CoreView
Complete Diagnostic Solution

Table of Contents

Introduction

What is Core View?

Core View is a complete computer diagnostic utility designed for professionals. It can read and display useful data about the system it runs on without requiring additional administrative rights or tools.

Core View displays hardware information, software information, drivers, address maps, conflicts, processes and even event logs along with real-time graphs and SMART-powered hard disk data.

It can also store any gathered information into a database, known as the Heuristic Database, so it is able to precisely compare the computer it is running on to previous machines at the press of a button – unveiling a completely new way of aiding the diagnosis of computer faults.

Requirements

Core View has been designed to run with as little requirements as possible, so it self-contains nearly all the files it needs to operate. The only requirements that Core View needs are:

- One of the following versions of Windows
    - o   Windows 2000
    - o   Windows XP
    - o   Windows Vista
    - o   Windows 7
- Microsoft .NET 2.0 (installed by default as of Windows XP SP2)
- The Windows Management Instrumentation Service (enabled by default on all mentioned operating systems)
- A storage location available for both reading and writing to store the program. This could be from a hard disk, USB thumb drive or even over network. As long as you can write to it, it will work. Recommended capacity: 500MB or more.

First Use

Core View is a standalone program meaning it can be run by itself without any installation. Simply copy the CoreView.exe file to a writeable location and double-click it to run.

Core View will then proceed to unpack any additional files it needs. This includes 2 DLL libraries, a user-editable XML configuration file and finally, the Heuristic Database Heuristics.db file (after the splash screen has closed).

You will proceed to see a splash screen (if enabled in configuration) which will show the progress of information loading. The program will then show the main window, where information is either already processed or still being fetched.

Interface

Splash Screen

This screen details the current operations that are being processed when the application starts. This is represented by the loading bar – which gives an *estimate* of the loading progress.
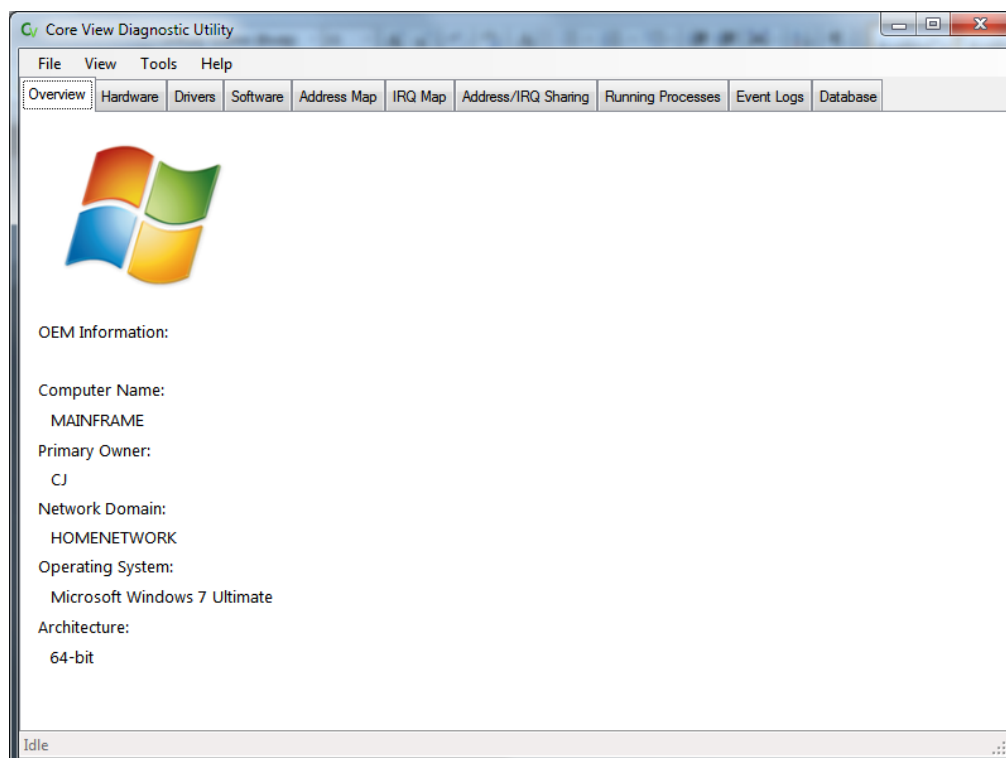

The splash screen can be skipped by pressing the 'Skip' button in the top right of the screen. This will still allow continuation of loading, but the main form will be shown immediately.

The splash screen can be toggled in Options.

Main Window

The main window is split up into sections denoted by the various tabs towards the top of the window.

These sections will show the respective information once clicked. Everything that the program has managed to gather will be displayed here. These tabs are designed to be able to be browsed with keyboard, mouse or both. Keyboard shortcuts are available for all tabs.

Furthermore, the native resolution of the program is 800x640, so it is suitable for all screen sizes without adjustment but is also manually expandable/contractible.

The 'Hardware' tab contains another set of tabs to further separate each component in the computer. Then a drop-down box in each of these tabs allows you to browse any specific hardware instance – such as the 3rd stick of RAM in the computer. The information displayed is refreshed each time a tab or drop-down box is changed.
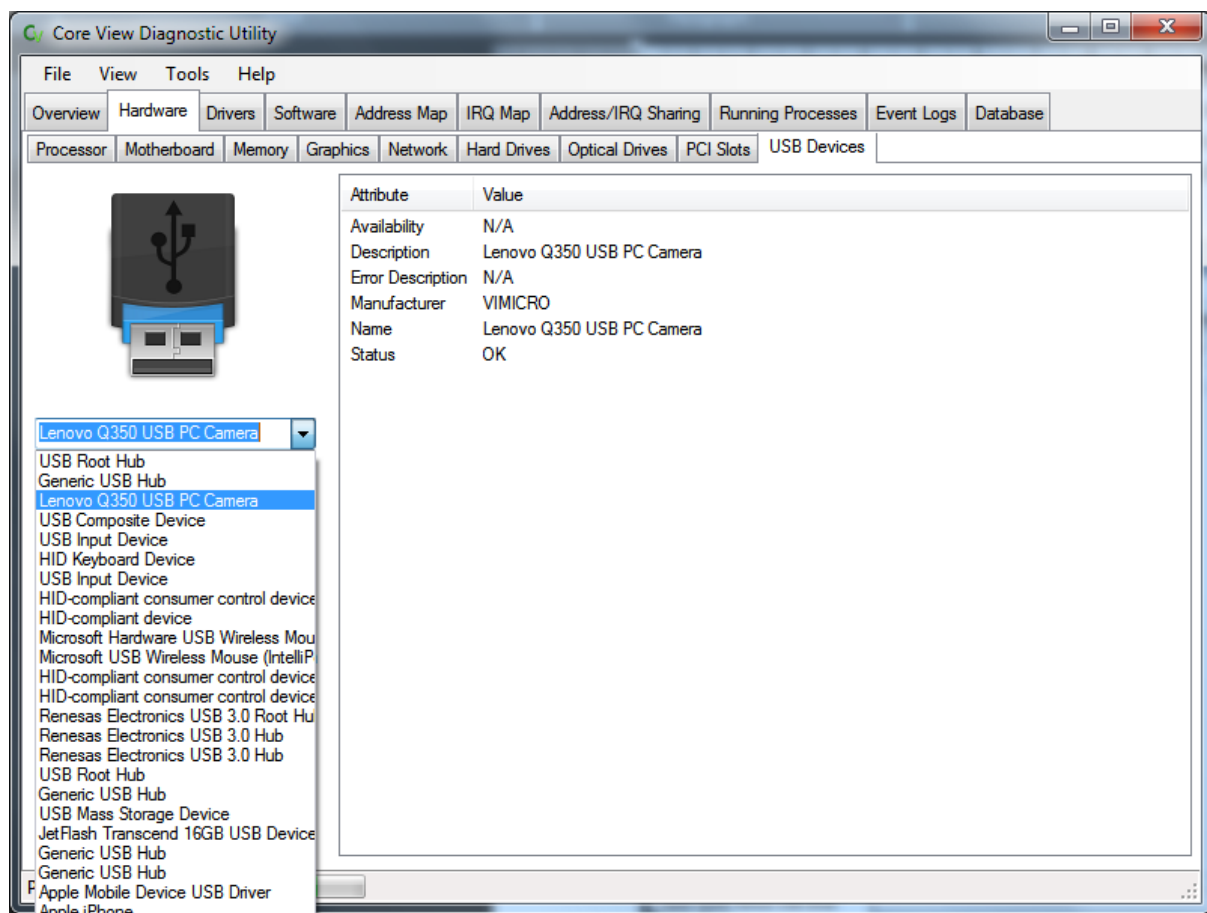

