Confio Igniter™ Suite

Copyright 2009 Confio Corporation. All rights reserved.

The programs (which include both the software and documentation) contain proprietary information of Confio Corporation; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or de-compilation of the programs is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. Confio Corporation does not warrant that this document is error free. Except as may be expressly permitted in your license agreement for these programs, no part of these programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Confio Corporation.
Table of Contents

Introduction........................................................................................................................ 1
  Welcome to Ignite™........................................................................................................... 1
  Wait-Time Analysis.......................................................................................................... 1
  Key Principles of Wait-Time Analysis ............................................................................. 1
  Ignite™ Architecture...................................................................................................... 2
System Requirements ......................................................................................................... 3
  Igniter™ Suite License Keys............................................................................................. 5
  Contacting Confio............................................................................................................. 6
Installing Ignite™ .............................................................................................................. 7
  Getting Started ............................................................................................................... 7
  Installation Steps Overview............................................................................................. 8
  Installing the Ignite™ Server Software............................................................................ 9
  Accessing Ignite for the First Time.................................................................................. 9
  Installing a New Repository ............................................................................................. 10
  Registering a Database for Monitoring ....................................................................... 13
Using Ignite™ .................................................................................................................. 14
  The Main Screen .......................................................................................................... 14
  Performing Performance Analysis with Ignite............................................................... 16
Trend Reports .................................................................................................................... 17
  Report Types ................................................................................................................ 18
Alerts ................................................................................................................................. 19
  Alert Attributes ............................................................................................................. 20
  SNMP Alerting ............................................................................................................... 21
Ignite for E-Business ........................................................................................................ 22
  Enabling Ignite for E-Business ..................................................................................... 22
User Administration .......................................................................................................... 23
Monitor Black-out Periods ............................................................................................... 23
Introduction

Welcome to Ignite™

Ignite™ for Oracle, SQL Server, DB2 and Sybase, gives database administrators a means of easily and quickly monitoring, diagnosing, and resolving performance problems.

Ignite™ is a 24/7 database performance monitoring and problem isolation tool, based on industry best practice Performance Intelligence and Wait-Time analysis methods. Ignite supports Oracle, SQL Server, DB2 LUW and Sybase databases across all operating systems.

Wait-Time Analysis

Wait-time analysis is an approach to application and database performance improvement that allows users to make tuning decisions based on optimal service impact. Traditionally, database performance is measured in the number of events and hit ratios. While somewhat meaningful and easy to capture, these statistics do not reflect a relevant view of the end-user experience nor do they reveal with any precision where the problem originated. Assessing performance without focusing on time impact leaves database administrators (DBAs) guessing about what actions to take in order to address their most important user-oriented problems. This is where Confio’s Igniter™ Suite comes in.

Key Principles of Wait-Time Analysis

Focus on What Impacts End-Users

Rather than tuning to make changes in specific parameters of the server or meet desired hit ratios, tune with a singular focus of making a difference for the end-users of the application. Since these are the real customers, this makes the tuning effort more successful in delivering real benefits.

Focus on the “Long Bars”

A key principle of Wait Time analysis is to identify the specific factors having largest impact. While there could be hundreds of small, technically interesting issues, the “Long Bars” indicate the significant accumulation of delay where there are opportunities to make a real performance difference. Wait Time analysis is unique in its ability to identify where tuning can result in dramatic improvements vs. efforts that go unnoticed.

Break Down Each Step

Generalizing performance across the entire database or server hides all the improvement potential. Wait Time analysis breaks everything down to its atomic level, so that individual hidden steps associated with individual SQL execution can be separately observed. When you know exactly which step is causing the accumulation of delay, you know exactly what to fix.

Time is on Your Side

When collecting statistics, setting alerts, or diving deep into a specific session, always use time accumulation as the driving statistic. Counting operations and trending on general server performance leads to mistaken conclusions. Using wait time as the leading indicator of problem root cause will always lead to the solution that best delivers noticeable performance improvement for the application "customer".
Ignite™ Architecture

Ignite™ consists of a multi-tiered architecture consisting of an Ignite™ Server, Web Clients, an Ignite™ Performance Warehouse database (also called the Ignite™ Repository) and Monitored Database Instances.

