Qtime
The Qddb Time Tracking System
User's Guide

Version 2

Herrin Software Development, Inc.

2001

Contents

List of Figures

0.1 Copyright and Warranty

Qtime is based on Qddb's Fx library. For more information on Qddb, visit:

`http://www.hsdi.com/qddb`

Support and commercial binary distributions are available via:

`http://www.hsdi.com/qddb/orders`

Chapter 1

Introduction

Qtime is a `Qddb` database application that provides a convenient interface for managing time spent on projects.

Qtime's features include:

- powerful graphical user interface based on the Qddb Fx library
- automatic job number and client id generation
- each client can have many individual jobs and each job has many time splits
- stopwatch functionality to automatically track time
- capability to manually add new time intervals or modify existing time splits created from the timer
- individual time splits along with total time charge
- record your comments per individual time split or per project
- computation of "amount due" based on total time charged and hourly rate values
- discount percent field available. Totals are automatically updated whenever user modifies hourly rate and/or discount percent
- convenient history menu that enables quick access to timers that have been used most recently
- ability to print/email statements
- report generation capability based on Qddb's builtin report generator to find out exactly where all your time goes

This guide is for users that want to install and learn how to use *Qtime*.

Chapter 2

Obtaining and Installing *Qtime*

To acquire a copy of *Qtime*, visit the *Qddb* applications page:

http://www.hsdi.com/qddb/applications

2.1 Installing *Qtime*

To install *Qtime*, perform these steps:

```
1$ gunzip -c qtime-<version>.tar.gz | tar -xvf -
2$ cd qtime-<version>
3$ vi Makefile
4$ make install
```

Figure 2.1: Steps to installing *Qtime*

Line 1 unpacks the *Qtime* tarball. A directory named qtime-<version> is created beneath the current directory containing the *Qtime* distribution. Line 2 instructs you to change to the qtime-<version> directory. Line 3 instructs you to edit the Makefile. There are instructions in the Makefile to tell you how to prepare for installation. You must be sure to set the appropriate environment variables to tell the "make" process where your qwish executable is located. The information you specify in the fields: YOUR_COMPANY, YOUR_STREET, YOUR_CITY, YOUR_STATE, etc tell Qtime the default information that first appears in the "User information" window. This information can be modified at any time from the xqtime "User information" window through the xqtime configuration menu option. xqtime displays this information on the printed and email reports.

After you've saved your changes to "Makefile", Line 4 starts the "make install" process. After this step, you've completed your installation and are ready to run the *Qtime* programs: newqtime and xqtime.

Chapter 3

Using *Qtime*

Qtime is comprised of two executables:

newqtime creates a new *Qtime* database in the current directory

xqtime is the main *Qtime* program. `xqtime` displays a graphical interface that allows you to (1) search, add, modify, and delete entries from the Qtime database, (2) launch timers (i.e. stopwatches) to automatically track time spent on tasks and (3) produce statements for printing or sending Email to clients.

Each of these programs is discussed in the following sections.

3.1 newqtime

Before using *xqtime*, you first need to setup your *Qtime* database using the *newqtime* command.

At your command prompt, enter `newqtime RelationName` where `RelationName` is the name of your new `Qtime` database name. A new directory named `RelationName` will be created beneath your current directory. Alternatively, you can specify a full path for your new database. A common approach is to store all your databases in a common location. For example, suppose you like to keep all your databases in `/usr/local/databases`. If you wanted to create a new "MyProjects" Qtime database, you would enter the command shown in figure 3.1.

```
1$ newqtime /usr/local/databases/MyProjects
```

Figure 3.1: Running newqtime

If you run *newqtime* without any arguments, `newqtime` will create a relation named *Qtime-db* in your current directory.

Once you've created your database using `newqtime`, you can add, modify, import/export, delete entries from this database using the `xqtime` program.

3.2 xqtime

3.2.1 Setting the QDDBDIRS environment variable

In order for `xqtime` to find your databases, you should be sure to have the environment variable QDDBDIRS set appropriately. For example, if all your Qddb `Qtime` databases are in `/usr/local/databases`, then you would enter the following command (bash shell syntax): `export QDDBDIRS=/usr/local/databases:$QDDBDIRS`

3.2.2 Launching the xqtime program

To launch the `xqtime` program, the syntax is:

```
$ xqtime RelationName
```

Since Qtime (version 1.2 or later) is based on `QddbScript`, you can launch Qtime simply by typing:

```
$ nxqddb RelationName
```

where RelationName is a database that was previously created using the `newqtime(1)` program.

3.2.3 Using xqtime

xqtime is an Fx-based Qddb application. Hence, all the built-in features and behavior of other Fx-based applications are present in *xqtime*. For example, you can search on any field and configure the many options available via the *Configure* pulldown menu.

Reference the *Qddb User's Guide* for details on how to operate the *xqtime* window.

3.2.4 Setting up your initial user information

When you (or your system administrator) installed `Qtime`, there were options in the Makefile to default certain values used by `Qtime` when printing or mailing statements. You can modify these at any time via the "Configure/User Preferences" menubutton. Clicking this menu button displays the user setup window shown in figure 3.2.

