

Oracle9iAS Discoverer Plus/Viewer for WindowsNT/2000

構成ガイド

リリース 4.1

2001 年 7 月

部品番号 : J04368-01

ORACLE®

Oracle9iAS Discoverer Plus/Viewer for WindowsNT/2000 構成ガイド, リリース 4.1

部品番号 : J04368-01

原本名 : Oracle9iAS Discoverer Plus and Viewer Configuration Guide, Release 4.1 for Windows NT/2000

原本部品番号 : A90287-01

Copyright © 2001, Oracle Corporation. All rights reserved.

Printed in Japan.

制限付権利の説明

プログラム（ソフトウェアおよびドキュメントを含む）の使用、複製または開示は、オラクル社との契約に記された制約条件に従うものとします。著作権、特許権およびその他の知的財産権に関する法律により保護されています。

当プログラムのリバース・エンジニアリング等は禁止されています。

このドキュメントの情報は、予告なしに変更されることがあります。オラクル社は本ドキュメントの無謬性を保証しません。

*オラクル社とは、Oracle Corporation（米国オラクル）または日本オラクル株式会社（日本オラクル）を指します。

危険な用途への使用について

オラクル社製品は、原子力、航空産業、大量輸送、医療あるいはその他の危険が伴うアプリケーションを用途として開発されておりません。オラクル社製品を上述のようなアプリケーションに使用することについての安全確保は、顧客各位の責任と費用により行ってください。万一かかる用途での使用によりクレームや損害が発生いたしましても、日本オラクル株式会社と開発元である Oracle Corporation（米国オラクル）およびその関連会社は一切責任を負いかねます。当プログラムを米国国防総省の米国政府機関に提供する際には、『Restricted Rights』と共に提供してください。この場合次の Notice が適用されます。

Restricted Rights Notice

Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication, and disclosure of the Programs, including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are "restricted computer software" and use, duplication, and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software - Restricted Rights (June, 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

このドキュメントに記載されているその他の会社名および製品名は、あくまでその製品および会社を識別する目的にのみ使用されており、それぞれの所有者の商標または登録商標です。

目次

はじめに	ix
------------	----

1 Oracle9iAS Discoverer の概要

1.1	Oracle9iAS Discoverer および Oracle9i Application Server	1-2
1.2	Oracle9iAS Discoverer のアーキテクチャ	1-2
1.2.1	Discoverer Plus の 3 層アーキテクチャ	1-2
1.2.2	Discoverer Viewer の 4 層アーキテクチャ	1-3
1.2.3	クライアント層 (Discoverer Plus および Viewer)	1-4
1.2.3.1	クライアント層および Discoverer Plus	1-4
1.2.3.2	クライアント層および Discoverer Viewer	1-5
1.2.4	HTTP サーバー層 (Discoverer Viewer のみ)	1-5
1.2.4.1	サーブレットとは	1-5
1.2.4.2	サーブレット・エンジンとは	1-5
1.2.4.3	Discoverer Viewer Servlet とは	1-5
1.2.5	Discoverer Services 層 (Discoverer Plus および Viewer)	1-6
1.2.5.1	Discoverer Plus Java アプレットおよび HTML ページ (Discoverer Plus のみ)	1-6
1.2.5.2	Session コンポーネント (Discoverer Plus および Viewer)	1-7
1.2.5.3	Locator コンポーネント (Discoverer Plus および Viewer)	1-7
1.2.5.4	Preference コンポーネント (Discoverer Plus および Viewer)	1-7
1.2.6	データベース層 (Discoverer Plus および Viewer)	1-8
1.2.7	Discoverer Services コンポーネントをインストールするサーバー・マシン	1-8
1.2.7.1	HTTP サーバー・マシン	1-8
1.2.7.2	アプリケーション・サーバー・マシン	1-9
1.2.7.3	マスター Discoverer サーバー・マシン	1-9

1.3	Oracle9iAS Discoverer の動作について	1-10
1.3.1	Discoverer Plus のプロセス	1-10
1.3.2	Discoverer Viewer のプロセス	1-11
1.3.2.1	Discoverer Viewer Servlet の HTTP 要求の解釈方法	1-12
1.4	スケーラビリティおよび負荷均衡 (ロード・バランシング)	1-12

2 Oracle9iAS Discoverer の実行、メンテナンスおよびサポート

2.1	Oracle9iAS Discoverer の実行	2-2
2.1.1	Discoverer Plus の実行	2-2
2.1.2	Discoverer Viewer の実行	2-5
2.2	Discoverer NT サービスの使用	2-6
2.2.1	NT サービスとは	2-6
2.2.2	Discoverer NT サービスとは	2-6
2.2.3	Discoverer NT サービスの起動方法	2-7
2.2.4	Discoverer NT サービスの手動による起動および停止方法	2-7
2.2.5	Discoverer NT サービスを手動による起動に変更する方法	2-8
2.2.6	サービス・エラーの表示	2-8
2.3	Discoverer Services コンポーネントの登録および登録解除	2-8
2.3.1	OAD が実行中で応答できることの確認	2-9
2.3.2	1 つのマシンに登録されたコンポーネントの表示	2-9
2.3.3	ネットワークに登録されたすべてのコンポーネントの表示	2-10
2.3.4	Windows の「スタート」メニューからのコンポーネントの登録および登録解除	2-11
2.3.5	コマンドラインからの登録および登録解除	2-11
2.3.5.1	Preferences および Session コンポーネントの OAD への登録	2-11
2.3.5.2	Preferences および Session コンポーネントを OAD から登録解除するには	2-12
2.4	エンド・ユーザー Preferences の編集	2-13
2.4.1	エンド・ユーザー Preferences とは	2-13
2.4.2	エンド・ユーザー Preferences の設定方法	2-13
2.5	シャットダウン	2-14
2.5.1	個々のサーバーのシャットダウン	2-15
2.5.1.1	個々のサーバーを段階的にシャットダウンするには	2-15
2.5.1.2	個々のサーバーを即時シャットダウンするには	2-16
2.5.2	Discoverer システム全体のシャットダウン	2-16
2.5.2.1	Discoverer システムを段階的にシャットダウンするには	2-16
2.5.2.2	Discoverer システムを即時シャットダウンするには	2-17

2.6	サーバー・マシンの追加および削除	2-18
2.6.1	その他の Discoverer サーバー・マシンの追加	2-18
2.6.2	その他の Discoverer サーバー・マシンを完全に削除するには	2-18
2.7	Discoverer Viewer JavaScript 拡張ユーザー・インタフェースの利用	2-19
2.8	よくある質問	2-21

3 Oracle9iAS Discoverer の複数のマシンへのインストール

3.1	Oracle9iAS Discoverer のデフォルト・インストールと 分散インストールの違い	3-2
3.1.1	Oracle9iAS Discoverer のデフォルト・インストール	3-2
3.1.2	Oracle9iAS Discoverer の分散インストール	3-2
3.2	分散インストールの実行	3-4
3.2.1	デフォルトのマスター Discoverer サーバー・マシンへの接続の確認	3-4
3.2.2	マスター Discoverer サーバー・マシンの名前の確認	3-4
3.2.3	その他の Discoverer サーバー・マシンへの Discoverer のインストール	3-5
3.2.4	Discoverer Services の pref.txt ファイルの編集 (Discoverer Plus および Viewer)	3-8
3.2.4.1	pref.txt ファイル	3-8
3.2.4.2	pref.txt ファイルを編集する理由	3-9
3.2.4.3	負荷均衡 (ロード・バランシング)	3-9
3.2.4.4	負荷均衡 (ロード・バランシング) の例	3-10
3.2.4.5	pref.txt ファイルの編集方法	3-11
3.2.5	tnsnames.ora ファイルの編集 (Discoverer Plus および Viewer)	3-13
3.2.5.1	tnsnames.ora ファイルとは	3-13
3.2.5.2	tnsnames.ora ファイルを編集する理由	3-13
3.2.5.3	tnsnames.ora ファイルのコピー方法	3-14
3.2.5.4	tnsnames.ora ファイルの編集方法	3-14
3.2.6	disco4iv.xml ファイルの編集 (Discoverer Viewer のみ)	3-14
3.2.6.1	disco4iv.xml ファイルとは	3-14
3.2.6.2	disco4iv.xml ファイルを編集する理由	3-14
3.2.6.3	disco4iv.xml ファイルの編集方法および変更箇所	3-15
3.2.7	その他の Discoverer サーバーへの接続の確認	3-16

4 URL を使用した Oracle9iAS Discoverer の自動的な起動

4.1	Oracle9iAS Discoverer を起動するための URL の書式	4-2
4.2	Discoverer Plus の URL 引数	4-3
4.3	Discoverer Viewer の URL 引数	4-6

5 Discoverer Viewer のカスタマイズ

5.1	カスタマイズした Web アプリケーション向けの Discoverer Viewer の使用	5-2
5.1.1	一般的な外観のカスタマイズ	5-2
5.1.2	アプリケーションのカスタマイズ	5-2
5.1.3	HTML の作成方法	5-2
5.1.4	特定の Discoverer Application の作成に必要な HTML 作成のステップ	5-3
5.1.4.1	ステップ 1 ブラウザによる URL の送信	5-3
5.1.4.2	ステップ 2 XML の作成	5-4
5.1.4.3	ステップ 3 XSL の適用	5-5
5.1.4.4	ステップ 4 HTML の作成	5-5
5.2	カスタマイズの例	5-6
5.2.1	カスタマイズに必要なファイル	5-6
5.2.2	クイック・カスタマイズ	5-6
5.2.2.1	クイック・カスタマイズの属性	5-7
5.2.2.2	クイック・カスタマイズの手順	5-7
5.2.2.3	style.xml ファイルの編集例	5-8
5.2.3	フル・カスタマイズ	5-8
5.2.3.1	背景	5-9
5.2.3.2	前提条件	5-10
5.2.3.3	ファイル構造	5-10
5.2.3.3.1	disco4iv.xml	5-10
5.2.3.3.2	page_layouts.xml	5-10
5.2.3.3.3	gui_components.xml	5-11
5.2.3.3.4	style.xml	5-11
5.2.3.3.5	functions.xml	5-11
5.2.3.3.6	errors.xml	5-11
5.2.3.3.7	scripts.xml	5-11
5.2.3.3.8	render_table.xml	5-11
5.2.3.3.9	viewer_config.xml	5-12
5.2.3.4	Servlet のプログラミング・インタフェース	5-12
5.2.3.4.1	Servlet からのデータ受信	5-12
5.2.3.4.2	Servlet へのデータ送信	5-12
5.2.3.5	外観のカスタマイズ - XML/XSL の例	5-13
5.2.3.5.1	手順 1 - XML ファイル	5-13
5.2.3.5.2	手順 2 - XSL ファイル	5-13
5.2.3.5.3	手順 3 - XML+XSL=HTML	5-14
5.2.3.5.4	手順 4 - XSL ファイルのカスタマイズ	5-16

5.2.4	Video Stores Demo - 代替ユーザー・インタフェースの例	5-18
5.2.4.1	インストール内容	5-19
5.2.4.2	セットアップの手順	5-19
5.2.4.3	Web サーバーの設定	5-20
5.2.4.4	Video Stores demo の実行	5-20
5.2.4.5	video.xml ファイルの変更	5-21
5.2.5	その他のカスタマイズの例	5-22
5.2.5.1	自動ログイン	5-22
5.2.5.1.1	手順 1: 新規テンプレートを page_layouts.xml に追加する	5-22
5.2.5.1.2	手順 2: ユーザー名、パスワードおよびデータベースを指定する	5-22
5.2.5.1.3	手順 3: disco4iv.xml を変更する	5-23
5.2.6	カスタマイズの概要	5-23

6 Oracle9/AS Discoverer における SSL サポート

6.1	詳細情報	6-2
6.2	SSL および Discoverer	6-2
6.3	SSL について、および SSL を使用する理由	6-2
6.4	SSL の動作方法	6-3
6.5	公開鍵暗号化技術について	6-3
6.6	SSL の認証および証明書	6-3
6.7	次に必要なステップ	6-4
6.7.1	Discoverer Plus での SSL の構成方法	6-4
6.7.2	Discoverer Viewer での SSL の構成方法	6-4
6.8	Discoverer Plus で SSL を使用するための構成	6-5
6.8.1	Discoverer Plus の SSL のサポート	6-5
6.8.2	SSL を使用するための Visibroker Gatekeeper の構成方法	6-5
6.8.3	SSL を使用するための Visibroker Gatekeeper のインストール	6-5
6.8.4	SSL の証明書の取得	6-6
6.8.4.1	証明書情報の指定	6-8
6.8.4.2	秘密鍵のパスワードの指定	6-8
6.8.4.3	出力ファイル名の指定	6-9
6.8.4.4	鍵のサイズの指定	6-9
6.8.4.5	S/MIME 証明書チェーンの使用	6-9
6.8.5	SSL の証明書のインストール	6-10
6.8.6	x509cert.jar ファイルのインストール	6-12
6.8.7	必要なダイナミック・リンク・ライブラリ (DLL)・ファイルのインストール	6-13

6.8.8	Discoverer Plus の起動ページでの SSL の有効化	6-13
6.8.9	ファイアウォール内での SSL の使用	6-14
6.9	HTTPS を使用した Discoverer Viewer の構成	6-14
6.9.1	httpd.conf ファイルの編集による HTTPS の設定	6-14
6.9.2	disco4iviewer.properties ファイルの編集による HTTPS の設定	6-15
6.9.3	disco4iv.xml ファイルの編集による HTTPS の設定	6-16
6.9.4	viewer_config.xml ファイルの編集による HTTPS の設定	6-16
6.9.5	Discoverer Viewer での SSL の有効化	6-17

7 Discoverer Plus for Firewall Support の構成

7.1	用語	7-2
7.2	詳細情報	7-2
7.3	はじめに	7-2
7.4	インターネット・ファイアウォールについて	7-3
7.4.1	インターネット・ファイアウォール	7-3
7.4.2	Demilitarized Zone (DMZ) とは	7-3
7.4.3	保護ポート	7-4
7.5	Visibroker Gatekeeper について	7-5
7.5.1	Visibroker Gatekeeper	7-5
7.5.2	Discoverer Plus に Visibroker Gatekeeper が必要な理由	7-5
7.5.3	必要な Visibroker Gatekeeper のバージョン	7-6
7.6	Discoverer 用の Gatekeeper 構成の選択	7-6
7.6.1	ファイアウォールの構成について	7-6
7.6.2	DMZ 構成の例	7-7
7.6.3	Gatekeeper 構成の選択	7-8
7.6.3.1	IIOProxying の構成	7-8
7.6.3.2	HTTP Tunnelling の構成	7-8
7.7	IIOProxying 構成での Visibroker Gatekeeper の使用	7-9
7.7.1	概要	7-9
7.7.2	ファイアウォールを Discoverer Services と分離させない Gatekeeper の構成	7-10
7.7.3	内部のファイアウォールの内側に配置した Discoverer Server を使用した Gatekeeper の構成	7-11
7.8	HTTP Tunnelling 構成での Visibroker の使用	7-12
7.8.1	概要	7-12
7.8.2	ファイアウォールを Discoverer Services と分離させない Gatekeeper の構成	7-13

7.8.3	内部のファイアウォールの内側に配置した Discoverer Server を使用した Gatekeeper の構成	7-14
7.8.4	永続的な HTTP 接続	7-15
7.9	Discoverer Client の構成	7-15
7.9.1	Discoverer Client の接続順序	7-15
7.9.2	特定の接続方法を使用する方法	7-16
7.10	Visibroker Gatekeeper の構成	7-17
7.10.1	Visibroker Gatekeeper の実行場所	7-18
7.10.2	HTTP サーバーでの Visibroker Gatekeeper の実行	7-18
7.10.3	別のサーバーでの Visibroker Gatekeeper の実行	7-19
7.10.4	Visibroker Gatekeeper ポートの構成	7-19
7.10.5	Visibroker Gatekeeper ポートの変更	7-20
7.10.6	Visibroker Gatekeeper からのログ情報の取得	7-21
7.10.7	Gatekeeper の前面で動作する NAT デバイスの構成	7-22

A Oracle9/AS Discoverer および Oracle Applications

A.1	Discoverer Viewer の接続ダイアログでの「Applications ユーザー」チェックボックスの表示の指定	A-1
A.2	Discoverer Plus の接続ダイアログでの「Applications ユーザー」チェックボックスの表示の指定	A-2
A.3	Oracle Applications ユーザーのための NT レジストリ設定	A-3

B Discoverer Viewer の構成ファイル

B.1	viewer_config.xml の構成設定値	B-1
B.2	disco4iv.xml 内の構成設定値	B-3

C Oracle9/AS Discoverer レジストリの設定値

C.1	pref.txt 内のレジストリ設定値	C-1
C.2	その他の Registry 設定値	C-7
C.3	レジストリに使用するメモリーの割当て	C-7

索引

はじめに

Oracle9iAS Discoverer へようこそ

このマニュアルは、Discoverer Plus および Discoverer Viewer を Oracle9i Application Server の一部としてインストールした後の、構成およびカスタマイズの方法を説明します。

Oracle9iAS Discoverer のインストール方法は説明していません。インストールについては、『Oracle9i Application Server インストレーション・ガイド』を参照してください。

このマニュアルを読む前に、HTTP サーバーおよびデータベースを理解しておくことをお勧めします。

最新情報については、このマニュアルの他に Oracle9iAS Discoverer のリリース・ノートを参照してください。

対象読者

このマニュアルは、Discoverer 管理者を対象としています。

表記規則

このマニュアルで使用している用語の意味は、次のとおりです。

- EUL (End User Layer) は、データベースと Discoverer 間のメタデータ・インタフェースを表します。
- <iSUITES_HOME> は、Oracle HTTP Server (Apache ベース) と Jserv サブレット・エンジン・ソフトウェアが Oracle RDBMS とともにインストールされているディレクトリを表します。たとえば、<iSUITES_HOME> は D:\oracle\iSuites になります。
- <ORACLE_806_HOME> は、Oracle 8.0.6 RSF (Required Support Files) に依存する製品 (たとえば、Oracle9iAS Discoverer) がインストールされるディレクトリを表します。デフォルトの Oracle ホームが存在しない場合は、<ORACLE_806_HOME> が

<iSUITES_HOME> の他に作成されます。たとえば、<ORACLE_806_HOME> は D:\orant¥6iserver になります。

次の表に、このマニュアルで使用しているその他の規則を示します。

表記	意味
.	例示中の縦に並んだ点は、例示する必要のない情報の省略を表します。
...	文またはコマンド内の横に並んだ点は、その文またはコマンドに必要な情報の省略を表します。
太字テキスト	テキスト中、太字で示された部分は、コマンド名、メニュー名、ファイル名、キーボードのキーまたはその他の選択事項を表します。
イタリック・テキスト	イタリックで表記されたテキストは、関数等の変数を表します。
<>	ユーザーが指定する名前を、このカッコで囲んで表します。
[]	ユーザーが選択可能な（または選択しなくてもよい）オプション句を、このカッコで囲んで表します。
「メニュー名」→「コマンド」	一連の選択を表します。たとえば、メニューを選択し、そのメニューのコマンドを実行します。
固定幅フォント (Courier text)	この書体のテキストはタイプ入力するコマンドラインを表します。

Oracle9iAS Discoverer は、Windows NT および Windows 2000 (Service Pack1) でサポートされることに注意してください。このマニュアルの手順は、(特に記載のない限り) Windows NT を使用していることを前提に記述されています。Windows 2000 を使用している場合は、Windows NT の「スタートメニュー」オプションを、Windows 2000 の相当する「スタートメニュー」オプションに置き換えてください。たとえば、「サービス」ダイアログを表示するには、次のようにします。

- Windows NT では、「スタート」→「設定」→「コントロール パネル」の順に選択して、「サービス」アイコンをダブルクリックします。
- Windows 2000 では、「スタート」→「プログラム」→「管理ツール」→「サービス」を選択します。

Oracle9iAS Discoverer の概要

Oracle9iAS Discoverer は、データ分析のためのビジネス・インテリジェンス・ツールです。受賞実績のある Discoverer のユーザー・インタフェースにより、データベースのデータにアクセスし、データの分析ができます。

Oracle9iAS Discoverer には、2 種類の製品があります。

- Discoverer Plus は、受賞実績のある、Windows 向け製品の Discoverer Desktop Edition のインターネット版です。Discoverer を使用すれば、データベースの複雑な概念を理解しなくても、企業のデータベースからデータを取得して分析できます。Discoverer に備わっているウィザード形式のダイアログやメニューを使用すれば、容易にデータを取得および分析でき、ビジネスの決断に役立ちます。
- Discoverer Viewer は、Discoverer Plus および Discoverer Desktop Edition を使用して作成されたワークブックを閲覧するためのツールです。また、データベースの出力を Web サイトやポータルに統合できます。この製品はカスタマイズが容易で、特定の Web サイトの外観に合わせたり、Web 用に Discoverer のカスタム・アプリケーションを構築することができます。Discoverer Viewer はパフォーマンスの最適化が行われており、ネットワークの通信量を最小限に抑えるよう設計されています。

1.1 Oracle9iAS Discoverer および Oracle9i Application Server

Oracle9iAS Discoverer は、Oracle9i Application Server の一部としてインストールされます。デフォルト・インストールでは、1 つのマシンに Discoverer のコンポーネントがすべてインストールされます。ただし、コンポーネントを複数のマシンに分けてインストールしてパフォーマンスの低下を防ぐこともできます。詳細は、[第 3 章「Oracle9iAS Discoverer の複数のマシンへのインストール」](#)を参照してください。

1.2 Oracle9iAS Discoverer のアーキテクチャ

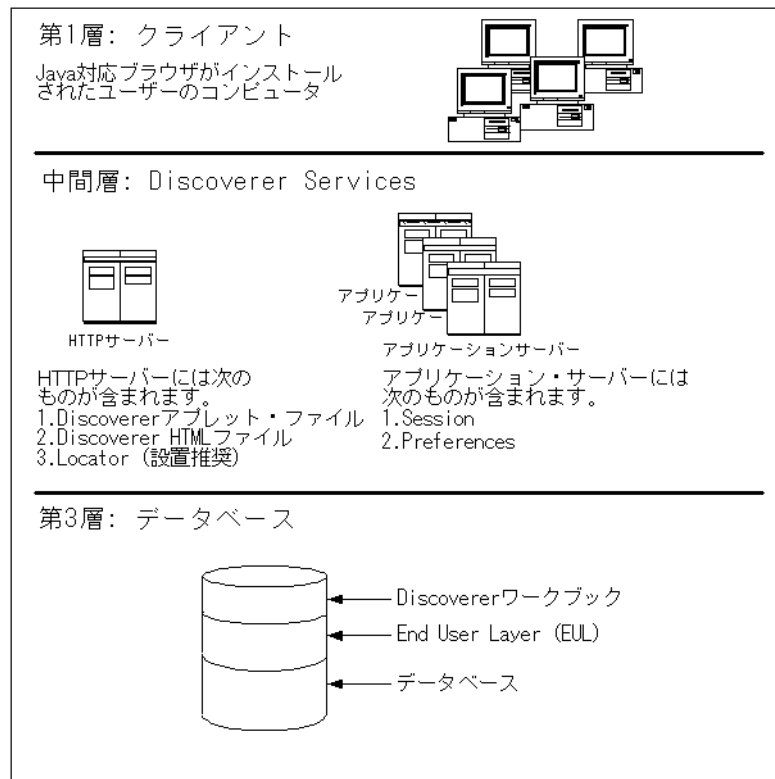
Oracle9iAS Discoverer では、多層アーキテクチャを採用しています。このアーキテクチャは、Web 環境の分散化特性を活用しています。1 台のマシンに Oracle9iAS Discoverer すべての層をインストールすることも可能ですが、パフォーマンスを最大限に高めるため、各層を複数のマシンに分散することをお薦めします。

1.2.1 Discoverer Plus の 3 層アーキテクチャ

Discoverer Plus では、3 層アーキテクチャを採用しています。

- クライアント
- Discoverer Services
- データベース

図 1-1 Discoverer Plus の 3 層アーキテクチャ

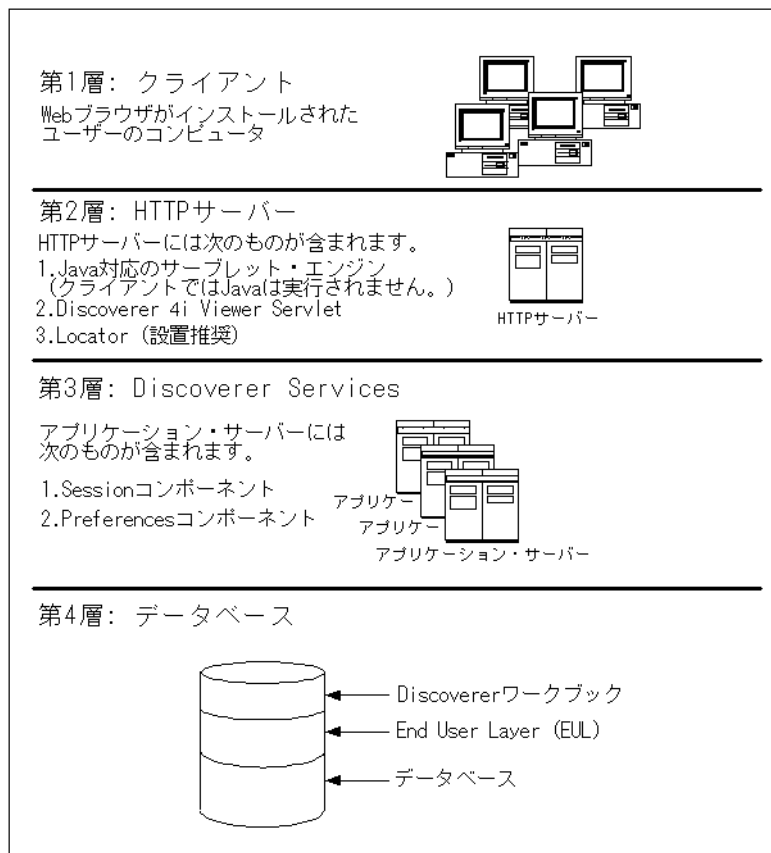


1.2.2 Discoverer Viewer の 4 層アーキテクチャ

Discoverer Viewer では、4 層アーキテクチャを採用しています。

- クライアント
- HTTP サーバー
- Discoverer Services
- データベース

図 1-2 Discoverer Viewer の 4 層アーキテクチャ



1.2.3 クライアント層 (Discoverer Plus および Viewer)

Oracle9iAS Discoverer アーキテクチャの第1層はクライアント、つまりユーザーのコンピュータのことです。

クライアント・マシンにはいかなる設定作業もインストール作業も必要ありません。また、このマシンがデータベースにアクセスする必要もありません。

1.2.3.1 クライアント層および Discoverer Plus

Discoverer Plus の場合、クライアント・マシンの唯一の要件は、Microsoft Internet Explorer や Netscape Navigator など Java 対応の Web ブラウザを、Oracle JInitiator と一緒に作動さ

せることです。ユーザーは、Discoverer 管理者が提供する URL にアクセスすることによって Discoverer にログインして実行し、データ分析を行えます。

1.2.3.2 クライアント層および Discoverer Viewer

Discoverer Viewer の場合、クライアント・マシンの唯一の要件は、Microsoft Internet Explorer や Netscape Navigator など HTML を実行できる Web ブラウザを作動させることです。ユーザーは、Discoverer 管理者が提供する URL にアクセスすることによって Discoverer Viewer に接続して実行し、データを表示できます。

1.2.4 HTTP サーバー層（Discoverer Viewer のみ）

Discoverer Viewer の場合、Discoverer Viewer Servlet がインストールされている場所が HTTP サーバー層（および、特に HTTP サーバーのサーブレット・エンジン）になります。HTTP サーバーで Java 対応のサーブレット・エンジンを作動させると、クライアント層で Java を実行する必要がなくなります。

複数のマシンにコンポーネントをインストールしている場合、Discoverer Services 層のコンポーネント（Locator コンポーネント）は通常、HTTP サーバー層と同じマシンにインストールされます。

1.2.4.1 サーブレットとは

サーブレットは、サーバー・マシンで実行する Java コードのモジュールで構成されており、クライアント・マシンから送信される HTTP 要求に答えます。サーブレットを使用することにより、クライアント側の処理を最小限に抑えられます。

1.2.4.2 サーブレット・エンジンとは

サーブレット・エンジンは HTTP サーバー・ソフトウェアの一部（またはプラグイン）であり、サーブレットの実行環境です。サーブレット・エンジンには Java Virtual Machine (JVM) が組み込まれており、Java Servlet API を実装します。

1.2.4.3 Discoverer Viewer Servlet とは

Discoverer Viewer Servlet は、XML/XSL プロセッサを使用して、HTTP サーバーから要求された HTML ページを送信します。

Discoverer Viewer Servlet および XML/XSL プロセッサは、HTTP サーバーのサーブレット・エンジン内に組み込まれます。

1.2.5 Discoverer Services 層 (Discoverer Plus および Viewer)

Discoverer Services 層は、Discoverer 管理者によりインストールおよび管理される Discoverer のアーキテクチャの一部です。

Discoverer Services 層は、次のコンポーネントで構成されます。

- Discoverer Plus Java アプレットおよび HTML ページ (Discoverer Plus のみ)
- Locator コンポーネント (Discoverer Plus および Viewer)
- Session コンポーネント (Discoverer Plus および Viewer)
- Preferences コンポーネント (Discoverer Plus および Viewer)

Discoverer Java アプレットと HTML ページは、HTTP サーバー・マシンにインストールします。Locator も通常 HTTP サーバーにインストールします。その他のコンポーネントは 1 台以上のアプリケーション・サーバーにインストールします。Discoverer Services は、1 つのサーバー・マシンにインストールすることも、複数のマシンに分散してインストールすることもできます。

さらに、次の CORBA (Common Object Request Broker Architecture) コンポーネントが VisiBroker からインストールされます。

- Object Activation Daemon (OAD)
- SmartAgent

これらの CORBA コンポーネントは、ユーザーが Discoverer セッションの開始を要求したときに、新規コンポーネントのアクティブ化およびロケーション・サービスの提供を行います。

1.2.5.1 Discoverer Plus Java アプレットおよび HTML ページ (Discoverer Plus のみ)

Discoverer Java アプレットにより、データベース内のデータの分析に必要な Discoverer Plus のユーザー・インタフェースおよび機能を使用できるようになります。アプレットは、クライアント・マシンにインストールされた Java 対応のブラウザで実行されます。

HTML ページは、ユーザーが Discoverer Plus の URL の位置にログインする際に参照する Web ページです。

注意： Discoverer Java アプレットは、ユーザーが初めてログインするときに、初期化されてユーザーのマシンにキャッシュされます。それ以降にユーザーがログインした場合は、Discoverer Java アプレットはローカル・キャッシュから実行されるため、ダウンロードは行われません。

1.2.5.2 Session コンポーネント (Discoverer Plus および Viewer)

Session コンポーネントにより、Discoverer Plus または Discoverer Viewer を実行しているクライアント・マシンとデータベースがリンクされます。Session コンポーネントには、アプリケーション・ロジックのすべてが含まれており、データベースへの接続、ワークブックを開く、といった Discoverer の全動作を実行します。

Session コンポーネントがインストールされているマシンでは、複数のクライアントのセッションを同時に実行できます。複数のマシンに Session コンポーネントをインストールし、各マシンで複数のセッションを同時に実行させると、全体的なパフォーマンスを向上させることができます。Session コンポーネントを複数のマシンに分散することで、可用性も向上します。あるマシンが作動していないときは、他のマシンでセッション要求が処理されます。

Session コンポーネントは、Discoverer Services の構成に組み込まれた、HTTP サーバー・マシンやアプリケーション・サーバー・マシンなどの NT サーバー・マシンで実行できます。

1.2.5.3 Locator コンポーネント (Discoverer Plus および Viewer)

Locator コンポーネントの役割は、Discoverer Plus または Discoverer Viewer のいずれかを実行するクライアント・マシンからのセッション要求を受けること、次に使用可能なアプリケーション・サーバー・マシンを検出すること、新規セッションを開始すること、および新規セッションの参照情報をクライアントに返すことです。クライアントは、いったんこの参照情報を受信すると、セッションと直接通信します。Locator は次の要求を待ちます。

Locator コンポーネントは、HTTP サーバー・マシンにインストールします。HTTP サーバー・マシンとは、HTTP サーバー・ソフトウェアを実行しているマシンのことです。

Discoverer Services に対しては、ネットワーク上に Locator コンポーネントが 1 つ必要です。

この構造では、いったんセッションが開始されると、HTTP サーバー・マシンはクライアントとセッションの通信には関与しなくなります。この構造の大きな利点は、セッションが開始された後、HTTP サーバー・マシンを通過する通信量が新たに増加することがないため、セッションの数が増えても HTTP サーバー・マシンのパフォーマンスが影響を受けないことです。HTTP サーバー・マシンまたは Locator コンポーネントに障害が発生しても、クライアントはアプリケーション・サーバーと直接通信しているため、ユーザー・セッションが中断されないという利点もあります。

Locator コンポーネントは、要求されたセッションを、どのアプリケーション・サーバーで開始させるかを判断するため、使用可能なアプリケーション・サーバー間の負荷を分散させる上でも役立ちます。Locator のセッションの割当て方法を、最大のパフォーマンスが得られるように指定することもできます。

1.2.5.4 Preference コンポーネント (Discoverer Plus および Viewer)

Preferences コンポーネントは、すべての Oracle9iAS Discoverer (Discoverer Plus および Discoverer Viewer) ユーザーの作業環境設定を、1 つの場所でできるようにします。

Discoverer Services は、格納された作業環境設定に従って、動作の特定の部分を決めます。

Discoverer Services には、1 つの Preferences コンポーネントをインストールします。

様々なコンポーネントが様々なマシンで実行されるような分散環境では、インストールする Preferences コンポーネントを 1 つのみにすることが重要です。Preferences コンポーネントは、コンポーネントが実行される場所を問わず、全コンポーネントに対して一貫性のある作業環境設定を行います。Preferences コンポーネントは、作業環境の格納および読取りを Windows レジストリとの間で行います。Preferences コンポーネントは、Windows NT で作動するサーバーにインストールする必要があります。

1.2.6 データベース層（Discoverer Plus および Viewer）

Discoverer のアーキテクチャのデータベース層は、次のもので構成されています。

- ユーザーが必要とするデータ
- データをわかりやすく表示する End User Layer（EUL）
- データの表示および分析に使用される Discoverer ワークブック

EUL を作成および管理するには、Discoverer Administration Edition を使用します。

注意： ユーザーが Discoverer を使用してデータを分析する前に、データベースには Discoverer 4.1 で作成された EUL が含まれている**必要があります**。Oracle Discoverer Administration Edition リリース 4.1 以上を使用して、EUL を作成または更新してください。