- **Ignite™ Server**: The Ignite™ Server performs two key functions. First, it collects data from the monitored database instances. Second, it presents web pages to Ignite™ users. It is installed on a server that has network access to the repository database and each of the monitored databases. Although this process performs many functions, it appears as a single process on the operating system.

- **Web Clients**: Ignite™ users view performance data in a web browser. From this interface, users configure monitoring and use advanced Ignite features such as alerting and emailing reports.

- **Ignite™ Performance Warehouse**: This Repository database is accessed by the Ignite™ Server, and holds all of the performance information collected by Ignite™. Typically, it is a test or development server.

- **Monitored Database Instances**: Ignite’s agentless monitors remotely connect to each database instance (using a JDBC connection) and cause less than 1% overhead on the instance. No software is installed on the monitored server.

![Agentless Architecture Diagram](image)
System Requirements

These configurations have been tested and certified by Confio. Amounts for CPU, RAM and disk space are the minimum requirements for Ignite PI.

- **Web Client (browsers):**
  - Internet Explorer 6 and higher, Firefox 3.0 and higher

- **Ignite PI Server:**
  
  The Ignite software can be installed on almost any Windows, Unix or Linux server. It must have network access to all servers holding monitored database instances, as well as the Ignite Repository. The Ignite PI Server should not be installed on a server running production applications or databases. It must comply with the following:

  - Operating System Support for: JDK 1.4.2.08 or higher
  - Minimum Free Disk Space: 500 MB
  - CPU and RAM: see chart below

<table>
<thead>
<tr>
<th># Monitored Database Instances</th>
<th>CPU's</th>
<th>RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 20</td>
<td>1</td>
<td>1 GB</td>
</tr>
<tr>
<td>20 to 50</td>
<td>2</td>
<td>2 GB</td>
</tr>
<tr>
<td>51 to 200*</td>
<td>4</td>
<td>4 GB</td>
</tr>
</tbody>
</table>

* Ask Confio Support for installation requirements above 200 monitored database instances.

- **Ignite Repository (Performance Data Warehouse):**

  - Oracle 8.1.7 to Oracle 11g
  - Installation requires access to a DBA (Oracle) or Sysadmin (SQL Server) user on the instance that will hold the Repository.

**Notes:**

- “Express” editions of both Oracle and SQL Server can hold a Repository, but will likely run out of space due to the 4 GB limitation in these editions.
- Do not put the Repository on an instance that will be monitored. Doing so will affect the performance of that instance.

<table>
<thead>
<tr>
<th># Monitored Database Instances</th>
<th>CPU's</th>
<th>RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>512 MB</td>
</tr>
<tr>
<td>Up to 5</td>
<td>2</td>
<td>1 GB</td>
</tr>
<tr>
<td>5 to 200</td>
<td>4</td>
<td>4 GB</td>
</tr>
</tbody>
</table>

**Repository Disk Space:** varies with the number of monitored databases, and the level of activity on each monitored database, as follows:

<table>
<thead>
<tr>
<th>Monitored Database Instance Activity Level</th>
<th>Repository Disk Space per Monitored DB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Light</td>
<td>1 GB to 3 GB</td>
</tr>
<tr>
<td>Medium</td>
<td>5 GB to 5 GB</td>
</tr>
<tr>
<td>Heavy</td>
<td>5 GB to 10+ GB</td>
</tr>
</tbody>
</table>
Repository Database Configuration

<table>
<thead>
<tr>
<th>Minimum Oracle Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>db_blockBuffers (or db_cache_size)</td>
</tr>
<tr>
<td>Note: this is a critical parameter.</td>
</tr>
<tr>
<td>shared_pool_size</td>
</tr>
<tr>
<td>sort_area_size</td>
</tr>
<tr>
<td>log_buffer</td>
</tr>
<tr>
<td>session_cached_cursors</td>
</tr>
<tr>
<td>Redo Log Size (select min(bytes) from v$log)</td>
</tr>
</tbody>
</table>

