

**WebSphere Application Server**



## **Edge Components 概念、計画とインストール**

*バージョン 6.0*



**WebSphere Application Server**



## **Edge Components 概念、計画とインストール**

*バージョン 6.0*

**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、89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。

この版は、以下のプログラムに適用されます。

WebSphere Application Server、バージョン 6.0

また、新しい版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。

資料を注文する場合は、IBM 担当者または最寄の IBM 営業所にご連絡ください。

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：** GC31-6855-00  
WebSphere Application Server  
Concepts, Planning, and Installation for Edge Components  
Version 6.0

**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 2004.10

この文書では、平成明朝体™W3、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™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 2004. All rights reserved.

© Copyright IBM Japan 2004

# 目次

図	v
本書について	vii
本書の対象読者	vii
アクセシビリティ	vii
本書で使用されている規則と用語	vii

## 第 1 部 概説 . . . . . 1

### 第 1 章 WebSphere Application Server Edge components の紹介 . . . . . 3

Caching Proxy	3
Load Balancer	4
Dispatcher	5
Content Based Routing	6
Site Selector	6
Cisco CSS Controller	6
Nortel Alteon Controller	6
Metric Server	6

### 第 2 章 Edge components と WebSphere ファミリー . . . . . 7

Tivoli Access Manager	7
WebSphere Portal Server	7
WebSphere Site Analyzer	7
WebSphere Transcoding Publisher	8

### 第 3 章 Application Server および Edge components の詳細情報 . . . . . 9

## 第 2 部 Edge コンポーネントの概念 および説明 . . . . . 11

### 第 4 章 キャッシング . . . . . 13

基本キャッシング	13
拡張キャッシング	14
ロード・バランスされた Caching Proxy クラスタ	14
動的コンテンツのキャッシュ	15
その他のキャッシング機能	16

### 第 5 章 ネットワーク・パフォーマンス . . . . . 17

ネットワーク・ハードウェア	17
メモリーに関する考慮事項	17
ハード・ディスクに関する考慮事項	17
ネットワークに関する考慮事項	18
CPU に関する考慮事項	18
ネットワーク・アーキテクチャー	18

Web サイトの人気とプロキシ・サーバーの負荷に関する考慮事項	18
トラフィック・タイプに関する考慮事項	20

### 第 6 章 可用性 . . . . . 23

ロード・バランシング	23
複数のコンテンツ・ホストのロード・バランシング	23
複数のプロキシ・サーバーのロード・バランシング	24
フェイルオーバー・サポート	26

### 第 7 章 Content Based Routing . . . . . 29

## 第 3 部 シナリオ . . . . . 33

### 第 8 章 企業消費者間ネットワーク . . . . . 35

フェーズ 1	35
フェーズ 2	37
フェーズ 3	38

### 第 9 章 企業クライアント間バンキング・ソリューション . . . . . 41

### 第 10 章 Web ポータル・ネットワーク . . . . . 43

## 第 4 部 Edge components のインストール . . . . . 45

### 第 11 章 Edge components の要件 . . . . . 47

ハードウェアおよびソフトウェアの前提条件	47
AIX システムでの Caching Proxy のインストール前提条件	47
AIX システムでの Load Balancer のインストール前提条件	48
HP-UX システムでの Caching Proxy のインストール前提条件	48
HP-UX システムでの Load Balancer のインストール前提条件	49
Linux システムでの Caching Proxy のインストール前提条件	50
Linux システムでの Load Balancer のインストール前提条件	51
Solaris システムでの Caching Proxy のインストール前提条件	53
Solaris システムでの Load Balancer のインストール前提条件	54
Windows システムでの Caching Proxy のインストール前提条件	55

Windows システムでの Load Balancer のインストール前提条件 . . . . .	55
Caching Proxy 「構成および管理」フォームを使用するブラウザの使用 . . . . .	56
Load Balancer オンライン・ヘルプでのブラウザの使用 . . . . .	57
<b>第 12 章 セットアップ・プログラムを使用した Edge components のインストール . . . . .</b>	<b>59</b>
Windows でのセットアップ・プログラムの使用 . . . . .	59
Linux および UNIX でのセットアップ・プログラムの使用 . . . . .	61
<b>第 13 章 システム・パッケージ・ツールを使用した Caching Proxy のインストール . . . . .</b>	<b>63</b>
<b>第 14 章 システム・パッケージ・ツールを使用した Load Balancer のインストール . . . . .</b>	<b>67</b>
AIX へのインストール . . . . .	68
インストールする前に . . . . .	68
インストール手順 . . . . .	69
HP-UX へのインストール . . . . .	71
インストールする前に . . . . .	71
インストール手順 . . . . .	71
Linux へのインストール . . . . .	73
インストールする前に . . . . .	73
インストール・ステップ . . . . .	73
Solaris へのインストール . . . . .	75

インストールする前に . . . . .	75
インストール・ステップ . . . . .	75

---

## 第 5 部 Edge components を使用したネットワークの構築 . . . . . 77

<b>第 15 章 Caching Proxy ネットワークの構築 . . . . .</b>	<b>79</b>
ワークフロー . . . . .	79
必須コンピューター・システムおよびソフトウェアの確認 . . . . .	80
サーバー 1 の構築 (Linux および UNIX システム) . . . . .	80
サーバー 1 の構築 (Windows システム) . . . . .	80
サーバー 1 の構成 . . . . .	80
Caching Proxy ネットワークのテスト . . . . .	81

<b>第 16 章 Load Balancer ネットワークの構築 . . . . .</b>	<b>83</b>
ワークフロー . . . . .	83
必須コンピューター・システムおよびソフトウェアの確認 . . . . .	83
ネットワークの構成 . . . . .	84
Dispatcher の構成 . . . . .	85
コマンド行を使用した構成 . . . . .	85
構成ウィザードを使用した構成 . . . . .	86
グラフィカル・ユーザー・インターフェース (GUI) を使用した構成 . . . . .	86
Load Balancer ネットワークのテスト . . . . .	87

<b>特記事項 . . . . .</b>	<b>89</b>
商標 . . . . .	90



1. 基本プロキシ構成 . . . . .	13	7. CBR による HTTP 要求の経路指定 . . . . .	30
2. ロード・バランスの対象となるクラスターの プロキシ・サーバーとして機能する Caching Proxy . . . . .	14	8. CBR により経路指定された HTTP 要求のロード・ バランシング . . . . .	31
3. 複数のコンテンツ・ホストのロード・バラン シング . . . . .	24	9. 企業消費者間ネットワーク (フェーズ 1) . . . . .	36
4. 複数のプロキシ・サーバーおよびコンテン ツ・ホストのロード・バランシング . . . . .	25	10. 企業消費者間ネットワーク (フェーズ 2) . . . . .	38
5. Web コンテンツを高可用にするためのプライマ リーおよびバックアップ Load Balancer の使用 . . . . .	26	11. 企業消費者間ネットワーク (フェーズ 3) . . . . .	39
6. コンテンツ・ホストへのバックアップ Load Balancer の配置 . . . . .	27	12. 企業消費者間バンキング・ソリューション . . . . .	42
		13. Web ポータル . . . . .	44
		14. Caching Proxy デモンストレーション・ネット ワーク . . . . .	79
		15. Load Balancer デモンストレーション・ネット ワーク . . . . .	83





---

## 本書について

本書 *WebSphere® Application Server Edge Components* 概念、計画とインストール は、WebSphere Application Server Edge components の入門書です。本書は、製品についての高水準な概説、キー・コンポーネントの機能についての詳細な解説、ネットワーク・エッジ・シナリオ、インストールおよび初期構成情報、およびデモンストラーション・ネットワークを提供します。

---

## 本書の対象読者

*WebSphere Application Server Edge Components* 概念、計画とインストール は、オペレーティング・システムと、インターネット・サービスの提供に精通した、経験豊富なネットワークおよびシステム管理者を対象として作成されています。

WebSphere Application Server または WebSphere Application Server Edge components についての知識が事前になくても、問題はありません。

---

## アクセシビリティ

アクセシビリティ機能は、運動障害または視覚障害など身体に障害を持つユーザーがソフトウェア・プロダクトを快適に使用できるようにサポートします。以下は、WebSphere Application Server、バージョン 6.0 の主要なアクセシビリティ機能です。

- ユーザーは、スクリーン・リーダー・ソフトウェアやデジタル・スピーチ・シンセサイザーを使用して、画面上に表示された内容を聞くことができます。また、IBM® ViaVoice™ などの音声認識ソフトウェアを使用して、データを入力し、ユーザー・インターフェースをナビゲートすることもできます。
- マウスの代わりにキーボードを使用することによって、機能を操作することができます。
- 提供されたグラフィカル・インターフェースの代わりに、標準テキスト・エディターまたはコマンド行インターフェースを使用して、Application Server 機能を構成および管理することができます。特定の機能のアクセシビリティについての詳細は、それらの機能に関する資料を参照してください。

---

## 本書で使用されている規則と用語

本書では、以下のような書体およびキー操作の規則を使用しています。

表 1. 本書の規則

規則	意味
太字	グラフィカル・ユーザー・インターフェース (GUI) に関しては、太字は、メニュー、メニュー項目、ラベル、ボタン、アイコン、およびフォルダーを示します。また、太字にしないと周りのテキストと混同される恐れがあるコマンド名を強調するためにも使用されます。
モノスペース	コマンド・プロンプトから入力する必要のあるテキストを示します。また、モノスペースは、画面上のテキスト、コード例、およびファイルからの引用も示します。

表 1. 本書の規則 (続き)

規則	意味
イタリック	指定する必要のある可変値を示します (例 : <i>fileName</i> でファイルの名前を指定します)。イタリックは、強調表示および書名の表示にも使用されます。
Ctrl-x	x がキーの名前である場合、制御文字のシーケンスを示します。例えば、Ctrl-c は、Ctrl キーを押しながら c キーを押すという意味になります。
Return	Return、Enter または左矢印が表示されていたキーを指します。
%	Linux および UNIX® のコマンド・シェル・プロンプト (ルート権限を必要としないコマンド用) を示します。
#	Linux および UNIX のコマンド・シェル・プロンプト (ルート権限を必要とするコマンド用) を示します。
C:¥	Windows コマンド・プロンプトを表します。
コマンド入力	コマンドを“入力”する、または“発行”すると指示された場合は、コマンドを入力してから Return を押します。例えば、「 <b>ls</b> コマンドを入力してください」という指示は、コマンド・プロンプトに <b>ls</b> とタイプしてから Return を押すという意味になります。
[ ]	構文記述内のオプション項目を囲みます。
{ }	構文記述内の、項目を選択する必要があるリストを囲みます。
	構文記述の中で { } (中括弧) で囲まれた選択項目リストにある項目を区切るために使用されます。
...	構文記述内の省略記号は、前の項目を 1 回以上繰り返すことができることを示します。例にある省略記号は、簡潔にするために例から情報が省略されていることを示します。

---

## 第 1 部 概説

この部では、WebSphere Application Server Edge components、Caching Proxy、および Load Balancer の概要と、これらを Application Server と統合する方法について説明します。また、Caching Proxy および Load Balancer のコンポーネントを定義します。さらに、このセクションでは、他の関連する WebSphere ファミリー製品も紹介します。

この部は、以下の章で構成されています。

- 3 ページの『第 1 章 WebSphere Application Server Edge components の紹介』
- 7 ページの『第 2 章 Edge components と WebSphere ファミリー』
- 9 ページの『第 3 章 Application Server および Edge components の詳細情報』



---

## 第 1 章 WebSphere Application Server Edge components の紹介

WebSphere は、企業間 e-commerce 用アプリケーションなどの次世代 e-business アプリケーションの、企業による開発、展開、および統合を可能にするインターネット・インフラストラクチャー・ソフトウェアです。WebSphere ミドルウェアは、単純な Web 公開から企業規模のトランザクション処理までのビジネス・アプリケーションをサポートします。

WebSphere Application Server は、WebSphere プラットフォームの基礎として、ユーザーによるビジネス・アプリケーションの設計、インプリメント、展開、および管理を可能にする 包括的なミドルウェアのセットを提供します。これらのアプリケーションは、単純な Web サイト・ストアフロントから組織のコンピューター・インフラストラクチャーの完全な改訂まで多岐にわたります。

パーソナライゼーションなどのプロセッサ集中機能は、すべての e-business の競争力を高めます。ただし、これらの機能を中央サーバーに常に任せていると、インターネットへの重要な機能の拡張が妨げられる可能性があります。したがって、新しい Web アプリケーションを常に追加して、企業のインターネット・インフラストラクチャーの範囲を広げて性能を高める必要があります。また、e-business にとって、信頼性とセキュリティは極めて重要です。サービスがごく短い間中断するだけで、取引を失うこともあります。

Edge components (従来の Edge Server) は、WebSphere Application Server オフラインの一部になりました。Edge components を WebSphere Application Server と共に使用すると、Web サーバーへのクライアント・アクセスを制御して、インターネットまたは企業のイントラネットを介して Web ベース・コンテンツにアクセスするユーザーに企業がより良いサービスを提供できるようになります。Edge components を使用すると、Web サーバーの輻輳 (ふくそう) を削減し、コンテンツの可用性を高め、Web サーバーのパフォーマンスを向上させることができます。Edge components は、その名前が示すとおり、通常、企業のイントラネットとインターネットの境界に (ネットワーク構成上) 近い位置にあるマシンで実行されます。

WebSphere Application Server には Caching Proxy および Load Balancer Edge components が組み込まれています。

---

### Caching Proxy

Caching Proxy は、1 つ以上のバックエンド・コンテンツ・サーバー用の POP (Point-of-Presence) ノードを提供して、帯域幅使用量を削減し、Web サイトの速度と信頼性を高めます。Caching Proxy は、静的コンテンツおよび WebSphere Application Server が動的に生成したコンテンツをキャッシュに入れて処理できません。

プロキシ・サーバー では、クライアントのデータ要求をインターセプトして、コンテンツ・ホスティング・マシンの要求情報を引き出し、そのコンテンツをクライ

アントに引き渡します。通常エンド・ユーザーの要求の大部分は、Web サーバー・マシン (起点サーバー またはコンテンツ・ホスト ともいう) に保管されている文書を HTTP を使用して引き渡すよう求めるものです。しかし、プロキシ・サーバーの構成によっては、ファイル転送プロトコル (FTP) および Gopher など、他のプロトコルも扱うことができます。

プロキシ・サーバーは、キャッシュ可能コンテンツを、要求側に引き渡す前にローカル・キャッシュに保管します。キャッシュ可能コンテンツの例として、静的 Web ページと、動的に生成されたほとんど変更されない情報をもつ JavaServer Pages ファイルが挙げられます。キャッシングにより、プロキシ・サーバーは、以後同じコンテンツが要求されたときにそのコンテンツをキャッシュから直接引き渡すことができるので、改めてコンテンツ・ホストから検索するよりはるかに迅速に処理できます。

Caching Proxy 用のプラグインを使用すると、プロキシ・サーバーに機能が追加されます。

- ICP プラグインを使用すると、プロキシ・サーバーが HTML ページおよびその他のキャッシュ可能なリソースを検索したときに、インターネット・キャッシング・プロトコル (ICP) に準拠したキャッシュを照会することが可能になります。
- Tivoli® Access Manager (旧 Policy Director) プラグインは、プロキシ・サーバーが Access Manager の統合化された認証・許可サービスを使用できるようにします。
- PAC-LDAP 認証モジュールは、許可または認証ルーチンの実行時にプロキシ・サーバーが LDAP サーバーにアクセスできるようにします。
- WebSphere Transcoding Publisher プラグインを使用すると、プロキシ・サーバーが WebSphere Transcoding Publisher と共に使用されたときに、モバイル装置用に変換されたコンテンツの複数のバージョンをキャッシュすることが可能になります。

アプリケーション・プログラミング・インターフェース (API) に対するカスタム・プラグイン・モジュールを作成することによって、Caching Proxy の機能をさらに拡張することができます。API はフレキシブルで使用しやすく、プラットフォームに依存しません。プロキシは、処理するクライアント要求ごとに一連のステップを実行します。プラグイン・アプリケーションは、クライアント認証または要求フィルター操作などの要求処理ワークフロー内のステップを変更または置換します。例えば、強力な Transmogriphy インターフェースは、HTTP データへのアクセスを提供し、URL と Web コンテンツの置換または変換を可能にします。指定した処理ステップをプラグインを使用して変更または置換することが可能であり、特定のステップのために複数のプラグインを起動することができます。

---

## Load Balancer

Load Balancer は、ネットワーク・トラフィック・フローを管理するネットワーク・エッジ・システムを作成して、輻輳を削減し、その他のさまざまなサービスとシステムに対する負荷のバランスを行います。Load Balancer により、サイト選択、ワークロード管理、セッション類縁性、および透過性フェイルオーバーが可能になります。

Load Balancer は、インターネットと企業のバックエンド・サーバーの間にインストールされます。バックエンド・サーバーには、コンテンツ・ホストまたは Caching Proxy マシンを使用できます。大量の要求に応えるためまたは大量のコンテンツを扱うために企業で複数のバックエンド・サーバーを使用している場合にも、Load Balancer はインターネットにおける企業の単一の POP (Point-of-Presence) ノードとして機能します。また、プライマリー Load Balancer に一時的に障害が起きたときにそれを引き継ぐバックアップ Load Balancer をインストールすることによって、ハイ・アベイラビリティを保証することができます。

Load Balancer は、クライアントからのデータ要求をインターセプトして、現在各要求を満たすのに最も適しているサーバーに要求を転送します。つまり、Network Dispatcher は、同じタイプの要求にサービスを行うマシンの定義済みセット間で、着信要求のロードのバランスを取ります。Load Balancer は、WebSphere Application Server および Caching Proxy マシンを含むさまざまなタイプのサーバーに要求を分散することができます。カスタム advisor を使用することによって、ロード・バランシングを特定のアプリケーションまたはプラットフォーム用にカスタマイズできます。WebSphere Application Server のロード・バランシング用の情報を取得するために、特定目的の advisor を使用できます。

Content Based Routing コンポーネントが Caching Proxy と共にインストールされている場合、HTTP および HTTPS 要求を URL または管理者が決定したその他の特性に基づいて分散することも可能になります。これにより、すべてのバックエンド・サーバーに同一のコンテンツを保管する必要がなくなります。Dispatcher コンポーネントも HTTP 要求についての同じ機能を提供できます。

ロード・バランシングを使用すると、コンテンツ・サーバーが透過的にクラスター化されるため、Web サイトの可用性とスケーラビリティが向上します。コンテンツ・サーバーには、HTTP サーバー、アプリケーション・サーバー、および代理コンテンツ・サーバーであるプロキシ・サーバーが含まれます。並列性、ロード・バランシング、およびフェイルオーバー・サポートによって、可用性が向上します。サーバーがダウンしても、取引は妨げられません。バックエンド処理能力を透過的に追加できることによって、インフラストラクチャーのスケーラビリティは非常に向上します。

Load Balancer には以下のコンポーネントが組み込まれています。

## Dispatcher

HTTP、FTP、HTTPS、および Telnet のようなすべてのインターネット・サービスの場合、Dispatcher コンポーネントは、ローカル・エリア・ネットワーク (LAN) または広域ネットワーク (WAN) 内のサーバーでロード・バランシングを実行します。HTTP サービスの場合、Dispatcher はサーバーのロード・バランシングをクライアント要求の URL コンテンツに基づいて実行できます。

Dispatcher コンポーネントを使用すると、大規模でスケーラブルなサーバーのネットワークを継続的かつ効率よく管理できます。Dispatcher を使用して多くの個々のサーバーをリンクし、単一の仮想サーバーとして扱うことができます。したがって、サイトは単一の IP アドレスとして表示されます。

## Content Based Routing

HTTP および HTTPS サービスで、Content Based Routing コンポーネントは、クライアント要求のコンテンツに基づいたサーバーのロード・バランシングを実行します。Content Based Routing コンポーネントは、Application Server Caching Proxy コンポーネントと共に機能します。

## Site Selector

Site Selector コンポーネントは、ロード・バランシング・システムをネットワークの POP (Point-of-Presence) ノードとして機能させ、DNS 名を IP アドレスにマップすることによって着信要求のロード・バランシングを可能にすることで、ロード・バランシング・システムの機能を拡張します。Site Selector は、Metric Server と共に使用すると、サーバー上のアクティビティー・レベルをモニターし、サーバーの負荷がいつ最小になるかを検出し、障害の起きたサーバーを検出することができます。

## Cisco CSS Controller

Cisco CSS Controller コンポーネントは、サーバー選択、負荷最適化、およびフォールト・トレランスのために Cisco CSS スイッチに送信される、サーバー加重メトリックを生成します。

## Nortel Alteon Controller

Nortel Alteon Controller コンポーネントは、サーバー選択、負荷最適化、およびフォールト・トレランスのために Nortel Alteon スイッチに送信される、サーバー加重メトリックを生成します。

## Metric Server

Metric Server コンポーネントは、ロード・バランシングされたサーバーでデーモンとして実行され、Load Balancer コンポーネントにシステム負荷についての情報を提供します。



---

## 第 2 章 Edge components と WebSphere ファミリー

IBM WebSphere ファミリーは、ユーザーによる e-business の可能性の実現を助けることを目的に設計されています。これは、ユーザーによる高性能な Web サイトの開発と管理、および新規または既存の非 Web ビジネス情報システムとの Web サイトの統合に役立つ、一連のソフトウェア製品です。

WebSphere ファミリーは、Edge components を含む WebSphere Application Server と、WebSphere Application Server と緊密に統合されてそのパフォーマンスを向上させるその他の WebSphere ファミリー・ソフトウェアで構成されています。

WebSphere Application Server とそのコンポーネントの概説については、3 ページの『第 1 章 WebSphere Application Server Edge components の紹介』を参照してください。

