

IBM Communications Server for Linux



入門

バージョン 6.2.2

IBM Communications Server for Linux



入門

バージョン 6.2.2

ご注意

本書および本書で紹介する製品をご使用になる前に、121 ページの『特記事項』に記載されている情報をお読みください。

本書は、IBM Communications Server for Linux バージョン 6.2.2、および新しい版またはテクニカル・ニュースレターで明記されていない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。

本マニュアルに関するご意見やご感想は、次の URL からお送りください。今後の参考にさせていただきます。

<http://www.ibm.com/jp/manuals/main/mail.html>

なお、日本 IBM 発行のマニュアルはインターネット経由でもご購入いただけます。詳しくは

<http://www.ibm.com/jp/manuals/> の「ご注文について」をご覧ください。

(URL は、変更になる場合があります)

お客様の環境によっては、資料中の円記号がバックスラッシュと表示されたり、バックスラッシュが円記号と表示されたりする場合があります。

原 典： GC31-6768-01
IBM Communications Server for Linux
Quick Beginnings
Version 6.2.2

発 行： 日本アイ・ビー・エム株式会社

担 当： ナショナル・ランゲージ・サポート

第1刷 2006.5

この文書では、平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、および平成角ゴシック体™W7を使用しています。この(書体*)は、(財)日本規格協会と使用契約を締結し使用しているものです。フォントとして無断複製することは禁止されています。

注* 平成明朝体™W3、平成明朝体™W7、平成明朝体™W9、平成角ゴシック体™W3、平成角ゴシック体™W5、平成角ゴシック体™W7

© Copyright International Business Machines Corporation 1998, 2006. All rights reserved.

© Copyright IBM Japan 2006

目次

表	vii
図	ix
IBM Communications Server for Linux へようこそ	xi
本書の使い方	xi
参照リスト	xi
表記上の規則	xii
本書で使用される省略名	xiii
本リリースでの新機能	xiii
新しい機能	xiv
廃止された機能	xv
第 1 章 IBM Communications Server for Linux について	1
IBM Communications Server for Linux の機能とパッケージ化	1
IBM Communications Server for Linux	1
拡張ネットワーキング機能	6
機能および利点	12
多用途の構成要素	12
クライアント/サーバー・オペレーション	13
容易な構成	13
追加の管理用ユーザー・インターフェースの選択	14
パフォーマンスの改善	14
セキュリティー・オプション	15
ネットワーク管理の柔軟性	15
信頼性、可用性、および保守容易性	16
ネットワークの統合、成長および変更	16
第 2 章 ネットワークおよび IBM Communications Server for Linux の計画	19
ネットワーク計画の段階	19
ネットワークの機能要件の確認	19
Communications Server for Linux の構成方法の決定	20
インストールおよび操作のリソース要件の確認	20
クロスプラットフォーム構成における互換性の保証	26
命名規則	26
第 3 章 Linux サーバーでの Communications Server for Linux のインストール	29
Communications Server for Linux ライセンス・プログラムのパッケージ内容	29
Communications Server for Linux インストールの準備	30
パワー・マネージメント機能の使用不可化	30
前提ソフトウェアのインストール	30
製品インストール詳細の表示	31
言語環境変数の変更	31
旧レベルの Communications Server for Linux からのマイグレーション	31
Communications Server for Linux ライセンス・プログラムのインストール	34
Communications Server for Linux のインストール	35
Communications Server for Linux のオンライン資料	35
Host Access Class Libraries (HACL)	36
WebSphere Application Server の構成	36
WebSphere Application Server のセキュア証明書のセットアップ	36

WebSphere Application Server の構成	36
サーバー構成ファイルのインストール	37
インストール後の作業	38
クライアント/サーバー・オペレーション	38
PDF 資料の表示	38
現行リリース情報の検討	38
TN サーバーまたは TN リダイレクターで使用するための SSL の構成	39
Communications Server for Linux 構成ファイルのバックアップ	39
Communications Server for Linux 構成ファイルのバックアップ・コピーの復元	40
構成ファイルの再初期化	41
Communications Server for Linux のアンインストール	42
第 4 章 Linux での IBM Remote API Client のインストール	43
ハードウェア要件とソフトウェア要件	43
ハードウェア要件	43
Linux オペレーティング・システムのバージョン	44
Java	44
GSKIT	44
製品インストール詳細の表示	44
言語環境変数の設定	45
Linux での Remote API Client のインストール	45
GSKIT を使用した HTTPS セキュリティー証明書のセットアップ	46
Linux 上の Remote API Client のアンインストール	48
第 5 章 Linux for System z への IBM Remote API Clients のインストール	49
ハードウェア要件とソフトウェア要件	49
ハードウェア要件	49
Linux オペレーティング・システムのバージョン	49
Java	50
GSKIT	50
製品インストール詳細の表示	50
言語環境変数の設定	50
Linux for System z への Remote API Client のインストール	51
GSKIT を使用した HTTPS セキュリティー証明書のセットアップ	52
Linux for System z での Remote API Client のアンインストール	54
第 6 章 AIX システムでの IBM Remote API Client のインストール	55
ハードウェア要件とソフトウェア要件	55
ハードウェア要件	55
オペレーティング・システムのバージョン	55
Java	55
GSKIT	55
言語環境変数の変更	56
AIX での Remote API Client のインストール	56
AIX ワークステーションへのファイルのコピーによる Remote API Client のインストール	56
CD からの Remote API Client のインストール	57
GSKIT を使用した HTTPS セキュリティー証明書のセットアップ	58
AIX 上の Remote API Client のアンインストール	59
第 7 章 Windows 上の Remote API Client の計画とインストール	61
ハードウェア要件とソフトウェア要件	61
Setup プログラムへのアクセス	62
Setup プログラムを使用した Windows での Remote API Client のインストール	63
Remote API Client 構成の拡張オプション	66
コマンド行からの Remote API Client ソフトウェアのインストール	67
GSKIT ソフトウェアのインストールおよびセキュリティー証明書のセットアップ	70

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ	71
インストール後の Remote API Client ソフトウェアのカスタマイズ	72
Remote API Client ソフトウェアの再インストール	72
Remote API Client ソフトウェアのアンインストール	73
GSKIT ソフトウェアのアンインストール	73
ヘルプ	74
第 8 章 Communications Server for Linux の構成と使用	75
Communications Server for Linux 構成の計画	76
計画ワークシート	76
作業シート	77
Motif 管理プログラムの使用	77
Communications Server for Linux プログラムのパスの指定	77
Communications Server for Linux の使用可能化	78
Motif 管理プログラムによる Communications Server for Linux の管理	78
クライアント/サーバー機能の構成	83
ノードの構成	84
接続の構成	87
従属トラフィック用の SDLC リンクの構成	87
従属トラフィックと独立トラフィックをサポートするイーサネット・リンクの構成	89
エンタープライズ・エクステンダー・リンクの構成	90
タイプ 0 から 3 LU の構成	91
タイプ 0 から 3 LU の定義	92
LU プールの定義	92
APPC 通信の構成	93
単純な APPN ネットワークの構成	95
従属 APPC の構成	100
CPI 通信のための構成	101
LUA の構成	101
LU プールの定義	102
SNA ゲートウェイの構成	103
暗黙のダウンストリーム LU のサポート	105
ダウンストリーム LU の定義	106
DLUR の構成	106
ローカル・ノードでの DLUR サポートの構成	108
ダウンストリーム・ノードをサポートするパススルー DLUR の構成	109
TN サーバーの構成	109
3270 LU の定義	111
LU プールの定義	112
TN3270 サーバーの構成	113
TN リダイレクターの構成	115
TN リダイレクターの構成	115
Communications Server for Linux の使用不可化	116
Communications Server for Linux の自動開始	117
Communications Server for Linux の使用可能化	117
SNA ノードの初期化	117
ポートおよびリンク・ステーションの活動化	117
リポート時の Communications Server for Linux の開始	118
第 9 章 Communications Server for Linux と SNA の情報リソース	119
SNA ライブラリー	119
ネットワークで入手可能な情報	119
推薦資料	120
付録. 特記事項	121
商標	123

参考文献	125
Communications Server for Linux バージョン 6.2.2 資料	125
システム・ネットワーク体系 (SNA) 関連資料	126
ホスト構成関連資料	127
z/OS Communications Server 関連資料	127
TCP/IP 関連資料	127
X.25 関連資料	127
APPC 関連資料	128
プログラミング関連資料	128
その他の IBM ネットワーキング関連資料	128
索引	129

表

1. 本書内容の参照リスト	xii
2. 表記上の規則	xii



1. 複数のダウンストリーム Linux コンピューターをホスト・コンピューターにリンクする SNA ゲートウェイ	7
2. 分岐エクステンダー	9
3. TN サーバー	11
4. 「Node (ノード)」ウィンドウ	80
5. Communications Server for Linux ツールバー	83
6. ホストと直接通信する Communications Server for Linux ノード	85
7. APPN ネットワークの Communications Server for Linux ノード	86
8. SNA ゲートウェイ	104
9. DLUR を提供する Communications Server for Linux ノード	107
10. TN サーバー用に構成された Communications Server for Linux ノード	110
11. TN サーバー	111

IBM Communications Server for Linux へようこそ

本書では、IBM® Communications Server for Linux® について紹介します。この IBM ソフトウェア製品は、Linux を実行するコンピューターがシステム・ネットワーク体系 (SNA) ネットワークの他のノードと情報を交換できるようにするものです。

IBM Communications Server for Linux には、作動するためのハードウェアに応じた 2 つ異なるインストール変種があります。

Communications Server for Linux

Communications Server for Linux、プログラム製品番号 5724-i33 は、次のハードウェア上で作動します。

- Linux が稼働する 32 ビット Intel® ワークステーション (i686)
- Linux が稼働する 64 ビット AMD64/Intel EM64T ワークステーション (x86_64)
- Linux が稼働する IBM pSeries® コンピューター (ppc64)

Communications Server for Linux on System z

Communications Server for Linux on System z (プログラム製品番号 5724-i34) は、Linux for System z が稼働する System z メインフレーム (s390 または s390x) で作動します。

「*Communications Server for Linux 入門*」資料には、異なる 2 種類の版が用意されています (上記 2 つのインストール変形のそれぞれについて 1 つずつ)。お手元の資料が、ご使用の Communications Server for Linux インストール用の正しいコピーであることを確認してください。本書は、i686、x86_64、および ppc64 プラットフォーム用の Communications Server for Linux に適用されます。

Communications Server for Linux は、多種多様なネットワークのニーズとソリューションのためのビルディング・ブロックを提供します。これらを使用すると、SNA ネットワークのノードと情報を交換することもできるし、伝送制御プロトコル/インターネット・プロトコル (TCP/IP) を介して通信する Telnet プログラムへのホスト・アクセスを行うこともできます。

本書の使い方

このセクションでは、本書の編成および表記方法について説明します。

参照リスト

本書は、ネットワーク計画に関する管理および技術担当者、および Communications Server for the Linux オペレーティング・システムに関心を持つ方々を対象としています。

Communications Server for Linux を初めてご使用になる際に必要な情報は、xii ページの表 1 を参照してください。

表 1. 本書内容の参照リスト

内容	参照先
Communications Server for Linux について読む	1 ページの『第 1 章 IBM Communications Server for Linux について』
ネットワークで Communications Server for Linux を使用する 方法を計画する	19 ページの『第 2 章 ネットワークおよび IBM Communications Server for Linux の計画』
Linux サーバーでの Communications Server for Linux のインストール	29 ページの『第 3 章 Linux サーバーでの Communications Server for Linux のインストール』
Linux (32 ビット Intel、64 ビット Intel/AMD、または pSeries) に Remote API Client をインストールする	43 ページの『第 4 章 Linux での IBM Remote API Client のインストール』
Linux for System z に Remote API Client をインストールする	49 ページの『第 5 章 Linux for System z への IBM Remote API Clients のインストール』
AIX® での Remote API Client のインストール	55 ページの『第 6 章 AIX システムでの IBM Remote API Client のインストール』
Windows® での Remote API Client のインストール	61 ページの『第 7 章 Windows 上の Remote API Client の計画とインストール』
Communications Server for Linux の構成	75 ページの『第 8 章 Communications Server for Linux の構成と使用』
Communications Server for Linux の資料と他の資料に関する 情報 (オンライン情報を含む) を検索する	119 ページの『第 9 章 Communications Server for Linux と SNA の情報リソース』
特記事項および商標情報について	121 ページの『特記事項』

表記上の規則

表 2 に、本書で使用されている表記上の規則を示します。

表 2. 表記上の規則

内容	表記例
強調	削除する前にファイルのバックアップをとります。
資料名	「 <i>Communications Server for Linux</i> 管理ガイド」
ファイル名およびパス名	<code>/usr/spool/uucp/myfile.bkp</code>
プログラムまたはアプリケーション	<code>snaadmin</code>
ユーザー入力	0p1
コンピューターからの出力	CLOSE

本書で使用される省略名

本書で使用されている省略名を以下に示します。

AIX	拡張対話式エグゼクティブ
API	アプリケーション・プログラミング・インターフェース
APPC	拡張プログラム間通信機能
APPN	拡張対等通信ネットワーク機能
BrNN	分岐ネットワーク・ノード
COS	サービス・クラス
CPI-C	共通プログラミング・インターフェース・コミュニケーション
CSV	Common Service Verb
DDDLU	従属 LU の動的定義
DES	データ暗号化規格
DLC	データ・リンク制御
DLUR	従属 LU リクエスト
DLUS	従属 LU サーバー
FTP	ファイル転送プロトコル
HPR	高性能ルーティング
IETF	Internet Engineering Task Force
ISO	国際標準化機構
ISR	中間セッション・ルーティング
LAN	ローカル・エリア・ネットワーク
LDAP	Lightweight Directory Access Protocol
LEN	ローエントリー・ネットワーク
LLC2	論理リンク制御 2
LU	論理装置
LUA	従来型 LU アプリケーション・プログラミング・インターフェース
MDS-NMVT	マルチドメイン・サポート・ネットワーク管理ベクトル・トランスポート
MPC	マルチパス・チャネル
MS	管理サービス
NMVT	ネットワーク管理ベクトル・トランスポート
NOF	ノード・オペレーター・ファシリティー
OSI	オープン・システム相互接続
PU	物理装置
RFC	Request For Comments
RLE	ラン・レンダス・エンコード
SAA	システム・アプリケーション体系
SAP	サービス・アクセス・ポイント
SNA	システム・ネットワーク体系
SSL	Secure Sockets Layer
TCP/IP	TCP/IP プロトコル
TN	Telnet
TP	トランザクション・プログラム
VT	仮想端末
WAN	広域ネットワーク

本リリースでの新機能

Communications Server for Linux バージョン 6.2.2 は、Communications Server for Linux バージョン 6.2 および Communications Server for Linux バージョン 6.2.1 に置き換わります。

本リリースでの新機能

本製品のリリースのうち、現在でもサポートされているリリースは次のとおりです。

- Communications Server for Linux バージョン 6.2
- Communications Server for Linux バージョン 6.2.1

本製品の以下のリリースは、現在ではサポートされていません。

- Communications Server for Linux バージョン 6.0.1 (V6.0.1) は、PRPQ 5799-RQA として提供されていました。

新しい機能

このリリースで Communications Server for Linux に追加された機能は、以下のとおりです。

- クライアント/サーバー・サポートに、より広い範囲のサーバー・コンピュータおよびクライアント・コンピュータが包含されました。
 - 使用できるサーバーは次のいずれかです。
 - Linux が稼働する 32 ビット Intel ワークステーション (i686)
 - Linux が稼働する 64 ビット AMD64/Intel EM64T ワークステーション (x86_64)
 - Linux が稼働する IBM pSeries コンピューター (ppc64)
 - クライアントは、i686、x86_64、ppc64、s390、または s390x システム、AIX ワークステーション、または 32 ビット・バージョンの Microsoft® Windows (MicrosoftWindows2000、2003、or XP) または x64 バージョンの Microsoft Windows (Microsoft WindowsServer 2003 x64 Edition または Microsoft WindowsXP Professional x64 Edition) が稼働する PC です。これらのクライアントは、TCP/IP を使用して、あるいは WebSphere® サーバー経由で HTTPS を使用して、Communications Server for Linux サーバー (または CS/AIX サーバー) と通信することができます。
 - Linux、Linux for System z、および AIX クライアントは、Linux または Linux for System z サーバーで使用可能なすべての API ライブラリーをサポートします。(早期の CS/AIX v4.2 製品に組み込まれていた API は、このバージョンではサポートされません。)
 - Windows クライアント (32 ビット・バージョンまたは x64 バージョンの Microsoft Windows 上のもの) は、APPC、CPI-C、CSV、LUA (LUA RUI と SLI の両方を含む)、および NOF API をサポートします。
 - NOF API 用のクライアント・サポートは (すべてのクライアント・タイプにおいて) 照会機能のみを提供します。リソースを定義、削除、開始、または停止するための呼び出しは提供されません。
 - Windows クライアントは、Windows Terminal Server (WTS) 環境で稼働できます。この環境では、複数のユーザーが同一クライアントを共用できますが、それぞれのアプリケーションは独立に稼働し、互いに干渉し合うことはありません。
- SLES10 Linux ディストリビューションが (サーバーとクライアントの両方について) サポートされるようになりました。
- Communications Server for Linux は、Linux サーバー上で WAN アダプターを使用する SDLC および X.25 (QLLC) 接続をサポートするようになりました。この

接続には、アダプター・ベンダーのデバイス・ドライバーおよび DLC コードが必要です。IBM はこれらのアダプターまたはデバイス・ドライバーをサポートしません。適切な WAN アダプター・ベンダーについては、Communications Server for Linux のサポート Web ページを参照してください。

- TN サーバーおよび TN リダイレクターには、TN パススルー・サービスがクライアント接続を listen する特定のローカル・アドレスを指定する機能が含められました。このフィーチャーはオプションです。ユーザーは任意のローカル・アドレスで TN3270 クライアント接続をサポートするようにサービスを構成することができ、また特定のアドレスにサービスを制限することもできます。

廃止された機能

クライアントおよびサーバー・コードは、今後 RedHat Advanced Server 2.1 ではサポートされません。

第 1 章 IBM Communications Server for Linux について

この章では、Communications Server for Linux のパッケージ化の方法、ならびに機能、フィーチャー、および利点について説明します。

IBM Communications Server for Linux の機能とパッケージ化

Communications Server for Linux は、Linux オペレーティング・システムで稼働する通信用ソフトウェアです。これは、『IBM Communications Server for Linux』および 6 ページの『拡張ネットワーキング機能』で説明されている機能から構成されます。

IBM Communications Server for Linux

IBM Communications Server for Linux は、SNA ネットワークおよび TCP/IP ネットワークを通してアプリケーションを接続します。Linux が稼働しているワークステーションに SNA リソースおよびプロトコルを備えることによって、そのワークステーションを SNA ノードに変換します。これにより、そのワークステーションは、SNA ネットワーク内のその他のワークステーションやホスト・コンピュータと通信できるようになります。TCP/IP 機能も提供するので、TCP/IP ネットワーク内において、あるいは TCP/IP ネットワークと SNA ネットワークの境界においても IBM Communications Server for Linux を使用できます。

Communications Server for Linux では、以下のサービスが提供されます。

ネットワーク・サポート

Communications Server for Linux は、サブエリア・ネットワークおよび対等ネットワークをサポートします。

SNA サブエリア・ネットワーク

これらのネットワーク (ホスト仲介ネットワークともいう) は階層編成になっており、1 つ以上のホスト・コンピュータがコンピュータ間の通信を制御し、ネットワークを管理し、処理サービスと大容量データ・ストレージを提供します。ネットワーク内のその他のすべてのノードは、ホストの制御に依存します。

Linux コンピューターは、ホスト従属ノードとして構成すれば、サブエリア・ネットワークに組み入れることができます。

対等ネットワーク

分散処理環境の場合、Communications Server for Linux は APPN ネットワークをサポートします。これらの対等ネットワーク内で、Linux コンピューターは処理機能を保持し、互いに対等に直接通信します。対等ネットワークは、現在では高価なホスト・コンピュータの能力にも匹敵する Linux コンピューターの能力を完全に使用しています。

APPN ネットワークは、以下のタイプの対等ノードで構成されます。

- APPN ネットワーク・ノード (トラフィック制御、動的経路計算および選択サービス、ならびにネットワーク管理サービスを提供します)
- APPN エンド・ノード (APPN ネットワーク・ノード・サービスを使用して、対等ノードと通信します)
- LEN ノード (隣接ノードまたは隣接に見えるように構成されているノードと直接に通信します)

注: ホスト・コンピューターは、独立 LU 6.2 を使用して APPN ネットワーク内の Linux コンピューターおよび他のホストと通信することによって、ネットワーク内の対等ノードとして機能することができます。

APPN ネットワーク内でのサブエリア機能の提供

従属 LU リクエスター (DLUR) 機能により、ホストとホスト従属ノードの間のトラフィックを、APPN ネットワーク内で受け渡すことができます。

データ・リンク制御オプション

リンク・レベルで、Communications Server for Linux によりさまざまな接続オプションが提供され、ネットワークのサイズ、速度、セキュリティ、およびコストに関する要件を満たすのに役立ちます。(サポートされるリンク・タイプの詳細リストについては、22 ページの『インストール要件』を参照してください。) また、以下のように、さまざまな異なるネットワーク・タイプ用のデータ・リンクをサポートします。

ローカル・エリア・ネットワーク

LAN 接続の場合、トークンリング、標準イーサネット、および 802.3 イーサネット・プロトコルを使用して通信する適切なリンクをインストールできます。

広域ネットワーク

Communications Server for Linux は SDLC および X.25 (QLLC) 接続をサポートします。このサポートは各プラットフォームの OEM アダプター・サポートに依存します。

ローカル接続

Communications Server for Linux はローカル接続用のマルチパス・チャンネル (MPC) 接続をサポートします (Communications Server for Linux on System z のみ)。

IP 統合

企業のバックボーン・ネットワークが IP ベースの場合、Communications Server for Linux の Enterprise Extender (HPR/IP) 機能を使用して、これを SNA に統合するので、SNA アプリケーションが IP ネットワークを通して通信できるようになります。

LU サポート

論理装置 (LU) は、アプリケーション固有のネットワーク・リソースであり、SNA ネットワーク内の各ノードにあります。各 LU は、ネットワーク

IBM Communications Server for Linux の機能とパッケージ化

上の他のノード上のパートナー・アプリケーションと通信するために、アプリケーションがリンクにアクセスする際に使用するインターフェースとして機能します。

Communications Server for Linux は、アプリケーションの異なるクラスごとに、異なるタイプの LU をサポートします。

- サブエリア・ネットワークでは、Communications Server for Linux は、次のタイプのいずれかの従属 LU をサポートしています。
 - LU 0
 - LU 1
 - LU 2
 - LU 3
 - LU 6.2

LU 0 は、基本的なプログラム間の通信をサポートします。これは、通常、小売業や銀行の販売時点トランザクションで使用します。LU 2 は、Linux コンピューターが IBM 3270 ファミリー端末をエミュレートできるように、端末エミュレーション・アプリケーションをサポートします。その他の LU タイプは、アプリケーションが分散処理に参加したり、各種プリンターや対話式ディスプレイ端末と通信したりできるようにします。

Communications Server for Linux は、従属 LU の動的定義 (DDDLU) を使用するホスト・システムをサポートします。DDDLU は、SNA システムからホストへの通信リンクが確立されたとき、SNA システムの従属 LU をホスト構成に追加できるホスト機能です。DDDLU を使用すると、LU をホストに静的に構成する必要はありません。(しかし、Communications Server for Linux ノードで従属 LU を定義していなければなりません。) これにより、ホストに必要な初期構成が少なくなり、後の拡張が簡単になります。

Communications Server for Linux は、構成を変えなくても、DDDLU 可能なホストと DDDLU 可能でないホストの両方と通信できます。

Communications Server for Linux ノードからホストへの通信リンクが確立されると、DDDLU 可能なホストは、DDDLU をサポートしていることをノードに通知します。ノードは、次に、リンクを使用する従属 LU を定義するために必要な情報を送信します。ホストが DDDLU 可能でない場合、Communications Server for Linux はこの情報を送信しません。これは、LU がホストで既に静的に定義されていることを前提としています。

- 独立 LU 6.2 は、APPN ネットワーク内の独立トラフィックをサポートします。独立 LU 6.2 は、自立走行式通信およびネットワーク管理ならびに分散処理をサポートします。

さらに、Communications Server for Linux の DLUR 機能により、従属 LU のトラフィックが APPN ネットワーク上を移動できるようになります。

- 1 次 RUI サポートは、Communications Server for Linux アプリケーションが、メインフレームであるかのように、ダウンストリームの

LAN/WAN 接続された従属 LU デバイスを管理できる機能を提供します。この機能には接続性に関していくつかの制約事項がありますが、この機能によって、アプリケーションは完全なメインフレーム・アプリケーションを必要とせずに、従属 LU デバイス同士の間でデータを渡すことができます。

セッション・サポート

セッションは、パートナー LU 間の一時的な論理チャネルです。通常、各 LU に関連するパートナー・アプリケーションは、セッションを介して通信を行います。Communications Server for Linux は、何千ものセッションをサポートできます。Communications Server for Linux は、U 型セッション（「『ローカル/リモート透過型』」ともいいます）もサポートします。同じ Linux コンピューターに、1 次 LU と 2 次 LU の両方があります。これにより、リンク接続を必要とせずに、1 台のコンピューターでソースとターゲットのトランザクション・プログラムのペアの開発やテストを行うことができます。

セッションの 2 つのパートナー LU 間を流れるデータは、必要な帯域幅を削減するために圧縮されることがあります。

- LU タイプ 6.2 の場合、Communications Server for Linux では、セッションで使用するモードの構成で圧縮を使用するように指定できます。使用される圧縮アルゴリズムとして、圧縮のレベルが異なるさまざまな方法を指定できます (RLE、LZ9、または LZ10)。セッションでデータが流れる方向によって異なる圧縮レベルを指定したり、一方方向にだけ圧縮を指定して別の方向には指定しないことも可能です。
- LU タイプ 0 - 3 の場合、Communications Server for Linux では、セッションで使用するリンク・ステーションまたは PU の構成で圧縮を使用するように指定できます。インバウンド方向には RLE 圧縮が使用され、アウトバウンド方向には LZ9 圧縮が使用されます。

API サポート

Communications Server for Linux には、アプリケーション・プログラミング・インターフェース (API) があり、特定のタイプの LU、分散処理、ネットワーク管理、および Communications Server for Linux 自身を管理するアプリケーション開発に用いられます。Communications Server for Linux は、その他のオペレーティング・システム上で稼働する Communications Server ファミリーのメンバーが提供する API と互換性のある一連の API を提供します。

API は、トランザクション・プログラム (TP) が、それがサポートする LU と通信できるようにするインターフェースです。API は、verb (関数、コール、サブルーチンともいう) のライブラリーから構成されています。ここから、TP は、SEND_DATA などのアクションを要求するために LU に渡す必要のあるものを選択します。LU は、verb を処理し、適切なプロトコルに従ってデータ・ストリームを作成し、宛先アドレスを示すヘッダーを付加し、リンクを通してパートナー LU へこのデータを送信します。

共通プログラミング・インターフェース・コミュニケーション (CPI-C) は、移植性があるため、最強の API の 1 つになっています。従属型および独立型 LU 6.2 をサポートするために導入された CPI-C は、システム・アプリケーション体系® (SAA®) に準拠して、異なるプラットフォームおよび

IBM Communications Server for Linux の機能とパッケージ化

オペレーティング・システムを統合します。CPI-C は、すべてのシステムに共通の構文規則のセットを使用します。これが、CPI-C が標準になった理由です。

Communications Server for Linux には、標準 C 言語 CPI-C API の他に Java™ アプリケーション用の CPI-C API もあります。詳しくは、「*Communications Server for Linux CPI-C プログラマーズ・ガイド*」を参照してください。Communications Server for Linux 資料では、特に断りのない限り、CPI-C の参照箇所すべてに Java CPI-C が含まれています。

その他の Communications Server for Linux API には、以下のものが含まれます。

- APPC API。LU 6.2 を使用するアプリケーション・プログラム間の対等通信用です。この API には、非ブロッキングにするオプションがあります。TP が非ブロッキング verb を使用すると、要求されたアクションが完了する前に、この API は TP に制御を戻すことができます。後にアクションが完了した時点で、TP はその通知を受け取ります。
- LUA API。ホスト・アプリケーションとの通信用です。
- CSV (共通サービス Verb) API。文字変換やアプリケーション・トレース制御などのユーティリティー関数用です。

さらに、Communications Server for Linux には、所有権のある以下のプログラミング・インターフェースがあります。

- MS (管理サービス) API。ネットワーク・メッセージ処理機能用です。
- NOF (ノード・オペレーター機能) API。アプリケーションによって Communications Server for Linux リソースを構成および管理します。

API についての詳細は、API のプログラミング・ガイドを参照してください。

クライアント/サーバーのサポート

Communications Server for Linux を実行するコンピューターは、クライアント/サーバー・プロトコルを使用して通信するように構成できます。クライアント/サーバー・プロトコルが使用されているネットワークにおいて、クライアント/サーバー・プロトコルを通信に使用するすべてのコンピューターを 1 つの「ドメイン」と呼びます。

クライアント/サーバー構成で Communications Server for Linux を実行するコンピューターは、以下のロールを持つことができます。

- サーバーは、SNA ノードと、それに関連した接続コンポーネントを含みます。サーバーは、ローカル・システムのアプリケーション、または Communications Server for Linux ドメイン内の他のコンピューターのアプリケーションへの SNA 接続を行います。サーバーは Linux システムでなければなりません。
- Remote API クライアントは、SNA ノード・コンポーネントは含みませんが、サーバーを通してそれらのコンポーネントにアクセスします。1 つのクライアントは同時に 1 つ以上のサーバーにアクセスでき、必要に応じて同時アプリケーションを実行できます。クライアントでは、AIX、Linux、または Windows のどれが稼働していてもかまいません。(1 台の Linux コンピューターは、サーバーとクライアントのどちらにもで

きますが、両方にはできません。つまり、同じコンピューターにサーバーとクライアントの両方をインストールすることはできません。)

サーバーとクライアントは、TCP/IP を使用して Communications Server for Linux ドメインと通信します。あるいは、サーバーとクライアントは、WebSphere サーバー経由で HTTPS を使用して通信することができます。この場合、セキュリティー証明書を使用してクライアント接続が認証されます。通常、クライアントが公衆ネットワーク経由で接続している場合は、HTTPS を使用します。

複数の Communications Server for Linux サーバーを持つドメインでは、1つのサーバーが Communications Server for Linux ドメイン構成ファイルのマスター・コピーを保持しています。このサーバーをマスター・サーバーと呼びます。ドメイン内の他のサーバーは、バックアップ・サーバーと定義するか、そのままピア・サーバーにしておくことができます。ドメイン構成ファイルは、すべてのバックアップ・サーバーが最新情報のコピーを保持できるよう、バックアップ・サーバーが開始されたときか、マスター・コピーが変更されたときのいずれかの時点で、バックアップ・サーバーにコピーされます。ピア・サーバーは、必要に応じてドメイン構成情報をマスター・サーバーから入手しますが、バックアップ・サーバーのように動作することはできません。

マスター・サーバーに障害が起こると、そのドメインに対して定義されたサーバー・リストの先頭にあるバックアップ・サーバーがマスターを引き継ぎます。そのサーバー上のドメイン構成ファイルがマスター・コピーとして使用され、必要に応じて他のサーバーにコピーされます。マスター・サーバーが再始動されると、マスター・サーバーは、現在マスターとして動作しているバックアップ・サーバーからドメイン構成ファイルのコピーを受け取った後、マスターとしての役割を取り戻します。

分散アプリケーションのサポート

クライアント/サーバー Communications Server for Linux システムにおいて、Remote API Client 上で稼働するアプリケーションは、サーバー上の接続リソースと協働して単一タスクを実行します。他の (Communications Server for Linux 以外の) コンピューターで稼働しているアプリケーションも、Communications Server for Linux コンピューターのアプリケーションと協力して分散処理を行うことができます。

Communications Server for Linux は、APPC (LU 6.2 と呼ばれる) を使用して分散アプリケーションをサポートします。

拡張ネットワーク機能

Communications Server for Linux のベース製品には、拡張ネットワーク機能を追加するための機能セットが組み込まれています。これらの機能には、以下のものがあります。

- SNA ゲートウェイは、LAN をサブエリア SNA ネットワークに接続します。
- 1 次 LU サポートは、ホスト・メインフレーム・アプリケーションと同様にダウンストリーム従属型 LU デバイスを制御するサポートを提供します。
- 分岐エクステンダーは、リソースを異なるロケーション (たとえば、大規模な組織内の異なる事業所など) に分割して、大規模 APPN ネットワークを単純化しま

す。これにより、効率の良いリソース・ロケーションを維持しながら、保管が必要なトポロジー情報の量を削減できます。

- APPC Application Suite は、APPN ネットワークで使用するための各種アプリケーションを提供します。
- エンタープライズ・エクステンダー (EE、HPR/IP と呼ばれる) は、SNA トラフィックが IP ネットワークを経由してそのままトランスポートできるようにします。
- TN サーバーは、TN3270 クライアントと総称される TN3270 クライアントと TN3270E クライアントが、SNA 経由でホストにアクセスできるようにします。
- TN リダイレクターは、TN3270、TN3270E、TN5250、および VT クライアント (これらは、Telnet クライアントと総称されます) に、パススルー TCP/IP ホスト・アクセスを提供します。

SNA ゲートウェイ

ゲートウェイは、異種のネットワークまたはコンピューター・システムを接続する、ユーザーに透過的な装置であり、接続されている両方の環境をサポートします。エンド・ユーザー同士は、互いに同じネットワーク内に存在しているように認識します。

SNA ゲートウェイを使用すると、Communications Server for Linux コンピューターは、SNA ネットワーク内の複数のダウンストリーム・コンピューターを 1 つ以上のホスト物理装置 (PU) にリンクさせるゲートウェイとして動作できるようになります。これについては、図 1 で説明しています。ホスト接続を単純化し、余分なリンクを減らすため、SNA ゲートウェイは、PU コンセントレーターとして動作します。つまり、複数のコンピューターを 1 つの PU (これは SNA ゲートウェイ・ノード上にあるように見えます) として扱い、1 つの物理接続を介してホストと通信します。

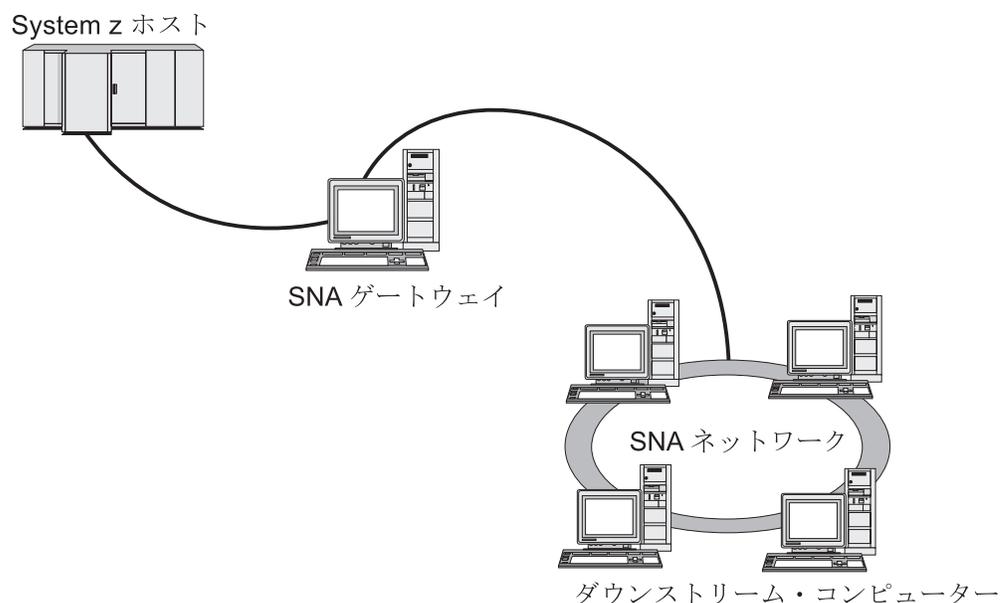


図 1. 複数のダウンストリーム Linux コンピューターをホスト・コンピューターにリンクする SNA ゲートウェイ

1 次 LU サポート

1 次 LU サポートにより、Linux アプリケーションはホスト・メインフレーム・アプリケーションと同様にダウンストリーム従属型 LU デバイスを制御することができます。

通常、LUA アプリケーションは 2 次 LU としてホスト・メインフレームに接続されるので、ホスト・アプリケーションはセッションの定義を制御し、セッションを開始するための BIND の送信を受け持ちます。また、Communications Server for Linux は、1 次 RUI インターフェースを使用することにより、LAN を介してダウンストリーム従属 SNA デバイスに対する 1 次 LU として機能することができます。このインターフェースを使用すると、アプリケーションはホスト・メインフレームを必要とせずにダウンストリーム従属 LU セッションに接続できます。

1 次 LU アプリケーションを使用するには、#PRIRUI# というホスト LU 名を使用して、ノードをダウンストリーム LU (または ダウンストリーム PU テンプレート) で構成する必要があります。この構成により、1 次 RUI を使用するアプリケーションがこれらの PU およびそれらに割り当てられた LU リソースを制御することを、サーバーに指示します。PU は LAN ポートでのみ使用することができます。1 次 RUI を使用するアプリケーションのプログラミングについては、

「*Communications Server for Linux LUA プログラマーズ・ガイド*」を参照してください。

分岐エクステンダー

APPN ネットワーク内のネットワーク・ノードは、トポロジー情報 (ネットワーク内の他のノードのロケーションとノード間の通信リンクに関する情報) を保持し、トポロジーの変更時にはネットワーク全体にこの情報を転送する必要があります。ネットワークのサイズが大きくなると、保管する情報量やトポロジー関連のネットワーク・トラフィックが大きくなり、管理が難しくなる可能性があります。

ネットワークを複数のサブネットワークに分けると、こうした問題は避けられ、各ノードは、そのサブネットワーク内のノードに関するトポロジー情報を保持しているのみで済みます。ただし、こうすると、他のサブネットワーク内のリソースを探し出そうとするときに、ネットワーク・トラフィックを増加させることとなります。

