Oracle9*i*AS Clickstream Intelligence

Administrator's Guide

Release 2 (9.0.2)

May 2002

Part No. A90500-02

ORACLE

Oracle9*i*AS Clickstream Intelligence Administrator's Guide, Release 2 (9.0.2)

Part No. A90500-02

Author: Diane Smith

Contributors: Paul Mitchell, Roger Bolsius, Matthew Jakubiak, Kevin Malaney, Olaf Fermum, Biao Cheng, Nikolai Rochnik, Hong Xia, Jinning He, Sjon Link, Yiming Lu, Poonam Karkhanis, Jay Hauser, Alin Craciun, Yuanwei He, Donghui Wu, Dragan Djorovich.

Contents

Send Us Your Comments xi

Preface xiii

Intended Audience xiii
Documentation Accessibility xiii
Structure xiv
Related Documents xv
Conventions xvi

1 Introducing Clickstream Intelligence

Introduction to Oracle9iAS Clickstream Intelligence 1-1
Oracle9iAS Clickstream Intelligence Architecture 1-2
Contents of This Administrator's Guide 1-4

2 Getting Started

Introduction to the Runtime Administrator 2-1
Launching the Runtime Administrator 2-2
Logging In and Logging Out 2-2
The Runtime Administrator User Interface 2-3
The Configure Tab 2-3
The Manage Tab 2-5
Navigation 2-7
Workflow Outline 2-7
Task Summary by Interface Region 2-7

Oracle9iAS Clickstream Intelligence Architecture Revisited 2-8
Detailed Overview 2-8
Loading Web Data 2-10
Populating the Dimensions 2-12
Your Role as Clickstream Intelligence Administrator 2-12

3 Configuring Clickstream Intelligence

Site Configuration 3-1
The Site Summary Page 3-1
Create a Web Site Definition 3-2
Delete a Site 3-2
Edit an Existing Site 3-3
The Site Navigation Menu 3-4
Defining Site Parameters for the First Time 3-4
Editing a Site's Configuration Parameters 3-5
Deleting Site Parameter Definitions 3-5
Creating New Parameter Definitions 3-6
Site Details 3-6
User Identification Parameters 3-6
Web Server Authentication 3-7
User Identification by Query String 3-7
User Identification by Cookie Value 3-9
Visitor Identification Parameters 3-10
Client Hostname or IP Address 3-10
Visitor Identification by Query String 3-11
Visitor Identification by Cookie Value 3-11
Visitor Identification by Visitor Field 3-11
Session Identification Parameters 3-12
Session Identification by Visitor ID and Timeout Period 3-12
Session Identification by Query String 3-13
Session Identification by Cookie Value 3-13
Session Identification by Session Field 3-13
Data Sources 3-14
View All Data Sources 3-14
Create a Data Source 3-15

Internal Domains and Internal Sites 3-16
Define a New Internal Domain or Internal Site 3-17
Local Search Engines 3-17
Define a New Local Search Engine 3-18
Page Parameter Rules 3-18
Global Page Parameter Rules 3-20
Individual Page Parameter Rules 3-20
Configuration Summary 3-21
Site Validation 3-21
System Configuration 3-22
The System Navigation Menu 3-22
Data Formats 3-23
Define a New Data Format 3-23
Data Filters 3-24
Create a Data Filter 3-25
Resource Types 3-26
Create a Resource Type 3-27
Referring Search Engines 3-28
Create a Referring Search Engine 3-28
System Parameters 3-29
Process Definitions 3-29
Create a Process Definition 3-30
Load Clickstream 3-31
Load Dimensions 3-33
Refresh Summaries 3-35
Resolve IP Addresses 3-36
Restore a Previous Version 3-37

4 Loading the Clickstream Database

Database Processes 4-1
The Process Status Page 4-2
Controlling the Clickstream Daemons 4-2
Starting and Stopping a Process 4-3
How to Start a Database Process 4-4
Stopping a Database Process 4-4

Show Details for a Previous Run 4-4
Process Types 4-5
Load Clickstream 4-5
Load Dimensions 4-5
Refresh Summaries 4-6
Resolve Unknown IP Addresses 4-6
Restore a Previous Version 4-6
Process Execution States 4-6
A Successful Process 4-7
Stop a Process and Resume 4-8
Stop a Process and Undo 4-8
Resume a Process with an Error 4-8
Undo a Process with an Error 4-8
The Process Navigation Menu 4-9
Process Details 4-9
Process Messages 4-10
Process Definition 4-10
Subprocesses 4-10
Subprocess Details 4-10
Job Details 4-10
Process Data Packets 4-11
Delete Data Packets 4-11
Data Sources 4-11
Data Packets 4-11
The Data Packets Details Page 4-12

A Log File Types Supported by Clickstream Intelligence

Apache Log Format A-2
Common Log File Format A-2
Combined Log File Format A-2
Apache Log File Format Example A-2
Apache Log Format Details A-3
W3C Extended Log File Format A-5
W3C Extended Log File Format Example A-6
WC3 Extended Log File Format Details A-7

Microsoft IIS Extended Log File Format A-9
Microsoft IIS Extended Log File Format Example A-9
Microsoft IIS Extended Format Details A-11

B Installing a Dedicated Clickstream Database

Dedicated Clickstream Database Installation Summary B-2
Hardware Specifications B-4
Software Specifications B-5
Configuring the Database B-5
Tablespaces B-6
Creating Tablespaces With the Database Installation Wizard B-7
Manually Creating Tablespaces B-8
Tablespace Sizing Formula B-8
Calculated Tablespace Values B-9
Analytics Tablespace (CANA): B-10
Warehouse Data Tablespace (CDATA): B-10
Warehouse Index Tablespace (CINDX): B-10
Warehouse Summary Layer Tablespace (CSUMM): B-10
Warehouse Staging Tablespace (CSTG): B-10
Temporary Tablespace (TEMP): B-10
Initialization Parameters File B-11
Initialization Parameter Details B-11
Partitioning B-13
CREATE_ALL_PARTITIONS B-13
Syntax B-14
Example B-14
Parameters B-14
CREATE_ALL_FACT_PARTITIONS B-15
Procedure B-15
Example B-15
Parameters B-16
CREATE_ALL_MVIEW_PARTITIONS B-16
Syntax B-16
Example B-16
Parameters B-17

CREATE_FACT_PARTITIONS B-17
Syntax B-17
Procedure B-18
Parameters B-18
CREATE_MVIEW_PARTITIONS B-18
Syntax B-19
Example B-19
Parameters B-19
Running the Database Installation Wizard B-20
Launching the Database Installation Wizard B-20
How to Launch the Database Installation Wizard B-20
The Welcome Page B-20
The Install / Deinstall Page B-21
The SYS Logon Page B-21
The Runtime Administrator Schema Page B-21
The Analytics Schema Page B-22
The Language Selection Page B-22
The Tablespace Option Page B-22
The Database Sizing Parameters Page B-23
The Suggested Tablespace Size Page B-24
The Datafile Directory Page B-25
The Temporary Tablespace Page B-25
The Tablespaces Selection Page B-25
The Summary Page B-25
Database Installation Log B-26

C Installing a Standalone Collector Agent

Introduction to the Collector Agent C-1
Installing a Standalone Collector Agent C-2
Pre-installation Requirements C-2
Installation C-3
Configuring the Collector Agent C-4
The collector-agent.properties File C-4
The agent.xml File C-5
How to Start and Stop the Collector Agent C-5

Commands for Specific Collector Agents C-6
Commands for All Collector Agents C-6

D Error Messages

Generic Clickstream Errors D-1
Clickstream Loader Errors D-3

E Metadata Repository Postinstallation and Configuration

Metadata Repository Post-Installation Tasks E-1
Installing the Metadata Repository in a non-English Language E-4

Glossary

Index

Send Us Your Comments

Oracle9*i*AS Clickstream Intelligence Administrator's Guide, Release 2 (9.0.2)

Part No. A90500-02

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: clickdoc_us@oracle.com
- FAX: 650-506-7409 Attn: Paul Mitchell, Oracle Clickstream Intelligence
- Postal service:
  Oracle Corporation
  Oracle Clickstream Intelligence Documentation
  500 Oracle Parkway, MS 2op3
  Redwood Shores, CA 94065
  USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

This preface contains the following topics:

- **Intended Audience**
- **Documentation Accessibility**
- **Structure**
- **Related Documents**
- **Conventions**

Intended Audience

This manual is intended for individuals who will perform the following tasks:

- Configure and set up Oracle9*i*AS Clickstream Intelligence.
- Load data into the Oracle9*i*AS Clickstream Intelligence database.
- Manage other processes associated with the Oracle9*i*AS Clickstream Intelligence database.

The content in this manual is best utilized by System Administrators, Database Administrators, or other individuals who are familiar with the setup and layout of the Web site(s) you want to analyze.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of

assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle Corporation is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at

```
http://www.oracle.com/accessibility/
```

Accessibility of Code Examples in Documentation

JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle Corporation does not own or control. Oracle Corporation neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Structure

This document contains:

Chapter 1, "Introducing Clickstream Intelligence"

This chapter provides a high-level overview of Oracle9*i*AS Clickstream Intelligence and its architecture.

Chapter 2, "Getting Started"

Provides information that enables you to launch, log in to, and navigate through the Runtime Administrator user interface. Discusses and illustrates the Clickstream Intelligence architecture.

Chapter 3, "Configuring Clickstream Intelligence"

Explains how to configure the Runtime Administrator for optimal Web log data aquisition. Describes how to set up Clickstream sites, and defines all configurable parameters that influence or characterize data stored in the database.

Chapter 4, "Loading the Clickstream Database"

Explains how to load Web data and defines all other database processes.

Appendix A, "Log File Types Supported by Clickstream Intelligence"

Describes the Web server log file formats supported by Clickstream Intelligence. Provides example log file entries and explains all log file fields for each supported log format.

Appendix B, "Installing a Dedicated Clickstream Database"

Describes installation of a database dedicated solely to collection of clickstream data, and provides a step-by-step tour through the Clickstream Intelligence Database Installation Wizard.

Appendix C, "Installing a Standalone Collector Agent"

Describes standalone Collector Agent installation and defines all configurable attributes. Provides the commands used to start, stop, and manage one or more standalone Collector Agents.

Appendix D, "Error Messages"

Provides a list of all error messages that may be displayed when using Clickstream Intelligence.

Appendix E, "Metadata Repository Postinstallation and Configuration"

Provides the steps necessary to begin using Clickstream Intelligence with the metadata repository (also referred to as the *infrastructure database*) installed with Oracle9*i* Application Server.

Glossary

Defines terms and concepts used throughout the *Oracle9iAS Clickstream Intelligence Administrator's Guide.*

Related Documents

For more information, see the following documentation:

- *Oracle9iAS Oracle9iAS Clickstream Intelligence User's Guide*
- *Oracle9i Clickstream Intelligence Data Model Reference*
- *Oracle9iAS Discoverer Plus User's Guide*

- *Oracle9iAS Discoverer Configuration Guide*
- *Oracle9i Application Server Administrator's Guide*
- *Oracle9i Application Server Concepts*

In North America, printed documentation is available for sale in the Oracle Store at

`http://oraclestore.oracle.com/`

Customers in Europe, the Middle East, and Africa (EMEA) can purchase documentation from

`http://www.oraclebookshop.com/`

Other customers can contact their Oracle representative to purchase printed documentation.

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network (OTN). You must register online before using OTN; registration is free and can be done at

`http://otn.oracle.com/membership/index.htm`

If you already have a username and password for OTN, then you can go directly to the documentation section of the OTN Web site at

`http://otn.oracle.com/docs/index.htm`

Conventions

This section describes the conventions used in the text and code examples of the this documentation set. It describes:

- Conventions in Text
- Conventions in Code Examples

Conventions in Text

We use various conventions in this text to help you quickly identify special terms. The following list describes those conventions, as applicable to the *Oracle9iAS Clickstream Intelligence Administrator's Guide*.

Convention	Meaning	Example
Bold	Bold typeface indicates key terms defined in the text or in the glossary. Boldface type also denotes links and buttons that appear as user interface (UI) components, and may be used for emphasis in lists or text.	Click the **Show Details** button to view and track information about the database process.
Italic	Italic typeface indicates book titles, chapter titles within this book, and may be used for emphasis.	*Oracle9iAS Clickstream Intelligence Administrator's Guide*
`lowercase monospace (fixed-width font)`	Lowercase monospace typeface indicates executables, filenames, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names, and connect identifiers, as well as user-supplied database objects and structures, column names, packages and classes, usernames and roles, program units, and parameter values. **Note:** Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Populate the EUL with the Clickstream EUL objects contained in the `clickstream_intelligence_eul_full.eex` file.
`UPPERCASE monospace italic (fixed-width font) italic`	Uppercase monospace italicized text indicates placeholders or variables for which you must supply particular values.	*`CLICK_HOME`* indicates the directory in which Oracle9*i*AS Clickstream Intelligence is installed.

Conventions in Code Examples

Code examples illustrate command-line statements. They are displayed in a monospace (fixed-width) font and separated from normal text.

The following table describes typographic conventions used in code examples and provides examples of their use.

Convention	Meaning	Example
[ ]	Brackets enclose one or more optional items. Do not enter the brackets.	`DECIMAL (`*`digits`* `[ , `*`precision`* `])`
{ }	Braces enclose two or more items, one of which is required. Do not enter the braces.	`{ENABLE | DISABLE}`

Convention	Meaning	Example
|	A vertical bar represents a choice of two or more options within brackets or braces. Enter one of the options. Do not enter the vertical bar.	{ENABLE | DISABLE} [COMPRESS | NOCOMPRESS]
...	Horizontal ellipsis points indicate either:	
	■ That we have omitted parts of the code that are not directly related to the example	CREATE TABLE ... AS *subquery*;
	■ That you can repeat a portion of the code	SELECT *col1*, *col2*, ... , *coln* FROM employees;
. . .	Vertical ellipsis points indicate that we have omitted several lines of code not directly related to the example.	
UPPERCASE ITALIC	Italicized text indicates placeholders or variables for which you must supply particular values.	*CLICK_HOME*/bin
lowercase	Lowercase typeface indicates executables, filenames, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names, and connect identifiers, as well as user-supplied database objects and structures, column names, packages and classes, usernames and roles, program units, and parameter values. URLs are also written in lowercase typeface. **Note:** Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	http://otn.oracle.com/ *INSTALL_DIR*/click/bin

1

Introducing Clickstream Intelligence

This chapter provides an overview of Oracle9*i*AS Clickstream Intelligence, briefly outlines its architecture, and describes the contents of this Administrator's Guide.

It contains the following sections:

- **Introduction to Oracle9iAS Clickstream Intelligence**
- **Oracle9iAS Clickstream Intelligence Architecture**
- **Contents of This Administrator's Guide**

Introduction to Oracle9iAS Clickstream Intelligence

Welcome to Oracle9*i*AS Clickstream Intelligence!

Clickstream Intelligence is a Web-based, comprehensive analytical tool that enables you to to acquire, analyze, and report on Web interactions with customers, suppliers, and employees. It is one of the key Business Intelligence components of **Oracle9*i* Application Server** (Oracle9*i*AS), a comprehensive and integrated application server that manages the technical complexity of assembling a complete middle-tier Internet infrastructure that can scale with your business. Clickstream Intelligence leverages the business intelligence tools in Oracle9*i*AS and Oracle9*i*DS -- Oracle9*i* Warehouse Builder, Oracle9*i*AS Discoverer, and Oracle9*i* database -- to provide an integrated and extensible solution for measuring Web traffic and improving Web site effectiveness.

The **Runtime Administrator** simplifies the setup, management, and deployment of clickstream data sources. This Web-based tool enables you to define how Clickstream Intelligence interprets Web log file data, determine which types of data you want to track, and configure and populate the database for optimal storage of clickstream data. The *Oracle9iAS Clickstream Intelligence Administrator's Guide* describes how to use the Runtime Administrator.

Clickstream Analytics provides more than 150 pre-configured reports that display data acquired from your Web sites. Clickstream reports, called worksheets, are grouped with other reports of a similar nature to form workbooks that can be accessed and viewed with Oracle9*i*AS Discoverer Viewer and Oracle9*i*AS Discoverer Plus. To learn more about Clickstream Analytics, please reference the *Oracle9iAS Clickstream Intelligence User's Guide*.

Oracle9*i*AS Clickstream Intelligence can be installed into the Oracle9*i* database included in Oracle9*i* Application Server, or can be installed into a standalone Oracle9*i* Enterprise Edition database. For more information about installing a dedicated Clickstream database, see Appendix B, "Installing a Dedicated Clickstream Database", located in this Administrator's Guide.

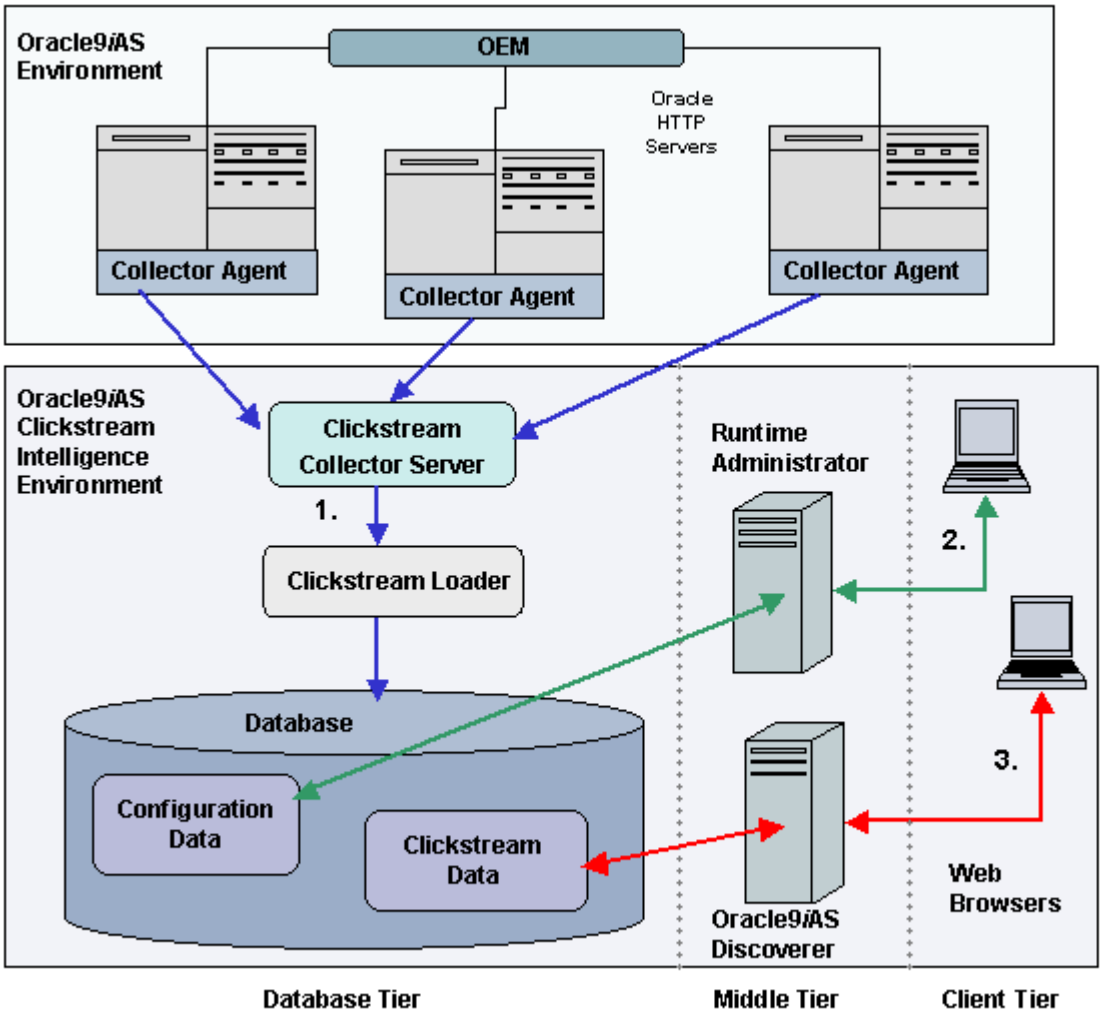
The Oracle9*i*AS Clickstream Intelligence database model is built via Oracle9*i* Warehouse Builder, a tool for designing and deploying databases, and from which it also derives its extensible platform. For further information about the Clickstream Intelligence data model, see the *Oracle9i Clickstream Intelligence Data Model Reference*.

Oracle9iAS Clickstream Intelligence Architecture

A graphical representation of Oracle9*i*AS Clickstream Intelligence architecture is displayed in the figure that follows. The **blue arrows (1)** represent overall data flow from the Oracle9*i*AS environment (Oracle HTTP servers) to the Clickstream Intelligence environment. **Log files** containing Web data are produced by **Oracle9*i* Application Server**. The Clickstream **Collector Agent**, which resides on the server machine, breaks the log files into **data packets**. The packets are compressed to facilitate transfer to the **Collector Server**, which subsequently uncompresses the log file data. The **Clickstream Loader** then parses, transforms, and filters the raw log files as they are loaded into the **database**.

Note: You can use Oracle Enterprise Manager (OEM) for administering the Clickstream Intelligence Collector Agent. OEM provides a Web-based tool that enables you to configure individual Collector Agents, and view certain details related to Collector Agent activity. For more information, see *Oracle9i Application Server Administrator's Guide*.

Figure 1–1 Oracle9iAS Clickstream Intelligence Architecture



The **green arrows** **(2)** illustrate the flow of information during Clickstream Intelligence configuration and management. The **Runtime Administrator's** Web-based interface enables your company's System Administrator to manage and

configure parameters that handle the collection and processing of data for your Clickstream sites. This **configuration data** is stored in the **database**. Your Clickstream Intelligence Administrator can access and modify all stored configuration data with a **Web browser**.

The **red arrows (3)** represent data flow between your **Web browser**, **Oracle9*i*AS Discoverer**, and **clickstream data** stored in the **database** each time you run Clickstream Analytics. Both Oracle9*i*AS Discoverer Plus and Oracle9*i*AS Discoverer Viewer enable you to view analytical reports based on data from your Web sites. When using **Oracle9*i*AS Discoverer Plus**, your client machine must run a Java-enabled Web browser (such as Microsoft Internet Explorer or Netscape Navigator with Oracle JInitiator). In the case of **Oracle9*i*AS Discoverer Viewer**, the only requirement for the client machine is that it can run HTML and JavaScript through a Web browser. For more information about the capabilities and functionality of Oracle9*i*AS Discoverer, please reference the *Oracle9iAS Discoverer Configuration Guide* and the *Oracle9iAS Discoverer Plus User's Guide*.

Contents of This Administrator's Guide

The *Oracle9iAS Clickstream Intelligence Administrator's Guide* describes how to use the Clickstream Intelligence Runtime Administrator. Specifically, you'll learn how to:

- **Launch and log in** to the Runtime Administrator. (Chapter 2, "Getting Started")
- Interpret the **Clickstream Intelligence architecture** in the context of Oracle9*i* Application Server, the product suite to which it belongs. A more detailed description of the architecture also describes the characteristics and functionality of all **Clickstream components**. (Chapter 2, "Getting Started")
- Navigate and understand all **user interface** (UI) components of the Runtime Administrator. (Chapter 2, "Getting Started")
- **Define and configure** the way that Oracle9*i*AS Clickstream Intelligence acquires, processes, and stores data acquired from your Web site(s). (Chapter 3, "Configuring Clickstream Intelligence")
- **Load data** into the Clickstream Intelligence database. (Chapter 4, "Loading the Clickstream Database")
- Run and monitor specific **processes** related to the Clickstream Intelligence database. (Chapter 4, "Loading the Clickstream Database")
- Identify the **log formats** supported by Clickstream Intelligence. (Appendix A, "Log File Types Supported by Clickstream Intelligence")

- Install a **dedicated Oracle9*i* Enterprise Edition database** for storage of all log data collected from your Web sites. (Appendix B, "Installing a Dedicated Clickstream Database")
- **Install a standalone Collector Agent** for use with Web servers other than Oracle9*i* Application Server. (Appendix C, "Installing a Standalone Collector Agent")
- Interpret and understand **error messages** that may be displayed when using Oracle9*i*AS Clickstream Intelligence. (Appendix D, "Error Messages")
- Perform the **post-installation tasks** that enable use of Oracle9*i*AS Clickstream Intelligence with the **infrastructure database (metadata repository)** included with Oracle9*i*AS. (Appendix E, "Metadata Repository Postinstallation and Configuration")
- Understand the **concepts and terminology** used in the Oracle9*i*AS Clickstream Intelligence documentation set. (Glossary)

2

Getting Started

This chapter provides an overview of the **Runtime Administrator** user interface (UI) and architecture, and describes the tasks for which a Clickstream Intelligence Administrator is responsible. You'll find information on the following topics:

- **Introduction to the Runtime Administrator**
- **The Runtime Administrator User Interface**
- **Oracle9iAS Clickstream Intelligence Architecture Revisited**
- **Your Role as Clickstream Intelligence Administrator**

Introduction to the Runtime Administrator

The **Runtime Administrator** is the tool that enables you to specify the criteria with which Oracle9*i*AS Clickstream Intelligence acquires and processes Web data. After configuring the Runtime Administrator, you will be able to retrieve and process log data from one or more sites on the World Wide Web (WWW), and then load that data into your database. The data stored in the database provides an information resource from which analytical reports can be created.

To get started, you need to first define a **Clickstream site** from which Web log data will be acquired. For each site created, you must define data collection parameters that determine the type of Web data you want to store in the Clickstream Intelligence database. You must also specify the **data source**(s) from which log data will be acquired. With Oracle9*i*AS Clickstream Intelligence, you collect only the data your company needs because you define the parameters that characterize the information obtained from Web server log files.

Launching the Runtime Administrator

To launch the Runtime Administrator, open an Internet browser window and enter the URL provided by your Clickstream Intelligence Administrator. The URL is written in the following format:

`http://HOST:PORT/click/`

The *HOST* is the name of the computer on your network that provides access to Oracle9*i*AS Clickstream Intelligence. The *PORT* is the HTTP port number on which the host (server) listens. To determine the port number for your host, ask your Database Administrator (DBA) or Clickstream Administrator. You can also go to the following configuration file located in the directory in which Oracle9*i*AS was installed:

`(UNIX) ORACLE_HOME/Apache/Apache/conf/httpd.conf`

Upon entry of the appropriate URL, the **Clickstream Intelligence Entry** page appears with hyperlinks to the **Runtime Administrator**, **Analytics- Discoverer Viewer**, and **Analytics- Discoverer Plus**. From this page, click the link for the **Runtime Administrator**.

