How To

by "Cajun" David Richard

Greetings all. This month, I will be discussing some basic techniques of computer diagnosis and bug tracking. If you own a computer, and I suspect that if you are reading this you do, chances are that you will eventually experience some problems with your computer. Presented here are some general processes that you can go through that will help you iron out these kinks.

You don't have to be a "Computer Guru" to discover what is happening with your machine. Through an orderly process of elimination, anyone can whittle down possibilities to a point where the problem can be solved. The main thing to keep in mind is that computers are blind - they do what you tell them. Don't be afraid of what you do not know about your computer. With a little experimentation you can conquer any problem.

Control your environment and duplicate the bug

One of the most important things to do when looking for bugs is to control your environment. Making haphazard changes will make your bug search more complicated. When you do bug tracking, you need to rule out possibilities. If you add additional software or change your situation when you are looking for a bug, finding it can be more work than necessary.

Another important thing to do is to try and duplicate the bug. This isn't always possible, because some crashes are seemingly random. However, if you can set up a situation where you can make your machine crash consistently, you are well on your way to finding the bug.

INIT's and the memory structure

I will deviate from the topic for a moment to explain the basic structure of the Macintosh. With the introduction of System 7, Apple changed, rather simplified their operating system. You hear many terms - DA's, Extensions, Control Panels, Applications - but how they all fit together is rather simple.

Basically there are three types of software entities: The System Software, INIT's and Applications. The System Software is always present. It allows you to work with your files on your disks and use hardware such as your keyboard and mouse. The important point is that the System Software is always present and is the overseer of the system. Every application on your computer works with the System Software. There is also an application associated with the System Software - you know it as the Finder. The Finder is your link to the System Software, and allows you to control the things the System Software does.

INIT's are files you keep in your Extensions and Control Panels folders. These programs are additions to your System Software (like a luggage rack may be to your station wagon). They are called INIT's because they load in with your System Software at startup time (initialization routines). This is why you need to restart your machine in order for an INIT to take effect. Essentially, they become part of your System software, and should be considered as such.

We are all familiar with applications. You need to launch them when you want to use them and quit them when you are done. DA's are also treated like applications. Applications are given their own memory space to work, similar to a fenced in yard. Applications will normally work within their own area, and don't usually interfere with each other. The Finder is also given it's own memory space. This is best demonstrated in the "About this Macintosh..." dialog box.

OK, quick review - The Finder is an application that is always running and is your link to the System Software (the overseer of the system). INIT's are additions to the System Software. Applications are programs that are similiar to tools that you use when you need them. The main thing to keep in mind about Applications is that they usually do not interfere with each other. Why have I told you all of this? Well, it is good to know how different software entities relate to each other when looking for bugs. The preceding information is all you really need to know. From here on, you will use this information - along with an intelligent trial and error process - to hunt down your problem.

Back to bugs

Now that we have laid a foundation, lets talk about bugs. There are three types of bugs:

- Bugs that "just started happening"
- Bugs that have always been there
- The mysterious crashing bug

Bugs that "just started happening"

This is the most common kind of bug. If you are having a problem that just appeared out of nowhere, think about what you have added to your machine recently. Chances are a new extension or a new system software upgrade could be causing the problem. Keep in mind that a recently added application won't normally cause these types of problems. Applications stay in their own space, whereas INIT's and the System Software affect everything on your computer.

Now, if you have recently added an application - and that application is running when you have your crash - there could be a problem with the application. I will go over this more in a bit.

Bugs that have always been there

Bugs that have always been there are bugs that "just started happening" along time ago but have been ignored or dealt with over time. When you encounter a bug, you can either ignore it, or solve it. Either way, bugs never start out as "having always been there." But, if you never find it and eliminate it, or if you can't get rid of it, it will always be there and it won't go away.

If you can't get rid of a bug right away (like by removing an extension or if it is a System Software bug), there may be a way to work around it, or a way to simply avoid it. These cases are rather inconvenient, but as they say, knowing is half the battle.