---

### Tivoli Access Manager

Tivoli Access Manager (旧 Tivoli Policy Director) は個別に入手可能です。これは、既存の Web アプリケーションのアクセスの制御と集中セキュリティー、および複数の Web リソースへのアクセスでの一回限りの認証機能を提供します。Caching Proxy プラグインは、プロキシ・サーバーに Access Manager に組み込まれた許可または認証サービスを使用可能にし、Access Manager のセキュリティー・フレームワークを活用します。

---

### WebSphere Portal Server

WebSphere Portal Server (別個に使用可能) は、ポータルに関連した、プレゼンテーション、セキュリティー、スケーラビリティ、および可用性の課題に応えるフレームワークを提供します。Portal Server を使用すると、企業が従業員、ビジネス・パートナー、および顧客の要求に応えるための、独自のカスタム・ポータル Web サイトを作成することができます。ユーザーは、ポータルにサインオンして、必要な情報、個人、アプリケーションにアクセスできるパーソナライズされた Web ページを受信できます。必要なすべてのリソースへのこのパーソナライズされた単一アクセス・ポイントは、情報の過負荷を削減し、生産性を上げ、Web サイトの使用量を増やします。

WebSphere Portal Server は WebSphere Application Server クラスター内で実行され、スケーラビリティと信頼性を実現します。さらに、ロード・バランシングとハイ・アベイラビリティを向上させるために、Application Server Load Balancer コンポーネントも使用できます。

---

### WebSphere Site Analyzer

WebSphere Site Analyzer (別個に使用可能) を使用すると、企業は容量とパフォーマンスの問題を予測しやすくなります。Site Analyzer があると、Caching Proxy および Load Balancer ログと、その他の管理容易化機能を使用して、Web サイトの使用率をモニターし、分析し、報告させることで、追加リソースがどれだけ必要になる

かを予測することができます。さらに、Site Analyzer 管理容易化コンポーネントは、Edge components のインストールとアップグレード、構成の管理と保管、リモートでの Edge components の操作、およびイベントの表示と報告を行うユーザーに役立ちます。

---

## WebSphere Transcoding Publisher

WebSphere Transcoding Publisher (別個に使用可能) は、インターネットに接続できる電話などのモバイル装置で表示するための Web ページの変換、ユーザーの希望する各国語への Web コンテンツの翻訳 (WebSphere Translation Server を起動)、マークアップ言語の変換を行うことができます。Transcoding Publisher は、Caching Proxy の能力を拡張して、コンテンツをさまざまなデバイスとユーザーに提供可能にします。Web サーバーからのコンテンツへのアクセス後に、Caching Proxy の Transmogrify インターフェースを構成して、Transcoding Publisher を起動してデータを変換し、バリエーション・キャッシングおよび可能な再利用のためのタグを付けることができます。次に、Transcoding Publisher は、Caching Proxy の認証後インターフェースで、コンテンツがユーザーとデバイスの要件を満たすかどうかプロキシ・サーバーを検査して、要件を満たすコンテンツが見つかったらプロキシ・サーバーのキャッシュからそのコンテンツを提供します。

---

## 第 3 章 Application Server および Edge components の詳細情報

以下の WebSphere Application Server Edge Components 専門資料を、Edge Components Information Center で入手することができます。

- *Edge Components プログラミング・ガイド* GC88-7045-00
- *Caching Proxy 管理ガイド* GC88-7050-00
- *Load Balancer 管理ガイド* GC88-7053-00

他の WebSphere Application Server 文書は、WebSphere Application Server ライブラリー・ページから入手できます。

Edge Components に関する技術サポート情報は、WebSphere Application Server サポート・ページから入手できます。

以下に、Edge Components の情報またはその関連情報を入手するための Web サイトを挙げます。

- IBM Web サイト・ホーム <http://www.ibm.com/>
- IBM WebSphere Application Server <http://www.ibm.com/software/webservers/appserv/>
- IBM WebSphere Application Server ライブラリー Web サイト  
<http://www.ibm.com/software/webservers/appserv/library.html>
- IBM WebSphere Application Server サポート Web サイト  
<http://www.ibm.com/software/webservers/appserv/support.html>
- IBM WebSphere Application Server Information Center  
<http://www.ibm.com/software/webservers/appserv/infocenter.html>
- IBM WebSphere Application Server Edge Components Information Center  
<http://www.ibm.com/software/webservers/appserv/ecinfocenter.html>



---

## 第 2 部 Edge コンポーネントの概念および説明

この部では、Edge コンポーネントで使用可能ないくつかの機能について特に詳しく説明します。Application Server の Caching Proxy コンポーネントの概説については、3 ページの『第 1 章 WebSphere Application Server Edge components の紹介』を参照してください。

この部は、以下の章で構成されています。

13 ページの『第 4 章 キャッシング』

17 ページの『第 5 章 ネットワーク・パフォーマンス』

23 ページの『第 6 章 可用性』

29 ページの『第 7 章 Content Based Routing』



## 第 4 章 キャッシング

Caching Proxy のキャッシング機能を使用すると、ネットワーク帯域幅の使用率が最小化され、エンド・ユーザーはより高速で信頼できるサービスを受け取ることができます。これはプロキシ・サーバーがキャッシングを実行して、バックエンド・サーバーとピア・リンクがオフロードされるために実現可能となります。Caching Proxy は、静的コンテンツと、WebSphere Application Server によって動的に生成されたコンテンツをキャッシュに入れることができます。また、拡張キャッシングを行うために、Caching Proxy は Application Server Load Balancer コンポーネントと連動して機能します。これらのシステムの概要については、3 ページの『第 1 章 WebSphere Application Server Edge components の紹介』を参照してください。

### 基本キャッシング

Caching Proxy マシンは、インターネットと企業のコンテンツ・ホストとの間に位置付けられます。プロキシ・サーバーは代理として機能し、インターネットから到着したユーザー要求をインターセプトして、それを該当するコンテンツ・ホストに転送し、戻りデータをキャッシュに入れてから、そのデータをインターネット経由でユーザーに引き渡します。このプロセスでキャッシングを行うことにより、Caching Proxy は以後同じコンテンツが要求されたときにその要求をキャッシュから直接処理することができるため、改めてコンテンツ・ホストから検索するよりはるかに迅速に処理できます。情報は、その有効期限、キャッシュのサイズ、および情報の更新時期にしたがってキャッシュに入れることが可能です。キャッシュ・ヒットをより高速にダウンロードできるということは、カスタマーにとっては、サービスの品質がよりよいということになります。図 1 はこの基本 Caching Proxy 機能を示しています。

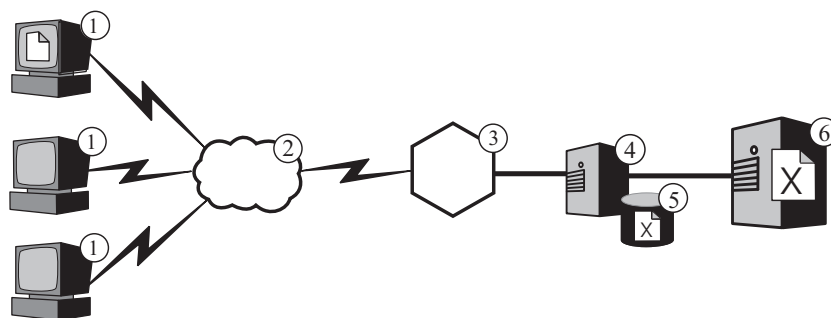


図 1. 基本プロキシ構成： 凡例: 1—クライアント 2—インターネット 3—ルーター / ゲートウェイ 4—Caching Proxy 5—キャッシュ 6—コンテンツ・ホスト

この構成において、プロキシ・サーバー (4) は、URL にコンテンツ・ホストのホスト名 (6) が含まれている要求をインターセプトします。クライアント (1) がファイル X を要求すると、その要求はインターネット (2) を経由し、企業のインターネット・ゲートウェイ (3) を通過してその企業の内部ネットワークに入ります。プ

ロキシー・サーバーは、要求をインターセプトし、自分自身の IP アドレスを発信アドレスとして持つ新規要求を生成して、その新規要求をコンテンツ・ホスト (6) に送信します。

コンテンツ・ホストは、ファイル **X** を、エンド・ユーザーに直接戻さないでプロキシー・サーバーに戻します。ファイルがキャッシュ可能なら、Caching Proxy はファイルをエンド・ユーザーに引き渡す前にキャッシュ (5) にコピーを保管します。キャッシュ可能コンテンツの最もわかりやすい例は静的 Web ページです。ただし、Caching Proxy にも、WebSphere Application Server によって動的に生成されたコンテンツをキャッシュに入れて、処理する機能があります。

## 拡張キャッシング

### ロード・バランスされた Caching Proxy クラスター

より高度なキャッシング機能を利用するには、Application Server の Load Balancer コンポーネントとともに Caching Proxy を使用します。キャッシング機能とロード・balancing機能を統合することによって、能率的で非常に扱いやすい Web パフォーマンス・インフラストラクチャーを作成できます。

図2 は、Caching Proxyを Load Balancer と結合して、大量要求の状況でも Web コンテンツを効率的に引き渡せる方法を描いたものです。この構成では、プロキシー・サーバー (4) は、Load Balancer (6) によってロード・balancingされているコンテンツ・ホスト (7) のクラスターのホスト名がその URL に含まれている要求をインターセプトするように構成されます。

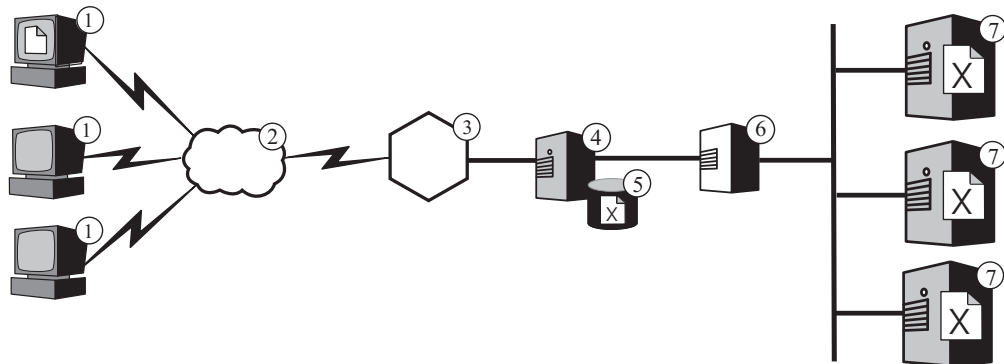


図2. ロード・balancingの対象となるクラスターのプロキシー・サーバーとして機能する Caching Proxy : (凡例: 1—クライアント 2—インターネット 3—ルーター / ゲートウェイ 4—Caching Proxy 5—キャッシュ 6—Load Balancer 7—コンテンツ・ホスト)

クライアント (1) がファイル **X** を要求すると、その要求はインターネット (2) を経由し、企業のインターネット・ゲートウェイ (3) を通過してその企業の内部ネットワークに入ります。プロキシー・サーバーは、要求をインターセプトし、自分自身の IP アドレスを発信アドレスとして持つ新規要求を生成して、その新規要求をクラスター・アドレスの Load Balancer に送信します。Load Balancer はそのロード・balancingのアルゴリズムを使用して、ファイル **X** に関する要求を満たすのに現在どのコンテンツ・ホストが最も適しているかを判断します。そのコンテンツ・ホストは、Load Balancer を通じてではなく、ファイル **X** をプロキシー・サー



バーに戻します。プロキシ・サーバーは、ファイルをキャッシュするかどうかを判断し、前に説明したのと同じ方法でエンド・ユーザーに引き渡します。

## 動的コンテンツのキャッシュ

Caching Proxy の Dynamic Caching プラグインも、拡張キャッシング機能を提供します。WebSphere Application Server とともに使用すると、Caching Proxy は、WebSphere Application Server によって生成された JavaServer Pages (JSP) およびサーブレット応答という形式の動的コンテンツをキャッシュ、処理、および無効化できます。

時間に基づく標準的なキャッシュ有効期限ロジックでは、有効期限が無期限に設定されている動的コンテンツは適切な状況で必ず除去されるとは限らないため、一般にこのようなコンテンツは「キャッシュに入れない」とマークする必要があります。Dynamic Caching プラグインのイベント・ドリブン有効期限ロジックを使用すると、プロキシ・サーバーによって無期限の有効期限を持つコンテンツをキャッシュに入れることができます。このようなコンテンツがネットワークの端にある場合、これをキャッシュに入れると、コンテンツ・ホストがクライアントからの要求を満たすためにアプリケーション・サーバーを繰り返し起動する必要がなくなります。これには次のような利点があります。

- Web サーバー、WebSphere Application Server、およびバックエンド・コンテンツ・ホストの作業負荷が削減される
- ネットワーク遅延を除去することによりユーザーへの応答が速くなる
- インターネットの全探索が少なくなるため帯域幅の使用量が削減される
- 動的に生成されたコンテンツを処理する Web サイトのスケラビリティが向上する

アプリケーション・ロジックまたはデータベースからのメッセージなどのイベントに基づいて有効期限が切れる、動的に生成された Web ページの場合、サーブレット応答キャッシングが理想的です。このようなページの存続時間は限定されていますが、有効期限トリガーを前もって認識することはできないので、作成時に存続時間値を設定することは不可能です。このようなページの存続時間をゼロに設定すると、コンテンツ・ホストは、動的コンテンツの処理時に大きなペナルティーを被ります。

Caching Proxy および Application Server の動的キャッシュの同期化は、両方のシステムによって履行されます。例えば、現在の天気予報を提供するアプリケーションによって動的に作成された公開 Web ページを、Application Server でエクスポートして、Caching Proxy でキャッシュに入れることができます。これで、Caching Proxy は、このページが無効になったことを通知されるまで、アプリケーションの実行結果を多くの別のユーザーに繰り返し供給できます。キャッシュが輻輳しているか、Caching Proxy の構成ファイル内の ExternalCacheManager ディレクティブによって設定されたデフォルト・タイムアウトの有効期限が切れたか、またはキャッシュからコンテンツをパージするように指示する無効化 (Invalidate) メッセージを Caching Proxy が受け取った、という理由でプロキシ・サーバーが項目を除去するまで、Caching Proxy のサーブレット応答キャッシュ内のコンテンツは有効です。Invalidate 無効化 (Invalidate) メッセージはコンテンツを所有する WebSphere Application Server で生成され、構成された各 Caching Proxy に伝搬されます。

**注:** 一般に、動的に生成されたプライベート・ページ (ユーザーのショッピング・カートの内容を表示するページなど) は Caching Proxy によってキャッシュに入れることはできず、また入れられないようになっています。意図されたユーザーにのみ供給されることを保証するために認証および許可を行うようにプライベート・ページが構成されているときにのみ、Caching Proxy はそのプライベート・ページをキャッシュに入れて、処理することができます。

## その他のキャッシング機能

Caching Proxy は、その他にも以下の重要な拡張キャッシング機能を提供します。

- 非常に大規模なキャッシュを使用するための機能
- 最も頻繁にアクセスされるページに関して自動的にキャッシュをリフレッシュするためのオプション
- ヘッダー情報が毎回取り出すように指示するページをもキャッシュに入れることが可能
- サーバーのパフォーマンスを向上させ、キャッシュの保守を確実にを行うための構成可能日次ガーベッジ・コレクション
- リモート・キャッシュ・アクセス (RCA)。この機能を使用すると、複数の Caching Proxy マシンで、同じキャッシュを共用し、キャッシュに入れられたコンテンツの冗長度を減らすことができる
- ICP プラグイン。これを使用すると、HTML ページおよびその他のキャッシュ可能なリソースの検索で、Caching Proxy がインターネット・キャッシング・プロトコル (ICP) 対応のキャッシュを照会できるようになる

---

## 第 5 章 ネットワーク・パフォーマンス

Caching Proxy 機能の導入はネットワーク・パフォーマンスに影響を与えます。ネットワークのパフォーマンスを向上させるには、Caching Proxy を単独で使用するか、または Load Balancer と共に使用してください。これらのシステムの概要については、3 ページの『第 1 章 WebSphere Application Server Edge components の紹介』を参照してください。

企業内の Caching Proxy のパフォーマンスは、実行されるハードウェアと、導入されるシステムの全体的なアーキテクチャーによって決まります。ネットワーク・パフォーマンスを最適化するには、ハードウェアおよび全体的なネットワーク・アーキテクチャーをプロキシ・サーバーの特性に合わせてください。

**注:** もちろん、Caching Proxy ソフトウェアの基本構成と管理、およびオペレーティング・システム・レベルでのチューニングも、Caching Proxy のパフォーマンスに大きく影響します。パフォーマンスを向上させるために、多くのソフトウェア構成を変更できます。これには、ロギング・ディレクティブ、マッピング・ルール、プラグイン、タイムアウト値、キャッシュ構成値、およびアクティブ・スレッド値の調整が含まれますが、これらに限定されるものではありません。パフォーマンスを向上させるために、多くのオペレーティング・システム構成を変更できます。これには、TCP と ARP のチューニング、ファイル記述子制限の増加、システム・クロックの同期化、ネットワーク・インターフェース・カード (NIC) のチューニング、およびシステム管理タスクを実行するときには有効な一般的方法に従うことが含まれますが、これらに限定されるものではありません。Caching Proxy ソフトウェア構成の詳細については、*Caching Proxy 管理ガイド* を参照してください。

---

### ネットワーク・ハードウェア

このセクションでは、ネットワークに Caching Proxy 機能を導入するときを考慮すべきネットワーク・ハードウェアの問題について説明します。

#### メモリーに関する考慮事項

プロキシ・サーバーは大量のメモリーを占有します。大規模なメモリー専用キャッシュが構成されると、Caching Proxy が 2 GB の仮想アドレス・スペースを消費することがあります。メモリーは、カーネル、共用ライブラリー、およびネットワーク・バッファのためにも必要になります。このため、プロキシ・サーバーが 3 または 4 GB の物理メモリーを消費する可能性があります。メモリー専用キャッシュはロー・ディスク・キャッシュに比べて著しく高速であり、この変更のみがパフォーマンスを向上させると見なすことができることに注意してください。

#### ハード・ディスクに関する考慮事項

Caching Proxy がインストールされているマシンに大きなディスク・スペースを確保することは重要です。これは特にディスク・キャッシュを使用する場合に重要になります。ハード・ディスクからの読み込みと書き込みは、コンピューターを集中的

に使用するプロセスです。Caching Proxy の入出力プロシーダは効率的ですが、Caching Proxy がディスク・キャッシュを使用する構成になっていると、ハード・ディスクの機械的な制限が原因でパフォーマンスが制限される可能性があります。ディスク入出力のボトルネックは、ロー・キャッシュ・デバイスおよびログ・ファイル用に複数のハード・ディスクを使用したり、シーク・タイム、回転速度、転送速度が高速であるディスク・ドライブを使用したりするなどの操作によって軽減できます。

## ネットワークに関する考慮事項

速度、タイプ、および NIC の数などのネットワーク要件、およびプロキシー・サーバーへのネットワーク接続の速度は、Caching Proxy のパフォーマンスに影響を与えます。一般的に、1 つのプロキシー・サーバー・マシンで、着信トラフィック用と発信トラフィック用の 2 つの NIC を使用すると、パフォーマンスが最適化されます。単一の NIC では、HTTP 要求とそれに対する応答トラフィックだけで、限界に達してしまう可能性があります。さらに、NIC は少なくとも 100 MB でなければならず、常に全二重オペレーション用に構成される必要があります。これは、ルーティング装置とスイッチング装置の間の自動折衝によって、エラーが発生したりスループットが犠牲になったりする可能性があるためです。最後に、ネットワーク接続の速度は非常に重要です。例えば、Caching Proxy マシンへの接続が、飽和状態の T1 キャリアである場合、高い要求負荷に対応して最適なスループットを達成することは期待できません。

## CPU に関する考慮事項

Caching Proxy マシンの中央演算処理装置 (CPU) は、パフォーマンスを制限する原因になる可能性があります。CPU の能力は要求を処理する時間に影響を与え、ネットワーク内の CPU の数はスケラビリティに影響を与えます。プロキシー・サーバーの CPU 要件を環境に合わせることで、特に、プロキシー・サーバーが処理するピーク要求負荷をシミュレーションすることは重要です。

---

## ネットワーク・アーキテクチャー

一般に、全体的なパフォーマンスのためには、個々のハードウェアを追加するだけでなく、アーキテクチャーを調整することが得策です。単一のマシンにハードウェアをいくつ追加するとしても、そのハードウェアにはパフォーマンスの最大レベルが存在します。

このセクションでは、ネットワークに Caching Proxy 機能を導入するときに考慮すべきネットワーク・アーキテクチャーの問題について説明します。

## Web サイトの人気とプロキシー・サーバーの負荷に関する考慮事項

企業の Web サイトに人気がある場合、単一のプロキシー・サーバーによって有効に処理できる量を超える要求がコンテンツに対して寄せられて、応答時間が長くなる可能性があります。ネットワーク・パフォーマンスを最適化するには、クラスター化されてロード・バランシングが行われる Caching Proxy マシンを組み込むか、ま

たはネットワーク・アーキテクチャー全体で、リモート・キャッシュ・アクセス (RCA) を使用する共用キャッシュ・アーキテクチャーを使用することを考慮してください。

- **ロード・バランシングが行われるクラスター**

アーキテクチャーを調整する 1 つの方法として、プロキシ・サーバーをクラスター化して Load Balancer コンポーネントを使用し、負荷のバランスを取ることが挙げられます。プロキシ・サーバーのクラスター化は、パフォーマンスとスケーラビリティを向上させるためだけではなく冗長度と信頼性の問題を解決する上でも有益な設計上の配慮です。単一のプロキシ・サーバーは Single Point of Failure (SPOF) になります。つまり、これに障害が発生するかネットワーク障害が原因でアクセス不能になった場合、ユーザーは Web サイトにアクセスできなくなります。