Logging In and Logging Out

After you launch Oracle9*i*AS Clickstream Intelligence, you must log in from the **Login to Clickstream Intelligence Runtime Administrator** page. To do this, enter a valid username and password and then click the **Login** button. If you have not been assigned a username and password, see your Clickstream Administrator.

You can log out of the Runtime Administrator in any of the following ways:

- Click the **Logout** global navigation button located in the upper right corner of the Runtime Administrator UI.
- Click the **Logout** link located at the bottom of any page.

Before configuring sites with the Runtime Administrator, you should first familiarize yourself with the UI components and navigational pathways used to complete the specific tasks described in Chapter 3, "Configuring Clickstream Intelligence" The sections that follow introduce the Runtime Administrator UI and outline a typical navigational scheme to get you started.

The Runtime Administrator User Interface

All UI components can be found within two major functional areas of the Runtime Administrator: the **Configure** tab and the **Manage** tab. These first-level tabs are located on the right side of the **Horizontal Navigation** bar at the top of your screen.

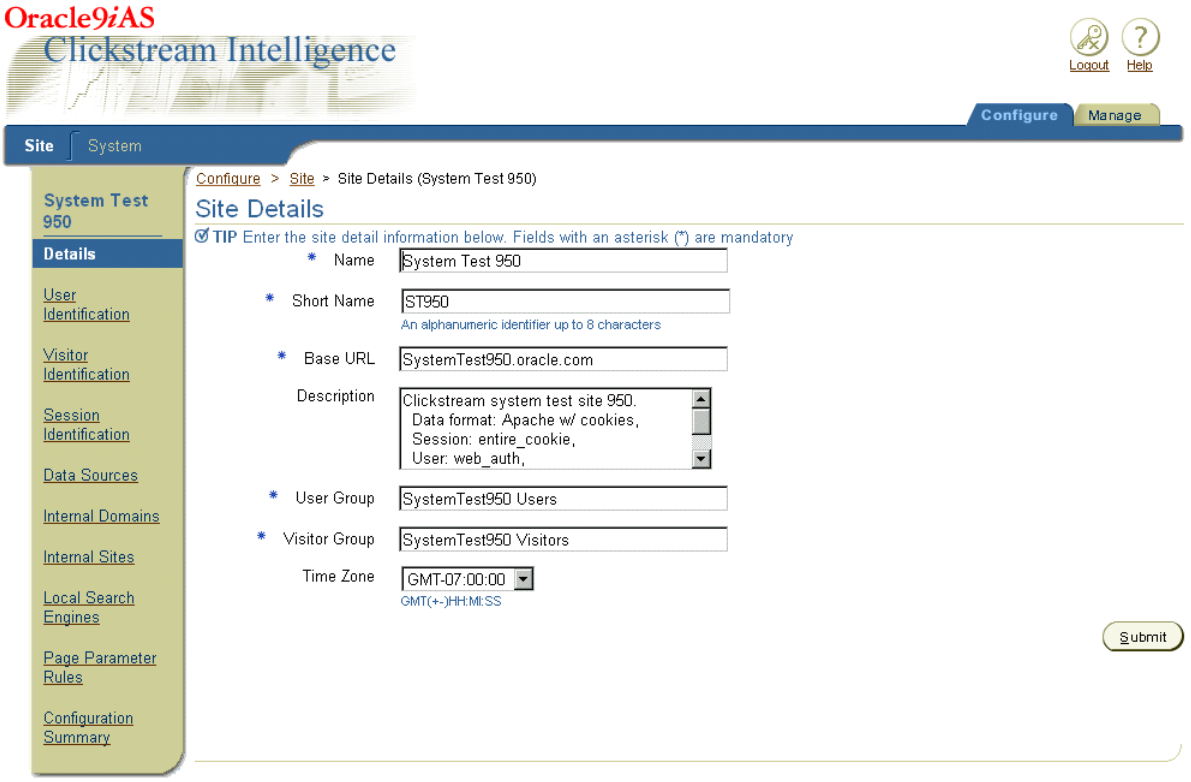
The Configure Tab

The **Configure** tab is where you define and modify all information related to data collection. It is divided into two sections, each represented by a **link** on the left side of the **Horizontal Navigation** bar. The links described below appear only when the **Configure** tab is selected.

- **Site:** location of all site-related parameters. All changes are made on a site-by-site basis and apply only to the particular site whose configuration information you have accessed.
- **System:** location of system-wide parameters and operations. Parameters are not configurable on a per-site basis; additions or changes you make apply to the entire system (all sites collectively).

The figure below displays UI components that can be accessed from the **Configuration Summary** page, located within the **Configure** tab. Labeled elements are identified below the figure.

Figure 2–1 The Configuration Summary Page



1. Global Navigation buttons (**Logout** and **Help**)
2. Tabs (**Configure** and **Manage**)
3. **Horizontal Navigation** bar (the blue bar across the top of your screen that displays specific links when a particular tab is selected).
4. Tab-specific links on the Horizontal Navigation bar (**Site** and **System** - displayed only when the **Configure** tab is selected)
5. Site Navigation menu (This is displayed only when the **Site** link is selected. Similarly, the **System Navigation** menu appears when the **System** link is selected)
6. **Site Navigation** menu options (**Details**, **User Identification**, etc.)
7. Page button (**Create Site**)

The Manage Tab

The **Manage** tab is where you perform operations related to your database. It is comprised of two sections, each represented by a link on the left side of the **Horizontal Navigation** bar. The links described below appear only when the **Manage** tab is selected.

- **Processes:** contains functionality that enables you to load data for validated sites, load dimension data, restore to a previous version of data, resolve unknown IP addresses, and refresh summaries. You can also view the status of current and previous processes.
- **Data Sources:** enables you to view the status of the Web log data packets as they move from the Collector Server to the Clickstream Loader, and finally to the Clickstream fact tables. You can also view all data packets associated with a particular process or data source.

The figure below displays the **Process Status** page, located within the **Manage** tab. Labeled UI components are identified below the figure.

Figure 2–2 The Process Status Page

1. Global Navigation buttons (**Logout** and **Help**)
2. Tabs (The **Manage** tab is selected; the **Confgure** tab is not selected)
3. **Horizontal Navigation** bar (the blue bar across the top of your screen that displays specific links when a particular tab is selected).
4. Tab-specific links on the Horizontal Navigation bar (**Processes** and **Data Sources -** displayed only when the **Manage** tab is selected.
5. Page header (**Process Status**)
6. Page subheader (**Currently Running**)
7. Page button (**Undo**)

Navigation

The administrative tasks for which you will use the Runtime Administrator follow a logical navigation scheme, as outlined in the following sections.

Workflow Outline

A general workflow, from creating a site to loading the database, is briefly outlined below. For details about any of the following tasks, please reference the appropriate sections of this Administrator's Guide.

1. Create a site (**Configure** tab, **Site** link).
2. Configure the options on the **Site Navigation** menu.

   (Definition of global page parameters is optional and not required for site validation.)
3. View the **Configuration Summary** and modify as needed.
4. Successfully validate the site (**Site** link, **Configuration Summary** page).
5. Define one or more process definitions (**System** link, **Process Definitions** page).
6. Load data into the database by starting a process type, such as "Load Clickstream" (**Manage** tab, **Process Status** page).

Task Summary by Interface Region

The paragraphs below summarize the types of tasks you can complete within specific navigational regions of the Runtime Administrator.

- **Configure tab**: Define and modify criteria related to data collection and processing.
  - **Site link** - Create and configure sites, validate each site, view summarized parameters for individual sites.
  - **System link** - Define global (system-wide) parameters, such as data formats, filters, resource types, referring search engines, and process definitions.
- **Manage tab**: Execute processes related to the database, view all data sources, and view data packets details.
  - **Processes link** - Start and monitor database processes, view details for previous runs.

– **Data Sources link** - View data packets associated with a specific data source, view details about all data packets for any particular data source.

Oracle9iAS Clickstream Intelligence Architecture Revisited

The sections that follow take a closer look at Oracle9*i*AS Clickstream Intelligence architecture.

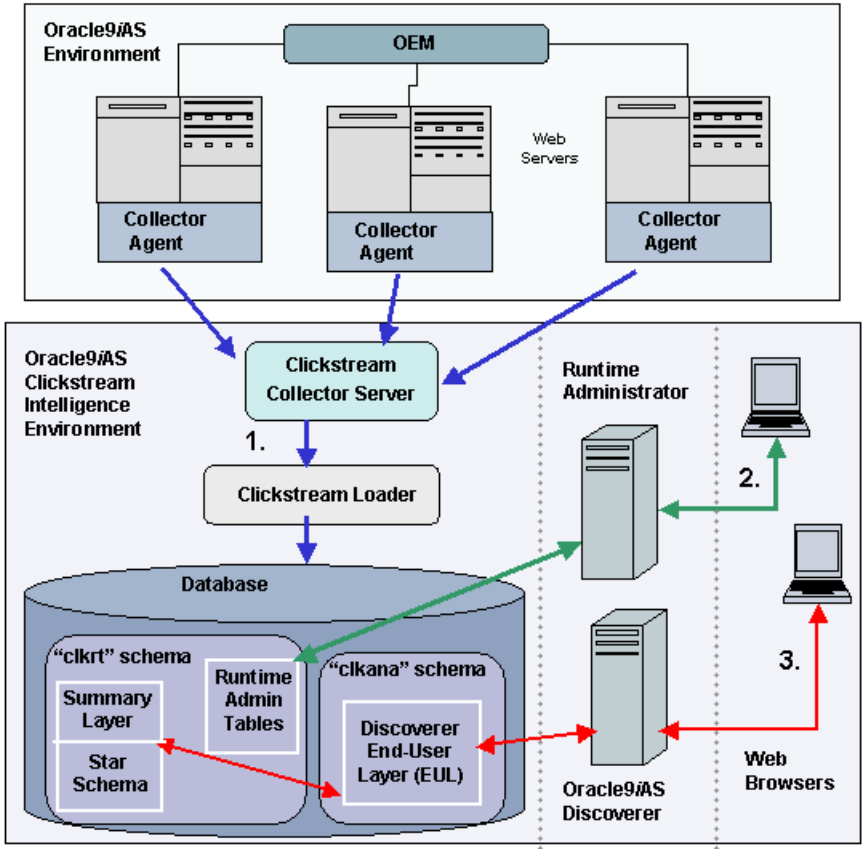
Detailed Overview

The Oracle9*i*AS Clickstream Intelligence architecture is represented in the following diagram. The **blue arrows (1)** represent data flow from the server farm to the Oracle9*i*AS Clickstream Intelligence database. Log files containing clickstream data are produced by **Web servers**, such as Oracle HTTP Server in Oracle9*i*AS, or on other "non-*i*AS" servers. A **Collector Agent** is installed on the server machine, or **data source**. It breaks log files into **data packets**, which are then compressed to facilitate transmission to the Collector Server. The **Collector Server** retrieves and uncompresses the data packets; the **Clickstream Loader** then parses and transforms the log data prior to database loading.

To define data collection parameters and processes, simply access the Runtime Administrator from your **Web browser**. The **green arrows (2)** represent the flow of information during Clickstream Intelligence set-up and management. All configuration data is accessed via Web browser and stored in the **Runtime Administrator tables**, located within the **`clkrt` database schema**.

The **red arrows (3)** represent data flow when you run the pre-defined reports that comprise **Clickstream Analytics**. All Clickstream reports created with **Oracle9*i*AS Discoverer** technology can be accessed with a **Web browser**. Both Oracle9*i*AS Discoverer Plus and Oracle9*i*AS Discoverer Viewer enable you to view analytical reports based on data from your Web sites. To create and run reports that contain your Web data, Discoverer uses information in the **Discoverer End User Layer** (EUL), located in the **`clkana` database schema**. Discoverer also utilizes data from the **summary layer** and **star schema**, both located in the **`clkrt` database schema**. For more information about Clickstream Analytics, see the *Oracle9iAS Clickstream Intelligence User's Guide*. For more information about Oracle9*i*AS Discoverer, please reference the *Oracle9iAS Discoverer Configuration Guide* and the *Oracle9iAS Discoverer Plus User's Guide*.

Figure 2–3 Oracle9iAS Clickstream Intelligence Detailed Architecture



The list that follows describes some of the components displayed in the architecture diagram:

- **Web Server:**

  the storage location of log files containing clickstream data.

- **Collector Agent:**

  monitors the Web server's log directory and creates Clickstream **data packets** from the log files. The Collector Agent resides on the Web server machine.

- **Collector Server:**

retrieves data packets from the **Collector Agent**, which runs on the Web server. It uncompresses the data packets for subsequent transfer to the **Clickstream Loader.** The Collector Server resides on the database machine.

- **Clickstream Loader:**

  filters and transforms raw log files as it loads them into the database. The Loader uses a site's **log format** information and other **site configuration** parameters to parse and transform the Clickstream log files before populating the database. The Clickstream Loader resides on the same machine as the Collector Server.

- **Clickstream Intelligence Database:**

  the storage location of all Web data. After data is loaded into the database, summaries should be refreshed. Use of a dedicated Clickstream database is highly recommended. For more information, see Appendix B, "Installing a Dedicated Clickstream Database".

- **Oracle9iAS Discoverer Plus/Viewer:**

  the two components of Oracle9*i* Discoverer than enable you to view Clickstream Analytics While Oracle9*i*AS Discoverer Plus provides both creation and viewing of reports, Oracle9*i*AS Discoverer Viewer enables viewing of pre-configured reports via Web browser only.

- **Runtime Administrator:**

  the component of Oracle9*i*AS Clickstream Intelligence that is used by System Administrators to create and validate sites, configure Web data collection parameters, and manage the database.

- **Runtime Administrator Schema (`clkrt`):**

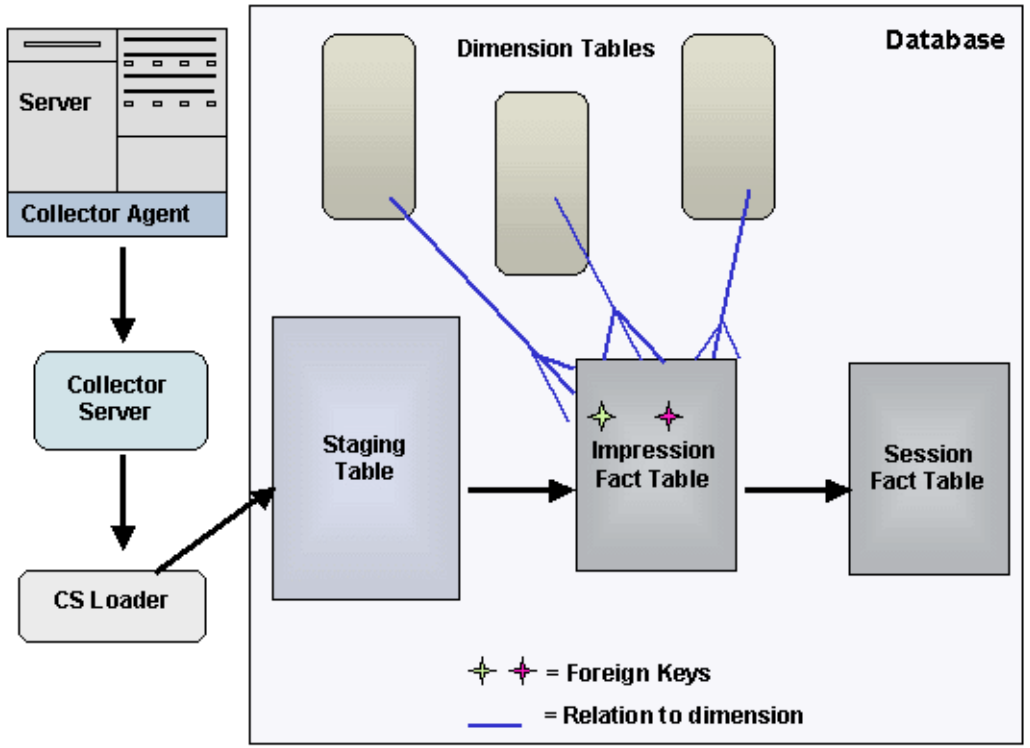
  the storage location of all configuration data. The Runtime Administrator schema also stores the audit trail for each load. The audit trail, which contains information such as loading errors, statistics, and load status, is maintained historically. This enables you to view details and results from previous loads.

Loading Web Data

Each time Web log data is loaded into the Clickstream Intelligence database, it is first stored in a generalized **staging table** (see the figure below). This staging table is used to populate both the Impression fact table and the Session fact table when you perform a **Clickstream Load** process.

The Impression fact table stores the most granular data in the database. Every page view that is recorded in a Web log corresponds to one row in the fact table. Columns consist of measures and dimension table foreign keys. (Clickstream Intelligence computes some measures, while others are derived directly from the Web log). Because measures are calculated on the session rather than on the impression level, data in the Session fact table is less granular than data in the Impression fact table. Collectively, both fact tables comprise the base structure of information that is stored in the database and used to produce business reports via Clickstream Analytics.

Figure 2–4 Staging and Fact Tables in the Clickstream Database



Populating the Dimensions

The Clickstream database is comprised of 14 pre-defined dimensions. Dimensions can be populated with data from incoming Web log files, or can be populated with dimension keys and level attributes that you load into the interface tables. To populate the dimensions with external (non-Web) data, your company's Clickstream Administrator (or System Administrator) must first load data into the interface tables. Then use the Runtime Administrator load data into the database dimensions by executing a **Load Dimensions** process.

Interface tables are organized hierarchically; interface tables within a given hierarchy are related via foreign and natural keys. The **foreign key** of the parent interface table (the table that is lower in the hierarchy) is related to the **natural key** of the interface table above it (the child).

When you run a **Load Dimensions** process,data from the interface tables is loaded into the corresponding dimension **level tables**. Relationships among level tables in a hierarchy are established with foreign and primary keys. The **foreign key** of each level table relates to the **primary key** of the level table above it.

Data from all dimension level tables, in one or more hierarchies, comprise a **dimension**. Columns in a dimension correspond to the attributes from all dimension level tables; the number of rows in a dimension corresponds to the number of rows in the lowest dimension level table. To learn more about the Clickstream Intelligence data model, see the *Oracle9i Clickstream Intelligence Data Model Reference*.

Your Role as Clickstream Intelligence Administrator

As an Oracle9*i*AS Clickstream Intelligence Administrator, you will typically perform the following tasks:

- Install the product set.
- Configure Oracle9*i*AS Clickstream Intelligence with the Runtime Administrator's Web-based interface.
- Add or update configuration information as needed.
- Load data into the database. In addition, you can:
  - refresh summaries
  - load fact and staging tables
  - resolve unknown IP addresses

- restore data to a previous version
- load dimensional data via the level interface tables

The Clickstream Administrator should also understand the design and function of the Web site(s) and other information sources that provide Web log data and support your company's decision-makers. To facilitate site configuration (and thereby report creation), you must understand what types of data fit your company's analytical needs. Moreover, you should be familiar with the contents and design of the database, as you will have access to (and control of) tables, views, and other database objects.

3

Configuring Clickstream Intelligence

This chapter provides information about configuring and validating the sites for which data is stored in the Clickstream Intelligence database. The items described below appear as links on the **Horizontal Navigation** bar when you click the **Configure** tab.

In the sections that follow, you'll learn about the steps and procedures required for:

- **Site Configuration**: configuration of individual Web sites
- **System Configuration**: definition and modification of parameters that affect all sites in the system

Site Configuration

You can configure data collection parameters for individual Web sites by selecting the **Site** link on the Horizontal Navigation bar. (When you log in to the Runtime Administrator, the **Site** link is selected by default). To configure global site parameters, see the "System Configuration" section of this chapter.

Any time you access a Clickstream Intelligence site, use the **Site Navigation** menu to configure the parameters that define the specific types of data you want to extract from Web server log files. The information you collect forms the basis of the pre-defined reports that comprise Clickstream Analytics.

The **Site Configuration** section of this chapter describes how to create, configure, and modify all options on the **Site Navigation** menu.

The Site Summary Page

The **Site Summary** page is the first page that appears upon successful login to Clickstream Intelligence. From the **Site Summary** page, you can:

- Define a new Web site
- Edit an existing site
- Delete a site.

Every site that you create appears on the **Site Summary** page.

Create a Web Site Definition

To begin using Clickstream Intelligence, you must first identify the Web site(s) from which log data is collected.

To create a site:

1. Log in to Clickstream Intelligence with your username and password.

   The **Site Summary** page appears.

2. Click **Create a Site.**

   The **Create Site** page appears.

3. Supply the following information:

   - **Name**—the name of the source Website, no longer than 100 characters in length.
   - **Short Name**—a one-word name that uniquely identifies a site. You may use up to eight alphanumeric characters, no spaces, and no single quotes.
   - **Base URL**—the URL for the site.
   - **Description**—(optional) a brief description of the site.
   - **User Group**—enables you to categorize authenticated Web site users by specific groups that are customized and defined by your company.
   - **Visitor Group**—enables you to categorize unregistered site users by specific groups that are customized and defined by your company.
   - **Time Zone**—The default value is set to GMT

4. When you have entered all required site information, click **Submit**.

   The **Site Summary** page appears with a message to confirm that the site has been created.

Delete a Site

If you want to permanently remove a site, do so from the **Site Summary** page.

To delete a Clickstream site:

1. From the **Site Summary** page, click the radio button beside the site you want to delete.
2. Click **Delete**.

   A **Delete Confirmation** appears.
3. If you are certain you want to delete the site, click **Delete**.

   The **Site Summary** page appears with a **Confirmation** message indicating that the site has been deleted.

> **Note:** Because the directory for a deleted site remains in the file system directory structure, you must manually remove the directories associated with deleted sites. If the deleted site hasn't been loaded before, then no action is required.

Edit an Existing Site

To add or modify configuration information for a site, you must first select the site on the **Site Summary** page. Once the site is selected, simply select an item on the **Site Navigation** menu that describes the type of information you want to modify.

To select and edit an existing site:

1. From the **Site Summary** page, select the radio button for the site you want to modify.
2. Click **Go**.

   The **Site Details** page displays (by default).

> **Note:** The **Site Details** page represents the first of several options on the **Site Navigation menu**. To modify information other than **Site Details**, click the appropriate option on the **Site Navigation** menu. For detailed information about each **Site Navigation** menu option, reference the appropriate sections in this chapter.

3. Review and edit the appropriate site information.
4. When you are finished, click **Submit.**

The **Site Details** page displays a **Confirmation** message to indicate that your changes are saved.

The Site Navigation Menu

The **Site Navigation menu** enables you to define and edit a site's configuration information. The menu is displayed on the left side of your screen whenever you select a site from the **Site Summary** page. The items displayed on the menu are listed below:

- **Site Details**
- **User Identification**
- **Visitor Identification**
- **Session Identification**
- **Data Sources**
- **Internal Domains**
- **Internal Sites**
- **Local Search Engines**
- **Page Parameter Rules**
- **Configuration Summary**

It is important that you know which fields are available within your log format when you define the **Site Navigation** menu items described in the following sections. If you attempt to create or update configurations that are not recognized by your log format, your site(s) will fail **Site Validation**.

Defining Site Parameters for the First Time

When you configure a **Site Navigation** menu option for the first time, you may need to start the **Configuration Wizard**. For other menu options, you are prompted to create a new parameter. Specifically:

- **User, Visitor, and Session Identification**: Click the **Start Configuration Wizard** button that appears on the main page for the menu item you select.
- **All other configurable Site Navigation menu items**: Click the **Create** button.

For detailed information about defining specific **Site Navigation** menu options, please reference the appropriate sections in this chapter.

Editing a Site's Configuration Parameters

Once you have selected a site from the **Site Summary** page, you can edit data collection specifications by selecting the appropriate item on the **Site Navigation** menu. Every time you click a **Site Navigation** menu option, its main page displays a summary of all existing definitions.

To edit the settings for a particular **Site Navigation** menu option, follow the steps below:

1. Click the **Site Navigation** menu item that corresponds to the configuration settings you want to edit.

   The **Summary** page (main page) for the menu item appears.

2. Select the definition you want to modify and click **Update**.

   The **Details** page for the definition appears.

3. Modify the appropriate information and click **Submit**.

   Your changes are saved.

> **Note:** The steps above do not apply to **User**, **Visitor**, **Session, and Page** parameters. Modification of these parameters requires reconfiguration. To do this, click the **Reconfig** button that appears below the current settings. For **Page Parameters**, click the **Reset** button.

Deleting Site Parameter Definitions

Once you have selected a site from the **Site Summary** page, you can delete certain data collection settings. To do this, first select an option on the **Site Navigation** menu. The main page for that option appears with a summary of all existing definitions. Then, choose the specific definition you want to remove.

To delete settings for an item on the **Site Navigation** menu:

1. Click the **Site Navigation** menu option for which you'd like to make a deletion.

   The **Summary** page for that option appears.

2. Click the radio button beside the definition you want to remove.

3. Click **Delete**.

   A **Delete Confirmation** appears.

4. If you are certain you want to delete the definition, click **Delete** again.

A second **Confirmation** message appears to indicate that the definition has been deleted.

Note: The steps above apply to all Site Navigation menu options *except* **User**, **Visitor**, and **Session Identification**- definitions for these options cannot be deleted. Instead, you can modify your specifications by clicking the **Reconfig** button.

Additionally, you cannot delete settings for the **Configuration Summary** menu item, since it simply displays a summary of information that is configured elsewhere. To alter the information displayed on the **Configuration Summary** page, go to the Summary page for the specific **Site Navigation** menu item.

Creating New Parameter Definitions

New definitions can be created for all **Site Navigation** menu options *except* User, Visitor, and Session Identification. Every time you select a site menu option, the **Summary** page for the item is displayed. From this page, you can create a new definition by clicking the **Create** button at the bottom of the page.

Due to the similarities in the modification and deletion of site data collection parameters, the sections that follow focus on the *creation* of definitions for all configurable **Site Navigation** menu options. The following sections provide detailed instructions and explanatory text for all **Site Navigation** menu items.

Site Details

When you select a site from the **Site Summary** page and click **Go**, the **Site Details** page displays by default. This page enables you to enter or modify descriptive site information, such as the name, description, and user / visitor groups.

The fields displayed on the **Site Details** page are the same as those displayed on the **Create a Site** page. For information about specific fields on the **Site Details** page, see "Create a Web Site Definition".

User Identification Parameters

The **User Identification** page enables you to identify the method by which Clickstream Intelligence recognizes authenticated users. **Authenticated users** are individuals who have registered with a Web site or own a valid email address that is recognized by the Web site.

For a given site, identification of registered users is a three-step process:

1. **Select a method** for identifying users. Information about each method is provided in the sections that follow.
2. **Define the parameters** associated with the identification method you selected.
3. **Save** the configuration.

