

FLEXNET LICENSING

エンドユーザガイド



製品バージョン 11.4
ドキュメント改訂 01

Copyright Notice

Copyright © 2006 Macrovision Corporation and/or Macrovision Europe Ltd. All Rights Reserved.

This product contains proprietary and confidential technology provided by and owned by Macrovision Europe Ltd., UK, and Macrovision Corporation of Santa Clara, California, U.S.A. Any use, copying, publication, distribution, display, modification, or transmission of such technology in whole or in part in any form or by any means without the prior express written permission of Macrovision Europe Ltd. and Macrovision Corporation is strictly prohibited. Except where expressly provided by Macrovision Europe Ltd. and Macrovision Corporation in writing, possession of this technology shall not be construed to confer any license or rights under any of Macrovision Europe Ltd. and Macrovision Corporation's intellectual property rights, whether by estoppel, implication, or otherwise.

ALL COPIES OF THE TECHNOLOGY and RELATED INFORMATION, IF ALLOWED BY MACROVISION CORPORATION, MUST DISPLAY THIS NOTICE OF COPYRIGHT AND OWNERSHIP IN FULL.

Trademarks

Macrovision, AdminStudio, DemoNow, DemoShield, FLEXenabled, FLEX/m, FLEXnet, FLEXnet Certified, FLEXnet Connector, FLEXnet Manager, FLEXnet Publisher, *Globetrotter*, Hawkeye, InstallFromTheWeb, InstallShield, InstallShield Developer, InstallShield DevStudio, InstallShield Professional, It All Starts Here, OneClickInstall, Package For The Web, QuickPatch, ReadySell, RipGuard, SafeCast, Trymedia Systems, and Zero G Software are registered trademarks or trademarks of Macrovision Corporation in the United States of America and/or other countries. All other brand and product names mentioned herein are the trademarks and registered trademarks of their respective owners.

Restricted Rights Legend

The software and documentation are "commercial items," as that term is defined at 48 C.F.R. §2.101, consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.2702, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §227.2702-1 through 227.2702-4, as applicable, the commercial computer software and commercial computer software documentation are being licensed to U.S. government end users *ÀiAAj* only as commercial items and *ÀiBAj* with only those rights as are granted to all other end users pursuant to the terms and conditions set forth in the Macrovision Corporation standard commercial agreement for this software. Unpublished rights reserved under the copyright laws of the United States of America.

Disclaimer

Information in this document is subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. The provision of such information does not represent any commitment on the part of Macrovision Corporation. Macrovision makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Macrovision shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

June 2006

目次

はじめに	9
ドキュメントの表記規約.....	11
Macrovision へようこそ	13
Macrovision ソリューション.....	13
Macrovision プロフェッショナル サービス.....	15
テクニカル サポート リソース.....	17
Macrovision の連絡先.....	18
1 ライセンス ファイルの基本情報	21
ライセンス ファイルの場所を指定する.....	21
環境変数でパスを設定する.....	23
ライセンス ファイル フォーマットの概要.....	24
ライセンス ファイルの種類.....	24
フローティング (同時使用) ライセンス.....	25
ノードロック ライセンス.....	25
ノードロック ライセンスとフローティング ライセンスの混在.....	26
2 複数ベンダーのライセンスの管理	27
複数ライセンスの管理方式の概要.....	27
複数のマシン.....	28

ライセンス サーバ システムを起動する	29
複数のライセンス サーバ システムを持つ 1 台のマシン	29
ライセンス サーバ システムを起動する	30
1 つのライセンス サーバ システムと複数のライセンス ファイルを持つ 1 台のマシン	31
ライセンス サーバ システムを起動する	32
複数のライセンス ファイルを管理する	32
その他の検討項目	33
ライセンス ファイルを結合する	33
ライセンス サーバ システムを起動する	34
ライセンス ファイルを結合する規準	35
ライセンス ファイルの結合方法	35
コンポーネントのバージョン互換性	36
3 ライセンス サーバ マシンの選択	37
サーバが使用するリソース	37
ソケット	37
CPU 時間	38
ディスク スペース	38
メモリ	38
ネットワークの帯域幅	39
リモートにマウントされているディスク	39
冗長構成のライセンス サーバ システム	39
ライセンスファイル リストによる冗長構成	40
3 サーバによる冗長構成	41
3 サーバによる冗長構成とライセンスファイル リストによる冗長構成を比較する ..	41
カウントされるライセンスとカウントされないライセンス	42
4 ライセンス管理ツール	43
管理ツールを実行する	44
lmutil の汎用引数	45
lmborrow	45
借用を開始する	46
借用中のライセンスの設定をクリアする	47
借用ライセンスのステータスを確認する	47
借用したライセンスを期限前に返却する	47
lmdiag	48

lmdown	50
lmhostid	51
lminstall	53
lmnewlog	54
lmpath	54
lmremove	56
lmreread	57
lmstat	59
lmswitch	61
lmswitchr	62
lmver	63
ライセンス管理ツール – Windows 用の LMTOOLS	63
ライセンス ファイルを使用する設定	64
サービスを使用する設定	64
5 オプション ファイル	65
オプション ファイルを作成する	66
オプション ファイルの構文	67
BORROW_LOWWATER	70
DEBUGLOG	71
EXCLUDE	72
EXCLUDE_BORROW	72
EXCLUDE_ENTITLEMENT	73
EXCLUDEALL	74
FQDN_MATCHING	75
GROUP	77
GROUPCASEINSENSITIVE	77
HOST_GROUP	78
INCLUDE	78
INCLUDE_BORROW	79
INCLUDE_ENTITLEMENT	80
INCLUDEALL	81
LINGER	81
MAX	82
MAX_BORROW_HOURS	83
MAX_OVERDRAFT	84

NOLOG	84
REPORTLOG.....	85
LM_PROJECT を使用してプロジェクトについてレポートする	85
RESERVE	85
TIMEOUT	86
TIMEOUTALL	87
ベンダー デーモンがオプション ファイルを使用するプロセス.....	88
オプション ファイルにおける優先順位の規則.....	88
オプション ファイルの例.....	89
単純なオプション ファイルの例.....	89
複数のユーザに対してアクセスを制限する	89
EXCLUDE の例	90
EXCLUDE_ENTITLEMENT の例.....	91
INCLUDE の例	91
INCLUDE_ENTITLEMENT の例	92
6 モバイル ライセンス管理.....	93
ラップトップ コンピュータにノードロック.....	94
FLEXid にノードロック (Windows のみ)	94
FLOAT_OK 付きの FLEXid にノードロック (Windows のみ)	94
FLOAT_OK 付きの FLEXid を開始する	95
FLOAT_OK 付きの FLEXid を返却する	96
FLOAT_OK 付きの FLEXid の例	96
BORROW によるライセンス借用.....	97
ライセンスの借用を開始する	97
アプリケーション インタフェース.....	98
lmborrow ユーティリティを起動する.....	98
環境変数 LM_BORROW を直接設定する.....	98
ライセンスを借用する	99
借用期間をクリアする	100
借用のステータスをチェックする	100
借用したライセンスを期限前に返却する	100
ライセンス借用に対するサポート	101
ユーザ名にノードロック.....	102
プリペイド ライセンス プールからのライセンス発行	102

7	FLEXnet Licensing がサポートするプラットフォームの ホスト ID ..	103
	ホスト ID のフォーマット.....	103
	FLEXnet Licensing のホスト ID	104
	特殊な FLEXnet Licensing ホスト ID.....	106
8	ライセンス ファイルのフォーマット	109
	ライセンス ファイルの構文	110
	SERVER 行	110
	3 サーバによる冗長構成.....	112
	VENDOR 行	114
	USE_SERVER 行	115
	FEATURE 行と INCREMENT 行	116
	優先順位	120
	PACKAGE 行	121
	UPGRADE 行	124
	10 進数のフォーマット.....	125
	ライセンス ファイルの行順	125
9	ライセンス サーバ マネージャ	127
	lmgrd コマンドライン構文	127
	UNIX プラットフォームでライセンス サーバ マネージャを起動する	129
	手動起動	129
	自動起動	130
	Windows でライセンス サーバ マネージャを起動する	131
	コマンド ラインから手動で起動する.....	131
	Windows サービスとしてライセンス サーバ マネージャを設定する	132
	LMTOOLS から手動で起動する	133
	システム起動時に自動的に起動する	135
A	トラブルシューティング ガイド	137
	一般的なトラブルシューティングのヒント	137
	FLEXLM_DIAGNOSTICS.....	138
	レベル 1 の内容	139
	レベル 2 の内容	139
	レベル 3 の内容 (バージョン 6.0 以降のみ).....	140

B	FLEXnet Licensing 環境変数	141
	環境変数の設定方法	141
	レジストリ	141
	優先順位	142
	環境変数	142
C	FLEXnet Licensing のエラー コード	145
	エラー メッセージのフォーマット	145
	フォーマット 1 (短文)	146
	フォーマット 2 (長文、バージョン 6 以降)	146
	エラー コードの説明	147
D	レポート ログ ファイル	157
	レポート ログ出力を管理する	157
	ベンダー デーモンのレポート ログ出力を有効にする	158
	ベンダー デーモンのレポート ログ出力先を変更する	158
E	デバッグ ログ ファイル	159
	デバッグ ログ出力を管理する	159
	ライセンス サーバ システムのデバッグ ログ出力をキャプチャする	160
	特定のベンダー デーモンのデバッグ ログ出力をキャプチャする	160
	実行中のベンダー デーモンのデバッグ ログの出力先を変更する	160
	ベンダー デーモンのデバッグ ログ出力を制限する	161
	デバッグ ログ メッセージ	161
	情報メッセージ	162
	設定の問題のメッセージ	163
	デーモン ソフトウェアのエラー メッセージ	164
F	FLEXnet Licensing のバージョン	167
	FLEXnet Licensing コンポーネントのバージョンの互換性	167
	ライセンス ファイルのバージョンの見分け方	168
	バージョンのまとめ	168
	索引	175

はじめに

このマニュアル、『FLEXnet Licensing エンド ユーザ ガイド』では、FLEXnet Licensing の基本概念、ライセンス モデル、ライセンス ポリシー設計の概念を説明します。本書は、以下の章から構成されています。

表 1: ヘルプ参照一覧

トピック	内容
「はじめに」	『FLEXnet Licensing エンド ユーザ ガイド』の概要を説明します。ヘルプ ライブラリの表記規則と使用法が記載されています。
「Macrovision へようこそ」	Macrovision Corporation の概要を説明します。テクニカルサポート、オンラインコミュニケーション、他の Macrovision 製品などが記載されています。
第 1 章. 「ライセンス ファイルの基本情報」	ライセンス ファイルと正しい使用法の基本情報が記載されています。
第 2 章. 「複数ベンダーのライセンスの管理」	ライセンス ファイル管理の概要を説明します。異なるいくつかのシナリオ、可能なライセンス管理方式などが記載されています。
第 3 章. 「ライセンス サーバマシンの選択」	この章では、ライセンス サーバ マシンとしてどのマシンを使うべきかを判断する手がかりを提供します。
第 4 章. 「オプション ファイル」	オプション ファイルを使用して、ライセンス管理者が FLEXnet Licensing のさまざまな操作パラメータをコントロールする方法を説明します。

表 1: ヘルプ参照一覧

トピック	内容
第 5 章. 「 ライセンス管理ツール 」	異なるライセンス管理ユーティリティ、およびそれらの使用法を説明します。
第 6 章. 「 モバイル ライセンス管理 」	各種モバイル ライセンスの正しい使用法を説明します。
第 7 章. 「 FLEXnet Licensing がサポートするプラットフォームのホスト ID 」	FLEXnet Licensing がサポートする各種プラットフォームに関する情報を提供します。
第 8 章. 「 ライセンス ファイルのフォーマット 」	ライセンス ファイルのフォーマットおよび FLEXnet Licensing で使用する構文に関する情報を提供します。
第 9 章. 「 ライセンス サーバマネージャ 」	ライセンス サーバ管理について説明します。正しいコマンドライン構文、および異なるプラットフォームでライセンス サーバマネージャを起動する方法が記載されています。
付録 A. 「 トラブルシューティングガイド 」	FLEXnet Licensing の基本的なトラブルシューティング情報を提供します。
付録 B. 「 FLEXnet Licensing 環境変数 」	FLEXnet Licensing のすべての環境変数およびその定義の一覧を提供します。それらの設定方法も記載されています。
付録 C. 「 FLEXnet Licensing のエラーコード 」	FLEXnet Licensing が出力するすべてのエラーコードを示します。
付録 D. 「 レポート ログ ファイル 」	レポート ログ ファイルの使用法、およびその出力の管理方法を説明します。
付録 E. 「 デバッグ ログ ファイル 」	デバッグ ログ ファイルの使用法、およびその出力の管理方法を説明します。
付録 F. 「 FLEXnet Licensing のバージョン 」	各種モバイル ライセンスのサポート状況や、ノードロックライセンスのサポート状況など、FLEXnet Licensing がどのようにライセンスをサポートしているかを示します。








ドキュメントの表記規約

このドキュメントでは、特定の情報に注意を引き付け、読者が情報を見分ける手助けとするために、以下の表記規約を使用します。

行頭アイコンの規約

行頭アイコンは、主要な情報か補足的な情報かを示すために、このドキュメント全体にわたって使用されています。次の表に、各アイコンの意味を示します。

表 2: 行頭アイコンの規約

イメー ジ	アイコン名	説明
	注	「注」は、注意を引き付けて情報を際立たせるために使用されます。
	重要な注	「重要な注」は、ユーザが読むべき主要な情報に対して使用されます。
	注意	「注意」は、希望の機能や製品の機能を使用するために重要な情報であることを示します。
	ヒント	「ヒント」は、希望の機能を使用するうえで手助けとなる情報を示すために使用されます。
	最善の方法	「最善の方法」は、目的を達成するための最善の方法を示します。
	セキュリティ	「セキュリティ」は、セキュリティの問題を示します。
	操作方法	「操作方法」は、操作方法の説明の開始を示します。

表記上の規約

このドキュメント全体にわたって、以下の表記上の規約が使用されています。

表 3: 表記上の規約

スタイル	例	説明
ユーザインタフェース要素	[ファイル]メニューから 開く を選択します。	ユーザインタフェース要素は、文中ではゴシック体で表記します。
変数	<i>filename</i>	変数は、斜体で表記します。
コード	#define HWND_BROADCAST 0xffff	コードは、等幅フォントで表記します。
ユーザ入力文字	\$D(install) と入力します。	記載されているとおりに入力する文字列は、等幅、太字、青色で表記します。
ファイル名とフォルダパス	ユーザファイルは、 C:¥MyDocuments¥SampleCode フォルダに保存されています。	ファイル名とフォルダパスは、等幅フォントで表記します。
コマンドライン文	サイレントインストールを実行するには、次のように入力します: Setup.exe /s /v/qn	コマンドライン文とパラメータは、等幅フォントで表記します。
環境変数	環境変数 windir の値を設定します。	環境変数は、等幅フォントで表記します。
例	2つのグループ(たとえば、Admins と General)を作成します。	例は、太字体で表記します。
関数	FeatureAddItem は、スクリプトで作成された機能セットに新しい機能を追加します。	関数は、太字体で表記します。
プロパティ	Name プロパティに、プロジェクト内のすべてのコントロールに対して一意のカスタムコントロール名を入力します。	プロパティは、太字体で表記します。
画面出力	間違ったパラメータを入力すると、「 The system cannot find the path specified. 」というメッセージが表示されます。	画面出力(ログファイルまたはコンソールから)は、等幅、太字体で表記します。

Macrovision へようこそ

会社情報

Macrovision Corporation は、エレクトロニック ライセンシング、インストール、およびデジタル著作権管理 (DRM) 技術におけるマーケットリーダーとして活躍する企業です。ソフトウェアの価値を最大限に引き出すことができる Macrovision のテクノロジーは、5 万を超えるソフトウェア ベンダーと、ほぼすべての Fortune 1000 社によって利用されています。ソフトウェア バリュー マネージメント ソリューションは、開発側におけるソフトウェアの価格設定とパッケージングのギャップ、企業側におけるソフトウェアの購入と管理のギャップを埋めるために開発されました。Macrovision が取り扱っている FLEXnet ソフトウェア バリュー マネージメント プラットフォームには、ソフトウェア インストール、リパッケージング、アップデート ソリューションからなる InstallShield スイートが含まれています。これらのソリューションは、世界各地で 5 億台以上のデスクトップで利用されています。Macrovision は、全世界において 910 を超えるソフトウェア ライセンシング、DRM、およびコンテンツ プロテクションの特許を所有しています。Macrovision は、カリフォルニア州サンタクララに本社を持ち、その他世界各地にオフィスがあります。

Macrovision ソリューション

ソフトウェアの価値を最大限に引き出す

ソフトウェア バリュー マネージメントとは、ソフトウェア ベンダーおよびそのエンタープライズ ユーザーが、作成、使用、保守においてソフトウェア アプリケーションの価値を最大限に引き出すことができる 1 セットのベスト プラクティスです。

アプリケーションがひとたびエンジニアリング部門の開発者たちの手を離れると、パブリッシャはソフトウェアバリュー マネージメント ツールを利用して収入の最大化を図ることができます。これらのツールには、パブリッシャが製品を柔軟にパッケージ、価格設定、保護できるように支援するインストーラやライセンシング ツールなどがあります。エンタープライズ ユーザはソフトウェアバリュー マネージメント ツールを利用して、購入したソフトウェアから最大限の生産性を引き出すことができます。これらのツールを利用して、企業の IT スタッフは、アプリケーションの再パッケージ、潜在的に起こりうる競合の解決、ライセンス購入の最適化、およびアップデートの管理を行うことができます。

表 1: Macrovision ソフトウェア テクノロジグループ ソリューション

業界とアクティビティ	FLEXnet ソフトウェア バリュー マネージメント ソリューション
ソフトウェア ベンダー <ul style="list-style-type: none">● ソフトウェアのマーケティング● ソフトウェアの販売● ソフトウェアの配布● ソフトウェアのサービス● ソフトウェアの更新	<ul style="list-style-type: none">● InstallShield: あらゆる OS で使用できるインストール作成およびソフトウェア マネジメントのツールです。● FLEXnet Publisher: 製品の全ライフサイクルを通じて、価格設定、パッケージ、自社製品の保護およびソフトウェア ライセンスの管理を行い、市場のニーズにより的確に応え、収益の最大化を図ります。
エンタープライズ IS と IT: <ul style="list-style-type: none">● ソフトウェアの購入● ソフトウェアの作成と配布● ソフトウェアの保守● ソフトウェアの更新	<ul style="list-style-type: none">● InstallShield: あらゆる OS で使用できるインストール作成およびソフトウェア マネジメントのツールです。● FLEXnet Publisher: 製品の全ライフサイクルを通じて、価格設定、パッケージ、自社製品の保護およびソフトウェア ライセンスの管理を行い、市場のニーズにより的確に応え、収益の最大化を図ります。● FLEXnet AdminStudio: エンタープライズ向けのアプリケーションパッケージング、カスタマイズ、テスト、およびワークフロー管理を支援します。● FLEXnet Manager: 正確な利用状況分析レポートおよび集中型ライセンス マネージメントにより、貴社のソフトウェア資産の価値を最大限に引き出します。

今すぐ Macrovision ソリューションをお試しください！

評価版は、Macrovision Web サイト、または弊社のセールス スタッフから入手できます。詳しくは、<http://www.macrovision.com> をご覧いただくか、(1) (800) 809-5659 (北米)、(44) (117) 903- 0650 (ヨーロッパ、中東、アフリカ) までお電話ください。

Macrovision プロフェッショナル サービス

Macrovision から提供されているサービスは、業界トップのソフトウェアだけではありません。Macrovision は、ソフトウェア インストール、アップデート、およびエンタープライズ アプリケーション管理におけるすべてのニーズを解決するためのトレーニング サービス、テクニカル サポート サービス、およびコンサルティング サービスも提供しています。

エキスパート トレーニング サービス

初心者の方から今ある技術をさらに洗練させたいと考えている経験者の方まで、Macrovision のトレーニング コースはすべてのユーザの方々のニーズに応えます。

Macrovision のプロフェッショナル トレーナーはすべて、インストール エンジニアリング、アップデート、およびエンタープライズ ソフトウェア パッケージング、ソフトウェア ライセンシング、ライセンス マネジメントに精通しています。コースに参加することで、エキスパートに必要なスキルを学ぶことができます。

- Macrovision 認定のトレーナーによるコースは、Macrovision のみが提供しています。
- コースでは、日々直面するさまざまな問題を取り扱い、実践的なトレーニングを受けることができます。
- プライベート オンサイト コースも、お客様のニーズに合うようにカスタマイズして提供しています。

詳しくは、<http://www.macrovision.com/education> をご覧ください。オンライン フォームを使用して、直接お申し込みいただくこともできます。

製品サポート サービス

サポートが必要なときはいつでも、Macrovision のテクニカル サポート スタッフがご質問に答え、問題解決のお手伝いをします。

- Macrovision では、当社 Web サイトで入手できる数多くの賞を取ったサポート リソースに加え、1 年ごとに更新可能な多数の保守プランを提供し、お客様の特殊なニーズに応えられるよう常に努力しています。
- お客様のニーズと予算に最も適した保守プランを選ぶことができます。
- Macrovision のシニア エンジニアおよびサポート担当者によるパーソナル アシスタンスがご利用できます。

詳しくは、<http://www.macrovision.com/support/index.shtml> をご覧ください。

エキスパート コンサルティング サービス

Macrovision は、世界各地において、インストール エンジニアリング、アップデート、エンタープライズ ソフトウェア パッケージング、ソフトウェア ライセンシング、およびライセンス マネジメント サービスの最も信頼できるソースです。

- 必要に応じて、当社のエキスパートがプロジェクトをスケジュールおよび予算通りに完成するためのお手伝いをします。
- 御社のリソースを拘束することなく、プロフェッショナルな成果を得ることができます。
- エキスパートと同じ現場で作業することで、貴重なテクニックおよびベストプラクティスを学ぶことができます。
- 詳しくは、<http://www.macrovision.com/consulting> をご覧ください。オンライン フォームを使用して、直接お申し込みいただくこともできます。

その他の連絡先情報

表 2: Macrovision プロフェッショナル サービスのその他の連絡先

場所	電話番号
アメリカ、オーストラリア、ニュージーランド	(1) (847) 466-6000
ヨーロッパ、中東、アフリカ	(44) (0) (117) 903-0650
アジア太平洋	(81) (3) 5774-6253

テクニカル サポート リソース

Macrovision の製品サポート サービス チームでは、ユーザのすべてのニーズにお応えするためのリソースを包括的に提供しています。当社の製品に関するユーザ ドキュメントはすべて、製品のヘルプ ライブリで見つけることができます。ナレッジ ベースのようなオンライン リソースでは、当社の長年の経験に基づく貴重な情報に無料ですばやくアクセスできます。また、保守プランでは、専門エンジニアがより強固なサポートを提供します。以下のオプションを検討して、適切な解決方法をお選びください。

ナレッジ ベース

ナレッジ ベースは、当社 Web サイト (<http://www.macrovision.com/support/index.shtml>) の [サポート] セクションにあります。多数の一般的な質問に対する回答と、ドキュメントに含まれていないものを含む FLEXnet Licensing に関する最新情報が掲載されています。ナレッジ ベース検索エンジンを使用して、語句、番号、プラットフォーム、バージョンなどで項目を検索できます。

サポート サイト

ナレッジ ベースの検索の他に、Macrovision のサポート サイト (<http://www.macrovision.com/support/index.shtml>) では次のようなサービスを利用することができます。

- FLEXnet Licensing アップデート版の取得。
- FLEXnet Licensing のフィードバックの送信。
- オンラインセミナーの利用。
- FLEXnet Licensing の登録。
- FLEXnet Licensing ホワイト ペーパー、ケース スタディ、マーケティング資料、およびトレーニング資料の入手。
- オブジェクト ライブラリおよびサンプルプロジェクトの入手。

オンライン コミュニティ

オンライン コミュニティも、FLEXnet Licensing の使用法について不明な点があるときに、大変便利なリソースです。これらのコミュニティを利用すると、ユーザは、ヒントや知識を共有することで、FLEXnet Licensing の機能を最大限に引き出すことができます。コミュニティは、当社 Web サイトの [サポート] セクション (<http://community.macrovision.com>) からご利用いただけます。Centralized Enterprise Licensing User Group コミュニティ (<http://celug.com>) をご利用いただくこともできます。

Macrovision テクニカル サポートへ連絡する

Macrovision では、すべてのユーザのニーズに応えるためのさまざまな保守プランを用意しています。 <http://www.macrovision.com/support/maintenance.shtml> で、お客様に最も適したプランを見つけてください。

Macrovision の連絡先

Macrovision Web サイト <http://www.macrovision.com> にアクセスして、世界中どこからでも弊社へご連絡いただけます。

米国

表 3: 米国の連絡先

オフィス	住所	電話
Corporate Headquarters Santa Clara Office Video and Music Technologies FLEXnet Publisher	Macrovision Corporation 2830 De La Cruz Blvd. Santa Clara, CA 95050	Tel: (408) 562-8400 Fax: (408) 567-1800 Email: corpinfo@macrovision.com
Macrovision Chicago Office Licensing Technologies InstallShield, InstallAnywhere, FLEXnet AdminStudio, FLEXnet Manager	Macrovision Corporation 900 National Pkwy., Suite 125 Schaumburg, IL 60173	Tel: (847) 466-4000 Toll Free (800) 374-4353 Sales: (847) 466-6000 Sales Toll Free: (800) 809-5659 ご注文専用 FAX: (847) 619-0788
Macrovision San Francisco Office Games Technologies	Macrovision Corporation 795 Folsom Street, 2nd Floor - Suite 200 San Francisco, CA 94107	Tel: (415) 255-3060 Fax: (415) 255-0910 Email: gamesales@macrovision.com
Macrovision New York Office Information Commerce Group, eMeta, eRights Suite, RightAccess, RightCommerce, eRightsWEB	Macrovision Corporation 81 Franklin Street, Suite 500 New York, NY 10013	Tel: (212) 219-4400 Toll Free: (800) 804-0103 Fax: (212) 219-4401

ヨーロッパ、中東、アフリカ

表 4: ヨーロッパ、中東、アフリカの連絡先情報

オフィス	住所	電話
Macrovision UK Maidenhead Office	Macrovision UK Ltd. Malvern House 14-18 Bell Street Maidenhead Berkshire, SL6 1BR United Kingdom	Tel: (44) (870) 871 1111 Fax: (44) (870) 871 1161
Macrovision UK/Europe Cheshire Office FLEXnet Publisher, FLEXnet Manager	Macrovision Corporation Vision House, Priory Court Preston Brook, Cheshire, WA7 3FR United Kingdom	Tel: (44) (870) 873 6300 Fax: (44) (192) 870 6329
Macrovision UK/Europe Cheshire Office InstallShield, InstallAnywhere, Update Service, FLEXnet AdminStudio	Macrovision Corporation Vision House, Priory Court Preston Brook, Cheshire, WA7 3FR United Kingdom	Tel: (44) (117) 903 0650 German: (44) (192) 870 6315 French: (44) (192) 870 6314 Spanish: (44) (192) 870 6311 Fax: (44) (192) 870 6329
Macrovision France FLEXnet Manager	Macrovision Corporation 33 rue de Galill仔 75016 Paris France	Tel: (33) 1 44 43 53 44 Fax: (33) 1 44 43 53 91 すべての InstallShield と AdminStudio に関するお問い合わせは、(44) (192) 870-6314 までご連絡ください。
Macrovision Netherlands FLEXnet Publisher, FLEXnet AdminStudio, FLEXnet Manager	Macrovision Corporation Busitel 1, Orlyplein 85 1043 DS Amsterdam The Netherlands	Tel: (31) (0) 20 403 7673 Fax: (31) (0) 20 403 7676
Macrovision Alicante Office Games Technologies	Macrovision Corporation Av. Jaime I El Conquistador 1-3 Bajo El Campello Alicante 03560 Spain	Tel: (34) 956 107 771 Email: gamesales@ macrovision.com

アジア

表 5: Macrovision アジア地域の連絡先情報

オフィス	住所	電話
Macrovision Corporation Greater China Office	Macrovision Corporation Unit 301B, 3/F Malaysia Building 50 Gloucester Road, Hong Kong	Tel: (852) 2868-5131 Fax: (852) 2850-6690 Email: Eric.So@ macrovision.com
マクロヴィジョン ジャパン アンアジア K.K. 日本オフィ ス	マクロヴィジョン コーポ レーション 〒150-0002 東京都渋谷区渋谷1-1-8 青山ダイヤモンドビル	Tel: (81) (3) 5774- 6253 Fax: (81) (3) 5774- 6269
Macrovision Korea Office	Macrovision Corporation Suite 417, Korea City Air Terminal B/D 159-6 Samsung-dong, Kangnam-ku, Seoul 135-728 Korea	Tel: (82) (2) 6241- 5700 Fax: (82) (2) 6241- 5701 Mobile: (82) (11) 779- 0445
Macrovision Corporation Taiwan Office	Macrovision Corporation 6F-13, No. 12, Lane 609, Sec. 5 Chung-Hsin Rd., San-Chung Taipei, Taiwan 241	Tel: (886) (2) 2999- 7979 Fax: (886) (2) 2999- 7575 Email: Keith.Lien@ macrovision.com

1

ライセンス ファイルの基本情報

ライセンス ファイルには、FLEX 対応アプリケーションのライセンスを FLEXnet Licensing が管理するために必要な情報が含まれています。次の情報が含まれます。

- ライセンス サーバシステムの名前とホスト ID
- ベンダー名とベンダー デーモン実行プログラムへのパス
- 機能情報

一般的に、FLEX 対応アプリケーションを実行するすべてのマシンと、ライセンス サーバマシンとして指定されたマシンが、ライセンス ファイル、またはそれをコピーしたファイルにそれぞれアクセスできなければなりません。

ライセンス ファイルの場所を指定する

一般的にソフトウェア ベンダーは、ライセンス ファイルを特定の場所に置くように推奨します。アプリケーションを複数のマシンで実行している場合、そのすべてのマシンでライセンスを有効にするオプションを次に示します。

- ライセンス ファイルを必要とするネットワーク内のすべてのマシンが使用できるパーティションに、ライセンス ファイルを置く。
- ライセンス ファイルを必要とするすべてのマシンに、ライセンス ファイルをコピーする。
- 環境変数 `LM_LICENSE_FILE` または `VENDOR_LICENSE_FILE` (この `VENDOR` はベンダー デーモン名) を、`port@host` に設定する (`host` および `port` はライセンス ファイルの `SERVER` 行から適用)。または、ライセンス ファイルの `SERVER` 行が

既定値のポート範囲 (27000-27009) に TCP/IP ポートを指定するか、ポートを指定しない (システムが既定値範囲から選択可能) ならば、ショートカット指定 `@host` を使用します。

- 3 サーバによる冗長構成のサーバ設定に対しては、3 サーバによる冗長構成の設定であることを示す、カンマで区切られた 3 つの `port@host` 指定のリストを使用します。例を次に示します。

```
port1@host1,port2@host2,port3@host3
```

上の例は、`host1`、`host2`、`host3` で構成された 3 サーバによる冗長構成の設定を指定しています。

環境変数 `LM_LICENSE_FILE` と `VENDOR_LICENSE_FILE` の設定例を [表 1-1](#) に示します。

表 1-1: 環境変数の指定例

SERVER 行	LM_LICENSE_FILE または VENDOR_LICENSE_FILE の設定
SERVER myserver 17007ea8 40000 以下に詳細を示します。 <ul style="list-style-type: none">● host = myserver● port = 40000	40000@myserver
SERVER myserver 17007ea8 27001 以下に詳細を示します。 <ul style="list-style-type: none">● host = myserver● port = 27001、既定値の範囲内	@myserver
SERVER myserver 17007ea8 以下に詳細を示します。 <ul style="list-style-type: none">● host = myserver● port = 何も指定しません。この場合、27000 - 27009 の範囲で既定値の TCP/IP ポート番号が使用されます。	@myserver

- Windows では、アプリケーションがライセンス ファイルを発見できなかった場合、ライセンス ファイルの場所あるいはライセンス サーバシステムの指定を求める、またはインターネット上でのライセンス手続きの完了を確認するダイアログが表示されます。

ベンダー デーモンはライセンスの使用状況を追跡しており、またライセンス ファイルには改ざんを防ぐために暗号化されたデータが含まれているため、ライセンス ファイルを必要だけコピーまたは移動することができます。

カウントされるライセンスについては、どのようなオプションを選択しても、FLEX 対応アプリケーションがアクセスできる場所に、lmgrd とベンダー デーモンを最初にコピーする必要があります。

環境変数でパスを設定する

ほとんどのアプリケーションは、指定場所にライセンス ファイルを自動的にインストールし、そこからライセンス ファイルを検索します。一方、環境変数 `LM_LICENSE_FILE` を `license_file_list` に設定してライセンス ファイルの場所を変更することもできます。`license_file_list` を指定する場合、リストは次のコンポーネンで構成することができます。

- ライセンス ファイルへの完全パス
- 拡張子 `.lic` が付いた 1 つまたは複数のライセンス ファイルを収容するディレクトリ
- `port@host` の設定。この `port` と `host` は、ライセンス ファイルの `SERVER` 行に指定された TCP/IP ポート番号とホスト名です。または、ライセンス ファイルの `SERVER` 行で TCP/IP ポートの既定値またはポート範囲の既定値 (27000-27009) が指定されている場合、ショートカット指定 `@host` を使用できます。
- カンマで区切られた 3 つの `port@host` のリストは、3 サーバによる冗長構成の設定を示します。例を次に示します。

```
port1@host1,port2@host2,port3@host3
```

上の例は、`host1`、`host2`、`host3` で構成された 3 サーバによる冗長構成の設定を指定しています。

アプリケーションは `VENDOR_LICENSE_FILE` のような名前 (たとえば、`MVSN_LICENSE_FILE`) の環境変数 (または Windows のレジストリ) を利用できます。ここで `VENDOR` はベンダー デーモン名です。この環境変数の適用範囲は、そのベンダー `VENDOR` のアプリケーションに制限されます。UNIX の場合、ライセンス ファイルのリスト コンポーネントはコロンで区切られ、Windows ではセミコロンで区切られます。

`lmgrd` と `lmutil` (`lmstat`、`lmdown` など) に関しては、`-c` オプションを使用すると環境変数 `LM_LICENSE_FILE` の設定が無効になります。



注: アプリケーションによっては、環境変数 `LM_LICENSE_FILE` を認識しないものもあります。特に、FLEX 対応 JAVA アプリケーションはこの環境変数を認識しません。

関連事項:

`LM_LICENSE_FILE` の詳細は、「[複数のライセンス ファイルを管理する](#)」を参照してください。
[「環境変数」](#)

既定値の TCP/IP ポート番号の範囲は、バージョン 6.0 の `lmgrd` で導入

ライセンス ファイルの場所に関するダイアログは、バージョン 6.0 のクライアント ライブラリで導入

ライセンス ファイル フォーマットの概要

通常、ライセンス ファイルは先頭に SERVER 行が 1 行 (冗長構成のサーバの場合は 3 行) 置かれ、その後 1 行または複数行の VENDOR 行、さらに 1 行または複数行の FEATURE 行または INCREMENT 行が続きます。場合によっては、ライセンス ファイルに SERVER 行も VENDOR 行も必要ないことがあります。

ライセンス ファイルで修正できる要素を次に示します。

- SERVER 行のホスト名
- SERVER 行の TCP/IP ポート番号
- VENDOR 行のパス
- VENDOR 行のオプション ファイルのパス
- VENDOR 行のオプション TCP/IP ポート番号 (ファイアウォールのサポートのみ)
- USE_SERVER 行
- FEATURE 行の *keyword=value* のペアで、*keyword* が小文字で指定された場合の値

関連事項:

[「冗長構成のライセンス サーバ システム」](#)

[「カウントされるライセンスとカウントされないライセンス」](#)

これらのコンポーネントの詳細については、[「ライセンス ファイルのフォーマット」](#)を参照してください。

USE_SERVER は、バージョン 5.0 の クライアント ライブラリで導入

VENDOR 行は、バージョン 6.0 以前の lmgrd とベンダー デーモンでは DAEMON 行

ライセンス ファイルの種類

ライセンス ファイルは、ソフトウェア ベンダーが作成します。ライセンス ファイルは、フローティング (同時) 使用、ノードロック (カウントされる場合とカウントされない場合の両方)、およびフローティング、カウントされる、カウントされない、などの任意の組み合わせを指定します。

フローティング (同時使用) ライセンス

「フローティング ライセンス」は、ネットワーク上の任意のユーザが FLEX 対応アプリケーションを使用できることを意味し、その上限はライセンス ファイルで指定します (「同時使用」または「ネットワーク ライセンス」とも呼ばれます)。フローティング ライセンスの場合、個々の FEATURE 行にホスト ID は含まれません。フローティング ライセンスでは、ライセンスの同時使用をカウントするために `lmgrd` およびベンダー デーモンが実行されている必要があります。

フローティング ライセンスを実現するライセンス ファイルの例を次に示します。

```
SERVER lulu 17007ea8
VENDOR sampled
FEATURE f1 sampled 1.00 1-jan-2005 2 SIGN=signature1
FEATURE f2 sampled 1.00 1-jan-2005 6 SIGN=signature2
FEATURE f3 sampled 1.00 1-jan-2005 1 SIGN=signature3
```