1.2.7 Discoverer Services コンポーネントをインストールするサーバー・マシン

Discoverer Services ソフトウェア・コンポーネントのそれぞれをサーバーにインストールする必要があります。Discoverer のアーキテクチャを構成するサーバー・マシンには、次の 3 つがあります。

- HTTP サーバー・マシン
- アプリケーション・サーバー・マシン
- マスター Discoverer サーバー・マシン

それぞれのサーバーを別のマシンにインストールする必要はありません。実際に、Oracle9i のデフォルト・インストールでは、HTTP サーバー、アプリケーション・サーバーおよびマスター Discoverer サーバーがすべて同じマシンにインストールされます。

1.2.7.1 HTTP サーバー・マシン

HTTP サーバー・マシンとは、HTTP サーバー・ソフトウェアを実行しているマシンを指し、Locator コンポーネントはこのマシンにインストールされます。Discoverer 環境に加える必要がある HTTP サーバー・マシンは 1 台のみです。

1 台のマシンに異なる Discoverer Services ソフトウェア・コンポーネントをインストールすると、そのマシンは、HTTP サーバー・マシンおよびアプリケーション・サーバー・マシンの両方の機能を果たします。

1.2.7.2 アプリケーション・サーバー・マシン

アプリケーション・サーバー・マシンには、Preferences コンポーネントおよび Session コンポーネントがインストールされます。任意の数のアプリケーション・サーバー・マシンを Discoverer 環境に加えることが可能です。

1 台のマシンに異なる Discoverer Services ソフトウェア・コンポーネントをインストールすると、そのマシンは、アプリケーション・サーバー・マシンおよび HTTP サーバー・マシンの両方の機能を果たします。

複数のマシンに異なる Discoverer Services ソフトウェア・コンポーネントを分散してインストールした場合、Preferences コンポーネントではなく Session コンポーネントをすべてのマシンにインストールしてください。このようにインストールした場合、各マシンがアプリケーション・サーバーとなり、それぞれのマシンで個別にセッションを開始できます。

Preferences コンポーネントをインストールできるのは、1 台のアプリケーション・サーバー・マシンのみです。このマシンをマスター Discoverer サーバー・マシンと呼びます。それ以外のアプリケーション・サーバー・マシンをその他の Discoverer サーバー・マシンと呼びます。

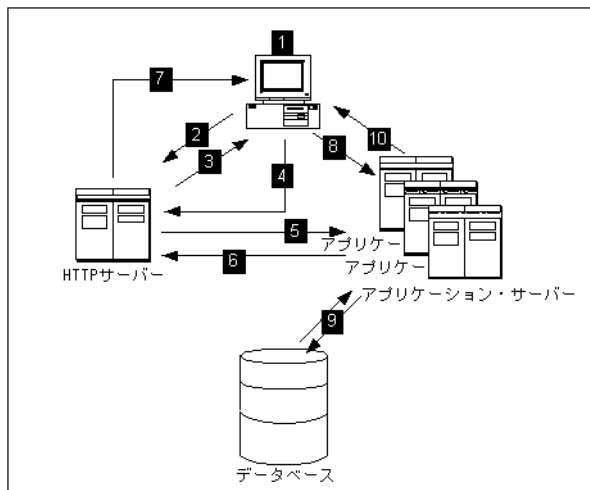
1.2.7.3 マスター Discoverer サーバー・マシン

複数のアプリケーション・サーバー・マシンに Session コンポーネントをインストールした場合、そのうちいずれか 1 台をマスター Discoverer サーバーとして指定する必要があります。マスター Discoverer サーバーには、Session コンポーネントのみでなく、Preferences コンポーネントもインストールしてください。Preferences コンポーネントがインストールされるのは、マスター Discoverer サーバーのみです。

1.3 Oracle9iAS Discoverer の動作について

1.3.1 Discoverer Plus のプロセス

図 1-3 Discoverer Plus のプロセス

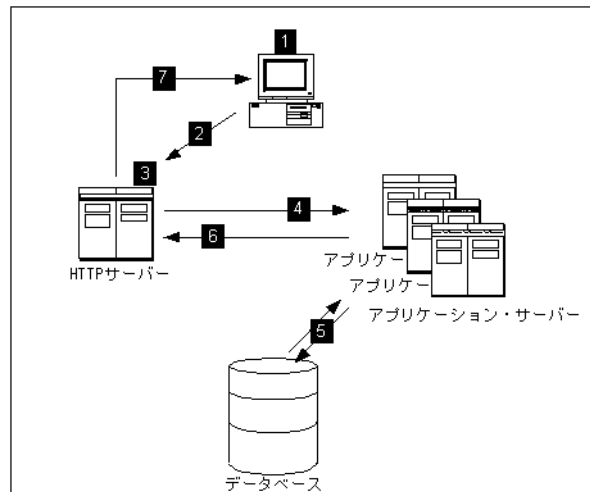


1. クライアント・コンピュータ上の Web ブラウザを起動し、HTTP サーバーの URL を入力します。
2. Web ブラウザが HTTP サーバーの URL にアクセスします。
3. HTTP サーバーが HTML ページおよび Discoverer アプレットを取り出し、クライアント・コンピュータに送信します。
4. Discoverer アプレットが Locator (HTTP サーバーへのインストールを推奨) を検索します。
5. Locator がアプリケーション・サーバー・マシン上で Session コンポーネントを起動し、Discoverer セッションを開始します。
6. Session コンポーネントが参照情報を作成して Locator に渡します。
7. Locator が参照情報をクライアント・コンピュータに渡します。
8. クライアント・コンピュータがこの参照情報を使用して、Discoverer セッションと直接コンタクトします。HTTP サーバーへのアクセスはこの時点で終了します。

9. Discoverer セッションが要求を送信し、データベースからデータを取得します。
10. Discoverer セッションがクライアント・マシン上の Web ブラウザにデータを伝送します。

1.3.2 Discoverer Viewer のプロセス

図 1-4 Discoverer Viewer のプロセス



1. クライアント・コンピュータ上の Web ブラウザを起動し、HTTP サーバーの URL を入力します。
2. Web ブラウザが HTTP サーバーの URL にアクセスします。
3. Discoverer Viewer Servlet が HTTP 要求を解釈します。
4. Locator がアプリケーション・サーバー・マシン上で Session コンポーネントを起動し、Discoverer セッションが開始します。
5. Discoverer セッションが要求を送信し、データベースからデータを取得します。
6. Discoverer セッションが Discoverer Viewer Servlet にデータを伝送します。
7. Discoverer Viewer Servlet が HTML ページを生成し、ブラウザに送信します。

1.3.2.1 Discoverer Viewer Servlet の HTTP 要求の解釈方法

Discoverer Viewer Servlet は、クライアントのブラウザから送信された HTTP 要求を解釈し、必要に応じて Discoverer Services を呼び出します。

XML 形式のサーバー応答が Servlet により生成され、XML/XSL プロセッサに送信されます。

XML/XSL プロセッサでは、XML を XSL 構成ファイルと組み合わせます。XSL 構成ファイルは、ユーザー・インタフェースの表示を定義し、HTML などの出力を生成してブラウザに送り返します。

Discoverer Viewer Servlet は、主にアプリケーション・ロジック層と表示ロジック層の 2 層で構成されています。

アプリケーション・ロジック層は、HTTP 要求の処理や Discoverer の状態を管理し、Discoverer Server の論理状態を表示する XML および任意のクライアントに渡すデータを生成します。

表示ロジック層はアプリケーション・ロジック層で生成される XML を利用し、XSL を使用して任意の形式（HTML、Microsoft Excel、XML など）で応答を生成します。XSL ファイルを編集することにより表示ロジックをカスタマイズし、ユーザー・インタフェースの表示を変更することができます。詳細は、[第 5 章「Discoverer Viewer のカスタマイズ」](#)を参照してください。

1.4 スケーラビリティおよび負荷均衡（ロード・バランシング）

Discoverer Plus および Discoverer Viewer は、Oracle9i Application Server 上で実行するために自動的にシングルマシン・インストールでインストールされます。ただし、これらのコンポーネントは複数のマシンに配置できるように設計されています。詳細は、[第 3 章「Oracle9iAS Discoverer の複数のマシンへのインストール」](#)を参照してください。

Discoverer Services を複製して複数のマシンでセッションを開始できるようにし、負荷を分散させることもできます。

この方法はボトルネックが解消されるため、多数の Discoverer ユーザーにとって便利です。複数の Discoverer セッションは、1 台のマシンで実行することも、必要に応じて複数のマシンに分散して実行することもできます。

Discoverer Viewer Servlet も同様に、分散して実行することができます。

Oracle9iAS Discoverer の実行、メンテナンス およびサポート

この章では、Oracle9iAS Discoverer の実行、メンテナンスおよびサポートを説明します。
説明するトピックは次のとおりです。

- [Oracle9iAS Discoverer の実行](#)
- [Discoverer NT サービスの使用](#)
- [Discoverer Services コンポーネントの登録および登録解除](#)
- [エンド・ユーザー Preferences の編集](#)
- [シャットダウン](#)
- [サーバー・マシンの追加および削除](#)
- [Discoverer Viewer JavaScript 拡張ユーザー・インタフェースの利用](#)
- [よくある質問](#)

2.1 Oracle9iAS Discoverer の実行

Oracle9i Application Server のインストール（『Oracle9i Application Server インストール・ガイド』を参照）が完了すると、Discoverer Plus および Oracle Discoverer Viewer が実行できるようになります。

注意： ユーザーが Discoverer を使用してデータを分析する前に、データベースには Discoverer 4.1 で作成された EUL が含まれている必要があります。Oracle Discoverer Administration Edition リリース 4.1 以上を使用して、EUL を作成または更新してください。

2.1.1 Discoverer Plus の実行

Discoverer Plus を実行するには、次の手順を完了してください。

1. ブラウザ・ウィンドウを起動し、次のガイドに従ってアドレスを入力します。

`http://hostname.domain/discwb4/html/japanese/welcome.htm`

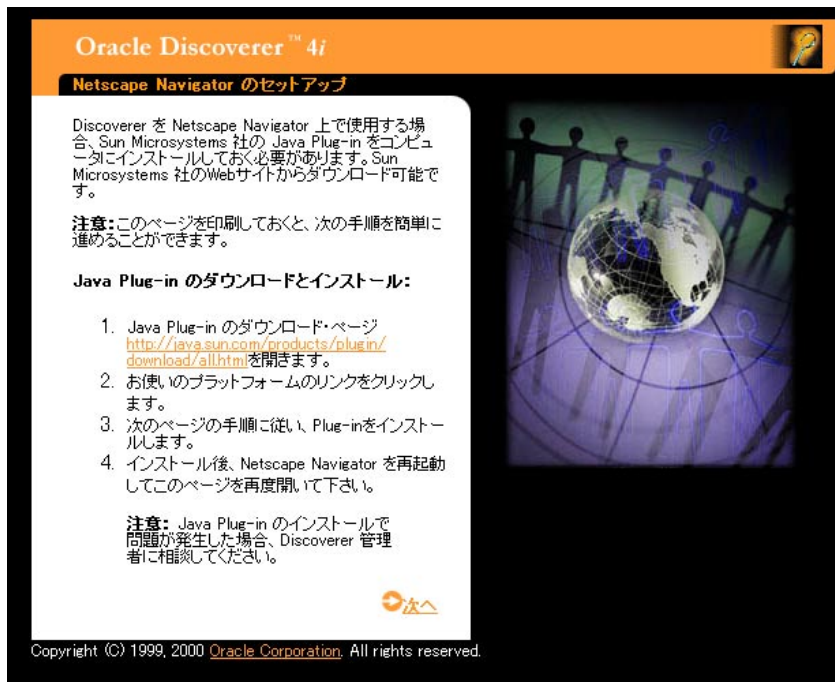
`hostname.domain` は、HTTP サーバーをインストールした Web サーバー・マシンを表します。Web サーバー名を指定するときには、完全修飾ホスト名（必要に応じてポート番号を含む）を使用する必要があります。

例：`http://machinename.oracle.com:7777/discwb4/html/japanese/welcome.htm`



2. 言語および地域を指定して、「次へ」をクリックします。

Netscape ブラウザを使用している場合には、Oracle JInitiator をダウンロードしてインストールするようプロンプトされます。Oracle JInitiator のダウンロードおよびインストールがプロンプトされない場合には、次の手順に進んでください。



Oracle Discoverer™ 4i

Netscape Navigator のセットアップ

Discoverer を Netscape Navigator 上で使用する
場合、Sun Microsystems 社の Java Plug-in をコンピュ
ータにインストールしておく必要があります。Sun
Microsystems 社のWebサイトからダウンロード可能で
す。

注意: このページを印刷しておく、次の手順を簡単に
進めることができます。

Java Plug-in のダウンロードとインストール:

1. Java Plug-in のダウンロード・ページ
[http://java.sun.com/products/plugin/
download/all.html](http://java.sun.com/products/plugin/download/all.html)を開きます。
2. お使いのプラットフォームのリンクをクリックし
ます。
3. 次のページの手順に従い、Plug-inをインスト
ールします。
4. インストール後、Netscape Navigator を再起動
してこのページを再度開いて下さい。

注意: Java Plug-in のインストールで
問題が発生した場合、Discoverer 管理
者に相談してください。

次へ

Copyright (C) 1999, 2000 Oracle Corporation. All rights reserved.

3. 前述の画面が表示された場合は、画面上の指示に従って Oracle JInitiator をダウンロー
ドおよびインストールします。

Oracle JInitiator のインストールが終了した場合、または Oracle JInitiator のインストー
ルが不要な場合には、次の画面が表示されます。



4. Discoverer Plus を起動するイメージ画像「Click to Start」をクリックします。
Discoverer Plus アプレットがロードされると、ログイン画面が表示されます。



5. 「ユーザー名」、「パスワード」および「データベース」接続文字列を入力します。
6. 「接続」をクリックします。

Discoverer Plus が実行されます。

2.1.2 Discoverer Viewer の実行

Discoverer Viewer を実行するには、次の手順を完了してください。

1. ブラウザ・ウィンドウを起動し、次のガイドに従ってアドレスを入力します。

`http://hostname.domain/discoverer4i/viewer`

`hostname.domain` は、HTTP サーバーをインストールした Web サーバー・マシンを表します。この変数は、完全修飾ホスト名（必要に応じてポート番号を含みます）に置換する必要があります。

例 : `http://machinename.oracle.com:7777/discoverer4i/viewer`

Discoverer Viewer が起動し、次のログイン画面が表示されます。



ORACLE
Discoverer™ 4i Viewer

ユーザー名

パスワード

データベース

言語

接続

2. 「ユーザー名」、「パスワード」および「データベース」接続文字列を入力します。
3. Oracle Applications ユーザーとして Oracle Applications データベースに接続している場合には、「Oracle Applications ユーザー」チェックボックスをオンにしてください（オプション）。

前述の接続画面に「Oracle Applications ユーザー」チェックボックスが表示されない場合は、[付録 A.1 「Discoverer Viewer の接続ダイアログでの「Applications ユーザー」チェックボックスの表示の指定」](#)を参照してください。

4. 「接続」をクリックします。

Discoverer Viewer が実行されます。

2.2 Discoverer NT サービスの使用

2.2.1 NT サービスとは

NT サービスは、起動時に NT によって自動的に起動される特別なプロセスです。NT サービスにより、ネットワークやリモート・アクセス手順などの様々なコンピュータ操作、Server および Telephony プロセスが起動されます。

2.2.2 Discoverer NT サービスとは

Discoverer をインストールすると、Discoverer NT サービスが Oracle Discoverer4i と呼ばれる自動 NT サービスとして作成されます。NT マシンが起動されるたびに、Discoverer NT サービスにより Locator コンポーネントおよび CORBA サービスが自動的に開始されます。Discoverer NT サービスを使用すると、サービスの実行を停止することでこれらのコンポーネントをすべて停止することもできます。

Discoverer Services Locator コンポーネントおよび CORBA サービス・コンポーネントを 1 つの NT サービスとして起動する主な利点は、サービス (Locator および CORBA サービス) の実行を続けている間、マシンにログインしている必要はないということです。各マシン上でそれぞれのコンポーネントを個別に起動した場合は、管理者でマシンにログインしている必要があります。ログオフすると、コンポーネントは自動的に停止されます。

Discoverer NT サービスでは Discoverer のインストール時に選択されたサーバーのタイプに基づいて、適切なコンポーネントが実行されます。選択されたサーバーと、Discoverer NT サービスによって起動されるコンポーネントを次の表に示します。

表 2-1 Discoverer NT サービス・コンポーネント

サーバー	Locator	CORBA サービス	OAD
マスター Discoverer サーバー		X	X
HTTP サーバー	X	X	
その他の Discoverer サーバー		X	X
シングル・マシン・インストール	X	X	X

2.2.3 Discoverer NT サービスの起動方法

Discoverer をインストールすると、Discoverer NT サービスの「スタートアップ」プロパティが「自動」に設定されます。この設定では、マシンが起動されるたびにサービスが自動的に開始します。ただし、サービスが自動的に開始された後は、いつでも手動で停止および再起動できます。

Discoverer NT サービスの「スタートアップ」プロパティを「手動」に変更することも可能です。この設定では、マシンを起動してもサービスが自動的に開始されません。特定の順序での起動と停止を必要とする特別なサービスがネットワークで必要な場合に有効です。他のサービスを起動した後に、適当な時点で Discoverer NT サービスを手動で起動できます。

2.2.4 Discoverer NT サービスの手動による起動および停止方法

1. Windows の「スタート」メニューから「設定」→「コントロール パネル」の順に選択し、「サービス」アイコンをダブルクリックし、「サービス」ダイアログ・ボックスを表示します。

「サービス」ダイアログ・ボックスには、「OracleDiscoverer4i」が表示されます。



「状態」：サービスが起動され、実行中であるかどうかが表示されます。実行中でなければこの列は空白です。

「スタートアップ」：サービスが手動で起動されたのか、自動的に起動されたのかが表示されます。Discoverer NT サービスは、自動的に起動するように設定されます。

2. Discoverer NT サービスを停止するには、ダイアログ・ボックスでサービスを選択し、「停止」をクリックします。次に、ダイアログ・ボックスを閉じます。
3. サービスを起動するには、ダイアログ・ボックスでサービスを選択し、「開始」をクリックします。次に、ダイアログ・ボックスを閉じます。

注意：エンド・ユーザーが Oracle Application に対して Discoverer を実行する場合は、Locator を起動する前に必ず Listener をシャットダウンします。

2.2.5 Discoverer NT サービスを手動による起動に変更する方法

1. Windows の「スタート」メニューから「設定」→「コントロール パネル」の順に選択し、「サービス」アイコンをダブルクリックします。
「サービス」ダイアログ・ボックスが表示されます。
2. 「サービス」ダイアログ・ボックスで「OracleDiscoverer4i」を選択します。
3. 「スタートアップ」ボタンをクリックします。
Oracle Discoverer 4i の「サービス」ダイアログ・ボックスが表示されます。
4. 「サービス」ダイアログ・ボックスの「スタートアップの種類」セクションで「手動」を選択します。
5. 「OK」をクリックして「サービス」ダイアログ・ボックスを閉じます。
6. 「閉じる」をクリックして「サービス」ダイアログ・ボックスを閉じます。

2.2.6 サービス・エラーの表示

NT イベント・ログでは、Discoverer NT サービスの起動時に発生するエラーが表示されます。NT イベント・ログを表示するには、「スタート」→「プログラム」→「管理ツール」→「イベント ビューア」の順に選択します。

Discoverer では、HTTP サーバー上で Locator を実行している際に発生するエラーの内部ログも保持されます。Locator エラーのリストを表示するには、HTTP サーバーのルート・ディレクトリ内の <i>SUITES_HOME>%apache%\apache\%htdocs%\DISCWB4\applet ディレクトリ（または、Discoverer Client を他のディレクトリにインストールしている場合はそのディレクトリ）にある Locator.log ファイルを開きます。

2.3 Discoverer Services コンポーネントの登録および登録解除

Discoverer Services コンポーネントは、Visibroker OAD（Object Activation Daemon）に登録されます。コンポーネントが登録されるのは、インストール時です。マシンをシャットダウンまたは削除する場合は、Discoverer Services コンポーネントを登録解除してください。

コンポーネントの登録および登録解除は、Windows の「スタート」メニューまたはコマンドラインを使用して行います。

1 つのマシン、またはネットワーク全体の登録済みコンポーネントを表示することもできます。

コンポーネントを登録および登録解除して登録済みコンポーネントを表示するには、VisiBroker OAD が実行中で応答できる必要があります。

2.3.1 OAD が実行中で応答できることの確認

OAD が実行中であるかどうかわからない場合

1. [Ctrl] と [Alt] を押しながら [Del] を押し、「タスク マネージャ」 ボタンをクリックして Windows のタスク・マネージャを開きます。
2. 「プロセス」タブを選択します。
3. 「プロセス」タブを選択して、oad.exe が実行中で応答できることを確認します。

OAD が実行中でない場合

1. Windows の「スタート」メニューから「設定」→「コントロールパネル」の順に選択し、「サービス」アイコンをダブルクリックし、「サービス」ダイアログ・ボックスを表示します。
2. 「OracleDiscoverer4i」を選択し、「開始」ボタンをクリックします。
3. OAD が開始された場合は、「閉じる」をクリックして「サービス」ダイアログ・ボックスを閉じます。

OAD は実行中だが応答しない場合

1. Windows の「スタート」メニューから「設定」→「コントロールパネル」の順に選択し、「サービス」アイコンをダブルクリックし、「サービス」ダイアログ・ボックスを表示します。
2. 「OracleDiscoverer4i」を選択します。
3. 「停止」ボタンをクリックしてから、しばらくした後「開始」ボタンをもう一度クリックします。
4. OAD が開始された場合は、「閉じる」をクリックして「サービス」ダイアログ・ボックスを閉じます。

2.3.2 1 つのマシンに登録されたコンポーネントの表示

1. OAD が実行中であることを確かめます。
2. コマンド・プロンプトを開きます（「スタート」→「プログラム」→「コマンドプロンプト」の順に選択）。
3. コマンド・プロンプトで次のように入力し、リターン・キーを押します。

```
oadutil list
```

注意： oadutil コマンドを実行すると、そのマシンに登録されたコンポーネントのリストのみが表示されます。Discoverer マシンのネットワークに登録されたすべてのコンポーネントが表示されるわけではありません。



```

D:\orant\DISCWB4\util>oadutil list
oadutil list: located 3 record(s)

Implementation #1:
-----
repository_id = IDL:DCISessionManager:1.0
object_name = 4ivOracleDiscovererSession4
reference_data =
path_name = D:\orant\DISCWB4\dis4ws.exe
activation_policy = UNSHARED_SERVER
args = (length=4)[-session; 4ivOracleDiscovererSession4; -preference; 4ivOracleDiscovererPreferences4; ]
env = NONE

Implementation #2:
-----
repository_id = IDL:DCICollector:1.0
object_name = 4ivOracleDiscovererCollector4
reference_data =
path_name = D:\orant\DISCWB4\dis4col.exe
activation_policy = SHARED_SERVER
args = (length=2)[-collector; 4ivOracleDiscovererCollector4; ]
env = NONE

Implementation #3:
-----
repository_id = IDL:DCICORBAInterface:1.0
object_name = 4ivOracleDiscovererPreferences4
reference_data =
path_name = D:\orant\DISCWB4\dis4pr.exe
activation_policy = SHARED_SERVER
args = (length=2)[-preference; 4ivOracleDiscovererPreferences4; ]
env = NONE

D:\orant\DISCWB4\util>

```

この例のコンポーネントは、Discoverer インスタンス名「4iv」で登録されています。このマシンに登録されているコンポーネントは、Preferences と Session の 2 つです。ログを記録するための Collector ファイルも登録されています。

その他の Discoverer サーバーでこのコマンドを使用した場合は、Session コンポーネントのみが登録されているはずです。

2.3.3 ネットワークに登録されたすべてのコンポーネントの表示

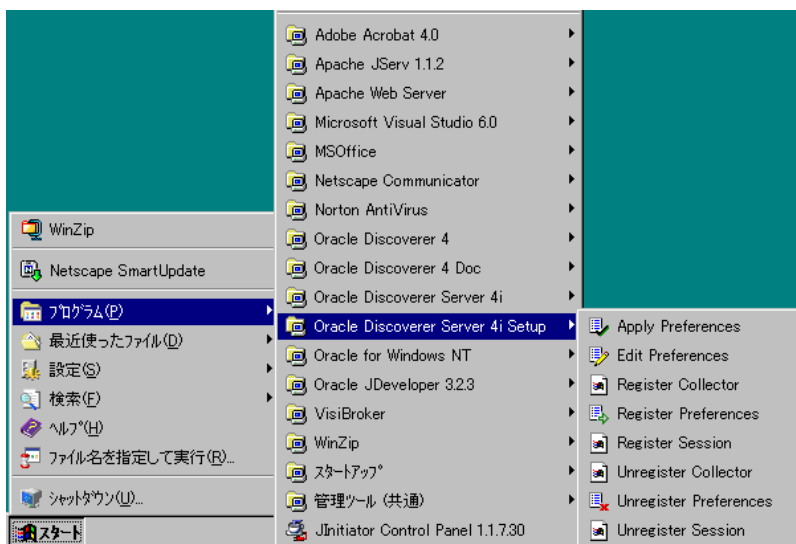
1. OAD が実行中であることを確かめます。
2. コマンド・プロンプトを開きます（「スタート」→「プログラム」→「コマンドプロンプト」の順に選択）。
3. コマンド・プロンプトで osfind と入力し、リターン・キーを押します。

ネットワークで実行されているすべてのコンポーネントのリストが表示されます。

2.3.4 Windows の「スタート」メニューからのコンポーネントの登録および登録解除

コンポーネントの登録または登録解除を行うマシン上で、次の操作を実行します。

1. Windows の「スタート」メニューから「プログラム」→「Oracle Discoverer Server 4i Setup」の順に選択します。



2. 登録または登録解除するコンポーネントを選択します。
確認画面が表示されます。

2.3.5 コマンドラインからの登録および登録解除

Windows の「スタート」メニューから、「プログラム」→「コマンドプロンプト」を使用します。コンポーネントの登録にはバッチ・スクリプトが使用できます。このバッチ・スクリプトは、コンポーネントがインストールされたマシンの <ORACLE_806_HOME>\discwb4\util ディレクトリにあります。

2.3.5.1 Preferences および Session コンポーネントの OAD への登録

Preferences および Session コンポーネントは、OAD に登録する必要があります。OAD では、クライアントの接続が行われるたびに新規セッションが開始されます。OAD への Preferences および Session コンポーネントの登録に役立つスクリプトが用意されています。

このスクリプト名は次のとおりです。

- registerSession.bat
- registerPreference.bat

コマンドラインから Session を起動するには、次のようにします。

1. MS-DOS ウィンドウを開きます。
2. ディレクトリを <ORACLE_806_HOME>%discwb4%util に変更します。
3. registerSession.bat と入力して、リターン・キーを押します。

コマンドラインから Preferences を起動するには、次のようにします。

1. MS-DOS ウィンドウを開きます。
2. ディレクトリを <ORACLE_806_HOME>%discwb4%util に変更します。
3. registerPreference.bat と入力し、リターン・キーを押します。

2.3.5.2 Preferences および Session コンポーネントを OAD から登録解除するには

OAD による Preferences および Session コンポーネントの登録解除に役立つスクリプトが用意されています。

OAD から Preferences コンポーネントを登録解除するには、次のようにします。

1. MS-DOS ウィンドウを開きます。
2. ディレクトリを <ORACLE_806_HOME>%discwb4%util に変更します。
3. unregisterPreference.bat と入力し、リターン・キーを押します。

OAD から Session コンポーネントを登録解除するには、次のようにします。

1. MS-DOS ウィンドウを開きます。
2. ディレクトリを <ORACLE_806_HOME>%discwb4%util に変更します。
3. unregisterSession.bat と入力し、リターン・キーを押します。

2.4 エンド・ユーザー Preferences の編集

2.4.1 エンド・ユーザー Preferences とは

エンド・ユーザー Preferences により、Oracle9iAS Discoverer のデフォルトの動作が制御されます。

Preferences ファイル (pref.txt) は、すべてのエンド・ユーザーに適用されるデフォルト設定を制御します。新しいセッションが開始されると、pref.txt ファイルの中の設定が有効になります。

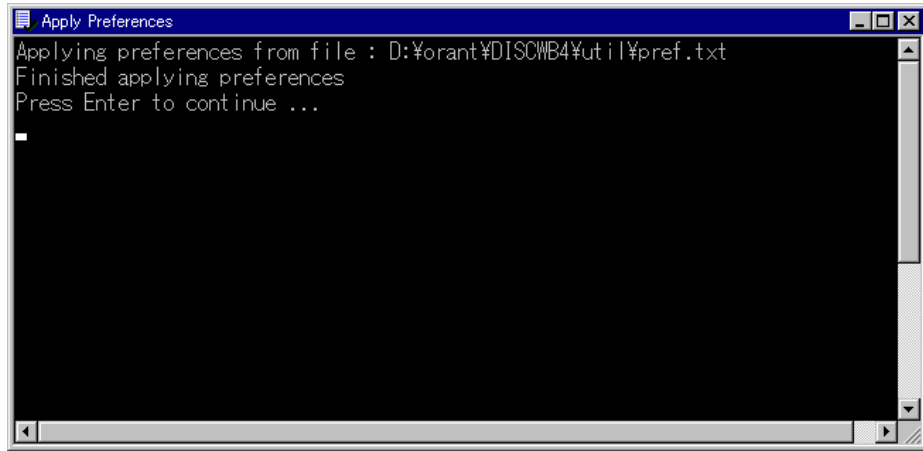
場合によって、エンド・ユーザーは Discoverer Plus の「オプション」ダイアログからこれらの設定を変更できます。個々の作業環境は、データベースおよびユーザー ID の一意の組合せとして各ユーザーごとに保存され、ユーザーが新しいセッションを開始するたびにロードされます。したがって、ユーザーが別のクライアント・マシンからログインした場合でも、個々の設定は有効です。個々のエンド・ユーザーの作業環境設定は、Windows NT レジストリ・エディタで表示できます。pref.txt 内のレジストリ設定値の詳細は、[付録 C「Oracle9iAS Discoverer レジストリの設定値」](#)を参照してください。

注意： マシンの IP アドレスを追加するなど、Preferences の編集を行う場合、Preferences コンポーネントをシャットダウンする必要はありません。したがって、編集中でも、ユーザーは作業を継続できます。

2.4.2 エンド・ユーザー Preferences の設定方法

1. Windows の「スタート」メニューから「プログラム」→「Oracle Discoverer Server 4i Setup」→「Edit Preferences」の順に選択します。
2. Pref.txt ファイルがメモ帳に表示されます。
3. pref.txt ファイル内の項目を編集します。
4. ファイルを保存し、メモ帳を終了します。
5. Windows の「スタート」メニューから「プログラム」→「Oracle Discoverer Server 4i Setup」→「Apply Preferences」の順に選択します。

新しい作業環境が保存されたことを示す確認ウィンドウが表示されます。ファイルにエラーがあると考えられる場合は、カレント・ディレクトリの error.txt ファイルをチェックします。



6. 続行する場合は、[Enter] キーを押します。

Windows NT レジストリの Preferences が更新されます。これで Discoverer Services の Preferences が設定されました。

2.5 シャットダウン

メンテナンス、ハードウェアの設定変更またはその他の作業を行う場合は、サーバー・マシンをオフラインにしてください。これにより、メンテナンス作業中に、ユーザーがそのサーバーでセッションを開始してしまうことを防ぐことができます。Discoverer Server をシャットダウンする方法には、次の 2 通りがあります。

- 個々のサーバー・マシンをシャットダウンする方法
- Discoverer システム全体をシャットダウンする方法

注意： Discoverer Preferences を編集する場合は、シャットダウンする必要はありません。

シャットダウンするマシンで Discoverer Services Locator コンポーネントまたは Preferences コンポーネントが実行されている場合は、Discoverer システム全体をシャットダウンする必要があります。どのサーバーで Session コンポーネントがオンになっているかにかかわらず、これらのコンポーネントはセッションの作成時に使用されます。したがって、これらのコンポーネントのうちどちらかでも停止してしまうと、Discoverer システム全体に影響が及びます。

2.5.1 個々のサーバーのシャットダウン

サーバーをシャットダウンする方法は次の 2 通りです。

- 準備ができた現行ユーザーから段階的に切断していく方法
- 現行ユーザーの Discoverer セッションをただちに強制終了させる方法

2.5.1.1 個々のサーバーを段階的にシャットダウンするには

サーバーを段階的にシャットダウンすると、現行ユーザーはセッションを完了できますが、新規セッションを開始できなくなります。

1. シャットダウンするマシン上で、Windows の「スタート」メニューから「設定」→「コントロール パネル」の順に選択し、次に「サービス」アイコンをダブルクリックします。

「サービス」ダイアログ・ボックスが表示されます。



2. 「OracleDiscoverer4i」を選択し、「停止」ボタンをクリックします。

Oracle Discoverer NT サービスを停止すると、CORBA サービスと Locator コンポーネントが停止され、このマシン上で新規セッションを開始できなくなります。Oracle Discoverer NT をシャットダウンしても、進行中のセッションには影響ありません。したがって、作業中のユーザーが作業を中断されることはありません。

現行ユーザー全員がセッションから切断されると、メンテナンス作業が可能になります。コンポーネントの登録または登録解除する場合は、まず、Oracle Discoverer NT サービスを再起動する必要があります。

ユーザー全員が切断したかチェックするには、Windows のタスク・マネージャを開き、Session 実行可能ファイル (dis4ws.exe) のインスタンスがいずれも実行中でないことを確かめます。

2.5.1.2 個々のサーバーを即時シャットダウンするには

即時シャットダウンでは、新規ユーザー・セッションが開始できなくなる他、現行のセッションも停止されます。即時シャットダウンを行うには、マシンの「Oracle Discoverer NT」サービスを停止した後（これにより新規セッションが開始されなくなります）、現在実行中のセッションを停止します。

1. シャットダウンするマシン上で、Windows の「スタート」メニューから「設定」→「コントロール パネル」の順に選択し、次に「サービス」アイコンをダブルクリックします。
「サービス」ダイアログ・ボックスが表示されます。
2. 「OracleDiscoverer4i」を選択し、「停止」ボタンをクリックします。
3. [Ctrl] と [Alt] を押しながら [Del] を押し、Windows のタスク・マネージャを開きます。
4. 「プロセス」タブを選択します。
5. dis4ws.exe を選択します。
6. 「プロセスの終了」をクリックします。
7. タスク・マネージャで実行されている各セッション（dis4ws.exe）に対し、手順 4 および 5 を繰り返します。