9 ページの図 2 に示す、APPN の分岐エクステンダー機能を使用すれば、こうした問題は解消されます。

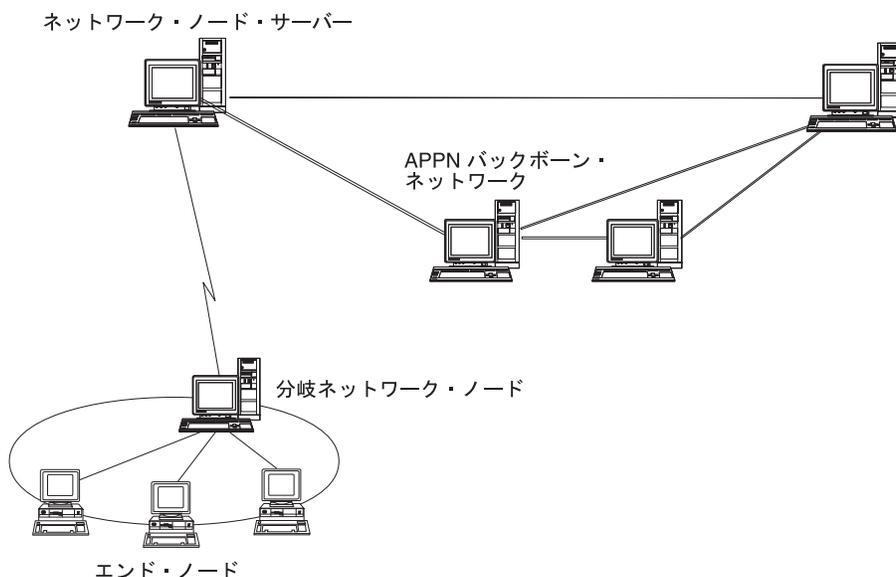


図2. 分岐エクステンダー

名前のおり、分岐エクステンダーは、大規模な組織から分割した部門のように、ネットワークを個々の領域に分割できるよう設計されています。これは、メインのバックボーン APPN ネットワーク (たとえば、組織の本社内のネットワーク) から各分岐に分割することで機能します。

各分岐は、分岐ネットワーク・ノード (BrNN) という新しいタイプのノードを持ち、そのノードが、メインの APPN バックボーン・ネットワーク内の 1 つのネットワーク・ノードに接続されています。BrNN は、APPN ネットワーク・ノードと APPN エンド・ノードの機能を結合します。

- 以下に説明するように、バックボーン・ネットワークからは、BrNN はバックボーン・ネットワーク内のネットワーク・ノード・サーバー (NNS) に接続されたエンド・ノードとして認識されます。
 - バックボーン・ネットワーク内のノードは、分岐内のノードを認識しないため、保管すべきトポロジー情報量が削減されます。
 - BrNN はエンド・ノードとして認識されるため、バックボーン・ネットワークからトポロジー情報を受信することはありません (トポロジー情報は、ネットワーク・ノード間でしか送信されません)。
 - BrNN は、その分岐内のすべてのリソースを、BrNN 自身が保有しているかのように、NNS に登録します。つまり、バックボーン・ネットワーク内のノードは、分岐内のノードを個別に認識する必要なく、分岐内のリソースを探し出すことができます。
- 分岐ネットワークからは、BrNN は、分岐内のエンド・ノードの NNS として動作する、ネットワーク・ノードと認識されます。分岐内の各ノードは、ネットワーク内の残りのノードを、標準 NNS と同様に、NNS を経由して接続されているものと認識します。

APPC Application Suite

APPC Application Suite は、APPN ネットワークの分散処理能力をデモンストレーションする一連のアプリケーションであり、構成の検証および問題の判別に役立ち

ます。APPC Application Suite を使用すると、ファイル転送など、ネットワークを介して頻繁に実行される操作をサポートできます。

APPC Application Suite は、以下のアプリケーションで構成されています。

- **ACOPY** (APPC COPY)
- **AFTP** (APPC File Transfer Protocol)
- **ANAME** (APPC Name Server)
- **APING** (APPC Ping)
- **AREXEC** (APPC Remote EXECution)
- **ATELL** (APPC TELL)

これらのアプリケーションには、サーバー、Linux クライアント、または Windows クライアントからアクセスできます。

エンタープライズ・エクステンダー

エンタープライズ・エクステンダー (HPR/IP と呼ばれます) は、SNA アプリケーションを IP ネットワークと統合する仕組みを提供します。

SNA アプリケーションは、SNA プロトコルを使用し、SNA ネットワークを介して、他の SNA アプリケーションと通信するよう設計されています。エンタープライズ・エクステンダーを使用して TCP/IP ネットワークにインストールすれば、SNA アプリケーションは引き続き通信可能です。エンタープライズ・エクステンダー機能により、IP ネットワークを経由して SNA プロトコルをトランスポートする機構が提供されるためです。特に、APPN 高性能ルーティング (HPR) 機能が提供されているので、アプリケーションに APPN と IP の両方の接続の利点が与えられます。

Communications Server for Linux の Enterprise Extender は、単に通信リンクとしてインプリメントされます。IP 経由で 2 つの SNA アプリケーションを接続するには、SDLC またはイーサネットなどの他のリンク・タイプの場合のように、エンタープライズ・エクステンダー・リンクを定義します。

TN サーバー

SNA ネットワークではなく TCP/IP を使用して通信を行う 3270 エミュレーション・プログラムは、「TN3270 プログラム」(Telnet 3270 エミュレーション・プログラム) と呼ばれます。

TN3270 プログラムには、TN3270E (Telnet 3270 標準拡張) サポートを組み込むこともできます。TN3270E は、Telnet を使用する 3270 装置エミュレーション (端末とプリンターの両方を含む) をサポートします。これにより、Telnet クライアントから (LU 名または LU プールの名前を指定して) 特定の装置を選択できるようになります。また、ATTN キーおよび SYSREQ キー、SNA 応答処理などの各種機能も拡張サポートされます。

注: 本書では、TN3270、TN3287、TN3270E プロトコルのすべてに適用できる情報については、TN3270 という用語を使用します。

Communications Server for Linux TN サーバーを使用すると、3270 ホスト・コンピューター以外のコンピューターの TN3270 ユーザーが、この 3270 ホスト・コンピ

IBM Communications Server for Linux の機能とパッケージ化

ユーザーへアクセスできるようになります。TN サーバーにより、直接リンクがなくても、TN3270 ユーザーは Communications Server for Linux ユーザーまたはその他の TN3270 ユーザーとホストへの接続を共有できます。また、TN3270 ユーザーは、TCP/IP を実行していないホストへもアクセスできます。

Communications Server for Linux TN サーバー機能を、図 3 に示します。

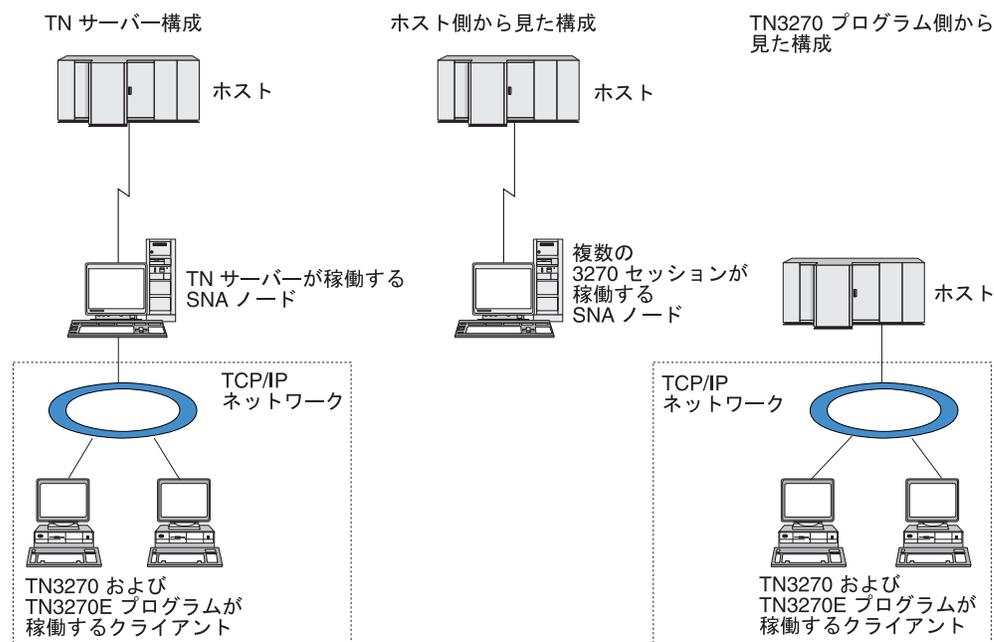


図 3. TN サーバー

Communications Server for Linux TN サーバー機能は、TN3270 ユーザーと Communications Server for Linux 3270 LU との関連付けを行います。TN3270 ユーザーからのデータはすべて、関連付けられた LU へ転送されます。つまり、ホストと TN3270 ユーザーのどちらの構成でも、両者が直接接続しているように構成されています。したがってホストもユーザーも、データが TN サーバーを介して転送されることを認識する必要はありません。

Communications Server for Linux TN サーバーは、IETF RFC 1123、1576、1646、1647、2355 で定義されているプロトコルを正しくインプリメントしている、TN3270 クライアント・エミュレーション・プログラムをすべてサポートします。

セキュリティー機能: Communications Server for Linux TN サーバーは、Secure Sockets Layer (SSL) ソフトウェアを使用して、データ暗号化、サーバー認証、クライアント認証、および高速ログオンをサポートします。

- データ暗号化とは、TN サーバーと TN3270 エミュレーターの間のデータ・フローが暗号化された形式になっていることです。
- サーバー認証により、TN3270 クライアントは、接続先の TN サーバーが適切であるかどうか検証できます。

- クライアント認証により、TN サーバーは、そのサーバーに接続する TN3270 クライアントが適切であるかどうか検証できます。TN サーバーは、クライアントの許可が取り消されていないか確認するため、外部ディレクトリー・サーバーの取り消しリストも検査できます。
- 高速ログオンは、クライアント認証と連動して、TN3270 クライアントがホストに接続する際にユーザー ID とパスワードを指定する必要をなくします。その代わりに、クライアントのセキュリティー証明書を使用することによって、必要なユーザー ID とパスワード情報が取得されます。

TN リダイレクター

Communications Server for Linux TN リダイレクター機能は、TCP/IP による 3270、5250 または VT セッションにパススルー・サービスを提供します。Telnet ユーザーは、TCP/IP 接続を介して Communications Server for Linux との通信を行います。Communications Server for Linux は、次に別の TCP/IP 接続を介してホストとの通信を行います。

Communications Server for Linux TN リダイレクターは、3270 用の TN サーバーの場合と同じように、Secure Sockets Layer (SSL) ソフトウェアを使用して、データの暗号化、サーバー認証およびクライアント認証をサポートします。これにより、ユーザーは、ユーザーからホストへの完全な接続を行わなくても、必要に応じて、Secure Sockets Layer (SSL) のセキュリティー検査を使用することができます。たとえば、次のとおりです。

- クライアントが、検査を必要としない TCP/IP LAN を経由して Communications Server for Linux に接続しており、しかも SSL が必要なリモート・ホストに接続している場合、Communications Server for Linux とホストの間の TCP/IP を介して SSL を使用できます。つまり、すべてのクライアントに対するセキュリティーを一度検査すれば、個々のクライアントがセキュリティー情報を提供する必要はないということです。
- Communications Server for Linux がホストと同じサイトにインストールされていても、クライアントが外部サイトから接続している場合、ホストに SSL ソフトウェアをインストールしなくても、Communications Server for Linux にクライアント接続すれば SSL を使用できます。

機能および利点

Communications Server for Linux の機能および利点は、構成の単純化から問題診断の改善、ネットワーク・パフォーマンスの強化まで多岐にわたっています。

多用途の構成要素

Communications Server for Linux は、ほとんどの環境およびノード機能をサポートします。サブエリアまたは APPN など、どのタイプのネットワークであっても、CS/AIX により、Linux コンピューターは、以下のいずれかとして、または、以下を組み合わせたものとして機能できます。

- ホスト従属ノード
- 対等ノード (APPN 対等ノードについては、1 ページの『IBM Communications Server for Linux』の対等ネットワークの説明を参照してください)
- 分散アプリケーションにおけるパートナー (ソースまたは宛先のいずれか)

- SNA ネットワークを相互接続するゲートウェイ・ノード

ネットワーク管理 API を通して、Linux コンピューターは、分散ネットワーク管理をサポートするための管理サービス (MS)・エントリー・ポイントとして機能するよう構成することもできます。リンク・レベルでは、Linux コンピューターは、サポートされているリンク・タイプ (1 ページの『IBM Communications Server for Linux』および 22 ページの『インストール要件』で説明) のいずれかを使用することにより、各種の LAN および WAN に接続できます。

クライアント/サーバー・オペレーション

クライアント/サーバー構成には、以下の利点があります。

- SNA リソースをサーバーに集中させることによりクライアントの負荷が軽減し、クライアントのパフォーマンスが向上し、SNA サービスをクライアントに提供するために必要なストレージが最小化されます。
- 異なるマシン上の複数のユーザーが単一のデータ・リンクを共用できるので、各マシンが物理 SNA ネットワーク接続を持つ必要がありません。
- 複数のサーバーが冗長接続を準備できます (たとえば、複数のサーバーから同じホストへアクセスできるようにして)。1 つの SNA リソースへ複数のパスを用意しておくことによって、複数の異なるサーバー間でのロード・บาลancingが可能になり、特定のサーバーまたはリンクに障害が起きた場合に即時バックアップを行うことができます。
- 複数のサーバーにわたる LU プールを使用することにより、管理者はサーバーとユーザーの構成および追加を簡単に行うことができます。
- ホスト接続用のリンクと PU が少なくなるので、ホスト VTAM[®] 定義のサイズが減少します。
- 管理ユーティリティを使用して、ノード・リソース (ドメイン内のいずれかのコンピューター用) と共用リソースの両方を構成し、管理することができます。Communications Server for Linux 管理ツールによって提供されるクライアント/サーバー・サポートを利用することにより、ドメイン内の任意のコンピューターから取得したすべてのドメイン・リソースを透過的に管理することができます。
- ファイアウォールの通過、および認証とセキュリティのために TCP/IP および HTTPS を使用するインターネット・プロトコルを介して、SNA アプリケーションを接続できます。

容易な構成

Communications Server for Linux は、構成時間およびネットワークの複雑性を低減するための構成オプションおよび機能を備えて設計されています。たとえば、次のとおりです。

Motif 管理プログラム

Communications Server for Linux 構成を定義および変更する場合、Motif 管理プログラム (`xснаadmin`) を使用する方法が最も簡単です。このプログラムのグラフィカル・ユーザー・インターフェースにより、Communications Server for Linux リソースを表示および管理できます。このプログラムは、インストール環境ごとに異なる値を持つことが多いフィールドのみを表示し、その他のフィールドにはデフォルト値を使用して、構成を単純化します。

Motif 管理プログラムには、ヘルプ画面があります。この画面では、SNA および Communications Server for Linux に関する概要、Communications Server for Linux ダイアログの参照情報、特定の操作の説明が表示されます。

APPN ネットワークの動的構成

ノードまたはネットワークの構成も、APPN ネットワークの動的構成を使用すると簡単になります。たとえば、APPN エンド・ノードおよびアプリケーションは、LU 6.2 セッションをサポートする構成データを動的に登録するので、セッション構成はオプションで行えばよいようになります。さらに、ノード制御点をデフォルトのローカル LU として機能させて、LU 6.2 構成を完全に避けることができます。

APPN は、構成されたリンク・ステーションがない場合の動的リンク・ステーション構成もサポートします。

追加の管理用ユーザー・インターフェースの選択

Motif 管理プログラムは、Communications Server for Linux を構成および管理するために推奨されるインターフェースです。ただし、ユーザーは Communications Server for Linux のインターフェースを選択できます。これにより、装置、ニーズ、およびプリファレンスに合ったインターフェースで作業できるようになります。

コマンド行管理プログラム

コマンド行管理プログラム (**snaadmin**) を使用して、個々の Communications Server for Linux リソースを管理するためのコマンドを発行できます。

snaadmin は、Linux コマンド・プロンプトから直接使用することも、シェル・スクリプト内から使用することもできます。

NOF API

Communications Server for Linux NOF API は、コマンド行管理プログラムと同じ機能を提供します。これは、(コマンド・スクリプトではなく) プログラム内で使用するのに適したインターフェースを提供します。NOF API を使用して、Communications Server for Linux を管理するためのアプリケーション・プログラムを作成できます。

パフォーマンスの改善

Communications Server for Linux は、SNA ネットワークの本来の高性能を強化し、サービス・オペレーションのクラスを使用します。また、Communications Server for Linux は、以下のように、LU 0-3 セッション・データに対して SNA データ圧縮を使用し、ネットワーク・サイズに従ってトラフィック・フローのバランスを取る各種トラフィック管理メソッドを使用して、ネットワーク速度を最適化します。

- APPN ネットワークでは、Communications Server for Linux は、高性能ルーティング (HPR) と中間セッション・ルーティング (ISR) の両方をサポートし、接続ネットワーク・オプションを提供します。ISR は、小規模ネットワークの場合は効率的に機能しますが、大規模ネットワークの場合はパフォーマンスが低下します。
- LAN 接続オプション (トークンリングまたはイーサネットなど) またはエンタープライズ・エクステンダーを使用する大規模ネットワークの場合、接続ネットワーク・オプションを使用して通信効率を上げることができます。接続ネットワー

ク・オプションは、ノード間に通信パスを直接作成します。これにより、トラフィックは中間ネットワーク・ノードをバイパスできるようになります。

- もう 1 つのトラフィック制御機能、適応セッション・レベル・ペーシングは、LU がメッセージ単位をパートナー LU に送信する速度を調節して、輻輳を自動的に調整します。

セキュリティー・オプション

ネットワークがさらに複雑化するにつれ、また、オープン・アーキテクチャーへ移行するにつれて、セキュリティーが重要な問題として取り上げられるようになってきました。Communications Server for Linux が稼働する SNA ネットワークでは、構成で各種レベルのセキュリティーを定義して、特定のタイプのリンクをインプリメントすることにより、お客様の資産を保護します。たとえば、次のとおりです。

- クライアント/サーバー・システムでは、WebSphere サーバーが Remote API Client からサーバーへの HTTPS アクセスを提供するようにセットアップできます。これは、クライアント接続がセキュリティー証明書を使用して認証されることを意味します。(この機能には、標準の Communications Server for Linux 製品のほかに、何らかの機能ソフトウェアが必要です。詳しくは、22 ページの『インストール要件』を参照してください。)
- LU 6.2 ユーザーは、3 つまでのレベルのセキュリティー (セッション、リソース、会話) を定義できます。セッション・レベルのセキュリティーは、セッションで適切な LU が利用されることを確実にし、リソース・レベルのセキュリティーは、特定の LU に関連付けられたすべてのアプリケーションへのアクセスを制限します。会話レベルのセキュリティーは、特定のアプリケーションへのアクセスを制限します。データ暗号化ルーチンによって、追加のセキュリティーが可能です。
- Communications Server for Linux TN サーバーおよび TN リダイレクターは、Secure Sockets Layer (SSL) ソフトウェアを使用して、Communications Server for Linux サーバーと TN3270 または Telnet クライアントの間で、データの暗号化、サーバー認証およびクライアント認証をサポートすることができます。(この機能には、標準の Communications Server for Linux 製品のほかに、何らかの機能ソフトウェアが必要です。詳しくは、22 ページの『インストール要件』を参照してください。)

ネットワーク管理の柔軟性

Communications Server for Linux は、複数ドメイン・サポート・ネットワーク管理ベクトル・トランスポート (MDS-NMVT) ネットワーク管理方式をサポートします。この方式は、集中、分散、または階層管理方式として機能できます。これは、フォーカル・ポイント/エントリー・ポイント・アーキテクチャーに基づいて高度な柔軟性を提供するものです。

フォーカル・ポイントは、エントリー・ポイント (ネットワーク内の他のすべてのノードにある管理アプリケーション) から収集するデータに基づいて、ネットワークを管理する制御ノードです。

- 集中管理では、単一フォーカル・ポイントが、ネットワーク全体の制御点として機能します。
- 分散管理では、複数のフォーカル・ポイントがネットワーク管理を行ないます。

- 階層管理では、複数のフォーカル・ポイントが機能に従ってネスト構造を取りま
す。

MDS-NMVT は、こうして、サブエリア、標準 APPN、および超大規模ネットワー
クを管理するように適合させることができます。

信頼性、可用性、および保守容易性

信頼性のあるシステム操作を保守しやすくするように、Communications Server for
Linux は、表示機能から問題診断ツールまでの一連の機能を提供します。

- Motif 管理プログラムは、以下のように、優れた構成および管理ツールを提供し
ます。
 - 構成情報の即時更新
 - リンク、セッション、およびノード・リソースの状況情報
- query コマンドと status コマンドを使用すると、以下に関する情報を取得できま
す。
 - LU-LU セッション
 - APPN 中間セッション
 - アクティブ・リンク
 - APPN トポロジー・データベース。これは、リンク情報を保管します。
- 問題診断ツールは、構成および操作のさまざまな段階で役に立ちます。これらの
ツールには、以下のものがあります。
 - 診断情報収集ツール (**snagetpd**)。サービス情報を容易に収集できます。
 - 状況およびエラー・メッセージ。構成ルーチンおよびシステム操作の問題解決
に役立ちます。
 - ログ。ネットワークのエラー、障害、および監査情報を収集します。
 - トレース機能。詳細な問題情報を収集し、形式設定します。

その他のユーティリティーは、アプリケーション間のリンク接続および通信のテスト
に役立ちます。

Communications Server for Linux には、管理サービス API があります。これは、ネット
ワーク・アラートや問題データの送受信のためのツールを開発します。

これらの管理ツールと問題診断ツールは、すべて Communications Server for Linux
Client/Server モデルに完全に組み込まれています。このため、Communications Server
for Linux ドメイン全体を管理したり、ネットワーク内の単一点から診断情報を収集
したりできます。

ネットワークの統合、成長および変更

ネットワークの統合、成長、および変更をサポートするために、Communications
Server for Linux API を使用して、ビジネス・ニーズに合った、特定の LU、プラッ
トフォーム、またはオペレーティング・システム用のアプリケーションを開発でき
ます。CPI-C は、異種のプラットフォームおよびオペレーティング・システムを通
じて一貫性があるため、特に重要な API です。これを使用して、任意のシステムで
実行可能なアプリケーションを開発できます。

エンタープライズ・エクステンダーも、SNA ネットワークと TCP/IP ネットワークを統合するためのメカニズムを提供します。

第 2 章 ネットワークおよび IBM Communications Server for Linux の計画

この章では、Communications Server for Linux が稼働するネットワーク計画の段階の概要を説明します。また、Linux コンピューターに構成可能な機能の要約と、それらの機能をサポートするために必要なリソース量の見積もりのガイドラインも示します。

ネットワーク計画の段階

このセクションでは、Communications Server for Linux および Linux の各種ユーティリティを使用してネットワークの計画、構成、管理を行う際の一般的なガイドラインを示します。

ネットワークの計画には、機能の平衡化、パフォーマンス、リソースおよびコストが関係します。ネットワークに最適な計画を 1 つだけ示すことはできませんが、いくつかの一般ガイドラインおよび技法が、確実にニーズに合う計画を立てる際に役立ちます。ネットワークを計画するために、以下の作業を行います。

- ネットワークが提供する機能（ファイル転送や 3270 エミュレーションなど）およびパフォーマンス要件を決定します。
- 必要な機能を提供するための Communications Server for Linux の構成方法を決定します。
- Communications Server for Linux のインストール、パフォーマンスおよび容量要件のサポート、Communications Server for Linux 機能のサポートを行うために必要なリソースを見積もり、そのために要するコストを算出します。

ネットワークの機能要件の確認

ネットワークで提供すべき機能を判別するには、以下の設問を考慮する必要があります。

- ネットワークは APPN である必要があるか。
- Communications Server for Linux をクライアント/サーバー・システムとして実行するか。実行する場合、すべてのコンピューターを単一の Communications Server for Linux ドメインで操作するか、または 2 つ以上の別個のドメインを定義する必要があるか。
- 接続リソースのロード・バランシングを取るために、複数のサーバーを Communications Server for Linux ドメインに設ける必要があるか。必要な場合、どのサーバーがマスター構成サーバーになるか。1 つ以上のバックアップ構成サーバーが必要か。
- HTTPS を使用して Remote API Client から Communications Server for Linux サーバーへの接続をサポートする必要があるか。
- ユーザー・アプリケーションはサーバー上で実行するのか、それとも Linux クライアント・コンピューター上で実行するのか。

ネットワーク計画の段階

- Windows クライアントで実行する Windows アプリケーション (たとえば、API トランザクション・プログラム) 用の接続リソースをサーバーが提供するのか。
- 各サーバーをセッションのエンドポイントにするか、あるいは、以下のいずれかのタイプのゲートウェイにするか。
 - APPN
 - LU 0
 - LU 2
 - TN サーバーまたは TN リダイレクター
- ネットワークはどのタイプの物理リンクを使用するか。

上記の質問の回答は、ネットワークに必要な Communications Server for Linux の機能を決定するのに役立ちます。

Communications Server for Linux の構成方法の決定

Communications Server for Linux の機能方法を決定するには、まず、ネットワークに流れる作業の種類を決定する必要があります。考慮すべき事項は、以下のとおりです。

- ネットワークを介して使用できる必要のあるリソース (アプリケーションなど)
- リモート・リソースにアクセスする必要があるユーザーの数
- 各リソースへのアクセス頻度
- ユーザーのネットワークへのアクセス取得方法
- ユーザー要求がネットワークを通してルーティングされる方法

Communications Server for Linux は、たとえば、以下のノードなど、多数の機能をサポートするように構成できます。

- 中間セッション・ルーティング (ISR) 用の APPN ネットワーク・ノード
- APPN エンド・ノード (隣接ノードとは自発的に通信しますが、隣接していない対等ノードとは、APPN ネットワーク・ノード・サービスを使用して通信します。)
- ローエントリー・ネットワーキング (LEN) ノード (隣接ノードまたは隣接しているように見えるよう構成されたノードと直接に通信します。)
- LU 0、LU 1、LU 2、LU 3 および LU 6.2 (従属および独立) の使用
- ホストへの SNA ゲートウェイ接続

ニーズに応じて、ある 1 つのノードに対して 1 つ以上の機能を構成できます。たとえば、Communications Server for Linux を APPN ネットワーク・ノードとして構成し、ルーティング・サービスおよび ISR を提供したり、SNA ゲートウェイに対して同じノードを使用して、LU 0 や LU 2 などの従属 LU セッションを経路指定したりできます。同様に、TN サーバーを実行し共用データベースをサポートするように Communications Server for Linux を構成することもできるし、ホスト上の MQSeries® への独立 LU 6.2 接続を構成することもできます。

インストールおよび操作のリソース要件の確認

Communications Server for Linux の機能のサポートを見積もるには、以下の問題の回答が必要です。

- 要員の必要スキルは何か
- どのトランスポート・メディアを使用したいか
- 選択する構成のインストール要件は何か
- 操作に必要なメモリーおよびページング・スペース量はどのくらいか

上記の質問の回答により、19ページの『ネットワークの機能要件の確認』で説明している多くの機能のうち、1つ以上をサポートするように構成する場合に、Communications Server for Linux で使用するリソースのタイプを確認できます。これらの設問に回答することは、Communications Server for Linux 機能、Linux リソース、およびネットワーク・リソース間の関係を理解するのにも役立ちます。

ノードに対するリソースの割り振り方法により、ネットワークの実現方法が決定します。

要員要件

Communications Server for Linux のインストール、操作、および調整を行うには、以下の担当者が必要です。

- ネットワーク管理者。ネットワークの計画、新しい装置の追加、ネットワーク全体のパフォーマンスの保守および拡張を行います。
- システム管理者。Communications Server for Linux とそこで操作するハードウェアのインストールおよび保守、ネットワーク接続のシステム構成を行います。
- プログラマー。トランザクション・プログラムやネットワーク管理ルーチンなど、カスタマイズされたアプリケーションを開発します。

ネットワーク管理者およびシステム管理者は、Communications Server for Linux が作動するハードウェア、および Linux オペレーティング・システムに精通していなければなりません。また、さまざまなシステムが接続されるネットワークについての知識があり、一般的な SNA の概念も理解していなければなりません。さらに、以下のことにも精通している必要があります。

- Motif インターフェース
- rpm 、Linux 用のインストール・ツール
- TCP/IP (クライアント/サーバー機能、TN サーバーまたは Enterprise Extender の使用を計画している場合)
- Windows 2000、Windows XP、または Windows Server 2003 オペレーティング・システム (Communications Server for Linux システムに Windows 上の Remote API Client が含まれる場合)
- WebSphere Application Server (Communications Server for Linux システムに、HTTPS を使用してサーバーに接続する Remote API Client が含まれる場合)

SNA 用のカスタマイズ・アプリケーションを開発するプログラマーは、C 言語 (または、Java CPI-C を使用している場合は、Java) の経験者で、Communications Server for Linux で使用可能な API に精通している必要があります。

トランスポート・メディア

Communications Server for Linux が、基盤のトランスポート・メディア (SDLC、トークンリングなど) を他の通信プロトコルと共用する必要がある場合があります。

したがって、物理層の帯域幅要件は、トランスポート・メディアを共用するすべてのプロトコルおよびアプリケーションに適応できなければなりません。

注: Communications Server for Linux は、トークンリングとイーサネット・アダプターを他のプロトコル (TCP/IP など) と共用することができます。使用するプロトコルごとに、固有のサービス・アクセス・ポイント (SAP) アドレスを指定する必要がある場合があります。

インストール要件

Communications Server for Linux に (19 ページの『ネットワークの機能要件の確認』 から) 割り当てる機能もインストール要件を決めます。このセクションでは、Communications Server for Linux のインストールに必要なコンピューター・リソースの概要について説明します。詳しくは、各製品に同梱されている資料を参照してください。

ハードウェア

Communications Server for Linux には、以下の Linux ディストリビューションの 1 つがサポートするコンピューターが必要です。

コマンド `uname -m` を使用して、ターゲット・コンピューターの CPU クラスを確認してください。次の表には、サーバー・タイプごとに適切なハードウェアと、そのハードウェアに対する `uname -m` からの応答が示されています。

サーバー・タイプ	ハードウェア	uname 応答
32 ビット Intel	Pentium® II またはそれ以降の 32 ビット Intel システム	i686
64 ビット x86_64	AMD64 または Intel EM64T システム	x86_64
pSeries	pSeries POWER5™ または OpenPower™ システム	ppc64

Linux オペレーティング・システム

Linux ワークステーションに Communications Server for Linux をインストールして使用できるようにするには、事前に Linux オペレーティング・システムの適切なバージョンをインストールしておかなければなりません。

Communications Server for Linux は、次の Linux 変形をサポートします。それぞれの変形に対してサポートされる特定のバージョン番号およびカーネル・ビルドに関する最新情報については、Communications Server for Linux CD イメージの **README** ファイルを参照してください。

- RedHat Enterprise Linux 3 (RHEL3): i686 のみ
- RedHat Enterprise Linux 4 (RHEL4)
- SUSE Linux Enterprise Server 8 (SLES8): i686 のみ
- SUSE Linux Enterprise Server 9 (SLES9)
- SUSE Linux Enterprise Server 10 (SLES10)

リンク・ハードウェア

リンク・ハードウェアは、サーバー上でのみ必要であり、クライアント上では不要です。

PC を 1 つ以上のネットワークにインストールするには、選択したネットワークの通信プロトコルに準拠するリンクが必要です。リンク・ハードウェアは、コンピューターに取り付けられた通信アダプターと、それにマッチするネットワーク接続用ケーブルからなります (デバイス・ドライバー・ソフトウェアも必要です)。

注:

1. 装着する通信アダプターには、必ず適切なケーブルを接続してください。たとえばイーサネット・アダプターの場合、リンクを正しく機能させるにはイーサネット・ケーブルが必要です。
2. Enterprise Extender を唯一のリンク・タイプとして使用する場合は、このセクションにリストされているアダプターは必要ありません。ただし、TCP/IP で必要なアダプターを 1 つ用意する必要があります。

Communications Server for Linux は、次のリンク・プロトコルの通信アダプターをサポートしています。

- トークンリング
- イーサネット (標準または IEEE 802.3)
- SDLC
- X.25 QLLC

Communications Server for Linux でテストを完了しているアダプターについては、<http://www.ibm.com/software/network/commsserver> を参照してください。ニーズに適したアダプターまたはハードウェアについて不明点がある場合は、IBM 営業担当員にお問い合わせください。

追加ソフトウェア:Linux

Communications Server for Linux には、次の追加ソフトウェアが必要です。特定のバージョン要件 (Linux 変形によって異なる) およびこれらのソフトウェア・パッケージのインストール手順について詳しくは、Communications Server for Linux CD イメージの **README** ファイルを参照してください。

- LiS ストリーム。必要な LiS ストリーム・バージョンとインストール手順について詳しくは、**README** ファイルを参照してください。
- OpenMotif (サーバー上でのみ必要であり、クライアント上では不要です)。これは、Motif 管理プログラムを使用するのに必要であり、Communications Server for Linux を構成し管理するための推奨方式です。
- Java (Java CPI-C を使用する場合に必要)。Java Runtime Environment (JRE) が必要になります。Java CPI-C アプリケーションと使用するために新規 Java クラスをコンパイルする必要がある場合、Java SDK も必要になります。

WebSphere Application Server (HTTPS アクセス用)

Remote API Client が HTTPS を使用して Communications Server for Linux サーバーに接続するようなクライアント/サーバー・システムを実行する予

定の場合、それらのクライアントからサーバーへの HTTPS アクセスを提供するために WebSphere Application Server が必要になります。

Communications Server for Linux は、WebSphere Application Server バージョン 5 と連動して作動します。WebSphere Application Server は、WebSphere がサポートするオペレーティング・システムが稼働しているコンピューターにインストール可能です。(必要な場合、Communications Server for Linux サーバーと同じ Linux コンピューターにインストールできます。) インストールについての詳細は、WebSphere Application Server の資料を参照してください。さらに、36 ページの『WebSphere Application Server の構成』に説明されているように、WebSphere を Communications Server for Linux と一緒に使用するために、追加の Communications Server for Linux プラグインをこのコンピューターにインストールする必要があります。

メモリーおよびストレージ

すべての範囲にわたる構成やサービスをサポートするには、Communications Server for Linux を稼働するワークステーションに、Linux 配布版に必要な最小メモリーと 32 MB、および 120 MB のディスク・スペースが必要です。また、インストール時には 250 MB の一時ストレージが必要となります。

Communications Server for Linux の資料をソフトコピー (PDF) 形式でインストールする場合は、追加のハード・ディスク・ストレージが必要です。すべてのソフトコピー資料をインストールするには、75 MB のディスク・スペースが必要です。

注: その他のライセンス・プログラム、ユーザー・アプリケーション、データに必要なメモリーおよびハード・ディスクの要件は、上記の要件には含まれていません。システム、メモリー、ハード・ディスクのすべての要件について、IBM 担当員または特約店と共に慎重に検討してください。

拡張構成

基本構成以外の構成を必要とするアプリケーションを実行する場合には、コンピューター・リソースが余分に必要となります。

一般に、Communications Server for Linux を実行するためのメモリーおよびストレージの所要量は、Linux ワークステーションの機能とその環境によって異なるさまざまな要因によって決まります。ただし、コンピューター・リソースを主に消費するのは、LU、アクティブ・リンクおよび進行中のセッションであるという経験法則を利用して、最も要求の多いアプリケーションをサポートするのに必要な、追加のメモリーおよびディスク・スペースの容量を見積もることはできます。

操作に必要なメモリーおよびストレージ容量

このセクションでは、Communications Server for Linux を実行するワークステーションのメモリーおよびストレージ要件について説明します。

カーネル・メモリー

Linux は、カーネル・メモリーを使用して、通信サブシステムと AIX DLC 間で通信できるようにします。カーネル・メモリーは共用リソースであり、

システム・レベルで管理する必要があります。 Communications Server for Linux は、カーネル・メモリーを使用して、ネットワーク上のデータを送受信しますが、それは、カーネル・メモリーを使用するサブシステムの 1 つにすぎません。 Communications Server for Linux のカーネル・メモリー・リソースの使用状況が、他のサブシステム (たとえば TCP/IP、NFS など) のパフォーマンスに影響を与える場合があります。カーネル・メモリー所要量を決定するには、ネットワーク管理者と相談してください。

メモリー、ディスク、およびページング・ストレージ

Communications Server for Linux は、メイン・メモリーおよび、ランダム・アクセス・メモリー (RAM) と呼ばれる 1 次ストレージ、永続ディスク・ストレージ、およびページング・ディスク・ストレージ (ページ・スペースとも呼ばれます) を使用します。

- SNA リソース (LU、リンク、セッションなど) の制御ブロックは、1 次ストレージを使用します。
- カーネル・メモリーは 1 次ストレージのみを使用します。
- Communications Server for Linux 実行可能プログラム、構成ファイル、およびソフトコピー資料は、永続ディスク・ストレージを使用します。
- アプリケーション・プログラム、セッション、およびそれに関連したプロセスにはページング・スペースが必要ですが、これは、永続ディスク・ストレージを使用します。

メモリーおよびディスク・ストレージの所要量はさまざまな要因によって決まりますが、これらの要因は、環境によって大きく異なります。メモリーおよびディスク・ストレージを大量に消費するのは、トランザクション・プログラム (TP) および SNA セッションです。

TP は、SNA アプリケーション・プログラミング・インターフェース (API) を使用して、ネットワークにアクセスするプログラムです。セッションは、パートナー・ノード上の 2 つの LU 間の一時的論理チャネルです。TP はこのチャネルを使用して、互いに通信を行います。

メモリーおよびディスク・ストレージの使用量は、主に、セッションの全体数、割り振られたセッション (会話) の数、会話のトラフィック量、アクティブ・ユーザー TP 数の影響を受けます。メモリー所要量は、主として、データ・トラフィック量とその結果のカーネル・メモリー使用量による影響を受けます。ディスク・ストレージ所要量は、TP、セッション、および関連プロセスの、ページング・スペース所要量によって決まります。

LU のすべてのタイプにおいて、アイドル・セッションと、割り振られたセッションは区別されます。アイドル・セッションとは、TP が使用していないアクティブ・セッションです。割り振られたセッションとは、進行中の会話をサポートしているセッションです。メモリー所要量の合計を見積もるには、トラフィック量と、総セッション数のうち、平均して割り振られる数およびある一時点で割り振られる数を見積もる必要があります。ピーク時の要求処理に合わせて、システムを構成することもできます。これらの見積もりは、環境によって異なります。

セッション負荷がシステム・メモリー容量を超えるような場合は、ページングのコストが増大します。

プロセス

Communications Server for Linux は、最小 20 Linux プロセスを必要とします。

クロスプラットフォーム構成における互換性の保証

異なるプラットフォームで稼働する SNA 製品 (たとえば、IBM パーソナル・コミュニケーションズまたは Communications Server for Windows、ホスト上の VTAM NCP など) は、一定の要件が満たされた場合は Communications Server for Linux と共に機能します。

一般に、SNA ネットワーク製品の現行リリースは、PU 2.1 ノードをサポートし、Communications Server for Linux がサポートするリンク・タイプの少なくとも 1 つをサポートしているため、Communications Server for Linux と同様の機能を果たすことができます。ただし、かなり以前のリリース (VTAM V2 など) は、確実に機能するとは限りません。それぞれのネットワーク製品の機能は、当該製品のユーザーズ・ガイドに記載されています。

考慮すべきその他の要因には、次のようなものがあります。

- 独立 LU 6.2 を使用しており、ネットワークが APPN でない場合は、パートナー LU がローカル・システムに対して定義されていることを確認する必要があります。
- DLC ウィンドウ・サイズとタイマーのデフォルト値はデバイスによって異なるので、リモート・デバイスが正しい値を使用しているかどうか、確認してください。たとえば、ある DLC ウィンドウ・サイズが Communications Server for Windows NT[®] ノードでは良好なパフォーマンスをもたらしても、3172 ノードでもそうなるとは限りません。
- 複数の LAN セグメントを持つ異種混合環境 (TCP/IP プロトコルと SNA プロトコルが、同一 LAN 上にあるなど) で作業する場合は、LAN 相互接続デバイスが、TCP/IP の「ルーティング」と SNA フレームの「ブリッジ」を同時に実行することができるかどうか、確認する必要があります。