このライセンス ファイルは、ライセンス サーバ `lulu` にアクセスできるネットワーク上のマシンから、機能 `f1` の 2 つのライセンス、機能 `f2` の 6 つのライセンス、および機能 `f3` の 1 つのライセンスが使用可能であることを指定します。`lmgrd` は既定値の FLEXnet Licensing TCP/IP ポートの 1 つを使用します。

ノードロック ライセンス

ノードロックとは、FLEX 対応ソフトウェアが 1 台のマシンまたは数台で構成される 1 つのマシングループでしか使用できないことを意味します。ノードロック ライセンスでは、特定のホストにノードロックされた FEATURE 行にホスト ID が含まれます。ノードロック ライセンスには、カウントされないライセンスとカウントされるライセンスの 2 種類があります。

ライセンス数が 0 (ゼロ) (または `uncounted`) に設定された場合、ライセンスはカウントされず、指定されたマシンで無制限に使用することが許可されます。この設定では機能の同時使用がカウントされないため、`lmgrd` またはベンダー デーモンは必要ありません。

次に示すライセンス ファイルは、ホスト ID が `17007ea8` および `1700ab12` であるマシンで機能 `f1` を無制限に使用することを許可します。

```
FEATURE f1 sampled 1.000 1-jan-2005 uncounted SIGN=signature1 ¥
HOSTID=17007ea8
FEATURE f1 sampled 1.000 1-jan-2005 uncounted SIGN=signature2 ¥
HOSTID=1700ab12
```

または上記 2 行の FEATURE 行は、「ホスト ID リスト」を使用して、ソフトウェア ベンダーが次のように発行していることもあります。

```
FEATURE f1 sampled 1.000 1-jan-2005 uncounted SIGN=signature ¥  
HOSTID="17007ea8 1700ab12"
```

このライセンス ファイルに、他に FEATURE 行がない場合、lmgrd デーモンは不要であり、起動する必要もありません。

次のライセンス ファイルは、ホスト ID 1300ab43 を持つマシンにロックされた f1 機能に対して 3 つのライセンスを提供しています。ライセンス サーバ システムとライセンスが同じマシンにロックされているため、デーモンは、FLEX 対応アプリケーションを実行するマシンと同じマシン上で実行されます。

```
SERVER lulu 1300ab43 1700  
VENDOR sampled /etc/sampled  
FEATURE f1 sampled 1.00 1-jan-2005 3 SIGN=signature ¥  
HOSTID=1300ab43
```

ノードロック ライセンスとフローティング ライセンスの混在

カウントされないノードロック ライセンスと、同時使用ライセンスは同じライセンス ファイルに混在できます。

次に示すライセンス ファイルは、ホスト ID が 17007ea8 および 1700ab12 であるマシンに機能 f1 の無制限の使用を許可する一方、ネットワーク上の任意のマシンに機能 f1 を使用できるライセンスを 2 つ許可します。

```
SERVER lulu 17001234 1700  
VENDOR sampled C:\¥flexlm¥sampled.exe  
FEATURE f1 sampled 1.00 1-jan-2005 uncounted SIGN=signature1 ¥  
HOSTID=17007ea8  
FEATURE f1 sampled 1.00 1-jan-2005 uncounted SIGN=signature2 ¥  
HOSTID=1700ab12  
FEATURE f1 sampled 1.00 1-jan-2005 2 SIGN=signature3
```

この設定では、3 行目の FEATURE 行の 2 つの同時使用のライセンスがカウントされるため、lmgrd とベンダー デーモンが必要です。

複数ベンダーのライセンスの管理

2,500 社以上のベンダーがライセンス管理に FLEXnet Licensing を採用しているため、複数ベンダーの FLEXnet Licensing ライセンスの管理が必要となる可能性が高いと思われま

複数ライセンスの管理方式の概要

複数ベンダーの FLEX 対応製品を実行する場合、ライセンスの衝突を回避するための処理をインストール中に行う必要があります。このための方式がいくつかありますが、次にそのうちの 3 例を示します。

- 複数のマシンでそれぞれ 1 つの `lmgrd` と 1 つのベンダー デーモンを実行し、1 つのライセンス ファイルを使用する
- 1 台のサーバ マシンで複数の `lmgrd` を実行し、それぞれ 1 つのベンダー デーモンを実行し、1 つのライセンス ファイルを使用する
- 1 台のサーバ マシンで 1 つの `lmgrd` を実行し、それぞれのライセンス ファイルを使用して複数のベンダー デーモンを実行する。これらのライセンス ファイルは 1 つの共通ディレクトリにある

これらの方式の中では、1 番目がベンダー間の独立性が最も高く、3 番目が独立性が最も低いものとなっています。上記の 1 番目の方式では複数のライセンス サーバを監視する必要がありますが、3 番目の方式では、1 つのサーバと 1 つの `lmgrd` だけを管理すればよいことになります。次のセクションで、この 3 つの方式を詳しく説明します。バリエーションについては、「[その他の検討項目](#)」で説明します。

複数のマシン

このモデルでは、異なるベンダー デーモンとそれに関係付けられた 1 つまたは複数のライセンス ファイルが、それぞれ別のサーバ マシンに配置されます。各マシンはそのベンダー デーモンのライセンスのみを交付し、ローカルの `lmgrd` を実行します。図 2-1 は、このモデルを表しています。

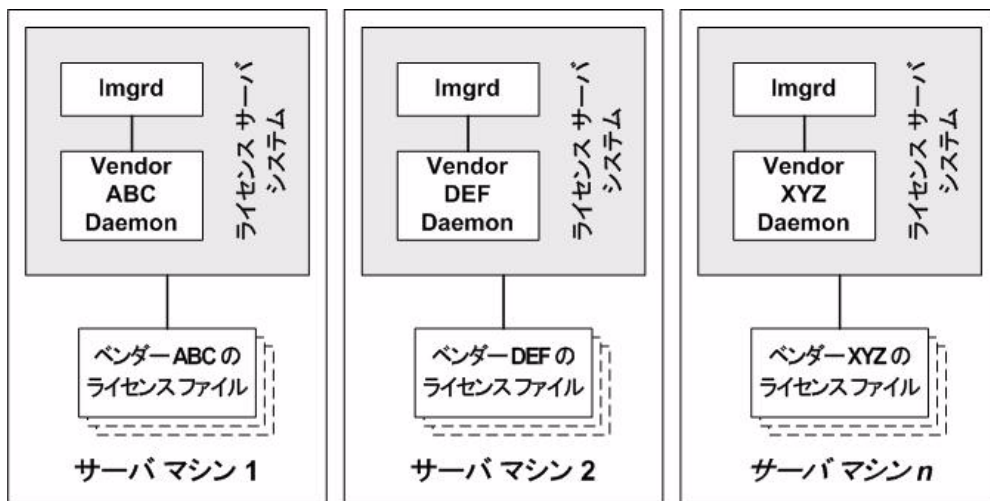


図 2-1: 複数のサーバ システム マシン

長所

- 各ベンダーに対するライセンス ファイルは互いに独立している。
- マシンは別々に保持される。1 台のマシンがダウンしても、その他のマシンで提供されているベンダーのライセンス交付が続けられます。
- 各サーバは個別にデバッグ ログを持つ。
- ライセンス交付の負荷が分散される。

短所

- 管理負担が最も高い。
- 1 台のマシンがダウンすると、そのマシンに関連するベンダー製品に対するライセンス交付機能がダウンする。

ライセンス サーバ システムを起動する



操作手順 **ライセンス サーバ システムを起動するには:**

- 次のように、各マシンのライセンス サーバ マネージャを起動します。

```
lmgrd -c server_machine_n_license_list
```

ここで `server_machine_n_license_list` は「[複数のライセンス ファイルを管理する](#)」で説明するライセンス ファイル リストです。各 `lmgrd` は、そのライセンス ファイル内で参照されるベンダー デーモンを起動します。

複数のライセンス サーバ システムを持つ 1 台のマシン

このモデルでは、各ベンダー デーモンとそれに関連する 1 つまたは複数のライセンス ファイルが、それ自身の `lmgrd` プロセスによって管理され、すべてが 1 台のサーバ マシンに収容されます。[図 2-2](#) はこのモデルを表しています。

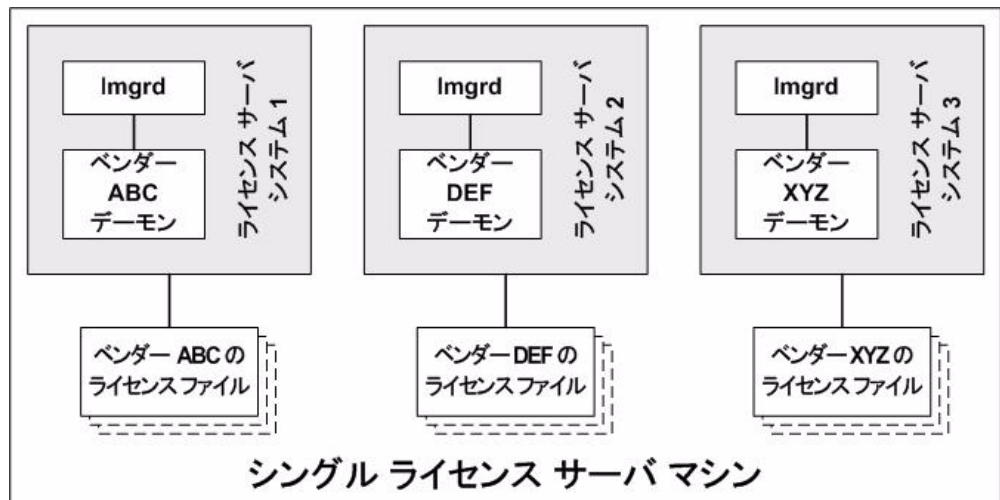


図 2-2: 複数の `lmgrd`、複数のライセンス ファイル

同一マシン上で独立したライセンスサーバ システムを複数保持する場合には、次のことに注意してください。

- TCP/IP ポート番号を SERVER 行で指定する場合、そのポート番号は各ライセンス サーバ システムに対してそれぞれ異なる番号を指定しなければなりません。標準のテキスト エディタを使用して、各ライセンス ファイルの TCP/IP ポート番号を重複しないように変更することができます。10 以下のライセンス サーバ システムを実行している場合は、すべてのポート番号を省略し、既定値範囲の 27000 ~ 27009の中から `lmgrd` が一意の番号を選択するようにできます。
- 特定のライセンス ファイルごとに起動した `lmgrd` のバージョンに互換性があることを確認する必要があります。そのためには、`lmgrd` への明示的なパスを設定します。「[コンポーネントのバージョン互換性](#)」を参照してください。
- ライセンス サーバ システムの数は、マシンの CPU メモリとネットワーク上の制限を受けるだけです。

長所

- 各ベンダーに対するライセンス ファイルは互いに独立している。
- ライセンス サーバ システムは個別に保持される。1 つのサーバがダウンしても、その他のサーバで提供されているベンダーのライセンス交付が続けられます。
- 各サーバは個別にデバッグ ログを持つ。

短所

- 管理経費が比較的高い。
- マシンがダウンした場合は、すべてのライセンス交付が不能になる。
- ライセンス交付の負荷が 1 台のマシンに集中する。

ライセンス サーバ システムを起動する



操作手順 **ライセンス サーバ システムを起動するには:**

- ライセンス サーバ システムを次のように起動します。
 - a. サーバ 1 に対しては、`lmgrd -c vendor_ABC_license_dir_list`
 - b. サーバ 2 に対しては、`lmgrd -c vendor_DEF_license_dir_list`
 - c. サーバ 3 に対しては、`lmgrd -c vendor_XYZ_license_dir_list`

ここで `vendor_nnn_license_list` は「[複数のライセンス ファイルを管理する](#)」で説明するライセンス ファイル リストです。各 `lmgrd` は、そのライセンス ファイル内で参照されるベンダー デーモンを起動します。

1つのライセンス サーバシステムと複数のライセンス ファイルを持つ1台のマシン

このモデルでは、1つの `lmgrd` プロセスがサーバマシン上で実行され、これにより、各自が1つまたは複数のライセンス ファイルを持つ1つまたは複数のベンダー デーモンを管理します。通常、ライセンス ファイルは同じディレクトリ内に置かれます。ライセンス ファイルに付けられる標準のファイル名拡張子は `.lic` です。FLEXnet Licensing によってベンダー デーモンの数が制限されることはありません。図 2-3 はこのモデルを表しています。

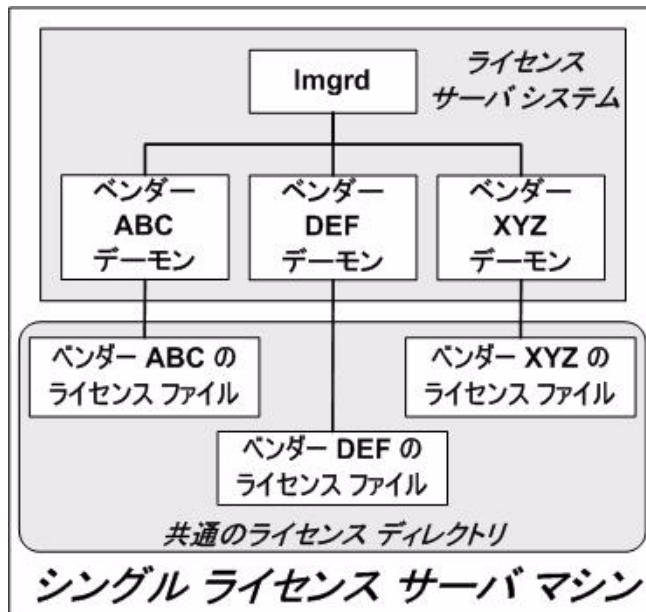


図 2-3: 1つの `lmgrd`、複数のライセンス ファイル

長所

- ライセンスファイルを個別に保持できる。
- 少ない管理経費。

短所

- 1つの `lmgrd` がすべてのベンダー デーモンを管理する。`lmgrd` がダウンした場合は、すべてのライセンス交付が不能になります。
- マシンがダウンした場合は、すべてのライセンス交付が不能になる。

- 各ベンダーのオプション ファイル内で DEBUGLOG を使用して個別にデバッグ ログを指定しないと、すべてのベンダー デーモンからの出力は 1 つの共通デバッグ ログに書き込まれる。1 つの共通デバッグ ログを使用すると、個々のベンダー デーモンの問題をデバッグするのが難しくなります。
- 1 台のマシンと 1 つの lmgrd プロセスに対してライセンス交付の負荷が最大となる。

ライセンス サーバ システムを起動する



操作手順 **ライセンス サーバ システムを起動するには:**

- 次のように、サーバ マシンのライセンス サーバ マネージャを 1 回起動します。

```
lmgrd -c common_license_directory
```

lmgrd は、common_license_directory 内の拡張子 .lic を持つすべてのファイル を処理し、それらのファイル内で参照されるすべてのベンダー デーモンを起動し ます。そのため、lmgrd コマンド行で各ライセンス ファイル名を列挙する必要はあ りません。

関連事項:

[「複数のライセンス ファイルを管理する」](#)

[「特定のベンダー デーモンのデバッグ ログ出力をキャプチャする」](#)

1 つの lmgrd が複数のライセンス ファイルを処理する機能は、バージョン 6.0 の lmgrd で導入

複数のライセンス ファイルを管理する

同じサーバ マシン上にある複数のライセンス ファイルを、ライセンスファイル リスト を介して管理できます。ライセンスファイル リストは次に示す 2 つの方法で指定できます。

- lmgrd に対して -c オプションを使用する。

```
lmgrd -c license_file_list [他の lmgrd オプション]
```

- 環境変数 LM_LICENSE_FILE を lmgrd プロセス環境の範囲内に指定する。

ライセンス ファイルをサーバ マシン上の任意の場所にインストールし、次に license_file_list を定義します。

license_file_list を指定する場合、リストは次の 1 つまたは複数のコンポーネンで構成さ れるようになります。

- ライセンス ファイルへの完全パス
- 拡張子 .lic が付いた 1 つまたは複数のライセンス ファイルを収容するディレクトリ

- `port@host` の設定。この `port` と `host` は、ライセンス ファイルの `SERVER` 行に指定された TCP/IP ポート番号とホスト名です。または、ライセンス ファイルの `SERVER` 行で TCP/IP ポートの既定値またはポート範囲の既定値 (27000-27009) が指定されている場合、ショートカット指定 `@host` を使用できます。
- カンマで区切られた 3 つの `port@host` のリストは、3 サーバによる冗長構成の設定を示します。例を次に示します。

```
port1@host1,port2@host2,port3@host3
```

上の例は、`host1`、`host2`、`host3` で構成された 3 サーバによる冗長構成の設定を指定しています。



注: コロン(:)を使用して、UNIX 上のライセンス ファイル名を区切ります。Windows の場合はセミコロン(;)を使用します。

`lmgrd` は起動時に各ライセンスファイルリストのコンポーネントをリストされた順番に構文解析し、内部ライセンスファイル リストを作成します。

ライセンスファイル リストを使用するシナリオは、「[複数のマシン](#)」、「[複数のライセンス サーバシステムを持つ 1 台のマシン](#)」または「[1 つのライセンス サーバシステムと複数のライセンス ファイルを持つ 1 台のマシン](#)」にも記述されています。

関連事項:

- 「[環境変数でパスを設定する](#)」
- 「[ライセンスファイル リストによる冗長構成](#)」
- 「[FLEXnet Licensing 環境変数](#)」

その他の検討項目

ライセンス ファイルを結合する

2 つ以上の製品のライセンスを、ライセンス ファイルの `SERVER` 行に指定されたように同じマシンに使用する場合、それらのライセンス ファイルを単一のライセンス ファイルに結合することもできます。一定の基準を満たしていれば、「[複数のライセンス サーバシステムを持つ 1 台のマシン](#)」および「[1 つのライセンス サーバシステムと複数のライセンス ファイルを持つ 1 台のマシン](#)」で説明したモデルのライセンス ファイルを結合することができます。詳細は、「[ライセンス ファイルを結合する規準](#)」を参照してください。図 2-4 は、結合したライセンス ファイルを使用するモデルの一例を示しています。

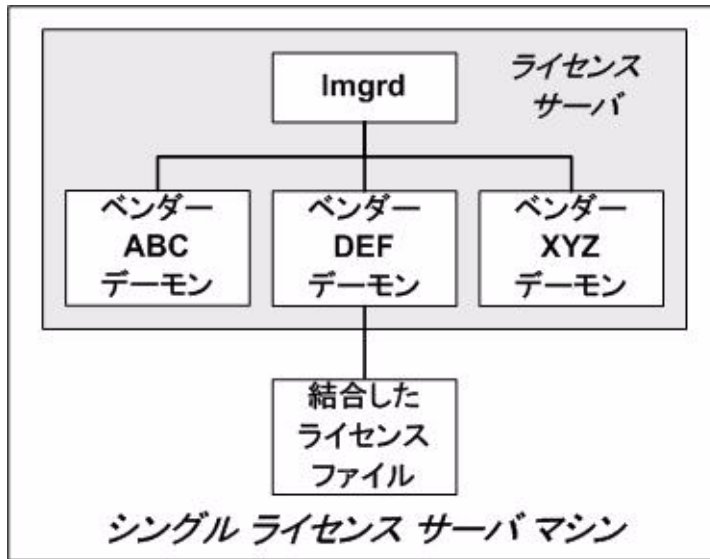


図 2-4: 1つの lmgrd、1つのライセンスファイル

長所

- 管理者にとってライセンスファイルが1つである。
- 一度ファイルを結合すると、管理経費が少なくてすむ。

短所

- 複数ベンダーのライセンスファイルを1つのファイルに結合する作業は、注意深く計画することが必要で、これが初めから継続的に必要となる。

ライセンスサーバシステムを起動する



操作手順 ライセンスサーバシステムを起動するには:

- 次のように、サーバマシンのライセンスサーバマネージャを1回起動します。

```
lmgrd -c combined_license_file
```

ライセンス ファイルを結合する規準

製品のライセンス ファイルは、ライセンス ファイル内の SERVER 行のホスト名とホスト ID によりライセンス サーバ マシンを定義しています。次の条件を満たす場合、ライセンス ファイルの結合が可能です。

- 各ファイルに含まれる SERVER 行の数が同じである。
- 一方のファイルの各 SERVER 行のホスト ID フィールドが、もう一方のファイルの各 SERVER 行のホスト ID フィールドに「正確に」一致する。

ライセンス ファイルを結合できない理由として考えられる原因をいくつか示します。

- ライセンス ファイルを異なるサーバ マシン上で実行するよう設定したため、ホスト ID が異なる。
- 1 つのライセンス ファイルはシングル サーバ システム用 (SERVER 行は 1 行のみ) に設定されているが、その他のライセンス ファイルが 3 サーバによる冗長構成のライセンス サーバ設定用 (SERVER 行が複数含まれる) に設定されている。
- 同一マシンに対して異なるタイプのホスト ID を使用されている。たとえば、あるライセンス ファイルの SERVER 行がそのホスト ID に対して INTERNET= を使用しており、それ以外のファイルではそのホスト ID のタイプにイーサネットの MAC アドレスを使用している、などの場合。

使用しているライセンス ファイルに前述のような互換性がある場合、[図 2-4](#) や後述する「[ライセンス ファイルの結合方法](#)」で示すようにライセンス ファイルを結合できます。ただし、ライセンス ファイルに互換性があっても、必ずしも結合する必要がないことに注意してください。ファイルを結合しなくても性能またはシステム負荷には実質的に何の影響もありません。

ライセンス ファイルの結合方法

ライセンス ファイルに互換性がある場合、任意のテキスト エディタを使用してファイルを結合します。ライセンス ファイルを結合するには、互換性のあるライセンス ファイルをすべて 1 つのファイルに読み込み、余分な SERVER 行を削除して、SERVER 行が 1 セットだけ残るようにします。このデータを保存すれば、結合されたライセンス ファイルが完成します。[図 2-5](#) はライセンス ファイルを結合する例を示しています。

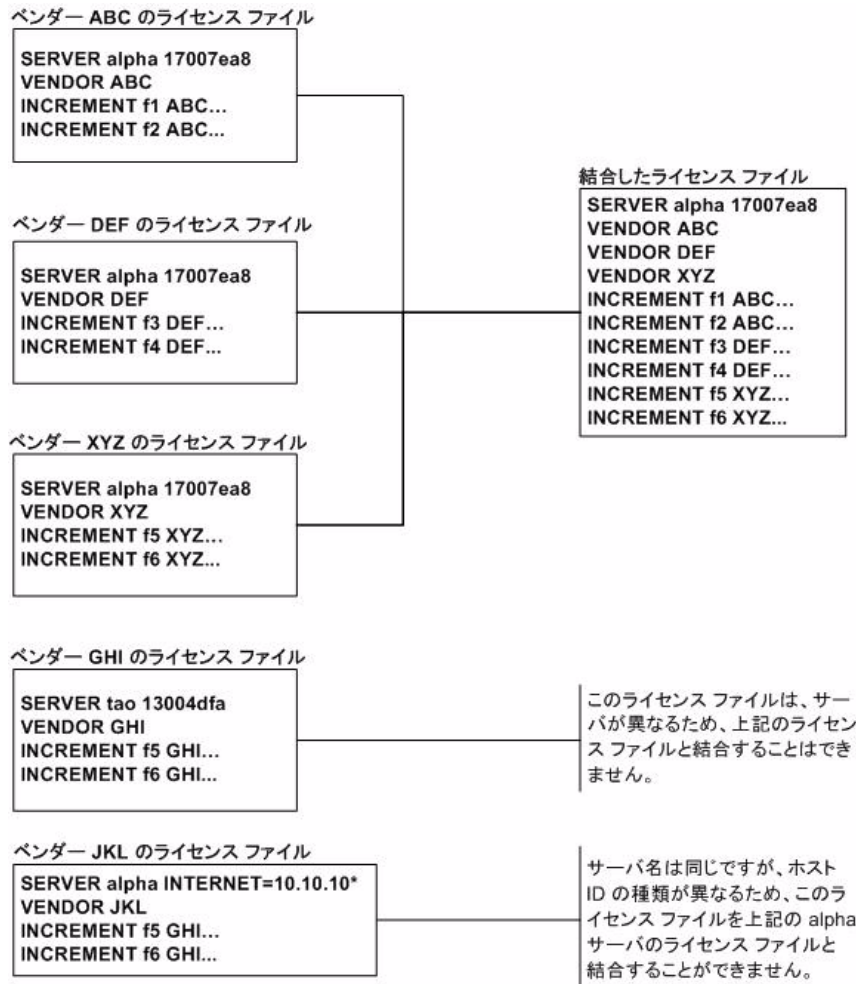


図 2-5: ライセンス ファイルを結合

コンポーネントのバージョン互換性

1 つの lmgrd プロセスが複数のベンダー デーモンを管理している場合、これらのデーモンが使用する FLEXnet Licensing のバージョンが同じであるとは限りません。「[FLEXnet Licensing コンポーネントのバージョンの互換性](#)」で説明する FLEXnet Licensing のバージョンの互換性規則に注意することで、使用中のすべての FLEXnet Licensing コンポーネントの互換性を確保することができます。

特定の FLEX 対応アプリケーションに対しては、新バージョンまたは古いバージョンのどちらかを使用します(もちろんそのアプリケーション用のベンダー デーモンは、少なくともアプリケーションと同じかそれ以降のバージョンである必要があります)。

ライセンス サーバ マシンの 選択

この章では、ライセンス サーバマシンとしてどのマシンを使うべきかを判断する手がかりを提供します。

サーバが使用するリソース

このセクションでは、ライセンス サーバシステムが使用するリソースについて説明します。サーバマシンを選択する際には、これらのリソースに関するシステム上の制限について検討が必要になる場合があります。ライセンスが少数(約 100 以下)の場合、このシステム上の制限のほとんどは、どのワークステーションでも問題にはなりません。

ソケット

TCP/IP ポートを使用すると、ライセンス サーバに接続している各 FLEX 対応アプリケーションが 1 つまたは複数のソケットを使用します。各 FLEX 対応アプリケーションが必要とするソケット数は、FLEXnet Licensing の実装方法によって異なります。この情報についてはベンダーに確認してください。ライセンス サーバシステムが使用できるソケットの数は、ファイル記述子に対するプロセスごとのシステムの制限によって定義されます。ライセンス サーバシステムが使用するソケットの総数は、それが管理する FLEX 対応アプリケーションに必要な総数より少し多くなります。

単一マシン上のライセンス サーバシステムが要求するソケット数が制限を上回った場合は、ライセンス ファイルを複数のファイルに分割して他のサーバ上に配置し、ネットワークトラフィックを軽減する解決策が考えられます(このためには、ソフトウェア

ベンダーに新しいライセンスの発行を依頼する必要があります)。FLEX 対応アプリケーションは、環境変数 `LM_LICENSE_FILE` を介してライセンスファイルリストを使用することによって、複数のサーバからライセンスをチェックアウトできるようになります。

CPU 時間

クライアントの数が少ない場合、ライセンス サーバ システムは CPU 時間をほとんど使用しません。長期間にわたって、サーバが CPU 時間をほんの数秒しか使用しない可能性もあります。

クライアントの数が多く、それぞれがハートビート メッセージをサーバと交信している場合、またはチェックアウトやチェックインの頻度が高い場合 (毎秒数百回)、サーバが消費する CPU 時間の合計が著しく大きくなる場合があります。ただし、その場合でも、CPU の使用率は通常高くありません。この場合、サーバとして選んだマシンが、CPU サイクルに十分余裕があることを確認する必要があります。

ディスク スペース

ライセンス サーバ システムが作成する出力ファイルは、デバッグ ログ ファイルとレポート ログ ファイルだけです。レポート ログ ファイルは、ライセンス使用状況の正確なレポートを生成するために FLEXnet Manager が使用します。ライセンス動作が頻繁に発生する場合、これらのログ ファイルのサイズが非常に大きくなります。したがって、これらのファイルの保存場所や、ローテーション頻度、保存頻度を検討する必要があります。ディスク スペースに余裕がない場合、ライセンス管理者はログ ファイルの出力を抑制することができます。

ネットワークに依存しなくてよいように、ログ ファイルをサーバ マシンのローカル ファイルとすることをお勧めします。

関連事項:

[「環境変数でパスを設定する」](#)

[「UNIX プラットフォームでライセンス サーバ マネージャを起動する」](#)

[「レポート ログ ファイル」](#)

[「デバッグ ログ ファイル」](#)

メモリ

FLEXnet ライセンス サーバ システムは、メモリをほとんど使用しません。SunOS では、`lmgrd` は約 2 MB、ベンダー デーモンは約 2 MB 使用します。ただし、ライセンス ファイルのサイズが大きくなり、同時に存在するユーザの数が増えるにつれ、ベンダー デーモンのメモリの使用量は増加します。

ネットワークの帯域幅

FLEXnet Licensing がネットワークに送るデータは比較的少量です。チェックアウトやチェックインなど各トランザクションは、通常 1 KB 未満のデータを転送するだけで十分です。したがって、クライアントが少数の場合 FLEXnet Licensing によるライセンス交付は、ダイヤルアップの SLIP 回線などの通信速度の遅いネットワーク上でも効率的に実行できます。

FLEX 対応アプリケーションの数が多い場合(数百規模の場合)、それぞれのアプリケーションがハートビート メッセージをバンダー デーモンと交信するために、使用するネットワークの帯域幅が大きくなってきます。このような場合は、FLEX 対応アプリケーションとサーバを同じローカル エリア ネットワーク上で実行します。それに、2つのサーバ用にライセンスを2つのファイルに分割する必要があるかもしれません。ユーザは、環境変数 `LM_LICENSE_FILE` 内のライセンスファイル リストを使用して、両方のサーバに効率よくアクセスすることができます。

関連事項:

「ライセンス ファイルの場所を指定する」

リモートにマウントされているディスク

ライセンス サーバ システムを稼働する場合、リモートにマウントされているディスクを使用しないことをお勧めします。つまり、`lmgrd`、バンダー デーモン、ライセンス ファイル、デバッグ ログ ファイル、レポート ログ ファイルはすべて、ローカルにマウントされたディスク上に置くことをお勧めします。これらのファイルのいずれかがリモートにマウントされたディスクにある場合、エラーの起きうる箇所が倍になり、全ライセンスの一時的な喪失につながる可能性があります。すべてのファイルがローカルにマウントされている場合、サーバマシンが稼働している限りライセンスは利用可能です。一方、ファイルが別のマシン上にある場合、ライセンス サーバ マシンかファイル サーバ マシンのいずれかがダウンするとライセンスが利用できなくなります。

冗長構成のライセンス サーバ システム

冗長構成のライセンス サーバ システムを採用する場合は、安定したシステムを選択してください。つまり、何らかの理由で頻繁に再起動やシャット ダウンを必要とするようなマシンは選択しないでください。冗長構成のライセンス サーバ マシンは、ライセンス サーバ システムをサポートする任意のマシンです。

FLEXnet Licensing は、冗長構成を実現する手段として2つの方法をサポートしています。

- 環境変数 `LM_LICENSE_FILE` にライセンスファイル リストを設定して実現する。
- 3つのライセンス サーバ システムを冗長構成することで実現する。

LM_LICENSE_FILE リストの冗長構成の場合は、ライセンス サーバ システム グループの各サーバは、ライセンス全体の一部を提供します。エンド ユーザは、LM_LICENSE_FILE をライセンス ファイルのリストに設定します。これらのライセンス ファイルは、それぞれライセンス サーバ システムの1つを参照します。アプリケーションは、アクセスに成功するか、またはリストの終わりに達するまで、リスト内の各サーバへのアクセスを順番に試みます。

3 サーバによる冗長構成の場合は、3 つのライセンス サーバ システムのうち2つが稼働中ならば(3 つのライセンス サーバ システムのうち2つを指して「定足数」と呼びます)、システムは機能し、必要なライセンスをすべて交付します。

関連事項:

[「複数のライセンス ファイルを管理する」](#)

ライセンスファイル リストによる冗長構成

例を示すのが最もわかりやすいでしょう。f1 と f2 の両方に 10 のライセンスが必要な場合、ベンダーは f1 と f2 にそれぞれ、5 つのライセンスを 2 組発行します。サーバ マシンは(3 サーバによる冗長構成と違い)物理的に遠く離れていてもかまいません。

ライセンス ファイルは次のようなものです。

"chicago" へのライセンス 1

```
SERVER chicago 17007ea8 1700
VENDOR sampled /etc/mydaemon
FEATURE f1 sampled 1.000 01-jan-2005 5 26C7DD9C0186
FEATURE f2 sampled 1.000 01-jan-2005 5 8CE46C57041D
```

"tokyo" へのライセンス 2

```
SERVER tokyo 17a07e08 1700
VENDOR sampled /etc/mydaemon
FEATURE f1 sampled 1.000 01-jan-2005 5 16BE40E1D98D
FEATURE f2 sampled 1.000 01-jan-2005 5 6DB6F3E402DF
```

Chicago のユーザは LM_LICENSE_FILE を次のように設定できます。

```
1700@chicago:1700@tokyo
```

Tokyo のユーザは LM_LICENSE_FILE を次のように設定できます。

```
1700@tokyo:1700@chicago
```

ライセンス ファイル名を、UNIX の場合はコロン(:)で区切り、Windows の場合はセミコロン(;)で区切ることに注意してください。アプリケーションはリスト内の最初のサーバにアクセスを試み、何らかの理由で失敗した場合、2 番目のサーバを試します。

3 サーバによる冗長構成

3 サーバによる冗長構成は高品質のネットワークが必要で、3 つのライセンス サーバが物理的に近い場所になければなりません。この形態の冗長構成を実現するにはサーバがハートビートを定期的に交換する必要があるため、通信が不調な場合は十分なパフォーマンスを得られない可能性があります。通信速度の遅い通信回線またはダイヤルアップリンクによってサーバの冗長構成をすべきではありません。

3 サーバによる冗長構成は、ハードウェアのフェールオーバー保護を行うために設計されているのであり、負荷の平衡化を目的としていません。負荷を平衡化するには、この形態ではなく LM_LICENSE_FILE リストを使用してください。3 サーバによる冗長構成では、ライセンスを発行できるのが、3 つのサーバの内の「マスター」と呼ばれる 1 つだけであることがその理由です。したがって、すべてのクライアントは「マスター」との通信が必要なため、各クライアントはその 1 台のマシンへの信頼性の高いネットワークを確立する必要があります。詳細は、「[3 サーバによる冗長構成](#)」を参照してください。

3 サーバによる冗長構成とライセンスファイルリストによる冗長構成を比較する

ライセンスファイル リストによる冗長構成を採用することに何か欠点がありますか？

はい。既定では、FLEX 対応アプリケーションがライセンス サーバからライセンスをチェックアウトすることに成功すると、それ以降はすべて同じライセンス サーバからチェックアウトをしなければなりません。アプリケーションが複数のライセンスを必要とする場合、最初にアクセスしたサーバにライセンスが残っていなければ、別のサーバにライセンスが残っていたとしても、ライセンス要求は拒否されます。アプリケーションが複数の FLEXnet の「ライセンス ジョブ」を使用するようにコーディングされているならば、この制限を回避することができます。アプリケーションがそのようなプログラムされているかどうかを知るには、そのアプリケーションのメーカーに問い合わせる以外に方法はありません。

アプリケーションがライセンスのキューイングをサポートしている場合、ライセンス要求は、すべて最初のライセンス サーバのキューに入り、次のサーバへ移されることはありません。

また、あるサーバが利用できなくなった場合、一部のライセンスは利用不能になります。

3 サーバによる冗長構成のサーバでなく、ライセンスファイル リストを使用する冗長構成を採用する方が望ましいのはどういう場合ですか？

- ライセンス サーバ システムの監視に利用できるシステム管理リソースが不足している場合
- 離れた場所 (例: ロンドンと東京) にある FLEX 対応アプリケーション間で負荷の平衡化が必要な場合、ローカルにサーバを使用できるようにし、リモート サーバはバックアップとして使用できるようにします。
- 定足数に満たない場合、ライセンスファイル リストの方が許容性があります。
- ライセンスファイル リストは、3 つのサーバに限定されません (任意の数のサーバが可能です)。
- ライセンスファイル リストを利用するクライアントは、1 台のマシンへの信頼できるネットワークを確立しなくてもよいので、ネットワーク自体に冗長構成が必要な場合にはこの方法をお勧めします。

カウントされるライセンスとカウントされないライセンス

ライセンス ファイルによって、ライセンス サーバシステムが必要かどうかが決まります。すべての FEATURE (または INCREMENT) 行のライセンス カウント数が 0 (無制限) または uncounted の場合、サーバは必要ありません。このタイプのライセンスは、カウントされないライセンスと呼ばれます。一方、ライセンス カウント数が 0 でない FEATURE 行が存在する場合、サーバはそのライセンスを数えます。ベンダーがサーバなしで FLEXnet Licensing の使用を求めている場合、ベンダーはカウントされないライセンスを発行する必要があります。

なお、ライセンス サーバシステムがカウントされないライセンスを交付することもできます。目的は次のとおりです。

- すべてのライセンス要求のトランザクションをレポート ログに記録することができます。そうすることで、このトランザクションも FLEXnet Manager でレポートできるようになります。
- オプション ファイルの制限をライセンスに適用できます。