- **Monitored Database Instances:**
  - **Oracle**
    - Versions 8.1.7.2 and higher
    - SYS password (used during database registration)
  - **SQL Server**
    - Versions 2000 (SP3 and higher), 2005 and 2008
    - SYSADMIN privileges are required to setup Ignite monitoring
  - **Sybase ASE**
    - Versions 12.5.0.3 or higher
    - SA_ROLE privileges are required to setup Ignite monitoring
  - **DB2 UDB**
    - Versions 8.2.2 (8.1 FixPack 13) or higher
    - SYSADM privileges are required to setup Ignite monitoring
Igniter™ Suite License Keys

Once you have created a Repository and registered a database for monitoring, you must obtain an Ignite™ license key to view your data. Ignite™ will still monitor your database(s) if you have not yet applied a key, or if your key has expired.

Three types of license keys are available for Ignite™:

- **Trial Key**: Used during an evaluation period (2 weeks).

- **Permanent Server-based Key** (for Oracle, Sybase & DB2 monitored databases): Server-based license used after you have purchased the software from Confio. Permanent keys are valid for only one server (all databases or application servers running on the server) and limited by the number of CPUs purchased.

- **SQL Server Permanent Instance-based Key**: Instance-based licenses are supported for Microsoft SQL Server. These licenses are valid for specific named instances (or the default instance) running on a server.

**Obtaining a License Key**

To obtain a license key, send an email to license@confio.com requesting to trial the software. Please be sure to include contact information.

License keys are added to the Repository using the License Management screen:

![License Management Screen](image)

**Deleting License Keys**

License keys do not need to be deleted. If a key has expired, simply installing a newer key will override the expired key. An expired key that is no longer being used (such as an expired trial key) will not affect the performance of valid licenses.
Contacting Confio

The best method of contacting us for any product related question is via email at:

support@confio.com

Other Methods of Contact

Phone: +1 303.938.8282
Toll free: 1.866.CONFIO.1 (1.866.266.3461)

Confio Support

Current Confio customers requiring technical support should request assistance via support@confio.com, visit the Confio support site (http://support.confio.com) or call one of the following numbers:

+1 303.938.8282 option 2
1.866.CONFIO.1 option 2
Installing Ignite™

Getting Started

Here is a list of preparation items that need to be completed before installing Ignite™ to make the installation process more efficient.

1. Identify the server where the Ignite™ software will be installed. Make sure the server:
   a. Is powerful enough to handle the load of the potential number of monitored databases that you will register.
   b. Has network connectivity to the Ignite™ Repository database and each of the monitored databases that you will register. Ignite™ software can be installed on the same server as the Repository instance, although it is not required.

2. Identify the Oracle database or Microsoft SQL Server instance that will hold the Ignite™ Repository:
   a. Ensure the Repository instance is NOT installed in a critical production database instance.
   b. For a Microsoft SQL Server Repository:
      a. Ensure that a username/password with SYSADMIN privileges is available during installation.
   c. For an Oracle Repository:
      a. Ensure that a username/password with DBA privileges is available during installation.
      b. Ensure that the SYS password is available or that the user can manually login to the database as SYS.

3. Obtain appropriate login credentials for each of the monitored databases:

<table>
<thead>
<tr>
<th>Database</th>
<th>Credentials</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>• non-SYS DBA username/password</td>
</tr>
<tr>
<td></td>
<td>• SYS password</td>
</tr>
<tr>
<td>SQL Server</td>
<td>• SYSADMIN username/password</td>
</tr>
<tr>
<td></td>
<td>• If the SQL Server instance uses Windows Authentication, Ignite will need a SYSADMIN username/password that it will store and use for monitoring.</td>
</tr>
<tr>
<td>DB2</td>
<td>• Username/password with SYSADM</td>
</tr>
<tr>
<td>Sybase</td>
<td>• Username/password that is a member of the sa_role</td>
</tr>
</tbody>
</table>

Helpful tips

- Ignite™ monitoring performs best if a high-speed network exists between the Repository and the Monitored Database Instance. Although monitoring will work across a WAN or low-speed network, it may be necessary to reduce the frequency that Ignite collects the performance data.
Installation Steps Overview