命名規則

ネットワーク ID を使用すると、物理ネットワークを論理的にセグメント化することができます。また、他のネットワークに接続する計画がある場合は、ネットワーク名の競合を避けるためにも、ネットワーク ID を登録することを強くお勧めします。

ネットワーク名および LU 名は、次のように定義できます。

ネットワーク名

それぞれ異なるネットワーク名 (ネットワーク ID) を定義することによって、APPN ネットワークのセグメンテーションを行うことができます。セグメンテーションにより、ネットワーク・トポロジー・データベースのサイズ、および個々のネットワークを流れるブロードキャスト LOCATE 要求の頻度を規制できます。

ネットワーク ID の固有性を保証するために、ネットワーク管理者は、IBM のワールド・ワイド・レジストリーにネットワーク ID を登録するこ

とができます。この IBM レジストリーは、そこに登録されているネットワーク ID がそれぞれ固有の ID であることを保証します。登録規格は、国際標準化機構 (ISO) が確立した、オープン・システム間相互接続 (OSI) の規格 (OSI 国別コードを含む) に準拠しています。登録について詳しくは、「*User's Guide for SNA Network Registry*」を参照してください。

LU 名 LU 名にワイルドカードを使用すると、システム定義とネットワーク検索を最小限にとどめることができます。

第 3 章 Linux サーバーでの Communications Server for Linux のインストール

この章では、Communications Server for Linux を Linux サーバーにインストールするために必要な実行ステップについて概説します。インストール・プロセスのステップバイステップの手順について詳しくは、Communications Server for Linux CD イメージの **README** ファイルを参照してください。そこでは、ご使用の Linux 変形に固有の詳しい最新情報が示されています。**README** ファイルには、以下のものが含まれています。

- 必要な追加のソフトウェア・パッケージに関する正確なパッケージ名およびバージョン番号
- インストールおよびセットアップ・コマンドの詳細

Communications Server for Linux ライセンス・プログラムのパッケージ内容

Communications Server for Linux ライセンス・プログラムは CD イメージとして配布され、以下のものを収録しています。

CD #1: クイック・スタート

この CD には、Communications Server for Linux の PDF 資料の完全セットが (ディレクトリー **/DOCS** に) 収録されています。

CD #2: サーバー

この CD には、サーバーのインストールに必要な以下のファイルが収録されています。

- **README** ファイル (Communications Server for Linux 資料の発行後に行われた製品変更に関する情報が入っている)
- インストール・スクリプト
- サーバーおよび PDF 資料のインストール・イメージ

CD #3: クライアント

この CD には **README** ファイル、以下の各タイプのクライアントをインストールするために必要なインストール・スクリプトとインストール・イメージが収録されています。

- 32 ビット Intel (i686) 用 Linux クライアント
- 64 ビット AMD64/Intel EM64T (x86_64) 用Linux クライアント
- pSeries (ppc64) 用 Linux クライアント
- System z9™ または System z (s390 / s390x) 用 Linux クライアント
- AIX クライアント
- 32 ビット Windows クライアント
- x64 Windows クライアント

CD イメージに収録されているファイルの詳細については、**README** ファイルを参照してください。

注: 必要なストレージの容量については、22 ページの『インストール要件』を参照してください。

Communications Server for Linux インストールの準備

Communications Server for Linux をインストールする前に、次の作業を行います。

- パワー・マネージメント機能を使用可能にして、Linux PC が稼働中ではないことを確認する。
- 22 ページの『インストール要件』にリストされている前提ソフトウェアが、すべてインストール済みであることを確認する。

以下のセクションでは、これらの作業について説明します。

パワー・マネージメント機能の使用不可化

Communications Server for Linux をインストールする前に、ご使用の Linux PC に拡張パワー・マネージメント機能が組み込まれているかどうか調べる必要があります。組み込まれている場合は、これらの機能が使用不可であることを確認してください。現在、多くの PC にこの機能が組み込まれています。この機能は、キーボードやマウスを使用していない場合に、(ディスク・ドライブなどの装置への電力を中断したり、CPU クロック速度を落としたりすることによって) 電力を節約するように設計されています。拡張パワー・マネージメント機能は、ワークステーションとしての PC で使用するように設計されています。サーバーは、キーボードやマウスを動かさずに通常どおりに操作できるので、PC をサーバーとして操作しているときにこの機能を使用することはお勧めできません。

この機能を使用不可にするには、PC の起動時に BIOS 構成にアクセスし (通常、ハードウェアの初期化時に **F1** を押します)、拡張パワー・マネージメントとハードウェア・パワー・マネージメントの両方を使用不可にします。

拡張パワー・マネージメント機能を PC から除去するには、次のコマンドを使用します。

```
rpm -e apmd
```

前提ソフトウェアのインストール

Communications Server for Linux をインストールする前に、22 ページの『インストール要件』にリストされている前提ソフトウェアをインストールする必要があります。

- LiS ストリーム
- OpenMotif
- Java
- SSL

必要なソフトウェア・パッケージの詳細およびそれらのソフトウェア・パッケージをインストールするためのステップバイステップの手順については、Communications Server for Linux CD イメージの **README** ファイルを参照してください。

製品インストール詳細の表示

インストール済みの Communications Server for Linux と関連ソフトウェア・パッケージに関する情報を表示することができます。インストール済みのパッケージをすべてリストするには、以下のコマンドを使用します。

```
rpm -q -a
```

ある特定のパッケージの詳細を表示するには、以下のコマンドを使用します。

```
rpm -q -i packagename
```

packagename は、インストール済みパッケージのベース・ネーム (たとえば **ibm-commserver**) です。

言語環境変数の変更

Communications Server for Linux を使用する場合は、LANG 変数が、使用する言語を示すように正しく設定されていることを確認してください。

LANG 変数を変更するには、次のコマンドを使用します。

```
export LANG=language
```

language は、使用したい言語の ID に置き換えてください。以下のいずれかを使用できます。

ID	言語
en_US	英語 (US)
ja_JP	日本語 (PC)
de_DE	ドイツ語
es_ES	スペイン語
fr_FR	フランス語
ko_KR	韓国語
pt_BR	ポルトガル語
zh_CN	中国語 (簡体字 EUC)
zh_TW	中国語 (繁体字)

旧レベルの Communications Server for Linux からのマイグレーション

旧レベルの Communications Server for Linux がインストールされているときに、バージョン 6.2 にマイグレーションしたい場合は、以下のステップを実行する必要があります。

カスタマイズ済み構成ファイルを保管する

以下にリストされているファイルのいずれかが存在している場合は、それらを一時ディレクトリに保管してください。これらのすべてのファイルが、すべてのインストール・システムに存在しているわけではありません。

旧レベルの Communications Server for Linux からのマイグレーション

```
/etc/opt/sna/sna_node.cfg
/etc/opt/sna/sna_domn.cfg
/etc/opt/sna/sna.net
/etc/opt/sna/sna_tps
/etc/opt/sna/ibmcs.kdb
/etc/opt/sna/ibmcs.sth
/etc/opt/sna/ibmcs.rdb
/etc/opt/sna/ibmcs.crl
/etc/opt/ibm/sna/sna_node.cfg
/etc/opt/ibm/sna/sna_domn.cfg
/etc/opt/sna/sna.net
/etc/opt/ibm/sna/sna_tps
/etc/opt/ibm/sna/ibmcs.kdb
/etc/opt/ibm/sna/ibmcs.sth
/etc/opt/ibm/sna/ibmcs.rdb
/etc/opt/ibm/sna/ibmcs.crl
```

さらに、スタートアップ・ファイル `/etc/rc.d/init.d/snastart` を 78 ページの『Communications Server for Linux の使用可能化』の説明に従ってカスタマイズし `sna start` コマンドを除去して、システム起動時に Communications Server for Linux が自動的に開始されないようにした場合は、このファイルに行った変更をメモしておいてください。

古いリリースをアンインストールする

以下のコマンドを使用して Communications Server for Linux を停止し、それをアンインストールします。インストールされていた旧バージョン、およびそのインストール方法によっては、ここにリストされている RPM パッケージがすべてシステムに存在しているとは限りません。

```
sna stop
rpm -e CS-LINUX-ptf
rpm -e CS-LINUX-Docs
rpm -e CS-LINUX-ecl
rpm -e CS-LINUX
rpm -e ibm-conmserver-ptf
rpm -e ibm-commserver-docs
rpm -e ibm-commserver-ecl
rpm -e ibm-commserver
rpm -e gsk5bas
rpm -e gsk6bas
```

LiS をアンインストールする

以下のコマンドを使用して、現行レベルの LiS オープン・ソース・パッケージをアンインストールします。

```
PATH=$PATH:/sbin
unset LD_PRELOAD
rmmod streams
cd /usr/src/LiS
make uninstall
```

旧レベルの **Communications Server for Linux** からのマイグレーション

```
make very-clean
cd /usr/src
rm -rf LiS*
```

PATH および他の環境変数を変更する

旧 **Communications Server for Linux** バージョンで以下のいずれかの環境変数を変更した場合は、パスが **Communications Server for Linux** バージョン 6.2.2 プログラム・プロダクトと異なることがあるので、変更したものを除去する必要があります。

```
PATH
LD_LIBRARY_PATH
LD_RUN_PATH
LD_PRELOAD
CLASSPATH
```

env コマンドを使用して、**sna** を参照するすべての環境変数を調べることができます。

```
env | grep sna
```

その他のパッケージ

ご使用の **Java** インストールを調べ、必要な場合、それを最新レベルに更新することをお勧めします。

最終クリーンアップ

次のコマンドは、残っているすべての項目を旧レベルの製品から除去します。インストールされていた旧バージョンによっては、ここにリストされているディレクトリーがすべてシステムに存在しているとは限りません。

```
rm -rf /etc/opt/sna /var/opt/sna /opt/sna
```

```
rm -rf /etc/opt/ibm/sna /var/opt/ibm/sna /opt/ibm/sna
```

新規レベルの **Communications Server for Linux** バージョン 6.2.2 プログラム・プロダクトをインストールする

本書と **README** ファイルの手順に従って、製品をインストールします。

保管構成を復元する

このプロセスの最初のステップでいずれかの構成ファイルを保管した場合は、ここでそれらの構成ファイルを復元します。まず、次のコマンドを使用して、**Communications Server for Linux** ソフトウェアを停止します。

```
/opt/ibm/sna/bin/sna stop
```

このプロセスの最初のステップでいずれかの **ibmcs.*** ファイルを保管した場合は、ここですべての **ibmcs.*** ファイルを **/etc/opt/ibm/sna** ディレクトリーから除去します。たとえば、**ibmcs.kdb** と **ibmcs.sth** を保管した場合、これらのファイルだけでなく **ibmcs.crl** と **ibmcs.rdb** も (置き換えのために保管していなくても) 除去する必要があります。保管済みのファイルと新しいファイルを混用して実行しないことが重要です。

```
/etc/opt/ibm/sna
```

 ディレクトリーに保管したファイルを復元します。

さらに、スタートアップ・ファイル `/etc/rc.d/init.d/snastart` への変更を保管した場合は、78 ページの『**Communications Server for Linux** の使用可能化』の説明に従って、このファイルの新しいコピーにも同じ変更を行って、システム起動時に **Communications Server for Linux** が自動的に開始されないようにしてください。

ここで次のコマンドを使用して、**Communications Server for Linux** ソフトウェアを再び開始します。

```
/opt/ibm/sna/bin/sna start
```

Communications Server for Linux ライセンス・プログラムのインストール

30 ページの『前提ソフトウェアのインストール』に示されている前提ソフトウェアをインストールすると、**Communications Server for Linux** のインストールが可能になります。以下の手順を使用してください。

1. root 権限によりログインします。
2. 35 ページの『**Communications Server for Linux** のインストール』の手順に従って、**Communications Server for Linux** をインストールします。
3. SSL 機能をもつ TN サーバーまたは TN リダイレクターを使用するために GSKIT ソフトウェアをインストールした場合は、**Communications Server for Linux** をインストールしてから SSL ソフトウェアを構成する必要があります。詳しくは、39 ページの『TN サーバーまたは TN リダイレクターで使用するための SSL の構成』を参照してください。
4. 必要に応じて **Communications Server for Linux** 構成情報をカスタマイズします (75 ページの『第 8 章 **Communications Server for Linux** の構成と使用』または「*Communications Server for Linux* 管理ガイド」を参照してください)。
5. **Communications Server for Linux** を開始します。インストール後は、マシンがリブートされると自動的に開始されることに注意してください。

```
cd /
```

```
sna start
```

注:

1. **Communications Server for Linux** をインストールすると、ローカル・ノード上の SNA デバイス・ドライバーも自動的に再構成されます。小さいメモリのコンピューター (たとえば、64 MB) にインストールする場合は、**Communications Server for Linux** のインストール後にリブートしないと、SNA ノードを開始することはできません。
2. **Communications Server for Linux** のインストール後に、アダプターをインストールできます。アダプターを追加した後で **Communications Server for Linux** を再インストールする必要はありません。

Communications Server for Linux のインストール

1. インストール CD を Linux ワークステーションの適切なドライブに挿入します。
2. root 権限によりログインします。
3. **installibmcs** プログラムを CD から実行します。この作業を実行するために必要なコマンドの詳細については、Linux CD イメージの **README** ファイルを参照してください。

プログラムは、通常、次のプロンプトを出します。

- Communications Server for Linux ライセンス条件を読んで受諾したことの確認。
- Communications Server for Linux ドメインのマスター・サーバーになるサーバーの名前。Communications Server for Linux をスタンドアロン・ノードとして実行したい場合は、このパラメーターを指定しないでください。この場合、ノードはクライアント/サーバー機能をサポートしません。

これらのプロンプトをオーバーライドするには、以下のように、**installibmcs** コマンドに追加パラメーターを指定します。

4. Communications Server for Linux PDF 資料をインストールしたい場合は、**installibmcsdocs** プログラムを CD から実行します。
5. Communications Server for Linux 構成情報をカスタマイズします (75 ページの『第 8 章 Communications Server for Linux の構成と使用』または「*Communications Server for Linux 管理ガイド*」を参照してください)。

無人インストールを実行する必要がある場合は、**installibmcs** コマンドに追加パラメーターを指定して、Communications Server for Linux ライセンス条件の受諾を確認し、マスター・サーバーの名前を指定します。この場合、シェル・スクリプトは、追加情報のプロンプトなしで実行されます。次のコマンドを使用します。

```
./installibmcs license_accepted [ master_name ]
```

master_name は、マスター・サーバーの名前です。Communications Server for Linux をスタンドアロン・ノードとして実行したい場合は、このパラメーターを指定しないでください。この場合、ノードはクライアント/サーバー機能をサポートしません。

注: Communications Server for Linux をインストールすると、ローカル・ノード上の SNA デバイス・ドライバーも自動的に再構成されます。

Communications Server for Linux のオンライン資料

『Communications Server for Linux のインストール』のステップに従って、Communications Server for Linux インストール CD をロードし、root 権限でログインしてから、**installibmcsdocs** シェル・スクリプトを実行します。

```
./installibmcsdocs
```

Host Access Class Libraries (HACL)

Communications Server for Linux をインストールすると、HACL ファイルが自動的にインストールされます。ライブラリー・コードは `rpm` の下の `ibm-commserver-ecl` パッケージです。それらのファイル (README を含む) は `/opt/ibm/sna/ecl` にありますが、次のコマンドを実行して見つけることもできます。

```
rpm -ql ibm-commserver-ecl
```

WebSphere Application Server の構成

22 ページの『インストール要件』で説明されているように、Remote API Client が HTTPS を使用して Communications Server for Linux サーバーに接続するようなクライアント/サーバー・システムを実行する予定の場合、それらのクライアントからサーバーへの HTTPS アクセスを提供するために、WebSphere Application Server を実行するコンピューターが必要になります。

このセクションでは、Communications Server for Linux と共に使用するための WebSphere のセットアップ方法について説明します。

- クライアントに対して提示される WebSphere サーバー上のセキュア証明書のセットアップ
- WebSphere Application Server が Communications Server for Linux と共に機能するための構成
- WebSphere サーバーへのサーバー構成ファイルのインストール

WebSphere Application Server にアクセスするため、各 Remote API Client 上にクライアント・セキュリティー証明書とクライアント・ネットワーク・データ・ファイルをセットアップする必要もあります。詳しくは、該当するクライアント・タイプのインストールに関する章を参照してください。

WebSphere Application Server のセキュア証明書のセットアップ

WebSphere Application Server 資料で、サーバー上のセキュア証明書のセットアップに関する手順を参照してください。これは、HTTPS を使用して接続しようとするときの認証プロセス中に Remote API Client に提示される、サーバーの証明書です。

クライアント認証を施行するように WebSphere を構成することをお勧めします。詳しくは、WebSphere Application Server 資料を参照してください。この構成が意味するのは、WebSphere は、認証プロセス中に Remote API Client からのセキュリティー証明書を要求し、クライアントの証明書の認証性を検証できた場合にのみ Remote API Client から送られてくる接続を受け入れるということです。

WebSphere Application Server の構成

Communications Server for Linux と共に作動するように WebSphere Application Server を構成するには、以下のステップを実行します。詳しくは、WebSphere Application Server 資料を参照してください。

1. 2 つのファイル **snahttpsrv.ear** および **snahttpsrv.cfg** を、Remote API Client インストール CD 上の **ibm-commserver-https** ディレクトリーから、WebSphere 管理コンソールが稼働しているコンピューター上のディレクトリー、またはこのコンピューターからアクセス可能なネットワーク・ディレクトリーに、コピーするか FTP で転送します。

管理コンソールが Windows 上で稼働している場合、これらのファイルには CD から直接アクセスできるので、コピーする必要はありません。Remote API Client インストール CD を Windows コンピューターの CD ドライブに挿入するだけです。

2. WebSphere 管理コンソールを開始します。
3. WebSphere 資料の説明に従って、SSL セキュア接続を介してのみアクセス可能な仮想ホストを作成します。この仮想ホストは、SNA HTTPS 接続を管理する Java プラグインで使用されます。
4. メニューバーから、「アプリケーション」、「新規アプリケーションのインストール」を選択します。
5. **snahttpsrv.ear** ファイルの場所を指定します。「次へ」ボタンを選択します。
6. 最初の 2 画面で、仮想ホスト名を指定するようプロンプトが出されたら、HTTPS 用にセットアップした仮想ホストの名前を入力します。その他のすべてのパラメーターについては、何らかの特別な WebSphere 構成を使用する必要がない場合はデフォルト・オプションを受け入れます。以降のダイアログでは「次へ」ボタンを選択し、「完了」ボタンが表示されたら「完了」ボタンを選択します。そうすると、「**アプリケーションは正常にインストールされました**」というメッセージが画面に表示されます。
7. 「マスター構成に保管する」をクリックし、「保管」ボタンをクリックします。
8. メニューバーから、「アプリケーション」、「エンタープライズ・アプリケーション」を選択します。
9. アプリケーション・リストで **SnaHttpTransport** を見つけ、その横のチェック・ボックスをクリックし、このアプリケーションを開始するため「開始」ボタンをクリックします。（これ以降は、WebSphere Application Server が開始されると、このアプリケーションは自動的に開始されます。）
10. メニューバーから、「環境」、「Web サーバー・プラグインの更新」を選択し、「OK」ボタンをクリックします。これで WebSphere 構成が更新されます。

サーバー構成ファイルのインストール

WebSphere Application Server が Communications Server for Linux と共に作動するには、HTTPS を使用してアクセスされる Communications Server for Linux サーバーのリストが必要です。以下のステップを使用して、このリストを作成し、インストールしてください。

1. WebSphere 管理コンソールのメニューバーで、「環境」、「WebSphere 変数の管理」を選択します。

- このリストで **USER_INSTALL_ROOT** 変数を探し、その値 (WebSphere サーバー上のディレクトリーのパスです) をメモします。環境変数のリストは、複数のページにわたることがあるので、その場合は「次へ」ボタンを使用してリストをスクロールします。
- snahttpsrv.cfg** ファイルを、36 ページの『WebSphere Application Server の構成』で保管した場所から (またはインストール CD から) **USER_INSTALL_ROOT** 変数で指定されたディレクトリーへコピーします。次に、テキスト・エディターを使用してこのファイルを編集し、HTTPS を使用して Remote API Client によってアクセス可能な Communications Server for Linux サーバーのリストを組み込みます。ファイル中では、各サーバーはそれぞれ別の行に以下の形式で指定する必要があります。

```
server=servername.domainname.com
```

インストール後の作業

この項では、Communications Server for Linux をインストールした後に必要な保守作業の実行方法について説明します。

クライアント/サーバー・オペレーション

インストール後、Communications Server for Linux は最初はスタンドアロン・サーバー (すべてのコンポーネントが単一の Linux システム上にある) として作動します。クライアント/サーバー・ドメイン内のサーバーとして稼働させたい場合の手順については、「*Communications Server for Linux 管理ガイド*」の Communications Server for Linux クライアント/サーバー・システムの管理に関する章を参照してください。

PDF 資料の表示

この製品のインストール・メディアには、マニュアルが Portable Document Format (PDF) 形式で収録されています。ソフトコピー形式では、関連情報へのハイパーテキスト・リンクを使用して情報を簡単に検索、表示、印刷することができます。また、PDF ビューアーは多くの異なるプラットフォームで使用可能なので、サイトでのライブラリーの共有も簡単に行うことができます。

製品のインストール時に PDF マニュアルのインストールを選択すると、それらのマニュアルはディレクトリー **/opt/ibm/sna/docs** にインストールされます。それらのマニュアルは、Communications Server for Linux インストール・メディアのディレクトリー **/DOCS** にも収録されています。

PDF マニュアルは、任意の PDF ビューアー (たとえば、Windows 上の Adobe Acrobat、または Intel Linux 上の **xpdf**) を使用して読むことができます。

現行リリース情報の検討

この製品の **README** ファイルの最新の更新情報 (**/opt/ibm/sna** ディレクトリーにある) には、Communications Server for Linux ライブラリーの発表以後に製品に対して行われたすべての変更に関する情報が記載されています。このファイルは、

Communications Server for Linux インストール・メディアのルート・ディレクトリにも収録されています。製品の更新を受け取ったら、必ず **README** ファイルをお読みください。

TN サーバーまたは TN リダイレクターで使用するための SSL の構成

SSL 機能をもつ TN サーバーまたは TN リダイレクターを使用するために GSKIT ソフトウェアをインストールした場合は、Communications Server for Linux をインストールしてから SSL ソフトウェアを構成する必要があります。

SSL ソフトウェアには以下の 2 つのコンポーネントが必要です。

- データ暗号化および暗号化解除を実行できるようにするため、鍵ペアが必要です。
- サーバー認証を可能にするため、証明書が必要です。

証明書および鍵ペアは、鍵リング・データベース内に単一のレコードを作成します。このレコードは、TN サーバーまたは TN リダイレクターを実行している Communications Server for Linux サーバーに保管されます。Communications Server for Linux は、このデータベースを使用して、SSL をインプリメントします。

鍵リング・データベースを管理するには、Linux コマンド・プロンプトで以下のコマンドを入力します。

snakeyman

snakeyman コマンドは、Java プログラムを起動します。詳しい指示については、このプログラムと共に提供されるヘルプを参照してください。

データベース内の各レコードは、ラベルと呼ばれる固有名で識別されます。異なる TN サーバー・セッションまたは TN リダイレクター・セッションで使用するレコードが複数ある場合には、データベースを設定するときに割り当てるラベルを書き留めておく必要があります。これらのラベルは、各セッションで使用されるレコードを識別するために使用されます。レコードの 1 つをデフォルトにすることもできます。そうすると、別のレコードのラベルを明示的に指定しなければ、セッションはこのレコードを使用します。

snakeyman を使用してサーバー証明書を更新した後は、**snakeyman** プログラムを終了し、更新された証明書を使用するために Communications Server for Linux ノードを停止して再始動する必要があります。ノードの停止と再始動には、以下のコマンドを使用します。

```
snaadmin term_node
snaadmin init_node
```

Communications Server for Linux 構成ファイルのバックアップ

Communications Server for Linux は、ユーザーがノード・ファイル、ドメイン・ファイル、および TP 構成ファイルに影響を与えるような変更を行う Communications Server for Linux 管理ツールを使用して) たびに、これらのファイルのバックアップ

インストール後の作業

を自動的に作成します。たとえば、ノード構成ファイル (`sna_node.cfg`) に影響を与える変更を行うと、Communications Server for Linux は、`sna_node.bk n` という名前のバックアップ・ファイルを作成します。ここで、*n* は 1 または 2 のどちらかです。

- 初めてファイルを変更すると、既存の構成は `sna_node.bk1` に保管されます。
- 2 度目にファイルを変更すると、既存の構成は `sna_node.bk2` に保管され、`sna_node.bk1` は変更されません。
- 3 度目以降にファイルを変更すると、`sna_node.bk1` は破棄され、`sna_node.bk2` は `sna_node.bk1` に名前が変更されて、既存の構成が `sna_node.bk2` に保管されます。

このプロセスは、ノード構成ファイルのバックアップ・ファイルは、常に、最大でも 2 つしかないことを意味します。他のバックアップ・ファイルのファイル名拡張子も、同じプロセスで生成されます。

次の状況では、自動バックアップの他に、データの消失を防ぐために構成ファイルのバックアップを各自作成しておく必要があります。

- 新しいレベルの Linux オペレーティング・システムをインストールする前
- 新しいリリースの Communications Server for Linux をインストールする前
- 新しい構成を作成したあと

構成ファイルは、以下のコマンドを使用してバックアップできます。

```
cd /etc/opt/ibm/sna
tar cvf Devicename sna_node.cfg sna.net sna_tps sna_domn.cfg
ibmcs.*
```

Communications Server for Linux 構成ファイルのバックアップ・コピーの復元

バックアップした Communications Server for Linux 構成ファイルを復元するには (39 ページの『Communications Server for Linux 構成ファイルのバックアップ』を参照)、次のようにします。

1. Communications Server for Linux が活動状態でないことを確認してください。アクティブかどうかを判別するには、次のコマンドを入力します。

```
snaadmin status_node
```

Communications Server for Linux が活動状態である場合には、このコマンドは、ローカル・ノードの状態に関する情報を表示します。そうでない場合には、Communications Server for Linux が非活動状態であることを示すメッセージを表示します。

Communications Server for Linux が活動状態である場合は、次のコマンドを入力して非活動化します。

```
sna stop
```

2. 以下のコマンドを入力します。

```
cd /etc/opt/ibm/sna
tar xvf Devicename
```

上記コマンドの *Devicename* は、ファイルをバックアップしたときに使用したデバイスのパスとファイル名です。

このコマンドは、`/etc/opt/ibm/sna` ディレクトリーにある同じ名前の既存の構成ファイルを上書きします。

構成ファイルの再初期化

Communications Server for Linux 構成ファイルが誤って変更されたために、ファイル内の情報を使用できなくなった場合は、構成ファイルを再初期化する必要があります。構成ファイルを再初期化すると、Communications Server for Linux を新たにインストールした状態で再構成できます。この操作は、構成情報を修復できない場合にのみ行ってください。

注: 有効なバックアップ構成ファイルがある場合は、まずこれらのバックアップ・ファイルを `/etc/opt/ibm/sna` ディレクトリーにコピーし、`sna start` コマンドを使用してノードを初期化する際にこれらのファイルを使用します。

再初期化できる構成ファイルを次に示します。

- ノード構成ファイル `sna_node.cfg`
- ドメイン構成ファイル `sna_domn.cfg`
- TP 構成ファイル `sna_tps`
- SSL 鍵リング・データベース・ファイルとパスワード・スタッシュ・ファイル

構成ファイルを再初期化する手順は次のとおりです。

1. 管理プログラムが活動状態である場合は管理プログラムを終了し、次のコマンドを発行して Communications Server for Linux を使用不可にします。

```
sna stop
```

2. 再初期化するファイルをすべて別の場所へコピーすることによって、既存の構成ファイルをバックアップします。
3. 再初期化するファイルを削除します。
4. ドメイン構成ファイルを削除した場合は、以下のコマンドを発行して、そのデータベース・ファイルを再作成します (Communications Server for Linux とともに配送された空のドメイン構成ファイルからコピーして)。

```
cp -p /opt/ibm/sna/samples/empty.cfg /etc/opt/ibm/sna/sna_domn.cfg
```

このコマンドは、Communications Server for Linux の始動に必要な新規のドメイン構成ファイルを作成します。

5. SSL 鍵リング・データベース・ファイルを削除した場合は、以下のコマンドを発行して、そのデータベース・ファイルを再作成します (Communications Server for Linux とともに送達されたサンプル・ファイルからコピーして)。

```
cp -p /opt/ibm/sna/samples/ibmcs.* /etc/opt/ibm/sna
```

6. 次のコマンドを入力して、Communications Server for Linux を再始動します。

インストール後の作業

sna start

7. Motif 管理プログラムを開始します。

xsnaadmin &

この管理プログラムは、**sna_node.cfg** ファイルが存在しない場合、ノードを構成するように指示するプロンプトを出します。75 ページの『第 8 章 Communications Server for Linux の構成と使用』または「*Communications Server for Linux 管理ガイド*」に記述されているように、ノードおよび他のリソースの構成を続けることができます。

有効な **sna_node.cfg** ファイルを使用した場合は、この新規構成ファイルがノードの初期化に使用されます。

Communications Server for Linux のアンインストール

Communications Server for Linux 製品は、いつでもアンインストールできます。以下の手順を使用してください。

1. root 権限によりログインします。
2. Communications Server for Linux が活動状態でないことを確認してください。アクティブかどうかを判別するには、次のコマンドを入力します。

snaadmin status_node

Communications Server for Linux が活動状態である場合には、このコマンドは、ローカル・ノードの状態に関する情報を表示します。そうでない場合には、Communications Server for Linux が非活動状態であることを示すメッセージを表示します。

Communications Server for Linux が活動状態である場合は、次のコマンドを入力して非活動化します。

sna stop

3. 以下の手順を使用して、Communications Server for Linux パッケージと関連ソフトウェア・パッケージを除去します。

```
rpm -e ibm-commserver-docs
```

```
rpm -e ibm-commserver-ecl
```

```
rpm -e ibm-commserver
```

```
/sbin/shutdown -r now
```

第 4 章 Linux での IBM Remote API Client のインストール

この章では、Linux での IBM Remote API Client のインストールの方法について説明します。このインストールにより、Linux ワークステーションは、SNA スタックがフルにインストールされていなくても SNA アプリケーションを実行できるようになります。Linux の Remote API Client は、TCP/IP ネットワークを使用して 1 つ以上の Communications Server for Linux サーバー (または CS/AIX サーバー) に接続できます (ただし、同時に両方のサーバーに接続することはできません)。(CS Linux サーバーは CS/AIX サーバーと同じドメインでは操作できません。)

この章では、32 ビット Intel (i686)、64 ビット AMD64/Intel EM64T (x86_64)、および pSeries (ppc64) コンピューターで実行される IBM Remote API Client について説明します。IBM Remote API Client を System z コンピューター (s390 / s390x) にインストールする場合は、49 ページの『第 5 章 Linux for System z への IBM Remote API Clients のインストール』を参照してください。

インストール・プログラムおよび関連ファイル (IBM Remote API Client README ファイルを含む) は、インストール CD の、以下のクライアント・タイプに該当するディレクトリーに入っています。

クライアント・タイプ	CD のディレクトリー
32 ビット Intel (i686)	<code>/ibm-commserver-clients/linux</code>
64 ビット AMD64/Intel EM64T (x86_64)	<code>/ibm-commserver-clients/linux-x86_64</code>
pSeries (ppc64)	<code>/ibm-commserver-clients/linux-ppc64</code>

このソフトウェアをインストールする前に、IBM Remote API Client README ファイルを読むことをお勧めします。

ハードウェア要件とソフトウェア要件

ハードウェア要件

IBM Remote API Client は、以下の Linux ディストリビューションの 1 つがサポートするコンピューターを必要とします。

コマンド `uname -m` を使用して、ターゲット・コンピューターの CPU クラスを確認してください。次の表には、クライアント・タイプごとに適切なハードウェアと、そのハードウェアに対する `uname -m` からの応答が示されています。

クライアント・タイプ	ハードウェア	uname 応答
32 ビット Intel	Pentium II またはそれ以降の 32 ビット Intel システム、または Opteron ベースのシステム	i686

クライアント・タイプ	ハードウェア	uname 応答
64 ビット AMD64/Intel EM64T	x86_64 (AMD64 または Intel EM64T) システム	x86_64
pSeries	pSeries POWER5 または OpenPower システム	ppc64

Linux オペレーティング・システムのバージョン

現行バージョンの IBM Remote API Client は、以下のバージョンの Linux オペレーティング・システムでテスト済みです。これ以外の Linux ディストリビューションでも十分に稼働する場合があります。

- RedHat Enterprise Linux 3 (RHEL3): i686 のみ
- RedHat Enterprise Linux 4 (RHEL4)
- SUSE Linux Enterprise Server 8 (SLES8): i686 のみ
- SUSE Linux Enterprise Server 9 (SLES9)
- SUSE Linux Enterprise Server 10 (SLES10)

必要なオプション・パッケージの詳細については、インストール CD にある **README** ファイルを参照してください。

Java

Java CPI-C API を使用する場合、Java ソフトウェアが必要になります。詳しくは、インストール CD にある **README** ファイルを参照してください。

GSKIT

クライアントが HTTPS を使用して Communications Server for Linux サーバーに接続する場合、WebSphere サーバーを通してサーバーに HTTPS アクセスできるようにするため、GSKIT ソフトウェアが必要です。GSKIT ソフトウェアは、インストール CD に収録されていますが、それをインストールするために、オプションの Linux オペレーティング・システム・パッケージがいくつか必要な場合があります。必要なオプション・パッケージについての詳細は、インストール CD の **README** ファイルを参照してください。

この章の後で説明するように、クライアントのインストール・プロセスを実行するときに前提条件パッケージがすべてインストールされている場合は、GSKIT ソフトウェアはそのプロセスの一部として自動的にインストールされます。それ以外の場合は、後でインストールできます。

製品インストール詳細の表示

既にインストール済みの Remote API Client および関連ソフトウェア・パッケージに関する詳細を表示できます。インストール済みのパッケージをすべてリストするには、以下のコマンドを使用します。

```
rpm -q -a
```

ある特定のパッケージの詳細を表示するには、以下のコマンドを使用します。

```
rpm -q -i packagename
```

packagename は、インストール済みパッケージのベース名 (たとえば、**ibm-commsserver-client**) です。

言語環境変数の設定

LANG 変数を変更して、使用したい言語を指示するには、以下のコマンドを使用します。

```
export LANG=language
```

language は、使用したい言語の ID に置き換えてください。以下のいずれかを使用できます。

ID	言語
en_US	英語 (US)
ja_JP	日本語 (PC)
de_DE	ドイツ語
es_ES	スペイン語
fr_FR	フランス語
ko_KR	韓国語
pt_BR	ポルトガル語
zh_CN	中国語 (簡体字 EUC)
zh_TW	中国語 (繁体字)

Linux での Remote API Client のインストール

前提条件ソフトウェアのインストールが完了したら、IBM Remote API Client をインストールすることができます。

前のレベルの IBM Remote API Client が既にインストールされている場合、新しいレベルをインストールする前に、48 ページの『Linux 上の Remote API Client のアンインストール』に説明されているステップに従ってそれを除去してください。新規インストールで使用できるよう、構成情報はそのまま残されます。

1. root 権限によりログインします。
2. CD をマウントし、それを現行ディレクトリーにします。

```
mount /dev/cdrom
cd /media/cdrom
```

RHEL3 (32 ビット Intel クライアントのみ) の場合は、**/media/cdrom** を **/mnt/cdrom** で置き換えてください。それ以外の場合は、DVD ドライブがあるとディレクトリー名 **/media/cdrom** は別の名前になることがあります。コマンド **df** を使用すると、Linux が CD をどこにマウントしたのかを確認できます。

3. CD の適切なサブディレクトリーに移動し、シェル・スクリプトを実行してクライアントをインストールします。次の例は、32 ビット Intel (i686) クライアント用の **/linux** サブディレクトリーを示しています。必要なら **/linux-x86_64** または **/linux-ppc64** に置き換えてください。

Linux での Remote API Client のインストール

```
cd ibm-commserver-clients/linux
./installibmcscli
```

このシェル・スクリプトは、一定の前提条件をテストし、満たされていない場合は警告メッセージを発行します。ご使用条件を読んで同意するように指示するプロンプトが出され、その後、スクリプトは RPM をインストールします。適切な前提条件が既にインストール済みであれば、スクリプトは GSKIT ソフトウェアもインストールします。

4. IBM Remote API Client バイナリー・ディレクトリーを PATH に追加します。これを自動的に行うようにプロファイルを変更することもできます。

```
export PATH="$PATH:/opt/ibm/sna/bin"
```

```
export LD_LIBRARY_PATH=/usr/lib:/opt/ibm/sna/lib
```

```
export LD_RUN_PATH=/usr/lib:/opt/ibm/sna/lib
```

Java CPI-C アプリケーション用に、以下の環境変数も設定する必要があります。

```
export CLASSPATH=$CLASSPATH:/opt/ibm/sna/java/cpic.jar
```

一部のアプリケーション用に、LD_PRELOAD 環境変数も設定する必要がある場合もありますが、プロファイル内での一括変更でこれを行うべきではありません。

```
export LD_PRELOAD=/usr/lib/libpLiS.so
```

5. IBM Remote API Client を開始します。インストール後は、マシンがリポートされたら自動的に開始されるようになります。開始するときには、まだ CD のディレクトリーにいないことがないようにしてください。

```
cd /
sna start
```

注: IBM Remote API Client が HTTPS を使用してサーバーに接続できるようにするには、GSKIT キー・マネージャー・プログラムを使用して、セキュリティー証明書構成をクライアントにセットアップする必要があります。詳しくは、『GSKIT を使用した HTTPS セキュリティー証明書のセットアップ』を参照してください。

また、クライアント・ネットワーク・データ・ファイルを更新して、クライアントが接続可能な Communications Server for Linux サーバーと、HTTPS サポートを提供する WebSphere サーバーの名前を指定する必要があります。詳しくは、「*Communications Server for Linux 管理ガイド*」の Remote API Client の管理に関するセクションを参照してください。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

クライアントが HTTPS を使用して Communications Server for Linux サーバーに接続する場合、クライアントには GSKIT キー・マネージャー・ソフトウェアがインストール済みでなければなりません。これは、インストール CD の **README** ファイルに説明されている、必要な Linux オペレーティング・システム前提条件がイ

GSKIT を使用した HTTPS セキュリティ証明書セットアップ

インストール済みであれば、通常はクライアントのインストールの一部として実施されます。GSKIT がクライアントのインストールの一部としてインストールされなかったが、現在は前提条件をインストール済みである場合、以下のステップを使用して GSKIT ソフトウェアをインストールできます。

1. root 権限によりログインします。
2. CD をマウントし、それを現行ディレクトリーにします。

```
mount /dev/cdrom
cd /media/cdrom
```