カウントされないライセンスを交付するには、ライセンス ファイルに SERVER 行、および SERVER 行の直後に USE_SERVER 行を挿入します。ベンダー デーモンはカウントされないライセンスを交付し、USE_SERVER 行は認証がライセンス サーバ システムによって行われることをアプリケーションに示します。

- uncounted キーワードは、バージョン 6 の FLEXnet Licensing クライアント ライブラリで導入されました。

4

ライセンス管理ツール

FLEXnet Licensing はネットワーク上でのライセンス交付の管理に役立つユーティリティを、ライセンス管理者に提供します。できるだけ最新バージョンのユーティリティを使用してください。これは www.macrovision.com からダウンロードすることができます。表 4-1 にユーティリティの一覧を示します。

表 4-1: ライセンス管理ユーティリティ

ユーティリティ	説明
lmborrow	ライセンスの借用をサポートします。
lmdiag	ライセンスのチェックアウト問題を診断します。
lmdown	ライセンス サーバマシン (または 3 サーバによる冗長構成の場合、3 台すべてのマシン) 上の指定されたライセンス デーモン (lmgxd とすべてのベンダー デーモンの両方) を安全にシャットダウンします。
lmhostid	システムのホスト ID をレポートします。
lminstall	ライセンス ファイルを異なるフォーマットに変換します。
lmnewlog	既存のレポート ログ情報を新たなファイル名に変更し、既存のファイル名で新たなレポート ログ ファイルを作成します。
lmpath	ライセンス ファイルのパス設定を、ユーザが直接コントロールできるようにします。
lmremove	ハンガアップになったライセンスを解放し、使用可能なライセンスのプールに戻します。

表 4-1: ライセンス管理ユーティリティ (続き)

ユーティリティ	説明
<code>lmreread</code>	ライセンス デーモンにライセンス ファイルを再読み込みさせ、すべての新たなベンダー デーモンを起動します。
<code>lmstat</code>	ライセンス サーバ システムのステータスを表示します。
<code>lmswitch</code>	デバッグ ログの格納場所とサイズをコントロールします。
<code>lmswitchr</code>	レポート ログを新たなファイル名に切り替えます。
<code>lmver</code>	ライブラリまたはバイナリ ファイルの FLEXnet Licensing のバージョンをレポートします。

- `lmpath` ユーティリティは、バージョン 7.0 のユーティリティで導入
- `lmborrow` ユーティリティは、バージョン 8.0 のユーティリティで導入
- `lmswitch` ユーティリティは、バージョン 8.0 のベンダー デーモンで導入
- `lmswitchr` ユーティリティは、バージョン 5.0 のベンダー デーモンで導入

管理ツールを実行する

すべての FLEXnet Licensing ユーティリティは、`lmutil` と呼ばれる単一実行ファイルとしてパッケージ化されています。`lmutil` は、個々のコマンド名を持つ `lmutil` のリンクまたはコピーという形で、独立したコマンドとしてインストールされます。または、`lmutil command` として各コマンドを実行するラッパーとしてインストールされます。例としては、`lmutil lmstat`、または `lmutil lmdown` があります。

Windows システムでは、`lmutil command` というフォーマットのコマンドを使用できます。また、このようなコマンドに使用できるグラフィカル ユーザ インターフェイスも提供されています。「[ライセンス管理ツール - Windows 用の LMTOOLS](#)」を参照してください。

lmutil の汎用引数

ほとんどの lmutil ユーティリティに有効な引数は次のとおりです。

表 4-2: lmutil の有効な引数

引数	説明
<code>-c license_file_path</code>	ほとんどの lmutil ユーティリティは、ライセンス ファイルへのパスを指定する必要があります。パスの指定は、 <code>-c license_file_path</code> 引数、または環境変数 <code>LM_LICENSE_FILE</code> の設定で行います。パスが指定されない場合、既定の場所が使用されます。また、このユーティリティには、すべての環境変数 <code>VENDOR_LICENSE_FILE</code> も有効です。一部のユーティリティは、ライセンス ファイル リスト内に、UNIX の場合はコロン、Windows の場合はセミコロンで区切られた複数のライセンス ファイルへのパスを使用できます。スペースを含むパス名は、二重引用符で囲む必要があります。
<code>-help</code>	使用方法を表示して終了します。
<code>-v</code>	ユーティリティの FLEXnet Licensing のバージョンを表示します。
<code>-verbose</code>	発見されたすべてのエラーに関して、詳しい説明を表示します。

- 環境変数 `VENDOR_LICENSE_FILE` は、バージョン 7.0 ユーティリティ以降のユーティリティに有効
- `-verbose` ユーティリティのバージョン 6.0 で導入

lmborrow

`lmborrow` コマンドは `BORROW` 属性を持つライセンスの借用をサポートします。このコマンドはライセンスを借用しているマシン上で実行する必要があります。このコマンドを使用して次のことを実行します。

- 借用時間を設定して借用を開始する。
- 借用時間をクリアする。
- 借用のステータスを確認する。
- 借用したライセンスを期限前に返却する。

借用を開始する

借用を開始するには、コマンドラインから `lmborrow` を実行するか、または LMTOOLS によって借用時間を設定します。

```
lmborrow {vendor | all} enddate [time]
```

以下に詳細を示します。

表 4-3:

引数	説明
<code>vendor</code>	借用するライセンスを交付するベンダーデーモン名、またはそのライセンスサーバシステム内のすべてのベンダーデーモンを指定する <code>all</code>
<code>enddate [time]</code>	<code>dd-mmm-yyyy</code> のフォーマットによるライセンス返却日。 <code>time</code> はオプションであり、FLEX 対応アプリケーションのローカル時間の 24 時間フォーマット (<code>hh:mm</code>) で指定します。 <code>time</code> を指定しないと、指定された終了日の終わりまでチェックアウトが持続します。

例を次に示します。

```
lmborrow sampled 20-aug-2001 13:00
```

この例は、レジストリ (Windows) または `$HOME/.flexlmborrow` (UNIX) 内の `LM_BORROW` に借用期間を設定します。

必要なベンダーのライセンスを借用するには、ユーザが `lmborrow` の実行と *同じマシンで同じ日に*、ライセンスをチェックアウトするアプリケーションを実行します。同じ日に 2 回以上アプリケーションを実行した場合、借用済みのライセンスは重複して借用されません。借用開始日と異なる日にアプリケーションを実行しても、ライセンスは借用されません。

`lmborrow` ユーティリティを使用する他にも借用を開始する方法があります。

- アプリケーション内のインタフェースを使用する (アプリケーションに提供されている場合)。
- 環境変数 `LM_BORROW` を直接設定する。

この 2 つの方法の詳細は、「[ライセンスの借用を開始する](#)」を参照してください。

借用中のライセンスの設定をクリアする



操作手順 レジストリまたは `$HOME/.flexlmborrow` 内の `LM_BORROW` の設定をクリアするには

- コマンド `lmborrow -clear` を発行します。

`LM_BORROW` 設定をクリアすると、再び借用を開始するまで、ライセンスの借用は停止されます。ネットワークとの接続を切る前に、`vendor`によって交付されるその他の機能を借用せずにチェックアウトして、アプリケーションを実行したい場合に、`lmborrow -clear` を、オフラインで使用する機能のライセンスを借用した後に実行します。`LM_BORROW` をクリアしても、既に借用したライセンスのステータスは変更されません。

借用ライセンスのステータスを確認する



操作手順 借用した機能に関する情報を出力するには

- その機能を借用したマシンから次のコマンドを発行します。

```
lmborrow -status
```

ステータスを確認するために借用システムをネットワークに接続する必要はありません。

借用したライセンスを期限前に返却する



操作手順 借用したライセンスを期限前に返却するには

1. 借用システムをネットワークに再接続します。
2. 借用を開始したマシンと同じマシンから次のコマンドを発行します。

```
lmborrow -return [-fqdn][-c license_file_list] [-c display] feature
```

以下に詳細を示します。

表 4-4:

引数	説明
-fqdn	借用システムにアクセスするために、完全修飾ホスト名を使用するように <code>lmborrow</code> に指示します。一般ホスト名ではなく、完全修飾ホスト名に基づいてライセンスが借用された場合、このオプションを使用します。 <code>lmstat</code> を使用して、ライセンスが借用されたときに使用されたホスト名の形式を確認します。
-c <code>license_file_list</code>	指定されたライセンス ファイルを使用します。設定によっては、借用期限前にライセンスを返却するためにライセンス ファイルを指定する必要があります。
-d <code>display</code>	最初にライセンスの借用が行われたディスプレイの指定に使用します。現在のディスプレイが借用を開始したディスプレイと異なる場合に必須です。 Windows では、これはシステム名、またはターミナル サーバのクライアント名(ターミナルサーバ環境の場合)です。UNIX では、これは <code>/dev/ttyxx</code> の形式で存在するか、または X-Display の名前です。
<code>feature</code>	期限前に返却される借用した機能の名前 <code>lmborrow -status</code> を使用して借用した機能名のリストを取得します。

期限前のライセンス返却は、返却ライセンスを交付するベンダー デーモンに対して `LM_BORROW` 設定をクリアする効果があります。

期限前の返却を行うときに借用システムがネットワーク上に戻されていない場合は、ライセンスは返却されず、`LM_BORROW` の設定も変更されません。他に、システムがネットワークに接続する必要があることを知らせるエラー メッセージがエンド ユーザに発行されます。

- 期限前のライセンス返却は、バージョン 8.3 ユーティリティで導入

lmdiag

`lmdiag` を使用すると、ライセンスをチェックアウトできない場合の問題を診断することができます。

使用法

```
lmdiag [-c license_file_list] [-n] [feature[:keyword=value]]
```

以下に詳細を示します。

表 4-5: lmdiag の引数の使用法

引数	説明
-c <i>license_file_list</i>	指定されたファイルを診断します。
-n	非インタラクティブモードで実行します。このモードでは、lmdiag は入力を一切要求しません。このモードでは、拡張接続診断は行えません。
<i>feature</i>	この機能のみを診断します。
<i>keyword=value</i>	ライセンス ファイルが特定の機能に対して複数の行を含む場合、lmdiag がレポートを行う特定の行を選択します。例を次に示します。 lmdiag fl:HOSTID=12345678 ホスト ID が "12345678" の行に対してチェックアウトを試みます。 <i>keyword</i> には、VERSION、HOSTID、EXPDATE、KEY、VENDOR_STRING、ISSUER のいずれかを使用します。

feature が指定されない場合、lmdiag は、リスト内のライセンス ファイルのすべての機能に対して診断を行います。最初に lmdiag はライセンスの情報を出力し、次に各ライセンスのチェックアウトを試みます。チェックアウトに成功した場合、lmdiag はそのことを示します。チェックアウトに失敗した場合、lmdiag は失敗の理由を示します。lmdiag がライセンス サーバシステムに接続できないためにチェックアウトが失敗した場合は、「拡張接続診断」を実行することもできます。

この拡張診断は、ライセンス サーバマシン上の各 TCP/IP ポートに接続を試み、ライセンス ファイル内のポート番号が正しくない場合は、そのことを検出します。lmdiag はリスンされている各 TCP/IP ポート番号を示し、そのポートが *lmgrd* プロセスの場合は、そのことも示されます。lmdiag が、テスト中の機能のベンダー デーモンを見つけた場合、問題を修正するためにライセンス ファイルの正しいポート番号を示します。

関連事項:

[「FLEXLM_DIAGNOSTICS」](#)

lmdown

lmdown ユーティリティを使用すると、すべてのマシン上の選択されたライセンス デーモン (lmgrd と選択されたベンダー デーモンの両方) を安全にシャットダウンすることができます。

使用法

```
lmdown -c license_file_list [-vendor vendor_daemon] [-q] [-all] [-force]
```

以下に詳細を示します。

表 4-6: lmdown の引数の使用法

引数	説明
-c license_file_list	指定されたライセンス ファイルを使用します。lmdown は -c license_file_list の指定を常に推奨します。
-vendor vendor_daemon	このベンダー デーモンのみをシャットダウンします。ただし、lmgrd は実行を継続します。バージョン 6.0 の lmdown と lmgrd が必要です。
-q	入力を要求しません。また、ヘッダを出力しません。これを指定しない場合、lmdown によって "Are you sure? [y/n]:" というメッセージが示されます。
-all	複数のサーバが指定された場合、それらすべてを自動的にシャットダウンします。-q は、-all に暗黙的に含まれます。
-force	ライセンスが借用されている場合、lmdown は、ライセンス サーバシステムが動作しているマシンからのみ、かつユーザが -force を追加した場合に限って実行します。

lmdown が複数のサーバを見つけた場合 (たとえば -c が多くの *.lic ファイルを含むディレクトリを指定した場合) で、-all が設定されていないと、シャットダウンするライセンス サーバシステムの選択肢が表示されます。



注: UNIX では、ライセンス サーバをシャットダウンするために kill -9 を使用しないでください。Windows では、Task Manager を使用して FLEXnet Licensing サービスを終了しなければならない場合には、必ず最初に lmgrd プロセスを終了し、次にすべてのベンダー デーモンのプロセスを終了してください。

3 サーバによる冗長構成のライセンス サーバをシャットダウンするときには、サーバがシャットダウンされるまで 1 分かかります。lmdown は冗長構成のライセンス サーバシステムを構成する 3 つのライセンス サーバシステムすべてをシャットダウンします。冗長構成のライセンス サーバシステム セット内の 1 つのサーバをシャットダウンする必要がある場合 (2 つの障害ポイントになるため、これはお勧めしません)、そのライセンス サーバ マシン上の lmgrd とベンダー デーモン プロセスの両方を強制終了する必要があります。

ライセンス サーバ マネージャ lmgrd を起動するときに、lmdown の未許可の実行を防止できます。サーバをシャットダウンすると、ユーザはライセンスを失います。

関連事項:

lmdown へのアクセスの保安についての詳細は、「[lmgrd コマンドライン構文](#)」を参照してください。
[lmreread](#)

- all オプションは、バージョン 7.0 lmdown ユーティリティで導入
- force オプションは、バージョン 8.0 lmdown ユーティリティで導入

lmhostid

lmhostid ユーティリティは、現在のプラットフォームの FLEXnet Licensing ホスト ID を返します。引数なしで起動された場合は、lmhostid は現在のプラットフォームの既定のホスト ID タイプを表示します。引数を指定すると、現在のプラットフォームでサポートされている場合、要求された type に対応するホスト ID が表示されます。

使用法

```
lmhostid [-n] [-type] [-utf8]
```

以下に詳細を示します。

表 4-7: lmhostid の引数の使用法

引数	説明
-n	ホスト ID そのものだけが、ライセンス ファイル内の HOSTID= に使用できる文字列として返されます。ヘッダ テキストは返されません。

表 4-7: lmhostid の引数の使用法

引数	説明
-type	<p>次のホスト ID タイプの 1 つです。ホスト ID タイプが指定されない場合は、現在のプラットフォームの既定のホスト ID が表示されます。既定のタイプのリストについては、「FLEXnet Licensing がサポートするプラットフォームのホスト ID」を参照してください。</p> <p>プラットフォームに依存するホスト ID</p> <ul style="list-style-type: none">• ether: イーサネットのアドレス• string: 文字列の ID• -vsn: ポリユームのシリアル番号 (Windows の場合のみ)• -flexid: パラレルまたは USB の FLEXid ドングル ID。これは、FLEXid ドングルをサポートしているプラットフォームに対してのみ該当します。完全なリストについては、「FLEXnet Licensing のホスト ID」を参照してください。• -long: 32 ビット ホスト ID <p>プラットフォームに依存しないホスト ID</p> <ul style="list-style-type: none">• -user: 現在のユーザ名• -display: 現在のディスプレイ名。Windows では、これはシステム名、またはターミナル サーバのクライアント名 (ターミナル サーバ環境の場合) です。UNIX では、これは /dev/ttyxx の形式で存在するか、または X-Display の名前です。• -hostname: 現在のホスト名• -internet: ####.####.####.#### 形式の現在のプラットフォームの IP アドレス
-utf8	<p>ホスト ID は、ASCII 文字列ではなく UTF-8 エンコード文字列として出力されます。ホスト ID に、ASCII の A ~ Z, a ~ z, または 0 ~ 9 以外の文字が含まれる場合に、lmhostid とともにこのオプションを使用します。このホスト ID が正しく表示されることを確認するには、UTF-8 エンコード文字列を表示できる「メモ帳」などのユーティリティを使用します。</p>

このコマンドの出力は、次のよう表示されます。

```
lmhostid - Copyright (c) 1989, 2002 Macrovision Corporation  
The FLEXlm hostid of this machine is "69021c89"
```

関連事項:

[「FLEXnet Licensing がサポートするプラットフォームのホスト ID」](#)

lminstall

バージョン 6.0 で導入された lminstall は、主に、10 進フォーマットでライセンスを入力して、可読フォーマットのライセンス ファイルを生成するように設計されています。

使用法

```
lminstall [-i in_lic_file ] [-maxlen n] [-e err_file] [-o out_lic_file] ¥
          [-overfmt {2 | 3 | 4 | 5 | 5.1 | 6 | 7 | 7.1 | 8}] [-odecimal]
```

通常、10 進フォーマットから可読フォーマットに変換する場合、lminstall は引数をとらずに使用され、出力ライセンス ファイル名の入力が必要されます。既定のファイル名は、yyyymmdd.lic という形式の今日の日付です。アプリケーションの既定のライセンス ファイルのディレクトリがソフトウェア ベンダーによって指定されている場合、このファイルをその場所に移動します。指定されていない場合は、LM_LICENSE_FILE または環境変数 VENDOR_LICENSE_FILE を使用して、*.lic ファイルがあるディレクトリを指定します。

入力を終了するには、**q** だけを行に入力するか、空白行を 2 行入力します。

入力ファイルが指定されていて出力ファイルが指定されていない場合は、stdout に出力されます。入力ファイルも出力ファイルも指定されていない場合、lminstall は、入力が stdin から行われると想定し、出力ファイル名の入力をユーザに要求します。

lminstall は、ライセンスを可読フォーマットから 10 進フォーマットに変換したり、異なるバージョンの FLEXnet ライセンス フォーマットに変換することもできます。



操作手順 可読形式から 10 進フォーマットに変換するには

```
lminstall -i in_lic_file -o out_lic_file -odecimal
```



操作手順 v5.1 フォーマットに変換するには

```
lminstall -i in_lic_file -o out_lic_file -overfmt 5.1
```



操作手順 行の長さを最大(たとえば 50 文字)にするには

```
lminstall -maxlen 50
```

必要に応じて変換エラーをレポートし、`-e err_file` を指定してそのレポートをファイルに書き込ませることができます。lminstall は最大 1000 行まで入力できます。

lmnewlog

lmnewlog ユーティリティは、既存のレポート ログ情報を新たなファイルに移動し、元のレポート ログ ファイルの名前を使用して新たなレポート ログを作成し、レポート ログ ファイルを切り替えます。lmswitchr ではなく lmnewlog を使用してレポート ログを切り替える場合は、ベンダーのオプション ファイルの REPORTLOG 行内のファイル名を変更する必要はありません。これには、バージョン 7.1 以降のベンダー デーモンが必要です。

使用法

```
lmnewlog [-c license_file_list] feature renamed_report_log
```

または

```
lmnewlog [-c license_file_list] vendor renamed_report_log
```

以下に詳細を示します。

表 4-8: lmnewlog の引数の使用法

引数	説明
<code>-c license_file_list</code>	指定されたライセンス ファイルを使用します。
<code>feature</code>	このライセンス ファイル内の任意の機能
<code>vendor</code>	このライセンス ファイル内のベンダー デーモン
<code>renamed_report_log</code>	既存のレポート ログ情報が移動される新たなファイルのパス

Impath

Impath ユーティリティを使用すると、FLEXnet のライセンス パスの設定を直接コントロールできます。このユーティリティは、現在のライセンス パス設定への追加、上書き、またはその取得のために使用されます。

使用法

```
lmpath {-add | -override} {vendor | all} license_file_list
```

以下に詳細を示します。

表 4-9: lmpath の引数の使用法

引数	説明
-add	license_file_list を現在のライセンスファイルリストに追加します。または、このリストが存在しない場合は、リストを license_file_list に初期化して、ライセンスファイル リストを作成します。重複するものは破棄されます。
-override	既存のライセンスファイルリストを license_file_list で上書きします。license_file_list が空の文字列 "" の場合、指定されたリストは削除されます。 <ul style="list-style-type: none"> • lmpath -override all "": LM_LICENSE_FILE の値を削除します。 • lmpath -override vendor "": VENDOR_LICENSE_FILE の値を削除します。
vendor	ベンダー デーモン名。VENDOR_LICENSE_FILE の値に影響します。
all	すべてのベンダー デーモンを参照します。LM_LICENSE_FILE の値に影響します。
license_file_list	UNIX ではコロン (:)、Windows ではセミコロン (;) で区切られるリスト。license_file_list が空の文字列 "" の場合、指定されたエントリは削除されます。



注: lmpath は、Windows では FLEXnet Licensing レジストリ エントリを設定することで、UNIX では \$HOME/.flexlmrc を設定することで動作します。



操作手順 現在のライセンス パスを表示するには

```
lmpath -status
```

次のように表示されます。

```
lmpath - Copyright (C) 1989-2002 Macrovision Corporation
Known Vendors:
```

```
demo:      ./counted.lic:./uncounted.lic
```

Other Vendors:

/usr/local/flexlm/licenses/license.lic

パスにディレクトリが設定されている場合、個々の *.lic ファイルがすべて一覧表示されることに注意してください。

lmremove

lmremove ユーティリティを使用すると、特定の機能に対してユーザ ライセンスを個別に回収できます。アプリケーションがアクティブの場合、ライセンスが lmremove によって解放された後、アプリケーションはライセンスを再びチェックアウトします。

使用法

```
lmremove [-c license_file_list] feature user user_host display
```

または

```
lmremove [-c license_file_list] -h feature server_host port handle
```

以下に詳細を示します。

表 4-10: lmremove の引数の使用法

引数	説明
<code>-c license_file_list</code>	ライセンス ファイルを指定します。
<code>feature</code>	ユーザによってチェックアウトされる機能の名前
<code>user</code>	lmstat -a によってレポートされる、ライセンスが回収されるユーザの名前
<code>user_host</code>	lmstat -a によってレポートされる、ユーザがログインしているホストの名前
<code>display</code>	lmstat -a によってレポートされる、ユーザが作業しているディスプレイの名前
<code>server_host</code>	ライセンス サーバシステムが起動しているホストの名前
<code>port</code>	lmstat -a によってレポートされる、ライセンス サーバが起動している TCP/IP ポート番号
<code>handle</code>	lmstat -a によってレポートされるライセンス ハンドル

`user`、`user_host`、`display`、`server_host`、`port`、`handle` の情報は、`lmstat -a` の出力から取得する必要があります。

`lmremove` は、`user_host` および `display` 上のすべての `user` のインスタンスが、`feature` を使用できないようにします。オプションの `-c license_file_list` が指定された場合、示されたファイルは、ライセンス ファイルとして使用されます。

`-h` オプションは、`lmstat -a` がレポートする `server_host`、`port`、`license handle` を使用します。`lmstat -a` が出力した次の例を見てください。

```
joe nirvana /dev/tty5 (v1.000) (cloud9/7654 102), start Fri 10/29  
18:40
```

この例では、ユーザは `joe`、ユーザのホストは `nirvana`、ディスプレイは `/dev/tty5`、サーバのホストは `cloud9`、TCP/IP ポートは `7654`、ライセンス ハンドルは `102` です。



操作手順 このライセンスを回収するには、次のコマンドのいずれかを発行します。

```
lmremove fl joe nirvana /dev/tty5
```

または

```
lmremove -h fl cloud9 7654 102
```

ハンドルによって回収する場合、ライセンスが重複ライセンスとしてグループ化されていると、すべての重複したライセンスも回収されます。ライセンスの保持が設定されている場合、`lmremove` を使用してライセンスの返還を要求すると、`lmremove` が開始されますが、ライセンスの保持時間は変更されません。

ライセンス サーバ マネージャ `lmgrd` を起動するときに、ユーザのライセンスの回収が問題を引き起さないように、`lmremove` の未許可の実行を防止できます。

関連事項:

`lmremove` へのアクセスの保安についての詳細は、「[lmgrd コマンドライン構文](#)」を参照してください。

lmreread

`lmreread` ユーティリティは、ライセンス サーバ マネージャにライセンス ファイルを再読み込みさせ、追加されたすべての新たなベンダー デモンを起動します。さらに、起動中のすべてのデモンには、機能のライセンス情報とオプション設定の変更を反映するため、ライセンス ファイルとエンド ユーザ オプション ファイルを再読み込みするようシグナルが送られます。レポート ログが有効な場合、ベンダーデーモン内部のデータ バッファにその時点まで存在するすべてのレポート ログ データがフラッシュされます。`lmreread` はサーバ マシンのホスト名に対する変化を認識しますが、サーバの TCP/IP ポート番号の変更には使用できません。

オプションでベンダー デーモン名が指定された場合、その名前のデーモンのみがライセンス ファイルとエンド ユーザのオプションファイルを再読み込みします(ただしこの場合、lmgrd はライセンス ファイルを再読み込みしません)。

使用法

```
lmreread [-c license_file_list] [-vendor vendor] [-all]
```

以下に詳細を示します。

表 4-11: lmreread の引数の使用法

引数	説明
-c license_file_list	指定されたライセンス ファイルを使用します。
-vendor vendor	ベンダー デーモン vendor のみがライセンス ファイルおよびそのオプションファイルを再読み込みします。また、lmgrd は必要に応じて vendor を再起動します。
-all	複数の lmgrd が指定された場合、すべての lmgrd に再読み込みを命令します。



注: -c license_file_list オプションを使用した場合、指定されたライセンス ファイルは、lmgrd ではなく lmreread によって読み込まれます。lmgrd は、当初読み込んだファイルを再読み込みします。

ライセンス サーバ マネージャ lmgrd を起動するときに、lmreread の未許可の実行を防止できます。

関連事項:

lmreread へのアクセスの保安についての詳細は、「[lmgrd コマンドライン構文](#)」を参照してください。ベンダー デーモンが自身のオプションファイルを再読み込みする機能は、バージョン 8.0 のベンダー デーモンで導入

lmstat

lmstat ユーティリティは、すべてのネットワーク ライセンス交付のステータスを監視するのに役立ちます。次にその監視対象を示します。

- 起動中のデーモン
- ライセンス ファイル
- 個々の機能のユーザ
- 特定のベンダー デーモンが交付する機能のユーザ
- 借用された BORROW ライセンス

lmstat は、ライセンス サーバシステムから受け取った情報を出力します。したがって、カウントされないライセンスなどの直接利用型ライセンスに関してはレポートされません。カウントされないライセンスについてレポートするには、該当ライセンスが交付型のライセンス ファイルに追加されており、`@host`、`port@host` または `USE_SERVER` が、そのライセンス ファイルに対してライセンス サーバシステムを使用するよう、アプリケーションに指示していなければなりません。また、lmstat は、キューに入れられたユーザや重複のグループ化のために共有されたライセンスも返しません。

使用法

```
lmstat [-a] [-c license_file_list] [-f [feature]] [-i [feature]] [-s [server]]
        [-s [vendor]] [-t timeout_value]
```

以下に詳細を示します。

表 4-12: lmstat の引数の使用法

引数	説明
-a	すべての情報を表示します。
-c license_file_list	指定されたライセンス ファイルを使用します。
-f [feature]	feature のユーザを一覧表示します。feature が指定されていない場合は、すべての機能の使用に関する情報が表示されます。
-i [feature]	指定した feature、または feature が指定されていない場合はすべての機能について FEATURE/INCREMENT 行の情報を表示します。
-s [server]	server、または server が指定されていない場合はすべてのサーバ上で、\$VENDOR_LICENSE_FILE または \$LM_LICENSE_FILE にリストされたすべてのライセンス ファイルのステータスを表示します。

表 4-12: lmstat の引数の使用法

引数	説明
-S [<i>vendor</i>]	<i>vendor</i> の機能のユーザをすべて一覧表示します。
-t <i>timeout_value</i>	接続のタイムアウトを <i>timeout_value</i> に設定します。 lmstat が <i>server</i> に接続するときに消費する時間の長さを制限します。

lmstat -a の出力は、次のようなものです。

```
License server system status: 27000@myhost1
License file(s) on myhost: install_dir/flexlm/v9.3/sun4_u5/
counted.lic:
myhost: license server system UP (MASTER) v9.3
Vendor daemon status (on myhost1):

demo: UP v9.3
Feature usage info:
Users of f1: (Total of 4 licenses issued; Total of 1 license in use)
  "f1" v1.0, vendor: demo
  floating license
    daniel myhost2 19.36.18.26 (v1.0) (myhost1/27000 102), start Fri
    5/3 7:29
```

以下に詳細を示します。

表 4-13: lmstat の出力

出力	引数	説明
daniel	<i>user</i>	ユーザの名前
myhost2	<i>user_host</i>	ユーザが作業を実行中のホスト
19.36.18.26	<i>display</i>	ユーザが作業を実行中のディスプレイ
v1.0	<i>version</i>	機能のバージョン
myhost1	<i>server_host</i>	ライセンス サーバシステムが起動中のホスト
27000	<i>port</i>	ライセンス サーバシステムが起動中の <i>server_host</i> 上の TCP/IP ポート
102	<i>handle</i>	ライセンスのハンドル
start Fri 5/3 7:29	<i>checkout_time</i>	このライセンスがチェックアウトされた時間

`user`, `user_host`, `display`, `server_host`, `Apport`, `handle` の情報は、`lmremove` でライセンスを回収するときに使用されます。



注: `lmstat -a` は、コンピュータ資源を大量に消費する可能性のあるコマンドです。多くのユーザがこのコマンドを使用すると、ネットワークに大きな負荷が生じます。

- バージョン 8.3 以前の `lmstat` ユーティリティで使用できる `-A` オプションを使用して、使用中のすべてのユーザの一覧表示が可能。

lmswitch

`lmswitch` ユーティリティは、特定のベンダー デーモンの既存のデバッグ ログを閉じ、新しいファイル名でそのベンダー デーモンのデバッグ ログを開始することによって、そのベンダー デーモンが書き込むデバッグ ログ ファイルを切替えます。また、既存のデバッグ ログ ファイルがない場合は、そのベンダー デーモンによって書き込まれる新しいデバッグ ログ ファイルも開始します。

使用法

```
lmswitch [-c license_file_list] vendor new_debug_log
```

以下に詳細を示します。

表 4-14: `lmswitch` の引数

引数	説明
<code>-c license_file_list</code>	指定されたライセンス ファイルを使用します。
<code>vendor</code>	このライセンス ファイル内のベンダー デーモン
<code>new_debug_log</code>	新たなデバッグ ログ ファイルへのパス

既定では、`lmgrd` から出力されるデバッグ ログと、その `lmgrd` が起動したすべてのベンダー デーモンは、同じデバッグ ログ ファイルに書き込まれます。

このベンダー デーモンのデバッグ ログ出力が、既に別のファイルに出力するように命令されている場合、`lmswitch` は、現在のデバッグ ログ ファイルを閉じ、デバッグ ログ出力ファイル `new_debug_log` に書き込みを始めるようベンダー デーモンに指示します。デバッグ ログ出力が、このベンダー デーモンの別のファイルに出力するように命令されていない場合、`lmswitch` は、そのデバッグ ログ出力をファイル `new_debug_log` に書き込みを始めるようベンダー デーモンに指示します。



注: lmswitch はベンダー デーモンがシャット ダウンされるか、そのオプション ファイルが lmreread によって再読み込みされるまで有効です。ベンダー デーモンが再起動されるか、そのオプション ファイルが再読み込みされる場合、これはオプション ファイル内の DEBUGLOG 行を探し、そのデバッグ ログ出力をベンダー デーモン自身のファイルに書き込むかどうか、また書き込むとすればどのファイルに書き込むかを決定します。

関連事項:

[「DEBUGLOG」](#)

[「lmreread」](#)

[「デバッグ ログ ファイル」](#)

lmswitchr

lmswitchr ユーティリティは、既存のレポート ログを閉じ、新たなファイル名で新たなレポート ログを作成することによって、レポート ログ ファイルを切り替えます。また、レポート ログがまだ存在しない場合、新たなレポート ログ ファイルを作成します。

使用法

```
lmswitchr [-c license_file_list] feature new_report_log
```

またはバージョン 5.0 以降のベンダー デーモンの場合は次のように記述します。

```
lmswitchr [-c license_file_list] vendor new_report_log
```

以下に詳細を示します。

表 4-15: lmswitchr の引数の使用法

引数	説明
<code>-c license_file_list</code>	指定されたライセンス ファイルを使用します。
<code>feature</code>	このライセンス ファイル内の任意の機能
<code>vendor</code>	このライセンス ファイル内のベンダー デーモン
<code>new_report_log</code>	新たなレポート ログ ファイルへのパス

レポート ログがベンダー デーモンに対して有効でない場合、lmswitchr は、レポート ログ出力を `new_report_log` に書き込むよう、そのベンダー デーモンに指示します。レポート ログがベンダー デーモンに対して既に有効な場合、lmswitchr は、既存のレポート ログ ファイルを閉じて、新たなレポート ログ 出力を `new_report_log` に書き込むよう、そのベンダー デーモンに指示します。



注: `lmswitchr` はベンダー デーモンがシャットダウンされるか、そのオプション ファイルが `lmreread` によって再読み込みされるまで有効です。ベンダー デーモンが再起動されるか、そのオプション ファイルが再読み込みされる場合、これはオプション ファイル内の `REPORTLOG` 行を探し、そのレポート ログ出力をベンダー デーモンのファイルに書き込むかどうか、また書き込むとすればどのファイルに書き込むかを決定します。

関連事項:

[「REPORTLOG」](#)

[「lmnewlog」](#)

[「lmreread」](#)

[「レポート ログ ファイル」](#)

lmver

`lmver` ユーティリティは、ライブラリまたはバイナリ ファイルの FLEXnet Licensing のバージョンをレポートします。

使用法

`lmver filename`

`filename` には次の 1 つが含まれます。

- FLEXnet Licensing でビルドされた実行可能ファイルの名前
- `lmgrd`
- ライセンス管理ツール
- ベンダー デーモン

たとえば、`spell` というアプリケーションがある場合、`lmver spell` と入力します。

ライセンス管理ツール – Windows 用の LMTOOLS

32 ビット Windows プラットフォームには、ライセンス サーバ マネージャ ツールのグラフィカル ユーザ インタフェースが LMTOOLS という名前で提供されています。できるだけ最新バージョンの LMTOOLS を使用してください。これは www.macrovision.com からダウンロードすることができます。

LMTOOLS には主に次のような機能があります。

- FLEXnet ライセンス サーバシステムの起動、停止、設定
- ホスト ID を含むシステム情報の取得
- サーバステータスの取得

LMTOOLS には、ライセンス サーバシステムを設定する 次の 2 つのモードがあります。

- ライセンス ファイルを使用する設定
- サービスを使用する設定

ライセンス ファイルを使用する設定

操作は、特定のライセンス ファイル上で行われます。そのファイルはローカルまたはリモートのファイルです。このモードでは、lmgrd プロセスを起動することはできませんが、他のことはすべて行えます。



操作手順 このモードを設定するには

1. LMTOOLS を起動します。
2. [Configuration using License File] ボタンをオンにします。
3. 1 つまたは複数のライセンス ファイル名を入力するか、または `port@host` を指定します。

サービスを使用する設定

操作はサービスに対して実行され、lmgrd を、LMTOOLS が起動中のシステムのローカルプロセスとして起動できます。サービス設定の詳細については、「[Windows サービスとしてライセンス サーバ マネージャを設定する](#)」を参照してください。

オプション ファイル

オプション ファイルを使用すると、ライセンス管理者は、FLEXnet Licensing のさまざまな操作パラメータをコントロールできます。ユーザは、ユーザ名、ホスト名、ディスプレイ、IP アドレス、または環境変数 LM_PROJECT によって設定される PROJECT によって識別されます。

同時使用 (フローティング) ライセンスに対して、ライセンス管理者は次のことができます。

- 機能の使用許可
- 機能の使用拒否
- ライセンスの予約

同時使用ライセンスは、ライセンス ファイルまたは高信頼性ストレージ内のフルフィルメント レコードに格納できます。

アクティベーション可能なライセンスに対して、ライセンス管理者は次のことができます。

- 特定のフルフィルメントレコード内のライセンスのアクティベーション許可
- 特定のフルフィルメントレコード内のライセンスのアクティベーション拒否

すべてのライセンスに対して、ライセンス管理者は次のことができます。

- 利用可能なライセンス数の制限
- ライセンス使用に関するログ情報量のコントロール
- レポート ログ ファイルの有効化

ライセンス管理者は、オプション ファイルを使用して、ライセンス許可の基準を自由に設定できます。

オプション ファイル行に書き込める文字数は、最大 2048 文字までです。文字 ¥ はオプション ファイル行の継続文字です。

- 環境変数 LM_PROJECT で設定するオプション ファイルの PROJECT ID は、バージョン 7.0 のベンダー デーモンで導入
- 高信頼性ストレージ内のフルフィルメント レコードに格納されているライセンス に対するオプション ファイルのコントロールは、11.3 のベンダー デーモンで導入