2.5.2 Discoverer システム全体のシャットダウン

Discoverer システム全体をシャットダウンする場合は、個々のマシンを特定の順序でシャットダウンしていきます。

Discoverer システム全体のシャットダウン方法には次の 2 通りがあります。

- 準備ができた現行ユーザーから段階的に切断していく方法
- 現行ユーザーの Discoverer セッションをただちに強制終了させる方法

2.5.2.1 Discoverer システムを段階的にシャットダウンするには

Discoverer システムを段階的にシャットダウンすると、現行ユーザーはセッションを完了できますが、新規セッションを開始できなくなります。段階的にシャットダウンする場合は、各サーバーの Oracle Discoverer NT サービスを停止した後に、ユーザーにセッションを終了させます。

1. 各サーバーで、Windows の「スタート」メニューから「設定」→「コントロール パネル」の順に選択し、「サービス」アイコンをダブルクリックします。
2. 「OracleDiscoverer4i」を選択し、「停止」ボタンをクリックします。

Oracle Discoverer NT サービスを停止すると、CORBA サービスと Locator コンポーネントが停止され、このマシン上で新規セッションを開始できなくなります。Oracle

Discoverer NT をシャットダウンしても、進行中のセッションには影響ありません。したがって、作業中のユーザーが作業を中断されることはありません。

これで、Preferences コンポーネントをシャットダウンしました。

警告：すべてのセッションが停止するまで、Preferences コンポーネントはシャットダウンしないでください。セッションがアクティブである間に Preferences コンポーネントをシャットダウンすると、アクティブなセッションに問題が発生する可能性があります。

3. [Ctrl] と [Alt] を押しながら [Delete] を押し、Windows のタスク・マネージャを開きます。
4. dis4pr.exe を選択して「プロセスの終了」をクリックし、Preferences コンポーネントをシャットダウンします。

現行ユーザー全員のセッションが完了すると、Discoverer システムがシャットダウンされます。

2.5.2.2 Discoverer システムを即時シャットダウンするには

即時シャットダウンでは、新規ユーザー・セッションが開始できなくなる他、現行のセッションも停止されます。即時シャットダウンを実行するには、次の操作を行います。

- すべてのマシンにおいて Oracle Discoverer NT サービスを停止し、新規セッションを開始できないようにします。
 - 現在実行中のセッションを停止します。
 - Preferences コンポーネントを停止します。
1. 各サーバーで次の操作を行います。
 - a. Windows の「スタート」メニューから「設定」→「コントロール パネル」の順に選択し、「サービス」アイコンをダブルクリックします。
 - b. 「OracleDiscoverer4i」を選択し、「停止」ボタンをクリックします。
 2. 現行のセッションが実行されている各マシンで、次の操作を行います。
 - c. [Ctrl] + [Alt] + [Del] を押し、Windows のタスク・マネージャを開きます。
 - d. dis4ws.exe を選択します。
 - e. 「プロセスの終了」をクリックします。
 - f. タスク・マネージャで実行されている各セッション (dis4ws.exe) および Preferences (dis4pr.exe) に対して、手順 4 および 5 を繰り返します。

2.6 サーバー・マシンの追加および削除

Discoverer システムからサーバー・マシンをいつでも追加したり削除することができます。

2.6.1 その他の Discoverer サーバー・マシンの追加

Discoverer システムに新しいサーバー・マシンを追加するには、[第 3 章「Oracle9iAS Discoverer の複数のマシンへのインストール」](#)の説明に従ってください。

2.6.2 その他の Discoverer サーバー・マシンを完全に削除するには

Discoverer NT サービスを「無効」にするか、スタートアップを「手動」に変更すると、その他の Discoverer サーバー・マシンが Discoverer システムから削除されます。ソフトウェアを削除する必要はありません。ただし、その他の Discoverer サーバー・マシンの IP アドレスを Discoverer システムから削除するためには、マスター Discoverer サーバー・マシンの pref.txt ファイルを編集する必要があります。

1. Windows の「スタート」メニューから「プログラム」→「Oracle Discoverer Server 4i Setup」→「Edit Preferences」の順に選択します。

Prefs.txt ファイルがメモ帳で開かれます。

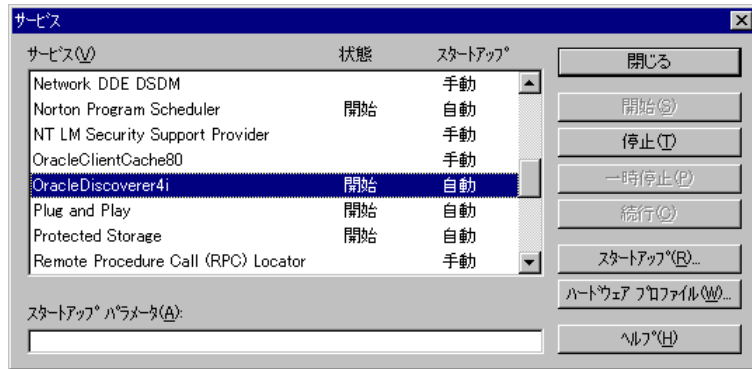
2. pref.txt ファイルから、マシンの IP アドレスを削除します。
3. pref.txt ファイルを保存して、メモ帳を閉じます。
4. 「スタート」、「プログラム」、「Oracle Discoverer Server 4i Setup」、「Apply Preferences」の順に選択します。

5. Windows の「スタート」メニューから、「プログラム」→「Oracle Discoverer Server 4i Setup」の順に再度選択し、登録解除するサーバー上のコンポーネントを選択します。

選択するコンポーネントは、どのサーバーから登録解除するかによって異なります。

- マスター Discoverer サーバー・マシンから削除するには、Session コンポーネントおよび Preferences コンポーネントの両方を登録解除します。
- その他の Discoverer サーバー・マシンから削除するには、Session コンポーネントを登録解除します。

6. 登録解除する各コンポーネントごとに前述の作業を繰り返します。
7. Windows の「スタート」メニューから「設定」→「コントロール パネル」の順に選択し、「サービス」アイコンをダブルクリックします。
「サービス」ダイアログ・ボックスが表示されます。



8. 「OracleDiscoverer4i」を選択し、「開始」ボタンをクリックします。
「サービス」ダイアログ・ボックスが表示されます。
9. 「サービス」ダイアログ・ボックスの「スタートアップ」セクションで、「手動」または「無効」をクリックし、「OK」をクリックします。

これでサーバー・マシンは、Discoverer NT サービスを自動的に起動しなくなり、したがって Discoverer システムの一部ではなくなります。

注意： 一時的にサーバーを削除する場合は、Discoverer NT サービスをシャットダウンします。

2.7 Discoverer Viewer JavaScript 拡張ユーザー・インタフェースの利用

Discoverer Viewer には、次の 2 種類のユーザー・インタフェースが用意されています。

- 純粋な HTML ユーザー・インタフェース
- JavaScript 拡張ユーザー・インタフェース

JavaScript 拡張ユーザー・インタフェースには、次の機能が備わっています。

- アイテム・ドリル用のポップアップ・メニュー
- ドラッグ・アンド・ドロップ式のピボット機能
- 自動適用ページ・アイテム

viewer_config.xml ファイルの use_javascript_enhanced_ui オプションを使用して、ユーザーが JavaScript 拡張ユーザー・インタフェースを使用できるかどうかを指定できます。

次のように変更します。

1. viewer_config.xml ファイルを開いて編集します。

viewer_config.xml ファイルは、<i>SUITES_HOME>%apache%apache%htdocs に格納されています。

2. 必要に応じて use_javascript_enhanced_ui オプションを変更します。

変更例を次に示します。

viewer_config.xml - 変更前

```
<option id="use_javascript_enhanced_ui" state="true"/>
```

viewer_config.xml - 変更後

```
<option id="use_javascript_enhanced_ui" state="false"/>
```

ユーザーが JavaScript 拡張ユーザー・インタフェースを実行するには、各ユーザーのブラウザで JavaScript と Cookie が有効になっている必要があります。

2.8 よくある質問

どのような環境変数が設定されますか？

インストール作業中には、VisiBroker 製品用の 2 つの環境変数が設定されます。

```
VBROKER_JAVAVM = <ORACLE_806_HOME>%jre11728o%bin%jre
VBROKER_TAG = -D
```

CLASSPATH 変数は変更されません。

PATH 変数は次のように更新されます。

```
PATH = <iSUITES_HOME>%bin%;<ORACLE_806_HOME>%vbroker%bin%;<ORACLE_806_
HOME>%bin%; <iSUITES_HOME>%Apache%Perl%5.00503%bin%mswin32-x86;%PATH%
```

これにより、他の製品との競合の可能性が最小限になります。

Locator オブジェクトまたは Preferences コンポーネントが停止するとどうなりますか？

Locator コンポーネントがダウンすると、再起動するまで、新規ユーザーはシステムにログインできなくなります。既存のセッションは影響を受けません。

Locator コンポーネントは、pref.txt ファイルを編集する場合でも、シャットダウンする必要がないように設計されています。デフォルトでは、Locator コンポーネントは、次のクライアントが接続された場合に、編集済みの値を使用します。

これと同様に、Preferences コンポーネントがダウンした場合でも Locator コンポーネントは引き続き機能し、すでに実行中のセッションは影響を受けません。Preferences コンポーネントが再起動すると、Locator コンポーネントは再び Preferences コンポーネントにバインドされます。

ネットワーク・エラーが発生した場合には何をする必要がありますか？

ログイン中に Oracle9iAS Discoverer でネットワークの例外を受信した場合は、次のことを確認してください。

- CLASSPATH または PATH の内部に、Oracle9iAS Discoverer を妨害していると考えられるディレクトリまたは jar ファイルが存在していませんか。Oracle9iAS Discoverer では、VisiBroker CORBA V3.4 および JDK 1.2 が使用されます。
- Locator コンポーネントが起動していることを確かめます。locator.log ファイルにエラー・メッセージが表示されていませんか。locator.ior ファイルが discwb4%applet ディレクトリにあることを確かめます。
- VisiBroker SmartAgent および OAD が起動しており、新規セッションを開始できることを確かめます。
- Preferences コンポーネントが起動していることを確かめます。Preferences コンポーネントが起動していることを確認するには、Windows のタスク・マネージャを使用して、dis4pr.exe が実行されているかどうか確かめます。

「ORA-12154 TNS: サービス名を解決できませんでした。」と表示された場合には何を行いますか？

これは Oracle のエラーの 1 つで、「データベース」フィールドに指定されたデータベースの別名に、セッションが接続できないときに表示されます。次の項目をチェックしてください。

- データベースの別名が、tnsnames.ora ファイルに存在しているかどうか。
- データベースの別名が、セッションを実行する各マシンの tnsnames.ora ファイルに存在しているかどうか。

ヒント: SQL*Plus または他の Oracle 製品をそのマシンで実行している場合は、それらを使用してデータベースに接続してください。

Locator コンポーネントが Preferences コンポーネントにバインドできないというメッセージが表示された場合はどうしますか？

Locator コンポーネントが Preferences コンポーネントにバインドできない場合は、次の項目をチェックしてください。

- Preferences コンポーネントが起動していますか。Preferences コンポーネントが実行中であるかどうかは、Windows のタスク・マネージャを使用して確認します。Preferences コンポーネントを再起動する場合は、[2.2 項「Discoverer NT サービスの使用」](#)を参照してください。
- サブネットで SmartAgent が実行されていますか。コマンド・プロンプトを開き、コマンドラインで osfind と入力し、エージェント、OAD およびサブネット上で実行されているその他の登録済みコンポーネントのリストを確認します。エージェントはリストの最初に表示されていることに注意してください。SmartAgent が実行されていない場合は、それを起動するために [2.2 項「Discoverer NT サービスの使用」](#)を参照してください。
- Preferences コンポーネントがインストールされていますか。Preferences コンポーネントが起動していることを確認するには、Windows のタスク・マネージャを使用して、dis4pr.exe が実行されているかどうか確かめます。
- Preferences コンポーネントが OAD に登録されていますか。Preferences コンポーネントが登録されていることを確認するには、[2.3.2 項「1 つのマシンに登録されたコンポーネントの表示」](#)を参照してください。これは、Preferences コンポーネントがインストールされているマシンで行う必要があります。Preferences コンポーネントが登録されていない場合は、[2.3.5.1 項「Preferences および Session コンポーネントの OAD への登録」](#)を参照してください。
- Preferences コンポーネントを実行しているマシンは、Locator コンポーネントを実行しているマシンおよび Session コンポーネントを実行しているマシンと同じサブネット上にありますか。コマンド・プロンプトを開き、コマンドラインで osfind と入力し、エージェント、OAD およびサブネット上で実行されているその他の登録済みコンポーネントのリストを確認します。エージェントはリストの最初に表示されていることに注意してください。Preferences コンポーネントは、同じサブネット上にインストールする必要があります。

- pref.txt ファイルが破損していませんか。破損されていることをファイルから直接確認することはできません。pref.txt ファイルが破損していることを確認するには、次の手順を実行します。
 1. pref.txt ファイルのコピーを保存します。
 2. 元の pref.txt ファイルを削除します。
 3. Windows の「スタート」メニューから「ファイル名を指定して実行」を選択します。
 4. 「ファイル名を指定して実行」ウィンドウに regedit と入力し、[Enter] キーを押します。
 5. 次の場所でレジストリ・キーを検索します。
HKEY_LOCAL_MACHINE¥software¥oracle¥WebDisco 4i
 6. 検索したレジストリ・キーを削除します。
 7. ファイル <ORACLE_806_HOME>¥discwb4¥util¥defaults.txt を pref.txt としてコピーします。
 8. 「スタート」、「プログラム」、「Oracle Discoverer Server 4i Setup」、「Apply Preferences」の順に選択します。
 9. Discoverer セッションを実行します。
セッションが正しく実行された場合は、元の pref.txt ファイルが破損している可能性があります。
 10. セッションを終了します。
 11. 元の pref.txt ファイルには破損した要素が含まれている可能性があるため、保存済みのコピーを削除します。
 12. エンド・ユーザー Preferences をリセットするには、[2.4 項「エンド・ユーザー Preferences の編集」](#)の説明に従って新しい pref.txt ファイルを編集します。

Oracle9iAS Discoverer の複数のマシンへのインストール

Discoverer Plus および DiscovererViewer は、Oracle9i Application Server 上で実行するために自動的にシングルマシン・インストールでインストールされます。ただし、Oracle9iAS Discoverer は、複数のマシンに分散できるように設計されています。構成ファイルにおいて Discoverer がインストールされたマシンを指定すると、Discoverer Services を複製できます。これにより、Discoverer Services のセッションが複数のマシンで開始され、負荷均衡（ロード・バランシング）につながります。

この方法はボトルネックが解消されるため、多数の Discoverer ユーザーにとって便利です。複数の Discoverer セッションは、1 台のマシンで実行することも、必要に応じて複数のマシンに分散して実行することもできます。

説明するトピックは次のとおりです。

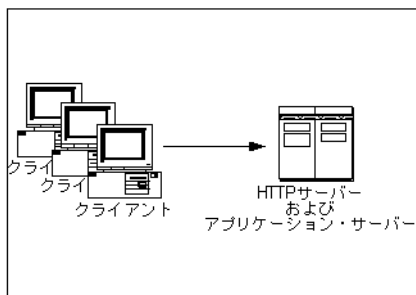
- [Oracle9iAS Discoverer のデフォルト・インストールと分散インストールの違い](#)
- [分散インストールの実行](#)

3.1 Oracle9iAS Discoverer のデフォルト・インストールと分散インストールの違い

3.1.1 Oracle9iAS Discoverer のデフォルト・インストール

Oracle9iAS Discoverer は、Oracle9i Application Server の一部としてインストールされます。デフォルト・インストールはシングルマシン・インストールで、この場合、Discoverer コンポーネントがすべて 1 台のマシンにインストールされます。

図 3-1 Oracle9iAS Discoverer のシングルマシン・インストールによるデフォルト構成



Oracle9i Application Server のインストールでは、同じマシンに次の両方がインストールされます。

- HTTP サーバー・ソフトウェア
- すべての Discoverer Services コンポーネント (Locator、Session、Preferences、SmartAgent および Object Activation Daemon)

したがって、このマシンは HTTP サーバー・マシンとしても、アプリケーション・サーバー・マシンとしても機能を果たします。

クライアント・マシンで Web ブラウザを起動し Oracle9iAS Discoverer を開始すると、Web ブラウザによりサーバー・マシンの HTTP サーバー・ソフトウェアに要求が送信されます。これにより、サーバー・マシンで要求が処理され、Discoverer セッションが開始します。

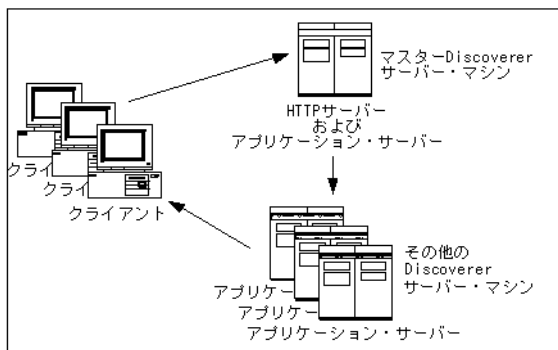
3.1.2 Oracle9iAS Discoverer の分散インストール

1 台のマシンを HTTP サーバーおよびアプリケーション・サーバーとして使用すると、Discoverer ユーザーが多数いる場合に非効率的となる可能性があります。

これは、Discoverer Services コンポーネントが複数のマシンに分散するように設計されているためです。使用するマシンを構成ファイルで指定すると、Discoverer Services セッションが複数のマシンで開始し、負荷均衡 (ロード・バランシング) を行うことができます。複数の

Discoverer セッションは、1 台のマシンで実行することも、必要に応じて複数のマシンに分散して実行することもできます。

図 3-2 Oracle9iAS Discoverer を複数のマシンに分散した場合の構成



この構成では、Oracle9i Application Server のデフォルト・インストールと同様、同じマシンに次の両方がインストールされます。

- HTTP サーバー・ソフトウェア
- すべての Discoverer Services コンポーネント（Locator、Session、Preferences、SmartAgent および Object Activation Daemon）

したがって、このマシンは HTTP サーバー・マシンとしても、アプリケーション・サーバー・マシンとしても機能を果たします。このマシンは、マスター Discoverer サーバー・マシンと呼ばれます。

マスター Discoverer サーバーのみでなく、別のマシンにも Oracle9i Application Server をインストールできます。マスター Discoverer サーバー以外のマシンをその他の Discoverer サーバー・マシンと呼びます。

クライアント・マシンで Web ブラウザを起動し Oracle9iAS Discoverer を開始すると、Web ブラウザによりマスター Discoverer サーバー・マシンの HTTP サーバー・ソフトウェアに要求が送信されます。これにより、マスター Discoverer サーバー・マシンまたはその他の Discoverer サーバー・マシンのいずれかで要求が処理され、Discoverer セッションが開始します。

その他の各 Discoverer サーバー・マシンでは、それぞれ複数のセッションを実行できます。したがって、セッションに使用するマシンの総数は、ユーザーが必要とする同時セッション数、利用可能なメモリーの量、ハードウェアのスピードなどによって異なります。

3.2 分散インストールの実行

3.2.1 デフォルトのマスター Discoverer サーバー・マシンへの接続の確認

デフォルト・インストールの変更を開始する前に、デフォルトのマスター Discoverer サーバー・マシンに接続してデフォルト・インストールが正しく動作していることを確認することをお勧めします。

1. クライアント・マシンの Web ブラウザを起動します。
2. 次の URL を入力して、デフォルトのマスター Discoverer サーバーに接続します。

`http://hostname.domain/discoverer4i/viewer`

`hostname.domain` は、Oracle9i Application Server がインストールされているマシンの完全修飾名です。

デフォルト・インストールが正しく動作している場合は、Discoverer Viewer のログイン画面が表示されます。

3.2.2 マスター Discoverer サーバー・マシンの名前の確認

インストール実行中、マスター Discoverer サーバーのインスタンス名としてマシンの名前が使用されます。マスター Discoverer サーバー・マシンの名前を記録しておいてください。その他の Discoverer サーバー・マシンになるマシンにインストールする場合、この名前が必要となります。

マスター Discoverer サーバー・マシンの名前がわからない場合は、次の説明に従ってください。

1. マスター Discoverer サーバー・マシンで Windows の「スタート」メニューから「ファイル名を指定して実行」を選択します。
2. 「ファイル名を指定して実行」ウィンドウに `regedit` と入力し、[Enter] キーを押します。
3. レジストリ・エディタで、`HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE/` キーを開きます。

`DCW4_INSTANCE_NAME` 文字列の値がマスター Discoverer サーバー・マシンの名前です。

3.2.3 その他の Discoverer サーバー・マシンへの Discoverer のインストール

その他の Discoverer サーバー・マシンとして機能させるすべてのマシンで、次の操作を行います。

1. 『Oracle9i Application Server インストール・ガイド』の説明に従って、その他の Discoverer サーバー・マシンとして使用するマシンに Oracle9i Application Server をインストールします。
2. その他の Discoverer サーバー・マシンで、Windows の「スタート」メニューから次の操作を行います。
 - a. 「プログラム」、「Oracle Discoverer Server 4i Setup」、「Unregister Preferences」の順に選択します。
 - b. 「プログラム」、「Oracle Discoverer Server 4i Setup」、「Unregister Collector」の順に選択します。
 - c. 「プログラム」、「Oracle Discoverer Server 4i Setup」、「Unregister Session」の順に選択します。
3. その他の Discoverer サーバー・マシンで、Windows の「スタート」メニューから「ファイル名を指定して実行」を選択します。
4. 「ファイル名を指定して実行」ウィンドウに regedit と入力し、[Enter] キーを押します。
5. レジストリ・エディタで、HKEY_LOCAL_MACHINE/SOFTWARE/ORACLE/ キーを開きます。
6. 次の表を参照して、文字列の値を変更します。

文字列	変更前	変更後
DCW4_START_LOCATOR	1	0
DCW4_INSTANCE_NAME	その他の Discoverer サーバー・マシンの名前	マスター Discoverer サーバー・マシンの名前 (3.2.2 項「マスター Discoverer サーバー・マシンの名前の確認」を参照)
DCW4_OADPORT	現在のポート番号	マスター Discoverer サーバー・マシンの文字列と同じ値

マスター Discoverer サーバー・マシンのレジストリ・エディタを開いて、DCW4_OADPORT の値を検索する必要があります。

7. マスター Discoverer サーバー・マシンで次の操作を行います。
 - a. Windows の「スタート」メニューから「設定」→「コントロール パネル」の順に選択し、「サービス」アイコンをダブルクリックし、「サービス」ダイアログ・ボックスを表示します。
 - b. 「サービス」ダイアログ・ボックスの「OracleDiscoverer4i」サービスを選択し、「停止」ボタンをクリックします。
 - c. 「サービス」ダイアログ・ボックスの「[OracleiAS_Home] HTTPServer」サービスを選択し、「停止」ボタンをクリックします。
 - d. ダイアログ・ボックスを閉じます。
8. その他の Discoverer サーバー・マシンで次の操作を行います。
 - a. Windows の「スタート」メニューから「設定」→「コントロール パネル」の順に選択し、「サービス」アイコンをダブルクリックし、「サービス」ダイアログ・ボックスを表示します。
 - b. 「サービス」ダイアログ・ボックスの「OracleDiscoverer4i」サービスを選択し、「停止」ボタンをクリックします。
 - c. 「サービス」ダイアログ・ボックスの「[OracleiAS_Home] HTTPServer」サービスを選択し、「停止」ボタンをクリックします。
 - d. 「サービス」ダイアログ・ボックスで「OracleiAS_HomeHTTPServer」サービスを選択し、「スタートアップ」ボタンをクリックして「サービス」ダイアログ・ボックスを表示し、「スタートアップの種類」で「手動」を選択します。
 - e. 「OK」をクリックして「サービス」ダイアログ・ボックスを閉じます。
 - f. 「閉じる」をクリックして「サービス」ダイアログ・ボックスを閉じます。
9. マスター Discoverer サーバー・マシンおよびその他の Discoverer サーバー・マシンの両方で、次の説明に従って、現在実行中の Discoverer の処理を停止します。
 - a. [Ctrl]+[Alt]+[Del] を押し「タスク マネージャ」ボタンをクリックして、Windows のタスク・マネージャを表示します。
 - b. 「プロセス」タブに次に示す Discoverer のプロセスのうちいずれかが表示されている場合はクリックし、「プロセスの終了」をクリックします。すべてのプロセスを終了します。

dis4ws.exe

dis4pr.exe

dis4srv.exe
 - c. Windows のタスク・マネージャを閉じます。

10. マスター Discoverer サーバー・マシンからマシンに次のディレクトリおよびサブディレクトリをコピーします。

<ORACLE_806_HOME>%discwb4%

<ORACLE_806_HOME> は、Discoverer コンポーネントがインストールされている場所です。

discwb4 ディレクトリの内容すべてをコピーするには、複数の方法があります。まず、Windows のエクスプローラを使用してその他の Discoverer サーバー・マシンのネットワーク・ドライブをマスター Discoverer サーバー・マシンの適切なドライブにマップする方法です。また、FTP ソフトウェアを使用してファイルをコピーすることも可能です。

11. マスター Discoverer サーバー・マシンを再起動します。
12. その他の Discoverer サーバー・マシンを再起動します。
13. その他の Discoverer サーバー・マシンで、Windows の「スタート」メニューから「プログラム」→「Oracle Discoverer Server 4i Setup」→「Register Session」の順に選択します。

その他の Discoverer サーバー・マシンに、不要なメニュー項目とバッチ・ファイルがインストールされます。Discoverer Services コンポーネントへの不注意なアクセスを防ぐために、これらのメニュー項目とバッチ・ファイルを削除することをお勧めします。

14. その他の Discoverer サーバー・マシンから Discoverer Services の不要なメニュー項目を削除するには、次の操作を行います。
 - a. Windows の「スタート」メニュー・ボタンを右クリックし、「エクスプローラ - All Users」を選択して Windows のエクスプローラを開きます。
 - b. Start Menu\Programs\Oracle Discoverer Server 4i Setup ディレクトリを開きます。
 - c. 次のメニュー項目を削除します。

「Apply Preferences」

「Edit Preferences」

「Register Collector」

「Register Preferences」

「Unregister Collector」

「Unregister Preferences」

- d. Windows のエクスプローラを閉じます。

15. その他の Discoverer サーバー・マシンから Discoverer Services の不要なバッチ・ファイルを削除するには、次の操作を行います。

- a. Windows のエクスプローラを開いて、次のディレクトリに移動します。

<ORACLE_806_HOME>%discwb4%util

<ORACLE_806_HOME> は、Discoverer コンポーネントがインストールされている場所です。

- b. 次のバッチ・ファイルを削除します。

RegisterCollector.bat

RegisterPreference.bat

StartLocator.bat

unRegisterCollector.bat

unRegisterPreferences.bat

その他の Discoverer サーバー・マシンにするすべてのマシンに Discoverer をインストールしている場合は、次の操作を行ってください。

- マスター Discoverer サーバー・マシン (Discoverer Plus および Viewer。3.2.4 項を参照) の pref.txt ファイルを編集します。
- すべての Discoverer Server マシン (Discoverer Plus および Viewer。3.2.5 項を参照) の tnsnames.ora ファイルを編集します。
- マスター Discoverer サーバー・マシン (Discoverer Viewer のみ。3.2.6 項を参照) の disco4iv.xml ファイルを編集します。

3.2.4 Discoverer Services の pref.txt ファイルの編集 (Discoverer Plus および Viewer)

3.2.4.1 pref.txt ファイル

マスター Discoverer サーバーに置かれている pref.txt ファイルには、すべてのエンド・ユーザーに適用されるデフォルト設定が多数含まれています。

pref.txt ファイルで必要となる情報は、Oracle9iAS Discoverer が使用されているマシンの IP アドレスまたはマシン名のみです。pref.txt ファイルのその他の項目は、ネットワークまたはユーザーの要件に合わせて必要な場合に編集します。詳細は、3.2.4.5 項「[pref.txt ファイルの編集方法](#)」を参照してください。

pref.txt ファイルのバックアップ・コピーは、マスター Discoverer サーバー・マシンの <ORACLE_806_HOME>%discwb4%util%defaults.txt に保存されています。編集中にミスをし

た場合、または `pref.txt` ファイルが壊れるか失われた場合は、このバックアップファイルを使用してデフォルトの値に設定し直します。

3.2.4.2 `pref.txt` ファイルを編集する理由

Discoverer の分散インストールを行う場合、Oracle9iAS Discoverer のセッションを実行するすべてのアプリケーション・サーバー・マシンの IP アドレスまたはマシン名がマスター Discoverer サーバーに置かれた `pref.txt` ファイルに含まれている必要があります。

Session コンポーネントをインストールした各マシンでのセッションの開始順序は、`pref.txt` にリストされた IP アドレスの順序によって決まります。`pref.txt` ファイルを編集すると、その他の Discoverer サーバー・マシンを追加して、負荷均衡（ロード・バランシング）を行うことができます。

`pref.txt` ファイルはカンマ・スペース区切り形式です。つまり、エントリの間にはカンマ、スペースが順に挿入されます。エントリは二重引用符で囲んでください。

注意： マシン名を使用すると参照が 1 回増えるため、IP アドレスを使用した方が幾分速くなります。IP アドレスの使用とマシン名の使用の比較については、システム管理者にお問い合わせください。

3.2.4.3 負荷均衡（ロード・バランシング）

負荷均衡（ロード・バランシング）とは Discoverer インストールセッションのパフォーマンスを向上させる手段で、複数のマシンにセッションを分散させ、各マシンの使用効率を高めます。`pref.txt` ファイルで、使用するマシンの IP アドレスまたはマシン名を指定します。

Discoverer Services Locator コンポーネントでは、`pref.txt` ファイルのマシン・リストを使用して、次に要求されたセッションを開始するマシンを決定します。各マシンの Locator は、ラウンドロビン方式に基づいて、セッションを IP アドレスのリストの最後に到達するまで順番に要求していきます。リストの最後に到達すると、Locator は再びリストの最初から前述の動作を行います。

ラウンドロビン方式では次のことを行ってマシンの負荷を均衡させます。

- マシンを特定の順序で設定する
- 同じマシンを何度も使用する

たとえば、1 つのマシンで 5 つのセッションを処理した後で、次のマシンを使用することも考えられます。その場合は、最初のマシンの IP アドレスを 5 回入力した後、次のマシンのアドレスを `pref.txt` ファイルから入力します。

負荷均衡（ロード・バランシング）は、パフォーマンスを著しく向上させます。`pref.txt` ファイルを編集する場合は、まず負荷均衡（ロード・バランシング）について計画することをお勧めします。各サーバー・マシンの速度およびパフォーマンスについて考えてください。

参考： 1 つのセッションにつき約 3.8 ～ 18 MB のメモリが必要になります。

3.2.4.4 負荷均衡（ロード・バランシング）の例

例 1: 同じ性能のマシンによる負荷均衡（ロード・バランシング）

Discoverer Session コンポーネントが、2 つのアプリケーション・サーバーにインストールされています。各マシンのプロセッサ・スピードおよび RAM は同じです。これら 2 つのマシンの IP アドレスは次のようになります。

表 3-1 サンプル・サーバー名

サーバー	マシン名	IP アドレス
アプリケーション・サーバー・マシン 1	disco1	123.456.789.1
アプリケーション・サーバー・マシン 2	disco2	123.456.789.2

アプリケーション・サーバー 1 で最初の要求を、アプリケーション・サーバー 2 で次の要求をそれぞれ処理し、その次の要求はまたアプリケーション・サーバー 1 で処理します。

この場合、pref.txt ファイルのエントリは次のようになります。

```
Machine IPs="disco1, disco2"

または

Machine IPs="123.456.789.1, 123.456.789.2"
```

例 2: 速いマシン 1 台とその他 2 台のマシンによる負荷均衡（ロード・バランシング）

Discoverer Sessions が、3 つのアプリケーション・サーバー・マシンにインストールされています。1 番目のアプリケーション・サーバー・マシンのプロセッサが最速で、RAM も最大です。2 番目と 3 番目のアプリケーション・サーバー・マシンは同じプロセッサ・スピードと RAM を備えています。

これら 3 つのマシンの IP アドレスは次のようになります。

表 3-2 サンプル・サーバー名

サーバー	マシン名	IP アドレス
アプリケーション・サーバー・マシン 1	serve1	123.456.789.1
アプリケーション・サーバー・マシン 2	serve2	123.456.789.2
アプリケーション・サーバー・マシン 3	serve3	123.456.789.3

アプリケーション・サーバー 1 で最初の要求、アプリケーション・サーバー 2 で次の要求、アプリケーション・サーバー 3 でさらに次の要求を処理し、最初の要求から 3 番目の要求はまたアプリケーション・サーバー 1 で処理します。

この場合、pref.txt ファイルのエントリは次のようになります。

```
Machine IPs="serve1, serve1,serve1, serve2, serve3"
```

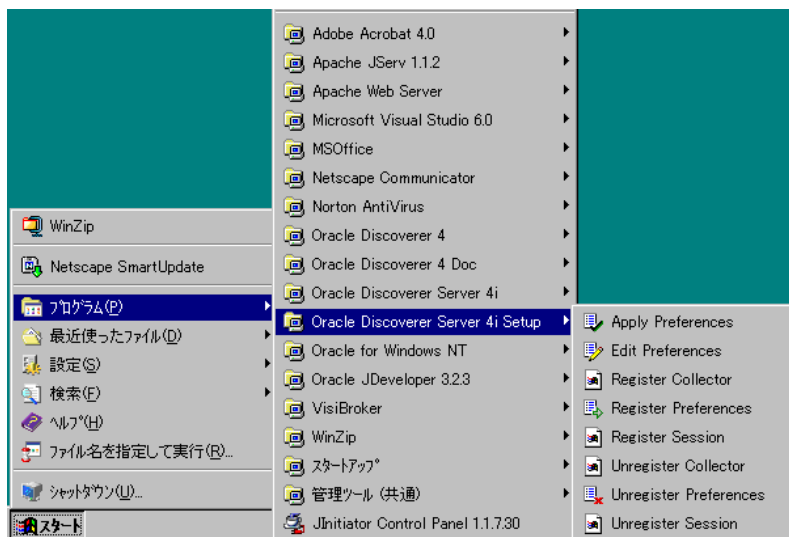
または

```
Machine IPs="123.456.789.1,123.456.789.1,123.456.789.1, 123.456.789.2, 123.456.789.3"
```