- **共用キャッシュ・アーキテクチャー**

RCA を使用した共用キャッシュ・アーキテクチャーの使用も考慮してください。共用キャッシュ・アーキテクチャーでは、全体の仮想キャッシュが複数の Caching Proxy サーバーに広げられ、それらのサーバーでは、通常、ICP (Internet Cache Protocol) または Cache Array Routing Protocol (CARP) などのキャッシュ間プロトコルが使用されます。RCA は、大規模な仮想キャッシュの提供によって、クラスター化されたキャッシュのヒット率を最大化することを目的に設計されています。

プロキシ・サーバーの RCA 配列を使用すると、単一のスタンドアロン Caching Proxy だけではなくスタンドアロン Caching Proxy マシンのクラスターに比較しても、パフォーマンスが向上します。パフォーマンスは、主として合計仮想キャッシュ・サイズが増加によって向上します。これにより、キャッシュ・ヒット率が最大になり、キャッシュの矛盾と待ち時間が最小になります。RCA を使用すると、特定の文書の 1 つのコピーのみがキャッシュに入れられます。プロキシ・サーバーのクラスターを使用すると、合計キャッシュ・サイズは大きくなりますが、複数のプロキシ・サーバーが同じ情報を取り出したりキャッシュに入れたりする可能性があります。このため、合計キャッシュ・ヒット率は高くなりません。

一般的に、RCA は大規模企業のコンテンツ・ホスティングのシナリオで使用されます。ただし、RCA の有用性は、非常に大規模な企業での展開に限定されるものではありません。ネットワークの負荷がキャッシュ・サーバーのクラスターを必要とする場合、および要求の多くがキャッシュ・ヒットの場合に、RCA の使用を考慮してください。ネットワークの設定によっては、RCA が構成されるとクライアントが使用する TCP 接続の数が増えるため、RCA が常に企業のパフォーマンスを向上させるとは限りません。これは、RCA メンバーが最高スコアの URL の提供を担当するだけでなく、最高スコアではない URL の要求を受けた場合に他のメンバーまたはクラスターに要求を送信する必要もあるからです。これは、RCA 配列の任意のメンバーが、スタンドアロン・サーバーとして作動する場合よりもオープンな TCP 接続を持つ可能性があることを意味します。



## トラフィック・タイプに関する考慮事項

パフォーマンスは、主に Caching Proxy のキャッシング能力によって向上します。ただし、プロキシ・サーバーのキャッシュは、正しく構成されていない場合にボトルネックになる可能性があります。最良のキャッシュ構成を判別するには、トラフィックの特性の分析に力を注がなければなりません。コンテンツのタイプ、サイズ、量、および属性は、起点サーバーから文書を取り出すための時間とサーバーにかかる負荷の点で、プロキシ・サーバーのパフォーマンスに影響を与えます。Caching Proxy がプロキシとなるかキャッシュから提供するトラフィックのタイプを判別すれば、プロキシ・サーバーを構成するときにそれらの特性を考慮に入れることができます。例えば、キャッシュに入れられるオブジェクトの 80% がイメージ (\*.gif または \*.jpg) であり、そのサイズが約 200 KB であることが判別できれば、キャッシング・パラメーターのチューニングとキャッシュのサイズの決定に間違いなく役立ちます。さらに、コンテンツの多くがキャッシングの対象ではないパーソナライズされた動的ページであることの判別も、Caching Proxy のチューニングに直接関連します。

トラフィック特性の分析を行うと、メモリー・キャッシュとディスク・キャッシュのどちらを使用するとキャッシュのパフォーマンスを最適化できるかを判別することができます。また、ネットワークのトラフィック特性に精通すると、Caching Proxy の動的キャッシング機能の使用によってパフォーマンスが向上するかどうかの判別が可能になります。

### • メモリー・キャッシュ対ディスク・キャッシュ

ディスク・キャッシュは、大量の情報をキャッシュに入れるサイトに適しています。例えば、サイト・コンテンツが 5 GB を超えるほど大きく、キャッシュ・ヒット率が 80% から 90% までの場合は、ディスク・キャッシュが推奨されます。ただし、メモリー (RAM) キャッシュを使用すれば処理速度は上がり、メモリー専用キャッシュを使用することが大規模サイトに適しているというケースが多数存在することも明らかです。例えば、Caching Proxy のキャッシュ・ヒット率があまり重要ではない場合か、共用キャッシュ構成が使用されている場合には、メモリー・キャッシュが現実的な選択になります。

### • 動的に生成されたコンテンツのキャッシング

Caching Proxy は、ネットワーク・ベース・キャッシュへの Application Server キャッシュの仮想拡張を提供して、WebSphere Application Server 動的キャッシュによって生成された動的コンテンツ (JSP およびサーブレットの結果) をキャッシュに入れたり無効にしたりできます。動的に生成されたコンテンツのキャッシングを使用可能にすると、アプリケーション論理またはデータベースからのメッセージなどのイベントによって有効期限が切れる、動的に生成された公開 Web ページへの要求が多い環境でのネットワーク・パフォーマンスが向上します。ページの存続時間は有限ですが、有効期限トリガーを作成時に設定することはできません。このため、動的キャッシングおよび無効化機能のないホストは、そのようなページの存続時間値をゼロと指定しなければなりません。

動的に生成されたこのようなページが存続時間中に 1 人以上のユーザーから複数回要求される場合、動的キャッシングによってオフロードが行われ、ネットワークのコンテンツ・ホストに対するワークロードを削減します。動的キャッシングを使用すると、インターネットの横断がより少なくなるためにネットワークの遅

延が除去されて帯域幅の使用量が削減されて、ユーザーへの応答がより高速になることによって、ネットワーク・パフォーマンスも向上します。





---

## 第 6 章 可用性

Application Server Load Balancer コンポーネントは、WebSphere Application Server などのコンテンツ・ホスト、または Application Server Caching Proxy コンポーネントとともに機能して、ネットワークの可用性およびスケーラビリティを向上させます。(これらの Edge components の概要については、3 ページの『第 1 章 WebSphere Application Server Edge components の紹介』を参照してください。) Load Balancer は企業ネットワークで使用され、インターネットと企業のバックエンド・サーバーとの間にインストールされます。大量の要求に応えるためまたは大量のコンテンツを扱うために企業で複数のバックエンド・サーバーを使用している場合にも、Load Balancer はインターネットにおける企業の単一の POP (Point-of-Presence) として働きます。

可用性はロード・バランシングとフェイルオーバー・サポートによって実現されます。

---

### ロード・バランシング

ロード・バランシングは、プロキシ・サーバーとアプリケーション・サーバーを透過的にクラスター化することによって、Web サイトの可用性およびスケーラビリティを向上させます。バックエンド処理能力は透過的に追加することができるため、IT インフラストラクチャーのスケーラビリティがかなり向上されます。

#### 複数のコンテンツ・ホストのロード・バランシング

複数のホストにコンテンツの複製を置いて大量の要求を満たすことはできますが、その場合はそれらホスト間のロード・バランシングを取る方法が必要です。ドメイン・ネーム・サービス (DNS) は基本的なラウンドロビン・ロード・バランシングを提供できますが、これがうまく実行できない場合もあります。

複数のコンテンツ・ホストのロード・バランシングのための高度なソリューションの 1 つとして、24 ページの図 3 に示す Load Balancer の Dispatcher コンポーネントを使用する方法があります。この構成では、すべてのコンテンツ・ホスト (5 の番号の付いたマシン) に同じコンテンツが保管されます。これらのホストはロード・バランシングが行われるクラスターを形成するものとして定義され、Load Balancer マシン (4) のネットワーク・インターフェースのいずれかにそのクラスター専用のホスト名および IP アドレスが割り当てられます。1 の番号の付いたマシンの 1 つを使用しているエンド・ユーザーがファイル X を要求すると、その要求はインターネット (2) を経由し、企業のインターネット・ゲートウェイ (3) を通ってその企業の内部ネットワークに入ります。URL は Dispatcher のホスト名および IP アドレスにマップされているため、Dispatcher は、要求をインターセプトします。Dispatcher は、現在クラスターの中のどのコンテンツ・ホストが要求に応えるために最も適しているかを判断し、要求をそのコンテンツ・ホストに転送します。MAC 転送方式が構成されていると、このコンテンツ・ホストはファイル X を直接クライアントに戻します (つまり、ファイル X は Load Balancer を通りません)。

注: Dispatcher は、以下の 3 つの転送方式を提供します。

- MAC 方式は、サーバーへの着信要求のロード・バランスを取るために使用されます。応答は直接クライアントに戻されます。
- NAT/NAPT 転送方式は、リモートに置かれたサーバーに使用されます。着信要求は Dispatcher によってロード・バランシングが行われ、Dispatcher がサーバーからの応答を受け取り、これをクライアントに戻します。
- Content Based Routing 方式 (cbr) は、Caching Proxy を使用せずに、HTTP および HTTPS 要求の Content Based Routing を提供します。Content Based Routing は、HTTP の場合は「コンテンツ」タイプ・ルールを使用して、および HTTPS の場合は SSL セッション ID 類縁性を使用して実行されます。

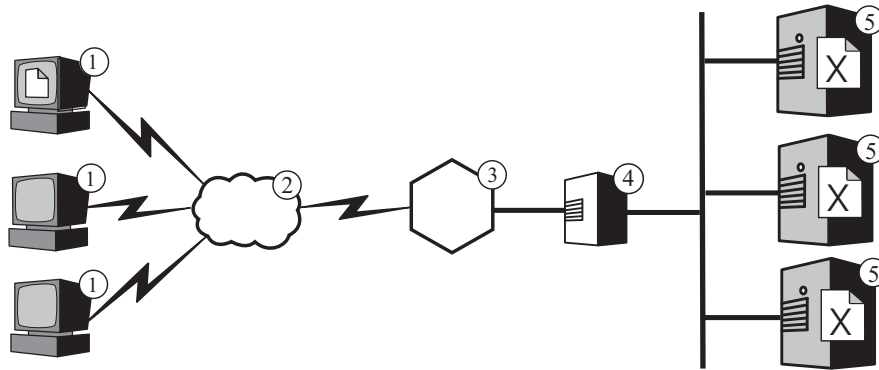


図3. 複数のコンテンツ・ホストのロード・バランシング：(凡例: 1—クライアント 2—インターネット 3—ルーター / ゲートウェイ 4—Dispatcher 5—コンテンツ・ホスト)

デフォルトでは、Dispatcher は DNS のようにラウンドロビン・ロード・バランシングを使用しますが、その場合でも DNS の欠点の多くが解消されます。Dispatcher は、DNS と異なり、コンテンツ・ホストが使用不可またはアクセス不能かどうかを追跡しているため、使用不能なコンテンツ・ホストにクライアントを送り込むことはありません。また、Dispatcher は、新しい接続、活動状態の接続、終了した接続を追跡し、コンテンツ・ホストの現在の負荷を考慮します。Load Balancer の advisor および manager コンポーネントを活動化すると、ロード・バランシングをさらに最適化することができます。これらのオプション・コンポーネントは、コンテンツ・ホストの状況をさらに正確に追跡して、ロード・バランシング決定プロセスに追加情報を取り入れます。manager を使用すると、決定プロセスで使用される各種の要因に異なる重みを割り当てて、ロード・バランシングをサイトに合わせてさらにカスタマイズすることができます。

## 複数のプロキシ・サーバーのロード・バランシング

Load Balancer の Dispatcher は、複数の Caching Proxy マシンに対してロード・バランシングを行うこともできます。企業の Web サイトが有名である場合、単一のプロキシ・サーバーでは効果的に満足させることができないほど大量の要求がサイトのコンテンツに寄せられることがあり、プロキシ・サーバーのパフォーマンスを潜在的に低下させることがあります。

複数の Caching Proxy システムで単一のコンテンツ・ホストのためのプロキシ機能を実行することができますが (13 ページの図 1 に示した構成のように)、複数のプロキシ・サーバーが必要なほどサイトへのアクセスが多い場合は、Load

Balancer によってそのロード・バランスを取る複数のコンテンツ・ホストが必要になります。図4にこの構成が示されています。4の番号の付いた Dispatcher は2つのプロキシ・サーバー(5)のクラスターのロード・バランスを取り、7の番号の付いた Dispatcher は3つのコンテンツ・ホスト(8)のクラスターのロード・バランスを取ります。

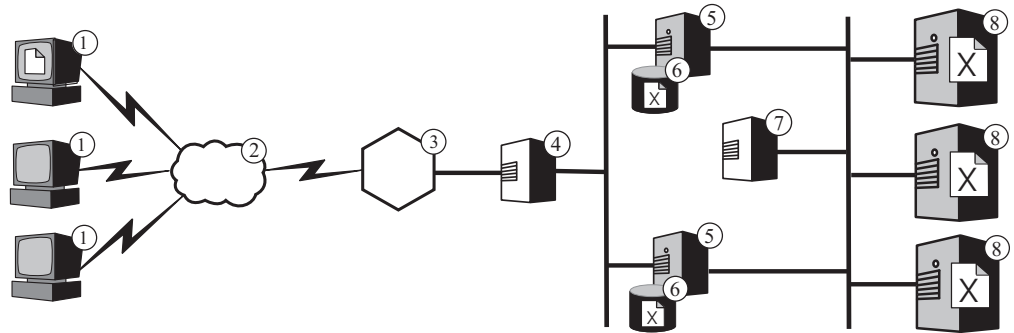


図4. 複数のプロキシ・サーバーおよびコンテンツ・ホストのロード・バランスング：(1—クライアント 2—インターネット 3—ルーター / ゲートウェイ 4—Dispatcher 5—Caching Proxy 6—キャッシュ 7—Dispatcher 8—コンテンツ・ホスト)

4の番号の付いた Dispatcher のクラスター・ホスト名は、企業の Web コンテンツの URL に出現するホスト名です (つまり、インターネットで表示される Web サイトの名前です)。7の番号の付いた Dispatcher のクラスター・ホスト名は、インターネットでは表示されないで、任意の値にすることができます。例えば、ABC Corporation という会社の場合、4の番号の付いた Dispatcher のホスト名には `www.abc.com` が適切ですが、7の番号の付いた Dispatcher には `http-balancer.abc.com` などの名前を付けることができます。

1の番号の付いたクライアント・マシンの1つのブラウザから、8の番号の付いたコンテンツ・サーバーに保管されているファイル X にアクセスする必要があるとします。HTTP 要求はインターネット (2) を経由してゲートウェイ (3) から企業の内部ネットワークに入ります。ルーターは、4の番号の付いた Dispatcher に要求を送り、これはロード・バランスング・アルゴリズムに従って、現在その要求の処理に最も適しているプロキシ・サーバー (5) に要求を渡します。プロキシ・サーバーのキャッシュ (6) にファイル X が入っている場合は、Caching Proxy は4の番号の付いた Dispatcher をバイパスして、ファイルを直接ブラウザに戻します。

プロキシ・サーバーのキャッシュにファイル X が入っていない場合は、Caching Proxy はヘッダーの起点フィールドに自分のホスト名を入れた新しい要求を作成して、7の番号の付いた Dispatcher に送信します。Load Balancer は、現在要求を満たすのに最も適したコンテンツ・ホスト (8) を判断し、そのコンテンツ・ホストに要求を送ります。コンテンツ・ホストはストレージからファイル X を検索し、7の番号の付いた Dispatcher をバイパスして直接プロキシ・サーバーに戻します。プロキシ・サーバーは、適切な場合はファイル X をキャッシュに入れ、4の番号の付いた Dispatcher をバイパスしてブラウザに転送します。

## フェイルオーバー・サポート

Load Balancer は、企業のコンテンツ・ホストの単一の POP (point-of-presence) として機能します。この利点は、各コンテンツ・ホストのホスト名およびアドレスではなく、クラスターのホスト名およびアドレスを DNS に公示することで、企業の Web サイトに、不測の事態に対する一定レベルの保護と統一された外観が備わることです。さらに Web サイトの可用性を高めるには、図 5 に示すように、プライマリー Load Balancer のバックアップとして機能するように別の Load Balancer を構成します。一方の Load Balancer に障害が起こった場合、またはネットワーク障害によりアクセス不能になった場合でも、エンド・ユーザーはコンテンツ・ホストに到達できます。

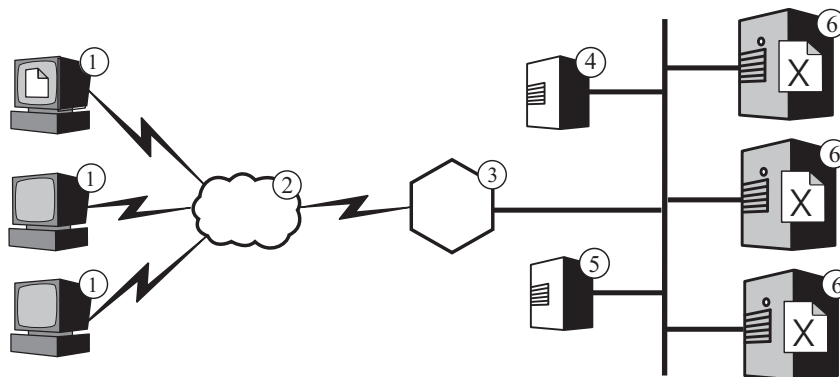


図 5. Web コンテンツを高可用性にするためのプライマリーおよびバックアップ Load Balancer の使用：(凡例: 1—クライアント 2—インターネット 3—ルーター / ゲートウェイ 4—1 次 Dispatcher 5—バックアップ Dispatcher 6—コンテンツ・ホスト)

通常の場合、1 の番号の付いたマシンの 1 つで稼働しているブラウザは、ファイル X が要求されると、1 次 Load Balancer (4) に対応付けられているクラスター・ホスト名にその要求を送ります。Dispatcher は、Dispatcher のロード・バランシング基準を基に選択されたコンテンツ・ホスト (6) へ要求の経路を定めます。コンテンツ・ホストは、企業のゲートウェイ (3) を通じてインターネット (2) 経由でファイル X をブラウザに直接 (Load Balancer をバイパスして) 送信します。

1 次 Dispatcher が作動可能である限り、バックアップ Dispatcher (5) はロード・バランシングを実行しません。1 次 Dispatcher とバックアップ Dispatcher はハートビートと呼ばれるメッセージを定期的に交換して、互いに相手の状況を追跡します。バックアップ Dispatcher は、1 次 Dispatcher の障害を検出すると、1 次 Dispatcher のクラスター・ホスト名と IP アドレスに送られた要求をインターセプトして、ロード・バランシングの任務を自動的に引き継ぎます。

また、2 つの Dispatcher を相互高可用性のために構成することもできます。この場合、2 つの Dispatcher はそれぞれ別々のコンテンツ・ホスト・クラスターのロード・バランシングをアクティブに実行し、同時に相手のバックアップとしても働きます。

Dispatcher は一般に大量の処理リソースまたはメモリー・リソースを消費しないので、Load Balancer マシンで他のアプリケーションを実行することができます。設備費を最小限に抑えることが重要な場合は、ロード・バランシングの対象となるクラ

スター内のマシンの 1 つでバックアップ Dispatcher を実行することもできます。そのような構成を示したのが図 6 です。この場合、バックアップ Dispatcher はクラスター内のコンテンツ・ホストの 1 つ (5) で実行されます。

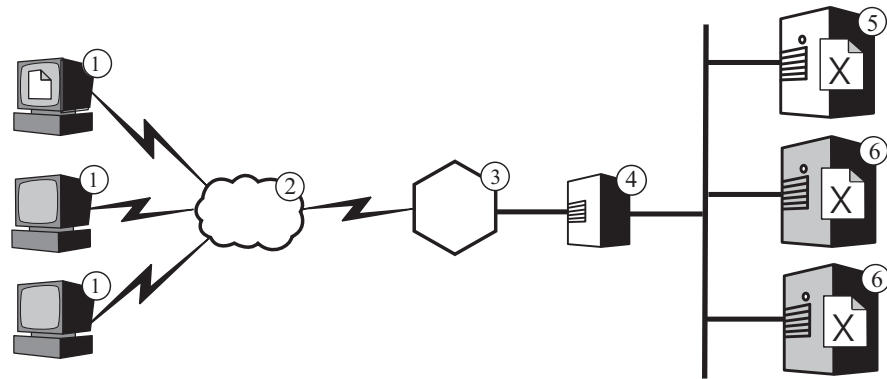


図 6. コンテンツ・ホストへのバックアップ *Load Balancer* の配置：(凡例: 1—クライアント 2—インターネット 3—ルーター / ゲートウェイ 4—1 次 Dispatcher 5—バックアップ Dispatcher とコンテンツ・ホスト 6—コンテンツ・ホスト)



---

## 第 7 章 Content Based Routing

Application Server Load Balancer コンポーネントは Application Server Caching Proxy コンポーネントとともに機能して、さまざまなコンテンツのホストとして機能する複数のバックエンド・サーバーに要求を配布できるようにします。(これらの Edge components の概要については、3 ページの『第 1 章 WebSphere Application Server Edge components の紹介』を参照してください。)

Load Balancer の Content Based Routing (CBR) コンポーネントを Caching Proxy とともにインストールすると、URL または管理者が決定したその他の特性を基に HTTP 要求を分散することができ、同一のコンテンツをすべてのバックエンド・サーバーに保管する必要がなくなります。

**注:** Content Based Routing 方式 (Caching Proxy を使用せずに、HTTP および HTTPS 要求の Content Based Routing を提供する) を使用するように Load Balancer の Dispatcher コンポーネントを構成できます。この場合、Content Based Routing は、HTTP の場合は「コンテンツ」タイプ・ルールを使用して、および HTTPS の場合は SSL セッション ID 類縁性を使用して実行されます。