RHEL3 (32 ビット Intel クライアントのみ) の場合は、`/media/cdrom` を `/mnt/cdrom` で置き換えてください。それ以外の場合は、DVD ドライブがあるとディレクトリー名 `/media/cdrom` は別の名前になることがあります。コマンド `df` を使用すると、Linux が CD をどこにマウントしたのかを確認できます。

3. CD の適切なサブディレクトリーに移動し、シェル・スクリプトを実行して GSKIT ソフトウェアをインストールします。次の例は、32 ビット Intel (i686) クライアント用の `/linux` サブディレクトリーを示しています。必要なら `/linux-x86_64` または `/linux-ppc64` に置き換えてください。

```
cd ibm-commsserver-clients/linux
./installgskit
```

IBM Remote API Client が HTTPS を使用してサーバーに接続できるようにするには、GSKIT キー・マネージャー・プログラムを使用して、セキュリティ証明書構成をクライアントにセットアップする必要があります。次のステップを実行してください。

1. 以下のコマンドを使用して GSKIT キー・マネージャーを実行します。

```
/opt/ibm/sna/bin/snakeyman
```

キー・マネージャー・ユーザー・インターフェース内から、鍵データベース・ファイル `/etc/opt/ibm/sna/ibmcs.kdb` を開きます。このファイルは CMS 形式です。

2. 鍵データベースの初期パスワードは `ibmcs` です。セキュリティ証明書をセットアップする前に、構成を保護するため、このパスワードを変更しなければなりません。パスワードを変更するためのダイアログでは、「Stash the password to a file? (パスワードをファイルへ stash する?)」というチェック・ボックスにマークを付ける必要があります。そうすると、新規パスワードが保管されるので、クライアントは鍵データベースをオープンできます。
3. Web サーバーのセキュリティ証明書に署名するのに使用された認証局 (CA) 証明書のコピーを入手し、それを鍵データベースにインストールします。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Signer Certificates (署名者証明書)」を選択し、「Add (追加)」をクリックします。
4. WebSphere サーバーがクライアント・セキュリティ証明書を必要とするように構成されている場合、クライアントは、自身の証明書が Web サーバーのセキュリティ証明書データベース内にある CA によって発行された証明書を持っている必要があります。新規証明書を要求するには、以下を実行します。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

- a. キー・マネージャー・ユーザー・インターフェースから、「Create (作成)」、「New Certificate Request (新規証明書要求)」を選択し、要求される詳細情報を入力します。
- b. 証明書を保管し、それをファイルに抽出し、CA に送信します。
- c. 証明書が発行されたら、それを Web サーバーのデータベースに保管します。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Personal Certificates (個人証明書)」を選択し、「Receive (受信)」をクリックします。

個人的な内部テスト用の一時的な手段として、CA から証明書を入手する代わりに、自己署名クライアント証明書を作成することもできます。ただし、これでは必要なレベルのセキュリティは確保されないため、実際のシステムで使用するべきではありません。自己署名証明書を作成するには、以下を実行します。

- a. キー・マネージャー・ユーザー・インターフェースから、「Create (作成)」、「New Self-Signed Certificate (新規自己署名証明書)」を選択し、要求される詳細情報を入力します。
 - b. 証明書を保管し、それをファイルに抽出します。
 - c. その証明書ファイルを Web サーバーのデータベースに保管します。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Personal Certificates (個人証明書)」を選択し、「Receive (受信)」をクリックします。
5. 証明書の構成が完了したら、GSKIT キー・マネージャーを終了します。

Linux 上の Remote API Client のアンインストール

Linux 上の Remote API Client は、以下のコマンドを使用してアンインストールできます。

```
/opt/ibm/sna/bin/sna stop  
rpm -e ibm-commserver-ptf  
rpm -e ibm-commserver-docs  
rpm -e ibm-commserver-ecl  
rpm -e ibm-commserver-cli  
rpm -e ibm-commserver  
rpm -e gsk7bas  
/sbin/shutdown -r now
```

これらのコマンドにリストされているパッケージのすべてが、どのシステムにもインストールされるわけではありません。

Linux 上の IBM Remote API Client をアンインストールしても、カスタマイズされた構成情報は、後のインストールで使用できるように残されます。

第 5 章 Linux for System z への IBM Remote API Clients のインストール

この章では、Linux への IBM Remote API Client のインストールについて説明します。このインストールにより、System z メインフレームは完全な SNA スタックがインストールされていなくても SNA アプリケーションを実行できるようになります。Linux for System z 上の Remote API Client は、TCP/IP ネットワークを使用して、1 つ以上の Communications Server for Linux サーバー (または CS/AIX サーバー) に接続できます。

このソフトウェアをインストールする前に、IBM Remote API Client README ファイルを読むことをお勧めします。このファイルは、インストール CD の `/ibm-commsserver-clients/linux-systemz` ディレクトリーにあります。

ハードウェア要件とソフトウェア要件

ハードウェア要件

IBM Remote API Client は、『Linux オペレーティング・システムのバージョン』にリストされている Linux ディストリビューションの 1 つでサポートされている 32 ビットまたは 64 ビットの System z システムを必要とします。

CPU クラスを検証するには、コマンド `uname -m` を使用します。このコマンドが、31 ビット環境を示す `s390`、または 64 ビット環境を示す `s390x` を報告しなければなりません。

Linux オペレーティング・システムのバージョン

現行バージョンの IBM Remote API Client は、以下のバージョンの Linux オペレーティング・システムでテスト済みです。これ以外の Linux ディストリビューションでも十分に稼働する場合があります。

- RedHat Enterprise Linux 3 for S/390® (RHEL3-s390)
- RedHat Enterprise Linux 3 for zSeries® (RHEL3-s390x)
- RedHat Enterprise Linux 4 for S/390 (RHEL4-s390)
- RedHat Enterprise Linux 4 for zSeries (RHEL4-s390x)
- SUSE Linux Enterprise Server 8 for IBM Mainframe (SLES8-s390*)
- SUSE Linux Enterprise Server 9 for IBM Mainframe (SLES9-s390*)
- SUSE Linux Enterprise Server 10 for IBM Mainframe (SLES10-s390x)

必要なオプション・パッケージの詳細については、インストール CD にある **README** ファイルを参照してください。

Java

Java CPI-C API を使用する場合、Java ソフトウェアが必要になります。詳しくは、インストール CD にある **README** ファイルを参照してください。

GSKIT

クライアントが HTTPS を使用して Communications Server for Linux サーバーに接続する場合、WebSphere サーバーを通してサーバーに HTTPS アクセスできるようにするため、GSKIT ソフトウェアが必要です。GSKIT ソフトウェアは、インストール CD に収録されていますが、それをインストールするために、オプションの Linux オペレーティング・システム・パッケージがいくつか必要な場合があります。必要なオプション・パッケージについての詳細は、インストール CD の **/ibm-commsrver-clients/linux-systemz** ディレクトリーにある **README** ファイルを参照してください。

この章の後で説明するように、クライアントのインストール・プロセスを実行するときに前提条件パッケージがすべてインストールされている場合は、GSKIT ソフトウェアはそのプロセスの一部として自動的にインストールされます。それ以外の場合は、後でインストールできます。

製品インストール詳細の表示

既にインストール済みの Remote API Client および関連ソフトウェア・パッケージに関する詳細を表示できます。インストール済みのパッケージをすべてリストするには、以下のコマンドを使用します。

```
rpm -q -a
```

ある特定のパッケージの詳細を表示するには、以下のコマンドを使用します。

```
rpm -q -i packagename
```

packagename は、インストール済みパッケージのベース名 (たとえば、**ibm-commsrver-client**) です。

言語環境変数の設定

LANG 変数を変更して、使用したい言語を指示するには、以下のコマンドを使用します。

```
export LANG=language
```

language は、使用したい言語の ID に置き換えてください。以下のいずれかを使用できます。

ID	言語
en_US	英語 (US)
ja_JP	日本語 (PC)
de_DE	ドイツ語
es_ES	スペイン語
fr_FR	フランス語
ko_KR	韓国語

ID	言語
pt_BR	ポルトガル語
zh_CN	中国語 (簡体字 EUC)
zh_TW	中国語 (繁体字)

Linux for System z への Remote API Client のインストール

前提条件ソフトウェアのインストールが完了したら、IBM Remote API Client をインストールすることができます。

前のレベルの IBM Remote API Client が既にインストールされている場合、新しいレベルをインストールする前に、54 ページの『Linux for System z での Remote API Client のアンインストール』に説明されているステップに従ってそれを除去してください。新規インストールで使用できるよう、構成情報はそのまま残されません。

1. **ibm-commserver-client-6.3.0.1-s390.tgz** ファイルを CD-ROM の **/ibm-commserver-clients/linux-systemz** ディレクトリーから Linux System z システムにコピーまたは FTP で転送します。コピーまたは FTP でのファイル転送を行う際は、バイナリー・モードを使用してください。
2. Linux System z システムに **root** としてログインします。
3. 上記 **tar** ファイルを解凍またはアンパックして、空の一時ディレクトリーに入れます。

```
mkdir /tmp/ibmcs
```

```
cd /tmp/ibmcs
```

```
tar -xzf ibm-commserver-client-6.3.0.1-s390.tgz
```

4. **installibmcscli** シェル・スクリプトを実行します。

```
./installibmcscli
```

このシェル・スクリプトは、一定の前提条件をテストし、満たされていない場合は警告メッセージを発行します。このシェル・スクリプトは、Communications Server for Linux ライセンス条件を読んで受諾したことを確認するよう要求するプロンプトも出します。後述のように、**installibmcscli** コマンドにパラメーターを追加指定すれば、このプロンプトを無効にすることができます。プロンプトへの応答が終わると、シェル・スクリプトは **rpm** パッケージをインストールします。適切な前提条件が既にインストール済みであれば、スクリプトは **GSKIT** ソフトウェアもインストールします。

5. IBM Remote API Client バイナリー・ディレクトリーを **PATH** に追加します。これを自動的に行うようにプロファイルを変更することもできます。

```
export PATH="$PATH:/opt/ibm/sna/bin"
```

```
export LD_LIBRARY_PATH=/usr/lib:/opt/ibm/sna/lib
```

```
export LD_RUN_PATH=/usr/lib:/opt/ibm/sna/lib
```

Linux for System z への Remote API Client のインストール

64 ビット・アプリケーションを実行する予定がある場合は、次のオプションを使用します。

```
export LD_LIBRARY_PATH=/usr/lib64:/opt/ibm/sna/lib64
```

```
export LD_RUN_PATH=/usr/lib64:/opt/ibm/sna/lib64
```

Java CPI-C アプリケーション用に、以下の環境変数も設定する必要があります。

```
export CLASSPATH=$CLASSPATH:/opt/ibm/sna/java/cpic.jar
```

一部のアプリケーション用に、LD_PRELOAD 環境変数も設定する必要がある場合もありますが、プロファイル内での一括変更でこれを行うべきではありません。

```
export LD_PRELOAD=/usr/lib/libpLiS.so
```

6. IBM Remote API Client を開始します。インストール後は、マシンがリブートされたら自動的に開始されるようになります。開始するときには、まだ CD のディレクトリーにいないことがないようにしてください。

```
cd /
```

```
sna start
```

7. インストールが完了したら、インストール・プロセス中に作成された **tgz** ファイルと一時ディレクトリーは削除してもかまいません。

注: IBM Remote API Client が HTTPS を使用してサーバーに接続できるようにするには、GSKIT キー・マネージャー・プログラムを使用して、セキュリティー証明書構成をクライアントにセットアップする必要があります。詳しくは、『GSKIT を使用した HTTPS セキュリティー証明書のセットアップ』を参照してください。

また、クライアント・ネットワーク・データ・ファイルを更新して、クライアントが接続可能な Communications Server for Linux サーバーと、HTTPS サポートを提供する WebSphere サーバーの名前を指定する必要があります。詳しくは、「*Communications Server for Linux 管理ガイド*」の Remote API Client の管理に関するセクションを参照してください。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

クライアントが HTTPS を使用して Communications Server for Linux サーバーに接続する場合、クライアントには GSKIT キー・マネージャー・ソフトウェアがインストール済みでなければなりません。これは、インストール CD の **README** ファイルに説明されている、必要な Linux オペレーティング・システム前提条件がインストール済みであれば、通常はクライアントのインストールの一部として実施されます。GSKIT がクライアントのインストールの一部としてインストールされなかったが、現在は前提条件をインストール済みである場合、以下のステップを使用して GSKIT ソフトウェアをインストールできます。

1. **ibm-commserver-client-6.3.0.1-s390.tgz** ファイルを CD-ROM の **/ibm-commserver-clients/linux-systemz** ディレクトリーから Linux System z シ

GSKIT を使用した HTTPS セキュリティ証明書セットアップ

システムにコピーまたは FTP で転送します。コピーまたは FTP でのファイル転送を行う際は、バイナリー・モードを使用してください。

- Linux System z システムに root としてログインします。
- 上記 tar ファイルを解凍またはアンパックして、空の一時ディレクトリーに入れます。

```
mkdir /tmp/ibmcs
```

```
cd /tmp/ibmcs
```

```
tar -xzf ibm-commsserver-client-6.3.0.1-s390.tgz
```

- installgskit** シェル・スクリプトを実行します。

```
./installgskit
```

- インストールが完了したら、インストール・プロセス中に作成された **tgz** ファイルと一時ディレクトリーは削除してもかまいません。

IBM Remote API Client が HTTPS を使用してサーバーに接続できるようにするには、GSKIT キー・マネージャー・プログラムを使用して、セキュリティ証明書構成をクライアントにセットアップする必要があります。次のステップを実行してください。

- 以下のコマンドを使用して GSKIT キー・マネージャーを実行します。

```
/opt/ibm/sna/bin/snakeyman
```

キー・マネージャー・ユーザー・インターフェース内から、鍵データベース・ファイル **/etc/opt/ibm/sna/ibmcs.kdb** を開きます。このファイルは CMS 形式です。

- 鍵データベースの初期パスワードは **ibmcs** です。セキュリティ証明書をセットアップする前に、構成を保護するため、このパスワードを変更しなければなりません。パスワードを変更するためのダイアログでは、「Stash the password to a file? (パスワードをファイルへ stash する?)」というチェック・ボックスにマークを付ける必要があります。そうすると、新規パスワードが保管されるので、クライアントは鍵データベースをオープンできます。
- Web サーバーのセキュリティ証明書に署名するのに使用された認証局 (CA) 証明書のコピーを入手し、それを鍵データベースにインストールします。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Signer Certificates (署名者証明書)」を選択し、「Add (追加)」をクリックします。
- WebSphere サーバーがクライアント・セキュリティ証明書を必要とするように構成されている場合、クライアントは、自身の証明書が Web サーバーのセキュリティ証明書データベース内にある CA によって発行された証明書を持っている必要があります。新規証明書を要求するには、以下を実行します。
 - キー・マネージャー・ユーザー・インターフェースから、「Create (作成)」、「New Certificate Request (新規証明書要求)」を選択し、要求される詳細情報を入力します。
 - 証明書を保管し、それをファイルに抽出し、CA に送信します。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

- c. 証明書が発行されたら、それを Web サーバーのデータベースに保管します。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Personal Certificates (個人証明書)」を選択し、「Receive (受信)」をクリックします。

個人的な内部テスト用の一時的な手段として、CA から証明書を入手する代わりに、自己署名クライアント証明書を作成することもできます。ただし、これでは必要なレベルのセキュリティは確保されないため、実際のシステムで使用するべきではありません。自己署名証明書を作成するには、以下を実行します。

- a. キー・マネージャー・ユーザー・インターフェースから、「Create (作成)」、「New Self-Signed Certificate (新規自己署名証明書)」を選択し、要求される詳細情報を入力します。
 - b. 証明書を保管し、それをファイルに抽出します。
 - c. その証明書ファイルを Web サーバーのデータベースに保管します。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Personal Certificates (個人証明書)」を選択し、「Receive (受信)」をクリックします。
5. 証明書の構成が完了したら、GSKIT キー・マネージャーを終了します。

Linux for System z での Remote API Client のアンインストール

Linux for System z 上の Remote API Client は、以下のコマンドを使用してアンインストールすることができます。

```
/opt/ibm/sna/bin/sna stop  
rpm -e ibm-commsserver-ptf  
rpm -e ibm-commsserver-docs  
rpm -e ibm-commsserver-ecl  
rpm -e ibm-commsserver-cli  
rpm -e ibm-commsserver  
rpm -e gsk7bas  
/sbin/shutdown -r now
```

これらのコマンドにリストされているパッケージのすべてが、どのシステムにもインストールされるわけではありません。

Linux for System z 上の IBM Remote API Client をアンインストールしても、カスタマイズされた構成情報は、後のインストールで使用できるように残されます。

第 6 章 AIX システムでの IBM Remote API Client のインストール

この章では、AIX での IBM Remote API Client のインストールの方法について説明します。このインストールにより、AIX ワークステーションは、SNA スタックがフルにインストールされていなくても SNA アプリケーションを実行できるようになります。AIX 上の Remote API Client は、TCP/IP ネットワークを使用して 1 つ以上の Communications Server for Linux サーバー (または CS/AIX サーバー) に接続できます。

このソフトウェアをインストールする前に、IBM Remote API Client README ファイルを読むことをお勧めします。このファイルは、インストール CD の `/ibm-commserver-clients/aix` ディレクトリーにあります。

ハードウェア要件とソフトウェア要件

ハードウェア要件

IBM Remote API Client は、『オペレーティング・システムのバージョン』のリスト中のいずれかの AIX オペレーティング・システムでサポートされている pSeries システムを必要とします。

オペレーティング・システムのバージョン

現行バージョンの IBM Remote API Client は、以下のバージョンのオペレーティング・システムでテスト済みです。

- AIX v5.2-ML5
- AIX v5.3-ML1

Java

Java CPI-C API を使用する場合、Java ソフトウェアが必要になります。

<http://www.ibm.com/developerworks/java/jdk> から入手できる Java 1.4.2 SDK は、すべての要件を満たします。64 ビット Java も使用可能ですが、Java CPI-C API は 32 ビット JVM を必要とします。

Java SDK パッケージは、`installp` コマンドを使用してインストールします。

GSKIT

クライアントが HTTPS を使用して Communications Server for Linux サーバーに接続する場合、WebSphere サーバーを通してサーバーに HTTPS アクセスできるようにするため、GSKIT ソフトウェアが必要です。詳しくは、インストール CD の `/ibm-commserver-clients/aix` ディレクトリーにある **README** ファイルを参照してください。この章の後で説明するように、GSKIT ソフトウェアは、クライアントをインストールするメイン・プロセスの一部として自動的にインストールされます。

言語環境変数の変更

Remote API Client を使用するときには、LANG 変数が C に設定されていないことを確認してください。

使用されている LANG 変数の表示、または LANG 変数の変更に使用する手順は以下のとおりです。

1. SMIT メイン・メニューから「**System Environments (システム環境)**」を選択します。
2. 次の SMIT メニューから「**Manage Language Environment (言語環境の管理)**」を選択します。
3. 次の SMIT メニューから「**Change/Show Primary Language Environment (1 次言語環境の変更/表示)**」を選択します。
4. 次の SMIT メニューから「**Change/Show Cultural Convention, Language, or Keyboard (国別情報、言語、キーボードの変更/表示)**」を選択します。
5. 使用したい言語を選択します。たとえば、米国英語のメッセージを使用するには「en_US」を選択します。

AIX での Remote API Client のインストール

前提条件ソフトウェアのインストールが完了したら、IBM Remote API Client をインストールすることができます。

前のレベルの IBM Remote API Client が既にインストールされている場合、新しいレベルをインストールする前に、59 ページの『AIX 上の Remote API Client のアンインストール』に説明されているステップに従ってそれを除去してください。新規インストールで使用できるよう、構成情報はそのまま残されます。

AIX ワークステーションへのファイルのコピーによる Remote API Client のインストール

Remote API Client をインストールするには、次のステップを実行します。

1. **sna.client.6.3.0.1.bff** ファイルを、CD-ROM の **/ibm-commserver-clients/aix** ディレクトリーから、AIX ワークステーションにコピーするか、FTP で転送します。コピーまたは FTP でのファイル転送を行う際は、バイナリー・モードを使用してください。

クライアントが HTTPS を使用して Communications Server for Linux サーバーに接続する場合、さらに 2 つのファイル、**gskta.*.I** と **gksa.*.I** を、CD の同じディレクトリーからコピーまたは FTP 転送する必要があります。これらのファイルには、クライアントからの HTTPS アクセスに必要な GSKIT ソフトウェアが含まれています。

2. root として AIX ワークステーションにログインします。
3. **smit** または **installp** のいずれかを使用して、AIX Client をインストールします。その方法について詳しくは、インストール CD の **/ibm-commserver-clients/aix** ディレクトリーにある **README** ファイルを参照してください。

4. クライアントが HTTPS を使用して Communications Server for Linux サーバーに接続する場合は、**README** ファイルの指示に従って GSKIT ファイルをインストールしてください。
5. インストール・プロセスが完了したら、**sna.client.6.3.0.1.bff** ファイルおよび GSKIT ファイルを作業ディレクトリーから削除してもかまいません。
6. IBM Remote API Client を開始します。インストール後は、マシンがリブートされたら自動的に開始されるようになります。

```
cd /  
sna start
```

注: IBM Remote API Client が HTTPS を使用してサーバーに接続できるようにするには、GSKIT キー・マネージャー・プログラムを使用して、セキュリティー証明書構成をクライアントにセットアップする必要があります。詳しくは、58 ページの『GSKIT を使用した HTTPS セキュリティー証明書のセットアップ』を参照してください。

また、クライアント・ネットワーク・データ・ファイルを更新して、クライアントが接続可能な Communications Server for Linux サーバーと、HTTPS サポートを提供する WebSphere サーバーの名前を指定する必要があります。詳しくは、「*Communications Server for Linux 管理ガイド*」の Remote API Client の管理に関するセクションを参照してください。

CD からの Remote API Client のインストール

Remote API Client をインストールするには、次のステップを実行します。

1. root として AIX ワークステーションにログインします。
2. 以下のコマンドを使用して、CD を AIX ワークステーションにマウントします。

```
mount -o ro /dev/cd0 /mnt
```

3. **smit** または **installp** のいずれかを使用して、AIX Client をインストールします。その方法について詳しくは、インストール CD の **/ibm-commserver-clients/aix** ディレクトリーにある **README** ファイルを参照してください。
4. クライアントが HTTPS を使用して Communications Server for Linux サーバーに接続する場合は、**README** ファイルの指示に従って GSKIT ファイルをインストールしてください。
5. インストール・プロセスが完了したら、以下のコマンドを使用して CD をアンマウントします。

```
umount /mnt
```

6. IBM Remote API Client を開始します。インストール後は、マシンがリブートされたら自動的に開始されるようになります。開始するときには、まだ CD のディレクトリーにいないことがないようにしてください。

```
cd /  
sna start
```

注: IBM Remote API Client が HTTPS を使用してサーバーに接続できるようにするには、GSKIT キー・マネージャー・プログラムを使用して、セキュリティー証明書構成をクライアントにセットアップする必要があります。詳しくは、『GSKIT を使用した HTTPS セキュリティー証明書のセットアップ』を参照してください。

また、クライアント・ネットワーク・データ・ファイルを更新して、クライアントが接続可能な Communications Server for Linux サーバーと、HTTPS サポートを提供する WebSphere サーバーの名前を指定する必要があります。詳しくは、「*Communications Server for Linux 管理ガイド*」の Remote API Client の管理に関するセクションを参照してください。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

IBM Remote API Client が HTTPS を使用してサーバーに接続できるようにするには、GSKIT キー・マネージャー・プログラムを使用して、セキュリティー証明書構成をクライアントにセットアップする必要があります。次のステップを実行してください。

1. 以下のコマンドを使用して GSKIT キー・マネージャーを実行します。

`/usr/bin/snakeyman`

キー・マネージャー・ユーザー・インターフェース内から、鍵データベース・ファイル `/etc/sna/ibmcs.kdb` を開きます。このファイルは CMS 形式です。

2. 鍵データベースの初期パスワードは `ibmcs` です。セキュリティー証明書をセットアップする前に、構成を保護するため、このパスワードを変更しなければなりません。パスワードを変更するためのダイアログでは、「Stash the password to a file? (パスワードをファイルへ stash する?)」というチェック・ボックスにマークを付ける必要があります。そうすると、新規パスワードが保管されるので、クライアントは鍵データベースをオープンできます。
3. Web サーバーのセキュリティー証明書に署名するのに使用された認証局 (CA) 証明書のコピーを入手し、それを鍵データベースにインストールします。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Signer Certificates (署名者証明書)」を選択し、「Add (追加)」をクリックします。
4. WebSphere サーバーがクライアント・セキュリティー証明書を必要とするように構成されている場合、クライアントは、自身の証明書が Web サーバーのセキュリティー証明書データベース内にある CA によって発行された証明書を持っている必要があります。新規証明書を要求するには、以下を実行します。
 - a. キー・マネージャー・ユーザー・インターフェースから、「Create (作成)」、「New Certificate Request (新規証明書要求)」を選択し、要求される詳細情報を入力します。
 - b. 証明書を保管し、それをファイルに抽出し、CA に送信します。
 - c. 証明書が発行されたら、それを Web サーバーのデータベースに保管します。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Personal Certificates (個人証明書)」を選択し、「Receive (受信)」をクリックします。

GSKIT を使用した HTTPS セキュリティ証明書のセットアップ

個人的な内部テスト用の一時的手段として、CA から証明書を入手する代わりに、自己署名クライアント証明書を作成することもできます。ただし、これでは必要なレベルのセキュリティは確保されないため、実際のシステムで使用するべきではありません。自己署名証明書を作成するには、以下を実行します。

- a. キー・マネージャー・ユーザー・インターフェースから、「Create (作成)」、「New Self-Signed Certificate (新規自己署名証明書)」を選択し、要求される詳細情報を入力します。
 - b. 証明書を保管し、それをファイルに抽出します。
 - c. その証明書ファイルを Web サーバーのデータベースに保管します。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Personal Certificates (個人証明書)」を選択し、「Receive (受信)」をクリックします。
5. 証明書の構成が完了したら、GSKIT キー・マネージャーを終了します。

AIX 上の Remote API Client のアンインストール

Remote API Client は、以下のコマンドを使用してアンインストールできます。

1. クライアント・ソフトウェアが実行中であれば、以下のコマンドを使用して停止します。

sna stop

2. root 権限によりログインします。
3. 以下のコマンドのいずれかを使用して、Remote API Client パッケージおよび関連ソフトウェア・パッケージを除去します。

installp を使用してパッケージを除去する場合:

installp -u sna.client

smit を使用してパッケージを除去する場合:

smit remove

第 7 章 Windows 上の Remote API Client の計画とインストール

この章では、Windows での IBM Remote API Client のインストールの方法について説明します。このインストールにより、PC に SNA スタックをフルにインストールしなくても PC で SNA アプリケーションを実行できるようになります。Windows 上の Remote API Client は、TCP/IP ネットワークを使用して 1 つ以上の Communications Server for Linux サーバー (または CS/AIX サーバー) に接続できます。

Windows 上の IBM Remote API Client には、使用する固有のハードウェア、および Windows のバージョンに応じた 2 つの変種があります。この章の情報は、相違を明示して注記する場合を除き、両方の変種に適用されます。

- 32 ビット・クライアントは、Microsoft Windows 2000、2003、または XP が稼働する 32 ビット Intel ベースのコンピューターで実行されます。
- x64 クライアントは、Microsoft Windows Server 2003 x64 Edition または Microsoft Windows XP Professional x64 Edition が稼働する AMD64 または Intel EM64T コンピューターで実行されます。

Windows 上の IBM Remote API Client によって提供されるインターフェースは、IBM Communications Server for Windows および Microsoft Host Integration Server 製品によって提供されるインターフェースとおおむね互換性があります。

IBM Remote API Client on Windows Software Development Kit (SDK) は、オプションのパッケージであり、Remote API Client を使用して、APPC、CPI-C、LUA、および CSV の各 API を使用するアプリケーション・プログラムを開発することを可能にします。これらの API についての詳細は、適切なプログラマー用解説書を参照してください。Remote API Client が、新規アプリケーションを開発するためではなく、既存アプリケーションを実行するためにのみ使用される場合は、このパッケージをインストールする必要はありません。

ハードウェア要件とソフトウェア要件

Windows 上で **Setup** プログラムおよび Remote API Client を実行するには、コンピューターは以下の要件を満たしている必要があります。

- 以下のいずれかのオペレーティング・システムを実行していなければなりません。
 - 32 ビット Windows クライアントの場合:
 - Windows 2000
 - Windows XP
 - Windows 2003
 - x64 Windows クライアントの場合:
 - Microsoft Windows XP Professional x64 Edition

ハードウェア要件とソフトウェア要件

- Microsoft Windows Server 2003 x64 Edition
- 1 つ以上の Communications Server for Linux サーバーへの、以下のいずれかのメカニズムを使用するアクセスが可能でなければなりません。
 - TCP/IP ネットワークを介して行われる、サーバーへのアクセス
 - Communications Server for Linux サーバーへの HTTPS アクセスを提供する WebSphere サーバーへのアクセス (この場合、クライアント・セットアップ・プロセスの一部として GSKIT ソフトウェアもコンピューターにインストールされていなければなりません)。

Setup プログラムへのアクセス

Remote API Client と SDK ソフトウェア、GSKIT ソフトウェア、および **Setup** プログラムは、インストール CD に Windows 形式で収録されているので、それらを CD から Windows コンピューターにインストールできます。Remote API Client ソフトウェアは、各 Windows クライアント PC にインストールする必要があります。SDK が必要なのは、クライアントを使用して Windows Remote API を使用する新規アプリケーションを開発する場合のみであり、既存アプリケーションを実行するだけの場合は必要ありません。GSKIT ソフトウェアが必要なのは、クライアントが HTTPS を使用して Communications Server for Linux サーバーにアクセスする場合のみです。

Windowsインストール・イメージ上の Remote API Client は自己解凍型 ZIP 実行可能ファイルで、インストールCD で送達されます。

- 32 ビット・クライアントの場合は **i_w32cli.exe** です。CD のディレクトリー **/ibm-commsserver-clients/windows** に入っています。
- x64 クライアントの場合は **i_w64cli.exe** です。CD のディレクトリー **/ibm-commsserver-clients/win-x64** に入っています。

ネットワークを介してこのファイルを他の Windows PC にコピーできるので、Communications Server for Linux 配布 CD に直接アクセスすることなく、それらのクライアントをインストールすることができます。この実行可能ファイルを実行すると、インストール・イメージが unzip され、自動的に **Setup** プログラムが実行されます。単にインストール・イメージを unzip して一時ディレクトリーに入れたい場合 (たとえば **Setup** プログラムをコマンド行から入力するために)、上記の自己解凍型実行可能ファイルを適当な unzip プログラムにロードすることによって、それを行うことができます。

ある特定のコンピューターで **Setup** プログラムが初めて実行されるときには、このプログラムは選択されたソースから実行します。このプログラムは、完全インストール・プロセスを処理し、基本的な構成をセットアップし、プログラム自身のアイコンのインストールと作成も行います。インストールの完了後、当ソフトウェアの再インストールが必要になった場合も、**Setup** プログラムを使用できます (「File Manager (ファイル・マネージャー)」からこのプログラムを選択するか、アイコンを選択します)。

Remote API Client インストール・イメージを一時ディレクトリーに抽出した後は、以下の 2 つの方法のいずれかでこのソフトウェアをインストールできます。

- 『Setup プログラムを使用した Windows での Remote API Client のインストール』に説明されているように、Windows で **Setup** プログラムを実行する。SDK をインストールしたい場合は、この方法を使用する必要があります。
- 67 ページの『コマンド行からの Remote API Client ソフトウェアのインストール』に説明されているように、**setup** コマンドをコマンド行から入力する。この方法では、SDK はインストールできません。

注: IBM Remote API Client が HTTPS を使用してサーバーに接続できるようにするには、GSKIT ソフトウェアをインストールし、GSKIT キー・マネージャー・プログラムを使用して、セキュリティ証明書構成をクライアントにセットアップする必要があります。詳しくは、70 ページの『GSKIT ソフトウェアのインストールおよびセキュリティ証明書のセットアップ』と 71 ページの『GSKIT を使用した HTTPS セキュリティ証明書のセットアップ』を参照してください。

また、クライアント・ネットワーク・データ・ファイルを更新して、クライアントが接続可能な Communications Server for Linux サーバーと、HTTPS サポートを提供する WebSphere サーバーの名前を指定する必要があります。詳しくは、「*Communications Server for Linux 管理ガイド*」の Remote API Client の管理に関するセクションを参照してください。

Setup プログラムを使用した Windows での Remote API Client のインストール

setup プログラムは、自己解凍型 ZIP 実行可能ファイル **i_w32cli.exe** (32 ビット・クライアント) または **i_w64cli.exe** (x64 クライアント) の実行の一部として自動的に実行させるか、または、コマンド行から手動で実行します。このプログラムは、最初に「Choose Setup Language (Setup 言語の選択)」画面を表示します。

1. Remote API Client のインストールと構成に使用したい言語を選択し、「**OK**」をクリックします。

そうすると、**Setup** プログラムを紹介する初期画面が表示されます。

2. インストールを続けるには、「**Next (次へ)**」を選択します。

ソフトウェアのご使用条件が表示されるので、それを読んで理解します。

3. 使用条件を受け入れる場合は、「**Accept (同意する)**」をクリックして続けます。

ファイルをインストールする宛先ディレクトリーを指定するようプロンプトが出されます。

4. 宛先ディレクトリーを入力します。

希望するインストールのタイプを選択するよう求められます。

Standard (標準)

SDK をインストールする必要がない場合は、このオプションを選択します。SDK が必要なのは、クライアントを使用して Windows Remote

Setup プログラムを使用した Windows での Remote API Client のインストール

API を使用する新規アプリケーションを開発する場合のみであり、既存アプリケーションを実行するだけの場合は必要ありません。

Developer (開発者)

SDK をインストールする必要がある場合、つまりクライアントを使用して、Windows Remote API を使用する新しいアプリケーションを開発する場合は、このオプションを選択します。

注: SDK をインストールする場合は、「**Developer (開発者)**」を選択する必要があります。

5. インストールのタイプを選択します。

次に、プログラムは、Windows 用 Remote API Client のアイコンを表示するプログラム・フォルダーの名前の入力を要求します。

6. フォルダー名を入力します。
7. この **Setup** プログラムで使用されるファイルと同じ名前だが、Remote API Client ファイルではない (たとえば、他の SNA ソフトウェアからのファイル) **.DLL** ファイルが既にシステム・ディレクトリー内にある場合、以下のいずれかを実行するように指示するプロンプトが出されます。

- Remote API Client **.DLL** ファイルを既存の **.DLL** ファイルに上書きコピーする。
- 既存の **.DLL** ファイルを、インストール・ディレクトリーの下に **OTHERSNA** という名前で作成したサブディレクトリーにコピーし、その後で Remote API Client **.DLL** ファイルをインストールする。このオプションを選択すると、後でファイルをアンインストールする場合、Remote API Client インストール前の元のセットアップを復元できます (73 ページの『Remote API Client ソフトウェアのアンインストール』を参照してください)。
- クライアント・ソフトウェアのインストールをキャンセルする。

Remote API Client **.DLL** ファイルが既に存在する場合、**Setup** プログラムはそのことを知らせるメッセージを表示します。既存のファイルが **Setup** プログラムの **.DLL** ファイルよりも低いバージョン番号である場合に限り、新しい **.DLL** ファイルが既存の **.DLL** ファイルを上書きします。

8. この時点で、**Setup** プログラムは、指定されたソースからファイルをコピーし、それらを適当な場所にインストールします。このプロセス中は、インストールのどの部分が完了したかを表す通知バーが表示されます。**.DLL** ファイルは、System ディレクトリーまたはそれと同等のディレクトリーにコピーされ、その他のファイルは、ステップ 2 で指定した宛先ディレクトリーにコピーされます。各ファイルの転送操作ごとに 1 つのレコードが、指定したディレクトリーに作成される **setup.log** ファイルに書き込まれます。ファイルのどれかが「読み取り専用」に上書きされることになる場合、または、他の何らかの理由でコピーできないファイルがある場合、新しいファイル群は除去され、**setup.log** ファイルを調べるように通知するメッセージが表示されます。
9. **Setup** プログラムの実行元のソースにすべての必要ファイルが含まれていない場合、ディレクトリー名の入力を求めるプロンプトがこのプログラムから出されます。必要ファイルが置かれているディレクトリーの名前を入力してください。

Setup プログラムを使用した Windows での Remote API Client のインストール

指定した情報が Remote API Client ファイルを見つけるのに十分でない場合、同じ画面がもう一度表示されます。

10. 必要ファイルのコピーが完了したら、**Setup** プログラムは「Configuration (構成)」ウィンドウを表示します。

デフォルトの構成値は、ドメイン構成ファイルから取られます。詳しくは、「*Communications Server for Linux 管理ガイド*」を参照してください。それらのデフォルト値を使用したくない場合は、下記のように構成することができます。

Domain (ドメイン)

Communications Server for Linux クライアント/サーバー・ドメイン・ネームを指定します。

以下の設定はオプションです。

Server Name (サーバー名)

このクライアントが接続できる 9 個までのサーバーが画面に表示されます。そのリスト中でサーバーが現れる順序に従って、クライアントはサーバーを選択します。リストの先頭にあるサーバーにクライアントが接続できない場合、その次のサーバーが試されます。

- 新しいサーバーをリストに追加するには、「**Add (追加)**」ボタンを使用します。
- リストからサーバーを除去するには、サーバーを選択してから「**Remove (除去)**」ボタンを使用します。
- サーバーをリスト内で上下に移動するには、サーバーを選択し、リストの横にある移動ボタンを使用します。

クライアントが、サーバーと同じプライベート・ネットワークにあり、サーバーへのアクセスに TCP/IP を使用している場合、各サーバーは単純にそのサーバー名で識別されます。

クライアントがサーバーへのアクセスに HTTPS を使用している場合、各サーバーを識別するには、HTTPS サポートを提供している WebSphere サーバーの名前、および Communications Server for Linux サーバーの名前を、以下の形式で指定する必要があります。

webservername : *servername1*

これは、WebSphere が HTTPS 接続にデフォルトのポート 443 を使用するようセットアップされていると想定しています。ネットワーク管理者によって、別のポート番号を使用するように WebSphere が構成されている場合、以下の形式でポート番号を組み込む必要があります。

webservername : *portnumber* : *servername1*

HTTPS 接続サポートに関する WebSphere の構成について詳しくは、36 ページの『WebSphere Application Server の構成』を参照してください。

UDP broadcasts (UDP ブロードキャスト)

このクライアントがサーバーへの接続に UDP ブロードキャストを使用するかどうかを指定します。このオプションが選択されている場合、ク

Setup プログラムを使用した Windows での Remote API Client のインストール

クライアントは、特定のサーバーに直接接続しようとする代わりに、UDP ブロードキャストをネットワーク上へ送信してサーバー接続を見つけます。

UDP ブロードキャストを使用するのが、デフォルトの設定です。この設定を変更するには、ボックスをクリックします。

Advanced (拡張)