オプション ファイルを作成する



操作手順 オプション ファイルを作成するには

1. 「オプション ファイルの構文」に記載されている適切なオプションを使用して、任意のテキスト エディタでベンダー デーモンのオプション ファイルを作成します。
2. オプション ファイルを任意の場所に置きます。ただし、オプション ファイルをライセンス ファイルと同じディレクトリに置くことをお勧めします。
3. オプション ファイルへのパスを、アプリケーションのベンダー デーモンに関する VENDOR 行の 4 番目のフィールドとしてライセンス ファイルに追加します。例を次に示します。

```
VENDOR sampled /etc/sampled ¥  
[options=]/sample_app/sampled/licenses/sampled.opt
```

これにより、ベンダー デーモン sampled は、指定されたオプション ファイルを参照できます。

このパスを省略すると、ベンダー デーモンは次の規準に従って自動的にファイルを検索します。

- vendor.opt という名前のファイル (vendor はベンダー デーモンの名前)
- lmgrd が使用するライセンス ファイルが格納されているディレクトリ



注: 既定のオプション ファイル名 vendor.opt は、バージョン 6 のベンダー デーモンで導入

オプションファイルの構文

オプションファイルの構文の概要を次に説明します。例と詳細情報については、「[オプションファイルの例](#)」を参照してください。

ファイルの各行は、オプションを1つコントロールします。表 5-1 にオプション キーワードの一覧を示します。

表 5-1: オプション キーワード

オプション キーワード	説明
<code>「BORROW_LOWWATER」</code>	借用できない BORROW ライセンスの数を設定します。
<code>「DEBUGLOG」</code>	このベンダー デーモンに関するデバッグ ログ情報を指定したファイルに書き込みます (バージョン 8.0 以降のベンダー デーモン)。
<code>「EXCLUDE」</code>	ユーザが機能にアクセスするのを拒否します。
<code>「EXCLUDE_BORROW」</code>	ユーザが BORROW ライセンスを借用する機能を拒否します。
<code>「EXCLUDE_ENTITLEMENT」</code>	高信頼性ストレージ内のフルフィルメントレコードに格納されているライセンスを、ユーザがアクティベーションするのを拒否します。
<code>「EXCLUDEALL」</code>	このベンダー デーモンによって提供されるすべての機能にユーザがアクセスするのを拒否します。
<code>「FQDN_MATCHING」</code>	ホスト名の一致のレベルを設定します。
<code>「GROUP」</code>	任意のオプションの使用に対してユーザグループを定義します。
<code>「GROUPCASEINSENSITIVE」</code>	GROUP および HOST_GROUP キーワードに指定されたユーザおよびホストのリストに対して、大文字と小文字の区別を設定します。
<code>「HOST_GROUP」</code>	任意のオプションの使用に対してホストグループを定義します (バージョン 4.0以降)。
<code>「INCLUDE」</code>	ユーザが機能を使用するのを許可します。
<code>「INCLUDE_BORROW」</code>	ユーザが BORROW ライセンスを借用するのを許可します。
<code>「INCLUDE_ENTITLEMENT」</code>	高信頼性ストレージ内のフルフィルメントレコードに格納されているライセンスを、ユーザがアクティベーションするのを許可します。

表 5-1: オプション キーワード (続き)

オプション キーワード	説明
<code>[INCLUDEALL]</code>	このベンダー デーモンによって提供されるすべての機能をユーザが使用するのを許可します。
<code>[LINGER]</code>	チェックイン後にユーザが機能の保持時間を延長するのを許可します。
<code>[MAX]</code>	グループに対して特定の機能の使用を制限し、ユーザ間での使用の優先順位を調整します。
<code>[MAX_BORROW_HOURS]</code>	指定された機能の最長借用期間を変更します。
<code>[MAX_OVERDRAFT]</code>	超過使用数を、ライセンス ファイルで指定された数未満に制限します。
<code>[NOLOG]</code>	デバッグ ログ ファイル内の特定項目の記録を停止します。
<code>[REPORTLOG]</code>	ライセンス使用状況のレポーティング ツールである FLEXnet Manager が使用できるレポート ログ ファイルを作成するように指定します。
<code>[RESERVE]</code>	ユーザまたはユーザとホストのグループにライセンスを予約します。
<code>[TIMEOUT]</code>	機能に対してアイドル タイムアウトを指定し、他のユーザが使用できるように機能をフリー プールに返します。
<code>[TIMEOUTALL]</code>	すべての機能にタイムアウトを設定します。

コメント

オプション ファイル内にコメントを記入するには、各コメント行をシャープ記号 (#) で始めます。

機能を指定する

オプション ファイル エントリ内で使用するとき、キーワードと値のペアをオプションで使用して機能名を修飾し、機能を限定することができます。この表記法は、1 つの機能に対して複数の FEATURE 行がある場合に、ライセンスの特定のグループを区別するのに使用されます。構文は次のとおりです。

```
feature:keyword=value
```

例を次に示します。

```
f1:VERSION=2.0
```

機能 f1 のバージョン 2.0 のライセンス プールを指定します。



注: コロン(:)は有効な機能名の文字です。コロンが機能名に含まれる場合は、クォーテーションマークとスペースを使用した次の代替構文でライセンスのグループを指定します。

```
"feature keyword=value"
```

次のオプション キーワードは機能名の修飾語として使用され、ライセンスの特定のグループを示します。

- VERSION=
- HOSTID=
- EXPDATE=
- KEY=
- SIGN=
- ISSUER=
- NOTICE=
- VENDOR_STRING= (プール識別要素としてベンダーによって設定されている場合)
- dist_info=
- user_info=
- asset_info=

キーワード USER_BASED または HOST_BASED が機能行に存在する場合は、この機能指定構文を使用しなければなりません。

機能名の代わりにパッケージ名を使用すると、そのオプションがパッケージ内のすべてのコンポーネントに適用されます。

タイプを使用してライセンスの制限を指定する

いくつかのオプションキーワードは、ライセンスを使用できるユーザや、ライセンスを使用できる場所を制限します。これらのオプションは、その制限が何に基づくかを指定する引数 `type` をとります。

オプション キーワード EXCLUDE、EXCLUDE_ENTITLEMENT、EXCLUDEALL、EXCLUDE_BORROW、INCLUDE、INCLUDE_ENTITLEMENT、INCLUDEALL、INCLUDE_BORROW、MAX、RESERVE を使用するとき、次の値をタイプに使用できません。

- **USER:** FLEX 対応アプリケーションを実行しているユーザのユーザ名。ユーザ名は大文字と小文字が区別されます。
- **HOST:** アプリケーションを実行しているホストマシン名または IP アドレス。ホスト名は大文字と小文字が区別されます。IP アドレスには、ワイルドカード文字を使用できません。

IP アドレスには、ワイルドカード文字を使用できます。

オプション キーワード EXCLUDE、EXCLUDEALL、EXCLUDE_BORROW、INCLUDE、INCLUDEALL、INCLUDE_BORROW、MAX、RESERVE を使用するときは、次の値をタイプに使用できます。

- **DISPLAY:** アプリケーションが表示されているディスプレイ。UNIX では、DISPLAY は /dev/ttyxx (アプリケーションがバックグラウンドで実行中は常に /dev/tty)、または X-Display の名前です。Windows では、これはシステム名、またはターミナル サーバのクライアント名です(ターミナル サーバ環境の場合)。ディスプレイ名は大文字と小文字が区別されます。
- **INTERNET:** アプリケーションを実行中のマシンの IP アドレス (IP アドレス内でワイルドカード文字を使用できます)。
- **PROJECT:** FLEX 対応アプリケーションを実行しているユーザによって設定される環境変数 LM_PROJECT。プロジェクト名は大文字と小文字が区別されます。

ターミナル サーバがない Windows システムでは、HOST 名と DISPLAY 名はどちらも Windows のシステム名に設定されます。ターミナルサーバからチェックアウトできるライセンス (TS_OK キーワードが機能行に存在) に対しては、USER 名、HOST 名、DISPLAY 名が互いに異なることがあります。

上記のタイプは単一のメンバーをとります。例を次に示します。

```
EXCLUDE coolsoft USER joe
```

ユーザまたはホストのリストを指定するには、最初に GROUP または HOST_GROUP オプション行を使用してリストを定義し、次に GROUP または HOST_GROUP タイプを使用してグループ名を指定します。例を次に示します。

```
GROUP stars joe barbara susan  
EXCLUDE coolsoft GROUP stars
```

- HOST を指定する IP アドレスは、バージョン 8 のベンダー デーモンで導入
- 環境変数 LM_PROJECT は、バージョン 5 クライアント ライブラリとベンダー デーモンで導入
- 機能名内のコロンは、バージョン 8 のベンダー デーモンで導入

BORROW_LOWWATER

このオプションは、ライセンス ファイルに格納されているライセンスに使用されます。ライセンスが高信頼性ストレージ内で利用可能なときは、通常は BORROW ではなく アクティベーションが提供されます。

```
BORROW_LOWWATER feature[:keyword=value] n
```

借用できない BORROW 機能のライセンスの数を設定します。

表 5-2: BORROW_LOWWATER の項目

項目	説明
<i>feature</i>	影響を受ける機能の名前
<i>keyword=value</i>	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。
<i>n</i>	ライセンス借用機能では借用できないライセンスの数

たとえば、機能 "f1" が 10 のカウントを持ち、アプリケーション内で借用が可能であり、FEATURE 行が次のように記述されているとします。

```
FEATURE f1 ... 10 ... BORROW SIGN=...
```

オプションファイルに次の行を追加すると、7つのライセンスの借用のみが許可されます。

```
BORROW_LOWWATER f1 3
```

DEBUGLOG

```
DEBUGLOG [+] debug_log_path
```

このオプションファイルに関連付けられているベンダー デーモンから出力するデバッグ ログの場所を指定します。 *debug_log_path* の前に + 記号を付けることにより、ログ エントリが順に付け足されます。この記号を付けないと、デーモンが起動されるたびに、ファイルが上書きされます。これは、このオプションに関連付けられたベンダー デーモンの出力だけに影響することに注意してください。同じライセンス ファイル内の lmgrd およびその他のベンダー デーモンのデバッグ ログ出力はこのファイルにキャプチャされません。

Windows では、スペースを含むパラメータを二重引用符で囲む必要があります。lmgrd をサービスとして開始した場合、完全パスが指定されていない限り、レポート ログ ファイルの既定の場所は、`c:\¥winnt¥System32` フォルダです。

関連事項:

「[Windows サービスとしてライセンス サーバ マネージャを設定する](#)」

「[lmswitch](#)」

「[デバッグ ログ ファイル](#)」: ベンダー デーモンの出力のみに制限されるデバッグ ログ出力は、バージョン 8 のベンダー デーモンで導入

EXCLUDE

このオプションは、ライセンス ファイルおよび高信頼性ストレージに格納されている同時使用ライセンスに対して適用されます。

```
EXCLUDE feature[:keyword=value] type {name | group_name}
```

機能を使用することが許可されているユーザのリストから、ユーザまたはあらかじめ定義されているユーザ グループを除外します。EXCLUDE は INCLUDE を無効にします。EXCLUDE のリストと INCLUDE のリストで矛盾が生じた場合は、EXCLUDE を優先して解決します。

表 5-3: EXCLUDE の項目

項目	説明
<i>feature</i>	影響を受ける機能またはパッケージの名前
<i>keyword=value</i>	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。
<i>type</i>	USER、HOST、DISPLAY、INTERNET、PROJECT、GROUP、HOST_GROUP のいずれか 1 つ。詳細は、「 タイプを使用してライセンスの制限を指定する 」を参照してください。
<i>name</i>	ライセンスの使用を除外するタイプ <i>type</i> の項目名
<i>group_name</i>	除外するグループの名前



操作手順 機能 f1 を使用できるユーザのリストからユーザ hank を除外するには、次のように記述します。

```
EXCLUDE f1 USER hank
```

EXCLUDE_BORROW

このオプションは、ライセンス ファイルに格納されているライセンスに使用されます。ライセンスが高信頼性ストレージ内で利用可能なときは、通常は BORROW ではなくアクティベーションが提供されます。


```
EXCLUDE_BORROW feature[:keyword=value] type ¥
                {name | group_name}
```

このBORROW機能のライセンスを借用することが許可されているユーザのリストから、ユーザまたは事前定義されているユーザグループを除外します。

EXCLUDE_BORROWのリストとINCLUDE_BORROWのリストで矛盾が生じた場合は、EXCLUDE_BORROWを優先して解決します。

表 5-4: EXCLUDE_BORROW の項目

項目	説明
<i>feature</i>	影響を受ける機能の名前
<i>keyword=value</i>	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。
<i>type</i>	USER、HOST、DISPLAY、INTERNET、PROJECT、GROUP、HOST_GROUP のいずれか 1 つ。詳細は、「 タイプを使用してライセンスの制限を指定する 」を参照してください。
<i>name</i>	ライセンスの借用を除外するタイプ <i>type</i> の項目名
<i>group_name</i>	借用から除外するグループの名前



操作手順

BORROW 属性を持つ機能 *f1* を借用できるユーザのリストからユーザ *fred* を除外するには、次のように記述します。

```
EXCLUDE_BORROW f1 USER fred
```

EXCLUDE_ENTITLEMENT

このオプションは、高信頼性ストレージ内に格納されていて、かつアクティベーションを使用して提供されるライセンスに対してのみ適用されます。

```
EXCLUDE_ENTITLEMENT entitlementId type {name | group_name}
```

高信頼性ストレージ内のフルフィルメントレコードに格納されているライセンスのアクティベーションを許可されているユーザのリストから、ユーザまたは事前定義されているユーザグループなどを除外します。EXCLUDE_ENTITLEMENTはINCLUDE_ENTITLEMENTを無効にします。EXCLUDE_ENTITLEMENTのリストとINCLUDE_ENTITLEMENTのリストで矛盾が生じた場合は、EXCLUDE_ENTITLEMENTを優先して解決します。

表 5-5: EXCLUDE_ENTITLEMENT の項目

項目	説明
<i>entitlementId</i>	ライセンス アクティベーションの要求時に使用される資格 ID
<i>type</i>	USER、HOST、GROUP、HOST_GROUP のいずれか 1 つ。詳細は、「 タイプを使用してライセンスの制限を指定する 」を参照してください。
<i>name</i>	ライセンスの使用を除外するタイプ <i>type</i> の項目名
<i>group_name</i>	除外するグループの名前



重要: 資格 ID が "AB456" のフルフィルメントレコードに提供されているライセンスをアクティベーションできるユーザのリストから、ユーザ "pete" を除外するには、次のように記述します。

```
EXCLUDE_ENTITLEMENT AB456 USER pete
```

EXCLUDEALL

このオプションは、ライセンス ファイルおよび高信頼性ストレージに格納されている同時使用ライセンスに対して適用されます。

```
EXCLUDEALL type {name | group_name}
```

このベンダー デーモンによって提供されるすべての機能を使用することが許可されているユーザのリストから、ユーザまたは事前定義されているユーザグループを除外します。

表 5-6: EXCLUDEALL の項目

項目	説明
<i>type</i>	USER、HOST、DISPLAY、INTERNET、PROJECT、GROUP、HOST_GROUP のいずれか 1 つ。詳細は、「 タイプを使用してライセンスの制限を指定する 」を参照してください。
<i>name</i>	ライセンスの使用を除外するタイプ <i>type</i> の項目名
<i>group_name</i>	除外するグループの名前

マシン "chaos" 上のすべてのユーザが、このベンダー デーモンによって提供されるすべての機能を使用できなくするには、次のように記述します。

EXCLUDEALL HOST chaos

FQDN_MATCHING

このオプションは、ライセンスファイルまたは高信頼性ストレージに格納されているすべてのライセンスに対して適用されます。

FQDN_MATCHING exact | lenient

HOST タイプの指定で使用されたホスト名と、FLEX 対応アプリケーションが送信したホスト名の一致レベルを設定します。アプリケーションは、HOST タイプ指定を使用して検証するために、そのホスト名または完全ドメイン名 (FQDN) のいずれかをベンダーデーモンに送信するように設定されています。ベンダーに問い合わせ、完全ドメイン名をサポートしているかどうかを確認してください。

表 5-7: FQDN_MATCHING の項目

項目	説明
exact	HOST タイプ指定のホスト名は、アプリケーションが送信した内容および形式と一致していなければなりません。これは、既定の設定です。
lenient	アプリケーションが送信したホスト名は、HOST タイプ指定で与えられた範囲、またはアプリケーションによって与えられた範囲に一致する必要があります。これは、従来より制限を軽減します。

オプションファイル内の最後の FQDN_MATCHING キーワードのみが有効です。他は、すべて無視されます。

表 5-8 に、オプションファイル内の HOST タイプ指定と、アプリケーションが送信したホスト名のマッチングの結果を示します。

表 5-8: ホスト名のマッチング結果

		FQDN 対応に設定されたアプリケーションの送信: myhost.abc.com	FQDN 対応に設定されていないアプリケーションの送信: myhost.abc.com
FQDN_MATCHING exact	HOST myhost	no	yes
	HOST myhost.abc.com	yes	no

表 5-8: ホスト名のマッチング結果

		FQDN 対応に設定されたアプリケーションの送信: myhost.abc.com	FQDN 対応に設定されていないアプリケーションの送信: myhost.abc.com
FQDN_MATCHING lenient	HOST myhost	yes	yes
● オプション ファイル	HOST myhost.abc.com	yes	yes

例

ホスト名のマッチングの制限について、次の例を見てください。

```
INCLUDE f1 HOST myhost.abc.com
```

```
FQDN_MATCHING exact
```

これは、機能 f1 を使用できるホストのリストに myhost.abc.com を含みます。さらに、アプリケーションが送信するホスト名は、myhost.abc.com と完全に一致する完全ドメイン名でなければなりません。

一方、次の例では、制限はよりゆるやかです。

```
INCLUDE f2 HOST myhost.abc.com
```

```
FQDN_MATCHING lenient
```

これは、機能 f2 を使用できるホストのリストに myhost.abc.com を含みます。送信されるホスト名は、myhost.abc.com または、単に myhost に一致します。しかし、myhost.xyz.com、yourhost、yourhost.abc.com には一致しません。

下の例は、さらにゆるやかです。

```
INCLUDE f2 HOST myhost
```

```
FQDN_MATCHING lenient
```

これは、機能 f3 に対するホストのリストにホスト名 myhost を含みます。lenient マッチングが指定されているので、ホスト名は myhost、myhost.abc.com、myhost.xyz.com などに一致します。一方、yourhost や yourhost.abc.com には一致しません。

関連事項:

[「タイプを使用してライセンスの制限を指定する」](#)

FQDN_MATCHING は、バージョン 9.3 クライアント ライブラリおよびベンダー デーモンで導入

GROUP

```
GROUP group_name user_list
```

INCLUDE、INCLUDEALL、INCLUDE_ENTITLEMENT、EXCLUDE、EXCLUDEALL、EXCLUDE_ENTITLEMENT、RESERVE オプション行の中で使用するユーザのグループを定義します。

表 5-9: GROUP の項目

項目	説明
group_name	定義されるグループの名前。グループ名は大文字と小文字が区別されます。
user_list	そのグループのユーザ名のリスト。名前は大文字と小文字が区別されます。大文字と小文字を区別したくない場合は、GROUPCASEINSENSITIVE オプションファイルキーワードを設定してください。詳細は、「GROUPCASEINSENSITIVE」を参照してください。

同じグループ名の複数の GROUP 行が、指定されたすべてのユーザをグループに追加します。

bob、howard、james からなるグループ Hackers を定義するには、次のように記述します。

```
GROUP Hackers bob howard james
```



注: USER_GROUP は GROUP の別名です。

GROUPCASEINSENSITIVE

```
GROUPCASEINSENSITIVE OFF|ON
```

ON に設定すると、オプションファイル GROUP と HOST_GROUP キーワードにより指定されたユーザ名とホスト名は、それぞれ大文字と小文字が区別されません。

GROUPCASEINSENSITIVE が OFF (既定) の場合、ユーザ名とホスト名は、大文字と小文字が区別されます。

HOST_GROUP

```
HOST_GROUP group_name host_list
```

INCLUDE、INCLUDEALL、INCLUDE_ENTITLEMENT、EXCLUDE、EXCLUDEALL、EXCLUDE_ENTITLEMENT、RESERVE オプション行の中で使用するホストのグループを定義します。複数の HOST_GROUP 行が、指定されたすべてのホストをグループに追加します。

表 5-10: HOST_GROUP の項目

項目	説明
group_name	定義されるグループの名前。ホストグループ名は大文字と小文字が区別されます。
host_list	そのグループのホスト名のリスト。名前は大文字と小文字が区別されます。大文字と小文字を区別したくない場合は、GROUPCASEINSENSITIVE オプション ファイル キーワードを設定してください。詳細は、「GROUPCASEINSENSITIVE」を参照してください。

tokyo, seattle, auckland からなるホストグループ Pacific を定義するには、次のように記述します。

```
HOST_GROUP Pacific tokyo seattle auckland
```

オプションファイルでホスト名が使用できる個所に、代わりに IP アドレスを使用できます。

INCLUDE

このオプションは、ライセンス ファイルおよび高信頼性ストレージに格納されている同時使用ライセンスに対して適用されます。

```
INCLUDE feature[:keyword=value] type {name | group_name}
```

ユーザまたはあらかじめ定義されているユーザグループを、この機能を使用することが許可されているユーザのリストに含めます。INCLUDE 文の中のないユーザは、その機能の使用を許可されません。EXCLUDE は INCLUDE を無効にします。EXCLUDE のリストと INCLUDE のリストで矛盾が生じた場合は、EXCLUDE を優先して解決します。

表 5-11: INCLUDE の項目

項目	説明
feature	影響を受ける機能またはパッケージの名前

表 5-11: INCLUDE の項目

項目	説明
keyword=value	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。
type	USER、HOST、DISPLAY、INTERNET、PROJECT、GROUP、HOST_GROUP のいずれか 1 つ。詳細は、「 タイプを使用してライセンスの制限を指定する 」を参照してください。
name	ライセンスの使用を許可するタイプ type の項目名
group_name	ライセンスの使用を許可するグループ名

機能 f1 を使用できるユーザのリストにユーザ bob を含めるには、次のように記述します。

```
INCLUDE f1 USER bob
```



注: USER_BASED または HOST_BASED 機能には INCLUDE が必要です。システム管理者は、INCLUDE を使って、どのユーザに製品の使用を許可するか指定します。INCLUDE 可能なユーザの数は、そのライセンスによって制限されます。

INCLUDE_BORROW

このオプションは、ライセンス ファイルに格納されているライセンスに使用されます。ライセンスが高信頼性ストレージ内で利用可能なときは、通常は BORROW ではなく アクティベーションが提供されます。

```
INCLUDE_BORROW feature[:keyword=value] type {name | group_name}
```

ユーザまたはあらかじめ定義されているユーザ グループを、BORROW 機能を借用することが許可されているユーザのリストに含めます。INCLUDE_BORROW 文の中になくユーザは、ライセンスの借用が許可されません。EXCLUDE_BORROW のリストと INCLUDE_BORROW のリストで矛盾が生じた場合は、EXCLUDE_BORROW を優先して解決します。

表 5-12: INCLUDE_BORROW の項目

項目	説明
feature	影響を受ける機能の名前
keyword=value	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。

表 5-12: INCLUDE_BORROW の項目

項目	説明
<code>type</code>	USER、HOST、DISPLAY、INTERNET、PROJECT、GROUP、HOST_GROUP のいずれか 1 つ。詳細は、「 タイプを使用してライセンスの制限を指定する 」を参照してください。
<code>name</code>	ライセンスの借用を許可するタイプ <code>type</code> の項目名
<code>group_name</code>	ライセンスの借用を許可するグループ名

機能 `f1` を借用できるユーザのリストにユーザ `tom` を含めるには、次のように記述します。

```
INCLUDE_BORROW f1 USER tom
```

INCLUDE_ENTITLEMENT

このオプションは、高信頼性ストレージ内に格納されているライセンスに対してのみ適用されます。

```
INCLUDE_ENTITLEMENT entitlementId type {name | group_name}
```

ユーザまたは事前定義されているユーザグループを、高信頼性ストレージ内のフルフィルメント レコードに格納されているライセンスのアクティベーションを許可されているユーザのリストに含めます。

EXCLUDE_ENTITLEMENT は INCLUDE_ENTITLEMENT を無効にします。

EXCLUDE_ENTITLEMENT のリストと INCLUDE_ENTITLEMENT のリストで矛盾が生じた場合は、EXCLUDE_ENTITLEMENT を優先して解決します。

表 5-13: INCLUDE_ENTITLEMENT の項目

項目	説明
<code>entitlementId</code>	ライセンス アクティベーションの要求時に最初に使用される資格 ID
<code>type</code>	USER、HOST、GROUP、HOST_GROUP のいずれか 1 つ。詳細は、「 タイプを使用してライセンスの制限を指定する 」を参照してください。
<code>name</code>	ライセンスの使用を許可するタイプ <code>type</code> の項目名
<code>group_name</code>	含めるグループの名前

資格 ID が `AB456` のフルフィルメント レコードに提供されているライセンスをアクティベーションできるユーザのリストに、ユーザ `claire` を含めるには、次のように記述します。

```
INCLUDE_ENTITLEMENT AB456 USER claire
```


INCLUDEALL

このオプションは、ライセンス ファイルおよび高信頼性ストレージに格納されている同時使用ライセンスに対して適用されます。

```
INCLUDEALL type {name | group_name}
```

このベンダー デーモンによって提供されるすべての機能を使用することが許可されているユーザのリストに、ユーザまたは事前定義されているユーザグループを含めます。INCLUDEALL 文の中にないうーザは、そのすべての機能の使用を許可されません。

表 5-14: INCLUDEALL の項目

項目	説明
type	USER、HOST、DISPLAY、INTERNET、PROJECT、GROUP、HOST_GROUP のいずれか 1 つ。詳細は、「 タイプを使用してライセンスの制限を指定する 」を参照してください。
name	ライセンスの使用を許可するタイプ type の項目名
group_name	含めるグループの名前

このベンダー デーモンによって提供されるすべての機能をユーザ jane が使用するのを許可するには、次のように記述します。

```
INCLUDEALL USER jane
```

LINGER

このオプションは、ライセンス ファイルおよび高信頼性ストレージに格納されている同時使用ライセンスに対して適用されます。

```
LINGER feature[:keyword=value] seconds
```

保持ライセンスは、チェックイン後または FLEX 対応アプリケーションの終了後(早く発生した方)、一定の期間チェックアウトされたままになります。このオプションによって、FLEX 対応アプリケーションのベンダーによって設定された既定の保持時間を延長できます。



注: この機能が動作するために、ベンダーは FLEX 対応アプリケーション内でこの機能を有効にしなければなりません。この機能が実装されているかどうかは、使用しているソフトウェアのベンダーにお問い合わせください。

表 5-15: LINGER の項目

項目	説明
feature	機能の名前
keyword=value	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。
seconds	ライセンスを保持する秒数。ベンダーが最小値を設定します。ベンダーの設定した最小値より短い <i>seconds</i> を指定した場合、最小値が使用されます。

機能 f1 に対する保持時間値を 1 時間 (3600 秒) に設定するには、次のように記述します。

```
LINGER f1 3600
```

ベンダー デーモンが毎分 1 回だけすべての保持状態のライセンスをチェックするため、実際の保持時間はある程度変化します。ただし、新しいライセンス要求が拒否されそうになった場合、保持ライセンスのチェックがただちに行われ、新しい要求に対応しようとしています。

MAX

このオプションは、ライセンス ファイルおよび高信頼性ストレージに格納されている同時使用ライセンスに対して適用されます。

```
MAX num_lic feature[:keyword=value] type {name | group_name}
```

グループまたはユーザに対して使用を制限します。

表 5-16: MAX の項目

項目	説明
num_lic	このユーザまたはグループに対する使用制限
feature	この制限が適用される機能またはパッケージ
keyword=value	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。

表 5-16: MAX の項目

項目	説明
type	USER、HOST、DISPLAY、INTERNET、PROJECT、GROUP、HOST_GROUP のいずれか 1 つ。詳細は、「 タイプを使用してライセンスの制限を指定する 」を参照してください。
name	使用を制限するタイプ <i>type</i> の項目名
group_name	制限するグループの名前

たとえば、ユーザ `jan` を機能 `f1` の 5 ライセンスまで制限する場合は、オプション ファイルに次の行を記述します。

```
MAX 5 f1 USER jan
```

MAX_BORROW_HOURS

このオプションは、ライセンス ファイルに格納されているライセンスに使用されます。ライセンスが高信頼性ストレージ内で利用可能なときは、通常は BORROW ではなく アクティベーションが提供されます。

```
MAX_BORROW_HOURS feature[:keyword=value] num_hours
```

ライセンスを借用できる最長期間を、*feature* のライセンス証明に指定された期間から変更します。新たに設定する期間はライセンス証明の期間を超えてはいけません。オプション ファイルに複数の MAX_BORROW_HOURS キーワードが現れる場合は、最後のキーワードが *feature* に適用されます。

表 5-17: MAX_BORROW_HOURS の項目

項目	説明
feature	この借用期間を適用する機能。 <i>feature</i> に対するライセンス証明には有効な BORROW の存在が必要です。
keyword=value	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。
num_hours	新しい借用期間の時間数。この値は機能に対するライセンス証明に指定された時間を超えてはいけません (指定しない場合の既定は 168 時間)。

MAX_OVERDRAFT

このオプションは、ライセンス ファイルおよび高信頼性ストレージに格納されている同時使用ライセンスに対して適用されます。

```
MAX_OVERDRAFT feature[:keyword=value] num_lic
```

ライセンスの超過使用 (OVERDRAFT) を、ライセンス ファイルによって許可されている超過数 (OVERDRAFT) より下に制限します。

表 5-18: MAX_OVERDRAFT の項目

項目	説明
feature	この制限が適用される機能
keyword=value	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。
num_lic	このユーザまたはグループに対する使用制限

NOLOG

```
NOLOG { IN | OUT | DENIED | QUEUED }
```

デバッグ ログ ファイル内の、選択されたイベント タイプの記録を停止します。



操作手順 チェックインの記録を停止するには

```
NOLOG IN
```

チェックアウトの記録とキューに入れられた要求の記録を停止するには、独立した 2 行の NOLOG 行が必要です。



操作手順 チェックアウトの記録とキューに入れられた要求の記録を停止するには

```
NOLOG DENIED  
NOLOG QUEUED
```



注: ライセンス管理者はこのオプションを使用して、デバッグ ログ ファイルのサイズを減らすことができます。ただしこれを行うと、ライセンス サーバシステムの問題をデバッグする上で、デバック ログの効用が低下します。

関連事項:

[「lmswitch」](#)

REPORTLOG

REPORTLOG `[+] report_log_path`

REPORTLOG は、このベンダー デーモンに対してレポート ログ ファイルを指定します。report_log_path の前に + 記号を付けて、ログ エントリを追加することをお勧めします。この記号を付けないと、デーモンが起動されるたびにファイルが上書きされます。

Windows では、スペースを含むパラメータを二重引用符で囲む必要があります。lmgrd をサービスとして開始した場合、完全有効パスが指定されていない限り、レポート ログ ファイルの既定の場所は、`c:\%winnt%\System32` フォルダです。



注: Macrovision から別売されている *ΦAEENET MANAGER* は、FLEXnet Licensing レポート ログ ファイルを処理するために使用されます。FLEXnet Manager が処理できるのはレポート ログ ファイルのみで、デバッグ ログ ファイルは処理できません。

LM_PROJECT を使用してプロジェクトについて レポートする

FLEXnet Manager レポート ライターは、プロジェクトについてレポートします。同じプロジェクト上で作業するすべてのユーザが、プロジェクトを説明する文字列に各自の環境変数 LM_PROJECT を設定することで (Windows ではレジストリを設定することで)、プロジェクトが設定されます。FLEXnet Manager は、アプリケーション実行時の LM_PROJECT の設定内容による定義に従って、使用状況をプロジェクトごとにグループ化します。

関連事項:

[「Windows サービスとしてライセンス サーバ マネージャを設定する」](#)

[「FLEXnet Licensing 環境変数」](#)

[「レポート ログ ファイル」](#)

RESERVE

このオプションは、ライセンス ファイルおよび高信頼性ストレージに格納されている同時使用ライセンスに対して適用されます。

```
RESERVE num_lic feature[:keyword=value] type {name | group_name}
```

特定のユーザにライセンスを予約します。

表 5-19: RESERVE の項目

項目	説明
num_lic	このユーザまたはグループに予約するライセンスの数
feature	この予約によって適用される機能またはパッケージ
keyword=value	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。
type	USER、HOST、DISPLAY、INTERNET、PROJECT、GROUP、HOST_GROUP のいずれか 1 つ。詳細は、「 タイプを使用してライセンスの制限を指定する 」を参照してください。
name	ライセンスの使用を予約するタイプ <code>type</code> の項目名
group_name	ライセンスの使用を予約するグループ名

ユーザ `mel` に機能 `f1` のライセンスを 1 つ予約するには、次のように記述します。

```
RESERVE 1 f1 USER mel
```

ユーザやグループに、それぞれライセンスを予約する場合、各ユーザやグループごとに RESERVE 行を記述する必要があります。パッケージ名が指定された場合、パッケージを構成するすべてのコンポーネントが予約されます。



注: ユーザに予約されたすべてのライセンスは、そのユーザ専用です。そのユーザがライセンスを実際に使用していないときであっても、そのライセンスを他のユーザが使用することはできません。ただし、RESERVE されたライセンスが実際に使用されていなければ、FLEXnet Manager はその使用をレポートしません。

TIMEOUT

このオプションは、ライセンス ファイルおよび高信頼性ストレージに格納されている同時使用ライセンスに対して適用されます。

```
TIMEOUT feature[:keyword=value] seconds
```

使用されていないライセンスが解放され、ベンダー デーモンによって再利用されるまでの時間を設定します。



注: この機能が動作するために、ベンダーは FLEX 対応アプリケーション内でこの機能を有効にしなければなりません。この機能が実装されているかどうかは、使用しているソフトウェアのベンダーにお問い合わせください。

表 5-20: TIMEOUT の項目

項目	説明
feature	機能の名前
keyword=value	ライセンスのグループを示す機能名の修飾語。詳細は、「 機能を指定する 」を参照してください。
seconds	使用されていないライセンスが返還要求されるまでの秒数。ベンダーは最小値を設定しています。ベンダーの設定した最小値より小さな <i>seconds</i> を指定した場合はこの最小値が使用されます。

機能 *f1* に対するタイムアウト値を 1 時間(3600 秒)に設定するには、次のように記述します。

```
TIMEOUT f1 3600
```

FLEX 対応アプリケーションの非アクティブな状態が指定された時間以上続いている場合、TIMEOUT はそのライセンスをチェックインします。アクティブな FLEX 対応アプリケーションはハートビートを送信するので、逆にデーモンはプロセスからハートビートの受信がない場合、そのプロセスが非アクティブであると宣言します。

ベンダーが提供するタイムアウト機能を有効にするには、TIMEOUT 行をオプションファイル内に記述する必要があります。

TIMEOUTALL

このオプションは、ライセンス ファイルおよび高信頼性ストレージに格納されている同時使用ライセンスに対して適用されます。

```
TIMEOUTALL seconds
```

すべての機能に適用される点を除き、TIMEOUT と同じです。

ベンダー デーモンがオプション ファイルを使用するプロセス

ベンダー デーモンが `lmgrd` によって起動されるとき、ベンダー デーモンはそのオプション ファイルを読み込みます。ベンダー デーモン 1 つにつきオプション ファイルは 1 つだけです。各ベンダー デーモンはそれぞれ固有のオプション ファイルを必要とします。オプション ファイルでの変更を有効にするには、ベンダー デーモンがそれぞれのオプション ファイルを必ず読み込む必要があります。`lmreread` ユーティリティは、ベンダー デーモンにそのオプション ファイルを再読み込みさせます。

- ベンダー デーモンにオプション ファイルを再読み込みさせるよう、バージョン 8.0 のベンダー デーモンで `lmreread` ユーティリティを拡張。旧バージョンをお使いの場合、オプション ファイルを再読み込みさせるためには、ベンダー デーモンを一旦停止し、再起動してください。

オプション ファイルにおける優先順位 の規則

`INCLUDE` 文と `EXCLUDE` 文が同じオプション ファイル内に出現し、同じ機能(ライセンス ファイル内)または同じフルフィルメント レコード(高信頼性ストレージ内)へのアクセスをコントロールする場合は、優先順位の規則が機能します。両タイプの文がともに出現した場合の優先順位は次のように決められています。

- `EXCLUDE` リストのみ存在する場合、リストに記載されていないすべてのユーザーに、その機能の使用が許可されます。
- `INCLUDE` リストのみ存在する場合、リストに記載されているユーザーのみに、その機能の使用が許可されます。
- どちらのリストも存在しない場合、すべてのユーザーにその機能の使用が許可されます。
- `EXCLUDE` リストは `INCLUDE` リストより先にチェックされ、両方のリストに記載されているユーザーは、その機能の使用が許可されません。

`INCLUDE` リストまたは `EXCLUDE` リストを作成すると、他のユーザーは暗黙的にグループ "外" となります。この機能を使用すると、管理者は、アクセスを許可または拒否したい各ユーザーを明示的にリストに記載しなくても、ライセンスをコントロールできます。つまり、次に示す 2 つの方法のいずれかを行うことができます。