The information is split into a quick-info panel, and a full information box. The full information box, on the right hand side of the form, will list every attribute for the element in concern. In some cases, there may be more than one of these to separate the data; for example, Hard Drives and SMART Data are split into two different containers.
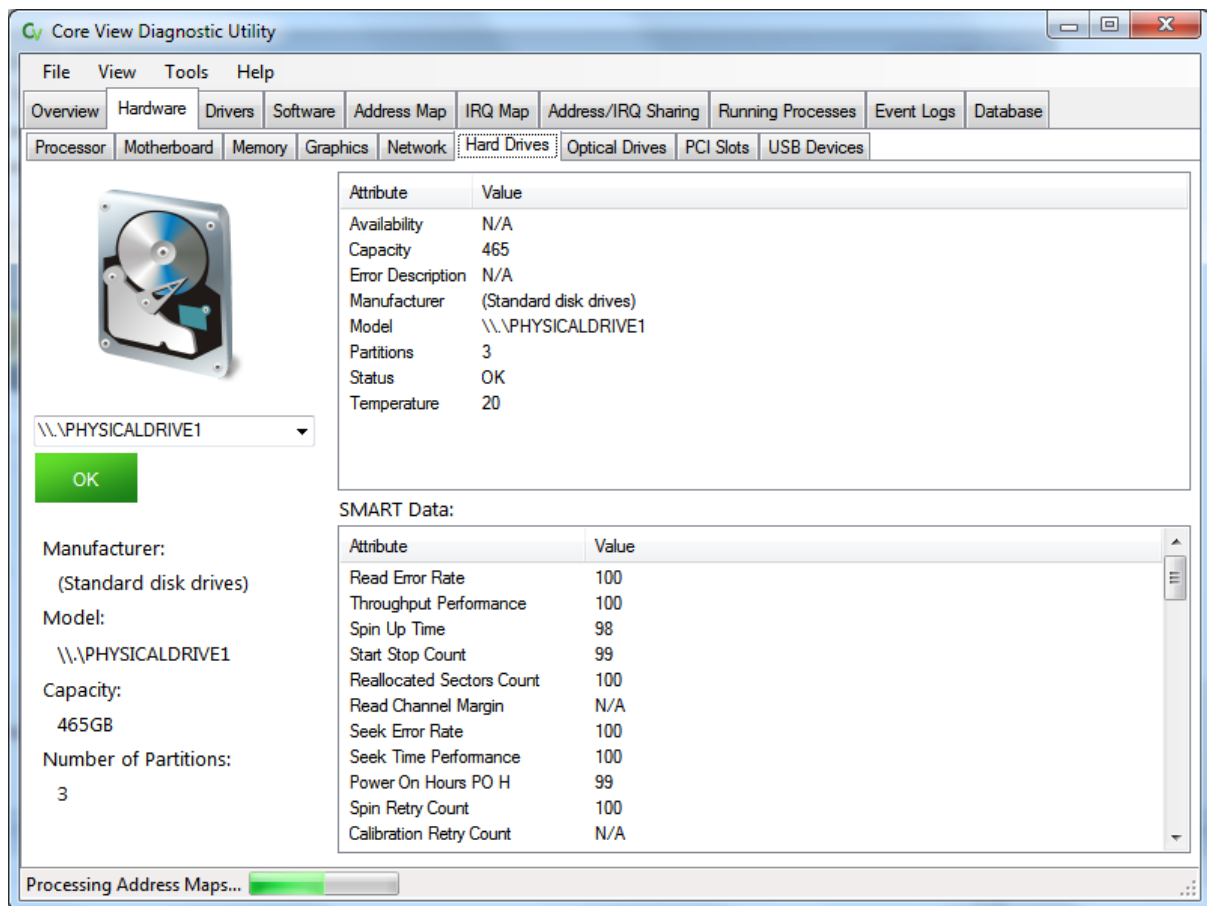
The quick-info panel displays a brief overview of the selected element. This is shown as text towards the bottom-left of the window.
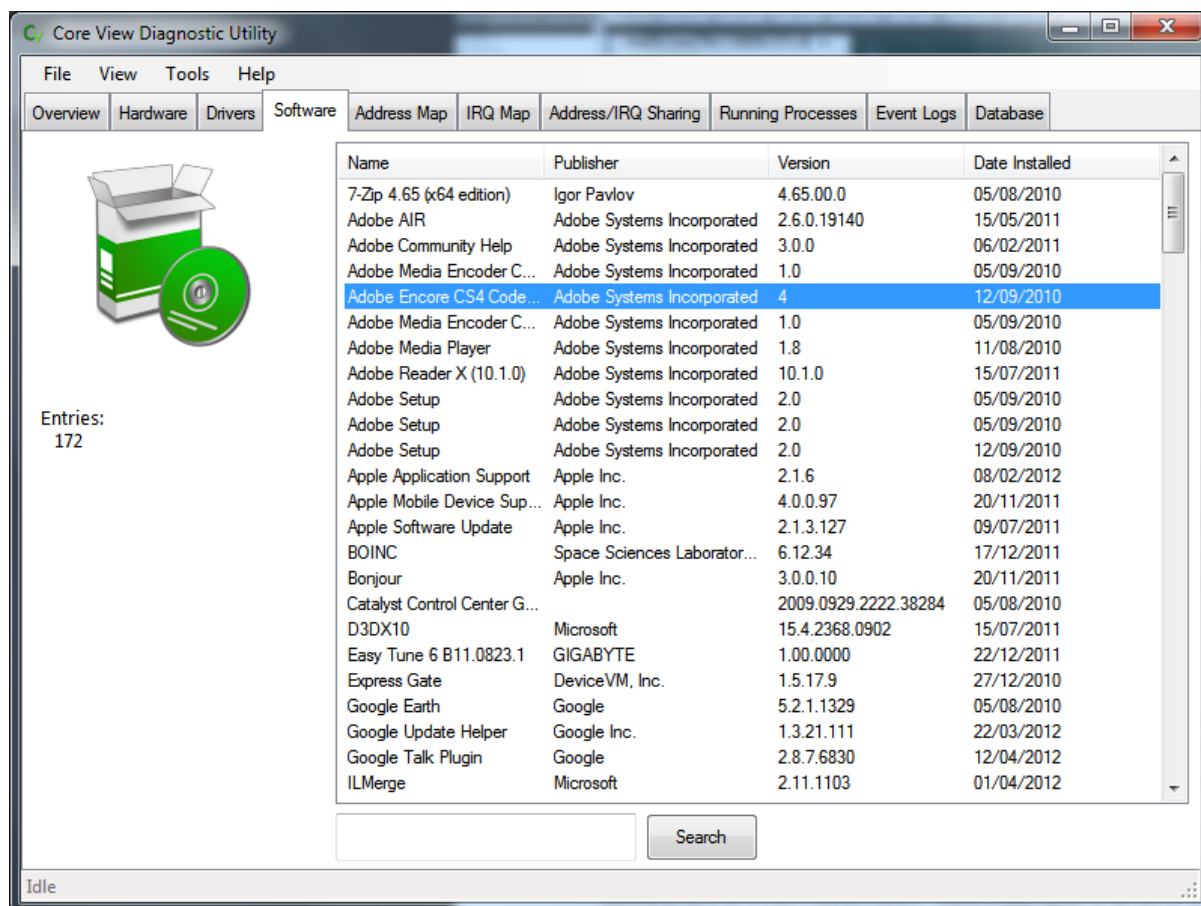
If any information is clipped off because it's too long, hovering your mouse cursor over the text will reveal the full information in a tooltip (see next page).
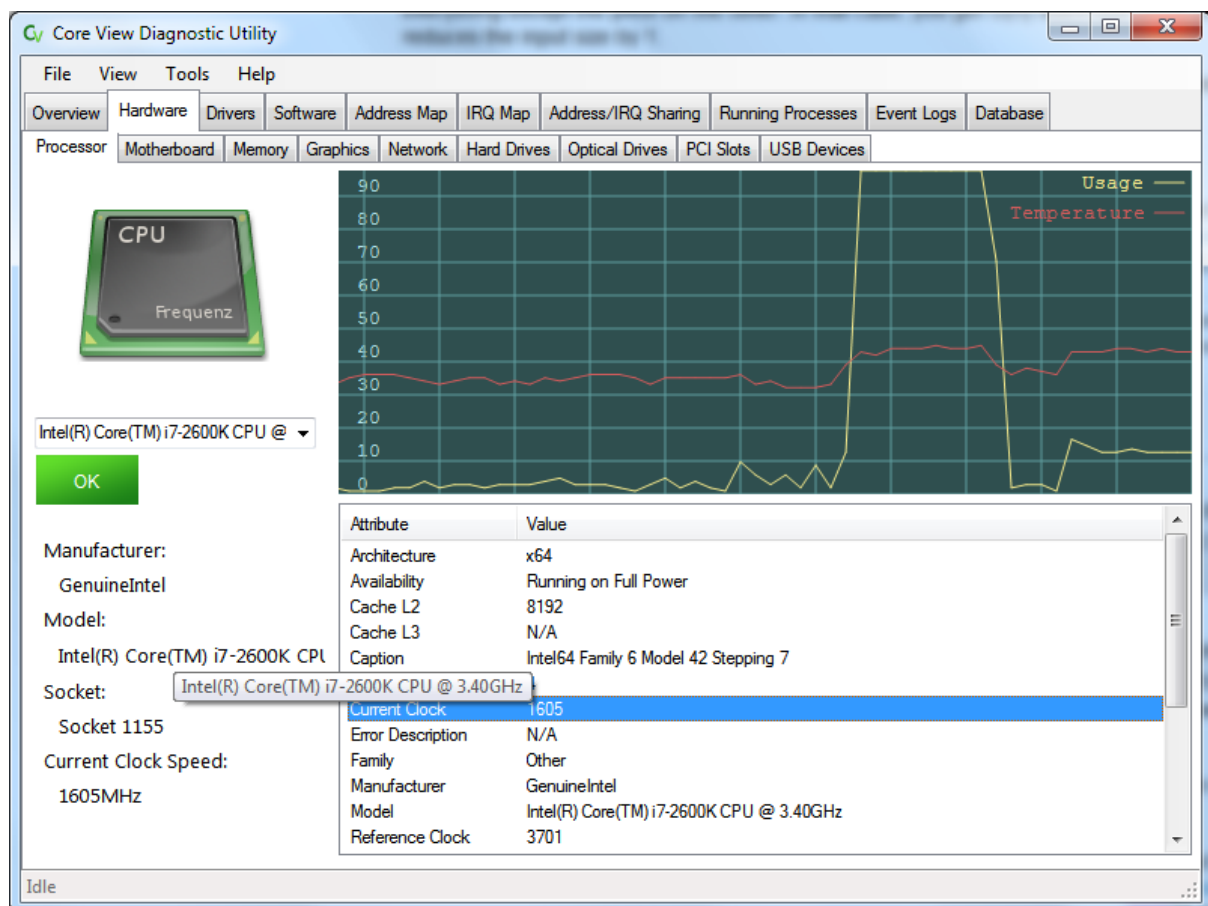
The tabs 'Drivers', 'Software', 'Address Map', 'IRQ Map, 'Address/IRQ Sharing', 'Running Processes' and 'Event Logs' are all considered *Software* tabs.

