



# **CAPIX Job Scheduler User Guide**

**Version 1.1**

**December 2009**

# Table of Contents

<b>Table of Contents .....</b>	<b>2</b>
<b>Introduction .....</b>	<b>3</b>
<b>CJS Installation .....</b>	<b>5</b>
<b>Writing CJS VBA Functions .....</b>	<b>7</b>
<b>CJS.EXE Command Line Parameters .....</b>	<b>8</b>
<b>Windows Task Scheduler.....</b>	<b>12</b>
<b>Log File of Events .....</b>	<b>17</b>

## Introduction

The CAPIX Job Scheduler (CJS) is a software utility used to allow automatic execution of user-defined tasks and functions.

CJS is designed to be installed and configured by technical staff with programming skills. An advanced knowledge of the Microsoft Visual Basic for Applications (VBA) programming language is assumed.

The CJS software is not installed as part of other CAPIX applications. It is an additional installation that will cohabitate with other CAPIX software.

The purpose of CJS is to allow unattended and automatic execution of tasks and functions in the CAPIX applications. A common use of CJS would be to automatically generate batches of daily reports before the start of each day. CJS could also be used for other tasks such as importing electronic bank statements or market data etc.

A "stub" application program CJS.EXE is executed which specifies an "entry-point" function to be executed in an MS Access database (CJS11.mdb).

Scheduling software, such as Microsoft Scheduled Tasks, is used to run CJS.EXE at the required times and frequency.

Implementing CJS involves the following steps:

- Install the CJS software
- Write custom VBA functions in CJS11.mdb
- Add a Microsoft Task Scheduler jobs

- Test and review operation of CJS and custom functions

## CJS Installation

The CAPIX Job Scheduler (CJS) software can be downloaded from the CAPIX website ([http://download.capix.net/CJS/CJS\\_Setup.exe](http://download.capix.net/CJS/CJS_Setup.exe) ).

CJS installs using a standard MSI script generated using Installshield.

The default installation directory is:

*C:\Program Files\CAPIX\*

This is also the default install folder for other CAPIX desktop software.

The following files will be installed in this folder:

- **CJS.EXE** is the main executable program file that is used to automatically run CAPIX functions.
- **CJS.log** is a log file in ASCII text format that records the results of running CJS.EXE. Check this log file regularly to ensure CJS.EXE is being executed correctly. If CJS.EXE does not execute correctly, details of the error will be written to this log file.
- **CJS11.mdb** is an MS Access 2003 (version 11) database file that contains functions to be executed by CJS.EXE.
- **CAPIX\_Job\_Scheduler\_Guide.pdf** is an Adobe PDF file containing documentation about using CJS.
- **CAPIX\_Sample\_Task.job** is a sample Microsoft Scheduled Tasks file which can be copied into the Scheduled Tasks folder as an example of how to automatically execute CJS.EXE.

Note: CJS.exe requires that Microsoft Access 2003 (version 11) or later be installed on the local PC for operation.

CJS.exe will operate successfully with either the full or run-time versions of MS Access.

## **Writing CJS VBA Functions**

The CAPIX Job Scheduler (CJS) is a software utility used to allow automatic execution of user-defined tasks and functions.

## CJS.EXE Command Line Parameters

The program file CJS.EXE takes a command line function that specifies a VBA function to execute.

CJS.EXE allows a VBA function to be executed via a command line parameter. This simplifies the complex task of executing a VBA function within the CAPIX environment and security framework.

It is likely multiple Task Scheduler jobs or Program File shortcuts will be created to execute various VBA functions using CJS.EXE.

The CJS.EXE application accepts one mandatory parameter and four optional parameters:

```
CJS.exe <Function_name> [Database] [Workgroup] [UserName] [Password]
```

Where:

<function\_name> - name of the VBA function to execute. This is a mandatory parameter. This parameter should have following format:

```
fnSomeFunction('this is sample string param',4567,34567)
```

where **fnSomeFunction** is sample VBA function inside MS Access module, which has 3 arguments: first string argument and two integer arguments.

Or simply:

```
fnSomeFunction2(), if function does not have any arguments.
```

Also, please follow following rules when specifying function arguments:

- 1). All function arguments must be included in the parenthesis.
- 2). Arguments must be separated with coma.
- 3). Do not use spaces between arguments (before or after coma).
- 4). Only 2 types of function arguments are supported: text and integer.
- 5). All text arguments must be wrapped with single quote (') characters. You can use spaces only inside text arguments.