There are three notebook tabs on this window (see figures 3.2, 3.3 and 3.4):

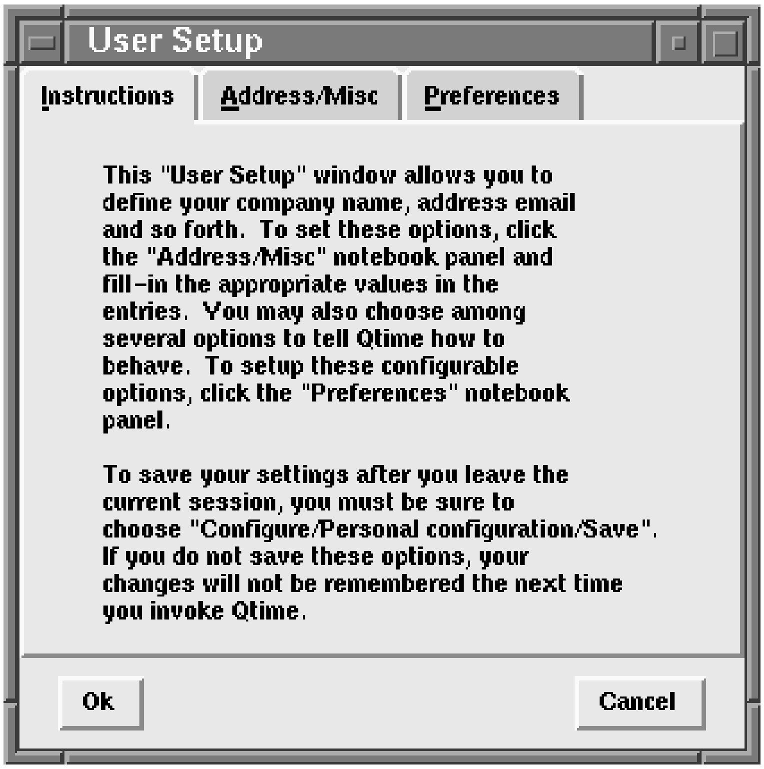

Figure 3.2: User Preferences Window (Instructions Panel)

- The first panel, "Instructions", provides the instructions for how to define your preferences.
- The "Address/Misc" panel contains entry boxes for you to specify the information that is output on statements produced by `Qtime`.
- The "Preferences" panel contains two options:

Split refresh option tells `Qtime` how long to wait before updating the time interval computed in the `Qtime` stopwatch window. A value of 30 seconds is a reasonable value that works for most users.

Report seconds option allows you to choose how `Qtime` reports seconds. These values are defined below:

Yes value will include seconds on the time reports.

Truncate value will truncate seconds and create a negative time interval to adjust the job's time to be on the minute.

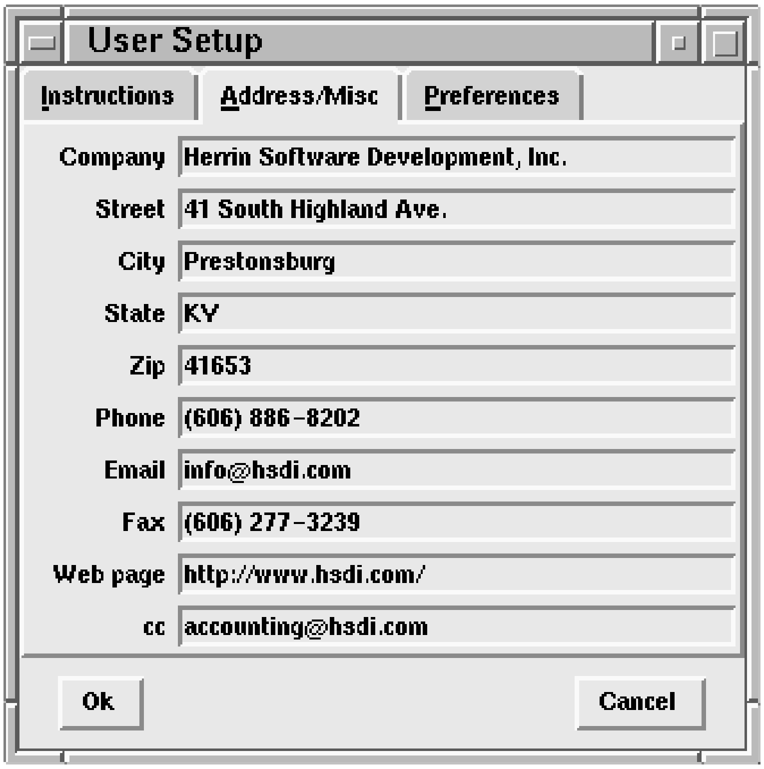

Figure 3.3: User Preferences Window (Address Panel)

Round value will round to the nearest minute and also create a negative time interval to adjust the job's time to be on the minute.

Roundup value will always round up to the nearest minute and also create a negative time interval to adjust the job's time to be on the minute.

Be sure to save your personal and/or global configuration after modifying your user setup to tell `Qtime` to use your settings the next time you start up the `Qtime` application. Turning on the *Auto-save Personal configuration* checkbutton is a nice feature to ensure against users leaving without saving their changes. Also, the *Force-field formatting* is another option that is useful; especially, since `Qtime` contains several date fields. For example, when you type the word "today" in a date field and tab out of the date field, `Qtime` will immediately substitute the current date for the word "today".