Methods for identifying site users include **Web server authentication**, **query string**, or **cookie value**. To use any of these methods, you should first check that your Web log format supports identification by the method you select prior to configuring Clickstream Intelligence. Then, you must set your Web server to include the appropriate fields in the Web log. If you intend to identify users by the **Web Server Authentication** method, for example, the **Authenticated User Name** field must be included on the list of **Selected Fields** for the **Data Format** parameter, and also must be logged by your Web server. For more information, see "Define a New Data Format".

Web Server Authentication

The **WebServer Authentication** method identifies registered users by the username that is authenticated by the site's HTTP server. Following authentication by the Web server that hosts site content, this username is included in every record in the Web log file. This enables association of a session and its requests with a specific registered user of your site. You can select this method only if your Web log format supports Web server identification.

1. Select the site you want to configure from the **Site Summary** page and click **Go**.
2. Select **User Identification** from the **Site Navigation** menu.
3. From the **Select a Method for Identifying Authenticated Users** page, click the radio button beside **Web Server Authentication**.
4. Click **Next**.

   A **Confirmation** message appears to indicate that your information is saved.

User Identification by Query String

The **query string** method of user identification enables you to define the criteria for interpreting the query string portion of the URI in the clickstream Web server log. You can set Clickstream Intelligence to use any of the following query string components:

- The entire query string
- A unique value within the query string
- A positional value within the query string

To identify users by query string, see the instructions that follow. To return to a previous step, click the **Back** button.

1. Select the site you want to configure from the **Site Summary** page and click **Go**.
2. Select **User Identification** from the **Site Navigation** menu on the left side of your screen.
3. From the **Select a Method for Identifying Authenticated Users** page, select the radio button beside **Query String Parameter**.
4. Click **Next**.

   The **Define Query String Method** page appears.
5. Select one of the following three methods to define query strings:
   - **Use The Entire Query String:** there are no additional parameters to configure for this option.
   - **Use a variable in the query string:** the **Define Query String Named Parameter** page appears. On this page, enter the following information:
     - **Variable Name-** the name of the parameter that represents the query string identifier or username
     - **Variable Name Terminator-** the character or string that follows each parameter name and separates it from its corresponding value
     - **Variable Value Terminator-** the character or string that separates entire name-value pairs in the query string
   - **Use a positional value in the query string:** the **Define Query String Positional Parameter** page appears. On this page, enter the following information:
     - **Delimiter**- enter the string or character that is used to separate the positional query string parameter in the query string
     - **Position**- locate the numeric position (beginning with 0) of the query string identifier
6. When you have entered all required information for the query string option you chose above, click **Next**.

A **Confirmation** message appears to indicate that your settings are saved.

User Identification by Cookie Value

Cookies contain information related to users and their online preferences. Identifying users by cookie value enables you to track registered site users by the entire cookie string with which they are associated, or by other identifiers located within the cookie itself.

1. Select the site you want to configure from the **Site Summary** page and click **Go**.
2. Select **User Identification** from the **Site Navigation** menu on the left side of your screen.
3. From the **Select a Method for Identifying Authenticated Users** page, select the radio button beside **by Cookie Value.**
4. Click **Next**.

   The **Define Cookie Method** page appears.
5. Enter the **cookie variable name** in the field provided.
6. Select one of the following three options to describe the cookie variable name:
   - **Use the Entire Cookie Variable**: there are no additional parameters to configure for this option.
   - **Use a subvariable in the cookie variable**: the **Define Cookie Named Parameter** page appears. On this page, enter the following information:
     - **Variable Name-** the name of the parameter that identifies the cookie variable
     - **Variable Name Terminator-** the character or string that follows each variable name and separates it from its corresponding value
     - **Variable Value Terminator-** the character or string that separates entire name-value pairs in the cookie string
   - **Use a positional value in the cookie variable** : the **Define Cookie Positional Parameter** page appears. On this page, enter the following information:
     - **Delimiter-** enter the string or character that is used to separate the positional value in the cookie string
     - **Position**- locate the numeric position (beginning with 0) of the query string identifier

7. When you have entered all required information for the query string option you chose above, click **Next**.

   A **Confirmation** message appears to indicate that your settings are saved.

Visitor Identification Parameters

The **Visitor Identification** page enables you to define the way Clickstream Intelligence identifies site visitors. Visitors are defined as unregistered Web site users. Many visitors are often first-time users of the site.

For a given site, identification of visitors is a three-step process:

- **Select a method** for identifying visitors. Information about each method is provided in the sections that follow.
- **Define the parameters** associated with your chosen identification method.
- **Save** the configuration

Methods for identifying site visitors include **Client hostname or IP address only**, **query string**, **cookie value**, or **visitor field**. To use any of these methods, you should check that your Web log format supports identification by the method you select prior to configuring Clickstream Intelligence. Then, you must set your Web server to include the appropriate fields in the Web log. To identify visitors by Client Hostname or IP Address, for example, either the **Client Hostname** or the **Client IP Address** field must be included on the list of **Selected Fields** for the **Data Format** parameter, and also must be logged by your Web server. For more information, see "Define a New Data Format".

Client Hostname or IP Address

The **Client Hostname or IP Address Only** method tracks the IP address or hostname of the client that visitors use to connect to your Web site. To use this method of visitor identification, follow the steps below. If you want to return to a previous step, click the **Back** button.

1. Select the site you want to configure from the **Site Summary** page and click **Go**.
2. Select **Visitor Identification** from the **Site Navigation** menu on the left side of your screen.
3. From the **Select a Method for Identifying Unique Visitors** page, click the radio button beside **Client Hostname or IP Addess Only**.
4. Click **Next**.

A **Confirmation** message appears to indicate that your information is saved.

Visitor Identification by Query String

You can set Clickstream Intelligence to use any of the following query string components:

- The entire query string
- A uniquely named value within the query string
- A positional value within the query string

The configuration procedure for visitor identification by query string is the same as that required to identify users by query string. For detailed information on how to specify visitor query string identification settings, see the section titled "User Identification by Query String".

Visitor Identification by Cookie Value

When visitors are identified by cookie value, you must apply one of the following three options to the cookie variable that you specify:

- Use the entire cookie variable
- Use a subvariable in the cookie variable
- Use a positional value in the cookie variable

The configuration procedure for visitor identification by cookie value is the same as the steps required to identify users by cookies. For detailed instructions on how to do this, see the section titled "User Identification by Cookie Value".

Visitor Identification by Visitor Field

Follow the steps below if you want Clickstream to recognize visitors by a visitor field in the Web log. To return to a previous step, click the **Back** button.

1. Select the site you want to configure from the **Site Summary** page and click **Go**.
2. Select **Visitor Identification** from the **Site Navigation** menu on the left side of your screen.
3. From the **Select a Method for Identifying Unique Visitors** page, select the radio button beside **Visitor Field.**
4. Click **Next**.

   A **Confirmation** message appears to indicate that your information is saved.

Session Identification Parameters

The **Session Identification** page enables you to specify the way in which Clickstream Intelligence identifies unique sessions. A **session** is defined as the sequence of requests made by a user or visitor during a particular visit to a Web site. Once a user stops making Web site requests for a given period of time—referred to as a **session threshold**—the next hit to the Web site begins a new session.

For a given site, session identification is a three-step process:

- **Select a method** for identifying a session. Information about each method is provided in the sections that follow.
- **Define the parameters** associated with your chosen identification method.
- **Save** the configuration

Methods for identifying sessions include **visitor ID and timeout period**, **query string parameter**, **cookie value**, and **session field**. To use any of these methods, you should check that your Web log format supports identification by the method you select prior to configuring Clickstream Intelligence. Then, you must set your Web server to include the appropriate fields in the Web log. To identify sessions by cookie value, for example, the **Cookie String** field must be included on the list of **Selected Fields** for the **Data Format** parameter, and must also be logged by your Web server. For more information, see "Define a New Data Format"

Session Identification by Visitor ID and Timeout Period

To identify sessions by visitor and a pre-determined timeout period, follow the steps below. If you want to return to a previous step, click the **Back** button.

1. Select the site you want to configure from the **Site Summary** page and click **Go**.
2. Select **Session Identification** from the **Site Navigation menu** on the left side of your screen.
3. From the **Select Session Construction Method** page, click the radio button beside **By Visitor ID and Timeout Period**.
4. Click **Next**.

   The **Define Session Timeout Period** page appears.
5. Enter a value into the **Maximum Period of Inactivity** field. This number represents the length of time, in minutes, before a visitor is timed out from the Web site.

6. Click **Next**.

   A **Confirmation** message appears to indicate that your information is saved.

Session Identification by Query String

You can set Clickstream Intelligence to identify sessions by query string in three different configurations:

- By using the entire query string
- By using a uniquely named value
- By using a positional value within the query string

The configuration procedure for session identification by query string is the same as that required to identify users and visitors by query string. For detailed information on how to set up session query string identification parameters, see the section titled "User Identification by Query String".

Session Identification by Cookie Value

To identify sessions by cookie value, choose one of the following three options for the cookie variable that you specify:

- Use the entire cookie variable
- Use a subvariable in the cookie variable
- Using a positional value in the cookie variable

The configuration procedure for session identification by cookie value is the same as that required to identify users and visitors by cookies. For detailed instructions on how to do this, see the "User Identification by Cookie Value" section of this chapter.

Session Identification by Session Field

To use the session field in the Web log to identify distinct sessions, follow the steps below. If you want to return to a previous step, click the **Back** button.

1. Select the site you want to configure from the **Site Summary** page and click **Go**.
2. Select **Session Identification** from the **Site Navigation** menu on the left side of your screen.
3. From the **Select Session Construction Method** page, select the radio button beside **Session Field.**

4. Click **Next**.

   A **Confirmation** message appears to indicate that your information is saved.

Data Sources

The **Data sources** parameter designates the location of the Web server from which clickstream data is obtained. You can utilitze Clickstream's **Collector Agent** to automatically retrieve Web logs from a remote host via HTTP or FTP protocol, or you can specify a local directory on which data is located (called a **local data source**).

For each designated data source, you must define one **data format**. If you want to add new formats to the list of options displayed on the **Create Data Source** page, go to the **System Configuration** tab and click the **Data Formats** option on the **System Navigation** menu.

You may also place **data filters** on your defined data source. Filters "pre-process" Web data and enable you to define exactly the type of information you want to extract from log files. If no filter options are available for selection on the **Create Data Source**s page, you must first define one or more data filters via the **System Navigation** menu. To learn more about defining data filters for your system, see the "Data Filters" section of this chapter.

The **Data Source Summary** page enables you to **view**, **create**, **update**, and **delete** a site's data sources. The sections below describe how to view and create new data sources. For information about editing and deleting data sources, see the general instructions in the following sections of this chapter:

- **Editing a Site's Configuration Parameters**
- **Deleting Site Parameter Definitions**

View All Data Sources

To view all data existing sources, click the **View All Data Sources** button at the bottom of the **Data Source Summary** page.

The **All Data Sources** page displays, in table format, the data sources that have been defined for all sites in the system. Specific information - such as the name, short name, data format, type, and the site to which the data source belongs - is displayed for each data source listed in the table.

Create a Data Source

To specify a new data source for a site, follow the steps below. Fields accompanied by an asterisk (*) are required.

1. Select **Data Sources** from the **Site Navigation** menu.

   The **Data Source Summary** page appears.

2. Click **Create Data Source**.

   The **Create Data Source** page appears.

3. Supply the following descriptors for the new data source:

   - **Name**: a unique name that identifies the data source
   - **Short name**: a one-word name for the data source
   - **Description**: (optional)
   - **Protocol**: Clickstream supports HTTP and FTP; you can also designate a local source.

4. Provide source **Definition** information. The fields displayed are dependent upon your protocol selection: local, HTTP, or FTP. Any of the following fields may be displayed:

   - **Host:** (HTTP and FTP) The computer from which a site's Web log data originates
   - **Port:** (HTTP and FTP) Enter a positive integer for non-standard ports only. Default values are set to: HTTP=80 and FTP=21.
   - **Username:** (HTTP and FTP) The name used to connect to the data source machine. This field is required for FTP.
   - **Password:** (HTTP and FTP) The password that corresponds to the username which enables access to the data source machine.
   - **Path:** (all) Indicates the location of your Web log files; typically the portion of a URL following the host:port
   - **Data Format:** (all) Select a format from the pull-down menu, or **Define a New Data Format**
   - **Time Zone:** (all) The time zone in which dates are recorded by in the log files. This parameter enables Clickstream Intelligence to adjust the log file time zone to that of the Web site. This field is optional and has no effect on Apache formats.

- **Default Server Name:** (all) Enables you to specify the hostname of the server, for situations in which the specified data format does not include a "Server Name" field. This field is optional.

5. If you want to use **filters** to pre-process data retreived from the data source, follow the steps below. Filters are optional.

   - Go to the **Filters** section of the **Create Data Source** page and select the filter(s) you want from the **Available Filters** window.

     (If no filters are available for selection, you must first Create a Data Filter.)

   - Use the **Move** arrows to move the filters to the **Selected Filters** window.

6. When you are finished entering data source information, click **Submit**.

   A **Confirmation** message appears and the new data source appears on the **Data Source Summary** page.

Internal Domains and Internal Sites

Internal domains and internal sites are used by Clickstream Intelligence to classify and categorize Web server log data. An **internal domain** is typically a domain name that resides within your organization, or a domain name that you want to classify as "internal" for reporting purposes. Similarly, an **internal site** is a site that is internal to your organization and located somewhere within the realm of your Web site. Both can be used to further classify and characterize log data that is utilized in the creation of analytical reports.

For example, when you specify one or more **internal domain names** for a particular site, you enable Clickstream to distinguish "internal domains" from all other domains that appear in the site's log files. When you run a **Top Referrers by Internal Domain** report, Clickstream views the site's loaded server log files and creates a report by analyzing data pertaining to internal domains only.

Configuration of **Internal Domain** and **Internal Site** parameters is optional because neither influences how a site's log data is transformed when it is loaded into the database. Configuration of both parameters is recommended, however, to ensure the generation of accurate report results. For example, if you do not specify at least one internal domain, all domains retrieved from your log files are considered to be external. This misrepresentation of log data may ultimately affect analytical report results.

The following section describes how to define a new internal domain/site. For specific information about editing and deleting internal domains and internal sites, see the general instructions at the beginning of this chapter:

- **Editing a Site's Configuration Parameters**
- **Deleting Site Parameter Definitions**.

Define a New Internal Domain or Internal Site

The following steps describe the specification of an **internal domain** name for a given Web site. To set up an **internal site**, follow the same procedure, but select the **Internal Sites** option from the **Site Navigation** menu.

1. Go to the **Site Summary** page and select the site for which you want to specify a new internal domain.
2. Click **Go**.
3. Select **Internal Domains** from the **Site Navigation** menu on the left side of your screen

   The **Internal Domains** page appears.
4. Click **Create Internal Domain**.

   The **Create Internal Domain** page appears.
5. Enter the internal domain name in the space provided.
6. Click **Submit**.

   A **Confirmation** message appears to indicate that your information is saved.

Local Search Engines

A **local search engine** is one that enables customers to search within your Web site for pages containing specific keywords. The local search engine parameter influences how log data from an internal search application is transformed when loaded into the database. Specifically, it determines the way that search phrases are parsed from the Web log and loaded into the **Search dimension**. If a local search engine is not defined for a site, certain data cannot be loaded into the Search dimension and reports, such as the **Top Local Search Phrases** report, cannot be created.

The section that follows details the steps required to define a new internal search engine. For specific information about editing and deleting local search engines, see the general instructions in the following sections of this chapter:

- **Editing a Site's Configuration Parameters**
- **Deleting Site Parameter Definitions**.

Define a New Local Search Engine

1. Go to the **Site Summary** page and select the site for which you want to specify a local search engine.
2. Click **Go**.
3. Select **Local Search Engines** from the **Site Navigation** menu.

   The **Local Search Engines** main page appears.
4. Click the **Create Local Search Engine** button.

   The **Create Local Search Engine** page appears.
5. Provide the following information
   - **URI Stem**- the portion of a URL that follows the host and port, but precedes the query string. The URI stem indicates the actual path to the program that returns your search results.
   - **Parameter**- (optional) the name of the query string parameter that identifies the search expression. The **parameter** typically follows the URI stem and immediately precedes the search keyword; it may be a word, number, symbol, or any combination of these elements.
6. Click **Submit**.

   A **Confirmation** message appears to indicate that your information is saved.

Page Parameter Rules

Page parameters refer to the set of query string parameters in a URL (called **name-value pairs**) that indicate unique page content. Clickstream Intelligence identifies new pages by combining the stem, or file name, of the URI with the names and values of all page parameters that are configured for that particular stem.

As an example, consider the following Web log entries that represent multiple hits to a particular file or URI stem:

www.yoursite.com / pageXYZ.html?action=create

www.yoursite.com / pageXYZ.html?action=update

www.yoursite.com / pageXYZ.html?action=delete

Although each hit refers to the same **stem** (pageXYZ.html), the actual page content returned by the server varies with the parameter values contained in the query

string of the request. In the example above, **action** is a parameter with values equal to "create," "update," and "delete." Depending on parameter value, the server returns different content for the file "pageXYZ.html." This and similar analyses of query string parameters may enable you to determine which pages customers access on your site.

While some parameters contribute to the identity and content of the requested page, other parameters in the query string do not affect which page is returned to the user. In order to accurately identify unique pages requested from your site, these **non-page parameters** should be excluded from consideration. For example, a shopping cart identifier or session identifier in the query string has no influence upon the page content displayed to the user.

Clickstream Intelligence enables you to define page parameter rules on two levels:

- **Global Page Parameter Rules**
- **Individual Page Parameter Rules**

At both the **Global** and **Individual** levels, you can specify the query string parameters that are to be considered **page parameters** by selecting one of the following options:

- **All -** every parameter in the query string is relevant when identifying page content. Use this option when all query string parameters are necessary to determine page content.
- **None -** no parameters in the query string are considered when determining page content. Typically, this option is used when the URI stem itself is indicative of page content served by your Web site.
- **Include Following -** enables you to specify only parameters that identify unique page content. When you select **include** from the pull-down menu, only the parameters you list are considered; no other query string parameters are deemed relevant. Use this option when it is easier to list parameters that contribute to page content identification than it is to list all others that do not.
- **Exclude Following -** enables you to exclude from analysis parameters that are *not* used to identify unique page content, such as non-page parameters. When you select **exclude** from the pull-down menu, all query string parameters are considered when identifying page content *except* those you specify. Use this option when it is easier to list the exceptions rather than all parameters that do contribute to page content identification.

Global Page Parameter Rules

When you define **global page parameters**, you indicate which query string parameters are relevant when identifying page content for *all* files or URI stems on your Web site. Global parameters apply only to the pages for which individual page parameter rules are *not* defined.

1. Go to the **Site Summary** page and select the site for which you want to configure page parameter ruless.
2. Click **Go**.
3. Select **Page Parameter Rules** from the **Site Navigation** menu.

   The **Global Page Parameter Rules** page appears.
4. Select one of the following options from the pull-down menu:
   - **All**
   - **None**
   - **Include Following**- the screen refreshes and a parameter field appears. Define a parameter to include, and click **Add Parameter**. Repeat this process to specify additional parameters.
   - **Exclude Following**- the screen refreshes and a parameter field appears. Define a parameter to exclude, and click **Add Parameter**. Repeat this process to specify additional parameters.
5. Click **Submit**.

   A **Confirmation** message appears to indicate that your information is saved.

Individual Page Parameter Rules

Individual page parameters apply to a specific file or URI stem that belongs to your site. Parameters specified on an individual (file-specific) level *override* any settings that may be defined on the Global Parameter Rules page for your Web site.

1. Go to the bottom of the **Global Page Paramater Rules** page and click the **Create Individual Page Parameter Rule** button.

   The **Individual Page Parameter Rule** page appears.
2. Provide the following information:
   - **URI Stem:** the portion of a URL that follows the host and port, but precedes the query string.

- **Parameters:** specify **All**, **None**, **Include Following**, or **Exclude Following**. To include or exclude parameters, you must enter a **Parameter Name** in the field provided and click **Add Parameter**. (To include or exclude additional parameters, repeat this step.)

3. Click **Submit**.

   A **Confirmation** message appears to indicate that your information is saved.

Configuration Summary

Once you have configured Clickstream Intelligence, you can see a summary of your settings by selecting **Configuration Summary** from the **Site Navigation** menu.

Settings for all **Site Navigation** menu items are displayed on the **Configuration Summary** page. If you have not configured particular menu options or need to modify existing information, simply click the appropriate item on **Site Navigation** menu and edit as necessary. Otherwise, you can **validate** your site to see if the configuration is valid.

Site Validation

Before you can load data into the Clickstream database, a site's configuration must be validated. **Validation** is the process by which site parameters are checked for proper configuration.

The validation step checks for the following:

- **Proper configuration of user, visitor, and session identifications.** If you choose to identify users by cookie, for example, then the cookie field must be present in the data format of all the data sources for the site.
- **The existence and validity of one or more data sources.** In order for a data source to be valid, its data format and data filter parameters must also be valid. A filter can only act on a field that is present in the data format selected for the **Data Sources** parameter.

To validate a site, click the **Validate** button at the bottom of the **Configuration Summary** page. If one or more validation processes are unsuccessful, an error screen appears to indicate what has failed. To correct the faulty setting(s), you must return to the page containing the error.

When you have correctly configured Clickstream Intelligence, the **Validate Site Configuration** page displays a **Confirmation** message to indicate that your site configuration is **Valid**. Now your site's data can be loaded into the Clickstream

warehouse. For more information on populating the database, see *Chapter 4, "Loading the Clickstream Database"*.

Note: Each time you alter a site's configuration parameters, you need to re-validate the site from the **Configuration Summary** page.

System Configuration

System configuration parameters are used to effect changes in some or all of the sites listed on your **Site Summary** page. To access these configuration parameters, click the **System** link located on the left side of the Horizontal Navigation bar at the top of your screen. The **System Cofiguration** section of this chapter describes how to set up all items displayed on the **System Navigation** menu.

The System Navigation Menu

All system configuration parameters can be accessed from the **System Navigation** menu that is located along the left side of your screen. You can select menu options to do any of the following:

- Define the options displayed on pull-down menus when certain site parameters are defined (such as **Data Formats** for individual sites).
- Create global mechanisms that facilitate data loading (such as **Data Filters**).
- Define the **Resource Types** available for each site.
- Define site parameters that affect your sites globally (such as **Referring Search Engines**).
- Define **System Parameters.**
- Define **Process Definitions**.

Configuration of some System Navigation menu items is optional. Defining additional **resource types**, for example, is not essential for site validation. Similarly, when incoming Web data has already been filtered, there is no need to configure settings for **data filters**. On the other hand, if your log format does not correspond to one of the pre-defined types, you need to specify one or more **data formats**.

Data Formats

When you click the **System** tab, the **Data Format Summary** page appears by default. It lists all data formats defined for the system. These data formats correspond to the options from which you may choose when setting up a site's Data Sources. For more information about data sources, see "Create a Data Source".

For any given **data format**, you must specify the log file **fields** that appear in the Clickstream log. Fields that appear in a given data format's log file must be listed in the **Selected Fields** window; data fields not present in the log should be listed in the **Available Fields** window. All **Selected Fields** must be listed in the order that they appear in the Web log file.

The **Data Format Summary** page displays all log formats defined for your system. From this page, you may create, edit, and delete data formats.

- **To define a new data format**: see "Define a New Data Format" below.
- **To edit a data format**: select the radio button beside the data format and click **Update**.
- **To delete a data format**: select the radio button for the data format you want to remove and click **Delete**. Deletion of a data format results in its removal as an available **Data Format option** in the **Data Sources** settings for *all* your sites.

Define a New Data Format

Data formats are defined from the **Create Data Format** page. For each data format, you must indicate which log file **Fields** are logged by the Web server. To identify the fields that appear in the clickstream log, list them in the **Selected Fields** window by clicking the **arrow** buttons that move fields from the **Available Fields** to the **Selected Fields** window. You must also specify the order that the fields appear in the log file for the format you're defining. To do this, use the **arrow** buttons embedded in the **Selected fields** window.

If you plan to identify registered users by the **Web Server Authentication** method, for example, the **Authenticated Username** field must be listed in the **Selected Fields** window for the appropriate **data format**. By doing this, you indicate that **Authenticated Username** is a field that is logged by the Web server, and therefore appears in the log file for the particular data format.

To define a new data format:

1. Select **Data Formats** from the **System Navigation** menu.

   The **Data Format Summary** page appears.

2. Click the **Create a Data Format** button.

   The **Create Data Format** page appears.

3. Enter the following information:

   - **Type:** select from Apache, W3C Extended Log File Format, or IIS Extended Log File Format. The data format you select determines the options that are displayed in the **Available Fields** window (see Step 4).
   - **Name**: the name that identifies the data format.
   - **Short Name**: a unique name that describes the data format. You may use up to eight alphanumeric characters.
   - **Description**: (optional) a brief description of the data format.

4. Select the **Fields** that describe the type of clickstream data contained in the log files with your defined data format type. You must select *all* fields that appear in your log file, even if you do not intend to collect data all of the selected fields.

   - **Selected Fields:** fields that appear in the Web server log files.
   - **Available Fields:** fields that represent Web log data types that do not appear in the Web log.

   To select a field, click on the field in the **Available Fields** window and click the **right arrow** button to move it to your **Selected Fields** window.

5. When you have entered all required data format settings, click **Submit**.

   A **Confirmation** message appears to indicate that your information is saved.

Note: The ten **General Purpose Fields** listed in the **Available Fields** window are generic fields that can be defined and customized according to your particular data format.

Data Filters

Clickstream Intelligence uses **data filters** to identify the characteristics of the data retrieved from Web server log files. Each time you create or modify a data source for a site, you can select one or more filters to refine the data collected. (For more information about data sources, see "Create a Data Source".)

The **Data Filter Summary** page enables you to **create**, **update**, and **delete** data filters.

- **To define a new data filter**: see "Create a Data Filter" below.
- **To edit a data filter**: select the filter you want to modify and click **Update**.
- **To delete a data filter**: select the filter you want to remove and click **Delete**.

All data filters listed on the **Data Filter Summary** page can be used by any site in your system when you configure the **Data Source** option on the Site Navigation menu. Specifically, each data filter that you create from the **Create Data Filter** page becomes an **Available Filter** for the **Data Sources** parameter. When you delete a data filter from the **Data Filter Summary** page, it is removed from the **Available Filters** section on the Data Sources form for all sites.

Create a Data Filter

The following instructions describe the steps required to define a data filter. Each time a data filter is created, you can specify **conditions** for the data it acts upon. Each time you select a type of condition, you must enter a specific value and then designate how that value relates to, or **operates** on, the filter you're defining. All data filters must contain at least one condition.

1. Select **Data Filters** from the **System Navigation** menu.

   The **Data Filter Summary** page displays.

2. Click the **Create Data Filter** button.

   The **Create Data Filter** page appears.

