

By Code 66

Welcome to Content Checkup

Content Checkup is a DAZ Studio script for DAZ Studio 4.9.3.166 or newer, designed to help artists and content developers alike to understand why some content items give error messages and to avoid distributing unused resources with products.

There are several simple and stupid errors that can be detected by using Content Checkup, below is a sample of what I've found in content from different content creators, different stores and freebies. One important thing to keep in mind though, is that DAZ Studio, rightfully in my opinion, is more rigid in how it treats content that is wrongly setup. Poser will let errors slip, as it tries to locate files by it self when paths are bad or wrongly formed.

List of some of the errors that Content Checker will catch for you:

- *An unused texture map can easily be a texture map that should be used, but one material preset uses the wrong bump or normal map. it's not really visible unless you look closely at the render, can happen, I've seen that in many products.*

- *A texture map that is in the texture map folder but is not used my any item or material preset, also very common.*

- *Poser content that has a path that starts with Runtime instead of :Runtime or /Runtime, which makes the path absolute and not relative, one of those errors Poser seems to ignore and correct by itself.*

- *Content files where a path begins with C:\Runtime instead of :Runtime or /Runtime. This works as long as the content ins installed in C:\, and only on Windows, as Mac OS X doesn't have drive letters, all paths starts with / Volume or /.*

- *Zip files that were zipped on Mac OS X and contain OS X resource proxy files, the zip file hasn't been cleaned with ZipCleaner.*



Keep your content happy!

How does it work?

Content Checkup has three basic modes in which it works. Each mode has a different purpose and are briefly described below.

Zip file analysis

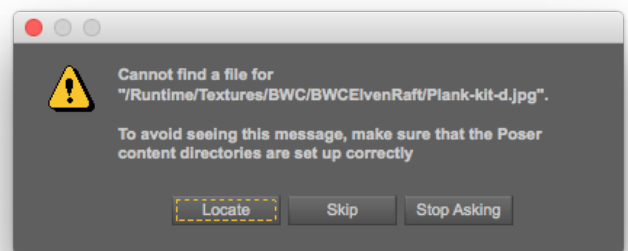
This mode is for analysing a zipped content file. As long as the structure of the zip file is reasonably sane. The zip file should be unzipped and not extracted when this is performed.

Content directory analysis

This mode is for analysing a content directory. This is not meant to analyse your main content library, but a small content library like the one you use for development. The script will start to collect information on the files, but if the operation takes too much time, it will give up and tell you that content directory is too large.

Content file analysis

This mode is to analyse selected item(s) in the content library and is the mode you're most likely to use whenever you stumble upon this dialogue in DAZ Studio.



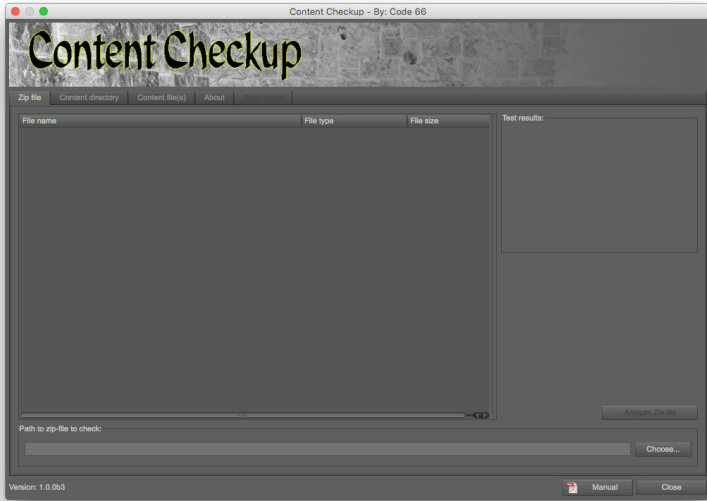
We will use a couple of different products from different vendors and stores in this manual as examples. This is not to cast any discredit on the vendors, I use the items as examples as they are great items that I use myself a lot but they do come with errors, small or big.

As stated before, most of the errors only surface when the item is used in DAZ Studio, and products not made for DAZ Studio usually mean that the content creator never tested the product in DAZ Studio, or the product was created before DAZ Studio existed.