These tabs list information in the full information box slightly differently to the way hardware is displayed. Here, each instance of software will be displayed as a new row in the box; showing details of the entry such as date installed and version rather than having a drop-down box to select an instance (see next page).

Also present on these 'software tabs' is a search box. The search box can be used to find items in the list by highlighting any matches it finds. All columns will be searched.

Name	Publisher	Version	Date Installed
7-Zip 4.65 (x64 edition)	Igor Pavlov	4.65.00.0	05/08/2010
Adobe AIR	Adobe Systems Incorporated	2.6.0.19140	15/05/2011
Adobe Community Help	Adobe Systems Incorporated	3.0.0	06/02/2011
Adobe Media Encoder C...	Adobe Systems Incorporated	1.0	05/09/2010
Adobe Encore CS4 Code...	Adobe Systems Incorporated	4	12/09/2010
Adobe Media Encoder C...	Adobe Systems Incorporated	1.0	05/09/2010
Adobe Media Player	Adobe Systems Incorporated	1.8	11/08/2010
Adobe Reader X (10.1.0)	Adobe Systems Incorporated	10.1.0	15/07/2011
Adobe Setup	Adobe Systems Incorporated	2.0	05/09/2010
Adobe Setup	Adobe Systems Incorporated	2.0	05/09/2010
Adobe Setup	Adobe Systems Incorporated	2.0	12/09/2010
Apple Application Support	Apple Inc.	2.1.6	08/02/2012
Apple Mobile Device Sup...	Apple Inc.	4.0.0.97	20/11/2011
Apple Software Update	Apple Inc.	2.1.3.127	09/07/2011
BOINC	Space Sciences Laborator...	6.12.34	17/12/2011
Bonjour	Apple Inc.	3.0.0.10	20/11/2011
Catalyst Control Center G...		2009.0929.2222.38284	05/08/2010
D3DX10	Microsoft	15.4.2368.0902	15/07/2011
Easy Tune 6 B11.0823.1	GIGABYTE	1.00.0000	22/12/2011
Express Gate	DeviceVM, Inc.	1.5.17.9	27/12/2010
Google Earth	Google	5.2.1.1329	05/08/2010
Google Update Helper	Google Inc.	1.3.21.111	22/03/2012
Google Talk Plugin	Google	2.8.7.6830	12/04/2012
ILMerge	Microsoft	2.11.1103	01/04/2012

Entries:
172

Search

Idle

If you chose to skip the splash screen, then some of these tabs may not be populated. In this case, you may see that there is a loading bar at the bottom left of the window. This works the same way as the splash screen's loading bar.

Once the bar denotes that the section of information you are looking for is loaded, the tab will be populated. Note that the information will not be automatically added to the information box – you must leave and re-enter the tab to reload the data (see next page).
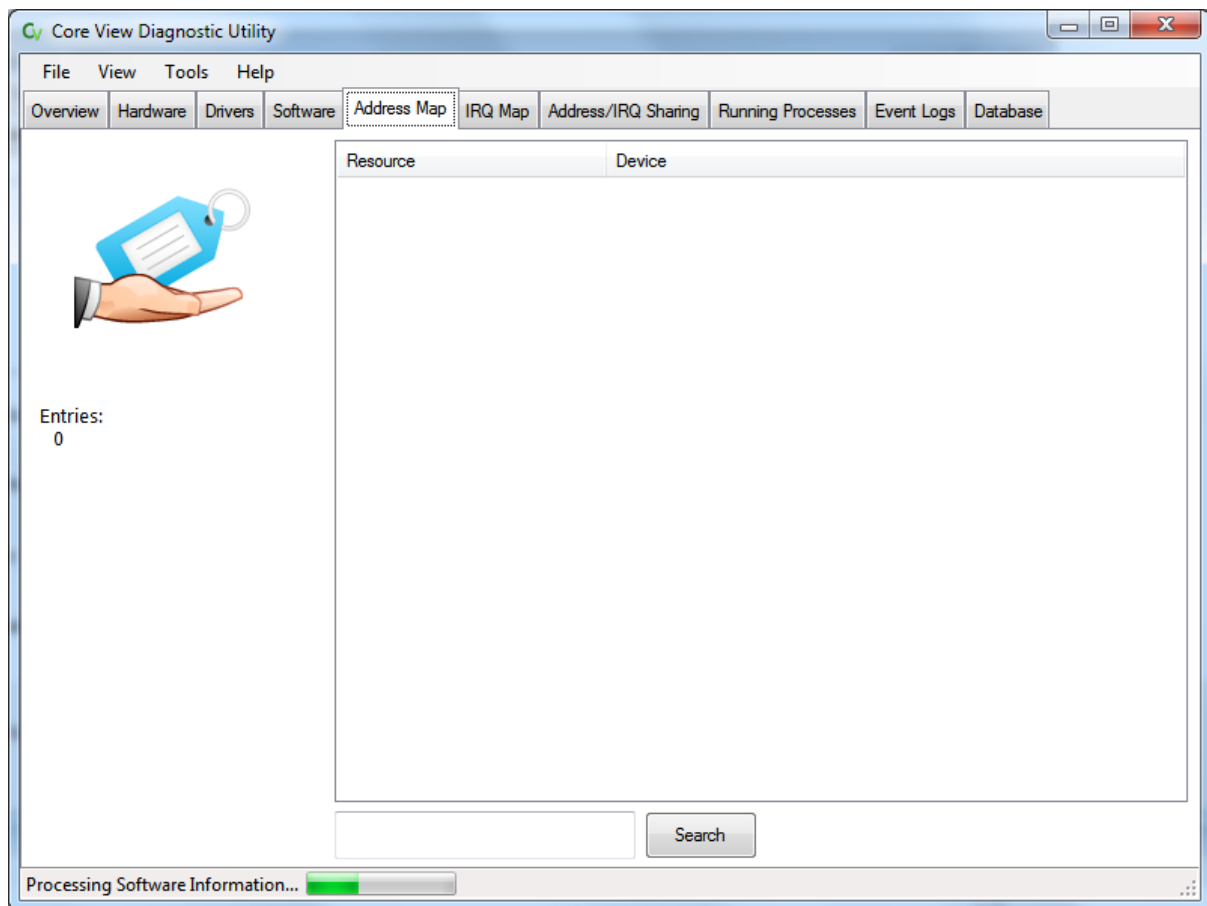
If instead you chose 'Load hardware at startup' or 'Load only on tab switch', then some or all of the form's data will be only be loaded once the tab is selected. This may take some time to complete, so please be patient.