The mysterious crashing bug

Now some people will have crashes that don't make sense. You will be working on your machine and it will just up and crash. These types of problems are hard to find because it is usually difficult to duplicate the problem. But if you work with the machine long enough, keeping your software constant (not adding things, changing your controlled environment), you should begin to see a pattern. These random crashes are usually due to a poorly written INIT or maybe something in the System Software. There are ways to discover this, as we will soon find out.

Squishing the sucker

When you are tracking a bug, your tracking needs to be done in an orderly fashion. There are many factors that can contribute. If you remove the factors one at a time, and only one at a time, finding your problem should be easy. Take the time to rule out every factor in a logical manner, and try not to make many changes at once. If you do, you can loose track of what may be causing the problem.

Restart with your INIT's off

If you have ever called a tech support department, more than likely you probably have heard this before. This is the first step because it removes many variables that can cause problems. If you have a bug that happens across the board (like the mysterious crashing bug), it will probably be eliminated by restarting with your INIT's off (restarting with the Shift key held down). Likewise, a crash in an application that is eliminated by doing this is an INIT conflict. This should always be your first step.

If the problem continues, you can be sure it isn't one of your INIT's. It can't be because you have removed them from the equation. If the problem doesn't continue, you can assume that it is one of your INIT's. If this is the case, you will have to find which INIT is causing the problem.

Finding the right INIT

There is an easy way and a fast way to do this. The easy way is to add your INIT's back in your system one at a time (using an extension manager), restarting between the addition of each one and trying to duplicate the bug. This can be time consuming, but it will work. When the bug is duplicated, the last INIT loaded usually is the culprit. You can be certain of this by loading that INIT, and only that INIT, in your system and trying to duplicate the bug.

The other way is called "Divide and Conquer." It follows three simple rules. 1) Load half of the INIT's in the current set. 2) If the bug does not occur, load the other half. 3) If the bug does occur, go to step #1. Dividing your INIT's in half this way is a faster way of seeking out a problem extension, but you need to be a little more organized when you use this method.

Sometimes you can have an INIT that works fine by itself, but will cause problems when another INIT is loaded. INIT's that conflict with each other are a little harder to find, but with an orderly method of testing you can discover under what circumstances you are having the problem. Just make sure to watch closely the effect your changes make on the system.

If you find an INIT that is causing a problem, contact the author and explain to them what you have found. More than likely there will be a newer version available or in the works.

System Software bugs

System Software bugs are a little easier to find, but are usually mistaken for INIT or application problems. The MacOS released by Apple is usually pretty stable, but you can run into circumstances where problems do arise. If you rule out all of your INIT's, and the problem doesn't happen while working in an application, or it happens in all apps, you can bet the problem is with the System Software. What can you do? Wait for the next System Upgrade and try to avoid the situation until then.

Application bugs

These are the easiest to find because the bug only shows when the application is running, and will also occur with the INIT's off. You can usually duplicate the bug pretty consistently. In this situation, find out if you have the latest version of the application by contacting the developer, and upgrade your copy.

Sometimes, the addition of System Software or an INIT will uncover a bug in an application. If this is the case, you probably do not have the latest version of the application. Developers are usually up to date with changes to the System, and if something is changed, it is usually fixed in a later version of the software. The best rule of thumb is to always have the latest version of everything.

In conclusion

Wow, this has turned out to be a rather long article. I hope that this ranting has helped shed a little light on why things go wrong on your System. I also hope that this helps you diagnose your own computer's problems. If you still have trouble, call tech support for Pete's sake! That's what they're there for!

Ed Note - Cajun is very happy that the snow has melted in Rochester. He is no longer restricted to the plush, warm, comfortable automobile that he usually drives around. He can now ride around town on a skinny little bicycle, cheating death around every turn. Cajun calls this "living." If it wasn't for the bike helmet...