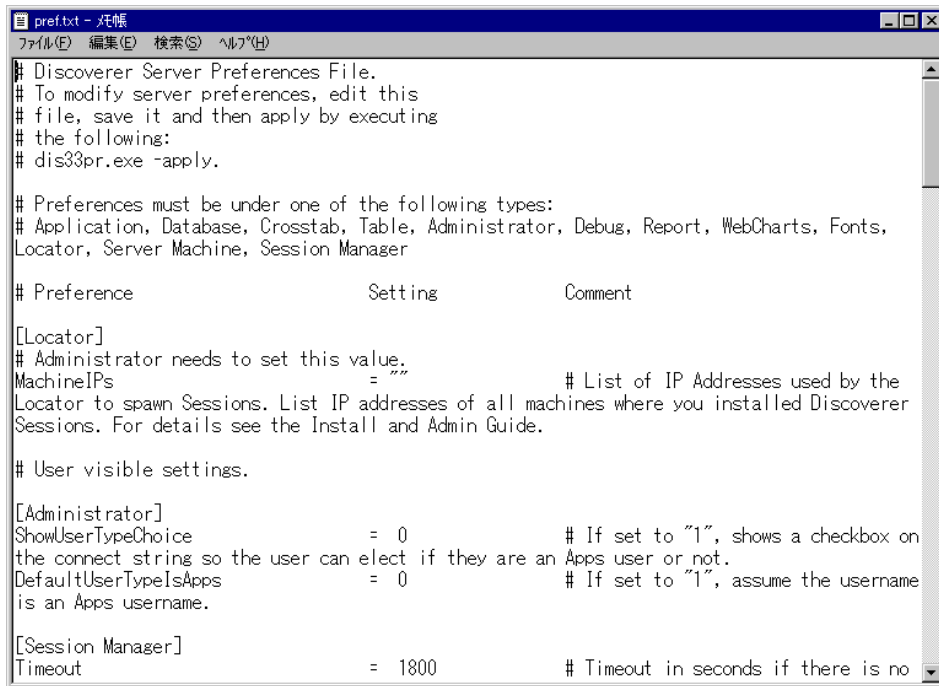
3.2.4.5 pref.txt ファイルの編集方法

pref.txt ファイルの編集は、マスター Discoverer サーバー・マシンなど、Preferences コンポーネントがインストールされているマシンで行います。

1. Windows の「スタート」メニューから「プログラム」→「Oracle Discoverer Server 4i Setup」→「Edit Preferences」の順に選択します。



ファイル Pref.txt がメモ帳アプリケーションで開きます。



```
# Discoverer Server Preferences File.
# To modify server preferences, edit this
# file, save it and then apply by executing
# the following:
# dis33pr.exe -apply.

# Preferences must be under one of the following types:
# Application, Database, Crosstab, Table, Administrator, Debug, Report, WebCharts, Fonts,
# Locator, Server Machine, Session Manager

# Preference                Setting                Comment

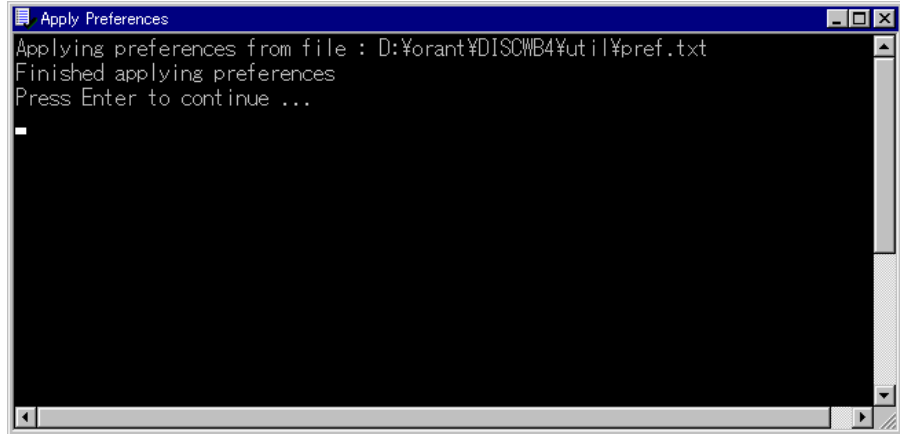
[Locator]
# Administrator needs to set this value.
MachineIPs                    = ""                # List of IP Addresses used by the
# Locator to spawn Sessions. List IP addresses of all machines where you installed Discoverer
# Sessions. For details see the Install and Admin Guide.

# User visible settings.

[Administrator]
ShowUserTypeChoice            = 0                # If set to "1", shows a checkbox on
# the connect string so the user can elect if they are an Apps user or not.
DefaultUserTypeIsApps         = 0                # If set to "1", assume the username
# is an Apps username.

[Session Manager]
Timeout                        = 1800                # Timeout in seconds if there is no
```

2. 各マシンの IP アドレスまたはマシン名を、ファイルの先頭付近の「MachineIPs」というラベルの付いた行に入力します。
3. IP アドレスまたはマシン名をカンマで区切りその後にスペースを入力します。リスト全体は二重引用符で囲みます。
例: "server1, server2, server3"
4. pref.txt ファイルの編集が完了した後、保存してメモ帳を終了します。
pref.txt の変更内容を有効にするには、Prefereces を適用する必要があります。
5. Windows の「スタート」メニューから「プログラム」→「Oracle Discoverer Server 4i Setup」→「Apply Preferences」の順に選択します。
新しい作業環境が保存されたことを示す確認ウィンドウが表示されます。



pref.txt ファイルにエラーが検出されると、error.txt ファイルの確認を促すウィンドウが表示されます。error.txt ファイルのエラーを修正し、Preferences を再度適用してから続行します。

6. 続行する場合は、[Enter] キーを押します。

Windows NT レジストリの Preferences が更新されます。これで Discoverer Server インスタンスの Preferences が設定されました。

3.2.5 tnsnames.ora ファイルの編集（Discoverer Plus および Viewer）

3.2.5.1 tnsnames.ora ファイルとは

tnsnames.ora ファイルには、ユーザーが Oracle9iAS Discoverer またはその他の Oracle 製品を使用してアクセスする、全データベースの名前および別名が記載されています。

3.2.5.2 tnsnames.ora ファイルを編集する理由

Discoverer を分散インストールしたサーバー・マシンの tnsnames.ora ファイルはすべて同一である必要があります。

その他の Discoverer サーバー・マシンに Discoverer をインストールした場合は、次の操作を行って、各マシンの tnsnames.ora ファイルが同一であることを確認してください。

- マスター Discoverer サーバー・マシンの tnsnames.ora ファイルを各その他の Discoverer サーバー・マシンにコピーします。
- 各その他の Discoverer サーバー・マシンの tnsnames.ora ファイルがマスター Discoverer サーバー・マシンのファイルと同一になるように編集します。

同じマシンに `tnsnames.ora` ファイルのバージョンが複数存在する可能性があることに注意してください。正しいファイルをコピーまたは編集することが重要です。

3.2.5.3 tnsnames.ora ファイルのコピー方法

1. マスター Discoverer サーバー・マシン上の次の場所にある `tnsnames.ora` ファイルを、その他の Discoverer サーバー・マシンの同じ場所にコピーして既存のファイルに置き換えます。

<ORACLE_806_HOME>%net80¥admin

<ORACLE_806_HOME> は、Discoverer コンポーネントがインストールされている場所です。

他にも、`tnsnames.ora` ファイルをコピーする方法が複数あります。まず、Windows のエクスプローラを使用してその他の Discoverer サーバー・マシンのネットワーク・ドライブをマスター Discoverer サーバー・マシンの適切なドライブにマップする方法です。また、FTP ソフトウェアを使用してファイルをコピーすることも可能です。

3.2.5.4 tnsnames.ora ファイルの編集方法

`tnsnames.ora` ファイルの編集方法の一例を示します。

1. Windows の「スタート」メニューから「プログラム」→「Oracle for Windows NT - Oracle_806」→「Oracle Net8 Easy Config」の順に選択します。

プログラムが実行され、Net8 構成の編集が可能になります。

正しいファイルを編集するために、「プログラム」→「Oracle - iSuites」→「Network Administration」の Net8 Configuration Assistant は使用しないでください。

3.2.6 disco4iv.xml ファイルの編集（Discoverer Viewer のみ）

3.2.6.1 disco4iv.xml ファイルとは

マスター Discoverer サーバー・マシンの `disco4iv.xml` ファイルにより、Discoverer Viewer に構成情報が提供されます。また、Discoverer Viewer は `disco4iv.xml` ファイルを使用して、マスター Discoverer サーバー・マシンを他のマシンと識別します。

3.2.6.2 disco4iv.xml ファイルを編集する理由

Discoverer Viewer を複数のマシンにインストールした場合、`disco4iv.xml` ファイルを編集してマスター Discoverer サーバーを識別できるようにする必要があります。

3.2.6.3 disco4iv.xml ファイルの編集方法および変更箇所

disco4iv.xml ファイルを編集する前に、将来的に参照する場合のためにコピーをとっておくことをお勧めします。

1. メモ帳を使用して disco4iv.xml ファイルを開きます。ファイルはマスター Discoverer サーバーの次の場所にあります。

```
<iSUITES_HOME>%apache%apache%htdocs%disco4iv%html
```

<iSUITES_HOME> は、Oracle9i Application Server コンポーネントをインストールする場所です。

2. disco4iv.xml ファイルの Session 名のエントリが次のとおりかどうか確認します。

```
session name="instance_nameOracleDiscovererSession4"
```

instance_name は、マスター Discoverer サーバーのインスタンス名です。

3. disco4iv.xml ファイルの Locator 名のエントリが次のとおりかどうか確認します。

```
locator name="instance_nameOracleDiscovererLocator4"
```

instance_name は、マスター Discoverer サーバーのインスタンス名です。

4. disco4iv.xml ファイルのパス・エントリを次のように変更します。

変更前 path="http://%LOCATOR_URL%"

変更後 path="http://**locator_location**"

locator_location は、locator.ior ファイルの場所です。

例 path="http://machinename.oracle.com/discwb4/Applet/"

5. disco4iv.xml ファイルから次の 2 行を削除します。

```
<!-- REMOVE THIS COMMENT LINE (A) only if Discoverer 4i Viewer is installed on a different subnet than the Discoverer Services, or you are performing a distributed Discoverer Services installation
```

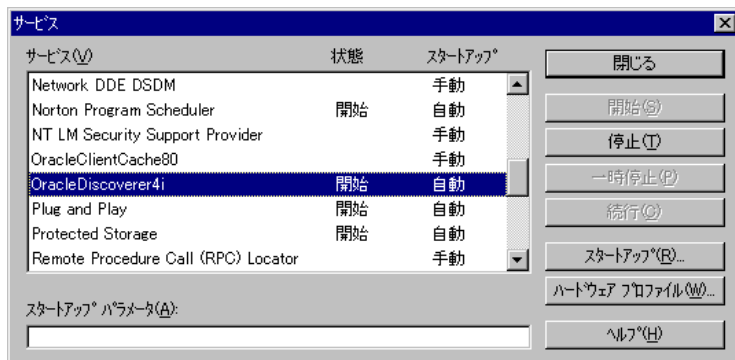
```
REMOVE THIS COMMENT LINE (B) only if Discoverer 4i Viewer is installed on a different subnet than the Discoverer Services, or you are performing a distributed Discoverer Services installation-->
```

6. ファイルを保存し、メモ帳を終了します。

disco4iv.xml ファイルの変更内容を有効にするには、マスター Discoverer サーバー・マシンの Windows NT サービスを再起動する必要があります。

7. Windows の「スタート」メニューから「設定」→「コントロールパネル」の順に選択し、「サービス」アイコンをダブルクリックします。

「サービス」ダイアログ・ボックスが表示されます。



8. 「サービス」ダイアログ・ボックスの「OracleDiscoverer4i」サービスを選択し、「停止」ボタンをクリックします。
9. 「サービス」ダイアログ・ボックスの「[OracleiAS_Home] HTTPServer」サービスを選択し、「停止」ボタンをクリックします。
10. 数秒後、サービスが停止していることを確認します。
11. 「サービス」ダイアログ・ボックスの「OracleDiscoverer4i」サービスを選択し、「開始」ボタンをクリックします。
12. 「サービス」ダイアログ・ボックスの「[OracleiAS_Home] HTTPServer」サービスを選択し、「開始」ボタンをクリックします。
13. ダイアログ・ボックスを閉じます。

3.2.7 その他の Discoverer サーバーへの接続の確認

その他の Discoverer サーバーが正しく作成されたことを確認するために、次のように各その他の Discoverer サーバーに接続することをお勧めします。

1. 各その他の Discoverer サーバー・マシンで、[Ctrl]+[Alt]+[Del] を押し「タスク マネージャ」ボタンをクリックして、Windows のタスク・マネージャを表示します。次に「Processes」タブを選択します。
2. クライアント・マシンの Web ブラウザを起動します。
3. 次の URL を入力して、マスター Discoverer サーバーに接続します。

`http://hostname.domain/discoverer4i/viewer`

`hostname.domain` は、マスター Discoverer サーバー・マシンの完全修飾名です。

pref.txt ファイルで最初に指定したマシンで処理が開始され、Discoverer Viewer のログイン画面が表示されます。

4. インストールが成功したかどうかを確認するには、最初のその他の Discoverer サーバー・マシンで起動される `dis4ws.exe` プロセスを探します。
5. `pref.txt` ファイルで指定したその他の Discoverer サーバーすべてのマシンについて、前述の手順を繰り返します。

URL を使用した Oracle9iAS Discoverer の自動的な起動

エンド・ユーザーは、ログインによって Oracle9iAS Discoverer を起動し、ワークブックを選択し、パラメータ値を選択するのが一般的です。この手順を短縮するには、特定の設定（たとえば、開くワークブックや使用するパラメータ値の設定）を行った状態で Oracle9iAS Discoverer を起動するための URL を作成します。URL を使用して Discoverer を自動的に起動すると、エンド・ユーザーの時間が節約できるだけでなく、エンド・ユーザーが使用するワークブックの制御も可能になります。

URL を作成しておくと、次の作業が可能になります。

- Web ブラウザに入力する URL をエンド・ユーザーに与えます。
- エンド・ユーザーが Discoverer をシングル・クリックで起動できるように、URL をリンクとして企業のイントラネット・サイトに追加することもできます。

この章のトピックは、次のとおりです。

- [Oracle9iAS Discoverer を起動するための URL の書式](#)
- [Discoverer Plus の URL 引数](#)
- [Discoverer Viewer の URL 引数](#)

4.1 Oracle9iAS Discoverer を起動するための URL の書式

作成する URL は、次のような標準の URL コマンドライン構文に従う必要があります。

```
http://hostname.domain/prog_name?arg1=value1&arg2=value2&...&argN=valueN
```

この内容は、次のとおりです。

- `hostname.domain` は、その他の Discoverer サーバー・マシンの完全修飾名です。
- `prog_name` は、次のいずれかです（起動するのが Discoverer Plus または Discoverer Viewer のどちらかに依存します）。
 - 「Discoverer Plus スタート・ページ」ページのロケーションおよび名前（`discwb4/html/japanese/welcome.htm` など）。「Discoverer Plus スタート・ページ」でエンド・ユーザーが「**Click to Start**」をクリックすると、Discoverer Plus アプレットが起動します。
 - Discoverer Viewer Servlet のロケーションおよび名前（`servlets/disco4iv` など）。
- `arg1=value1` は、最初の引数とその引数に指定する値です。

URL での引数の順序は重要ではありません。

次の必須引数が URL に指定されていないと、エンド・ユーザーはその入力をプロンプトされます。

- ユーザー名
- パスワード
- データベース
- ワークブック名
- ワークシート名
- パラメータ

URL に特殊文字を挿入する場合には、それらを等価の ASCII（または UNICODE）コードに置換します。この処理は、URL エンコーディングと呼ばれます。たとえば、URL では、スペースをプラス（+）記号に置換する必要があります。次のリストに含まれない文字は、URL エンコーディングが必要です。

- 大文字（例：`ABCDEFGHIJKLMNOPQRSTUVWXYZ`）
- 小文字（例：`abcdefghijklmnopqrstuvwxyz`）
- 数字（例：`01234567890`）
- 一部の特殊文字（例：`:-_!~()*'`）

4.2 Discoverer Plus の URL 引数

表 4-1 Discoverer Plus の URL 引数と値

引数と値	目的	例
BrandImage=< イメージ >	Discoverer のロゴと置換する GIF ファイルの URL を指定します。たとえば、Discoverer のロゴを会社のロゴに置換できます。	BrandImage=http://.../logo.gif logo.gif に会社のロゴを指定します。
connect=[APPS_SECURE] < ホスト名 _ インスタンス名 >	Oracle Applications のエンド・ユーザーのために、保護モードで接続されるように指定できます。< ホスト名 _ インスタンス名 > には、データベースのホスト名とアプリケーション・サーバーのインスタンスを指定します。	connect=[APPS_SECURE]genledger_payables
connect=< 接続文字列 >	接続文字列によって提供される接続情報を使用して、データベースに接続するよう Discoverer に指示します。この引数により、エンド・ユーザーに対する「接続」ダイアログは完全に省略されます。	connect=username/password@demodb または、ODBC 接続の場合は connect=username/password@ODBC:demodb
ConnectImage=< イメージ >	「Discoverer スタート・ページ」の虫眼鏡イメージと置換する GIF ファイルの URL を指定します。	ConnectImage=http://.../logo.gif logo.gif に会社のロゴを指定します。
DefaultConnect= < データベース名または別名 >	Discoverer が起動すると、エンド・ユーザーに「接続」ダイアログが表示されます。この引数を使用すると、「データベース」フィールドに値が自動的に入力されます。ユーザーはこのデータベース名または別名を上書きできます。	DefaultConnect=demodb または、ODBC 接続の場合は DefaultConnect=ODBC:demodb
DefaultPassword=< 文字列 >	Discoverer が起動すると、エンド・ユーザーに「接続」ダイアログが表示されます。この引数を使用すると、「パスワード」フィールドに値が自動的に入力されます。ユーザーはこのパスワードを上書きできます。	DefaultPassword=secret_code
DefaultUsername= < データベースのユーザー名 >	Discoverer が起動すると、エンド・ユーザーに「接続」ダイアログが表示されます。この引数を使用すると、「ユーザー名」フィールドに値が自動的に入力されます。ユーザーはこのユーザー名を上書きできます。	DefaultUsername=video_user

表 4-1 Discoverer Plus の URL 引数と値（続き）

引数と値	目的	例
eul=< スキーマ名 >	エンド・ユーザーがデフォルトで接続する EUL が含まれているデータベース・スキーマの名前を指定します。この引数を使用しない場合、Discoverer は他の方法を使用してデフォルトの EUL を検出します（たとえば、エンド・ユーザー・オプション設定から）。	eul=VIDEOEUL Discoverer は、VIDEOEUL スキーマで検索した EUL のメタデータを使用します。
FrameDisplayStyle= <launched または separate>	Discoverer アプリケーション・フレームを表示する方法を指定します。 ■ 値「launched」は、アプレットの「Start」ボタンをコール側 HTML に埋め込みます。エンド・ユーザーは、「Start」ボタンをクリックする必要があります。 ■ 値「separate」は、エンド・ユーザーのために Discoverer アプレットを自動的に起動します。エンド・ユーザーは、「Start」ボタンをクリックする必要はありません。	FrameDisplayStyle=separate エンド・ユーザーが「Start」ボタンをクリックしないときも、Discoverer フレームがすぐに別のウィンドウで開きます。
Locale=< 言語 [_ 国][_ バリ アント・コード]>	エンド・ユーザーが Discoverer を使用する ときに表示される言語（オプションと して、国およびバリエーション・コード）を 選択します。この引数により、エンド・ ユーザーのクライアント・マシンのロ ケール設定が変更されます。 言語、国およびバリエーション・コードを指 定するための Java ネーミング規則。	Locale=es_ES 言語はスペイン語で、国はスペインになり ます。
NLSDateFormat= < 日付書式 >	特定の地域に対する日付の書式を指定し ます。値としては、DD-Mon-YY などの SQL 日付書式マスクを入力します。	NLSDateFormat=DD-Mon-YY
NLSDateLanguage= < 日付言語 >	エンド・ユーザーが Discoverer を使用する ときに表示される曜日と月に対する言 語を選択します。	NLSDateLanguage=Spanish
NLSLang=< 言語 >	エンド・ユーザーが Discoverer を使用する ときに表示される言語、地域および キャラクタ・セットを選択します。 言語、地域およびキャラクタ・セットを 指定するための Oracle ネーミング規則。	NLSLang=Spanish

表 4-1 Discoverer Plus の URL 引数と値（続き）

引数と値	目的	例
NLSNumeric_Characters= < 区切り文字 >	数値に対する小数点文字およびグループ・セパレータ文字を選択します。	NLSNumericCharacters=.,
NLSSort=< ソートのタイプ >	文字ソート基準を指定します。	NLSSort=XSpanish
opendb=< ワークブック名 >	開くワークブックが（スケジュールされているものではなく）データベースに保存されていることを前提として、その名前を Discoverer に提供します。URL で opendb 引数を 2 度以上使用すると、Discoverer は最後の引数を使用します。	opendb=Video Sales Workbook Discoverer はワークブックがデータベースに保存されていると想定して、自動的にそれを開きます。
param_< パラメータ名 >= < パラメータ値 >	パラメータが含まれるワークブックに対して、この引数を使用して値を選択します。ワークブックにその名前のパラメータが含まれていない場合、Discoverer はその引数を見捨てます。パラメータの名前の先頭に param_ が付くことに注意してください。	param_regionparam=East regionparam という名前のパラメータに East という値を設定するよう Discoverer に指示します。
Responsiblity= < 職責名 >	Oracle Applications のエンド・ユーザー（たとえば、Oracle General Ledger）の場合には、その職責を指定できます。	Responsiblity=Manager Discoverer は「職責」ダイアログを省略して、エンド・ユーザーにマネージャの Oracle Applications 職責を割り当てます。
Sheet=< ワークブック名 >	ユーザーがデフォルトで開くワークブックの名前を Discoverer に提供します。URL で Sheet 引数を 2 度以上使用すると、Discoverer は最後の引数を使用します。	Sheet=Sales Detail Sheet Discoverer は Sales Detail Sheet ワークシートに対する問合せを実行し、結果をエンド・ユーザーに表示します。
WindowHeight= < ピクセル数 >	Discoverer アプリケーション・フレームの高さを、ピクセルで指定します。この引数を使用しない場合、Discoverer のデフォルト値を使用します。	WindowHeight=600 Discoverer フレームの高さは 600 ピクセルとなります。
WindowWidth= < ピクセル数 >	Discoverer アプリケーション・フレームの幅をピクセルで指定します。この引数を使用しない場合、Discoverer のデフォルト値を使用します。	WindowWidth=800 Discoverer フレームの幅は 800 ピクセルとなります。
WorkbookName= < ワークブック名 >	ユーザーがデフォルトで開くワークブックの名前を Discoverer に提供します。	WorkbookName=Video Sales Workbook Discoverer はデータベースに保存されている Video Sales Workbook というワークブックを検索します。

表 4-1 Discoverer Plus の URL 引数と値（続き）

引数と値	目的	例
WorkbookSource=<database または scheduled>	次のいずれかで、開くワークブックの保存されている場所を Discoverer に指定します。 「database」を指定すると、データベースに保存されているワークブックを開きます。 「scheduled」を指定すると、周期的に更新されているワークブックを開きます。	WorkbookSource=database Discoverer はデータベースに保存されているワークブックを検索します。

4.3 Discoverer Viewer の URL 引数

表 4-2 Discoverer Viewer の URL 引数と値

引数と値	目的	例
anlsdf=< 日付書式 >	Oracle Applications のエンド・ユーザーの場合、セッションのデフォルトの日付書式を変更できます。	anlsdf='MM/DD/YY'
anlsdl=< 日付言語 >	Oracle Applications のエンド・ユーザーの場合、曜日と月名および日付の略式の綴り（AM、PM、AD、BC）を変更できます。	anlsdl=fr
anlsl=< 言語 >	Oracle Applications のエンド・ユーザーの場合、セッション言語を変更できます。	anlsl=en-gb
anlsnc=< 区切り文字 >	Oracle Applications のエンド・ユーザーの場合、グループ・セパレータおよび小数点として使用するデフォルトの文字を変更できます。 小数点区切りを指定してから、続けてグループ・セパレータを指定する必要があります。	anlsnc=',.'

表 4-2 Discoverer Viewer の URL 引数と値（続き）

引数と値	目的	例
anlss= < ソート名または binary >	Oracle Applications のエンド・ユーザーの場合、ORDER BY 問合せおよび文字列比較時のセッション照合順序を変更できます。 値は次のとおりです。 ■ 言語ソート順序の名前 ■ バイナリ・ソートを指定する場合には binary	anlss=binary
cp_height=< ピクセル数 >	グラフの高さをピクセル単位で指定します。	cp_height=200
cp_show_legend=<true または false>	グラフのタイトルを表示するかどうかを指定します。	cp_show_legend=true
cp_width=< ピクセル数 >	グラフの幅をピクセル単位で指定します。	cp_width=300
cs=[APPS_SECURE] < ホスト名_インスタンス名 >	Oracle Application のエンド・ユーザーの場合、保護モードで接続されるように指定できます。< ホスト名_インスタンス名 > には、データベースのホスト名とアプリケーション・サーバーのインスタンスを指定します。	cs=[APPS_SECURE]genledger_payables
db=< データベース名 >	接続するデータベースを指定します。	db=video
eul=<EUL 名 >	接続する際の EUL の名前を指定します。このパラメータを指定するのは、デフォルトの EUL を変更する場合のみです。	eul=myEUL

表 4-2 Discoverer Viewer の URL 引数と値（続き）

引数と値	目的	例
fm=<エクスポート形式>	次のエクスポート形式のいずれかを指定します。 ■ xml ■ xls（Microsoft Excel） ■ htm（HTML） ■ txt（テキスト） ■ csv（カンマ区切りの値） ■ prn（書式化されたテキスト） ■ dcs（Express） ■ dif（データ交換形式） ■ slk（記号リンク） ■ wks（Lotus 1-2-3） ■ sql（構造化問合せ言語） 注意： このパラメータを使用してデータをエクスポートするには、問合せを実行する必要があります。	fm=xls
lm=<applications または discoverer>	ログイン・メソッドを指定します。 ■ 「applications」と設定すると、Oracle Applications ユーザーでログインします。 ■ 「discoverer」と設定すると、Discoverer ユーザーでログインします。 このパラメータを指定しない場合、デフォルトのログイン・メソッド（viewer_config.xml 構成ファイルで指定したメソッド）が使用されます。	lm=applications
nlsl=<言語>	使用する言語の言語コードを指定します。	nlsl=en-gb
pg=<ページ番号>	表示するワークシートのページを指定します。	pg=4
pi_<ページ・アイテム名>=<ページ・アイテムの値>	ページ・アイテムの名前と、そのページ・アイテムに設定する値を指定します。	pi_Region=West 「Region」ページ・アイテムに「West」を設定します。

表 4-2 Discoverer Viewer の URL 引数と値（続き）

引数と値	目的	例
pw=<パスワード>	指定したデータベースへの接続で使用するユーザー名に対応するパスワードを指定します。	pw=tiger
qp_<パラメータ名>=<パラメータ値>	問合せパラメータの名前と、そのパラメータに設定する値を指定します。	qp_City=Denver
qpd=<数秒>	「Query Progress」ページが表示（または再表示）されるまでのディレイを秒単位で指定します。この引数を使用すると、Viewer Servlet を使用して作成したアプリケーションヘッダーを戻すときに、「Query Progress」ページを処理するためのコードを書き込む必要がなくなります。	qpd=60 この指定では、問合せが 60 秒間実行された後、「Query Progress」ページが表示されます。「Query Progress」ページは、問合せ結果が得られるまで 60 秒ごとに再表示されます。
rd=<数秒>	ブラウザが元の URL にリダイレクトされるまでのディレイを秒単位で指定します。この引数は、エクスポート操作中にブラウザがタイムアウトすることを防ぐために使用します。この引数は、fm 引数にも有効な値が指定されている場合にのみ適用されることに注意してください。	rd=90 この指定では、エクスポート操作が完了するまで、90 秒ごとにブラウザが元の URL にリダイレクトされます。
rs=<職責>	接続する際の Oracle Applications 職責を指定します。	Rs=Manager
sg=<セキュリティ・グループ>	接続する際の Oracle Applications セキュリティ・グループを指定します。	Sg=securityGroup
us=<ユーザー名>	使用するユーザー名を指定します。	us=video4
us=<接続文字列>	ユーザー名およびデータベースの他に、オプションの接続情報を指定します。接続文字列では、次の構文を使用します。 <username>[:<responsibility>[:<securityGroup>]][/<password>]&@<database>	us=scott/tiger@disco
wb=<ワークブック名>	開くワークブックの名前（または ID）を指定します。	wb=My+Workbook
wbd=<ワークブック日付>	スケジュールされたワークブックを開きます。スケジュールされたワークブックが実行された日付を指定します。	wbd=31-OCT-00

表 4-2 Discoverer Viewer の URL 引数と値（続き）

引数と値	目的	例
wbt=< ワークブック時刻 >	スケジュールされたワークブックを開きます。スケジュールされたワークブックが実行された時刻を指定します。wb パラメータを使用してスケジュール・ワークブックの ID を指定した場合には、このパラメータを指定する必要はありません。	wbt=4%3a00+PM 4:00 PM を指定します。
ws=< ワークシート名 >	開くワークシートを指定します。	ws=My+Worksheet

Discoverer Viewer のカスタマイズ

Discoverer Viewer の容易なカスタマイズ機能により、Web サイトの外観の適合、企業のロゴやその他の図版の組込み、または Web 向けの Discoverer カスタム・アプリケーションの作成を容易に行うことができます。

この章では、次のトピックについて説明します。

- [カスタマイズした Web アプリケーション向けの Discoverer Viewer の使用](#)
- [カスタマイズの例](#)

5.1 カスタマイズした Web アプリケーション向けの Discoverer Viewer の使用

5.1.1 一般的な外観のカスタマイズ

単一のカスタマイズ・ファイルで HTML 書式設定の属性を指定し、Discoverer Viewer の外観をカスタマイズできます。HTML 書式設定の知識があれば、フォント、色およびグラフィックを容易に変更できます。

5.1.2 アプリケーションのカスタマイズ

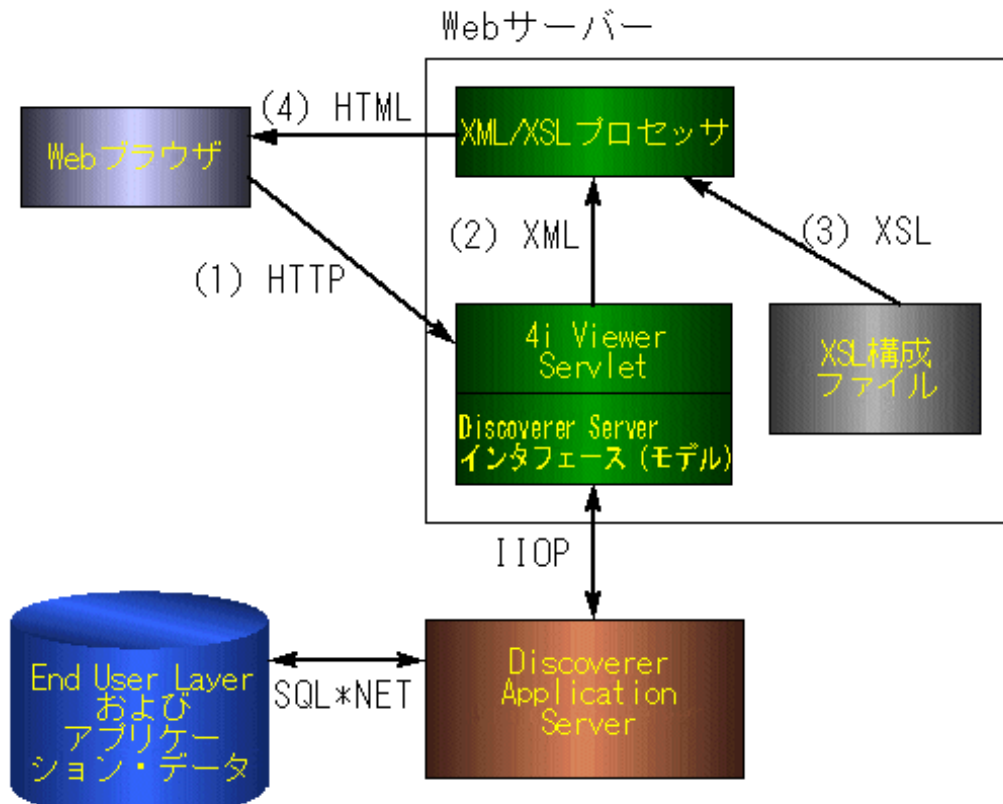
Discoverer Viewer を使用して、完全なカスタム Web アプリケーションを構築したり、他の中間層 Web システムにデータを伝送できます。このコンポーネントでは、データおよびアプリケーションの状態の表示に業界標準の XML、ユーザー・インタフェースの書式設定に XSL スタイルシート言語を使用しています。標準の XSL ツールを使用すると、ユーザー・インタフェースのカスタマイズや、埋込み型のビジネス・インテリジェンス・アプリケーションの作成ができます。

5.1.3 HTML の作成方法

Discoverer Viewer の HTML は、使用可能な情報を示す XML と HTML での情報の表示方法を定義する XSL との結合により作成されます。

XSL は、使用するフォントや色などの単純な属性を定義するのみでなく、各ページのレイアウトやユーザーとの対話も可能にします。XSL をカスタマイズすることで、特定の Discoverer Application を Web 上に作成し配布することができます。

図 5-1 XSL/XML を使用した Discoverer Viewer での HTML の作成



5.1.4 特定の Discoverer Application の作成に必要な HTML 作成のステップ

5.1.4.1 ステップ 1 ブラウザによる URL の送信

Discoverer Viewer ユーザーがログインした後、業務分析を行うために、開く権限を与えられているワークブックの一覧を要求したとします。発行される URL は次のとおりです。

<http://myserver.mycompany.com/Discoverer4i/Viewer?us=video&db=Disco>

この URL で、サーブレットがインストールされているマシン、および使用するユーザー名とデータベースの接続文字列が指定されます。パスワードは通常、セキュリティ保護のため URL に表示されません。

5.1.4.2 ステップ 2 XML の作成

Discoverer Servlet により URL が処理されると、Discoverer Server にこのユーザーのセキュリティ設定を確認するよう指示が行われます。確認後、このユーザーがアクセス権を付与されたワークブックの詳細が戻されます。セキュリティ設定は、データベース内の End User Layer 表に保管されています。サーバーから情報が戻されると、Servlet により次の XML が作成されます。