1. Install the Ignite™ server software on a Windows, UNIX or Linux machine.

2. Use the Igniter Suite Setup Wizards to:
   a. Create and configure a Repository database on a non-production instance of Oracle or Microsoft SQL Server.
   b. Register database instances to be monitored by the Repository. You should not monitor database instance from multiple Repositories.
Installing the Ignite™ Server Software

Installation on Windows

The Ignite™ server software is installed through a conventional Windows installer (InstallShield). The Ignite™ installer is normally provided as a single executable named IgnitePI_X_Y_Z_setup.exe, where X, Y, and Z are version numbers (i.e., the installer for Ignite™ version 4.3.35 would be named Ignite4_3_35.exe.)

Download the installer from the website, then run the installer by double-clicking the executable file and follow the installation steps.

By default, Ignite™ will be installed to the c:\program files\confio\ignite pi directory. This directory, or an alternate that you specify during the install, is the Ignite Home directory.

Note: Ignite™ may require a reboot of the system to finish installing.

The Ignite™ installer will create a Windows service called Ignite PI Server. This service is set to run automatically when the server is restarted. Ignite™ can be stopped by shutting down this service. The service startup type may be changed to “Manual” if desired.

Installation on Unix/Linux

1. Download the zip file from the website.
2. Create a directory on the server to hold the new installation. This will be the Ignite Home directory.
3. Extract the files into the new directory.
4. Start Ignite™ (view the readme.txt file in the Ignite Home directory for startup instructions).

Accessing Ignite for the First Time

By default, Ignite™ will be available at the following web address. Open your browser and connect to this URL and should see the Ignite Repository Creation Wizard screen.

http://yourserver:8123/

Troubleshooting

View the logs available in the following directory:

<Ignite Home>/iwc/tomcat/logs/

Make sure a firewall is not running on this machine.

Make sure another process isn’t using the default Ignite™ ports. If the ports are being used by another process, change the default ports of 8123 and 8127 by editing <Ignite Home>/iwc/tomcat/conf/server.xml and updating the following lines with new port numbers:

<Connector port="8123"/>
<Server port="8127" shutdown="SHUTDOWN"/>
Installing a New Repository

When you connect to the Ignite web server for the first time, it will require you to create a Repository.

The following page will appear:

![Repository Wizard](image)

This page presents 2 choices:

1. Click “Create New Ignite PI Repository” to start with a brand new Repository and then register databases to monitor.

2. If you want to upgrade a Repository that you run with Ignite 6.5 to monitor your databases, click “Upgrade Existing Ignite 6.5 Repository” to take you through the upgrade steps. Make sure you read the following knowledge base article before upgrading:

   support.confio.com/kb/1535/
If you chose to create a new Repository, you will be presented with the following wizard pages.

Choose the type of database instance that will hold the Repository.

The wizard will guide you through the appropriate steps to create the Repository. If the Repository is stored in an Oracle database, it will create a new Oracle schema and populate it with some tables, indexes and initial data. If the Repository is stored in a Microsoft SQL Server instance, it will typically create a new database and populate it with some tables, indexes and initial data.
When the “Repository Creation Wizard” is complete, it will allow you to register a database instance for monitoring.
Registering a Database for Monitoring

Now that Ignite™ has a repository, it is time to register a database instance for monitoring.

The wizard begins by asking the type of database to be monitored.

Choose the type of database that you want to monitor.

The wizard will continue through various steps specific to the database type.

When the wizard has completed successfully, Ignite™ will begin monitoring the database instance. Ignite publishes the performance data every 10 minutes so it is possible that data will not be immediately available.
Using Ignite™

The Main Screen

After at least one database instance has been registered for monitoring, Ignite™ will show the main screen.

![Main Screen Screenshot](image)

The main screen shows a table of the database instances being monitored. Notice that different database types (Oracle, Microsoft SQL Server, Sybase and DB2) can be seen on the same screen. The monitoring can be started or stopped from this screen.

Instance Groups

You can also group monitored databases using the Instance Groups feature. The repository above has databases groups defined as the various database types, but you can define any mix of groups. Ignite™ will automatically group Oracle RAC instances; all other databases are not automatically grouped.