- ほとんどのユーザーにアクセス権を与え、例外のユーザーのみをリストに記載する。または、
- アクセスを厳格に制限し、アクセス権を持つユーザーのみをリストに記載する。

オプションファイルの例

ライセンスへのアクセスを有効にコントロールする方法を説明するため、オプションファイルの例を次にいくつか示します。

単純なオプションファイルの例

```
RESERVE 1 compile USER robert  
RESERVE 3 compile HOST mainline  
EXCLUDE compile USER lori  
NOLOG QUEUED
```

このオプションファイルは、同時使用ライセンスの使用を次のように制限します。

- ユーザ robert に、機能 compile のライセンスを1つ予約します。
- ホスト名が mainline であるシステム上のすべてのユーザに、機能 compile のライセンスを3つ予約します。
- ネットワーク上のすべてのマシンでユーザ lori が、機能 compile を使用できないようにします。
- QUEUED(キューに入れられた)メッセージが、デバッグ ログ ファイルに記録されないようにします。

予約されるライセンスの合計数は、FEATURE 行で指定されたライセンス数以下でなければなりません。上にあげた例では、最低4つのライセンスが、機能 compile の FEATURE 行になければなりません。それより少ないライセンス数しか利用できない場合、ライセンス数の上限まで、最初からの予約のみが有効です。

このデータがファイル /a/b/sampled/licenses/sampled.opt にある場合、ライセンスファイルの VENDOR 行を次のように変更します。

```
VENDOR sampled /etc/sampled /sample_app/sampled/licenses/sampled.opt
```

複数のユーザに対してアクセスを制限する

INCLUDE、INCLUDEALL、INCLUDE_BORROW、INCLUDE_ENTITLEMENT、EXCLUDE、EXCLUDEALL、EXCLUDE_BORROW、EXCLUDE_ENTITLEMENT、MAX、RESERVE 行にはそれぞれ、1つのユーザ名(またはグループ)が記載されていなければなりません。複数のユーザ名に影響を与えるには、GROUP を作成します。たとえば、bob、howard、james が機能 toothbrush を使用できないようにするには、次のようなオプションファイルを作成します。

```
EXCLUDE toothbrush USER bob
EXCLUDE toothbrush USER howard
EXCLUDE toothbrush USER james
```

しかし、もっと簡単な方法があります。それは、GROUP を作成して、ユーザのリストを機能の使用から除外する方法です。前の例と同様に、次のオプション ファイルは、bob、howard、james が機能 toothbrush を使用できないようにします。

```
#最初にグループ "Hackers" を定義
GROUP Hackers bob howard james
#次にそのグループを除外
EXCLUDE toothbrush GROUP Hackers
```

これで、このグループに対して任意の機能へのアクセスを許可または拒否したい場合、それを簡単に行えるような別名のリストができました。

複数のホストを、許可、拒否、または予約するには、HOST_GROUP を使用します。たとえば、ホスト fred と barney にログインしているすべてのユーザが、機能 f1 を使用できないようにするには、次に示す行をオプション ファイルに追加します。

```
HOST_GROUP writers fred barney
EXCLUDE f1 HOST_GROUP writers
```

関連事項:

グループの定義についての詳細は、「[HOST_GROUP](#)」を参照してください。

EXCLUDE の例

```
#最初にグループ "painters" を定義
GROUP painters picasso mondrian klee
EXCLUDE spell GROUP painters
EXCLUDE spell USER bob
EXCLUDE spell INTERNET 123.123.123.*
```

このオプション ファイルは次のことを行います。

- ネットワーク上のすべてのマシンでユーザ picasso、mondrian、klee が、機能 spell を使用できないようにします。
- ネットワーク上のすべてのマシンでユーザ bob が、機能 spell を使用できないようにします。
- IP アドレスが 123.123.123.0 ~ 123.123.123.255 のホストにログインしているすべてのユーザが、機能 spell を使用できないようにします。
- 他のすべてのユーザが、除外された IP アドレス上に存在せず、なおかつ painters GROUP のメンバーでなく、なおかつ bob でもない限り、機能 spell を使用することを (暗黙的に) 許可します。

bob をグループ painters に追加することも可能だったことに注意してください。ただし、painters は将来他の目的に使用される可能性があるため、ここではライセンス管理者は、bob を特別な場合として扱うことにしています。この場合、2 つの EXCLUDE 文は連結して、4 人のユーザのリストを作成します。

EXCLUDE_ENTITLEMENT の例

```
#最初にグループ "admin" を定義
GROUP admin johns adrianp maryt
EXCLUDE_ENTITLEMENT qf573k GROUP admin
EXCLUDE_ENTITLEMENT qf573k USER bob
EXCLUDE_ENTITLEMENT qf573k HOST cordelia
```

このオプションファイルは次のことを行います。

- ネットワーク上のすべてのマシンでユーザ johns, adrianp, maryt が、資格 ID qf573k を使用して取得したフルフィルメントレコードに格納されているすべてのライセンスをアクティベーションできないようにします。
- ネットワーク上のすべてのマシンでユーザ bob が、資格 ID qf573k を使用して取得したフルフィルメントレコードに格納されているすべてのライセンスをアクティベーションできないようにします。
- マシン cordelia 上のすべてのユーザが、資格 ID qf573k を使用して取得したフルフィルメントレコードに格納されているすべてのライセンスをアクティベーションできないようにします。
- cordelia 以外のすべてのマシン上の他のすべてのユーザが、資格 ID qf573k を使用して取得したフルフィルメントレコードに格納されているライセンスを暗黙的にアクティベーションできるようにします。

INCLUDE の例

```
INCLUDE paint USER picasso
INCLUDE paint USER mondrian
INCLUDE paint HOST bigbrush
```

このオプションファイルは次のことを行います。

- ネットワーク上のすべてのマシンでユーザ picasso が、機能 paint を使用できるようにします。
- ネットワーク上のすべてのマシンでユーザ mondrian が、機能 paint を使用できるようにします。
- ホスト bigbrush 上のすべてのユーザが、機能 paint を使用できるようにします。

- picasso、mondrian、またはホスト bigbrush 上のユーザ以外のすべてのユーザに対して、機能 paint へのアクセスを(暗黙的に)拒否します。

INCLUDE_ENTITLEMENT の例

```
INCLUDE_ENTITLEMENT gy7210 USER tom  
INCLUDE paint USER anthony  
INCLUDE paint HOST jupiter
```

このオプション ファイルは次のことを行います。

- ネットワーク上のすべてのマシンでユーザ tom が、資格 ID gy7210 を使用して取得したフルフィルメント レコードに格納されているすべてのライセンスをアクティベーションできるようにします。
- ネットワーク上のすべてのマシンでユーザ anthony が、資格 ID gy7210 を使用して取得したフルフィルメント レコードに格納されているすべてのライセンスをアクティベーションできるようにします。
- ホスト jupiter 上のすべてのユーザが、資格 ID gy7210 を使用して取得したフルフィルメント レコードに格納されているすべてのライセンスをアクティベーションできるようにします。
- tom、anthony、またはホスト jupiter 上のユーザ以外のすべてのユーザに対して、資格 ID gy7210 を使用して取得したフルフィルメント レコードに格納されているすべてのライセンスのアクティベーションを暗黙的に拒否します。

6

モバイル ライセンス管理

エンド ユーザが、FLEXnet ライセンス サーバシステムと常時接続しないコンピュータ上でアプリケーションを使用したい場合があります。たとえば次のような状況です。

- ラップトップ コンピュータで作業する。
- 職場と自宅の両方でコンピュータを使用する。
- ライセンス サーバシステムに接続されていない複数のコンピュータから作業する。

FLEXnet Licensing は、次にあげるモバイル ライセンス管理の 1 つを許可するライセンスをサポートします。

- ラップトップにノードロック
- FLEXid にノードロック (Windows のみ)
- FLOAT_OK 付きの FLEXid にノードロック (Windows のみ)
- BORROW キーワードによるライセンス借用
- ユーザ名にノードロック
- プリペイド ライセンス プールからのライセンス発行

上記の方法以外でライセンスを移動するには、リホスティングの必要があります。このことは、新たなクライアント コンピュータそれぞれに対して

ラップトップ コンピュータにノードロック

ライセンスが排他的に 1 台のラップトップ コンピュータ上で使用される場合、そのライセンスはそのコンピュータに付いているアドレスに単純にノードロックされます。このライセンス ファイルはラップトップ コンピュータに常駐します。

FLEXid にノードロック (Windows のみ)

異なる Windows マシン間でライセンスを移動しなければならない場合、そのライセンスは FLEXid (パラレル ポートまたは USB ポートに接続する dongle) にノードロックされます。このライセンスをマシン間で移動するには、各マシンにライセンス ファイルのコピーをインストールし、FLEXid をマシン間で移し替えます。ライセンスは FLEXid に連結されるため、FLEXid を装着したマシンのみがライセンスを使用できます。

FLOAT_OK 付きの FLEXid にノードロック (Windows のみ)

このライセンス移動方式には、FLEXid がライセンス サーバマシンに装着されると、そのライセンスがネットワーク上でフロートするため、単純に FLEXid にノードロックするライセンスよりメリットがあります。FLEXid にノードロックされた FLOAT_OK 付きのライセンスは、FLEX 対応アプリケーションとライセンス サーバシステムが Windows 上で起動している場合にのみサポートされます。

ベンダーは、FLEXid にノードロックされ、FLOAT_OK キーワード付きの FEATURE 行を含むライセンス ファイル、およびその FLEXid を発行します。FLOAT_OK キーワードを含む FEATURE 行と FLEXid が、モバイルライセンスの各インスタンスごとにそれぞれ 1 つ必要です。FLEXid がライセンス サーバマシンに装着されると、ライセンスはネットワーク上でフロートします。FLEXid がライセンス サーバマシンから取り外されると、ライセンスはスタンドアロン コンピュータ上でしか使用できません。

この方法は、パラレルまたは USB FLEXid をサポートしています。コンピュータに複数の USB dongle を装着することは簡単なので、USB FLEXid の方が好ましいかもしれません。

FLOAT_OK 付きの FLEXid を開始する

ベンダーは、FLEXid、FLEXid ドライバのインストーラ、および FLOAT_OK キーワードを含み、FLEXid にノードロックされた FEATURE 行を持つライセンス ファイルを発行します。エンド ユーザは、それを次の手順で処理します。

1. ライセンス サーバマシン上にライセンス ファイルをインストールします。
2. すべての FLEXid をライセンス サーバマシンに装着します。
3. ライセンス サーバマシン上に FLEXid ドライバをインストールします。
4. ライセンス サーバ システムを再起動するか、ライセンス ファイルを再読み込みさせます。

FLEXid がライセンス サーバマシンに装着されると、それに関連付けられたライセンスはネットワーク上でフロートします。FLOAT_OK で、カウントされず、かつノードロックされた状態の各 FEATURE 行は、ネットワーク上で利用できる間は、1 とカウントされます。エンド ユーザは、次の手順に従います。



操作手順

フローティング ライセンス プールから、接続されていないコンピュータにライセンスを転送するには

1. FLOAT_OK のノードロックされた FEATURE 行を含むライセンス ファイルを、ライセンス サーバマシン上のライセンス ファイルから、クライアント コンピュータ上のライセンス ファイルにコピーします。ただしこのコピー先は、FLEX 対応アプリケーションがそのライセンス ファイルを検索できる場所に存在しなければなりません。
2. ノードロックされた FEATURE 行 と一致する FLEXid を、ライセンス サーバマシンからクライアント コンピュータに移し替えます。FLEXid をライセンス サーバマシンから除去すると、ライセンスをネットワーク上で使用できなくなります。
3. まだ FLEXid ドライバをインストールしていない場合は、クライアント コンピュータにインストールします。
4. クライアント コンピュータをネットワークから切り離します。これで、コンピュータがネットワークから切り離されても、FLEXid を装着したコンピュータ上でこのライセンスを使用できます。

FLOAT_OK 付きの FLEXid を返却する



操作手順 ライセンスをライセンス サーバ マシンに戻し、ライセンスが再びネットワーク上でフロートするようには、エンド ユーザは次のように行います。

1. クライアント マシンから FLEXid を取り外し、それをライセンス サーバ マシンに取り付けます。
2. `lmreread` を実行し、フローティング バージョンのライセンスを提供するライセンス サーバ システムのライセンス ファイルを再読み込みします。FLEXid をライセンス サーバ マシンに戻しても、再び `lmreread` が実行されるまで、FLOAT_OK ライセンスはネットワーク上でフロートしません。

FLOAT_OK 付きの FLEXid の例

次の例は、エンドユーザのサイトに発行されたライセンス ファイルのサンプルです。2つの FLEXid (FLEXID=7-b28520b9 と FLEXID=7-b2857678) が添付されたとします。

```
SERVER myhost ANY
VENDOR sampled
FEATURE f1 sampled 1.0 permanent uncounted FLOAT_OK ¥
      HOSTID=FLEXID=7-b28520b9 SIGN=123456789012
FEATURE f1 sampled 1.0 permanent uncounted FLOAT_OK ¥
      HOSTID=FLEXID=7-b2857678 SIGN=ABCDEF123456
```

エンド ユーザは、ライセンス ファイルと 2つの FLEXid をライセンス サーバ マシンにインストールします。ライセンス サーバ マシンに、この 2つの ID が装着されると、それぞれのカウントされない FLOAT_OK ライセンスはネットワーク上でフロートし、1つの使用ができるようになります。したがって、最大 2人のエンド ユーザが、自分のネットワーク上で f1 を使用できます。ただし、ライセンス サーバ マシンではライセンスを使用できません。

エンド ユーザが自宅で作業したい場合、FLEXID=7-b28520b9 にノードロックされた FEATURE 行を含むライセンス ファイルをインストールします (1回だけ実行します)。次に、FLEXid FLEXID=7-b28520b9 をライセンス サーバ マシンからクライアント コンピュータに移し替え、FLEXid ドライバをクライアント コンピュータにインストールします (これも 1回だけ実行します)。エンド ユーザは、ネットワークからクライアント コンピュータを切り離し、クライアント コンピュータに転送した FLOAT_OK ライセンスを使用します。ライセンス サーバ システムは、残りの FLOAT_OK ライセンス 1つだけをネットワーク上でフロートさせることができます。

FLEXid がライセンス サーバ マシンに戻された後に、エンド ユーザ (またはシステム管理者) は、`lmreread` を実行して返却されたライセンスを再びフロートさせます。

- キーワード `FLOAT_OK` は、v8.0 `FLEXlm` のクライアント ライブラリ、ライセンス サーバ マネージャ、ベンダー デーモンで導入。`FLOAT_OK` を使用するには、すべてのコンポーネントが v8.0 以上でなければなりません。

BORROW によるライセンス借用

ライセンス サーバ システムに不定期に接続されるコンピュータ上でライセンスを使用しなければならない場合、そのライセンスを `BORROW` キーワードを含むフローティング ライセンスとして発行できます。特殊なチェックアウト方法で `BORROW` ライセンスをライセンス サーバ システムから借用し、そのライセンスを使用して、ライセンス サーバ システムに接続していないコンピュータ上でアプリケーションを実行することができます。これには、エンド ユーザがライセンスを借用する前に、ベンダーがライセンス借用を有効に設定しておく必要があります。

ライセンス借用の際に、ベンダーは、`BORROW` キーワードを含む `FEATURE` 行を持つフローティング ライセンスを発行します。エンド ユーザは、ネットワークに接続している間に、借用したいライセンスを返却する有効期限を指定し、アプリケーションを実行します。借用情報は、クライアント コンピュータに書き込まれます。ライセンス サーバ システムは、借用ライセンスのチェックアウト状態を保持します。`FLEX` 対応アプリケーションは、ローカルの借用データを自動的に使用して、借用期間中にチェックアウトします。ベンダーによって有効に設定されている場合、借用ライセンスを早期に、つまり借用期間が終了する前に返却できます。借用期間の終了により、または早期返却として借用ライセンスを返却すると、ローカルの借用データによるチェックアウトは承認なくなり、ライセンス サーバ システムによって借用ライセンスが使用可能なライセンス プールに返却されます。ライセンス サーバ マシンと、`FLEX` 対応アプリケーションを実行中のマシンとの間で、クロックを同期させる必要はありません。

ライセンスの借用を開始する

ベンダーが、`BORROW` キーワードを含む `FEATURE` 行を持つライセンス ファイルを発行することでライセンスの借用を有効にした場合、エンド ユーザは次の 3 つのいずれかの方法でライセンスの借用を開始します。

- アプリケーション内の借用インタフェースを使用する (アプリケーションに提供されている場合)。
- `lmborrow` ユーティリティを実行して `LM_BORROW` を設定する。
- 環境変数 `LM_BORROW` を直接設定する。

アプリケーション インタフェース

アプリケーションが借用インタフェースを提供する場合にのみ、ユーザはこの方法でライセンスの借用を開始します。この方法に関する詳細は、ベンダーが提供します。

lmborrow ユーティリティを起動する

lmborrow は、lmutil/LMTOOLS および LMTOOLS ユーティリティの 1 つです。借用を開始するには、コマンド ラインまたは LMTOOLS から lmborrow を実行します。

```
lmborrow {vendor | all} enddate [time]
```

ここで、*vendor* は、借用ライセンスを交付するベンダー デーモンを指定します。または **all** により、ライセンス サーバシステム内のすべてのベンダー デーモンを指定します。*enddate* はライセンスの返却日を指定し、*dd-mmm-yyyy* のフォーマットで記述します。*time* はオプションです。FLEX 対応アプリケーションのローカル時間の 24 時間フォーマット (*hh:mm*) で指定します。*time* を指定しないと、所定の終了日が満了するまでチェックアウトが持続します。

例を次に示します。

```
lmborrow sampled 20-aug-2001 13:00
```

環境変数 LM_BORROW を直接設定する。

lmborrow ユーティリティはユーザ インタフェースであり、LM_BORROW をレジストリ (Windows) または \$HOME/.flexlmborrow (UNIX) に設定します。LM_BORROW を、次のように環境変数として直接設定することもできます。

```
today:{vendor|all}:enddate[:time]
```

以下に詳細を示します。

表 6-1: LM_BORROW の環境変数引数

引数	説明
<i>today</i>	<i>dd-mmm-yyyy</i> フォーマットによる今日の日付。この日付でチェックアウトが行われると、ローカルな借用情報が作成されます。この日付と異なる日付でチェックアウトしても、ローカルな借用情報は作成されません。
<i>vendor</i>	借用ライセンスを交付するベンダー デーモン名、または all でそのライセンス サーバシステム内のすべてのベンダー デーモンを指定します。

表 6-1: LM_BORROW の環境変数引数

引数	説明
<i>enddate</i>	ライセンスを返却する <i>dd-mm-yyyy</i> フォーマットでの日付
<i>time</i>	オプション。 <i>time</i> は、FLEX 対応アプリケーションのローカル時間で 24 時間フォーマット (<i>hh:mm</i>) で指定されます。 <i>time</i> を指定しないと、所定の終了日が満了するまでチェックアウトが持続します。

例を次に示します。

```
LM_BORROW=15-aug-2006:sampled:20-aug-2006:13:00
```

この例では、**sampled** ベンダー デーモンによって交付された 1 つまたは複数のライセンスが、2006 年 8 月 15 日に借用され、2006 年 8 月 20 日の午後 1 時に返却されることになっています。

ライセンスを借用する

必要な機能のライセンスを借用するには、エンド ユーザは **lmborrow** の実行または **LM_BORROW** の設定を行った**同じマシン上でかつ同じ日付に**、アプリケーションを実行し、ライセンスをチェックアウトして借用します(この間ネットワークとは接続)。同じ日に 2 回以上、そのアプリケーションを実行した場合、重複するライセンスは借用されません。開始日に設定された借用日と異なる日にアプリケーションを実行しても、ライセンスは借用されません。

たとえば、**PageWizard** 機能のライセンスを今日から 1 週間借用する必要があるとします。**PageWizard** 機能は、ベンダー デーモン **sampled** が交付します。今日、ネットワークに接続中に **lmborrow** を実行するか、**LM_BORROW** を直接設定します。例を次に示します。

```
lmborrow sampled enddate
```

今日、**lmborrow** を実行した後に、ネットワークと接続中に **PageWizard** 機能のライセンスをチェックアウトするアプリケーションを実行します。そのライセンスをチェックアウトした後に、アプリケーションを終了し、マシンをネットワークから切り離します。いまチェックアウトしたライセンスは、借用期限がくるまでライセンス サーバシステムからチェックアウトされた状態が続きます。つまり、そのライセンスは、サーバと接続していないマシン上で借用期限がくるまで使用されます。一度チェックアウトされると、そのライセンスは借用期間が満了するまでチェックアウトが続きます。その期間が満了する前に、借用期間を更新することはできません。

借用期間をクリアする

環境変数 `LM_BORROW` で定義した現在の借用期間に必要なすべてのライセンスを借用した後は、`lmborrow -clear` を実行し、その他の機能のライセンスの借用を防止します。`lmborrow -clear` は、レジストリ内の `LM_BORROW` 設定 (Windows の場合)、または `$HOME/.flexlmborrow` (UNIX の場合) をクリアしますが、既に借用されているライセンスに関するローカルの情報はクリアしません。

借用のステータスをチェックする



操作手順 借用した機能に関する情報を出力するには:

1. 機能を借用したマシンから次のコマンドを発行します。

```
lmborrow -status
```

ステータスを確認するために借用システムをネットワークに接続する必要はありません。

借用したライセンスを期限前に返却する



操作手順 借用したライセンスを借用期限前に返却するには:

1. 借用システムをネットワークに再接続します。
2. 次に借用を開始したマシンと同じマシンから次のコマンドを発行します。

```
lmborrow -return [-c license_file_list] feature
```

このオプションは、ベンダーによっては許可されない場合があります。期限前のライセンス返却がサポートされているかどうかは、ベンダーに直接問い合わせてください。

期限前のライセンス返却は、返却ライセンスを交付するベンダー デーモンに対して `LM_BORROW` 設定をクリアする効果があります。

ライセンス借用に対するサポート

ユーティリティと、ライセンス借用をサポートするエンド ユーザのオプションに関する詳細は、次のセクションを参照してください。

- 「lmborrow」
- 「lmdown」
- 「lmstat」
- 「BORROW_LOW WATER」
- 「EXCLUDE_BORROW」
- 「INCLUDE_BORROW」

- キーワード BORROW は、バージョン 8.0 クライアント ライブラリ、ライセンス サーバ マネージャ、ベンダー デーモンで導入。BORROW を使用するには、すべてのコンポーネントがバージョン 8.0 以上でなければなりません。

ユーザ名にノードロック

ライセンスをさまざまなマシン上で 1 ユーザが排他的に使用する場合、そのユーザのユーザ名にそのライセンスをノードロックすることができます。ライセンス ファイルは、そのユーザが作業する個々のマシンにコピーされます。ユーザのユーザ名は各マシンで一意でなければなりません。この方法を活用するには、組織内の個々のユーザ名が一意でなければなりません。

プリペイド ライセンス プールからの ライセンス発行

この方法では、エンド ユーザは、ライセンスの使用日数に応じて支払を済ませたライセンスを、ベンダーから購入します。したがってエンド ユーザは、必要な借用期間に応じてライセンスの全日数の一部を利用し、特定のマシンにノードロックされたライセンスの発行が実行できます。たとえば、出張の準備中に(出張中であっても)、エンド ユーザは、持って行くラップトップにノードロックされた 5 日間使用できるライセンスの発行を実行します。各ライセンス発行の実行は異なるマシン(または同一マシンに複数回)ノードロックすることができます。このようにして、支払済みのライセンスの日数分だけライセンスを使用するという、ライセンスの可動性を実現します。

このモデルは、各ライセンスを実行するとライセンス日数が減少するという点で、ペイパーユースと似ています。しかし、一度マシンにノードロックすると、そのマシンはライセンスの有効期限内であればアプリケーションを無制限に使用できるという点で、ペイパーユースとは異なります。この短期間のライセンスは、期限前に返却することはできません。一度ライセンスの発行が実行されたら、該当ライセンス日数を未実行(未使用)状態に戻すことはできません。一方、他のペイパーユースモデルは、アプリケーションの使用回数により課金されます。

FLEXnet Licensing がサポートするプラットフォームのホスト ID

FLEXnet Licensing は、マシンのアーキテクチャ別に、異なるマシン ID を使用します。たとえば、Sun Microsystem 社のすべてのマシンには一意のホスト ID が付与されていますが、DEC 社のマシンには一切付与されていません。このような理由から、一部のマシンアーキテクチャには、イーサネットアドレスをホスト ID として使用します。イーサネットアドレスは 6 バイト数で、各バイトは 2 桁の 16 進数で指定されます。イーサネットアドレスをホスト ID として使用するときは、12 桁の 16 進数をすべて指定します。たとえば、イーサネットアドレスが "8:0:20:0:5:ac," であれば、ホスト ID として "0800200005ac" と指定します。

ホスト ID のフォーマット

32 ビットのマシン ID の番号は、通常 16 進数のフォーマットを使います。あるシステムでは、システム コマンドは 10 進数のフォーマットで ID を返します。10 進数を示すには、ホスト ID の前に # を付けます。たとえば、システム コマンドが 2005771344 を返す場合、FLEXnet Licensing では、#2005771344 と表記すれば OK です。または、10 進数の値を 16 進数に変換します。

FLEXnet Licensing のホスト ID

lmhostid ユーティリティは、任意のマシン上で FLEXnet Licensing が使用されると思われる正確なホスト ID を出力します。ホスト ID に、ASCII の A ~ Z, a ~ z, または 0 ~ 9 以外の文字が含まれる場合に、lmhostid の `-utf8` オプションを使用します。このホスト ID が正しいことを確認するには、UTF-8 エンコード文字列を表示できる「メモ帳」などのユーティリティを使用します。

次の表に、各マシンアーキテクチャで必要なホスト ID を取得する代替方法の一覧を示します。FLEXnet Licensing は、いくつかの特殊なホスト ID やベンダー定義のホスト ID もサポートします。

表 7-1: ホスト ID を取得する代替方法

ハードウェアプラットフォーム	ホスト ID	ライセンス サーバ システムで 入力するコマンド	例
AIX (RS/6000, PPC)	32 ビット ホスト ID	<code>uname -m</code> (000276513100 を返します)、次に下位の 2 桁を削除して、残りの下位 8 桁を使用します。	02765131
DEC Alpha	イーサネット アドレス	<code>netstat -i</code>	080020005532
HP (32 ビットおよび 64 ビットプラット フォーム、非 Itanium)	32 ビット ホスト ID	<code>uname -i</code> で返される値を 16 進数に変換、または前に # を付けます。	778DA450 または #2005771344
HP (64 ビット Itanium)	マシン ID	<code>getconf ¥ CS_PARTITION_ IDENT</code> で返される値に接頭語 "ID_STRING=" を付けます。	ID_STRING=9c766319- db72-d411-af62- 0060b05e4c05
Mac OS X	イーサネット アドレス	<code>/sbin/ifconfig eth0</code> で返さ れる <code>ether</code> の値からコロンを削除 します。	000A277EA17E
	FLEXid USB ポート ドングル	<code>lmhostid -flexid</code>	FLEXID=9-b28520b9

表 7-1: ホスト ID を取得する代替方法

ハードウェアプラットフォーム	ホスト ID	ライセンス サーバ システムで 入力するコマンド	例
Linux	イーサネット アドレス	/sbin/ifconfig eth0 で返される HWaddr からコロンを削除します。	00400516E525
	FLEXid USB ポート ドングル	lmhostid -flexid	FLEXID=9-b28520b9
SCO	ホスト ID 文字列	uname -x で返される値(シリア ▲ル番号は SCO00354)に接頭語 "ID_STRING=" を付けます。	ID_STRING=SCO00354
SGI	32 ビット ホスト ID	/etc/sysinfo -s、16 進数に変 換、または接頭語 # を付けます。	69064C3C または #1762020412
SUN	32 ビット ホスト ID	ホスト ID	170a3472
	イーサネット アドレス	lmhostid -ether	00400516E525
Windows	イーサネット アドレス	lmhostid	00B0A9DF9A32
	ハード ディス ク ドライブの シリアル番号	DIR C: (ボリューム シリアル番号 を探して、- を削除します)。	DISK_SERIAL_NUM= 3e2e17fd
	FLEXid パラ レルまたは USB ポート ドングル	lmhostid -flexid FLEXid はベンダーによって使用が認められます。ベンダーは、 すべての FLEXid 用のドライバをインストールするインストー ラと一緒に提供することもできます。 パラレル FLEXid の場合、パラレル ポートは双方向モードに設 定されていなければなりません。	FLEXID=8-b28520b9

特殊な FLEXnet Licensing ホスト ID

FLEXnet Licensing には、すべてのプラットフォームに適用される「特殊な」ホスト ID のタイプがいくつかあります。このタイプのホスト ID は、ホスト ID を必要とする SERVER 行または FEATURE 行のどちらに使用しても有効です。次にそれを示します。

表 7-2: 特殊なホスト ID のタイプ

ホスト ID	説明
ANY	ソフトウェアをあらゆるマシンにロックします(つまり、なにもロックしません)。
DEMO	ANY と似ていますが、カウントされない FEATURE 行だけに使用します。
COMPOSITE= <i>composit_hostid</i>	ソフトウェアを合成ホスト ID にロックします。合成ホスト ID は、ベンダーによって指定された1つまたは複数の単純なホスト ID タイプの値を合成した、12 文字の 16 進数ハッシュ値です。
DISPLAY= <i>display</i>	ソフトウェアをディスプレイ <i>display</i> にロックします。UNIX では、 <i>display</i> は /dev/ttyxx (アプリケーションがバックグラウンドで実行中は常に /dev/tty)、または X-Display の名前です。Windows では、これはシステム名、またはターミナル サーバのクライアント名(ターミナル サーバ環境の場合)です。(バージョン 8 以上の FLEX 対応アプリケーションのみ)
HOSTNAME= <i>host</i>	ソフトウェアをコンピュータのホスト名 <i>host</i> にロックします。
ID= <i>n</i>	機能的に "ANY" ホスト ID と同等。つまり、あらゆるマシンで動作します。異なる点はライセンスが一意であること、およびエンド ユーザの識別に使用できることです。このホスト ID は、ライセンス サーバ システム (SERVER 行に使用)、または FLEX 対応アプリケーション (FEATURE と INCREMENT 行に使用) をロックするために使用されます。読みやすくするために、番号にはダッシュを使用できます(ただし、ダッシュは無視されます)。 例: <ul style="list-style-type: none">● ID=12345678 は下記と同じです。● ID=1234-5678 は下記と同じです。● ID=1-2-3-4-5-6-7-8
INTERNET= <i>###.###.###.###</i>	ソフトウェアをインターネット IP アドレス、または IP アドレス グループにロックします。ワイルドカードが使用できます。たとえば、198.156.*.* は、対応するインターネット IP アドレスを持つ任意のホストを意味します。主に、地理的な領域を示すサブネットに基づいてアクセスを制限します。このため、ワイルドカードが、FEATURE 行と INCREMENT 行でホスト ID のロックとして使用されます。

表 7-2: 特殊なホスト ID のタイプ

ホスト ID	説明
USER= <i>user</i>	ソフトウェアをユーザ名 <i>user</i> にロックします。

例

```
FEATURE f1 demo 1.0 1-jan-2005 uncounted ¥  
HOSTID=FLEXID=6-a6300015f SIGN=AB28E0011DA1
```

または次のように記述します。

```
FEATURE f1 demo 1.0 1-jan-2005 uncounted ¥  
HOSTID=INTERNET=10.10.10.* SIGN=EB78201163B0
```


ライセンス ファイルの フォーマット

通常、ライセンス ファイルは先頭に SERVER 行が 1 行 (3 サーバによる冗長構成のサーバの場合は 3 行) 置かれ、その後に 1 行または複数行の VENDOR 行、さらに 1 行または複数行の FEATURE 行または INCREMENT 行が続きます。場合によっては、ライセンス ファイルに SERVER 行も VENDOR 行も必要ないことがあります。

ライセンス ファイルで修正できる要素を次に示します。

- SERVER 行のホスト名
- SERVER 行の TCP/IP ポート番号
- SERVER 行の 3 サーバによる冗長構成のサーバ設定
- VENDOR 行のパス
- VENDOR 行のオプション ファイルのパス
- VENDOR 行のオプション TCP/IP ポート番号 (ファイヤウォールのサポートのみ)
- USE_SERVER 行
- FEATURE 行の *keyword=value* のペアの値 (*keyword* が小文字で指定されている場合)

長い行を改行するには、継続行を示す文字 \backslash を使用します。

8 ビットのラテンベース文字は、ライセンス ファイル、オプション ファイル、ログ ファイル、FLEX 対応アプリケーション環境において、完全にサポートされています。

SERVER 行と VENDOR 行の要件についての詳細は、「[カウントされるライセンスとカウントされないライセンス](#)」を参照してください。

- USE_SERVER はバージョン 5.0 のクライアント ライブラリで導入
- 行継続文字は、バージョン 7.0 以前のクライアント ライブラリでは必須
- 8 ビットのラテンベース文字へのサポートは、バージョン 8.0 のクライアント ライブラリで導入

ライセンス ファイルの構文

このセクションでは、SERVER 行や VENDOR 行などの基本的なライセンス ファイルフォーマットについて説明します。

サンプル ライセンス ファイル

次に示すのは、2 つの機能を持つ 1 つのベンダーのライセンス ファイルの例です。

```
SERVER my_server 17007ea8 1700
VENDOR sampled
FEATURE f1 sampled 1.000 01-jan-2005 10 SIGN=9BFAC0316462
FEATURE f2 sampled 1.000 01-jan-2005 10 SIGN=1B9A308CC0F7
```

上記のライセンス ファイルにより、ホスト ID が 17007ea8 のライセンス サーバ システム my_server が、ネットワーク上のユーザに機能 f1 および f2 のフローティング ライセンスを 10 交付します。

SERVER 行

SERVER 行は、ライセンス サーバ システムのホスト名とホスト ID、およびライセンス サーバ マネージャ (lmgrd) の TCP/IP ポート番号を指定します。通常、ライセンス ファイルには SERVER 行が 1 行あります。SERVER 行が 3 行ある場合は、3 サーバによる冗長構成のライセンス サーバ システムを使用していることを意味します。SERVER 行がない場合は、ライセンス ファイルのすべての FEATURE 行または INCREMENT 行がカウントされないことを意味します。

SERVER 行のホスト ID は、すべての FEATURE 行および INCREMENT 行でライセンス キーまたは署名の計算に使用されています。このため、ベンダーから送信されてきた SERVER 行は、FEATURE/INCREMENT 行とともに必ず保存してください。

SERVER 行のフォーマットを次に示します。

```
SERVER host hostid [port] [PRIMARY_IS_MASTER]
[HEARTBEAT_INTERVAL=seconds]
```

以下に詳細を示します。

表 8-1: SERVER 行のフォーマット

フィールド	説明
<i>host</i>	システムのホスト名または IP アドレス。UNIX で <code>hostname</code> コマンドまたは <code>uname -n</code> コマンドを実行すると取得できる文字列です。Windows NT/2000/XP では <code>ipconfig /all</code> 、Windows 95/98/ME では <code>winipcfg /all</code> を実行するとホスト名を取得できます。
<i>hostid</i>	通常は <code>lmhostid</code> コマンドを実行して取得できる文字列です。この値を変更できるのは、ソフトウェアの提供者のみです。
<i>port</i>	<p>使用する TCP/IP ポート番号。有効な番号は、0 ~ 64000 までの未使用の番号です。UNIX では、1024 より大きいポートを選択します。1024 未満のポートは特権ポート番号です。TCP/IP ポート番号を指定しないと、既定の 27000 ~ 27009 の中からポート番号が 1 つ使用されます。</p> <p>3 サーバによる冗長構成のライセンス サーバ システムの設定で、サーバを指定する SERVER 行には、ポート番号の指定が必要です。Macrovision は、27000 ~ 27009 の範囲以外のポート番号を使用することをお勧めします。</p>
PRIMARY_IS_MASTER	<p>3 サーバによる冗長構成では、プライマリ サーバと 2 台のセカンダリ サーバとの間でマスター コントロールの移動の方法を指定します。</p> <p>PRIMARY_IS_MASTER がセットされていてプライマリ サーバが使用不能になったときは、セカンダリ サーバがマスターになります。プライマリ サーバが復旧したら速やかにマスター コントロールを返却します。</p> <p>PRIMARY_IS_MASTER がセットされていなくてプライマリ サーバが使用不能になったときは、セカンダリ サーバがマスターになります。プライマリ サーバが復旧してもセカンダリ サーバがマスター コントロールを保持し続けます。</p> <p>プライマリ サーバとセカンダリ サーバの両方が使用不能になったときは、ライセンスは提供できません。その場合には、第 3 のサーバがマスターになります。</p> <p>このパラメータは省略可能です。ライセンス ファイルの最初の SERVER 行に記します。このパラメータを使用するには、バージョン 10.8 以降のベンダー デーモンを実行する必要があります。</p>