3.2.5 Fields on the xqtime window

The `xqtime` window is divided into two major screen panels (see figure 3.5). The topmost

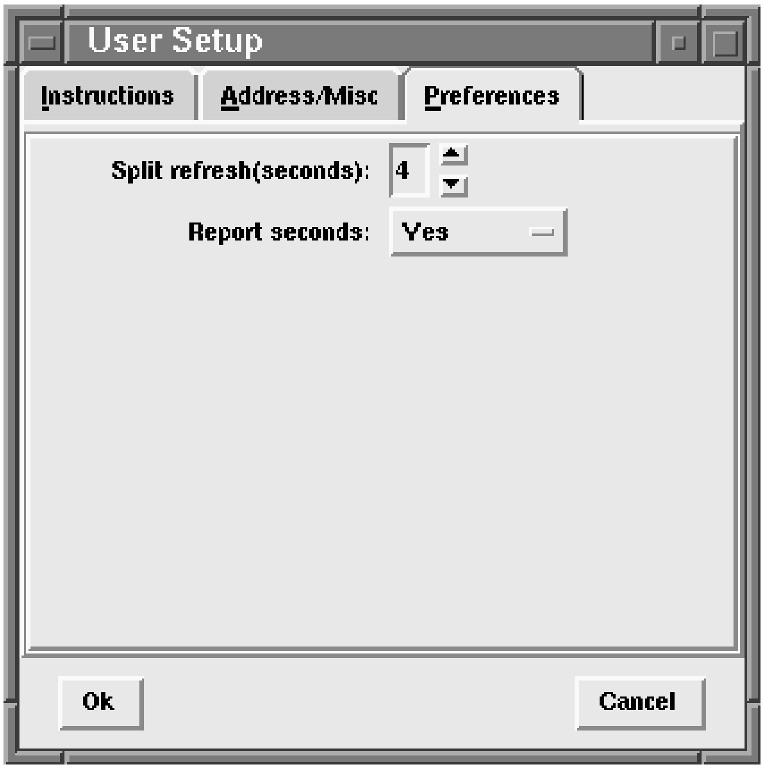

Figure 3.4: User Preferences Window (Preferences Panel)

panel contains notebook panels that contain information pertaining to the client being charged for the project:

Name panel is the default panel and contains the client's name, client id, company, tax exempt number, social security number (SSN), customer type, customer class and referred by. The *client id* field is a mandatory field field as distinguished graphically by the darker shaded entry. This *client id* field is a read-only field that is automatically populated by `xqtime`. `xqtime` does this to ensure that all client ids remain unique and to ease data entry for the `xqtime` user.

Addresses panel contains the client's address information(see figure 3.6).

Phones panel contains the client's phone numbers(see figure 3.7).

Comments panel allows you to record notes pertaining to a particular client(see figure 3.8)

The lower portion of the `xqtime` window contains the individual jobs (or projects) being worked for the client noted in the upper portion of the `xqtime` window. The first header, "Job", denotes the job information and contains four buttons that allow the user to perform specific actions. Namely, the user can (1) add new jobs, (2) view existing jobs, (3) delete existing jobs, and (4) recompute the totals by retabulating each individual time split and figuring the hourly rate and discount percent to arrive at a total amount due value.

The "Job" panel contains the following three notebook panels:

Summary panel is the default panel and contains the primary job information(see figure 3.9).

Intervals panel records each time split(see figure 3.10). These fields can be populated through the use of a timer or can be populated manually while in *Change Mode*.

Comments panel records remarks particular to a specific job(see figure 3.11).

3.3 Timer

To launch a timer for a particular project, follow these steps:

1. Go to search mode. Key in the search information to find your project. Remember, within any `Qddb` application, you can search on any field.

2. Once you've found the desired project, the *Timer/Launch* menu option will become enabled. Click this menubutton to launch the window shown in figure 3.12.

3. From the timer window, click the *Start* button.

Alternatively, if you recently used a timer for a project and have saved either your personal or global settings, `xqtime` will include the project in the *History* pulldown menu. Clicking this selection in the History menu will automatically launch the timer from any mode.

Once a timer is launched for a job, the *Timer* menu is updated by appending the *job number and name*. Thus, you can just look at the Timer pulldown menu to see the timers that you have running. Clicking on one of the timers from the *Timer* menu will deiconify a timer and bring the timer to the front of your display.

3.3.1 Starting the timer

Clicking the *Start* button on the timer window will begin the timer for that particular time interval. Once the timer is ticking the "Stop" button will be enabled and the "Start" button becomes stippled (see figure 3.13). The focus is placed on the *Stop* button when the user first starts a stopwatch.

While the timer is actively timing, you can freely enter text in the `Work unit log` field. The text entered in this field will be stored in the *Interval* comment's field when you choose the stop button.

3.3.2 Stopping the timer

Clicking the *Stop* button will update the work interval's fields as well as the *Total time charged* and *Amount due* fields. If you have the record still displayed, `xqtime` will update the field contents. If the record is not displayed (e.g. you decided to look for another record), the appropriate fields are still updated and will be current the next time you visit the project. The *Start* and *Dismiss* buttons will become enabled and the *Stop* button becomes disabled.