Errors shown in this manual has been deliberately created using my own content to show the problem needed to be shown.

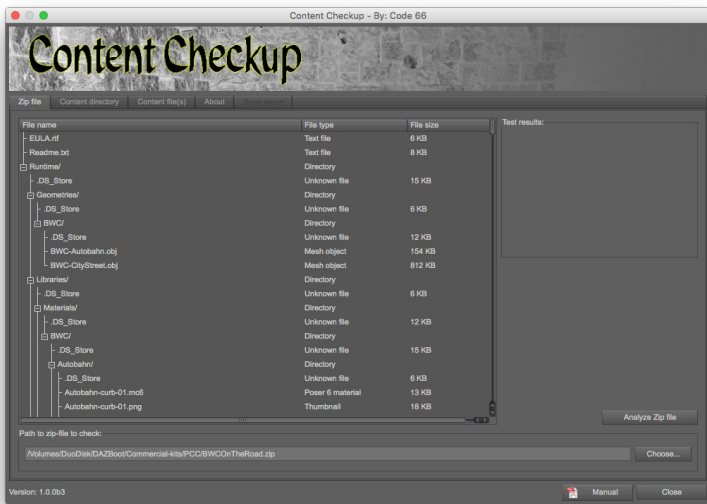
The Zip-file analysis

The Zip-file analysis is the default tab visible when you start Content Checkup.



To select a Zip-file to analyse, click the **Choose...**-button, and the standard OS file chooser will be displayed.

When you have selected a file, and clicked OK, the file will be loaded and parsed and the file structure will be displayed in the Zip file list, showing the structure of the Zip-file, the files, directories, file size and file types. Some files will be displayed as unknown, usually read-me files. Content Checkup really only care about content related files. In this example I opened a Zip-file of an old product of mine, no longer available. I did zip it without running ZipCleaner, but any other errors shown are genuine, are you curious now?



Unused texturemaps:

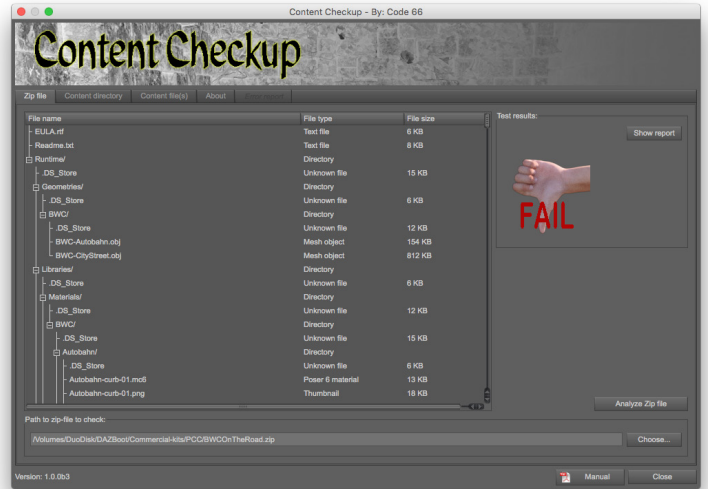
File: Runtime/Textures/BWC/BWCHighway/BWC-highway-lanes-1-OYWS-d.jpg

This zip file from Mac OS X needs to be cleaned with ZipCleaner!

When the Zip-file as been processed, it's time to analyse it. The analysis might take some time, depending on the size of the Zip-file, and mostly, the number of files in it.