CBR は、Web サーバーで複数の異なる機能の実行または複数の異なるタイプのサービスの提供が必要な場合にとくに適しています。例えば、オンライン小売業の Web サイトでは、カタログの表示と受注の両方を行う必要があります。カタログの大部分は静的ですが、受注では Common Gateway Interface (CGI) スクリプトなどの対話式アプリケーションを実行して品目番号や顧客情報を受け入れる必要があります。別々の 2 セットのマシンに異なる機能を実行させ、CBR を使用して異なるタイプのトラフィックを別々のマシンに経路指定する方法により効率が改善される場合がよくあります。また、購買要求をより強力な Web サーバーに経路指定して、Web サイトの偶発的な訪問者よりも、購買意欲のある顧客に対するサービスを優先するように、CBR を使用することもできます。

CBR は、ユーザーが作成したルールを基にして要求を経路指定します。最も一般的なタイプは コンテンツ・ルール で、このルールは URL のパス名に基づいて要求を送ります。例えば、ABC Corporation という会社の場合、URL `http://www.abc.com/catalog_index.html` への要求をあるサーバーのクラスターに送り、`http://www.abc.com/orders.html` への要求を別のクラスターに送るように、ルールを作成することができます。また、要求を送信したクライアントの IP アドレスまたはその他の特性に基づいて要求を経路指定するルールもあります。詳しい説明は、*WebSphere Application Server Load Balancer 管理ガイド* の CBR の構成に関する章および Load Balancer と CBR の拡張機能に関する章を参照してください。ルールの構文定義については、*WebSphere Application Server Load Balancer 管理ガイド* の CBR ルール・タイプに関する付録を参照してください。

30 ページの図 7 に示すものは単純な構成の 1 つです。この構成では、Load Balancer の CBR コンポーネントと Caching Proxy が共に 4 の番号の付いたマシンにインストールされ、異なるコンテンツを保管する 3 つのコンテンツ・ホスト (6、7、8) に要求を経路指定します。1 の番号の付いたマシンの 1 つを使用して、あるエンド・ユーザーがファイル X を要求すると、その要求はインターネット (2)



を經由し、企業のインターネット・ゲートウェイ (3) を通ってその企業の内部ネットワークに入ります。プロキシー・サーバーは要求をインターセプトし、同じマシン上の CBR コンポーネントに渡します。CBR は要求の中の URL を解析して、ファイル X がコンテンツ・ホスト 6 にあると判断します。プロキシー・サーバーはファイル X に関する新しい要求を生成し、キャッシング機能が使用可能になっている場合は、ホスト 6 がファイルを戻したときにそのファイルがキャッシュに適格であるかどうかを判断します。ファイルがキャッシュ可能なら、プロキシー・サーバーはファイルをエンド・ユーザーに引き渡す前にキャッシュ (5) にコピーを保管します。他のファイルの経路指定もこれと同様に行われます。ファイル Y に関する要求はコンテンツ・ホスト 7 に送られ、ファイル Z に関する要求はコンテンツ・ホスト 8 に送られます。

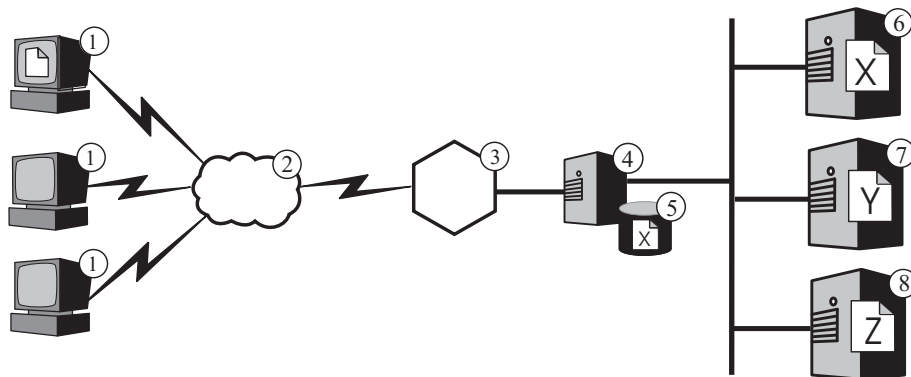


図7. CBR による HTTP 要求の経路指定：(凡例: 1—クライアント 2—インターネット 3—ルーター / ゲートウェイ 4—Caching Proxy と Load Balancer の CBR コンポーネント 5—キャッシュ 6、7、8—コンテンツ・ホスト)

31 ページの図 8 に、オンライン小売業に適していると考えられるより複雑な構成を示します。Load Balancer の CBR コンポーネントとプロキシー・サーバーは、共に 4 の番号の付いたマシンにインストールされ、要求を 2 つの Load Balancer マシンに経路指定します。6 の番号が付いた Load Balancer は、主として小売業者のカタログの静的コンテンツを保管するコンテンツ・ホスト (8) のクラスターのロード・バランシングを取り、7 の番号の付いた Load Balancer は、受注を処理する Web サーバー (9) のクラスターのロード・バランスを取ります。

1 の番号の付いたマシンの 1 つを使用しているエンド・ユーザーが小売業者のカタログの URL にアクセスすると、その要求はインターネット (2) を經由し、企業のインターネット・ゲートウェイ (3) を通ってその企業の内部ネットワークに入ります。プロキシー・サーバーは要求をインターセプトし、同じマシン上の CBR コンポーネントに渡します。CBR は URL を解析して、6 の番号の付いた Load Balancer が URL を処理すると判断します。プロキシー・サーバーは新しいアクセス要求を作成して Load Balancer に送信します。Dispatcher は (ユーザー定義の基準を基にして)、8 の番号の付いたコンテンツ・ホストのうち、どのホストが現在最もよく要求にサービスを提供できるかを判断します。このコンテンツ・ホストは Load Balancer をバイパスして、カタログ・コンテンツをプロキシー・サーバーに直接引き渡します。前の例の場合と同様、プロキシー・サーバーは、コンテンツがキャッシュ保管の対象であるかどうかを判断し、該当する場合はそのコンテンツをキャッシュ (5) に保管します。



通常はカタログのハイパーリンクを經由して小売業者の注文 URL にアクセスすることによって、エンド・ユーザーは発注します。この要求がたどるパスは、マシン 4 の CBR コンポーネントにより 7 の番号の付いた Load Balancer マシンに送られる点を除けば、カタログ・アクセスの要求の場合と同じです。Load Balancer は 9 の番号の付いた Web サーバーのうち最も適したものに要求を転送し、その Web サーバーはプロキシ・サーバーに直接応答します。通常、注文情報は動的に生成されるので、プロキシ・サーバーが注文情報をキャッシュに入れることはありません。

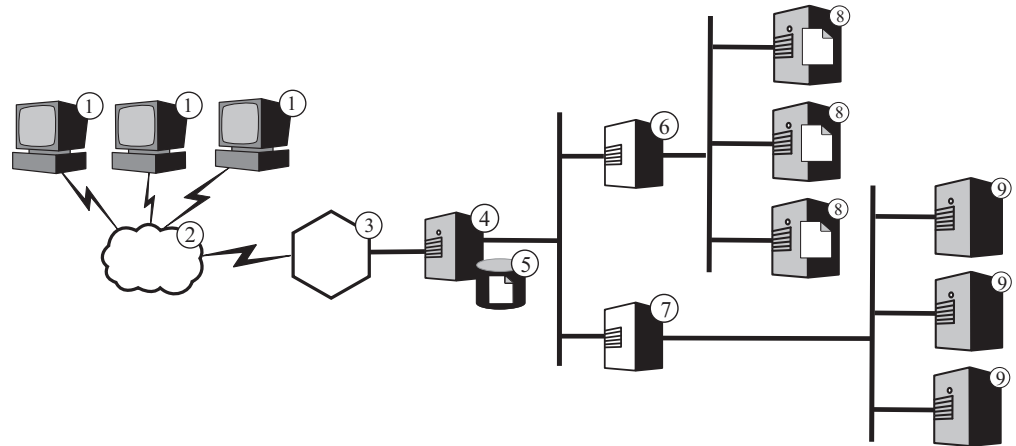


図8. CBR により経路指定された HTTP 要求のロード・バランシング：(凡例: 1—クライアント 2—インターネット 3—ルーター / ゲートウェイ 4—Caching Proxy と Load Balancer の CBR コンポーネント 5—キャッシュ 6, 7—Load Balancer 8—コンテンツ・ホスト 9—Web サーバー)

Load Balancer の CBR 機能は *cookie* 類縁性をサポートします。これは、エンド・ユーザーの最初の要求にサービスを提供するサーバーの識別が、サーバーの応答に含まれる特殊なデータ・パケット (*cookie*) に記録されるということです。定義した期間内にエンド・ユーザーが同じ URL に再びアクセスすると、要求に *cookie* が含まれている場合 CBR は標準ルールを適用する代わりに要求を元のサーバーに経路指定します。一般に、サーバーが再び取得する必要のないエンド・ユーザー情報 (クレジット・カード番号など) がサーバーに保管されている場合、*cookie* により応答時間が改善されます。



---

## 第 3 部 シナリオ

この部では、IBM WebSphere Application Server Edge components を使用するビジネス・シナリオについて説明します。これらは、アーキテクチャーについて安全なテスト済みのソリューションであり、優れたパフォーマンス、可用性、スケーラビリティ、および信頼性を提供するものです。

この部は、以下の章で構成されています。

35 ページの『第 8 章 企業消費者間ネットワーク』

41 ページの『第 9 章 企業クライアント間バンキング・ソリューション』

43 ページの『第 10 章 Web ポータル・ネットワーク』



---

## 第 8 章 企業消費者間ネットワーク

基本的なエレクトロニック・コマース Web サイトは、企業消費者間ネットワークです。一般的に、企業がインターネットに進出する最初のフェーズでは、Web 上にサイトを作ることにのみに焦点が置かれます。企業情報と商品カタログがデジタル形式に変換され、Web サイトで表示可能になります。ショッピングは電子メール・アドレス、電話番号、および FAX 番号の提供によって可能になり、さらに、自動化されたフォームによっても可能になります。しかし、本当のオンライン・ショッピングを行うことはできません。注文の処理に人的な作業を必要とするため、すべてのトランザクションに必ず待ち時間が伴います。

2 番目のフェーズでは、企業は直接オンライン購入用のセキュア・ショッピング・カートをインプリメントすることによって、この待ち時間を取り除き、販売操作を能率的にします。これらの販売トランザクションを完全なものにするには、ウェアハウス・データベースとの同期化を図り、バンキング・システムとの統合が非常に重要になります。在庫のない商品は販売できず、その品目について顧客の会計に課金することはできません。同様に、有効な会計上のトランザクションが発生するまで、在庫の商品を取り出して顧客に配送することはできません。

3 番目のフェーズでは、企業の Web サイトはさらに進化し、顧客はクライアントへと発展し、各クライアントにはそれぞれ固有のコンテンツが示される、動的なプレゼンテーション・サイトへと発展します。

---

### フェーズ 1

36 ページの図 9 は、効率のよいカタログのブラウズを目的に設計された小さな商用 Web サイトを示しています。すべてのクライアント要求はファイアウォールを経由して Dispatcher へとパススルーして、Web サーバーの代理サーバーとして働く、アクティブ・キャッシュを使用するプロキシー・サーバーのクラスターに経路指定されます。Metric Server がプロキシー・サーバーと同じ場所に配置されて、Dispatcher にロード・バランシング・データを提供します。この配置は、Web サーバーにかかるネットワーク負荷を軽減し、インターネットとの直接的接触から Web サーバーを切り離します。

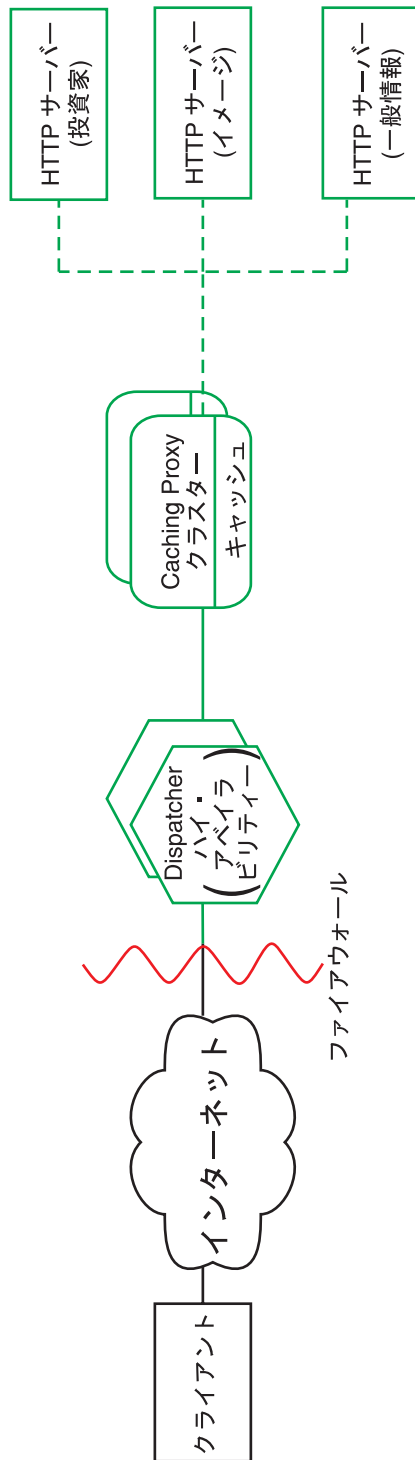


図9. 企業消費者間ネットワーク (フェーズ 1)

---

## フェーズ 2

38 ページの図 10 は、商用 Web サイトの発展の 2 番目のフェーズを示しています。このサイトは、効率的なカタログのブラウズおよび潜在的な顧客に高速なセキュア・ショッピング・カートを提供することを目的に設計されています。すべての顧客の要求は、インターネット・プロトコルに基づいて要求を分離する Dispatcher によって、ネットワーク内の該当するブランチに経路指定されます。HTTP 要求は静的 Web サイトに移動します。HTTPS 要求はショッピング・ネットワークに移動します。基本の静的 Web サイトはこのフェーズでも、Web サーバーの代理として機能するアクティブ・キャッシュを持つプロキシ・サーバーのクラスターによって処理されます。ネットワークのこの部分は、最初のフェーズのネットワークを反映しています。

Web サイトのエレクトロニック・コマースの部分も、プロキシ・サーバーのクラスターによって処理されます。ただし、Caching Proxy ノードはいくつかのプラグイン・モジュールによって拡張されています。SSL ハンドシェイクは暗号ハードウェア・カードにオフロードされ、認証は Access Manager (旧 Policy Director) プラグインを通じて実行されます。動的キャッシング・プラグインは、共通データを保管して WebSphere アプリケーション・サーバー上のワークロードを削減します。アプリケーション・サーバー上のプラグインは、必要であれば動的キャッシングのオブジェクトを無効にします。

すべてのショッピング・カート・アプリケーションはユーザーの認証に使用された顧客データベースに結合されています。これにより、ユーザーが個人情報を認証に 1 回、ショッピングに 1 回の計 2 回システムに入力しないで済むようになります。

このネットワークは、クライアントの使用法に応じてトラフィックを分割し、基本 Web サイトからプロセッサ集中の SSL 認証およびエレクトロニック・コマース・ショッピング・カートを切り離します。この二重トラック Web サイトにより、ネットワーク内のサーバーの役割に基づいて優れたパフォーマンスを提供するようにさまざまなサーバーを調整できます。

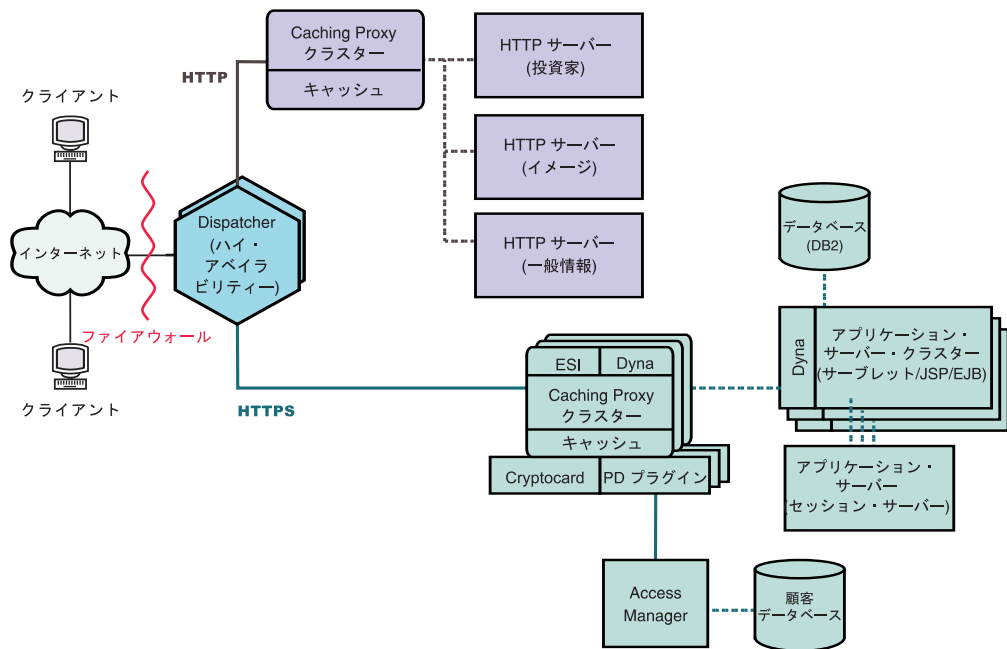


図 10. 企業消費者間ネットワーク (フェーズ 2)

## フェーズ 3

39 ページの図 11 は、企業消費者間ネットワークの発展の 3 番目のフェーズを示しています。ここでは、静的 Web に動的プレゼンテーション・メソッドが取り入れられています。プロキシ・サーバー・クラスターが拡張されて、動的 Web コンテンツのキャッシングと、Edge Side Includes (ESI) プロトコルに準拠して作成されたページ・フラグメントのアセンブリーをサポートしています。サーバー側インクルード機構を使用して、コンテンツ・サーバー上に Web ページを作成してから、これらのクライアント固有の、キャッシュ不可能なページをネットワーク全体に伝搬するのとは違い、ESI 機構はキャッシュに入れられたコンテンツからネットワークのエッジでページをアセンブルすることを許可し、これによって帯域幅使用量と応答時間を削減します。

この 3 番目のフェーズのシナリオでは ESI 機構が重要になります。このシナリオでは、パーソナライズされたホーム・ページを各クライアントが Web サイトから受信します。これらのページのビルディング・ブロックは、一連の WebSphere Application Servers から検索されます。機密ビジネス・ロジックと、セキュア・データベースへのタイを含むアプリケーション・サーバーは、ファイアウォールの後ろで分離されています。



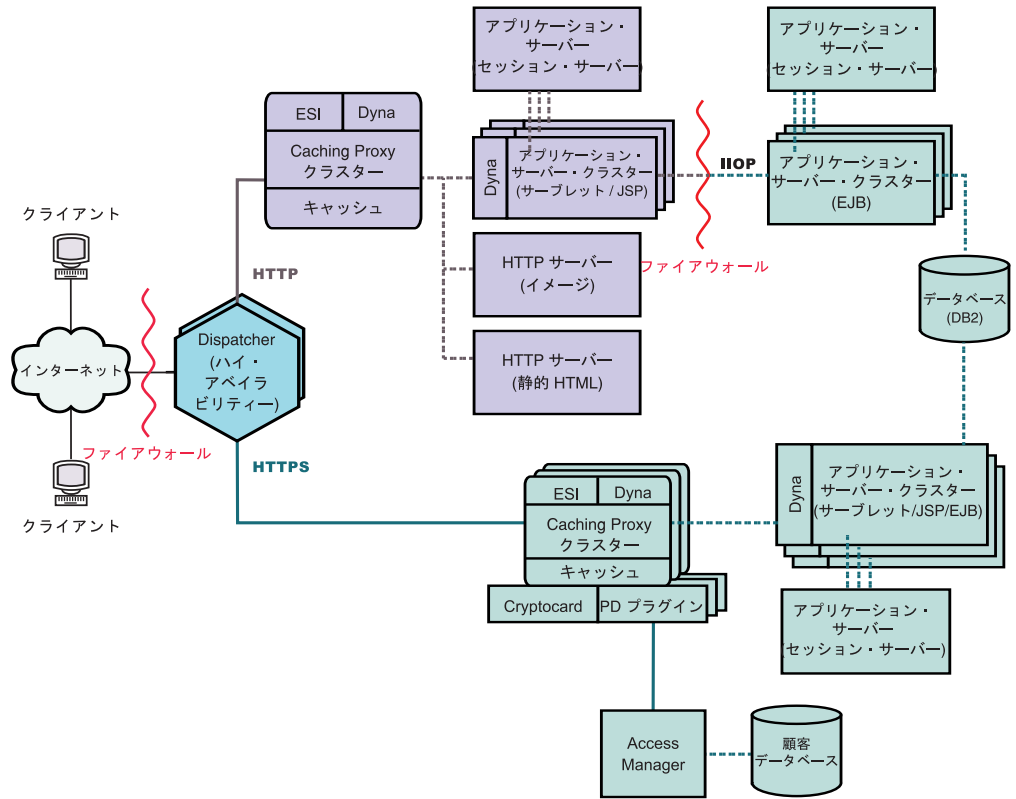


図 11. 企業消費者間ネットワーク (フェーズ 3)