3.4 Manually adding/modifying time splits

To manually add a new time split, follow these steps:

1. From change mode, click the *Intervals* notebook tab to display your project's time splits. Note, you must be in change mode before you can add or modify time splits.
2. Populate the (1) start date, (2) start time, (3) end date, and (4) end time fields. Note, the duration field is read-only and gets populated when you click the *Recompute* button.
3. Click the **Recompute** button after you've entered the above fields. Clicking this recompute button automatically figures the duration by subtracting the "start date and time" from the "end date and time". All other appropriate amounts are updated to reflect your new time split, including: *total time charged*, *amount due*, and *discount amount*(if you populated the discount percent field).

Modifying existing time splits works just like adding new time splits. First, you find your project and while in change mode, modify one of the date/time fields. Clicking the "Recompute" button automatically updates the appropriate fields.

Note, you must choose the "File/Save" to record your changes permanently in the *Qtime* database.

3.5 Statements

To produce a statement to be mailed or printed, follow these steps:

1. Search for the project that you would like to generate a statement.

2. Once you've found the project and are in *Change mode*, `xqtime` will enable the *Statement/Print* and *Statement/Email* menubuttons.

3. Choose the *Statement/Print* button to generate a printout of the statement. Alternatively, choose the *Statement/Email* button to email the statement to the client.

3.5.1 Print statement

After you choose *Statement/Print*, `xqtime` displays the standard print dialog(see figure 3.14) box to allow you to direct your output to (1) printer, (2) floppy disk, (3) hard disk file, or (4) preview the statement on screen.

3.5.2 Email statement

After you choose *Statement/Email*, `xqtime` displays a window similar to the one shown in figure 3.15. Note, `xqtime` automatically populates the information from the `xqtime` database record. You can modify the statement text within the Email window before issuing the statement. From this window, click the *Send message* button to email the statement or abort by clicking the *Cancel message* button.

3.6 Restabilizing your Qtime database

As with any other *Qddb* database, you can stabilize your database from the command line by typing `qstall <RelationName>`. Alternatively, inside a *Qtime* application, you can click the *Misc/Restabilize* menubutton to stabilize the *Qtime* database. Note, you must be in *Search mode* before the *Misc/Restabilize* menubutton becomes available (i.e. unstippled).