6). **NOTE: If you used spaces in this first parameter (e.g. in one of the function arguments) you must wrap it with double quotes ("), for example:**

**CJS.exe "fnSampleFunction('hello world and bye',545)"**

*[Database]* - name of the database to open. This is an optional parameter which allows multiple MS Access databases to be used. **NOTE: if path to the database contains spaces you must include it within double quotes (") characters, e.g.**

CJS.exe fnSomeFuntion() "C:\Program files\Some path\some\_db.mdb"

Default value is "C:\Program files\CAPIX\CJS11.mdb"

*[Workgroup]* – path to the security workgroup file (.mdw) to be used. This is an optional parameter.

Default value is "**C:\Program files\CAPIX\CAPIX.mdw**"

*[UserName]* – valid username within above workgroup file. This is an optional parameter. If you change default location of the workgroup file you must supply valid username as well.

*[Password]* – valid password within above workgroup file. This is an optional parameter. If you change default location of the workgroup file you must supply valid password as well.

For example, if a VBA function called gfnSampleFunction() was defined in the CSJ11.mdb database, this function could be executed directly from the command line using the following syntax from the command prompt or a short cut:

```
C:\Program Files\CAPIX\CJS.EXE gfnSampleFunction()
```

In this example CJS.EXE will

- Start CJS.EXE
- Open CJS11.mdb
- Initiate OLE automation
- [Acquiring the pointer to Access.Application COM object.](#)
- Execute the target function i.e. gfnsampleFunction()
- Close CJS11.mdb
- Exit CJS.EXE

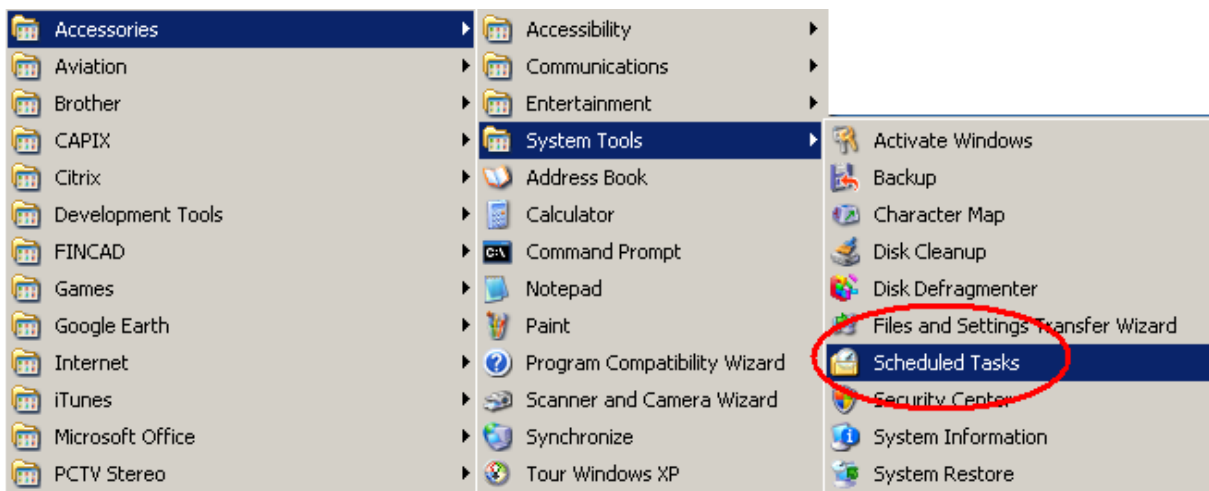
## Windows Task Scheduler

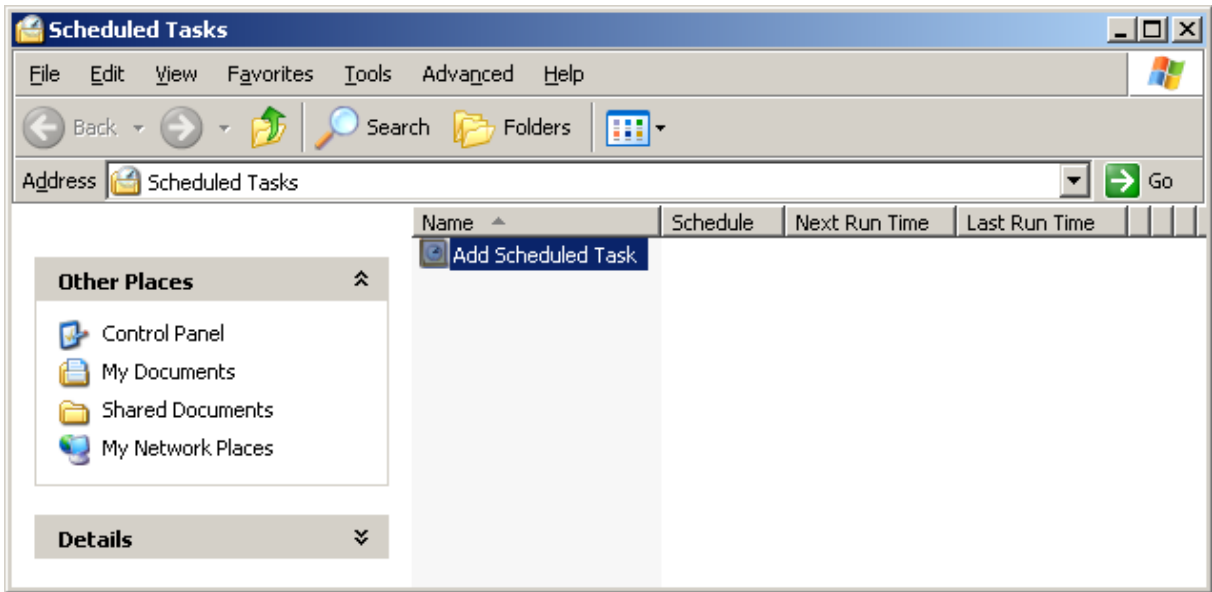
There are several different software packages that can be used as schedulers in conjunction with CJS.EXE.