Setup プログラムで提供されるデフォルトの代わりに値を追加指定するには、ウィンドウの下部にある「**Advanced (拡張)**」ボタンをクリックします。**Setup** プログラムにより、Windows クライアント構成の拡張設定が含まれている「**Advanced Options (拡張オプション)**」ウィンドウが表示されます。ほとんどのユーザーがこれらのパラメーターに対してデフォルトの設定値を使用でき、このダイアログで設定値を変更する必要はありません。

これらのパラメーターについての詳細は、『Remote API Client 構成の拡張オプション』を参照してください。

構成パラメーターまたは設定値についての詳細は、「**Help (ヘルプ)**」をクリックしてください。

11. 「**Configuration (構成)**」ウィンドウを完了したら、「**OK**」をクリックします。この画面を正しく完了していない場合は、**Setup** プログラムからメッセージが表示されます。
12. インストールが正常に完了したら、「**Finish (完了)**」ウィンドウが表示されます。インストール・プログラムの終了後に実行されるアクションとして、以下のいずれかまたは両方を選択できます。

View README file (README ファイルの表示)

README ファイルを表示します。

Start client (クライアントの開始)

この Communications Server for Linux クライアントの実行を開始します。

「**Finish (完了)**」を選択して、インストール・プログラムを終了します。

Remote API Client 構成の拡張オプション

「**Advanced Options (拡張オプション)**」ウィンドウで、Remote API Client に対するいくつかの拡張パラメーターを構成できます。大部分のユーザーはこれらのパラメーターを変更する必要はありませんが、必要な場合はデフォルト設定を調整できます。

LAN access time-out (LAN アクセス・タイムアウト)

クライアントからサーバーへの接続が、アイドルのままどれだけの時間が経過するとクローズされるのかを秒単位で指定します。このチェック・ボックスを空のままにすると、LAN アクセスにはタイムアウトが指定されません(従って、無限のタイムアウトが使用されます)。このボックスをチェックすると、その横のフィールドにタイムアウト値を秒単位で入力できます。最小

値は 60 (60 秒) です。このボックスをブランクのままにするか、60 よりも少ない値を指定した場合、Remote API Client は最小値の 60 を使用します。

Max. broadcast attempts (最大ブロードキャスト試行回数)

クライアントがブロードキャストによってサーバーへの接続を試行する最大回数を指定します。「Advanced Options (拡張オプション)」ウィンドウが開いたときには、デフォルト値の 5 が表示されます。このボックス内の値は、メインの「Configuration (構成)」ウィンドウの「UDP broadcasts (UDP ブロードキャスト)」チェック・ボックスがチェックされている場合のみ使用されます。

Reconnect time-out (再接続タイムアウト)

サーバーがダウンした後、クライアントはどれだけ待機してからそのサーバーへの再接続を試みるのかを、秒単位で指定します。「Advanced Options (拡張オプション)」ウィンドウが開いたときには、デフォルト値の 200 が表示されます。

これらのパラメーターについての詳細は、「Help (ヘルプ)」を押してください。

「Advanced Options (拡張オプション)」ウィンドウを完了したら、「OK」をクリックします。ユーザーがこの画面を正しく完了した場合、Setup プログラムは、「Configuration (構成)」ウィンドウに戻ります。新しい Remote API Client をインストールする場合、ステップ 11 (66 ページ) に戻ってください。それ以外の場合、「Configuration (構成)」ダイアログで「OK」ボタンをクリックします。

コマンド行からの Remote API Client ソフトウェアのインストール

注: SDK をインストールしたい場合は、63 ページの『Setup プログラムを使用した Windows での Remote API Client のインストール』に説明されているように、Setup プログラムを使用する必要があります。SDK をコマンド行からインストールすることはできません。

Remote API Client インストール・イメージを一時ディレクトリーに抽出した後は、Windows で Setup プログラムを使用する代わりに、コマンド行から Remote API Client ソフトウェアをインストールできます。コマンド行で、1 つ以上のオプションを付けて **setup** コマンドを入力します。オプションは、大文字でも小文字でも入力でき、オプションの前に / (スラッシュ) または - (ハイフン) を付けることができます。スペースを含む文字列からなるパラメーター (たとえば *folder*) は、二重引用符で囲む必要があります。

ユーザーが **setup** コマンドを入力すると、Setup プログラムは、ユーザーがコマンド行に組み入れなかった情報があれば入力を求めるプロンプトを出し、セットアップのさまざまな段階で確認メッセージを表示します。Setup プログラムからのプロンプトが表示されないようにしたい場合は、**-accept -s** オプションを使用すると、このプログラムはサイレント・モードで実行し、ソフトウェア・ライセンス使用条件を受け入れます。

setup コマンドのオプションは、次のとおりです。

コマンド行からの Remote API Client ソフトウェアのインストール

- ?** コマンド行オプションのリストを表示します。これは、**-h** オプションと同じです。
- h** コマンド行オプションのリストを表示します。これは、**-?** オプションと同じです。
- accept -s**

インストールをサイレント・モードで実行し、ソフトウェア・ライセンス使用条件を受け入れます。この使用条件は、Windows インストール・イメージの **license** サブディレクトリーにあります。

-s オプションは、コマンド行の最後に置かなければならず、ドメイン名 (**-i** オプションを使用) およびその他の指定したいパラメーターをすべて指定したことを確認する必要があります。インストールがサイレント・モードで実行される場合、パラメーターを求めるプロンプトは出されず、確認メッセージも表示されません。**-s** の後にあるコマンド行引数は無視されます。
- f2** サイレント・モードでのインストール (**-s** オプションを使用) 中に作成されるインストール・ログ・ファイルの絶対パス名を指定します。

このオプションを指定しない場合、インストール・プログラムを実行したディレクトリー内に、**setup.log** としてファイルが作成されます。CD ドライブからサイレント・モードでインストールしている場合、このオプションを指定して、ファイルがコンピューター上に作成されるようにしなければなりません (CD ドライブには作成できないため)。

- kfolder**

プログラム・フォルダーを指定します。
- pdirectory**

インストール・ディレクトリーを指定します。
- idomain**

このクライアントのドメイン名を指定します。このパラメーターは必須です。デフォルトはありません。
- wdirectory**

ソースがディスクまたは CD 上にある場合に、Communications Server for Linux クライアント・ソフトウェア・ファイルが含まれているソース・ディレクトリーを指定します。それ以外の場合、**-v** オプションを使用してください。
- vserver**

クライアント・ソフトウェアのファイルをダウンロードする元のサーバーを指定します。サーバー名または TCP/IP アドレスのいずれかで指定します。ソース・ファイルをディスクまたは CD からコピーする場合、**-v** オプションの代わりに **-w** オプションを使用してください。
- lserver**

このクライアントがアクセスできるサーバーのリストに入れるサーバーを指定します。

クライアントが、サーバーと同じプライベート・ネットワークにあり、サーバーへのアクセスに TCP/IP を使用している場合、各サーバーは単純にそのサーバー名で識別されます。

コマンド行からの Remote API Client ソフトウェアのインストール

クライアントがサーバーへのアクセスに HTTPS を使用している場合、各サーバーを識別するには、HTTPS サポートを提供している WebSphere サーバーの名前、および Communications Server for Linux サーバーの名前を、以下の形式で指定する必要があります。

webservername : servername1

これは、WebSphere が HTTPS 接続にデフォルトのポート 443 を使用するようセットアップされていると想定しています。ネットワーク管理者によって、別のポート番号を使用するように WebSphere が構成されている場合、以下の形式でポート番号を組み込む必要があります。

webservername : portnumber : servername1

HTTPS 接続サポートに関する WebSphere の構成について詳しくは、36 ページの『WebSphere Application Server の構成』を参照してください。

- o 既存の **.DLL** ファイルを上書きします。Remote API Client **.DLL** ファイルが既に存在する場合、それらのファイルが **Setup** プログラムの **.DLL** ファイルよりも大きいバージョン番号である場合でも、**Setup** プログラムは、それらのファイルを上書きします。
- y 既存の **.DLL** ファイルを保管します。必要なディレクトリー内に既に Remote API Client **.DLL** ファイルが存在する場合、**Setup** プログラムは、それらの既存の **.DLL** ファイルをインストール・ディレクトリーの下の子ディレクトリーにコピーし、その後で Remote API Client **.DLL** ファイルをインストールします。サブディレクトリーにコピーされたファイルによって、Remote API Client ソフトウェアをアンインストールする場合に、アンインストール・プロセスが完了することが保証されます。
- n 既存の Remote API Client **.DLL** ファイルが検出された場合、インストールをキャンセルします。

-atimeout

LAN アクセスのタイムアウト時間を秒単位で指定します。これは、クライアントからサーバーへの接続が、アイドルのままどれだけの時間が経過するとクローズされるのかを表します。値 0 は、タイムアウトにならないことを示します。

-bmax-broadcast

UDP ブロードキャストの最大試行回数を指定します。UDP ブロードキャストとは、ある特定のサーバーではなく、ドメイン内の任意のサーバーへの、クライアントからの接続の試行です。値 0 は、ブロードキャスト試行が行われないことを示します。

-jreconnect-timeout

サーバーがダウンした後、クライアントはどれだけ待機してからそのサーバーへの再接続を試みるのかを、秒単位で指定します。

GSKIT ソフトウェアのインストールおよびセキュリティー証明書のセットアップ

注: Windows 用 x64 Remote API Client の初期リリースは、サーバーへの HTTPS 接続をサポートしません。このフィーチャーは、インストールおよび構成に関する詳細な指示と共に PTF で提供されます。以下の説明は、32 ビット・クライアントにのみ適用されます。

IBM Remote API Client が HTTPS を使用してサーバーに接続できるようにするには、GSKIT ソフトウェアをインストールしてセキュリティー証明書を管理する必要があります。以下のステップを実行します。

1. Windows コンピューター上に、インストール・ファイル格納用の一時ディレクトリーを作成します。
2. 32 ビット・クライアントの場合:
 - ファイル **gsk7bas.exe** をインストール CD の **/ibm-commsserver-clients/windows** ディレクトリーから一時ディレクトリーにコピーします。
 - コマンド・ウィンドウで、一時ディレクトリーに移動します。
 - 以下のコマンドを実行してインストール・ファイルを抽出します。

```
gsk7bas tempdir /D
```

tempdir は、一時ディレクトリーの絶対パスです。

- 以下のコマンドを実行して、インストール・プログラムを開始します。

```
setup sxclient
```

3. 画面の指示に従って、GSKIT ソフトウェアをインストールしてください。
4. インストール完了後、最初のステップで作成した一時ディレクトリーとその内容を削除します。

これで、以下のセクションに説明されているように、セキュリティー証明書をセットアップすることができます。

32 ビット・クライアントへの無人インストールを実行する必要がある場合 (たとえば、多数のクライアント・コンピューターに GSKIT ソフトウェアをインストールするため、プロセスを自動化するバッチ・ファイルを作成したい場合)、以下のコマンドを使用すると、インストール・プログラムをサイレント・モードで実行できます。

```
setup gskit -s -fl "tempdir\SETUP.ISS"
```

tempdir は、最初のステップで作成した一時ディレクトリーの絶対パスです。

このコマンドを使用すると、インストール・プログラムのユーザー・インターフェースが表示されることなく、ユーザー入力を必要とせずに、ソフトウェアがインストールされます。インストールのデフォルトは、インストール・プロセスの最初に一時ディレクトリーに抽出されたファイルの 1 つである **SETUP.ISS** ファイルから取られます。GSKIT ソフトウェアのインストール先ディレクトリーを変更する必要

がある場合は、一時ディレクトリーへのファイル抽出が終わった後に、テキスト・エディターを使用して **SETUP.ISS** を編集します。更新したこのファイルも含めて、このディレクトリー内のファイルは、他のクライアント・コンピューターへのインストールにも使用できるので、そのたびごとにファイルを抽出したり、インストール先ディレクトリーを修正する必要はありません。

GSKIT を使用した HTTPS セキュリティ証明書のセットアップ

IBM Remote API Client が HTTPS を使用してサーバーに接続できるようにするには、GSKIT キー・マネージャー・プログラムを使用して、セキュリティ証明書構成をクライアントにセットアップする必要があります。次のステップを実行してください。

1. GSKIT キー・マネージャー・プログラム *installdir\snakeyman.exe* を実行します。*installdir* は、クライアント・ソフトウェアをインストールしたディレクトリーで、クライアントのインストール中に別の場所を指定していない限り、**C:\IBMCS\w32cli** (32 ビット・クライアント) です。

キー・マネージャー・ユーザー・インターフェース内から、鍵データベース・ファイル *installdir\ibmcs.kdb* を開きます。このファイルは CMS 形式です。

2. 鍵データベースの初期パスワードは *ibmcs* です。セキュリティ証明書をセットアップする前に、構成を保護するため、このパスワードを変更しなければなりません。パスワードを変更するためのダイアログでは、「Stash the password to a file? (パスワードをファイルへ stash する?)」というチェック・ボックスにマークを付ける必要があります。そうすると、新規パスワードが保管されるので、クライアントは鍵データベースをオープンできます。
3. Web サーバーのセキュリティ証明書に署名するのに使用された認証局 (CA) 証明書のコピーを入手し、それを鍵データベースにインストールします。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Signer Certificates (署名者証明書)」を選択し、「Add (追加)」をクリックします。
4. WebSphere サーバーがクライアント・セキュリティ証明書を必要とするように構成されている場合、クライアントは、自身の証明書が Web サーバーのセキュリティ証明書データベース内にある CA によって発行された証明書を持っている必要があります。新規証明書を要求するには、以下を実行します。
 - a. キー・マネージャー・ユーザー・インターフェースから、「Create (作成)」、「New Certificate Request (新規証明書要求)」を選択し、要求される詳細情報を入力します。
 - b. 証明書を保管し、それをファイルに抽出し、CA に送信します。
 - c. 証明書が発行されたら、それを Web サーバーのデータベースに保管します。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Personal Certificates (個人証明書)」を選択し、「Receive (受信)」をクリックします。

個人的な内部テスト用の一時的な手段として、CA から証明書を入手する代わりに、自己署名クライアント証明書を作成することもできます。ただし、これでは必要なレベルのセキュリティは確保されないため、実際のシステムで使用すべきではありません。自己署名証明書を作成するには、以下を実行します。

GSKIT を使用した HTTPS セキュリティー証明書のセットアップ

- a. キー・マネージャー・ユーザー・インターフェースから、「Create (作成)」、「New Self-Signed Certificate (新規自己署名証明書)」を選択し、要求される詳細情報を入力します。
 - b. 証明書を保管し、それをファイルに抽出します。
 - c. その証明書ファイルを Web サーバーのデータベースに保管します。これを行うには、キー・マネージャー・ユーザー・インターフェースから「Personal Certificates (個人証明書)」を選択し、「Receive (受信)」をクリックします。
5. 証明書の構成が完了したら、GSKIT キー・マネージャーを終了します。

インストール後の Remote API Client ソフトウェアのカスタマイズ

カスタマイズした設定は、インストール後にいつでも変更できます。そのためには、Communications Server for Linux プログラム・グループに含まれている **Configuration Utility** プログラムを実行します。このプログラムは、初期インストール・プロセスで表示されたのと同じ「Configuration (構成)」ウィンドウを表示します。63 ページの『Setup プログラムを使用した Windows での Remote API Client のインストール』に説明されている手順に従って、任意のフィールド内の情報を変更できます。

初期インストール中に SDK ファイルをインストールしなかった場合で、この時点でそれらを追加したい場合、もう一度 Setup プログラムを実行すればそれを行うことができます。インストール・タイプに「**Custom (カスタム)**」を選択し、SDK パッケージを選択します。

Remote API Client ソフトウェアの再インストール

Remote API Client ソフトウェアは、たとえば、このソフトウェアをアップグレードしたいときなど、任意の時点で再インストールできます。

再インストールを行うには、63 ページの『Setup プログラムを使用した Windows での Remote API Client のインストール』または 67 ページの『コマンド行からの Remote API Client ソフトウェアのインストール』の説明に従って、前と同様に Setup プログラムを実行します。Setup プログラムは、初期インストール中にクライアント・ソフトウェアのファイルのコピー元になった場所を表示します。この同じ場所からファイルの新規コピーを取得する場合は、「**OK**」をクリックします。「**OK**」をクリックすると、Setup プログラムは、ファイルをコピーしてから「Options (オプション)」画面に戻ります。

注: サイレント・モードで Remote API Client ソフトウェアを再インストールする場合 (67 ページの『コマンド行からの Remote API Client ソフトウェアのインストール』に説明されているように)、コンピューターを再始動しないとインストールを完了できないことがあります。なぜなら、一部のプログラム・ファイルがインストール・プロセス中に使用中である場合があり (たとえば、Remote API Client が稼動中である場合など)、新しいファイルで置換できないためです。このような場合には、新しいファイルは一時ディレクトリーにコピーされ、コンピューターが次に再始動されたときに自動的に移動されます。

コンピューターを再始動する必要があるかどうかを確認するには、インストール・プロセス完了時に、Notepad などのテキスト・エディターを使用して、イ

インストール・ログ・ファイルの内容を表示します。インストール・ログ・ファイルは **setup.log** という名前で、**-f2** コマンド行オプションで別のパスとファイル名を指定していない限り、**Setup** を実行したディレクトリーに作成されます。

このファイルの最後の部分で、**Response Result** という見出しの下に、テキスト **Result Code** に続けて 0 (ゼロ) または -12 の 2 つの値のいずれかがあるはずですが、値が 0 の場合はコンピューターを再始動する必要はありませんが、値が -12 の場合は、Windows クライアントを使用する前にコンピューターを再始動してください。

Remote API Client ソフトウェアのアンインストール

Remote API Client ソフトウェアは、「アプリケーションの追加と除去」オプションを Windows コントロール・パネルから使用すればいつでもアンインストールできます。アンインストール・プロセスが確定されると、Windows は以下を行いません。

- すべてのインストール済みファイルを削除します。
- 初期インストール中にサブディレクトリーに保管された **.DLL** ファイルがある場合、それらのファイルを元の場所にリストアします。
- 保管された **.DLL** ファイルを収容していたサブディレクトリーが空であれば削除します。
- プログラム・フォルダーおよび作成されたディレクトリーが空であれば削除します。
- アンインストールが正常終了した場合、すべてのファイル転送および削除が含まれている **setup.log** ファイルを削除します。
- アンインストールが正常終了したことを示すメッセージ、または、インストールの一部が失敗したので **setup.log** ファイルのチェックが必要であることを示すメッセージを表示します。

「**Exit (終了)**」ボタンをクリックすると、Windows に戻ります。

32 ビット Windows Remote API Client と共に使用するために GSKIT ソフトウェアをインストールした場合で、他にはそれを使用しているアプリケーションがない場合、『GSKIT ソフトウェアのアンインストール』の説明に従って、このソフトウェアをアンインストールしてもかまいません。

GSKIT ソフトウェアのアンインストール

32 ビット Windows Remote API Client と共に使用するために GSKIT ソフトウェアをインストールした場合で、他にはそれを使用しているアプリケーションがない場合、Windows Remote API Client をアンインストールした後、このソフトウェアをアンインストールしてもかまいません。次のステップを実行してください。

1. コマンド・ウィンドウで、Windows がインストールされているディレクトリーに移動します。通常、これは **C:\¥Windows** です。
2. 以下のコマンドを実行します。

```
gskbui sxclient
```

GSKIT ソフトウェアのアンインストール

GSKIT を使用している他のアプリケーションがない場合、Windows はそれらのファイルを削除します。まだ GSKIT を使用中のアプリケーションがある場合、それらのアプリケーションがアンインストールされるまでファイルは残されます。

ヘルプ

F1 キーを押すと、いつでもヘルプを利用できます。「Configuration (構成)」ウィンドウおよび「Advanced Options (拡張オプション)」ウィンドウにも、それぞれ「**Help (ヘルプ)**」ボタンがあります。

第 8 章 Communications Server for Linux の構成と使用

Communications Server for Linux 構成を定義および変更する場合、Motif 管理プログラム (**xsnaadmin**) を使用する方法が最も簡単です。このプログラムのグラフィカル・ユーザー・インターフェースから、ローカル・ノード上の SNA リソースを表示および管理できます。コマンド行管理など、それ以外の管理ツールを使用することもできますが、Motif プログラムの使用をお勧めします。

Motif 管理プログラムには、ヘルプ画面があります。この画面では、SNA および Communications Server for Linux に関する概要、Communications Server for Linux ダイアログの参照情報、特定の操作の説明が表示されます。この管理プログラムは、各タスク (ノードの構成など) または通信タイプ (TN3270 または APPC など) について、必要なリソースの構成をユーザーがセットアップする作業を支援します。

Motif 管理プログラムでは、標準 Communications Server for Linux 構成に必要なすべてのパラメーターを設定できます。拡張パラメーターには、Motif 管理プログラムによりデフォルト値が設定されます。指定する必要があるのは主要な構成情報のみなので、SNA 通信を簡単かつ迅速に設定できます。

また、Motif 管理プログラムを使用して、稼働中の Communications Server for Linux システムを管理することもできます。この管理プログラムでは、Communications Server for Linux が活動状態であるときでも、変更内容を構成に適用できます。また、ノード・リソースの状況情報へも簡単にアクセスできます。

Motif 管理プログラムでは、Communications Server for Linux リソースに関する状況情報が自動的に表示されます。状況情報の大部分は、「Node (ノード)」ウィンドウに表示されます (78 ページの『Motif 管理プログラムによる Communications Server for Linux の管理』を参照してください)。さらに「Node (ノード)」ウィンドウの、「**Start (開始)**」ボタンと「**Stop (停止)**」ボタンを使用して、ノードやリンク・ステーションなどの特定のリソースを管理することもできます。その他のリソースの始動と停止は常に自動的に実行されるため、手動で制御する必要はありません。

注:

1. Communications Server for Linux のリソースを定義または変更するには、ログイン・グループ `sna` のメンバーでなければなりません。
2. Motif 管理プログラムを使用するには、X 端末が必要です。
3. ウィンドウ内に表示されるボタンやアイコンなど、Motif 管理プログラムのユーザー・インターフェースに関する詳細については、そのプログラムのヘルプ画面または、「*Communications Server for Linux 管理ガイド*」を参照してください。
4. 特定のダイアログで選択した項目によっては、Motif 管理プログラムのウィンドウやダイアログが、本書に記載されているものと異なる場合があります。

その他の Communications Server for Linux 管理ツール (コマンド行管理および NOF アプリケーション・プログラムを含む) については、「*Communications Server for Linux 管理ガイド*」、「*Communications Server for Linux Administration Command Reference*」、または「*Communications Server for Linux NOF Programmer's Guide*」を参照してください。

Communications Server for Linux 構成の計画

構成を変更する場合は、事前に全体的な計画を立てておくことが大切です。変更内容によっては、ローカル・ノードのユーザーのみでなく、ネットワーク全体のユーザーの混乱を招く恐れがあります。

ネットワーク・トポロジーに対するすべての変更内容を表す図を作成すると便利です。他のノードへの接続を追加または除去しようとしている場合は、自分のノードと他のノードの関係図を作成してください。Motif 管理プログラムを使用して、既存のすべての接続に関する構成情報を収集し、その情報をノード図に書き加えることができます。

図に新しいリソースを追加するときには、新しいリソースが既存のリソースと重複していないかどうか、名前が競合していないかどうかを簡単に確認できます。同様に、図を使用すれば、除去する必要があるリソースを判別し、重要なリソースの削除を防止できます。

複数のノードを持つ Client/Server Communications Server for Linux システムを構成する場合は、すべての Communications Server for Linux ノードとその接続リソースをご使用のダイアグラムに組み込んでください。次に、この章での説明に従って、単体ノードを構成する場合と同様に、各ノードを順に構成します。

必要な変更が決まったら、その変更に必要な構成情報を収集します。特定の Communications Server for Linux 機能の構成情報を収集する際には、そのガイドとして、Motif 管理プログラムのオンライン・ヘルプにある作業シート、あるいは「*Communications Server for Linux 管理ガイド*」にある計画ワークシートを使用すると便利です。

この章では、Communications Server for Linux で使用可能な機能のうち、最も頻繁に使用される機能を構成する方法について説明します。各構成手順では、リソースを構成する前に収集する必要がある情報についても記述しています。

注: 本書では、Communications Server for Linux ダイアログに入力する必要がある構成情報の詳細については説明していません。特定のダイアログのフィールドの詳細については、Motif 管理プログラムの該当ダイアログのオンライン・ヘルプを参照してください。

計画ワークシート

Communications Server for Linux のリソースを構成する前に、新しいリソースの構成データをすべて収集します。「*Communications Server for Linux 管理ガイド*」にある計画ワークシートを使用して、サポートする必要がある特定の機能またはアプリケーションの情報をすべて記録します。

ほとんどの場合、ネットワーク管理者、ホスト管理者、アプリケーション・プログラマー、エンド・ユーザーなど、いくつかの情報源から構成情報を収集する必要があります。

他のノードへ接続しようとしている場合、そのノードの管理者と連絡をとることが大切です。ノード上のすべてのリソースについて、名前、アドレス、特性をそのノードの管理者から入手できます。また、ローカル・ノードで入力される構成パラメーターと、リモート・ノードで入力される構成パラメーターが一致していることを、確認する必要がある場合があります。

作業シート

Motif 管理プログラムのオンライン・ヘルプ画面には、特定の構成タスクをガイドする作業シートがあります。作業シートには、構成情報を入力するために使用される、各ダイアログのヘルプ画面へのポインターが示されます。これらのポインターを使用して、ヘルプ情報を表示し、収集が必要なデータを特定することができます。

作業シートでは、構成情報を入力するために使用する、ウィンドウやダイアログごとの詳細なヘルプも参照できます。これらのヘルプ画面では、入力または選択する必要のある各フィールドの説明が表示されます。

Motif 管理プログラムの使用

Motif 管理プログラムを使用する前に、システムが実行可能プログラムを検出できるように、各自の **.login** ファイルまたは **.profile** ファイルに、パス情報を追加します (『Communications Server for Linux プログラムのパスの指定』を参照してください)。また、Communications Server for Linux ソフトウェアを使用可能にしておかないと、管理プログラムは使用できません (78 ページの『Communications Server for Linux の使用可能化』を参照してください)。

Motif 管理プログラムの起動方法、および使用方法の概要については、78 ページの『Motif 管理プログラムによる Communications Server for Linux の管理』を参照してください。

Communications Server for Linux プログラムのパスの指定

Communications Server for Linux プログラムを実行するには、Communications Server for Linux 実行可能プログラムが入っているディレクトリーのパスを指定する必要があります。パスを指定するには、プログラムを初めて実行する前に環境変数 PATH にディレクトリーを追加するか、プログラムを実行するたびにディレクトリー名を指定します。

Motif 管理プログラムは **/opt/ibm/sna/bin/X11** ディレクトリーに保管され、その他のプログラムは **/opt/ibm/sna/bin** に保管されています。**.login** ファイルまたは **.profile** ファイルの環境変数 PATH の定義に、これらのディレクトリーを追加すると、Communications Server for Linux により自動的にプログラムが検出されます。あるいは次のように、プログラムの実行時にディレクトリー名を指定することもできます。

```
/opt/ibm/sna/bin/sna start
```

`/opt/ibm/sna/bin/X11/xsnaadmin`

本書に記載されているコマンド行の例では、環境変数 `PATH` にディレクトリーを追加していることを前提としているため、ディレクトリー名は含まれていません。

Communications Server for Linux の使用可能化

ローカル・ノードの構成や管理を行うには、Communications Server for Linux が使用可能な状態でなければなりません。他の X/Motif アプリケーションと同様、適切な X サーバーを指定するため、環境変数 `DISPLAY` を設定する必要があります。

Communications Server for Linux を使用可能にするには、Linux コマンド・プロンプトに、次のコマンドを入力します。

sna start

Communications Server for Linux をインストールすると、インストール・ユーティリティーは、スタートアップ・ファイル `/etc/rc.d/init.d/snastart` を自動的に更新して **sna start** コマンドを組み込みます。これによって、Communications Server for Linux は確実に、システム起動時に自動的に始動します。Communications Server for Linux が自動的に始動しないようにしたい場合は、この行を除去するかコメント化し、次にこのセクションの指示に従って、Communications Server for Linux ソフトウェアを手動で使用可能にしてください。

標準エラー表示 (通常は端末の画面) に、Communications Server for Linux が初期化中であることを示すメッセージと、初期化が正常に終了したかどうかを示すメッセージが書き込まれます。

Motif 管理プログラムによる Communications Server for Linux の管理

Communications Server for Linux で Motif 管理プログラムを使用するには、まず最初に、『Communications Server for Linux の使用可能化』で説明されているように、Communications Server for Linux が初期化されていることを確認します。(適切な X サーバーを指定するために、環境変数 `DISPLAY` も設定する必要がある場合もあります。)

Motif 管理プログラムをバックグラウンドで始動するには、次のコマンドを入力します。

xsnaadmin &

Communications Server for Linux は「Domain (ドメイン)」ウィンドウを表示します。このウィンドウでは、すべての定義済みノードが表示され、ノードの開始と停止を実行できます。いずれかのノードをダブルクリックすると、そのノードの「Node (ノード)」ウィンドウが表示されます (80 ページの図 4 を参照)。

「Node (ノード)」ウィンドウには、ノードとそのリソースに関する情報が表示されます。ノードを構成していない場合は、管理プログラムから、ノードを構成するようという指示が出されます。ノードの構成については、84 ページの『ノードの構成』で説明します。

注: 本書では、ウィンドウという用語は、Communications Server for Linux リソースの情報を表示する Motif ウィンドウを意味します。ウィンドウは、1 つ以上のセクションあるいはペイン (画面区画) からなります。ダイアログは、ユーザーが情報を入力できる Motif ウィンドウです。

「Node (ノード)」ウィンドウには必要な情報のほとんどが表示されており、どの情報へも簡単にアクセスできるようになっています。このウィンドウには、ローカル・ノード上の主要なリソースがすべて表示されています。

複数のノードを持つ Client/Server Communications Server for Linux システムを構成する場合は、本章の手順を実行して、各ノードを順に構成します (「ドメイン (Domain)」ウィンドウに戻って、次のノードを選択します)。

その他のウィンドウへは、「Node (ノード)」ウィンドウの「**Windows (ウィンドウ)**」メニューからアクセスできます。このようなウィンドウには、次のものがあります。

- 「LU Pools (LU プール)」ウィンドウ
- 「CPI-C Destination Names (CPI-C 宛先名)」ウィンドウ

「Node (ノード)」ウィンドウの「**Services (サービス)**」メニューでは、リソースの追加が容易に実行できるようになっています。また構成作業および管理作業のヘルプも表示されます。「**Diagnostics (診断)**」メニューからは、「Logging (ロギング)」ダイアログと「Tracing (トレース)」ダイアログに進むことができます。

「Node (ノード)」ウィンドウ

「Node (ノード)」ウィンドウの例を 80 ページの図 4 に示します。タイトル・バーには、Linux システムの名前が表示されます。

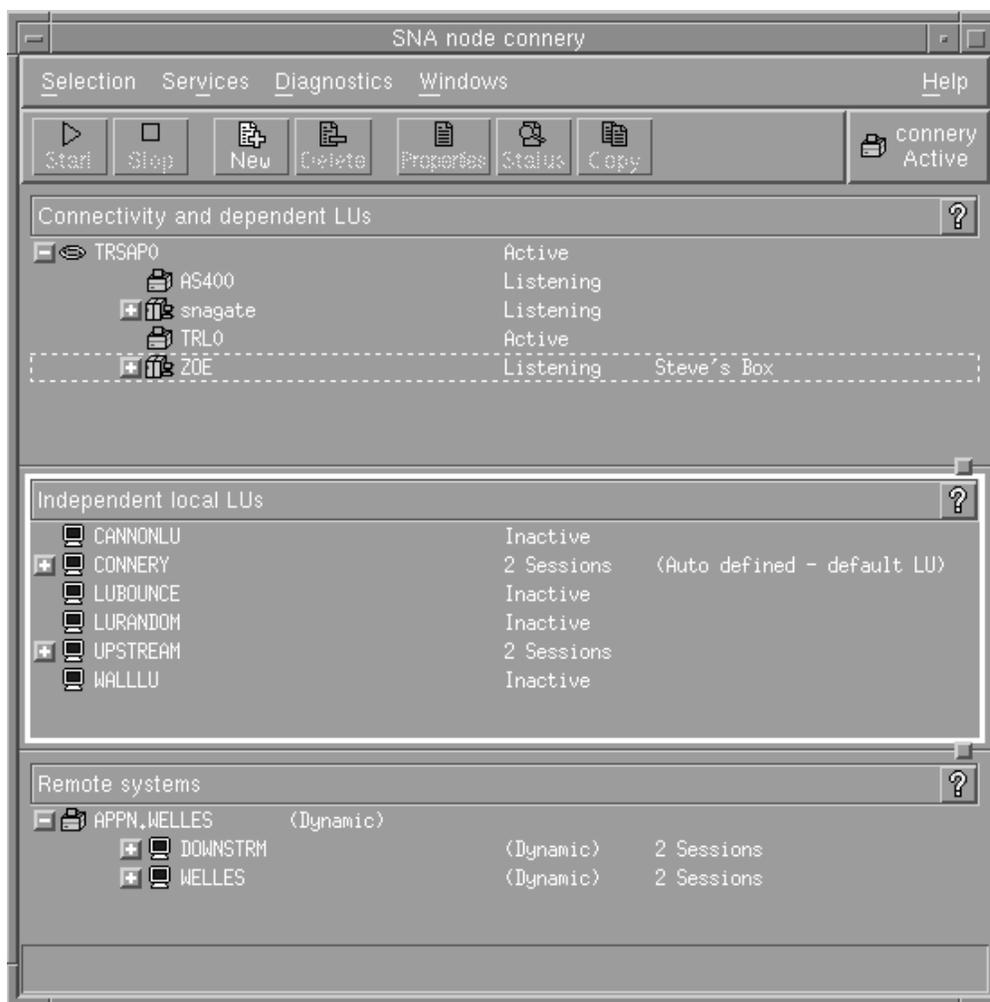


図4. 「Node (ノード)」ウィンドウ

「ノード (Node)」ウィンドウでは、Communications Server for Linux ノードのすべてのリソースとコンポーネントの構成と管理を行うことができます。

- ポート
- リンク・ステーション
- タイプ 0 から 3 の LU およびタイプ 6.2 の従属 LU
- DLUR 内部 PU
- 独立型ローカル LU
- リモート・ノード
- パートナー LU

「Node (ノード)」ウィンドウでは、上記のリソースの追加、削除、変更、および管理を行うことができます。ウィンドウに表示されるリソースのレイアウトでリソース間の関係が示されます。また、どのリソースが表示されるのかを制御することができます。

ポート、ローカル LU、リモート・ノードは常に表示されます。「Node (ノード)」ウィンドウには、各リンク・ステーションがその親ポートの下に表示され、各従属

LU がその親リンク・ステーションの下に表示されます。また、パートナー LU はローカル LU とリモート・ノードの下に表示されます。

「Node (ノード)」ウィンドウでは、ノードのリソースがタイプ別に各セクションにまとめられています。

- 「Node (ノード)」ウィンドウの右上隅にある「Node (ノード)」ボックスには、ノードが Active または Inactive のいずれであるかが示されます。
- 「Node (ノード)」ウィンドウの最上部のペイン (「Connectivity (接続)」ペイン) には、ポート、リンク・ステーション、各ポートの PU、特定のリンク・ステーションまたは PU の従属 LU など、ノードの接続リソースのリストが表示されます。各リソースごとの現行の状況情報も表示されます。
- 中央のペイン (「Independent Local LUs (独立型ローカル LU)」ペイン) には、ローカル・ノード上で定義されている独立 LU が表示されます。このウィンドウには、特定の LU を使用するセッションに関する情報と、パートナー LU にアクセスするために使用されるリンク・ステーションによる、パートナー LU の位置を定義するレコードも表示されます。
- 最下部のペイン (「Remote Systems (リモート・システム)」ペイン) には、リモート・ノードとパートナー LU に関する情報が表示されます。また、各リモート・ノードまたは各パートナー LU のセッション情報も表示されます。

ペインを選択するには、ペインをクリックします。ペインの中に表示されている特定のリソースを選択するには、そのリソースの行をクリックします。特定の項目の構成を表示または変更するには、その項目をダブルクリックします。(特定のリソースの構成情報にアクセスする場合は、このウィンドウのボタンとメニューを使用してアクセスできます。)

リストに表示されている項目ごとに、その項目に属するリソースが、その項目の情報内でネストされて表示されます。たとえば、リンク・ステーションは、属しているポートの下にまとめられます。項目のリソースが表示されていない場合には、その項目の横にある「**Expand (展開)**」ボタン (+) をクリックすると表示されます。また、「**Contract (縮小)**」ボタン (-) をクリックするとリソースを非表示にできます。

「Node (ノード)」ウィンドウから次の管理タスクを実行できます。

リソースの始動と停止

リソースを選択して、「**Start (開始)**」ボタン、または「**Stop (停止)**」ボタンをクリックします。(あるいは「**Selection (選択)**」メニューから、「**Start item (項目の始動)**」または、「**Stop item (項目の停止)**」を選択します。)

項目にリソースを追加

項目を選択して、「**New (新規)**」ボタンをクリックします(あるいは、「**Selection (選択)**」メニューから「**New (新規)**」を選択します)。たとえば、ポートのリンク・ステーションを追加するには、ポートを選択してから「**New (新規)**」ボタンをクリックします。

リソースの削除

項目を選択して、「**Delete (削除)**」ボタンをクリックします(あるいは、「**Selection (選択)**」メニューから「**Delete (削除)**」を選択します)。

リソース構成の表示または変更

リソースを選択して、「**Properties (プロパティ)**」ボタンをクリックします (あるいは、「**Selection (選択)**」メニューから、「**Properties (プロパティ)**」を選択します)。

リソースの状況情報の取得

項目を選択して、「**Status (状況)**」ボタンをクリックします (あるいは、「**Selection (選択)**」メニューから「**Status (状況)**」を選択します)。

リソース構成のコピー

リソースを選択して、「**Copy (コピー)**」ボタンをクリックします (あるいは、「**Selection (選択)**」メニューから「**Copy (コピー)**」を選択します)。

さらに、ノードの特定の構成タスクの選択 («**Services (サービス)**」メニュー)、ロギング (ドメインの場合) とトレース (ノードの場合) の制御 («**Diagnostics (診断)**」メニュー)、ドメイン・リソースの表示、または変更 («**Windows (ウィンドウ)**」メニューで項目を 1 つ選択) などの操作が可能です。

リソース項目

ウィンドウに表示されるリソースのレイアウトによって、リソース間の関係が示されます。

項目に 1 つ以上の子項目が関連付けられている場合、項目の隣に「**Expand (展開)**」記号 (+)、または「**Contract (縮小)**」記号 (-) が表示されます。

- 「**Expand (展開)**」記号は、関連付けられている子項目が隠されていることを示します。隠れている子項目を表示するには、「**Expand (展開)**」記号をクリックするか、数字キーパッドの + キーを押します。
- 「**Contract (縮小)**」記号は、子項目が表示されていることを示します。表示されている子項目を隠すには、「**Contract (縮小)**」記号をクリックするか、数字キーパッドの - キーを押します。
- 項目の隣にどちらの記号も表示されていない場合は、関連する子リソースがないことを示します。