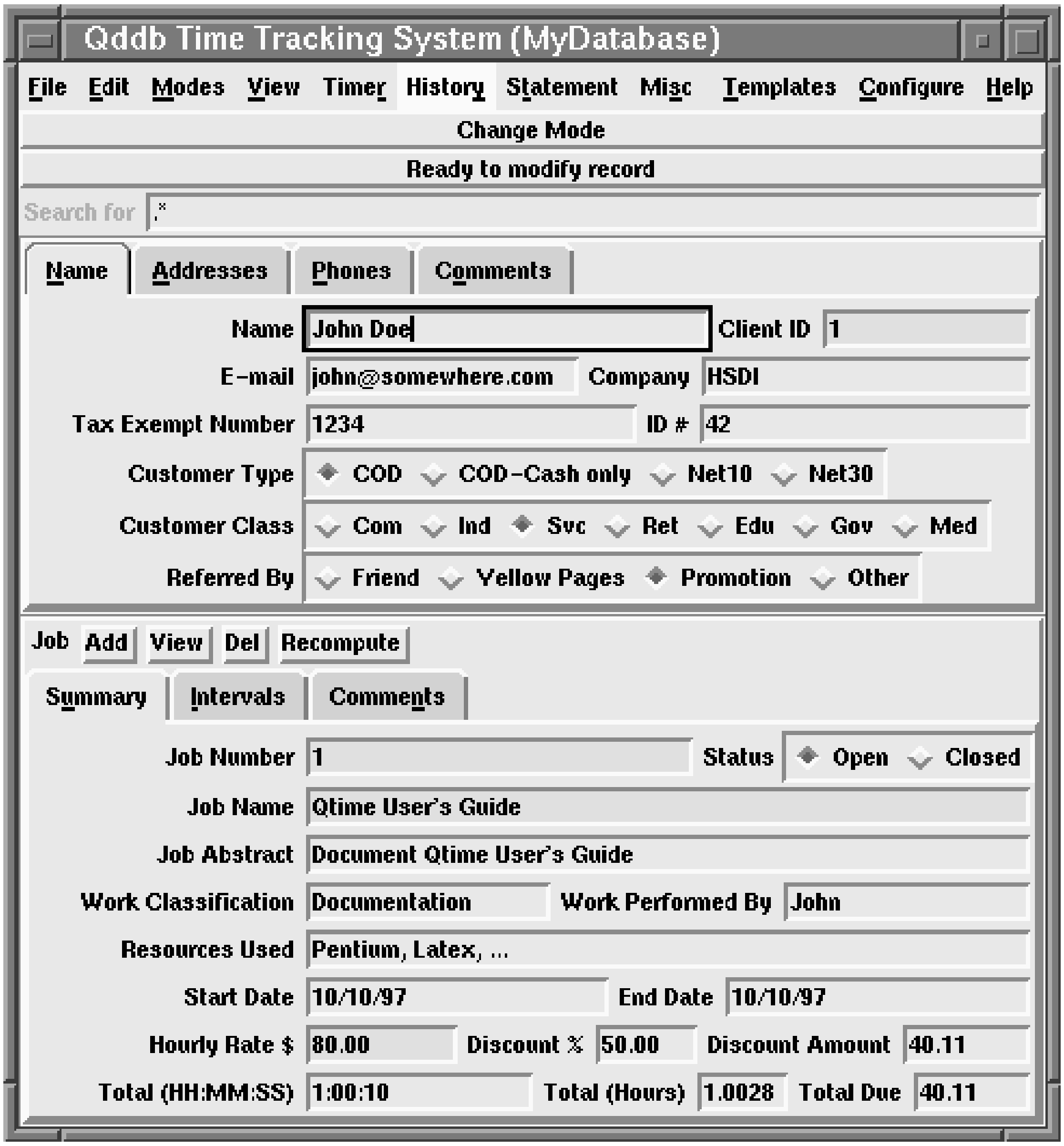

Figure 3.5: Qtime window

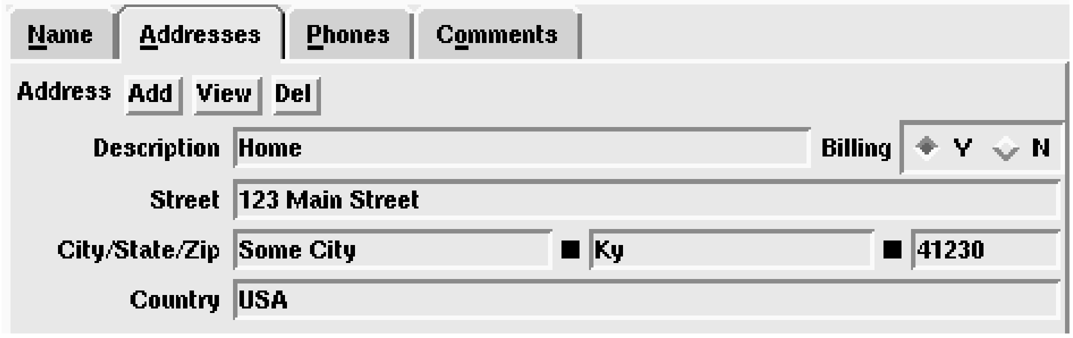

Figure 3.6: Addresses panel

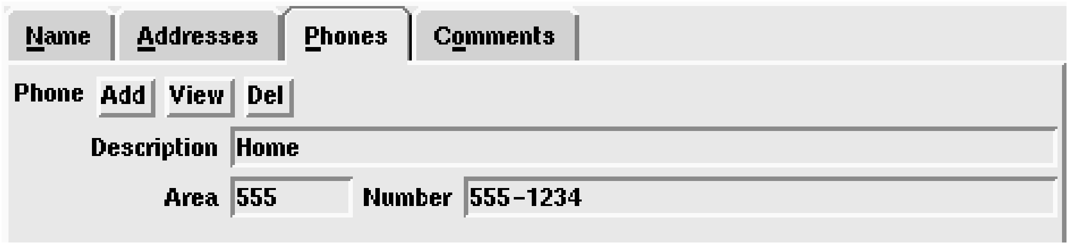

Figure 3.7: Phones panel

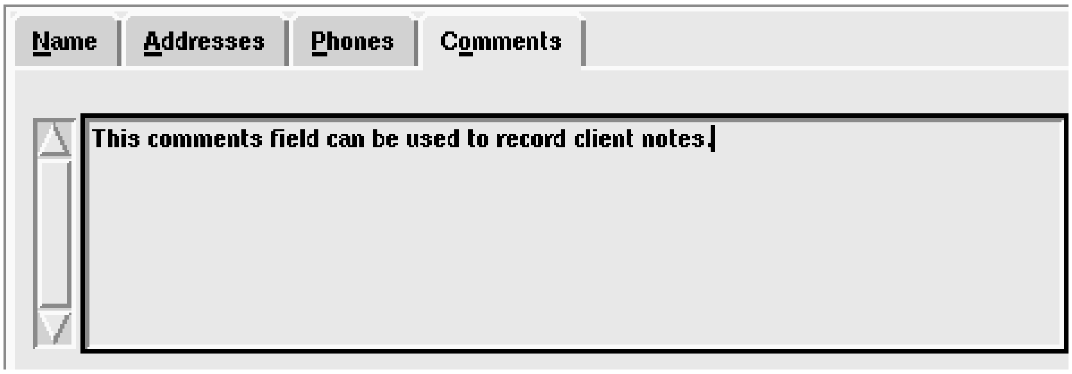

Figure 3.8: Comments panel

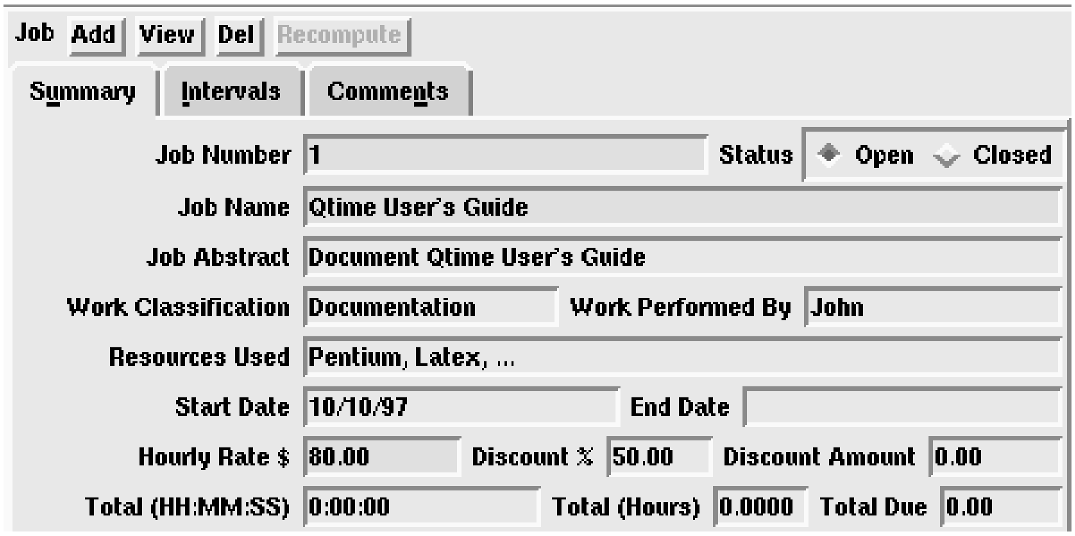

Figure 3.9: Job summary panel