---

## 第 9 章 企業クライアント間バンキング・ソリューション

42 ページの図 12 は、35 ページの『第 8 章 企業消費者間ネットワーク』に説明されている企業消費者間ネットワークに類似した、効率的なオンライン・バンキング・ソリューションを示しています。すべてのクライアント要求はファイアウォールを抜けて Dispatcher へとパススルーし、Dispatcher はトラフィックをインターネット・プロトコルに基づいて分離します。HTTP 要求は、Web サーバーの代理サーバーとして働く、アクティブ・キャッシュを使用するプロキシ・サーバーのクラスターに移動します。Metric Server はプロキシ・サーバーと同じ場所に配置されて、Dispatcher にロード・バランシング・データを提供します。この配置により、Web サーバーにかかるネットワーク負荷が軽減され、Web サーバーとインターネットの間に追加のバッファが作成されます。

HTTPS 要求は、クライアントに個人会計情報を提供してオンライン・バンキング・トランザクションを許可するように設計されたセキュア・ネットワークに渡されます。拡張されたプロキシ・サーバーのクラスターにより、サイトにスケーラビリティが与えられます。これらのプロキシ・サーバーは、動的 Web コンテンツのキャッシングと、Edge Side Includes (ESI) プロトコルに準拠して作成されたページ・フラグメントのアセンブリーをサポートしています。暗号ハードウェア・カードが SSL ハンドシェイクを管理しているため、プロキシ・サーバー・ホストで行う必要のある処理が著しく削減されます。また、Access Manager (旧 Policy Director) がクライアント認証を指示します。

アプリケーション・サーバー・クラスターのコレクションは、EJB コンポーネントに含まれるビジネス・ロジックを、サーブレットと JSP ファイルに含まれるプレゼンテーション層から分離することによって、要求の処理を配布します。これらのクラスターのそれぞれが、個別のセッション・サーバーによって管理されます。

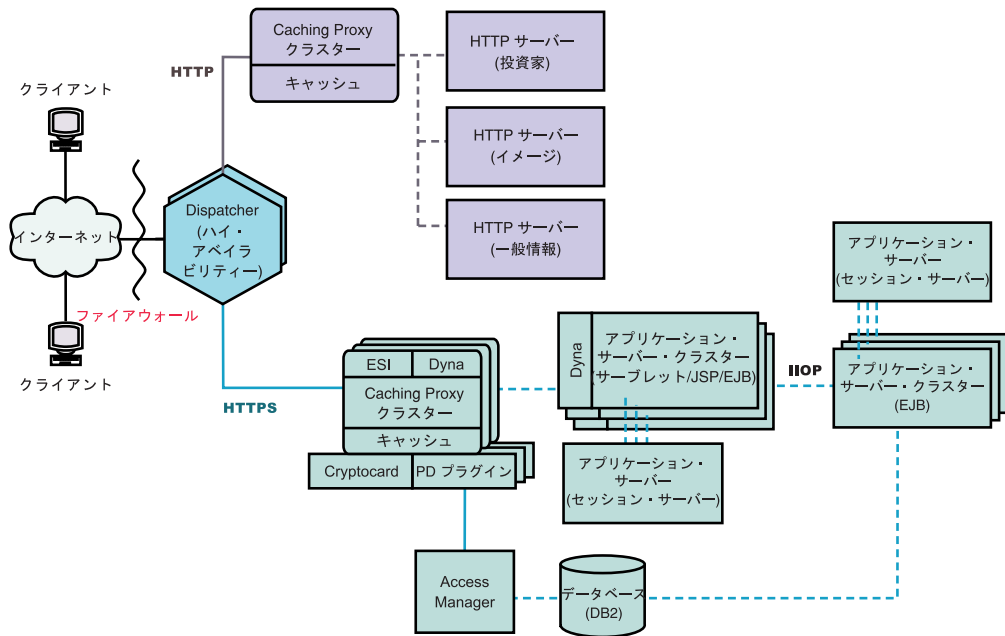


図 12. 企業消費者間バンキング・ソリューション

---

## 第 10 章 Web ポータル・ネットワーク

44 ページの図 13 は、各クライアントにパーソナライズされたコンテンツを提供するときの大量のトラフィックをサポートすることを目的に設計された Web ポータル・ネットワークを示しています。さまざまなサーバーにかかる処理負荷を最小化するため、ネットワークのどの部分も SSL トラフィックを運びません。ポータルは機密データを送達しないため、セキュリティは重要な問題ではありません。クライアント ID、パスワード、および設定を含むデータベースが適度に安全で破壊されていないことは重要ですが、この要件は Web サイトの残りの部分のパフォーマンスを損なうものではありません。

すべてのクライアント要求はファイアウォールを介して Dispatcher へとパススルーします。Dispatcher は、Web サーバーの代理サーバーとして働く、アクティブ・キャッシュを使用するプロキシ・サーバーのクラスター内でネットワーク負荷のバランシングを行います。Metric Server はプロキシ・サーバーと同じ場所に配置されて、Dispatcher にロード・バランシング・データを提供します。

実際の動的 Web サイトは、プロキシ・サーバーにアセンブリーのために渡される ESI フラグメントを生成するアプリケーション・サーバーのクラスターです。セキュリティがそれほど考慮されていないため、Web サイトを構成するために必要なすべての機能を各アプリケーション・サーバーが実行します。すべてのアプリケーション・サーバーは同一です。1 つのアプリケーション・サーバーのサービスが休止されると、セッション・サーバーが要求を他のサーバーに経路指定して、サイト全体のハイ・アベイラビリティを保ちます。またこの構成では、ポータルが特殊イベントのホストになるなどして過度のトラフィックが発生した場合に、Web サイトの急速なエスカレーションも可能になります。追加のプロキシ・サーバーとアプリケーション・サーバーを、素早くサイトに構成できます。

イメージ・ファイルおよび定形文面テキストなどのすべての静的コンテンツは、個別の Web サーバーに保管されます。このため、より複雑なアプリケーション・サーバーを破壊してしまうようなリスクを負わずに、必要に応じて更新を行うことができます。

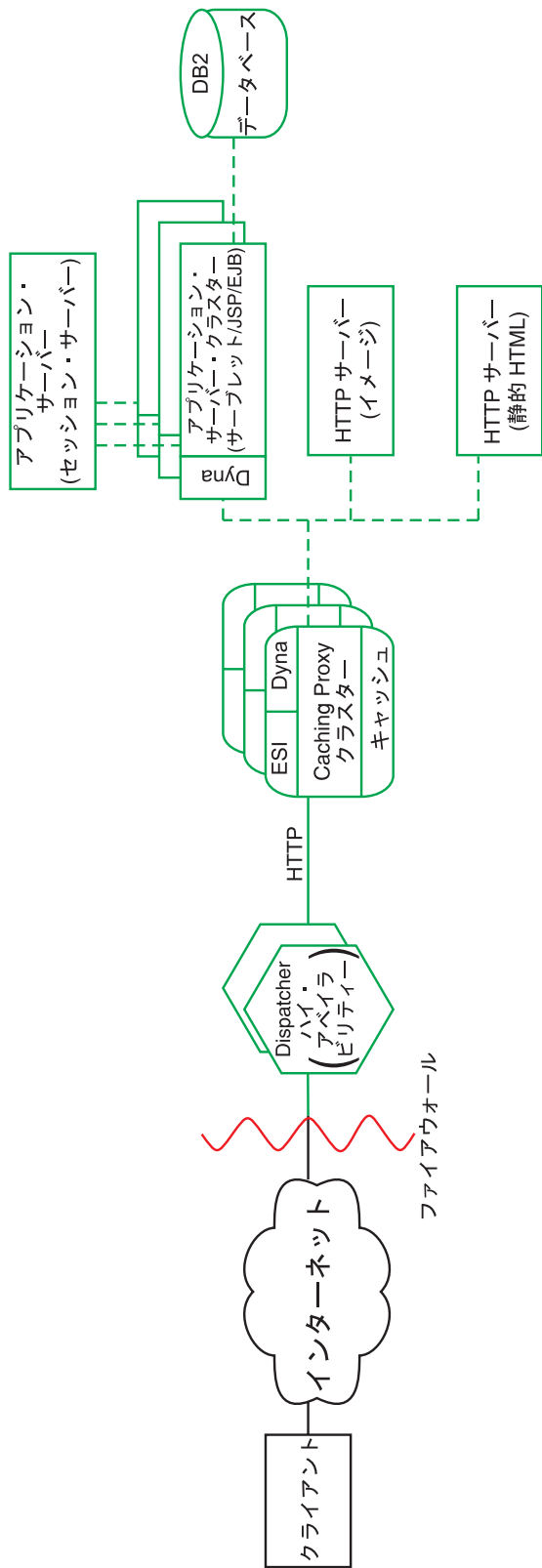


図 13. Web ポータル

---

## 第 4 部 Edge components のインストール

この部では、Edge components のハードウェアおよびソフトウェア要件と、それらをインストールする手順を説明します。

この部は、以下の章で構成されています。

47 ページの『第 11 章 Edge components の要件』

59 ページの『第 12 章 セットアップ・プログラムを使用した Edge components のインストール』

63 ページの『第 13 章 システム・パッケージ・ツールを使用した Caching Proxy のインストール』

67 ページの『第 14 章 システム・パッケージ・ツールを使用した Load Balancer のインストール』





---

## 第 11 章 Edge components の要件

この章では、Edge components のハードウェアとソフトウェアの要件、また Caching Proxy 構成および管理フォームおよび Load Balancer オンライン・ヘルプで Web ブラウザーを使用するためのガイドラインを記載します。

重要: ハードウェアおよびソフトウェアの要件に関する最新の情報については、<http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html> を参照してください。

---

### ハードウェアおよびソフトウェアの前提条件

このセクションは、WebSphere Application Server、バージョン 6.0 Edge components のハードウェアおよびソフトウェアの前提条件について説明します。

#### AIX システムでの Caching Proxy のインストール前提条件

このセクションでは、AIX<sup>®</sup> オペレーティング・システムが稼動しているマシンに Caching Proxy をインストールする場合のハードウェアおよびソフトウェアの前提条件について説明します。

- **ハードウェア・アーキテクチャー:** IBM RS/6000<sup>®</sup>。
- **AIX のバージョン:** AIX 5.1 (保守レベル 5100-05)、AIX 5.2 (保守レベル 5200-02 または 5200-03)、または AIX 5.3。 32 ビット・モードまたは 64 ビット・モードがサポートされます。
- **ソフトウェアおよび文書のための空きディスク・スペース:** 50 MB にログ用のスペースを加えた容量。
- **通信ハードウェア・アダプター:** TCP/IP を使用してネットワーク接続を行うように構成する必要があります。
- **RAM:** 64 MB 以上。ただし、容量が大きければ大きいほどパフォーマンスが向上します。
- **ディスク・キャッシング用の空きディスク・スペース:** ユーザーが Web サーバーから検索するファイルのサイズと数に依存します。キャッシュが大きいほど、一般にキャッシュ・アクセス率が高くなります。推奨最小値は、メモリー・キャッシングの場合と同じです。
- **ページング用空きディスク・スペース:** RAM 容量の 2 倍以上。
- **LC\_FASTMSG 環境変数:** Caching Proxy を英語以外の言語環境で実行する場合、`/etc/environment` ファイルを編集して、`LC_FASTMSG` 環境変数の値を `false` に設定します。(名前の文字列に、2 本の下線があることに注意してください。)
- **iKeyman:** これは Secure Sockets Layer (SSL) が使用する暗号鍵の操作作用として Caching Proxy に含まれているユーティリティーです。このユーティリティーは、32 ビットの Java<sup>™</sup> ランタイム環境 (JRE) バージョン 1.4 で使用する必要があります。
- 動的キャッシングは、WebSphere Application Server バージョン 4、5、および 6 を使用する場合にのみサポートされます。

- オプションとして、暗号ハードウェア・カード Rainbow CryptoSwift Accelerator を使用できます。

## AIX システムでの Load Balancer のインストール前提条件

このセクションでは、AIX オペレーティング・システムが稼働しているマシンに Load Balancer コンポーネントをインストールする場合のハードウェアとソフトウェアの前提条件について説明します。

- **ハードウェア・アーキテクチャー:** IBM RS/6000
- **AIX のバージョン:** AIX 5.1 (保守レベル 5100-05)、AIX 5.2 (保守レベル 5200-02 または 5200-03)、または AIX 5.3。 32 ビット・モードまたは 64 ビット・モードがサポートされます。
- **ソフトウェアおよび文書のための空きディスク・スペース:** 50 MB にログ用のスペースを加えた容量。
- 以下のいずれかのネットワーク・インターフェース・カード (NIC):
  - 16 Mb トークンリング。
  - 10 Mb イーサネット。
  - 100 Mb イーサネット。
  - 1 Gb イーサネット。
  - 光ファイバー分散データ・インターフェース (FDDI)。
  - マルチポート・イーサネット NIC。

**注:** マルチポート NIC のインプリメンテーションは、ベンダーごとに異なります。したがって、一部のマルチポート NIC のサポートが制限される場合があります。

- **32 ビットの Java 2 ランタイム環境:** このソフトウェアは、IBM AIX Developer Kit, Java 2 Technology Edition, 1.4.2 で使用可能です。
- **CBR コンポーネントと Web ベース管理**の両方が、インストールおよび構成された Caching Proxy を必要とします。
- **Web ベース管理**は Perl v5.5 を必要とします。
- **Consultant for Cisco CSS Switch** は、Cisco CSS 11000 シリーズ・スイッチをインストールして構成する必要があります。
- **Nortel Alteon Controller** には、インストールされ、構成済みの Alteon AD3、AD4、180e、184、または Passport 8600 用の layer 4/7 blade が必要です。 Web OS バージョン 9 または 10 は、Nortel Alteon Web Switch ファミリーでサポートされるソフトウェアです。

## HP-UX システムでの Caching Proxy のインストール前提条件

このセクションでは、HP-UX オペレーティング・システムが稼働しているマシンに Caching Proxy をインストールする場合のハードウェアおよびソフトウェアの前提条件について説明します。

- **ハードウェア・アーキテクチャー:** PA-RISC 1.1 および PA-RISC 2.0 アーキテクチャー・マシン
- **HP-UX のバージョン:** HP-UX 11iv1 (32 ビット・モードまたは 64 ビット・モード)。

フィックスパック (HP-UX 11i Quality Pack (GOLDQPK11i)) の最新バージョンが必要です。最新の Quality Pack の入手に関する詳細およびダウンロード方法については、HP の Support Plus Web サイト ([http://www.software.hp.com/SUPPORT\\_PLUS/qpk.html](http://www.software.hp.com/SUPPORT_PLUS/qpk.html)) を参照してください。

- **ソフトウェアおよび文書のための空きディスク・スペース:** 300 MB にログ用のスペースを加えた容量。
- **通信ハードウェア・アダプター:** TCP/IP スタックを使用してネットワーク接続を行うように構成する必要があります。
- **RAM:** 64 MB 以上。ただし、容量が大きければ大きいほどパフォーマンスが向上します。
- **ディスク・キャッシング用の空きディスク・スペース:** ユーザーが Web サーバーから検索するファイルのサイズと数に依存します。キャッシュが大きいほど、一般にキャッシュ・アクセス率が高くなります。推奨最小値は、メモリー・キャッシングの場合と同じです。
- **ページング用空きディスク・スペース:** RAM 容量の 2 倍以上。
- **iKeyman:** これは Secure Sockets Layer (SSL) が使用する暗号鍵の操作用として Caching Proxy に含まれているユーティリティーです。このユーティリティーは、32 ビットの Java ランタイム環境 (JRE) バージョン 1.4 で使用する必要があります。
- 動的キャッシングは、WebSphere Application Server バージョン 4、5、および 6 を使用する場合にのみサポートされます。
- オプションとして、暗号ハードウェア・カード Rainbow CryptoSwift Accelerator を使用できます。

## HP-UX システムでの Load Balancer のインストール前提条件

このセクションでは、HP-UX オペレーティング・システムが稼働しているマシンに Load Balancer コンポーネントをインストールする場合のハードウェアおよびソフトウェアの前提条件について説明します。

- **ハードウェア・アーキテクチャー:** PA-RISC 1.1 および PA-RISC 2.0 アーキテクチャー・マシン
- **HP-UX のバージョン:** HP-UX 11iv1 (32 ビット・モードまたは 64 ビット・モード)。

フィックスパック (HP-UX 11i Quality Pack (GOLDQPK11i)) の最新バージョンが必要です。最新の Quality Pack の入手に関する詳細およびダウンロード方法については、HP の Support Plus Web サイト ([http://www.software.hp.com/SUPPORT\\_PLUS/qpk.html](http://www.software.hp.com/SUPPORT_PLUS/qpk.html)) を参照してください。

- **ソフトウェアおよび文書のための空きディスク・スペース:** 50 MB にログ用のスペースを加えた容量。
- 以下のいずれかのネットワーク・インターフェース・カード (NIC):
  - 10 Mb イーサネット。
  - 100 Mb イーサネット。
  - 1 Gb イーサネット。

- マルチポート・イーサネット NIC。モード 1 だけがサポートされます。モード 2 (フォールト・トレランス) およびモード 3 (ポート集約) はサポートされません。

**注:** マルチポート NIC のインプリメンテーションは、ベンダーにより異なります。したがって、一部のマルチポート NIC のサポートが制限される場合があります。

- 32 ビットの Java 2 ランタイム環境、標準版、バージョン 1.4.2。
- **CBR コンポーネント**と **Web ベース管理**の両方が、インストールおよび構成された Caching Proxy を必要とします。
- **Web ベース管理**は Perl v5.5 を必要とします。
- **Consultant for Cisco CSS Switch** は、Cisco CSS 11000 シリーズ・スイッチをインストールして構成する必要があります。
- **Nortel Alteon Controller** には、インストールされ、構成済みの Alteon AD3、AD4、180e、184、または Passport 8600 用の layer 4/7 blade が必要です。Web OS バージョン 9 または 10 は、Nortel Alteon Web Switch ファミリーでサポートされるソフトウェアです。

## Linux システムでの Caching Proxy のインストール前提条件

このセクションでは、Linux オペレーティング・システムが稼働しているマシンに Caching Proxy をインストールする場合のハードウェアおよびソフトウェアの前提条件について説明します。

- **ハードウェア・アーキテクチャーとソフトウェア・バージョン**

以下のテーブルは、Linux をサポートするシステムをリストしています。ハードウェアおよびソフトウェアの前提条件に関する更新および追加情報については、<http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html> を参照してください。

表 2. サポートされる Linux システム

オペレーティング・システム	Linux for Intel (32 ビット・モード)	Linux for S/390 zSeries	Linux for iSeries または pSeries
Red Hat Enterprise Linux v3 Update 2 (2.4 カーネル)	x	x	x
SuSE Linux Enterprise Server 8.0 SP3 (2.4 カーネル)	x	x	x
SuSE Linux Enterprise Server 9.0 (2.4 カーネル)	x	x	x
UnitedLinux 1.0 SP3	x	x	x

**注:** Red Hat Enterprise Linux v3 for IA-32 プラットフォームでは、以下のパッケージをインストールする必要があります。

```
compat-gcc-7.3-2.96*
compat-libstdc++-7.3-2.96*
compat-libstdc++-devel-7.3-2.96*
compat-glibc-7.x-2.2.4.32*
compat-gcc-c++-7.3-2.96*
compat-db-4.0.14*
```

Red Hat Enterprise Linux v3 for S/390 (z/VM および VM/ESA) プラットフォームでは、以下のパッケージをインストールする必要があります。

```
compat-db-4.0.14*
compat-pwdb-0.62*
compat-libstdc++-7.2-2.95*
```

これらの rpm がないと、Caching Proxy が始動せず、IBM Runtime Environment for Linux、Java 2 Technology Edition が正常に実行されません。

- **ソフトウェアおよび文書のための空きディスク・スペース:** 50 MB にログ用のスペースを加えた容量。
- **通信ハードウェア・アダプター:** TCP/IP スタックを使用してネットワーク接続を行うように構成する必要があります。
- **RAM:** 64 MB 以上。ただし、容量が大きければ大きいほどパフォーマンスが向上します。
- **ディスク・キャッシング用の空きディスク・スペース:** ユーザーが Web サーバーから検索するファイルのサイズと数に依存します。キャッシュが大きいほど、一般にキャッシュ・アクセス率が高くなります。推奨最小値は、メモリー・キャッシングの場合と同じです。
- **ページング用空きディスク・スペース:** RAM 容量の 2 倍以上。
- **iKeyman:** これは Secure Sockets Layer (SSL) が使用する暗号鍵の操作作用として Caching Proxy に含まれているユーティリティーです。このユーティリティーは、32 ビットの Java ランタイム環境 (JRE) バージョン 1.4 で使用する必要があります。
- 動的キャッシングは、WebSphere Application Server バージョン 4、5、および 6 を使用する場合にのみサポートされます。
- オプションとして、暗号ハードウェア・カード Rainbow CryptoSwift Accelerator を使用できます。

## Linux システムでの Load Balancer のインストール前提条件

このセクションでは、Linux オペレーティング・システムが稼働しているマシンに Load Balancer コンポーネントをインストールする場合のハードウェアおよびソフトウェアの前提条件について説明します。

- **ハードウェア・アーキテクチャーとソフトウェア・バージョン**

以下のテーブルは、Linux をサポートするシステムをリストしています。ハードウェアおよびソフトウェアの前提条件に関する更新および追加情報については、<http://www.ibm.com/software/webservers/appserv/doc/latest/prereq.html> を参照してください。

表 3. サポートされる Linux システム

オペレーティング・システム	Linux for Intel (32 ビット・モード)	Linux for S/390 zSeries	Linux for iSeries または pSeries
Red Hat Enterprise Linux v3 Update 2 (2.4 カーネル)	x	x	x
SuSE Linux Enterprise Server 8.0 SP3 (2.4 カーネル)	x	x	x
SuSE Linux Enterprise Server 9.0 (2.4 カーネル)	x	x	x
UnitedLinux 1.0 SP3	x	x	x