Using these modes is designed to allow the user to see a single piece of information quicker than having to load everything first.

Options

The options dialogue can be viewed by pressing F8 on the keyboard, or navigating to Tools ➜ Options. Here, common settings can be altered to customise the running of the program. These changes are stored in the file 'config.xml', in the working directory of the program.
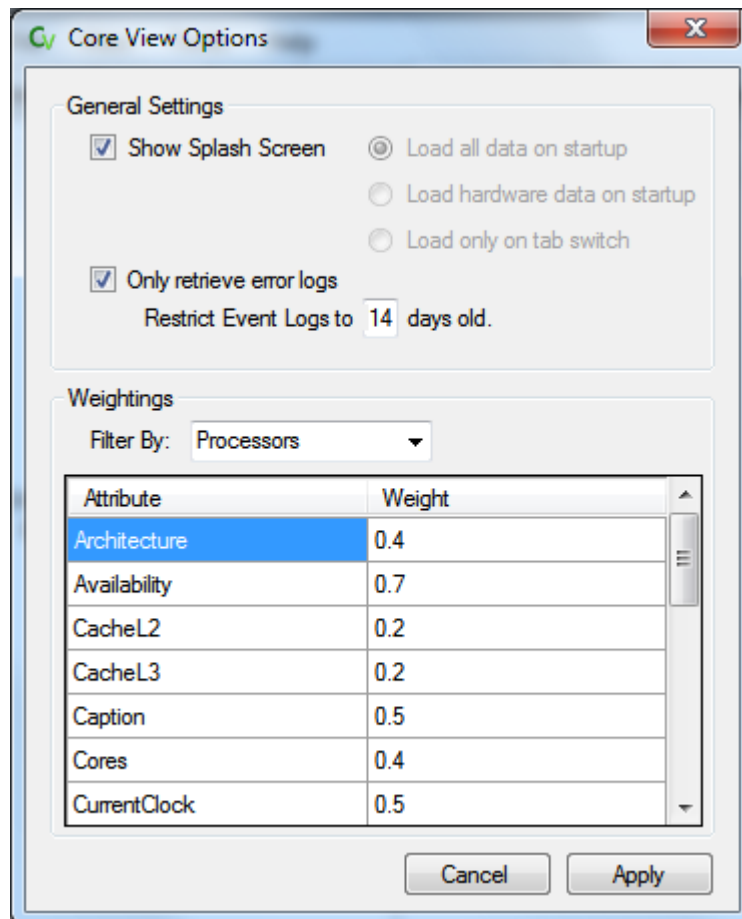
Main Options

Show Splash Screen

> Toggles whether the splash screen is shown or not. If the splash screen is to be displayed, then all data must be loaded on startup.

Load all data on startup

> All data is loaded sequentially when the program starts. This does not affect viewing the main form, as the splash screen can be skipped at any time if enabled.

Load hardware data on startup

> Only hardware data is loaded when the program starts. Software information will be loaded when the respective tabs are selected. Hardware loading is very fast compared to software, so this option is beneficial if you only want to perform hardware monitoring.

Load only on tab switch

No information is loaded when the program starts. All information must be gathered when the tabs are switched in the main program.

Only retrieve error logs
Event Logs come in several flavours – including 'Error' and 'Information'. Because Information logs take up the bulk of the space, but are relatively useless, this option saves a lot of time and resources; so it is recommended to keep enabled.

Restrict Event Logs to x days old
Only retrieves Event Logs if they are newer than 'x' days old.

Weightings

The Options dialogue also provides a graphical way of editing weightings. Weightings are the numbers used to compare different elements of the computer system and are ranked by importance. The more important one particular component is, the higher its weighting will be.

Weightings are split into groups like hardware instances in the main window. Selecting an element from the drop-down list will expose the relevant weightings for that particular class.

Weightings are typically between 0 and 1, but they can technically be any decimal number, even negative!

The example below shows the weighting for the Graphics Adapter class. Architecture has a 'neutral weighting', which is considered to be 0.3. Availability has a much higher weighting of 0.7, which means it's quite an important factor when comparing for faults with the database. BitsPerPixel, however, isn't as important and it has a weighting of 0.2 (see next page).
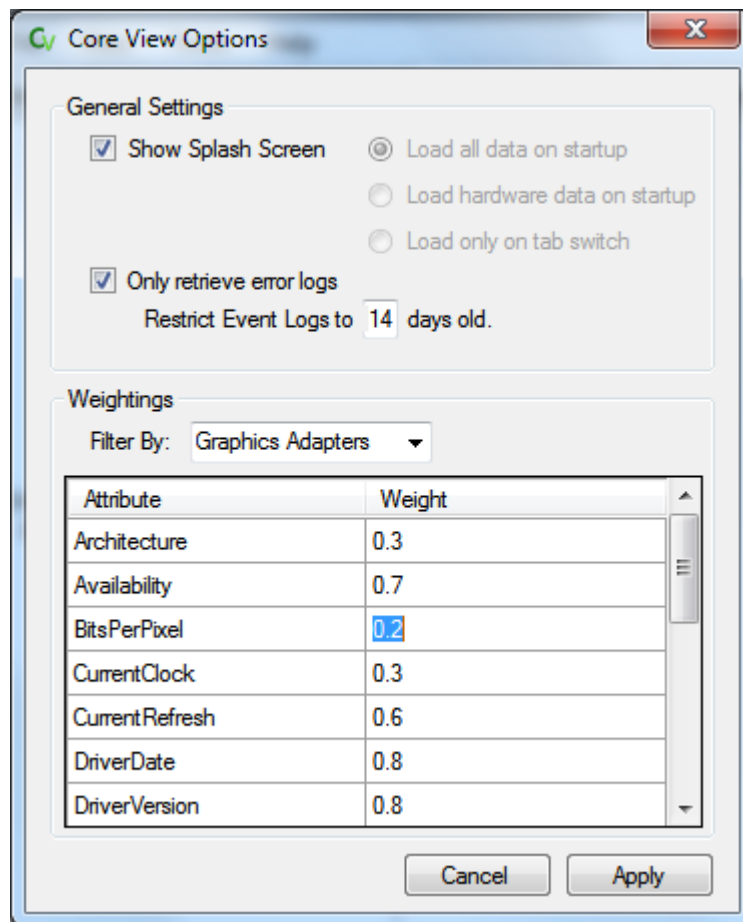
The lower portion of the 'Filter By' list contains 'Complaint weights'. These are weightings that designate the importance of individual classes for a given complaint. For example, a Graphics Adapter is far more important than an Optical Drive when considering screen issues. Therefore, its weighting should be higher (and the Optical Drive should really be 0).

Like general settings, weightings are also stored in 'config.xml', and can be edited with any text editor by opening the file. If weights are edited while the program is in use, the configuration file should be reloaded by pressing F12 or going to File ➜ Reload Configuration.

Database

The Heuristic Database section is the most significant part of the program. It's accessed via the Database tab on the main form, and has display boxes similar to other 'Software' tabs.

The Heuristic Database stores information about the computer the application is running on for later reference. Once stored, they can be compared to other computers that the application runs on, and a numerical score is returned showing the likeness of each computer.

This score is calculated using weightings, which can be edited with the Options dialogue. The actual database file is 'Heuristics.db', which is written with SQLite.