Monitoring

Once monitoring is started it will always monitor if possible. In other words, it is not necessary to restart the Ignite monitor if either the repository instance or the monitored database instance was unavailable for a period of time. Monitoring will automatically resume when both are available again. Using the blackout schedule feature, you can configure times that monitoring should not occur. This feature is described later in this document.

Logs

Clicking the ‘Log’ link for a database shows the Ignite™ monitoring log for the associated database. This is helpful to review if the Ignite™ monitor cannot start for some reason or seems to be missing some data. The ‘Log’ link in the top right corner of the page shows the log entries for all monitored database instances.
More Features

Other buttons/links on the page include:

- **Register Database for Monitoring**: registers another database instance to be monitored by Ignite™
- **Alerts**: create/edit/view alerts
- **License Management**: add Ignite trial or permanent license keys
- **Options**: Ignite settings and advanced features such as custom Ignite users
- **Reports**: create/edit custom reports
Performing Performance Analysis with Ignite

To see specific examples of how to use Ignite™, view the embedded tutorial. It can be accessed by two methods:

- The main page link titled: “How do I use a performance intelligence tool?”
- http://www.confio.com/tutorials/
Trend Reports

Trend Reports let Ignite™ users communicate the important, long-term performance of their databases across the organization in published reports with easy-to-understand charts. With Trend Reports, the DBA can quickly capture the results of performance tuning efforts, database resources of concern, or database trends in the form of graphical reports and publish these reports for managers, team members and customers. Trend Reports give Ignite™ users a vehicle to show evidence to support their work across the organization, highlighting database trends for up to 5 years.

This standard report and customization feature is easy to use, and provides a high degree of flexibility. Without complex set up or configuration, reports can be instantly published to, for example, highlight Top 5 Resource bottlenecks or highest SQL statement utilization. Then, DBAs can select the unique set of SQLs, Resources, or Programs of interest, and tailor their own reports for a customized period.

Trend Reports Use Summarized Data

Trend Reports are different from the detailed data displayed in the typical Ignite™ charts because Reports are available for summarized data that extends over months or years. Detailed Repository data is accessible for a shorter detailed data storage period, typically 30 days. Trend Reports can show data captured over longer intervals and allows DBAs to display and capture longer term trends.

To generate Trend Reports, Ignite™ summarizes and condenses repository data to make long-term information available within a manageable size. This process is automatic, and no user management of the Report data summary process is required. The data is summarized by hour, and this hourly data is available for 90 days. After 90 days, the data is summarized by day and is available for five years.

Minimum Data to Generate a Report

Because the Trend Reports are designed to show data trends over hours, days, months and years, a display of Reports requires a minimum of 3 hours of collection. Detailed repository data, in contrast, with time granularity down to the second, will display information within 10 minutes of initial collection. Best results are achieved after letting the Repository collection run for at least a full day before generating Reports.

Creating a Report

You can create a new report by clicking the ‘Reports’ link from the main page. The report menu page is displayed.

Once a report type has been selected, you may be prompted for some mandatory parameters. In all cases, you can choose “Advanced Options” to change the default report parameters selected by Ignite™. Common report parameters include date range, interval, and time of day.
This page will also show any saved reports and report groups. A report group is simply a group of reports that will be displayed together.

This page also provides a link to report schedules. Report schedules are used to automatically email a report (or group of reports) at a regular interval. For example, you could create a schedule to email a list of the top ten worst performing SQL statements every Monday at 9am.

### Report Types

Ignite™ is pre-configured with many standard reports, highlighting the most commonly used Wait Time statistics. DBAs can use these standard reports, or customize these standard reports and save their own versions that reflect the unique trending picture they wish to communicate. Once saved, the custom reports can be easily published with up-to-date data whenever they are needed. Below are some examples of the reports that are available.

**Top Waits**
This report charts Wait-Events for an entire database. By default, Ignite™ displays the top five Wait-Events with highest accumulated Wait Time. The user has the option to display the top ten or select specific Wait-Events of interest on the Wait-Events tab in the Report Properties dialog.