表 8-1: SERVER 行のフォーマット

フィールド	説明
HEARTBEAT_INTERVAL= <i>seconds</i>	<p>3 サーバによる冗長構成では、サーバをシャットダウンする前に、他のサーバのハートビート信号を受信するために何秒間待機するかを設定します。<i>seconds</i> はタイムアウト時間を計算するために次の式で使用されます。</p> <ul style="list-style-type: none">• $timeout = (3 \times seconds) + (seconds - 1)$ <p>指定しない場合、既定値の <i>seconds</i> は 20 です。上の式によりタイムアウト時間は 79 秒になります。<i>seconds</i> の正しい値は 0 から 120 です。</p> <p>このパラメータは省略可能です。ライセンス ファイルの最初の SERVER 行に記します。このパラメータを使用するには、バージョン 10.8 以降のベンダー デーモンを実行する必要があります。</p>

例:

```
SERVER my_server 17007ea8 21987
```

3 サーバによる冗長構成

3 サーバによる冗長構成は高品質のネットワークが必要です。この形態の冗長構成を実現するにはサーバがハートビートを定期的に交換する必要がありますので、通信が不調な場合は十分なパフォーマンスを得られない可能性があります。通信速度の遅い通信回線またはダイアルアップリンクによってサーバの冗長構成をすべきではありません。

ライセンス ファイルの同一のコピー (lmgrd およびベンダー デーモン バイナリ向け) をファイルサーバではなく各サーバのローカルディスクに置きます。このような置き方をしない場合、ライセンス ファイルを置いたファイルサーバが障害ポイントになるため、冗長サーバを構成する利点がすべて失われてしまいます。

ライセンス ファイルの中に 3 つの SERVER 行があれば、3 サーバによる冗長構成であることを示します。すべてのライセンス ファイルで、各サーバを示す 3 つの SERVER 行は同じ順番で記述されていなければなりません。lmgrd は次に示す処理を行うマスターサーバを常に動作させています。

- ライセンスを交付
- デバッグ ログに情報を記録
- レポート ログに警告を記録

既定ではプライマリサーバがマスターになります。プライマリサーバが使用不能になったときのマスターサーバの機能の移動方法は PRIMARY_IS_MASTER パラメータによって指定します。

冗長性を持たせた構成に 3 つのライセンス サーバが必要な理由は何ですか？

ライセンス サーバのトラブルに備えるために、複数のコンピュータで構成される冗長サーバシステムは、それぞれのサーバで、同一のライセンスのセットを提供できるようにしなければなりません。ソフトウェアのライセンス提供の一貫性とセキュリティを確保するために、冗長ライセンス サーバ システムでは常に 1 つのサーバだけがライセンスの提供を行うように設定されます。

ライセンス ファイルの SERVER 行によって冗長ライセンス サーバのコンピュータを示す番号であるホスト ID にライセンスのセットが結び付けられます。このようにして、各ライセンス サーバは同じライセンス セットを持った他の冗長ライセンス サーバと通信できるようになります。起動時に、それぞれのサーバは他の冗長ライセンス サーバと通信できるかどうかを確認します。グループのすべてのメンバーが同じグループの中のすべてのメンバーと通信ができる場合に、冗長ライセンス サーバのグループが形成されます。

いったんライセンス サーバのグループが形成されれば、グループは、複数のライセンス サーバの中の 1 つがいつでもライセンスを提供できることを保障します。しかし FLEXnet Licensing は、複数の冗長ライセンス サーバからグループを 1 つしか形成しません。1 つのグループしかないことを保障するために、過半数の冗長ライセンス サーバを持った 1 つのグループのみがライセンスの提供を許可されます。このライセンス サーバのグループはマジョリティと呼ばれます。マジョリティのメンバーでないライセンス サーバや単一のライセンス サーバは、他の冗長ライセンス サーバと通信できないのでライセンスを提供できません。マジョリティのメンバーでないライセンス サーバを実行させ続けることもできますが、既存のマジョリティにサーバを参加させるか、マジョリティが形成されていないときにはマジョリティを形成するために他のサーバと接続することができます。

ライセンス サーバがマジョリティと通信できなくなることを検出すると、通信できるようになるまでライセンスの提供を拒否します。マジョリティのメンバーは、他のライセンス サーバとの通信ができなくなると、マジョリティを見失ったとみなして、ライセンスの提供を拒否します。

FLEXnet Licensing がライセンスのセットを 2 つの冗長ライセンス サーバに結びつけることを許可すれば、上記の規則によって、冗長ライセンス サーバの過半数を含むグループにのみ、ライセンスの提供を許可します。グループにサーバが 2 台しかない場合、両方のライセンス サーバが常に通信できる状態で、どちらにもトラブルが発生してはならないことを意味します。つまり、2 台の過半数は 2 台であるため、これはフェイルオーバーのソリューションになりません。

マジョリティの要件によって、冗長ライセンス サーバ システムに必要なコンピュータの数は奇数になります。ライセンス サーバを偶数にすると、マジョリティになるために追加のライセンス サーバが要求されてしまいます。単純にするために、FLEXnet Licensing は 3 つの冗長ライセンス サーバで構成することだけをサポートします。3 は 1 の次に小さい奇数だからです。

関連事項:

カウントされない機能についての詳細は、「FEATURE 行とINCREMENT 行」を参照してください。
冗長構成のサーバについての詳細は、「ライセンス サーバ マシンの選択」を参照してください。
host に対する IP アドレスの指定は、バージョン 5.0 の *lmgrd*、ベンダー デーモン、クライアント ライブラリで導入
port の指定は、バージョン 6.0 以前の *lmgrd*、ベンダー デーモン、クライアント ライブラリでは必須
PRIMARY_IS_MASTER と HEARTBEAT_INTERVAL は、バージョン 10.8 の *lmgrd* とベンダー デーモンで導入

VENDOR 行

VENDOR 行はデーモン名を指定し、*lmgrd* はこの行を使用してベンダー デーモンを起動し、ベンダー デーモンはこの行を使用してオプション ファイルを読み込みます。VENDOR 行のフォーマットを次に示します。

```
VENDOR vendor [vendor_daemon_path]¥  
                [[OPTIONS=]options_file_path] [[PORT=]port]
```

以下に詳細を示します。

表 8-2: VENDOR 行のフォーマット

フィールド	説明
<i>vendor</i>	ファイル内の一部の機能を提供するために使用されるベンダー デーモンの名前。管理者はこの名前を変更できません。
<i>vendor_daemon_path</i>	オプションでこのデーモンの実行プログラムへのパス。通常、ライセンス管理者はデーモンを任意のディレクトリにインストールできます (ただし、ライセンス サーバ マシンのローカル ディレクトリにインストールすることをお勧めします)。 パスの指定を省略した場合、 <i>lmgrd</i> は、次のパス内のベンダー デーモンのバイナリ ファイルを探します。 <ul style="list-style-type: none">● 現在のディレクトリ● <i>lmgrd</i> の環境変数 \$PATH に指定されたパス● <i>lmgrd</i> が置かれたディレクトリ内 <i>vendor_daemon_path</i> がブランクの場合は、任意のオプションまたは TCP/IP ポート番号の指定には、文字列 OPTIONS= と PORT= が必要です。

表 8-2: VENDOR 行のフォーマット

フィールド	説明
<code>options_file_path</code>	<p>このデーモンに対するエンド ユーザのオプション ファイルへの完全パス。FLEXnet Licensing ではオプション ファイルが必須ではありません。</p> <p>このパスの指定を省略すると、既定により、ライセンス ファイルと同じディレクトリに置かれている <code>vendor.opt</code> というファイル (ここで <code>vendor</code> はベンダー デーモン名) を探します。</p>
<code>port</code>	<p>ベンダー デーモンの TCP/IP ポート番号。</p> <p><code>port</code> が指定されていない場合、既定ではオペレーティング システムが実行時にポート番号を選択します。インターネットのファイヤウォールを使用しているサイトでは、デーモンが使用する TCP/IP ポート番号を指定する必要があります。VENDOR 行で TCP/IP ポート番号を指定した場合、ベンダー デーモンの再起動に時間がかかることがあります。</p>

関連事項:

オプション ファイルの内容の詳細は、「[オプション ファイル](#)」を参照してください。

`vendor_daemon_path` は、バージョン 6.0 以前のベンダー デーモンでは必須

`options_file_path` は、バージョン 6.0 以前のベンダー デーモンでは必須

VENDOR 行は、バージョン 6.0 以前の `lmgrd` とベンダー デーモンでは DAEMON 行

バージョン 6.0 以降の場合を次に示します。

```
VENDOR sampled
```

バージョン 6.0 以前の場合を次に示します。

```
DAEMON sampled /etc/sampled ¥
      /etc/sampled/licenses/sampled.opt
```

- キーワード `options=` は、バージョン 5.0 のベンダー デーモンで導入

USE_SERVER 行

USE_SERVER は引数がなく、サーバには何の影響も与えません。アプリケーションは USE_SERVER を見つけると、ライセンス ファイルの先行する SERVER 行以外の行をすべて無視し、ベンダー デーモンにチェックアウトの検証を転送します。

ライセンス サーバ システムを使用する場合、性能向上に役立つので USE_SERVER を使用することをお勧めします。カウントされない機能に対しては、USE_SERVER を使用して強制的にデーモンに使用ログを記録させます。

FEATURE 行とINCREMENT 行

FEATURE 行は、製品を使用するために必要なライセンスを記述します。INCREMENT 行は、ライセンス ファイルの先行する FEATURE 行または INCREMENT 行にライセンスを追加するだけでなく、FEATURE 行の代わりに使用することもできます。

ベンダー デーモンは、指定された機能に対する最初の FEATURE 行のみを処理します。同一機能の追加コピーを必要とする場合(たとえば、ノードロックされた、カウントされる機能を複数使用する場合)、複数の INCREMENT 行を使用する必要があります。INCREMENT 行は、ライセンス グループまたは「プール」を、次のフィールドに基づいて形成します。

- feature name
- version
- DUP_GROUP
- FLOAT_OK
- HOST_BASED
- HOSTID
- PLATFORM
- USER_BASED
- VENDOR_STRING (プール識別要素としてベンダーによって設定されている場合)

FEATURE/INCREMENT 行の上記のフィールドが 1 つでも異なる場合、「ライセンス プール」と呼ばれる新しいライセンス グループがベンダー デーモンに作成され、同じ機能名を持つ他のライセンス プールとは別にカウントされます。INCREMENT 行は常に追加ライセンス数を指定しますが、FEATURE 行は追加ライセンス数を指定しません。

基本的な FEATURE/INCREMENT 行のフォーマットを次に示します。

```
{FEATURE|INCREMENT} feature vendor feat_version exp_date ¥  
num_lic SIGN=sign [optional_attributes]
```

FEATURE/INCREMENT 行のキーワードの後の 6 つのフィールドは必須であり、その順番は決まっています。これらのフィールドはベンダーによって定義され、変更できません。表 8-3 に、これらのフィールドを記述する順番に示します。

表 8-3: FEATURE/INCREMENT 行の必須フィールド

フィールド	説明
<i>feature</i>	ベンダーが機能に付けた名前
<i>vendor</i>	ベンダー デーモンの名前。VENDOR 行でも指定されます。指定されたデーモンがこの機能を交付します。
<i>feat_version</i>	このライセンスがサポートするこの機能のバージョン

表 8-3: FEATURE/INCREMENT 行の必須フィールド (続き)

フィールド	説明
<code>exp_date</code>	ライセンスの有効期限で、たとえば、07-may-2006 のように <code>dd-mmm-yyyy</code> のフォーマットで指定します。注: <code>exp_date</code> が文字列 "permanent" または年が 0 (または 00, 000, 0000) の場合、ライセンスは永久に有効です。
<code>num_lic</code>	この機能の同時ライセンスの数。 <code>num_lic</code> が、文字列 "uncounted" または 0 に設定されている場合、この機能のライセンスはカウントされず、 <code>lmgrd</code> も必要ありませんが、ホスト ID は FEATURE 行に記述する必要があります。詳細は、「 カウントされるライセンスとカウントされないライセンス 」を参照してください。
<code>SIGN=sign</code> または <code>AUTH=...</code>	この FEATURE 行を認証するための SIGN = 署名 コモン ベンダー デーモン テクノロジを使用してベンダー デーモンが配布する場合には、キーワード AUTH にライセンス証明を記述します。詳細は、ベンダーにお問い合わせください。

表 8-4 に、FEATURE 行または INCREMENT 行で指定される属性の一覧を示します。これらの属性はベンダーの裁量で提供され、特定のライセンス交付方式を規定します。FEATURE 行または INCREMENT 行にそのような属性が存在する場合、それはその行に存在しなくてはならず、エンド ユーザはそれを変更できません。これらの属性は、`keyword=value` 構文を持っており、ここでの `keyword` が大文字となっています。

`value` が二重引用符 ("...") で囲まれた文字列の場合、その場所の文字列には引用符以外のすべての文字が使用できます。

表 8-4: ベンダーが交付する属性

属性	説明
<code>BORROW[=n]</code>	特定の FEATURE/INCREMENT 行に対してライセンス借用を可能にします。 <code>n</code> はライセンスを借用する時間数です。既定の借用期間は 168 時間、つまり 1 週間です。

表 8-4: ベンダーが交付する属性 (続き)

属性	説明
DUP_GROUP=...	<p>構文は次のとおりです。</p> <pre>DUP_GROUP=NONE SITE [UHDV] U = DUP_USER H = DUP_HOST D = DUP_DISPLAY V = DUP_VENDOR_DEF</pre> <p>UHDV は任意に組み合わせることができ、DUP_MASK はその組み合わせの論理和になります。たとえば、DUP_GROUP=UHD は、重複のグループ化が、(DUP_USER DUP_HOST DUP_DISPLAY) であり、同一ホストの同一ディスプレイ上のユーザはライセンスを追加取得しなくても該当機能の追加使用ができることを意味します。</p>
FLOAT_OK [= <i>server_hostid</i>]	<p>FLOAT_OK を持つ FLEXid によって、特定の FEATURE/INCREMENT 行に対してモバイルライセンス管理を可能にします。また、FEATURE/INCREMENT 行も FLEXid にノードロックしなければなりません。</p> <p>FLOAT_OK=<i>server_hostid</i> が FEATURE 行で指定されている場合は、次の点に注意してください。</p> <p><i>server_hostid</i> はライセンス ファイルの SERVER 行に存在する同じホストを参照する必要があります。</p> <p>ライセンス サーバ システムは、Imhostid が返すホスト ID が、FLOAT_OK によって指定された Imhostid と同一のホスト ID であるマシン上でのみ動作します。</p>
HOSTID= "hostid1 [hostid2 ... hostidn]"	<p>FEATURE 行にバインドされたホスト ID。<i>hostid</i> は Imhostid ユーティリティによって決定されます。これは、カウントされないライセンスの必須フィールドですが、カウントされるライセンスにも使用できます。詳細は、「FLEXnet Licensing がサポートするプラットフォームのホスト ID」を参照してください。</p>
HOST_BASED[= <i>n</i>]	<p>ホスト名をエンドユーザのオプションファイル内の INCLUDE 文で指定する必要があり、ホストの数は <i>num_lic</i> または <i>n</i> で指定された値を最大値とします。</p>
ISSUED= <i>dd-mmm-yyyy</i>	発行日付
ISSUER="..."	ライセンスの発行者
NOTICE="..."	知的所有権告知用のフィールド

表 8-4: ベンダーが交付する属性 (続き)

属性	説明
OVERDRAFT= <i>n</i>	オーバードラフト ポリシーでは、ベンダーは購入ライセンス数の他にユーザが使用できる追加ライセンス数を指定できます。これにより、ユーザは「一時的な超過」状態にあるときでもサービスを拒否されません。上限を超えるライセンスの使用は、レポーティング ツール FLEXnet Manager により報告されます。
PLATFORMS="..."	リストに含まれるプラットフォームにのみ使用を許可します。
SN= <i>serial_num</i>	シリアル番号。FEATURE 行または INCREMENT 行の識別に使用します。
START= <i>dd-mm-yyy</i>	開始日
SUITE_DUP_GROUP=.	DUP_GROUP と似ていますが、パッケージを有効にする FEATURE 行だけに有効です。パッケージのユーザ総数をライセンス数の枠内に制限し、SUITE をチェックアウトしたユーザ間でパッケージを共有できるようにします。
SUPERSEDE= " <i>f1 f2 ...</i> "	この行が使用された場合、ISSUED= で指定される日付以前に発行されたすべてのライセンスは、この行によって破棄され、無効になります。
TS_OK	FLEXnet Licensing は、ノードロックされてカウントされないライセンスが、Windows のターミナル サーバで実行されるのを検出します。リモート ターミナル サーバのクライアント ウィンドウによってアプリケーションを実行するには、FEATURE 行に TS_OK を追加する必要があります。TS_OK がないと、ターミナル サーバクライアント上で実行しようとするユーザはライセンスを拒否されます。
USER_BASED[= <i>n</i>]	ユーザをエンド ユーザのオプション ファイル内の INCLUDE 文で指定する必要があり、ユーザの数は <i>num_lic</i> または <i>=n</i> で指定された値を最大値とします。
VENDOR_STRING= "..."	ベンダーが定義する文字列で、二重引用符で囲まれます。

表 8-5 にリストされた次の属性はオプションであり、エンド ユーザがコントロールします。これらの属性は、*keyword=value* 構文を持っており、ここでの *keyword* は小文字となっています。

表 8-5: エンド ユーザの属性

属性	説明
asset_info="..."	資産管理のためにライセンス管理者が提供する追加情報

表 8-5: エンド ユーザの属性 (続き)

属性	説明
<code>dist_info="..."</code>	ソフトウェアのディストリビュータが提供する追加情報
<code>sort=nnn</code>	ライセンス ファイルの行のソート順序を指定。詳細は、「優先順位」を参照してください。
<code>user_info="..."</code>	ライセンス管理者が提供する追加情報
<code>vendor_info="..."</code>	ソフトウェア ベンダーが提供する追加情報

例

```
FEATURE sample_app sampled 2.300 31-dec-2005 20 ¥  
SIGN=123456789012  
INCREMENT fl sampled 1.000 permanent 5 ¥  
HOSTID=INTERNET=195.186.*.* NOTICE="Licensed to ¥  
Sample corp" SIGN=901234567890
```

優先順位

FEATURE/INCREMENT ライセンス ファイルの行は、FLEXnet Licensing によって処理されるときに、自動的にソートされます。既定のソート規則は次のとおりです。

1. ライセンス ファイル。自動ソートは、ライセンス ファイル リスト内の複数ファイルを跨って行われません。
2. 機能名。
3. INCREMENT の前に FEATURE。
4. 「カウントされる」の前に「カウントされない」。
5. バージョン。古いバージョンの前に新しいバージョン。
6. 発行された日付。逆順。最新のデータが先頭。日付は、ISSUED= または START= から取得。
7. 以上に該当しなければ、元の順序が保持されます。

自動ソートをオフにするには、FEATURE/INCREMENT 行に `sort=nnn` を記入します。ここで、`nnn` は、すべての行で同じです。`nnn` は、相対ソート順序を指定します。既定のソート順序の値は、100 です。100 より小さい値のソート順序の値が指定された行は、この属性が指定されていないすべての行の前にソートされます。100 より大きいソート順序の値が指定された行は、値が指定されていないすべての行の後ろに現れます。同じ番号が指定された行は、ファイル内と同じ順序になります。

- バージョン 7.1 以前の FEATURE/INCREMENT 行は次のような *license_key* を使用します。

```
{FEATURE|INCREMENT} feature vendor feat_version exp_date  
num_lic license_key [optional_attributes]
```

現在のクライアント ライブラリとベンダー デーモンは、バージョン 7.1 以前のフォーマットを読み取ることができます。

- SIGN= キーワードは、バージョン 7.1 クライアント ライブラリとベンダー デーモンで導入
- バージョン 7.1 ~ バージョン 8.0 クライアント ライブラリとベンダー デーモンについては、FEATURE/INCREMENT 行には SIGN= 署名が含まれていなければならないが、バージョン 8.1 以前の後方互換性については、次のような *license_key* を含めることができます。

```
{FEATURE|INCREMENT} feature vendor feat_version exp_date ¥  
num_lic [license_key] SIGN=sign ¥  
[optional_attributes]
```

- バージョン 8.1 クライアント ライブラリとベンダー デーモンでは、*license_key* は使用されない。
- *exp_date* のキーワード "permanent" は、バージョン 6 クライアント ライブラリで導入
- *num_lic* のキーワード "uncounted" は、バージョン 6 クライアント ライブラリで導入
- キーワード BORROW は、バージョン 8.0 クライアント ライブラリとベンダー デーモンで導入
- キーワード FLOAT_OK は、バージョン 8.0 クライアント ライブラリとベンダー デーモンで導入
- キーワード TS_OK は、バージョン 8.0 クライアント ライブラリとベンダー デーモンで導入
- キーワード AUTH は、バージョン 10.8 クライアント ライブラリとベンダー デーモンで導入

PACKAGE 行

PACKAGE 行は、次の 2 つのライセンス管理の必要性に対応します。

- 製品スイートをライセンスする。
- 大量の機能が含まれるライセンス ファイル(そのほとんどが同じ FEATURE 行の引数を共有)を配布する効率的な方法を提供する。

PACKAGE 行は、それ自体は何もライセンスしません。パッケージ全体をライセンスするために、PACKAGE 行と対応する FEATURE/INCREMENT 行を必要とします。PACKAGE 行は、ソフトウェア ベンダーによって、ライセンスとは別個に製品に添付して提供されます。後でそのパッケージ用のライセンスを購入したときに、1 つまたは複数の対応する FEATURE /INCREMENT 行により PACKAGE 行が有効になります。

例:

```
PACKAGE package vendor [pkg_version] COMPONENTS=pkg_list ¥  
    [OPTIONS=SUITE] [SUPERSEDE=["p1 p2 ..."]] ISSUED=date]  
SIGN=pkg_sign
```

表 8-6 に PACKAGE 行のフィールドの一覧を示します。これらのフィールドは、表に記載されている順番に記述されます。

表 8-6: PACKAGE 行のフィールド

フィールド	説明
<i>package</i>	パッケージ名。対応する FEATURE/INCREMENT 行は、同一名でなければなりません。
<i>vendor</i>	このパッケージをサポートするベンダー デーモンの名前
<i>pkg_version</i>	パッケージのバージョンを指定するオプションのフィールド。これを指定する場合、パッケージを有効にする FEATURE/INCREMENT 行は、同一バージョンでなければなりません。
COMPONENTS=pkg_list	パッケージ コンポーネントのリスト。次に、そのフォーマットを示します。 <i>feature[:version[:num_lic]]</i> パッケージには少なくとも 1 つのコンポーネントが入っていなければなりません。バージョンとカウントは任意に指定できます。省略した場合は、対応する FEATURE/INCREMENT 行の値が使用されます。 <i>num_lic</i> は、 OPTIONS=SUITE が設定されない場合のみ意味を持ちます。この場合、ライセンス数は COMPONENTS 行の <i>num_lic</i> に FEATURE/INCREMENT 行のライセンス数を乗じた値になります。例: COMPONENTS="comp1 comp2 comp3 comp4" COMPONENTS="comp1:1.5 comp2 comp3:2.0:4"

表 8-6: PACKAGE 行のフィールド (続き)

フィールド	説明
OPTIONS=SUIE	<p>オプションフィールド。この PACKAGE 行は製品パッケージスイート用であることを示します。</p> <p>これを指定すると、チェックアウトされているコンポーネント機能の他に、パッケージと同じ名前に対応する機能がチェックアウトされます。</p> <p>これを指定しないと、パッケージと同じ名前に対応する機能はパッケージが有効になった後に除去され、コンポーネント機能がチェックアウトされても、この機能はチェックアウトされません。</p>
OPTIONS=SUIE_RESERVED	<p>オプションフィールド。これを設定すると、パッケージコンポーネントセットが予約されます。1つのパッケージコンポーネントがチェックアウトされると、その他のすべてのコンポーネントはそれと同じユーザに予約されます。</p>
SUPERSEDE [="p1 p2 ..."]	<p>オプションフィールド。ISSUED の発行日と連携して使用されます。dd-mm-yyy 以前の ISSUED の発行日を持つ同じパッケージ名のすべての PACKAGE 行を置き換えます。</p>
ISSUED=dd-mm-yyy	<p>オプションフィールド。SUPERSEDE と連携して使用されます。dd-mm-yyy 以前の ISSUED の発行日を持つ同じパッケージ名のすべての PACKAGE 行を置き換えます。</p>
SIGN=sign または AUTH=...	<p>この FEATURE 行を認証するための SIGN = 署名</p> <p>コモンベンダーデーモンテクノロジーを使用してベンダーデーモンが配布する場合には、キーワード AUTH にライセンス証明を記述します。詳細は、ベンダーにお問い合わせください。</p>

例

```
PACKAGE suite sampled 1.0 SIGN=3B24B2F508CB ¥
      COMPONENTS="comp1 comp2" OPTIONS=SUIE
FEATURE suite sampled 1.0 1-jan-0 5 SIGN=4193E6ABCCCB
```

これは典型的な OPTIONS=SUIE の例です。"comp1"、"comp2" という 2 つの機能があり、どちらもバージョンが 1.0 です。それぞれには 5 つの無期限ライセンスが付与されています。"comp1" または "comp2" がチェックアウトされると、"suite" もチェックアウトされます。

```
PACKAGE suite sampled 1.0 SIGN=2CBF44FCB9C1 ¥  
    COMPONENTS="apple:1.5:2 orange:3.0:4"  
FEATURE suite sampled 1.0 1-jan-2005 3 SIGN=321E78A17EC1 SN=123
```

この例では、コンポーネントのバージョンが機能のバージョンを上書きします。コンポーネントに対して使用可能なライセンス数は、"suite" の 3 つのライセンスとそのコンポーネントに対するライセンス数の積です。結果は、次の定義と等しくなります。

```
FEATURE apple sampled 1.5 1-jan-2005 6 SIGN=0D3AD5F26BEC SN=123  
FEATURE orange sampled 3.0 1-jan-2005 12 SIGN=EB16C5AE61F0 SN=123
```

- PACKAGE 行を別のファイルに格納する機能は、バージョン 6 クライアント ライブラリで導入
- *pkg_version* フィールドは、バージョン 7.1 以前のクライアント ライブラリでは必須
- キーワード AUTH は、バージョン 10.8 クライアント ライブラリとベンダー デモンで導入

UPGRADE 行

```
UPGRADE feature vendor from_feat_version to_feat_version ¥  
exp_date num_lic [options ... ] SIGN=sign
```

FEATURE 行または INCREMENT 行とまったく同じデータに、*from_feat_version* フィールドが追加されています。UPGRADE 行は、指定されたライセンス数を上限として、*from_feat_version* で指定されたバージョン以降の旧バージョンのライセンスを削除し、削除されたライセンス数分の新しいバージョンを作成します。

たとえば、次の 2 行は、

```
INCREMENT f1 sampled 1.000 1-jan-2005 5 SIGN=9BFAC0316462  
UPGRADE f1 sampled 1.000 2.000 1-jan-2005 2 SIGN=1B9A308CC0F7
```

f1 の 3 つのバージョン 1.0 ライセンスと f1 の 2 つのバージョン 2.0 のライセンスを規定しています。

UPGRADE 行は、先行する FEATURE 行および INCREMENT 行の中で一番近くにあり、*from_feat_version* 以降かつ *to_feat_version* より前のバージョン番号を持つ行に作用します。



注: UPGRADE 行は、ノードロックされ、カウントされないライセンスには作用しません。

10 進数のフォーマット

ライセンスを 10 進数のフォーマットで表すことができます。10 進数には、入力が簡単であることと、ライセンスを非常に短くできるという利点があります。

可読フォーマットの簡単なデモ ライセンスを次に示します。

```
FEATURE f1 sampled 1.00 1-jan-2006 0 key1 HOSTID=DEMO
```

これを 10 進数で表すと、次のようになります。

```
sampled-f1-00737-55296-1825
```

ライセンス ファイルで 10 進数の行と可読フォーマットを行を、必要に応じて一緒に使用することもできます。10 進数の行を可読フォーマットに変換するには、`lminstall` コマンドを使用します。

関連事項:

`lminstall` コマンドの詳細は、「[lminstall](#)」を参照してください。

10 進数のフォーマットは、バージョン 6 クライアント ライブラリとベンダー デーモンで導入

ライセンス ファイルの行順

ライセンス ファイル内の行の順序は重要ではありません。処理されるときにこれらの行は並べ替えられるので、ほとんどの場合結果に影響しません。ただし、バージョン 7.0 以前の FLEX 対応アプリケーションとライセンス サーバシステムでは、暗黙的にライセンス ファイルの行を順序付けます。ライセンス ファイル内の行の順序付けには、次の点を注意してください。

- 同一機能の場合は、FEATURE 行を INCREMENT 行の前に置きます。

FEATURE 行に関しては、ライセンス サーバシステムが有効とするのは最初にカウントされる FEATURE 行のみであり、FEATURE 行と INCREMENT 行がある場合は FEATURE 行を最初に記述する必要があるという規則があります

- 同一機能にカウントされる FEATURE 行が複数存在する場合、希望する FEATURE 行が最初に記述されていることを確認します。

最初に記述された FEATURE 行以外はすべて無視されます。

- ノードロックされ、カウントされない FEATURE 行を同一機能のフローティング FEATURE 行の前に置きます。このような順序にしないと、ノードロックされたライセンスを使用できる場合でもフローティング ライセンスが使用され、他のユーザが拒否されます。
- `USE_SERVER` 行の配置が、動作に影響を与えます。 `USE_SERVER` 行を使用することをお勧めします。通常、 `USE_SERVER` 行は `SERVER` 行の直後に記述し

ます。ただし、SERVER により交付されず、カウントされないライセンスを、USE_SERVER 行の前に置きます。カウントされないライセンスを必要とする各ユーザが、ライセンス ファイルの最新のコピーに直接アクセスできることを確認してください。USE_SERVER 行を SERVER 行の直後に記述する利点は、ユーザがライセンス ファイルの最新のコピーを必要としないことです。

関連事項:
「優先順位」

ライセンス サーバ マネージャ

ライセンス サーバ マネージャ `lmgrd` は、ライセンス サーバ システムを構成する 2 つの FLEXnet Licensing コンポーネントの 1 つです (もう一方はベンダー デーモンです)。ライセンス マネージャ デーモンは FLEX 対応アプリケーションと最初に通信し、適切なベンダー デーモンと接続させます。ライセンス サーバ マネージャ `lmgrd` は次にあげる目的で使用されます。

- ライセンス ファイルの `VENDOR` 行でリストされた、すべてのベンダー デーモンを起動し、保持する。
- アプリケーションのチェックアウト要求 (またはその他の要求) を、正しいベンダー デーモンに差し向ける。

新しい `lmgrd` は、古いベンダー デーモンまたは FLEX 対応アプリケーションと共に使用できますが、新しいベンダー デーモンまたは FLEX 対応アプリケーションは、古い `lmgrd` とでは正しく動作しない場合があります。できるだけ最新バージョンの `lmgrd` を使用してください。これは www.macrovision.com からダウンロードすることができます。

lmgrd コマンドライン構文

`lmgrd` は、FLEXnet Licensing 用のメインデーモンプログラムです。`lmgrd` を呼び出すと、`lmgrd` はベンダーと機能の情報を含むライセンス ファイルを探し、それらのベンダー デーモンを起動します。

次のように使用します。

```
lmgrd [-c license_file_list] [-l [+]debug_log_path]  
      [-2 -p] [-local] [-x lmdown] [-x lmremove] [-z ] [-v] [-help]
```

以下に詳細を示します。

表 9-1: lmgrd コマンドライン構文の使用法

オプション	説明
-c <i>license_file_list</i>	指定されたライセンス ファイルを使用します。
-l [+] <i>debug_log_path</i>	デバッグ情報をファイル <i>debug_log_path</i> に書き込みます。このオプションは、数字の 1 ではなく、文字 "l"(エル)を使用します。 <i>debug_log_path</i> の前に + 記号を付けるとログ エントリが追加されます。このファイルの詳細は、「 デバッグ ログ ファイル 」を参照してください。
-2 -p	lmdown, lmreread, lmremove の使用を、FLEXnet Licensing 管理者(既定値では root ユーザ)のみに制限します。UNIX の lmadm グループが存在する場合、そのグループのメンバーのみに使用が制限されます。root がこのグループのメンバーでない場合、root は、上に示したユーティリティのいずれをも使用する権限を持ちません。-2 -p が lmgrd の起動時に使用された場合、Windows 上のどのユーザも、lmdown によってライセンス サーバ システムをシャットダウンすることはできません。
-local	lmdown および lmreread コマンドが、lmgrd が実行中のマシンのみから起動されるように制限します。
-x lmdown	lmdown コマンドを無効にします(どのユーザも lmdown を実行することはできません)。lmdown が無効にされた場合、kill pid によって lmgrd を停止するか (UNIX の場合)、タスク マネージャまたは Windows サービスによって lmgrd とベンダー デーモンプロセスを停止 (Windows の場合) します。UNIX の場合、kill コマンドに -9 引数を使用しないでください。(バージョン 4 以降の lmgrd)
-x lmremove	lmremove コマンドを無効にします(どのユーザも lmremove を実行することはできません)。
-z	フォアグラウンドで実行します。既定値では、バックグラウンドで実行されます。-l (エル) <i>debug_log_path</i> が存在する場合、新たにウィンドウは開きませんが、-l (エル) 引数が指定されていない場合、lmgrd と各ベンダー デーモン用に別々のウィンドウが使用されます。
-v	lmgrd のバージョン番号と著作権を表示して終了します。
-help	使用法を表示して終了します。

UNIXプラットフォームでライセンスサーバマネージャを起動する

ライセンスファイルにカウントされるライセンスがある場合(ライセンス数 > 0)、先にライセンスサーバマネージャを、結果的にライセンスサーバシステムを起動しないと FLEX 対応アプリケーションを使用することはできません。

ライセンスサーバマネージャ `lmgrd` は、コマンドラインから手動で起動するか、システムの起動時に自動的に起動されます。この両方の起動方法を次のセクションで説明します。



注: `lmgrd` は、ライセンスファイルの `SERVER` 行で指定されたサーバマシンでのみ起動します。

3サーバによる冗長構成のライセンスサーバシステムを実行している場合、各ライセンスファイルのコピーを、`lmgrd` とベンダーデーモンのバイナリファイルと共に、ファイルサーバではなく、それぞれのサーバマシンでローカルに維持してください。このような置き方をしない場合、それらのファイルを置いたファイルサーバが単独の障害ポイントになるため、冗長サーバを構成する利点がすべて失われてしまいます。

手動起動

次の構文を使用して、UNIX のコマンドラインから `lmgrd` を起動します。

```
lmgrd -c license_file_list -L [+]debug_log_path
```

以下に詳細を示します。

- `license_file_list` には次の 1 つまたは両方が含まれます。
 - 1 つのライセンスファイルへの完全パス
 - ディレクトリ。そのディレクトリ内の `*.lic` という名前のファイルがすべて使用されます。
 - `debug_log_path` は、デバッグログファイルへの完全パスです。

`debug_log_path` の前に `+` 記号を付けると、ログエントリが付け足されていきます。

`root` ユーザから起動したプロセスはセキュリティ上リスクがあるため、"`root`" ではないユーザから `lmgrd` を起動します。`lmgrd` を `root` ユーザから起動しなければならないときは、`su` コマンドを使用して、`lmgrd` を特権のないユーザとして実行します。

```
su username -c "lmgrd -c license_file_list -l debug_log_path"
```

ここで、`username` は特権のないユーザです。ライセンス ファイルにリストされたベンダー デーモンに、`username` に対する実行許可があることを確認する必要があります。すべてのベンダー デーモンへのパスは、ライセンス ファイルの各 `VENDOR` 行に示されます。

自動起動

UNIX 上で、`/etc/rc.boot`、`/etc/rc.local`、`/etc/rc2.d/Sxxx`、`/sbin/rc2.d/Sxxxx` などから適切なブート スクリプトを編集します。次の例のようなコマンドを含めます。詳細については、後述の注意事項を参照してください。

```
/bin/su daniel -c 'echo starting lmgrd > ¥  
/home/flexlm/v5.12/hp700_u9/boot.log'
```

```
/bin/nohup /bin/su daniel -c 'umask 022; ¥  
/home/flexlm/v5.12/hp700_u9/lmgrd -c ¥  
/home/flexlm/v5.12/hp700_u9/license.dat >> ¥  
/home/flexlm/v5.12/hp700_u9/boot.log'
```

```
/bin/su daniel -c 'echo sleep 5 >> ¥  
/home/flexlm/v5.12/hp700_u9/boot.log'
```

```
/bin/sleep 5
```

```
/bin/su daniel -c 'echo lmdiag >>¥  
/home/flexlm/v5.12/hp700_u9/boot.log'
```

```
/bin/su daniel -c '/home/flexlm/v5.12/hp700_u9/lmdiag -n -c¥  
/home/flexlm/v5.12/hp700_u9/license.dat >> ¥  
/home/flexlm/v5.12/hp700_u9/boot.log'
```

```
/bin/su daniel -c 'echo exiting >>¥  
/home/flexlm/v5.12/hp700_u9/boot.log'
```