- ソフトウェアおよび文書のための空きディスク・スペース: 50 MB にログ用のスペースを加えた容量。
- 以下のいずれかのネットワーク・インターフェース・カード (NIC):
  - 10 Mb イーサネット。
  - 100 Mb イーサネット。
  - 1 Gb イーサネット。
  - マルチポート・イーサネット NIC。モード 1 だけがサポートされます。モード 2 (フォールト・トレランス) およびモード 3 (ポート集約) はサポートされません。

**注:** マルチポート NIC のインプリメンテーションは、ベンダーにより異なります。したがって、一部のマルチポート NIC のサポートが制限される場合があります。

- bash シェル (Bourne シェルの GNU バージョン) のバージョンをインストールする必要があります。これは、Red Hat および SuSE の両方の Linux システムで配布されるデフォルトのシェルです。
- **32 ビットの Java 2 ランタイム環境:** このソフトウェアは、IBM Runtime Environment for Linux, Java 2 Technology Edition, バージョン 1.4.2 で使用可能です。
- JAVA\_HOME および PATH 環境変数は、Java 言語サポート・ソフトウェアをインストールしてある場所へのパスを参照するように設定する必要があります。**export** コマンドを使用してください。設定する値の例を次に示します。JAVA\_HOME の例にある Java2-14/jre 句の代わりにユーザーのパスを使用します。
 

```
export JAVA_HOME=/opt/IBMJava2-14/jre
export PATH=$JAVA_HOME/bin:$PATH
```
- 高可用性およびコロケーションと合わせて Dispatcher コンポーネントの MAC 転送方式を使用する場合、Linux カーネルのパッチが必要です。パッチをダウンロードしてインストールする方法については、*Load Balancer 管理ガイド* を参照してください。
- **CBR コンポーネントと Web ベース管理**の両方が、インストールおよび構成された Caching Proxy を必要とします。



- **Web ベース管理**は Perl v5.5 を必要とします。
- **Consultant for Cisco CSS Switch** は、Cisco CSS 11000 シリーズ・スイッチをインストールして構成する必要があります。
- **Nortel Alteon Controller** には、インストールされ、構成済みの Alteon AD3、AD4、180e、184、または Passport 8600 用の layer 4/7 blade が必要です。Web OS バージョン 9 または 10 は、Nortel Alteon Web Switch ファミリーでサポートされるソフトウェアです。

**注:** どの Linux 環境でも、Appletalk を使用した場合、Load Balancer は正常に機能しません。executor **start** コマンドを実行すると、Load Balancer は処理を中断します。

## Solaris システムでの Caching Proxy のインストール前提条件

このセクションでは、Solaris オペレーティング・システムが稼働しているマシンに Caching Proxy をインストールする場合のハードウェアおよびソフトウェアの前提条件について説明します。

- **ハードウェア・アーキテクチャー:** SPARC または UltraSPARC。
- **Solaris のバージョン:** Solaris 8 または Solaris 9。どちらも、2004 年 6 月に公開されたパッチ・クラスターが必要です。32 ビット・モードまたは 64 ビット・モードがサポートされます。

Solaris 8 の場合、インストール・ウィザードで、レベル 109147-16 またはそれ以上のリンカー、およびレベル 108434-8 またはそれ以上の C++ 用の共用ライブラリーが必要です。

整合性のある動作になるように、<http://sunsolve.sun.com> で Sun Microsystems から最新の Solaris パッチをダウンロードしてください。

- **ソフトウェアおよび文書のための空きディスク・スペース:** 50 MB にログ用のスペースを加えた容量。
- **通信ハードウェア・アダプター:** TCP/IP スタックを使用してネットワーク接続を行うように構成する必要があります。
- **RAM:** 64 MB 以上。ただし、容量が大きければ大きいほどパフォーマンスが向上します。
- **ディスク・キャッシング用の空きディスク・スペース:** ユーザーが Web サーバーから検索するファイルのサイズと数に依存します。キャッシュが大きいほど、一般にキャッシュ・アクセス率が高くなります。推奨最小値は、メモリー・キャッシングの場合と同じです。
- **ページング用空きディスク・スペース:** RAM 容量の 2 倍以上。
- **使用できるファイル記述子の数:** 1024。Solaris 用の Caching Proxy 初期化スクリプト (`/etc/init.d/ibmproxy` としてインストールされる) には、適切な `ulimit` コマンドが組み込まれているため、インストール担当者または管理者の処置は必要ありません。
- **iKeyman:** これは Secure Sockets Layer (SSL) が使用する暗号鍵の操作作用として Caching Proxy に含まれているユーティリティです。このユーティリティは、32 ビットの Java ランタイム環境 (JRE) バージョン 1.4 で使用する必要があります。

- 動的キャッシングは、WebSphere Application Server バージョン 4、5、および 6 を使用する場合にのみサポートされます。
- オプションとして、暗号ハードウェア・カード Rainbow CryptoSwift Accelerator を使用できます。

## Solaris システムでの Load Balancer のインストール前提条件

このセクションでは、Solaris オペレーティング・システムが稼働しているマシンに Load Balancer コンポーネントをインストールする場合のハードウェアおよびソフトウェアの前提条件について説明します。

- **ハードウェア・アーキテクチャー:** SPARC または UltraSPARC 60 サーバー。
- **Solaris のバージョン:** Solaris 8 または Solaris 9。どちらも、2004 年 6 月に公開されたパッチ・クラスターが必要です。32 ビット・モードまたは 64 ビット・モードがサポートされます。

Solaris 8 の場合、インストール・ウィザードで、レベル 109147-16 またはそれ以上のリンカー、およびレベル 108434-8 またはそれ以上の C++ 用の共用ライブラリーが必要です。

整合性のある動作になるように、<http://sunsolve.sun.com> で Sun Microsystems から最新の Solaris パッチをダウンロードしてください。

- **ソフトウェアおよび文書のための空きディスク・スペース:** 50 MB にログ用のスペースを加えた容量。
- 以下のいずれかのネットワーク・インターフェース・カード (NIC):
  - 10 Mb イーサネット。
  - 100 Mb イーサネット。
  - 1 Gb イーサネット (UltraSparc 60 サーバー・ハードウェアでのみサポート)。
  - マルチポート・イーサネット NIC。モード 1 だけがサポートされます。モード 2 (フォールト・トレランス) およびモード 3 (ポート集約) はサポートされません。

**注:** マルチポート NIC のインプリメンテーションは、ベンダーにより異なります。したがって、一部のマルチポート NIC のサポートが制限される場合があります。

- 32 ビットの Java 2 ランタイム環境、標準版、バージョン 1.4.2。
- **CBR コンポーネント**と **Web ベース管理**の両方が、インストールおよび構成された Caching Proxy を必要とします。
- **Web ベース管理**は Perl v5.5 を必要とします。
- **Consultant for Cisco CSS Switch** は、Cisco CSS 11000 シリーズ・スイッチをインストールして構成する必要があります。
- **Nortel Alteon Controller** には、インストールされ、構成済みの Alteon AD3、AD4、180e、184、または Passport 8600 用の layer 4/7 blade が必要です。Web OS バージョン 9 または 10 は、Nortel Alteon Web Switch ファミリーでサポートされるソフトウェアです。



## Windows システムでの Caching Proxy のインストール前提条件

このセクションでは、Windows® オペレーティング・システムが稼働しているマシンに Caching Proxy をインストールする場合のハードウェアおよびソフトウェアの前提条件について説明します。

- **ハードウェア・アーキテクチャー:** Intel x86 プロセッサ
- **Windows のバージョン:**
  - Windows 2000 Server SP4、Windows 2000 Advanced Server SP4、または Windows 2000 Professional SP4
  - Windows Server 2003 Standard、Windows Server 2003 Enterprise、または Windows Server 2003 Datacenter Edition
- **ソフトウェアおよび文書のための空きディスク・スペース:** 50 MB にログ用のスペースを加えた容量。
- **通信ハードウェア・アダプター:** TCP/IP スタックを使用してネットワーク接続を行うように構成する必要があります。
- **RAM:** 64 MB。ただし、容量が大きければ大きいほどパフォーマンスが向上します。
- **ディスク・キャッシング用の空きディスク・スペース:** ユーザーが Web サーバーから検索するファイルのサイズと数に依存します。キャッシュが大きいほど、一般にキャッシュ・アクセス率が高くなります。推奨最小値は、メモリー・キャッシングの場合と同じです。
- **ページング用空きディスク・スペース:** RAM 容量の 2 倍以上。
- **iKeyman:** これは Secure Sockets Layer (SSL) が使用する暗号鍵の操作用として Application Server に含まれているユーティリティです。このユーティリティは、32 ビットの Java ランタイム環境 (JRE) バージョン 1.4 で使用する必要があります。
- 動的キャッシングは、WebSphere Application Server バージョン 4、5、および 6 を使用する場合にのみサポートされます。
- オプションとして、暗号ハードウェア・カード Rainbow CryptoSwift Accelerator を使用できます。

## Windows システムでの Load Balancer のインストール前提条件

このセクションでは、Windows オペレーティング・システムが稼働しているマシンに Load Balancer コンポーネントをインストールする場合のハードウェアおよびソフトウェアの前提条件について説明します。

- **ハードウェア・アーキテクチャー:** Intel x86 プロセッサ
- **Windows のバージョン:**
  - Windows 2000 Server SP4、Windows 2000 Advanced Server SP4、または Windows 2000 Professional SP4
  - Windows Server 2003 Standard、Windows Server 2003 Enterprise、または Windows Server 2003 Datacenter Edition
- **ソフトウェアおよび文書のための空きディスク・スペース:** 50 MB にログ用のスペースを加えた容量。
- 以下のいずれかのネットワーク・インターフェース・カード (NIC):

- 16 Mb トークンリング。
- 10 Mb イーサネット。
- 100 Mb イーサネット。
- 1 Gb イーサネット。
- マルチポート・イーサネット NIC。

**注:** マルチポート NIC のインプリメンテーションは、ベンダーごとに異なります。したがって、一部のマルチポート NIC のサポートが制限される場合があります。

- **32 ビットの Java 2 ランタイム環境:** このソフトウェアは、IBM Cross-Platform Technologies for Windows, バージョン 2.0 (バージョン 1.4.2 の SDK) で使用可能です。
- **CBR コンポーネントと Web ベース管理**の両方が、インストールおよび構成された Caching Proxy を必要とします。
- **Web ベース管理**は Perl v5.5 を必要とします。
- **Consultant for Cisco CSS Switch** は、Cisco CSS 11000 シリーズ・スイッチをインストールして構成する必要があります。
- **Nortel Alteon Controller** には、インストールされ、構成済みの Alteon AD3、AD4、180e、184、または Passport 8600 用の layer 4/7 blade が必要です。Web OS バージョン 9 または 10 は、Nortel Alteon Web Switch ファミリーでサポートされるソフトウェアです。

---

## Caching Proxy 「構成および管理」 フォームを使用するブラウザの使用

### ブラウザの最小要件

「構成および管理」フォームを使用して Caching Proxy を構成するには、ブラウザは以下のことが可能である必要があります。

- フレームを表示すること。
- JavaScript™ と Java の両方を使用可能にすること。
- カラー解像度を少なくとも 256 カラー (オペレーティング・システムの設定) に設定すること。
- 文書をキャッシュし、キャッシュされた文書を常時ネットワーク文書と比較するように設定すること。

**Linux および UNIX システムの場合:** 推奨ブラウザは Mozilla 1.4 です。

**Windows システムの場合:** 推奨ブラウザは Microsoft® Internet Explorer 5.5 以降、または Mozilla 1.4 です。

制限: 展開エレメントの数が多すぎてブラウザのウィンドウに表示しきれない場合、管理フォームの左側の垂直スクロール・バーが表示されない場合があります。これにより、リスト下部の展開エレメントがブラウザの現行表示ウィンドウから押し出され、エレメントにアクセスできなくなります。この問題を解決するには、左側のメニューの展開エレメントの数を制限します。展開エレメントの数が多すぎる場合は、一部のエレメントを縮小表示して、ブラウザのウィンドウでリスト下部のエレメントが表示されるようにします。

フォームを適切に表示するには、フォームを実際に表示する (ブラウザが常駐する) オペレーティング・システムに、フォームが書かれた言語に該当するフォント・セットが含まれていなければなりません。ただし、ブラウザ・インターフェースが必ずしもフォームと同じ言語である必要はありません。

例えば、Solaris 9 システムで中国語版のプロキシ・サーバーが実行されているとします。インターフェースが英語バージョンの Mozilla ブラウザーが、Solaris ホストにロードされています。このブラウザは、「構成および管理」フォームのローカルな編集に使用できます。(フォームは、プロキシ・サーバーが使用する文字セットでブラウザに提供されます (この例では中国語)。ただし、プロキシ・サーバーが送信した文字セットを表示するようにブラウザとその基礎オペレーティング・システムが適切に構成されていない場合、フォームが正しく表示されないことがあります。)

あるいは、中国語をサポートする Windows ワークステーションをプロキシ・サーバーにリモート接続できる場合には、Netscape ブラウザーの中国語版を Windows ワークステーション上にロードし、このブラウザを使用してフォームに値を入力することが可能です。この 2 番目のソリューションには、管理者に対し一貫した言語インターフェースを維持できる利点があります。

オペレーティング・システムに特有のフォント・セットは、ブラウザ内の各種の言語、特に 2 バイト文字の表示に大きく影響します。例えば、AIX 上の特定中国語フォント・セットは、Windows プラットフォーム上の中国語フォント・セットとは正確に同じには見えません。このため、「構成および管理」フォームの中で HTML テキストと Java アプレットの外観に多少の不ぞろいが生じます。最良の外観を得るためにお勧めできるのは、Windows オペレーティング・システムで実行されるブラウザだけです。

#### **Mozilla 1.4 ブラウザーに関する注意**

Mozilla 1.4 とともにインストールされている Java プラグインは、バージョン 1.4.2 以降に更新する必要があります (管理フォームを適切に表示できるようにするため)。プラグインを更新するには、以下のステップを行います。

1. <http://plugindoc.mozdev.org> にアクセスします。
2. 「文書」セクションでプラットフォームを選択します。
3. 「Java ランタイム環境」セクションでリストされている説明に従ってプラグインを更新します。

---

## **Load Balancer オンライン・ヘルプでのブラウザの使用**

Load Balancer オンライン・ヘルプを使用するには、ブラウザが以下をサポートしている必要があります。

- HTML 4
- Cascading Style Sheets
- JavaScript テクノロジー
- Java アプレット

上記の要件をサポートしていないブラウザを使用すると、ページのフォーマットに問題が発生したり、機能が正しく動作しない場合があります。次のブラウザは上記の要件をサポートします。

- **Linux および UNIX システムの場合:** デフォルトのブラウザは Mozilla です。推奨されるバージョンは Mozilla 1.4 です。
- **Windows システムの場合:** デフォルトのブラウザは、システムのブラウザです。推奨されるブラウザは、Microsoft Internet Explorer 5.5 以降、または Mozilla 1.4 です。

---

## 第 12 章 セットアップ・プログラムを使用した Edge components のインストール

この章では、セットアップ・プログラム を使用して Edge components をインストールする方法について説明します。

重要: インストール後に、Caching Proxy パッケージ内のスクリプトが、デフォルトの構成を使用してプロキシ・サーバーを開始しようとします。ポート 80 が (別の Web サーバーなどによって) 使用中の場合、プロキシ・サーバーの開始は失敗します。

---

### Windows でのセットアップ・プログラムの使用

セットアップ・プログラムを使用して、Edge Components をご使用の Windows システムにインストールします。手順は、以下のとおりです。

1. Windows システムが、47 ページの『第 11 章 Edge components の要件』で説明するハードウェアおよびソフトウェア要件をすべて満たしていることを確認します。
2. 管理者特権を持つユーザーとしてログインします。
3. Edge Components の CD-ROM をマシンの CD-ROM ドライブに挿入します。ランチパッドが自動的に開始されます。
4. 「**WebSphere Application Server - Edge Components のインストール・ウィザードを立ち上げる**」をクリックします。セットアップ・プログラムは自動的に開始されます。InstallShield ウィザードが準備され「ようこそ」ウィンドウがオープンします。

**注:** ご使用のマシンが Autoplay オプションをサポートしないか、またはこれがオフになっていると、CD-ROM の最上位レベルのディレクトリーにある setup.exe プログラムを実行してセットアップ・プログラムを手動で開始してください。

5. 「**次へ**」をクリックしてインストールを続行します。「ソフトウェア・ライセンス契約」ウィンドウがオープンします。
6. ライセンス契約を読んで「**はい**」をクリックし、すべての条項を受け入れます。「コンポーネント選択」ウィンドウがオープンします。

**注:** Edge components がすでにインストールされている場合は、「保守オプション」ウィンドウが開いてから、「コンポーネント選択」ウィンドウが開きます。「**変更**」ラジオ・ボタンを選択してから「**次へ**」をクリックします。「コンポーネント選択」ウィンドウがオープンします。

7. インストールするコンポーネントを選択します。
8. 所定のコンポーネントでインストールされるサブコンポーネントの選択を変更するには、コンポーネントの名前をクリックしてそれを選択してから、「**サブコンポーネントの変更**」をクリックします。活動状態のコンポーネントの別のサブコンポーネントを示す、別の「コンポーネント選択」ウィンドウがオープン

ンします。インストールするサブコンポーネント、コンポーネントの言語、およびコンポーネントをインストールする場所を選択するには、以上の同じ手順を使用してください。

9. 「**現行言語**」メニューを使用して、Edge components をインストールしたい言語 (1 つまたは複数) を選択します。使用可能な言語がメニューの左側にリストされます。選択された言語がメニューの右側にリストされます。
10. 「コンポーネント選択」ウィンドウを使用して Edge components のインストール場所を確認します。デフォルト値を受け入れるか、あるいは「**フォルダーの変更**」をクリックして新しい場所を指定できます。

**注:** デフォルト以外のインストール場所を選択する場合、パス名の中にブランク・スペースがないことを確認してください。例えば、  
C:\¥My Files¥edgeserver¥ のようなパス名は避けてください。

11. 「コンポーネント選択」ウィンドウを使用して、選択したインストール場所を使用可能なスペースが十分であることを確認します。選択した場所を使用可能なスペースが十分でない場合、「**フォルダーの変更**」をクリックして新しいインストール場所を指定します。
12. Edge components、インストール場所、および言語を選択したら、「**次へ**」をクリックします。オープンした「インストールの確認」ウィンドウ内の情報を確認します。1 つ以上の選択を変更するには、「**戻る**」をクリックして「コンポーネント選択」ウィンドウに戻り、変更を行います。選択項目を確認したら、「**終了**」をクリックします。
13. Edge components 製品セットアップ・プログラムは、選択した Edge components および GSK (必要な場合) の指定したインストール場所へのインストールを開始します。
14. 「セットアップ完了」ウィンドウがオープンします。Edge components README ファイルをお読みにになりたい場合は、「**はい、README ファイルを表示します**」チェック・ボックスが選択されていることを確認します。README ファイルがデフォルト・ブラウザでオープンされます。
15. 「**はい、コンピューターを再始動します**」チェック・ボックスが選択されていることを確認してから、「**終了**」をクリックします。README ファイルを表示するように選択した場合、ファイルを表示するブラウザのウィンドウをクローズするとマシンが再始動されます。選択しなければ、Edge components 製品セットアップ・プログラムは即時にクローズされてマシンが再始動されます。新たにインストールした Edge components を使用する前にマシンを再始動する必要があることに注意してください。

---

## Linux および UNIX でのセットアップ・プログラムの使用

セットアップ・プログラムを使用して、Linux および UNIX システムに Edge components をインストールするには以下の手順を行います。

1. コンピューター・サーバーが、47 ページの『第 11 章 Edge components の要件』で説明するハードウェアおよびソフトウェア要件をすべて満たしていることを確認します。
2. スーパーユーザー (通常は、root) としてログインします。
3. Edge components CD-ROM を CD-ROM ドライブに挿入します。必要であれば、CD-ROM をマウントします。
4. 作業ディレクトリーを CD-ROM の最上位レベルのディレクトリーに変更します。
5. 次のコマンドを入力して、セットアップ・プログラムを起動します。

```
# ./install
```

「ようこそ」ウィンドウがオープンします。

6. 「次へ」をクリックしてインストールを続行します。「ソフトウェア・ライセンス契約」ウィンドウがオープンします。
7. ライセンス契約を読んで「はい」をクリックし、すべての条項を受け入れます。「言語選択」ウィンドウがオープンします。
8. Edge components の今回のインストールでサポートされる言語を選択します。「次へ」をクリックします。「コンポーネント選択」ウィンドウがオープンします。
9. インストールするコンポーネントを選択します。
10. 「次へ」をクリックします。「インストールの確認」ウィンドウがオープンします。
11. 「インストールの確認」ウィンドウの情報を確認します。1 つまたは複数の選択を変更したい場合、「戻る」をクリックして「コンポーネント選択」ウィンドウに戻り、変更を行います。選択項目を確認したら、「次へ」をクリックします。

セットアップ・プログラムは、選択した Edge components および必須パッケージのインストールを開始します。

12. 「インストールの結果要約」ウィンドウがオープンします。結果を確認したら、「終了」をクリックします。





---

## 第 13 章 システム・パッケージ・ツールを使用した Caching Proxy のインストール

この章では、システム・パッケージ・ツールを使用して Caching Proxy をインストールする方法について説明します。

**重要:** インストール後に、Caching Proxy パッケージ内のスクリプトが、デフォルトの構成を使用してプロキシ・サーバーを開始しようとします。ポート 80 が (別の Web サーバーなどによって) 使用中の場合、プロキシ・サーバーの開始は失敗します。