**Top Waits for single SQL**
This report identifies the top Wait-Events (up to 50), ranked by Wait Time, for a specific SQL statement. The user has the option to display the top N or select specific Wait-Events of interest from the Wait-Events tab.

**Top SQLs**
This report shows the top five SQL statements ranked by total Wait Time across a user selected interval. The user has the option to display the top ten or select specific SQL statements from the SQL Statements tab on the Report Properties dialog. The report will show a descriptive name for the SQL if it has been defined previously by the user.

**Top SQLs for Single Wait**
This report is similar to Top Waits for single SQL except it shows Wait Time for a single specific database Wait only.

**Top Programs**
This reports the total accumulated Wait-Time for Programs. It shows the top five programs by default and allows the user to display up to ten or select specific programs of interest from the Programs tab on the Report Properties dialog.

**Top Files**
Ignite™ calculates the total Wait Time for all I/O operations on each file for selected intervals. This report displays the top five busiest files ranked by total I/O Wait Time. The user has the option to display up to the top ten or select specific files from the Database Files tab on the Report Properties dialog.

**Typical Day of Wait for single SQL**
For an individual SQL statement, this report displays a stacked graph showing average hourly Wait Time for the top five Wait-Events contributing to the total Wait Time for the SQL statement. The uppermost portion of the bar is the summation of all other Wait-Events. The user has the option to display the top ten or select specific Wait-Events to display from the Wait Events tab. This report is useful for identifying the peak loads during a business day based on long-term observation of the system.

**Typical Day of Wait**
This report displays a bar graph showing the average hourly Wait Time for a specific database. Similar to Report #7 above, the report displays the distribution of average Wait Time versus time-of-day.
Alerts

Ignite™ Alerts give DBAs proactive control of their database, notifying them of issues before they become problems for database customers. By setting thresholds on key Wait Time statistics or on standard administration indicators, the DBA group gets an early warning of potential problems, and can take steps to solve the underlying issue before users are affected. The result is improved customer service, fewer trouble tickets, and increased compliance with database SLAs.

Alerts, like Trend Reports, are designed for both instant use out-of-the-box and for a high degree of customization to give expert DBAs a tool tailored to their specific needs. Ignite™ is preconfigured with three types of alerts:

**Wait Time Alerts**

Wait time alerts are based on the amount of time a user or application(s) waited on the database. For example, the *Average SQL Wait Time* alert fires when a SQL statement causes more wait time for the user than is acceptable. These alerts are critical because they ensure that the DBA is only alerted when users and applications are being affected.

**Database Administration Alerts**

DB Admin alerts are typical alerts surrounding the health of the database system. For example, the *Database Parameter Changes* alert will notify when any database parameter has changed. The *Database Availability* alert notifies when the database instance is no longer accessible (perhaps it has crashed or the network has gone down).

**Custom Alerts**

Custom Alerts are user-specified queries that are run against the monitored database or the Ignite repository. The query returns a number (or set of numbers) that may trigger an alert depending on user-defined threshold settings. For example, a DBA could enter a query to detect the number of cancelled orders in the last ten minutes.

```
sel ect count(*) from orders where status = 'CANCELLED' and date > sysdate – 10/1440;
```

They also might want to see which type of orders had high cancellation rates.

```
sel ect order_type, count(*) from orders
where status = 'CANCELLED' and date > sysdate – 10/1440
group by order_type;
```

Finally, they may want complex logic to determine if an order has truly been cancelled. In this case, they could write a function or procedure on the monitored database instance and call it.

```
Oracle: select mycustomfunction(parm1) from dual
SQL Server: select mycustomfunction(parm1)
Sybase: mycustomprocedure(parm1)
DB2: select dbo.mycustomfunction(parm1) from SYSIBM.SYSDUMMY1
```

These functions or procedures are written in the native database programming language such as PL/SQL for Oracle, T-SQL for MSSQL, T-SQL for Sybase, and IBM SQL for DB2.
Alert Attributes

When creating an alert, you need to specify the following attributes:

**Active**
Indicates the alert will be automatically executed by Ignite. Uncheck this box if you would like to temporarily disable an alert but do not want to delete it.