このスクリプトを作成するときは、次のことに注意してください。

- すべてのパスを完全パスで指定します。これはブート時にはパスの仮定ができないためです。
- パスの仮定ができないので、ベンダー デーモンを `lmgrd` と同じディレクトリに置くか、`VENDOR` 行にベンダー デーモンへの完全パスが含まれるように編集する必要があります。
- `root` 以外のユーザ `daniel` として `lmgrd` を実行するためには `su` コマンドを使用します。`root` 権限を必要としないプログラムを "`root`" として実行するのはセキュリティリスクになる可能性があるため、`lmgrd` を "`root`" として実行しないことをお勧めします。`lmgrd` は `root` 権限を必要としません。

- `daniel` は `csch` にログインするので、`daniel` として実行するコマンドはすべて `csch` の構文で記述する必要があります。`daniel` として実行しないコマンドはすべて、ブートスクリプトが使用する `/bin/sh` の構文で記述する必要があります。
- HP-UX や Digital UNIX など一部のオペレーティング システムでは、`nohup` と `sleep` を使用する必要があります。`Solaris` や他のオペレーティング システムではそれらのコマンドを使用する必要はありませんが、どのオペレーティング システムでも使用の方が安全です。
- `lmdiag` は、サーバが実行中でライセンスを供給していることを確認するための診断ツールとして使用します。



注: 上記設定は、ライセンス サーバ マシンを再起動するまではデーモンを起動しません。

Windows でライセンス サーバ マネージャを起動する

これは、Windows プラットフォームで起動するライセンス サーバ マネージャに対する一般的な方法ではありません。このセクションでは、コマンド ラインから手動で起動する方法、およびライセンス サーバ マネージャをサービスとして設定する方法を説明します。

コマンド ラインから手動で起動する



操作手順 コマンド ラインから `lmgrd` を起動するには

1. Windows のコマンド シェルで次の構文を入力し、`lmgrd` をアプリケーションとして起動します。

```
C:¥flexlm> lmgrd -c license_file_list -L [+]debug_log_path
```

以下に詳細を示します。

- `license_file_list` には次の 1 つまたは両方が含まれます。
 - 1 つのライセンス ファイルへの完全パス
 - ディレクトリ。そのディレクトリ内の `*.lic` という名前のファイルがすべて使用されます。
- `debug_log_path` は、デバッグ ログ ファイルへの完全パスです。

`debug_log_path` の前に + 記号を付けると、ログ エントリが付け足されていきます。

パス名内にスペースが存在する場合は、パスの前後に二重引用符を付ける必要があります。

Windows では、`lmgrd` をサービスとしてインストールすることができます。その結果、`lmgrd` がバックグラウンドで実行されますが、ユーザインタフェースから起動、停止できるようにします。

Windows サービスとしてライセンス サーバ マネージャを設定する

ライセンス サーバシステムをサービスの1つとして設定するには、管理者権限が必要です。



操作手順 ライセンス サーバシステムをサービスとして設定するには

1. `LMTOOLS` を実行します。
2. `[Configuration using Services]` ボタンをオンにし、`[Config Services]` タブをクリックします。
3. `[Service Name]` テキスト ボックスに、定義するサービスの名前、たとえば `Myvendor License Manager` と入力します。
4. `[Path to the lmgrd.exe file]` フィールドで、このライセンス サーバ システム用の `lmgrd.exe` のパスを直接入力するか、または参照してこれを入力します。
5. `[Path to the license file]` フィールドで、このライセンス サーバ システムのライセンス ファイルを直接入力するか、または参照してこれを入力します。
6. `[Path to the debug log file]` フィールドで、このライセンス サーバ システムのデバッグ ログ ファイルを直接入力するか、または参照してこれを入力します。デバッグ ログ ファイル名の前に + 記号を付けると、ログ エントリが付け足されていきます。デバッグ ログ ファイルの既定値の場所は、`c:\%winnt%\System32` フォルダです。別の場所を指定するには、完全パスを指定する必要があります。

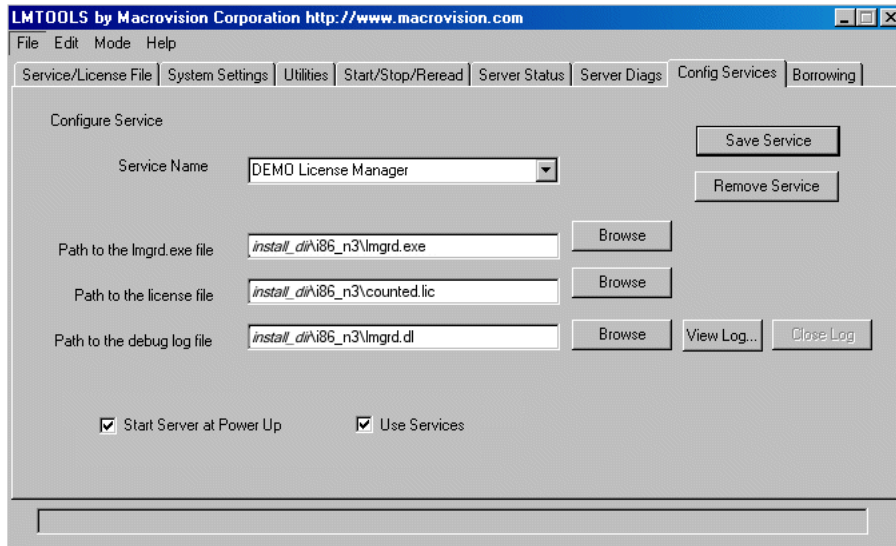


図 9-1: 設定が完了した LM_TOOLS の画面

7. 新しい Myvendor License Manager サービスを保存するには、[Save Service] をクリックします。

LMTOOLS から手動で起動する

ライセンス サーバ マネージャ ツールのグラフィカル ユーザ インタフェースは、LMTOOLS という名前で提供されています。LMTOOLS には主に次のような機能があります。

- FLEXnet ライセンス サーバ システムの起動、停止、設定
- ホスト ID を含むシステム情報の取得
- サーバ ステータスの取得

lmgrd の操作を LMTOOLS ユーザ インタフェースからコントロールするには、はじめにこれをライセンス サーバ マネージャ サービスとして設定する必要があります。設定をはじめる前に、「[Windows サービスとしてライセンス サーバ マネージャを設定する](#)」の手順に従ってください。

ライセンス サーバ マネージャ サービスを設定すると、LMTOOLS インタフェースから lmgrd を起動したり停止することができます。



操作手順 LMTOOLS インタフェースからサービスを起動するには

1. LMTOOLS を起動します。

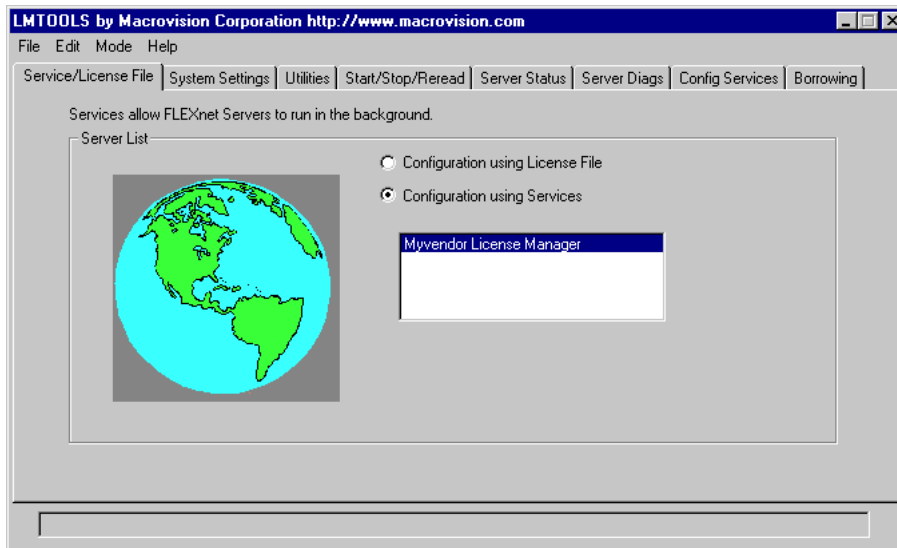


図 9-2: [Service/License File] タブが開いた LMTOOLS が表示されます。

2. [Configuration using Services] ボタンをオンにします。
3. 選択ボックス内のリストからサービス名を選択します。この例では、サービス名は Myvendor License Manager です。
4. [Start/Stop/Reread] タブをクリックします。

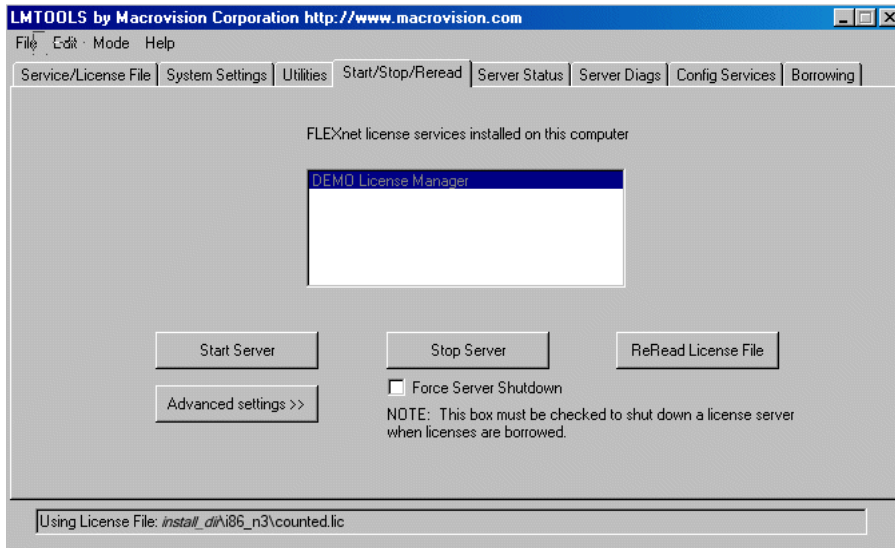


図 9-3: LMTOOLS の [Start/Stop/Reread] タブ

5. Myvendor License Manager サービスを起動するには、[Start Server] をクリックします。

Myvendor License Manager ライセンス サーバシステムが起動し、そのデバッグ ログ出力を `c:\¥flexlm¥mgrd.dl` に書き込みます。

システム起動時に自動的に起動する

システムの起動時に `lmgrd` を起動し、そこからライセンス サーバシステムを自動的に起動するには、これをライセンス サーバ マネージャ サービスとして設定する必要があります。設定をはじめの前に、「[Windows サービスとしてライセンス サーバ マネージャを設定する](#)」の手順に従ってください。



操作手順 ライセンス マネージャ サービスとして Imgrd を設定するには

1. LMTOOLS を起動し、サービス名を選択して、[Config Services] タブをクリックします。

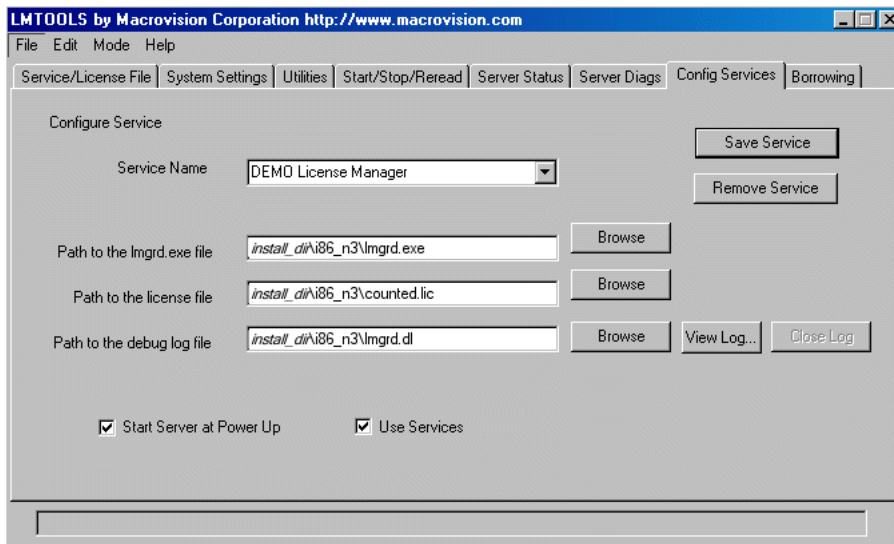


図 9-4: LMTOOLS の [Config Services] タブ

2. このライセンス サーバ マネージャを Windows のサービスにするには、次の手順で行います。[Use Services] チェック ボックスをオンにします (さもないと、FLEXnet Licensing サービスとなってしまいます)。
3. [Start Server at Power Up] チェック ボックスをオンにして、システムの起動時にこのサービスが起動するように設定します。

これ以降、マシンを再起動すると、ライセンス サーバ マネージャが Windows サービスとして自動的に起動します。

A

トラブルシューティング ガイド

この付録では、これまでユーザが直面してきた FLEXnet Licensing に関する問題について説明します。

一般的なトラブルシューティングのヒント

以下に、一般的なデバッグ情報を示します。

- ライセンス サーバ システムを起動するときに、後で検証できるよう、ログ ファイルをローカルに出力するように設定してください。ログ ファイルには役立つ情報が含まれていることがよくあります。問題があるときはよく調査し、サポート要員に説明するときに質問に答えられるように準備をしてください。
- ライセンス サーバ システムが正しく立ち上がっている場合は(ログ ファイルを調べて確認できます)、`lmstat -a` および `lmdiag` の実行を試み、使用中のアプリケーションと同じ問題があるかどうかを調べます。
- アプリケーションがバージョン 4.1 以降 (Windows の場合は、バージョン 5 以降)であれば、環境変数 `FLEXLM_DIAGNOSTICS` を使用できます。 `FLEXLM_DIAGNOSTICS` を 1、2、または 3 に設定します。3 に設定すると、2 よりも多くの情報が、2 に設定すると 1 よりも多くの情報(特に、拒否された機能名)が得られます。 詳細は、「[FLEXLM_DIAGNOSTICS](#)」を参照してください。

- サポート担当者に説明するときは、次の質問に対する答えを準備しておいてください。
 - どのような種類のマシンでライセンス サーバシステムを実行していますか？
 - オペレーティング システムのバージョンはなんですか？
 - どのようなマシンとオペレーティング システムで、アプリケーションを実行していますか？
 - どのバージョンの FLEXnet Licensing を、FLEX 対応アプリケーションで使用していますか？

lmver スクリプトを使用するか、または UNIX 上で次のコマンドを lmgrd、ベンダー デーモン、およびアプリケーションに対して実行します。

```
strings binary_name | grep Copy
```

この代わりとして lmgrd -v は、lmgrd のバージョンを知らせます。これはベンダー デーモンでも動作します。

- ログ ファイルには、どんなエラーや警告メッセージがありますか？
- サーバは正しく起動しましたか？ 次のようなメッセージを探してください。

```
server xyz started for: feature1 feature2.
```
- lmstat -a を実行した結果、どのように出力されましたか？
- 他の FLEX 対応製品を実行していますか？
- 結合したライセンス ファイルを使用していますか、それとも個々のライセンス ファイルを使用していますか？
- 3 サーバによる冗長構成のライセンス サーバ システム (ライセンス ファイル内の複数の SERVER 行) を使用中ですか？

FLEXLM_DIAGNOSTICS



注: FLEXnet Licensing が診断出力を生成する機能は、使用しているソフトウェアのベンダーによって管理されています。

FLEXLM_DIAGNOSTICS は、チェックアウトが拒否されたときに、アプリケーションに診断情報を作成させる環境変数です。時代とともに診断情報のフォーマットが変わる可能性があります。

UNIX では、診断の出力先は stderr です。

Windows では、出力は flexpid.log という現在のディレクトリにあるファイルです。pid は、アプリケーションのプロセス ID です。

レベル 1 の内容

FLEXLM_DIAGNOSTICS を 1 に設定すると、アプリケーションが使用しようとしたすべてのライセンス ファイルのリストの他に、標準の FLEXnet Licensing エラー メッセージが出力されます。例を次に示します。

```
setenv FLEXLM_DIAGNOSTICS 1
FLEXlm checkout error: Cannot find license file (-1,73:2) No such file
or directory
license file(s): /usr/myproduct/licenses/testing.lic license.lic
```

レベル 2 の内容

FLEXLM_DIAGNOSTICS を 2 に設定すると、レベル 1 の出力に加え、チェックアウト引数が出力されます。例を次に示します。

```
setenv FLEXLM_DIAGNOSTICS 2
FLEXlm checkout error: No such feature exists (-5,116:2) No such file
or directory
license file(s): /usr/myproduct/licenses/testing.lic license.lic
lm_checkout("f1", 1.0, 1, 0x0, ..., 0x4000)
```

エラー メッセージに、チェックアウト中に起きた 2 つの別々の問題が実際に含まれることに注目してください。

- 見つかったライセンスには、該当する機能はありません。
- その他のライセンス ファイルは見つかりませんでした。メッセージは、**No such file or directory** となります

次に、`lm_checkout` に対する引数について説明します。

```
lm_checkout(feature, version, num_lic, queue_flag, ..., dupgroup_mask)
```

以下に詳細を示します。

表 A-1: `lm_checkout` の引数

引数	説明
<code>feature</code>	要求した機能
<code>version</code>	要求したバージョン。ライセンス ファイルには、要求したバージョン以降のバージョンが含まれていなければなりません。
<code>num_lic</code>	要求したライセンス数。通常は 1 です。

表 A-1: lm_checkout の引数

引数	説明
queue_flag	0 のときは、順番待ちなし 1 のときはライセンス待ち(「ブロックされた」順番待ち) 2 のときはライセンス待ちですが、アプリケーションに戻ります(「ブロックされない」順番待ち)。
dupgroup_mask	ライセンスの共有とも呼ばれる重複グループ化を指示します。ユーザ、ホスト、およびディスプレイは、lmstat -a で出力されます。

レベル 3 の内容(バージョン 6.0 以降のみ)

FLEXLM_DIAGNOSTICS を 3 に設定すると、レベル 1, 2 の出力に加えて、チェックアウトが正常に行われた場合には、ライセンスがどのように認可されたかを説明する情報が出力されます。

```
setenv FLEXLM_DIAGNOSTICS 3
app
Checkout succeeded: f0/14263EAEA8E0
License file: ./servtest.lic
No server used
app2
Checkout succeeded: f1/BC64A7B120AE
License file: @localhost
License Server System: @localhost
app3
Checkout succeeded: f1/BC64A7B120AE
License file: servtest.lic
License Server System: @speedy
```

機能名、ライセンス キーが、ライセンス ファイルの位置(または @host が使用されていたときはホスト名) およびサーバのホスト名(利用されれば)と共に出力されることに注目してください。

FLEXnet Licensing 環境変数

FLEX 対応アプリケーションを使用するときは、環境変数が必須ではありません。環境変数は、通常、デバッグするとき、またはライセンスの既定の場所を変更するときに使用します。

環境変数の設定方法

FLEXnet Licensing 環境変数は、次の 2 つの方法で設定されます。

- プロセス環境内で設定
- レジストリ (Windows でバージョン 6 以降)、または UNIX 上で FLEXnet Licensing レジストリのように機能する `$HOME/.flexlmrc` (UNIX でバージョン 7 以降) 内で設定

レジストリ

Windows では、FLEXnet Licensing レジストリは次の場所にあります。

```
HKEY_LOCAL_MACHINE\Software\FLEXlm License Manager
```

UNIX では、同様の情報は `$HOME/.flexlmrc` に格納されています。このファイルでは、構文は `variable=value` です。

優先順位

変数が LM_LICENSE_FILE または VENDOR_LICENSE_FILE のときは、環境変数とレジストリの両方が使用され、環境変数はパスの先頭に、レジストリは後ろに付けられます。

他の変数の場合、環境変数が設定されると環境変数だけが使用され、その他の場合はレジストリが使用されます。つまり、レジストリは環境変数が設定されない場合にのみ使用されます。

環境変数

下の表に FLEXnet Licensing の環境変数とその定義を示します。

表 B-1: FLEXnet Licensing 環境変数

変数	定義
FLEXLM_BATCH	Windows のみ: 対話ポップアップが表示されないようにします。バッチアプリケーションでは、1 に設定します。(バージョン 7.0 以降のクライアント)
FLEXLM_DIAGNOSTICS	アプリケーションが FLEXnet Licensing エラー メッセージ テキストを出力しない場合にデバッグに使用します。希望する診断情報量に従って、1、2、または 3 に設定します。詳細は、 「FLEXLM_DIAGNOSTICS」 を参照してください。(バージョン 5.0 以降のクライアント)
FLEXLM_TIMEOUT	Windows のみ: FLEX 対応アプリケーションがライセンス サーバシステムのポート (27000-27009 の範囲) に接続を試みるときのタイムアウトを設定します。 値は μ 秒 (100 分の 1 秒) であり、範囲は 0 から 2147483647 です。既定値は 100,000 μ 秒です。
LM_BORROW	ライセンスの借用の開始や借用期間の設定に使用します。詳細は、 「ライセンスの借用を開始する」 を参照してください。UNIX プラットフォームでは、 <code>\$HOME/.flexlmrc</code> ではなく、 <code>\$HOME/.flexlmborrow</code> がレジストリに使用されます。

表 B-1: FLEXnet Licensing 環境変数

変数	定義
LM_PROJECT	<p>LM_PROJECT の値はレポート ログ ファイルにログ記録され、後で FLEXnet Manager によってレポートされます。最大 30 文字に制限されます。(バージョン 5.0 以降のクライアントに必要)</p> <p>ライセンスの RESERVE、INCLUDE などの設定に PROJECT で使用できません。例を次に示します。</p> <p>RESERVE 1 fl PROJECT airplane</p> <p>この機能には、バージョン 5.0 以降のクライアントとバージョン 7.0 以降のベンダー デーモンが必要です。</p>
LM_SERVER_HIGHEST_FD	それ以上ではライセンス サーバがアクセスしない、最も高いファイル記述子の値を設定するために使用します。
LM_UTIL_CASE_SENSITIVE	FLEXlm ユーティリティが使用します。1 にすると、ユーティリティはライセンス ファイルの大文字小文字を区別します。既定値では、この環境変数は 0 です。ライセンス ファイルの大文字小文字は区別されません。ライセンス ファイルが大文字小文字を区別するようにベンダーによって環境設定された場合には、この環境変数はライセンス サーバだけに適用されます。
TCP_NODELAY	ライセンス要求を処理するときの、FLEXnet ライセンス サーバのパフォーマンスを向上します。1 にすると、パフォーマンス改善機能を有効にします。使用上の注意: 有効にするとネットワーク通信量が増加します。
LM_LICENSE_FILE または次のように記述します。	ライセンス ファイルへのパスを設定します。UNIX の場合は ":"(コロン)、Windows の場合は ";"(セミコロン)で区切って、ライセンス ファイルのリストにすることも可能です。VENDOR_LICENSE_FILE を使用する場合、VENDOR はアプリケーションで使用されるベンダー デーモン名です。たとえば、Macrovision の製品は MVS_LICENSE_FILE を使用します。ファイル名または、port@host とすることができます。 「環境変数でパスを設定する」も参照してください。 (VENDOR_LICENSE_FILE はバージョン 6.0 以降のクライアントが必要)
VENDOR_LICENSE_FILE	

FLEXnet Licensing のエラーコード

このセクションでは、FLEXnet Licensing のエラー メッセージについて、一般的なフォーマットとエラー メッセージの内容を説明します。

エラー メッセージのフォーマット

アプリケーションが提示する FLEXnet Licensing エラー メッセージには、次の要素があります。

- FLEXnet Licensing のエラー番号は、-1 から始まる負の番号です。
- FLEXnet Licensing のエラーテキストは、短文 (80 文字未満) で、問題を要約しています。
- FLEXnet Licensing のエラー説明 (オプション) は、短い段落 (3-5 行) で、問題および可能な解決方法、または回避策を説明しています。
- FLEXnet Licensing のマイナー エラー番号は、1 から始まる正の番号です。この番号は、一意のエラー識別子で FLEXnet Licensing ベンダーにより高度なサポート支援に使用されます。その意味については、文書化されていません。
- システム エラー番号 (オプション) は、UNIX または Windows OS のエラーコードで、オペレーティング システムで設定された最新のものです。
- システム エラー説明 (オプション) は、短文 (80 文字未満) で、システム エラーを説明しています。
- その他のサポート情報 (オプション)

エラー メッセージは、バージョン 6 で改良されました。FLEXnet Licensing のエラー説明、およびサポート情報は、バージョン 6.0 以降を使用するアプリケーションのみが利用できます。

このエラー メッセージは、FLEXnet Licensing で利用できる 次の 2 つの フォーマットで表示されるか、またはアプリケーションでカスタマイズされたフォーマットで表示されます。

フォーマット 1 (短文)

```
FLEXlm error text (-lm_errno, minor_num[:sys_errno]) [sys_error_text]  
システム エラー情報はないことがあります。
```

例:

```
Can't connect to license server system (-15,12:61) Connection refused
```

フォーマット 2 (長文、バージョン 6 以降)

```
FLEXlm error text  
FLEXlm error explanation  
[Optional Supporting information]  
FLEXlm error: -lm_errno, minor num. [System Error: sys_errno]  
["system_error_text"]
```

例:

```
Cannot connect to license server system  
The server (lmgrd) has not been started yet, or  
the wrong port@host or license file is being used, or the  
port or hostname in the license file has been changed.  
Feature:      f1  
Server name:  localhost  
License path: @localhost:license.dat:./*.lic  
FLEXlm error: -15,12. System Error: 61 "Connection refused"
```

エラー コードの説明

次の表に、FLEX 対応製品が出力する一般的なエラーの一覧を示します。

表 C-1: FLEXnet Licensing のエラー コード

エラー コード	説明
-1	ライセンス ファイルを見つけることができません。
-2	無効なライセンス ファイルの構文です。
-3	この機能のライセンス サーバ システムがありません。
-4	ライセンスの最大ユーザ数にすでに達しています。
-5	その機能はありません。
-6	ライセンス ファイルに TCP/IP ポート番号がなく、 <i>FLEXnet Licensing</i> サービスもありません。(v6 より前のみ)
-7	ライセンス サーバ マネージャ サービスへのソケット接続がありません。
-8	無効な(矛盾する)ライセンス キーまたは署名です。 このシステムのホスト ID が、ライセンス ファイルで指定されたホスト ID と一致しません。これは通常ライセンス ファイルが変更されたときに起こります。
-9	無効なホストです。 このシステムのホスト ID が、指定されたライセンス ファイルのホスト ID と一致しません。
-10	機能は期限切れになりました。
-11	ライセンス ファイル内の日付のフォーマットが無効です。
-12	ライセンス サーバ システムから戻されたデータが無効です。
-13	ライセンス ファイルに SERVER 行がありません。
-14	ネットワーク データベースで SERVER ホスト名を見つけることができません。 ライセンス ファイルの SERVER 行のホスト名の検索に失敗しました。これは NIS、DNS またはホスト名が誤っている場合によく発生します。回避策: IP アドレス(たとえば、123.456.789.123)をホスト名の代わりに使用します。

表 C-1: FLEXnet Licensing のエラー コード (続き)

エラー コード	説明
-15	ライセンス サーバ システムに接続できません。 サーバ (Imgrd) がまだ起動されていません、または間違った <code>port@host</code> 、またはライセンス ファイルが使用されています。またはライセンス ファイルの TCP/IP ポートまたはホスト名が変更されました。 Windows XP SP2 プラットフォームには、毎秒あたり接続可能な TCP/IP の数に制限があり、ご使用のアプリケーションがこの制限を超えた可能性があります。この制限を変更する方法については、製造業者のドキュメントを参照してください。
-16	ライセンス サーバ システムからデータを読み込むことができません。
-17	ライセンス サーバ システムにデータを書き出すことができません。
-18	ライセンス サーバ システムはこの機能をサポートしていません。
-19	select システム コールのエラーです。
-21	ライセンス サーバはこのバージョンをサポートしていません。
-22	ライセンス サーバ システムで、機能のチェックイン失敗が検出されました。
-23	ライセンス サーバ システムは、一時的にビジー状態です (新規のサーバを接続中)。
-24	ユーザはこの機能の順番待ち行列に入れられました。
-25	ライセンス サーバ システムは、このバージョンのこの機能をサポートしません。
-26	サポートされているこの機能の数を超えるライセンスの要求です。
-29	イーサネット デバイスが見つかりません。
-30	ライセンス ファイルが読み込めません。
-31	機能開始日が未来の日付です。
-32	そのような属性はありません。
-33	ベンダー デーモンとの暗号化ハンドシェークはできません。
-34	クライアントとライセンス サーバ システム間のクロック差が大きすぎます。
-35	この機能は順番待ちです。
-36	ベンダー デーモンで機能データベースが破損しています。

表 C-1: FLEXnet Licensing のエラー コード (続き)

エラー コード	説明
-37	この機能にはない重複選択です。バージョン 8.0 以降のベンダー デーモンでは使用しません。
-38	EXCLUDE 機能リストにあるユーザまたはホストです。
-39	INCLUDE 機能リストにないユーザまたはホストです。
-40	動的なメモリを割り当てることができません。
-41	いままでチェックアウトされたことがない機能です。
-42	無効なパラメータです。
-47	ベンダー デーモンでは、クロック設定チェックは利用できません。
-52	ベンダー デーモンがタイムアウト間隔内に応答しません。
-53	チェックアウト要求がベンダー定義のチェックアウト フィルタで拒否されました。
-54	ライセンス ファイルに FEATURESET 行がありません。
-55	ライセンス ファイルの FEATURESET 行が不正です。
-56	ライセンス ファイルからの FEATURESET データを計算できません。
-57 ^a	socket コールに失敗しました。
-59	メッセージのチェックサムが失敗しました。
-60	ライセンス サーバシステム メッセージのチェックサムが不正です。
-61	ライセンス サーバシステムからライセンス ファイルデータが読み込めません。
-62	ネットワーク ソフトウェア (TCP/IP) が利用できません。
-63	ライセンス管理者ではありません。
-64	最小 Imremove 間隔より前の Imremove 要求です。
-67	借用可能な機能がありません。
-68	ライセンスの BORROW サポートが有効ではありません。
-69	FLOAT_OK がライセンス サーバシステム上でスタンドアロンで利用できません。

表 C-1: FLEXnet Licensing のエラー コード (続き)

エラー コード	説明
-71	無効な環境変数 TZ です。
-73	ローカルのチェックアウト フィルタが要求を拒否しました。
-74	ライセンス ファイル パスの最後より後を読み込もうとしています。
-75 ^{<Super>a}	SYSS\$SETIMR コールに失敗しました (VMS)。
-76	<i>FLEXnet Licensing</i> 内部のエラーです。Macrovision Corporation に連絡してください。
-77	不正なバージョン番号です。文字を含まない浮動小数点番号でなければなりません。
-82	ライセンス ファイルの PACKAGE 行が無効です。
-83	クライアントの <i>FLEXnet Licensing</i> バージョンがサーバより新しいです。
-84	USER_BASED ライセンスに指定のユーザがありません。ライセンス サーバシステムのログを参照してください。
-85	ライセンス サーバシステムはこの要求をサポートしません。
-87	チェックアウトがオプション ファイルで指定された MAX を超えています。
-88	システム クロックが巻き戻されました。
-89	このプラットフォームは、ライセンスで許可されていません。
-90	該当ファイルはこのプログラムの理解できない新バージョンの <i>FLEXnet Licensing</i> で発行されたものです。 このプログラムでは理解できない最新バージョンの <i>FLEXnet Licensing</i> でファイルが発行されました。
-91	暗号化のシードがユニークではありません。
-92	lmreread の間に機能が削除されたか、SERVER 行のホスト ID が間違っています。
-93	この機能は別のライセンス プールで利用できます。 警告状態です。サーバが1 つまたは複数の INCREMENT 行を単一プールにプールし、リクエストされた INCREMENT 行はプールされています。
-94	互換性のない属性でライセンスを発行しようとしています。

表 C-1: FLEXnet Licensing のエラー コード (続き)

エラー コード	説明
-95	THIS_HOST へのネットワーク接続が失敗しました。 ライセンス ファイルの SERVER 行上の this_host を、実際のホスト名に変更します。
-96	ライセンス サーバマシンがダウンしているか、応答しません。 サーバの起動については、システム管理者に確認するか、または正しいホストを参照していることを確認します(環境変数 LM_LICENSE_FILE を参照してください)。
-97	対象とするベンダー デーモンがダウンしています。 1) lmgrd ログ ファイルをチェックします。または 2) lmread を試してみます。
-98	この FEATURE 行は 10 進数のフォーマットに変換できません。
-99	10 進数フォーマットのライセンスが間違っって入力されました。
-100	保持中のライセンスが削除できません。
-101	ライセンスはすべて、その他のユーザに予約済みです。 システム管理者がその他のユーザ向けにすべてのライセンスを予約しました。予約は、オプション ファイルで実行されます。オプション ファイルの変更を有効にするには、サーバを再起動することが必要です。
-102	FLEXid の借用エラーが発生しました。
-103	許可されていないターミナル サーバのリモート クライアントです。
-104	それを長期間借用できません。
-106	ライセンス サーバシステムが利用できるネットワーク接続はなくなりました。 ベンダー デーモンはこれ以上のユーザを扱うことができません。詳細情報については、デバッグ ログを参照してください。
-110	dongle を読み取れません。 dongle またはドライバを調べてください。 dongle が装着されていないか、このタイプの dongle に必要なソフトウェア ドライバがインストールされていないかのいずれかです。
-112	dongle ドライバがありません。 FLEXid のホスト ID を読み取るには、正しいドライバをインストールすることが必要です。このドライバは、ソフトウェア ベンダーから入手できます。

表 C-1: FLEXnet Licensing のエラー コード (続き)

エラー コード	説明
-114	SIGN= キーワードが必要とされましたが、ライセンスの証明に存在しません。ベンダーから SIGN= バージョンのこのライセンスを入手することが必要です。
-115	公開キー パッケージでのエラーです。
-116	TRL はこのプラットフォームにサポートされていません。
-117	BORROW が失敗しました。
-118	BORROW は期限切れです。
-119	lmdown と lmread はライセンス サーバマシンで実行すべきです。
-120	ライセンスが借用されているときに、サーバを lmdown できません。
-121	FLOAT_OK ライセンスには、FLEXid ホスト ID が必ず 1 つだけ必要です。
-122	ローカルの借用情報を削除できません。
-123	借用されたライセンスの早期返還は、サポートされていません。詳細は、ベンダーにお問い合わせください。
-124	借用ライセンスを返却する際にエラーが発生しました。
-125	1 つの PACKAGE コンポーネントを指定しなければなりません。
-126	合成ホスト ID が初期化されていません。
-127	合成ホスト ID に必要な項目が見つからないか無効です。
-128	エラー。借用したライセンスが既知のサーバライセンスと一致しません。
-135	イベント ログを有効にする際にエラーが発生しました。
-136	イベント ログは無効です。
-137	イベント ログに書き込む際にエラーが発生しました。
-139	通信がタイムアウトになりました。
-140	メッセージ コマンドが不正です。
-141	ソケットに書き出す際にエラーが発生しました。ソケットは閉じられました。
-142	エラー。単一のホスト ID に結び付けられたバージョン固有のライセンスを生成できません。これは、合成ホスト ID です。

表 C-1: FLEXnet Licensing のエラー コード (続き)

エラー コード	説明
-143	バージョン固有の署名は、カウントされないライセンスに対してサポートされていません。
-144	ライセンス テンプレートに、余分な署名指定が含まれています。
-145	V71_LK 署名が不正です。
-146	V71_SIGN 署名が不正です。
-147	V80_LK 署名が不正です。
-148	V80_SIGN 署名が不正です。
-149	V81_LK 署名が不正です。
-150	V81_SIGN 署名が不正です。
-151	V81_SIGN2 署名が不正です。
-152	V84_LK 署名が不正です。
-153	V84_SIGN 署名が不正です。
-154	V84_SIGN2 署名が不正です。
-155	ライセンス キーが必要ですが、ライセンス証明に存在しません。アプリケーションには、ライセンス証明内にライセンス キーが必要です。ベンダーからこの証明のライセンス キー パージョンを入手する必要があります。
-156	AUTH= keyword で指定された署名が無効です。
-157	Trusted storage が異常です。修復する必要があります。ベンダーに問い合わせ、修復に必要な情報を入手してください。
-158	Trusted storage のオープンに失敗しました。詳細は、ベンダーにお問い合わせください。
-159	Fulfillment レコードが不正です。詳細は、ベンダーにお問い合わせください。
-160	受信したアクティベーション要求が無効です。詳細は、ベンダーにお問い合わせください。
-161	trusted storage に、要求に合う fulfillment が存在しません。詳細は、ベンダーにお問い合わせください。
-162	受信したアクティベーション応答が無効です。詳細は、ベンダーにお問い合わせください。

表 C-1: FLEXnet Licensing のエラー コード (続き)