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="example1.xsl"?>
<discoverer version="4.1.x.x" login_method="discoverer">
  <request source="http://myserver.mycompany.com/servlets/viewer">
    <command name="db">disco</command>
    <command name="us">VIDEO</command>
  </request>
  <account>
    <user>VIDEO</user>
    <database>disco</database>
    <eul default="true" name="VIDEO">
      <workbook name="Annual Sales Report" ref="Annual~20Sales~20Report">
        <description>Shows yearly and quarterly sales of products</description>
      </workbook>
      <workbook name="Store and Band Analysis"
ref="Store~20and~20Band~20Analysis">
        <description>Shows sales by Store, broken into sales bands</description>
      </workbook>
      <workbook name="Video Sales Analysis" ref="Video~20Sales~20Analysis">
        <description>General purpose analysis of the Business </description>
      </workbook>
    </eul>
    <option name="qpw" enable="false">60</option>
    <option name="qrl" enable="false">10000</option>
    <option name="qtl" enable="false">1800</option>
    <option name="msa" enable="false">60</option>
    <option name="qif">250</option>
    <option name="qll">15</option>
    <option name="aq">true</option>
    <option name="nv">NULL</option>
    <option name="ftd">true</option>
    <option name="rpp">25</option>
  </account>
  <export name="xls" format="application/vnd.ms-excel">Microsoft Excel Workbook
(*.xls)</export>
```



```
<export name="htm" format="text.html">Hyper-Text Markup Language (*.htm)</export>
<export name="txt" format="text.plain">Text (Tab delimited) (*.txt)</export>
<export name="csv" format="text.plain">CSV (Comma delimited) (*.csv)</export>
<export name="prn" format="text.plain">Formatted Text (Space delimited) (*.prn)</
export>
<export name="dcs" format="text.plain">DCS (Express Format) (*.dcs)</export>
<export name="dif" format="application.vnd.ms-excel">DIF (Data Interchange
Format) (*.dif)</export>
<export name="slk" format="application.vnd.ms-excel">SYLK (Symbolic Link)
(*.slk)</export>
<export name="wks" format="application.vnd.ms-excel">WKS (Lotus 1-2-3) (*.wks)</
export>
</discoverer>
```

ここでは、「Store and Band Analysis」、「Video Sales Analysis」および「Annual Sales Report」の3つのワークブックに関する情報が戻されたことがわかります。XMLには、ワークブックの名前や内容をユーザーに表示する方法を判断する機能はありません。その処理はXSLファイルで行われます。

5.1.4.3 ステップ3 XSLの適用

XSLは、World Wide Web Consortiumにより業界標準に指定されているスタイルシート言語です。XSLはXMLファイルから要素を選択してHTMLテンプレートと結合させ、Webブラウザ用のHTML出力を生成します。Discoverer Viewerのユーザー・インタフェースは、すべてXSLで定義されています。つまり、HTMLエディタやXSLエディタなど標準のWeb開発ツールを使用してユーザー・インタフェースをカスタマイズまたはコピーし、ユーザー・インタフェースを変更できます。XSLおよびXMLはXSLプロセッサで結合され、その後でHTMLが作成されます。

5.1.4.4 ステップ4 HTMLの作成

前述のXMLを受けて、標準のDiscoverer ViewerのXSL構成ファイルにHTMLが作成されます。作成されたHTMLは、最初のURLに応答する形でブラウザに戻されます。Discoverer Viewerでは、使用するブラウザまたはインターネット・デバイスに対し最小限のHTMLを作成します。これにより、Discoverer Viewerをその他のWebアプリケーションまたはポータルへ統合しやすくなります。

5.2 カスタマイズの例

ここでは、Discoverer Viewer のユーザー・インタフェースのカスタマイズについて説明します。カスタマイズを行うことにより、フォント、色およびグラフィックなどの基本的な属性の変更から、カスタム・ユーザー・インタフェースを使用した独自の外観の作成まで実現可能となります。

5.2.1 カスタマイズに必要なファイル

クイック・カスタマイズまたは**フル・カスタマイズ**を実行するには、次に示す **html/** ディレクトリに格納されたファイルが必要です。これらのディレクトリは、Discoverer Viewer を Web サーバーのルート・ディレクトリにインストールする際に作成されます。

```
disco4iv/
  disco4iv.jar
  html/
    disco4iv.xml
    disco4iv.xsl
    errors.xsl
    functions.xsl
    gui_components.xsl
    page_layouts.xsl
    render_table.xsl
    scripts.xsl
    style.xsl
    viewer_config.xml
  demos/
  help/
  images/
  resources/
```

Video Stores Demo
HTML ヘルプ・ファイル
ユーザー・インタフェースおよびヘルプ
別の言語による Message ファイル

5.2.2 クイック・カスタマイズ

企業の標準に合せたフォントや色の変更、ロゴの表示および商標の追加を簡単に行うことができます。このような変更処理はすべて、Servlet ディレクトリ (disco4iv¥) にある単一の XSL スタイルシート・ファイル **style.xsl** で一括して行います。このファイルの特殊なタグを使用して、各種スタイルを変更できます。

また、事前定義済みの一連の表示オプションを指定するための表示テンプレートを **viewer_config.xml** ファイルに作成することもできます。このような表示オプションを使用することにより、ページ・ヘッダー、ワークシート領域およびページ・フッターの表示を制御できます。エンド・ユーザーは、「表示オプション」ページで使用する表示テンプレートを選択します。**viewer_config.xml** ファイルには、「Corporate Look」という表示テンプレートがサンプルとして用意されています。

style.xsl および **viewer_config.xml** を変更するには、サーバー側の XSL ファイルにアクセスする必要があります。

style.xsl および **viewer_config.xml** で全体的なスタイルを変更することはできますが、ユーザー・インタフェースの包括的な操作方法自体は変更されません。この他に、XSL を使用し

て Discoverer Viewer をカスタマイズすることもできます。この方法を使用すると、完全にカスタマイズされたアプリケーションを作成できます。詳細は、[5.2.3 項「フル・カスタマイズ」](#)を参照してください。

5.2.2.1 クイック・カスタマイズの属性

style.xml ファイルに含まれるカスタマイズ可能な属性は次のとおりです。

- 会社のロゴ
- テキストの色
- ヘッダー
- フォント
- スタイル
- リンク
- イメージ

viewer_config.xml ファイルでは、「Presentation Options」ページで、異なった選択を使用可能にする設定を指定できます。

5.2.2.2 クイック・カスタマイズの手順

クイック・カスタマイズを実行するには、次の手順を完了してください。

1. style.xml ファイルをテキスト・エディタで開きます。ファイルは、Servlet ディレクトリ（¥disco4iv¥html）に格納されています。
2. 各変数の <山カッコ> 内のテキストを目的に沿うように編集します。詳細は、ファイル内のコメントを参照してください。
3. style.xml の変更を保存し、ファイルを閉じます。
4. viewer_config.xml ファイルをテキスト・エディタで開きます。ファイルは、Servlet ディレクトリ（¥disco4iv¥html）に格納されています。
5. 次の項をコピーします。

```
<look name="Corporate Look">
<!-- Logo -->
<component name="log" state="false"/>
.
.
.
<!-- "Show XML" Link -->
<component name="xml" state="false"/>
</look>
```

6. コピーした内容を `</layout_templates>` タグの直前に貼り付けます。
7. "Corporate Look" を任意のテンプレート名に変更します。ここで付けた名前はユーザー側に表示されることに注意してください。
8. 必要に応じて、コンポーネントの値を TRUE または FALSE に変更します。
9. `viewer_config.xml` の変更を保存し、ファイルを閉じます。

5.2.2.3 style.xsl ファイルの編集例

ファイル編集の詳細は、ファイル内のコメントを参照してください。

ロゴを挿入するには

```
<xsl:variable name="logo_src"> </xsl:variable>
```

を次のように変更します。

```
<xsl:variable name="logo_src"> http:www.mycompany.com/images/mylogo.gif </xsl:variable>
```

テキストの色を変更するには、

```
<xsl:variable name="text_color">#000000</xsl:variable>
```

を変更して、適切な色コードを追加します。

5.2.3 フル・カスタマイズ

この項では、Discoverer Viewer Servlet のユーザー・インタフェースを完全にカスタマイズする方法について説明します。独自の外観のユーザー・インタフェースを作成します。

この項は、次のトピックで構成されています。

- [背景](#)
- [前提条件](#)
- [ファイル構造](#)
- [Servlet のプログラミング・インタフェース](#)
- [外観のカスタマイズ - XML/XSL の例](#)

5.2.3.1 背景

Discoverer Viewer Servlet のインタフェースをカスタマイズする前に、このコンポーネントの機能について理解しておくくと便利です。

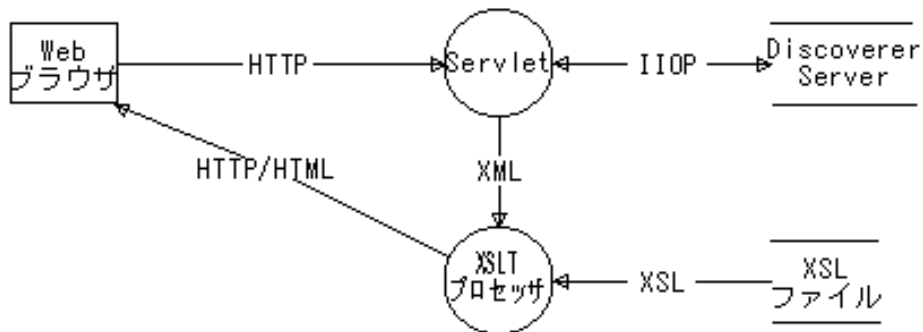
次に、要求が送信されたときに発生するデータ・フローについて説明します。

- クライアント（Web ブラウザ）から HTTP を使用する Servlet に要求が送信されます。通常、要求は URL でエンコードされますが、Servlet では GET および POST にも対応しています。
- Servlet がこの要求を解釈し、必要なデータを Discoverer Server から取り出します。
- Discoverer Server から戻されたデータが Servlet で XML 形式に変換されます。
- XML ファイルが XSLT プロセッサに送信されます。ここで、XSL テンプレートを使用して最終的な出力（通常は HTML）に変換されます。
- 要求を送信したクライアントに出力が戻されます。

このモデルでは、Servlet で処理されるアプリケーション・ロジックと XSL ファイルで処理される表示ロジックが明確に分かれています。したがって、XSL ファイルを編集すれば、ユーザー・インタフェースの外観を完全に変更することができます。

図 5-2 に、Discoverer Viewer のデータ・フローについて示します。

図 5-2 Discoverer Viewe のデータ・フロー



5.2.3.2 前提条件

Discoverer Viewer のユーザー・インタフェースを独自にカスタマイズするには、次の条件を前提とします。

- HTML に関する知識
- XML、XSLT および XPath に関する知識
- サーバーの XSL ファイルにアクセスできること
- Servlet の構成ファイルにアクセスできること

5.2.3.3 ファイル構造

Servlet の構成ファイルでは、Servlet により作成される XML に適用される XSL ファイルの名前および場所が指定されています。XSL ファイルはさらに、各種 XSL ファイルをインポートして、表示ロジックに必要なテンプレートをすべてまとめます。また XML ファイルは、追加構成オプションの提供でも使用されます。これらのファイルについて、次に説明します。

5.2.3.3.1 disco4iv.xsl

マスター XSL ファイルです。機能は次のとおりです。

- Servlet の構成ファイルにある一定のパラメータ・セットの値の取得
- クライアントから送信された Presentation コマンドの値の取得
- 他の XSL ファイルのインポート
- XML の検査および情報の表示に最適なページ・レイアウトの選択

5.2.3.3.2 page_layouts.xsl

このファイルには、表示可能な各ページの定義が含まれています。各定義は、一連の GUI コンポーネントで構成されます。コンポーネントは、ページに表示される順序で並んでいます。

定義済みのページ・レイアウトは次のとおりです。

- 「Connect」 ページ
- 「Reconnect」 ページ
- 「Choose Oracle Applications Responsibility」 ページ
- 「Options」 ページ
- 「Choose Workbook」 ページ
- 「Prepare to Run Query」 ページ
- 「Modify Query Parameters」 ページ

- 「Query Progress」 ページ
- 「Data Available」 ページ
- 「Export Worksheet」 ページ
- 「Presentation Options」 ページ
- 「Edit Layout」 ページ

5.2.3.3.3 gui_components.xsl

このファイルには、ページの作成に使用される、`page_layouts.xsl` と呼ばれる GUI コンポーネントが含まれています。GUI コンポーネントの例を次に示します。

- Page Title
- Page Footer
- List of Worksheets
- Options Form
- Query Progress Meter

5.2.3.3.4 style.xsl

このファイルは、ヘッダーなどのテキストやグラフィックの表示を定義することで、スタイルシートとして機能します。詳細は、ファイル内のコメントを参照してください。

5.2.3.3.5 functions.xsl

このファイルには、共通のタスクを実行する一連の機能が含まれています。たとえば、ハイパーテキスト・リンク作成時の `href` 属性の作成などです。

5.2.3.3.6 errors.xsl

このファイルは、Discoverer Server で生成されたエラー、警告またはメッセージを解釈し、わかりやすい形式で表示します。

5.2.3.3.7 scripts.xsl

このファイルには、JavaScript 拡張ユーザー・インタフェースを使用する場合に必要な JavaScript の部分が含まれています。詳細は、[2.7 項「Discoverer Viewer JavaScript 拡張ユーザー・インタフェースの利用」](#)を参照してください。

5.2.3.3.8 render_table.xml

このファイルは、テーブルおよびクロス集計の作成時に使用します。詳細は、ファイル内のコメントを参照してください。このファイルを編集する必要はありません。

5.2.3.3.9 viewer_config.xml

このファイルでは、次の項目が指定されています。

- 次に示す制御の表示の有無
 - 接続ダイアログの「言語」ドロップダウン・リスト
 - 「Applications ユーザー」チェックボックス
 - 各ページ下部の XML リンクの表示
- JavaScript 拡張ユーザー・インタフェースの使用の有無
- 名前を付けられた一連の表示オプションを定義する表示テンプレート。「Corporate Look」と呼ばれる表示テンプレートは、サンプル・ファイルに含まれています。
- 特定の言語で使用するメッセージ・ファイルおよびヘルプ・ファイル。このファイルでは、接続ダイアログの「言語」ドロップダウン・リストの表示順序に従って、言語が表示されます。

5.2.3.4 Servlet のプログラミング・インタフェース

5.2.3.4.1 Servlet からのデータ受信

Servlet から XSL ファイルに情報を渡すには、次の 2 通りの方法があります。

- XML
XML には、エンド・ユーザーの要求に対して Discoverer Server から戻される情報がすべて含まれます。
- XSL パラメータ
この方法は、disco4iv.xml ファイルおよび Presentation コマンドのパス・エントリを URL から XSL に渡す場合に使用されます。

5.2.3.4.2 Servlet へのデータ送信

前述のように、Servlet はクライアント (Web ブラウザ) から受信したコマンドにより駆動します。これらのコマンドを送信するには、URL でパラメータを使用して Servlet を呼び出すか、アクションとして Servlet を定義する HTML を作成します。どちらの方法でも HTTP を使用します。

アンダースコア () を使用することで、独自の Presentation コマンドを作成できます。コマンドの先頭にアンダースコアが使用されている場合、XSL パラメータとして直接 XSL ファイルに渡されます。たとえば、URL で `_filter=sales` をパラメータとして指定すると、Servlet の設定の XSL パラメータ `"url_filter"` が `"sales"` になります。このパラメータを使用して、ワークブック・リストをフィルタすることができます。

5.2.3.5 外観のカスタマイズ - XML/XSL の例

XSL について、わかりやすく例示します。次に示す XML および XSL のフラグメントを使用すると、Web ブラウザのカスタマイズを実験することができます。

5.2.3.5.1 手順 1 - XML ファイル

実際に扱うデータは標準の XML ファイルです。次に類似した例を示します。

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="example1.xsl"?>
<discoverer4iv version="4.1.x.x" state="login"
url="http://myserver.mycompany.com/disco/disco4iv" login_method="disco_login">
  <account nv="NULL" aq="confirm">
    <user>video</user>
    <database>disco</database>
    <eul>
      <name>VIDEO</name>
      <workbook@name="Store and Band Analysis">
        <description>Shows sales by Store, broken into sales bands</description>
      </workbook>
      <workbook@name="Video Sales Analysis">
        <description>General purpose analysis of the Business </description>
      </workbook>
      <workbook@name="Annual Sales Report">
        <description>Shows yearly and quarterly sales of products</description>
      </workbook>
    </eul>
  </account>
</discoverer4iv>
```

1 行目は、XML のバージョンを指定しています。2 行目は、データ処理に適用される XSL ファイル example1.xsl を指定しています。残りの部分は Discoverer Viewer から作成されています。最初の 2 行をここに追加したことにより、テキスト・エディタを使用してファイルにテキストを入力し、編集後に Web ブラウザで開いて XSL の変更内容を視覚的に確認することができますようになります。この操作を行うには、拡張子を xml に指定します。ただし、その場合は Internet Explorer 5 以上が必要です。

5.2.3.5.2 手順 2 - XSL ファイル

XSL ファイル (example1.xsl) は次のようになっています。

```
<?xml version='1.0'?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/TR/WD-xsl">
  <xsl:template match="/">
    <HTML>
      <BODY bgcolor="#FFFFFF" link="#663300" text="#000000">
        <b><i>Choose a Workbook :</i></b><P></P>
```

```
<TABLE BORDER="2">
  <xsl:for-each select="/discoverer/account/eul/workbook">
    <TR>
      <TD width="242"><font face="Arial"><b><a href="link.htm">
        <xsl:value-of select="./@name"/></a></b></font>
      </TD>
      <TD>
        <xsl:value-of select="description"/>
      </TD>
    </TR>
  </xsl:for-each>
</TABLE>
</BODY>
</HTML>
</xsl:template>
</xsl:stylesheet>
```

5.2.3.5.3 手順 3 - XML+XSL=HTML

XML ファイルを Internet Explorer 5 以上の Web ブラウザで開くと、XSL スタイルシートに読み込まれ、次のような HTML が作成されます。

図 5-3 XML および XSL を使用したワークブック・リストの例

<i>Choose a Workbook :</i>	
Store and Band Analysis	Shows sales by Store, broken into sales bands
Video Sales Analysis	General purpose analysis of the Business
Annual Sales Report	Shows yearly and quarterly sales of products

XSL ファイルをよく見ると、HTML が作成される過程がわかります。XML のバージョンを指定してファイルを再度開くと、2 行目にはこのファイルがスタイルシートであることが記載されます。4 行目の HTML タグ以降が <HTML> テンプレートです。

```
<BODY bgcolor="#FFFFFF" link="#663300" text="#000000">
```

この行では、使用する色が定義されています。

```
<b><i>Choose a Workbook:</i></b><P></P>
```

HTML でボールド・イタリックのフォントおよび「Choose a workbook:」という文字を挿入することが指定されています。

```
<TABLE BORDER="2">
```

```
<xsl:for-each select="/discoverer/account/eul/workbook">
```

これで、2 の罫線を使用した HTML テーブルが起動します。次の行は、最初に実際に表示される XSL です。この行の意味は次のとおりです。

「XML データ・ファイルにアクセスし、各ワークブックのタグについて、最後のタグ `</xsl:for-each>` に到達するまですべてのステップを実行する。」

つまり、XML ファイルに表示される各ワークブックについて次の XSL が処理され、検出されたすべてのワークブックの HTML テーブルに行が挿入されます。

```
<TR>
  <TD width="242"><font face="Arial"><b><a href="link.htm">
    <xsl:value-of select="@name"/></a></b></font>
  </TD>
  <TD>
    <xsl:value-of select="description"/>
  </TD>
</TR>
```

`<TR>` により、テーブルの新しい行が開始します。`<TD ... >` は、1 列目に挿入されるテーブル・データを定義するタグです。ここでは、列幅が 242 ピクセル、フォントが Arial に設定されています。次のハイパーリンクには、link.htm ファイルをクリックするとジャンプする場所が定義されています。Discoverer Viewer では、このリンク先は動的に作成されますが、わかりやすくするため、ここでは静的なリンクとして表示しています。

```
<b><a href="link.htm"><xsl:value-of select="@name"/></a></b></font>
```

XSL 行により、各ワークブック要素の name 属性に対し、XML ファイルからテキストが挿入されます。

```
<TD>
  <xsl:value-of select="description"/>
</TD>
```

この行により、HTML テーブルの 2 列目が定義され、XML ファイルの `<description>` 要素を使用してワークブックの内容のテキストが挿入されます。つまり、HTML テーブルの各列にはワークブックの名前、リンク先およびワークブックの内容がテキストで表示されます。この XML ファイルには 3 つのワークブックが含まれているため、テーブルの列数は 3 です。

この例は、Discoverer Viewer でのワークブック・リストの表示方法を正確に示したものではありません。わかりやすくするために簡略化し、XSL がどのように出力表示を制御するかを図示しています。また、Discoverer Viewer では、XML および XSL が結合されます。この処理が行われるのは Web ブラウザではなく中間層の XSL プロセッサです。

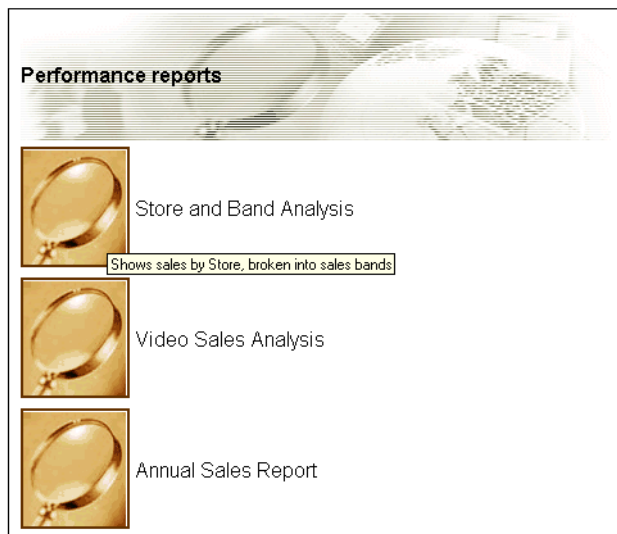
5.2.3.5.4 手順 4 - XSL ファイルのカスタマイズ

XSL ファイルを次のように変更できます。

```
<?xml version='1.0'?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/TR/WD-xsl">
  <xsl:template match="/">
    <HTML>
      <BODY bgcolor="#FFFFFF" link="#663300" text="#000000">
        <b><i>Choose a Workbook :</i></b><P></P>
        <TABLE BORDER="0">
          <TR>
            <TD width="500" height="100" background="../../images/disco_banner.gif">
              <font face="Arial"><b>Performance Reports</b></font>
            </TD>
          </TR>
        </TABLE>
        <TABLE border="0">
          <xsl:for-each select="/discoverer/account/eul/workbook">
            <TR>
              <TD width="242">
                <a href="link.htm">
                  
                    <xsl:attribute name="alt">
                      <xsl:value-of select="description"/>
                    </xsl:attribute>
                  </img>
                </a>
              </TD>
              <TD>
                <font face="Arial">
                  <xsl:value-of select="@name"/>
                </font>
              </TD>
              <TD>
                <xsl:value-of select="description"/>
              </TD>
            </TR>
          </xsl:for-each>
        </TABLE>
      </BODY>
    </HTML>
  </xsl:template>
</xsl:stylesheet>
```

ファイルが同じ XML と結合されると、次のように表示されます。

図 5-4 同じ XML で変更した XSL スタイルシートを使用した場合のワークブック・リスト



これでユーザー・インタフェースは完全に変更されました。以前よりグラフィックが多く使用されています。レポート作成用のテキストのリンクがグラフィカルなボタンに変更され、ボタンにマウス・カーソルを置くとヒントが動的に表示されます。

XSL ファイルの内容は次のように変更されています。

```
<TABLE BORDER="0">
  <TR>
    <TD width="500" height="100" background="../../images/disco_banner.gif">
      <font face="Arial"><b>Performance Reports</b></font>
    </TD>
  </TR>
</TABLE>
```

この行によりテーブルが作成され、グラフィックおよび Performance Reports というヘッダーが挿入されます。

```
<TABLE BORDER="0">
  <xsl:for-each select="/discoverer/account/eul/workbook">
    <TR>
```

これによりメイン・テーブルが起動し、ワークブック名が以前と同じように表示されます。ただし、テーブルは罫線で囲まれておらず、列の定義も変更されています。

```
<TD width="242">
  <a href="link.htm">
    
      <xsl:attribute name="alt">
        <xsl:value-of select="description"/>
      </xsl:attribute>
    </img>
  </a>
</TD>
```

テーブル・データの 1 列目は以前と同様にハイパーリンクとして定義されます。ただし今回は、テキストのかわりに **start1.gif** というイメージがリンクに使用されています。イメージにカーソルを置くとヒントが表示されるようにするために、HTML の ALT 属性が使用されています。

通常、ALT 属性は次のような簡単な文字列で使われます。

```

```

ここでは、ヒント (Tooltip) を動的にするため、XML ファイルの **<description>** 要素のテキストを使用して ALT タグを作成しています。**<xsl:attribute>** タグはこの操作を行うためのものです。

最後に、

```
<TD>
  <font face="Arial">
    <xsl:value-of select="@name"/></font>
</TD>
```

テーブルの 2 列目は、単に表示するワークブックの名前を選択して表示します。以前と同様に XSL を使用して XML ファイルから名前を取り出します。

5.2.4 Video Stores Demo - 代替ユーザー・インタフェースの例

Video Stores demo は Discoverer Viewer とともにインストールされます。表示されるようにするには、次の手順を完了してセットアップしてください。

Discoverer Viewer が完全にインストールされている場合、Video Stores demo を実行するように Web サーバーを構成できます。

Video Stores demo では、Discoverer Viewer のユーザー・インタフェースがどのようにカスタマイズできるのかを、サンプルの「Video Store」を使用して説明します。これを使用すれば、ユーザー・インタフェースの基盤の詳細を理解することができます。

5.2.4.1 インストール内容

インストールの実行中、Video Stores のデモ・ディレクトリ (**demos¥video**)、サブディレクトリおよびファイルが作成されます。**demos¥video** ディレクトリは、`<iSUITES_HOME>¥apache¥apache¥htdocs¥disco4iv¥html` ディレクトリの下にインストールされます。

Video Stores のデモ・ディレクトリ (**demos¥video**) には、次のファイルがインストールされます。

- **disco4iv.xml**
このファイルを編集して、正しいセッション名を参照しているかどうかを確認できます。
disco4iv.xml ファイルには、構成情報が含まれています。詳細は、disco4iv.xml の箇所を参照してください。
- **video.xsl**
このファイルを編集して、新しいユーザー・インタフェースを作成できます。
このファイルには、フォント、イメージ、背景および領域の詳細など、ユーザー・インタフェースの外観を制御する定義がすべて含まれています。
- **render_table.xsl**
このファイルを編集する必要はありません。
このファイルは、テーブルおよびクロス集計の作成時に使用します。詳細は、ファイル内のコメントを参照してください。
- ワークブック - Video Stores Demo で使用されます。詳細は、[5.2.4.2 項「セットアップの手順」](#)を参照してください。
 - Food and Drink.dis
 - Games.dis
 - Movies.dis

demos¥video¥images ディレクトリには、ユーザー・インタフェースに使用されるイメージがすべて含まれています。ユーザー・インタフェースに必要な新しいイメージはすべてここに保存されます。

5.2.4.2 セットアップの手順

次の手順を実行してください。

- Video Stores Demo のデータベース・ユーザーを作成します。
新規データベース・ユーザーを作成する方法については、『Oracle Discoverer Administration Edition 管理ガイド』を参照してください。
- 前の手順で作成した Video Stores Demo ユーザーに Video Stores Tutorial をインストールします。
詳細は、『Oracle Discoverer Administration Edition 管理ガイド』のチュートリアル of theインストールの項を参照してください。

- Video Stores Demo ユーザーの権限を使用して、Discoverer Plus で Video Stores Demo のワークブック 3 つをデータベースに保存します。
- **video.xml** ファイルを編集して、新しく作成した Video Stores Demo ユーザーに対して、**ユーザー名**、**パスワード**および**データベース名**を指すようにします。現行の **video.xml** ファイルは、ユーザー名 **us=videodemo**、パスワード **ps=videodemo** およびデータベース **db=disco** を指すようにセットアップされています。これで、**video.xml** ファイルの変更は終了です。

5.2.4.3 Web サーバーの設定

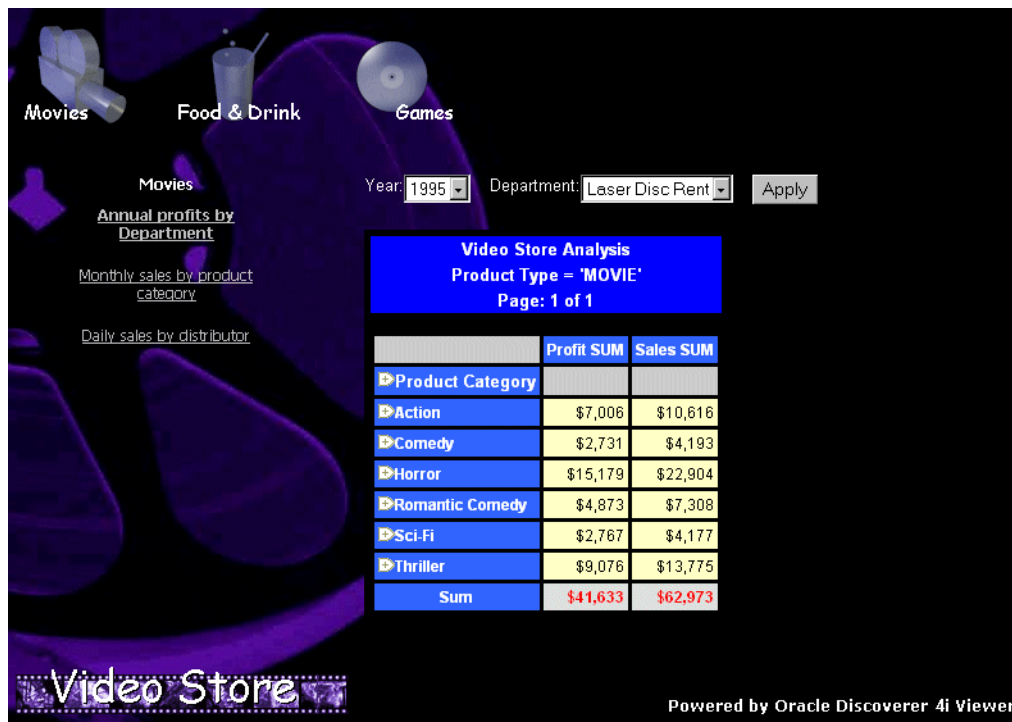
Web サーバーを、ゾーン名を使用して構成する必要があります。Video Stores Demo の新規ゾーンをセットアップして、¥demos¥video¥disco4iv.xml（Video Stores Demo 構成ファイル）の場所を指定します。

5.2.4.4 Video Stores demo の実行

Web ブラウザを開き、次の URL を入力します。

- <http://yourmachineURL/'Your new Video Demo Zone'>

Video Stores demo が表示されます。



5.2.4.5 video.xml ファイルの変更

video.xml ファイルは、

<iSUITES_HOME>%apache%¥apache¥htdocs¥disco4i¥html¥demos¥video ディレクトリに格納されており、Discoverer View の Video Stores Customization の外観を指定します。

たとえば、**video.xml** ファイルを編集して背景などの表示イメージを変更できます。Demo で表示するワークブックを指定することも可能です。

video.xml ファイルを編集することにより、表示されるほぼすべての項目を変更できます。この機能を使用すれば、自社向けの外観を構築するのみでなく、データの一意の表示も定義できます。

5.2.5 その他のカスタマイズの例

5.2.5.1 自動ログイン

page_layouts.xml ファイルを変更して、Discoverer Viewer ユーザーが、URL 引数としてパスワードを入力せずに自動ログイン（ログイン画面を省略するなど）できるように設定できます。ただし、この機能を使用するには JavaScript 対応のブラウザが必要です。

5.2.5.1.1 手順 1: 新規テンプレートを page_layouts.xml に追加する

次の行を page_layouts.xml ファイルに追加します。

```
<xsl:template name="layout_auto_login">
  <html>
    <head><title>Auto Login</title></head>
    <body>
      <form method="post" name="autoLogin" action="{/discoverer/request/@source}">
        <!-- Specify the User Name in the value field below -->
        <input type="hidden" name="us" value="username"/>
        <!-- Specify the Password in the value field below -->
        <input type="hidden" name="pw" value="password"/>
        <!-- Specify the Database in the value field below -->
        <input type="hidden" name="db" value="database"/>
        <!-- Specify the Language Code in the value field below (en is English) -->
        <input type="hidden" name="nls1" value="en"/>
        <!-- To show a list of Database Workbooks, leave the following two value
fields set to "dwb" otherwise, set them to "swb" for Scheduled Workbooks. -->
        <input type="hidden" name="in" value="dwb"/>
        <input type="hidden" name="_in" value="dwb"/>
        <!-- The following line requests the user is logged in using the default
method (as set in viewer_config.xml) -->
        <input type="hidden" name="lc" value="false"/>
      </form>
      <script language="JavaScript">
        document.autoLogin.submit();
      </script>
    </body>
  </html>
</xsl:template>
```

5.2.5.1.2 手順 2: ユーザー名、パスワードおよびデータベースを指定する

コメントの指示に従って layout_auto_login テンプレートにユーザー名、パスワードおよびデータベースの情報を入力します。

5.2.5.1.3 手順 3: disco4iv.xsl を変更する

disco4iv.xsl で次のテキストを検索します。

```
<xsl:call-template name="layout_connected"/>
```

このテキストを次のテキストに置換します。

```
<xsl:call-template name="layout_auto_login"/>
```

5.2.6 カスタマイズの概要

Discoverer Viewer では、XML および XSL を強力なカスタマイズ環境として採用しています。簡単な書式設定の変更は単一のスタイルシート・ファイルで実行でき、高度なカスタム・アプリケーションの作成は、XSL を使用した別の方法で実行できます。このように、Discoverer Viewer には、ビジネス・インテリジェンスを他の Web アプリケーションに直接統合する機能が備えられています。

Oracle9iAS Discoverer における SSL サポート

この章では、Secure Sockets Layer (SSL) プロトコルで Discoverer Plus および Discoverer Viewer を構成する方法について説明します。

説明するトピックは次のとおりです。

- [詳細情報](#)
- [SSL および Discoverer](#)
- [SSL について、および SSL を使用する理由](#)
- [SSL の動作方法](#)
- [公開鍵暗号化技術について](#)
- [SSL の認証および証明書](#)
- [次に必要なステップ](#)
- [Discoverer Plus で SSL を使用するための構成](#)
- [HTTPS を使用した Discoverer Viewer の構成](#)