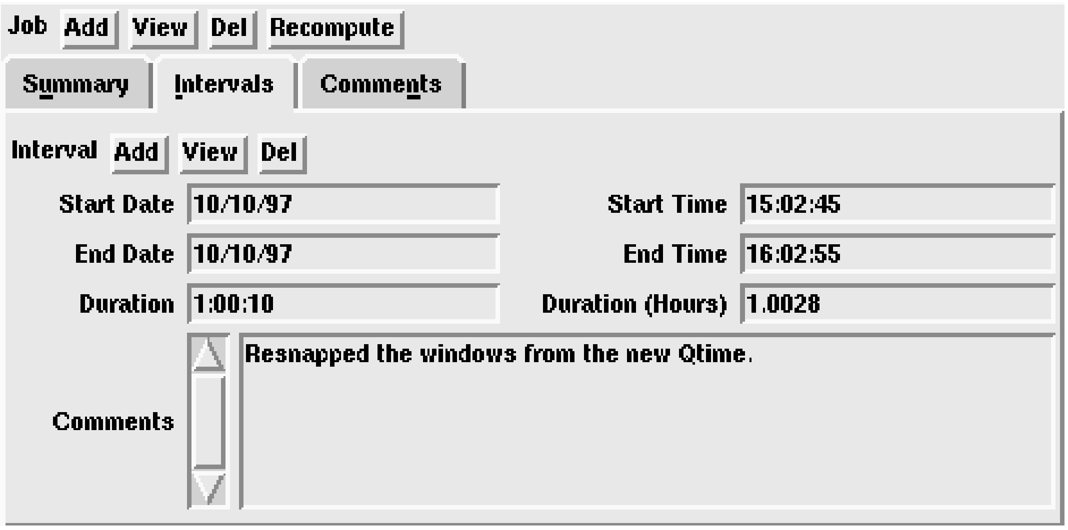

Figure 3.10: Intervals panel

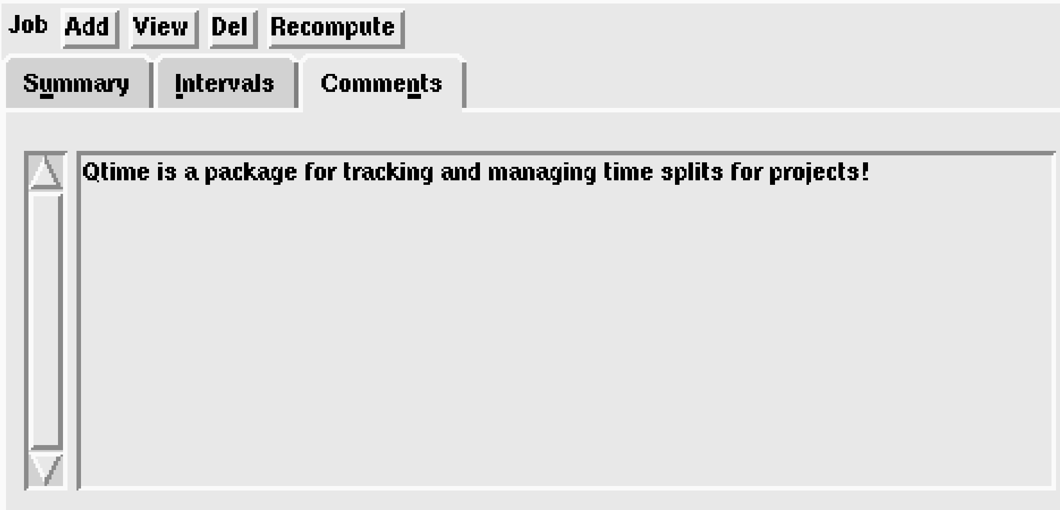

Figure 3.11: Job comments panel

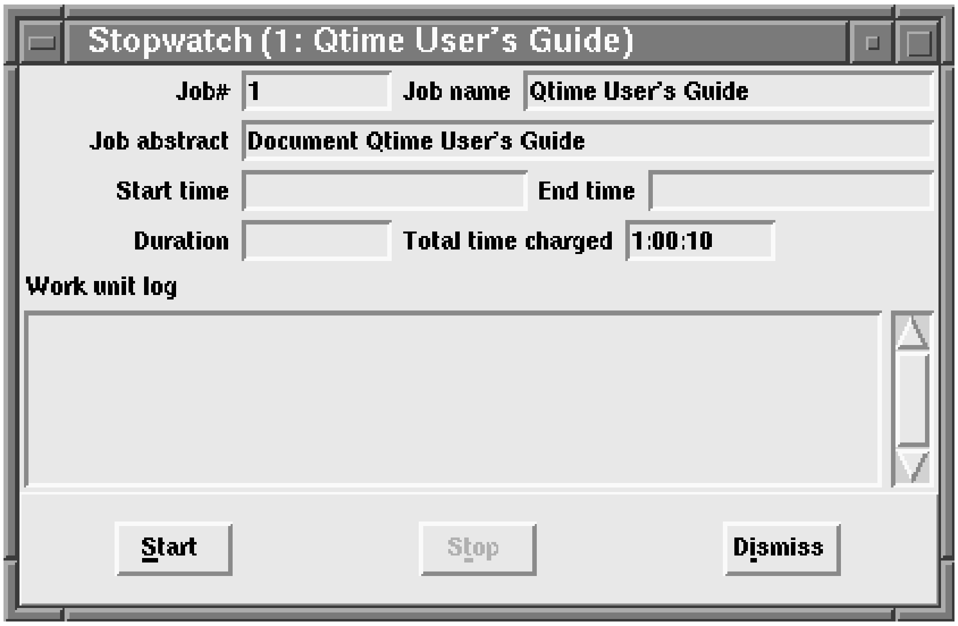

Figure 3.12: Timer window in *Stop* mode

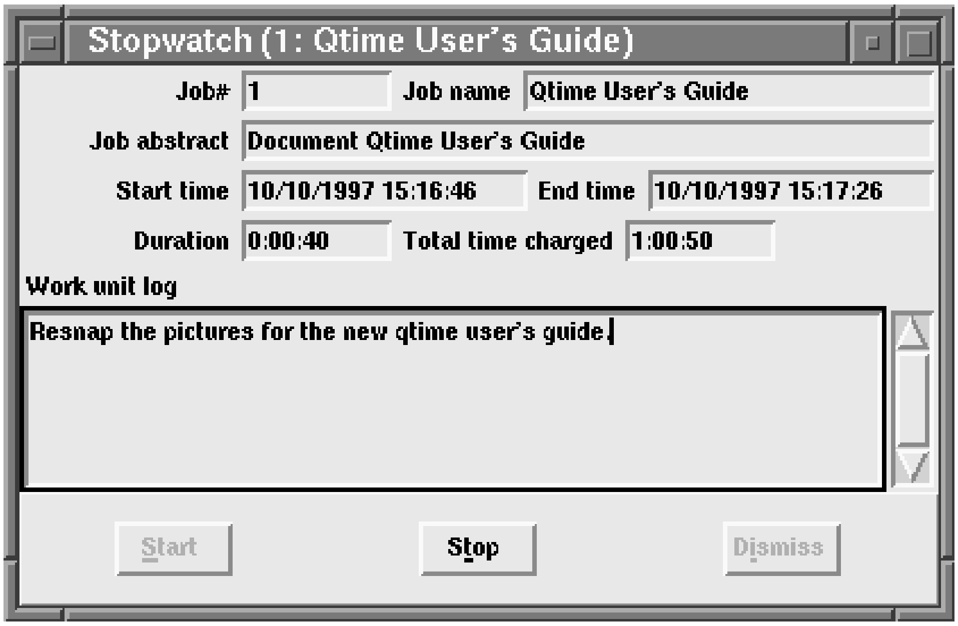

Figure 3.13: Timer in timing mode

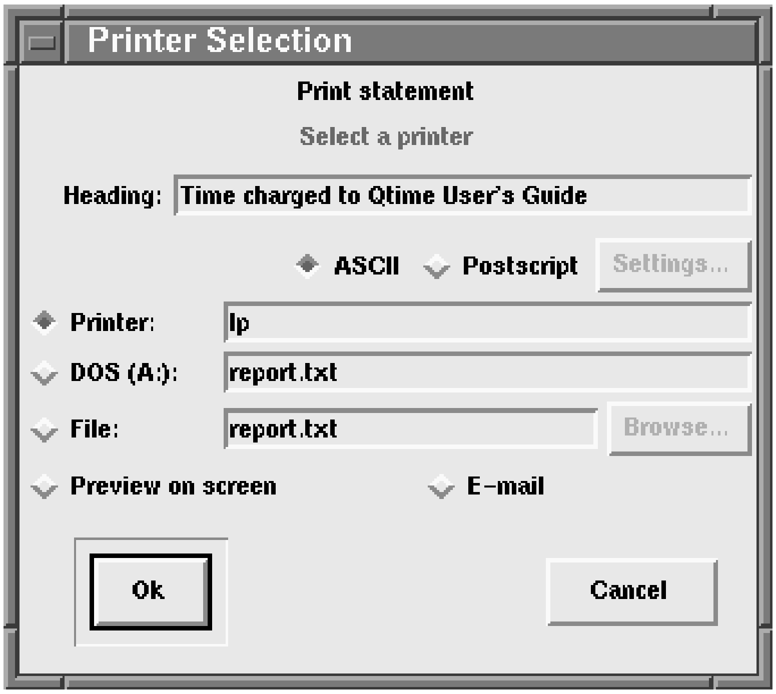

Figure 3.14: Print statement dialog box