エラー コード	説明
-163	指定されたアクティベーションを返すことができません。詳細は、ベンダーにお問い合わせください。
-164	リターン数が、fulfillment の最大数を超えます。詳細は、ベンダーにお問い合わせください。
-165	修復数が残っていません。ベンダーに問い合わせて、オーソライゼーションの修復に必要な情報を入手してください。
-166	指定された操作を実行できません。詳細は、ベンダーにお問い合わせください。
-167	ユーザーまたはホストが、オプション ファイルの指定によって、この所有資格のアクティベーションが禁止されているため、要求されたアクティベーションは否定されました。
-168	オプション ファイルには所有資格の指定が含まれ、それらの指定に現在のユーザまたはホストが含まれていません。
-500	サーバポート番号が無効です。
-501	ライセンス内の値が無効です。整数値でなければなりません。
-502	カウントに対して無効な値が指定されました。
-503	ライセンスで無効なホスト ID が指定されました。
-504	無効なホスト ID のタイプが指定されました。
-505	FEATURE 行の構文が不正です。
-506	<i>FLEXnet Licensing</i> 内部のエラーです。
-507	ライセンス ファイル内の日付のフォーマットが不正です。
-508	SERVER 行が不正です。
-509	ライセンス文字列が不正です。
-510	サーバの機能は、クライアント側で認証しません。
-511	チェックアウトされたライセンスは、ありません。
-512	ライセンスは、既にチェックアウトされています。
-513	エラー リストが返されました。
-514	certicom モジュールがありません。

表 C-1: FLEXnet Licensing のエラー コード (続き)

エラー コード	説明
-515	certicom モジュールが不正、または破損しています。
-516	ライセンス証明には、SIGN または SIGN2 が必要です。
-517	機能オブジェクトにライセンス ソースがありません。
-518	同じライセンスが、このライセンス ソース上で既にチェック アウトされています。
-519	このライセンスのチェックアウトは、非同期の順番待ち状態になっています。
-521	ネイティブ ホスト ID のライブラリをロードできません。
-522	既に他のベンダー デーモンに接続されています。
-523	そのようなユーザ、ホスト、ディスプレイはありません。
-524	ライセンス サーバ システムのシャットダウンに失敗しました。
-525	シャットダウンに失敗しました。既にライセンス サーバ システムに接続されています。
-526	無効なライセンス ソース文字列です。
-527	ログ ファイル スイッチ エラー。

a. オペレーティング システムの障害によるエラーを示しています。

D

レポート ログ ファイル

ライセンス サーバシステムは、レポート ログ ファイルとデバッグ ログ ファイルの両方を作成します。この付録ではレポート ログ ファイルを説明します。デバッグ ログ ファイルの詳細は、「[デバッグ ログ ファイル](#)」を参照してください。

レポート ログ ファイルは機能の使用状況に関する情報を格納し、このファイルはベンダーデーモンによって生成されます。ただし、既定値では、ベンダーデーモンはレポート ログを書き出しません。この動作を有効にする必要があります。レポート ログのデータは圧縮、暗号化されており、リポジトリで管理できます。

Macrovision のソフトウェア ライセンス管理ソリューションである FLEXnet Manager を使用して、ライセンス使用データに対する優れた視点を提供し、ライセンスの有効性や使用法などの重要な情報に関する有益なレポートを作成します。FLEXnet Manager は、スケジュールに基づいて、これらのレポートの実行を完全に自動化でき、Windows NT、Linux、UNIX サーバを含む異種ネットワークでのライセンス サーバと使用状況の追跡に使用することができます。ユーザ企業の FLEXnet Manager の評価版を入手する方法の詳細については、Macrovision (www.macrovision.com) に問い合わせてください。

レポート ログ出力を管理する

ベンダーデーモンをある程度の時間実行すると、レポート ログの量が増加します。ライセンス動作が頻繁に発生する場合、これらのログ ファイルのサイズが非常に大きくなります。したがって、これらのファイルの保存場所や、ローテーション頻度、保存頻度を検討する必要があります。したがって、レポート ログの出力先を途中から別のファイルにローテーションしたり切り換えたりする必要があります。各ファイルには、それぞれ別の期間のライセンス アクティビティのログが含まれます。

レポート ログ データは、ベンダー デーモンによって収集され、出力ファイルにフラッシュされるまで、内部データ バッファ領域に置かれます。デーモンの内部バッファは、毎分、または先に一杯になったたびにフラッシュされます。最新のデータをレポート ログ ファイルで入手できるようにするには、`lmreread` コマンドを使用してデーモンのバッファをフラッシュします。標準のファイル圧縮ツールを使用して、書き込みが終わった時点でレポート ログ ファイルのサイズを小さくします。

破損を回避し、パフォーマンスを向上させるために、ベンダー デーモンが、ベンダー デーモンを実行中のシステムに対してローカルなディスク上のファイルにそのレポート ログ を書き込むことをお勧めします。各ベンダー デーモンは、それぞれ独自のレポート ログ ファイルに書き込む必要があります。

ベンダー デーモンのレポート ログ出力を有効にする

特定のベンダー デーモンのレポート ログ の記録を、ライセンスサーバシステムの起動前(または起動後)に有効にするには、2 つの方法があります。

- `REPORTLOG` 行をベンダー デーモンのオプション ファイルに追加します。詳細は、「[REPORTLOG](#)」を参照してください。
- ベンダー デーモンに対して `lmswitchr` を起動します。詳細は、「[lmswitchr](#)」を参照してください。

ベンダー デーモンのレポート ログ出力先を変更する

特定のベンダー デーモンのレポート ログ 出力を別のファイルに変更することができます。各ファイルは別の時間帯の動作を表します。これを実行するには、ベンダー デーモンが実行中かどうかによって、3 通りの方法があります。

- ベンダー デーモンのオプション ファイルの `REPORTLOG` 行を変更し、`lmreread` を起動するか(バージョン 8.0 以降のベンダー デーモン)、または再起動して、そのオプション ファイルを再読み込みします。
- ベンダー デーモンに対して `lmswitchr` を起動します。詳細は、「[lmswitchr](#)」を参照してください。
- ベンダー デーモンに対して `lmnewlog` を起動します。これには、バージョン 7.1 以降のベンダー デーモンが必要です。詳細は、「[lmnewlog](#)」を参照してください。

デバッグ ログ ファイル

ライセンス サーバシステムは、デバッグ ログ ファイルとレポート ログ ファイルの両方を作成します。この付録ではデバッグ ログ ファイルを説明します。レポート ログ ファイルの詳細は、「[レポート ログ ファイル](#)」を参照してください。

デバッグ ログ ファイルには、ライセンス サーバシステムをデバッグするときに役に立つステータスやエラー メッセージが含まれます。ライセンス サーバシステムは常にデバッグ ログ出力を生成します。デバッグ ログの出力には、lmgrd 固有のイベントを記述するものや、各ベンダー デーモン固有のイベントを記述するものがあります。

デバッグ ログ出力を管理する

lmgrd とベンダー デーモンが一定時間起動すると、この出力の量が増加します。ログが古くなるとデバッグ ログ出力の価値が減少します。したがって、古いデバッグ ログ出力を現在の出力から切り離し、その古い出力を記録するか、削除する必要があります。

パフォーマンスのために、lmgrd とそのベンダー デーモンが実行されているマシンに対してローカルとなるディスクに、各デバッグ ログ ファイルを置くことをお勧めします。ただし、デバッグ ログ ファイルをリモートにマウントされたディスクに置かなければならず、しかもライセンス サーバシステムの実行速度が非常に遅い場合は、パフォーマンスを改善するために、`-nfs_log` オプションを付けて lmgrd を起動します。

デバッグ ログ出力フォーマットの説明は、「[デバッグ ログ メッセージ](#)」を参照してください。

ライセンス サーバ システムのデバッグ ログ 出力をキャプチャする

既定値では、`lmgrd` と `lmgrd` が管理するベンダー デーモンはデバッグ ログ出力を標準の出力へ書き込みます。このデバッグ ログ出力をファイルに書き込むには、ライセンス サーバ システムの出力をファイルにリダイレクトするか、`-l debug_log_path` オプションを付けて、`lmgrd` を起動します。

特定のベンダー デーモンのデバッグ ログ出力 をキャプチャする

同じライセンス サーバ システムがコントロールする異なるベンダー デーモンからのデバッグ ログ出力を、それぞれのベンダー デーモン独自のファイルに書き込むことができます (バージョン 8.0 以降のベンダー デーモン)。これを実現する方法は 2 つあります。

- `DEBUGLOG` 行をベンダー デーモンのオプション ファイルに追加します。詳細は、「[DEBUGLOG](#)」を参照してください。
- ベンダー デーモンに対して `lmswitch` を起動します。詳細は、「[lmswitch](#)」を参照してください。

`lmgrd` が、自身のデバッグ ログ出力を標準出力に書き込みます。

実行中のベンダー デーモンのデバッグ ログの 出力先を変更する

個々のベンダー デーモンに対するデバッグ ログの出力先を別のファイルに変更することができます。これを実現する方法は 2 つあります。

- `DEBUGLOG` 行を ベンダー デーモンのオプション ファイルに変更し、`lmreread` を起動して、そのオプション ファイルを再読み込みします。詳細は、「[DEBUGLOG](#)」を参照してください。
- ベンダー デーモンに対して `lmswitch` を起動します。詳細は、「[lmswitch](#)」を参照してください。

ベンダー デーモンのデバッグ ログ出力を制限する

既定では、デバッグ ログ出力にはすべてのイベントが含まれます。特定のベンダー デーモンのために記録されるイベントを制限するには、NOLOG 行を対象のベンダー デーモンのオプション ファイルに追加します。記録するイベントを制限する理由のひとつは、デバッグ ログ出力のサイズを小さくすることです。

関連事項:

[「NOLOG」](#)

デバッグ ログ メッセージ

FLEXnet Licensing 処理は、次のフォーマットでデバッグ ログ ファイルを生成します。

```
hh:mm:ss (daemon) message
```

以下に詳細を示します。

表 E-1: デバッグ ログ メッセージ

メッセージ	説明
<i>hh:mm:ss</i>	メッセージが記録された時間
<i>daemon</i>	lmgrd またはベンダー デーモン名のいずれかです。デーモンの単一コピーが要求されたライセンスのすべてを扱うことができなない場合は、後に番号が続くオプションの "_" (アンダーバー) で、このメッセージが分岐デーモンからであることを示します。
<i>message</i>	メッセージのテキスト

デバッグ ログ ファイルは次の目的に使用します。

- 設定の問題の診断
- デーモン ソフトウェア エラーの診断



注: デバッグ ログ ファイルは、FLEXnet Manager による使用状況のレポートには使用できません。

情報メッセージ

表 E-2 に FLEXnet Licensing で使用される情報メッセージの一覧を示します。

表 E-2: FLEXnet Licensing の情報メッセージ

メッセージ	説明
Connected to host	このデーモンは、host 上の同格のデーモンに接続しました。
CONNECTED, master is host	定数に達し、すべてがマスタを選択した時点で、ライセンス デーモンはこのメッセージを記録します。
DENIED: num_lic feature to user	user は、feature の num_lic ライセンスへのアクセスを拒否されました。
EXITING DUE TO SIGNAL nnn EXITING with code nnn	すべてのデーモンはデーモンが終了した理由を一覧表示します。
EXPIRED: feature	feature が期限切れです
IN: "feature" user (num_lic licenses)	user は、feature ライセンスの num_lic にチェックインしました。
Lost connection to host	デーモンは、もはや host ノードにある同格のデーモンとは通信できません。そのためクライアントが再接続を行うことになるか、デーモンが最低定足数未満になります。後者の場合、クライアントが終了することがあります。ライセンス デーモンがマスタへの接続を失うと、すべてのベンダーデーモンをシャットダウンします。ベンダーデーモンは、自身のシャットダウンを行います。
Lost quorum	デーモンが定足数を失いました。したがって、その他デーモンからの接続要求だけを処理します。
MULTIPLE vendor servers running. Kill and restart license daemon.	ライセンス サーバマネージャ lmgrd が、vendor の複数のベンダーデーモンが動作中であることを検出しました。lmgrd とすべての vendor デーモンを、lmdown でシャットダウンし、次に lmgrd を再起動します。
OUT: feature user (num_lic licenses)	user は、feature の num_lic ライセンスをチェックアウトしました。
RESERVE feature for USER user RESERVE feature for HOST host	feature のライセンスが user または host のいずれかに予約されています。
REStarted vendor (internet port nnn)	ベンダーデーモン vendor がTCP/IP ポート nnn で再起動されました。

表 E-2: FLEXnet Licensing の情報メッセージ

メッセージ	説明
Retrying socket bind (address in use)	"使用中のアドレス" エラーを検出すると、約 6 分間、ライセンス サーバ システムはソケットのバインドを試みます。
Selected (EXISTING) master host.	ライセンス デーモンは既存のマスタ <i>host</i> をマスタとして選択しました。
SERVER shutdown requested.	デーモンは、ユーザからの <code>kill</code> コマンドによってシャットダウンが要求されました。
Server started on host for: <i>feature_list</i>	リストされた機能で(おそらく新規の)サーバが起動されました。
Shutting down vendor	ライセンス サーバ マネージャがベンダー デーモン、 <code>vendor</code> をシャットダウン中です。
SIGCHLD received. Killing child servers.	ライセンス デーモンからシャットダウンを要求されたときは、ベンダー デーモンはこのメッセージを記録します。
Started vendor	新規のベンダー デーモンを起動するときは必ず、ライセンス サーバ マネージャはこのメッセージを記録します。
Trying to connect to host	デーモンが <i>host</i> に接続しようとしています。

設定の問題のメッセージ

表 E-3 に FLEXnet Licensing に現われる設定の問題のメッセージの一覧を示します。

表 E-3: FLEXnet Licensing の設定の問題のメッセージ

メッセージ	説明
host: Not a valid server host, exiting	このデーモンが無効なホスト名で実行されました。
host: Wrong hostid, exiting	<i>host</i> の ホスト ID が間違っています。
BAD CODE for feature	指定された機能名に不正なライセンス キーまたは署名があります。エンド ユーザが間違っ入力したか、または修正したと思われます。
CANNOT OPEN options file	ライセンス ファイルで指定されたオプション ファイルを開くことができません。
Couldn't find a master	デーモンはマスタについて合意できません。

表 E-3: FLEXnet Licensing の設定の問題のメッセージ

メッセージ	説明
License daemon: lost all connections	このメッセージはサーバへのすべての接続が失われたときに記録されます。ネットワークの問題を示していることがよくあります。
Lost lock, exiting Error closing lock file Unable to re-open lock file	ベンダー デーモンのロック ファイルに問題があります。通常は単一のノードで複数のデーモン コピーを実行しようとするものによるものです。ps コマンドを介して動作しているその他のデーモンを探して、kill -9 で強制終了させます。
No DAEMON line for vendor	ライセンス ファイルに、vendor の DAEMON 行または VENDOR 行がありません。
No DAEMON lines, exiting	ライセンス ファイルに DAEMON 行または VENDOR 行がないときは、ライセンス デーモンはこのメッセージを記録します。起動するベンダー デーモンがないので、ライセンス デーモンには何もすることがありません。
No features to serve!	ベンダー デーモンが交付する機能が見つかりません。壊れているか、または不正に入力されたライセンス ファイルが原因と思われます。
UNSUPPORTED FEATURE request: feature by user	ユーザがこのベンダー デーモンがサポートしない機能を要求しました。考えられる理由がいくつかあります。ライセンス ファイルが不正、機能の期限切れ、またはデーモンが間違ったライセンス ファイルにアクセス中の場合です。
Unknown host: host	ライセンス ファイルの SERVER 行で指定されたホスト名が、ネットワークのデータベースにありません (/etc/hosts と考えられます)。

デーモン ソフトウェアのエラー メッセージ

表 E-4 にデーモン ソフトウェアのエラー メッセージの一覧を示します。

表 E-4: デーモン ソフトウェアのエラー メッセージ

メッセージ	説明
accept: message	accept システム コールで、エラーが検出されました。
Can't allocate server table space	malloc エラーです。スワップ空間をチェックしてください。

表 E-4: デモン ソフトウェアのエラー メッセージ

メッセージ	説明
Connection to <i>host</i> TIMED OUT	デーモンが <i>host</i> に接続できません。
Illegal connection request to <i>vendor</i>	<i>vendor</i> に接続要求がありましたが、このベンダー デーモンは <i>vendor</i> ではありません。
read: error message	"read" システム コールで、エラーが検出されました。
select: message	"select" システム コールで、エラーが検出されました。これは通常、システムのネットワーク障害を示します。
Server exiting	サーバが終了中です。これは通常、エラーによるものです。

FLEXnet Licensing のバージョン

FLEXnet Licensing コンポーネントのバージョンの互換性

一般的に、www.macrovision.com から入手できる常に最新の `lmgrd` および `lmutil/LMTOOLS` を使用すると、FLEXnet Licensing の最新バージョンで利用できる多くの拡張機能を自動的に使用することができます。ただし、拡張機能には、最新バージョンの FLEXnet Licensing でビルドされたベンダーデーモン、さらには最新バージョンの FLEXnet Licensing でビルドされた FLEX 対応アプリケーションが必要なものもあります。ベンダーデーモンの最新バージョンについては、ソフトウェアベンダーにお問い合わせください。

FLEXnet Licensing のバージョンの互換性についての規則は、次のように要約することができます。

- `lmutil/LMTOOLS` のバージョン \geq
- `lmgrd` のバージョン \geq
- ベンダーデーモンのバージョン \geq
- FLEX 対応アプリケーションにリンクされたクライアントライブラリのバージョン \geq
- ライセンスファイルフォーマットのバージョン

ライセンスファイルを除いては、`lmver` を使用すると、これらすべての FLEXnet Licensing のコンポーネントのバージョンを知ることができます。ベンダーデーモン、`lmgrd`、および `lmutil` では、`-v` 引数を使用して、バージョンを出力することもできます。

ライセンス ファイルのバージョンの見分け方

次の規則が個々の FEATURE 行、INCREMENT 行、または UPGRADE 行に適用されます。1 つのファイルに、さまざまなバージョンを含めることは可能です。特定のアプリケーションのチェックアウト機能により、その機能のライセンスバージョンが決まります。

表 F-1: ライセンス ファイルのバージョンの見分け方

バージョン	説明
バージョン 2	空白の引用符、または引用文字列は FEATURE 行の最後にある。
バージョン 3 以降	INCREMENT 行または UPGRADE 行。
バージョン 4 以降	OVERDRAFT、DUP_GROUP、INTERNET または、PACKAGE が現れる。
バージョン 5 以降	SUPERSEDE、ISSUED、USER_BASED、HOST_BASED、または SN が現れる。
バージョン 6 以降	START が現れる。
バージョン 7.1 以降	SIGN= キーワードが現れる。
バージョン 8 以降	BORROW、FLOAT_OK、および TS_OK が現れる。

バージョンのまとめ

バージョン 1.0–1988 年

基本的な FLEXlm 機能のすべてを含む、最初の FLEXlm リリース

バージョン 1.5–1990 年 2 月

初めて広く使用された DEMO を含むバージョン

バージョン 2.1–1991 年 3 月

- TIMEOUT サポート改良
- イーサネット ホスト ID サポート改良

バージョン 2.21 – 1991 年 11 月

- 多くのプラットフォームのサポート、およびホスト ID など、いくつかのプラットフォーム固有の改良点を追加
- ホスト ID "ANY"の追加

バージョン 2.26 – 1992 年 3 月 (Sun 専用)

- ライセンス保持機能の追加

バージョン 2.4 – 1992 年 12 月

- 差分ライセンス配布用に、"use-all-feature-lines" 機能の追加
- ベンダー カスタマイズルーチンの強化
- エンドユーザ オプション ファイルの強化
- 新規ホスト ID タイプ USER、HOSTNAME、および DISPLAY の追加
- ライセンス ファイル検索のための `port@host` の追加 (サーバからライセンス ファイルをダウンロード)

バージョン 2.61 – 1993 年 3 月 (Sun 専用)

- ライセンス ファイルに INCREMENT 行、UPGRADE 行の追加

バージョン 3.0 – 1994 年 5 月

- INCREMENT 動作、UPGRADE 動作の変更と改良
- UDP プロトコル サポートを追加
- HP向けに `uname -i` ホスト ID を追加
- ライセンス ファイル リストとしての環境変数 `LM_LICENSE_FILE` のサポートを強化するために複数のジョブを追加
- オプションの新機能用の `keyword=value` 構文による新しいオプションのライセンス ファイル フォーマット。 `asset_info`、`ISSUER`、および `NOTICE`、"" ライセンス ファイルの行継続文字、1 機能あたり 2,048 文字の制限

バージョン 4.0 – 1994 年 12 月

- 信頼性向上のために、浮動小数点の使用取り止め
- FEATURE 行の属性追加: `ck`、`OVERDRAFT`、`DUP_GROUP`、`INTERNET` ホスト ID
- PACKAGE 行
- ライセンス ファインダ

- エンドユーザの診断用の lmdiag および FLEXLM_DIAGNOSTICS

バージョン 4.1 – 1995 年 5 月

- パフォーマンスの向上、新規プラットフォームのサポート

バージョン 4.1 – Patch Release 6、1995 年 10 月

- 各種のパフォーマンス向上をもたらす Windows 95 用の Windows パッチ リリース

バージョン 5.0 – 1996 年 3 月

- `port@host` 動作の改良 (FLEX 対応アプリケーションは、ライセンス ファイルを読み込まない)
- ライセンス ファイル内の `USE_SERVER` 行による自動 `port@host`
- ホスト ID リスト (機能を複数のホスト ID にロックする)
- 新規 FEATURE 属性▲ : SN (シリアル番号)、USER_BASED、HOST_BASED、MINIMUM、SUPERSEDE、ISSUED (発行日)、CAPACITY (システム性能を基に請求)
- ホスト名の代わりに IP アドレスを使用することで、NIS および DNS を迂回するオプション
- レポート ログ ファイル フォーマットの改良
- サーバは起動時に 2 週間以内に期限切れとなるライセンスを通知する
- エンドユーザ オプション ファイルの機能性の改良

バージョン 5.11 – 1997 年 2 月

- SUPERSEDE リスト、PLATFORMS=ライセンス属性
- 新規エンドユーザ オプション MAX、TIMEOUTALL
- Windows コントロールパネルの追加
- Windows ライセンス ジェネレータ GENLIC の追加

バージョン 5.12 – 1997 年 4 月

- パフォーマンスの向上、新規プラットフォームのサポート

バージョン 6.0 – 1997 年 9 月

- `lmgrd` は、複数のライセンス ファイルの読み込み可能
- `FLEXlm` ライセンス ディレクトリのサポート: *.lic を自動で使用
- エンドユーザ側での使用時、ライセンス ファイルの編集不要

- DAEMON/VENDOR 行でのオプションのパス; 環境変数 \$PATH の使用
- ライセンスの10進数フォーマットとライセンス入力用の `lminstall` ユーティリティ
- 短く、理解と入力が容易な FEATURE 行
- PACKAGE 行は、ユーザ編集をまったく必要としない別ファイルでの出荷が可能
- SERVER 行のポート番号をオプションとする既定値の TCP/IP ポート番号
- 既定値のエンドユーザ オプション ファイルパス
- SERVER 行上でサポートされる `this_host` ホスト名
- VENDOR_LICENSE_FILE をサポート (たとえば、MVS_N_LICENSE_FILE)
- 既定のポート番号を使用する `@host` をサポート
- Windows のみ: ユーザにライセンス ファイルまたはライセンス サーバ名を入力要求
- オプションにより、大文字、小文字を区別しないライセンス ファイル
- `lmdown` および `lmreread` の `-vendor vendor` 引数の受け入れ
- `START=dd-mm-yyyy` のオプション ライセンス属性

バージョン 6.1—1998 年 6 月

- パフォーマンスの向上

バージョン 7.0—1999 年 8 月

- 自動ライセンス発行の実行をサポートする License Certificate Manager (ライセンス 認証マネージャ)
- 「試用」ライセンスのサポート
- メールに挿入された改行に対応するライセンス ファイル
- ライセンス行の最適な自動ソート
- Windows 用 LMTTOOLS のインタフェースの改良
- Windows のコマンド ラインで実行時、既定値ではバックグラウンドで実行する `lmgrd`
- 3 サーバによる冗長構成の信頼性の改良 (バージョン 7.0 ベンダー デーモンおよび `lmgrd`)
- すべての `lmgrds` をシャット ダウンまたは再読み込みする `lmreread` および `lmdown` の `-all` 引数の受け入れ
- FLEXlm の環境変数用にレジストリ (Windows) および `$HOME/.flexlmrc` (UNIX) のサポート
- チェックアウト正常終了後のレジストリまたは `$HOME/.flexlmrc` への自動ライセンスパスの自動インストール

- PROJECT に伴う LM_PROJECT に対するオプションのサポート
- 特に Windows NT でのパフォーマンス向上
- Intel Pentium III CPU-ID (1999 年 11 月のバージョン 7.0d 以降)

バージョン 7.1—2000 年 8 月

- セキュリティの強化
- SIGN= キーワードの属性
- lmnewlog ユーティリティ (v7.1 以降のベンダー デーモン)

バージョン 7.2—2000 年 12 月

- パフォーマンスの向上

バージョン 8.0—2001 年 10 月

- lmborrow (バージョン 8.0 以降のコンポーネント)、lmpath (バージョン 8.0 以降のベンダー デーモン)、および lmswitch (バージョン 8.0 以降のベンダー デーモン) ユーティリティ
- エンドユーザのオプション ファイルと SERVER のホスト名を読み込む lmreread
- BORROW キーワードによるライセンス借用

バージョン 8.1—2002 年 1 月

- CRO セキュリティの強化

バージョン 8.2—2002 年 8 月

- Windows XP 対応のためのサポート追加

バージョン 8.3—2002 年 10 月

- 借用ライセンスの早期返却のサポート追加

バージョン 8.4—2003 年 1 月

- 製品パッケージの予約サポート

バージョン 9.0—2003 年 3 月

- COMPOSITE= ホスト ID タイプのサポート

バージョン 9.2 – 2003 年 1 月 ■ July?

- オプションファイルのキーワード GROUPCASEINSENSITIVE および MAX_BORROW_HOURS の追加

バージョン 9.5 – 2004 年 12 月

- 新しい環境変数: LM_UTIL_CASE_SENSITIVE

バージョン 10.0 – 2004 年 4 月

- FLEXnet Licensing としてリリース
- 完全修飾ドメイン名 (FQDN) のサポート

バージョン 10.1 – 2004 年 11 月

- USB ドングルに対応した追加の FLEXid ドライバのサポート

バージョン 10.8 – 2005 年 4 月

- ホスト ID に対応した IPv6 アドレスのサポート
- 強化された 3 サーバによる冗長構成のサーバ設定のサポート
- 共通ベンダー デーモンのサポート

バージョン 11.1 – 2005 年 11 月

- trusted storage でのライセンス権のサポート
- このリリースで復帰したホスト ID に対する IPv6 のサポート

索引

数字

10 進フォーマットのライセンス 53
3 サーバによる冗長構成 41

A

ANY ホスト ID 106
asset_info 119
AUTH 117, 123

B

BORROW_LOWWATER 70

C

COMPOSITE
 ホスト ID 106

D

DAEMON 行 115
DEBUGLOG 71
DEMO ホスト ID 106
DISPLAY
 タイプ 70
 ホスト ID 106
dist_info 120

DUP_GROUP 118

E

EXCLUDE 72
EXCLUDE_BORROW 72
EXCLUDEALL 74

F

FEATURE 行 116
 asset_info 119
 AUTH 117
 dist_info 120
 DUP_GROUP 118
 FLOAT_OK 118
 HOST_BASED 118
 HOSTID 118
 ISSUED 118
 ISSUER 118
 NOTICE 118
 OVERDRAFT 119
 PLATFORMS 119
 SIGN 117
 SN 119
 START 119
 SUPERSEDE 119
 TS_OK 119
 USER_BASED 119
 user_info 120

vendor_info 120
 VENDOR_STRING 119
 機能のバージョン 116
 構文 121
 署名 117
 シリアル番号 119
 ソート 120
 ソート順序 120
 ベンダー デーモン名 116
 有効期限 117
 優先順位 120
 ライセンス数 117
 Feature 行
 SUITE_DUP_GROUP 119
 FLEXid, FLOAT_OK 付きの 94
 FLEXLM_BATCH 142
 FLEXLM_DIAGNOSTICS 138
 レベル 1 139
 レベル 2 139
 レベル 3 140
 FLEXLM_TIMEOUT 142
 FLEXnet Manager 85
 FLOAT_OK 118

G

GROUP タイプ 77
 GROUPEASEINSENSITIVE 77

H

HOST タイプ 69
 HOST_BASED 118
 HOST_GROUP タイプ 78
 HOSTID 118
 HOSTNAME ホスト ID 106
 http
 //www.macrovision.com/services/education/
 index.shtml 15

I

ID ホスト ID 106
 INCLUDE 78
 INCLUDE_BORROW 79
 INCLUDEALL 81
 INCREMENT 行 116
 INTERNET
 タイプ 70
 ホスト ID 106
 ISSUED 118

ISSUER 118

L

LINGER 81
 LM_BORROW 142
 LM_LICENSE_FILE 143
 LM_PROJECT 143
 使用, オプション ファイルで 70
 報告, プロジェクトの 85
 LM_SERVER_HIGHEST_FD 143
 lmdiag
 構文 48
 トラブルシューティング 48
 lmdown
 構文 50
 制限, アクセスの 128
 無効にする 128
 lmgrd
 開始, デバッグ ログを 128
 概要 127
 起動 127, 129
 構文 127
 互換性, バージョン間の 127
 シャットダウン 50
 使用, 最新版を 167
 冗長サーバと 112, 129
 デバッグ ログ ファイル 161
 メモリの使用量 38
 lmhostid
 構文 51
 lmhostid, 構文 51
 lminstall
 構文 53
 ライセンス ファイル フォーマット 53
 lmnewlog, 構文 54
 lmremove
 構文 56
 制限, アクセスの 128
 無効にする 128
 lmread
 構文 57
 制限, アクセスの 128
 lmstat
 構文 59
 出力, lmread 用 59
 lmswitch, 構文 61
 lmswitchr, 構文 62
 LMTOOLS 63
 lmutil
 lmdiag 48
 lmdown 50

lmhostid 51
 lminstall 53
 lmnewlog 54
 lmremove 56
 lmreread 57
 lmstat 59
 lmswitch 61
 lmswitchr 62
 lmver 63
 lmver, 構文 63

M

Macrovision 13
 コンサルティング サービス 16
 サポートの Web サイト 17
 製品 13
 製品サポート 15
 テクニカル サポート 17
 トレーニング 15
 プロフェッショナル サービス 15
 連絡先 18
 連絡先情報 16
 MAX 82
 MAX_BORROW_HOURS 83
 MAX_OVERDRAFT 84

N

NOLOG 84
 NOTICE 118

O

OPTIONS=SUITE 123
 OPTIONS=SUITE_RESERVED 123
 OVERDRAFT 119

P

PACKAGE 行 121
 AUTH 123
 OPTIONS=SUITE 123
 OPTIONS=SUITE_RESERVED 123
 SIGN 123
 構文 122
 署名 123
 PLATFORMS 119
 PROJECT タイプ 70

R

REPORTLOG 85
 RESERVE 85

S

SERVER 行 110
 既定のポート番号 111
 結合, ライセンス ファイルを 35
 構文 110
 冗長構成のサーバ 110
 ホスト 111
 ホスト ID 111
 ポート番号 111
 SIGN 117, 123
 SN 119
 START 119
 SUITE_DUP_GROUP 119
 SUPERSEDE 119

T

TIMEOUT 86
 TIMEOUTALL 87
 TS_OK 119

U

UPGRADE 行, 構文 124
 USE_SERVER 行 115
 USER タイプ 69
 USER ホスト ID 107
 USER_BASED 119
 user_info 120

V

VENDOR 行 114
 オプション ファイルのパス 115
 ベンダー デーモンのパス 114
 ベンダー デーモン名 114
 ポート番号 115
 vendor.opt 66, 115
 vendor_info 120
 VENDOR_LICENSE_FILE 23, 143
 VENDOR_STRING 119

え

エラー コード

- 説明 147
- フォーマット 145

お

オプション ファイル

- BORROW_LOWWATER 70
- DEBUGLOG 71
- DISPLAY タイプ 70
- EXCLUDE 72
- EXCLUDE_BORROW 72
- EXCLUDEALL 74
- GROUP タイプ 77
- GROUPCASEINSENSITIVE 77
- HOST タイプ 69
- HOST_GROUP タイプ 78
- INCLUDE 78
- INCLUDE_BORROW 79
- INCLUDEALL 81
- INTERNET タイプ 70
- LINGER 81
- MAX 82
- MAX_BORROW_HOURS 83
- MAX_OVERDRAFT 84
- NOLOG 84
- PROJECT タイプ 70
- REPORTLOG 85
- RESERVE 85
- TIMEOUT 86
- TIMEOUTALL 87
- type 引数 69
- USER タイプ 69
- 規則, 優先順位の 88
- 作成 66
- 要件, HOST_BASED の 118
- 要件, USER_BASED の 119
- 読み込む, ベンダー デーモンにより 88
- 例 89
- オプション ファイルのパス 115
- オンライン コミュニティ
- ヘルプ 17

か

環境変数

- FLEXLM_BATCH 142
- FLEXLM_DIAGNOSTICS 142
- FLEXLM_TIMEOUT 142

- LM_BORROW 142
- LM_LICENSE_FILE 143
- LM_PROJECT 143
- LM_SERVER_HIGHEST_FD 143
- VENDOR_LICENSE_FILE 143
- 設定 141

き

起動, Imgrd を 129

機能

- バージョン 116
- 切り替え, デバッグ ログを
lmswitch 61
- 切り替え, レポート ログを
lmnewlog 54
- lmswitchr 62

こ

混在ライセンス 26

さ

作成, オプション ファイルを 66

し

指定, ライセンス ファイルの場所を 21

借用 97

署名 117, 123

診断, チェックアウトの問題を
トラブルシューティング
チェックアウト 48

順序, ライセンス ファイルの行の 26, 125

冗長構成のサーバ

3 サーバによる冗長構成 41

SERVER 行 110

選択, サーバ ノードを 39

別々のライセンス ファイル 129

ライセンス ファイル リストによる 40

す

ステータス, ライセンス サーバの 59

せ

制限, アクセスの

lmdown 128
 lmremove 128
 lmreread 128
 製品サポート
 Macrovision 15
 設定, 環境変数を 141

そ

ソート 120
 ソケット
 使用数, ライセンス サーバによる 37

て

テクニカル サポート
 Macrovision 17
 デバッグ, ライセンス サーバ システムの 137

と

トラブルシューティング
 FLEXLM_DIAGNOSTICS 138
 lmdiag 48
 トレーニング
 Macrovision 15
 同時使用ライセンス 25

な

ナレッジ ベースの項目
 ヘルプ 17

ね

ネットワーク帯域幅と FLEXnet Licensing 39
 ネットワーク ライセンス 25

の

ノードロックされたライセンス 25

は

パッケージ式 123

ふ

フローティング ライセンス 25
 プロフェッショナル サービス
 Macrovision 15

へ

ヘルプ
 オンライン コミュニティ 17
 サポートの Web サイト 17
 ナレッジ ベースの項目 17
 ヘルプ ライブラリの表記規約 11
 変換, ライセンス形式を 53
 ベンダー デーモン
 lmnewlog 54
 lmreread 57
 lmswitchr 62
 VENDOR_LICENSE_FILE 143
 オプション ファイル 66
 カウントされないライセンス 42
 冗長サーバと 112, 129
 デバッグ ログ ファイル 161
 バージョンの互換性 127
 メモリの使用量 38
 ベンダー デーモンのパス 114
 ベンダー デーモン名
 FEATURE 行 116
 VENDOR 行 114

ほ

報告, プロジェクトの 85
 ホスト ID
 ANY 106
 COMPOSITE 106
 DEMO 106
 DISPLAY 106
 HOSTNAME 106
 ID 106
 INTERNET 106
 SERVER 行 111
 USER 107
 特殊 106
 ホスト, SERVER 行 111
 ポート番号
 SERVER 行 111
 VENDOR 行 115
 サーバの既定の範囲 111

む

無効にする

lmdown 128

lmremove 128

め

メモリの使用法, デーモン 38

も

モバイル ライセンス

FLEXid, FLOAT_OK 付きの 94

借用 97

ノードロック, FLEXid に対する 94

ノードロック, ユーザ名に対する 102

ノードロック, ラップトップに対する 94

プリペイド ライセンス プールからのライセンス発行 102

ゆ

有効期限 117

有効にする, レポート ログを 85

優先順位, FEATURE 行の 120

ら

ライセンス

混在 26

借用 97

同時使用 25

ネットワーク ライセンス 25

ノードロック 25

フローティング 25

ライセンス サーバ

インストール, Windows サービスとして 64

決定, ノードの数 39

ソケット, 使用される 37

ディスク スペース, 使用される 38

デバッグ 137

ライセンス サーバ マネージャ 127

ライセンス数 117

ライセンス ディレクトリ 129, 131

ライセンス ファイル

10 進数のフォーマット 125

DAEMON 行 115

FEATURE 行 116

INCREMENT 行 116

lminstall 53

PACKAGE 行 121

SERVER 行 35

UPGRADE 行 124

USE_SERVER 行 115

VENDOR 行 114

結合方法 35

互換性, 異なるバージョン間の 35

再読み込み, 更新後の 57

指定, 場所を 21

種類 24

順序, 行の 26, 125

フォーマット 24

複数のサーバ 112, 129

ライセンス ファイル リスト 32

ライセンス ファイル リストによる冗長構成 40

ライセンス プール 68, 116

ライセンス リホスティング 93

り

リホスティング, ライセンス 93

リモート ディスク, 使用の指針 39

れ

レポート ログ ファイル 38

連絡先情報

Macrovision 16