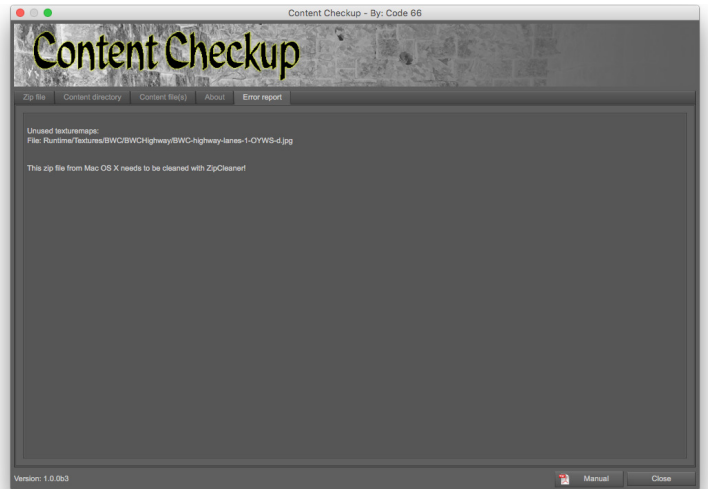
When you click **Analyze Zip file**, the analyser is started and it will show the progress using the standard DAZ Studio progress bar.

When the analysis is done, the test result box will show an OK or a FAIL icon.



In this case the test result was a FAIL, which means that the Zip-file do have content related errors in it.

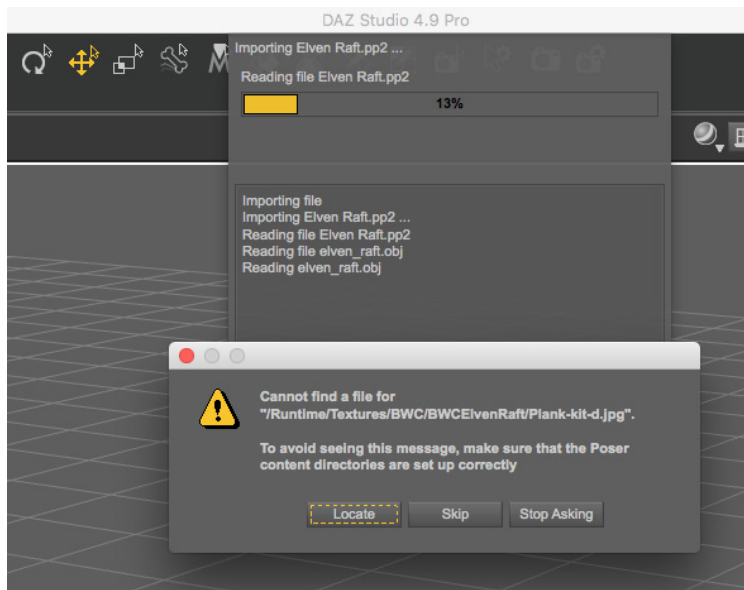
To see the errors, click the **Show report** button in the Test result box.



First, an unused texture map, this most probably mean that one material setup uses the wrong texture map in some material.

Secondly, Content Checkup reports that the file needs to be cleaned by ZipCleaner.

Next up we will use another old set, BWC Elven Raft For Poser. This set is for Poser, and works as expected in Poser, so the problem described here is only related to when using this set in DAZ Studio.



The prop gives an error while loading it.

As I have the Zip-file handy, I give it a spin in the Zip-file analysis.

The results I get are the following:

Files with wrong paths or files are missing:

File: BWCElvenRaft-Poser/Runtime/libraries/Props/BWC/Elven-Raft/Elven Raft.pp2
Path: :Runtime:Textures:BWC:BWCElvenRaft:Plank-kit-d.jpg
The directory '/Runtime/Textures/BWC/BWCElvenRaft/' does not exist!

File: BWCElvenRaft-Poser/Runtime/libraries/Props/BWC/Elven-Raft/Elven Raft.pp2
Path: :Runtime:Textures:BWC:BWCElvenRaft:Plank-kit-b.jpg
The directory '/Runtime/Textures/BWC/BWCElvenRaft/' does not exist!

File: BWCElvenRaft-Poser/Runtime/libraries/Props/BWC/Elven-Raft/Elven Raft.pp2
Path: :Runtime:Textures:BWC:BWCElvenRaft:barrel-dirty-diffuse.jpg
The directory '/Runtime/Textures/BWC/BWCElvenRaft/' does not exist!

File: BWCElvenRaft-Poser/Runtime/libraries/Props/BWC/Elven-Raft/Elven Raft.pp2
Path: :Runtime:Textures:BWC:BWCElvenRaft:barrel-bump.jpg
The directory '/Runtime/Textures/BWC/BWCElvenRaft/' does not exist!