6.1 詳細情報

この章には、『Visibroker SSL Pack 3.3 Programmer's Guide』からの引用が含まれています。

Visibroker の SSL 機能の詳細は、次に記載する Borland Software Corporation のインターネット・サイトにアクセスしてください。

<http://www.borland.com/techpubs/books/addons/vbssl/vbssl33/index.html>

6.2 SSL および Discoverer

Discoverer Plus および Discoverer Viewer を、SSL プロトコルを使用するように構成できます。

- Discoverer Viewer では、Apache に基づいた Oracle HTTP Server の SSL 機能を使用して SSL をサポートしています。
- Discoverer Plus では、HTTP サーバーおよび Visibroker Gatekeeper の SSL 機能を使用して SSL をサポートしています。Discoverer Plus を使用すると、サーバーのファイアウォールの外側にあるクライアントと Visibroker Gatekeeper の間で伝送されるデータを暗号化できます。

6.3 SSL について、および SSL を使用する理由

SSL プロトコルは、クライアントとサーバー間でセキュリティ保護された接続を確立することを目的とした業界標準プロトコルです。このプロトコルには次に記載するセキュリティ機能が備わっており、インターネットなどの保護されていないネットワークでも機密データを伝送することができます。

- 認証：クライアントがサーバーの身元を判別し、偽りのないことを確認します。サーバーでも、必要に応じてクライアント・アプリケーションの認証を行うことができます。
- プライバシー：クライアントおよびサーバー間で伝送されるデータが暗号化されます。したがって、第 3 者がこのメッセージを傍受しようとしても、解読することができません。
- 整合性：暗号化されたデータを受信すると、第 3 者により破損または変更されているかどうかわかります。

SSL が有効である場合、施錠された鍵のアイコン () またはそれと同等の記号（ブラウザによって異なる）がブラウザのステータス・バーに表示されます。また、Web ページの URL は `http://` ではなく `https://` で始まります。

6.4 SSL の動作方法

SSL では、公開鍵および共通鍵の技術を組み合わせてトランスポート層レベルで複数のセキュリティ・サービスを提供し、2 者間にセキュリティ保護された通信チャネルを構築します。通常、ここで言う 2 者は Web ブラウザと Web (HTTP) サーバーです。ただし、SSL はトランスポート層で動作するため、アプリケーションに依存します。したがって、Telnet、ファイル転送プロトコル (FTP: File Transfer Protocol) および HTTP (HyperText Transfer Protocol) などのプロトコルを階層の最上位に置くことができます。

6.5 公開鍵暗号化技術について

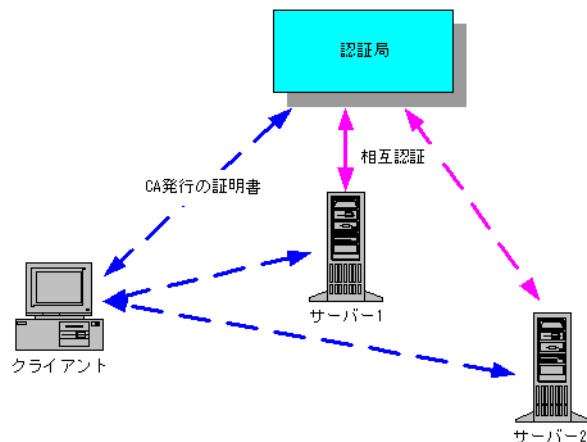
公開鍵暗号化では、公開鍵および秘密鍵の 2 つの鍵を組み合わせてメッセージの暗号化および解読を行います。この 2 つの鍵を使用すると、秘密鍵を使用して暗号化されたメッセージは、対応する公開鍵を使用した場合にのみ解読することができます。また反対に、公開鍵を使用して暗号化されたメッセージは対応する秘密鍵を使用した場合にのみ解読することができます。

6.6 SSL の認証および証明書

セキュリティ保護された接続を確立する前に、サーバー（およびクライアント）の認証が行われます。これは、正しいサーバーと通信していることを確認するためです。認証にはデジタル証明書または公開鍵証明書が使用されます。SSL で使用される証明書は ITU (International Telecommunications Union) の基準により定められた X.509 です。次の図を参照してください。

サーバーが適切な証明書を所有していることを証明するために、SSL では、VeriSign、Inc などの信頼ある認証局 (CA: Certificate Authority) によるデジタル署名が証明書に要求されます。証明書はパスポート、認証局は政府管轄のパスポート・オフィスのようなものと考えてください。政府管轄のパスポート・オフィスが確認したパスポート所有者の身元を信頼すると、その所有者に対してパスポートが発行されます。

図 6-1 認証局による証明書の確認



6.7 次に必要なステップ

6.7.1 Discoverer Plus での SSL の構成方法

1. 6.8 項「Discoverer Plus で SSL を使用するための構成」の手順に従ってください。
2. x509cert.jar ファイルがクライアント・マシンにインストールされていることを確認します。6.8.6 項「x509cert.jar ファイルのインストール」の手順に従ってください。
3. 必要な DLL ファイルがインストールされていることを確認します。詳細は 6.8.7 項「必要なダイナミック・リンク・ライブラリ (DLL) ・ファイルのインストール」を参照してください。
4. 6.8.8 項「Discoverer Plus の起動ページでの SSL の有効化」の手順に従って、Discoverer の起動ページで SSL を有効にします。

6.7.2 Discoverer Viewer での SSL の構成方法

1. 6.9 項「HTTPS を使用した Discoverer Viewer の構成」の手順に従ってください。
2. 6.9.5 項「Discoverer Viewer での SSL の有効化」の手順に従って、HTTPS プロトコルを使用して Discoverer のページで SSL を有効にします。

注意： Discoverer Viewer はファイアウォールに準拠した HTTP および HTTPS プロトコルを使用するため、Visibroker Gatekeeper は不要です。

6.8 Discoverer Plus で SSL を使用するための構成

6.8.1 Discoverer Plus の SSL のサポート

Discoverer Plus では、Visibroker Gatekeeper を使用して SSL をサポートしています。Visibroker Gatekeeper の詳細は、[7.5.1 項「Visibroker Gatekeeper」](#)を参照してください。

Discoverer Plus では、サーバーのファイアウォールの外側にあるクライアントに対してのみ SSL モードの通信をサポートしています。したがって、Discoverer Server コンポーネントに接続する場合は、Visibroker Gatekeeper を使用してください。サーバーのファイアウォールの内部にあるイントラネットに組み込まれたクライアントでは、Discoverer Server コンポーネントの通信に SSL を使用しません。

Discoverer Plus で SSL を使用するように、Visibroker Gatekeeper を構成する必要があります。

6.8.2 SSL を使用するための Visibroker Gatekeeper の構成方法

SSL を使用するように Visibroker Gatekeeper を構成するには、次の手順を実行してください。

1. ファイアウォールの構成に基づいて、Visibroker Gatekeeper を HTTP サーバー・マシンまたはそれ以外のマシンにインストールします。詳細は、[6.8.3 項「SSL を使用するための Visibroker Gatekeeper のインストール」](#)を参照してください。
2. 認証局から SSL の証明書を取得します。詳細は、[6.8.4 項「SSL の証明書の取得」](#)を参照してください。

企業または組織の証明書をすでに取得している場合は、Visibroker Gatekeeper で再利用することができます。

3. 認証局から SSL の証明書を取得した後、その証明書を Visibroker Gatekeeper にインストールします。詳細は、[6.8.5 項「SSL の証明書のインストール」](#)を参照してください。

注意：Gatekeeper に SSL の証明書をインストールしている場合、URL パラメータ ORBalwaysTunnel=yes を使用できません。詳細は、[7.9.2 項「特定の接続方法を使用する方法」](#)を参照してください。

6.8.3 SSL を使用するための Visibroker Gatekeeper のインストール

Discoverer の SSL サポート機能は、単一のファイアウォールでも複数のファイアウォールでも動作します。

- 単一のファイアウォールで使用する場合、どのポートでも Visibroker Gatekeeper を実行することができます。標準の SSL ポートであるポート 443 で Visibroker Gatekeeper を実行する場合、SSL 通信を行う際に、ファイアウォールを開く（ファイアウォールに「穴を開ける」と呼ばれる）処理を行う必要がありません。443 以外のポートで

Visibroker Gatekeeper を実行する場合は、SSL の通信を行うためにファイアウォールを開ける必要があります。

- 複数のファイアウォールで使用する場合、ポート 443 で Visibroker Gatekeeper を実行します。

注意：Visibroker Gatekeeper を規則に基づく標準ポートであるポート 443 で実行している場合は、SSL のプロキシとの通信も、ポート 443 でのみ許可されます。

Discoverer で SSL を有効にするには、SSL を使用するように Visibroker Gatekeeper を構成する必要があります。Visibroker Gatekeeper を構成するには、次の 2 つのオプションがあります。

1. HTTP サーバーでポート 443 以外のポートで SSL の通信を行っている場合、HTTP サーバー・マシンに Visibroker Gatekeeper をインストールして実行できます。次に、Visibroker Gatekeeper の Exterior Port を 443 に設定します。

詳細は、[7.10.2 項「HTTP サーバーでの Visibroker Gatekeeper の実行」](#) および [7.10.5 項「Visibroker Gatekeeper ポートの変更」](#) を参照してください。

2. HTTP サーバーでポート 443 を使用して SSL の通信を行っている場合、別のマシンに Visibroker Gatekeeper をインストールする必要があります。次に、Visibroker Gatekeeper の Exterior Port を 443 に設定します。

詳細は、[7.10.3 項「別のサーバーでの Visibroker Gatekeeper の実行」](#) および [7.10.5 項「Visibroker Gatekeeper ポートの変更」](#) を参照してください。

6.8.4 SSL の証明書の取得

注意：企業向けに SSL の証明書をすでに取得している場合は、この手順を省略してもかまいません。

Discoverer Plus には、証明書の要求処理を自動化する vbcertrq ツールが用意されています。必要な情報を指定すると、vbcertrq ツールにより次に記載する 3 つのファイルが作成されます。

- 証明書要求ファイル
- 証明書要求の値が含まれたファイル
- 秘密鍵が含まれたファイル

これらのファイルを作成した後、認証局（CA）に証明書の要求を送信してください。CA への証明書の要求送信手順は、送信先の認証局により異なります。社内の CA を使用する場合は、システム管理者に手順を問い合せてください。商用 CA を使用する場合は、目的の機関に証明書の要求送信手順をお問い合せください。

vbcertreq ツールの実行方法

1. コマンド・プロンプトで `vbcertreq` と入力するか、Windows のエクスプローラから `vbcertreq` ツールのアイコンをダブルクリックします。

「Certificate Request Tool」ダイアログが表示されます。

図 6-2 「Certificate Request Tool」ダイアログ

The screenshot shows the 'Certificate Request Tool' dialog box. It has a title bar with the text 'Certificate Request Tool' and standard window controls. Below the title bar is a menu bar with 'File' and 'Help'. The main area is divided into several sections. The first section is 'Certificate Information' and contains the following fields: 'Common-Name:' with a text box, 'Organization:' with a text box, 'Country:' with a dropdown menu showing 'UNITED STATES' and a small '...' button, 'Email:' with a text box, 'Phone:' with a text box, 'Organization Unit:' with a text box, 'Locality/City:' with a text box, and 'State/Province' with a text box. The second section is 'Password' and contains two text boxes: 'Type Password:' and 'Confirm Password:'. The third section is 'Output File Name:' with a text box and a 'Browse' button. The fourth section is 'Key Size:' with a text box. At the bottom of the dialog is a 'Generate Certificate' button.

2. 証明書情報を指定します。詳細は、[6.8.4.1 項「証明書情報の指定」](#)を参照してください。
3. パスワードを指定します。詳細は、[6.8.4.2 項「秘密鍵のパスワードの指定」](#)を参照してください。
4. 鍵のサイズを指定します。詳細は、[6.8.4.4 項「鍵のサイズの指定」](#)を参照してください。
5. 出力ファイルの名前を指定します。詳細は、[6.8.4.3 項「出力ファイル名の指定」](#)を参照してください。

6. 「Generate Certificate Request」ボタンをクリックして証明書要求ファイル、証明書要求値ファイルおよび秘密鍵ファイルを作成します。[SSL の証明書の取得](#)を参照してください。

6.8.4.1 証明書情報の指定

証明書情報を使用して、一意の名前が作成されます。必要な情報は `vbcertrq` ツールのダイアログに赤で表示されます。次の情報を指定します。

- 「Common-Name」(必須)：個人名またはホスト名のいずれか。証明書がユーザー向けであるかサーバー向けであるかにより異なります。`Gatekeeper` で証明書を使用する場合、通常、`Gatekeeper` を実行しているマシンのホスト名を指定します。
- 「Organization」(必須)：証明書の受信者が属する組織または企業。
- 「Country」(必須)：証明書の受信者が居住する国の名前。
- 「Email」(オプション)：証明書の受信者またはそのユーザーの詳細の問合せ先の電子メール・アドレス
- 「Phone」(オプション)：ユーザーの電話番号。
- 「Organization Unit」(オプション)：ユーザーの部門名。
- 「Locality/City」(オプション)：ユーザーが居住する市区町村。
- 「State/Province」(オプション)：ユーザーが居住する州の名前。州の名前をフルネームで入力してください。省略形は使用しないでください。たとえば、「CA」ではなく「California」と入力してください。

6.8.4.2 秘密鍵のパスワードの指定

秘密鍵の暗号化に使用するパスワードを指定する必要があります。このパスワードを指定しなければ、秘密鍵を使用することはできません。パスワードを指定することにより、秘密鍵の不当な使用を防ぐことができます。

パスワードは大文字と小文字を区別します。また、キーボードから入力できる文字はすべて使用できます。パスワードは 6 ～ 99 文字の範囲で指定します。他のパスワードと同様、破られやすい一般的な言葉の使用は避けてください。一般的な語句よりも、数字や括弧などの文字を含めた方が安全です。

秘密鍵のパスワードを指定するには、次のフィールドに入力してください。

- 「Type Password」：秘密鍵を暗号化するパスワード。英数字を組み合わせで指定します。
- 「Confirm Password」：確認のために、もう一度パスワードを入力します。

6.8.4.3 出力ファイル名の指定

ツールにより作成される 3 つのファイルには、「Output File Name」フィールドで指定した名前が付けられます。たとえば、ファイルで mycertreq と指定すると、次のファイルが作成されます。

- mycertreq には証明書の要求が含まれています。
- mycertreq##### には、証明書要求のフィールド値が含まれます。##### は一意の連続番号です。
- mycertreqKEY には、PKCS #8 形式で暗号化された秘密鍵のパスワードが含まれます。

6.8.4.4 鍵のサイズの指定

「Key Size」フィールドで含める鍵のビット数を指定することができます。秘密鍵のビット数を大きく指定するほど、暗号化データのセキュリティが高くなります。サイズは、360 ～ 2,048 ビットの範囲で指定します。

注意：米国連邦輸出規定では、輸出ソフトウェアの鍵のサイズが 512 ビットに制限されています。

6.8.4.5 S/MIME 証明書チェーンの使用

認証局は、複数の形式で作成された証明書チェーンを付与することができます。Visibroker Gatekeeper では、分割された X.509 の証明書が最も一般的であるとされています。S/MIME (PKCS #7) 形式の証明書チェーンを持っている場合、その証明書を分割しなければ Visibroker Gatekeeper で使用できません。

S/MIME 証明書を分割するには

1. <ORACLE_806_HOME>%vbroker%ssleay ディレクトリに移動し、コマンド・プロンプトで次のように入力し、証明書チェーンを複数の証明書に分割します。

```
> ssleay pkcs7 -print_certs -inform DER -in cert-chain-file -out cert-file
```

出力ファイルには、1 つまたは複数の証明書と秘密鍵のリストが表示されます。各証明書は、次の文で区切られます。

```
-----BEGIN CERTIFICATE-----
```

```
-----END CERTIFICATE-----
```

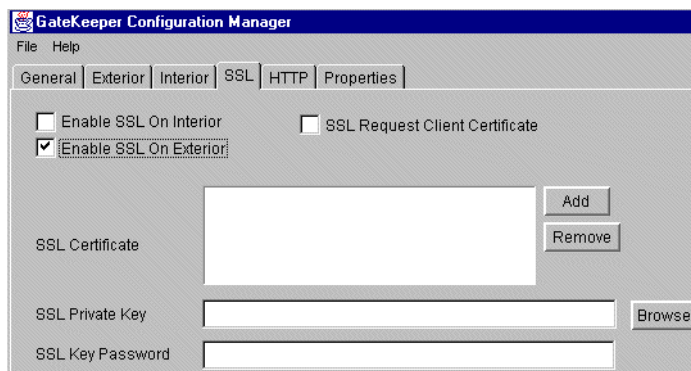
証明書が複数ある場合、最初の証明書が自分のもので、後続の証明書は Thawte などのルートまでの認証局を示します。

6.8.5 SSL の証明書のインストール

必要な証明書を取得した後、証明書からルート認証局までのすべてを Visibroker Gatekeeper にインストールします。

1. <ORACLE_HOME>%vbroker%bin から **gkconfig.exe** を実行します。
2. 「File」 → 「Open」 を選択してファイル
<iSUITES_HOME>%apache%apache%htdocs%Discwb4%applet%gatekeeper.properties を開きます。
3. 「SSL」 タブをクリックして SSL の構成の詳細を表示します。次の図 6-3 を参照してください。

図 6-3 「Gatekeeper Configuration Manager」の「SSL」タブ



4. 「Enable SSL on Exterior」チェックボックスをオンにします。
「SSL Certificate」、「SSL Private Key」および「SSL Key Password」プロパティがアクティブになり、Visibroker Gatekeeper の構成モデルの SSL アクティブ・インジケータ（「マフラー」とも呼ばれる）に SSL が有効であることが表示されます。詳細は、次の図 6-4 を参照してください。

図 6-4 外部で SSL が有効になった Gatekeeper Configuration Manager

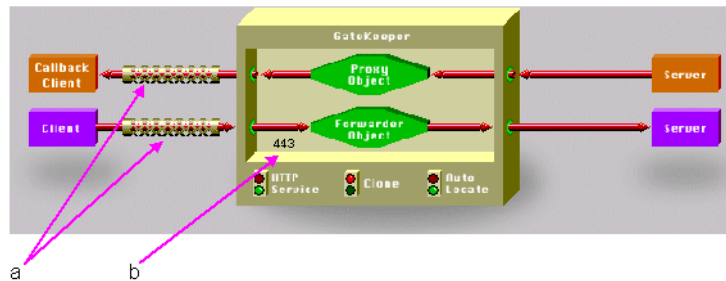


図 6-4 の解説

- a - 外部接続の SSL アクティブ・インジケータが SSL が有効であることを示しています。
- b - 「Exterior Port」（Gatekeeper Configuration Manager の「Exterior」タブで設定）が SSL の標準ポートである 443 に設定されていることに注意してください。

5. 「Add」をクリックします。

「Open」ダイアログ・ボックスが表示されます。

6. SSL の証明書を含むファイルに移動します。

7. SSL の証明書ファイルを選択し、「Open」をクリックします。

ファイルのフルパス名が「SSL Certificate」ボックスに表示されます。

8. 必要に応じて手順 5～7 を繰り返し、証明書チェーンを構築します。

任意の数の証明書ファイルを指定できます。証明書は 1 ファイルにつき 1 つです。ただし、証明書ファイルのリストを指定できます。リストを指定した場合、証明書チェーンとして機能します。証明書チェーンを起動するには、Visibroker Gatekeeper の証明書および発行者の証明書を使用し、さらに必要であればルートまで使用します。

9. 「SSL Private Key」フィールドの横に表示された「Browse」ボタンをクリックし、SSL の秘密鍵を指定します。秘密鍵は Visibroker Gatekeeper の SSL アダプタを正しく操作するために必要です。

「Open」ダイアログ・ボックスが表示されます。

10. 「Gatekeeper Configuration Manager」の「SSL」タブで、「Browse」をクリックします。

「Open」ダイアログ・ボックスが表示されます。

11. SSL の秘密鍵を含むファイルに移動します。

12. ファイルを選択し、「Open」をクリックします。

外部 SSL アダプタの秘密鍵を設定すると、ファイルのフルパス名が「SSL Private Key」ボックスに表示されます。

13. 「SSL Key Password」ボックスに SSL の鍵のパスワードを入力します。

SSL の秘密鍵が暗号化されている場合は、その鍵の外部および内部 SSL アダプタ向けパスワードを入力します。vbcertreq ツールを使用して証明書要求を作成した場合は、そこで指定したパスワードを入力します。

パスワードを入力すると、文字がアスタリスクで表示されます。パスワードは「SSL Key Password」ボックスに表示されません。

注意：SSL の鍵のパスワードはテキスト形式で gatekeeper.properties ファイルに保存されます。したがって、権限を付与されたユーザーのみがこのファイルにアクセスすることができます。

14. 「File」→「Save」を選択します。

15. 「File」→「Exit」を選択します。

6.8.6 x509cert.jar ファイルのインストール

SSL を使用するには、Discoverer のクライアント・マシンに x509cert.jar ファイルがインストールされている必要があります。

x509cert.jar を Windows NT クライアントにインストールするには、次の手順を実行してください。

1. HTTP サーバー・マシンの <i>SUITES_HOME>%apache%apache%htdocs%discwb4%util ディレクトリにある x509cert.jar ファイルをクライアント・マシンの %Program Files%Oracle%JInitiator 1.1.7.30%lib ディレクトリにコピーします。

必要であれば、<http://hostname.domain/discwb4/util/x509cert.jar> からファイルをダウンロードできます。

x509cert.jar を UNIX クライアントにインストールするには、次の手順を実行してください。

1. HTTP サーバー・マシンの <i>SUITES_HOME>%apache%apache%htdocs%discwb4%util ディレクトリにある x509cert.jar ファイルをクライアント・マシンの <Java-Plugin>/lib/ ディレクトリにコピーします。<Java-Plugin> は、Java プラグインがインストールされているディレクトリです。

必要であれば、<http://hostname.domain/discwb4/util/x509cert.jar> からファイルをダウンロードできます。

6.8.7 必要なダイナミック・リンク・ライブラリ（DLL）・ファイルのインストール

SSL 対応の ORB を Windows NT クライアントで使用するには、クライアント・マシンにファイル **vbj30ssl.dll** および **vbrnissl.dll** ファイルがインストールされている必要があります。これらの DLL ファイルをインストールするには、次の手順を実行してください。

1. いずれか一方または両方のファイルを HTTP サーバー・マシンの次のディレクトリからクライアント・マシンの %winnt ディレクトリにコピーします。
 - <iSUITES_HOME>%apache%apache%htdocs%discwb4%util%vbj30ssl.dll
— Jinitiator ユーザー
 - <iSUITES_HOME>%apache%apache%htdocs%discwb4%util%vbrnissl.dll
— Internet Explorer ユーザー

SSL 対応の ORB を UNIX クライアントで使用するには、クライアント・マシンに **libvbj30ssl.so** ファイルがインストールされている必要があります。このファイルをインストールするには、次の手順を実行してください。

1. HTTP サーバー・マシンの <iSUITES_HOME>/Apache/Apache/htdocs/discwb4/util ディレクトリにある libvbj30ssl.so ファイルをクライアント・マシンの <Java-Plugin>/lib/ ディレクトリにコピーします。<Java-Plugin> は、Java プラグインがインストールされているディレクトリです。

必要であれば、<http://hostname.domain/discwb4/util/vbj30ssl.dll> および <http://hostname.domain/discwb4/util/vbrnissl.dll> からそれぞれのファイルをダウンロードできます。

6.8.8 Discoverer Plus の起動ページでの SSL の有効化

Discoverer の起動ページで SSL を有効にするには、次の URL パラメータを起動ページの URL に追加します。

ORBenableSSL=yes

例: <http://server.com/discwb4/html/english/welcome.htm?ORBenableSSL=yes>

注意： 同一の URL に複数の URL パラメータが設定されている場合、アンパサンド（&）で URL パラメータを区切ります。

例：

<http://server.com/discwb4/html/english/welcome.htm?ORBenableSSL=yes&ORBalwaysProxy=yes>

注意： URL パラメータ ORBenableSSL および ORBalwaysTunnel を同時に使用することはできません。

6.8.9 ファイアウォール内での SSL の使用

デフォルトでは、SSL はファイアウォール外部のクライアントに対してのみ動作します。URL パラメータを次のように設定することにより、ファイアウォール内で SSL を有効にすることができます。

- ORBalwaysProxy=yes IIOP Proxying を使用した接続（「[特定の接続方法を使用する方法](#)」を参照してください）
- ORBenableSSL=yes URL 要求時に SSL を有効化

注意：ファイアウォール内で SSL を使用する場合は、どのポートでも使用することができます。

6.9 HTTPS を使用した Discoverer Viewer の構成

Discoverer Viewer 文書が安全であることを確認し、ブラウザに安全なページを認識させるには、次の手順を実行してください。

- HTTP サーバーに SSL の証明書をインストールし、Discoverer Viewer Servlets に対して SSL を有効にします。詳細は、『Oracle9i Application Server インストレーション・ガイド』を参照してください。
- 次のファイルを編集して、HTTPS を設定します。
 - httpd.conf ファイル。詳細は、「[httpd.conf ファイルの編集による HTTPS の設定](#)」を参照してください。
 - disco4ivviewer.properties ファイル。詳細は、「[disco4ivviewer.properties ファイルの編集による HTTPS の設定](#)」を参照してください。
 - disco4iv.xml ファイル。詳細は、「[disco4iv.xml ファイルの編集による HTTPS の設定](#)」を参照してください。
 - viewer_config.xml ファイル。詳細は、「[viewer_config.xml ファイルの編集による HTTPS の設定](#)」を参照してください。

6.9.1 httpd.conf ファイルの編集による HTTPS の設定

httpd.conf ファイルを次のように変更します。

1. httpd.conf ファイルを開いて編集します。

httpd.conf ファイルは、<i>SUITES_HOME>%apache%apache%conf に格納されています。
2. ファイルの「Alias」の項に次に記載する行が存在しない場合は追加し、disco4iv ディレクトリの場所を指定します。

```
Alias /disco4ivfiles/ "<i>SUITES_HOME>%apache%apache%htdocs%disco4iv/"
```

- disco4ivfiles には任意の名前を指定します。
- <iSUITES_HOME> は、Oracle9i Application Server コンポーネントをインストールする場所です。

例

(httpd.conf)

```
# Aliases: Add here as many aliases as you need (with no limit). The format is
# Alias fakename realname
#
# Note that if you include a trailing / on fakename then the server will
# require it to be present in the URL. So "/icons" isn't aliased in this
# example, only "/icons/".
#
Alias /icons/ "D:¥apache¥icons/"
Alias /disco4ivfiles/ "D:¥iAS_Home¥apache¥apache¥htdocs¥disco4iv/"
```

6.9.2 disco4iviewer.properties ファイルの編集による HTTPS の設定

disco4iviewer.properties を次のように変更します。

1. disco4iviewer.properties ファイルを開いて編集します。
disco4iviewer.properties ファイルは、<iSUITES_HOME>¥apache¥jserv¥servlets に格納されています。
2. disco4iv.xml ファイルへのパスを HTTP プロトコルから FILE プロトコルへ変更します。

例

(disco4iviewer.properties - 変更前)

```
servlet.Viewer.initArgs=config=http://mywebserver.company.com/disco4iv/html/
disco4iv.xml
```

```
servlet.viewer.initArgs=config=http://mywebserver.company.com/disco4iv/html/
disco4iv.xml
```

(disco4iviewer.properties - 変更後)

```
servlet.Viewer.initArgs=config=file:///D:¥iAS_Home¥apache¥apache¥htdocs¥disco4iv
¥html¥disco4iv.xml
```

```
servlet.viewer.initArgs=config=file:///D:¥iAS_Home¥apache¥apache¥htdocs¥disco4iv
¥html¥disco4iv.xml
```

6.9.3 disco4iv.xml ファイルの編集による HTTPS の設定

disco4iv.xml ファイルを次のように変更します。

1. disco4iv.xml ファイルを開いて編集します。

disco4iv.xml ファイルは、`<iSUITES_HOME>%apache%apache%htdocs%disco4iv%html` に格納されています。

2. disco4iv.xml ファイルの `xsl_path` 引数を変更して、FILE プロトコルおよびフルパスを含めます。

注意：`xsl_path` 引数を設定する場合に使用されるのは、HTTPS プロトコルではなく FILE プロトコルです。次を参照してください。

例

(disco4iv.xml - 変更前)

```
<argument name="xsl_path" type="href">disco4iv.xsl</argument>
```

(disco4iv.xml - 変更後)

```
<argument name="xsl_path" type="href">file:///D:/iAS_Home%apache%apache%htdocs%disco4iv%html%disco4iv.xsl</argument>
```

6.9.4 viewer_config.xml ファイルの編集による HTTPS の設定

viewer_config.xml ファイルを次のように変更します。

1. viewer_config.xml ファイルを開いて編集します。

viewer_config.xml ファイルは、`<iSUITES_HOME>%apache%apache%htdocs%disco4iv%html` に格納されています。

2. 構成要素の `base_dir` 属性の値を、disco4iv.xml ファイルへの完全に安全なパスに変更します。

例

(viewer_config.xml - 変更前)

```
<configuration application="discoverer_viewer"
base_dir="." image_dir="images">
```

(viewer_config.xml - 変更後)

```
<configuration application="discoverer_viewer"
base_dir="https://mywebserver.company.com/disco4ivfiles/html/"
image_dir="images">
```

6.9.5 Discoverer Viewer での SSL の有効化

Discoverer Viewer の URL で SSL を有効化するには、URL の接頭辞に **http** ではなく **https** を使用します。

例

`https://company.com/Discoverer4i/Viewer/<Viewer parameters>`

Discoverer Plus for Firewall Support の構成

この章では、Visibroker Gatekeeper バージョン 3.4 を使用して、企業のファイアウォールと連携して動作するように Oracle Discoverer Plus を構成する方法を説明します。

注意：この章は、Discoverer Viewer には適用されません。Discoverer Viewer はファイアウォールに準拠した HTTP および HTTPS プロトコルを使用するため、Visibroker Gatekeeper は不要です。

説明するトピックは次のとおりです。

- [用語](#)
- [詳細情報](#)
- [はじめに](#)
- [インターネット・ファイアウォールについて](#)
- [Visibroker Gatekeeper について](#)
- [Discoverer 用の Gatekeeper 構成の選択](#)
- [IIOP Proxying 構成での Visibroker Gatekeeper の使用](#)
- [HTTP Tunnelling 構成での Visibroker の使用](#)
- [Discoverer Client の構成](#)
- [Visibroker Gatekeeper の構成](#)

7.1 用語

- HTTP サーバー・マシンは、HTTP サーバー・ソフトウェアをインストールしたマシンを指します。このマシンは、一般的に Web サーバーのことを意味します。
- HTTP サーバー・ソフトウェアは、HTTP ソフトウェアそのもの（Oracle HTTP Server など）を指します。
- Discoverer Services は、Locator、Preference および Session コンポーネントを指します。
- プロキシ・サーバーは、ワークステーションとインターネットの中間に介在するマシンを指します。プロキシ・サーバーは、セキュリティ、管理およびキャッシュのサービスを提供します。

7.2 詳細情報

Visibroker Gatekeeper の詳細は、次に記載する Borland Software Corporation のインターネット・サイトにアクセスしてください。

[http://www.borland.com/techpubs/books/security/bss45\(iss41\)/framesetindex.html](http://www.borland.com/techpubs/books/security/bss45(iss41)/framesetindex.html)

7.3 はじめに

会社のファイアウォールを経由して Discoverer Plus を配置する方法を決める前に、ネットワーク管理者に確認して、会社のファイアウォール・ポリシーを認識しておく必要があります。インターネットを介して自分の Discoverer のデータにアクセスしてくる組織のファイアウォール・ポリシーも考慮するようにしてください。

ファイアウォール・ポリシーをネットワーク管理者と検討するときには、次の基本的な概念を理解しておく必要があります。

- ファイアウォールおよび保護 TCP/IP ポート - ファイアウォール、Demilitarized Zone (DMZ) および保護 TCP/IP ポートについては、[7.4 項「インターネット・ファイアウォールについて」](#)を参照してください。
- Visibroker Gatekeeper - Visibroker Gatekeeper とその使用目的については、[7.5 項「Visibroker Gatekeeper について」](#)を参照してください。
- ファイアウォール・トポロジー - ネットワークの物理的な編成については、[7.6.1 項「ファイアウォールの構成について」](#) および [7.6.2 項「DMZ 構成の例」](#)を参照してください。
- Visibroker Gatekeeper の構成 - 自社のファイアウォール・ポリシーに合致するネットワーク・プロトコルを判別するには、[7.6.3 項「Gatekeeper 構成の選択」](#)を参照してください。

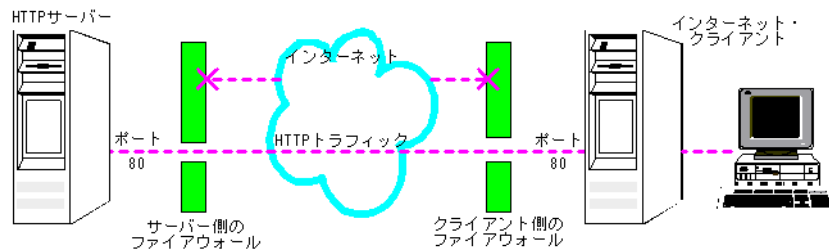
7.4 インターネット・ファイアウォールについて

7.4.1 インターネット・ファイアウォール

インターネット・ファイアウォールとは、インターネットと企業のネットワークの間のセキュリティ・ポリシーを確立するためのシステムまたは複数のシステムの集まりを指します。

つまり、インターネット・ファイアウォールはネットワークを囲む電子的な壁であり、許可されていないアクセスからネットワークを保護します。

図 7-1 クライアント側およびサーバー側のファイアウォールとの中の代表的なインターネット接続



インターネット経由で通信する Web サーバー・マシンの場合、その HTTP サーバーとインターネットの間にファイアウォールを設置するのが一般的です。これは、サーバー側のファイアウォールと呼ばれます。この Web サーバー・マシンに接続する他の企業（または遠隔地にある部門）は、クライアント側のファイアウォールと呼ばれる専用のファイアウォールを設置するのが一般的です。企業のファイアウォール・ポリシーに準拠した情報はファイアウォールを通過することが許されるため、サーバー・マシンとクライアント・マシンとの間の通信が可能です。

7.4.2 Demilitarized Zone (DMZ) とは

Demilitarized Zone (DMZ) とは、セキュリティのレベルを追加するファイアウォール構成のことです。この構成では、保護ネットワークとインターネットの間にあるエクストラネットが DMZ です。DMZ 内部のリソースはパブリック・インターネットから参照可能ですが、保護されています。DMZ には、企業のパブリック Web サイト、ファイル転送プロトコル (FTP) サイトおよびシンプル・メール転送プロトコル (SMTP) のホストとなるサーバーが存在するのが一般的です。