To start comparing previous computers stored in the database, click 'Start Search'. The current computer does not have to added to the database in order to do this.

As the search is executed, entries will be added to the information box on the right side. Clicking the entry will expand the comments into the box below.
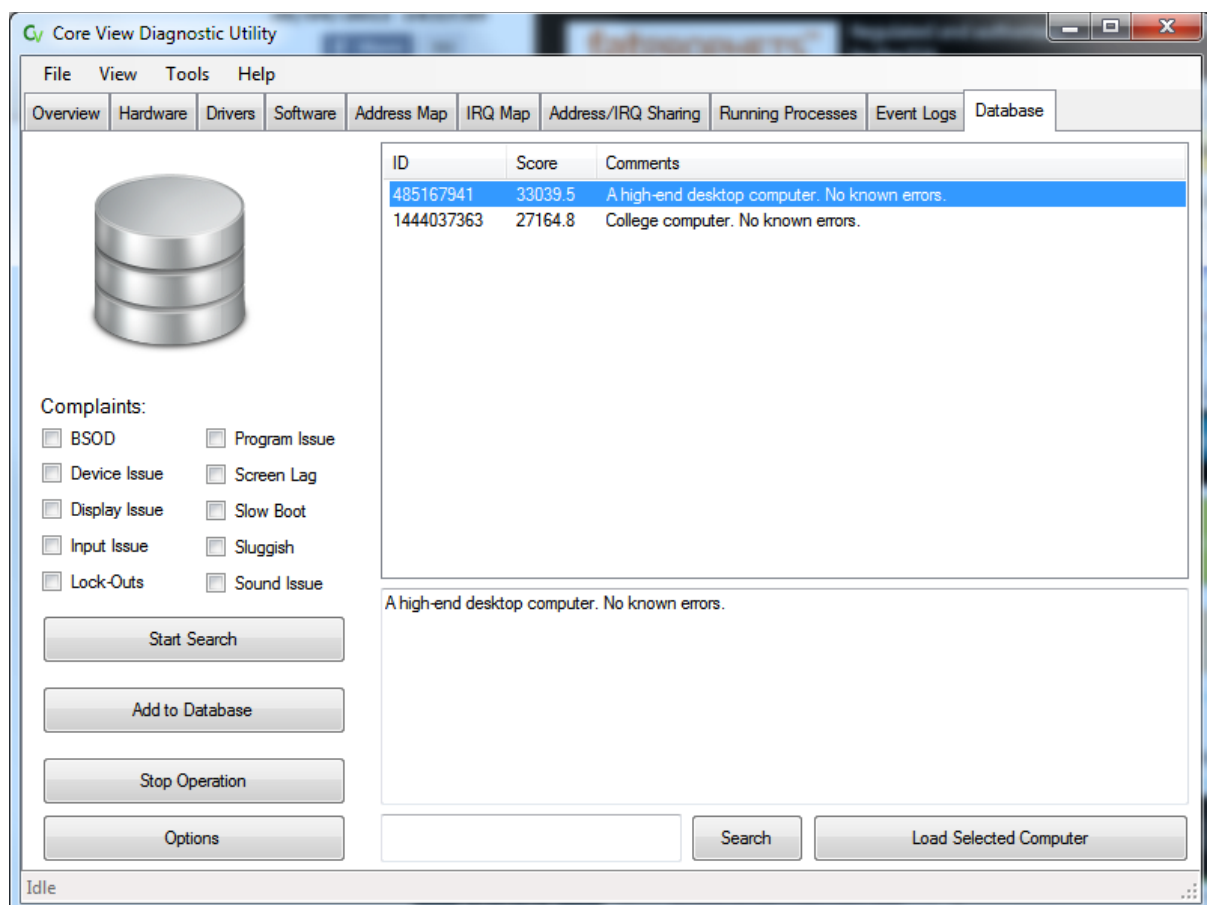
After the search is complete, previous computers can be loaded into the program as if you were running the program on them. This is done by clicking 'Load Selected

Computer'. After you are finished, the same button can be pressed again to return the program to its normal state.
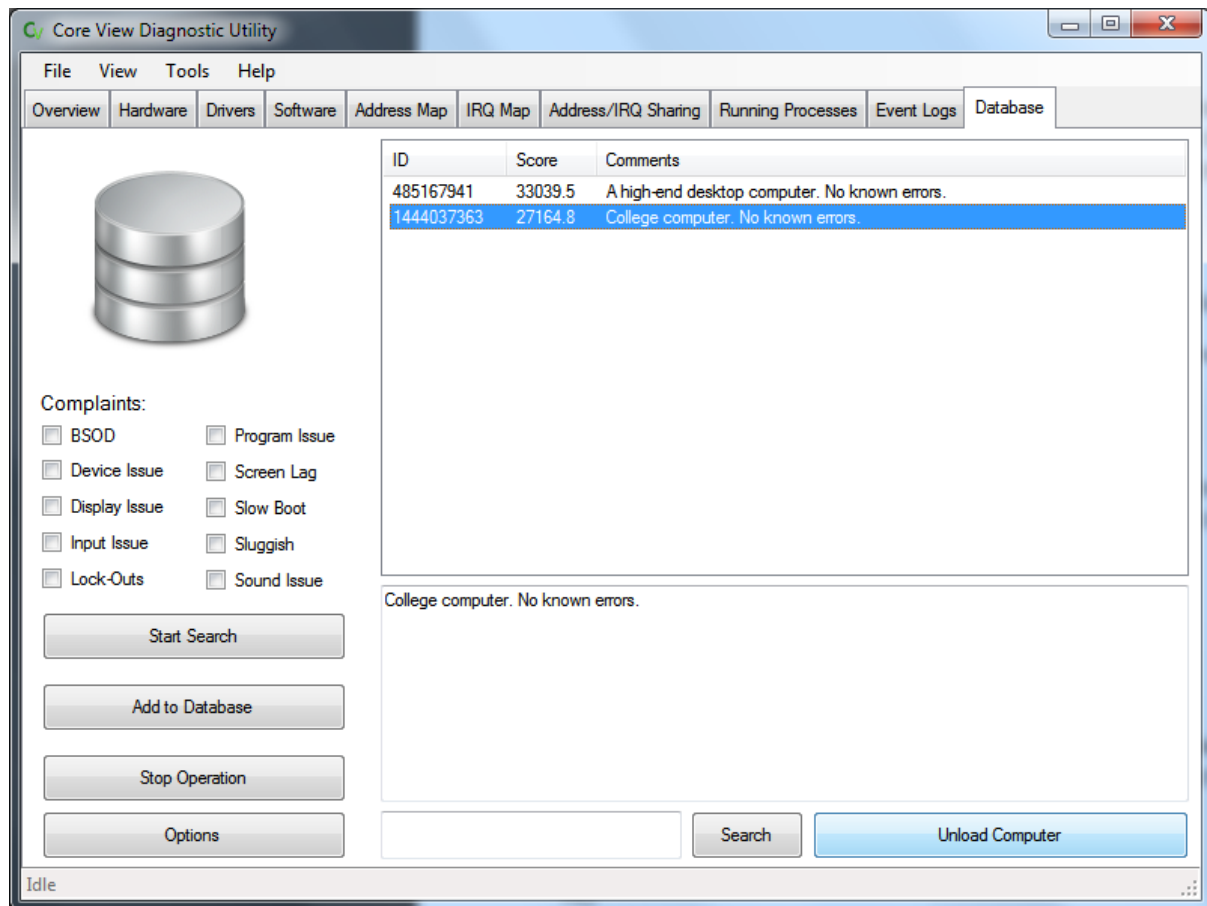
If you wish to add the current computer to the database, click 'Add to Database'.

