 35  
ファイル, 32  
日付/時刻  
COBOL, 73  
PL/I, 73  
管理, 35  
形式の変更, 38  
現在の日付/時刻の表示, 37  
指示, 38  
設定, 35  
日付/時刻と Sun MTP 領域の同期化, 32

## ふ

ファイル  
「環境ファイル」も参照  
bam\_authority\_file, 169  
bam.log, 85, 150  
batchenv。「batchenv ファイル」を参照

batch\_shut\_authority\_file, 169  
batch\_start\_authority\_file, 169  
crtact\_authority\_file, 169  
dltact\_authority\_file, 169  
ebm\_console, 170  
ebmsys\_authority\_file, 169  
elgfile, 136  
HistoryFile, 20, 170  
\$HOME/.btshrc, 152  
.install  
内容, 167  
例, 173  
\$KIXSYS/datefile, 32  
\$KIXSYS/unikixmain.err, 156  
\$PACK/btshrc, 152  
Sun MBM コンソール, 170  
\$SYSINDIR/ebm\_console, 20, 23, 170  
\$SYSINDIR/HistoryFile, 20, 170  
エラーログ, 136  
許可, 111, 113, 114, 115, 117, 118, 169  
コンソール, 23, 24  
索引編成, 71  
循環, 23  
ジョブアカウンティング, 53, 58, 172  
ジョブ設定, 151  
ジョブのログ, 18, 19, 20  
セキュリティ, 169  
設定, 151  
設定 (\$SETUP), 183  
ソケット, 178  
日付, 32  
プリプロセス, 153  
ポストプロセス, 153  
ユーザー固有オブジェクト, 74  
ユーザー設定, 97, 98, 151  
履歴, 20, 25, 26, 168, 170  
ログのサイズ, 137  
プリプロセスファイル, 153  
プロセスの追跡, 142

## ほ

- ポート番号
  - 遠隔ノード, 49
  - ローカルノード, 46
- ポストプロセスファイル, 153

## め

- メインフレーム互換性
  - MAXCC 値, 79
  - MVS OUTPUT 文のオプション, 78
  - MVS REPRO オプション, 76
  - MVS RLS オプション, 80
  - VSE REPRO オプション, 81
- メッセージ
  - Sun MBM システム, 20
  - コンソール
    - 接頭辞, 170
    - タイプ, 170
  - 終了, 20
  - ジョブの開始, 20
  - 追跡, 143
  - 履歴ファイル, 170
  - レベル, 23, 25
  - ログの表示, 143
  - ログの無効化, 145
  - ログの有効化, 143
- メッセージのルーティング, 20

## も

- 問題の特定
  - サブシステムの追跡, 140
  - メッセージ追跡の表示, 144
  - メッセージの追跡, 143
  - メッセージログの無効化, 145
  - メモリーのダンプ, 145

## ゆ

- 有効なサブシステム、一覧表示, 15
- 有効なジョブ、一覧表示, 16

- ユーザー環境、設定, 125
- ユーザー環境の設定, 125
- ユーザー固有
  - オブジェクト, 74
  - 環境変数, 152
- ユーザー設定ファイル、更新, 98
- ユーザーの権限
  - クラスとアクティビティの削除, 118
  - クラスとアクティビティの作成, 117
  - サブシステムの管理, 115
  - ノードの管理, 114
  - ノードの起動, 111
  - ノードの停止, 113
- ユーザーの作業用ディレクトリ
  - .install ファイル, 169
  - アクセス, 120
  - 一覧表示, 121
  - 管理, 119
  - 削除, 124
  - 作成, 122
  - 変更, 123

## ら

- ライセンス、ソフトウェア, 5
- ライセンスファイルの環境変数, 178

## り

- 領域
  - サブシステムとの再同期, 94
  - 日付/時刻と Sun MBM の同期化, 33
- リレーショナルデータベース管理システム。  
「RDBMS」を参照
- 履歴ファイル
  - サイズ, 170
  - 循環, 24, 170
  - 説明, 20
  - デフォルト, 168
  - 表示, 24
  - 無効化, 26
  - メッセージ, 170

メッセージレベル, 25  
有効化, 25

## れ

レコード形式、ジョブアカウンティング, 58, 173

## ろ

ローカルノード

  .install ファイル, 169

  一覧表示, 44

  サーバーの削除, 48

  設定, 46

ログデーモン, 136

ログファイル

  エラー, 136

  サイズの変更, 137

  ジョブの消去, 19

  表示, 136

  複数のシリアルファイル, 139

ログファイルのデバッグ, 136

ロケール時間, 180