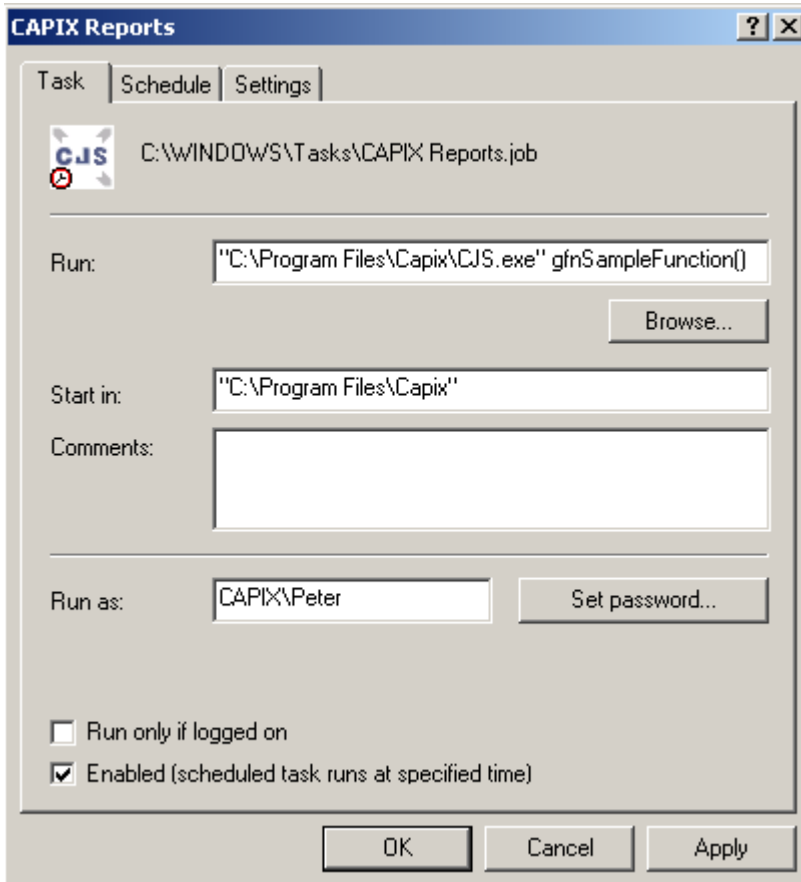
We recommend the standard *Windows Task Schedule software* as it comes standard with Windows, does the job well and is free. Other similar software packages could also be used.

