

TimeTrak  
by  
John W. Jaster  
**Offsite technical Solutions**  
**(714) 337-6637**  
**Compuserv: 73770,2233**

This Application is programmed in Visual Basic, and uses the Paradox Engine for Windows (pxengwin.dll). You need both Dll's to execute TimeTrak.

TimeTrak is a simple time tracking device which lets you enter a project name, the hours worked, date, and employee name. It keeps track of the totals for each project, and permits both screen and printer reports. I did it as an exercise in VB & Pdox, and also to learn something about printing reports. It is a good learning tool for anyone just getting started with the paradox engine under VB, and includes a declaration file (**TTRAKPXG.BAS**) of all the Paradox engine functions, as well as a file (**TTRAKGBL.BAS**) of all the constant global variables used by the engine. These two files can be included in other VB applications and used as is.

Use this application and code as you wish. It is not copyrighted and I don't carry grudges if you hack it up. All I ask is that you upload your own examples when they are ready. I am a senior partner in **Offsite Technical Solutions** and have a masters degree in Computer science, so if I can help with any questions, feel free to message me on CIS or call our Company number. \* **We are specialists in Downsizing mainframe applications to the world of Windows and beyond.**

Points of Interest

1) **Sorted files in Pdox** - the paradox engine insists that all keys be unique. Therefore, to create a file like timetrak.db that is sorted by project name, it is necessary to include a second field in the master key to force its uniqueness. I use record number. In this manner Paradox stashes each new record in order of project/recno, effectively giving me a sorted file. This cuts down on access time because when the SEARCHNEXT fails the first time, you can be sure that no other records of that key exists. I use this method in **project.click**.

2) **Using Header and Detail records** - using the sorted method above lends itself easily to the header/detail file maintenance approach. Each project/0 key is the beginning of a new project in the file and can thus be used to store totals. In TimeTrak I update the header (0 recordno) each time I add a record, keeping track of total hours, total entries, and last entry date. (notice that by adding a negative hours record the total is decreased). This saves a lot of I/O in displaying totals, especially with larger projects.

3) **Printing reports in VB** - both print methods **printer.print** and **form.print** are used in the application to either display a report in a form and hardcopy it to the printer, or print a report directly to the printer. There are some tricks in paging that could be used more effectively, but hey, I didn't get paid for this!

- 4) **Reducing the size of exe** - I purposely included all the function declarations for the pdox engine so that others can use them. However, in practice, when you are ready to compile for the final time in preparation for packaging (or whatever), it is best to go through both the declarations and the globals and delete any that this particular application doesn't use. This (along with the deletion of spaces and comments) drastically cuts the size of the final exe file.
  
- 5) **DBPATH constant** - if for some reason you can't access the timetrak.db file its probably because I use a global constant named DBPATH to point the program to where the table is residing. Look in the global file (**TTRAKGBL.BAS**) and change the constant to point to your particular drive\path.
  
- 6) **Home grown subroutines** - the function declaration file (**TTRAKPXG.BAS**) also includes some generic Paradox retrieval subroutines to make using the engine function easier. These are good examples of bundling similar functions into a generic subroutine. I have only scratched the surface here, but many of the functions lend themselves to such groupings.
  
- 7) **Invalid Object** - if you get this error it is probably because of the Fontname value set in the **Print\_Header** subroutine in the file **Report.bas**. Windows has this font but your selected printer driver does not. Change this to conform with your printer.