A dialogue will ask you for comments. If needed, add information about any found faults (and how they may be fixed) to refer to in the future.
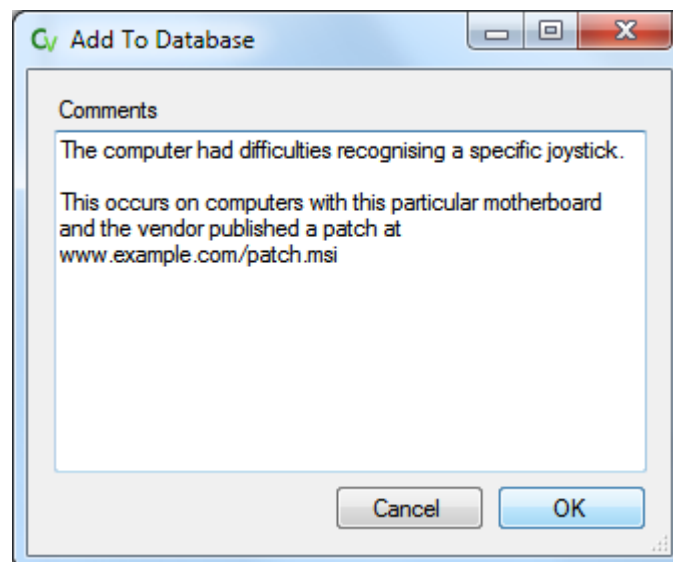
Note this operation will take longer than searching for computers, so please be patient. The amount of data recorded will also slow this process down.
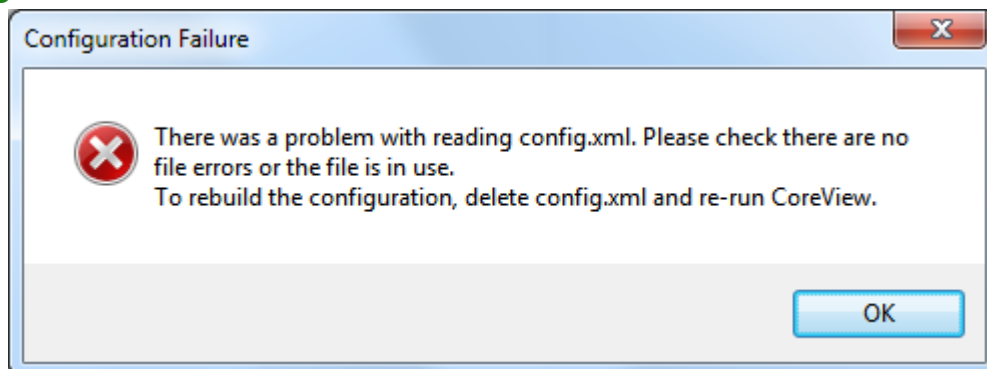
Errors You May Encounter

Unable to load dependency files
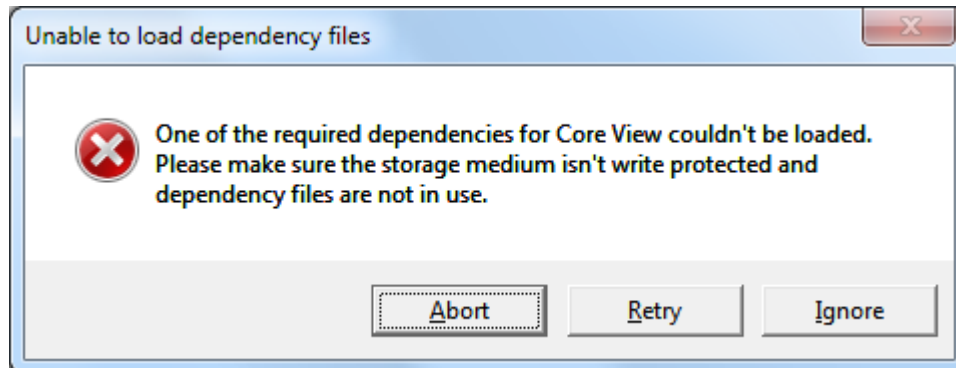
Typically this error will be encountered if Core View is not able to properly read and write to the folder it is in. Make sure the folder is writable and that no other programs are using the two DLL files.

Configuration Failure

This error is caused by a configuration file that isn't in the correct format. If 'config.xml' has been manually edited, please check that all tags have been properly closed and all nodes are present.

Also make sure that the file is readable. Close any programs that may be using the file and try again.

In the event that the configuration file still cannot be loaded, copy 'config.xml' to a backup location and delete the original file. When Core View is re-run, it will create a new one from defaults.

If you wish to keep your previous settings, you should then open the files in a text editor of your choice to copy/paste any desired nodes.
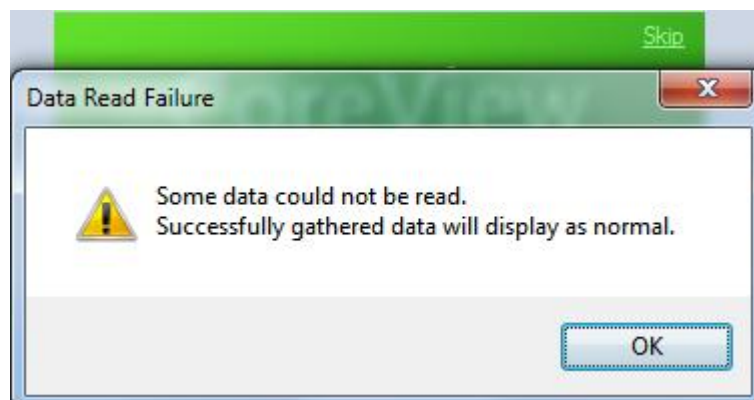
Permission Denied
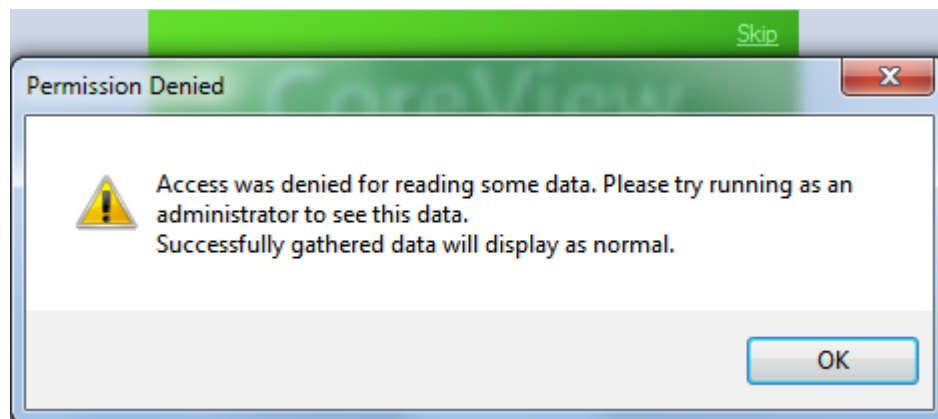
This error will display as Core View gathers information if there is a permission problem with any of the sources. This is most often to do with getting SMART data for hard drives, which is often protected.

Running the file as administrator as shown below may fix this issue, but otherwise, the program will continue as normal.
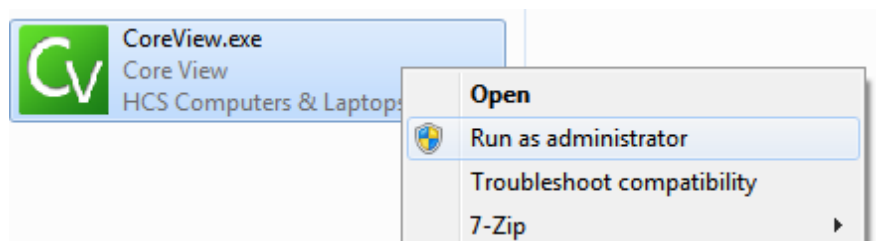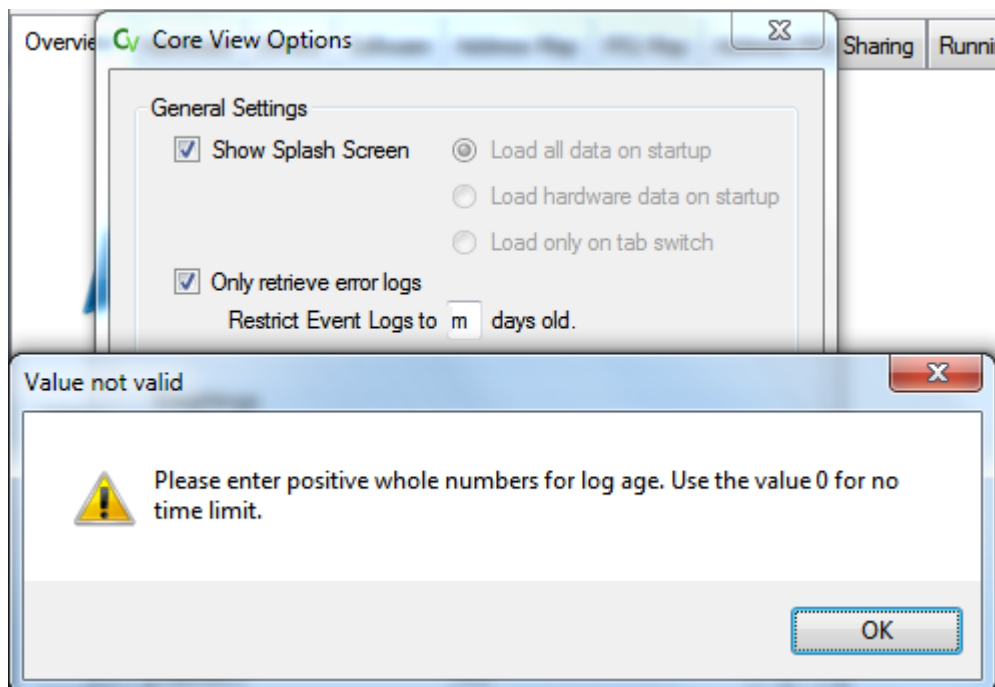