たとえば、特定のポートに関連付けられているリンク・ステーションがあるとします。「**Node (ノード)**」ウィンドウの「**Connectivity (接続)**」ペインでは、このリンク・ステーションが、同じ親ポートに関連付けられているその他のすべてのリンク・ステーションと共に、親ポートの下に表示されます。ポートは常に表示されますが、関連付けられているリンク・ステーションについては、表示または非表示を選択できます。同様に、関連付けられている LU のリストがあるリンク・ステーションの場合も、展開して LU を表示することも、縮小して LU を隠すこともできます。

親リソースは必ず子リソースよりも前に構成しなければなりません。親リソースを削除すると、その子リソースもすべて削除されます。

ツールバー・ボタン

リソース・ウィンドウには、主な機能を簡単に実行できるように、ツールバー・ボタンが組み込まれています。83 ページの図 5 に、Communications Server for Linux のツールバーを示します。



図 5. Communications Server for Linux ツールバー

各リソース・ウィンドウのツールバーに、すべてのボタンが表示されるわけではありません。現在選択している項目に対して、ボタンの操作が無効な場合（あるいは、操作対象の項目を選択する必要があるが、選択されていない場合）は、ボタンの枠線がぼかし表示されます。この場合、ボタンの機能は選択できません（ボタンをクリックしても何も実行されません）。リソース・ウィンドウには、次のボタンが表示されます。

Start (開始)

選択した項目を始動します。

Stop (停止)

選択した項目を停止します。

New (新規)

新しいリソース項目を追加します。

Delete (削除)

選択したリソースを削除します。

Properties (プロパティ)

選択した項目の構成を表示および変更できるダイアログをオープンします。

Status (状況)

選択した項目の現在の状況を表示します。

Copy (コピー)

選択した項目をコピーします。このボタンをクリックすると、表示されるダイアログのフィールドに、選択された項目の構成が複製されます。新しいリソースを追加するには、ダイアログのフィールドに情報をすべて入力します（新しい項目の名前を入力します）。

ポートやリンク・ステーションなどの多くのリソースは、アクティブである場合には変更できません。ただし、アクティブなリソースのパラメータを表示することはできます。パラメータを表示するには、リソースを選択して、「**Properties (プロパティ)**」ボタンをクリックし、ダイアログをオープンします。完了したら、「**Close (クローズ)**」ボタンをクリックしてください。

クライアント/サーバー機能の構成

このセクションは、クライアント/サーバー環境で実行するために Communications Server for Linux をインストールしている（同一ネットワークで複数の Communications Server for Linux ノードを使用している）場合にのみお読みください。

クライアント/サーバー環境では、サーバーを構成サーバーとしてマークできます。Communications Server for Linux は、これらの構成サーバーのリストを維持します。リスト中の先頭のサーバーがマスター・サーバーであり、他のサーバーはすべてバックアップ・サーバーです。サーバーは順番にリストされていて、マスター・

クライアント/サーバー機能の構成

サーバーが使用不可の場合には、リスト中の 2 番目のサーバー (最初のバックアップ・サーバー) が引き継ぎ、マスター・サーバーも最初のバックアップ・サーバーも使用不可の場合には、リスト中の 3 番目のサーバー (2 番目のバックアップ・サーバー) が引き継ぎます。

ドメイン内のいずれかのノードが活動状態の場合は、ドメイン内で使用可能な先頭の構成サーバー (アクセスが可能で、かつ Communications Server for Linux ソフトウェアが稼働している先頭のサーバー) がマスター・サーバーになります。現在のマスターが使用不可になる (ネットワーク障害などでアクセスできないため、あるいは、そこで稼働している SNA ソフトウェアが停止したため) と、リスト中にある次に使用可能な構成サーバーが新しいマスターになります。

Communications Server for Linux は、マスターなしで稼働できます。このような状態は、構成サーバー・リスト中のどのサーバーにもアクセスできない場合に発生します。このような状態が発生した場合は、アクセス可能なサーバーについてのみ、ノード・リソースの表示および構成を行うことができます。

注: どのノードがマスター・サーバーとして機能するかを直接指示することはできません。マスター・サーバーは、構成サーバー・リストにノードが追加される順序に基づいて選択されます。あるサーバーをリストの先頭に移動したい場合は、他のすべてのノードをリストから除去してから再び追加します。

Motif 管理プログラムの「Domain (ドメイン)」ウィンドウで「**Selection (選択)**」メニューから「**Make configuration server (構成サーバーの作成)**」を選択することによって、構成サーバーを追加できます。サーバーは、リストの末尾に追加され、他のすべてのサーバーが使用不可である場合にのみマスターになります。サーバーを除去するには、「**Selection (選択)**」メニューから「**Remove configuration server (構成サーバーの除去)**」を選択します。

注: リストされているあるサーバーが、Communications Server for Linux ソフトウェアを実行している唯一のサーバーであれば、そのサーバーを削除することはできません。なぜならば、この場合、マスター・サーバーとして引き継ぐことができるサーバーがほかにはないからです。クライアント/サーバー構成には、少なくとも 1 つの使用可能なマスター・サーバーが必要です。

クライアント/サーバー Communications Server for Linux システムの構成と管理についての詳細は、「*Communications Server for Linux 管理ガイド*」を参照してください。本書では、拡張クライアント/サーバー構成に関する情報 (クライアントおよびサーバーを異なる Communications Server for Linux ドメインに移動する方法、およびクライアント操作の詳細を構成する方法を含む) も記載しています。

ノードの構成

システム上で Communications Server for Linux を構成する最初の手順は、ローカル・ノードを構成することです。ノードの構成により、SNA ネットワーク内で通信するためにノードに必要な基本情報が作成されます。ノードの接続やその他のリソースを定義する前に、まずノードを構成する必要があります。

ノードがすでに構成されている場合には、この項で説明する手順に従ってノードの構成を変更できます。ただし、構成を変更する前に、必ずノードを停止させてください。

ノードを構成する前に、APPN ノードとして構成するか、APPN 以外のノードとして構成するかを決定します。この決定は、通信相手になる他の SNA ノードの機能に基づきます。

図 6 に、ホスト・コンピューターと直接通信する Communications Server for Linux ノードを示します。

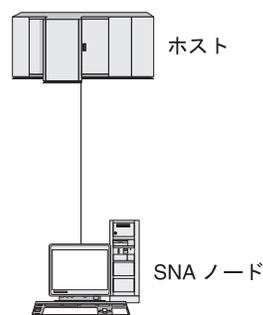


図 6. ホストと直接通信する *Communications Server for Linux* ノード

ホストで APPN がサポートされていない場合は、Communications Server for Linux ノードを LEN ノードとして構成してください。ホストが APPN をサポートする場合は、Communications Server for Linux ノードを APPN ネットワーク・ノードまたは APPN エンド・ノードのいずれかとして構成できます (Communications Server for Linux が SNA をホストとの通信にのみ使用している場合は、Communications Server for Linux ノードをエンド・ノードまたはブランチ・ネットワーク・ノードとして構成できます)。

86 ページの図 7 に、APPN ネットワークの各種 Communications Server for Linux ノードを示します。

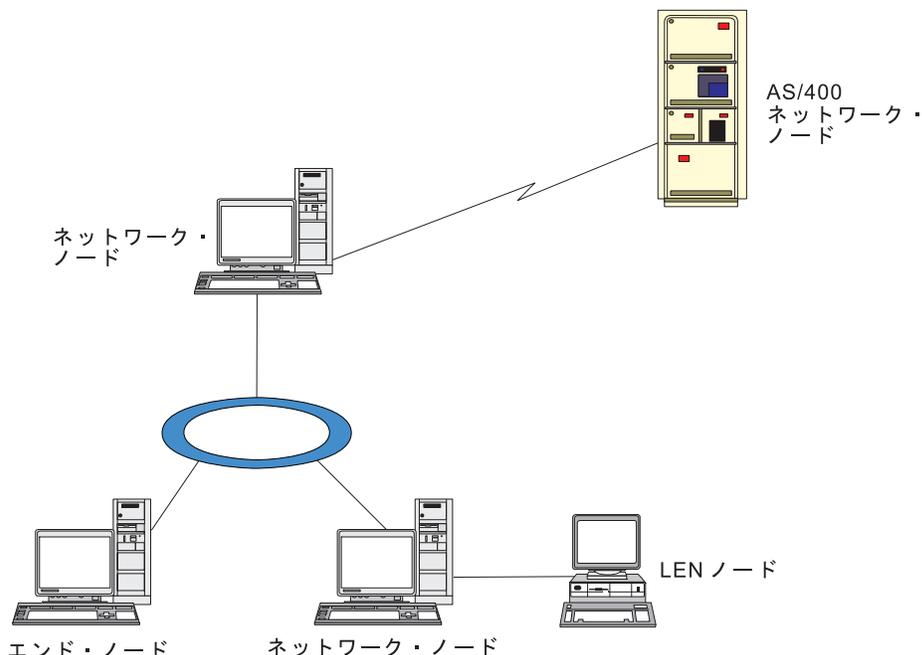


図7. APPN ネットワークの Communications Server for Linux ノード

ローカル・ノードが APPN ネットワークの一部であり、他のノードに対して APPN ルーティング・サービスを提供する場合は、それを APPN ネットワーク・ノードとして構成します。他のノードがルーティング・サービスを提供する場合は、そのローカル・ノードを APPN エンド・ノードとして構成します。ローカル・ノードが、直接接続された 1 つのノード (タイプは問いません) のみと通信を行う場合、このノードを LEN ノードとして構成します。

ノードを構成する前に、次の情報を収集してください。

- APPN サポートのタイプ (ネットワーク・ノード、分岐ネットワーク・ノード、エンド・ノード、LEN ノード)
- 制御点名 (別名が異なる場合は、別名も)。制御点名を確認するには、ネットワーク計画担当者に問い合わせてください。
- デフォルトのノード ID。(個々の通信リンクを構成するときに、このデフォルトをオーバーライドできます。)

ノードを構成するには、「Node (ノード)」ウィンドウから次の手順を行います。

1. 「Services (サービス)」メニューから「Configure node parameters (ノード・パラメーターの構成)」を選択するか、「Node (ノード)」ウィンドウの右上隅にある「Node (ノード)」ボックスをダブルクリックします。Communications Server for Linux は「Node Parameters (ノード・パラメーター)」ダイアログを表示します。
2. APPN サポートのレベル、制御点名、デフォルトのノード ID (必要な場合) を指定します。
3. 「OK」ボタンをクリックします。これでノードが定義されます。ノードを定義すると、Communications Server for Linux は、制御点と同じ名前を持つデフォルトの LU を自動的に定義します。

入力した値を保管しないで終了するには、「**Cancel (取り消し)**」ボタンをクリックします。

接続の構成

他のノードと通信する **Communications Server for Linux** ノードの場合、少なくとも 1 つの隣接ノードとの接続を構成する必要があります。接続リンクは、従属トラフィックと独立トラフィックのいずれか、または両方をサポートするように構成できます。

コンピューターには、1 つ以上のリンク・プロトコル用にアダプター・カードを装着できます。接続の構成に入力する必要がある情報の大部分は、使用しているリンク・プロトコルに応じて決まります。**Communications Server for Linux** でサポートされているリンク・プロトコルのリストについては、22 ページの『インストール要件』を参照してください。

リンクを構成するには、ポートとリンク・ステーション (ほとんどの場合) を定義する必要があります。**Motif** 管理プログラムを使用している場合は、ポート構成の一部として **DLC** (データ・リンク制御) が自動的に構成されます。さらに、接続ネットワークの一部としてポートを定義することもできます。

構成する必要があるリンクは、使用しているネットワークが **APPN** ネットワークであるかどうかと、現在の構成しようとしている内容によって異なります。必要な情報は、リンク・プロトコルによって異なり、また、リンクが従属トラフィックと独立トラフィックのいずれかまたはその両方をサポートするかによっても異なります。

このセクションでは、例として次のタイプのリンクの構成方法について説明します。

- **SDLC** 回線を使用してホスト・システムとの従属トラフィックをサポートするリンク
- イーサネット・リンク・プロトコルを使用している **APPN** ネットワークへの従属トラフィックと独立トラフィックの両方をサポートするリンク。この例では、イーサネット・ポート上で接続ネットワークも定義します。
- **APPN** ネットワークへのエンタープライズ・エクステンダー・リンク (ただし、エンタープライズ・エクステンダー・リンクがサポートするのは、独立トラフィックのみです)。

その他のリンク・プロトコルについては、「*Communications Server for Linux* 管理ガイド」または **Motif** 管理プログラムのオンライン・ヘルプを参照してください。

従属トラフィック用の **SDLC** リンクの構成

SDLC (同期データ・リンク制御) ポートについて、次の情報が必要となります。

- **SNA** ポート名 (通常、デフォルト名を使用できます)。また、**SDLC** デバイス番号も確認しておく必要があります。
- ノード始動時にポートを自動的にアクティブにするかどうか。
- 回線の種類 (交換回線 (発呼、着呼)、専用回線)

接続の構成

- リンク・ロール (1 次、2 次、折衝可能、1 次マルチドロップ、2 次マルチドロップ)
- ポーリング・アドレス (1 次ポート以外のポートの交換着呼回線の場合のみ)。その他のタイプのポートについては、リンク・ステーションのポーリング・アドレスを構成します。

SDLC リンク・ステーションについて、次の追加情報が必要となります。

- アクティブ化の方法 (管理者が行う、ノード始動時に行う、要求時に行う)
- サポートされているトラフィックのタイプ (「従属トラフィックのみ」など)
- リモート・ノード・ロール (ホストなど)

SDLC リンクを構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. ポートを構成します。
 - a. ウィンドウの「Connectivity (接続)」ペインを選択します。
 - b. 「**Services (サービス)**」メニューの、「**Connectivity (接続)**」サブメニューから、「**New port (新規ポート)**」を選択します (あるいは、ボタン・バーの「**New (新規)**」ボタンをクリックします)。
 - c. その結果表示されるダイアログのオプション・メニューから、プロトコルのタイプを選択し、次にポートの定義を選択します。

「**OK**」ボタンをクリックすると、Communications Server for Linux は「SDLC ポート (SDLC Port)」ダイアログを表示します。

- d. ダイアログのフィールドに適切な値を入力します。
- e. 「**OK**」ボタンをクリックします。これでポートが定義されます。

「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインに、ポートが表示されます。

2. ポートのリンク・ステーションを定義します。
 - a. 「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインで、リンク・ステーションの追加先のポートが選択されていることを確認します。
 - b. 「**Services (サービス)**」メニューの、「**Connectivity (接続)**」サブメニューから、「**New link station (新規リンク・ステーション)**」を選択します (あるいは、ボタン・バーの「**New (新規)**」ボタンをクリックします)。
 - c. 「**OK**」ボタンをクリックします。

Communications Server for Linux によって「SDLC Link Station (SDLC リンク・ステーション)」ダイアログが表示されます。

- d. ダイアログのフィールドに適切な値を入力します。
- e. 「**OK**」ボタンをクリックすると、リンク・ステーションが定義されます。

リンク・ステーションは、「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインに表示されている、所属先ポートの下に表示されます。

従属トラフィックと独立トラフィックをサポートするイーサネット・リンクの構成

この例では、APPN ネットワークへの従属トラフィックと独立トラフィックの両方をサポートするイーサネット・リンクの構成方法を説明します。また、イーサネット・ポート上で接続ネットワークも定義します。

イーサネット・ポートについて、次の情報を収集する必要があります。

- SNA ポート名 (通常、デフォルト名を使用できます)。複数のイーサネット・ネットワーク・アダプター・カードがある場合は、イーサネット・カード番号も確認しておく必要があります。また、ローカル SAP (サービス・アクセス・ポイント) 番号も、指定する必要があります (Intel および OSA2 アダプターの場合、通常は 04 です)。OSA Express アダプターの場合、ローカル SAP 番号は、この Linux イメージ上の ethX インターフェースに対応する入出力装置アドレスとして、OSA/SF で定義されている番号と一致していなければなりません。
- ノード始動時にポートを自動的にアクティブにするかどうか。
- 接続ネットワーク名 (1 つの接続ネットワーク内では、すべてのポートで同一の名前が使用されていなければなりません)。

イーサネット・リンク・ステーションについて、次の追加情報を収集する必要があります。

- アクティブ化の方法 (管理者が行う、ノード始動時に行う、要求時に行う)
- サポートされているトラフィックのタイプ (この例では、従属トラフィックと独立トラフィックの両方)
- リモート・ノードの制御点名 (LEN ノードの場合のみ必要)
- リモート・ノードのタイプ (ネットワーク・ノード、エンド・ノード、または自動検出)
- リモート・ノード・ロール (ダウンストリーム SNA ゲートウェイ、またはパススルー DLUR など)
- 選択リンク・ステーションを構成するには、リモート・ステーションの MAC (メディア・アクセス制御) アドレスと、SAP 番号 (通常は 04) が必要です。アドレス情報がない場合に、「*Activation* (アクティブ化)」フィールドに、「By administrator (管理者)」を指定すると、リンク・ステーションは非選択 listen リンク・ステーションになります。

イーサネット・リンクを構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. ポートを構成します。
 - a. ウィンドウの「Connectivity (接続)」ペインを選択します。
 - b. 「Services (サービス)」メニューの、「Connectivity (接続)」サブメニューから、「New port (新規ポート)」を選択します (あるいは、ボタン・バーの「New (新規)」ボタンをクリックします)。
 - c. その結果表示されるダイアログのオプション・メニューから、プロトコルのタイプを選択し、次にポートの定義を選択します。

「OK」ボタンをクリックします。Communications Server for Linux は、「イーサネット SAP (Ethernet SAP)」ダイアログを表示します。

- d. ダイアログのフィールドに適切な値を入力します。
- e. 「OK」 ボタンをクリックします。これでポートが定義されます。

「Node (ノード)」 ウィンドウの「Connectivity (接続)」 ペインに、ポートが表示されます。

2. ポートのリンク・ステーションを定義します。
 - a. 「Node (ノード)」 ウィンドウの「Connectivity (接続)」 ペインで、リンク・ステーションの追加先のポートが選択されていることを確認します。
 - b. 「Services (サービス)」 メニューの、「Connectivity (接続)」 サブメニューから、「New link station (新規リンク・ステーション)」 を選択します (あるいは、ボタン・バーの「New (新規)」 ボタンをクリックします)。
 - c. 「OK」 ボタンをクリックします。

Communications Server for Linux は、「イーサネット・リンク・ステーション (Ethernet Link Station)」 ダイアログを表示します。

- d. ダイアログのフィールドに適切な値を入力します。
- e. 「OK」 ボタンをクリックすると、リンク・ステーションが定義されます。

リンク・ステーションは、「Node (ノード)」 ウィンドウの「Connectivity (接続)」 ペインに表示されている、所属先ポートの下に表示されます。

エンタープライズ・エクステンダー・リンクの構成

この例では、APPN ネットワークへの、エンタープライズ・エクステンダー・リンクの構成方法を説明します。エンタープライズ・エクステンダー・リンクがサポートするのは、独立 LU トラフィックのみであることに注意してください。

エンタープライズ・エクステンダー・ポートについて、次の情報を収集する必要があります。

- SNA ポート名 (通常、デフォルト名を使用できます)。IP を実行しているネットワーク・アダプター・カードが複数ある場合、使用したい IP インターフェースも指定する必要があります (例: eth0)。
- ノード始動時にポートを自動的にアクティブにするかどうか。

エンタープライズ・エクステンダー・リンク・ステーションについて、次の追加情報を収集する必要があります。

- アクティブ化の方法 (管理者が行う、ノード始動時に行う、要求時に行う)
- リモート・ノードのタイプ (ネットワーク・ノード、エンド・ノード、または自動検出)
- 選択リンク・ステーションを構成するには、リモート・ステーションの IP ホスト名または IP アドレスが必要です。この情報を提供せず、「Activation (アクティブ化)」 フィールドで「By administrator (管理者)」を指定すると、リンク・ステーションは非選択 listen リンク・ステーションになります。

エンタープライズ・エクステンダー・リンクを構成するには、「Node (ノード)」 ウィンドウで次の手順を行います。

1. ポートを構成します。

- a. ウィンドウの「Connectivity (接続)」ペインを選択します。
- b. 「Services (サービス)」メニューの、「Connectivity (接続)」サブメニューから、「New port (新規ポート)」を選択します (あるいは、ボタン・バーの「New (新規)」ボタンをクリックします)。
- c. その結果表示されるダイアログのオプション・メニューから、プロトコルのタイプを選択し、次にポートの定義を選択します。

「OK」ボタンをクリックすると、Communications Server for Linux は「IP ポート (IP Port)」ダイアログを表示します。

- d. ダイアログのフィールドに適切な値を入力します。
- e. 「OK」ボタンをクリックします。これでポートが定義されます。

「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインに、ポートが表示されます。

2. ポートのリンク・ステーションを定義します。
 - a. 「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインで、リンク・ステーションの追加先のポートが選択されていることを確認します。
 - b. 「Services (サービス)」メニューの、「Connectivity (接続)」サブメニューから、「New link station (新規リンク・ステーション)」を選択します (あるいは、ボタン・バーの「New (新規)」ボタンをクリックします)。
 - c. 「OK」ボタンをクリックします。

Communications Server for Linux は、「IP リンク・ステーション (IP Link Station)」ダイアログを表示します。

- d. ダイアログのフィールドに適切な値を入力します。
- e. 「OK」ボタンをクリックすると、リンク・ステーションが定義されます。

リンク・ステーションは、「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインに表示されている、所属先ポートの下に表示されます。

タイプ 0 から 3 LU の構成

タイプ 0 から 3 の LU を使用するユーザー・アプリケーションをサポートするには、従属 LU を構成する必要があります。従属 LU を構成する前に、次の構成を実行する必要があります。

- 84 ページの『ノードの構成』の説明に従って、ノードを構成します。
- 87 ページの『接続の構成』の説明に従って、従属 LU トラフィックをサポートするリンクを構成します。

SNA ゲートウェイを使用する別のノードにアップストリーム・リンクを有している場合、または DLUR を使用している場合は、ホストに対する直接リンクを構成する必要はありません。詳細については、103 ページの『SNA ゲートウェイの構成』および 106 ページの『DLUR の構成』を参照してください。

ホスト・システムとの通信をサポートするには、タイプ 0 から 3 の従属 LU を構成する必要があります。このセクションにある説明に従って、LUA、DLUR、また

タイプ 0 から 3 LU の構成

は PU コンセントレーションをサポートする LU を定義できます。また、多数の LU を定義しておくことで、1 回の操作で同じタイプの複数の LU を構成できます。

LU を定義するときその LU をプールに割り当てるか、前に定義済みの LU をプールに割り当てることによって、必要な場合に使用される LU のプールを定義できます。

タイプ 0 から 3 LU の定義

3270 LU を構成する前に、次の情報を収集してください。

- LU 名。(これはローカル ID であり、ホスト構成に一致する必要はありません。)
- LU 番号 (ある範囲の複数の LU の場合は複数の番号)
- LU タイプ (3270 ディスプレイ・モデルまたは 3270 プリンター)
- プール名 (LU をプールに追加する場合)

前に定義済みのリンク・ステーションについて、タイプ 0 から 3 の LU を構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. このウィンドウの「Connectivity (接続)」ペインから、ホスト接続リンク・ステーションを選択します。
2. 「New (新規)」ボタンをクリックします。
3. 上記の操作の結果表示されたダイアログで、LU タイプ (「New 3270 display LU (新規 3270 ディスプレイ LU)」、または「New 3270 printer LU (新規 3270 プリンター LU)」) を選択します。

この項目を選択して「OK」をクリックすると、Communications Server for Linux は「LU タイプ 0 - 3 (LU Type 0-3)」ダイアログを表示します。

4. ダイアログのフィールドに適切な値を入力します。
5. 「OK」をクリックすると、LU が定義されます。

LU は、「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインに表示されている、ホスト接続リンク・ステーションの下に表示されます。

LU プールの定義

LU タイプ 0 から 3 に対して、LU プールを定義すると、ユーザー構成が単純化され、ホスト・セッションの確立の柔軟性が大幅に向上します。たとえば、いくつかの LU を 1 つの LU プール内に定義し、次に、この LU プールを使用する複数のユーザーを構成することができます。このようにすると、ユーザーのセッションを容易に構成でき、任意のセッションがプール内の任意の LU を使用できます。

注: ユーザーのセッションは、特定の LU または LU プールのいずれかに割り当てることができます。

- ユーザーのセッションをプール内の特定の LU に割り当てると、セッションはこの LU が使用可能であればこれを使用します。それ以外の場合、セッションはそのプール内のフリーのいずれかの LU を、特定の LU の代わりにそれが LU プールに割り当てられたかのように使用します。

- 指定された LU だけをユーザーが使用するようにして、その LU が使用中であればユーザーのセッションが確立できないようにしたい場合、その LU がプール内に入っていないようにしてください。

ローカル Communications Server for Linux ノードの LU プールは、「LU プール (LU Pools)」ウィンドウで確認できます。このウィンドウには、ローカル・システム上に構成されている LU プールがリストされます。また、LU プールに追加する LU を選択することもできます。

プールに追加できる LU タイプを次に示します (1 つのプール内では異なるタイプの LU を混在させないでください)。

- 3270 display LU (3270 ディスプレイ LU)
- Unrestricted LU (制限なしの LU)

LU は、プールに追加する前に必ずローカル・ノードで定義してください。

LU プールを構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. 「**Windows (ウィンドウ)**」メニューから、「**LU Pools (LU プール)**」を選択します。

Communications Server for Linux は「LU Pools (LU プール)」ウィンドウを表示します。

2. 「**New (新規)**」ボタンをクリックします。

Communications Server for Linux は「LU Pool Configuration (LU プール構成)」ダイアログを表示します。

右側のボックスに、まだプールに割り振られていない LU のリストが表示されます。このリストに表示されている LU はすべて、新しいプールに割り振ることができます。

3. プールに追加する LU (1 つ以上) を選択してから、「**New (新規)**」ボタンをクリックします。選択された LU が左側のボックスへ移動します。

左側のボックスから LU を除去するには、LU を選択して「**Remove (除去)**」ボタンをクリックします。

4. 「**OK**」をクリックすると、LU プールが定義されます。

左側のボックスに表示されている LU がすべて LU プールに追加されます。

定義されたプールが「LU Pools (LU プール)」ウィンドウに表示されます。

APPC 通信の構成

APPC アプリケーションと CPI-C アプリケーションを使用する場合には、まず最初に APPC を構成する必要があります。APPC アプリケーションは、ホストまたは対等コンピューター上の他の APPC アプリケーションまたは CPI-C アプリケーションと、指定されたモードで通信を行う際に、ノードの LU タイプ 6.2 リソースを使用します。

APPC 通信を構成する前に、次の構成を行ってください。

1. 84 ページの『ノードの構成』の説明に従って、ノードを構成します。
2. 87 ページの『接続の構成』の説明に従って、接続を構成します。

これ以降の構成手順は、構成でサポートされているトラフィックの種類（従属トラフィックと独立トラフィックのいずれか、または両方）によって異なります。

独立 APPC

独立 APPC では独立 LU が使用されます。各 LU-LU セッションでは、ローカル LU とパートナー LU が使用されます。

ローカル LU の場合、ノード制御点に関連付けられている事前定義のデフォルト LU を使用できます。あるいは新しいローカル LU を構成することもできます。

Communications Server for Linux ノードが APPN ネットワーク内のエンド・ノードまたはネットワーク・ノードの場合は、APPN により LU が動的に検出されるため、パートナー LU を構成する必要はありません。ただし、APPN ネットワーク以外のネットワークの場合や LEN ノードの場合は、パートナー LU を構成する必要があります。この場合、まずパートナー LU があるリモート・ノードを構成してから、このリモート・ノード上のパートナー LU を定義してください。

従属 APPC

リモート・ノードが、独立 LU 6.2 をサポートしないホストである場合は、従属トラフィックをサポートするように構成します。従属 APPC では、ローカル LU を構成する必要があります。

アプリケーションで CPI-C が使用される場合には、APPC の構成後にさらに、CPI-C を構成しなければならないことがあります（101 ページの『CPI 通信のための構成』を参照してください）。CPI-C アプリケーションは、ホストまたは対等コンピューター上の他の APPC アプリケーションまたは CPI-C アプリケーションと通信を行う際に、ノードの LU タイプ 6.2 およびモードのリソースを使用します。CPI-C アプリケーションと APPC アプリケーションには、同じリソースを定義します。また、Communications Server for Linux コンピューターの TP が始動 TP、つまりソース TP（会話を開始する TP）の場合、TP に対して 1 つ以上のサイド情報エントリを定義しなければならないことがあります。この手順については、101 ページの『CPI 通信のための構成』で説明します。各エントリには、パートナー TP の情報、パートナー TP にアクセスするために使用される LU やその他のリソースの情報、必須セキュリティー情報が示されます。

この節では、単純な APPN ネットワーク（独立 LU 6.2 を使用）の構成方法について説明します。このネットワークは、95 ページの『単純な APPN ネットワークの構成』に説明があるように、1 つのネットワーク・ノード、1 つのエンド・ノード、1 つの LEN ノードで構成されています。（この例では、2 つのノード間の CP-CP セッションの状況情報の取得方法についても説明します。）

また、この節では、100 ページの『従属 APPC の構成』で説明されている、従属 APPC 通信の構成方法についても説明します。

どちらの例でも、APPC セッションで、標準モードとサービス・クラス (COS) が使用されていることを前提としています。

モード、セキュリティー、始動可能な (ターゲット) TP など、その他の APPC 情報の構成方法については、「*Communications Server for Linux* 管理ガイド」を参照してください。

単純な APPN ネットワークの構成

最も単純な APPN ネットワークの構成は、APPN ネットワーク・ノードと APPN エンド・ノードの 2 つのノードのみで構成できます。ネットワーク・ノードは、エンド・ノードのためにセッション・ルーティングを処理します。

ネットワーク・ノードの構成

このシナリオでは、制御点 LU と標準モードを使用していること、および LAN リンク・タイプ (トークンリング、イーサネット) を使用していることを想定しています。この場合にネットワーク・ノードを構成する手順は次のとおりです。

1. 84 ページの『ノードの構成』の説明に従って、ノードを構成します。「*APPN support (APPN サポート)*」フィールドでは、値「*Network node (ネットワーク・ノード)*」を選択します。制御点名をメモしておいてください。
2. 87 ページの『接続の構成』の説明に従って、接続を構成します。独立トラフィックをサポートするリンクを構成します。

隣接するエンド・ノードからこのネットワーク・ノードへ通信するには、ネットワーク・ノードのポートの MAC アドレスと SAP 番号が必要です。Communications Server for Linux ノードの MAC アドレスを取得するには、次の手順を使用してください。

1. 「Node (ノード)」ウィンドウでポートを選択します。
2. 「**Start (開始)**」ボタンをクリックすると、ポートが始動します。
3. ポートの状況情報を取得するため、「**Status (状況)**」ボタンをクリックします。「Port Status (ポートの状況)」ダイアログに、MAC アドレスと SAP 番号が表示されます。
4. MAC アドレスと SAP 番号は、あとでエンド・ノードのリンク・ステーション構成ダイアログで入力する必要があるため、これらの値をメモしておいてください。

エンド・ノードの構成

このシナリオでは、制御点 LU と標準モードを使用していること、および LAN リンク・タイプ (トークンリング、イーサネット) を使用していることを想定しています。この場合にネットワーク・ノードを構成する手順は次のとおりです。

1. 84 ページの『ノードの構成』の説明に従って、ノードを構成します。「*APPN support (APPN サポート)*」フィールドでは、値「*End node (エンド・ノード)*」を選択します。
2. 87 ページの『接続の構成』の説明に従って、接続を構成します。独立トラフィックをサポートするリンクを構成して、リンク・ステーション用に次の情報を入力します。
 - 「*Remote node (リモート・ノード)*」フィールドへの値として、ネットワーク・ノードの名前 (『ネットワーク・ノードの構成』を参照) を入力します。

- リンク・ステーション構成ダイアログの、「Contact Information (接続情報)」ペインに、ネットワーク・ノードのポートの MAC アドレスと SAP 番号を入力します。

APPN ネットワークでは、隣接するネットワーク・ノードに接続する 1 つのリンク・ステーションを使用して、ネットワーク内の他のリモート・ノードと通信できます。このため、リモート・ノードごとに個別のリンク・ステーションを構成する必要はありません。

2 つのノード間の接続の確認

この例では、95 ページの『ネットワーク・ノードの構成』と 95 ページの『エンド・ノードの構成』の説明に従って、すでにネットワーク・ノードとエンド・ノードが構成済みであることを前提としています。エンド・ノードから次の手順を行います。

1. 「Node (ノード)」ウィンドウで、隣接するネットワーク・ノードに接続するリンク・ステーションを選択します。
2. 「Start (開始)」ボタンをクリックします。リンク・ステーションが始動します。

リンク・ステーションの始動時に、2 つのノード間での CP-CP セッションも自動的に確立されます。これらのセッションは、「Node (ノード)」ウィンドウの、「Independent Local LUs (独立型ローカル LU)」ペインに表示されます。

3. セッションの状況情報を取得するには、「Node (ノード)」ウィンドウでセッションを選択してから、「Status (状況)」ボタンをクリックします。

独立 APPC LU の構成

多くの場合、アプリケーションではローカル・ノードの制御点 LU を使用できます。この制御点 LU は、ノードの構成時に自動的に定義されます。これはデフォルトの LU です。アプリケーションでは、特定の LU が指定されない場合には、このデフォルト LU を使用できます。アプリケーションでデフォルト LU を使用する場合は、ローカル LU を定義する必要はありません。使用している APPC アプリケーションの資料を参照するか、アプリケーション・プログラマーに問い合わせてください。

独立 LU 6.2 を構成するには、次の情報を収集する必要があります。

- ローカル LU 名
- ローカル LU の別名 (この LU がサポートする TP で別名が使用されている場合)

独立型ローカル LU を構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. このウィンドウで「Independent Local LUs (独立型ローカル LU)」ペインを選択します。
2. 「Services (サービス)」メニューの「APPC」サブメニューから、「New independent local LU (新しい独立型ローカル LU)」を選択します (あるいは、「New (新規)」ボタンをクリックします)。

Communications Server for Linux は「Local LU (ローカル LU)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力します。
4. 「OK」ボタンをクリックして、ローカル LU を定義します。「Node (ノード)」ウィンドウの「Independent Local LUs (独立型ローカル LU)」ペインに、独立型 LU が表示されます。

LEN ノードのパートナー LU の構成

次の状況では、リモート・ノード (およびそのノード上のパートナー LU) を定義する必要があります

- ローカル・ノードが LEN ノードの場合、すべてのリモート・ノードと、このローカル・ノードが APPC を使用して通信を行うリモート・ノード上のパートナー LU を定義しなければなりません。LEN ノードはパートナー LU を動的に検出することはできませんが、リモート・ノードの定義により、それが可能になります。
- ローカル・ノードが APPN ネットワークの一部ではない場合 (たとえば、ネットワーク・ノード・サーバーを使用せずに、2 つのエンド・ノードを直接接続している場合) は、LU を動的に検出できません。この場合には、各パートナー LU を個別に構成する必要があります。
- リモート・ノードが LEN ノードであり、ローカル・ノードが、この LEN ノードのネットワーク・ノード・サーバーとして動作するネットワーク・ノードである場合、LEN ノード (およびそのパートナー LU) をネットワーク・ノード・サーバー上のリモート・ノードとして定義しなければなりません。このように定義することで、APPN ネットワークのその他のノードから、LEN ノードの LU を検出できるようになります。
- リモート・ノードが別の APPN ネットワークにある場合、このリモート・ノードは動的に検出できないため、定義しておく必要があります。

ローカル・ノードとリモート・ノードの両方が、同じ APPN ネットワークに接続している場合は、パートナー LU を定義しないでください。

リモート・ノード定義を追加すると、リモート・ノードと同じ名前のパートナー LU が自動的に追加されます。これは、リモート・ノードの制御点 LU となります。このパートナー LU をアプリケーションで使用する場合、その他のパートナー LU を追加する必要はありませんが、このパートナー LU に LU 別名を追加できます。別名を追加するには、パートナー LU をダブルクリックし、「Partner LU Configuration (パートナー LU の構成)」ダイアログに別名を入力します。

アプリケーションで、LU 別名を使用してパートナー LU を参照する場合には、パートナー LU 別名の定義を追加する必要があります。

ローカル・ノードまたはリモート・ノードのいずれかが LEN ノードの場合、LEN ノードでは LU を動的に検出できないため、パートナー LU をリモート・ノードの子として定義しなければなりません。アプリケーションで、リモート・ノードの制御点 LU をそのパートナー LU として使用する場合、リモート・ノードの定義時に自動的に制御点 LU を定義します。

Motif 管理プログラムでは、パートナー LU 別名の追加 (『パートナー LU 別名の定義』を参照)、特定のリモート・ノード上のパートナー LU の定義の追加 (99 ページの『リモート・ノードでのパートナー LU の定義』を参照)、ワイルドカードを使用した、複数のパートナー LU の定義 (99 ページの『ワイルドカードを使用した複数のパートナー LU の定義』を参照) を行うことができます。

リモート・ノードの定義: リモート・ノードを構成する前に、次の情報を収集する必要があります。

- ノードの SNA ネットワークの完全修飾名

リモート・ノードを構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. このウィンドウの「Remote Systems (リモート・システム)」ペインを選択します。
2. 「Services (サービス)」メニューの「APPC」サブメニューから、「New remote node (新規リモート・ノード)」を選択します (あるいは、ボタン・バーの「New (新規)」をクリックし、次に「Define remote node (リモート・ノードの定義)」を選択します)。

Communications Server for Linux は「Remote Node Configuration (リモート・ノード構成)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力します。
4. 「OK」ボタンをクリックすると、リモート・ノードが定義されます。「Node (ノード)」ウィンドウの「Remote Systems (リモート・システム)」ペインに、リモート・ノードが表示されます。

リモート・システムを定義すると、Communications Server for Linux は、リモート・ノードの制御点 LU をローカル・ノードのパートナー LU として自動的に定義します。

パートナー LU 別名の定義: パートナー LU の別名を定義するには、次の情報を収集する必要があります。

- パートナー LU の完全修飾名 (SNA ネットワーク名と LU 名)
- ローカル TP が使用するパートナー LU 別名

パートナー LU の別名を追加するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. このウィンドウの「Remote Systems (リモート・システム)」ペインを選択します。
2. 「Services (サービス)」メニューから、「APPC」、「New partner LUs (新規パートナー LU)」、「Partner LU alias (パートナー LU の別名)」を選択します (あるいは、ボタン・バーの「New (新規)」ボタンをクリックし、次に「Define partner LU alias (パートナー LU 別名の定義)」を選択します)。

Communications Server for Linux は「Partner LU Alias Configuration (パートナー LU 別名の構成)」ダイアログを表示します。

3. ダイアログにパートナー LU の名前と別名を入力します。

4. 「OK」ボタンをクリックすると、パートナー LU の別名が定義されます。
「Node (ノード)」ウィンドウの「Remote Systems (リモート・システム)」ペインに、(ネットワーク定義の一部として) パートナー LU の別名が表示されません。

リモート・ノードでのパートナー LU の定義: 特定のリモート・ノード上のパートナー LU を定義するには、次の情報を収集する必要があります。

- パートナー LU の完全修飾名
- パートナー LU の別名 (ローカル TP で別名が使用されている場合)
- パートナー LU のディレクトリー情報を含むノードの完全修飾名