Unused texturemaps:

File: BWCElvenRaft-Poser/Runtime/Textures/BWC/ElvenRaft/barrel-bump.jpg
File: BWCElvenRaft-Poser/Runtime/Textures/BWC/ElvenRaft/barrel-diffuse.jpg
File: BWCElvenRaft-Poser/Runtime/Textures/BWC/ElvenRaft/barrel-dirty-diffuse.jpg
File: BWCElvenRaft-Poser/Runtime/Textures/BWC/ElvenRaft/Plank-kit-b.jpg
File: BWCElvenRaft-Poser/Runtime/Textures/BWC/ElvenRaft/Plank-kit-d.jpg

What does this report tell us?

First there is a list of files with paths that does not exist. They all say that the directory /Runtime/Textures/BWC/BWCElvenRaft/ does not exist.

It also tells us that there are five unused texture maps in the directory Runtime/Textures/BWC/ElvenRaft, and those files have the exact same name as those missing.

Looking in the Zip-file analysis tab again, in the Zip-file content list, we see this:

There is no directory in the Zip-file under textures that is named BWC/BWCElvenRaft, but one named BWC/ElvenRaft The directory has the wrong name in the Zip-file, and the only reason it works in Poser is that Poser will look for the missing files elsewhere, but if the texture map had been

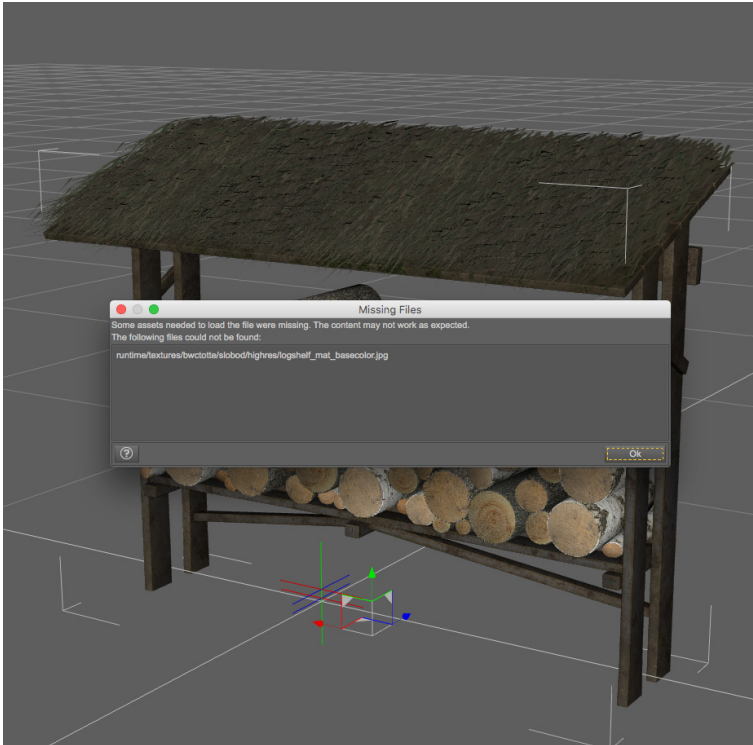
Textures/	Directory	
├ BWC/	Directory	
├ ElvenRaft/	Directory	
├ barrel-bump.jpg	Texturemap	246 KB
├ barrel-diffuse.jpg	Texturemap	556 KB
├ barrel-dirty-diffuse.jpg	Texturemap	561 KB
├ Plank-kit-b.jpg	Texturemap	3.6 MB
├ Plank-kit-d.jpg	Texturemap	5.7 MB

named something like wood1.jpg, and there had been another wood1.jpg texture map, Poser could easily have picked that one instead, something to think about when creating content for Poser.

As you see, there are many small hidden errors in content, no content creator is better or worse than others. But as a content creator, creating content for DAZ Studio, or a DAZ Studio user using content, Content Checkup will help you understand the problems or find them before someone else does.

Content that give errors upon loading

Many have suffered from the dreadful situation with content, that we want to use, making DAZ Studio throw an error dialogue in our faces because there is something wrong with the content or the installation of it.