3. Enter the following information:

   - **Filter Name:** (required) specify a name that identifies the filter.
   - **Description**: (optional) a brief description of the filter.

4. Create the **Filter Definition**. To do this, you must define one or more **conditions**:

   - **Select a Data Field** from the pull-down menu that describes the type of condition you are defining, and click **Add Condition**. The data field you choose determines which operators you can select.
   - **Select an Operator** from the pull-down menu that indicates how your defined value is handled.

     For example, you can set your filter to contain, equal, begin, or end with your defined value. Time and date fields display different operators, as described above.

- **Enter a value** in the field provided that defines the word, number, or symbol that appears in your specified data field.

  For some fields, you must enter values in a specific format. **Request Date**, for example, enables you to select a date via the **Date Picker** (click the calendar icon). For the **Request Time** field, you must enter a time in HH:MM:SS format

5. (Optional) Click one of the following radio buttons below the condition list:
   - **Match all of the conditions**- the data filter acts upon log data only when all of your defined conditions are met
   - **Match any of the conditions** -the data filter acts upon log data when any of your conditions are met
6. When you are finished defining the data filter, click **Submit**.

   A **Confirmation** message appears to indicate that your information is saved.

Resource Types

Resource types are the file extensions that characterize data received from a Web server. A server provides many types of resources to clients, and associates each kind of resource with a unique extension. This extension defines the resource type. For example, text documents are often categorized by the resource type .txt, while images may be identified by the extensions .gif, .jpg, or .png. You can configure Clickstream Intelligence to recognize one or more resource types. Many are pre-defined and set to default values upon installation.

The **Resource Type** summary page displays all exising resource types for your system. From this page, you may create, edit, and delete the resource types you want to track:

- **To create a new resource type**: see "Create a Resource Type" below.
- **To edit a resource type:** select the radio button beside the file extension and click **Update**. You should check the default values for all pre-defined resource types to make sure the settings are appropriate.
- **To delete a resource type:** select the radio button beside the resource extension and click **Delete**. Deletion of resource types is not recommended.

Typically, the **Resource Type** page is used for addition and modification of file extension definitions. While deleting a resource type removes its definition from Clickstream, it does not remove the resource from which it originates. When

resource types sent from the server correspond to deleted extensions, Clickstream Intelligence assumes the following specific default values:

- **Description**: Hypertext Markup File
- **MIME Type**: Text / HTML
- **Delivery Method**: Static
- **Identifies Page**: Yes (unknown extensions will be used to identify the page)
- **Type**: Content

Create a Resource Type

The following procedure describes how to define a resource type from the **Create Resource Type** page.

1. Select **Resource Types** from the **System Navigation** menu.

   The **Resource Type Summary** page displays.

2. Click **Create Resource Type**.

   The **Create Resource Type** page appears.

3. Enter the following information:

   - **Name**: the letter combination that represents the file extension
   - **Description**: (optional) characterizes and describes the resource type
   - **MIME Type**: (optional)
   - **Delivery Method**: select dynamic (such as personalized content on an HTML page) or static (such as an image file)
   - **Identifies Page**: (Yes / No)
   - **Type**: use the pull-down menu to select the best descriptor of your resource; such as *audio* for a music file, *image* for a picture, and so on.

4. When you are finished entering information, click **Submit**. (Or to clear all fields and start over, click **Reset**.)

   A **Confirmation** message displays on the **Resource Type Summary** page to indicate that your entry is saved.

Referring Search Engines

A referring search engine is used to query the Internet. Unlike a local search engine, which is internal to your company, a referring search engine is external to your company and not affiliated with your business. When analyzing log data, information related to referring search engines may indicate key areas of customer interest, common keywords entered by Internet users, or the search engines from which site visitors arrive.

The **Referring Search Engines** summary page displays all external search engines that have been defined for your system. From this page, you may create, edit, and delete referring search engines.

- **To define a new referring search engine**: see "Create a Referring Search Engine" below.
- **To edit a referring search engine**: select the radio button beside the search engine and click **Update**.
- **To delete a referring search engine:** select the radio button for the search engine you want to remove and click **Delete**.

Create a Referring Search Engine

To define a new referring search engine for your system, follow the instructions below.

1. Select the **Referring Search Engines** option from the **System Navigation** menu.

   The **Referring Search Engines summary** page displays.

2. Click the **Create Referring Search Engine** button.

   The **Create Referring Search Engine** page appears.

3. Enter the following information:

   - **Host**: the name of the search engine (Use the following format: www.typesearchenginehere.com)
   - **URI Stem**: the portion of the URL that follows the host and port, but precedes the query string. The URI stem indicates the actual path to the search program that provides the search results.
   - **Parameter**- (optional) the name of the query string parameter that identifies the search expression. The **parameter** typically follows the URI stem and immediately precedes the search keyword; it may be a word, number, symbol, or any combination of these elements.

4. When you are finished entering the information above, click **Submit**.

   A **Confirmation** message appears to indicate that your entry is saved.

System Parameters

System parameters identify the directory in which Clickstream Intelligence is installed, the directory to which Clickstream Intelligence writes log files, and the directory from which remote data source log files are collected. These system-level properties are stored in the `click-app.xml` file and can be modified or updated online from the **System Parameters** page.

All parameters displayed on the **System Parameters** main page must be defined.

- **Log Directory**: the directory path that indicates where Clickstream Intelligence puts the logs that it writes, such as the Execution Engine log and the Clickstream Loader logs.
- **Load Directory**: the directory path for the location to which Web log files are transferred prior to a load, and the location from which files are loaded during the load process.
- **Clickstream Home**: the path to the directory in which Clickstream Intelligence is installed.

Process Definitions

Process definitions define the criteria that Clickstream Intelligence uses to perform specific tasks, or **processes**, related to the database. When clickstream data is loaded into the database, for example, you perform a specific type of process called "Load Clickstream." The parameters that specify how to perform this process comprise each **process definition**.

The Process Definitions page enables you to do the following:

- **Create a new process definition**: see "Create a Process Definition" below.
- **Edit a process definition**: select the radio button beside a process definition and click **Update**.
- **Delete a process definition**: select the radio button for the process definition you want to remove and click **Delete**.

Every time you create a new definition, you must provide a unique name and specify the process type. After a type is selected, the **Create Process Definition** page refreshes and fields specific to the process type are displayed. You can create

definitions for five process types. To learn more about the specific parameters associated with each type of process, see the one of the following sections in this chapter:

- **Load Clickstream**: loads clickstream data into the database.
- **Load Dimensions**: loads database dimension data into the warehouse.
- **Refresh Summaries**: refreshes the database summary view.
- **Resolve IP Addresses**: performs a reverse DNS lookup to obtain host names from the numerical IP addresses.
- **Restore a Previous Version**: sets the data stored in the database back to values that were set at a specific (previous) point in time.

Create a Process Definition

To create a process definition from the **Process Definitions** page, select the process type for which you are creating a definition. Then, enter information for the specific parameters that apply to your definition type.

1. Select **Process Definitions** from the **System Navigation** menu.

   The **Create Process Definition** page displays.

2. Provide a **name** for the process definition you are creating.
3. Enter a **short name** (8 alphanumeric characters or less) to identify the definition.
4. Select the **type** of process you want to define. Use the pull-down menu to select one of the following process types:
   - **Load Clickstream**
   - **Load Dimensions**
   - **Refresh Summaries**
   - **Resolve IP Addresses**
   - **Restore a Previous Version**

   The page refreshes and parameters specific to your process type appear.
5. Provide information in all required fields for your chosen process type.
6. Click **Submit**.

   A **Confirmation** message appears on the **Process Definitions** page to indicate that your settings are saved.

Load Clickstream

The **Load Clickstream** process enables the transfer and storage of data into the database. The following descriptions correspond to the fields for which you must enter information when creating a definition for the **Load Clickstream** process.

- **Parallel Factor**: the maximum number of jobs that should be executed simultaneously by the Clickstream Intelligence Execution Engine.
- **Loader Parallel Factor**: the maximum number of data packets that the Clickstream Intelligence Loader should simultaneously load into the database.
- **Loader Message Level**: controls the amount of detail logged by the Clickstream Loader.
  - Log Errors Only - Write only error messages to the log files.
  - Log Errors and Warnings - Write both error and warning messages to the log files.
  - Log All Messages - Write errors, warnings, and filter debugging information to the log files.
- **Write Bad Records**: indicates if rejected records will be written to a .bad file. When enabled (select **Yes**), rejected records are written to a file (*.bad) under the Clickstream Log Directory.
- **Write Discarded Records**: indicates if filtered records should be written to a discard file. When enabled (select **Yes**), filtered records are written to a file (*.dis) under the Clickstream Log Directory.
- **Analyze Levels**: indicates whether to gather statistics for Clickstream Intelligence level tables (CLK_L_%.) When enabled, statistics for the level tables are gathered each time the process is executed.
- **Analyze Dimensions**: indicates whether to gather statistics for Clickstream Intelligence dimension tables (CLK_AGENT, CLK_CLIENT_HOST, etc). When enabled, statistics for the dimension tables are gathered each time the process is executed.
- **Analyze Facts**: indicates whether to gather statistics for Clickstream Intelligence fact tables (CLK_IMPRESSION_FACT, CLK_SESSION_FACT). When enabled, statistics for the fact tables are gathered each time the process is executed.
- **Analyze Summaries**: indicates whether to gather statistics for Clickstream Intelligence Summary Layer objects (CLK_SL_%). When enabled, statistics

for the Summary Layer objects are gathered each time the process is executed.

- **Percent to Analyze**: the percentage of rows (from 0.000001 to 99.99999) to sample when gathering table statistics. These statistics are used by the Oracle Cost-Based Optimizer to choose better execution plans for queries.
- **Analysis Granularity**: the granularity of statistics to collect (only pertains to partitioned tables).
  - Global, Partition, and Subpartition: Gather all (subpartition, partition, and global) statistics.
  - Global and Partition: Gather global and partition-level statistics only.
  - Global: Gather global statistics only.
  - Partition: Gather partition-level statistics only.
  - Subpartition: Gather subpartition-level statistics only.
- **Execute DNS Lookup**: when enabled (select **Yes**), Clickstream Intelligence attempts to resolve the hostname of all new client IP addresses by querying the server specified in the DNS Server parameter.
- **DNS Server**: the hostname or IP address of the server to use when performing reverse DNS lookups.
- **Parallel DNS Factor**: the maximum number of concurrent DNS queries allowed by the Clickstream Intelligence DNS Lookup job. This number should be relatively high (~1000), depending on the machine and number of CPUs. With multiple CPUs, select a high number; with a single CPU, leave as default.
- **DNS Query Timeout in Milliseconds**: the maximum time, in milliseconds, to wait for a response from the DNS server.
- **Refresh Summaries**: specifies whether to refresh tables and materialized views registered with the Clickstream Intelligence Summary Layer. When enabled, the summary layer is refreshed at the end of this process.
- **Summary Refresh Type**: indicates the type of refresh to use when refreshing the Clickstream Summary Layer.
  - Default: Incrementally refresh the summary layer. If this is not possible, then perform a complete refresh. This option is often the fastest and is the recommended setting.
  - Complete: Truncate all tables and resummarize all data.

  - Incremental: Add all new data to existing summary data.

- **Summary Refresh Time Period**: specifies which summaries to refresh.
  - Year: Refresh all summaries.
  - Quarter: Only refresh summaries that include data for quarterly, monthly, and daily reports.
  - Month: Only refresh summaries that include data for monthly and daily reports.
  - Day: Only refresh summaries that include data for daily reports.
- **Log Detailed Summary Messages**: specifies whether or not to count the number of rows before and after a summary object has been refreshed.
- **Truncate Staging Tables**: indicates if the Clickstream Intelligence staging tables should be truncated at the end of the load. When enabled, the tables are automatically truncated once the data is no longer needed.

Load Dimensions

The **Load Dimensions** process enables the loading of new data into the Clickstream dimensions via the interface tables. The following descriptions correspond to the fields for which you must enter information when creating a definition for the **Load Dimensions** process.

- **Parallel Factor**: the maximum number of jobs that should be executed simultaneously by the Clickstream Intelligence Execution Engine.
- **Analyze Levels**: indicates whether to gather statistics for Clickstream Intelligence level tables (CLK_L_%.) When enabled, statistics for the level tables are gathered each time the process is executed.
- **Analyze Dimensions**: indicates whether to gather statistics for Clickstream Intelligence dimension tables (CLK_AGENT, CLK_CLIENT_HOST, etc). When enabled, statistics for the dimension tables are gathered each time the process is executed.
- **Analyze Summaries**: indicates whether to gather statistics for Clickstream Intelligence Summary Layer objects (CLK_SL_%). When enabled, statistics for the Summary Layer objects are gathered each time the process is executed.

- **Percent to Analyze**: the percentage of rows (from 0.000001 to 99.99999) to sample when gathering table statistics. These statistics are used by the Oracle Cost-Based Optimizer to choose better execution plans for queries.
- **Analysis Granularity**: the granularity of statistics to collect (only pertains to partitioned tables).
  - Global, Partition, and Subpartition: Gather all (subpartition, partition, and global) statistics.
  - Global and Partition: Gather global and partition-level statistics only.
  - Global: Gather global statistics only.
  - Partition: Gather partition-level statistics only.
  - Subpartition: Gather subpartition-level statistics only.
- **Refresh Summaries**: specifies whether to refresh tables and materialized views registered with the Clickstream Intelligence Summary Layer. When enabled, the summary layer is refreshed at the end of this process.
- **Summary Refresh Type**: indicates the type of refresh to use when refreshing the Clickstream Summary Layer.
  - Default: Incrementally refresh the summary layer. If this is not possible, then perform a complete refresh. This option is often the fastest and is the recommended setting.
  - Complete: Truncate all tables and resummarize all data.
  - Incremental: Add all new data to existing summary data.
- **Summary Refresh Time Period**: specifies which summaries to refresh.
  - Year: Refresh all summaries.
  - Quarter: Only refresh summaries that include data for quarterly, monthly, and daily reports.
  - Month: Only refresh summaries that include data for monthly and daily reports.
  - Day: Only refresh summaries that include data for daily reports.
- **Log Detailed Summary Messages**: specifies whether or not to count the number of rows before and after a summary object has been refreshed.

Refresh Summaries

The **Refresh Summaries** process enables you to pre-calculate and aggregate data from the facts and dimensions, and then store the summarized data in a set of tables that are pre-defined by Clickstream Intelligence. This data is used to enhance performance and is required to keep Clickstream Analytics updated with the most current data.

The following descriptions correspond to the fields for which you must enter information when creating a definition for the **Refresh Summaries** process.

- **Parallel Factor**: the maximum number of jobs that should be executed simultaneously by the Clickstream Intelligence Execution Engine.
- **Analyze Summaries**: indicates whether to gather statistics for Clickstream Intelligence Summary Layer objects (CLK_SL_%). When enabled, statistics for the Summary Layer objects are gathered each time the process is executed.
- **Percent to Analyze**: the percentage of rows (from 0.000001 to 99.99999) to sample when gathering table statistics. These statistics are used by the Oracle Cost-Based Optimizer to choose better execution plans for queries.
- **Analysis Granularity**: the granularity of statistics to collect (only pertains to partitioned tables).
  - Global, Partition, and Subpartition: Gather all (subpartition, partition, and global) statistics.
  - Global and Partition: Gather global and partition-level statistics only.
  - Global: Gather global statistics only.
  - Partition: Gather partition-level statistics only.
  - Subpartition: Gather subpartition-level statistics only.
- **Summary Refresh Type**: indicates the type of refresh to use when refreshing the Clickstream Summary Layer.
  - Default: Incrementally refresh the summary layer. If this is not possible, then perform a complete refresh. This option is often the fastest and is the recommended setting.
  - Complete: Truncate all tables and resummarize all data.
  - Incremental: Add all new data to existing summary data.
- **Summary Refresh Time Period**: specifies which summaries to refresh.

  - Year: Refresh all summaries.
  - Quarter: Only refresh summaries that include data for quarterly, monthly, and daily reports.
  - Month: Only refresh summaries that include data for monthly and daily reports.
  - Day: Only refresh summaries that include data for daily reports.

- **Log Detailed Summary Messages**: specifies whether or not to count the number of rows before and after a summary object has been refreshed.
- **Truncate Interface Tables**: indicates if the level interface tables should be truncated at the end of the load. When enabled (select **Yes**), the tables are truncated at the end of the process.

Resolve IP Addresses

The **Resolve IP Addresses** process enables you to perform reverse DNS lookups for IP addresses that are loaded into the Clickstream Intelligence database. Once the reverse DNS lookup is complete, Clickstream Intelligence can use the user-friendly hostnames to determine the demographics for particular IP addresses.

The following descriptions correspond to the fields for which you must enter information when creating a definition for the **Resolve IP Addresses** process.

- **Parallel Factor**: the maximum number of jobs that should be executed simultaneously by the Clickstream Intelligence Execution Engine.
- **DNS Server**: the hostname or IP address of the server to use when performing DNS lookups.
- **Parallel DNS Factor**: the maximum number of concurrent DNS queries allowed by the Clickstream Intelligence DNS Lookup job.
- **DNS Query Timeout in Milliseconds**: the maximum time, in milliseconds, to wait for a response from the DNS server.
- **Resolve Only Unresolved IP Addresses**: indicates whether to resolve only IP addresses that have not been already resolved, or resolve all IP addresses including previously resolved addresses.
- **Refresh Summaries**: specifies whether to refresh tables and materialized views registered with the Clickstream Intelligence Summary Layer. When enabled, the summary layer is refreshed at the end of this process.

- **Summary Refresh Type**: indicates the type of refresh to use when refreshing the Clickstream Summary Layer.
  - Default: Incrementally refresh the summary layer. If this is not possible, then perform a complete refresh. This option is often the fastest and is the recommended setting.
  - Complete: Truncate all tables and resummarize all data.
  - Incremental: Add all new data to existing summary data.
- **Summary Refresh Time Period**: specifies which summaries to refresh.
  - Year: Refresh all summaries.
  - Quarter: Only refresh summaries that include data for quarterly, monthly, and daily reports.
  - Month: Only refresh summaries that include data for monthly and daily reports.
  - Day: Only refresh summaries that include data for daily reports.
- **Log Detailed Summary Messages**: specifies whether or not to count the number of rows before and after a summary object has been refreshed.

Restore a Previous Version

The **Restore a Previous Version** process enables you to undo any changes to data that have been performed since a given point in time. Clickstream Intelligence enables you to **roll back** to a previous version of data stored in your warehouse by clicking the **Undo** button that appears when a process has an error or has been temporarily stopped.

The following descriptions correspond to the fields for which you must enter information when creating a definition for the **Restore a Previous Version** process.

- **Parallel Factor**: the maximum number of jobs that should be executed simultaneously by the Clickstream Intelligence Execution Engine.
- **Version After Rollback**: indicates the prior Clickstream Intelligence database version you want to restore. A version is applied to data in the database every time a process is run. When you rollback to a previous version, all data loaded after that version is removed.
- **Analyze Levels**: indicates whether to gather statistics for Clickstream Intelligence level tables (CLK_L_%.) When enabled, statistics for the level tables are gathered each time the process is executed.

- **Analyze Dimensions**: indicates whether to gather statistics for Clickstream Intelligence dimension tables (CLK_AGENT, CLK_CLIENT_HOST, etc). When enabled, statistics for the dimension tables are gathered each time the process is executed.
- **Analyze Facts**: indicates whether to gather statistics for Clickstream Intelligence fact tables (CLK_IMPRESSION_FACT, CLK_SESSION_FACT). When enabled, statistics for the fact tables are gathered each time the process is executed.
- **Analyze Summaries**: indicates whether to gather statistics for Clickstream Intelligence Summary Layer objects (CLK_SL_%). When enabled, statistics for the Summary Layer objects are gathered each time the process is executed.
- **Percent to Analyze**: the percentage of rows (from 0.000001 to 99.99999) to sample when gathering table statistics. These statistics are used by the Oracle Cost-Based Optimizer to choose better execution plans for queries.
- **Analysis Granularity**: the granularity of statistics to collect (only pertains to partitioned tables).
  - Global, Partition, and Subpartition: Gather all (subpartition, partition, and global) statistics.
  - Global and Partition: Gather global and partition-level statistics only.
  - Global: Gather global statistics only.
  - Partition: Gather partition-level statistics only.
  - Subpartition: Gather subpartition-level statistics only.
- **Refresh Summaries**: specifies whether to refresh tables and materialized views registered with the Clickstream Intelligence Summary Layer. When enabled, the summary layer is refreshed at the end of this process.
- **Summary Refresh Type**: indicates the type of refresh to use when refreshing the Clickstream Summary Layer.
  - Default: Incrementally refresh the summary layer. If this is not possible, then perform a complete refresh. This option is often the fastest and is the recommended setting.
  - Complete: Truncate all tables and resummarize all data.
  - Incremental: Add all new data to existing summary data.
- **Summary Refresh Time Period**: specifies which summaries to refresh.

- Year: Refresh all summaries.
- Quarter: Only refresh summaries that include data for quarterly, monthly, and daily reports.
- Month: Only refresh summaries that include data for monthly and daily reports.
- Day: Only refresh summaries that include data for daily reports.

- **Refresh User-Defined Summaries**: indicates whether to include user-defined summaries from the refresh. When disabled, only the tables and materialized views shipped with Clickstream Intelligence are refreshed.

  This is a debug-only parameter, and should be used only when having problems with the refresh. The **Summary Refresh Time Period** parameter is not effective when this flag is enabled, and all non- user-defined summaries will be refreshed regardless of their time period.

- **Log Detailed Summary Messages**: specifies whether or not to count the number of rows before and after a summary object has been refreshed.

4

Loading the Clickstream Database

This chapter describes how to execute, monitor, and manage operations related to the Clickstream database. All functionality discussed in this chapter is contained within the **Manage** tab located on the right side of the Horizontal Navigation bar.

The items listed below represent the links that appear and become active on the Horizontal Navigation bar when you select the **Manage** tab. In this chapter, you'll learn about:

- **Database Processes**: perform tasks related to the Clickstream database (such as data loading), monitor currently running processes, and view details for specific database processes.
- **Data Sources**: view all data sources in the system, and view details about specific data packets associated with those data sources.

Database Processes

The **Database Processes** section of this chapter describes how to start, manage, and interpret details about the database processes you execute. This section covers the following topics:

- **The Process Status Page**
- **Starting and Stopping a Process**
- **Show Details for a Previous Run**
- **Process Types**
- **Process Execution States**
- **The Process Navigation Menu**
- **Process Details**

- **Process Messages**
- **Process Definition**
- **Subprocesses**
- **Process Data Packets**

The Process Status Page

The **Process Status** page is displayed by default when you click the **Processes** link on the Horizontal Navigation bar. It provides information related to the activity of your database, and is the page from which you can start a database process.

The **Process Status** page contains the following two headings:

- **Currently Running**: Information displayed under this heading indicates when a database process is currently in progress. If a process is running, descriptive information is displayed; if there is not a currently running process, you can **Starting and Stopping a Process**.
- **Previous Runs**: lists all completed and failed database processes. From this table, you can view the details about previously-run database processes, and view other types of information from the **The Process Navigation Menu**.

The status of the **Execution Engine** is also displayed on the **Process Status** page. The Execution Engine should run continually in the background; it *must* be running in order to execute any Clickstream processes. If the Execution Engine is not currently running, a **Warning** message is displayed at the top of the **Process Status** page. (No message appears when the Engine is currently running.)

Controlling the Clickstream Daemons

To **start** or **stop** the Execution Engine, use the same commands that also apply to the Collector Server. At the command prompt, go to the appropriate directory below.

```
(UNIX)  CLICK_HOME/click/bin

(Windows)  CLICK_HOME\click\bin
```

In the expression above, *CLICK_HOME* is the directory in which Oracle9*i*AS Clickstream Intelligence is installed. Then, use the start and stop commands below.

```
clkctl start DATABASE_LOGIN

clkctl stop DATABASE_LOGIN
```

In the expressions above, *DATABASE_LOGIN* is equal to the username for the Runtime Administrator schema, which should be entered as `clkrt`.

To connect to a Clickstream database instance other than the one specified in the `click-app.xml` file, the following commands must be used with the appropriate variables.

```
clkctl start DATABASE_LOGIN

clkctl stop DATABASE_LOGIN
```

In the commands above, *DATABASE_LOGIN* is equal to `username/password@`*`HOST:PORT:SID`* The username entered must be `clkrt`, the password corresponds to the one defined for the `clkrt` schema, the *HOST* is the name of the machine on which the database is installed, the *PORT* is the port number of the database TNS listener, and *SID* corresponds to the Oracle SID.

To **abort** the Execution Engine (and Collector Server), enter the following command:

```
clkctl abort DATABASE_LOGIN
```

where *DATABASE_LOGIN* uses the appropriate variables described above. This command stops the daemons when a process is hanging or simply taking too long to complete. After using the abort command, you must run `execln.sql` from the appropriate directory below:

(UNIX) *CLICK_HOME*`/admin`

(Windows) *CLICK_HOME*`\admin`

To **view the status** of the Clickstream daemons, use the command below:

```
clkctl status DATABASE_LOGIN
```

where *DATABASE_LOGIN* equals the appropriate variables described above.

Starting and Stopping a Process

The **Start Process** page enables you to select the database process you want to run. It lists the five Clickstream process types, accompanied by a pull-down menu that lists all existing definitions for that process type. If at least one definition does not appear for a particular process, you must first **Create a Process Definition**. (See Chapter 3, "Configuring Clickstream Intelligence".)

Only one database process can be run at a time. If a process is already running, you must either wait until the process has finished, or stop the currently running process. You can determine if a process is already in progress by checking the "Currently Running" section of the **Process Status** page. If a process is not already underway, then a new database process can be started.

How to Start a Database Process

To start a database process, follow the steps below:

1. Click the **Start Process** button located on the **Process Status** page.

   The **Start Process** page appears.

2. Select the radio button for the **Process Type** you would like to start. (For more information about process types, see the "Process Types" section that follows.)

3. Use the pull-down menu to select a **Process Definition** for your chosen process type.

4. Click **Start**.

   The process is started by the Execution Engine. The **Process Status** page displays the process status as "Running" under the **Currently Running** heading.

Stopping a Database Process

To stop a currently running process, click the **Stop** button on the **Process Status** page. Because the process actually stops only after all currently running jobs are executed, the process may appear to remain in the **Stopping** state for an extended period of time.

When the **Stopped** execution state is finally reached, however, the process is still not entirely finished- you must click the **Undo** button to completely terminate the **Stopped** process.

Show Details for a Previous Run

The **Previous Runs** section of the **Process Status** page displays all completed and failed database processes. To view the details for a previously-run process definition, follow the steps below:

1. Select the **Manage** tab.

   The **Process Status** page appears (by default).