特定のリモート・ノードのパートナー LU 定義を追加するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. リモート・ノードを選択します。
2. 「Services (サービス)」メニューから、「APPC」、「New partner LUs (新規パートナー LU)」、「Partner LU on remote node (リモート・ノードのパートナー LU)」を選択します (あるいは、ボタン・バーの「New (新規)」ボタンをクリックし、次に、「Define partner LU on remote node (リモート・ノードのパートナー LU の定義)」を選択します)。

Communications Server for Linux は「Partner LU Configuration (パートナー LU の構成)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力します。
4. 「OK」ボタンをクリックすると、パートナー LU が定義されます。パートナー LU の別名が、「Node (ノード)」ウィンドウの「Remote Systems (リモート・システム)」ペインで、この LU が所属するリモート・システムの下に表示されます。

ワイルドカードを使用した複数のパートナー LU の定義: 1 つのリモート・ノードにあり、名前が同じ文字で始まる複数のパートナー LU を構成する場合には、ワイルドカードを使用できます。ワイルドカードを使用すれば、パートナー LU を個別に構成する必要がなくなります。

ワイルドカードを使用して複数のパートナー LU を定義する際には、次の情報を入力する必要があります。

- ワイルドカード・パートナー LU 名。ワイルドカード・パートナー LU 名は、定義する複数のパートナー LU の完全修飾 LU 名に一致する、1 から 8 文字のタイプ A の EBCDIC スtring 2 つからなります。

最初の String には、パートナー LU の SNA ネットワーク名と厳密に一致する完全な SNA ネットワーク名か、ネットワーク名の先頭の文字に一致するワイルドカード接頭部のいずれかを使用できます。ネットワーク名にワイルドカード接頭部を入力する場合は、2 番目の String をブランクにしてください。

最初の String に完全な SNA ネットワーク名を入力した場合は、2 番目の値も入力できます。(この場合、最初の String に有効な SNA ネットワーク名

を指定していないと、2 番目のストリングに入力できません。) 2 番目のストリングは、パートナー LU の完全修飾名の後半の部分の先頭に一致する、ワイルドカード接頭部として処理されます。

- パートナー LU があるノードの名前。

複数のパートナー LU を追加するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. 定義するパートナー LU があるリモート・ノードを選択します。
2. 「Services (サービス)」メニューから、「APPC」、「New partner LUs (新規パートナー LU)」、「Wildcard partner on remote node (リモート・ノードのワイルドカード・パートナー)」を選択します (あるいは、ボタン・バーの「New (新規)」ボタンをクリックし、次に、「Define wildcard partner LUs on remote node (リモート・ノードのワイルドカード・パートナー LU の定義)」を選択します)。

Communications Server for Linux は「Wildcard Partner LU Configuration (ワイルドカード・パートナー LU の構成)」ダイアログを表示します。

3. ダイアログのフィールドに、適切な情報を入力します。
4. 「OK」ボタンをクリックすると、パートナー LU が定義されます。パートナー LU が、「Node (ノード)」ウィンドウの「Remote Systems (リモート・システム)」ペインで、この LU が属するリモート・ノードの下に表示されます。

従属 APPC の構成

従属 LU 6.2 を構成するには、次の情報を収集する必要があります。

- ローカル LU 名
- ローカル LU の別名 (この LU がサポートする TP で別名が使用されている場合)
- ホストへの接続を提供するリンク・ステーションの名前
- LU 番号
- LU を従属 LU 6.2 のデフォルト・プールに割り当てる必要があるかどうか

APPC アプリケーションまたは CPI-C アプリケーションで使用するために、タイプ 6.2 の従属 LU を構成している場合は、これらの LU をデフォルト・プールのメンバーとして定義することがあります。特定のローカル LU を指定しないアプリケーションには、デフォルト LU として定義されている LU のプールから、未使用の LU が割り当てられます。

従属型ローカル LU を構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. ウィンドウの「Connectivity (接続)」ペインでリンク・ステーションを選択します。
2. 「Services (サービス)」メニューの「APPC」サブメニューから、「New dependent Local LU (新しい従属型ローカル LU)」を選択します (あるいは、ボタン・バーの「New (新規)」をクリックし、次に「New dependent local LU (新しい従属型ローカル LU)」を選択します)。

Communications Server for Linux は「Local LU (ローカル LU)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力します。
4. 「OK」ボタンをクリックして、ローカル LU を定義します。従属型ローカル LU は、「Connectivity (接続)」ペインでこの LU が属するリンク・ステーションの下に表示されます。

CPI 通信のための構成

CPI-C シンボリック宛先名を使用する CPI-C アプリケーションをサポートする場合には、CPI-C サイド情報を定義する必要があります。サイド情報により、シンボリック宛先名と、会話のパートナー TP、パートナー LU、モード、セキュリティーに関する情報が関連付けられます。

CPI-C 用のシンボリック宛先名を決定するには、アプリケーション開発者に相談してください (サード・パーティーのアプリケーションの場合は、その製品の資料を参照してください)。

CPI-C サイド情報を構成する前に、次の情報を収集する必要があります。

- TP により使用されるシンボリック宛先名
- パートナー TP 名
- パートナー LU 名または別名
- モード名

CPI-C サイド情報を構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. 「Services (サービス)」メニューの「APPC」サブメニューから、「CPI-C」を選択します。

Communications Server for Linux は「CPI-C Destination Names (CPI-C 宛先名)」ウィンドウを表示します。

2. 「New (新規)」ボタンをクリックします。

Communications Server for Linux は「CPI-C Destination Configuration (CPI-C 宛先の構成)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力します。
4. 「OK」ボタンをクリックすると、CPI-C サイド情報が定義されます。

LUA の構成

LUA API は、LU タイプ 0 から 3 を使用して、ホスト・コンピューターと通信を行うアプリケーションに使用できます。(LUA API の詳細については、「*Communications Server for Linux プログラマーズ・ガイド*」を参照してください。)

LUA を構成する前に、次の構成を行ってください。

1. 84 ページの『ノードの構成』の説明に従って、ノードを構成します。

LUA の構成

2. 87 ページの『接続の構成』の説明に従って、従属トラフィック用の接続を構成します。(アップストリーム SNA ゲートウェイまたは DLUR を使用している場合は、ホストへの直接リンクではなく、アップストリーム・ノードへのリンクを構成してください。)

LUA を構成するには、次の情報を収集する必要があります。

- LU 名または LU プール名。
- 各 LU の LU 番号。LU 番号は、ホストで構成されている LU 番号と一致していません。

LUA を構成するため、次の手順に従って LU を定義します。

1. 「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインで、ホストへのリンク・ステーションを選択します。
2. 「New (新規)」ボタンをクリックします。
3. 表示されるダイアログから、「New LU for LUA (LUA の新しい LU)」を選択します。
4. ダイアログのフィールドに適切な値を入力します。LU タイプとして「Unrestricted (制限なし)」を選択します。
5. 「OK」ボタンをクリックします。LU は、「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインに表示されている、ホスト接続リンク・ステーションの下に表示されます。
6. LU プールを使用する場合は、92 ページの『LU プールの定義』の説明に従って LU プールを定義します。

LU プールの定義

LU プールを定義することにより、ユーザー構成が単純化され、ホスト・セッションの確立を柔軟に行うことができます。たとえば、複数の LU を 1 つの LU プールに定義しておくと、この LU プールを使用して複数の LUA アプリケーションを構成できます。これにより、アプリケーションの構成が容易になり、またアプリケーションでプール内の LU をすべて使用できるようになります。

注: ユーザーのセッションは、特定の LU または LU プールのいずれかに割り当てることができます。

- ユーザーのセッションをプール内の特定の LU に割り当てると、セッションはこの LU が使用可能であればこれを使用します。それ以外の場合、セッションはそのプール内のフリーのいずれかの LU を、特定の LU の代わりにそれが LU プールに割り当てられたかのように使用します。
- 指定された LU だけをユーザーが使用するようにして、その LU が使用中であればユーザーのセッションが確立できないようにしたい場合、その LU がプール内に入っていないようにしてください。

ローカル Communications Server for Linux ノードの LU プールは、「LU プール (LU Pools)」ウィンドウで確認できます。このウィンドウには、ローカル・システム上に構成されている LU プールがリストされます。また、LU プールに追加する LU を選択することもできます。

3270 が使用するプールに追加できる LU タイプを次に示します (ただし、1 つのプールには異なる種類の LU を混在させないでください)。

- 3270 display LU (3270 ディスプレイ LU)
- Unrestricted LU (制限なしの LU)

LU は、プールに追加する前に必ずローカル・ノードで定義してください。

LU プールを構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. 「**Windows (ウィンドウ)**」メニューから、「**LU Pools (LU プール)**」を選択します。

Communications Server for Linux は「LU Pools (LU プール)」ウィンドウを表示します。

2. 「**New (新規)**」ボタンをクリックします。

Communications Server for Linux は「LU Pool Configuration (LU プール構成)」ダイアログを表示します。

右側のボックスに、まだプールに割り振られていない LU のリストが表示されます。「無制限 LU」タイプを持つこれらの LU は、LUA の新規プールに組み込むことができます。

3. プールに追加する LU (1 つ以上) を選択してから、「**New (新規)**」ボタンをクリックします。選択された LU が左側のボックスへ移動します。

左側のボックスから LU を除去するには、LU を選択して「**Remove (除去)**」ボタンをクリックします。

4. 「**OK**」をクリックすると、LU プールが定義されます。

左側のボックスに表示されている LU がすべて LU プールに追加されます。

定義されたプールが「LU Pools (LU プール)」ウィンドウに表示されます。

SNA ゲートウェイの構成

Communications Server for Linux には、ホスト・コンピューターへの直接アクセス機能のみでなく、SNA ゲートウェイ機能が組み込まれています。この機能により、他のコンピューターが、Communications Server for Linux ノードを介してホスト・コンピューターに接続できます。各コンピューターからホストへ個別に接続する必要はありません。

104 ページの図 8 に SNA ゲートウェイの機能を示します。

SNA ゲートウェイの構成

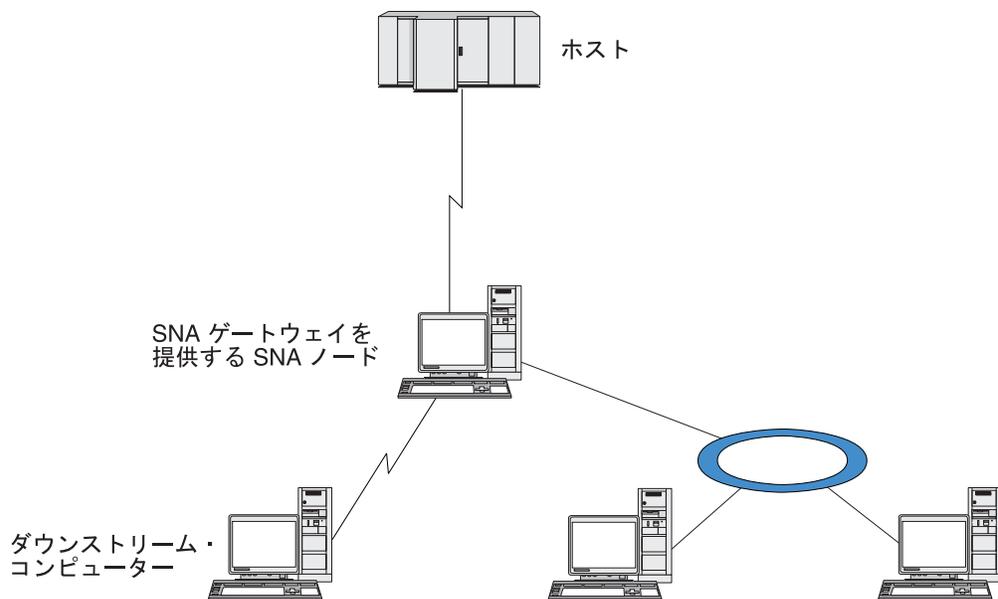


図8. SNA ゲートウェイ

ダウストリーム・コンピューターには、従属 LU をサポートする、タイプ 2.0 または 2.1 の SNA PU がインストールされていなければなりません。たとえば、別の Communications Server for Linux コンピューターや、Communications Server for Windows NT が稼働する PC を、ダウストリーム・コンピューターとして使用できます。

ローカル Communications Server for Linux ノードで SNA ゲートウェイ機能を使用すると、ホストとダウストリーム・コンピューター間で送受信されるデータはすべて、ローカル・ノードを介して転送されます。これにより、ダウストリーム・コンピューターは、直接リンクがなくても、Communications Server for Linux やその他のダウストリーム・コンピューターとホスト接続を共有できます。たとえば、複数のダウストリーム・コンピューターを、ローカル・トークンリング・ネットワークを介して Communications Server for Linux に接続するように設定すると、これらのコンピューターはすべて、同一の長距離専用回線を介して、Communications Server for Linux からホストにアクセスできるようになります。

SNA ゲートウェイを使用すると、ダウストリーム・コンピューターと、それらのコンピューターとの通信リンクを定義する必要がなくなるため、ホスト側での構成が単純化されます。ホストの構成では、Communications Server for Linux コンピューターとそのホスト通信リンクのみを含めます。ダウストリーム・コンピューターの LU は、Communications Server for Linux コンピューターのリソースの一部として構成されます。ホスト・コンピューターでは、SNA ゲートウェイの使用を認識しません。

SNA ゲートウェイを構成する前に次の構成作業を行ってください。

- 84 ページの『ノードの構成』の説明に従って、ローカル・ノードを定義します。
- 87 ページの『接続の構成』の説明に従って、ローカル・ノードとホスト間の、従属トラフィック用のポートとリンク・ステーションを構成します。また、ローカル・ノードとダウストリーム・ノード間の、従属トラフィック用のポートとリンク・ステーションを構成します。事前に定義されていないダウストリーム

LU をサポートする必要がある場合は、暗黙の PU とダウンストリーム LU をサポートするように、ポートのテンプレートを定義できます (『暗黙のダウンストリーム LU のサポート』を参照してください)。

- ホストとの通信に使用する、ローカル・ノードの LU (アップストリーム LU) を定義します。アップストリーム LU は、LU タイプが Unrestricted (制限なし) (unknown (不明)) の、LU タイプ 0 から 3 として定義してください。(ダウンストリーム・ノードの LU にはどの LU タイプでも定義できます。)
- LU プールを使用する場合は、102 ページの『LU プールの定義』の説明に従って LU プールを定義します。

暗黙のダウンストリーム LU のサポート

Communications Server for Linux に対して事前に定義されていないダウンストリーム LU をサポートするには、暗黙のダウンストリーム PU と LU に対して、ポートのテンプレートを定義できます (基本的なポート構成については、87 ページの『接続の構成』を参照してください)。これらのテンプレートにより、ダウンストリーム LU がサポートされます。この際、ダウンストリーム・ノードのすべての LU をサポートするように、ローカル・ノードに LU を 1 つ構成する必要はありません。

SNA ゲートウェイに対してダウンストリーム LU を構成する前に、次の情報を収集する必要があります。

- ダウンストリーム LU をサポートする LU 番号の範囲
- ホスト LU 名

暗黙のダウンストリーム LU についてテンプレートを定義する手順は、次のとおりです。

1. ポートをすでに構成している場合は、「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインの、ポート定義をダブルクリックします。Communications Server for Linux は「Port configuration (ポート構成)」ダイアログを表示します。

ポートをまだ構成していない場合は、ポートを構成します。

- a. 「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインを選択します。
- b. 「New (新規)」ボタンをクリックします。
- c. 表示されたダイアログから、ポートを定義してリンク・プロトコルのタイプを選択します。

Communications Server for Linux は「Port configuration (ポート構成)」ダイアログを表示します。

- d. 基本ポート・パラメーターを入力します。詳細については、87 ページの『接続の構成』を参照してください。
2. ダイアログの下部にある、「Advanced (詳細設定)」ボタンをクリックします。

Communications Server for Linux は「Port Parameters (ポート・パラメーター)」ダイアログを表示します。下のペインに表示されている設定が、ダウンストリーム LU テンプレートに反映されます。

SNA ゲートウェイの構成

3. 「Configure downstream LUs for implicit PU access (暗黙 PU アクセス用にダウンストリーム LU を構成)」オプションを選択します。
4. 「OK」をクリックします。

Communications Server for Linux は「Downstream LU Template Configuration (ダウンストリーム LU テンプレートの構成)」ダイアログを表示します。

5. ダイアログのフィールドに適切な値を入力します。
6. 「OK」をクリックすると、暗黙のダウンストリーム LU テンプレートが定義されます。

ダウンストリーム LU の定義

SNA ゲートウェイに対してダウンストリーム LU を構成する前に、次の情報を収集する必要があります。

- 各ダウンストリーム LU の LU 名。(これはローカル ID であり、ダウンストリーム・システムの構成と一致する必要はありません。)
- 各ダウンストリーム LU の LU 番号
- ダウンストリーム・ノードへのリンク・ステーション
- アップストリーム LU 名 (ホスト LU の場合)

SNA ゲートウェイに対してダウンストリーム LU を構成する手順は次のとおりです。

1. 「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインで、ダウンストリーム・ノードへのリンク・ステーションを選択します。
2. 「New (新規)」ボタンをクリックします。
3. 「New downstream LU (新しいダウンストリーム LU)」を選択してから、「OK」をクリックします。

Communications Server for Linux は「Downstream LU (ダウンストリーム LU)」ダイアログを表示します。

4. ダイアログのフィールドに適切な値を入力します。
5. 「OK」をクリックすると、ダウンストリーム LU が定義されます。

LU 定義は、「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインで、ダウンストリーム・ノード接続リンク・ステーションの下に表示されます。

DLUR の構成

Communications Server for Linux は、ホスト・コンピューターへの直接アクセス機能のみでなく、従属 LU リクエスト (DLUR) 機能を備えています。この機能により、ホストへ直接接続しなくても、APPN ネットワーク内の複数のノードを介して従属 LU のセッションを実行できます。

従属 LU セッションでは、通常ホスト・コンピューターとの直接通信リンクが必要となります。1 つの APPN ネットワーク内で多数のノード (ホスト・ノードも含む) が互いに接続されている場合、一部のノードには、ホストとの直接接続がなく、他のノードを介した間接接続しかないことがあります。このような間接接続ノードの LU からホストへ、従属 LU セッションを確立することはできません。

従属 LU リクエスター (DLUR) は、このような制限を解決するように設計された APPN 機能です。

APPN ノード (Communications Server for Linux ノードなど) 上の DLUR は、ホストの従属 LU サーバー (DLUS) と連携して動作し、DLUR ノードの従属 LU から、APPN ネットワークを介した DLUS ホストまでのセッションの経路を指定します。ホストへの経路では、複数のノードを経由できるため、APPN のネットワーク管理、ダイナミック・リソース検出、経路計算などの機能を利用できます。DLUR は LU のあるノードで使用可能な状態であり、DLUS はホスト・ノードで使用可能な状態でなければなりません。ただし、セッション経路の中間ノードでは DLUR は必要ありません。

Communications Server for Linux DLUR ノードがネットワーク・ノードまたはブランチ・ネットワーク・ノードの場合は、Communications Server for Linux ノードに接続しているダウンストリーム・コンピューターの従属 LU に、パススルー DLUR 機能を提供することもできます。このような LU では、ネットワークを介してホストにアクセスするために、ノード内部の LU と同様に Communications Server for Linux ノードの DLUR を使用できます。ダウンストリーム・コンピューターは DLUR を実行しないので、DLUR が使用されていることを認識する必要はありません。

図 9 に示すのは、APPN ネットワーク・ノードとして構成されている Communications Server for Linux サーバーで、パススルー DLUR をインプリメントしているため、ホスト (アップストリーム・ノード) の LU と、APPN ネットワーク内のノード (ダウンストリーム・ノード) の LU とのセッションをサポートしています。

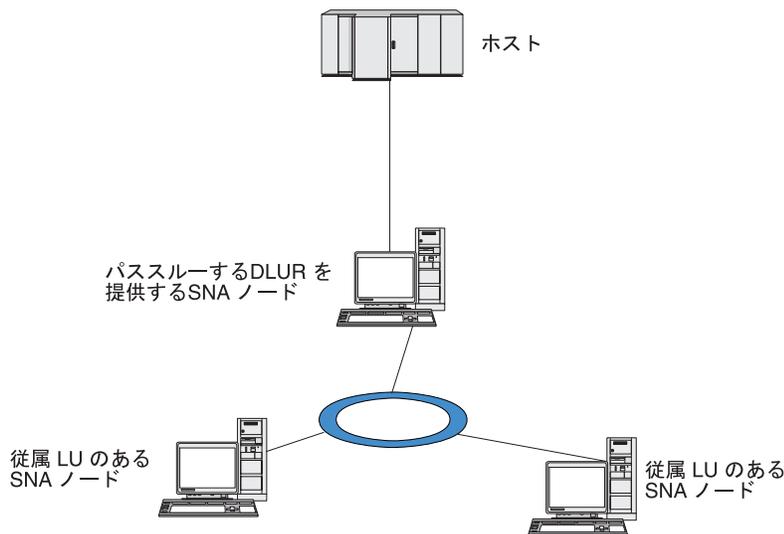


図 9. DLUR を提供する Communications Server for Linux ノード

注:

1. LEN ノードでは DLUR を構成できません。
2. パススルー DLUR は、ネットワーク・ノードまたは分岐ネットワーク・ノード上でしか構成できません。

3. 分岐エクステンダーを使用する場合は、その分岐のエンド・ノード (分岐ネットワーク・ノードを、ネットワーク・ノード・サーバーとして使用する) 上で DLUR を構成することはできません。しかし、分岐ネットワーク・ノード上でパススルー DLUR を構成すれば、このノードから従属 LU アプリケーションをサポートすることができます (つまり、分岐上のエンド・ノードは DLUR は実行しませんが、分岐ネットワーク・ノード上のパススルー DLUR は使用するということです)。

DLUR の構成手順は、従属 LU が、ローカル・ノードとダウンストリーム・ノードのどちらにあるかによって異なります。

ローカル・ノードでの DLUR サポートの構成

この手順を実行するために、次の情報を収集する必要があります。

- ローカル・ノード上の PU の ID。
- PU 名。(これはローカル ID であり、ホスト構成に一致する必要はありません。)
- ホスト上の DLUS の名前 (および、バックアップ DLUS がある場合はその名前)。
- 各ダウンストリーム LU の LU 名、LU 番号、LU タイプ。LU 番号は、ホストで構成されている番号と一致していなければなりません。

ローカル・ノードで DLUR サポートを構成する手順は次のとおりです。

1. 84 ページの『ノードの構成』の説明に従って、ローカル・ノードを定義します。ダウンストリーム・ノードに対してパススルー DLUR をサポートする場合は、ノードを APPN ネットワーク・ノードまたは分岐ネットワーク・ノードとして定義してください。
2. APPN ネットワークとの接続を構成します。APPN 接続では、ローカル・ノードと隣接する APPN ネットワーク・ノード間の、独立トラフィック用のポートとリンク・ステーションが、少なくとも 1 つは必要になります。詳細については、87 ページの『接続の構成』を参照してください。
3. ローカル・ノードの DLUR PU を定義します (DLUR PU により、ホストとの接続がサポートされます)。

DLUR PU を構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

- a. 「**Services (サービス)**」メニューから、「**Connectivity (接続)**」サブメニュー、「**New DLUR PU (新規 DLUR PU)**」の順に選択します (あるいは、ボタン・バーの「**New (新規)**」ボタンをクリックして、「**DLUR PU**」を選択します)。

「**OK**」ボタンをクリックすると、Communications Server for Linux は「**DLUR PU Configuration (DLUR PU 構成)**」ダイアログを表示します。

- b. ダイアログのフィールドに適切な値を入力します。
- c. 「**OK**」ボタンをクリックすると、DLUR PU が定義されます。

「Connectivity (接続)」ペインの DLUR 項目の下に DLUR PU が表示されません。

- ローカル・ノード上の LU をサポートするように DLUR を構成するには、ローカル・ノードに LU を追加する必要があります。LU は、101 ページの『LUA の構成』で説明されているように、LUA をサポートするように構成されなければなりません。それらの LU によってサポートされるユーザー・アプリケーションの要件によっては、さらに他の構成を実行する必要がある場合もあります。

ダウンストリーム・ノードをサポートするパススルー DLUR の構成

この手順を実行するために、次の情報を収集する必要があります。

- 各ダウンストリーム・ノードごと、またはダウンストリーム・ノード上の各 PU ごと、ダウンストリーム PU 名。(これはローカル ID であり、ホスト構成に一致する必要はありません。)
- ホストの DLUS の名前

ダウンストリーム・ノードでパススルー DLUR サポートを構成する手順は次のとおりです。

- ローカル・ノードを、APPN ネットワーク・ノードとして定義します (84 ページの『ノードの構成』を参照してください)。
- ダウンストリーム・ノードとの接続を構成します。87 ページの『接続の構成』の手順に従って、ローカル・ノードと各ダウンストリーム・ノード間の従属トラフィック用の、ポートとリンク・ステーションを構成します。(ダウンストリーム・ノードのパススルー DLUR をサポートするために、DLUR PU を定義する必要はありません。)
- 1 つのダウンストリーム・ノードが複数の PU をサポートできます。この場合、各ダウンストリーム PU には、それぞれ異なるリンクが関連付けられているため、Communications Server for Linux DLUR ノードとダウンストリーム・ノード間に、複数のリンクを構成する必要があります。また、各リンクのダウンストリーム PU 名を確認しておく必要があります。

TN サーバーの構成

SNA ネットワークではなく TCP/IP を使用して通信を行う 3270 エミュレーション・プログラムは、「TN3270 プログラム」(Telnet 3270 エミュレーション・プログラム) と呼ばれます。

TN3270 プログラムには、TN3270E (Telnet 3270 標準拡張) サポートを組み込むこともできます。TN3270E は、Telnet を使用して、3270 装置エミュレーション (端末およびプリンターの両方を含む) をサポートするオープン・プロトコルです。これにより、Telnet クライアントから、(LU 名を指定することにより) 特定の装置を選択できるようになります。また、ATTN キーおよび SYSREQ キー、SNA 応答処理などの各種機能も拡張サポートされます。

注: 本書では、TN3270、TN3287、TN3270E プロトコルのすべてに適用できる情報については、TN3270 という用語を使用します。

Communications Server for Linux TN サーバーを使用すると、3270 ホスト・コンピューター以外のコンピューターの TN3270 ユーザーが、この 3270 ホスト・コンピューターへアクセスできるようになります。TN サーバーにより、直接リンクがな

TN サーバーの構成

くても、TN3270 ユーザーは Communications Server for Linux ユーザーまたはその他の TN3270 ユーザーとホストへの接続を共有できます。また、TN3270 ユーザーは、TCP/IP を実行していないホストへもアクセスできます。

図 10 に、TN3270 クライアント向けの TN サーバー・サポートを備えた Communications Server for Linux ノードを示します。TN サーバー・ノードとクライアントは、TCP/IP ネットワークを介して通信を行います。

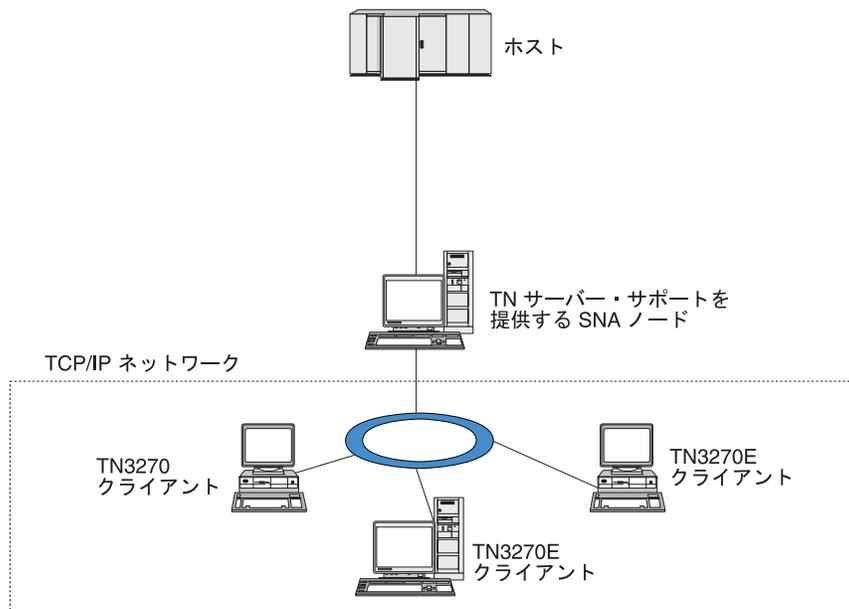


図 10. TN サーバー用に構成された Communications Server for Linux ノード

Communications Server for Linux TN サーバー機能は、TN3270 ユーザーと Communications Server for Linux 3270 LU との関連付けを行います。TN3270 ユーザーからのデータはすべて、関連付けられた LU へ転送されます。つまり、ホストと TN3270 ユーザーのどちらの構成でも、両者が直接接続しているように構成されています。したがってホストもユーザーも、データが TN サーバーを介して転送されることを認識する必要はありません。

Communications Server for Linux TN サーバーは、IETF RFC 1123、1576、1646、1647、2355 で定義されているプロトコルを正しくインプリメントしている、TN3270 クライアント・エミュレーション・プログラムをすべてサポートします。

TN3270 プログラムが TN サーバーと通信を行うと、Communications Server for Linux では、このプログラムが実行されているコンピューターの TCP/IP アドレスにより、プログラムを識別します。Communications Server for Linux では、同じコンピューター上で異なるユーザーに使用されている、2 つの TN3270 プログラムを区別することはできません。Communications Server for Linux のマニュアルでは、TN サーバー・ユーザーという用語は、TN3270 プログラムを使用している個人ユーザーではなく、TN3270 プログラムが実行されているコンピューターを指します。

111 ページの図 11 に示すように、ホスト側から見た場合と、TN サーバー・ユーザー側から見た場合とでは、TN サーバー構成が異なります。

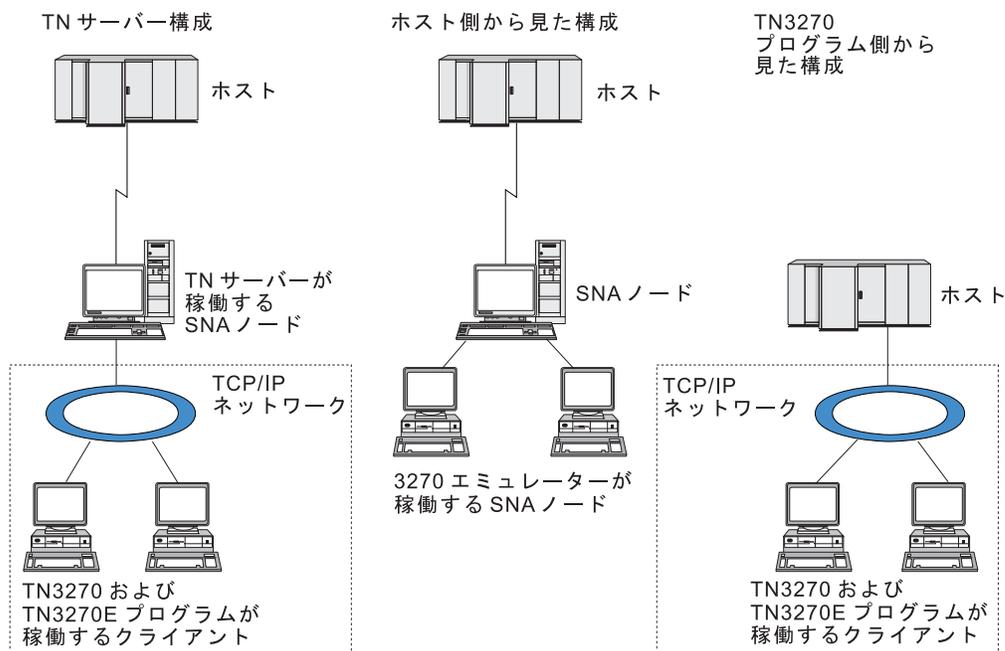


図 11. TN サーバー

通常、TN3270 サーバー機能を使用して Communications Server for Linux に接続する TN サーバー・ユーザーは、それぞれ 1 つの 3270 LU にアクセスするように構成されているため、1 回に確立できるホスト・セッションは 1 つに制限されます。しかし、TN サーバー・ユーザーが、それぞれ専用の 3270 LU を 1 つずつ持つように構成するのではなく、3270 LU プールにアクセスするよう構成することもできます。このように設定すると、ユーザーは、プール内の使用可能な LU と同数のセッションにアクセスできます。

TN サーバーのアクセスを構成する前に、次の構成作業を行ってください。

- 84 ページの『ノードの構成』の説明に従って、ローカル・ノードを定義します。
- 87 ページの『接続の構成』の説明に従って、ローカル・ノードとホスト間の、従属トラフィック用のポートとリンク・ステーションを構成します。

TN サーバーのアクセスを構成するには、次の構成作業を行う必要があります。

- ホストとの通信に使用するローカル・ノードの 3270 LU を定義します。LU の追加については、『3270 LU の定義』を参照してください。
- LU プールを使用する場合は、112 ページの『LU プールの定義』の説明に従って LU プールを定義します。

3270 LU の定義

3270 LU を構成する前に、次の情報を収集してください。

- LU 名。(これはローカル ID であり、ホスト構成に一致する必要はありません。)
- LU 番号 (ある範囲の複数の LU の場合は複数の番号)
- LU タイプ (3270 ディスプレイ・モデルまたは 3270 プリンター)
- プール名 (LU をプールに追加する場合)

前に定義済みのリンク・ステーションについて、タイプ 0 から 3 の LU を構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. このウィンドウの「Connectivity (接続)」ペインから、ホスト接続リンク・ステーションを選択します。
2. 「New (新規)」ボタンをクリックします。
3. 上記の操作の結果表示されたダイアログで、LU タイプ (「New 3270 display LU (新規 3270 ディスプレイ LU)」、または「New 3270 printer LU (新規 3270 プリンター LU)」) を選択します。

この項目を選択して「OK」をクリックすると、Communications Server for Linux は「LU タイプ 0 - 3 (LU Type 0-3)」ダイアログを表示します。

4. ダイアログのフィールドに適切な値を入力します。
5. 「OK」をクリックすると、LU が定義されます。

LU は、「Node (ノード)」ウィンドウの「Connectivity (接続)」ペインに表示されている、ホスト接続リンク・ステーションの下に表示されます。

LU プールの定義

3270 に対して、LU プールを定義すると、ユーザー構成が単純化され、ホスト・セッションの確立の柔軟性が大幅に向上します。たとえば、いくつかの 3270 LU を 1 つの LU プール内に定義し、次に、この LU プールを使用する複数の TN3270 クライアントを構成することができます。このようにすると、3270 セッションを容易に構成でき、任意のクライアントがプール内の任意の LU を使用できます。

注: TN3270 クライアントは、特定の LU または LU プールのいずれかに割り当てることができます。

- クライアントをプール内の特定の LU に割り当てると、クライアントはこの LU が使用可能であればこれを使用します。それ以外の場合、クライアントはそのプール内のフリーのいずれかの LU を、特定の LU の代わりにそれを LU プールに割り当てられたかのように使用します。
- 指定された LU だけをクライアントが使用するようにして、その LU が使用中であればクライアントのセッションが確立できないようにしたい場合、その LU がプール内にないことを確認してください。

ローカル Communications Server for Linux ノードの LU プールは、「LU プール (LU Pools)」ウィンドウで確認できます。このウィンドウには、ローカル・システム上に構成されている LU プールがリストされます。また、LU プールに追加する LU を選択することもできます。

3270 が使用するプールに追加できる LU タイプを次に示します (ただし、1 つのプールには異なる種類の LU を混在させないでください)。

- 3270 display LU (3270 ディスプレイ LU)
- Unrestricted LU (制限なしの LU)

LU は、プールに追加する前に必ずローカル・ノードで定義してください。

LU プールを構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. 「Windows (ウィンドウ)」メニューから、「LU Pools (LU プール)」を選択します。

Communications Server for Linux は「LU Pools (LU プール)」ウィンドウを表示します。

2. 「New (新規)」ボタンをクリックします。

Communications Server for Linux は「LU Pool Configuration (LU プール構成)」ダイアログを表示します。

右側のボックスに、まだプールに割り振られていない LU のリストが表示されます。このリストに表示されている LU はすべて、新しいプールに割り振ることができます。

3. プールに追加する LU (1 つ以上) を選択してから、「New (新規)」ボタンをクリックします。選択された LU が左側のボックスへ移動します。

左側のボックスから LU を除去するには、LU を選択して「Remove (除去)」ボタンをクリックします。

4. 「OK」をクリックすると、LU プールが定義されます。

左側のボックスに表示されている LU がすべて LU プールに追加されます。

定義されたプールが「LU Pools (LU プール)」ウィンドウに表示されます。

TN3270 サーバーの構成

TN3270 サーバーを構成する前に、次の情報を収集する必要があります。

- サーバーが TN3270 のみをサポートしているのか、TN3270E (TN3270 サポートも含まれます) もサポートしているかどうか。
- TN3270E クライアントが特定の LU を要求できるかどうか。
- 各クライアントのディスプレイとプリンターの LU 名 (あるいは LU プール名)。(プリンター LU 名が必要となるのは、TN3270E をサポートしている場合のみです。)
- 特定のクライアントのみが許可されている場合や、特定の LU に対してクライアントを制限する場合には、対象となるクライアントの TCP/IP 名またはアドレスが必要です。
- TN サーバー・ノードの TCP/IP ポート番号。
- SSL データの暗号化、クライアント認証、およびサーバー認証が必要かどうか (このオプションは、このサポートに必要な追加のソフトウェアがインストールされている場合にのみ使用可能です)。

ディスプレイ LU とプリンター LU を関連付ける場合は、これらの LU の名前も必要です。TN サーバーのアソシエーション・レコードにより、プリンター LU とディスプレイ LU の関連付けが定義されるため、この 2 つの LU は TN3270E プロトコルによって接続できます。TN3270E をサポートしない場合、または、プリンター LU をサポートしない場合は、アソシエーション・レコードを定義する必要はありません。

TN サーバーの構成

TN サーバーのデフォルト・レコードで定義されるパラメーターは、すべての TN3270 クライアント・セッションで使用されます。各サーバーにデフォルト・レコードを 1 つずつ定義できます。

TN3270 サーバーを構成するには、「Node (ノード)」ウィンドウで次の手順を行います。

1. TN サーバーのアクセス・レコードを定義します。
 - a. 「**Services (サービス)**」メニューから、「**TN Server (TN サーバー)**」を選択します。

Communications Server for Linux は、「TN サーバー (TN Server)」ウィンドウを表示します。このウィンドウの上部のペインには、TN サーバーの構成済みアクセス・レコードが表示され、下部のペインには、TN サーバーのアソシエーション・レコードが表示されます。

- b. TN3270 サーバーのアクセス・レコードが表示されているペインを選択してから、「**New (新規)**」ボタンをクリックします。

Communications Server for Linux は「TN Server Access (TN サーバー・アクセス)」ダイアログを表示します。

- c. ダイアログのフィールドに適切な値を入力します。
 - d. 「**OK**」をクリックすると、TN サーバー・アクセス・レコードが定義されます。定義されたレコードが「TN Server (TN サーバー)」ウィンドウに表示されます。