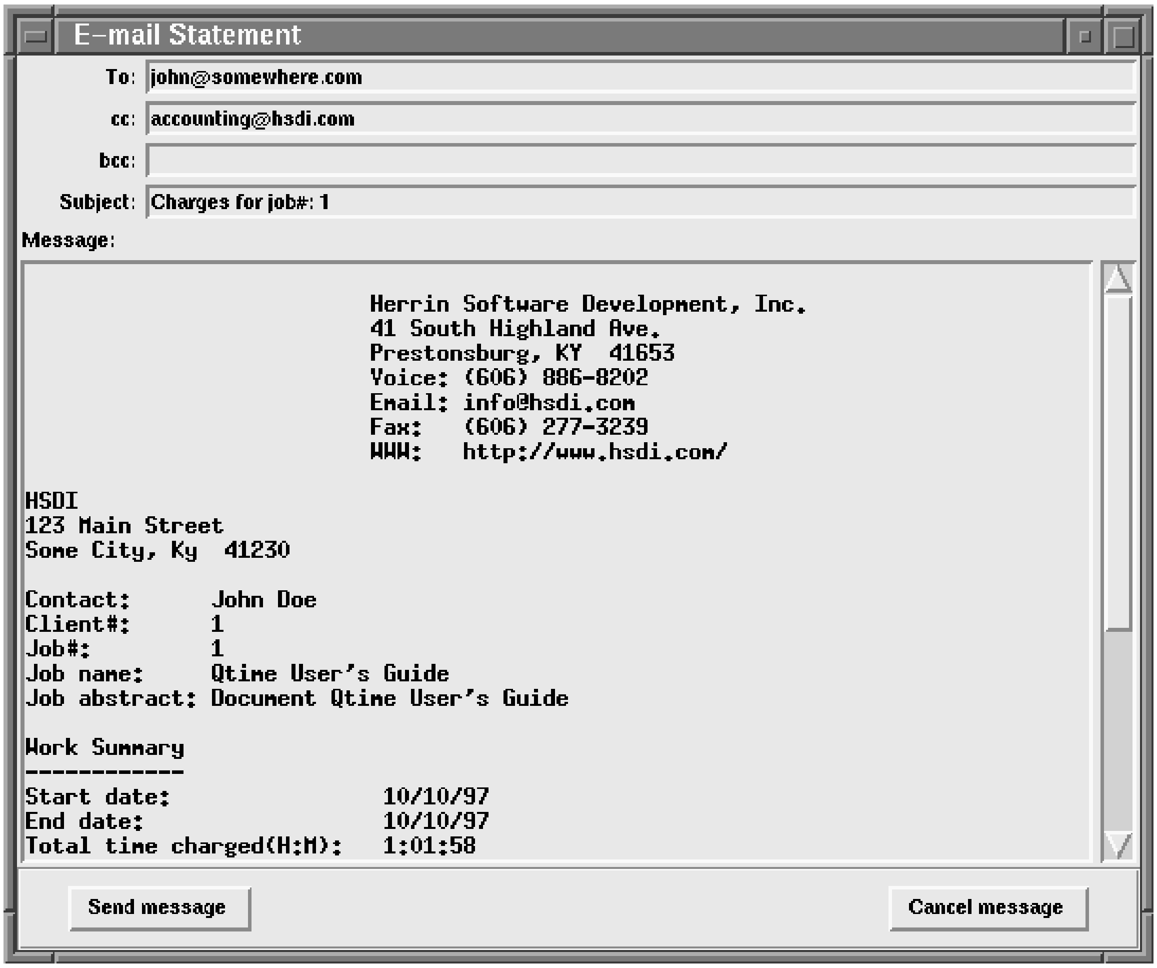

Figure 3.15: Email window

Chapter 4

About HSDI

Herrin Software Development, Inc.(HSDI) offers a wide variety of services including:

- Software development, customization
- Customer support
- Training
- Hardware consulting and support
- Internet services such as online support, web site host

4.1 Visit our Web Page

Visit our web page on the Internet, **http://www.hsdi.com**. From our web page, you can explore new products, updates, documentation or contact us through electronic mail at **info@hsdi.com**.

4.2 Write a letter

If you prefer, write a letter with your comments and mail to:

Herrin Software Development
Software Division
146 South Highland Ave.
Prestonsburg, Kentucky 41653