DMZ の詳細は、[7.6.1 項「ファイアウォールの構成について」](#)を参照してください。

7.4.3 保護ポート

インターネット・サーバーとクライアントは、保護ポートと呼ばれる伝送制御プロトコル / インターネット・プロトコル (TCP/IP) ポートを使用して相互接続されます。インターネット・サーバー・マシンは、様々なサービス (HTTP や FTP など) を様々な TCP/IP ポートで実行します (下図参照)。

表 7-1 一般的に使用される TCP/IP ポートとそのサービス	
ポート	サービス
21	FTP
23	Telnet
79	Finger
80	HTTP
443	SSL

1 つのポートで動作するサービスは 1 つのみです。たとえば、ポート 80 は HTTP トラフィック専用で、ポート 443 は SSL トラフィック専用です。

ファイアウォールは、(HTTP 専用のポート 80 のように) 通信量を制限して TCP/IP ポートを保護することによって、ネットワークを保護します。

次に示すように、ファイアウォールは、ポート 80 およびポート 443 を除くすべてのポートに対する通信をブロックするのが一般的です。

- ポート 80 は HTTP トラフィック専用です。
- ポート 443 は SSL トラフィック専用です。

他のタイプのトラフィックは、いずれのポートでもブロックされます。そのため、ポート 80 で FTP を使用してインターネット・サーバーにアクセスし、ファイルを取り出すことはできません。FTP による通信は、ポート 80 ではブロックされるためです。

7.5 Visibroker Gatekeeper について

この項では、Visibroker Gatekeeper の概要と、Visibroker Gatekeeper を Discoverer で使用する目的を説明します。

7.5.1 Visibroker Gatekeeper

Visibroker Gatekeeper は、Inprise Corporation が開発したサード・パーティー製ソフトウェアです。このソフトウェアにより、Visibroker CORBA のクライアントおよびサーバーが、インターネット・ブラウザおよびファイアウォールによって規定される制限事項に準拠しながらネットワークを介して通信できます。

7.5.2 Discoverer Plus に Visibroker Gatekeeper が必要な理由

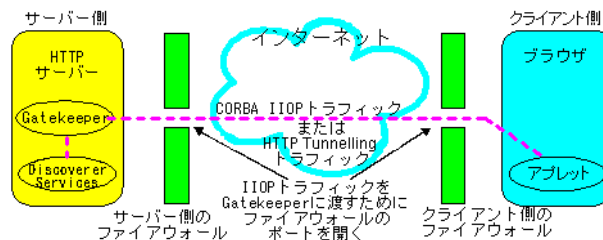
Discoverer Plus は、クライアントとサーバーの間の通信に CORBA (Common Object Request Broker Architecture) を使用します。CORBA は、様々なプログラム（オブジェクトと呼ばれる）が、互いのプログラミング言語や実行されているオペレーティング・システムを意識することなく通信できるようにするためのアーキテクチャです。

CORBA は、IIOP (Internet Inter-ORB Protocol) プロトコルを使用したインターネット経由で実装されます。IIOP プロトコルを使用すると、サーバーとクライアントの間で、整数や配列の他に複合オブジェクトも交換することができます。一方、HTTP (ハイパー・テキスト転送プロトコル) プロトコルは、テキストの送信のみをサポートするインターネットで使用される、基礎的なプロトコルです。

セキュリティ上の理由から、IIOP トラフィックが直接通過することを許可していないパブリック・ファイアウォールもあります。

Visibroker Gatekeeper を使用すると、CORBA ベースの IIOP トラフィックがファイアウォールを通過できます。

図 7-2 HTTP サーバーにインストールされた Gatekeeper に CORBA IIOP トラフィックをパスルするためのポートをオープンしたファイアウォール



Visibroker Gatekeeper は、HTTP サーバー・マシン上のサーバー側ファイアウォールの内側で実行するのが一般的です。Visibroker Gatekeeper は、次のことを行います。

- クライアントからのすべての IIOP 要求または HTTP 要求をリスニング（またはモニター）します。
- 適切な要求を Discoverer Services に転送します。
- Discoverer Services からのリプライをリスニングし、それらをクライアントに戻します。

したがって、Visibroker Gatekeeper は、プロキシ・サーバー・マシン（ワークステーションとインターネットの間に介在するマシン）のように動作することになります。

7.5.3 必要な Visibroker Gatekeeper のバージョン

Oracle Discoverer Plus は、Visibroker Gatekeeper バージョン 3.4 を使用します。Visibroker Gatekeeper は、Discoverer Plus の一部としてインストールされます。

7.6 Discoverer 用の Gatekeeper 構成の選択

この項は、インターネットなどのパブリック・ネットワークを介して Discoverer Plus を配置するために必要となる、代表的なファイアウォールの構成と Visibroker Gatekeeper の構成に関するアドバイスを記載します。

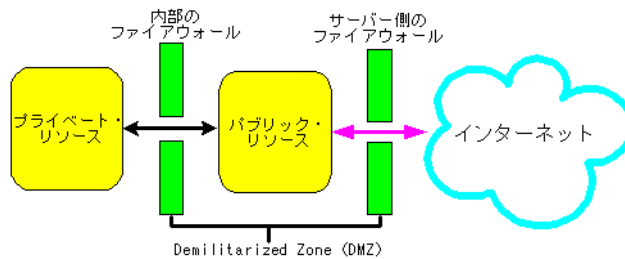
7.6.1 ファイアウォールの構成について

ファイアウォールのポリシーは組織によって異なり、入手が容易な既成のファイアウォール・パッケージも様々です。すべてのファイアウォールの使用例を示すことは不可能です。このマニュアルでは、最も一般的なファイアウォールの構成を取り上げます。

代表的な DMZ 構成は、次の図に示す 2 つのファイアウォールに準拠しています。

- インターネットとパブリック・リソースの間のサーバー側ファイアウォール
- パブリック・リソースとプライベート・リソースの間の内部ファイアウォール

図 7-3 Demilitarized Zone (DMZ)



理想的なファイアウォールの構成は、DMZ 内部のリソースが漏れると想定し、その場合でも、内部ネットワークとそのネットワーク上の機密データの破損を最小限にとどめられることです。これには、次の 2 つの手順があります。

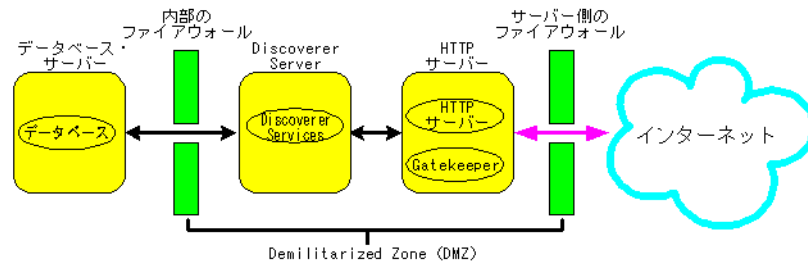
- 機密のプライベート・リソース（少なくとも、データベースおよびアプリケーション・ロジック）を、DMZ からインターネット・ファイアウォールの内側にある内部ネットワークに移動します。
- 機密のプライベート・リソースに対する内部ネットワークからのアクセスを制限し、同時に DMZ からのアクセスも制限します。

7.6.2 DMZ 構成の例

Discoverer Server を配置する際には、次の図に示す 2 つの代表的なセキュリティ構成があります。

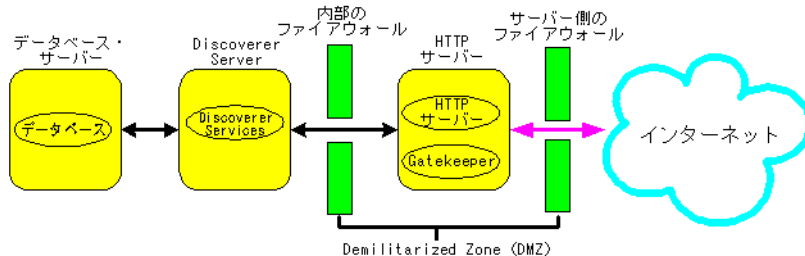
1. Discoverer Server および HTTP サーバーを DMZ 内に配置する方法。データベースを内部のファイアウォールの内側に配置します（下図参照）。この方法はセキュリティのレベルが高いため、Discoverer Plus にお勧めします。

図 7-4 DMZ 内に配置した Discoverer Server および HTTP



2. HTTP サーバーを DMZ 内に配置する方法。Discoverer Server およびデータベースを、インターネット・ファイアウォールの内側に配置します（下図参照）。この方法はセキュリティのレベルがより高くなりますが、実装も難しくなります。

図 7-5 内部のファイアウォールの内側に配置した Discoverer Server



7.6.3 Gatekeeper 構成の選択

Visibroker Gatekeeper は通常、クライアントの IIOP または HTTP プロトコルを処理し、Discoverer Services に転送します。つまり、Visibroker Gatekeeper では、IIOP Proxying 構成または HTTP Tunnelling 構成を使用します。どちらの構成を選択するかは、ファイアウォール・ポリシーで定められた制限により異なります。

7.6.3.1 IIOP Proxying の構成

- この構成では、サーバー側のファイアウォールを開いて TCP/IP トラフィックを特定のポートに渡します。また、クライアント側のファイアウォールは、クライアントからファイアウォール外部に TCP/IP トラフィックを渡せる必要があります。
- この構成は、Discoverer Plus のパフォーマンスを最大限に活用できるため、お勧めします。
- ただし、クライアントのファイアウォールは頻繁に制御されないため、IIOP 構成は HTTP Tunnelling 構成に比べて実装が複雑です。

詳細は、7.7 項「IIOP Proxying 構成での Visibroker Gatekeeper の使用」を参照してください。

7.6.3.2 HTTP Tunnelling の構成

- この構成では、IIOP トラフィックを HTTP パケットに埋め込んでファイアウォールに渡します。
- この方法は、TCP/IP トラフィックを通すためにサーバー側またはクライアント側のファイアウォールを開く必要がないという利点があります。
- つまり、HTTP Tunnelling 構成は IIOP Proxying 構成に比べて安全性が高いと言えます。

- ただし、この構成は間接的に著しくパフォーマンスを低下させるため、IIOProxying 構成ほど効率的ではありません。

詳細は、7.8 項「[HTTP Tunnelling 構成での Visibroker の使用](#)」を参照してください。

7.7 IIOProxying 構成での Visibroker Gatekeeper の使用

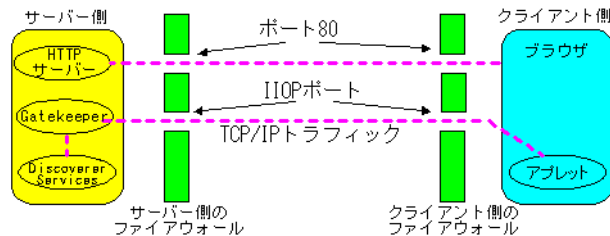
この項では、IIOProxying 構成を使用した Discoverer Plus の構成方法について説明します。

7.7.1 概要

IIOProxying 構成では、サーバー側およびクライアント側のファイアウォールを開いて TCP/IP トラフィックを通します。

Discoverer Plus を起動すると、HTTP サーバーがポート 80 を経由したブラウザに Discoverer アプレットを送信します。Discoverer アプレットが初期化されると、指定された IIOProxying のポートを使用して Gatekeeper と通信します。

図 7-6 IIOProxying 構成を使用した Discoverer



Visibroker Gatekeeper は、HTTP サーバー・マシンのサーバー側のファイアウォールの内側で実行され、クライアントと Discoverer Services の通信をサポートします。この操作を行うために、Gatekeeper はクライアントからの要求をすべて解釈し、クライアントのかわりに Discoverer Services へ送信します。Discoverer Services からクライアントへの応答も、Gatekeeper を経由して渡されます。

IIOProxying 構成を異なる 2 つのセキュリティ・レベルで使用するよう Gatekeeper を構成することができます。

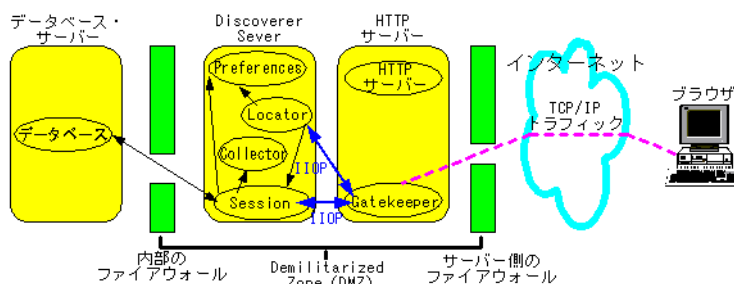
- [ファイアウォールを Discoverer Services と分離させない Gatekeeper の構成](#) - この構成では、Discoverer Server を DMZ に配置します。
- [内部のファイアウォールの内側に配置した Discoverer Server を使用した Gatekeeper の構成](#) - この構成では、Discoverer Server が DMZ に配置されません。

7.7.2 ファイアウォールを Discoverer Services と分離させない Gatekeeper の構成

この構成では、Discoverer Server および HTTP サーバーを DMZ に配置します。

IIOProxying パケットは Discoverer Services に制限されず、どの場所へも送信されるようになります。Discoverer Services および Locator は同じサブネットに配置されます。次を参照してください。

図 7-7 Discoverer を内部のファイアウォールの内側に配置した IIOProxying 構成



Visibroker Gatekeeper の構成方法は、次のとおりです。

- Discoverer Plus のインストール時には、HTTP サーバーの次の場所に Visibroker Gatekeeper がインストールされます。

```
<ORACLE_806_HOME>\vbroker\bin\gatekeeper.exe
```

- Visibroker Gatekeeper が、Discoverer Plus NT サービスによって自動的に起動されます。
- Visibroker Gatekeeper は、デフォルトではポート 15000 をリスニングします。デフォルトのポート番号を変更する方法は、7.10.5 項「Visibroker Gatekeeper ポートの変更」を参照してください。
- Visibroker Gatekeeper を特定のポートで実行するように設定した場合は、TCP/IP トラフィックが Visibroker Gatekeeper のホストおよびポートに到達できるようサーバーのファイアウォールを設定する必要があります。

クライアント側のファイアウォールを制御しない場合は、サーバー側のファイアウォールを構成する解決策が 3 通りあります。

- クライアントのファイアウォール管理者が、クライアントのファイアウォールを TCP/IP トラフィックがすべてのポートに伝送されるように設定していれば、IIOProxying 構成は正しく動作します。

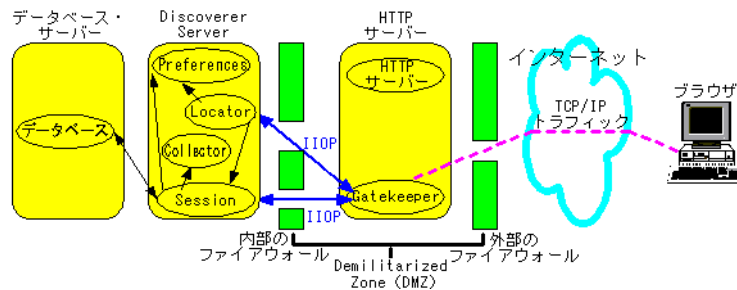
- Real Audio などのために TCP/IP 用に一定のポートを設定している場合は、開かれているポートを使用して Gatekeeper を実行することができます。
- クライアント側のファイアウォールの内側以外にクライアントが存在する場合は、IIOProxying 構成はそのクライアントを対象に動作します。

注意：ファイアウォールの構成は、サイトで使用しているファイアウォールに特有のものです。詳細は、ファイアウォールの管理者にお問い合わせください。

7.7.3 内部のファイアウォールの内側に配置した Discoverer Server を使用した Gatekeeper の構成

この構成では、HTTP サーバーを DMZ に配置します。Discoverer Server およびデータベースは、ファイアウォールの内側に配置します（下図参照）。

図 7-8 Discoverer を内部のファイアウォールの内側に配置した IIOProxying 構成



Demilitarized Zone での Visibroker Gatekeeper の構成方法は、次のとおりです。

- 7.7.2 項「ファイアウォールを Discoverer Services と分離させない Gatekeeper の構成」の手順に従ってください。
- HTTP サーバーと Discoverer Server の間にあるファイアウォールを開き、すべての TCP/IP トラフィックが全ポートの Discoverer Server マシンに到達するようにします。次に、アクセスを制限するポートを選択してブロックします。たとえば、HTTP ポートや Telnet ポートなどです。
- HTTP サーバー・マシンでは、Locator が実行されないようにしてください。このためには、「レジストリ」からレジストリ・キー `¥HKEY_LOCAL_MACHINE¥SOFTWARE¥ORACLE¥DCW4_START_LOCATOR` を 0 に設定します。NT サービスの OracleDiscoverer4i をいったん停止して再起動してください。
- Discoverer Server の 1 つで、Locator を実行します。これを行うためには、レジストリ・キー `¥HKEY_LOCAL_MACHINE¥SOFTWARE¥ORACLE¥DCW4_START_LOCATOR` を 1 に設定します。NT サービスの OracleDiscoverer4i をいったん停止して再起動してください。

- Locator および Discoverer Session および Preferences が同一のサブネット上にあることを確認します。
- Locator を起動するたびに、Locator が生成する locator.ior ファイルが HTTP サーバー・マシンの `<i>SUITES_HOME>%apache%apache%htdocs%discwb4%applet` ディレクトリにコピーされるようにしてください。

7.8 HTTP Tunnelling 構成での Visibroker の使用

この項では、HTTP Tunnelling 構成を使用した Discoverer の構成方法について説明します。

7.8.1 概要

クライアントが HTTP トラフィックのみを許可するファイアウォールの内側で実行されている場合、サーバーに送信される IIOP データが HTTP パケットに埋め込まれます。これにより、ファイアウォールをデータが通過できるようになります。

HTTP Tunnelling 構成では、Visibroker Gatekeeper で HTTP パケットから IIOP 要求が抽出され、Discoverer Server コンポーネントに渡されます。応答が HTTP 応答パケットとしてクライアントに戻されると、クライアントは IIOP パケットを抽出します。

図 7-9 HTTP Tunnelling 構成での Discoverer の使用



HTTP Tunnelling 構成を使用する最大の利点は、TCP/IP トラフィックを通すためにファイアウォールを開ける必要がないことです。ただし、Gatekeeper が実行されているホストおよびポートに HTTP トラフィックを送信できるかどうかファイアウォールの設定を確認する必要があります。詳細は、7.10.1 項「Visibroker Gatekeeper の実行場所」を参照してください。

注意： 複数のファイアウォールを使用することで、ファイアウォールを開かずに Discoverer を実行することもできます。この方法では、Visibroker Gatekeeper がポート 443 で実行されます。詳細は、6.8 項「Discoverer Plus で SSL を使用するための構成」を参照してください。

ポート 80 は HTTP サーバーのデフォルトのポートであるため、HTTP サーバーで Visibroker Gatekeeper を実行する場合は、Visibroker Gatekeeper は別のポート番号を使用する必要があります。Visibroker Gatekeeper でデフォルトで使用されるポート番号は 15000 です。HTTP サーバーを経由して別のマシンで Visibroker Gatekeeper を実行することもできます。詳細は、7.10.3 項「別のサーバーでの Visibroker Gatekeeper の実行」を参照してください。

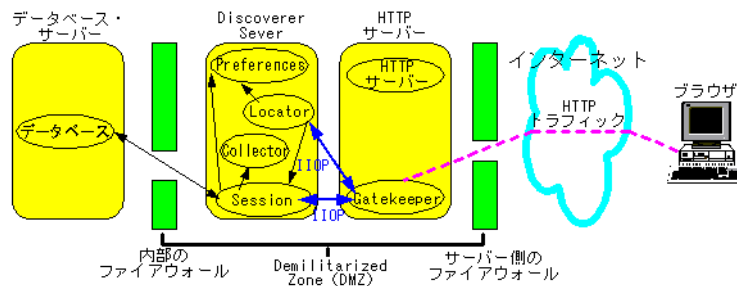
HTTP Tunnelling 構成を異なる 2 つのセキュリティ・レベルで使用するよう Gatekeeper を構成することができます。

- ファイアウォールを Discoverer Services と分離させない Gatekeeper の構成 - この構成では、Discoverer Server を DMZ に配置します。
- 内部のファイアウォールの内側に配置した Discoverer Server を使用した Gatekeeper の構成 - この構成では、Discoverer Server を DMZ に配置します。

7.8.2 ファイアウォールを Discoverer Services と分離させない Gatekeeper の構成

この構成では、Discoverer Server および HTTP サーバーを DMZ に配置します。次を参照してください。

図 7-10 Discoverer を内部のファイアウォールの内側に配置した HTTP Tunnelling 構成



Visibroker Gatekeeper の構成方法は、次のとおりです。

- Discoverer Plus のインストール時には、HTTP サーバーの次の場所に Visibroker Gatekeeper がインストールされます。

<ORACLE_806_HOME>%vbroker%bin%gatekeeper.exe

- Visibroker Gatekeeper が、Discoverer Plus NT サービスによって自動的に起動されます。

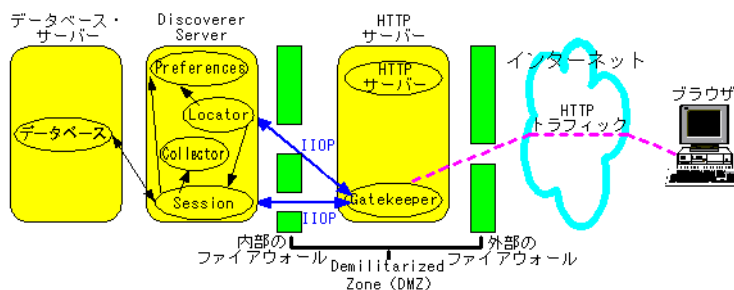
- Visibroker Gatekeeper は、デフォルトではポート 15000 をリスニングします。デフォルトのポート番号を変更する方法は、7.10.5 項「Visibroker Gatekeeper ポートの変更」を参照してください。
- Visibroker Gatekeeper を特定のポートで実行するように設定した場合は、HTTP トラフィックが Visibroker Gatekeeper のホストおよびポートに到達できるようファイアウォールを設定する必要があります。

注意：ファイアウォールの構成は、サイトで使用しているファイアウォールに特有のものです。詳細は、ファイアウォールの管理者にお問い合わせください。

7.8.3 内部のファイアウォールの内側に配置した Discoverer Server を使用した Gatekeeper の構成

この構成では、HTTP サーバーを DMZ に配置します。Discoverer Server およびデータベースは、ファイアウォールの内側に配置します（下図参照）。

図 7-11 Discoverer を内部のファイアウォールの内側に配置した HTTP Tunnelling 構成



Visibroker Gatekeeper の構成方法は、次のとおりです。

- 7.8.2 項「ファイアウォールを Discoverer Services と分離させない Gatekeeper の構成」の手順に従ってください。
- HTTP サーバーと Discoverer Server の間にあるファイアウォールを開き、すべての TCP/IP トラフィックが全ポートの Discoverer Server マシンに到達するようにします。次に、アクセスを制限するポートを選択してブロックします。たとえば、HTTP ポートや Telnet ポートなどです。
- HTTP サーバー・マシンでは、Locator が実行されないようにしてください。このためには、「レジストリ」からレジストリ・キー `¥HKEY_LOCAL_MACHINE¥SOFTWARE¥ORACLE¥DCW4_START_LOCATOR` を 0 に設定します。NT サービスの OracleDiscoverer4i をいったん停止して再起動してください。

- Discoverer Server の 1 つで、Locator を実行します。これを行うためには、レジストリ・キー `¥HKEY_LOCAL_MACHINE¥SOFTWARE¥ORACLE¥DCW4_START_LOCATOR` を 1 に設定します。NT サービスの `OracleDiscoverer4i` をいったん停止して再起動してください。
- Locator および Discoverer Session および Preferences が同一のサブネット上にあることを確認します。
- Locator を起動するたびに、Locator が生成する `locator.ior` ファイルが HTTP サーバー・マシンの `<i>SUITES_HOME>¥apache¥apache¥htdocs¥discwb4¥applet` ディレクトリにコピーされるようにしてください。

7.8.4 永続的な HTTP 接続

HTTP は接続しないプロトコルであるため、Discoverer Client を Gatekeeper 経由で通信させる場合は、その都度新しい HTTP 接続を確立する必要があります。接続が確立されると応答を受信し、その後接続が閉じられます。

Discoverer Client および Gatekeeper は、Discoverer Client が終了するまですべての要求に対して確立および使用された接続を介して通信することが可能です。これは、永続接続と呼ばれます。HTTP Tunnelling を使用している場合、Discoverer Plus で永続接続を使用することをお勧めします。

永続接続を使用するには、クライアントと Gatekeeper の間のファイアウォールで実行されている HTTP プロキシが永続接続に対応している必要があります。

注意：ファイアウォールの構成は、サイトで使用しているファイアウォールに特有のものです。詳細は、ファイアウォールの管理者にお問い合わせください。

7.9 Discoverer Client の構成

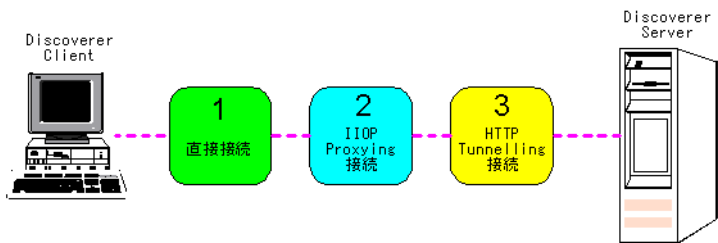
この項では、Visibroker Gatekeeper を使用するための Discoverer Client の構成方法について説明します。

7.9.1 Discoverer Client の接続順序

デフォルトでは、Discoverer Client のセッションは次の 3 ステップの順序で Discoverer Server への接続を試みます。

1. Discoverer Client が Locator への直接接続を試みます（Gatekeeper 非使用）。
2. Discoverer Client から直接接続できない場合、IIOP Proxying を使用して Discoverer Server への接続を試みます。
3. IIOP Proxying が設定されていない場合、HTTP Tunnelling を使用して Discoverer Server との通信を試みます。

図 7-12 Discoverer Client におけるデフォルトの接続順序の 3 ステップ



接続順序を変更して Discoverer のパフォーマンスを改善することができます。詳細は、[7.9.2 項「特定の接続方法を使用する方法」](#)を参照してください。

7.9.2 特定の接続方法を使用する方法

次に記載する URL パラメータのいずれか 1 つを使用して、Discoverer Client の接続方法を指定することができます。

- ORBalwaysProxy - IIOP Proxying を使用した接続（次の表を参照）
- ORBalwaysTunnel - HTTP Tunneling を使用した接続（次の表を参照）

注意：1 つの接続文字列で両方の URL パラメータを使用しないでください。

表 7-2 Discoverer URL パラメータによる接続方法の指定

引数と値	目的	例
ORBalwaysProxy=yes	このパラメータを yes に設定すると、直接接続を省略し、IIOP Proxying を介して接続しようとします。成功しなかった場合、HTTP Tunneling を使用した接続を試みます。	http://hostname.domain/discwb4/html/japanese/ms_ie/start_ie.htm?ORBalwaysProxy=yes
ORBalwaysTunnel=yes	このパラメータを yes に設定すると、直接接続および IIOP Proxying を使用した接続を省略し、HTTP Tunneling を介して接続しようとします。	http://hostname.domain/discwb4/html/japanese/ms_ie/start_ie.htm?ORBalwaysTunnel=yes

注意：このパラメータを使用して、常に接続しているクライアント用に、内側のファイアウォールとは切り離された URL を設定します。

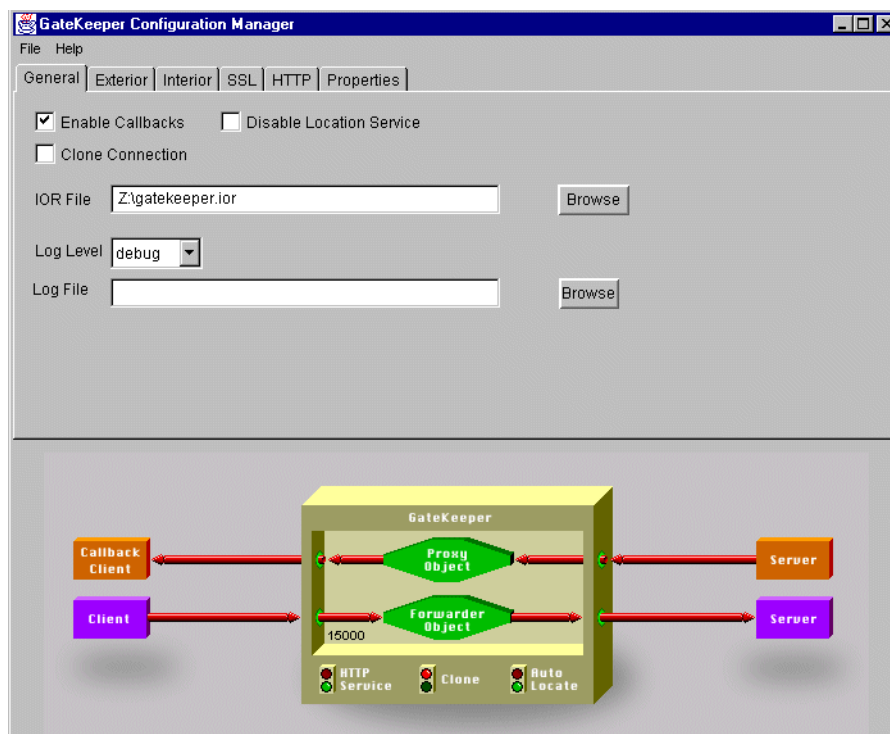
詳細は、[6.8.8 項「Discoverer Plus の起動ページでの SSL の有効化」](#)を参照してください。

7.10 Visibroker Gatekeeper の構成

この項では、Discoverer Plus で使用するための Visibroker Gatekeeper の構成方法について説明します。

Gatekeeper Configuration Manager を使用して Visibroker Gatekeeper を構成します。次の図を参照してください。

図 7-13 Visibroker GateKeeper Configuration Manager



7.10.1 Visibroker Gatekeeper の実行場所

Visibroker Gatekeeper を構成するには、次の 2 つのオプションがあります。

1. ファイアウォール・ポリシーで HTTP サーバーがポート 80 以外のポートで HTTP トラフィックを受信できるように定められている場合は、その HTTP サーバー・マシンに Visibroker Gatekeeper をインストールして実行します。

注意：この構成は、Discoverer Plus で使用することをお薦めします。

Visibroker Gatekeeper のデフォルトのポート番号は 15000 ですが、80 を除く、どの番号にでも変更することができます。詳細は、[7.10.2 項「HTTP サーバーでの Visibroker Gatekeeper の実行」](#)を参照してください。

2. ファイアウォール・ポリシーで HTTP サーバーがポート 80 のみで HTTP トラフィックを受信するように定められている場合は、別のマシンに Visibroker Gatekeeper をインストールします。

注意：この構成はセットアップに時間がかかるのみでなく管理も必要となるため、オプション 1 が使用できない場合以外は Discoverer Plus では使用しないことをお薦めします。

Visibroker Gatekeeper を再起動するたびに Visibroker Gatekeeper ファイル `gatekeeper.ior` を HTTP サーバーにコピーする必要があります。詳細は、[7.10.3 項「別のサーバーでの Visibroker Gatekeeper の実行」](#)を参照してください。

注意：SSL (Secure Sockets Layer) を使用する場合は、SSL の標準ポートであるポート 443 で Visibroker Gatekeeper を実行してください。詳細は、[6.8.3 項「SSL を使用するための Visibroker Gatekeeper のインストール」](#)を参照してください。

7.10.2 HTTP サーバーでの Visibroker Gatekeeper の実行

この項では、HTTP サーバーで Visibroker Gatekeeper を実行する方法について説明します。構成の選択の詳細は、[7.10.1 項「Visibroker Gatekeeper の実行場所」](#)を参照してください。

- Visibroker Gatekeeper は HTTP サーバーの次の場所にインストールされます。
<ORACLE_806_HOME>%vbroker%bin%gatekeeper.exe
- Visibroker Gatekeeper が、Discoverer Plus NT サービスによって自動的に起動されます。
- Visibroker Gatekeeper は HTTP サーバーのドキュメント・ディレクトリにある
<iSUITES_HOME>%apache%apache%htdocs%discwb4%applet ディレクトリで起動されます。Gatekeeper が、起動したディレクトリに `gatekeeper.ior` ファイルを書き出します。Discoverer Client がこのファイルを
<iSUITES_HOME>%apache%apache%htdocs%discwb4%applet ディレクトリから検索します。
- `gatekeeper.ior` ファイルが HTTP サーバーのドキュメント・ディレクトリにある
<iSUITES_HOME>%apache%apache%htdocs%discwb4%applet ディレクトリに表示されて

いるかどうかチェックして Gatekeeper が起動していることを確認します。また、gatekeeper.ior ファイルのタイム・スタンプが Discoverer のサービスが起動された時間と対応していることを確認する必要があります。対応していない場合は、ファイアウォールの外部から Discoverer に接続することはできません。

7.10.3 別のサーバーでの Visibroker Gatekeeper の実行

この項では、HTTP サーバー以外のサーバーで Visibroker Gatekeeper を実行する方法について説明します。構成の選択の詳細は、7.10.1 項「Visibroker Gatekeeper の実行場所」を参照してください。

- Visibroker Gatekeeper を別のサーバー・マシンにインストールするには、次の 2 つのオプションがあります。
 1. Visibroker Gatekeeper がインストールされている別のサーバー・マシンに iAS をインストールします。
 2. Gatekeeper のみをサーバー・マシンにインストールするために iAS の一部のみをインストールする場合は、Borland Software Corporation から Visibroker for Java 3.4 のスタンド・アロン版を別途購入することをお勧めします。詳細は、次の Visibroker の Web サイトを参照してください。
<http://www.borland.com/visibroker/>
- Visibroker Gatekeeper により作成される gatekeeper.ior ファイルを HTTP サーバーのドキュメント・ディレクトリにある
<iSUITES_HOME>%apache%apache%htdocs%discwb4%applet ディレクトリにコピーします。Visibroker Gatekeeper を再起動するたびにこのファイルをコピーしてください。
- Visibroker Gatekeeper は HTTP サーバー・マシン以外のマシンでのみ実行してください。Visibroker Gatekeeper が HTTP サーバー・マシンで自動的に起動しないようにするには、次のレジストリ設定を 0 に変更します。