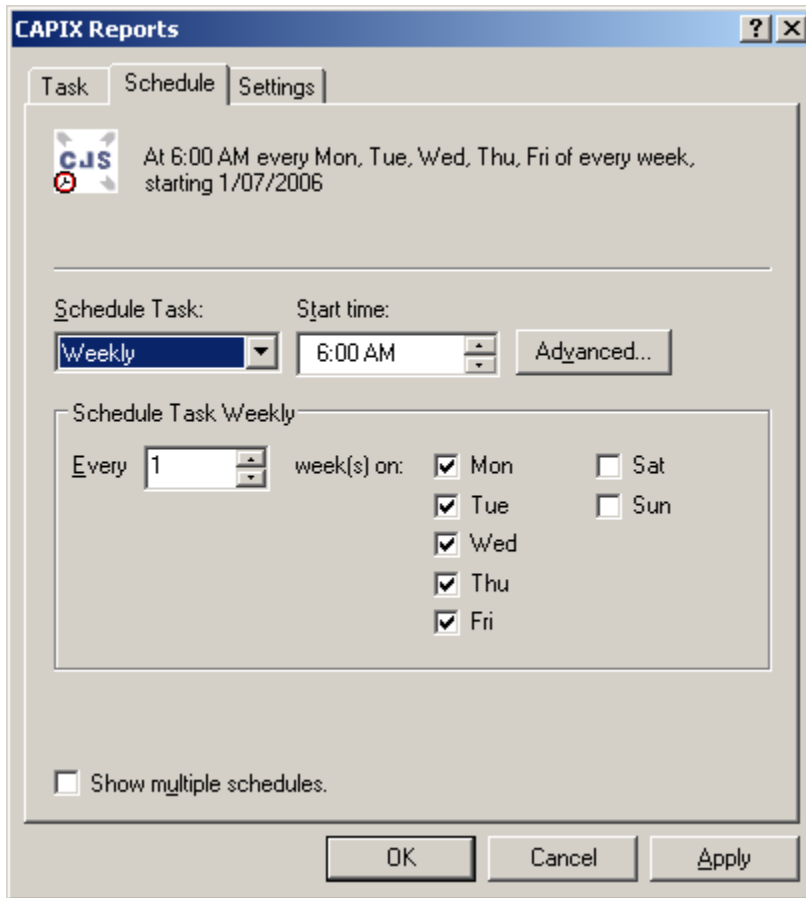
To start the *Windows Task Schedule software*, select the following menu items from the Windows start menu:

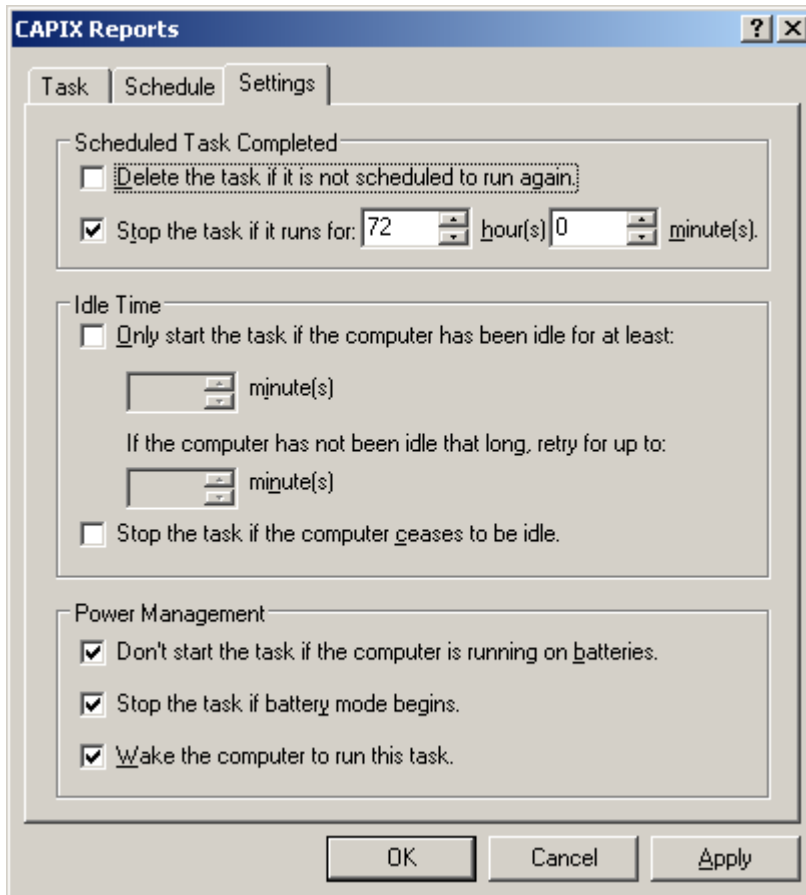
- Start Menu
- Accessories
- System tools
- Scheduled Tasks











## Log File of Events

Every time CJS.EXE is executed, the application results are written to a log file **CJS.log**.

The full path to this ASCII text file is:

*C:\Program Files\CAPIX\CJS.log*

A time stamp shows when each entry in the log file was generated.

The results for both errors and successful completion of the tasks are stored in this log file.

It is recommended that CJS.log be reviewed regularly to ensure the tasks run using CJS.exe are being completed as expected.

CJS.log is also a useful tool for testing new jobs being written for CJS.exe.

All successfully executed jobs are marked with "JOB INFO" prefix at the beginning of the log line. All errors are marked with "ERROR:" prefix.

The test file CJS.log may be periodically deleted or cleared if the existing entries are no longer required. CJS.exe will create a new log file if none already exists.

CJS.exe will automatically remove and create new log file if the size of the existing CJS.log has grown to 10MB.