オペレーティング・システムのパッケージ・インストール・システムを使用して、64 ページの表 4 でリストされた順序でパッケージをインストールします。以下の手順には、このタスクを完了するのに必要な通常ステップが示されています。

1. 必要に応じて、Edge Components の CD を CD-ROM ドライブに挿入し、ドライブをマウントします。
2. ローカル・スーパーユーザー `root` になります。

```
su - root  
Password: password
```

3. CD の、適切なディレクトリーに変更します。

```
cd mount_point/package_directory/
```

4. パッケージをインストールします。

AIX の場合:

```
installp -acXd ./packagename
```

HP-UX の場合:

```
swinstall -s ./packagename
```

Linux の場合:

```
rpm -i ./packagename
```

Solaris の場合:

```
pkgadd -d ./packagename
```

表 4. Caching Proxy コンポーネント

コンポーネント	インストール・パッケージ (推奨順序)
Caching Proxy	<ol style="list-style-type: none"> <li>1. gskit7</li> <li>2. icu</li> <li>3. admin</li> <li>4. msg-cp-lang</li> <li>5. cp</li> </ol>
Edge component 文書	doc-lang <sup>1</sup>
<p>注:</p> <ol style="list-style-type: none"> <li>1. Load Balancer 資料は、2 つのパッケージで提供されます。ibmlb-doc パッケージによって、Load Balancer 資料だけがインストールされ、これらは <code>../edge/lb/ directory</code> ディレクトリ内のサブディレクトリに入れられます。doc-lang パッケージには、Load Balancer 資料を含む Edge components 資料が組み込まれていて、これらは、<code>../edge/doc/ directory</code> ディレクトリに入れられます。</li> </ol>	

表 5. AIX、HP-UX、および Solaris パッケージ・ファイル名

総称パッケージ名	Solaris ファイル名	AIX ファイル・セット	HP-UX ファイル・セット
admin	WSESadmin	wses_admin.rte	WSES-ADMIN
cp	WSEScp	wses_cp.base	WSES-CP
doc-lang	WSESdoclang <sup>1</sup>	wses_doc.lang <sup>2</sup>	WSES-DOC-lang <sup>3</sup>
gskit7	gsk7bas	gskkm.rte	gsk7bas
icu	WSESicu	wses_icu.rte	WSES-ICU
msg-cp-lang	WSEScpmlang <sup>1</sup>	wses_cp.msg.lang <sup>2</sup> .base	WSES-cpmlang <sup>3</sup>
<p>注:</p> <ol style="list-style-type: none"> <li>1. Solaris の場合、変数 lang は、言語固有コード (br、cn、cw、de、en、es、fr、it、ja、kr) のいずれかに置き換えます。</li> <li>2. AIX の場合、変数 lang は、言語固有コード (en_US、pt_BR、zh_CN、Zh_TW、fr_FR、de_DE、it_IT、Ja_JP、ko_KR、es_ES) のいずれかに置き換えます。</li> <li>3. HP-UX の場合、変数 lang は、言語固有コード (en_US、pt_BR、zh_CN、zh_TW、fr_FR、de_DE、it_IT、ja_JP、ko_KR、es_ES) のいずれかに置き換えられます。</li> </ol>			

表 6. Linux パッケージ・ファイル名

総称パッケージ名	Linux ファイル名
admin	WSES_Admin_Runtime-6.0.0-0.hardw <sup>1</sup> .rpm
cp	WSES_CachingProxy-6.0.0-0.hardw <sup>1</sup> .rpm
doc-lang	WSES_Doc_lang <sup>2</sup> -6.0.0-0.hardw <sup>1</sup> .rpm
gskit7	gsk7bas.rpm
icu	WSES_ICU_Runtime-6.0.0-0.hardw <sup>1</sup> .rpm
msg-cp-lang	WSES_CachingProxy_msg_lang <sup>2</sup> -6.0.0-0.hardw <sup>1</sup> .rpm

表 6. Linux パッケージ・ファイル名 (続き)

総称パッケージ名	Linux ファイル名
注:	
1. 変数 <i>hardw</i> は、値 <i>i686</i> 、 <i>s390</i> 、 <i>ppc64</i> のいずれかに置き換えられます。	
2. 変数 <i>lang</i> は、言語固有コード ( <i>en_US</i> 、 <i>pt_BR</i> 、 <i>zh_CN</i> 、 <i>zh_TW</i> 、 <i>fr_FR</i> 、 <i>de_DE</i> 、 <i>it_IT</i> 、 <i>ja_JP</i> 、 <i>ko_KR</i> 、 <i>es_ES</i> ) のいずれかに置き換えられます。	

パッケージをアンインストールするには、以下を行います。

AIX の場合:

```
installp -u packagename
```

Caching Proxy パッケージをすべてアンインストールするには、以下のコマンドを使用します。

```
installp -u wses
```

HP-UX の場合:

```
swremove packagename
```

Caching Proxy パッケージをすべてアンインストールするには、以下のコマンドを使用します。

```
swremove wses
```

Linux の場合:

```
rpm -e packagename
```

インストール済みの Caching Proxy パッケージを照会するには、以下のコマンドを使用します。

```
rpm -qa |grep -i wses
```

パッケージは、インストール時とは逆の順番で除去されます。

Solaris の場合:

```
pkgadd -d ./packagename
```

インストール済みの Caching Proxy パッケージを照会するには、以下のコマンドを使用します。

```
pkginfo | grep wses
```

パッケージは、インストール時とは逆の順番で除去されます。



---

## 第 14 章 システム・パッケージ・ツールを使用した Load Balancer のインストール

この章では、AIX、HP-UX、Linux、および Solaris システムへの Load Balancer のインストールについて説明します。

- 68 ページの『AIX へのインストール』
- 71 ページの『HP-UX へのインストール』
- 73 ページの『Linux へのインストール』
- 75 ページの『Solaris へのインストール』

### 注:

1. 以前のバージョンの Load Balancer からマイグレーションする場合、またはオペレーティング・システムを再インストールする場合は、インストールに先立ち、Load Balancer の既存の構成ファイルやスクリプト・ファイルを保管しておきます。
  - インストール後にその構成ファイルを、  
...ibm/edge/lb/servers/configurations/**component** ディレクトリー (ここで **component** は、dispatcher、cbr、ss、cco、または nal のいずれかです) に配置してください。
  - インストール後にスクリプト・ファイル (goIdle や goStandby など) を、  
.../ibm/edge/lb/servers/bin ディレクトリーに配置して、実行できるようにします。
2. Load Balancer をインストールした後にマシンからログオフした場合、再度ログオンしたときにすべての Load Balancer サービスを再始動する必要があります。
3. Load Balancer に必要な 32 ビット Java レベルは、バージョン 1.4.2 です。Load Balancer マシンに配置されている一部のアプリケーションでは、別のバージョンの Java が必要となる可能性があるため、アップグレードを行う際には、正しいバージョンの Java をインストールする必要があります。

複数のバージョンがインストールされているときに、Load Balancer コンポーネントが正しいバージョンの Java を使用していることを確認するには、以下を実行してください。

- a. オペレーティング・システムに応じて正しいバージョンの Java 1.4 をインストールします。
- b. Load Balancer スクリプト・ファイルを、Java 1.4 を使用するように編集します。デフォルトでは、スクリプト・ファイルは以下のディレクトリーに配置されます (これらのファイルはデフォルトで読み取り専用となっているため、変更を保管する前に、これらのファイルのアクセス権を変更しておく必要があります):

```
/usr/bin/scriptfile
```

アップグレードする Load Balancer のコンポーネントについて、以下のスクリプト・ファイルを編集します。

- **管理:** lbadmim、lbkeys
  - **Dispatcher:** dsserver、dscontrol、dswizard
  - **Content Based Routing (CBR):** cbrserver、cbrcontrol、cbrwizard
  - **Site Selector:** sssserver、sscontrol、sswizard
  - **Cisco CSS Controller:** ccoserver、ccocontrol
  - **Nortel Alteon Controller:** nalserver、nalcontrol
- c. スクリプト・ファイルのどこに **java** または **javaw** コマンドが記述されていても、コマンドが Java 1.4.2 インストール・ディレクトリーのどこに配置されているのかを示すパスを、接頭部として含めてください。

たとえば Windows システムで、Java 1.4.2 が C:\Program Files\IBM\Java142\jre\bin にインストールされている場合、dsserver.cmd ファイルで javaw を以下のように変更します。

```
C:\Program Files\IBM\Java142\jre\bin\javaw
```

## AIX へのインストール

表 7 には、Load Balancer の AIX ファイル・セットがリストされています。

表 7. AIX ファイル・セット

Load Balancer コンポーネント	AIX ファイル・セット
Load Balancer コンポーネント (メッセージ付き)	ibmlb.component.rte ibmlb.msg.language.lb
デバイス・ドライバー	ibmlb.lb.driver
ベース	ibmlb.base.rte
管理 (メッセージ付き)	ibmlb.admin.rte ibmlb.msg.language.admin
文書 (メッセージ付き)	ibmlb.doc.rte ibmlb.msg.language.doc
ライセンス	ibmlb.lb.license
Metric Server	ibmlb.ms.rte

注:

1. 変数 *component* は次に置き換えることができます。 disp (dispatcher)、 cbr (CBR)、 ss (Site Selector)、 cco (Cisco CSS Controller)、 または nal (Nortel Alteon Controller)。
2. 変数 *lang* は次に置き換えることができます。 en\_US、 de、 es\_ES、 fr、 it、 ja\_JP、 Ja\_JP、 ko\_KR、 pt\_BR、 zh\_CN、 zh\_TW、 Zh\_TW。

## インストールする前に

AIX への Load Balancer のインストールを行う前に、以下を確認してください。

- root としてログインしている。
- Edge components メディアが挿入されている、または Web からインストールする場合は、インストール・イメージがディレクトリーにコピーされている。前のバージョンの製品はアンインストールされている。アンインストールを行うには、すべての executor とすべてのサーバーが停止されていることを確認します。次に、製品全体をアンインストールするために、次のコマンドを入力します。

```
installp -u ibmlb
```

または、前のバージョンに対して次のコマンドを入力します。

```
installp -u ibmnd
```

特定のファイル・セットをアンインストールするには、パッケージ名 `ibmlb` を指定する代わりにファイル・セットを個別にリストします。

製品をインストールするとき、以下の一部またはすべてをインストールするためのオプションが提供されます。

- Dispatcher コンポーネント (メッセージ付き)
- CBR コンポーネント (メッセージ付き)
- Site Selector コンポーネント (メッセージ付き)
- Cisco CSS Controller コンポーネント (メッセージ付き)
- Nortel Alteon Controller コンポーネント (メッセージ付き)
- ベース管理
- 管理 (メッセージ付き)
- デバイス・ドライバー (必須)
- ライセンス (必須)
- 文書 (メッセージ付き)
- Metric Server

## インストール手順

SMIT では、すべてのメッセージが自動的にインストールされるため、SMIT を使用して Load Balancer for AIX をインストールすることをお勧めします。

### SMIT を使用した Load Balancer for AIX のインストール

1. 「ソフトウェア・インストールおよび保守 (Software Installation and Maintenance)」を選択します。
2. 「ソフトウェアのインストール / 更新 (Install and Update Software)」を選択します。
3. 「最新の使用可能なソフトウェアからインストール / アップデート (Install and update from latest Available Software)」を選択します。
4. ファイル・セットを含むデバイスまたはディレクトリーを入力します。
5. 「\*インストールするソフトウェア (\*SOFTWARE to Install)」フィールドに、オプションを指定するための適切な情報を入力します (または「リスト (List)」を選択します)。
6. 「OK」を押します。
7. コマンドが完了したら、「Done (完了)」を押します。
8. 「終了 (Exit)」メニューの「Smit の終了 (Exit Smit)」を選択するか、または **F12** を押して SMIT を終了します。SMITTY を使用している場合は、**F10** を押してプログラムを終了します。

## コマンド行からの Load Balancer のインストール

1. CD からインストールを行う場合は、以下のコマンドを入力して CD をマウントします。

```
mkdir /cdrom
mount -v cdrfs -p -r /dev/cd0 /cdrom
```

2. 以下の表を参照して、必要な AIX 用 Load Balancer パッケージをインストールするために入力するコマンド (1 つ以上) を判別してください。

表 8. AIX インストール・コマンド

パッケージ	コマンド
Load Balancer コンポーネント (メッセージ付き)。次を含む: Dispatcher、CBR、Site Selector、Cisco CSS Controller、および Nortel Alteon Controller	<code>installp -acXgd device ibmlb.component.rte ibmlb.msg.language.lb</code>
デバイス・ドライバー	<code>installp -acXgd device ibmlb.lb.driver</code>
文書 (メッセージ付き)	<code>installp -acXgd device ibmlb.doc.rte ibmlb.msg.language.lb</code>
ベース	<code>installp -acXgd device ibmlb.base.rte</code>
管理 (メッセージ付き)	<code>installp -acXgd device ibmlb.admin.rte ibmlb.msg.language.admin</code>
ライセンス	<code>installp -acXgd device ibmlb.lb.license</code>
Metric Server	<code>installp -acXgd device ibmlb.ms.rte</code>

ここで、*device* は以下のとおりです。

- CD からインストールする場合、*/cdrom*
  - ファイル・システムからインストールする場合、*ldir* (ファイル・セットを含むディレクトリー)
3. インストール (APPLY) する Load Balancer の各パーツについて、要約に示される結果の列に **SUCCESS** が含まれていることを確認してください。インストールするパーツがすべて正常に適用されない限り、続行しないでください。

**注:** 特定のデバイスについてファイル・セットのリスト (使用可能なすべてのメッセージ・カタログを含む) を生成するには、次を入力します。

```
installp -ld device
```

CD をアンマウントするには、次のコマンドを入力します。

```
umount /cdrom
```

次のコマンドを入力して製品がインストールされたことを検査します。

```
lsllpp -h | grep ibmlb
```

全製品をインストールした場合、このコマンドは以下を戻します。

```
ibmlb.admin.rte
ibmlb.base.rte
ibmlb.doc.rte
ibmlb.ms.rte
ibmlb.msg.language.admin.rte
ibmlb.msg.language.doc
```



```
ibmlb.msg.language.lb.rte
ibmlb.lb.driver
ibmlb.lb.license
ibmlb.component.rte
```

**Load Balancer** インストール・パスには、以下が入っています。

- 管理 — /opt/ibm/edge/lb/admin
- Load Balancer コンポーネント — /opt/ibm/edge/lb/servers
- Metric Server — /opt/ibm/edge/lb/ms
- 文書 (管理ガイド) — /opt/ibm/edge/lb/documentation

---

## HP-UX へのインストール

このセクションでは、製品 CD を使用して、Load Balancer を HP-UX にインストールする方法について説明します。

### インストールする前に

インストール手順を開始する前に、ソフトウェアをインストールする root 権限を持っていることを確認してください。

以前のバージョンをインストールしている場合は、そのバージョンをアンインストールしてから現行バージョンをインストールしてください。まず、`executor` とサーバーの両方を停止したことを確認してください。次に、Load Balancer アンインストール手順について、72 ページの『パッケージのアンインストール手順』を参照してください。

### インストール手順

表 9 には、Load Balancer に必要なインストール・パッケージ名のリストと、システムのパッケージ・インストール・ツールを使用してパッケージをインストールする順番が掲載されています。

表 9. Load Balancer 用の HP-UX パッケージのインストールの詳細

パッケージの説明	HP-UX パッケージ名
ベース	ibmlb.base
管理	ibmlb.admin
Load Balancer ライセンス	ibmlb.lic
Load Balancer コンポーネント	ibmlb.component
文書	ibmlb.lang
Metric Server	ibmlb.ms
注:	
1. 変数 <code>component</code> は次のいずれか 1 つに置き換えられます。 <code>disp</code> (dispatcher)、 <code>cbr</code> (CBR)、 <code>ss</code> (Site Selector)、 <code>cco</code> (Cisco CSS Controller)、または <code>nal</code> (Nortel Alteon Controller)。	
2. 変数 <code>lang</code> は、次の言語固有コードのいずれか 1 つに置き換えられます。 <code>nlv-de_DE</code> 、 <code>nlv-en_US</code> 、 <code>nlv-es_ES</code> 、 <code>nlv-fr_FR</code> 、 <code>nlv-it_IT</code> 、 <code>nlv-ja_JP</code> 、 <code>nlv-ko_KR</code> 、 <code>nlv-zh_CN</code> 、または <code>nlv-zh_TW</code> 。英語の場合は、変数 <code>lang</code> は <code>doc</code> に置き換えられます。	

**注:** HP-UX では ブラジル・ポルトガル語 (pt\_BR) ロケールはサポートされていません。 HP-UX でサポートされているロケールは、以下のとおりです。

- en\_US.iso88591
- es\_ES.iso88591
- fr\_FR.iso88591
- de\_DE.iso88591
- it\_IT.iso88591
- ja\_JP.SJIS
- ko\_KR.eucKR
- zh\_CN.hp15CN
- zh\_TW.big5

## パッケージのインストール手順

以下の手順では、この作業の完了に必要なステップの詳細を示します。

1. ローカル・スーパーユーザー `root` になります。

```
su - root
Password: password
```

2. インストール・コマンドを実行し、パッケージをインストールします。

```
インストール・コマンド
swinstall -s source/package_name
```

を実行します。ここで、*source* はパッケージの入るディレクトリーであり、*package\_name* はパッケージの名前です。

例えば、`CD` のルートからインストールしている場合は、次のコマンドで Load Balancer のベース・パッケージ (`ibmlb.base`) がインストールされます。

```
swinstall -s /lb ibmlb.base
```

3. Load Balancer パッケージがインストールされていることを確認します。

**swlist** コマンドを実行し、これまでにインストールしたすべてのパッケージをリストします。例えば、

```
swlist -l fileset ibmlb
```

## パッケージのアンインストール手順

**swremove** コマンドを使用してパッケージをアンインストールできます。パッケージは、最後にインストールしたものから順に除去する必要があります。例えば、次のコマンドを実行します。

- すべての Load Balancer パッケージをアンインストールする場合:

```
swremove ibmlb
```

個々のパッケージ (例えば、Cisco CSS Controller) をアンインストールする場合:

```
swremove ibmlb.cco
```

Load Balancer インストール・パスには、以下が組み込まれています。

- 管理 — `/opt/ibm/edge/lb/admin`
- Load Balancer コンポーネント — `/opt/ibm/edge/lb/servers`

- Metric Server — /opt/ibm/edge/lb/ms
- 文書 (管理ガイド) — /opt/ibm/edge/lb/documentation

---

## Linux へのインストール

このセクションでは、Edge components CD を使用して Load Balancer を Linux にインストールする方法について説明します。

### インストールする前に

Load Balancer をインストールする前に、以下を確認してください。

- root としてログインしている。
- 前のバージョンの製品はアンインストールされている。アンインストールを行うには、すべての `executor` とすべてのサーバーが停止されていることを確認します。次に、製品全体をアンインストールするために、次のコマンドを入力します。

```
rpm -e pkgname
```

アンインストールを行うときは、パッケージのインストールで使った順序を逆にして、管理パッケージが最後にアンインストールされるようにします。

### インストール・ステップ

1. Edge Components メディアを挿入するか、Web サイトから製品をダウンロードして、RPM (Red Hat Packaging Manager) を使用してインストール・イメージをインストールします。

インストール・イメージのファイルは、`lblinux-version.tar` 形式です。

2. 次のコマンドを入力して一時ディレクトリーに `tar` ファイルを展開します。

```
tar -xf lblinux-version.tar
```

その結果、`.rpm` 拡張子を持った以下の一連のファイルが生成されます。

- `ibmlb-base-release-version.hardw.rpm` (ベース)
- `ibmlb-admin-release-version.hardw.rpm` (管理)
- `ibmlb-lang-release-version.hardw.rpm` (文書)
- `ibmlb-ms-release-version.hardw.rpm` (Metric Server)
- `ibmlb-component-release-version.hardw.rpm` (LB コンポーネント)
- `ibmlb-lic-release-version.hardw.rpm` (ライセンス)

ここで、

- `release-version` は現行のリリースを示します (6.0-0 など)。
- `hardw` は、`i386`、`s390`、`ppc64` のいずれかの値になります。
- `component` は、`disp` (Dispatcher コンポーネント)、`cbr` (CBR コンポーネント)、`ss` (Site Selector component)、`cco` (Cisco CSS Controller)、または `nal` (Nortel Alteon Controller) のいずれかの値になります。

- *lang* は次の言語固有コードのいずれか 1 つになります。 doc (English)、nlv-de\_DE、nlv-es\_ES、nlv-fr\_FR、nlv-it\_IT、nlv-ja\_JP、nlv-ko\_KR、nlv-pt\_BR、nlv-zh\_CN、nlv-zh\_TW。

3. RPM ファイルがあるディレクトリーから、次のコマンドを実行してそれぞれのパッケージをインストールします。例:

```
rpm -i package.rpm
```

以下の、各コンポーネントに必要なパッケージのリストに示された順序でパッケージをインストールすることが重要です。

- ベース (base)
- 管理 (admin)
- ライセンス (lic)
- Load Balancer コンポーネント (ds、cbr、ss、cco、nal)
- Metric Server (ms)
- 文書 (doc)

**注:** RPM ファイルのうち、少なくとも 1 つは、Java がインストールされていて、RPM データベースに登録済みであることを必要とします。Java がインストールされていても、RPM データベースに登録されていない場合は、以下のように非依存オプションを指定したインストール・コマンドを使用してください。

```
rpm -i --nodeps package.rpm
```