Data Read Failure

Like the Permission Denied error, this will show during information gathering if Core View found data, but was unable to retrieve anything meaningful. This is generally not fixable as it often means the computer gave an empty information container to the program.
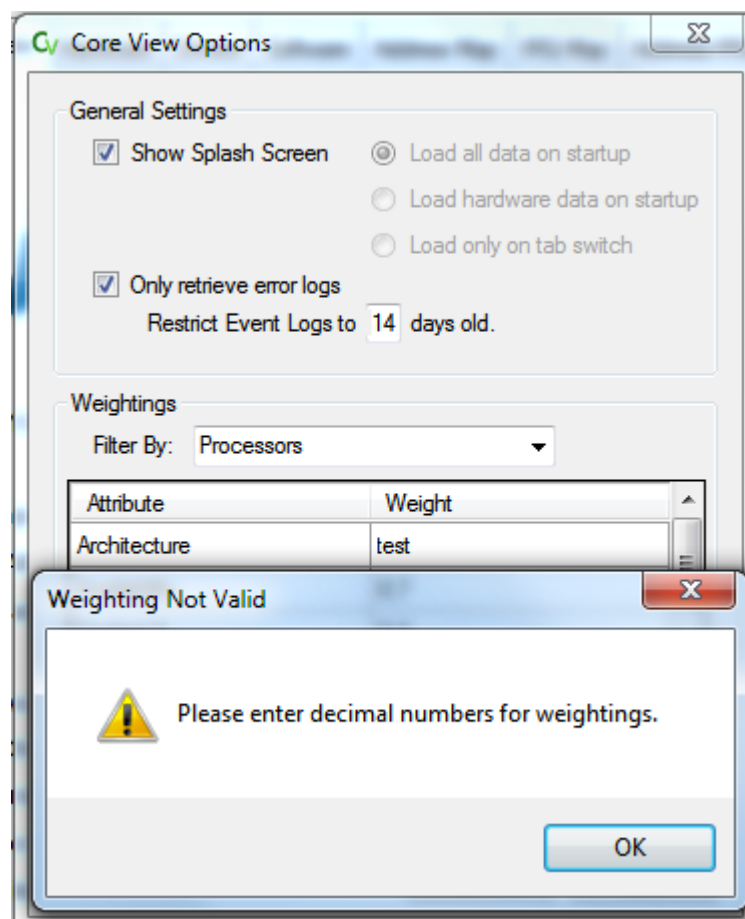
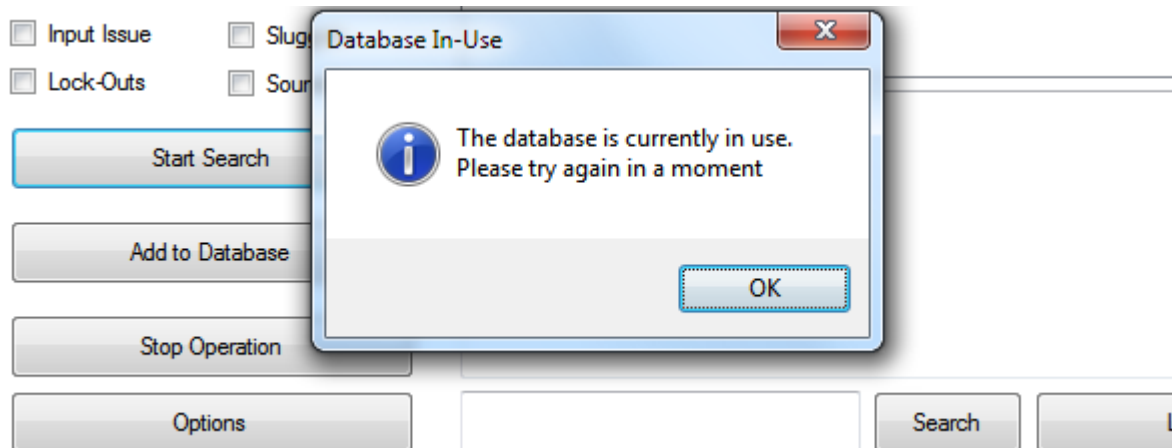The program will continue as normal.

Value Not Valid

The values entered for 'maximum log age' can only be positive whole numbers (including 0 if no limit is to be set).
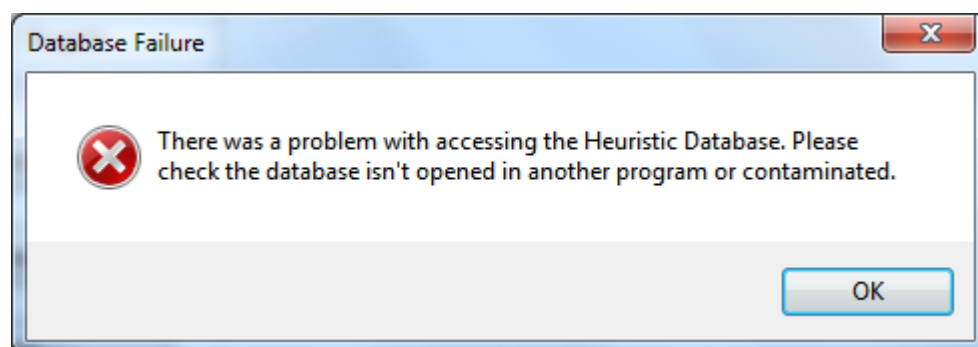
Weighting Not Valid

The values entered for individual weightings can only be positive or negative decimal numbers.

Database In-Use

If there is already a database operation being performed, this message will show. Please wait for the current operation to finish, or click 'Stop Operation' to manually abort it.

Database Failure

This error will display if the database could not be properly read from or written to. The most common cause for the error is if the file is in use by another program that doesn't allow for simultaneous access. Close all programs accessing 'Heuristics.db' and try again.

The other cause is if the database is not formatted correctly. This could be from corruption, or more likely, from a bad rearrangement of fields in the database structure.

The easiest way to fix this is to copy 'Heuristics.db' to a backup location, delete the original file, and re-run the program. A new database will be created.