2. Go to the **Previous Runs** heading and select the radio button beside the process for which you want to view details.
3. Click **Show Details**.

   The **Process Details** page displays information about the process, such as process type, definition, start date, status, and warehouse version.
4. To return to the **Process Status** page, click the **Processes** link on the Horizontal Navigation bar.

Process Types

You can define definitions for five types of database processes (see "Create a Process Definition" in **Chapter 3**.) To start a process, use the pull-down menu to select a particular process definition for one of the five available process types. The sections that follow describe each process type.

Load Clickstream

The **Load Clickstream** process enables the transfer and storage of Web log data into the Clickstream database. When you start a **Load Clickstream** process, the Clickstream Loader begins processing and transforming Web log data and then loads it into the database.

Note: When a **Load Clickstream** process is executed, the interface tables are first truncated (emptied) before Web log data is loaded into the Clickstream Intelligence database. Therefore, if you have loaded external (non-Web) data into the interface tables, a **Load Dimensions** process must be run immediately thereafter to ensure that interface table data is loaded into the database dimensions.

If external data is loaded into the interface tables and then a **Load Clickstream** process (instead of a **Load Dimensions** process) is executed, all interface table data will be lost when the interface tables are emptied.

Load Dimensions

The **Load Dimensions** process transfers existing interface table data into the database levels, which are used to populate the dimensions. This process does not

load data into the interface tables - the user must load external or non-Web data into the interface tables before running the **Load Dimensions** process.

Refresh Summaries

Summary Refresh updates the Summary Layer with the most current version of data in the database. It may be useful to note that you can automatically refresh summaries as part of the **Load Clickstream** process when the "Refresh Summaries" option is enabled. For more information, see the "Refresh Summaries" section of Chapter 3, "Configuring Clickstream Intelligence".

Resolve Unknown IP Addresses

The **Resolve Unknown IP Address** process operates only on the Client Host dimension. This process is typically performed when the Web log contains the IP address of the host, but does not include the resolved name. To resolve the IP address, the **Resolve Unknown IP Addresses** process queries the DNS server (in a **reverse DNS lookup**) to determine the user-friendly hostname from the client's numerical IP address.

Restore a Previous Version

The **Restore a Previous Version** process enables you to undo any changes to data that have been made since a given point in time. The database version number to which you want to roll back is specified in the "Version after Rollback" field when you initially create a process definition of this type. For example, if you run a definition that indicates "2" as the database version after roll back, running the **Restore a Previous Version** process will return the database to the version previously labeled "2."

Process Execution States

Status information about currently running processes is useful from both a tracking and a troubleshooting standpoint. For a process that is "Currently Running" on the **Process Status** page, you can click the **Show Details** button to view and track general process information via the **Process Details** page.

When a running process displays a status indicative of a problem (such as **Error** or **Failed** status), you can troubleshoot the problem by viewing **Process Details** and other information available from **The Process Navigation Menu**. Troubleshooting typically involves viewing **Process Messages** and analyzing details about **Subprocesses** (and the jobs of which they are comprised.)

Consider the following scenario in which a Load Clickstream process is running. A problem arises that causes the process to stop. The status of this process changes from **Running** to **Error** on the **Process Status** page, and three buttons are displayed:

- **Show Details**: Displays the **Process Details** page.

  This page is the first option on the **Process Navigation** menu and provides general information about the process, such as its name and definition. You can use the other options on **The Process Navigation Menu** to further investigate the process error (as described below.)

- **Undo**: Cancels the process and returns the database to its original state (just before the process started).

  The process is listed under the "Previous Runs" heading on the **Process Status** page with the status **Failed**. To view process information and access the Process Navigation menu, click the **Show Details** button.

- **Resume**: After the problem causing the process error is fixed or removed, click this button to continue the process from the point at which it originally stopped.

Depending on the option selected above, one of several execution states may ultimately result. **Completed** or **Failed** status indicates that execution for a given process has finished. An intermediate state, such as **Error**, indicates a process that has temporarily stopped, but must ultimately **Resume** or be cancelled via the **Undo** button.

Note: A process may also reach the **Unrecoverable** execution state when a user restarts the Execution Engine while a process is running, or if a job loses its database connection during execution. When the **Unrecoverable** process status is displayed, the user must click the **Undo** button before any new database processes can be run.

The following sections outline typical process execution states and scenarios that may be encountered when running a database process.

A Successful Process

- Click the **Start** button to begin a process - status is **Starting**.
- The Execution Engine begins process execution - status is **Running**.

- The process finishes successfully - status is **Completed**.

Stop a Process and Resume

- Click the **Start** button to begin a process - status is **Starting**.
- The Execution Engine begins process execution - status is Running.
- Click the **Stop** button - status is **Stopping**.
- The Execution Engine stops process execution - status is **Stopped**.
- Click the **Resume** button - status is **Starting** once again.

Stop a Process and Undo

- Click the **Start** button to begin a process - status is **Starting**.
- The Execution Engine begins process execution - status is **Running**.
- Click the **Stop** button - status is **Stopping**.
- The Execution Engine stops process execution - status is **Stopped**.
- Click the **Undo** button - status is **Undoing**.
- Process status changes to **Failed** - you can now start a new process.

Resume a Process with an Error

- Click the **Start** button to begin a process - status is **Starting**.
- The Execution Engine begins process execution - status is **Running**.
- An error occurs during execution and execution is temporarily stopped - status is **Error**.
- Click the **Resume** button - status is **Starting** once again.
- If previous errors were fixed - process reaches **Completed** status.

  If errors were not fixed - process reaches **Error** status again.

Undo a Process with an Error

- Click the **Start** button to begin a process - status is **Starting**.
- The Execution Engine begins process execution - status is **Running**.
- An error occurs during execution and execution is temporarily stopped - status is **Error**.

- Click the **Undo** button - status is **Undoing**.
- Process status changes to **Failed** - you can now start a process once again (see "Starting and Stopping a Process").

The Process Navigation Menu

The **Process Navigation** menu is displayed whenever you view specific information about a process. It is used to view information about a particular process- its details, definition, messages, subprocesses and jobs, and data packets details.

The following options are accessible from the **Process Navigation** menu:

- **Details**
- **Messages**
- **Definition**
- **Subprocesses**
- **Data Packets**

The following sections describe each item displayed on the **Process Navigation** menu.

Process Details

The **Process Details** page provides additional information about a process. For a given process, the following information is displayed:

- **Type**: the name of the process type
- **Process Definition**: the name of the specific definition for the process type indicated above
- **Start Date**: date and time that the process began
- **End Date**: the date and time at which the process reached completion
- **Elapsed Time**: amount of time needed to execute the process from start to finish, equal to (End Time - Start Time)
- **Status**: Completed, Failed, Error
- **Warehouse Version**: the database version label to which this process belongs

Process Messages

The **Messages** option on the **Process Navigation** menu provides information about the jobs that have been run for a process. The messages displayed typically indicate job errors, and can be used for troubleshooting a process that has failed or has an error.

The **Process Messages** page displays:

- **Process Messages**: Displays the (error) messages generated while the process was underway.
- **Failed Job Messages**: Lists the jobs that failed, accompanied by a date, time, and error message. To view job details, click the link for any job on the list.

Process Definition

The **Definition** menu option displays the parameters (and values) that characterize the process. This information can be modified from the **Process Definitions** page, located under the **Configure** tab.

Subprocesses

The **Subprocesses** page lists all subprocesses that were executed for a particular process. To view the jobs the comprise a particular subprocess, select the radio button beside the subprocess and click **Show Details**.

Subprocess Details

This page lists all jobs that comprise a given subprocess. For each job, additional statistics are displayed, such as the Job Type, Start Date, End Date, and Status. To drill down into the details of any given job, click the **Show Details** button.

Job Details

The **Job Details** page provides the most granular data about a database process - details about a specific job that was executed in the database. The job name is displayed with the actual command that was used to start the execution of the job and the job messages (if any) that were generated. The information provided on the **Job Details** page is typically used for detailed analysis of database processes, perhaps when a process fails or has an error.

Process Data Packets

Data packets consist of Web log files that are grouped together and compressed by the **Collector Agent** (installed on the Web server). Formation of data packets facilitates transmission from the Web server to the Clickstream **Collector Server**.

The **Process Data Packets Details** page displays specific parameters associated with the data packets for a given file. Information such as File Name, Status, Date Created, Lines, Lines Loaded, Lines Rejected, and Lines Discarded may be displayed for any data packet associated with a given data source.

Delete Data Packets

To delete data packets from the **Process Data Packets** page, select the radio button for the packets you want to permanently remove and click **Delete**. Deletion of data packets is permanent. Periodic deletion of data packets is recommended, as it creates free (unused) disk space on the Collector Server host.

Data Sources

The **Data Sources** link on the **Horizontal Navigation** bar provides access to the **Data Sources** main page. This page displays data sources for all sites in your system. To view all data packets associated with any source listed on the **Data Sources** page, select the appropriate radio button and click **Show Data Packets**.

Data Packets

The **Data Packets** main page lists, in tabular format, all data packets that have been downloaded by the Collector Server.

Note: All data packets are listed for a data source, even if the packets have not been loaded into the database.

From the **Data Packets** page, you can:

- **Show Details** - **The Data Packets Details Page** displays information about a particular data packets file.
- **Delete Data Packets** - A Delete Confirmation appears. If you are sure you want to delete the data packets, click **Delete**.

The Data Packets Details Page

When you select a data source from the **Data Sources** main page and click the **Show Data Packets** button, the **Data Packets** page appears with a listing of *all* data packets that have been downloaded for that specific data source.

To view details about an item listed on the **Data Packets** page, locate the data packets file name and click the corresponding **Show Details** link. The **Data Packets Details** page appears with the following information:

- **Date Created**: indicates when the data packet was downloaded from the Web server.
- **File Name**: the name of the data packet, as it appears on the Collector Server.
- **Status**: describes the current state of the data packets, as outlined below.
  - **Loading** or **Loaded** - this status may appear if a load process is currently running
  - **Completed** - the packet was successfully loaded into the database and the rest of the "Load Clickstream"process completed successfully.
  - **Load Failed** - the packet was not loaded into the database due to format problems or other errors. Consult the **Process Messages** page for details about the cause of the packet load failure.
  - **Transfer Failed** - indicates a problem downloading the data packet to the Collector Server.
  - **Transferred** - the packet was successfully downloaded from the Web server to the Collector Server.
- **Lines Loaded**: the number of lines loaded from the Collector Server to the Clickstream database.
- **Lines Rejected**: the number of lines *not* loaded from the Collector Server because the lines were corrupt, poorly formatted, or in the wrong data format. (The number of lines rejected does not include lines that were discarded due to filtering.)
- **Lines Discarded**: the number of lines thrown out due to filtering only.
- **Details**: click the **Show Details** link to view all infomation about the data packet from the **Data Packets Details** page.

A

Log File Types Supported by Clickstream Intelligence

Oracle9*i*AS Clickstream Intelligence supports the collection and analysis of data from the following standard log file types:

- **Apache Log Format**
- **W3C Extended Log File Format**
- **Microsoft IIS Extended Log File Format**

The log formats above contain some fields that are case-insensitive. Case-insensitive log fields are converted to lower-case when loaded into the Clickstream Intelligence database. Case-insensitive log fields include the following:

- Client IP Address
- Client Hostname
- Server IP Address
- Server Hostname
- Server Name
- The "Hostname" portion of the Referrer field

All other log fields are treated in a case-sensitive manner - that is, the original case of the field is not changed as it is loaded into the database.

This appendix provides information about each log file type supported by Oracle9*i*AS Clickstream Intelligence. Log format fields are defined in a summary table at the end of each section.

Note: Unlike Oracle9*i* Application Server, Oracle9*i*AS Clickstream Intelligence does not support Netscape Web Server log files.

Apache Log Format

Apache is an open-source Web server supported by the Apache Software Foundation. The Apache HTTP Server Project develops and maintains the Apache HTTP server, one of the most popular servers on the internet. The two formats used most often with Apache are **Common** and **Combined**.

Common Log File Format

Apache Common format displays each request as a separate line in the Web log. Fields are separated with a space, and may be enclosed in quotation marks (" "), as with the "Request Line" field in the **Apache Log File Format Example** that follows. Time in the "Date and Time" field includes an offset from Greenwich Mean Time (GMT). Null values for log fields are represented by a dash (-).

Apache Common Log Format contains all basic Web log parameters (see **Apache Log Format Details**), but does not include information about the Referrer, Agent, Time to Serve (transfer time), Domain Name, or Cookie string.

Combined Log File Format

Apache Combined format is simply an extension of Common Log Format. It contains the same fields as Apache Common Log Format, with the addition of two fields: Referrer and User Agent. Apache Combined Log Format does not provide Domain Name, Time to Serve (transfer time), or Cookie information.

Apache Log File Format Example

Consider the following example of a Web log entry in the **Apache Combined Log Format**:

```
203.93.245.97 - oracleuser [28/Sep/2000:23:59:07 -0700] "GET
/files/search/search.jsp?s=driver&a=10 HTTP/1.0" 200 2374
"http://datawarehouse.us.oracle.com/datamining/contents.htm" "Mozilla/4.7 [en]
(WinNT; I)"
```

In the entry above, the following parameters can be identified:

- **Client IP Address:** 203.93.245.97
- **Remote Logname:** - (null; value not logged by server)
- **Authenticated Username:** oracleuser
- **Request Date and Time:** [28/Sep/2000:23:59:07 -0700]
- **Request Line** (from client to server)**:**
  GET/files/search/search.jsp?s=driver&a=10 HTTP/1.0
- **Status Code** (also called the **Server Response Code**)**:** 200
- **Bytes Sent from Server to Client:** 2374
- **Referrer:** http://datawarehouse.us.oracle.com/datamining/contents.htm
- **User Agent:** Mozilla/4.7 [en] (WinNT; I)

  The User Agent field is comprised of the following data:

  – **Browser:** Mozilla (code name for Netscape)

  – **Browser Version:** 4.7

  – **Browser Language:** en (English)

  – **Browser Operating System:** WinNT
- **Filename:** search.jsp
- **Request Method:** GET
- **Transport Protocol** (and version)**:** HTTP/1.0
- **Request Stem** (path of requested document relative to its root directory on the Web server)**:** /files/search/search.jsp
- **Request Query String:** ?s=driver&a=10

Apache Log Format Details

Apache log files may contain some or all of the following fields.

Table A–1 Apache Log Format details

Field Name	Description	Field Directive	Example Values
Client IP Address	IP address of the host making the request	%a	123.1.2.3

Table A–1 (Cont.) Apache Log Format details

Field Name	Description	Field Directive	Example Values
Remote Logname	Remote logname (from identd). This field is almost always null ("-")	%l	-
Authenticated User Name	Name or identifier of the authenticated user	%u	j.user
Request Date and Time	Date and time at which the request was received by the server (in Common Log time format)	%t	[14/Jan/2001 23:59:09 -0800]
Request Line	Line in an HTTP request that contains the Method, Request-URI, and Protocol ending with <CR><LF>	%r	GET /path/to/resource?query+string HTTP/1.1
Status Code	The server HTTP response code	%s	200, 304, 404
Bytes Sent	Number of bytes transferred from the server to the client	%b, %B	14378
Referrer	URI of the resource (typically a web site) from which the requested URI was obtained	"%{Referrer"	http://www.google.com/search?q=oracle
User Agent	Information about the browser that made the request	"%{User-Agent}i"	Mozilla/4.51 [en] (WinNT; U)
Cookie String	Cookie name-value pairs separated by a semicolon and space character	"%{Cookie}i"	COOKIE1=value; COOKIE2=value
Client Hostname	DNS hostname of the host making the request	%h	client-123lp.domain.net
Server IP Address	IP address of the host fulfilling the request	%A	123.1.2.3
Filename	Filename of the requested URI	%f	index.html
Request Method	HTTP method of the request	%m	GET, POST

Table A–1 (Cont.) Apache Log Format details

Field Name	Description	Field Directive	Example Values
Transport Protocol	HTTP protocol version string	%H	HTTP / 1.1
Server Port	Port number of the listener fulfilling the request	%p	80
Server Process ID	Identifier of the process that fulfilled the request	%P	4971
Request Stem	Stem (path) component of the requested URI	%U	/ path / to / reso urce
Request Query String	Query component of the requested URI	%q	?page=catalog &x=100&y=0
Time to Serve	Time taken to serve the request (in seconds)	%T	0, 1, 2, 802
Server Name	Server name of the host fulfilling the request	%v, %V	server1
Session Identifier Field	Session identifier as a separate field		1227584, E034080020CB 1B7C
Visitor Identifier Field	Visitor identifier (such as a cookie) as a separate field		710EA2571666 2CACE0
General Purpose Fields 1-10	Users may define (customize) up to ten log fields		Any value(s)

W3C Extended Log File Format

W3C Extended is a flexible, highly configurable Web log format developed by the World Wide Web Consortium (W3C) as a common standard to support the needs of servers, clients, and proxies. Each W3C Extended format log file is self-identifying -- it displays at the beginning of each log file a header containing information about the data types recorded, as well as the version of the extended log file format used. As with **Microsoft IIS Extended Log File Format**, field directives may begin with:

- **c** (client)
- **s** (server)
- **cs** (client to server)

- **sc** (server to client)

W3C Extended log fields are separated by a whitespace (although tabs are most commonly used and are encouraged by the W3C specification). A null value for a field is typically represented by a dash (-); the "Date" and "Time" fields are expressed as GMT time.

W3C Extended Log File Format Example

Consider the following example of a Web log entry generated in **W3C Extended Log File Format**:

```
#Version: 1.0
#GMT-Offset: -0800
#Software: Oracle9iAS Web Cache/2.0.0.2.0
#Start-Date: 2001-10-31 00:00:18
#Fields: c-ip c-dns c-auth-id date time cs-method cs-uri sc-status
bytes cs(Cookie) cs(Referrer) time-taken cs(User-Agent)
#Date: 2001-10-31 00:00:18
64.103.37.2      client_joaz7      DMS.user      2001-10-31     00:00:18
GET      /admin/images/oc_bottomleft.gif    200     350    "BIGipServerwww_
webcache_pool=1443321748.19460.0000;ORA_UCM_
AGID=%2fMP%2f8M7%3etSHPV%40%2fS%3f%3fDh3VHO"
"http://www.oracle.com/nl/partner/content.html"      370879      "Mozilla/4.5
[en] (WinNT; I)"
```

In the entry above, the following parameters can be identified. The **#Fields** header indicates the order of fields that appear in the log file entry.

- **Client IP Address:** 64.103.37.2
- **Client Resolved IP Address:** client_joaz7
- **Authenticated Username:** DMS.user
- **Request Date:** 2001-10-31
- **Request Time:** 00:00:18
- **Client to Server Request Method:** GET
- **Request URI:** /admin/images/oc_bottomleft.gif
- **Status Code** (or **Server Response Code**)**:** 200
- **Bytes Sent from Server to Client:** 350

- **Cookie String:** BIGipServerwww_webcache_pool=1443321748.19460.0000;ORA_UCM_AGID=%2fMP%2f8M7%3etSHPV%40%2fS%3f%3fDh3VHO
- **Referrer:** http://www.oracle.com/nl/partner/content.html
- **Time Taken** (to serve the request)**:** 370879 (seconds)
- **User Agent:** Mozilla/4.5 [en] (WinNT; I)

  The User Agent field is comprised of the following data:

  – **Browser:** Mozilla (
  – **Browser Version:** 4.5
  – **Browser Language:** en (English)
  – **Browser Operating System:** WinNT

WC3 Extended Log File Format Details

W3C Extended Format log files may contain the following fields:

Table A–2 WC3 Extended Log File Format details

Field Name	Description	Field Directive	Example Values
Request Date	Date on which the request was received by the server	date	2001-01-14
Request Time	Time at which the request was received by the server	time	23:59:09
Request Date and Time	Date and time at which the request was received by the server (separated by a space)	date time	2001-01-14 23:59:09
Time to Serve	Time taken to complete the request (in seconds)	time-taken	0.062, 0.392, 2, 802.1
Bytes Sent	Number of bytes transferred from the server to the client	bytes	14378
Cache Hit	Indicates if the request was fulfilled using cached content (0 = cache miss, 1 = cache hit)	cached	0, 1
Client IP Address	IP address of the host (client) making the request	c-ip	123.1.2.3

Table A–2 (Cont.) WC3 Extended Log File Format details

Field Name	Description	Field Directive	Example Values
Server IP Address	IP address of the host (server) fulfilling the request	s-ip	123.1.2.3
Client Hostname	Resolved DNS hostname of the host (client) making the request	c-dns	client-123lp.domain.net
Server Hostname	Resolved DNS hostname of the host fulfilling the request	s-dns	server1, server1.domain.com
Server Port	Port number of the listener fulfilling the request	s-port	80
Status Code	HTTP response code returned from the server	sc-status	200, 304, 404
Status Comment	Comment that defines the server status code	sc-comment	OK, Not Found
Authenticated User Name	Name or identifier of the authenticated user	c-auth-id	j.user
Request Method	HTTP method of the request	cs-method	GET, POST
Transport Protocol	HTTP protcol version string	cs- protocol	HTTP / 1.1
Request URI	Requested Uniform Resource Identifier (Request-URI)	cs-uri	/ path / to / resource?page=catalog&x=100&y=0
Request Stem	Stem (path) component of the requested URI	cs-uri- stem	/ path / to / resource
Request Query String	Query component of the requested URI	cs-uri- query	?page=catalog&x=100&y=0
User Agent	Information about the user agent (browser) originating the request	cs (User-Agent)	Mozilla / 4.51 [en] (WinNT; U)
Cookie String	Cookie name-value pairs separated by a semicolon and space character	cs (Cookie)	COOKIE1=value; COOKIE2=value
Referrer	URI of the resource (typically a web site) from which the requested URI was obtained	cs (Referrer)	http: / / www.google.com / search?q=oracle
Session Identifier Field	Session identifier as a separate field		1227584, E034080020CB1B7C

Table A–2 (Cont.) WC3 Extended Log File Format details

Field Name	Description	Field Directive	Example Values
Visitor Identifier Field	Visitor identifier (such as a cookie) as a separate field		710EA25716662CACE0
General Purpose Fields 1-10	Users may define up to ten log fields		Any value(s)

Microsoft IIS Extended Log File Format

Microsoft IIS Extended Web logs are loosely based upon W3C Web log format. Microsoft IIS Extended logs contain the basic information found in both Apache and W3C format Web logs, in addition to other unique fields (such as "Win32 Status"). The beginning of every Microsoft IIS Extended log file (IIS 4.0 and higher) displays a header indicating the fields in the order that they were recorded by the Web server. Most field indentifiers begin with one of the following prefixes:

- **c** (client)
- **s** (server)
- **cs** (client to server)
- **sc** (server to client)

Fields in Microsoft IIS Extended format logs can be separated by a space. To ensure that spaces contained in the data are not misinterpreted as field delimiters, fields are URL-encoded. Null values are displayed as a dash (-). The "Date" field is expressed in local time.

Microsoft IIS Extended Log File Format Example

Consider the following example of a Web log entry generated in **Microsoft IIS Extended Log File Format**:

```
2000-09-28 06:59:07 203.93.245.97 oracleuser W3SVC1 DATAWAREHOUSE
144.25.86.192 GET /files/search/search.jsp s=driver&a=10 200 0 2374 369
2938 80 HTTP/1.0 Mozilla/4.7+[en]+(WinNT;+I) -
http://datawarehouse.us.oracle.com/datamining/contents.htm
```

In the entry above, the following parameters can be identified:

- **Request Date:** 2000-09-28

- **Request Time:** 06:59:07
- **Client IP Address:** 203.93.245.97
- **Authenticated User Name:** oracleuser
- **Site Name:** W3SVC1
- **Web Server Name:** DATAWAREHOUSE
- **Web Server IP Address:** 144.25.86.192
- **Request Method:** GET
- **Request Stem** (relative to root directory on Web server)**:** /files/search/search.jsp
- **Request Query String:** s=driver&a=10
- **Status Code** (or **Server Response Code**)**:** 200
- **Win 32 Status:** 0 (internal status code specific to Microsoft IIS Web server)
- **Bytes Sent** (from server to client)**:** 2374
- **Bytes Received** (from client to server)**:** 369
- **Time to Serve the Request:** 2938 (milliseconds)
- **Server Port:** 80
- **Transport Protocol** (and version)**:** HTTP/1.0
- **User Agent:** Mozilla/4.7+[en]+(WinNT;+1)

  The User Agent field is comprised of the following data:

  - **Browser:** Mozilla (Netscape)
  - **Browser Version:** 4.7
  - **Browser Language:** en (English)
  - **Browser Operating System:** WinNT
- **Cookie String:** - (null; no value recorded by server)
- **Referrer:** http://datawarehouse.us.oracle.com/datamining/contents.htm

Microsoft IIS Extended Format Details

Microsoft IIS Extended Format log files may contain the following fields:

Table A–3 Microsoft IIS Extended Format details

Field Name	Description	Field Directive	Example Values
Request Date	Date at which the request was received by the server	date	2001-01-14
Request Time	Time at which the request was received by the server	time	23:59:09
Client IP Address	IP address of the host making the request	c-ip	123.1.2.3
Authenticated User Name	Name or identifier of the authenticated user	c-auth-id	d.smith
Site Name	Name of the server that MS IIS sets up	s-sitename	MYSITE
Server Name	Server name of the host fulfilling the request	%v, %V	server1
Server IP Address	IP address of the host fulfilling the request	s-ip	123.1.2.3
Request Method	HTTP method of the request	cs-method	GET, POST
Request Stem	Stem (path) component of the URI requested by the client	cs-uri- stem	/path/to/resource
Request Query String	Query component of the URI requested by the client	cs-uri- query	?page=catalog&x=100&y=0
Status Code	HTTP status code returned from the server	sc-status	200, 304, 404
Win32 Status	Debug status code - an internal status code specific to Microsoft IIS Web server	sc-win32-status	Any number from zero to infinity.
Bytes Sent	Number of bytes transferred from the server to the client	sc-bytes	14378
Bytes Received	Number of bytes transferred from the client to the server	cs-bytes	14378
Time to Serve	Time taken to serve the request (in milliseconds)	time-taken	0.062, 0.392, 2, 802.1

Table A–3 (Cont.) Microsoft IIS Extended Format details

Field Name	Description	Field Directive	Example Values
Server Port	Port number of the listener fulfilling the request	s-port	80
Transport Protocol	HTTP protcol version string	cs- protocol	HTTP / 1.1
User Agent	Information about the user agent (browser) originating the request	cs (User-Agent)	Mozilla / 4.51 [en] (WinNT; U)
Cookie String	Cookie name-value pairs separated by a semicolon and space character	cs (Cookie)	COOKIE1=value; COOKIE2=value
Referrer	URI of the resource (typically a web site) from which the requested URI was obtained	cs (Referrer)	http: / / www.google.com / search?q=oracle
Session Identifier Field	Session identifier as a separate field		1227584, E034080020CB1B7C
Visitor Identifier Field	Visitor identifier (such as a cookie) as a separate field		710EA25716662CACE0
General Purpose Fields 1-10	Users may define (customize) up to ten log fields		Any value(s)

B

Installing a Dedicated Clickstream Database

The information in this appendix is intended for use by System Administrators, Database Administrators (DBAs), or individuals who have installed Oracle9*i* Application Server. In this appendix, you'll learn how to configure an Oracle9*i* Enterprise Edition database for use with Oracle9*i*AS Clickstream Intelligence via the Oracle9*i*AS Database Installation Wizard.