4. 製品がインストールされたことを検査します。次のコマンドを入力します。

```
rpm -qa | grep ibmlb
```

全製品をインストールした場合、以下が出力されます。

- *ibmlb-base-release-version*
- *ibmlb-admin-release-version*
- *ibmlb-doc-release-version*
- *ibmlb-ms-release-version*
- *ibmlb-dsp-release-version*
- *ibmlb-cbr-release-version*
- *ibmlb-ss-release-version*
- *ibmlb-cco-release-version*
- *ibmlb-nal-release-version*
- *ibmlb-lic-release-version*

Load Balancer インストール・パスには、以下が入っています。

- 管理 — /opt/ibm/edge/lb/admin
- Load Balancer コンポーネント — /opt/ibm/edge/lb/servers
- Metric Server — /opt/ibm/edge/lb/ms
- 文書 — /opt/ibm/edge/lb/documentation

パッケージをアンインストールする必要がある場合、パッケージのインストールで使用した順序を逆に、管理パッケージが最後にアンインストールされるようにします。

---

## Solaris へのインストール

このセクションでは、Edge components CD を使用して Load Balancer を Solaris にインストールする方法について説明します。

### インストールする前に

インストール手順を開始する前に、root としてログインしていること、および前のバージョンの製品はアンインストールされていることを確認してください。

アンインストールを行うには、すべての `executor` およびサーバーが停止されていることを確認します。次に、次のコマンドを入力します。

```
pkgrm pkgname
```

### インストール・ステップ

1. Load Balancer ソフトウェアが収納されている CD-ROM を適切なドライブに挿入します。
2. コマンド・プロンプトで、次のコマンドを入力します。

```
pkgadd -d pathname
```

ここで、`-d pathname` は、CD-ROM ドライブのデバイス名またはこのパッケージが入っているハード・ディスクのディレクトリーです  
(例: `-d /cdrom/cdrom0/`)。

以下のパッケージのリストが表示されます。

- `ibmlbadm` (管理)
- `ibmlbase` (ベース)
- `ibmlblic` (ライセンス)
- `ibmlbdisp` (Dispatcher コンポーネント)
- `ibmlbcb` (CBR コンポーネント)
- `ibmlbss` (Site Selector コンポーネント)
- `ibmlbcc` (Cisco CSS Controller コンポーネント)
- `ibmlbnal` (Nortel Alteon Controller コンポーネント)
- `ibmlbms` (Metric Server)
- `ibmlblang` (文書)

上記の変数 `lang` は、言語固有コード (`deDE`、`esES`、`frFR`、`itIT`、`jaJP`、`koKR`、`ptBR`、`zhCN`、`zhTW`) のいずれかに置き換えられます。英語の場合、変数 `lang` は、`doc` に置き換えます。

すべてのパッケージをインストールしたい場合は、`a11` とだけ入力して `Return` を押します。いくつかのコンポーネントをインストールしたい場合は、インストールするパッケージに対応する名前をスペースまたはコンマで区切って入力し、

Return を押します。既存のディレクトリーまたはファイルに対する許可を変更するように指示されることがあります。Return を押すか、または yes と応答します。前提条件のパッケージをインストールする必要があります (インストールが、前提条件のパッケージからではなく、パッケージのアルファベット順に行われるため)。all と入力し、次にすべてのプロンプトに yes と応答した場合、インストールが正常に完了します。

すべてのパッケージは、共通パッケージ `ibmlbadm` に依存しています。この共通パッケージは、他のいずれかのパッケージとともにインストールしなければなりません。

例えば、Dispatcher コンポーネントのみを文書および Metric Server と一緒にインストールする場合、`ibmlbadm`、`ibmlbbase`、`ibmlblic`、`ibmdisp`、`ibmlbms`、および `ibmlbdoc` をインストールしなければなりません。

リモート管理をインストールしたい場合、`ibmlbadm` のみをインストールします。

3. 製品がインストールされたことを検査します。次のコマンドを実行します。

```
pkginfo | grep ibm
```

Load Balancer インストール・パスには、以下が入っています。

- 管理 — `/opt/ibm/edge/lb/admin`
- Load Balancer コンポーネント — `/opt/ibm/edge/lb/servers`
- Metric Server — `/opt/ibm/edge/lb/ms`
- 文書 — `/opt/ibm/edge/lb/documentation`

---

## 第 5 部 Edge components を使用したネットワークの構築

この部では、Edge components を使用して基本デモンストレーション・ネットワークを構築する手順を説明します。これらのネットワークは、実稼働環境での使用を意図していません。ネットワークの最初の構成プロセスによって、この製品を初めて扱う管理者に対して、多くのエッジ・オブ・ネットワークの概念をわかりやすく説明します。すべてのコンポーネント機能の詳細な解説および構成情報については、「*Caching Proxy* 管理ガイド」および「*Load Balancer* 管理ガイド」を参照してください。

この手順では、コンポーネントによってサポートされる任意のコンピューター・システムを任意のノードで使用できます。

この部は、以下の章で構成されています。

79 ページの『第 15 章 *Caching Proxy* ネットワークの構築』

83 ページの『第 16 章 *Load Balancer* ネットワークの構築』





---

## 第 15 章 Caching Proxy ネットワークの構築

図 14 は、3 つのネットワーク・ノードにある 3 台のコンピューター・システムを使用する基本的なプロキシ・サーバー・ネットワークを示しています。このネットワークは、プロキシ・サーバーをサーバー 2 にある専用コンテンツ・ホスト (IBM HTTP Server) に結合し、プロキシ・サーバーがホストを処理します。これは、インターネットによってワークステーションとサーバー 1 の間に位置するよう見えます。

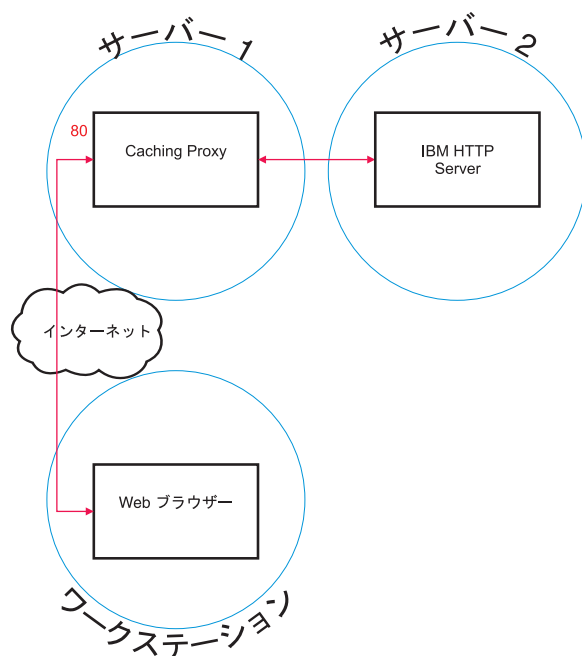


図 14. Caching Proxy デモンストレーション・ネットワーク

---

### ワークフロー

Caching Proxy ネットワークを構築するには、以下の順序で手順を実行します。

1. 必須コンピューター・システムおよびソフトウェアの確認
2. サーバー 1 の構築 (Linux および UNIX システム) またはサーバー 1 の構築 (Windows システム)
3. サーバー 1 の構成
4. Caching Proxy ネットワークのテスト

---

## 必須コンピューター・システムおよびソフトウェアの確認

以下のコンピューター・システムおよびソフトウェア・コンポーネントが必要です。

- サーバー 1 として作動するコンピューター・システム。このシステムは、インターネットへアクセスできる必要があります。
- サーバー 2 として作動するコンピューター・システム。コンテンツ・ホストに HTTP サーバーがインストールされていなければなりません。
- ワークステーションとして作動するコンピューター・システム。Web ブラウザーをインストールしておく必要があります。

---

## サーバー 1 の構築 (Linux および UNIX システム)

以下のように Caching Proxy をインストールして構成します。

1. コンピューター・サーバーが、ハードウェアおよびソフトウェア要件をすべて満たしていることを確認します。
2. スーパーユーザー (通常は、root) としてログインします。
3. Caching Proxy コンポーネントをインストールします。
4. 次のコマンドを入力して、「構成および管理」フォームにアクセスするための管理者 ID およびパスワードを作成します。

```
# htadm -adduser /opt/ibm/edge/cp/server_root/protect/webadmin.passwd
```

プロンプトが出されたら、**htadm** プログラムに、管理者のユーザー名、パスワード、および実名を指定します。

5. 『サーバー 1 の構成』を続けます。

---

## サーバー 1 の構築 (Windows システム)

以下のように Caching Proxy をインストールして構成します。

1. Windows 2000 と Windows 2003 の両オペレーティング・システムが、すべてのハードウェアおよびソフトウェア要件を満たしていることを確認します。
2. 管理者特権を持つユーザーとしてログインします。
3. Caching Proxy コンポーネントをインストールします。
4. 次のコマンドを入力して、「構成および管理」フォームにアクセスするための管理者 ID およびパスワードを作成します。

```
cd "Program Files\IBM\edge\cp\server_root\protect"  
htadm -adduser webadmin.passwd"
```

プロンプトが出されたら、**htadm** プログラムに、管理者のユーザー名、パスワード、および実名を指定します。

5. 『サーバー 1 の構成』を続けます。

---

## サーバー 1 の構成

ワークステーションから、以下の手順を実行してください。

1. Web ブラウザーを開始します。

2. ブラウザーの「アドレス」フィールドに、 `http://server_1` を入力します。  
`server_1` は、サーバー 1 として指定したマシンの実際のホスト名または IP アドレスを指します。
3. 「構成および管理フォーム」をクリックします。
4. 管理者名およびパスワードを入力します。「構成および管理」フォームがブラウザでオープンされます。
5. 「サーバー構成」 → 「要求処理」 → 「要求経路指定」をクリックします。
6. 「前に挿入」ラジオ・ボタン、および既存のワイルドカード・マッピング・ルールの索引値を選択して、新しいワイルドカード・マッピング・ルールを既存のルールの前に挿入します。
7. 「アクション」ドロップダウン・ボックスから「プロキシー」を選択します。
8. 「URL 要求テンプレート」フィールドに `/*` を入力します。
9. 「サーバー IP アドレスまたはホスト名」フィールドに、HTTP 要求を宛先変更するサイトのホスト名を入力します。この値の前に `http://` を付けます。
10. 「実行依頼」をクリックします。
11. 「前に挿入」ラジオ・ボタン、およびステップ 6 で作成されたマッピング・ルールの索引値を選択して、「構成および管理」フォームにアクセスできるマッピング・ルールを作成します。
12. 「アクション」ドロップダウン・ボックスから「パス」を選択します。
13. 「URL 要求テンプレート」フィールドに `/pub/*` を入力します。
14. 「構成および管理」フォームの場所を入力します。
  - Caching Proxy が Linux または UNIX マシンにある場合には、`/opt/ibm/edge/cp/server_root/pub/en_US/*` を「サーバー IP アドレスまたはホスト名」フィールドに入力します。
  - Caching Proxy が Windows マシンにある場合には、`"C:%Program Files%IBM%edge%cp%server_root%pub%en_US%*"` を「サーバー IP アドレスまたはホスト名」フィールドに入力します。
15. 「実行依頼」をクリックします。
16. 構成フォームの最上部で、「サーバーの再始動」アイコンをクリックします。
17. 『Caching Proxy ネットワークのテスト』を続けます。

---

## Caching Proxy ネットワークのテスト

ワークステーションから、以下の手順を実行してください。

1. Web ブラウザーを開始します。
2. ブラウザーの「アドレス」フィールドに `http://server_1` を入力します。サーバー 2 からの HTML ページはサーバー 1 を経由してプロキシーに入り、Web ブラウザーに引き渡されます。
3. 「構成および管理」フォームにアクセスするには、ブラウザの「アドレス」フィールドに `http://server_1/pub/` を入力します。「構成および管理」フォームのホーム・ページが表示されます。



## 第 16 章 Load Balancer ネットワークの構築

図 15 は、3 つのローカル接続ワークステーションを持つ基本的な Load Balancer ネットワークを示しています。このネットワークでは、Dispatcher コンポーネントの MAC 転送方式を使用して、Web サーバー間の Web トラフィックのロード・バランスを取っています。構成は、他の任意の TCP またはステートレス UDP アプリケーション・トラフィックのロード・バランスを取る場合と同じです。

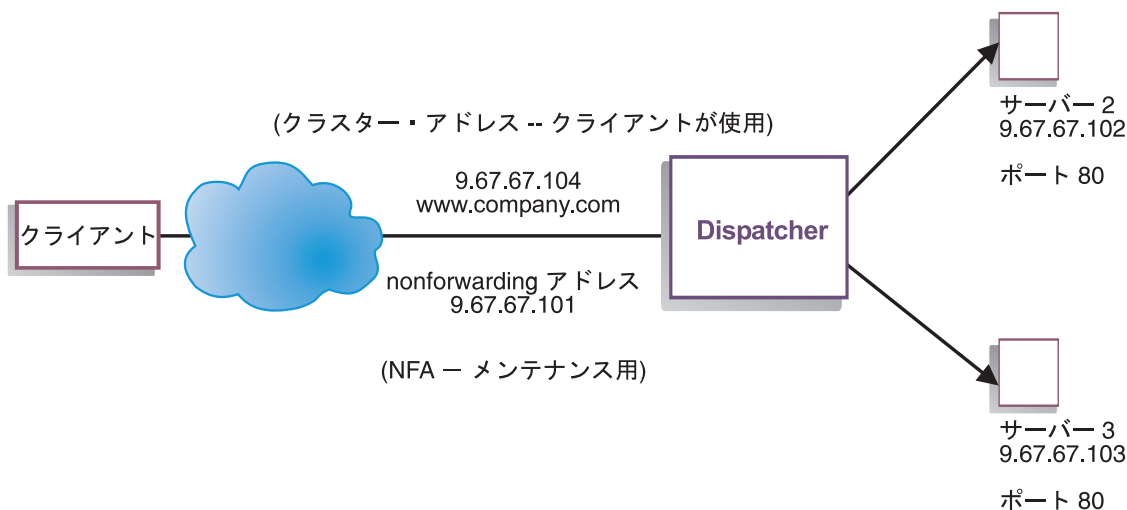


図 15. Load Balancer デモンストレーション・ネットワーク

注: AIX、HP-UX、Linux、または Solaris バージョンの Dispatcher の場合、いずれかの Web サーバー・ワークステーションに配置されている Dispatcher を使用して、ワークステーションを 2 つ使用するだけで構成を完了することができます。これは連結構成を表します。

### ワークフロー

Load Balancer ネットワークを構築するには、以下の順序で手順を実行します。

1. 必須コンピューター・システムおよびソフトウェアの確認
2. ネットワークの構成
3. Dispatcher の構成
4. Load Balancer ネットワークのテスト

### 必須コンピューター・システムおよびソフトウェアの確認

以下のコンピューター・システムおよびソフトウェア・コンポーネントが必要です。

- Dispatcher として機能するコンピューター・システム。このシステムには、実 IP アドレス 1 つと、ロード・バランスを取るためのアドレスが 1 つ必要です。
- Web サーバーとして機能する 2 つのコンピューター・システム。各 Web サーバーには IP アドレスが 1 つずつ必要です。

## ネットワークの構成

1. すべて同じ LAN セグメント上に配置するようにワークステーションをセットアップします。3 つのマシンの間のネットワーク・トラフィックが、ルーターまたはブリッジを一切通過する必要がないようにします。
2. 3 つのワークステーションのネットワーク・アダプターを構成します。この例では、以下のネットワーク構成を想定しています。

ワークステーション	名前	IP アドレス
1	server1.company.com	9.67.67.101
2	server2.company.com	9.67.67.102
3	server3.company.com	9.67.67.103
ネットマスク = 255.255.255.0		

各ワークステーションには、標準のイーサネット・ネットワーク・インターフェース・カードが 1 つだけ装備されています。

3. server1.company.com が server2.company.com と server3.company.com の両方を ping できるようにします。
4. server2.company.com と server3.company.com が server1.company.com を ping できるようにします。
5. 2 つの Web サーバー (サーバー 2 およびサーバー 3) の上でコンテンツが同じであることを確認します。これを行うには、データを両方のワークステーションに複製するか、あるいは NFS、AFS<sup>®</sup>、または DFS<sup>™</sup> などのファイル共有システムを使用します。また、サイトに合ったその他の方法を使用することもできます。
6. server2.company.com および server3.company.com にある Web サーバーが操作可能であることを確認します。Web ブラウザーを使用して <http://server2.company.com> および <http://server3.company.com> から直接ページを要求します。
7. この LAN セグメント用に別の有効な IP アドレスを取得します。このアドレスは、サイトにアクセスしたいクライアントに与えるアドレスです。この例では、この情報は次のようになります。  
Name= www.company.com  
IP=9.67.67.104
8. www.company.com のトラフィックを受け入れるように 2 つの Web サーバー・ワークステーションを構成します。

server2.company.com および server3.company.com にあるループバック・インターフェースに www.company.com の別名を追加してください。

- AIX の場合:

```
ifconfig lo0 alias www.company.com netmask 255.255.255.0
```

- Solaris 7 の場合:

```
ifconfig lo0:1 www.company.com 127.0.0.1 up
```

9. ループバック・インターフェースの別名割り当ての結果として作成された可能性があるエクストラ経路を削除します。

これで、2 つの Web サーバー・ワークステーションに必要なすべての構成ステップが完了しました。

---

## Dispatcher の構成

Dispatcher があれば、コマンド行、構成ウィザード、またはグラフィカル・ユーザー・インターフェース (GUI) を使用して構成を作成できます。

注: パラメーター値は、英字で入力する必要があります。例外は、ホスト名およびファイル名のパラメーター値だけです。

### コマンド行を使用した構成

コマンド行を使用する場合は、以下のステップに従ってください。

1. Dispatcher で `dsserver` を開始します。

- AIX、Linux、HP-UX、または Solaris の場合は、`dsserver` コマンドを root ユーザーとして実行します。
- Windows プラットフォームの場合、`dsserver` はサービスとして実行され、自動的に開始されます。

2. Dispatcher の `executor` 機能を開始します。

```
dscontrol executor start
```

3. クラスタ・アドレスを Dispatcher 構成に追加します。

```
dscontrol cluster add www.company.com
```

4. http プロトコル・ポートを Dispatcher 構成に追加します。

```
dscontrol port add www.company.com:80
```

5. Web サーバーをそれぞれ Dispatcher 構成に追加します。

```
dscontrol server add www.company.com:80:server2.company.com
```

```
dscontrol server add www.company.com:80:server3.company.com
```

6. クラスタ・アドレスに対するトラフィックを受け入れるようにワークステーションを構成します。

```
dscontrol executor configure www.company.com
```

7. Dispatcher の `manager` 機能を開始します。

```
dscontrol manager start
```

これで、Dispatcher は、サーバー・パフォーマンスに基づいてロード・バランシングを行うようになります。

8. Dispatcher の `advisor` 機能を開始します。

```
dscontrol advisor start http 80
```

これで Dispatcher はクライアント要求が失敗 Web サーバーに送信されないようにします。

ローカル接続サーバーの基本構成はこれで完了です。

## 構成ウィザードを使用した構成

構成ウィザードを使用する場合は、以下のステップにしたがってください。

1. Dispatcher で dsserver を開始します。
  - AIX、HP-UX、Linux、または Solaris の場合、以下のコマンドをルート・ユーザーとして実行します。

```
dsserver
```
  - Windows システムの場合、dsserver はサービスとして実行され、自動的に開始されます。
2. Dispatcher のウィザード機能 dswizard を開始します。

このウィザードは、Dispatcher コンポーネントの基本構成を作成するプロセスを段階的に案内します。このウィザードはネットワークについての確認を行い、Dispatcher のクラスターのセットアップをガイドします。このクラスターによって、サーバー・グループのトラフィックのロード・バランシングが行われます。

構成ウィザードには、以下のパネルがあります。

- ウィザードの紹介
- これから行われること
- セットアップの準備
- 構成するホストの選択 (必要な場合)
- クラスターの定義
- ポートの追加
- サーバーの追加
- advisor の開始
- サーバー・マシンのセットアップ

## グラフィカル・ユーザー・インターフェース (GUI) を使用した構成

GUI を開始するには、以下のステップに従ってください。

1. dsserver プロセスが実行されていることを確認します。
  - AIX、HP-UX、Linux、または Solaris の場合、以下のコマンドをルートとして実行します。

```
dsserver
```
  - Windows システムの場合、dsserver はサービスとして実行され、自動的に開始されます。
2. 次に、以下のいずれかを行います。
  - AIX、Linux、HP-UX、または Solaris の場合、lbadm in を入力します。
  - Windows システムの場合、「スタート」>「プログラム」>「IBM WebSphere」>「Edge Components」>「IBM Load Balancer」>「Load Balancer」をクリックします。



---

## Load Balancer ネットワークのテスト

1. Web ブラウザーから、ロケーション `http://www.company.com` に移動してページが表示されることを確認します。
2. このページを Web ブラウザーに再ロードします。
3. コマンド `dscontrol server report www.company.com:80:` を実行します。2 つのサーバーを加算した合計接続数の欄が **2** になることを確認します。



---

## 特記事項

### 初版 (2004 年 11 月)

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。

IBM Corporation  
ATTN: Software Licensing  
11 Stanwix Street  
Pittsburgh, PA 15222-9183  
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 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が、その測定値が、一般に利用可能なシステムのもの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。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

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。

この情報をソフトコピーでご覧になっている場合は、写真やカラーの図表は表示されない場合があります。

---

## 商標

以下は、IBM Corporation の商標です。

- AFS
- AIX
- DFS

- IBM
- iSeries™
- RS/6000
- SecureWay®
- Tivoli
- ViaVoice
- WebSphere

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 Inside (ロゴ)、MMX 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。

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。







Printed in Japan

GC88-7036-00



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