**Execution Interval**
Indicates how often the alert is run. It has a minimum of ten minutes to ensure that the alert doesn’t put too much load on the monitored database instance.

For Wait Time alerts, this value also indicates the time period that Ignite™ uses to examine data, looking backwards from the time the alert runs. For example, if the execution interval is 60 minutes, Ignite™ will execute the alert every 60 minutes and also query the last 60 minutes of Ignite™ performance data as alert input.

**Notification Text**
This text will be sent with the notification email or page. It should include an explanation of the alert as well as the suggested resolution steps.

**Database Instance to Monitor**
Specifies which database instances will be monitored with this alert. Some alert types allow multiple database instances to be specified.

**Alert Levels and Recipients**
This section is used to specify the ranges of values that will trigger an alert notification. For example, let’s say a particular SQL has an SLA to execute in under 4 seconds, but typically executes in less than 2 seconds. A ‘Low’ alert threshold could be set at the 2 second mark to send an email to the DBA. Another ‘High’ alert threshold could be set to the 4 second mark to page the same DBA.
Notification Group or Contact
Alerts can be sent to an individual, a group, or to an enterprise management console such as HP Openview or Tivoli using SNMP traps. It is important to note that when Alert Levels are reached the appropriate notification group/contact is notified only once. As Alert Levels change (i.e. go higher or lower), a group will not be notified again if it has already been notified once. This will continue until the Alert Level goes back to Normal.

SNMP Alerting
Ignite™ alerts can be configured to send SNMPv2c Traps to a SNMP-enabled Network Management Station (NMS) when an alert level or threshold is reached. The trap will contain the name of the monitored database, alert name, alert level, and response instructions. The NMS that will receive the trap is represented as a contact (SNMP Contact) in Ignite™.

SNMP Contacts
SNMP Contacts represent an NMS that will receive traps from Ignite Alerts. An SNMP Contact is created and modified through an Ignite™ Contact Management dialog. To create an SNMP Contact, click the Create SNMP Contact button to display the Create New SNMP Contact dialog.

![Create New SNMP Contact dialog](image)

- **Name:** HP Openview
- **Description:** Sends a trap to HP Openview
- **Trap Receiver Host:** openviewserver
- **Trap Receiver Port:** 162
- **Community String:** public
- **Response Instructions:** Page the on-call DBA

The SNP Contact dialog also includes options for adding or removing available groups and groups joined.
Ignite for E-Business

Confio Ignite™ for Oracle E-Business is an optional extension of Confio Ignite™ for Oracle. It provides increased visibility into the causes of performance problems in Oracle E-Business Suite, Oracle ERP and Oracle Apps environments. More specifically, Ignite™ for Oracle E-Business captures data passed from E-Business Suite to the database that can uniquely identify the screens, modules, and users generating database requests.

Ignite™ for E-Business captures data passed in the module and action fields of Oracle SQL statements. When Ignite™ for E-Business is enabled, these fields will be monitored and displayed as options in the Ignite™ interface. Any application that stores data in the module and action session fields can be monitored with the Ignite™ for E-Business feature. In particular, if custom developed applications utilize these fields, Ignite™ for E-Business will monitor and display the wait time accumulation based on these parameters.

Enabling Ignite for E-Business

To enable Confio Ignite™ for Oracle E-Business:

1. Click ‘Options’ and then click ‘Advanced Options’.
2. Select the monitored database that holds the Oracle E-Business application (or similar application that populates the module and/or action session fields).
3. Edit the parameter called ‘ORACLE_ERP’ and set it to true.
User Administration

Ignite™ provides a way to create and manage several different types of users who can access the Repository data. Some users can only view the data while others can start/stop agents or alter monitoring parameters. They must be unique user names within the Repository database and they are deleted with the Repository is dropped.

To create a user, go to the 'Options' page and click menu and click ‘Custom Users’. Click the ‘Create User’ button.

Monitor Black-out Periods

Ignite™ can be setup to not monitor during certain periods. To create a black-out schedule, go to the ‘Options’ page and select ‘Database Monitor Blackout Periods’.