Note: Throughout this appendix and the Oracle9iAS Clickstream Intelligence documentation set, the terms "dedicated (Clickstream) database" and "standalone database" are used interchangeably with "customer database." All terms refer to a separate, Oracle9*i* Enterprise Edition database that is used for collection of Clickstream data exclusively. The term **customer database** is preferred terminology, as it is used throughout the Oracle9*i* Application Server documentation set.

This appendix contains the following sections:

- **Dedicated Clickstream Database Installation Summary**: Provides a summary of all steps required to start using Clickstream Intelligence with a cusomer database.
- **Hardware Specifications**: Suggests the optimal hardware configuration for installation of a dedicated Clickstream database.
- **Software Specifications**: Describes the software required to install a customer database.
- **Configuring the Database**: Describes configuration of the database for use with Oracle9*i*AS Clickstream Intelligence.

- **Running the Database Installation Wizard**: Describes the parameters required for installation of the Clickstream Intelligence database schemas. Also includes information about automatic or manual creation of tablespaces.

Dedicated Clickstream Database Installation Summary

To begin using Clickstream Intelligence with a dedicated customer database, complete the following tasks (in the order specified).

1. Install Oracle9*i* Enterprise Edition database and set the database password.
2. Install the appropriate Patchset for your operating system. To do this, go to the following URL:

   `http://metalink.oracle.com`

   Following successful login with a valid username and password, click **Patches** from the menu on the left side of your screen. Provide the following **Patch Download** query data. Fields not listed below require no modification; leave the default values in the fields provided.

   - **Product Family:** Oracle Server
   - **Release:** 9.0.1.3
   - **Platform:** (select the same platform as your Oracle9*i* database)
   - **Language:** (select the language that matches the one you selected during Oracle9*i* database installation.)
   - **Limit Search To:** All Product Patches

   Click the **Submit** button. From the list of patches displayed, click the link for one of the patches listed below. Download the patch as specified in the *ReadMe* file.

   - **(UNIX)** Oracle 9.0.1.3 Patch Set for Oracle Data Server. ID: 1502843; Patchset: 2271678.
   - **(Windows)** Oracle9*i* 9.0.1.3.1 Patch 3 for Windows NT and Windows 2000. ID: 1670440; Patchset: 2367681.
3. Launch the Database Installation Wizard. For details, see "Launching the Database Installation Wizard".
4. Stop and then re-start all application server processes to ensure that Apache server recognizes and uses the modifications you made to the `click-app.xml` file via the Database Installation Wizard. Use the following commands:

```
ORACLE_HOME/opmn/bin/opmnctl stopall

ORACLE_HOME/opmn/bin/opmnctl startall
```

In the expressions above, the *ORACLE_HOME* environment variable represents the directory in which Oracle9*i*AS is installed.

5. Create an End User Layer (EUL) with the Oracle9*i* Discoverer Administrator "EUL Manager dialog."

   The EUL should be created by the DBA who installed the dedicated Clickstream database. To learn how to create an EUL, see the *Oracle9i Discoverer Administrator Administration Guide.*

6. Use the Oracle9*i* Discoverer Administrator Import Wizard to populate the EUL with the Clickstream EUL objects contained in the `clickstream_intelligence_eul_full.eex` file. This file is located in the following directory:

   ```
   CLICK_HOME/install/analytics/discoverer/LANG
   ```

   In the expression above, *CLICK_HOME* is the directory in which Oracle9*i* Clickstream Intelligence is installed, and *LANG* is the code corresponding to the language selected during Oracle9*i*AS installation.

   For information about populating the EUL with Clickstream metadata, see the *Oracle9i Discoverer Administrator Administration Guide.*

7. Access the Clickstream Intelligence Home Page. To do this, launch an Internet browser window and enter the following URL:

   ```
   http://HOST:PORT/click/
   ```

   The *HOST* is the name of the computer on which Clickstream Intelligence is installed, and the *PORT* is the logical channel on which the host listens for requests.

8. From the Clickstream Intelligence Home page, click the link for the **Runtime Administrator**. Log on to the Runtime Administrator with the username **clkrt** and the password you defined with the Clickstream Database Installation Wizard.

9. Using the Runtime Administrator, specify the types of data you want to acquire, process, and store in the customer database. To learn more, see the *Oracle9iAS Clickstream Intelligence Administrator's Guide.*

10. Load Web server log file data into the dedicated Clickstream database. For information about this and other database processes, see Chapter 4, "Loading the Clickstream Database".

11. Enable access to Clickstream Analytics by creating a "Private" database connection. To do this, access the Clickstream Intelligence Home page and click the **Analytics-Discoverer Viewer** link. The **Connect to Discoverer Viewer** page is displayed.

12. On the **Connect to Discoverer Viewer** page, provide the Connection Name, Connection Description, and Locale (the language used to view Clickstream Analytics).

    Specify the **Database Connection Details** below, and then click **Apply** to return to the **Connect to Discoverer Viewer** page.

    - **User Name**: enter **clkana** as your user name.
    - **Password**: enter the `clkana` schema password you defined via the Clickstream Database Installation Wizard.
    - **Database**: enter the customer database SID.

13. To access Clickstream Analytics, click the link for the **name** of the private connection you just created.

    For additional information on creating a private connection and creating public connections for multiple Clickstream Analytics users, see the *Oracle9i Discoverer Administrator's Administration Guide.*

Hardware Specifications

It is recommended that you install the Oracle9*i*AS database on a dedicated server for the following reasons:

- The operations or queries performed on a Oracle9*i*AS Clickstream Intelligence database are CPU-intensive.
- A typical Oracle9*i*AS Clickstream Intelligence database instance uses a significantly large amount of memory when processing some querys.
- The storage requirements of Oracle9*i*AS grow quickly over time and may reach terabytes in size.

The minimum hardware requirements for Oracle9i Enterprise Edition database can be found in the *Oracle9i Database Installation Guide.*

Software Specifications

An Oracle9*i* Enterprise Edition database is required for use with Oracle9*i*AS because Oracle9*i*AS Clickstream Intelligence requires specific features that are only available in that database. Those features are listed below:

- **Partitioning**: Partitioning improves database performance by breaking large tables and indexes into components of a much smaller size. This enables users to perform operations, querys, or other actions on an individual partition rather than the entire table. Additionally, partitioning reduces the number of data management problems that can occur within a large tablespace.
- **Bitmapped indexes**: Bitmapped indexes use bitmaps instead of a list of `rowids` to quickly identify key values
- **Parallel processing**: Parallel processing improves query performance by enabling the database to use multiple processes to execute a single SQL statement.

For more information about installing an Oracle9*i* Enterprise Edition database, see the *Oracle9i Database Installation Guide*.

Configuring the Database

To configure Oracle9*i* Enterprise Edition database for Clickstream Intelligence, you must complete the following tasks:

- Create tablespaces in the database
- Modify the database initialization parameter file
- Partition the database (optional, but recommended)

The **Database Installation Wizard** enables you to complete the tasks above. With the Database Installation Wizard you can automatically or manually create tablespaces, and install the Clickstream Intelligence database schemas. If you do not run the Database Installation Wizard, the Clickstream Intelligence database schemas will not be installed.

Information regarding customer database configuration is contained in the following sections of this appendix:

- **Tablespaces** - Provides a high-level discussion of tablespaces and describes the tablespaces required by Clickstream Intelligence.

- **Creating Tablespaces With the Database Installation Wizard** - Explains how to use the Database Installation Wizard for creation of required tablespaces.
- **Manually Creating Tablespaces** - Explains how to create tablespaces manually, and provides sizing formulas and examples.
- **Partitioning** - Describes the various aspects of partitioning required by Clickstream Intelligence.

Tablespaces

Tablespaces are logical containers within an Oracle database that enable data storage and access. Tablespaces are comprised of physical data files (as well as database schema objects such as indexes and tables). Data files define a tablespace's capacity - adding files to a tablespace, for example, increases its data storage capacity. A database can contain one or more tablespaces, and tablespaces can contain one or more data files.

Tablespace size is typically specified by the Database Administrator (DBA) when the database is installed. The size of the data files within a tablespace can automatically extend (as needed), or can be set to a finite value. For more information about tablespaces, see *Oracle9i Database Concepts*.

Oracle9*i*AS Clickstream Intelligence requires the following tablespaces:

- **Analytics Tablespace (CANA)**

  Provides storage for the Discoverer End User Layer (EUL), as well as the Discoverer workbooks and worksheets that comprise Clickstream Analytics.

- **Warehouse Audit Tablespace (CAUDT)**

  Provides storage for historical audit records and details about previously-run processes and jobs.

- **Warehouse Data Tablespace (CDATA)**

  Provides storage for all dimensional data and serves as the default tablespace for fact records if fact table partitions are not created.

- **Warehouse Index Tablespace (CINDX)**

  Provides storage for dimension indexes, and serves as the default tablespace for fact table bitmap indexes if fact table partitions are not created.

- **Warehouse Stage Tablespace (CSTGE)**

Provides storage for incoming data as it is transformed and loaded into the warehouse.

- **Warehouse Summary Layer Tablespace (CSUMM)**

  Provides storage for both partitioned and non-partitioned summary layer tables and materialized views.

- **Temporary Tablespace (TEMP)**

  Provides temporary storage for the database.

There are two ways to create the tablespaces listed above. You can use the Oracle9*i*AS Database Installation Wizard, or you can create the tablespaces yourself. The following sections describe these options:

- **Creating Tablespaces With the Database Installation Wizard**
- **Manually Creating Tablespaces**

The tablespace size information provided in the "Manually Creating Tablespaces" section is based on a formula that uses the average number of hits a Web site receives per day, the amount of Web log data those hits generate, and how long the Web log data is stored in the database before it is archived, to determine optimal size. Consider your own Web site and your own storage needs as you determine the appropriate tablespace sizes for the Clickstream database.

Creating Tablespaces With the Database Installation Wizard

The **Database Installation Wizard** assists you with the creation and sizing of the tablespaces described in the **Tablespaces** section. if you want to create tablespaces with the assistance of the Database Installation Wizard, select **Typical** on the **Tablespace Option** page. Based on the information you specify on the **Database Sizing Parameters** page, the Database Installation Wizard automatically calculates the number and size of the tablespaces needed in the database. Tablespace information provided by the Database Installation Wizard is influenced by the following factors:

- The average number of page requests served by all Web sites that will store data in the database.
- The maximum number of page requests served by all Web sites that will store data in the database.
- The number of days data will be stored in the database. (It is recommended that you plan ahead when determining this number.)

- The frequency with which data will be loaded into the database.

For more information about the **Database Installation Wizard**, see "Running the Database Installation Wizard" in this appendix.

Manually Creating Tablespaces

To manually create and size the required tablespaces for the Clickstream database, you must still run the **Database Installation Wizard**. When you reach the **Tablespace Option** page, however, you must select **Custom** instead of **Typical**. By selecting **Custom**, you inform the Database Installation Wizard that it should search for pre-existing tablespaces in the database.

It is required that you create the following tablespaces within your dedicated Clickstream database:

- **Analytics Tablespace (CANA)**
- **Warehouse Audit Tablespace (CAUDT)**
- **Warehouse Data Tablespace (CDATA)**
- **Warehouse Index Tablespace (CINDX)**
- **Warehouse Stage Tablespace (CSTGE)**
- **Warehouse Summary Layer Tablespace (CSUMM)**
- **Temporary Tablespace (TEMP)**

To size your tablespaces appropriately, use the**Tablespace Sizing Formula** provided below. For examples of tablespace size calculations via the Tablespace Sizing Formula, see the "Calculated Tablespace Values" section. For more information about creating tablespaces, see the *Oracle9i Database Administrator's Guide*.

Tablespace Sizing Formula

Tablespace Size = C_1 * (Average Number of Hits Per Day) * (Number of Days to Store Historical Data) + C_2

The components of the formula above explained in the text that follows. If you use the Tablespace Sizing Formula to calculate tablespace size(s), you should be familiar with the log format(s) of data stored in the tablespace, as well as the volume of Web traffic associated with your site.

- **C_1:** This numerical constant is dependent on log format. It is the average amount of space used by each hit in each tablespace after Web log data is

loaded into the customer database. (Example values for this constant can be found in the "Calculated Tablespace Values" section.)

- **Average Number of Hits Per Day**: This variable is equal to the average number of hits per day served by all Web servers. As the number of hits to your Web site increases, tablespace size requirements also increase.

> **Note:** When using the Tablespace Sizing Formula to calculate the size of the Warehouse Stage Tablespace, substitute **(Maximum Number of Hits Per Day)** in place of the value you would typically enter for (Average Number of Hits Per Day).

- **Number of Days to Store Historical Data**: This variable is equal to the number of days of historical data stored in the tablespace before archiving. Storing data in the database for long periods of time increases the size of the tablespace, while storing data in the database for shorter time periods decreases tablespace size requirements.

> **Note:** When using the formula to calculate the Warehouse Stage Tablespace size, substitute **(Number of Days Per Load)** in place of (Number of Days to Store Historical Data).

- $\mathbf{C_2}$. This constant is equal to the minimum space requirement for the tablespace. Hypothetical values for this constant are contained in the examples provided in the "Calculated Tablespace Values" section of this appendix.

Calculated Tablespace Values

This section provides example tablespace values, calculated via the **Tablespace Sizing Formula**. All of the calculated example values below assume the following scenario for a hypothetical Web site:

- The Web site receives an average of 500,000 hits per day. (This data is stored in the database.)
- Data is stored in the database for 90 days before it is archived.

You can use the examples below as a guide when calculating tablespace sizes for your own dedicated Clickstream database. To calculate tablespace sizes for your

Clickstream database, substitute your own numerical values for the number of hits per day and the number of days data is stored in the database.

Note: The **Analytics Tablespace (CANA)** and **Temporary Tablespace (TEMP)** *do not* adhere to the **Tablespace Sizing Formula**.

Analytics Tablespace: When creating the Analytics tablespace, You must use the value provided below.

Temporary Tablespace: There is no one correct size for the Temporary tablespace. We recommend a minimum size based on the assumptions above. Your needs may be larger or smaller. Plan accordingly.

Analytics Tablespace (CANA): 100 MB (Required)

Warehouse Data Tablespace (CDATA): 19,975 MB

```
Size = 0.45 KB per hit * (500,000 hits per day)* (90 days) + 200 MB = 19,975 MB
(19.50 GB)
```

Warehouse Index Tablespace (CINDX): 8,989 MB

```
Size = 0.20 KB per hit * (500,000 hits per day) * (90 days) + 200 MB = 8,989 MB
(8.78 GB)
```

Warehouse Summary Layer Tablespace (CSUMM): 11,186 MB

```
Size = 0.10 KB per hit * (500,000 hits per day) * (90 days) + 200 MB = 4,595 MB
(4.49 GB)
```

Warehouse Staging Tablespace (CSTG): 600 MB

```
Size = 0.60 KB per hit * (500,000 hits per day) * (1 day) + 300 MB = 600 MB
(0.59 GB)
```

Temporary Tablespace (TEMP): Greater than 350 MB Recommended

Initialization Parameters File

An Oracle initialization parameter file contains a list of database parameters and a value for each parameter. This file is used to start the database and determine the names and locations of the database control files, in addition to other functions. For more information about Oracle initialization parameter files, see the *Oracle9i Database Reference*. Modifying the default parameter values in the Oracle9*i* Enterprise Edition database initialization parameter file optimizes the performance of the Clickstream database by adjusting certain parameter values to better acommodate your storage or processing needs.

The "Initialization Parameter Details" section below provides the parameters and values for an initialization parameter file used by an Oracle9*i* database. The examples that are calculated below assume the following:

- The database stores data for a Web site averaging 500,000 hits per day.
- Data is loaded into the database daily.
- Data is stored for 180 days.

Use the information below as an example to assist you with determining how to set the values in your database initialization parameter file.

Initialization Parameter Details

The information in this section is displayed in the following format:

Parameter Type

- Example Parameter: Value

For more information about the functions of the parameters listed below, see the *Oracle9i Database Reference*.

Note: Required parameter values appear in *italic* and must correspond to the values specified below.

Cache and I/O

- db_block_size: 16384
- db_cache_size: 256MB

File Configuration

- db_files: 1000

Cursors and Library Cache

- open_cursors: 1000

Job Queues

- job_queue_processes: *16*

Miscellaneous

- compatible: 9.0.1

Pools Parameter

- java_pool_size: 50MB
- large_pool_size: 256MB
- shared_pool_size: 128MB

Optimizer

- optimizer_features_enable: 9.0.1
- optimizer_max_permutations: 40000
- optimizer_mode: Choose
- query_rewrite_enabled: *True*
- query_rewrite_integrity: *Trusted*
- star_transformation_enabled: True

Parallel Queries

- parallel_automatic_tuning: True

Processes and Sessions

- processes: 200

Redo Log and Recovery

- fast_start_mttr_target: 300

Resource Manager

- resource_manager_plan: system_plan

Sort, Hash Joins, Bitmap Indexes

- pga_aggregate_target: 512MB
- workarea_size_policy: Auto

System Managed Undo and Rollback Segments

- undo_management: Auto

Transactions

- dml_locks: 1000

Partitioning

Partitioning a database enables you to quickly access data in large tables and indices by dividing those objects into smaller, more manageable pieces called **partitions**. Partitioning improves the data management, processing, and querying capabilities of the database by making data easier to access. While there are several ways to partition the tables in a database, Clickstream Intelligence requires range partitioning.

Range partitioning enables the storage of data in specific partitions that are based on a partition key. Within Oracle9*i*AS, that partition key is a date. Range partitioning is most effective for Web log data because it can be easily defined and sorted by date. Range partitioning the database tables and indexes into equal segments also improves the database's management of objects. For more information on the benefits of range partitioning, see *Oracle9i Database Concepts*.

Oracle9*i*AS provides the PL/SQL package called **CLK_MNT_PARTITIONS** to create range partitions in the Fact and Mview (materialized view) tables. To optimize performance, it is recommended that you use the **CLK_MNT_PARTITIONS** package to equipartition all Fact and Mview tables. The **CLK_MNT_PARTITIONS** package also provides additional procedures that enable you to more effectively control partitioning in the database. The following procedures are available:

- **CREATE_ALL_PARTITIONS**
- **CREATE_ALL_FACT_PARTITIONS**
- **CREATE_ALL_MVIEW_PARTITIONS**
- **CREATE_FACT_PARTITIONS**
- **CREATE_MVIEW_PARTITIONS**

Range Partitioning the Database

1. Launch SQL*Plus.
2. Select the appropriate partitioning method from the choices below.
3. Enter the code string associated with the partitioning method you chose into SQL*Plus, replacing the code variables with your own data.

CREATE_ALL_PARTITIONS

This procedure creates partitions on every partitioned object in the CLKRT schema. The advantage of using this procedure is that it enables you to simultaneously

equipartition all partitioned database objects within the CLKRT schema. Equipartitioning the CLKRT schema improves the refresh performance of all materialized view objects.

The syntax, an example, and the parameters of a call used to invoke the CREATE_ALL_PARTITIONS procedure appear below:

Syntax

```
procedure create_all_partitions(
    p_fact_tablespace         IN     VARCHAR2,
    p_mview_tablespace        IN     VARCHAR2,
    p_first_date              IN     DATE,
    p_date_less_than          IN     DATE,
    p_days_per_part           IN     NUMBER,
    p_fact_index_tablespace   IN     VARCHAR2 default null,
    p_mview_index_tablespace  IN     VARCHAR2 default null
);
```

Example

```
begin
    clk_mnt_partitions.create_all_partitions(
        p_fact_tablespace=>'CDATA',
        p_mview_tablespace=>'CSUMM',
        p_first_date=>to_date('01-NOV-2001','DD-MON-YYYY'),
        p_date_less_than=>to_date('30-NOV-2001','DD_MON_YYYY'),
        p_days_per_part=>1,
        p_fact_index_tablespace =>'CFACTINDX',
        p_mview_index_tablespace=>'CMVIEWINDX'
        );
end;
```

Parameters

- **p_fact_tablespace**: The name of the tablespace in which the created fact partitions are placed. The default value is CDATA.
- **p_mview_tablespace**: The name of the tablespace in which the created materialized view partitions are placed. The default value is CSUMM.
- **p_first_date**: The beginning date in the date range for which you want to create partitions.
- **p_date_less_than**: The ending date in the date range for which you want to create partitions.

- **p_days_per_part**: The number of days contained in each partition.
- **p_fact_index_tablespace**: The name of the tablespace in which the fact index partitions are placed. If a parameter is not supplied, the fact index partitions are placed in the same tablespace as fact data. Placing these partitions in the same tablespace as fact data can create storage problems if your data tablespace has a large extent size.
- **p_mview_index_tablespace**: The name of the tablespace in which the materialized view index partitions are placed. If a parameter is not supplied, thet materialized view index partitions are placed in the same tablespace as materialized view data. Placing these partitions in the same tablespace as materialized view data can create storage problems if your data tablespace has a large extent size.

CREATE_ALL_FACT_PARTITIONS

The CREATE_ALL_FACT_PARTITIONS procedure creates partitions for every fact object within the database. Call this procedure if you want all fact objects partitioned equally. This procedure enables you to partition fact objects differently from the materialized view objects.

Partitioning fact objects and materialized view objects differently is *not* recommended.

The syntax, an example, and the parameters of the call used to invoke the CREATE_ALL_FACT_PARTITIONS procedure appear below:

Procedure

```
procedure create_all_fact_partitions(
    p_fact_tablespace         IN     VARCHAR2,
    p_first_date              IN     DATE,
    p_date_less_than          IN     DATE,
    p_days_per_part           IN     NUMBER,
    p_fact_index_tablespace   IN     VARCHAR2 default null
    );
```

Example

```
begin
    clk_mnt_partitions.create_all_fact_partitions(
        p_fact_tablespace=>'CDATA',
        p_first_date=>to_date('01-NOV-2001','DD-MON-YYYY'),
        p_date_less_than=>to_date('30-NOV-2001','DD_MON_YYYY'),
        p_days_per_part=>1,
```

```
        p_fact_index_tablespace=>'CFACTINDX'
        );
end;
```

Parameters

- **p_fact_tablespace**: The name of the tablespace in which the created fact partitions are placed.
- **p_first_date**: The beginning date in the date range for which you want to create partitions.
- **p_date_less_than**: The ending date in the date range for which you want to create partitions.
- **p_days_per_part**: The number of days contained in each partition.
- **p_fact_index_tablespace**: The name of the tablespace in which the fact index partitions are placed. If a parameter is not supplied, the fact index partitions are placed in the same tablespace as fact data. Placing these partitions in the same tablespace as fact data can create storage problems if your data tablespace has a large extent size.

CREATE_ALL_MVIEW_PARTITIONS

The CREATE_ALL_MVIEW_PARTITIONS procedure creates partitions for every materialized view object within the database.

Partitioning materialized view objects and fact objects differently is *not* recommended.

The syntax, an example, and the parameters of the call used to invoke the CREATE_ALL_MVIEW_PARTITIONS procedure appear below:

Syntax

```
procedure create_all_mview_partitions(
    p_mview_tablespace        IN     VARCHAR2,
    p_first_date              IN     DATE,
    p_date_less_than          IN     DATE,
    p_days_per_part           IN     NUMBER,
    p_mview_index_tablespace  IN     VARCHAR2 default null
);
```

Example

```
begin
```

```
    clk_mnt_partitions.create_all_mview_partitions(
        p_mview_tablespace=>'CSUMM',
        p_first_date=>to_date('01-NOV-2001','DD-MON-YYYY'),
        p_date_less_than=>to_date('30-NOV-2001','DD_MON_YYYY'),
        p_days_per_part=>1,
        p_mview_index_tablespace=>'CMVIEWINDX'
        );
end;
```

Parameters

- **p_mview_tablespace**: The name of the tablespace in which the created materialized view partitions are placed. The default value is CSUMM.
- **p_first_date**: The beginning date in the date range for which you want to create partitions.
- **p_date_less_than**: The ending date in the date range for which you want to create partitions.
- **p_days_per_part**: The number of days contained in each partition.
- **p_mview_index_tablespace**: The name of the tablespace in which the materialized view index partitions are placed. If a parameter is not supplied, thet materialized view index partitions are placed in the same tablespace as materialized view data. Placing these partitions in the same tablespace as materialized view data can create storage problems if your data tablespace has a large extent size.

CREATE_FACT_PARTITIONS

The CREATE_FACT_PARTITIONS procedure creates partitions on a specific fact table object. Call this procedure if you only want to partition a subset of all fact table objects within the database. This procedure provides finer control over creating partitions for specific facts.

The syntax, an example, and the parameters of the call used to invoke the CREATE_FACT_PARTITIONS procedure appear below:

Syntax

```
procedure create_fact_partitions(
    p_fact_name               IN     VARCHAR2,
    p_fact_tablespace         IN     VARCHAR2,
    p_first_date              IN     DATE,
    p_date_less_than          IN     DATE,
```

```
    p_days_per_part           IN     NUMBER,
    p_fact_index_tablespace   IN     VARCHAR2 default null
);
```

Procedure

```
begin
    clk_mnt_partitions.create_fact_partitions(
        p_fact_name=>'CLK_IMPRESSION_FACT',
        p_fact_tablespace=>'CDATA',
        p_first_date=>to_date('01-NOV-2001','DD-MON-YYYY'),
        p_date_less_than=>to_date('30-NOV-2001','DD_MON_YYYY'),
        p_days_per_part=>1,
        p_fact_index_tablespace=>'CFACTINDX'
        );
end;
```

Parameters

- **p_fact_name**: The name of the fact object you want to partition.
- **p_fact_tablespace**: The name of the tablespace in which the created fact partitions are placed. The default value is CDATA.
- **p_first_date**: The beginning date in the date range for which you want to create partitions.
- **p_date_less_than**: The ending date in the date range for which you want to create partitions.
- **p_days_per_part**: The number of days contained in each partition.
- **p_fact_index_tablespace**: The name of the tablespace in which the fact index partitions are placed. If a parameter is not supplied, the fact index partitions are placed in the same tablespace as fact data. Placing these partitions in the same tablespace as fact data can create storage problems if your data tablespace has a large extent size.