2. TN サーバーのアソシエーション・レコードを定義します。

- a. 「TN Server (TN サーバー)」ウィンドウで、アソシエーション・レコードが表示されているペインを選択してから、「**New (新規)**」ボタンをクリックします。

Communications Server for Linux は「TN Server Association Record (TN サーバー・アソシエーション・レコード)」ダイアログを表示します。

- b. ダイアログのフィールドに適切な値を入力します。
 - c. 「**OK**」をクリックすると、TN サーバーのアソシエーション・レコードが定義されます。定義されたレコードが「TN Server (TN サーバー)」ウィンドウに表示されます。

3. プリンター応答を強制する必要がある場合は、TN3270 のすべてのセッションにキープアライブ・メソッドを指定するか、TN3270 クライアントの許可検査に使用される取り消しリストを保有する、外部 LDAP サーバーへのアクセス方法を指定するか、あるいは、TN3270 SLP (サービス・ロケーション・プロトコル)、または TN サーバー拡張パラメーター・ダイアログを使用してください。

TN サーバーに対する SSL サポートの構成についての詳細は、IBM Communications Server Support Web ページ (<http://www.ibm.com/software/network/commsserver/support/>) を参照してください。

TN リダイレクターの構成

Communications Server for Linux TN リダイレクター機能により、TN3270、TN3270E、TN5250、および VT クライアント (まとめて Telnet クライアントと呼ばれます) への、パススルー TCP/IP ホスト・アクセスが可能になります。Telnet ユーザーは、TCP/IP 接続を介して Communications Server for Linux との通信を行います。Communications Server for Linux は、次に別の TCP/IP 接続を介してホストとの通信を行います。これにより、ユーザーは、ユーザーからホストへの完全な接続を行わなくても、必要に応じて、Secure Sockets Layer (SSL) のセキュリティチェックを使用することができます。たとえば、次のとおりです。

- クライアントが、チェックを必要としない TCP/IP LAN を経由して Communications Server for Linux に接続しており、しかも SSL が必要なリモート・ホストに接続している場合、Communications Server for Linux とホストの間の TCP/IP を介して SSL を使用できます。つまり、すべてのクライアントに対するセキュリティを一度検査すれば、個々のクライアントがセキュリティ情報を提供する必要はないということです。
- Communications Server for Linux がホストと同じサイトにインストールされていても、クライアントが外部サイトから接続している場合、ホストに SSL ソフトウェアをインストールしなくても、Communications Server for Linux にクライアント接続すれば SSL を使用できます。

TN リダイレクターの構成

TN リダイレクター・アクセスの構成を行う前には、84 ページの『ノードの構成』の説明に従って、ローカル・ノードを定義しなければなりません。さらに、次の情報を収集する必要もあります。

- 特定のクライアントのみが許可されている場合は、そのクライアントの TCP/IP 名またはアドレスが必要です。
- クライアントが TN リダイレクター・ノードに接続する際に使用する、TCP/IP ポート番号。
- ホストの TCP/IP 名またはアドレス。
- TN リダイレクター・ノードがホストに接続する際に使用する、TCP/IP ポート番号。
- クライアントと TN リダイレクター・ノードの間で、SSL データの暗号化、クライアント認証、およびサーバー認証が必要かどうか (このオプションは、そのサポートに必要な追加のソフトウェアがインストールされている場合にのみ使用可能です)。
- TN リダイレクター・ノードとホストの間で、SSL データの暗号化が必要かどうか。

TN リダイレクターのデフォルト・レコードで定義されるパラメーターは、すべての TN リダイレクター・クライアント・セッションで使用されます。個々のクライアント TCP/IP ポート番号には、デフォルト・レコードを 1 つずつ定義することができます。

TN リダイレクターを構成するには、「Node (ノード)」ウィンドウで次の手順を実行して、TN リダイレクター・アクセス・レコードを定義します。

TN リダイレクターの構成

1. 「**Services (サービス)**」メニューから、「**TN Server (TN サーバー)**」を選択します。

Communications Server for Linux は、「TN サーバー (TN Server)」ウィンドウを表示します。このウィンドウには、TN3270 サーバーの構成済みアクセス・レコード、TN3270 サーバーのアソシエーション・レコード、TN リダイレクターのアクセス・レコードが表示されます。

2. TN リダイレクターのアクセス・レコードが表示されているペインを選択してから、「**New (新規)**」ボタンをクリックします。

Communications Server for Linux は「TN Redirector Access (TN リダイレクター・アクセス)」ダイアログを表示します。

3. ダイアログのフィールドに適切な値を入力します。
4. 「**OK**」をクリックすると、TN リダイレクターのアクセス・レコードが定義されます。定義されたレコードは、「TN Server (TN サーバー)」ウィンドウの「TN Redirector (TN リダイレクター)」ペインに表示されます。

注: TN リダイレクターを使用するには、SNA ノードがアクティブでなければなりません (TN リダイレクターがそのノードの SNA リソースをどれも使用しない場合でも)。

Communications Server for Linux の使用不可化

Communications Server for Linux ソフトウェアを使用不可にすると、自動的に Communications Server for Linux ノードと関連する接続コンポーネントも停止します。Communications Server for Linux を使用不可にすると、他のどのプロセスも (LUA アプリケーションなど)、このサーバー上の Communications Server for Linux リソースを使用できなくなります。

通常、サービスを終了するユーザーとして、サービスを個別に停止して Communications Server for Linux 上で何も実行されていない状態にしてから、システムを使用不可にしてください。

ユーザーが活動状態であるときに Communications Server for Linux を使用不可にする必要がある場合は、そのソフトウェアを使用不可にする前に、これらのユーザーに Communications Server for Linux が停止することを通知し、各ユーザーがアクティビティを終了するための時間を与えてください。

Communications Server for Linux ソフトウェアを使用不可にすると、APPC、CSV、LUA、NOF、または MS API を使用しているアプリケーションは、戻りコード COMM_SUBSYSTEM_ABENDED によって通知され、CPI-C アプリケーションは、戻りコード CM_PRODUCT_SPECIFIC_ERROR によって通知されます。

Communications Server for Linux ソフトウェアを使用不可にするには、Linux コマンド・プロンプトで以下のコマンドを入力します。

sna stop

Communications Server for Linux が正常に使用不可になると、**sna stop** により終了コード 0 が戻されます。これ以外の終了コードの場合は、エラーが発生して、

Communications Server for Linux ソフトウェアを使用不可にできなかったことを示します。終了コード値についての詳細は、「*Communications Server for Linux 診断用ガイド*」を参照してください。

Communications Server for Linux の自動開始

Communications Server for Linux は、次のようないくつかのステップによって、完全な稼働状況になります。

- Communications Server for Linux ソフトウェアを使用可能にする
- SNA ノードを初期化する
- ノード上に構成されたポートとリンク・ステーションを活動化する

これらの各ステップは独立に管理でき、すべてのステップは、必要な場合、ブート時に実行できます。このセクションの残りの部分では、これらのステップについて説明し、ブート時にそれらを制御する方法について説明します。

Communications Server for Linux の使用可能化

Communications Server for Linux を作動するには、いくつかのカーネル・モジュールをロードする必要があります。これらのモジュールは、Communications Server for Linux ソフトウェアを最初に使用可能にしたときにロードされ、カーネル実行レベルを変更したときにのみアンロードされます。

Communications Server for Linux では、多くのデーモン (プログラム) も実行しなければなりません。これらのデーモンが実行されるまで、Communications Server for Linux を構成または使用することはできません。

- デーモンを開始し、Communications Server for Linux ソフトウェアを使用可能にするには、**sna start** コマンドを実行します。
- デーモンを停止し、Communications Server for Linux ソフトウェアを使用不可にするには、**sna stop** コマンドを実行します。

SNA ノードの初期化

Communications Server for Linux ソフトウェアを使用可能にすると、通常、Motif 管理プログラム **xsnaadmin** を使用して、SNA ノードとそのリソースを構成することができます。ただし、ノードは、初期化しないと使用できません。ノードは、**snaadmin init_node** コマンドを使用してコマンド行管理プログラムから初期化することもできるし、Motif 管理プログラム **xsnaadmin** から初期化することもできます。

ポートおよびリンク・ステーションの活動化

ポートとリンク・ステーションは、さまざまな環境で開始するように構成できます。

- オペレーター介入によってのみ
- 要求時 (リンクのリソースを使用するアプリケーションを開始したとき)
- ノード起動時 (**snaadmin init_node** を実行するか、またはノードを Motif 管理プログラムから開始したとき)

Communications Server for Linux の自動開始

オペレーターによる開始がデフォルトですが、特定のポートまたはリンクの場合に、コマンド行管理プログラムまたは Motif 管理プログラムを使用して、デフォルト設定を変更することができます。

注: あるポートを開始すると、そのポートは、他のコンピューターからの着呼を受け取ることができますが、発呼はできません。リンク・ステーションを開始すると、Communications Server for Linux はリモート・コンピューターと連絡を取ろうとします。

リブート時の Communications Server for Linux の開始

他の Linux サービスと同様に、Communications Server for Linux はリブート時に使用可能になります。つまり、デフォルトではリブート後に **sna start** コマンドが実行されますが、SNA ノードは開始されません。

この初期化は、Communications Server for Linux のブート時初期化スクリプト **/etc/rc.d/init.d/snastart** で行われます。従来の開始スクリプトの場合と同様に、これは、さまざまなブート・レベルの **/etc/rc?.d/init.d/snastart** にリンクされます。

/etc/rc.d/init.d/snastart を編集して、リブート時に発生する事象を変更することができます。最も一般的な変更は、ノードの初期化の追加です。このためのコマンド **snaadmin init_node** は、すでにファイルに組み込まれていますが、コメント化されているため、単にそれをアンコメントするだけで済みます。このコマンドを組み込むことで、ノード開始時に活動化するように構成された任意のポートまたはリンク・ステーションの活動化を起動することもできます。

Communications Server for Linux を使用するアプリケーションは、ノードの初期化が完了するまで開始してはなりません。必要な場合は、これらのアプリケーションをブート時に自動的に開始し、ブート時に実行する必要がある他のすべての **snaadmin** コマンドを、次の 2 つの方法のうちのいずれかによって実行する必要があります。

- **snaadmin init_node** コマンドの後、**/etc/rc.d/init.d/snastart** の末尾にコマンドを追加します。
- 95 より大きな番号を持つ **/etc/rc?.d/init.d** スクリプトを作成します。これにより、Communications Server for Linux の開始後にそれが実行され、コマンドがこのスクリプトに追加されます。

注: **/etc/rc.d/init.d/snastart** ファイルに対して行った変更は、Communications Server for Linux をより新しいバージョンにアップグレードしたときに保管されません。変更結果のコピーを保管していることを常に確認し、それらをアップグレード版に再適用できるようにしてください。

第 9 章 Communications Server for Linux と SNA の情報リソース

この章では、SNA テクノロジーに関する情報を提供する SNA ライブラリー内のリソースと、IBM が提供するさまざまなネットワーキング製品およびサービスについて説明します。また、ネットワーク・フォーラムで入手できる情報についても説明します。

SNA ライブラリー

SNA ライブラリーには、マーケティング・パンフレット、資料、ユーザー・ガイド、チュートリアルが含まれ、以下のトピックに関する入門情報および詳細情報を提供します。

- SNA 理論
- SNA 製品
- 製品のインプリメンテーション
- システムおよびネットワークの構成
- SNA アプリケーション・プログラムおよび API
- 全体の計画、パフォーマンスおよびチューニング
- 問題診断
- ネットワーク管理
- ネットワーク・セキュリティー

IBM 関連資料は、IBM 担当員または地区担当の IBM 営業所を通じて注文できます。

それ以外の資料に関する情報は、IBM 担当員にお尋ねください。

ネットワークで入手可能な情報

情報交換を促進するため、IBM は電子フォーラムと電子掲示板を提供しています。また、インターネット上にホーム・ページを開設し、WWW でアクセスできるオンライン資料を公開しています。

IBMLink™ を使用した製品サポート

IBMLink フォーラムは、IBM が所有するネットワーク上に開設されています。このフォーラムは、IBM 製品のライセンスをお持ちのお客様に対して、技術的問題またはお客様のシステムやネットワークに関するその他の問題の解決を支援できるよう設計されています。IBM の担当者が質問にお答えし、IBM 製品のユーザー間のオンライン・ディスカッションを仲介します。

IBMLink についての詳細は、<http://www.ibm.link.ibm.com> をご覧ください。

IBM ホーム・ページの情報

インターネット上のさまざまな IBM ホーム・ページでは、フォーラムにアクセスできるようになっています。包括的なヘルプについて、IBM のメイン・ホーム・ページを使用して、インターネット上または WWW 上のインフォメーション・センターを検索できます。メイン・ホーム・ページにアクセスするには、<http://www.ibm.com> (日本語の情報は、<http://www.ibm.co.jp>) を使用します。

<http://www.ibm.com/software/network> にアクセスすると、Communications Server for Linux も含めて、IBM ネットワーク・ソフトウェアに関する情報を入手することができます。Communications Server for Linux に関する情報は、<http://www.ibm.com/software/network/commserver> (日本語の情報は、<http://www.ibm.co.jp/software/secureway/cms/cmsAIX.html/>) にあります。

Communications Server for Linux のサポートに関する詳細情報は、<http://www.ibm.com/software/network/commserver/support> (英語サイト) をご覧ください。

ダウンロードに関する情報

ユーザーはワールド・ワイド・ウェブ (WWW) の <http://www.redbooks.ibm.com> から、レッドブック資料をダウンロードすることができます。

IBM ソフトウェアについての情報は、<http://www.ibm.com/software> (英語サイト) でアクセスできます。ここからは、Communications Server for Linux およびすべての IBM ソフトウェア・サーバーに関するページに、リンクすることができます。

推薦資料

SNA の知識を深めたい場合は、SNA の理論と Communications Server for Linux の実践方法を扱っている以下の資料が参考になります。これらの資料は、SNA を知る第一歩が必要な初心者にも、この主題を深く掘り下げる必要がある専門家にも、役立ちます。

- *Systems Network Architecture: Technical Overview* (英文番号 GC30-3073)
- *IBM Communications Server for Linux 管理ガイド*

特定の項目について関心をお持ちの場合は、IBM 担当員にお問い合わせください。

付録. 特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものであり、本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒106-0032
東京都港区六本木 3-2-31
IBM World Trade Asia Corporation
Licensing

以下の保証は、国または地域の法律に沿わない場合は、適用されません。IBM およびその直接または間接の子会社は、本書を特定物として現存するままの状態を提供し、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

P.O. Box 12195
3039 Cornwallis Road
Research Triangle Park, NC 27709-2195
U.S.A.

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができませんが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのもと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っておりません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者にお願いします。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。お客様は、IBM のアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。

それぞれの複製物、サンプル・プログラムのいかなる部分、またはすべての派生した創作物にも、次のように、著作権表示を入れていただく必要があります。「® (お

お客様の会社名 (西暦年)」このコードの一部は、IBM Corp. のサンプル・プログラムから取られています。® Copyright IBM Corp. 2000, 2005, 2006. All rights reserved.

商標

以下は、IBM Corporation の商標です。

Advanced Peer-to-Peer Networking	Power5
AIX	pSeries
Application System/400	S/390
AS/400	SP
CICS	System/370
IBM	System/390
MQSeries	VTAM
MVS	WebSphere
MVS/ESA	z/OS
MVS/XA	z9
NetView	zSeries
OpenPower	
OS/2	

Java およびすべての Java 関連の商標およびロゴは、Sun Microsystems, Inc. の米国およびその他の国における商標または登録商標です。

Microsoft、Windows、Windows NT および Windows ロゴは、Microsoft Corporation の米国およびその他の国における商標です。

Intel、Intel (ロゴ)、Intel Inside、Intel Inside (ロゴ)、Intel Centrino、Intel Centrino (ロゴ)、Celeron、Intel Xeon、Intel SpeedStep、Itanium、および Pentium は、Intel Corporation の米国およびその他の国における商標です。

UNIX は、The Open Group の米国およびその他の国における登録商標です。

Linux は、Linus Torvalds の米国およびその他の国における商標です。

他の会社名、製品名およびサービス名等は、それぞれ各社の商標です。

参考文献

下にリストする IBM 資料には、このライブラリーで扱われているトピックについての情報が記載されています。資料は、次のトピック別に大きく分けてあります。

- Communications Server for Linux、バージョン 6.2.2
- システム・ネットワーク体系 (SNA)
- ホスト構成
- z/OS® Communications Server
- 伝送制御プロトコル / インターネット・プロトコル (TCP/IP)
- X.25
- 拡張プログラム間通信機能 (APPC)
- プログラミング
- その他の IBM ネットワーキング・トピック

Communications Server for Linux ライブラリーの資料については、その要旨が説明されています。その他の資料については、タイトル、資料番号を記し、一部の資料については本書で使用している略称タイトルを記しています。

Communications Server for Linux バージョン 6.2.2 資料

Communications Server for Linux ライブラリーは、以下の資料により構成されています。なお、これらの資料のソフトコピー版が CD-ROM で提供されています。CD-ROM のソフトコピーへのアクセスの方法については、「*IBM Communications Server for Linux 入門*」を参照してください。これらのソフトコピー・ブックをシステムにインストールするには、9 から 15 MB のハード・ディスク・スペースが必要になります (このスペースは、どの各国語バージョンをインストールするかによって異なります)。

- *IBM Communications Server for Linux 入門* (GC88-9996-01)

この資料は Communications Server for Linux の概要を示すもので、サポートされているネットワークの特性、インストール、構成、および操作について説明しています。

- *IBM Communications Server for Linux 管理ガイド* (SC88-9999-01)

この資料には、SNA および Communications Server for Linux の概要、および Communications Server for Linux の構成および操作に関する情報が記載されています。

- *IBM Communications Server for Linux 管理コマンド・リファレンス* (SC88-9998-01)

この資料には、SNA および Communications Server for Linux のコマンドに関する情報が記載されています。

- *IBM Communications Server for Linux CPI-C プログラマーズ・ガイド* (SC88-8582-02)

この資料には、熟練した“C”または Java プログラマー向けに、Communications Server for Linux CPI コミュニケーション API を使用した SNA トランザクション・プログラムの作成に関する情報が記載されています。

- *IBM Communications Server for Linux APPC プログラマーズ・ガイド* (SC88-8569-01)

この資料では、拡張プログラム間通信機能 (APPC) を使用するアプリケーション・プログラムを作成するために必要な情報を記載しています。

- *IBM Communications Server for Linux LUA プログラマーズ・ガイド* (SC88-8584-02)

この資料では、従来型 LU アプリケーション・プログラミング・インターフェース (LUA) を使用してアプリケーション・プログラムを作成するために必要な情報を記載しています。

- *IBM Communications Server for Linux CSV プログラマーズ・ガイド* (SC88-8583-02)

この資料では、Common Service Verbs (CSV) アプリケーション・プログラミング・インターフェース (API) を使用してアプリケーション・プログラムを作成するために必要な情報を記載しています。

- *IBM Communications Server for Linux MS Programmer's Guide* (SC31-67770-02)

この資料では、管理サービス (MS) API を使用してアプリケーション・プログラムを作成するために必要な情報を記載しています。

- *IBM Communications Server for Linux NOF プログラマーズ・ガイド* (SC88-8591-02)

この資料では、Node Operator Facility (NOF) API を使用してアプリケーション・プログラムを作成するために必要な情報を記載しています。

- *IBM Communications Server for Linux 診断ガイド* (SC88-8601-01)

この資料では、SNA ネットワークの問題解決について説明しています。

- *IBM Communications Server for Linux APPC Application Suite ユーザーズ・ガイド* (SC88-8500-02)

この資料には、Communications Server for Linux で使用される APPC アプリケーションに関する情報が記載されています。

- *IBM Communications Server for Linux Glossary* (GC31-6780-02)

この資料は、IBM Communications Server for Linux 関連資料で頻繁に使用される用語とその定義を包括的に収録しています。

システム・ネットワーク体系 (SNA) 関連資料

次の資料では、SNA ネットワークについての情報を記載しています。

- *Systems Network Architecture: Format and Protocol Reference Manual—Architecture Logic for LU Type 6.2* (SC30-3269)
- *Systems Network Architecture: Formats* (GA27-3136)

- *Systems Network Architecture: Guide to SNA Publications* (GC30-3438)
- *Systems Network Architecture: Network Product Formats* (LY43-0081)
- *Systems Network Architecture: Technical Overview* (GC30-3073)
- *Systems Network Architecture: APPN Architecture Reference* (SC30-3422)
- *Systems Network Architecture: Sessions between Logical Units* (GC20-1868)
- *Systems Network Architecture: LU 6.2 Reference—Peer Protocols* (SC31-6808)
- *Systems Network Architecture: Transaction Programmer's Reference Manual for LU Type 6.2* (GC30-3084)
- *IBM 3270 情報表示システム データストリーム プログラマー用解説書* (GA23-0059)
- *Networking Blueprint Executive Overview* (GC31-7057)
- *Systems Network Architecture: Management Services Reference* (SC30-3346)

ホスト構成関連資料

次の資料では、ホスト構成についての情報を記載しています。

- *ES/9000, ES/3090 IOCP User's Guide Volume A04* (GC38-0097)
- *3174 Establishment Controller Installation Guide* (GG24-3061)
- *3174 制御装置: 計画の手引き 構成サポート C リリース 5* (GA27-3918)
- *OS/390 ハードウェア構成定義 ユーザーズ・ガイド* (SC28-1848)

z/OS Communications Server 関連資料

次の資料では、z/OS Communications Server についての情報を記載しています。

- *z/OS V1R7 Communications Server: SNA ネットワーク導入の手引き* (SC88-8928)
- *z/OS V1R7 Communications Server: SNA Diagnostics* (Vol 1: GC31-6850, Vol 2: GC31-6851)
- *z/OS V1R6 Communications Server SNA リソース定義解説書* (SC88-8929)

TCP/IP 関連資料

次の資料では、伝送制御プロトコル / インターネット・プロトコル (TCP/IP) ネットワーク・プロトコルについての情報を記載しています。

- *z/OS V1R7 Communications Server: IP 構成ガイド* (SC88-8926)
- *z/OS V1R7 Communications Server: IP 構成解説書* (SC88-8927)
- *z/VM V5R1 TCP/IP 計画およびカスタマイズ* (SC24-6125)

X.25 関連資料

次の資料では、X.25 ネットワーク・プロトコルについての情報を記載しています。

- *Communications Server for OS/2[®] Version 4 X.25 Programming* (SC31-8150)

APPC 関連資料

次の資料では、拡張プログラム間通信機能 (APPC) についての情報を記載しています。

- *APPC Application Suite V1 User's Guide* (SC31-6532)
- *APPC Application Suite V1 Administration* (SC31-6533)
- *APPC Application Suite V1 Programming* (SC31-6534)
- *APPC Application Suite V1 Online Product Library* (SK2T-2680)
- *APPC Application Suite Licensed Program Specifications* (GC31-6535)
- *z/OS V1R2.0 Communications Server: APPC Application Suite User's Guide* (SC31-8809)

プログラミング関連資料

次の資料では、プログラミングについての情報を記載しています。

- *共通プログラミング・インターフェース コミュニケーション・インターフェース CPI-C 解説書* (SC26-4399)
- *Communications Server for OS/2 Warp 日本語版 32 ビット アプリケーション・プログラミングの手引き バージョン 6* (SC88-5585)

その他の IBM ネットワーキング関連資料

次の資料には、Communications Server for Linux に関連するその他のトピックについての情報が記載されています。

- *同期データ・リンク制御 (SDLC) 解説書* (GA27-3093)
- *Local Area Network Concepts and Products: LAN Architecture* (SG24-4753)
- *Local Area Network Concepts and Products: LAN Adapters, Hubs and ATM* (SG24-4754)
- *Local Area Network Concepts and Products: Routers and Gateways* (SG24-4755)
- *Local Area Network Concepts and Products: LAN Operating Systems and Management* (SG24-4756)
- *ネットワーク制御プログラムおよびシステム・サポート・プログラム 資源定義の手引き* (SC30-3349)

索引

日本語, 数字, 英字, 特殊文字の順に配列されています。なお, 濁音と半濁音は清音と同等に扱われています。

[ア行]

圧縮, LU セッション・データ 4
アップストリーム・ノード 107
アプリケーション・プログラミング・インターフェース (API) 4
アラート 16
アンインストール, AIX 上の Remote API Client 59
アンインストール, Communications Server for Linux の 42
アンインストール, Linux for System z での Remote API Client の 54
アンインストール, Linux 上の Remote API Client 48
暗黙のダウストリーム LU の構成 105
イーサネット
 リンクの構成 89
 SAP ダイアログ 89
インストール 35
 既存のパッケージの詳細 31
 実行後の保守作業 38
 準備 30
 AIX 上の Remote API Client 56, 57
 Linux for System z 上の Remote API Client 51
 Linux 上の Remote API Client 45
インストール, AIX 上の Remote API Client 56, 57
インストール, Communications Server for Linux の 35
インストール, Linux for System z への Remote API Client の 51
インストール, Linux 上の Remote API Client 45
インストール後の作業 38
インストール前の作業 30
インストール要件 22
インターフェースの選択 14
ウィンドウ 79
エンタープライズ・エクステンダー
 概要 10
 ポート・ダイアログ 91
 リンクの構成 90
エントリー・ポイント 15
オープン・システム間相互接続 (OSI) 27
オペレーティング・システム要件 22
オンライン
 ディスカッション・グループ 119
 フォーラム 120
 文書 119
 ヘルプ 13
 APPC 120

オンライン資料, Communications Server for Linux の 35

[カ行]

カーネル・メモリー 24
開始, リソースの 81
開始, Communications Server for Linux の
 システム起動時に自動的に 78
開始, Communications Server for Linux を自動的に 117
 開始, リポート時の 118
 活性化, ポートおよびリンク・ステーションの 117
 使用可能化, Communications Server for Linux ソフトウェアの 117
 SNA ノードの初期化 117
開始, Communications Server for Linux をリポート時に 118
拡張構成, メモリーおよびストレージ容量 24
拡張パワー・マネージメント 30
拡張プログラム間通信機能 (APPC) 6
活性化, ポートおよびリンク・ステーションの 117
関数 4
管理プログラム
 コマンド行 14
 Motif 13, 16
 NOF API 14
機能要件 19
共通プログラミング・インターフェース (CPI) 4
クライアント 5
クライアント/サーバー
 構成 83
クライアント/サーバーのインストール 38
クライアント/サーバーのサポート 5
クライアント・ソフトウェア
 アップグレード 72
 アンインストール 73
 カスタマイズ 72
 再インストール 72
ゲートウェイ
 定義 7
 SNA ゲートウェイ 7
計画ワークシート 76
言語環境変数 31, 56
 Linux for System z 上の Remote API Client 50
 Linux 上の Remote API Client 45
コール 4
広域ネットワーク (WAN) 2
構成 20
 暗黙のダウストリーム LU 105
 計画 76
 接続 87
 ノード 84
 バックアップ 39

構成 (続き)

- 表示 82
 - ファイル 39, 41
 - 変更 82
 - ポート 88
 - リモート・ノード 98
 - 例 85, 107, 110
 - APPC 通信 93
 - CPI-C サイド情報 101
 - DLUR 106
 - LEN ノードのパートナー LU 97
 - LU 6.2 96, 100
 - LU タイプ 0 から 3 91
 - SNA ゲートウェイのダウンストリーム LU 103
 - TN サーバーのアソシエーション・レコード 113
 - TN サーバーのデフォルト 114
 - TN リダイレクターのデフォルト 115
- 構成ウィンドウ、Windows 上の Remote API Client
- 拡張パラメーター 66
 - パラメーター 65
- 構成サーバー 83
- 除去 84
 - 追加 84
- 互換性計画 26
- 国際標準化機構 (ISO) 27
- コマンド行管理プログラム 14
- コンポーネントの管理 79

[サ行]

- サーバー 5
 - 除去 84
 - 追加 84
 - Linux ハードウェア要件 22
- サーバー、Telnet 10
- サービス・アクセス・ポイント (SAP) 22, 89
- 作業シート 77
- 削除、リソースの 81
- サブルーチン 4
- システム・アプリケーション体系 (SAA) 5
- 自動開始、Communications Server for Linux の 117
- 始動コマンド 78
- 従属 LU 6.2 100
- 従属 LU サーバー (DLUR) 107
- 従属 LU の動的定義 (DDDLU) 3
- 従属 LU リクエスター (DLUR) 2
- 使用可能化、Communications Server for Linux ソフトウェアの 117
- 使用可能化、Communications Server for Linux の
 - ローカル・システムでの 78
- 使用不可化、Communications Server for Linux の 116
- 情報リソース 119
- 初期化、SNA ノードの 117
- 資料、Communications Server for Linux の 35
- 推薦資料 120
- スキル要件 21

- ストレージ要件 24
- セキュリティー・オプション 15
- セッション
 - サポート 4
 - ペーシング 15
 - ルーティング 14
 - U 型 4
- 接続
 - オプション 2
 - 構成 87
- 接続ネットワーク 14
- 接続ネットワーク、構成 89
- ソフトウェア機能 6
- ソフトウェア要件 22, 23
 - AIX 上の Remote API Client 55
 - Linux for System z 上の Remote API Client 49
 - Linux 上の Remote API Client 44
 - Windows 上の Remote API Client 61

[タ行]

- ダイアログ 79
- 対等
 - ネットワーク 1
- ダウンストリーム LU
 - 構成 106
 - ハードウェアの例 104
 - SNA ゲートウェイ 103
- ダウンストリーム・コンピューター 104
- ダウンストリーム・ノード 107
- 中間セッション・ルーティング (ISR) 14
- ツールバー・ボタン 82
- 追加、リソースの 81
- 通信アダプター 23
- データ・ストリーム 4
- データ・リンク制御オプション 2
- 停止、リソースの 81
- 停止コマンド 116
- ディスカッション・グループ、オンラインの 119
- ディスク・ストレージ 25
- ディレクトリー、Communications Server for Linux 実行可能プログラムの 77
- 適応セッション・レベル・ペーシング 15
- 手順
 - Windows での Remote API Client のインストール 63, 67
- 透過性、ローカル/リモート 4
- 同期データ・リンク制御 (SDLC) 87
- 動的構成 14
- 独立 LU 6.2 の構成 96
- ドメイン 5
- トラブルシューティング・ツール 16
- トランザクション・プログラム (TP) 4
- トランスポート・メディア 21

[ナ行]

- ネットワーク
 - アラート 16
 - 管理 15
 - 計画 19, 20, 26
 - サブエリア 1
 - サポート 1
 - 対等 1
 - 入手可能な情報 119
 - 命名規則 26
- ノード
 - アップストリーム 107
 - 構成 84
 - 初期化 117
 - ダウンストリーム 109
 - ホスト通信 85
 - リモート 97, 98
 - ローカル 108
 - APPN 構成の 86

[ハ行]

- ハードウェア
 - リンク 23
- ハードウェア要件
 - サーバー 22
 - AIX 上の Remote API Client 55
 - Linux for System z 上の Remote API Client 49
 - Linux 上の Remote API Client 43
 - Windows 上の Remote API Client 61
- パートナー LU
 - 構成 97
 - 別名 98
 - リモート・ノード 99
 - ワイルドカードを使用した定義 99
- パートナー・アプリケーション 6
- ハード・ディスク・ストレージ 24
- バス、Communications Server for Linux 実行可能プログラムの 77
- バックアップ
 - 構成ファイル 39
 - 復元 40
- バックアップ、マスター・サーバーの 83
- バックアップ・サーバー 6
- パワー・マネージメント 30
- ピア・サーバー 6
- 表示、構成の 82
- フォーカル・ポイント 15
- フォーラム、オンラインの 120
- 複数サーバー、1 ドメイン内の 6
- 物理装置 (PU) 7
- 分岐エクステンダー 8
- 分岐ネットワーク・ノード 8
- 分散処理
 - アプリケーションのサポート 6

- 分散処理 (続き)
 - 環境 1
- 文書、オンラインの 119
- ペーシング、セッション・レベル 15
- ペーシング・ストレージ 25
- 別名、パートナー LU の定義 98
- ヘルプ
 - Motif 管理プログラム 13
- 変更、構成の 82
- ポート
 - 活動化 117
- ポートの構成 88, 89, 90
- ホスト
 - サブエリア・ネットワーク内で 1
 - APPN ネットワーク内での 2
 - LU サポート 3
- ボタン、リソース・ウィンドウの 82

[マ行]

- マイグレーション、旧レベルの Communications Server for Linux からの 31
- マスター・サーバー 6
- マルチドメイン・サポート・ネットワーク管理ベクトル・トランスポート (MDS-NMVT) 15
- 命名規則 26
- メイン・メモリー 25
- メディア・アクセス制御 (MAC) 89
- メモリー所要量 24
 - 拡張構成 24
 - ストレージのタイプ 25
- 問題
 - 診断ツール 16
 - データ 16

[ヤ行]

- 要員要件 21
- 要件
 - インストール 22
 - オペレーティング・システム 22
 - ソフトウェア 22, 23
 - メモリーおよびストレージ 24
 - 要員とスキル 21
 - HTTPS 23
 - WebSphere Application Server 23

[ラ行]

- ライセンス・プログラム 34
 - AIX 上の Remote API Client 56
 - Linux for System z 上の Remote API Client 51
 - Linux 上の Remote API Client 45
- ランダム・アクセス・メモリー (RAM) 25

リソース

- 開始 81
- 管理 79
- 項目 82
- 削除 81
- 情報 119
- 定義 81
- 停止 81
- 要件 20

リモート・ノード

- 構成 97, 98
- パートナー LU の構成 99

リリース情報 38

リンク・ステーション

- 活動化 117
- ポートでの定義 88

リンク・ハードウェア 23

ローエントリー・ネットワークング (LEN) ノード 89

ローカル LU の定義 96

ローカル/リモート透過性 4

ローカル・エリア・ネットワーク (LAN) 2

論理装置 (LU) 2, 106

[ワ行]

ワークシート、計画 76

ワイルドカード 99

[数字]

1 次 RUI 3

1 次ストレージ 25

3270 LU

- 定義 111
- TN サーバーの 11, 110

A

API

- サポート 4
- Communications Server for Linux 管理用 14
- Communications Server for Linux のタイプ 5

APPC

- オンライン 120
- 構成 93
- 従属 LU 6.2 100
- 独立 LU 6.2 96
- 分散アプリケーションのサポート 6
- application suite 7, 9

APPC Application Suite 9

APPN

- アプリケーション 7, 9
- エンド・ノード 2
- 構成 95, 96
- サブエリア機能 2

APPN (続き)

- 接続ネットワーク 14
- 動的構成 14
- 独立 LU サポート 3
- ネットワークのセグメンテーション 26
- ネットワーク・ノード 2
- ノード・タイプ 1
- ホスト・サポート 2
- ルーティング 14
- DLUR サポート 2

C

CPI コミュニケーション (CPI-C) 4

CPI-C

- 構成 101
- 相互運用性 16
- API 4

D

DDDLU (従属 LU の動的定義) 3

DLC

- 構成 89, 90
- ポート構成の 87

DLUR

- 構成 106
- 説明 2
- ダウンストリーム・ノードのサポート 109
- ローカル・ノード上の 108
- PU 構成 108

DLUS 107

Domain (ドメイン) パラメーター 65

G

GSKIT

- アンインストール、Windows 上の Remote API Client 73
- AIX 上の Remote API Client 55, 58
- Linux for System z 上の Remote API Client 50, 52
- Linux 上の Remote API Client 44, 46
- Windows 上の Remote API Client 70, 71

H

HACL 36

Host Access Class Libraries 36

HPR

- ISR との比較 14

HPR/IP 10

HTTPS

- 構成 36
- 要件 23
- AIX 上の Remote API Client 58
- Linux for System z 上の Remote API Client 52

HTTPS (続き)
Linux 上の Remote API Client 46
Windows 上の Remote API Client 70, 71

I

installibmcs 35
installibmcsdocs 35
IP ポート・ダイアログ 91
ISO (国際標準化機構) 27
ISR (中間セッション・ルーティング) 14

J

Java
AIX 上の Remote API Client 55
Linux for System z 上の Remote API Client 50
Linux 上の Remote API Client 44

L

LAN access time-out (LAN アクセス・タイムアウト) パラメーター 66
LAN (ローカル・エリア・ネットワーク) 2
LEN ノード
説明 2
パートナー LU の構成 97
リモート・ノードの識別 89
Linux for System z 上の Remote API Client
既存のパッケージの詳細 50
Linux 上の Remote API Client
既存のパッケージの詳細 44
LU
構成 96, 100, 112
サポート 2
ダウンストリーム 104, 106
パートナーの定義 97
プール 102, 103, 111, 112
命名規則 27
LU タイプ 0 から 3 91, 92
LU プール
構成 93
定義 92
表示 93

M

MAC (メディア・アクセス制御) 89
Max. broadcast attempts (最大ブロードキャスト試行回数) パラメーター 67
MDS-NMVT (マルチドメイン・サポート・ネットワーク管理ベクトル・トランスポート) 15
Motif 管理プログラム
管理機能 16
使用 75

Motif 管理プログラム (続き)
説明 13
ヘルプ 13

N

「Node (ノード)」ウィンドウ 79
NOF API 14

O

OSI (オープン・システム間相互接続) 27

P

PC 拡張パワー・マネージメント機能 30
PDF、資料の表示 38
PU コンセントレーション 103
PU コンセントレーター 7
PU (物理装置) 7

Q

query コマンド 16

R

RAM (ランダム・アクセス・メモリー) 25
Reconnect time-out (再接続タイムアウト) パラメーター 67
Remote API Client
AIX ソフトウェア要件 55
AIX ハードウェア要件 55
Linux for System z のソフトウェア要件 49
Linux for System z のハードウェア要件 49
Linux ソフトウェア要件 44
Linux ハードウェア要件 43
rpm 21

S

SAA (システム・アプリケーション体系) 5
SAP (サービス・アクセス・ポイント) 22, 89
SDK ソフトウェア
Windows 上の Remote API Client 62
SDLC 2
構成 88
従属トラフィックの 87
Secure Sockets Layer (SSL)
クライアント認証 113, 115
サーバー認証 39, 113, 115
データ暗号化 39, 113, 115
Server Name (サーバー名) 65
SNA
ライブラリー 119

SNA ゲートウェイ
概要 7
構成 103
snaadmin プログラム 14
snastart ファイル 78, 118
status コマンド 16

T

TN サーバー
アクセス・レコードの構成 114
アソシエーション・レコードの構成 113, 114
概要 10
構成 109
デフォルト構成 114
複数セッションのサポート 111
ユーザー 110
TN リダイレクター
アクセス・レコードの構成 115
概要 12
構成 115
デフォルト構成 115
TN3270
サーバー 10
プログラム 10
TN3270 プログラム 109
TN3270 ユーザー 11, 110
TP (トランザクション・プログラム) 4

U

U 型セッション 4
UDP broadcasts (UDP ブロードキャスト) パラメーター 65

V

verb 4

W

WAN (広域ネットワーク) 2
WebSphere Application Server
構成 36
要件 23
Windows 上の Remote API Client
インストール 62
コマンド行からのインストール 67
GSKIT のアンインストール 73
setup プログラムによるインストール 63

X

xsnaadmin プログラム 13, 75



プログラム番号: 5724-i33

Printed in Japan

GC88-9996-01



日本アイ・ビー・エム株式会社
〒106-8711 東京都港区六本木3-2-12