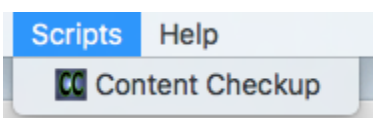
In this example I will be using a rigged content file for one of my sets, the names have been changed to protect the innocent. Loading the content works, but when some of the material options that is available in the Expansion is used in DAZ Studio, an error will be shown.

As this is to simulate a texture expansion, just analysing the zip-file will most likely give lots of errors for references to files in the original set, so zip-file analysis is not the way to go.

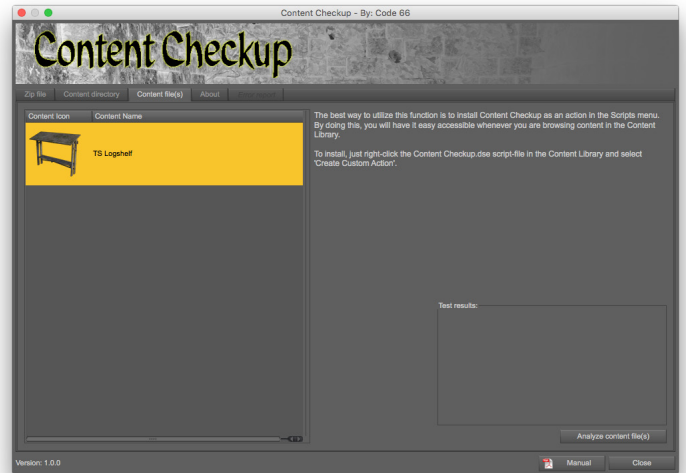
If you installed the base product plus the expansion into a separate content library you could use that option, but this is not what you as a user wants to do just to understand why a specific item does not load correctly.



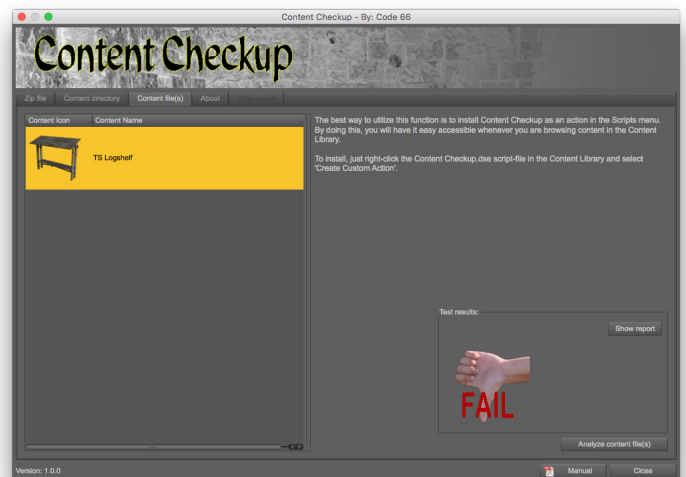
So, we will use the third option available in Content Checkup, the Content file(s) tab. To do this, you first need to browse to the Content Checkup.dse script in the Scripts directory in the Content Library Pane, and there add it to the Scripts Menu. Right-click on the Content Checkup.dse script and select **Create Custom Action**. Now Content Checkup will be in your Scripts menu on the top Menu Bar.



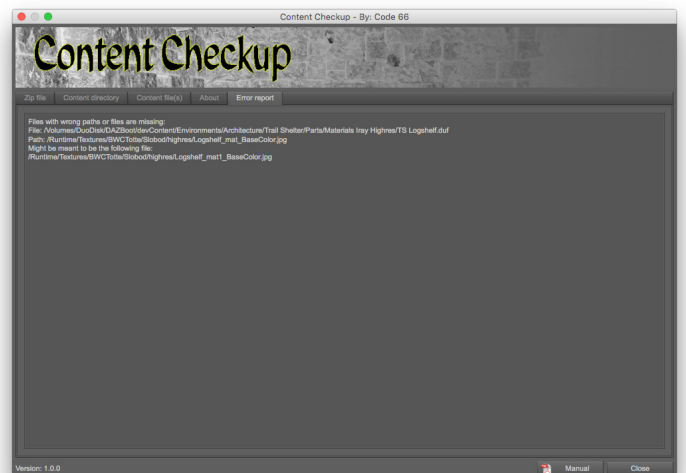
Now, we go the offending content in the Content Library Pane and select the item, or several if you have several items that give you errors when loading. Now, we select Content Checkup from the Scripts menu, and switch to the **Content file(s)** tab.