HKEY_LOCAL_MACHINE¥Software¥Oracle¥DISCWB4_START_GATEKEEPER

7.10.4 Visibroker Gatekeeper ポートの構成

Discoverer で作業するには、Visibroker Gatekeeper を次のように構成します。

- Visibroker Gatekeeper を HTTP サーバーまたは別のサーバーにインストールします。
- Visibroker Gatekeeper は、デフォルトではポート 15000 をモニターまたはリスニングします。デフォルトのポート番号を変更する方法は、7.10.5 項「Visibroker Gatekeeper ポートの変更」を参照してください。
- Discoverer サービスにより、HTTP サーバーまたは別のサーバーで Visibroker Gatekeeper が起動されます。レジストリ・キー
HKEY_LOCAL_MACHINE¥SOFTWARE¥ORACLE¥DCW4_START_GATEKEEPER
は 1 に設定されます。

注意：Visibroker Gatekeeper を HTTP サーバーで実行している場合、Visibroker Gatekeeper のポートを 80 に設定することはできません。これは、ポート 80 が HTTP サーバーのデフォルトのポートであり、大部分の HTTP サーバーがこのポートにインストールされているためです。

7.10.5 Visibroker Gatekeeper ポートの変更

Visibroker Gatekeeper のデフォルトのポートを変更するには、次の手順を実行してください。

1. <ORACLE_806_HOME>%Vbroker%bin%gkconfig.exe を実行し、Gatekeeper Configuration Tool を起動します。
2. 「File」 → 「Open」を選択して、HTTP サーバー（またはその他のサーバー）のドキュメント・ルート・ディレクトリ（<i>SUITES_HOME>%apache%apache%htdocs%discwb4%applet など）にある Visibroker Gatekeeper の構成ファイル discwb4/applet/gatekeeper.properties を開きます。
3. 「Exterior」タブをクリックします。

図 7-14 Visibroker Gatekeeper のデフォルト・ポートの構成

General	Exterior	Interior	SSL	HTTP	Properties
Exterior Address					
Exterior Port: 15000					
Exterior Proxy Address					
Exterior Proxy Port					
Exterior Callback Port					
Exterior Callback Proxy Port					

4. 「Exterior Port」フィールドに新しいポート番号を入力します。
5. 「File」 → 「Save」を選択します。
6. Visibroker Gatekeeper Configuration Tool をクローズします。
7. Windows の「スタート」メニューから「設定」 → 「コントロール パネル」の順に選択し、「サービス」アイコンをダブルクリックし、「サービス」ダイアログ・ボックスを表示します。
8. 「OracleDiscoverer4i」を選択します。

9. 「停止」 ボタンをクリックしてから、「開始」 ボタンをクリックします。

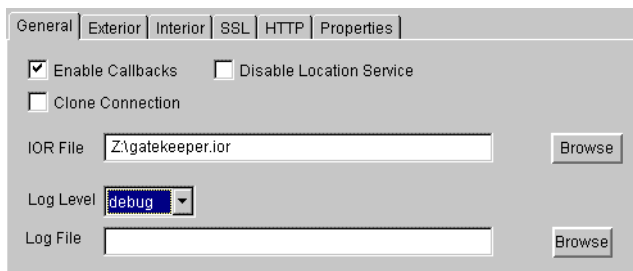
これで、Visibroker Gatekeeper が新しいポート番号を使用するように構成されました。

7.10.6 Visibroker Gatekeeper からのログ情報の取得

Visibroker Gatekeeper のログにより生成された情報量を変更するには、次の手順を実行してください。

1. <ORACLE_806_HOME>%Vbroker%bin%gkconfig.exe を実行し、Gatekeeper Configuration Tool を起動します。
2. 「File」 → 「Open」 を選択して、HTTP サーバー（またはその他のサーバー）のドキュメント・ルート・ディレクトリ（<i>SUITES_HOME</i>%apache%apache%htdocs%discwb4%applet など）にある Visibroker Gatekeeper の構成ファイル **gatekeeper.properties** を開きます。
3. 「General」 タブをクリックします。

図 7-15 Visibroker Gatekeeper のデバッグ・レベルの構成



4. 次のオプションからログ・レベルを選択します。
 - 「Quiet」 - メッセージを印刷しません。
 - 「Warning」 - エラー発生時にメッセージを印刷します。
 - 「Normal」 - エラー・メッセージおよび情報メッセージを印刷します。
 - 「Debug」 - すべてのメッセージを印刷します（デフォルト）。
5. 「File」 → 「Save」 を選択します。
6. Visibroker Gatekeeper Configuration Tool をクローズします。

7. Windows の「スタート」メニューから「設定」→「コントロールパネル」の順に選択し、「サービス」アイコンをダブルクリックし、「サービス」ダイアログ・ボックスを表示します。
 8. 「OracleDiscoverer4i」を選択します。
 9. 「停止」ボタンをクリックしてから、「開始」ボタンをクリックします。
- これで、Visibroker Gatekeeper が、選択したログ・レベルを使用するようになります。

7.10.7 Gatekeeper の前面で動作する NAT デバイスの構成

Network Address Translation (NAT) デバイスが Gatekeeper の前で動作するよう Gatekeeper を構成するには、次の手順を実行してください。

1. <ORACLE_806_HOME>%Vbroker%bin%gkconfig.exe を実行し、Gatekeeper Configuration Tool を起動します。
2. 「File」→「Open」を選択して、HTTP サーバー（またはその他のサーバー）のドキュメント・ルート・ディレクトリ（<SUITES_HOME>%apache%apache%htdocs%discwb4%applet など）にある Visibroker Gatekeeper の構成ファイル **gatekeeper.properties** を開きます。
3. 「Exterior」タブをクリックします。

図 7-16 Gatekeeper の前面で動作する NAT デバイスの構成

General	Exterior	Interior	SSL	HTTP	Properties
Exterior Address					
Exterior Port					
Exterior Proxy Address					
Exterior Proxy Port					
Exterior Callback Port					
Exterior Callback Proxy Port					

4. 「Exterior Proxy Address」フィールドで、NAT デバイスにより変換される Gatekeeper の IP アドレスを入力します。

たとえば、Gatekeeper の IP アドレスが 101.20.34.6 で、NAT デバイスにより 105.23.45.6 に変換される場合は、フィールドに 105.23.45.6 を入力します。

5. 「**Exterior Proxy Port**」フィールドで、NAT デバイスにより変換される Gatekeeper のポートを入力します。

たとえば、Gatekeeper のデフォルト・ポートが 15000 で、NAT デバイスにより 235 に変換される場合、フィールドには 235 と入力します。

6. 「**File**」→「**Save**」を選択します。
7. Visibroker Gatekeeper Configuration Tool をクローズします。
8. Windows の「**スタート**」メニューから「**設定**」→「**コントロールパネル**」の順に選択し、「**サービス**」アイコンをダブルクリックし、「サービス」ダイアログ・ボックスを表示します。
9. 「**OracleDiscoverer4i**」を選択します。
10. 「**停止**」ボタンをクリックしてから、「**開始**」ボタンをクリックします。

これで、Visibroker Gatekeeper が NAT デバイスを Gatekeeper の前面で使用するよう構成されました。

注意： Discoverer では、Gatekeeper の内側で NAT デバイスを使用することをサポートしていません。

Oracle9iAS Discoverer および Oracle Applications

Oracle9iAS Discoverer では、Oracle Applications の作業をユーザーに意識させずに実行するための追加機能が提供されます。

A.1 Discoverer Viewer の接続ダイアログでの「Applications ユーザー」チェックボックスの表示の指定

Discoverer Viewer の接続ダイアログを、Oracle Applications ユーザーがアプリケーションの接続機能を使用してログインするようにセットアップできます。そのためには、「Applications ユーザー」チェックボックスを表示するか、Oracle Applications に対して Discoverer がデフォルトで実行されるよう指定します。あるいは、これらの両方を行ってもかまいません。

接続ダイアログで「Applications ユーザー」チェックボックスを表示（または非表示）にセットアップするには、次の場所にある構成ファイルにアクセスします。

`<SUITES_HOME>apache¥apache¥htdocs¥disco4iv¥html¥viewer_config.xml`

このファイルの、`display_login_method_control` オプションを「true」または「false」に設定します。

Oracle Application に対して Discoverer Viewer がデフォルトで実行されるようユーザーが指定する場合（チェックボックスを表示している場合には、チェックまたはチェック解除します）には、`disco4iv.xml` ファイルの次の行を変更します。

`disco4iv.xml` - 変更前

```
<application xml_version="viewer41"/>
```

`disco4iv.xml` - 変更後

```
<application xml_version="viewer41" apps_logon="デフォルト値"/>
```

「デフォルト値」は、「true」または「false」です。

disco4iv.xml ファイルの編集については、[3.2.6 項「disco4iv.xml ファイルの編集 \(Discoverer Viewer のみ\)」](#)を参照してください。

A.2 Discoverer Plus の接続ダイアログでの「Applications ユーザー」チェックボックスの表示の指定

Discoverer Plus の接続ダイアログを、Oracle Applications ユーザーが Oracle Applications への接続機能を使用してログインするようにセットアップできます。そのためには、「Applications ユーザー」チェックボックスを表示するか、Oracle Applications に対して Discoverer がデフォルトで実行されるよう指定します。あるいは、これらの両方を行ってもかまいません。

接続ダイアログで「Applications ユーザー」チェックボックスを表示（または非表示）にセットアップするには、次の処理を行います。

1. pref.txt を開き、次の行の値を変更します。

変数	設定
ShowUserTypeChoice	設定する値 ■ 0: チェックボックスを非表示にする場合 ■ 1: チェックボックスを表示する場合
DefaultUserTypeIsApps	設定する値 ■ 0: ユーザーが Discoverer Plus を Oracle Application に対してデフォルトで実行しないよう指定する場合（チェックボックスを表示している場合には、チェックボックスがチェック解除されます）。 ■ 1: ユーザーが Discoverer Plus を Oracle Application に対してデフォルトで実行するよう指定する場合（チェックボックスを表示している場合には、チェックボックスがチェックされます）。

pref.txt の編集については、[2.4 項「エンド・ユーザー Preferences の編集」](#)を参照してください。

2. Windows の「スタート」メニューから「プログラム」→「Oracle Discoverer Server 4i Setup」→「Apply Preferences」の順に選択します。
3. 「OracleDiscoverer4i」サービスを、いったん停止してから再起動します。

「OracleDiscoverer4i」サービスの停止および再起動の詳細は、[2.2.4 項「Discoverer NT サービスの手動による起動および停止方法」](#)を参照してください。

A.3 Oracle Applications ユーザーのための NT レジストリ設定

pref.txt の NT レジストリ設定、特に Oracle Applications エンド・ユーザー用に設定できる NT レジストリ設定は 4 つあります。

- AppsGWYUID
- AppsFNDNAM
- ShowUserTypeChoice
- DefaultUserTypeIsApps

詳細情報：

- レジストリ設定については、[付録 C 「Oracle9iAS Discoverer レジストリの設定値」](#)を参照してください。
- pref.txt ファイルについては、[2.4 項「エンド・ユーザー Preferences の編集」](#)を参照してください。

Discoverer Viewer の構成ファイル

B.1 viewer_config.xml の構成設定値

領域	設定	説明
構成	base_dir="."	イメージおよびヘルプのディレクトリの位置がデフォルトと異なる場合、または disco4iv.xml の xsl_path 属性が http: 以外のプロトコル（xsl_path=file:/// <パス> など）を使用するように設定されている場合に指定します。
	image_dir="images"	Viewer イメージを含むディレクトリへのパス（絶対パスまたは構成要素の base_dir 属性によって指定されたディレクトリへの相対パス）を指定します。
外観	option id="display_login_method_control" state="false"	「Applications ユーザー」チェックボックスを表示するかどうかを指定します。
	option id="display_show_xml_link" state="true"	「XML の表示」リンクを各ページの下に表示するかどうかを指定します。
	option id="display_language_control" state="true"	「言語」ドロップダウン・リストを接続ダイアログに表示するかどうかを指定します。
	option id="use_javascript_enhanced_ui" state="true"	JavaScript 拡張ユーザー・インタフェースを使用するかどうかを指定します。

領域	設定	説明
presentation_templates	look name="Corporate Look"	表示テンプレートの名前。
	component name="xxx" state="true false"	現在の表示テンプレートの特定の UI コンポーネントを表示または非表示に設定します（コンポーネント名の詳細は、viewer_config.xml 内のコメントを参照してください）。
nl5	resource id="xxx" href="yyy" help="zzz"	特定言語 xxx に使用するメッセージ・ファイル yyy およびヘルプ・ファイル zzz を指定します。
ヘルプ	base_dir="help"	ヘルプ・ディレクトリを含むディレクトリへのパス（絶対パスまたは構成要素の base_dir 属性によって指定されたディレクトリへの相対パス）を指定します。
	page name="reconnect" file="connect.htm"	「reconnect」 オンライン・ヘルプ・ファイルのファイル名
	page name="roles" file="connect.htm"	「roles」 オンライン・ヘルプ・ファイルのファイル名
	page name="workbooks" file="list_of_workbooks.htm"	「workbooks」 オンライン・ヘルプ・ファイルのファイル名
	page name="worksheets" file="list_of_worksheets.htm"	「worksheets」 オンライン・ヘルプ・ファイルのファイル名
	page name="run_query" file="run_query.htm"	「run_query」 オンライン・ヘルプ・ファイルのファイル名
	page name="running_query" file="query_progress.htm"	「running_query」 オンライン・ヘルプ・ファイルのファイル名
	page name="data" file="worksheet_display.htm"	「data」 オンライン・ヘルプ・ファイルのファイル名
	page name="parameters" file="edit_parameters.htm"	「parameters」 オンライン・ヘルプ・ファイルのファイル名
	page name="options" file="options.htm"	「options」 オンライン・ヘルプ・ファイルのファイル名
	page name="edit_layout" file="edit_layout.htm"	「edit_layout」 オンライン・ヘルプ・ファイルのファイル名

領域	設定	説明
	page name="presentation" file="presentation_options.htm"	「presentation」 オンライン・ヘルプ・ファイルのファイル名
	page name="export" file="export.htm"	「export」 のファイル名

B.2 disco4iv.xml 内の構成設定値

設定	説明
apps_logon	ユーザーがデフォルトで Oracle Application に対して Discoverer Viewer を実行するように指定します。 - ユーザーがデフォルトで Oracle Application に対して Discoverer Viewer を実行するように指定する場合には、「true」を設定します。 - ユーザーがデフォルトで Oracle Application に対して Discoverer Viewer を実行しないように指定する場合には、「false」を設定します。 詳細は、付録 A.1 「Discoverer Viewer の接続ダイアログでの「Applications ユーザー」チェックボックスの表示の指定」を参照してください。
query_progress_delay	「Query Progress」 ページが表示（または再表示）されるまでのディレイを秒単位で指定します。この引数を使用すると、Viewer Servlet を使用して作成したアプリケーションヘッダーを戻すときに、「Query Progress」 ページを処理するためのコードを書き込む必要がありません。
redirect_delay	ブラウザが元の URL にリダイレクトされるまでのディレイを秒単位で指定します。この引数は、エクスポート操作中にブラウザがタイムアウトすることを防ぐために使用します。この設定は、fm 引数にも有効な値が指定されている場合にのみ適用されることに注意してください。
show_graphs	ワークブックとともに保存されたグラフを表示するかどうかを指定します。 - グラフを表示する場合には「true」を設定します（デフォルト）。 - グラフを表示しない場合には「false」を設定します（グラフが不要な場合は、これによりパフォーマンスが向上します）。
xml_version	XML ドキュメントが準拠する必要がある Discoverer Viewer DTD のバージョンを指定します。 - Oracle Discoverer 3i Viewer DTD に準拠する XML ドキュメントを生成する場合には、「viewer33」を設定します。 - Oracle9iAS Discoverer Viewer DTD に準拠する XML ドキュメントを生成する場合には、「viewer41」を設定します（デフォルト）。

設定	説明
xsl_path	disco4iv.xsl ファイルへのパスを指定します。
xsl_cache	サーブレットで xsl をキャッシュするかどうかを指定します。

Oracle9iAS Discoverer レジストリの設定値

C.1 pref.txt 内のレジストリ設定値

pref.txt 内のレジストリ設定値の変更については、[2.4 項「エンド・ユーザー Preferences の編集」](#)を参照してください。

影響される機能	Preference アイテム・キーの 名前	説明	デフォルト / 値
Locator	MachineIPs	すべての Discoverer Server Application サーバーの IP アドレスまたはマシン名。 pref.txt 内の唯一の必須設定項目です。	
	DCW4_LOCATOR_JVM	オプション・キー。HKEY_LOCAL_MACHINE \ SOFTWARE \ ORACLE 以下のレジストリにある場合には、Locator で使用される JVM を指している必要があります。このキーを設定すると、Locator がインストールされたマシンから Discoverer 管理者が Locator を停止することなくログオフできるようになります。	
Administrator	ShowUserTypeChoice	Discoverer Plus を Oracle Application に対して実行するかどうかを指定するエンド・ユーザー・チェックボックスをアクティブにします。Discoverer Viewer では未使用（viewer_config.xml で指定します）。	0（0=チェックボックスの非表示、1=チェックボックスを表示）

影響される機能	Preference アイテム・キーの名前	説明	デフォルト / 値
	DefaultUserTypeIsApps	ユーザーがデフォルトで Oracle Application に対して Discoverer Plus を実行するように指定します。Discoverer Viewer では未使用 (disco4iv.xml で指定します)。	0 (0= 実行する、1= 実行しない)
セッション・マネージャ	Timeout	Discoverer が、アイドル・セッションをデータベースから切断するまでの時間。最小時間は 180 秒です。	1800 (秒)
	RowsPerHTML	HTML の 1 ページに表示される行数。	25 (任意の整数)
アプリケーション	CacheFlushPercentage	キャッシュがいっぱいになった場合に、消去されるキャッシュの割合 (パーセント)。	25
	MaxVirtualDiskMem	データ・キャッシュに使用できるディスクの最大量。	1024000000
	MaxVirtualHeapMem	データ・キャッシュに使用できるヒープ・メモリーの最大量。	1024000000
	QueryBehavior	ワークブックを開いた後で取るアクション。	0 (0 = 問合せを実行する、1 = 問合せを実行しない、2 = 確認を求める)
	ShowDialogBitmaps	エンド・ユーザーの「Discoverer Plus」ダイアログに、ビットマップ・グラフィックを表示します。	1 (0 = オフ、1 = オン)
	DataFormat	ワークシート内のデータ・セルに適用される HTML 書式設定。	"<fontFormat fontName="Dialog" pitch="10" bold="false" italic="false" underline="false" strikethrough="false" foreground="0,0,0" background="255, 255, 255" valign="top"></fontFormat>"

影響される機能	Preference アイテム・キーの名前	説明	デフォルト / 値
	HeadingFormat	ワークシート内のヘディング・セルに適用される HTML 書式設定。	"<fontFormat fontName="Dialog" pitch="10" bold="false" italic="false" underline="false" strikethrough="false" foreground="0,0,0" background="204, 204, 204" halign="left" valign="top"></ fontFormat>"
	TotalsFormat	ワークシート内の総計セルに適用される HTML 書式設定。	"<fontFormat fontName="Dialog" pitch="10" bold="false" italic="false" underline="false" strikethrough="false" foreground="0,0,0" background="255, 255, 255" halign="left" valign="top"></ fontFormat>"
	NullValue	ワークシートに NULL 値を表示する方法を、たとえば "NULL"、"N/A"、"0" などと指定します。	"NULL"
	ExcelVersion	エクスポート先の Excel のバージョンを指定します。 次のように、Excel のバージョンによって 1 シートあたりの行数の制限が異なります。 ■ Excel95 の制限では、1 シートあたり 16384 行。 ■ Excel97 の制限では、1 シートあたり 65536 行。 この制限を超えると、Discoverer によって新しいシートが作成され、エクスポートされた行が配置されます。	"Excel95" ("Excel97")
データベース	DisableFanTrapDetection	ユーザー問合せのファントラップ検出を無効にします。	0 (0 = オン、1 = オフ)
	DisableMultiJoinDetection	複数結合検出を無効にします。	1 (0 = オン、1 = オフ)

影響される機能	Preference アイテム・キーの 名前	説明	デフォルト / 値
	DisableAutoQuery	0 に設定した場合、ワークシートが開かれたときに問合せは自動的に実行されません。	0 (0 = オン、1 = オフ)
	DisableAutoOuterJoinsOnFilters	フィルタの Automatic Outer Joins をオフにします。	0 (0 = オン、1 = オフ)
	ItemClassDelay	値リストを取り出す時間の上限値。	15 (秒)
	PredictionThresholdSeconds	問合せ予測時間が設定された秒数を超過する場合、ユーザーに警告します。	60 (秒)
	PredictionThresholdSecondsEnabled	問合せ予測しきい値を無効 (0) または有効 (1) にします。	1
	QueryTimeLimit	問合せ時間の上限を秒で示します。	1800 (秒)
	QueryTimeLimitEnabled	問合せ時間の制限を無効 (0) または有効 (1) にします。	1
	RowFetchLimit	取り出す行数の最大値。	10000 (行)
	RowFetchLimitEnabled	RowFetchLimit パラメータを無効 (0) または有効 (1) にします。	1
	RowsPerFetch	一度に取り出す行数。	100 (行)
	SummaryThreshold	サマリー表がこの日数以内に更新されている場合にのみ、サマリー表を使用します。	30 (日) 0 の場合はサマリーを使用しない。
	SummaryThresholdEnabled	有効 (1) にすると、SummaryThreshold で指定した値が使用されます。そうでない場合、Discoverer は使用できる限りサマリー表を使用します。	1
	MaterializedViewRedirectionBehavior	8.1.6 マテリアライズド・ビューへのリダイレクション	0 (0= 可能な限り常時、1= サマリー・データが古くない場合には常時、2= 行わない)
	DefaultEUL	すべてのユーザーが接続する EUL を設定します。各ユーザーは、「Discoverer Plus オプション」ダイアログからこのデフォルトを変更できます。	作成した任意の EUL

影響される機能	Preference アイテム・キーの 名前	説明	デフォルト / 値
Oracle Applications モード	AppsGWYUID	AOL セキュリティ DLL のパブリック・ユーザー名およびパブリック・パスワードを指定します。	"APPLSYSPUB/PUB"
	AppsFNDNAM	Oracle Applications のデータを格納する際のスキーマを指定します。	"APPS"
内部設定	BusinessAreaFastFetchLevel	ビジネス・エリアを取り出す際にキャッシュするコンポーネントやアイテムの量をレベルで指定します。	1
	ObjectsAlwaysAccessible	ビジネス・エリア上のオブジェクトおよびアイテムが、データベースに存在するかどうかを検証します。	0 (0= 検証する。1= オブジェクトおよびアイテムが存在するものと想定し、検証を行わない。)
	SummaryObjectsUseCachedAccessibility	キャッシュ内のサマリー導出オブジェクトにアクセスするかどうかを指定します。	0
	AvoidServerWildcardBug	NLS_LANG を Japanese_Japan.JA16SJIS に設定した RDBMS 7.3.2 の場合、特定のリスト・ボックスが空白で表示されないようにするには、この値を 1 に設定します。	0
	RdbFastSQLOff		0
問合せ予測とパフォーマンス	QPPEnable	1 に設定した場合、問合せ予測 / パフォーマンス (QPP) を使用します。	1 (0= 偽、1= 真)
	QPPCreateNewStats	1 に設定すると、新規の統計が記録されます。	1 (0= 偽、1= 真)
	QPPLoadStatsByObjectUseKey	1 に設定すると、同じオブジェクトの統計が最初に記録されます。	1 (0= 偽、1= 真)
	QPPUseCpuTime	アルゴリズム内で CPU 時間を使用して問合せ予測します。	1 (0= 偽、1= 真)
	QPPAvgCostTimeAlgorithm		2
	QPPMaxObjectUseKey	問合せ予測用にメモリーにキャッシュする統計量を決定します。	30

影響される機能	Preference アイテム・キーの名前	説明	デフォルト / 値
	QPPCBOEnforced	1 に設定すると、コスト・ベース・オブティマイザが使用されます。0 に設定すると、デフォルトのオブティマイザが使用されます。	1
	QPPObtainCostMethod		1
	QPPMinCost	この値より大きいコストの統計のみが記録または使用されます。	0
	QPPMaxStats	設定値の数のみ、以前の統計がロードされます。	500
	QPPMinActCpuTime	この値より大きい CPU 時間の統計のみが記録または使用されます。	0
	QPPMinActElapsedTime	この値より大きい実行時間の統計のみが記録または使用されます。	0
	QPPMinEstElapsedTime	この値より大きい予想経過時間の統計のみが記録または使用されます。	0
	UseOptimizerHints	次の場合、オブティマイザ・ヒントが SQL に追加されます。 1 に設定。	0
	QuerySQLFastFetchLevel		1
	SQLTrace	false (オフ)。	0
クロス集計レイアウト	Title	Discoverer 3.1 のワークシートで作成されたタイトルを表示します。	1 (0 = いいえ、1 = はい)
	Cell XGridline	水平罫線を表示します。	0 (0 = いいえ、1 = はい)
	Cell YGridline	垂直罫線を表示します。	0 (0 = いいえ、1 = はい)
	Axis Style	クロス集計の軸の位置。	2 (1 = インライン、2 = アウトライン)
表レイアウト	Title	Discoverer 3.1 のワークシートで作成されたタイトルを表示します。	1 (0 = いいえ、1 = はい)
	Cell XGridline	水平罫線を表示します。	0 (0 = いいえ、1 = はい)

影響される機能	Preference アイテム・キーの 名前	説明	デフォルト / 値
	Cell YGridline	垂直罫線を表示します。	0 (0 = いいえ、1 = はい)
	Row Headings	表形式のワークシート上に行数を表示します。	0 (0 = いいえ、1 = はい)

C.2 その他の Registry 設定値

影響される機能	Preference アイテム・キーの 名前	説明	デフォルト / 値
Locator コンポーネント	DCW4_START_LOCATOR	インストール中に設定します。 Locator を起動するかどうかを決定します。	1 (1, 0)
Preferences、 Collector および Session オブジェクト名	DCW4_INSTANCE_NAME	インストール中に設定します。 Discoverer Server マシン名 (その他のマシンまたはマスター・マシンの名前)。	ホスト名
Visigenic object activation daemon	DCW4_OADPORT	インストール中に設定します。マスター Discoverer サーバーごとのポート番号。	5432

C.3 レジストリに使用するメモリーの割当て

Preferences は、単一マシン (マスター Discoverer サーバー・マシン) のレジストリに格納されます。Discoverer ユーザーの数が増えた場合、まれに、レジストリのサイズを増やすための仮想メモリーが不足する場合があります。レジストリの割当てが不足した状態でシステムが実行しているためにレジストリのサイズを増やせない場合には、警告のメッセージが表示されます。

レジストリに割当て可能な仮想メモリーの量を増やすには、次の処理を行います。

1. Windows の「スタート」メニューから、「設定」→「コントロールパネル」を選択します。
2. 「システムのプロパティ」ダイアログを表示するための「システム」プログラム項目を選択します。

3. 「システムのプロパティ」ダイアログの「パフォーマンス」タブを表示し、「変更」ボタンをクリックして「仮想メモリ」ダイアログを表示します。
4. 「最大レジストリ サイズ (MB)」フィールドの値を増やします。

C

CORBA (Common Object Request Broker Architecture), 7-5

D

Demilitarized Zone (DMZ), 7-3

構成の例, 7-7

disco4iviewer.properties ファイル, 6-15

disco4iv.xml ファイル, 6-16, B-3

概要, 3-14

編集, 3-14

Discoverer

NT サービス, 2-6

URL 引数, 4-1

アーキテクチャ, 1-2

および Oracle Applications, A-1

および Oracle9i Application Server, 1-2

実行, 2-2

複数のマシンへのインストール, 3-1, 3-2

プロセス, 1-10

Discoverer Viewer の JavaScript 拡張ユーザー・インタフェース, 2-19

Discoverer Plus

URL 引数, 4-3

アーキテクチャ, 1-2

実行, 2-2

ファイアウォールのサポート, 7-1

プロセス, 1-10

Discoverer Services コンポーネント

Discoverer Viewer Java アプレットおよび HTML ページ, 1-6

Locator コンポーネント, 1-7

Preferences コンポーネント, 1-7

Session コンポーネント, 1-7

インストール先のマシン, 1-8

登録, 2-8

登録解除, 2-8

Discoverer Services コンポーネントの登録, 2-8

Windows の「スタート」メニューから, 2-11

コマンドラインから, 2-11

Discoverer Services コンポーネントの登録解除, 2-8

Windows の「スタート」メニューから, 2-11

コマンドラインから, 2-11

Discoverer Services の Locator コンポーネント, 1-7

Discoverer Services の Preferences コンポーネント, 1-7

Discoverer Services の Session コンポーネント, 1-7

Discoverer Viewer

JavaScript 拡張ユーザー・インタフェース, 2-19

SSL の証明書のインストール, 6-14

SSL の有効化, 6-14

XML/XSL, 5-2

アーキテクチャ, 1-3

カスタマイズ, 5-1

構成ファイル, B-1

実行, 2-5

Discoverer Viewer のカスタマイズ, 5-1

例, 5-6, 5-8

Discoverer Viewer の構成ファイル, B-1

Discoverer Viewer を使用した HTML の作成, 5-2

Discoverer Viewer を使用した自動ログイン, 5-21

Discoverer のインストール

Oracle9i Application Server の一部, 1-2, 3-2

複数のマシンへの, 3-1, 3-2

Discoverer の階層構造

Discoverer Services 層 (Plus および Viewer), 1-6

HTTP サーバー層 (Viewer のみ), 1-5

クライアント層 (Plus および Viewer), 1-4
データベース層 (Plus および Viewer), 1-8
DMZ, 「Demilitarized Zone」を参照, 7-3

G

Gatekeeper, 「Visibroker Gatekeeper」を参照, 7-5

H

HTTP Tunnelling の Gatekeeper 構成, 7-8, 7-12
httpd.conf ファイル, 6-14
HTTPS および Discoverer Viewer, 6-14
HTTP プロキシ, 7-15

I

iAS, 「Oracle9i Application Server」を参照, 1-2
IIOP Proxying の Gatekeeper 構成, 7-8, 7-9
IIOP (Internet Inter-ORB Protocol) プロトコル, 7-5

L

libvbj30ssl.so ファイル, 6-13

N

NAT デバイス
Gatekeeper の前面で動作する構成, 7-22
Network Address Translation デバイス, 「NAT デバイス」を参照, 7-22
NT サービス、Discoverer, 2-6

O

OAD (Object Activation Daemon), 2-8
Oracle Applications
「Applications ユーザー」チェックボックスの表示,
A-1, A-2
および Oracle9iAS Discoverer, A-1
レジストリ設定, A-3
Oracle Discoverer 4i NT サービス, 2-6
Oracle9i Application Server
および Oracle9iAS Discoverer, 1-2
Oracle9iAS Discoverer, 「Discoverer」を参照, 1-1
ORB - SSL 対応, 6-13
ORBalwaysProxy, 6-14, 7-16

ORBalwaysTunnel, 7-16
ORBenableSSL, 6-14

P

Plus, 「Discoverer Plus」を参照, 1-1
pref.txt ファイル
概要, 3-8
内部の設定値, C-1
編集, 3-11

S

SSL
Discoverer Plus, 6-5
Discoverer Plus 起動ページでの有効化, 6-13
Discoverer Viewer, 6-14
Discoverer Viewer での有効化, 6-17
SSL の証明書の取得, 6-6
X.509, 6-3
安全なリンクの設定, 6-14
概要, 6-2
公開鍵, 6-3
証明書, 6-3
認証局, 6-3
必要なファイルのインストール, 6-12, 6-13
秘密鍵, 6-3
ファイアウォール内での使用, 6-14

T

TCP/IP ポート, 7-4
tnsnames.ora ファイル
概要, 3-13
編集, 3-13

U

URL および Discoverer, 4-1
Discoverer Plus の URL 引数, 4-3
Discoverer Viewer の URL 引数, 4-6
Discoverer 用の URL の書式, 4-2
URL パラメータ - 複数使用, 6-13
URL を使用した Discoverer の起動, 4-1

V

vbj30ssl.dll ファイル, 6-13
vbrnissl.dll ファイル, 6-13
Video Stores Demo の例, 5-18
video.xml ファイル, 5-21
viewer_config.xml ファイル, 6-16, B-1
Viewer, 「Discoverer Viewer」を参照, 1-1
Visibroker Gatekeeper
 Discoverer Client の構成, 7-15
 Gatekeeper Configuration Manager, 7-17
 および SSL, 6-5
 概要, 7-5
 構成, 7-17
 実行先のマシン, 7-18
 詳細情報参照先の URL, 7-2
 バージョン, 7-6
 ポート, 7-19, 7-20
 ログ情報, 7-21

X

x509cert.jar, 6-12
X.509 証明書, 6-3
XSL
 HTML 作成のために使用, 5-2
 スタイルシート, 5-2
 ツール, 5-2

え

永続的な HTTP 接続, 7-15
エンド・ユーザー Preferences, 2-13, 3-8
 pref.txt, C-1
 概要, 2-13
 設定, 2-13

こ

公開鍵暗号化および SSL, 6-3

さ

サーバー・マシン
 削除, 2-18
 シャットダウン, 2-14
 追加, 2-18

し

シャットダウン, 2-14
 Discoverer システム全体, 2-16
 個々のサーバー, 2-15
証明書および SSL, 6-3

す

スケーラビリティおよび Discoverer, 1-12

せ

セキュリティ
 SSL, 6-1
 ファイアウォール, 7-1

そ

その他の Discoverer サーバー・マシン, 3-2, 3-5

と

トラブルシューティング, 2-21

ひ

秘密鍵暗号化および SSL, 6-3

ふ

ファイアウォール, 7-1
 Demilitarized Zone, 7-3
 Discoverer Plus との使用, 7-5
 Discoverer Server の配置, 7-7
 DMZ, 7-3
 FTP, 7-3
 HTTP Tunnelling の構成, 7-8, 7-12
 IIOP Proxying の構成, 7-8, 7-9
 SMTP, 7-3
 SSL, 6-14
 Visibroker Gatekeeper, 7-5
 Visibroker Gatekeeper の構成, 7-17
 概要, 7-3
 クライアント側, 7-3
 サーバー側, 7-3
 様々な構成, 7-6

- ポート 443, 7-4
- ポート 80, 7-4
- 保護ポート, 7-4
- ポリシー, 7-6
- 負荷均衡 (ロード・バランシング) および Discoverer,
1-12, 3-9
- 例, 3-10
- 分散インストール, 3-1

ほ

- ポート, 7-4
- 使用, 7-4
- ポート 443, 7-4
- ポート 80, 7-4
- 保護ポート, 7-4

ま

- マスター Discoverer サーバー・マシン, 3-2, 3-4

れ

- レジストリ設定値
 - pref.txt ファイル以外, C-7
 - pref.txt ファイル内, C-1