In order to keep your old data, you should open the backed-up database in a database browsing program, such as SQLite Database Browser (sourceforge.net/projects/sqlitebrowser/). You can then choose to export the database and insert the rows into the new restructured database.
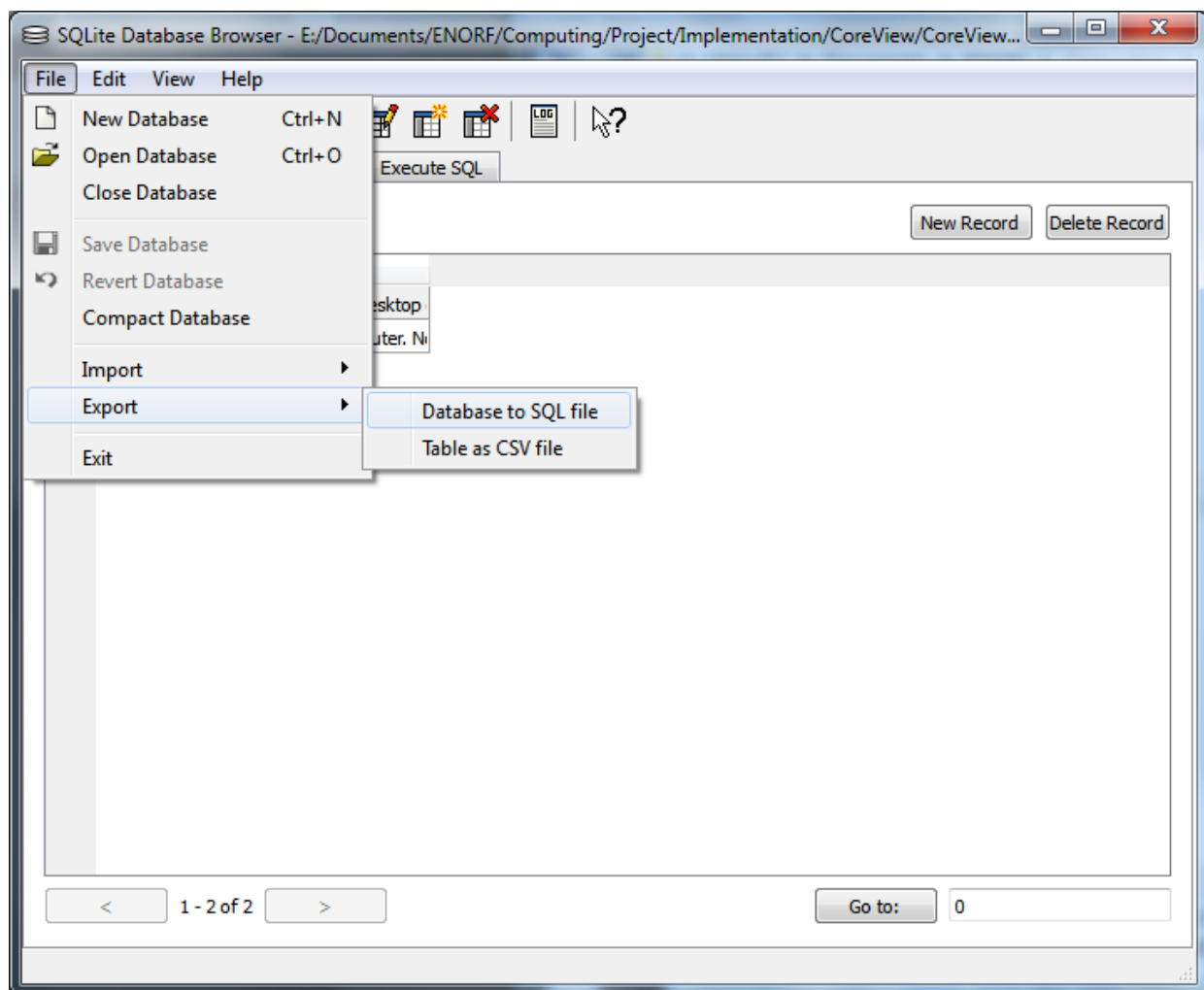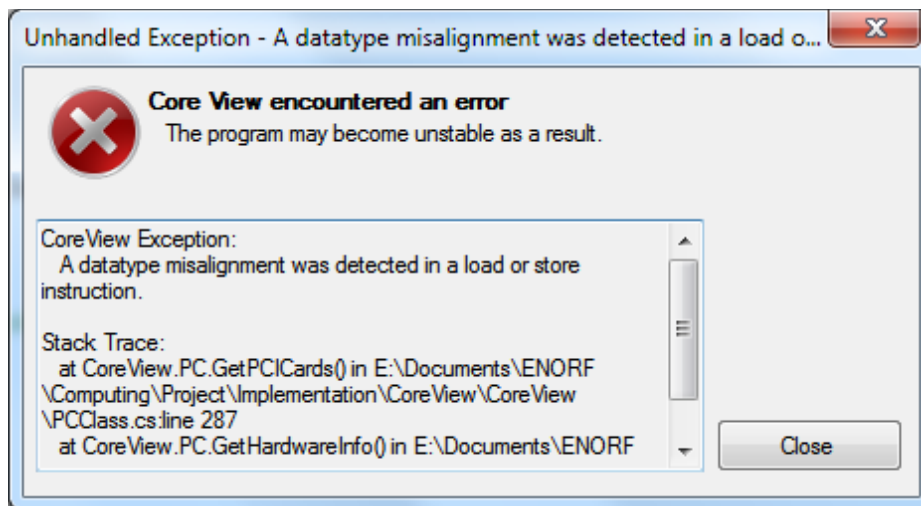
Configuration failure
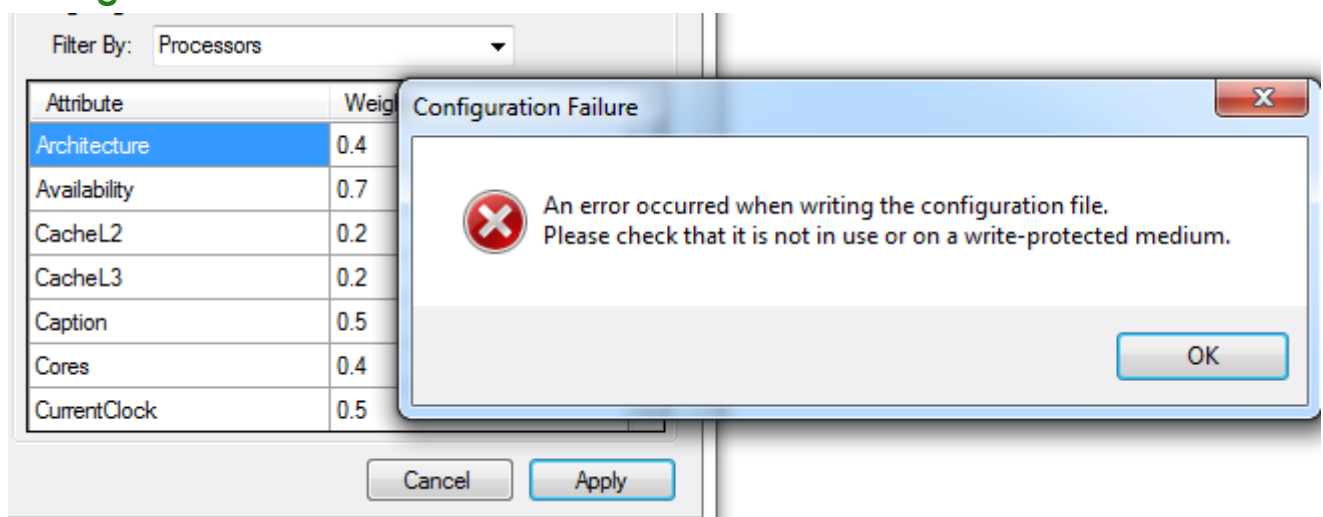
This error can occur if the configuration file could not be saved. It often results from the configuration file being write-protected; so please check that the file isn't in-use or not in a writeable condition.

Unhandled Errors

If Core View discovers an error that it can't handle, it may display a message box like this. This box will try to prevent Core View from crashing unexpectedly, and it will also provide additional information to help fix the issues for later.

If you experience one of these boxes, try to continue as normal, or write an issue report at the Project Home (github.com/CJxD/CoreView/issues).

Frequently Asked Questions

How do I backup/restore the database?
Simply copy the 'Heuristics.db' file in the Core View folder to somewhere else/overwrite the existing one.

For more advanced backups, browse the database with a database browsing program, such as SQLite Database Browser (sourceforge.net/projects/sqlitebrowser/).

How do weightings work?
Whenever one computer is compared to another, is uses the weightings dictionary to compare elements to each other.

Standard weightings are given to each attribute of each class. For example, the Processor class contains 20 attributes and each have a different level of importance when considering possible faults.

The weights set the importance of each of these attributes: the higher the weight, the more important it is. Weights typically range as decimal numbers between 0.0 and 1.0, but they can really be any decimal number allowing for limitless comparison possibilities.

These weights are added to the score for every match that exists.

Furthermore, because each class of elements is of different importance when considering the faults at hand, there is another set of weights for each given complaint. These weights are multiplied to the score to give a more accurate representation depending on the complaints set.

Complaint weights also range typically from 0.0 to 1.0.

Can I use this as an API/resource?

Certainly. Core View has been designed to work as an API if needed. You will need to first compile the source code into a DLL file in Visual Studio, then follow the Wiki pages at the Project Home (github.com/CJxD/CoreView) to access the inbuilt classes.

What's the possibility of this program becoming 'Skynet'?

Interesting question. Due to the way the program is constructed to compare computers, it is essentially a simulated neural network with link weightings.

Because it can be modelled as such, it's possible to let the neural network train itself. This is very easily accomplishable if a method is developed to let Core View calculate a score, and have a human moderator determine how valid the score is.

Because this hasn't been developed though, the probability of Core View becoming an omnipotent artificial intelligence is negligible. If it was developed, it would be possible. If each comparison had its own thread, 'Skynet' becomes very real.