CREATE_MVIEW_PARTITIONS

The CREATE_MVIEW_PARTITIONS procedure creates partitions on a specific materialized view object. Call this procedure if you only want to partition a subset of all materialized view objects within the database. This procedure provides finer control over creating partitions for specific materialized views.

The syntax, an example, and the parameters of the call used to invoke the CREATE_MVIEW_PARTITIONS procedure appear below:

Syntax

```
procedure create_mview_partitions(
    p_mview_name              IN     VARCHAR2,
    p_mview_tablespace        IN     VARCHAR2,
    p_first_date              IN     DATE,
    p_date_less_than          IN     DATE,
    p_days_per_part           IN     NUMBER,
    p_mview_index_tablespace  IN     VARCHAR2 default null
  );
```

Example

```
begin
    clk_mnt_partitions.create_mview_partitions(
        p_mview_tablespace=>'CSUMM',
        p_mview_name=>'CLK_SL_SESSION_MV',
        p_first_date=>to_date('01-NOV-2001','DD-MON-YYYY'),
        p_date_less_than=>to_date('30-NOV-2001','DD_MON_YYYY'),
        p_days_per_part=>1,
        p_mview_index_tablespace=>'CMVIEWINDX'
        );
end;
```

Parameters

- **p_mview_name**: The name of the materialized view object you want to partition.
- **p_mview_tablespace**: The name of the tablespace in which the created materialized view partitions are placed. The default value is CSUMM.
- **p_first_date**: The beginning date in the date range for which you want to create partitions.
- **p_date_less_than**: The ending date in the date range for which you want to create partitions.
- **p_days_per_part**: The number of days contained in each partition.
- **p_mview_index_tablespace**: The name of the tablespace in which the materialized view index partitions are placed. If a parameter is not supplied, thet materialized view index partitions are placed in the same tablespace as materialized view data. Placing these partitions in the same tablespace as materialized view data can create storage problems if your data tablespace has a large extent size.

Running the Database Installation Wizard

This section explains how to use the **Database Installation Wizard** to install the Clickstream Intelligence database schemas. In order to use Clickstream Intelligence with a separate Oracle9*i* Enterprise Edition database, you must use the Database Installation Wizard to create two database schemas: `clkana` (for Clickstream Analytics) and `clkrt` (for the Runtime Administrator). Both schemas provide all necessary database objects to run Oracle9*i*AS Clickstream Intelligence. The Database Installation Wizard also modifies the database connection information contained in the `click-app.xml` file. After running the Database Installation Wizard, the Clickstream Intelligence Runtime Administrator will connect to the customer database you just installed (and not the metadata repository installed with Oracle9*i* Application Server.

Note: Your Clickstream database, or *customer database*, must already be running prior to using the Database Installation Wizard.

Launching the Database Installation Wizard

How to Launch the Database Installation Wizard

1. Connect to the Oracle9*i* Application Server host.
2. At the command prompt, enter the appropriate command:

   *(***UNIX***)* *ORACLE_HOME*/click/bin/clkinstall --click-home=*ORACLE_HOME*/click
   '

   *(***Windows***)* *ORACLE_HOME*/click/bin/clkinstall.bat --click-home="*ORACLE_HOME*/click"

   In the expression above, *ORACLE_HOME* represents the location in which the customer database was installed.

The Welcome Page

The **Welcome** page provides information about the steps needed to install information into a Clickstream Intelligence database. To proceed, click **Next.**

The following function buttons appear on all Database Installation Wizard pages.

- **Cancel**: Quits the installation and closes the Wizard.

- **Help**: Accesses a help file that explains the functionality of each page.
- **Back**: Enables you to view the previous page.
- **Next**: Enables you to view the next page.

The Install/Deinstall Page

The **Install/Deinstall** page enables you to either create a Oracle9iAS Clickstream Intelligence schema within the database or drop pre-existing schemas.

Select a schema option from the choices below and click **Next**.

- **Create a new Oracle9iAS Clickstream Intelligence schema within the database:** This option creates the dedicated Clickstream database instance.
- **Delete pre-existing Clickstream Intelligence schemas from the database:** This option permanently removes all Clickstream Intelligence schemas from the database.

The SYS Logon Page

The **SYS Logon** page enables you to set all parameters needed for the DBA to gain access to the database.

Enter the appropriate password, host name, port, and Oracle SID information and click **Next**.

- **Sys Password**: The password that will grant a DBA access to the database.
- **Host Name**: The name of the computer on which the database resides (for example, DEDHOST)
- **Port Number**: The port number the database uses to listen (for example, 1521)
- **Oracle SID**: The unique system identifier for the database (for example, dw80pk)

The Runtime Administrator Schema Page

The **Runtime Administrator Schema** page enables you to set and confirm the password used by a System Administrator to access data stored in the Runtime Administrator schema (CLKRT) created by the Wizard for this database. The CLKRT schema defines within the database all the information necessary to create and manage the Oracle9*i*AS Runtime Administrator.

Enter the password associated with the Runtime Administrator schema and click **Next**. There are two fields you must fill in on this page:

- **Password**: Create a password. This is the password that the System Administrator will use to access or configure the Runtime Administrator.
- **Confirm Password**: The same password as entered in the **Password** field.

The Analytics Schema Page

The **Analytics Schema** page enables you to set and confirm the password used by a System Administrator and end users to access data stored in the Analytics schema (CLKANA). The CLKANA schema defines within the database all the information necessary to create and manage Oracle9*i*AS Clickstream Intelligence Analytics (Clickstream Analytics).

Enter the password associated with the Viewer schema and click **Next**. There are two fields you must fill in on this page:

- **Password**: Create a password. This is the password that users of Oracle9*i*AS will use to access Clickstream Analytics.
- **Confirm Password**: The same password as entered in the **Password** field.

The Language Selection Page

The **Language Selection** page enables you to select the language used for both the Runtime Administrator and Clickstream Analytics.

1. Select a language from the Runtime Administrator drop-down list.
2. Select a language from the Clickstream Analytics drop-down list.
3. Click **Next**.

The Tablespace Option Page

Use the **Tablespace Option** page to select the way in which you will set up the tablespaces required when using Oracle9*i*AS. After making a selection, click **Next**.

- Select **Typical** if you want use the Database Installation Wizard to set up tablespace parameters for your Clickstream database. If you select **Typical**, the **Database Sizing Parameters** page appears next to help you size and name the tablespaces.

- Select **Custom** if you have already installed an Oracle9*i* Enterprise Edition database and created tablespaces within it. When you select **Custom**, the Database Installation Wizard immediately displays the **Tablespaces Selection** page.

Note: Remember that if you select **Custom**, you can go to the "The Tablespaces Selection Page" section of this chapter for further instructions. You do not need to read the following sections:

- **The Database Sizing Parameters Page**
- **The Suggested Tablespace Size Page**
- **The Datafile Directory Page**
- **The Temporary Tablespace Page**

The Database Sizing Parameters Page

This page appears if you selected **Typical** on the **Tablespace Option** page. Use the **Database Sizing Parameters** page to configure the tablespaces in your Clickstream database. The Database Installation Wizard utilizes a mathematical formula (as described in the "Tablespace Sizing Formula" section of this appendix) to determine the appropriate size for all tablespaces. When you have provided information for the required fields (described below), click **Next**.

- **What is the Average Daily Number of Page Requests Served by All Your Web Sites**: Enter the average number of requests served by your Web site per day. The number you provide enables size estimatation for the Warehouse Data tablespace, the Warehouse Index tablespace, and the Warehouse Summary Layer tablespace.
- **What is the Maximum Daily Number of Page Requests Served by All Your Web Sites**: Enter the maximum number of requests served by your Web site per day. The value you provide enables the size of the Warehouse Stage tablespace to be estimated by the Database Installation Wizard.
- **How Many Days of Web Site Data Will be Stored in the Clickstream Database**: Enter the number of days Web log data should be stored in the database before it is archived. The number you supply enables size estimation for the Warehouse Data tablespace, the Warehouse Index tablespace, and the Warehouse Summary Layer tablespace.

- **How Often Will You Load Web Site Data into the Clickstream Database**: Enter the number of days the Clickstream Loader should wait before loading new data into the database. If you want to load data more than once a day, enter the appropriate decimal fraction, such as .5. The number you supply enables the Database Installation Wizard to estimate the size of the Warehouse Stage tablespace.

The Suggested Tablespace Size Page

The **Suggested Tablespace Size** page enables you to review all tablespaces that will be created by the Database Installation Wizard. To adjust any of the calculated tablespace sizes, enter a new value in the appropriate text field. After you've reviewed all tablespace values, click **Next**.

Note: This page appears only if you selected **Typical** on the **Tablespace Option** page.

The Database Installation Wizard calculates size values for the following tablespaces:

- **Analytics tablespace (CANA)**: Provides storage for the Discoverer End User Layer (EUL), as well as Discoverer workbooks and worksheets.
- **Warehouse Audit tablespace (CAUDT)**: Provides storage for historical audit records and details about previously-run processes and jobs.
- **Warehouse Data tablespace (CDATA)**: Provides storage for all dimensional data and serves as the default tablespace for fact records (if fact table partitions are not created).
- **Warehouse Index tablespace (CINDX)**: Provides storage for dimension indexes and serves as the default tablespace for fact table bitmap indexes (if fact table partitions are not created).
- **Warehouse Stage tablespace (CSTGE)**: Provides storage for incoming data as it is transformed and loaded into the database.
- **Warehouse Summary Layer tablespace (CSUMM)**: Provides storage for non-partitioned summary layer tables and materialized views, and serves as the default tablespace for partitioned summary layer tables and materialized views.

The Datafile Directory Page

The **Datafile Directory** page enables you to select the directory in which you want the Database Installation Wizard to create data files for the customer database. To designate a storage location, enter the path for an existing directory on the database host and click **Next**.

The Temporary Tablespace Page

The **Temporary Tablespace** page enables you to create a temporary tablespace within the database. to do this, select a temporary tablespace from the drop-down list, specify the requested parameters, and click **Next**.

Note: This page appears only if you selected **Typical** on the **Tablespace Option** page.

The Tablespaces Selection Page

The **Tablespaces Selection** page enables you to assign roles to the tablespaces you created in the database. To do this, use the drop-down menu under each tablespace option to select the appropriate tablespaces. When you are finished, click **Next**.

Note: This page appears only if you selected **Custom** from the **Tablespace Option** page. The tablespaces that appear are identical to those listed in the "The Suggested Tablespace Size Page" section of this appendix.

The Summary Page

The **Summary** page enables you to review all the settings before installation of the Clickstream Intelligence schemas. These settings include:

- System
- DBA User
- New Runtime Administrator Schema
- New Viewer Schema

- Warehouse Tablespace Configurations

Review the summary, and then click **Finish** to begin the installation. If you want to modify settings, click **Back** to return to the appropriate page(s).

Installation takes about 40 minutes. During this time, the progress bar moves quite slowly - this is normal behaviour. Do not stop the installation due to a lack of feedback or status information. After the Database Installation Wizard completes, you can load data into the database. For more information, see Chapter 4, "Loading the Clickstream Database".

Database Installation Log

To view details related to customer database installation, reference the `install-click.log` file located in the following directory:

- **(UNIX)** *`ORACLE_HOME`*`/click/log`
- **(NT)** *`ORACLE_HOME`*`\click\log`

The variable *`ORACLE_HOME`* refers to to the directory in which Oracle9*i*AS was installed.

C

Installing a Standalone Collector Agent

This appendix describes how to install a standalone Clickstream Intelligence Collector Agent.

This appendix contains the following sections:

- **Introduction to the Collector Agent**
- **Installing a Standalone Collector Agent**
- **Configuring the Collector Agent**
- **How to Start and Stop the Collector Agent**

Introduction to the Collector Agent

The **Collector Agent** is the component of Oracle9*i*AS Clickstream Intelligence that resides on the Web server machine and collects Web server logs files. The Collector Agent compresses log files into units of manageable size called **data packets**, which enable more efficient storage of Web log data.

If you want to collect Web logs from a Web server other than Oracle HTTP Web Server, a **standalone Collector Agent** must be installed on your server machine. Data packets created by the standalone Collector Agent are retrieved by the Collector Server using HTTP or FTP protocol, and then parsed and loaded into the database by the Clickstream Loader**.** The platform of the server on which you install the standalone Collector Agent can differ from the platform on which you installed Oracle9*i*AS. (See "Installing a Standalone Collector Agent" in this appendix for further details.

To help determine if a standalone Collector Agent is needed for your Web server, see the criteria below. If any of the following criteria are met, you do *not* need to install a standalone Collector Agent:

- **Data is collected from a local data source:** installation of a standalone Collector Agent is not required because all log files reside on the same machine on which you installed Oracle9*i*AS Clickstream Intelligence. For more information about **Data Sources**, see Chapter 3, "Configuring Clickstream Intelligence".
- **Log files are collected from Oracle HTTP Server**: when Oracle HTTP Server is installed as part of Oracle9*i*AS core install, a Collector Agent is automatically installed to collect your Web log files (and can be managed via the Oracle Enterprise Manager (OEM) Management Console).

Note: All log files from which the Collector Agent creates data packets must be **unzipped** (uncompressed). If a log file is zipped, the Collector Agent will still create data packets from it - which ultimately causes failure of the "Load Clickstream" process.

Installing a Standalone Collector Agent

During a typical Oracle9*i*AS Clickstream Intelligence install, the Collector Agent is pre-configured to retrieve Apache Web logs from Oracle HTTP Server. To install one or more Collector Agents for data sources *other than* Oracle HTTP Server, you must follow the installation steps described in the sections that follow.

Pre-installation Requirements

In order to successfully install, configure, and use a standalone Collector Agent, the following conditions must be met:

- **Java Runtime Environment (JRE) 1.3.1** or higher must be installed on your computer to enable use of the Collector Agent's install files.
- Approximately **10 MB** of disk space is required to enable successful installation.

Note: Data packets consume disk space on the server machine; therefore, disk space requirements will increase as the number of data packets stored on the server increases.

Installation

The platform of the server on which you install the Collector Agent can be the same as, or may differ from, the platform of your Oracle9*i*AS CD. In both cases, installation files can be downloaded from the Oracle Technology Network Web site.

1. Go to the Oracle Technology Network Web site located at:

   ```
   http://otn.oracle.com/
   ```

2. Download the standalone Collector Agent install files that match the platform of your server.

3. Next, unpack the Collector Agent files in accordance with the platform you are using. (For example, .tar or .zip files correspond to UNIX or Windows, respectively.)

4. Go to the appropriate directory below:

   (UNIX) *INSTALL_DIR*/click/utl

   (Windows) *INSTALL_DIR*\click\utl

5. Then, enter the appropriate command below to create the standalone Collector Agent configuration files:

   (UNIX) ./instantiate-collagent *JDK_HOME COLAGENT_PACKET_HOME COLAGENT_PORT COLAGENT_LOGS APACHE_LOGS*

   (Windows) .\instantiate-collagent-java *JDK_HOME COLAGENT_PACKET_HOME COLAGENT_PORT COLAGENT_LOGS APACHE_LOGS*

   In the expressions above, substitute values for the appropriate environment variables below:

   - *INSTALL_DIR* - represents the directory into which the standalone Collector Agent was unpacked
   - *JDK_HOME* - represents the Java Development Kit home directory
   - *COLAGENT_PACKET_HOME* - the data packets destination home directory
   - *COLAGENT_PORT* - the port corresponding to the Collector Agent you are installing
   - *COLAGENT_LOGS* - location of the Collector Agent log files
   - *APACHE_LOGS* - location of the Web server log files you want to collect

6. Go to the appropriate directory:

   (UNIX) *INSTALL_DIR*/click/bin

   (Windows) *INSTALL_DIR*\click\bin

   The *INSTALL_DIR* environment variable represents the directory in which you unpacked the standalone Collector Agent.

7. Now, you can execute any of the commands listed in the following section of this appendix: "How to Start and Stop the Collector Agent".

Configuring the Collector Agent

Upon installation and instantiation of a standalone Collector Agent, all configuration file attributes are set to default values . If you want to modify the default values, you can use a text editor (such as Notepad) to manually configure the following files:

(UNIX) *INSTALL_DIR*/click/conf/collector-agent.properties

INSTALL_DIR/click/conf/agent.xml

(Windows) *INSTALL_DIR*\click\conf\collector-agent.properties

INSTALL_DIR\click\conf\agent.xml

The following sections describe all attributes associated with the standalone Collector Agent configuration files.

The collector-agent.properties File

The `collector-agent.properties` file contains name=value pairs. You can configure the following attributes:

- **loggingHome**: the directory in which you want the Collector Agent to log information.
- **collectorPort:** the port on which the Collector Agent will communicate. This port must be used by Clickstream Intelligence only.

The agent.xml File

The `agent.xml` file may contain several entries, wherein each entry corresponds to a specific data source. All configurable attributes for each entry are listed below.

- **logFileLoc**: the location of the log files that need to be processed by the Collector Agent.
- **packetSize**: the size, in lines, of the data packets that the Collector Agent will create.
- **packetHome**: the location in which the completed (compressed) data packets are placed after they are created from the log files.
- **timeToSleep**: the time (milliseconds) that elapses between the intervals in which the Collector Agent checks for new Web log files to compress into data packets.
- **packetTimeout**: the length of time, in days, after which data packets are flushed from the packet directory on the Web server.
- **filterPattern**: the pattern to use when targeting the files on the Web server you want to process. It is important to note that the filter pattern is used to *select* (and not exclude) the files you want to process from the Web server. Data packets are created from log data in the files that are selected according to your filter pattern definition.

  For example, if you want to process from the Web server only the files that contain "dms" in the file name, then enter *dms* as your filter pattern identifier (the * represents a wild-card character).

Note: The value of **packetHome** must be exclusive (unique) among all standalone Collector Agents.

How to Start and Stop the Collector Agent

To start, stop, and view the status of one or more Collector Agents, use the commands listed in the following sections. For each command, note the following variable definitions:

- ***INSTALL_DIR***: the directory in which the standalone Collector Agent is installed.

- ***agentName***: the name of the standalone Collector Agent to which the command pertains.

Commands for Specific Collector Agents

The following commands apply only to a specific Collector Agent:

- **Start a Collector Agent:**

  (UNIX) *INSTALL_DIR*/click/bin/clkagtctl start *agentName*

  (Windows) *INSTALL_DIR*\click\bin\clkagtctl start *agentName*

- **Stop a Collector Agent:**

  (UNIX) *INSTALL_DIR*/click/bin/clkagtctl stop *agentName*

  (Windows) *INSTALL_DIR*\click\bin\clkagtctl stop *agentName*

- **View the Status of a Specific Agent:**

  (UNIX) *INSTALL_DIR*/click/bin/clkagtctl status *agentName*

  (Windows) *INSTALL_DIR*\click\bin\clkagtctl status *agentName*

Commands for All Collector Agents

The following commands pertain to all Collector Agents in your system:

- **Start All Agents:**

  (UNIX) *INSTALL_DIR*/click/bin/clkagtctl startAll

  (Windows) *INSTALL_DIR*\click\bin\clkagtctl startAll

- **Stop All Agents:**

  (UNIX) *INSTALL_DIR*/click/bin/clkagtctl stopAll

  (Windows) *INSTALL_DIR*\click\bin\clkagtctl stopAll

- **View the Status of the Collector Agent Manager:**

  (UNIX) *INSTALL_DIR*/click/bin/clkagtctl managerStatus

  (Windows) *INSTALL_DIR*\click\bin\clkagtctl managerStatus

D

Error Messages

This appendix lists error numbers and messages that may appear when using the Clickstream Intelligence Runtime Administrator.

This appendix contains the following topics:

- **Generic Clickstream Errors**
- **Clickstream Loader Errors**

Generic Clickstream Errors

CLK-10000: invalid site code
CLK-10000: invalid site code
CLK-10001: invalid site id
CLK-10002: invalid or inappropriate table name
CLK-10003: specified date does not exist in the CLK_DATE dimension
CLK-10004: a non-null parameter was passed in as null
CLK-10005: invalid tablespace
CLK-10016: invalid load id
CLK-10021: null value found where data expected
CLK-10024: invalid flag
CLK-10026: invalid column name
CLK-10063: invalid message id
CLK-10064: wrong number of parameters to message format
CLK-10100: no initial partition created
CLK-10101: creating partitions would require splitting more than one existing partition
CLK-10102: invalid partition date range

CLK-10103: days per partition is greater than date range
CLK-10104: cannot drop maxvalue partition for site if other site partitions exist
CLK-10105: only the initial partition exists, cannot continue validation
CLK-10106: load date exceeds the highest partition for this table
CLK-10107: invalid partition key
CLK-10108: specified materialized view not found
CLK-10150: open jobs exist, cannot continue until active jobs are closed
CLK-10151: invalid job type
CLK-10152: job failure
CLK-10153: there is a process already running
CLK-10154: process is not resumable
CLK-10155: there is already a pending request
CLK-10156: execution engine is not running
CLK-10157: process definition does not exist
CLK-10158: there is no process currently running
CLK-10159: process does not exist
CLK-10160: process with this status cannot be rolled back
CLK-10200: operation is locked, another process is executing the operation
CLK-10220: field value already exists
CLK-10221: invalid parameter
CLK-10222: the data has been modified since read
CLK-10223: site does not exist
CLK-10224: data format does not exist
CLK-10225: resource does not exist - {0}
CLK-10226: deployment session has blocked
CLK-10227: lock time-out while requesting lock
CLK-10228: data source does not exist
CLK-10229: data filter does not exist
CLK-10300: invalid refresh type
CLK-10301: invalid summary type
CLK-10302: summary does not exist
CLK-10303: summary already exists in database
CLK-10304: invalid summary name
CLK-10305: invalid time period
CLK-10306: invalid summary query

CLK-10350: invalid mapping name
CLK-10370: invalid data packet status code
CLK-10371: invalid data packet status
CLK-10372: cannot delete a packet while a process is running
CLK-10373: packet(s) not found
CLK-10999: no message
CLK-10998: internal application error

Clickstream Loader Errors

CLKLDR-00000: normal, successful completion
CLKLDR-00001: unspecified error
CLKLDR-00010: OCI call failed
CLKLDR-00011: user requested cancel of current operation
CLKLDR-00012: missing or invalid password
CLKLDR-00013: invalid username/password; logon denied
CLKLDR-00014: data conversion error
CLKLDR-00015: direct path load error
CLKLDR-00020: data value was truncated
CLKLDR-00030: query string or cookie parameter with given name not found
CLKLDR-00031: query string or cookie parameter is not properly terminated
CLKLDR-00032: query string or cookie parameter offset is out of bounds
CLKLDR-00033: parameter name is empty
CLKLDR-00040: time format not recognized
CLKLDR-00041: a non-numeric character was found where a numeric was expected
CLKLDR-00042: not a valid year
CLKLDR-00043: not a valid month
CLKLDR-00044: not a valid day
CLKLDR-00045: not a valid hour
CLKLDR-00046: not a valid minute
CLKLDR-00047: not a valid second
CLKLDR-00048: not a valid time zone offset hour
CLKLDR-00049: not a valid time zone offset minute
CLKLDR-00050: time string does not match format
CLKLDR-00051: time format ends before converting entire string

CLKLDR-00060: filename exceeds maximum length
CLKLDR-00061: filename does not contain an extension
CLKLDR-00062: cannot open
CLKLDR-00063: cannot close file
CLKLDR-00064: cannot make directory
CLKLDR-00070: no records from the data packet were loaded
CLKLDR-00071: no data packets are available to load
CLKLDR-00080: field terminator not found
CLKLDR-00081: unexpected field terminator after last field
CLKLDR-00082: unexpected characters after last field
CLKLDR-00083: left field enclosure not found
CLKLDR-00084: right field enclosure not found
CLKLDR-00085: no delimiter found after method
CLKLDR-00086: no delimiter found before protocol
CLKLDR-00087: invalid request URI
CLKLDR-00088: matched filter
CLKLDR-00090: one or more data packets failed to load
CLKLDR-00091: error while creating worker process or thread
CLKLDR-00600: internal error code

E

Metadata Repository Postinstallation and Configuration

This appendix lists the post-installation tasks that enable you to begin using Oracle9*i*AS Clickstream Intelligence with the metadata repository provided with Oracle9*i* Application Server (Oracle9*i*AS).

This appendix contains the following topics:

- **Metadata Repository Post-Installation Tasks**
- **Installing the Metadata Repository in a non-English Language**

Metadata Repository Post-Installation Tasks

Follow the steps below to begin using Oracle9*i*AS Clickstream Intelligence with the infrastructure database provided with Oracle9*i* Application Server. Platform-specific instructions are embedded within the text, where applicable.

1. Use the Oracle9*i*AS Universal Installer to install Oracle9*i*AS Clickstream Intelligence. For Oracle9*i*AS installation details, see the *Oracle9i Application Server Installation Guide*.
2. Install the appropriate Patchset for your operating system. To do this, go to the following URL:

   `http://metalink.oracle.com`

   Following successful login with a valid username and password, click **Patches** from the menu on the left side of your screen. Provide the following **Patch Download** query data. Fields not listed below require no modification; leave the default values in the fields provided.

   - **Product Family:** Oracle Server

- **Release:** 9.0.1.3
- **Platform:** (select the same platform as your Oracle9*i* database)
- **Language:** (select the language that matches the one you selected during Oracle9*i* database installation.)
- **Limit Search To:** All Product Patches

Click the **Submit** button. From the list of patches displayed, click the link for one of the patches listed below. Download the patch as specified in the *ReadMe* file.

- **(UNIX)** Oracle 9.0.1.3 Patch Set for Oracle Data Server.
- **(Windows)** Oracle9*i* 9.0.1.3.1 Patch 4 for Windows NT and Windows 2000.

Note: If you are installing the metadata repository in a language other than English, go to the "Installing the Metadata Repository in a non-English Language" section of this appendix and perform those tasks *before* performing the remaining installation tasks below.

---

3. Use Oracle Enterprise Manager (OEM) to modify the pre-generated passwords that were automatically created during installation. To learn more about changing the `clkrt` and `clkana` schema passwords, see the "Changing Component Schema Passwords" section of Chapter 10, "Reconfiguring the Application Server," in *Oracle9iAS Admimistrator's Guide*.
4. Continue post-installation tasks as described below. The following steps require use of Oracle9*i* Discoverer Administrator, which is part of Oracle9*i* Developer Suite (Oracle9*i*DS).