Here you will see a list of all the content items selected. You can select any of the items in the list and then click the **Analyse Content File(s)** button.



When the script has analysed the selected content files, it will show either OK or FAIL. In case of fail, there will be a button visible named Show report. To show the report, just click that button.



The report might take a while to generate, depending on how many errors there are to display.

In this case, the report will have the following information:

Files with wrong paths or files are missing:

File: /DAZBoot/Environments/Architecture/Trail Shelter/Materials/TS Logshelf.duf

Path: /Runtime/Textures/BWCTotte/Slobod/highres/Logshelf_mat_BaseColor.jpg

Might be meant to be the following file:

/Runtime/Textures/BWCTotte/Slobod/highres/Logshelf_mat1_BaseColor.jpg

So, how do we interpret this information?

First off **Files with wrong paths or files are missing:** will tell you that this section will list files that has internal path-links that points to files, that do not exist at the location to where the content file think they are. That can either mean that the directory pointed to has a different name, or the file is missing.

File: tells you which file, and as you see it is the full path, not just the relative path to the file. Now you know my content sits on a drive named DAZBoot.

Path: is the relative path in that file that points to an non existing file, in this case **Logshelf_mat_BaseColor.jpg**, which was what the error dialogue in DAZ Studio showed in the first place.

Might be meant to be the following file: Content Checkup uses Fuzzy string compare and also in this case where the .ds file is a companion file to a Poser .pz2 file, the .pz2 file to analyse which file could be the correct one, and in this case there is a file named **Logshelf_mat1_BaseColor.jpg** in the texture map directory, which most probably is the correct file.

Now you know what the problem is, now you can either fix it yourself, or report the error to the artist who created the content to get it fixed. Old content, like this set, might not be fixed, so you might have to fix it yourself.

Content directory analysis

Next up, we will look at the third option, **Content directory analysis**. When you switch to the Content directory tab, you will see a list of available and scannable content directories.

Each will show a file count, **not prepared...**, meaning that Content Checkup hasn't done any scanning preparations.

When the preparation is done, it will show how many files are installed in that content directory. If the scanning process exceeds one minute, it will be interrupted and the file count column will say - **timeout** - which means there are too many files in the content directory selected. As said earlier, this option is not designed to run on a large content directory, but on a small one, where you have your product in development.

Now, when the content directory has been prepared, you can select it and click on the **Check content directory** -button. The time of this process is dependent on how many files you have, and of course the speed of your computer and disk.

The error report has the same structure as before, but I prepared a piece of content for this one to show another error, which is a malformed path.

Files with bad paths:

File: Runtime/libraries/pose/Dummy/MAT Files/Chest Green_slime.pz2

Path: C:\Runtime\Textures\BWC\dummy\slime_bump.jpg

This is also a common error in content, that makes it work only on Windows and only if the Runtime directory is located directly on the C:-drive. These errors seems to randomly happen when setting up content, both in Poser and in DAZ Studio, that the a full, not relative path ends up inside the content file.

Conclusion

I really hope you have got the idea on how to use Content Checkup and that you will have a lot of use for it. There are a few things that I know people will ask about, and I will try to close those matters here already.

Q: Why doesn't Content Checkup fix the errors it finds?

A: Because tampering with content files even if you know what you are doing is a risky business. No changes will be made to anything by Content Checkup, it is a read only operation only, nothing is written or changed.

Q: Why doesn't it detect problem X?

A: There can be a number of reasons, but most likely I haven't seen problem X yet so I don't know about it and therefore cannot detect it.

Q: Why does the Zip-file analyser not find the problems in Zip-file Y?

A: See problem X, or the Zip-file is malformed with wrong paths all together, like /Runtime/Texture instead of /Runtime/Textures or /mydata instead of /data. There are millions of ways to create malformed zips.