   - Create an End User Layer (EUL) with the Oracle9*i* Discoverer Administrator **EUL Manager Dialog**. Instructions for creating an empty EUL can be found in the *Oracle9i Discoverer Administrator Administration Guide*.
   - Use the Oracle9*i* Discoverer Administrator Import Wizard to populate the EUL with the Clickstream EUL objects contained in the `clickstream_intelligence_eul_full.eex` file. This file is located in the appropriate directory below:

     (UNIX) *ORACLE_HOME*`/install/analytics/discoverer/`*LANG*

(Windows) *ORACLE_HOME*\install\analytics\discoverer*LANG*

ORACLE_HOME is the directory in which Oracle9*i*AS is installed, and the *LANG* variable is the code for the language selected during Oracle9*i*AS installation.

For information about populating the EUL with Clickstream metadata, see the *Oracle9i Discoverer Administrator Administration Guide*.

5. Access the **Clickstream Intelligence Home** page at the following URL:

   http://*HOST:PORT*/click/

   The *HOST* is the name of the computer on which Clickstream Intelligence is installed, and the *PORT* is the primary port on which the Web server listens for HTTP requests.

6. Log on to the Runtime Administrator with the username **clkrt** and the password you defined via OEM.

7. Define and configure the Runtime Administrator parameters that specify the way you want data to be acquired and processed from Web server log files. For additional information about the Runtime Administrator, see the *Oracle9iAS Clickstream Intelligence Administrator's Guide*.

8. Load Web server log file data into the infrastructure database. For information about the **Load Clickstream** process and other database processes, see Chapter 4, "Loading the Clickstream Database".

9. Enable viewing of Clickstream Analytics by creating a "Private" database connection. To do this, go to the **Clickstream Intelligence Home** page and click the **Analytics- Discoverer Viewer** link. The **Connect to Discoverer Viewer** page appears.

10. Click **Create Connection**. The **Connection Details** page appears. Provide the Connection Name, Connection Description, and Locale (the language used to view Clickstream Analytics).

11. Provide the following Database Connection Details:

    - **User Name:** enter **clkana**.
    - **Password:** the clkana schema password you defined via OEM.
    - **Database:** the TNS alias for the Oracle9*i*AS metadata repository.

12. Click **Apply**. You return to the **Connect to Discoverer Viewer** page.

13. To access Clickstream Analytics, click the link for the name of the (private) connection you created.

Installing the Metadata Repository in a non-English Language

When you want to install the metadata repository in a language that is not English, run the `clkinstall` program immediately after Oracle9*i*AS installation. This script is located in the appropriate directory below, where the *ORACLE_HOME* variable represents your Oracle9*i*AS home directory.

(UNIX) *ORACLE_HOME*`/bin`

(Windows) *ORACLE_HOME*`\bin`

To select the language for the Clickstream Intelligence Runtime Administrator and Clickstream Analytics, enter the following command:

```
clkinstall seed --click-home=CLICK_HOME --hostname=HOST --sid=SID
--port=PORT --rtadmin-lang=RTADMIN_LANG --analytics-lang=ANALYTICS_LANG
```

In the expression above, the *CLICK_HOME* variable is equivalent to:

(UNIX) *ORACLE_HOME*`/click`

(Windows) *ORACLE_HOME*`\click`

Additionally, the *HOST* is the name of the machine on which the metadata repository is installed, *SID* is the database system ID, *PORT* is the port number of the database TNS listener, *RTADMIN_LANG* is the language you want to use for the Clickstream Intelligence Runtime Administrator, and *ANALYTICS_LANG* is the language in which you want to view Clickstream Analytics.

Note: You must run the `clkinstall` program before changing the SYS password in the metadata repository, since the program uses the default password **change_on_install** when connecting to the database. If you have already changed the SYS password, you must change it back to the default password, run the `clkinstall` program, and then re-set your password again.

Glossary

This glossary defines terms and concepts used in the *Oracle9iAS Clickstream Intelligence Administrator's Guide*.

ad hoc

Intermittent, spontaneous access to the database.

ad-hoc analysis

A type of analysis in which questions are answered by manipulating the dimensions, dimension values, and layout of data in a database.

agent

The browser or other application that makes a request (typically to a Web server).

In a log file entry, the identification string and number that identifies each Web browser request. The *agent* string typically identifies the browser, its version number, and the operating system upon which it is running.

Apache

Widely used open-source Web server software available from the Apache Group.

authentication

The process of verifying the identity of a registered user, often as a prerequisite for allowing access to system resources. A visitor to a Web site, for example, may be required to supply a valid username and password as part of the *authentication* process.

See also: user.

bitmap index

A specialized form of index that indicates the existence or non-existence of a record with a series of ones and zeros.

business intelligence

Any computerized process used to extract or analyze business data for the purpose of indicating the historical performance, current status, and possible forecast for a company. *Business intelligence* may be used to increase a company's competitive advantage, develop innovative business solutions, or create solutions that focus company efforts on projects with the highest return on investment.

cache

A temporary storage area in computer memory where frequently-accessed data is be stored for fast access.

child

A value at the level below a particular value in a hierarchy. This value can be a *child* for more than one parent if the child value belongs to multiple hierarchies.

See also: dimension, hierarchy, level.

clickstream

The series of clicks, or page transitions, that mark an individual's navigational path through the Internet or a particular Web site.

Clickstream Analytics

The analytical reports created from data that Clickstream Intelligence acquires from your Web site(s). The reports (called worksheets) that comprise *Clickstream Analytics* can be viewed with Oracle9*i*AS Discoverer.

Clickstream Loader

The component of Oracle9*i*AS Clickstream Intelligence that parses, filters, and transforms Web server log files before loading them in to Oracle9*i* database. The *Clickstream Loader* resides on the same machine as the Collector Server, the component from which uncompressed Web log files are obtained.

Clickstream site

A Web site for which Oracle9*i*AS Clickstream Intelligence provides analytics. For each *Clickstream site*, you define the parameters that control the way clickstream data is acquired and processed.

client

A user, software application (such as a browser), or computer that requests and relies upon the services, data, or processing of another application or computer (the server). The *client* is usually the machine that runs your Internet browser.

In a client-server architecture, the *client* is the front-end portion that sends a request to the back-end, or server, portion. The server processes the request and sends a response to the client.

See also: server.

client-server

The architecture in which several personal computers or workstations (*clients*) are linked to one or more large processors (*servers*).

Collector Agent

The component of Oracle9*i*AS Clickstream Intelligence that gathers and compresses Web server log files into units of manageable size, called data packets. The *Collector Agent* is installed on the Web server machine.

See also: Collector Server.

Collector Server

The component of Oracle9*i*AS Clickstream Intelligence that retrieves data packets from the Web server and uncompresses the log files for subsequent transfer to the Clickstream Loader. The *Collector Server* resides on the same machine as the Clickstream Loader.

cookie

The text string stored onto the client browser by the server during an HTTP request. *Cookies* enable an application Web server to retrieve information about a client, such as the domain, path, and other session variables. When a Web browser is requested to send cookie information to the server, individual Web site users can be recognized again by the unique *cookie* that was originally assigned to them.

crawler

See spider.

database

A collection of data that is treated as a unit. The purpose of a database is to store and retrieve related information.

data filter

A conditional or rule-based routine for restricting data; a method of selecting or qualifying data, typically from a larger data set. A Clickstream *data filter* enables you to exclude specific Web log records from those which are loaded into the database. All filtered records appear as "Lines Discarded" on the **Data Packets Detail**s page.

data format

The structure of the fields and field names in a Web log. Apache, W3C Extended, and Microsoft IIS Extended are standard *data formats* supported by Clickstream Intelligence.

data packet

A grouping of Web server log files. The Clickstream Collector Agent compresses log files into *data packets* to facilitate transfer from the Web server to the Collector Server. The data packets are then uncompressed by the Collector Server.

See also: Collector Agent, Collector Server.

data source

A database, file, or repository that provides information to the database. The *data source* specified for a Clickstream site, for example, indicates the location of log files that are loaded into the Clickstream database.

data type

A category of data, such as a character, string, integer, or date. In a relational database, data has only one data type assigned to it.

database

A relational database that is designed for query and analysis. This central repository typically contains historical (static) data, and may contain data from diverse sources. Storage of data in a *database* enables you to consolidate and integrate data from disparate sources, and analyze trends or changes in data over time.

delimiter

A separator in a sequence of values in a Web log file.

See also: enclosure, terminator.

dimension

A structure, often composed of one or more hierarchies, that categorizes data in a database. *Dimension* data (described by dimensional attributes) is typically collected at the lowest level of detail and then aggregated, or "rolled up," into higher levels that comprise hierarchies.

See also: hierarchy.

DNS

See: Domain Name System (DNS).

DNS alias

The numeric IP address that directly corresponds to a Web site's domain name.

DNS lookup

Domain Name Server lookup. The process by which a domain name is translated into the form of an IP address, or DNS alias.

A *DNS lookup* performed on the domain name www.oracle.com, for example, reveals 148.87.9.44 as the corresponding the IP address.

See also: reverse DNS lookup.

domain name

A unique name that identifies a location, or site, on the Internet. It is comprised of a suffix (such as .com or .gov), attached to a hostname.

A *domain name* is the resolved, or user-friendly, version of an IP address.

Domain Name System (DNS)

The system for naming computers and network services that organizes the Internet into a hierarchy of domains. *DNS* identifies each computer within a domain by a unique hostname. For example, a computer named "dms" in the us.oracle.com domain would be uniquely identified on the Internet as dms.us.oracle.com.

A system for naming computers and network services that is organized into a hierarchy of domains. DNS is used in TCP/IP networks to locate computers through user-friendly names. DNS resolves a friendly name into an IP address, which is understood by computers.

dwell time

The amount of time (in seconds) that a user or visitor remains on a Web page during a given session.

e-business

Also called "electronic business." A company that integrates Web technology with company dealings, such as the distribution of goods and services to clients, or collaboration with business partners.

e-business management tools

A set of applications (such as Oracle e-Business Management Tools) that enable centralized management of a complete e-business infrastructure, including the client, middle-tier HTTP servers, and database.

enclosure

Characters that enclose a field in a Web log file. An *enclosure* is a specific type of delimiter.

See also: delimiter.

facts

Alphanumeric data items obtained from a Web log or computed from Web log data. Facts, stored in fact tables, are the products of measures and dimensions - every column in a fact table is either a numeric measure or a foreign key to a dimension. Dimensions, therefore, describe and establish a context for the facts.

Also called a *measure*. Data, usually numeric and additive, that is described by several database dimensions. Values for *facts* are usually not known *a priori*- they are observed and stored.

Certain *facts*, such as averages, totals, or percentages, are generated from existing data through a mathematical operation or data transformation. Examples of *facts* include Sales, Cost, and Profit.

fact table

A table in a star schema that contains two types of columns: those that contain measures (facts) and those that are foreign keys to dimension tables. The primary key of a *fact table* is usually a composite key that is made up of all of its foreign keys.

A *fact table* might contain either detail-level facts or facts that have been aggregated (fact tables that contain aggregated facts are often called summary tables). A fact table usually contains facts with the same level of aggregation.

See also: facts, foreign key, primary key.

foreign key

An integrity constraint that requires each value in a column or set of columns (such as a fact table) to match a value in a related table's *primary key.*

See also: fact table, primary key.

guest user

An individual that is not an anonymous user, yet does not have a specific user entry to a Web site.

hierarchy

A logical structure that uses ordered levels as a means of organizing data and which may be used to define data aggregation. A *hierarchy* can also be used to define a navigational drill path, regardless of whether the levels in the hierarchy represent aggregated totals.

For example: In the Time dimension, a hierarchy might be used to aggregate data from the Month level to the Quarter level to the Year level.

See also: dimension, level.

hit

A single file that is requested from the Web server and transferred to the Web browser. Each item on a page (such as inline images), as well as the page itself, counts as a *hit*.

hostname

The character string that identifies a computer within the DNS domain; the name or IP address of the computer making a request.

impression

A single page view.

See also: session.

internal domain

A specific domain that belongs to a company or organization.

internal site

A site that typically belongs to or pertains to your company or organization.

IP address

A unique numeric code that identifies hosts and networks; a numeric identifier that represents the location of a computer on the Internet. Each computer on a network is assigned one unique *IP address*.

An *IP address* is written as four numbers separated by periods. The IP address 148.87.9.44, for example, corresponds to the oracle.com Web site. The numerical IP address can be translated into a user-friendly domain name via a reverse DNS lookup.

level

A position in a hierarchy. For example, a Time dimension might contain a hierarchy that represents data at the Month, Week, and Day *levels*.

log file

A file that lists certain actions or events that have taken place. For example, a *log file* on a Web server may contain information about all requests that have been made to the server.

materialized view

Summarized data from a fact table in your Clickstream database. A *materialized view* provides access to table data by storing the results of a query in a separate schema object.

metadata

Information (data) that describes data and other structures in a database, such as objects, business rules, and processes.

Metadata may indicate how the data is formatted, how a specific set of data was collected, or when the data was acquired. A repository may contain metadata.

natural key

A unique identifier for an interface table or level table; a group of columns in an interface or level table that uniquely identifies rows in that table.

Natural keys can be conceptualized as "user-defined" primary keys among dimension interface tables.

object

A thing of significance within which information is stored. In a relational database, Tables and Views are the two of the most common *objects*.

Oracle9*i* Application Server

A comprehensive, integrated application server that provides all of the infrastructure and functionality needed to assemble and run a successful e-business.

Oracle9*i* Warehouse Builder

The Oracle product used to build and manage a database.

See also: database.

Oracle architecture

Memory and process structures used by an Oracle server to manage a database.

Oracle Enterprise Manager

A separate Oracle product that combines a graphical console, agents, common services, and tools to provide an integrated and comprehensive systems management platform for managing Oracle products.

packet

A block of information sent over the network each time a connection or data transfer is requested. The information contained in *packets* depends on their type- Clickstream data packets, for example, contain Web log data.

See also: data packet.

platform

The specific underlying computer hardware or software for a system. A *platform* may refer to the combination of computer hardware (such as a specific processor) with a particular operating system.

port

An endpoint to a logical (as opposed to physical) connection; a logical channel by which a client program specifies a particular server program on a computer in a network. Several ports may exist on the same computer.

On the Internet, the type of *port* is often identified by a port number -- the number in a URL that appears after the colon (:), written in the format **hostname:port**. HTTP protocol, for example, uses port 80 by default.

See also: hostname, protocol.

primary key

The column or set of columns included in the definition of a table's *primary key constraint*, which disallows duplicate values and nulls in a column or set of columns.

Primary key values uniquely identify the rows in a table; therefore, only one primary key can be defined for each table.

process

A sequence or series of jobs / actions performed on a database. Clickstream *processes* include loading the warehouse, loading dimensions, refreshing summaries, resolving IP addresses, and restoring the warehouse to a previous version.

process definition

Specifies and characterizes the parameters and values for a process.

See also: process.

protocol

The "language" or set of formal rules used by computers or other devices to send data across a network. Use of *protocols* ensures communication between different programs or computers on a network.

query

The question or specific criteria sent to a database for the purpose of retrieving information. When you *query* the Clickstream warehouse, data is retrieved that fits the specific set of conditions that you submitted.

query string

The dynamic portion of a URL that passes state information or user input from the browser to the server. Typically, the *query string* portion of a URL immediately follows the question mark (?) symbol.

RAID

Redundant **A**rray of **I**ndependent **D**isks. A particular category of disk drives typically used on servers for fault tolerance and performance.

referrer

A Web server or Web page that directs users or visitors to a specific place within a Web site. *Referrers* may be internal or external.

repository

A central location where aggregated data is stored and maintained in an organized fashion. The Clickstream *repository* for Web data is part of your database.

See also: metadata.

resource

A file object hosted by the Web server. Examples of *resources* include static HTML files, CGI programs, and image files.

resource type

The type of physical or logical component that is generated by a resource. *Resource types* may include HTML (files) or GIF (images).

reverse DNS lookup

The process by which an IP address is resolved into its associated user-friendly hostname.

See also: DNS lookup.

roll back

To undo any changes to data that have been performed since a given point in time. Oracle9*i*AS Clickstream Intelligence enables you to *roll back* to a previous version of data stored in the database by clicking the **Undo** button (typically performed when a process has an error or has been temporarily stopped.)

schema

A structure or set of rules that defines the organization of a database. In a relational database, objects - such as the tables, the fields within each table, and their relationship to each other - are defined by a *schema*.

See also: object.

server

In a client / server architecture, the computer that receives, stores, and processes requests that originate from client applications. A *server* handles the functions required for concurrent, shared data access over a network through the use of a specific protocol.

See also: client.

server farm

Several interconnected Web servers used to port content to the Internet.

session

The length of time that measures a single user's activity during a particular visit to a Web site. A collection of impressions; the sequence of requests made by one user to a Web site.

A *session* lasts from the time the user connects until the time the user disconnects or exits the Web site.

A See also: session threshold, session timeout.

session threshold

The period of time after which a given session expires.

session timeout

The point at which a session is terminated due to inactivity for a given period of time.

spider

Also called a *crawler* or "*bot*" (robot). A program that automatically fetches some or all of the Web pages that are referenced from a Web site. Spiders are often used by search engines.

For example, when you register your Web site's URL with a particular search engine, one or more *spiders* automatically index page keywords, as well as all links to pages both within and external to your Web site. When the search engine displays the URL for your site in its query results page, it may also list many of the sites referenced from your Web page if they are relevant to the search criteria.

source

A database, application, file, or other storage facility from which the data in your Clickstream warehouse is derived.

SQL

The acronym for Structured Query Language, the standard query language that is used to update and request information from a database.

staging area

An intermediate database component in which incoming clickstream data is cleaned and prepared for loading into the fact tables.

table

A database object that contains and stores data. Tables are comprised of many columns, each with an associated data type.

terminator

A character that separates fields in a Web log file. A *terminator* is a specific type of delimiter.

URI

Universal Resource Indicator.

See also: URL.

URI Stem

The portion of a URL that appears after the host and port, but precedes the query string. The *URI stem* typically refers to a directory that is accessed, relative to the Web server's root directory.

For example, in the URL **http://www.oracle.com/test.jsp?hello=y** the URI stem is **/test.jsp**

URL

Uniform Resource Locator. The global addressing standard that is used to locate pages on the World Wide Web.

URLs are used by browsers to navigate the World Wide Web and consist of a protocol prefix, port number, domain name, (sub)directory name, and file name.

For example- http://technet.oracle.com:80/tech/xml/index.htm specifies the location and path a browser will travel to find Oracle Technology Network's XML site on the World Wide Web.

user

An individual who has registered with and been authenticated by a Web site. An individual who successfully logs on and is granted access to a Web site.

user agent

Software to access Web content, including desktop graphical browsers, text browsers, voice browsers, mobile phones, multimedia players, plug-ins, and some software assistive technologies used in conjunction with browsers such as screen readers, screen magnifiers, and voice recognition software.

user Interface (UI)

The combination of menus, screens, keyboard commands, mouse clicks, and command language that defines how a user interacts with a software application.

The means with which a user interacts and uses a computer or computer program.

For example: Upon accessing Clickstream Analytics, a user may use buttons, links, drop-down menus and other UI components to produce reports, set up a site, or access other functionality provided by Clickstream Intelligence.

view

A customized presentation of data from one or more tables. A *view* can be conceptualized as a "stored query." Views do not actually contain or store data; they derive their data from the tables on which they are based.

Like tables, views can be queried, updated, inserted into, and deleted from - with some restrictions. All operations performed on a view affect the base tables of the view.

visitor

An unauthenticated user. V*isitors* are often first-time users of a Web site.

See also: authentication.

workbook

A collection of worksheets in Oracle9*i*AS Discoverer. A *workbook* contains related data that is organized to show different perspectives. Each organized subdivision of data, or perspective, is represented by a worksheet.

worksheet

A *worksheet* contains specific data that you'd like to analyze or share. Each worksheet is created by its own query - each time you query the Clickstream database, a worksheet is created that contains query results. Whenever you open the worksheet, the original query is sent to the database and the most current data is displayed.

Index

A

Administrator's Guide
  contents, 1-4
agent, Glossary-1
Apache Log Format Details, A-3
audit trail, 2-10
authenticated user, 3-6
authenticated username, 3-23
authentication, Web server, 3-7

B

business intelligence, Glossary-2

C

clickstream, Glossary-2
Clickstream Administrator
  role, 2-12
Clickstream Analytics, Glossary-2
  introduction, 1-2
Clickstream home, 3-29
Clickstream Intelligence
  architecture, 1-2, 2-5, 2-8
  data model, 1-2
  error messages, D-1
  generic errors, D-1
  introduction, 1-1
  Loader errors, D-3
Clickstream Loader, 2-8, C-1, Glossary-2
  definition, 2-10
  errors, D-3
Clickstream site, Glossary-2
client, Glossary-3
Collector Agent, 1-2, 2-8, 2-9, 3-14, 4-11, C-1
  commands, C-6
  configuration, C-4
  installation, C-1
  start, C-5
  stop, C-5
Collector Server, 1-2, 2-8, 2-9, 4-11, C-1
configuration
  system, 3-22
configuration summary, 3-21
Configure tab, 2-3

D

data filter, 3-22, 3-24
  conditions, 3-25
data format, 3-14, 3-15, 3-22, 3-23, Glossary-4
  define new, 3-23
  general purpose fields, 3-24
  type, 3-24
Data Format Summary page, 3-23
data packet, Glossary-4
data packets, 2-8, 4-11, C-1
  details, 4-12
  lines discarded, 4-12
  lines loaded, 4-12
  lines rejected, 4-12
  status, 4-12
Data Packets Details page, 4-12
Data Packets page, 4-11
data source, 2-1, 3-14, Glossary-4
  create new, 3-15
  host, 3-15

local, 3-14
port, 3-15
protocol, 3-15
time zone, 3-15
view all, 3-14
Data Source Summary page, 3-14
Data Sources link, 4-11
Data Sources page, 4-11
data type, Glossary-4
database, 1-4, 2-10
initialization parameters, B-11
loading, 4-1
partitioning, B-13
populating, 2-12
processes, 4-1
start process, 4-4
definition
create new, 3-6
definition, Web site, 3-2
delete a site, 3-2
Delete Confirmation, 3-3, 3-6
delimiter, 3-8, 3-9, Glossary-4
dimension, 2-12
DNS lookup, 3-32
DNS server, 3-36

E

error messages, D-1
Execution Engine, 4-2
starting, 4-2
execution states, 4-6

F

Fact table, B-13
fact table
Impression, 2-11
Session, 2-11
failed job messages, 4-10
failed load, 4-12
foreign key, 2-12

G

generic errors, D-1

H

hit, Glossary-7
host, 3-15
remote, 3-14

I

identification
session, 3-4
user, 3-4, 3-6, 3-9
visitor, 3-4
internal domain, 3-16
define new, 3-17
internal site, 3-16
define new, 3-17

J

Java Runtime Environment, C-2
JRE, C-2

L

level table, 2-12
Load Clickstream, 3-31
Load Dimensions, 3-33
local search engine, 3-17
define new, 3-18
log formats
Apache, A-2
Microsoft IIS Extended, A-9
supported, A-1
W3C Extended, A-5

M

Manage tab, 2-5
metadata repository
post-installation, E-1
MIME type, 3-27
Mview table, B-13

N

name-value pairs, 3-18
natural key, 2-12
non-page parameters, 3-19

O

Oracle Enterprise Manager
  definition, Glossary-9
Oracle Warehouse Builder, 1-2
Oracle9i Application Server, 1-1
Oracle9iAS Discoverer, 1-4
  Plus, 1-4, 2-10
  Viewer, 1-4, 2-10

P

packet, Glossary-9
page
  Define Cookie Positional Parameter, 3-9
  Define Query String Positional Parameter, 3-8, 3-9
  delete confirmation, 3-3
page parameter
  query string, 3-19
  rules, 3-18
page parameter rules
  global, 3-20
  individual, 3-20
port, 3-15
previous runs, 4-2, 4-4
process data packets, 4-11
Process Data Packets Details page, 4-11
process definition, 3-29, 4-9, 4-10
  create new, 3-30
  type, 3-29, 3-30
process details, 4-9
process messages, 4-10
Process Navigation menu, 4-7, 4-9
Process Status page, 4-2, 4-6
process type, 4-5
process types
  Load Clickstream, 2-10
processes
  currently running, 4-2
  data packets, 4-11
  error, 4-8
  execution states, 4-6
  Load Clickstream, 3-31, 4-5
  Load Dimensions, 3-33, 4-5
  Refresh Summaries, 3-35, 4-6
  Resolve IP Addresses, 3-36
  Resolve Unknown IP Address, 4-6
  Restore a Previous Version, 3-37, 4-6
  resume, 4-8
  start, 4-3
  successful, 4-7
  type, 4-5
  undo, 4-8
protocol, 3-15

R

Reconfig button, 3-5
referring search engine, 3-28
  create new, 3-28
  parameter, 3-28
  URI stem, 3-28
Refresh Summaries, 3-35
Resolve IP Addresses, 3-36
resource type, 3-22, 3-26
  create new, 3-27
Restore a Previous Version, 3-37
resume a process, 4-7
rollback, 3-37
Runtime Administrator
  definition, 2-10
  introduction, 1-1, 2-1
  launching, 2-2
  log out, 2-2
  login, 2-2
  summary, 2-1
  user interface (UI), 2-3
  workflow, 2-7

S

schema
  clkana, 2-8
  clkrt, 2-8, 2-10

star, 2-8
Search dimension, 3-17
session identification, 3-12
cookie value, 3-13
query string, 3-13
timeout period, 3-12
visitor ID, 3-12
session identificationconfiguration, 3-12
settings
session identification, 3-12
site
configuration, 3-1
create new, 3-2
delete, 3-2, 3-5
details, 3-6
edit, 3-3
editing parameters, 3-5
first configuration of, 3-4
user group, 3-2
valid, 3-21
validation, 3-21
visitor group, 3-2
Site Configuration
configuration
site, 3-1
Site Details page, 3-3
Site link, 3-1
Site Navigation menu, 3-1, 3-4
Site Summary page, 3-1
staging table, 2-10
Standalone Collector Agent install, C-1
start a process, 4-3
subprocess, 4-10
details, 4-10
system configuration, 3-22
System Navigation menu, 3-22
system parameters, 3-29

T

tablespace sizing formula, B-8
tablespaces
calculated values, B-9

U

undo a process, 4-7
Uniform Resource Identifier (URI), 3-7
URI stem, 3-18
user identification, 3-6
cookie value, 3-9
entire query string, 3-8, 3-11
positional value, 3-8, 3-11
query string, 3-7
uniquely named value, 3-8, 3-11
Web server authentication, 3-23
user identification by cookie value, 3-13
user identification by query string, 3-11

V

validation, 3-21
visitor identification, 3-10
client hostname, 3-10
cookie value, 3-11
IP address, 3-10
query string, 3-11
visitor field, 3-11

W

warehouse version, 4-9
WC3 Extended Log File Format details, A-7, A-11
Web server, 2-9
Web server authentication, 3-7
Web site definition, 3-2